

Impacto del nuevo software de simulación en las prácticas de estructuras de computadores

Juan-Carlos Cano, Salvador Petit, Julio Sahuquillo

Departamento de Informática de Sistemas y Computadores

E. U. I.-Universidad Politécnica de Valencia

e-mail: {jucano, spetit, jsahuqui}@disca.upv.es

Resumen

Hasta hace pocos años las asignaturas de estructuras de computadores no contemplaban prácticas de laboratorio. De hecho, en la Universidad Politécnica de Valencia (UPV) los planes del 94 fueron los primeros que las incorporaron. La mayoría de las universidades elaboraron estas prácticas en unas condiciones precarias en cuanto a infraestructura (ordenadores, entornos de simulación, entrenadores lógicos, puestos de trabajo, etc.). A medida que los equipos han ganado en potencia y velocidad, la mayoría de estos puestos de trabajo han quedado desfasados y obsoletos para utilizar los entornos de simulación casi profesionales que existen hoy en día. Por otra parte, los nuevos planes de estudio que se están poniendo en marcha en la mayoría de las universidades españolas ofrecen un excelente escenario para la preparación de nuevas prácticas y/o adaptación de las ya existentes a estas nuevas herramientas.

En el presente trabajo se presenta la evolución de una práctica que ha mantenido los conceptos teóricos pero que se ha adaptado a los nuevos avances tecnológicos, que en principio, no diferirán mucho, de los que utilizarán los futuros ingenieros.

1. Introducción

En las enseñanzas de Ingeniería Informática, la realización de prácticas que impliquen el estudio de componentes de un computador requiere, en general, la utilización de una herramienta de simulación digital. El Software de simulación se ha convertido una herramienta de importancia

creciente ya que su carácter visual ayuda y refuerza la comprensión de los conceptos teóricos previamente tratados en clases magistrales y seminarios, y como consecuencia ejerce de elemento catalizador despertando en el alumno el interés por la materia.

Estas características se hacen especialmente visibles cuando dichas herramientas han sido diseñadas con carácter didáctico. Sin embargo, las herramientas didácticas tienen como mayor inconveniente el distanciamiento de la realidad con respecto a otras herramientas con características profesionales de elevado coste utilizadas en las empresas.

Con el avance tecnológico y la incorporación masiva del PC como elemento indispensable en la realización de las prácticas de laboratorio en carreras de Ingeniería, las empresas de Software profesional ofrecen versiones, que además de compartir características con sus “hermanas profesionales”, tienen características educativas a un bajo coste que las hacen especialmente atractivas para su utilización docente.

La selección de la herramienta de simulación no suele ser tarea fácil, ya que se tienen que dar una serie de circunstancias para que su uso reporte los beneficios que de ellas se esperan. Asimismo dicho Software debe ser adecuado al nivel del conocimiento y preparación del alumno, así como a los equipos disponibles. Por otra parte, si se considera que el ciclo de preparación y ajuste de prácticas de laboratorio suele ser de varios años, una elección equivocada nos retrasaría de forma importante.

En este trabajo, los autores presentan la evolución en el uso de dos herramientas de simulación diferentes en las asignaturas de Estructuras de Computadores, de las titulaciones de Ingeniería Informática de la Universidad Politécnica de

Valencia. En particular se contextualiza una práctica de laboratorio en el entorno de dos planes de estudio diferentes. Los planes del 94 y los planes de 2001.

En los planes del 94 se seleccionó como herramienta de simulación una herramienta educativa denominada *Computer Assited Simulation for Circuits Análisis & Design* (CASCAD) [1], con características distantes de herramientas profesionales, pero que se adaptaba bastante bien no solamente a la formación de los alumnos sino también a la potencia de los equipos disponibles.

En los planes de 2001 se ha optado por una versión educativa de una herramienta profesional denominada XILINX [2] que presenta características interesantes no solamente para asignaturas de primer curso sino también para asignaturas específicas de cursos avanzados. Aunque dicha herramienta se encuentra actualmente en fase de implantación, hay un compromiso entre los profesores por ponerla en marcha durante este curso e ir utilizándola en las sucesivas asignaturas.

