

Medición de distancias como ejemplo práctico de diseño en FPGAs

Carlos Jiménez Fernández, Pilar Parra Fernández, Carmen Baena
Oliva, Manuel Valencia Barrero
Dpto. Tecnología Electrónica
Universidad de Sevilla
Instituto Microelectrónica de Sevilla
IMSE-CNM CSIC/US
Sevilla, España
[cjesus.parra.baena.manolov}@imse-cnm.csic.es](mailto:{cjesus.parra.baena.manolov}@imse-cnm.csic.es)

F. Eugenio Potestad Ordóñez
Instituto Microelectrónica de Sevilla
IMSE-CNM CSIC/US
Sevilla, España
potestad@imse-cnm.csic.es

Abstract— El aprendizaje de diseño digital a nivel RT requiere de ejemplos prácticos y conforme se avanza en el aprendizaje se precisa que los ejemplos aumenten de complejidad. Las FPGA y las placas de desarrollo ofrecen una plataforma muy adecuada para la implementación de estos diseños. Sin embargo las sesiones de prácticas presenciales suelen tener una duración de dos horas, lo que no facilita que la complejidad de los diseños sea alta. Por ello se requiere diseños que se puedan realizar en varias sesiones y que además sean prácticos. En esta comunicación se presenta, a modo de demostrador, la construcción de un sistema medidor de distancia. Para ello se cuenta con un módulo de medición de distancias con ultrasonidos y los datos se muestran en visualizadores 7-segmentos de una placa Nexys4.

Keywords— VHDL, diseño de sistemas digitales, medidor de distancia, FPGA.

I. INTRODUCCIÓN

Las asignaturas de diseño digital en cursos avanzados de titulaciones electrónicas pueden utilizar lenguajes de descripción de hardware para la descripción de circuitos y dispositivos FPGA como plataforma para probar de forma experimental el comportamiento de los circuitos diseñados. Esta metodología tiene muchas ventajas. Una de ellas es la utilización de un único entorno de CAD para el diseño, la verificación y la programación de los dispositivos. Además este software es ofrecido de forma gratuita para uso docente por los fabricantes de FPGA. Otra ventaja es la disponibilidad de placas de desarrollo, que incluyen además de la FPGA elementos de visualización e interfaz que permiten introducir valores a las entradas y ver resultados en las salidas. Por todo esto la enseñanza de diseño digital con el tándem VHDL-FPGA es una alternativa con un coste aceptable y, sobre todo, muy práctica con un alto atractivo para los alumnos.

Esta alternativa se está aplicando en la asignatura “Diseño Digital Avanzado” [1], optativa de cuarto curso de la titulación del Grado en Electrónica Industrial impartida en la Escuela Politécnica Superior de la Universidad de Sevilla. En esta asignatura se pretende que el alumno aprenda los conceptos más importantes del diseño digital. El único apoyo que tiene esta asignatura son las asignaturas “Electrónica Industrial” y “Electrónica Digital”, ambas de segundo curso.

La asignatura “Electrónica Industrial” es una asignatura del bloque de formación común de la rama industrial, impartida en el primer cuatrimestre del segundo curso. Es la primera asignatura que tienen los alumnos en esta titulación relacionada con la electrónica. Los contenidos de esta asignatura [2] constan, básicamente, de un bloque analógico y un bloque digital. En el bloque digital se introducen los conceptos básicos de la electrónica digital, desde el álgebra de conmutación hasta el diseño de máquinas de estado, pasando por los conceptos de puertas lógicas y biestables.

La asignatura “Electrónica Digital” es una asignatura obligatoria en el plan de estudios de la titulación. Se imparte en el segundo cuatrimestre del segundo curso. Sus contenidos [3] desarrollan los iniciados en Electrónica Industrial. Incluyen características reales de las puertas lógicas y los biestables, análisis y diseño de circuitos digitales (combinacionales y secuenciales) y subsistemas combinacionales y secuenciales. En los últimos temas de la asignatura se introducen conceptos relacionados con el diseño a nivel RT (estructura de circuitos basadas en unidad de control y unidad de datos), cartas ASM y principios básicos de microprocesadores.

