

Proyecto de co-diseño HW/SW de un sistema digital con prototipado en FPGA

José A. Moreno Zamora*, José V. Valverde Sánchez and Francisco J. Álvarez García
Escuela de Ingenierías Industriales de Badajoz
Universidad de Extremadura, 06006 Badajoz, Spain
Email*: josan@unex.es

Resumen—En esta comunicación se plantea la aplicación del aprendizaje basado en proyectos mediante la realización del co-diseño hardware/software de un sistema multiprocesador de aplicación específica, como trabajo final de una asignatura optativa de intensificación en Electrónica Industrial y Automática. El sistema utiliza un procesador diseñado a medida y otro configurable a partir de módulo IP con periféricos de propósito general y otros especializados, que se han ido desarrollando en las sesiones prácticas.

I. INTRODUCCIÓN

Este trabajo se enmarca dentro de la asignatura “Diseño Digital” del Grado en Ingeniería en Electrónica y Automática (Rama Industrial) de la Escuela de Ingenierías Industriales de Badajoz, Universidad de Extremadura. Esta asignatura se encuadra dentro del módulo de optatividad de la materia de “Intensificación en Electrónica Industrial y Automática” a continuación de la asignatura de Electrónica Digital.

Los descriptores de la asignatura orientan el contenido de la misma hacia el estudio de “Circuitos digitales de aplicación específica: PLD’S, FPGA’S, SC. Lenguajes de descripción hardware. Adquisición, tratamiento y procesado de datos. Co-diseño hardware-software.”. Al ser una asignatura del último curso de una titulación, que conforma un perfil especializado de un ingeniero a desarrollar su labor en un marco tecnológico muy evolutivo, los contenidos que aquí se imparten deben ser acordes al posible entorno de trabajo de estos ingenieros. Estas particularidades se plasman en el programa de esta asignatura dirigiendo el mismo a contenidos teóricos con aplicación práctica directa y contenidos prácticos que permitan al alumno acostumbrarse a un entorno de trabajo real, orientado en este caso al diseño e implementación de sistemas digitales integrados.

Se pretende con esta nueva *metodología* y los contenidos asociados capacitar a los alumnos para aplicar sus conocimientos a la resolución de problemas de diseño electrónico concretos, para comunicarse adecuadamente en el mundo profesional y para lograr buenas bases para la autoformación.

Las *competencias* relacionadas con esta materia que se trabajan con esta metodología son las siguientes:

- Resolución de problemas relevantes, relacionados con situaciones reales y acordes a los intereses y necesidades de los alumnos.
- Capacidad de auto-organización, determinación de objetivos, planificación, realización y control.

- Investigación, consulta y discusión de alternativas
- construir, diseñar, componer, adaptar, seguir o establecer estándares.
- Desarrollo de prototipos, hacer pruebas, evaluar, propiciar la reingeniería del proyecto.
- Trabajo en equipo, incluyendo la asignación de roles a los integrantes.
- Exposición de resultados al conocimiento, la valoración y la crítica.
- Calidad y razonamiento crítico.

En el desarrollo de las actividades se establecen puntos de control y evaluación de cada alumno, del profesor, y del grupo en total por fases específicas, concretamente en la planificación, investigación, análisis, diseño, informe de resultados y evaluación.

I-A. Contenidos de la asignatura

La asignatura se estructura en cinco unidades temáticas con prácticas relacionadas:

- Unidad Temática 1. Circuitos digitales.
- Unidad Temática 2. Diseño lógico.
- Unidad Temática 3. Procesamiento de datos.
- Unidad Temática 4. Entrada y salida.
- Unidad Temática 5: Codiseño hardware/software

El contenido resumido de las mismas y su dedicación temporal teórica (T) y práctica (P) queda reflejado en la Tabla I.

Los contenidos prácticos de esta asignatura se desarrollan en torno a un proyecto práctico de diseño con cuatro fases programadas consecutivas, distribuidas a su vez en tareas, ponderadas con un número determinado de sesiones prácticas. Se pretende con ello, que el alumno ponga en práctica de forma progresiva los conocimientos que vaya adquiriendo en las clases teóricas, centrando su atención en un único proyecto que tiene todas las componentes de un proyecto real, dando así homogeneidad a su trabajo. Se fijan unos tiempos límite para la finalización de cada fase y se realiza un seguimiento continuado del desarrollo de las mismas.

