

# Guía visual y dinámica del funcionamiento de un procesador didáctico sencillo

Jon Cortés Ayala, Txelo Ruiz Vázquez, Izaskun Etxeberria Uztarroz

Dpto. de Arquitectura y Tecnología de Computadores  
Facultad de Informática, Universidad del País Vasco UPV-EHU  
20018 Donostia - San Sebastián

e-mail: [jon@sc.ehu.es](mailto:jon@sc.ehu.es), [txelo.ruiz@ehu.es](mailto:txelo.ruiz@ehu.es), [izaskun@si.ehu.es](mailto:izaskun@si.ehu.es)

## Resumen

En este trabajo se presenta una aplicación gráfica cuyo objetivo es proporcionar una guía visual y dinámica del funcionamiento, a nivel de los componentes *hardware*, de un procesador didáctico muy sencillo. Dicha aplicación puede ser utilizada tanto por el profesor en el aula, en lecciones magistrales, como por los alumnos de manera autónoma, para profundizar en la comprensión del funcionamiento del procesador. Por ello, cuenta con una interfaz visual atrayente y su utilización es fácil e intuitiva. Dispone también de ayuda *on-line* para resolución de dudas.

## 1. Motivación

Dentro de los tópicos que habitualmente se imparten dentro de una asignatura introductoria de Arquitectura de Computadores está el de la unidad de proceso o ruta de datos de un procesador relativamente sencillo [4] [5] [6]. La comprensión de su funcionamiento por parte del alumnado es fundamental para la adquisición posterior de conceptos más avanzados de Arquitectura de Computadores, como puede ser la segmentación de la ruta de datos.

En nuestra Facultad, en una asignatura introductoria de primer curso, denominada “Diseño de Sistemas Digitales” e impartida en el primer semestre, se presenta una versión reducida de un procesador sencillo [1] —diseñado por miembros de esta Facultad en base al procesador MIPS [4]— que es utilizado en asignaturas posteriores para introducir más conceptos de Arquitectura de Computadores. En dicha asignatura, planteada según un enfoque *bottom-up* [2], inicialmente se exponen los componentes básicos de un sistema digital, trabajándose exhaustivamente tanto el análisis como la síntesis de sistemas sencillos, y se con-

tinúa con una introducción al diseño de sistemas síncronos con unidades de control cableadas basadas en máquinas de estados finitas (algoritmos ASM). Como colofón de la misma, se realiza el diseño (guiado por el profesor) de un procesador sencillo, tipo RISC, con un reducidísimo conjunto de instrucciones (concretamente, cuatro) y sólo dos modos de direccionamiento: absoluto y directo de registro. Se diseñan tanto la Unidad de Proceso como la Unidad de Control cableada (mediante su especificación ASM).

El objetivo fundamental de dicho proceso de diseño contempla dos facetas: por una parte, que los alumnos sean capaces de analizar exhaustivamente el funcionamiento del procesador, distinguiendo todas las fases de ejecución de las instrucciones y dándose cuenta del funcionamiento concurrente de los componentes *hardware*; por otra parte, que adquieran la visión de conjunto necesaria para deducir qué componentes habría que añadir o modificar en la Unidad de Proceso para poder ampliar el conjunto de instrucciones del procesador con una instrucción determinada, así como cuáles serían los pasos de ejecución que debería realizar la Unidad de Control para obtener un funcionamiento correcto de la nueva instrucción añadida. Es obvio que la segunda faceta depende de la primera: si no se alcanza un conocimiento satisfactorio del funcionamiento del procesador, difícil será modificar correctamente el diseño dado para ampliar su funcionamiento.

Para evaluar la adquisición de ambas competencias, en el examen de la asignatura se plantea un ejercicio (habitualmente, con un peso de 2,5 puntos sobre 10) sobre el procesador sencillo, que recoge ambas facetas de manera independiente: por un lado, preguntas muy concisas y concretas sobre el funcionamiento del procesador, y, por otro, la realización de una nueva instrucción.

Dada la falta de costumbre del alumnado de primer curso a la hora de comprender en profun-

didad el funcionamiento a nivel *hardware* de un sistema digital de cierta complejidad, como es el sencillo procesador diseñado, surgía el problema de que un elevado número de alumnos no llegaba a alcanzar el nivel mínimo deseado, lo cual era un inconveniente en asignaturas posteriores.

Dada la amplia utilización de simuladores en la enseñanza de materias de Arquitectura y Organización de Computadores [8] [9] [3] y las ventajas que en el proceso de enseñanza/aprendizaje reportan [7], en el momento en que se diseñó el procesador sencillo, se realizó también su simulador correspondiente [1]. Dicho simulador es utilizado en la asignatura de primer curso “Lenguaje Máquina”, impartida en el segundo semestre, y es muy adecuado para trabajar los aspectos relacionados con la programación en lenguaje ensamblador, permitiendo visualizar los resultados obtenidos en cada paso de ejecución —contenido del conjunto de registros y de memoria principal—, pero sin hacer hincapié en el funcionamiento del *hardware*, que es el aspecto que interesa recalcar en la asignatura de Diseño de Sistemas Digitales.

