



Universidad Nacional de San Luis

Facultad de Ciencias Físico, Matemáticas y Naturales  
Ingeniería Electrónica con Orientación en Sistemas Digitales

# Diseño e implementación de plataforma aceleradora de cálculos de percolación, utilizando FPGAs

Autor : Orellana Mesas, Miguel Angel  
Director : Mg. Ing. Palavecino, Mauricio Nicotra

13 Diciembre 2019

# Índice

- 1** **Introducción**
  - Teoría de Percolación
  - Trabajo Base
  - Objetivos
- 2** **Descripción de la Plataforma**
  - Soporte Físico
  - Hardware de la Plataforma
  - Software de la Plataforma
- 3** **Aportes y Mejoras**
  - Red Cúbica
  - Mejoras sobre red cuadrada de monómeros
- 4** **Resultados y Simulaciones**
  - Plataforma Implementada
  - Curvas de probabilidad de Percolación
  - Análisis mejoras propuestas
- 5** **Conclusiones**



# Índice

## 1 Introducción

- Teoría de Percolación
- Trabajo Base
- Objetivos

## 2 Descripción de la Plataforma

- Soporte Físico
- Hardware de la Plataforma
- Software de la Plataforma

## 3 Aportes y Mejoras

- Red Cúbica
- Mejoras sobre red cuadrada de monómeros

## 4 Resultados y Simulaciones

- Plataforma Implementada
- Curvas de probabilidad de Percolación
- Análisis mejoras propuestas

## 5 Conclusiones



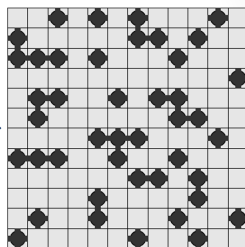


# Aplicaciones de la Teoría de Percolación

- Filtración de Líquidos a través de Medios Porosos
- Estudio de Incendios Forestales
- Estudios Sociales y Poblacional
- Propagación de Epidemias
- Propagación de Información en Redes Inalámbricas
- Estudio de Terremotos y Patrones de Fallas
- Estudio de Redes Eléctricas Aleatorias y Semiconductores
- Simulación de Formación de Macromoléculas y Polímeros
- Reacciones Químicas en Medios Porosos
- Simulación de Formación de Estrellas
- Estudio de Ruteo en FPGA



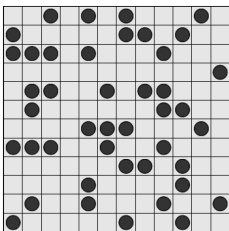
# Trabajo Base - Modelado



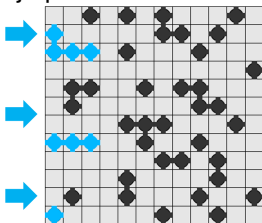
El abstracción consistió en modelar el **medio poroso** como una **red cuadrada bidimensional**, en la que cada **sitio** representa una **parte sólida** o un **poro** del medio. La **cantidad de poros** en la red representa la **porosidad del medio**, y su distribución se realiza en forma aleatoria, emulando la distribución natural aleatoria de los poros en el medio real.



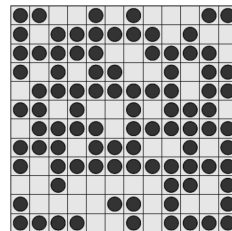
# Trabajo Base - Concepto de percolación



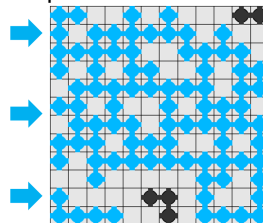
Baja porosidad del medio



Medio no percolante



Alta porosidad del medio



Medio percolante



# Trabajo Base - Abstracciones de la Teoría de Percolación

## Modelo de B. y H.

Medio poroso  
Punto sólido  
Poro  
Medio percolante  
Porosidad

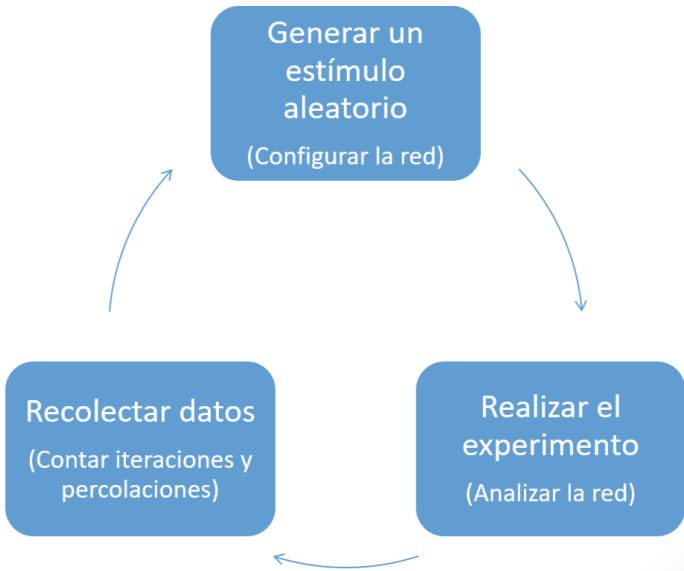
## Abstracciones

Red  
Sitio libre o vacío  
Sitio ocupado  
Red percolante  
Ocupación o  
cubrimiento de la  
red ( $\theta$ )





# Trabajo Base - Método Monte-Carlo



# Trabajo Base - Sistema implementado

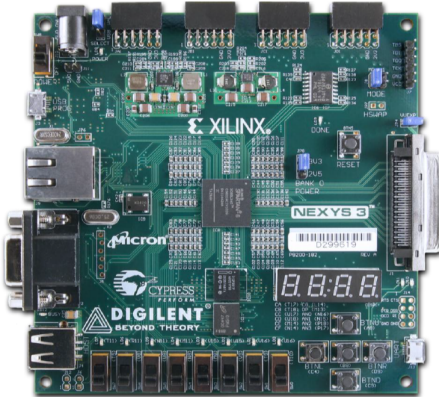


Redes cuadradas con distribuciones de:

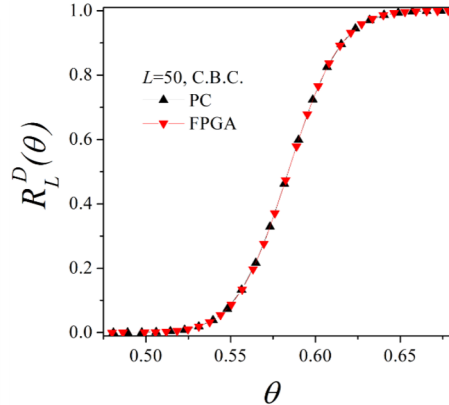
1. Monómeros con condiciones de borde abiertas
2. Monómeros con condiciones de borde cerradas
3. Dímeros con condiciones de borde abiertas
4. Dímeros con condiciones de borde cerradas



# Trabajo Base - Resultados



Placa de desarrollo Nexys 3



Curva de probabilidad de percolacion



# Trabajo Base - Alcances

## Resultados exitosos del Trabajo Base

- Resultados de la misma calidad y con una mejora de performance de un orden de magnitud respecto de algoritmo en PC.
- Se implementaron cuatro diseños escalables en tamaño.
- Menor consumo de energía.

