

## IMPLEMENTACION DEL PROTOCOLO DE ENRUTAMIENTO OSPF EN UN DISPOSITIVO FPGA

### RESUMEN

La implementación del protocolo de enrutamiento OSPF (primero la ruta libre mas corta) en el dispositivo lógico programable esta enmarcado dentro de la línea de investigación en telecomunicaciones del grupo SIRIUS. Uno de los objetivos principales al interior del grupo, es el diseño y ensamblaje completo de un enrutador a bajo costo y de fácil acceso para pequeñas y medianas empresas de la región. Para esto se ha dividido el diseño en módulos más pequeños que conservan una relación con el desarrollo de proyectos de pregrado, tales como cifrado, control de flujo y enrutamiento de datos.

Para el desarrollo de dicho proyecto se cuenta con la plataforma de desarrollo Xilinx Platform Studio, que permite utilizar el lenguaje de programación C++ , permitiendo así el manejo de los diferentes mensajes por medio de estructuras las cuales almacenan los diferentes campos para formar las tramas de los mensajes que maneja el protocolo OSPF.

**PALABRAS CLAVES:** enrutamiento, algoritmo Dijkstra, red, OSFP, FPGA.

### ABSTRACT

The implementation of the OSPF protocol (Open Shortest Path First) in the PLD its framed within the research field of in telecommunications of the SIRIUS group. One of the primary targets in the group is the design and complete assembly of low cost router and easy access for small and medium companies of the region. Therefor, the design has been divided in small modules but conserve a relation with the development of undergraduate projects, such as coding, flow control and routing of data.

For the development of the mentioned project we count with the platform of development Xilinx Platform Studio, which allows to use the C++ programming language to manage the different messages through structures that store the different fields to form the messages that the OSPF protocol handles.

**KEYWORDS:** Routing, Dijkstra algorithm, net, OSPF, FPGA

### 1. IMPLEMENTACION

La propuesta se enfoca en la realización del análisis, diseño e implementación del protocolo de enrutamiento OSPF en una red punto a punto, conformada por dos Spartan-3 conectadas por medio de un enlace serial. Estos dispositivos se encargaran de mostrar el dato enviado y así comprobar el enrutamiento.

La tarjeta Spartan-3 suministra una plataforma de bajo costo, fácil desarrollo y evaluación para todos los diseños.



Figura 1. Diagrama de conexión

Para el desarrollo del software se utilizó la plataforma de desarrollo Xilinx Platform Studio, el cual permite realizar la implementación con el lenguaje C++, y de esta manera poder realizar de forma estructurada todo el desarrollo del proyecto, lo que quiere decir que para el manejo de los diferentes paquetes que OSPF debe enviar se han manejado estructuras con cada uno de los campos necesarios para formar las tramas de dichos paquetes, como lo son: el paquete Hello, la descripción de la base de datos y la petición de estado-enlace.

### 2. DISEÑO DEL SOFTWARE

El objetivo del análisis del software será escribir correctamente una serie de reglas que usen adecuadamente dicho medio, definiendo cómo se codifican los mensajes, cómo se inicia o finaliza una comunicación, etc. Hay que evitar, por todos los medios,

### ANA MARIA LOPEZ

Ingeniera Electricista  
Docente  
Universidad Tecnológica de Pereira  
anamayi@utp.edu.co

### ANDREA BETANCOURT

Aspirante a Ingeniero de Sistemas y Computación  
Estudiante  
Universidad Tecnológica de Pereira  
andreabeta@utp.edu.co

### CATALINA ZULUAGA MUÑOZ

Aspirante a Ingeniero de Sistemas y Computación.  
Estudiante  
Universidad Tecnológica de Pereira  
catazuluaga@utp.edu.co

dos tipos de errores: el diseño de un conjunto de reglas incompleto y el diseño de reglas contradictorias.

Un protocolo proporciona comunicación entre dos máquinas, definiendo un medio físico, codificaciones de caracteres compatibles, transmisión y recepción de señales de los medios aproximadamente a la misma y velocidad.

El protocolo OSPF utiliza 5 estructuras de mensajes, los cuales le permiten tener constantemente la base de datos y la topología de red actualizada. Estos mensajes se tendrán en cuenta para el diseño posterior del protocolo.

