



**FACULTAD
DE INGENIERIA**

Universidad de Buenos Aires

Sistema de ensayos de relés ferroviarios de seguridad basado en computación en la nube

Autor:

Ing. Gaspar Santamarina

Director:

Ing. Adrián Laiuppa (CONICET-GICSAFe)

Jurados:

Nombre y Apellido (1) (pertenencia (1))

Nombre y Apellido (2) (pertenencia (2))

Nombre y Apellido (3) (pertenencia (3))

*Este trabajo fue realizado en el curso de Gestión de proyectos
entre el 22 de junio de 2020 y el 22 de Agosto de 2020.*

Índice

Registros de cambios	3
Acta de constitución del proyecto.	4
Descripción técnica-conceptual del Proyecto a realizar	5
Identificación y análisis de los interesados.	7
1. Propósito del proyecto	7
2. Alcance del proyecto	7
3. Supuestos del proyecto.	7
4. Requerimientos	7
5. Entregables principales del proyecto	9
6. Desglose del trabajo en tareas	9
7. Diagrama de Activity On Node	11
8. Diagrama de Gantt.	12
9. Matriz de uso de recursos de materiales	14
10. Presupuesto detallado del proyecto	14
11. Matriz de asignación de responsabilidades	15
12. Gestión de riesgos	16
13. Gestión de la calidad	17
14. Comunicación del proyecto	20
15. Gestión de Compras	20
16. Seguimiento y control.	21
17. Procesos de cierre	22

Registros de cambios

Revisión	Detalles de los cambios realizados	Fecha
1.0	Creación del documento: propósito, alcance, supuestos y tareas.	09/07/2020
1.1	Se agrega AoN, diagrama Gantt, matriz de recursos, presupuesto y matriz de responsabilidades.	30/07/2020
1.2	Se agrega Gestión de riesgos, calidad y compras, comunicación del proyecto, seguimiento y proceso de cierre. Se cambia “sistema” por “firmware” en los requerimientos. Se agrega transformador al presupuesto. Se corrigen requerimientos. Se agrega entregable (informe de ensayos).	08/08/2020

Acta de constitución del proyecto

Buenos Aires, 22 de junio de 2020

Por medio de la presente se acuerda con el Ing. Ing. Gaspar Santamarina que su Trabajo Final de la Carrera de Especialización en Sistemas Embebidos se titulará “Sistema de ensayos de relés ferroviarios de seguridad basado en computación en la nube”, consistirá esencialmente en el prototipo preliminar de un probador de relés ferroviarios de seguridad, basado en hardware digital, con la posibilidad de ser operado de forma remota, y tendrá un presupuesto preliminar estimado de 687 hs de trabajo, con fecha de inicio 22 de junio de 2020 y fecha de presentación pública 22 de junio de 2021.

Se adjunta a esta acta la planificación inicial.

Ariel Lutenberg
Director posgrado FIUBA

Martín Harris
Trenes Argentinos

Ing. Adrián Laiuppa
Director del Trabajo Final

Nombre y Apellido (1)
Jurado del Trabajo Final

Nombre y Apellido (2)
Jurado del Trabajo Final

Nombre y Apellido (3)
Jurado del Trabajo Final

Descripción técnica-conceptual del Proyecto a realizar

Las barreras automáticas de los pasos a nivel y los sistemas de cambios de vía del sistema ferroviario de la Argentina dependen mayormente de componentes electromecánicos. Estos componentes deben cumplir altos niveles de seguridad para alcanzar la fiabilidad necesaria. Un componente importante de estos sistemas son los relés de señalamiento, llamados también relés de seguridad (“safety relays” en inglés) o relés vitales (“vital relays” en inglés).

Un único paso a nivel automático puede emplear decenas de relés. Estos solo se consiguen por importación y cada uno tiene un valor superior a los U\$S 1,000 (mil dólares estadounidenses). Es conveniente entonces el desarrollo de una industria nacional que fabrique relés certificados; un sistema de ensayos de relés es una parte esencial para elaborar una certificación local.

El diseño del sistema de ensayos de relés se basa en la norma UNE-EN 50578. En la misma se establecen valores máximos y mínimos en las variaciones de los valores iniciales de la corriente de excitación, la corriente de caída y el factor K. El sistema de ensayos de relés debe permitir el monitoreo de estos valores eléctricos.

En la Figura 1 se puede ver la arquitectura completa del sistema, incluyendo el servidor remoto encargado de monitorear el ensayo a lo largo de todo el proceso.