Con anterioridad al comienzo del proyecto se introducen tres prácticas tutorizadas orientadas al uso de las herramientas utilizadas en su desarrollo, que en este caso incluyen tanto hardware como software de diseño asistido, así como cuatro prácticas que formarán parte como módulos del proyecto final, por lo que el programa práctico queda de la siguiente forma:

Tabla I
CONTENIDOS TEÓRICOS Y PLANIFICACIÓN

Unidad	T	P	Contenido
Circuitos digitales	2	1	Introducción a los circuitos digitales integrados. Lógica CMOS. Lógica programable. Ciclo de diseño con los circuitos digitales integrados.
Diseño lógico	10	5	Herramientas y entornos integrados de diseño digital. Metodologías de diseño ascendente y descendente. Síntesis e implementación de sistemas combinacionales. Síntesis e implementación de sistemas secuenciales. Bancos de prueba para la simulación funcional y temporal. Gestión del diseño.
Procesamiento de datos	6	6	Procesadores: Repertorios de instrucciones. Modos de direccionamiento. Diseño del camino de datos. Segmentación. Diseño de unidades de control. Técnicas de secuenciamiento. Aritmética avanzada: Sumadores rápidos. Multiplicadores. Aritmética de punto flotante. Memorias: ROM, RAM, estructuras LIFO y FIFO.
Entrada y salida	5	6	Control de periféricos: Controladores gráficos. Controladores de memoria externa. Comunicaciones: Transmisión de datos. Detección y corrección de errores. Protocolos.
Codiseño hardware/software	4	3.5	Diseño SoC, SoPC y CSoC. Módulos IP y cores. Arquitecturas SoC. Simulación avanzada. Diseño físico. Emulación y prototipado.

- Práctica 1. Entorno de diseño con dispositivos lógicos programables.
- Práctica 2. Diseño jerárquico y modular.
- Práctica 3. Diseño de procesadores.
- Práctica 4. Entorno de codiseño Xilinx EDK.
- Práctica 5. Controladores de video y comunicaciones serie.
- Práctica 6. Verificación HW/SW.
- Proyecto de codiseño HW/SW de un sistema digital con prototipado en FPGA y generación de ASIC.

I-B. Planificación de la asignatura

En la planificación de la parte teórica fueron necesarios ciertos ajustes temporales para coordinar la teoría y la práctica, por lo que se llevo a cabo una distribución no totalmente uniforme de las horas semanales en laboratorio. La tabla II resume la distribución temporal de las prácticas en laboratorio (L) y las actividades teóricas de grupo grande (GG), tutorías programadas (TP) y estudio personal (EP) a lo largo del cuatrimestre en que se desarrolla la asignatura de Diseño Digital.

II. DESCRIPCIÓN DEL TRABAJO

La docencia práctica de la asignatura conlleva la utilización de herramientas de diseño digital basado en lenguajes de descripción hardware y programación de microprocesadores, así como la implementación en plataformas de prototipado basadas en FPGAs, para lo cual se optó por Xilinx [1] por las facilidades de acceso a las herramientas y plataformas a través de su programa universitario.

El contenido teórico de la asignatura comprende las siguientes

Tabla II
ACTIVIDADES FORMATIVAS

Horas por unidad		Presencial					No presencial
Unidad / Eval	Total	GG	S	O	L	TP	EP
0	1,5	1					0,5
1	8	2			1		5
2	35	10			5		20
3	32,5	6			5	1,5	20
4	30	5			5		20
5	31	4			3,5	1,5	22
Evaluación	12	2					10
Total	150	30	0	0	19,5	3	97,5

materias: circuitos digitales de aplicación específica, lenguajes de descripción hardware, adquisición, tratamiento y procesamiento de datos, así como co-diseño hardware-software. En el desarrollo práctico de la asignatura, que corresponde a 20 horas distribuidas en 10 sesiones de laboratorio, una primera sesión tutorizada, corresponde a la utilización de plataformas y herramientas del fabricante. En un segundo bloque se introduce al alumno al procesamiento digital de señal, así como al diseño de procesadores y sistemas. El tercer bloque práctico corresponde al diseño de periféricos y comunicaciones. Un último bloque, que hace uso de los diseños anteriores a modo de proyecto final, para la realización de un sistema multiprocesador.

El proyecto consiste en la integración del procesador diseñado por el alumno con dos periféricos básicos: una salida LCD y una UART, por otro lado, un sistema complejo configurado a partir de módulos IP, incluyendo periféricos básicos: GPIO, VGA, UART, y por otro un periférico de diseño específico unido a este último. Ambos subsistemas basados en microprocesador se comunican en modo bidireccional a través de UART para la reconfiguración de parámetros del periférico especializado.

II-A. Práctica 1 - Entorno de diseño con dispositivos lógicos programables

En la primera sesión práctica se proporciona al alumno la plataforma hardware utilizada a lo largo del curso como soporte de emulación, prototipado y verificación de los distintos sistemas a diseñar. En este curso se optó por la plataforma basada en dispositivos lógicos programables Xilinx Spartan 3E Starter Board [1] [2] mostrada en la Figura 1, que se obtuvo por donación del fabricante mediante su programa universitario XUP [3].

