

Sistema Reconfigurable para el Prototipado Rápido de Aplicaciones basadas en Bus PCI

FJ. Rincón, PJ. Navarro, P. Ortiz, C. Jiménez, A. Toledo, J. Suardíaz, A. Iborra

División de Sistemas e Ingeniería (DSIE)

Universidad Politécnica de Cartagena. Campus Muralla del Mar. 30202 Cartagena

andres.iborra@upct.es

Resumen

Uno de los principales objetivos en la implantación de un sistema de control consiste en garantizar que cumplirá determinadas restricciones temporales, las cuales dependen en gran medida del hardware usado para implementar el sistema. Actualmente, frente a los costes tanto económicos como temporales en el desarrollo de sistemas mediante ASICs, las arquitecturas reconfigurables y las técnicas de codiseño constituyen unas herramientas muy valiosas en la búsqueda del sistema óptimo.

Este artículo propone una arquitectura reconfigurable basada en dispositivos FPGAs, que permite aprovechar la potencia que ofrece el bus PCI; resultando idónea para afrontar el diseño de sistemas de inspección visual automatizados desde el punto de vista de codiseño.

La arquitectura que se presenta se basa en las tarjetas de desarrollo PCI Proto-Lab/PLX, de HK Meßsysteme GMBH, y en la XS40 de la empresa Xess.

1. Introducción

En la actualidad, el desarrollo de un sistema de control implica una metodología de diseño en la que hay que conjugar tanto aspectos hardware (HW) como software (SW). A esto hay que añadir una creciente tendencia en el uso del paralelismo y unos requerimientos de funcionamiento en tiempo real muy restrictivos. Todo ello hace que hoy en día sea muy difícil encontrar la solución óptima de diseño que permita decidir lo que se implementa en software y lo que se implementa en hardware, siendo necesario el análisis de diferentes sistemas antes de decantarse por uno en concreto.

En el caso del software se han realizado numerosos avances en el campo de la ingeniería del software que permiten agilizar el proceso de desarrollo. No obstante, en el campo del hardware existían severas limitaciones debidas al coste y tiempos de desarrollo asociados a la fabricación de circuitos específicos (ASICs). La aparición de los dispositivos lógicos programables, y en concreto las FPGAs, ha aportado no sólo una herramienta idónea para un prototipado rápido del software, sino también valiosos medios para la evolución del diseño gracias a los numerosos y potentes paquetes de simulación que suelen aportar los sistemas de desarrollo. Finalmente, la evolución que han experimentado los lenguajes de descripción de hardware (HDLs) como el VHDL, han posibilitado una mejora sustancial en el flujo de diseño desde el punto de vista del hardware. Con relativa rapidez se puede pasar del modelo a una descripción VHDL que es posible descargar y/o emular sobre una FPGA.

Sin embargo, a la par que se agiliza el flujo de diseño, en el caso de los sistemas de inspección visual automatizados, día tras día se demanda una cantidad cada vez mayor de datos que han de ser recogidos y procesados. Esto constituye un posible cuello de botella en el sistema a diseñar. Por ejemplo, una imagen de 1280x1024 pixels en la que la información asociada a cada píxel se codifique sobre 24 bits supone una cantidad de información equivalente a 4Mbytes. Si se tratase de diseñar un sistema de inspección que fuera capaz de procesar en tiempo real una serie de imágenes con estas características, sería posible que se llegasen a requerimientos del orden 80 Mbytes/s. En la Tabla 1 se muestra una comparativa de las características que presentan las arquitecturas de bus más usuales. Se puede observar que la arquitectura PCI presenta los valores más

Bus	Velocidad (MHz)	Ciclos/transfer.	Ancho de Datos (bits)	Máx Velocidad de transfer (Mbytes/s)	Ventajas	Desventajas
ISA	8	2	16	8	Es el más usado. De bajo coste y compatible	Lento
Microcanal	10	1	32	40	Más rápido que el ISA	Difícil de encontrar. No compatible y mayor coste
PCI	33	1	32	132	Rápido y potente. Posibilidad de Plug&Play	Mayor coste y no compatible con arquitecturas anteriores

Tabla 1. Comparativa de las arquitecturas de bus más habituales.

adecuados para los requerimientos que los sistemas de inspección automatizados necesitan.

