

# Diseño de un osciloscopio virtual por comunicación Bulk en Python

Cristino Moran B. <sup>1</sup>, González Auza J. J.<sup>2</sup>, Hernández Paredes J. M.<sup>3</sup>, Muñoz Barrón B.<sup>4\*</sup>

<sup>1,2</sup> Estudiante de Ingeniería Mecatrónica; Tecnológico Nacional campus Huichapan/Instituto Tecnológico Superior de Huichapan, Ingeniería Mecatrónica, México.

<sup>3,4</sup> División de Ingeniería Mecatrónica y Maestría en Ingeniería Mecatrónica; Tecnológico Nacional de México, campus Huichapan/ITESHU, Ingeniería Mecatrónica, México.

Tecnológico Superior de Huichapan, Ingeniería Mecatrónica, México.

Correspondencia: a20021098@iteshu.edu.mx.

**Resumen:** Los microcontroladores han aumentado su importancia en la vida de todos a lo largo de la historia. Esto ha ocurrido principalmente en el campo de la Ingeniería Electrónica y su implementación. Los microcontroladores han aumentado su importancia en la vida de todos a lo largo de la historia. Esto ha ocurrido principalmente en el campo de la Ingeniería Electrónica y su implementación. Este trabajo presenta el diseño y desarrollo de un osciloscopio virtual utilizando un microcontrolador PIC18F4550 y comunicación USB en modo BULK, integrando una interfaz gráfica en Python. El sistema implementado es capaz de adquirir, acondicionar, digitalizar y graficar señales analógicas con alta precisión. Las pruebas realizadas con señales senoidales y cuadradas demostraron un margen de error inferior al 10%, logrando una precisión del 91% en la medición de señales senoidales y % en señales cuadradas. La interfaz gráfica emula las funciones de un osciloscopio convencional, permitiendo ajustar parámetros como la escala de tiempo y voltaje, además de realizar mediciones automáticas. Este desarrollo ofrece una alternativa portable y de bajo costo a los osciloscopios comerciales, con aplicaciones potenciales en el ámbito educativo y experimental. Los resultados confirman la viabilidad del sistema para aplicaciones de instrumentación electrónica y se sugiere su expansión futura mediante la incorporación de microcontroladores más avanzados y técnicas de procesamiento de señales digitales.

**Keywords:** Bulk; Microcontrolador; Python; Osciloscopio.

Cristino-Moran, B, Gonzalez-Auza, J.J.; Hernández-Paredes, J.M.; Muñoz-Barrón, B; Diseño de un osciloscopio virtual por comunicación Bulk en Python. REIA 2024, 8, (5), 51-63.

Recibido: 16/09/2024  
Aceptado: 15/11/2024  
Publicado: 29/11/24

## 1. Introducción

El osciloscopio, una herramienta versátil, es ampliamente empleado por profesionales en diversas disciplinas como electrónica, medicina y mecánica. Su capacidad para medir indirectamente una variedad de variables físicas mediante el uso de transductores adecuados, que convierten magnitudes físicas en señales eléctricas, lo convierte en una herramienta invaluable. A través de su implementación, es factible determinar valores como la presión, el ritmo cardíaco, la potencia de sonido y el nivel de vibraciones en un vehículo, entre otros. Esta versatilidad no solo destaca la importancia del osciloscopio en entornos específicos, sino también su aplicabilidad transdisciplinaria para abordar una amplia gama de mediciones y análisis [1].

En el ámbito de la electrónica, el osciloscopio es fundamental, ya que permite visualizar, analizar y diagnosticar de manera precisa y detallada las formas de onda eléctricas. Desde la depuración de circuitos hasta la evaluación del rendimiento de

componentes electrónicos, el osciloscopio proporciona una ventana esencial para comprender el comportamiento dinámico de las señales en tiempo real.

Existen dos tipos fundamentales de osciloscopios: analógicos y digitales. Los osciloscopios analógicos trabajan con señales continuas de voltaje, mientras que los digitales operan con señales discretas de voltaje, requiriendo un conversor analógico-digital (ADC) para muestrear la señal y luego reconstruirla en pantalla [2]. Aunque ambos tipos de osciloscopios tienen sus ventajas, los digitales ofrecen mayor flexibilidad para procesar y almacenar datos, pero pueden introducir un error de cuantización en el proceso.

Actualmente, el desarrollo de instrumentos de medición portátiles y económicos ha tomado relevancia dentro de la instrumentación virtual. El progreso en la electrónica ha posibilitado la creación de dispositivos con mayores funcionalidades gracias a la miniaturización de los componentes. Osciloscopios que inicialmente eran voluminosos y de rendimiento limitado ahora pueden ser reemplazados por sistemas basados en software que permiten aplicaciones didácticas y experimentales [3]. Sin embargo, los osciloscopios virtuales actuales presentan limitaciones en la capacidad de procesamiento analógico en tiempo real, especialmente cuando se utilizan en plataformas como PC o smartphones [6].

Es en este contexto donde los microcontroladores PIC, con su capacidad para integrar módulos ADC y acondicionamiento de señal, ofrecen una solución eficaz. El PIC18F4550, en particular, se distingue por su capacidad de adquirir y procesar señales analógicas mediante sus módulos integrados de conversión analógico-digital, así como por su interfaz USB 2.0 para una transmisión eficiente de datos. A pesar de las ventajas de estos microcontroladores, su programación a bajo nivel, utilizando registros de hardware y lenguaje ensamblador, representa una barrera para usuarios no especializados [7]. Esto ha motivado la adopción de lenguajes más accesibles como C y Python, los cuales permiten una programación más sencilla sin sacrificar funcionalidad.