Con estos conocimientos previos por parte de los alumnos, en la asignatura “Diseño Digital Avanzado” se enseña a los alumnos la descripción de circuitos digitales usando el lenguaje de descripción de hardware VHDL y la forma de implementarlos sobre dispositivos FPGA (en nuestro caso de Xilinx). La asignatura se plantea de forma muy práctica, de manera que los alumnos van realizando laboratorios en los que van haciendo el diseño de pequeños circuitos. Conforme avanza la asignatura estos laboratorios van subiendo un poco de complejidad, pero en el tiempo destinado a los laboratorios (un máximo de dos horas) no da tiempo a realizar diseños que tengan una mínima complejidad.

Una solución para este problema es la realización de un diseño en varias sesiones. Pero para que esta solución sea interesante debe tener varias características: por una parte cada una de las partes debe ser autocontenida: debe tener un objetivo de diseño que se pueda conseguir en una sesión de laboratorio. También debe suponer una construcción paulatina de la funcionalidad a realizar.

En este contexto, en esta comunicación presentamos a modo de demostrador un conjunto de cuatro prácticas que

tienen como objetivo la construcción de un medidor de distancia utilizando como base un dispositivo medidor de distancias por ultrasonidos y los visualizadores 7-segmentos como elemento para mostrar el resultado.

La estructura de esta comunicación es como sigue: en el segundo apartado se explica brevemente el diseño a realizar. En el apartado III se detallan los contenidos y los objetivos de cada una de las sesiones en las que se ha dividido el diseño así como los resultados que se pretenden obtener. Finalmente se extraen algunas conclusiones.

II. DISEÑO A REALIZAR

El objetivo del conjunto de prácticas que se propone es diseñar un medidor de distancia basado en un dispositivo comercial. Se trata del HC-SR04 [4] (“Fig. 1”) que mediante ultrasonidos es capaz de medir distancias dentro de un rango determinado y presentar las medidas realizadas en visualizadores 7-segmentos.

El sistema ha de incorporar dos modos de funcionamiento: modo continuo y modo unitario. En el modo continuo, el sistema activará el medidor de forma permanente y en todo momento se mostrarán las medidas que se estén realizando. Sin embargo, en modo unitario, el sistema realizará una única medida cada vez que se dé la orden correspondiente.

La realización del sistema completo implica el desarrollo de diversos elementos adicionales como convertidores de códigos y una unidad de control que formarán parte del que será el módulo principal del sistema.

En el diagrama de bloques del sistema medidor, véase “Fig. 2”, se muestra la conexión entre el módulo HC-SR04, el módulo principal y los visualizadores 7-segmentos. Respecto a las entradas externas del sistema, además de las señales de *reset* y reloj *ck*, se muestran la señal *modo* que permite elegir entre los dos modos de funcionamiento (unitario y continuo) y la señal *mide* que activa la operación de medida en el modo unitario. En cuanto a las salidas, *data_valid*, informa de que el dato mostrado en los visualizadores ya contiene la distancia medida y *alarma* es una señal que se activa cuando la distancia al obstáculo se encuentre dentro de un rango predefinido.

El módulo HS-SR04 interactúa con el módulo principal a través de dos señales, una de entrada al medidor (*trigger*) y otra de salida (*echo*). Tras recibir un pulso en la entrada *trigger* el medidor emite una señal de ultrasonidos y espera a recibir la señal retornada tras incidir en el obstáculo. Durante este tiempo



Fig. 1 Medidor de distancia HC-SR04.

de espera el medidor genera un pulso positivo por la salida *echo* que será proporcional a la distancia a medir (“Fig. 3”). Esta distancia está contenida en un rango entre 2cm y 4m.

