



Pontificia Universidad
JAVERIANA
Cali

**Facultad de Ingeniería
y Ciencias**
Ingeniería Electrónica

MONOGRAFÍA DE TRABAJO DE GRADO

Sistema de Adquisición y Visualización de Temperatura para Banco de Transferencia de Calor

Santiago Garcia Barreto
Santiago Gomez paz

Director
Ing Simón Alfredo Plata Eastman

27 de julio de 2026

Nota de Aceptación

Aprobado por el Comité de Trabajo de Grado
en cumplimiento de los requisitos exigidos por la
Pontificia Universidad Javeriana para optar al
título de Ingeniero Electrónico.

Dr. Mateo Lõpez Victoria
Decano Facultad de Ingeniería y Ciencias

Dr. Luis Eduardo Tobon Llano
Director Carrera Ingeniería Electrónica



Ing Simón Alfredo Plata Eastman
Director Trabajo de Grado

Dr. Alexander Martinez Alvarez
Jurado 1

Ing Bleider Johan Arizala Cobo
Jurado 2

Santiago de Cali, 27 de julio de 2026

Señores
Pontificia Universidad Javeriana – Cali
Dr. Mateo López Victoria
Decano
Facultad de Ingeniería y Ciencias
Ciudad

Cordial Saludo.

Por medio de la presente nos permitimos presentarle el Trabajo de Grado titulado “Sistema de Adquisición y Visualización de Temperatura para Banco de Transferencia de Calor”.

Esperamos que este trabajo reúna todos los requisitos académicos, cumpla el propósito para el cual fue creado y sirva de apoyo para futuros proyectos relacionados con la materia.

Atentamente,



Santiago Garcia Barreto



Santiago Gomez paz

Santiago de Cali, 27 de julio de 2026

Señores

Pontificia Universidad Javeriana – Cali

Dr. Mateo López Victoria

Decano

Facultad de Ingeniería y Ciencias

Ciudad

Cordial Saludo.

Certifico que el presente Trabajo de Grado titulado “Sistema de Adquisición y Visualización de Temperatura para Banco de Transferencia de Calor”, realizado por Santiago Garcia Barreto y Santiago Gomez paz, estudiantes de Ingeniería Electrónica, se encuentra terminado y puede ser presentado para su sustentación.

Atentamente,

A handwritten signature in black ink, reading "Ing Simón Alfredo Plata Eastman". The signature is written in a cursive style and is positioned above a horizontal line.

Ing Simón Alfredo Plata Eastman
Director Trabajo de Grado

Agradecimientos

Agradecemos a Simón Alfredo Plata Eastman, por su acompañamiento, orientación y retroalimentación durante el desarrollo de este trabajo de grado. Sus observaciones fueron fundamentales para estructurar el proyecto, fortalecer la metodología y mejorar la calidad del documento.

De manera especial, agradecemos a la Pontificia Universidad Javeriana Cali, al programa de Ingeniería Electrónica y al CAP, por el acceso a recursos de laboratorio, equipos y espacios de trabajo necesarios para la implementación y validación del sistema. Finalmente, agradecemos a nuestras familias y a las personas cercanas que brindaron apoyo constante durante este proceso.

Glosario

Símbolos

T Temperatura [°C].

R Resistencia eléctrica [Ω].

f Frecuencia [Hz].

t Tiempo [s].

Acrónimos y Abreviaturas

BOM Lista de materiales (*Bill of Materials*).

CRC Verificación por redundancia cíclica (*Cyclic Redundancy Check*).

CSV Formato de datos separado por comas (*Comma-Separated Values*).

DAQ Adquisición de datos (*Data Acquisition*).

GPIO Entrada/salida de propósito general (*General Purpose Input/Output*).

IDE Entorno de desarrollo integrado (*Integrated Development Environment*).

JSON Formato de intercambio de datos (*JavaScript Object Notation*).

PCB Placa de circuito impreso (*Printed Circuit Board*).

RS-485 Estándar de comunicación serial diferencial multipunto.

UART Transmisor/Receptor Asíncrono Universal (*Universal Asynchronous Receiver/Transmitter*).

USB Bus Serial Universal (*Universal Serial Bus*).

Términos

Backplane Base de interconexión donde se conectan módulos (tarjetas) para compartir alimentación y señales comunes.

Bus 1-Wire Protocolo de comunicación digital que permite conectar uno o varios dispositivos usando una línea de datos compartida.

DS18B20 Sensor digital de temperatura que utiliza el protocolo 1-Wire y posee un identificador único (ROM) para cada dispositivo.

Firmware Software embebido encargado de controlar el hardware del sistema, incluyendo la lectura de sensores y la comunicación.

Pull-up Resistencia que fija el nivel lógico alto en reposo; en el protocolo 1-Wire es necesaria para mantener el estado *idle* del bus.

Trama Unidad estructurada de datos transmitida por un enlace de comunicación (por ejemplo, cabecera, longitud, carga útil y verificación).

Resumen

Este trabajo presenta el diseño e implementación de un sistema embebido para la adquisición y visualización de temperatura en un banco de transferencia de calor, orientado al apoyo de prácticas experimentales en contextos académicos. El objetivo general consiste en desarrollar una solución capaz de leer temperatura desde sensores distribuidos, transmitir los datos hacia un computador y visualizar la información en tiempo real para su análisis y almacenamiento.

Como objetivos específicos se plantearon seleccionar una tecnología de sensado adecuada, diseñar una arquitectura modular escalable, implementar un prototipo funcional (hardware y firmware) y validar la comunicación y la visualización de datos. El sistema propuesto integra sensores digitales de temperatura conectados en un bus común, un microcontrolador para la adquisición y procesamiento, y un enlace de comunicación para el envío de información hacia una aplicación de visualización. El desarrollo se realizó mediante un proceso iterativo, con prototipado progresivo del firmware, diseño electrónico de tarjetas modulares y construcción de una estructura de interconexión tipo backplane.

Los resultados principales evidenciaron la viabilidad de la lectura simultánea de múltiples sensores, la transmisión estable de datos hacia el computador y la generación de registros exportables para análisis posterior. En conjunto, el trabajo entrega una base técnica replicable para instrumentación didáctica de bancos térmicos, con potencial de expansión y mejoras en futuras iteraciones.

Palabras Clave: sistemas embebidos, adquisición de datos, medición de temperatura, transferencia de calor, visualización en tiempo real, instrumentación educativa.

Abstract

This work presents the design and implementation of an embedded system for temperature data acquisition and real-time visualization in a heat transfer test bench, aimed at supporting experimental activities in academic environments. The main objective was to develop a low-cost solution capable of reading temperature from distributed sensors, transmitting the data to a computer, and displaying the information in real time for analysis and storage.

The specific objectives included selecting an appropriate sensing technology, designing a scalable modular architecture, implementing a functional prototype (hardware and firmware), and validating both communication and data visualization. The proposed system integrates digital temperature sensors connected on a shared bus, a microcontroller for data acquisition and processing, and a communication link to deliver the information to a visualization application. The development followed an iterative process, including progressive firmware prototyping, electronic design of modular PCBs, and the construction of a backplane-like interconnection structure.

The main results demonstrated the feasibility of simultaneously reading multiple sensors, achieving stable data transmission to the computer, and generating exportable logs for further analysis. Overall, this work provides a replicable technical basis for didactic instrumentation of thermal test benches, with potential scalability and improvements in future iterations.

Keywords: embedded systems; data acquisition; temperature measurement; heat transfer; real-time visualization; educational instrumentation.

Índice general

1. Introducción	1
2. Planteamiento del Problema	4
3. Justificación	6
4. Objetivos	8
4.1. Objetivo General	8
4.2. Objetivos Específicos	8
5. Marco de Referencia	9
5.1. Áreas Temáticas	9
5.2. Marco Teórico	9
5.2.1. Transferencia de calor y medición experimental	10
5.2.2. Sistemas embebidos y adquisición de datos	10
5.2.3. Microcontroladores	10
5.2.4. Sensores de temperatura	11
5.2.5. Comunicación industrial RS485 y transceptor THVD1400	12
5.2.6. Arquitecturas modulares tipo VMEbus y portadoras 3U	13
5.2.7. Visualización y simulación térmica	14
5.2.8. Normatividad y limitaciones en contexto colombiano	14
5.3. Trabajos Relacionados	16
6. Metodología	20
6.1. Enfoque metodológico del proyecto	20
6.1.1. Tipo de investigación y enfoque	20
6.1.2. Aplicación de Design Thinking	21
6.2. Alcance, requerimientos y criterios de diseño	22
6.2.1. Alcance de la fase de diseño e implementación	23
6.2.2. Requerimientos del sistema	23
6.2.3. Criterios de Diseño	25
6.3. Selección y justificación de componentes del sistema	26
6.3.1. Selección del sensor	26
6.3.2. Selección del microcontrolador	28
6.3.3. Selección del sistema de comunicación	30
6.4. Arquitectura modular del sistema	32
6.4.1. Descripción general de la arquitectura	32
6.4.2. Tarjeta de alimentación (POWER)	33

6.4.3.	Tarjeta de adquisición/control (DAQ)	36
6.4.4.	Tarjeta de comunicación (COM)	37
6.4.5.	Tarjeta de BACKPLANE	39
6.5.	Desarrollo del firmware	41
6.5.1.	Lógica general del firmware	42
6.5.2.	Implementación del protocolo 1-Wire con DS18B20	44
6.5.3.	Temporización del bus 1-Wire	46
6.5.4.	Direccionamiento de sensores	48
6.5.5.	Verificación CRC	48
6.5.6.	Acceso por registros en ATmega328PB	50
6.5.7.	Uso de la interfaz USART para transmisión y diagnóstico	50
6.5.8.	Lógica de adquisición y envío	51
6.6.	Comunicación y estructura de tramas	52
6.6.1.	Empaquetado de los datos	52
6.6.2.	Transmisión de la trama	53
6.6.3.	Validación de la trama recibida	54
6.6.4.	Organización del intercambio entre módulos	54
6.7.	Diseño de hardware y evolución de PCBs	56
6.7.1.	Diseño esquemático y PCB	56
6.7.2.	Primera iteración de hardware	56
6.7.3.	Hallazgos y correcciones del rediseño	57
6.7.4.	Tarjeta DAQ/DAT	57
6.7.5.	Tarjeta COM	62
6.7.6.	Tarjeta POWER	66
6.7.7.	Backplane	69
6.7.8.	Integración física del sensado	72
6.7.9.	Fabricación y ensamble	73
6.7.10.	Diseño e implementación de la estructura mecánica del sistema	75
6.8.	Plataforma de visualización, registro y exportación	78
6.9.	Integración y validación del sistema	79
7.	Resultados y Discusión	81
7.1.	Plan de pruebas	81
7.2.	Resultados del sensado con DS18B20	82
7.2.1.	Detección de sensores y direccionamiento	82
7.2.2.	Lectura multisensor y resolución observada	83
7.2.3.	Estabilidad y registro en CSV	83
7.3.	Resultados de comunicación	85
7.3.1.	Diagnóstico por UART	85
7.3.2.	Solicitud y respuesta sobre RS-485	86
7.3.3.	Indicadores LED de estado	87

Índice general	11
7.4. Resultados de visualización en tiempo real	87
7.5. Hallazgos, limitaciones y trabajo futuro	88
7.5.1. Hallazgos principales	88
7.5.2. Limitaciones	90
8. Conclusiones	92
8.1. Conclusiones	92
9. Recomendaciones	94
9.1. Recomendaciones	94
Bibliografía	96

Índice de figuras

1.1. Estado de avance del banco de transferencia de calor desarrollado por el grupo asociado.	2
1.2. Diseño CAD del banco de transferencia de calor.	2
5.1. Microcontroladores	11
5.2. Sensores de temperatura	12
5.3. Red RS-485 típica.	13
5.4. Diagrama de la arquitectura	14
6.1. Estructura física modular del sistema (backplane y conectores de borde).	33
6.2. Diagrama de bloques de la tarjeta POWER	35
6.3. Diagrama de bloques de la tarjeta DAQ	37
6.4. Diagrama de bloques de la tarjeta COM	39
6.5. Arquitectura lógica de adquisición	41
6.6. Diagrama de flujo resumido del firmware para lectura DS18B20 (1-Wire).	43
6.7. Diagrama de comandos ROM en 1-Wire	45
6.8. Máquina de estados para la temporización	47
6.9. Diagramas de temporización 1-Wire	47
6.10. Rutina de verificación CRC del DS18B20	49
6.11. Diagrama de bloques del periférico USART (transmisor/receptor)[9]	51
6.12. Secuencia temporal del intercambio de datos entre los módulos de adquisición y comunicación	55
6.13. Esquemático de la tarjeta DAT	58
6.14. Diseño PCB de la tarjeta DAT	58
6.15. Vista 3D de la tarjeta DAT	59
6.16. Esquemático de la tarjeta COM	62
6.17. Diseño PCB tarjeta COM	63
6.18. Vista 3D de la tarjeta COM	63
6.19. Esquemático de la tarjeta POWER	66
6.20. Diseño PCB de la tarjeta POWER	67
6.21. Vista 3D de la tarjeta POWER	68
6.22. Esquemático del Backplane	70
6.23. Diseño PCB del Backplane	70
6.24. Vista 3D del Backplane	70
6.25. Esquemático del Backplane DC	71
6.26. Diseño del PCB del Backplane DC	71
6.27. Vista 3D del Backplane DC	71
6.28. Esquemático de la placa de integración física del sensado	72

6.29. Diseño PCB de la placa de integración física del sensado	72
6.30. Vista 3D de la placa de integración física del sensado	72
6.31. Tarjetas soldadas del Backplane	73
6.32. Tarjeta soldada de Power	74
6.33. Tarjeta soldada de COM	74
6.34. Tarjeta soldada de DAT	74
6.35. Detalle del soporte interno tipo canal	77
6.36. Integración interna del backplane y los soportes	77
6.37. Avance del montaje del chasis	77
6.38. Chasis mecánico ensamblado	77
6.39. Diagrama de diseño de la plataforma de visualización, registro y exportación (Python).	78
7.1. Sistema modular ensamblado y en funcionamiento.	81
7.2. Evidencia por MPLAB: <i>Presence OK</i> , ROM encontradas y lectura multisensor (S0–S6).	83
7.3. Evidencia de exportación a CSV.	84
7.4. Estructura general de la trama de respuesta implementada en el sistema.	86
7.5. Registro de temperatura y retorno al equilibrio en aproximadamente 180 s.	88
7.6. Registro extendido de temperatura y retorno al equilibrio en aproximadamente 280 s.	88

Índice de cuadros

5.1. Comparativa de trabajos relacionados	19
6.1. Trazabilidad entre las etapas del Design Thinking y el desarrollo del proyecto	22
6.2. Requerimientos funcionales	23
6.3. Requerimientos no funcionales	24
6.4. Requerimientos del sensor de temperatura	24
6.5. Requerimientos del sistema de adquisición y comunicación	25
6.6. Comparación de sensores de temperatura	27
6.7. Matriz de decisión para selección del sensor	28
6.8. Parámetros de las plataformas/microcontroladores evaluados	29
6.9. Matriz de decisión para selección del microcontrolador/plataforma	30
6.10. Comparación de alternativas de comunicación evaluadas	32
6.11. Resumen funcional de la tarjeta POWER	35
6.12. Resumen funcional de la tarjeta BACKPLANE	40
6.13. Componentes principales de la tarjeta DAQ/DAT	60
6.14. Conectores e interfaces principales de la tarjeta DAQ/DAT	61
6.15. Pinout funcional del conector principal de la tarjeta DAQ/DAT	61
6.16. LEDs indicadores de la tarjeta DAQ/DAT	61
6.17. Componentes principales de la tarjeta COM	64
6.18. Conectores e interfaces principales de la tarjeta COM	65
6.19. Pinout funcional de las interfaces principales de la tarjeta COM	65
6.20. LEDs indicadores de la tarjeta COM	66
6.21. Componentes principales de la tarjeta POWER	68
6.22. Conectores e interfaces principales de la tarjeta POWER	69
6.23. Pinout funcional de la tarjeta POWER	69
6.24. Componentes principales del backplane	70
6.25. Componentes principales del backplane	71
6.26. Materiales y especificaciones utilizados en la construcción del chasis.	75
7.1. Plan de pruebas para el sistema de adquisición y comunicación.	82
7.2. Muestra de lecturas multisensor.	83
7.3. Métricas por sensor extraídas de <code>temperaturas.csv</code>	85
7.4. Campos de la trama de respuesta RS-485.	86
7.5. Resumen de costos del sistema desarrollado por categoría.	88
7.6. Comparación del sistema desarrollado con soluciones comerciales.	89

Introducción

La comprensión de los fenómenos térmicos constituye un componente fundamental en la formación en ingeniería, debido a que permite analizar y diseñar sistemas en los que intervienen procesos de generación, transferencia y disipación de energía. En los cursos de transferencia de calor, el aprendizaje se fortalece cuando la teoría se contrasta con mediciones experimentales que permiten evidenciar mecanismos como conducción, convección y radiación, así como observar la evolución temporal de la temperatura bajo condiciones controladas [1].

Los bancos de transferencia de calor son herramientas fundamentales en la formación experimental de ingenieros. Permiten reproducir y estudiar de forma controlada los mecanismos de conducción, convección y radiación, y facilitan el contraste entre los modelos teóricos y las mediciones reales obtenidas en el laboratorio. Estudios recientes respaldan esta relevancia: Zárate-Navarro et al. (2024) reportan resultados positivos en el aprendizaje de transferencia de calor mediante actividades que integran simulación y validación experimental en cursos de ingeniería [2], mientras que Mahanta et al. (2023) documentan el diseño y la efectividad pedagógica de dispositivos experimentales para la enseñanza de transferencia de calor a nivel universitario. Estas evidencias coinciden en que la experimentación directa sobre fenómenos térmicos fortalece competencias analíticas y científicas que no se desarrollan con la misma profundidad mediante aproximaciones exclusivamente teóricas [3].

Es importante precisar que el presente trabajo hace parte de un desarrollo conjunto en el marco del programa de Ingeniería Electrónica de la Pontificia Universidad Javeriana Cali. El banco de transferencia de calor sobre el cual se instrumenta el sistema es desarrollado en paralelo por un grupo de trabajo asociado, cuyo proyecto abarca el diseño y construcción de la estructura física del banco —incluyendo la base en perfil de aluminio, la columna de conducción axial con discos de material térmico, el módulo de convección forzada y los conectores físicos de sensado. El presente trabajo se circunscribe exclusivamente al sistema de instrumentación: la adquisición de la señal de temperatura desde los sensores DS18B20, su transmisión mediante RS-485, la visualización en tiempo real y el almacenamiento de datos. La Figura 1.1 muestra el estado de avance del banco físico desarrollado por el grupo asociado.

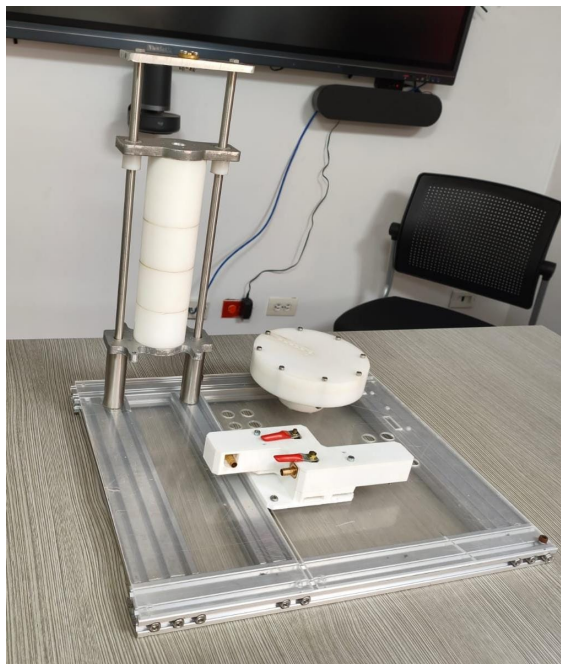


Figura 1.1: Estado de avance del banco de transferencia de calor desarrollado por el grupo asociado.

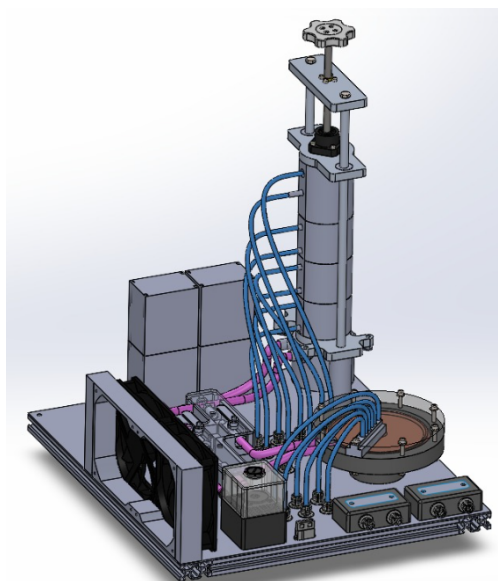


Figura 1.2: Diseño CAD del banco de transferencia de calor.

En muchos entornos académicos, especialmente en contextos con restricciones presupuestales, la disponibilidad de equipos de laboratorio para medición térmica en tiempo real suele ser limitada. Esta situación reduce la posibilidad de que los estudiantes registren datos confiables, analicen tran-

sitorios térmicos, validen modelos teóricos y comprendan con mayor profundidad el comportamiento real de un sistema. En este contexto, resulta pertinente el desarrollo de soluciones embebidas de bajo costo, configurables y replicables, que contribuyan a la modernización de bancos didácticos mediante procesos de adquisición, transmisión y visualización de datos [4][5][6].

A partir de esta necesidad, el presente trabajo plantea la instrumentación de un banco de transferencia de calor mediante un sistema capaz de: (i) adquirir datos de temperatura desde múltiples puntos de medición, (ii) transmitirlos de forma confiable a una plataforma externa y (iii) visualizarlos y almacenarlos en tiempo real para su análisis experimental y posterior procesamiento. Para ello, se propone una arquitectura modular orientada a facilitar el mantenimiento, la escalabilidad y la reutilización del sistema en futuros desarrollos académicos [7][8][9].

El objetivo general del proyecto consiste en diseñar e implementar un sistema embebido para la adquisición de datos de temperatura provenientes de sensores distribuidos en un banco de transferencia de calor, permitiendo su transmisión, visualización y análisis en tiempo real. En concordancia con este propósito, los objetivos específicos contemplan la selección de la tecnología de sensado más adecuada, el diseño de una arquitectura de adquisición y comunicación escalable, la implementación de un prototipo funcional con hardware y firmware dedicados, y la validación de la comunicación, la visualización y el almacenamiento de los datos adquiridos[10][11].

Desde el punto de vista tecnológico, la solución se basa en sensores digitales DS18B20, aprovechando su comunicación 1-Wire y su identificador único por dispositivo para facilitar la gestión de múltiples puntos de medición. La unidad de control se implementa con un microcontrolador ATmega328PB, seleccionado por la disponibilidad de periféricos y recursos apropiados para tareas de adquisición y comunicación. Adicionalmente, para incrementar la confiabilidad del enlace entre módulos en presencia de ruido eléctrico, se integra un canal de comunicación diferencial RS-485 mediante un transceptor THVD1400, adecuado para topologías multipunto y entornos de operación con interferencia electromagnética [10][11][12][13].

El desarrollo del sistema se llevó a cabo mediante un proceso iterativo de diseño, prototipado, evaluación y ajuste, que permitió refinar progresivamente tanto la arquitectura electrónica como la implementación del firmware y la plataforma de visualización. Como resultado, se consolidó una solución con hardware dedicado, orientada a mejorar la integración física del sistema y su viabilidad de uso en un entorno académico. En paralelo, se implementó una plataforma de visualización en computador para la recepción de datos, la graficación en tiempo real y la exportación en formatos estructurados, con el fin de facilitar la trazabilidad y el análisis posterior de la información registrada [14][15][16][17].

Finalmente, este documento se organiza de la siguiente manera: en los primeros capítulos se presentan el planteamiento del problema, la justificación y los objetivos del proyecto; posteriormente se desarrolla el marco de referencia; luego se describen la metodología, el diseño y la implementación del sistema, incluyendo hardware, firmware y plataforma de visualización; y, por último, se exponen los resultados obtenidos, las conclusiones, las recomendaciones y los anexos correspondientes[1][4][5][13].

Planteamiento del Problema

En el contexto de la educación en ingeniería térmica en Colombia, los bancos de transferencia de calor constituyen equipos de laboratorio diseñados para reproducir y estudiar de forma controlada los mecanismos fundamentales de transmisión de energía térmica: conducción, convección y radiación. Estos equipos permiten a los estudiantes observar experimentalmente cómo el calor se propaga a través de materiales y fluidos bajo condiciones definidas, contrastando los modelos teóricos con mediciones reales obtenidas directamente sobre el sistema físico. Contar con herramientas prácticas que complementen la teoría resulta crucial en esta área, ya que la experimentación en fenómenos como la transferencia de calor permite a los estudiantes consolidar un conocimiento más profundo y duradero, formándose como ingenieros capaces de confrontar la teoría con la realidad. No obstante, muchas instituciones educativas del país adolecen de equipos de laboratorio funcionales y precisos, particularmente en el área térmica, lo cual limita gravemente la interacción de los estudiantes con datos reales y el desarrollo de destrezas prácticas. En consecuencia, los futuros ingenieros a menudo se ven reducidos a un aprendizaje predominantemente teórico, sin la oportunidad de contrastar sus conocimientos con experimentos tangibles.

Actualmente, el vacío tecnológico en los laboratorios de transferencia de calor se evidencia en diversos síntomas. Por ejemplo, la ausencia de un sistema embebido confiable para adquirir y monitorear temperaturas en tiempo real conlleva retrasos en la validación experimental de los modelos estudiados y la pérdida de información valiosa durante las prácticas, al no disponerse de mediciones oportunas y fiables [18]. De hecho, la falta de equipos experimentales adecuados puede imposibilitar la comprobación práctica de resultados teóricos en este campo, dificultando reproducir o medir con claridad procesos de conducción térmica en el laboratorio. Aun cuando se han empezado a diseñar bancos didácticos de transferencia de calor para mitigar esta carencia, muchos de ellos no incorporan un sistema de instrumentación integrado y eficaz, por lo que persiste la dificultad para obtener datos experimentales de calidad. Además, las soluciones comerciales existentes que podrían cubrir esta necesidad presentan obstáculos importantes: suelen tener costos elevados, acceso restringido o diseños cerrados que no admiten personalización ni adaptaciones a las necesidades particulares [19]. A esto se suma que, cuando no se dispone de una arquitectura organizada y modular, las tareas de mantenimiento, expansión y reemplazo de elementos del sistema se vuelven más complejas, aumentando la probabilidad de errores de integración y dificultando su aprovechamiento en entornos académicos. Esto significa que las instituciones educativas, especialmente aquellas con presupuesto limitado, no pueden adquirir ni desarrollar fácilmente sistemas listos para usar, intensificando las deficiencias de instrumentación en sus laboratorios térmicos.

Las consecuencias de no contar con instrumentación adecuada en los laboratorios de transferencia

de calor no impiden la formación, pero sí la limitan de manera significativa. Los estudiantes pueden realizar actividades prácticas relacionadas con fenómenos térmicos sin sistemas de medición automatizados; sin embargo, la disponibilidad de instrumentación confiable amplía considerablemente las posibilidades de aprendizaje: permite registrar datos con mayor precisión, repetir experimentos bajo condiciones controladas y contrastar resultados con modelos teóricos de forma sistemática. Sin esta capacidad, el análisis experimental queda reducido a observaciones cualitativas o a mediciones puntuales de baja resolución, lo que limita el desarrollo de competencias clave como la interpretación de datos, el control de variables y el pensamiento crítico basado en evidencia. En particular, habilidades relacionadas con la instrumentación, el monitoreo y el análisis de sistemas térmicos se consolidan de manera más efectiva cuando el estudiante tiene acceso a herramientas que le permitan interactuar con el fenómeno en tiempo real. Estudios pedagógicos resaltan que la construcción y el uso de dispositivos experimentales en el aula favorecen el desarrollo de destrezas cognitivas fundamentales a partir de la práctica [cita]. En ese sentido, fortalecer la instrumentación de los laboratorios térmicos no es un requisito indispensable para la formación, pero sí una mejora con impacto directo en la profundidad y calidad del aprendizaje experimental.

Justificación

El desarrollo de un sistema embebido para la adquisición y visualización de temperatura en un banco de transferencia de calor responde a la necesidad de fortalecer los procesos educativos mediante la incorporación de herramientas tecnológicas accesibles, funcionales y contextualizadas. En muchas instituciones, especialmente en contextos latinoamericanos, existe una carencia de equipos experimentales que permitan a los estudiantes interactuar con fenómenos térmicos en tiempo real, lo cual limita la comprensión y aplicación de conceptos fundamentales en ingeniería térmica [20][21].

Desde la perspectiva teórica, el proyecto se fundamenta en enfoques de aprendizaje activo y enseñanza basada en la indagación, los cuales promueven la participación directa del estudiante en la experimentación, la observación y la interpretación de resultados. Diversos estudios resaltan que los laboratorios de tipo exploratorio favorecen el desarrollo de competencias analíticas y científicas en comparación con prácticas puramente demostrativas. En este sentido, la posibilidad de visualizar y analizar datos térmicos en tiempo real fortalece la reflexión, la formulación de hipótesis y la construcción del conocimiento a partir de evidencia experimental [20][21][22].

En el ámbito metodológico, se adopta un enfoque de diseño e implementación de sistemas embebidos que integra instrumentación, sensado digital y comunicación entre dispositivos. Este enfoque permite validar principios de transferencia de calor mediante mediciones precisas y reproducibles, siguiendo una lógica de prototipado iterativo, evaluación experimental y mejora progresiva del sistema. De acuerdo con Velásquez et al. (2023), el diseño sistemático de bancos térmicos con enfoque de bajo costo constituye una estrategia eficaz para la enseñanza aplicada de la ingeniería, al facilitar un aprendizaje significativo basado en la experimentación directa [23].

Desde el punto de vista del diseño del sistema, la modularidad constituye una decisión técnica y funcional clave. La separación de funciones en distintas tarjetas, tales como adquisición, comunicación, alimentación e interconexión, permite reducir el acoplamiento entre subsistemas, facilitar las labores de mantenimiento y mejorar la escalabilidad del prototipo. De esta manera, una falla o modificación en un módulo específico no compromete necesariamente el funcionamiento del sistema completo, lo cual resulta especialmente valioso en entornos académicos donde se requiere ajustar, depurar y reutilizar la plataforma en diferentes fases de desarrollo. Asimismo, el uso de un backplane como estructura de interconexión centraliza la distribución de alimentación y señales comunes, reduce errores de cableado, favorece el montaje ordenado del sistema y facilita la incorporación de nuevas tarjetas o expansiones futuras. En conjunto, esta arquitectura modular aporta flexibilidad, mantenibilidad y posibilidad de crecimiento, cualidades deseables en soluciones didácticas de bajo costo orientadas a la experimentación y la instrumentación académica [23][24][25][26][27].

En cuanto a la justificación práctica, la implementación del sistema propuesto ofrece una alternativa económica frente a los equipos comerciales de adquisición de datos, los cuales suelen ser costosos y de difícil acceso para instituciones educativas con presupuestos limitados. La plataforma desarrollada promueve competencias en electrónica, instrumentación, programación y análisis de datos, pilares fundamentales en la formación del ingeniero electrónico. De igual manera, la integración de sensores, microcontrolador, comunicación y visualización en una sola solución contribuye a que los estudiantes comprendan de forma más completa la relación entre medición, procesamiento y análisis de información experimental. Investigaciones recientes sobre la integración del Internet de las Cosas en entornos educativos también destacan su potencial para mejorar la autonomía del estudiante y optimizar la gestión de información experimental [28][29][5][30].

Desde la dimensión social y tecnológica, el proyecto impulsa la autonomía y sostenibilidad educativa al fomentar el desarrollo interno de recursos experimentales dentro del entorno universitario. Esto contribuye a la apropiación del conocimiento tecnológico, al fortalecimiento de capacidades locales de diseño e implementación y a la innovación aplicada en contextos académicos emergentes. Según Bernhard (2018), la calidad del aprendizaje en el laboratorio depende tanto de la metodología empleada como de la adecuación del equipamiento experimental a las necesidades reales del usuario, por lo que disponer de una plataforma funcional, ajustable y replicable representa un aporte significativo al proceso formativo [27].

Finalmente, el trabajo se alinea con los objetivos de la educación colombiana en torno al uso de tecnologías digitales, el fomento de la investigación aplicada y la formación basada en competencias. El sistema propuesto, al permitir la conexión de múltiples módulos de sensado, la visualización en tiempo real y la exportación de datos para análisis o simulación, se constituye en una herramienta pedagógica de alto impacto, replicable y adaptable a otros contextos académicos. En consecuencia, esta propuesta no solo responde a una necesidad técnica concreta dentro del banco de transferencia de calor, sino que también aporta una base metodológica y tecnológica para futuros desarrollos en instrumentación didáctica y sistemas embebidos aplicados a la educación en ingeniería [28][29][4][5].

Objetivos

4.1. Objetivo General

Diseñar e implementar un sistema embebido para la adquisición de datos de temperatura a partir de sensores distribuidos en un banco de transferencia de calor, que permita la transmisión, visualización y análisis de la información en tiempo real mediante una plataforma digital.

4.2. Objetivos Específicos

- Identificar y seleccionar la tecnología de sensado de temperatura más adecuada en términos de precisión, inmunidad al ruido, compatibilidad con sistemas embebidos y viabilidad de integración en el banco de transferencia de calor.
- Diseñar la arquitectura del sistema embebido para la adquisición simultánea de múltiples señales de temperatura, garantizando una transmisión confiable de los datos hacia una plataforma de visualización mediante protocolos de comunicación compatibles.
- Implementar un prototipo funcional del sistema embebido, incluyendo el desarrollo del software necesario para la lectura, procesamiento y transmisión de los datos provenientes de los sensores.
- Validar el desempeño del sistema mediante pruebas de comunicación y visualización, verificando la correcta recepción, almacenamiento y presentación en tiempo real de los datos de temperatura adquiridos.

Marco de Referencia

5.1. Áreas Temáticas

Las áreas temáticas del proyecto, utilizando la taxonomía IEEE (términos en inglés tal como aparecen en el documento), son:

- Sensors - Thermal sensors - Temperature sensors - Thermometers.
- Instrumentation and measurement - Measurement - Thermal variables measurement - Temperature measurement.
- Instrumentation and measurement - Monitoring - Remote monitoring.
- Circuits and systems - Microprocessors - Microcontrollers.
- Computers and information processing - System - Data systems - Data acquisition.
- Computers and information processing - Systems - Data systems - Data processing - Data transfer.
- Communications technology - Communication systems - Computer networks - Internet - Internet of Things.
- Computers and information processing - Systems - Software - Embedded software.
- Computers and information processing - Systems - Real-time systems.

Ver clasificación en <https://www.ieee.org/content/dam/ieee-org/ieee/web/org/pubs/ieee-taxonomy.pdf>.

5.2. Marco Teórico

Este proyecto integra fundamentos de transferencia de calor con instrumentación electrónica, adquisición de datos y comunicaciones industriales, con el propósito de habilitar medición y visualización de temperatura en tiempo real dentro de un banco de transferencia de calor de uso académico. El marco teórico se organiza en: (i) transferencia de calor y medición experimental, (ii) adquisición de datos mediante sistemas embebidos, (iii) sensado de temperatura con sensores digitales, (iv) comunicación robusta para entornos ruidosos, (v) arquitectura modular inspirada en buses industriales, y (vi) normatividad y limitaciones aplicables en Colombia [4][5][30].

5.2.1. Transferencia de calor y medición experimental

La transferencia de calor describe el flujo de energía térmica desde regiones de mayor temperatura hacia regiones de menor temperatura, y se presenta principalmente por conducción, convección y radiación. En un banco de transferencia de calor, estos mecanismos se estudian mediante perfiles de temperatura y condiciones controladas que permiten relacionar variables físicas (material, geometría y condiciones de frontera) con el comportamiento térmico observado [4].

En particular, la conducción térmica es clave cuando se analiza el gradiente de temperatura a lo largo de un sólido conductor. Este fenómeno se modela comúnmente mediante la ley de Fourier, donde el flujo de calor es proporcional al gradiente de temperatura y a la conductividad térmica del material, siendo un fundamento para comparar resultados experimentales con modelos teóricos y/o simulaciones [4].

