

Diseño e implementación de una plataforma para Voz sobre IP basado en FPGA.

Designing and Implementing a Voice over IP platform based on FPGA

Orlando Landrove Gámez

LACETEL[®], *landrove@lacetel.cu*

RESUMEN

El desarrollo de sistemas con microprocesadores embebidos dentro de un chip FPGA toma importancia pues permite diseños híbridos entre módulos hardware que ofrecen velocidad de procesamiento, e inteligencia a través de aplicaciones de alto nivel sobre un microprocesador, todo dentro de un mismo dispositivo. Muchos de ellos se sirven de sistemas operativos con aplicaciones multitareas que logran el procesamiento eficiente de protocolos modernos para las telecomunicaciones.

Asimismo es notable el crecimiento de aplicaciones de voz sobre IP (VoIP) sobre las múltiples redes estándares de telecomunicaciones, debido a su flexibilidad, gran capacidad de diseño y actualización que tributa a servicios cada vez más completos.

En este trabajo se describe la utilización del microprocesador MicroBlaze embebido sobre un Virtex5, una de las familias de FPGA de Xilinx. Se presentan sus módulos hardware para albergar un sistema operativo con el kernel 2.6 de Linux, probándose con la Tarjeta de Desarrollo ML507 de Xilinx.

Se utilizan aplicaciones sobre protocolos que se compilan e introducen dentro de los ficheros del Linux, soportado a su vez por el sistema MicroBlaze embebido dentro del FPGA.

Se obtiene un cliente VoIP capaz de establecer comunicación y registro con el servidor VoIP y demás clientes de la red IP, llámese teléfonos IP, software VoIP. Se logra el funcionamiento del protocolo SIP, así como la preparación del protocolo RTP para su utilización como transportador de paquetes de voz.

Palabras claves: FPGA, Sistema Embebido, MicroBlaze, Petalinux, VoIP.

ABSTRACT

The development of systems with embedded microprocessors within a FPGA chip, Field Programmable Gate Array, becomes important because it allows hybrid designs between hardware modules that provide processing speed, and intelligence through high-level applications on a microprocessor, all in one device. Many of them use operating systems with multitasking applications that achieve efficient processing of modern telecommunication protocols.

Also noteworthy is the growth of applications of Voice over IP (VoIP) over multiple networks telecommunication standards, due to its flexibility, high capacity and upgrade design that contribute to increasingly comprehensive services.

This paper describes the use of the MicroBlaze microprocessor embedded on a Virtex5, one of the Xilinx FPGA families. Hardware modules are presented to host operating system with Linux 2.6 kernel, using the Xilinx ML507 Development Board. Application of VoIP protocols are used, compiled and inserted into Linux files, supported in turn by the MicroBlaze embedded system within the FPGA.

VoIP client is obtained so able to communicate and register with the VoIP server and other IP network clients, like IP phones, VoIP software. SIP protocol performance is achieved, as well as the preparation of the RTP protocol to use it as a carrier of voice packets.

Keywords: FPGA, Embedded System, MicroBlaze, Petalinux, VoIP.

INTRODUCCION

En la actualidad crece el requerimiento de dotar de inteligencia y automaticidad al diseño de equipos y sistemas electrónicos. El constante desarrollo de elementos y dispositivos electrónicos hacen posible la inclusión de softwares cada vez más flexibles que añaden calidad en el funcionamiento y servicio del sistema.

Diversos productos de distintas tecnologías ofrecen la posibilidad de intercambiar con el usuario opciones y servicios que los convierten en herramientas multifuncionales. La incorporación de los llamados sistemas operativos embebidos o sistemas operativos de tiempo real (Real Time Operating System, RTOS) en dichos sistemas permite incrementar la capacidad configuración, administración y control.

El continuo auge de servicios y aplicaciones de VoIP demanda de plataformas capaces del manejo de los protocolos VoIP. Muchos de estos equipos además ofrecen otras posibilidades, como múltiples interfaces de configuración (CLI1, interfaces Web, entre otros), sistemas automatizados de alarmas y registros. Este valor agregado tributa a un mayor control, mantenimiento y administración del sistema.

