

EL DISEÑO DE UN PROCESADOR MIPS COMO GUÍA PARA EL APRENDIZAJE DE VHDL.

C. T. Medrano ¹, C. Catalán ², A. Blesa ¹, F. Serna ², F. Martínez ²

¹ Dpto. de Ingeniería Electrónica y Comunicaciones

² Dpto. de Informática e Ingeniería de Sistemas.

Universidad de Zaragoza

Escuela Universitaria Politécnica de Teruel

c/ Ciudad Escolar s/n

44003 Teruel

ctmedra@posta.unizar.es

Presentamos un grupo de prácticas cuyo objetivo es el diseño de un procesador MIPS utilizando VHDL como lenguaje de descripción textual. De esta forma el alumno, a la vez que aprende el lenguaje VHDL y un entorno de programación y simulación, percibe en el ámbito del laboratorio puntos comunes entre asignaturas de la titulación, reforzándose el concepto de unidad temática entre asignaturas de la carrera y abordando, desde varias asignaturas, objetivos didácticos más ambiciosos.

1. Introducción.

El microprocesador es el sistema digital de cierta complejidad que más se utiliza e influye en nuestra sociedad y actividad cotidiana. Por este motivo, existen ramas de la tecnología bien consolidadas que se dedican en exclusiva a este dispositivo. Su estudio se puede enfocar de forma diferente dependiendo de la titulación o del área de conocimiento que imparte docencia con contenidos relativos al mismo. En nuestra escuela se imparte la titulación de Ingeniería Técnica de Telecomunicaciones, especialidad sistemas electrónicos, y se aborda el estudio de los microprocesadores en varias asignaturas.

En la asignatura de Fundamentos y Arquitectura de Computadores, en el segundo curso de esta titulación, se enseña la arquitectura y la organización (en su versión no segmentada) del procesador MIPS [1]. Esta arquitectura ofrece un repertorio de instrucciones fácil de comprender y que se puede implementar en su versión no segmentada de una manera sencilla.

Por otro lado, una parte considerable del curso de Microelectrónica (tercer curso de la misma titulación) se centra en el lenguaje VHDL [2, 3]. Aprovechando los conocimientos adquiridos por el alumno en la asignatura de Fundamentos y Arquitectura de Computadores, se propone un conjunto de prácticas para la asignatura de Microelectrónica, cuya meta final sea el diseño de un procesador MIPS dotado de un subconjunto de instrucciones de la máquina [1].

arquitectura MIPS multiciclo no segmentada descrita en [1]. Esta implementación incluye un conjunto reducido de instrucciones: lw (cargar palabra), sw (almacenar palabra), add (suma), sub (resta), and, or, slt (inicializar sobre menor que), beq (salto si igual), j (salto incondicional). Los cambios que se realizan son sólo por conveniencia, pero no afectan a la filosofía de la máquina MIPS. Por ejemplo, el número de registros no debe ser necesariamente 32, ni las direcciones de memoria accesibles 2^{32} . Se ha realizado una implementación con 8 registros y 1K de memoria, lo que da un tiempo razonable de unos pocos segundos para la compilación del programa VHDL en MaxPlusII. El incremento del tamaño de la memoria o del número de registros, provoca un aumento considerable del tiempo de compilación. Por otro lado, para implementar la memoria, se ha hecho uso de los módulos parametrizados ofrecidos en el propio entorno MaxPlusII (lpm, library of parameterized modules), debido a la mayor facilidad con la que se compilan y a que permiten obtener mayores capacidades de memoria. El resto de los bloques no se ha realizado recurriendo a tales módulos, sino que se han programado por los mismos alumnos.

