

## 1. Introducción

La finalidad de este proyecto es analizar los aspectos teóricos y prácticos que conllevan la resolución del problema de las mediciones armónicas en sistemas de calidad de energía, para su uso en la industria con fines normativos y para maximizar el ahorro de energía, usando técnicas de procesamiento digital de señales en tiempo real.

Como primer paso, se hará un resumen de los fundamentos matemáticos del análisis espectral, partiendo de las herramientas clásicas para señales analógicas (Transformada de Fourier). Dado que estos tipos de sistemas requieren el paso al mundo digital nos vemos ante la necesidad de discretizar la señal, en tiempo y amplitud. Es así que debemos analizar los fundamentos del análisis de señales de dominio discreto (Transformada Discreta de Fourier). Dado que todo sistema real no posee una capacidad de almacenamiento infinita, se analizará la influencia de la limitación del número de muestras a procesar (Ventaneo) y sus consecuencias.

Luego, se hará un repaso de los parámetros eléctricos que sirven para indicar la calidad de energía de un sistema y los métodos que pueden usarse para obtener tales valores en un sistema de tiempo discreto digital.

La implementación del equipo estará basada en un Procesador Digital de Señales (DSP, por sus siglas en inglés) de la serie Blackfin, de Analog Devices. Completan el sistema los bloques de memoria, fuente de alimentación, conexión al puerto serie, adaptación de los niveles de señal y los conversores Analógico – Digital (ADC) encargados de la discretización de la señal. Se usará el kit de desarrollo para el DSP BF537, que incluye todos los subsistemas mencionados.

Se discutirán las técnicas de programación necesarias para la implementación del software del sistema, teniendo en cuenta los tiempos de ejecución así como los recursos necesarios para cada algoritmo (espacio en memoria, interrupciones, etc.).

Por último, se analizará la performance del prototipo tanto en la precisión de las mediciones como en la eficiencia del código implementado

## 2. Aspectos teóricos del análisis armónico

La señal de entrada es una función periódica, real y continua que cumple las condiciones de Dirichlet (1), es decir:

1 – Es absolutamente integrable sobre cualquier período

$$\int_{T_0} f(t) dt < \infty$$

2 – Tiene un número finito de máximos y mínimos en un periodo

3 – Tiene un número finito de discontinuidades finitas en un periodo (de hecho, no tiene ninguna discontinuidad)

Por lo tanto, posee un desarrollo en el dominio de la frecuencia dado por su transformada de Fourier (2):

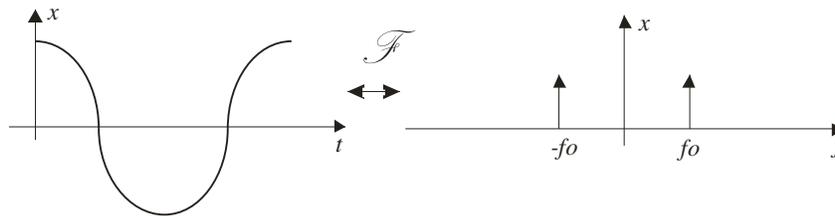
$$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt$$

Esta representación en el dominio de la frecuencia es denominada espectro, y en el caso más general, es una función compleja. Dado que la señal es periódica, su espectro es discontinuo. Más precisamente, será una sumatoria de deltas de Dirac (3):

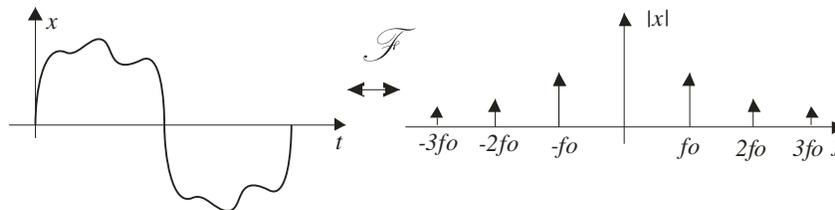
$$X(\omega) = 2\pi \sum_{n=0}^{\infty} C_n \delta(\omega - n\omega_0)$$

El análisis de Fourier nos lleva a conclusiones muy útiles para analizar una señal. Por ejemplo, si nuestra señal fuera una cosenoidal perfecta, su espectro serían dos funciones impulso deltas de Dirac, reales, ubicadas en la frecuencia de oscilación de la misma (4) simétricamente respecto al eje de:

$$x(t) = A \cos(\omega_0 t) \Leftrightarrow X(\omega) = A\pi [\delta(\omega - \omega_0) - \delta(\omega + \omega_0)]$$



Esto quiere decir que cualquier señal periódica que no sea senoidal poseerá un espectro compuesto, es decir, podemos expresarla como combinación lineal de señales senoidales y cosenoidales cuyas frecuencias sean múltiplos enteros de la frecuencia de la señal analizada. Dichas señales son llamadas armónicas. El espectro complejo queda formado entonces por un conjunto de deltas de Dirac:



La transformada de Fourier de señales periódicas da entonces los valores de amplitud de cada armónica. Usando esos valores de amplitud se puede expresar a la señal periódica del tiempo por medio de una sumatoria de señales armónicas, es decir de una serie de Fourier (5):

$$x(t) = c_0 + \sum_{n=1}^{\infty} c_n e^{jn\omega_0 t}$$

$c_0$  es el valor de continua. Si hacemos uso de la formula de Euler (6):

$$e^{jn\omega_0 t} = \cos n\omega_0 t + j \sin n\omega_0 t$$

Y teniendo en cuenta que los valores de  $c_n$  son complejos, podemos expresar la serie de Fourier como (7):

$$x(t) = a_0 + \sum_{n=1}^{\infty} a_n \cos n\omega_0 t + j b_n \sin n\omega_0 t$$

Aquí  $c_0$  devino en  $a_0$ . Esta "amplitud compleja" de las armónicas es consecuencia de que existen desfases entre ellas, lo cual queda de manifiesto si expresamos a cada coeficiente como (8):

$$c_n = \frac{a_n - j b_n}{2} = |c_n| e^{j\phi_n}$$

$$|c_n| = \frac{1}{2} \sqrt{a_n^2 + b_n^2}$$

$$\phi_n = \text{arctg} \left( -\frac{b_n}{a_n} \right)$$

Esta propiedad es la base del método que usaremos para encontrar el valor de distorsión de las señales. La definición de distorsión armónica viene dada por (9):

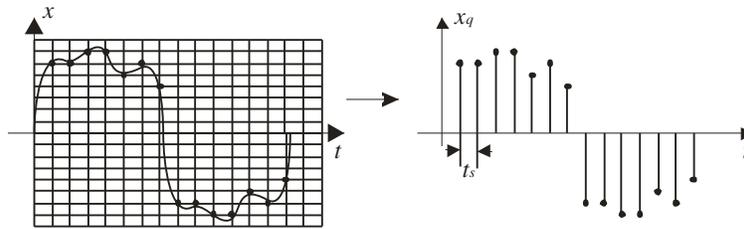
$$THD_X = \sqrt{\frac{\sum_{n=2}^{\infty} X_n^2}{X_{RMS}^2}}$$

Donde X puede referirse tanto a la tensión como a la corriente en el caso de evaluar la energía eléctrica.  $X_n$  se refiere al módulo de la amplitud de la n-sima armónica de la señal.

Esta expresión indica que la distorsión, como dijimos antes, es consecuencia exclusiva de la presencia de armónicos.

A la hora de implementar sistemas que realicen un análisis armónico en forma digital surgen nuevas consideraciones que pasaremos a enumerar y explicar.

La primera cuestión a considerar es el paso del mundo analógico al mundo digital. Esto es llevado a cabo por un conversor analógico – digital o ADC (por sus siglas en ingles). El ADC monitorea los valores de señal y los cuantiza. En los sistemas digitales, las variables no pueden tomar cualquier valor arbitrario, sino que quedan definidos un numero finito de amplitudes que la señal puede tomar. Estos valores pueden ser equidistantes o tener separaciones variables de acuerdo a una ley definida de acuerdo a la aplicación:



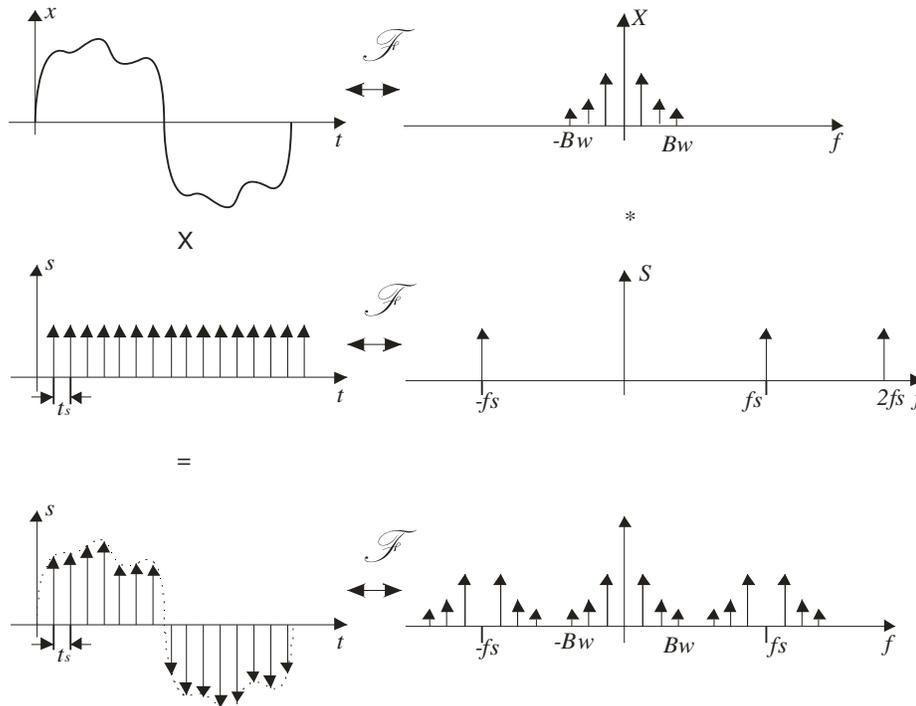
En esta figura también podemos ver otro hecho intrínseco de los sistemas digitales, que es el muestreo. Es decir, no solo cuantizamos los valores de la señal, sino que no podemos obtener todos los valores cuantizados de la misma, sino que tomamos valores de la señal en ciertos intervalos de tiempo (tiempo de muestreo  $t_s$ ). Esto es debido a varias razones:

- El proceso de cuantización no es instantáneo, lo cual ya da un mínimo para el tiempo de muestreo
- Si se pudiera cuantizar de forma instantánea la memoria necesaria para almacenar los datos sería infinita (esto traerá otras consecuencias)

La cuantización tendrá como resultado la introducción de un error numérico.

En cuanto al muestreo, hay ciertas condiciones que pueden guiarnos a la hora de elegir el mayor de tiempo de muestreo posible, y así, minimizar la cantidad de datos necesarios.

Considerando al muestreo como una multiplicación en el tiempo por un tren de impulsos delta de Dirac separados por el periodo  $t_s$ , obtendremos:



Podemos destacar que se presentan aquí 2 propiedades de la transformada de Fourier:

- El espectro de un tren de impulsos es otro tren de impulsos separados por una distancia igual a la inversa del tiempo de muestreo (10)

$$f_s = \frac{1}{t_s}$$

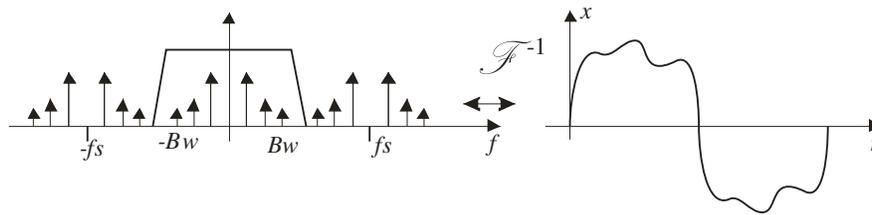
- La transformada de un producto de 2 señales en el tiempo es la convolución de sus espectros (11)

Y dado que la convolucion de una señal con un impulso da como resultado una traslación de la señal, vemos que el muestro genera un espectro "periódico".

Para que no hayan solapamientos, es fácil ver que la máxima componente de frecuencia de la señal no debe superar a la mitad de la frecuencia de muestro. Esta condición se conoce como criterio de Nyquist (12)

$$f_s > 2B_w$$

La reconstrucción de la señal se podrá hacer simplemente seleccionando aquella porción del espectro que queda centrada en la frecuencia cero, usando un filtro pasa bajos:



Donde  $B_w$  es el ancho de banda de la señal a muestrear. De no cumplirse la condición de Nyquist, o sea, si se solapan los espectros, será imposible la reconstrucción de la señal. Dicho efecto se denomina aliasing, y se puede eliminar limitando el ancho de banda de la señal bajo análisis por medio de un filtro pasa bajos, denominado "antialias", para luego muestrearla usando una frecuencia de muestro al menos 2 veces la frecuencia de corte superior del filtro.

