

UN LABORATORIO DE DISEÑO DIGITAL EN VHDL: APRENDIZAJE POR PROYECTOS

A. ROSADO, M. BATALLER, J. GUERRERO, J. MUÑOZ, J. VILA
*Departamento de Ingeniería Electrónica. Escuela Técnica Superior de Ingeniería.
Universidad de Valencia. 46100 Burjassot. Valencia. España.*

El aprendizaje del lenguaje VHDL en la Ingeniería Electrónica necesita de una constante realización práctica de ejercicios, tanto de simulación como de implementación. Por ello, este trabajo describe la metodología llevada a cabo para conseguir que el aprendizaje de este lenguaje de descripción hardware sea efectivo y rápido, especialmente en lo referido al VHDL orientado a síntesis hardware. Se han empleado dispositivos FPGA de Xilinx y el simulador ModelSim para la realización del laboratorio, desarrollando una placa específica de periféricos para ampliar las posibilidades de realización de proyectos por parte del estudiante.

1. Introducción.

En muchas ocasiones no resulta fácil la enseñanza de un lenguaje de descripción hardware como VHDL. Esto es debido generalmente a la concepción previa que el alumno suele adquirir a través de asignaturas de programación software en cursos precedentes donde los lenguajes aprendidos utilizan una filosofía de concepción diferente y esencialmente secuencial. Todo lo contrario ocurre en VHDL, donde la modularidad, ejecución concurrente y relación directa con el hardware son unos factores muy importantes a la hora de decidir cómo se realiza el diseño. Relacionado con este último aspecto (la implementación hardware), también influye decisivamente cómo se realice la descripción VHDL para que el sistema pueda ser implementado en hardware, o quede más o menos optimizado en cuanto a recursos y velocidad de operación del circuito resultante.

VHDL es un lenguaje no necesariamente conectado con síntesis hardware, pero viene asociado a la implementación de sistemas digitales, por lo que trataremos únicamente todo lo relacionado con el aprendizaje VHDL orientado a síntesis. Por ello, resulta imprescindible que el estudiante conozca las diferencias de estilo de una descripción para simulación y una descripción donde se debe considerar que el sistema a obtener se implementará en hardware (en nuestro caso, FPGA ó CPLD).

Existe una gran cantidad de información referente al aprendizaje del lenguaje en VHDL, tanto referencias bibliográficas como recursos online en Internet [1]. En cambio, en muchas ocasiones, la descripción del lenguaje se realiza basándose en la descripción de las sentencias, unidades de concepción, etc, incluyendo ejemplos cortos de cada posible tipo de descripción, sin incidir siquiera en la posibilidad de ser sintetizable o no. Por otro lado, las referencias bibliográficas que inciden en las descripciones VHDL para síntesis [2] aportan ejemplos que permiten comprender de mejor forma cómo se realizan descripciones apropiadas, pero en la mayoría de los casos sin tener en cuenta cómo esa descripción se convertirá en un hardware determinado. De este modo, de forma complementaria a las soluciones de aprendizaje descritas, se propone que el alumno se enfrente a los detalles de programación VHDL por sí mismo y de este modo vaya adquiriendo la destreza necesaria para desarrollar código de forma eficiente, sencilla y rápida, evaluando cómo se modifican los resultados dependiendo del tipo de descripción que se realice, y aborde de la forma más directa con la introducción de código en VHDL.

La propuesta para el desarrollo del laboratorio de VHDL se construye a través del concepto de “learn by doing” (aprendizaje por proyectos), es decir, una vez adquiridos los conceptos

generales y realizadas las primeras prácticas de familiarización con el entorno de programación, orientar el resto de sesiones al desarrollo de proyectos completos donde el objetivo final es la programación y prueba de un circuito hardware con una funcionalidad determinada.

2. Objetivos

Se pretende que el estudiante aprenda VHDL sin que éste posea conocimientos previos en lenguajes de descripción hardware, y que conozca todos los recursos del lenguaje, tanto para su empleo en simulación como en síntesis de circuitos digitales. De manera adicional, el desarrollo de este laboratorio intenta fomentar las habilidades de trabajo en grupos reducidos, manejo de información en inglés y aprendizaje autónomo, además de mejorar la expresión oral y escrita, que a menudo resultan factores que el alumno debe potenciar.

