

# **Diseño de Circuitos Integrados utilizando Herramientas de Software Libre**

Leandro Marsó

FLISol Villa María, IFDC, 2015

# Temario

1

## Introducción

- Definiciones generales
- Planteamiento del problema y motivación
- Diseño físico

2

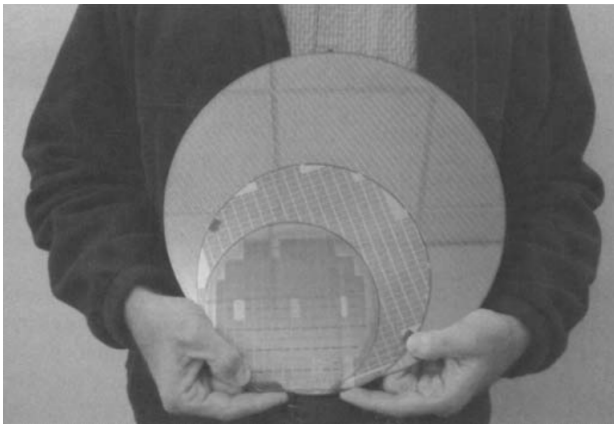
## Conclusiones

## 1 Introducción

- Definiciones generales
- Planteamiento del problema y motivación
- Diseño físico

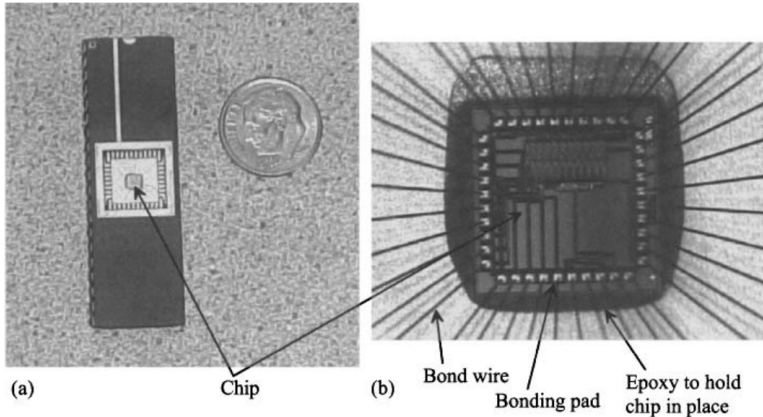
## 2 Conclusiones

\_\_\_\_\_



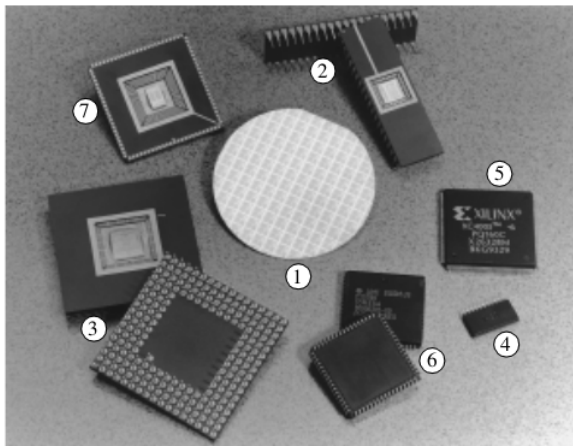
**Figura:** Obleas de silicio de 150, 200 y 300 mm de diámetro, de un proceso CMOS.

## ¿Qué es un circuito integrado?



**Figura:** Encapsulado del chip (a) y (b) una vista aumentada.

## ¿Qué es un circuito integrado?



- 1 Bare die
- 2 DIP
- 3 PGA
- 4 Small-outline IC
- 5 Quad flat pack
- 6 PLCC
- 7 Leadless carrier

**Figura:** Algunos tipos de encapsulados comunes.

## ¿Qué es un circuito integrado?

## ¿Cómo accedemos a fabricar circuitos integrados?

<b>Fabrica</b>	<b>Proceso CMOS</b>
TSMC	28 nm - 180 nm
Globalfoundries	14 nm - 180 nm
IBM	32 nm - 250nm
ON Semi	0.35 $\mu$ m - 0.7 $\mu$ m
Austria Micro Systems	180 nm - 0.35 $\mu$ m

**Tabla:** Procesos disponibles por medio de MOSIS

## ¿Qué es un circuito integrado?

## ¿Cómo accedemos a fabricar circuitos integrados?

Fabrica	Proceso CMOS
STMicroelectronics	28 nm - 130 nm
Austria Micro Systems	180 nm - 0.35 $\mu$ m

**Tabla:** Procesos disponibles por medio de CMP



## ¿Qué es un circuito integrado?

### ¿Cuánto podemos integrar?

#### CMOS 350 nm de AMS

- 18 kGates/mm<sup>2</sup>
- 650 €/mm<sup>2</sup>
- Área mínima 3 mm<sup>2</sup>
- 25 chips

#### CMOS 180 nm de AMS

- 118 kGates/mm<sup>2</sup>
- 1200 €/mm<sup>2</sup>
- Área mínima 5 mm<sup>2</sup>
- 25 chips

## ¿Qué es el Software Libre?

### Definición

«Software libre» es el software que respeta la libertad de los usuarios y la comunidad. A grandes rasgos, significa que los usuarios tienen la libertad de ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software. Es decir, el «software libre» es una cuestión de libertad, no de precio.

