

Laboratorio Web para Prototipado y Verificación de Sistemas Hardware/Software

Gómez-Arribas F J, González I, González J y Martínez J

Escuela Politécnica Superior, Universidad Autónoma de Madrid, España,
{Francisco.Gomez, Ivan.Gonzalez, Juan.Gonzalez, Javier.Martinez}@ii.uam.es
<http://www.ii.uam.es>

Resumen. En este trabajo se presenta un laboratorio *Web* basado en plataformas reconfigurables. El usuario sólo necesita un ordenador personal con acceso a *Web* para implementar un prototipo de un sistema digital, siendo posible además verificar su funcionamiento. El entorno permite trabajar en remoto con las herramientas de síntesis e implementación de un diseño digital en un circuito FPGA desde una descripción VHDL. También es posible la compilación en remoto de código fuente escrito en C para ser ejecutado en el procesador embebido de la plataforma. El laboratorio se ha probado con éxito para complementar los contenidos teóricos de un curso de postgrado. Los experimentos incluyen la realización de sistemas especializados que involucran: el estudio de la interface de los componentes hardware específicos y el procesador con el que intercambian datos, la verificación de un prototipo de procesador implementado en FPGA y el desarrollo de módulos hardware para aplicaciones en robótica. En todos los casos el usuario consigue una implementación de un prototipo del sistema completamente operativo para su utilización directa en aplicaciones prácticas.

1 Introducción

La gran difusión alcanzada por Internet y la evolución de las tecnologías para el desarrollo de aplicaciones *Web* ha hecho viable que un laboratorio experimental accesible en remoto a través de *Internet* sea un complemento ideal para el aprendizaje a distancia en materias con un fuerte componente práctico. Un laboratorio *Web* no pretende sustituir de ningún modo las prácticas de laboratorio *in situ*, que siguen siendo imprescindible en los cursos de ingeniería y ciencias, sino más bien descongestionar el laboratorio presencial y ofrecer experimentos adicionales. Otra ventaja radica en ofrecer una nueva perspectiva de enseñanza tanto a profesores como a los alumnos, ya que las prácticas se pueden realizar con un horario de trabajo personalizado por cada alumno, promoviendo de esta manera la responsabilidad en la organización del estudio y favoreciendo el autoaprendizaje. Las prácticas elegidas para este tipo de laboratorio pueden ser muy variadas. En este trabajo se propone la utilización de una plataforma reconfigurable accesible a través de un navegador *Web* para prototipar diseños digitales y verificar su funcionamiento.

El prototipado es una técnica de uso común en todas las ingenierías cuando los desarrollos son complejos, laboriosos y caros, o no están completamente especificados. Centrándonos en los dominios de aplicación de los sistemas digitales, un prototipo permite evaluar la viabilidad de diferentes soluciones propuestas para un problema. Pueden plantearse subsis-

temas formados por módulos *hardware* específicos, o por módulos programables que se adapten mediante software a la funcionalidad de las especificaciones. Un ejemplo de sistema digital que baraja ambas aproximaciones son los sistemas embebidos, formados por un procesador y circuitos FPGA. Desarrollando un prototipo en estas plataformas, se puede estudiar las ventajas de una solución completamente *hardware* frente a otra puramente *software* en términos de dificultad de diseño, recursos necesarios, tiempo de implementación, escalabilidad del diseño y adaptación al problema. También se puede comparar con una solución heterogénea basadas en codiseño *hardware-software*, siendo importante en este caso evaluar la interfaz entre ambas partes.

Las nuevas herramientas de diseño de circuitos digitales y los lenguajes de descripción de hardware, como por ejemplo VHDL y Handel-C, aproximan cada vez más la manera de diseñar a la programación. Aunque estas herramientas suelen utilizarse de manera interactiva, nada impide su operación en remoto ya que diseñar se basa en la correcta escritura de un código fuente de igual manera que en la programación tradicional, y por tanto solo se necesita para integrar su funcionamiento en un laboratorio Web controlar el envío de los ficheros de descripción del circuito y la devolución de errores de manera remota. En un laboratorio de sistemas digitales, este tipo de experimentos habían sido considerados hasta ahora impracticables para su realización remota y en términos docentes eran inviables para un aprendizaje a distancia. En este trabajo se presenta un sitio Web, junto con un conjunto de servicios y utilidades asociadas, que permiten el empleo en remoto de los recursos disponibles en la plataforma reconfigurable Labomat3 y se muestra de una manera práctica como se han realizado prototipos de sistemas digitales para complementar los contenidos de un curso de postgrado que trata sobre prototipado rápido de sistemas digitales y técnicas de codiseño.