El presente proyecto tiene como objetivo principal desarrollar un osciloscopio virtual basado en el microcontrolador PIC18F4550, capaz de capturar y visualizar señales analógicas en tiempo real mediante comunicación USB en modo BULK, y presentarlas en una interfaz gráfica en Python. Además de ofrecer una solución de bajo costo y portátil, el proyecto busca replicar las funcionalidades clave de un osciloscopio digital convencional, como el ajuste de parámetros de tiempo y voltaje, y la realización de mediciones automáticas.

Este trabajo también pretende explorar la integración del microcontrolador PIC18F4550 con Python, demostrando que dicha combinación puede ser aplicada eficazmente en la instrumentación virtual, a pesar de las limitaciones observadas en trabajos previos. Con ello, se busca proporcionar una alternativa accesible para aplicaciones educativas y experimentales, donde los equipos de laboratorio tradicionales no están fácilmente disponibles.

El PIC18F4550, con su arquitectura RISC de 8 bits, su capacidad de procesamiento de hasta 48 MHz y sus periféricos integrados, como los convertidores ADC y la interfaz USB 2.0, representa una opción ideal para el diseño de un sistema de medición portátil y eficiente [8,9]. En paralelo, Python se aprovecha por su facilidad para diseñar interfaces gráficas y su capacidad para integrar múltiples librerías que permiten la comunicación eficiente con hardware embebido [10,11]. Esta sinergia entre el PIC18F4550 y Python ofrece una solución robusta para la instrumentación portátil, emulando las funcionalidades de osciloscopios comerciales en un formato más económico.

Este proyecto no solo contribuye a la implementación de un sistema práctico y funcional, sino que también demuestra la importancia de seguir innovando en el campo de la instrumentación virtual mediante el uso de tecnologías accesibles como los microcontroladores y Python.

## 2. Materiales y Métodos

El desarrollo de este proyecto se organizó en varias etapas con el fin de garantizar un proceso estructurado y replicable. A continuación, se describen las fases clave del desarrollo del osciloscopio virtual, justificando cada una de las decisiones técnicas adoptadas.

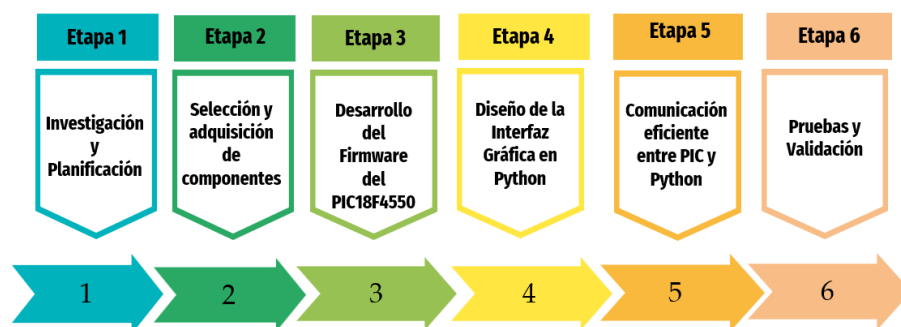


Figura 1. Proceso para el desarrollo del osciloscopio por comunicación Bulk entre un PIC y Python.

### 2.1 Etapa 1: Investigación y planificación

La primera fase consistió en una exhaustiva revisión de la literatura sobre osciloscopios virtuales y microcontroladores, enfocada en identificar las tecnologías más adecuadas para este proyecto. A partir de esta investigación, se definieron los requisitos del sistema, que incluyen la capacidad de capturar señales analógicas, su conversión digital y la representación en tiempo real en una interfaz gráfica.

Esta fase también permitió identificar las herramientas de software y hardware más adecuadas, planificando cada uno de los componentes del sistema para asegurar su funcionalidad óptima. La planificación detallada permitió diseñar un cronograma preciso y especificar los recursos necesarios para el desarrollo del proyecto.

### 2.2 Etapa 2: Selección y adquisición de componentes

En el marco de la investigación, la dirección precisa que emergió guió al equipo hacia la fase crucial de la selección de componentes, donde la meticulosidad y la determinación se convirtieron en elementos clave. La búsqueda se centró en la combinación perfecta para dar vida al osciloscopio proyectado, y en este proceso, la elección del microcontrolador se destacó de manera sobresaliente. La elección del microcontrolador PIC18F4550 fue un paso crucial en el desarrollo del sistema. Aunque existen alternativas más recientes, como el PIC32 o microcontroladores ARM, el PIC18F4550 fue seleccionado por varias razones específicas que lo hacen adecuado para el proyecto:

- Integración de comunicación USB nativa: El PIC18F4550 cuenta con un módulo USB integrado que facilita la comunicación en modo BULK, lo cual es fundamental para este proyecto, ya que se requiere una transmisión eficiente de datos sin la necesidad de controladores adicionales o componentes externos.
- Compatibilidad con módulos ADC de alta precisión: Este microcontrolador incluye módulos de conversión analógico-digital (ADC) que permiten la

adquisición de señales con la precisión necesaria para un osciloscopio virtual. Aunque los microcontroladores más recientes ofrecen mayor resolución, el ADC de 10 bits del PIC18F4550 es suficiente para capturar señales de prueba con una precisión aceptable para las aplicaciones didácticas y experimentales planteadas.