Si bien ya vimos como el muestreo y la cuantización influyen en el análisis y que precauciones hay que tomar, todavía restan otros detalles. De la misma forma que el muestro transformo nuestra señal desde el dominio continuo del tiempo al dominio discreto, en este último debemos aplicar la transformada discreta de Fourier (DFT) (13):

$$X(m) = \sum_{n=0}^{N-1} x(n) e^{-j2\pi nm/N}$$

Donde  $x(n)$  es una secuencia de muestras,  $m$  será un índice a otra secuencia pero de componentes complejas de frecuencia, y  $N$  es el número de muestras que procesaremos. Ahora el dominio de la frecuencia es discreto, es decir, queda cuantizado de acuerdo a la frecuencia de muestro y a la cantidad de muestras que usamos en el análisis (14):

$$\Delta f = \frac{f_s}{N}$$

Denominamos a dicho valor como resolución espectral. Podemos ver que si fuera posible procesar un número infinito de muestras obtendríamos un espectro continuo de frecuencias. Así también podemos ver que mientras más muestras sean procesadas, mas resolución espectral tendremos.

La implementación del cálculo de la DFT en un procesador digital implica que por cada muestra estemos efectuando:

- $N^2$  multiplicaciones complejas (que constan de 2 multiplicaciones reales y una suma, cada una)

- $N(N-1)$  sumas complejas (que consta de 2 sumas reales)

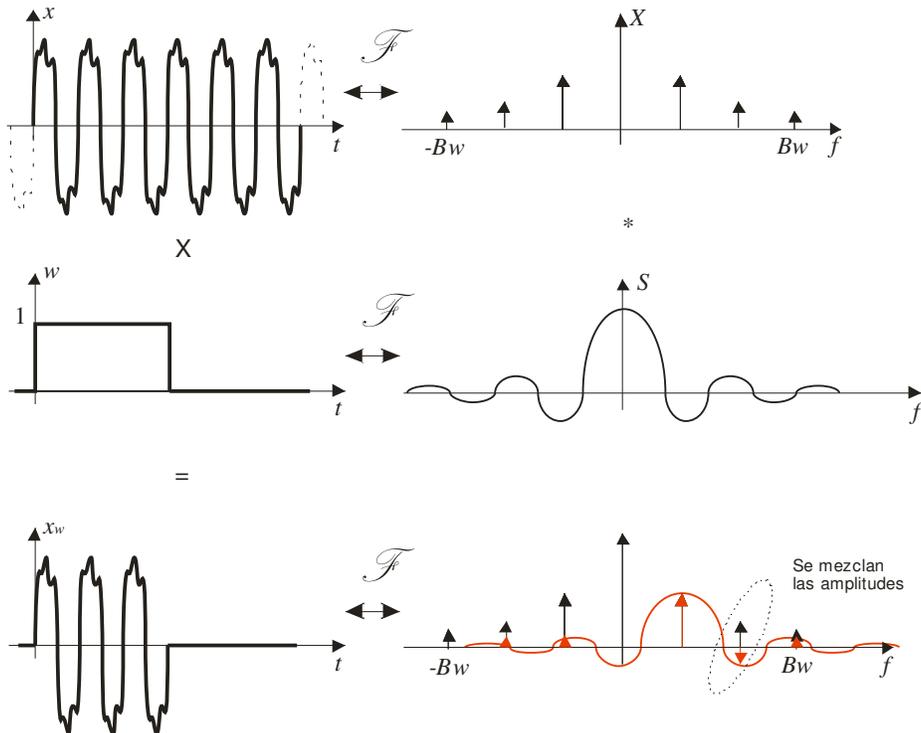
Aprovechando las simetrías propias de la DFT (cálculos que se repiten), se desarrolló un algoritmo que reduce la cantidad de operaciones necesarias, llamado transformada rápida de Fourier (FFT) (15), que tiene muchas menos operaciones:

- $\left(\frac{N}{2}\right) \log_2 N$  multiplicaciones complejas

Una condición que tenemos que tener en cuenta para aprovechar realmente la FFT es que  $N = 2^k$ , siendo k cualquier entero positivo.

Otra consecuencia muy importante del hecho de no tener un número infinito de muestras es que si queremos realizar un análisis de un señal en forma ininterrumpida, estamos obligados a limitar el número de muestras, trabajando con bloques de datos que puedan ser almacenados en una memoria para su posterior análisis, es decir, hay que tomar un numero finito de muestras. Esto se hace almacenando datos durante un periodo determinado, o sea, durante una ventana de tiempo. De allí deviene el nombre de este proceso: ventaneo.

Matemáticamente, equivale a multiplicar nuestra señal por una función del tiempo tal que la misma valdrá 0 para cualquier muestra que este fuera del periodo de la ventana. La forma más simple, y en realidad, intrínseca a este proceso, es el uso de la ventana rectangular, que se aplica simplemente tomando muestras durante un periodo dado. Supongamos una señal periódica continua, con su espectro continuo, formado por componentes puras de frecuencia, de acuerdo a su representación como serie de Fourier y tomamos un trozo de la señal para luego analizarla, equivale a multiplicar la señal por otra señal rectangular de duración igual al periodo de observación:



En la figura se muestra el efecto de distorsión sobre la amplitud de las componentes originales, dado que el espectro de la ventana se traslada sobre cada componente de la señal periódica

$$x'(t) = w(t)x(t)$$

Esto se manifestará como una convolución en el dominio frecuencial, como vimos antes:

$$X'(\omega) = X(\omega) * W(\omega)$$

De allí que, como resultado de esa convolución espectral, el espectro de la señal original se ve repetido y ponderado de acuerdo al espectro de la señal rectangular.

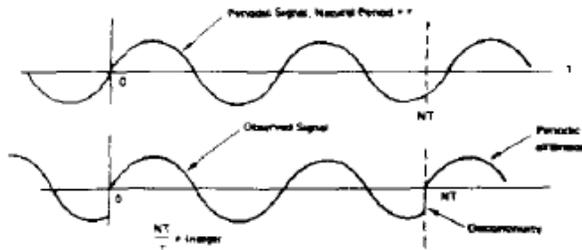


Fig 1. La señal no es periódica en el intervalo de observación, lo que produce la aparición de componentes de frecuencia adicionales, el leakage espectral muestreada, dando como resultado un espectro compuesto, es decir, el espectro de cada armónica se verá superpuesto, produciendo distorsión.

Podemos pensar este efecto de otro modo. Como vimos antes, la DFT nos da una resolución de frecuencia que determina que los valores de amplitud resultantes serán validos para componentes de frecuencia que sean múltiplos de la resolución, o sea, para  $\Delta f, 2\Delta f, 3\Delta f$ , etc. Si la señal posee componentes de frecuencia fuera de esos valores, por ejemplo en  $1.5 f_s / N$ , la energía de dicha componente aparecerá distribuida sobre casi todos los bins (así llamamos a las componentes de frecuencia obtenidas con la FFT) (Ver 16 para un análisis más detallado de

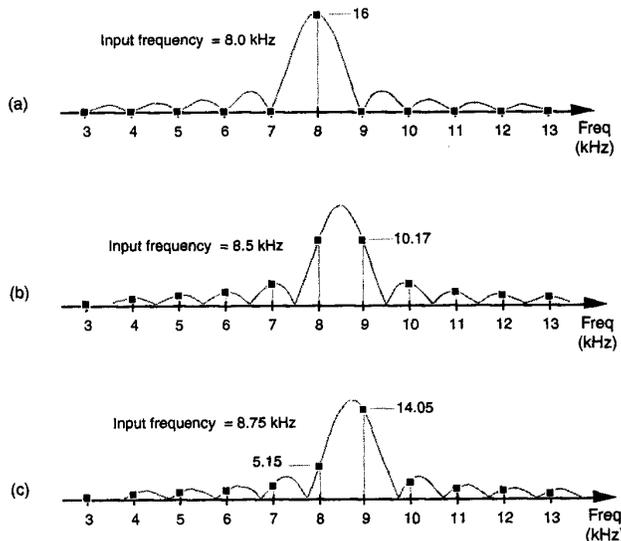


Fig 2. Efectos resultantes de medir señales de frecuencias no correspondientes a un bin

En los casos en que la longitud de la ventana (expresada en muestras) sea múltiplo exacto de un período de señal, ese efecto se anula (16) (figura 2.a). En caso contrario, se hará manifiesta “la no periodicidad” de la señal

este importantísimo problema, típico en estos sistemas).

Esto da lugar a un fenómeno llamado spectral leakage, que se manifiesta dando lugar a la aparición de armónicos en frecuencias donde en realidad, no hay nada. Esto complica la estimación de los valores de amplitud y la frecuencia de cada armónico.

En la figura se ve una grafica que muestra el error que se introduce por no medir señales cuyas

frecuencias caigan en un bin de manera exacta:

Surgen varios métodos para tratar de solucionar este tema. Por ejemplo, intuitivamente podríamos plantear el uso de algún método de sincronismo y variación de la frecuencia de muestreo para que siempre se obtenga una señal periódica respecto de la ventana, pero dado la complejidad que llevaría aparejado tal enfoque, se optará por un método menos exacto pero **de menor dificultad de implementación**.

Sin embargo, la elección adecuada de la ventana puede minimizar el error y el leakage. (18)

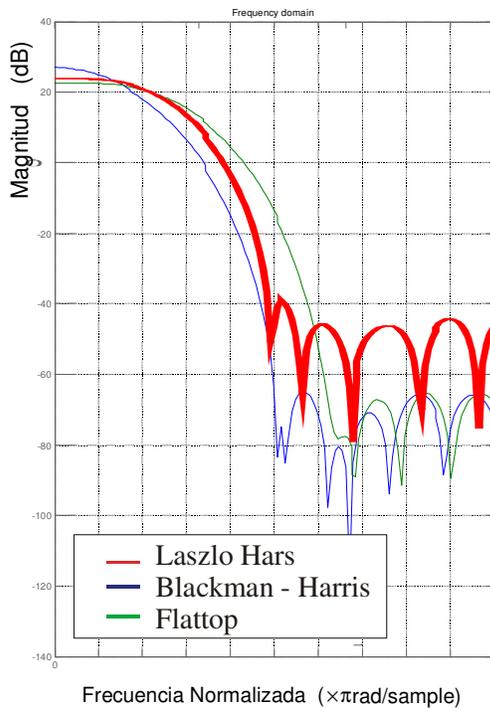
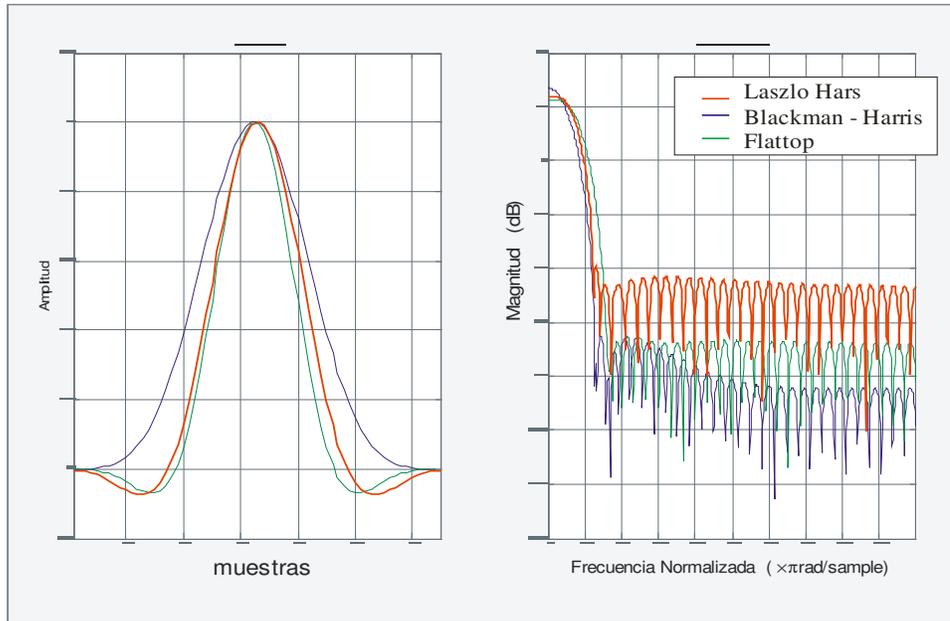
La idea sería que el espectro de la ventana sea “angosto” para que al superponerse con el espectro de la armónica adyacente no genere armónicos fantasma de gran amplitud. También, dado que seguramente la frecuencia fundamental de la señal a medir no será múltiplo exacto de los bins de frecuencia, será necesario que la amplitud de la banda de paso del espectro de la ventana sea lo más plana posible. Para definir que es “angosto” y que es “plano” definamos algunos términos útiles en las ventanas y sus espectros:

Cuando hablamos del ancho espectral de la ventana, nos referimos a la frecuencia donde la amplitud del espectro encuentra su primer mínimo. Por planicidad nos referimos a la velocidad de variación de la amplitud del lóbulo principal. Los lóbulos laterales son los que serán más importantes a la hora de tratar de reducir el leakage, ya que debemos buscar que sus amplitudes sean mínimas.

