

TARJETA DE DISEÑO ABIERTO PARA DESARROLLO Y EDUCACIÓN

Diego J. Brengi, Salvador E. Tropea, Juan P. D. Borgna

Electrónica e Informática
Instituto Nacional de Tecnología Industrial
Buenos Aires, Argentina
email: {brengi, salvador}@inti.gov.ar

RESUMEN

En este artículo se presenta el diseño e implementación de un circuito impreso con FPGA, pensado para ser utilizado como ayuda y referencia para laboratorios de desarrollo e instituciones educativas que deseen trabajar con esta tecnología. Este trabajo forma parte del proyecto FPGALibre cuyo objetivo principal es el de facilitar el intercambio de los elementos necesarios para el desarrollo con dispositivos FPGA. Por este motivo, para el diseño y la verificación del circuito impreso se utilizaron solamente herramientas de software libre. El diseño realizado se ofrece abiertamente para ser reproducido, modificado, adaptado o comercializado.

Palabras clave: hardware, FPGA, educación, desarrollo.

1. INTRODUCCIÓN

El intercambio y la copia de circuitos físicos (*hardware*) es complejo debido al costo de replicación y por eso no pueden aplicarse los mismos criterios que para el software libre [1]. Sin embargo puede mantenerse el mismo espíritu con los archivos y toda la información de diseño asociada a un circuito. A este concepto se lo llama “Hardware Libre”, “Hardware Abierto” o “*Free Hardware*” [2], [3], [4]. Es importante comprender además que las herramientas de software para realizar estos diseños también deberían ser libres para no condicionar el uso, adaptación y mejora del hardware en cuestión.

Basados en esta filosofía y dentro del proyecto FPGALibre [5], [6], se aborda el diseño e implementación de un circuito impreso con FPGA, pensado para ser utilizado en ámbitos de desarrollo e instituciones educativas.

Cabe mencionar que existen varios circuitos publicados con FPGAs [7], [8], [9], [10], utilizados para los mismos fines, sin embargo ninguno de los observados cumplía con los objetivos y pautas de diseño propuestos en la siguiente sección.

2. OBJETIVOS Y PAUTAS DE DISEÑO

Los objetivos concretos planteados para este desarrollo fueron:

- Obtener un diseño de referencia para la implementación de una placa con FPGA que sea de utilidad para cualquier interesado en abordar esta tecnología pensando en la utilización de componentes de fácil acceso y sin demandar demasiados recursos económicos.
- Adoptar la modalidad y los conceptos de “Hardware Libre” antes, durante y luego del desarrollo.

Para lograr estos objetivos se impusieron las siguientes pautas de diseño:

- Utilizar herramientas de software libre para permitir la modificación del circuito sin forzar la adquisición de costosos programas de diseño de PCBs.
- Seleccionar componentes que puedan soldarse en forma manual.
- Seleccionar, dentro de lo posible, componentes muy utilizados que puedan obtenerse fácilmente en los comercios.
- Circuito simple, sencillo y de funcionalidades básicas.
- Circuito impreso que pueda fabricarse regionalmente.
- Dispositivo FPGA con capacidad suficiente para permitir realizar prácticas y desarrollos de baja y mediana complejidad.
- Bajo costo. Si bien es imposible competir con productos fabricados en grandes cantidades, existen otros factores a tener en cuenta a la hora de abordar la fabricación de hardware como por ejemplo la capacidad para adaptar y modificar los circuitos, los altos costos de importación (monetarios y administrativos) para pocas unidades, el grado de independencia tecnológica, la experiencia adquirida, etc.

3. DESARROLLO

Como punto de partida se buscó realizar un diseño de similares características y prestaciones que la placa “Memec 2SLC board Spartan II demo board (DS-DB -2SLC)” que posee una FPGA Spartan II XC2S100 con encapsulado PQ208. Habiendo tomado experiencia con este dispositivo y debido a la facilidad de manejo del encapsulado mencionado con respecto a las alternativas BGA, se optó por utilizarlo en el diseño, dejando para un proyecto posterior la utilización de dispositivos más modernos como Spartan 3. De esta forma se logró un primer diseño bautizado S2PROTO.