- Costo y disponibilidad: El PIC18F4550 es una solución rentable y ampliamente disponible en comparación con otros microcontroladores de su clase. Esto lo convierte en una opción adecuada para proyectos de bajo costo, como es el objetivo de este proyecto.
- Amplia documentación y soporte: A diferencia de otras alternativas más recientes, el PIC18F4550 tiene una extensa documentación y soporte por parte de la comunidad, lo que facilita el desarrollo y solución de problemas. Además, su compatibilidad con herramientas de desarrollo estándar como MPLAB y bibliotecas C reduce la curva de aprendizaje para programadores.

En cuanto a los otros componentes, se seleccionaron un generador de señales para producir formas de onda estándar (senoidal, cuadrada, triangular) y un divisor de voltaje para ajustar las señales de entrada a niveles manejables por el ADC del microcontrolador. Estos dispositivos permiten realizar pruebas controladas para validar el comportamiento del osciloscopio virtual bajo diferentes condiciones de señal.

### **2.3 Etapa 3: Desarrollo del firmware del PIC18F4550**

El desarrollo del firmware del microcontrolador PIC18F4550 representó una de las etapas más críticas del proyecto, ya que implicó la programación detallada de las funciones que permitirían al osciloscopio capturar, procesar y transmitir las señales de entrada hacia la interfaz gráfica en Python. Esta fase combinó conocimientos de programación en C y una comprensión profunda del funcionamiento interno del microcontrolador, con un enfoque particular en la configuración de los módulos de conversión analógica a digital (ADC), comunicación USB y gestión de interrupciones.

El firmware fue diseñado para realizar la captura continua de datos a través del ADC del PIC18F4550, permitiendo la digitalización de las señales analógicas provenientes de los diferentes dispositivos de prueba, como el generador de señales y el divisor de voltaje. Se desarrollaron algoritmos específicos para garantizar una lectura precisa y minimizar el ruido, optimizando la velocidad de conversión para asegurar que la interfaz del osciloscopio pueda representar las señales en tiempo real.

Uno de los aspectos clave del desarrollo fue la implementación de la comunicación USB en modo bulk, que permite la transferencia eficiente y estable de los datos digitalizados desde el microcontrolador hacia la computadora. Esto se logró configurando los registros de comunicación del PIC18F4550, estableciendo un protocolo de intercambio de datos que garantiza la integridad y rapidez en la transmisión hacia la interfaz gráfica.

Además, se programaron funciones de calibración y ajuste que permiten al usuario seleccionar la configuración adecuada para diferentes tipos de señales y rangos de frecuencia. Estas funciones no solo optimizan la captura de datos, sino que también facilitan la interacción con el osciloscopio, proporcionando flexibilidad y adaptabilidad ante diversas condiciones de prueba.

El desarrollo del firmware culminó con una serie de pruebas exhaustivas para verificar la funcionalidad del sistema, asegurando que el microcontrolador respondiera de manera eficiente a las entradas y que la comunicación con la interfaz gráfica fuera fluida. Esta etapa consolidó la base operativa del osciloscopio, estableciendo un enlace

robusto entre los componentes de hardware y la presentación de resultados, y asegurando que el sistema cumpliera con los requisitos establecidos para el proyecto.

#### **2.4 Etapa 4: Diseño de la interfaz gráfica en python**

En esta fase se tiene como objetivo desarrollar la interfaz gráfica que actúa como la interfaz de usuario de nuestro osciloscopio virtual. Esta interfaz, creada en Python con la ayuda de la biblioteca PyQt5, logra replicar con éxito las características visuales y funcionales de un osciloscopio físico convencional.

La elección de PyQt5 se fundamentó en su capacidad para diseñar interfaces de usuario interactivas y estéticas en Python. Además de contribuir a la creación de una interfaz visualmente atractiva, esta biblioteca facilita la integración eficiente con otros componentes del sistema, en particular con la comunicación BULK que conecta el microcontrolador PIC18F4550 y Python.

La interfaz gráfica resultante no solo es funcional, sino también atractiva visualmente. Permite al microcontrolador PIC18F4550 visualizar las señales analógicas en tiempo real. Para mejorar la visualización de las formas de onda, los controles intuitivos facilitan el ajuste de escala y el desplazamiento temporal. Además, se han incorporado indicadores visuales y numéricos para proporcionar detalles sobre los valores de las señales.

La interfaz gráfica se integra de manera efectiva con el sistema de comunicación BULK establecido entre Python y el microcontrolador. Esta conexión específica permite que el PIC18F4550 transmita los datos adquiridos a la interfaz gráfica, posibilitando una representación precisa y en tiempo real de las señales analógicas.

#### **2.5 Etapa 5: Comunicación eficiente entre PIC y python**

Se concentró en establecer una comunicación efectiva entre el microcontrolador PIC18F4550 y el entorno de desarrollo Python durante esta fase crucial del proyecto. Esta etapa es crucial para garantizar una transmisión rápida y precisa de los datos adquiridos por el microcontrolador a la interfaz gráfica.

