

Bode Analyzer

Medición de magnitud y fase de transferencias de filtros electrónicos.

1st Alejo Domínguez Nimo
UTN FRBA
CABA, Argentina
adomnguez@frba.utn.edu.ar

2nd Joaquín Mariano Piñeiro
UTN FRBA
CABA, Argentina
jpineiro@frba.utn.edu.ar

Abstract—Conocer y caracterizar la respuesta de un filtro de un sistema LTI (lineal e invariante en el tiempo) resulta de gran utilidad para cualquier proyecto electrónico, en tanto que permite corroborar aquella respuesta física con la calculada analíticamente y simulada. Este trabajo detalla el desarrollo de un dispositivo capaz de medir la magnitud y la fase de la transferencia de un filtro de forma automatizada, a partir de un barrido en frecuencia acondicionado por un hardware externo y posterior procesamiento digital. Como resultado se obtuvo un dispositivo capaz de medir frecuencias entre 1Hz a 20KHz, con módulo entre 0dB y -45dB.

Index Terms—Module, Phase, Filter, Bode, OPAMP, Frequency

I. INTRODUCCIÓN

A la hora de diseñar un filtro, los pasos a seguir son diseño, simulación e implementación. Si bien la simulación permite corroborar la plantilla de diseño, y determinar si el filtro produce el efecto deseado a distintas frecuencias, los componentes que constituyen la arquitectura del filtro, están sometidos a distintos factores de variación como, tolerancias, temperatura, ruido eléctrico, y a otros factores vinculados con la tecnología de fabricación del componente elegido. Muchos de estos efectos pueden ser simulados, pero otros resultan difíciles de computar, o por lo menos requieren una verificación experimental. Por lo tanto, dependiendo de la importancia del comportamiento del filtro, no es trivial conocer la respuesta real del mismo.

Esta verificación experimental puede ser implementado con otros instrumentos de medición, como un osciloscopio y un generador de señales. Sin embargo, lo tedioso y dependiente del operario que es este método de medición resulta en mediciones prolongadas en el tiempo y con alto nivel de incertidumbre (en comparación a una medición con un dispositivo dedicado, como la serie de productos “STEMlab” de Redpitaya [1], o el dispositivo “Bode 100” de OMICRON LAB [2]). Por estas razones, este trabajo propone una medición automatizada de las magnitudes anteriormente mencionadas, a partir de un procesamiento analógico en conjunto con uno digital.

II. MARCO TEÓRICO

A. Sistemas Lineales e Invariantes en el Tiempo

El término sistema es sumamente amplio y abstracto. Sin embargo un enfoque general lo concibe como una estructura

o diseño que opera sobre señales de modo de producir algún efecto sobre ellas. Particularmente, no resulta necesario determinar la constitución física del sistema sino sus relaciones funcionales de forma tal de abstraerse de su construcción física.

Se denomina sistema (**H**) a un proceso que transforma señales temporales que ingresan a su entrada ($x(t)$, $x[n]$) en otras que egresan de su salida ($y(t)$, $y[n]$), mediante la interconexión de componentes, dispositivos o incluso subsistemas.

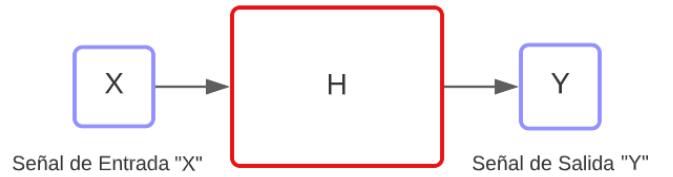


Fig. 1. La señal de entrada $x(t)$ es transformada por el sistema (operador H), dando como resultado la señal de salida $y(t)$

Los sistemas pueden ser descriptos en términos matemáticos a partir de ecuaciones algebraicas o ecuaciones diferenciales como $y(t) = 2x_{(t-1)}^2$ ó $y'(t) + 2y(t) = 4x(t)$. Adoptando el concepto de transferencia, donde una función de excitación (de entrada $x(t)$) ingresa al sistema, y se convierte en una función de salida, que puede ser descrita a partir del sistema conociendo la función matemática que lo describe: $y(t) = T[x(t)]$.

Existen muchas clasificaciones para los sistemas, pero particularmente los que interesan son los eléctricos que responden a cuestiones físicas. Sus características principales son: invariabilidad temporal y linealidad. Un sistema se denomina **lineal** si cumple simultáneamente con las propiedades de aditividad (1) y homogeneidad (2).

$$T[x_1(t) + x_2(t)] = y_1(t) + y_2(t) \quad (1)$$

$$T[\alpha \cdot x(t)] = \alpha \cdot y(t) \quad (2)$$

Un sistema es **invariante en el tiempo** si el comportamiento y características del mismo no cambian con el transcurso del tiempo, es decir, al desplazar su entrada un tiempo t_0 la

respuesta original se desplaza el mismo tiempo t_0 , sin cambios (3).

$$T[x_1(t)] = y(t) \Rightarrow T[x_1(t-t_0)] = y(t-t_0) \quad (3)$$

Por lo tanto un sistema se denomina LTI si cumple con (1), (2) y (3).

B. Respuesta al Impulso e Integral de Convolución

Cualquier señal temporal puede describirse en términos de un conjunto de señales impulso escaladas y desplazadas en el tiempo (4). Además, puede evaluarse la respuesta del sistema a una señal impulso $\delta(t)$, definida como “Respuesta Impulsional” (5). De esta manera, al expresar la señal $x(t)$ como combinación de señales impulsos escaladas y desplazadas y aplicarla a un sistema LTI, se obtendrá la suma de las respuestas impulsionales $h(t)$ escaladas y desplazadas. Así, **puede obtenerse la respuesta del sistema sin tener que recurrir a las ecuaciones diferenciales que lo modelan.**

$$x(t) = \int_{\tau=-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau \quad (4)$$

$$h(t) = T[\delta(t)] = y(t)|_{x(t)=\delta(t)} \quad (5)$$

Sabiendo esto, es posible obtener una forma matemática que describe la respuesta de salida $y(t)$ a partir de la excitación $x(t)$ y respuesta al impulso $h(t)$ (6):

$$y(t) = T\left[\int_{\tau=-\infty}^{\infty} x(\tau)\delta(t-\tau)d\tau\right] \quad (6)$$

$$y(t) = \int_{\tau=-\infty}^{\infty} x(\tau)h(t-\tau)d\tau = x(t) * h(t) \quad (7)$$

Donde la ecuación (7) se denomina **Integral de Convolución**.

C. Transformada de Fourier

Hasta este punto se conoce una forma de caracterizar el sistema en términos del tiempo. Pero un filtro se caracteriza en términos de la frecuencia. Por lo tanto es lógico aplicarle a la expresión (7) una transformada que vincule el dominio temporal con el frecuencial; la transformada de Fourier $F[]$. Aplicando la propiedad que posee esta transformada con la convolución se obtiene una expresión (8), mucho mas sencilla que vincula variables de salida con entrada. De esta manera el sistema puede ser descrito en términos del cociente entre la entrada y salida (9) a partir de su modulo $|H(f)|$ y su fase $\phi(f)$.

$$F[y(t)] = F[x(t) * h(t)] \Rightarrow Y(f) = X(f) \cdot H(f) \quad (8)$$

$$H(f) = \frac{Y(f)}{X(f)} = \frac{|Y(f)|}{|X(f)|} e^{j(\phi(Y)-\phi(X))} = |H(f)| \cdot e^{j\phi(f)} \quad (9)$$

D. Señales Armónicas

Conociendo las expresiones que vinculan frecuencialmente entrada y salida, ¿cómo se obtiene $H(f)$? ¿con que señal se excita al sistema? Estas dos preguntas no resultan triviales, pues hay una única señal que resulta la más conveniente para excitar un sistema. Hay que tener cuenta que una señal normalmente es poliarmónica, es decir, que esta compuesta por sumatorias de senos y cosenos. Tomando como ejemplo una señal cuadrada $x(t)$, al excitar con ella el sistema, el mismo responde con una determinada señal $y(t)$. Al aplicar la transformada de Fourier tanto a $x(t)$ como a $y(t)$, podemos obtener $X(f)$ e $Y(f)$ y realizar el cociente para hallar $H(f)$ como se vio anteriormente. Sin embargo, este proceso requiere del procesamiento de las señales temporales con la transformada discreta rápida de Fourier (FFT) para poder hacer el cociente.

