

# Procesamiento Digital de Señales en FPGA para Análisis de Alta Velocidad en Entornos Industriales

Mariano Lizárraga<sup>1</sup>, Sergio Gallina<sup>1</sup>, Juan Contreras<sup>1</sup>, Gastón Peretti<sup>2</sup>

<sup>1</sup>Laboratorio de Tratamiento de Señales, Facultad de Tecnología y Ciencias Aplicadas, Universidad Nacional de Catamarca, Maximio Victoria 55 – (4700) Catamarca, Argentina

<sup>2</sup>Universidad Tecnológica Nacional, Regional Villa María

{emlizarraga@conicet.gov.ar, sgallina@tecono.unca.edu.ar, cjuanjose@hotmail.com, gastonperetti@gmail.com}

**Resumen.** Este trabajo propone un sistema de procesamiento de información en alta velocidad cuyo diseño está distribuido entre partes de software, hardware y firmware, buscando mostrar la utilidad de incluir sistemas basados en FPGA (Field Programmable Gate Array) en aplicaciones industriales donde se requiera un procesamiento rápido, por ejemplo para el procesamiento digital de señales (Digital Signal Processing, DSP).

Así se muestra que el procesamiento realizado en la FPGA puede ser complementado con un tratamiento en una PC, en la que se puede aprovechar la potencia y funcionalidades especiales de herramientas de software estándar tales como Matlab®. El vínculo entre las plataformas en este caso se basa en una comunicación Ethernet que se implementa en la FPGA mediante una descripción VHDL de uso libre que se ha desarrollado para este trabajo. Para la conexión física se ha construido una interfaz de adaptación entre puertos de E/S de la FPGA y un puerto Ethernet. Finalmente, en la PC se administra la comunicación mediante una conexión Ethernet/IP/UDP convencional, lo que da versatilidad al diseño dado que permite independizarse del lenguaje de programación y ser aprovechado en otras aplicaciones específicas.

Se presenta un caso de estudio para el cual se utiliza un algoritmo basado en el cálculo de la transformada de Fourier (Fast Fourier Transformation, FFT) para la determinación de la velocidad de giro de un motor eléctrico a partir de señales vibratorias captadas.

**Palabras clave:** FPGA, DSP, Ethernet, FFT.

## 1 Introducción

Dada la conveniencia de realizar un mantenimiento predictivo a una máquina u equipo, es necesario determinar variables físicas relevantes para estimar su estado de funcionamiento [1].

Tiempo atrás se entendía por monitoreo la mera medición de una variable física, y con esta era posible realizar manualmente una comparación con ciertos patrones y, por ejemplo, determinar si una máquina funcionaba adecuadamente o debía ser reparada. En la actualidad, la tecnología ha permitido automatización en estas técnicas y es así que ahora se incluye en el monitoreo la adquisición, el procesamiento y el almacenamiento de los datos recogidos, con un sentido más general. De acuerdo con las necesidades tecnológicas de este esquema, se considera un sistema en el que se propone realizar la adquisición de la información

mediante un dispositivo electrónico basado en una FPGA, el cual permite calcular a gran velocidad operaciones lógicas que a su vez pueden ser operaciones aritméticas y ser útiles en el procesamiento de información [2], [3]. Con este procesamiento preliminar en la FPGA se transmite información hacia una PC donde se pueden realizar operaciones menos sensibles a la velocidad, y de esta forma el procesamiento se complementa y se facilita el despliegue de la información para un usuario, a su vez el almacenamiento de grandes volúmenes de datos es facilitado también. Para interconectar los dos componentes, se propone utilizar una red de área local basada en Ethernet.

Con respecto al caso particular de aplicación que se considera, se debe notar que existen varias técnicas aplicadas al mantenimiento preventivo en máquinas, de entre ellas nos hemos abocado al análisis de las ondas sonoras que emite el dispositivo. Esta metodología constituye un caso simplificado de un tratamiento más efectivo basado en señales de ultrasonido, el cual permite: detección de fricción en máquinas rotativas, detección de fallas y/o fugas en válvulas, detección de fugas de fluidos, pérdidas de vacío, detección de "arco eléctrico", etc. Sin embargo, el desarrollo presentado constituye una base suficiente para elaborar conclusiones en relación con las aplicaciones donde es conveniente este tipo de arquitecturas y brinda un punto de partida para el ajuste del sistema a una aplicación específica diferente.