**En la selección de la ventana se deben tener en cuenta dos aspectos fundamentales: el ancho del espectro de la ventana y la planicidad en la banda de paso. El ancho del espectro es importante en el solapamiento entre los espectros de las armónicas adyacentes, para evitar armónicos fantasma de gran amplitud. En cambio, la planicidad es necesaria ya que seguramente la frecuencia fundamental de la señal a medir no será múltiplo exacto de los bins de frecuencia, y será necesario que la amplitud que se obtenga en el bin más cercano sea igual a la de la fundamental.**

Hay varias ventanas que nos ofrecen ventajas y desventajas y que son usadas usualmente en aplicaciones de análisis espectral, de las cuales seleccionamos algunas para realizar un breve análisis comparativo entre ellas: Flattop, Blackman-Harris, y el método de hallar ventanas tipo serie de cosenos optimizando los parámetros de ancho de banda y atenuación de los lóbulos laterales. **Un grafico comparativo de dichas ventanas se encuentra en la siguiente página.**

La ventana Flattop es la que posee la mejor característica de amplitud, por su planicidad, pero su atenuación de lóbulos laterales es pobre en comparación a la de Blackman-Harris, lo que da mucho leakage y distorsionará a las amplitudes de las armónicas vecinas. Sin embargo, si la separación entre los armónicos es mayor al ancho espectral, el nivel de leakage no se verá afectado. En nuestra aplicación resulta importante lograr la máxima planicidad para que la amplitud del armónico bajo estudio no se vea afectada y así minimizar el error.



La expresión de estas ventanas es, en forma genérica:

$$w(n) = a_0 + a_1 \cos\left(\frac{2\pi}{N} n\right) + a_2 \cos\left(\frac{2\pi}{N} 2n\right) + a_3 \cos\left(\frac{2\pi}{N} 3n\right)$$

$$n = 0, 1, 2, \dots, N - 1$$

El estudio de Harris (19) tabula los coeficientes para la ventana Blackman – Harris en función de la amplitud relativa del lóbulo lateral:

	3-Term (-67 dB)	3-Term (-61 dB)	4-Term (-92 dB)	4-Term (-74 dB)
$a_0$	0.42323	0.44959	0.35875	0.40217
$a_1$	0.49755	0.49364	0.48829	0.49703
$a_2$	0.07922	0.05677	0.14128	0.09392
$a_3$	---	---	0.01168	0.00183

Hay que tener en cuenta que mientras menos amplitud del lóbulo lateral deseemos, más se ensancha el lóbulo principal.

Por último, Laszlo Hars (20) evaluó un método de optimización para el hallazgo de ventanas expresadas como series cosenoidales, de la misma estructura que la de Blackman-Harris.

Las 3 ventanas tienen las siguientes expresiones:

$$w_{\text{flattop}}[n] = 0.2156 - 0.4160 \cos\left(\frac{2\pi n}{N}\right) + 0.2781 \cos\left(\frac{4\pi n}{N}\right) - 0.0836 \cos\left(\frac{6\pi n}{N}\right) + 0.0069 \cos\left(\frac{8\pi n}{N}\right)$$

$$w_{\text{Blackman-Harris}}[n] = 0.35875 + 0.48829 \cos\left(\frac{2\pi n}{N}\right) + 0.14128 \cos\left(\frac{4\pi n}{N}\right) + 0.01168 \cos\left(\frac{6\pi n}{N}\right)$$

$$w_{\text{LazsloHars}}[n] = 1.0013591 - 1.8979304 \cos\left(\frac{2\pi n}{N}\right) + 1.056186 \cos\left(\frac{4\pi n}{N}\right) - 0.17908511 \cos\left(\frac{6\pi n}{N}\right)$$

La Blackman – Harris es la que logra la mejor atenuación de los lóbulos laterales, lo cual determina que es la ventana que menos leakage introduce. Tanto la Flattop como la de Hars son mas planas, pero tienen mucho mas leakage.

También hay que considerar que el ventaneo introduce un cambio en la amplitud de las componentes, que debe ser ajustado según un coeficiente igual a la inversa del área encerrada por ella.

$$y[n] = w[n]x[n] \Leftrightarrow Y[k] = W[k] * X[k]$$

Aplicando la identidad de Parseval, podemos ver que la energía en ambos dominios, temporal y de frecuencia, debe ser la misma:

$$\sum_{n=0}^{N-1} |x[n]w[n]|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X[k] * W[k]|^2$$

La ventana, al atenuar la amplitud en los extremos del bloque de datos, reduce la potencia total. Dado que estudiamos señales periódicas, y por ende, expresables en series de Fourier:

$$x(t) = a_0 + \sum_n c_n e^{jn\omega_0 t} \Leftrightarrow X(\omega) = a_0 \delta(\omega) + \sum_n c_n \delta(\omega - n\omega_0)$$

el espectro que resulta de aplicarle la correspondiente DFT, tal que  $\omega_0$  sea un múltiplo exacto de la resolución espectral, cosa que asumiremos por ahora, será una sumatoria de impulsos: Aplicando el ventaneo:

$$Y[k] = W[k] * X[k] = W[k] * \left\{ a_0 \delta[0] + \sum_n c_n \delta[k - nk_0] \right\} =$$

$$Y[k] = a_0 W[k] * \delta[0] + \sum_n c_n W[k] * \delta[k - nk_0] =$$

$$Y[k] = a_0 W[0] * \delta[0] + \sum_n c_n W[k - nk_0]$$

De aquí se desprende que la amplitud de cada armónico queda modificada por el factor  $W[0]$ , ya que:

$$k = nk_0 \rightarrow W[nk_0 - nk_0] = W[0] = \frac{1}{N} \sum_{n=1}^N w[n]$$

Que no es más que el término de continua de la ventana. Por eso se llama a este parámetro ganancia de continua (DC Gain). El valor que deseamos obtener es  $c_n$  :

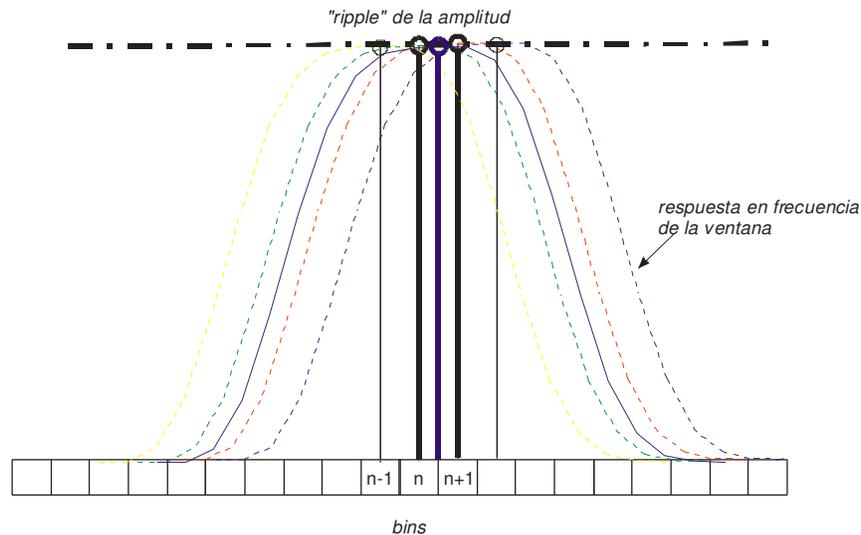
$$c_n = \frac{Y[nk_0]}{W[0]} = k \cdot Y[nk_0]$$

$$k = \frac{1}{W[0]}$$

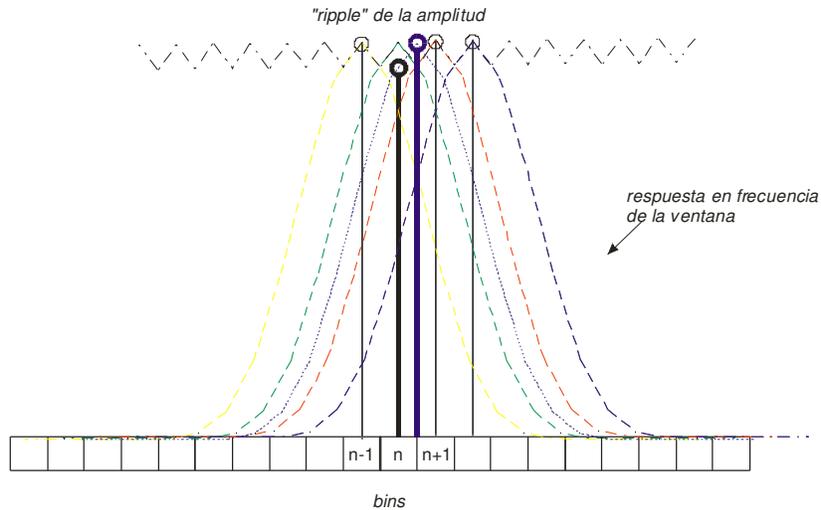
Esto es fundamental para hallar el verdadero valor de amplitud del armónico bajo estudio.

Este análisis es válido solo cuando los bins de frecuencia son tales que son divisores exactos de la frecuencia  $k_0$ . Esto nunca se cumple. Entran en juego aquí la planicidad de la ventana

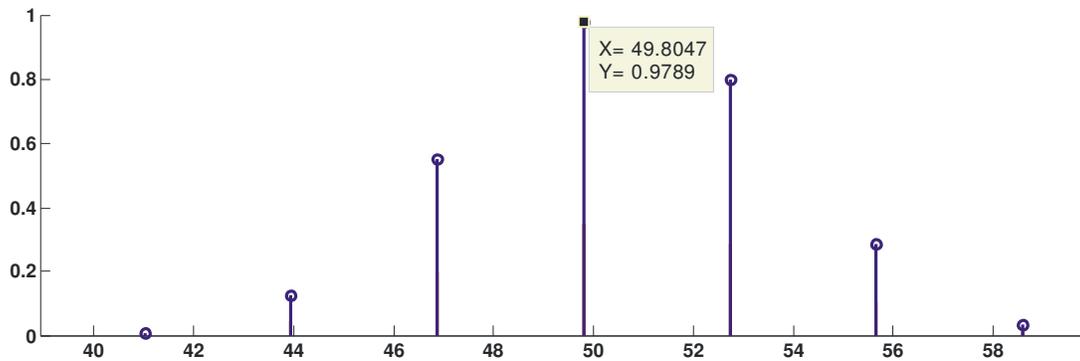
Podemos ver que la mayor planicidad resulta en el menor error en la determinación de la amplitud:



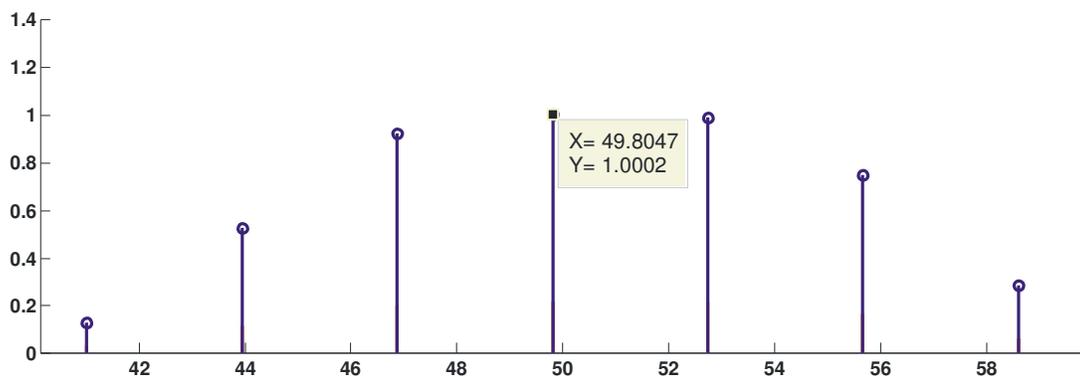
En la figura anterior vemos que cuando estamos midiendo una señal de una frecuencia que no es múltiplo del valor de resolución espectral, tomamos el valor correspondiente al bin más cercano. La amplitud medida casi no tiene error debido a la altísima planicidad de esta ventana. A continuación se ve el ripple que resultaría de usar otro tipo de ventana, en este caso, la Blackman-Harris, donde podemos observar el error resultante de medir una señal como la del caso anterior, debido a la menor planicidad de la ventana en cuestión. Sin embargo, en aplicaciones donde se necesite menor influencia entre bins vecinos, esta ventana muestra ser más angosta.