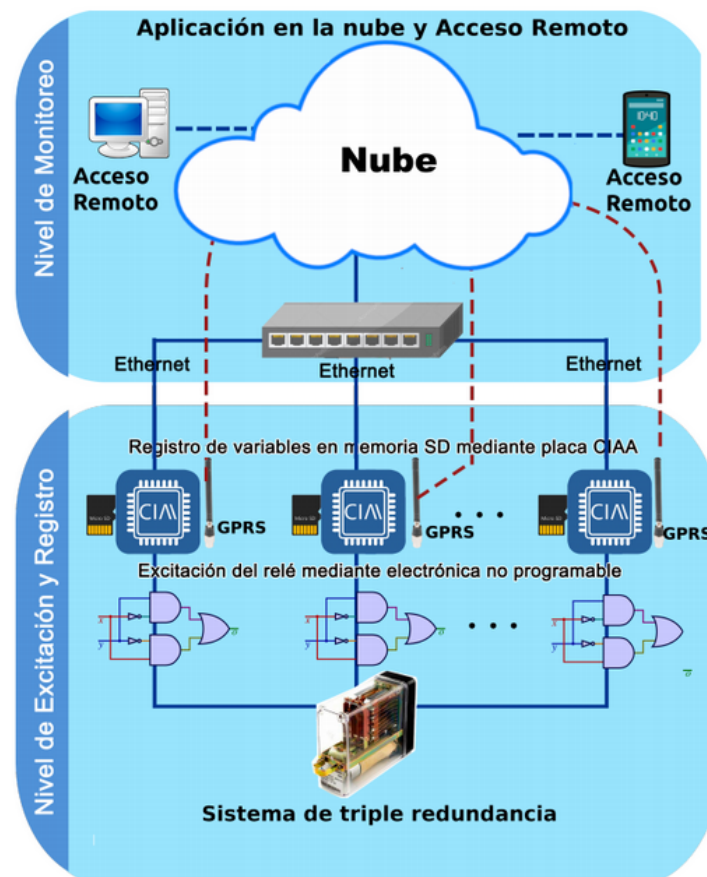


Figura 1: Diagrama en bloques del sistema

El firmware correrá sobre 3 placas CIAA-NXP en simultáneo para asegurar los niveles de disponibilidad deseados. Sobre cada una de las CIAA-NXP se monta una placa de lógica no programable. Este subsistema cuenta entre otras con las siguientes funciones:

- Generar la señal de activación del relé (incluyendo los transistores de potencia y circuitos de activación correspondientes).
- Realizar la sincronización automática de las señales de activación del relé generadas por tres placas independientes entre sí.
- Detectar cuando una de las tres placas falla y en ese caso desconectarla del relé en forma automática.
- Realizar la medición de múltiples valores de voltaje y corriente asociados al funcionamiento del relé. Estos valores son el voltaje aplicado al bobinado del relé y la corriente que circula, junto con los valores de voltaje aplicado y la corriente que circula por cada contacto. Se ensayarán relés que usualmente cuentan con hasta seis contactos normal cerrado y seis contactos normal abierto.
- Mostrar indicaciones luminosas relativas al estado del sistema mediante diodos LED.
- Establecer una comunicación MQTT sobre TCP/IP con el broker para recibir los comandos del usuario y enviar, en tiempo real, los datos recabados.

Para cumplir con los requerimientos de la norma, el sistema debe ser capaz de realizar los siguientes ensayos:

- **Sistema magnético, corriente de excitación y corriente de caída:** la corriente de excitación se define como la corriente mínima a través de la bobina que, partiendo de un valor nulo, alcanza para mover la armadura de la posición de reposo a la posición de trabajo y para aplicar la fuerza de contacto especificada por el fabricante, cerrando todos los contactos de trabajo. La corriente de caída es la corriente máxima a través de la bobina que, partiendo del valor de la corriente nominal, produce la apertura de todos los contactos de trabajo. Este ensayo permite medir el factor K. Para esto se debe generar una función de rampa que, luego de ser amplificada, se aplica al bobinado del relé.
- **Ensayo de vida útil mecánica:** este ensayo es en vacío. El usuario puede ingresar la cantidad de ciclos a ensayar siendo el valor máximo de $10 \cdot 10^6$ movimientos porque la norma se cumple cuando se llega esta cantidad de ciclos. Este ensayo registra la corriente y el voltaje del bobinado y el estado de los contactos.
- **Ensayo con carga:** este ensayo se realiza aplicando el voltaje nominal los contactos abiertos y la corriente nominal a los contactos cerrados. A la bobina del relé también se le aplica el voltaje nominal. La norma dice que en estas condiciones se debe asegurar una cantidad mínima de $2 \cdot 10^6$ movimientos durante la vida útil del relé. En este ensayo se registran los voltajes y las corrientes de la bobina y los voltajes y las corrientes de los contactos.

Identificación y análisis de los interesados

Rol	Nombre y Apellido	Organización	Puesto
Cliente	Martín Harris	Trenes Argentinos	Coordinador General de Desarrollo, de la Subgerencia de Desarrollo y Normas Técnicas
Responsable	Ing. Gaspar Santamarina	CESE 12Co2020	Alumno
Orientador	Ing. Adrián Laiuppa	CONICET-GICSAFe	Director Trabajo final
Colaboradores	Ing. Nicolás Locatelli	CONICET-GICSAFe	Integrante
	Ing. Gustavo Ramoscelli	CONICET-GICSAFe	Investigador

CONICET-GICSAFe: Grupo de Investigación en Calidad y Seguridad de las Aplicaciones Ferroviarias

1. Propósito del proyecto

El propósito de este proyecto es desarrollar el firmware para un probador de relés de seguridad ferroviarios. El mismo deberá comunicarse con un servidor remoto, desde el cuál se configurarán e iniciarán las pruebas. Además, deberá reportar en tiempo real el estado de las pruebas, junto con las mediciones que la conforman.

2. Alcance del proyecto

Se diseñará e implementará el firmware capaz de realizar, junto con el hardware, las tres pruebas mencionadas anteriormente en la descripción técnica del proyecto. En la entrega final no se incluirá el desarrollo de hardware del probador ni el software del servidor remoto.

3. Supuestos del proyecto

- Se contará con el hardware terminado al arrancar con el desarrollo de firmware.
- Se dispondrá, en una etapa avanzada del desarrollo, de un servidor de pruebas con las funcionalidades mínimas requeridas por el probador.

4. Requerimientos

1. Interfaces Externas

- 1.1. El firmware deberá ser capaz de leer el estado de 5 señales digitales.
- 1.2. El firmware deberá leer 15 señales analógicas a través de 2 integrados ADC externos.
- 1.3. El firmware deberá leer 2 señales analógicas a través de 2 ADC internos.
- 1.4. El firmware deberá leer el estado de un pin digital.
- 1.5. El firmware deberá controlar el estado de 2 pines digitales.

