

“Implementación de un Protocolo Básico de Comunicación (BaCoP) sobre Redes TCP/IP dedicado al Control en MicroBlaze”

MSc. Abel Lorenzo Morera¹, Dr. Wilfredo Falcón Urquiaga²,

Dr. José Raúl Vento Álvarez³, Ing. Felix Carlos Pérez Ruizcalderón⁴

1 ATI, Tecnologías de la Información y Automática (UNE). Ciudad de la Habana, Cuba.

abel@ati.une.cu Tel: +537 8360574

2 Dpto. de Telecomunicaciones. Facultad de Informática y Telecomunicaciones. Universidad de Pinar del Río, Cuba.

falcon@tele.upr.edu.cu Tel: +5348 755453

3 Dpto. de Telecomunicaciones. Facultad de Informática y Telecomunicaciones. Universidad de Pinar del Río, Cuba.

vento@tele.upr.edu.cu Tel: +5348 779359

4 Departamento de Investigación y Desarrollo, Centro de Desarrollo de la Electrónica y la Automática. Pinar del Río, Cuba.

fcarlos@cdea.co.cu

RESUMEN / ABSTRACT

En el presente trabajo se describen las tecnologías y procedimientos necesarios para el diseño e implementación de un protocolo de comunicación para control de recursos de hardware en dispositivos embebidos.

Con este propósito se muestra la perspectiva de implementar sistemas de comunicación sobre hardware reconfigurable, haciendo uso de FPGAs, específicamente el kit de desarrollo Spartan 3E, el procesador embebido MicroBlaze, el módulo IP Ethernet Lite para el control de acceso al medio físico; y las librerías de procedimientos de software lwIP y Xilkernel para atender el paquete de protocolos TCP/IP y un pequeño sistema operativo multitarea que gestione las conexiones concurrentes respectivamente.

Utilizando los recursos anteriores se implementó el Protocolo Básico de Comunicación (BaCoP) descrito en el presente documento.

Palabras Clave: Sistemas embebidos, Redes TCP/IP, Xilinx Spartan 3E, MicroBlaze, Ingeniería de Protocolos.

In this work the necessary technologies and procedures for designing and implementing a hardware resources control communication protocol, on embedded devices, are described. This research has taken place at the University of Pinar del Río “Hermanos Saíz Montes de Oca”.

With this purpose it has been shown the perspective of implementing communication systems on reconfigurable hardware, specifically discussing the use of FPGAs. In particular the Xilinx Spartan 3E Starter Kit; the MicroBlaze embedded processor; the Ethernet Lite IP module, for medium access control; and the lwIP and Xilkernel software procedure libraries have been taken into consideration.

Using these resources the Basic Communication Protocol (BaCoP), described in this document, has been implemented.

KeyWords: Embedded Systems, TCP/IP networks, Xilinx Spartan 3E, MicroBlaze, Protocol Engineering.

Implementation of a Basic Communication Protocol over TCP/IP dedicated to Control in Microblaze

INTRODUCCIÓN

La expansión creciente del uso de las redes Ethernet y TCP/IP en entornos de área local ha potenciado que cada vez más esfuerzos estén dirigidos a la interconexión de servicios y todo ello ha obligado a la incorporación de soluciones telemáticas que permitan interconectar aplicaciones de control a la red, incorporándolas al mundo online.

Ya no es ciencia ficción oír hablar de refrigeradores, hornos microondas o cámaras de vigilancia conectados directamente a la Internet. Todos estos son equipos electrodomésticos con implementaciones de Circuitos Integrados en su interior, capaces de comunicarse con la red y de prestar servicios específicos en función de su diseño.

Buscando cómo se han solucionado estos problemas en el mundo, se ha constatado que la forma empleada para lograr esta integración es el diseño de protocolos específicos de control, o en su defecto aplicaciones basadas en estándares de comunicación sobre redes TCP/IP ya consolidados, como es el caso de la comunicación web. [1]

Para el diseño de protocolos (o Ingeniería de Protocolos) existen diferentes metodologías, que en esencia buscan hacer verificar y validar así como describir las funciones y características básicas del protocolo con el fin de facilitar su futura implementación, eliminando estados de los que no se sale, garantizando que todos los estados se alcancen, eliminando ambigüedades y previendo todas las posibilidades [2]. Entre las más conocidas se encuentran las Máquinas de Estado Finitos y las Redes de Petri. Las primeras se utilizan básicamente en sistemas de parada y espera, con transiciones condicionadas a acciones independientes del tiempo y sin periodicidad, mientras que las segundas están más relacionadas con el disparo de temporizadores.