En bancos térmicos con circulación de fluido o disipación hacia el ambiente, la convección (natural o forzada) puede influir significativamente en la distribución de temperatura y en el equilibrio térmico del sistema. Por ello, es relevante contar con mediciones repetibles y sincronizadas para validar experimentalmente los efectos del entorno y del régimen de transferencia (por ejemplo, cambios de caudal o condiciones de aislamiento) [1][23].

5.2.2. Sistemas embebidos y adquisición de datos

Un sistema embebido es una combinación de hardware y software diseñada para ejecutar funciones específicas dentro de un sistema mayor, normalmente con operación en tiempo real, consumo energético moderado y alta confiabilidad. En este proyecto, el sistema embebido se encarga de leer sensores de temperatura, procesar los datos y transmitirlos a una plataforma de visualización/registro para análisis experimental [30].

La adquisición de datos (DAQ) se entiende como el proceso de capturar señales físicas (temperatura) y convertirlas en información digital útil, garantizando integridad, trazabilidad y sincronización. En aplicaciones académicas, una solución DAQ de bajo costo pero bien diseñada permite repetir prácticas, comparar ensayos y exportar datos para análisis y simulación, sin depender de equipos comerciales cerrados o de alto costo [5].

5.2.3. Microcontroladores

Un microcontrolador es un circuito integrado que combina en un solo chip un procesador, memoria no volátil para el almacenamiento del firmware, memoria RAM para datos en tiempo de ejecución y periféricos de entrada/salida. Esta integración permite desarrollar sistemas embebidos compactos y autónomos, capaces de ejecutar tareas de adquisición, procesamiento y comunicación sin depender de un computador externo [30]. En el contexto de sistemas de adquisición de datos para laboratorios académicos, los microcontroladores representan una solución de bajo costo y alta

flexibilidad, ya que pueden adaptarse a distintas arquitecturas de hardware mediante la selección de periféricos adecuados: interfaces seriales, temporizadores, convertidores analógico-digitales y puertos de propósito general, entre otros. Su capacidad de operar en un rango amplio de voltaje y su facilidad de integración en tarjetas modulares los hacen especialmente apropiados para prototipos educativos donde se requiere escalabilidad y control local del proceso de adquisición [30].

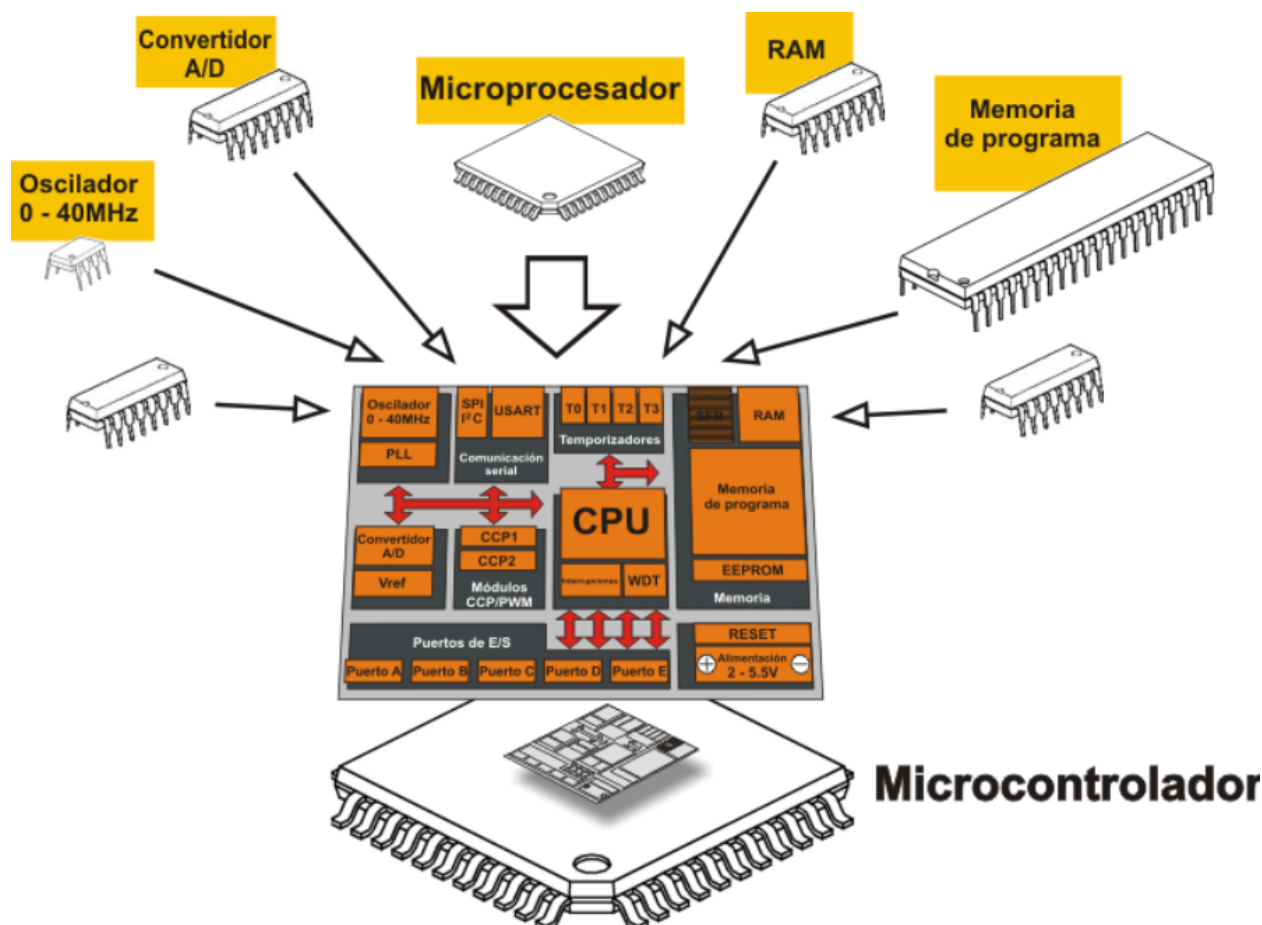


Figura 5.1: Microcontroladores

5.2.4. Sensores de temperatura

Un sensor de temperatura es un dispositivo que convierte una variable térmica en una señal eléctrica interpretable por un sistema electrónico. Existen diferentes principios de operación: los sensores resistivos como el PT100 varían su resistencia con la temperatura, los termopares generan una diferencia de potencial proporcional al gradiente térmico, y los sensores digitales integran el proceso de conversión analógico-digital en el propio dispositivo, entregando directamente un valor numérico por un protocolo de comunicación [31]. En aplicaciones de instrumentación para bancos de

transferencia de calor, donde se requiere medir temperatura en múltiples puntos de forma simultánea, la elección del tipo de sensor tiene implicaciones directas sobre la complejidad del cableado, la inmunidad al ruido y la facilidad de integración con el sistema de adquisición. La selección del sensor específico utilizado en este proyecto, junto con su justificación frente a otras alternativas, se presenta en la sección 6.3. [31].

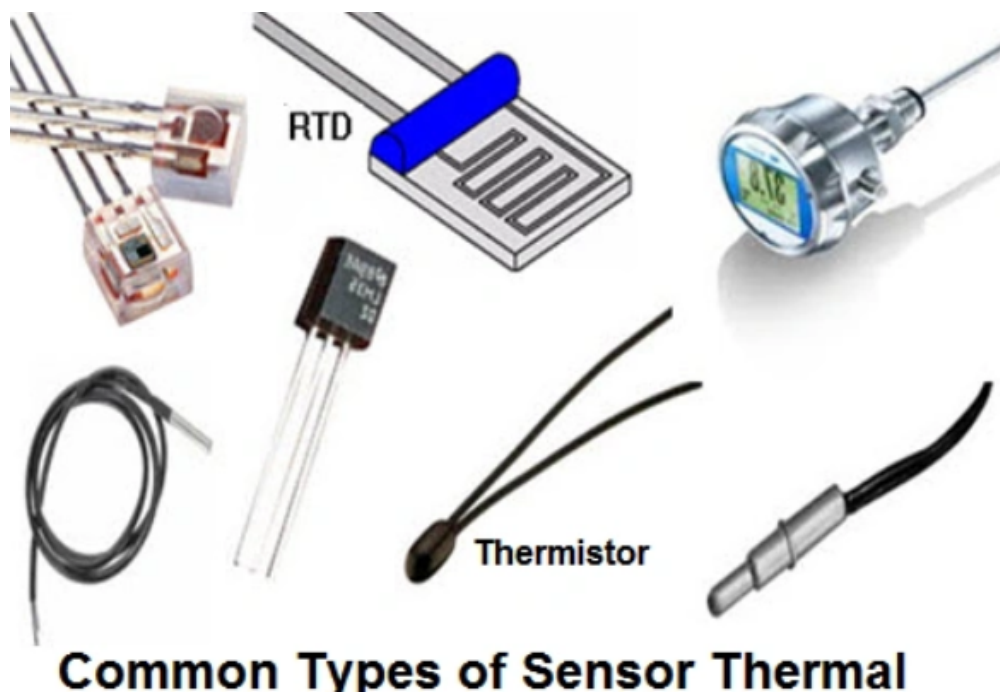


Figura 5.2: Sensores de temperatura

5.2.5. Comunicación industrial RS485 y transceptor THVD1400

En laboratorios y entornos con equipos eléctricos cercanos (fuentes, cargas, resistencias calefactoras), el ruido electromagnético puede degradar comunicaciones y corromper datos. Por ello, para enlaces entre tarjetas o módulos distribuidos, es común emplear transmisión diferencial (par trenzado) con alta inmunidad al ruido y capacidad de cubrir distancias mayores que buses TTL directos [13].

RS485 (basado en un esquema balanceado/diferencial) es ampliamente usado en aplicaciones industriales por su robustez en ambientes ruidosos y su capacidad de operar en redes multipunto. Su implementación adecuada requiere considerar topología de cableado, terminación en extremos del bus y buenas prácticas de puesta a tierra y apantallamiento, para minimizar reflexiones y asegurar integridad de señal [13].

El THVD1400 (Texas Instruments) es un transceptor RS485 half-dúplex orientado a aplicaciones industriales. Al integrarlo con un microcontrolador como el ATmega328PB, se obtiene un canal físico

robusto para conectar módulos de adquisición sobre un bus común, manteniendo confiabilidad en presencia de ruido eléctrico y permitiendo escalar el número de tarjetas en un banco térmico [12] [11].

Desde el diseño electrónico, el uso de transceptores RS485 implica decisiones prácticas: resistencias de terminación en los extremos, control de dirección (DE/RE en half-dúplex), y criterios de cableado (par trenzado, longitud, derivaciones cortas). Estos elementos son críticos para que el sistema de adquisición entregue datos completos y consistentes, condición necesaria para graficación y exportación posterior [13] [10].

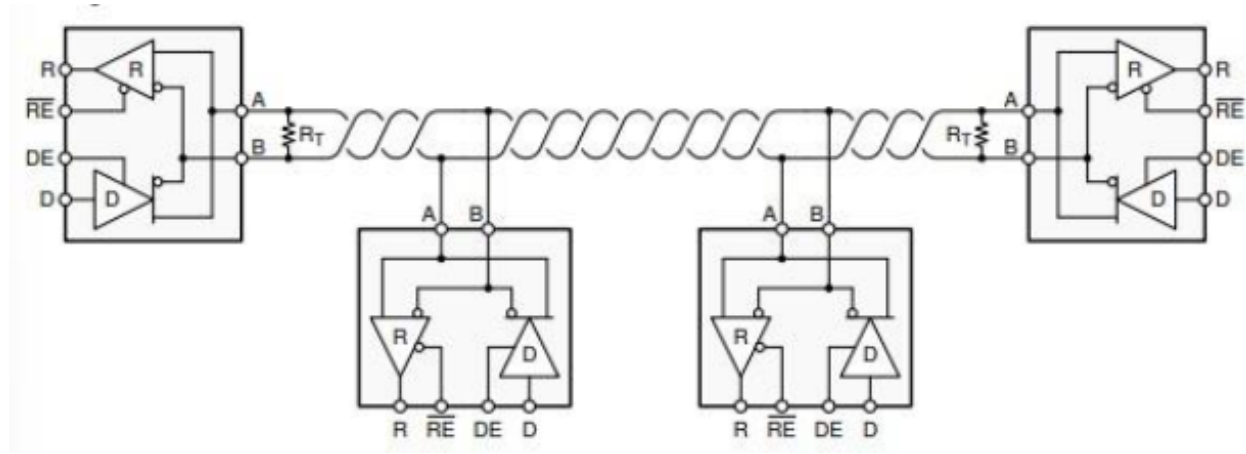


Figura 5.3: Red RS-485 típica.

5.2.6. Arquitecturas modulares tipo VMEbus y portadoras 3U

La modularidad es un criterio de diseño relevante en bancos didácticos porque permite crecer la capacidad sin rediseñar todo el sistema (más sensores, más tarjetas, más canales). En sistemas industriales, una referencia clásica de modularidad es VMEbus, donde múltiples tarjetas se conectan a un backplane y comparten alimentación y bus, facilitando mantenimiento, expansión y organización física del sistema [24].

En este proyecto, se adopta una aproximación conceptual similar: cada tarjeta de adquisición puede gestionar un conjunto fijo de sensores (por ejemplo, 8) y conectarse a un bus común para comunicar datos al nodo maestro, manteniendo independencia funcional por módulo. Esto habilita escalabilidad (más tarjetas) y permite aislar fallas o reemplazar módulos sin afectar el conjunto completo del banco [24].

Como referencia de implementación modular en formato 3U (half-height), existen tarjetas portadoras (carrier) que alojan módulos adicionales y exponen conexiones de campo, manteniendo una interfaz estándar hacia el bus del sistema. Por ejemplo, en portadoras VMEbus 3U se documenta la posibilidad de alojar módulos plug-in y centralizar las conexiones de E/S en el frontal de la tarjeta,

lo cual es consistente con la idea de tarjetas de adquisición replicables para un banco experimental [7] [8].

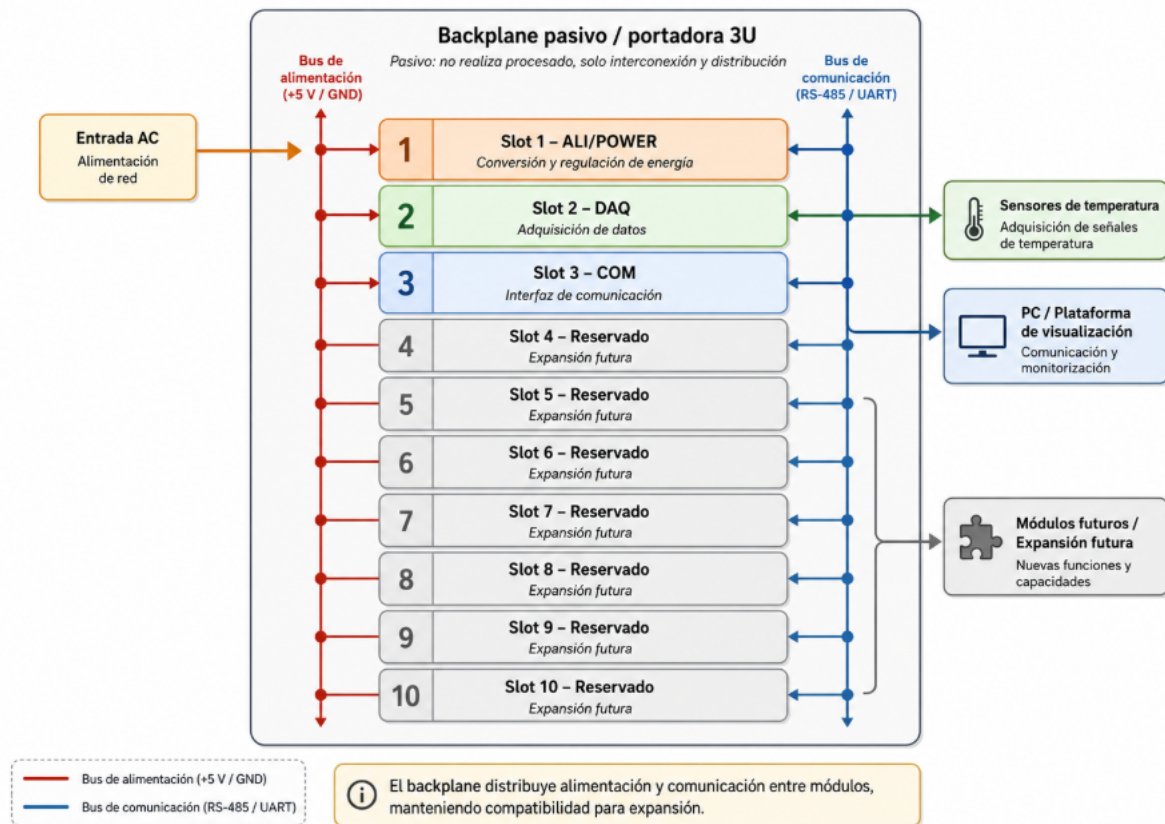


Figura 5.4: Diagrama de la arquitectura

5.2.7. Visualización y simulación térmica

La visualización de datos térmicos permite interpretar el comportamiento de un sistema a través de gráficas de temperatura vs. tiempo o perfiles espaciales, lo cual es esencial para identificar transitorios, estabilidad, errores de medición y patrones de disipación. En contextos educativos, la visualización en tiempo real fortalece la relación teoría-experimento al permitir que el estudiante observe el fenómeno mientras ocurre [9].

5.2.8. Normatividad y limitaciones en contexto colombiano

Desde la perspectiva normativa, el sistema propuesto (tarjetas electrónicas, fuente de alimentación, cableado y operación en laboratorio) debe cumplir criterios de seguridad eléctrica aplicables en Colombia. En este marco, el RETIE aplica principalmente en lo relacionado con: (i) seguridad

en instalaciones de baja tensión dentro del laboratorio (puesta a tierra, protecciones y condiciones de conexión), (ii) selección y uso seguro de equipos eléctricos que se conectan a la red (fuentes AC–DC, tomas, conductores y protecciones), y (iii) verificación y evaluación de conformidad cuando se emplean componentes certificados o se instalan equipos de carácter permanente. En este proyecto, lo anterior se refleja en la adopción de una alimentación aislada mediante módulo AC–DC, el uso de protecciones básicas, el cableado acorde a baja tensión y la recomendación de operar bajo infraestructura eléctrica del laboratorio que cumpla RETIE (tableros, protecciones, tierra física) [32].

Para el equipo electrónico desarrollado (tarjetas DAQ/COM y backplane), aunque no se pretende certificarlo como “producto comercial”, se toman como referencia criterios de seguridad de equipos de medición y laboratorio descritos en IEC 61010-1. En particular, se consideran aplicables: (i) protección contra choque eléctrico (aislamiento, separación y barreras), (ii) riesgos térmicos y de incendio (limitación de corrientes, disipación y materiales), (iii) marcado/identificación y documentación de operación segura, y (iv) accesibilidad a partes energizadas (evitar exposición de contactos en operación normal). En consecuencia, el diseño privilegia niveles de tensión seguros en el backplane, señalización de conectores, y recomendaciones de operación que reducen la manipulación con el sistema energizado [25]. En el contexto institucional colombiano, la ejecución del proyecto en laboratorio requiere alineación con el SG-SST (Decreto 1072 de 2015), especialmente en lo aplicable a: (i) identificación de peligros y valoración de riesgos durante ensamble y pruebas (riesgo eléctrico, térmico y mecánico), (ii) procedimientos de trabajo seguro (energización/desenergización, inspección previa, uso de EPP si aplica), y (iii) condiciones de mantenimiento y corrección de fallas. Por ello, se recomienda que las pruebas se realicen con protocolos de verificación, checklist de energización y procedimientos de intervención, reduciendo el riesgo durante prácticas académicas [26].

En el componente ambiental, al tratarse de un sistema prototipo con iteraciones de PCB y reemplazo de sensores, resulta aplicable la Ley 1672 de 2013 (RAEE) en cuanto a la gestión responsable de residuos electrónicos. Esto implica: (i) separación y almacenamiento temporal adecuado de tarjetas o componentes descartados, (ii) entrega a gestores autorizados cuando aplique, y (iii) promoción de prácticas de reutilización/recuperación de componentes en la medida de lo posible. Estas acciones reducen el impacto ambiental derivado de prototipado y mantenimiento [33].

Finalmente, desde el punto de vista de política pública y limitaciones socioeconómicas, el proyecto se articula con lineamientos nacionales de transformación digital al incorporar captura, registro y uso de datos en procesos formativos. En este sentido, el CONPES 3975 respalda la adopción de tecnologías digitales y el fortalecimiento de capacidades, lo cual enmarca la pertinencia de soluciones didácticas de adquisición y visualización de datos; sin embargo, la adopción institucional puede estar limitada por disponibilidad presupuestal, infraestructura de laboratorio y capacidad de mantenimiento técnico [34].

5.3. Trabajos Relacionados

A lo largo de la última década se han propuesto diversas soluciones para la adquisición y visualización de datos de temperatura mediante sistemas embebidos de bajo costo. Estas propuestas emplean diferentes plataformas de microcontroladores, protocolos de comunicación, estrategias de sensado y esquemas de visualización, lo que permite identificar enfoques que pueden adaptarse (o no) a un banco didáctico de transferencia de calor.

En esta sección se presentan cinco trabajos representativos que abordan directamente la problemática de la adquisición distribuida de temperatura, la transmisión de datos entre sistemas embebidos y un computador, y la visualización en tiempo real. La comparación de estos estudios permite establecer el estado actual del problema e identificar oportunidades para una propuesta orientada a un banco de laboratorio educativo, con requisitos de bajo costo, escalabilidad y visualización en tiempo real.

Estudio 1: Sistema DAQ hardware/software para monitoreo de temperatura en tiempo real con Arduino y LabVIEW

El trabajo presentado por Segura, Bartolucci y Andújar [Sensors, 2017, DOI: 10.3390/s17071600] desarrolla un sistema de adquisición de datos de temperatura en tiempo real basado en la plataforma embebida Arduino Micro (ATmega32u4), combinada con una interfaz de visualización implementada en NI LabVIEW. El sistema utiliza sensores de coeficiente de temperatura negativo (NTC) distribuidos a lo largo de una pila de celdas de combustible, y transmite los datos adquiridos hacia el computador mediante comunicación USB-serial, donde son procesados y presentados gráficamente en tiempo real.

La contribución principal de este estudio radica en demostrar la viabilidad de una arquitectura DAQ embebida de bajo costo con múltiples puntos de medición simultáneos, transmisión confiable hacia PC y visualización dinámica mediante instrumentación virtual. Estos elementos son directamente análogos a los requerimientos del banco didáctico desarrollado en el presente proyecto.

Sus limitaciones frente al contexto de este trabajo son dos: primero, el sistema emplea LabVIEW como entorno de visualización, que es un software privativo y de alto costo que no está disponible en todos los laboratorios universitarios; segundo, está diseñado para una aplicación industrial específica (celdas de combustible), sin contemplar criterios de replicabilidad educativa ni exportación y almacenamiento estructurado de datos para análisis experimental posterior [35].

Estudio 2: Microcontroller-Based Electronic Laboratory Measurement Device for Distance Education

Smolaninovs y Terauds [Electronics, 2025, DOI: 10.3390/electronics14030438] presentan el desarrollo de un dispositivo compacto y de bajo costo basado en microcontrolador con interfaz USB, diseñado específicamente para laboratorios educativos en entornos presenciales y remotos. El dispositivo integra un osciloscopio de dos canales, un generador de señales y una fuente de alimentación

bipolar, con interfaz gráfica desarrollada en MATLAB. Su costo de fabricación es de EUR 24 y fue validado con estudiantes durante dos semestres de práctica en electrónica analógica. Su aporte principal para el presente proyecto es evidenciar la vigencia de arquitecturas embebidas con comunicación USB para adquisición de datos en laboratorio, sin requerir drivers adicionales, y confirmar que estas soluciones de bajo costo continúan siendo una alternativa pertinente y funcional frente a equipos comerciales costosos. Las limitaciones de este estudio respecto al presente trabajo son su enfoque en señales analógicas genéricas — no orientado a temperatura como variable específica ni a bancos térmicos — y la ausencia de un mecanismo de almacenamiento y exportación estructurada de datos para validación experimental posterior [36].

Estudio 3: Sistema DAQ de bajo costo para caracterización experimental de colectores solares térmicos con ESP32

Argiriou [Electronics, 2022, DOI: 10.3390/electronics11060934] presenta el diseño y validación de un sistema automático de adquisición de datos (ADAS) descentralizado y de bajo costo, basado en el microcontrolador ESP32 y software de código abierto. El sistema está diseñado para la caracterización experimental de colectores solares térmicos que trabajan con agua o aire, y cumple con los requisitos de la norma ISO 9806:2017. En cada ciclo de muestreo, el ESP32 lee todos los sensores, calcula estadísticos (media y desviación estándar) y publica los datos a un servidor MQTT, desde donde son almacenados y visualizados mediante un panel web.

Este trabajo es altamente relevante para el presente proyecto porque demuestra que el ESP32 es una plataforma embebida capaz de gestionar adquisición multicanal, procesamiento local de datos, transmisión confiable y visualización remota, todo dentro de un presupuesto reducido y con software de código abierto. La arquitectura modular y la capacidad de escalar el número de sensores son características compartidas con el sistema desarrollado en este trabajo. Sus limitaciones frente al contexto del banco didáctico son la orientación hacia aplicaciones de energía solar (no transferencia de calor en laboratorio), la dependencia de conectividad WiFi y servidor MQTT (que puede no estar disponible en todos los entornos de laboratorio), y la ausencia de una interfaz gráfica local de escritorio para el docente o estudiante [37].

Estudio 4: Sistema multicanal de medición de temperatura con DS18B20 y LabVIEW para adquisición y visualización en tiempo real

Iqbal et al. [International Journal of Current Science Research and Review, 2023, Vol. 6(6), pp. 3091–3109] presentan el diseño e implementación de un sistema multicanal de medición de temperatura basado en el sensor digital DS18B20 y la plataforma de instrumentación virtual LabVIEW. El sistema integra módulos de conversión analógico-digital, módulos de medición de temperatura y módulos de adquisición de datos, e implementa el teorema de muestreo de Nyquist-Shannon junto con filtros digitales para reducir la incertidumbre de medición. Los resultados experimentales reportan una exactitud de medición inferior a 0.1 °C, con alta estabilidad y escalabilidad multicanal.

La pertinencia de este trabajo para el presente proyecto es directa: el sensor DS18B20 es el

mismo utilizado en el banco didáctico, el enfoque multicanal responde a la misma necesidad de múltiples puntos de medición, y el esquema de visualización en LabVIEW demuestra la utilidad de interfaces gráficas dinámicas para análisis térmico experimental. Además, el trabajo evidencia que la exactitud del DS18B20 es suficiente para aplicaciones de laboratorio académico.

Sus limitaciones respecto al presente trabajo incluyen el uso de LabVIEW (software privativo), la ausencia de un mecanismo de exportación de datos para análisis posterior, y que el sistema no contempla transmisión serial estructurada entre el microcontrolador y el computador, aspecto central en la arquitectura desarrollada en este proyecto [38].

Estudio 5: Sistema embebido IoT para monitoreo de temperatura con DS18B20, ESP32 y visualización en red para laboratorio de mecatrónica

Salahuddin et al. [International Journal of Advanced Multidisciplinary, 2024] presentan el diseño e implementación de un sistema de monitoreo de temperatura basado en el sensor DS18B20, el microcontrolador ESP32 y una pantalla LCD local con interfaz I2C. El sistema es capaz de medir temperatura en tiempo real y presentar los datos tanto en la pantalla local como a través de una red IoT. El trabajo está orientado explícitamente al contexto educativo vocacional (laboratorio de mecatrónica), con énfasis en la integración de tecnologías embebidas y comunicaciones para apoyar actividades prácticas estudiantiles.

Este trabajo comparte con el presente proyecto, el sensor de temperatura (DS18B20), el enfoque educativo y la necesidad de visualización local en tiempo real. Constituye el antecedente más cercano en cuanto a contexto de aplicación y componentes tecnológicos utilizados.

Sus limitaciones frente al sistema desarrollado en este proyecto son la ausencia de un esquema de comunicación serial estructurada hacia un computador, la falta de una interfaz gráfica de escritorio para visualización y análisis histórico, y la no contemplación del almacenamiento y exportación de datos para validación experimental con modelos teóricos, que es uno de los objetivos centrales del banco didáctico de transferencia de calor [39].

Comparación de trabajos

En la Tabla se resumen los cinco trabajos revisados según su contexto, aplicación, plataforma embebida, visualización, escalabilidad y limitaciones. Esta comparación permite identificar qué componentes aportan al diseño del sistema de adquisición y cuáles evidencian la necesidad de una solución embebida completa orientada al laboratorio educativo de transferencia de calor.

Tabla 5.1: Comparativa de trabajos relacionados

Estudio	Contexto	Plataforma embebida	Sensor de temperatura	Visualización en tiempo real	Escalabilidad	Limitaciones principales
1	Industrial (celdas de combustible)	Arduino Micro ATmega32u4 + LabVIEW	NTC (termistores)	Sí — LabVIEW, tiempo real	Media	LabVIEW es un software privativo y costoso; sin exportación estructurada de datos para análisis experimental.
2	Educativo / genérico (laboratorio de ingeniería)	Microcontrolador PIC con USB-HID integrado	Genérico (entradas ADC analógicas)	No — solo transmisión de datos, sin GUI gráfica	Alta	Sin visualización gráfica ni almacenamiento de datos; no orientado a temperatura como variable específica.
3	Investigación (colectores solares térmicos)	ESP32 + software <i>open source</i> + servidor MQTT	Temperatura y flujo (múltiples sensores)	Sí — panel web en tiempo real	Alta	Requiere conectividad WiFi y servidor MQTT; sin interfaz local de escritorio para el usuario.
4	Académico / investigación (instrumentación virtual)	Microcontrolador + LabVIEW (interfaz virtual)	DS18B20 (digital, protocolo 1-Wire)	Sí — LabVIEW, tiempo real, multicanal	Alta	LabVIEW privativo; sin transmisión serial estructurada desde microcontrolador; sin exportación de datos.
5	Educativo (laboratorio de mecatrónica vocacional)	ESP32 + pantalla LCD I2C local + red IoT	DS18B20 (digital, protocolo 1-Wire)	Parcial — LCD local + visualización IoT básica	Media	Sin interfaz de escritorio para análisis; sin almacenamiento histórico ni exportación de datos para validación experimental.

Vacío identificado

A partir de estos cinco trabajos se observa que existen aportes relevantes en: (i) arquitecturas DAQ embebidas de bajo costo con múltiples puntos de medición, (ii) uso de microcontroladores y el sensor DS18B20 para monitoreo de temperatura en tiempo real, y (iii) interfaces de visualización dinámica para análisis experimental. Sin embargo, ninguno de los trabajos revisados integra simultáneamente, para un banco didáctico de transferencia de calor: adquisición distribuida multicanal con DS18B20, transmisión serial estructurada y confiable hacia un computador, visualización gráfica en tiempo real mediante una aplicación de escritorio de código abierto, y almacenamiento y exportación de datos para comparación con modelos teóricos.

Por lo anterior, el presente proyecto se plantea como una solución orientada específicamente a un banco de transferencia de calor de uso educativo, buscando bajo costo, modularidad, visualización en tiempo real y soporte para análisis y validación experimental mediante registro y exportación de datos térmicos [35] [36] [37] [38] [39].

Metodologia

La metodología de este trabajo de grado es de tipo aplicada y tecnológica, con enfoque cuantitativo y experimental, ya que el resultado esperado es un sistema embebido funcional (hardware + firmware + software) validado mediante mediciones repetibles de temperatura y pruebas de comunicación/visualización en un banco de transferencia de calor. El desempeño se evalúa con métricas observables (exactitud de lectura, estabilidad del enlace, pérdida de tramas, latencia de visualización y consistencia de registro/exportación) [5] [30].

Para estructurar el desarrollo se adoptó un proceso iterativo basado en Design Thinking (Empatizar, Definir, Idear, Prototipar y Evaluar), por su orientación al usuario, prototipado rápido y mejora continua a partir de evidencia experimental. En este proyecto el “usuario” corresponde a estudiantes y docentes que operan el banco térmico, quienes requieren instalación sencilla, lecturas confiables y visualización clara en tiempo real [40] [41].

6.1. Enfoque metodológico del proyecto

6.1.1. Tipo de investigación y enfoque

El presente trabajo de grado se enmarca en una investigación de tipo aplicada y tecnológica, ya que está orientado al desarrollo de una solución funcional para una necesidad concreta dentro del contexto académico: la instrumentación de un banco de transferencia de calor mediante un sistema embebido de adquisición y visualización de temperatura. Su finalidad principal no es únicamente generar conocimiento teórico, sino diseñar, implementar e integrar una propuesta técnica que permita obtener mediciones confiables, transmitir las a una plataforma externa y facilitar su análisis en tiempo real.

El enfoque del proyecto es cuantitativo y experimental. Es cuantitativo porque el desempeño del sistema se evalúa a partir de variables observables y medibles, tales como precisión de lectura, tiempo de actualización, estabilidad de la comunicación, pérdida de tramas y consistencia del registro de datos. A su vez, es experimental porque la validación del prototipo se realiza mediante pruebas funcionales sobre el sistema desarrollado, verificando su comportamiento en condiciones de operación semejantes a las de un entorno de laboratorio.

Adicionalmente, el trabajo posee un carácter de desarrollo e integración tecnológica, dado que articula hardware, firmware y software en una misma solución. Esto implica no solo la selección y justificación de componentes, sino también el diseño de la arquitectura electrónica, la implementación del procesamiento embebido, la comunicación entre módulos y la visualización de la información

adquirida. En este sentido, la investigación no se limita a la formulación conceptual del sistema, sino que avanza hasta la construcción y validación de un prototipo funcional orientado a su uso en prácticas académicas de transferencia de calor.

6.1.2. Aplicación de Design Thinking

Para estructurar el desarrollo del proyecto se adoptó un proceso iterativo inspirado en la metodología Design Thinking, debido a su orientación al usuario, su énfasis en el prototipado progresivo y su utilidad para refinar soluciones a partir de evidencia experimental. En este trabajo, el usuario corresponde principalmente a estudiantes y docentes que interactúan con el banco de transferencia de calor y que requieren una herramienta de medición comprensible, confiable, escalable y de fácil integración en el entorno académico.

En la etapa de empatizar, se identificó la necesidad de contar con una solución que permitiera adquirir y visualizar temperatura en tiempo real dentro del banco térmico, atendiendo requerimientos de facilidad de uso, claridad en la visualización, mantenimiento sencillo y posibilidad de reutilización en futuras prácticas. En la etapa de definir, estas necesidades se tradujeron en requerimientos funcionales y no funcionales relacionados con número de sensores, tiempo de actualización, precisión, robustez ante ruido, escalabilidad, integración física y capacidad de exportación de datos.

Posteriormente, en la etapa de idear, se evaluaron distintas alternativas de sensado, plataformas de procesamiento, esquemas de comunicación y opciones de organización modular del sistema. A partir de este análisis se seleccionaron los sensores DS18B20, el microcontrolador ATmega328PB, la comunicación diferencial RS-485 mediante THVD1400 y una arquitectura basada en tarjetas funcionales interconectadas, buscando equilibrio entre costo, facilidad de implementación, mantenibilidad y desempeño en laboratorio.

La etapa de prototipar comprendió el desarrollo progresivo del sistema, iniciando con pruebas preliminares de lectura de temperatura y comunicación en montajes experimentales, y continuando con el diseño de tarjetas electrónicas dedicadas para consolidar la solución en hardware propio. Este proceso incluyó versiones iniciales, integración de módulos, diseño en PCB y ajustes físicos orientados a mejorar el montaje, la organización y la viabilidad de uso del prototipo en el contexto de presentación y validación del proyecto.

Finalmente, en la etapa de evaluación, el sistema fue sometido a pruebas funcionales de lectura multisensor, estabilidad del bus 1-Wire, comunicación, visualización y registro de datos. Los resultados de estas pruebas permitieron identificar errores, realizar correcciones de diseño, ajustar decisiones de implementación y refinar progresivamente tanto el hardware como el firmware. De esta manera, Design Thinking no se empleó sólo como una referencia conceptual, sino como una guía práctica para conducir el desarrollo iterativo del sistema hasta obtener una solución funcional y coherente con las necesidades del banco de transferencia de calor.

Con el fin de evidenciar que el Design Thinking no se empleó únicamente como referencia conceptual sino como guía práctica a lo largo de todo el desarrollo, la Tabla XX resume la trazabilidad

entre cada etapa de la metodología y las secciones específicas del documento donde se puede verificar su aplicación concreta.

