

# FILTRO DIGITAL ADAPTABLE PARA SISTEMAS EMBEBIDOS.

Requena, Joaquín

Rapallini, José A.

CeTAD- Centro de Técnicas Analógicas Digitales. Fac. de Ing. UNLP. Argentina.

[jrequena@barcala.ing.unlp.edu.ar](mailto:jrequena@barcala.ing.unlp.edu.ar)

[josrap@ing.unlp.edu.ar](mailto:josrap@ing.unlp.edu.ar)

## RESUMEN

Las señales pueden contener distorsiones y ruido que dificultan la lectura en forma correcta de los parámetros requeridos por el usuario, motivo por el cual en la mayoría de los equipos de medida se debe optimizar la medición a través de filtros digitales que procesen la señal [4, 5].

En este trabajo se estudian y diseñan filtros digitales con coeficientes variables, realizando la implementación con circuitos lógicos programables, obteniéndose un dispositivo digital para el análisis de datos muy flexible.

La metodología de diseño utilizada (codiseño HARD/SOFT) posibilita la fácil modificación de las características, adaptándolo a sistemas embebidos específicos, brindando la posibilidad inmediata de la implementación del circuito en una ASIC, con la ayuda de las herramientas adecuadas (para nuestro caso CADENCE).

Por último se presentan los resultados y conclusiones de la implementación y las distintas posibilidades de uso del dispositivo.

## 1. ANÁLISIS DE ALTERNATIVAS DEL DISEÑO.

Del análisis de las diferentes realizaciones de un filtro FIR, se ve cuales son las características de cada estructura al momento de llevarlas a la implementación [2], es por eso que según los objetivos que se persiguen se deben plantear las diferentes tecnologías en las que se puede llegar a implementar el circuito del filtro digital.

En una primera observación parecería obvio la utilización de un procesador DSP (*Digital Signal Processor*) como la mejor opción al diseño. A diferencia de los microcontroladores y microprocesadores que abarcan un contexto mas general, los DSP son procesadores con características específicas para el tratamientos de señales. Sin embargo, aunque los procesadores son programables a través de software, la arquitectura del hardware del procesador DSP no es flexible.

Los circuitos lógicos programables, PLD (*Programmable Logic Device*), FPGA (*Fields Programmable Gate Arrays*), suministran una solución reconfigurable y eficiente para implementar aplicaciones de procesamiento de señales digitales. Estos circuitos pueden alcanzar una mayor tasa de transferencia y una mayor potencia de procesamiento de datos que los procesadores DSP.

Debido a que los circuitos lógicos programables pueden ser reconfigurados en hardware, estos ofrecen un

hardware completamente dedicado o específico para implementar varias aplicaciones de procesamiento de señales digitales. Por lo tanto, sistemas DSP implementados en FPGAs pueden tener un arquitectura específica, bloques aceleradores de hardware específicos y un número variable de bloques multiplicadores-acumuladores (MAC), por lo tanto los FPGAs ofrecen una oportunidad para acelerar un procesamiento digital con respecto a un procesador DSP tradicional.

Por ejemplo, en muchos sistemas de procesamiento de señales el filtro FIR es usado para llevar a cabo diferentes tareas tales como el acondicionamiento de la señal, antialiasing, selección de la banda, interpolación y convolución. Sin embargo, un número limitado de circuitos filtros FIR están disponible en el mercado y estos circuitos frecuentemente limitan el desempeño del sistema. Por lo tanto, los circuito FPGAS son una alternativa ideal para implementar filtros FIR.

Nuestro laboratorio cuenta en particular, con dispositivos FLEX de Altera [3] que nos permitieron implementar filtros FIR. Un circuito FLEX permite implementar una o mas funciones de filtrado critico para una aplicación basada en un procesador DSP, liberando al procesador para que este lleve a cabo las operaciones algorítmicas complejas.