La utilización de la tecnología FPGA2 permite responder tanto a necesidades de velocidad de procesamiento, a través del diseño mediante lenguajes de descripción de hardware (Hardware Description Language, HDL), como requerimientos de análisis y cómputo utilizando microprocesadores empotrados en el chip utilizando los módulos de propiedad intelectual (Intellectual Property, IP)[1].

El desarrollo de μ Linux, porción escogida y adecuada del kernel Linux, ha dado lugar al desarrollo de variantes de Linux con el objetivo de que sea soportado por varios tipos de microprocesadores. Petalinux es una versión adaptada de μ Linux compatible con las características de MicroBlaze, este último microprocesador definido en las herramientas de diseño para Xilinx.

Petalinux se muestra como un sistema operativo a tener en cuenta en el desarrollo de sistemas embebidos en FPGA.

El trabajo se traza los siguientes objetivos:

1. Establecer los módulos y periféricos con el microprocesador MicroBlaze necesarios para el soporte de Petalinux.
2. Compilar el Petalinux e introducirlo dentro del sistema en el FPGA.
3. Compilar, adecuar las librerías y ficheros que responden a protocolos VoIP e insertarlos como parte del sistema de ficheros de Petalinux en el FPGA.

CONTENIDO

¹ CLI: Command Line Interface

² FPGA: Field Programmable Gate Array

Se cuenta con la Tarjeta de Desarrollo ML507, que presenta un FPGA Virtex5, de Xilinx.

En una primera fase se modela el sistema con los módulos IP correspondientes al sistema MicroBlaze con sus periféricos. Se dispone un sistema capaz de satisfacer los requerimientos de hardware que Petalinux requiere.

Luego se compila el Petalinux, utilizando como sistema operativo huésped a Ubuntu 9.04. Por cuanto se realiza la compilación cruzada, es necesario la correcta configuración de las variables de entorno para el compilador de Petalinux.

Se comprueba el funcionamiento del sistema Petalinux-MicroBlaze. Se añade a la red local, se le realizan pruebas como análisis de trazas de sus paquetes en TCP/IP.

Se busca las fuentes de un stack de protocolos VoIP libre en internet. Se comprueba el funcionamiento de las fuentes de los protocolos descargados compilándose dentro del entorno y con el compilador de Ubuntu. Se instala un servidor de telefonía IP, Asterisk3, como controlador de las llamadas dentro de la red LAN4. Se verifica una correcta comunicación con el servidor Asterisk y con los demás clientes VoIP convencionales, como teléfonos IP, softphones.

Se configura los ficheros "makefile"⁵ de los fuentes del Stack para adaptarlos al entorno y al compilador de Petalinux. Se aplican las variables de entorno específicas para la compilación cruzada desde Ubuntu 9.04. Se compila el stack VoIP.

Se introduce el ejecutable junto con los ficheros de configuración necesarios dentro del directorio que Petalinux dispondrá sus ejecutables cuando se compile su imagen. Se recompila parte del kernel de Petalinux y se generan los archivos imágenes con el Stack incluido.

De esta forma la Tarjeta de Desarrollo presenta un Linux embebido en su FPGA, con posibilidad de transferencia de voz sobre IP por su puerto de red, como un cliente más del sistema de la red telefónica IP de Asterisk.

MicroBlaze: Características Generales

MicroBlaze comparte la arquitectura tipo RISC, donde el limitado número de instrucciones permite simplificar la unidad de decodificación. El número de instrucciones que permite no es mucho más de 87, además de que cada instrucción se ha elegido para que el tamaño de la unidad lógica aritmética (Arithmetic and Logic Unit, ALU) sea también reducido. Aquellas instrucciones que requieran un procesamiento complejo deberán realizarse en un hardware específico, diseñándose sobre los restantes recursos del FPGA [2].