- 1.6. El firmware deberá comunicarse con la tarjeta MicroSD presente en el hardware.
- 1.7. El firmware deberá hacer uso de un puerto Ethernet para conectarse a Internet.
- 1.8. El firmware deberá ser capaz de generar una rampa de tensión configurable.

2. Funciones

- 2.1. El firmware debe ser capaz de enviar notificaciones de alerta al servidor remoto, generadas por el disparo de una señal de alerta del hardware.
- 2.2. El firmware debe ser capaz de enviar notificaciones de alerta al servidor remoto, generadas por sobretensión en cualquiera de las entradas analógicas.
- 2.3. El firmware debe ser capaz de interpretar los siguientes comandos enviados desde el servidor:
 - 1) Aplicar configuración de sistema.
 - 2) Definir los parámetros asociados a cada tipo de ensayo.
 - 3) Disparar ensayos.
 - 4) Parar ensayos.
- 2.4. El firmware deberá ser capaz de ejecutar los siguientes ensayos:
 - 1) **Sistema magnético, corriente de excitación y corriente de caída:** Se genera una rampa de tensión de pendiente positiva y luego otra de pendiente negativa. El valor inicial, valor final, paso de tensión y duración del paso de las señales deben ser configurables en ambos casos, además de la máxima corriente de excitación admitida. Se debe notificar al servidor entre paso y paso, indicando la tensión instantánea de la rampa y el valor de las 17 señales analógicas. La prueba finaliza al completarse la cantidad de repeticiones especificadas al inicio de la prueba.
 - 2) **Ensayo de vida útil mecánica con y sin carga:** Se debe especificar el número de ciclos. Al detectarse la secuencia de inicio se comienzan a contabilizar los ciclos, representados por el periodo entre cada secuencia de inicio. Se debe notificar al servidor cada vez que ocurre un flanco positivo de las señales CLOCK_1S o CLOCK_3S, indicando el número de ciclo y el valor de las 17 señales analógicas. La prueba finaliza al completarse la cantidad de ciclos especificados al inicio de la prueba.
- 2.5. Junto con cada paquete de datos generado a partir de una prueba, se debe anexar un ID que identifique el tipo de prueba.
- 2.6. Junto con cada paquete de datos se debe anexar una marca temporal y un ID que identifique a la placa dentro de las tres placas que conforman el sistema.
- 2.7. Cada paquete de datos debe ser almacenado en una tarjeta MicroSD a modo de copia de seguridad.
- 2.8. El firmware debe permitir realizar distintos tipos de ensayos en forma intercalada.
- 2.9. El firmware debe permitir configurar los siguientes parámetros:
 - 1) Hora del sistema.
 - 2) URL y puerto del broker MQTT.

3. Requisitos de Rendimiento

- 3.1. Para el ensayo de rampa se deben permitir hasta 25 ciclos.
- 3.2. Para el ensayo de vida útil mecánica **sin carga** se deben permitir hasta $10 \cdot 10^6$ ciclos.
- 3.3. Para el ensayo de vida útil mecánica **con carga** se deben permitir hasta $2 \cdot 10^6$ ciclos.

3.4. Para el ensayo de vida útil mecánica el periodo máximo entre ciclos es de 3 segundos.

4. Restricciones de Diseño

- 4.1. Se usará I²C como protocolo de comunicación con los ADC.
- 4.2. Se usará SPI como protocolo de comunicación con la SD.
- 4.3. Se usará MQTT sobre TCP/IP como protocolo de comunicación con el servidor remoto.
- 4.4. Se usará JSON como formato de texto para intercambio de datos con el servidor remoto.
- 4.5. La rampa de tensión debe ser capaz de alcanzar un valor máximo de hasta 10V.
- 4.6. Se usará la plataforma CIAA-NXP como placa de desarrollo.
- 4.7. Se usará un RTC para mantener la hora de forma local.
- 4.8. Se usará “Unix time” como formato de marca temporal.

5. Atributos del Sistema

- 5.1. El firmware debe ser capaz de recuperarse ante una falla que provoque el reinicio del microcontrolador, en un tiempo menor a los 3 segundos y retomar los ensayos en curso.
- 5.2. El firmware debe hacer uso de cifrado TLS para la capa de transporte.
- 5.3. El firmware debe hacer uso de Usuario/Contraseña para autenticarse con el broker MQTT.
- 5.4. El desarrollo de software debe seguir una metodología acorde a la norma UNE-EN 50128.

6. Otros Requisitos

- 6.1. El desarrollo debe ser documentado utilizando la herramienta Doxygen.

5. Entregables principales del proyecto

- Código fuente del firmware
- Informe de avance
- Memoria del trabajo
- Informes de ensayos

6. Desglose del trabajo en tareas

- | | |
|---|---------|
| 1. Planificación | (55 hs) |
| 1.1. Estudio del funcionamiento básico de un probador. | (25 hs) |
| 1.2. Elaboración del documento de Planificación del proyecto. | (30 hs) |
| 2. Pruebas preliminares | (80 hs) |

