

86.00 TESIS DE INGENIERÍA ELECTRÓNICA - INFORME DE TESIS

Rediseño e implementación de un Sistema de Aislado Limitado / Total (SAL/T) para trenes

Autor

Anus, Martín manus@fi.uba.ar padron n° 101033

Directores de tesis

Lutenberg, Ariel	alutenb@fi.uba.ar	Director
Gomez, Pablo	pgomez@fi.uba.ar	Co-director
Laiuppa, Adrián	alaiuppa@frbb.utn.edu.ar	Co-director

Jurados

Cruz, Juan Manuel	jcruz@fi.uba.ar
Germino, Santiago	sgermino@fi.uba.ar
Menéndez, Martín	mmenendez@fi.uba.ar

*Este trabajo fue realizado en la Ciudad Autónoma de Buenos Aires,
entre agosto de 2023 y diciembre de 2024.*

Resumen

Este trabajo presenta el diseño e implementación de un prototipo para un sistema de aislado limitado total para formaciones ferroviarias (SAL/T) basado en un pliego de especificaciones de la empresa Trenes Argentinos Operaciones y en una primera versión del prototipo realizado anteriormente en el contexto del Grupo de Investigación en Calidad y Seguridad de las Aplicaciones Ferroviarias (GICSAFe). En esta versión, se agregan nuevas funcionalidades, se actualiza el hardware utilizado y se mejora la portabilidad de su firmware.

Este sistema permite aumentar la seguridad de los trenes ante una falla en alguno de sus sistemas instrumentados de seguridad interfiriendo las señales de corte de tracción y freno de emergencia de la formación y aislando el subsistema en fallo. El SAL/T considera múltiples fuentes de medición de la velocidad de la formación y la comunicación con una central operativa para determinar cómo debe intervenir estas señales críticas para llevar a la formación a un estado seguro de manera controlada.

A lo largo del proyecto, se aplicaron conocimientos de diseño de circuitos, diseño de placas impresas, utilización de protocolos de comunicación e implementación de firmware basado en sistemas operativos en tiempo real.

Agradecimientos

Quiero expresar mi agradecimiento a los directores de esta tesis; Ariel, Pablo y Adrián por su constante apoyo, motivación, ayuda y guía que me brindaron a lo largo de este proyecto.

Al grupo de investigación GICSAFe que me introdujo en el mundo de los sistemas ferroviarios y me abrió la posibilidad de desarrollar este trabajo sobre las sólidas bases de conocimiento del grupo.

A las personas de la empresa Trenes Argentinos que dedicaron su tiempo para definir el objetivo y el alcance de este proyecto.

Al ingeniero Iván Di Vito por dejar una versión anterior de este sistema de una calidad enorme.

A todos los docentes que tuve a lo largo de mi vida por formarme como profesional y como persona.

Al arquitecto Martín Mendilaharzu por la ayuda en el diseño y fabricación del gabinete.

A Carlos, mi papá, por mostrarme la ingeniería en todos los detalles de la vida y contagiarme su pasión.

A Sandra, mi mamá, por su amor incondicional e infinito.

A Federico, mi hermano, por su complicidad y acompañamiento en este camino.

Y por último, pero no menor, agradecerle a todos mis amigos; por estar siempre presentes a mi lado y darle sentido a todo.

Índice

1. Introducción general	1
1.1. Contexto y motivación	1
1.2. Descripción conceptual	2
1.3. Objetivos y alcance	3
1.4. Requerimientos y visión general	4
1.4.1. Requerimientos	4
1.4.2. Sistemas críticos	7
2. Introducción específica	9
2.1. Módulos y componentes	9
2.1.1. STM32 Nucleo-144	9
2.1.2. ESP32 Devkit-C	11
2.1.3. Módulo GPS GY-NEO6MV2	12
2.1.4. Memoria SD	13
2.2. Protocolos de comunicación	15
2.2.1. MQQT	15
2.2.2. Wi-Fi	16
2.2.3. GPS	17
2.2.4. RS-485	18
2.3. Sistema operativo de tiempo real	20
3. Diseño e implementación	21
3.1. Descripción general de la solución	21
3.2. Implementación de hardware	26
3.2.1. Medición de estado de los SIS	26
3.2.2. Aislamiento de los SIS	28
3.2.3. Control de señales críticas	30
3.2.4. Registrador de eventos y conector de zona	32
3.2.5. Llaves de activación modo aislado limitado y total	33
3.2.6. Lectura de velocidad por RS-485	34
3.2.7. Módulo GPS	37
3.2.8. Módulo Wi-Fi y alimentación USB	38
3.2.9. Zócalo para memoria SD	39
3.2.10. Microcontrolador y placa STM32 Nucleo-144	40
3.2.11. Panel frontal	41
3.2.12. Fabricación y ensamblado	44
3.2.13. Gabinete	51
3.3. Implementación de firmware	53
3.3.1. Lógica a nivel general	53
3.3.2. Mecanismos de RTOS	55
3.3.3. Medición del estado de los SIS	57
3.3.4. Interacción con datos GPS	57
3.3.5. Procesamiento de velocidades por RS-485	58
3.3.6. Comunicación con central operativa	58
3.3.7. Lógica de control modo limitado e intermitente	61
3.3.8. Comunicación con controlador LED	63
3.3.9. Registro de eventos	63

4. Ensayos y resultados	65
4.1. Banco de pruebas	65
4.2. Pruebas de hardware	68
4.3. Pruebas de firmware	69
4.4. Pruebas integrales	71
5. Conclusiones	78
5.1. Conclusiones generales	78
5.2. Próximos pasos	79
6. Bibliografía	80

Índice de figuras

2.1. Diagrama en bloques del SAL/T	9
2.2. Placa de desarrollo NUCLEO F429ZI de STMicroelectronics,	10
2.3. Placa de desarrollo ESP32 Devkit-C	11
2.4. Módulo GPS GY-NEO6MV2	13
2.5. Memoria SD	14
2.6. Diagrama de conexión MQTT de ejemplo [28]	15
2.7. Modos de conexión Wi-Fi [30]	16
2.8. Trilateración utilizada en GPS [32]	18
2.9. Bus RS-485 en modo <i>half-duplex</i>	19
2.10. Asignación de tareas por prioridades en un RTOS	20
3.1. Diagrama con interfaces de comunicación del SAL/T	22
3.2. Diseño del panel frontal del SAL/T	23
3.3. Lineas de señales de CT y FE	24
3.4. Diseño de conexión con SIS y señales críticas en el SAL/T	25
3.5. Esquemático del circuito para generación de tensión de alimentación negativa	27
3.6. Esquemático del circuito de medición de los SIS	28
3.7. Funcionamiento de un relé de seguridad [48]	29
3.8. Esquemático del circuito de aislamiento de un SIS	30
3.9. Circuito de relés redundando para activación de señales críticas	31
3.10. Llave interruptor bipolar 20A Elibet 0-1 posiciones	33
3.11. Circuito de lectura del estado de la llave interruptora	34
3.12. Equipamiento Hasler en la formación ferroviaria	34
3.13. Trama emitida por el equipo Hasler Teloc 1500	35
3.14. Principio de funcionamiento de un generador de impulsos ópticos	36
3.15. Circuito conversor entre protocolos UART y RS-485	37
3.16. Circuito de reset del módulo GPS	38
3.17. Módem router Portátil USB 4G LTE Wi-Fi de Altanet	39
3.18. Circuito de conexión del zócalo para memoria SD	40
3.19. Esquemático de la alimentación de la placa STM32 Nucleo-144	40
3.20. Aplicación típica de un controlador AS1115-BSST	42
3.21. Modelos utilizados para las indicaciones luminosas del panel frontal	43
3.22. Circuito del selector de perfil intermitente	43
3.23. Circuito de activación del buzzer sonoro	44
3.24. Pistas y planos de la placa principal del SAL/T	47
3.25. Pistas y planos de la placa secundaria del SAL/T	47
3.26. Modelo 3D de la placa principal del SAL/T	48
3.27. Modelo 3D de la placa secundaria del SAL/T	49
3.28. Placa principal del SAL/T fabricada	50
3.29. Placa secundaria del SAL/T fabricada	51
3.30. Diseño de gabinete para impresión 3D	51
3.31. Gabinete acondicionado para el prototipo del SAL/T	52
3.32. Frente del del prototipo del SAL/T	53
3.33. Lógica general del firmware del SAL/T	54
3.34. Diagrama de flujo de modos de operación del SAL/T	55
3.35. Diagrama de la comunicación entre el SAL/T y la central operativa	60
3.36. Perfil de velocidad de una formación en modo intermitente	63
4.1. Banco de pruebas completo del SAL/T	65
4.2. Placa de desarrollo Arduino Uno	66
4.3. Módulo conversor UART a RS-485	67
4.4. Visualización de shifr.io conectando el SAL/T con la central operativa	68

4.5. Conexiones para prueba de comunicación con ESP32 y servidor MQTT	70
4.6. Módulo SD para pruebas de comunicación	70

Índice de tablas

1. Diferencia de tensión entre los terminales de un SIS	26
2. Posición de relés para los estados del circuito redundado	32
3. Listado de componentes del SAL/T	45
4. Restricciones de diseño utilizadas para el diseño del PCB	47
5. Identificación de <i>topic</i> MQTT	59

1 Introducción general

En este capítulo se presenta el contexto, la motivación, los objetivos, el alcance y una descripción conceptual del proyecto. También se exponen sus requerimientos y una breve introducción a los sistemas críticos.

1.1 Contexto y motivación

El Grupo de Investigación en Calidad y Seguridad de las Aplicaciones Ferroviarias (GICSAFe) fue creado en 2017 en el marco del Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET) de la República Argentina [1]. En el grupo participan investigadores, docentes y alumnos de diferentes instituciones públicas argentinas quienes realizan desarrollos de sistemas electrónicos e informáticos para aplicaciones ferroviarias relacionadas con la seguridad. Muchos de los prototipos desarrollados son instalados y entregados junto a toda la documentación, respetando normas internacionales de seguridad, para luego transferir el derecho de uso, fabricación y mantenimiento a los clientes.

La empresa estatal Trenes Argentinos Operaciones [2] mencionó en reiteradas oportunidades la disconformidad con la situación actual de las formaciones ferroviarias frente a algún fallo en los Sistemas Instrumentados de Seguridad (SIS) porque, al entrar alguno de estos en fallo, la formación se detiene y su tracción se bloquea aún si está lejos de una estación, en un lugar inseguro para que los pasajeros desciendan. En la práctica, se deshabilitan todos los SIS de la formación y se conduce hasta la próxima estación para descender a los pasajeros. Esto presenta una situación de riesgo porque la formación pasa a depender fuertemente del criterio del conductor sin ningún sistema de seguridad monitoreando y asistiendo la conducción [3].

Al mismo tiempo, la empresa no encuentra alternativas comerciales para satisfacer las necesidades de un Sistema de Aislado Limitado / Total, que permita deshabilitar el bloqueo generado por algún SIS en particular, ya que necesita integrar distintos SIS y fuentes de medición que no son necesariamente del mismo fabricante ni con especificaciones genéricas que permitan su fácil integración.

Por estos motivos, Trenes Argentinos solicitó en el año 2017 al grupo CONICET-GICSAFe que diseñara un SAL/T a medida para cumplir con los requerimientos específicos de sus formaciones. El ingeniero Iván Di Vito, en ese momento estudiante de la carrera de Ingeniería Electrónica en la UBA y allegado al grupo de investigación, tomó el proyecto que finalizó en el año 2019 con la entrega de un prototipo funcional del SAL/T luego de varias pruebas de campo en los talleres de Trenes Argentinos junto a toda la documentación requerida por las normas de seguridad ferroviarias para este tipo de desarrollos [4]. El trabajo realizado incluyó el relevamiento de los requerimientos del proyecto, el desarrollo conceptual de la solución propuesta, el diseño e implementación del hardware y firmware, la puesta en marcha y realización de pruebas de campo de un prototipo funcional. El mismo se realizó de acuerdo a la norma UNE-EN 50126 [5], aplicando técnicas de patrones de diseño de hardware y software [6]. Este trabajo fue publicado en el Congreso Argentino de Sistemas Embebidos (CASE) en el año 2019 [7] y en IEEE en 2020 [3].

Sin embargo, antes de requerir una versión productiva y reproducible en cantidades, se determinó que faltaban implementar algunas funcionalidades adicionales para que se pueda instalar un SAL/T en las formaciones. El pliego de especificaciones técnicas para la versión completa del SAL/T escrito por la gerencia de seguridad operacional de Trenes Argentinos se encuentra disponible en [8].

1.2 Descripción conceptual

En el 2019, el ingeniero Iván Di Vito presentó un prototipo de este sistema [4] con pruebas de campo en los talleres de Trenes Argentinos [2] siguiendo las fases y recomendaciones de la norma UNE-EN 50126 [5]; “Aplicaciones ferroviarias. Especificación y demostración de la fiabilidad, la disponibilidad, la mantenibilidad y la seguridad (RAMS)”. Este trabajo fue desarrollado con el acompañamiento del Grupo de Investigación en Calidad y Seguridad de las Aplicaciones Ferroviarias (GICSAFe) [1] como trabajo final de grado de Ingeniería Electrónica en la Universidad de Buenos Aires teniendo como directores al Dr. Ing. Ariel Lutenberg y al Dr. Ing. Pablo Gomez.

En este trabajo se busca hacer un rediseño del prototipo del SAL/T implementando algunas mejoras sugeridas por Trenes Argentinos en el pliego de especificaciones técnicas ET.SO.No 046/18-E3 [8] del 2022, utilizando un microcontrolador y componentes más modernos y un firmware basado en un sistema operativo de tiempo real que resulte más mantenible que el firmware de la primera versión del SAL/T desarrollada por Ivan Di Vito.

Las formaciones ferroviarias cuentan con varios Sistemas Instrumentados de Seguridad (SIS) a bordo con el objetivo de supervisar el correcto funcionamiento de los subsistemas críticos, como por ejemplo, la protección de coche a la deriva, el sistema de hombre vivo o la seguridad de puertas [3].

Estos sistemas están diseñados para que, ante la detección de una falla crítica, se produzca la detención de la formación de manera inmediata. Esto se logra controlando las señales de Corte de Tracción (CT) y de Freno de Emergencia (FE) [3]. En esta situación, la formación puede desactivar todos los SIS bajo estricta supervisión y continuar desplazándose con el control manual del conductor hasta una estación cercana para descender a los pasajeros y luego a un taller para ser reparada.

El Sistema de Aislado Limitado/Total, por sus siglas SAL/T, es un equipo que permite una circulación controlada y más segura de la formación cuando uno de los SIS se encuentra en falla. Al activar el Modo Aislado Limitado (MAL), el SAL/T va a deshabilitar solamente el SIS que está en fallo y controlar las señales de CT y FE para poder continuar con la circulación de la formación. Para hacerlo de manera segura, el SAL/T va a contar con múltiples fuentes de medición de la velocidad de la formación y va a activar las señales de CT y FE para frenarla en caso de que se excedan los límites estipulados. En el Modo Aislado Total (MAT), se libera la velocidad de precaución y puede ser aplicado solamente por personal superior en el caso que la formación se encuentre sin pasajeros y muy alejada del centro reparador. Ambas activaciones deben ser registradas dentro del equipo así como también cualquier falla detectada, cambio de configuración en las velocidades permitidas o cualquier evento significativo para la seguridad de la formación [3].

Además, el SAL/T cuenta con otras funcionalidades como la comunicación con el conductor a través de una interfaz donde se muestra el estado de los SIS, la velocidad medida en todo momento y un indicador de la zona donde se encuentra la formación. En caso de no contar con una medición de velocidad válida y activar el MAL, el SAL/T entra en Modo Intermitente (MI) y controla la formación activando y desactivando la tracción y el freno acorde a los perfiles de tiempos preconfigurados según las características de la formación para no exceder ciertas velocidades de precaución. Desde el panel formal, el conductor puede seleccionar distintos perfiles intermitentes. También, el SAL/T tiene la posibilidad de conectarse con una central operativa para informar su estado y recibir remotamente instrucciones [3].

Este equipo es considerado un sistema crítico porque, en caso de fallar, puede conllevar pérdidas de vidas humanas o daños importantes tanto de propiedades como del entorno [9].

Los cambios más relevantes mencionados en el pliego de especificaciones técnicas que dan lugar a rediseñar el prototipo antes de producirlo en masa son los siguientes:

- Utilización de un microcontrolador de mayor capacidad y más vigente.
- Generación de un firmware más mantenable basado en un sistema operativo de tiempo real.
- Aislamiento del SIS en falla en Modo Aislado Limitado sin interrumpir el funcionamiento del resto de los SIS.
- Selección de distintos perfiles intermitentes desde el panel frontal del SAL/T.
- Conector de zona para poder determinar distintas zonas de tránsito para la formación y activar cambios en otros sistemas de la formación.
- Registro interno de eventos para cambios de estado de SIS, activación de modos, excesos de velocidad y activación de señales de Corte de Tracción y Freno de Emergencia.

1.3 Objetivos y alcance

Se busca diseñar una solución para aumentar la seguridad en las formaciones ferroviarias ante fallas en los Sistemas Instrumentados de Seguridad mediante un mejor control y monitoreo de velocidad de la formación en tiempo real. La solución debe ser compatible con los mecanismos de seguridad y sistemas instalados actualmente en las formaciones de Trenes Argentinos. Los principales objetivos que se persiguen son:

- Rediseñar el sistema SAL/T a partir de los trabajos existentes y el pliego de especificaciones técnicas propuestos por Trenes Argentinos.
- Rediseñar el hardware del SAL/T utilizando componentes vigentes en el mercado.
- Generar un firmware portable basado en un sistema operativo de tiempo real.
- Construir un prototipo del proyecto y realizar pruebas de funcionamiento en un entorno similar al real.
- Favorecer el desarrollo de tecnología nacional.

El proyecto consiste en diseñar, desarrollar e implementar un nuevo prototipo del SAL/T teniendo en cuenta que se deben agregar nuevas funcionalidades requeridas, que el hardware debe ser actualizado con componentes vigentes y disponibles actualmente en el mercado internacional y que el firmware desarrollado debe estar basado en un sistema operativo de tiempo real que resulte portable.

Para establecer los requerimientos del proyecto, se contempló el prototipo preexistente del SAL/T y el pliego de especificaciones técnicas que generó Trenes Argentinos teniendo en consideración las limitaciones de tiempos y presupuesto que hace sentido para un proyecto de fin de carrera de grado.

Una vez finalizado el diseño del sistema, el prototipo debe ser fabricado sobre una placa PCB y se deben montar todos los componentes para su correcto funcionamiento. Se deberán realizar pruebas de laboratorio en las condiciones de funcionamiento para verificar que se cumplan todos los requerimientos establecidos.

1.4 Requerimientos y visión general

1.4.1 Requerimientos

A continuación, se muestran los requerimientos acordados para el proyecto contemplando la versión anterior del SAL/T y el pliego de especificaciones técnicas generado por Trenes Argentinos como referencia.

1. Grupo de requerimientos asociados con la interfaz humano-máquina
 - 1.1 La interfaz debe contar con una llave rotativa para activar el modo aislado limitado.
 - 1.2 La interfaz debe mostrar la velocidad media del equipo en km/h con 4 dígitos.
 - 1.3 La interfaz debe indicar el estado de la señal de corte de tracción.
 - 1.4 La interfaz debe indicar el estado de la señal de freno de emergencia.
 - 1.5 La interfaz debe indicar la presencia de un comando remoto de la central operativa.
 - 1.6 La interfaz debe indicar el estado del módulo GPS.
 - 1.7 La interfaz debe indicar el estado de la alimentación.
 - 1.8 La interfaz debe indicar el modo en el que se encuentra el sistema.
 - 1.9 La interfaz debe contener un control local por fuera del panel frontal para activar el modo aislado total.
 - 1.10 La interfaz debe indicar con un LED RGB en cuál de las 3 zonas geográficas predefinidas se encuentra el material rodante.
 - 1.11 La interfaz debe indicar el estado de los distintos Sistemas Instrumentados de Seguridad conectados al SAL/T con un LED RGB para cada SIS.
 - 1.12 La interfaz debe contar con una botonera o switch que permita cambiar el perfil para modo de tracción intermitente.
 - 1.13 La interfaz debe indicar el perfil seleccionado para modo de tracción intermitente.
 - 1.14 La interfaz debe contener un buzzer interno para indicaciones sonoras.
2. Grupo de requerimientos asociados a la comunicación con el registrador de eventos
 - 2.1 El sistema debe informar al registrador de eventos la activación del modo aislado limitado.
 - 2.2 El sistema debe informar al registrador de eventos si la alimentación es correcta.
 - 2.3 El sistema debe informar al registrador de eventos la activación del freno de emergencia.
 - 2.4 El sistema debe informar al registrador de eventos la activación del corte de tracción.
3. Grupo de requerimientos asociados al registro de datos
 - 3.1 El sistema debe mantener un registro de datos local donde se registran los siguientes eventos con marca de tiempo:
 - Activación/Desactivación del modo aislado limitado
 - Activación/Desactivación del modo aislado total (local o remota)
 - Referencia de velocidad adoptada y sus permutaciones
 - Cobertura de señal GPS
 - Ciclos de permiso, corte y freno en modo de tracción intermitente
 - SIS en fallo
 - Velocidad desarrollada
 - Zona de circulación

- 3.2 El sistema debe permitir descargar los datos registrados mediante una interfaz serie.
 - 3.3 El sistema debe poder descargar los datos registrados de manera remota.
4. Grupo de requerimientos asociados a la comunicación con la central operativa
 - 4.1 El sistema debe informar periódicamente (con un tiempo configurable) su estado a la central operativa a través de la red de datos Wi-Fi.
 - 4.2 Debe existir la posibilidad de usar 2 proveedores distintos de datos de manera simultánea.
 - 4.3 Debe existir la posibilidad de conectarse alternadamente a más de una red Wi-Fi.
 - 4.4 El protocolo de comunicación con la central operativa debe ser MQTT.
 - 4.5 El sistema debe ser capaz de recibir un comando remoto que anule el corte de tracción y el freno de emergencia bajo cualquier condición (modo aislado total).
 - 4.6 El sistema debe ser capaz de recibir un comando remoto que active el corte de tracción y el freno de emergencia bajo cualquier condición (modo parada total).
 - 4.7 El sistema debe ser capaz de recibir un comando remoto que active el corte de tracción y anule el freno de emergencia bajo cualquier condición (modo coche en deriva).
 - 4.8 El sistema debe ser capaz de recibir un comando remoto que active el corte de tracción y el freno de emergencia de forma intermitente en ciclos de tiempo configurables (modo intermitente).
 - 4.9 El sistema debe ser capaz de recibir un comando remoto que cancele cualquier comando remoto vigente.
 - 4.10 El sistema debe ser capaz de recibir comandos remotos que modifiquen sus parámetros internos configurables.
 - 4.11 Si no se recibe un nuevo comando remoto luego de un tiempo configurable (por defecto 10 segundos, máximo 1 minuto), debe volver al algoritmo de activación de corte de tracción y freno de emergencia por defecto.
 - 4.12 Ante un comando remoto recibido, debe enviar una confirmación de recepción que permita a la central operativa decidir si es necesaria o no una retransmisión.
 - 4.13 Debe utilizar algún mecanismo de encriptación para el enlace con la central operativa.
5. Grupo de requerimientos asociados al modo normal de funcionamiento
 - 5.1 El modo normal el sistema no debe intervenir en el funcionamiento del material rodante (prioridad alta).
 - 5.2 El sistema debe obtener en todo momento la mejor estimación posible de la velocidad de la formación.
 - 5.3 Debe ser capaz de recibir la velocidad a partir de una señal digital provista por el registrador de eventos Hasler Teloc 1500.
 - 5.4 Debe ser capaz de calcular la velocidad a partir de un sistema GPS integrado.
 - 5.5 Debe ser capaz de calcular la velocidad a partir de un generador de impulsos.
 - 5.6 El rango de velocidad soportado por el sistema tiene que estar entre 0 y 120 km/h.
 - 5.7 La estimación de velocidad debe tener una precisión del 2 % de fondo de escala.
 - 5.8 El sistema debe determinar la posición del material rodante para activar un contacto seco de relé según la zona de operación donde se encuentre.
6. Grupo de requerimientos asociados al modo aislado limitado

- 6.1 En modo aislado limitado el sistema debe evitar la aplicación del corte de tracción.
- 6.2 En modo aislado limitado el sistema debe evitar la aplicación del freno de emergencia.
- 6.3 Ante cualquier error interno, el sistema debe dejar de intervenir en la aplicación del corte de tracción.
- 6.4 Ante cualquier error interno, el sistema debe dejar de intervenir en la aplicación del freno de emergencia.
- 6.5 En modo aislado limitado el sistema debe emitir una señal sonora intermitente a través de un buzzer.
- 6.6 En modo aislado limitado el sistema debe evitar que la velocidad del material rodante supere una serie de límites configurados.
 - 6.6.1 Si al pasar de modo normal a modo aislado limitado no se cuenta con una estimación de velocidad, debe activar el corte de tracción y el freno de emergencia por 30 segundos.
 - 6.6.2 Si se supera una velocidad configurable (por defecto 30 km/h), debe activar el corte de tracción y emitir una señal sonora continua a través de un buzzer.
 - 6.6.3 Si se supera una velocidad configurable (por defecto 36 km/h), debe activar el freno de emergencia.
 - 6.6.4 Una vez aplicado, el corte de tracción debe dejar de aplicarse si la velocidad vuelve a ser menor a una velocidad configurable (por defecto 25 km/h).
 - 6.6.5 Una vez aplicado, el freno de emergencia sólo debe dejar de aplicarse luego de un tiempo configurable (por defecto 30 segundos) desde que se superó el límite.
 - 6.6.6 Si la lectura de velocidad es inválida, debe activar y desactivar el corte de tracción y freno de emergencia de manera alternada en ciclos de tiempo según perfiles preconfigurados (modo de tracción intermitente).
- 6.7 El sistema debe interactuar con 5 Sistemas Instrumentados de Seguridad.
- 6.8 El sistema debe ser capaz de identificar qué SIS está fallando y aislar solamente este dejando los demás SIS operativos.
- 6.9 El sistema debe medir la posición del material rodante para modificar las velocidades máximas según la zona de operación donde se encuentre.

7. Grupo de requerimientos asociados al modo aislado total

- 7.1 El sistema solo puede acceder a este modo accionando el control local o mediante comando remoto.
- 7.2 En el modo aislado total el sistema debe liberar la velocidad de circulación.
- 7.3 El sistema debe registrar en el registro de datos la activación de este modo.

8. Grupo de requerimientos asociados al modo de tracción intermitente

- 8.1 El sistema solo deberá acceder a este modo cuando se encuentre en modo aislado limitado y no cuente con ninguna medición de velocidad válida o se seleccione de manera remota.
- 8.2 El sistema debe permitir la configuración remota de distintos perfiles de operación de manera remota en donde se configuran los valores de Tiempo del permiso de Tracción, Tiempo de ciclo en deriva, Cantidad de ciclos antes de aplicar freno y Tiempo de espera antes de comienzo de nuevo ciclo.

9. Grupo de requerimientos asociados al hardware y al gabinete

- 9.1 El sistema utilizará una fuente externa de 5 V de tensión continua que pueda soportar hasta 25W de potencia.
- 9.2 Los conectores deben estar bien identificados para facilitar la correcta conexión.
- 9.3 El sistema debe poseer una placa de circuito impreso con el procesador y periféricos necesarios para el procesamiento de las señales del material rodante y una segunda placa para los elementos necesarios para la interacción con el conductor.
- 9.4 El diseño de la placa de circuito impreso debe contemplar las recomendaciones de la norma IPC 2221 [10] a fines de asegurar la manufacturabilidad del equipo.

10. Grupo de requerimientos asociados al desarrollo del firmware

- 10.1 El desarrollo del software debe contemplar las recomendaciones de las normas UNE-EN 50126 y UNE-EN 50128 [11].
- 10.2 El sistema debe trabajar con un sistema operativo en tiempo real para mejorar el desempeño y la seguridad del equipo.

Estos requerimientos permiten definir de manera clara las funcionalidades, comportamientos y especificaciones de diseño del sistema. Es fundamental tener bien definidos los requerimientos en las primeras etapas del proyecto e iterarlos con todos los involucrados hasta obtener una versión definitiva. Cualquier modificación realizada en fases más avanzadas puede provocar un gran desvío en la planificación y generar un esfuerzo considerable de cambio.

1.4.2 Sistemas críticos

Para comprender el contexto de diseño del proyecto, es necesario definir y presentar primero qué son los sistemas críticos. Los sistemas críticos son aquellos cuyo fallo puede resultar en consecuencias negativas significativas, como daño a las personas, el entorno o a la infraestructura [9]. En el contexto de los sistemas para trenes, estos deben cumplir con estrictos requisitos para garantizar una respuesta adecuada en situaciones de emergencia. Para lograr estos objetivos, se deben seguir las normas EN-50126 y EN-50129 [12]. La EN-50126 establece directrices para la fiabilidad, disponibilidad, mantenibilidad y seguridad (RAMS) de los sistemas ferroviarios, mientras que la EN-50129 se enfoca en la seguridad de los sistemas de control y protección. Ambas normas proporcionan parámetros de medición y procesos de diseño destinados a reducir los riesgos y asegurar que los sistemas ferroviarios operen de manera segura y eficiente durante su ciclo de vida.

Para los sistemas críticos, es importante considerar las siguientes características:

- **Fiabilidad:** La capacidad del sistema para operar correctamente durante un período especificado. Esto abarca la prevención de fallos y la capacidad para recuperar el funcionamiento normal en caso de que ocurran fallos.
- **Seguridad:** Asegura que el sistema operará sin causar daño a personas o propiedades, cumpliendo con todas las normas y regulaciones relevantes.
- **Disponibilidad:** El sistema debe estar operativo y disponible para su uso cuando sea necesario, minimizando el tiempo de inactividad.
- **Tolerancia a fallos:** Implementación de mecanismos para detectar y manejar fallos sin interrumpir la operación del sistema. Esto puede incluir redundancias y métodos de recuperación.
- **Tiempo real:** Capacidad del sistema para responder a eventos o comandos en un tiempo definido, esencial para la operación en situaciones de emergencia.

Los sistemas críticos deben estar diseñados para gestionar emergencias de manera efectiva, lo que incluye la detección rápida de situaciones de riesgo inesperadas y la notificación inmediata al personal responsable. Para lograr esto, se implementan sistemas de alerta y comunicación que permiten actuar con rapidez ante cualquier incidente. Además, se emplean diversas técnicas de diseño enfocadas en aumentar la seguridad, entre las cuales destaca la incorporación de sistemas redundantes que garantizan la continuidad de la operación, incluso si algún componente falla. En este contexto, cobra relevancia el concepto de *fail-safe* o fallo seguro, que asegura que cualquier error o falla en el sistema resulte en un estado seguro, evitando que dicha falla provoque problemas adicionales en otros componentes o sistemas relacionados.

Las fallas de un sistema de seguridad se pueden clasificar de la siguiente manera [13]:

- **Fallas aleatorias:** Son aquellas que ocurren sin un patrón predecible y son inherentemente impredecibles en términos de momento y causa. Estos fallos suelen estar asociados con la naturaleza estadística de la vida útil de los componentes.
- **Fallas sistemáticas:** Son causadas por defectos en el diseño, la fabricación o el proceso de integración del sistema. Estos fallos son predecibles si se conocen las debilidades en el diseño o el proceso.
- **Fallas de causa común:** Son aquellas que afectan a varios componentes o sistemas debido a una única causa raíz. Estos fallos pueden ser resultado de una debilidad en el diseño, un defecto en el proceso de fabricación, o una condición operativa que impacta a múltiples partes del sistema. Muchas veces está vinculado a eventos ambientales como inundaciones, tormentas u otros.

Los sistemas de seguridad pueden cambiar a distintos estados dependiendo de cómo sea el manejo de una falla. Estos estados muestran un compromiso entre la seguridad del sistema y su disponibilidad [9].

- **Estado normal:** En este estado no hay ninguna falla presente en el sistema. El sistema se encuentra disponible y operando con todas las medidas de seguridad habilitadas. Este es el estado deseado y el más seguro.
- **Estado seguro:** Existe una falla en el sistema que se manejó de forma segura tras la activación de alguno de los mecanismos de seguridad. El sistema entra en estado seguro para evitar daños adicionales. El sistema no se encuentra disponible y se debe reparar el sistema para retornar al estado normal.
- **Estado intermedio:** Existe una falla que está siendo controlada por alguna de las funciones de seguridad y permite que el sistema siga operativo. Las funciones de seguridad pueden ser realizadas ante nuevas fallas, pero el sistema funciona con limitaciones y mayor riesgo. En este estado, el sistema debe ser reparado para recuperar su grado de seguridad normal y evitar fallas adicionales.
- **Estado peligroso** Existe una falla y las funciones de seguridad no pueden ser activadas. El sistema está disponible, pero no protegido lo que implica un riesgo de operación mucho más alto al establecido. Este estado es el más crítico y se deben tomar medidas de emergencia inmediatas para restaurar la seguridad y minimizar el riesgo.

2 Introducción específica

En este capítulo se presentan los conceptos fundamentales necesarios para comprender el trabajo realizado, así como los módulos de hardware clave utilizados durante el desarrollo del sistema.

2.1 Módulos y componentes

La figura 2.1 muestra un diagrama en bloques del sistema, el cual proporciona una visión general que facilita la comprensión de los conceptos y módulos introducidos a lo largo de este capítulo.

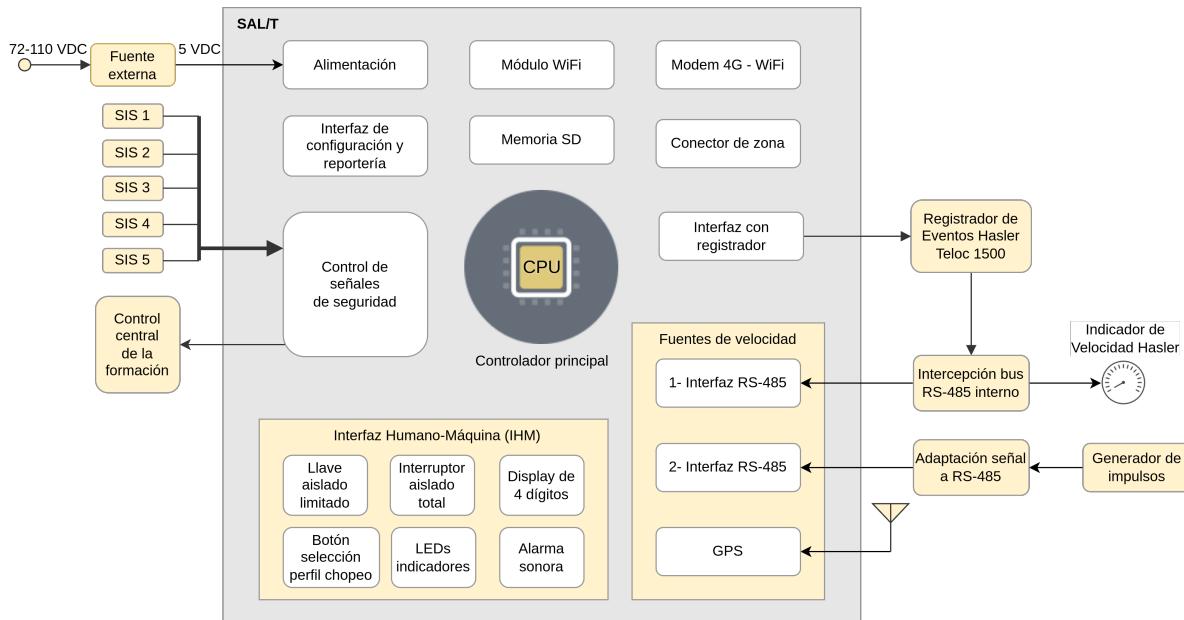


Figura 2.1: Diagrama en bloques del SAL/T

2.1.1 STM32 Nucleo-144

La STM32 Nucleo-144 es una familia de placas de desarrollo basada en los microcontroladores STM32 de la familia ARM Cortex-M, diseñada para facilitar la creación de prototipos y el desarrollo de sistemas embebidos [14]. Esta familia de placas es parte de la serie Nucleo de STMicroelectronics, conocida por ofrecer soluciones flexibles y accesibles para una amplia gama de aplicaciones, desde dispositivos IoT (*Internet of Things*) hasta sistemas críticos. En la figura 2.2 se puede visualizar la placa de desarrollo NUCLEO F429ZI de la familia STM32 Nucleo-144 que se utiliza en este proyecto.

