

AdC 2022

- Patterson, Hennesy, “**Computer Organization and Design - ARM edition**”, 2017.
- Stuart Sutherland, Simon Davidmann y Peter Flake, “**SystemVerilog For Design - A Guide to Using SystemVerilog for Hardware Design and Modeling**”, Second Edition, 2006.
- Hennesy, Patterson, “**Computer architecture - A quantitative approach**”, Sixth Edition, 2019.
- Harris, Harris, “**Digital Design and Computer Architecture - ARM edition**”, 2016.

# Parciales y Laboratorios

## Parciales:

- Parcial 1: Viernes 23 de septiembre
- Parcial 2: Viernes 28 de octubre
- Recuperatorios: Miércoles 23 de noviembre

## Laboratorios:

- Presentación Oral Laboratorio 1: Miércoles 16 de Noviembre
- Presentación Oral Laboratorio 2: Viernes 18 de Noviembre

# FIELD PROGRAMMABLE GATE ARRAYS (FPGAs)

Arquitectura de Computadoras 2022

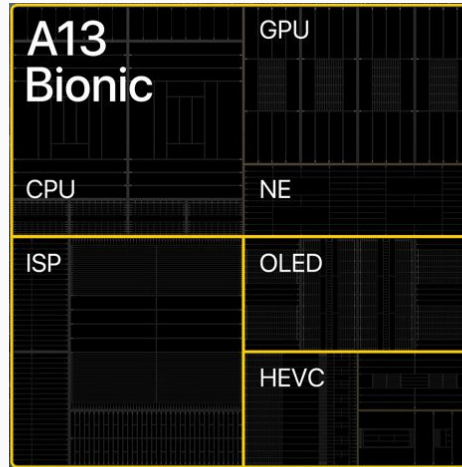
# FPGAs: ¿Qué son?

“Arreglos de Compuertas Programables en Campo”

Son circuitos integrados digitales que contienen bloques lógicos programables junto con interconexiones configurables entre dichos bloques.

# Microprocesadores, ASIC, FPGA

- Microprocesadores: el hardware es fijo. Las funciones se realizan en software.
- ASIC (Application Specific IC): diseñado para implementar una función lógica particular. Son “hechos a medida”. Son más rápidos que las FPGA, consumen menos y, fabricados en gran escala, son más baratos.
- FPGA: las funciones se realizan en hardware. No son hechas a medida, por lo que el usuario puede configurarlas de acuerdo a sus necesidades.



ASIC: A13 Bionic chip

# FPGAs: ¿Cómo se configuran?

- Algunas FPGA permiten ser programados una sola vez (OTP One Time Programmable).
- Otras pueden ser programadas múltiples veces “in the field”, es decir, no las programa el fabricante, sino el desarrollador.
- Si un dispositivo puede ser programado mientras embebido en un sistema mayor, se dice que es ISP (In System Programmable).