2.1. Desarrollo del prototipo de firmware.	(25 hs)
2.2. Pruebas de interacción con el hardware.	(35 hs)
2.3. Pruebas de comunicación con el servidor.	(20 hs)
3. Diseño de firmware	(100 hs)
3.1. Diseño de la arquitectura general y el flujo de datos.	(35 hs)
3.2. Diseño de las tareas del RTOS y los mecanismos de comunicación entre ellas.	(35 hs)
3.3. Configuración del entorno de desarrollo.	(15 hs)
3.4. Estudio de las herramientas de software disponibles para la plataforma de desarrollo elegida.	(15 hs)
4. Implementación de firmware	(229 hs)
4.1. Desarrollo del driver para los ADC.	(25 hs)
4.2. Desarrollo del módulo de adquisición.	(32 hs)
4.3. Desarrollo del driver ethernet.	(40 hs)
4.4. Desarrollo del módulo de comunicación.	(40 hs)
4.5. Desarrollo del módulo de almacenamiento SD.	(32 hs)
4.6. Desarrollo del módulo de detección de estados.	(20 hs)
4.7. Desarrollo del módulo de control principal.	(40 hs)
5. Verificación y Validación	(163 hs)
5.1. Pruebas individuales de los módulos de firmware.	(30 hs)
5.2. Ensayos de integración del firmware.	(20 hs)
5.3. Pruebas de integración con el hardware.	(20 hs)
5.4. Ensayos funcionales.	(40 hs)
5.5. Análisis de los efectos de los errores de software.	(20 hs)
5.6. Demostración formal.	(8 hs)
5.7. Elaboración del informe de validación.	(25 hs)
6. Documentación	(60 hs)
6.1. Elaboración de la Memoria del trabajo.	(40 hs)
6.2. Producción de la presentación final.	(20 hs)

Cantidad total de horas: (687 hs)

