

# Una Herramienta de Análisis de Retardos de Interconexión en FPGAs

Eduardo Boemo<sup>\*</sup>, Nelson Acosta<sup>#</sup> y Elías Todorovich<sup>#2</sup>

<sup>\*</sup>E.T.S. Ingeniería Informática, UAM  
Ctra. de Colmenar Km 15, 28049 Madrid, España

<sup>#</sup>ISISTAN - Facultad de Ciencias Exactas - UNCPBA  
Paraje Arroyo Seco, Campus Universitario, 7000 - Tandil

TE: 0293 - 32466 - Fax: 0293 - 40362 - Email: {nacosta;etodorov}@exa.unicen.edu.ar

**Resumen:** En este artículo se presenta una herramienta para el análisis de la interconexión de circuitos mapeados en FPGAs. A partir de los ficheros de información *post-layout*, el programa permite seleccionar subconjuntos de pistas de un determinado *fanout*, graficar histogramas de retardos, calcular estadísticas, analizar *skew* de reloj, o exportar esta información a formatos de programas de análisis más avanzados tales como Origin o Matlab. La herramienta se ejecuta en Windows 95-98-NT, ha sido programada utilizando Delphi 4 y es totalmente compatible con el programa Xilinx Foundation Series.

**Palabras clave:** FPGAs, retardo de interconexiones.

## 1.-INTRODUCCIÓN.

Una de las características distintivas de los circuitos integrados en tecnologías submicrónicas es el predominio de la interconexión sobre las demás fuentes de retardo. En el escalamiento ideal (reducción de todas las longitudes en un mismo factor) el retardo de la lógica disminuye mientras que el de las pistas cortas se mantiene constante; peor aún, el retardo de las pistas largas se incrementa con el factor de escala [Bak85].

Un segundo factor que contribuye a resaltar el efecto de la interconexión es la utilización de un proceso automático de particionado, emplazamiento y rutado (PPR), habitual en tecnologías del tipo de las FPGAs. En este caso, el retardo de interconexión no sólo constituye la fracción más importante del período de operación, sino que además su valor tiene un carácter

aleatorio, que dificulta calcular *a priori* la frecuencia final de operación. En efecto, cuando el retardo de pista es del orden del retardo combinacional de una puerta o LUT (*look-up table*), la determinación del camino crítico a partir de la profundidad lógica pierde exactitud, resultando más conveniente utilizar el concepto de "conjunto de caminos críticos estadísticos" [Jyu93]. Éste se define como todos los caminos que superan un determinado umbral. Así, un análisis "en papel" sólo servirá para determinar un conjunto de caminos "candidatos" a camino crítico, cuya importancia final dependerá no sólo del PPR, sino también de las variaciones *run-to-run* de los parámetros, o el efecto de un subconjunto particular de datos.

Finalmente, el efecto de la interconexión también afecta a la velocidad del circuito, a través del aumento del *skew* de reloj, o la diferencia (medida en unidad de tiempo) entre flancos del mismo pulso en distintos puntos del circuito [Fis85]. En [Kug90] se ha demostrado de manera general que no es posible implementar un reloj global de modo que el máximo *skew* esté limitado por una constante independiente del área. Esto conduce a dos alternativas para los sistemas síncronos: los circuitos autotemporizados o *self-timed* [Sut89] y los *waves pipelines* [Bur98]. En los primeros, el reloj global es reemplazado por una sincronización distribuida, que conduce a circuitos por pistas locales únicamente, tanto en las líneas de datos como de sincronización. En los *waves pipelines* el problema del *skew* es atacado de manera diferente: el árbol de reloj se reduce significativamente debido a la eliminación de los registros intermedios del circuito. Para una determinada tecnología, ambas técnicas requieren de un estudio previo de las características generales de la interconexión; en

