

Plataforma de Hardware Reconfigurable para el Diseño de Sistemas Digitales

Alexis Maximiliano Quiteros, Luis Alberto Guanuco, Sergio Daniel Olmedo
Centro Universitario de Desarrollo en Automoción y Robótica
Universidad Tecnológica Nacional
Facultad Regional Córdoba
Email: {50214,lguanuco}@electronica.frc.utn.edu.ar, solmedo@scdt.frc.utn.edu.ar

Resumen—La constante evolución de los sistemas electrónicos (digitales y analógicos) exige la búsqueda de nuevas herramientas para la formación académica. En el caso del diseño de sistemas digitales una excelente alternativa es el uso de placas de evaluación basadas en dispositivos lógicos programables (PLDs). En función de los requerimientos y necesidades académicas que demandan recursos de hardware, y las oportunidades concretas de desarrollar una plataforma ajustada a las necesidades planteadas es que se presenta una plataforma reconfigurable con especificaciones abiertas. Este diseño cuenta con una FPGA (Field Programmable Gate Array) que dispone de una gran cantidad de recursos internos para el uso en sistemas digitales avanzados, pero además cuenta con periféricos básicos con los que se puede interactuar en la implementación de sistemas digitales. El proyecto se publica en forma libre (licencia GPL) buscando incentivar a otros grupos académicos en la modificación y adaptación de este trabajo a sus necesidades como así también proponer mejoras en versiones futuras de la plataforma.

I. INTRODUCCIÓN

Las áreas académicas vinculadas a la electrónica y la computación se encuentran en constante demanda de recursos educativos de hardware y software en virtud de potenciar los conocimientos de los estudiantes. En el caso de las tecnologías con poca difusión o implementación en la industria regional, la principal opción es la importación de plataformas educativas adquiridas a empresas destinadas a la manufacturación de sistemas embebidos. Estas plataformas comerciales se clasifican según su implementación por lo que no siempre cubren los requerimientos académicos. Por ejemplo, en el área de las técnicas digitales, los requerimientos de hardware para las cátedras iniciales difieren de las cátedras avanzadas. Esta situación presenta la oportunidad de desarrollar una plataforma a la medida de las necesidades de las instituciones académicas. Si se dispone de las especificaciones por parte de los docentes y la articulación de laboratorios y grupos de investigación, es posible obtener un desarrollo que cubra las expectativas y aliente a la producción regional de plataformas educativas en un marco de transferencia de tecnología.

En el proceso de aprendizaje de las Técnicas Digitales necesariamente se deben implementar los diseños digitales. Desde el Álgebra de Bool, con operaciones digitales simples, hasta la implementación de un microprocesador son prácticas comunes de los sistemas digitales lógicos y resulta fundamental su ejercitación para concluir el ciclo de enseñanza.

Al comienzo de la década de los 90s surgieron varios trabajos donde se planteaba la necesidad de una plataforma educativa orientada a la implementación de diseños lógicos digitales basados en PLDs. Los principales demandantes eran diseñadores de arquitecturas de microprocesadores [1], desarrollos que años anteriores resultaban dificultosos por el costo de la implementación en hardware. El avance en el proceso de integración de los circuitos integrados ha llevado a que se desarrollen plataformas más complejas que ofrecen una gran cantidad de recursos de hardware. Al día de hoy se han generado varios proyectos desarrollados por instituciones académicas [2][3][4], otras con especificaciones abiertas [5][6] y también con fines comerciales [7]. Todos estos trabajos tienen algunas características en común¹:

- El dispositivo lógico programable central es una FPGA
- Poseen Memoria de configuración de la FPGA
- El acceso al dispositivo es a través de JTAG
- Disponen de algún software para interactuar con la plataforma desde una computadora
- Tienen dos perfiles de diseño:
 - Para la implementación de sistemas lógicos generales
 - Orientado a un área específica

En función del perfil del usuario de la plataforma se definen los dispositivos que se utilizarán. La Tabla I ilustra una clasificación de los recursos que ofrecen diferentes plataformas basada en dispositivos PLDs. A niveles iniciales en el estudio de la lógica digital se requieren periféricos básicos como ser llaves conmutadoras de estados lógicos, pulsadores, dispositivos indicadores como diodos LED, etc. A un nivel medio se manejan controladores para display gráficos LCD/LED, comunicaciones entre varios dispositivos mediante SPI, I2C, etc. Y por último, en la formación de especialistas de sistemas embebidos, se requieren recursos como interfaces físicos para ethernet, controladores HDMI, USB, y otros más.