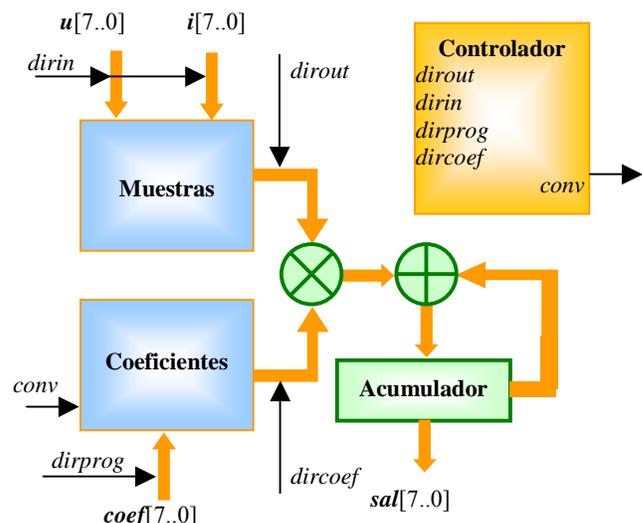


Figura 1

Además, en los dispositivos lógicos programables, el diseño del circuito del filtro digital se puede implementar mediante lenguajes de descripción de hardware. Por lo tanto, se tiene la posibilidad de evaluar diferentes tipos de diseño y poder dejar establecidos conceptos para que, en un proyecto posterior de microelectrónica, se desarrolle un circuito de aplicación específica (ASIC, *Application Specific Integrated Circuit*).

## 2. REALIZACIÓN SOBRE UNA FPGA

Una de las características más destacada de la *FLEX* es la capacidad de desarrollar bloques de almacenamientos que pueden llegar a reducir las exigencias de hardware en un diseño. A partir de esta capacidad se realiza un sistema para el cálculo del algoritmo que representa al filtro digital. Como se observa en la figura 1, el sistema esta constituido por diferentes bloques con propósitos específicos cada uno: Bloque de almacenamiento de coeficientes; Bloque de almacenamiento de las muestras; Multiplicador; Sumador; Acumulador; Controlador.

El proceso del sistema consiste en realizar la convolución, el *multiplicador* esta constantemente multiplicando una muestra y un coeficiente, cuales de ellos se encarga el direccionamiento del *controlador*. El *sumador* y el *acumulador* se encargan de sumar el resultado a la suma anterior. Todo el sistema está gobernado por una señal de reloj *clock* para lograr la sincronización y poder establecer en forma adecuada los tiempos de cálculo.

## 3. DESCRIPCIÓN HDL DEL FILTRO DIGITAL IMPLEMENTADO

La descripción se desarrolló en lenguaje AHDL (*Altera Hardware Description Language*) [2], corresponde a un filtro de 32 coeficientes con 32 muestras por ciclo y una señal de reloj de  $8MHz$ , es decir  $clock = 125ns$ . La señal que está ingresando al sistema tiene una frecuencia de  $f = 50Hz$ , el periodo de la señal es de  $T = 20ms$ , a una frecuencia de muestreo de  $1600Hz$  se obtiene una muestra cada

$$T = 20ms \cdot \frac{50Hz}{1600Hz} = 625\mu s$$

La resolución a tener en cuenta para los datos de entrada es de 8 bits. Todos estos parámetros están basados en los estudios realizados en los trabajos anteriores, correspondientes al análisis de filtros tipos Fourier, Seno y

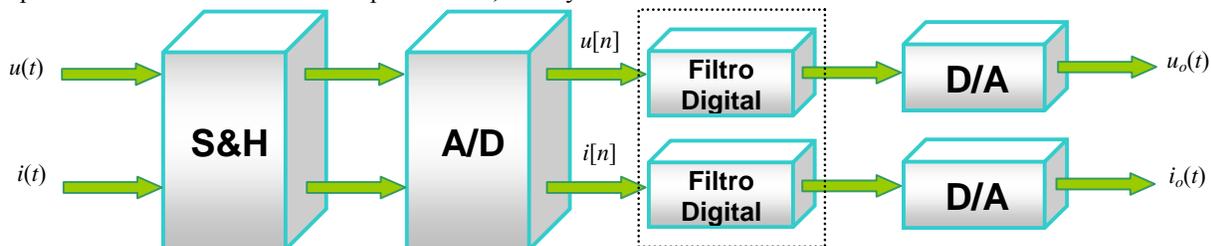


Figura 2

Coseno, para ensayos de sistemas de potencias eléctricos (1, 2, 3), trabajos que se basan en la obtención de dos señales (tensión y corriente) libres ruido para su combinación y posterior procesamiento para la obtención de una impedancia de cortocircuito.