Por esta razón la señal más conveniente para aplicar este análisis es una señal armónica sinusoidal (10). De esta manera, al excitar con un seno, de la ecuación (11) se deduce que la señal de salida también será un seno de la misma frecuencia pero con su amplitud y desfase afectados por $H(f)$.

$$x(t) = K \cdot e^{j \cdot \omega_0 t + \phi_0} \Rightarrow X(f) = K \cdot \delta(f-f_0) \cdot e^{j \cdot \phi_0} \quad (10)$$

$$Y(f) = K \cdot \delta(f-f_0) \cdot e^{j \cdot \phi_0} \cdot H(f) = K \cdot e^{j \cdot \phi_0} \cdot H(f_0) \quad (11)$$

$$Y(f) = A \cdot e^{j \cdot \phi_1} = K \cdot e^{j \cdot \phi_0} \cdot H(f_0) \quad (12)$$

$$Y(f) = K \cdot e^{j \cdot \phi_0} \cdot |H(f_0)| \cdot e^{j \phi_H} \quad (13)$$

Resultando en:

$$|H(f_0)| = \frac{A}{K} \quad (14)$$

$$\phi_{H(f)} = \phi_{1(f)} - \phi_{0(f)} \quad (15)$$

Entonces, midiendo la amplitud y el desfase existente entre $x(t)$ e $y(t)$ a una frecuencia f_0 se obtiene fácilmente el valor complejo de $H(f_0)$, sin usar la transformada de Fourier.

III. METODOLOGÍA GENERAL

El proyecto consiste en la construcción de un circuito impreso, la programación de un software bajo la supervisión de un sistema operativo *FreeRTOS*, en conjunto con un Framework para manejo de interfaces gráficas *TouchGFX* [3]; y la calibración del sistema de manera de cuantizar la incertidumbre de las mediciones. El circuito impreso consta de 2 capas y 130 vías, hecho con la técnica de planchado y acidado con percloruro férrico. Los componentes que posee la placa son resistencias y capacitores SMD 0805, y circuitos integrados DIP-8. Para la CPU se utilizó un cortex M4 ARM STM32F401RE [4] embebido en una placa de *STM NUCLEO-F401RE* [5] y para la interfaz gráfica una pantalla TFT con un controlador ILI9341 [6]. La señal senoidal de excitación es producida por un generador de señales programable *AD9833* [7], que utiliza tecnología DDS (*Direct Digital Synthesis*), la cual se basa en generar una señal digital variable en el tiempo y luego realizar una conversión digital a analógico de dicha señal.

IV. MEDICIÓN

A. Principio de Medición

Como se describió anteriormente, la medición completa consiste en poder obtener la fase y módulo de la transferencia de un filtro.

Para medir estos dos parámetros, podría realizarse un muestreo con la CPU y una posterior FFT. Sin embargo, este procedimiento para el microcontrolador utilizado resulta demasiado costoso computacionalmente, y limitante en cuanto al hardware necesario (demasiada velocidad de ADC requerida, etc.). Por esta razón se optó otro método de medición, en el que la amplitud y el desfase temporal entre las dos señales se realiza de manera directa.

Para ello, se propone para $|H(f)|$ acondicionar ambas señales transformándolas en valores de tensión continua que sean iguales al valor máximo de cada una respectivamente. Estas señales continuas son leídas por dos canales del conversor digital analógico (ADC) del MCU.

Para $\Phi_{H(f)}$ se propone detectar los cruces por cero de ambas señales de forma coherente, es decir en ambas pendientes ascendentes o ambas descendentes. En este caso, con un comparador se generan señales cuadradas a partir de las senoidales, para que sea más fácil determinar los instantes de tiempo a evaluar de forma digital. De esta manera, el MCU deberá contar el tiempo entre flancos ascendentes o descendentes de ambas señales (se decidió utilizar los ascendentes).

B. Diagrama en Bloques - Hardware

En la Fig. 2 se muestra un diagrama en bloques del dispositivo.

C. Acondicionamiento de Señales

La generación de una señal continua que representa el valor pico de ambas senoidales de entrada y salida se realiza a través de un circuito detector de pico (Fig. 3), constituido por un comparador **LM311** [8], un amplificador operacional **OPA604APG** [9], un diodo Schottky, un capacitor y un transistor mosfet canal N.

De forma breve, el funcionamiento se basa en la carga de un capacitor a través de un comparador que se activa cuando la tensión de entrada es mayor a la de salida del buffer (la cual es igual a la tensión del capacitor). Cuando dicha tensión de salida supera levemente la de entrada (VIN), el comparador se desactiva y el capacitor mantiene su tensión por un tiempo relativamente alto dado que la descarga es a través de la corriente en inversa del diodo y la de entrada del operacional.

Notar que el transistor presente se conecta en paralelo al capacitor y es controlado por una salida de la CPU. Al cambiar a otro punto de frecuencia, puede que el nuevo valor pico de salida sea menor que el medido anteriormente, por lo que el capacitor debe descargarse antes de volver a realizar una medición. Al activar el MOSFET, se cortocircuita el capacitor descargándolo rápidamente.

Por otro lado, la generación de las señales cuadradas para detectar el desfase se realiza con un comparador LM311

con un lazo de realimentación de histéresis (Fig. 4). De esta manera, ambas señales son más robustas al ruido que pueda haber en el sistema dado que impide que este genere falsos flancos a la salida de los comparadores cuando las señales senoidales se encuentran cerca del cero.

El valor pico de la señal senoidal de excitación generada por el AD9833 es de apenas unas centenas de mV y además con un nivel de continua indeseado para nuestro análisis. Para aprovechar mejor el rango dinámico del ADC del microcontrolador, se filtra el nivel de continua con un filtro pasa-altos, y se amplifica la señal a una tensión pico de 3.3V (tensión de referencia del ADC) con un amplificador no inversor como el de la (Fig. 5). El propósito del jumper J5 y R4 será explicado con mas detalle en el apartado de calibración (Sección VIII).

D. Método de Medición

El método de medición se subdivide en 3 etapas:

- 1) Selección de rango de medición desde la UI.
- 2) Medición de amplitud.
- 3) Medición de fase.

- 1) El usuario controlará desde la UI el rango de frecuencias que desea medir y la cantidad de puntos por década que desee. Luego con el botón "Medir" comenzará la medición, donde el MCU controlará todo el procedimiento.
- 2) Con los rango obtenidos de la UI, el MCU comanda al generador de señales para que genere una señal senoidal con la primer frecuencia del barrido. Se descargan los capacitores de los detectores de pico durante 1ms, y luego de un tiempo de espera para estabilizar las señales el ADC del MCU genera una conversión de 10 muestras para ambos canales (señal de entrada y salida). Estas son promediadas; con el valor obtenido se realiza el cociente de Vout y Vin para obtener $|H(f_0)|$.
- 3) Una vez realizada la medición de amplitud, el MCU comenzará la medición de fase si el módulo medido fue mayor a -30dB, y si la frecuencia es menor a 20kHz, explicado a continuación en la sección de *Rango de Medición*. Un timer dedicado del microcontrolador detectará un flanco ascendente de la señal de entrada, y cuando ocurra esto, almacenará el valor actual de la cuenta del mencionado timer. Al ocurrir luego un flanco ascendente pero esta vez de la señal de salida, nuevamente se almacena el valor de la cuenta. Con los valores obtenidos se calcula el desfase temporal, y posteriormente el desfase en grados sexagesimales.

Terminada la medición, el MCU cambia a la siguiente frecuencia (siguiente punto de medición) y se repite el procedimiento. Una vez terminado el barrido de frecuencia, en la UI se actualizan los gráficos de módulo y fase.