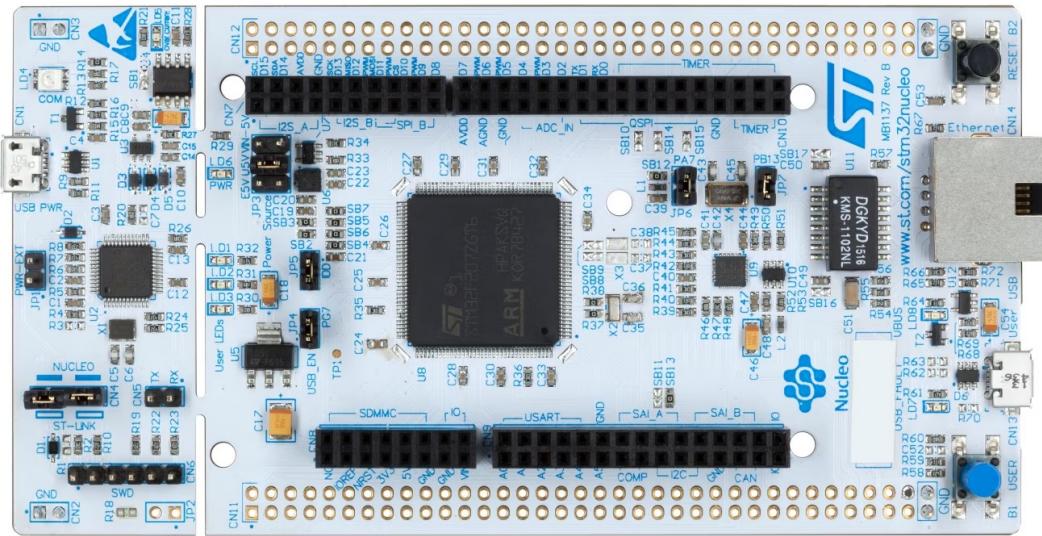


Figura 2.2: Placa de desarrollo NUCLEO F429ZI de STMicroelectronics,

La NUCLEO F429ZI se caracteriza por su gran versatilidad y un conjunto de características técnicas que la hacen adecuada para proyectos que requieren un rendimiento alto y conectividad avanzada. Entre sus especificaciones más destacadas se incluyen:

- **Microcontrolador:** está equipada con un microcontrolador basado en la arquitectura ARM Cortex-M que ofrece un excelente rendimiento junto con una amplia gama de interfaces de conectividad y capacidades avanzadas de procesamiento en tiempo real. La NUCLEO F429ZI incluye el microcontrolador STM32F429ZI [15] tiene un núcleo ARM Cortex-M4 que corre ST-LINK/V2-1a 180 MHz con unidad de punto flotante (FPU), memoria flash de 2 MB y RAM de 256 KB.
- **Conectividad:** dispone de múltiples interfaces de comunicación, como UART, I2C, SPI, USB, Ethernet y CAN, lo que la convierte en una opción idónea para sistemas que requieren interconexión de múltiples dispositivos y la transmisión de datos en tiempo real.
- **Pines:** incluye un conector de 144 pines, lo que ofrece una gran cantidad de líneas de entrada/salida (GPIO) para controlar y conectar diversos módulos y periféricos. Además, esto permite acceso a todos los pines del microcontrolador lo que permite mayor flexibilidad para la asignación de sus funciones.
- **Programación integrada y debugging:** incorpora una interfaz ST-LINK/V2-1 que permite la programación y depuración del microcontrolador sin necesidad de hardware adicional [16]. Esto simplifica el desarrollo y facilita la identificación y corrección de errores durante las pruebas del sistema.
- **Alimentación:** puede ser alimentada tanto por USB como por una fuente externa de 5 V o 3,3 V, lo que la hace adaptable a diferentes entornos operativos.

Una de las grandes ventajas de la NUCLEO F429ZI es la posibilidad de desarrollar firmware utilizando herramientas muy potentes como STM32CubeIDE [17], que proporciona un entorno

de desarrollo completo y gratuito. Mediante una interfaz gráfica, se puede configurar la función de cada uno de los pines, nombrarlos y asegurarse de que no haya superposiciones de definiciones o protocolos de comunicación definidos sin los pines necesarios para que funcione. También facilita la configuración de un sistema operativo en tiempo real, permitiendo la declaración de las distintas tareas y mecanismos de sincronización. Además, el uso de bibliotecas como STM32CubeMX [18] facilita la configuración de los periféricos del microcontrolador y la generación de código, reduciendo el tiempo de desarrollo y minimizando la posibilidad de errores.

2.1.2 ESP32 Devkit-C

El ESP32 Devkit-C es una placa de desarrollo compacta y versátil basada en el microcontrolador ESP32 de Espressif Systems, una solución popular en el ámbito de los sistemas embebidos y aplicaciones IoT [19]. Este módulo está diseñado para ofrecer una plataforma accesible para el desarrollo de aplicaciones que requieren conectividad Wi-Fi [20] y Bluetooth [21], así como capacidades de procesamiento avanzadas. En la figura 2.3 se puede visualizar el módulo ESP32 Devkit-C.

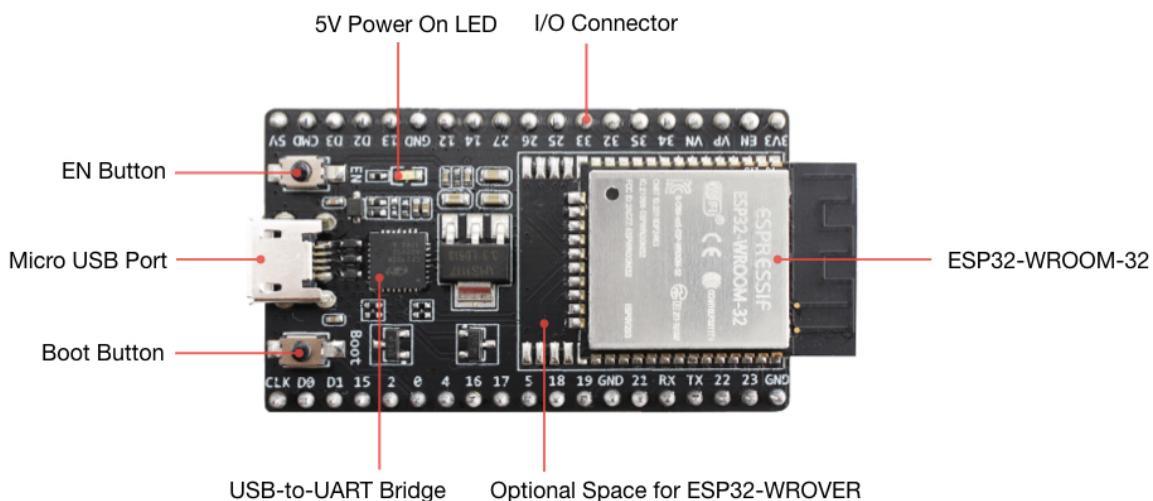


Figura 2.3: Placa de desarrollo ESP32 Devkit-C

Algunas de las características principales del ESP32 son las siguientes:

- **Microcontrolador dual-core:** cuenta con un procesador dual-core Xtensa LX6 que opera a una velocidad de hasta 240 MHz. Esta arquitectura multiprocesador permite ejecutar tareas paralelas, lo que resulta ventajoso en aplicaciones donde es necesario manejar múltiples procesos simultáneamente, como la gestión de comunicaciones y control de dispositivos.
- **Conectividad inalámbrica:** uno de los aspectos más destacados del ESP32 es su capacidad de conectividad inalámbrica:
 - **Wi-Fi 802.11 b/g/n:** incluye un módulo Wi-FI integrado que permite conectarse a redes inalámbricas, ya sea como cliente o como punto de acceso.
 - **Bluetooth 4.2 (BLE y clásico):** soporta Bluetooth de baja energía (BLE) y Bluetooth clásico, lo que facilita la comunicación con una amplia variedad de dispositivos, desde teléfonos móviles hasta sensores y actuadores.

- **Amplia gama de interfaces de comunicación:** ofrece varias interfaces periféricas, lo que lo hace adecuado para interactuar con otros dispositivos y sensores (UART, SPI, I2C, CAN, PWM, ADC y DAC).
- **Memoria:** incluye 520 KB de SRAM y hasta 16 MB de flash externa, lo que le permite ejecutar aplicaciones complejas y almacenar datos de manera eficiente. Esta combinación de memoria es adecuada para gestionar tanto tareas de conectividad como de procesamiento en tiempo real.
- **Bajo consumo:** gracias a su arquitectura optimizada para el ahorro de energía, el ESP32 ofrece múltiples modos de bajo consumo, como el modo de suspensión profunda. Esto es especialmente útil en aplicaciones IoT donde la eficiencia energética es crucial y los dispositivos deben funcionar durante largos períodos con baterías limitadas.
- **Capacidades de seguridad:** está equipado con una serie de características de seguridad integradas, como encriptación de hardware (AES, RSA), arranque seguro y protección de memoria, lo que lo hace adecuado para aplicaciones donde la integridad y confidencialidad de los datos son esenciales.

El ESP32 es compatible con una amplia variedad de entornos de desarrollo, entre ellos Arduino IDE [22], Espressif IoT Development Framework (ESP-IDF) [23] o PlatformIO [24]. Estas herramientas permiten a los desarrolladores adaptar el entorno de trabajo a sus necesidades específicas. El uso de estos entornos no solo mejora la productividad, sino que también ofrece ventajas clave como facilidad para el desarrollo, una amplia comunidad de soporte, y la capacidad de escalar los proyectos de prototipos a soluciones más avanzadas y robustas.

2.1.3 Módulo GPS GY-NEO6MV2

El módulo GPS GY-NEO6MV2 es un receptor GPS de alta precisión que permite obtener datos de ubicación geográfica a través del sistema de posicionamiento global [25]. Es ampliamente utilizado en aplicaciones de navegación y rastreo en sistemas embebidos gracias a su bajo consumo de energía y facilidad de integración.

Este módulo está basado en el chip NEO-6M de U-blox [26], que proporciona una alta sensibilidad de recepción, incluso en entornos con señal débil. Tiene un conector para antena UFL y es capaz de obtener información como coordenadas geográficas (latitud, longitud), altitud, velocidad y tiempo de posicionamiento, lo que lo hace adecuado para una amplia gama de aplicaciones. En la figura 2.4 se visualiza el módulo descripto.

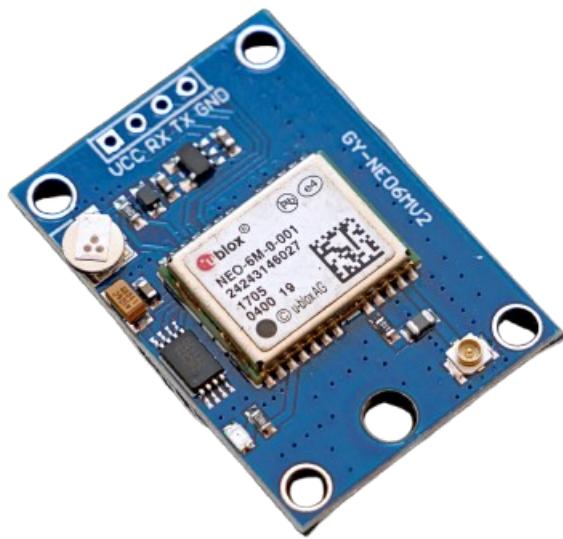


Figura 2.4: Módulo GPS GY-NEO6MV2

Algunas de sus características principales son:

- Conexión UART para la comunicación con otros dispositivos, facilitando su integración con microcontroladores.
- Alta sensibilidad y rápido tiempo de adquisición, incluso en condiciones de baja señal.
- Almacenamiento de configuración mediante una memoria EEPROM interna, lo que permite guardar parámetros de configuración para su uso posterior sin necesidad de reconfigurar el módulo al encenderlo.

Gracias a su tamaño compacto y bajo consumo, el GY-NEO6MV2 es ideal para sistemas que requieren monitoreo de posición en tiempo real con un costo y consumo energético reducidos.

2.1.4 Memoria SD

Las memorias SD (*Secure Digital*) son dispositivos de almacenamiento de datos que han ganado popularidad debido a su tamaño compacto, capacidad de almacenamiento y facilidad de uso. Son ampliamente utilizadas en diversas aplicaciones, desde cámaras digitales hasta sistemas embebidos, gracias a su versatilidad y eficiencia. En la figura 2.5 se visualiza una memoria SD.

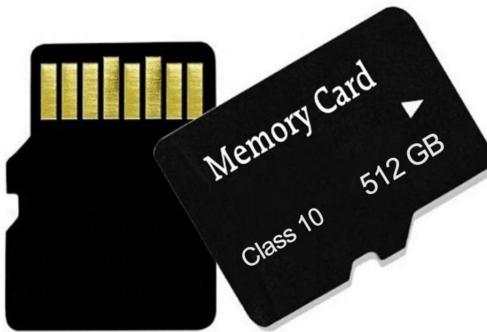


Figura 2.5: Memoria SD

Las memorias SD cuentan con las principales características:

- **Capacidad de almacenamiento:** las memorias SD están disponibles en varias capacidades, que varían desde unos pocos megabytes hasta varios terabytes. Esto permite a los usuarios elegir la memoria adecuada según sus necesidades específicas de almacenamiento.
- **Velocidades de transferencia:** existen diferentes clases de memorias SD, como Clase 2, 4, 6, 10, y UHS-I y UHS-II, que determinan la velocidad mínima de lectura y escritura. Las memorias de clase más alta son ideales para aplicaciones que requieren una rápida transferencia de datos, como la grabación de video en alta definición.
- **Interfaz de comunicación:** Las memorias SD utilizan una interfaz estándar que permite la conexión con dispositivos a través de protocolos como SPI (*Serial Peripheral Interface*) o SDIO (*Secure Digital Input Output*). Esto facilita su integración en una variedad de sistemas.
- **Durabilidad y fiabilidad:** muchas memorias SD están diseñadas para resistir condiciones adversas, como temperaturas extremas, golpes y vibraciones. Esto las hace adecuadas para su uso en entornos donde la robustez es fundamental.
- **Facilidad de uso:** la implementación de memorias SD es sencilla, ya que muchos dispositivos y plataformas de desarrollo ofrecen bibliotecas y herramientas que facilitan la lectura y escritura de datos en estas memorias.

En los sistemas embebidos las memorias SD se colocan sobre un zócalo que permite removérlas para luego insertar la memoria SD en una PC y descargar los datos, formatearla o repararla. También existen módulos para Arduino que permiten integrar una memoria SD al proyecto en etapas tempranas antes de fabricar un PCB propio y soldar un zócalo. Esto facilita mucho el mantenimiento y el desarrollo de sistemas que contienen almacenamiento de datos en memorias SD.

Existen tres formatos principales de memorias SD: SD estándar (32 x 24 mm), miniSD (21.5 x 20 mm) y microSD (15 x 11 mm). La microSD es la más compacta y comúnmente utilizada en aplicaciones embebidas por su tamaño reducido y gran capacidad de almacenamiento. Por estos motivos, se utiliza el formato de memoria microSD para este proyecto.

Las memorias SD se han convertido en una solución de almacenamiento clave en muchos dispositivos electrónicos, ofreciendo una combinación de flexibilidad, capacidad y rendimiento que se adapta a una amplia gama de aplicaciones.

2.2 Protocolos de comunicación

2.2.1 MQQT

MQTT (*Message Queuing Telemetry Transport*) es un protocolo de comunicación ligero diseñado específicamente para entornos con limitaciones de ancho de banda, latencia o recursos, lo que lo hace ideal para aplicaciones de IoT. Desarrollado por IBM en 1999 [27], MQTT sigue el modelo *publish/subscribe*, lo que permite la transmisión eficiente de mensajes entre dispositivos conectados sin necesidad de una conexión constante y directa entre ellos.

El protocolo MQTT utiliza una arquitectura basada en los siguientes componentes clave que se pueden visualizar su interacción en la figura 2.6:

- **Broker:** es el servidor central que gestiona la recepción y distribución de los mensajes. El *broker* es responsable de recibir los mensajes de los *publishers* y reenviarlos a los *subscribers* que se hayan suscrito al tema correspondiente.
- **Publisher:** es el dispositivo o cliente que envía un mensaje a un tema específico, sin necesidad de conocer a los destinatarios. Solo necesita enviar el mensaje al *broker*.
- **Subscriber:** es el dispositivo o cliente que recibe los mensajes de un tema al cual se ha suscrito previamente. El suscriptor se comunica únicamente con el *broker* y no directamente con los publicadores.
- **Topics:** los *topics* o temas son etiquetas jerárquicas utilizadas para categorizar los mensajes en MQTT. Cuando un publicador envía un mensaje, lo asocia con un tema específico. Los suscriptores que se suscriben a ese tema recibirán los mensajes. La flexibilidad de los *topics* permite organizar la información de manera eficiente y estructurada.

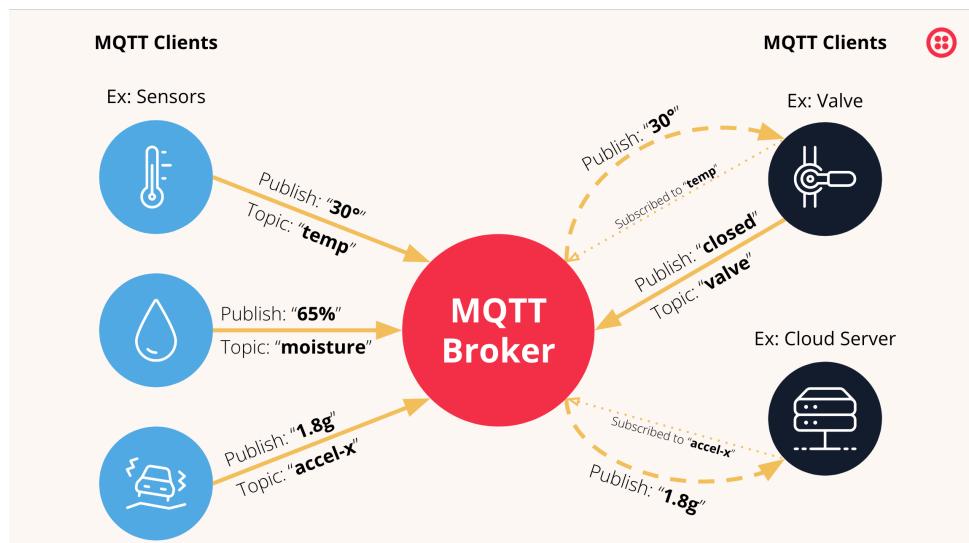


Figura 2.6: Diagrama de conexión MQTT de ejemplo [28]

Las principales ventajas del protocolo MQTT se pueden resumir en:

- **Ligereza:** MQTT está diseñado para ser extremadamente eficiente en cuanto a recursos, lo que lo hace ideal para dispositivos con capacidades limitadas, como sensores o microcontroladores.
- **Minimización del consumo de energía:** al ser un protocolo que no requiere mantener conexiones activas todo el tiempo, los dispositivos pueden ahorrar energía, lo que es fundamental en aplicaciones IoT.

- **Comunicación asíncrona:** gracias a su modelo de publicación y suscripción, los dispositivos no necesitan estar conectados constantemente. El *broker* gestiona las comunicaciones y distribuye los mensajes a los suscriptores según sea necesario.

MQTT es ampliamente utilizado en aplicaciones IoT, como monitoreo remoto, automatización industrial, gestión de energía y sistemas embebidos. También se ha implementado en sistemas de emergencia, control de dispositivos y telemetría, donde la eficiencia y la fiabilidad son esenciales.

2.2.2 Wi-Fi

Wi-Fi es un conjunto de tecnologías que permiten la comunicación inalámbrica entre dispositivos electrónicos utilizando ondas electromagnéticas. Es uno de los protocolos de comunicación más comunes en redes locales (WLAN) y se basa en la familia de estándares IEEE 802.11 [20], desarrollados y mantenidos por el Instituto de Ingenieros Eléctricos y Electrónicos (IEEE). Wi-Fi es ampliamente utilizado en aplicaciones que requieren conectividad a Internet o comunicación local sin cables, lo que lo convierte en una opción clave en proyectos de IoT y sistemas embebidos.

Wi-Fi opera en bandas de frecuencia de 2.4 GHz y 5 GHz, y permite a los dispositivos conectarse a una red a través de un punto de acceso (*Access Point*) o funcionar en un modo de conexión directa *peer-to-peer* (*ad-hoc*). En la figura 2.7 se visualizan los dos tipos de conexiones mencionadas. En un entorno típico, un *router* actúa como el punto de acceso central, al cual se conectan varios dispositivos para intercambiar datos entre sí y acceder a Internet.

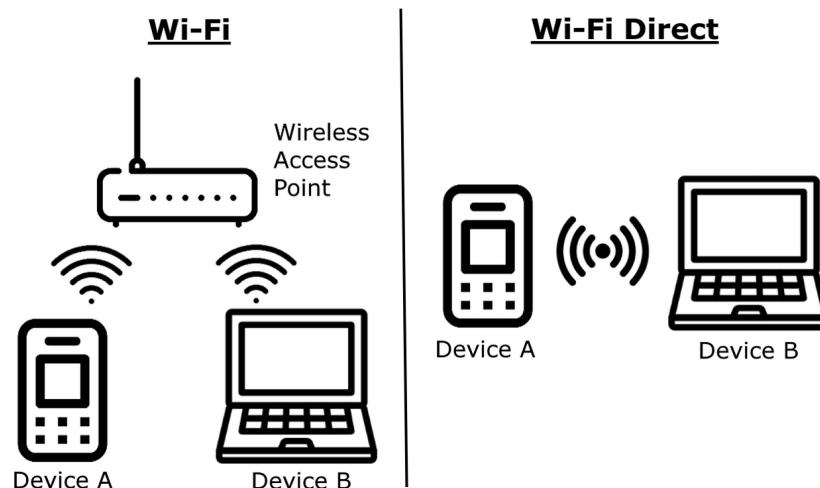


Figura 2.7: Modos de conexión Wi-Fi [30]

Las principales características de Wi-Fi son las siguientes:

- **Alcance:** ofrece un rango de cobertura considerable, que generalmente varía entre 20 a 50 metros en interiores y hasta 100 metros en exteriores, dependiendo de las condiciones de señal y el entorno.
- **Velocidad:** permite transmitir datos a distintas velocidades según la señal y las condiciones de la red, ofreciendo un rendimiento adecuado para la mayoría de las aplicaciones cotidianas.
- **Seguridad:** incorpora varios protocolos de seguridad, como WEP, WPA y WPA2 [29], que aseguran la comunicación cifrada entre dispositivos y protegen las redes inalámbricas de accesos no autorizados.

- **Compatibilidad:** es retrocompatible, lo que significa que los dispositivos más antiguos que soportan versiones previas del estándar, como 802.11a, b, g o n, pueden seguir conectándose a redes más modernas, aunque con limitaciones en velocidad y funcionalidad.

El protocolo Wi-Fi cuenta con algunas limitaciones como su consumo de energía; que suele ser mayor comparado con otros protocolos con el Bluetooth, o la congestión y colisiones que pueden ocurrir en entornos densos con muchos dispositivos conectados; esto puede generar interferencia y degradación en el rendimiento de las conexiones.

Sin embargo, el Wi-Fi se popularizó cada vez más debido las siguientes ventajas que genera:

- **Flexibilidad:** elimina la necesidad de cables, lo que facilita la instalación y operación en entornos móviles o de difícil acceso. Esto es especialmente útil en redes IoT donde se requiere la conectividad de múltiples dispositivos dispersos.
- **Conectividad global:** la mayoría de los dispositivos modernos, desde teléfonos móviles hasta microcontroladores y computadoras, tienen Wi-Fi integrado, lo que permite la interoperabilidad entre diferentes plataformas y tecnologías.
- **Alta velocidad de transmisión:** es adecuado tanto para aplicaciones que requieren grandes volúmenes de datos, como la transmisión de video en tiempo real, como para aquellas que necesitan una transferencia rápida, como la telemetría o la comunicación de sensores.

Wi-Fi es el protocolo preferido para aplicaciones que requieren transferencias rápidas y confiables de datos, como transmisión de video, comunicación entre dispositivos móviles, automatización del hogar y acceso a Internet en redes locales. También es fundamental en muchas aplicaciones de IoT que requieren conectividad en tiempo real y de alto ancho de banda.

2.2.3 GPS

El GPS (*Global Positioning System*) es una tecnología de navegación por satélite que permite determinar la ubicación geográfica exacta de un objeto o dispositivo en cualquier parte del mundo. El sistema fue desarrollado por el Departamento de Defensa de los Estados Unidos [31] y está compuesto por una red de satélites que transmiten señales de tiempo y posición generalmente en la banda de frecuencia L1 1.57542 GHz para usos civiles. Los dispositivos con receptores GPS pueden calcular su posición en la Tierra midiendo el tiempo que tarda la señal en viajar desde los satélites.

El GPS utiliza una técnica llamada trilateración para determinar la posición de un dispositivo. Para calcular su ubicación, el receptor GPS necesita recibir señales de al menos cuatro satélites. A partir de estas señales, el receptor puede calcular la distancia a cada satélite y determinar con precisión su posición en coordenadas geográficas (latitud, longitud y altitud). El GPS utiliza cuatro satélites en lugar de tres para calcular no solo la ubicación en tres dimensiones, sino también para corregir errores en el reloj del receptor y obtener mayor precisión. En la figura 2.8 se visualiza cómo un receptor recibe señales de 4 satélites lo que le permite determinar su ubicación.

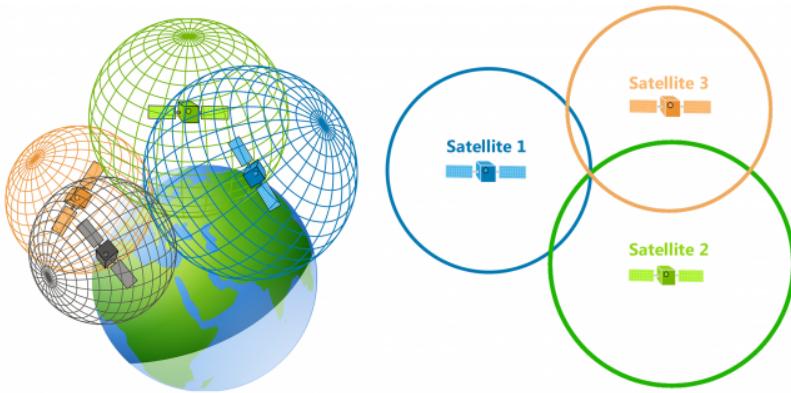


Figura 2.8: Trilateración utilizada en GPS [32]

En los sistemas embebidos, los módulos GPS se utilizan ampliamente para aplicaciones que requieren posicionamiento y geolocalización. Estos módulos son pequeños, de bajo consumo de energía y fáciles de integrar en una amplia variedad de dispositivos y proyectos. Algunas aplicaciones comunes en sistemas embebidos incluyen:

- **Monitoreo de vehículos:** el GPS es utilizado para rastrear vehículos en tiempo real, facilitando sistemas de logística, transporte público y servicios de emergencia.
- **Agricultura de precisión:** equipos agrícolas con GPS pueden optimizar la siembra, el riego y la cosecha mediante la geolocalización precisa.
- **Dispositivos IoT:** muchos dispositivos conectados a Internet, como drones, sensores o sistemas de automatización, utilizan GPS para la recolección de datos de ubicación.
- **Navegación y mapeo:** desde sistemas de navegación portátiles hasta drones y robots autónomos, el GPS permite trazar rutas y moverse eficientemente por entornos dinámicos.

La utilización de GPS provee una alta precisión que varía desde los pocos metros hasta los centímetros (dependiendo de las condiciones y el equipo). También tiene la ventaja de contar con una cobertura global lo que permite una solución de geolocalización sin necesidad de infraestructura adicional. Además, los módulos GPS para sistemas embebidos están diseñados para ser eficientes energéticamente, lo que los hace ideales para dispositivos portátiles o alimentados por baterías.

En resumen, la tecnología GPS ha revolucionado la capacidad de los sistemas embebidos para obtener y procesar datos de ubicación de manera eficiente y precisa. Su integración en dispositivos y aplicaciones es clave para una variedad de sectores, como la logística, la agricultura y el IoT, proporcionando soluciones versátiles y de bajo consumo para necesidades de geolocalización. La combinación de su cobertura global y su capacidad de adaptación lo convierte en una herramienta fundamental en el diseño de sistemas embebidos modernos.

2.2.4 RS-485

El RS-485 es un estándar de comunicación serial que se utiliza ampliamente en aplicaciones industriales y de automatización debido a su robustez y fiabilidad [33]. A diferencia de protocolos de comunicación como SPI, el RS-485 permite la comunicación de múltiples dispositivos en un mismo bus de datos sin necesidad de agregar una línea de selección por cada dispositivo, lo que lo hace ideal para redes en las que se requiere que varios dispositivos comparten una única línea de comunicación. En la figura 2.9 se visualiza un bus de RS-485 con varios dispositivos conectados en modo *half-duplex* (los dispositivos pueden enviar o recibir, pero no ambas cosas

al mismo tiempo).

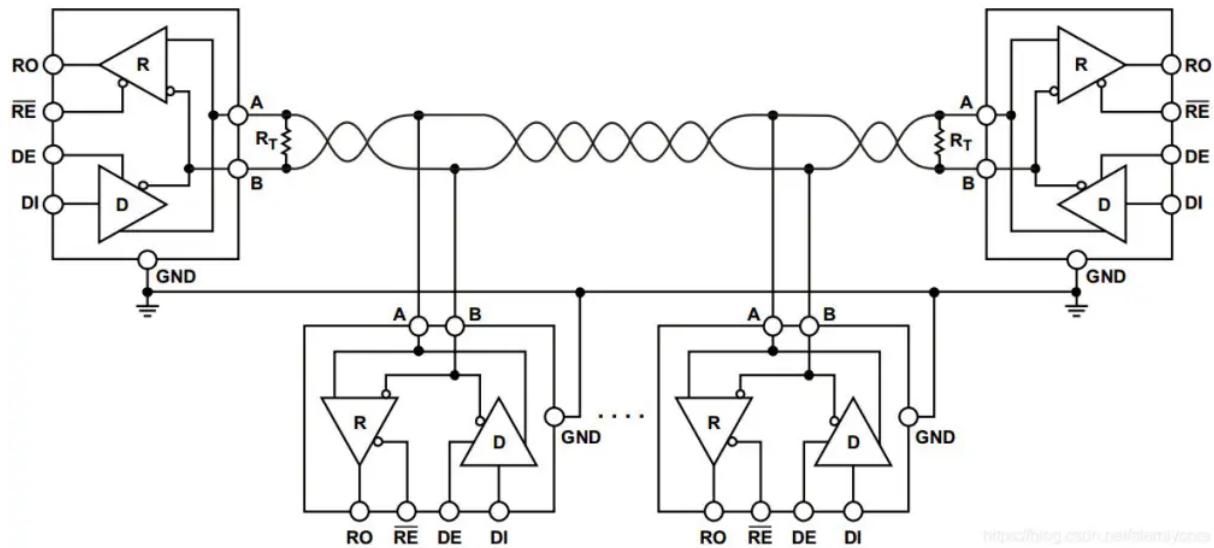


Figura 2.9: Bus RS-485 en modo *half-duplex*

Algunas de las características principales del protocolo RS-485 son las siguientes:

- **Comunicación diferencial:** RS-485 utiliza dos líneas para la transmisión de datos (A y B), lo que le permite operar bajo el principio de comunicación diferencial. Esto significa que el valor lógico de los datos se determina por la diferencia de tensión entre las dos líneas, lo que mejora la inmunidad al ruido y permite transmitir datos a largas distancias sin degradación de la señal.
- **Distancia y velocidad:** puede transmitir datos a distancias de hasta 1200 metros, dependiendo de la velocidad de comunicación. A velocidades bajas (por ejemplo, 100 kbps), puede cubrir distancias largas, mientras que a velocidades más altas (por ejemplo, 10 Mbps), la distancia máxima disminuye.
- **Topología multipunto:** una de las ventajas más notables de RS-485 es su capacidad para soportar comunicación multipunto, permitiendo que hasta 32 dispositivos se conecten en el mismo bus sin necesidad de repetidores. Esto es muy útil en entornos donde varios dispositivos necesitan intercambiar información, como en redes de sensores o controladores.
- **Modo half-duplex y full-duplex:** RS-485 se puede configurar tanto en modo half-duplex como en modo full-duplex (utilizando dos pares de cables para permitir la transmisión y recepción simultánea).
- **Robustez:** debido a su capacidad de resistir interferencias electromagnéticas y operar en entornos industriales hostiles, RS-485 es ampliamente utilizado en aplicaciones donde la fiabilidad y la estabilidad de la comunicación son cruciales.

La flexibilidad en la topología de red es una de las ventajas significativas del protocolo RS-485. A diferencia de otros estándares seriales como RS-232, que se limitan a la comunicación punto a punto, el RS-485 permite configuraciones de red en bus o *multidrop*. Esto lo convierte en una opción ideal para sistemas distribuidos donde varios nodos necesitan intercambiar datos de manera eficiente. Además, en aplicaciones que requieren una mayor distancia o más dispositivos en el bus, se pueden utilizar repetidores o extensores para ampliar el alcance y la capacidad de la red sin comprometer la calidad de la señal. Esta versatilidad, junto con su capacidad para operar en entornos hostiles, ha hecho que RS-485 sea una solución de comunicación robusta y ampliamente adoptada en el ámbito industrial y de control.

2.3 Sistema operativo de tiempo real

Un sistema operativo en tiempo real (RTOS por sus siglas en inglés) es un tipo de sistema operativo diseñado para gestionar tareas de manera determinista, garantizando que ciertas operaciones se completen dentro de plazos definidos [34]. A diferencia de los sistemas operativos convencionales, donde la prioridad de las tareas puede variar según la carga del sistema, un RTOS garantiza que las tareas críticas se ejecuten en tiempo y forma según su prioridad.

Los RTOS son fundamentales en aplicaciones embebidas donde es crucial que las respuestas a eventos externos, como la activación de un sensor o una interrupción, se produzcan dentro de tiempos predeterminados. Estos sistemas son ampliamente utilizados en sectores como el automotriz, la aviación, las telecomunicaciones, y los sistemas médicos.