7. Diagrama de Activity On Node

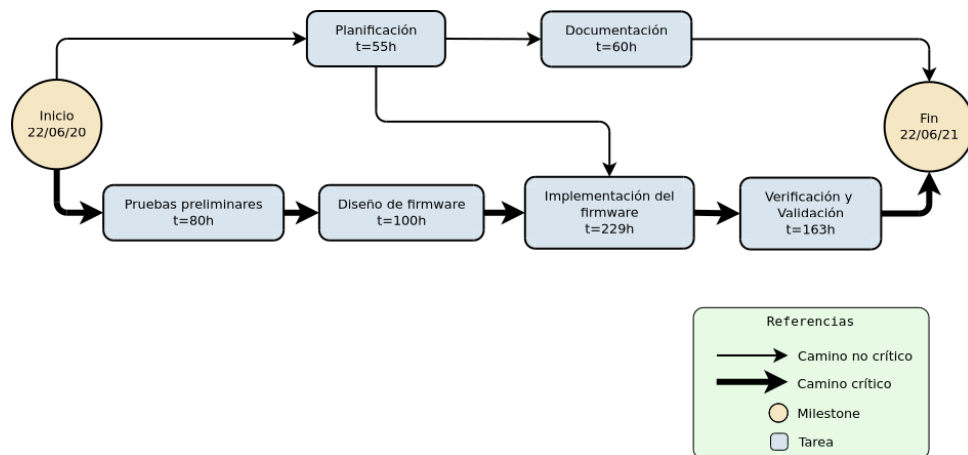


Figura 2: Diagrama en *Activity on Node*

8. Diagrama de Gantt

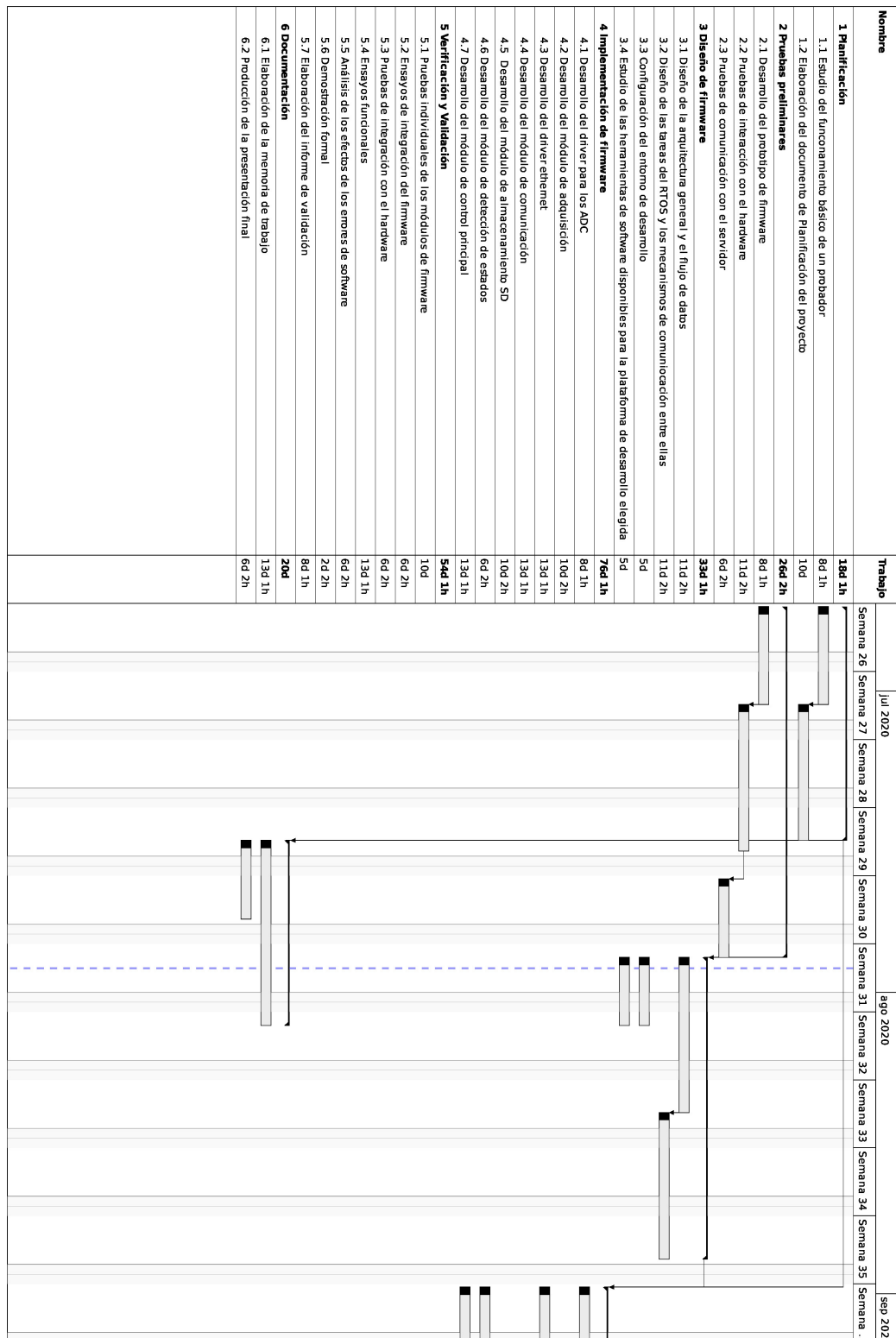


Figura 3: Diagrama de gantt

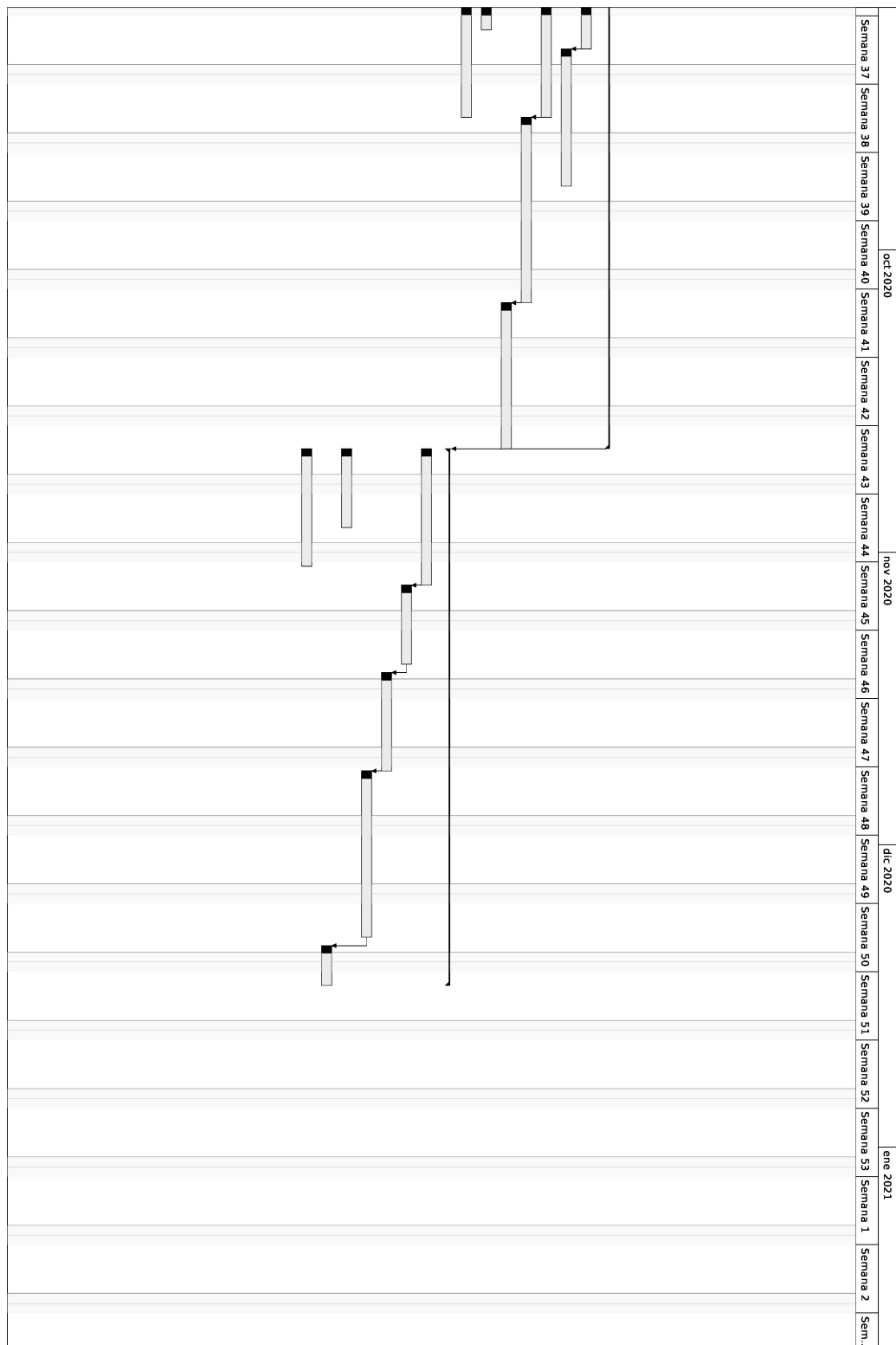


Figura 4: Diagrama de gantt (continuación)