2 Un Laboratorio Web basado en plataformas reconfigurables

El sitio Web desarrollado, <http://www.ii.uam.es/~laboweb/LabWeb/index.php3>, integra hasta el momento 2 plataformas reconfigurables de tipo Labomat3 [1] con una parte reconfigurable basada en circuitos FPGA y un procesador MC68360 embebido que permite la ejecución de programas de usuario, desarrollados en lenguajes de alto nivel.

Una descripción del resultado de la integración de todo un conjunto de servicios para permitir su operación en remoto, consiguiendo que todos los recursos de la plataforma sean accesible desde un navegador Web, ya fue presentada en un trabajo previo [2]. La solución desarrollada está basada en una arquitectura cliente-servidor que permite el funcionamiento en remoto del compilador de código fuente para aplicaciones a ejecutar en el procesador embebido, la carga de módulos HW diseñados con las herramientas del fabricante en los recursos configurables y la monitorización del funcionamiento conjunto de la parte HW y SW del sistema. Adicionalmente como mejora respecto a la versión anterior, el entorno de trabajo integra la operación en remoto de las herramientas de síntesis e implementación de un diseño digital en un circuito FPGA desde una descripción VHDL.

La principal razón de la elección para que Labomat3 sea parte integrante de laboratorio Web es su independencia, ya que dispone de su propio sistema operativo incluido con posibilidades de ejecución multitarea de programas SW y resuelve el interfaz de comunicación a través de Internet. Otra razón que hace de esta plataforma un candidato para su integra-

ción se debe a que los circuitos FPGA presentes en la misma, con el paso de los años, van quedando sin soporte en las últimas versiones del fabricante. Esto hace que al instalar nuevas versiones de las herramientas de diseño, la plataforma deja de ser válida como destino del diseño final. No obstante es indudable que mantiene su vigencia para la mayoría de las aplicaciones y en particular para su utilización docente. Esto se consigue solucionar sin demasiado coste de recursos ya que se puede mantener como servicios las diferentes versiones de los útiles de diseño en el servidor o servidores web y se utilizan estos servicios de compilación y síntesis de manera remota.

Se mantiene por tanto la filosofía de los laboratorios web descritos en la introducción, de tal manera que el usuario final accede a todas las posibilidades de la plataforma reconfigurable con solo disponer de un navegador Web.

3 Iniciación al codiseño con Labomatweb.

En la parte del curso dedicada al codiseño se ha propuesto manejar en remoto una de las plataformas Labomat3 disponibles, para implementar un coprocesador que interactúe con el procesador embebido y realice operaciones que no están implementadas en el juego de instrucciones. Básicamente se necesita emplear la FPGA XC4013E, el microprocesador embebido y las memorias DRAM y DPRAM. Como ejercicio previo se pide evaluar la eficiencia de la interfaz de comunicaciones para intercambio de datos entre el procesador y los recursos reconfigurables, así como los tiempos de acceso a las memorias.

3.1 Caracterización de los tiempos de acceso a los recursos de la plataforma.

Este ejercicio tiene como objetivo estudiar el tiempo empleado por el microprocesador para acceder a las distintas memorias disponibles en la plataforma y el acceso como periférico al dispositivo reconfigurable XC4013E, cuando realiza funciones de elemento de memoria.