## Limitaciones del Trabajo Base

- Simulación de la curva completa de probabilidad de percolación.
- Conversiones de bases numéricas por parte del usuario.
- Interfaz de usuario poco práctica y poco versátil.
- Manipulación necesaria del hardware por parte del usuario.



# Objetivos del presente Trabajo

## Objetivos principales

- Diseño y desarrollo de una plataforma basada en FPGA que establezca una comunicación efectiva con un usuario a través de una PC convencional, para la recepción de parámetros de simulación y la comunicación de los resultados obtenidos.
- El diseño, implementación e integración de redes tridimensionales cúbicas de sitios de geometría uniforme de monómeros.
- Búsqueda de mejoras en la performance del circuito de análisis de percolación y/o reducción en la utilización de recursos lógicos por parte de éste.
- Determinar las mayores dimensiones de redes que pueden ser implementadas, considerando los recursos que son destinados a implementar el sistema de control y comunicación mencionado en el primer objetivo.

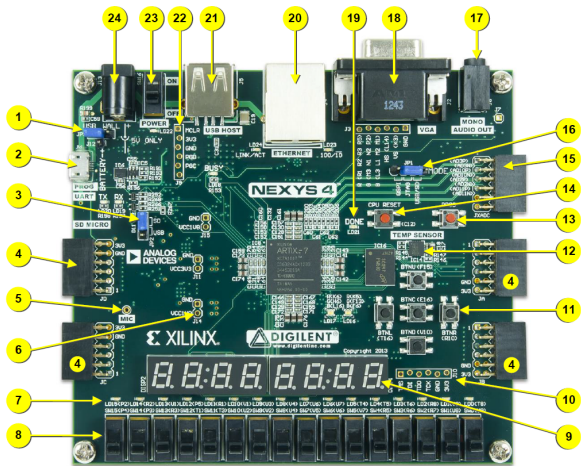


# Índice

- 1 **Introducción**
  - Teoría de Percolación
  - Trabajo Base
  - Objetivos
- 2 **Descripción de la Plataforma**
  - Soporte Físico
  - Hardware de la Plataforma
  - Software de la Plataforma
- 3 **Aportes y Mejoras**
  - Red Cúbica
  - Mejoras sobre red cuadrada de monómeros
- 4 **Resultados y Simulaciones**
  - Plataforma Implementada
  - Curvas de probabilidad de Percolación
  - Análisis mejoras propuestas
- 5 **Conclusiones**



# Soporte Físico



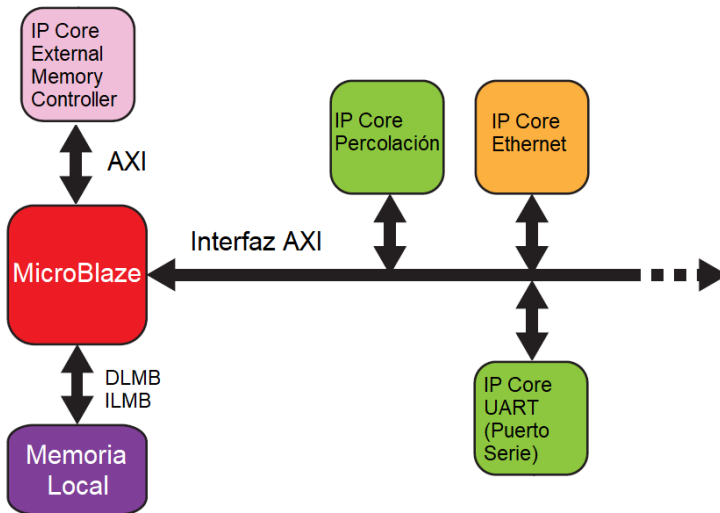
Nexys 4 - FPGA Artix 7, modelo  
XC7A100T-1CSG324C

Componentes utilizados en el desarrollo :

- Puerto Ethernet
- Puerto Micro-AB USB JTAG / UART compartido
- Memorias Cellular RAM y SPI Flash
- LEDs
- Switches
- Displays 7 segmentos
- Pulsadores