Para hacer uso de la misma, e introducir el entorno de diseño con dispositivos lógicos programables de este fabricante, se lleva a cabo, de forma tutorizada, la implementación de un pequeño circuito digital basado en puertas lógicas básicas, que con una combinación de interruptores permite encender un diodo led en la placa de acuerdo a la descripción en Verilog HDL de la expresión booleana del circuito combinacional multinivel representado en la Figura 2.

Los pasos seguidos en el tutorial incluyen la creación de un nuevo proyecto, los archivos y jerarquía de diseño, una simulación a nivel comportamental (Figura 3), los procesos a

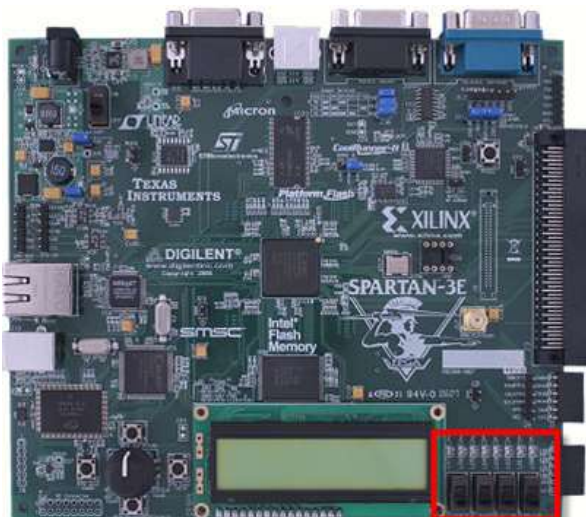


Figura 1. Plataforma de prototipo FPGA Spartan 3E Starter Board

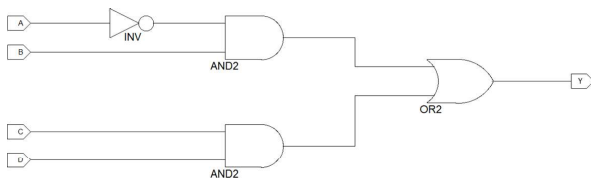


Figura 2. Circuito combinacional simple del tutorial

ejecutar para la implementación (Figura 4), la configuración de la FPGA y la verificación final en placa.

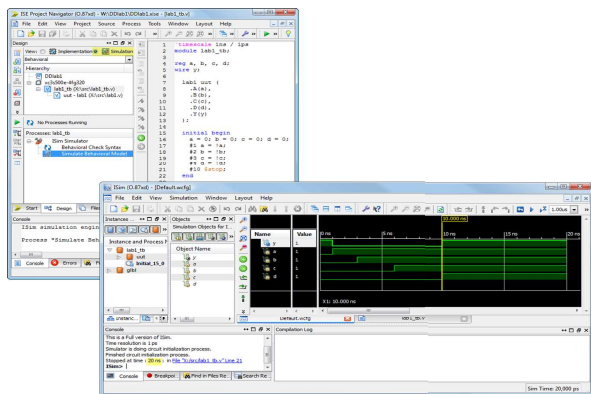


Figura 3. Banco de pruebas y simulación comportamental

II-B. Práctica 2 - Diseño jerárquico y modular

Este ejercicio práctico tiene como objetivos el diseño de módulos estructurales e instanciación múltiple en Verilog HDL, la síntesis de operadores aritméticos y elementos secuenciales básicos, y una introducción al procesamiento digital de señal con el diseño de filtros FIR, siguiendo el

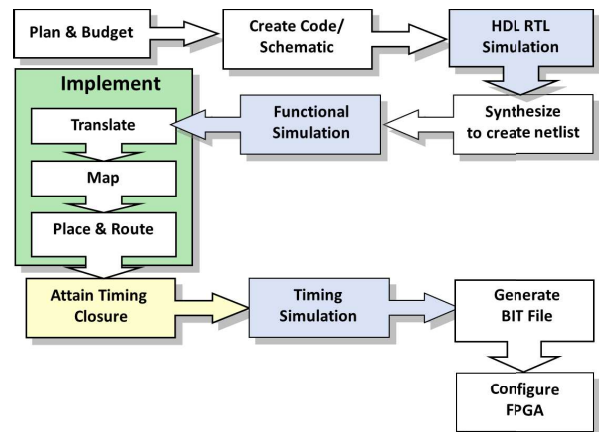


Figura 4. Flujo de diseño con Xilinx ISE

flujo de diseño ya conocido y la emulación/prototipado con la plataforma hardware.