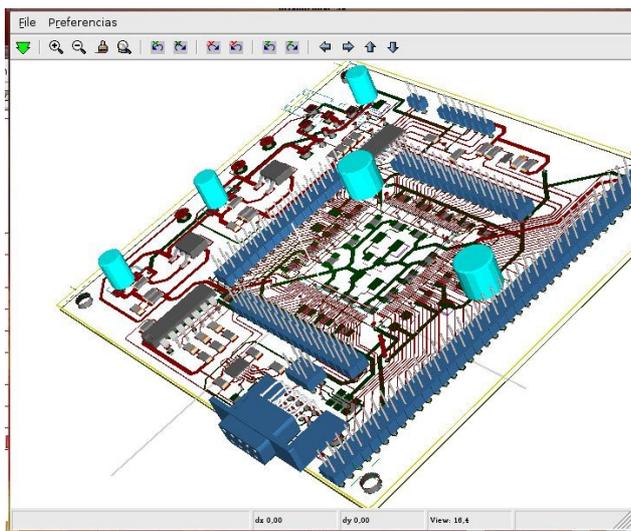


Fig. 1. Visualización 3D del diseño en Kicad.

Luego de la fabricación de los impresos, se ensamblaron tres de ellos, utilizados exitosamente en trabajos propios del laboratorio, con los que se estudiaron los defectos de diseño y la posibilidad de realizar mejoras en el mismo. A partir de la experiencia obtenida se encaró el desarrollo de un segundo diseño buscando las siguientes mejoras:

- Se agregaron leyendas en la serigrafía para facilitar el ensamble, y una mejor indicación de las señales y los conectores.
- Se utilizaron reguladores tipo LM1117 de bajo costo y fácil adquisición en las tiendas locales.
- Se bajó el costo del impreso mediante una reducción del 20% en el área del mismo al quitar el sector de prototipo rápido.
- Se aumentó la capacidad de E/S modificando algunos de los conectores y se agregó un conector DB-9 para la comunicación serie.

3.1. Software de diseño

Para el diseño de los esquemáticos y del circuito impreso se seleccionó el software Kicad [11], por considerarlo la herramienta más evolucionada de su tipo dentro de la categoría del software libre. Kicad posee licencia GPL [12], y existen versiones para sistemas operativos Windows y GNU/Linux. Los ejemplos suministrados por el autor demostraron que las capacidades del software cumplían con las necesidades del proyecto. Para este trabajo se utilizó sistema operativo Debian GNU/Linux [13]. La Fig. 1 muestra el modo de visualización tridimensional y la Fig. 2 muestra tanto el esquemático como el PCB.

3.2. Circuito impreso

El circuito impreso se realizó en doble faz teniendo un tamaño aproximado de 12x10 cm. Las pistas más delgadas