---

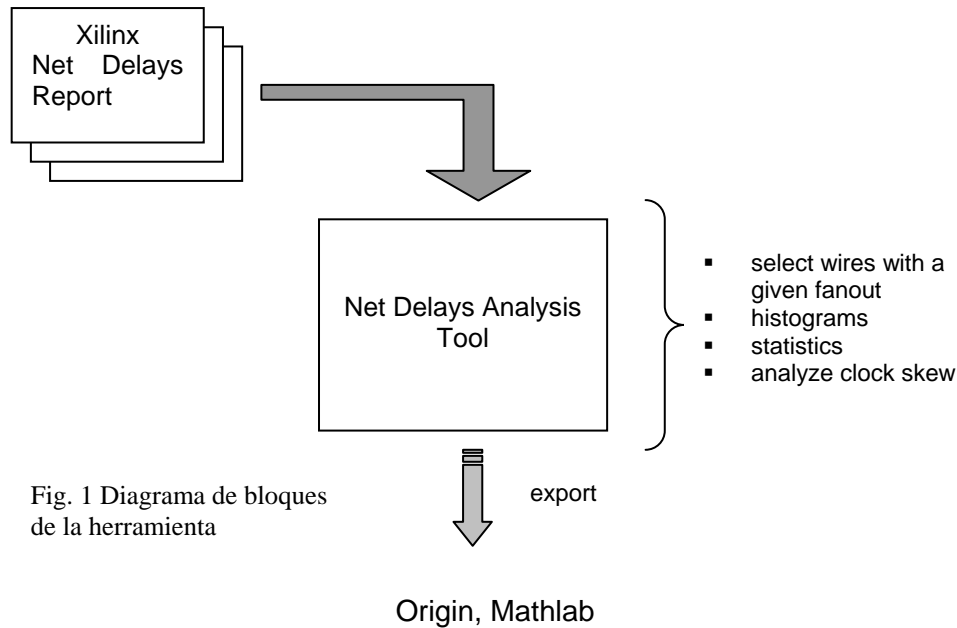
<sup>2</sup> Becario CONICET, Argentina

particular, de la dependencia del retardo con el *fanout*.

Este trabajo se ha organizado del siguiente modo: en la sección 2 se presenta la herramienta de análisis de retardos desarrollada. En la sección 3 se explican algunos ejemplos de uso de este programa y en la sección 4 se resumen las conclusiones del trabajo y se plantean algunas extensiones del

La disponibilidad en Internet de componentes de software de uso general, muchos de los cuales se proveen inclusive con el código fuente es otra importante facilidad a la hora de construir aplicaciones. En este sentido se usaron 3 componentes para:

- Seleccionar una carpeta específica. El diálogo OpenFileDialog que provee Delphi permite abrir



mismo.

## 2.-HERRAMIENTA DE ANALISIS.

La herramienta de análisis de retardos está desarrollada con Delphi 4 [Inp98a] [Inp98b] teniendo en cuenta sus características de:

- Construcción rápida de aplicaciones (RAD, *Rapid Application Development*),
- Ambiente integrado de desarrollo con características como editores que ayudan en la corrección sintáctica del código escrito (*Syntactic Editors*), compilación rápida, *debugging*, etc.,
- Diseño de reportes y formularios visual y dirigido por eventos,
- Lenguaje de programación orientado a objetos compilado,
- Ambiente extensible y abierto para programadores y desarrolladores,
- Facilidades para el desarrollo basado en componentes [Cal98].

solamente archivos y no se puede seleccionar una carpeta que es lo que se necesita en esta aplicación justamente para elegir el directorio base en el entorno Xilinx Foundation [Xil98].

- Buscar en un directorio (carpeta) archivos con una determinada extensión. Se basa en el uso de las funciones **FindFirst** / **FindNext** / **FindClose**, y además provee la habilidad de hacer recursiva la búsqueda a partir de un directorio dado.
- Dibujar gráficos sobre los ejes cartesianos. Los datos a ser representados se almacenan en series que se pueden cargar automáticamente desde varias fuentes. También, se pueden analizar estos conjuntos de datos estadísticamente. La clase **THistogram** forma parte del conjunto de clases que se provee con este conjunto de componentes para la representación gráfica de funciones y es la que se usó en esta herramienta. Asimismo, como se dispone en este caso del código fuente se

podieron realizar algunas extensiones que se aprovechan en la herramienta presentada cumpliendo con los objetivos de generalidad que deben tener estos componentes de software.

Con esta utilidad se selecciona el directorio base a partir del cual se quiere analizar una parte o uno o más proyectos y entonces el programa busca a partir del directorio seleccionado todos los archivos de reporte de demoras de conexión (\*.dly). Para cada uno de estos archivos se buscan todos los valores de retardo y se cargan en la serie de su histograma respectivo. Los histogramas se crean dinámicamente a medida que se van encontrando

términos absolutos o relativos, elegir el número de intervalos que definen el histograma, etc.