En una primera parte del enunciado se propone el diseño de una ALU (unidad aritmético-lógica) de 3 bits basada en lógica de acarreo serie a partir de módulos estructurales de 1 bit. Los operandos de entrada a la ALU serán 2 datos de 3 bits que provienen del codificador rotatorio que actúa como contador síncrono ascendente/descendente (derecha/izquierda) y cuyo valor en curso será representado en leds, cada valor se confirmará con el pulsador del codificador y se guardará en los correspondientes registros. Las operaciones a realizar vendrán determinadas por 3 switches y corresponden con una ALU diseñada para dar soporte al microprocesador MIPS que se verá en la práctica siguiente según [4]. Los resultados de salida calculada se mostrarán en leds junto a las indicaciones de desbordamiento aritmético y resultado 0.

La segunda parte de la práctica implica diseñar un filtro digital FIR de orden 4 con una estructura convencional directa 6 haciendo uso de recursos de multiplicación paralela especiales en el dispositivo FPGA para la implementación de estructuras MAC.

Se utilizarán unos coeficientes constantes $h(k)$ elegidos arbitrariamente, y una entrada $x(n)$ de 3 bits partir de 3 switches confirmada con el pulsador del codificador rotatorio, para obtener la salida $y(n)$ en 8 leds, un tercer switch hará de selector entre la operación de la ALU y el FIR.

II-C. Práctica 3 - Diseño de procesadores

El diseño digital de un microcomputador embebido sencillo con un microprocesador RISC segmentado de aplicación específica constituye el propósito principal de la tercera práctica. Se incluye también la síntesis de memorias con módulos parametrizables e inicialización de contenido utilizando la herramienta de generación Xilinx Core Generator, y la emulación/prototipado del sistema en la plataforma hardware.

Se realiza una implementación segmentada del

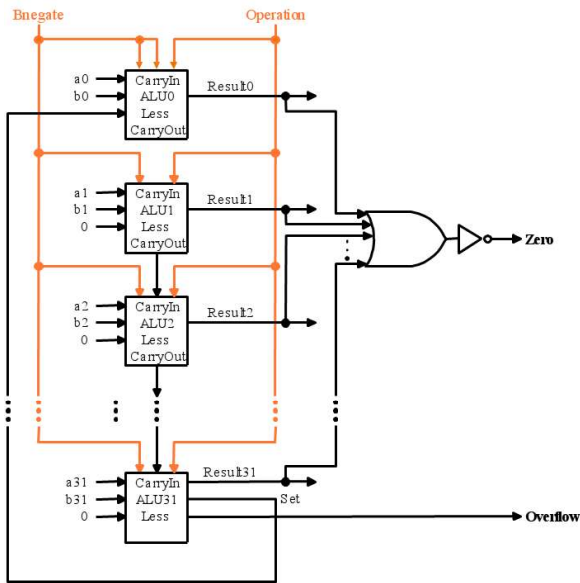


Figura 5. Unidad aritmético l3gica, implementaci3n serie

control. Y por 3ltimo, la cuarta versi3n incluir3 el soporte de excepciones. Los distintos bloques del dise1o se realizar3n de forma parametrizable respecto al tama1o de registros o se1ales que utilicen, extendiendo esos par3metros hasta el top del dise1o.

El prototipado en la placa S3ESK usar3 como interfaz

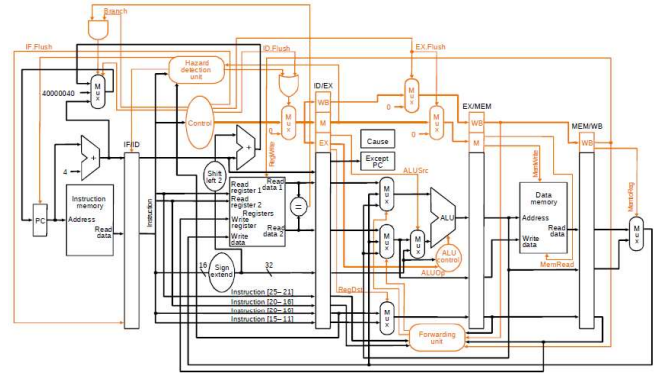


Figura 7. Esquema del dise1o final del procesador MIPS reducido

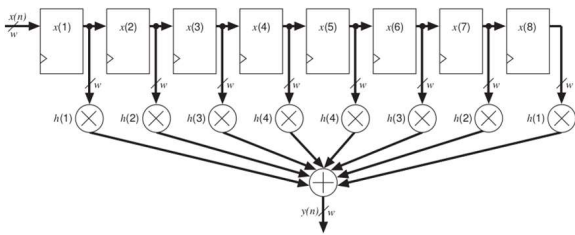


Figura 6. Filtro digital FIR, implementaci3n directa paralela

de entrada el codificador rotatorio que actuar3 selector de registros internos, cuyo valor y referencia del registro en curso se visualizar3 en los leds, y como generador de pulsos de reloj. Opcionalmente se utilizar3 como salida adicional el display LCD, haciendo uso de los m3dulos suministrados.