Veamos esto comparando los resultados simulados con Matlab para la medición de una señal cuya frecuencia no es la de un bin exacto. En el ejemplo, medimos una señal de amplitud unitaria y 50Hz aplicando una FFT cuyo bin más cercano esta en 49.8047Hz. En el primer gráfico observamos el resultado para la ventana de Blackman-Harris



En cambio para la ventana Flatop:



Podemos observar que si bien la Flatop es más ancha, cosa evidente en la mayor amplitud relativa de los bins adyacentes respecto al bin principal, tiene un error de amplitud muchísimo menor.

Un método alternativo para el cálculo de los valores de cada armónica es basarse en el hecho de que la energía de cada armónica queda distribuida a lo largo del leakage de la misma. Si una vez detectada cada armónica, sumamos las energías de los bins adyacentes (sus valores al cuadrado) y le aplicamos la raíz cuadrada, obtendremos la energía de la componente de frecuencia analizada:

$$X_{bin} = k \sqrt{\sum_{i=bin-4}^{bin+4} X_i^2}$$

Donde k es el valor de corrección, que es la inversa del valor rms de la ventana en sí:

$$k = \frac{1}{\sqrt{\sum_{i=1}^N w_i^2}}$$

La estimación de la amplitud de cada componente armónico puede llevarse a cabo de acuerdo a 2 métodos:

- 1- Basados en que la amplitud medida es la correspondiente a la frecuencia real de la componente en cuestión, se pondera su valor según la ventana usada
- 2- Se suman las energías de los bins cercanos a la frecuencia real de la componente a medir, y se pondera el resultado según el valor de corrección correspondiente a la ventana usada

El primero de estos métodos es más fácil y rápido de implementar que el segundo, aunque la simulación en Matlab reveló que usando esta metodología se obtiene un valor de amplitud más preciso sin importar la ventana usada. La desventaja del segundo método reside en que conlleva más cálculos, y por ende, es más lento que el método anterior.

Dado que estimar la frecuencia real de cada componente no es primordial en las mediciones de THD y FP, el ancho espectral deberá ser tal que permita la distinción entre componentes armónicas. Mientras más muestras tenga la ventana, más angosta espectralmente será. Quiere decir que tomando una cantidad de muestras adecuada, podemos manejar el ancho espectral.

Como deseamos una mayor velocidad de cómputo, optamos por usar el primer método, lo cual nos lleva a la elección de la ventana más plana, la Flattop.

### 3. Definiciones de los parámetros de calidad de energía buscados en la medición

Deseamos medir algunos parámetros esenciales en la calidad de energía: la distorsión armónica, tanto de tensión como la de corriente y el factor de potencia. La distorsión armónica de una señal mide la relación entre el valor energético del contenido armónico y el de la señal fundamental. En la práctica se suele medir la distorsión respecto a la señal con todo su contenido armónico. Esto es como se indicó antes:

$$THD_x = \sqrt{\frac{\sum_{n=2}^{\infty} X_n^2}{X_{RMS}^2}} = \sqrt{\frac{X_{RMS}^2 - X_1^2}{X_{RMS}^2}}$$

Donde el numerador contiene la energía de las armónicas y el denominador, la energía total de la señal

El factor de potencia, en cambio, es la relación entre la potencia activa y la potencia aparente:

$$FP = \frac{P}{S} = \frac{\frac{1}{T} \int_{n=1}^{\infty} v(t)i(t) dt}{V_{RMS} I_{RMS}}$$

En sistemas de tiempo discreto, la integral pasa a ser una sumatoria:

$$FP = \frac{P}{S} = \frac{\frac{1}{N} \sum_{n=1}^N v[n]i[n]}{V_{RMS} I_{RMS}}$$

Será de interés obtener el valor RMS partiendo de la formula de la THD:

$$THD = \sqrt{\frac{\sum_{n=2}^{\infty} V_n^2}{V_{RMS}^2}} = \sqrt{\frac{V_{RMS}^2 - V_1^2}{V_{RMS}^2}} = \sqrt{1 - \frac{V_1^2}{V_{RMS}^2}} \Rightarrow THD^2 = 1 - \frac{V_1^2}{V_{RMS}^2} \Rightarrow$$

$$\Rightarrow 1 - THD^2 = \frac{V_1^2}{V_{RMS}^2} \Rightarrow V_{RMS}^2 = \frac{V_1^2}{1 - THD^2} \Rightarrow V_{RMS} = \sqrt{\frac{V_1^2}{1 - THD^2}}$$

Para el caso del factor de potencia, buscamos una expresión que nos sirva ya que hemos obtenido la distorsión de corriente y de tensión:

$$FP = \frac{P}{S} = \frac{\frac{1}{T} \int_0^T v(t)i(t) dt}{V_{RMS} I_{RMS}} = \frac{\frac{1}{T} \int_0^T v(t)i(t) dt}{\sqrt{\frac{V_1^2}{1 - THD_v^2}} \sqrt{\frac{I_1^2}{1 - THD_i^2}}}$$

$$FP = \frac{\frac{1}{T} \int_0^T v(t)i(t) dt}{V_1 I_1} \sqrt{1 - THD_v^2} \sqrt{1 - THD_i^2}$$

La integral del numerador es el término de continua del producto VI, y se puede hallar fácilmente promediando los valores de las muestras a medida van ingresando, previa multiplicación entre los valores de tensión y corriente.

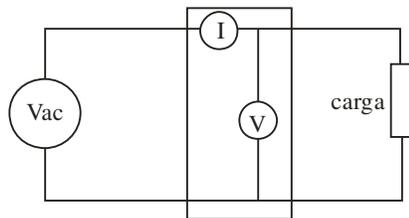
El FP puede entonces hallarse como:

$$FP = \frac{P}{S} = \frac{\frac{1}{N} \sum_{n=0}^{N-1} v(n)i(n)}{\frac{\hat{V}_1 \hat{I}_1}{2}} \sqrt{1 - THD_v^2} \sqrt{1 - THD_i^2}$$

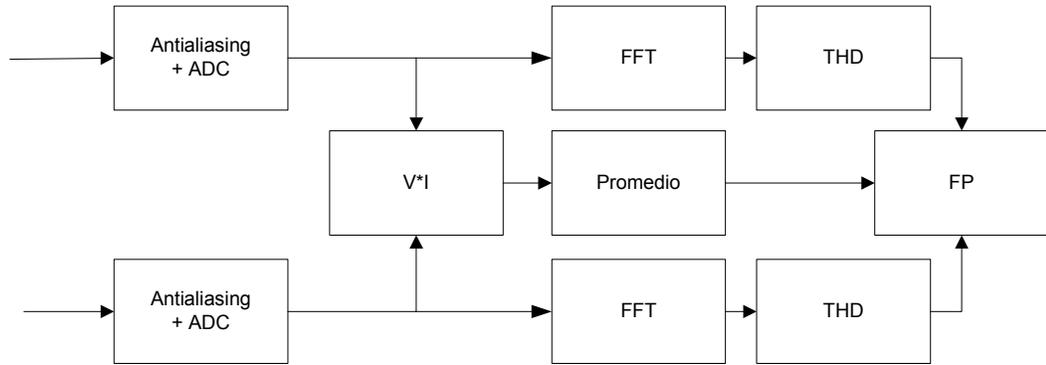
#### 4. Diagrama preliminar del sistema

Teniendo en cuenta el estudio teórico que hemos realizado, y antes de pasar a los detalles de la implementación podemos enumerar las características y bloques constitutivos básicos del sistema

- El sistema debe conectarse de forma tal que podamos obtener tanto el valor de la corriente que consume una carga determinada como la tensión aplicada a la misma:



- Una vez acondicionados los niveles de las señales de corriente y tensión, previo filtrado (antialiasing) serán muestreadas por sendos ADC, cuyas salidas digitales deberán estar conectadas al procesador digital de señales (DSP)
- Se almacenarán los datos en 2 buffers (uno para la señal de tensión y otro para la de corriente), al mismo tiempo que se promediara el valor instantáneo del producto  $v \cdot i$  a fines de obtener la potencia activa consumida por la carga.
- Con los datos almacenados, se calculará la FFT de ambas señales a fin de obtener el contenido armónico de ambas
- Con esa información se obtendrá la distorsión armónica total (THD) tanto de tensión como de corriente
- Con la THD se hallará el valor de la potencia aparente
- Se realizará el cociente de la potencia activa y la aparente para determinar el factor de potencia



**5. Requisitos de la implementación:**

La norma IEC 61000-2-3 determina los niveles máximos de armónicos introducidos por los aparatos eléctricos en la red de consumo en función de la potencia consumida por los mismos. La misma utiliza mediciones para sus consideraciones hasta el armónico número 40. Si queremos medir señales con una frecuencia fundamental de 50 o 60 Hz, la mayor frecuencia a medir, considerando variaciones del 10% en la misma es:

$$f_{\max} = 40 f_{0_{\max}} = 40 \cdot 60 \cdot 1.1 = 2640$$

De acuerdo a Nyquist, necesitamos una frecuencia de muestreo:

$$f_s \geq 2 f_{\max} = 5280$$

La resolución espectral nos pone ante el desafío de que si queremos medir señales con fundamentales de 50 y 60Hz, la resolución espectral ideal debería darnos bins que “caigan” en ambas frecuencias. Como la resolución espectral es:

$$\Delta f = \frac{f_s}{N}$$

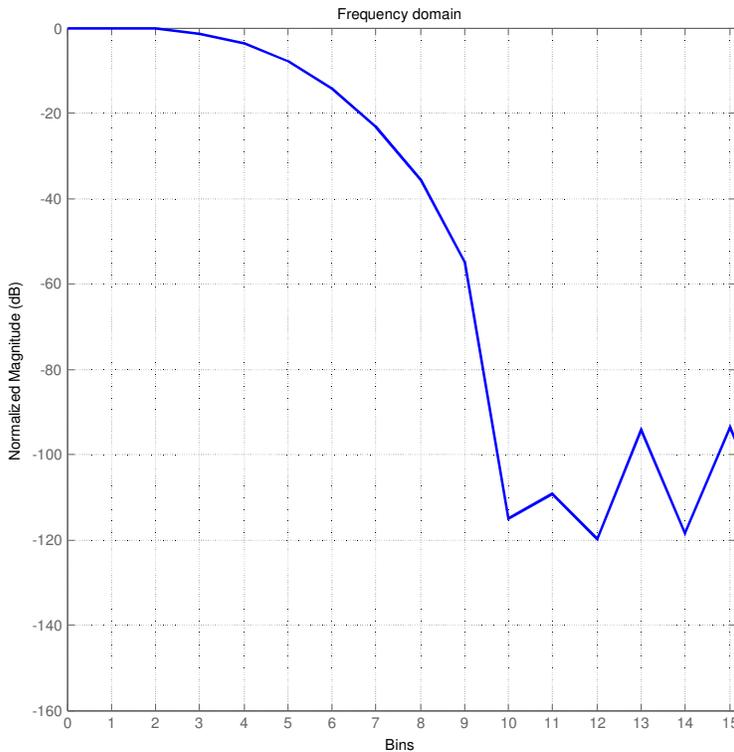
Frecuencia de submuestreo	
fs (Hz)	48000
M	f's (Hz)
4	12000
5	9600
6	8000
7	6857.14286
8	6000
9	5333.33333
10	4800
11	4363.63636
12	4000
13	3692.30769

$f_s$  debe ser múltiplo del MCM de 50, 60, o sea, múltiplo de 300. N además debe ser potencia de 2 para que el algoritmo de la FFT pueda ser llevado a cabo, con lo cual  $\Delta f$  podrá ser reducido a la mitad, la cuarta parte, etc. Además, el kit de evaluación tiene una frecuencia de muestreo de 48000Hz, con lo cual,  $f_s$  debe además ser divisor entero de 48000Hz para que sea posible llegar a  $f_s$  mediante submuestreo. Tal frecuencia no existe. Pero, teniendo en cuenta que nunca tendremos señales cuya frecuencia sean 50 o 60Hz exactos, no tiene sentido buscar tal valor de  $f_s$ .