De tal modo, una vez diseñado el protocolo entonces es posible programarlo en una plataforma habilitada con comunicación a una red, permitiéndole de este modo interactuar con el resto de los dispositivos conectados a ella. Si el objetivo del protocolo diseñado fuera entonces el control del hardware de la plataforma se estaría en presencia de un protocolo de comunicación, muy útil en aplicaciones domóticas o inmóticas. [3]

Por otro lado, la posibilidad de incorporar comunicación TCP/IP a sistemas de hardware reconfigurable, permitiendo acortar la fase de diseño y multiplicando la variedad de soluciones a desarrollarse, ofrece un espectro extraordinario de aplicaciones que sólo la imaginación y creatividad del hombre serán capaces de llenar. [4]

Hasta el momento en que se inició esta investigación no se tenían antecedentes de la utilización de estos kits en la UPR para diseños con comunicación TCP/IP, y es por ello que la inexistencia de un protocolo específico de comunicación en Redes TCP/IP para control con MicroBlaze, que permita el control remoto de los recursos de hardware de dicha plataforma, constituye el problema científico a tratar en esta investigación, a partir del cual se trazó como objetivo de la investigación: la implementación de un protocolo específico de

comunicación en Redes TCP/IP para control con el procesador software MicroBlaze. [5]

Para darle solución al problema expuesto se plantea la siguiente hipótesis:

Sobre la base de:

- Los conocimientos existentes sobre Ingeniería de Protocolos en el diseño de protocolos de comunicación en redes.
- Todos los conocimientos existentes en el campo de la telemática actual. Además teniendo en cuenta las posibilidades actuales de la creación de herramientas informáticas eficientes, así como de software orientados a complejos procesos.
- Que existe experiencia suficiente de trabajo con FPGA en la UPR, específicamente con el kit de desarrollo Spartan 3E-500 Revisión C.
- Que estos kit de desarrollo tienen incorporado una interfaz RJ45 y un circuito integrado capaz de realizar las funciones de control de la interfaz de red del modelo TCP/IP.

Suponiendo:

- El trabajo con una red local TCP/IP, orientado a conexión y confiable a nivel de transporte.
- Que los recursos de hardware del kit de desarrollo Spartan 3E-500 son suficientes para permitir la ejecución de la torre de protocolos TCP/IP, incluyendo protocolos de nivel de aplicación.
- El proceso de diseño de un protocolo simple de parada y espera.

Aplicando:

- El análisis de la arquitectura del procesador embebido MicroBlaze.
- La verificación del módulo de propiedad intelectual (IP core) Ethernet Mac Lite de Xilinx, para el manejo de la interfaz física, el control de acceso al medio (MAC) y la gestión del enlace lógico (LLC).
- La implementación de las funciones de la biblioteca de procedimientos de red lwIP (Light Weight Internet Protocol) para la implementación de la torre de protocolos TCP/IP, incluyendo ARP, ICMP y UDP; así como el sistema operativo multitarea Xilkernel.
- La Máquina de Estados Finitos como definición formal del Protocolo Básico de Comunicación (BaCoP).

Se puede afirmar:

- Que los resultados derivados de la integración de estas herramientas ofrecerán, junto a la definición formal de una máquina de estados finitos, la posibilidad de gestionar remotamente los recursos hardware del kit de desarrollo Spartan 3E-500 Revisión C.

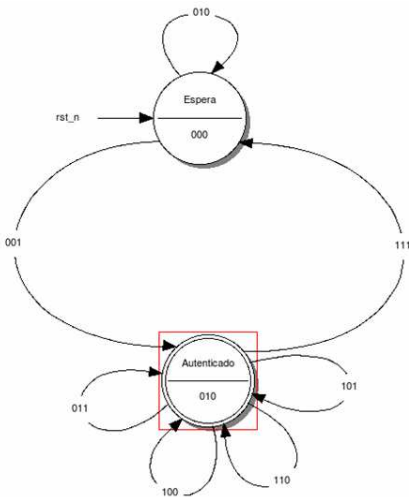


Figura 1
Diagrama de Transición del Autómata.