Tratándose de una asignatura orientada a adquirir el conocimiento de un lenguaje de programación, no resulta sencillo alcanzar todos los objetivos que se persiguen, ya que la impartición teórica de los fundamentos del lenguaje puede resultar densa y poco aprovechada por el alumno si ésta se plantea de un modo puramente enciclopédico y descriptivo. Por tanto, se ha optado por desarrollar una dinámica de trabajo que consiga que el alumno se vea más implicado en el aprendizaje del lenguaje y pueda aprender sus recursos de programación conforme resuelve problemas y realiza ejercicios prácticos.

Por tanto, el objetivo global que se persigue es el de proponer metodologías que faciliten el aprendizaje de un lenguaje de programación nuevo para el alumno, utilizando para ello el mayor número de ejemplos prácticos y resolución de problemas en la parte de clase teórica, complementado con el desarrollo de prácticas de laboratorio capaces de hacer que el alumno descubra toda la potencialidad del lenguaje mediante la propuesta de proyectos completos y de aplicación directa, donde no sólo deben poner a prueba sus habilidades de programación en VHDL sino en la concepción y diseño de un sistema digital, ya sea a nivel funcional, sincronismo de tiempos, retardos, etc.

3. Hardware empleado.

En este caso, en el laboratorio se emplea como elemento base una placa FPGA comercial Digilent (Fig. 1) con una FPGA Xilinx Spartan 3 XC3s200 y algunos periféricos integrados [3]. Esta placa, a pesar de su bajo coste, posee LED's (8), Display de 7 segmentos (4), puerto PS2, puerto serie RS232, 2 módulos de 256x16 de SRAM (1MByte), 2 Mbit de PROM para almacenamiento permanente de la programación de la FPGA, 4 pulsadores, 8 microinterruptores, reloj de 50MHz y 3 conectores de expansión. Todo esto permite la realización de varias prácticas de laboratorio sin necesidad de elementos periféricos adicionales.

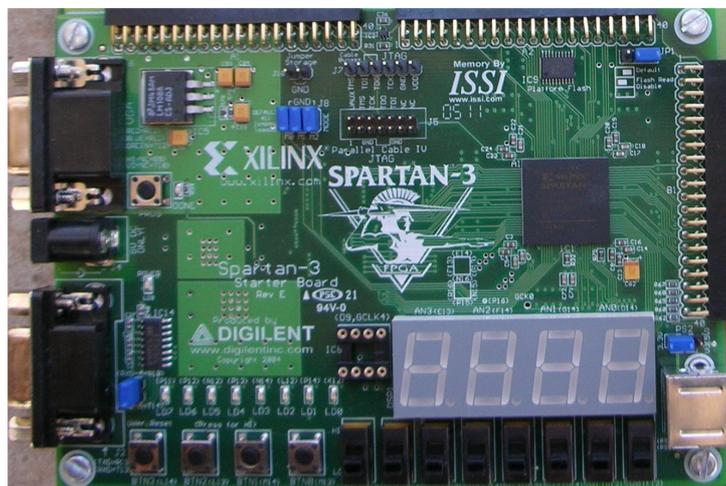


Figura 1. Placa comercial Spartan3 DIGILENT.

Aprovechando los conectores de expansión de la placa, para ampliar la funcionalidad de la misma, se ha desarrollado una placa de periféricos que dispone de:

1. Cuatro display de 7 segmentos controlados a través de un mismo bus BCD y habilitación individual que incluye el decodificador BCD a 7 segmentos 74LS47, con lo que se requieren 8 líneas para su control (4 líneas para el código BCD y 4 líneas de habilitación).
2. Un convertor analógico/digital serie de 12 bits y dos canales National Semiconductor ADC122S021 [4], 200ksps donde únicamente son necesarias 4 líneas de conexión con la FPGA.
3. Dos convertores digital/analógico Burr-Brown DAC7513 de 12 bits [5], cada uno con 3 líneas de conexión con la FPGA.
4. Un teclado hexadecimal pasivo con 4 líneas de entrada y 4 de salida para realizar lectura de tecla.
5. Una memoria I²C EEPROM serie de 8 bits Microchip 24LC08B [6] que precisa de dos líneas de comunicación para lectura/escritura.
6. Un display LCD 16x2 con controlador estándar HD44780[7].
7. Un reloj de tiempo real no volátil Dallas DS12C887 [8].
8. Un dispositivo de comunicación inalámbrica.