El presente trabajo se organiza como sigue. En el apartado 2 se presentan los contenidos teóricos de la práctica ejemplo dedicada a la implementación de un banco de registros. A continuación en los apartados 3 y 4 se presenta brevemente la implementación de dicha práctica utilizando el CASCAD y el XILINX respectivamente. El trabajo termina comparando ambas herramientas y presentando, en base a la experiencia de los autores, algunas de las conclusiones del trabajo.

2. Práctica ejemplo: el banco de registros

La práctica sobre la que nos centraremos en este trabajo es la implementación de un banco de registros. Hemos seleccionado esta práctica no solamente por que su diseño requiere de conceptos previamente tratados en temas de diseño digital tales como los Multiplexores y Decodificadores, sino además porque será un elemento de especial importancia en el diseño de ruta de datos y unidad de control del procesador que se utiliza en todas las enseñanzas relacionadas con la Estructura del Computador.

Un banco de registros es un circuito que proporciona un conjunto de registros, a los cuales

se puede acceder para modificar el contenido de cualquiera de ellos, operación que recibe el nombre de escritura, o para averiguar el estado de los biestables de un determinado registro. Esta última operación recibe el nombre de lectura.

Estas dos operaciones se pueden implementar de distintas formas, a continuación se detalla una posible implementación, Para más información se puede consultar [4].

2.1. Operación de escritura

La Figura 1 ilustra un puerto de escritura en un banco de registros, utilizando un decodificador. Para escribir en un registro, se envía el dato de n bits que se desea escribir, a todos los registros que constituyen el banco, por un conjunto de n líneas que constituyen el bus de datos.

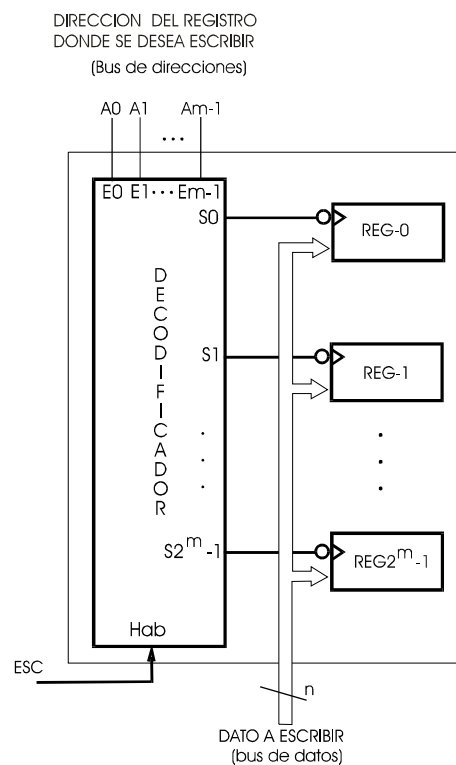


Figura 1. Implementación de un circuito de escritura en un banco de registros mediante un decodificador

Para seleccionar en qué registro se desea escribir, se utiliza un decodificador binario de m a 2^m . Se utilizan m líneas para indicar la dirección del registro $A=(A_{m-1}A_{m-2}...A_1A_0)$ (número de registro) sobre el que se desea escribir. Estas m líneas constituyen el bus de direcciones, y se conectarán a las m entradas del decodificador. Estas líneas de entrada activarán únicamente una de las 2^m salidas del decodificador. Como cada una de las salidas está conectada a un registro, la salida activa del decodificador habilitará el registro sobre el que se desea escribir. En estos momentos, los biestables del registro cambiarán su contenido si las líneas del bus de datos conectadas a sus entradas así lo indican. Obsérvese que para que esto sea factible, el decodificador debe estar activo; por lo tanto la señal de habilitación actúa como una señal de control que cuando está activa habilita la escritura.

2.1. Operación de lectura

En la Figura 2 se ha añadido un multiplexor que actúa como puerto de lectura sobre el banco de registros.