Tabla 6.1: Trazabilidad entre las etapas del Design Thinking y el desarrollo del proyecto

Fase	Qué se hizo	Evidencia en el documento
Empatizar	Identificación de necesidades del laboratorio CAP-PUJ: medir temperatura en múltiples puntos, visualización en tiempo real, facilidad de uso, bajo costo y escalabilidad.	Sección 2 (Planteamiento del problema) · Sección 3 (Justificación) · Tabla 2 y Tabla 3 (Requerimientos funcionales y no funcionales).
Definir	Traducción de necesidades en requerimientos medibles: mínimo 8 sensores, actualización $\leq 1,5$ s, precisión $\pm 0,5^\circ\text{C}$, arquitectura modular por tarjetas intercambiables y exportación a CSV.	Sección 6.2.2 (Requerimientos del sistema) · Tabla 2 (RF1–RF7) · Tabla 3 (RNF1–RNF6) · Sección 6.2.3 (Criterios de diseño).
Idear	Evaluación sistemática de alternativas de sensor, microcontrolador y protocolo de comunicación mediante matrices de decisión con criterios ponderados.	Sección 6.3 (Selección de componentes) · Tabla 7 (sensor) · Tablas 8 y 9 (MCU) · Tabla 10 (comunicación).
Prototipar	Desarrollo incremental: pruebas en protoboard → firmware 1-Wire → diseño de PCBs → fabricación en JLCPCB (2 iteraciones) → construcción del chasis metálico → integración completa del sistema.	Secciones 6.4 a 6.7 · Sección 6.7.9 (Fabricación y ensamble) · Sección 6.7.10 (Estructura mecánica) · Anexo D (Piezas auxiliares 3D).
Evaluar	Validación mediante plan de pruebas formal PT-01 a PT-06 sobre el prototipo funcional. Identificación de errores y correcciones iterativas de diseño a partir de los resultados obtenidos.	Sección 7.1 (Plan de pruebas · Tabla 26) · Secciones 7.2 a 7.4 (Resultados) · Sección 7.5 (Hallazgos y limitaciones).

6.2. Alcance, requerimientos y criterios de diseño

El alcance de este trabajo se limita al subsistema de instrumentación del banco de transferencia de calor. El diseño y la construcción del banco físico — estructura mecánica, elementos térmicos y sistema de fluidos — corresponde al proyecto de trabajo de grado del grupo asociado. En este contexto, el presente proyecto asume como punto de partida la existencia del banco físico y los puntos de medición definidos por ese grupo, y se concentra en desarrollar la solución de adquisición, comunicación y visualización que permita registrar la temperatura en esos puntos de forma confiable y en tiempo real. La arquitectura modular y el chasis diseñados en este trabajo fueron concebidos para integrarse físicamente al banco del grupo asociado, y para ser escalables de modo que futuras versiones del sistema puedan incorporar nuevos canales de sensado, tipos de sensores o módulos de comunicación adicionales sin modificar la estructura base.

6.2.1. Alcance de la fase de diseño e implementación

En esta fase del proyecto se abordan las actividades necesarias para materializar un prototipo funcional del sistema de adquisición y visualización de temperatura, incluyendo:

- Selección e integración de sensores de temperatura adecuados para el banco de transferencia de calor.
- Diseño y programación del sistema embebido para adquisición y procesamiento de datos.
- Transmisión hacia una plataforma externa (PC) mediante un enlace de comunicación compatible (p. ej., serial/RS-485/USB).
- Integración con una interfaz de visualización en tiempo real (gráficas, registros y valores) y generación de archivos exportables.

6.2.2. Requerimientos del sistema

Para asegurar trazabilidad entre diseño, implementación y pruebas, los requerimientos se organizaron en funcionales, no funcionales y requerimientos específicos de sensado y adquisición.

Tabla 6.2: Requerimientos funcionales

Código	Requerimiento
RF1	El sistema debe capturar datos de temperatura desde al menos 8 sensores por tarjeta lectora.
RF2	El sistema debe transmitir los datos adquiridos a una plataforma PC con una periodicidad máxima de 1 s por actualización. La conversión de temperatura del DS18B20 en modo de 12 bits tarda aproximadamente 750 ms y la transmisión de una trama de 19 bytes a 9600 bps requiere cerca de 20 ms. Considerando además el tiempo de procesamiento, un periodo de 1 segundo constituye un intervalo adecuado para garantizar que cada solicitud del maestro reciba una lectura fresca del sensor.
RF3	El sistema debe registrar las temperaturas en una estructura organizada y exportable (por ejemplo, CSV o JSON).
RF4	El sistema debe conectarse a una interfaz gráfica que muestre en tiempo real los valores de temperatura.
RF5	El sistema debe permitir una expansión futura para incluir más sensores y/o módulos.
RF6	Debe implementarse una organización tipo backplane pasivo para distribuir alimentación y comunicación entre placas o módulos.
RF7	El sistema debe contar con una fuente de alimentación regulada adecuada para sensores, microcontrolador y transceptores (por ejemplo, rieles de 5 V y/o 3.3 V según aplique).

Tabla 6.3: Requerimientos no funcionales

Código	Categoría	Requerimiento
RNF1	Precisión	Precisión de lectura objetivo de $\pm 0,5^{\circ}\text{C}$ en el rango de operación del banco.
RNF2	Tiempo de respuesta	Tiempo de actualización menor a 1.5 s por muestra total del sistema, considerando el tiempo de conversión del sensor, la transmisión RS-485 y el refresco de la plataforma de visualización.
RNF3	Usabilidad	La visualización debe ser comprensible para usuarios no técnicos (estudiantes en práctica).
RNF4	Escalabilidad	El sistema debe ser modular y permitir crecimiento por adición de tarjetas.
RNF5	Portabilidad / Integración	El sistema embebido debe integrarse físicamente al banco térmico.
RNF6	Robustez	Se debe priorizar la inmunidad al ruido eléctrico en la selección del bus de sensado y comunicación.

Tabla 6.4: Requerimientos del sensor de temperatura

Código	Requerimiento
RS1	El sensor debe ser digital y con buena inmunidad al ruido.
RS2	Debe tener un rango operativo mínimo de -10°C a 100°C , o superior, para cubrir los escenarios del banco.
RS3	Debe permitir conexión multipunto en un mismo bus (p. ej., 1-Wire o I2C).
RS4	Debe tener identificación única o direccionamiento para reconocer cada sensor individualmente.
RS5	Debe contar con una precisión mínima de $\pm 0,5^{\circ}\text{C}$.
RS6	Debe tener un formato compacto y compatible con montaje mecánico sobre el banco (canal térmico o superficie metálica).

Tabla 6.5: Requerimientos del sistema de adquisición y comunicación

Código	Requerimiento
RAQ1	El sistema debe adquirir datos desde varias placas conectadas a través de un bus compartido (p. ej., RS-485/UART según arquitectura).
RAQ2	Cada placa debe transmitir periódicamente sus datos al sistema central (<i>broadcasting</i> o <i>polling</i>).
RAQ3	Los datos deben incluir identificador de placa o módulo, tiempo (si aplica) y valores de temperatura.
RAQ4	El sistema debe permitir recolección continua minimizando pérdida de muestras.
RAQ5	Los datos deben ser compatibles con formatos estandarizados para análisis o simulación (CSV/JSON).
RAQ6	El microcontrolador debe contar con capacidad de procesamiento y periféricos suficientes para manejar lectura múltiple y comunicación.

6.2.3. Criterios de Diseño

A partir de los requerimientos definidos previamente, se establecieron criterios de diseño orientados a guiar tanto la selección de componentes como las decisiones de arquitectura e implementación del sistema. En primer lugar, se consideró la **escalabilidad y modularidad** como un criterio central, de manera que la solución permitiera crecer mediante la incorporación de nuevas tarjetas o módulos sin requerir un rediseño completo del sistema. Este criterio se relaciona con la posibilidad de expansión futura, la organización mediante *backplane* y la necesidad de mantener una estructura flexible para aplicaciones académicas.

En segundo lugar, se prioriza la **robustez ante ruido eléctrico**, favoreciendo el uso de buses diferenciales para la interconexión entre módulos y topologías adecuadas para minimizar interferencias, pérdidas de información y errores de comunicación. Este criterio resulta especialmente relevante en un entorno de laboratorio, donde pueden coexistir fuentes de alimentación, cargas térmicas y otros elementos que afecten la integridad de la señal.

Asimismo, se definió como criterio la **trazabilidad de los datos**, entendida como la capacidad de registrar, organizar y exportar la información adquirida de forma estructurada para su posterior análisis. En este sentido, se planteó el uso de tramas de comunicación y formatos de almacenamiento compatibles con herramientas de procesamiento y simulación, permitiendo conservar la identificación de cada sensor y la evolución temporal de sus mediciones.

Otro criterio importante fue la **viabilidad de implementación**, considerando la disponibilidad de componentes, el costo del sistema y la facilidad de manufactura y ensamble. Dado que se trata de un proyecto con orientación académica y de bajo costo, se buscó un equilibrio entre desempeño técnico, accesibilidad de los dispositivos y factibilidad de fabricación mediante PCB.

De igual forma, se incorporó la **mantenibilidad** como criterio de diseño, buscando que el sistema facilitara la detección de fallas, el reemplazo de sensores o módulos y el diagnóstico del estado general

mediante indicadores o señales de funcionamiento. Esto permite una mejor continuidad del proyecto y simplifica las tareas de ajuste y corrección durante el uso experimental.

Adicionalmente, se consideró la **integración física del prototipo** como un criterio relevante, procurando que el sistema pudiera incorporarse de manera ordenada al banco de transferencia de calor y presentara una disposición física coherente con su uso en laboratorio. Relacionado con esto, también se tuvo en cuenta la **facilidad de montaje**, con el fin de reducir la complejidad en la conexión de módulos, minimizar errores de cableado y favorecer una implementación más limpia y comprensible.

En términos de gestión de la información, se definió además la necesidad de contar con un **formato de datos estructurado**, como CSV o JSON, con marca temporal y campos por sensor, de modo que las mediciones pudieran ser analizadas posteriormente y comparadas con simulaciones o resultados experimentales adicionales. Para ello, la exportación de datos se planteó mediante herramientas estándar del ecosistema Python, facilitando la manipulación, serialización y almacenamiento de la información adquirida.

6.3. Selección y justificación de componentes del sistema

Con el fin de satisfacer los requerimientos funcionales y no funcionales del proyecto, se realizó un proceso de selección de los principales componentes del sistema a partir de criterios técnicos y de integración. En esta etapa se evaluaron variables como rango de medida, precisión, inmunidad al ruido, tipo de comunicación, complejidad de cableado, disponibilidad de interfaces, número de GPIO, memorias, costo y compatibilidad con una arquitectura modular.

Este análisis permitió no solo comparar alternativas, sino también justificar la elección de los elementos más adecuados para una solución académica de bajo costo orientada a la adquisición y visualización de temperatura en un banco de transferencia de calor.

6.3.1. Selección del sensor

La selección del sensor de temperatura se realizó a partir de una matriz comparativa que incluyó variables como rango de operación, precisión o resolución, inmunidad al ruido, tipo de comunicación, complejidad de cableado, número de pines requeridos y costo. Bajo estos criterios se analizaron alternativas como PT100 con MAX31865, termocupla tipo K con MAX6675, DS18S20, MAX31888 y DS18B20. Los resultados de la comparación mostraron que el DS18B20 ofrecía el mejor equilibrio entre facilidad de integración, inmunidad al ruido, simplicidad de cableado y costo, obteniendo el mayor puntaje dentro de la matriz de decisión.

La elección del DS18B20 se justifica además por su comunicación 1-Wire, que permite conectar múltiples sensores sobre un mismo bus de datos, reduciendo significativamente la complejidad del cableado en comparación con otras alternativas. Asimismo, su identificador único de 64 bits facilita la identificación individual de cada sensor dentro del arreglo de medición, aspecto importante para

la trazabilidad, el mantenimiento y la asignación de posiciones de sensado en el banco térmico. En conjunto, estas características lo convierten en una opción adecuada para una solución académica de bajo costo que requiere múltiples puntos de medición y una integración práctica con el sistema embebido [25].

De manera complementaria, durante la selección del sensor se consideraron criterios como rango de medida, precisión, tipo de comunicación, inmunidad al ruido o interferencia, compatibilidad con el sistema de adquisición, interfaz física o cableado, número de pines requeridos y costo. Estos criterios permitieron orientar la evaluación hacia una alternativa que no solo ofreciera buen desempeño técnico, sino que también facilitara la implementación real del sistema dentro del banco térmico.

Tabla 6.6: Comparación de sensores de temperatura

Sensor	Tipo	Rango	Precisión	Comunicación	Ruido	Adquisición	Interfaz	conexión	N.º de pines	Costo (COP)
PT100 – Omega (RTD)	Resistivo	-200 °C a 850 °C	±0,1 °C	Analogico	Media	ADC con circuito RTD	Placa + pines Du-pont	2, 3 o 4 hilos RTD	3–4	MAX31865: \$34.000; PT100: \$11.000
MAX6675 + termocupla tipo K – Mactrónica	Termop tipo K	0 °C a 1024 °C	±1,5 °C (0–700 °C)	SPI (digital)	Alta	Interno en módulo MAX6675	Módulo a pines	Cable mallado / blindado	3 (GND, VCC, SIG)	MAX6675: \$13.500; termocupla: \$9.500
DS18B20 – Analog Devices (Maxim)	Digital	-55 °C a +125 °C	±0,5 °C	1-Wire	Baja	Conversión interna digital	Pin o módulo	Cable UTP / Dupont	3 (comparten pin de datos)	Sonda: \$10.000; módulo: \$5.500
DS18S20 – Analog Devices (Maxim)	Digital	-55 °C a +125 °C	±0,5 °C	1-Wire	Baja	Conversión interna digital	Pin o módulo	Cable UTP / Dupont	3 (comparten pin de datos)	Sonda: \$10.000–\$55.110
MAX31888 – Analog Devices (Maxim)	Digital + RTD	-50 °C a 180 °C; hasta 600 °C con RTD	±0,1 °C a ±0,25 °C	I2C (digital)	Baja	Interfaz digital interna	Módulo o chip SMD	2 o 3 hilos (RTD)	4 (VCC, GND, SDA, SCL)	–

A partir de los criterios de selección (precisión, interferencia, integración y costos), se asignó una puntuación para obtener un veredicto comparativo.

Tabla 6.7: Matriz de decisión para selección del sensor

Sensor	Precisi	Interfere	Integra	Costo	Punt. (/8)	Veredicto
PT100 + MAX31865	✓	–	×	×	5	Alta precisión, pero costoso y complejo.
MAX6675 + termocupla tipo K	–	×	✓	✓	5	Aceptable, pero sensible al ruido.
DS18B20	✓	✓	✓	✓	8	Ideal para este proyecto.
DS18S20	–	✓	✓	✓	7	Menor resolución que el DS18B20.
MAX31888 + RTD	✓	✓	–	×	5	Excelente si hay presupuesto.

Criterio de puntuación: ✓ = 2 puntos, – = 1 punto y × = 0 puntos.

6.3.2. Selección del microcontrolador

La selección de la plataforma de procesamiento se llevó a cabo mediante una comparación entre diferentes alternativas de microcontroladores y placas de desarrollo, considerando conectividad, interfaces disponibles para sensores, memoria, número de GPIO, soporte de desarrollo y costo. Entre las opciones evaluadas se incluyeron ESP32, STM32, Raspberry Pi Pico W, BeagleBone Black, Raspberry Pi 4 y Arduino Nano 33 IoT. Aunque varias de estas plataformas presentaron puntajes altos en la matriz comparativa, la selección final del sistema no se basó únicamente en el puntaje global, sino en su pertinencia frente a la arquitectura propuesta y al tipo de prototipo que se buscaba implementar [11].

En este contexto, se adoptó el ATmega328PB como unidad de control del sistema, debido a que ofrece recursos suficientes para la lectura de sensores, el procesamiento de datos y la comunicación serial, manteniendo al mismo tiempo una complejidad moderada y una integración favorable dentro de una tarjeta dedicada. Su disponibilidad de periféricos como UART/USART, SPI e I2C, junto con su facilidad de programación a bajo nivel y su compatibilidad con una arquitectura modular de bajo costo, lo hicieron apropiado para un desarrollo académico centrado en control embebido, manufactura propia en PCB y depuración progresiva del firmware. De este modo, aunque otras plataformas podían ofrecer mayores prestaciones en conectividad o capacidad de cómputo, el ATmega328PB resultó más coherente con el alcance real del proyecto, la necesidad de simplicidad del

sistema y la intención de construir un prototipo funcional sobre hardware propio.

Adicionalmente, durante la selección del microcontrolador se tuvieron en cuenta criterios como tipo de conexión disponible, interfaces para sensores y comunicación, número de GPIO, arquitectura del núcleo, memorias Flash y RAM, buses integrados, herramientas de desarrollo compatibles y costo. Esta evaluación permitió identificar una opción que respondiera adecuadamente a las exigencias del sistema sin introducir complejidad innecesaria ni elevar el costo del prototipo.

Tabla 6.8: Parámetros de las plataformas/microcontroladores evaluados

Plataforma / MCU	Wi-Fi / Ethernet	Interfaz sensores	GPIO	Núcleo	Flash	RAM	Bus	IDE disponible	Costo (COP)
ESP32 DevKit V1 (DOIT)	Wi-Fi – Micro USB	I2C, SPI, UART, 1-Wire, ADC (12-bit)	~30	Tensilica Xtensa LX6 Dual-core (240 MHz)	4 MB externa	520 KB SRAM	32 bits	Arduino IDE, VS Code, ESP-IDF	12.644–27.000
STM32F103C8 (Blue Pill)	Requiere módulo – Micro USB	ADC (12-bit), SPI, I2C, UART, PWM	~37	ARM Cortex-M3 (72 MHz)	64 KB	20 KB SRAM	32 bits	STM32CubeIDE, Arduino IDE	20.900
STM32 Nucleo-F401RE	Requiere módulo – Micro USB	ADC (12-bit), SPI, I2C, UART, PWM	~50	ARM Cortex-M4 (84 MHz, FPU)	512 KB	96 KB SRAM	32 bits	STM32CubeIDE, MBED	17.900
Raspberry Pi Pico W	Wi-Fi – Micro USB	ADC (12-bit), SPI, I2C, UART	~28	ARM Cortex-M0+ Dual-core (133 MHz)	2 MB externa	264 KB SRAM	32 bits	Thonny, Arduino IDE, C SDK	33.900
BeagleBone Black	Ethernet, Wi-Fi mediante dongle	SPI, I2C, UART, ADC, GPIO	~65	ARM Cortex-A8 (1 GHz)	4 GB eMMC + microSD	512 MB DDR3	32 bits	Linux, VS Code, Cloud9	273.639
Raspberry Pi 4 Model B	Wi-Fi + Ethernet – USB-C	SPI, I2C, UART, GPIO (sin ADC nativo)	~26	ARM Cortex-A72 Quad-core (1.5 GHz)	microSD	2/4/8 GB LPDDR4	64 bits	Raspberry Pi OS, VS Code, Python	256.271
Arduino Nano 33 IoT	Wi-Fi – Micro USB	I2C, SPI, UART, ADC (12-bit), PWM	~20	ARM Cortex-M0+ (48 MHz)	256 KB	32 KB SRAM	32 bits	Arduino IDE	89.889
ATmega328P	No integrado; requiere interfaz externa	ADC (10-bit), SPI, I2C, UART, PWM, 1-Wire	~24	AVR 8-bit (16 MHz)	32 KB	2 KB SRAM	8 bits	MPLAB X IDE, Microchip Studio, Arduino IDE	Según proveedor

Tabla 6.9: Matriz de decisión para selección del microcontrolador/plataforma

MCU / Plataforma	Conector	Interfaz	Memoria	GPIOs	IDE	Punt. (/10)	Veredicto
ESP32 DevKit V1 (IoT)	✓	✓	✓	✓	✓	10	Recomendado: balance completo.
STM32F103C8T6 (Blue Pill)	×	✓	–	✓	✓	7	Bueno si se prioriza ADC y control.
STM32 Nucleo-F401RE	×	✓	✓	✓	✓	8	Potente; ideal si no se requiere Wi-Fi.
Raspberry Pi Pico W	✓	✓	✓	✓	✓	10	Ideal para educación e IoT básico.
BeagleBone Black	✓	✓	✓	✓	✓	10	Muy avanzado y costoso para este caso.
Raspberry Pi 4 Model B	✓	×	✓	✓	✓	8	Excesivo; no tiene ADC nativo.
Arduino Nano 33 IoT (SAMD21)	✓	✓	✓	–	✓	9	Muy bueno para prototipos compactos.
ATmega328PB	✓	✓	–	✓	✓	9	Seleccionado: adecuado por costo, control por registros y suficiencia para el prototipo.

6.3.3. Selección del sistema de comunicación

Para la comunicación entre módulos y la transferencia de datos hacia la etapa de visualización se evaluaron distintas alternativas de enlace, considerando criterios de distancia, inmunidad al ruido, facilidad de integración con el microcontrolador y adecuación a una arquitectura modular. Entre las opciones analizadas se contemplaron niveles lógicos tipo TTL/UART, RS-232, buses síncronos como I²C y SPI, y una solución diferencial basada en RS-485.

7

La comunicación TTL/UART presenta la ventaja de su implementación directa en microcontroladores y su baja complejidad de software, lo que la hace útil para pruebas locales o conexiones punto a punto a corta distancia. Sin embargo, al trabajar con señales no diferenciales y niveles lógicos referidos a tierra, su inmunidad al ruido es reducida y su uso resulta menos conveniente cuando se requiere interconectar varios módulos dentro de un entorno de laboratorio con cableado más extenso o presencia de interferencia electromagnética.

Por su parte, RS-232 constituye una interfaz serial clásica ampliamente documentada y apropiada

para enlaces punto a punto. No obstante, su naturaleza también limitada a conexiones entre dos nodos, junto con su menor flexibilidad para una arquitectura modular compartida, la hace menos adecuada para un sistema que busca escalabilidad mediante varias tarjetas conectadas a un *backplane* o bus común.

En cuanto a I²C, se trata de un bus de dos líneas muy eficiente para la interconexión de circuitos integrados en una misma tarjeta o en distancias cortas. Su ventaja principal radica en la simplicidad del cableado y en la posibilidad de direccionar múltiples dispositivos. Sin embargo, no fue considerado como la mejor alternativa para la comunicación principal entre módulos, debido a que está orientado a distancias reducidas y es más sensible a capacitancias parásitas, ruido y condiciones de cableado cuando se extiende fuera del ámbito estrictamente local de una PCB.

De forma similar, SPI ofrece alta velocidad de transferencia y una implementación sencilla para enlaces locales entre periféricos y microcontroladores. A pesar de ello, requiere varias líneas de interconexión y una lógica de selección adicional, lo cual incrementa el cableado a medida que aumenta el número de módulos. Por esta razón, SPI resulta más apropiado como bus interno de placa que como interfaz principal de comunicación entre tarjetas distribuidas.

Frente a estas alternativas, RS-485 resultó la opción más conveniente para el proyecto. Su señalización diferencial le proporciona mayor inmunidad al ruido eléctrico, mejor comportamiento ante diferencias de potencial de tierra y una mayor robustez para distancias superiores a las que normalmente admiten enlaces TTL directos. Además, su operación en configuración *half-duplex* permite compartir un mismo medio físico entre varios nodos, lo que se ajusta de forma natural al enfoque modular del sistema y a la posibilidad de expansión futura mediante nuevas tarjetas.

En consecuencia, se seleccionó una arquitectura de comunicación basada en UART a nivel del microcontrolador y RS-485 como medio físico de interconexión entre módulos, utilizando un transceptor dedicado. Esta solución permitió conservar la simplicidad de la transmisión serial asíncrona en el *firmware*, al tiempo que se mejoró la robustez del enlace en el entorno real de operación del banco de transferencia de calor [12, 32].

A partir de la comparación realizada, se resumieron las principales ventajas, limitaciones y el grado de adecuación de cada alternativa de comunicación para la arquitectura propuesta.

Tabla 6.10: Comparación de alternativas de comunicación evaluadas

Interfaz	Ventajas principales	Limitaciones principales	Adecuación al proyecto
TTL/UART	Implementación simple, bajo costo y compatibilidad directa con microcontroladores.	Baja inmunidad al ruido y poca robustez a mayor distancia.	Media
RS-232	Interfaz serial clásica y sencilla para comunicación punto a punto.	No adecuada para varios nodos y menor flexibilidad en una arquitectura modular.	Baja
I ² C	Poco cableado y direccionamiento de múltiples dispositivos.	Más adecuada para distancias cortas y comunicación local dentro de una placa.	Media-baja
SPI	Alta velocidad e implementación simple entre periféricos locales.	Requiere más líneas de interconexión y resulta menos conveniente al aumentar el número de módulos.	Baja
RS-485	Señalización diferencial, alta inmunidad al ruido y capacidad de comunicación con varios nodos.	Requiere control de dirección y un transceptor adicional.	Alta

6.4. Arquitectura modular del sistema

6.4.1. Descripción general de la arquitectura

Se definió una arquitectura modular basada en tarjetas con el propósito de aislar funciones críticas del sistema y facilitar su escalabilidad, mantenimiento e integración física. La solución se organizó en cuatro bloques principales: una tarjeta de alimentación (POWER), encargada de suministrar la energía requerida; una tarjeta de adquisición y control (DAQ), dedicada a la lectura de sensores y al procesamiento local; una tarjeta de comunicación (COM), orientada al enlace con la plataforma de visualización y registro; y un backplane, utilizado como estructura de interconexión entre los distintos módulos. Esta organización permite reducir el acoplamiento entre subsistemas, simplificar tareas de diagnóstico y posibilitar futuras ampliaciones sin necesidad de rediseñar completamente la solución [5][6][28].

Como referencia conceptual y mecánica se consideraron arquitecturas modulares inspiradas en sistemas tipo Eurocard/VME, en los que varias tarjetas comparten alimentación e interconexión mediante una base común. No obstante, el tamaño final de las tarjetas se ajustó al número real de componentes, a las restricciones de fabricación y a la necesidad de mantener un diseño compacto y de bajo costo, adecuado para un entorno académico. La adopción de esta arquitectura respondió, por tanto, tanto a criterios de modularidad como a consideraciones prácticas de manufactura e integración del prototipo [30][31][39].

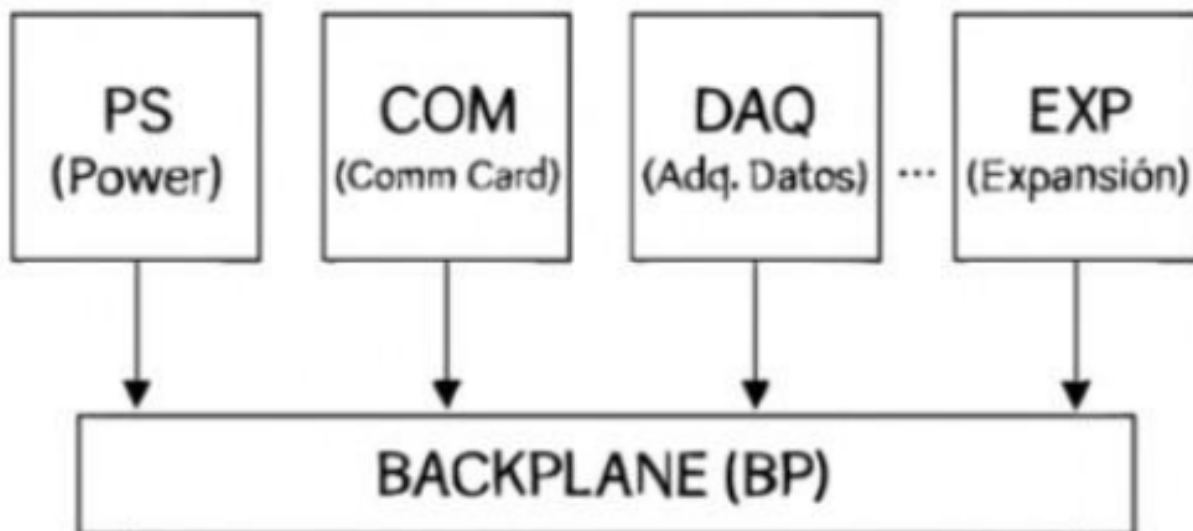


Figura 6.1: Estructura física modular del sistema (backplane y conectores de borde).

6.4.2. Tarjeta de alimentación (POWER)

La tarjeta de alimentación, identificada dentro del sistema como módulo POWER o ALI, fue concebida como el bloque encargado de recibir la energía desde la red eléctrica y distribuir, a través del *backplane*, los niveles de tensión requeridos por las demás tarjetas del prototipo. Su incorporación como módulo independiente respondió a la necesidad de desacoplar la función de potencia de las etapas de adquisición y comunicación, mejorando la organización del sistema, facilitando el mantenimiento y reduciendo interferencias entre subsistemas sensibles.

Desde el punto de vista funcional, esta tarjeta recibe una entrada de corriente alterna (AC) proveniente del tomacorriente, la cual es aplicada a un módulo de conversión AC/DC integrado en la propia placa. A partir de esta conversión se obtienen los rieles de tensión necesarios para la arquitectura modular propuesta. En la implementación desarrollada, la tarjeta fue diseñada para distribuir principalmente niveles de 7.5 V, 12 V y GND hacia el *backplane*, de manera que cada módulo conectado pudiera disponer de una alimentación común y ordenada según su función dentro del sistema [42].

Esta decisión de diseño permitió que la energía no tuviera que cablearse de forma individual hacia cada tarjeta, sino que se repartiera mediante el conector de borde y el *backplane* pasivo. De esta forma, la tarjeta POWER asumió el papel de fuente centralizada del sistema, mientras que los demás módulos quedaron enfocados en sus tareas específicas: adquisición de temperatura en la tarjeta DAT y transmisión/interfaz con PC en la tarjeta COM. Esta separación modular es coherente con la filosofía de arquitecturas tipo portadora 3U, donde cada tarjeta cumple una función concreta

y el *backplane* se encarga de la interconexión física y eléctrica [7, 8].

En términos eléctricos, la etapa principal de la tarjeta POWER se basó en un módulo de fuente conmutada AC/DC de baja potencia, suficiente para alimentar el prototipo y sus cargas asociadas. A partir de la información de diseño utilizada en el proyecto, se contempló la disponibilidad de salidas nominales de 7.5 V y 12 V, adecuadas para alimentar la electrónica del sistema o servir como tensión de entrada para etapas posteriores de regulación local en otras tarjetas. En particular, este enfoque resulta útil cuando algunas placas requieren un riel intermedio de entrada y posteriormente generan de forma local una tensión regulada de 5 V para el microcontrolador y periféricos. Así, la regulación principal se centraliza en la tarjeta POWER, mientras que la adaptación final puede realizarse cerca de la carga, reduciendo problemas de caída de tensión y mejorando la integridad de alimentación [11, 42].

Además de la conversión de energía, la tarjeta incorpora elementos auxiliares orientados a la operación segura y a la verificación del estado de la alimentación. Entre ellos se encuentran indicadores luminosos de presencia de tensión, conexiones definidas hacia el *backplane* y una organización de salidas que diferencia claramente los rieles de potencia de las señales de comunicación. Esta separación es importante porque evita confusiones durante el ensamble y mejora la mantenibilidad del sistema, especialmente en un entorno académico donde el prototipo puede ser manipulado en repetidas ocasiones.

Otro aspecto relevante es que la tarjeta POWER no fue planteada únicamente como un bloque de suministro, sino también como una pieza de integración dentro del enfoque escalable del prototipo. Al reservar posiciones adicionales en el *backplane* para futuras expansiones, la existencia de una tarjeta de alimentación dedicada permite prever la incorporación de nuevos módulos sin rediseñar por completo la distribución energética. En consecuencia, la tarjeta contribuye no solo al funcionamiento actual del sistema, sino también a su capacidad de crecimiento [7, 8].

En síntesis, la tarjeta de alimentación constituye el punto de entrada y acondicionamiento energético del prototipo. Su función principal es convertir la energía de entrada a niveles útiles para el sistema, distribuirlos a través del *backplane* y proporcionar una base eléctrica organizada para la operación conjunta de los módulos. Desde la perspectiva metodológica, esta tarjeta fue fundamental para consolidar la arquitectura modular, al separar claramente las funciones de potencia, adquisición y comunicación dentro del banco de adquisición de temperatura.



Figura 6.2: Diagrama de bloques de la tarjeta POWER

Tabla 6.11: Resumen funcional de la tarjeta POWER

Parámetro	Descripción
Función principal	Conversión y distribución de energía para el sistema modular.
Entrada	Alimentación AC desde tomacorriente.
Conversión principal	Conversión AC/DC mediante módulo de potencia integrado.
Salidas distribuidas	7.5 V, 12 V y GND.
Medio de distribución	Conector de borde (<i>edge connector</i>) y <i>backplane</i> pasivo.
Módulos alimentados	DAT, COM y futuras expansiones.
Potencia de referencia	15 W por módulo de conversión, según la variante empleada.
Corriente teórica máxima	2.0 A a 7.5 V y 1.25 A a 12 V.
Objetivo del diseño	Separar la etapa de potencia de las funciones de adquisición y comunicación, y facilitar la expansión del sistema.

6.4.3. Tarjeta de adquisición/control (DAQ)

La tarjeta de adquisición/control (DAQ) fue diseñada como el módulo encargado de realizar la lectura de temperatura, el procesamiento local de los datos y la preparación de la información para su envío hacia la tarjeta de comunicación. Dentro de la arquitectura modular propuesta, esta tarjeta concentra las funciones de sensado y control lógico, por lo que constituye el núcleo operativo del sistema.

Desde el punto de vista de *hardware*, la tarjeta integra un microcontrolador ATmega328PB, seleccionado por su disponibilidad de periféricos seriales, capacidad de procesamiento suficiente para el manejo de múltiples sensores y facilidad de implementación en un diseño embebido de complejidad moderada [11]. Sobre esta base se estructuró una solución orientada a la lectura de sensores digitales DS18B20 por bus 1-Wire, el control de señales de estado y la comunicación con otros módulos del sistema.

La tarjeta DAQ incorpora además un conjunto de elementos auxiliares que fortalecen su operación práctica dentro del prototipo. Entre ellos se encuentran LEDs de indicación para el estado general del sistema y para la detección individual de sensores, lo que permite una verificación visual rápida durante las pruebas de laboratorio y facilita la identificación de fallas de conexión o ausencia de dispositivos. De igual forma, la tarjeta cuenta con un puerto de programación *In-Circuit* (ICSP), que permite cargar y actualizar el *firmware* directamente sobre el microcontrolador sin necesidad de desmontarlo de la placa, mejorando la mantenibilidad y agilizando las tareas de depuración.

En la etapa de sensado, la tarjeta fue preparada para trabajar con varios sensores DS18B20, los cuales ofrecen comunicación digital por 1-Wire, identificador único de 64 bits y capacidad de conexión multipunto sobre un mismo bus, características especialmente convenientes para un arreglo multisensor como el implementado en este proyecto [10]. A partir de las lecturas obtenidas, el microcontrolador ejecuta el procesamiento básico necesario para organizar la información, asociarla al sensor correspondiente y dejarla lista para la transmisión al módulo de comunicación.

Adicionalmente, la tarjeta contempla la interfaz eléctrica necesaria para enlazarse con el sistema de comunicación basado en RS-485, mediante un transceptor dedicado. Esta integración permite que la lógica de adquisición permanezca separada de la capa física de transmisión, manteniendo modularidad entre las funciones de lectura, control y comunicación. En este sentido, la tarjeta DAQ no solo adquiere datos, sino que actúa como punto de coordinación entre el subsistema de sensores y el canal de intercambio con el resto del sistema [12].

En conjunto, la tarjeta DAQ fue concebida como un módulo funcionalmente completo dentro de la arquitectura propuesta: recibe la información de los sensores, la procesa localmente, proporciona realimentación visual por medio de LEDs, permite actualización de *firmware* mediante programación *In-Circuit* y entrega los datos al subsistema de comunicación. Esta organización refuerza la filosofía modular del prototipo y facilita tanto la validación experimental como futuras ampliaciones del sistema.