Algunas características clave de los RTOS incluyen:

- **Determinismo:** el comportamiento predecible en cuanto a la ejecución de tareas es esencial. Un RTOS asegura que las tareas se ejecuten de acuerdo con una planificación estricta, cumpliendo con los plazos impuestos por el sistema.
- **Multitarea:** un RTOS es capaz de manejar múltiples tareas concurrentemente, priorizando aquellas que son más críticas para el sistema.
- **Gestión de interrupciones:** la rápida respuesta a interrupciones de hardware es vital en un RTOS. Estos sistemas permiten que las tareas de mayor prioridad respondan rápidamente a eventos externos, sin demoras significativas.
- **Uso eficiente de recursos:** dado que los RTOS suelen implementarse en sistemas embebidos con recursos limitados, están optimizados para ser ligeros y eficientes en términos de memoria y procesamiento.

En la figura 2.10 se muestra un ejemplo de un RTOS realizando asignación de tareas con distintas prioridades e interactuando con una interrupción por hardware.

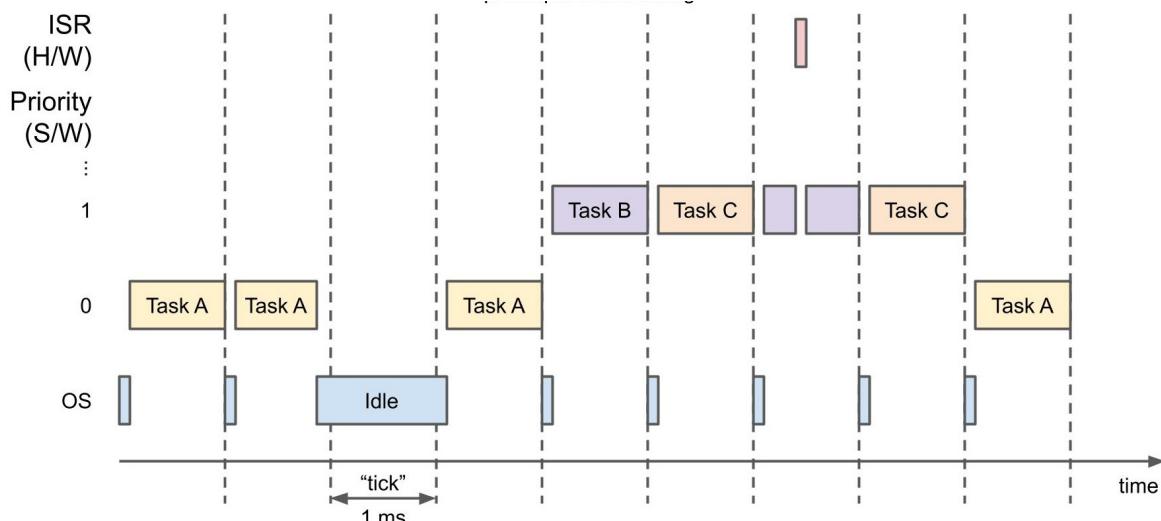


Figura 2.10: Asignación de tareas por prioridades en un RTOS

3 Diseño e implementación

En este capítulo se realiza una descripción general de la solución y se detalla el proceso de diseño e implementación del sistema a nivel hardware y firmware.

3.1 Descripción general de la solución

Una vez relevados todos los requerimientos asociados al proyecto y revisados los diseños del prototipo anterior del SAL/T, se generó un diagrama en bloques de la solución donde se contemplan todos los módulos involucrados, que se puede visualizar en la figura 2.1. Se observa que el sistema interactúa con distintos elementos externos pertenecientes a la formación ferroviaria, como los sistemas instrumentados de seguridad o el control central de la formación. Se considera también el uso de una fuente externa que permite alimentar al SAL/T con 5 V de tensión continua generados a partir de las baterías de alimentación de los trenes que están en un rango de entre 72-110 VDC dependiendo del modelo de la formación. También existe una interacción con el bus de datos que conecta el registrador de eventos Hasler Teloc 1500 [35] con el indicador de velocidad Hasler y otra interacción con un subsistema externo que realiza la interpretación de la velocidad a partir de la señal generada por un generador de impulsos; ambas interactúan con las interfaces RS-485 del sistema.

Con el diagrama en bloques del sistema, se pueden considerar las distintas interacciones que va a tener el controlador principal y elegir algún microcontrolador (MCU) que tenga las interfaces necesarias para conectarse con todos los módulos, y la capacidad de procesamiento adecuada para este tipo de aplicaciones. Después de analizar las distintas alternativas disponibles en el mercado, se eligió el microcontrolador STM32F429ZI ya que tiene un buen rendimiento por su core ARM Cortex-M4 que corre en una frecuencia de hasta 180 MHz con una unidad de punto flotante, soporta múltiples interfaces de comunicación (I2C, SPI, UART, GPIO, ADC, etc.) y cuenta con memoria flash y RAM suficientes para un firmware complejo. Otra ventaja de usar los microcontroladores de ST es que cuentan con el entorno de desarrollo STM32CubeIDE que es muy amigable para la programación del MCU y la configuración de todos los periféricos.

Una vez seleccionado el microcontrolador principal, se decidió armar el prototipo del SAL/T a partir de una placa de desarrollo que contiene al STM32F429ZI para mantener el alcance del proyecto dentro de lo esperado para este trabajo. Por lo tanto, se seleccionó la placa de desarrollo NUCLEO F429ZI que contiene el MCU mencionado. Esta placa incluye una interfaz ST-LINK/V2-1 que permite la programación y *debugging* del microcontrolador sin necesidad de hardware adicional. También trae circuitos adicionales como el oscilador de cristal, circuito de reset, LEDs, botones de prueba y reguladores de tensión que permiten alimentar la placa con distintos niveles de tensión. Todas estas características facilitan el desarrollo del proyecto porque permiten probar algunos circuitos, partes del firmware y la comunicación con otros módulos, sin tener todavía fabricado un PCB diseñado ad hoc para el proyecto.

Con la placa de desarrollo definida como base del sistema, se realizó una búsqueda de los posibles circuitos o módulos para implementar los distintos bloques que cumplan con los requerimientos acordados y que permitan hacer un uso distribuido de las interfaces que ofrece el MCU. Para la selección de componentes y módulos, se consideraron algunos criterios que facilitan la selección. Por ejemplo, se intentó reutilizar módulos o componentes ya utilizados en otros proyectos del grupo GICSAFe para poder ayudarse de diseños y experiencias de uso previas; esto reduce la posibilidad de error o mal funcionamiento de los circuitos y acelera tiempos de descubrimiento en fallas ocultas que pueden traer asociadas. Otro criterio importante fue utilizar dos distribuidores de componentes electrónicos grandes (Mouser [36] y Digikey [37]) para verificar que estas partes estén disponibles y en stock de manera general y que sean recomendados para

nuevos diseños y no en un estado cercano al final de su ciclo de vida (EOL, *End-of-Life*); de esta manera se asegura que el prototipo y su implementación sean reproducibles y sus partes fáciles de conseguir en el mediano plazo.

Con una idea más clara de los circuitos que se podían usar para implementar los distintos bloques y sus variantes de comunicación con el MCU, se realizó el diagrama de la figura 3.1 que muestra los protocolos de comunicación entre el MCU y los distintos circuitos del sistema. Esto permitió hacer una asignación de las interfaces disponibles para cada bloque.

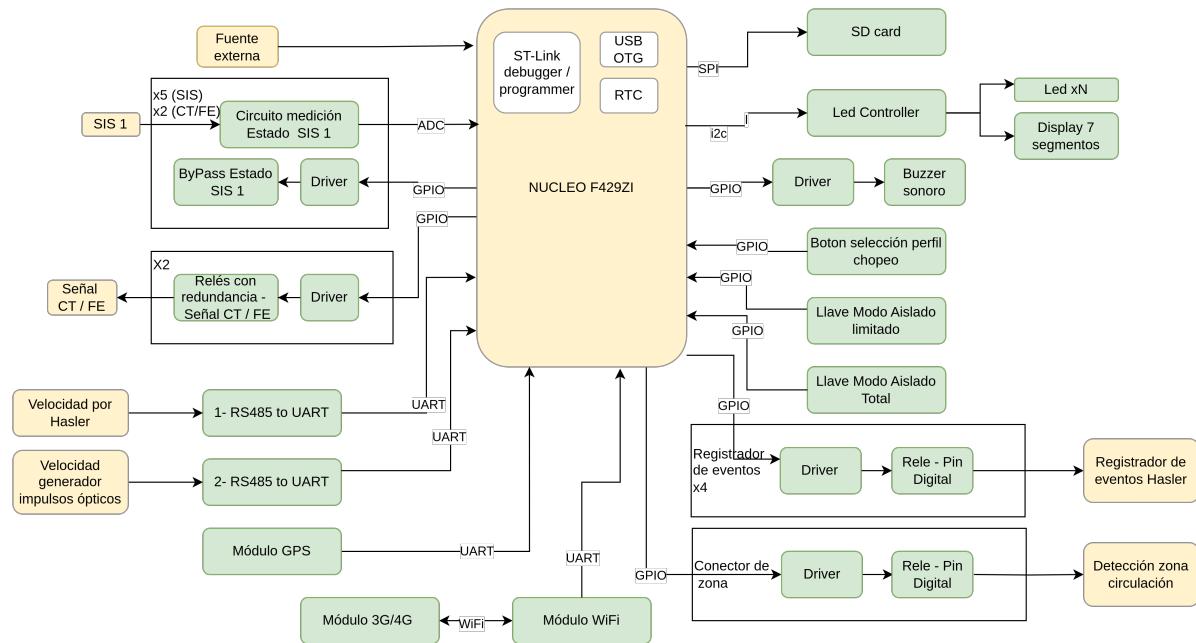


Figura 3.1: Diagrama con interfaces de comunicación del SAL/T

Además, considerando que según los requisitos el sistema cuenta con un panel frontal para interactuar con el maquinista de la formación, se realizó el diseño de este panel para validar que tuviera todas las indicaciones e interacciones acordadas en los requerimientos. En la figura 3.2 se puede visualizar la interfaz humano-máquina (IHM) que va a mostrar el estado del SAL/T en todo momento y permitir algunas interacciones mínimas al maquinista como la habilitación del modo aislado limitado o el cambio del perfil intermitente en caso de no contar con mediciones de velocidad activas.

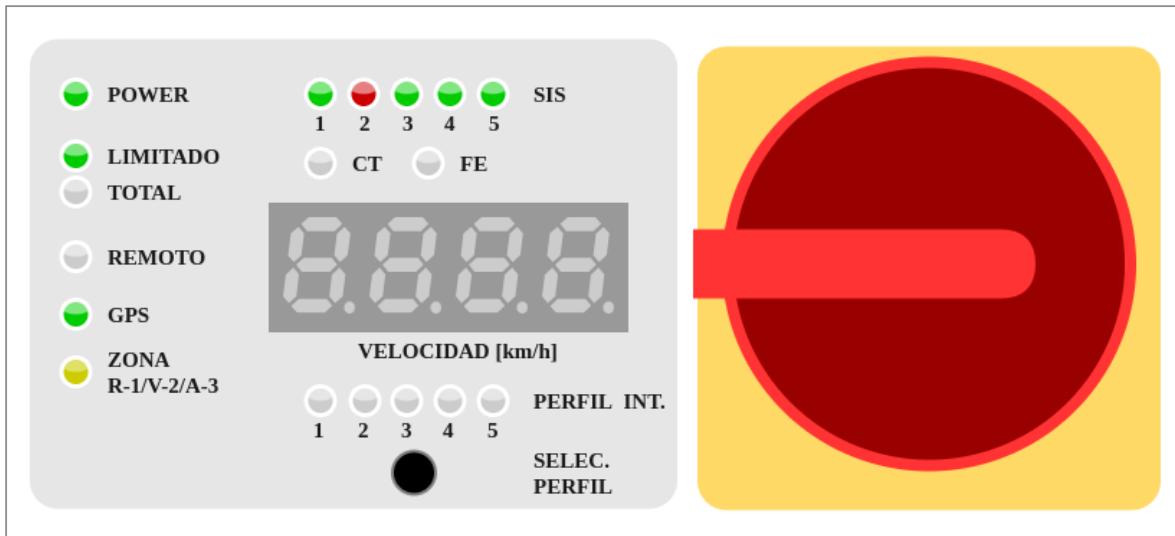


Figura 3.2: Diseño del panel frontal del SAL/T

Para comprender el diseño de los circuitos de medición y aislamiento de los SIS, así como la activación de las señales críticas de corte de tracción (CT) y freno de emergencia (FE), es importante entender cómo es la conexión en una formación ferroviaria entre los SIS y las señales críticas. Lo primero a considerar es que las líneas de activación de la señal CT y la de la señal FE son independientes; cada una tiene una electroválvula separada e interactúa con los SIS en una línea propia. Luego, para cada una de estas señales críticas, existe una conexión en serie entre la alimentación de la electroválvula, una llave cerrada que se abre en caso de falla del SIS para cada uno de los SIS, la electroválvula que habilita y deshabilita la señal crítica y tierra. Esta electroválvula, al seguir el concepto de *fail-safe*, debe estar alimentada para poder liberar alguna de las señales críticas, y en caso de cortarse la alimentación, ya sea por una falla en la alimentación o porque uno de los SIS está fallando y reporta un estado inseguro abriendo la llave, se activa la señal de CT o de FE para frenar la formación. Este esquema se puede visualizar en la figura 3.3.

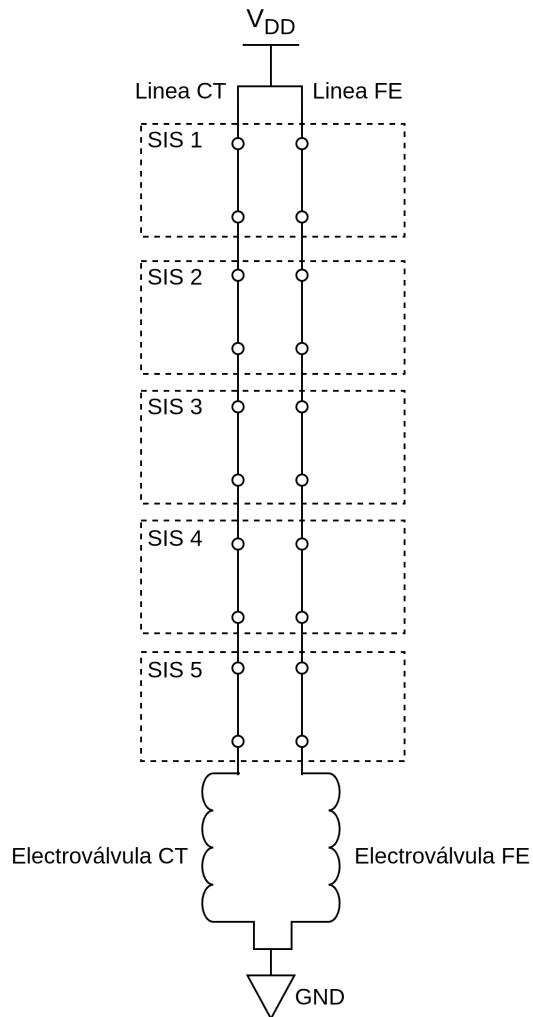


Figura 3.3: Lineas de señales de CT y FE

La idea del proyecto SAL/T es no modificar la arquitectura actual de los sistemas dentro de la formación, sino agregar algunos circuitos más para poder incrementar su seguridad sin necesidad de cambios en lo preexistente. Por esto, se propuso realizar las conexiones de la figura 3.4 para poder realizar mediciones del estado de cada SIS independientemente, poder aislar cualquiera de ellos en caso de entrar en un modo aislado limitado, y también tener un último control para la activación de las señales críticas imponiendo una alimentación por fuera de la línea de los SIS. En la figura se muestran 2 SIS a modo de ejemplo y la línea de la señal de FE, pero el proyecto contempla 5 SIS y este mismo esquema para la señal de FE y CT.

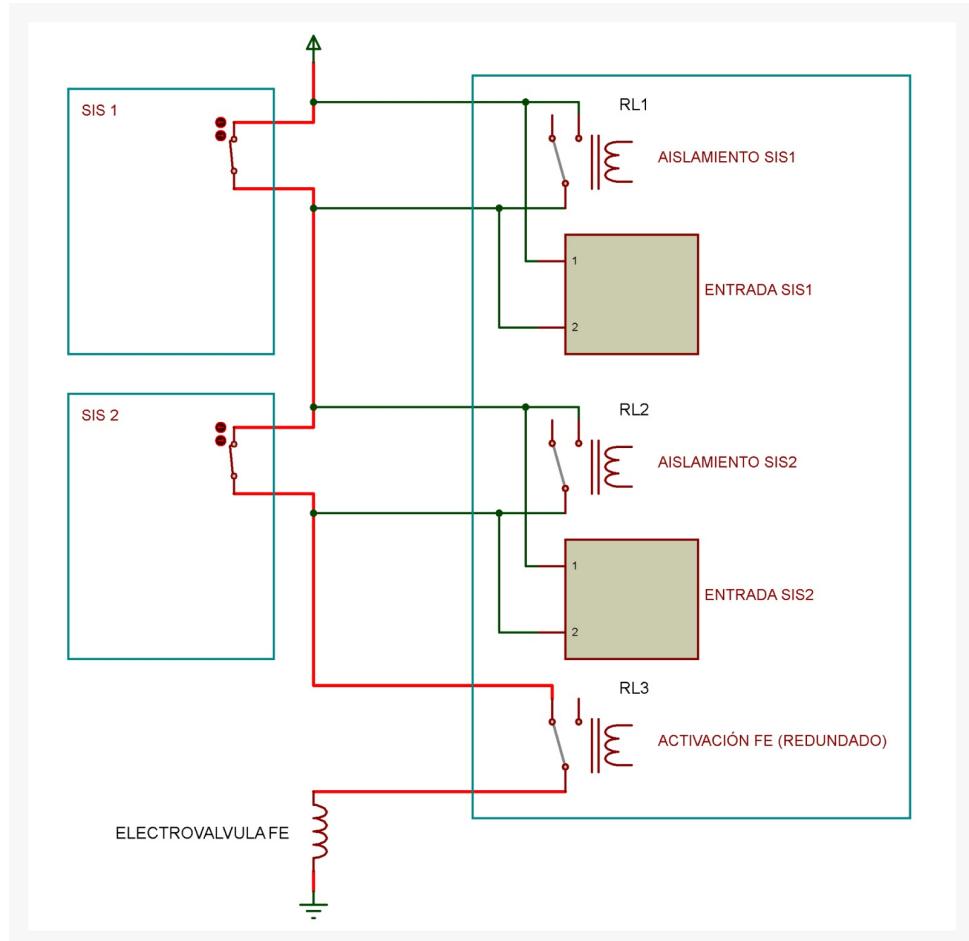


Figura 3.4: Diseño de conexión con SIS y señales críticas en el SAL/T

Con la arquitectura general de la solución ya definida, se decidió modularizar el sistema en dos placas separadas. Una placa principal donde se va a realizar todo el procesamiento del sistema, se van a recibir las mediciones de velocidad, del estado de los SIS, de la posición, los comandos remotos y donde se va a interactuar con el aislamiento de los SIS, la activación de las señales críticas, el registro de eventos internos y externos de manera local y remoto y la descarga de los eventos guardados. La segunda placa va a estar dedicada a la interacción con el maquinista y va a contar con el panel frontal, los LEDs indicativos, el display para mostrar la velocidad de la formación y un botón para seleccionar el perfil intermitente a ejecutar en el modo aislado limitado cuando no se cuenta con mediciones de velocidad válidas. De esta manera, se permite mayor flexibilidad en el diseño de las placas y una mejor modularización del sistema.

Para realizar el diseño del hardware se utilizó KiCad [38] versión 6; una herramienta de software libre utilizada para el diseño de esquemáticos y PCB. También se utilizó LTspice [39]; un simulador de circuitos de código abierto desarrollado por Analog Devices, utilizado para simular el comportamiento de los distintos circuitos en la etapa de diseño. Para llevar un control de versiones, tanto de los archivos de diseño de hardware como de firmware, se utilizó GitHub [40]; una plataforma de desarrollo colaborativo que utiliza Git [41] como sistema de control de versiones que permite almacenar, gestionar y rastrear cambios en el código fuente de sus proyectos. Se utilizó un repositorio donde se fueron avanzando los distintos módulos (tanto de hardware como de firmware) del proyecto en distintas *branches* para finalmente llegar a una versión unificada en la *branch* principal. El repositorio es público y accesible en github.com/martinanrus/salt [42].

3.2 Implementación de hardware

3.2.1 Medición de estado de los SIS

El SAL/T debe interactuar con cinco sistemas instrumentados de seguridad (SIS) que cada uno cuenta con dos llaves de estado; una para la línea de la señal crítica de corte de tracción (CT) y la otra para la señal crítica de freno de emergencia (FE). Estas llaves van a estar en un estado cerrado cuando el SIS esté funcionando correctamente y no detecte ninguna situación peligrosa en la formación, y se van a abrir para interrumpir la alimentación de las señales críticas ante una falla en el SIS o la detección de alguna condición peligrosa.

Considerando el diagrama de conexión ya presentado en la figura 3.4, se puede interpretar que cuando la llave está cerrada, la diferencia de tensión entre sus terminales va a ser nula. En cambio, si en toda la línea existe un único interruptor abierto, se espera que haya una diferencia de tensión de V_{ALIM} entre los terminales del interruptor. Esto se debe a que, al abrir el circuito, no va a circular corriente por la línea. Entonces, el terminal positivo tendrá una tensión de V_{ALIM} y el terminal negativo una tensión de tierra, ya que la electroválvula no va a introducir ninguna caída de tensión por la falta de corriente. Por lo tanto, se pueden resumir la diferencia de tensión esperada en los terminales de un SIS en la tabla 3.2.1.

Estado de llave	Tensión entre terminales
Cerrada	0 V
Abierta	V_{ALIM}

Tabla 1: Diferencia de tensión entre los terminales de un SIS

Los rangos de tensión de alimentación de las baterías de las formaciones consideradas son $72VDC < V_{ALIM} < 110VDC$. Y considerando que estos buses de mayor tensión pueden no tener una puesta de tensión en común con el SAL/T, se decidió utilizar un circuito de medición diferencial para calcular la tensión entre las terminales de un interruptor independientemente de su tensión contra la tierra del circuito y luego conectar la tensión medida en un conversor analógico-digital (ADC) del microcontrolador para poder obtener de manera digital un valor para la tensión.

Con el objetivo de alterar lo menos posible la línea de las señales críticas, se optó por utilizar un amplificador de instrumentación para poder medir la tensión diferencial entre los terminales de las llaves. Este dispositivo permite medir una tensión diferencial rechazando cualquier señal de modo común con una gran impedancia de entrada lo que permite derivar una pequeña corriente de la línea principal sin alterarla. El modelo seleccionado fue el INA823DGKR de Texas Instrument [43], un amplificador de instrumentación con salida *single-ended* con un rango de ganancia programable de 1 a 1000 con una resistencia externa, con una baja corriente de polarización de entrada de 1,2 nA y permite la alimentación con una fuente dual de $\pm 1,1$ a ± 18 V. Para la aplicación en el SAL/T, se va a utilizar una ganancia de 1 (sin resistencia externa conectada) y se va a alimentar el amplificador con una tensión dual de ± 5 V.

Para generar la tensión de -5 V a partir de la fuente de alimentación de +5 V, se utilizó el regulador de tensión ICL7660AIBAZA de Renesas Electronics [44] diseñado para invertir una tensión de entrada positiva y generar una salida negativa. Este permite la inversión de la tensión de entrada en el rango de 1,5 a 12 V, tiene un bajo consumo de corriente de 80 μ A, funciona con un oscilador interno a 10 kHz que elimina la necesidad de generar una señal de reloj externa y solo requiere dos capacitores externos. El circuito integrado (CI) viene en un encapsulado VSSOP-8 para montaje superficial. En la figura 3.5, se observa la conexión del ICL7660AIBAZA siguiendo las recomendaciones de la hoja de datos para una tensión de entrada de +5 V; se utilizaron dos capacitores electrolíticos de 10 μ F y se obtuvo una tensión de salida de -5 V.

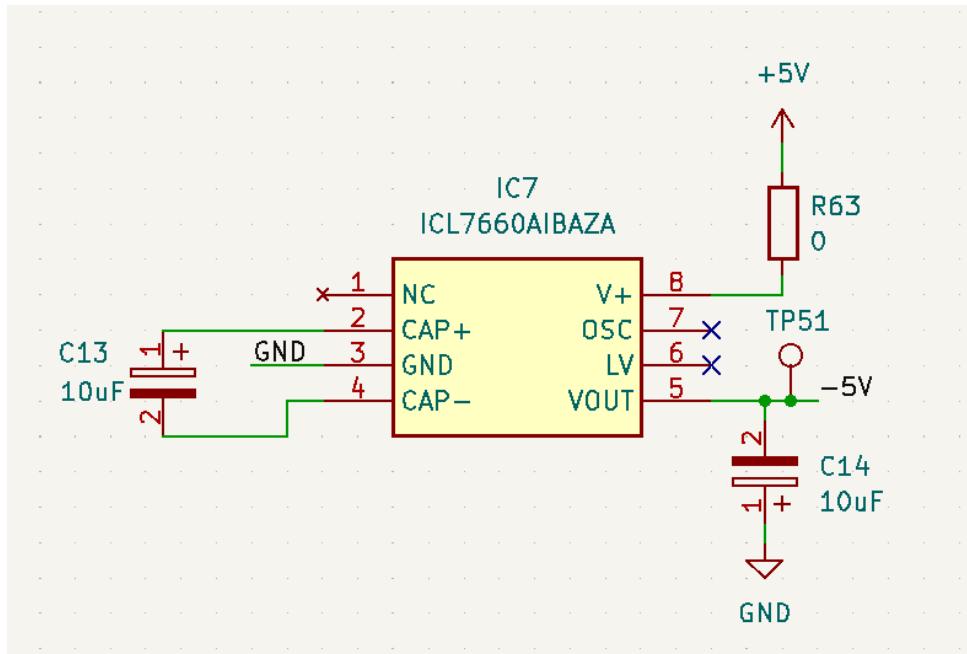


Figura 3.5: Esquemático del circuito para generación de tensión de alimentación negativa

Para poder medir la tensión diferencial en las terminales de las llaves del SIS utilizando el INA823DGKR alimentado con ± 5 V, es necesario adaptar las tensiones a medir a un rango dentro de los extremos de la alimentación, ya que podría haber una diferencia de hasta 110 V sobre esa llave. Para reducir esta tensión diferencial, se implementó una red de resistencias que permiten mantener la diferencia de tensión, pero reducida por un factor seleccionado y utilizando una tierra como referencia para tener las tensiones positivas y negativas a la entrada del amplificador de instrumentación.

Si bien el rango máximo para las entradas podía ser ± 5 V, considerando que la ganancia del amplificador es 1 y que la salida va a ingresar a un ADC del microcontrolador que trabaja en un rango de hasta 3,3 V de entrada, hay que reducir el rango máximo esperado hasta esa tensión. Por lo tanto, si consideramos 110 V como la máxima tensión diferencial en la llave del SIS, y armando una red de resistencias que divida esta tensión a la mitad colocando una tierra en la mitad del circuito, se debe calcular los valores de las resistencias tal que, en cada entrada del amplificador, haya como máximo una tensión de $\pm 1,65$ V, ya que la salida es *single-ended* y para esos valores máximos usaría el rango completo de 3,3 V del ADC. Considerando un divisor resistivo para cada terminal del SIS, que se conecten con una tierra a la mitad, se pueden calcular los valores de las resistencias con las siguientes ecuaciones:

$$\frac{V_{max_ADC}}{2} = \frac{V_{max_SIS}}{2} \cdot \frac{R2}{R1 + R2} \quad (1)$$

Despejando el divisor resistivo invertido para facilitar la selección de valores de resistencia y utilizando $V_{max_ADC} = 3,3$ V y $V_{max_SIS} = 110$ V:

$$\frac{R1 + R2}{R2} = \frac{V_{max_ADC}}{V_{max_SIS}} = 33,33 \quad (2)$$

Esto significa que la resistencia R1 tiene que ser, aproximadamente 33 veces más grande que R2. Además, considerando que se busca extraer la menor corriente posible de la línea de los SIS, se van a utilizar valores de resistencia muy altos así como también el amplificador cuenta con una impedancia de entrada muy alta. Con estos cálculos realizados, se seleccionaron los valores de $R1=1\text{ M}\Omega$ y $R2=30\text{ k}\Omega$ lo que genera un divisor resistivo de 34,33 veces la tensión original;

esto permite darle un poco de margen al rango máximo calculado.

Con las distintas partes del circuito resueltas, en la figura 3.6 se muestra un esquemático del circuito de medición de los SIS con los elementos explicados donde también se agregan 2 capacitores de desacople en la alimentación del amplificador para filtrar cualquier ruido que pueda venir en la línea.

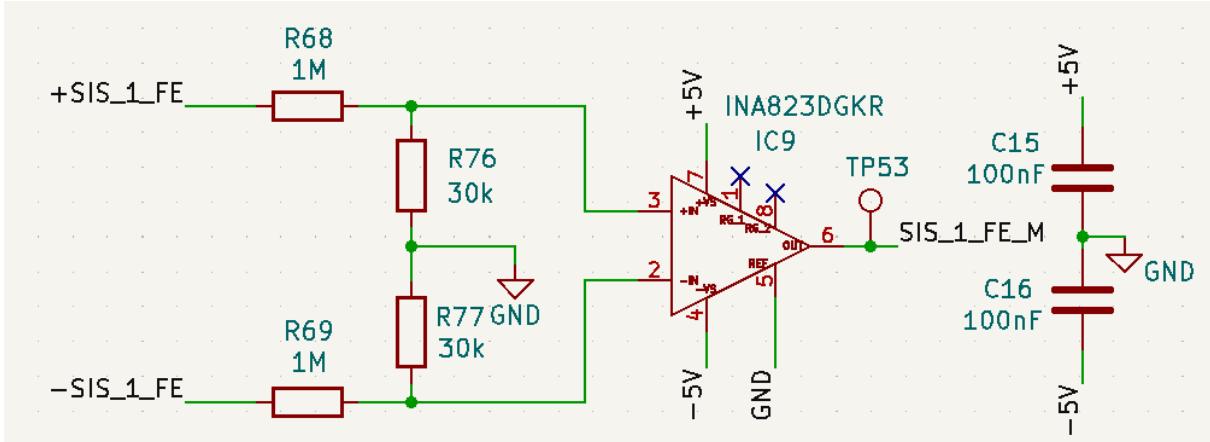


Figura 3.6: Esquemático del circuito de medición de los SIS

3.2.2 Aislamiento de los SIS

El propósito del SAL/T es permitir el movimiento de la formación de una manera segura ante una activación de las interrupciones de los SIS. Para eso, se produce una interacción con los sistemas instrumentados de seguridad en la medición del estado impuesto por los propios SIS, pero también en una imposición de estado por parte del SAL/T. La llave interruptora de los SIS se abre ante una falla o detección de condición insegura en ese SIS. En el caso de ingresar al modo aislado limitado, el sistema va a utilizar las mediciones de estado de cada uno de los SIS y aislar el que esté interrumpiendo la línea; de esta manera, la línea mantiene la continuidad, y el resto de los SIS sigue teniendo la capacidad de interrumpir las señales críticas. Además, cuando se ingresa en modo aislado total o en modo intermitente por comando remoto, o en modo parada total o en modo coche deriva, el SAL/T va a interrumpir las líneas de las señales críticas dejando sin efecto el estado de los SIS.

Para realizar el aislamiento de un SIS, se conectaron las mismas terminales donde se mide el estado del SIS a un relé. Su estado normal es abierto, por lo que no afecta el estado del SIS. Sin embargo, cuando se activa la señal de control del relé por algunos de los motivos mencionados anteriormente, se conectan las señales positiva y negativa de la llave del SIS, forzando un estado de continuidad en la línea.

Para el relé de aislamiento, se eligió el modelo JW2SN-DC5V de Panasonic Industrial Devices [45]. Este es un relé de potencia que se activa con 5 V en las entradas de control y que cuenta con un arreglo de contacto de 2 salidas tipo C conocido como DPDT (*double-pole double-throw*). En condiciones normales, tiene un consumo de 106 mA al activarse, ya que la resistencia de la bobina interna es de 47Ω .

Se eligió este modelo porque funciona como exacto reemplazo de un relé de seguridad de las mismas características de encapsulamiento y contactos como el V23047A1005A501 de TE Connectivity [46]. Estos relés de seguridad también son DPDT y activan con 5 VDC. Los relés de seguridad o también conocidos como relé de contactos guiados forzados tienen una configuración y técnica constructiva que cumplen con la norma EN 61810-3 [47]. Estos relés tienen al

menos un par de contactos normal cerrado y otro par normal abierto que están vinculados de manera mecánica lo que hace imposible que ambos estén en contacto simultáneamente; incluso en caso de falla. En la figura 3.7 se observa el funcionamiento de un relé de seguridad. Este tipo de relé es recomendado para aplicaciones en sistemas críticos porque permite utilizar el contacto primario como actuador, mientras que el contacto secundario se puede utilizar para monitorear su correcto funcionamiento.

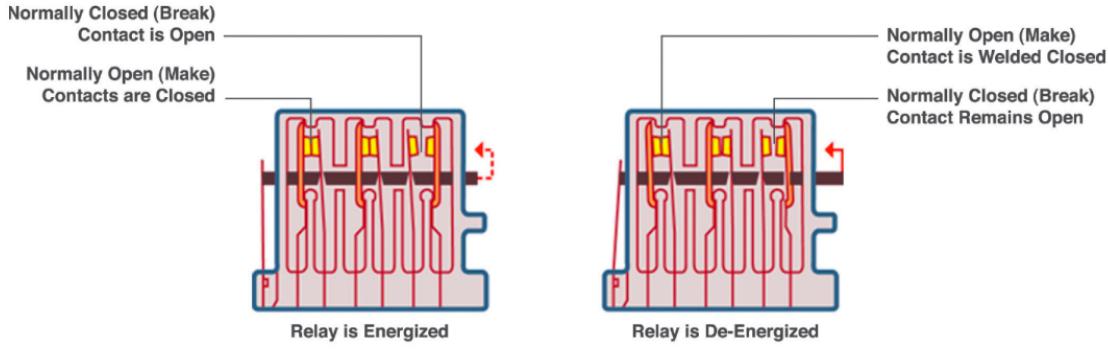


Figura 3.7: Funcionamiento de un relé de seguridad [48]

El modelo utilizado en este prototipo, el JW2SN-DC5V es un relé de uso general, pero se optó por reemplazar el relé de seguridad por este modelo de uso general por la gran diferencia que hay en los costos entre los modelos y considerando que el sistema total hace uso de más de 20 relés.

Una de las salidas del JW2SN-DC5V se va a utilizar como actuador y la otra para medir el correcto funcionamiento del relé. Para eso, las salidas del contacto secundario van a estar conectadas una a tierra y la otra a una tensión de 3,3 V y la llave secundaria que conmuta va a estar conectada a una entrada digital del MCU para medir el estado de esa llave. A partir de este estado, el sistema puede determinar luego si hay un estado de error y debe tomar medidas para llevar el sistema a un estado seguro.