Este microprocesador (Figura 1) usa el formato Big Indian para representar los datos. Los tipos de datos soportados son palabras de 32 bits, palabra de 16 bit, y bytes. Todas las instrucciones son de 32 bits; de igual tamaño cuenta con 32

3 Asterisk: Aplicación de código abierto para telefonía IP.

4 LAN: Local Area Network

5 Fichero makefile: Usados para describir y efectuar la compilación automática de los ficheros fuentes.

registros de propósito general, y de 18 registros de propósitos especiales. Igualmente cuenta con servicios de reset e interrupciones.

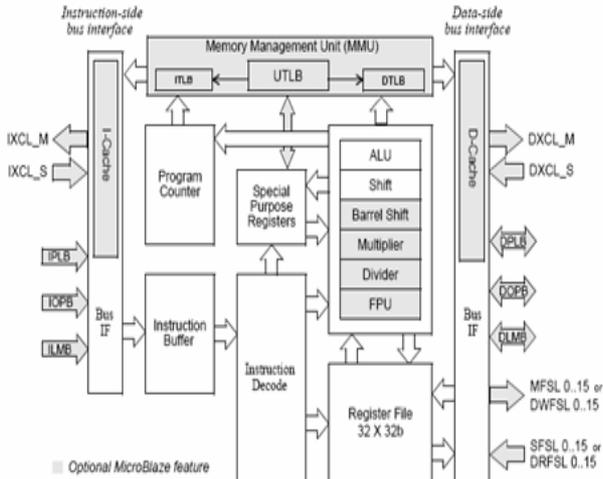


Figura 1 Diagrama en bloque del núcleo MicroBlaze[2].

La ejecución de las instrucciones tiene la forma “pipelined”, donde cada etapa de ejecución requiere 1 ciclo de reloj. Su arquitectura de memoria responde al diseño Harvard, donde el acceso a las instrucciones y a los datos se encuentran separados [3]. El procesador cuenta con 3 interfaces para el acceso a memoria:

- LMB (Local Memory Bus)
- PLB (Processor Local Bus) o OPB (On-Chip Peripheral Bus)
- XCL (Xilinx Cache Link)

Petalinux

Petalinux es el fruto del trabajo liderado por Petalogix, organización establecida en 2005 liderada por el Dr John Williams, en la Universidad de Queensland, Australia, para el establecimiento de µLinux en el microprocesador MicroBlaze de Xilinx.

Actualmente posee integración de las versiones 2.4x y 2.6x del kernel de Linux. Entre los ficheros fuentes de Petalinux aparecen scripts⁶ ofrecidos por sus desarrolladores para adaptar las aplicaciones de los usuarios al kernel, ofreciendo facilidades para la introducción de programas y módulos específicos. De esta manera, se puede implementar drivers para el manejo de módulos hardware que se desarrollen en los recursos libres del FPGA. Así se puede tener un diseño híbrido entre hardware que se responsabilice con tareas que requieran

6 Scripts: ficheros usados para describir la configuración de diversas aplicaciones.

de rapidez en su procesamiento, y de un sistema operativo que brinde inteligencia y control mediante aplicaciones que por su contenido puedan implementarse a partir de lenguajes de alto nivel, C o C++ entre otros.

Algunas características de Petalinux [4]:

- Integración con las herramientas ISE/EDK de Xilinx. Brinda la posibilidad de combinar ficheros del sistema con ficheros de Xilinx.
- Posibilidad de seleccionar y modular el kernel para aplicaciones específicas.
- Integración de U-boot como bootloader
- Ejemplos de sistemas predefinidos para algunas de las tarjetas de desarrollo de Xilinx.
- Soporta aplicaciones en C++.
- Librerías estándares disponibles.
- Posibilidad del kernel 2.6.31 suportando MMU
- Compatibilidad con muchos de los módulos IP de Xilinx

Modulos del Sistema MicroBlaze

Se utiliza la herramienta software Xilinx ISE/EDK 11.3, donde se dispone de módulos provistos por dicho sistema que permiten la concepción del sistema de microprocesador. Se prueba mediante un Kit de Desarrollo, ML507, de la familia de Virtex5 de Xilinx.