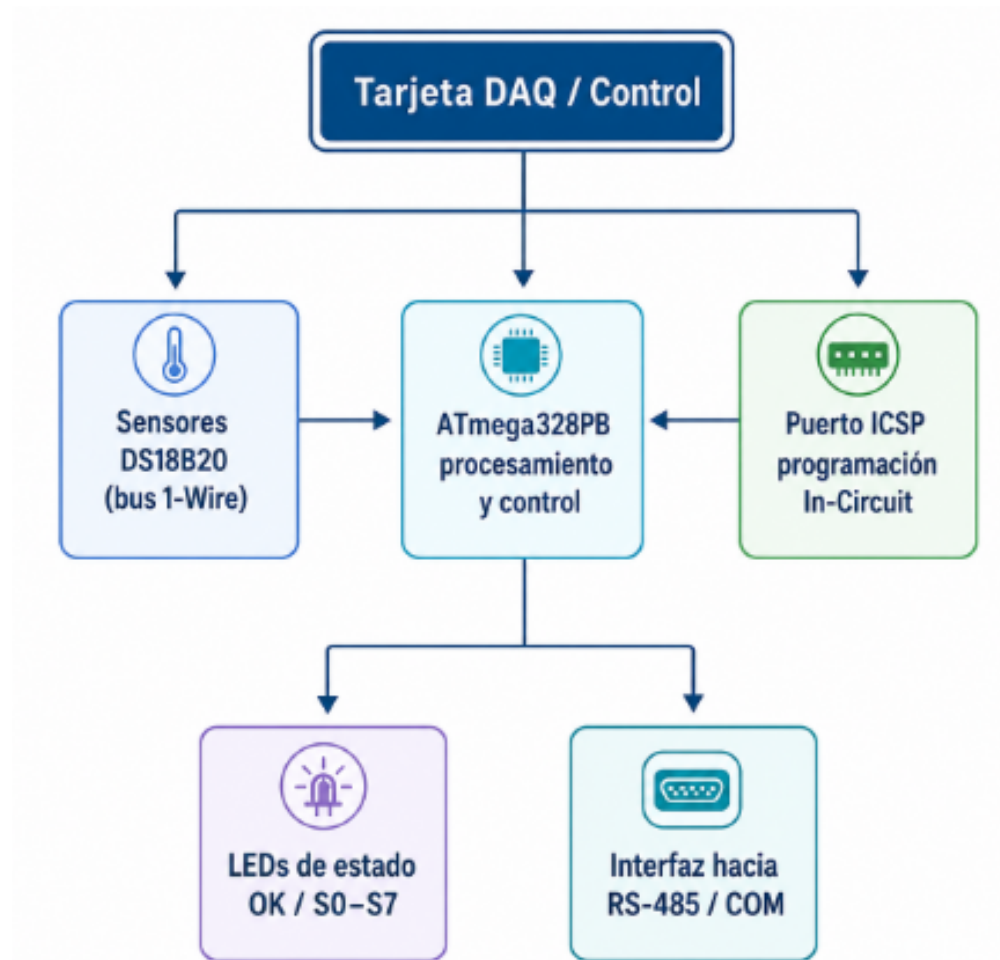


Figura 6.3: Diagrama de bloques de la tarjeta DAQ

6.4.4. Tarjeta de comunicación (COM)

La tarjeta de comunicación (COM) fue diseñada como el módulo encargado de enlazar el sistema embebido con el computador y de gestionar el intercambio de datos entre la red interna del prototipo y la plataforma de visualización. Dentro de la arquitectura modular propuesta, esta tarjeta cumple la función de interfaz entre el dominio de adquisición local y el entorno externo de monitoreo, permitiendo que la información generada por la tarjeta DAQ pueda ser recibida, interpretada y presentada en tiempo real.

Desde el punto de vista funcional, la tarjeta COM integra tres bloques principales: una etapa de procesamiento y control basada en el microcontrolador ATmega328PB, una interfaz de comunicación diferencial para el enlace con el *backplane* y una interfaz USB-serial para la conexión con el

computador [11, 12]. Esta organización permitió separar con claridad la lógica de control, la capa física de transmisión entre módulos y la salida final hacia la plataforma de *software*, manteniendo coherencia con el enfoque modular del sistema.

Para la comunicación interna del prototipo, la tarjeta incorpora un transceptor RS-485 THVD1400, seleccionado por su operación en entornos de 3.3 V a 5 V, su adecuada inmunidad al ruido y su conveniencia para arquitecturas multipunto o distribuidas [12]. Esta etapa recibe la información proveniente del *backplane* y permite mantener una transmisión más robusta que un enlace TTL directo, especialmente en una estructura donde la alimentación y la comunicación se distribuyen entre varios módulos [13, 14]. De esta manera, la tarjeta COM actúa como el punto de acceso del sistema al bus de comunicación modular.

Hacia el computador, la tarjeta dispone de una interfaz USB, implementada mediante un convertidor USB-serial, lo que facilita la conexión directa con el PC sin requerir *hardware* externo adicional. Gracias a esta integración, los datos recibidos desde la red interna pueden presentarse como un puerto serial virtual, el cual es utilizado posteriormente por la plataforma de visualización desarrollada en Python y por herramientas de monitoreo como MPLAB Data Visualizer. Esta decisión simplificó la etapa experimental, ya que permitió emplear una conexión ampliamente disponible en equipos de cómputo y compatible con las rutinas de adquisición implementadas en *software*.

Adicionalmente, la tarjeta COM incorpora elementos auxiliares orientados a la validación y al diagnóstico del sistema, entre ellos LEDs de indicación asociados a estados de alimentación y actividad de comunicación, así como conectores de programación tipo ICSP y elementos de control local como el circuito de *reset*. Estos recursos resultaron especialmente útiles durante las etapas de prueba, porque permitieron verificar de forma visual la presencia de alimentación, la actividad de transmisión/recepción y el correcto comportamiento del *firmware* sin depender exclusivamente del *software* de visualización.

En conjunto, la tarjeta de comunicación fue concebida como el puente entre la arquitectura modular del prototipo y la capa de supervisión externa. Su función no se limita a transmitir datos, sino que también organiza el enlace entre el bus RS-485, la interfaz USB hacia el PC y las señales auxiliares de diagnóstico. Por ello, esta tarjeta constituye un elemento clave para la integración del sistema completo, ya que hace posible que las mediciones adquiridas por la tarjeta DAQ se conviertan en información utilizable dentro de la plataforma de registro, exportación y visualización en tiempo real.



Figura 6.4: Diagrama de bloques de la tarjeta COM

6.4.5. Tarjeta de BACKPLANE

La tarjeta de *backplane* fue diseñada como el elemento de interconexión pasiva del sistema modular, encargado de distribuir tanto la alimentación como las señales de comunicación entre las distintas tarjetas insertadas en la estructura. A diferencia de los módulos funcionales, el *backplane* no realiza procesamiento ni conversión de datos, sino que actúa como una plataforma común de conexión eléctrica y mecánica para los módulos ALI/POWER, DAQ, COM y las posibles expansiones futuras.

Desde el punto de vista estructural, el *backplane* organiza el sistema en varias posiciones o *slots*, cada una asociada a un conector de borde compatible con las tarjetas desarrolladas. Esta disposición permite que los módulos compartan una infraestructura común, simplificando el montaje, facilitando el reemplazo de tarjetas y manteniendo la posibilidad de crecimiento del sistema sin modificar la base de interconexión.

El aspecto más importante del *backplane* dentro del prototipo es la distribución de sus buses principales, los cuales se dividieron en dos grupos: bus de alimentación y bus de comunicación. El bus de alimentación se encarga de repartir los niveles de tensión generados en la tarjeta POWER hacia las demás tarjetas del sistema, principalmente 7.5 V, 12 V y GND. De esta forma, la energía no se cablea de manera independiente a cada módulo, sino que se entrega a través de pistas comunes en el *backplane* y de los conectores de borde de cada ranura.

Por otra parte, el bus de comunicación permite enlazar lógicamente los módulos que intercambian datos dentro del sistema. En la arquitectura desarrollada, este bus se asoció a las líneas de comunicación serial implementadas sobre la interfaz RS-485, de modo que el *backplane* actúa como medio físico interno para distribuir las señales necesarias entre la tarjeta de adquisición y la tarjeta de comunicación. Esta separación entre potencia y comunicación fue relevante para mantener mayor orden en el diseño y reducir ambigüedades durante el montaje y la validación experimental.

Además de los buses principales, el *backplane* contempla una asignación organizada de contactos en cada conector, lo que permite conservar compatibilidad entre módulos y evitar errores de inserción o conexión. En este sentido, el diseño del *backplane* no solo responde a una necesidad eléctrica, sino también a una lógica de estandarización interna del prototipo, donde cada ranura comparte una misma base física de alimentación y comunicación.

En conjunto, la tarjeta de *backplane* constituye la base de integración del sistema modular. Su función principal es distribuir los rieles de alimentación y las líneas de comunicación entre tarjetas, manteniendo una arquitectura ordenada, escalable y coherente con el enfoque tipo portadora 3U adoptado como referencia conceptual.

Tabla 6.12: Resumen funcional de la tarjeta BACKPLANE

Bus	Señales principales	Función
Bus de alimentación	7.5 V, 12 V, GND	Distribuir energía desde la tarjeta POWER hacia los módulos.
Bus de comunicación	A, B, GND de referencia	Permitir el intercambio de datos entre módulos mediante RS-485.
Bus mecánico de inserción	Conector de borde por <i>slot</i>	Asegurar la interconexión física y la compatibilidad entre tarjetas.

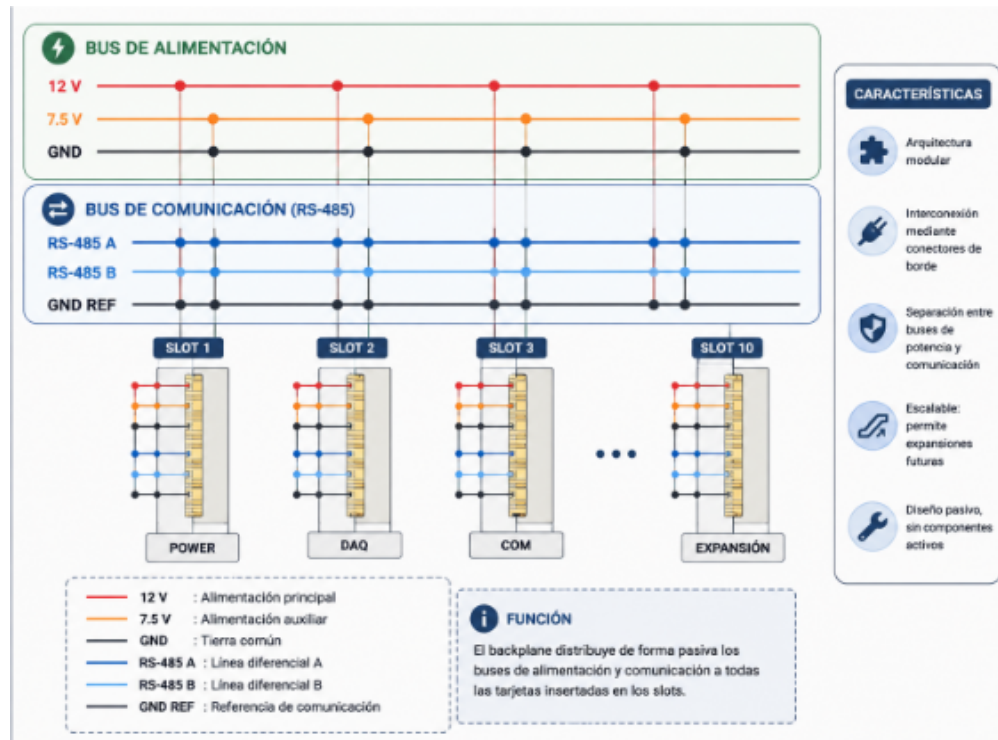


Figura 6.5: Arquitectura lógica de adquisición

6.5. Desarrollo del firmware

El desarrollo del *firmware* se llevó a cabo de forma incremental, con el propósito de validar progresivamente la lectura de temperatura, el comportamiento del bus 1-Wire y la comunicación del sistema. En una primera etapa se realizaron pruebas preliminares con sensores equivalentes o compatibles, empleando código de demostración y bibliotecas de apoyo como OneWire y Dallas-Temperature, con el fin de verificar la detección de sensores, la adquisición básica de datos y el comportamiento del bus en condiciones reales de cableado. Esta fase inicial permitió confirmar la viabilidad de la lectura por 1-Wire, la estabilidad de la medición, el comportamiento del *pull-up* y la forma general de los datos esperados para la visualización [10].

Posteriormente, el desarrollo migró hacia una implementación con mayor control temporal y menor dependencia de librerías externas, priorizando el determinismo del sistema, la robustez de la lectura y una mejor adaptación a la arquitectura final del *hardware*. De esta manera, el *firmware* evolucionó desde una validación funcional preliminar hacia una solución propia, coherente con los requerimientos del proyecto y con la necesidad de integrar adquisición, diagnóstico y transmisión de datos dentro de una misma plataforma embebida.

La versión final del *firmware* se desarrolló en lenguaje C, debido a su cercanía con el *hardware* y a la posibilidad de realizar configuración directa de registros, temporización precisa y control

detallado de periféricos. El entorno principal de desarrollo fue MPLAB X IDE, utilizando la cadena de compilación AVR-GCC para el microcontrolador ATmega328PB. Esta combinación permitió estructurar el proyecto en archivos fuente específicos para adquisición y comunicación, compilar el código, depurarlo y cargarlo al dispositivo de forma consistente con la arquitectura embebida adoptada.

Para la programación del microcontrolador se empleó la interfaz *In-Circuit Serial Programming* (ICSP) incluida en la tarjeta, lo que permitió grabar y actualizar el *firmware* directamente sobre la placa sin desmontar el circuito. Durante las pruebas también se utilizaron herramientas de monitoreo serial, particularmente la terminal y MPLAB Data Visualizer, con el fin de observar la salida de depuración, verificar la recepción de tramas y comprobar el comportamiento de las rutinas implementadas. En conjunto, estas herramientas facilitaron la validación progresiva del *firmware* y la integración entre la etapa de adquisición, la comunicación serial y la plataforma de visualización.

6.5.1. Lógica general del firmware

El *firmware* del sistema se estructuró en dos módulos principales e independientes, implementados sobre microcontroladores ATmega328PB operando a 16 MHz: un módulo de adquisición, correspondiente a la tarjeta esclava, y un módulo de comunicación, correspondiente a la tarjeta maestra. Esta organización permitió separar las funciones de lectura de sensores, procesamiento local, validación y transmisión de datos, facilitando la depuración del sistema y manteniendo coherencia con la arquitectura modular del *hardware*.

El módulo esclavo, asociado a la placa de adquisición, se encarga de ejecutar el proceso de lectura de temperatura desde los sensores DS18B20 mediante el protocolo 1-Wire, almacenar los valores obtenidos y responder a las solicitudes provenientes del maestro a través del enlace RS-485. Además de la adquisición periódica de datos, este módulo mantiene mecanismos de diagnóstico y actualización de estados, lo que le permite conservar una operación continua aun cuando la comunicación con el maestro ocurra de manera asíncrona respecto del ciclo de medición.

Por su parte, el módulo maestro, implementado en la placa de comunicación, controla el intercambio de información dentro del sistema. Su función principal consiste en enviar solicitudes periódicas al esclavo, recibir la trama de datos, verificar su integridad y transferir la información validada hacia el computador en un formato legible para la plataforma de visualización y registro. De esta manera, el maestro concentra la lógica de consulta, recepción y envío externo de la información, mientras que el esclavo concentra la lógica de adquisición local.

En términos de operación general, el sistema sigue una lógica maestro-esclavo en la que la tarjeta de adquisición actualiza periódicamente las temperaturas medidas, mientras que la tarjeta de comunicación solicita esa información a intervalos regulares y la transmite al PC. Este diseño permite desacoplar el proceso de sensado del proceso de visualización, mejorar la organización del *firmware* y mantener una estructura de funcionamiento clara dentro del prototipo desarrollado.

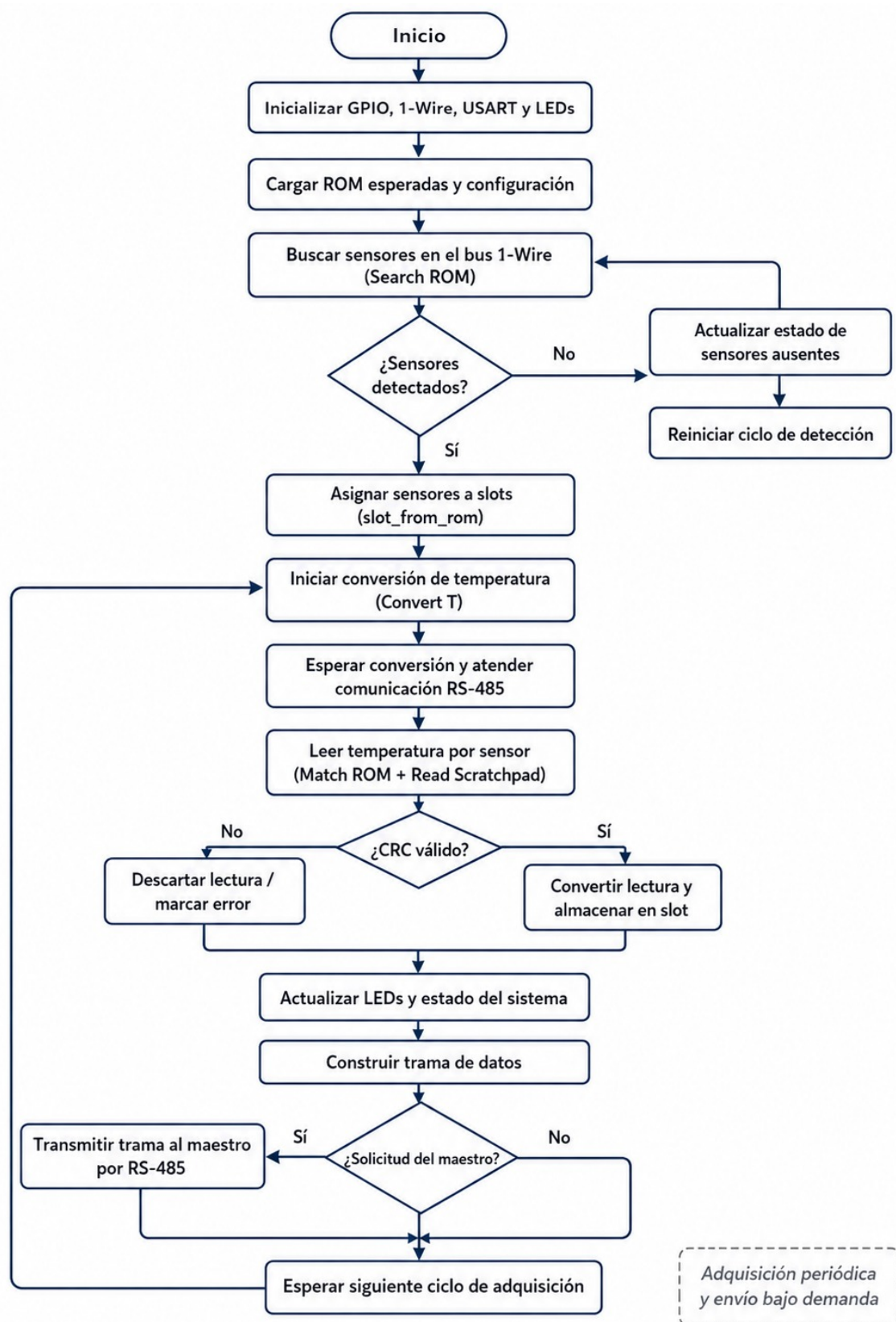


Figura 6.6: Diagrama de flujo resumido del firmware para lectura DS18B20 (1-Wire).

6.5.2. Implementación del protocolo 1-Wire con DS18B20

Para lograr un control más preciso de la interacción con los sensores DS18B20, el firmware incorporó rutinas desarrolladas a partir de la información técnica suministrada en el *datasheet* del dispositivo. Estas rutinas permiten implementar de forma explícita las operaciones fundamentales del protocolo 1-Wire, incluyendo la secuencia de *reset* y detección de presencia, la escritura y lectura de bits, la transmisión de bytes y el direccionamiento de dispositivos mediante comandos ROM [25].

La comunicación con los sensores se realizó sobre el pin PD3 del ATmega328PB, a través de primitivas de bajo nivel como `ow_reset()`, `ow_write_bit()`, `ow_read_bit()`, `ow_write_byte()` y `ow_read_byte()`. A partir de estas funciones se construyó la lógica necesaria para descubrir sensores conectados al mismo bus, seleccionar cada dispositivo de manera individual y ejecutar la secuencia de medición de temperatura.

En particular, el firmware empleó los comandos ROM del protocolo 1-Wire para soportar un esquema de lectura multisensor. Durante la fase de detección se utilizó el comando *Search ROM* (0xF0), que permitió identificar los dispositivos presentes en el bus y recuperar su código ROM único de 64 bits. Posteriormente, para la lectura de cada sensor, se empleó el comando *Match ROM* (0x55) junto con la dirección específica del dispositivo, lo que permitió ejecutar mediciones individualizadas y asociar cada lectura a una posición lógica dentro del sistema.

Una vez seleccionado el sensor, la secuencia de adquisición incluyó el envío del comando *Convert T* (0x44) para iniciar la conversión interna de temperatura y, tras el tiempo de espera correspondiente, el comando *Read Scratchpad* (0xBE) para recuperar los datos almacenados por el dispositivo. Este procedimiento permitió obtener el valor de temperatura de manera organizada y compatible con el esquema de direccionamiento adoptado para múltiples sensores.

Con el fin de simplificar la lógica del firmware y concentrar el desarrollo en las funciones realmente necesarias para el proyecto, se optó por implementar únicamente el subconjunto mínimo de comandos requerido para la detección, conversión y lectura de temperatura, evitando incorporar funciones no utilizadas que aumentarían la complejidad del sistema. Esta decisión favoreció una mejor adaptación del protocolo a las necesidades reales del banco de transferencia de calor y facilitó la integración del sensado con el resto de la arquitectura embebida.

Como apoyo conceptual para esta implementación, se tomaron como referencia los diagramas de flujo de comandos ROM y los esquemas funcionales del protocolo 1-Wire presentados en la documentación del DS18B20. La Figura 6.7 resume la lógica general de los comandos ROM empleados para el direccionamiento y selección de dispositivos dentro del bus.

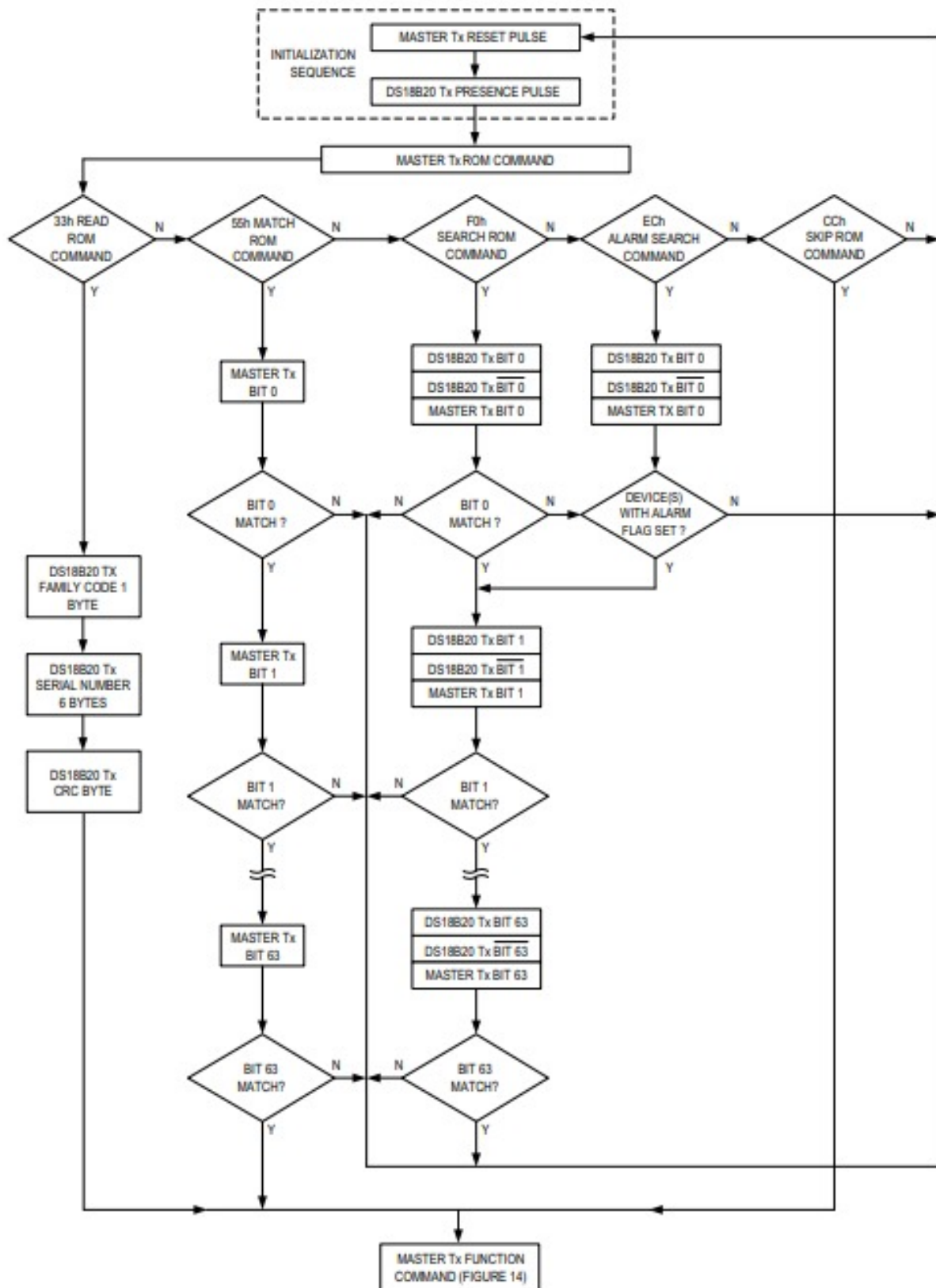


Figura 6.7: Diagrama de comandos ROM en 1-Wire

6.5.3. Temporización del bus 1-Wire

Una vez definida la secuencia de comandos ROM y de función empleada para interactuar con los sensores DS18B20, fue necesario garantizar el cumplimiento de las restricciones temporales propias del protocolo 1-Wire. A diferencia de otros esquemas de comunicación más tolerantes, el correcto funcionamiento de este bus depende directamente de respetar intervalos específicos en las operaciones de *reset*, detección de presencia, escritura y lectura de bits. Por esta razón, la temporización se trató como un aspecto central dentro del desarrollo del firmware, y no como un detalle secundario de implementación.

Para ello, se utilizaron las funciones `_delay_us()` y `_delay_ms()` de AVR-libc, ajustadas a una frecuencia de operación de 16 MHz en el ATmega328PB. Los tiempos adoptados se tomaron de los valores de referencia indicados en el *datasheet* del DS18B20, incluyendo la duración mínima del pulso de *reset*, el tiempo de espera para detección de presencia, la duración de los *slots* de escritura para los bits lógicos '0' y '1', y el instante de muestreo durante la lectura. Esto permitió generar señales compatibles con el estándar Dallas/Maxim y mantener una interacción estable con los sensores incluso en condiciones reales de cableado y con varios dispositivos conectados al mismo bus.

El control preciso de estos intervalos mejoró la confiabilidad del sistema, especialmente en la fase de lectura multisensor. En particular, se aseguró que las rutinas de *reset* y presencia permitieran verificar correctamente la existencia de dispositivos en el bus, mientras que los tiempos de escritura y lectura garantizaron que cada bit fuera interpretado adecuadamente por el microcontrolador y los sensores. De esta forma, la temporización se convirtió en un elemento esencial para la repetibilidad de las mediciones y para la estabilidad general del proceso de adquisición.

La temporización del bus 1-Wire se implementó mediante una lógica no bloqueante durante la conversión de temperatura del DS18B20. En lugar de aplicar un único retardo prolongado, el firmware fragmentó el tiempo de conversión en intervalos cortos de espera, dentro de los cuales también atendía eventos de comunicación RS-485. De este modo, mientras el sensor completaba internamente la conversión térmica, la placa de adquisición conservaba capacidad de respuesta frente al maestro. Esta estrategia mejoró la integración entre sensado y comunicación, sin comprometer los tiempos críticos requeridos por el protocolo 1-Wire.

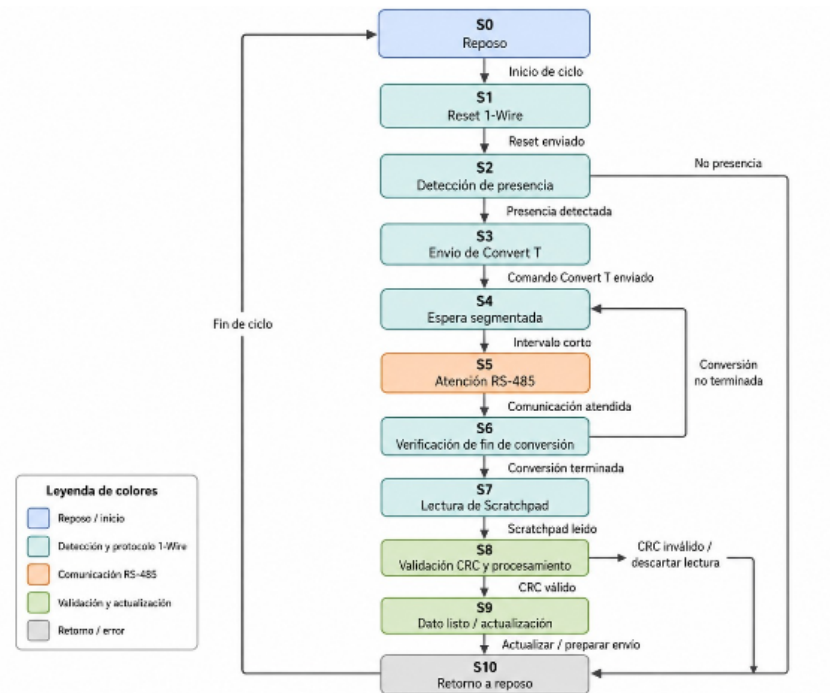


Figura 6.8: Máquina de estados para la temporización

La Figura 6.9 ilustra los principales diagramas temporales utilizados como referencia para la implementación del protocolo, incluyendo los *slots* de escritura y lectura, el pulso de *reset* y la ventana de detección de presencia [10].

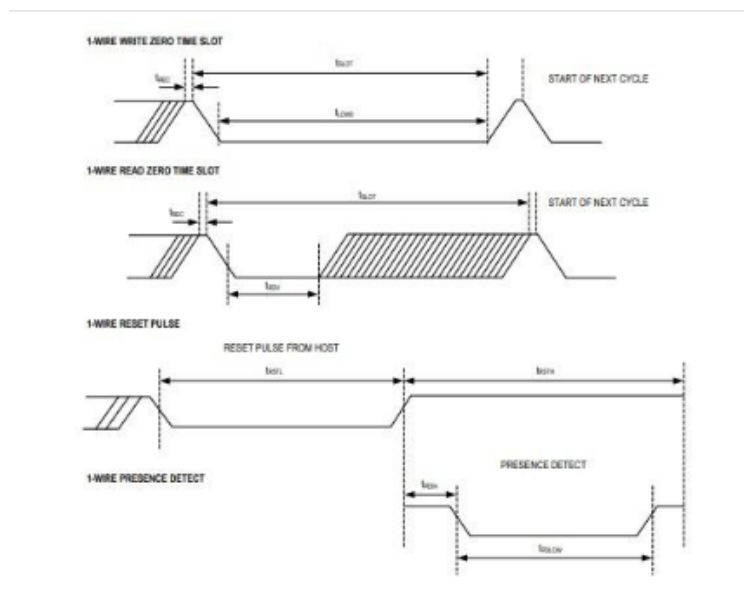


Figura 6.9: Diagramas de temporización 1-Wire

6.5.4. Direccionamiento de sensores

Una vez establecida la secuencia de lectura y la temporización requerida por el protocolo 1-Wire, fue necesario definir un mecanismo que permitiera identificar de forma estable cada sensor dentro del arreglo multisensor. Para ello, el firmware aprovechó una de las características fundamentales del DS18B20: cada dispositivo posee un código ROM único de 64 bits, compuesto por un byte de familia, un número de serie y un byte de verificación CRC. Esta propiedad permitió distinguir individualmente los sensores conectados al mismo bus y asociar cada lectura a una posición lógica específica dentro del sistema.

Con base en esta lógica, el firmware implementó un esquema de direccionamiento por *slots*, donde cada posición representa un canal lógico de medición. Para soportar este mecanismo, se utilizaron dos tablas de referencia: `expected_roms`, almacenada como base fija de identificación, y `runtime_expected_roms`, empleada como copia de trabajo modificable durante la ejecución. Esta organización permitió conservar una correspondencia estable entre sensores físicos y posiciones lógicas, incluso cuando el orden de conexión sobre el bus no coincidía con el orden deseado en la adquisición.

Durante el barrido del bus mediante el comando *Search ROM* (0xF0), el firmware detecta los dispositivos presentes y compara cada código ROM encontrado con las entradas almacenadas en `runtime_expected_roms`. Esta comparación se realiza mediante la función `slot_from_rom()`, cuya finalidad es determinar a qué *slot* pertenece cada sensor reconocido. Si existe coincidencia, la temperatura medida se almacena en la posición correspondiente del arreglo global de temperaturas; en caso contrario, el dispositivo se descarta para evitar que sensores no previstos alteren la organización de las lecturas. De igual forma, los *slots* sin definición válida pueden mantenerse vacíos sin afectar el funcionamiento general del sistema.

Este mecanismo de direccionamiento resultó especialmente importante para el proyecto porque permitió mantener una organización consistente de la información adquirida, independientemente de la disposición física de los sensores en el bus. Gracias a ello, el sistema no solo pudo leer múltiples dispositivos de manera ordenada, sino también preparar las mediciones para su posterior validación y procesamiento. En este punto, la identificación correcta de cada sensor se complementa de forma natural con la verificación de integridad de los datos, aspecto que se desarrolla en la siguiente subsección mediante el uso de CRC.

6.5.5. Verificación CRC

Una vez completada la lectura del sensor y la recuperación de los datos almacenados en su memoria interna, el firmware incorporó mecanismos de verificación orientados a garantizar la integridad de la información adquirida. En este contexto, la validación mediante CRC constituyó un elemento fundamental para aumentar la confiabilidad del sistema, especialmente al operar con múltiples sensores conectados al mismo bus 1-Wire.

La obtención de la temperatura se realizó mediante la lectura del *scratchpad* del DS18B20, donde se almacenan tanto los bytes asociados al valor medido como el byte de verificación CRC.

A partir de estos datos, el firmware calculó el CRC correspondiente y lo comparó con el valor recibido desde el sensor. Esta comprobación permitió reducir la probabilidad de aceptar lecturas corruptas ocasionadas por interferencias eléctricas, errores de temporización o fallas transitorias en la comunicación sobre el bus [10].

Además de la validación del *scratchpad*, el esquema de verificación también resultó coherente con la lógica de direccionamiento y descubrimiento de sensores descrita en los apartados anteriores, ya que el uso de CRC contribuye a reforzar la consistencia de la información intercambiada entre el microcontrolador y los dispositivos conectados al bus. De esta manera, el procesamiento de los datos no se limitó a extraer el valor de temperatura, sino que incorporó un criterio de aceptación que fortaleció la robustez general del firmware.

En el contexto del proyecto, esta validación fue especialmente importante porque el sistema debía mantener un nivel adecuado de confiabilidad en la adquisición, aun cuando existieran varios sensores en operación simultánea. Por ello, la rutina de CRC se constituyó en un mecanismo de verificación reutilizable dentro del firmware, no solo para la lectura del DS18B20, sino también como base para la validación de la información intercambiada en la etapa de comunicación, la cual se desarrolla con mayor detalle en la sección siguiente.