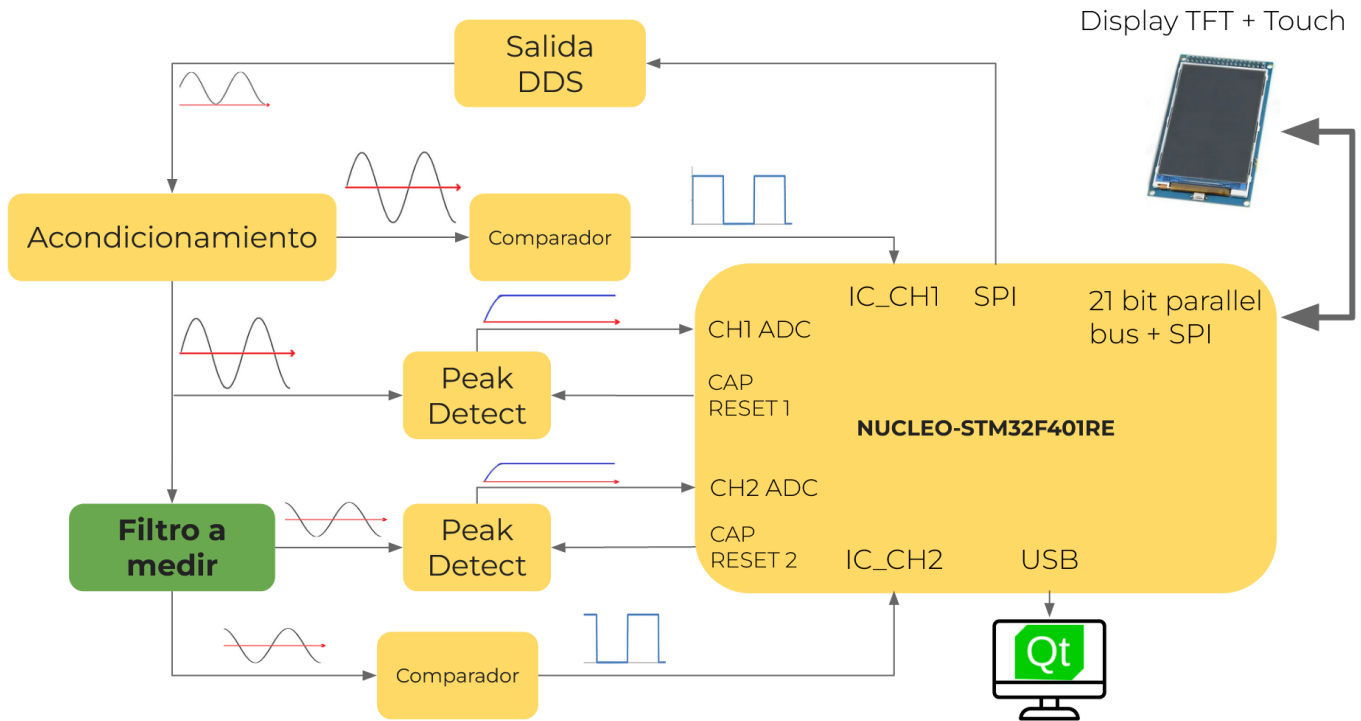


Fig. 2. Diagrama en bloques del hardware

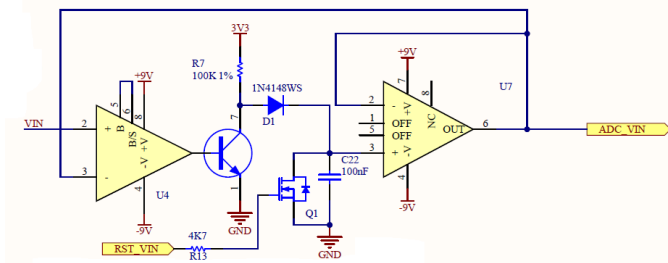


Fig. 3. Circuito detector de pico

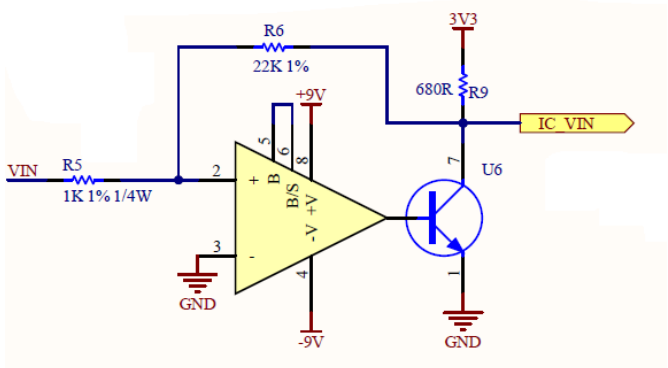


Fig. 4. Circuito comparador con histéresis

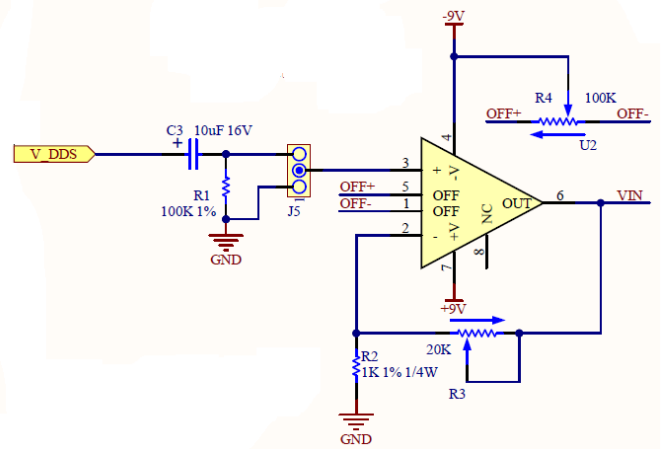


Fig. 5. Acondicionamiento de señal generada: filtro pasa altos y amplificador no inversor

E. Rango de Medición

El rango de medición queda determinado por las características del filtro, y las limitaciones de hardware analógico y digital.

1) *Magnitud*: El límite superior es siempre de 0dB para toda frecuencia dado que se excita al filtro con una señal que tiene el pico igual a la tensión de referencia del ADC (de manera de aprovechar su rango dinámico), si el filtro es activo o tiene un realce eso no se podría medir teniendo en cuenta que saturan las cuentas del conversor. Para frecuencias menores a 20Hz el rango en la medición de magnitud queda

limitado entre 0dB y -16dB debido a que el *ripple* por la descarga del capacitor del detector de pico a baja frecuencia es considerable a amplitudes bajas (menores a 200mV). Para frecuencias mayores a 20Hz y hasta 20kHz el rango está limitado entre 0dB y -44dB dado que el detector de pico conectado a la salida del filtro puede disminuir su valor sólo hasta 20mV.

2) *Fase*: La medición de fase está limitada por la histéresis y el umbral propio que tiene el comparador para discernir en un cruce por cero, y además por las limitaciones propias del timer de la CPU.

No es posible medir desfases de señales menores a 35mV debido a que están por debajo de los umbrales de histéresis de los comparadores. Por esta razón solo se mide fase si el módulo medido con anterioridad es mayor a -30dB (valor elegido con cierto margen de seguridad). Por otro lado, también hay limitaciones para el desfase a frecuencias altas (más de 25kHz) debido a que los canales de input capture del timer del microcontrolador dejan de responder de forma rápido a los flancos ascendentes producidos por el comparador.

Por estas razones, la medición de fase está limitada a un rango entre 1Hz y 20KHz, cuando la magnitud es mayor a -30dB. Teniendo en cuenta estas restricciones, los valores de fase no tienen limitaciones en su medición: se muestran en grados sexagesimales desde -180° hasta 180°.

V. FIRMWARE

A. Principio de funcionamiento

El algoritmo está compuesto de un total de 6 tareas: inicialización del hardware "*StartHardwareTask*"; monitoreo de medición "*MeasureTask*"; medición de amplitud "*ModuleTask*"; medición de fase "*PhaseTask*"; envío de datos por USB "*USBTask*"; refresco de pantalla que corre dentro del motor gráfico, ejecutado periódicamente a través de un timer.

- 1) "*StartHardwareTask*" se encarga de inicializar el motor gráfico de TouchGFX y establecer la comunicación USB.
- 2) "*MeasureTask*" se encarga descargar los capacitores de los detectores de pico durante 1ms, recalculer y cambiar la frecuencia de la señal de entrada, generar una demora de 1s para que se estabilicen las señales, iniciar la medición de amplitud con "*ModuleTask*" y, si la frecuencia y la magnitud medida están dentro del rango permitido, procede a realizar la medición de fase con "*PhaseTask*". Finalizadas las mediciones, se comunica con el motor gráfico para actualizar las pantallas con los valores de módulo y fase, y con *USBTask* para enviar los datos por USB, para mostrar los gráficos en la PC.
- 3) "*ModuleTask*" sincronizado con "*MeasureTask*", genera 10 conversiones del ADC separadas $54.67\mu s$ entre sí de ambas señales pico de entrada y salida. Luego de la adquisición se promedia y se calcula la magnitud en decibels.
- 4) "*PhaseTask*" sincronizado con "*MeasureTask*", se activan las entradas que detectan flancos, y una vez suce-

didos dos eventos de flancos ascendentes como se describió anteriormente, se calcula la diferencia temporal y posteriormente el desfase en grados sexagesimales. Obtenido este valor se analizan los casos en que la fase supera 360° para descartar una mala detección (posible a alta frecuencia), y el caso en que la fase supere -180°. Para el primero genera una nueva medición, y para el segundo se realiza un wrap de la fase para llevarla al primer cuadrante. Para este procedimiento, también se toman 10 muestras de la fase.

- 5) "*USBTask*" sincronizado con "*MeasureTask*", envía por comunicación USB la medición de magnitud, fase y los valores de frecuencia.

En la siguiente página se muestra un diagrama de flujo del algoritmo completo (Fig. 7).

VI. APLICACIÓN EN TOUCHGFX

A. Pantallas

La aplicación está compuesta de 2 *Pantallas*. La primera consiste en el menú de configuración (Fig. 6) en donde el usuario puede a través de tres sliders configurar:

- 1) Frecuencia máxima de medición. El valor límite de este parámetro es 1MHz
- 2) Frecuencia mínima de medición. El valor límite de este parámetro es 1Hz
- 3) Puntos por década. El valor mínimo es 1 y el valor máximo 600.