A continuación se describe los módulos que se utilizaron embebidos en el FPGA Virtex5.

- MicroBlaze v 7.20c. Microprocesador del sistema y elemento principal.
 - Módulo para el controlador de la memoria FLASH de la tarjeta ML507. La memoria se utiliza para guardar la imagen de Petalinux.
 - Controlador Ethernet físico dentro del FPGA. Capacidad de protocolos de red para el sistema.
 - Módulos para atención del puerto RS232. Se utiliza para la interfaz con el sistema operativo, así como para descargarle los ficheros imagen.
 - Módulo para memoria DDR-2 de la tarjeta ML507. En la memoria se copiará los datos del sistema operativo y se trabajará sobre esta.
 - Módulo para el controlador de interrupciones.
 - Módulo para el temporizador.
 - Módulos para memoria EPROM. Se utiliza para grabar la imagen que configura al FPGA con el sistema MicroBlaze.
 - Botones, Switches. Utilizados para reiniciar el sistema
- Se establece 100 MHz como reloj del sistema. Además se configura el módulo MicroBlaze con MMU. De esta forma brinda la posibilidad de ventajas como traducción de direcciones virtuales a direcciones físicas, protección de memoria, reducción de fragmentación [5].

Con EDK 11.3, se generan las librerías y demás ficheros necesarios para la compilación de cada uno de los módulos, y la imagen final para descargarla al FPGA.

Compilacion de Petalinux como Sistema Operativo utilizado

Para la preparación de Petalinux, se escoge a Ubuntu 9.04 como sistema host para la compilación del kernel del Linux embebido.

La Figura 2 muestra el árbol de directorio para trabajar con Petalinux.

```

PETALINUX
+ TOOLS
| + COMMON
| | + PETALOGIX
| | + BIN
| + LINUX-I386
| + MICROBLAZE-UCLINUX-TOOLS
|
+ SOFTWARE
| + PETALINUX-DIST
| + LINUX-2.6.X-PETALOGIX
| + UCLINUX-2.4.X
| + USER-APPS
| + USER-MODULES
|
+ HARDWARE
  + REFERENCE-DESIGNS
  + USER-PLATFORMS
  + EDK_USER_REPOSITORY
  
```

Figura 2 Arbol de los directorios fuentes de Petalinux

Dentro del directorio “Hardware” se ubica los ficheros de referencias de tarjetas definidas por Xilinx.

Igualmente para cada proyecto embebido que se realice por el desarrollador se debe situar los ficheros dentro de este directorio.

En “Software” se encuentra todo lo referente a librerías de Linux, módulos del kernel, entre otros. Igualmente esta localización se dispone para albergar las aplicaciones específicas de usuarios compiladas con el compilador de Petalinux.

Dentro de “Tool” se encuentra todo lo referente a ficheros del sistema, compiladores a utilizar, entre otros.

Para la compilación cruzada del Petalinux, se aplica las variables de entorno de Ubuntu que validan los scripts para el manejo de las opciones de compilación.

Se utiliza herramientas de compilación provistas por los desarrolladores de Petalinux para escoger los drivers y aplicaciones que el sistema operativo albergará.

Los archivos generados se descargan al sistema embebido mediante el puerto RS232 de la tarjeta ML507.

A continuación se muestra los archivos que son compilados:

- image.bin

- image.elf
- image.srec
- image.ub
- Linux.bin
- Roms.img
- Rootfs.cpio
- Ub.config.img
- U-boot.bin
- U-boot.srec
- U-boot-s.srec

Con la utilización de un servidor tftp, se puede almacenar los ficheros anteriores. Después de configurar el FPGA con el sistema embebido definido, a través de TCP/IP la tarjeta adquiere los ficheros desde dicho servidor. La interfaz del Petalinux con el usuario se lleva a cabo a través de consola por el puerto RS232 utilizando cualquier software que maneje el puerto serie, como Hyperterminal por ejemplo.