Tipo de Memoria	Ciclos /operación	Tiempo (us)/operación	Valor relativo
DRAM	2.5	0.1	1
4KB DPRAM	24	0.9	9
Registros en FPGA	10	0.4	4

Tabla 1. Tiempo medio de un acceso 32 bit a memoria tanto en lectura como en escritura

Para llevar a cabo este experimento se sugiere implementar en la FPGA un contador con el objetivo de almacenar el número de ciclos de reloj empleados en la operación de escritura o lectura del dispositivo que se elija como elemento de memoria. En la FPGA como elemento de almacenamiento se añaden un par de registros en lógica reconfigurable, a los que se puede acceder como periférico del procesador. Se propone realizar tres pruebas considerando los accesos de lectura y escritura para cada tipo de memoria: DRAM, DPRAM y FPGA. Como datos de entrada, el programa de usuario permite seleccionar el número de operaciones de acceso a realizar y el tipo de operación, es decir, si se trata de lecturas o escrituras. El funcionamiento del programa consta de los siguientes pasos: activación del contador, realización el número de operaciones indicadas y parada del contador. El usuario recibe como información el número de ciclos de reloj empleados en ejecutar todas las ope-

raciones, almacenado en el contador, y su valor medio. En la tabla 1 se muestran los resultados obtenidos para los diferentes recursos disponibles en la plataforma Labomat3. Los valores medios coinciden para las operaciones de lectura y escritura considerando un número de accesos suficientemente grande. A partir de estos valores, se debe estimar en los siguientes ejercicios la penalización en rendimiento de los distintos diseños cuando se realicen con módulos hardware que intercambian datos con el procesador, y también se debe evaluar el coste en términos de rendimiento de las operaciones de acceso a memoria.

3.2 Módulo coprocesador para multiplicación en coma flotante

El primer prototipo que se propone para verificar su operatividad es una versión simplificada de un coprocesador que realice una operación de multiplicación en coma flotante y complemente el juego de instrucciones del procesador MC68360. Se propone un formato de números en coma flotante de 16 bits simplificado y el multiplicador recibe dos operandos como entrada en un registro. El diseño es completamente combinacional, y debe estar dividido en varios bloques que operen en paralelo implementados en la FPGA XC4013E para realizar las siguientes funciones:

- Normalización de las entradas: ambos operandos necesitan ser normalizados.
- Multiplicación de mantisas: el resultado de la multiplicación de mantisas (cada una de 10 bits) se guarda en un registro de 20 bits, con lo que no se produce *overflow*.
- Suma de exponentes: los exponentes ya normalizados de cada uno de los operandos se sumarán en un registro de 4 bits, aunque finalmente pasarán a uno de 5 bits, por el posible carry de la suma.
- Cálculo del signo: el signo se calculará simplemente haciendo un XOR de la posición 15 de las entradas, y será guardado en la salida.
- Normalización de la salida.

El programa de verificación que se ejecuta en el microprocesador consta de un bucle principal que solicita al usuario los dos operandos de entrada, los envía al multiplicador, lee el resultado y lo muestra al usuario. Para la verificación del correcto funcionamiento se compara también con el resultado de un programa codificado en lenguaje C, que implementa el algoritmo de multiplicación de dos números en coma flotante.

4 Labo-micro: Entorno para verificar prototipos de •procesadores

El entorno Labo-micro es una de las aplicaciones más interesantes que se pueden plantear en este tipo de plataformas: la verificación del diseño de un •procesador diseñado en VHDL e implementado en una FPGA. Para ello se utiliza un microcontrolador, concretamente el MC68360 embebido en la plataforma. Para facilitar la tarea de test se fijan las siguientes especificaciones: el •procesador experimental se basa en una arquitectura Hardvard y para implementar la memoria se utiliza la DPSRAM disponible en la plataforma que al mismo tiempo es accesible por el microprocesador embebido. Con ellos se permite conocer en todo momento el contenido de las distintas memorias que maneja el •procesador a verificar. Por simplicidad conviene dividir los 4KB de DPSRAM en dos mitades de 2KB, una para la memoria de datos y otra para la memoria de programa.