- **Saludo** Se usa para identificar a los vecinos, es decir, los enrutadores adyacentes en un área para elegir un enrutador designado para una red multienvio.
- **Descripción de la base de datos** Durante la inicialización, se usa para intercambiar información de manera que un enrutador puede descubrir los datos que le faltan en la base de datos.
- **Petición del estado del enlace** Se usa para pedir datos que un enrutador se ha dado cuenta que le faltan en su base de datos o que están obsoletos.
- **Actualización del estado del enlace** Se usa como respuesta a los mensajes de petición del estado del enlace y también para informar dinámicamente de los cambios en la topología de la red.
- **ACK de estado del enlace** Se usa para confirmar la recepción de una actualización del estado del enlace. El emisor retransmitirá hasta que se confirme la recepción del mensaje.

Es importante observar que se necesita un canal bidireccional para intercambiar información y esto se lleva a cabo mediante el establecimiento de una adyacencia.

A continuación se explicara detalladamente los cambios en los estados de un enrutador para lograr la adyacencia total con un vecino y poder enrutar los mensajes mediante la ruta mas corta.

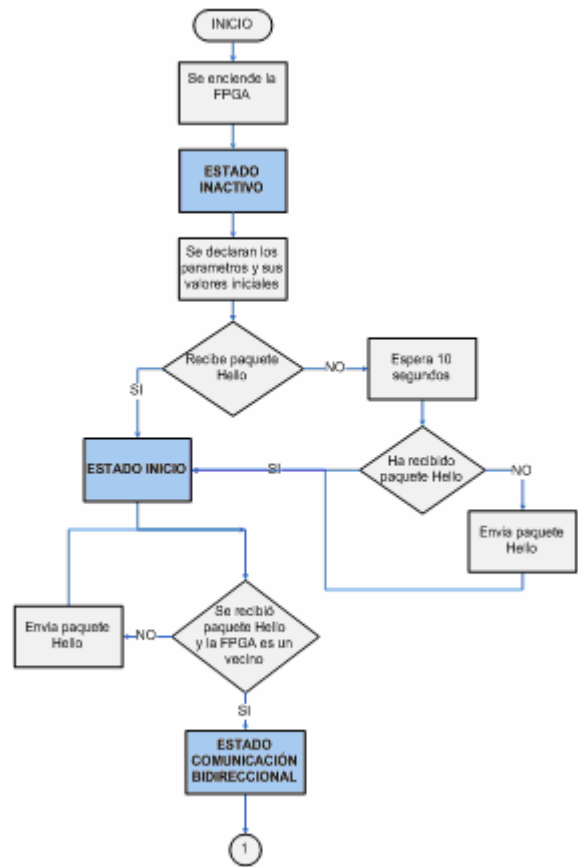


Figura 2. Cambios de estado del enrutador en el protocolo OSPF parte 1.

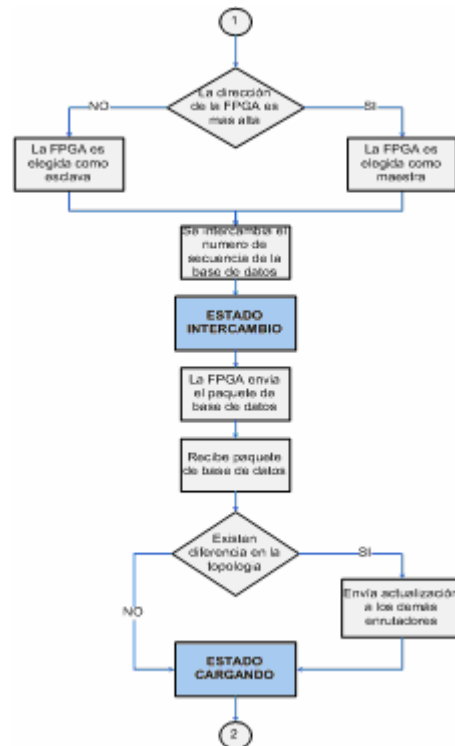


Figura 3. Cambios de estado del enrutador en el protocolo OSPF parte 2.

