

Arquitecturas FPGA

Diseño de Sistemas con FPGA
Patricia Borensztein

Micron: millonésima parte del metro

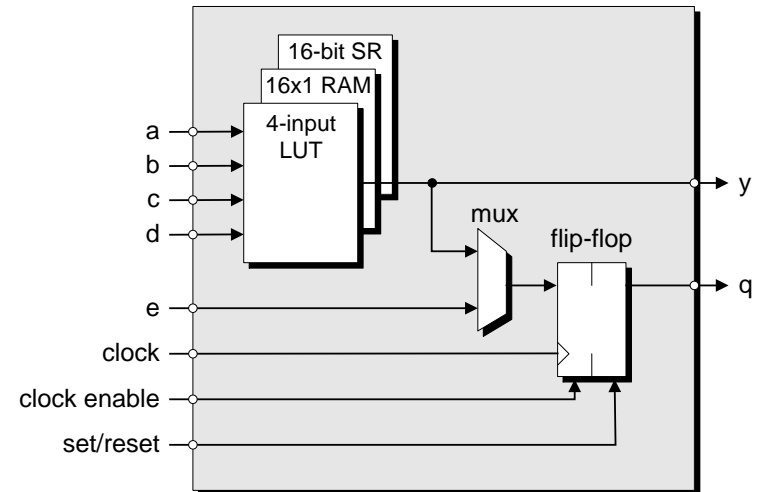
DSM: deep submicron

Nanometro: mil millonésima parte del metro

UDSM: ultra deep submicron

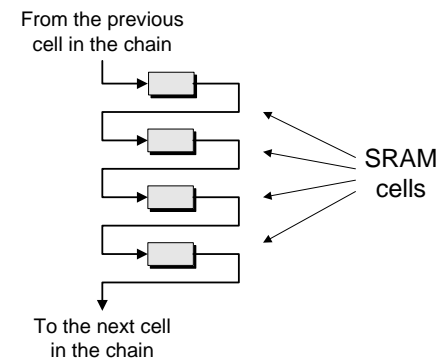
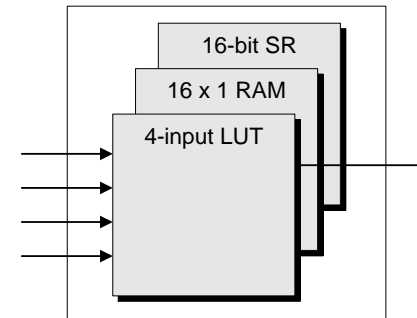
Bloques Básicos de Xilinx

- Se llama Logic Cell (LC) y está compuesto (básicamente) por:
 - Una LUT de 4 entradas
 - Un multiplexor
 - Un registro (flip-flop (actúa por flanco de reloj) o latch (actúa por nivel, monitorea siempre las entradas))