Respecto de los convertores D/A y A/D, es posible conectarlos a un conector de clavija de 3,5mm para desarrollar aplicaciones de audio (se permite el control de dos canales para permitir procesado estéreo de la señal), siendo posible conectar cualquier dispositivo de entrada (reproductor MP3, ordenador PC, walkman, etc.) y salida (auriculares, altavoces autoamplificados, etc.), aunque es posible elegir si la entrada/salida proviene de la clavija, o bien de un conector de tornillo o un potenciómetro incorporado en la placa (el potenciómetro permite probar si se leen valores correctos de ADC sin necesidad de conectar señal externa). Como regla general, se incluyen puntas de prueba sobre las señales en todos los componentes, para poder depurar con analizador lógico u osciloscopio las señales de comunicación entre la FPGA y cada uno de los periféricos.

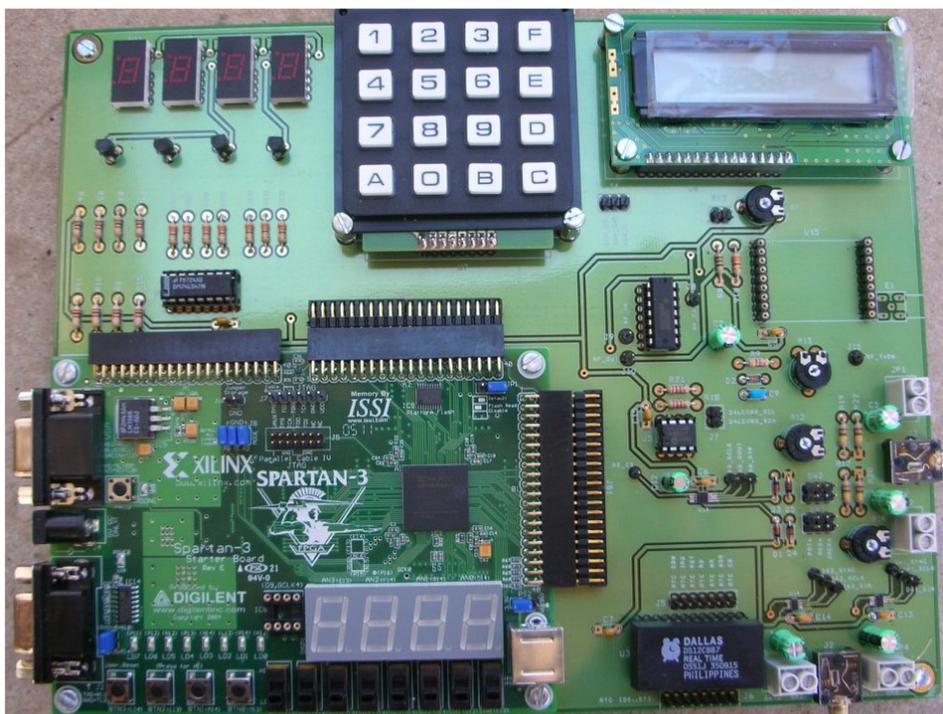


Figura 2. Placa de periféricos con placa Spartan3 incorporada.

4. Descripción de la propuesta

El módulo Diseño de Circuitos y Sistemas Electrónicos es una asignatura troncal de primer curso de la titulación de segundo ciclo de Ingeniería Electrónica, que tiene una carga lectiva de 12 créditos (7,5 teoría + 4,5 laboratorio), en la que se pretende proporcionar al alumno una formación adecuada en el diseño de circuitos integrados y sistemas electrónicos, no sólo sobre diseño digital basado en FPGA sino el diseño electrónica desde diferentes puntos de vista (nivel de silicio, ASIC, lógica programable, diseño analógico, etc.). Una parte importante del temario se dedica al diseño de sistemas digitales (80% del total) y el lenguaje VHDL (30% del tiempo en la teoría y 75% en el laboratorio).