## 2 Procesamiento Distribuido

El análisis propuesto se basa en un estudio espectral que desarrollamos mediante la utilización de la transformada rápida de Fourier (*Fast Fourier Transformation*, FFT) como base para procesar, diagnosticar y pronosticar el funcionamiento de un motor eléctrico. Sin embargo, el algoritmo de procesamiento de la información se implementa dentro de un conjunto de componentes distribuidas entre software, hardware y firmware, donde se busca poner en evidencia la utilidad de sistemas basados en FPGA para aplicaciones industriales donde se requiere procesamiento de alta velocidad, por ejemplo procesamiento digital de señales (DSP) como en este caso. Así se propone que tal procesamiento embebido en el circuito FPGA puede ser complementado con un tratamiento en una PC, aprovechando

la potencia de Matlab® y contando con un medio de comunicación adecuado, para esto se ha descrito una arquitectura, que se brinda como código abierto de uso libre en [4], para generar tramas Ethernet/UDP/IP y es necesaria una interfaz de adaptación física entre los puertos de E/S de la FPGA y un puerto Ethernet (de tipo RJ-45).

En Matlab® la información recibida se almacena en variables convencionales, y por lo tanto puede realizarse cualquier procesamiento típico utilizando las funciones incluidas en este software. Aunque en el computador la conexión se establece mediante un *socket* UDP controlado por un código Matlab, las conexiones *socket* pueden ser administradas mediante cualquier otro lenguaje, incluyendo C/C++, Java, etc. y esto le da una gran versatilidad a la aplicación que describimos. Con Matlab se ha definido un código capaz de abrir un *socket*, interpretar la trama recibida y almacenar la información de usuario en una variable de tipo arreglo, para esto son necesarios tratamientos intermedios que contemplan las variables de punto fijo utilizadas en la FPGA.

En los puntos 3 y subsiguientes se describe con detalle el procesamiento en el caso analizado, no obstante se destaca que sería posible realizar cualquier otro procesamiento en Matlab de forma convencional, entre los que se encuentran el filtrado, el cálculo de otras transformadas, etc. e incluso la graficación, dado que este software ha sido considerablemente optimizado para este objetivo.

### 3 Caso de estudio

Se considera la estimación de la velocidad de giro de un motor eléctrico a partir de la señal sonora captada con un micrófono convencional. Este sistema se grafica en la Fig. 1 y las funciones implementadas en cada bloque se discuten con más detalle a continuación.

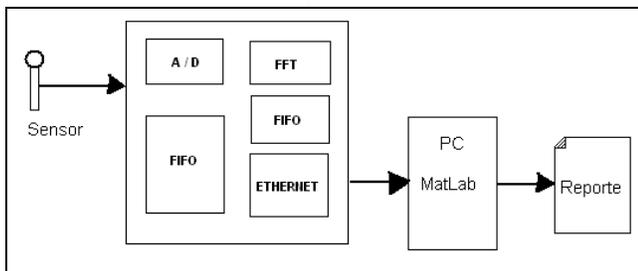


Fig 1. Diagrama de bloques del sistema analizado

#### 3.1 FFT

Dado que un motor eléctrico genera una onda vibratoria, esta es percibida por un prototipo electrónico basado en una FPGA, el cual recibe la señal directamente mediante un códec de audio Texas Instruments modelo TLV320AIC23 conectado a un micrófono; sobre esto son interesantes las siguientes consideraciones, por un lado la simplicidad del esquema de adquisición de la señal, pues no se requiere conexión mecánica a la maquina analizada y además se utiliza un sensor simple,

tal como lo es un micrófono convencional; por otro lado, la utilización de este sensor plantea un análisis ulterior para examinar el desempeño para señales de mayor frecuencia como las de ultrasonido, recurriendo a un sensor adecuado.

El control electrónico del códec, que constituye un conversor analógico-digital con frecuencia de muestreo máxima de 96 kHz, representa un caso simple de control de señales de reloj y habilitación. Así la información es cuantizada y convertida a tiempo discreto para ser almacenada en una memoria FIFO. Desde esta memoria se alimenta un bloque especial descrito mediante VHDL que calcula la FFT [5], [6] de acuerdo con la siguiente multiplicación matriz-vector equivalente

$$s = F \cdot x$$

donde  $F$  es una matriz de  $N$  filas y  $N$  columnas que determinan el largo de la transformada. Los vectores  $x$  y  $s$  son de largo  $N$  y se representan mediante valores de punto fijo de 14 bits. El vector  $s$  representa la transformada de Fourier de la señal de entrada  $x$ .