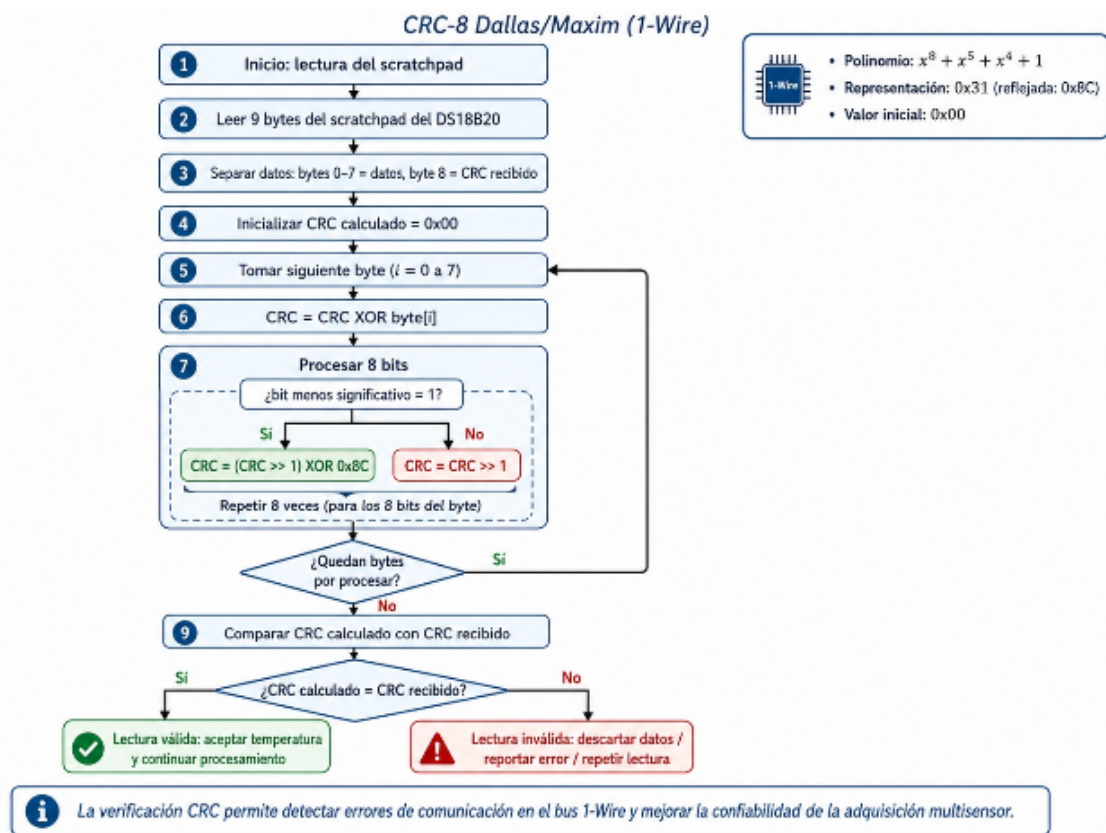


Figura 6.10: Rutina de verificación CRC del DS18B20

6.5.6. Acceso por registros en ATmega328PB

Después de definir la secuencia de lectura, direccionamiento y verificación de los sensores, fue necesario asegurar que la interacción con el microcontrolador se realizara con el nivel de precisión temporal requerido por el protocolo 1-Wire. Por esta razón, el firmware se desarrolló mediante acceso directo a los registros del ATmega328PB, evitando capas de abstracción de más alto nivel que pudieran introducir sobrecarga o variaciones no deseadas en la ejecución.

En particular, se utilizaron registros de puerto como DDRx, PORTx y PINx para controlar la configuración de entradas y salidas, así como los cambios de estado necesarios durante la comunicación con los sensores. Este enfoque permitió manejar la línea 1-Wire de manera compatible con el comportamiento requerido por el protocolo, alternando entre estado de salida en bajo y estado de entrada con liberación de la línea, lo cual equivale a una implementación tipo *open-drain* por software. En la práctica, el control del bus se ajustó al pin efectivamente utilizado en el diseño, garantizando coherencia entre el firmware y el hardware desarrollado [11].

Además del manejo de la línea 1-Wire, el acceso por registros también se empleó para controlar señales auxiliares del sistema, entre ellas el pin de habilitación de transmisión del transceptor RS-485. Esto permitió gestionar con mayor precisión el cambio entre recepción y transmisión dentro del esquema *half-duplex*, manteniendo sincronía con la lógica general de adquisición y envío de datos.

Desde el punto de vista del diseño embebido, esta decisión aportó una mayor consistencia temporal en la ejecución del firmware y permitió alinear mejor el comportamiento del sistema con las exigencias del protocolo y de la arquitectura modular. En consecuencia, el acceso directo a registros no solo respondió a una necesidad técnica de temporización, sino también a la intención de desarrollar una solución sobre hardware propio, donde el control detallado de los recursos del microcontrolador constituye una ventaja frente a implementaciones más dependientes de librerías generales.

6.5.7. Uso de la interfaz USART para transmisión y diagnóstico

Además de la lectura de sensores, el firmware integró el uso de las interfaces USART del microcontrolador para soportar tanto la transmisión de datos como las tareas de diagnóstico del sistema. Esta integración resultó fundamental para articular la adquisición local con el resto de la arquitectura modular, permitiendo que la información obtenida desde los sensores pudiera ser organizada, transferida y supervisada durante las etapas de prueba, depuración e integración del prototipo.

En la implementación desarrollada, las interfaces seriales del ATmega328PB se emplearon con propósitos diferenciados. Por una parte, USART0 se utilizó para la comunicación con el computador, incluyendo funciones de depuración, reporte de estados y envío de información en un formato interpretable por la plataforma de visualización. Por otra parte, USART1 se destinó al intercambio de datos sobre el bus RS-485 entre la placa de adquisición y la placa de comunicación, permitiendo mantener separada la salida externa hacia el PC de la comunicación interna entre módulos.

La configuración serial se realizó en modo 8N1 a 9600 baudios, utilizando los registros del

periférico para ajustar la velocidad de transmisión, habilitar recepción y transmisión, y controlar el flujo de datos. Esta organización permitió que el firmware no solo ejecutara la lógica de lectura del bus 1-Wire, sino que también preparara y transfiriera la información hacia la capa de comunicación, manteniendo coherencia con la arquitectura maestro-esclavo adoptada en el sistema.

Asimismo, el uso de USART facilitó la incorporación de funciones de diagnóstico y robustez durante la consolidación del firmware en MPLAB X. Entre ellas se incluyeron el reporte de estados de operación, la supervisión de la comunicación y la verificación del comportamiento general del sistema durante la validación experimental. De esta manera, la interfaz serial no solo cumplió una función de transporte de datos, sino que también se convirtió en un apoyo importante para la observación, depuración y seguimiento del prototipo durante su desarrollo.

La Figura 6.11 presenta el diagrama de bloques del periférico USART del ATmega328PB, utilizado como referencia para la configuración y aprovechamiento de la comunicación serial dentro del firmware.

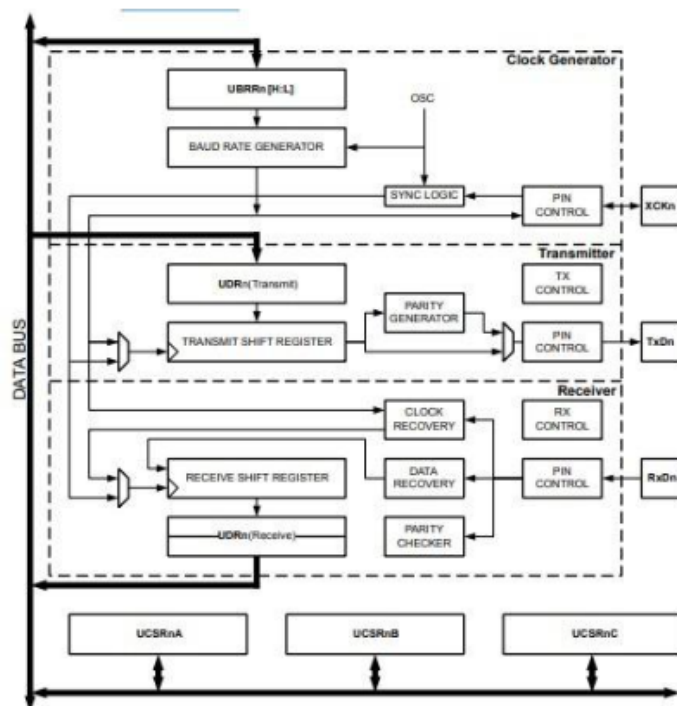


Figura 6.11: Diagrama de bloques del periférico USART (transmisor/receptor)[9]

6.5.8. Lógica de adquisición y envío

Una vez definidas las rutinas de lectura 1-Wire, la validación de datos, el acceso por registros y el uso de las interfaces seriales, el firmware se organizó como un flujo completo de adquisición y envío de información dentro del sistema. En la placa de adquisición, esta lógica comenzó con la detección de los sensores presentes en el bus, seguida de su identificación mediante el código ROM

correspondiente y su asociación con una posición lógica o *slot* dentro del arreglo de medición. Esto permitió mantener una organización estable de las lecturas, independientemente del orden físico en que los dispositivos estuvieran conectados.

Posteriormente, el firmware ejecutó la secuencia de medición para cada sensor reconocido: inicio de conversión, espera del tiempo correspondiente, lectura del *scratchpad*, validación por CRC y almacenamiento del valor obtenido en una estructura de datos interna. Una vez completada esta fase, el sistema actualiza los indicadores de estado asociados a los sensores y prepara la información para su transmisión hacia el módulo de comunicación.

De esta forma, la lógica de adquisición y envío cerró el ciclo funcional del firmware: primero se obtienen y validan las temperaturas a nivel local, luego se organizan en una trama y finalmente se transfieren hacia la etapa de comunicación. Esta secuencia enlaza de manera natural el desarrollo del firmware con la siguiente sección del informe, en la que se describe cómo se estructura, transmite y verifica la información intercambiada entre módulos.

6.6. Comunicación y estructura de tramas

A partir de la lógica de adquisición y envío implementada en el firmware, fue necesario definir un esquema de comunicación que permitiera transferir la información de temperatura entre la tarjeta de adquisición y la tarjeta de comunicación de forma ordenada, robusta y verificable. Para ello, se empleó la interfaz UART del microcontrolador enlazada con un transceptor THVD1400, operando sobre un bus RS-485 en configuración *half-duplex*. Esta decisión permitió integrar la etapa de procesamiento local con un medio físico de transmisión más robusto, adecuado para una arquitectura modular y para condiciones de laboratorio en las que pueden presentarse interferencias eléctricas, distancias de cableado y diferencias de referencia de tierra entre módulos [12][13].

Con el fin de garantizar un intercambio confiable de información, se definió una estructura de tramas que permitiera delimitar cada mensaje, organizar los valores transmitidos y verificar su integridad. De esta manera, la comunicación no se redujo únicamente al envío serial de bytes, sino que incorporó un formato específico para representar las temperaturas medidas, detectar errores y mantener coherencia en el intercambio entre el módulo esclavo de adquisición y el módulo maestro de comunicación [13].

6.6.1. Empaquetado de los datos

La información de temperatura generada por la placa de adquisición se organizó en una trama de longitud fija de 19 bytes. Esta estructura se diseñó para simplificar la interpretación de los datos por parte del módulo receptor y mantener un formato consistente durante toda la operación del sistema. La trama inicia con dos bytes de cabecera, 0xA5 y 0x5A, utilizados como patrón de sincronización, seguidos por 16 bytes de carga útil correspondientes a ocho valores de temperatura y, finalmente, un byte adicional de verificación CRC8.

Cada temperatura se representa como un entero con signo de 16 bits en formato *Little Endian*, utilizando una escala en décimas de grado Celsius (`temp x10`). Esta decisión permitió evitar el uso de variables de punto flotante dentro del firmware, simplificando el procesamiento en el microcontrolador y reduciendo la sobrecarga computacional. Así, por ejemplo, un valor de 25.3 °C se transmite como 253, mientras que las temperaturas negativas también pueden representarse mediante enteros con signo.

La carga útil de la trama contiene, por tanto, ocho temperaturas consecutivas, cada una ocupando dos bytes: un byte menos significativo y un byte más significativo. En los casos en que un sensor no se detecta o la lectura resulta inválida, el sistema utiliza el valor centinela -9999 para indicar que el dato correspondiente no debe interpretarse como una temperatura válida. Este criterio permitió mantener fija la longitud de la trama, incluso cuando existieran errores o ausencia de sensores en alguno de los canales.

Finalmente, una vez construido el bloque de sincronización y la carga útil, se calculó un byte de CRC8 sobre los primeros 18 bytes de la trama. Este valor se añadió al final del mensaje para permitir la verificación de integridad en el módulo receptor. Con ello, el empaquetado de los datos no solo organizó la información para su transmisión, sino que también incorporó una primera capa de confiabilidad dentro del protocolo de comunicación adoptado.

6.6.2. Transmisión de la trama

Una vez definidos los campos que componen la trama, fue necesario establecer la forma en que esta sería enviada entre la placa de adquisición y la placa de comunicación. Para ello, el sistema empleó un bus RS-485 operando en modo *half-duplex* a 9600 bps, aprovechando la robustez de esta interfaz frente al ruido eléctrico y su conveniencia para una arquitectura modular con varios bloques interconectados. En este esquema, la dirección del bus se controló mediante el pin DE del transceptor, implementado en PD5, de modo que el módulo pudiera alternar explícitamente entre los estados de transmisión y recepción [13].

En la placa esclava, la secuencia de transmisión consistió en habilitar el modo de envío, esperar un breve tiempo de estabilización del transceptor y, posteriormente, transmitir la trama completa de 19 bytes por USART1. Finalizado el envío, el firmware esperó la confirmación de salida del último bit antes de deshabilitar nuevamente la transmisión y devolver el bus al modo de recepción. Para mejorar la estabilidad del enlace, se incorporaron retardos de aproximadamente 200 μ s antes y después de la transmisión, evitando conmutaciones bruscas en la dirección del bus y reduciendo la probabilidad de errores durante el cambio de estado [11].

Por su parte, el maestro siguió el mismo principio de control de dirección, aunque su intervención en el bus fue más breve, ya que únicamente transmitió 1 byte, correspondiente al comando de solicitud 0x01. Esta decisión redujo el tiempo de ocupación del bus por parte del maestro y dejó una mayor ventana disponible para la recepción de la respuesta proveniente del esclavo. De esta manera, la transmisión de la trama quedó integrada a una lógica de comunicación eficiente, donde el esclavo concentra la carga útil de información y el maestro actúa principalmente como iniciador

del intercambio [14].

6.6.3. Validación de la trama recibida

Una vez recibida la información por parte del maestro, fue necesario aplicar mecanismos de validación que permitieran asegurar que la trama correspondiera realmente a una respuesta válida del sistema y que no hubiera sido alterada por ruido o errores de transmisión. Para ello, el firmware implementó una validación en dos niveles: sincronización por cabecera y comprobación mediante CRC8 [10].

El primer nivel de validación se basó en la detección de la secuencia de cabecera `0xA5 0x5A`, utilizada como patrón de inicio de trama. El maestro empleó una lógica de resincronización capaz de verificar el primer byte, esperar el segundo y reiniciar la búsqueda cuando la secuencia no coincidía con el encabezado esperado. Este mecanismo permitió mantener alineada la recepción aun cuando aparecieran bytes espurios o parciales en el bus, evitando interpretar datos fuera de contexto como parte de una trama válida [13].

El segundo nivel de validación se realizó mediante el cálculo de CRC8 Dallas sobre los primeros 18 bytes de la trama, es decir, los dos bytes de cabecera y los dieciséis bytes de carga útil. El resultado obtenido se comparó con el byte de CRC recibido al final del mensaje. Si ambos valores no coincidían, la trama era descartada y el receptor reiniciaba su proceso de búsqueda de sincronización, sin aceptar la información ni generar retransmisiones adicionales. Solo cuando la cabecera y el CRC resultaban correctos, el maestro procedía a actualizar los valores de temperatura, procesarlos e imprimirlos en formato CSV hacia el computador.

Esta estrategia de doble validación fortaleció la robustez del protocolo, ya que permitió rechazar tramas corruptas sin comprometer la continuidad del sistema. En lugar de implementar un mecanismo complejo de retransmisión, el diseño asumió que una nueva solicitud periódica del maestro bastaría para recuperar información válida en el siguiente ciclo de comunicación.

6.6.4. Organización del intercambio entre módulos

El intercambio de información entre la placa de comunicación y la placa de adquisición se organizó bajo un esquema de solicitud-respuesta controlado por el maestro. En este modelo, la tarjeta de comunicación envía de manera periódica el comando `0x01`, mientras que la tarjeta de adquisición responde con la trama de temperaturas más reciente disponible en su memoria. Esta estrategia evita colisiones en el bus RS-485 y mantiene un control claro del flujo de información dentro del sistema.

Desde el punto de vista temporal, el maestro realiza una solicitud aproximadamente cada 1000 ms y espera una respuesta durante un máximo de 800 ms. En paralelo, el esclavo actualiza el conjunto completo de sensores cada 5000 ms, por lo que una misma trama puede ser consultada varias veces antes de que ocurra una nueva adquisición. Esta relación entre el período de solicitud y el período de medición resulta aceptable para la aplicación desarrollada, dado que la dinámica térmica del banco es lenta en comparación con los tiempos de comunicación [12].

En condiciones normales, la secuencia de intercambio puede resumirse de la siguiente manera: el maestro transmite el byte de solicitud, el esclavo detecta el comando, construye o reutiliza la trama de 19 bytes y la envía por RS-485; a continuación, el maestro recibe la cabecera, valida el CRC y decodifica las ocho temperaturas para su impresión en formato CSV. Considerando la velocidad de transmisión utilizada, el envío completo de la trama ocupa del orden de 20 ms, a lo que se suman los retardos de estabilización del transceptor y el tiempo de procesamiento interno del firmware [13].

Además del esquema de respuesta a solicitud, el firmware contempla mecanismos de robustez como el uso de *timeout* en la recepción y la resincronización automática cuando una trama no es válida. Esto permite que una falla puntual de comunicación no bloquee el sistema, sino que simplemente se descarte el ciclo afectado y se continúe con la siguiente consulta periódica. En consecuencia, la organización del intercambio entre módulos no solo responde a una decisión funcional, sino también a una estrategia de confiabilidad coherente con el comportamiento maestro-esclavo definido para el prototipo [14].

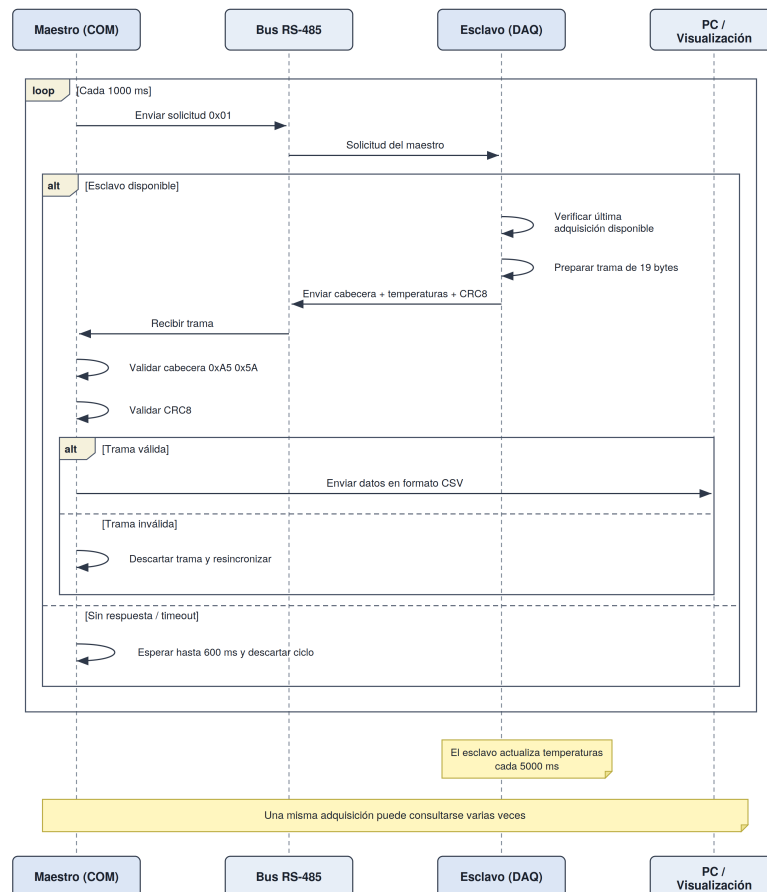


Figura 6.12: Secuencia temporal del intercambio de datos entre los módulos de adquisición y comunicación

6.7. Diseño de hardware y evolución de PCBs

6.7.1. Diseño esquemático y PCB

El diseño de hardware del sistema se desarrolló íntegramente en *KiCad*, abarcando la elaboración de esquemáticos, la asignación de huellas, el enrutamiento de pistas y la verificación de reglas de diseño. Dentro de este proceso también fue necesario crear elementos propios de biblioteca, particularmente para el módulo de alimentación AC/DC de la tarjeta POWER y para el conector de borde (*edge connector*) empleado en el *backplane*, debido a que estos componentes resultaban críticos para la integración física y eléctrica de la arquitectura modular [17].

La arquitectura del sistema se organizó en cuatro tarjetas principales: DAQ/DAT, COM, POWER y *backplane*. Adicionalmente, se desarrolló una segunda variante del *backplane* para incorporar la conexión de fase, neutro y tierra requerida por la tarjeta de alimentación. En esta etapa no se considera la placa de sensores como parte central de la tesis, ya que su función fue complementaria y estuvo orientada principalmente a sustituir la protoboard utilizada durante las pruebas.

Desde el punto de vista físico, se adoptó una dimensión aproximada de 8 cm de ancho por 10 cm de largo para las tarjetas, buscando un formato compacto, fácil de manipular y compatible con la futura integración en el chasis del sistema. Asimismo, todas las placas se diseñaron en dos capas, decisión motivada tanto por la necesidad de enrutar las señales y la alimentación de forma ordenada como por el uso del *edge connector*, cuya implementación requería una distribución apropiada en ambas caras de la PCB [41].

En cuanto a los criterios de ruteo, se priorizó el uso de pistas de mayor ancho en las zonas de entrada de potencia, especialmente en la tarjeta POWER, y se ubicaron los capacitores de desacoplamiento lo más cerca posible de los componentes principales, en particular del microcontrolador. Además, se aprovechó una de las capas como plano de tierra, con el fin de simplificar el enrutamiento de GND y mejorar la integridad eléctrica de las placas. Para las tarjetas DAT y COM se tomó como referencia funcional parte del esquema de un Arduino Nano, adaptándolo al uso de hardware propio y complementándolo con el transceptor THVD1400 y los indicadores LED requeridos por el sistema [43].

6.7.2. Primera iteración de hardware

La primera iteración del hardware correspondió al desarrollo inicial de todas las tarjetas del sistema, aunque las placas DAT y COM fueron las primeras en consolidarse a nivel de diseño. Posteriormente, el conjunto completo de PCBs fue enviado a fabricación con el propósito de validar no solo la arquitectura modular propuesta, sino también la interconexión física entre tarjetas mediante el sistema de conectores de borde.

Durante esta primera iteración se presentó un error importante en la huella del *edge connector*. El componente implementado en las placas no coincidía dimensionalmente con el conector adquirido,

por lo que los pines no hacían contacto de forma correcta y las tarjetas no podían interactuar entre sí como estaba previsto en el diseño original. Este inconveniente afectó la integración física del sistema, aunque no invalidó por completo el proceso de validación funcional.

A pesar de este problema, fue posible realizar pruebas del firmware y de la lógica de comunicación mediante conexiones auxiliares con cables soldados directamente en los puntos correspondientes del conector de borde. Esta solución temporal permitió confirmar que el código desarrollado funcionaba correctamente y que el problema principal estaba concentrado en la huella mecánica del conector, no en el diseño lógico del sistema. Una vez verificada esta situación, se procedió a corregir el diseño y solicitar nuevamente las placas, orientando la segunda fabricación hacia una versión más cercana al producto final que al prototipo inicial [43].

6.7.3. Hallazgos y correcciones del rediseño

El principal hallazgo derivado de la primera iteración fue la incompatibilidad de la huella del *edge connector* con el componente realmente adquirido. Este error impedía la conexión correcta entre las tarjetas y, por tanto, afectaba directamente la funcionalidad del *backplane* como elemento de interconexión. A partir de esta observación, se realizó un rediseño centrado en corregir dicha huella y ajustar el sistema al conector efectivamente disponible [41].

Más allá de esta corrección, no se identificaron fallas eléctricas relevantes en los circuitos principales de las tarjetas. El funcionamiento de la comunicación serial, el uso del transceptor RS-485, la integración USB-serial en la tarjeta COM, los indicadores LED y la lógica de adquisición se mantuvieron consistentes con lo esperado. En este sentido, el rediseño no implicó cambios profundos en el comportamiento electrónico del sistema, sino una corrección puntual pero crítica para lograr una integración mecánica y eléctrica adecuada entre los módulos.

Como parte de esta evolución también se desarrolló una segunda fase del *backplane*, incorporando una conexión específica para fase, neutro y tierra asociada a la tarjeta POWER. Esta ampliación respondió a una necesidad de integración de la alimentación dentro del conjunto modular, y no a una falla funcional previa del resto de las placas. En consecuencia, el rediseño del hardware puede interpretarse como una etapa de ajuste fino que permitió pasar de una solución funcional en pruebas a una realización física más coherente y mejor integrada.

6.7.4. Tarjeta DAQ/DAT

La tarjeta DAT se diseñó como el módulo principal de adquisición del sistema. En ella se integran el microcontrolador ATmega328PB, el transceptor THVD1400 para la comunicación RS-485, la regulación local de voltaje, la interfaz de programación ICSP y los indicadores LED asociados al estado general del sistema y a la actividad de los sensores. Su función principal es realizar la lectura de temperatura, procesar localmente la información y preparar los datos para su envío hacia el módulo de comunicación.

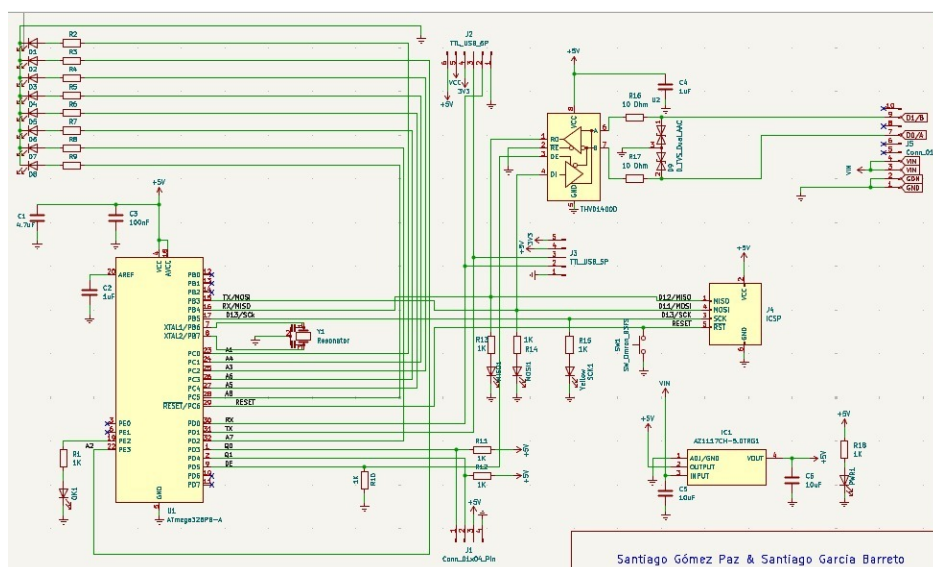


Figura 6.13: Esquemático de la tarjeta DAT

En la Figura 6.13 se observa la integración del ATmega328PB como núcleo de procesamiento, junto con el transceptor RS-485, la etapa de regulación, la interfaz ICSP y los LED de monitoreo. También se evidencia la disposición de conectores auxiliares para depuración y prueba, así como la conexión mediante *edge connector* hacia el *backplane*.

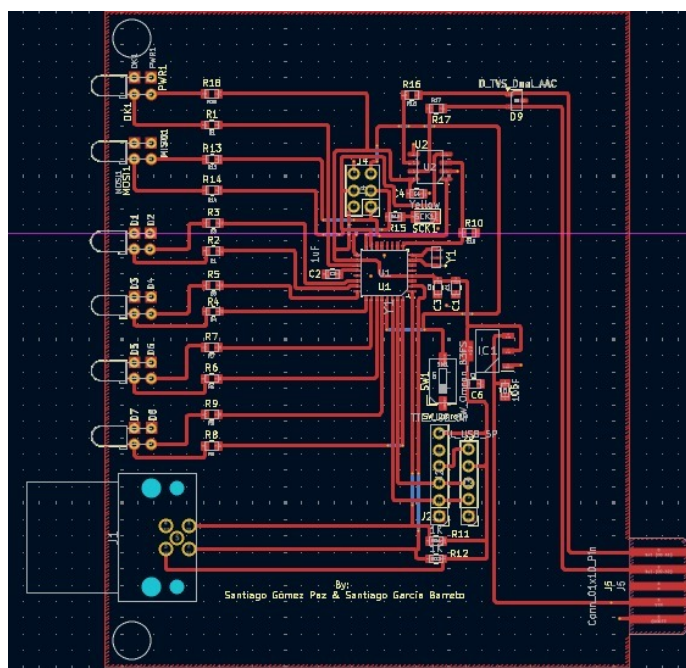


Figura 6.14: Diseño PCB de la tarjeta DAT

El diseño PCB de la tarjeta DAT, presentado en la Figura 6.14, organiza el microcontrolador y los periféricos principales de forma compacta, priorizando la cercanía de los capacitores de desacoplamiento y el orden del ruteo entre procesamiento, sensado y comunicación. La presencia de los ocho LED laterales permite, además, una visualización rápida del estado de los sensores durante la validación del sistema.

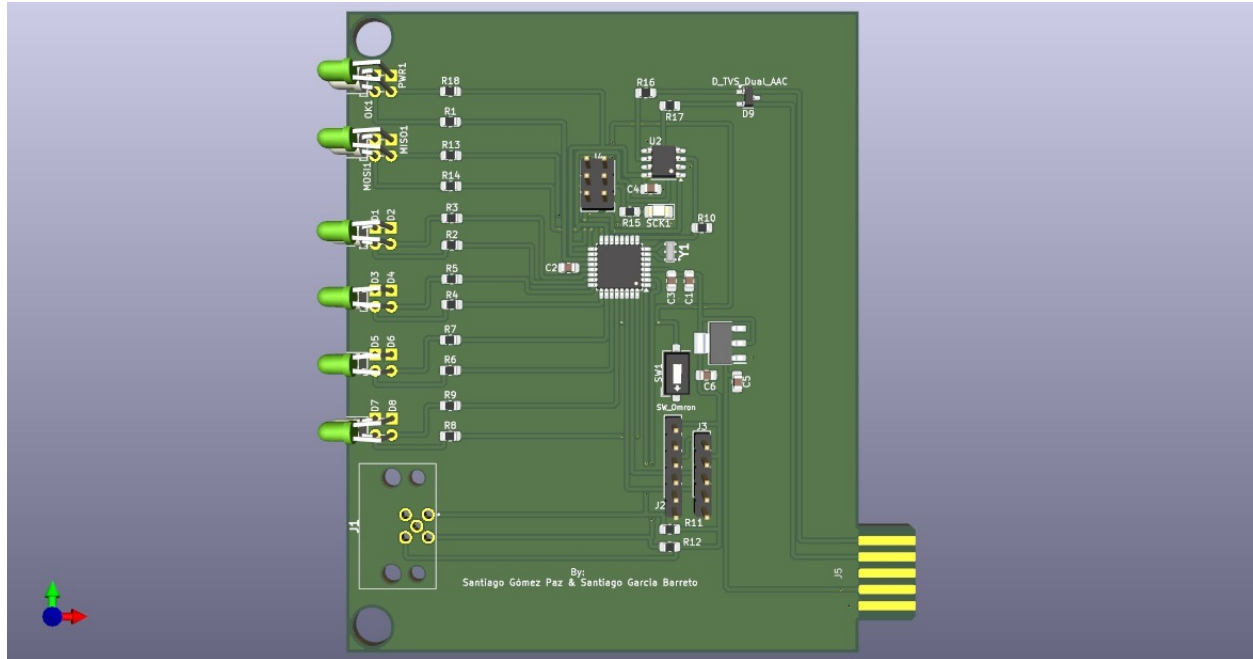


Figura 6.15: Vista 3D de la tarjeta DAT

La vista 3D presentada en la Figura 6.15 permite apreciar la distribución final de los componentes y la manera en que la tarjeta se integra físicamente al sistema modular. En particular, se observa la posición del *edge connector* y de los indicadores de estado, elementos clave para su acoplamiento al *backplane* y para el diagnóstico visual durante las pruebas.

Tabla 6.13: Componentes principales de la tarjeta DAQ/DAT

Referencia	Qty	Valor	Tipo / Encapsulado	Función principal
U1	1	ATmega328PB-A	TQFP-32	Microcontrolador principal para el procesamiento y control del sistema
U2	1	THVD1400D	SOIC-8	Transceptor RS-485 para la comunicación industrial
IC1	1	AZ1117CH-5.0	SOT-223	Regulador LDO de 5 V para la alimentación del sistema
Y1	1	Resonador 16 MHz	SMD	Fuente de reloj para el microcontrolador
D9	1	D_TVS_Dual	SOT-23-3	Diodo TVS para protección contra transientes
C1	1	4.7 μ F	0805 SMD	Desacoplamiento de baja frecuencia
C2, C4	2	1 μ F	0805 SMD	Desacoplamiento de alimentación
C3	1	100 nF	0805 SMD	Desacoplamiento de alta frecuencia
C5, C6	2	10 μ F	0805 SMD	Filtrado de alimentación
R1–R15, R18	16	1 k Ω	0805 SMD	Resistencias de pull-up y limitación para LEDs y señales
R16, R17	2	10 Ω	0805 SMD	Resistencias de terminación y corriente
D1, D3, D5, D7	4	LED (clear)	THT 3 mm horizontal	Indicadores de estado con montaje bajo
D2, D4, D6, D8	4	LED	THT 3 mm horizontal	Indicadores de estado con montaje estándar
MISO1, MOSI1, OK1, PWR1, SCK1	5	LEDs indicadores	THT / SMD 1206	LEDs de diagnóstico SPI: MISO, MOSI, SCK, OK y PWR
SW1	1	SW_Omron_B3F5	SMD DIP slide	Pulsador de reset del sistema
J1	1	Conn 1 $\times$ 4	T414501501001	Conector de señal externo de 4 pines
J2	1	TTL_USB 6P	PinHeader 1 $\times$ 6, 2.54 mm	Puerto TTL/USB para programación
J3	1	TTL_USB 5P	PinHeader 1 $\times$ 5, 2.54 mm	Puerto TTL alternativo
J4	1	ICSP	PinHeader 2 $\times$ 3, 2.54 mm	Conector de programación en circuito (ICSP)
J5	1	Conn 1 $\times$ 10	Edge connector 5 $\times$ 2	Conector de borde para la interconexión con el backplane

Tabla 6.14: Conectores e interfaces principales de la tarjeta DAQ/DAT

Interfaz	Tipo de conector	Función principal	Señales asociadas / observaciones
Conector de sensor	M12	Interfaz de conexión con los sensores DS18B20	Línea de datos 1-Wire, alimentación y referencia de tierra
Conector ICSP	Header de programación	Programación In-Circuit del ATmega328PB	Permite cargar y actualizar el firmware directamente en la tarjeta
Edge connector	Conector de borde	Interfaz con el backplane	Distribuye alimentación y señales de comunicación hacia el sistema modular
Línea de sensado 1-Wire	Interna a la tarjeta	Lectura de temperatura desde los DS18B20	Implementada sobre el pin digital asignado al bus 1-Wire
Interfaz serial / comunicación	Interna hacia backplane	Envío de datos al módulo de comunicación	Usada para integrar la tarjeta DAT con la arquitectura general

Tabla 6.15: Pinout funcional del conector principal de la tarjeta DAQ/DAT

Pin / contacto	Señal	Función
1	GND	Referencia de tierra
2	Vcc	Alimentación del sensor
3	DQ	Línea de datos 1-Wire
4	–	Reserva / no conectado, si aplica

Tabla 6.16: LEDs indicadores de la tarjeta DAQ/DAT

LED	Función	Estado esperado
PWR	Indicación de presencia de alimentación en la tarjeta	Encendido cuando la tarjeta está energizada
OK	Estado general del sistema	Encendido cuando la operación básica de la tarjeta es correcta
S0	Estado del sensor asociado al slot 0	Encendido cuando el sensor es detectado y leído correctamente
S1	Estado del sensor asociado al slot 1	Encendido cuando el sensor es detectado y leído correctamente
S2	Estado del sensor asociado al slot 2	Encendido cuando el sensor es detectado y leído correctamente
S3	Estado del sensor asociado al slot 3	Encendido cuando el sensor es detectado y leído correctamente
S4	Estado del sensor asociado al slot 4	Encendido cuando el sensor es detectado y leído correctamente
S5	Estado del sensor asociado al slot 5	Encendido cuando el sensor es detectado y leído correctamente
S6	Estado del sensor asociado al slot 6	Encendido cuando el sensor es detectado y leído correctamente
S7	Estado del sensor asociado al slot 7	Encendido cuando el sensor es detectado y leído correctamente

6.7.5. Tarjeta COM

La tarjeta COM se diseñó como el módulo encargado de recibir la información proveniente de la tarjeta DAT y transferirla al computador para su visualización y registro. Para ello, integra un microcontrolador ATmega328PB, un transeceptor THVD1400, un conversor USB-serial CH340C, regulación local, interfaz de programación ICSP y LED de diagnóstico.

