



field-programmable gate array

A central word cloud containing the following terms:

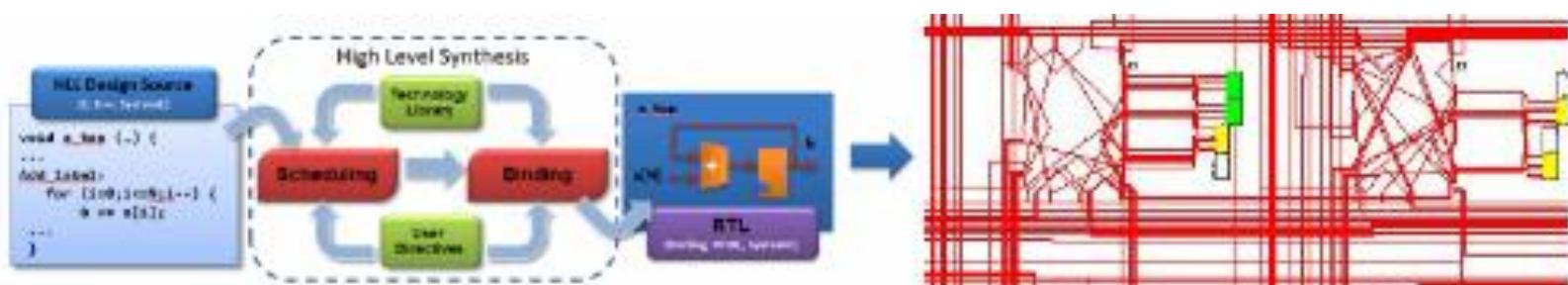
- reprogrammable
- core validation
- validation
- array
- embeded
- track logic
- routing
- function
- sprouted
- multipliers
- map
- external
- internal
- pin
- configuration
- common
- headcount
- manufacturers
- schematic
- number
- area
- power
- compared
- proprietary
- description
- fabric
- multiple
- device
- hardware
- gate
- silicon
- box
- complex
- converters
- similar
- advantages
- circuit
- platform
- system
- prototyping
- flash
- segment
- load
- simulation
- functionalities
- integrated
- reconfiguration
- width
- resources
- analog
- large figure
- implementation
- switch
- routed
- software
- architecture
- microprocessor
- peripherals
- channel
- planar
- configurability
- adjacent
- field

Conjunto Configurable de Compuertas (FPGA, Field Programmable Gate Array)

Introducción

FPGA's

- ▶ Para implementar un diseño óptimo sobre un FPGA, es necesario:
 - ▶ Conocer la arquitectura interna del dispositivo a utilizar.
 - ▶ Utilizar una metodología de diseño adecuada.
 - ▶ Utilizar y conocer una herramienta de diseño adecuada.
 - ▶ Determinar como acceder a los recursos del dispositivo.
 - ▶ Describir con un enfoque orientado al modelado de hardware.
 - ▶ Poder determinar que hardware inferirá el modelado.
 - ▶ Utilizar técnicas de diseño adecuadas a la aplicación y al dispositivo.



FPGA's

- ▶ Historia.
 - ▶ El 1er procesador, 4004 Intel, fue presentado el 15 nov. 1971 (1970).
 - ▶ Derivados de éste: Microprocesador, Microcontrolador y DSP.
 - ▶ Desde entonces y hasta la fecha, estos elementos permiten resolver eficazmente la mayor parte de los problemas electrónicos.
 - ▶ Sin embargo, existe una serie de problemas donde los procesadores no son suficientes, ejemplos de éstos son:
 - ▶ Cuando la E/S de datos combina gran cantidad con gran velocidad.
 - ▶ Cuando el número de operaciones por muestra es elevado.
 - ▶ La misma década vio surgir a los primeros CDC's, el PLA y el PAL (PLDs).
 - ▶ Usados ampliamente, pero su arquitectura los limitó a aplicaciones de baja complejidad.

FPGA's

► Historia.

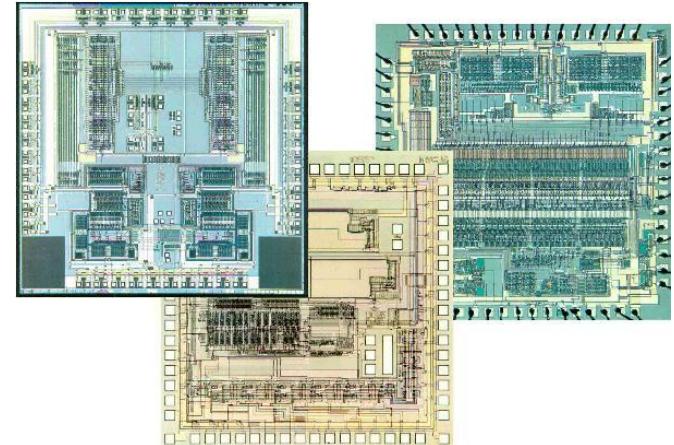
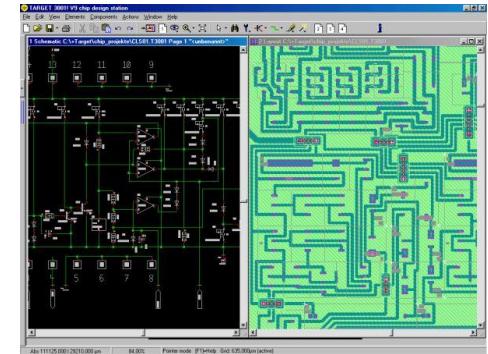
- ▶ A inicios de la década de los 80, cuando los diseñadores se enfrentaron a los problemas que los procesadores presentaban y a las limitantes de los PLDs, apareció la tecnología de los Circuitos Integrados de Aplicación Específica (ASICs, *Application-Specific Integrated Circuits*) - *Semicustom*.
- ▶ Esto derivo en la existencia de dos alternativas para realizar un hardware digital:
 - ▶ Codificación del algoritmo en un procesador
 - ▶ Mapeo directo del algoritmo en hardware (ASIC).

FPGA's

▶ Las principales características de los ASICs son:

- ▶ La mejor relación área-velocidad-potencia,
- ▶ Alta fiabilidad,

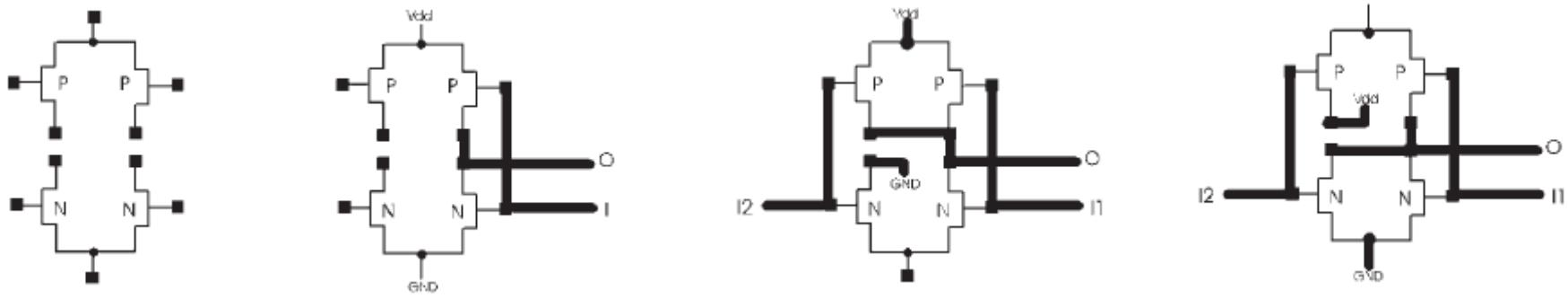
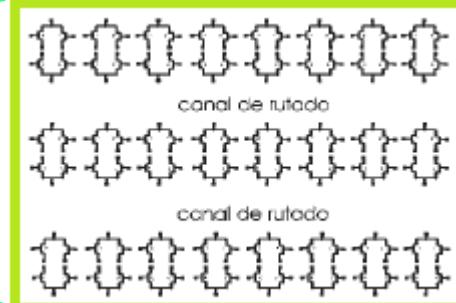
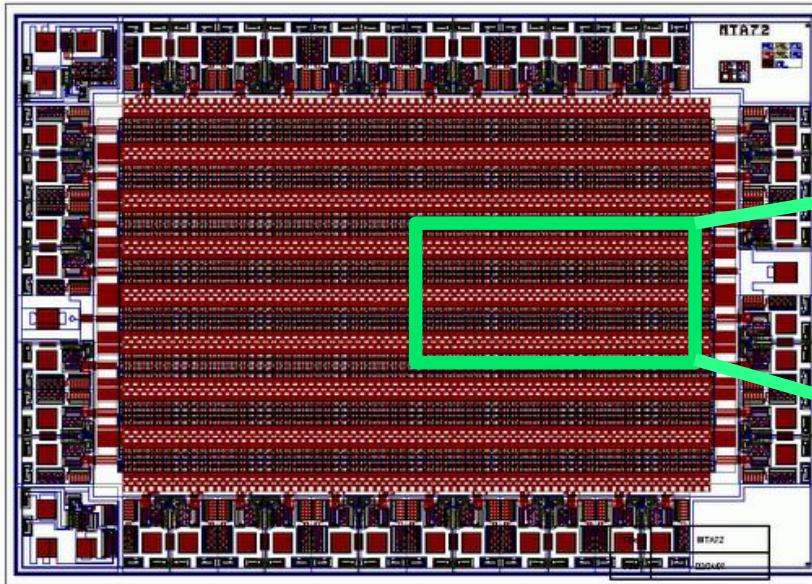
- ▶ Alto costo,
- ▶ Problemas con el stock,
- ▶ Problemas de ventana de mercado,
- ▶ Alta complejidad en su diseño,
- ▶ Herramientas EDA complejas.



La utilización de los ASICs queda restringida a grandes diseños.

FPGA's

- ▶ Los ASIC's basados en arreglos de compuertas (Gate Array) son de interés para este tema.



FPGA's

- ▶ La idea de desarrollar un componente estándar (re)configurable por el usuario, donde las interconexión, las funciones lógicas y la E/S fueran (re)programables, fue propuesta y patentado por S. Wahlstrom en 1967.
- ▶ Pero, se trataba de una idea demasiado adelantada respecto a la tecnología disponible para su fabricación.
- ▶ A mediados de los 80's, los PLDs presentaban limitantes para diseñar sistemas complejos y, en la mayoría de los casos, el costo de los ASICs los hacían una opción prohibitiva.
- ▶ Entonces, la tecnología del FPGA actual, inventado por Ross Freeman y Bernard Vonderschmitt (co-fundadores de Xilinx), en principio llamadas LCA (*Logic Cell Array*), nace en 1984 con una idea sencilla: un Gate Array configurable o reconfigurable por el usuario.
- ▶ Por lo que, un FPGA es la evolución natural de los ASICs, buscando darle el carácter de reconfiguración que un PLD poseía, y surgida de la necesidad de subsanar las limitantes de ambos.

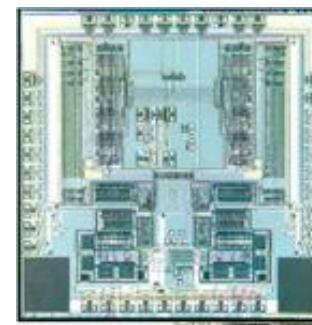
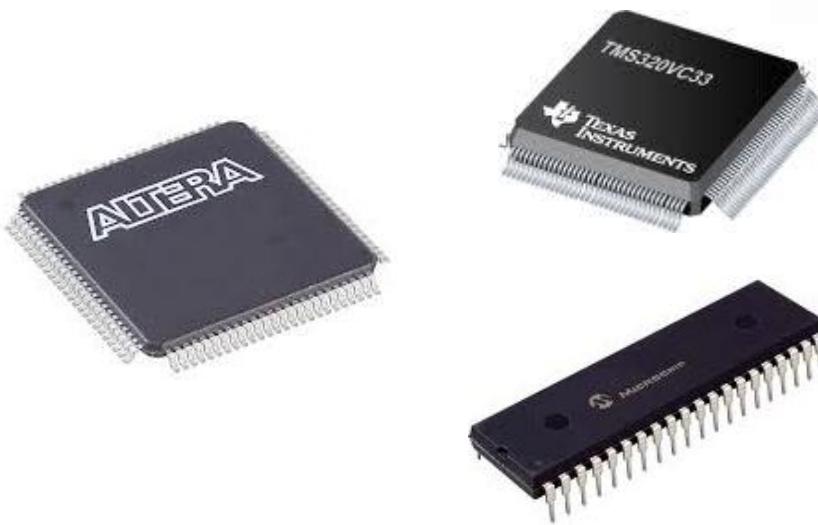
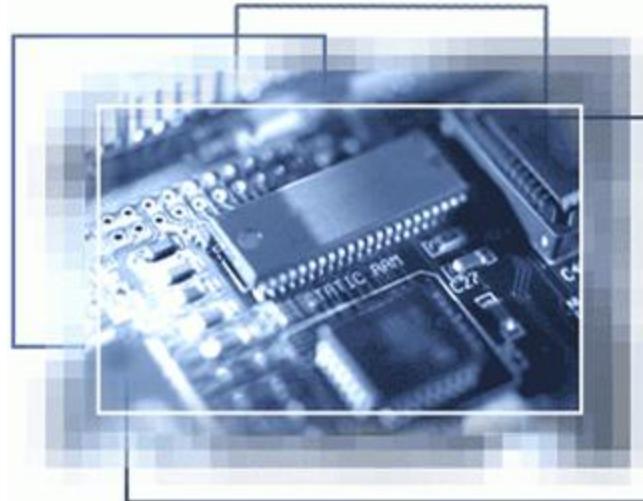
FPGA's

- ▶ En 1985 la compañía Xilinx lanza al mercado su primer FPGA, el XC2064 formado por un arreglo de 64 CLBs, cada uno integrado por 2 LUT's de 3 entradas (con una equivalencia de 10000 compuertas lógicas),
- ▶ A partir de ese momento, un FPGA representa la alternativa ideal, a los PLD's y a los ASIC, en el diseño de sistemas digitales.

FPGA's

▶ Elementos de procesamiento de un sistema embebido.

- ▶ CPLD
- ▶ Procesador
- ▶ ASIC
- ▶ FPGA



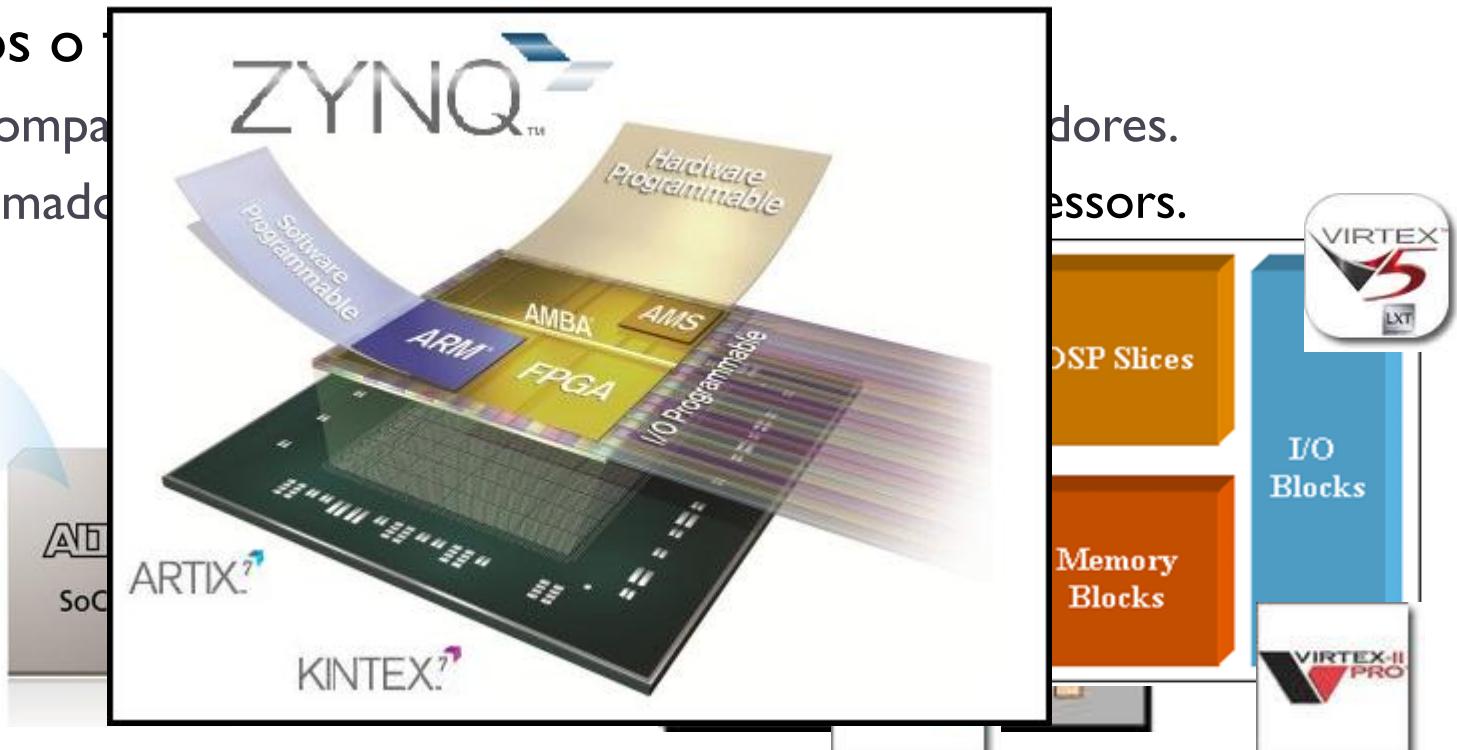
FPGA's

- ▶ FPGAs vs. ASICs.
 - ▶ Los FPGAs se utilizan en aplicaciones similares a los ASICs.
 - ▶ Sin embargo,
 - ▶ Son más lentos,
 - ▶ Tienen un mayor consumo de potencia y,
 - ▶ No pueden abarcar sistemas tan complejos como los ASICs.
 - ▶ A pesar de esto, los FPGAs tienen las ventajas de
 - ▶ Sus costes de desarrollo y adquisición son mucho menores para pequeñas cantidades de dispositivos,
 - ▶ El tiempo de desarrollo es también menor y,
 - ▶ Ser reconfigurables (lo que añade una enorme flexibilidad al flujo de diseño),
 - ▶ Ideal para desarrollo de prototipos.

FPGA's

▶ FPGA vs. PLD

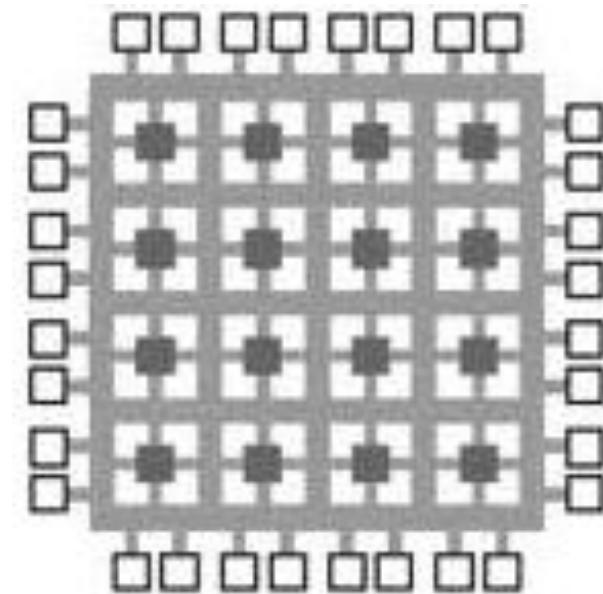
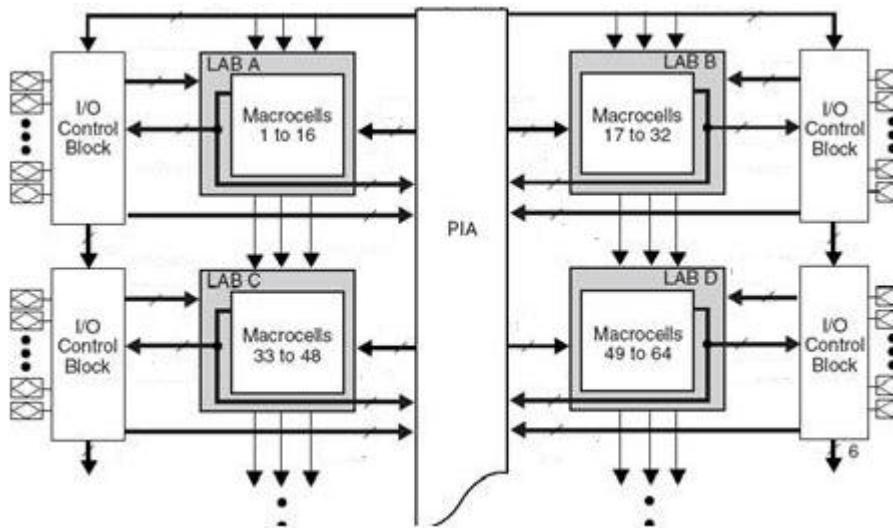
- ▶ Principales diferencias entre un FPGA y un PLD:
 1. Número de elementos lógicos programables (escala de integración).
 - ▶ CPLD está en el orden de millones de compuertas lógicas.
 - ▶ FPGA está en el orden de miles de millones de compuertas lógicas.
 2. Elementos o bloques que integran la FPGA
 - ▶ CPLDs. Compartidos con el resto de la FPGA.
 - ▶ FPGAs. Sumados a los bloques mencionados.