Tendremos que apelar a la planicidad de la ventana, esto es, dado que no podemos obtener una frecuencia de muestro

que nos de bins exactos para 50 y 60 Hz, buscaremos minimizar el tamaño de los bins aumentando en número de muestras de la FFT y se estimará la frecuencia de muestreo que nos de los bins más cercanos posibles a dichas frecuencias.

La frecuencia de muestreo, de acuerdo a la condición de Nyquist, debe al menos duplicar la frecuencia de la máxima armónica a analizar. En nuestra aplicación, además, debe ser divisor entero de 48000Hz para poder hacer un submuestreo, es decir, tomar una muestra cada M muestras y lograr así otra frecuencia de muestreo:



Dado que ya habíamos determinado que

$$f_s \geq 2f_{\max} = 5280$$

Elegimos:

$$f_s = 6000\text{Hz}$$

Ubicación de los bins

bin	N=2048
10	29.296875
11	32.2265625
12	35.15625
13	38.0859375
14	41.015625
15	43.9453125
16	46.875
17	49.8046875
18	52.734375
19	55.6640625
20	58.59375
21	61.5234375
22	64.453125
23	67.3828125

es decir, tomaremos una muestra cada 8. Esto nos va a imponer la necesidad de realizar un filtrado antialiasing con frecuencia de corte

$$f_c = 3000\text{Hz}$$

Para la selección del valor de N (cantidad de muestras que procesaremos con la FFT) se precisa alcanzar una resolución espectral tal que de acuerdo a la ventana elegida el leakage de los armónicos no influya notablemente en los armónicos. **Para la ventana Flattop se gráfico en la figura el ancho de la misma en bins**

Se puede observar que la ventana Flattop posee un ancho de 60dB de casi 9 bins, y que la menor frecuencia a medir será 45Hz (50Hz – 10%):

Resolución espectral

N	$\Delta f$ (Hz)
128	46.875
256	23.4375
512	11.71875
1024	5.859375
2048	2.9296875
4096	1.46484375

$$\Delta f < \frac{f_{\min}}{B_w [bins]} = \frac{45Hz}{9} = 5$$

Como N debe ser una potencia de 2 para que el algoritmo de FFT sea eficiente, el valor mas cercano se obtuvo para N=2048.

Llegamos a la conclusión de que podemos usar:

$$f_s = 6000Hz \quad M = 8$$

$$N = 2048$$

$$\Delta f = \frac{f_s}{N} = \frac{6000Hz}{2048}$$

$$\Delta f = 2.9296875Hz$$

Teniendo en cuenta el rango de variación típico de la frecuencia de red:

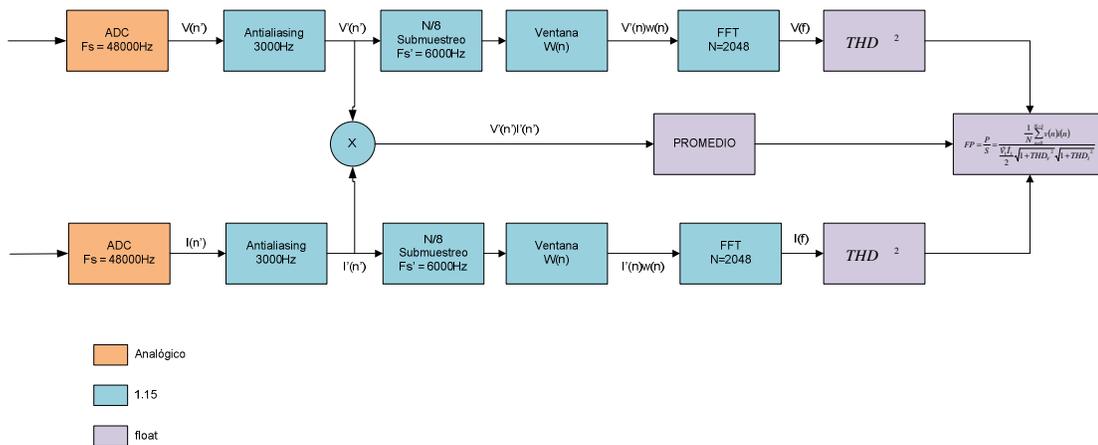
$$f = 50Hz \pm 10\% \rightarrow 45Hz < f < 55Hz$$

$$f = 60Hz \pm 10\% \rightarrow 54Hz < f < 66Hz$$

En la tabla de la página anterior se observa que la ubicación en frecuencia de los bins cubre el rango de frecuencias a medir.

### 6. Implementación práctica.

La implementación se llevará a cabo de acuerdo al diagrama en bloques de la siguiente página, donde se ha indicado además el tipo de notación que se usa en los cálculos correspondientes a cada etapa.



En el podemos ver que en primera instancia llevaremos a cabo un filtrado antialiasing de las señales adquiridas con una frecuencia de corte de 3000Hz.

Con las señales ya filtradas, procedemos a multiplicar sus valores para hallar luego la potencia activa como el promedio de dicho producto.

Luego se procede al submuestreo, que no es más que tomar una de cada M muestras (8 en nuestro caso) y se las multiplica por la ponderación correspondiente, de acuerdo a la ventana elegida. Estos valores ya ponderados son almacenados en buffers de memoria, para luego llevar a cabo sobre ellos una FFT.

Con la información del contenido armónico de ambas señales, de corriente y tensión, procedemos a calcular la THD de cada una de ellas.

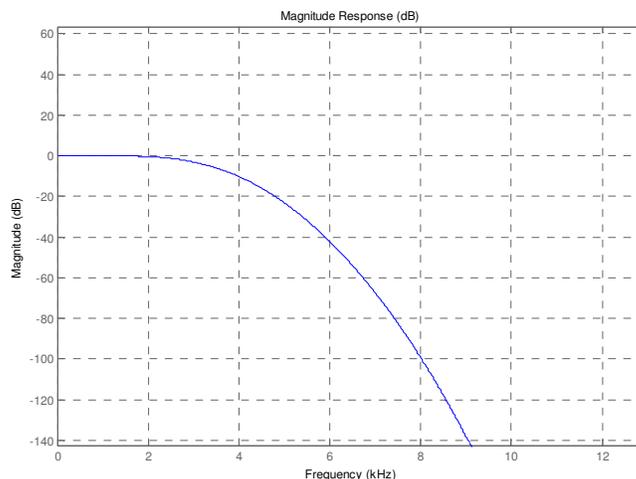
Con estos valores y el valor de la potencia activa, obtenemos finalmente el valor del factor de potencia, y enviamos los valores obtenidos a una PC para su visualización.

## 7. Detalles de la implementación

Dado que usaremos el kit de evaluación del DPS Blackfin BF-537 de Analog Devices, partimos de que sus entradas analógicas tienen un filtro antialiasing con  $f_c = 24\text{kHz}$ , y una velocidad de muestreo de  $48000\text{Hz}$ , la interrupción del muestreador ADC se ejecuta cada  $1/48000\text{Hz} = 20.8\mu\text{s}$ . En ese ínterin debe ejecutarse la convolución del filtro antialiasing.

El mecanismo de cálculo nos lleva a tomar dos muestras en cada interrupción, una de tensión y otra de corriente. Antes de llevar a cabo el submuestreo tomando una de cada ocho muestras, deben ser filtradas a  $3000\text{Hz}$  de corte, para evitar un aliasing al submuestrear a  $6000\text{Hz}$ . Esto lo hacemos con un filtro FIR, convolucionando las señales de entrada con filtros obtenidos con el **toolbox fdatool** de Matlab, que se han incorporado al soft.

El filtro que se implementará se diseño buscando la máxima planicidad en la banda de paso para minimizar el error de amplitud. Se eligió la opción “maximally flat” del fdatool, pero sus resultados no fueron satisfactorios. Para el caso de un filtro de orden 130 (máximo permitido por la herramienta) se obtuvo la siguiente respuesta en frecuencia:



La misma tiene una atenuación muy pobre fuera de la banda de paso, y es necesario un filtro lo mas abrupto posible. Sacrificamos entonces la planicidad y pasamos a un filtro del tipo equiripple. Esta vez, la herramienta nos permite especificar el orden o la frecuencia de máxima atenuación, fstop.

Dado que la máxima frecuencia a medir es:

$$f_{\max} = 40 \cdot 60 \text{ Hz} \cdot 1.1 = 2640 \text{ Hz}$$

Buscamos un filtro con las siguientes especificaciones:

*Equiripple*

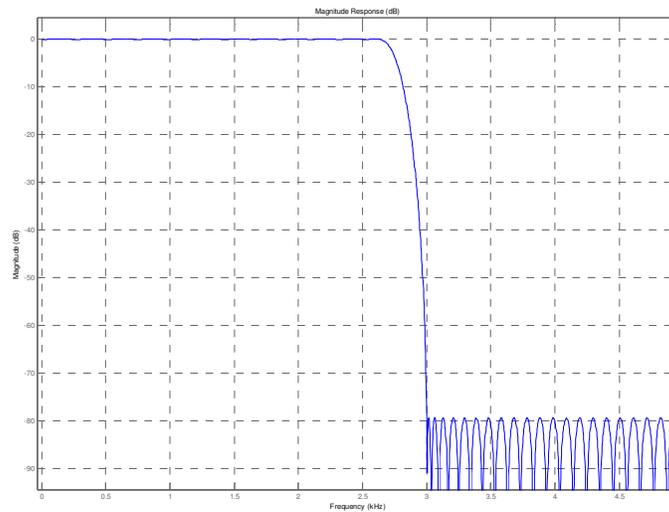
$$f_{\text{pass}} = 2640 \text{ Hz}$$

$$f_{\text{stop}} = 3000 \text{ Hz}$$

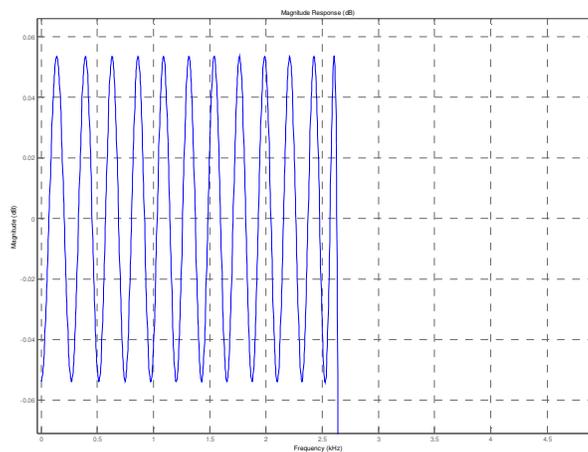
$$A_{\text{pass}} = 0.1 \text{ dB}$$

$$A_{\text{stop}} = 80 \text{ dB}$$

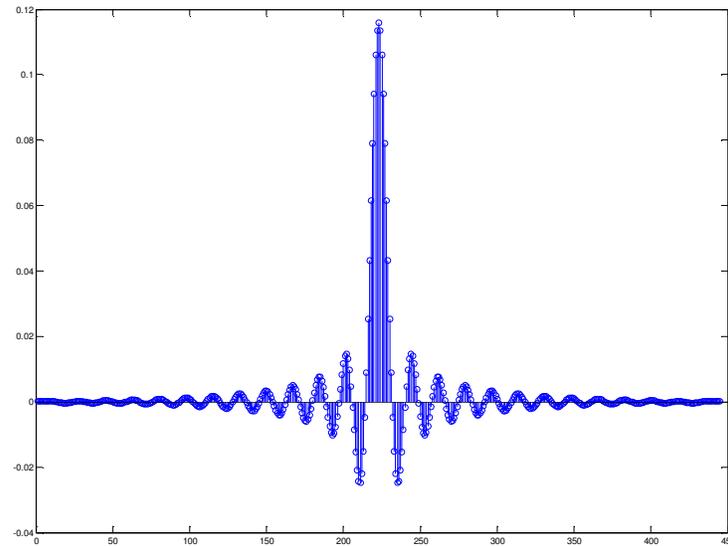
El orden obtenido fue de 444 coeficientes.