Respecto al laboratorio, de un total de 10 sesiones de 4 horas (quincenalmente repartidas de Noviembre a Mayo), se proponen 2 sesiones dedicadas al diseño microelectrónico con el editor de layout a Microwind2 [9] y 8 sesiones al diseño con FPGAs, en su totalidad centradas en VHDL, aunque en algunas ocasiones se emplea de forma complementaria el diseño esquemático.

Centrándose en las 8 sesiones de VHDL, 4 de ellas son comunes a todos para alcanzar un nivel de conocimientos suficiente como para afrontar el trabajo autónomo. En estas primeras cuatro sesiones, las prácticas a realizar se plantean de modo tradicional, es decir, se propone un guión de trabajo, se plantean diversas cuestiones que el alumno debe resolver, y en el plazo de una semana el alumno debe entregar una memoria del trabajo realizado junto con los ficheros de diseño realizados (incluyendo bancos de prueba). Las prácticas que se proponen son [10]:

1. Introducción al diseño en VHDL con Xilinx ISE. Se realiza un pequeño diseño de un multiplicador de 1 bit, para posteriormente construir jerárquicamente un multiplicador de 3 bits. El propósito principal es que el estudiante se familiarice con el entorno software.
2. Diseño de máquinas de estados en VHDL. Como ejercicio, se debe gestionar el control de acceso a un laboratorio a través de determinadas claves.
3. Implementación de un filtro de media móvil de orden 4. Mediante este ejercicio, se pretende que el estudiante refuerce los conocimientos en creación de bancos de prueba, especialmente en lo que se refiere a acceso lectura-escritura de ficheros con datos para la simulación. Por otro lado, también se hace hincapié en los resultados de la implementación sobre FPGA para establecer las posibles prestaciones del diseño obtenido.
4. Cronómetro digital que muestre décimas de segundo, segundos, y minutos, basado en el reloj de 50 MHz, mostrando las décimas de segundo en los LED, y los segundos y minutos en los display de 7 segmentos que vienen integrados en la placa Spartan3. En este caso, como última práctica común, se realiza una sesión que incorpora todo los elementos que deberán tener en cuenta cuando realicen el proyecto, permitiendo de este modo una mayor familiarización con la placa y los procedimientos a seguir.

Las 4 sesiones restantes son las dedicadas a proyectos por grupos en los que cada grupo desarrolla un diseño diferente, y cada proyecto ocupa 4 sesiones de laboratorio. Se establecen grupos de trabajo dentro de la clase (aproximadamente 8 grupos de 2 ó 3 personas cada uno), de modo que durante la duración de las 4 sesiones deben ser capaces de desarrollar la aplicación propuesta partiendo de unas especificaciones y las hojas de datos del dispositivo o dispositivos que deben controlar. Así, en la última sesión se realiza la presentación por grupos de los resultados obtenidos junto con una memoria escrita del proyecto realizado. Durante la realización del proyecto, el trabajo del profesor es el de ayudar a avanzar en el desarrollo del mismo, y evaluar la actitud de cada uno de los grupos frente al problema planteado. Los proyectos planteados son:

1. Escritura y lectura en memoria SRAM IS61LV25616AL-10T [11], introduciendo valores (8 bit) e instrucciones (leer o escribir) a través de los microinterruptores, y leyendo datos para visualizarlos en los LED.
2. Visualización de imagen en VGA. Partiendo de un código base donde se les indica cómo se realizan los barridos de pantalla, mostrar diferentes figuras en la pantalla (cuadrado, círculo, etc.) e intentar mostrar algún objeto en movimiento.
3. Visualización de datos en el display LCD 16x2. En este caso, se pide complementar la práctica del control de acceso propuesta en las sesiones comunes, añadiendo visualización de información a través del display LCD e introduciendo datos mediante el teclado hexadecimal.
4. Control de un reloj de tiempo real Dallas DS12C887, donde se debe controlar una funcionalidad reducida del mismo, en este caso se propone poder modificar y leer año, mes, día y hora. Todo ello mediante un control de órdenes a través de los microinterruptores, y visualización de datos a través de los display de 7 segmentos. Es necesario considerar el tipo de salida hacia los display de 7 segmentos dependiendo de si pertenecen a la placa Spartan3 o placa de periféricos, ya que en cada caso el control de los mismos es diferente.
5. Lectura y escritura de datos de memoria EEPROM serie con interfaz I²C. A través de los pulsadores, microinterruptores y visualizadores, se gestiona la escritura/lectura de datos en EEPROM, comprobando que, efectivamente, tras desconectar la alimentación, los datos se pueden volver a leer.
6. Filtrado de señal de audio. Se debe realizar la conversión A/D, leyendo vía serie los valores provenientes de los dos canales estéreo de un reproductor MP3 portátil, para posteriormente realizar diversos tipos de filtrado de señal dependiendo de los valores de los microinterruptores y sacar los dos canales de salida hacia dos convertidores D/A y así poder escuchar mediante altavoces autoamplificados el efecto de los diferentes filtros. Inicialmente se proponen algoritmos de filtrado básicos similar al filtro de media móvil.
7. Envío y recepción de mensajes a través del puerto serie. Se debe poder realizar un diseño capaz de comunicarse con un PC a través del puerto serie. Como prueba sencilla, desde la FPGA se debe ser capaz de enviar caracteres que “hyperterminal” de Windows visualice, y a la inversa, es decir, que desde “hyperterminal” se escriba un carácter y el código ASCII de éste sea mostrado por los LED de la placa Spartan3.
8. Control de un teclado PS2. Empleando el conector PS2 de la placa Spartan3, realizar el protocolo de comunicación para que la FPGA reconozca el código ASCII de la tecla pulsada en el teclado, mostrando su valor en los LED.

Con estos proyectos, los alumnos pueden trabajar sobre el diseño de interconexión de sistemas electrónicos, teniendo la necesidad de considerar todos los factores involucrados en el diseño real (excepto trazado de PCB), con lo que además de emplear VHDL como herramienta de diseño, deben desarrollar un código que además de ser sintetizable debe cumplir todos los requerimientos exigidos en el diseño (temporizaciones, control de flujo de datos, etc.).

Adicionalmente, partiendo de los proyectos planteados, la interrelación entre ellos puede ser completa, es decir, se pueden combinar los proyectos para, por ejemplo, almacenar pulsaciones de tecla en memoria SRAM o EEPROM y visualizarlas en LCD. Así, es posible construir un sistema más complejo que el constituido por un único proyecto, lo que puede permitir a los estudiantes interactuar entre diferentes grupos y proponer incluso proyectos combinados que pueden resultar muy interesantes.

5. Análisis de resultados

La estructuración del laboratorio según la propuesta descrita ha permitido que tras un breve periodo de aprendizaje guiado, los estudiantes puedan desarrollar un trabajo autónomo. La ventaja de disponer de cuatro sesiones de laboratorio para el desarrollo del proyecto permite que el estudiante pueda conocer la marcha de su trabajo en puntos intermedios sin necesidad de esperar al final, pudiendo probar sobre la placa hardware las diferentes etapas en las que debe

estructurar el desarrollo del proyecto. Para ello, en cada sesión intermedia el profesor establece un tiempo para revisión y evaluación del trabajo de cada grupo. Este tipo de proyectos requiere que los estudiantes trabajen fuera del laboratorio, realizando el trabajo de modo que el tiempo del laboratorio se dedique lo más posible a la prueba de las diferentes posibilidades de diseño que los estudiantes hayan planteado previamente, habiendo realizado las correspondientes simulaciones, etc.

Como ya se dijo anteriormente, la última sesión de laboratorio se emplea para la exposición del trabajo realizado. Por tanto, en esta sesión se debe presentar una memoria escrita del proyecto asignado, realizar la exposición oral del trabajo a lo largo de 20 minutos como máximo, y mostrar funcionalmente en placa el diseño implementado.