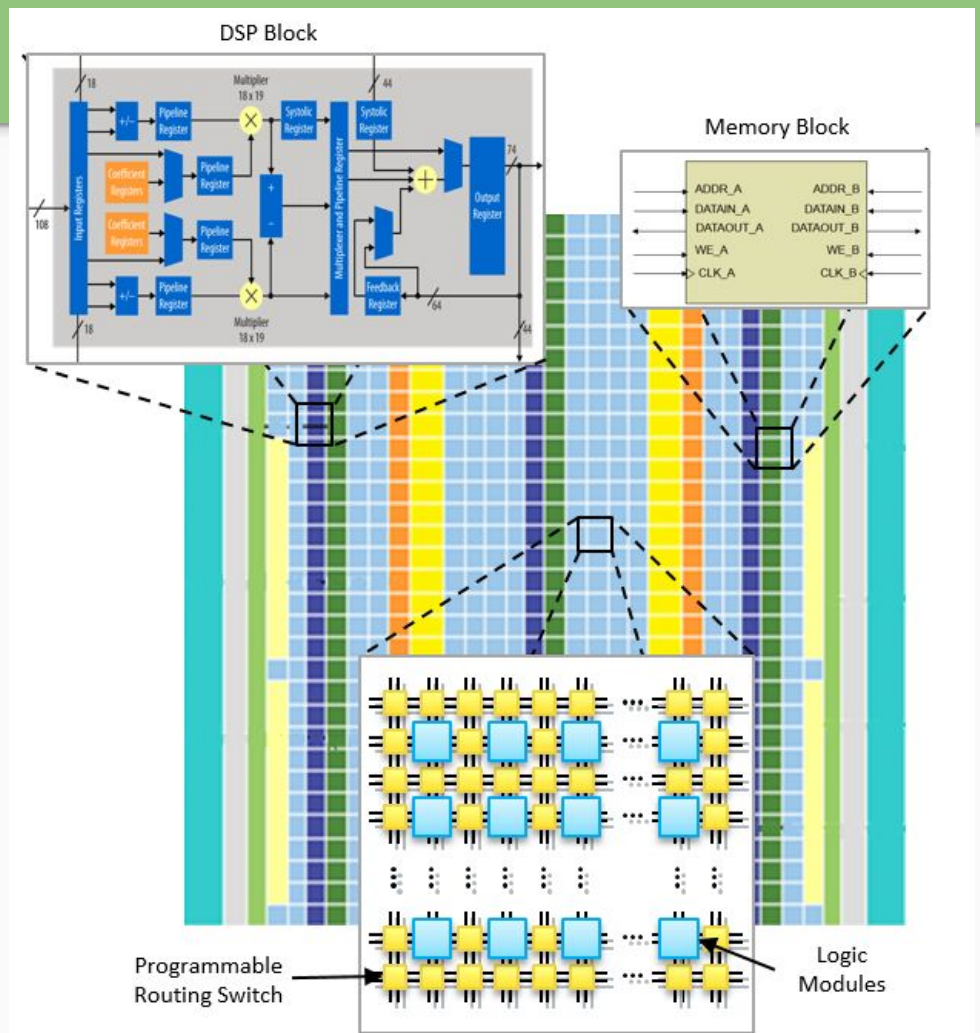
# FPGAs: Elementos básicos

- Elementos lógicos
- Recursos de memoria
- I/O configurables
- Recursos de ruteo
- Recursos adicionales

