 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01- 27

Sistema de comunicación embebido para pruebas experimentales de transferencia sincronizada de variables eléctricas

Aristides Jaramillo Henao


Ingeniería Mecatrónica

Diego Alejandro Herrera Jaramillo, MSc.

Jhon Jairo Rojas Montaña, MSc.

INSTITUTO TECNOLÓGICO METROPOLITANO

Marzo del 2026

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22


RESUMEN

Este trabajo presenta el diseño, implementación y validación de un sistema de comunicación embebido para transferencia sincronizada de variables eléctricas, verificando la integridad y estabilidad de los datos transmitidos mediante análisis experimental con un osciloscopio. El sistema está orientado a la adquisición y reconstrucción numérica de las tramas.

El objetivo principal es implementar y validar la comunicación maestro-esclavo entre diferentes microcontroladores como Arduino, ESP32 y dispositivos de Texas Instruments, verificando la integridad y estabilidad de los datos transmitidos mediante análisis experimental con un osciloscopio.


La metodología empleada se fundamenta en una arquitectura maestro-esclavo, donde el dispositivo maestro gestiona la señal de reloj, la selección de esclavos y la sincronización del intercambio de datos. Se definieron parámetros comunes para garantizar la operatividad entre microcontroladores, tales como una frecuencia de reloj, modo SPI y orden de bits MSB. La transferencia de información se realizó mediante tramas estructuradas de 13 bytes, compuestas por un valor booleano y tres variables en formato flotante IEEE 754.

La validación experimental incluyó configuraciones con uno y múltiples esclavos, evaluando las señales SPI mediante un osciloscopio y verificando la correspondencia entre las tramas binarias transmitidas y los valores numéricos reconstruidos en el dispositivo receptor. Los resultados evidencian una transmisión confiable y consistente de los datos, identificando además limitaciones en la velocidad de procesamiento en modo esclavo en algunos de los dispositivos.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Finalmente, se desarrollaron tableros de visualización con Streamlit y ThingsBoard, que permiten diferentes formas de uso del sistema según su método de comunicación, desde monitoreo local por puerto serial hasta supervisión remota mediante comunicación WiFi.

Palabras clave: SPI, sistemas embebidos, comunicación maestro-esclavo, microcontroladores, adquisición de datos.


	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

RECONOCIMIENTOS

Expreso mi más sincero agradecimiento a mi familia por su apoyo incondicional, comprensión y motivación constante a lo largo de mi formación profesional y durante el desarrollo de este trabajo de grado. Su respaldo ha sido fundamental para alcanzar esta meta.

Agradezco de manera especial a mis asesores, los profesores Diego Herrera y Jhon Rojas por su orientación, acompañamiento técnico y aportes académicos, los cuales fueron determinantes en la estructuración, ejecución y validación del presente proyecto.

De forma particular, expreso mi profundo agradecimiento al profesor Elkin por brindarme la oportunidad de participar en este proyecto, por su confianza y por el espacio académico que permitió el desarrollo de esta investigación.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

ACRÓNIMOS

ADC – Analog to Digital Converter.

BESS – Battery Energy Storage Systems.

DMA – Direct Memory Access.

IEEE – Institute of Electrical and Electronics Engineers.

IoT – Internet of Things.

MCU – Microcontrolador Unit.

MISO – Master In Slave Out.

MOSI – Master Out Slave In.

MSB – Most Significant Bit.

SCK – Serial Clock.

SPI – Serial Peripheral Interface.

TI – Texas Instruments.

UART – Universal Asynchronous Receiver-Transmitter.

USB – Universal Serial Bus.

WiFi – Wireless Fidelity.



 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

TABLA DE CONTENIDO

Contenido

Sistema de comunicación embebido para pruebas experimentales de transferencia sincronizada de.....	1
variables eléctricas.....	1
1. INTRODUCCIÓN	8
Objetivos	8
Objetivo general	8
Objetivos específicos.....	9
2. MARCO TEÓRICO.....	10
3. METODOLOGÍA.....	15
4. RESULTADOS Y DISCUSIÓN.....	19
Comunicación SPI Maestro–Esclavo	19
Comunicación SPI Maestro-Multi-Esclavo	27
Tableros de monitoreo de datos.....	36
5. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO	39
CONCLUSIONES	39
RECOMENDACIONES.....	39
TRABAJO FUTURO	40

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

REFERENCIAS 42

APÉNDICE 44

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

1. INTRODUCCIÓN


La transición energética en Colombia ha impulsado la incorporación de fuentes renovables y sistemas de almacenamiento con baterías, los cuales requieren mecanismos confiables de supervisión, control y gestión de información para su correcta operación en aplicaciones como microrredes, respaldo y autoconsumo (Ministerio de Minas y Energía, 2023; UPME, 2023). A nivel internacional, el almacenamiento energético es reconocido como un habilitador clave para la integración de fuentes renovables variables (International Energy Agency, 2022). En este contexto, la coordinación entre los controladores de carga y descarga y la trazabilidad de variables eléctricas críticas resultan fundamentales para garantizar la estabilidad y confiabilidad del sistema, tanto en escenarios reales como en entornos de laboratorio (Chen & Zhang, 2020; García et al., 2021).

Dicha coordinación depende de arquitecturas de comunicación embebidas que permitan la transferencia sincronizada de información con baja latencia y alta confiabilidad. El protocolo SPI se presenta como una alternativa adecuada para este propósito debido a su estructura determinista y su amplia adopción en sistemas embebidos (Analog Devices, 2020; Microchip Technology, 2021). Sin embargo, es necesario validar experimentalmente su desempeño, estandarizar la estructura de los datos transmitidos y documentar sus limitaciones operativas, con el fin de asegurar la integridad de la información y su utilidad en sistemas de supervisión y análisis.

Objetivos

Objetivo general


Desarrollar un sistema de comunicación entre dispositivos embebidos que permita la transferencia sincronizada de información de medidas de tensión o corriente mediante una metodología maestro–esclavo.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Objetivos específicos

- Identificar los requisitos técnicos de los controladores y del banco de pruebas, tales como variables, tasas de muestreo, eventos, alarmas y límites de seguridad para la transferencia de información.
- Seleccionar los sistemas embebidos compatibles con los controladores a implementar en la transferencia de información bajo la metodología maestro–esclavo.
- Validar la arquitectura de comunicación y el modelo de transferencia de datos.

En este trabajo se presenta una estructura de manera que, inicialmente, se establece contexto y la problemática que motivan el desarrollo del sistema de comunicación. Posteriormente, se profundiza en los fundamentos conceptuales del protocolo en sistemas embebidos, proporcionando la base teórica necesaria para el diseño. El trabajo presenta resultados de las pruebas experimentales y el análisis de señales capturadas por osciloscopio, evaluando el rendimiento real frente a las expectativas de diseño. Finalmente, se condensan las conclusiones del estudio y se proponen mejoras a futuro.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22


2. MARCO TEÓRICO

Los Sistemas de Almacenamiento de Energía en Baterías (BESS) son tecnologías clave dentro del proceso de transición energética, ya que permiten almacenar energía proveniente de fuentes renovables intermitentes y entregarla de manera controlada al sistema eléctrico (International Energy Agency, 2022). Un BESS está compuesto principalmente por módulos de baterías, sistemas de gestión de baterías (BMS), convertidores de potencia, sistemas de control y supervisión, e infraestructura de comunicación (Chen & Zhang, 2020).

La operación segura y eficiente de estos sistemas depende de la medición continua de variables críticas como tensión, corriente, temperatura y estado de carga. Estas variables deben ser adquiridas, procesadas y transmitidas con alta confiabilidad entre los diferentes subsistemas electrónicos (García et al., 2021). En este contexto, los sistemas embebidos desempeñan un papel fundamental al actuar como el núcleo de procesamiento encargado de la lógica de control, la adquisición de datos y la coordinación entre dispositivos (Chen & Zhang, 2020).

En consecuencia, garantizar la integridad y sincronización de los datos mediante arquitecturas de comunicación confiables es esencial. El diseño de instrumentación eléctrica moderna prioriza protocolos seriales para optimizar el hardware y reducir la complejidad del cableado, donde la elección entre comunicación síncrona o asíncrona constituye una decisión de diseño crítica en función de la latencia requerida, la velocidad de transmisión y la distancia entre nodos (Texas Instruments, 2019).

Dentro de los protocolos seriales síncronos, el protocolo SPI (Serial Peripheral Interface) se destaca por su simplicidad estructural, alta velocidad de transferencia y comportamiento, características especialmente relevantes en aplicaciones donde se requiere sincronización

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

precisa entre dispositivos embebidos. SPI opera bajo una arquitectura maestro–esclavo, en la cual el dispositivo maestro genera la señal de reloj y controla el intercambio de datos con uno o varios dispositivos periféricos (Analog Devices, 2020; Microchip Technology, 2021).

El protocolo SPI es un estándar de comunicación serial síncrona de alta velocidad que opera bajo una arquitectura maestro–esclavo (Analog Devices, 2020; Microchip Technology, 2021). Como se observa en la Figura 1, esta configuración emplea cuatro líneas principales de señal: SCK (reloj), MOSI (Master Out Slave In), MISO (Master In Slave Out) y SS/CS (selección de esclavo). El dispositivo maestro genera la señal de reloj y controla la activación del esclavo mediante la línea SS, garantizando que la transferencia de datos se realice de manera sincronizada.

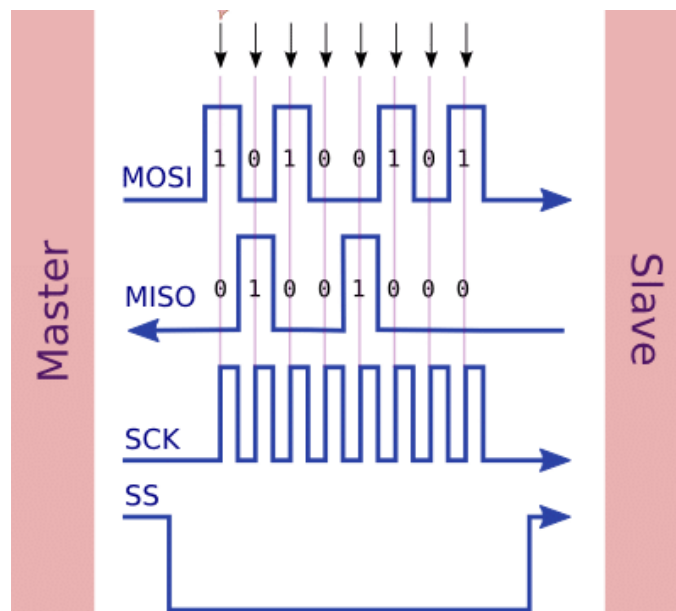



Figura 1. Esquema de funcionamiento del protocolo SPI.