Tras la experiencia desarrollada hasta ahora, se aprecia que los estudiantes aprenden realmente la metodología de programación en VHDL para síntesis, ya que son capaces de ver en qué casos unas descripciones de código que pensaban que podrían funcionar correctamente (porque en simulación lo hacían), realmente no funcionan sobre la FPGA, debiendo además tener en cuenta que es necesario realizar diversos procesos de forma concurrente, por lo que deben saber explotar esta característica del lenguaje. Por otro lado, el desarrollo del proyecto les obliga a realizar un trabajo en grupo organizado, ya que generalmente todos los proyectos propuestos están pensados para que se puedan dividir en subtareas a repartir entre los miembros del grupo de trabajo. Obviamente, siempre aparecen casos donde unos miembros trabajan más que otros, cosa que el profesor debe ser capaz de detectar a lo largo de las sesiones de seguimiento del proyecto.

6. Conclusiones

De forma global se puede concluir que los alumnos, a través de la resolución de los problemas que surgen durante cada una de las fases del proyecto, obtienen una visión global del proceso de diseño de un sistema digital desde las especificaciones hasta la fase final, en la que además de un diseño correcto se deben cumplir requerimientos de tiempo y sincronismo entre dispositivos que el estudiante debe resolver. Los resultados obtenidos indican el buen nivel de conocimientos adquiridos por el alumno en el diseño de sistemas digitales basados en descripciones VHDL, así como el grado de satisfacción alcanzado al completar la fase de diseño con su implementación hardware.

Esta metodología pretende por otro lado iniciar una aproximación a la docencia acorde con el Espacio Europeo de Educación Superior (EEES), para que una vez implantados los nuevos títulos de grado y posgrado, se pueda planificar la docencia habiendo tenido experiencias previas.

Como posibilidad de ampliación futura, se prevé una propuesta de trabajo interdisciplinar, de modo que estos proyectos se puedan proponer de forma coordinada con otras asignaturas. Por ello, en la asignatura de “Filtros Digitales” se propone el diseño de filtros que posteriormente se puedan implementar en la placa FPGA, también existe un espacio en la placa para introducir una placa de comunicación inalámbrica y realizar trabajos relacionados con el “Tratamiento y Transmisión de Señales”, y por otro lado, la placa Spartan3 admite la programación del microprocesador embebido Microblaze, lo que supondría el empleo de esta placa para el desarrollo de proyectos y ejercicios utilizando los periféricos con programación en C para Microblaze.

7. Agradecimientos

Queremos expresar nuestro agradecimiento al Ministerio de Educación y Ciencia por su apoyo parcial a la presentación de este trabajo a través de la red temática del Capítulo Español de la Sociedad de la Educación del IEEE (TSI2005-24068-E).

Referencias

- [1] R. Lipsett, C.F. Schaefer, C. Ussery. *VHDL: Hardware Description and Design*. Springer. USA. 1989.
- [2] J. Bhasker. *A VHDL Synthesis Primer*. Second Edition. Star Galaxy Pub., USA. 1998.
- [3] Xilinx. *Spartan 3 Starter Kit Board User Guide*. USA. 2005.
- [4] National Semiconductor Corp. *ADC122S021 Data Sheet*. USA. 2005.
- [5] Texas Instruments. *Burr-Brown DAC7513 Data Sheet*. USA. 2003.
- [6] Microchip Technology Inc. *24AA08/24LC08B Data Sheet*. USA. 2002.
- [7] Hitachi. *HD44780U LCD Controller*. Japan. 1999.
- [8] Dallas Semiconductor. *Real Time Clock DS12C887 Data Sheet*. USA. 2000.
- [9] Etienne Sicard. *Microwind 2 Software Manual*. INSA Toulouse. France. 2003.
- [10] A. Rosado, M. Bataller. *Página Web de la asignatura DCSE*. Actualizada hasta curso 04-05, posteriormente se introduce Aula Virtual. <http://www.uv.es/rosado/dcse/dcse.html>
- [11] Integrated Silicon Solutions Inc. *SRAM IS61LV25616AL Data Sheet*. USA. 2005.