FPGA's

3. Arquitectura.

- ▶ CPLD - Recursos concentrados. Su arquitectura es más rígida y consiste en sumas de productos programables cuyos resultados pueden ser canalizados a un número reducido de biestable.
- ▶ FPGA - Recursos distribuidos. Su arquitectura se basa en un gran número de sencillos bloques lógicos, que cuentan a su vez con biestables, y abundantes recursos de interconexión. Esto confiere a los FPGAs una gran flexibilidad.

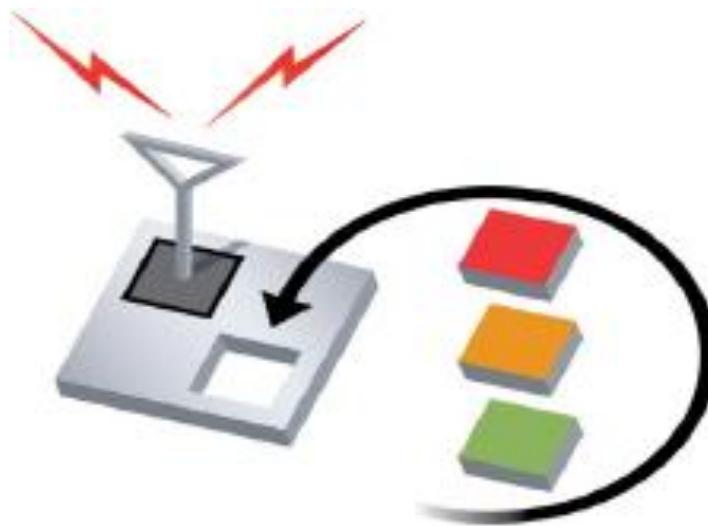


► **Prestaciones: Retardos Predecibles
Hasta 200 MHz**

**Dependiente de aplicación
hasta 500MHz**

4. Reconfiguración dinámica.

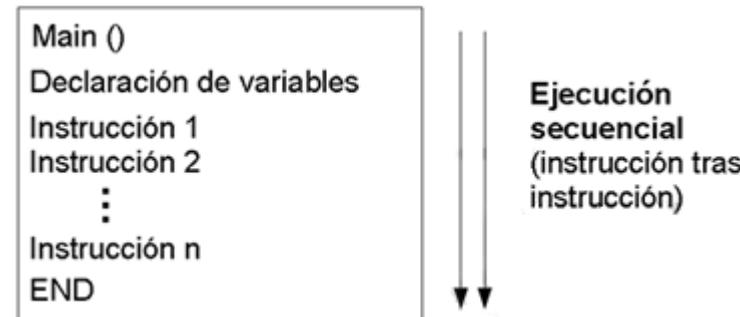
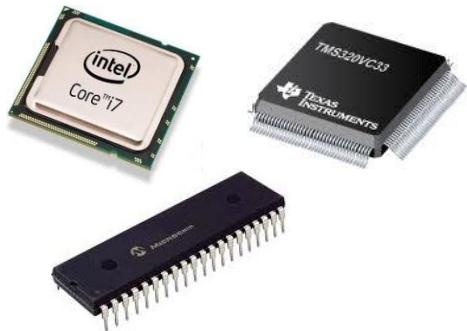
- ▶ Una diferencia importante con respecto a un PLD es que muchos FPGAs modernos soportan una total o parcial re-configureación dinámica del sistema.
- ▶ Esto permite que una parte del diseño sea re-configureado mientras las otras partes siguen funcionando.



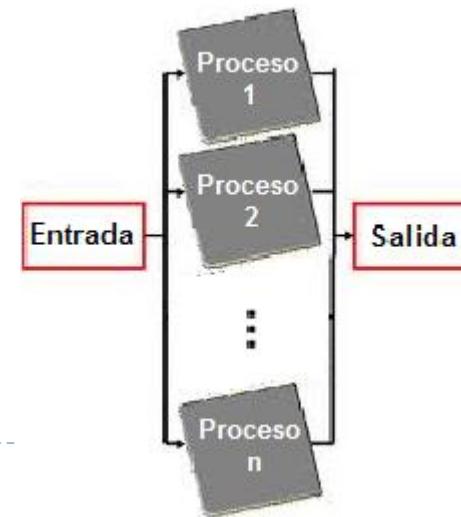
FPGA's

▶ FPGA vs. Procesador

- ▶ Los procesadores (microprocesador, microcontrolador, DSP) son dispositivos basados en CPUs, la tarea que desempeñan está codificada en un conjunto de instrucciones que son ejecutadas en forma secuencial.



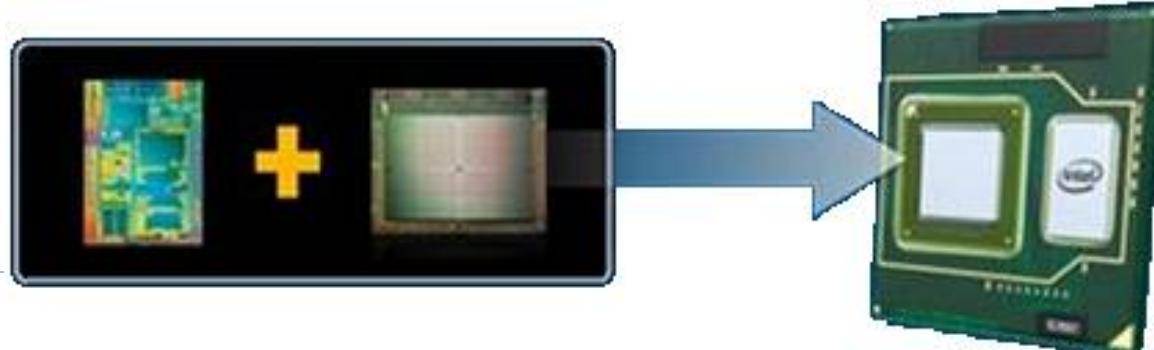
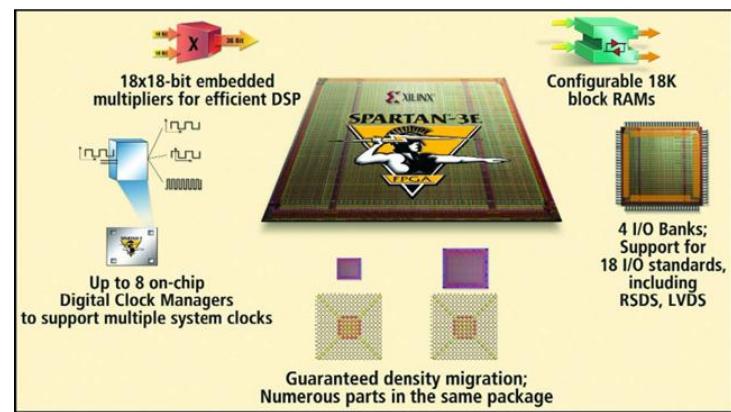
- ▶ Por su parte, Un FPGA es capaz de realizar varias tareas en forma concurrente.



FPGA's

▶ FPGA vs. Procesador

- ▶ Los procesadores pueden incluir gran cantidad y variedad de periféricos embebidos.
- ▶ En un FPGA los periféricos son limitados y orientados al modelado de sistemas generales.
- ▶ Finalmente, existen los llamados Field-Programmable System-on-Chip, Lógica reconfigurable + Procesador en el mismo dispositivo.



FPGA's

- ▶ En general, los FPGAs están consolidados a nivel mundial.
 - ▶ Casi 2/3 de los diseñadores de sistemas digitales usan lógica reconfigurable de alguna clase.
 - ▶ Los promedios varían de acuerdo a la industria, por ejemplo,
 - ▶ En automoción sigue siendo un nicho para los uCs y los ASICs
 - ▶ Pero, En la industria aeroespacial tiene un 74% de penetración.
 - ▶ En la industria de video un 72%
 - ▶ En la industria militar un 69%

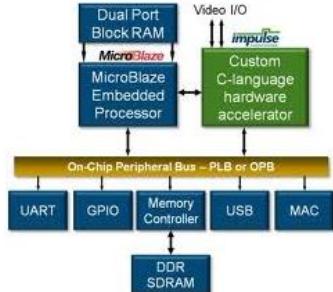


FPGA's

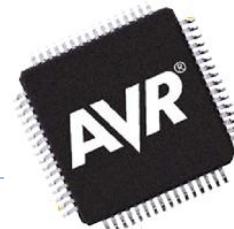
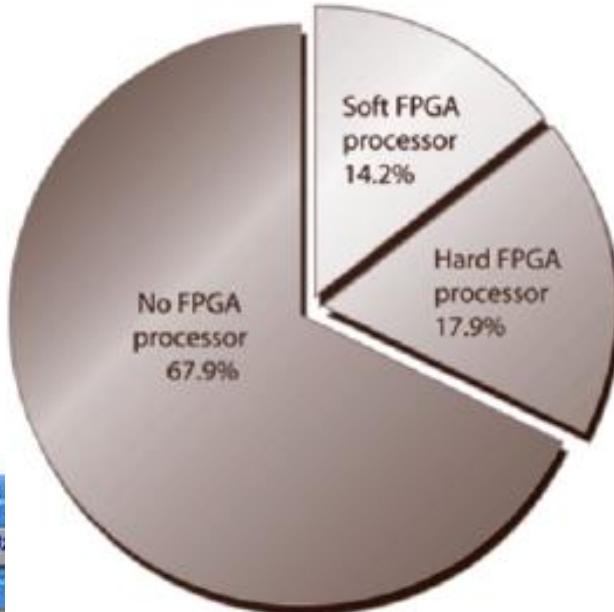
- ▶ Otro punto importante es la invasión de los FPGAs en el mercado de los DSP.
- ▶ Actualmente cerca del 60% de los usuarios de DSP usan un FPGA para realizar un algoritmo DSP.
- ▶ También es evidente que los FPGAs han entrado en el mercado de los procesadores embebidos.
 - ▶ Soft-Processors.
 - ▶ Hard-Processors.



MicroBlaze



PicoBlaze™

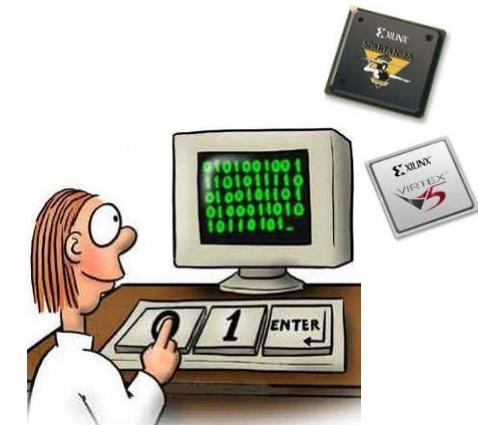
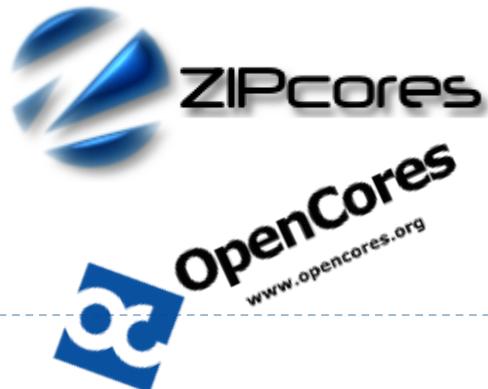


FPGA's

- ▶ Finalmente, existe un poco explotado mercado de los llamados IP (*Intellectual Property*) cores (bloques funcionales existentes).
- ▶ ¿Qué es un IP core? Tipos de IP cores:
 - ▶ Es el modelado de un sistema que incluye su verificación funcional y la documentación necesaria para incluirlo en el diseño final.
 - ▶ **Soft IP.** Es el modelado, normalmente usando un HDL a nivel RTL, de un sistema específico, no optimizado para alguna arquitectura determinada de FPGA.
 - ▶ **Firm IP.** Se trata también de modelados mediante un HDL, pero optimizados para una arquitectura específica de FPGA.
 - ▶ **Hard IP.** Es un elemento de arquitectura fija modelado a un bajo nivel, compuerta o transistor. Debido a la naturaleza de su representación en bajo nivel un Hard IP es ofrecido en formato “Layout” y su implementación se lleva a cabo hasta su fabricación.
- ▶ Los IP cores tienen gran uso en aplicaciones para ASICs y FPGAs
- ▶ Estos bloques están diseñados para ser lo más eficiente posible en términos de consumo de potencia, espacio y rendimiento.

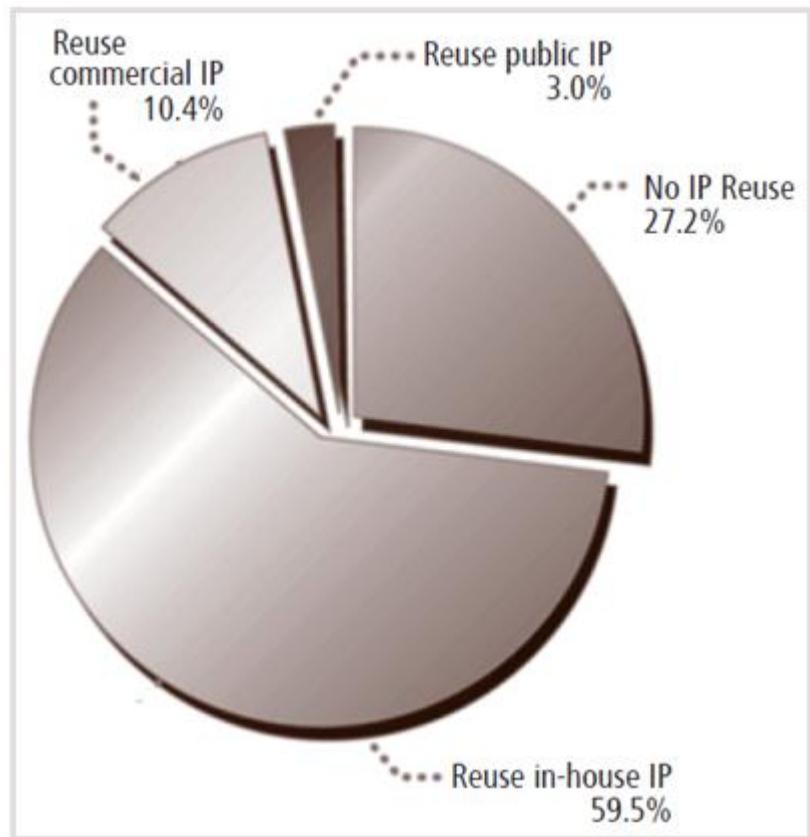
FPGA's

- ▶ ¿Por qué es útil usar IP cores?
- ▶ Los diseños actuales con FPGA son tan grandes y complejos que sería impráctico crear cada porción del diseño desde cero.
- ▶ Una solución a esto es reutilizar bloques funcionales existentes y dedicar la mayor parte del tiempo en crear las nuevas porciones del diseño que lo hacen diferente de otros.
- ▶ Existen tres fuentes principales de IPs:
 - ▶ Fabricantes de FPGA.
 - ▶ Proveedores de IPs externos.
 - ▶ Bloques creados en diseños previos (re-utilización).



FPGA's

- ▶ Con respecto a su impacto comercial,
 - ▶ El mercado de los IP cores comerciales sigue siendo bajo.
 - ▶ El porcentaje tan bajo de utilización de HW de dominio público contrasta con la popularidad del software libre.
 - ▶ Un alto porcentaje de diseñadores no usan IP cores debido a que el hardware tipo IP es normalmente difícil de modificar, no está bien documentado y es aún más difícil de verificar.
 - ▶ La mayoría de los diseñadores realiza o adapta su propio IP core.



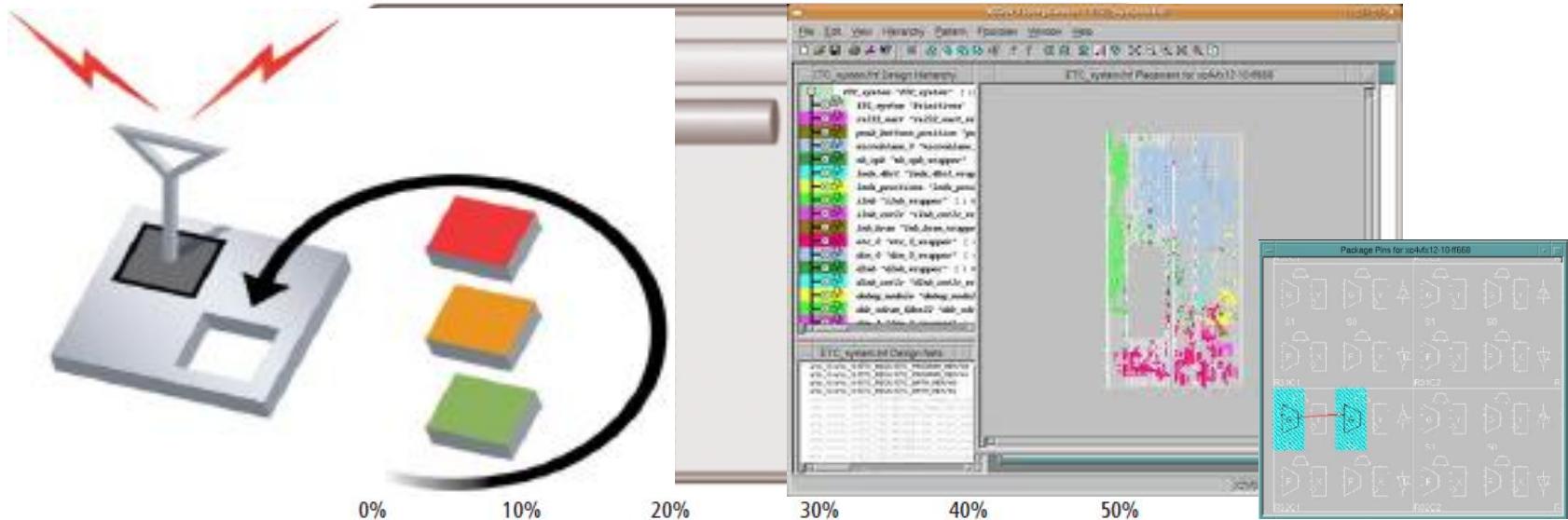
FPGA´s

- ▶ Fabricantes de FPGAs



FPGA's

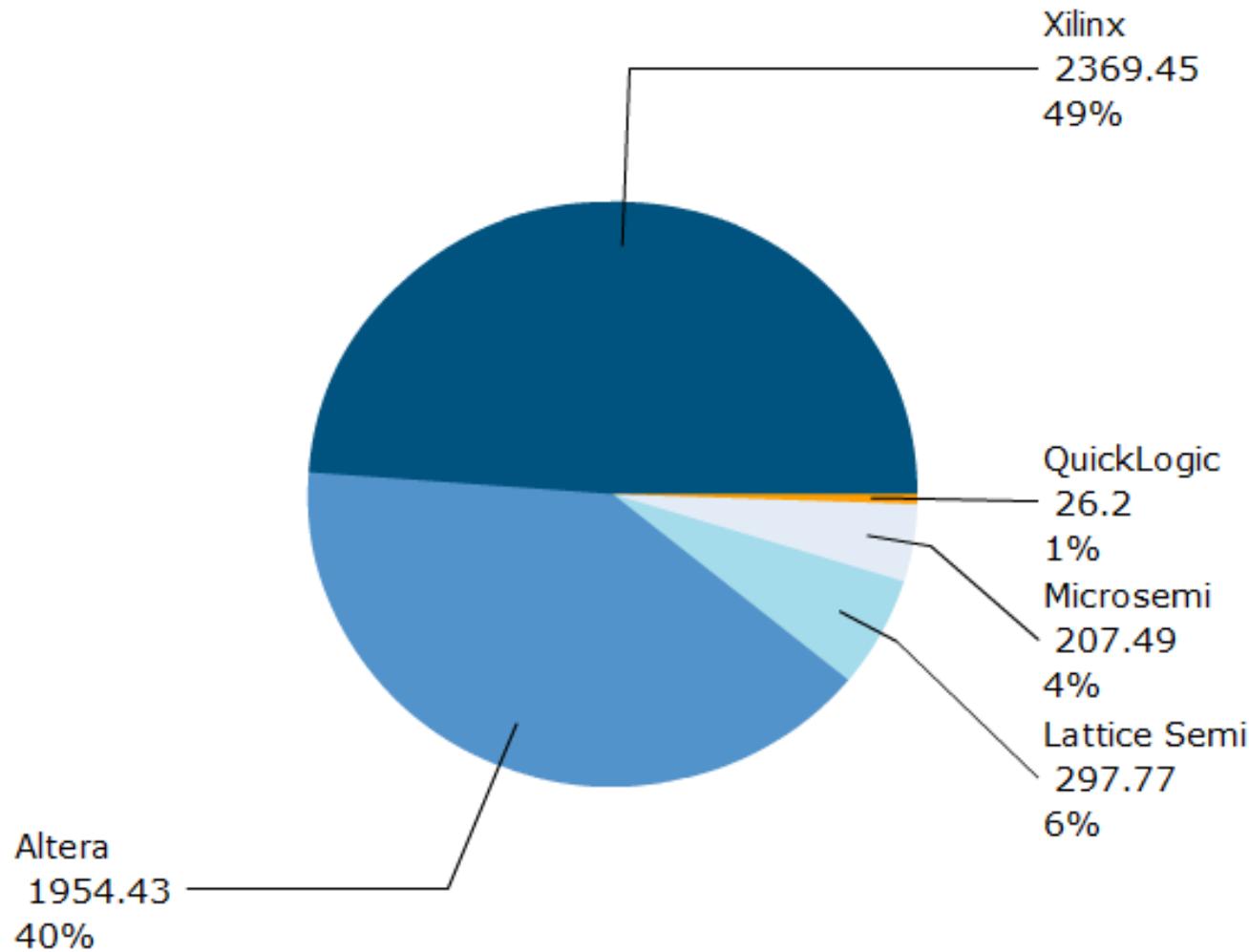
► Mercado actual de los FPGAs.



