

## SÍNTESIS HARDWARE DE SISTEMAS DIFUSOS MEDIANTE EL ENTORNO DE DESARROLLO XFUZZY

A. BARRIGA, I. BATURONE, S. SÁNCHEZ SOLANO, R. SENHADJI  
*Instituto de Microelectrónica de Sevilla.  
Centro Nacional de Microelectrónica,  
Edificio CICA, Avda. Reina Mercedes s/n,  
41012-Sevilla, SPAIN.*

*En esta comunicación se describe una metodología para el diseño microelectrónico de sistemas difusos utilizada en la docencia de materias relacionadas con las aplicaciones industriales de la lógica difusa. Dicha metodología se basa en una serie de herramientas de CAD específicas para los sistemas basados en lógica difusa, así como en otras herramientas genéricas de diseño de circuitos integrados digitales.*

### 1. Introducción

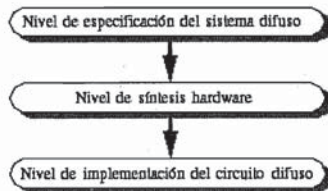
En los últimos años se ha incrementado considerablemente el número de aplicaciones que utilizan técnicas de lógica difusa para resolver problemas de control en tiempo real y, como consecuencia, se han desarrollado nuevas arquitecturas que permiten implementar de forma eficiente algoritmos difusos mediante hardware [1]. Las realizaciones hardware de sistemas difusos pueden estar basadas en dos estrategias genéricas. En la estrategia off-line las salidas del controlador son precalculadas para todas las posibles combinaciones de las entradas y el proceso de inferencia se lleva a cabo mediante una tabla de búsqueda implementada en una RAM o mediante circuitos combinatoriales [2]. Aunque esta estrategia permite una gran flexibilidad en la definición del controlador difuso, su mayor inconveniente es el crecimiento exponencial de la memoria requerida cuando aumenta el número de entradas o el número de elementos en los universos de discurso. Para solventar este problema la mayoría de las realizaciones de controladores difusos siguen una estrategia on-line, donde el proceso de inferencia se realiza de forma concurrente con los cambios en las entradas por medio de hardware dedicado [3].

### 2. Metodología de diseño

La metodología de diseño se ilustra en la figura 1 y se basa en tres niveles de abstracción. El flujo de diseño se apoya en el entorno Xfuzzy [4]. Este entorno facilita las diferentes etapas de descripción, verificación y síntesis de sistemas difusos. El primer nivel corresponde a la descripción de las especificaciones del sistema de inferencia difuso. Para ello se definen las funciones de pertenencia para los antecedentes y consecuentes, la base de reglas y los operadores y conectivos difusos (implicación, defuzzificación, etc).

---

Este trabajo ha sido parcialmente financiado por los proyectos de CICYT TIC98-0869 y TXT1998-1384.



**Figura 1:** Niveles de diseño de sistemas difusos mediante el entorno de desarrollo Xfuzzy.

El segundo nivel corresponde a la síntesis hardware. El proceso de síntesis consiste en trasladar la especificaciones de alto nivel del sistema difuso en una descripción del circuito que constituye dicho sistema. La formato de la descripción del circuito dependerá de la estrategia de diseño (off-line mediante una tabla de búsqueda o bien on-line basada en una arquitectura específica de procesador difuso).

El tercer nivel consiste en la implementación del circuito. Dicha implementación puede realizarse siguiendo dos técnicas: como ASIC (Application Specific Integrated Circuit) en una tecnología determinada o bien sobre un FPGA (Field Programmable Gate Arrays). En este nivel el flujo de diseño corresponde a un proceso de diseño estándar de circuitos integrados digitales y se basa en herramientas convencionales de diseño de circuitos.

### 3. Nivel de especificación del sistema difuso

El primer paso en el proceso de diseño consiste en la definición de la estructura del sistema difuso. Ello significa establecer tanto la base de conocimientos del sistema como definir los mecanismos de inferencia. En este nivel se realiza una descripción del sistema difuso basada en el lenguaje de especificación XFL [5]. El entorno de desarrollo Xfuzzy proporciona una serie de editores gráficos que facilitan la definición de las funciones de pertenencia de los antecedentes y consecuentes, la edición de la base de reglas, y la elección de los diferentes operadores difusos (Figura 2).

En este nivel la verificación de la especificación XFL se realiza mediante simulación y ajuste de los parámetros del sistema. En el caso de diseño de sistemas de control resulta especialmente útil combinar el programa C que describe al controlador difuso con un modelo, también en C, de la planta bajo control para analizar el comportamiento en lazo cerrado del conjunto formado por el controlador y la planta. Una segunda fase de verificación corresponde a la simulación del modelo software del controlador interactuando con la planta real. Para ello el entorno de diseño dispone de una herramienta denominada xflab que se basa

en el uso de una tarjeta de adquisición de datos para monitorizar el estado de la planta y actuar sobre ella.

