

EXPERIENCIA EN LA DIDÁCTICA DE MÉTODOS AUTOMÁTICOS DE SÍNTESIS DE SISTEMAS DIGITALES

C. Rubio^a, S.A. Bota^b, R. Holgado^b, A. Herms^b

a. Escola Universitaria Salesiana de Sarria (UAB)

C/ Rafel Batlle, 7. 08017 Barcelona

Tel 93 280 52 44. Fax 93 280 66 42. email: crubio@euss.es

b. Departament d'Electrónica. Universitat de Barcelona

C/. Martí i Franques, 1. 08028 Barcelona

Tel. 93 402 90 68. Fax 93 402 11 48. email: sbota@el.ub.es

RESUMEN

La didáctica de las asignaturas de sistemas digitales, tanto en titulaciones de primer ciclo como de segundo ciclo, ha evolucionado como consecuencia de la aparición de nuevas metodologías de diseño y síntesis de sistemas digitales. El factor que han contribuido a acelerar esta tendencia han sido la popularización de los lenguajes de descripción hardware, la aparición de nuevas herramientas de síntesis en casi todos los entornos de desarrollo y el éxito de los dispositivos programables de altas prestaciones. En este trabajo evaluamos el impacto de esta evolución en nuestros respectivos centros: L'Escola Universitaria Salesiana de Sarria y la Facultad de Física de la Universidad de Barcelona.

1. INTRODUCCIÓN

En los últimos años, como resultado de un esfuerzo intensivo realizado tanto en centros académicos como industriales, se ha establecido el concepto de Diseño Automático, en referencia a la aparición de metodologías y herramientas de síntesis automática de sistemas digitales. Durante la década de los 80 y principios de los 90 este nuevo tipo de metodologías de diseño

estaba ligado casi exclusivamente al desarrollo de circuitos integrados de aplicación específica (ASIC), un tipo de componentes con altos tiempos de desarrollo y elevados costes de diseño [1]; en consecuencia, era una materia que, difícilmente se podía incluir en cursos universitarios introductorios. El panorama cambia con la aparición y posterior desarrollo de los dispositivos programables por el usuario de alta complejidad (FPGAs y CPLDs) y la popularización de los lenguajes de descripción hardware. Esta evolución ha influido en la docencia universitaria, bien sea a través del replanteamiento del programa de las asignaturas de electrónica digital y sistemas digitales [2][3], o por la implementación de nuevas asignaturas [4]. En este trabajo se analiza la incidencia de esta nueva situación en los programas docentes de nuestros respectivos centros: (i) Ingeniería Técnica Industrial en Electrónica Industrial, Titulación de primer ciclo en la *Universidad Salesiana de Sarrià* (EUSS) de Barcelona. (ii) Ingeniería Electrónica, Titulación de segundo ciclo en la Facultad de Física de la Universidad de Barcelona (UB).

2. PLANIFICACIÓN

2.1 EUSS

Se ha creado una asignatura optativa, *Microelectrónica*, para alumnos que ya han cursado *Electrónica Digital (ED)* y *Experimentación en Circuitos Electrónicos II (ECEII)*, asignaturas de introducción y profundización en sistemas digitales. ECEII es una asignatura primordialmente práctica, consta de 14 sesiones de trabajo en el entorno Max-Plus II de Altera. Por lo tanto, los alumnos que cursen esta nueva asignatura poseen conocimientos, no sólo en sistemas digitales, sino que también han adquirido suficiente práctica en el manejo de herramientas de diseño. Los objetivos de esta nueva asignatura optativa son introducir la metodología de diseño, desde el nivel de descripción hasta el de implementación sobre dispositivos programables. Obviamente, dentro del contexto de un plan de estudios de primer ciclo, que pretende acentuar el uso de las tecnologías microelectrónicas en aplicaciones prácticas en entornos industriales.

2.2 UNIVERSIDAD DE BARCELONA

Se ha programado una nueva asignatura optativa para alumnos que ya han cursado las asignaturas obligatorias de *Sistemas Digitales (SD)*, *Diseño Microelectrónico I (DMI)* y *Diseño Microelectrónico II (DMII)*. Los programas de estudio de estas asignaturas se han adaptado para facilitar la programación de la nueva asignatura. A priori los alumnos de esta asignatura tienen nociones de diseño digital a nivel transistor, lógico y RTL, diseño *Full-Custom* y *Semi-Custom*.