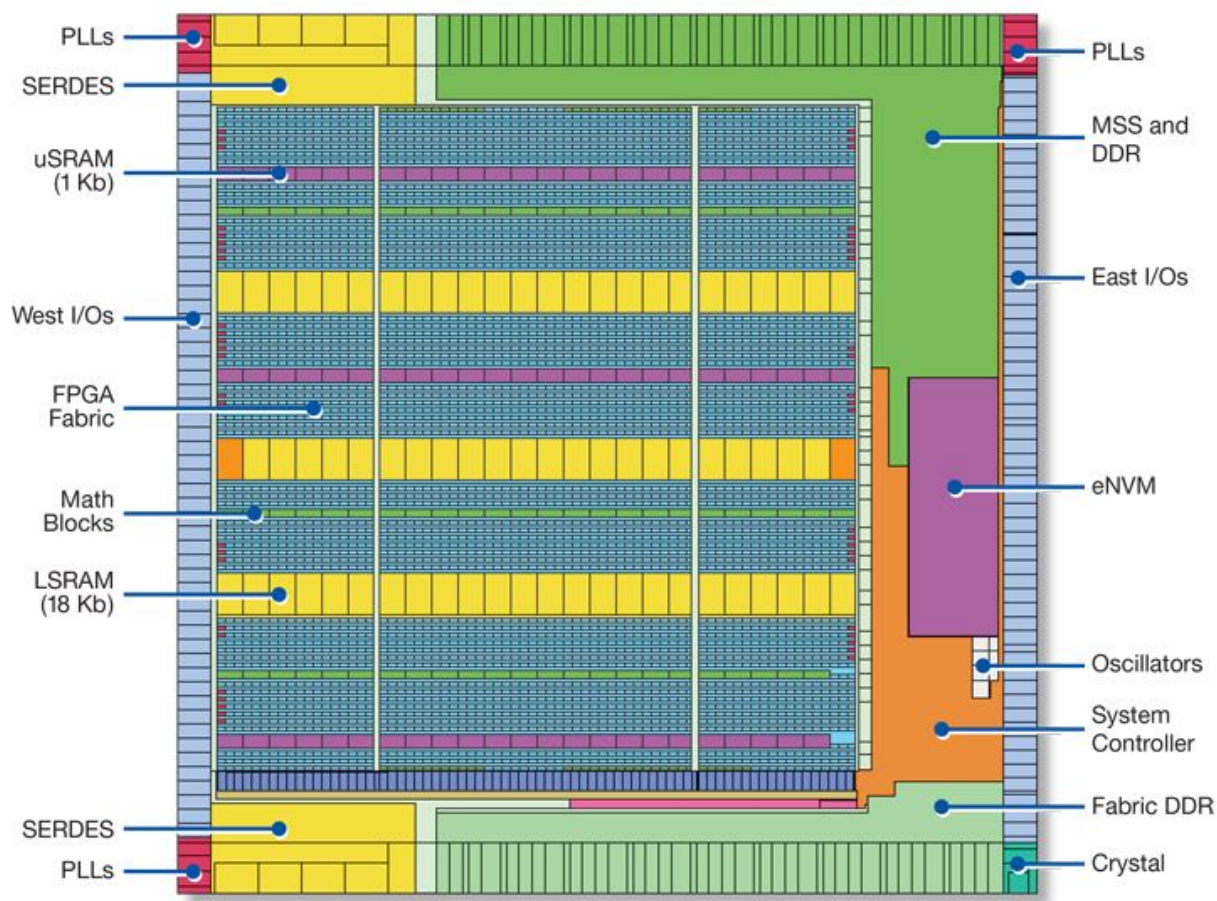


# FPGAs:

## Arquitectura genérica

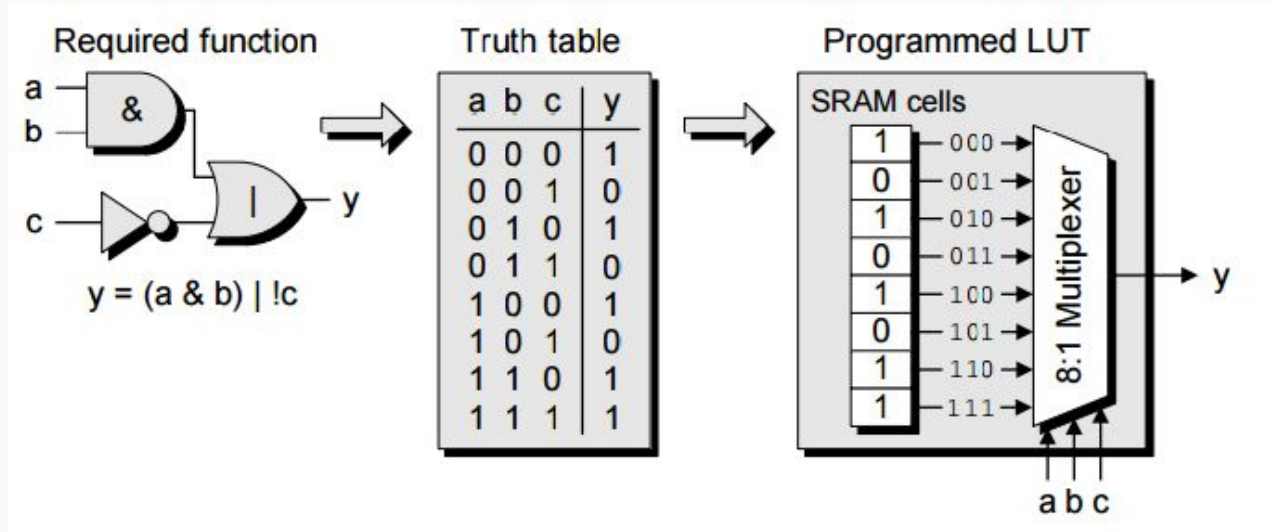


# FPGAs: Arquitectura genérica



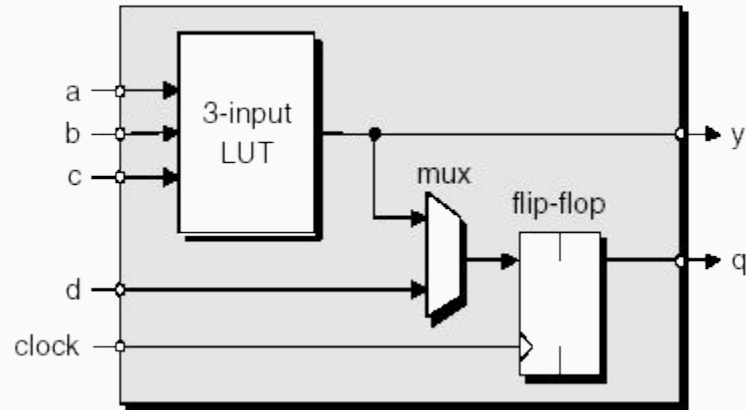
# FPGAs: Elementos lógicos (LUT)

La función lógica se almacena en una tabla de verdad de 16x1 (para las LUTs de 4 entradas). La columna de valores de salida de la función combinacional son los valores que realmente se almacenan en la LUT.