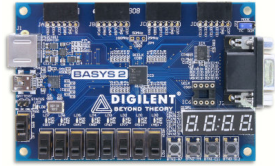
Las principales empresas fabricantes de sistemas embebidos basados en dispositivos PLDs son Xilinx, Altera y Digilent. Estos desarrollos se encuentran orientados a,

- Sistemas de comunicaciones
- Procesamiento de Señales Digitales (DSP)

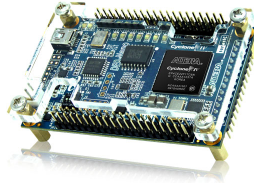
¹La caracterización anterior no es un intento de generalizar a todas las plataformas educativas basadas en PLDs, pero sí resulta útil para definir el perfil de la plataforma que se describe en este trabajo.

Tabla I
RECURSOS DE HARDWARE EN FUNCIÓN DE LOS NIVELES DE APRENDIZAJE

Nivel	Llaves/pulsadores Diodos LED	ADC&DAC/SPI Display LCD/VGA	USB/ETH HDMI
Inicial	✓		
Medio	✓	✓	
Avanzado	✓	✓	✓



(a) BASYS2 (Digilent)



(b) DE0-Nano (Altera)



(c) Avnet Spartan-6 LX150T (Xilinx/Avnet)

Fig. 1. Plataformas comerciales de desarrollo educativas basadas en FPGAs.

- Automoción

En la Fig. 1 se pueden ver tres diferentes plataformas orientadas al diseño de sistemas digitales². Los recursos de hardware que ofrecen estos desarrollos son:

- FPGA
- Memoria de configuración de la FPGA
- Periféricos básicos (LEDs, display, pulsadores, llaves, etc.)
- Puerto USB
- Puerto para módulos externos
- Puerto para propósitos generales
- Varias señales de reloj (clock)
- VGA
- PS/2
- Memorias ROM/RAM
- ADC/DAC

En nuestra región las tecnologías PLD se encuentran integradas en varias líneas de investigación y desarrollos hace algunos años. Instituciones gubernamentales de defensa [8], aeroespaciales, comunicaciones [9] están implementando dispositivos como FPGAs y CPLDs en sus sistemas electrónicos. Además existe una constante actualización por parte de las instituciones académicas en los programas analíticos de las carreras relacionadas a los sistemas embebidos [10].

La primera experiencia sobre el desarrollo de una plataforma orientada a la enseñanza de lógica programable ha sido realizada en el año 2006 [11]. En esta versión se

²Alguna de estas plataformas disponen de módulos conversores ADC y DAC, por lo que se podría decir que también permiten la implementación de sistemas analógicos en dominio discreto.

trabajó con un CPLD de Xilinx, a éste se conectaron algunos periféricos simples necesarios para las cátedras de Técnicas Digitales.

El Instituto Nacional de Tecnología Industria impulsa un proyecto denominado FPGALibre [12]. Este proyecto busca desarrollar y brindar herramientas de software libre y diseños de hardware abiertos para trabajar con tecnologías FPGA [13]. Dentro de este proyecto se destaca el desarrollo de una plataforma basada en una FPGA orientada a las áreas de educación y desarrollo de prototipos [14].

Las plataformas anteriormente nombradas son referencias de desarrollos nacionales usados en laboratorios universitarios. Sin embargo, la mayoría de las plataformas de evaluación comerciales son fabricadas en el exterior del país.