- Xilinx ocupa desde hace varios años el primer puesto
- Además, esta compañía es el 3er fabricante de ASICs del mundo, por detrás de IBM y NEC.
- Diferencias con Altera:
 - Xilinx sigue apostando por la reconfiguración dinámica mientras que Altera la ha descartado totalmente.
 - Otra diferencia es el nivel de las herramientas. Históricamente, Xilinx permite trabajar a un nivel más bajo que Altera.

FPGA's

- ▶ Comparativa de ingresos de compañías fabricantes de FPGAs, en millones de dólares (2011).



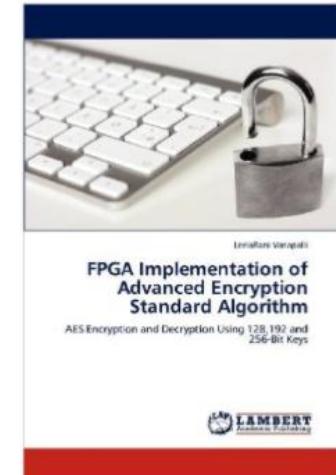
FPGA's

- ▶ Aplicaciones FPGA's:
 - ▶ Procesamiento de datos.

Compresión de datos

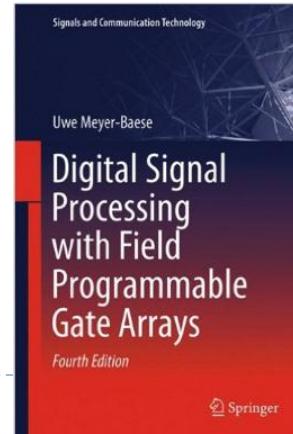
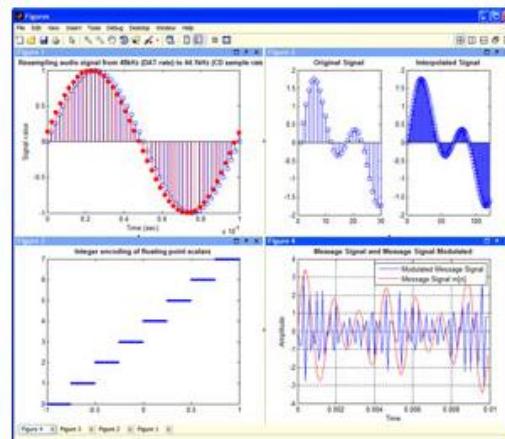
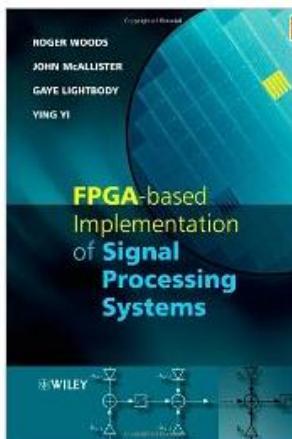


Encriptación de datos



LAMBERT Academic Publishing

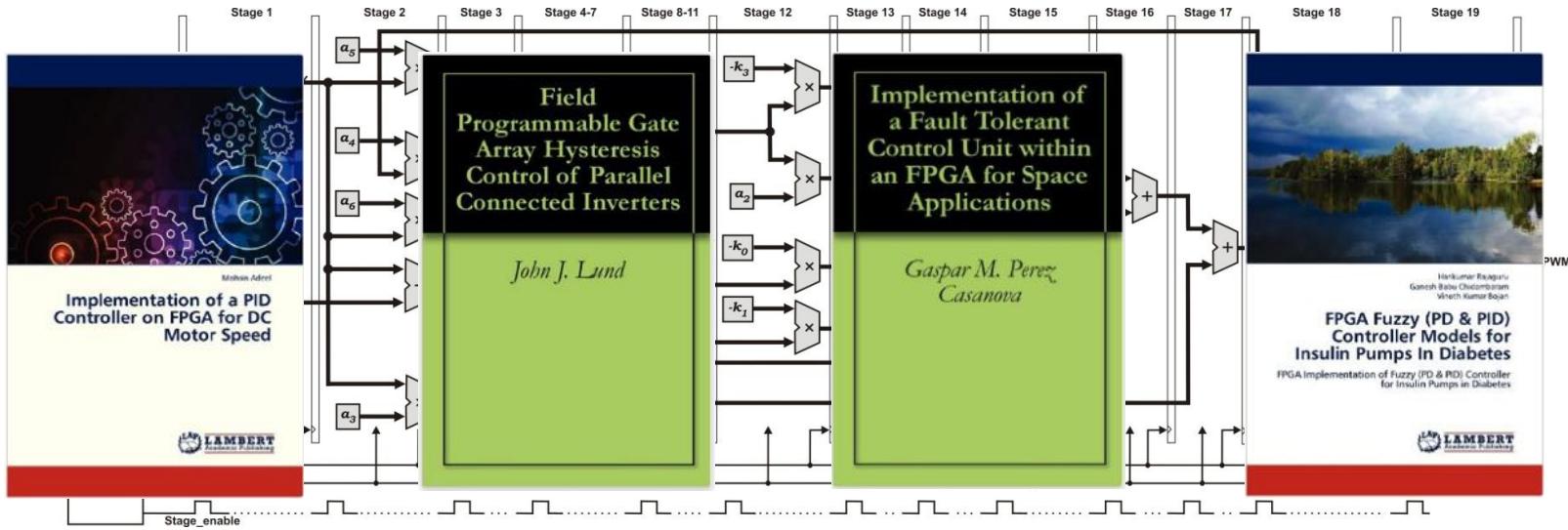
Procesamiento de señales



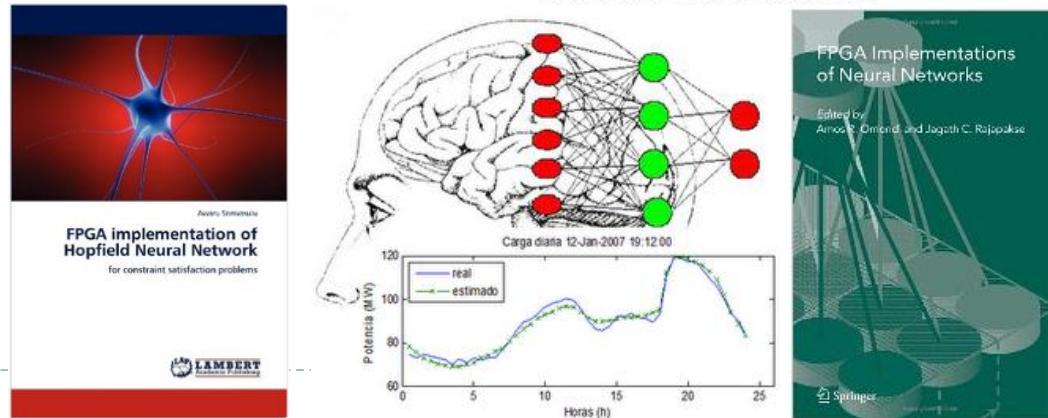
Springer

FPGA's

- ▶ Aplicaciones FPGA's:
 - ▶ Sistemas altamente concurrentes y alta necesidad de computo.



Redes Neuronales Artificiales



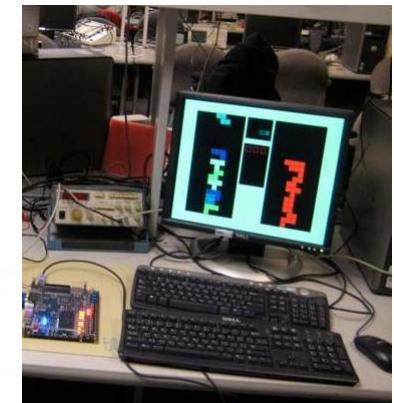
FPGA's

- ▶ Aplicaciones FPGA's:
 - ▶ Juegos de video.

FPGA Altera Cyclone II EP2C35F672C6



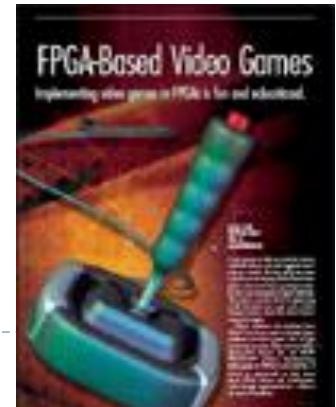
Emulador de video juegos FP



Fighting Video-game

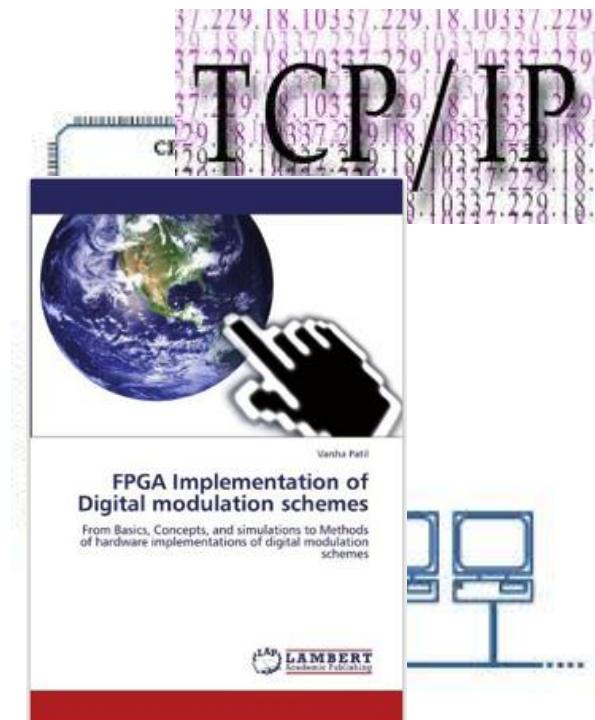
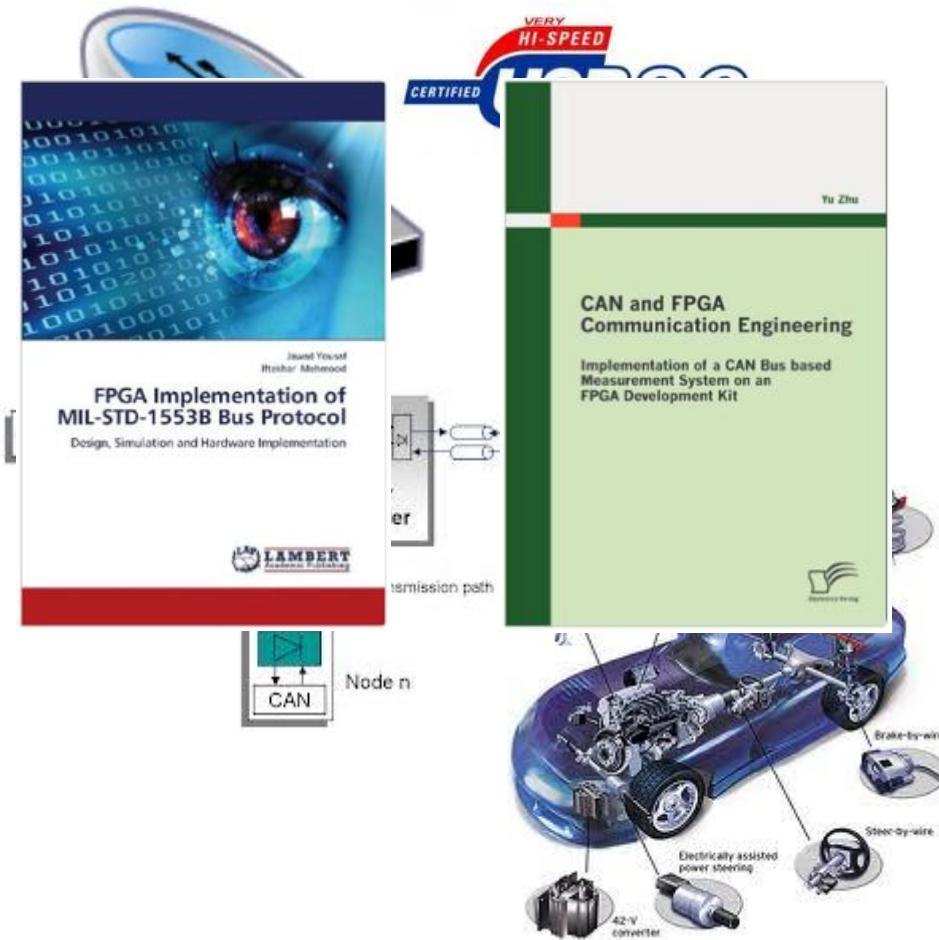


Nintendo Entertainment System cloned in a FPGA



FPGA's

- ▶ Aplicaciones FPGA's:
 - ▶ Protocolos de comunicación.



FPGA's

- ▶ Aplicaciones FPGA's:
 - ▶ Diseño de Procesadores.



FPGA's

- ▶ Aplicaciones
- ▶ Lógica de al



Imagenes y video



Jiwesh Kumar
Pranjal Pandey

FPGA Implementation of MPEG-2 Video decoder

FPGA Implementation of base layer MPEG-2
Video decoder

Design & Implementation of Digital Image Processing using FPGA

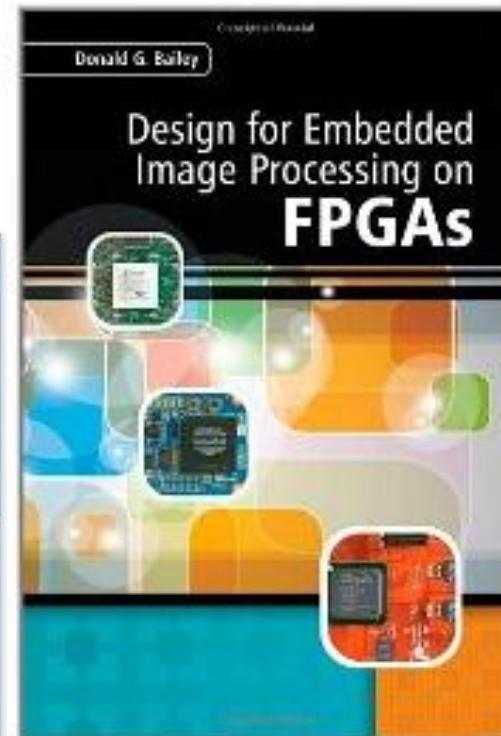
FPGA-based digital image processing

LAP LAMBERT
Academic Publishing



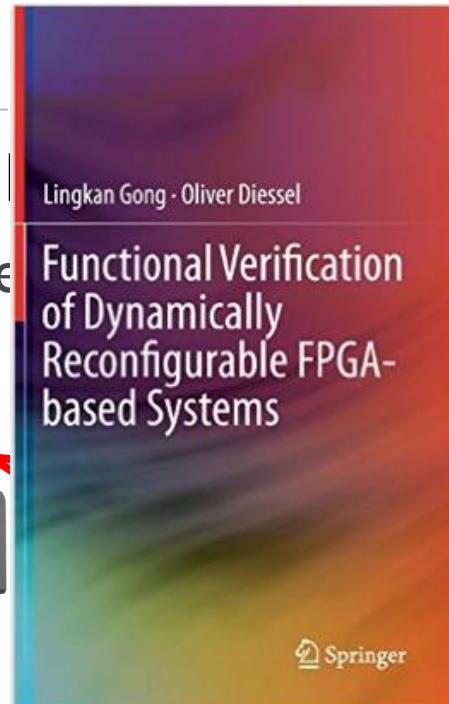
Khaled Benkrid
AbdSamad Benkrid

Software and Hardware Architectures for FPGA- based Image Processing

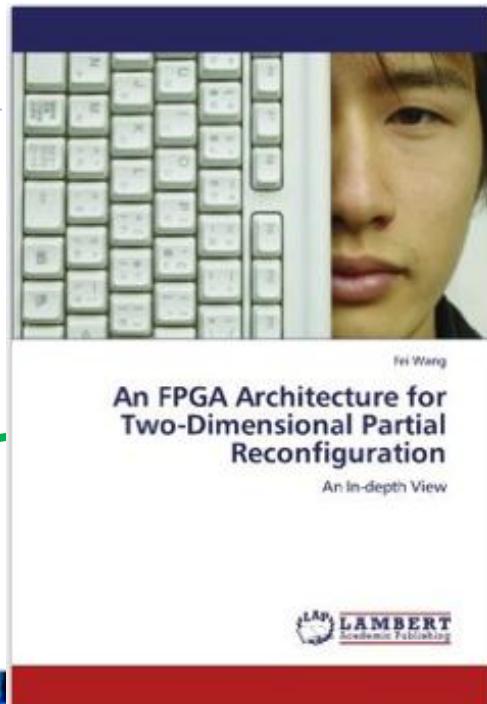


FPGA's

- ▶ Aplicaciones
- ▶ Hardware reconfigurable



ámicamente.