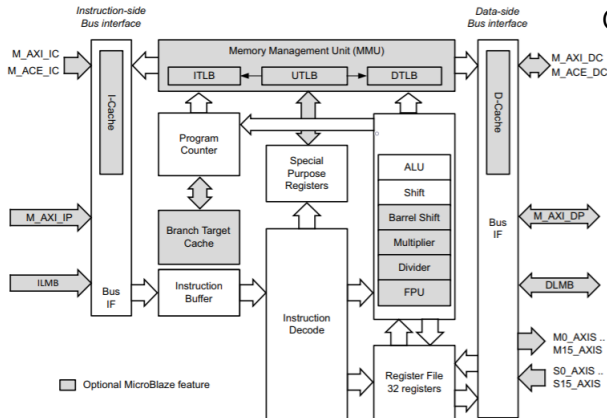


# Hardware de la Plataforma - Diagrama simplificado





# Hardware de la Plataforma - MicroBlaze (v11.0)



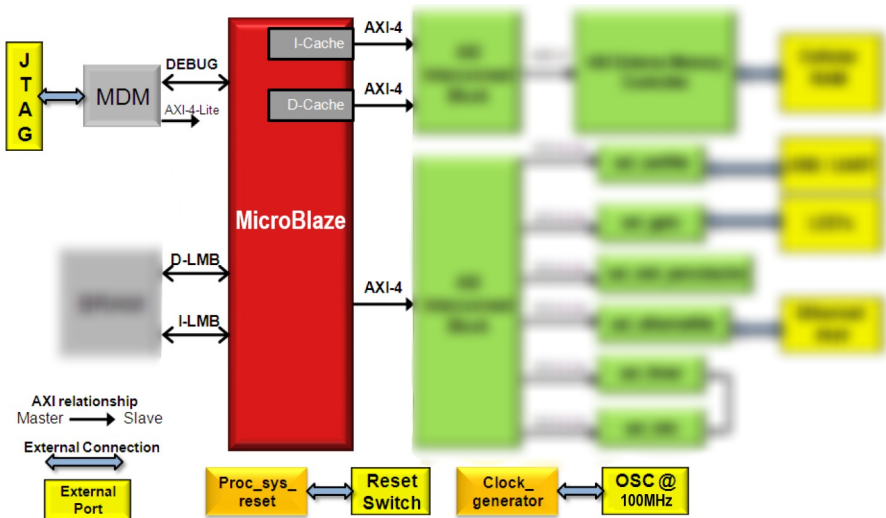
## Características :

- Arquitectura Hardvard
- Soft Processor RISC
- 32 Registros de 32 o 64bits
- Bus de dirección de 32 bits extensible a 64
- Incluye tanto formato Big-Endian como Little-Endian con inversión de bits
- Soporta LMB,AXI,ACE,Debug

**MicroBlaze**



# Hardware de la Plataforma - MicroBlaze



# Hardware de la Plataforma - AXI

AXI(Advanced eXtensible Interface) es parte del estándar abierto ARM AMBA (Advanced Microcontroller Bus Architecture).

Las características clave del protocolo AXI son :

- Buses de direcciones/control y datos independientes.
- Soporte para transferencias de datos no alineados.
- Utiliza transacciones basadas en ráfagas (es decir, el envío de grandes bloques de datos que pueden ser leídos en paralelo).
- Canales de datos de lectura y escritura separados.



# Hardware de la Plataforma - Tipos de interfaces AXI4

## AXI4

- Proporciona el mejor rendimiento.
- Permite ráfagas de transferencia de datos de tamaño limitado.
- Posee mapeo de memoria.

## AXI4-Lite

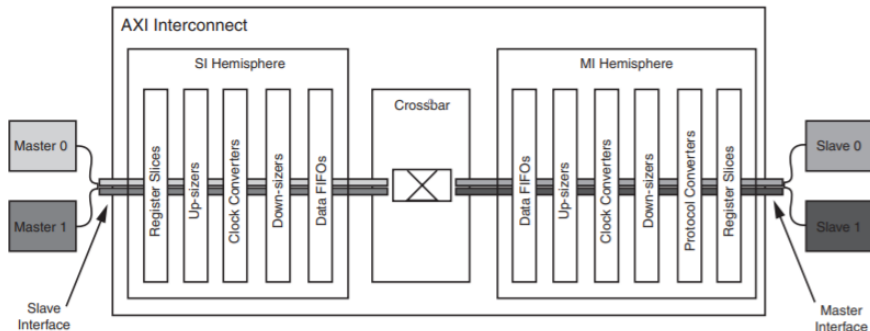
- Utilizado para comunicaciones con registros de control y estado.
- Sólo una transferencia de datos por conexión (sin ráfagas).
- También posee mapeo de memoria.

## AXI4-Stream

- Utilizado para transmisión de datos a alta velocidad.
- Admite transferencias en ráfaga de tamaño ilimitado.
- No posee mapeo de memoria.



# Hardware de la Plataforma - AXI Interconnect

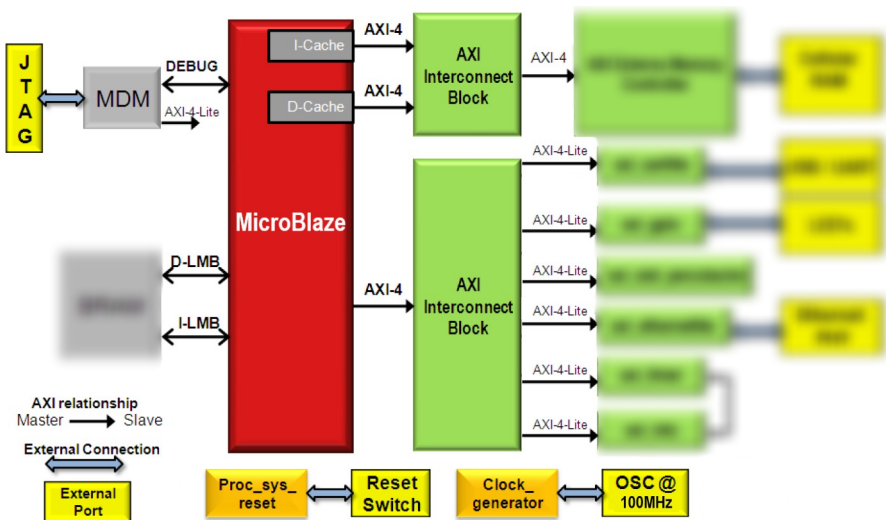


Las especificaciones AXI describen una interfaz entre un único maestro AXI y un solo esclavo AXI.

De modo que para realizar la conexión entre varios maestros y esclavos mapeados en memoria es necesario un bloque interconector, denominado Xilinx AXI Interconnect IP.



# Hardware de la Plataforma - AXI Interconnect



# Hardware de la Plataforma - Memorias del Sistema

- Memoria Local : Implementada con Bloques RAM de la FPGA. Interconectada por buses LMB (Local Memory Bus)
- Memoria Externa : Cellular RAM localizada en la placa de desarrollo Nexys4. Interconectada por interfaz AXI4

