

# Actualización, Investigación y Desarrollo en el Entorno Docente de Arquitectura de Computadores en la Universidad de Extremadura

Juan A. Gómez Pulido, Juan M. Sánchez Pérez, Miguel A. Vega Rodríguez, José A. Moreno Zamora, David Rodríguez Lozano, Germán Galeano Gil, Julio Ballesteros Rubio

Departamento de Informática.  
Escuela Politécnica. Campus Universitario s/n. 10071 Cáceres.  
Fax: 927-257202 <http://atc.unex.es/gacdl>

## Resumen.

*En este artículo exponemos las experiencias docentes de diversos profesores que imparten docencia en una unidad temática formada por asignaturas del área de arquitectura de computadores, en la titulación de Ingeniería Informática de la Universidad de Extremadura. Resaltamos la innovación en la docencia a través de la implementación de laboratorios avanzados (dotados de hardware y software que son utilizados actualmente en el mercado tecnológico), el uso de simuladores de propósito específico (que cubren aspectos puntuales de la docencia, algunos de los cuales se han desarrollado específicamente con este fin), la actualización de los contenidos docentes (mediante la renovación continuada de los fondos bibliográficos y mediante la relación con otras universidades), y la conexión con la investigación (resultados de la investigación desarrollada que revierten en la docencia).*

## 1. Introducción.

En el Departamento de Informática (DI) de la Universidad de Extremadura (UEX) se ha configurado un grupo de docencia en asignaturas en las que la arquitectura de computadores es el área de conocimiento exclusiva. Este grupo está formado por seis profesores que imparten clases en varias asignaturas de la titulación en Ingeniería Informática (tabla 1), dentro del área de Arquitectura y Tecnología de Computadores (ATC) [1].

Curso	Caracter	Créditos	Alumnos
<b>Estructura de Computadores</b>			
2º	Obligatoria / anual	12	150
<b>Diseño Automático de Sistemas</b>			
3º	Optativa / cuatrimest.	6	80
<b>Arquitectura de Computadores</b>			
4º	Obligatoria / anual	9	92
<b>Diseño de Circuitos Integrados</b>			
5º	Optativa / cuatrimest	6	35
<b>3 Cursos de doctorado en arquitectura de comput.</b>			
I oct.	Plan de doctorado	18	15

Tabla 1

Inicialmente, al problema de la masificación del alumnado se añadía la escasez de medios materiales con los que afrontar una formación de calidad. Paulatinamente, este último problema se fue solucionando mediante la adquisición de software y hardware a través de los proyectos subvencionados que, tanto de infraestructura como de investigación, se han ido consiguiendo. Esto permitió, entre otras cosas:

- Implementar laboratorios con un hardware actualizado y con el número de puestos suficiente como para formar grupos de alumnos que se puedan atender cómodamente.
- Adquirir licencias de software especializado para el desarrollo de prácticas de calidad.
- Propiciar la creación de grupos reducidos de investigación y desarrollo.

En estas circunstancias, se abordaron los contenidos y metodologías docentes más adecuados en el ámbito de la enseñanza de arquitectura de computadores, y que fueran capaces de transmitir a los alumnos unos conocimientos actualizados sustentados por herramientas usadas en la industria. Estas estrategias se explican en los siguientes apartados.

## 2. Adecuación de los contenidos teóricos impartidos al estado actual de la Arquitectura de Computadores.

Debido al constante avance en esta área, es imprescindible que, sin perder de vista los fundamentos y bases que permanecen en el tiempo, se oferten unos contenidos teóricos que reflejen el estado actual, sin detrimento de una buena pedagogía. Así, es importante apoyarse en textos básicos que son continuamente reeditados, escritos por profesores de reconocido prestigio que tienen una fuerte vinculación con la investigación y que son referencia en la historia de la arquitectura de computadores [2][3][4][5][6].

También, el contacto que se mantiene con otros grupos de universidades nacionales y extranjeras, permite tener conocimiento del estado actual del área ATC, intercambiar experiencias docentes y coordinar objetivos de investigación, que se ven reflejados en la exposición de los temas teóricos de las asignaturas.

## 3. Innovación en las clases prácticas

La actualización de nuestras clases prácticas se apoya en dos pilares:

- Simuladores de propósito específico.