Para la implementación de cada bloque se utilizan las megafunciones. Son funciones de tal utilidad que los fabricantes de dispositivos programables acordaron en definir elementos similares con una misma sintaxis, creando una biblioteca de módulos parametrizados o LPMs (*Library of Parameterized Modules*), que incluye: Compuertas y bloques funcionales simples: Funciones aritméticas y Elementos de memoria.

Estos módulos poseen gran cantidad de opciones parametrizables, lo que hace innecesario conocer una enorme variedad de productos y cada fabricante de circuitos integrados ha puesto enormes esfuerzos de ingeniería para que los resultados de su uso sean óptimos, siendo por eso recomendable su empleo en cualquier diseño.

## 4. DESARROLLO DEL SISTEMA DIGITAL DE FILTRADO

A partir de las técnicas estudiadas para diseñar filtros digitales FIR, se procedió a aplicarlas en un sistema digital de procesamiento de señales. Como herramienta de diseño se utilizó el entorno MATLAB, que permitió los cálculos para evaluar los filtros, y MAX+plus II, en el cual se pudo desarrollar las etapas de *descripción* y *simulación funcional* del sistema.

Como se comento anteriormente, se decidió implementar el sistema basándose en un dispositivo *FLEX*. Se desarrolló un programa escrito en AHDL apoyándose en la descripción realizada por las librerías parametrizadas (LPM).

Todo el trabajo se realizó sobre un esquema convencional de adquisición de datos compuesto por el muestreo, conversión analógica digital, procesamiento de las señales y una etapa de conversión digital a analógica para observar el resultado obtenido. (figura 2)

### 4.1. Unidad procesadora.

Todo el sistema se centra en el bloque que implementa los filtros digitales, el cual se desarrolla sobre el dispositivo de lógica programable *FLEX10K10*.

La verificación de la configuración de la *FPGA* se realizó

mediante el equipo de prueba HP16500B, donde se verificó su funcionamiento acorde a los cálculos realizados en el entorno MATLAB y con sus coeficientes configurados externamente por el usuario.

Se configura la FLEX10K10 tal que realice el filtrado de dos señales y además se encarga de comunicarle a los dispositivos externos cuando operar.

Lo más significativo del diseño, es la reutilización de las operaciones que constituyen al filtro, es decir que el cálculo de convolución que se realiza para cada muestra nueva que ingresa al sistema, se hace en forma secuencial. Por lo tanto primero se hace el cálculo para una señal y una vez finalizado se procede con el mismo cálculo para la otra señal. Esto es posible debido a la alta velocidad de procesamiento que ofrece la FLEX10K10, logrando ahorrar una importante cantidad de hardware.

El dispositivo de lógica programable FLEX10K10 se encuentra en una placa de desarrollo UPx10K10, que posee un cristal que genera una señal de reloj de 125ns.

#### 4.2. La adquisición de los datos. Sample & Hold.

Esta etapa se encarga de mantener los datos durante un tiempo determinado para obtener la información a procesar por el convertor A/D.

Básicamente, un circuito sample & hold se comporta como una memoria analógica que almacena un nivel de tensión dado de muy bajas pérdidas. En este caso se utilizan dos LF398, porque es necesario que las dos señales sean muestreadas al mismo tiempo ya que en un proceso posterior al filtrado es conveniente realizar los cálculos en el mismo instante de tiempo.

Una vez que cada LF398 (sample & hold) recibe la señal *conv* en alto (5V durante 2µs), en un tiempo estimado de 5µs trasladan a su salida el valor tomado del nivel de la señal analógica de entrada.