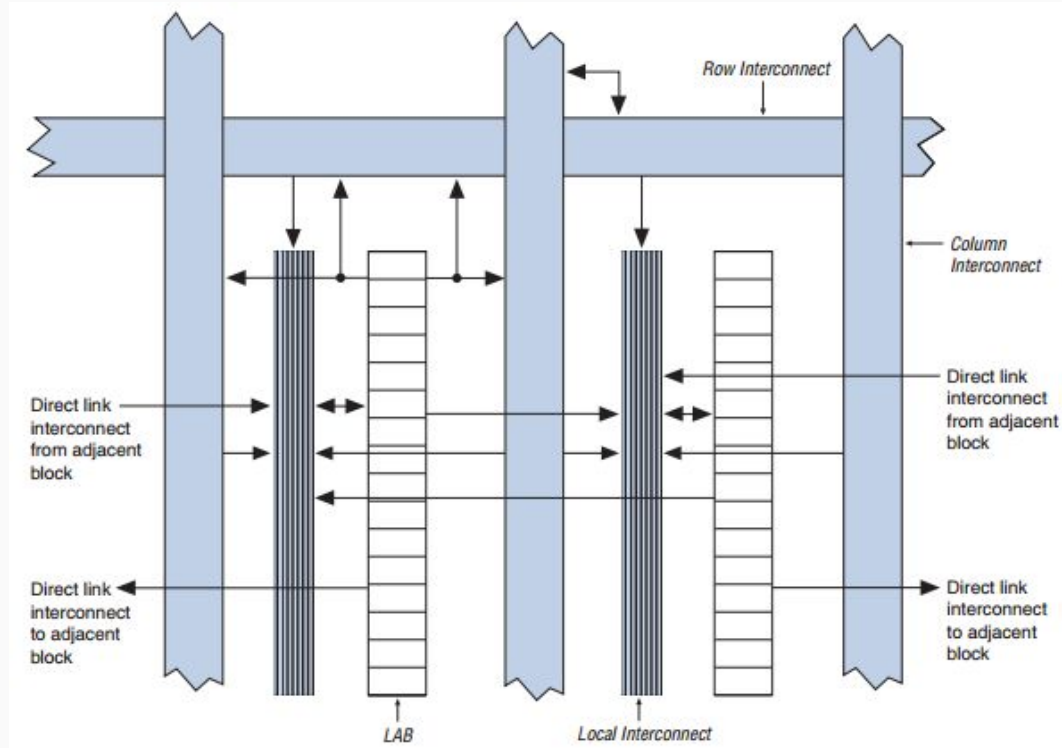
# FPGAs: Elementos lógicos

Todas las FPGA se basan en arrays de pequeños elementos de lógica digital. Para usar un determinado dispositivo, los problemas de lógica digital deben ser descompuestos en circuitos lógicos que puedan ser mapeados en una o más de estas “celdas lógicas”.