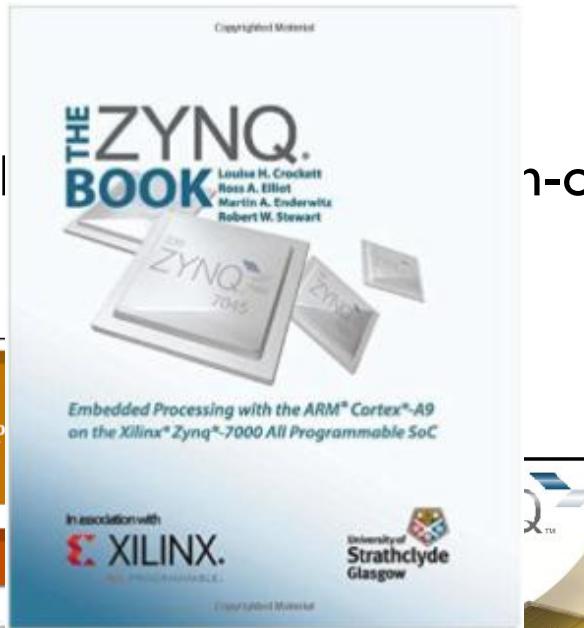
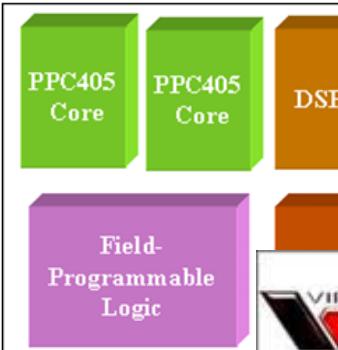
Luis



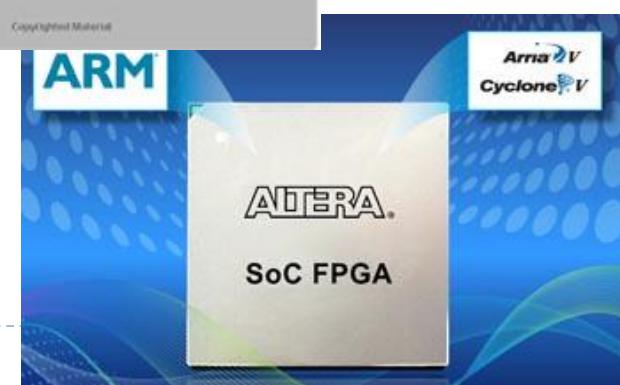
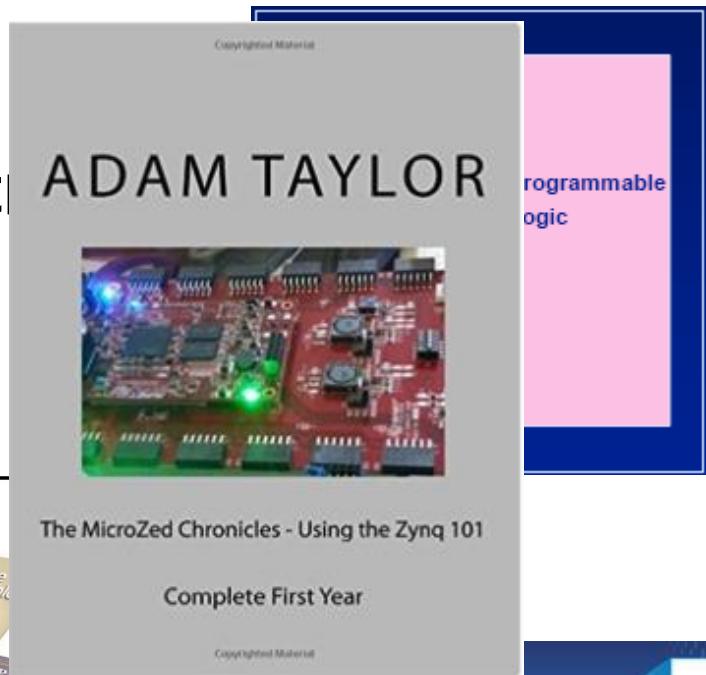
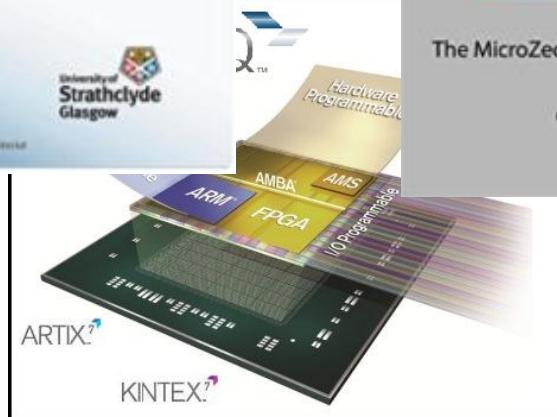
FPGA's

- ▶ Aplicaciones FPGA's:
 - ▶ Sistemas en un sólo chip (Lógica reconfigurable + procesador).

Field-

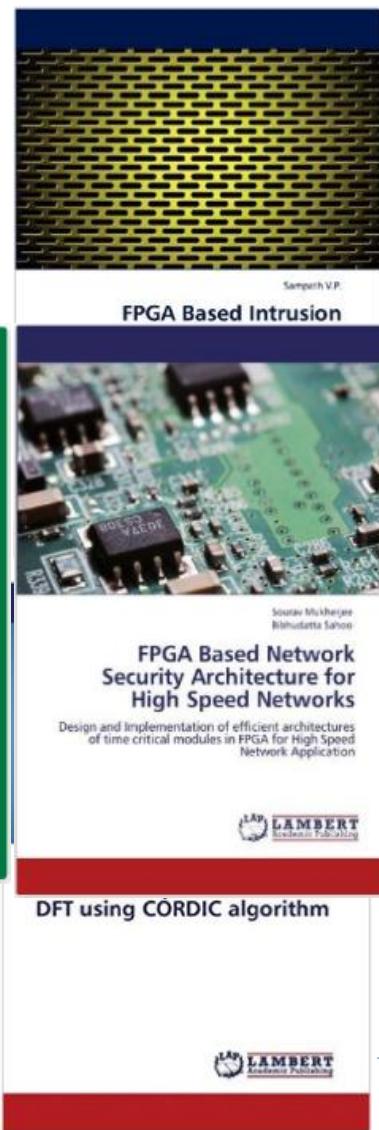
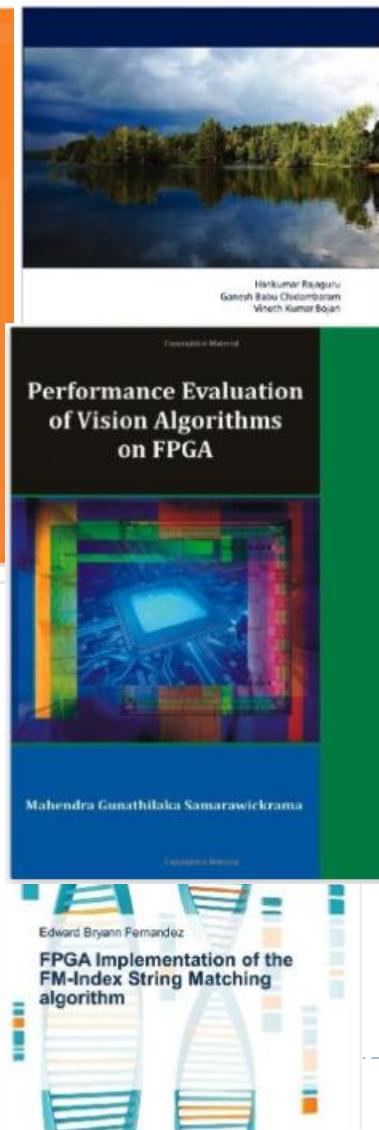
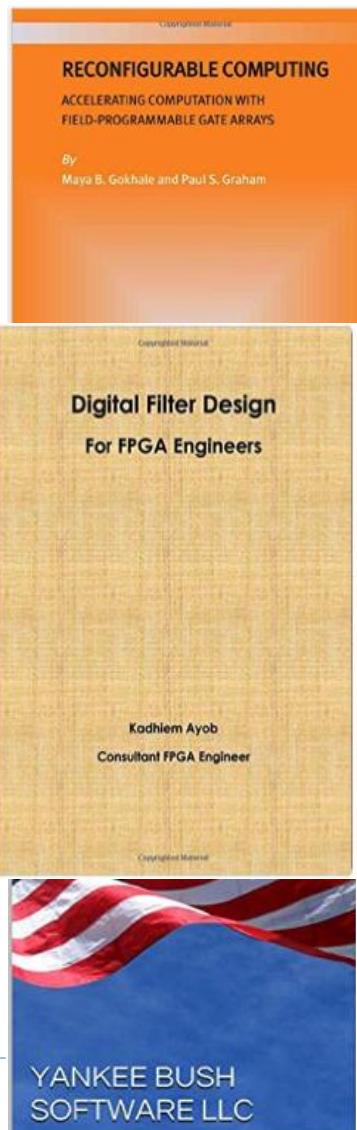


on-Chip

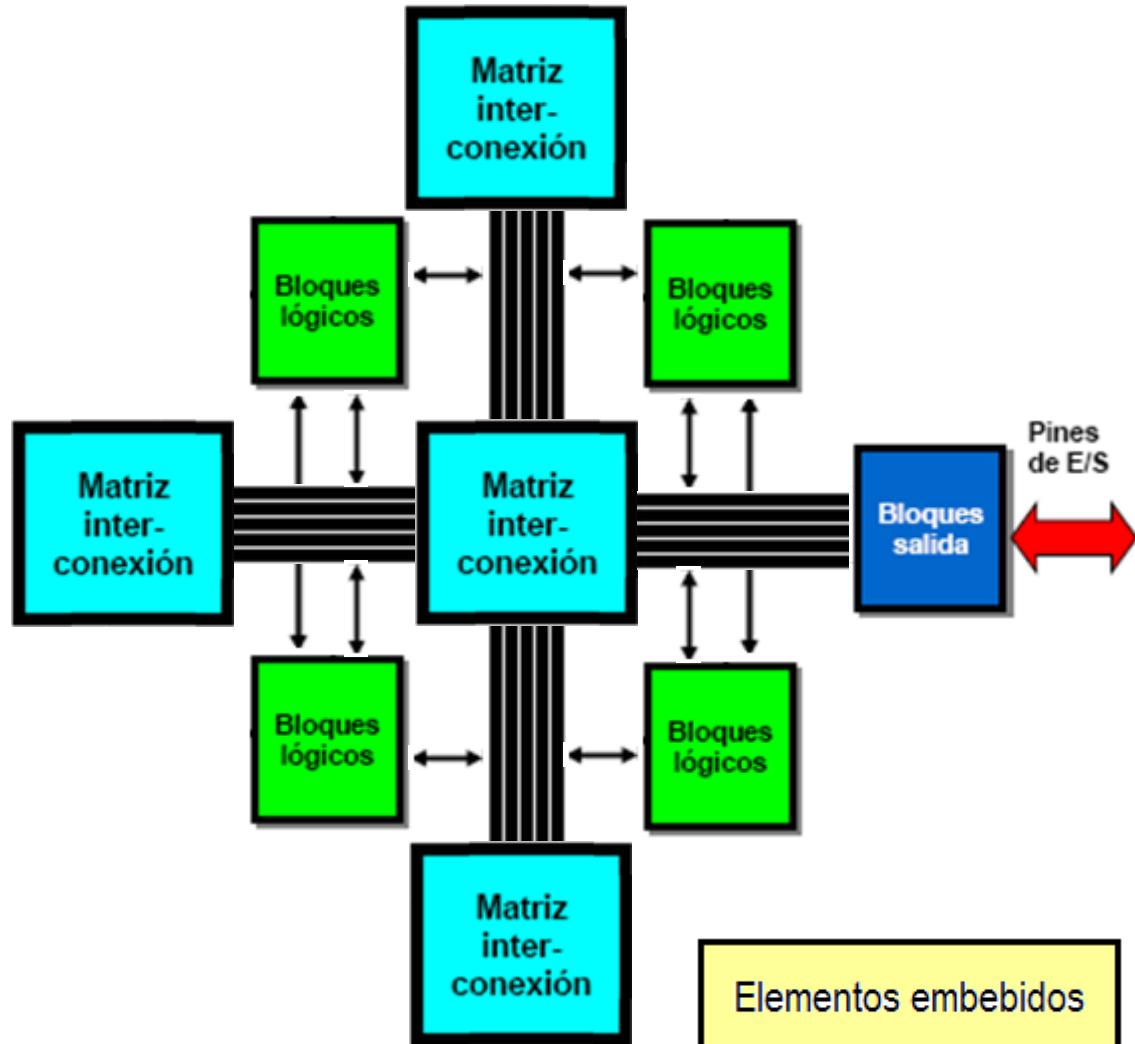


FPGA's

► Aplicaciones FPGA's:



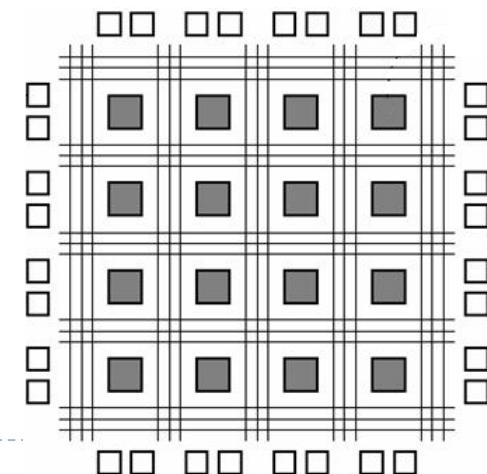
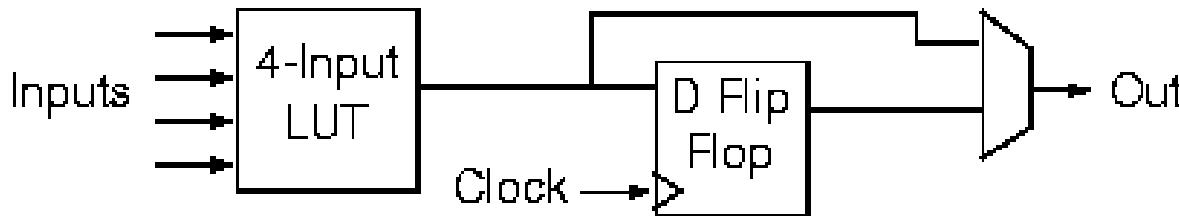
Estructura general de un FPGA



FPGA's

▶ Bloques lógicos (CLB, Configurable Logic Block)

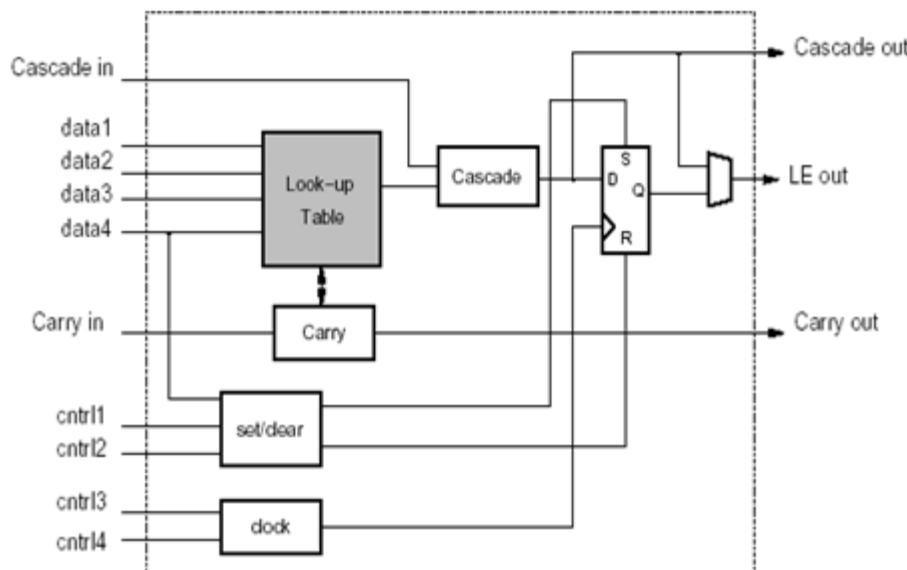
- ▶ Elemento principal de un FPGA, utilizados para implementar las funciones lógicas.
- ▶ Distribución matricial.
- ▶ Compuestos principalmente por tablas de búsqueda (LUT, *Look-up table*), combinadas con biestables y elementos de control.
- ▶ La arquitectura de los CLBs varia entre familia de dispositivos
 - ▶ Sin LUT's se trata de un FPGA de granulidad fina
 - ▶ Con LUT's es un FPGA de granulidad gruesa.



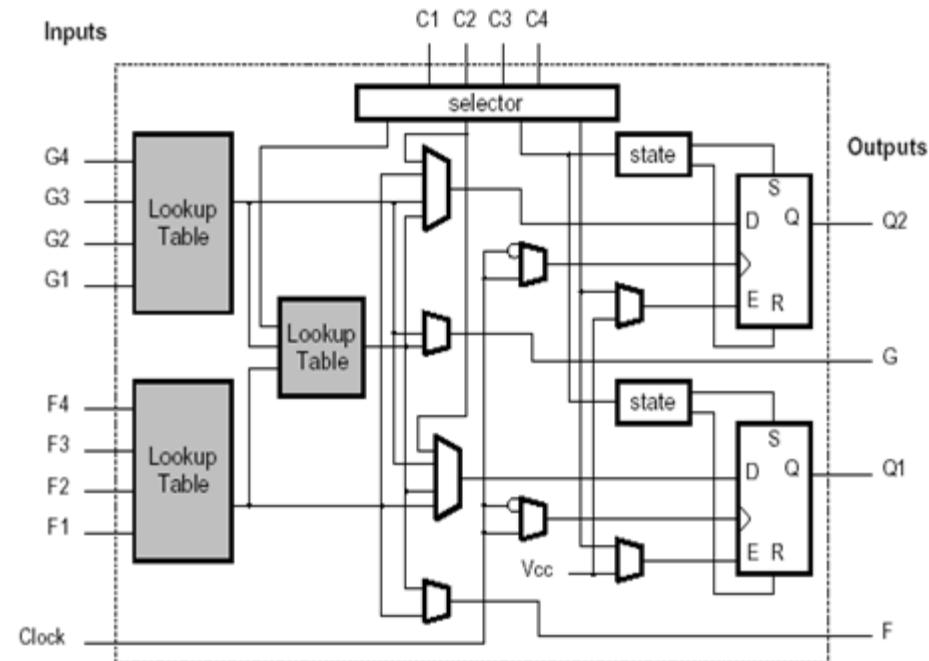
FPGA's

- ▶ Bloques lógicos basados en LUT's.