Para poder energizar el relé con una corriente de poco más de 100 mA es necesario utilizar un *driver* de potencia, ya que la corriente es demasiada alta para drivear directamente desde el MCU además de que la tensión de activación es de 5 V y el MCU tiene salidas digitales de 3,3 V. Considerando que el sistema contempla cinco SIS y cada uno tiene dos llaves interruptoras de posible aislamiento, se diseñó un circuito que contemple un mismo *driver* de potencia para múltiples señales.

Se eligió el circuito integrado ULN2003D1013TR de STMicroelectronics [49]. Este CI contiene 7 *drivers* NPN Darlington [50] donde cada uno puede manejar hasta 500 mA y soportar tensiones de 50 V. Incluye diodos para cargas inductivas lo que lo hace ideal para controlar relés. Para el aislamiento de los SIS, se van a utilizar 2 ULN2003 para manejar los diez relés de aislamiento.

En la figura 3.8, se observa el esquemático del *driver* de potencia ULN2003 y un relé con su salida primaria conectada a los terminales del SIS y la salida secundaria a un pin de entrada del MCU. También se observa que se agregó un LED sobre las entradas de la bobina del relé; esto sirve para identificar el estado del relé durante las pruebas.

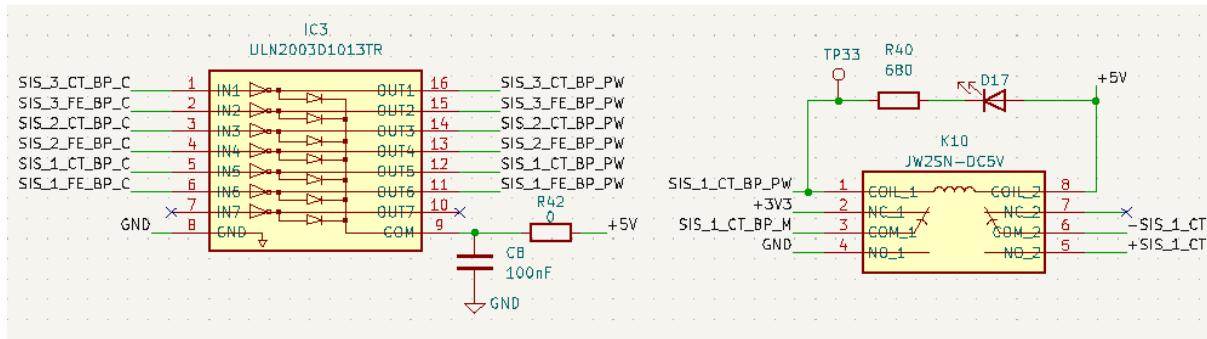


Figura 3.8: Esquemático del circuito de aislamiento de un SIS

3.2.3 Control de señales críticas

El SAL/T debe tener la capacidad de controlar directamente las señales críticas de control de tracción (CT) y freno de emergencia (FE). Estas señales son las más críticas que interactúan con el sistema, ya que pueden controlar el movimiento de la formación, bloquearla o permitirle el avance ignorando el estado de todos los SIS. Por estos motivos, se implementó un circuito redundante de mayor seguridad para la activación y corte de estas señales; uno para la señal de CT y otro para la señal de FE como ya se explicó que trabajan con líneas independientes. Este circuito reutiliza los conceptos del prototipo anterior explicados en su informe [4].

El circuito de control de las señales críticas cuenta con dos entradas y una salida. Una entrada es la línea de la señal crítica después de haber pasado por las llaves interruptoras de los distintos SIS y de los circuitos de aislamiento de estos (IN); la otra entrada es una alimentación provista por la formación capaz de energizar la electroválvula (ACT); se espera una tensión de entre 72 V y 110 V acorde a la tensión de las baterías del tren; pero al trabajar con relés, solo se va a conectar o desconectar esta señal de la línea y no se va a trabajar en su procesamiento. La salida (OUT) va directamente a la conexión con la electroválvula. En la figura 3.7 se observa cómo es la configuración del sistema de activación de una señal crítica respecto a los interruptores de los SIS y sus sistemas de aislamiento (expuesto de manera simplificada) y respecto a la electroválvula de la señal crítica; también se muestra el arreglo de tres relés con sus respectivas entradas y salidas mencionadas.

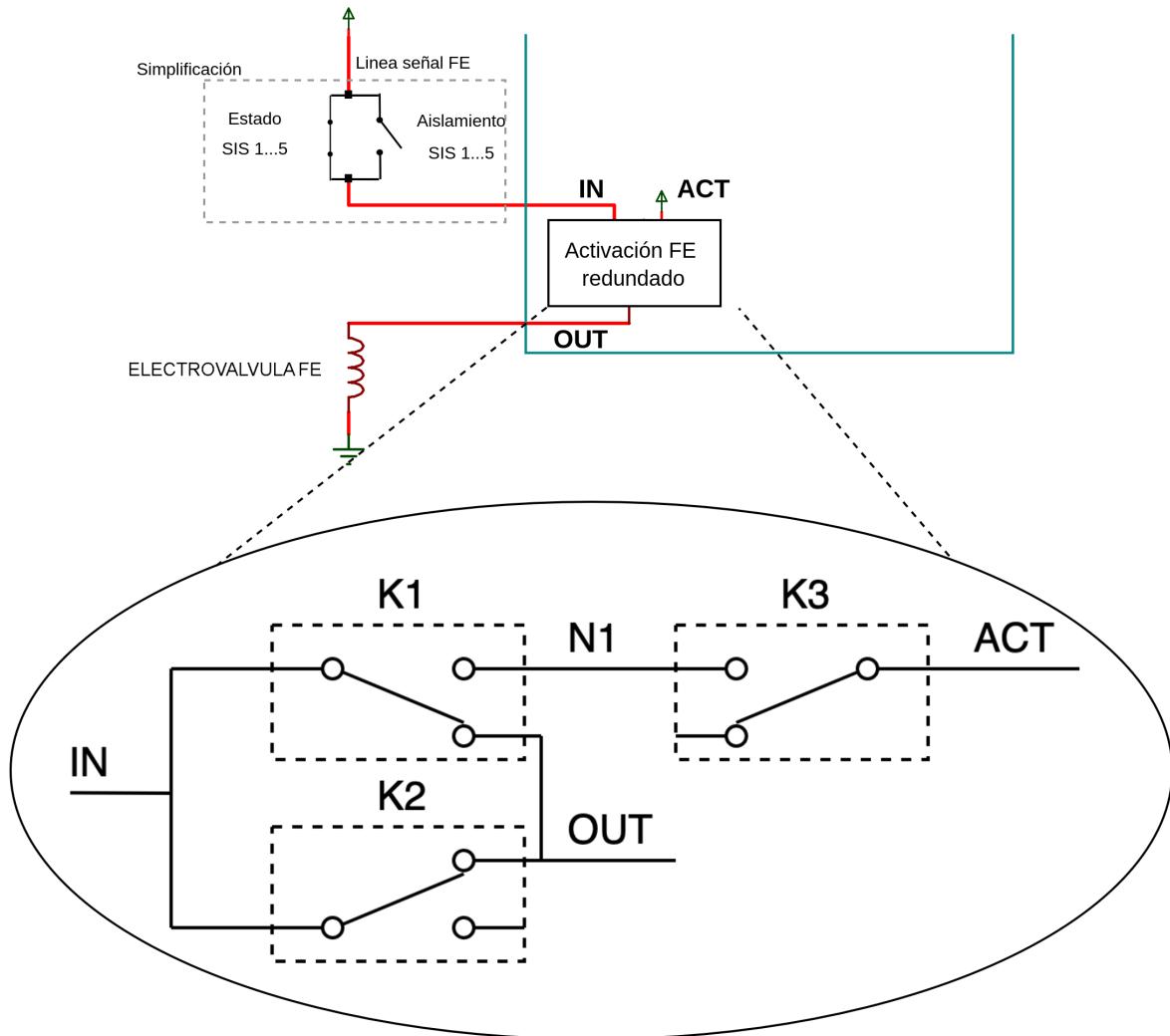


Figura 3.9: Circuito de relés redundado para activación de señales críticas

Para estos relés, se utilizó el mismo modelo que para el aislamiento de los SIS (JW2SN-DC5V) pero con la misma consideración de que sirven de reemplazo para relés de seguridad de las mismas características de contactos. Para alimentar estos relés, se utilizó un nuevo driver de potencia ULN2003 que permite activar los seis relés (tres para CT y tres para FE) de estos circuitos.

Los estados del circuito se pueden resumir de la siguiente manera:

- **Estado normal:** el circuito no modifica la línea de la señal crítica. La entrada IN queda conectada directamente con OUT. La señal de ACT no interviene en la línea. Este estado no realiza ninguna modificación respecto a la línea tradicional sin el SAL/T y debe asegurarse que, ante una falla en el sistema, se imponga este estado dejando la activación de la señal crítica controlada por los SIS.
- **Activación señal crítica o circuito abierto:** la salida OUT queda desconectada de las entradas IN y ACT. De esta manera, se asegura que el circuito queda abierto y la electroválvula quede sin alimentación lo que fuerza que la señal crítica se ejecuta. Este estado debe forzarse cuando se desea detener la formación activando el freno de emergencia o anulando el control de tracción.
- **Alimentación forzada o bypass:** la salida OUT se conecta directamente con la entrada

ACT. De esta manera, la electroválvula queda alimentada independientemente de la tensión en IN y del estado de los SIS de la línea. Este estado es el más crítico porque desactiva todos los SIS de la formación y le permite el movimiento en este estado inseguro.

Para forzar los estados mencionados, se deben controlar los relés considerando que la figura 3.9 muestra el estado normal (desenergizado) de cada uno de ellos de acuerdo a la tabla 2.

Estado del circuito	Posición de los relés
Estado normal	K1, K2 y K3 desenergizados
Circuito abierto	K1 y K2 energizados. K3 desenergizado
Bypass	K1 y K3 energizados. K2 desenergizado

Tabla 2: Posición de relés para los estados del circuito redundante

Este sistema aumenta la seguridad de las conexiones, ya que los relés K1 y K2 cumplen una función de redundancia para volver al estado normal. En caso de que alguno de los tres relés falle, esto significa que quede pegado en una posición no deseada, se pueden utilizar los otros dos relés para asegurarse de volver al estado normal donde IN está conectado con OUT. Por ejemplo, si el relé K1 falla y queda pegado conectado al nodo N1, la conexión entre IN y OUT va a seguir existiendo y no se va a generar ningún problema mientras K3 esté desenergizado. Si K2 falla, K1 conecta directamente IN y OUT. Si K3 falla, se mantiene el estado normal, ya que el nodo N1 no participa de este estado.

Al utilizar relés con dos salidas, una para este circuito de activación y otra para la medición del estado del relé, se puede detectar en todo momento cuando un relé quedó en un estado incorrecto y forzar al SAL/T en un estado de error interno restaurando la conexión normal de las señales críticas.

3.2.4 Registrador de eventos y conector de zona

Las formaciones ferroviarias cuentan con un registrador de eventos que es un dispositivo que monitorea y almacena datos clave sobre el funcionamiento del tren, similar a una caja negra en aviones [51]. Registra información como la velocidad, la activación de frenos, la posición del acelerador, el estado de las señales y otros parámetros críticos durante el viaje. Estos datos son fundamentales para analizar el rendimiento, investigar accidentes o fallos, y mejorar la seguridad operativa en el sistema ferroviario. En las formaciones para las que se diseñó el SAL/T, se consideró la interacción con un registrador de eventos modelo Hasler Teloc 1500 [35].

Para el sistema SAL/T se solicitó la comunicación del estado de las siguientes características:

- Estado de la alimentación del SAL/T
- Activación del modo limitado
- Activación del freno de emergencia
- Activación del corte de tracción

La comunicación con el registrador de eventos se debe realizar a través de un camino de baja impedancia entre dos nodos. Para esto, se utilizó nuevamente los relés JW2SN-DC5V con un nuevo ULN2003 como driver de potencia para estos cuatro relés.

Otra comunicación de estado que realiza el SAL/T mediante un quinto relé, también conectado al mismo driver de potencia, es con un conector de zona. Este relé debe activar un camino de baja o alta impedancia dependiendo de la zona en la que circule el tren. La ubicación

es obtenida mediante GPS y, cuando se supera una distancia configurada respecto al punto de la estación de origen del recorrido, se activa un camino de baja impedancia. Esto activa en la formación un cambio en los límites de velocidad de circulación y es utilizado para formaciones que tienen parte de su recorrido en zonas rurales o menos urbanizadas. De esta manera, permite mantener distintos límites de velocidad dependiendo de la zona de circulación de la formación.

3.2.5 Llaves de activación modo aislado limitado y total

El SAL/T permite la activación de los modos aislado limitado y modo aislado total de manera local. Para el modo aislado limitado, existe una llave colocada en el panel frontal del dispositivo que va a permitir la activación del modo. Para el modo aislado total, se coloca una llave por fuera del panel frontal, instalada dentro de algún lugar con acceso restringido al maquinista y solo accesible para el personal de mayor autoridad.

La activación de estas llaves altera completamente el funcionamiento de la formación por lo que la correcta lectura de su estado es un requisito importante para el sistema. Por esto se eligió un modelo de llave interruptora bipolar de dos posiciones, ya que cuenta con dos líneas y permite medir el estado en dos entradas diferentes del MCU y asegurarse de leer la posición correcta. El modelo seleccionado es una llave interruptor Bipolar 20 A de dos posiciones de la empresa Elibet [52] que se puede visualizar en la figura 3.10 y está diseñada para ser colocada en un panel frontal.



Figura 3.10: Llave interruptor bipolar 20A Elibet 0-1 posiciones

En la figura 3.11 se muestra el circuito implementado para medir el estado de cada llave. Se conectaron resistencias de *pull-up* de $40.2\text{ k}\Omega$ y se conectaron las entradas de la llave a tierra. Cuando la llave está abierta, las señales que llegan al MCU, ON_SW_MAL_1 y ON_SW_MAL_2 quedan a 3,3 V y el MCU toma esos nodos como entradas digitales por lo que va a detectar un estado alto. Cuando la llave se cierra, los nodos quedan conectados a tierra y el MCU va a leer un estado bajo. En caso de inconsistencia entre los valores leídos entre las dos entradas de una misma llave, se considera el estado de no activación de la llave, ya que es el estado de mayor seguridad y no puede asegurarse el correcto funcionamiento del sistema sin una lectura correcta.

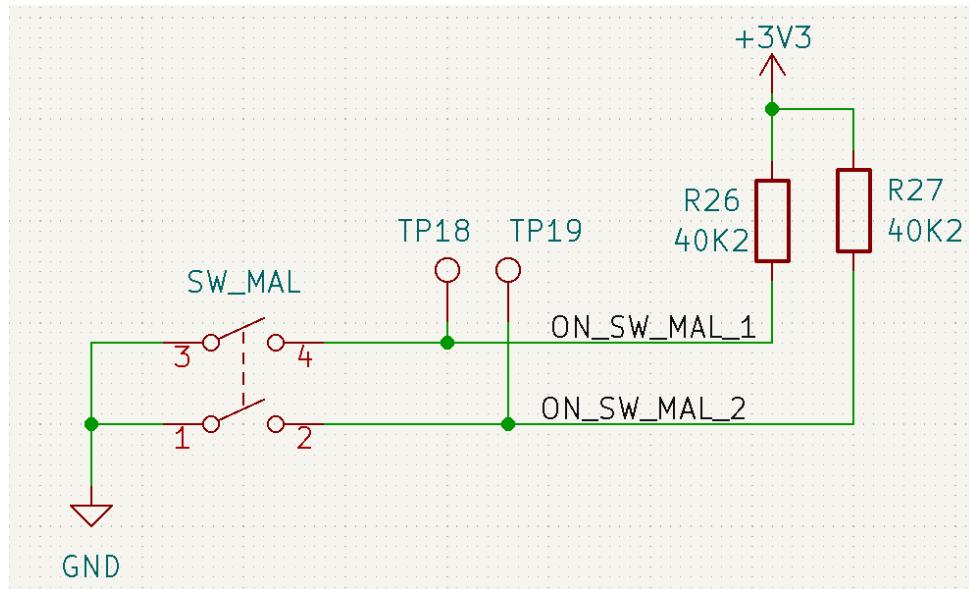


Figura 3.11: Circuito de lectura del estado de la llave interruptora

3.2.6 Lectura de velocidad por RS-485

El SAL/T cuenta con tres fuentes independientes para medir la velocidad de la formación; dos de ellas provienen de una comunicación por el protocolo RS-485 con otros sistemas de la formación, la tercera proviene del módulo GPS instalado dentro del mismo SAL/T.

La fuente de medición de velocidad más confiable, y por lo tanto priorizada por el SAL/T, es la indicada por el registrador de eventos Hasler Teloc 1500. La formación cuenta con un generador de pulsos ópticos Hasler que opera en el dominio infrarrojo conectado al registrador de eventos de la formación. Al mismo tiempo, el registrador de eventos está comunicado con un indicador de velocidad que tiene disponible el maquinista [53]. En la figura 3.12 se observan los dispositivos presentes en la formación.



Figura 3.12: Equipamiento Hasler en la formación ferroviaria

En el prototipo anterior del SAL/T, se realizó una tarea de ingeniería inversa para detectar cómo era la comunicación entre el registrador de eventos y el indicador de velocidad para poder

interceptar esa señal y utilizar el dato de la velocidad dentro del SAL/T [4]. Revisando la documentación del equipo, con la ayuda del personal del laboratorio de material rodante de la estación de Victoria y midiendo las señales en el bus de comunicación entre estos dispositivos, se concluyó que la comunicación utilizaba el protocolo RS-485 *half-duplex* a 115.200 bps. Las pruebas permitieron determinar que la señal cuadrada medida utiliza una tensión de 0,5 V para el estado bajo y de 3,3 V para el estado alto. En reposo, el bus queda en estado alto y la configuración es de 8 bits con un bit extra de stop. La trama enviada está compuesta por paquetes de 31 bytes. En la figura 3.13 se puede observar la medición de un paquete de datos completos realizado en este proceso de descubrimiento.

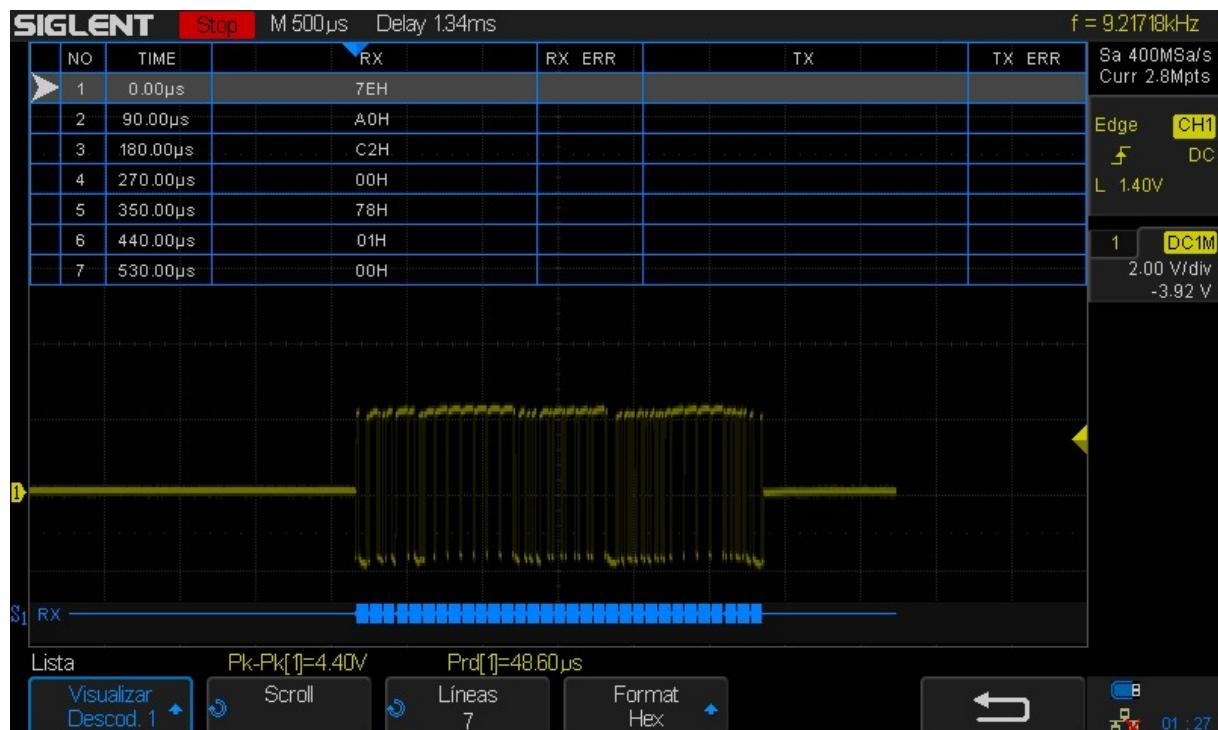


Figura 3.13: Trama emitida por el equipo Hasler Teloc 1500

Las tramas empiezan y finalizan con un byte de start y stop, respectivamente, de valor hexadecimal 0x7E. Conociendo el byte de start y stop es más fácil aislar las tramas y analizarlas de manera independiente. Se determinó que la información asociada a la velocidad está expresada en los bytes 7 y 8 (repetida en los bytes 9 y 10) en formato hexadecimal. También, se determinó que los bytes 29 y 30 funcionan como código de redundancia cíclica en la variante CRC-16/IBM-3740 [54]. El resto de la información contenida en la trama no es relevante para el SAL/T como, por ejemplo, la cantidad de kilómetros recorridos por la formación.

Por otro lado, la segunda fuente de medición de velocidad por RS-485 proviene de un circuito que procesa las señales de un generador de impulsos. El generador de impulsos ópticos es un dispositivo que detecta interrupciones de un haz de luz, generalmente producido por un LED, y convierte esos cambios en señales eléctricas. Se utiliza para medir la velocidad o posición de objetos en movimiento, como ruedas o ejes, generando pulsos proporcionales a la rotación o desplazamiento. En la figura 3.14 se puede ver el principio de funcionamiento de este dispositivo.

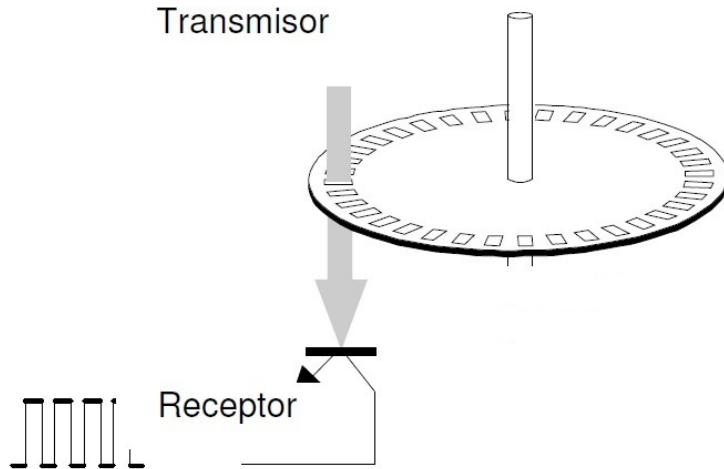


Figura 3.14: Principio de funcionamiento de un generador de impulsos ópticos

Para determinar la velocidad a partir de la señal eléctrica, un sistema con un microcontrolador que tiene previamente configuradas algunas características del generador óptico y de la formación (números de pulsos por revolución, diámetro o circunferencia de la rueda) debe considerar la frecuencia de los pulsos así como el tiempo entre pulsos para una medición más precisa de la velocidad instantánea. Una vez determinada la velocidad, este sistema externo va a montar el dato sobre un bus RS-485 que se va a conectar con el SAL/T comunicando directamente el dato de la velocidad.

Dentro del SAL/T se incluyeron dos circuitos para realizar la conversión entre el protocolo RS-485 y las interfaces UART del MCU. Para armar este circuito, se tomó como referencia el circuito del mismo propósito incluido en la placa EDU-CIAA-NXP [55]. El circuito está basado en el chip SN65HVD1176DR de Texas Instruments [56]; este es un transceptor diferencial que trabaja en *half-duplex* con características optimizadas para su uso en aplicaciones PROFIBUS [57]. El voltaje diferencial de salida del controlador supera los requisitos de PROFIBUS de 2,1 V con una carga de $54\ \Omega$. Una tasa de señalización de hasta 40 Mbps permite un crecimiento tecnológico hacia velocidades altas de transferencia de datos. La baja capacitancia del bus reduce la distorsión de la señal.

El circuito se muestra en la figura 3.15 y se observa en la entrada las señales de recepción y transmisión que se conectan al MCU así como un pin de salida digital que selecciona si se van a transmitir o recibir datos ya que es un sistema *half-duplex* y se comparte el bus para ambas transmisiones. Se observan también dos capacitores de desacople conectados en la alimentación del circuito integrado. Del lado de la salida se colocaron las resistencias de $390\ \Omega$ y $220\ \Omega$ conocidas como *bias resistors* en las recomendaciones de PROFIBUS. Por otro lado, se colocaron los *jumpers* JP6, JP7 y JP8 que deben ser cortocircuitados en caso de ser el último nodo de la red. También se colocó la resistencia R22 de $100\ \Omega$ para conectar la tierra y disipar cualquier pequeña diferencia de tensión que pueda haber entre las distintas tierras de los dispositivos conectados de acuerdo a la nota de aplicación de Texas Instruments SLLA070D [33]; esto permite reducir el ruido de modo común en las líneas de transmisión. El circuito también utiliza el diodo supresor de transitorios (TVS) modelo SZP6SMB12CAT3G de Littelfuse [58] diseñado para proteger componentes electrónicos sensibles de picos de voltaje y transitorios y el fusible reajustable MF-USMF020-2 de Bourns [59] diseñado para proteger circuitos electrónicos contra sobrecorrientes. A diferencia de los fusibles tradicionales, este dispositivo puede ser reajustado manualmente después de que se ha activado, lo que proporciona una solución más conveniente y económica.

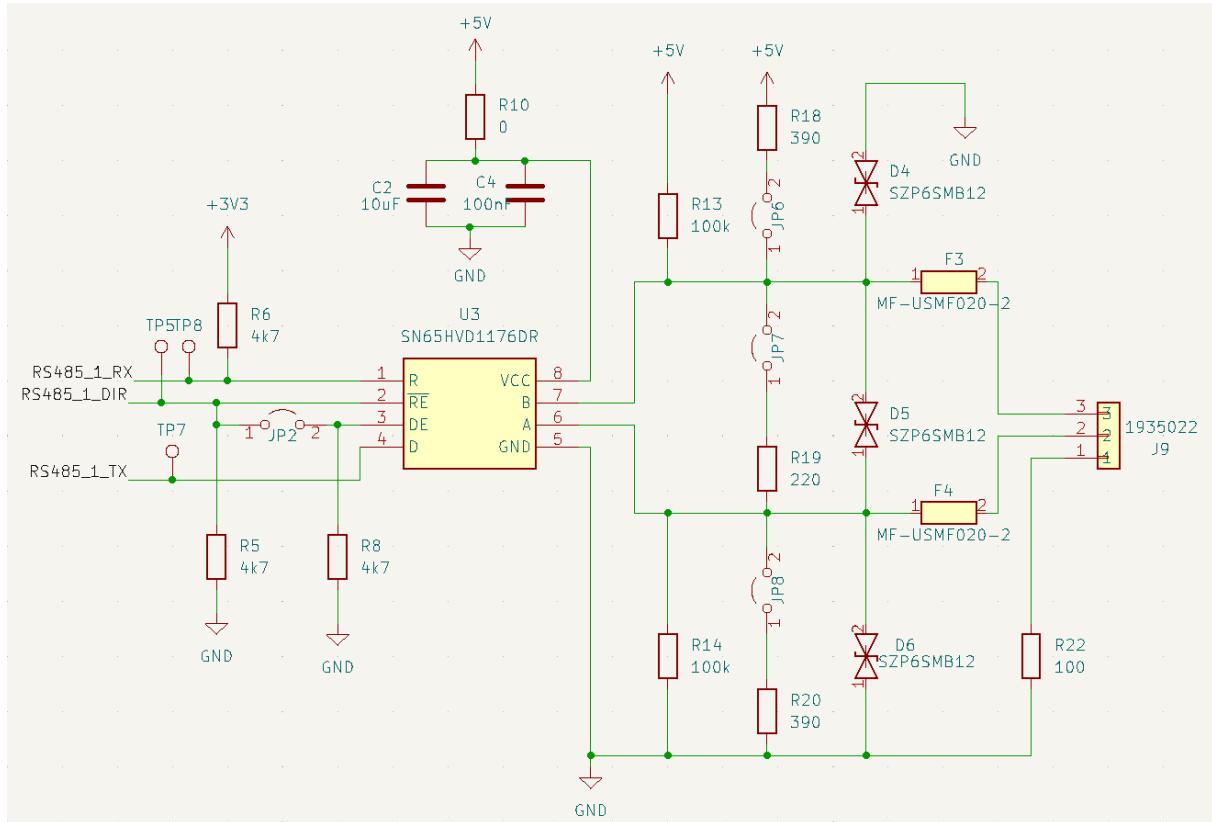


Figura 3.15: Circuito conversor entre protocolos UART y RS-485

3.2.7 Módulo GPS

El sistema utiliza el módulo GPS GY-NEO6MV2 para determinar la posición y velocidad del material rodante. Este módulo basado en el chip NEO-6M se alimenta con 5 V y permite la comunicación con el MCU a través de una interfaz UART. Sin embargo, no cuenta con una entrada de reset y a partir de experiencias previas de otros proyectos con módulos de estas características, se decidió armar un circuito de reset externo por en caso de detectar un error o falla en el funcionamiento del módulo.

El circuito de reset utilizado se puede observar en la figura 3.16 y utiliza el transistor MOSFET de canal P SI2343DS-T1-E3 de Vishay Semiconductors [60] diseñado para aplicaciones de commutación y control de carga en circuitos de baja potencia. Tiene una baja resistencia de encendido $R_{ds(on)}$ lo que permite generar una pequeña caída de tensión respecto a los 5 V que se buscan llevar al módulo. Para polarizar el circuito y permitir la circulación de corriente para encender el módulo, se debe forzar la entrada GPS_PW_ON a su estado bajo de 0 V. Para cortar la circulación de corriente y apagar el módulo, se debe colocar la entrada en estado de alta impedancia; esto se logra configurándolo como un pin de entrada.

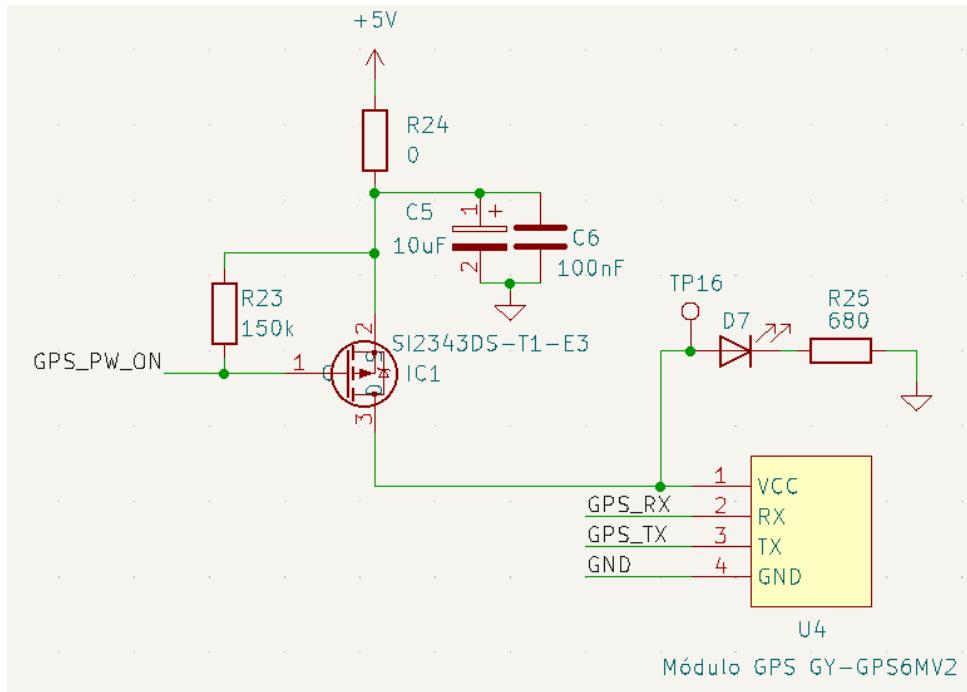


Figura 3.16: Circuito de reset del módulo GPS

Además, se agregó un LED con su respectiva resistencia para indicar cuando el módulo está alimentado y cuando no lo está, ya que el módulo no trae integrado ningún indicador luminoso. También se colocaron dos capacitores de desacople, uno electrolítico de $10 \mu\text{F}$ y otro cerámico de 100nF para filtrar los ruidos que pueda haber en la línea de alimentación considerando también que es un módulo que trabaja con altas frecuencias y genera mayor ruido.

3.2.8 Módulo Wi-Fi y alimentación USB

La comunicación del SALT con la central operativa de control se realiza utilizando la red de Internet. Para lograr esa comunicación, se utilizó el módulo ESP32 Devkit-C que tiene la capacidad de conectarse a una red Wi-Fi. Este módulo permite su alimentación con 5 V ya que tiene un regulador interno que alimenta el MCU interno con 3,3 V. La interfaz con el SAL/T se realiza mediante el protocolo UART y se utiliza un pin GPIO para el pin de EN (*enable*) que trae el módulo, lo que permite forzar un reset del módulo ante cualquier falla detectada. Para facilitar la programación y el manejo de este módulo, se colocó en la placa principal un zócalo que permita insertar y remover el módulo sin necesidad de soldarlo a la placa.

Considerando que la formación puede no tener instalada una red Wi-Fi propia, se consideró la instalación de módulos Wi-Fi - 4G, para poder generar hasta dos redes Wi-Fi utilizando la red de datos celular. Estos módulos utilizan las tarjetas SIM (*Subscriber Identity Module*), pequeño chip utilizado en dispositivos móviles para identificar de manera única a un usuario en una red de telecomunicaciones que contiene información esencial como el número de teléfono, datos de autenticación y almacenamiento de contactos. Con estos módulos, se puede generar una red Wi-Fi utilizando la red de datos de distintos proveedores de servicio de datos celulares. Existen muchos módulos de estas características y son ampliamente utilizados en la vida cotidiana; por ejemplo, la empresa Altanet cuenta con un módem router Portátil USB 4G LTE Wi-Fi [62] que se visualiza en la figura 3.17. Este módulo trabaja con todas las bandas argentinas incluyendo la banda rural y solo necesita insertarle una tarjeta nano SIM y alimentarlo por USB.



Figura 3.17: Módem router Portátil USB 4G LTE Wi-Fi de Altanet

En el diseño del SAL/T, se colocó en la placa dos conectores USB tipo A del modelo 87583-2010BLF de Amphenol FCI [61], con los pines de alimentación conectados a 5 V. Estos conectores están pensados para alimentar estos módulos Wi-Fi - 4G. Los módulos Wi-Fi - 4G no fueron incluidos en el prototipo por su costo y por su fácil reemplazo por cualquier red Wi-Fi generada por un router o teléfono para realizar las pruebas del sistema.