Considerando la situación expuesta es que se impulsa el desarrollo de la Plataforma de Hardware Reconfigurable (PHR). Esta plataforma es un proyecto a medida de las necesidades en la enseñanza de los sistemas digitales lógicos en las cátedras iniciales. Ofrece recursos básicos para que los estudiantes interactúen con la tecnología de los dispositivos PLDs, pero también dispone de puertos para conectar otros recursos físicos permitiendo que estudiantes avanzados puedan hacer uso de ellas sin limitaciones. Al ser publicado bajo licencia libre/abierta permitirá que el diseño, o parte de él, sirva como referencia a otras instituciones académicas que se encuentren en búsqueda de una plataforma para implementar en sus diferentes cátedras.

II. DISPOSITIVOS PRINCIPALES

Son varios los dispositivos principales que se deben definir antes de comenzar a conectar algún componente electrónico. En función de estos dispositivos es que se deben seleccionar los restantes. Se podrían listar:

- FPGA
- Memoria de configuración

A. FPGA

La FPGA que se utiliza pertenece a la familia Spartan-3 de Xilinx Inc. Esta familia a la vez se clasifican en

- Familia Spartan-3A extendida (bajo costo):
 - Spartan-3A
 - * Ideal para uso de interfaz entre dispositivos.
 - Spartan-3A DSP
 - * Mayor densidad de recursos en comparación que la familia Spartan-3A
 - * Dispone de un dispositivo DSP (DSP48A)
 - Spartan-3AN
 - * Dispositivos no volátiles
 - * Ideal para aplicaciones con restricciones de espacio
- Familia Spartan-3E
- Familia Spartan-3

Altera, Atmel y otros fabricantes de FPGAs también presentan familias similares a las Spartan-3. Aquí se optó por Xilinx Inc. debido a la experiencia en software/hardware con

Tabla II
CARACTERÍSTICA DE LA FAMILIA SPARTAN-3A

Devices	System Gates	Block RAM bits	Dedicated Multipliers	Maximum User I/O
XC3S50A	50K	54K	3	144
XC3S200A	200K	288K	16	248
XC3S400A	400K	360K	20	311
XC3S700A	700K	360K	20	372
XC3S1400A	1400K	576K	32	502

Tabla III
TIPO DE MEMORIA PARA LA FAMILIA SPARTAN-3A

Devices	Configuration Bits	ISP PROM Solution
XC3S50A	437,312	XCF01S
XC3S200A	1,196,128	XCF02S
XC3S400A	1,886,560	XCF02S
XC3S700A	2,732,640	XCF04S
XC3S1400A	4,755,296	XCF08P

que cuenta el Centro de Investigación³ donde se desarrolla el proyecto. La familia extendida Spartan-3A es la que se utiliza en el diseño de la PHR, que se distingue en la comparativa entre costo y recursos de hardware. Las Spartan-3A, permiten una gran variedad de modos de configuración en contraste con la familia Spartan-3. Por otro lado, no es necesaria una gran capacidad de procesamiento que justifique la inclusión de un DSP, debido al perfil del usuario de la plataforma que se desarrolla. Las principales características de las FPGAs Spartan-3A se describen en la Tabla II.

El dispositivo seleccionado, como se puede ver en la Tabla II, es el XC3S200A. Éste cuenta con una gran densidad de recursos de hardware (200K compuertas lógicas) a la vez que se puede encontrar en un encapsulado de pequeñas dimensiones (VQ100) que facilita el diseño del PCB (Printed Board Circuit). En este encapsulado se puede contar con 68 puertos de entrada/salida (I/O) para ser utilizados externamente a diferentes tecnologías programables (LVTTTL, LVCMOS33/25/18, entre otros). El perfil del diseño de la PHR no requiere de una gran cantidad de puertos de I/O debido a las aplicaciones para las que se lo diseña.

B. Memoria de configuración

La tecnología utilizada en las FPGAs Spartan-3A requieren de una memoria externa que configure al dispositivo ya que es volátil. Esta familia permite la utilización de varios tipos de memorias como modos de configuración para embeber el diseño digital en la FPGA. Xilinx comercializa memorias Flash PROM para todas sus familias de FPGA. Hay una relación directa entre la capacidad lógica de una FPGA con el tamaño de la memoria de configuración, en la Tabla III se puede apreciar esta relación para el caso de la familia Spartan-3A.