MAX128, Altera



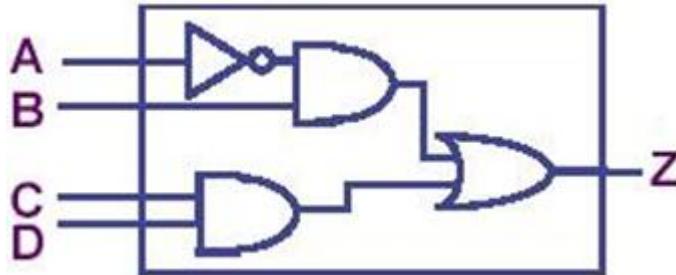
XC4000, Xilinx



FPGA's

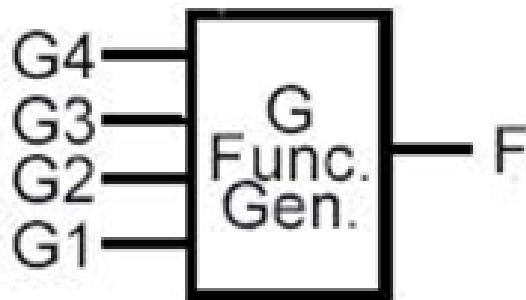
► LUT (*Look-Up Table*)

Tablas de búsqueda, Generadores de funciones → RAM



$$Z \leq (C \text{ and } D) \text{ or } (\text{not } A \text{ and } B);$$

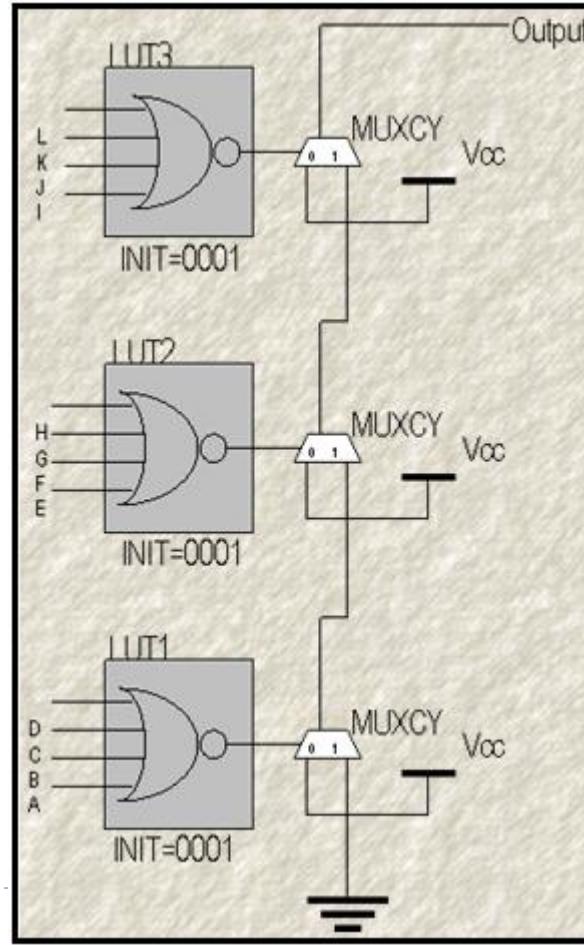
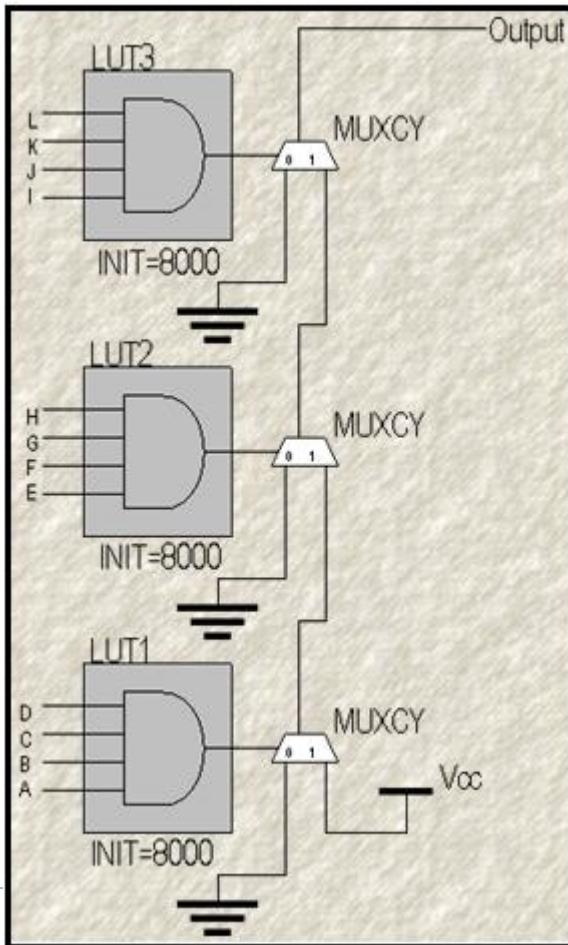
A	B	C	D	Z
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
.
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1



FPGA's

► LUT (*Look-Up Table*)

Tablas de búsqueda, Generadores de funciones → RAM



FPGA's

▶ Funciones independientes en LUTs.

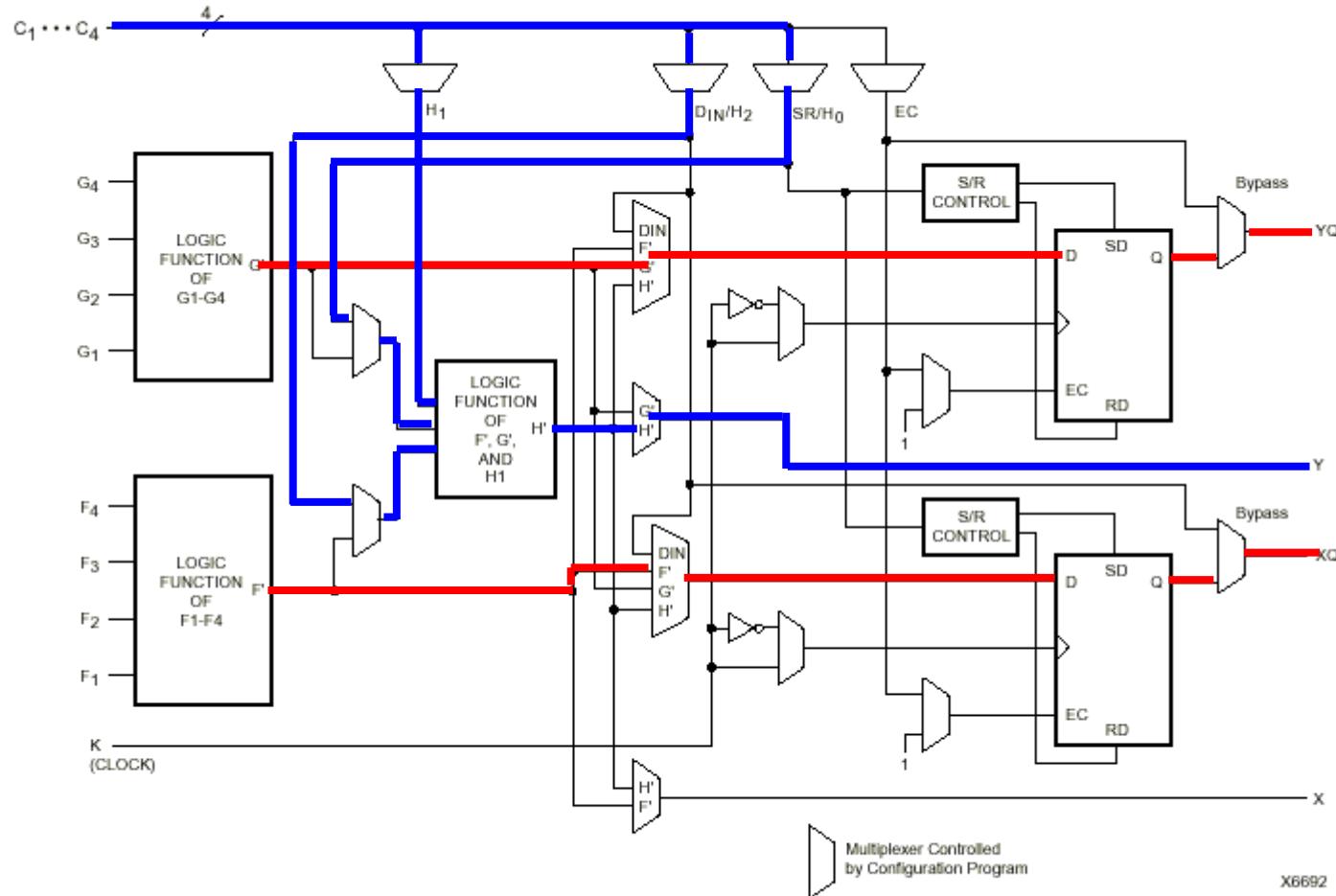


Figure 1: Simplified Block Diagram of XC4000 Series CLB (RAM and Carry Logic functions not shown)

FPGA's

► Agrupación de LUTs.

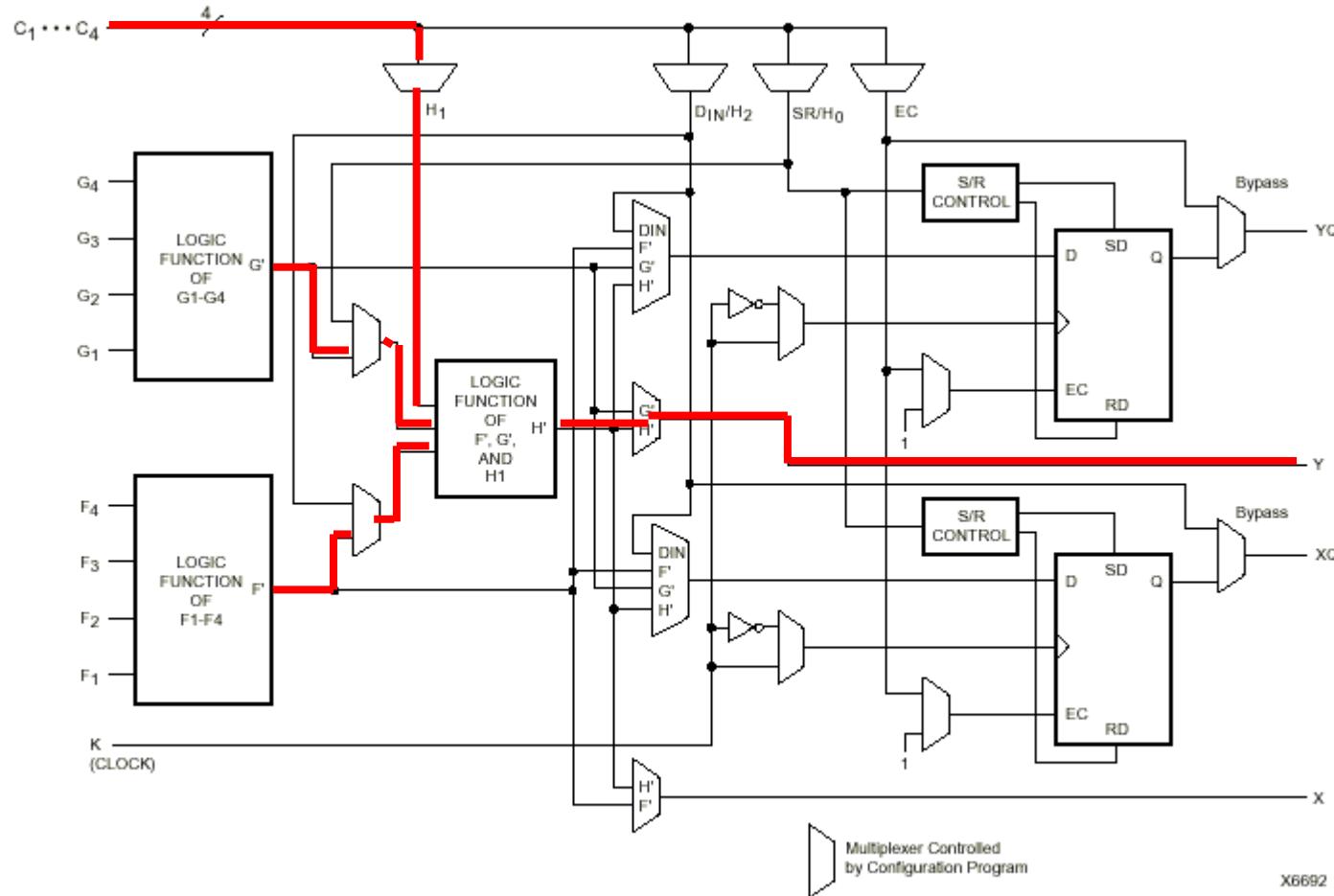
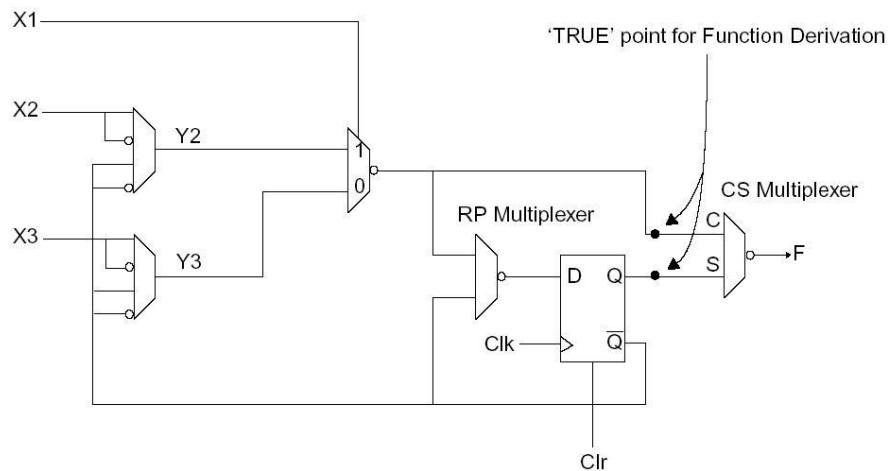


Figure 1: Simplified Block Diagram of XC4000 Series CLB (RAM and Carry Logic functions not shown)

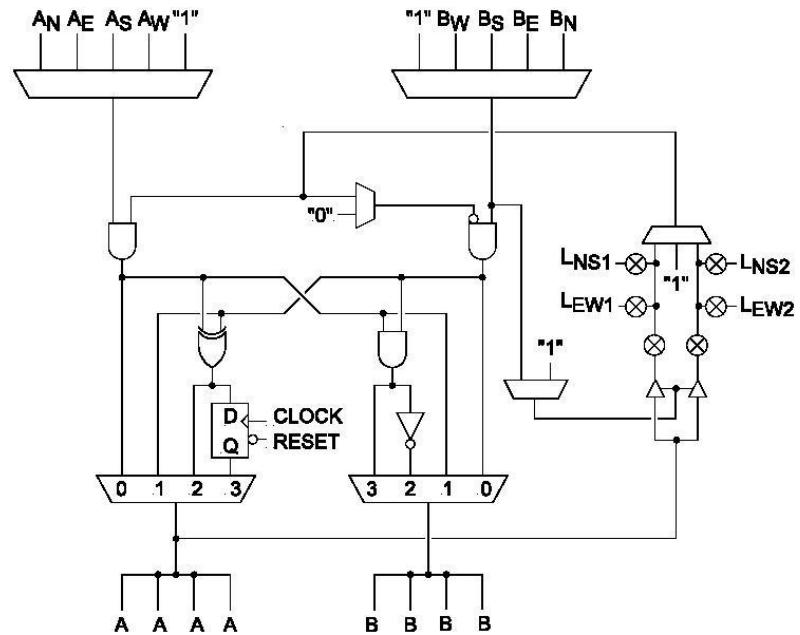
FPGA's

► Bloque lógico basado en multiplexores:

XC6200, Xilinx



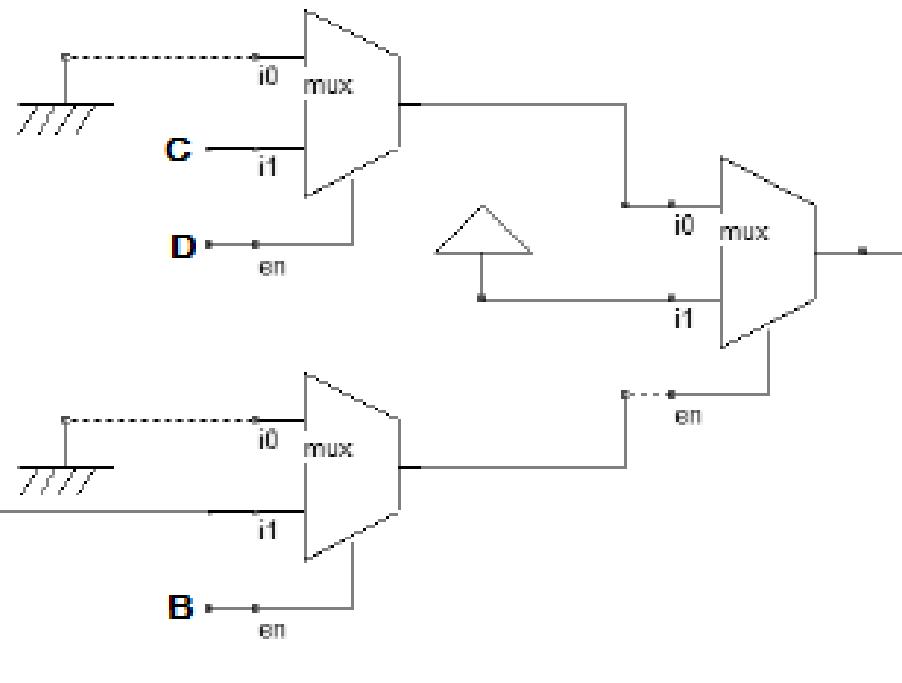
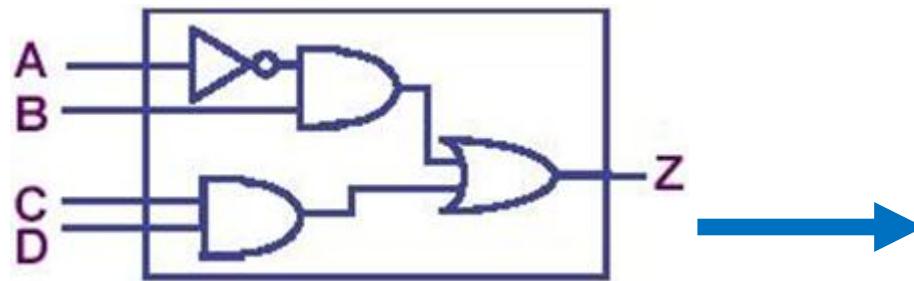
AT600, Atmel



FPGA's

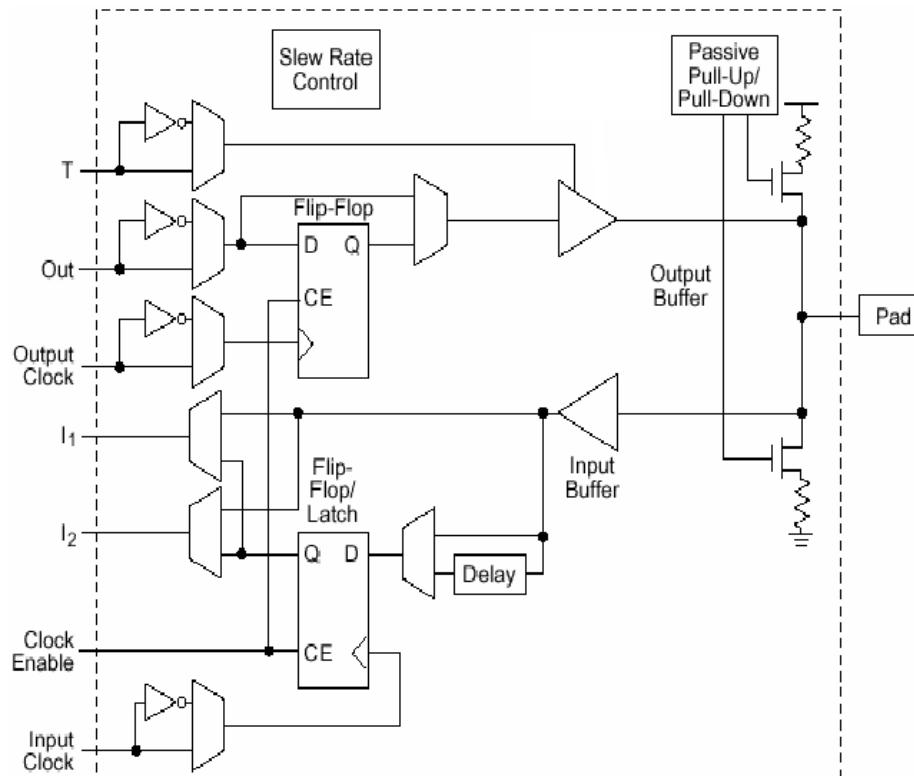
- ▶ Bloque lógico basado en multiplexores:

$$Z \leq (C \text{ and } D) \text{ or } (\text{not } A \text{ and } B);$$



FPGA's

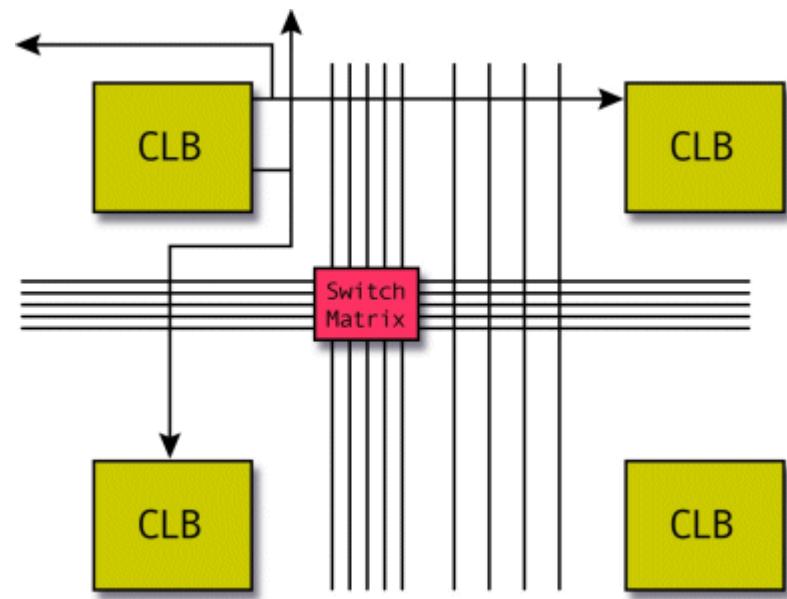
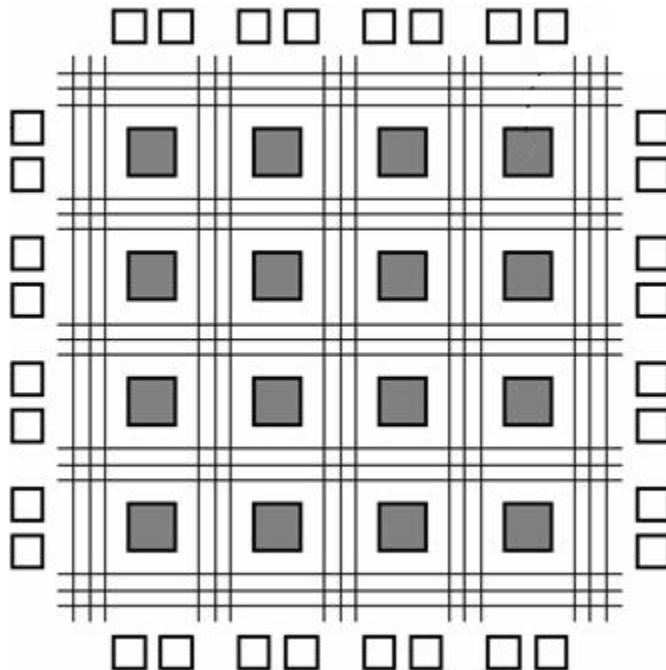
- ▶ Bloque de entrada/salida.
 - ▶ Tiene por función establecer el enlace entre los bloques lógicos y las terminales de entrada/salida.