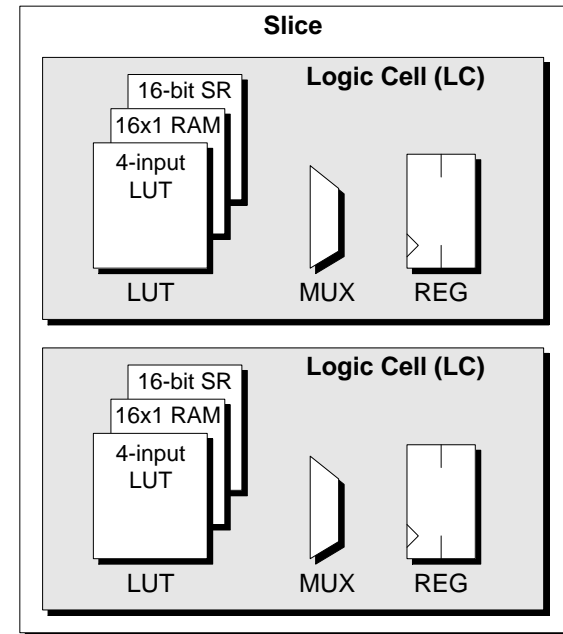
LUT's de SRAM

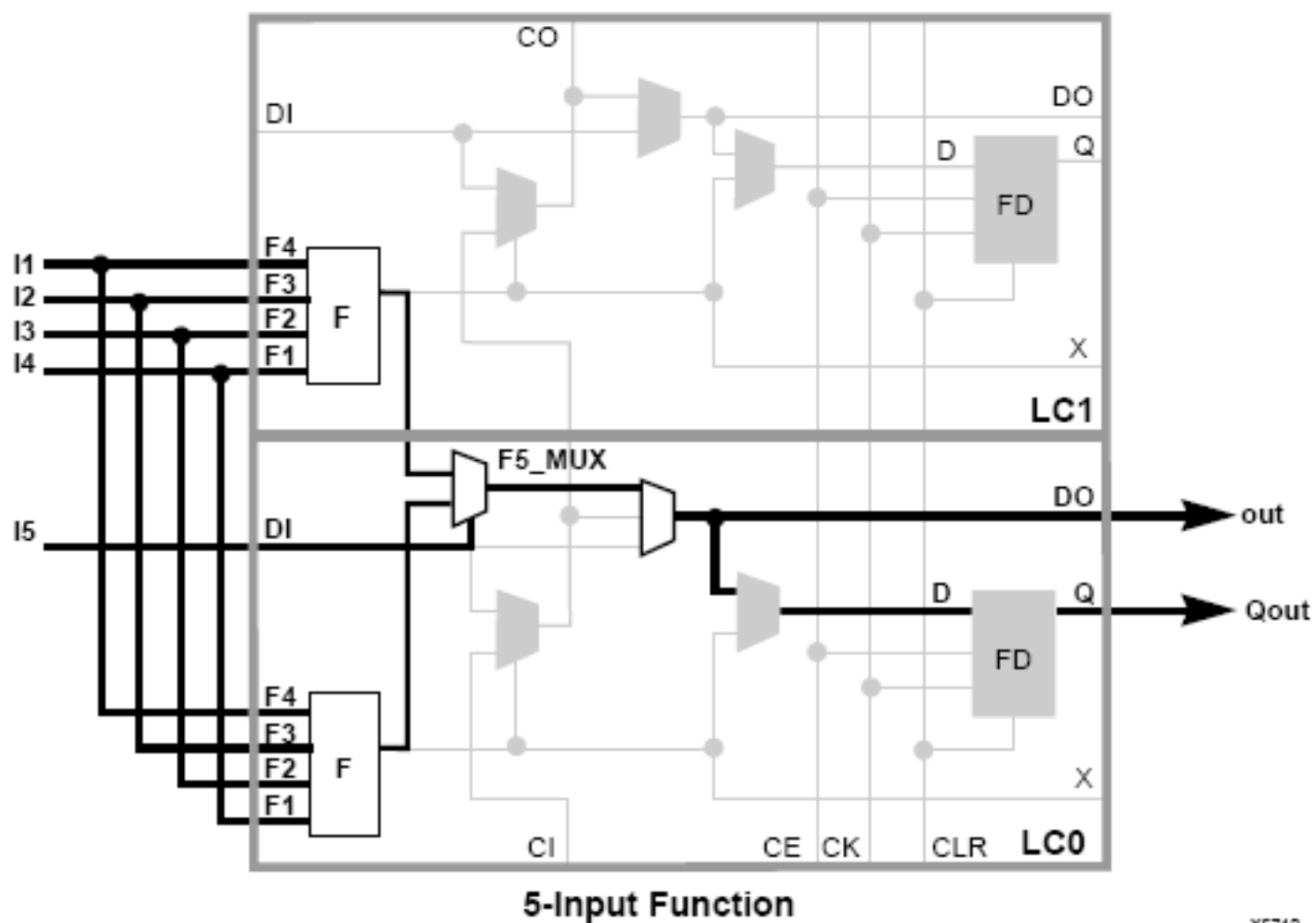
- Cuando una arquitectura basada en LUT se implementa con celdas de SRAM, las celdas, además de LUT's, pueden ser utilizadas como pequeños bloques de memoria RAM (una LUT de 16 entradas por un bit, puede ser una RAM de 16x1). Esto se denomina RAM Distribuida. (Distributed RAM)
- Como las celdas de SRAM están encadenadas unas con otras (las celdas de SRAM de configuración, pero también las de las LUT,s) formando una larga cadena, algunos fabricantes permiten que las que forman las LUT's sean tratadas de forma independiente, como un shift register.