³CUDAR – Centro Universitario de Desarrollo en Automoción y Robótica.

Tanto la FPGA como la memoria de configuración Flash PROM se encuentran conectadas en cadena a través de una interfaz JTAG Boundary-Scan (IEEE 1149.1) que Xilinx Inc. implementa en sus dispositivos FPGAs, CPLDs y memorias Flash PROM para transferir los diseños sintetizados.

III. SISTEMA DE ALIMENTACIÓN

Por su arquitectura interna y la gran densidad de recursos lógicos que ofrecen, la necesidad de alimentar estos dispositivos en forma eficiente es de significancia en el diseño de la plataforma. Los sistemas complejos como las FPGAs requieren minimizar los ruidos presentes en las fuentes de alimentación y es aquí donde las fuentes de alimentación lineales son las ideales. Pero la desventaja de estos circuitos radica en la baja eficiencia que presentan (menor que 50%). Las fuentes de alimentación conmutadas (Switch-Mode) tienen una eficiencia mayor (alrededor del 90%) pero son más ruidosas en comparación con las lineales [15]. Con esta primera observación sobre los dos principales tipos de fuentes de alimentación, se deben considerar otros aspectos sobre el tipo de regulador a utilizar,

- Secuencia de arranque
- Inicio monotónico de la rampa de tensión
- Arranque suave
- Encapsulado y diseño del PCB

El TPS75003, fabricado por Texas Instruments, es un dispositivo que integra tres reguladores de tensión en una sola pastilla. Este integrado fue diseñado para aplicaciones donde se deben alimentar a FPGAs y DSPs. En especial, el TPS75003 fue testeado con las familias Spartan-3 de Xilinx proporcionando las tensiones necesarias para estas FPGAs. Esta compuesto por dos contradores Buck que logran una gran eficiencia y un regulador lineal LDO (Low-Dropout).

El Laboratorio de Desarrollo Electrónico con Software Libre, perteneciente al Instituto Nacional de Tecnología Industrial (INTI), ha realizado la implementación de un módulo de alimentación para placas con dispositivos FPGA [16]. En este desarrollo se utiliza el TPS75003 como dispositivo central y fue probado con una FPGA de la familia Spartan-3E. El diseño se encuentra disponible bajo una licencia de libre uso y modificación. Esta última aclaración no resulta un dato menor, pues la plataforma PHR persigue el mismo fin que el desarrollo realizado por el INTI. La idea de compartir y transferir desarrollos a la comunidad. Para la plataforma PHR se utiliza el módulo desarrollado por el INTI.

IV. PLACA PHR

Luego de las consideraciones sobre los dispositivos principales se presenta un diagrama en bloque de la plataforma PHR. La Fig. 2 incluye no solo la placa base que contiene la FPGA, la memoria de configuración de la FPGA y los periféricos sino también las placas de programación JTAG y la fuente de alimentación para todo el sistema.

La placa de alimentación llamada S3Power es el desarrollo mencionado en la Sección III. Es una placa basada en el dispositivo TPS75003 que proporciona los tres niveles de

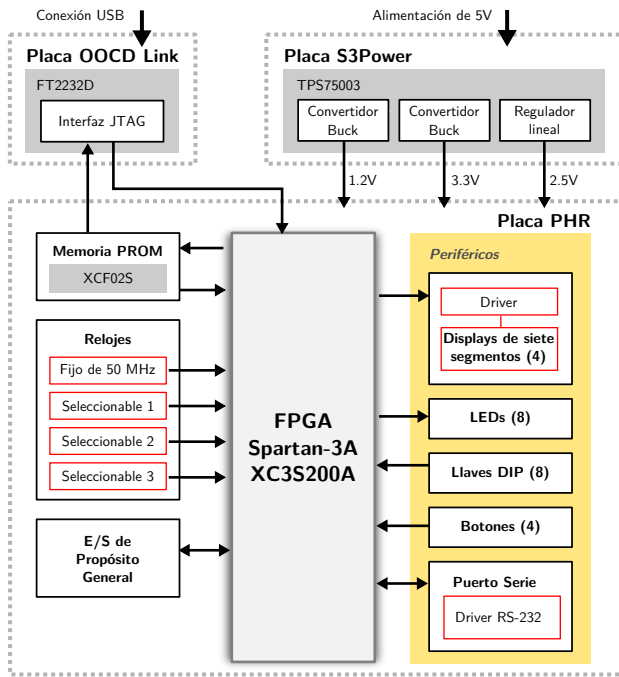


Fig. 2. Diagrama en bloque de la PHR.