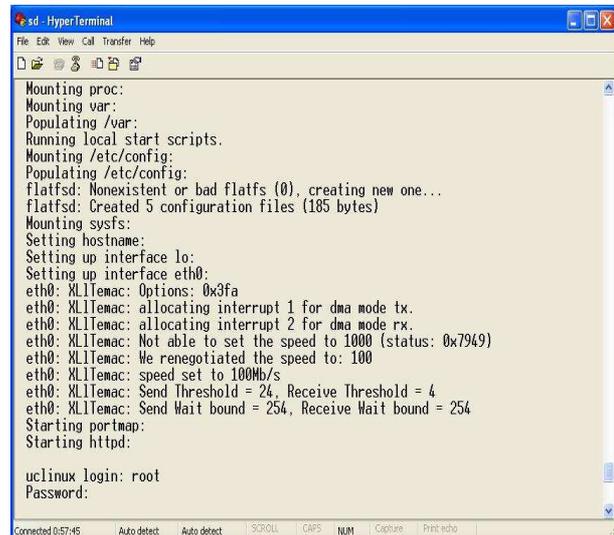


Figura 3 Arranque e inicio de Petalinux

Otros procesos para levantar el sistema operativo es la llamada a U-boot (Universal Bootloader). Es el proyecto software que despierta la imagen del sistema operativo. En este caso es construido junto con el kernel. A través de U-boot se le configura las direcciones IP del servidor tftp y del controlador Ethernet de la Tarjeta de Desarrollo, para el acceso de los ficheros imagen que se encuentran en el servidor tftp.

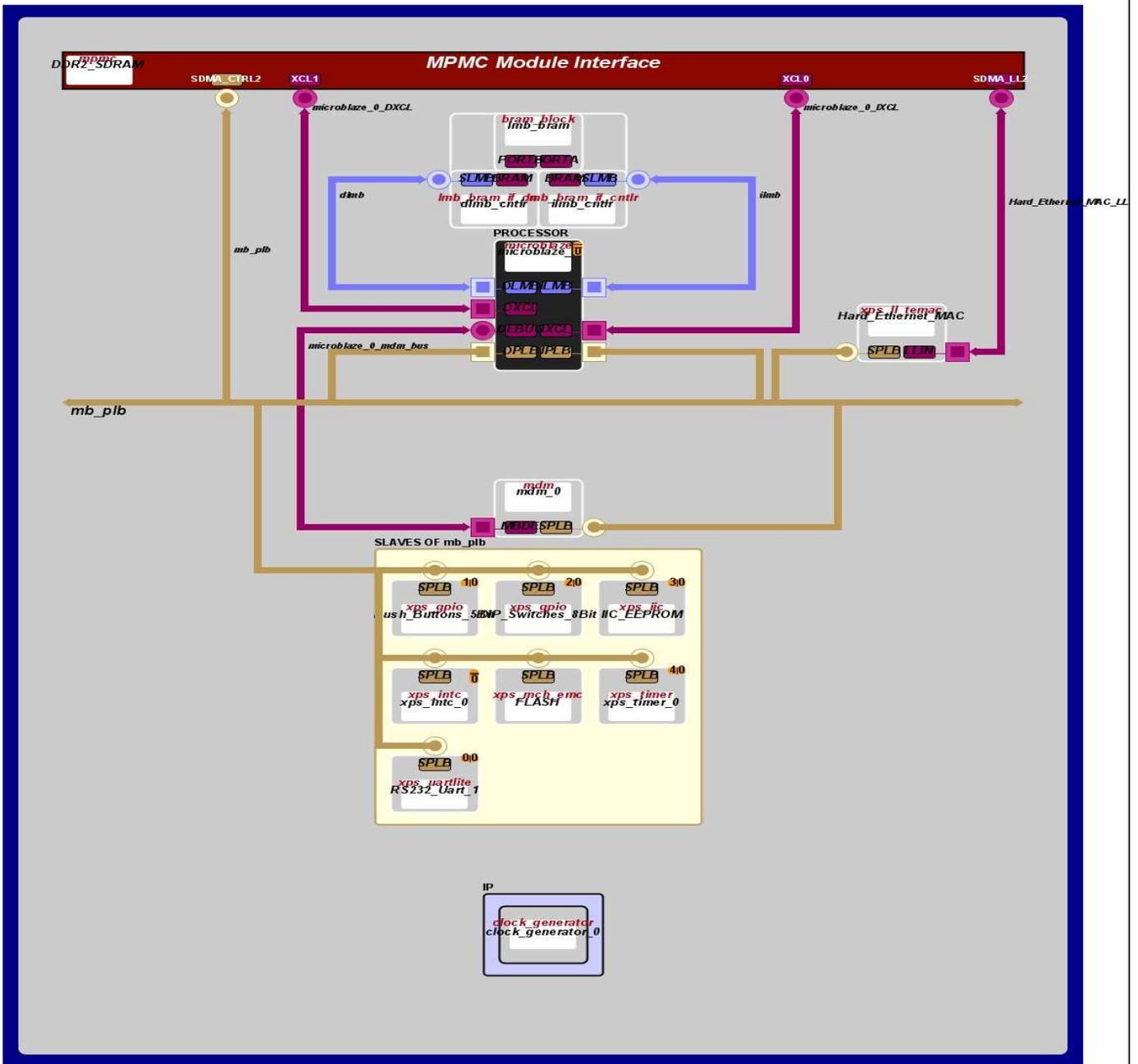
Una vez que las variables de entorno se configuren correctamente entonces U-Boot comienza el arranque de la imagen de Petalinux.