Numerosas prácticas se desarrollan utilizando simuladores que cubren aspectos específicos de la docencia. Algunos de estos simuladores han sido desarrollados a medida para este fin, con el consiguiente esfuerzo por parte del profesorado. Entre otros, cabe destacar:

⇒ Simulador de memoria caché multinivel [7][8], desarrollado en el DI de la UEX. Usado para realizar prácticas que evalúan sistemas de memoria caché: configuraciones, modelos, algoritmos de correspondencia, reemplazo, tasas de fallos, aciertos, rendimientos, jerarquías multinivel, etc. Los alumnos pueden, así, plasmar de forma práctica varios capítulos del temario teórico que con anterioridad no podían ser tratados de otra forma. En la **figura 1** puede verse el aspecto de esta aplicación mediante dos de sus pantallas.

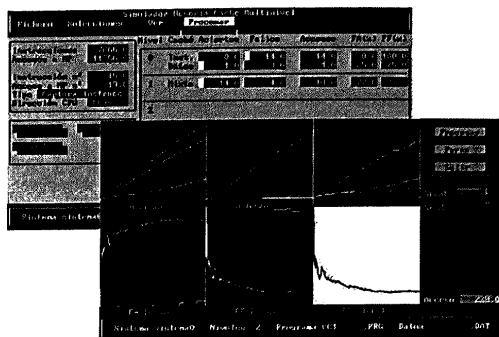


Figura 1

⇒ Simulador SPIM (**figura 2**) del ensamblador MIPS [9], desarrollado en la Universidad de Wisconsin. Es usado para programar en un repertorio de instrucciones RISC, con lo cual se ofrece al alumno una alternativa al repertorio convencional CISC de los procesadores basados en la familia 80x86.

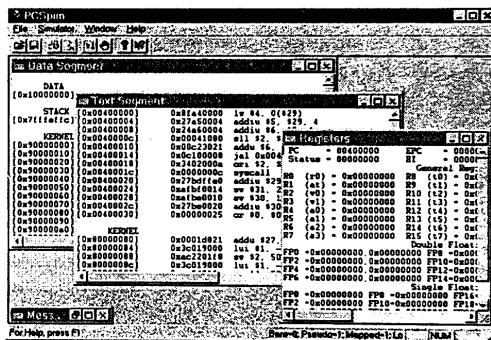


Figura 2

⇒ Simuladores (**figura 3**) de los procesadores monociclo, multiciclo y segmentado basados en un subconjunto del repertorio de instrucciones del procesador MIPS, según el texto [2]. Estos simuladores gráficos, desarrollados en la Universidad de Deusto, suponen un gran apoyo a buena parte de la docencia, pues se pueden tratar muchos conceptos teóricos del diseño hardware de los procesadores de una forma práctica.

⇒ Simulador (**figura 4**) del procesador basado en el texto [4]. Desarrollado en el DI de la UEX, es utilizado en parte del temario de la asignatura *Estructura de Computadores*, para ilustrar los aspectos del control microprogramado.

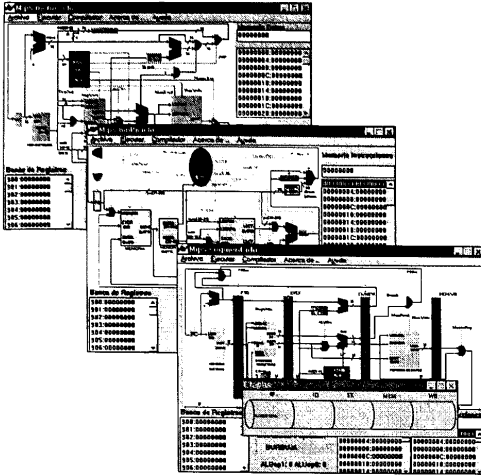


Figura 3

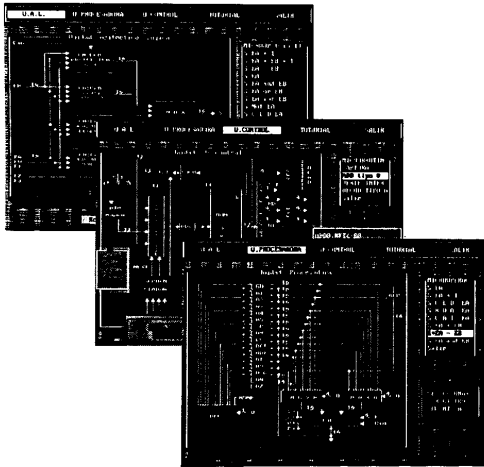


Figura 4

• Software de implantación industrial.