Como se aprecia en la figura 1, se definen 3 Salidas de Moore pues así se logran establecer las 7 salidas posibles $\Sigma = \{001, 010, 011, 100, 101, 110, 111\}$.

Ya se ha comentado anteriormente la razón de utilizar 3 bits para la representación, corriendo el riesgo de perder nivel de Completamiento (Completeness), tal y como se muestra en la figura 2, haciendo referencia a uno de los parámetros que se ofrecen como resultado del chequeo de integridad de la máquina de estados finitos en Qfsm, pero ganando en claridad sobre el funcionamiento general del protocolo. Este término se refiere al uso de cada una de las combinaciones binarias posibles, dado el número de salidas de Moore para representar los estímulos en cada uno de los estados. El resto de esos parámetros, como se puede apreciar, muestran resultados óptimos de diseño del protocolo.

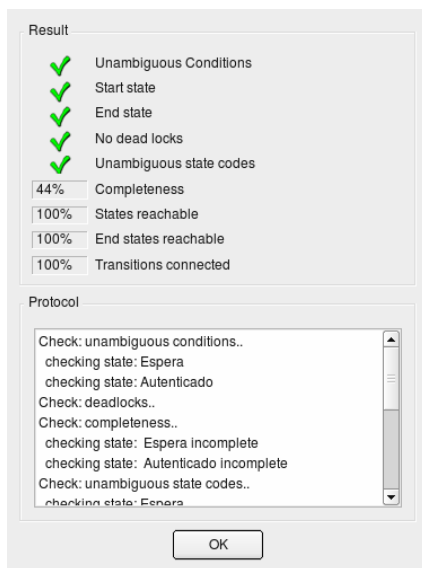


Figura 2
Chequeo de Integridad.

Luego de realizar el chequeo de integridad, la máquina de estados finitos está lista para su simulación que se realiza mediante la interfaz mostrada en la figura 3.

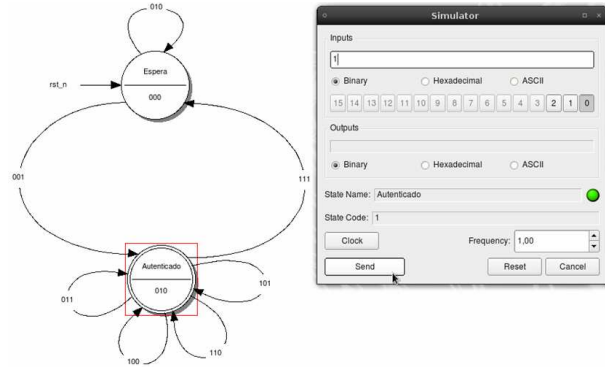


Figura 3
Simulación de la Máquina de Estados Finitos.

Aquí se introducen las posibles entradas al sistema en el apartado Input. Específicamente en la figura anterior se aprecia el resultado de haber introducido el estímulo “001” en el estado “Espera”, observándose en el diagrama de transición de la máquina de estados finitos que ha ocurrido la transición esperada.

Implementación práctica del protocolo BaCoP en Microblaze.

Una vez que se tiene diseñada, modelada y validada la máquina de estados finitos del protocolo en cuestión, es posible entonces pasar a la fase de implementación, llevando a lenguaje C++ las rutinas que posibiliten la comunicación en función de la aplicación que se esté implementando el protocolo.

En el caso particular de la aplicación implementada en el marco de esta tesis sólo se ha tenido en cuenta la activación y la desactivación de los leds presentes en la arquitectura de la Spartan 3E-500.

Sin embargo lo que se ha pretendido es crear las bases para futuras aplicaciones dando además las posibilidades de extender el uso del protocolo BaCoP, así como tomar este proyecto como un acercamiento al mundo de la Ingeniería de Protocolos, en el que Cuba apenas comienza a dar sus primeros pasos.