3.2.9 Zócalo para memoria SD

Para incluir una memoria SD en el sistema, se utilizó un zócalo modelo DM3AT-SF-PEJM5 de Hirose Connector [63]. Este conector admite memorias micro SD y tiene un sistema de *push-pull* que permite manipular más fácilmente la colocación de la memoria. Para la memoria micro SD se utilizó el modelo Micro SD Ultra 8GB Clase 10 de Sandisk [64] porque se reutilizó de otro proyecto. Sin embargo, al ser un componente de simple extracción y reemplazo, se considera que se debe utilizar una unidad de almacenamiento de mayor memoria.

En la figura 3.18 se observa la conexión del circuito con el zócalo para micro SD. Se utilizó la interfaz SPI del MCU para la comunicación con la SD y se colocaron algunas resistencias de *pull-up* recomendadas en los buses de datos o líneas desconectadas. También se incluyeron capacitores de desacople para reducir el ruido en la alimentación de la SD. El zócalo incluye además una conexión mecánica que permite determinar si existe una memoria colocada en el zócalo o no; esto se conecta con una entrada del MCU para poder considerar el estado físico real de la memoria.

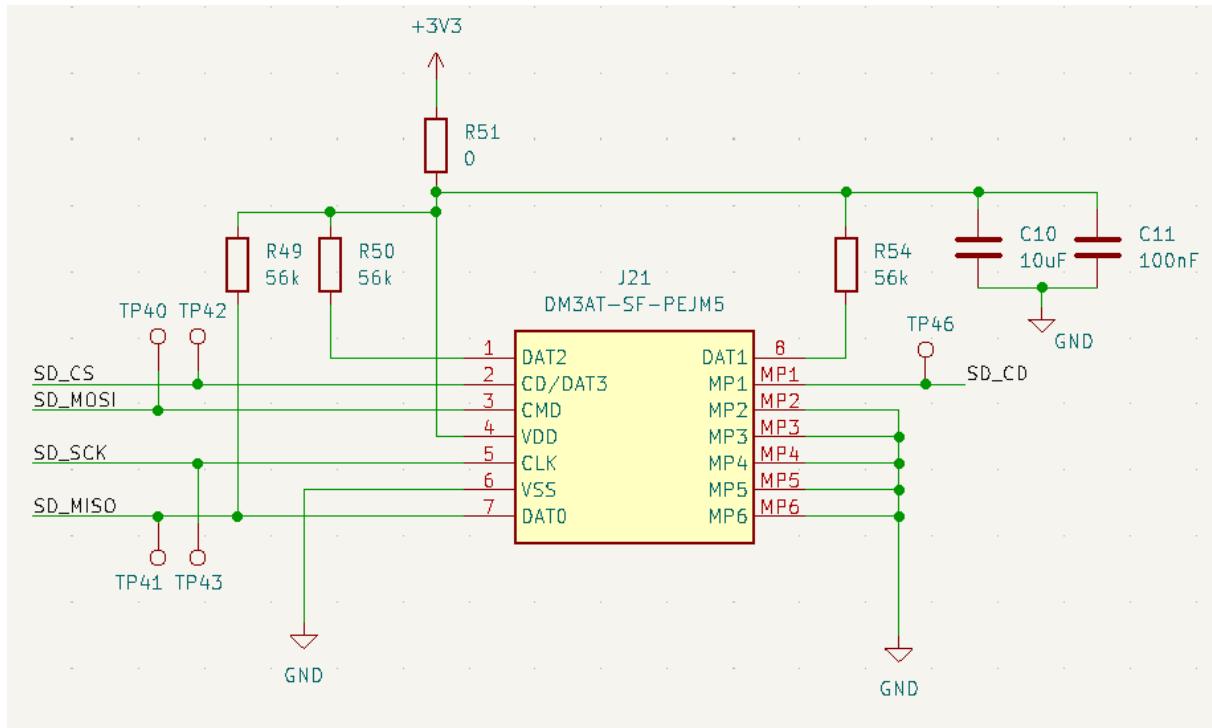


Figura 3.18: Circuito de conexión del zócalo para memoria SD

3.2.10 Microcontrolador y placa STM32 Nucleo-144

El procesamiento central del sistema se realiza en el microcontrolador STM32F429ZI incluido en la placa STM32 Nucleo-144. La placa permite la alimentación del MCU y de todos sus circuitos a través de distintas entradas como se puede observar en la figura 3.19. Para el SAL/T, se utilizó la entrada de E5V para alimentar la placa utilizando la fuente de alimentación del sistema y para eso se debe colocar en el JP3 un cortocircuito entre los pines 1 y 2 para tomar la tensión de allí. Se observa que la placa incluye un circuito de regulación de tensión a 3,3 V para alimentar el MCU y otros circuitos internos.

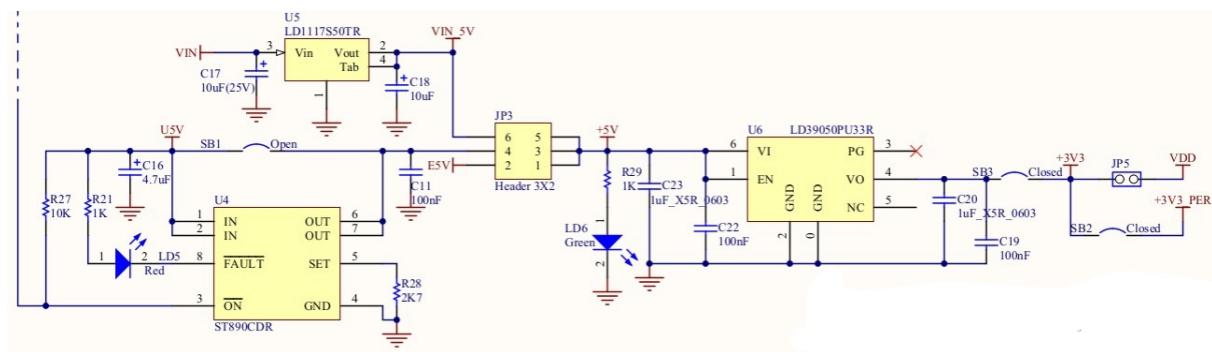


Figura 3.19: Esquemático de la alimentación de la placa STM32 Nucleo-144

Para la inserción de la STM32 Nucleo-144 en la placa principal del SAL/T, se utilizaron los conectores de expansión morpho que permiten el acceso a todos los pines del MCU al usuario. Para eso, se soldaron los pines en la Nucleo y se colocaron conectores hembra en la placa principal. Se necesitaron dos conectores de 2x36 pines cada uno para conectar la placa. Este tipo de conectores permite colocar y sacar la placa fácilmente lo que permite manipularla mejor para la programación y cualquier prueba de circuitos que se quiera realizar sin la placa colocada. En la figura 3.1 se mostraron las distintas interfaces utilizadas del microcontrolador en las que se incluye ADC, SPI, UART, I2C y GPIO.

3.2.11 Panel frontal

El panel frontal del SAL/T sigue el diseño de la figura 3.2 utilizando una placa secundaria donde se colocan directamente los LEDs (*Light-Emitting Diode*), el display de velocidad, el botón seleccionador de perfil intermitente y un buzzer sonoro.

El panel frontal cuenta con cuatro LEDs verdes utilizados para indicar el estado de la alimentación, el modo aislado limitado, el modo aislado total y el perfil intermitente seleccionados; diez LEDs RGB (*Red Green Blue*) para indicar el estado de los SIS, de las señales de CT y FE, la presencia de un comando remoto activo, el estado del receptor GPS y un indicador de la zona de circulación actual de la formación; un botón para seleccionar el perfil intermitente a utilizar; el display de velocidad de cuatro dígitos 7 segmentos y la llave interruptora que queda por fuera de la placa secundaria.

Para controlar todos estos LEDs y el display de velocidad, se utilizó el controlador LED AS1115-BSST de ams OSRAM [65]. Este permite controlar hasta sesenta y cuatro LEDs u ocho dígitos de un display 7 segmentos reduciendo la cantidad de pines necesarios a utilizar del MCU ya que se conecta por protocolo I2C utilizando solamente dos cables. Este controlador permite setear la corriente de cada LED utilizando una resistencia externa; en este caso, siguiendo los valores normalizados y recomendados por la hoja de datos del controlador, se colocó una resistencia de $22\text{ k}\Omega$ para obtener una corriente de 10 mA. El controlador también trae entradas para la detección de hasta diecisésis teclas en una matriz de teclas, pero esa funcionalidad no se utiliza en esta aplicación.

El AS1115-BSST utiliza un método de multiplexado en el cual cada grupo de LEDs o segmentos es controlado secuencialmente, encendiéndolos en rápida sucesión. Esto reduce la cantidad de pines necesarios para el control, ya que todos los LEDs no están encendidos simultáneamente, pero debido a la alta velocidad de escaneo, se perciben como si estuvieran encendidos de forma continua. Además, a través de la interfaz I2C, se envían comandos para controlar el brillo, encendido y apagado de los LEDs o segmentos, y se pueden ajustar los valores de corriente para mejorar la eficiencia energética.

En la figura 3.20 se observa una aplicación típica de este controlador; para el SAL/T, se utilizó una alimentación de 5 V, un $R_{set} = 22\text{ k}\Omega$, la comunicación con el MCU solo con las líneas de SDA y SCL de I2C trabajando a 100 kHz sin el pin de interrupción, no se utilizaron las entradas de detección de teclas, y se conectó al display de cuatro dígitos 7 segmentos y a los LEDs mencionados previamente.

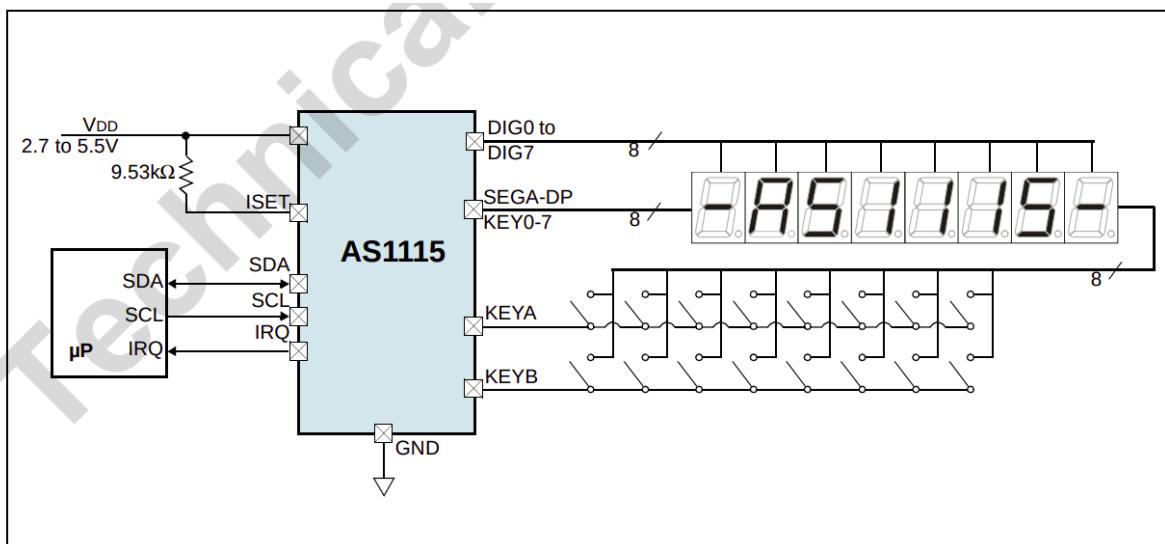


Figura 3.20: Aplicación típica de un controlador AS1115-BSST

Para los LEDs RGB utilizados se seleccionó el modelo WP154A4SEJ3VBDZGW/CA de Kingbright [66] diseñado para aplicaciones que requieren indicación de estado multicolor. Ofrece alta luminosidad y bajo consumo de energía, con un encapsulado transparente de 5 mm, lo que lo hace ideal para proyectos que requieren señalización visual con múltiples colores en un solo componente. Trabajan con una corriente máxima de 20 mA por color, y tienen una caída de tensión e intensidad luminosa para cada color de:

- Rojo: 2 V - 300 mcd
- Verde: 3,3 V - 600 mcd
- Azul: 3,3 V - 500 mcd

Estos LEDs se montan *through hole* y quedan sobresalientes a la placa para luego insertarlos dentro de un gabinete con agujeros para permitir la visualización de estos componentes. Tienen un encapsulado transparente y un lente de tipo redondo con un ángulo de visión de 30°.

Para los LEDs verdes se utilizó el modelo WP7113LZGCK de Kingbright [67] que cuenta con características muy similares solo para el color verde del modelo RGB presentado. Las diferencias son que tiene una tensión $V_f = 2,1$ V, tolera una corriente máxima de 30 mA y su intensidad luminosa es un poco mayor, ya que puede llegar hasta 3000 mcd dependiendo de la corriente.

Para el display de velocidad se utilizó el modelo LTC-5723HR de Lite-On [68]; contiene cuatro dígitos con formato de 7 segmentos de color rojo en configuración cátodo común. Cada dígito tiene un tamaño de 8,1 x 14,2 mm. Funciona con una tensión directa de $V_f = 2$ V y corriente de hasta 20 mA por segmento. Es de montaje *through-hole* lo que permite dejarlo visible por fuera de la placa con cierta separación.

En la figura 3.21 se visualiza el display mencionado junto a un LED RGB y un LED verde utilizado para la implementación.



Figura 3.21: Modelos utilizados para las indicaciones luminosas del panel frontal

Para el botón de selección del perfil intermitente se utilizó un interruptor con pulsador *push switch* modelo de 1543-650-149 de Bourns [69] utilizando un circuito con una resistencia de *pull-down* para forzar 0 V en la entrada del MCU cuando el interruptor está abierto y una resistencia para realizar un divisor resistivo de 5 V (alimentación de la placa) a 3,3 V (tensión de nivel alto de entrada digital del MCU). También se agregó una resistencia en serie de 100 Ohms. En la figura 3.22 se visualiza el circuito.

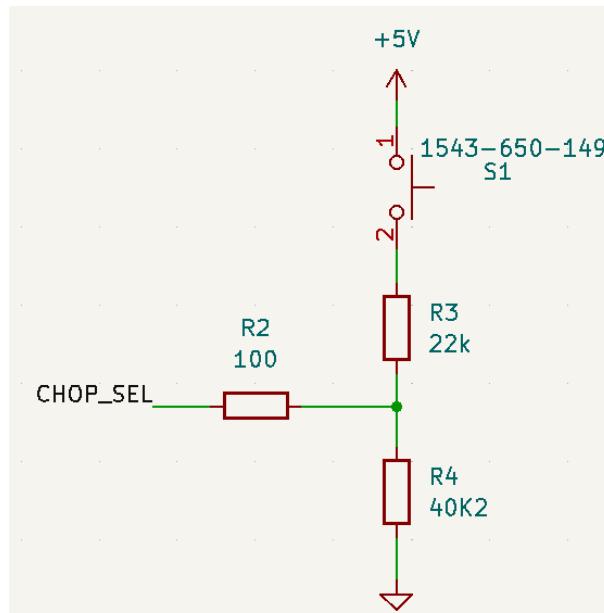


Figura 3.22: Circuito del selector de perfil intermitente

El panel frontal cuenta también con un buzzer sonoro que se activa de manera intermitente cuando se circula en modo aislado limitado y suena de manera continua cuando se exceden los límites de velocidad establecidos. Para el buzzer sonoro se utilizó el modelo CEM-1205-IC de SameSky [70]. Este módulo cuenta con un oscilador interno que al energizarse con 5 VDC emite una señal sonora de 2400 Hz consumiendo 30 mA de corriente. Para el control del encendido de este módulo, se utilizó un transistor MOSFET canal P modelo NVTR0202PLT1G de onsemi [71] conectado como se visualiza en la figura 3.23. Cuando la señal de control BUZZER_C está en estado bajo 0 V, el circuito permite la corriente y alimenta el buzzer sonoro; cuando la señal se coloca en un estado de alta impedancia, no circula corriente apagando el buzzer sonoro. También se colocó un capacitor de $10 \mu\text{F}$ para suavizar el sonido reduciendo los ruidos que puedan entrar por la línea de alimentación.

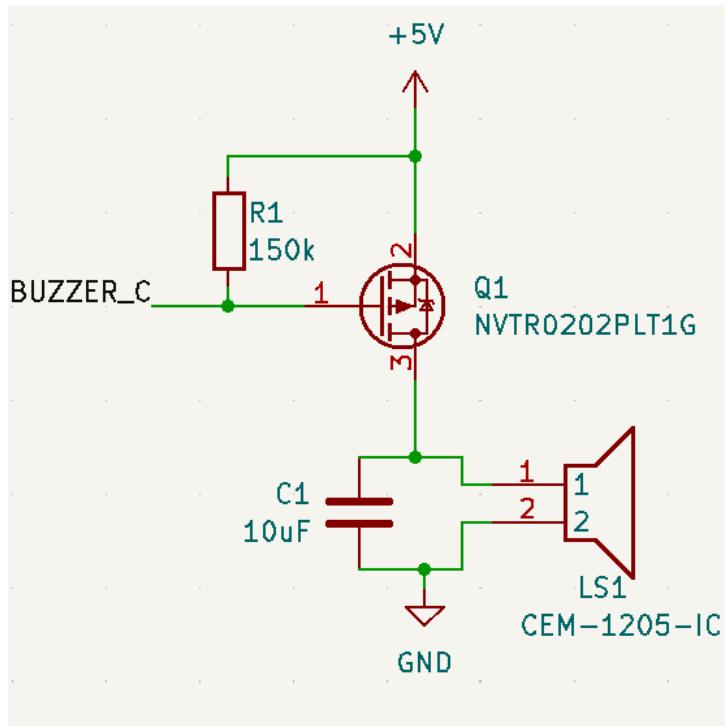


Figura 3.23: Circuito de activación del buzzer sonoro

3.2.12 Fabricación y ensamblado

El diseño de hardware del SAL/T incluye algunos otros componentes no mencionados en los puntos anteriores. Las resistencias utilizadas en el proyecto son resistencias de película gruesa de montaje superficial (SMD), de 1/4 W, tolerancia del 1 % y tamaño 0805; se utilizaron los distintos valores que ofrece la empresa YAGEO para su serie RC. Para los capacitores se utilizaron generalmente capacitores cerámicos multicapa de KEMET, de montaje superficial tamaño 0805, tolerancia del 10 % y operan hasta 16 VDC. Para los capacitores electrolíticos, se utilizaron capacitores de aluminio de Panasonic de la serie HA con tolerancia 20 % y operación hasta 35 VDC.

En la tabla 3, se visualiza el listado completo de los componentes (BOM) necesarios para fabricar el prototipo del SAL/T con sus precios en dólares estadounidenses expuestos por el distribuidor Mouser Electronics que suma un total de 363 USD:

	Fabricante	Modelo	Cantidad	Precio unitario	Precio del item
1	KEMET	C1206C106K4RACTU	5	0,35	1,75
2	KEMET	C0805C104K4RACTU	25	0,14	3,50
3	Panasonic	EEE-HAV100WAR	3	0,38	1,13
4	LITTELFUSE	SZP6SMB12CAT3G	6	0,43	2,58
5	Wurth Elektronik	150080VS75000	22	0,19	4,11
6	Lite-On	LTC-5723HR	1	2,99	2,99
7	Bourns	MF-USMF020-2	4	0,34	1,36
8	Vishay	SI2343DS-T1-E3	1	0,75	0,75
9	STMicroelectronics	ULN2003D1013TR	4	0,73	2,91
10	Texas Instruments	INA823DGKR	10	2,35	23,50
11	Renesas Electronics	ICL7660AIBAZA	1	2,03	2,03
12	ams	AS1115-BSST	1	6,58	6,58
13	Phoenix Contact	1757491	8	1,32	10,56
14	Phoenix Contact	1757488	11	1,06	11,66
15	Amphenol FCI	87583-2010BLF	2	1,07	2,14
16	Hirose	DM3AT-SF-PEJM5	1	2,63	2,63
17	Phoenix Contact	1757475	1	0,73	0,73
18	SparkFun	PRT-09044	8	0,38	3,04
19	Panasonic	JW2SN-DC5V	21	2,10	44,10
20	Kingbright	WP7113LZGCK	8	0,36	2,89
21	Kingbright	WP154A4SEJ3VBDZGW/CA	10	1,36	13,60
22	CUI Devices	CEM-1205-IC	1	1,91	1,91
23	onsemi	NVTR0202PLT1G	1	0,38	0,38
24	YAGEO	RC0805FR-104K7L	6	0,10	0,60
25	YAGEO	RC0805FR-10100KL	4	0,10	0,40
26	YAGEO	RC0805FR-07390RL	4	0,10	0,40
27	YAGEO	RC0805FR-13220RL	2	0,10	0,20
28	YAGEO	RC0805FR-10100RL	3	0,02	0,05
29	YAGEO	RC0805FR-13150KL	2	0,10	0,20
30	YAGEO	RC0805FR-13680RL	22	0,10	2,20
31	YAGEO	RC0805FR-0740K2L	5	0,10	0,50
32	YAGEO	RC0805FR-1056KL	3	0,10	0,30
33	YAGEO	RC0805FR-101ML	20	0,10	2,00
34	YAGEO	RC0805FR-0730KL	20	0,10	2,00
35	YAGEO	RC0805FR-1022KL	2	0,10	0,20
36	Bourns	1543-650-149	1	1,21	1,21
37	STMicroelectronics	NUCLEO-F429ZI	1	24,47	24,47
38	Texas Instruments	SN65HVD1176DR	2	2,94	5,88
39	u-Blox	GPS GY-NEO6MV2	1	13,48	13,48
40	Espressif Systems	ESP32-DevKitC-32E	1	13,87	13,87
41	Harwin	M20-9774046	6	1,47	8,82
42	Elibet	20002/0	2	10,60	21,20
43	Altanet	AltaNet Stick ST10 4G WiFi	2	44,77	89,54
44	Phoenix Contact	1934887	8	1,00	7,96
45	Phoenix Contact	1935022	11	0,71	7,77
46	Phoenix Contact	1934861	1	0,55	0,55
47	Samtex	SSW-119-01-T-S	2	2,44	4,88
48	Adam Tech	RS1-04-G	1	0,23	0,23
49	SparkFun	PRT-16581	4	1,81	7,24

Tabla 3: Listado de componentes del SAL/T

Con el listado de materiales y sus hojas de datos presentes (que pueden ser accedidas mediante el hipervínculo colocado sobre cada modelo de componente en la tabla 3), se calculó el consumo máximo de potencia del SAL/T. La potencia total máxima que puede consumir el sistema, considerando que todos los componentes están trabajando al máximo de consumo lo que es un caso improbable, es de 18,4 W. El principal componente de la potencia son los relés que individualmente consumen 106 mA cada uno al estar activados y hay 21 relés en todo el sistema. Los módulos USB Wi-Fi-4G, considerados para el cálculo de potencia, también tienen un gran consumo de 500 mA cada uno, ya que necesitar generar la señal de la red Wi-Fi. Los módulos de GPS, ESP32 y Nucleo consumen alrededor de 100 mA cada uno. El resto de los componentes suman de manera insignificante al número total. Por eso, se considera necesario una fuente de 5 V con potencia de 20 W para alimentar el SAL/T.

El diseño del sistema considera para todos los circuitos integrados un capacitor de 100 nF de desacople para conectar cerca de su pin de alimentación entre VCC y tierra con el objetivo de filtrar ruidos que puedan estar presentes en la línea de alimentación. Para los casos más sensibles, también se contempló un capacitor electrolítico de 10 μ F con la idea de reforzar el filtrado para ruidos de distintas frecuencias. También, en los esquemáticos se colocó una resistencia de 0 Ω al previo a conectar la línea de alimentación con el pin VCC de los chips para permitir el armado progresivo y segmentado del circuito una vez fabricado. La selección de componentes consideró utilizar la menor cantidad posible de componentes *through-hole* priorizando los de montaje superficial porque en aplicaciones ferroviarias, los componentes *through-hole* pueden sufrir cortes o daños por las vibraciones de la formación.

Para conectar los módulos a la placa, se eligió utilizar conectores hembra tipo zócalo soldados a la placa para permitir colocar y sacar los módulos de manera más simple para su manipulación, programación y eventual reemplazo. Para la conexión con los sistemas externos, se utilizaron conectores de tipo bornera con tornillo soldados a la placa que permiten la conexión de cables. El sistema cuenta con veinte borneras en total y es por eso que cada una tiene identificada su función así como cada pin tiene indicada a qué señal corresponde. Esto aplica también para todos los componentes del sistema que tienen señalado qué número de componente corresponde a cada *footprint* para facilitar el armado y el mantenimiento de las placas ante cualquier inconveniente. También se colocaron puntos de prueba en la mayoría de las señales relevantes para el sistema para poder medir el estado eléctrico de cada señal durante el armado, *debugging* y pruebas del sistema.

Se intentó utilizar anchos de pista grandes para evitar quiebres en las pistas por vibraciones o flexiones de la placa y reducir la resistencia de las pistas. También fue necesario utilizar distintos anchos para poder realizar el ruteo entre pines de la placa Núcleo. Se utilizaron anchos de 0,3, 0,4, 0,5 y 0,8 mm considerando el mayor ancho para las señales de alimentación o que transportan mayor corriente. Las vías se hicieron de 1 mm con agujero de 0,4 mm. Para los conectores de las borneras, se siguió la recomendación de utilizar agujeros grandes de 1,4 mm de diámetro con un anillo anular de 1,6 mm y pistas anchas de 0,8 mm para reducir el daño que pueda causar las fricciones provocadas por el procedimiento de instalación utilizando los tornillos de la bornera. Los puntos de prueba se hicieron de 1,02 mm de diámetro con un anillo anular de 0,51 mm para permitir utilizar la punta de un tester común para medir las señales. Además, las placas cuentan con agujeros para el montaje de la placa en el gabinete que se hicieron siguiendo el formato de tornillo M3 de diámetro 3,2 mm.

En la tabla 4 se visualizan algunas de las restricciones utilizadas como reglas de diseño:

Restricción	Valor
Cobre - mínima distancia entre pistas	0,2 mm
Cobre - mínimo ancho de pista	0,3 mm
Cobre - mínimo ancho anular	0,25 mm
Cobre - mínimo diámetro de vía	1 mm
Holes - mínimo diámetro del agujero	0,3 mm
Holes - mínimo distancia entre agujeros	0,254 mm

Tabla 4: Restricciones de diseño utilizadas para el diseño del PCB

En ambas placas, se utilizó una de las capas como plano de tierra intentando utilizar la menor cantidad de pistas por ese lado mientras que la otra capa se utilizó para hacer todas las conexiones posibles. En las figuras 3.24 y 3.25 se visualizan los trazos de las pistas diseñados en ambas placas:

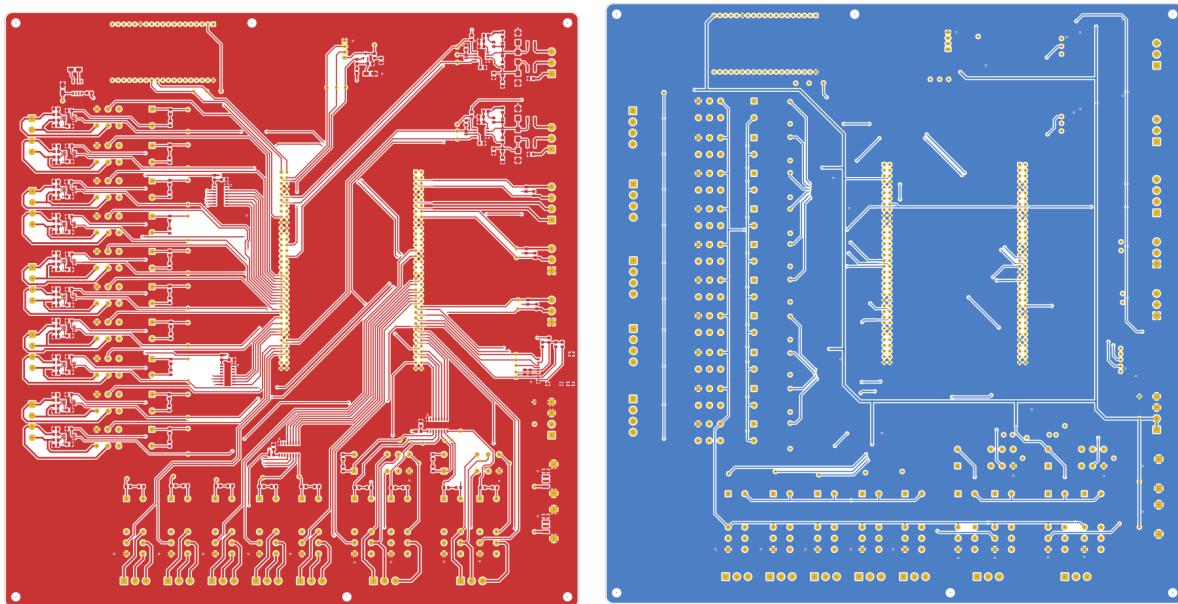


Figura 3.24: Pistas y planos de la placa principal del SAL/T

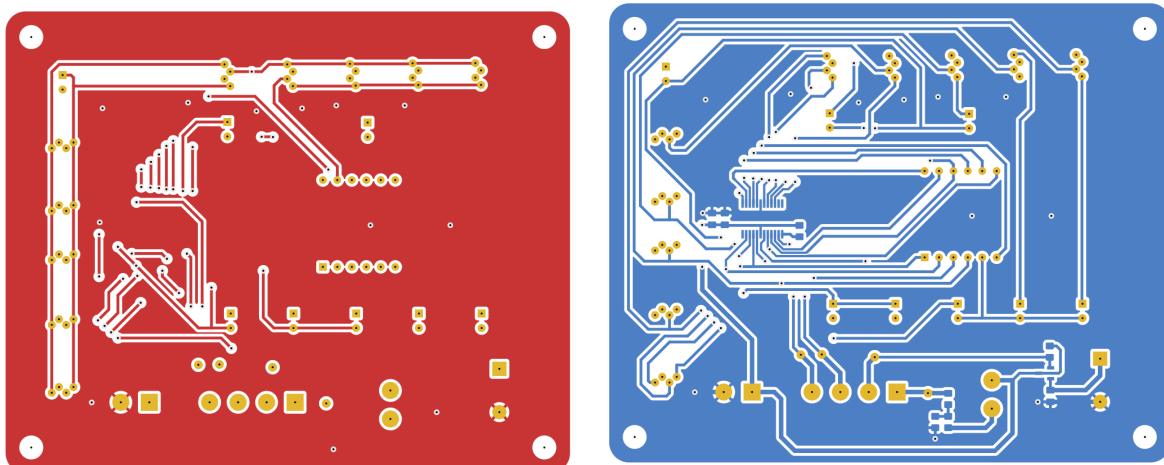


Figura 3.25: Pistas y planos de la placa secundaria del SAL/T

Respecto al PCB (*Printed Circuit Board*), se mandó a fabricar con la empresa JLCWAY [72] en China. Se utilizó el material FR-4, compuesto de fibra de vidrio y resina epoxi, para el

substrato por su aislamiento eléctrico, su resistencia térmica y la relación entre rendimiento y costo, lo que lo hace el estándar de la industria de PCBs. Ambas placas se fabricaron con dos capas, la principal con dimensiones de 270 x 260 mm mientras que la placa secundaria utilizada para el panel frontal mide 82 x 100 mm. Se utilizó un grosor de placa de 1,6 mm, placas de color verde con el *silkscreen* (capa de serigrafía que contiene textos y símbolos impresos sobre la placa para identificar componentes, pines y otras marcas importantes durante la fabricación y ensamblaje) de color blanco. En las figuras 3.26 y 3.27 se visualizan los modelos 3D de las placas diseñadas:

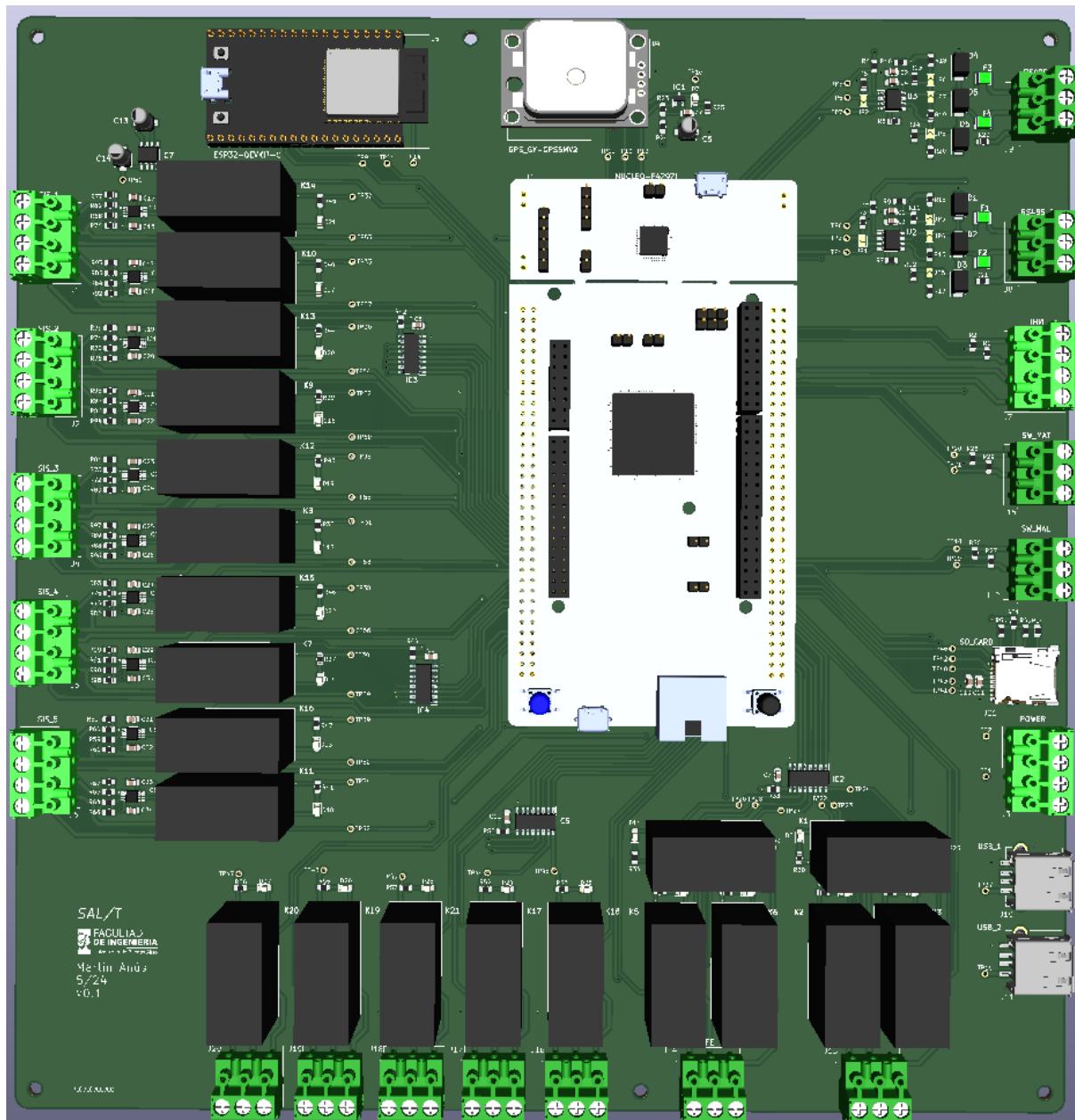


Figura 3.26: Modelo 3D de la placa principal del SAL/T



Figura 3.27: Modelo 3D de la placa secundaria del SAL/T

Los componentes fueron soldados a mano utilizando soldaduras de estaño. En las figuras 3.28 y 3.29 se visualiza la placa principal y la placa secundaria armada con todos los componentes montados.