En lo referente al desarrollo software, se proporciona al alumno el c3digo m3quina correspondiente a un peque1o programa que permite testear las instrucciones en el procesador, y se deja abierta la posibilidad de implementar de forma opcional nuevas instrucciones del repertorio original, para lo cual se hace necesario el desarrollo de un nuevo programa a incluir en la memoria asociada al procesador.

microprocesador RISC MIPS R2000/R3000, con un repertorio de instrucciones muy reducido (add, sub, and, or, slt, lw, sw, beq), de acuerdo a las especificaciones conocidas seg3n [4] y de acuerdo al esquema de la 7. Se implementar3 un pipeline de 5 etapas (IF, ID, EX, MEM, WB) y la gesti3n de riesgos de datos y de control mediante las unidades de anticipaci3n y de detecci3n de riesgos, e incluir3 soporte para las excepciones de desbordamiento e instrucci3n no v3lida. Se seguir3 una metodolog3a de dise1o descendente (top-down) por etapas del pipeline, y una metodolog3a de dise1o ascendente (bottom-up) por versiones del dise1o, con incremento de la complejidad y la funcionalidad en cada nueva versi3n, documentando las distintas etapas de dise1o. En una primera versi3n, se dise1ar3 y verificar3 una implementaci3n monociclo estructurada en bloques reutilizables para las siguientes versiones; sobre esta versi3n se construir3 una segunda versi3n en la que se introducir3n los registros de segmentaci3n correspondientes entre los bloques funcionales y se dise1ar3 el nuevo control segmentado. En una tercera versi3n se introducir3 la unidad de anticipaci3n para solventar los riesgos de datos, y la unidad de detecci3n de riesgos para la detenci3n del pipeline en riesgos de datos y de

II-D. Pr3ctica 4 - Entorno de co-dise1o Xilinx EDK

En un nivel de abstracci3n y complejidad superior, la cuarta pr3ctica aborda el co-dise1o HW/SW de sistemas integrados en l3gica programable. Sobre la misma plataforma hardware y el kit de herramientas de desarrollo HW/SW de Xilinx EDK [5] (Figura 9, de forma tutorizada examinamos el proceso de creaci3n y verificaci3n de una plataforma hardware basada en el procesador soft Microblaze (Figura 8) [6].

En el tutorial se siguen una serie de pasos a partir de una descripci3n de los sistemas basados en MicroBlaze, comenzando por el uso de las herramientas b3sicas XPS (Xilinx Processor System) y el asistente BSB (Base System Builder) para incluir la infraestructura de buses y m3dulos IP (Intellectual Property) como perif3ricos del sistema, su conexi3n y direccionamiento. A continuaci3n se sigue el proceso de creaci3n de nuevos m3dulos como perif3ricos, co-procesadores o aceleradores hardware, de prop3sito espec3fico y su inclusi3n en el sistema, para la generaci3n del sistema completo dise1ado y su modelo de simulaci3n. La 3ltima parte corresponde a la definici3n del dise1o software con

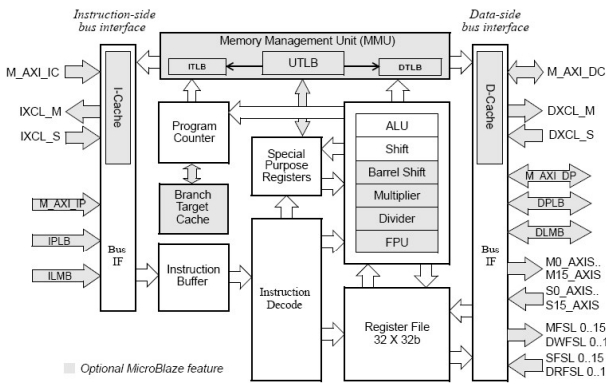


Figura 8. Estructura del procesador MicroBlaze

la herramienta XSDK (Xilinx Software Development Kit), comenzando por la generación de una capa de software BSP (Board Support Package) asociada al mapa de memoria del sistema, junto a los controladores software asociados a módulos y periféricos, que permitirá ejecutar el software de aplicación o sistemas operativos. El código máquina generado se combina con el hardware diseñado previamente, ocupando la memoria asociada al microprocesador, para descargar la configuración completa en la FPGA y su ejecución o depuración.