El módulo principal está compuesto de una unidad de control, *control_maxsonar*, y un convertidor de código binario a 7-segmentos. La unidad de control es la encargada de suministrar la señal de *trigger* al medidor y evaluar la salida *echo* para obtener la distancia al objeto en binario así como de activar la señal *alarma*. El convertidor recibe el resultado de la distancia y lo suministra a los visualizadores.

El sistema completo es diseñado por los alumnos a lo largo de cuatro prácticas. En primer lugar, se diseña y simula la unidad *control_maxsonar* (Práctica 1). Posteriormente, en la Práctica 2, se emula mediante código VHDL el medidor HC-SR04. Esto permite disponer de señales *echo* de duración aleatoria y así poder simular medidas de distancias. En la Práctica 3, se lleva el sistema completo a la placa. Para ello, se termina de construir el módulo principal añadiendo el convertidor y eliminando el emulador del medidor, puesto que ya el módulo principal podrá interactuar con el medidor real. Por último, en la Práctica 4 se completa el módulo principal con la salida *alarma* que se conectará a un zumbador.

III. DESARROLLO DE LAS PRÁCTICAS

A continuación pasamos a detallar cada una de las prácticas en que se ha descompuesto el diseño.

A. Práctica 1

En la Práctica 1 se crea la unidad de control, *control_maxsonar*. La misma tiene las siguientes entradas y salidas.

Respecto a las entradas: una señal de reloj *ck* de 100Mhz procedente de la placa de desarrollo, una señal de reinicio asíncrono activa en alta (*reset*), una señal de inicio de medición de distancia (*inicio*) y la señal *echo* procedente del módulo medidor. En cuanto a las salidas: una señal *trigger* que activa cada medición y se conectará con el módulo medidor, un bus de salida que contiene el valor de la distancia medida (*distancia*) y una señal que indica la validez de dicha medida (*data_valid*) (véase el bloque *control_maxsonar* de “Fig. 4”).

Las tareas que ha de realizar este módulo son las siguientes:

- Generar un pulso en la salida *trigger* cada vez que la señal *inicio* se active. Dado que *trigger* se conectará con el módulo medidor, se ha de garantizar que cumple las

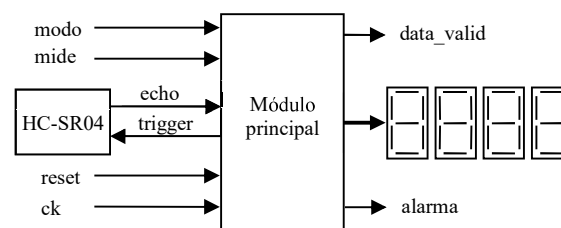


Fig. 2 Diagrama de bloques sistema medidor

especificaciones de dicho módulo (ha de ser mayor de 10µs) (“Fig. 3”).

- Medir el tiempo que la señal *echo* está a 1. Para ello se utilizará una señal de frecuencia adecuada a la precisión de la medida, que en nuestro caso es de 1cm. Con este fin se incorpora en el diseño un bloque IP de Xilinx para dividir la frecuencia de entrada que es excesiva. Con el reloj dividido se contará el número de ciclos que la señal *echo* está a 1 siendo ésta la medida de la distancia en centímetros.
- Activar la señal *data_valid* cuando se termine el proceso de medición de la distancia.

El alumno en la realización de la práctica tiene que determinar el número de bits necesario para la salida distancia en función de su límite superior (4m).

El diseño debe realizarse utilizando una máquina de estados cumpliendo las restricciones de síntesis. También ha de crear un testbench para simular la funcionalidad del módulo diseñado.

B. Práctica 2

En la Práctica 2 se diseña un modelo funcional del medidor de distancia que nos permitirá probar el correcto funcionamiento del conjunto medidor-control_maxsonar mediante simulación antes de pasar a la prueba con el medidor real y la placa.

Una vez que la simulación es completada con éxito los alumnos dispondrán de la placa de desarrollo Nexys4 [5] junto con el medidor real para hacer las pruebas experimentales. Para la conexión entre ambos se establece que la salida *trigger* y la entrada *echo* del control se conecten a pines determinados de la placa. La señal *ck* deberá conectarse al reloj de 100Mhz de la placa, la señal *reset* y la señal *inicio* a sendos pulsadores, la salida *distancia* a un conjunto de leds y, por último, la señal *data_valid* también a otro led.