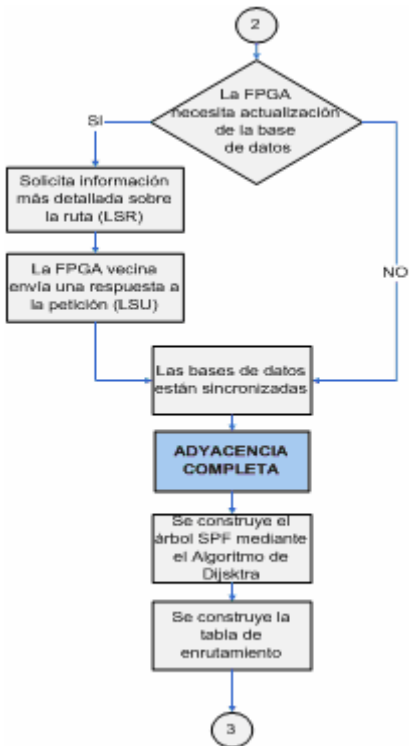


Figura 4. Cambios de estado del enrutador en el protocolo OSPF parte 3.

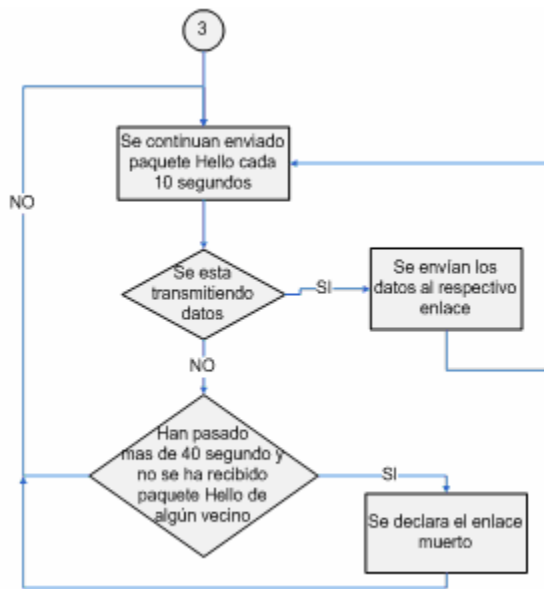


Figura 4. Cambios de estado del enrutador en el protocolo OSPF parte 4.

El diseño se realizó por medio de una máquina de estados que permite, de acuerdo al estado manejar los diferentes paquetes enviados por el protocolo OSPF.

En primer lugar nos encontramos en el estado donde se definen las diferentes variables y se les da su valor inicial. (estado INACTIVO) Además se espera la

recepción del paquete hello, sino hay ningún paquete en la cola de recepción se pasa al siguiente estado

En el segundo estado, estado de INICIO, si se ha recibido un paquete hello, se pasa al siguiente estado, sino se envía un mensaje hello y espera respuesta de sus vecinos.

En el tercer estado, estado COMUNICACIÓN BIDIRECCIONAL, al saber la fpga que tiene un vecino, empieza a comparar las direcciones IP para así elegir la esclava y la maestra, que permite definir el número de secuencia de cada una de las fpga y se pasa al siguiente estado.



Figura 5. Visualización de los estados en la fpga.

En el cuarto estado, estado INTERCAMBIO, se envía y recibe el paquete de la base de datos, si hay diferencias en la topología envía actualizaciones a sus vecinos, sino pasa al siguiente estado.

El quinto estado, CARGANDO, pregunta si la fpga necesita actualización de la base de datos, si la necesita se envían los mensajes de actualización del estado-enlace, y se dicen que las dos tarjetas están sincronizadas, pasando al siguiente estado

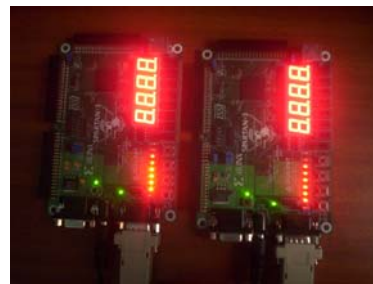


Figura 5. Visualización del estado de error en la fpga.

El estado de ADYACENCIA COMPLETA, se construye el árbol SPF mediante el algoritmo de Dijkstra, se construye la tabla de enrutamiento y se continúa enviando el paquete Hello cada 10 segundos sino hay respuesta de sus vecinos por 40 segundos o mas, se declara el enlace muerto.