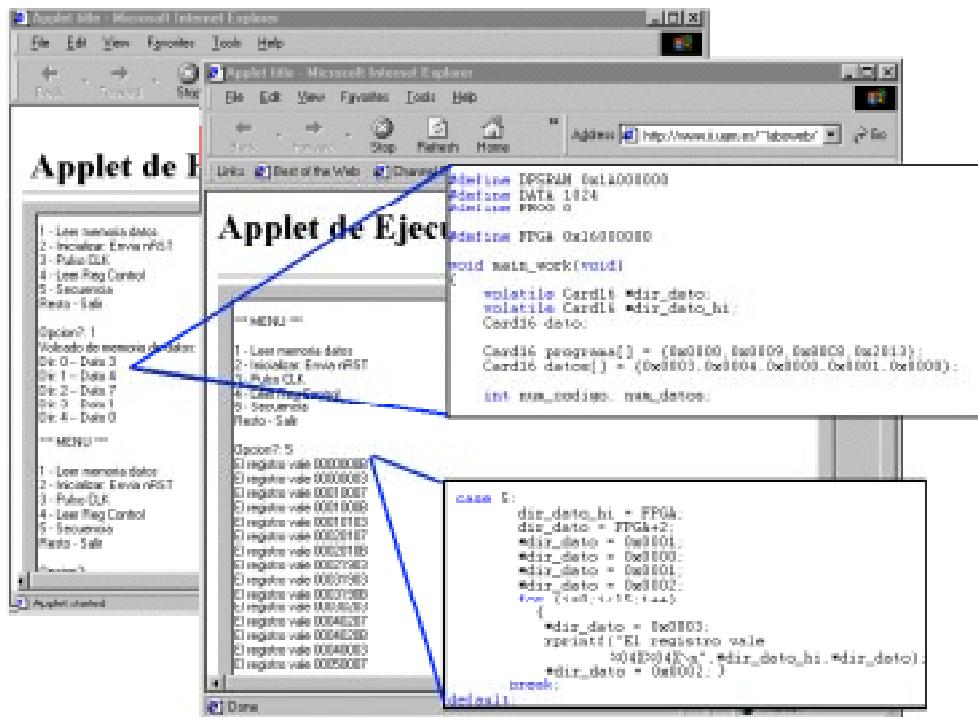


Fig 1. Entorno de verificación de microprocesadores implementados en FPGA.

La secuencia de operaciones a realizar una vez diseñado el prototipo es la siguiente:

- Cargar el diseño reconfigurando remotamente en la FPGA.
- Desarrollar un programa de test para ejecutarlo en el microcontrolador embebido que realice la carga de la DPSRAM con los valores correspondientes a la memoria de datos y de instrucciones que utiliza el procesador que va a ser verificado.
- Controlar la señal de reset y de reloj correspondientes al •procesador implementado en la FPGA y generar la señal de reloj paso a paso.
- Comprobar los resultados de la ejecución del código a través de los valores que se modifican de la memoria de datos implementada en la memoria de doble puerto

Estas opciones son accesibles mediante un menú y en la figura 1 se presenta el resultado mostrado remotamente en un applet después de realizar la secuencia que produce un reset al •procesador, ejecuta paso a paso un programa que suma los valores de las dos primeras posiciones de memoria, y almacena el resultado en la siguiente posición. El programa de test es una aplicación en modo texto muy efectiva aunque poco intuitiva. Para una descripción mas detallada del •procesador implementado y del interfaz de control, el enunciado del ejercicio y la descripción del interfaz de control se encuentra en la pagina Web [3] y en la referencia [4].

5 Labobot: Prototipado de sistemas de control

El objetivo principal de la aplicación Labobot es convertir la plataforma Labomat3 en un sistema de desarrollo para trabajar en robótica, y más concretamente aplicado al diseño de robots articulados. Para crear las articulaciones se emplean servomecanismos que se posicionan mediante señales PWM de 50Hz. Estas señales de control se suelen generar con microcontroladores de 8 bits. Mediante una plataforma reconfigurable, como la tarjeta Labomat3, se pretende delegar esta tarea en un hardware específico, liberando al microprocesador para realizar otros cálculos de más alto nivel.

Se propone como ejercicio el desarrollo un módulo en VHDL para la generación de las señales PWM para servos del tipo Futaba 3003. A través de un interfaz, el software que se ejecuta en el microprocesador debe posicionar los servos. Mediante unas librerías, el usuario debe poder controlar robots articulados sin tener que conocer los detalles de bajo nivel. Otro de los objetivos es optimizar el diseño hardware para determinar la cantidad máxima de servos que se pueden llegar a controlar, y así conocer el número máximo de articulaciones que pueden tener los robots diseñados.