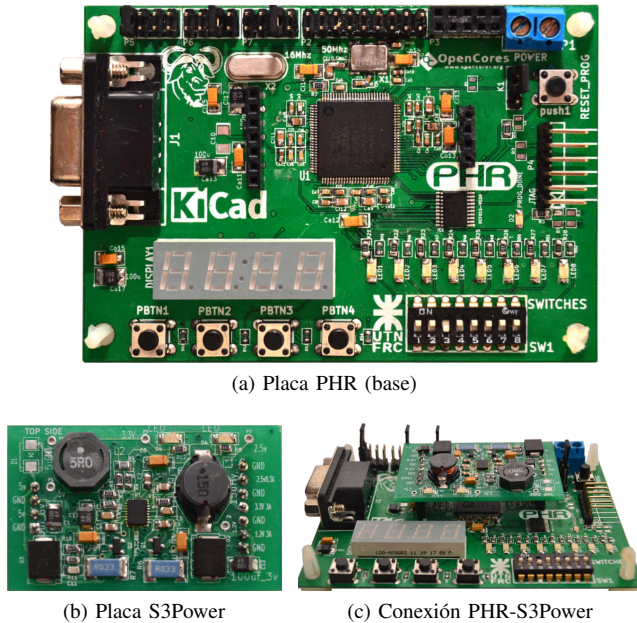


Fig. 3. Placas PHR y S3Power.

tensión para la FPGA (1.2V, 2.5V y 3.3V). La S3Power se conecta a la placa base PHR a través de dos conectores, uno para tomar la tensión de alimentación de todo el sistema (5V) y otro conector donde entrega las correspondientes tensiones para la FPGA. La ventaja de esta implementación es la posibilidad de reutilizar la S3Power en otros diseños que requieran las mismas especificaciones de potencia que la PHR. En la Fig. 3 se puede ver la placa PHR y su conexión con la placa S3Power.

A. Periféricos

Los periféricos que se implementan en este diseño permiten a los usuarios iniciales realizar prácticas sencillas. Pero también los usuarios avanzados requieren de indicadores de señales lógicas, pulsadores, etc. (Tabla I). Los periféricos que ofrece la plataforma PHR son:

- Salidas
 - 8 LEDs indicadores
 - Display de 7-segmentos cuádruples
- Entradas
 - 8 Llaves (DIP switch)
 - 4 Pulsadores
 - Relojes (Clocks)
 - * 50Mhz
 - * Divisor de reloj de 16Mhz a ~1Mhz
- Entrada/Salida
 - Puerto Serie (RS-232)
 - Puerto con I/O para propósitos generales (conectores IDE)

V. INTERFAZ JTAG

La plataforma PHR requiere interactuar con una computadora personal, sobre la cual el usuario realiza su diseño lógico mediante un lenguaje descriptivo de hardware (HDL). Para obtener la síntesis del diseño se utilizan las herramientas de software proporcionadas por el fabricante de la FPGA, luego se transfiere el diseño directamente a la FPGA o se almacenan los datos sobre la memoria Flash PROM.

Actualmente el puerto serie y paralelo, muy utilizados antiguamente, están quedando obsoletos. Por otro lado el puerto USB es la interfaz cableada más utilizada para la comunicación entre una computadora y dispositivos externos.

Los requerimientos planteados para la interfaz JTAG son:

JTAG Comunicarse con la plataforma PHR utilizando un dispositivo externo que implemente el estándar IEEE 1149.1.