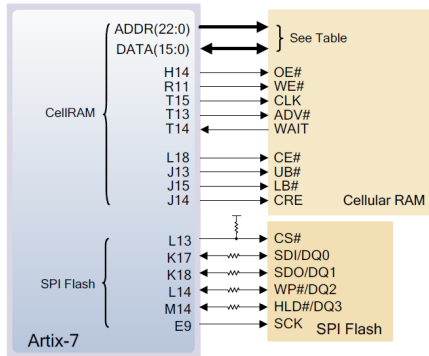
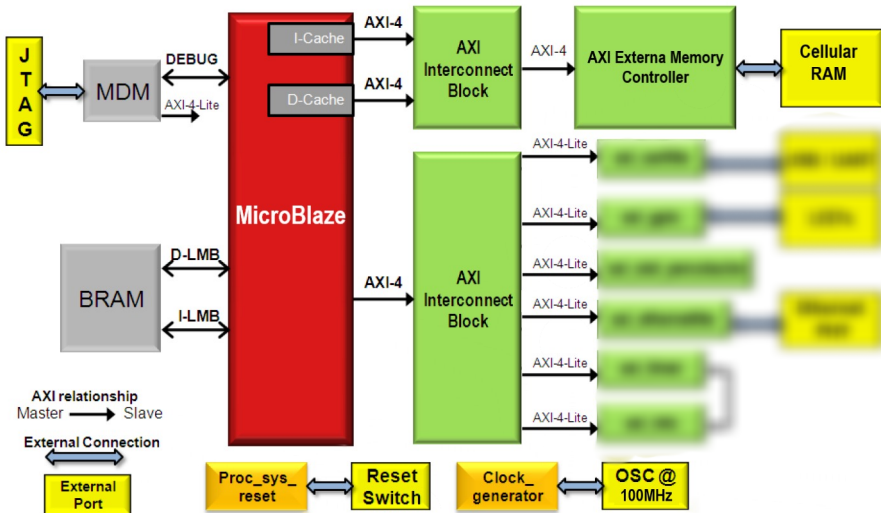


FIGURE – Memoria Cellular RAM y SPI Flash.



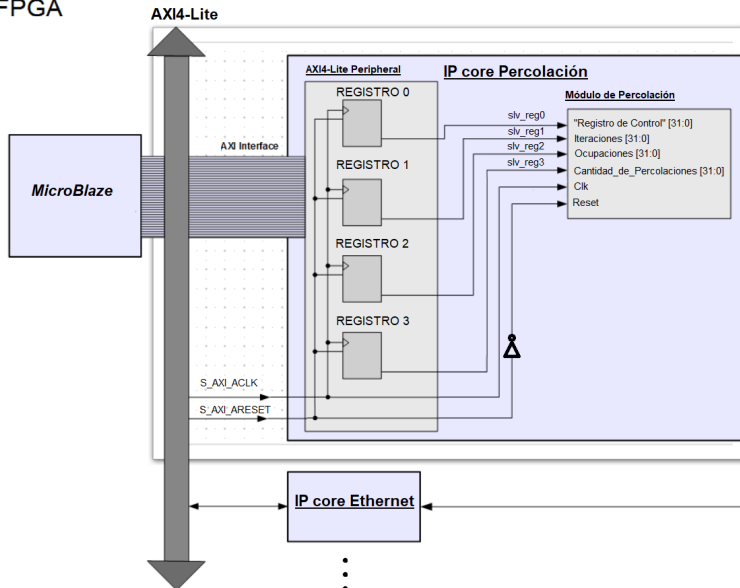
# Hardware de la Plataforma - Memorias del Sistema





# Hardware de la Plataforma - IP core Percolación

FPGA



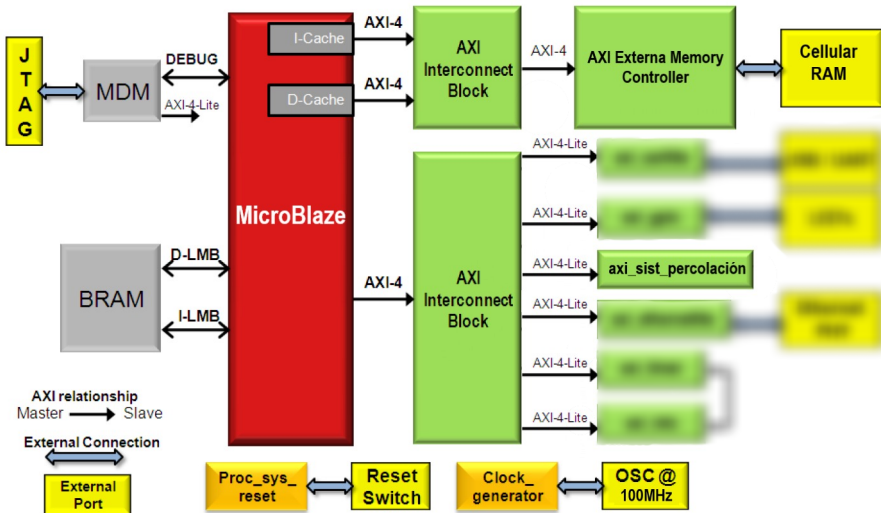
# Hardware de la Plataforma - IP core Percolación

Registros	Bits del Registro	Funcionalidad
Registro_0 (Control/Estado)	slv_reg0[31:3]	sin funcionalidad
	slv_reg0(2)	fin de la simulación - Lectura
	slv_reg0(1)	simulación corriendo - Lectura
	slv_reg0(0)	habilitación de lectura - Escritura
Registro_1 (Valor Sim)	slv_reg1[31:0]	Iteraciones - Escritura
Registro_2 (Valor Sim)	slv_reg2[31:0]	Ocupaciones - Escritura
Registro_3 (Resultados)	slv_reg3[31:0]	Cantidad de Percolaciones - Lectura

FIGURE – Funcionalidad de registros de IP Core de Percolación



# Hardware de la Plataforma - IP core Percolación



# Hardware de la Plataforma - Ethernet

El estándar Ethernet 802.3 integra la capa física y la capa de enlace de datos de modelo OSI, en las denominadas capas PHY(física) y capa MAC(control de acceso al medio) respectivamente.

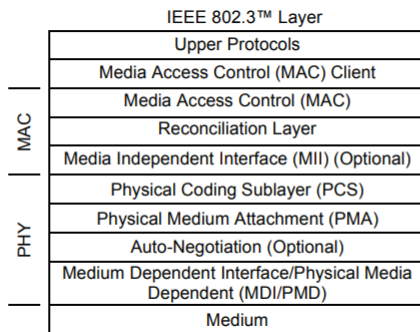


FIGURE – Descripción de sub protocolos integrados en las capas 1 y 2 de modelo OSI.



# Hardware de la Plataforma - Ethernet

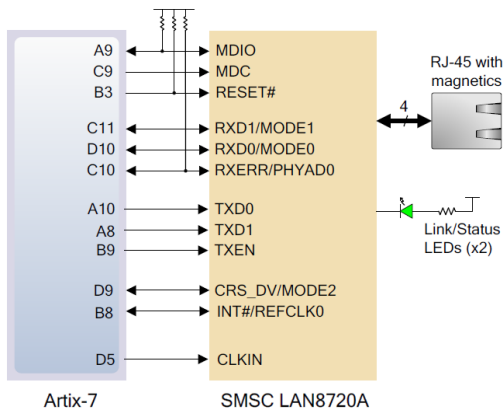


FIGURE – Capa física Ethernet implementada en placa Nexys 4.