La planicidad se puede ver en detalle:



Se observa que el mayor error de amplitud es de 0.055dB, lo cual es despreciable. La respuesta al impulso del filtro es la siguiente:



Se desarrolló una rutina de convolución que lleva a cabo ambas convoluciones en paralelo, tomando ventaja del hecho de que el kernel es el mismo en ambos casos.

```
// Se inicializan los contadores de lazo.
r0 = [I0++] || r1.1 = W[I1--];
// esta vez hace 1 ciclo menos
lsetup( Loop1, Loop1.end ) lc0 = p3;
Loop1:
r4.1 = (a0 += r0.1 * r1.1 );
r4.h = (a1 += r0.h * r1.1 );
r0 = [I0++] ;
Loop1.end:
r1.1 = W[I1--];
```

Luego, Se aprovecho el hecho del paralelismo que tiene el Blackfin para tomar simultáneamente 2 valores de señal en una sola palabra de 32 bits, con una muestra en su parte alta y otra en su parte baja.

```
// Se inicializan los contadores de lazo.
r0 = [I0++] || r1.1 = W[I1--];
// esta vez hace 1 ciclo menos
lsetup( Loop1, Loop1.end ) lc0 = p3;
Loop1:
r4.1 = (a0 += r0.1 * r1.1 );
r4.h = (a1 += r0.h * r1.1 );
Loop1.end:
r0 = [I0++] || r1.1 = W[I1--];
```

Finalmente, dado que se puede paralelizar la operación de multiplicación y acumulación (dual MAC), junto la búsqueda de datos:

```
// Se inicializan los contadores de lazo.
  r0 = [I0++] || r1.1 = W[I1--];
// esta vez hace 1 ciclo menos
  lsetup( Loop1, Loop1.end ) lc0 = p3;
Loop1:
Loop1.end:
  r4.1 = (a0 += r0.1 * r1.1) , r4.h = (a1 += r0.h * r1.1) , \
  r0 = [I0++] || r1.1 = W[I1--];
```

Los tiempos que se obtuvieron para la ejecución de esta rutina, en cada caso fueron los siguientes:

Rutina	Tiempo (uS)
Pasos Individuales	4.42
Búsqueda paralelizada	3.218
Mac Paralelizada + Búsqueda Paralelizada	1.609

Para la medición de los tiempos se implementaron en el kit rutinas que encienden y apagan leds cada vez que inician y terminan las rutinas bajo análisis.

En la tabla anterior se mostró la mejora en los tiempos de ejecución al paralelizar las operaciones. Para ellos se comparó con el tiempo de ejecución en el caso de efectuar todas las operaciones en forma separada (búsqueda de la 1er muestra -> búsqueda de la 2da muestra -> 1er multiplicación y acumulación -> 2da multiplicación y acumulación) y con la búsqueda paralelizada de las muestras.

Una mejora que también se aplicó fue ubicar expresamente el buffer de muestras y el kernel en distintos bancos de memoria para minimizar los stall de lectura, pero la mejora de performance no es perceptible, ya que se obtuvo el mismo tiempo de ejecución de 1.608uS

La rutina de interrupción continua multiplicando y acumulando los valores de ambas señales, para la posterior obtención del valor de la potencia activa, y finaliza llevando a cabo el submuestreo. Para ello, toma una de cada M (8 en nuestro caso) muestras ya filtradas, las pondera de acuerdo a la ventana Flattop, y las almacena en el buffer de entrada de cada FFT (una para tensión y otra para corriente)

Una vez que las muestras submuestreadas llenan 2 buffers de 2048 muestras con las señales de corriente y tensión, ya ventaneadas, se llevan a cabo 2 FFTs para hallar la amplitud de cada armónica.

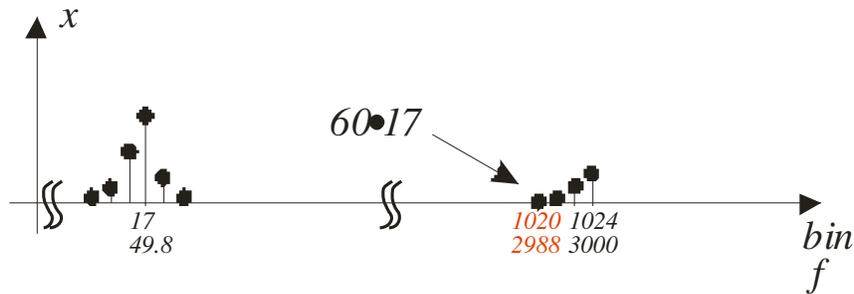
Se trato de simplificar el algoritmo FFT teniendo en cuenta que las señales de entrada solo tienen valores reales, que los valores de los coeficientes (twiddle) son iguales para ambas señales y que, como en el caso de la convolución podemos paralelizar la busca de datos. Sin embargo, dicha paralelización es virtualmente imposible para la FFT, ya que el algoritmo ya

utiliza gran cantidad de recursos y resulta más conveniente aplicarlo en forma consecutiva para cada señal. El algoritmo tardó 3.348mS en ejecutarse. Esto muestra que es imprescindible ejecutar las FFT en el main y no en la rutina de interrupción, ya que si hacemos eso perderemos muestras, ya que no podremos atender las interrupciones que se suceden cada 20uS aprox. (1/48000Hz).

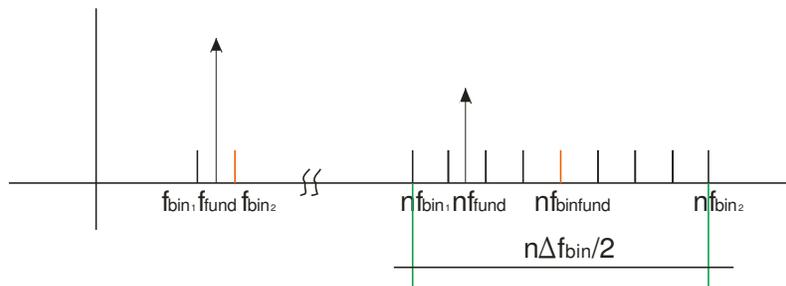
Otra cuestión que surge del algoritmo FFT es que la implementación puede utilizar menos memoria mediante la superposición del vector de muestras de entrada y el de salida, a costo de una menor performance. Hay que tener en cuenta que los datos de entrada para la FFT con entrada real tienen la mitad de la longitud que los de salida, pues estos serán complejos.

Una vez obtenidas las FFT debemos obtener los valores amplitud de cada armónica. Para ello suponemos, como en general sucede, que la mayor armónica es la fundamental. De esta forma buscamos la mayor armónica y el índice que la apunta en el vector resultado de la FFT (considerando valores absolutos). De esta forma, en la vecindad de las posiciones del vector con índice múltiplo entero del bin de la fundamental podremos encontrar las otras armónicas. Para analizar a que nos referimos con vecindad hay que considerar que el peor caso se puede dar cuando la fundamental cae en la mitad exacta entre 2 bins, con lo cual, cada múltiplo de la misma incorporará dicho error multiplicado por el numero de armónica.

Por ejemplo para 50Hz, que está cerca del bin 17 (49.81Hz), la 60ava armónicas (3000Hz) corresponde al bin 1024, y no en el  $60 \times 17 = 1020$ :



El algoritmo tendrá que considerar este efecto. Para ello, una vez hallada la frecuencia fundamental, los máximos correspondientes a las otras armónicas se buscaran en una vecindad del armónico múltiplo entero de la bin fundamental y una cantidad de bins por exceso y por defecto que sumen en total la mitad del número del armónico, como se grafica a continuación:



Si consideramos que los bins de las frecuencias armónicas (múltiplos de la fundamental) están en promedio cada 16 bin para el peor caso, que se da con la menor frecuencia de fundamental posible, 45Hz, en las armónicas más alejadas, en especial, a partir de 32ava armónica, el intervalo de búsqueda incluirá a la armónica anterior:

$$n = 32 \rightarrow \frac{n\Delta f}{2} = \frac{32 \cdot 2.9296\text{Hz}}{2} = 46.875 > f_0 = 45\text{Hz}$$

El intervalo entonces tendrá que iniciarse en el bin adyacente al correspondiente al máximo anterior para evitar tomar un dato falso.

Si bien es cierto que la ventana Flattop no es buena para la detección del bin mas exacto, debemos tener en cuenta que dada su excelente planicidad y el ancho de su respuesta en frecuencia, no será necesario saber el bin exacto, sino su amplitud. Dado que, como se mostro antes, luego del 9no bin la amplitud de la respuesta en frecuencia de la Flattop queda atenuada en 60dB, o sea, 1000 veces, la influencia de un armónico sobre otro es mínima. En nuestra aplicación, lo más importante es la amplitud de cada armónico y no su ubicación.

A la hora de obtener los valores pico de cada armónica hay que considerar el efecto del ventaneo en las amplitudes, como se demostró al hablar de las ventanas. Dicho valor viene dado por:

$$A = \frac{1}{N} \sum_{n=0}^{N-1} w(n)$$

Y las amplitudes reales de cada armónica serán:

$$V_n = \frac{V'_n}{A}$$

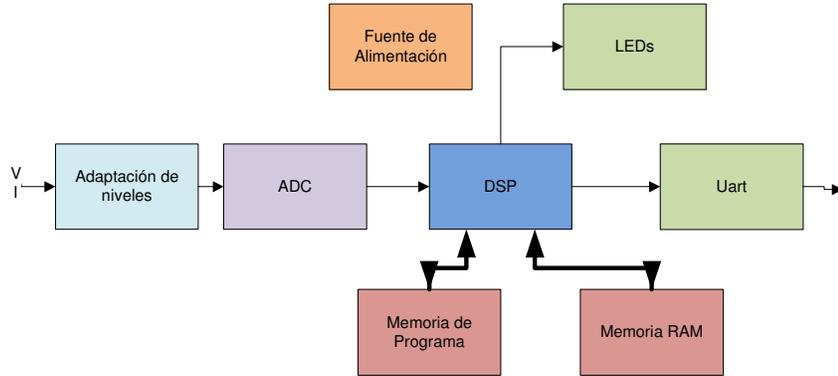
Para la ventana de Flattop se obtiene:

$$A = 0.2154$$

$$\frac{1}{A} = 4.6409$$

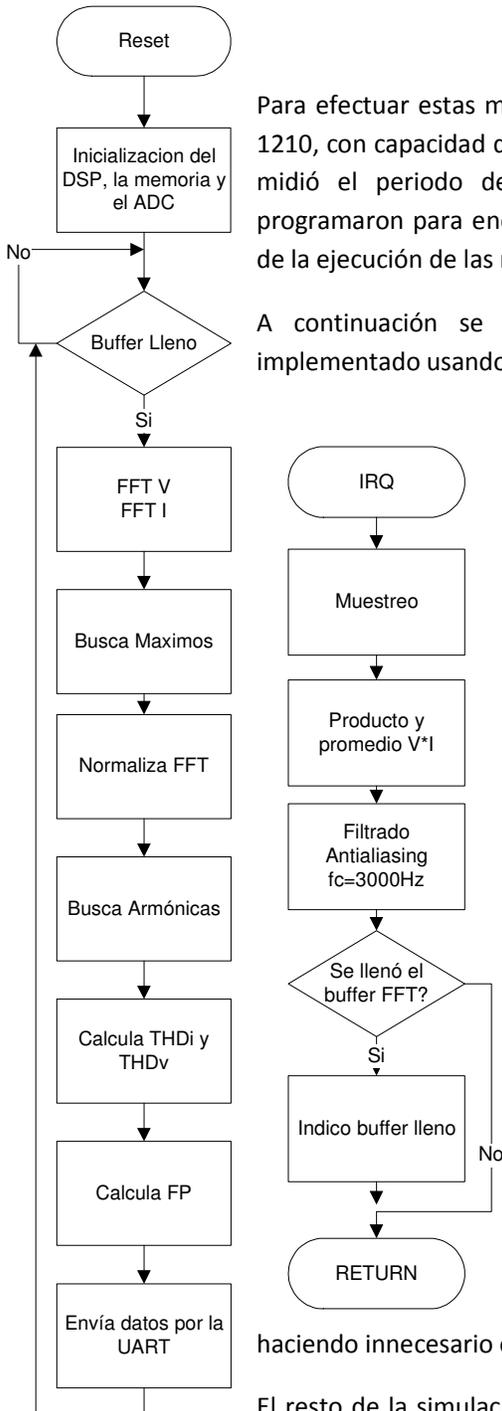
Para los cálculos de potencia ( $P_{dc}$  y  $V_1 \cdot I_1$ ), además de la corrección anterior, se deben ajustar los valores ya convertidos según sendas constantes que tendrán en cuenta el hardware y su relación de transferencia de tensión. Estos valores ajustarán los cálculos para hallar valores de potencia promedio reales. Estas constantes considerarán la ganancia (relación de reducción) de la etapa adaptadora de señal desde la punta de prueba hasta la entrada del ADC y la resolución del mismo. En el kit para una amplitud de +3.3V el ADC devuelve un valor digital de 0x7FFF. Sin embargo esto no es necesario a la hora de implementar los cálculos de THD ya que normalizando todos los valores de las armónicas respecto a la armónica de mayor amplitud ya podemos hallar el valor buscado, ya que no importan los valores absolutos de las amplitudes, sino su relación respecto a la fundamental.

Las rutinas de soft que se encargan de llevar a cabo los pasos necesarios para el cálculo de la THD y el FP tardan unos 1.642mS.