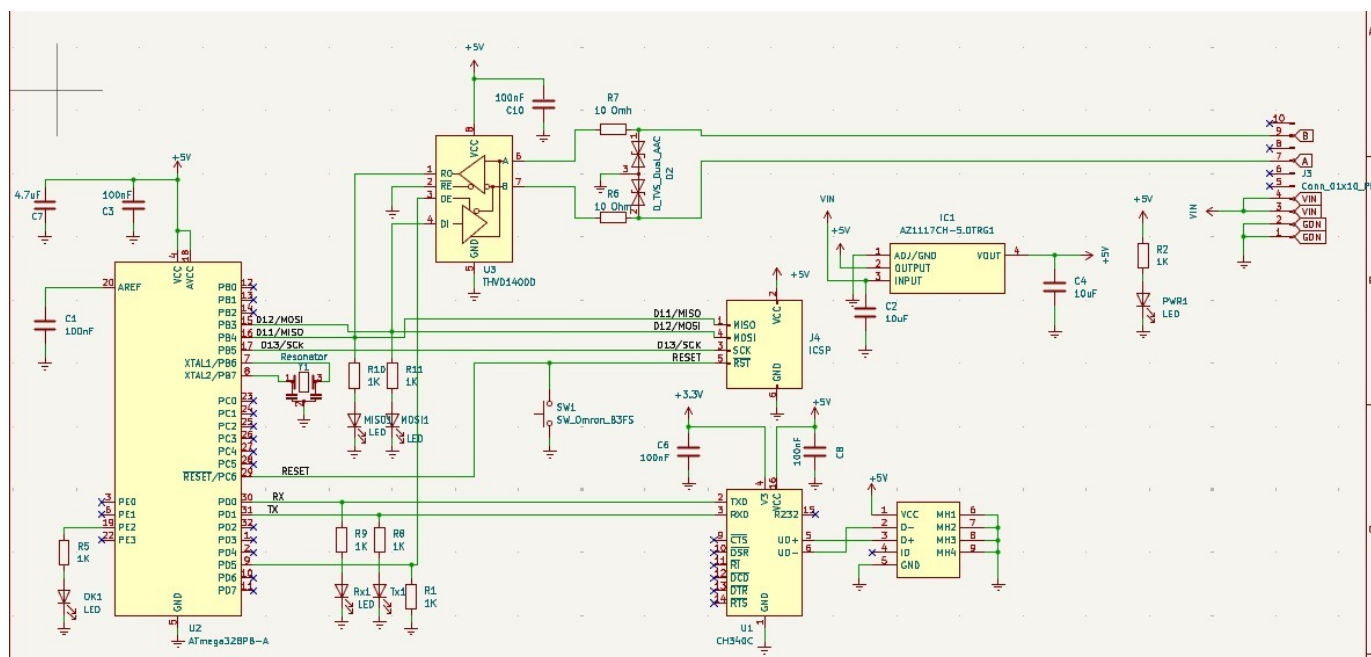


Figura 6.16: Esquematico de la tarjeta COM

En la Figura 6.16 se distinguen claramente tres bloques funcionales: el procesamiento mediante el ATmega328PB, la conversión USB-serial mediante el CH340C y la interfaz RS-485 implementada mediante el THVD1400. Esta combinación permite que la tarjeta opere como puente entre la red interna del sistema y el computador.

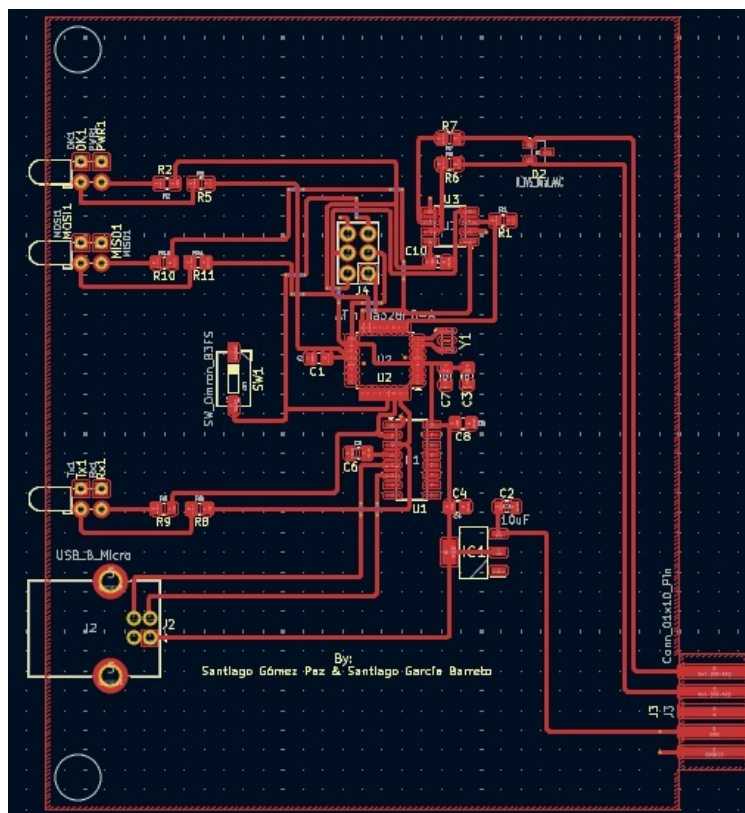


Figura 6.17: Diseño PCB tarjeta COM

En la PCB presentada en la Figura 6.17 se aprecia una disposición orientada a organizar la comunicación interna y externa de forma compacta, ubicando el conector USB, el conversor serial, el transceptor RS-485 y el microcontrolador dentro de un mismo módulo. Esto facilita el envío de los datos adquiridos hacia el PC para su posterior visualización y registro.

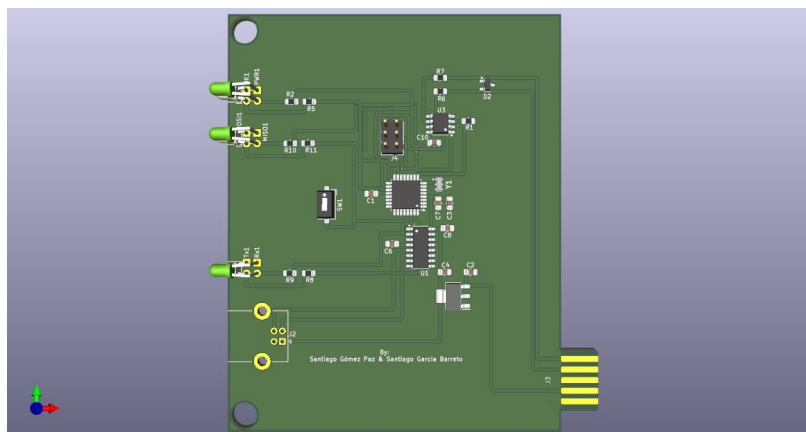


Figura 6.18: Vista 3D de la tarjeta COM

La vista 3D presentada en la Figura 6.18 muestra la integración física del conector USB tipo B, el *edge connector* y los componentes principales de comunicación. Esta representación permite comprender de forma más clara el papel de la tarjeta como interfaz entre el sistema embebido y la plataforma externa de visualización.

Tabla 6.17: Componentes principales de la tarjeta COM

Referencia	Qty	Valor	Tipo / Encapsulado	Función principal
U2	1	ATmega328PB-A	TQFP-32	Microcontrolador principal para el procesamiento y la comunicación
U1	1	CH340C	SOIC-16	Convertidor USB-Serial para la interfaz con el PC
U3	1	THVD1400D	SOIC-8	Transceptor RS-485 para la comunicación industrial
IC1	1	AZ1117CH-5.0	SOT-230	Regulador LDO de 5 V para la alimentación del sistema
Y1	1	Resonador 16 MHz	SMD	Fuente de reloj para el microcontrolador
D2	1	D_TV5_Dual	SOT-23-3	Protección contra transientes en la línea
C1, C3, C6, C8, C10	5	100 nF	0805 SMD	Desacoplamiento de alta frecuencia
C2, C4	2	10 μ F	0805 SMD	Filtrado bulk de alimentación
C7	1	4.7 μ F	0805 SMD	Desacoplamiento de baja frecuencia
R1, R2, R5, R8-R11	7	1 k Ω	0805 SMD	Pull-up y limitación de corriente para LEDs y señales
R6, R7	2	10 Ω	0805 SMD	Resistencias de terminación de línea
MISO1, PWR1, Rx1	3	LED	THT 3 mm horizontal	Indicadores MISO, alimentación y recepción (Rx)
MOSI1, OK1, Tx1	3	LED (clear)	THT 3 mm horizontal	Indicadores MOSI, estado OK y transmisión (Tx)
SW1	1	SW_Omron_B3FS	SMD DIP slide	Pulsador de reset del sistema
J2	1	USB_B_Micro	Lumberg horizontal	Conector USB micro-B para la comunicación con el PC
J3	1	Conn 1 $\times$ 10	Edge connector 5 $\times$ 2	Conector de borde hacia el backplane
J4	1	ICSP	PinHeader 2 $\times$ 3, 2.54 mm	Conector de programación en circuito (ISP)

Tabla 6.18: Conectores e interfaces principales de la tarjeta COM

Identificador / interfaz	Tipo de conector	Función principal	Señales asociadas / observaciones
Conector USB	USB Tipo B	Enlace de la tarjeta COM con el computador	Permite la transferencia de datos hacia el PC mediante interfaz USB-serial
Conector ICSP	Header de programación	Programación In-Circuit del ATmega328PB	Permite cargar y actualizar el firmware directamente en la tarjeta
Edge connector	Conector de borde	Interfaz con el backplane	Recibe alimentación y señales de comunicación desde la arquitectura modular
Interfaz RS-485	Interna a la tarjeta / hacia backplane	Comunicación con los demás módulos del sistema	Implementada mediante el transceptor THVD1400
Convertidor USB-serial	Interno a la tarjeta	Adaptación de la UART del microcontrolador hacia USB	Facilita el reconocimiento del sistema como puerto serial virtual en el PC

Tabla 6.19: Pinout funcional de las interfaces principales de la tarjeta COM

Interfaz	Pin / contacto	Señal	Función
USB Tipo B	VBUS	5 V	Alimentación de la interfaz USB
USB Tipo B	D−	USB D−	Línea diferencial de datos USB
USB Tipo B	D+	USB D+	Línea diferencial de datos USB
USB Tipo B	GND	Tierra	Referencia de la interfaz USB
ICSP	MISO	Master In Slave Out	Programación y depuración
ICSP	MOSI	Master Out Slave In	Programación y depuración
ICSP	SCK	Reloj serial	Programación y depuración
ICSP	RESET	Reset	Reinicio/control para programación
ICSP	VCC	Alimentación	Referencia de alimentación
ICSP	GND	Tierra	Referencia de tierra

Tabla 6.20: LEDs indicadores de la tarjeta COM

LED	Función	Estado esperado
PWR	Indicación de presencia de alimentación en la tarjeta	Encendido cuando la tarjeta COM está energizada
OK	Estado general de operación	Encendido cuando la tarjeta se encuentra operando correctamente
MOSI	Actividad de programación / transmisión SPI	Puede parpadear durante la programación o el intercambio asociado
MISO	Actividad de programación / recepción SPI	Puede parpadear durante la programación o el intercambio asociado
TX	Actividad de transmisión serial	Parpadea cuando la tarjeta envía datos hacia el PC o hacia el sistema
RX	Actividad de recepción serial	Parpadea cuando la tarjeta recibe datos desde el sistema o desde el PC

6.7.6. Tarjeta POWER

La tarjeta POWER se concibió como el módulo encargado de acondicionar y distribuir la alimentación del sistema. Su elemento principal es un módulo AC/DC tipo EPS-15-XX, complementado por un LED indicador y conectores de borde para su integración con el *backplane*.

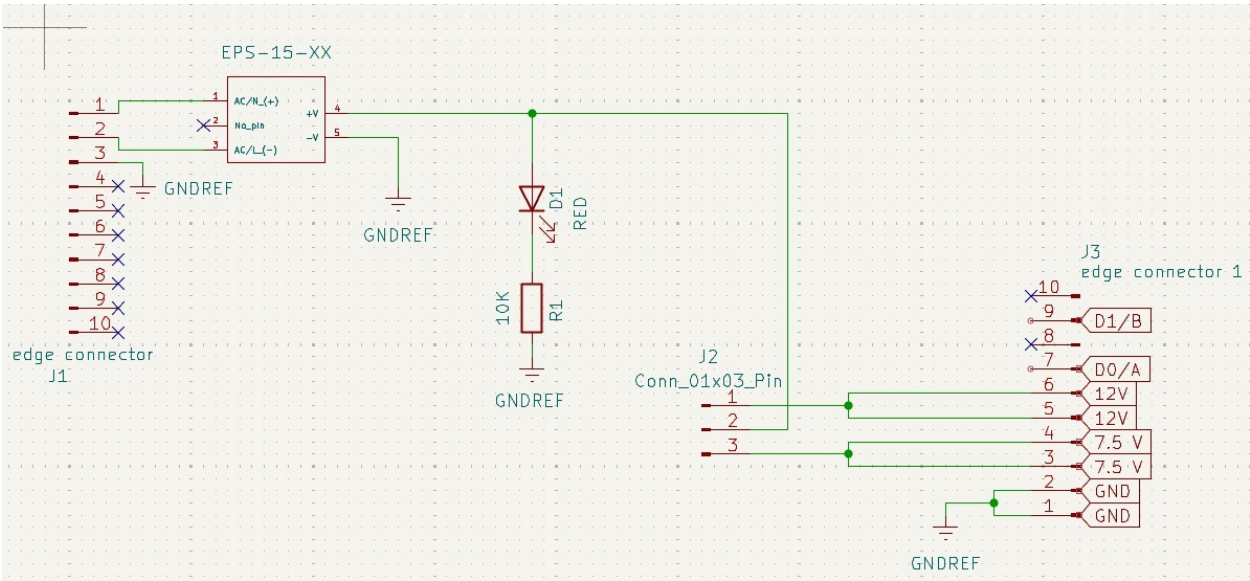


Figura 6.19: Esquemático de la tarjeta POWER

El esquemático de la tarjeta POWER, presentado en la Figura 6.19, muestra una estructura sencilla centrada en el módulo de alimentación, el LED de presencia de energía y los conectores necesarios para distribuir la tensión hacia el sistema modular. La simplicidad del circuito responde

a su función específica de suministro eléctrico.

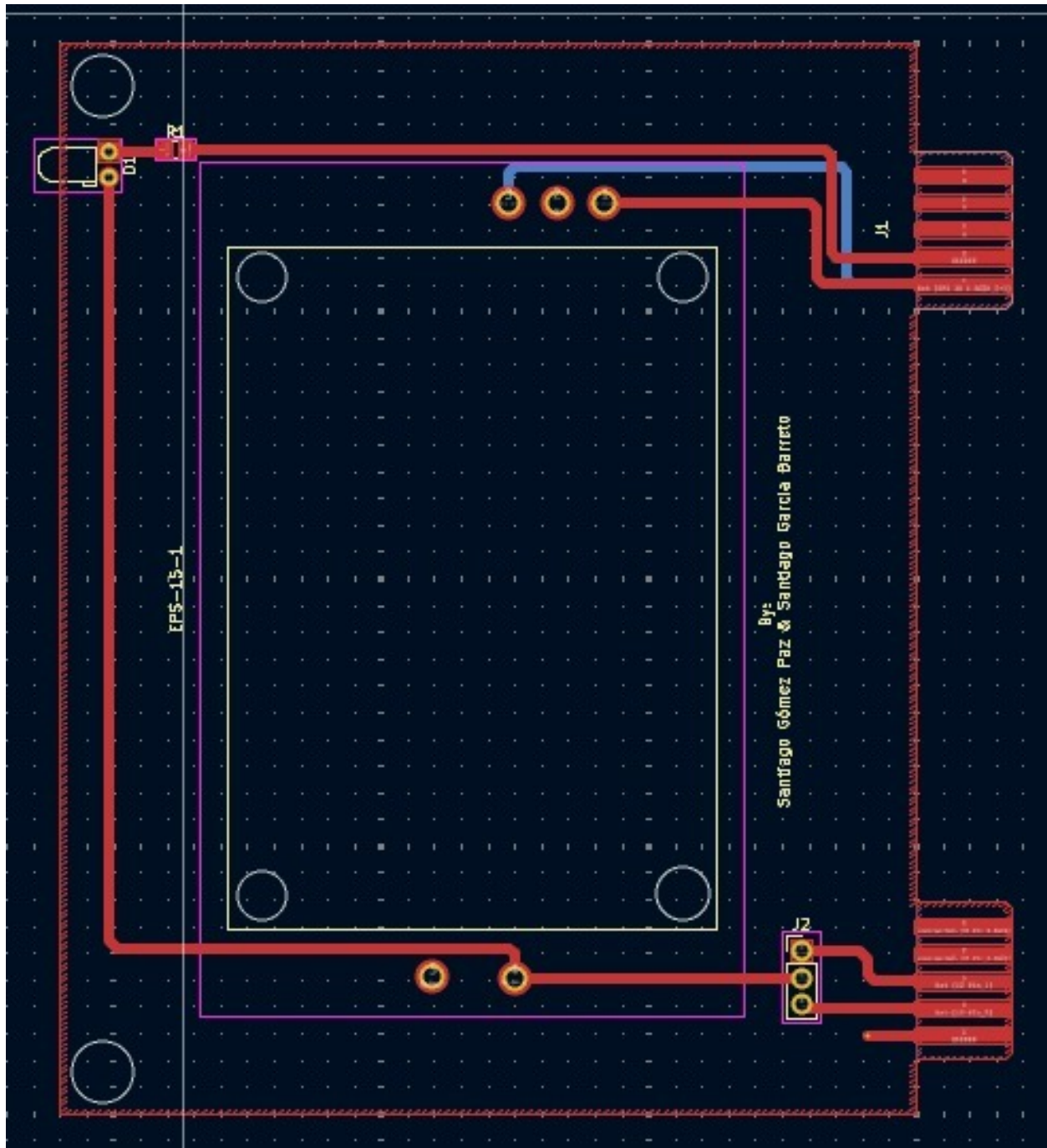


Figura 6.20: Diseño PCB de la tarjeta POWER

En la PCB presentada en la Figura 6.20 se observa que el diseño prioriza pistas de mayor ancho en las zonas de potencia y una disposición robusta del módulo AC/DC. Esta decisión se relaciona con la necesidad de manejar de forma segura la entrada y distribución de energía en el prototipo.

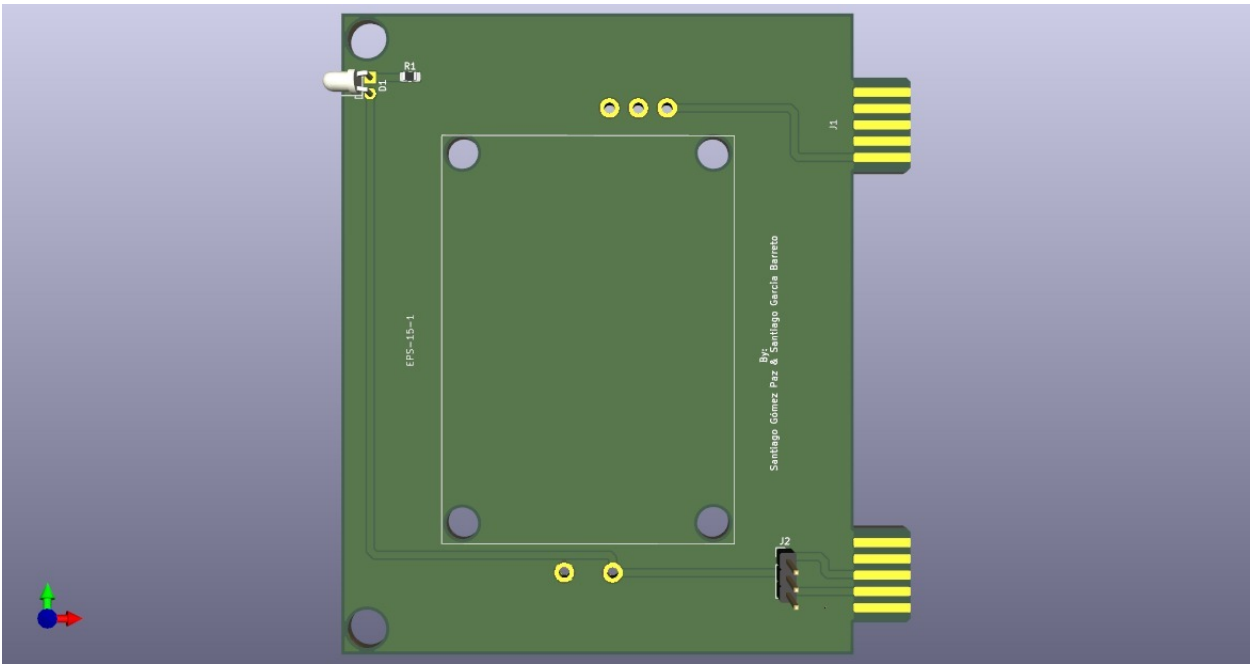


Figura 6.21: Vista 3D de la tarjeta POWER

La vista 3D presentada en la Figura 6.21 permite apreciar cómo el módulo de alimentación domina físicamente la tarjeta y cómo se integra al sistema mediante conectores de borde. Esta visualización refuerza el papel de la tarjeta POWER como fuente central de suministro para el resto de los módulos.

Tabla 6.21: Componentes principales de la tarjeta POWER

Referencia	Qty	Valor	Tipo / Encapsulado	Función principal
EPS-15-1	1	EPS-15-XX	Módulo PSU	Fuente de alimentación conmutada — entrada principal de potencia
D1	1	LED RED	THT 3 mm horizontal	Indicador visual de presencia de alimentación
R1	1	10 kΩ	0805 SMD	Resistencia de pullup / limitación del LED de potencia
J1	1	Edge connector	PCB edge 5×2	Conector de borde principal hacia backplane
J2	1	Conn 1×3	PinHeader 1×3 2.54 mm	Conector auxiliar de 3 pines (señal / selección)
J3	1	Edge connector	PCB edge 5×2	Segundo conector de borde (salida de potencia)

Tabla 6.22: Conectores e interfaces principales de la tarjeta POWER

Identificador / interfaz	Tipo de conector	Función principal	Señales asociadas / observaciones
Entrada AC	Terminal / conector de entrada	Recepción de energía desde la red eléctrica	Fase, neutro y tierra de protección, según el montaje realizado.
Módulo AC/DC	Integrado en la tarjeta	Conversión de entrada AC a niveles DC útiles	Etapa principal de alimentación del sistema.
Header / conector auxiliar	Header de 3 pines	Distribución o prueba de tensiones internas	Asociado a salidas de alimentación según el diseño.
Edge connector 1	Conector de borde	Inserción en el backplane y distribución de energía	Entrega rieles de alimentación al sistema modular.
Edge connector 2	Conector de borde	Segunda interfaz mecánica y eléctrica de inserción	Usado para asegurar compatibilidad de montaje y distribución hacia el backplane.

Tabla 6.23: Pinout funcional de la tarjeta POWER

Interfaz	Pin / contacto	Señal	Función
Entrada AC	1	AC/L	Línea de alimentación de entrada.
Entrada AC	2	AC/N	Neutro de alimentación de entrada.
Entrada AC	3	GND / PE	Tierra de protección.
Salida al backplane	1	GND	Referencia de tierra.
Salida al backplane	2	GND	Referencia de tierra compartida.
Salida al backplane	3	7.5 V	Riel de alimentación intermedio.
Salida al backplane	4	7.5 V	Riel de alimentación intermedio.
Salida al backplane	5	12 V	Riel de alimentación principal.
Salida al backplane	6	12 V	Riel de alimentación principal.
Salida al backplane	7	A	Línea reservada / distribución auxiliar.
Salida al backplane	8	B	Línea reservada / distribución auxiliar.

6.7.7. Backplane

El backplane se diseñó como la base de interconexión del sistema modular. Su función principal es permitir el acoplamiento de las distintas tarjetas mediante *edge connectors* y distribuir las líneas comunes de alimentación y comunicación.

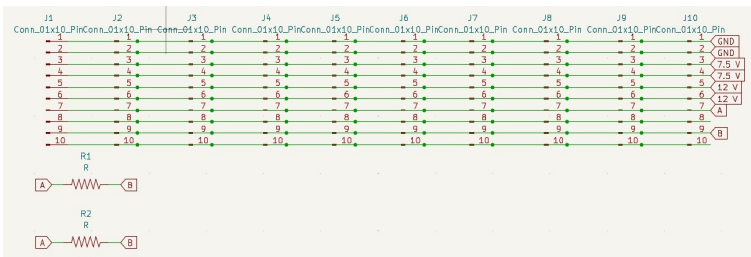


Figura 6.22: Esquemático del Backplane

En el esquemático del backplane se observa una estructura esencialmente pasiva, donde los conectores constituyen el elemento principal y permiten enrutar las señales compartidas entre las tarjetas del sistema.

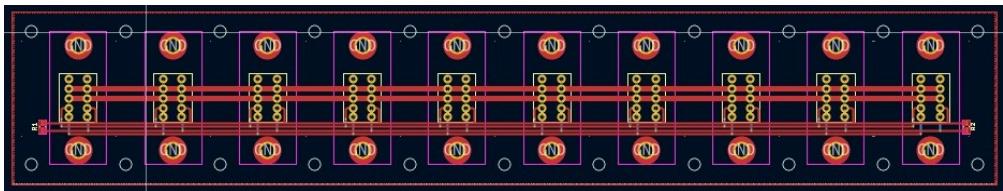


Figura 6.23: Diseño PCB del Backplane

La PCB del backplane evidencia la organización lineal de los conectores de borde, lo que facilita la inserción de los módulos y garantiza una interconexión ordenada. Esta disposición fue fundamental para materializar físicamente la arquitectura modular propuesta.

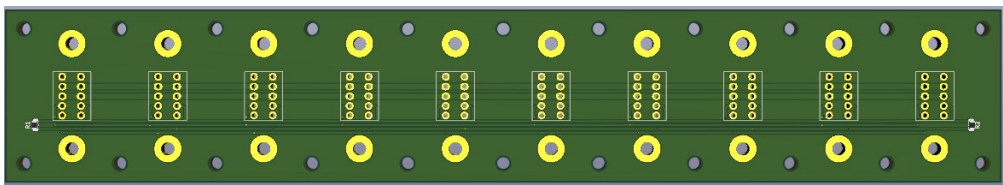


Figura 6.24: Vista 3D del Backplane

La vista 3D permite apreciar el backplane como elemento estructural y funcional del sistema, mostrando cómo sirve de soporte e interconexión para las tarjetas DAT, COM y POWER.

Tabla 6.24: Componentes principales del backplane

Referencia	Qty	Valor	Tipo / Encapsulado	Función principal
J1–J10	10	Conn 1×10	Card edge connector	Conectores de borde para alojamiento de tarjetas hija (slots)
R1, R2	2	R 1k	0805 SMD	Resistencias de terminación del bus

6.7.7.1. Backplane DC

Como complemento del backplane principal, se desarrolló una variante orientada específicamente a la integración de la alimentación de red, incorporando conexiones para fase, neutro y tierra.

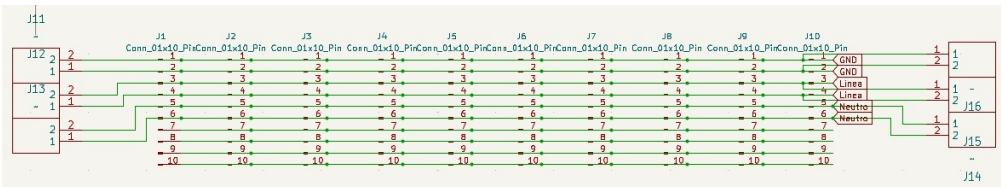


Figura 6.25: Esquemático del Backplane DC

El esquemático del backplane DC incorpora, además de la disposición general de conectores, las conexiones adicionales necesarias para la integración de fase, neutro y tierra dentro del sistema.

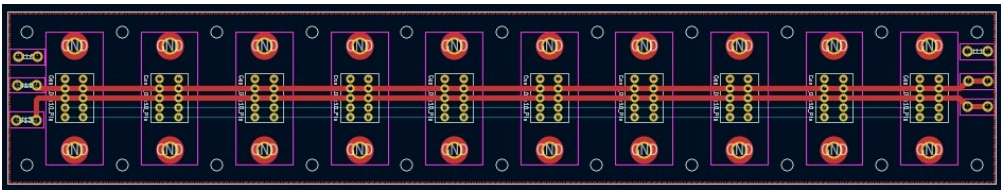


Figura 6.26: Diseño del PCB del Backplane DC

La PCB correspondiente mantiene la lógica lineal del backplane principal, pero añade los conectores externos asociados a la alimentación. Esto permite una integración más completa de la tarjeta POWER con el resto de la arquitectura.

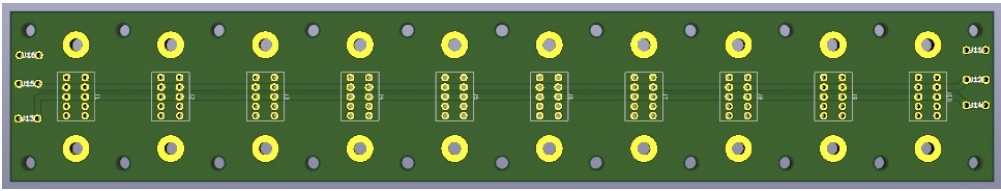


Figura 6.27: Vista 3D del Backplane DC

La vista 3D permite distinguir claramente esta variante del backplane, en la que se aprecian los conectores adicionales destinados a la alimentación de red.

Tabla 6.25: Componentes principales del backplane

Referencia	Qty	Valor	Tipo / Encapsulado	Función principal
J1-J10	10	Conn 1x10	Card edge connector	Conectores de borde para distribución de bus DC entre tarjetas
R1, R2	2	R 1k	0805 SMD	Resistencias de terminación del bus

6.7.8. Integración física del sensado

Como complemento al desarrollo principal de la tesis, se propuso una placa para la organización física del sensado basada en ocho sensores DS18B20. Esta solución surgió con el objetivo de sustituir la protoboard utilizada durante las pruebas y ofrecer una disposición más ordenada, limpia y profesional del arreglo de sensores. No obstante, es importante aclarar que esta placa no hace parte central del alcance de la tesis, ya que el trabajo se concentró en la adquisición, procesamiento y transmisión de datos de temperatura, y no en el desarrollo del soporte físico del arreglo de sensado [10, 31].

Desde el punto de vista funcional, esta placa reproduce la misma lógica de conexión que podía lograrse en protoboard, enrutando las tres líneas básicas requeridas por cada sensor: datos, alimentación y tierra. Su principal aporte no fue una mejora sustancial en la estabilidad de la medición, sino una mejora en el orden, montaje y presentación del sistema. Además, su conexión con el resto del prototipo se planteó mediante conectores M12, permitiendo una integración física más limpia con la tarjeta correspondiente.

Por tratarse de una mejora adicional y no de un elemento esencial del objetivo principal del trabajo, esta placa se menciona de manera breve dentro del documento, sin necesidad de otorgarle el mismo nivel de detalle que a las tarjetas DAT, COM, POWER y backplane. En este sentido, se presenta como un apoyo para la integración física del sensado, más que como un resultado central de la arquitectura electrónica desarrollada.

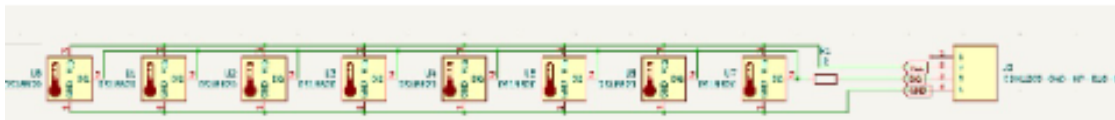


Figura 6.28: Esquemático de la placa de integración física del sensado

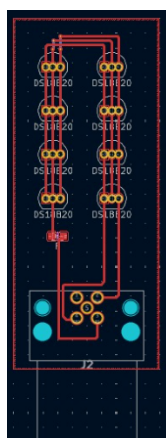


Figura 6.29: Diseño PCB de la placa de integración física del sensado

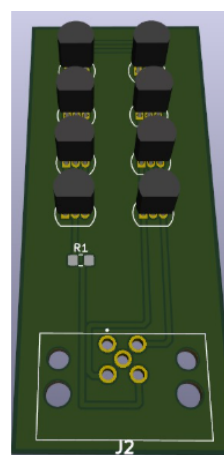


Figura 6.30: Vista 3D de la placa de integración física del sensado

6.7.9. Fabricación y ensamble

Una vez consolidadas las versiones corregidas del diseño, se generaron los archivos de fabricación de las tarjetas y se procedió a enviarlos a JLCPCB, proveedor ubicado en China. En total, las placas se mandaron a fabricar en dos ocasiones. La primera correspondió a la iteración inicial del hardware, en la cual se detectó el inconveniente asociado a la huella del *edge connector*; la segunda correspondió a la versión corregida, que es la empleada en el montaje final del sistema [43].

El ensamble de las tarjetas se realizó de forma manual por los autores del proyecto. Esta decisión respondió, por una parte, a la intención de adquirir experiencia directa en el montaje de PCBs y, por otra, a la dificultad de integrar todos los componentes mediante un servicio externo, dado que no todos se encontraban disponibles en el proveedor de fabricación. Durante esta etapa, los componentes que presentaron mayor dificultad fueron el ATmega328PB y el resonador, principalmente por el cuidado requerido en su alineación y soldadura.

Después del ensamble se llevaron a cabo pruebas básicas de verificación, entre ellas continuidad eléctrica, alimentación, reconocimiento USB, lectura de sensores, funcionamiento de los LED e intercambio de datos entre módulos. En una fase inicial, estas pruebas se realizaron incluso antes de contar con la segunda versión funcional del *edge connector*, empleando conexiones temporales con cables sobre los pines correspondientes. Esto permitió validar que el código, la adquisición y la comunicación operaban correctamente, y que la única falla crítica de la primera iteración estaba asociada a la interconexión mecánica y no al diseño electrónico principal.

Una vez recibidas las placas corregidas, se repitieron las pruebas con el ensamble definitivo. En esta etapa se verificó la lectura de sensores, el encendido de los LED de estado y la comunicación entre las tarjetas DAT y COM, confirmando que el sistema completo operaba de acuerdo con lo esperado. En consecuencia, el proceso de fabricación y ensamble no solo permitió materializar físicamente la arquitectura propuesta, sino también cerrar el ciclo de validación del hardware antes de su integración en la estructura mecánica final.