Además, se han representado las m líneas que constituyen el bus de direcciones mediante una sola línea, relativamente ancha, que representa el bus. Las salidas de los registros se encuentran permanentemente enviando su estado al multiplexor. Para leer el contenido de un determinado registro, se enviará el número de registro que se desea leer por el bus de direcciones, cuyas líneas se conectarán a las entradas de selección del multiplexor. En estos momentos, si el multiplexor se encuentra habilitado, las n salidas del multiplexor de n bits (n multiplexores de un bit), contendrán el dato que se encontraba en el registro.

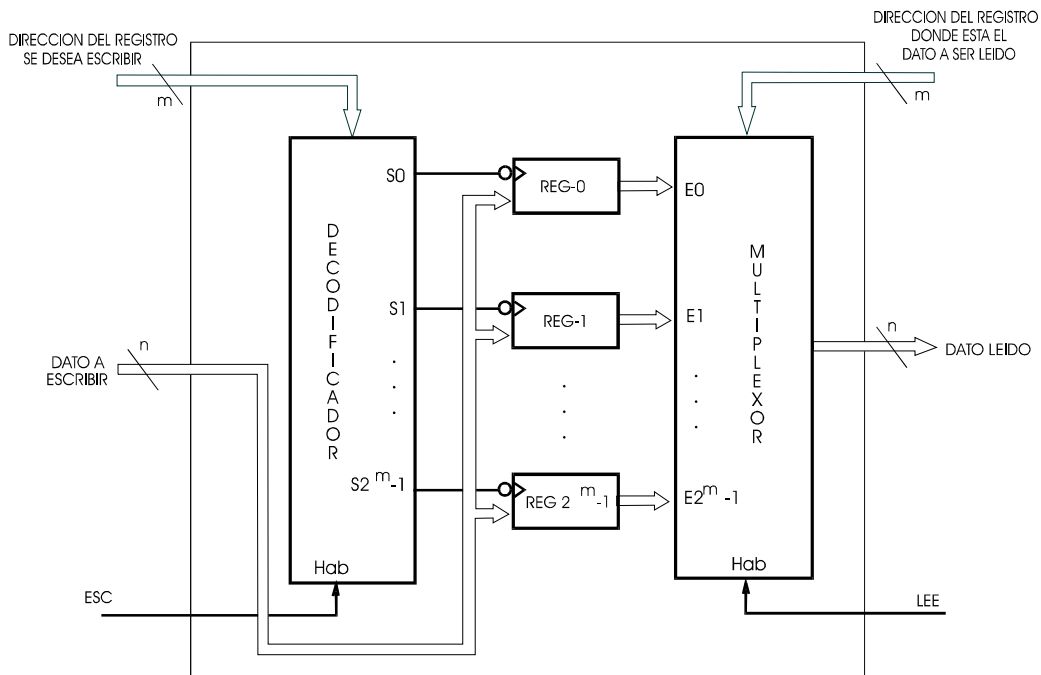


Figura 2. Implementación de un banco de registros utilizando un decodificador como puerto de escritura y un multiplexor como puerto de lectura

La ruta de datos del procesador que posteriormente estudiará el alumno, requerirá de un banco de registros con un puerto de escritura y dos puertos de lectura. Para añadir otro puerto de lectura al esquema representado en la Figura 2 simplemente habrá que conectar las salidas de los registros a otro multiplexor, que actuará como segundo puerto. La Figura 3 muestra las entradas y salidas del banco de registros a implementar.

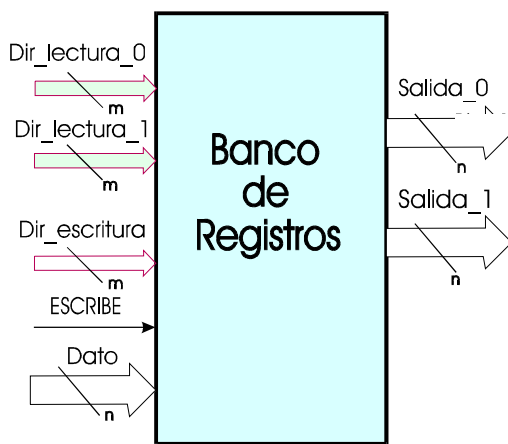


Figura 3. Entradas y salidas de un banco de registros, con un puerto de escritura y dos puertos de lectura

3. Realización de la práctica con CASCAD

3.1. Descripción del Programa

En los planes del 94 se utilizó ampliamente el programa CASCAD en las prácticas de estructura de computadores. El programa CASCAD es un entorno integrado para diseño y simulación lógica de circuitos digitales de mediana complejidad. No se trata de un paquete profesional para diseño de circuitos, sino que es una herramienta didáctica, con un tiempo mínimo estimado en unas tres horas para familiarizarse con el entorno. En esta fase de aprendizaje se planificaban una serie de ejercicios diseñados por el profesor y que permitían ir conociendo el programa mientras se construían elementos útiles para prácticas posteriores.