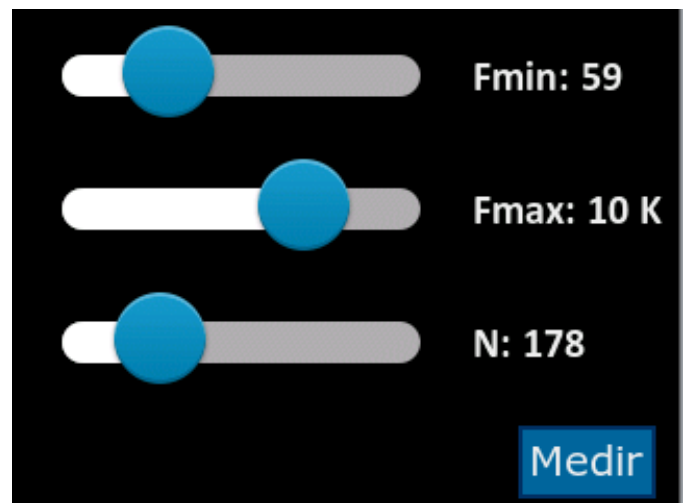


Fig. 6. Pantalla 1 de configuración de frecuencia

La segunda es donde se muestra la magnitud y fase, en un gráfico con el eje horizontal graduado en una escala logarítmica para la frecuencia, y el eje vertical en una escala en decibels para la magnitud (Fig. 8) o en grados sexagesimales al mostrar la fase (Fig. 9). Así mismo, en la esquina superior derecha de la pantalla hay un botón para cambiar entre el gráfico de la magnitud y la fase, y en la esquina superior izquierda, un botón para volver al menú de configuración de frecuencia.

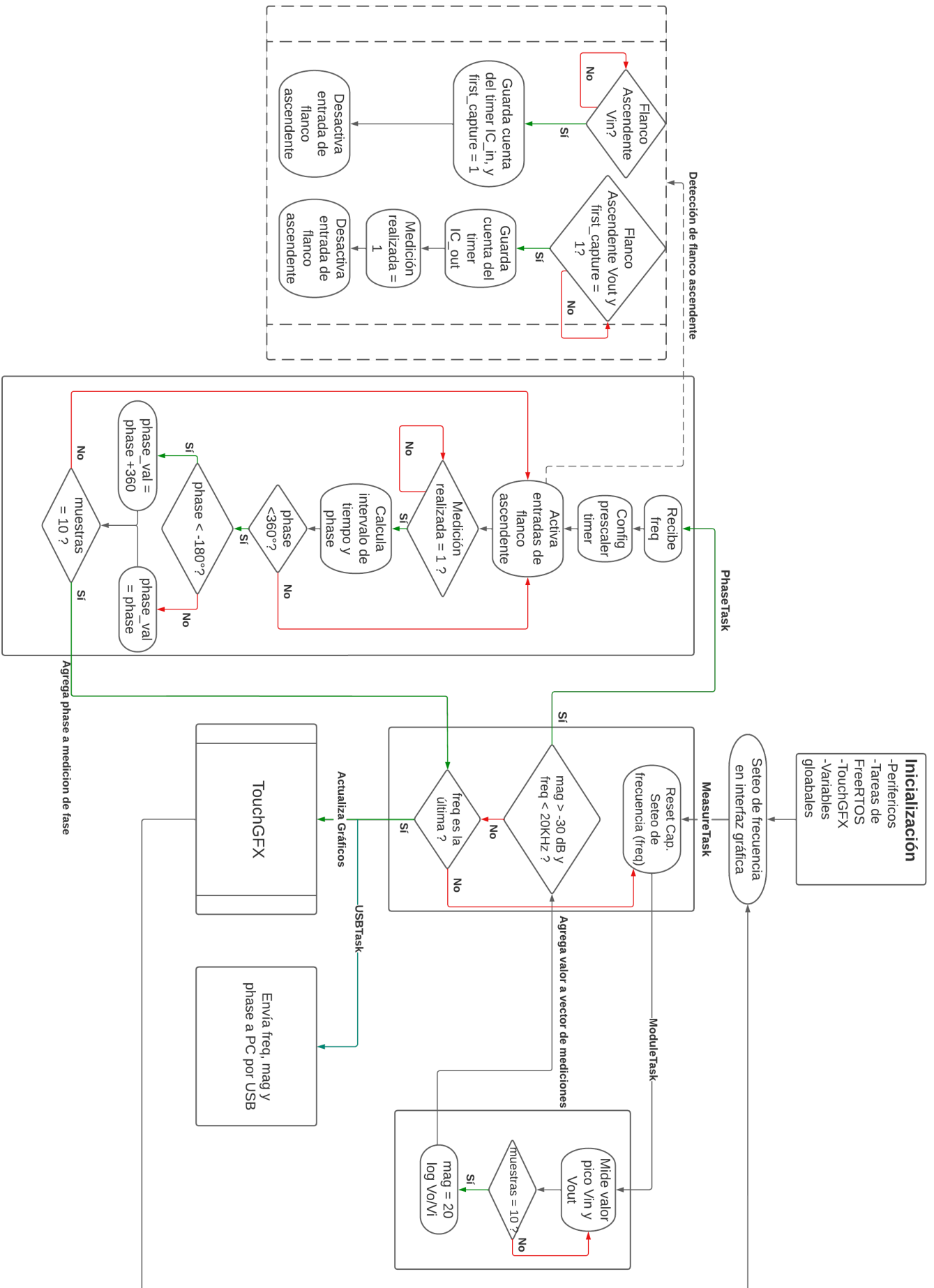


Fig. 7. Diagrama de flujo del firmware

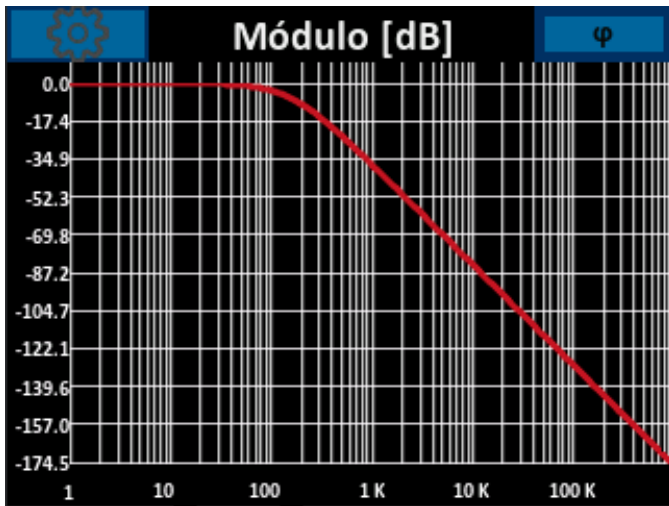


Fig. 8. Pantalla 2: gráfico de módulo

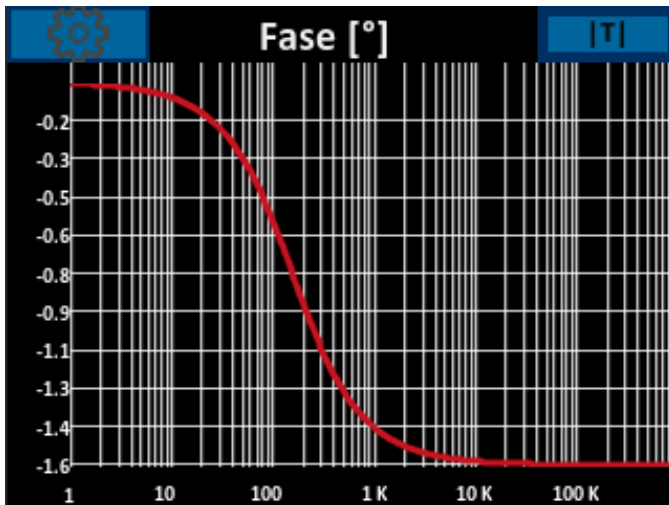


Fig. 9. Pantalla 2: gráfico de fase

B. Lógica de funcionamiento

A través de los sliders el usuario puede configurar el intervalo de frecuencia de medición en conjunto con los puntos por década. Al presionar el botón "Medir", se calcula el valor máximo y mínimo de frecuencia que se va a graficar en pantalla. La escala horizontal siempre mostrará potencias de 10 como décadas, por ejemplo, si se selecciona una frecuencia mínima de 3Hz y máxima de 500Hz, la medición será desde 3Hz hasta 500Hz pero el gráfico se mostrará desde 1Hz hasta 1KHz.

Luego el framework se comunica con el back-end del programa para calcular las frecuencias intermedias entre la máxima y la mínima en una escala logarítmica, separado un valor del otro geométricamente equidistante. Calculados todos los valores, el back-end comienza la medición.

Finalizada la adquisición de datos, el back-end se comunica con el framework para actualizar los gráficos.

VII. ANÁLISIS DE DISTORSIÓN ARMÓNICA DDS

Como se mencionó anteriormente, el DDS, junto con el acondicionamiento realizado, es utilizado para generar una señal armónica que es requerida como se aclara en la teoría. Sin embargo, en la práctica no ocurre que la señal sea puramente armónica, debido a distintos factores como la resolución del conversor digital analógico del DDS o, en caso de no ser correctamente elegido, el slew rate del operacional en la etapa de acondicionamiento. Esto provoca una distorsión respecto de la señal senoidal deseada. Una forma de medir qué tanto se distorsiona es analizándola en el dominio de la frecuencia. Si no hubiese distorsión (caso ideal) habría una sola componente. En cambio, al haber distorsión aparecen otras componentes armónicas en el espectro.