A partir del tutorial realizado se propone al alumno la creación

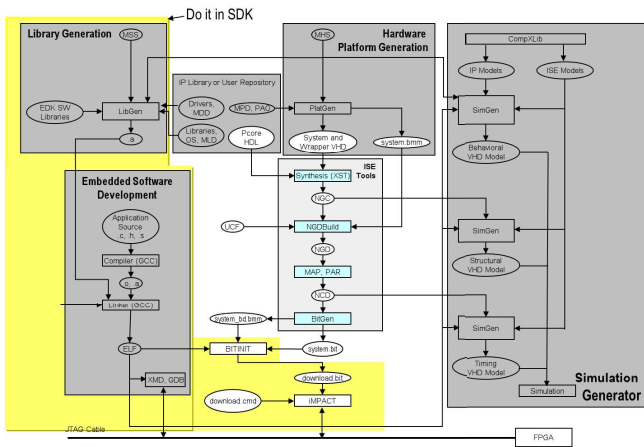


Figura 9. Flujo de diseño con Xilinx EDK

de un nuevo periférico y el desarrollo de una aplicación, que usando funciones de bajo nivel, permita la verificación del mismo y el resto de periféricos de tipo GPIO (Global Purpose Input Output).

II-E. Práctica 5 - Controladores gráficos y comunicaciones serie

La quinta práctica comprende tanto el diseño digital de sistemas con salida gráfica, a partir de controladores de periféricos de gráficos/texto sencillos previamente diseñados, como el diseño digital de comunicaciones serie, partiendo del uso de

controladores de periféricos de entrada/salida serie sencillos previamente diseñados. En cualquiera de los casos se requiere el conexionado de dichos módulos o periféricos a sistemas basados en microprocesador y su emulación/prototipado con la plataforma hardware.

Partiendo de la implementación del microprocesador MIPS32 de la práctica 3 y un controlador de video VGA facilitado, se propone representar el valor interno de los registros de dicho microprocesador utilizando como interfaz de salida una pantalla conectada a la placa S3ESK, previamente se habrán realizado pruebas con ejemplos facilitados de salida de video simple en modo gráfico y textual, y dotar al mismo de comunicación serie, incluyendo un controlador RS232C con recepción y transmisión full-duplex a 9600bps, diseñado como periférico del mismo mapeado en la memoria del sistema computador, utilizando en la medida de lo posible la interfaz original de los registros de terminal y unas estructuras de memoria FIFO asociadas a los canales de recepción y transmisión, opcionalmente se integrarán también controladores PS2 de teclado/ratón.

En cuanto a la comunicación serie se recomienda realizar unas pruebas previas conectando la placa S3ESK al puerto serie (real/USB2serial) de un PC, en primer lugar de transmisión S3ESK a PC de valores en secuencia a partir de un contador controlado por un pulsador, y a continuación, una transmisión-recepción en bucle, enviando un dato desde el PC y devolviendo el mismo, procesado o no (eco), como respuesta al envío desde la placa S3ESK.

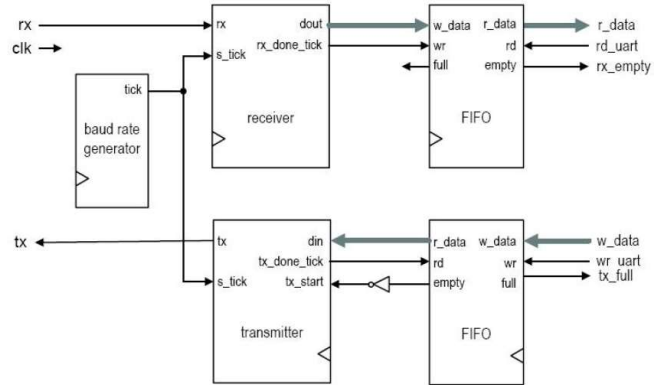


Figura 10. Diseño del controlador RS232C

II-F. Práctica 6 - Verificación HW/SW

La última sesión práctica en laboratorio versa sobre herramientas de instrumentación virtual para la depuración y verificación hardware de los sistemas digitales diseñados e implementados en FPGA, mediante la inserción de un circuito que actúa como analizador lógico en el mismo, y permite monitorizar en tiempo real las señales del sistema con la herramientas software ChipScope [7] e interfaz JTAG [8]. De forma tutorizada, y partiendo del diseño realizado en la práctica anterior, donde se diseña un controlador serie para

comunicar el procesador, se siguen los pasos necesarios para insertar la lógica de análisis, indicar las señales a monitorizar y sus condiciones de disparo, que en este caso corresponden a las líneas de recepción y transmisión serie, y de este modo verificar su correcto funcionamiento a nivel físico.