Nota. Tomado de Comunicación SPI con PIC, por Control Automático Educación (s. f.).

SPI admite comunicación full-dúplex, lo que permite la transmisión y recepción simultánea de datos durante un mismo ciclo de reloj. Como se muestra en la Figura 2, la arquitectura SPI se basa en una configuración maestro–esclavo donde el dispositivo maestro genera la

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

señal de reloj (SCK) y controla la selección del esclavo mediante la línea CS o SS. Las líneas MOSI (Master Out Slave In) y MISO (Master In Slave Out) permiten el intercambio bidireccional de datos sincronizado por la señal de reloj, garantizando una comunicación determinista y de baja latencia (Texas Instruments, 2019).

En configuraciones con múltiples dispositivos, cada esclavo debe disponer de una línea de selección independiente para evitar conflictos eléctricos en la línea MISO. Cuando un esclavo no está seleccionado, su salida permanece en estado de alta impedancia, permitiendo que únicamente el dispositivo habilitado transmita información hacia el maestro. Esta organización mejora la integridad de los datos y asegura un flujo de comunicación confiable dentro de la plataforma experimental desarrollada (Texas Instruments, 2019).

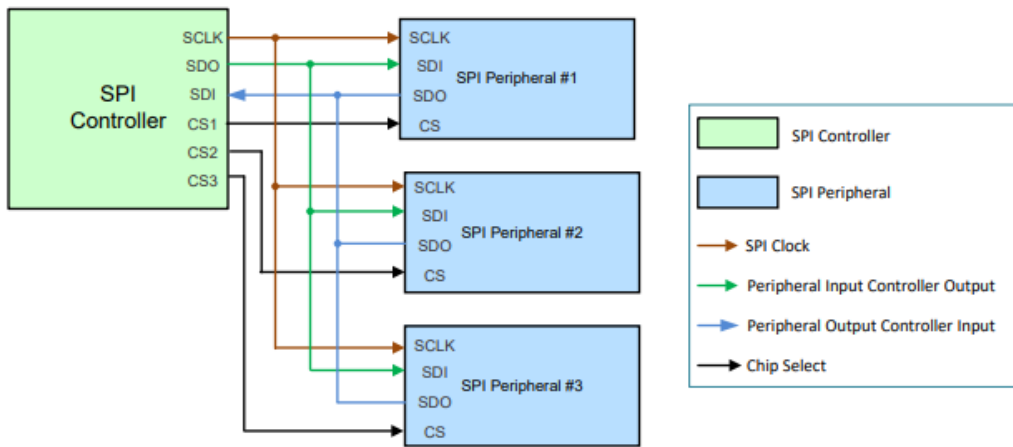



Figura 2. Configuración típica del bus SPI Multi slave.

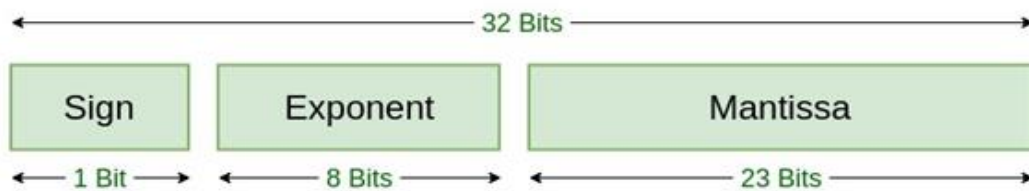
Nota. Tomado de *Understanding the SPI Bus (SBOA621)*, por Texas Instruments, 2019.

En comparación con protocolos como I²C y UART, SPI ofrece mayor velocidad de transmisión, menor latencia y mayor determinismo temporal, características relevantes para la transferencia sincronizada de variables eléctricas en plataformas experimentales aplicables a sistemas de almacenamiento energético. Su simplicidad estructural y amplio

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

soporte en microcontroladores comerciales facilitan su implementación en entornos de instrumentación electrónica (Analog Devices, 2020).

Para la transmisión de valores numéricos a través del protocolo SPI se emplea el estándar IEEE 754, el cual define la representación de números en punto flotante en diferentes niveles de precisión (IEEE, 2019). En su formato de precisión simple (single-precision floating-point), el número se representa mediante una estructura de 32 bits compuesta por un bit de signo, un exponente de 8 bits y una mantisa de 23 bits, como se muestra en la Figura 3.



Single Precision IEEE 754 Floating-Point Standard

Figura 3. Representación de los campos del número de punto flotante IEEE 754 de 32 bits (single-precision).


Nota. Tomado de IEEE Standard 754 Floating Point Numbers, por GeeksforGeeks (s. f.).

De acuerdo con el estándar IEEE 754-2019 (IEEE, 2019), en el formato de precisión simple la representación se organiza en tres campos:

- 1 bit de signo (S).
- 8 bits de exponente (E).
- 23 bits de mantisa (M).

El valor decimal correspondiente se obtiene mediante la siguiente expresión:

$$Valor = (-1)^S \times (1 + M) \times 2^{(E-127)}$$


	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

donde:

- S determina el signo del número.
- E es el exponente almacenado en forma sesgada (bias = 127).
- M representa la fracción binaria de la mantisa (interpretada como 0.M en base 2).

El estándar IEEE 754 es adoptado por defecto en los lenguajes de programación C y C++ para el tipo de dato float, lo que facilita la compatibilidad entre el procesamiento interno del microcontrolador y la transmisión de datos entre dispositivos, garantizando uniformidad en la reconstrucción de las variables eléctricas intercambiadas dentro del sistema.

La adecuada interpretación de los 32 bits según el estándar (IEEE, 2019) permite obtener nuevamente el valor decimal transmitido. Cuando no se respeta el orden de los bytes o la organización de los campos, pueden presentarse errores en el valor calculado.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22


3. METODOLOGÍA

Se desarrolló e implementó un sistema de comunicación SPI bajo arquitectura maestro–esclavo para evaluar la interoperabilidad entre microcontroladores de múltiples fabricantes. El estudio tuvo un enfoque experimental, basado en pruebas controladas de transmisión y validación de integridad de datos a nivel de trama y señal eléctrica. Las plataformas evaluadas fueron:

- Arduino UNO.
- ESP32-S3.
- LaunchXL F280049C de Texas Instruments.

El protocolo SPI se implementó en configuración full-dúplex, bajo topología maestro–esclavo y maestro–múltiples esclavos. Las líneas MOSI, MISO y SCK fueron compartidas entre los dispositivos, mientras que cada esclavo contó con una línea **CS** independiente, con el fin de evitar contención en el bus de datos.

En las interfaces entre dispositivos que operan a 5 V y 3.3 V se utilizó un convertidor de nivel bidireccional Level Converter MH (Figura 4), garantizando la compatibilidad eléctrica y la protección de los pines de entrada y salida.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

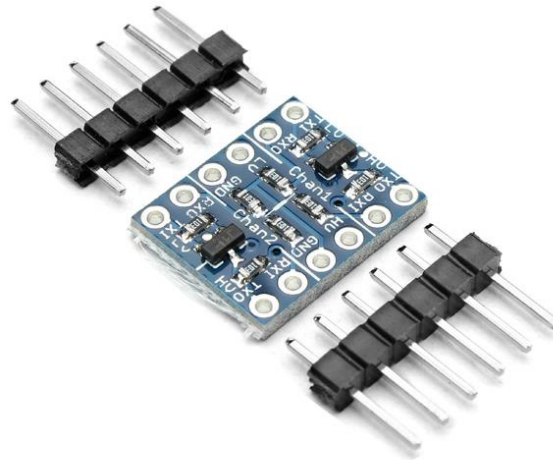


Figura 4. Convertidor de niveles bidireccional.

Nota. Tomado de *Convertor niveles 2 canales bidireccional 5V-3.3V*, por Electronilab (s. f.).

Para asegurar compatibilidad entre las diferentes plataformas, se definieron parámetros de operación comunes. Como se muestra en la Figura 5, correspondiente al diagrama de temporización del protocolo SPI configurado en Modo 0, el reloj presenta polaridad CPOL = 0 (estado de reposo en nivel bajo) y fase CPHA = 0, por lo que el muestreo de los datos ocurre en el flanco ascendente de la señal CLK.

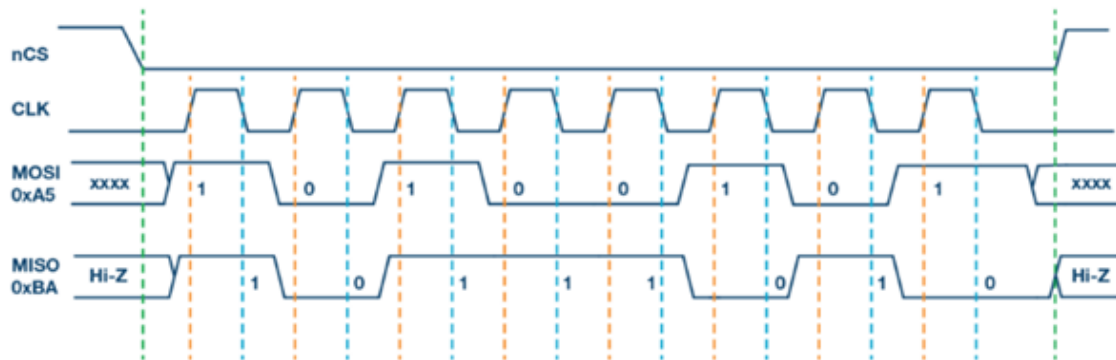



Figura 5. Diagrama del modo 0 del protocolo SPI.

Nota. Tomado de *Qué son los modos del SPI*, por SoC-eame, 2024.