En la Figura 3 se muestra parte del proceso de arranque de Petalinux en el FPGA. Se accede mediante Hyperterminal.

Protocolos VoIP. Compilación e inserción dentro de Petalinux.

Algunos protocolos importantes en la comunicación de VoIP son el protocolo SIP y protocolo RTP. SIP es un protocolo de

señalización, y presenta generalmente una arquitectura cliente-servidor [6]. Generalmente se encapsula sobre UDP, aunque también en algunos casos se implementa sobre TCP.



SPECS	
EDK VERSION	11.3
ARCH	virtex5
PART	xc5vfx70tff1136-1
GENERATED	Fri Oct 29 16:30:02 2010

KEY			
SYMBOLS			
	Bus connections	External Ports	Interrupts
	master or initiator	input	Interrupt Controller
	slave or target	output	Interrupt Target
	master slave	inout	Xilinx Interrupt Sources
	monitor		<i>X = Controller ID</i> <i>Y = Interrupt Priority</i>
COLORS			
Bus Standard			
DCR	FSL	OPB	SOCM
FCB	LMB	PLB	USER P2P
			Xilinx P2P

Figura 4. Diagrama en bloque de los módulos empotrados en Virtex5 para Petalinux.

Para el transporte de la voz se utiliza generalmente RTP. Este protocolo es mayormente soportado por UDP, debido a la exigencia de demora que este demanda.

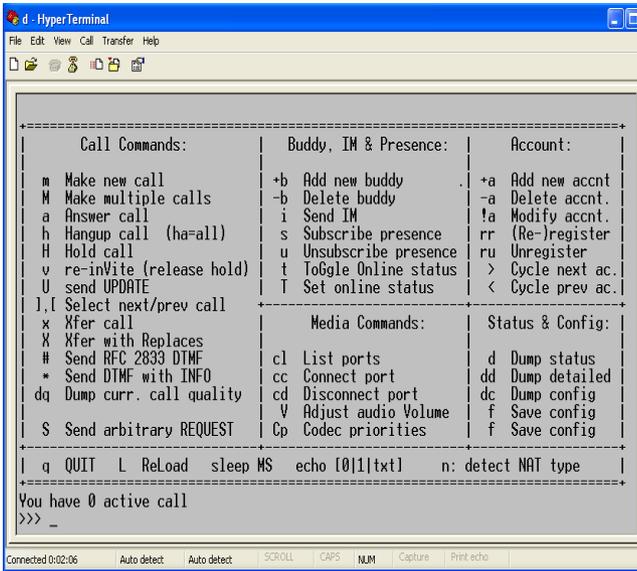


Figura 5 Softphone dentro del FPGA

En este diseño se utiliza un stack7 SIP-RTP con código abierto de manera libre en internet.