Este software, que es utilizado actualmente por las empresas para sus desarrollos dentro de sus ámbitos de actuación particulares, está siendo utilizado por los alumnos para desarrollar cierto tipo de prácticas, con lo que se les ofrece una formación que puede ser demandada en el mercado laboral. Cabe mencionar las siguientes herramientas:

⇒ Xilinx Foundation [10]. Se trata de un conjunto de herramientas que combinan software y hardware, y que permiten realizar el ciclo de diseño de arquitecturas digitales. Estas herramientas modelan

sistemas mediante lenguajes HDL (Verilog, VHDL, ABEL), construyen circuitos mediante captura de esquemas, simulan su comportamiento, etc. (figura 5). Lo más interesante de esta herramienta es su capacidad de sintetizar prototipos sobre FPGAs o CPLDs. De esta forma, por ejemplo, un alumno puede realizar el ciclo de diseño completo de un sencillo procesador, llegando a implementarlo físicamente. Para ello, en las prácticas que los alumnos desarrollan, se utilizan unas tarjetas (figura 6) que contienen una FPGA en la que se realiza un prototipo, el cual puede ser programado y posteriormente utilizado.

El software de Xilinx tiene dos versiones. La versión profesional está protegida mediante licencia, y lleva una llave hardware ("mochila") sin la cual no es posible utilizar todas las características de la herramienta. La versión de estudiante, asequible y económica, permite hacer diseños como en la versión profesional, pero limitando la capacidad de la implementación física. Así, la configuración de los laboratorios contempla un conjunto de licencias profesionales (para prácticas de envergadura) que son administradas por un servidor Windows NT, y un cierto número de puestos con versiones de estudiante para realizar, sobre todo, prácticas de programación VHDL.

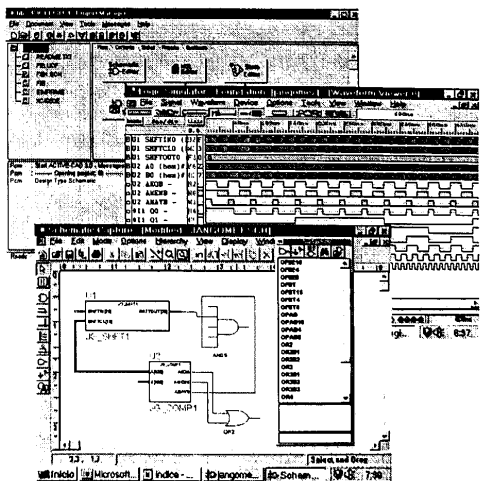


Figura 5

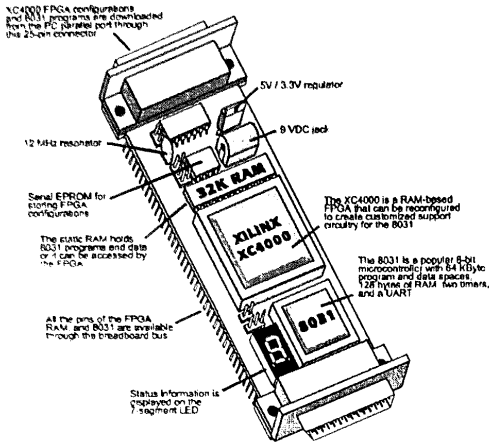


Figura 6

⇒ MATLAB y Simulink [12]. MATLAB (figura 7) es un lenguaje de programación de muy alto nivel orientado a cálculos matriciales, y Simulink (figura 8) es una interfaz gráfica que ayuda al diseño de procesadores digitales de la señal (DSP). Estas herramientas, en un principio destinadas a ayudar en la investigación que desarrollamos en el área, permiten al alumno de proyectos fin de carrera y de doctorado trabajar cómodamente en el ámbito del procesamiento de la señal y los sistemas discretos, para la consecución de prototipos de procesadores DSP. Mediante unas determinadas tarjetas hardware [13][14] (figura 9), los alumnos pueden implementar rápidamente prototipos electrónicos DSP sustentados por los algoritmos modelados con MATLAB y simulados mediante Simulink.