	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

La transmisión se configuró en formato MSB first, enviando primero el bit más significativo. Se evaluó la comunicación a diferentes frecuencias de operación. En el caso del Arduino UNO se realizaron pruebas en el rango de 250 kHz a 1 MHz, mientras que en las plataformas basadas en ESP32 y en el dispositivo de Texas Instruments se efectuaron ensayos entre 1 MHz y 10 MHz, con el propósito de analizar la estabilidad y confiabilidad de la transferencia de datos.

Se diseñó una trama de 13 bytes compuesta por: 1 variable booleana, 3 variables tipo float (32 bits, formato IEEE 754 single precision). En el caso del LaunchXL F280049C de Texas Instruments, la transmisión se realizó en bloques de 16 bits, enviando primero los bits más significativos.

La validación del sistema se desarrolló mediante un procedimiento experimental estructurado que permitió verificar tanto la integridad de los datos transmitidos como la estabilidad de la comunicación entre las diferentes plataformas. Este proceso incluyó la captura y análisis de las señales SPI mediante osciloscopio digital, la verificación del tamaño de la trama transmitida y la comparación de los datos recibidos con su representación binaria teórica bajo el estándar IEEE 754.

Adicionalmente, se incorporó un proceso iterativo de verificación, en el cual, en caso de detectarse discrepancias entre los valores transmitidos y los esperados, el procedimiento de captura y análisis se repetía hasta obtener coincidencia exacta en los datos. Una vez validada la integridad de la transmisión, se procedió a evaluar la estabilidad del sistema a diferentes frecuencias de operación. El flujo general del procedimiento de validación implementado se presenta en la Figura 6.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

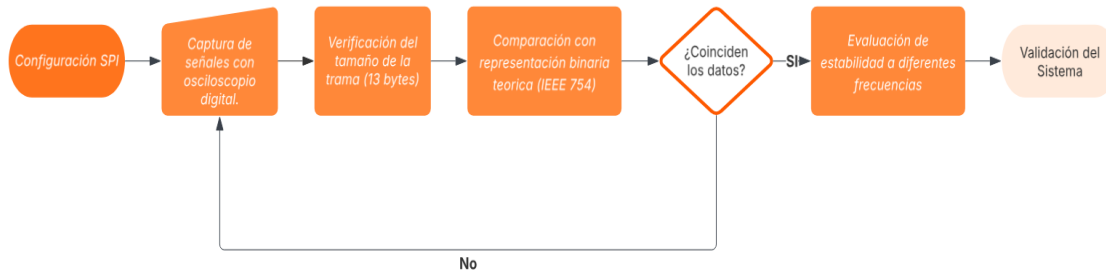



Figura 6. Diagrama de flujo del proceso de validación experimental del sistema SPI.

En configuraciones con el F280049C se verificó la transmisión en 13 bytes en separaciones de 16 bits y la correcta recepción mediante DMA en el ESP32-S3. Finalmente, la comunicación se consideró válida únicamente cuando se obtuvo coincidencia exacta entre los datos recibidos y los valores esperados.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

4. RESULTADOS Y DISCUSIÓN


Comunicación SPI Maestro–Esclavo

En esta sección se presentan los resultados obtenidos en la implementación del protocolo SPI bajo una arquitectura maestro–esclavo, evaluando 3 diferentes combinaciones de hardware. Con los dispositivos mencionados en la sección anterior.

Se implementó una comunicación SPI en la que el Arduino Uno operó como dispositivo maestro y la ESP32-S3 como esclavo, tal como se muestra en la Figura 7. Debido a la diferencia en los niveles lógicos de operación entre ambos dispositivos (5 V en el Arduino y 3.3 V en la ESP32-S3), fue necesario incorporar un convertidor de niveles bidireccional para garantizar la compatibilidad eléctrica y evitar posibles daños en el dispositivo que opera a 3.3 V. La conexión empleando este convertidor de niveles se presenta en la Figura 8.



Figura 7. Diagrama SPI Arduino Uno - ESP32-S3.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

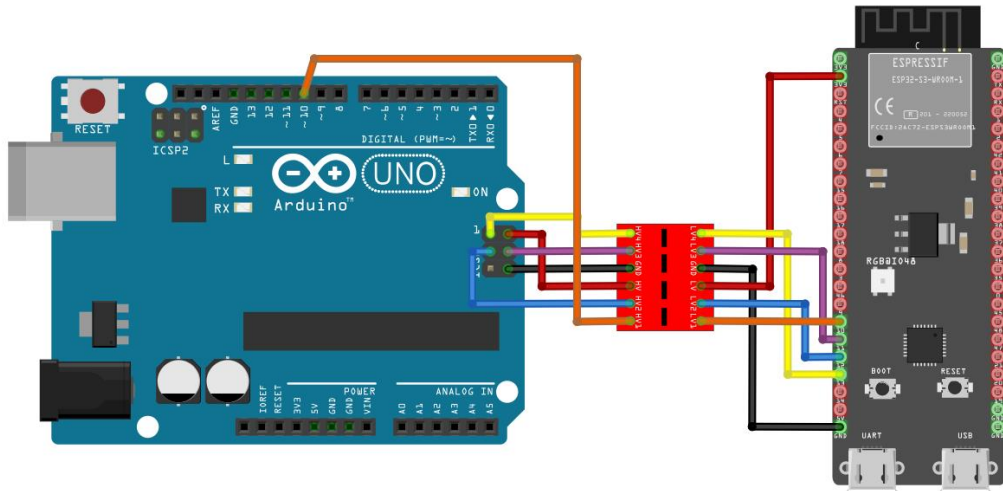


Figura 8. Esquema de conexión SPI Arduino Uno - ESP32-S3.


Los parámetros de transmisión fueron: frecuencia 1 MHz, orden de bits: MSB first, modo SPI: Modo 0 y tamaño de trama: 13 bytes, transmitidos en bloques de 8 bits por cada byte.

Para verificar la correcta transmisión, se utilizaron los valores de prueba mostrados en la Tabla 1.

Tabla 1. Valores de prueba utilizados para la validación de la trama SPI.

Variable	Valor decimal	Binario
swichtState	1.00	10000000
voltage	1.00	0 01111111 000000000000000000000000
current	9.50	0 1000010 001100000000000000000000
temperature	12.54	0 1000010 10010001000111101011100

La captura realizada con el osciloscopio OWON permitió visualizar las señales correspondientes a la transmisión SPI, evidenciando la coherencia entre las formas de onda registradas y los valores presentados en la Tabla 1. En la Figura 9 se muestran las formas de

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

onda obtenidas, donde la señal amarilla corresponde a la línea *MISO*, utilizada por el esclavo para enviar los datos hacia el maestro, mientras que la señal azul representa la línea *SCK*, correspondiente al reloj generado por el maestro y encargado de sincronizar la transmisión de cada bit.

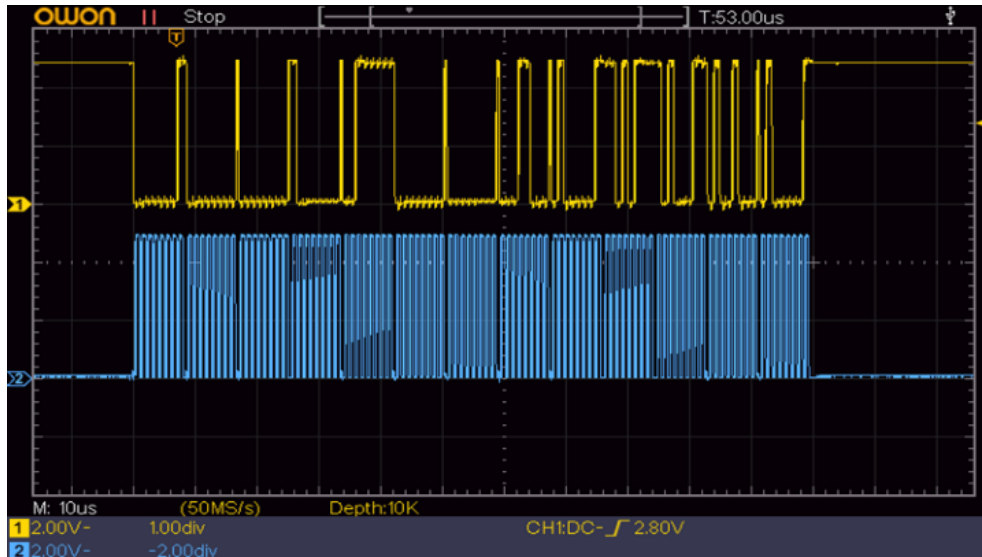



Figura 9. Señales *MOSI* y *SCK* en la comunicación *SPI* *Arduino UNO – ESP32-S3*.

El análisis detallado de la forma de onda permitió comparar cada byte transmitido con la representación binaria previamente definida en la Tabla 1 de valores de prueba. Se comprobó que los datos observados en la señal coincidían exactamente con los valores esperados, confirmando la integridad y consistencia de la comunicación *SPI* a una frecuencia de 1 MHz. Adicionalmente, se verificó que los datos fueron muestreados en el flanco ascendente del reloj, coherente con la configuración del Modo 0, lo que confirma la correcta sincronización temporal entre maestro y esclavo.

Se implementó una segunda comunicación *SPI* en la que un módulo *ESP32-S3* actuó como maestro y otro *ESP32-S3* como esclavo, tal como se muestra en la Figura 10. A diferencia de la implementación realizada entre el *Arduino Uno* y la *ESP32-S3*, en esta configuración no fue necesario emplear convertidores de nivel lógico, debido a que ambos dispositivos operan con niveles de 3.3 V. Esto simplifica el diseño físico del sistema y reduce posibles

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

fuentes de interferencia o ruido. La conexión utilizada para esta configuración se presenta en la Figura 11.



Figura 10. Diagrama SPI ESP32-S3 - ESP32-S3.