9. Matriz de uso de recursos de materiales

WBS	Nombre de la tarea	Recursos requeridos (horas)	
		PC	Probador+CIAA
1.1	Estudio del funcionamiento básico de un probador	25	10
1.2	Elaboración del documento de planificación del proyecto	30	
2.1	Desarrollo del prototipo de firmware	25	10
2.2	Pruebas de interacción con el hardware	35	10
2.3	Pruebas de comunicación con el servidor	20	
3.1	Diseño de la arquitectura general y el flujo de datos	35	
3.2	Diseño de las tareas del RTOS y los mecanismos de comunicación entre ellas	35	
3.3	Configuración del entorno de desarrollo	15	
3.4	Estudio de las herramientas de software disponible para la plataforma de desarrollo elegida	15	
4.1	Desarrollo del driver para los ADC	25	15
4.2	Desarrollo del módulo de adquisición	32	5
4.3	Desarrollo del driver ethernet	40	5
4.4	Desarrollo del módulo de comunicación	40	
4.5	Desarrollo del módulo de almacenamiento SD	32	5
4.6	Desarrollo del módulo de detección de estados	20	5
4.7	Desarrollo del módulo de control principal	40	
5.1	Pruebas individuales de los módulos de firmware	30	10
5.2	Ensayos de integración del firmware	20	20
5.3	Pruebas de integración con el hardware	20	20
5.4	Ensayos funcionales	40	40
5.5	Análisis de los efectos de los errores de software	15	
5.6	Demostración formal	8	8
5.7	Elaboración del informe de validación	25	
6.1	Elaboración de la memoria de trabajo	40	
6.2	Producción de la presentación final	20	

10. Presupuesto detallado del proyecto

COSTOS DIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
CIAA-NXP	1	\$ 25.600	\$ 25.600
Poncho Probador de relés	1	\$ 12.693	\$ 12.693
Transformador 220VAC/12VAC	1	\$ 918	\$ 918
Mano de obra	687 hs	\$ 850	\$ 583.950
SUBTOTAL			\$ 622.243
COSTOS INDIRECTOS			
Descripción	Cantidad	Valor unitario	Valor total
30 % de los costos directos	N/A	N/A	\$ 186.673
SUBTOTAL			
TOTAL			

11. Matriz de asignación de responsabilidades

Código WBS	Nombre de la tarea	Nombres y roles del proyecto				Cliente Martín Harris
		Responsable Ing. Gaspar Santamarina	Orientador Ing. Adrián Laiuppa	Colaborador Ing. Nicolás Locatelli	Colaborador Ing. Gustavo Ramoscelli	
1.1	Estudio del funcionamiento básico de un probador	P	C			
1.2	Elaboración del documento de planificación del proyecto	P	A			I
2.1	Desarrollo del prototipo de firmware	P	C			
2.2	Pruebas de interacción con el hardware	P	C			
2.3	Pruebas de comunicación con el servidor	P		C	C	
3.1	Diseño de la arquitectura general y el flujo de datos	P				
3.2	Diseño de las tareas del RTOS y los mecanismos de comunicación entre ellas	P				
3.3	Configuración del entorno de desarrollo	P				
3.4	Estudio de las herramientas de software disponible para la plataforma de desarrollo elegida	P				
4.1	Desarrollo del driver para los ADC	P	C			
4.2	Desarrollo del módulo de adquisición	P	I			
4.3	Desarrollo del driver ethernet	P				
4.4	Desarrollo del módulo de comunicación	P		I	I	
4.5	Desarrollo del módulo de almacenamiento SD	P	I			
4.6	Desarrollo del módulo de detección de estados	P	C			
4.7	Desarrollo del módulo de control principal	P	I			
5.1	Pruebas individuales de los módulos de firmware	P				
5.2	Ensayos de integración del firmware	P	I			
5.3	Pruebas de integración con el hardware	P	A			
5.4	Ensayos funcionales	P	A	A	A	A
5.5	Análisis de los efectos de los errores de software	P	C			
5.6	Demostración formal	P	A	I	I	A
5.7	Elaboración del informe de validación	P	A			
6.1	Elaboración de la memoria de trabajo	P	A			
6.2	Producción de la presentación final	P	A			

Referencias:

- P = Responsabilidad Primaria
- S = Responsabilidad Secundaria
- A = Aprobación
- I = Informado
- C = Consultado

12. Gestión de riesgos

12.1. Identificación de riesgos y estimación de sus consecuencias

Riesgo 1: avería de la CIAA o del poncho

- *Severidad: 8* - Dificultaría la implementación del firmware y sería necesario reponer el hardware para continuar con el desarrollo.
- *Ocurrencia: 2* - Solo se interactuará con el hardware a través del firmware. Es poco probable que se dañe el equipamiento.

Riesgo 2: Errores en el diseño de hardware del probador

- *Severidad: 6* - Dificultaría las pruebas para validar el correcto funcionamiento del firmware.
- *Ocurrencia: 2* - El hardware fue probado exhaustivamente, por lo que es poco probable que surjan nuevos errores de diseño.

Riesgo 3: Demoras en los tiempos estipulados para cada tarea

- *Severidad: 6* - Retrasaría el desarrollo del proyecto. Sin embargo se cuenta con márgenes de tiempo, por lo que no sería un problema, siempre y cuando las demoras no sean excesivas.
- *Ocurrencia: 8* - Dada la poca experiencia en planificación de proyectos, existe una alta probabilidad de que los tiempos no hayan sido correctamente estimados.

Riesgo 4: Dificultades en el desarrollo del driver de red

- *Severidad: 10* - Se estima que el driver de red junto con el stack TCP/IP es el módulo mas complejo en todo el diseño. No lograr implementar dicho módulo haría imposible conseguir los objetivos planteados.
- *Ocurrencia: 2* - Si bien el manejo del hardware de red se presenta como una tarea compleja, se cuenta con el soporte de la comunidad de desarrolladores en torno a la CIAA.

Riesgo 5: No lograr que el firmware cumpla con los requisitos de rendimiento

- *Severidad: 8* - En caso de que no se logre realizar la lectura de todas las señales analógicas y el envío del paquete al servidor entre ciclo y ciclo, será necesario recurrir a un microprocesador de mayores prestaciones.
- *Ocurrencia: 2* - Se estima que el microprocesador de la CIAA-NXP cuenta con capacidad de sobra para realizar las tareas mencionadas con la frecuencia requerida.