El chip SMSC PHY utiliza la interfaz RMII (Reduced Media Independent Interface) la cual es una versión reducida de la MII.



# Hardware de la Plataforma - Ethernet

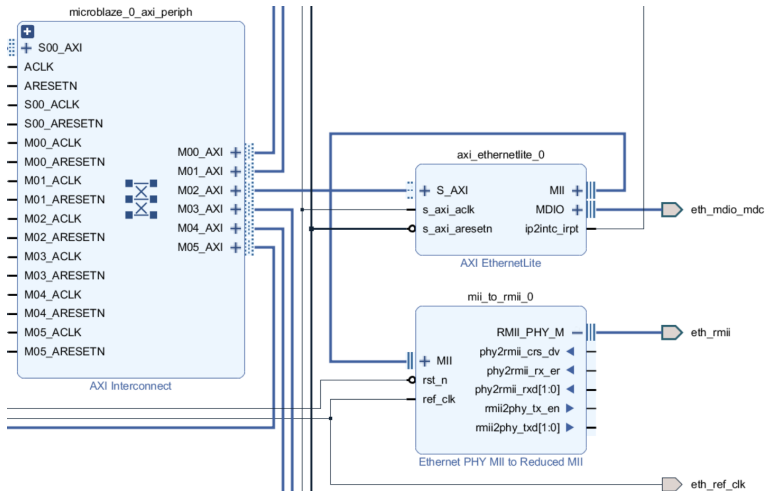
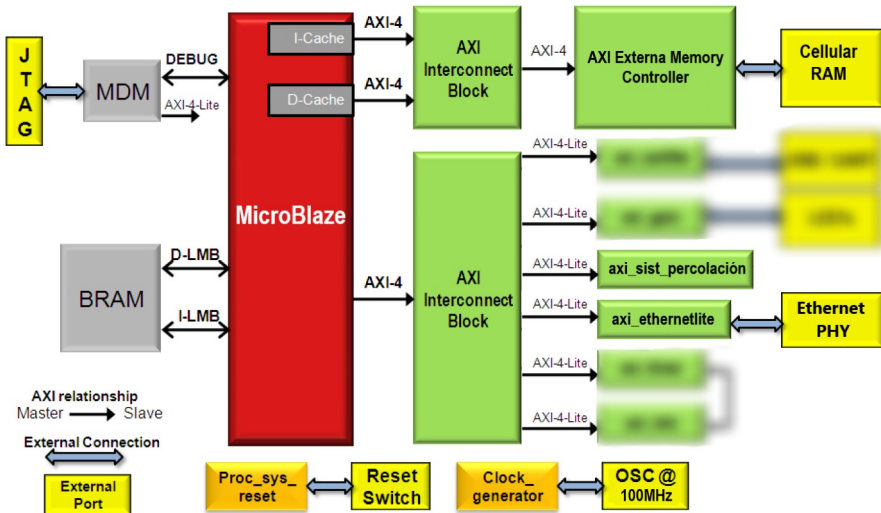


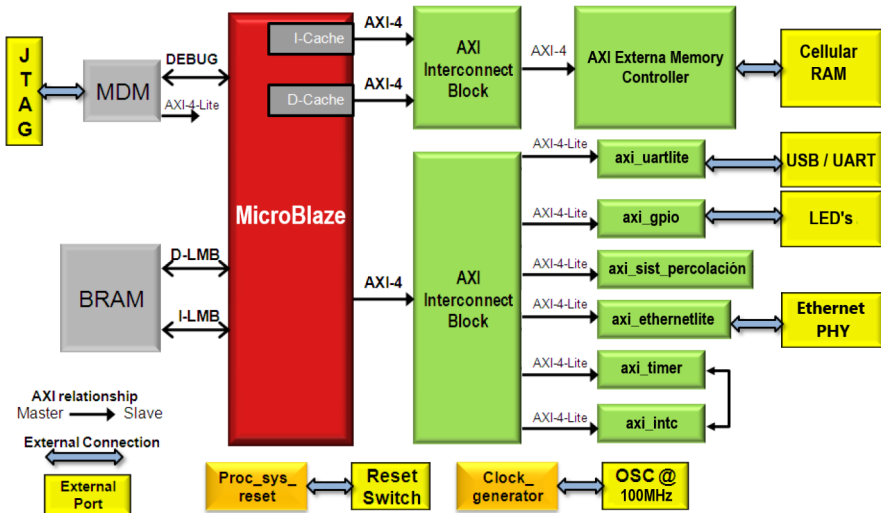
FIGURE – Diagrama de conexión en IP Integrator de Vivado de la capa de enlace de datos en Ethernet.



# Hardware de la Plataforma - Ethernet



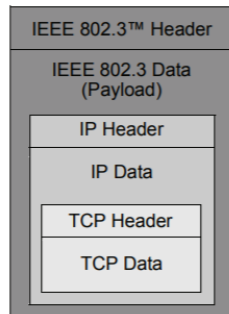
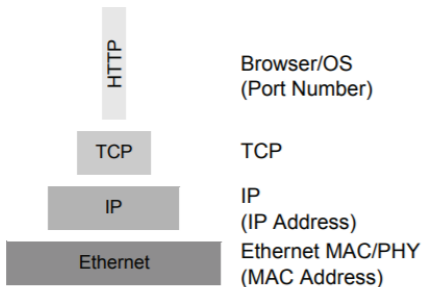
# Hardware de la Plataforma





# Software de la Plataforma - Modelo TCP/IP

Pila de protocolos utilizados en la plataforma



# Software de la Plataforma - LWIP

Para implementar las capas superiores del conjunto de protocolos que dan soporte a nuestra aplicación se ha decidido utilizar la biblioteca LwIP(Lightweight IP).