Los ficheros y librerías se encuentran desarrollados en el lenguaje C, y dispone de archivos “makefile” para su configuración sobre los distintos escenarios en que se desenvolverá.

Básicamente se puede comprobar la efectividad del protocolo utilizando el compilador GCC y el entorno de Ubuntu 9.04. Se configura el Stack y se pueden realizar llamadas vía IP a través de un Servidor Asterisk, por ejemplo.

Se modifican ficheros “makefiles” que establecen los parámetros de compilación automática de librerías y ficheros fuentes del proyecto. En este caso se le introduce a los ficheros

la utilización del compilador de Petalinux, así como localizaciones de librerías dinámicas, estáticas, y banderas a tener en cuenta para el entorno de Petalinux.

Las características de compilación dentro de Petalinux varían si el sistema hardware que lo soporta presenta MMU. En este caso se utiliza el compilador para sistemas MMU de Petalinux. De esta manera se compilan todas las librerías que este stack presenta, obteniéndose el fichero ejecutable para el entorno de Petalinux. Este fichero se coloca junto con los demás ejecutables que Petalinux mostrará como sistema operativo cuando actúe dentro del FPGA.

Con la ayuda de una aplicación de muestra, compilada también junto las librerías del stack, se comprueba la efectividad de los protocolos. Esta aplicación actúa como softphone, utilizando los protocolos SIP y RTP compilados.

Dentro del árbol de directorio de Petalinux se le agrega un archivo de configuración, utilizado por el softphone para su trabajo con la red real en que trabajará.

A través de un servidor de telefonía IP, Asterisk, por ejemplo, se puede comprobar el trabajo del softphone dentro de la tarjeta de Xilinx y otros softphone convencionales, así como un teléfono IP.

En la **¡Error! No se encuentra el origen de la referencia. 5** se muestra la interfaz de la aplicación que utiliza los protocolos SIP y RTP, todo embebido dentro del FPGA. Existe la posibilidad de establecimiento, recepción y administración de las llamadas.

En la **Figura** se muestra una traza de comunicación VoIP. A través del software analizador de red Wireshark se recoge una traza del registro del softphone que corre dentro del FPGA, en la Tarjeta de Desarrollo.

Se puede divisar el intercambio satisfactorio de una llamada IP (mensajes SIP y RTP) entre la tarjeta, el servidor Asterisk, y un teléfono IP.

No obstante los mensajes RTP mostrados aún no codifican la señal real de voz. En la etapa de confección de este documento