12.2. Tabla de gestión de riesgos

Riesgos	Detalle	S	O	RPN	S*	O*	RPN*
1	Avería de la CIAA o el poncho	8	2	16			
2	Errores en el diseño de hardware del probador	6	2	12			
3	Demoras en los tiempos estipulados para cada tarea	6	8	48	6	4	24
4	Dificultades en el desarrollo del driver de red	10	2	20			
5	No lograr que el firmware cumpla con los requisitos de rendimiento	8	2	16			

Criterio adoptado: Se tomarán medidas de mitigación en los riesgos cuyos números de RPN sean mayores a **40**.

Nota: los valores marcados con (*) en la tabla corresponden luego de haber aplicado la mitigación.

12.3. Plan de mitigación de riesgos

Riesgo 3: Demoras en los tiempos estipulados para cada tarea

- *Plan de mitigación:* Reveer el plan de trabajo y aumentar las horas dedicadas al proyecto.
- *Severidad: 6* - Retrasaría el desarrollo del proyecto. Sin embargo se cuenta con márgenes de tiempo, por lo que no sería un problema, siempre y cuando las demoras no sean excesivas. Se mantiene la severidad.
- *Ocurrencia: 4* - Si se aumentan las horas de trabajo disminuye la probabilidad de tener demoras.

13. Gestión de la calidad

1. Interfaces Externas

Verificación: Durante la fase de Pruebas preliminares, se desarrollará un prototipo de firmware que haga uso de todos los periféricos mencionados. De esta forma se verificará el correcto funcionamiento de los mismos.

Validación: Se mostrará este firmware funcionando al Ing. Adrián Laiuppa, quién validará o no el funcionamiento de los periféricos.

2.1 El firmware debe ser capaz de enviar notificaciones de alerta al servidor remoto, generadas por el disparo de una señal de alerta del hardware.

2.2 El firmware debe ser capaz de enviar notificaciones de alerta al servidor remoto, generadas por sobretensión en cualquiera de las entradas analógicas.

2.3 El firmware debe ser capaz de interpretar los siguientes comandos enviados desde el servidor:

- *Aplicar configuración de sistema.*
- *Definir los parámetros asociados a cada tipo de ensayo.*
- *Disparar ensayos.*
- *Parar ensayos.*

2.5 Junto con cada paquete de datos generado a partir de una prueba, se debe anexar un ID que identifique el tipo de prueba.

2.6 Junto con cada paquete de datos se debe anexar una marca temporal y un ID que identifique a la placa dentro de las tres placas que conforman el sistema.

2.9 El firmware debe permitir configurar los siguientes parámetros:

- *Hora del sistema.*
- *URL y puerto del broker MQTT.*

Verificación: Se acordará con los Ingenieros Gustavo Ramoscelli y Nicolás Locatelli, encargados del desarrollo de la aplicación servidor, las pruebas a realizar para el envío de mensajes de alerta, así como la recepción por parte del firmware de los comandos mencionados.

Validación: Una vez realizada la verificación y con el firmware completamente implementado, en el marco de los Ensayos funcionales, se validará la comunicación con el servidor.

2.4 El firmware deberá ser capaz de ejecutar los siguientes ensayos: (ver requerimientos)

2.8 El firmware debe permitir realizar distintos tipos de ensayos en forma intercalada.

Verificación: Se conectará un relé de baja intensidad en los terminales de prueba y se corroborará que las secuencias de ensayo se desarrollen correctamente.

Validación: Se conectará el hardware en su totalidad y se lo hará funcionar durante un tiempo prolongada, verificando en todo momento el correcto funcionamiento del mismo.

2.7 Cada paquete de datos debe ser almacenado en una tarjeta MicroSD a modo de copia de seguridad.

Verificación: Al finalizar las pruebas de verificación del requerimiento 2.4, se corroborará que los archivos almacenados en la SD y los datos almacenados en el servidor coincidan.

Validación: Se llevará a cabo el mismo procedimiento que para la verificación pero con la totalidad del hardware funcionando.

3.1 Para el ensayo de rampa se deben permitir hasta 25 ciclos.

*3.2 Para el ensayo de vida útil mecánica **sin carga** se deben permitir hasta $10 \cdot 10^6$ ciclos.*

*3.3 Para el ensayo de vida útil mecánica **con carga** se deben permitir hasta $2 \cdot 10^6$ ciclos.*

3.4 Para el ensayo de vida útil mecánica el periodo máximo entre ciclos es de 3 segundos.

Verificación: Se verificará por revisión de diseño que el firmware sea capaz de manejar la cantidad máxima estipulada de ciclos. Para ello se tendrá en cuenta el tamaño de las variables utilizadas, además del diseño de los algoritmos encargados del control de los ensayos.

Validación: -

5.1 El firmware debe ser capaz de recuperarse ante una falla que provoque el reinicio del microcontrolador, en un tiempo menor a los 3 segundos y retomar los ensayos en curso.

Verificación: Se corroborará que el tiempo de inicialización del firmware sea menor al estipulado.

Validación: Se producirán fallas intencionales que provoquen el reinicio del microcontrolador y se medirá el tiempo necesario para dicho reinicio.

5.2 El firmware debe hacer uso de cifrado TLS para la capa de transporte.

5.3 El firmware debe hacer uso de Usuario/Contraseña para autenticarse con el broker MQTT.

Verificación: Una vez configurado el servidor para que solo acepte conexiones que utilicen cifrado TLS y el broker para que utilice autenticación por Usuario/Contraseña, se intentará establecer comunicación con el mismo.