La versión utilizada en el proyecto es la LwIP 2.0.2 la cual da soporte a los siguientes protocolos :

- Internet Protocol (IP)
- Internet Control Message Protocol (ICMP)
- User Datagram Protocol (UDP)
- Transmission Control Protocol (TCP)
- Address Resolution Protocol (ARP)
- Dynamic Host Configuration Protocol (DHCP)
- Internet Group Message Protocol (IGMP)



# Software de la Plataforma - Web Server

## Front-end

Parte del servicio web a la que un usuario puede acceder directamente. Son todas las tecnologías de diseño y desarrollo web que corren en el navegador y que se encargan de la interactividad con los usuarios.

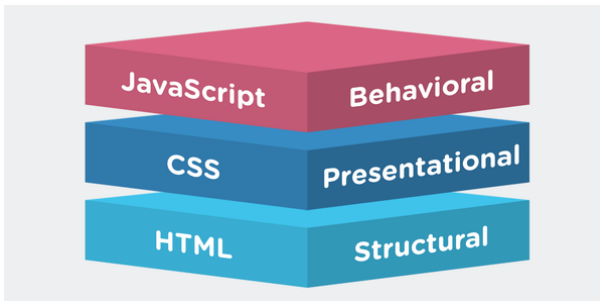
## Back-end

Backend es la capa de acceso a datos de la plataforma, que no es directamente accesible por los usuarios, además contiene la lógica que maneja dichos datos.



# Software de la Plataforma - Front-end

El stack de lenguajes habitualmente utilizado para diseño de front-end :



La biblioteca utilizada para el Front-end es la YUI 2.3.1 de Yahoo, la cual proporcionó, entre otras cosas, interactividad a la interfaz.



# Software de la Plataforma - Front-end

Plataforma de percolacion

Archivo | C:/Users/Miguel/Desktop/project\_1/project\_1.sdk/web\_server/Debug/memfs/inde...

Universidad Nacional de San Luis

## Plataforma de simulación de Red de Percolación en FPGA

Esta plataforma te permite simular una red de percolación de sitios **cúbica de monómeros de 12x12x12 con condiciones de bordes abiertos** en una placa de desarrollo basada en FPGA.

### Documentación

La documentación necesaria para la utilización de esta herramienta la encontraras en el siguiente link: [here](#).

### Plataforma de simulación

Aquí se ingresan los valores correspondientes a la cantidad de iteraciones deseada para el Metodo Monte Carlo, la simulación (siempre que los valores ingresados se mantengan). Para descargar la simulación como un archivo con extensión .txt, para graficar luego la curva de probabilidad de percolación en algún programa, basta con hacer click en **Descargar simulación**.

Cant. de iteraciones

Puntos de muestreo

Ocupación inicial

Ocupación final

Carga - Consulta - Captura

[Descargar simulación](#)



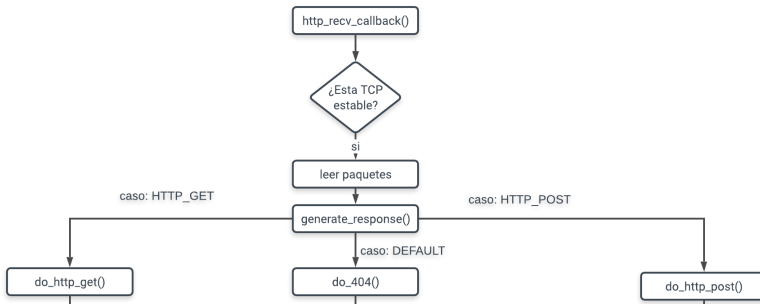
# Software de la Plataforma - Back-end

## Método GET HTTP

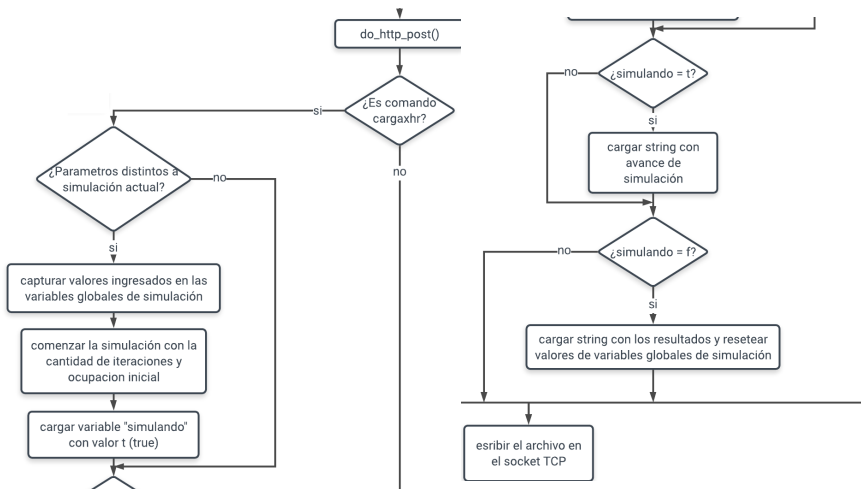
Habitualmente utilizadas para hacer peticiones de datos. La página web, archivos, imágenes, etc.

## Método POST HTTP

El método POST utilizado habitualmente para enviar los datos de un formulario HTML.



# Software de la Plataforma - Back-end



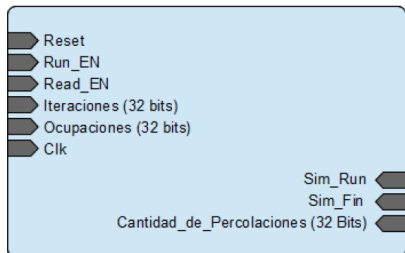
# Índice

- 1 **Introducción**
  - Teoría de Percolación
  - Trabajo Base
  - Objetivos
- 2 **Descripción de la Plataforma**
  - Soporte Físico
  - Hardware de la Plataforma
  - Software de la Plataforma
- 3 **Aportes y Mejoras**
  - Red Cúbica
  - Mejoras sobre red cuadrada de monómeros
- 4 **Resultados y Simulaciones**
  - Plataforma Implementada
  - Curvas de probabilidad de Percolación
  - Análisis mejoras propuestas
- 5 **Conclusiones**