Como aplicación práctica, se controlan 4 servos que permiten a dos minicámaras describir una secuencia de movimiento fija. Las especificaciones que debe satisfacer el prototipo son:

- Las señales de control de los servos son de 50Hz y una anchura del pulso comprendida entre 0.3 y 2.3ms (ver figura 2). Para su generación se emplea la unidad de PWM mostrada en la figura 3.
- El rango de movimiento del servo es de 180 grados. Se utilizan 8 bits para la posición, que está comprendida entre los valores 31 (un extremo) y 236 (el otro extremo). La precisión es de 0.88 grados.
- La posición deseada se almacena en un registro y se compara con el valor de un contador que determina el tiempo que la señal PWM permanece a nivel alto. La señal de reloj del sistema debe ser de 102.4 KHZ.

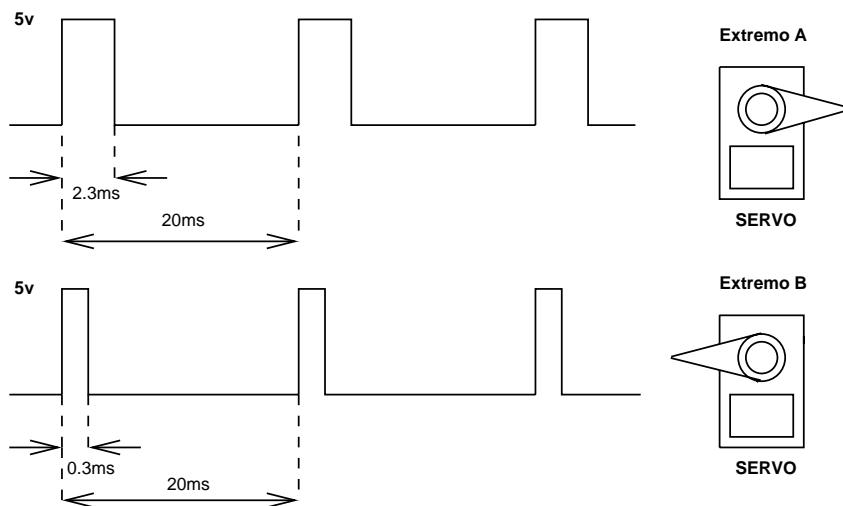


Fig 2. Señales PWM para controlar servos del tipo Futaba 3003
3 bits MSB

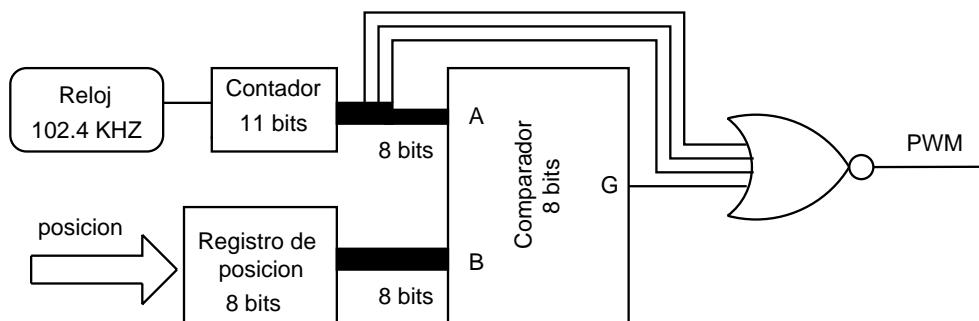


Fig 3. Esquema de una unidad PWM

El hardware y software desarrollado se ha aplicado al control y posicionamiento de los servos de un sistema compuesto por dos minicámaras (ver figura 4). Cada una con dos grados de libertad. En el software hay una tabla de control que contiene las posiciones en las que situar los servos así como el tiempo que deben permanecer en ellas. El programa recorre la tabla y cuando termina vuelve a comenzar desde el principio. Las cámaras describen una secuencia de movimiento que se repite

Como principal resultado obtenido cabe destacar que se ha diseñado el software y hardware mínimo necesario para poder trabajar con robots articulados. El número máximo de servos que se pueden conectar es de 28, lo que permite controlar robots bastante complejos. Se dispone del hardware en VHDL para la generación de las señales PWM y de las librerías en C para permitir que el usuario pueda controlar los servos sin conocer los detalles de bajo nivel.