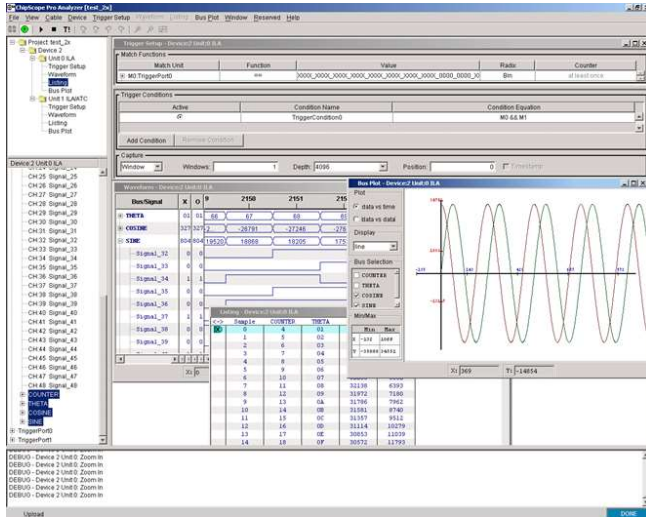


Figura 11. Análisis lógico en tiempo real con ChipScope

III. DESARROLLO DEL PROYECTO DE CODISEÑO HW/SW

A modo de proyecto final se propuso integrar las prácticas realizadas durante el curso en un único SoC con dos subsistemas basados en los dos microprocesadores utilizados MIPS y MicroBlaze, comunicados entre sí por un canal serie, donde las funcionalidades hardware y software que se deseasen incorporar en ambos subsistemas eran libres por parte del diseñador, si bien se deberá justificar el particionado HW/SW, documentar y verificar completamente el sistema de acuerdo a la metodología propuesta durante el curso.

Inicialmente se aporta al alumno información de objetivos generales y específicos, las especificaciones generales y una aproximación a la arquitectura del sistema y al plan de desarrollo que el alumno tendrá que concretar en fases posteriores, correspondientes a las etapas características de un proyecto de este tipo como diseño lógico, diseño físico y prototipado en FPGA, así como diseño para testabilidad.

En la primera fase, cada alumno debe proponer una solución preliminar a nivel arquitectural y un plan de desarrollo. Dicha solución se discutirá en grupo y se asignarán tareas dentro de cada fase a cada componente del grupo, participando todos en todas las fases.

Para el resto de las fases se establecen los puntos de control y fechas de exposición y discusión de resultados, previo informe relativo a las tareas desarrolladas por cada alumno.

Finalmente, se evaluarán en grupo los resultados finales, considerando el producto final creado, aplicando un razonamiento crítico sobre su introducción en el mercado, la mejora del producto y el desarrollo del mismo.

IV. EVALUACIÓN

La evaluación de la asignatura Diseño Digital se llevará de acuerdo a las fines expuestos en la metodología, y de forma más específica mediante la realización de un examen escrito de teoría, la valoración del proyecto práctico y la valoración personal del profesor sobre el alumno, en base al interés mostrado por el alumno en las clases teóricas y prácticas de laboratorio. Téngase en cuenta que desde principio de curso el alumno conoce sus responsabilidades acerca de la asistencia obligada, salvo justificación, a las clases prácticas, la asistencia sino obligada si indicada a las clases teóricas, y los criterios de evaluación y calificación aquí propuestos.

Ponderando los distintos factores se asigna un peso porcentual de un 30 % al examen de teoría y el 60 % restante al proyecto práctico, el resto de aspectos planteados se consideran como complemento sobre el total.

IV-A. Criterios

Los *criterios* aplicados y su correspondiente *ponderación* son los siguientes:

- Demostrar la adquisición y comprensión de los conocimientos de la asignatura mediante un examen teórico. (30%)
- Realizar las prácticas y proyectos a partir de los conocimientos teóricos y resultados experimentales, elaborando la documentación necesaria de acuerdo a una metodología propuesta. (50%)
- Exponer con claridad los resultados y conclusiones sobre el proyecto realizado. (10%)
- Participación en clase y en el grupo de trabajo. (10%)

IV-B. Instrumentos

Los *instrumentos de evaluación* derivados de estos criterios serán:

- Seminarios y tutorías ECTS, que permiten la valoración continua de las actividades registradas en prácticas y la dedicación en el desarrollo de las mismas.
- Exposición pública de trabajos y resultados del proyecto con auto-evaluación y evaluación del resto del resto de compañeros participantes en el grupo y asistentes a la exposición.
- Evaluación de problemas propuestos en clase.
- Examen final, donde se evalúan los conceptos fundamentales de la asignatura y se plantea un diseño que el alumno debe resolver desde un punto de vista conceptual.
- El control de asistencia a las prácticas obligatorias y puntos de control del proyecto.