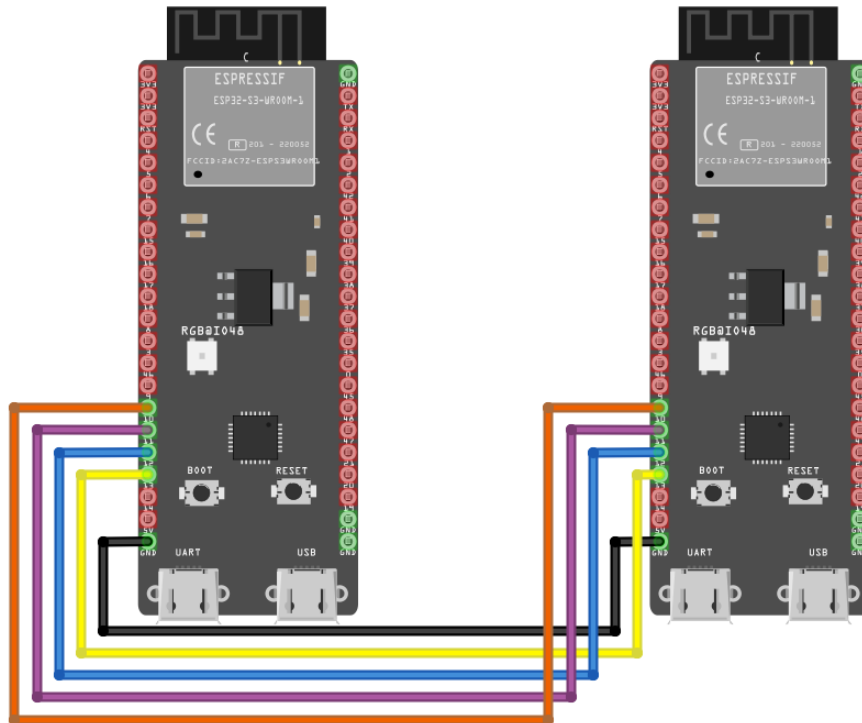



Figura 11. Esquema de conexión SPI ESP32-S3 - ESP32-S3.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22


Los parámetros de transmisión fueron: frecuencia 1 MHz, orden de bits: MSB first, modo SPI: Modo 0 y tamaño de trama: 13 bytes, transmitidos en bloques de 8 bits por cada byte.

Para verificar la correcta transmisión, se utilizaron los valores de prueba mostrados en la Tabla 2.

Tabla 2. Valores de prueba utilizados para la validación de la trama SPI.

Variable	Valor decimal	Binario
switchState	1.00	10000000
voltage	1.00	0 01111111 000000000000000000000000
current	9.50	0 10000010 001100000000000000000000
temperature	12.54	0 10000010 10010001000111101011100

La captura realizada con el osciloscopio OWON permitió visualizar las señales correspondientes a la transmisión SPI, evidenciando la coherencia entre las formas de onda registradas y los valores presentados en la Tabla 2. En la Figura 12 se muestran las formas de onda obtenidas, donde la señal amarilla corresponde a la línea *MISO*, utilizada por el esclavo para enviar los datos hacia el maestro, mientras que la señal azul representa la línea *SCK*, correspondiente al reloj generado por el maestro y encargado de sincronizar la transmisión de cada bit.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

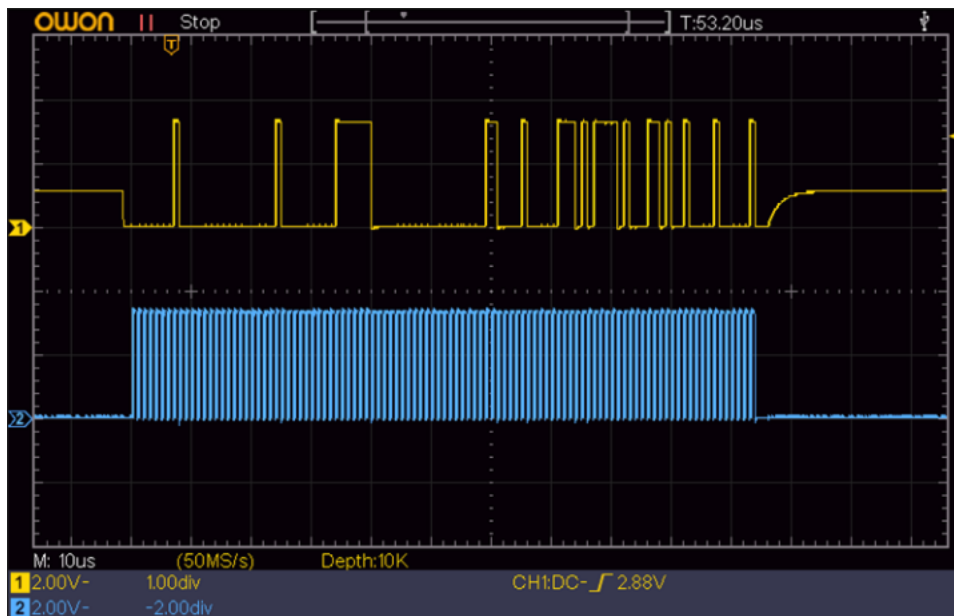



Figura 12. Señales MOSI y SCK en la comunicación ESP32-S3 – ESP32-S3.

El análisis de la forma de onda permitió comparar cada byte transmitido con la representación binaria previamente definida en la tabla de valores de prueba. Se comprobó que los datos observados coincidían exactamente con los valores esperados, garantizando la integridad y consistencia de la comunicación SPI a una frecuencia de 1 MHz. Además, debido al uso del controlador DMA en la ESP32-S3, se observó la transmisión continua de los 13 bytes de la trama, sin segmentación en bloques independientes.

En comparación con la implementación Arduino–ESP32, esta configuración presentó una señal más limpia, atribuida a la ausencia de conversión de niveles lógicos y a la mayor capacidad de procesamiento de ambos dispositivos. Esto evidencia que la compatibilidad eléctrica directa reduce distorsiones y mejora la estabilidad en la transmisión a altas frecuencias.

Se implementó una última comunicación SPI bajo una arquitectura maestro–esclavo, en la cual el módulo ESP32-S3 actuó como maestro y la tarjeta de desarrollo LaunchXL-F280049C como esclavo, tal como se muestra en la Figura 13. A diferencia de la implementación realizada entre el Arduino Uno y la ESP32-S3, en esta configuración no fue necesario

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

emplear convertidores de nivel lógico, ya que ambos dispositivos operan con niveles de 3.3 V. Esto simplifica el diseño físico del sistema y reduce posibles fuentes de ruido o interferencia. La conexión utilizada para esta configuración se presenta en la Figura 14.



Figura 13. Diagrama SPI ESP32-S3 - LaunchXL F280049C.

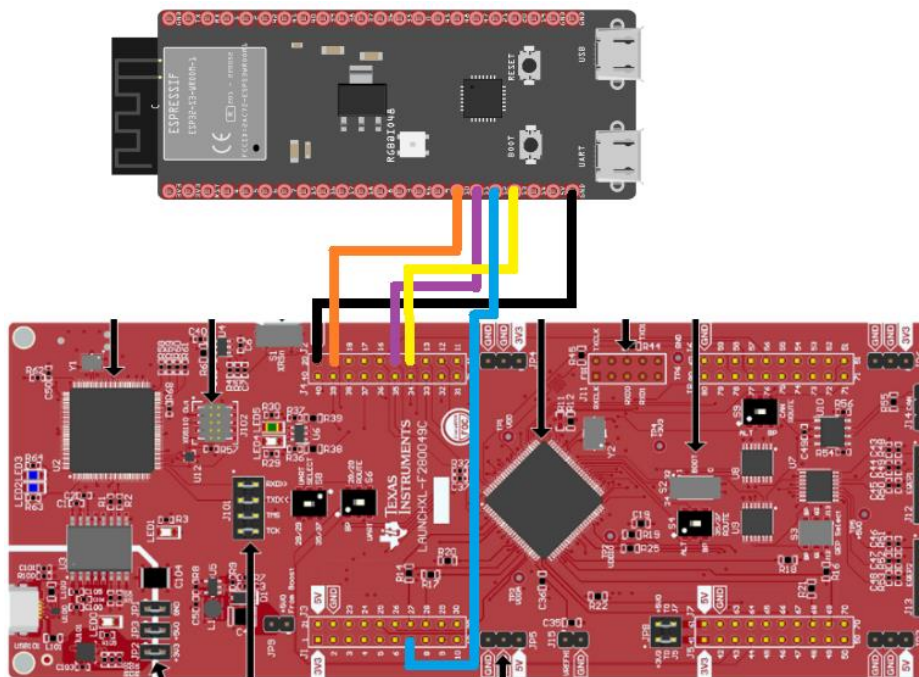


Figura 14. Esquema de conexión SPI LaunchXL F280049C - ESP32-S3

Los parámetros de transmisión fueron: frecuencia 1 MHz, orden de bits: MSB first, modo SPI: Modo 0 y tamaño de trama: 13 bytes, transmitidos en palabras de 16 bits por ciclo de transferencia. Para verificar la correcta transmisión, se utilizaron los valores de prueba mostrados en la Tabla 3.


 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Tabla 3. Valores de prueba utilizados para la validación de la trama SPI.

Variable	Valor decimal	Binario
switchState	1.00	10000000
voltage	2.00	0 01111111 000000000000000000000000
current	8.50	0 1000010 000100000000000000000000
temperature	12.54	0 1000010 10010001000111101011100

La captura realizada con el osciloscopio OWON permitió visualizar las señales correspondientes a la transmisión SPI, evidenciando la coherencia entre las formas de onda registradas y los valores presentados en la Tabla 3. En la Figura 15 se muestran las formas de onda obtenidas, donde la señal amarilla corresponde a la línea MISO, utilizada por el esclavo para enviar los datos hacia el maestro, mientras que la señal azul representa la línea SCK, correspondiente al reloj generado por el maestro y encargado de sincronizar la transmisión de cada bit.

