Implementación HW/SW de Arquitecturas de Clasifiación de Paquetes Sobre Lógica Reconfigurable.

Jairo N. Trad y Luis R. Romano

Laboratorio de Comunicaciones Digitales Universidad Nacional de Córdoba, Facultad Ciencias Exactas, Físicas y Naturales

January 4, 2012





- Requerimientos
- Soluciones
- Problema marco
- Objetivos
- Sistema
 - Solución Propuesta
 - Descripcion funcional de cada bloque
 - Algoritmos de Clasificación
 - Modulo extractor de cabeceras
- 3 Arquitectura
 - Diagrama en bloques: Lineas E/S
- Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación









Requerimientos

- Soluciones
- Problema marco
- Objetivos



- Sistema
- Solución PropuestaDescripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras



- Diagrama en bloques: Lineas E/S
- Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación



- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Requerimientos de procesamiento en Redes

Características de Tráfico

- Las redes de datos crecen en Complejidad: nuevas aplicaciones, multimedia
- Las redes de datos crecen en Velocidad: n x 100Gbps (1Tbps@2015)
- Consolidación de múltiples servicios sobre redes Ethernet
- Redes Locales, Metropolitanas y Extensas utilizan Conmutación de Paquetes
- Adopción de tecnologías para virtualización en redes y servidores

Procesamiento de Paquetes

- Los enlaces ofrecen alta capacidad. El procesamiento de paquetes es crítico y debe optimizarse
- El rocesamiento a velocidad de línea
- Paquete Ethernet mínimo = 64bytes \rightarrow 6nanosegundos/paquete

Requerimientos de procesamiento en Redes

Características de Tráfico

- Las redes de datos crecen en Complejidad: nuevas aplicaciones, multimedia
- Las redes de datos crecen en Velocidad: n x 100Gbps (1Tbps@2015)
- Consolidación de múltiples servicios sobre redes Ethernet
- Redes Locales, Metropolitanas y Extensas utilizan Conmutación de Paquetes
- Adopción de tecnologías para virtualización en redes y servidores

Procesamiento de Paquetes

- Los enlaces ofrecen alta capacidad. El procesamiento de paquetes es crítico y debe optimizarse
- El rocesamiento a velocidad de línea
- Paquete Ethernet mínimo = 64bytes → 6nanosegundos/paquete



Motivación

- Requerimientos
 - Soluciones
- Problema marco
 - Objetivos



- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras
- Arquitectur
 - Diagrama en bloques: Lineas E/S
- Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación
- 5 Resultado
 - Introducción
 - Caso algoritmos unicamente
 - Caso loopback
 - Implementacion completa
 - Comparativa inter-algoritmos



Soluciones

Granularidad

- Actualmente → paquetes de longitud variable
- Peor caso → mínima longitud (64 bytes en Ethernet)
- Tendencia → agregación de paquetes en flujos
- Ejemplos: Multi-protocol Label Switching (MPLS), VLANs (802.1Q)

Agregacion de flujos

Virtualizacion de redes



Tecnologías Actuales

Requerimientos → flexibilidad, performance

Circuitos de Propósito Específico (ASICs)

- Cientos de bloques especializados trabajando en paralelo
- Alto desempeño. No programables, alto costo y tiempo de desarrollo.

Network Processors (NPs

- Múltiples elementos de procesamiento, buena performance para ciertas tareas. IXP(Intel), PowerNP (IBM)
- Difícil portabilidad, interfaces propietarias

Procesadores de Propósito General (GPPs)

- Arquitectura PC + Software especializado: Click, Zebra/Xorp/Quagga
- Alta flexibilidad, bajo costo. Limitación por transacciones con RAM y naturaleza secuencial



Tecnologías Actuales

Requerimientos \rightarrow flexibilidad, performance

Circuitos de Propósito Específico (ASICs)

- Cientos de bloques especializados trabajando en paralelo
- Alto desempeño. No programables, alto costo y tiempo de desarrollo.

Network Processors (NPs)

- Múltiples elementos de procesamiento, buena performance para ciertas tareas. IXP(Intel), PowerNP (IBM)
- Difícil portabilidad, interfaces propietarias

Procesadores de Propósito General (GPPs)

- Arquitectura PC + Software especializado: Click, Zebra/Xorp/Quagga
- Alta flexibilidad, bajo costo. Limitación por transacciones con RAM y naturaleza secuencial



Tecnologías Actuales

Requerimientos \rightarrow flexibilidad, performance

Circuitos de Propósito Específico (ASICs)

- Cientos de bloques especializados trabajando en paralelo
- Alto desempeño. No programables, alto costo y tiempo de desarrollo.