USB Comunicarse con una computadora personal a través de este puerto sin restricción al sistema operativo a utilizar (GNU/Linux, Mac OS y Microsoft Windows).

Uno de los dispositivos comerciales que presenta las características anteriormente definidas es el FT2232D fabricado por Future Technology Devices International Ltd. (FTDI). El FT2232D dispone de una interfaz llamada MPSSE (Multi-Protocol Synchronous Serial Engine). Esta tecnología proporciona un medio flexible para comunicar dispositivos seriales síncronos a un puerto USB[17]. Por ser "Multi-Protocolo", el MPSSE permite comunicar con diferentes tipos de dispositivos síncronos, los más populares son SPI, I²C y JTAG. Otra característica del FT2232D es la disponibilidad de dos canales independientes. De esta forma en uno de los canales se puede implementar JTAG y en otro una UART, funcionalidad muy útil para comunicarse por RS-232 desde una computadora sin puerto serie. El esquema de la implementación para este

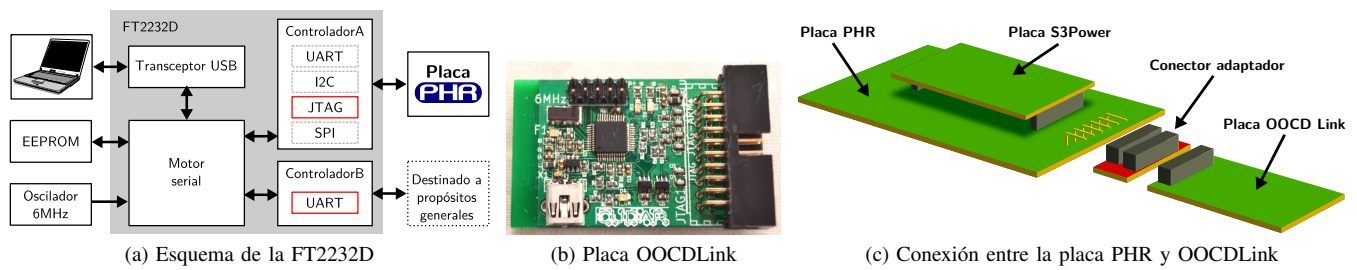


Fig. 4. Interfaz JTAG (implementación FT2232D).

dispositivo se presenta en la Fig. 4a, la placa se denomina OOCDDLink⁴.

En la Fig. 4b se muestra la placa OOCDDLink. En la misma se puede ver el conector mini-USB para la PC y un conector IDE de 20 pines donde se mapean las señales de JTAG. El puerto JTAG de la FPGA y la memoria Flash PROM utiliza solo cuatro pines del estándar IEEE 1149.1 (TDI, TDO, TCK y TMS). Pero la mayoría de los microcontroladores utilizan dos señales más desde el puerto JTAG para realizar debugging. Es por esta razón que la placa OOCDDLink utiliza el conector IDE de 20 pines agregando las señales TRST y SRST. En la Fig. 4c se puede ver como se conectarían las placas PHR y OOCDDLink. El adaptador puede también ser evitado realizando un cableado simple.

VI. PROCESO DE CONFIGURACIÓN Y PROGRAMACIÓN

Además del hardware se desarrollan algunas herramientas de software necesarias para la configuración/programación del diseño lógico en la FPGA/Flash PROM.

Los lenguajes descriptivos como VHDL y Verilog se definen en los estándares ANSI/IEEE 1076-1993 y IEEE 1364-1995 respectivamente. Esto asegura que los diseños descritos puedan ser reutilizados. En cambio la implementación sobre el dispositivo PLD difiere según el fabricante (a través de sus diferentes herramientas de software). En nuestro caso, se utilizan las herramientas de Xilinx para realizar la implementación y generación de archivos de configuración para la FPGA.

A. PHR GUI

La transferencia a la FPGA o la memoria Flash PROM es mediante un software llamado xc3sprog. Su nombre, xc3sprog, hace referencia a que inicialmente fue diseñado para la familia de FPGA Spartan-3 de Xilinx. Sin embargo se ha extendido el manejo a varios otros tipos de dispositivos que incluyen otras FPGAs, CPLDs, XCF flash PROMs, microprocesadores AVR de Atmel y memorias flash SPI. El xc3sprog soporta varios cables JTAG, incluyendo cables de puerto paralelo y programadores USB.