3. DESARROLLO

A continuación enumeramos algunos aspectos que surgen al plantear una nueva asignatura o actualizar un programa ya existente y que inciden en la planificación, tanto en el campo teórico como en los aspectos más prácticos de la asignatura.

- i) Planteamiento de la docencia en sintonía con una metodología de diseño concretada en flujos de diseño descendentes (desde las especificaciones iniciales hasta el diseño físico), figura 1, y flujos de información ascendente (información tecnológica ligada a la tecnología escogida y información de retro- anotación proveniente de una descripción de nivel inferior de abstracción).
- ii) Reflejar el éxito de los lenguajes de descripción del hardware (HDL) como herramientas de definición de un sistema electrónico, el hecho que un sistema digital pueda ser descrito en varios niveles de abstracción mediante el mismo lenguaje y sus semejanzas con otros lenguajes existentes tales como C o Pascal facilita enormemente su aprendizaje por parte de aquellos alumnos que tengan un poco de experiencia en programación. Hoy en día los lenguajes de descripción más frecuentes son Verilog y VHDL, en principio la elección de un lenguaje no debe afectar al planteamiento global de la asignatura. No obstante, el requisito de disponer de fundamentos de programación, obliga a cuidar la ubicación temporal de las asignaturas en el plan de estudios. Este hecho es crucial en el caso de una enseñanza de primer ciclo.
- iii) Evaluación de la importancia de las etapas de simulación y el análisis y su relación con la verificación en cada una de las fases del diseño.
- iv) Introducción a las herramientas de síntesis. Existen varias herramientas de síntesis de propósito general (compatibles con los lenguajes anteriores). Además, podemos encontrar una gran variedad de herramientas de carácter específico que utilizan lenguajes de descripción propios.
- v) Planteamiento de casos prácticos. Resulta fundamental planificar adecuadamente los ejemplos y casos prácticos para que los alumnos vean todo el espectro de posibilidades que ofrecen estas herramientas. Aunque la limitación temporal conlleve una tendencia a la simplificación, no hay que abandonar la posibilidad de ofrecer al alumno un conjunto de prácticas que le obliguen a transitar por todas las fases del diseño.

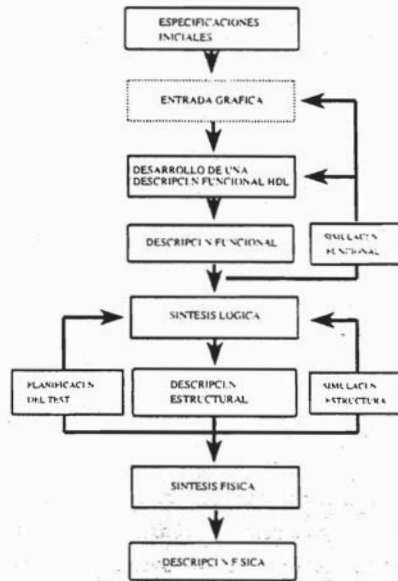


Figura 1. Flujo de diseño

- vi) Tecnología que se utilizará para implementar el circuito. Poder llegar hasta la implementación del dispositivo supone un aliciente para el alumno. Hoy en día, son asequibles todas las ventajas que ofrecen los dispositivos programables [6].

4. DISCUSIÓN

Los puntos anteriores admiten varios planteamientos, influenciados a su vez por las particularidades de nuestros respectivos centros: en la EUSS se pretende dar a conocer las ventajas y aplicaciones de las nuevas técnicas de diseño automático y de la microelectrónica en un entorno industrial. la actividad docente en este campo se centra en la programación HDL y en la síntesis lógica. Por otra parte, en la UB se pretende dar una visión detallada de todas las etapas de diseño automático, su relación con las opciones de implementación, la microelectrónica y las técnicas de diseño digital. No obstante en ambos casos surgen cuestiones similares como por ejemplo:

i) Interacción con asignaturas relacionadas. La síntesis automática de un circuito es comparable a la compilación de un programa de ordenador escrito mediante un lenguaje de alto nivel. entonces, ¿Es necesario seguir enseñando conceptos relativos a las características de los dispositivos microelectrónicos?, ¿Hay que replantearse la forma de presentar los principios del álgebra de Boole o los fundamentos de las maquinas de estado?. En principio, la comparación entre HDL y lenguaje de alto nivel, es correcta, aunque con ciertos matices: A diferencia de lo que sucede con los programadores de lenguajes de alto nivel, el experto en síntesis no debe ser únicamente un experto en programación HDL, también debe conocer las características de la tecnología que utilizará, no sólo se busca una solución que cumpla las especificaciones funcionales, sino que además se busca una solución óptima en cuanto a ahorro de área y velocidad de funcionamiento. Se ha comprobado que los alumnos no tienen demasiados problemas en aprender VHDL, de hecho tienen los mismos problemas que en aprender C, Pascal u otro lenguaje similar, los conceptos que pueden causar confusión son la concurrencia y la distinción entre variables y señales. Un problema detectado es que cuando los conocimientos de electrónica son bajos, el estilo de programación no refleja aspectos relacionados con la integración hardware. Los estudiantes de ingeniería electrónica tienden a pensar más en RTL ignorando a veces todas las prestaciones del HDL por lo que los programas resultantes pueden ser más complejos y difíciles de interpretar. La mejor solución para minimizar esta problemática consiste en revisar el planteamiento de asignaturas previas (ED, ECEII, SD, DMI, DMII etc.).

ii) Elección de herramientas CAD. La puesta a punto de las prácticas exige la utilización de herramientas CAD, aquí aparece el dilema sobre las herramientas a elegir: Herramientas de dominio público o comerciales. En ambos casos nos hemos decantado por herramientas comerciales, siguiendo los razonamientos expuestos en [4] y [6] (Hay que tener presente la pobre oferta de herramientas de dominio público en este campo [7]). En la UB se ha elegido el entorno Max-Plus II de Altera y Sinergy de Cadence. En la EUSS se trabaja con Altera y con Workview Office de ViewLogic.

iii) Programación de las prácticas. La planificación de las prácticas se caracteriza por un número de sesiones limitadas y los recursos disponibles. Conviene definir la gradualidad y temporalidad en la asimilación de los conceptos lingüísticos y semánticos básicos de los lenguajes escogidos. Así mismo, es fundamental para el éxito posterior, conocer las limitaciones de los lenguajes de descripción hardware en la síntesis de sistemas. El entorno de trabajo suele ser muy complejo y crea confusión a los alumnos, como solución se recomienda trabajar siempre con el mismo entorno y elaborar guiones que eviten pérdidas de tiempo en el manejo del CAD, intentando que el alumno sepa en todo momento que tarea está realizando. En esta programación debe contemplarse que los problemas de test y simulación son bastante frecuentes, y requieren bastante tiempo. Debe intentarse que las conclusiones obtenidas por el alumno sean

independientes de la herramienta de trabajo.

5. CONCLUSIONES

En este trabajo se han tratado algunos aspectos relacionados con la docencia de diseño automático en la EUSS, donde se imparte la titulación de Ingeniería Técnica Industrial en Electrónica Industrial, y en la UB, donde se imparte el segundo ciclo de Ingeniería Electrónica. Si bien el planteamiento en cada centro es muy diferente: en la UB se cubre todo el proceso de diseño, desde descripciones de comportamiento hasta diseño físico; en la EUSS se introduce al alumno en las metodologías de diseño, haciendo especial énfasis en síntesis mediante dispositivos programables. En ambos casos surgen problemáticas similares que puede abordarse con planteamientos equivalentes.

BIBLIOGRAFÍA

- [1] M.J.S. Smith. "Application-Specific Integrated Circuits". Ed. Prentice Hall. 1998.
- [2] M.S. Nixon. "On the programmable approach to introducing digital design". IEEE Trans. On education. vol. 40. Nº 3. pp 195-206. 1997
- [3] D.D. Gajski. "Principles of Digital Design". Ed. Prentice Hall. Cap. 3. 1997.
- [4] I. Katchan, F. Mayer, D. Schmid. "VLSI Top-Down design for students of computer science - A practical course". "Mixed Design of Integrated Circuits and Systems" editado por A. Napieralski et al. Kluwer Academic Publishers. pp 221-226. 1998
- [5] D.V. Bout. "The practical Xilinx Designer Lab Book". Ed. Prentice Hall. 1998.
- [6] S. Bota, A. Herms, B. Rigau, J.R. Morante. "Organización de una aula de diseño microelectrónico: experiencias con DF II". Libro de Actas TAAE 94. pp 131-137. 1994.
- [7] www.vhdl.org
- [8] K.C. Chang. "Digital Design and Modelling with VHDL and SYNTHESYS". IEEE Computer Society Press, 1997.