En la implementación práctica del protocolo se ha apelado fundamentalmente a las funciones read y write, para el intercambio de información entre el cliente y el servidor, cuyas respectivas sintaxis se muestran a continuación:

```
read(sd, receiveBuffer, RECV_BUFFER_LENGTH);
write(sd, sendBuffer, strlen(sendBuffer));
```

donde además ha sido necesario definir las variables: sd, identificador de socket; receiveBuffer, buffer de recepción; sendBuffer, buffer de transmisión y las longitudes de los arreglos de caracteres definidos por estos dos últimos.

Ha sido necesario incluir las bibliotecas de procedimientos, anteriormente mostradas, que permitirán utilizar las distintas funciones que el control del hardware de la tarjeta, el tratamiento de las cadenas de texto para detectar los comandos y la atención a los recursos de redes. A continuación se muestra el segmento de código que habilita las bibliotecas en cuestión:

```
/* Xilinx Includes */
#include "xparameters.h"
#include "xemaclite.h"
#include "xgpio_1.h"
#include "xutil.h"
#include "xstatus.h"
#include "xmk.h"
/* C Library Includes */
#include "string.h"
#include "stdlib.h"
/* lwIP Includes */
#include "netif/xemacliteif.h"
#include "lwip/tcpip.h"
#include "lwip/memp.h"
#include "netif/etharp.h"
#include "lwip/sys.h"
#include "lwip/sockets.h"
```

También es posible, a través de una conexión RS232, comprobar las distintas etapas de la inicialización de los servicios de redes en el diseño (ver figura 4).

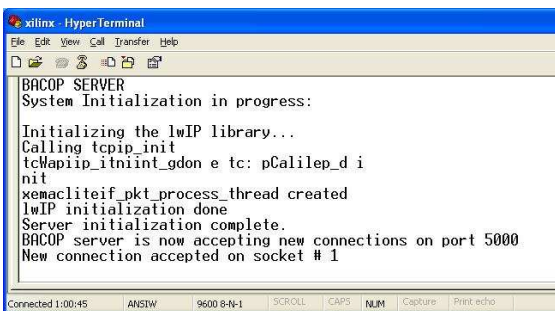


Figura 4
Inicio de servicio BaCoP en MicroBlaze.

Una vez establecida una nueva conexión el sistema comienza a procesar los comandos introducidos, como muestra la rutina siguiente, encargada de detectar la correcta recepción del comando “on n”, con $1 \leq n \leq 8$, que permite la activación del led identificado con el número n.

```
if(!strcmp(comandoPtr, "on")&& atoi(ledPtr)<=8 &&
    atoi(ledPtr)>=1)
{
    strcpy(char_buffer, cadena);
    comandPtr = strtok(char_buffer, " ");
    dataPtr = strtok(NULL, "\r");
    pos=atoi(dataPtr);
    set_state(pos, 1);
    activate_state();
    sprintf(sendBuffer, "Led %s en %s\r\nEntre
el comando:", dataPtr, comandPtr);
    write(sd,sendBuffer, strlen(sendBuffer));
}
```

La función activate_state(), desarrollada en el marco de este proyecto, es la encargada de invocar a la función XGpio_mSetDataReg() que es la que a su vez interactuará directamente con el hardware de la tarjeta.

Validación del protocolo.

Es preciso añadir que la implementación práctica del protocolo en este proyecto ha estado concentrada en el caso del servidor. Se ha utilizado un cliente Telnet (BaCoP utiliza TCP como protocolo de transporte y permite la transferencia de caracteres ASCII) con el que son perfectamente realizables las pruebas necesarias para la validación de la aplicación. En la figura 5 se muestra el resultado de las pruebas de comunicación mediante Telnet.

```
if (bytesReceived<=2)
{
    if(bytesReceived==1)
    {
        while (bytesReceived==1)
        {
            strcat(char_buffer, receiveBuffer);
            bytesReceived = read(sd, receiveBuffer,
RECV_BUFFER_LENGTH);
        }
        } else if(!strcmp(receiveBuffer, "\r\n"))
            strcat(char_buffer, receiveBuffer);
}
```

```

C:\WINDOWS\system32\cmd.exe
Servicio de Control de Leds con protocolo BaCoP. Bienvenido!
Entre el Comando: status
El estado del sistema es:

El led 1 est# encendido
El led 2 est# apagado
El led 3 est# apagado
El led 4 est# apagado
El led 5 est# apagado
El led 6 est# apagado
El led 7 est# apagado
El led 8 est# apagado
Entre el comando:Error de Sintaxis
Entre el comando:off 1
Led 1 en off
Entre el comando:quit
Saliendo del sistema...

Connection to host lost.
C:\Documents and Settings\Administrator>

```

Figura 5
Comunicación BaCoP a través de cliente Telnet.