Sin embargo, desarrollar una tarjeta de expansión basada en bus PCI se convierte habitualmente en una tarea ardua y compleja, requiriendo una fuerte inversión de tiempo y dinero antes de obtener resultados aceptables [1] [2]. La tarjeta PCI Proto-Lab/PLX [3] que oferta HK Meßsysteme GmbH ofrece una solución mejor. Permite a los diseñadores sacar partido de su bus de 32 bits y de su decodificador PCI preconfigurado para así poder centrarse en lo que es el diseño y análisis de funcionamiento de los sistemas que se desarrollan, sin tener que entrar en detalle en los vericuetos del PCI.

2. Estructura del sistema propuesto

Las figuras 1 y 2 esquematizan el sistema propuesto, en el cual se conecta la PCI Proto-Lab/PLX por una parte al bus PCI de un PC y por otro a la tarjeta XS40 de Xess. De esta manera se consigue que cada tarjeta aporte las siguientes características al conjunto:

Tarjeta PCI Proto-Lab/PLX

Esta tarjeta de desarrollo PCI se compone de los siguientes elementos [3]:

Controlador PLX9054: Constituye el corazón de la tarjeta de desarrollo. Según el modo de transferencia seleccionado (especificación PCI), realiza la conversión de las señales del bus PCI para el bus local y viceversa. Es en este elemento donde se encuentra el árbitro del bus local.

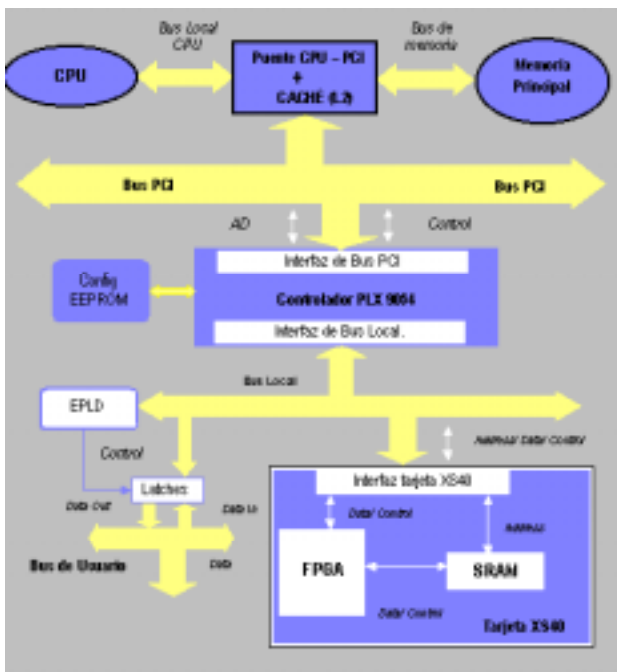


Fig. 1. Esquema del sistema propuesto.

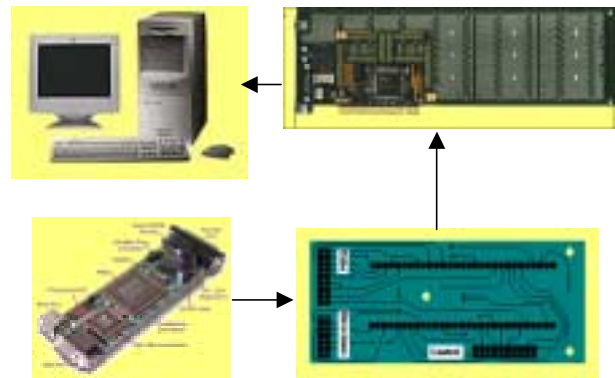


Fig. 2. Elementos del sistema.

Presenta tres formas diferentes de transferencia:

Operación PCI Initiator: El sistema conectado al bus local actúa como Maestro y el PC en el que se encuentra insertada la tarjeta actúa como esclavo.

Operación PCI Target: En este caso el sistema conectado al bus PCI (el PC) actúa como maestro del proceso de transferencia; mientras que el dispositivo conectado al bus local actúa como esclavo.