Estas distorsiones pueden provocar un error en la medición si lo suficientemente grandes, por ejemplo en la detección del pico. Por lo tanto, es necesario caracterizar este efecto para saber si el DDS seleccionado es aceptable para los requerimientos tecnológicos del sistema. Existen dos criterios para garantizar que la distorsión del generador sea despreciable. Como primera instancia, se compara el pico más alto del armónico indeseado con medio cuanto del ADC del microcontrolador, para evaluar si el armónico afecta o no al código obtenido del conversor. Si el valor de tensión de entrada es cercano a una transición del ADC, entonces puede que el armónico afecte al código, pero esto mismo sucede con el ruido. Por esta razón, como segundo criterio, se compara el valor eficaz de la armónica indeseada con el valor eficaz del ruido. Si la distorsión comienza a ser considerablemente mayor al piso de ruido, entonces además de una variación aleatoria en las cuentas existe una variación en las cuentas por la distorsión armónica del generador cuando la tensión de entrada esta cercana a las transiciones del ADC.

El ensayo consiste en obtener 20 curvas de tensión en función del tiempo de distintas frecuencias desde 1Hz hasta 20KHz (eligiendo valores separados exponencialmente), que es el rango normal de trabajo del equipo. Luego de esto se realiza una FFT a cada curva para detectar posibles armónicas no deseadas, aplicando la ventana flat-top para medir correctamente los valores pico y la hanning para distinguir correctamente los picos si están muy próximos entre sí en frecuencia (Fig. 10 y Fig. 11).

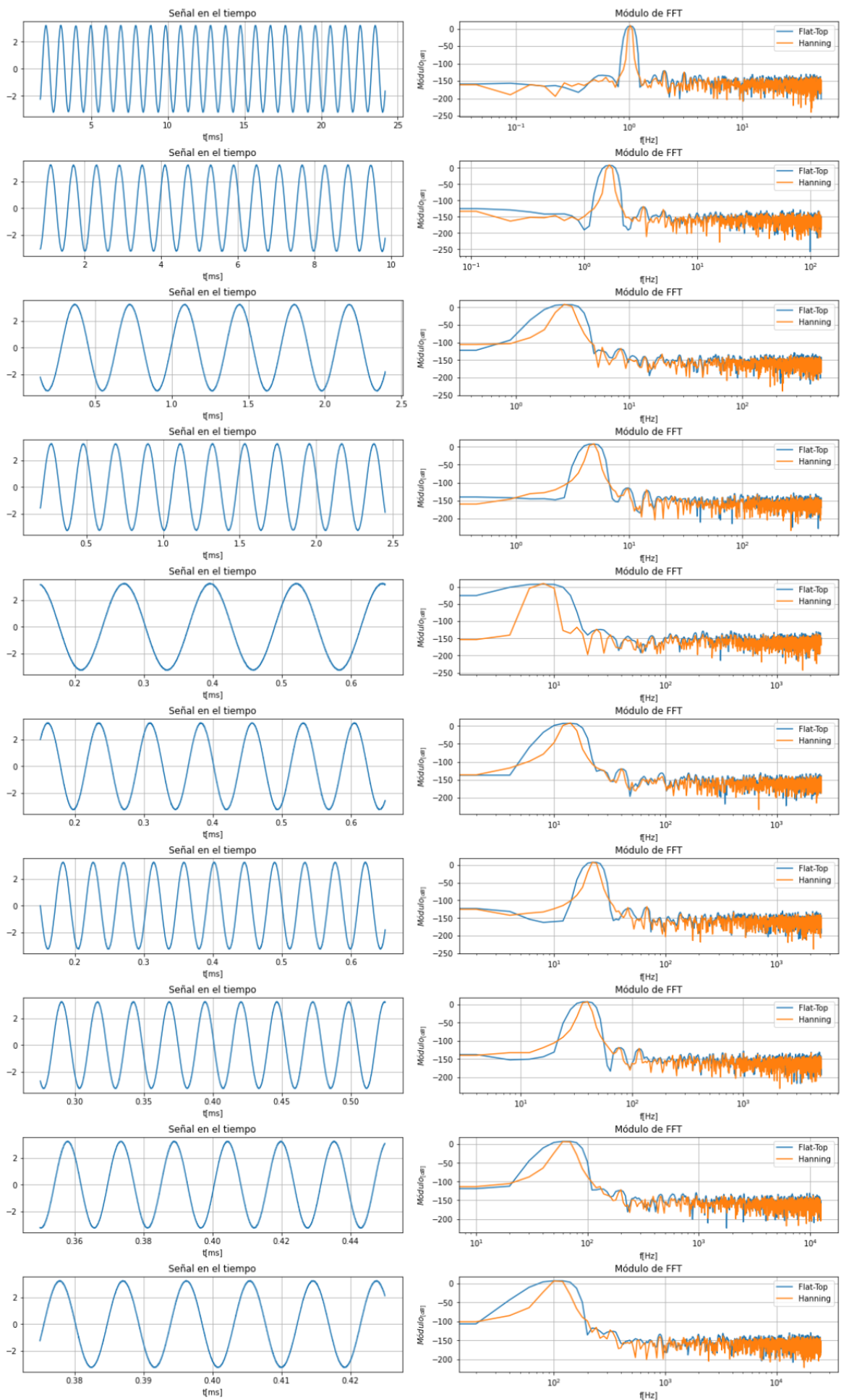


Fig. 10. Ensayo de THD de 1Hz a 108.976Hz

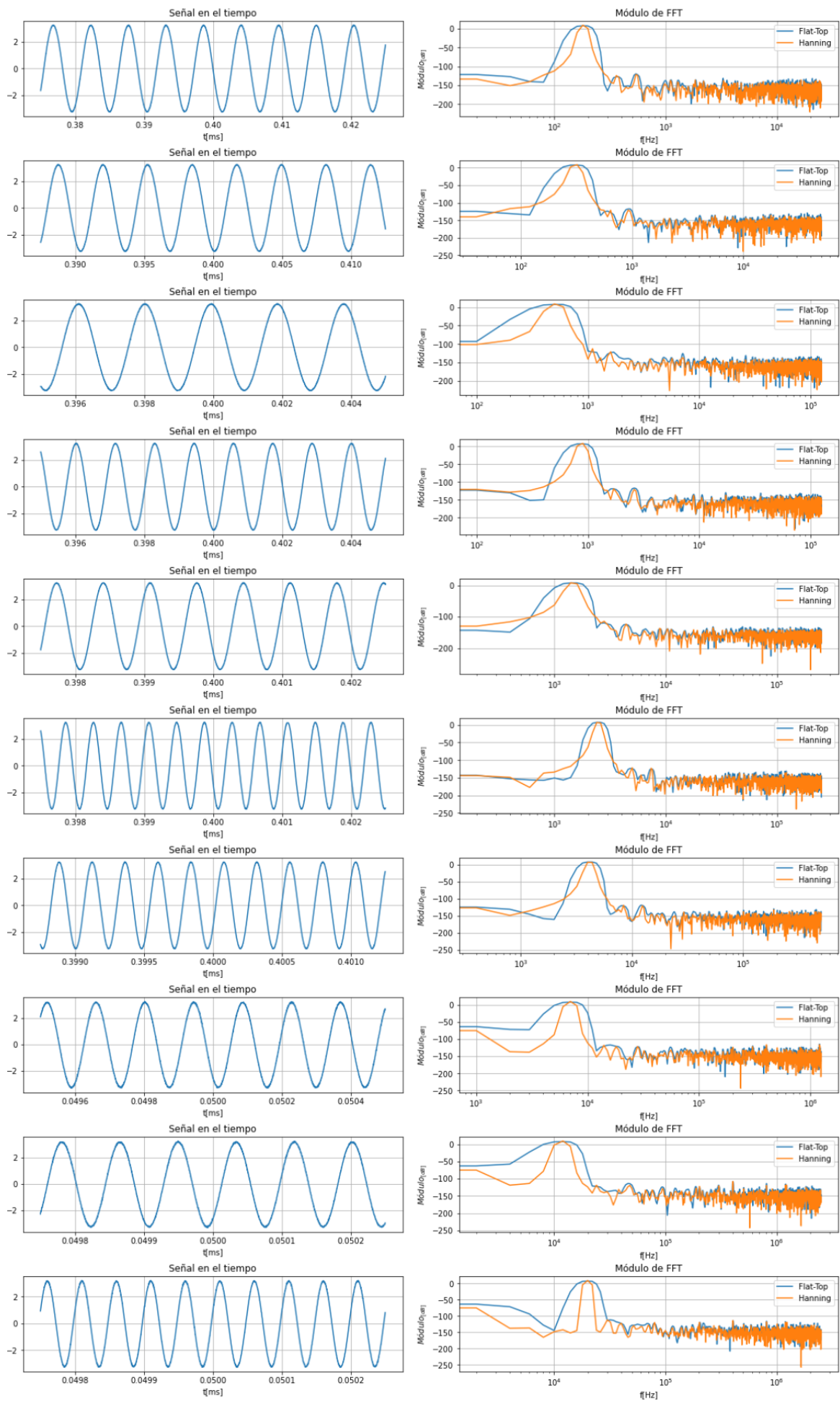


Fig. 11. Ensayo de THD de 183.527Hz a 20000Hz

A. Análisis de Resultados

Como se observa en la Fig.10 y 11, los posibles picos que podrían deberse a la distorsión armónica, se encuentran siempre a menos de -100dBV. Por lo tanto aplicando el primer criterio mencionado, se compara medio cuanto q (16) con el valor pico de -100dBV multiplicado por dos, debido a que si bien se muestra el espectro positivo, la FFT distribuye la potencia de la señal en espectro positivo y negativo (V_x (17)).