Por ello, con el objetivo de paliar las deficiencias observadas, y con la idea de obtener mejores resultados, se pensó que sería de gran ayuda didáctica el contar con un simulador específico del procesador sencillo, que hiciera hincapié en el funcionamiento de los componentes *hardware* (PC, IR, UAL, conjunto de registros,...) y que presentara una interfaz gráfica “amistosa” y sencilla, de manera que pudiera ser utilizado tanto por

el profesor en el aula durante sus lecciones magistrales, como por los alumnos de manera autónoma, lo que les permitiría profundizar a su ritmo en la comprensión del funcionamiento del procesador, sin presiones.

Al tratarse de un procesador específico, no era viable la utilización de simuladores ya existentes, sino que exigía el desarrollo de un simulador específico, para lo que se recurrió a plantear un proyecto fin de carrera (PFC). Dada la limitación de tiempo, inherente al PFC, para el desarrollo de un simulador completo, que permitiera la posibilidad de ejecución de diferentes programas, se decidió reducirlo, de manera que lo que se ha obtenido es la guía visual dinámica del funcionamiento del procesador sencillo que aquí se presenta.

## 2. Estructura del procesador sencillo

La unidad de proceso del procesador sencillo (figura 1) consta de un conjunto de 32 registros de propósito general (CR), una unidad aritmético-lógica sencilla (UAL), el contador de programa (PC), el registro de instrucción (IR), y una serie de registros y circuitería auxiliar (básicamente multiplexores). Se considera una memoria principal de 64K x 16 bits. Los datos son de 16 bits, mientras que las instrucciones son de 32 bits (de ahí la división del registro de instrucción, IR, en dos partes, IR1 e IR2, ya que se necesitan dos lecturas en memoria).

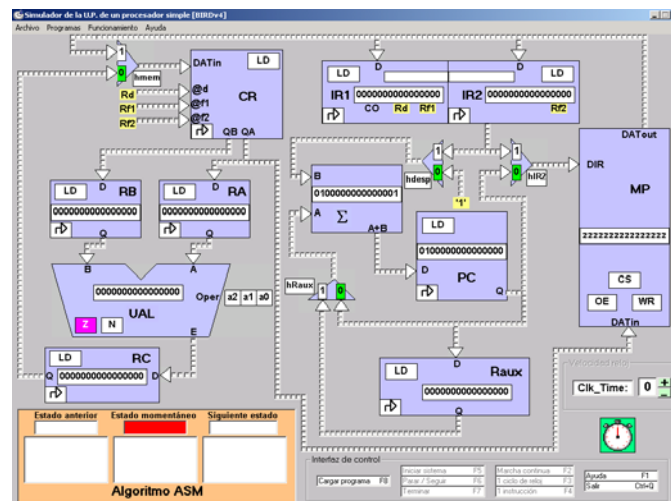


Figura 1. Estructura de la unidad de proceso del procesador sencillo (pantalla principal de la aplicación).

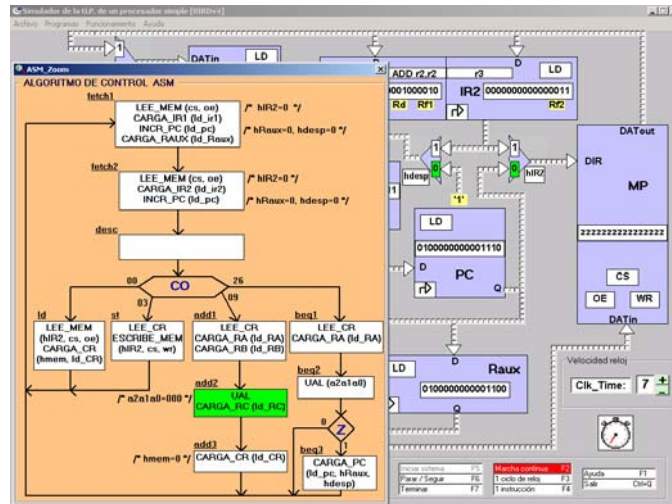


Figura 2. Resultado de hacer zoom sobre el algoritmo de control ASM.

En la figura 2 se puede ver el algoritmo ASM que refleja el funcionamiento de la unidad de control del procesador, tal como lo muestra el programa al hacer zoom sobre el recuadro de la parte inferior izquierda de la pantalla principal.