# Logic Array Blocks

Contienen grupos de LEs:



# FPGAs: Recursos de memoria

Para aplicaciones que requieren acceso a memoria, las FPGAs cuentan con bloques de memoria disponibles. La cantidad de bloques disponibles depende del tamaño de la FPGA.

Se pueden conectar diferentes RAMs en cascada, ya sea para tener un mayor ancho de la palabra de datos, para tener un mayor tamaño de la memoria o ambos.

Un mismo bloque de memoria puede ser configurado para que funcione como RAM, ROM, FIFO (First Input First Output), convertidor de ancho de palabra, registro de desplazamientos, etc.

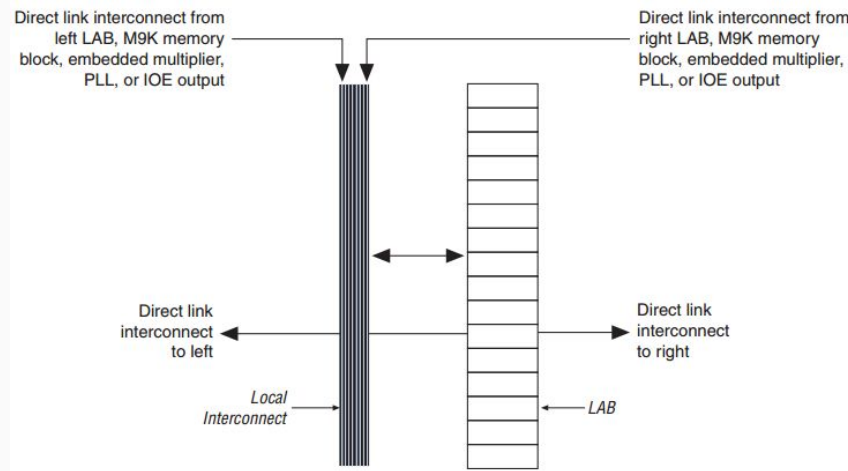
# FPGAs: I/O Configurables

Para poder recibir y transmitir señales digitales, las FPGAs disponen de un complejo bloque de E/S que posibilita su uso en muy diversos rangos de tensiones, frecuencias de trabajo, estándares de señales digitales, etc., lo que las hace muy adaptables a las necesidades del sistema del que forman parte.

Existe un bloque E/S por cada terminal de la FPGA, por lo que cada una puede ser configurada como entrada, como salida o bidireccional.