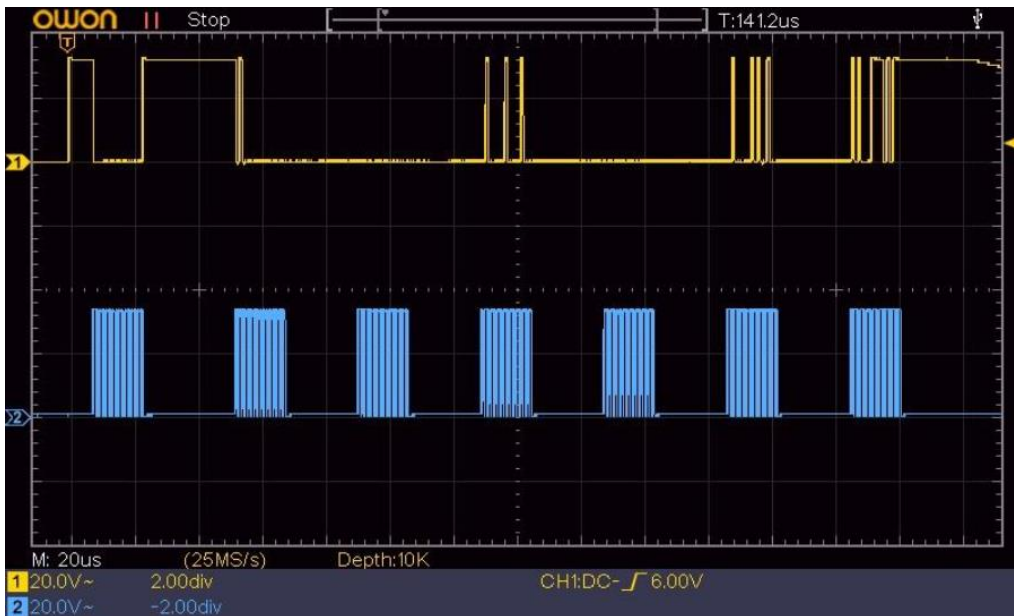



Figura 15. Señales MOSI y SCK en la comunicación ESP32-S3 – LaunchXL F280049C.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

En la captura se observa que la transmisión se realiza en bloques de 16 bits. Asimismo, se aprecia que el tiempo total de envío de la trama es menor en comparación con las configuraciones anteriores, lo cual indica una mayor eficiencia en la transferencia de datos.

Comunicación SPI Maestro-Multi-Esclavo

Se evaluó una arquitectura SPI multi-esclavo en la cual un único maestro controla múltiples dispositivos mediante líneas de selección (CS) independientes, asegurando que solo un esclavo habilite la línea MISO en cada instante y evitando conflictos en el bus. Con los mismos dispositivos utilizados en la configuración maestro-esclavo se implementaron dos variantes experimentales, permitiendo que actuaran tanto como maestro como esclavo; en el caso del dispositivo de Texas Instruments, este se configuró únicamente como esclavo debido a los lineamientos establecidos en el proyecto TULATO.

Primero se implementó una arquitectura en la que un Arduino Uno actuó como maestro, comunicándose con dos dispositivos esclavos: otro Arduino Uno y un módulo ESP32-S3. Cada esclavo fue controlado mediante una línea de selección dedicada (SS1 y SS2), lo que permitió su activación individual dentro del bus SPI compartido, tal como se muestra en la Figura 16. La conexión empleada para esta configuración se presenta en la Figura 17.

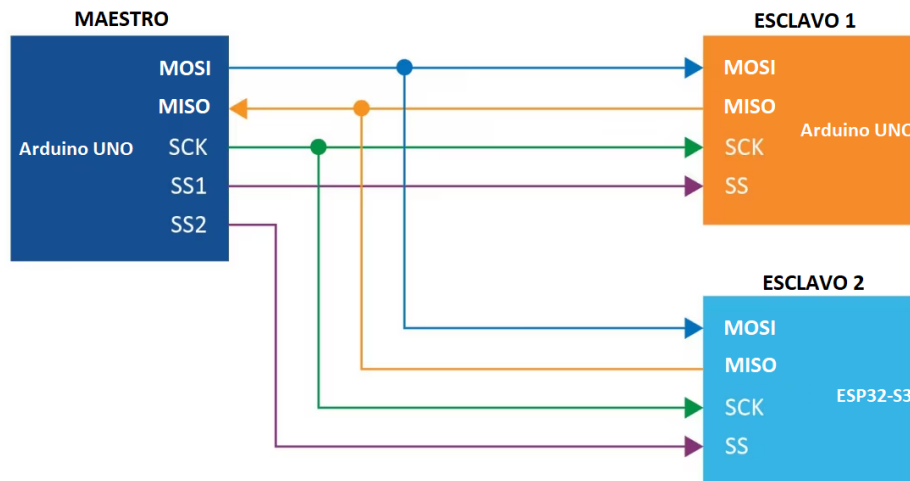



Figura 16. Diagrama SPI Arduino Uno - Arduino UNO - ESP32-S3.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

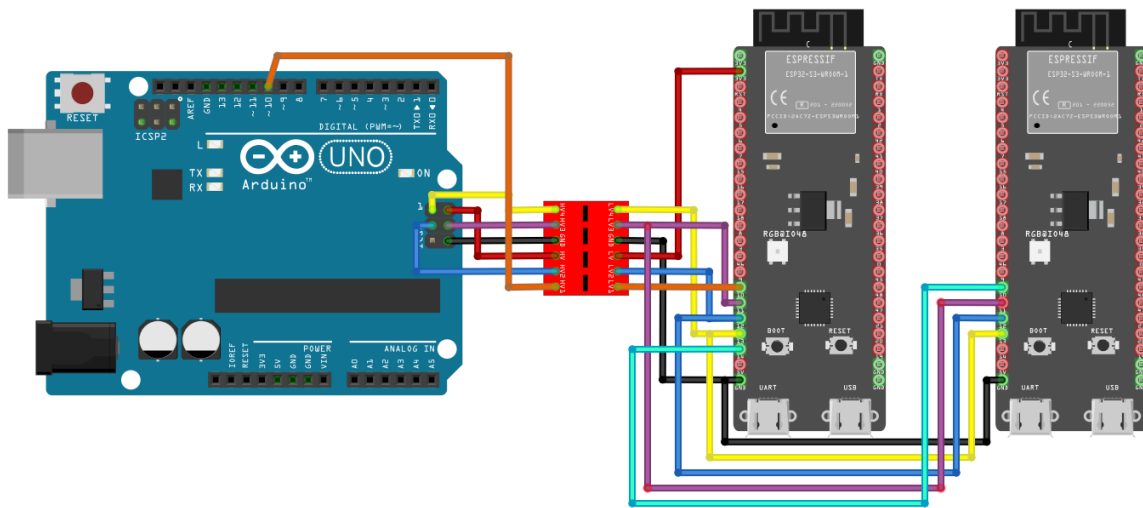


Figura 17. Esquema de conexión SPI Arduino Uno - ESP32-S3 - ESP32-S3.

Para el montaje físico fue necesario utilizar un convertidor de nivel lógico bidireccional, debido a la diferencia de niveles de tensión entre los dispositivos (5 V en el Arduino UNO y 3.3 V en la ESP32-S3). Esto garantizó la protección de los pines de la ESP32-S3 y la compatibilidad eléctrica del sistema.

Los parámetros de transmisión configurados fueron los siguientes: frecuencia de reloj de 250 kHz y 1 MHz en el Arduino UNO (según la prueba realizada), y 1 MHz en la ESP32-S3; orden de bits MSB first; modo SPI 0; y un tamaño de trama de 13 bytes. La transmisión se efectuó en bloques de 8 bits por cada byte en el caso del Arduino UNO, mientras que en la ESP32-S3 se empleó transferencia mediante DMA, optimizando la recepción de los datos y reduciendo la intervención directa del procesador durante el proceso de comunicación. Se utilizaron valores de prueba distintos para cada esclavo, con el fin de diferenciar claramente las tramas transmitidas.


	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Tabla 4. Paquete de datos SPI del Arduino UNO como esclavo.


Variable	Valor decimal	Binario
switchState	0.00	00000000
voltage	2.00	0 10000000 000000000000000000000000
current	6.50	0 10000001 101000000000000000000000
temperature	17.52	0 10000011 00011100011010101110000

Tabla 5. Paquete de datos SPI del ESP32-S3 como esclavo.

Variable	Valor decimal	Binario
switchState	1.00	10000000
voltage	1.00	0 01111111 000000000000000000000000
current	9.50	0 10000010 001100000000000000000000
temperature	12.54	0 10000010 100100010001111101011100

Debido a la presencia del convertidor bidireccional de nivel lógico, fue necesario realizar mediciones independientes en cada extremo del bus: una desde los pines del Arduino UNO esclavo (5 V) y otra desde el lado de la ESP32-S3 (3.3 V). Esto permitió garantizar una interpretación adecuada de las señales y evitar posibles interferencias o daños derivados de una medición incorrecta en el punto equivocado.

La captura realizada con el osciloscopio OWON permitió analizar la transmisión SPI, evidenciando coherencia entre las formas de onda registradas y los valores reportados en las Tablas 4 (derecha) y 5 (izquierda). En las mediciones, la señal amarilla corresponde a la línea MISO, mientras que la azul representa la señal de reloj SCK. En la configuración operando a 250 kHz presentados en la Figura 18, se observó una transmisión estable de las tramas de 13 bytes, sin pérdida de datos y con adecuada sincronización entre reloj y datos.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

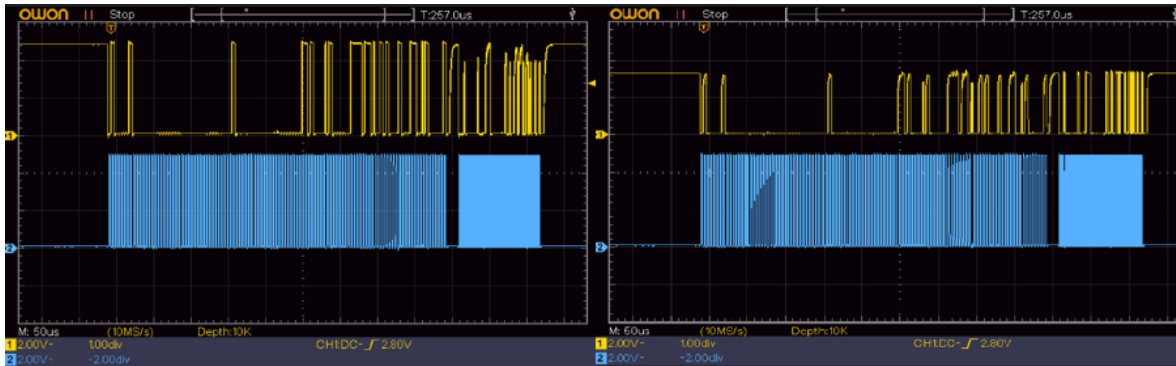


Figura 18. Señales MOSI y SCK en la comunicación Arduino UNO - Arduino UNO - ESP32-S3 250 kHz.