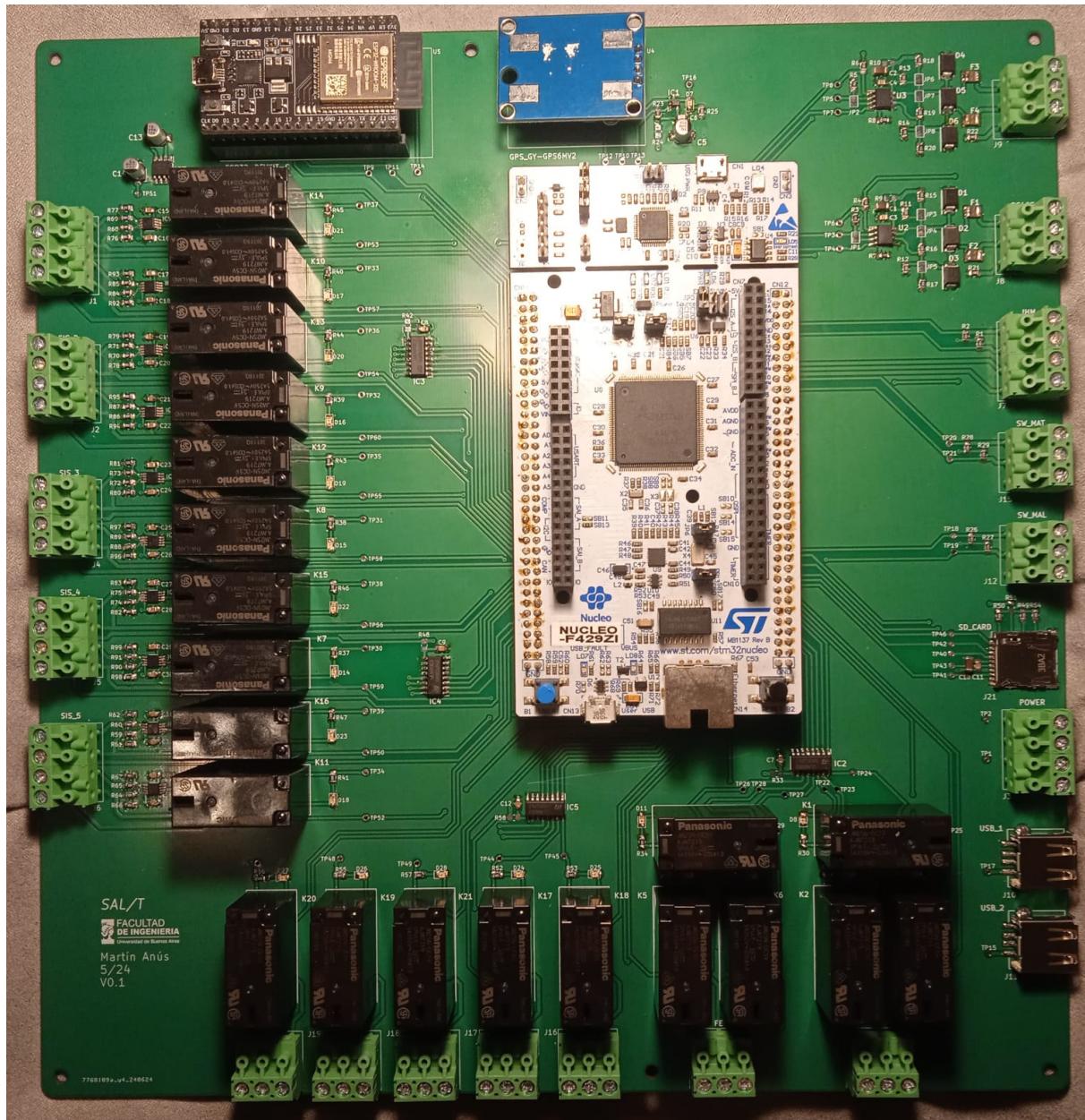


Figura 3.28: Placa principal del SAL/T fabricada



Figura 3.29: Placa secundaria del SAL/T fabricada

3.2.13 Gabinete

Una vez fabricadas y armadas las placas, se avanzó con el diseño de un gabinete para finalizar el diseño de hardware del prototipo. Considerando las medidas de las placas y que una de ellas funciona como parte del panel frontal del SAL/T, se realizó un diseño pensado para su fabricación en impresión 3D con medidas de 300 x 300 x 100mm. El panel frontal cuenta con los agujeros y espacios necesarios para que los LEDs, display y botones de la placa secundaria sean visibles por fuera. En la figura 3.30 se visualiza el diseño del gabinete.



Figura 3.30: Diseño de gabinete para impresión 3D

La fabricación del gabinete estaba pensada para realizarse utilizando la impresora 3D Prusa P3 Steel que tiene el Laboratorio Abierto (LABi) de la Facultad de Ingeniería de la Universidad

de Buenos Aires [73]. Considerando las dimensiones máximas que se pueden fabricar por pieza, se realizó la división del prototipo en 22 piezas que no excedan las dimensiones de 19 x 19cm.

Una vez finalizada la separación de las partes, se realizó la consulta con el LABi para su fabricación y, si bien el diseño y las piezas eran fabricables de manera individual, la fabricación del prototipo entero requería de demasiadas horas de uso de la impresora y demasiado material plástico debido a su gran tamaño y a la velocidad de fabricación del modelo de la impresora, lo que implicaba un costo innecesariamente alto para la fabricación del prototipo.

Por lo tanto, se siguieron las recomendaciones de utilizar un gabinete prefabricado para contener las placas del prototipo y realizar los ajustes necesarios para conseguir un panel frontal indicativo similar al diseñado previamente. Los gabinetes pensados para placas electrónicas son de menor dimensión, por lo que no se pudo conseguir ninguna para el prototipo. Finalmente, se consiguió un gabinete pensado para instalaciones eléctricas de 310 x 310 x 110mm con tapa atornillable que resultó ideal para el prototipo; el modelo es una caja estanca plástica blanca IP65 de Sistelectric, una empresa de Genrod [74]. El gabinete conseguido con las modificaciones realizadas se puede visualizar en la figura 3.31. Las placas están atornilladas al gabinete y montadas sobre unos pequeños separadores para permitir que la placa no queda apoyada ni en contacto con el gabinete.



Figura 3.31: Gabinete acondicionado para el prototipo del SAL/T

Se realizaron las perforaciones necesarias para el panel frontal y la llave de activación. En la figura 3.32 se visualiza el frente del gabinete con la placa insertada y la llave de activación del modo aislado total.

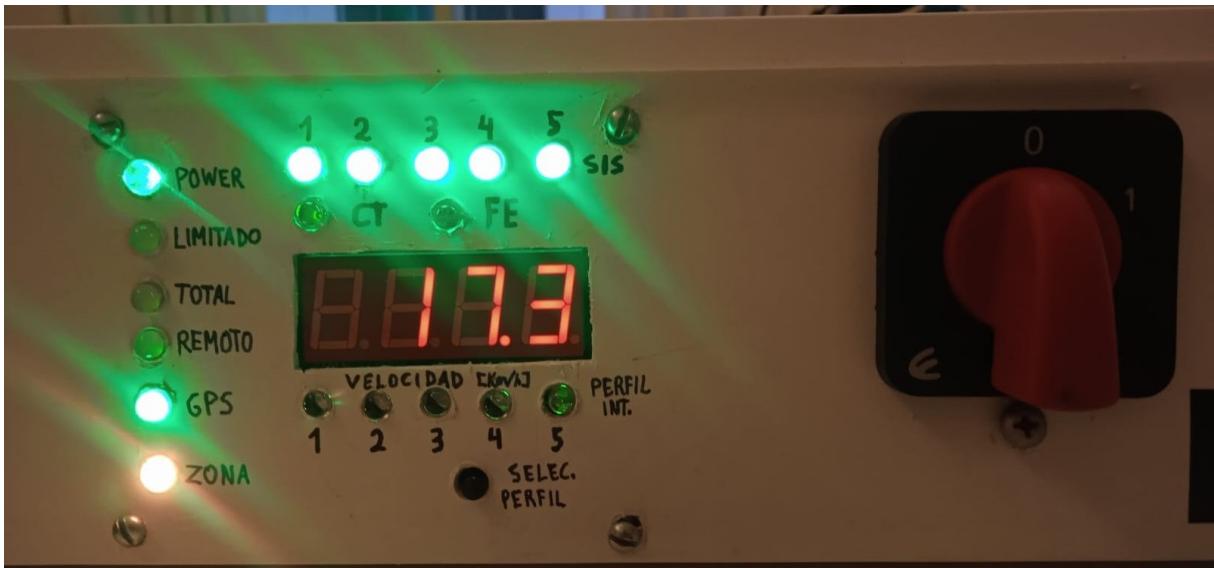


Figura 3.32: Frente del prototipo del SAL/T

3.3 Implementación de firmware

3.3.1 Lógica a nivel general

El SAL/T resulta un sistema bastante complejo por la cantidad de entradas y variantes de comportamiento que considera para determinar su modo de operación y la activación de las señales críticas que resulta el componente de salida más relevante del sistema que le da propósito al SAL/T. La lógica de alto nivel que se puede considerar para el SAL/T se visualiza en la figura 3.33.

Al comienzo, se inicializan todos los periféricos, las comunicaciones, los módulos y la memoria que forman parte del sistema. Por su lado, el ESP32 va a inicializarse y comunicarse con el servidor MQTT a través de alguna de las redes Wi-Fi que tenga configurado, para suscribirse a la recepción de comandos y poder reportar el estado y registro de eventos del sistema de manera remota.

Luego, se leen las entradas del sistema. Por un lado, el sistema lee el estado de las llaves de activación de modo aislado limitado y total, y los comandos remotos para determinar el modo de operación del SAL/T. Por otro lado, se leen las entradas del estado de los SIS, la ubicación GPS, la velocidad que puede ser obtenida por cualquiera de sus fuentes de medición siguiendo la prioridad de Hasler Tesloc 1500, luego el generador de impulsos ópticos y luego la señal GPS.

Una vez que el sistema tiene el dato de todas las entradas, va a mostrar a través del panel frontal y registrar a través del almacenamiento interno y remoto, cualquier estado o cambio relevante del sistema.

Luego, se determina cuál debe ser el modo de operación del SAL/T basado en el primer grupo de entradas mencionado, y se activan las rutinas de entrada en cada modo.

Finalmente, con el modo de operación ya configurado y el resto de las entradas, se determina los valores que deben tomar las salidas de señales críticas como el corte de tracción, freno de emergencia o bypass de los SIS. Al finalizar, se vuelve al comienzo de lectura de entradas y se mantiene funcionando esta lógica de manera general en todo momento.

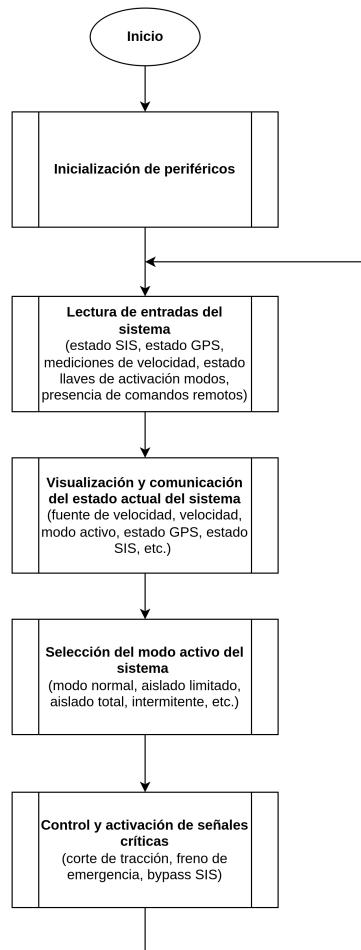


Figura 3.33: Lógica general del firmware del SAL/T

En la figura 3.34 se visualiza un diagrama con la lectura de los comandos más relevantes y llaves de activación local que participan de la selección del modo de operación y dentro de cada modo el estado general de las señales críticas.

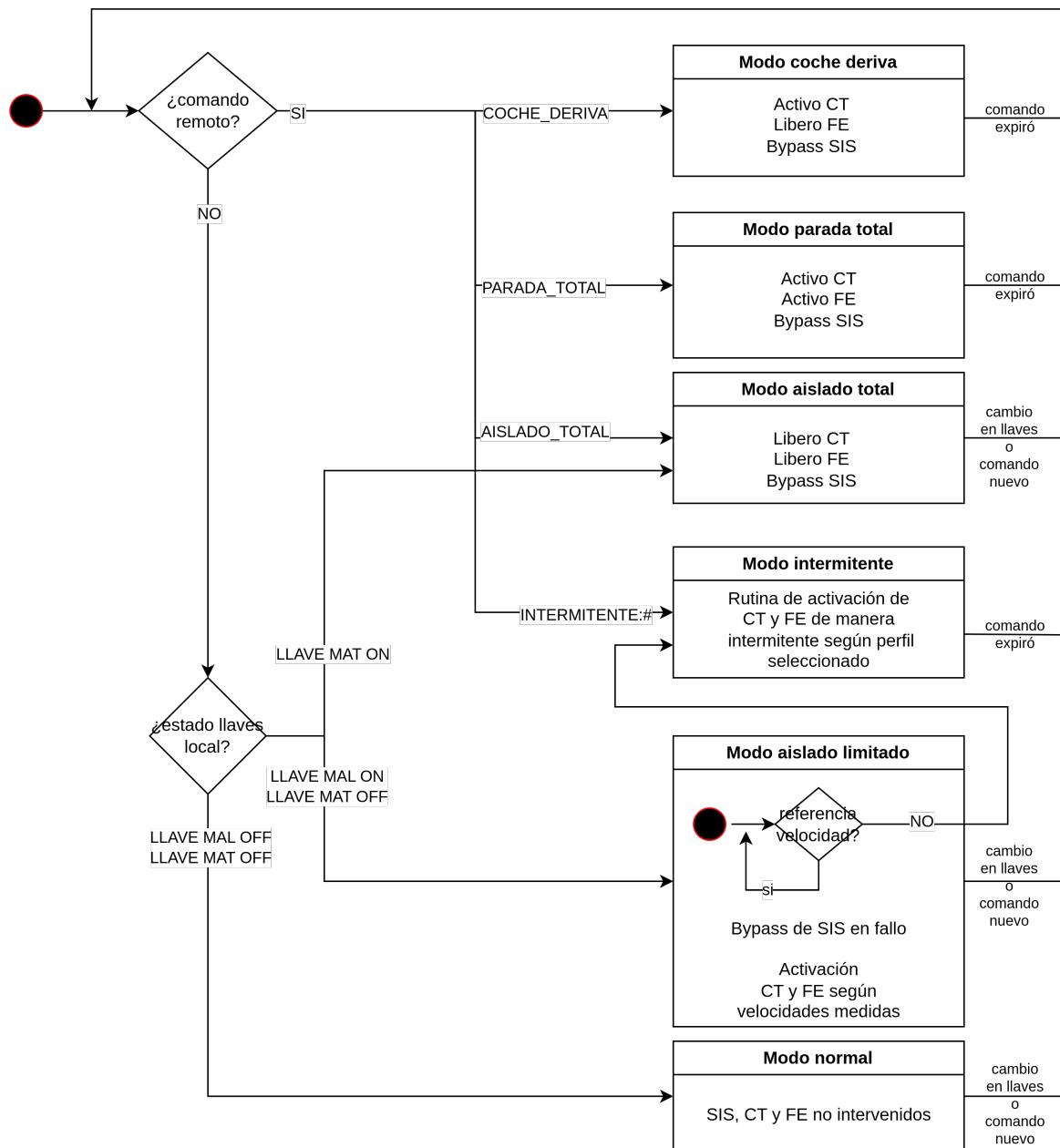


Figura 3.34: Diagrama de flujo de modos de operación del SAL/T

3.3.2 Mecanismos de RTOS

El firmware del SAL/T está implementado utilizando FreeRTOS [75] para permitir una ejecución pseudo-concurrente de las distintas tareas de manera eficiente, mejorando la respuesta del sistema ante eventos y mejorando el manejo de sus recursos. Se obtiene mayor control sobre la prioridad de las tareas y tiempos de respuesta, lo que lo hace ideal para aplicaciones de tiempo real críticas. Además, facilita la integración de sistemas complejos a través de las funcionalidades de asignación de tareas, comunicación entre tareas y manejo de memoria, siendo a la vez ligero y altamente personalizable para entornos de hardware específicos.

Se utilizó la interfaz de FreeRTOS-CMSIS versión 1.02, configurada con la unidad de punto flotante habilitada, permitiendo que las tareas de mayor prioridad interrumpan a las de menor prioridad (*preemption*), asignando 15360 bytes de memoria para el tamaño total del *heap*.

Las distintas *tasks* o tareas utilizadas en el sistema son las siguientes:

- **commsTask - prioridad alta:** esta tarea se encarga de procesar los mensajes recibidos por cualquiera de las comunicaciones con dispositivos externos. Su prioridad es alta porque estas comunicaciones pueden traer datos que modifiquen completamente el estado del sistema y deben ser considerados lo antes posible para corregir el estado del sistema.
- **systemTask - prioridad media:** esta tarea se encarga de leer de manera sistemática las entradas disponibles del sistema y luego de visualizarlas y registrarlas.
- **modeTransTask - prioridad media:** esta tarea va a utilizar los últimos datos de entrada recibidos para determinar el modo de operación en el que se debe ejecutar el sistema.
- **criticalSigTask - prioridad media:** esta tarea se encarga de, basado en el modo de operación activo y en las demás entradas del sistema, determinar la activación o liberación de las señales críticas del sistema.
- **reportTask - prioridad baja:** esta tarea reporta de manera periódica el estado del sistema de manera remota. Su prioridad es la más baja porque no modifica el comportamiento real del sistema y resulta no crítico el desfasaje en la transmisión.

Las tareas de prioridad media ejecutan su rutina, y al terminar se pone un delay del sistema operativo de 500 ms y se le devuelve el control al *scheduler* para que asigne los recursos de procesamiento a la tarea que siga en la cola de prioridades establecida. A ese delay se le suma una baja cantidad aleatoria de milisegundos para evitar la posible sincronización de las tareas. Por otro lado, la tarea de prioridad baja se ejecuta de manera periódica cada un tiempo configurable, por defecto 5 segundos. Las rutinas de todas las tareas se finalizan de manera rápida en pocos milisegundos, permitiendo que el asignador de tareas le dé el procesamiento a todas las tareas mientras se cumple el delay configurado para cada uno; de esta manera se evita lo que se conoce como *starving* (que las tareas de menor prioridad nunca se ejecuten).

En cuanto a la tarea de mayor prioridad, se utilizó otro mecanismo de sincronización de FreeRTOS para facilitar el procesamiento de las comunicaciones; se utilizó un semáforo binario. Este mecanismo de sincronización tiene dos estados, disponible (1) o no disponible (0). Se utiliza para gestionar el acceso a un recurso compartido o para sincronizar tareas e interrupciones. Una tarea da o libera el semáforo cuando un evento ocurre, y otra tarea lo toma cuando está disponible; permitiendo la ejecución controlada de procesos dependientes.

En el SAL/T, todas las comunicaciones con sistemas externos (módulo GPS, módulo Wi-Fi, fuentes de medición de velocidad, conexión serie local) están conectadas por interfaz UART al MCU y tienen habilitada la recepción por interrupción. Esta interrupción por hardware tiene mayor prioridad que las tareas del kernel lo que permite interrumpir cualquier tarea que se esté ejecutando. En la función de *callback*, se mantienen buffers de recepción separados (uno por interfaz) hasta detectar la recepción de una línea completa de un mensaje. En ese momento, se almacena la línea completa en una variable global, accesible desde la rutina de las comunicaciones, y se libera el semáforo de comunicaciones.

Al terminar la interrupción por hardware, el asignador de tareas reprioriza las tareas y en caso de haber liberado el semáforo, le pasa el control a la tarea de mayor prioridad. En esta tarea se verifica qué comunicación tiene un mensaje nuevo y allí se procesan los datos correspondientes. Al finalizar, se vuelve a tomar el semáforo bloqueando la tarea hasta la recepción de una nueva línea de información.

3.3.3 Medición del estado de los SIS

La medición del estado de los distintas señales de los SIS se realiza utilizando el conversor analógico digital (ADC) que trae el MCU; este permite la conversión de señales analógicas valores digitales que pueden ser procesados por el microcontrolador. En este caso, se utiliza el ADC3 del STM32, configurado para realizar conversiones en 10 canales diferentes.

La configuración del ADC se realizó para que opere con una resolución de 12 bits, lo que proporciona un rango de valores digitales de 0 a 4095, permitiendo una representación precisa de las señales analógicas; para lograr esta resolución, se consumen 15 ciclos del reloj de ADC. El prescaler del reloj se ajustó con un divisor de 8 lo que determina una operación a 10,5 MHz, y el modo de conversión en escaneo está habilitado para permitir la lectura de múltiples canales de manera secuencial.

El ADC está configurado para iniciar las conversiones mediante software, sin utilizar un disparador externo, lo que otorga mayor control sobre el inicio de las conversiones. Las conversiones no son continuas, y el DMA (Acceso Directo a Memoria) está deshabilitado de manera continua, permitiendo un mayor control del flujo de datos en el sistema. Además, los resultados se alinean a la derecha, lo que permite un procesamiento eficiente de los datos en 12 bits.

Se declara una función de callback HAL_ADC_ConvCpltCallback que se ejecuta al finalizar la 10ma conversión del ADC. En esta, se prende un flag de que hay una serie de valores medidos y se reinicia el proceso de adquisición de datos mediante DMA en los 10 canales configurados. El DMA permite que los datos del ADC se transfieran directamente a la memoria sin la intervención del procesador, lo que reduce la carga de la CPU y mejora el rendimiento. En modo no continuo, el ADC realiza conversiones de datos de manera intermitente o bajo demanda, activando el DMA solo cuando es necesario transferir una nueva muestra.

Cuando se corre la rutina de lectura de entradas del sistema, se verifica si hay una nueva serie de valores medidos. En ese caso, se compara para cada uno de los canales medidos (5 señales de CT y 5 de FE, una para cada SIS) con un valor umbral configurado dependiendo la tensión con la que trabaje el tren. Conociendo el circuito de hardware que reduce la tensión al rango de los 3,3 V y sabiendo cuál es la tensión esperada en una llave abierta, se puede determinar qué valor es esperable encontrar en el ADC para una llave abierta. Para una llave cerrada, la tensión en sus terminales es cero y el valor que arroja el ADC también es cercano a 0. Por lo tanto, utilizando un umbral cercano al valor medio entre 0 y el esperado cuando la llave esté abierta, se puede determinar con el mejor margen el estado real de las señales del SIS.

3.3.4 Interacción con datos GPS

El módulo GPS utilizado en el sistema se comunica mediante interfaz UART con el MCU del sistema. Se configuró la UART5 para esta comunicación, con un *baudrate* de 9600, una longitud de palabra de 8 bits, sin bit de paridad y un bit de stop conocido como configuración 8N1. No se emplea control de flujo por hardware y el sobremuestreo está configurado en 16, lo que mejora la precisión en la recepción de datos.

El módulo GPS transmite datos a una frecuencia de 1 Hz, es decir, una vez por segundo, enviando múltiples líneas de información conocidas como frases NMEA (*National Marine Electronics Association*). Estas sentencias incluyen detalles sobre la posición (latitud y longitud), velocidad, altitud, fecha, hora y estado de los satélites [76]. En este proyecto, se utiliza la línea de GPRMC (*Recommended minimum specific GPS/Transit data*); una de las más importantes en la transmisión de datos GPS. Contiene información esencial como la hora, fecha, posición (latitud y longitud), velocidad sobre el suelo y dirección, y es utilizada para obtener los datos

mínimos necesarios para la navegación.

La recepción de la información se realiza utilizando la comunicación por interrupción donde un *buffer* receptor permite identificar el comienzo de esta frase porque siempre comienza por los caracteres \$GPRMC y termina por el carácter de retorno de línea. Una vez leída una línea completa, se activa el semáforo de procesamiento de comunicaciones y una vez terminada la interrupción de hardware de la UART, el RTOS va a priorizar el procesamiento de esta línea. Se almacena una marca de tiempo de la recepción de este dato para más adelante poder verificar la validez o expiración de la información si no se recibieron nuevos datos. La línea trae definida el orden de los datos separados por una coma. De esta manera, se generó un *parser* que permite asignar a una estructura los distintos datos presentes en la frase GPRMC.

El dato más relevante a consultar es el estado de la transmisión. Si este dato indica que la conexión está ausente (probablemente por falta de señal de la antena con los satélites), todos los otros datos deben ser descartados. En caso de leer un estado de conexión exitoso, se determinan los datos de latitud, longitud y velocidad transmitidos por el módulo.

3.3.5 Procesamiento de velocidades por RS-485

Para la comunicación con las fuentes externas de medición de velocidad, se utilizaron las interfaces UART7 y UART8 del MCU que, a través de un circuito conversor de UART a RS-485, logra comunicarse con los sistemas externos. La configuración de ambos UART establece la comunicación serie con una velocidad de 115.200 baudios, utilizando 8 bits de datos, 1 bit de parada y sin paridad. El modo está configurado sin control de flujo por hardware, y con sobremuestreo a 16, lo que asegura una comunicación eficiente y precisa.

La interfaz RS-485 1, conectada con la señal interceptada entre el Hasler Teloc 1500 y el indicador de velocidad de la formación, recibe el dato de la velocidad encapsulado dentro de un paquete de 31 bytes. Este paquete comienza y termina siempre por el byte 0x7E lo que permite delimitar la trama. Esta interfaz trabaja con la recepción por interrupción y al detectar el byte de inicio comienza almacenar la información recibida y al recibir el byte de stop, validando una longitud de 31 bytes totales, termina la línea de datos. En ese momento, se libera el semáforo para que el RTOS permita la ejecución de la rutina de procesamiento de esta línea donde se extrae de los bytes 7 y 8 la información de la velocidad. Nuevamente se incluye una marca de tiempo para verificar la validez del dato antes de ser utilizado en otras rutinas.

La interfaz RS-485 2, conectada con la medición de velocidad proveniente del generador de impulsos ópticos, recibe el dato de la velocidad de manera directa. Se espera la recepción de un mensaje numérico seguido de un carácter de retorno de línea. Esta interfaz, conectada al UART8, funciona utilizando la recepción por interrupción y mediante un *buffer* que almacena cada carácter hasta encontrar la finalización de la línea. Una vez completa la línea, se activa el semáforo para que al finalizar la rutina de interrupción, el RTOS pueda darle mayor prioridad al procesamiento de esa línea donde se va a intentar convertir los caracteres leídos a un valor numérico y almacenarlo como dato de velocidad. También se almacena una marca de tiempo para asociarla al dato de velocidad y permitir la expiración de este dato tras algunos segundos sin recepción de nuevos valores.

3.3.6 Comunicación con central operativa

La comunicación del SAL/T con la central operativa se va a realizar a través del módulo ESP32 que permite la conectividad Wi-Fi utilizando el protocolo MQTT para el intercambio de

mensajes.

El módulo ESP32 se conecta al MCU utilizando la interfaz UART4 con una configuración 8N1 con un *baudrate* de 115.200 y sobremuestreo de 16 veces. Esta comunicación va a ser de manera bidireccional, ya que ambos dispositivos van a transmitir y recibir datos. El ESP32 está configurado para hacer principalmente de puente entre el MCU y el broker MQTT; esto significa que cualquier mensaje que reciba del MCU, lo va a retransmitir al MQTT y cualquier mensaje que recibe de MQTT lo va a retransmitir al MCU. También tiene la funcionalidad de mantener la fecha y hora del SAL/T actualizado ante un eventual corte en la alimentación consultando un cliente NTP (Network Time Protocol) para obtener la hora actual de un servidor en Internet.

El ESP32 permite la configuración de los datos de hasta dos redes Wi-Fi y constantemente va a estar verificando la conexión y, en caso de desconexión, intentando conectarse de manera alternada a cualquiera de ellas. Una vez conectado a una red Wi-Fi, se va a registrar en el broker MQTT configurado donde va a suscribir al *topic* salt_remote_command. En este *topic*, la central operativa va a publicar los comandos a aplicar sobre el SAL/T por lo que cualquier mensaje recibido va a ser enviado al MCU para su procesamiento. Además, cada 10 segundos el módulo va a consultar el servidor NTP pool.ntp.org [77] para obtener la fecha y hora exacta y transmitírsela al SAL/T.

En sentido inverso, el ESP32 está constantemente levantando cualquier mensaje que pueda recibir por la interfaz UART. Al recibir un mensaje (terminado por un retorno de línea), el módulo va a leer los primeros caracteres (contando hasta un carácter delimitador) para determinar a qué *topic* MQTT corresponde enviar el mensaje siguiendo la tabla 5.

Comienzo mensaje	<i>Topic</i>
LOG	salt_remote_log
ACK	salt_remote_ack
STATUS	salt_status

Tabla 5: Identificación de *topic* MQTT

En la figura 3.35 se visualiza un diagrama de la comunicación entre el SAL/T y la central operativa con las distintas partes involucradas previamente explicadas.

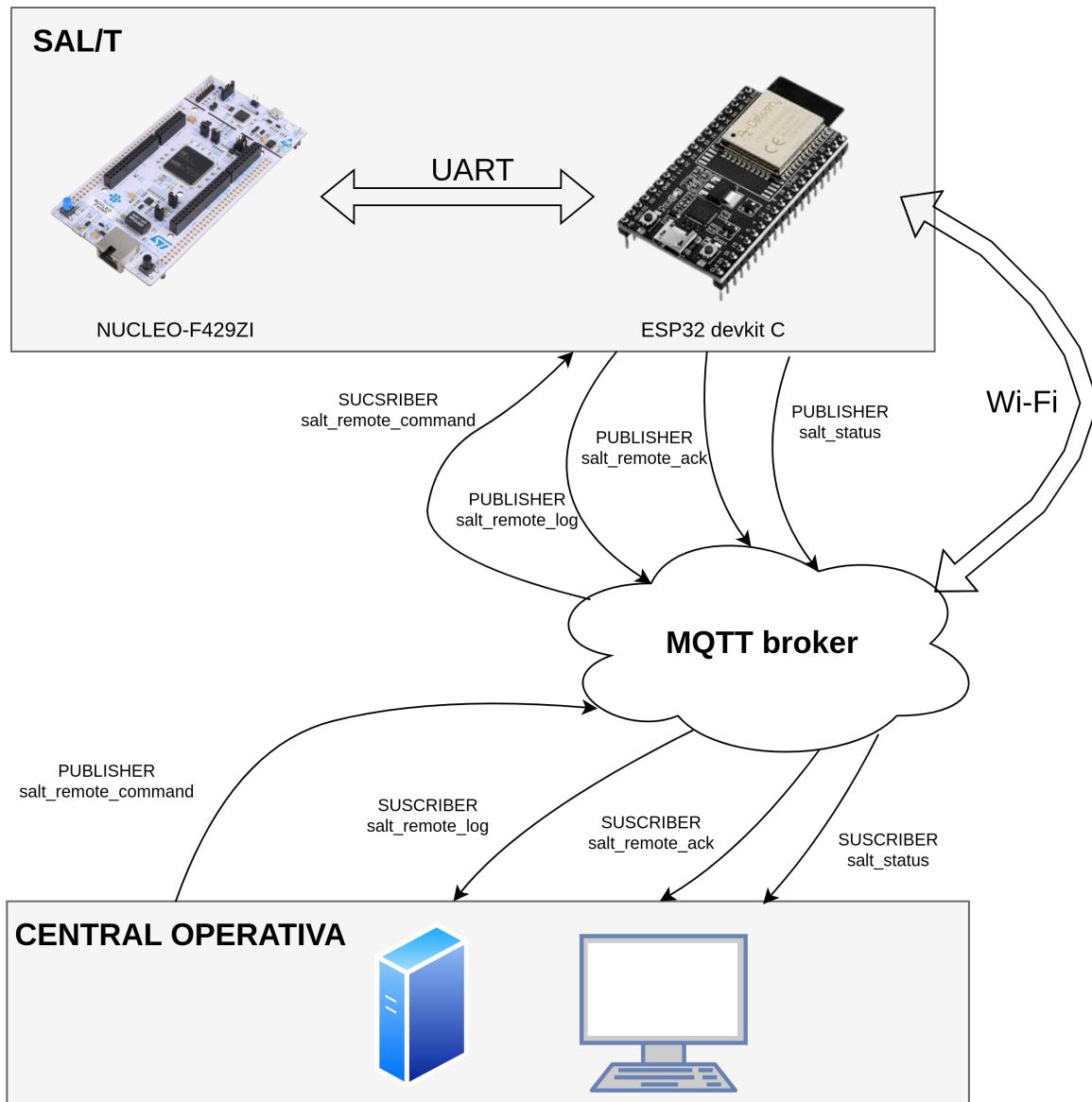


Figura 3.35: Diagrama de la comunicación entre el SAL/T y la central operativa

El *topic* `salt_remote_status` se utiliza para que el SAL/T reporte su estado de manera periódica independientemente de los cambios que puedan ocurrir en el sistema. Por defecto, se reporta el estado general del sistema cada 5 segundos, siendo este tiempo un parámetro configurable del sistema. En este mensaje, se reporta el modo que esté activo en el SAL/T. El *topic* `salt_remote_log` se utiliza para transmitir todos los cambios de estado del sistema y permitir guardar un registro de eventos en la central operativa en tiempo real.

Los *topics* `salt_remote_commands` y `salt_remote_ack` se utilizan para el envío de comandos y confirmación de recepción entre la central operativa y el SAL/T. Los comandos enviados tienen una estructura de ID|COMANDO:PARÁMETROS que permite luego al sistema separar los distintos datos. De recibir y procesar correctamente el comando, el SAL/T va a responder por el *topic* `salt_remote_ack` un mensaje con el formato ACK:ID utilizando el mismo ID recibido en el comando para confirmarle a la central operativa la recepción y ejecución del comando enviado. Los comandos que modifican el estado del SAL/T de manera continua tienen un tiempo de validez configurable, por defecto de 10 segundos, que una vez transcurrido son considerados inválidos.

A continuación se detallan los distintos comandos aceptados por el SAL/T y una breve explicación del comportamiento del sistema al recibirlos.