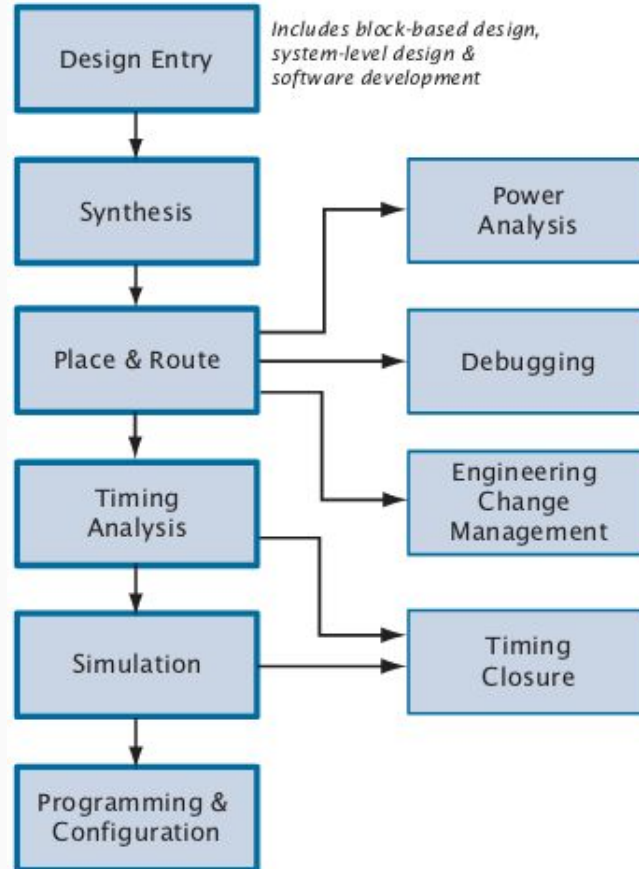
# FPGAs: Recursos de ruteo

Las conexiones entre LEs dentro de un LAB se realizan mediante columnas y filas internas (Local Interconnect). Las conexiones con otros LABs, PLLs, memorias y multiplicadores adyacentes también pueden realizarse a través de la línea Local Interconnect (Direct link interconnect), minimizando el uso de filas y columnas externas y proporcionando un mayor rendimiento.





# Flujo de diseño



# FPGAs: Compilación del circuito

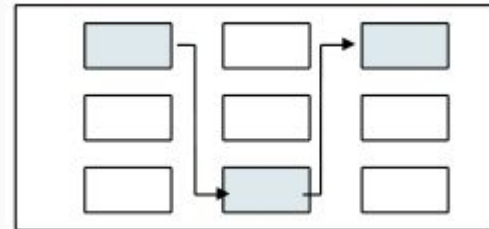
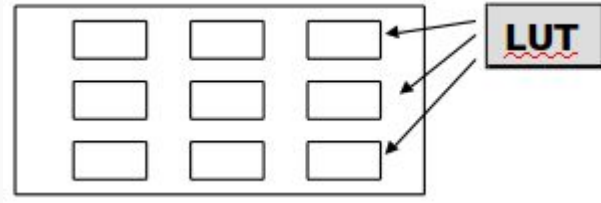
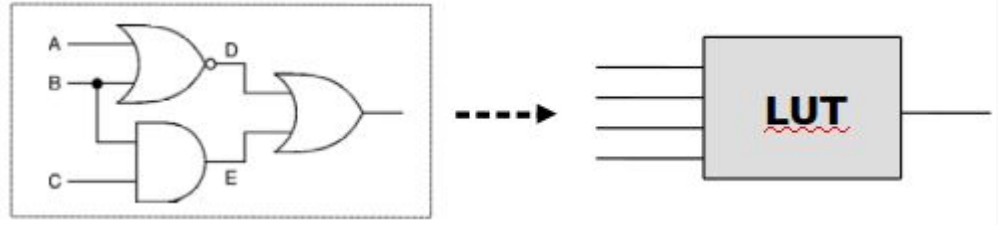
Technology Mapping