Network Processors (NPs)

- Múltiples elementos de procesamiento, buena performance para ciertas tareas. IXP(Intel), PowerNP (IBM)
- Difícil portabilidad, interfaces propietarias

Procesadores de Propósito General (GPPs)

- Arquitectura PC + Software especializado: Click, Zebra/Xorp/Quagga
- Alta flexibilidad, bajo costo. Limitación por transacciones con RAM y naturaleza secuencial



Nuevas tecnologías

Dispositivos Lógicos Programables (FPGAs)

- Permiten reconfiguración y reprogramación, contando con librerías de Open Hardware.
- Su performance no es lejana a la de un ASIC. Fabricantes: Altera, Xilinx, Actel.
- Incorporación creciente de bloques hardcore especializados

Implementación con GPPs

Implementación con NPs

Implementación con FPGAs





Motivación

- Requerimientos
 - Soluciones
 - Problema marco
 - Objetivos



Sistem

- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras



Diagrama en bloques: Lineas E/S



- NIOS II
- Herramientas / Recursos Utilizados
- Verificación



- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Clasificación de Paquetes

Clasificación

- La necesidad de procesar cada vez más paquetes de datos lleva a lo que se conoce como clasificación.
- Es el proceso de categorización de paquetes en distintos flujos.
- Efectuada en base a un número de campos de una cabecera.
- En general, para una clasificación basada en N campos, se dice que la misma es N-dimensional (o multidimensional)
- Un caso en particular de la clasificación unidimensional (N=1) es lo que se conoce como lookup.

LookUp

Lookup

- Se lleva a cabo en el dispositivo de enrutamiento.
- Un paquete llega por una interfaz de entrada. Éste porta una dirección IP determinada.
- El dispositivo consulta una tabla de forwardeo para determinar la interfaz de salida para el paquete en cuestión
- Dicha tabla contiene un conjunto de prefijos con sus correspondientes interfaces de salida.
- El paquete es correspondido con el prefijo más largo que esté contenido en la dirección de destino y luego es redirigido a la correspondiente interfaz de salida.



Motivación

- Requerimientos
 - Soluciones
 - Problema marco
 - Objetivos



Sisten

- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras



- Diagrama en bloques: Lineas E/S
- 4 Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación



- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Objetivos

Generales

- Estudiar las diversas arquitecturas de clasificación de paquetes para poder encontrar las limitaciones en la implementación de las mismas tanto en software como en hardware.
- Ganar conocimiento acerca las diversas posibilidades que ofrecen las FPGA para la implementacion de este tipo de algoritmos y las opciones con las que se cuenta a la hora de implementar un sistema embebido en este tipo de dispositivos.

Específicos

- Implementar un sistema embebido que realice la clasificiación unidimensional de paquetes mediante una arquitectura mixta, Hardware-Software, en lógica reprogramable y que permita contratastar algunos de los algoritmos de clasificación existentes.
- Implementar como mínimo dos algoritmos de clasificación.
- Mejorar las algoritmos anteriormente mencionados, poniendo el foco en optimizar el código.



Objetivos

Generales

- Estudiar las diversas arquitecturas de clasificacion de paquetes para poder encontrar las limitaciones en la implementacion de las mismas tanto en software como en hardware.
- Ganar conocimiento acerca las diversas posibilidades que ofrecen las FPGA para la implementacion de este tipo de algoritmos y las opciones con las que se cuenta a la hora de implementar un sistema embebido en este tipo de dispositivos.

Específicos

- Implementar un sistema embebido que realice la clasificiación unidimensional de paquetes mediante una arquitectura mixta, Hardware-Software, en lógica reprogramable y que permita contratastar algunos de los algoritmos de clasificacion existentes.
- Implementar como mínimo dos algoritmos de clasificación.
- Mejorar las algoritmos anteriormente mencionados, poniendo el foco en optimizar el código.





- Requerimientos
 - Solucione:
 - Problema marco
 - Objetivos



Solución Propuesta

- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras

Arquitectura

Diagrama en bloques: Lineas E/S

Implementación

- NIOS II
- Herramientas / Recursos Utilizados
- Verificación



- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Solución

... Diagrama de bloques ...