$$\frac{q}{2} = \frac{V_{ref}}{2^N \cdot 2} = \frac{3.3V}{4096 \cdot 2} = 402.8\mu V \quad (16)$$

$$V_x = 2 \cdot 10^{\frac{-100dB}{20}} = 20\mu V \quad (17)$$

Como se puede observar el valor pico de las armónicas que distorsionan no llegan a medio cuanto del ADC, por lo que no produce un error grosero para que el ADC resuelva entre dos cuentas adyacentes.

Por otra parte, tampoco se observa que dichas armónicas superen en gran medida al piso de ruido. Entonces, estas no producen más error que el ruido cuando la tensión de entrada está próxima a una transición del ADC.

VIII. CALIBRACIÓN

En este apartado se explica como se realiza la calibración del conversor analógico digital del MCU utilizado y a su vez cómo utilizar los resultados obtenidos para corregir una medición de cuentas. También se explica la calibración de un intervalo de tiempo medido con canales de input capture del MCU; al igual que para la calibración del ADC, se desarrolla el procedimiento para utilizar los resultados obtenidos en la del intervalo de tiempo.

A. Calibración de ADC

Se obtiene la transferencia del ADC, y se almacena en la memoria FLASH del microcontrolador. Luego se realiza una interpolación lineal para cada cuenta obtenida del ADC, de manera de obtener la tensión de entrada aproximada que la generó.

La calibración consta de un generador de señales arbitrario Rigol DG5070 [10] controlado con una computadora mediante USB, que genera una rampa escalonada de 1024 niveles de tensión de 0V a 3.3V mantenidas 4ms cada una. El microcontrolador realiza 3 lecturas separadas 1ms entre sí comandado por la misma PC que controla al generador, y devuelve el resultado de la medición a la PC.

Se promedian los valores de tensión correspondientes a un mismo código, y luego se hace un ajuste lineal de la transferencia (Fig. 12). Con los valores de la transferencia ajustada a una recta se genera una matriz de 1024 filas (puntos de medición), y dos columnas siendo la primera de cuentas y la segunda de la tensión de entrada en función de la cuenta obtenida.

Teniendo esta matriz en la flash del microcontrolador (const float transf_ADC[PUNTOS]), cada vez que se obtenga una

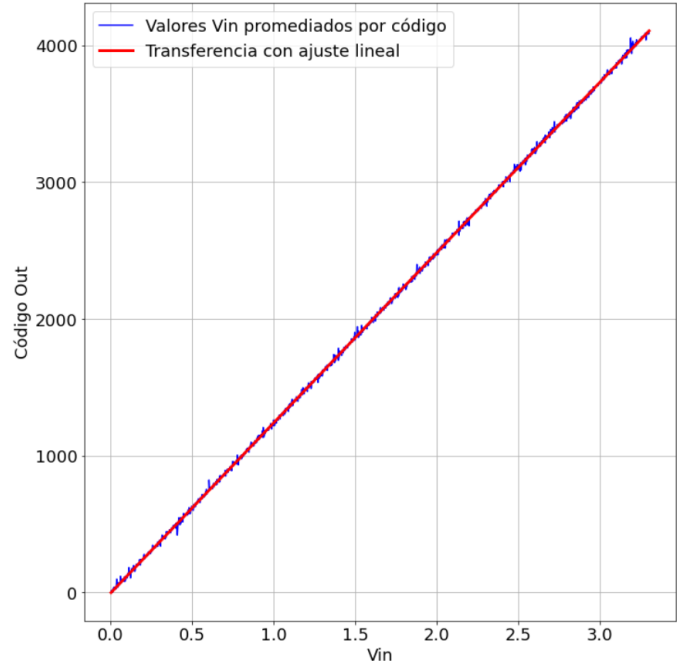


Fig. 12. Transferencia de ADC

cuenta (C) con el conversor se debe encontrar la cuenta de la matriz que sea inmediatamente menor a la cuenta obtenida (C_{INF}), la cual tendrá su respectiva tensión de entrada (V_{INF}), y la cuenta que sea inmediatamente mayor (C_{SUP}) también con su respectiva tensión analógica (V_{SUP}). Con estos valores se realiza el siguiente cálculo:

$$V_{IN} = \frac{V_{INF}(C_{SUP} - C) + V_{SUP}(C - C_{INF})}{C_{SUP} - C_{INF}} \quad (18)$$

El cual se deduce de averiguar la ecuación de la recta que pasa por los puntos (C_{INF}, V_{INF}) y (C_{SUP}, V_{SUP}). En otras palabras, una interpolación lineal.

B. Calibración de timer con Input Capture

Se utiliza como patrón un frecuencímetro Protek U2000A [11] utilizado como medidor de intervalo de tiempo promediado por 100 mediciones. Con el generador de señales Agilent 33500 [12] se generan dos señales cuadradas a modo de evitar error de trigger del frecuencímetro, y se mide el desfase temporal de las mismas. La medición se realiza desde 1us hasta 1s con el microcontrolador y con el frecuencímetro. En este ensayo también se obtiene una matriz de dos columnas (valor del timer y valor de patrón) y 25 filas (puntos de medición). Las mediciones se encuentran separadas exponencialmente debido a que es más probable que se produzcan intervalos de tiempo de microsegundos hasta las decenas milisegundos, que a del orden de las décimas segundo en adelante (dado que esto solo ocurre con valores de fase grandes en baja frecuencia) (Fig. 13). Luego esta matriz es almacenada en la FLASH y utilizada por el MCU en cada medición de tiempo, para corregir el resultado utilizando una interpolación lineal idéntica a la utilizada para calibrar el ADC.

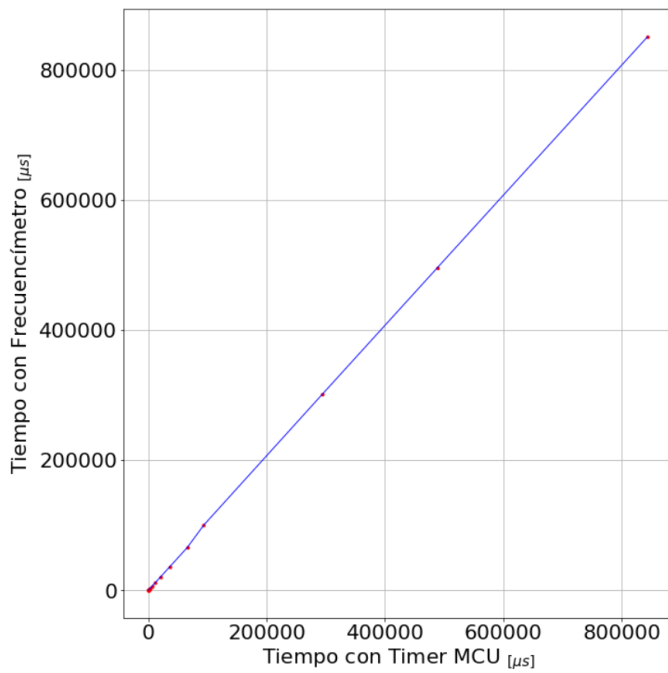


Fig. 13. Calibración de medición de intervalo de tiempo

IX. INCERTIDUMBRE

Se tiene incertidumbre tipo A en los valores de tensión medidos con el ADC, y en los valores de intervalo de tiempo medidos con el timer, dado que se toman 10 muestras de la tensión de entrada al filtro, 10 muestras en la tensión de salida y 10 muestras de intervalo de tiempo respectivamente.

Estas muestras son almacenadas por el MCU, y en cada punto de medición de frecuencia se realiza una transferencia por USB con una aplicación que se ejecuta en una computadora aparte.

La aplicación calcula los valores medios y las incertidumbres TIPO A para cada punto de frecuencia de: la tensión de entrada al filtro, la tensión de salida y el intervalo de tiempo.

Con los datos aportados por los manuales del generador arbitrario utilizado en la calibración del ADC y el manual del frecuencímetro utilizado en la calibración del input capture, se calcula la incertidumbre tipo B de las mismas magnitudes mencionadas en el párrafo anterior, y con ellas las combinadas.