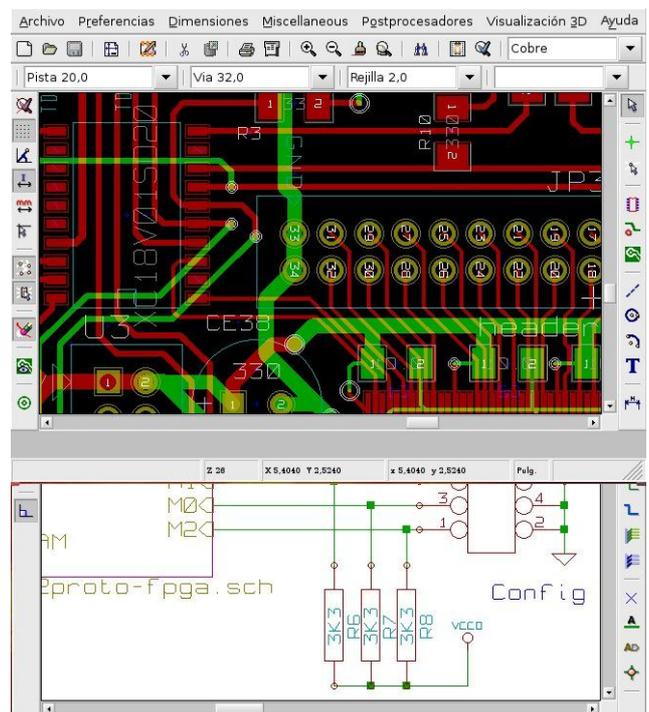


Fig. 2. Vista del pcb y el esquemático diseñados con Kicad.

fueron de 10 mils y para la verificación de espaciado mínimo se utilizó una distancia de 8 mils. La fabricación de circuito impreso se encargó a una empresa local a la cual se le enviaron los archivos Gerber generados con Kicad.

3.3. Componentes electrónicos utilizados

La placa posee muy pocos componentes. Mencionados brevemente son:

- FPGA: Xilinx XC2S150-PQ208 o XC2S100-PQ208
- Driver RS-232: MA3221
- Memoria: Xilinx XC18V01-SOC20
- Reguladores: LM1117-2.5 y LM1117-3.3
- Resistores, leds, capacitores y un transistor.
- Conectores, llave y pulsador.

Los componentes utilizados se consiguieron, en su mayoría, en las tiendas de electrónica e importadores del rubro. Las pocas excepciones, que debieron traerse por pedido, fueron el dispositivo FPGA, su memoria de configuración, el chip adaptador para niveles RS-232 y algunos de los capacitores SMD. En la documentación del trabajo se ofrece un listado de los proveedores utilizados para los componentes involucrados.

3.4. Soldadura de los componentes

La soldadura de los componentes se realizó manualmente en el laboratorio. Para esta tarea se utilizó una estación de soldadura para trabajos de reparación y montaje de compo-

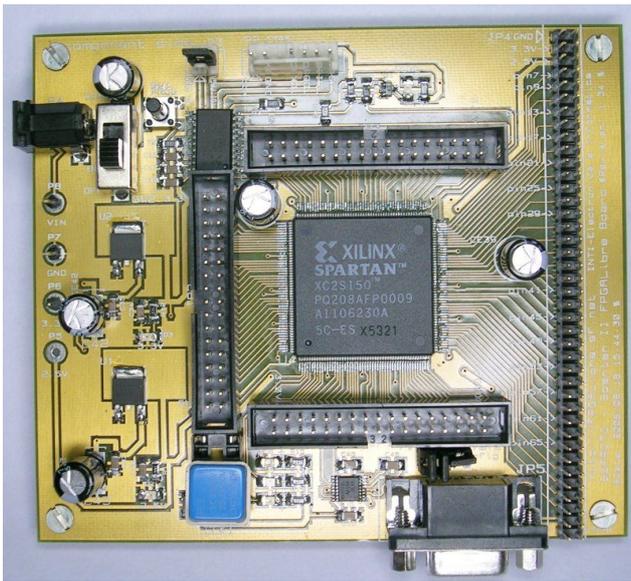


Fig. 3. Vista frontal de la placa S2PROTO v1.34.

mentos SMD. Sin embargo, esta estación sólo es necesaria para los componentes SMD de paso pequeño, tales como el dispositivo FPGA, el driver RS-232 y la memoria. Para soldar los primeros dos se utilizó la técnica de mini-ola (mini-wave). Posteriormente fue necesario corregir algunas soldaduras de la FPGA con una punta fina de 0.4 mm. La Fig. 3 muestra el circuito impreso con todos los componentes soldados.