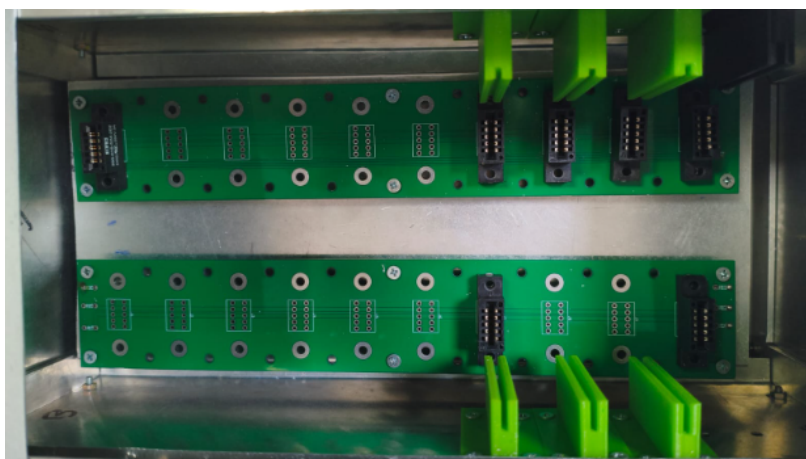


Figura 6.31: Tarjetas soldadas del Backplane

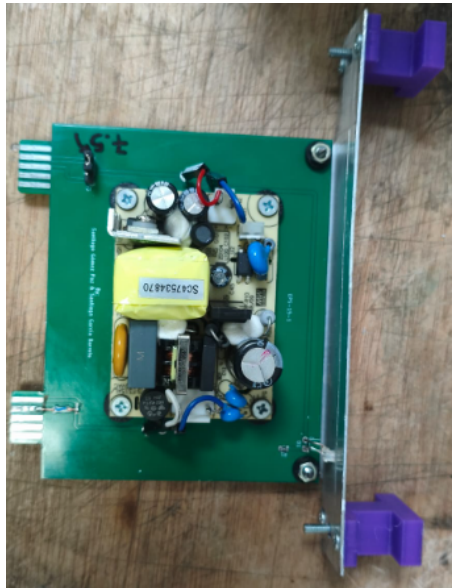


Figura 6.32: Tarjeta soldada de Power

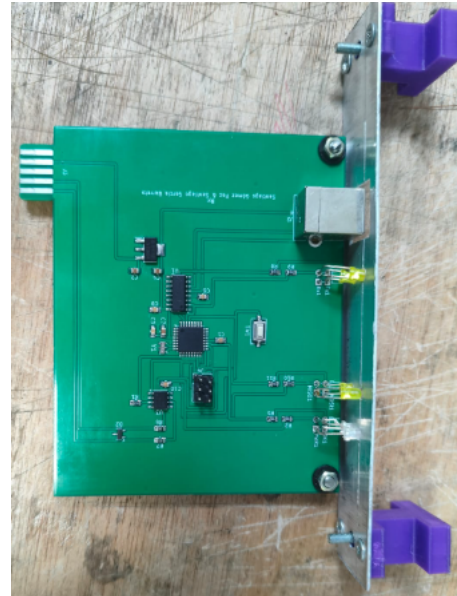


Figura 6.33: Tarjeta soldada de COM



Figura 6.34: Tarjeta soldada de DAT

La Figura 6.31 muestra las tarjetas ensambladas en su versión definitiva, evidenciando el resultado del proceso de fabricación y montaje manual descrito.

6.7.10. Diseño e implementación de la estructura mecánica del sistema

Como etapa final de integración del prototipo, se desarrolló una estructura mecánica destinada a soportar las tarjetas del sistema, organizar su interconexión mediante el backplane y proporcionar una presentación más robusta y funcional del conjunto. Esta estructura se concibió con una lógica modular inspirada en configuraciones tipo *rack*, tomando como referencia el concepto de chasis VME, pero ajustando sus dimensiones y disposición a las necesidades reales del proyecto y al tamaño adoptado para las placas electrónicas [15, 24].

El chasis se fabricó con lámina de aluminio y se definió con unas dimensiones generales aproximadas de 30 cm de largo, 15 cm de ancho y 15 cm de alto. En su interior se integró un canal central de 30 cm de largo, 12 cm de ancho y 1 cm de altura, ubicado de forma centrada respecto a la base. Sobre este canal se montaron los backplanes del sistema, los cuales constituyen la base de interconexión eléctrica y mecánica de las tarjetas. Esta disposición permitió mantener una estructura compacta, ordenada y adecuada para la interacción del usuario con el prototipo.

Tabla 6.26: Materiales y especificaciones utilizados en la construcción del chasis.

Elemento	Material	Especificación	Cantidad	Origen
Láminas laterales	Aluminio	15 × 15 cm	2	CAP-PUJ Cali
Láminas base y tapas	Aluminio	15 × 30 cm	3	CAP-PUJ Cali
Ángulos longitudinales	Aluminio 3/4 × 1/8 pulgadas	30 cm c/u, corte a 45° en los extremos	4	DISAL Cali
Ángulos laterales estructurales	Aluminio 3/4 × 1/8 pulgadas	15 cm c/u, corte a 45° en los extremos	8	DISAL Cali
Ángulos de soporte de módulos	Aluminio 3/4 × 1/8 pulgadas	15 cm c/u, corte de 3.5 cm en un extremo	10	DISAL Cali
Tornillería para ensamble del chasis	Acero, avellanado M3	3 mm	58	CAP-PUJ Cali
Tornillería para guías	Acero, avellanado M3	3 mm	16	CAP-PUJ Cali
Tornillería para ángulos de módulos	Acero, avellanado M3	3 mm	16	CAP-PUJ Cali
Tuercas	Acero, M3	3 mm	74	CAP-PUJ Cali
Piezas auxiliares 3D	PLA	Ver Anexo D	6 tipos	CAP-PUJ Cali

El proceso de fabricación del chasis se llevó a cabo íntegramente en el Centro de Ayudas Pedagógicas (CAP) de la Pontificia Universidad Javeriana Cali, utilizando los equipos y herramientas disponibles en el taller del laboratorio. Los materiales principales fueron lámina de aluminio y ángulo de aluminio crudo de 3/4 × 1/8 pulgadas, adquirido en la Distribuidora de Aluminio S.A.S. (DISAL, Cali). La tornillería utilizada consistió en tornillos avellanados M3, con un total de 90 unidades distribuidas de la siguiente manera: 58 para el ensamble general del chasis, 16 para la

fijación de las guías de inserción de tarjetas y 16 para los ángulos de soporte de los módulos.

Las láminas de aluminio se cortaron en las siguientes dimensiones: dos piezas de 15×15 cm para los laterales y tres piezas de 15×30 cm para la base y las tapas. El proceso de corte se realizó en dos etapas: inicialmente, mediante tijera de hojalatero para aproximar las dimensiones y, posteriormente, mediante la guillotina manual integrada a la dobladora Universal SBR1016 disponible en el CAP, lo que permitió obtener cortes más precisos y limpios. Los dobleces necesarios para conformar la estructura del chasis se realizaron con esta misma dobladora manual.

Los ángulos de aluminio se cortaron con una ingletadora en las siguientes piezas: cuatro ángulos de 30 cm para los rieles longitudinales de la estructura y ocho ángulos de 15 cm para los refuerzos laterales, a los cuales se les realizó un corte a 45° en cada extremo para lograr un ensamble limpio en las esquinas. Adicionalmente, se fabricaron diez ángulos de soporte de módulos de 15 cm, a los cuales se les realizó un corte de 3.5 cm en un solo extremo, a diferencia de los ángulos estructurales, donde el corte se realizó en ambos extremos. Esta modificación permitió generar la entrada necesaria para anclar cada tarjeta a la estructura.

Las perforaciones para la tornillería se realizaron con un taladro de banco utilizando una broca de 1/8 de pulgada, equivalente aproximadamente a 3 mm. Posteriormente, se aplicó un proceso de avellanado para permitir el embutido de las cabezas de los tornillos y obtener superficies sin protuberancias. El proceso de ajuste y verificación dimensional se realizó mediante un calibrador pie de rey. Después del ensamble, las superficies metálicas se lijaron para eliminar rebabas y se aplicó una protección superficial con el propósito de mejorar la presentación y la resistencia del chasis.

El proceso constructivo completo, incluyendo las etapas de diseño, prototipado iterativo y correcciones de alineación, tomó aproximadamente cuatro meses. Durante este período se realizaron múltiples ajustes en la posición de los soportes internos, la altura de las guías y el paralelismo de los ángulos, hasta lograr que la inserción de las tarjetas en los *edge connectors* del backplane fuera precisa y repetible. Este proceso iterativo es coherente con la metodología *Design Thinking* adoptada en el proyecto, específicamente en la fase de prototipado, donde cada iteración permitió identificar y corregir desviaciones geométricas antes de avanzar al ensamble definitivo.

Uno de los criterios más importantes del diseño mecánico fue lograr que las tarjetas pudieran insertarse de forma precisa en los *edge connectors* del backplane. Para ello, la estructura incorporó un sistema de guías superiores e inferiores cuya función es orientar el desplazamiento de cada tarjeta durante el montaje, evitando desviaciones y facilitando el acople con el conector correspondiente. Este sistema de guiado fue especialmente importante porque permitió mejorar la repetibilidad del ensamble y reducir el riesgo de desalineación durante la inserción de los módulos.

Adicionalmente, se emplearon soportes angulares en la parte frontal de las tarjetas, los cuales ayudaron a definir su posición dentro del chasis y aportaron rigidez al montaje. Estos elementos también permitieron incorporar las perforaciones necesarias para el acceso a conectores y elementos visibles de cada módulo, como el conector circular del sistema de sensado en la tarjeta DAT y la apertura para el conector USB en la tarjeta COM. De esta forma, la estructura no solo soporta las tarjetas, sino que también facilita su uso práctico y su interacción con el usuario final.

Para garantizar una integración adecuada entre el backplane y la estructura metálica, se utilizaron separadores que elevaron las placas respecto al canal de soporte, evitando contactos eléctricos no deseados y proporcionando la altura necesaria para que las tarjetas quedaran correctamente alineadas al momento de insertarse. De manera similar, también se incorporaron elementos de separación entre las tarjetas y los soportes metálicos, con el fin de evitar el contacto directo entre los puntos de soldadura y las superficies conductoras del chasis.

En conjunto, la estructura mecánica desarrollada permitió consolidar el sistema en una plataforma más estable, ordenada y visualmente robusta, adecuada para su uso en laboratorio y para futuras ampliaciones. Los modelos 3D de todas las piezas auxiliares de soporte, guiado y separación se documentan en detalle en el Anexo D, donde se presentan las vistas, dimensiones y especificaciones de fabricación de cada elemento.

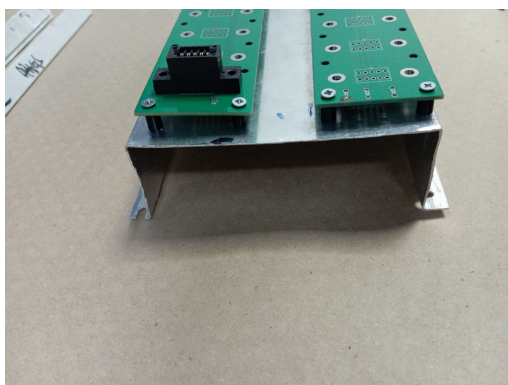


Figura 6.35: Detalle del soporte interno tipo canal

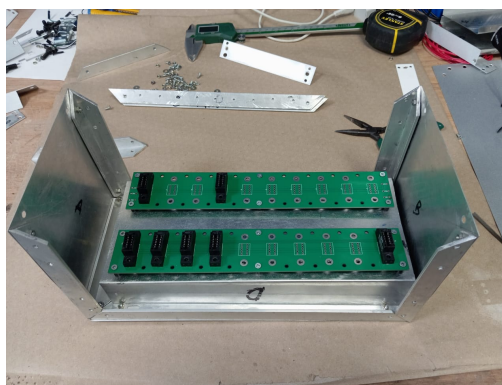


Figura 6.36: Integración interna del backplane y los soportes



Figura 6.37: Avance del montaje del chasis



Figura 6.38: Chasis mecánico ensamblado

6.8. Plataforma de visualización, registro y exportación

Una vez implementadas las etapas de adquisición, procesamiento embebido y comunicación entre módulos, fue necesario desarrollar una plataforma externa capaz de recibir, organizar, visualizar y almacenar la información generada por el sistema. Para ello, se implementó una aplicación en Python orientada a la supervisión en tiempo real de las temperaturas medidas y al registro estructurado de los datos obtenidos durante la operación del sistema [3].

La plataforma se desarrolló en Python debido a su facilidad de integración con puertos seriales, su flexibilidad para el procesamiento de datos y la disponibilidad de herramientas para visualización y exportación. La recepción de la información proveniente de la tarjeta de comunicación se realizó mediante PySerial, mientras que el manejo de estructuras tabulares y la escritura de archivos se apoyó en pandas. Adicionalmente, se contempló la exportación de datos en formatos estructurados, principalmente CSV, con el fin de facilitar su análisis posterior [24].

En cuanto a la visualización, se empleó Matplotlib para la actualización periódica de las curvas de temperatura en tiempo real. La aplicación se organizó mediante una lógica desacoplada entre la adquisición y la actualización gráfica, permitiendo recibir las líneas provenientes del puerto serial, procesar la información, actualizar los *buffers* internos y reflejar los cambios en la gráfica sin bloquear la ejecución del sistema. De forma paralela, los datos validados podían almacenarse en archivos de registro, lo que permitió combinar la supervisión en tiempo real con la trazabilidad de la información experimental.

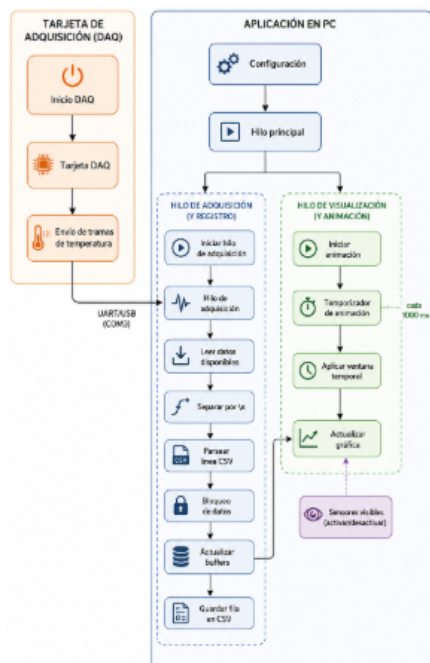


Figura 6.39: Diagrama de diseño de la plataforma de visualización, registro y exportación (Python).

La Figura 6.36 presenta el flujo general de la plataforma de visualización, registro y exportación implementada en Python. En ella se resume la interacción entre la etapa de adquisición embebida y la aplicación ejecutada en el computador, incluyendo la configuración inicial, la lectura serial, el procesamiento interno de los datos, la actualización de los *buffers*, el registro de la información en archivos CSV y la actualización de la gráfica en tiempo real.

En particular, los archivos CSV generados permiten realizar un análisis estadístico básico de las lecturas mediante herramientas como pandas o Excel, incluyendo el cálculo de promedios, desviaciones estándar y tendencias temporales para cada sensor. Asimismo, estos registros facilitan la comparación entre sensores ubicados en distintos puntos del sistema y el contraste de los perfiles de temperatura medidos con los modelos teóricos de transferencia de calor estudiados en el curso.

6.9. Integración y validación del sistema

Una vez finalizadas las etapas de diseño de hardware, desarrollo de firmware, comunicación entre módulos y desarrollo de la plataforma de visualización, se procedió a la integración y validación del sistema completo. Dado que el banco de transferencia de calor del grupo asociado se encontraba en fase de construcción durante el período de pruebas, la validación experimental se realizó sobre el prototipo ensamblado en condiciones de laboratorio, con los sensores DS18B20 conectados al sistema y sometidos a perturbaciones térmicas controladas. Esta aproximación permitió validar el funcionamiento del sistema de instrumentación de manera independiente al banco físico, lo cual resulta consistente con el alcance del proyecto y con la naturaleza modular de la arquitectura propuesta, que facilita su posterior integración al banco una vez este se encuentre finalizado.

La integración del sistema consistió en conectar la tarjeta de adquisición, la tarjeta de comunicación, la tarjeta de alimentación, el backplane y la plataforma de visualización ejecutada en el computador, con el propósito de evaluar la operación completa de la solución propuesta. En esta etapa se verificó la lectura simultánea de múltiples sensores DS18B20, la estabilidad del bus 1-Wire, la transmisión de datos mediante RS-485, la recepción de la información en el PC y su posterior representación gráfica y almacenamiento en archivos estructurados.

La validación experimental se llevó a cabo mediante pruebas funcionales orientadas a comprobar tanto la continuidad operativa del sistema como la coherencia de la información adquirida. Para ello, se consideraron criterios como la detección de sensores ausentes, la repetibilidad de las mediciones, la latencia de actualización, la consistencia del registro en archivos y la tasa de pérdida de tramas durante la transmisión. Estos criterios permitieron evaluar el sistema no solo como un conjunto de módulos independientes, sino como una solución integrada para la adquisición, transmisión, registro y visualización de temperatura en tiempo real.

En términos prácticos, las pruebas permitieron confirmar que el sistema era capaz de mantener una lectura organizada de los sensores, transmitir las temperaturas hacia el módulo maestro, enviarlas al computador y registrarlas para su posterior análisis. Asimismo, la plataforma de visualización permitió observar la evolución de las temperaturas en tiempo real, facilitando la supervisión del

comportamiento térmico durante las pruebas experimentales.

Como parte del proceso de validación, el prototipo fue presentado a uno de los usuarios potenciales del sistema, quien tuvo la oportunidad de observar su funcionamiento integral durante la sustentación. Esta instancia permitió obtener una apreciación externa sobre la utilidad y las posibilidades de aplicación futura de la plataforma desarrollada. Al respecto, el profesor manifestó:

“La herramienta desarrollada es de gran utilidad para su aplicación en distintos laboratorios de ing. Mecánica. Más allá del propósito inicial enfocado en la experimentación de fenómenos de transferencia de calor por conducción, esta plataforma será orientada en el corto plazo a otros fenómenos de transferencia de calor y hacia estudio de fuerzas en construcciones mecánicas.”

Esta apreciación evidencia que la arquitectura modular desarrollada presenta posibilidades de aplicación más allá del escenario inicialmente planteado, particularmente en otros montajes experimentales que requieran adquisición, procesamiento, registro y visualización de variables físicas. De esta manera, la validación no se limitó únicamente a comprobar el funcionamiento técnico del prototipo, sino que también permitió identificar posibilidades de reutilización y ampliación de la plataforma en futuros desarrollos académicos y de laboratorio.

Resultados y Discusión

Antes de presentar los resultados obtenidos, se describe el arreglo experimental empleado para la validación del sistema. El prototipo se instaló sobre una superficie de trabajo de laboratorio, con las tarjetas DAT, COM y POWER insertadas en el backplane e integradas dentro del chasis mecánico ensamblado. Los sensores DS18B20 se conectaron a la tarjeta DAT mediante el conector M12 y la placa de integración física del sensado descrita en la Sección 6.7.8. Por su parte, la tarjeta COM se conectó al computador mediante una interfaz USB, mientras que la plataforma de visualización desarrollada en Python se ejecutó en el mismo equipo.

Esta configuración permitió evaluar el funcionamiento conjunto de las etapas de adquisición, comunicación, visualización y registro de datos en condiciones de laboratorio. La Figura 7.1 presenta el sistema modular ensamblado y en funcionamiento durante la etapa de validación del hardware.

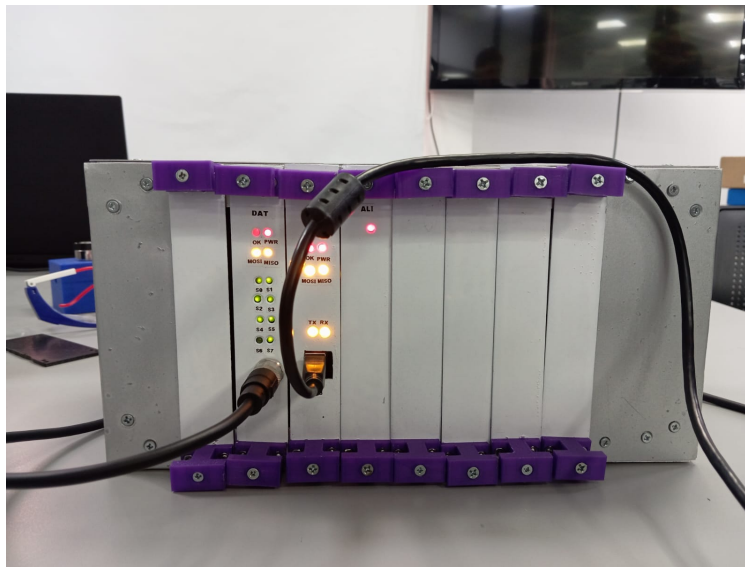


Figura 7.1: Sistema modular ensamblado y en funcionamiento.

7.1. Plan de pruebas

La Tabla 7.1 resume el conjunto de pruebas aplicadas al sistema de adquisición y comunicación. Estas pruebas permitieron verificar la detección de sensores, la lectura multisensor, la estabilidad temporal del registro, la respuesta ante perturbaciones térmicas, el intercambio de información

mediante RS-485 y el funcionamiento de la plataforma de visualización y exportación. En conjunto, este plan de pruebas sirvió como base para estructurar la validación experimental y analizar los resultados presentados en las secciones siguientes.

Tabla 7.1: Plan de pruebas para el sistema de adquisición y comunicación.

ID	Objetivo	Procedimiento (resumen)	Criterio de aceptación
PT-01	Detección de sensores DS18B20 (<i>Presence</i> + ROM)	Ejecutar la rutina 1-Wire mediante la secuencia de <i>reset/presence</i> y la identificación de la ROM de cada sensor.	Detección correcta de presencia y reconocimiento de los sensores conectados.
PT-02	Lectura multisensor (S0-S6)	Ejecutar la conversión de temperatura (<i>Convert T</i>) y la lectura del <i>Scratchpad</i> para todos los sensores.	Obtención de lecturas coherentes y dentro de la resolución esperada del sensor.
PT-03	Estabilidad en reposo	Realizar un registro continuo bajo condiciones ambientales estables y exportar los datos obtenidos a un archivo CSV.	Variación acotada de las mediciones en el tiempo y comportamiento coherente entre sensores.
PT-04	Respuesta ante perturbación térmica	Aplicar un estímulo térmico y observar la evolución de las temperaturas registradas en tiempo real.	Se observa un incremento de temperatura y su posterior evolución durante el proceso de enfriamiento o relajación térmica.
PT-05	Comunicación mediante comando/respuesta	Enviar el comando de solicitud desde el maestro y verificar la recepción de la trama de respuesta mediante RS-485.	Recepción correcta del comando de solicitud y transmisión de una trama válida de datos.
PT-06	Visualización y exportación	Visualizar las curvas de temperatura en la plataforma desarrollada en Python y verificar la generación del archivo de registro CSV.	Visualización de las curvas en tiempo real y generación correcta del archivo CSV.

7.2. Resultados del sensado con DS18B20

7.2.1. Detección de sensores y direccionamiento

En la prueba PT-01 se verificó la presencia de dispositivos sobre el bus 1-Wire mediante la rutina de *reset/presence* y la lectura del identificador ROM. La evidencia obtenida mostró un mensaje de *Presence OK* y la detección de siete sensores, lo que confirma la disponibilidad del bus y la posibilidad de direccionamiento individual mediante el identificador único de cada DS18B20. Este resultado fue importante porque permitió validar la base del arreglo multisensor sin necesidad de asignar identificadores manuales a cada canal.

```

Presence OK
ROM found: 28 BA 2F A5 00 00 00 69
ROM found: 28 96 54 A8 00 00 00 22
ROM found: 28 39 D3 A9 00 00 00 4F
ROM found: 28 08 EE A4 00 00 00 39
ROM found: 28 57 3B 57 04 E1 3C 2D
ROM found: 28 2F 9B A6 00 00 00 EB
ROM found: 28 EF 36 49 F6 D5 3C 9E
S0: Temp: 26.8750 C
S1: Temp: 26.8750 C
S2: Temp: 26.5000 C
S3: Temp: 26.7500 C
S4: Temp: 26.7500 C
S5: Temp: 26.6250 C
S6: Temp: 26.5625 C
[RS485] Comando recibido: 0xFF
[RS485] Enviando temperaturas: 26 26 26 26 26 26 26 255

Presence OK
ROM found: 28 BA 2F A5 00 00 00 69
ROM found: 28 96 54 A8 00 00 00 22
ROM found: 28 39 D3 A9 00 00 00 4F
ROM found: 28 08 EE A4 00 00 00 39
ROM found: 28 57 3B 57 04 E1 3C 2D
ROM found: 28 2F 9B A6 00 00 00 EB
ROM found: 28 EF 36 49 F6 D5 3C 9E
S0: Temp: 28.6250 C
S1: Temp: 28.6875 C
S2: Temp: 28.3125 C
S3: Temp: 28.0625 C
S4: Temp: 28.3750 C
S5: Temp: 27.8750 C
S6: Temp: 27.6875 C
[RS485] Comando recibido: 0xFF
[RS485] Enviando temperaturas: 28 28 28 28 28 27 27 255

```

(a) Lecturas alrededor de 26–27°C.

(b) Lecturas alrededor de 27–29°C.

Figura 7.2: Evidencia por MPLAB: *Presence OK*, ROM encontradas y lectura multisensor (S0–S6).

La Figura 7.2 ilustra esta etapa de detección y muestra, además, lecturas de temperatura representativas en dos condiciones térmicas distintas. En ambos casos se observa que el sistema identifica correctamente los sensores y entrega valores coherentes para los canales S0–S6, lo cual confirma el funcionamiento del direccionamiento por ROM implementado en el firmware.

7.2.2. Lectura multisensor y resolución observada

En la prueba PT-02 se registraron lecturas simultáneas para los siete sensores activos. Los valores observados presentan incrementos de 0.0625 °C, por ejemplo, 26.5625 °C y 28.6875 °C, lo cual coincide con la cuantización típica del DS18B20 en configuraciones de alta resolución. Este comportamiento confirma que la rutina de adquisición no solo detecta los sensores, sino que además preserva la resolución esperada del dispositivo durante la lectura del *scratchpad*.

Tabla 7.2: Muestra de lecturas multisensor.

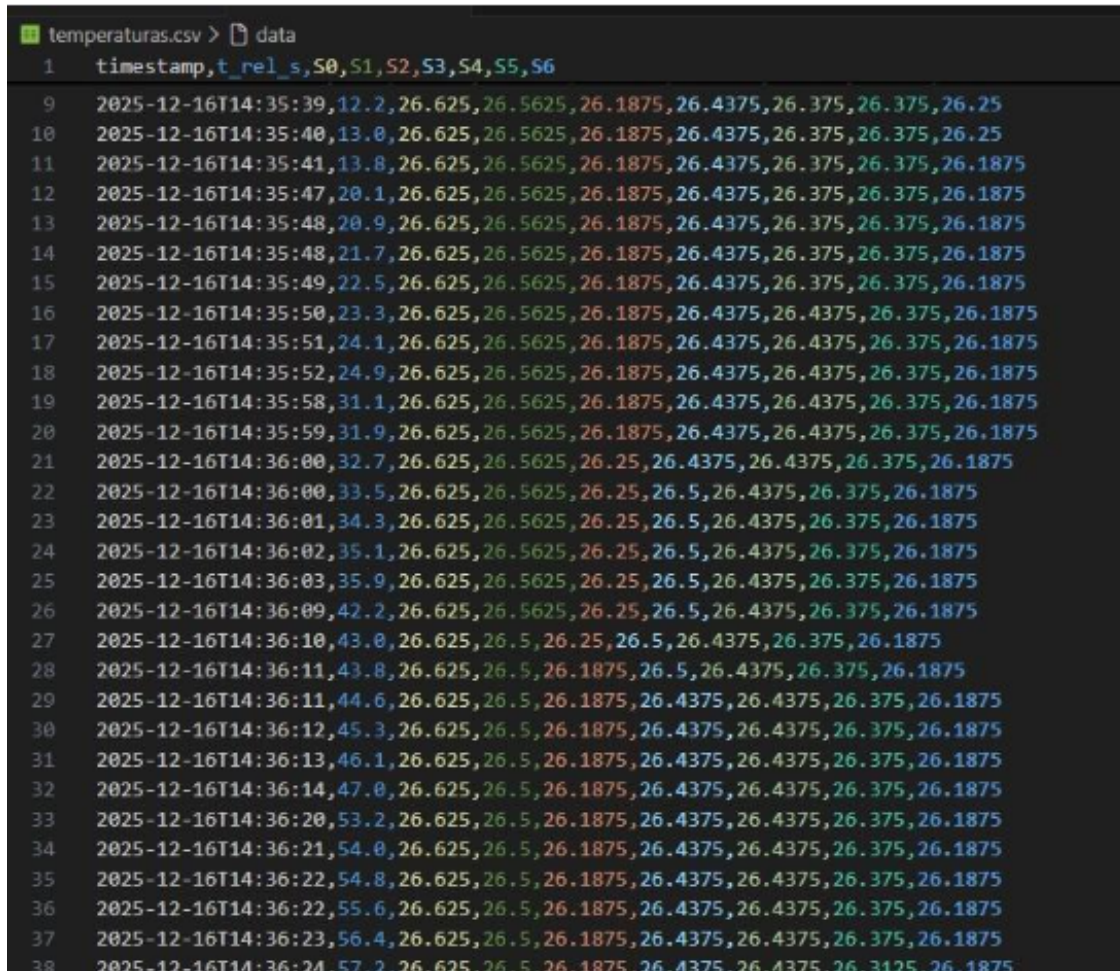
Caso	S0	S1	S2	S3	S4	S5	S6
Caso A	26.8750	26.8750	26.5000	26.7500	26.7500	26.6250	26.5625
Caso B	28.6250	28.6875	28.3125	28.0625	28.3750	27.8750	27.6875

La Tabla 7.2 resume dos casos representativos extraídos del registro. En el Caso A las temperaturas se mantienen alrededor de 26–27 °C, mientras que en el Caso B se observa un incremento general del nivel térmico, con valores cercanos a 27–29 °C. Esta comparación evidencia que el sistema responde de manera consistente a cambios en la condición térmica y mantiene coherencia entre canales dentro de una misma captura.

7.2.3. Estabilidad y registro en CSV

En la prueba PT-03 se verificó el registro continuo de datos en el archivo `temperaturas.csv`. La Figura 7.3 muestra que el formato de almacenamiento incluye una marca temporal tipo ISO, el tiempo relativo en segundos y una columna independiente para cada sensor, lo que facilita la

trazabilidad del experimento y el análisis posterior. Este resultado valida que la plataforma no solo visualiza los datos en tiempo real, sino que también los conserva en una estructura adecuada para su tratamiento posterior.



```
temperaturas.csv > data
1 timestamp,t_rel,s,S0,S1,S2,S3,S4,S5,S6
9 2025-12-16T14:35:39,12.2,26.625,26.5625,26.1875,26.4375,26.375,26.375,26.25
10 2025-12-16T14:35:40,13.0,26.625,26.5625,26.1875,26.4375,26.375,26.375,26.25
11 2025-12-16T14:35:41,13.8,26.625,26.5625,26.1875,26.4375,26.375,26.375,26.1875
12 2025-12-16T14:35:47,20.1,26.625,26.5625,26.1875,26.4375,26.375,26.375,26.1875
13 2025-12-16T14:35:48,20.9,26.625,26.5625,26.1875,26.4375,26.375,26.375,26.1875
14 2025-12-16T14:35:48,21.7,26.625,26.5625,26.1875,26.4375,26.375,26.375,26.1875
15 2025-12-16T14:35:49,22.5,26.625,26.5625,26.1875,26.4375,26.375,26.375,26.1875
16 2025-12-16T14:35:50,23.3,26.625,26.5625,26.1875,26.4375,26.4375,26.375,26.1875
17 2025-12-16T14:35:51,24.1,26.625,26.5625,26.1875,26.4375,26.4375,26.375,26.1875
18 2025-12-16T14:35:52,24.9,26.625,26.5625,26.1875,26.4375,26.4375,26.375,26.1875
19 2025-12-16T14:35:58,31.1,26.625,26.5625,26.1875,26.4375,26.4375,26.375,26.1875
20 2025-12-16T14:35:59,31.9,26.625,26.5625,26.1875,26.4375,26.4375,26.375,26.1875
21 2025-12-16T14:36:00,32.7,26.625,26.5625,26.25,26.4375,26.4375,26.375,26.1875
22 2025-12-16T14:36:00,33.5,26.625,26.5625,26.25,26.5,26.4375,26.375,26.1875
23 2025-12-16T14:36:01,34.3,26.625,26.5625,26.25,26.5,26.4375,26.375,26.1875
24 2025-12-16T14:36:02,35.1,26.625,26.5625,26.25,26.5,26.4375,26.375,26.1875
25 2025-12-16T14:36:03,35.9,26.625,26.5625,26.25,26.5,26.4375,26.375,26.1875
26 2025-12-16T14:36:09,42.2,26.625,26.5625,26.25,26.5,26.4375,26.375,26.1875
27 2025-12-16T14:36:10,43.0,26.625,26.5,26.25,26.5,26.4375,26.375,26.1875
28 2025-12-16T14:36:11,43.8,26.625,26.5,26.1875,26.5,26.4375,26.375,26.1875
29 2025-12-16T14:36:12,44.6,26.625,26.5,26.1875,26.4375,26.4375,26.375,26.1875
30 2025-12-16T14:36:12,45.3,26.625,26.5,26.1875,26.4375,26.4375,26.375,26.1875
31 2025-12-16T14:36:13,46.1,26.625,26.5,26.1875,26.4375,26.4375,26.375,26.1875
32 2025-12-16T14:36:14,47.0,26.625,26.5,26.1875,26.4375,26.4375,26.375,26.1875
33 2025-12-16T14:36:20,53.2,26.625,26.5,26.1875,26.4375,26.4375,26.375,26.1875
34 2025-12-16T14:36:21,54.0,26.625,26.5,26.1875,26.4375,26.4375,26.375,26.1875
35 2025-12-16T14:36:22,54.8,26.625,26.5,26.1875,26.4375,26.4375,26.375,26.1875
36 2025-12-16T14:36:22,55.6,26.625,26.5,26.1875,26.4375,26.4375,26.375,26.1875
37 2025-12-16T14:36:23,56.4,26.625,26.5,26.1875,26.4375,26.4375,26.375,26.1875
38 2025-12-16T14:36:24,57.2,26.625,26.5,26.1875,26.4375,26.4375,26.3125,26.1875
```

Figura 7.3: Evidencia de exportación a CSV.

A partir de este archivo se calcularon métricas por sensor, resumidas en la Tabla 7.3. En ella se observan promedios base entre aproximadamente 26.18 °C y 26.61 °C, desviaciones estándar bajas y diferencias de pico que oscilan entre 1.95 °C y 3.30 °C. En general, estas métricas muestran que el sistema fue capaz de registrar variaciones térmicas de forma estable, diferenciando tanto el nivel base como la magnitud de respuesta de cada canal.

Tabla 7.3: Métricas por sensor extraídas de `temperaturas.csv`.

Sensor	μ_{base} (°C)	σ_{base} (°C)	Mín. (°C)	Máx. (°C)	ΔT_{pico} (°C)	t_{pico} (s)	t_{rec} (s)
S0	26.6125	0.0254	26.5625	28.5625	1.9500	86.4	174.6
S1	26.5375	0.0311	26.5000	28.8750	2.3375	87.1	164.4
S2	26.2000	0.0254	26.1875	28.4375	2.2375	99.0	154.2
S3	26.4500	0.0254	26.4375	29.7500	3.3000	99.7	177.0
S4	26.4304	0.0202	26.3750	29.2500	2.8196	100.5	155.7
S5	26.3589	0.0277	26.3125	28.9375	2.5786	101.3	156.5
S6	26.1893	0.0106	26.1875	29.1250	2.9357	102.1	168.4

7.3. Resultados de comunicación

Durante la validación del sistema se emplearon dos niveles de comunicación complementarios: la UART0 como canal de diagnóstico local y la comunicación RS-485 como enlace funcional entre la placa de adquisición de datos y la placa de comunicación. Esta separación permitió, por una parte, verificar el estado interno del firmware y, por otra, validar el intercambio efectivo de información entre módulos dentro de la arquitectura propuesta.

7.3.1. Diagnóstico por UART

En la placa de adquisición de datos, la UART0 se utilizó como canal de diagnóstico durante la etapa de verificación del firmware. A través de este puerto fue posible observar mensajes asociados a la detección de presencia en el bus 1-Wire, las ROM encontradas, el mapeo de sensores por *slot* y las temperaturas medidas en cada canal. En este sentido, la UART no solo funcionó como una salida textual de depuración, sino también como una herramienta para comprobar la ejecución correcta de las rutinas de descubrimiento, asignación y lectura de sensores antes de validar la operación completa del sistema.

El firmware de adquisición realiza la búsqueda de dispositivos DS18B20 mediante el comando *Search ROM* y compara las direcciones encontradas con una tabla fija de ROMs esperadas. Estas direcciones se almacenan inicialmente en memoria de programa y luego se copian a una tabla de trabajo en RAM, con el fin de asociar cada sensor a un *slot* específico del sistema. De esta manera, el equipo puede identificar si un sensor esperado está presente, ausente o si se detectó una ROM no mapeada durante la ejecución.