#### 4.3. Conversión analógica/digital.

Existen diversos circuitos integrados que realizan diferentes técnicas la conversión, se escogió para esta etapa un microcontrolador PIC16F877 de Microchip, que además de tener integrado el convertor analógico-digital de aproximaciones sucesivas de 10 bit de resolución, brinda la posibilidad de utilizarlo para realizar operaciones posteriores al filtrado de la señal (ver Conclusiones)

Para habilitar la conversión se utiliza un pin del dispositivo (RA4) que al recibir un pulso desde la FPGA comenzará la conversión A/D. El tiempo de conversión es  $T_{AD} = 2T_{OSC} = 2\mu s$  y el resultado es justificado hacia la izquierda, es decir que los 8 bits más significativos se cargan en el registro ADRESH.

Al PIC16F877 también llega la señal *conv*, a partir de ese momento se hace un retardo de 10µs para permitir que cada LF358 adquiera su muestra. Luego de la espera empieza la conversión.

#### 4.4. Obteniendo resultados. Conversión digital/analógica.

Esta etapa es la dedicada a recibir la información digital y convertirla en una señal analógica para la visualizar en el osciloscopio el resultado obtenido. En particular se utiliza el dispositivo DAC0808 de National, que es un convertor digital-analógico de 8 bits de resolución con un circuito de amplificación que convierte una tensión de referencia en una corriente y un circuito espejo de corriente para alimentar el circuito escalera R-2R.

### 5. VERIFICACIÓN DEL FILTRO.

Para verificar si el sistema funciona con un determinado filtro, se propone lograr filtrar los ruidos que contaminan señales que tienen una componente fundamental de 50Hz. Para ello se utilizan los coeficientes correspondientes a un filtro Coseno [2, 4, 5] cuya respuesta en frecuencia se observa en la figura 3. La gráfica fue realizada con MATLAB y la respuesta en frecuencia se calculó con la *fft* de 4096 puntos.

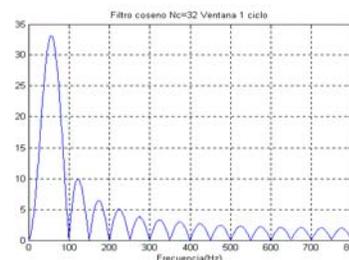


Figura 3

En el mismo entorno de MATLAB se genera una señal de entrada equivalente a una sinusoidal de 50Hz,  $\sin(2\pi 50t)$  y se le agrega ruido representado por los armónicos de la componente fundamental de 50Hz:  $0,3 \sin(2\pi 100t) + 0,2 \sin(2\pi 150t)$ . En la figura 4 puede observarse la señal de entrada contaminada con ruido en línea punteada y su correspondiente salida obtenida una vez aplicado el filtro.

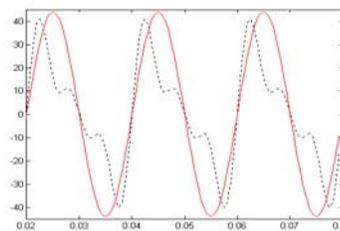


Figura 4

Utilizando un equipo de test HP16500B se generan los mismos valores que se obtuvieron en MATLAB y son ingresados al dispositivo EPF10K10LC84 ya configurado. Por medio del mismo equipo se pueden adquirir los valores a la salida del dispositivo de manera de comprobar si son de igual valor que los obtenidos en MATLAB. Los valores obtenidos por el equipo HP16500B fueron exactamente los mismos que en MATLAB. Como la prueba realizada fue exitosa la próxima experiencia es comprobar todo el sistema de filtrado completo.

## 6. FUNCIONAMIENTO DEL SISTEMA

Antes de proceder a comprobar el correcto funcionamiento del sistema, se realizan ciertos análisis para luego poder verificar si los circuitos están trabajando tal como se desea.

El EPF10K10LC84 es quien se encarga de notificar a los dispositivos de retención y muestreo (*sample & hold*) y al convertor A/D (PIC16F877), que cada tiempo  $T$  deben realizar sus operaciones. Esta orden lo realiza a través del pin *conv*.