## Las cuatro libertades del Software Libre

Un programa es software libre si los usuarios tienen las cuatro libertades esenciales:

- La libertad de ejecutar el programa
- La libertad de estudiarlo
- La libertad de redistribuir copias
- La libertad de distribuir copias de sus versiones modificadas

## Las cuatro libertades del Software Libre

Un programa es software libre si los usuarios tienen las cuatro libertades esenciales:

- La libertad de ejecutar el programa
- La libertad de estudiarlo
- La libertad de redistribuir copias
- La libertad de distribuir copias de sus versiones modificadas

## Las cuatro libertades del Software Libre

Un programa es software libre si los usuarios tienen las cuatro libertades esenciales:

- La libertad de ejecutar el programa
- La libertad de estudiarlo
- La libertad de redistribuir copias
- La libertad de distribuir copias de sus versiones modificadas

## Las cuatro libertades del Software Libre

Un programa es software libre si los usuarios tienen las cuatro libertades esenciales:

- La libertad de ejecutar el programa
- La libertad de estudiarlo
- La libertad de redistribuir copias
- La libertad de distribuir copias de sus versiones modificadas

# Temario

1

## Introducción

- Definiciones generales
- Planteamiento del problema y motivación
- Diseño físico

2

## Conclusiones

¿Cómo diseñar circuitos integrados con herramientas flexibles y accesibles para todo tipo de uso: académico e industrial?

**Económico**

Factibilidad de proyectos según la escala

**Académico**

Oportunidad de abordaje multidisciplinario

**Otras razones**

Soberanía tecnológica



# Temario

1

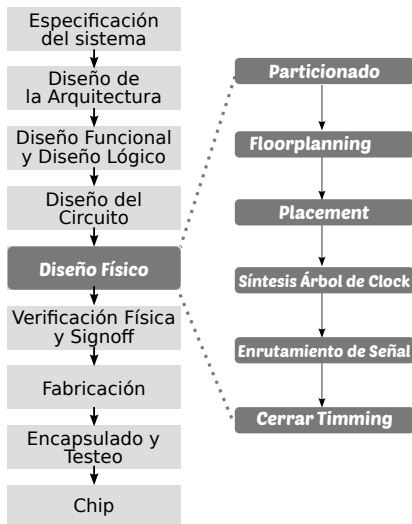
## Introducción

- Definiciones generales
- Planteamiento del problema y motivación
- Diseño físico

2

## Conclusiones

# Flujo de diseño físico



# Flujo de diseño físico - Herramientas



<http://opencircuitdesign.com/qflow/>

## Selección del proceso de fabricación

Podemos trabajar con tecnologías de hasta 130nm:

- Existen herramientas de software libre para estas tecnologías.
- Posibilidad de integrar sistemas de gran complejidad y alta performance, o simples y de bajo costo de fabricación.

## Selección del proceso de fabricación

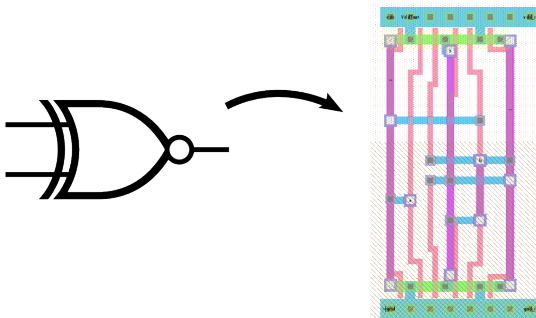
Ejemplos de microprocesadores que fueron fabricados en 180nm:

Procesador	Año de lanzamiento
Intel Coppermine E	1999
AMD Athlon Thunderbird	2000
Intel Celeron (Willamette)	2002
Motorola PowerPC 7445 y 7455 (Apollo 6)	2002