Como existen muy buenas herramientas de análisis estadístico y de graficación, queda claro que el objetivo principal de este trabajo es el de extraer la información que puede aparecer dispersa dentro de un proyecto de diseño de Xilinx usando los programas Foundation Series. Entonces es razonable proveer la capacidad de exportar la información recopilada a formatos de programas de análisis más avanzados tales como Origin® o Mathlab® (ver fig. 1).

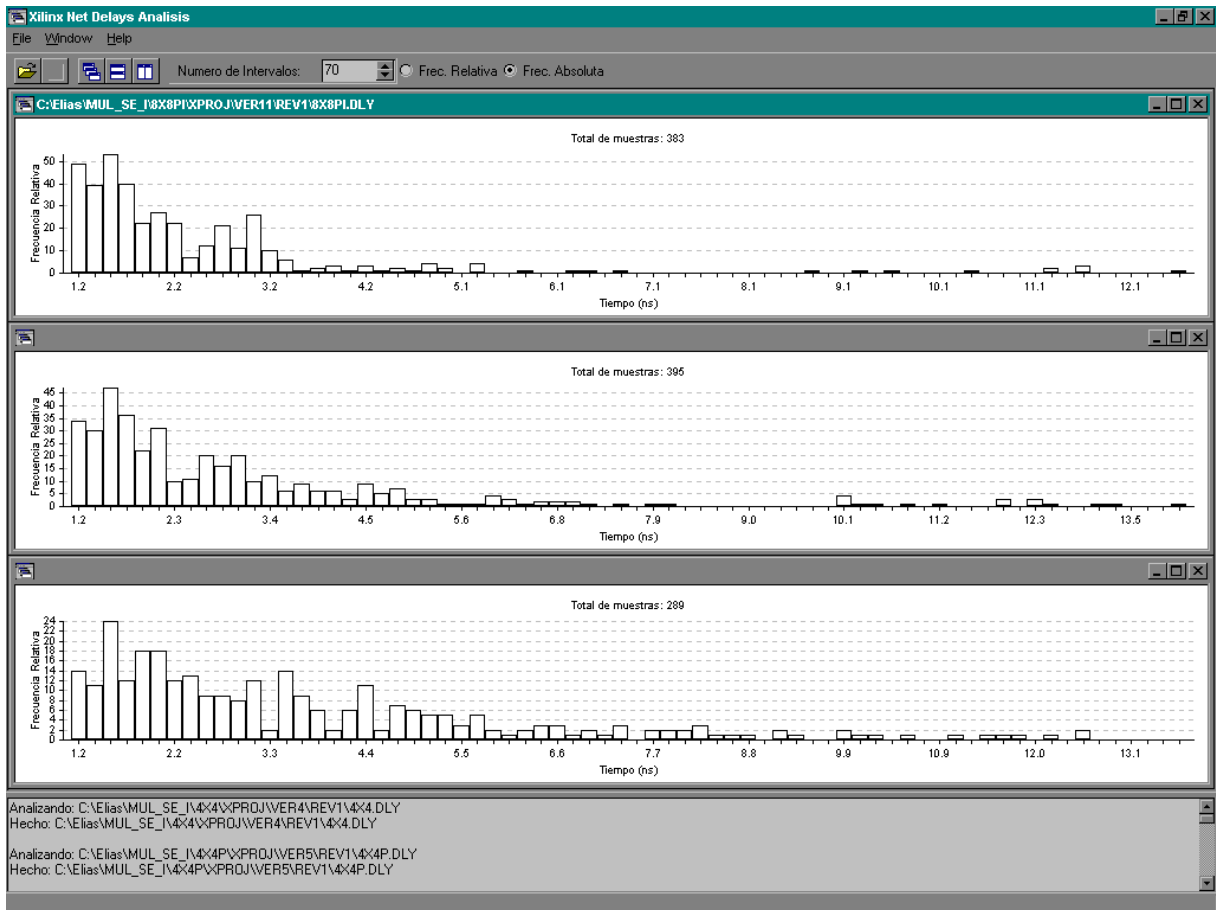


Fig. 2. Multiplicadores. Análisis de conexiones locales vs. globales.

los archivos dly.

Una vez terminada esta etapa, se pueden seleccionar subconjuntos de pistas de un determinado *fanout*, calcular estadísticas, analizar *skew* de reloj, configurar la apariencia del histograma como por ejemplo ver los datos en

### 3. ALGUNOS EJEMPLOS DE APLICACIÓN

Para mostrar la forma de uso de esta herramienta se presentan algunas posibles utilidades de la misma. Como caso de estudio, han sido utilizado

algunos ficheros “de reporte” correspondientes a experimentos desarrollados en [Boe96].