Al incrementar la frecuencia a 1 MHz Figura 19, cuyos valores se presentan en las Tablas 4 (izquierda) y 5 (derecha), se evidenció que el Arduino UNO operando en modo esclavo presentaba limitaciones de procesamiento que impedían una transmisión continua sin interrupciones. Por ello, fue necesario introducir un retardo mínimo de 4 μ s entre la transmisión de cada byte. Una vez implementado este ajuste, la comunicación se estabilizó, confirmando que la limitación observada estaba asociada a la capacidad de respuesta del microcontrolador en modo esclavo y no a deficiencias del protocolo SPI.

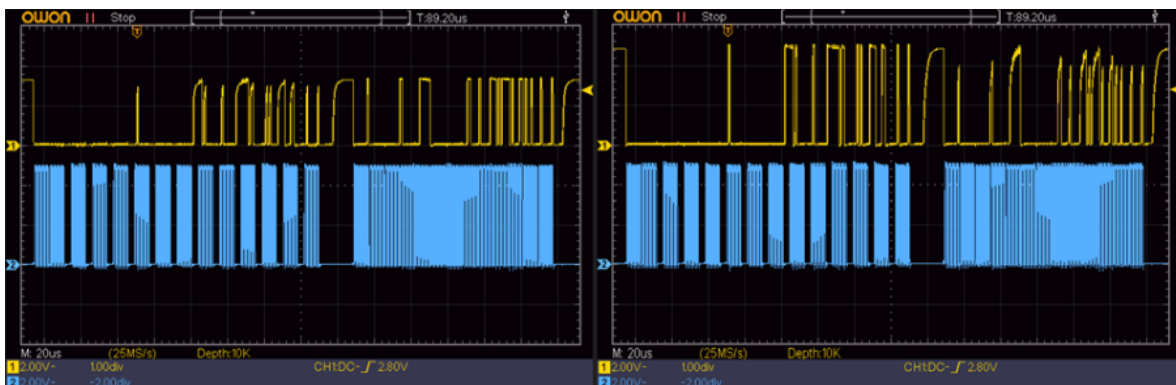



Figura 19. Señales MOSI y SCK en la comunicación Arduino UNO - Arduino UNO - ESP32-S3 1 MHz.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Por último, se implementó una arquitectura SPI maestro–multi-esclavo en la que una ESP32-S3 actuó como dispositivo maestro, comunicándose con dos esclavos: una LaunchXL-F280049C de Texas Instruments y una segunda ESP32-S3. Cada dispositivo esclavo fue gestionado mediante una línea de selección independiente (SS1 y SS2), lo que permitió su habilitación individual dentro del bus SPI compartido y evitó conflictos en la línea MISO, tal como se muestra en la Figura 20. La conexión utilizada para esta configuración se presenta en la Figura 21.

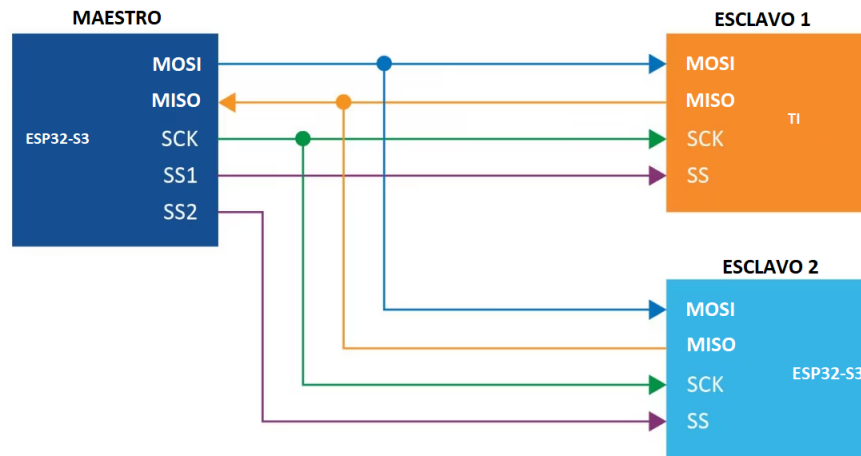



Figura 20. Diagrama SPI ESP32-S3 - LaunchXL F280049C - ESP32-S3.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

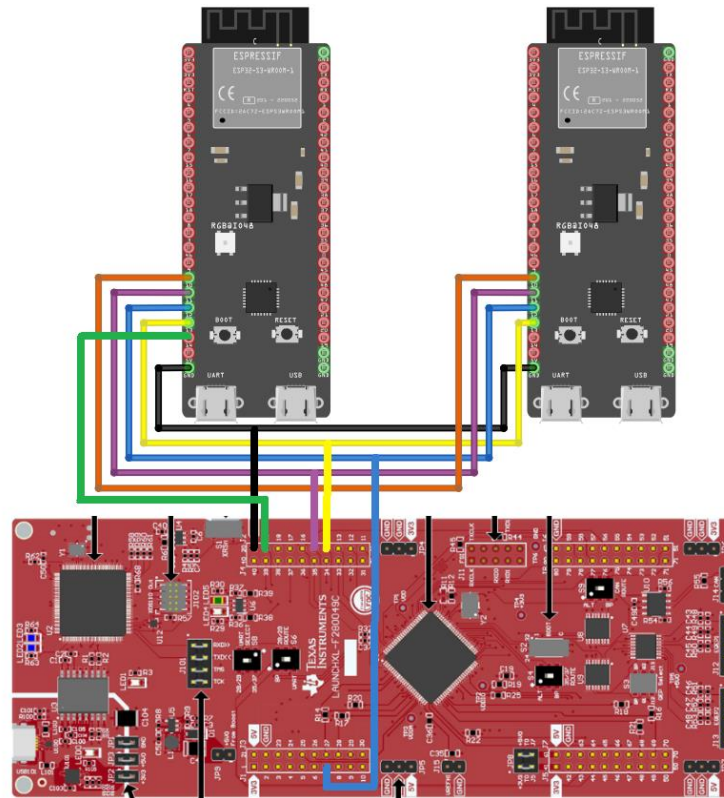


Figura 21. Esquema de conexión SPI ESP32-S3 – LaunchXL F280049C - ESP32-S3.

Los parámetros de transmisión configurados fueron los siguientes: frecuencia de reloj de 1 MHz, orden de bits MSB first y modo SPI 0 para ambos esclavos. El tamaño de la trama fue de 13 bytes. En el caso de la ESP32-S3 esclavo, la recepción se realizó mediante el uso de DMA, optimizando la transferencia de datos y reduciendo la carga del procesador. Para el LaunchXL F280049C, la transmisión se organizó en bloques de 16 bits, de acuerdo con su configuración interna, enviando una palabra de 16 bits por cada ciclo de transferencia iniciado por el maestro.

Se utilizaron valores de prueba distintos para cada esclavo, con el fin de diferenciar claramente las tramas transmitidas.


 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Tabla 6. Paquete de datos SPI del LaunchXL F280049C como esclavo.


Variable	Valor decimal	Binario
switchState	1.00	10000000
voltage	2.00	0 01111111 000000000000000000000000
current	8.50	0 10000010 000100000000000000000000
temperature	12.54	0 10000010 10010001000111101011100

Tabla 7. Paquete de datos SPI del ESP32-S3 como esclavo.

Variable	Valor decimal	Binario
switchState	0.00	00000000
voltage	3.00	0 10000000 100000000000000000000000
current	10.00	0 10000010 010000000000000000000000
temperature	15.25	0 10000010 111010000000000000000000

La captura realizada con el osciloscopio OWON permitió analizar la transmisión SPI, evidenciando coherencia entre las formas de onda registradas y los valores reportados en las Tablas 6, se observa la señal amarilla correspondiente a la línea MISO y la señal azul correspondiente a la línea SCK, generada por el maestro.

En la Figura 22, la señal SCK (azul) presenta un tren de pulsos uniforme y continuo, característico de una frecuencia de 1 MHz, evidenciando una transmisión de alta velocidad. Como se mencionó anteriormente, la trama de 13 bytes se transmitió en bloques de 16 bits, donde los primeros 8 bits corresponden a los bits más significativos (MSB) de cada bloque.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

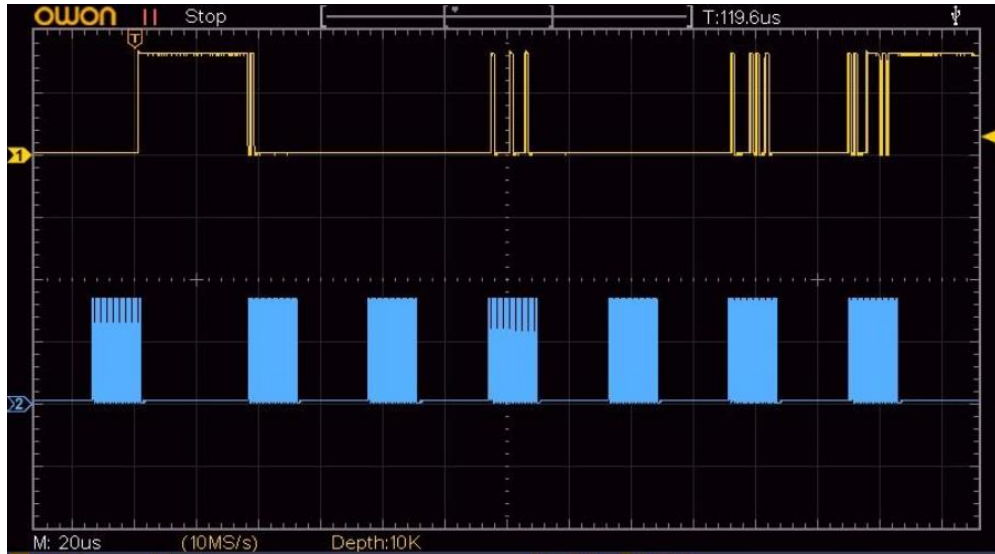


Figura 22. Señales MOSI y SCK en la comunicación LaunchXL F280049C 1 MHz.