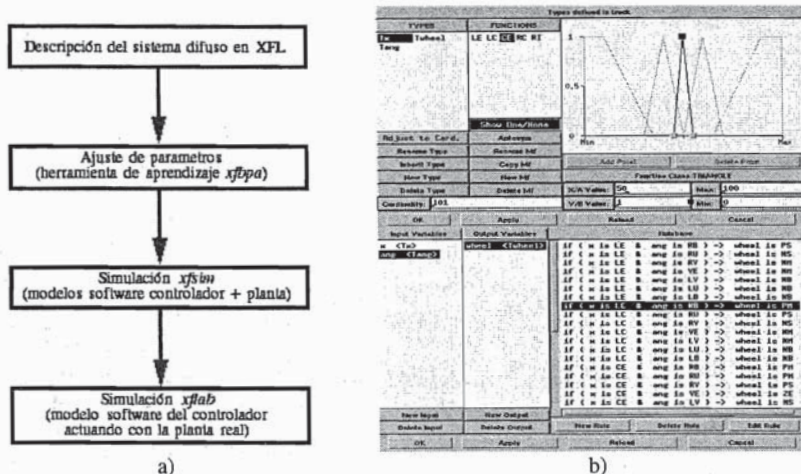


Figura 2: a) Flujo de diseño en el nivel de especificaciones. b) Ventanas de edición de Xfuzzy: editor de funciones de pertenencia (superior) y editor de reglas (inferior).

#### 4. Síntesis hardware e implementación del circuito

El segundo nivel de abstracción corresponde a la síntesis hardware del sistema difuso. La síntesis puede hacerse de acuerdo con las dos estrategias de realización comentadas previamente: basada en tabla de búsqueda y basada en hardware específico (Figura 3). En el primer caso, a partir de la especificación XFL del controlador, se genera una tabla entrada/salida con formato PLA de Berkeley, que puede ser minimizada mediante cualquier herramienta de síntesis lógica compatible con dicho formato. Por otra parte, la síntesis basada en hardware específico utiliza una arquitectura dirigida por reglas activas y con métodos de defuzzificación simplificados, que permite la implementación eficiente en silicio de controladores difusos de propósito general [6]. En este caso, a partir también de la especificación XFL, se genera una descripción VHDL del controlador, que utiliza los módulos incluidos en una librería VHDL de celdas parametrizadas y sintetizables. A partir de este punto el flujo de diseño continúa de acuerdo con una metodología top-down que puede ser automatizada en gran medida [7]. Para ambos casos, el entorno Xfuzzy proporciona herramientas que facilitan la definición de los diferentes parámetros de síntesis.

El tercer nivel en nuestra metodología corresponde a la implementación del circuito. Entre las dos posibles técnicas de implementación (ASIC y FPGA) se ha optado como tecnología de implementación por el uso de FPGAs. Esta elección presenta una serie de ventajas. En primer lugar, permite prototipar rápidamente sistemas que más tarde pueden ser realizados como



circuitos integrados. Además, los sistemas construidos con FPGAs exhiben una programabilidad intrínseca, proporcionando así un mecanismo simple para cambiar o adaptar su función durante la etapa de desarrollo. Por último, existen numerosas herramientas informáticas para diseño de FPGAs disponibles tanto a nivel comercial como a través de convenios educativos.

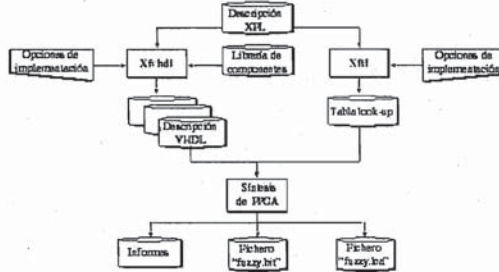


Figura 3: Flujo de diseño para una implementación de sistemas difusos sobre FPGA

En este nivel la verificación del sistema, constituido por el controlador difuso sobre el FPGA operando con la planta real, se realiza monitorizando el estado del circuito mediante la herramienta xflab.

## 5. Conclusiones

La metodología y las herramientas de CAD que se han descrito en esta comunicación constituyen un material didáctico que permite introducir al alumno en el diseño digital de circuitos integrados genéricos y, en particular, de circuitos integrados para aplicaciones difusas.

## Referencias

- [1] A. Costa, A. De Gloria, P. Faraboshi, A. Pagni y G. Rizzotto, Hardware Solutions for Fuzzy Control, Proc. of the IEEE, 83, 422-434, (1995).
- [2] J.Y. Leong, M.H. Lim y K.T. Lau, A General Approach to Encoding Heuristics on Programmable Logic Devices, Proc. fifth IFSA World Congress, 917-920, (1993).
- [3] D. L. Hung, Dedicated Digital Fuzzy Hardware, IEEE Micro, 15, 31-39, (1995).
- [4] D. R. López, C. J. Jiménez, I. Baturone, A. Barriga y S. Sánchez-Solano, Xfuzzy: A design environment for fuzzy systems, Proc. IEEE Int. Conf. on Fuzzy Systems, 1060-1065, (1998).
- [5] D. López, F.J. Moreno, A. Barriga y S. Sánchez-Solano, XFL: A language for the definition of fuzzy systems, Proc. IEEE Int. Conf. on Fuzzy Systems, 1585-1591, (1997).
- [6] I. Baturone, A. Barriga, S. Sánchez-Solano, C.J. Jiménez, y D. López, Microelectronic Design of Fuzzy Logic-Based Systems, CRC Press, (2000).
- [7] A. Barriga, S. Sánchez-Solano, C.J. Jiménez, D. Galán y D.R. López, Automatic Synthesis of Fuzzy Logic Controllers, Mathware & Soft Computing, III, 425-434, (1996).