- Estudio del efecto de la dirección de segmentación (*pipelining*) sobre las conexiones locales y globales.
- Análisis de *skew* en la señal de reloj en una FPGA.
- Análisis del retardo en función del fanout.

### 3.1.- Análisis de conexiones locales y globales

Para mostrar la manera como se usa el software aquí presentado se utilizan los multiplicadores binarios segmentados (*pipeline*) de McCanny-McWinter [McC82] de 8 bits y el de Hatamian-Cash. En estos ejemplos se tiene la misma arquitectura pero se usan distintas direcciones de segmentación para poder transformar a líneas locales todas las interconexiones del circuito.

La tecnología FPGA usada es la de Xilinx, más específicamente se usa el modelo XC4010EPC84-4 [Xil98] que se compiló desde esquemáticos usando

En la fig. 2 se muestra la captura de pantalla con los histogramas de los retardos para los multiplicadores mencionados más otro sin segmentar.

### 3.2.- Skew de la señal de reloj en FPGA.

En la Fig.3 se muestra un ejemplo de aplicación de la herramienta para analizar la línea de reloj de una FPGA. Los datos se han extraído de un conjunto de ficheros *report* correspondientes a diferentes *pipelines* realizados utilizando una XC3090PC84-7. Luego se exportaron los datos obtenidos para ser usados en Origin. Se puede observar que el retardo máximo entre el *driver* de reloj y los registros no llega a 1 ns, con independencia del tamaño de los circuitos de prueba, mientras que el *skew* máximo es cercano a 0.6 ns.

### 3.3.- Retardo vs. Fanout

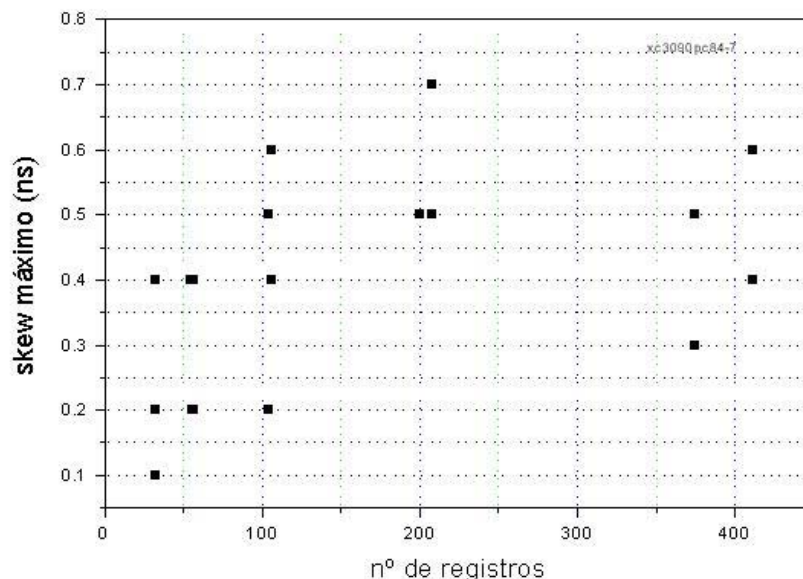


Fig. 3. Análisis de la línea de reloj de una FPGA

Foundation Series 1.4 con las opciones de compilación por defecto.