Esta práctica cumplía una doble función, por una parte servía para introducir al alumno en el

conocimiento de la herramienta, y además se implementaban circuitos que posteriormente se utilizaban en la construcción del procesador.

Los requerimientos Hardware del programa son mínimos, aunque hace algunos años estos eran un *handicap*, hoy en día han pasado a un segundo plano teniendo en cuenta las características de los equipos de laboratorio. Por otra parte su ejecución bajo entorno DOS, y su no disponibilidad en licencia libre representan algunos de sus mayores inconvenientes. La Figura 4 muestra el entorno de trabajo de la herramienta CASCAD.



Figura 4. Pantalla principal de CASCAD

Dado que CASCAD es un programa que se ejecuta en entorno MSDOS, para la selección de las distintas funciones del entorno, CASCAD utiliza un conjunto de menús organizados en secuencia. Para hacer aparecer los menús sobre la ventana de trabajo basta pulsar el botón derecho del ratón. El primer menú que aparece es el menú principal (*main menu*). La selección de cualquiera de sus opciones hace aparecer un segundo menú para dicha opción y así sucesivamente formando una cadena, como se muestra a continuación.

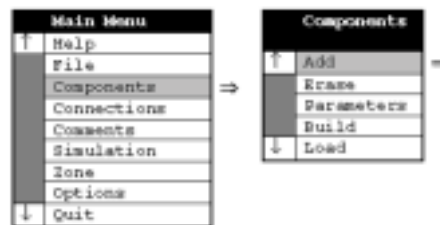


Figura 5. Secuencia de menús en CASCAD

3.2. Implementación del Banco de Registros

Como el CASCAD no ofrece ningún banco de registros como componente, se deberá construir conectando entre sí los diferentes elementos necesarios. Una vez construido, el CASCAD ofrece la posibilidad de encapsular el circuito realizado para su posterior utilización en circuitos de mayor complejidad.

Los pasos generales para la implementación de cualquier circuito en general y del banco de registros en particular son los siguientes: 1) Identificación de los componentes necesarios, 2) Realización de las conexiones; 4) Simulación y circuito de prueba del circuito. Teniendo en cuenta la descripción del banco de registros del apartado 2, la Figura 6 muestra una posible implementación de un banco de registros de ocho registros de 16 bits cada uno.

Como se puede observar, la operación de lectura se ha implementado con dos Multiplexores. La operación de escritura se ha implementado se ha realizado utilizando un Demultiplexor debido a que el CASCAD no incorpora Decodificadores. El alumno conoce la equivalencia existente entre Decodificadores y Demultiplexores.

3.3. Circuito de prueba

De cara a la simulación y prueba del banco de registros, la Figura 7 muestra el circuito de prueba que se debe implementar utilizando una Unidad Aritmética Lógica o ALU. Se puede observar como el banco de registros se ha encapsulado en un nuevo componente denominado banco de registros que se puede utilizar en un nuevo circuito. Esta característica de encapsular componentes es de especial importancia de cara a implementar librerías de componentes específicos. Finalmente hay que comentar que para un aprovechamiento de las sesiones de prácticas, debe realizarse una planificación concienzuda que requiere de un trabajo previo por parte de los profesores de la asignatura. En este sentido a los alumnos de primer curso se les suministra un proyecto inicial de partida en base al cual trabajan hasta obtener la versión definitiva.

Inicialmente se pedía al alumno que realizase todo el diseño completo; pero la experiencia nos demuestra que no es un buen método didáctico, ya que producía una pronta desmotivación en aquellos alumnos que no llegaban a obtener los resultados esperados.