Téngase en cuenta que con un cliente Telnet los caracteres son enviados uno a uno y ha sido por tanto necesario almacenarlos hasta recibir el comando totalmente. En la rutina de lenguaje C mostrada anteriormente se muestra cómo se almacenan los caracteres recibidos hasta que se recibe el carácter de retorno “\r\n”.

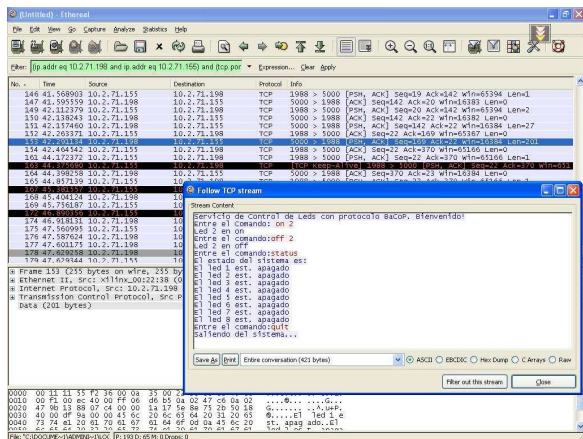


Figura 6
Captura de tramas con Ethereal filtrando el puerto 5000 en una comunicación con cliente Telnet.

Utilizando el software sniffer Ethereal se realizaron capturas de tramas durante la comunicación entre una PC y la tarjeta con el servicio BaCoP implementado. En la figura 6 se muestra el resultado del filtrado de dicha captura de tramas usando como parámetro el puerto 5000, por el cual el servidor instalado en MicroBlaze estará escuchando nuevas solicitudes de conexión.

Específicamente en este caso se ha diseñado el servidor para que responda a comandos que activen o desactiven los leds presentes en la Spartan 3E-500. En la siguiente figura se muestra un ejemplo del estado de los leds en el kit de entrenamiento y la respuesta del protocolo al realizar la solicitud con el comando correspondiente. Como muestra la figura 7 la comunicación a través de una red TCP/IP permitirá

que desde cualquier punto de la red se pueda acceder a los recursos de hardware de la tarjeta.

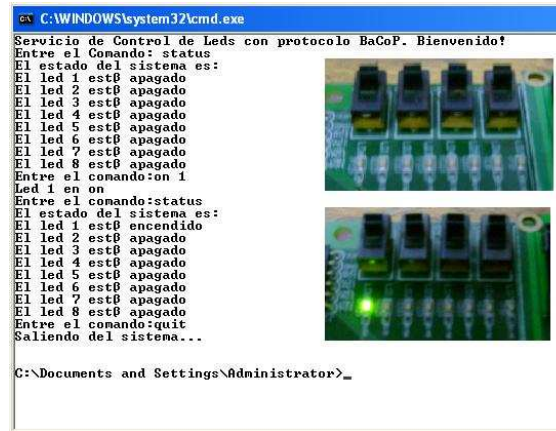


Figura 7
Ejemplo de aplicación utilizando BaCoP.

CONCLUSIONES

Se ha podido comprobar que es posible desarrollar la transmisión confiable de datos sobre redes TCP/IP utilizando el kit de desarrollo Spartan 3E 500 Starter Kit - revisión C - y el procesador embebido MicroBlaze.

Específicamente:

- Se utilizó el núcleo de propiedad intelectual Ethernet Lite para la manipulación de las tramas de datos en una LAN Ethernet.
- Se empleó la biblioteca de procedimientos de red lwIP en su interfaz Socket.
- Se implementó el Protocolo Básico de Comunicación (BaCoP).

Ante las limitaciones que implica la utilización de herramientas propietarias para modelado, Qfsm como software libre constituye una alternativa eficaz que logra implementar, modelar y simular el protocolo deseado.