3.5. Verificación y prueba

Terminado el ensamble se procedió con la etapa de verificación del circuito. Luego de comprobar las tensiones de alimentación se empleó la cadena JTAG para realizar pruebas de los terminales de entrada y salida, los leds, la transmisión y la recepción serie. Para esta tarea se empleó el programa GNU Jtag [14]. Por último, y como prueba final, se realizó la grabación de un *bitstream*. La documentación ofrece una guía detallada sobre como realizar paso a paso la primera puesta en marcha y la verificación inicial del circuito. La Fig. 4 muestra la placa desarrollada conectada al circuito JTAG.

3.6. Documentación

Para cumplir con los objetivos planteados, se generó extensa documentación del circuito desarrollado:

- Documento introductorio.
- Archivos de diseño del esquemático y del circuito impreso en formato Kicad.
- Archivos Gerber enviados para fabricación del impreso.
- Listado de proveedores de componentes utilizados.
- Listado de precios y estimación de costo unitario.
- Pasos para la verificación y prueba del circuito.

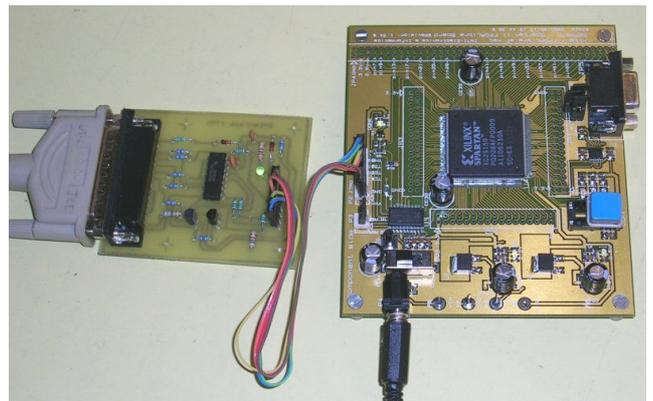


Fig. 4. Circuito JTAG y placa bajo prueba.

- *Bitstream* para verificación.
- Ejemplo VHDL para verificación.
- Licencia de uso y modificación.

En la Fig. 5 se presenta un diagrama en bloques descriptivo que forma parte de la documentación.

4. CARACTERISTICAS PRINCIPALES

Algunas de las principales características del circuito son:

- Diseño brindado bajo licencia GPL para permitir su libre utilización, implementación, modificación y comercialización.
- Diseño ensamblado y probado.
- Desarrollado y probado con herramientas de software libre: Kicad y GNU Jtag.
- Impreso doble faz de 12x10 cm de fácil fabricación y posibilidad de montaje manual de los componentes.
- Soporte para dispositivos Xilinx Spartan II PQ208 (hasta XC2S150 equivalente a 150.000 compuertas).
- Puerto de niveles RS-232 para implementar transmisión y recepción serie.
- Puerto JTAG compatible con el cable Parallel III de Xilinx, también conocido como DLC5[15]. Circuito realizado en Kicad, y también disponible en el proyecto FP-GALibre [16].
- 152 pines de I/O disponibles para el usuario distribuidos en cinco conectores. Estos permiten conectar plaquetas adicionales con pulsadores, leds, displays, etc.
- Alimentación simple de 5V CC.

5. RESULTADOS

Se ha obtenido un diseño básico y de referencia para el desarrollo de aplicaciones con FPGA, realizado totalmente con herramientas de software libre, de diseño claramente documentado y construcción accesible para cualquier pequeño laboratorio de electrónica.

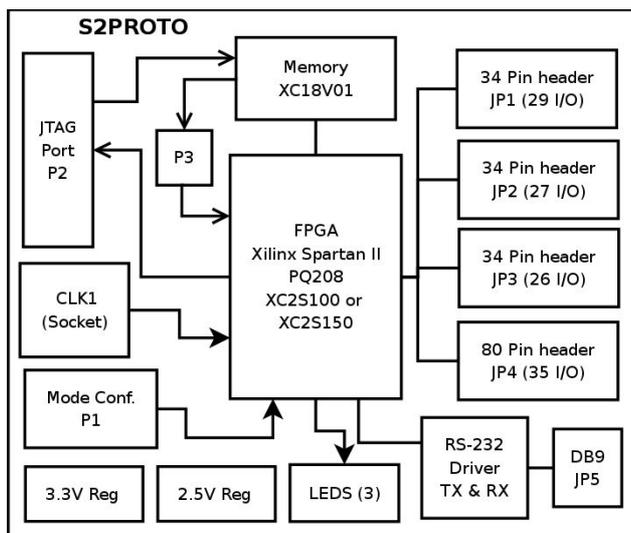


Fig. 5. Diagrama en bloques descriptivo.