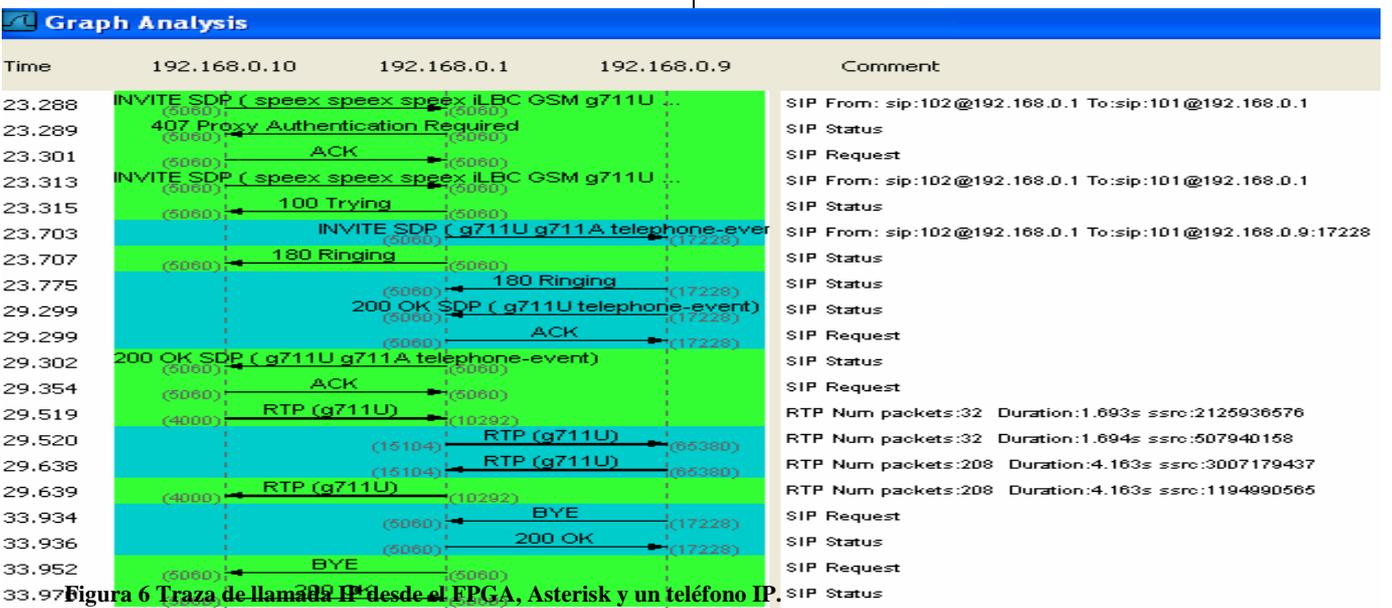


Figura 6 Traza de llamada IP desde el FPGA, Asterisk y un teléfono IP.

se desarrolla la utilización de módulos de sonido en la tarjeta ML507, su integración al sistema MicroBlaze y sus respectivos drivers y aplicaciones para Petalinux.

CONCLUSIONES

1. El sistema embebido sobre FPGA utilizando MicroBlaze puede ser utilizado para el soporte de sistemas operativos con kernel estándares como μ Linux.
2. Petalinux, basado en μ Linux para sistemas con MicroBlaze, puede ser el controlador del sistema y ser soportado por dicho hardware. Ofrece gran parte de librerías estándares para la implementación de aplicaciones y drivers que permite flexibilidad al sistema.
3. Varios protocolos de código libre en internet pueden ser utilizado como aplicaciones específicas dentro de los llamados Sistemas Operativos Embebidos, como Petalinux. De esta forma se obtiene un sistema basado en FPGA capaz de soportar e intercomunicarse con protocolos relativamente complejos.
4. El sistema MicroBlaze, Petalinux y Stack VoIP puede ser la base de alguna plataforma de telefonía IP, dígase Gateways, Teléfono IP, entre otros.

REFERENCIAS

1. **CABRERA SARMIENTO, ALEJANDRO:** “Transferencia de tecnologías de diseño digital avanzado” en <http://www.redciencia.info.ve/memorias/ProyProsp/trabajos/t5.doc>.
2. **XILINX.** “MicroBlaze Processor Reference Guide”.2008 en http://www.xilinx.com/support/documentation/sw_manuals/mb_ref_guide.pdf
3. **PETALOGIX** [En línea]. Disponible en: <http://www.petalogix.com/products/petalinux>
4. **PELLERIN D, THIBAUDL.:** *Practical FPGA Programming in C*. Ed. Prentice Hall, 2002.

AUTOR

Orlando Landrove Gámez, Ingeniero en Telecomunicaciones y Electrónica. Graduado en 2007. Aspirante a Investigador y Profesor Instructor en la CUJAE, Facultad de Eléctrica. Labora en **LACETEL**, Instituto de Investigación y Desarrollo de Telecomunicaciones. Actualmente realiza investigaciones acerca de las Redes de Próxima Generación y los microprocesadores empotrados en FPGA.