Para probar el conjunto, el alumno deberá activar el *reset* y a continuación activar *inicio* mediante los pulsadores y esperar el encendido del led asociado a *data_valid* e interpretar en binario el valor de la distancia mostrado en los leds.

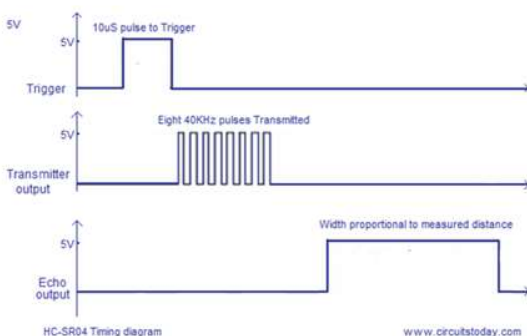


Fig. 3 Funcionamiento del medidor de distancia HC-SR04.

C. Práctica 3

En la Práctica 3 se realiza un diseño llamado *sistema_maxsonar* que contiene el ya realizado *control_maxsonar*, un sistema de conversión de código binario a 7-segmentos para la salida *distancia* y un circuito adicional que simplemente es un multiplexor (*mux*) para controlar la señal inicio en función de las entradas *modo* y *mide*.

Para el sistema de conversión se ha querido reutilizar un diseño que el alumno desarrolló en una práctica anterior que consistía en un convertidor BCD/7-segmentos de 4 dígitos. Por tanto, a este módulo habrá que añadirle un nuevo convertidor binario/BCD.

En la “Fig. 4” se muestra el diagrama de bloques del *sistema_maxsonar*.

Para probar el funcionamiento del sistema en la placa de desarrollo se utilizarán los mismos pines utilizados en la Práctica 2 para *echo* y *trigger* mientras que la entrada *mide* y *modo* se conectarán a un pulsador y a un conmutador respectivamente.

D. Práctica 4

En la Práctica 4 se completa el diseño añadiendo la salida *alarma*. El objeto de esta salida es poder generar una señal sonora que indique la proximidad a un objeto. Para ello se dispone de un zumbador al que conectaremos la salida *alarma*. El modo de funcionamiento es el siguiente: para que el zumbador esté apagado la salida *alarma* debe tomar el valor 1 y para que esté encendido debe tomar el valor 0.

Se considerarán cuatro casos en función de la distancia al objeto:

- Si el valor de la distancia al objeto está comprendida entre 100cm y 75cm, se producirán pitidos cortos pero distanciados en el tiempo.
- Si el valor de la distancia está en el intervalo entre 75cm y 50cm los pitidos serán algo más largos y menos distanciados.
- En el caso de que la distancia al objeto esté entre 50cm y 25cm, los pitidos estarán aún menos distanciados.
- Finalmente, si la distancia es inferior a 10cm, el

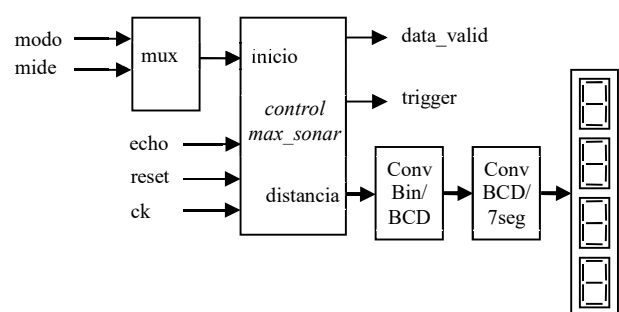


Fig. 4 Diagrama de bloques detallado del sistema max_sonar

pitido habrá de ser continuo.

Para conseguir estos objetivos, el alumno debe modificar el código anterior y programar la placa comprobando posteriormente el funcionamiento del sistema completo.