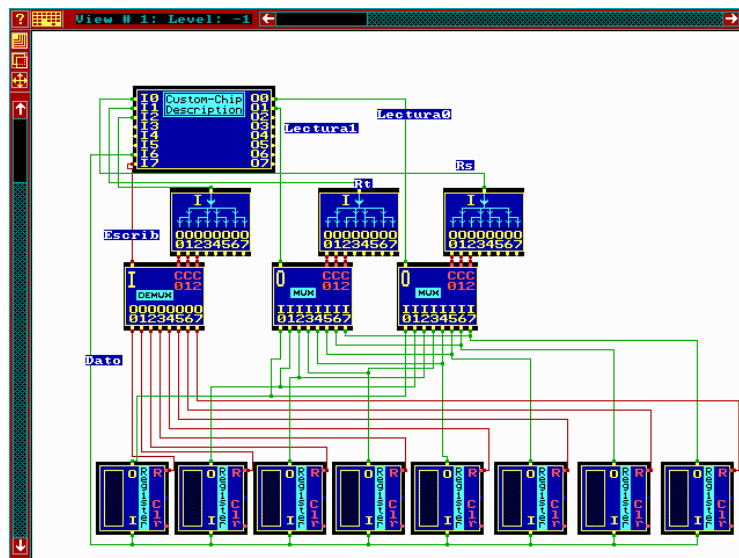


Figura 6. Implementación interna de un banco de ocho registros en CASCAD

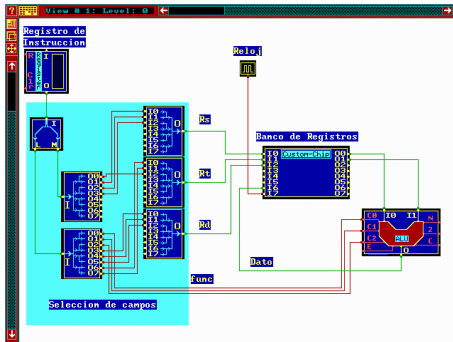


Figura 7. Circuito de prueba del banco de registros

4. Realización de la práctica con Xilinx

Debido a la importancia de la práctica, en nuestro bloque de prácticas, consideramos necesario que la práctica continúe realizándose. Sin embargo, nos encontramos en la necesidad de adaptarnos a las nuevas herramientas que ofrecen una mayor presencia y funcionalidad.

4.1. Descripción del Programa

El simulador XILINX *Foundation 2.1* de la compañía Xilinx es un entorno profesional integrado de diseño, simulación y síntesis de circuitos y Hardware digital. Esta herramienta incorpora el lenguaje estándar de descripción hardware VHDL, permitiendo tanto el diseño de circuitos y sistemas digitales completos sin depender de un fabricante específico, como su implementación directa en Hardware.

La utilización de una herramienta de tales características ha de planificarse de forma progresiva. En los planes de estudio de 2001, hay un compromiso de profesores de distintas asignaturas, para utilizar esta herramienta progresivamente, desde primeros cursos, hasta cursos avanzados de diseño VLSI, llegando incluso a proponer Proyectos Finales de Carrera donde se lleguen a utilizar sus características más avanzadas. De esta forma, el tiempo necesario para conocer y utilizar las características de la herramienta se va distribuyendo y amortizando en diferentes cursos, acorde con la formación progresiva del alumno. La Figura 8, muestra la pantalla principal de la herramienta. Para la implementación del banco de registros, solamente

utilizaremos el diseño esquemático y la simulación funcional, sin entrar en detalles específicos de implementación propios de cursos avanzados.