Adicionalmente, se implementaron comandos de diagnóstico por UART0 para consultar y manipular temporalmente la tabla de asignaciones: L para listar las ROMs activas, D para limpiar la tabla de trabajo y R para restaurar las asignaciones por defecto desde PROGMEM. Estos comandos facilitaron la verificación del estado del sistema sin necesidad de modificar el código fuente y confirmaron que el esquema de asignación utilizado corresponde a una tabla fija de sensores esperados, en lugar de una reasignación automática en tiempo de ejecución.

7.3.2. Solicitud y respuesta sobre RS-485

La comunicación funcional entre tarjetas se implementó mediante RS-485 usando la USART1 del ATmega328PB. En esta arquitectura, la placa de comunicación actúa como maestro, mientras que la placa de adquisición opera como esclavo. El maestro envía por el bus un comando de solicitud 0x01 y, cuando el esclavo lo recibe correctamente, responde con una trama estructurada compuesta por una cabecera 0xA5 0x5A, seguida de 16 bytes de datos correspondientes a ocho temperaturas codificadas como enteros de 16 bits en formato *little-endian* y, finalmente, un byte de verificación CRC8 Dallas calculado sobre los primeros 18 bytes de la trama.

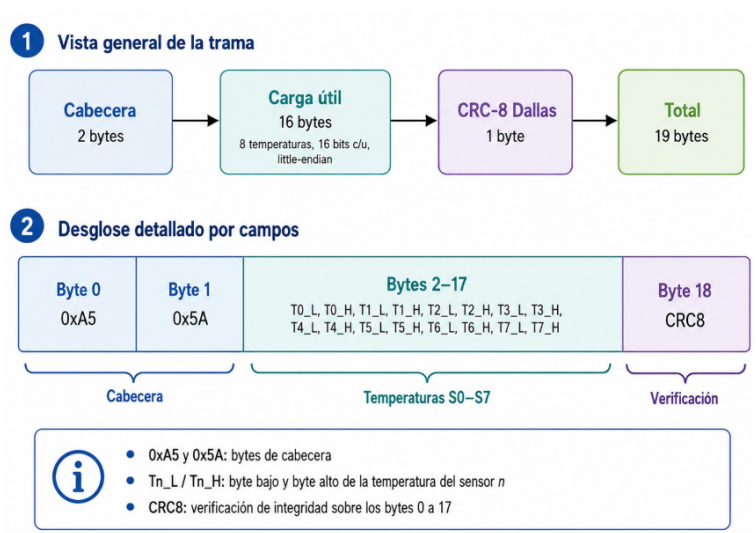


Figura 7.4: Estructura general de la trama de respuesta implementada en el sistema.

La Figura 7.4 presenta la estructura general de la trama de respuesta implementada para el intercambio de información entre las tarjetas DAT y COM. La organización de los campos permite mantener una longitud fija de 19 bytes y facilita tanto la sincronización como la validación de la información recibida.

Tabla 7.4: Campos de la trama de respuesta RS-485.

Campo	Tamaño	Descripción
Cabecera	2 bytes	Bytes de sincronización 0xA5 y 0x5A.
Datos de temperatura	16 bytes	Ocho temperaturas, con 2 bytes asignados a cada sensor.
CRC8	1 byte	Verificación de integridad calculada sobre los primeros 18 bytes de la trama.

Cada temperatura se transmite como un valor entero en décimas de grado Celsius. Por ejemplo, un valor de 253 corresponde a 25.3 °C. Si un sensor no está presente o su lectura no es válida, el

esclavo envía el valor centinela -9999 , el cual es interpretado por el maestro como un canal sin una lectura válida. Esta estrategia permitió representar explícitamente la ausencia de información sin modificar la estructura fija de la trama.

En la placa de comunicación, el firmware espera la cabecera `0xA5 0x5A`, recibe los 16 bytes de datos y verifica el CRC recibido. Si la validación es correcta, los valores se transforman en una línea en formato CSV enviada por UART0 hacia el computador; en caso contrario, la trama se descarta. Este mecanismo demuestra que la comunicación entre módulos no se limita a una conexión física, sino que implementa un flujo funcional completo de solicitud, respuesta, validación y entrega de datos para su visualización y registro externo.

7.3.3. Indicadores LED de estado

Como complemento al diagnóstico textual y al intercambio por RS-485, el sistema incorpora indicadores LED para facilitar la validación visual durante las pruebas. En la placa de adquisición, cada *slot* de sensor cuenta con un LED asociado, el cual se enciende cuando el sensor esperado está presente en el bus 1-Wire y se apaga cuando no se detecta. Además, se dispone de un LED OK que indica la presencia general del bus y el funcionamiento básico del proceso de adquisición.

En la placa de comunicación también se implementó un LED OK, el cual se activa cuando el maestro recibe una trama válida desde el esclavo y se desactiva si no se recibe respuesta o si la comunicación falla. De esta manera, los indicadores LED complementan la información entregada por la UART y permiten una comprobación rápida del estado del sistema durante la integración y validación experimental.

7.4. Resultados de visualización en tiempo real

En las pruebas PT-04 y PT-06 se validó la interfaz desarrollada en Python para la representación en tiempo real de la temperatura medida por los sensores S0–S6. La Figura 7.6 presenta dos registros representativos en los que puede observarse un incremento transitorio de temperatura, seguido de un retorno progresivo hacia un estado cercano al equilibrio. Este comportamiento es consistente con la respuesta esperada ante una perturbación térmica aplicada sobre el sistema.

Además de evidenciar el comportamiento térmico registrado durante las pruebas, la interfaz permitió comprobar que la información adquirida por el sistema embebido era efectivamente recibida, procesada y graficada en tiempo real. En este sentido, la visualización en Python no solo funcionó como herramienta de observación, sino también como mecanismo de validación del flujo completo de adquisición, transmisión y representación de datos.

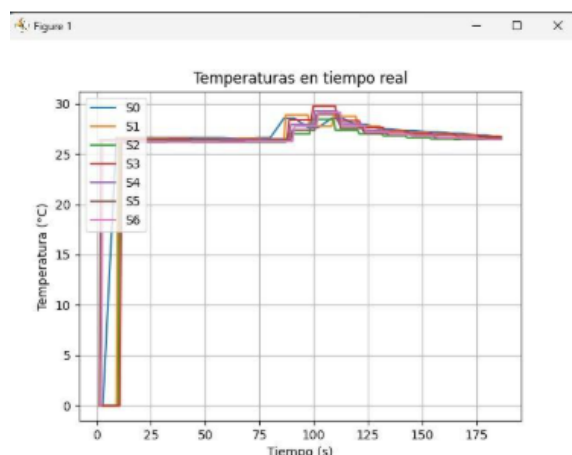


Figura 7.5: Registro de temperatura y retorno al equilibrio en aproximadamente 180 s.

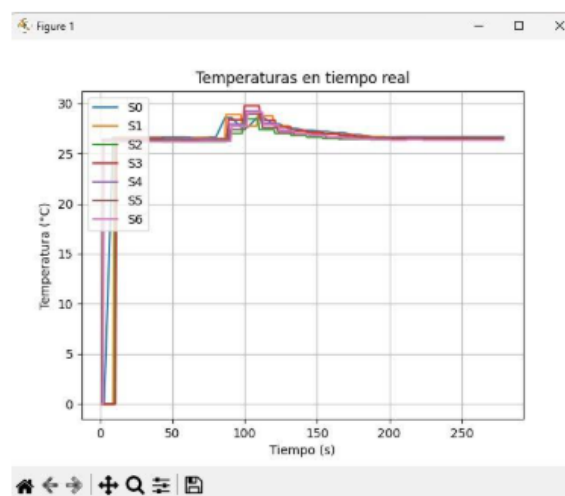


Figura 7.6: Registro extendido de temperatura y retorno al equilibrio en aproximadamente 280 s.

7.5. Hallazgos, limitaciones y trabajo futuro

7.5.1. Hallazgos principales

Un hallazgo adicional relevante corresponde a la viabilidad económica de la solución desarrollada. A diferencia de los sistemas comerciales de adquisición de temperatura disponibles en el mercado, el sistema propuesto fue implementado con componentes de disponibilidad estándar, adquiridos a través de distribuidores electrónicos convencionales, y con tarjetas PCB fabricadas mediante servicios de prototipado accesibles. El desglose completo de los costos reales del proyecto, verificados mediante las facturas y comprobantes de compra correspondientes, se presenta en el Anexo B.

El costo total del prototipo, considerando los componentes electrónicos, la fabricación de las tarjetas PCB en dos iteraciones y los materiales empleados en la construcción de la estructura mecánica, fue de USD 262,14, equivalente aproximadamente a COP 1.101.000 para una tasa representativa de referencia de COP 4.200 por dólar. La Tabla 7.5 presenta el resumen de costos por categoría.

Tabla 7.5: Resumen de costos del sistema desarrollado por categoría.

Categoría	Subtotal (USD)	Subtotal (COP aprox.)
A. Componentes electrónicos	\$128,26	\$538.700
B. Fabricación de tarjetas PCB (JLCPCB)	\$63,76	\$267.800
C. Estructura mecánica del chasis	\$70,12	\$294.500
TOTAL SISTEMA DESARROLLADO	\$262,14	\$1.101.000

Este valor resulta considerablemente inferior al de distintas alternativas comerciales de adquisición de datos y temperatura. Como se observa en la Tabla 7.6, las soluciones comerciales consideradas presentan costos superiores y, en algunos casos, requieren módulos, sensores o licencias adicionales para alcanzar una funcionalidad comparable.

Tabla 7.6: Comparación del sistema desarrollado con soluciones comerciales.

Solución	Fabricante	Precio (USD)	Características principales
NI USB-6001 [44].	National Instruments	\$344,00	DAQ multifunción USB con ocho entradas analógicas. No está orientado específicamente a la medición de temperatura y puede requerir instrumentación o software adicional según la aplicación.
Omega OM-USB-TEMP-AI [45]	Omega Engineering	\$1.345,00	Sistema de adquisición de temperatura por USB con ocho canales y compatibilidad con termopares, RTD y termistores. Corresponde a una solución comercial cerrada y sin la arquitectura modular desarrollada en este proyecto.
NI cDAQ-9171 + NI-9211[46]	National Instruments	\$1.850,00	Configuración basada en un chasis compacto y un módulo de adquisición de temperatura de cuatro canales. Ofrece mayores capacidades de instrumentación, pero presenta un costo considerablemente superior para la aplicación desarrollada.
Sistema propuesto	PUJ Cali – Trabajo de grado	\$262,14	Ocho canales de temperatura simultáneos, arquitectura modular ampliable por tarjetas, comunicación RS-485, firmware y software adaptables, exportación automática a CSV y compatibilidad con herramientas como Python y Excel.

La comparación evidencia que el sistema desarrollado presenta una ventaja económica significativa frente a las soluciones comerciales orientadas específicamente a la adquisición de temperatura. Tomando como referencia el costo total de USD 262,14, el prototipo representa una reducción aproximada del 80 % frente al Omega OM-USB-TEMP-AI y del 86 % frente a la configuración NI cDAQ-9171 con el módulo NI-9211.

Adicionalmente, el sistema propuesto incorpora características relevantes para un entorno académico, entre ellas una arquitectura modular ampliable mediante tarjetas, acceso al código fuente para su modificación y adaptación, comunicación diferencial RS-485 integrada y compatibilidad con herramientas de análisis de uso extendido, como Python y Excel, mediante la generación automática de archivos CSV. Estas características facilitan el mantenimiento, la modificación y la futura

ampliación del sistema de acuerdo con las necesidades de nuevas prácticas de laboratorio.

En conjunto, los resultados económicos muestran que fue posible desarrollar una plataforma funcional de adquisición, comunicación y visualización de temperatura con un costo inferior al de las alternativas comerciales consideradas. El desglose detallado de los componentes, materiales, costos de fabricación y comprobantes asociados se presenta en el Anexo B.

7.5.2. Limitaciones

Las evidencias presentadas en este capítulo corresponden principalmente a registros y capturas representativas del funcionamiento del sistema. Aunque estas permiten demostrar el comportamiento general del prototipo, no constituyen por sí solas una caracterización exhaustiva de toda la información obtenida durante las pruebas. A continuación, se describen las principales limitaciones identificadas y las razones técnicas por las cuales no fueron abordadas dentro del alcance del presente trabajo.

En primer lugar, no se realizó un procesamiento estadístico exhaustivo sobre la totalidad de los datos almacenados en los archivos CSV. Esta limitación responde a que las pruebas PT-03 y PT-04 fueron diseñadas para evaluar el funcionamiento del sistema bajo condiciones representativas de estabilidad en reposo y respuesta ante una perturbación térmica, y no para caracterizar estadísticamente el arreglo de sensores durante períodos prolongados. Un análisis de este tipo habría requerido establecer un protocolo experimental de mayor duración, con condiciones térmicas controladas, repetibles y comparables entre diferentes ensayos, lo cual excedía el alcance experimental del trabajo y las condiciones disponibles en el laboratorio durante la fase de validación.

En segundo lugar, no se cuantificó experimentalmente la latencia de extremo a extremo del sistema, entendida como el tiempo transcurrido desde la adquisición de una medición de temperatura hasta su actualización en la interfaz gráfica desarrollada en Python. La medición precisa de este parámetro requiere instrumentación que permita identificar y correlacionar temporalmente los diferentes eventos involucrados en la cadena de adquisición, entre ellos la conversión del sensor, la lectura mediante el bus 1-Wire, el procesamiento local, la transmisión por RS-485, la recepción en la tarjeta COM, el envío mediante la interfaz USB-serial y la actualización de la gráfica.

Durante la fase de validación no se realizó esta caracterización mediante instrumentación específica, como un osciloscopio digital o un analizador lógico. No obstante, a partir de los tiempos de operación configurados en el sistema y de las características del DS18B20, se estimó que la respuesta global se encuentra en el orden de un segundo, siendo el tiempo de conversión del sensor en resolución de 12 bits uno de los principales componentes temporales del proceso. Esta estimación se consideró suficiente para la aplicación desarrollada, cuya dinámica térmica es considerablemente más lenta que la velocidad de actualización del sistema.

En tercer lugar, no se cuantificó formalmente la tasa de pérdida o error de tramas durante la comunicación RS-485 en pruebas de larga duración. La medición rigurosa de este parámetro requeriría incorporar mecanismos adicionales de registro en el firmware, tales como contadores acumulativos

de tramas con CRC inválido, eventos de *timeout* y pérdidas de sincronización. Estos mecanismos permitirían realizar una caracterización estadística del comportamiento del enlace durante sesiones extendidas de operación.

Esta funcionalidad no fue incluida en la versión actual del firmware, debido a que se encontraba fuera del alcance de las pruebas definidas en el plan de validación, las cuales priorizaron la comprobación funcional de la adquisición, transmisión, recepción y visualización de los datos. Durante las pruebas realizadas no se observaron pérdidas de trama que afectaran de manera perceptible la operación del sistema; sin embargo, esta observación cualitativa no sustituye una medición cuantitativa formal de la tasa de error del enlace.

En conjunto, estas limitaciones no impiden verificar el cumplimiento de los objetivos planteados ni la funcionalidad del prototipo desarrollado. Sin embargo, delimitan el alcance de la validación experimental realizada y permiten identificar aspectos que pueden ser abordados posteriormente para lograr una caracterización más completa del desempeño del sistema.

Conclusiones

8.1. Conclusiones

El desarrollo de este trabajo permitió diseñar e implementar un sistema embebido orientado a la adquisición, transmisión, visualización y registro de temperatura para su aplicación en un banco de transferencia de calor, cumpliendo con el objetivo general propuesto. En particular, se logró la lectura simultánea de múltiples sensores DS18B20 conectados mediante un bus 1-Wire, la transferencia de los datos hacia una plataforma externa y su visualización y registro en tiempo real, dejando la información disponible en formato CSV para su posterior análisis. Los resultados obtenidos evidencian que el enfoque modular planteado constituye una alternativa viable de instrumentación para entornos académicos. El prototipo alcanzó un costo de implementación aproximado de USD 262,14, inferior al de las soluciones comerciales consideradas en la Tabla 7.6, y ofrece además capacidad de expansión mediante tarjetas adicionales y potencial de adaptación a otros desarrollos académicos.

Con respecto al primer objetivo específico, se concluye que el sensor DS18B20 fue una elección adecuada para el contexto del proyecto debido a su interfaz de comunicación 1-Wire, su identificador único por dispositivo y la posibilidad de integrar múltiples sensores sobre un mismo bus, reduciendo el cableado y favoreciendo la escalabilidad del sistema. No obstante, también se comprobó que el funcionamiento del bus depende de aspectos prácticos como la resistencia de *pull-up*, la longitud y disposición del cableado y las condiciones de ruido eléctrico. Por esta razón, la implementación del protocolo a bajo nivel y la incorporación de mecanismos de verificación de integridad mediante CRC resultaron fundamentales para detectar lecturas inválidas y mejorar la confiabilidad del proceso de adquisición.

En relación con el segundo objetivo específico, la arquitectura modular basada en las tarjetas POWER, DAT, COM y el backplane permitió separar las funciones de alimentación, adquisición y comunicación, facilitando las tareas de integración, depuración y mantenimiento del sistema. Asimismo, la incorporación de un enlace diferencial basado en RS-485 permitió establecer una comunicación robusta entre los módulos de adquisición y comunicación. La experiencia de implementación también evidenció que el desempeño de este tipo de enlace depende de una adecuada consideración de aspectos como la terminación, la polarización y la topología del bus, los cuales deben ser tenidos en cuenta en futuras ampliaciones de la arquitectura.

En cuanto al tercer objetivo específico, se logró desarrollar un prototipo funcional tanto a nivel de hardware como de firmware mediante un proceso incremental que inició con validaciones preliminares y evolucionó hacia una implementación basada en la documentación técnica de los dispositivos y el acceso directo a los registros del microcontrolador. Este proceso permitió ejercer un mayor

control sobre la temporización y el comportamiento del firmware. Como resultado, se comprobó que, en protocolos temporizados como 1-Wire, el manejo preciso de las secuencias de *reset/presence*, conversión, lectura del *scratchpad* y verificación mediante CRC es fundamental para mantener una adquisición repetible y detectar condiciones de fallo en un arreglo multisensor.

Respecto al cuarto objetivo específico, la plataforma de visualización desarrollada en Python permitió representar la evolución de las temperaturas en tiempo real y generar archivos CSV para su análisis posterior. Esto facilitó la trazabilidad experimental y confirmó que la información adquirida por el sistema embebido puede ser transmitida, recibida, interpretada y almacenada correctamente en una aplicación externa. Asimismo, la definición de una estructura de trama, la incorporación de mecanismos de validación y el establecimiento de un plan de pruebas permitieron evaluar de manera organizada el flujo completo de información, desde la adquisición de las temperaturas hasta su representación en el computador.

Finalmente, el desarrollo del proyecto permitió identificar aprendizajes relevantes relacionados con la integración entre hardware, firmware y estructura mecánica. El paso desde las pruebas iniciales hasta la fabricación de las PCB y su posterior integración en un chasis modular evidenció la importancia de considerar aspectos como la selección y verificación de huellas, la compatibilidad mecánica de los conectores, la alineación entre módulos y la validación iterativa del diseño. El inconveniente presentado durante la primera iteración del *edge connector* demostró que una incompatibilidad mecánica puede afectar la integración del sistema aun cuando su funcionamiento electrónico sea correcto, resaltando la importancia de verificar conjuntamente los aspectos eléctricos y mecánicos antes de la fabricación.

En conjunto, el proyecto permitió consolidar una plataforma modular y funcional para la adquisición de temperatura con fines académicos, integrando sensado, procesamiento embebido, comunicación, visualización, registro y una estructura física propia. La solución desarrollada constituye una base técnica y metodológica que puede ser ampliada y adaptada en futuros trabajos orientados a la instrumentación de bancos de laboratorio y a la enseñanza experimental de fenómenos térmicos.

Recomendaciones

9.1. Recomendaciones

Con el fin de mejorar el desempeño, la confiabilidad y la proyección académica del sistema desarrollado, se recomienda implementar un procedimiento formal de calibración y verificación periódica del arreglo de sensores. Este procedimiento podría realizarse mediante la comparación de las lecturas obtenidas con un instrumento de referencia en diferentes condiciones de temperatura, incluyendo, por ejemplo, temperatura ambiente y otros puntos térmicos controlados dentro del rango de operación previsto. De esta manera, sería posible cuantificar parámetros como el error, la repetibilidad y la posible deriva de las mediciones, así como identificar sensores con comportamientos atípicos antes de su utilización en prácticas de laboratorio [10].

En relación con la plataforma de visualización, se recomienda incorporar funciones orientadas a facilitar su utilización en contextos académicos, tales como la selección dinámica del puerto y de la velocidad de comunicación, indicadores del estado del enlace, almacenamiento automático de archivos mediante una nomenclatura estandarizada por práctica, fecha o sesión experimental, y opciones adicionales de exportación de datos. Asimismo, sería conveniente documentar de manera detallada los requisitos de instalación y configuración del entorno de software y, en una etapa posterior, desarrollar una versión ejecutable o un instalador que reduzca la complejidad de uso para estudiantes y usuarios no familiarizados con Python [16, 17, 47, 48].

Desde el punto de vista del hardware, se recomienda fortalecer el proceso de verificación previo a la fabricación de futuras versiones mediante una revisión cruzada entre el esquemático y el diseño de la PCB, la comprobación dimensional de las huellas con respecto a las hojas de datos y los componentes físicos disponibles, el uso sistemático de las herramientas ERC y DRC, y la implementación de listas de verificación para señales críticas, alimentación e interfaces de comunicación. Esta recomendación adquiere especial relevancia a partir de la experiencia obtenida con la primera iteración del *edge connector*, cuya incompatibilidad dimensional afectó la integración física del sistema. Para futuras versiones también sería conveniente incorporar puntos de prueba y rutinas de autodiagnóstico que faciliten la validación independiente de cada tarjeta antes de su integración al backplane [15, 49].

Como extensión de las pruebas realizadas, se recomienda desarrollar una metodología de validación cuantitativa de mayor duración. Esta podría incluir el procesamiento estadístico completo de los registros generados, la medición experimental de la latencia de extremo a extremo y la incorporación de contadores de errores de comunicación para cuantificar tramas con CRC inválido,

pérdidas de sincronización y eventos de *timeout*. Estas mejoras permitirían complementar la validación funcional presentada en este trabajo con una caracterización más exhaustiva del desempeño temporal y de la confiabilidad del sistema.

En cuanto a la estructura mecánica y la integración modular, se recomienda establecer y documentar formalmente un estándar de interconexión que defina el pinout, las dimensiones mecánicas y las señales disponibles en los conectores de borde. Esto permitiría mantener la compatibilidad entre las tarjetas actuales y futuros módulos desarrollados para la plataforma. De igual forma, sería pertinente definir procedimientos básicos de mantenimiento preventivo, incluyendo la inspección y limpieza de conectores, la verificación de la alineación de las guías, el ajuste de la tornillería y la revisión periódica del estado general de las tarjetas y del chasis.

Finalmente, para facilitar la continuidad, reproducibilidad y escalabilidad del proyecto, se recomienda mantener el código fuente del firmware y de la plataforma de visualización, los esquemáticos, los archivos de diseño de las PCB, los modelos de las piezas mecánicas y la documentación técnica en un repositorio con control de versiones y respaldos periódicos. Esta práctica permitiría conservar la trazabilidad de los cambios realizados y facilitaría que futuros estudiantes o grupos de trabajo puedan reproducir, mantener y ampliar la plataforma desarrollada para su aplicación en nuevos bancos de laboratorio y experiencias académicas.

Bibliografía

- [1] H. A. Abdul Wahhab, I. A. A. Mahmood, and A. Akroot, “Analysis of the characteristics and mechanism of forced convection heat transfer for a heated pipe in a tank,” [En línea]. Disponible en: <https://ouci.dntb.gov.ua/en/works/98eRZRw7/>, [Accedido: 2025].
- [2] M. A. Zárate-Navarro, S. D. Schiavone-Valdez, J. E. Cuevas, W. M. Warren-Vega, A. Campos-Rodríguez, and L. A. Romero-Cano, “STEM activities for heat transfer learning: Integrating simulation, mathematical modeling, and experimental validation in transport phenomena education,” *Education for Chemical Engineers*, vol. 49, pp. 81–90, 2024.
- [3] N. Mahanta, U. S. Dixit, and J. P. Davim, “A pedagogical gadget for teaching heat transfer,” in *Engineering Pedagogy*, J. P. Davim, Ed. Singapore: Springer, 2023.
- [4] Ansys, “What is heat transfer?” Ansys, Inc. [En línea]. Disponible en: <https://www.ansys.com/simulation-topics/what-is-heat-transfer>, [Accedido: 2025].
- [5] OMEGA Engineering, “Sistema de adquisición de datos,” OMEGA Engineering. [En línea]. Disponible en: <https://es.omega.com/prodinfo/adquisicion-de-datos.html>, [Accedido: 2025].
- [6] C. Wei, J. L. Tabjula, J. Sharma, and Y. Chen, “The modeling of two-way coupled transient multiphase flow and heat transfer during gas influx management using fiber optic distributed temperature sensing measurements,” *International Journal of Heat and Mass Transfer*, vol. 214, p. 124447, 2023.
- [7] Acromag, Inc., *AVME9630: VMEbus 3U IP carrier*, Manual/datasheet. [En línea]. Disponible en: <https://www.acromag.com/wp-content/uploads/2022/04/Acromag-AVME9630-VMEbus-3U-IP-Carrier-.pdf>, [Accedido: 2025].
- [8] Xycom Automation / XVME, *XVME-9630: VMEbus 3U carrier (Size A) (DS-79630(A))*, Datasheet. [En línea]. Disponible en: [http://www.xycomkorea.com/pdfs/datasheets/xvme/ds-79630\(a\).pdf](http://www.xycomkorea.com/pdfs/datasheets/xvme/ds-79630(a).pdf), [Accedido: 2025].
- [9] Smiths Interconnect, “¿qué es la simulación térmica y por qué es importante para las pruebas de rodaje de fiabilidad?” Smiths Interconnect Blog. [En línea]. Disponible en: <https://es.smithsinterconnect.com/smiths-interconnect-blog/what-is-thermal-simulation-and-why-is-it-important-to-reliability-burn-in-testing/>, [Accedido: 2025].
- [10] Maxim Integrated (Analog Devices), *DS18B20: Programmable resolution 1-Wire digital thermometer*, rev. 6 ed., Datasheet. [En línea]. Disponible en: <https://www.analog.com/media/en/technical-documentation/data-sheets/ds18b20.pdf>, 2019.

- [11] Microchip Technology Inc., *ATmega48A/PA/88A/PA/168A/PA/328/P megaAVR Data Sheet*, [En línea]. Disponible en: <https://ww1.microchip.com/downloads/aemDocuments/documents/MCU08/ProductDocuments/DataSheets/ATmega48A-PA-88A-PA-168A-PA-328-P-DS-D S40002061B.pdf>, [Accedido: 2025].
- [12] Texas Instruments, *THVD1400, THVD1420: 3.3-V to 5-V RS-485 transceivers*, rev. b ed., Datasheet. [En línea]. Disponible en: <https://www.ti.com/lit/ds/symlink/thvd1420.pdf>, 2021.
- [13] Eltima Software, “Guía de comunicación RS-485,” [En línea]. Disponible en: <https://www.eltima.com/es/article/rs485-communication-guide/>, 2021, [Accedido: 2025].
- [14] Texas Instruments, “RS-485/RS-422: Application report (SLLA545),” [En línea]. Disponible en: <https://www.ti.com/lit/wp/slla545/slla545.pdf>, Tech. Rep., [Accedido: 2025].
- [15] VMEbusdirect, “VME card dimensions,” VMEbusdirect.com. [En línea]. Disponible en: <https://vmebusdirect.com/images/vme-card-dimensions.jpg>, [Accedido: 2025].
- [16] R. Smith, “Incorporating Arduino Nano circuit into a single PCB,” Reddit r/PrintedCircuitBoard. [En línea]. Disponible en: <https://www.reddit.com/r/PrintedCircuitBoard/>, [Accedido: 2025].
- [17] The KiCad Team, “Comenzando en KiCad (v5.1, documentación en español),” [En línea]. Disponible en: https://docs.kicad.org/5.1/es/getting_started_in_kicad/getting_started_in_kicad.html, [Accedido: 2025].
- [18] E. Cabrera Estupiñán, R. Pérez Leira, J. Domínguez Gutiérrez, M. Alemán García, and M. A. Jara Ponce, *Guía metodológica para el trabajo final de titulación en la modalidad “Proyecto de investigación”*. Manta, Ecuador: Universidad Laica Eloy Alfaro de Manabí, 2023, [En línea]. Disponible en: <https://carreras.uleam.edu.ec/jurisprudencia/wp-content/uploads/sites/40/2023/05/GUIA-PROYECTO-DE-INVESTIGACION.pdf>.
- [19] Editorial Universidad Distrital Francisco José de Caldas, “Biohacking social: una estrategia pedagógica para el fortalecimiento de la educación y el fomento de la alfabetización científica en Medellín, Colombia,” *Revista Científica*, 2023, [En línea]. Disponible en: <https://revistas.udistrital.edu.co/index.php/revcie/article/view/20781>.
- [20] N. Kyriazis, G. Stylos, and K. Kotsis, “Impact of inquiry-based laboratory activities on understanding heat concepts and self-efficacy in pre-service teachers,” *Journal of Pedagogical Research*, vol. 9, no. 1, pp. 161–181, 2025.
- [21] K. Kotsis, “The significance of experiments in inquiry-based science teaching,” *European Journal of Education and Pedagogy*, vol. 5, no. 2, pp. 86–92, 2024.
- [22] B. R. Wilcox and H. J. Lewandowski, “Open-ended versus guided laboratory activities: Impact on students’ beliefs about experimental physics,” *Physical Review Physics Education Research*, vol. 12, no. 2, p. 020132, Oct. 2016.

- [23] R. A. Velásquez, J. V. Mejía Lara, and R. A. Velásquez, “Design methodology for heat transfer system with low-cost perspective,” in *2023 IEEE XXX International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, Lima, Peru, 2023, pp. 1–7.
- [24] U.S. Technologies VMEbus, “What is VME?” VMEbusdirect.com. [En línea]. Disponible en: <https://vmebusdirect.com/what-is-vme.php>, [Accedido: 2025].
- [25] International Electrotechnical Commission, *IEC 61010-1: Safety Requirements for Electrical Equipment for Measurement, Control, and Laboratory Use—Part 1: General Requirements*, [En línea]. Disponible en: <https://cdn.standards.iteh.ai/samples/12374/0585155bf3754df693b2fe1aa91e0938/IEC-61010-1-2010.pdf>, Std., 2017.
- [26] Presidencia de la República de Colombia, “Decreto 1072 de 2015: Decreto único reglamentario del sector trabajo (SG-SST),” [En línea]. Disponible en: <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=72173>, [Accedido: 2025].
- [27] J. Bernhard, “What matters for students’ learning in the laboratory? Do not neglect the role of experimental equipment!” *Instructional Science*, vol. 46, pp. 819–846, 2018.
- [28] A. K. Harvey, “Strategies for integrating the Internet of Things in educational institutions,” Ph.D. dissertation, Walden University, 2021, [En línea]. Disponible en: <https://scholarworks.waldenu.edu/dissertations/9864>.
- [29] R. Meylani, “Transforming education with the internet of things: A journey into smarter learning environments,” *International Journal of Research in Education and Science*, vol. 10, no. 1, pp. 161–178, 2024.
- [30] SD Industrial, “Sistemas embebidos: ¿qué son y para qué sirven?” SD Industrial Blog. [En línea]. Disponible en: <https://sdindustrial.com.mx/blog/sistemas-embebidos/>, [Accedido: 2025].
- [31] Rechner Sensors, “El sensor de temperatura,” Rechner Sensors. [En línea]. Disponible en: <https://www.rechner-sensors.com/es/documentacion/knowledge/el-sensor-de-temperatura>, [Accedido: 2025].
- [32] Ministerio de Minas y Energía, “Reglamento técnico de instalaciones eléctricas (RETIE),” Ministerio de Minas y Energía, Colombia. [En línea]. Disponible en: <https://www.minenergia.gov.co/es/misional/energia-electrica-2/reglamentos-tecnicos/reglamento-t%C3%A9cnico-de-instalaciones-el%C3%A9ctricas-retie/>, [Accedido: 2025].
- [33] Congreso de Colombia, “Ley 1672 de 2013: Gestión integral de residuos de aparatos eléctricos y electrónicos (RAEE),” [En línea]. Disponible en: <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=53825>, [Accedido: 2025].
- [34] Departamento Nacional de Planeación, Ministerio de Tecnologías de la Información y las Comunicaciones, and Departamento Administrativo de la Presidencia de la República, “Documento

- CONPES 3975: Política nacional para la transformación digital e inteligencia artificial,” Gobierno de Colombia, Tech. Rep., [En línea]. Disponible en: <http://hdl.handle.net/20.500.12324/36742>.
- [35] F. Segura, V. Bartolucci, and J. M. Andújar, “Hardware/software data acquisition system for real time cell temperature monitoring in air-cooled polymer electrolyte fuel cells,” *Sensors*, vol. 17, no. 7, p. 1600, 2017.
- [36] V. Smolaninovs and M. Terauds, “Microcontroller-based electronic laboratory measurement device for distance education,” *Electronics*, vol. 14, no. 3, p. 438, 2025.
- [37] A. A. Argiriou and O. Panagopoulos, “Low-cost data acquisition system for solar thermal collectors,” *Electronics*, vol. 11, no. 6, p. 934, 2022.
- [38] M. F. B. Iqbal, R. Abbasi, F. Sultan, and S. Bano, “Design and implementation of a multi-channel temperature measurement system based on LabVIEW,” *International Journal of Current Science Research and Review*, vol. 6, no. 6, pp. 3091–3109, 2023, [En línea]. Disponible en: <https://ijcsrr.org/wp-content/uploads/2023/06/02-01-2023.pdf>.
- [39] Salahuddin *et al.*, “Implementation of DS18B20 temperature sensor in ESP32 and I2C LCD based IoT monitoring system for mechatronics laboratory,” *International Journal of Advanced Multidisciplinary*, 2024, [En línea]. Disponible en: <https://greenpub.org/IJAM/article/view/1988>.
- [40] Microchip Technology Inc., *MPLAB X IDE User’s Guide (DS50002027D)*, [En línea]. Disponible en: <https://ww1.microchip.com/downloads/en/DeviceDoc/50002027D.pdf>, 2015.
- [41] The KiCad Team, *PCB Editor (KiCad 9.0 Documentation) — pcbnew.pdf*, [En línea]. Disponible en: <https://docs.kicad.org/9.0/en/pcbnew/pcbnew.pdf>, [Accedido: 2025].
- [42] Mean Well, *EPS-15 Series: 15W Single Output Switching Power Supply*, Datasheet. [En línea]. Disponible en: <https://www.meanwell.com/Upload/PDF/EPS-15/EPS-15-SPEC.PDF>, [Accedido: 2025].
- [43] JLCPCB, “PCB capabilities (fabrication capabilities),” [En línea]. Disponible en: <https://jlcpcb.com/capabilities/Capabilities>, [Accedido: 2025].
- [44] National Instruments, “USB-6001 Multifunction I/O Device,” National Instruments. [En línea]. Disponible en: <https://www.ni.com/en/shop/hardware/voltage/model-usb-6001>, [Accedido: 2026].
- [45] Omega Engineering, “OM-USB-TEMP-AI Eight Channel Temperature/Voltage Input USB Data Acquisition Module,” Omega Engineering. [En línea]. Disponible en: <https://www.omega.com/en-us/communication-and-connectivity/data-acquisition-modules/om-usb-temp-series/p/OM-USB-TEMP>, [Accedido: 2026].

-
- [46] National Instruments, “cDAQ-9171 CompactDAQ Chassis,” National Instruments. [En línea]. Disponible en: <https://www.ni.com/en/shop/hardware/products/compactdaq-chassis/model-cdaq-9171.html>, [Accedido: 2026].
- [47] Stack Overflow, “Imagen de referencia técnica (recurso en línea),” Stack Overflow Community. [En línea]. Disponible en: <https://i.sstatic.net/1moBo.jpg>, [Accedido: 2025].
- [48] The KiCad Team, “Getting started in KiCad (v6.0, documentation),” [En línea]. Disponible en: https://docs.kicad.org/6.0/en/getting_started_in_kicad/getting_started_in_kicad.html, [Accedido: 2025].
- [49] KiCad.info Forum, “SOT-23 3D component offset from pads (solved),” [En línea]. Disponible en: <https://forum.kicad.info/t/sot-23-3d-component-offset-from-pads-solved/15252>, [Accedido: 2025].