Para obtener las incertidumbres combinadas del módulo y de la fase, se aplica la teoría de propagación de incertidumbre, calculando las derivadas parciales que correspondan.

Luego de varias pruebas se determina que tanto las 10 mediciones de las cuentas devueltas por el timer del input capture como las 10 mediciones de las cuentas entregadas por el ADC, resultan precisas, es decir, que prácticamente no hay incertidumbre en el estudio TIPO A, por lo que se considera que la mayor parte del error proviene del estudio TIPO B. Entonces, todas las incertidumbres combinadas poseen

distribución uniforme y se expanden al 95% con $k = \sqrt{3}$.

La incertidumbre en cada magnitud medida se expresa de forma gráfica con una banda del intervalo de confianza sobre el mismo gráfico que muestra magnitud y fase por separado.

En las figuras 14, 15, 16, 17 se muestran como ejemplo las mediciones de un filtro pasa-bajos RC y un filtro Notch pasivo.

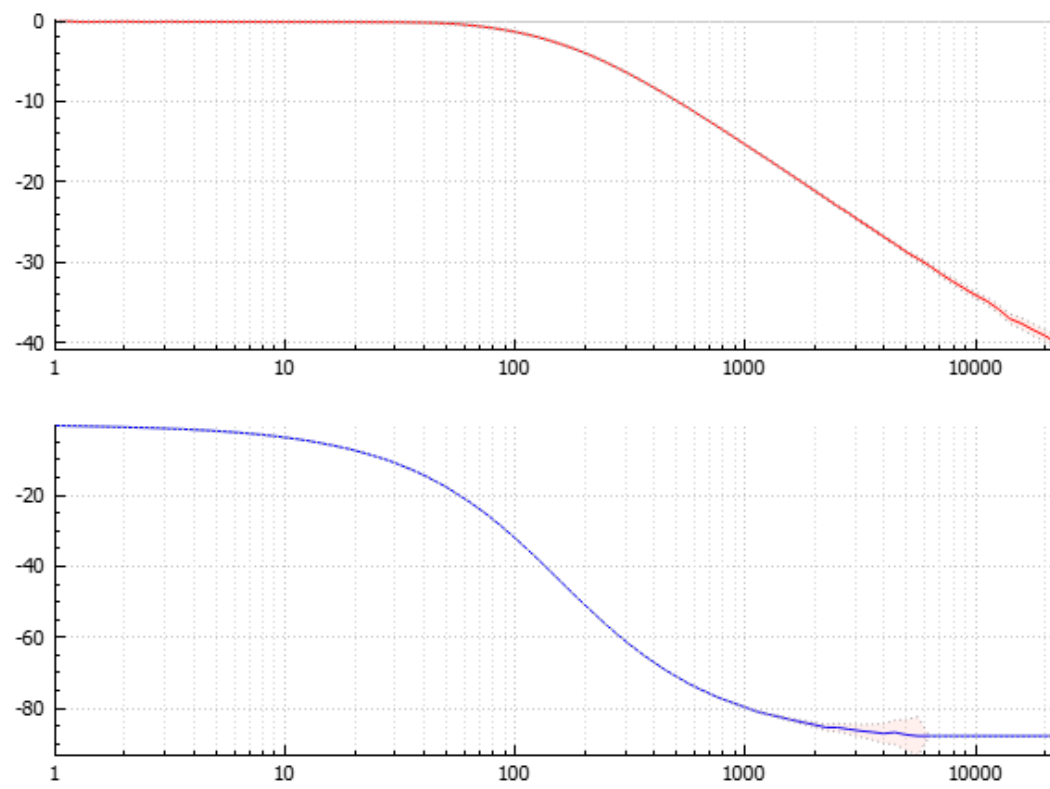


Fig. 14. Filtro pasa-bajos RC

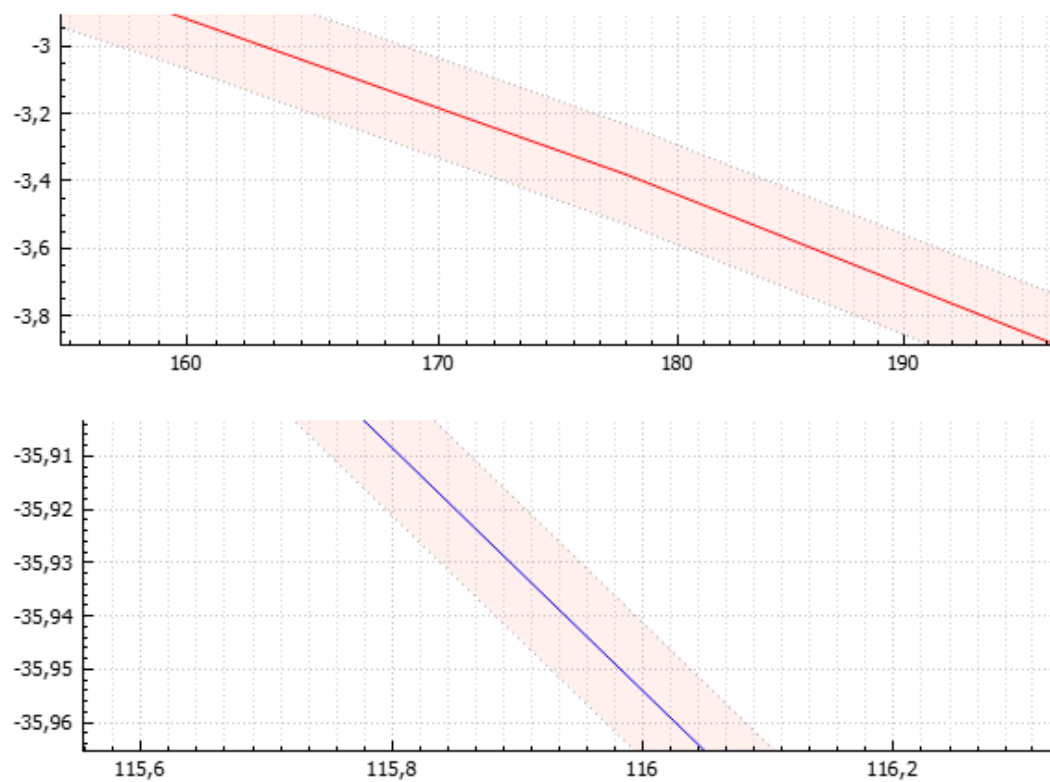


Fig. 15. Ampliación de gráficos Filtro pasa-bajos RC

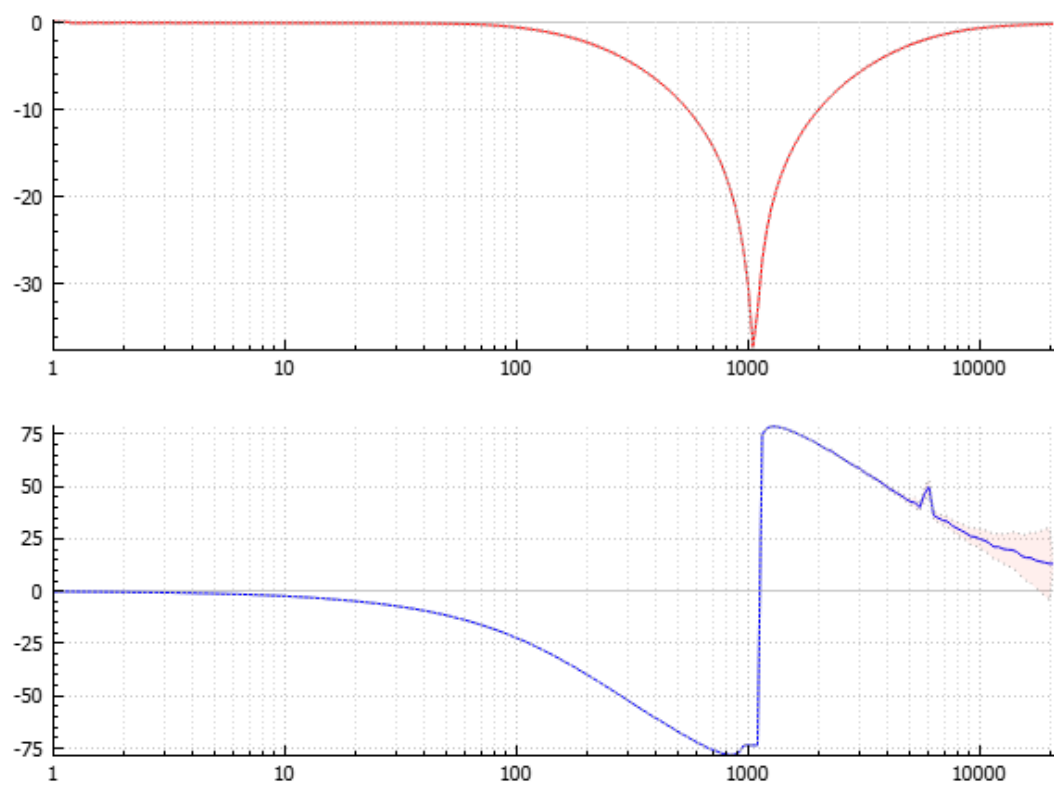


Fig. 16. Filtro Notch pasivo

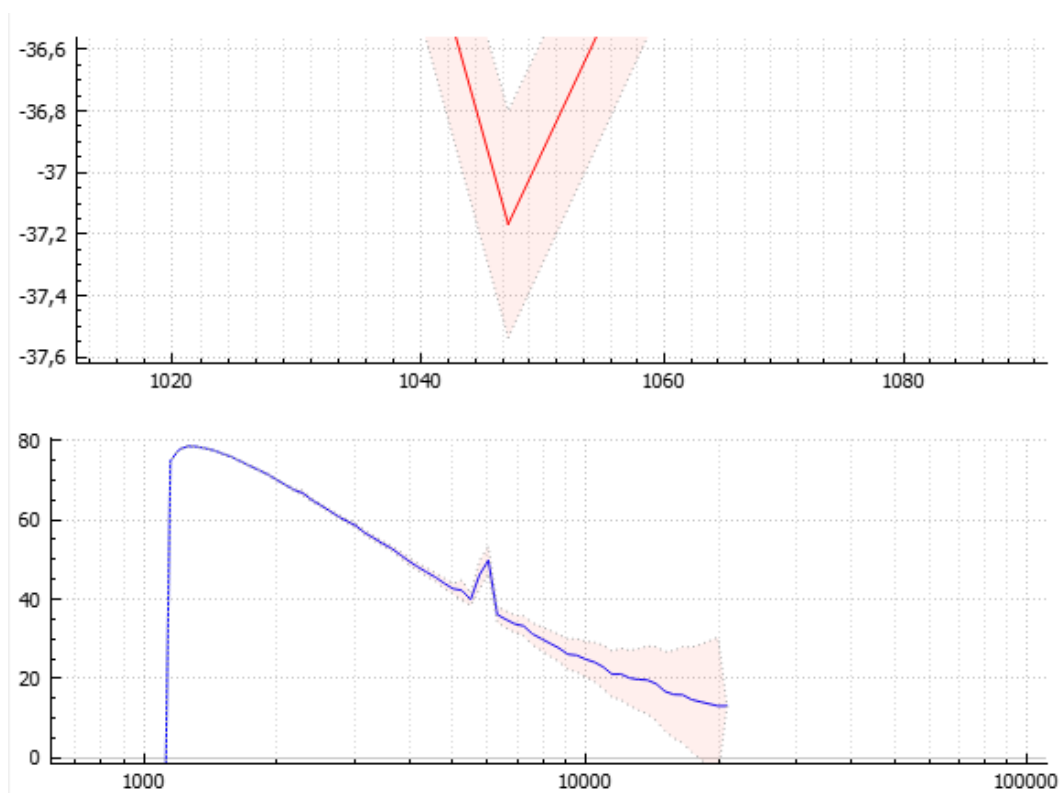


Fig. 17. Ampliación de gráficos filtro Notch pasivo

X. CONCLUSIONES

A. Resultados

Los resultados obtenidos del cálculo de incertidumbre reflejan que el dispositivo es suficientemente preciso y que no hay casi dispersiones en las curvas medidas. Sin embargo, a frecuencias más altas cercanas al límite del dispositivo, se puede observar como aumenta la incertidumbre en la fase debido a que la calibración tiene una limitación en la resolución del intervalo de tiempo, la cual es de 1ns. El estudio tipo A hecho mostró que la incertidumbre es casi despreciable en comparación a la que introduce el frecuencímetro utilizado a esas frecuencias.

Para la incertidumbre que muestra el módulo, ocurre algo similar a la fase en cuanto a que aumenta la incertidumbre pero cuando la señal a medir es pequeña. En este caso la incertidumbre la aporta casi toda la tipo A debido a que el dispositivo intenta medir señales que están cercanas al piso de ruido del sistema, por lo que aumenta la variación en las cuentas que arroja el ADC del MCU debido a este efecto; por lo tanto aumenta la incertidumbre tipo A.

B. Limitaciones

El ancho de banda resultó ser mucho menor al esperado antes de realizar el proyecto. Por un lado porque para la medición de módulo se optó por un circuito detector de pico con bajo nivel de ruido y que mida tensiones bajas en el orden de los milivolts a otro con buena respuesta en frecuencia pero con mayor nivel de ruido y por ende menor precisión. Por otro lado, al medir la fase, si bien los comparadores funcionan perfectamente hasta 1MHz, el input capture deja de detectar los flancos de las señales cuadradas una vez superados aproximadamente los 25kHz (por eso el límite a 20kHz dejando un margen de seguridad).

Otro problema, mencionado en la sección de rangos, tiene que ver con la medición de amplitud a baja frecuencia y baja amplitud debido que el ripple de descarga de los capacitores de los detectores de pico se vuelve considerable.

La medición se torna ciertamente lenta, ya que le toma 1 segundo a los capacitores de los detectores de pico para descargarse y cargarse nuevamente a la tensión correspondiente, por lo que para una medición con mucha resolución en frecuencia (muchos puntos), puede tardar hasta casi 10 minutos en medir.

C. Posibles Mejoras