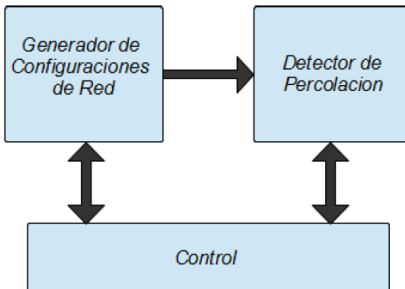


# Red Cúbica

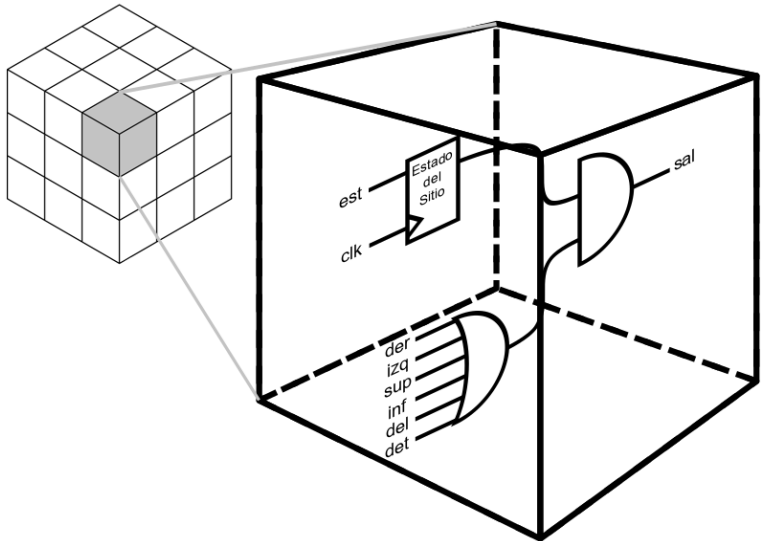


Partiendo del diseño de red cuadrada de monómeros se obtuvo una red cúbica de monómeros.

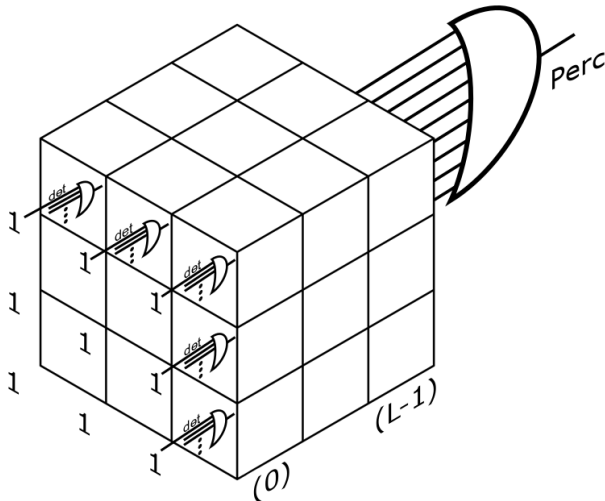
Los cambios se introdujeron en el bloque detector de percolación.



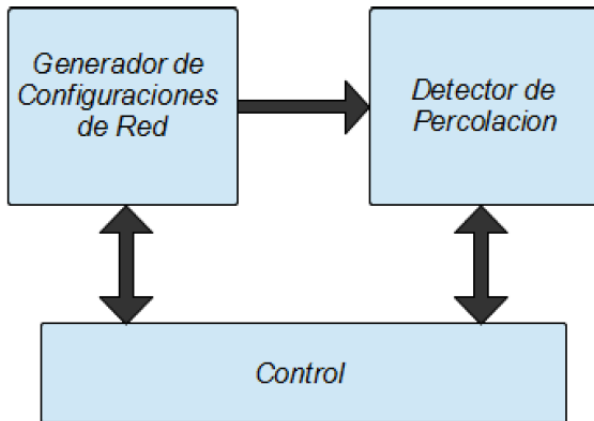
# Red Cúbica



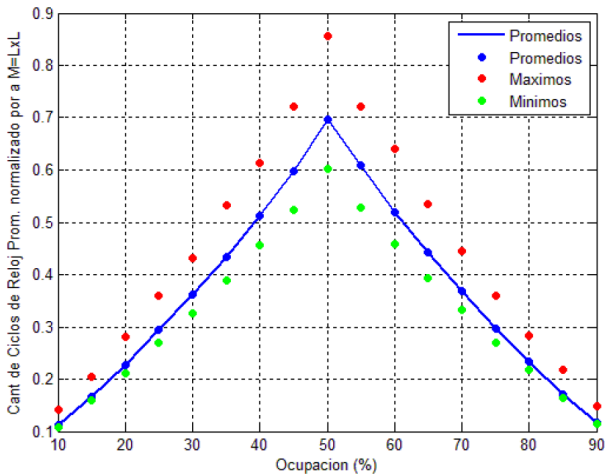
# Red Cúbica



# Mejoras sobre red cuadrada de monómeros



# Mejoras sobre red cuadrada de monómeros



# Índice

- 1 **Introducción**
  - Teoría de Percolación
  - Trabajo Base
  - Objetivos
- 2 **Descripción de la Plataforma**
  - Soporte Físico
  - Hardware de la Plataforma
  - Software de la Plataforma
- 3 **Aportes y Mejoras**
  - Red Cúbica
  - Mejoras sobre red cuadrada de monómeros
- 4 **Resultados y Simulaciones**
  - Plataforma Implementada
  - Curvas de probabilidad de Percolación
  - Análisis mejoras propuestas
- 5 **Conclusiones**



# Plataforma Implementada

Por si misma la plataforma es el primer y fundamental resultado del trabajo.

Este desarrollo soporta :

- uso de diferentes IP cores de Percolación
- utilización de IP Core Ethernet-Lite
- IP Core comunicación serie UART
- Core controlador de GPIOs
- Incorporación de memoria externas al sistema
- y todo IP Core con interfaz de comunicación AXI-Lite.

Los recursos de la FPGA Artix 7 utilizados son :

Recursos Fpga		Recursos de la plataforma		Disponible para sist. percolación	
Recurso	Disp.	Utiliz.	Utiliz. %	Utiliz.	Utiliz. %
LUT	63400	6498	10.24	56902	89.76
FF	126800	6498	5.12	120302	94.87

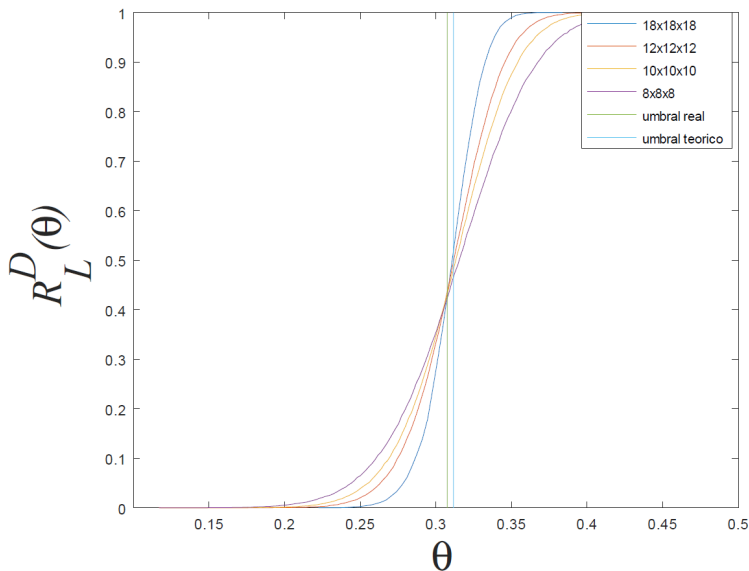
aprox. 10% de los recursos.