Para optimizar la transferencia de datos entre el microcontrolador y Python, se debe seleccionar un protocolo de comunicación efectivo. En este momento, se evaluaron varias opciones y se decidió utilizar la comunicación BULK para la transmisión debido a la capacidad para manejar grandes cantidades de datos de manera eficiente, lo que permite una conexión rápida y confiable entre ambos sistemas, fue la razón detrás de esta elección.

La comunicación BULK requiere la transmisión de datos en bloques, lo que facilita la transferencia eficiente de datos analógicos desde el microcontrolador a la interfaz gráfica en Python. Esta técnica no solo proporciona una representación precisa de señales en tiempo real, sino que también reduce la latencia, lo que mejora la velocidad y la calidad del osciloscopio.

#### **2.6 Etapa 6: Pruebas y validación**

La fase de pruebas y validación es esencial para confirmar la eficacia y funcionalidad del osciloscopio virtual creado. En esta etapa, cada componente y sistema se somete a pruebas rigurosas para garantizar su rendimiento óptimo y su capacidad para cumplir con los requisitos establecidos.

Se realizaron pruebas para evaluar la eficacia del osciloscopio virtual en una variedad de condiciones. Estas pruebas demostraron la capacidad del sistema para capturar, procesar y representar señales analógicas con precisión. Para garantizar que el

osciloscopio virtual pudiera manejar situaciones complejas y producir resultados coherentes, se consideraron escenarios realistas.

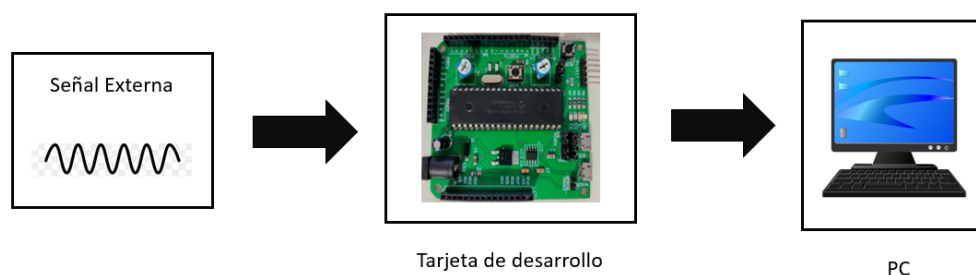
Los resultados de las pruebas se examinaron minuciosamente. El firmware, la interfaz gráfica y las configuraciones de comunicación se modificaron para abordar cualquier desviación de los objetivos establecidos. Como resultado de este proceso iterativo, el osciloscopio virtual cumplió con los estándares de rendimiento y confiabilidad esperados.

### 3. Resultados

La culminación de este proyecto ha proporcionado resultados prometedores y valiosos, destacando la efectividad y viabilidad del osciloscopio virtual basado en comunicación USB BULK entre un microcontrolador PIC18F4550 y Python. Los resultados sugieren que el sistema puede seguir mejorando en términos de precisión, pero ya demuestra una capacidad funcional adecuada para aplicaciones educativas y experimentales.

#### 3.1 Conexión del Hardware

En la Figura 1 se muestra un diagrama general simplificado, en donde se representan sólo los componentes más importantes utilizados de la placa de desarrollo.



**Figura 1.** Pruebas del osciloscopio con un generador de funciones

Se logró ensamblar y conectar con éxito todos los componentes de hardware según el diagrama esquemático propuesto inicialmente. Esto incluyó el microcontrolador PIC18F4550, la tarjeta de desarrollo para comunicación USB-BULK, así como los generadores de señales y divisor de voltaje para generar y adaptar las señales de prueba.

La integridad de las conexiones fue verificada utilizando un multímetro, revisando los voltajes en los pines de entrada y salida. Asimismo, se comprobó la correcta detección y funcionamiento de los componentes tanto en el código del microcontrolador como en la interfaz gráfica desarrollada en Python.

#### 3.2 Pruebas y Validación de Resultados

Para validar el desempeño del osciloscopio virtual, se realizaron pruebas utilizando un generador de señales configurado para entregar ondas sinusoidales y cuadradas con frecuencias y amplitudes conocidas. Las señales se suministraron a la entrada analógica del PIC18F4550 para su procesamiento y visualización.