Este diseño ha sido verificado y se emplea en los trabajos y prácticas realizadas en nuestro laboratorio. Toda la información del diseño se brinda libremente lo que permite copiar, adaptar, mejorar y comercializar este hardware según los requerimientos particulares de cada interesado.

Es nuestro deseo que este desarrollo junto con otras herramientas de software libre contribuyan a la difusión de esta tecnología dentro de laboratorios electrónicos de pequeñas y medianas empresas como así también en el ámbito universitario. Se contempla además la posibilidad de formar un grupo de interesados en este hardware y realizar una pequeña producción en serie para reducir y compartir los costos de fabricación de cada unidad.

Al momento de publicarse este artículo, toda la información de diseño mencionada podrá descargarse del sitio web del proyecto FPGALibre.

6. REFERENCIAS

[1] Free Software Foundation, Inc. “The Free Software Definition”, <http://www.gnu.org/philosophy/free-sw.html>.

[2] Opencollector.org, “Writings on Open Source Hardware”. <http://www.opencollector.org/Whyfree/>

[3] Ivan González, Juan González, Francisco Gómez-Arribas, “Hardware libre: clasificación y desarrollo de hardware reconfigurable en entornos GNU/Linux”, <http://www.learobotics.com/personal/juan/publicaciones/art4/hardware-libre.pdf>, Sept. 2003.

[4] D. J. Brengi, “Navegando en busca del “Free Hardware””, http://utic.inti.gov.ar/proyectos/free_hardware/index.html, May. 2004.

[5] Salvador E. Tropea, Diego J. Brengi, Juan P. D. Borgna, “FPGALibre: Herramientas de Software Libre para diseño con FPGAs” FPGA Based Systems. ISBN 84-609—8998-4, pp. 173–180, Mar. 2006.

[6] Proyecto FPGA Libre, <http://fpgalibre.sourceforge.net/>

[7] OPENCORES.ORG, <http://www.opencores.org/>

[8] Olmedo Sergio D., Pereyra Esteban N., Manfredi Gustavo A. “Kit de desarrollo educativo con CPLD” FPGA Based Systems. ISBN 84-609—8998-4, pp. 207–214, Mar. 2006.

[9] Juan P. Oliver, Sebastián Fernandez “Desarrollo de Plataformas Reconfigurables con Interfaz PCI como Proyecto de Grado” FPGA Based Systems. ISBN 84-609—8998-4, pp. 215–222, Mar. 2006.

[10] Fernando I. Szklanny, Elio A. De María, Carlos E. Maidana, Edgardo Gho. “Una plataforma didáctica para el aprendizaje de técnicas de diseño con circuitos lógicos programables” FPGA Based Systems. ISBN 84-609—8998-4, pp. 223–228, Mar. 2006.

[11] Jean-Pierre Charras, “Kicad: GPL PCB Suite”, http://www.lis.inpg.fr/realise_au_lis/kicad/

[12] GNU General Public License, <http://www.gnu.org/copyleft/gpl.html>

[13] Debian project. <http://www.debian.org/>

[14] Telka, M., et al, “JTAG Tools”, <http://openwince.sourceforge.net/jtag/>

[15] Xilinx, Inc. “JTAG Programmer Guide”, http://toolbox.xilinx.com/docsan/3_1i/pdf/docs/jtg/jtg.pdf

[16] Proyecto FPGA Libre, “Cable JTAG”, <http://fpgalibre.sourceforge.net/hard.html>