# Curvas de probabilidad de Percolación



# Curvas de probabilidad de Percolación

Red de Percolación	Conectividad	Umbral obtenido en la práctica	Umbral Teórico
Cúbica monómeros con condiciones de bordes cerrados	6	$\approx 0.3075$	0.3116
Cúbica monómeros con condiciones de bordes abiertos	6	$\approx 0.312$	-
Cuadrada de monómeros con condiciones de bordes cerrados	4	$\approx 0.59194$	0.592746
Cuadrada de monómeros con condiciones de bordes abiertos	4	$\approx 0.5936$	-
Cuadrada de dímeros con condiciones de bordes cerrados	4	$\approx 0.2811$	-
Cuadrada de dímeros con condiciones de bordes abiertos	4	$\approx 0.2822$	-



# Análisis mejoras propuestas

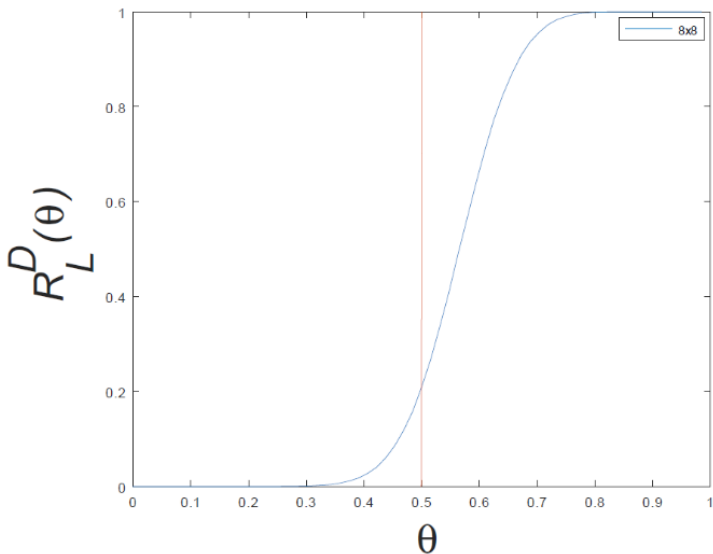


FIGURE – Red de monómeros de 8x8 con bordes cerrados



# Análisis mejoras propuestas

Recursos Fpga		Recursos red 8x8 monómeros	
Recurso	Disp.	Utiliz.	Utiliz. %
LUT	63400	290	0.46
FF	126800	323	0.25

Recursos Fpga		Recursos red modif. 8x8 monómeros	
Recurso	Disp.	Utiliz.	Utiliz. %
LUT	63400	297	0.47
FF	126800	263	0.21

Se tiene una reducción aproximada en utilización de FFs de un 18,6% y un aumento correspondiente de un 2,4% en la utilización de LUTs.

Analizando las dos redes (base y modificada) con una cantidad de iteraciones igual a 100000 se obtuvo en promedio un ahorro en tiempo del 10%, en el rango de ocupación donde aplica la mejora.



# Índice

- 1 **Introducción**
  - Teoría de Percolación
  - Trabajo Base
  - Objetivos
- 2 **Descripción de la Plataforma**
  - Soporte Físico
  - Hardware de la Plataforma
  - Software de la Plataforma
- 3 **Aportes y Mejoras**
  - Red Cúbica
  - Mejoras sobre red cuadrada de monómeros
- 4 **Resultados y Simulaciones**
  - Plataforma Implementada
  - Curvas de probabilidad de Percolación
  - Análisis mejoras propuestas
- 5 **Conclusiones**



# Conclusiones

- Consiguieron los objetivos planteados al inicio del proyecto.
- Se consiguió un ahorro de recursos de la FPGA y una mejora en la performance para la red cuadrada de monomeros, pudiendo extender estas mejoras a las demás redes obteniendo mejores resultados.
- Se realizó una plataforma portable entre las FPGA de la familia Xilinx, pudiendo ser utilizada como base para otros desarrollos.
- Representa un primer intento en relación al desarrollo de clusters de procesamiento remoto basados en FPGAs.



# Trabajos Futuros

En referencia a la plataforma de simulación :

- Administración de varias conexiones simultáneas, posibilitando la utilización de una misma red de percolación (o algún otro IP core) por varios usuarios alternados en el tiempo.
- Implementación de un sistema autorreconfigurable que permita la configuración por demanda de un IP core sintetizado a priori.
- Independización de los relojes del sistema de control y del IP Core de percolación para la mejora de performance en el sistema.
- Utilización de Hard-Processors para optimizar la plataforma, realizando a su vez, simulaciones mixtas entre la parte lógica y el procesador.



# Trabajos Futuros

En referencia al desarrollo de redes de percolación :

- Procurar ampliar el repertorio de redes de percolación, añadiendo por ejemplo redes Hipercúbicas de 4, 5 o más dimensiones, redes triangulares, redes con percolación de enlaces, etc.
- Adaptación de las mejoras (relacionadas al tiempo de ejecución y utilización de recursos) implementadas en la red de monomeros de 8x8 a las demás redes, tanto de dímeros como cúbicas, o incluso si se desarrollan redes de k-meros.
- Utilización de pipe-line interno en las redes para conseguir escalados considerables sin comprometer demasiado la frecuencia del sistema y por ende la velocidad de respuesta.





# Preguntas



# Agradecimientos

A Dios,

A mi Madre, Padre y familia,

A mis Compañeros de cursada y Amigos,

A mis Profesores,

A mi director de trabajo final, Mauricio.



# Agradecimientos

**MUCHAS  
GRACIAS**