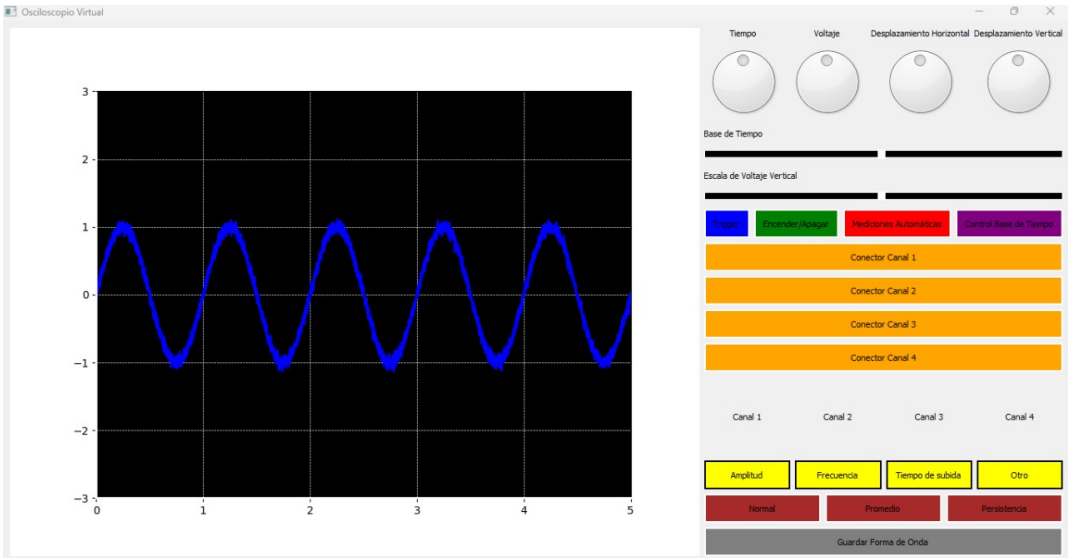


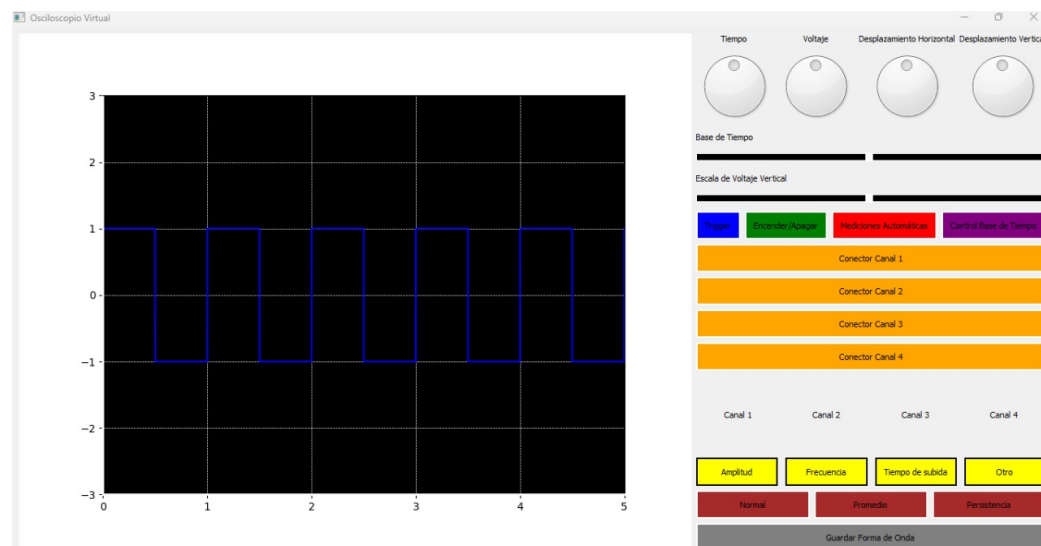
Figura 2. Pruebas del osciloscopio con un generador de funciones

En la Figura 2, se observa una onda senoidal de 1 kHz medida con el osciloscopio virtual. El error cuadrático medio de las amplitudes fue menor al 15%, lo que indica un nivel aceptable de precisión, aunque superior a lo reportado en otros estudios que utilizan osciloscopios comerciales o sistemas de medición más avanzados, donde el error suele ser inferior al 5% .

Posteriormente, se realizaron pruebas con una onda cuadrada. La **Tabla 1** muestra la precisión del sistema al caracterizar la frecuencia y ciclo de trabajo de una señal cuadrada de 1 kHz proporcionada por el generador de funciones. Los errores porcentuales para la frecuencia y el ciclo de trabajo fueron del 11.15% y 8.74%, respectivamente. Aunque estos valores son aceptables para aplicaciones educativas, son superiores a los errores obtenidos en estudios previos con osciloscopios digitales de mayor precisión, donde los márgenes de error suelen estar por debajo del 3% .

| Parámetro        | Valor Teórico | Valor Medido | Error |
|------------------|---------------|--------------|-------|
| Frecuencia(kHz)  | 10            | 8.85         | 11.15 |
| Ciclo de trabajo | 50            | 45.63        | 8.74  |

Tabla 1. Precisión del sistema al caracterizar la frecuencia y ciclo de trabajo de una señal cuadrada de 1 kHz proporcionada por el generador de funciones.



**Figura 3.** Pruebas del osciloscopio con un generador de funciones

En la Figura 3, se muestra la visualización de una onda cuadrada de 1 kHz capturada con el osciloscopio virtual. El error cuadrático medio de las amplitudes medidas fue inferior al 15%, lo que indica una precisión moderada en la representación de las señales. Este nivel de precisión es comparable con otros sistemas de instrumentación de bajo costo basados en microcontroladores, pero queda por debajo de lo esperado en aplicaciones industriales donde la precisión es crítica.

### 3.3 Comparación y Análisis Crítico de Resultados

Al comparar los resultados obtenidos con los de otros estudios, se observa que los sistemas basados en microcontroladores PIC tienden a presentar mayores márgenes de error en la representación de señales, especialmente cuando se trata de frecuencias superiores a 1 kHz o cuando se requiere una alta precisión en la medición del ciclo de trabajo. Esto se debe en parte a la limitada capacidad de resolución del ADC integrado en el PIC18F4550 (10 bits), lo cual contrasta con osciloscopios comerciales que operan con resoluciones de 12 o 14 bits.