FPGA's

▶ Recursos de interconexión.

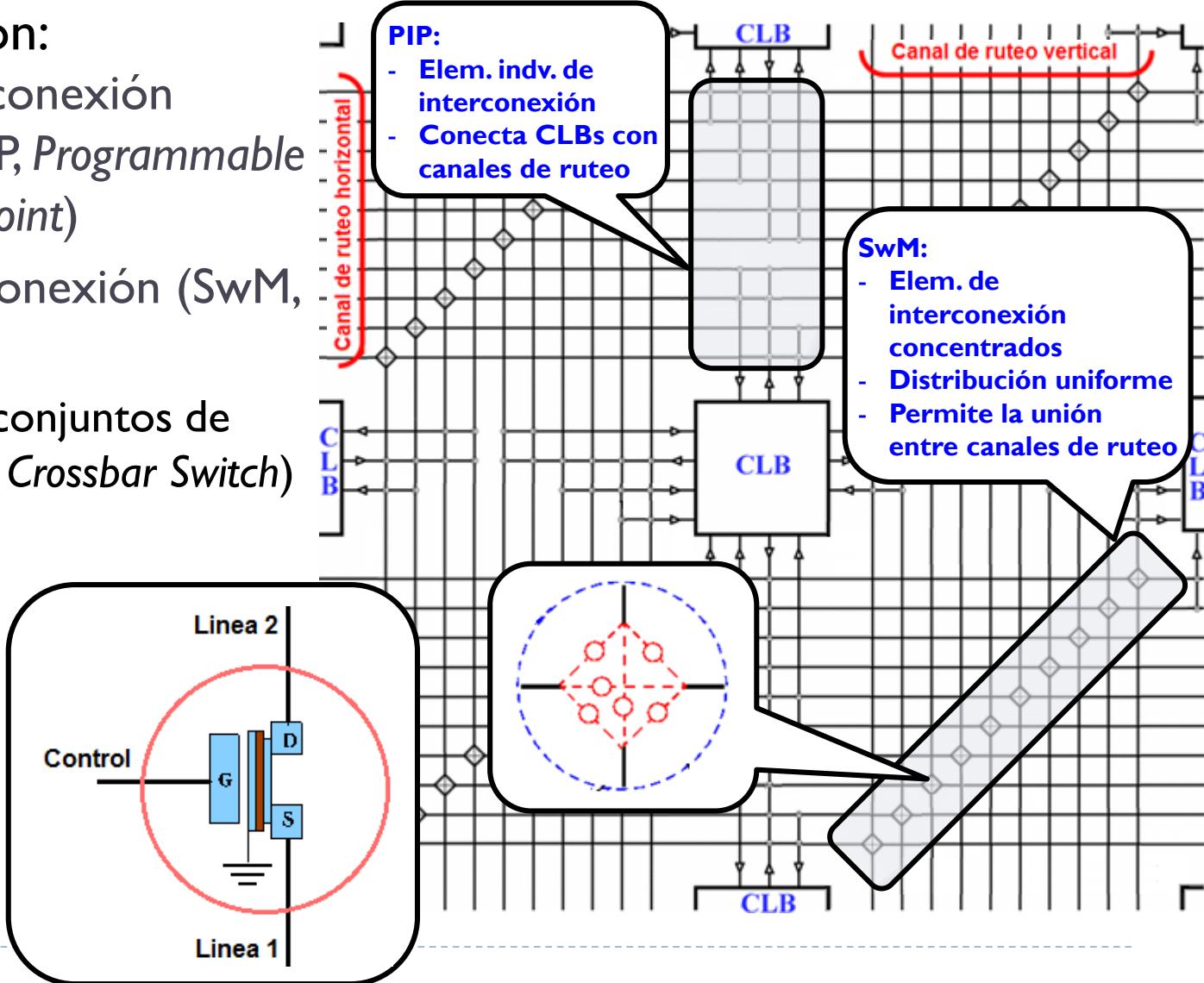
- ▶ Principalmente, están constituidas por líneas horizontales y verticales que recorren los espacios existentes entre los CLBs.
- ▶ Representan un conjunto de recursos que permiten conectar los CLBs entre sí y estos con los bloques I/O.



FPGA's

- ▶ Elementos adicionales que forman parte de los recursos de interconexión son:

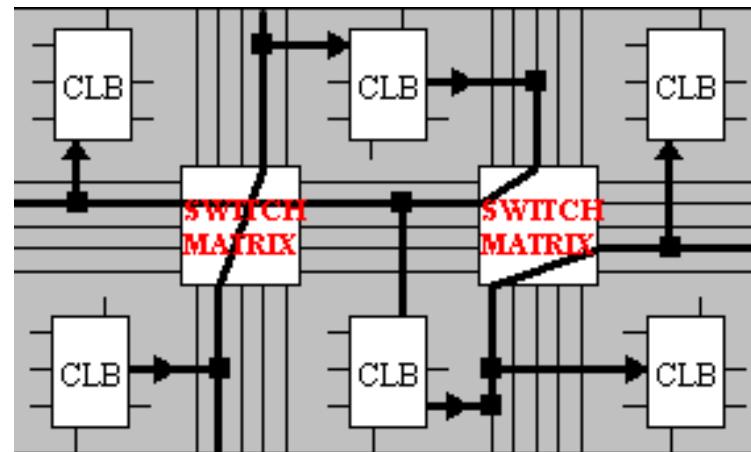
- ▶ Puntos de Interconexión Programable (PIP, Programmable Interconnection Point)
- ▶ Matriz de interconexión (SwM, Switch Matrix).
 - ▶ Formadas por conjuntos de PIPs (FCW, Full Crossbar Switch)



- ▶ Los PIP son “interruptores programables”

FPGA's

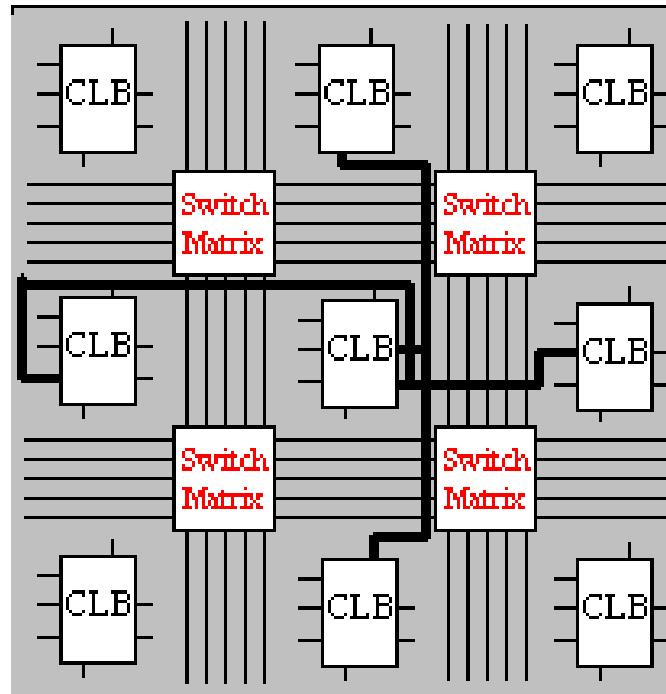
- ▶ Además, existen diferentes tipos de líneas de interconexión:
 - ▶ Líneas de propósito general
 - ▶ Líneas directas
 - ▶ Líneas largas
- ▶ *Líneas de propósito general*
 - ▶ Conjunto de líneas horizontales y verticales conectadas a una matriz de interconexión.



FPGA's

- ▶ *Líneas directas*

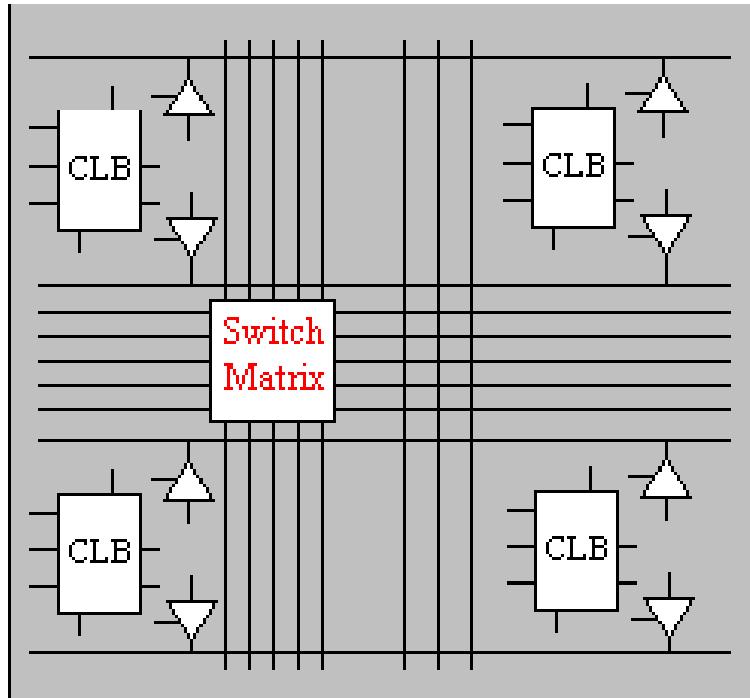
- ▶ Líneas de conexión directa entre bloques, sin tener que pasar por ninguna matriz de interconexión.



FPGA's

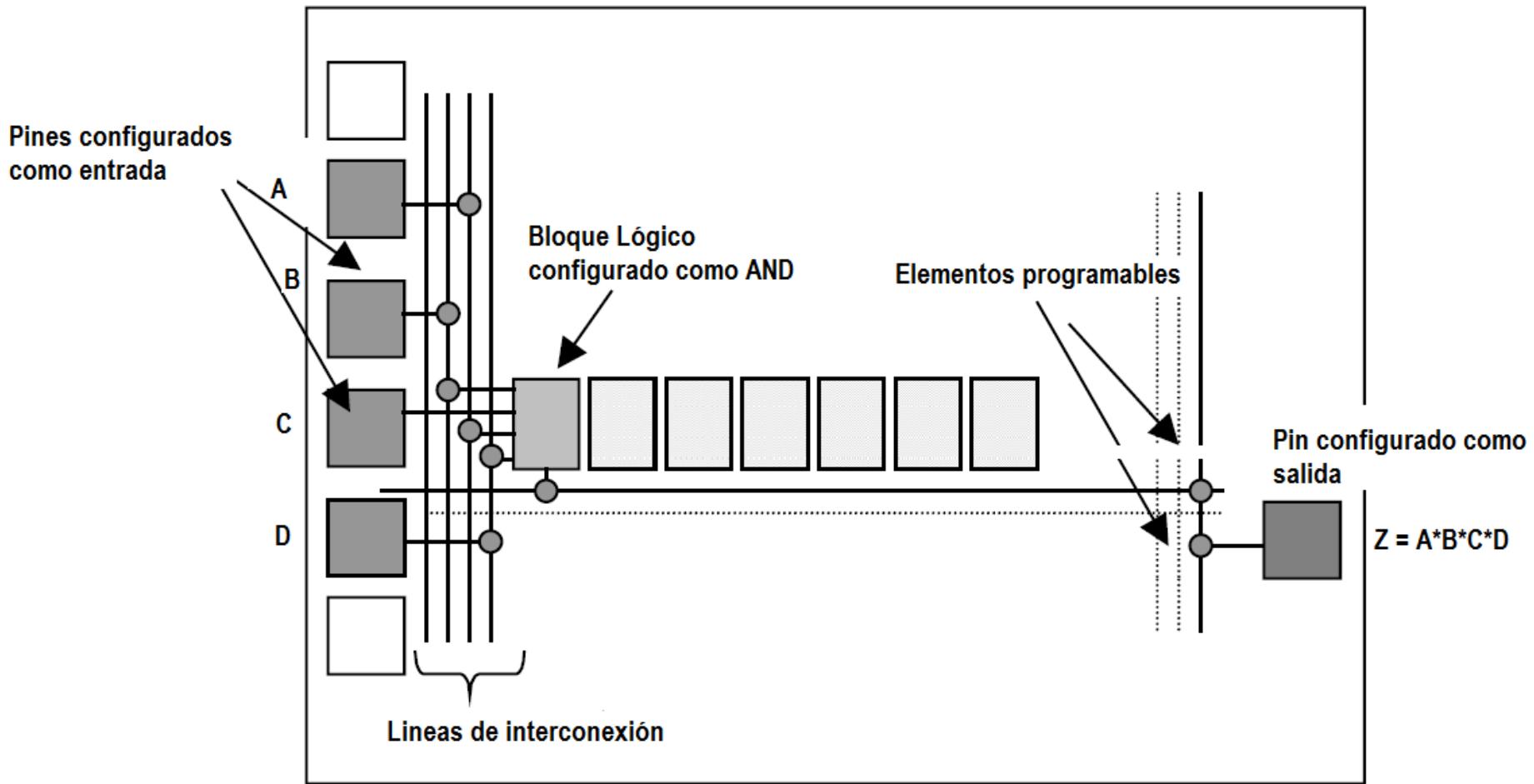
- ▶ *Líneas largas*

- ▶ Líneas de gran longitud que atraviesan al FPGA de arriba abajo y de izquierda a derecha.



FPGA's

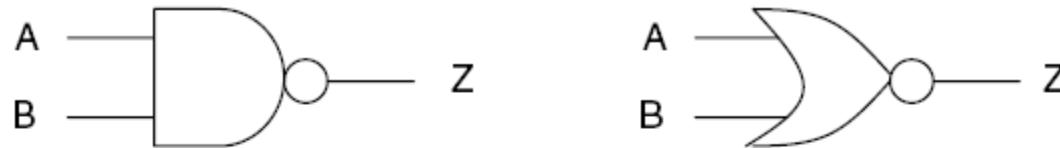
▶ Implementación física



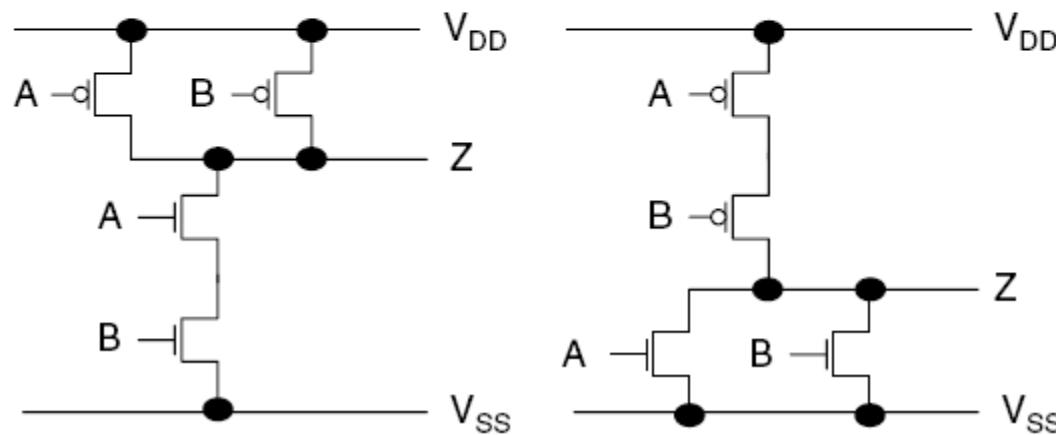
FPGA's

- ▶ Escala de integración
 - ▶ Un FPGA mide su escala de integración por el número de compuertas implementadas en este dispositivo o que puede implementar.

Simbolos lógicos de las compuertas NAND y NOR

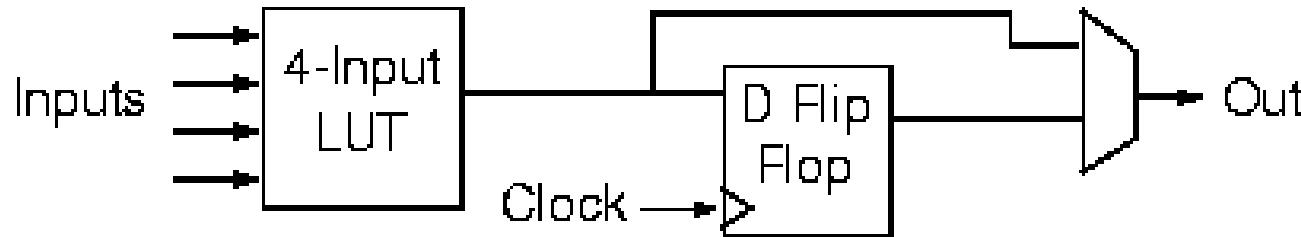


Esquemas de las compuertas NAND y NOR con base en transistores



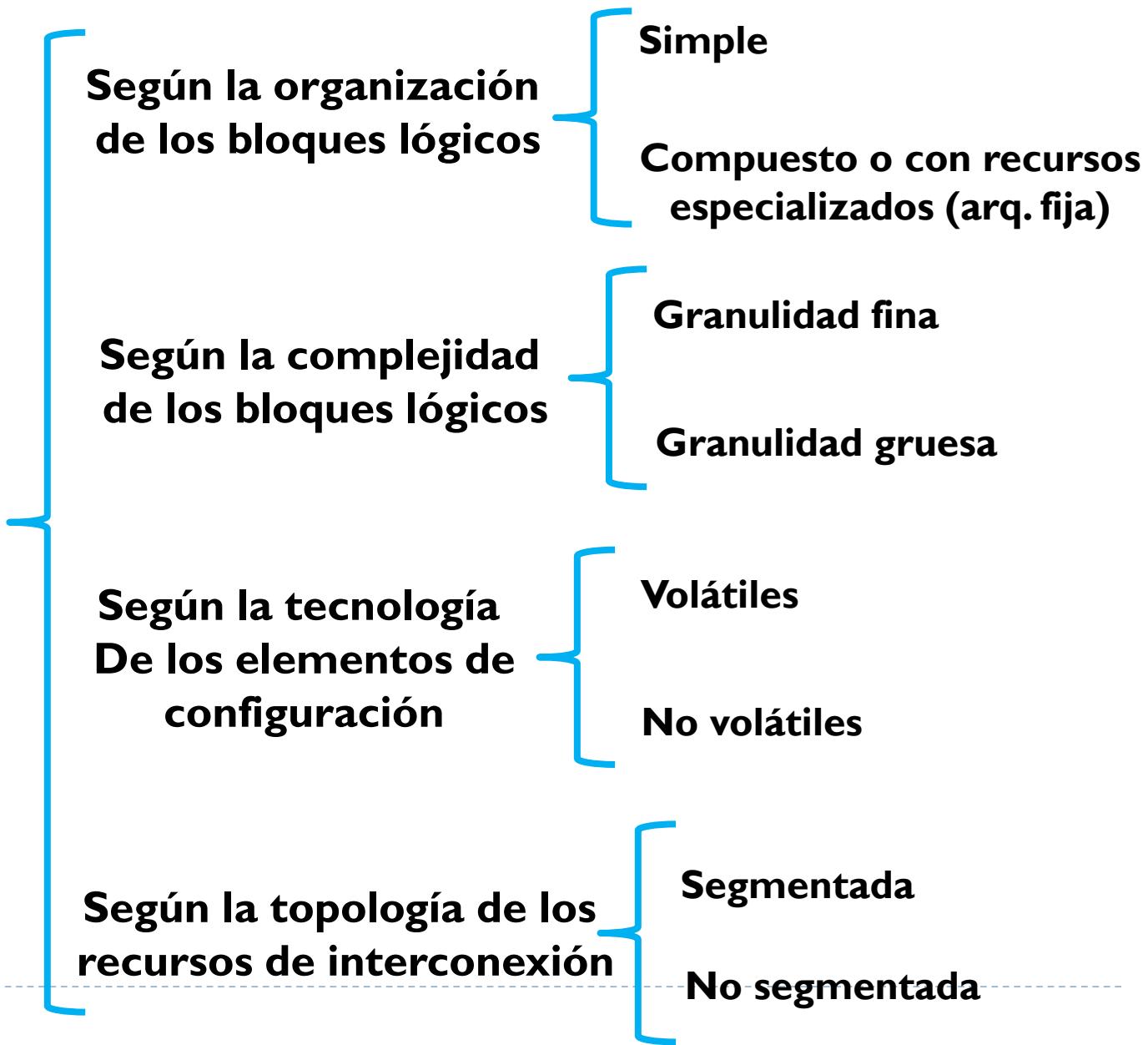
FPGA's

- ▶ Una alternativa para medir la escala de integración de un FPGA es el número de “celdas lógicas” que lo integran.