Una vez que el PIC16F877 recibe la señal *conv* en alto (5V durante 2 $\mu$ s) se hace un retardo de 10 $\mu$ s para permitir que cada LF398 adquiera su muestra. Luego de la espera empieza la conversión. Como el convertor A/D también posee un circuito de muestreo y retención, hay que realizar una demora mientras la señal se establece a la salida del capacitor. El tiempo mínimo necesario para adquirir la señal es de 20 $\mu$ s. El filtrado de cada señal funciona en forma secuencial, primero se calcula una señal y luego, basándose en el mismo hardware, la otra señal. Cada operación de filtrado, es decir el cálculo de la convolución, dura 66 $\mu$ s. Entonces después de un tiempo aproximado de 132 $\mu$ s a partir de la adquisición de una nueva muestra por parte del sistema, se tienen las dos señales ya filtradas.

La señal que se ingresa al sistema es una sinusoidal que varía entre un mínimo de 0,5V y un máximo de 1,5V. Como la mínima resolución que posee el convertor A/D es de  $5V / 1024 = 4,88mV$ , el mínimo valor que se obtendrá después de la conversión es 00011001=25, y el valor máximo 01001101=76.

Generando una señal en MATLAB que corresponda con los valores calculados y luego filtrando por medio de los coeficientes del filtro Coseno, se obtienen idénticos valores.

## 7. REALIZACIÓN EXPERIMENTAL DEL SISTEMA.

En los párrafos anteriores se especificó cada etapa que compone al sistema, sin embargo hay que recordar que para una adecuada utilización es conveniente agregar una etapa de filtrado anti-aliasing antes del muestreo. En la aplicación realizada en el laboratorio, se obviaré esta etapa porque las señales ingresadas al sistemas provienen de un generador de ondas Hewlett Packard 33120A y por lo tanto no es necesario contemplar ese fenómeno.

Las pruebas fueron realizadas ingresando señales sobre ambos canales y luego en un solo canal dejando libre el otro, de esta manera se logra observar si la salida del canal libre es afectado por la señal ingresada en el otro canal.

El filtrado de las señales funcionó correctamente según las características del filtro Coseno, como ser al insertar una señal de onda cuadrada de 50Hz y a la salida obtener una señal sinusoidal de igual periodo. Por otro lado, también se insertaron señales de periodos armónicos de la señal de

50Hz, es decir 100Hz, 150Hz, 200Hz, ..., etc., y en la salida se obtenía una señal nula.

Además, los valores en las salidas coinciden con los que se esperaban del cálculo analítico.

## 8. CONCLUSIONES.

En este trabajo se estudiaron las posibilidades técnicas de realizar un sistema con filtros digitales FIR con coeficientes variables. A partir de esto es posible diseñar un filtro determinado mediante el software MATLAB y realizar su implementación con el sistema digital de filtrado. Estas herramientas otorgan una amplia versatilidad de ensayos para diferentes procesos.

De manera de ampliar el sistema se está considerando aprovechar la características que ofrece el microcontrolador y así desarrollar otras operaciones. Una de esas operaciones puede ser desarrollar una interfase entre los filtros y algún dispositivo que sirva para configurar los coeficientes. Otras operaciones serían establecer características a partir de las señales filtradas, como ser el módulo, el ángulo, la diferencia de fase entre ellas o alguna operación matemática que determine conclusiones del sistema que provienen las señales.

## 9.REFERENCIAS

[1] Alan V. Oppenheim, Ronald W. Schaffer, "Discrete Time Signal Processing", Prentice Hall, 1999.

[2] H.O. Pascual, J.A. Rapallini, A.A. Quijano, "Implementación de un sistema de medida de impedancia para redes eléctricas en tiempo real." VII Workshop IBERCHIP IWS'2001, Montevideo, Uruguay, Marzo 2001.

[3] [www.altera.com](http://www.altera.com)

[4] G.Gastaldi, J.A. Rapallini, A.A. Quijano, "Diseño de un sistema digital de medición de impedancia a tiempo real utilizando la metodología de codiseño HD/SW." IX Workshop IBERCHIP IWS'2003, La Habana, Cuba, Marzo 2003.

[5] J. Rapallini, J Requena, A Quijano, "Filtros digitales aplicables en equipos de medición para ensayos eléctricos" X Workshop IBERCHIP IWS'2004, Cartagena de Indias, Colombia.