### 3. Características de la aplicación

La aplicación se ha desarrollado en el entorno Microsoft Visual Basic 6.0, sobre una plataforma PC

y el sistema operativo Windows XP. Se puede ejecutar en plataformas PC Pentium y en los sistemas operativos: Windows 9x, Milenium, NT 3.5 o superiores, 2000/ XP.

La pantalla principal (figura 3), permite ver el estado de cada componente de la unidad de proceso mediante una interfaz gráfica intuitiva, proporcionando toda la información necesaria para comprender el funcionamiento de la misma: señales de control que se activan en un momento dado; esta-

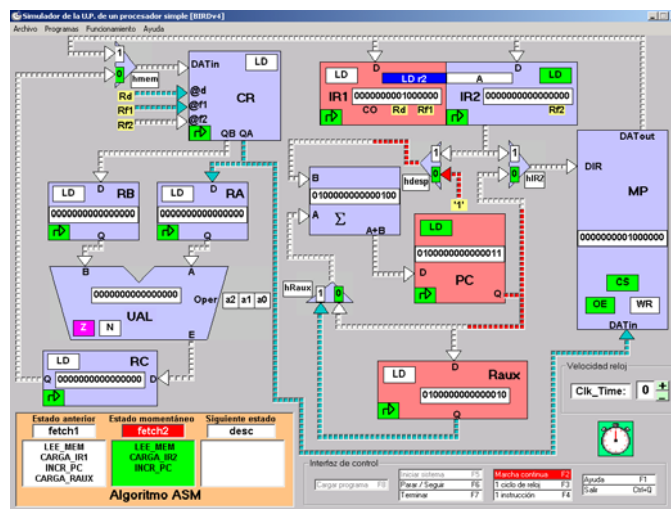


Figura 3. Evolución de la información a través de la unidad de proceso en un instante dado.

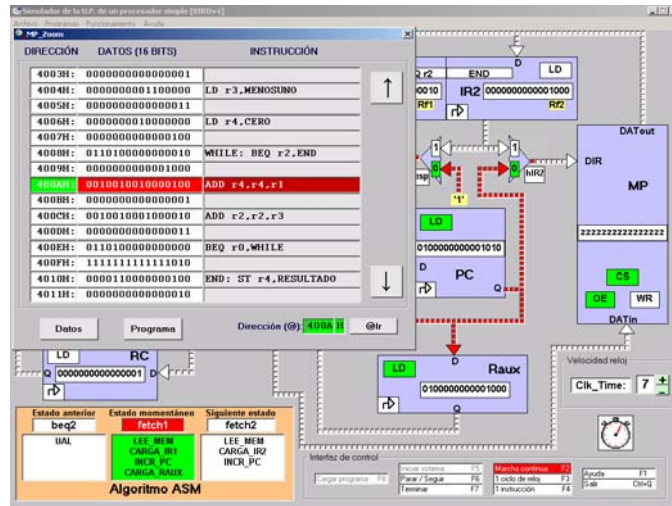


Figura 4. Resultado de hacer zoom sobre la memoria principal.

do en el que se encuentra el algoritmo ASM de control; y el camino seguido por la información al circular entre diferentes componentes.

Dicha información se va mostrando de manera dinámica, de forma que el usuario puede seguir intuitivamente todo el proceso de ejecución de cada instrucción y su efecto sobre la unidad de proceso.

Además, haciendo zoom bien sobre la memoria principal o bien sobre el conjunto de registros, se puede ver su contenido en cualquier instante de la ejecución del programa (figuras 4 y 5).

En cuanto a su funcionamiento, se toma como base un programa ejemplo que multiplica dos números, escrito usando únicamente las cuatro instrucciones del conjunto de instrucciones. El usuario puede elegir entre tres modos de simulación: continua, ciclo a ciclo, o una instrucción completa. Así, puede optar por el modo de simulación que mejor se adapta a su nivel de comprensión.

En esta línea, se ofrece la posibilidad de modificar la velocidad a la que evoluciona la información en la pantalla durante la simulación, así como

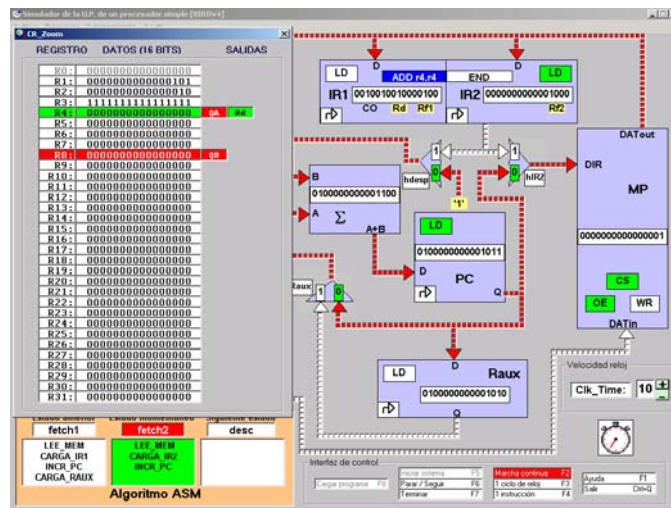


Figura 5. Resultado de hacer zoom sobre el conjunto de registros.

detenerla, para poder analizar detenidamente el resultado de alguna de las fases de ejecución, y posteriormente continuar con la simulación.