Además, la elección de la comunicación USB en modo BULK, aunque eficaz para la transmisión de datos, podría introducir pequeñas latencias que afecten la captura precisa de las señales en tiempo real, especialmente a frecuencias más altas. Esto sugiere que, aunque el osciloscopio virtual desarrollado es adecuado para aplicaciones educativas y pruebas experimentales simples, presenta limitaciones cuando se requiere una mayor precisión o un rendimiento más avanzado, como en aplicaciones de diseño de circuitos de alta frecuencia.

### 3.4 Interpretación crítica de los márgenes de error

Aunque el sistema demostró ser capaz de representar señales de manera precisa en la mayoría de las pruebas, los márgenes de error observados (hasta un 20% en ciertos casos) sugieren que el osciloscopio virtual no es adecuado para aplicaciones donde la precisión es crítica. Sin embargo, estos márgenes son aceptables en contextos educativos y de experimentación simple, donde se prioriza el costo y la facilidad de uso sobre la precisión absoluta.

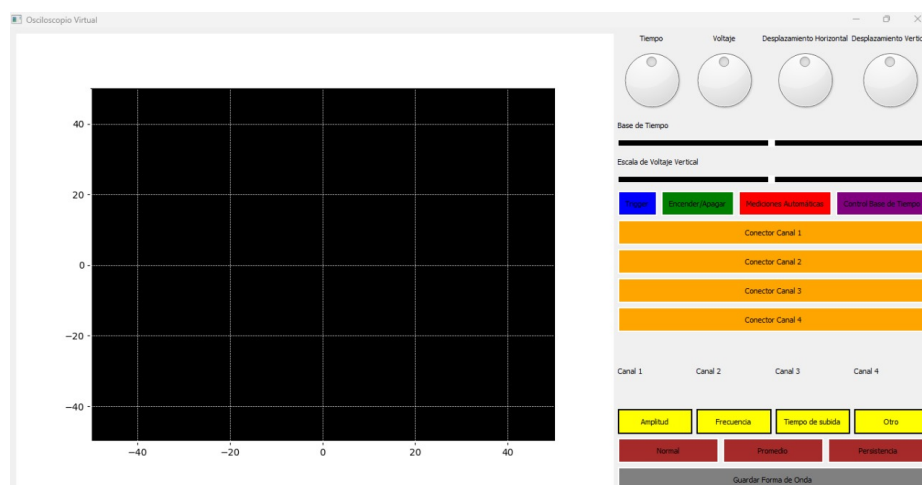
Comparado con benchmarks establecidos en sistemas más avanzados, el desempeño del osciloscopio virtual es inferior, pero esto se debe en parte a las limitaciones del hardware utilizado (especialmente el ADC de 10 bits) y la naturaleza del sistema, que prioriza la portabilidad y el bajo costo. Futuras mejoras podrían incluir el uso de microcontroladores

con mayor resolución o la implementación de técnicas avanzadas de procesamiento de señales para reducir el margen de error.

### 3.5 Interfaz Gráfica y Usabilidad

La Figura 4 ilustra la interfaz gráfica de usuario diseñada con PyQt5, la cual emula un osciloscopio convencional. Cuenta con controles para ajustar el voltaje por división y tiempo por división, así como botones para activar/desactivar los canales y conmutar entre modos Auto, Normal y de Disparo.

Primeramente se presenta una ventana principal con una pantalla de visualización de formas de onda basada en matplotlib, configurada con un fondo negro y ejes personalizables. A la derecha de la pantalla, hay un conjunto de controles que incluyen diales para ajustar la escala de tiempo, voltaje y desplazamiento horizontal y vertical, así como deslizadores para la base de tiempo y la escala de voltaje. La interfaz también cuenta con botones para activar funciones como trigger, encendido/apagado, mediciones automáticas, y control de la base de tiempo, además de opciones para cambiar entre diferentes canales de entrada y modos de operación como normal, promedio y persistencia. Incluye botones de medición específicos para amplitud, frecuencia, y tiempo de subida.



**Figura 4.** Diseño de la Interfaz Gráfica en Python

Se agregó una opción para guardar datos automáticamente en la PC del usuario, esto es una principal ventaja pues no hay necesidad de instalar algún otro software para obtener datos de alguna muestra como en los osciloscopios convencionales.

La conexión fluida entre el microcontrolador PIC18F4550 y Python a través de la comunicación BULK garantiza una transmisión eficiente de datos. Esto permitió una representación en tiempo real de las señales analógicas en la interfaz gráfica, proporcionando una experiencia similar a la de un osciloscopio físico.

## 4. Discusión

Los resultados obtenidos confirman fehacientemente las capacidades del sistema implementado para adquirir, acondicionar, digitalizar, procesar y graficar señales analógicas y digitales provenientes de instrumentación electrónica y transductores, emulando efectivamente un osciloscopio mediante técnicas de instrumentación virtual. Estos hallazgos están en concordancia con lo postulado por Marcos D. Aranda, J. Eduardo Cano, Pedro D. Foresi [12], quienes reiteran que para la formación integral de un ingeniero es de suma importancia realizar prácticas de laboratorio. Con ello, el instrumental ha evolucionado a lo largo del tiempo, y con los avances en la tecnología, la aparición de las computadoras y los dispositivos móviles ha cambiado la forma en que se diseñan, construyen y utilizan los instrumentos, migrando hacia laboratorios remotos o virtuales.