Placement



Routing



# Flujo de diseño

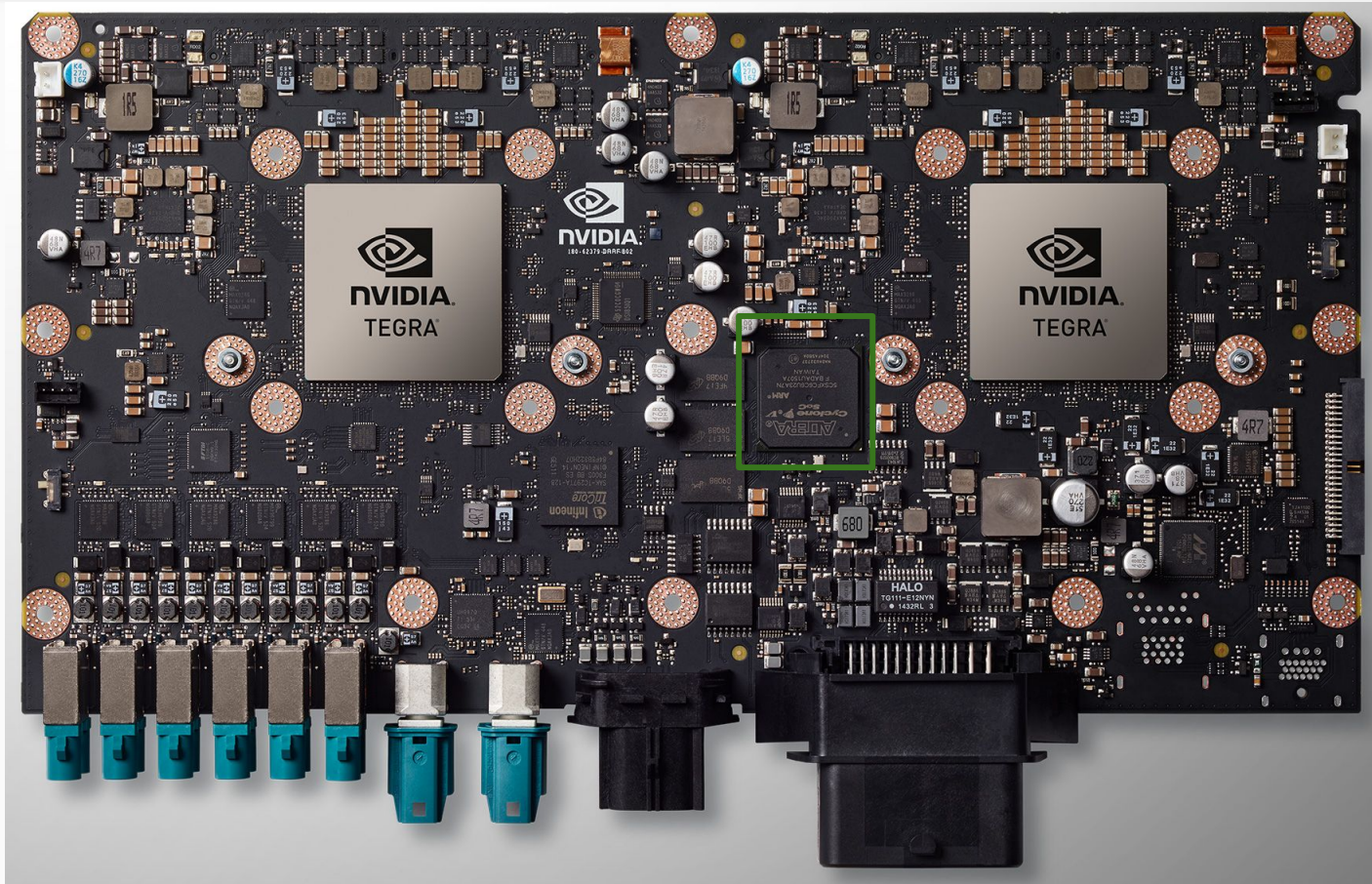
- **Design Entry:** El circuito deseado es especificado mediante un diagrama esquemático o utilizando algún lenguaje de descripción de Hardware como SystemVerilog.
- **Synthesis:** A partir del diseño ingresado, se infiere la lógica correspondiente y se sintetiza a un circuito utilizando los elementos lógicos (LEs) del chip FPGA.
- **Functional Simulation:** Se verifica la funcionalidad del diseño sintetizado mediante simulación.
- **Fitting:** La herramienta *Fitter* determina la ubicación de los LEs del diseño en los LEs disponibles en el chip FPGA y elige las interconexiones entre ellos.
- **Configuration:** El circuito diseñado es implementado físicamente en el chip FPGA.