- **AISLADO_TOTAL:** este comando fuerza el estado del SAL/T al modo aislado total donde todos los SIS son aislados y se liberan las señales de CT y FE.
- **PARADA_TOTAL:** este comando fuerza la activación de las señales de CT y FE del SAL/T llevando la formación a un estado de detención forzosa.
- **COCHE_DERIVA:** este comando fuerza la activación de la señal de corte de tracción, pero libera la señal de freno de emergencia aislando cualquier SIS que pueda estar activando la señal.
- **INTERMITENTE:int:** este comando recibe un comando numérico del 1 al 5 que indica el perfil intermitente que se debe activar. El SAL/T pasa a estado intermitente siguiendo los tiempos de aceleración, desaceleración y freno configurados para el perfil seleccionado.
- **CANCEL:** este comando cancela cualquier otro comando que esté vigente en el SAL/T.
- **REPORT_STATUS_PERIOD_CONFIG:int:** el parámetro corresponde a la cantidad de segundos que se desea configurar como período de transmisión de reporte de estado del SAL/T. Por defecto son 5 segundos pero se acepta la modificación en un rango de 1 a 60s.
- **COMMAND_VALIDITY_CONFIG:int:** el parámetro corresponde a la cantidad de segundos que se desea configurar como tiempo de validez de un comando remoto activo. Por defecto son 10 segundos, pero se acepta la modificación en un rango de 1 a 60s.
- **SPEED_CONFIG:int,int,int,int:** se configura uno de los perfiles de velocidades que controlan la formación en el modo aislado limitado. El primer parámetro corresponde al perfil a configurar (valor entre 0 y 3 que corresponde a la zona de circulación 1, 2 y 3 y cuando no hay referencia de ubicación), luego los otros 4 parámetros corresponden a la velocidad límite para desacelerar, la velocidad límite para volver a acelerar en caso de activación del CT, la velocidad límite para aplicar el freno de emergencia y el tiempo mínimo de frenado una vez superado el umbral anterior.
- **INTERMITENTE_CONFIG:int,int,int,int,int:** se configura uno de los perfiles que controla la formación en modo intermitente. El primer parámetro corresponde al perfil a configurar (valor entre 1 y 5), luego los otros 4 parámetros corresponden el tiempo de aceleración por ciclo, el tiempo de desaceleración por ciclo, la cantidad de ciclos antes de aplicar el freno, y el tiempo de frenado luego de transcurridos los ciclos configurados.
- **DOWNLOAD_LOGS:** este comando permite extraer toda la información de registro de eventos que tiene almacenado el SAL/T internamente. Se transmiten todos los eventos registrados a través del canal de logeo con la marca de tiempo correspondiente y un mensaje de inicio y de final de reporte para entender el contexto en el que se transmiten registros de eventos viejos.
- **DATETIME:dd/MM/AAAA hh:mm:ss:** este comando no es enviado por la central operativa sino que es obtenido directamente por el ESP32 cada 10 segundos y enviado al MCU para que pueda sincronizar su fecha y hora con el informado por un servidor.

3.3.7 Lógica de control modo limitado e intermitente

El SAL/T cuenta con dos modos donde las señales críticas de corte de tracción y freno de emergencia son activadas y liberadas de manera dinámica; en el modo aislado limitado y en el modo intermitente. El modo aislado limitado se activa de manera local con la llave rotativa

ubicada en el panel frontal del SAL/T; en este modo se utilizan las mediciones de velocidad y ubicación para determinar la activación o liberación de las señales críticas. En caso de no contar con referencias de velocidad, se pasa automáticamente al modo intermitente (también accesible de manera directa mediante comando remoto desde la central operativa); en este modo, no se consideran la velocidad de circulación sino los tiempos configurados para la activación de las señales. Si al momento de activación del modo aislado limitado no se cuenta con referencia de velocidad, se activa el corte de tracción y freno de emergencia por 30 segundos.

En el modo aislado total existen cuatro perfiles de velocidades; el primero corresponde al perfil activado cuando no existe referencia de la ubicación de la formación y por lo tanto no se puede determinar la zona de circulación, los otros tres corresponden a las tres zonas de circulación definidas para el recorrido de la formación. Cada perfil cuenta con sus propias velocidades configuradas. Los parámetros configurados en cada perfil corresponden a la velocidad a la cual se debe activar el corte de tracción para desacelerar, la velocidad a la cual se puede liberar la señal de corte de tracción para volver a acelerar, la velocidad límite para activar el freno de emergencia (debe ser superior a la velocidad de desaceleración ya que se activan ambas señales críticas juntas para detener la formación) y el tiempo mínimo que ambas señales deben mantenerse activas tras superar el último umbral hasta llegar a una velocidad de detención.

El SAL/T, en modo aislado limitado, va a estar constantemente monitoreando la ubicación para seleccionar el perfil que corresponda y midiendo la velocidad de circulación para determinar si es necesario activar o liberar alguna de las señales mencionadas anteriormente. Además, en este modo se activa el buzzer sonoro de manera intermitente y al exceder alguno de los umbrales configurados, se activa el buzzer de manera continua para indicar el sobrepaso del umbral.

El modo intermitente tiene cinco perfiles disponibles en el que cada uno de ellos cuenta con los parámetros de tiempo de aceleración, tiempo de desaceleración, cantidad de ciclos antes de frenar y tiempo de frenado. El sistema va a determinar el perfil a ejecutar acorde a lo que seleccione el maquinista con un botón ubicado en el panel frontal y su correspondiente indicación luminosa que le permite visualizar el perfil seleccionado en caso de activarse de manera local, o el perfil seleccionado en el comando remoto si se activó el modo desde la central operativa. La idea de tener distintos perfiles disponibles y configurables corresponde con que los tiempos para cada formación pueden ser distintos dependiendo el material rodante sobre el que se aplique, condiciones externas o incluso la zona de circulación del momento.

El SAL/T va a ejecutar estar rutina de aceleración (de T segundos) y desaceleración (de D segundos) N veces para luego activar el freno junto al corte de tracción por E segundos; luego recomienza la rutina desde el comienzo. En la figura 3.36 se visualizan los ciclos del modo intermitente viendo como se espera que varíe la velocidad a lo largo del tiempo.

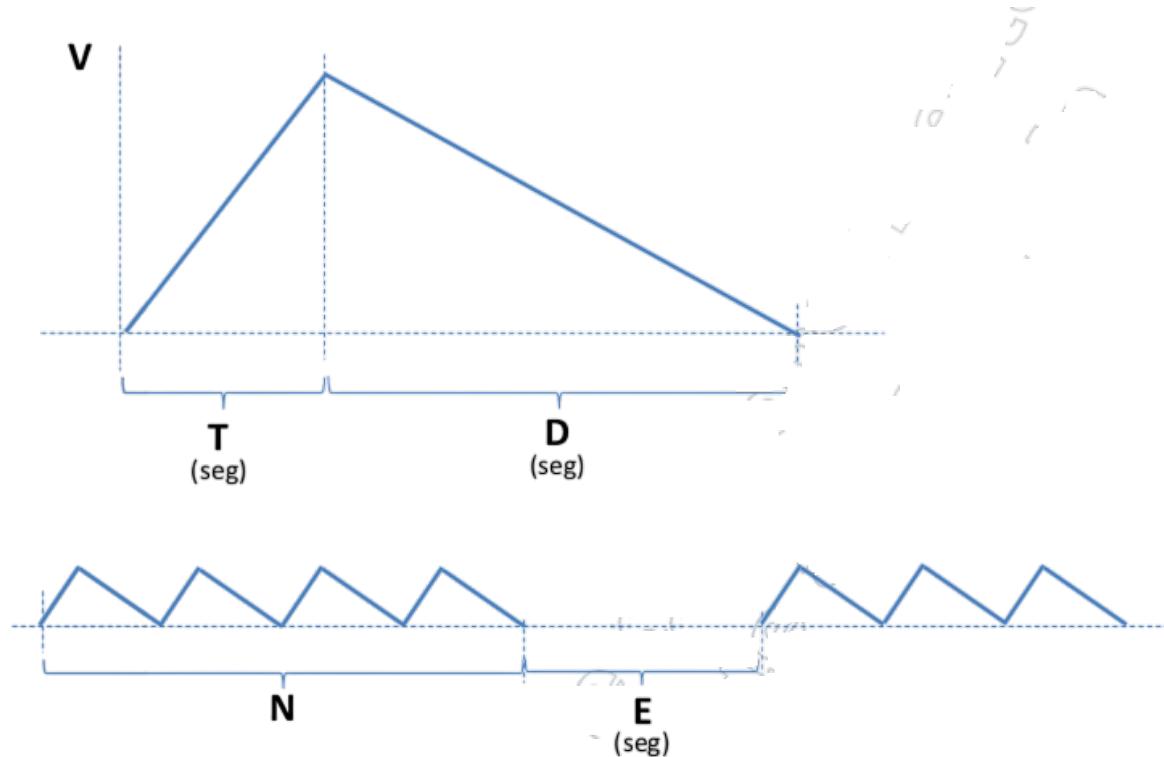


Figura 3.36: Perfil de velocidad de una formación en modo intermitente

3.3.8 Comunicación con controlador LED

Para la configuración y utilización del controlador LED AS1115-BSST se utilizaron los detalles de los registros y formatos de escritura de la hoja de datos del producto. La comunicación con el SAL/T se realiza por el protocolo I2C. Al iniciar el controlador, se configuran los registros de modo de operación para configurar el modo normal de operación, se configura el registro que selecciona cuantos de los 8 dígitos disponibles a controlar se van a utilizar con el valor máximo, ya que se utilizan los primeros 4 dígitos para el display de velocidad y los otros, aunque no de manera completa, para alimentar los demás LEDs del panel frontal. También se configura la intensidad en su valor medio para obtener el brillo esperado.

El controlador trae opciones de decodificación para dígitos 7 segmentos en formato Code-B o HEX, pero no existe la posibilidad de configurarlo solo para los primeros 4 dígitos y utilizar los otros segmentos de manera individual. Por esto, es que se desactivó la decodificación en todos los dígitos y se dejó del lado del MCU la transformación de los números a los segmentos de cada dígito correspondientes.

El SAL/T, en cada ciclo de operación, releva los valores correspondientes para el display de velocidad y para cada uno de los LEDs indicativos del panel frontal. Luego, transforma estos estados en los bits que corresponden en cada uno de los 8 registros utilizados del controlador LED y los envía por I2C para reflejar esos estados en las señales de activación de los LEDs en el panel frontal.

3.3.9 Registro de eventos

El SAL/T tiene la capacidad de registrar eventos tanto de manera local como de manera remota. Localmente, interactúa con la memoria SD conectada por interfaz SPI y ante cada evento registrado, se escribe dentro de un archivo de logeo donde se va acumulando el histórico de eventos. A nivel remoto, para cada evento, se utiliza la comunicación UART con el módulo ESP32 para que este luego retransmita el mensaje al broker MQTT en el *topic* de logs, y la central

operativa pueda recibir estos eventos en tiempo real y almacenarlos en una base de datos propia.

Todos los eventos registrados están acompañados de una marca de tiempo que lleva internamente el SAL/T. Cada 10 segundos, el ESP32 envía una actualización de la fecha y hora para que el SAL/T se pueda sincronizar en caso de reinicio o eventual desfasaje del reloj interno.

El SAL/T registra cualquier evento de cambio de estado que pueda detectar. Algunos de ellos son los siguientes:

- Detección de alimentación tras reinicio
- Cambio de estado en la medición de algún SIS
- Cambios de estado en la activación del aislamiento de algún SIS
- Cambio de estado de modo del sistema
- Cambio de zona de circulación
- Cambio de estado de la conexión GPS
- Cambio de fuente de medición de velocidad
- Cambio de velocidad en modo aislado limitado
- Aplicación de cambios de configuración tras recepción de comando remoto
- Aplicación de comando remoto

El SAL/T permite la descarga de todos los datos registrados en la memoria SD de manera local, conectándose por USB a una comunicación serie 8N1, *baudrate* 115.200 y enviando el comando DOWNLOAD_LOGS. Como respuesta, se envían secuencialmente todos los datos almacenados en la memoria. Esta opción también es posible de manera remota, enviando desde la central operativa el mismo comando y recibiendo la respuesta por el canal de logs del broker MQTT.

4 Ensayos y resultados

En este capítulo se presenta el banco de pruebas utilizado para realizar los ensayos del sistema y los resultados obtenidos de las pruebas tanto a nivel hardware y firmware por separado como de manera integral, permitiendo validar todas las funcionalidades del sistema.

4.1 Banco de pruebas

Para poder realizar pruebas del SAL/T fue necesario armar un banco de pruebas con componentes, módulos y sistemas externos que permitan simular las condiciones de operación real. En la figura 4.1 se visualiza el banco de pruebas completo que se utilizó para realizar las pruebas del dispositivo.

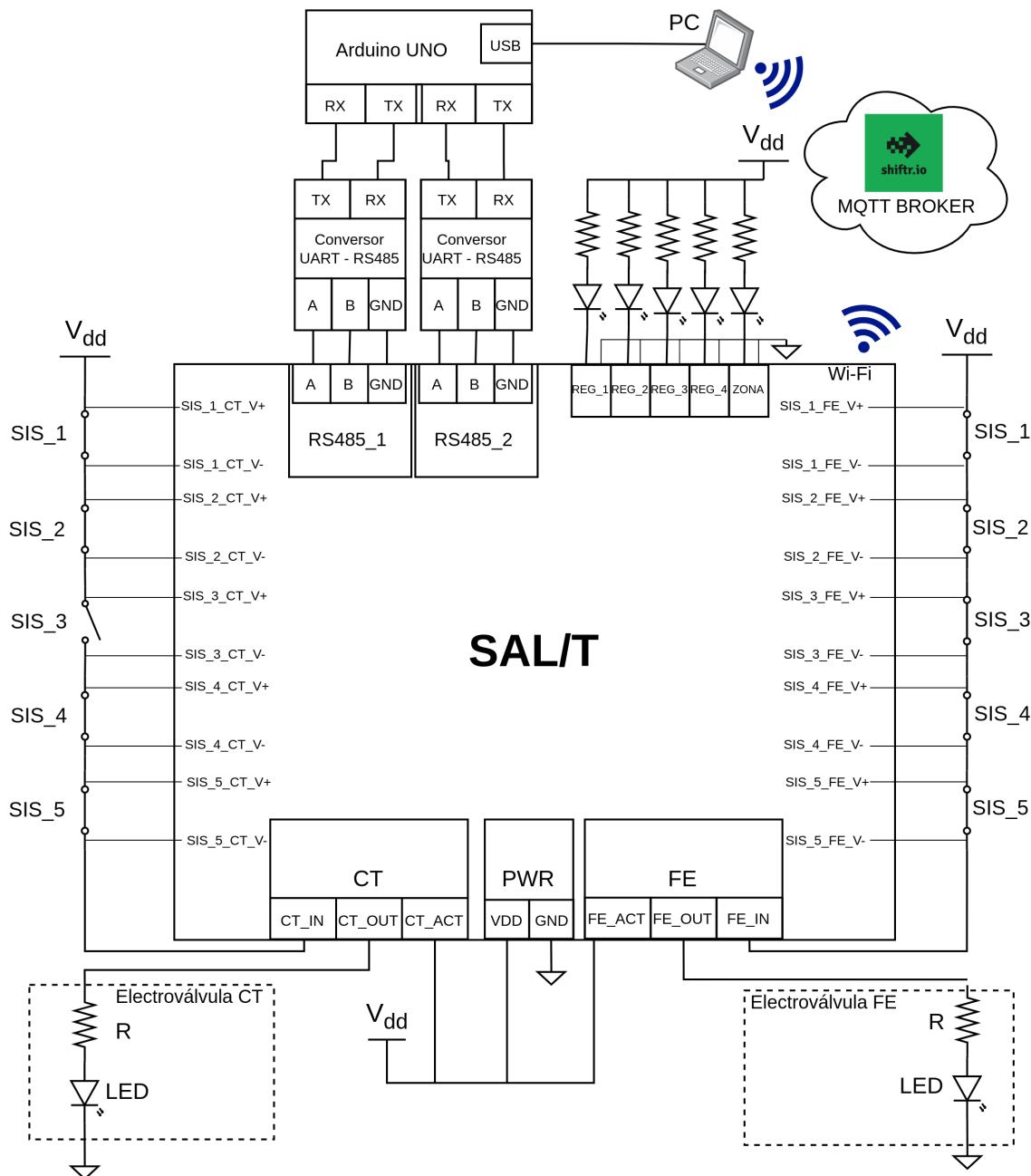


Figura 4.1: Banco de pruebas completo del SAL/T

Los buses de las señales de corte de tracción y freno de emergencia fueron representados cada uno con cinco llaves en serie donde cada una representa la llave de estado de cada SIS. El polo negativo de la llave del último SIS se conecta en la entrada del SAL/T CT_IN (o FE_IN para el otro bus). Luego, la salida CT_OUT (o FE_OUT) se conecta con la electroválvula correspondiente representada por una resistencia y un LED para visualizar su estado de activación. También se conectó la misma señal de activación que alimenta el bus a la entrada CT_ACT (o FE_ACT); en este banco de pruebas, se utilizó una tensión de 5 V para alimentar los buses porque facilitó la obtención de la fuente frente a una de 72 VDC o 110 VDC. Sin embargo, mediante el firmware se va a poder compensar esa diferencia en la tensión esperada luego del circuito de medición de los SIS modificando el umbral de los valores esperados en el ADC para determinar si una llave está abierta o cerrada. Por lo tanto, con este esquema se va a poder representar de una manera bastante similar al de operación real, el bus en serie de los SIS que termina en una electroválvula y la intención del SAL/T de intervenir y forzar estados luego de la posible apertura de la línea por alguno de los SIS.

Respecto a las salidas de registro de estado del SAL/T y al conector de zona, que permiten la conexión de baja o alta impedancia entre sus dos terminales dependiendo el estado a reportar, se conectó una resistencia y un LED en un terminal, y conexión a tierra en el otro. Esto se replicó en cada uno de los conectores para poder visualizar el estado reportado a la formación para cada una de las salidas.

Para simular las comunicaciones con los sistemas de medición de velocidad, se utilizó un Arduino Uno [79] y dos módulos conversores UART - RS-485 [80]. El Arduino Uno es una placa de desarrollo basada en el microcontrolador ATmega328P [81]. Cuenta con 14 pines digitales, 6 entradas analógicas, un único puerto UART para comunicación serie, y funciona a 5 V con una frecuencia de reloj de 16 MHz. Incluye una interfaz USB para la programación y alimentación, junto con un puerto de alimentación externa. Si bien el Arduino Uno cuenta solo con una interfaz UART, se puede simular una segunda utilizando la librería SoftwareSerial [82] que obtiene resultados suficientes cuando no es necesario realizar una comunicación bidireccional. En la figura 4.2 se visualiza la placa Arduino Uno utilizada.

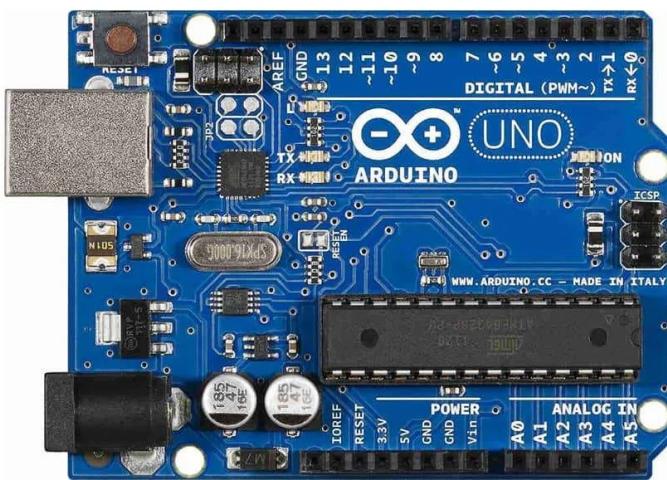


Figura 4.2: Placa de desarrollo Arduino Uno

El módulo conversor UART - RS-485 utilizado se puede visualizar en la figura 4.3; es un módulo fabricado por Hobbytrónica [83] basado en el chip MAX485 [84]; un transceptor de

bajo consumo que permite la comunicación serial diferencial a través del protocolo RS-485. Este módulo tiene entradas para el lado de la comunicación UART (TX, RX) y un selector para habilitar la transmisión o la recepción, ya que al pasar a RS-485, se convierte en una comunicación half-duplex. Por el lado del RS-485, tiene las salidas de A y B esperadas.

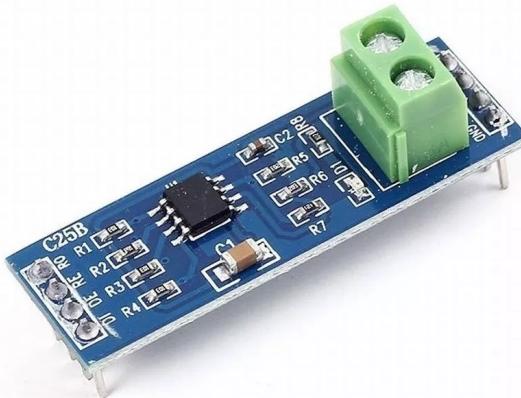


Figura 4.3: Módulo conversor UART a RS-485

El Arduino va a simular en una de sus interfaces la señal proveniente del Hasler Teloc 1500 generada tramas de 31 bytes iniciadas y terminadas por un byte de start/stop 0x7E y con la información de velocidad en los bytes 7 y 8. En la otra de sus interfaces va a comunicar el valor numérico de la velocidad que transmitiría el circuito de adaptación e interpretación del generador de impulsos ópticos. Ambas interfaces van a emitir comunicaciones continuas cada 1 segundo con el último valor configurado.

Por otro lado, el Arduino va a estar conectado a una PC por el conector USB y, mediante comunicación serie, va a estar recibiendo los datos de velocidad a transmitir en cada una de sus interfaces. De esta manera, se puede ir variando las velocidades reportadas al SAL/T en cada una de sus interfaces en todo momento desde una PC.

La PC también va a simular las interacciones de la central operativa conectada al broker MQTT generado con la plataforma shiftr.io [78]; una plataforma en línea que permite visualizar y gestionar flujos de datos en tiempo real mediante el protocolo MQTT. Proporciona un panel interactivo donde se pueden ver en vivo los mensajes publicados y suscritos entre dispositivos, facilitando el monitoreo y depuración de redes IoT. Es fácil de usar y ofrece herramientas para desarrollar, probar y supervisar la comunicación entre dispositivos conectados. Por lo tanto, la PC va a poder enviar comandos remotos al SAL/T a través del broker MQTT y al mismo tiempo va a recibir los mensajes de confirmación de recepción (ACK), los logs del sistema y el reporte periódico del estado del sistema. El SAL/T está configurado para interactuar con el mismo broker MQTT conectándose al servidor a través de la red Wi-Fi que le permite conectarse al módulo interno ESP32. En la figura 4.4 se visualiza el broker shiftr.io con el SAL/T y la central operativa conectados y suscriptos a los distintos *topics* MQTT intercambiando mensajes.

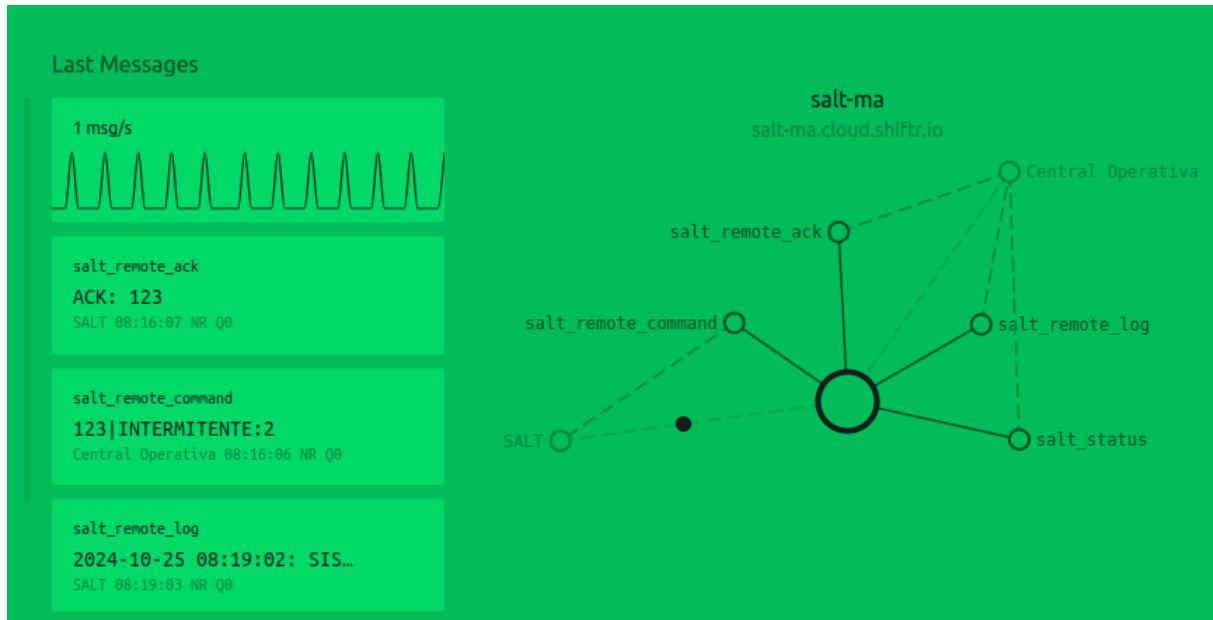


Figura 4.4: Visualización de shiftr.io conectando el SAL/T con la central operativa

4.2 Pruebas de hardware

Los PCB del SAL/T fueron diseñados para poder ser armados de manera progresiva conectando de a uno los circuitos que interactúan con el MCU. Para eso, se colocó el *footprint* para una resistencia de $0\ \Omega$ o un puente en la alimentación de cada circuito individual. También se colocaron puntos de prueba en casi todas las señales. Esto permitió armar el sistema de manera incremental y asegurándose que cada circuito esté en funcionamiento (a nivel hardware) antes de conectar el siguiente.

La primera prueba que se realizó fue conectar la alimentación de uno de los controladores ULN2003 y ejecutar una rutina de activación y desactivación de cada uno de los pines de control utilizados para los relés. Al mismo tiempo, se leía el estado de la salida secundaria del relé para verificar que se produjera el cambio de estado esperado. También se colocó un LED indicativo al lado de cada relé para poder visualizar dentro de la placa el estado de cada uno en estas pruebas. Esta prueba exitosa se repitió para todos los 21 relés del sistema, validando que sus circuitos de control, de activación y de salida estuvieran bien conectados y funcionando según lo esperado.

En segundo lugar, se probaron los circuitos de medición de los SIS. Se alimentó el chip ICL7660AIBAZA y se midió en el punto de prueba de su tensión de salida la tensión esperada de -5 V necesaria para alimentar los amplificadores de instrumentación INA823DGKR. Para probar los circuitos de medición, se utilizaron los buses de las señales CT y FE simulados con llaves. Con una rutina del ADC que leía continuamente los valores en las entradas y la función de *debugging* que permite hacer una ejecución controlada y visualizar los valores de los registros y utilizando los puntos de prueba sobre la placa, se verificó que los circuitos estén convirtiendo la tensión diferencial entre los terminales de la llave en la tensión *single-ended* esperada. Se probó alimentar los buses con 5 V y con 3,3 V para verificar que se obtuvieran distintos valores esperados a la salida del circuito y a nivel digital luego del ADC. No se pudo probar los circuitos con alimentación de 72 o 110 VDC pero se validó la linealidad del circuito que permitiría funcionar correctamente para las tensiones diseñadas.

Luego se probaron los circuitos de control simples con el reset del módulo GPS o la activación del buzzer sonoro en la placa secundaria. En ambos casos, se verificó que el circuito funcionara según lo esperado, pudiendo prender y apagar el módulo GPS o el buzzer sonoro con el valor del

pin digital al que se conecta el MCU. También se probó la lectura de los circuitos de las llaves de activación de modo aislado limitado, modo aislado total y el botón seleccionador de perfil intermitente, obteniendo la tensión esperada en los puntos de prueba y el valor digital en el MCU.

En cuanto a las interfaces de comunicación entre los distintos módulos o chips del sistema, se hicieron las mediciones de los puntos de prueba básicos con un voltímetro para verificar que las tensiones de reposo de las líneas de comunicación fueran las esperadas. Para los circuitos conversor UART a RS-485, se midieron los valores de tensión esperados para las señales A o B utilizando los pines del MCU como salidas GPIO digitales para poder fijar su valor antes de configurarlo para transmisiones UART. Para los LEDs y display de la placa secundaria, se configuró una rutina que encendía y apagaba progresivamente todos los LEDs disponibles y se verificó visualmente que la conexión sea correcta para cada uno.

4.3 Pruebas de firmware

A nivel firmware, también se realizaron pruebas muy modularizadas para asegurarse el correcto funcionamiento de las interfaces con cada uno de los circuitos o módulos del sistema.

La comunicación con el módulo GPS se probó inicialmente por fuera del PCB, ya que los módulos (GPS y Nucleo) estaban disponibles antes de la fabricación de la placa. Para esto, se copiaron las conexiones esperadas utilizando cables externos para conectar las interfaces UART de la Nucleo con el módulo GPS. El módulo GPS viene con una configuración por defecto que transmite a un *baudrate* de 9600 una serie de mensajes una vez por segundo. Para verificar la comunicación entre los módulos, del lado del MCU se configuró la recepción por interrupción y se reenviaron todos los datos recibidos a la interfaz de debugging para poder ser inspeccionados. Se observó que efectivamente una vez por segundo se recibían todas las frases NMEA que indicaba la hoja de datos. A partir de esa estructura de datos identificada, se armó un *parser* para poder extraer de la frase GPRMC los datos de estado de la comunicación GPS, la ubicación (latitud, longitud) y la velocidad informada. Estos datos también fueron enviados por la interfaz de debugging para luego verificar si estaban correctos. El módulo GPS puede perder la conexión con los satélites por lo que deja de informar todos los datos así que se probó el estado de conexión correcto y la pérdida de conexión desconectando la antena del sistema.

Con el módulo ESP32 también se realizaron las primeras pruebas de comunicación de manera externa. El ESP32 cuenta con una interfaz serie accesible mediante USB, por lo que se configuró para retransmitir todo lo recibido por el MCU a través de esta interfaz y también retransmitir lo recibido por USB al MCU. La misma configuración se realizó en el MCU. Por lo tanto, al enviar un mensaje desde la PC al MCU, se recibía por la comunicación con el ESP32 el mismo mensaje, y al enviar desde la PC un mensaje al ESP32, se recibía el mismo mensaje por la interfaz con el MCU. De esta manera, se verificó que la comunicación entre los módulos era correcta y se estaban procesando correctamente los mensajes transmitidos y recibidos. Luego, se configuró el ESP32 para conectarse a una red Wi-Fi y al servidor MQTT, por lo que se modificó el firmware para retransmitir ya no a través de la interfaz USB, sino a través del servidor MQTT. Desde la PC, se configuró el broker MQTT en shifr.io y se conectó como cliente utilizando la librería Paho [85] de Python para interactuar con el servidor. Con esta extensión, se pudo repetir la prueba de comunicación entre el MCU principal, el ESP32 y el servidor MQTT, ya que los mensajes llegaban correctamente de punta a punta en ambos sentidos. Las conexiones utilizadas para este último escenario se pueden visualizar en la figura 4.5

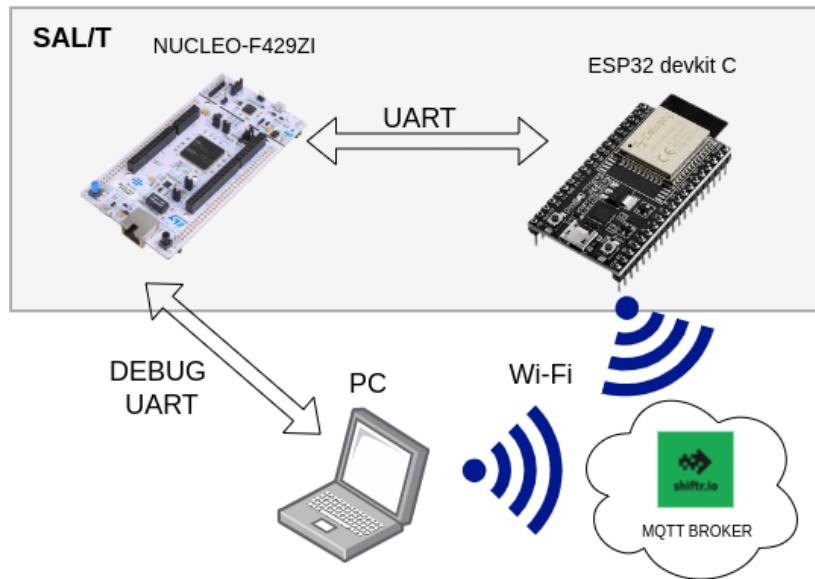


Figura 4.5: Conexiones para prueba de comunicación con ESP32 y servidor MQTT

Para el código del ESP32, también se realizó la prueba de configurarle la conexión a dos redes Wi-Fi y cuando no se pueda conectar a una, intentar la conexión con la otra de manera sistemática. Se levantaron dos redes Wi-Fi y se probó que al desconectar una, se conecta con la otra (envía un log al servidor MQTT cuando se conecta a una nueva red) y al desconectarse ambas se queda monitoreando qué red se levanta primero para establecer una conexión lo antes posible.

Para la comunicación con la memoria SD, también se logró realizar algunas pruebas anticipadas a tener la placa lista. Para esto, se utilizó un módulo para memorias microSD [86] reutilizado de otro proyecto que se puede observar en la figura 4.6. Este módulo permite conectar solo las señales de SPI con cables externos a la placa Núcleo. Se realizó esta conexión y se probaron algunas funciones básicas de la comunicación como listar los archivos disponibles dentro de la unidad, escribir en un archivo y leer el contenido de un archivo. Con estas pruebas básicas y el desarrollo de las funciones para interactuar, se verificó el funcionamiento en la comunicación por SPI con la memoria SD. Una vez recibidas las placas, se realizaron las mismas pruebas, pero utilizando el zócalo para la memoria microSD soldado a la placa principal junto a las resistencias diseñadas para las líneas de comunicación.

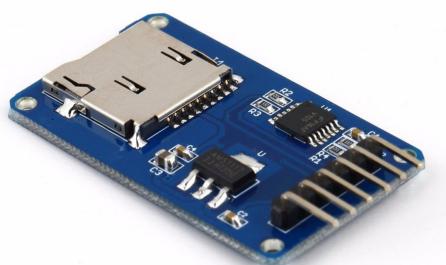


Figura 4.6: Módulo SD para pruebas de comunicación

Para la comunicación con las fuentes de velocidades externas, se utilizó un Arduino Uno y módulos conversor UART a RS-485 para poder conectar con el SAL/T. En estas comunicaciones, se configuró el Arduino para transmitir de manera continua distintas velocidades en el formato esperado de cada uno de los sistemas que simula. Para el generado de impulsos ópticos, se envía el dato numérico de la velocidad directo, y para la señal proveniente del Hasler Teloc 1500 se envía una trama de 31 bytes con el dato de velocidad en los bytes 7 y 8. Del lado del SAL/T, se conectan las interfaces RS-485 con el pin de selección de dirección de la comunicación en modo recepción y se procesan los mensajes recibidos y se envían por interfaz de debugging para visualizarlos desde una PC. En ambas comunicaciones se logró obtener el dato de la velocidad transmitido y asociado a la interfaz del cual proviene cada dato.

Por último, se realizaron varias pruebas de comunicación y configuración con el controlador LED. Una vez con las placas armadas, ya que el empaquetado del chip no permitía una conexión con cables externos, se realizaron algunas pruebas utilizando el protocolo I2C. Utilizando la hoja de datos como referencia, se escribieron algunos registros para validar la comunicación y el funcionamiento deseado del controlador LED y de los LEDs conectados. Primero se habilitaron todos los dígitos que puede controlar el chip y se fueron encendiendo uno a uno los segmentos y LEDs conectados en un orden específico. Esto permitió verificar el mapeo entre las variables que representan un estado y la posición dentro de un registro del controlador. Una vez validado el mapeo, se armaron algunas funciones que permiten escribir los registros del controlador LED solamente utilizando el estado de los LEDs individuales y el número a mostrar para los dígitos 7 segmentos. De esta manera, se hicieron varias pruebas de escribir todas las cifras en el display y de encender y apagar todos los estados visualizables para verlo reflejado en los LEDs de la placa secundaria. También se probó la escritura de registros de configuración como el ajuste del brillo o la deshabilitación de alguno de los dígitos controlados o el reset por software que permite el controlador.