Como se observa en la figura 1, la construcción de la máquina MIPS se presta como ejemplo de diseño modular. Cada módulo por separado no presenta gran complejidad, pero con todos ellos se puede formar un sistema capaz de actuar como microprocesador. Desde el punto de vista del aprendizaje de VHDL, el sistema es adecuado por varios motivos: 1) contiene tanto sistemas simples (multiplexores) como complicados (ALU), permitiendo un aprendizaje progresivo; 2) Se incluyen sistemas combinacionales y secuenciales (las señales de control que aparecen en la figura 1 provienen de una máquina de 10 estados [1]); 3) Se hace casi imprescindible la creación de una librería de componentes, forzando al alumno a trabajar en un diseño modular.

El entorno MaxPlusII permite la simulación del sistema, inicializando la memoria con los datos e instrucciones en lenguaje máquina. Se puede observar la forma de onda de las señales definidas en el programa VHDL y la ejecución paso a paso de las instrucciones. El alumno debe comparar la simulación de un sencillo programa con el funcionamiento de la máquina MIPS aprendido en la asignatura de Fundamentos y Arquitectura de Computadores.

4. Descripción de las prácticas.

El conjunto de prácticas, cada una de ellas en sesiones de 2 horas, se puede estructurar de la siguiente forma:

- 1) Introducción al entorno MaxPlus-II y al lenguaje VHDL.
- 2) Sistemas combinacionales en VHDL.
- 3) Sistemas secuenciales en VHDL.
- 4) Diseño y test de los bloques del procesador (I).
- 5) Diseño y test de los bloques del procesador (II).
- 6) Creación de una librería de componentes. Diseño y test de la máquina MIPS.

Evidentemente el alumno antes de pasar por el laboratorio debe realizar un trabajo previo de análisis y diseño, en coordinación con la enseñanza teórica de VHDL, para poder obtener el máximo rendimiento en las dos horas que dura cada sesión.

El diseño del procesador es, además, bastante flexible y permite al profesor plantear modificaciones arquitectónicas que den lugar a distintas implementaciones, como por ejemplo, incorporar nuevas instrucciones o nuevos modos de direccionamiento.

En este contexto, uno de los problemas que puede plantear la estructura MIPS es la dificultad de que el diseño pueda ser programado en una PLD, debido a la gran cantidad de lógica que necesita. Al respecto conviene matizar que, aunque el diseño final sólo se simula, las primeras sesiones de prácticas no son específicas del procesador MIPS. En ellas el alumno puede tomar contacto con PLD sencillas (22V10, EP910). Por otro lado, alguno de los bloques de la máquina MIPS sí pueden caber en ese tipo de PLD, bien sea directamente (la máquina de estados), o reduciendo su tamaño (una ALU de 4 bits en lugar de 32, lo cual desde el punto de vista de programación no supone más que un pequeño cambio).

5. Conclusiones.

Para el aprendizaje de VHDL dentro de la asignatura de Microelectrónica, se propone un conjunto de sesiones de trabajo encaminadas al diseño de un procesador MIPS, estructura conocida por los alumnos en la asignatura de Fundamentos y Arquitectura de Computadores. Este es un ejemplo concreto de una filosofía de docencia que pretende romper alguna de las inercias clásicas en nuestras asignaturas: débil conexión entre programas, sensación por parte del alumno de las asignaturas como "compartimentos estancos" o de prácticas muy teóricas sin objetivos concretos. También se busca reforzar aspectos tales como el trabajo interdisciplinar (entre asignaturas de diferentes áreas de conocimiento) y romper la barrera temporal que supone el curso académico para abordar problemas y objetivos más ambiciosos.

Referencias.

- [1] D.A. Patterson y J.L. Hennessy, "Organización y diseño de computadores: La interface hardware -software", Ed. MacGraw-Hill, 1995.
- [2] K. Skahill, "VHDL for programmable logic", Ed. Addison Wesley, 1996.
- [3] F. Pardo y J.A. Boluda, "VHDL: Lenguaje para síntesis y modelado de circuitos", Ed. Rama, 1999.
- [4] A.B.Tucker (ed.) "Computer Curricula 1991" ACM/IEEE-CS Joint Curriculum Task Force, 1991
- [5] <http://www.altera.com>