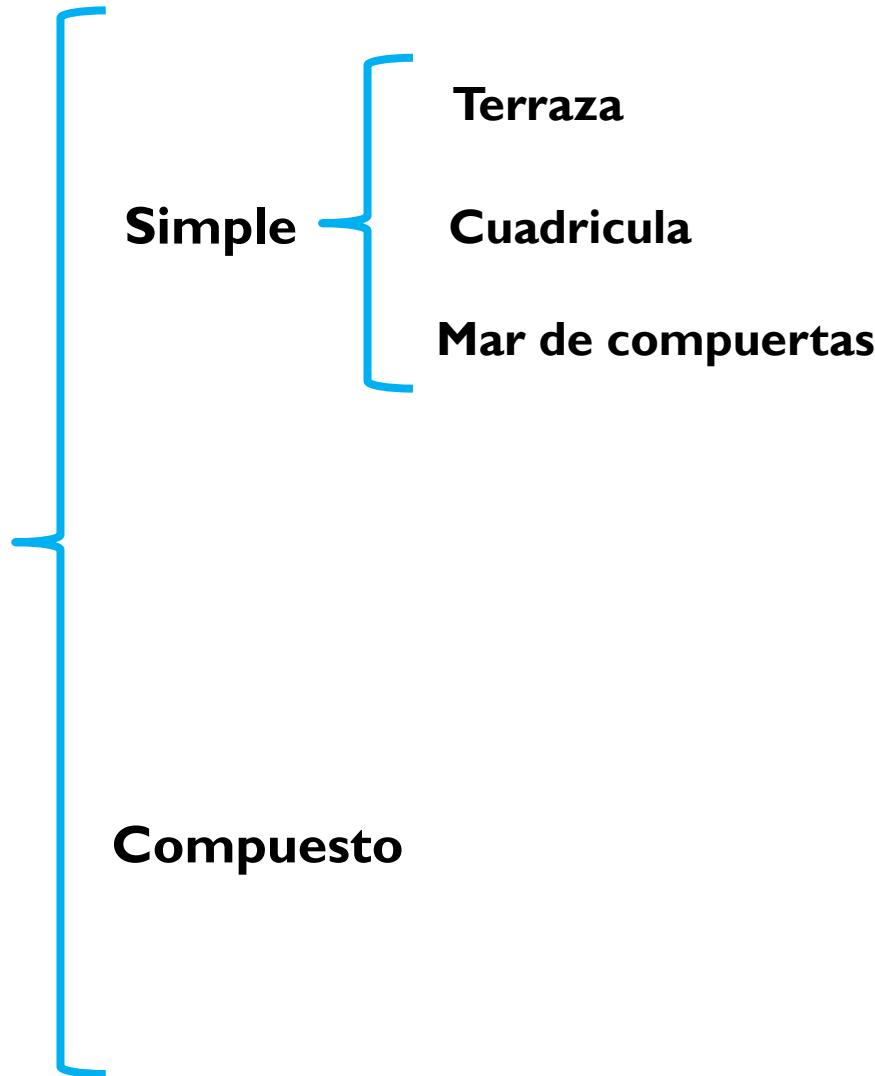
FPGA's

Clasificación de los FPGA's



FPGA's

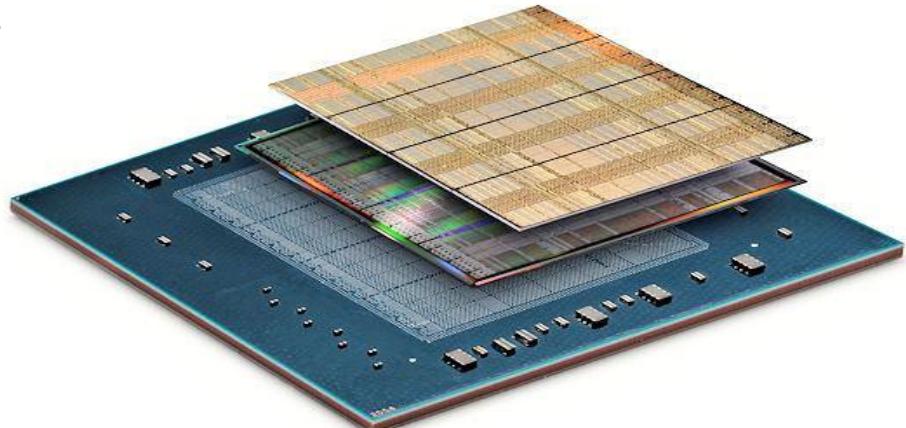
**Clasificación de los
FPGA's según la
organización de
los bloques lógicos**



FPGA's

▶ FPGA simple con organización tipo mar de compuertas.

- ▶ Bloques Lógicos distribuidos en filas y columnas.
- ▶ Los Recursos de interconexión vertical y horizontal se sitúan en capas diferentes.
- ▶ Resultado: mejor aprovechamiento del área del dispositivo, FPGA's de este tipo presentan una gran densidad de recursos lógicos y de recursos de interconexión.



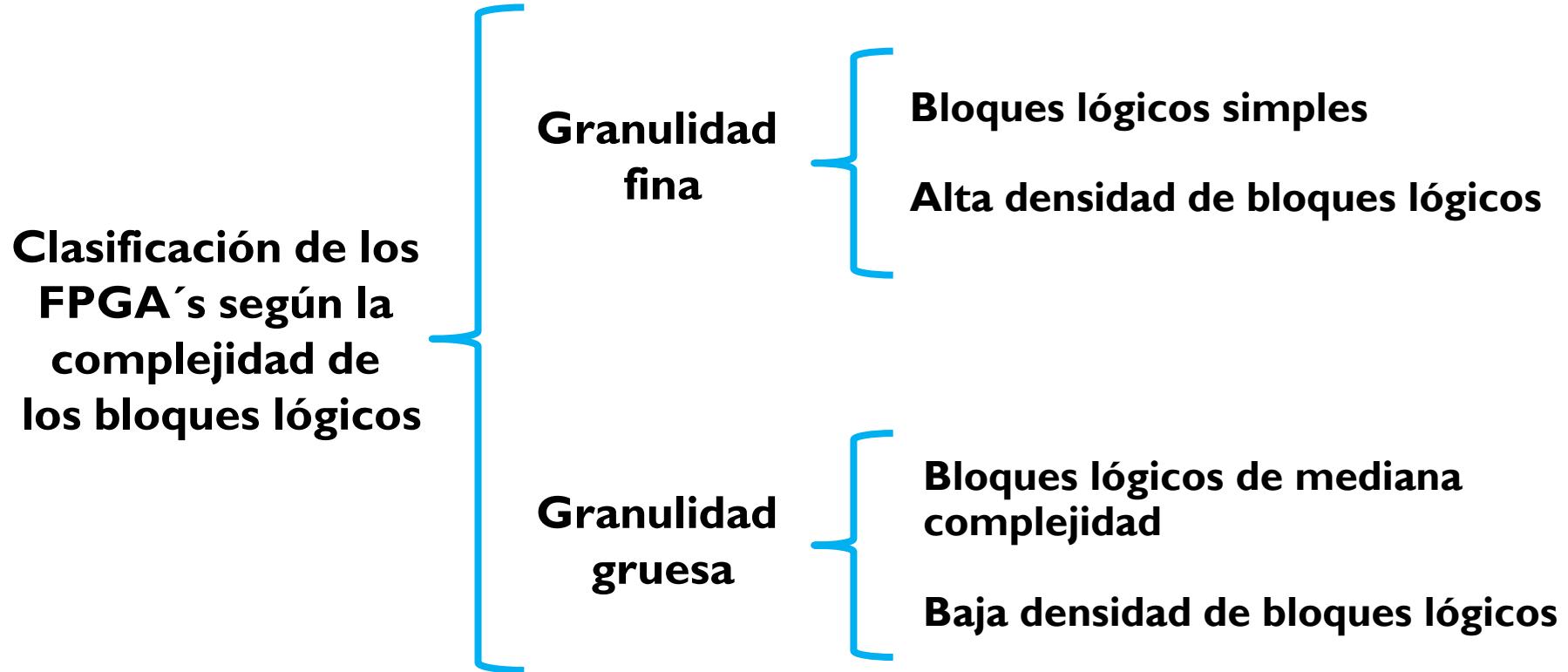
FPGA's

- ▶ **FPGA's compuestos.**
 - ▶ Este tipo de FPGA combina cualquier organización simple con recursos adicionales de arquitectura fija.
 - ▶ Implementa funciones específicas sin usar bloques lógicos.
 - ▶ Principalmente por:
 - ▶ Necesitar grandes recursos del FPGA para implementar estas funciones.
 - ▶ Se trata de funciones comúnmente usadas en una gran cantidad de aplicaciones.

FPGA's

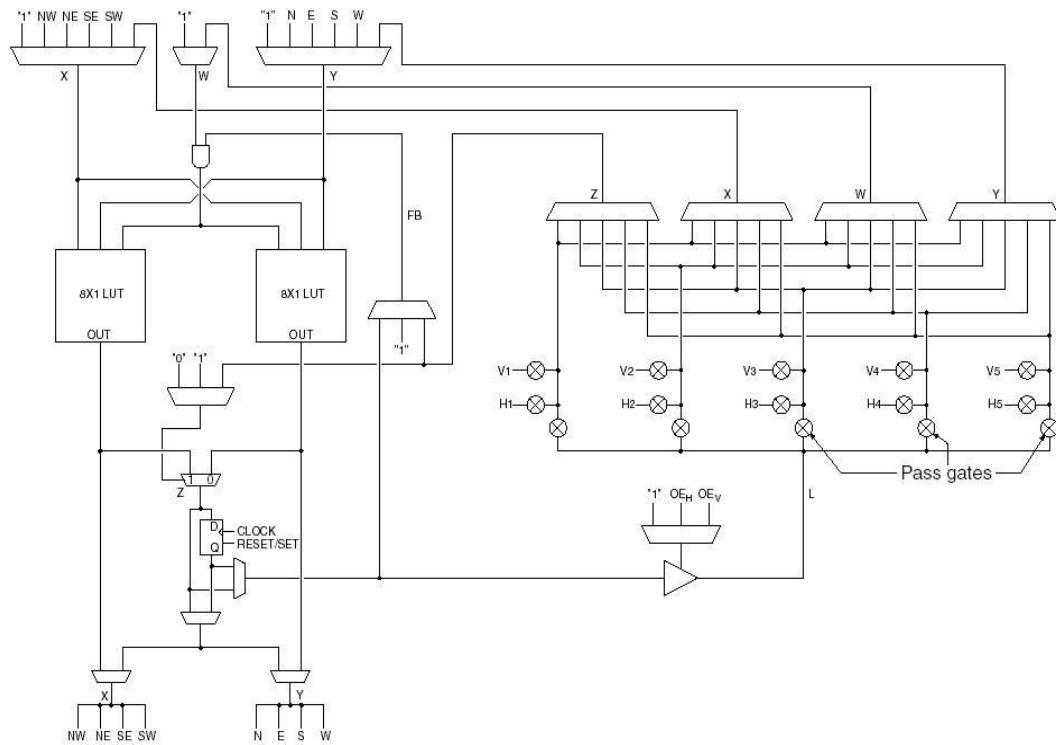
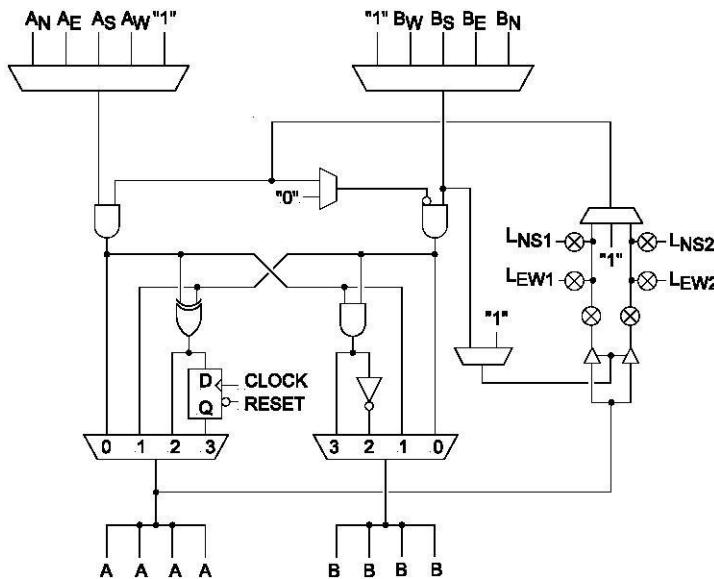
- ▶ **FPGA's compuestos.**
 - ▶ Recursos lógicos de arquitectura fija de aplicación general.
 - ▶ Estos bloques buscan optimizar el desempeño de las aplicaciones a través de la inclusión de elementos frecuentemente requeridos en diversas aplicaciones (Memorias RAM, DCMs).
 - ▶ Recursos lógicos de arquitectura fija de aplicación específica.
 - ▶ Dispositivos híbridos (Multiplicadores, controladores DDR, procesadores).

FPGA's



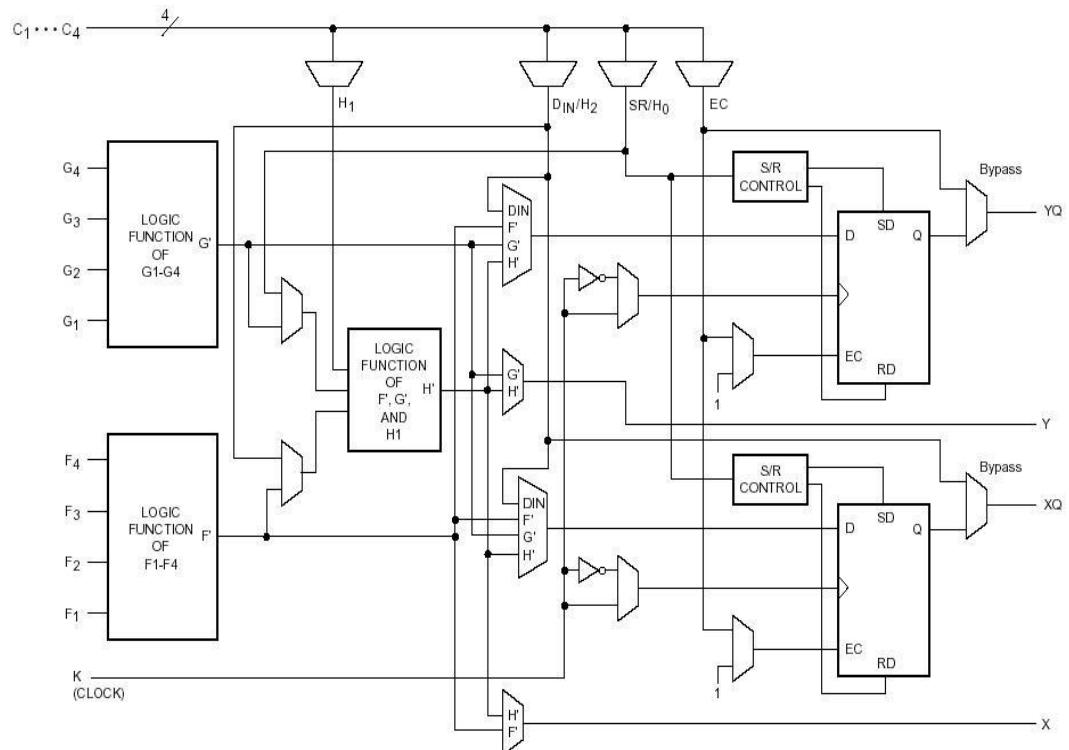
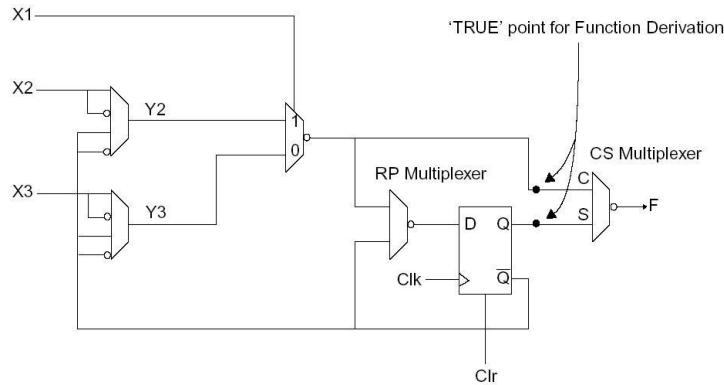
FPGA's

- ▶ AT600 Atmel, 6400 bloques lógicos.
- ▶ AT40K Atmel, 2300 bloques lógicos.



FPGA's

- ▶ XC6200 Xilinx, 16384 bloques lógicos.
- ▶ XC4000 Xilinx, 3136 bloques lógicos.