4.4 Pruebas integrales

Para realizar las pruebas integrales del sistema, se montaron todas las placas, conexiones internas, memorias y se utilizó el banco de pruebas completo de la figura 4.1. Se cargó en el MCU principal el firmware completo del sistema utilizando todos los mecanismos de RTOS, sincronización, lectura de entradas, activación de salidas y comunicación descriptas previamente. Dentro del módulo ESP32 también se cargó la versión final del firmware que permite la conexión a dos redes Wi-Fi de manera alternada ante una caída de la red actual, conectarse al servidor MQTT suscribiendo y publicando en sus canales e interactuando con el MCU principal por su interfaz serie; también envía un comando de actualización de la fecha y hora de manera periódica después de consultarla con un servidor NTP.

Para estructurar las pruebas integrales del sistema se siguieron punto a punto los requerimientos establecidos para armar un plan de pruebas que permita validar lo de manera completa. Si bien hay requerimientos que pueden estar cubiertos en más de una prueba, se aseguró que todos los requerimientos están cubiertos en al menos una prueba y, por lo tanto, el éxito de todas las pruebas asegura el cumplimiento de todos los requerimientos.

Prueba 1: Lectura de estado de los SIS

1. Se enciende el SAL/T en estado normal.
2. Se verifica que se haya activado el relé de registro correspondiente a la correcta alimentación del SAL/T.
3. Se abre una llave de una señal CT o FE de un SIS.

4. Se verifica que se encienda el LED rojo de falla del SIS correspondiente.
5. Se cierra la llave abierta.
6. Se verifica que se restaure el LED verde de estado normal del SIS correspondiente.
7. Se repite la prueba hasta agotar las llaves.
8. Se desconecta la alimentación del SAL/T.
9. Se verifica que se haya desactivado el relé de registro de alimentación del SAL/T.

Prueba 2: Aislamiento SIS en modo aislado limitado

1. Se enciende el SAL/T en estado normal.
2. Se transmite una velocidad continua de 10 km/h por debajo de cualquier umbral configurado.
3. Se abre una llave de una señal X (CT o FE) de un SIS N (1 al 5).
4. Se verifica que se encienda el LED de falla del SIS correspondiente.
5. Se verifica que el LED que representa la electroválvula de la señal X se haya apagado por la apertura de la llave.
6. Se activa la llave local de modo aislado limitado.
7. Se verifica que se haya encendido el LED del modo aislado limitado en el panel frontal.
8. Se verifica que se haya activado el relé de registro correspondiente al evento modo aislado limitado.
9. Se verifica que se haya activado el relé de aislamiento del SIS N para la señal X.
10. Se verifica que el LED que representa la electroválvula de la señal X se haya encendido por la continuidad de la línea.
11. Se desactiva la llave de modo aislado limitado retornando al estado normal.
12. Se verifica que se haya desactivado el relé de registro correspondiente al evento modo aislado limitado.
13. Se verifica que el aislamiento del SIS no exista más.
14. Se cierra la llave del SIS abierta.
15. Se repite la prueba hasta cubrir todas las llaves de los SIS.

Prueba 3: Activación modo total

1. Se enciende el SAL/T en estado normal.
2. Se transmite una velocidad continua de 10 km/h por debajo de cualquier umbral configurado.
3. Se abre una llave de una señal X (CT o FE) de un SIS N (1 al 5).
4. Se verifica que se encienda el LED de falla del SIS correspondiente.
5. Se verifica que el LED que representa la electroválvula de la señal X se haya apagado por la apertura de la llave.

6. Se activa la llave local de modo aislado total.
7. Se verifica que los circuitos de CT y FE haya activado el bypass de las señales conectando las señales ACT con OUT de cada una.
8. Se verifica que el LED que representa la electroválvula de la señal de CT y de FE se haya encendido por la continuidad forzada de las líneas.
9. Se transmite una velocidad continua de 50 km/h por encima de los umbrales configurados por otros modos.
10. Se verifica que el SAL/T no contempla la velocidad y mantiene alimentadas las señales de CT y FE.
11. Se desactiva la llave de modo aislado total.
12. Se verifica la liberación de las señales de CT y FE.
13. Se verifica que la línea de la señal X está interrumpida porque todavía existe una falla en el SIS N.

Prueba 4: Priorización de fuentes de medición de velocidad

1. Se enciende el SAL/T en estado normal.
2. Se transmite una velocidad continua de 10 km/h por debajo de cualquier umbral configurado en ambas interfaces RS-485.
3. Se verifica que el display del panel frontal indique la misma velocidad.
4. Se transmite una velocidad de 20 km/h reportada por la interfaz RS-485_2 (menor prioridad).
5. Se verifica que el display del panel frontal mantiene la visualización de la velocidad máxima prioridad 10 km/h.
6. Se varía la velocidad reportada por la interfaz RS-485_1 (mayor prioridad) a 25 km/h.
7. Se verifica que el display del panel frontal indique la velocidad de mayor prioridad 25 km/h.
8. Se desconecta la fuente comunicación de RS-485_1 de mayor prioridad.
9. Se verifica que el display del panel frontal indique la velocidad de mayor prioridad disponible 20 km/h.
10. Se desconecta la fuente comunicación de RS-485_2 .
11. Se verifica que el display del panel frontal indique la velocidad obtenida por el módulo GPS (cercano a 0 km/h estando quieto).
12. Se desconecta la antena de la comunicación GPS.
13. Se verifica que el display marque la ausencia de fuente de medición de velocidad con guiones en el display.

Prueba 5: Control de señales críticas ante variación de velocidad

1. Se enciende el SAL/T en estado normal.
2. Se reconoce la ubicación de la prueba dentro de la zona 1 de circulación según la configuración de los parámetros del GPS.
3. Se transmite una velocidad continua de 10 km/h por debajo de cualquier umbral configurado en ambas interfaces RS-485.
4. Se abre una llave de una señal X (CT o FE) de un SIS N (1 al 5).
5. Se activa la llave local de modo aislado limitado.
6. Se verifica que el LED que representa la electroválvula de la señal X se haya encendido por la continuidad de la línea producida por el aislamiento del SIS en fallo.
7. Se verifica que el buzzer sonoro emita sonido de manera intermitente.
8. Se varía la velocidad reportada por la interfaz RS-485_1 (mayor prioridad) de manera progresiva hasta alcanzar los 32 km/h.
9. Se verifica que el display muestra de manera continua la evolución de la velocidad.
10. Se verifica que al cruzar los 30 km/h (umbral definido para la zona actual como velocidad límite para desaceleración) se interrumpe la señal de CT.
11. Se verifica que el LED que representa la electroválvula de la señal CT se haya apagado por la interrupción de la línea.
12. Se verifica que el LED indicativo de corte en la señal de CT se haya encendido.
13. Se verifica que se haya activado el relé de registro correspondiente a la interrupción de la señal de CT.
14. Se verifica que el buzzer sonoro emita sonido de manera continua.
15. Se disminuye progresivamente la velocidad hasta llegar a los 20 km/h.
16. Se verifica que al cruzar los 24 km/h (umbral definido para la zona actual como velocidad límite para retomar la aceleración) se libera la señal de CT.
17. Se verifica que el LED que representa la electroválvula de la señal CT se haya encendido por la restauración de la continuidad de la línea.
18. Se verifica que el LED indicativo de corte en la señal de CT se haya apagado.
19. Se verifica que se haya desactivado el relé de registro correspondiente a la interrupción de la señal de CT.
20. Se verifica que el buzzer sonoro emita sonido de manera intermitente.
21. Se varía la velocidad reportada por la interfaz RS-485_1 (mayor prioridad) de manera progresiva hasta alcanzar los 40 km/h.
22. Se verifica que al cruzar los 30 km/h se interrumpe la señal de CT.
23. Se verifica que al cruzar los 36 km/h (umbral definido para la zona actual como velocidad límite para frenar) se interrumpe la señal de FE.
24. Se verifica que el LED que representa la electroválvula de la señal FE se haya apagado por la interrupción de la línea.

25. Se verifica que el LED indicativo de corte en la señal de FE se haya encendido.
26. Se verifica que se haya activado el relé de registro correspondiente a la interrupción de la señal de FE.
27. Se disminuye progresivamente la velocidad hasta llegar a los 0 km/h.
28. Se verifica que el estado de las señales de CT y FE permanezcan en estado de freno hasta alcanzar los 0 km/h y haber pasado al menos 30 segundos (tiempo mínimo configurado para activar el freno).
29. Se verifica que transcurridos los 30 segundos y con una velocidad nula, se liberan las señales de CT y FE.
30. Se verifica que se apaguen los LEDs indicativos de las señales de CT y FE.
31. Se verifica que se desactiven los relés de registro de CT y FE.

Prueba 6: Modo intermitente

1. Se enciende el SAL/T en estado normal.
2. Se desconectan o dejan de transmitir todas las mediciones de velocidad.
3. Se abre una llave de una señal X (CT o FE) de un SIS N (1 al 5).
4. Se activa la llave local de modo aislado limitado.
5. Se verifica que se haya aplicado el aislamiento de la llave del SIS N en la línea X.
6. Se verifica que se hayan activado las señales de CT y FE durante 30 segundos por la falta de referencia de velocidad.
7. Se verifica que se ejecute la rutina de modo intermitente para el perfil 1 seleccionado ($T=15$, $D=30$, $N=4$, $E=40$) durante algunos ciclos.
8. Se modifica el perfil del modo intermitente con el botón de cambio de perfil al perfil 3.
9. Se verifica que el LED indicador del perfil seleccionado se desplace cada vez que el botón se presione hasta llegar al perfil 3.
10. Se verifica que se ejecute la rutina de modo intermitente para el perfil 3 seleccionado ($T=30$, $D=60$, $N=3$, $E=60$) durante algunos ciclos.

Prueba 7: Comandos remotos para cambio de modo

1. Para cada comando que se envía dese la central operativa, se antepone un ID generado y se verifica la recepción de una confirmación del mensaje (ACK) para el mismo ID.
2. Se envía desde la central operativa un comando PARADA_TOTAL de manera periódica cada 1 segundo durante 60 segundos.
3. Se verifica que se haya encendido el LED de presencia de comando remoto en el panel frontal.
4. Se verifica que se hayan interrumpido las señales de CT y FE.
5. Se verifica que después de 10 segundos después de finalizada la última transmisión, el sistema vuelve al estado normal y apaga el LED de presencia comando remoto.

6. Se abre la llave de un SIS N en la línea de FE.
7. Se envía desde la central operativa un comando COCHE_DERIVA de manera periódica cada 1 segundo.
8. Se verifica que se haya interrumpido las señales de CT y liberado la de FE.
9. Se envía el comando CANCEL una única vez y se deja de transmitir el comando anterior.
10. Se verifica que el sistema vuelve al estado normal.
11. Se abre la llave de un SIS N en la línea de CT.
12. Se envía desde la central operativa un comando AISLADO_TOTAL de manera periódica cada 1 segundo durante 60 segundos.
13. Se verifica que se hayan liberado las señales de CT y de FE.
14. Se verifica que después de 10 segundos después de finalizada la última transmisión, el sistema vuelve al estado normal.
15. Se envía el comando COMMAND_VALIDITY_CONFIG:20.
16. Se envía el comando INTERMITENTE:1 de manera periódica durante 120 segundos.
17. Se verifica que el sistema entró en modo intermitente con el perfil 1 seleccionado ejecutando la rutina configurada (T=15, D=30, N=4, E=40).
18. Se verifica que después de 20 segundos después de finalizada la última transmisión (valor configurado), el sistema vuelve al estado normal.
19. Se envía el comando INTERMITENTE_CONFIG:4,20,40,5,60).
20. Se envía el comando INTERMITENTE:4 de manera periódica durante 120 segundos.
21. Se verifica que el sistema entró en modo intermitente con el perfil 4 seleccionado ejecutando la rutina con los parámetros configurados (T=20, D=40, N=5, E=60).
22. Se verifica que después de 20 segundos después de finalizada la última transmisión (valor configurado), el sistema vuelve al estado normal.

Prueba 8 - Registro de eventos

1. Se enciende el SAL/T en estado normal.
2. Para todos los mensajes de logeos, se verifica que la fecha y hora registrada sea correcta.
3. Se verifica durante toda la prueba que el sistema reporte por el canal de STATUS de MQTT el estado actual del sistema.
4. Se abren varias llaves de un SIS N para la línea X (CT o FE).
5. Se verifica la recepción del cambio de estado de cada llave en el canal de logs de MQTT desde la central operativa.
6. Se activa el modo aislado limitado y se verifica el reporte de cambio de modo.
7. Se verifica el reporte de la acción de aislamiento de los SIS intervenidos.
8. Se pierde la fuente de medición RS-485_1 de velocidad y se verifica el reporte de cambio de fuente de velocidad a la siguiente disponible.

9. Se pierden todas las fuentes de velocidad por lo que se activa el modo intermitente y se verifica el reporte de cambio de modo de operación.
10. Se verifica que se reporte cada activación y desactivación de las señales CT y FE.
11. Se restaura las fuentes de velocidad y se verifica el reporte del cambio de modo.
12. Se varía la velocidad y se verifica el reporte de las velocidades transcurridas en modo aislado limitado.
13. Se activa la llave de modo aislado total y se verifica el reporte del cambio de modo.
14. Se verifica el reporte de desintervención de los SIS afectados ya el SAL/T controla las señales de CT Y FE directamente.
15. Se desactivan las llaves MAL y MAT para restablecer el modo normal y se verifica el reporte de cambio de modo.
16. Se desconecta la antena GPS y se verifica que se reporte la perdida de cobertura y referencia de zona.
17. Se reconecta la antena GPS y se verifica el reporte de cobertura y referencia de zona 1.
18. Se envía el comando DOWNLOAD_LOGS desde la central operativa y se verifica la recepción de todos los eventos transmitidos previamente con la fecha y hora originales de los eventos.
19. Se conecta una PC por USB al SAL/T por comunicación serie, se envía de manera local el comando DOWNLOAD_LOGS y se verifica la recepción de todos los eventos transmitidos previamente con la fecha y hora originales de los eventos.

Prueba 9: Conectividad Wi-Fi

1. Se enciende el SAL/T en estado normal.
2. Se apaga la red Wi-Fi principal a la que se conecta el sistema.
3. Se verifica la recepción de un registro de evento de cambio de red a la red secundaria.
4. Se apaga la red Wi-Fi secundaria a la que se conecta el sistema.
5. Se verifica que se haya perdido la comunicación con la central operativa.
6. Se prende cualquiera de las redes Wi-Fi configuradas.
7. Se verifica que la conexión con la central operativa se restablece en pocos segundos y reporta la red a la que se conectó.

Las pruebas integrales descriptas cubren el 100 % de los requerimientos y se diseñaron de manera temprana en el proceso de desarrollo del firmware. Esto permitió tener claro cuál es el comportamiento esperado del sistema en distintas situaciones e ir escribiendo un firmware que responda a estas pruebas. Luego de varias iteraciones y pruebas, se obtuvo un resultado exitoso en todas las pruebas para la versión final del sistema.

5 Conclusiones

En este capítulo se presentan las conclusiones generales obtenidas a partir de la realización del proyecto y los próximos pasos a seguir para la implementación del sistema en formaciones ferroviarias operativas.

5.1 Conclusiones generales

En este trabajo se logró diseñar y armar un nuevo prototipo de un sistema de aislado limitado / total que cumple con todos los requerimientos acordados para el proyecto. Esto permite acercar el objetivo de realizar la fabricación de un SAL/T de manera nacional y mejorar la seguridad en la operación de las formaciones ferroviarias causando una mejora directa en la vida de las personas que utilizan este medio de transporte de manera cotidiana.

En este prototipo, se implementó un hardware más moderno que su versión anterior, pero sin dejar de reutilizar muchas de sus técnicas de diseño y lógicas de circuito. Esto permitió que este prototipo agregue funcionalidades y mejoras sobre el sistema sin dejar de contar con la robustez del prototipo predecesor. En cuanto al firmware, se logró implementar una lógica del sistema utilizando un sistema operativo en tiempo real que permita orquestar de manera eficiente los recursos y los tiempos de respuesta de cada uno de los subsistemas incluidos en el SAL/T.

Las pruebas integrales realizadas simulando todos los dispositivos externos propios de una formación ferroviaria permiten confirmar que el prototipo cumple con el comportamiento esperado en todas las situaciones especificadas en los requerimientos. Sin embargo, para poder validar el correcto funcionamiento del sistema en condiciones operativas de una formación ferroviaria, queda pendiente realizar pruebas conectadas a los sistemas reales de uso del SAL/T. Además de las potenciales diferencias con el banco de pruebas utilizado, es esperable que algunas funcionalidades o decisiones de implementación hayan quedado alejadas o diferentes a las imaginadas por el autor del pliego de especificaciones de Trenes Argentinos, ya que durante todo el proyecto no se logró mantener una iteración constante con la empresa para validar las decisiones de diseño y, si bien responde a los requerimientos del pliego, la interpretación de este puede llevar a distintos enfoques o pretensiones.

La etapa de relevamiento de los requerimientos y definición del alcance del proyecto resultó la etapa más relevante del trabajo porque todas las tareas subsiguientes se desprenden de esta primera etapa. Luego, el diseño de hardware significó la mayor carga de horas de trabajo por su naturaleza de incorporación de todos los requerimientos del proyecto y rigidez que conlleva la fabricación de un hardware incorrecto. Por esto, se hizo una primera etapa de diseño integral de la solución donde se estableció de manera modular los distintos subsistemas del SAL/T y la comunicación de cada uno con el controlador principal. Luego, se diseñaron e implementaron de manera independiente cada uno de los módulos respetando las interfaces definidas previamente. De esta manera se logró realizar un proceso ordenado de diseño desde lo general a lo particular.

Este trabajo tiene como mayor aporte la fabricación de un prototipo que considera todos los requerimientos para armar un sistema de seguridad productivo para instalar en las formaciones ferroviarias de la empresa Trenes Argentinos. El detalle de su diseño y consideraciones queda plasmado en este documento lo que permite entender y reutilizar una gran parte del proyecto al momento de realizar una versión productiva.

A nivel personal, este trabajo permitió consolidar y poner en práctica de manera integral gran parte de los conocimientos adquiridos durante la carrera de grado de ingeniería electrónica realizando un proyecto que atraviesa las etapas de relevamiento de los requerimientos, diseño de

hardware, diseño de firmware, fabricación, implementación, verificación, diseño y ejecución de pruebas y presentación del proyecto mismo.

5.2 Próximos pasos

Este trabajo acerca la idea de implementar un sistema de seguridad en las formaciones ferroviarias, pero para poder cumplir ese objetivo queda pendiente la realización de otros proyectos que se van a beneficiar y desprender de este. Algunos de los próximos pasos son los siguientes:

- Finalización del armado del software para la central operativa. Esto debe contar con interfaces de monitoreo, bases de datos donde almacenar la información y la capacidad de interactuar con distintas formaciones.
- Pruebas sobre laboratorios de Trenes Argentinos utilizando los sistemas externos reales para las entradas y salidas del sistema.
- Integración del microcontrolador y los circuitos utilizados de la placa Nucleo 144 sobre la placa de procesamiento diseñada para el SAL/T.
- Reemplazo de algunos componentes por su versión de mayor seguridad y robustez como pueden ser los relés de seguridad, los módems 4G - Wi-Fi o la antena para la conectividad GPS. Estos cambios fueron previstos para una próxima versión por lo que el diseño de hardware permite estas modificaciones de componentes sin cambios en el diseño del PCB.
- Fabricación de un gabinete con la robustez necesaria para su instalación en una formación ferroviaria considerando las vibraciones e interferencia electromagnética presentes en el entorno de operación.
- Diseño e implementación del sistema de adaptación de la señal del generador de impulsos ópticos a una comunicación RS-485.
- Diseño e implementación de una fuente de alimentación externa para el SAL/T que permite la reducción de tensión de las baterías de la formación en el rango de 72-110 VDC a 5 VDC.

6 Bibliografía

Referencias

- [1] **GICSAFe.** Grupo de Investigación en Calidad y Seguridad de las Aplicaciones Ferroviarias. <https://sites.google.com/view/conicet-gicsafe>.
- [2] **Trenes Argentinos** Trenes Argentinos Operaciones, empresa del estado. <https://www.argentina.gob.ar/transporte/trenes-argentinos>.
- [3] **Supervision system for emergency brake and traction-system isolation on train formations.** Di Vito Iván Mariano, Gomez Pablo Martin, Lutenberg Ariel, 2020. <https://ieeexplore.ieee.org/document/9085280>.
- [4] **Aplicación de la técnica de patrones de diseño a la implementación de un Sistema de Aislamiento Limitado/Total ferroviario.** Di Vito, Iván Mariano. 2019 <https://bibliotecadigital.fi.uba.ar/items/show/18446>.
- [5] **UNE-EN 50126-1.** Aplicaciones ferroviarias. Especificación y demostración de la fiabilidad, la disponibilidad, la mantenibilidad y la seguridad (RAMS). 2005. <https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0033106>
- [6] **Design Patterns for Safety-Critical Embedded Systems.** Ashraf Armoush. RWTH Aachen University. 2010.
- [7] **Sistema de supervisión de la seguridad del material ferroviario utilizando patrones de diseño.** Ivan Mariano Di Vito, Pablo Gomez, Ariel Lutenberg, Adrian Laiuppa. 2019. <https://drive.google.com/file/d/1FDtRL4We4daZ7XrJ81sL2f6HrZbYlOuG/view>.
- [8] **SISTEMA DE AISLADO LIMITADO / TOTAL (SAL-T).** ESPECIFICACIÓN TÉCNICA. Bypass Sistemas Instrumentados de Seguridad a Bordo del Material Rodante ET.SO. No 046 /18 – E3. 2022.
- [9] **UNE-EN 61508-4.** Seguridad funcional de los sistemas eléctricos/electrónicos/electrónicos programables relacionados con la seguridad. Parte 4: Definiciones y abreviaturas. 2011. <https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0047025>
- [10] **IPC-2221.** Estándar genérico sobre la PCB. Revisión C; 2023. <https://shop.ipc.org/ipc-2221/ipc-2221-standard-only/Revision-c/english>
- [11] **UNE-EN 50128.** Aplicaciones ferroviarias. Sistemas de comunicación, señalización y procesamiento. Software para sistemas de control y protección del ferrocarril. 2012. <https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0049040>
- [12] **UNE-EN 50129.** Aplicaciones ferroviarias. Sistemas de comunicación, señalización y procesamiento. Sistemas electrónicos relacionados con la seguridad para la señalización. 2020. <https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0063513>
- [13] **Random, systematic, and common cause failure: How do you manage them?.** Michela Gentile, Angela E. Summers. 2006. https://sis-tech.com/wp-content/uploads/2005/10/Random_Systematic_and_Common_Cause_Failure.pdf
- [14] **STM32 Nucleo-144 development board with STM32F429ZI MCU.** STMicroelectronics. <https://www.st.com/en/evaluation-tools/nucleo-f429zi.html>

- [15] **STM32F429ZI.** High-performance advanced line, Arm Cortex-M4 core. STMicroelectronics. <https://www.st.com/en/microcontrollers-microprocessors/stm32f429zi.html>
- [16] **ST-LINK/V2-1.** In-Circuit Debugger and Programmer. STMicroelectronics. <https://www.st.com/en/development-tools/st-link-v2.html>
- [17] **STM32CubeIDE.** Free Development Environment for STM32 Microcontrollers. STMicroelectronics. <https://www.st.com/en/development-tools/stm32cubeide.html>
- [18] **STM32CubeMX.** Initialization Code Generator for STM32 Microcontrollers. STMicroelectronics. <https://www.st.com/en/development-tools/stm32cubemx.html>
- [19] **ESP32-DevKitC.** Development board from Espressif. <https://docs.espressif.com/projects/esp-dev-kits/en/latest/esp32/esp32-devkitc/index.html>
- [20] **IEEE 802.11.** IEEE Computer Society, IEEE Standard for Information Technology—Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks—Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Standard 802.11-2020, 2021. https://standards.ieee.org/standard/802_11-2020.html
- [21] **Bluetooth.** Bluetooth Core Specification Version 5.4. 2023. <https://www.bluetooth.com/specifications/bluetooth-core-specification/>
- [22] **Arduino IDE.** Open source repository. <https://github.com/arduino/arduino-ide>
- [23] **ESP-IDS.** Espressif IoT Development Framework. <https://idf.espressif.com/>
- [24] **PlatformIO.** Cross-platform, cross-architecture, multiple framework, professional tool for embedded systems designers. <https://platformio.org/>
- [25] **Modulo GPS GY-NEO6MV2.** Módulo basado en NEO-6M. <https://www.todomicro.com.ar/ARDUINO/339-modulo-gps-gy-neo6mv2-con-antena.html>
- [26] **NEO-6M.** Módulo GNSS / GPS. Ublox. <https://www.u-blox.com/en/product/neo-6-series>
- [27] **Getting to know MQTT.** Michael Yuan. 2021. <https://developer.ibm.com/articles/iot-mqtt-why-good-for-iot/>
- [28] **What is MQTT.** Tobias Goebel Twilion. 2023. <https://www.twilio.com/en-us/blog/what-is-mqtt>
- [29] **Wi-Fi Security.** Wi-Fi Alliance, Wi-Fi Security Technologies: WEP, WPA, and WPA2. <https://www.wi-fi.org/discover-wi-fi/security>
- [30] **Wi-Fi Direct.** Wikipedia. https://en.wikipedia.org/wiki/Wi-Fi_Direct
- [31] **GPS.** U.S. Department of Defense, Global Positioning System Standard Positioning Service Performance Standard, 4th ed. 2008. <https://www.gps.gov/technical/ps/>
- [32] **Cómo funcionan los dispositivos GPS? Trilateración vs Triangulación.** Gabri. Arcgeek. 2018. <https://acolita.com/como-funcionan-los-dispositivos-gps-trilateracion-vs-triangulacion/>
- [33] **Texas Instruments, RS-485:** Theory and Application, Application Report SLAA070D. 2004. <https://www.ti.com/lit/an/slaa070d/slaa070d.pdf>

- [34] **What is a Real-Time Operating System (RTOS)?** Digikey article. 2021. <https://www.digikey.com/en/maker/projects/what-is-a-realtime-operating-system-rtos/28d8087f53844decafa5000d89608016>
- [35] **Hasler Rail.** Speed Sensing and Odometry. 2016 https://www.haslerrail.com/wp-content/uploads/2019/03/HaslerRail_DataRecordingSafety_EN_v1.pdf
- [36] **Mouser Electronics.** Global distributor of semiconductors and electronic components. <https://www.mouser.com/>
- [37] **DigiKey Corporation.** Global distributor of semiconductors and electronic components. <https://www.digikey.com/>
- [38] **KiCad EDA.** Version 6. KiCad Developers. <https://www.kicad.org/>
- [39] **LTSpice.** Free SPICE simulator. Analog devices. <https://www.analog.com/en/resources/design-tools-and-calculators/ltpice-simulator.html>
- [40] **GitHub, Inc..** A Platform for Version Control and Collaboration, <https://github.com>
- [41] **Git** A Distributed Version Control System. Linus Torvalds. 2005. <https://git-scm.com>
- [42] **martinanus/salt** Repositorio de firmware y hardware del proyecto. Martín Anús. 2024. <https://github.com/martinanus/salt>
- [43] **INA823DGKR.** Texas Instruments. Instrumentation Amplifiers Precision. <https://www.ti.com/lit/gpn/ina823>
- [44] **ICL7660AIBAZA.** Renesas. Switching Voltage Regulators. https://ar.mouser.com/datasheet/2/698/REN_icl7660s_a_DST_20200210-1997724.pdf
- [45] **JW2SN-DC5V.** Panasonic Industrial Devices. General Purpose Relays 5A 5VDC DPDT SEALED PCB. <https://ar.mouser.com/manufacturer/panasonic-industrial-devices/>
- [46] **V23047A1005A501.** TE Connectivity. Safety Relays 2 CO CONTACTS 5VDC https://ar.mouser.com/datasheet/2/418/6/ENG_DS_SR2M_0820-735797.pdf
- [47] **UNE-EN 61810-3.** Relés electromecánicos elementales. Parte 3: Relés con contactos guiados (unidos mecánicamente). 2015. <https://www.une.org/encuentra-tu-norma/buscata-norma?c=N0054926>
- [48] **Relay Terminology.** Altech Corp. <https://oylair.com/wp-content/uploads/2021/03/resources-library-3-relay-terminology.pdf>
- [49] **ULN2003D1013TR-** STMicroelectronics. Darlington Transistors Seven NPN Array. <https://ar.mouser.com/datasheet/2/389/uln2001-1852702.pdf>
- [50] **Darlington's Contributions to Transistor Circuit Design** David A. Hodges, Fellow, IEEE. 1999. <https://people.eecs.berkeley.edu/~hodges/DarlingtonCircuit.pdf>
- [51] **Use of Event Data Recorders in Rail Transit.** Federal Transit Administration, , FTA Report No. 0123. 2015. <https://www.transit.dot.gov/research-innovation/use-event-data-recorders-rail-transit>
- [52] **Interruptor Bipolar 20A de 2 posiciones.** Elibet. Serie LI. <https://www.elibet.com/wp-content/uploads/2014/08/Elibet-20A.pdf>
- [53] **Data Recording and Safety.** Hasler Rail. http://www.iotrains.com/fileadmin/user_upload/04_DataRecording.pdf

- [54] **Catalogue of parametrised CRC algorithms with 16 bits.** Greg Cook. <http://reveng.sourceforge.net/crc-catalogue/16.htm>
- [55] **EDU-CIAA-NXP.** Información general de la placa de desarrollo EDU-CIAA-NXP desarrollada por el proyecto CIAA (Computadora Industrial Abierta Argentina). <https://www.proyecto-ciaa.com.ar/devwiki/doku.php%3Fid=desarrollo:edu-ciaa:edu-ciaa-nxp.html>
- [56] **SN65HVD1176DR.** Texas Instruments. CI interfaz RS-485 PROFIBUS RS-485 Transceiver. <https://www.ti.com/lit/gpn/sn65hvd1176>
- [57] **Profibus International. PROFIBUS Technology Overview.** <https://www.profibus.com/technologies/profibus>
- [58] **SZP6SMB12CAT3G.** Littelfuse. ESD Protection Diodes / TVS Diodes. https://www.littelfuse.com/media?resourceType=datasheets&itemID=a54ab340-efe4-40d7-99cb-dd80d69f6f3a&filename=littelfuse_tvs_diode_szp6smb_datasheet.pdf
- [59] **MF-USMF020-2.** Bourns. Resettable Fuses. <https://www.mouser.co.uk/datasheet/2/54/mfusmf-777646.pdf>
- [60] **SI2343DS-T1-E3.** Vishay Semiconductors. MOSFETs 30V 4.0A 1.25W 53 mohms @ 10V. <https://www.vishay.com/doc?72079>
- [61] **87583-2010BLF.** Amphenol FCI. USB Connectors 4P RECEPTACLE TYPE A. <https://www.mouser.co.uk/datasheet/2/18/1/87583-2579162.pdf>
- [62] **Módem router Portátil USB 4G LTE Wi-Fi.** Altanet. <https://www.voipexperts.com.ar/productos/routers/routers-4g/alta-stick-st10.html>
- [63] **DM3AT-SF-PEJM5.** Hirose Connector. Memory Card Connectors R/A SMT MICROSD CON PUSH-PUSH. https://www.mouser.co.uk/datasheet/2/185/DM3AT_SF_PEJM5_CL0609_0031_0_00_2DDrawing_00009471-1614303.pdf
- [64] **Micro SD Ultra 8GB Class 10.** SanDisk. https://dcpsoltec.mercadoshops.com.ar/MLA-1346308183-micro-sd-sandisk-ultra-8gb-clase-10-_JM
- [65] **AS1115-BSST.** ams OSRAM. LED Display Drivers LED Driver with Keyscan. https://www.mouser.co.uk/datasheet/2/588/asset_pdf_25493045-3432722.pdf
- [66] **WP154A4SEJ3VBDZGW/CA.** Kingbright. Standard LEDs - Through Hole 5MM RGB LED. https://www.mouser.co.uk/datasheet/2/216/WP154A4SEJ3VBDZGW_CA-1145236.pdf
- [67] **WP7113LZGCK.** Kingbright. Standard LEDs - Through Hole 5MM LOW CURRENT GREEN LED. <https://www.mouser.co.uk/datasheet/2/216/WP7113LZGCK-535810.pdf>
- [68] **LTC-5723HR.** Lite-On. LED Displays and Accessories 4 Digit, Red. <https://www.mouser.co.uk/datasheet/2/239/C5723HR-1141844.pdf>
- [69] **1543-650-149.** Bourns. Pushbutton Switches Push Switch. <https://www.mouser.co.uk/datasheet/2/54/1543-55078.pdf>
- [70] **CEM-1205-IC.** Same Sky. Piezo Buzzers and Audio Indicators buzzer 5V, driving circuit. https://www.mouser.co.uk/datasheet/2/1628/cem_1205_ic-3509057.pdf
- [71] **NVTR0202PLT1G.** onsemi. MOSFETs PFET 20V 0.4A 80MOH. https://www.mouser.co.uk/datasheet/2/308/1/NTR0202PL_D-2319410.pdf

- [72] **JLCWAY**. PCB Manufacturing and Assembly Services. <https://jlpcb.com>
- [73] **Impresiones 3D**. Laboratorio Abierto de la Facultad de Ingeniería de la Universidad de Buenos Aires. <https://labi.fi.uba.ar/impresiones-3d>
- [74] **Caja estanca plástica 310x310x110 Blanca IP65**. Sistelectric de Genros. <https://sistelectric.com/producto/06313111g/caja-estanca-plastica-310x310x110-gris-ip65/>
- [75] **FreeRTOS**. Real-Time Operating System for Embedded Devices. <https://www.freertos.org>
- [76] **GPS - NMEA sentence information**. Glenn Baddeley. 2001. <https://aprs.gids.nl/nmea/>
- [77] **NTP Server** The NTP Pool Project <https://www.pool.ntp.org>
- [78] **shiftr.io**. MQTT Broker and Dataflow Visualizer <https://www.shiftr.io>
- [79] **Arduino**. Arduino Uno Rev3. <https://docs.arduino.cc/hardware/uno-rev3/>
- [80] **Modulo Conversor Rs485 Ttl Max485 Transceiver**. Hobbytronica. https://www.hobbytronica.com.ar/MLA-1412581125-x5-modulo-conversor-rs485-ttl-max485-transceiver-arduino-_JM
- [81] **ATMega328-p**. 8-bit AVR Microcontroller. Microchip Technology Inc. <https://www.microchip.com/wwwproducts/en/ATmega328p>
- [82] **Software Serial Library**. Arduino. <https://docs.arduino.cc/learn/built-in-libraries/software-serial/>
- [83] **Hobbytrónica**. Casa de electrónica y robótica. <https://www.hobbytronica.com.ar/>
- [84] **MAX485**. Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers. Analog devices. <https://www.analog.com/en/products/max485.html>
- [85] **Paho MQTT Python Client Library**. Eclipse Foundation. <https://eclipse.dev/paho/>
- [86] **Modulo lector de tarjetas SD**. Electrouno. https://www.unoelectro.com.ar/MLA-864661170-modulo-lector-de-tarjetas-micro-sd-arduino-unoelectro-_JM