Una forma de mejorar sustancialmente el sistema es utilizando un microcontrolador con un ADC más rápido para poder muestrear la señal, y con mayor frecuencia de clock para poder realizar un algoritmo de Goertzel [13], el cual se puede tomar como una transformada de rápida Fourier pero que sólo devuelve un número complejo correspondiente al bin de una frecuencia específica, la cual sería configurada para que sea la frecuencia que generamos con el DDS. De esta manera se puede reducir el hardware al máximo, retirando los detectores de pico y los comparadores, y no depender del input capture del timer interno del microcontrolador. Esto tiene como consecuencia una mejora en la exactitud de la medición

(al evitar errores y limitaciones del hardware adicional), en el ancho de banda, ya que solo depende de la velocidad del ADC y de procesamiento del microcontrolador, y en el tiempo de medición dado que no hay que esperar a que se descarguen y se carguen los capacitores de los detectores de pico.

Una mejora a este último sistema mencionado sería realizar un barrido en frecuencia casi instantáneo componiendo un espectro de agujas, centradas en las frecuencias de interés, y luego obtener una señal asociada este espectro mediante una transformada rápida de Fourier inversa. Esta sería la señal que excitaría al filtro a analizar. Luego a la señal filtrada se le aplica una transformada rápida de Fourier para analizar de forma simultánea como se modificaron las agujas en el espectro de salida.

REFERENCES

- [1] REDPITAYA, "Bode analyzer," Disponible en <https://redpitaya.com/applications-measurement-tool/bode-analyzer/>.
- [2] O. LAB, "Bode 100," Disponible en <https://www.omicron-lab.com/products/vector-network-analysis/bode-100#>.
- [3] STM, "Touchgxf documentation," Disponible en <https://support.touchgxf.com/4.21/docs/introduction/welcome>.
- [4] —, "Stm32f401xb/c and stm32f401xd/e advanced arm®-based 32-bit mcus," Disponible en https://www.st.com/resource/en/reference_manual/rm0368-stm32f401xb-c-and-stm32f401xd-e-advanced-armbased-32bit-mcus-stmicroelectronics.pdf (19/12/18).
- [5] —, "Stm32 nucleo-64 boards (mb1136)," Disponible en https://www.st.com/resource/en/user_manual/um1724-stm32-nucleo64-boards-mb1136-stmicroelectronics.pdf (20/08/20).
- [6] ILITEK, "a-si tft lcd single chip driver 240rgbx320 resolution and 262k color," Disponible en <https://cdn-shop.adafruit.com/datasheets/ILI9341.pdf>.
- [7] A. Devices, "Low power, 12.65 mw, 2.3 v to 5.5 v, programmable waveform generator," Disponible en <https://www.analog.com/media/en/technical-documentation/data-sheets/ad9833.pdf> (04/19).
- [8] T. Instruments, "Lm111, lm211, lm311 differential comparators," Disponible en <https://www.ti.com/lit/ds/symlink/lm311.pdf> (03/17).
- [9] —, "Fet-input, low distortion operational amplifier," Disponible en <https://www.ti.com/general/docs/suppproductinfo.tsp?distId=26&gotoUrl=https://www.ti.com/lit/gpn/opa604> (09/03).
- [10] RIGOL, "Arbitrary waveform generator specifications," Disponible en <https://beyondmeasure.rigoltech.com/acton/attachment/1579/f-0051/0/-/-/-/file.pdf>.
- [11] PROTEK, "Frequency counter specifications," Disponible en <http://www.arivigevano.net/files/pk9100.pdf>.
- [12] AGILENT, "Waveform generator specifications," Disponible en <https://electricayelectronica.uniandes.edu.co/sites/default/files/laboratorios/manual-33500-Op-and-Svc-US.pdf>.
- [13] WIKIPEDIA, "Goertzel algorithm," Disponible en https://en.wikipedia.org/wiki/Goertzel_algorithm (24/01/2023).