Para efectuar estas mediciones se utilizo un osciloscopio Tectroniks TDS-1210, con capacidad de memorizar los datos en pantalla. Con el mismo se midió el periodo de encendido de los leds del kit, los cuales se programaron para encender y apagar al principio y fin, respectivamente, de la ejecución de las rutinas en cuestión.

A continuación se puede ver el diagrama de bloques del Hard implementado usando el kit de Analog Devices



haciendo innecesario el cálculo de dichas estructuras en tiempo real.

El resto de la simulación se lleva a cabo en el ambiente de desarrollo de

Los diagramas de flujo del software implementado son los siguientes:

### 8. Simulación del algoritmo

Para la simulación de los algoritmos se utilizo el software MATLAB para la generación de las señales. Dado que se van a analizar señales periódicas, basta con crear un solo período de la misma y luego introducirlo en forma secuencial al algoritmo.

Se usan los objetos FiMath del Matlab, específicos para utilizar matemática de punto fijo, y se imprimen los resultados en un archivo, usando el formato de datos típico de un compilador C.

También se crearon los coeficientes de la ventana de Blackman-Harris y los coeficientes twiddle.

El archivo creado es luego incluido para inicializar vectores en la memoria del DSP,

Analog Devices, el visual-DSP++.

El mismo nos permite graficar las señales de entrada y salida, además de visualizar el estado de las variables del sistema, junto con los valores de los registros y la memoria interna del DSP.

Dado que no se pueden simular las interrupciones del ADC, se genera un loop infinito dentro del cual ejecutamos el algoritmo de cálculo. Las diferencias con la realidad surgirán del hecho de que el intervalo de tiempo entre interrupciones puede llegar a ser inferior al tiempo que toma el algoritmo en su peor condición, que se da cada vez que se llena el buffer de entrada del algoritmo FFT y se lleva a cabo dicha operación. Esto solo será observable en la prueba del prototipo físico.

Dentro de dicho lazo, cada vez que hemos recorrido un ciclo de la señal de entrada, volvemos al comienzo, evitando discontinuidades y manteniendo la periodicidad de la señal

Para la simulación, utilizamos el siguiente par de señales:

$$v(t) = \sin(\omega t)$$

$$i(t) = 0.5 \sin(\omega t) + 0.25 \sin(2\omega t) + 0.25 \sin(3\omega t)$$

Para dichas señales, se debe obtener:

$$THD_v = 0$$

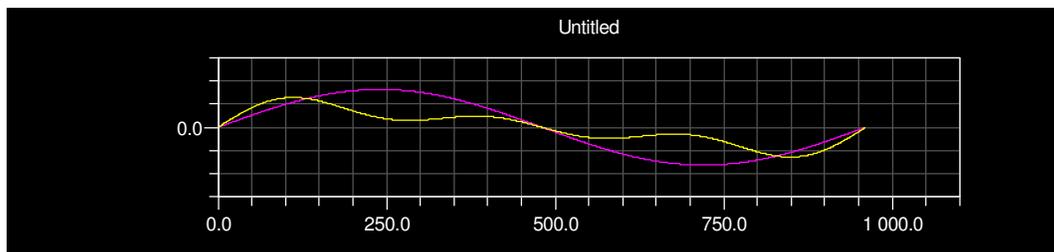
$$THD_i = \sqrt{\frac{V_2^2 + V_3^2}{V_1^2 + V_2^2 + V_3^2}} = \sqrt{\frac{0.25^2 + 0.25^2}{0.5^2 + 0.25^2 + 0.25^2}} = 0.57735 = 57.735\%$$

$$FP = \frac{P}{S} = \frac{\frac{1}{N} \sum_{n=0}^{N-1} v(n) i(n)}{\frac{\hat{V}_1 \hat{I}_1}{2}} \sqrt{1 - THD_v^2} \sqrt{1 - THD_i^2} = \frac{\sum_{n=1}^{\infty} \frac{\hat{V}_n \hat{I}_n}{2} \cos \varphi_n}{\frac{\hat{V}_1 \hat{I}_1}{2}} \sqrt{1 - THD_v^2} \sqrt{1 - THD_i^2}$$

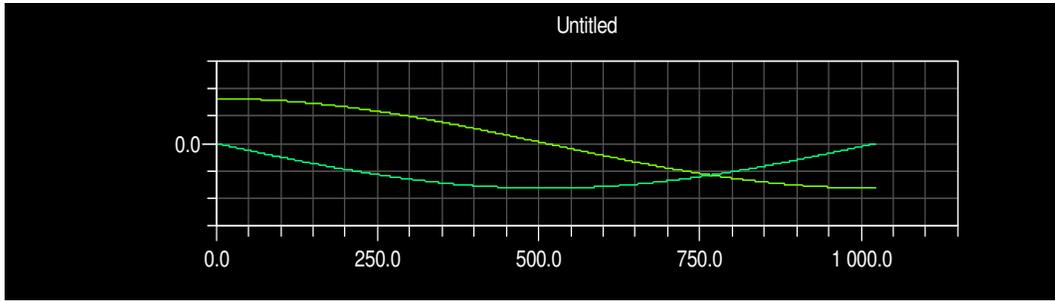
$$FP = \cos \varphi_n \sqrt{1 - THD_v^2} \sqrt{1 - THD_i^2} = 1 \cdot \sqrt{1 - (0.57735)^2} = 0.816497$$

Estos valores se verificaron en Matlab a partir de la definición de cada una de esas magnitudes.

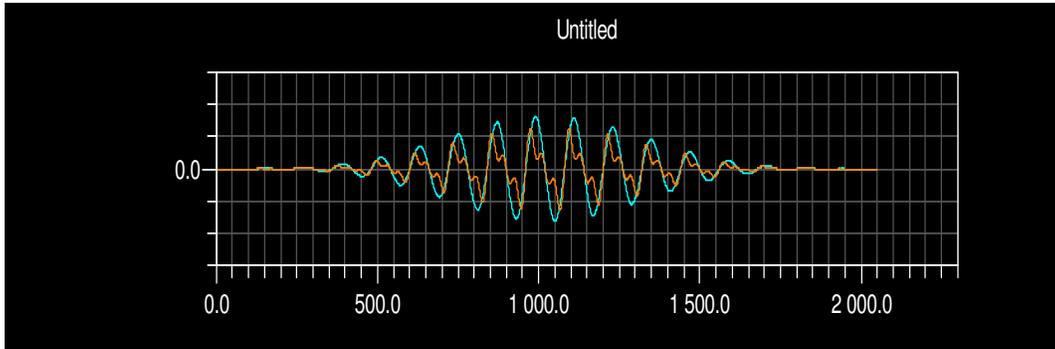
Las señales de tensión y corriente son:



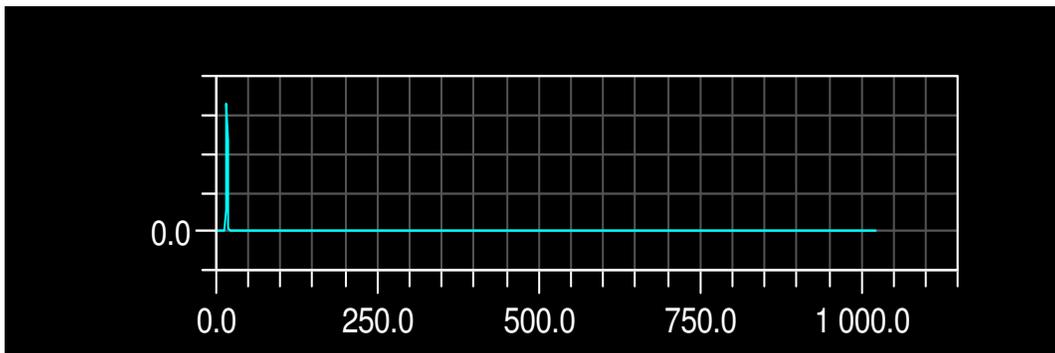
Los coeficientes de la mariposa son:



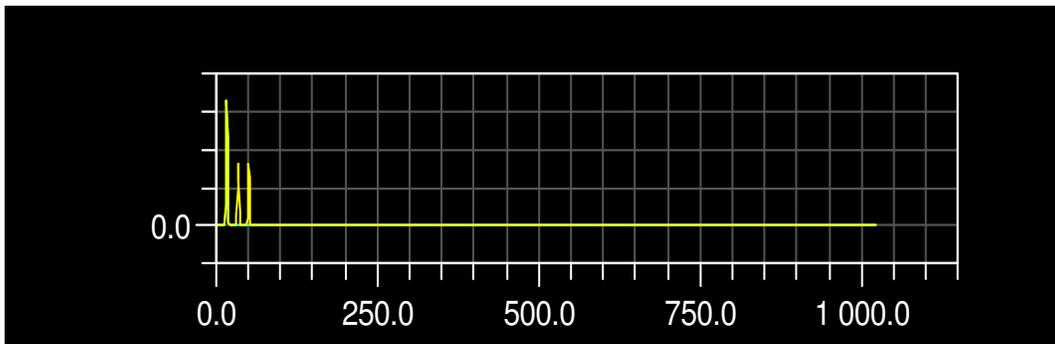
El buffer de entrada a la FFT, aplicada la ventana, queda así:



La FFT, con el ajuste de escala queda, para la tensión:



Y para la corriente:



La simulación del cálculo dio como resultado:

$$FP = 0.8254$$

$$THD_V = 0.0004$$

$$THD_I = 0.699$$

Lo que dio un error de:

$$\varepsilon_{FP} = \frac{|0.8254 - 0.81|}{0.8254} 100\% = 1.8\%$$

$\Delta THD_V = 0.00012$  (El calculo del error relativo no corresponde en este caso)

$$\varepsilon_{THD_I} = \frac{|0.699 - 0.707|}{0.699} 100\% = 1.14\%$$

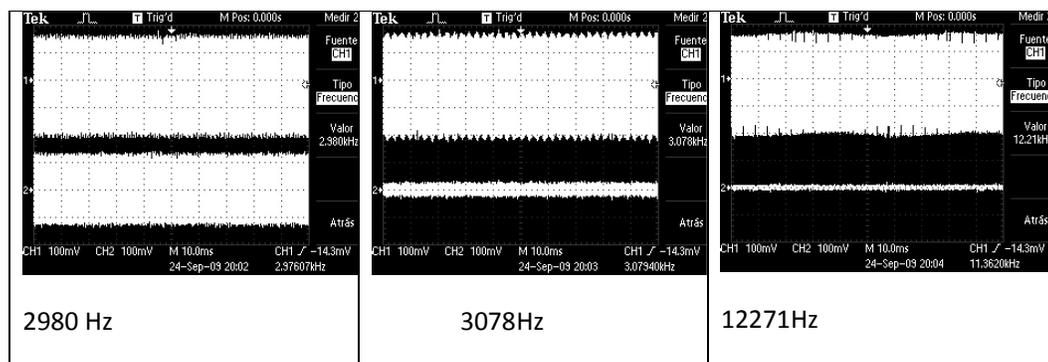
Donde no calculamos el error relativo de la distorsión de tensión, ya que cuando una magnitud se anula, no se puede relativizar el error.

### 9. Resultados prácticos

Para analizar la performance práctica del circuito se llevan a cabo las siguientes comprobaciones:

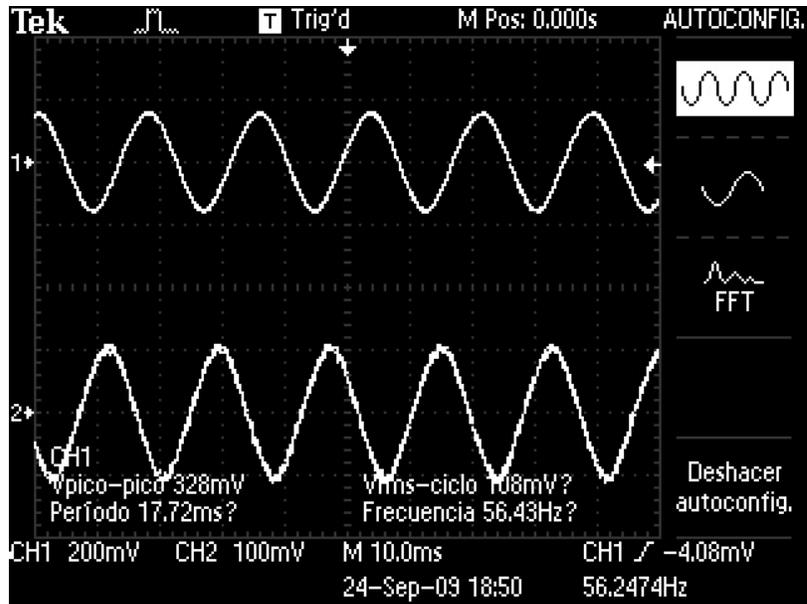
- Funcionamiento de los filtros antialiasing

El soft envía a la salida analógica de la placa de desarrollo la señal ya filtrada. Debemos introducir señales de frecuencias superiores a la de corte y comprobar que no hay salida alguna.