Validación: Se utilizará la herramienta de software Wireshark para corroborar que los paquetes enviados desde el firmware estén cifrados.

6.1 El desarrollo debe ser documentado utilizando la herramienta Doxygen.

Verificación: Se irán generando documentaciones intermedias a lo largo del desarrollo.

Validación: Al finalizar el desarrollo se generará la documentación en alguno de los formatos soportados por la herramienta.

14. Comunicación del proyecto

A continuación se detalla el plan de comunicación del proyecto:

PLAN DE COMUNICACIÓN DEL PROYECTO					
¿Qué comunicar?	Audiencia	Propósito	Frecuencia	Método de comunicac.	Responsable
Plan de Proyecto	Profesores de GdP, Director y Codirector del proyecto	Informar sobre el plan y la metodología de trabajo	Una única vez	Correo electrónico	Ing. Gaspar Santamarina
Desarrollo del módulo de red	Encargados del desarrollo web	Informar a los interesados sobre la disponibilidad del módulo de red para realizar pruebas	Una única vez	Correo electrónico	Ing. Gaspar Santamarina
Avances en las interfaces con el hardware	Ing. Adrián Laiuppa	Verificar el correcto funcionamiento del hardware	Semanal	Correo electrónico	Ing. Gaspar Santamarina
Presentación final	Director y codirector del proyecto y director de la Especialización	Informar sobre la finalización del proyecto	Una única vez	Correo electrónico	Ing. Gaspar Santamarina

15. Gestión de Compras

No se realizarán compras durante el desarrollo del proyecto.

16. Seguimiento y control

SEGUIMIENTO DE AVANCE				
Tarea del WBS	Indicador de avance	Frecuencia de reporte	Persona a ser informada	Método de comunic.
1.1	Cantidad de documentos leídos	Semanal	Ing. Adrián Laiuppa	Correo electrónico
1.2	Cantidad de secciones redactadas	Bimensual	Ing. Adrián Laiuppa	Correo electrónico
2.1	Cantidad de módulos implementados	Bimensual	Ing. Adrián Laiuppa	Correo electrónico
2.2	Interfaces de hardware testeadas	Semanal	Ing. Adrián Laiuppa	Correo electrónico
2.3	Cantidad archivos JSON implementados	Semanal	Ing. Gustavo Ramoscelli, Ing. Nicolás Locatelli	Grupo de chat
3.1	Cantidad de módulos diseñados	Semanal	Mg. Ing. Martin Menendez	Correo electrónico
3.2	Cantidad de tareas diseñadas	Semanal	Mg. Ing. Martin Menendez	Correo electrónico
3.3	-	-	-	-
3.4	-	-	-	-
4.1	Cantidad de líneas de código escritas	Semanal	Mg. Ing. Martin Menendez, Ing. Adrián Laiuppa	Correo electrónico
4.2	Cantidad de líneas de código escritas	Semanal	Mg. Ing. Martin Menendez, Ing. Adrián Laiuppa	Correo electrónico
4.3	Cantidad de líneas de código escritas	Semanal	Mg. Ing. Martin Menendez	Correo electrónico
4.4	Cantidad de líneas de código escritas	Semanal	Mg. Ing. Martin Menendez	Correo electrónico

SEGUIMIENTO DE AVANCE				
Tarea del WBS	Indicador de avance	Frecuencia de reporte	Persona a ser informada	Método de comunic.
4.5	Cantidad de líneas de código escritas	Semanal	Mg. Ing. Martin Menendez	Correo electrónico
4.6	Cantidad de líneas de código escritas	Semanal	Ing. Adrián Laiuppa	Correo electrónico
4.7	Cantidad de líneas de código escritas	Semanal	Mg. Ing. Martin Menendez	Correo electrónico
5.1	Cantidad de módulos probados	Semanal	Mg. Ing. Martin Menendez	Correo electrónico
5.2	Cantidad de módulos integrados	Semanal	Mg. Ing. Martin Menendez	Correo electrónico
5.3	Cantidad de bloques de hardware probados	Semanal	Ing. Adrián Laiuppa	Correo electrónico
5.4	Cantidad de ensayos realizadas	Semanal	Ing. Adrián Laiuppa, Martín Harris	Correo electrónico
5.5	Cantidad de ensayos realizados	Semanal	Mg. Ing. Martin Menendez	Correo electrónico
5.6	-	-	-	-
5.7	Cantidad de páginas escritas	Semanal	Ing. Adrián Laiuppa	Correo electrónico
6.1	Cantidad de páginas escritas	Bimensual	Ing. Adrián Laiuppa	Correo electrónico
6.2	Cantidad de slides producidos	Semanal	Ing. Adrián Laiuppa	Correo electrónico

El responsable de llevar a cabo el seguimiento de todas las tareas será el Ing. Gaspar Santamarina.

17. Procesos de cierre

A continuación se detallan las pautas para llevar a cabo la reunión final de evaluación del proyecto:

- Se contrastará el cronograma estimativo presentado en secciones precedentes con el cronograma real, actualizado a lo largo del desarrollo. Al final se realizará un informe que de cuenta de las principales divergencias entre el plan y la realidad.

Responsable: Ing. Gaspar Santamarina

- A partir del informe detallado en el punto anterior, se analizarán cuáles son las tareas y actividades que tuvieron mayor impacto en las eventuales demoras y cómo se podría haber minimizado dicho impacto.

Responsable: Ing. Gaspar Santamarina

- A modo de agradecimiento a los involucrados, además de mencionarlos como partícipes claves en el desarrollo durante la presentación final del proyecto, se los invitará a formar parte de la misma.

Responsable: Ing. Gaspar Santamarina