- Requerimientos
 - Soluciones
 - Problema marco
 - Objetivos



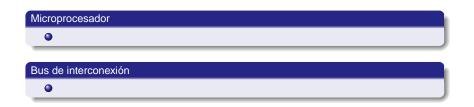
- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras
- Arquitectura
 - Diagrama en bloques: Lineas E/S
- Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación



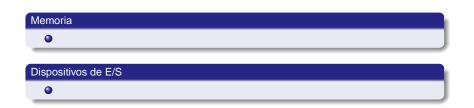
- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Microprocesador • Bus de interconexión









- Requerimientos
 - Soluciones
 - Problema marco
 - Objetivos
- Objetiv



- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras
- 3 Arquitectura
 - Diagrama en bloques: Lineas E/S
- 4 Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación
- Resultados
 - Introducción
 - Caso algoritmos unicamente
 - Caso loopback
 - Implementacion completa
 - Comparativa inter-algoritmos



Algoritmos de Clasificación



Algoritmos de Clasificación

```
Linear Lookup (LLU)

Unibit trie lookup (UTL)
```



- Requerimientos
- Soluciones
- Problema marco
- Objetivos



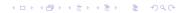
- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras



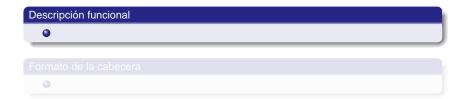
- Diagrama en bloques: Lineas E/S
- Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación



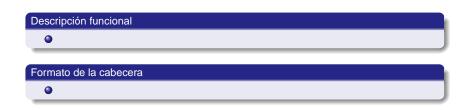
- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Modulo extractor de cabeceras



Modulo extractor de cabeceras





- Requerimientos
 - Soluciones
 - Problema marco
 - Objetivos



- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras



Diagrama en bloques: Lineas E/S



- NIOS II
- Herramientas / Recursos Utilizados
- Verificación



- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Diagrama en bloques: Lineas E/S

-aqui el diagrama ...

Arquitectura

Interfaz Avalon MM





- Requerimientos
 - Solucione:
 - Problema marco
 - Objetivos
- Objetivo



- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras
- Arquitectur
 - Diagrama en bloques: Lineas E/S
- 4 Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación
- 5 Resultados
 - Introducción
 - Caso algoritmos unicamente
 - Caso loopback
 - Implementacion completa
 - Comparativa inter-algoritmos



Microprocesador NIOS II

NIOS II



- Requerimientos
 - Solucione:
 - Problema marco
 - Objetivos



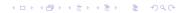
- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras



- Diagrama en bloques: Lineas E/S
- 4 Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación



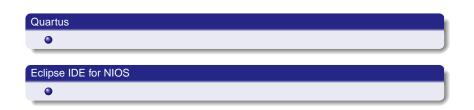
- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Herramientas / Recursos Utilizados



Herramientas / Recursos Utilizados





- Requerimientos
 - Soluciones
 - Problema marco
 - Objetivos



- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras



- Diagrama en bloques: Lineas E/S
- 4 Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación



- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Verificación

Etapas

- 1º Paso: Módulo más simple (simpleRW)
- 2º Paso: Implementación del modulo extractor. Debugging de señales.
- 3º Paso: Integración extractor-software. LLU y UTL.



- Requerimientos
 - Solucione:
 - Problema marco
 - Objetivos



- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras



Diagrama en bloques: Lineas E/S



- NIOS II
- Herramientas / Recursos Utilizados
- Verificación



- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Introducción

Presentación de los resultados





- Requerimientos
 - Soluciones
 - Problema marco
 - Objetivos



- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras



Diagrama en bloques: Lineas E/S



- NIOS II
- Herramientas / Recursos Utilizados
- Verificación



- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Caso algoritmos unicamente

.. Grafico correspondiente ...



- Requerimientos
 - Solucione:
 - Problema marco
 - Objetivos



- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras



Diagrama en bloques: Lineas E/S



- NIOS II
- Herramientas / Recursos Utilizados
- Verificación



- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algoritmos



Caso loopback

.. Grafico correspondiente ...



- Requerimientos
 - Solucione:
 - Problema marco
 - Objetivos



- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras
- 3 Arquitectura
 - Diagrama en bloques: Lineas E/S
- 4 Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación
- Resultados
 - Introducción
 - Caso algoritmos unicamente
 - Caso loopback
 - Implementacion completa
 - Comparativa inter-algortimos



Implementación completa: LLU

.. Graficos retardo mínimo ...

Implementación completa: LLU

.. Graficos retardo promedio ...

Implementación completa: LLU

.. Graficos retardo máximo ...

Implementación completa: UTL

.. Graficos retardo mínimo ...

Implementación completa: UTL

.. Graficos retardo promedio ...

Implementación completa: UTL

.. Graficos retardo máximo ...



- Requerimientos
 - Solucione
 - Problema marco
 - Objetivos



- Solución Propuesta
- Descripcion funcional de cada bloque
- Algoritmos de Clasificación
- Modulo extractor de cabeceras



- Diagrama en bloques: Lineas E/S
- Implementación
 - NIOS II
 - Herramientas / Recursos Utilizados
 - Verificación



- Introducción
- Caso algoritmos unicamente
- Caso loopback
- Implementacion completa
- Comparativa inter-algortimos



Comparativa inter-algortimos

.. Grafico comparativo ...

Conclusiones

Conclusiones