AGRADECIMIENTOS

Esta investigación se llevó a cabo en el marco del proyecto ramal del MIC “Plataforma Cubana de Conmutación”, con la participación de la UPR, el ISPJAE y GKT, S.A entre otras instituciones.

A todos los compañeros involucrados en dicho esfuerzo transmitimos nuestro eterno agradecimiento.

REFERENCIAS

1. PÉREZ RUIZCALDERÓN, F. C. “Implementación de un Servidor Web sobre Hardware Reconfigurable”. Tesis de

Grado. 2009. Universidad Hermanos Saiz Montes de Oca. Pinar del Río. Cuba.

2. **MENGUAL GALÁN L.** “*Ingeniería De Protocolos*”. Universidad Politécnica de Madrid. 2007.

3. **TRUJILLO DÍAZ, A.; PIÑA RIVERA Y.** “*Diseño e Implementación de Protocolo de Aplicación para Innótica*”. Tesis de Grado. 2009. Universidad Hermanos Saiz Montes de Oca. Pinar del Río. Cuba.

4. **CABRERA SARMIENTO, A. J.** “*Sistemas empotrados y módulos IP*”. Dpto. de Automática y Computación, Instituto Superior Politécnico “José Antonio Echeverría” CUJAE.

5. **PORTILLA LÓPEZ VIGIL, B. M.; IGLESIAS MARTÍNEZ, M. E. Y FALCÓN URQUIAGA, W.** “*Implementación de controladores para periféricos del kit de desarrollo Spartan -3E de Xilinx*”. UPR. Mayo, 2008.

6. **COMER, D. E.** “*Redes globales de información con Internet y TCP/IP*”. Principios básicos, protocolos y arquitectura”. Tercera Edición, Prentice Hall.

7. **BAZILIO MARTINS, C.** “*Formal Analysis of Protocols and Distributed Algorithms: A Based-Language Approach*”. Tesis Doctoral. Universidad Católica de Rio de Janeiro. PUC-Rio. Brasil. 2005.

8. **GUTIÉRREZ OROZCO J. A.** “*Máquinas de Estados Finitos*”. Escuela Superior de Cómputo, 2008.

AUTORES

Abel Lorenzo Morera, graduado de Ingeniería en Telecomunicaciones y Electrónica. Es Máster en Ciencias de Sistemas de Telecomunicaciones y Profesor Asistente. Trabajó durante 8 años en la Universidad de Pinar del Río como profesor de Telemática. Actualmente trabaja en la Empresa Tecnologías de la Información y Automática (ATI) de la Unión Eléctrica (UNE) como especialista supervisor de proyectos de Telecomunicaciones.

abel@ati.une.cu

Felix Carlos Pérez Ruizcalderón, graduado en Ingeniería en Telecomunicaciones y Electrónica. Es Profesor Instructor. En estos momentos se desempeña como Investigador en el Centro de Desarrollo de la Electrónica y la Automática (CDEA), su trabajo se enfoca en el desarrollo de Sistemas Embebidos para el Monitoreo y Control Automático, empleando Microcontroladores y FPGAs así como la creación de las interfaces de gestión de los referidos sistemas utilizando los lenguajes Java, JavaScript, PHP y Bases de Datos del tipo PostgreSQL.

fcarlos@cdea.co.cu

José Raúl Vento Álvarez, graduado de Ingeniería en Telecomunicaciones. Es Doctor en Ciencias Técnicas y Profesor Auxiliar. Labora actualmente en la Universidad de Pinar del Río como Director de Informatización de la Universidad de Pinar del Río y Profesor del Departamento de

Telecomunicaciones de la Facultad de Informática y Telecomunicaciones de la Universidad de Pinar del Río.

vento@tele.upr.edu.cu

Wilfredo Falcón Urquiaga, graduado en Ingeniería en Telecomunicaciones y Electrónica. Es Doctor en Ciencias y Profesor Auxiliar. Trabaja actualmente en la Universidad de Pinar del Río como profesor en la carrera de Telecomunicaciones y Electrónica e imparte los temas Electrónica Digital. Se desempeña como Decano de la Facultad de Informática y Telecomunicaciones y es asociado joven de la Academia de Ciencias de Cuba desde el año 2007.

falcon@tele.upr.edu.cu