# FPGAs: ¿Para qué se usan?

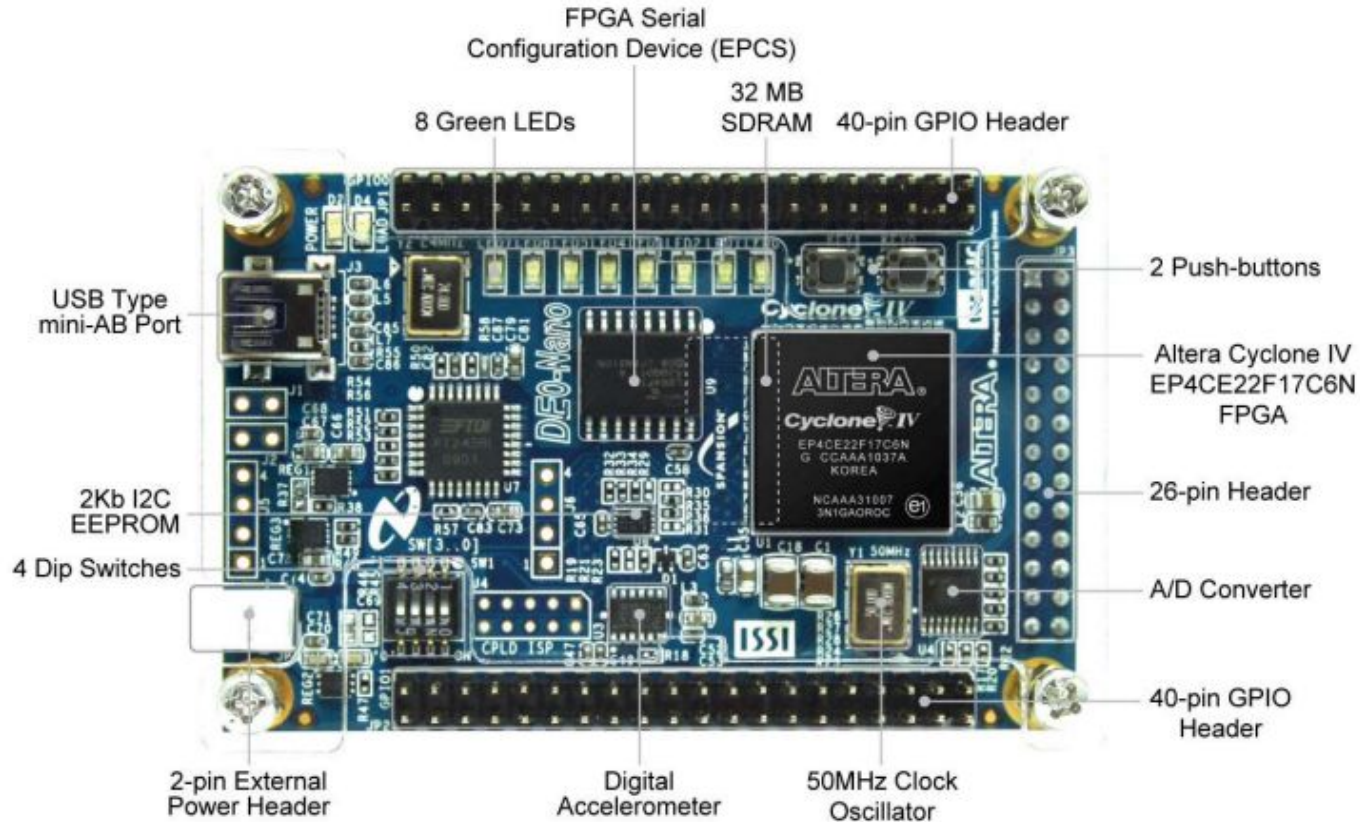
Cualquier circuito de aplicación específica puede ser implementado en una FPGA, siempre y cuando ésta disponga de los recursos necesarios.

- DSP (procesamiento digital de señales)
- Sistemas aeroespaciales y de defensa
- Prototipos de ASICs
- Sistemas para medicina
- Bioinformática
- Computación reconfigurable
- Inteligencia artificial
- Emulación de hardware de computadora, entre otras.

## Ejemplo: NVIDIA Drive PX2 (Autopilot)



# DE0-Nano - Altera Cyclone IV FPGA starter board



# Bibliografía de consulta

- G. Güichal. “Dispositivos Lógicos Programables (FPGAs)”. 2010.
- C. Sisterna. “Field Programmable Gate Arrays (FPGAs)”.
- Altera Corporation. “Cyclone IV Device Handbook”. 2009.
- D. M. Harris, S. L. Harris. “Digital Design and Computer Architecture - Arm Edition”.