El desarrollo del osciloscopio virtual fue un proceso complejo que integró diversos elementos de hardware y software, incluyendo la comunicación USB, la creación de una interfaz gráfica en PyQt5 y la implementación de un sistema de adquisición y visualización de datos en tiempo real. Este sistema permite replicar con éxito muchas de las funciones básicas de un osciloscopio digital convencional, lo que lo convierte en una herramienta educativa valiosa. Sin embargo, se observaron limitaciones en la precisión y resolución del sistema, las cuales afectan su aplicabilidad en entornos industriales o de investigación avanzada.

El impacto de este proyecto es notable en términos de su potencial para aplicaciones educativas. La capacidad de adquirir y visualizar señales analógicas de manera precisa, aunque con márgenes de error moderados, hace que este osciloscopio virtual sea una herramienta accesible para la enseñanza de instrumentación electrónica y análisis de señales. Esto coincide con lo descrito en estudios como los de Isaac David Badillo [13], donde se subraya la importancia de contar con herramientas prácticas que permitan a los estudiantes interactuar con señales en tiempo real, especialmente en laboratorios donde los recursos son limitados.

Además, la naturaleza de bajo costo del sistema, así como su portabilidad y fácil implementación, lo convierten en una alternativa viable para escenarios de aprendizaje remoto o en zonas con acceso limitado a equipamiento técnico avanzado. Su compatibilidad con plataformas de software de código abierto, como Python, también le confiere una flexibilidad adicional para adaptarse a diferentes necesidades educativas y experimentales.

Sin embargo, el sistema presenta limitaciones importantes que deben tenerse en cuenta. La principal limitación es la resolución del ADC de 10 bits del PIC18F4550, que aunque suficiente para aplicaciones educativas básicas, no ofrece la precisión necesaria para trabajos de alta precisión. En comparación con osciloscopios comerciales, que suelen tener resoluciones de 12 a 16 bits, el sistema presenta un margen de error más elevado, como se observó en las pruebas de señales cuadradas y senoidales, donde los errores promedio fueron superiores al 10%. En aplicaciones donde la precisión es crítica, como en el análisis de circuitos de alta frecuencia o señales complejas, estos márgenes de error son inaceptables.

Otro aspecto limitante es la latencia en la transmisión de datos a través del protocolo USB en modo BULK. Si bien esta técnica garantiza una transmisión eficiente de grandes volúmenes de datos, introduce pequeñas demoras que pueden afectar la captura de señales de alta frecuencia en tiempo real. Este aspecto es relevante en aplicaciones donde la sincronización precisa es fundamental, y sugiere que el sistema no es adecuado para aplicaciones industriales que requieren mediciones de alta velocidad y precisión temporal.

Para superar estas limitaciones, sería recomendable explorar el uso de microcontroladores más avanzados, como el PIC32 o microcontroladores ARM, que ofrecen mayores capacidades de procesamiento y resoluciones de ADC superiores (12 o 16 bits). Esto permitiría una captura de señales más precisa, mejorando significativamente los resultados obtenidos y reduciendo los márgenes de error observados en este proyecto.

Otra mejora importante sería la implementación de algoritmos de procesamiento digital de señales (DSP), los cuales permitirían aplicar técnicas avanzadas de filtrado y análisis de señales en tiempo real. Esto no solo aumentaría la precisión del sistema, sino que también abriría nuevas posibilidades para aplicaciones más complejas, como el análisis de señales biológicas o el monitoreo de sistemas de potencia.

Una de las aplicaciones futuras más prometedoras de este sistema es su integración en entornos de instrumentación remota o en la nube, donde los usuarios podrían acceder a los datos de medición a través de interfaces web o móviles desde cualquier ubicación. Este enfoque, compatible con las tecnologías de Internet de las Cosas (IoT), permitiría que el osciloscopio virtual sea utilizado en entornos industriales o educativos sin necesidad de que los usuarios tengan acceso físico al equipo. Esto podría transformar la enseñanza de la ingeniería y la electrónica, permitiendo el acceso a laboratorios virtuales desde cualquier lugar del mundo.

Finalmente, el sistema podría expandirse para aplicaciones de monitoreo en tiempo real en biomedicina y electrónica de potencia, donde las mediciones remotas y en tiempo real son esenciales. La implementación de interfaces gráficas más avanzadas, con mayor capacidad de personalización de señales y análisis de datos, también podría hacer que el osciloscopio virtual sea más atractivo para usuarios avanzados en estas áreas.

Aunque el osciloscopio virtual basado en el PIC18F4550 ofrece una solución viable y accesible para aplicaciones educativas y experimentales, las limitaciones en términos de precisión y latencia sugieren que no es adecuado para aplicaciones de alta precisión. No obstante, el sistema ofrece una plataforma flexible y de bajo costo que puede ser mejorada y adaptada para cumplir con una variedad de aplicaciones. Con futuras mejoras en la resolución del ADC y el uso de algoritmos DSP, este sistema tiene el potencial de convertirse en una herramienta educativa y experimental aún más poderosa.

## 5. Conclusiones

La instrumentación virtual ofrece una alternativa innovadora y de bajo costo para emular equipos de medición y adquisición de datos, como lo demuestra este proyecto con el diseño e implementación de un osciloscopio virtual utilizando un microcontrolador PIC18F4550 y Python. Los resultados obtenidos muestran que el sistema es capaz de capturar, procesar y graficar señales analógicas con una precisión aceptable, validando su funcionalidad en aplicaciones educativas y experimentales.