La salida del bloque FFT está conectada a una segunda memoria FIFO que se descarga con la transmisión del vector  $s$  hacia la PC. Este proceso se ejecuta iterativamente a alta velocidad.

#### 3.2 Ethernet

Un módulo independiente se encarga de la generación de una trama Ethernet que se ajusta a la pila de protocolos Ethernet/IP/UDP. Esto permite la integración del dispositivo FPGA en una red de área local y cuenta con las ventajas de la señalización diferencial de este estándar, lo que le otorga una buena inmunidad ante las interferencias propias de los ambientes industriales. El módulo utiliza una señal de reloj de 20 MHz y de ella se deriva la sincronización de las partes que conforman la trama, donde se incluye el preámbulo, los campos de dirección de origen y destino, el campo de datos de usuario, y el campo de control de errores.

En la Fig.2 Se muestran algunas ondas del bloque de transmisión Ethernet, donde se resalta la señalización diferencial que se obtiene mediante señales lógicas complementadas. Este recurso proporciona una comunicación que no cumple todas las especificaciones eléctricas del estándar pero es suficiente para lograr una comunicación efectiva con un cableado de corta distancia, sin embargo mediante cualquier dispositivo de la red esta señal se puede regenerar para cumplir con el estándar y permitir comunicaciones de mayor alcance. En la Fig. 2 También se indican algunas partes características de la trama, como el preámbulo.

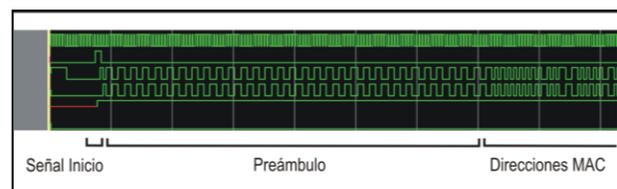


Fig. 2: Ondas en el bloque de comunicación Ethernet

### 3.3 Procesamiento de la información

Para el caso propuesto, el análisis de la información se completa mediante un procesamiento en una PC, por medio de Matlab. Con el código diseñado se controla la placa de conexión Ethernet y con esta se establece una conexión de tipo *socket*.

Mientras se recibe la trama, se carga una variable de tipo arreglo de largo  $N$ , donde se almacena la FFT calculada en la FPGA. Para obtener luego una estimación de la densidad de potencia espectral [3] siguiendo la expresión

$$P = |s|^2$$

Por lo que en la PC se realizan operaciones vectoriales simples y finalmente se grafica. En la Fig. 3 se muestra un diagrama de las operaciones mencionadas.

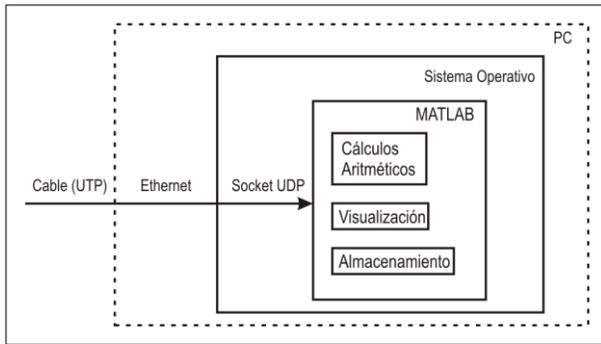


Fig. 3: Diagramación funcional y de protocolos para el tratamiento en la PC

### 3.4 Resultados

La prueba se configura fijando un largo de FFT 4096 puntos, dando  $N=4096$ . Esta operación se calcula sobre una señal muestreada a 44 KHz y con un total de 4096 puntos, con lo que se determina una ventana de análisis de 93 ms. Para esta configuración el consumo de recursos en la FPGA es la dada en la Tabla 1.

Recurso	Utilización	Total Disponible	%
Logic Elements Altera®	2922	68416	4
LABs (ut. parcial o total)	229	4276	5
Registros	1760	69634	3
Bits de Memoria	387062	1152000	34
M4Ks	84	250	34
Puertos E/S (I/O)	8	422	1.9
Multiplicadores Hardware	0	300	0
JTAGs	0	1	0

Tabla 1. Consumo de recursos en la FPGA

Por su parte, en la PC se obtiene una representación de la señal de entrada (que corresponde al vector  $s$ , obtenido luego de calcular la FFT) y esto se da en la Fig 4a). El gráfico de la

estimación de la densidad espectral de potencia, siguiendo la Ec. (2) se presenta en la Fig. 4b).