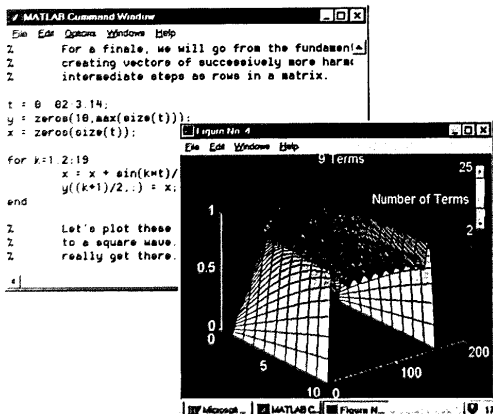


Figura 7

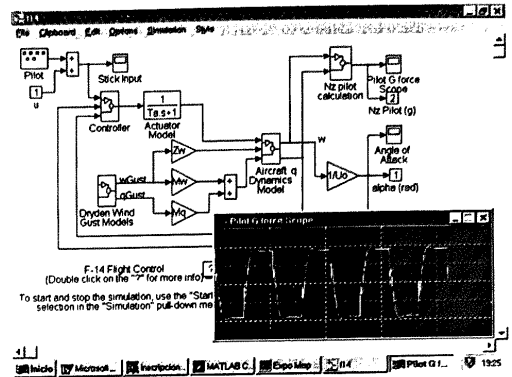


Figura 8

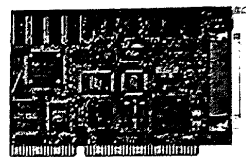


Figura 9

Al igual que en el caso de las herramientas de Xilinx, MATLAB y Simulink tienen versiones profesionales (protegidas por llave hardware) y de estudiante [15][16]. Estas últimas son muy asequibles, por lo que el alumno puede trabajar con estas herramientas al margen de los laboratorios existentes en la Universidad.

Es importante destacar que, al margen de los grupos de alumnos que realizan prácticas de las asignaturas ya mencionadas, la innovación en los laboratorios facilita la creación de grupos de I+D. La experiencia es que estos grupos de trabajo surgen y ofrecen una productividad científica adecuada cuando se ofrece un entorno de trabajo atractivo y con una buena dirección. Estos grupos, formados por profesores, becarios de investigación, alumnos de doctorado y alumnos de proyectos fin de carrera, trabajan en laboratorios dotados con hardware y software heterogéneo, pero de calidad:

- ⇒ Estaciones de trabajo Sun-UltraSparc.
- ⇒ Estaciones de trabajo Silicon graphics O2.
- ⇒ Computadores PC de altas prestaciones.
- ⇒ Tarjetas DSP, controladores, plataformas de síntesis con FPGAs y CPLDs.
- ⇒ Licencias CAD-EDA para el desarrollo de prototipos de arquitecturas especializadas.

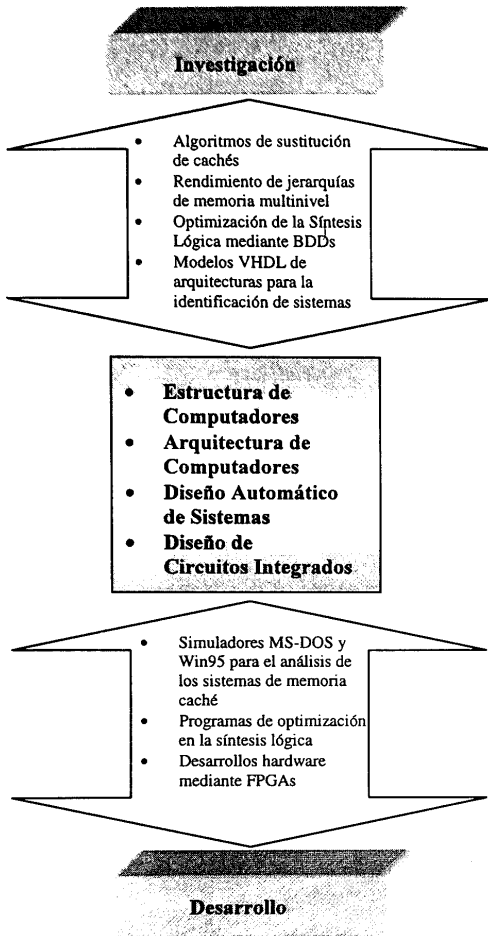


Figura 10

- ⇒ Licencias para el desarrollo de aplicaciones didácticas multimedia.
- ⇒ Licencias para el desarrollo de aplicaciones DSP, tratamiento de imágenes, control digital, redes neuronales y lógica borrosa.