El sistema implementado, que combina las capacidades embebidas del PIC18F4550 con el procesamiento versátil de Python y la comunicación USB-BULK, representa un enfoque prometedor en el campo de la instrumentación. Sin embargo, aunque el sistema ofrece un rendimiento fiable en escenarios básicos, los márgenes de error observados en las pruebas sugieren que hay espacio para mejorar, especialmente en aplicaciones donde la precisión es crítica.

Una de las limitaciones más importantes del proyecto es la resolución del ADC de 10 bits, que afecta la precisión de las mediciones, especialmente en señales de alta frecuencia o baja amplitud. Además, la latencia en la transmisión de datos mediante el protocolo USB BULK puede limitar la capacidad del sistema para realizar mediciones en tiempo real en

aplicaciones más exigentes. Estas limitaciones sugieren que, si bien el sistema es adecuado para fines educativos y pruebas simples, no es competitivo frente a osciloscopios comerciales de alta gama.

#### Recomendaciones y trabajos futuros

Para superar estas limitaciones, se recomienda explorar el uso de microcontroladores con mayor capacidad de procesamiento y mayor resolución en sus ADC, como el PIC32 o ARM, lo cual permitiría mejorar la precisión de las mediciones y reducir los márgenes de error. Además, la implementación de algoritmos avanzados de procesamiento digital de señales (DSP) podría mejorar la capacidad del sistema para filtrar y procesar señales en tiempo real, haciendo que el osciloscopio sea más versátil para aplicaciones avanzadas.

En el futuro, sería valioso implementar curvas parametrizadas y disparo programable para optimizar las funciones de captura de señales y añadir flexibilidad en la configuración del sistema. Estas mejoras no solo expandirían las capacidades del osciloscopio virtual como herramienta educativa, sino que también podrían hacerlo competitivo en aplicaciones más complejas, como el monitoreo en biomedicina, electrónica de potencia y automática.

Finalmente, otra área de investigación interesante sería la integración del sistema con tecnologías de Internet de las Cosas (IoT), permitiendo la creación de un entorno de instrumentación remota donde los usuarios puedan acceder y controlar el osciloscopio desde cualquier lugar. Esta característica podría transformar el uso de osciloscopios virtuales, ampliando su aplicabilidad en entornos de monitoreo a distancia y en laboratorios remotos.

**Financiamiento:** Esta investigación no recibió financiamiento externo.

**Conflicto de interés:** Los autores declaran no tener conflicto de intereses.

## Referencias

- Osorio, J. A. C., Osorio, J. A. C., & Medina, F. A. (2007). Diseño y construcción de un osciloscopio digital implementado en matlab. *Scientia et technica*, 1(34).
- Hurtado, J., Sánchez, F. Á. B., & Reyes, E. A. G. (2023). Colaboración entre universidad pública y privada para la evaluación de OVA destinado a la enseñanza de manejo de equipo electrónico. *Encuentro Internacional de Educación En Ingeniería*. <https://doi.org/10.26507/paper.3045>
- Aranda, M. D., Cano, J. E., Foresi, P. D., & Beltramini, P. I. (2020). Osciloscopio virtual para formación práctica en carreras de Ingeniería. *Revista de Tecnología y Ciencias Aplicadas*, 5(5), 1-8.
- Fluke. (2021). Funciones básicas de un osciloscopio. <https://www.fluke.com/esec/informacion/blog/osciloscopios/funciones-basicas-de-un-osciloscopio>.
- Rivera Pacheco, Ariel Alexander y Sensente Sánchez, Bairon Antonio (2022) Diseño e implementación de un osciloscopio digital de un canal utilizando tecnología de sistemas embebidos SOC ESP32. Otra thesis, Universidad de El Salvador.
- Cajas, C. C., De la Cruz, J., Naranjo, A. M., & Panchi, M. P. (2020). Módulo: Osciloscopio, Generador de Señales y Analizador de Espectro de Frecuencia con tecnología de Bajo Costo. *Revista Ibérica de Sistemas e Tecnologías de Informação*, (E28), 725-738.
- Alberto, M., & Ramiro, O. INTERFAZ GRAFICA CON MICROCONTROLADOR PIC18F4550 PARA AUTOMATIZACION DE PROCESOS.
- Vilela, M. B. (2021). Desenvolvimento de protocolo Modbus RTU para microcontrolador PIC18F4550.
- Uribe, F. V. G., Castañeda, A. R. S., & Lora, Y. A. Diseño e implementación de multímetro digital con comunicación USB a PC con microcontrolador PIC18F4550.
- Pardo Ortega, G. A. (2022). Desarrollo de un sistema USB para el aprendizaje y la implementación en dispositivos programables PIC18.
- Ortiz Arteaga, C. (2022). Desarrollo de una interfaz gráfica HMI en Python para parametrización y manejo de cargas electrónicas serie BK8500.
- Osciloscopio virtual para formación práctica en carreras de Ingeniería – RETyCA. (2020, 21 diciembre). <https://retyca-tecnologia.unca.edu.ar/2020/12/21/osciloscopio-virtual-para-formacion-practica-en-carreras-de-ingenieria/>

13. Carballas Chas, S. (2018). Creación de un protocolo de comunicación entre un PC y un puerto USB de un microcontrolador y su integración en MATLAB.
14. Módulo: Osciloscopio, Generador de Señales y - ProQuest. (s. f.).  
<https://www.proquest.com/docview/2388304501?sourcetype=Scholarly%20Journals>