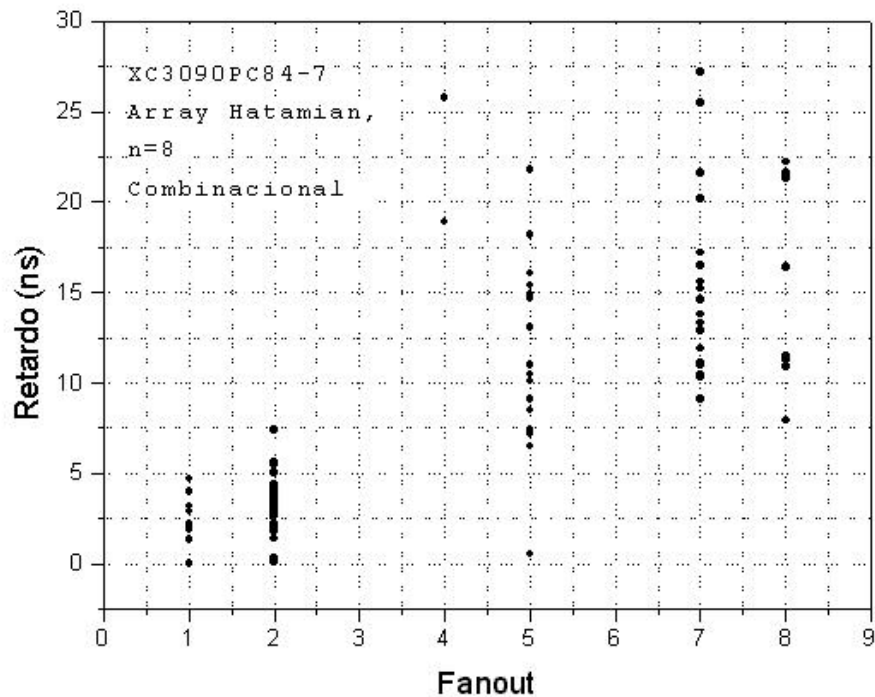


Fig. 4. Retardo vs *fanout*

La figura 4 muestra el retardo de interconexión en una FPGA tipo XC3090PC84-7 versus el *fanout* de cada nodo. Este tipo de gráficas resulta de gran utilidad cuando se realiza un PPR automático pues permite pronosticar el efecto de la interconexión. Se puede observar la importancia de la interconexión en esta tecnología: únicamente para *fanout*=1 el retardo de pistas resulta menor que el retardo combinacional de la LUT (7 ns).

#### 4.-CONCLUSIONES Y EXTENSIONES FUTURAS

La herramienta de análisis de retardos presentada es sencilla e intuitiva y tiene aplicación en varias área de investigación y desarrollo con FPGAs.

También puede notarse que, al usar componentes, se escribió poco código con la consecuente reducción en la probabilidad de errores de programación. Además, se ve que se puede usar este diseño para otras aplicaciones de recopilación de datos que se pueden extraer de múltiples

reportes del fabricante de FPGAs para la construcción de resúmenes específicos en aplicación con objetivos particulares.

#### REFERENCIAS

- [Bak85] H. Bakoglu, "Optimal Interconnections Circuits for VLSI", *IEEE Trans. on Electro Devices*, Vol.ED-32, nº5, pp-903-909, Mayo 1985.
- [Boe96] E. Boemo, "Contribución al Diseño de Arrays VLSI con Paralelismo de Grano Fino", Tesis Doctoral, ETSI Telecomunicación, Universidad Politécnica de Madrid, Enero 1996. Disponible en <http://www.ii.uam.es/~ivan/papers.htm>
- [Bur98] W.P. Burleson, M. Ciesielski, F. Klass and L. Pileggi, "Wave-Pipelining: A Tutorial and Reserch Survey", *IEEE Transactions on VLSI Systems*, vol. 6, no. 3, pp. 464-474,

September 1998

- [Cal98] Callan, James. Collaborative Computing with Delphi 3. Wordware Publishing, 1998.
- [Fis85] A. Fisher y H.T. Kung, "Synchronizing Large VLSI Processor Arrays", *IEEE Trans. on Computers*, pp.734-740. Agosto 1985.
- [Inp98a] Delphi 4 Object Pascal Language Guide, 1998.
- [Inp98b] Delphi 4 Developer's Guide, 1998
- [Jyu93] H. Jyu y S. Malik, "Statistical Timing Optimization of Combinational Logic Circuits", *Proc. 1993 IEEE Int. Conf. on Computer Design: VLSI in Computers & Processors*, pp.77-80. IEEE Computer Society Press: Los Alamitos-California, 1993.
- [Kug90] S. Kugelmass y K. Steiglitz, "An Upper Bound on Expected Clock Skew in Synchronous Designs". *IEEE Trans. on Computers*, vol.39, n°12, pp.1475-1477. Diciembre 1990.
- [McC82] McCanny J.V et al, "Completely iterative, pipelined multiplier array suitable for VLSI". - *IEEE Proc.* Pp.40-46. Vol.129, Part G, N°2. April 1982.
- [Sut89] I. Sutherland, "Micropipelines", *Communication of the ACM*, vol.22, n°6, pp.720-734. Jun. 1989.
- [Xil98] Xilinx Data Book, 1996.  
<http://www.xilinx.com>