Operación DMA: Permite un acceso directo a memoria (acceso DMA) entre el controlador y la memoria conectada al bus local o entre el controlador y la memoria principal del PC.

Oscilador programable del bus: Se trata de un generador de la señal de reloj que sincroniza el bus local.

EEPROM serie: Es una memoria donde se almacenan los registros de configuración, los cuales inicializan el controlador en el arranque.

Conjunto de biestables: Se trata de unos elementos adicionales que ofrece el fabricante para permitir que el usuario se despreocupe del control de las señales que gobiernan el bus local. Este conjunto de latches permiten almacenar un conjunto de 32 bits durante las operaciones de entrada y salida. La información entra o sale de estos biestables a través del bus de usuario; lográndose una comunicación lo más transparente posible para el usuario. Para esto se requiere una lógica adicional localizada en el dispositivo EPLD de la tarjeta, que también reserva espacio para poder implementar las modificaciones que el usuario desee. Es capaz de acceder a los datos de entrada y salida del bus local, por lo que permite simular un diseño antes de montarlo físicamente en la tarjeta. El principal inconveniente que presenta es que se paga un precio por la facilidad de comunicación y es que el usuario no tiene un control total del proceso de comunicación. Por este motivo no se ha utilizado esta posibilidad en nuestra aplicación de ejemplo.

La figura 3 muestra el diagrama de bloques de esta tarjeta de desarrollo PCI. Se puede observar que el controlador PLX9054 se comunica tanto con el bus PCI como con el bus local. Observe también que el CPLD se encuentra conectado al bus local; lo que permite realizar una simulación del diseño antes del verdadero diseño físico, tal y como se ha comentado anteriormente.

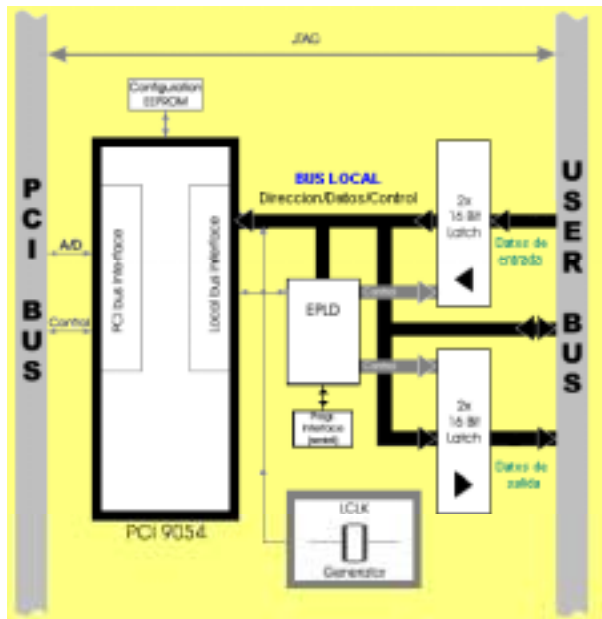


Fig. 3. Diagrama de bloques de la tarjeta PCI-ProtoLab/PLX.

Tarjeta XS40 de Xess

Se trata de una tarjeta (figura 4) de fácil configuración que, entre otros componentes, contiene: un dispositivo FPGA XC4010XL de Xilinx, un microcontrolador 8031, y una memoria SRAM de 32 Kb [4].

En el ejemplo que se muestra en este artículo, en la transferencia esta tarjeta desempeña la función de dispositivo esclavo; realizando además un procesamiento de los datos que el PC envía a través del bus PCI. Es posible también realizar transferencias del tipo "PCI Initiator", en las que la XS40 presentaría un comportamiento de maestro; o incluso usarla en transferencias de "tipo DMA".