En [www.ehu.es/acwruvac/SIM/simc.htm](http://www.ehu.es/acwruvac/SIM/simc.htm) se puede obtener la aplicación (2,72 MB).

#### 4. Conclusión

En este trabajo se ha presentado un programa que proporciona una guía visual dinámica de la unidad de proceso de un procesador didáctico muy sencillo. Este programa ha sido realizado como proyecto fin de carrera por un alumno de la titulación de Ingeniería Informática. La experiencia adquirida por el alumno al realizar esta aplicación ha sido muy enriquecedora, ya que no sólo se ha tenido que enfrentar al desarrollo de un programa de cierta envergadura, sino que también ha tenido que tener en cuenta las características de los posibles usuarios. Además, el índice de satisfacción del alumno ha sido alto, ya que ha comprobado que su trabajo es de utilidad para ayudar a otros alumnos y facilitar su aprendizaje.

Por otra parte, la experiencia ha confirmado la validez de utilizar los proyectos fin de carrera como generadores de recursos docentes *ad hoc*, totalmente adaptados a las necesidades concretas de cada centro/asignatura/profesor, ya que se pueden tener en cuenta un sinnúmero de particularidades; en concreto, una de ellas es la posibilidad de que los recursos docentes estén en los idiomas utilizados por los alumnos (en el caso de la aplicación aquí presentada, está en castellano y en euskara).

La aplicación se utilizó por primera vez en la asignatura de Diseño de Sistemas Digitales durante el segundo semestre del curso 2003-04 y también en el primer semestre del curso 2004-05. El nivel de aceptación tanto del profesorado como del alumnado ha sido alto. Su utilización en el aula por parte del profesor se ha revelado como una potente ayuda de cara a la explicación detenida del funcionamiento del procesador, ya que permite hacer hincapié en los conceptos deseados: fases de ejecución de las instrucciones, funcionamiento concurrente de los componentes *hardware*, etc., pudiendo el profesor detener la simulación en cualquier instante y continuar cuando desee. Antes de la utilización del simulador, se recurría a la realización manual, sobre transpa-

rencias, de cronogramas explicativos, lo que solía ser bastante engorroso, tanto desde el punto de vista del profesor, como desde el de seguimiento del alumnado.

En cuanto a resultados acerca de su influencia en el nivel de comprensión de los alumnos medido en una prueba objetiva, como es el examen, en comparación con los resultados del curso anterior cabe destacar que la nota media obtenida en el ejercicio correspondiente ha mejorado en un 30% aproximadamente. Por otra parte, se ha reducido en un 15% el número de alumnos que obtenían en dicho ejercicio una puntuación inferior a 0,5 puntos (sobre los 2,5 que vale el ejercicio).

#### Referencias

- [1] Arbelaitz, O, Arregi, O., et alter. *Nivel de lenguaje máquina: una aproximación*. ATC-FISS. ISBN: 84-600-9519-3, 1999.
- [2] Clements, A. *The Undergraduate Curriculum in Computer Architecture*. IEEE Micro Special Issue on Computer Architecture Education, Vol. 20, No. 3, pp. 13-22, May/June 2000.
- [3] Djordjevic, J., et alter. *An Integrated Environment for Teaching Computer Architecture*. IEEE Micro Special Issue on Computer Architecture Education, Vol. 20, No. 3, pp. 66-74, May/June 2000.
- [4] Patterson, D.A., Hennesy, J.L. *Estructura y diseño de computadores. Interficie circuitería/programación*, Editorial Reverté, 2000.
- [5] Stallings, W. *Organización y Arquitectura de computadores, 5ª edición*, Prentice Hall, 2000.
- [6] Tanenbaum, A.S. *Organización de computadores. Un enfoque estructurado*, 4ª edición. Pearson Educación, 2000.
- [7] Tangorra, F. *The Role of the Computer Architecture Simulator in the Laboratory*. SIGCSE Bulletin, Vol. 22, June 1990.
- [8] Wolffe, G., et alter. *Teaching Computer Organization/Architecture With Limited Resources Using Simulators*. Proc. of SIGCSE 2002.
- [9] Yurcik, W., et alter. *A Survey of Simulators Used in Computer Organization/Architecture Courses*. Proc. of the Summer Computer Simulation Conference (SCSC), July 2001.