V. APOYO DE E-LEARNING

El uso de las tecnologías de la información facilita en nuestro caso el aprendizaje mediante la creación, adopción y distribución de contenidos, así para el aumento de disponibilidad en tiempo de las herramientas necesarias para el desarrollo de las actividades. También se aplica para el intercambio de opiniones y resolución cooperativa de problemas que surgen

a lo largo del proyecto, así como para la entrega de material elaborado por los alumnos en los plazos fijados.

V-A. Instrumentos

- Planificación en secuencia semanal de actividades y distribución del material asociado (documentos y enlaces) mediante el aula virtual.
- Laboratorio virtual para la enseñanza de los sistemas digitales (dglab) [9] [10], desarrollado por los autores del trabajo, que incluye todas las herramientas necesarias para el codiseño en una máquina virtual.
- Servidor remoto de aplicaciones con disponibilidad continúa para realizar tareas prácticas fuera del aula a través de una conexión a internet.
- Comunicación síncrona o asíncrona entre alumnos, o entre alumno y profesor, mediante foros y chat del aula virtual, y el correo electrónico.
- Tareas de entrega de archivos, correspondientes a trabajos y proyecto, con fechas límites y especificaciones de formatos, planificadas en el calendario del aula virtual. El seguimiento y evaluación del proyecto se lleva a cabo en gran medida con ayuda de esta herramienta, tanto en lo relativo al tiempo dedicado a cada una de las tareas, como a la evaluación de resultados por parte del profesor y su posterior calificación.

VI. CONCLUSIÓN

La aplicación del método de aprendizaje basado en proyectos, en este caso, supone una gran ventaja para el alumno que se enfrenta a la implementación de un sistema digital complejo real a partir de un diseño jerárquico y modular, con tutorización en las primeras etapas de diseño y seguimiento del desarrollo y prototipado en las últimas.

Los problemas asociados se relacionan con la elevada carga no presencial y organización de tiempos por parte del alumno.

REFERENCIAS

- [1] *Xilinx website*, Xilinx Inc, 2016. dirección: <http://www.xilinx.com>.
- [2] *Spartan-3E Starter Board*, Digilent Inc., 2016. dirección: <http://store.digilentinc.com/spartan-3e-starter-board-limited-time/>.
- [3] *Xilinx University Program*, Xilinx Inc., 2016. dirección: <http://www.xilinx.com/support/university.html>.
- [4] D. A. Patterson y J. L. Hennessy, *Estructura y Diseño de Computadores: la interfaz Hardware / Software*, 4ª ed. Reverté, 2011.
- [5] *Embedded Development Kit (EDK)*, Xilinx Inc, 2016. dirección: http://www.xilinx.com/products/intellectual-property/chipscope_ila.html.
- [6] *Logicore IP MicroBlaze Micro Controller System, Product Specification, DS865*, Xilinx Inc, 2012. dirección: http://www.xilinx.com/support/documentation/sw_manuals/xilinx14_1/ds865_microblaze_mcs.pdf.
- [7] *Chipscope Integrated Logic Analyzer (ILA)*, Xilinx Inc, 2016. dirección: http://www.xilinx.com/products/intellectual-property/chipscope_ila.html.
- [8] IEEE Computer Society, *IEEE Standard 1149.1-1990 for Test Access Port and Boundary-Scan Architecture*, 2013. DOI: 10.1109/IEEESTD.2013.6515989. dirección: <http://standards.ieee.org/findstds/standard/1149.1-2013.html>.
- [9] *Laboratorio virtual para la enseñanza de sistemas digitales (dglab) desarrollado por los autores del trabajo*, 2015. dirección: <http://digital.unex.es/files>.
- [10] *Documentación para el laboratorio virtual para la enseñanza de sistemas digitales (dglab) desarrollado por los autores del trabajo*, 2015. dirección: <http://digital.unex.es/dglab>.
- [11] —, *IEEE Standard 1364-2005 for Verilog Hardware Description Language*, 2006. DOI: 10.1109/IEEESTD.2006.99495. dirección: <http://standards.ieee.org/findstds/standard/1364-2005.html>.
- [12] D. A. Patterson y J. L. Hennessy, *Computer Architecture and Design: The Hardware / Software Interface*, 4th ed. Morgan Kaufmann, 2012.
- [13] J. F. Wakerly y E. A. Miguel, *Diseño Digital: principios y prácticas*. Pearson Educación, 2001.
- [14] *Digilent website*, Digilent Inc., 2016. dirección: <http://www.digilentinc.com>.