En la Figura 23, la señal SCK (azul) presenta un tren de pulsos uniforme y continuo, característico de una frecuencia de 1 MHz, evidenciando una transmisión de más lenta en comparación a la TI. En este caso, la transferencia de datos se realiza mediante el uso de DMA en la ESP32-S3, lo que permite que los 13 bytes del paquete sean transmitidos en un solo bloque y no separados en 16 bits como en la TI.

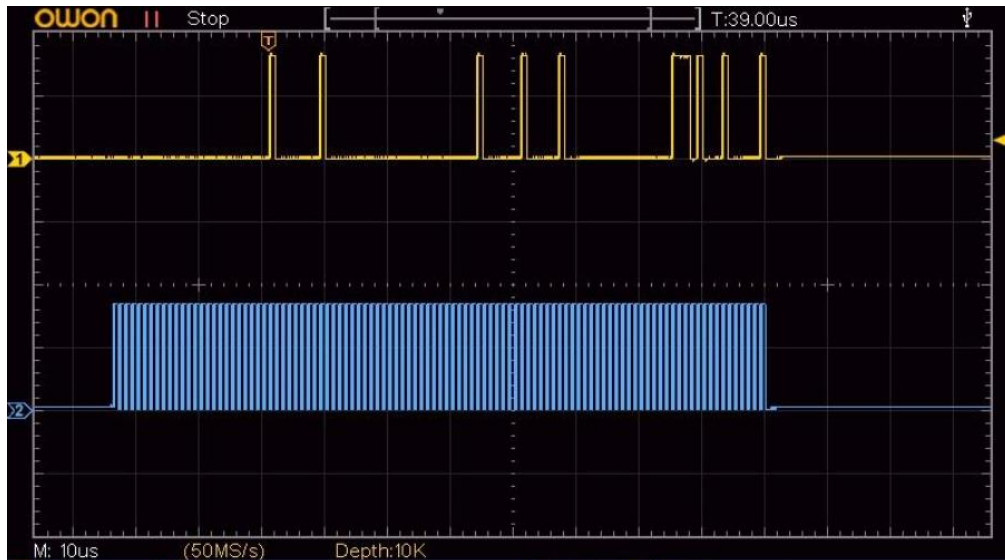



Figura 23. Señales MOSI y SCK en la comunicación ESP32-S3 1 MHz.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

Otra medición relevante corresponde a la distorsión observada en la tarjeta de TI al operar a 10 MHz, como se muestra en la Figura 24. Aunque la señal cuadrada de reloj (azul) presenta una ligera deformación con tendencia senoidal, esto no afectó la correcta lectura de los datos, los cuales coincidieron con los valores reportados en la Tabla 6. No obstante, no se descarta que esta distorsión pueda generar efectos asociados al ruido en escenarios posteriores, por lo que no es posible emitir aún una conclusión definitiva al respecto.

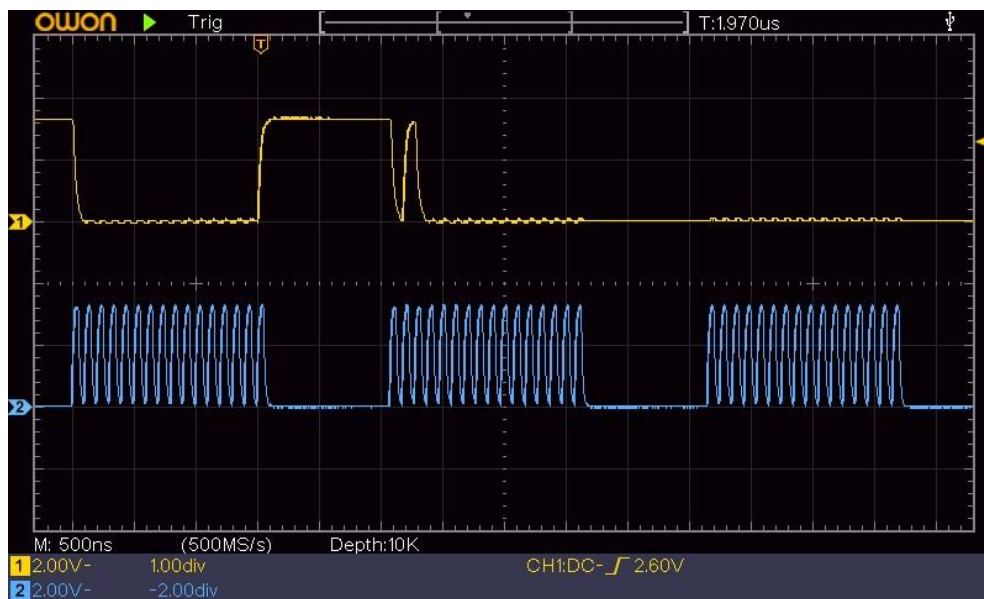



Figura 24. Señal SPI Texas Instruments a 10 MHz con zoom.

Durante la medición se observó que las tramas correspondientes a cada esclavo no aparecen simultáneamente en la línea MISO, sino de forma alternada. Esto se debe a que en una arquitectura SPI multi-esclavo solo el dispositivo cuyo SS se encuentra activo habilita su salida MISO, mientras los demás permanecen en estado de alta impedancia. Por lo tanto, la línea compartida refleja únicamente la transmisión del esclavo seleccionado en cada instante, evitando conflictos eléctricos en el bus.

En términos generales, la configuración con dispositivos que operan al mismo nivel lógico presentó menor distorsión y mayor estabilidad a 1 MHz. Por otro lado, el Arduino UNO

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

evidenció limitaciones de desempeño en modo esclavo a frecuencias altas, lo que obligó a introducir retardos entre bytes.

Tableros de monitoreo de datos.

Para la visualización de los datos se implementaron dos propuestas de tablero de monitoreo: una mediante comunicación por puerto serial y otra a través de conexión WiFi.

En la primera propuesta se desarrolló un tablero de monitoreo utilizando Streamlit, programado en Python, el cual permite el monitoreo en tiempo real de los datos transmitidos por comunicación serial. Este tablero permitió visualizar continuamente las variables recibidas desde el F280049C y la ESP32-S3, incluyendo: voltaje, corriente, temperatura y Estado del interruptor (switchState). Como se muestra en la Figura 25.

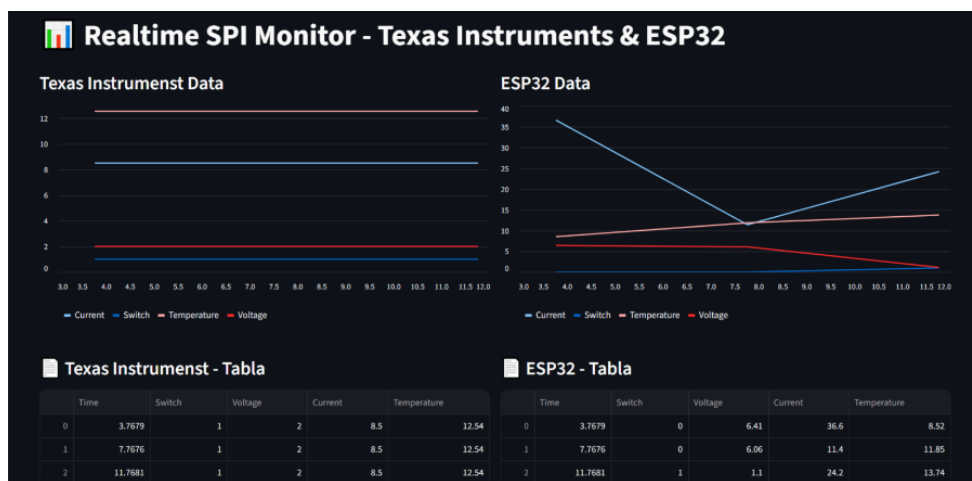



Figura 25. Tablero de visualización en Streamlit.

El segundo tablero de visualización fue desarrollado en ThingBoard, desplegado en un contenedor Docker, una plataforma que permite ejecutar aplicaciones de forma aislada y sencilla. En esta arquitectura, la ESP32-S3 transmitió los datos mediante conexión WiFi a través del protocolo MQTT, lo que permitió la visualización inalámbrica de la información.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

El tablero de ThingBoard permitió monitorear en tiempo real las variables transmitidas por el sistema, entre las cuales se incluyen voltaje, corriente, temperatura y el estado del interruptor (switchState). Como se muestra en la Figura 26.

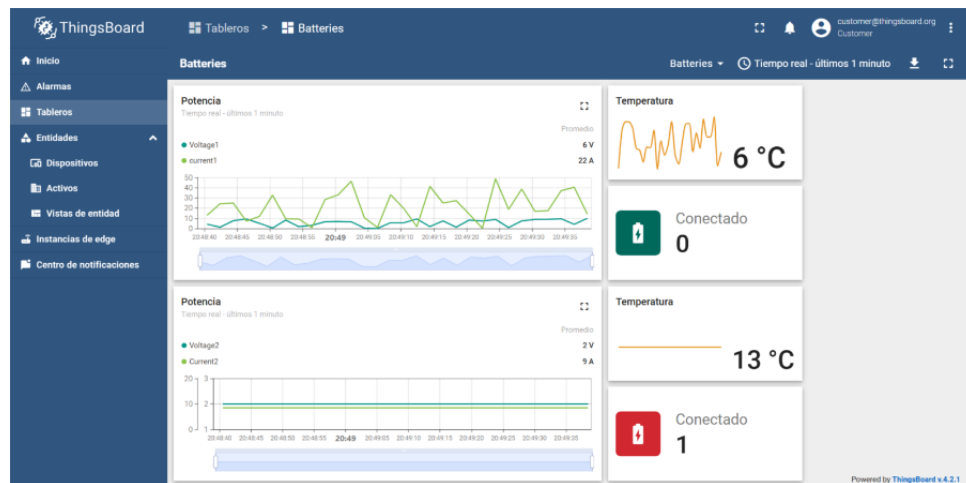



Figura 26. Tablero de visualización en ThingBoard.

A partir de las pruebas realizadas, se optó por implementar la plataforma ThingsBoard como solución principal de monitoreo. Esta decisión se tomó debido a que la aplicación está orientada al monitoreo de variables asociadas a sistemas de baterías, donde la reducción de cableado mejora la instalación, mantenimiento y escalabilidad del sistema.