La PHR GUI se desarrolla con el uso de herramientas libres python, wxGlade y el xc3sprog que se ejecuta por debajo de esta interfaz amigable para los usuarios de la PHR. La Fig. 5 es una captura de pantalla del software PHR GUI.

⁴El nombre OOCDDLink es tomado de un desarrollo publicado en una web de sistemas embebidos sobre el cual se baso la placa.

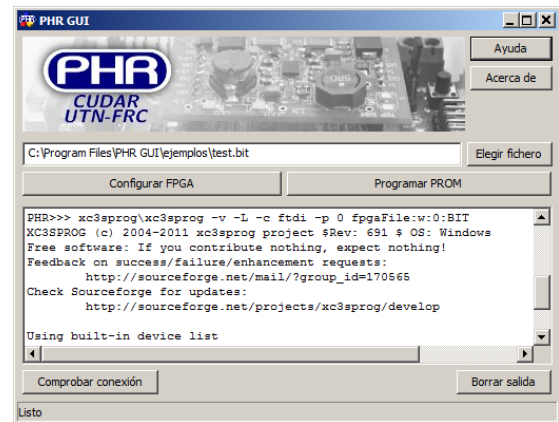


Fig. 5. Captura de pantalla del software PHR GUI.

El uso de este software es muy sencillo, solo se debe proporcionar el archivo generado por las herramientas de Xilinx (.bit) y elegir si se quiere configurar la FPGA o programar la memoria Flash PROM.

VII. CONCLUSIONES

El desarrollo del proyecto PHR ha requerido pasar por todas las etapas del proceso de producción de sistemas electrónicos. Desde los primeros diagramas en bloque, pasando por el diseño del esquemático y PCB de las diferentes placas. Además se realizó la compra de todos los componentes sin intermediarios debido al volumen requerido. Por cada una de estas etapas se realizaba documentación que permita afrontar proyectos similares o simplemente esta documentación sirva como referencias por parte de los estudiantes.

Desde un concepto estratégico se consideró disponer de la etapa de alimentación (Sección III) y la interfaz JTAG (Sección V) en forma independientes a la placa principal PHR. Ambas placas pueden ser reutilizadas en otros proyectos por parte de los estudiantes que tengan acceso al proyecto PHR. Y es que la placa S3power está diseñada para alimentar cualquier sistema basado en las FPGAs Spartan-3 de Xilinx. De la misma forma la placa OOCDDLink soporta el protocolo JTAG que es muy utilizado en los microcontroladores actuales.

La modularidad de las diferentes placas, en contraste con el párrafo anterior, presenta la desventaja del costo en la fabricación de los PCBs.

El proyecto se realizó en su totalidad con herramientas de

software libre/abiertas. Por cada etapa del desarrollo se buscó alternativas libres que cubrieran los requerimientos del caso. Se tenía referencias sobre proyectos de las mismas envergadura pero la plataforma PHR requería nuevas tecnologías a implementar que han sido resueltas con herramientas desarrolladas por la comunidad de software/ hardware libre/abierto.

La transferencia del desarrollo se encuentran en ejecución. Se realiza tareas conjuntas con la formación del personal a cargo del Laboratorio donde se instalarán las plataformas. En principio se tiene una plataforma funcional e instalada sin problema alguno.

AGRADECIMIENTOS

Los autores quieren agradecer a los docentes de las cátedras de Técnicas Digitales I y IV de la Facultad Regional Córdoba – Universidad Tecnológica Nacional (UTN – FRC). Sus aportes en el planteo y desarrollo de la PHR han sido de gran ayuda. Además se agradece a todos los estudiantes que han participado en las diferentes etapas de desarrollo de la plataforma. Agradecemos a los responsables del Centro Universitario de Desarrollo en Automoción y Robótica (CUDAR) por permitir realizar este proyecto en su espacio como también el soporte técnico recibido. Al personal del Laboratorio de Técnicas Digitales e Informática de la (UTN – FRC) por formar parte de la transferencia. Por último también agradecer a la Agencia para el Desarrollo Económico de la ciudad de Córdoba (ADEC) por financiar parte del proyecto.