Para trabajar remotamente a través de la web con aplicaciones robóticas, el usuario necesita ver el robot para saber si se está moviendo correctamente, y para ello se ha utilizado una webcam. Aunque el número de imágenes no permite un seguimiento en tiempo real, si consigue dar una idea del correcto funcionamiento de la secuencia de movimientos.

6 Conclusiones y trabajo futuro

Se ha presentado un sitio Web que permite la realización de prototipos de sistemas digitales y la verificación de su funcionamiento. Todos los procesos involucrados se realizan de manera remota y solo es necesario un ordenador con conexión Web para su utilización. La base para la realización del proceso de verificación es una plataforma reconfigurable para la que se han desarrollado una serie de aplicaciones accesibles vía Web que programan los recursos reconfigurables y el microcontrolador presente en la plataforma.

Consultando a los usuarios del laboratorio Web, el mayor aliciente de este tipo de herramienta es la implementación física del diseño en hardware y que no sea necesario trabajar con simuladores para realizar una verificación real del funcionamiento del prototipo que se ha diseñado.

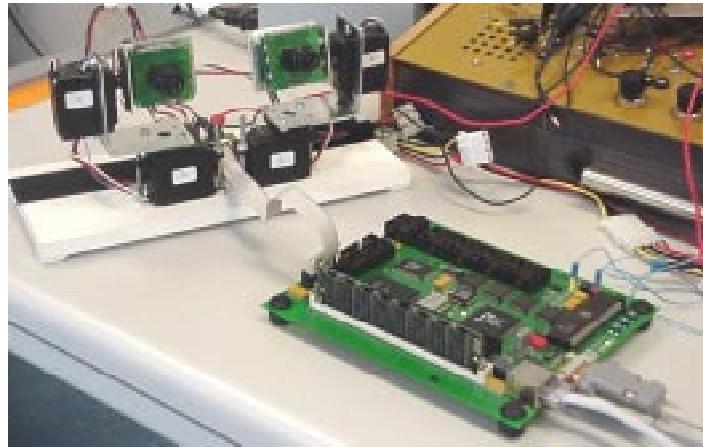


Fig 4. La tarjeta Labomat3 conectada al sistema de minicámaras

Como trabajo futuro está previsto el desarrollo de una aplicación gráfica de apariencia similar a las herramientas de diseño y debug comerciales para facilitar el proceso de verificación de procesadores. Esta herramienta permitirá visualizar el contenido de las posiciones de memoria además de introducir mejoras en la ejecución por pasos o por bloques funcionales del programa de test.

En breve el laboratorio Web dispondrá de otras plataformas reconfigurables como la tarjeta RC1000-PP de Celoxica y la plataforma Excalibur con el kit de desarrollo del procesador Nios de Altera y la herramienta Codenios [5] realizada en el laboratorio de sistemas lógicos del EPFL.

Agradecimientos

Este trabajo ha sido parcialmente financiado por los proyectos TIC2000-0464 y TIC2001-2688-C03-03 y del Ministerio de Ciencia y Tecnología.

Referencias

1. Teuscher C., Haenni J. O., Gomez F. J., Restrepo H. F., Sanchez E. "Labomat3: A Reconfigurable Platform for Academic Purposes". Proceedings of the IEEE Symposium Field-Programmable Custom Computing Machines. FCCM '99. Napa USA. April 1999. pp 282-283.
2. González I., Gómez F. J., Martínez J., "LabomatWeb: Recursos reconfigurables Remotos vía Word Wide Web". JCRA 2001, Alicante 2001, pp 102-109.
3. The Web Laboratory LabomatWeb <http://www.ii.uam.es/~laboweb/LabWeb>
4. González I., Venegas C., López-Buedo S., Gómez F., Martínez J., Garrido J. "Labo-Micro: Entorno de Test para la verificación de microprocesadores experimentales sobre circuitos FPGA". TAAE2002, Las palmas de Gran Canaria 2002, Vol I, pp 111-114.
5. Thoma Y., Sanchez E. "CoDeNios: A Function Level Co-Design Tool in Workshop on Computer Architecture Education". WCAE 2002, Anchorage - Alaska 2002, pp.73-78.