**4. Conexión con la investigación y el desarrollo.**

En la Figura 10 queda resumida la conexión entre la docencia de las asignaturas de arquitectura de computadores y la investigación que desarrolla

el grupo de profesores involucrados en dicha docencia. Normalmente, los resultados que revierten en la docencia son consecuencia, y no causa, de las líneas de investigación.

Esta investigación [17] está soportada por varios proyectos y líneas de trabajo, entre las que destacan dos bloques temáticos con una evidente conexión con la docencia, tal como se refleja en la **Tabla 2:**

Líneas de investigación y desarrollo	Conexión con la docencia
Algoritmos y rendimientos en las jerarquías de memoria caché multinivel	Simulador (parcialmente reducido) utilizado para prácticas de memorias caché. Simulador para la generación de trazas de memoria.
Desarrollo de modelos VHDL de procesadores DSP basados en algoritmos genéticos para la identificación de sistemas, y síntesis de prototipos en FPGAs	Prácticas en las que se trabaja el ciclo de diseño de sencillas arquitecturas digitales utilizando las mismas herramientas.

Tabla 2

Un ejemplo concreto lo tenemos en el simulador DGT (depurador y generador de trazas de memoria de ficheros ejecutables MS-DOS). Esta aplicación (**figura 11**) fue desarrollada específicamente dentro de la línea de investigación para la evaluación del rendimiento de jerarquías de memoria caché multinivel, en la que se necesitaba generar las trazas de memoria de nuestros propios benchmarks, escritos en ensamblador del 80x86. El programa creado sirvió posteriormente para dar soporte a las prácticas de memoria caché, en la que los alumnos generaban las trazas de memoria de sus propios programas, y que posteriormente analizaban mediante el simulador SISMEC.

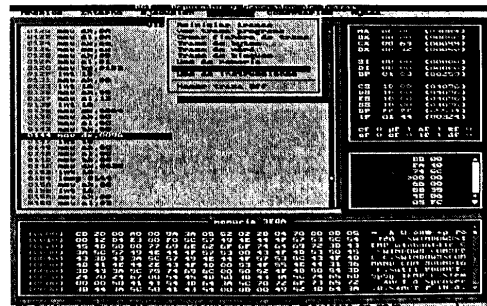


Figura 11

## 5. Referencias.

- [1]. Área de Arquitectura y Tecnología de Computadores de la Universidad de Extremadura. <http://atc.unex.es>
- [2]. Patterson, D and Hennessy, J. *Computer Organization and Design*, 2<sup>nd</sup> edition. Morgan Kaufmann. 1998.
- [3]. Stallings, W. *Computer Organization and Architecture*. 4<sup>th</sup> edition. Prentice Hall. 1996.
- [4]. Morris Mano, M. *Arquitectura de Computadores*. 3<sup>o</sup> edición. Prentice Hall. 1994.
- [5]. Hennessy, J and Patterson, D. *Computer Architecture: A Quantitative Approach*. 2<sup>nd</sup> Edition. Morgan Kauffman. 1995.
- [6]. Hayes, J.P. *Computer Architecture and Organization*. 3th edition. McGraw Hill. 1998.
- [7] Gómez Pulido, J. et al. "An Educational Tool for Testing Hierarchical Multilevel Caches". *Computer Architecture News*, ACM. Vol. 24, No. 4, September 1996.
- [8]. <http://atc.unex.es/gacdl/sismec.html>
- [9] <http://www.cs.wisc.edu/~larus/spim.html>
- [10] Xilinx, Inc. <http://www.xilinx.com>
- [11] Xilinx, Inc. *Xilinx Student Edition*. Prentice-Hall, 1998.
- [12] Mathworks. <http://www.mathworks.com>
- [13] HUMUSOFT. <http://www.humusoft.cz/>
- [14] Dspace. <http://www.dspace.de/>
- [15] Mathworks, Inc. *La edición de estudiante de MATLAB*. Prentice-Hall, 1998
- [16] Mathworks, Inc. *La edición de estudiante de Simulink*. Prentice-Hall, 1998
- [17] Grupo de Arquitectura de Computadores y Diseño Lógico. Universidad de Extremadura. <http://atc.unex.es/gacdl>