La configuración final seleccionada fue la arquitectura ESP32-S3 – LaunchXL F280049C (Texas Instruments) – ESP32-S3, donde la ESP32-S3 actúa como maestro del bus SPI y como nodo de comunicación inalámbrica. Esta elección se fundamentó en que la ESP32-S3 integra conectividad WiFi, lo que permite transmitir la telemetría directamente hacia la plataforma ThingsBoard sin requerir interfaces cableadas adicionales.

Por su parte, la tarjeta LaunchXL F280049C fue incorporada como uno de los dispositivos esclavos debido a que corresponde a un equipo previamente implementado dentro del proyecto TULATO, lo que garantiza compatibilidad con la arquitectura existente.

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

La conexión utilizada para el montaje final se presenta en la Figura 27, donde se observa la configuración previamente descrita entre los dispositivos ESP32-S3 – LaunchXL-F280049C – ESP32-S3.

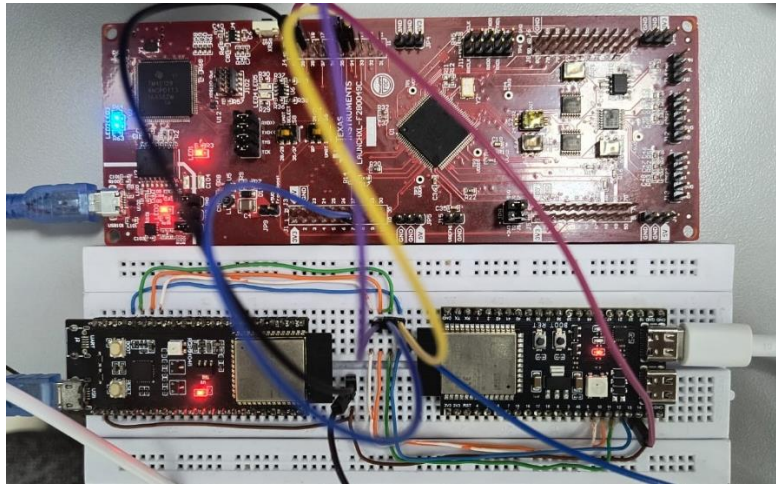


Figura 27. Conexión física del sistema final.

En la Figura 28 se muestra el montaje final del sistema en operación, donde se evidencia la interconexión de los dispositivos a través del bus SPI, la transmisión de datos vía WiFi hacia la plataforma de monitoreo y el proceso de verificación de las señales mediante un osciloscopio de la marca OWON.

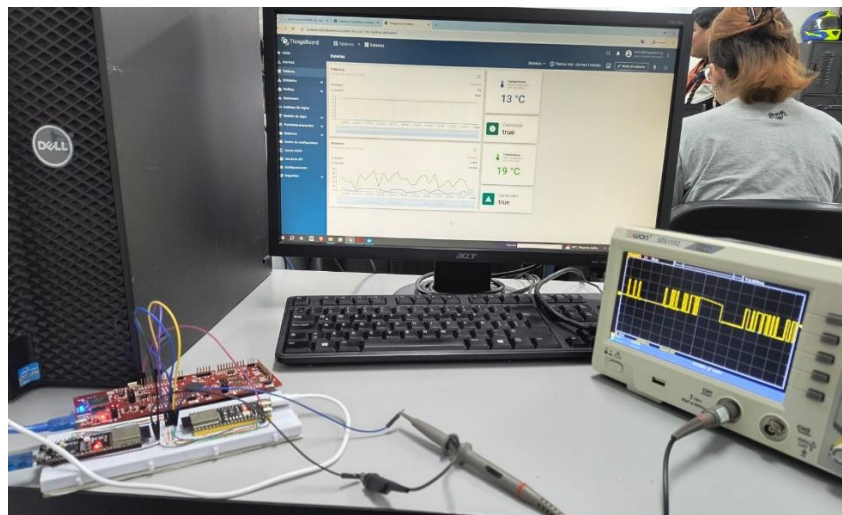



Figura 28. Sistema final

	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

5. CONCLUSIONES, RECOMENDACIONES Y TRABAJO FUTURO

CONCLUSIONES


El presente trabajo demuestra que es posible desarrollar e implementar un sistema de comunicación embebido basado en el protocolo SPI que permite la transferencia sincronizada y confiable de variables eléctricas entre diferentes plataformas de microcontroladores bajo una arquitectura maestro–esclavo.

Los resultados experimentales evidencian que la transmisión de tramas estructuradas de 13 bytes, compuestas por variables en formato IEEE 754, se realiza con integridad, sincronización temporal y correspondencia exacta entre los valores transmitidos y los reconstruidos en el dispositivo receptor.

Como fortalezas del sistema se destacan su determinismo temporal, la interoperabilidad entre dispositivos de distintos fabricantes y la validación eléctrica mediante análisis con osciloscopio. Como limitaciones, se identifican restricciones de desempeño en algunos microcontroladores en modo esclavo a frecuencias elevadas, así como la necesidad de considerar compatibilidad de niveles lógicos para evitar distorsiones en la señal.

RECOMENDACIONES

1. Considerar la distancia física entre dispositivos al implementar el sistema, dado que el protocolo SPI está diseñado para comunicación de corta distancia dentro de una misma placa o entre módulos cercanos. Se recomienda mantener longitudes de


	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

conexión reducidas y adecuadamente organizadas para evitar degradación de la señal.


2. Tener en cuenta que, aunque las pruebas demostraron estabilidad a 1 MHz, al incrementar la frecuencia hacia 10 MHz se pueden generar deformaciones en la trama de datos, tales como distorsión de flancos, atenuación y posibles errores de sincronización. Por lo tanto, frecuencias elevadas requieren trazados adecuados, cables cortos y, de ser necesario, técnicas de adaptación de impedancia.
3. Evaluar el uso de controladores con mayor capacidad de procesamiento cuando se requieran frecuencias altas en modo esclavo, con el fin de evitar retardos artificiales en la transmisión, como los observados en el Arduino UNO.
4. Tener especial precaución frente a la presencia de campos electromagnéticos en entornos eléctricos o de potencia, ya que estos pueden inducir ruido en las líneas de comunicación y afectar la integridad de la señal. En aplicaciones reales se recomienda emplear blindaje, adecuada gestión de tierras y rutas de señal separadas de líneas de potencia.

TRABAJO FUTURO

1. Evaluar el desempeño del sistema a mayores distancias físicas entre dispositivos, analizando la degradación de la señal SPI en función de la longitud del cableado y su impacto en la integridad de las tramas. Esto permitirá establecer límites prácticos de implementación más allá de entornos de laboratorio.
2. Realizar un estudio detallado de integridad de señal a frecuencias superiores a 1 MHz, particularmente alrededor de 10 MHz, donde se observaron deformaciones en los flancos de la señal y posibles alteraciones en la trama de datos. Este análisis podría complementarse con simulaciones eléctricas y mediciones de tiempo de subida.


 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

3. Analizar el comportamiento del sistema en entornos con alta interferencia electromagnética, típicos de sistemas de potencia y convertidores electrónicos, evaluando el uso de técnicas de blindaje, filtrado, separación de tierras y aislamiento galvánico.
4. Integrar la arquitectura desarrollada en un sistema de almacenamiento energético (BESS) o banco de pruebas real, permitiendo validar el desempeño del sistema bajo condiciones operativas continuas y no únicamente en pruebas controladas.


 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

REFERENCIAS

- Analog Devices. (2020). *Serial peripheral interface (SPI) overview*. Analog Devices, Inc.
- Chen, L., & Zhang, Q. (2020). Embedded communication systems for energy storage management. *IEEE Transactions on Industrial Electronics*, 67(8), 6523–6532.
- Control Automático Educación. (s. f.). *Comunicación SPI con PIC*. <https://controlautomaticoeducacion.com/sistemas-embebidos/microcontroladores-pic/comunicacion-spi/>.
- Electronilab. (s. f.). *Convertor niveles 2 canales bidireccional 5V–3.3V*. <https://electronilab.co/tienda/convertor-niveles-2-canales-bidireccional-5v-3-3v/>.
- García, M., López, F., & Ruiz, D. (2021). Control and testing of BESS systems in laboratory environments. *Energy Systems Journal*, 15(4), 1001–1013.
- IEC. (2020). *IEC 61850: Communication networks and systems for power utility automation*. International Electrotechnical Commission.
- IEEE. (2019). *IEEE standard for floating-point arithmetic (IEEE Std 754-2019)*. Institute of Electrical and Electronics Engineers.
- International Energy Agency. (2022). *Energy storage and renewable integration*. IEA.
- GeeksforGeeks. (2020, 16 marzo). *IEEE Standard 754 Floating Point Numbers*. Recuperado de <https://www.geeksforgeeks.org/computer-organization-architecture/ieee-standard-754-floating-point-numbers/>.
- Ministerio de Minas y Energía. (2023). *Avances de la transición energética en Colombia*. Gobierno de Colombia.
- SoC-eame. (2024, 2 junio). *Qué son los modos del SPI*. <https://soceame.wordpress.com/2024/06/02/que-son-los-modos-del-spi/>.
- Texas Instruments. (2019). *Understanding serial communication protocols in embedded systems*. Texas Instruments.


 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

- Unidad de Planeación Minero-Energética (UPME). (2023). *Plan energético nacional: Fuentes renovables y almacenamiento*. UPME.

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01- 22

APÉNDICE

- Apéndice A. Guía de montaje y uso del sistema de comunicación SPI.
- Apéndice B. Código Master (Master.ino).
- Apéndice C. Código Slave ESP32 (slave.rar).
- Apéndice D. Código Slave Texas Instruments (spi_ex1_loopback.rar).

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22


 FIRMA ESTUDIANTES _____



 FIRMA ASESOR


FECHA ENTREGA: _____

FIRMA COMITÉ TRABAJO DE GRADO DE LA FACULTAD _____

RECHAZADO__ ACEPTADO__ ACEPTADO CON
 MODIFICACIONES _____

ACTA NO. _____

FECHA ENTREGA: _____

 Institución Universitaria	INFORME FINAL DE TRABAJO DE GRADO	Código	FDE 089
		Versión	03
		Fecha	2015-01-22

FIRMA CONSEJO DE FACULTAD _____

ACTA NO. _____

FECHA ENTREGA: _____