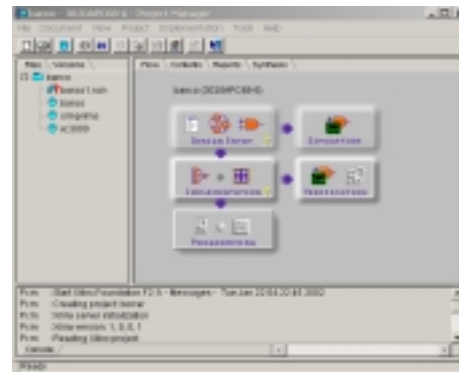


Figura 8. Pantalla principal de XILINX

4.2. Implementación del Banco de Registros

Al igual que sucedía con el programa CASCAD, el XILINX no incorpora un banco de registros. El primer paso es identificar los componentes necesarios en las librerías de XILINX. Para aquellos elementos que no se encuentren disponibles en las librerías por defecto, XILINX, incorpora una función denominada creación de módulos **logiBlox** que permite crear e incorporar como elementos de librería componentes específicos a partir de componentes genéricos. Para el diseño del banco de registros necesitaremos 1 decodificador de 3 a 8 para implementar la operación de escritura y 2 multiplexores de 8 entradas y 1 salida todas con un ancho de 16 bits. Tanto el decodificador como los multiplexores, se han creado utilizando módulos **logiBlox**. La Figura 9 muestra la creación de un Multiplexor de 8 a 1 con un ancho de bus de 16 bits a partir de un componente genérico Multiplexor.

La Figura 10 muestra la implementación del banco de registros de 8 registros de 16 bits, utilizando dos Multiplexores y el Decodificador implementados como **logiBlox**, y 8 registros de 16 bits que incorpora una de las librerías de XILINX. Para realizar diseños jerárquico, el XILINX permite la creación de nuevos componentes a partir de diseños

realizados. Una vez generado el nuevo componente, este se encuentra disponible como nuevo elemento de la librería del proyecto, pudiéndose utilizar en futuros diseños.



Figura 9. Creación de un multiplexor 8 a 1 de 16 bits

Para generar un nuevo componente necesitamos crear un símbolo a partir de un diseño ya existente. El símbolo consistirá en un nuevo componente donde el usuario sólo verá un bloque con el mismo número de entradas y de salidas que el esquema original, que englobará a éste y realizará su misma función. La Figura 11 muestra el editor de símbolos que incorpora XILINX.

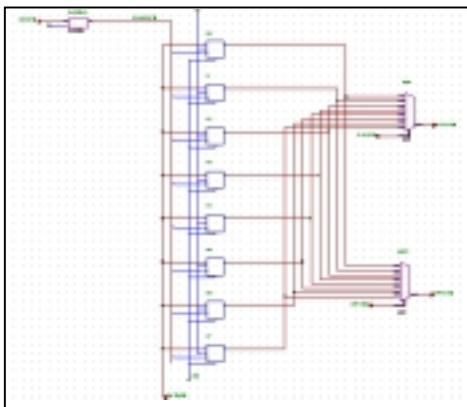


Figura 10. Implementación interna de un banco de ocho registros en XILINX.

La Figura 12 muestra el componente banco de registros. Como se puede observar, el editor de símbolos permite establecer una equivalencia

visual entre el interfaz del componente generado y el interfaz teórico que se presentó en la Figura 3 del apartado 2.

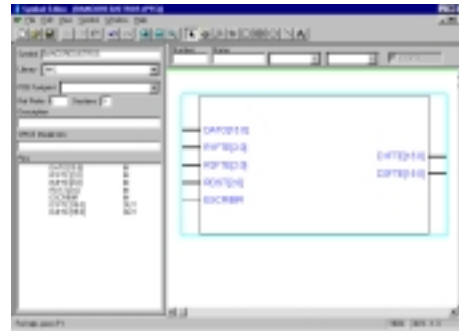


Figura 11. Editor de Símbolos de XILINX

Esta característica facilita notablemente la comprensión de los circuitos implementados, especialmente cuando la envergadura de un proyecto es considerable.



Figura 12. Componente Banco de Registros.