FPGA's

**Clasificación de los
FPGA's según la
tecnología de los
elementos de
configuración**

Volátiles

No volátiles

Reprogramables

No reprogramables

FPGA's

**Clasificación de los
FPGA's según la
topología de los
recursos de
interconexión**

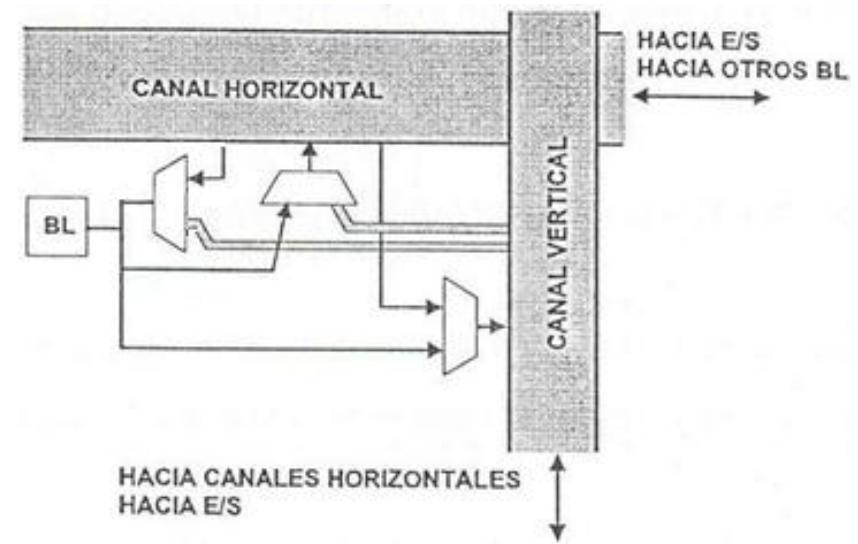
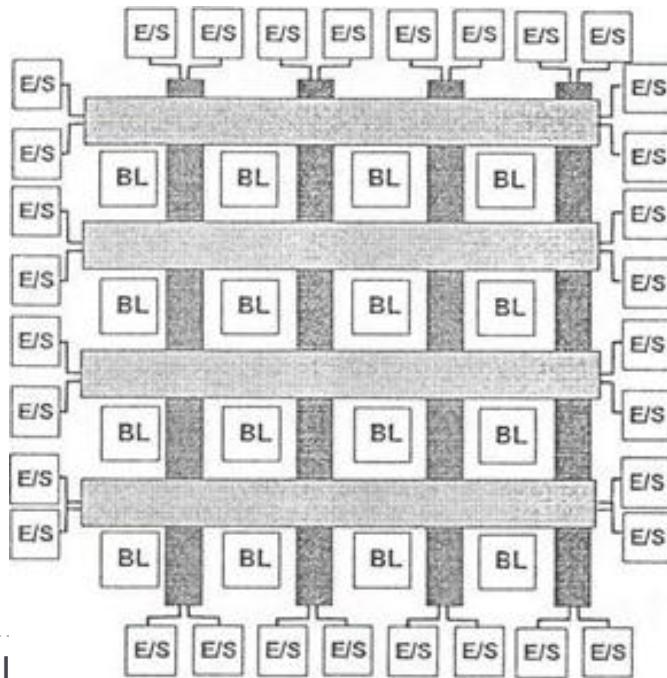
No segmentados

Segmentados

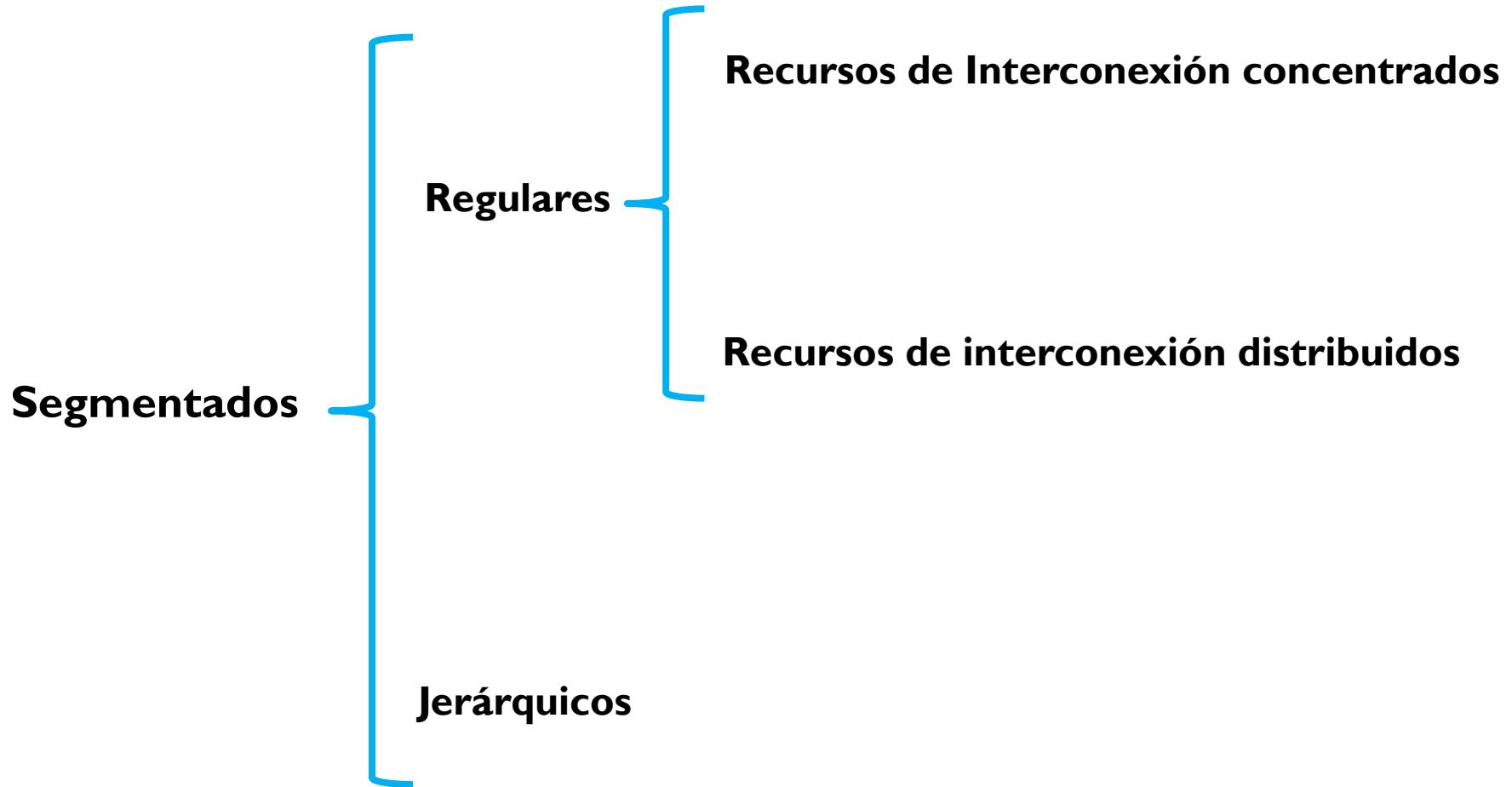


FPGA's

- ▶ Topología de interconexión no segmentada.
 - ▶ Líneas de comunicación continuas de longitud fija que se extienden a lo largo del dispositivo.
 - ▶ Ventaja: retardos predecibles y dispositivos mas veloces.
 - ▶ Desventaja: poco eficiente para conexiones entre bloques funcionales contiguos.

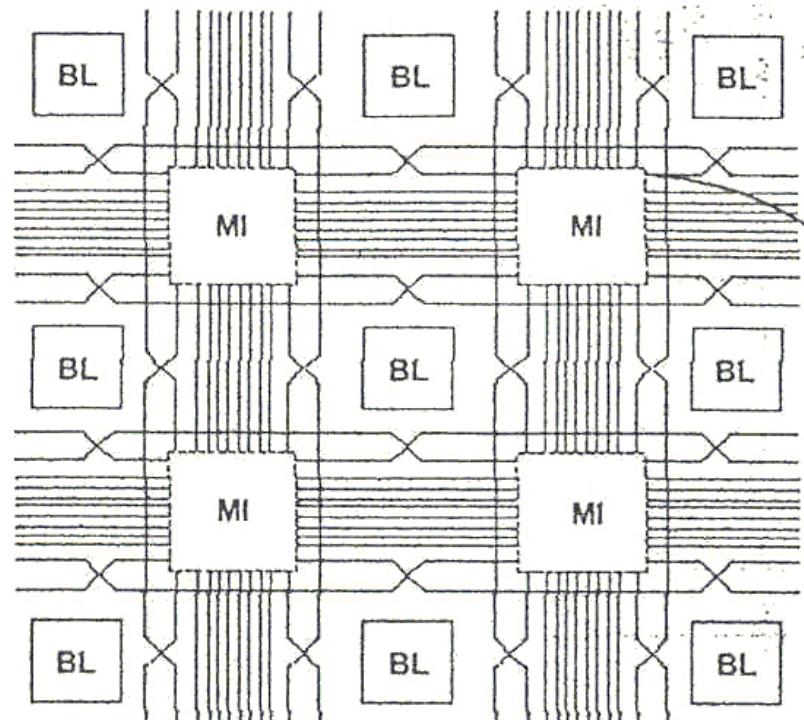


FPGA's según la topología de los recursos de interconexión



FPGA's

- ▶ Topología de interconexión segmentadas regulares.
 - ▶ En esta topología los recursos de interconexión no generan vínculo alguno entre bloques lógicos.
- ▶ Regulares con recursos de interconexión concentrados
 - ▶ Estos recursos se concentran en grupos y ocupan espacios específicos.
 - ▶ FPGA's basadas en células de memoria estática.

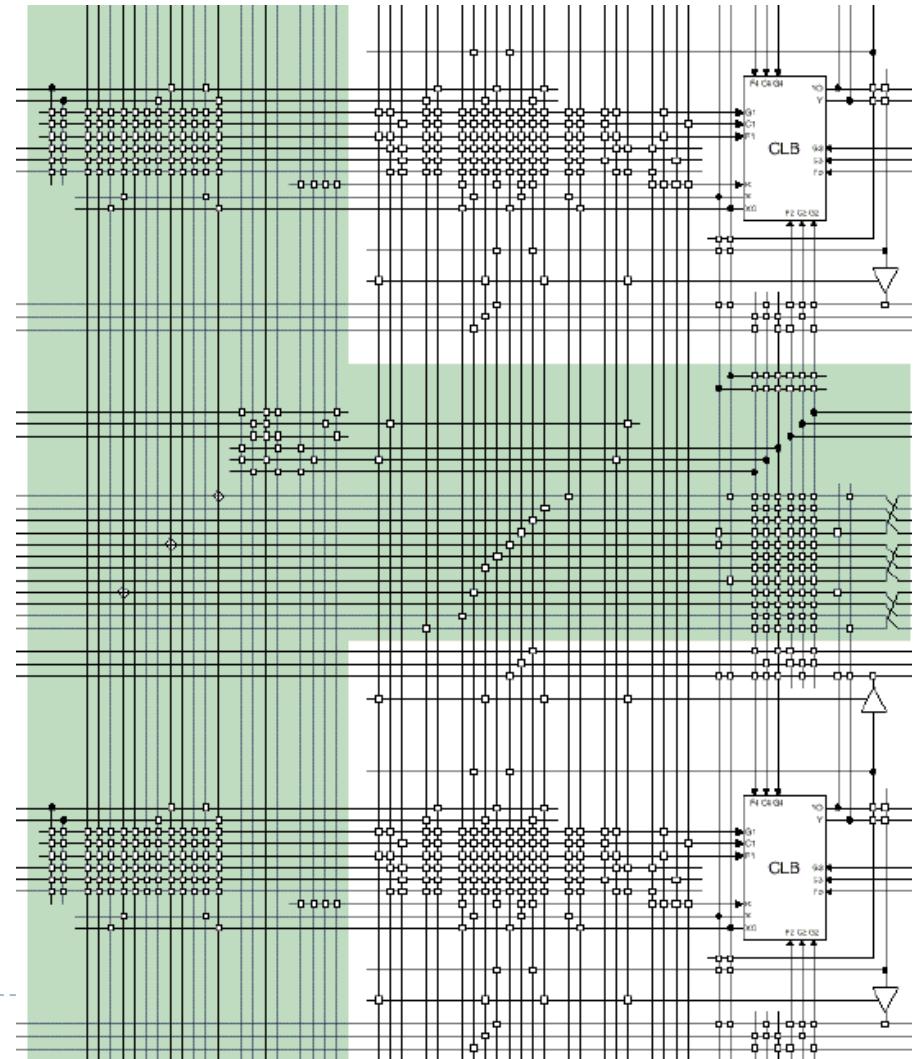


MI - Matriz de interconexión.
BL - Bloque Lógico.

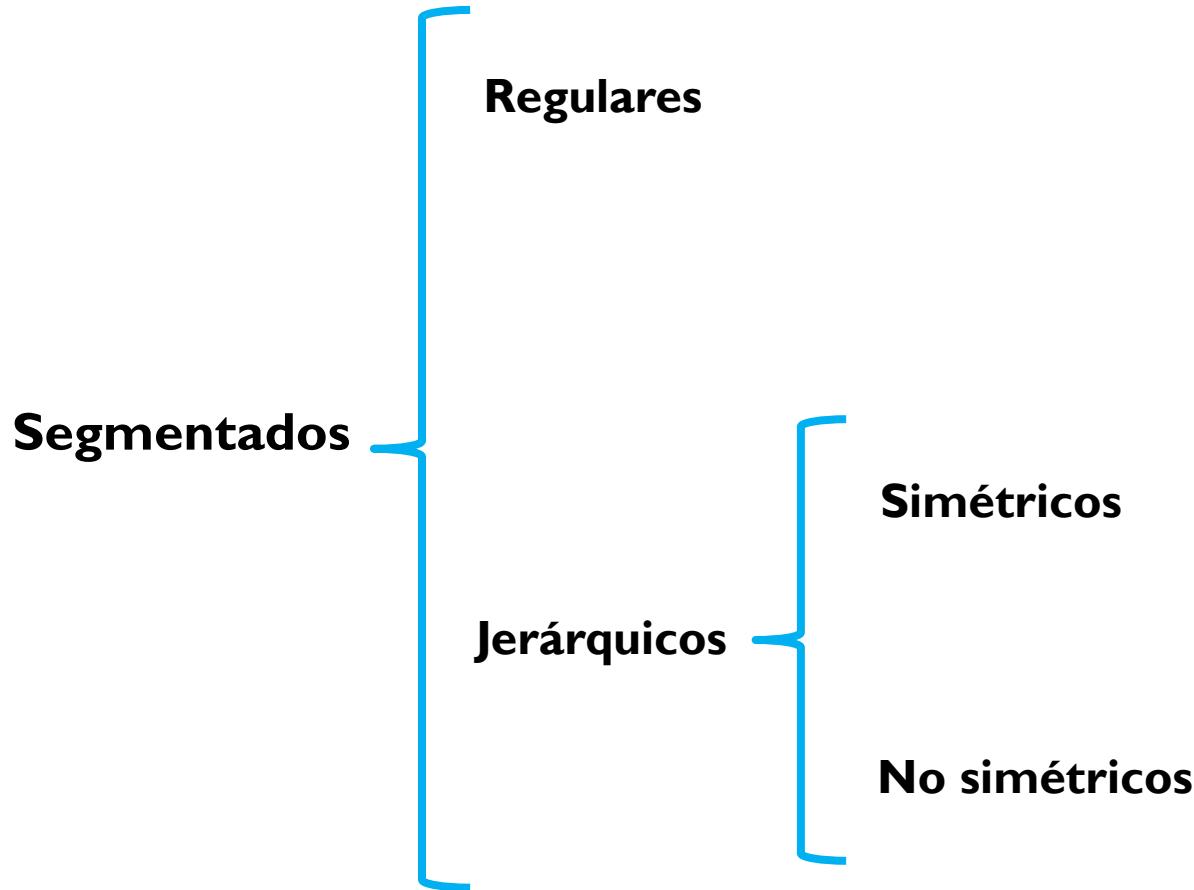
FPGA's

► Regulares con recursos de interconexión distribuidos

- Se caracteriza por tener abundantes recursos de interconexión que están distribuidos por toda el área del dispositivo.
- FPGA's basados en antifusibles.



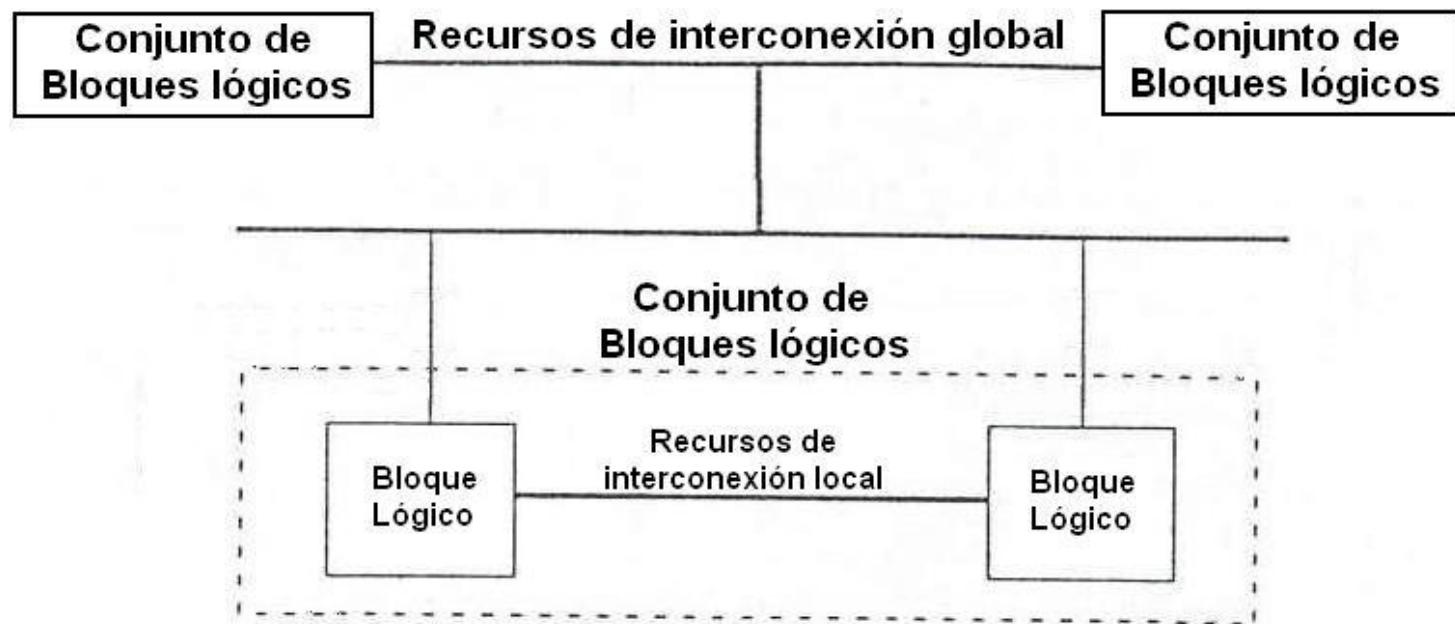
FPGA's según la topología de los recursos de interconexión



FPGA's

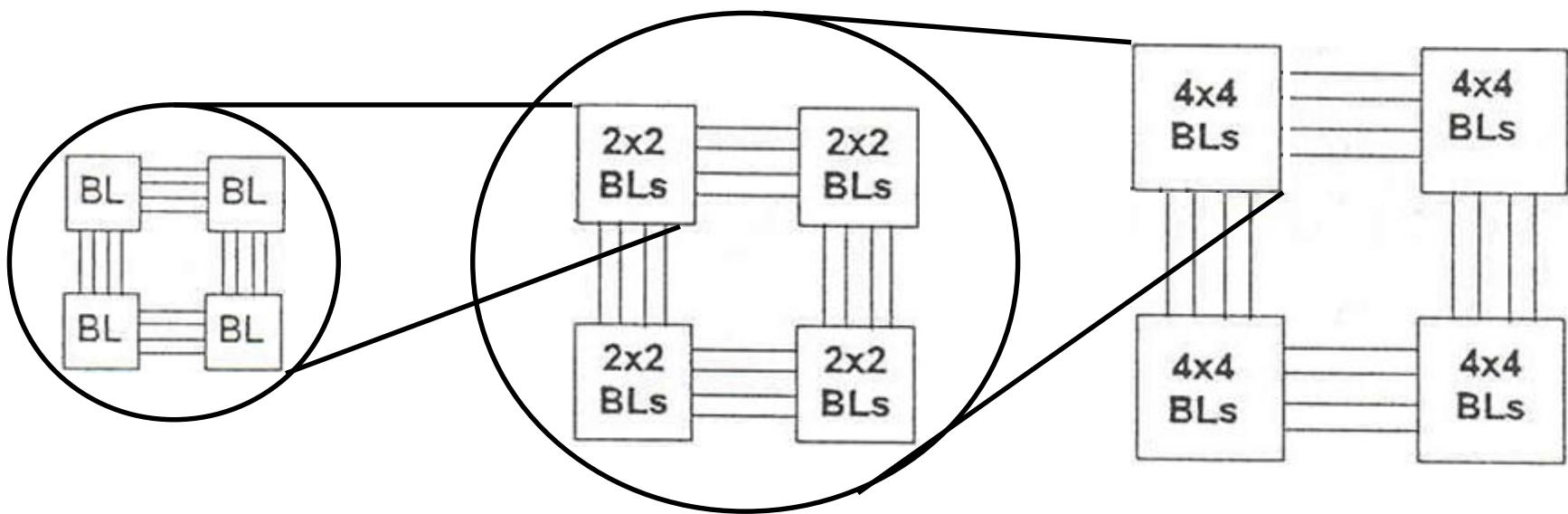
► Topologías segmentadas jerárquicas.

- En este tipo de FPGA's los bloques lógicos se agrupan en conjuntos que poseen recursos de interconexión propios, dichos conjuntos se interconectan entre si mediante otros recursos de interconexión, estableciendo así una estructura jerárquica de niveles.



FPGA's

- ▶ Jerárquicas no simétricas.



FPGA's

► Jerárquicas simétricas.

- BL Bloque lógico.
- Bloque de entrada/salida.
- ↔ Recurso de interconexión local.
- Recurso de interconexión media.
- Recurso de conexión de cuadrantes.
- Recurso de interconexión global.