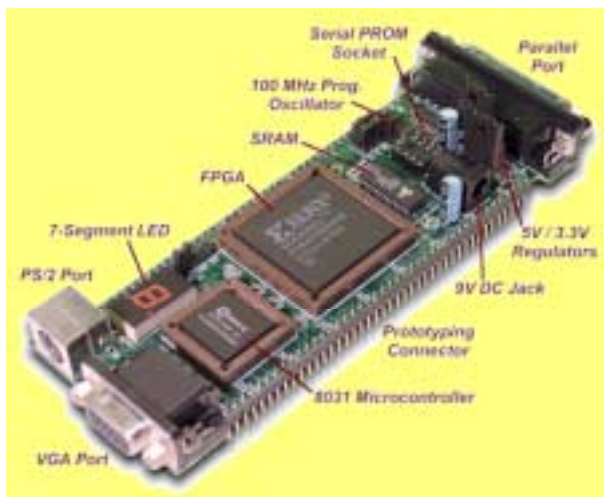


Fig. 4. Tarjeta XS40 de Xess.

El diseñador volcará sus diseños en la FPGA y/o en el microcontrolador que posee la tarjeta, teniendo así la función de dispositivo adicional conectado al bus Local de la tarjeta de desarrollo PCI. De esta forma se potencia la funcionalidad de esta última; dando la posibilidad de afrontar los diseños con una filosofía basada en el codiseño

y reconfiguración y constituyendo una plataforma idónea para el prototipado rápido de las diferentes opciones de particionado HW/SW, favoreciendo la posibilidad de realizar comparativas a fin de encontrar la idónea..

La principal desventaja que presenta esta tarjeta es que limita el bus de datos del sistema a 8 bits, lo que significa el desaprovechar los 32 bits que ofrece la tarjeta PCI.

3. Aplicación de ejemplo

La misión principal del controlador 9054 PLX es comunicar el bus PCI y el bus Local de forma acorde con la especificación PCI 2.2 [5]. De esta forma permite que el usuario no tenga que preocuparse en el protocolo de comunicación entre el controlador y el bus PCI. Por lo tanto, solo tendrá que preocuparse de las señales que controlan el bus local al que se conectan los periféricos hardware adicionales.

En el ejemplo que aquí se describe, se han conectado las tarjetas ya comentadas. Es posible observar de forma clara en las figuras 1 y 2 cómo la tarjeta se encuentra conectada al PC (sistema maestro) a través del bus PCI. En el bus local se encuentra la tarjeta XS40, que contiene la FPGA de Xilinx, que en este caso es el sistema esclavo.



Fig. 5. Módulo de procesamiento de imágenes.

La aplicación envía una imagen desde el PC a través del bus PCI; la cual pasa a través del controlador PLX9054, y finalmente alcanza la FPGA de la tarjeta XS40, en la que se efectúa un procesamiento de la imagen. Después, la imagen resultante se envía de vuelta al PC procedente de la memoria SRAM de la XS40, encargada de almacenar los bloques de información resultantes del procesamiento. De esta forma es posible efectuar un procesamiento de la imagen a través de hardware, lo que libera la carga del procesador del PC, el cual puede dedicarse a otra serie de tareas; logrando así potenciar las tareas de inspección.

El algoritmo de procesamiento que se lleva a cabo es una binarización, aunque es posible realizar operaciones más complejas cargando la biblioteca correspondiente en el módulo de procesamiento que se vuelca en la FPGA (figura 5). La FPGA no sólo se usa para el procesamiento de las imágenes, sino que también se usa para generar el protocolo de comunicación de la tarjeta XS40 con el bus local de la tarjeta de desarrollo. Para esto es necesario activar una serie de señales a fin de establecer una comunicación en modo esclavo a través del controlador 9054. El conjunto de señales implicadas en este proceso son las siguientes:

Líneas del controlador:

- LCLK, señal de reloj del bus local
- LRESET#, señal de RESET del bus local.
- LHOLD, solicitud de control del bus local.
- ADS#, address strobe
- BLAST#, bus last.
- LBE#, address lower bits

El dispositivo FPGA deberá responder mediante:

- LHOLDA, concesión del bus local
- READY#, para indicar la presencia de datos válidos en el bus local.

También se ha de controlar la memoria SRAM presente en la tarjeta XS40. Este se hace a través de las señales CE, WE y OE, que indican “Chip Enable”, “Write Enable” y “Output Enable” respectivamente.

La velocidad de procesamiento que puede alcanzar el sistema es del orden de 30 Mbytes/seg. No obstante, es posible alcanzar velocidades superiores, ya que la anchura del bus de datos del sistema está limitada a 8 bits, que es la que ofrece el bus de la XS40. A pesar de todo supone una velocidad considerablemente mayor que la que ofrecen otras arquitecturas de buses.