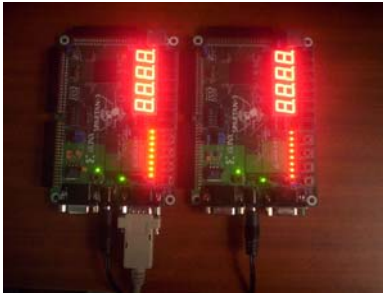


Figura 5. Visualización del estado de adyacencia completa en la fpga.

## 2. CONCLUSIONES

Se pudo trabajar en la capa física del modelo OSI manipulando mensajes a nivel de bits, ya que las tarjetas Spartan-3 contienen y manipulan tipos de datos por medio de bits.

El desarrollo de este prototipo permite el enrutamiento de datos, desde una tarjeta Spartan-3 a otra, por medio de un enlace serial.

El software utilizado para la realización de este proyecto Xilinx Platform Studio permitió desarrollar el prototipo del protocolo de enrutamiento OSPF de una forma simple con la ayuda del lenguaje c++, herramienta conocida y fácil de manejar.

Se fortaleció el manejo de la herramienta de desarrollo de Xilinx Platform Studio que permitió la programación a alto nivel de la tarjeta FPGA Spartan-3, además de la realización de la comunicación serial y ethernet de la tarjeta permitiendo el diseño de los paquetes necesarios para el correcto funcionamiento del protocolo OSPF.

Se utilizó la implementación de estructuras lógicas para el empaquetamiento de los datos enviados a través del enlace serial que interconecta las tarjetas FPGA Spartan-3.

Se consigue romper las barreras económicas para la consecución de tecnología que permita enrutar datos a través de una red por medio de un enlace serial.

Se agregó a la metodología de enrutamiento por medio del protocolo OSPF que permite encontrar la ruta más corta, entre dos dispositivos de red, en este caso entre dos FPGA.

El módulo NET1, no permitió la transmisión completa del paquete IP, ya que el microcontrolador UBICOM IP2022, no transmite el encabezado del paquete IP, el cual contiene la dirección IP que es la necesaria para enrutar el paquete de datos.

Al transmitir los datos por el puerto serial, se pudo ver que tanto la cola de recepción como de transmisión se llenaban, originando un cuello de botella que no permitía el paso de la información a través del puerto.

La herramienta de programación Xilinx Platform Studio, permite originar el código realizado en C++ en un lenguaje de bajo nivel, como lo es VHDL.

Se utilizó la programación estructurada la cual permitió manejar de forma más fácil el manejo y creación de los diferentes mensajes necesarios para la simulación del protocolo de enrutamiento OSPF.

## 3. RECOMENDACIONES

Realizar campañas de difusión del grupo de investigación y de sus proyectos, para que la comunidad educativa se comprometa con la investigación y desarrollo de nuevas líneas de investigación, donde los estudiantes mediante el conocimiento adquirido brinden a la sociedad la solución a nuevos problemas.

Para el desarrollo del proyecto fue necesario del apoyo de varios de los docentes vinculados al grupo de investigación, se recomienda entonces seguir brindando el apoyo a los estudiantes que así lo requieran, ya que es una buena estrategia para la puesta en marcha y realización de dichas investigaciones.

## 4. BIBLIOGRAFÍA

- [1] VISION GENERAL [en línea]. F. Consulta: 20050702. F. Actualización: 20040815. URL : <http://www ldc.usb.ve/~redes/Temas/Tema.21/vision.htm>
- [2] RFC 2328 OSPF [en línea]. F. Consulta: 20051012. URL: <http://www.ietf.org/rfc/rfc2328.txt>
- [3] Xilinx Inc., “User Core Templates Reference Guide”. [www.xilinx.com](http://www.xilinx.com). (Agosto 2003)
- [4] Xilinx Inc., “MicroBlaze Hardware Reference Guide”. [www.xilinx.com](http://www.xilinx.com). (Marzo 2002)
- [5] Xilinx Inc., “MicroBlaze Software Reference Guide”. [www.xilinx.com](http://www.xilinx.com). (Marzo 2002)
- [6] MAXINEZ, David G. y otros. El arte de programar sistemas digitales. CECSA. Mexico, 2004.
- [7] Xilinx Inc., “Embedded System Tool Guide”. [www.xilinx.com](http://www.xilinx.com). (Septiembre 2003).