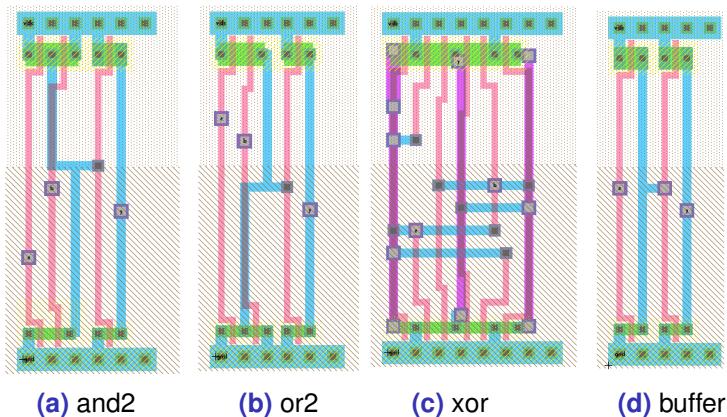
**Tabla:** Procesadores fabricados en CMOS 180nm

## Mapeo de lógica a compuertas

Mapeo de una función lógica a una celda estándar



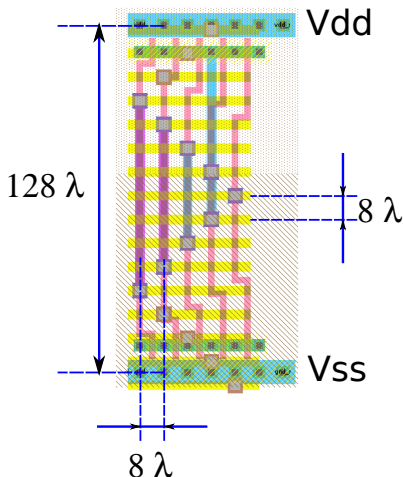
# Librería de celdas estándar



**Figura:** Conjunto de celdas estándar

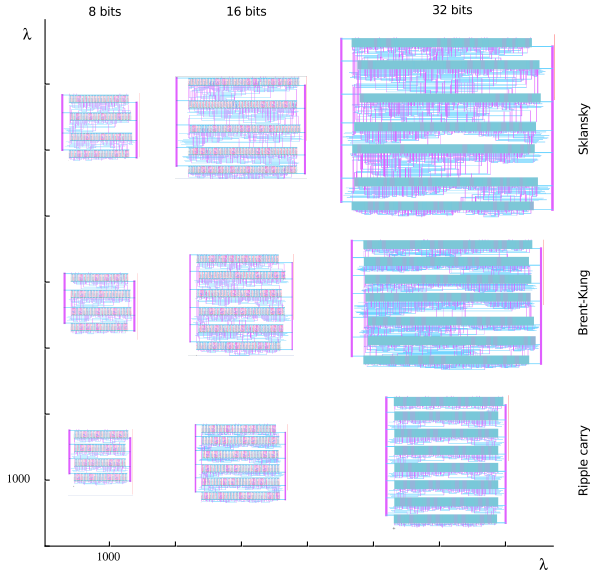
## Celdas estándar

Grilla de interconexión y riel de alimentación de las celdas estándar de  $128\lambda$

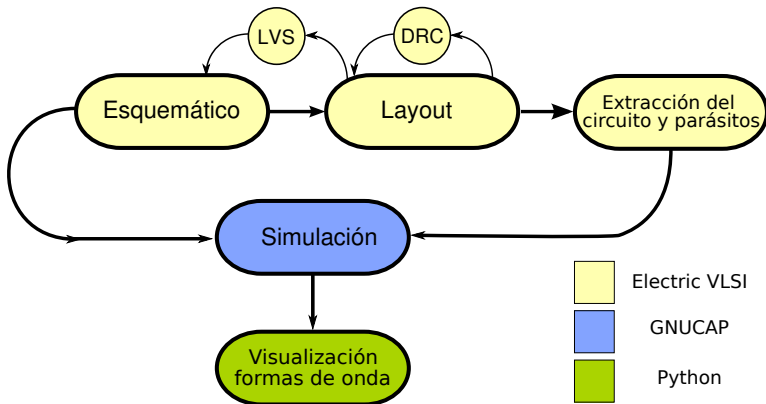




# Layout de todas las arquitecturas y tamaños



# Flujo para simulaciones analógicas



## Conclusiones

- Metodología flexible
  - Selección de la herramienta según la necesidad
  - Integración del flujo con modificaciones simples
  - Todos los procesos son automáticos
- Resultados del mismo orden de magnitud que otros estudios
  - Sumadores de cualquier tamaño
  - Sumadores rápidos, eficientes o de bajo consumo
- Metodología para circuitos combinacionales
  - Unidades aritméticas, decodificadores, codificadores, funciones lógicas
  - También diseños analógicos

## Conclusiones

Hemos encontrado una metodología completa de trabajo que nos permite ir desde la descripción funcional de sistemas digitales hasta el diseño del circuito integrado listo para ser enviado a fábrica.

## Desafíos futuros

- Implementar una ALU
- Implementar un circuito digital mayor, por ejemplo el openMSP430 (8kGates)
- Crear una batería de simulaciones para caracterizar celdas estándar en formato Liberty de forma automática
- Aportar al desarrollo del simulador gnuca, Yosys o Electric

**¡Gracias!**