En las “Fig. 5, 6 y 7” se presentan unas imágenes obtenidas en el laboratorio. En la “Fig. 5” se muestran los módulos que se conectan a la placa de desarrollo: el medidor de distancia HC_SR04 y el zumbador utilizado para generar la señal sonora. Las otras dos imágenes (“Fig. 6” y “Fig. 7”) corresponden a fotografías que fueron tomadas en el laboratorio para ilustrar el resultado al que llegaron los alumnos tras el desarrollo de las prácticas. En concreto, la “Fig. 6” es el resultado de la Práctica 3, donde la distancia al objeto se muestra en los 4 visualizadores 7-segmentos y la “Fig. 7” corresponde al resultado obtenido tras la realización de la Práctica 4 dado que se incorpora el módulo zumbador.

IV. CONCLUSIONES

Con este conjunto de prácticas hemos conseguido que el alumno se enfrente a realizar un diseño de cierta complejidad mediante la descomposición de éste en diseños más sencillos. También se ha conseguido el objetivo de que cada una de las partes en que se ha descompuesto el diseño corresponda a una única sesión de laboratorio. Además se incorpora, en cada caso, el proceso completo de diseño, es decir, la escritura del código

VHDL, depuración de errores, escritura del fichero de estímulos (*testbench*) que permite la simulación funcional y, en su caso, la implementación en la placa y su comprobación experimental.

AGRADECIMIENTOS

Este trabajo ha sido parcialmente soportado por los proyectos CESAR (TEC2013-45523-R), INTERVALO (TEC2016-80549-R) y LACRE (CSIC 201550E039).

REFERENCIAS

- [1] Puede consultarse el proyecto docente en el enlace: http://www.us.es/estudios/grados/plan_201/asignatura_2010034 (accedido en febrero 2018).
- [2] Puede consultarse el proyecto docente en el enlace: http://www.us.es/estudios/grados/plan_201/asignatura_2010011 (accedido en febrero 2018).
- [3] Puede consultarse el proyecto docente en el enlace: http://www.us.es/estudios/grados/plan_201/asignatura_2010018 (accedido en febrero 2018).
- [4] Pueden consultarse las especificaciones del medidor HC-SR04 en <http://www.micropik.com/PDF/HCSR04.pdf>
- [5] Pueden consultarse las especificaciones de la placa de desarrollo Nexys4 en: <https://reference.digilentinc.com/reference/programmable-logic/nexys-4-ddr/reference-manual>

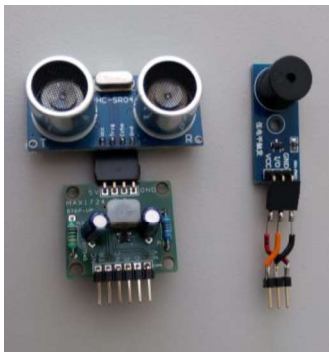


Fig 5 Medidor de distancia (izq) y zumbador (dcha).

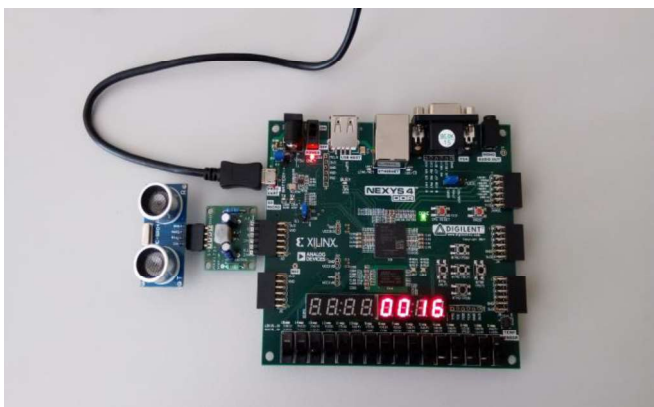


Fig. 6 Placa de desarrollo y medidor. Resultado tras Práctica 3.



Fig. 7 Placa de desarrollo, medidor y zumbador. Resultado tras Práctica 4.