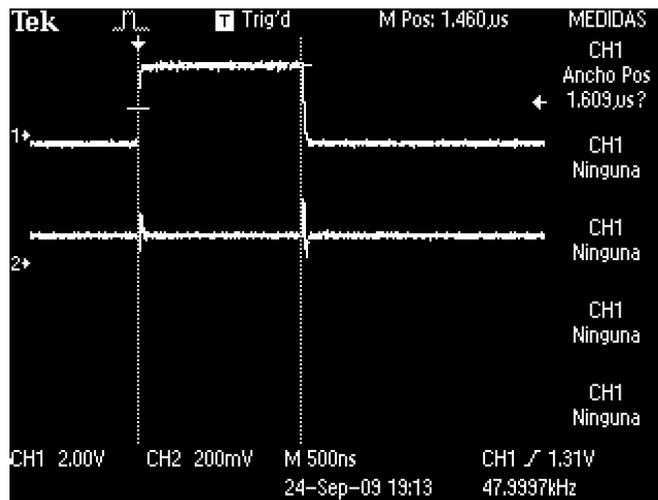
- Resultados de las mediciones

Aplicando señales de contenido armónico conocido, debemos comparar las mediciones obtenidas con la realidad. Se midieron señales senoidales, cuadradas y triangulares (de 50% de ciclo de trabajo). A continuación se observa la salida de los filtros y las mediciones obtenidas.

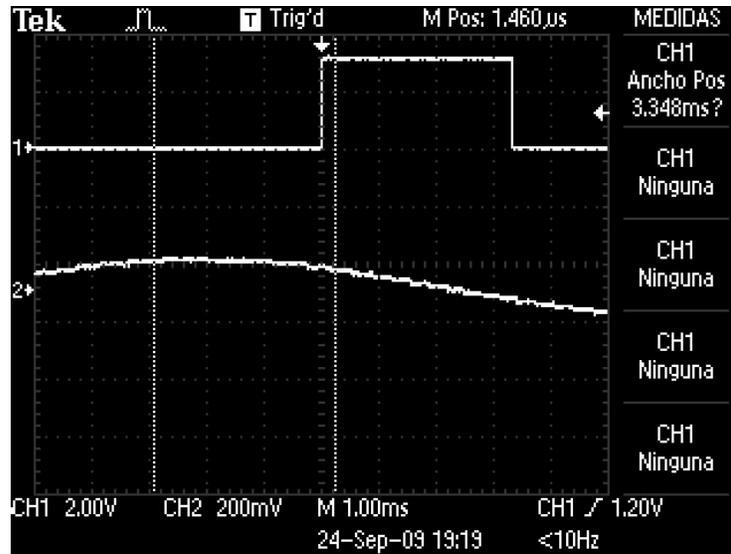


Aquí vemos como se mantiene la forma de la señal y como se introdujo un retraso, debido al filtrado digital

En la siguiente figura se observa la duración de la rutina de convolucion en ambos canales:



Y la siguiente muestra la duración de la rutina de FFT (un solo canal)



Para probar los valores medidos de THD partimos de 2 casos simulables fácilmente con generadores de señal comunes.

Para la señal cuadrada los coeficientes de Fourier dan

$$a_0 = 0 \text{ (por ser totalmente alternada)}$$

$$a_n = 0 \text{ (por ser impar)}$$

$$b_n = \frac{4X_{\max}}{n\pi}$$

Donde  $X_{\max}$  es la amplitud pico de la señal. El valor de THD será entonces

$$THD = \sqrt{\frac{X_{rms}^2 - X_1^2}{X_1^2}}$$

$$\hat{X}_1 = \frac{4X_{\max}}{\pi} \Rightarrow X_1 = \frac{\hat{X}_1}{\sqrt{2}} \Rightarrow X_1^2 = \left(\frac{\frac{4X_{\max}}{\pi}}{\sqrt{2}}\right)^2 = \frac{16X_{\max}^2}{2\pi^2} = \frac{8X_{\max}^2}{\pi^2}$$

$$X_{rms} = X_{\max}$$

$$THD = \sqrt{\frac{(X_{\max})^2 - \frac{8X_{\max}^2}{\pi^2}}{(X_{\max})^2}} = \sqrt{1 - \frac{8X_{\max}^2}{\pi^2 (X_{\max})^2}} = \sqrt{1 - \frac{8}{\pi^2}} = 0.435236$$

Para la señal triangular:

$a_0 = 0$  (por ser totalmente alternada)

$b_n = 0$  (por ser par)

$$a_n = \frac{2A\tau}{T} \left( \frac{\sin\left(\frac{n\omega_0\tau}{2\pi}\right)}{\frac{n\omega_0\tau}{2\pi}} \right)^2 = \frac{2AT}{T} \left( \frac{\sin\left(\frac{n\frac{2\pi T}{2}}{2}\right)}{\frac{n\frac{2\pi T}{2}}{2}} \right)^2 = 2A \left( \frac{\sin\left(\frac{n\pi}{2}\right)}{\frac{n\pi}{2}} \right)^2 = \frac{8A}{n^2\pi^2}$$

$$THD = \sqrt{\frac{X_{rms}^2 - X_1^2}{X_1^2}}$$

$$\hat{X}_1 = \frac{8X_{max}}{\pi^2} \Rightarrow X_1 = \frac{\hat{X}_1}{\sqrt{2}} \Rightarrow X_1^2 = \left( \frac{8X_{max}}{\pi^2} \right)^2 = \frac{64X_{max}^2}{2\pi^4} = \frac{32X_{max}^2}{\pi^4}$$

$$X_{rms} = \frac{X_{max}}{\sqrt{3}}$$

$$THD = \sqrt{\frac{\left(\frac{X_{max}}{\sqrt{3}}\right)^2 - \frac{32X_{max}^2}{\pi^4}}{\left(\frac{X_{max}}{\sqrt{3}}\right)^2}} = \sqrt{1 - \frac{32X_{max}^2}{\pi^4} \frac{3}{X_{max}^2}} = \sqrt{1 - \frac{96}{\pi^4}} = 0.120273$$

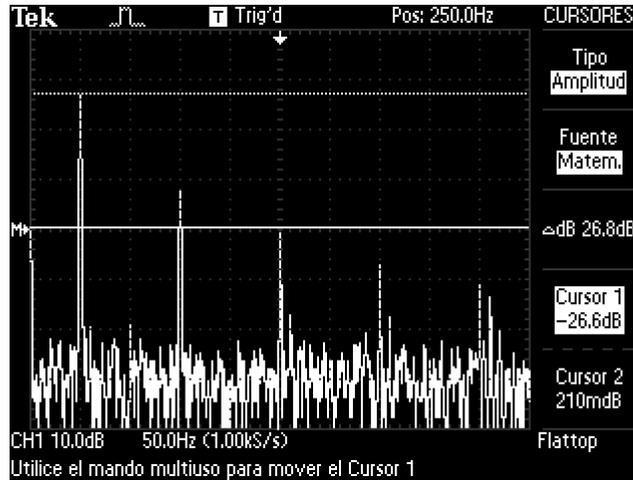
Los valores del FP serán para la señal cuadrada y la triangular respectivamente

$$FP = \frac{P}{S} = \frac{\frac{1}{N} \sum_{n=0}^{N-1} v(n)i(n)}{\frac{\hat{V}_1 \hat{I}_1}{2}} \sqrt{1-THD_v^2} \sqrt{1-THD_i^2} = \frac{\frac{2}{\pi}}{\frac{1}{2} \cdot \frac{4}{\pi}} \sqrt{1} \sqrt{1-0.435236^2} = 0.900316$$

$$FP = \frac{P}{S} = \frac{\frac{1}{N} \sum_{n=0}^{N-1} v(n)i(n)}{\frac{\hat{V}_1 \hat{I}_1}{2}} \sqrt{1-THD_v^2} \sqrt{1-THD_i^2} = \frac{\frac{4}{\pi^2}}{\frac{1}{2} \cdot \frac{8}{\pi^2}} \sqrt{1} \sqrt{1-0.120273^2} = 0.992741$$

Señal	Teórico		Medido	
	THD	FP	THD	FP
I + V Senoidales	0	1	0.001	1.00
I Cuadrada + V Senoidal	0.435236	0.900316	0.43333	0.89
I Triangular + V Senoidal	0.120273	0.992741	0.1205	0.9923
I Senoidal + V Senoidal 45°	0	0.707	0.0261	0.7348

Para la señal triangular se verificó la amplitud de las armónicas usando la función FFT del DSO:



Se observa la presencia únicamente de armónicas impares. La imagen muestra el resultado para la quinta armónica:

$$-26.8dB = 20 \log \left( \frac{V_1}{V_5} \right) \Rightarrow \frac{V_1}{V_5} = 10^{-\frac{26.8}{20}} = 0.045$$

$$\frac{8}{(5^2)(\pi^2)} = 0.033$$

Donde podemos atribuir la diferencia entre el valor teórico y el medido a los errores numéricos del procesamiento.

## 10. Conclusiones

Este tipo de sistemas de medición, que a simple vista parecen consistir simplemente de la implementación de una FFT, conllevan muchas particularidades que los complican. Sin embargo, son perfectamente implementables con costos computacionales relativamente bajos, dependiendo, como siempre, de la precisión y velocidad deseada.

La amplitud obtenida se condice con la medida sobre la entrada del ADC

## 11. Futuro del proyecto

Otra parte importante para que este sistema pueda integrarse al mundo real es la interface de hardware. La misma debe acondicionar los niveles de tensión y corriente hallados en las instalaciones a monitorear, garantizando linealidad en la conversión de niveles, un ancho de banda que permita el paso del contenido armónico, y por sobre todo, una aislación eléctrica segura entre el corazón del sistema, la interface del usuario y las líneas de potencia. Así también, el desarrollo de una interface gráfica amena para el usuario le daría más ventajas a la implementación a la hora de su promoción

## 12. Bibliografía

1. ARMENTANO, CRAIEM, "Análisis de sistemas lineales", Ed. Nuevos Tiempos, p. 198.
2. ARMENTANO, CRAIEM, "Análisis de sistemas lineales", Ed. Nuevos Tiempos, p. 215
3. ARMENTANO, CRAIEM, "Análisis de sistemas lineales", Ed. Nuevos Tiempos, p. 239

4. ARMENTANO, CRAIEM, "Análisis de sistemas lineales", Ed. Nuevos Tiempos, p. 232
5. ARMENTANO, CRAIEM, "Análisis de sistemas lineales", Ed. Nuevos Tiempos, p. 198
6. ARMENTANO, CRAIEM, "Análisis de sistemas lineales", Ed. Nuevos Tiempos, p. 199
7. ARMENTANO, CRAIEM, "Análisis de sistemas lineales", Ed. Nuevos Tiempos, p. 200
8. ARMENTANO, CRAIEM, "Análisis de sistemas lineales", Ed. Nuevos Tiempos, p. 200
9. RASHID, MUHAMMAD H., "Electrónica de potencia", Prentice Hall, p. 71
10. ARMENTANO, CRAIEM, "Análisis de sistemas lineales", Ed. Nuevos Tiempos, p. 240, p. 312
11. ARMENTANO, CRAIEM, "Análisis de sistemas lineales", Ed. Nuevos Tiempos, p. 233, p. 313
12. ARMENTANO, CRAIEM, "Análisis de sistemas lineales", Ed. Nuevos Tiempos, p. 314
13. LYONS, RICHARD G. "Understanding Digital Signal Processing", Prentice Hall, p.50
14. LYONS, RICHARD G. "Understanding Digital Signal Processing", Prentice Hall, p.53
15. LYONS, RICHARD G. "Understanding Digital Signal Processing", Prentice Hall, p.63
16. LYONS, RICHARD G. "Understanding Digital Signal Processing", Prentice Hall, p.71
17. LYONS, RICHARD G. "Understanding Digital Signal Processing", Prentice Hall, Cap. 3.8
18. LYONS, RICHARD G. "Understanding Digital Signal Processing", Prentice Hall, Cap 3.9
19. FREDRIC J. HARRIS, "On the use of windows for harmonic analysis with the discrete Fourier transform", Proceedings of the IEEE, VOL. 66, Nº 1, enero 1978
20. LASZLO HARS, "Optimum DFT Window Design", Panasonic Information and Networking Technologies Laboratory
21. LYONS, RICHARD G. "Understanding Digital Signal Processing", Prentice Hall, p. 76
22. Application Note 014, lds-group, [www.lds-group.com](http://www.lds-group.com)