4. Resultados y Conclusiones

Se ha desarrollado una plataforma de prototipado rápido de arquitecturas reconfigurables basadas en bus PCI que ofrece una alta velocidad de procesamiento junto a un bajo coste. El conjunto está formado por una tarjeta de desarrollo PCI Pto-Lab/PLX (450€) y una tarjeta XS40 de Xilinx (120€). Esto hace posible conectar de una forma sencilla periféricos adicionales a través del bus PCI del ordenador sin tener que preocuparse de la complejidad adicional que presenta el protocolo de control del bus PCI, lo que la hace similar a otro tipo de sistemas que hay en el mercado, como las tarjetas RC1000PP (Celoxica), la Strathnuey (Nallatech), o la famosa Aristotle (Mirotech), siendo si bien no tan potente como estos últimos, sí mas económico y abierto que ellos.

El sistema permite que las transferencias a y desde estos periféricos se puedan desarrollar de forma de esclavo, de maestro, o mediante acceso directo a memoria (DMA). Por otro lado, la frecuencia de operación de la FPGA alcanza los 80 MHz, lo que no produce retrasos en la transferencia, ya que la frecuencia a la que opera el bus local es de 50 MHz.

módulos e implementar cada módulo en HW y en SW. Comparar el tiempo que requiere cada uno y en función de los resultados decidir lo que se implementa en hardware y lo que no. A modo de ejemplo, la tabla 2 resume una comparativa de tiempos de ejecución de desarrollo de un algoritmo de binarización implementado a través de una tarjeta Aristotle con el realizado a través de software para el procesamiento de imágenes MIL de Matrox. Se puede observar que, una vez configurado el dispositivo hardware, los tiempos de procesamiento mediante hardware son significativamente inferiores que los asociados al software.

La principal desventaja de este primer prototipo es su bus de datos limitado a 8 bits, ancho del bus de la tarjeta XS40, lo cual limita la velocidad de transferencia a 132MB/s dividido entre 4. El próximo paso será el dotar al sistema de una tarjeta con un bus de datos de 32 bits a fin de aprovechar la anchura de 32 bit que ofrece el bus local de la tarjeta de desarrollo.

Las líneas de trabajo futuras van desde conectar una cámara a la tarjeta Xilinx a fin de realizar procesamientos hardware sobre las imágenes que la cámara mande hasta usarla en la comunicación con un robot o aprovecharla para controlar motores.

5. Agradecimientos

Los autores desean expresar su gratitud al gobierno español por el apoyo CICYT (TIC-200-1765-C03-02) que están prestando al proyecto COSIVA, en el que está incluido el trabajo desarrollado en el presente artículo.

Referencias

- [1] E. Solari, G. Willse. “PCI Hardware and Software Architecture and Design”. Annabooks ISBN 0-929392-59-0, 1998.
- [2] Tom Shanley, Don Anderson “PCI System Architecture”, 4th Edition MindShare, Inc., ISBN 0-201-30974-2.
- [3] “PCI-Proto Lab/PLX Technical Manual”, HK Meßsysteme GmbH.
- [4] “XS40 Manual v. 1.4”, Dave Vanden Bout, Xilinx Inc.
- [5] “PCI Local Bus Specification”, Revision 2.2, PCI Special Interest Group (PCI-SIG) Hillsboro, Oregon.

*Tiempos de Procesado: Imagen 153x367x8bit.
Algoritmo de Binarización (umbral=200)*

<i>Tiempo(ms)</i>	<i>Empaquetado</i>	<i>Configuración</i>	<i>Desempaquetado</i>	<i>Procesado</i>	<i>Total</i>	<i>Imág./segundo</i>
<i>Software</i>	0	0	0	109	109	9
<i>Hardware</i>	4	121	0	10	14 (configurada) 135 (sin config.)	71 7

Tabla 2. Procesado HD/SW del algoritmo de binarización..

Esta velocidad de cálculo posibilita una metodología de codiseño a la hora de desarrollar algoritmos de procesamiento. Es posible dividir un procesamiento en