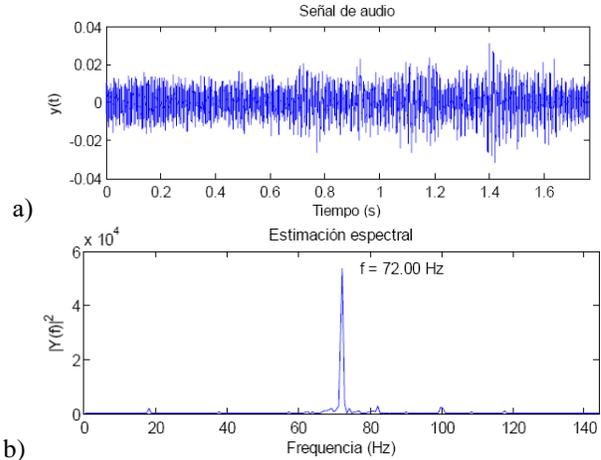


Fig.4: a) Representaciones Matlab de la señal recibida; b) Densidad espectral de potencia

Gráficamente puede encontrarse el máximo de energía que representa directamente la frecuencia de giro del motor. No obstante, esto es hecho algorítmicamente en Matlab dando

$$f_0 = \max\{P\} = 72$$

Por lo que el resultado del procesamiento es estimar la velocidad de giro en 72 RPM y mostrar la composición espectral de parte de la componente audible las vibraciones del motor.

## 4 Conclusiones

El sistema analizado responde a una arquitectura de procesamiento distribuido donde el procesamiento de alta velocidad se realiza en un circuito FPGA con un firmware diseñado en VHDL que le da características de procesador específico.

Los requerimientos de cálculo especificados son alcanzados y se verifica que para el caso de estudio descrito, si bien se requiere del cálculo de miles de operaciones aritméticas, el porcentaje de utilización de recursos disponibles en la FPGA es bajo; no obstante, permitiendo una elevada frecuencia máxima de operación de 50,997 MHz.

Por otra parte, se ha aprovechado la potencia de procesamiento de una PC y la facilidad para desplegar resultados visualizables, para realizar un análisis simple de la señal recibida. Una ventaja adicional de la herramienta Matlab® proviene de la posibilidad de generar una aplicación ejecutable independiente del entorno Matlab, con lo que sería posible conectar el sistema electrónico basado en la FPGA a cualquier PC, y corriendo en esta sólo la aplicación generada que se puede distribuir libre de licencia.

El sistema completo está constituido por partes de software y hardware, que podrían ser reutilizadas para propósitos específicos accediendo a los archivos de código dados.

## Referencias

1. Durocher, D.B.; Feldmeier, G.R.; , "Predictive versus preventive maintenance," Industry Applications Magazine, IEEE , vol.10, no.5, pp. 12- 21, Sept.-Oct. 2004
2. Hongzhong Ma; Qun Xu; Jingang Song; Jingdong Han; , "The application of zoom FFT technique to the extraction of fault character of induction motor," Condition Monitoring and Diagnosis, 2008. CMD 2008. International Conference on , vol., no., pp.221-225, 21-24 April 2008
3. Cabal-Yepez, E.; Osornio-Rios, R.A.; Romero-Troncoso, R.J.; Razo-Hernandez, J.R.; Lopez-Garcia, R.; , "FPGA-Based Online Induction Motor Multiple-Fault Detection with Fused FFT and Wavelet Analysis," Reconfigurable Computing and FPGAs, 2009. ReConFig '09. International Conference on , vol., no., pp.101-106, 9-11 Dec. 2009
4. Universidad Nacional de Catamarca, <http://www.tecno.unca.edu.ar/content/view/49/81>
5. Proakis , John G.; Manolakis, Dimitris K. "Tratamiento digital de señales", cuarta ed. Prentice Hall, 1999
6. Oppenheim, Alan V.; Schafer, R. W.; and Buck, J. R. (1999). "Discrete-time signal processing". Prentice Hall, 1999