3.3. Circuito de prueba

Para comprobar el funcionamiento del banco de registros, El alumno dispone de un circuito de prueba similar al implementado con CASCAD. Una vez que el alumno ha comprobado el funcionamiento del circuito, se le proporciona un circuito de mayor dificultad como el mostrado en la Figura 13 y se le plantean una serie de cuestiones teórico/prácticas. El objetivo que se persigue es analizar el nivel de comprensión del funcionamiento del banco de registros cuando este se conecta con otros elementos, como son contadores y Unidades Aritmético Lógicas, elementos que se utilizarán posteriormente en el diseño de la ruta de datos de un procesador básico

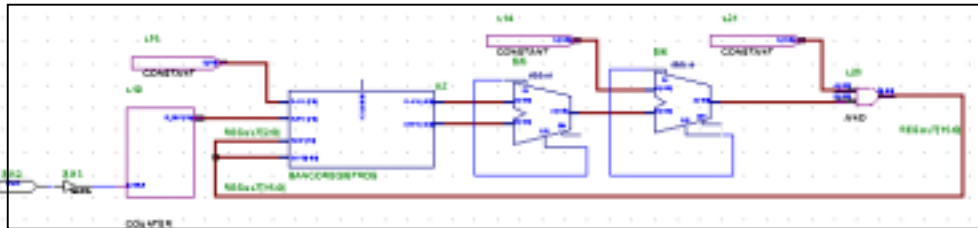


Figura 13. Circuito de prueba del banco de registros

5. Comparativa entre herramientas

En este apartado estableceremos una comparativa de ambas herramientas CASCAD (C) y XILINX (Xi), estableciendo un conjunto de características que se deben tener en cuenta de cara a seleccionar una u otra como herramienta de simulación de sistemas digitales.

- **Sistema Operativo:** (C) MSDOS. (Xi) Windows.
- **Requerimientos Procesador y RAM:** (C) 386, 1Mbyte. (Xi) Pentium III, 64Mbytes.
- **Espacio en disco:** (C) disquetes, (Xi) CDs.
- **Entorno del programa:** (C) Sencillo pero poco flexible. (Xi) Sencillo y flexible.
- **Comprobación y Simulación:** (C) Sencillo y poco flexible. (Xi) Complejo pero flexible.
- **Generación de nuevos componentes:** (C) Restringido. (Xi) Avanzado
- **Utilización en cursos avanzados:** (C) Restringido. (Xi) Aconsejable.
- **Entorno de uso:** (C) Educativo. (Xi) Educativo y profesional.

6. Conclusiones

En este trabajo se ha discutido el impacto que las nuevas herramientas de simulación han tenido y continúan teniendo sobre las prácticas docentes universitarias. El estudio se ha particularizado para la docencia en estructuras de computadores, aunque se puede generalizar a cualquier docencia universitaria.

La aparición de las nuevas herramientas, con una funcionalidad casi profesional, nos hace cuestionarnos si es adecuado continuar con las viejas prácticas y el nuevo software, o si sería preferible plantearnos la aparición de nuevas

prácticas que con el software anterior no eran factibles. Quizá la solución sea un mezcla de ambas alternativas, ya que muchas de las prácticas que vienen haciéndose durante años e incluso lustros, han sufrido las críticas y los retoques de muchos profesores, así como de varias generaciones de estudiantes. Algunas de estas prácticas tienen un valor añadido especial, sin embargo, es necesario que se adapten a las nuevas tecnologías si queremos mantener el prestigio y la calidad de las mismas.

En este trabajo no sólo nos hemos centrado en las cuestiones didácticas y metodológicas, sino que también se ha realizado un caso de estudio. A modo de ejemplo, se ha presentado la práctica del “Banco de Registros”, que llevamos haciendo durante años y que es la base de prácticas más complejas.

Finalmente recalcar, que creemos que la adaptación de nuevas prácticas es necesario tanto desde el punto de vista del alumno como del profesor. Desde el punto de vista del profesor porque siente satisfacción al realizar adecuadamente su función docente, y desde el punto de vista del alumno porque trabaja con una herramienta con apariencia profesional.

Referencias

- [1] Computer Assited Simulation for Circuits Análisis & Design (CASCAD), Manual de referencia, EduSoft.
- [2] Xilinx Foundation Series 2.1i Student Edtion software, <URL: <http://www.xilinx.com/>>.
- [2] David. E. Vandes, Logic Design with XILINX Foundation 2.1i.
- [4] J. Sahuquillo, H. Hassan, L. Lemus, J. Molero, R. Ors, F. Rodriguez, “Introducción a los Computadores”, Servicio de publicaciones de la UPV, 1997.