REFERENCIAS

- [1] Hiroyuki Ochi, *ASaver.1: An FPGA-Based Education Board for Computer Architecture/system Design*, Design Automation Conference 1997. Proceeding of the ASP-DAC'97. Asia and South Pacific. January 1997.
- [2] C. Chang, C. Huang, Y. Lin, Z. Huang and T. Hu, *FPGA Platform for CPU Design and Applications*, 5th. IEEE Conference on Nanotechnology. Nagoya, Japan. July 2005.
- [3] D. Kang, S. Hwang, K. Jhang, K. Yi, *A Low Cost and Interactive Rapid Prototyping Platform For Digital System Design Education*, IEEE International Conference on Microelectronic Systems Education, MSE'07. 2007.
- [4] J. Xing, W. Zhao and H. Hu, *An FPGA-Based Experiment Platform for Multi-Cores System*, 9th. International Conference for Young Computer Scientistis, ICYCS'08. 2008.
- [5] A. Cicuttin, M. Crespo, A. Shapiro, N. Abdallah, *Building an Evolvable Low-Cost HW/SW Educational Platform – Application to Virtual Instrumentation*, IEEE International Conference on Microelectronic Systems Education, MSE'07. 2007.
- [6] J. Lockwood, N. McKeown, G. Watson, G. Gibb, P. Hartke, J. Naous, R. Raghuraman and J. Luo, *NetFPGA - An Open Platform for Gigabit-rate Network Switching and Routing*, IEEE International Conference on Microelectronic Systems Education, MSE'07. 2007.
- [7] Z. Qingguo, Y. Qi, L. Chanjuan, H. Bin, *Port Embedded Linux to XUP Virtex-II Por Development Board*, IEEE. 2009.
- [8] Instituto de Investigación Científica y Técnicas para al defensa (CITEDEF), *Radar Láser*, url: <http://www.citedef.gob.ar/i-d/laser/>.
- [9] J. Siman, G. Jaquenod and H. Mascialino, *Fpga-Based Transmit/Receive Distributed Controller for the TR Modules of an L Band Antenna (SAR)*, 4th. Southern Conference on Programmable Logic, 2008.
- [10] P. Cayuela, *Actualización de la currícula – Incorporación de la lógica programable en ingeniería*, Jornada de Investigación y Desarrollo en Ingeniería de Software (JIDIS'07). Córdoba Argentina. 2007.
- [11] S. Olmedo, E. Pereyra, G. Manfredi, *Kit de desarrollo educativo con CPLD*, FPGA Based Systems. 2nd. Southern Conference on Programmable Logic, 2006.
- [12] INTI Electrónica e Informática, *Proyecto FPGA Libre*, url: <http://fpgalibre.sourceforge.net>.
- [13] S. Tropea, D. Brengi, and J. Borgna, *FPGAlibre: Herramientas de software libre para diseño con FPGAs*, FPGA Based Systems. Mar del Plata: Surlabs Project, 2nd. Southern Conference on Programmable Logic, 2006.
- [14] S. Tropea, D. Brengi, M. Visentin, C. Huy and R. Melo, *S3Protoni: Tarjeta de Hardware Libre con FPGA de encapsulado BGA*, XVIII Workshop Iberchip 2012. Playa del Carmen, México. February 29. 2012.
- [15] D. Canny, *Power-Supply Solutions for Xilinx FPGAs*, Maxim Integrated TM. (Application Notes). April 24, 2012.
- [16] C.Huy and D. Brengi, *Módulo de alimentación para placas con dispositivos FPGA*, Congreso de Microelectrónica Aplicada (μ EA2010). San Justo, Buenos Aires. 2010.
- [17] Future Technology Devices International Ltd., *FTDI MPSSE Basics*, (Application Notes). Document Reference No.:FT_000208, AN_135. December 2010.