Slice de Xilinx

- Dos LC forman un Slice.
- El slice comparte algunas señales:
 - Clock
 - Clock Enable
 - Set/Reset



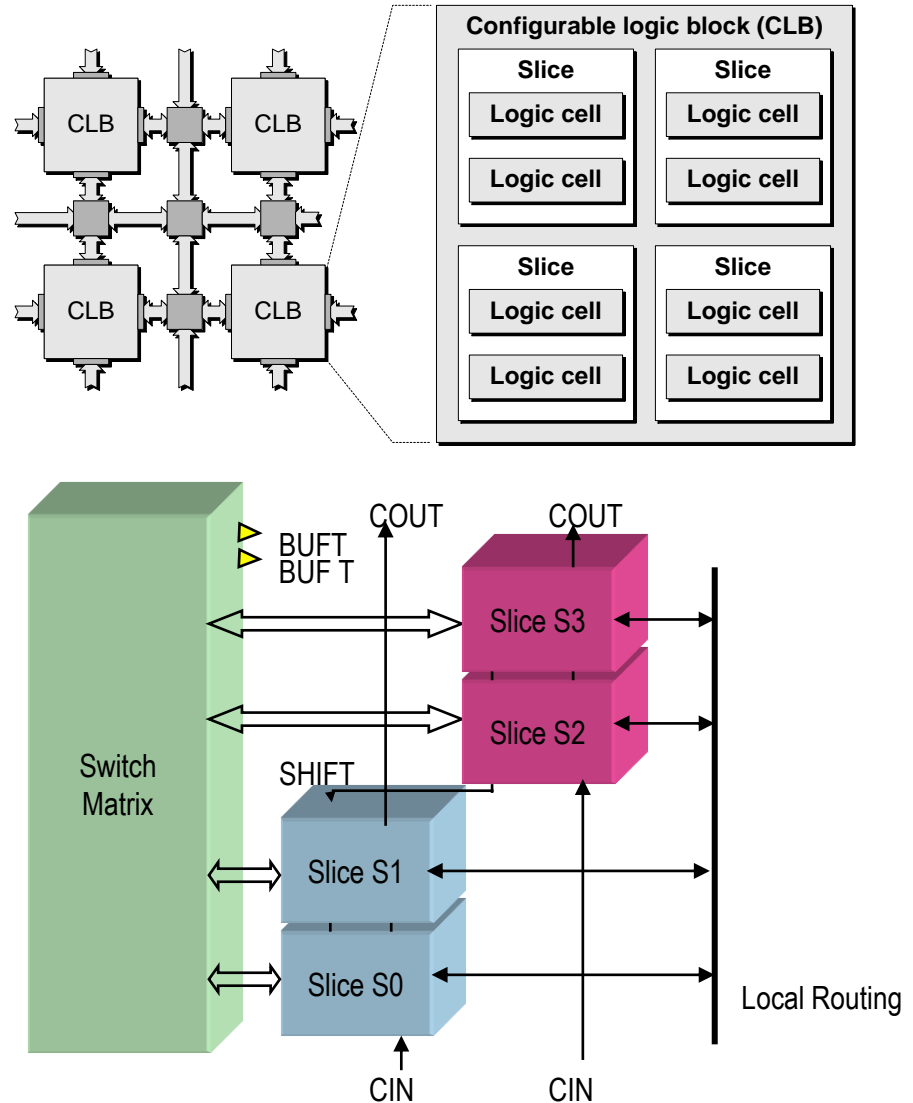


X5710

Figure 5: Two LUTs in Parallel Combined to Create a 5-input Function

CLB de Xilinx

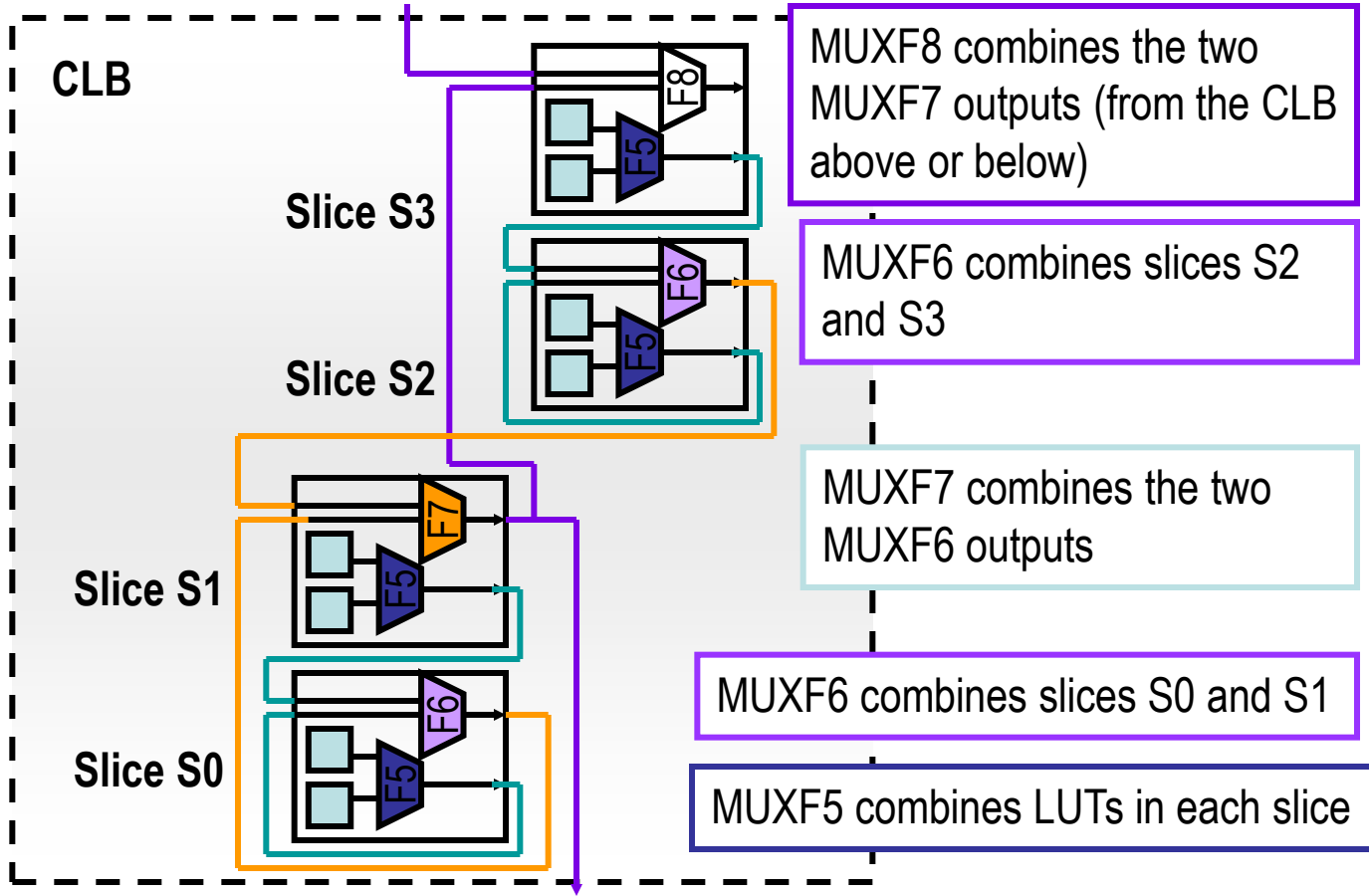
- Un nivel mas arriba están los CLB's: Configurable Logic Block. Pueden tener dos o cuatro slices, (o los que sean en el futuro)
- Un CLB corresponde a una "isla" de lógica programable dentro del mar de las conexiones programables.
- A la jerarquía de bloques, le corresponde una jerarquía en las interconexiones. Es decir, hay una muy rápida interconexión entre LC's, un poco menos rápida entre Slices y mas lenta entre CLB's.
- Local Routing comunica slices dentro del CLB y con otros CLB's cercanos.
- Una matriz de interconexión provee acceso a recursos generales



Lógica Dedicada

- Los FPGAs contienen lógica construida para acelerar operaciones y ahorrar recursos
 1. Lógica de Multiplexación: Conecta Slices y LUTs
 2. Cadenas de Acarreo (Carry Chains): Aceleran operaciones aritméticas
 3. Puerta MULT_AND : acelera la multiplicación
 4. Shift Register LUT: LUT-based shift register
 5. Multiplicadores embebidos: 18x18

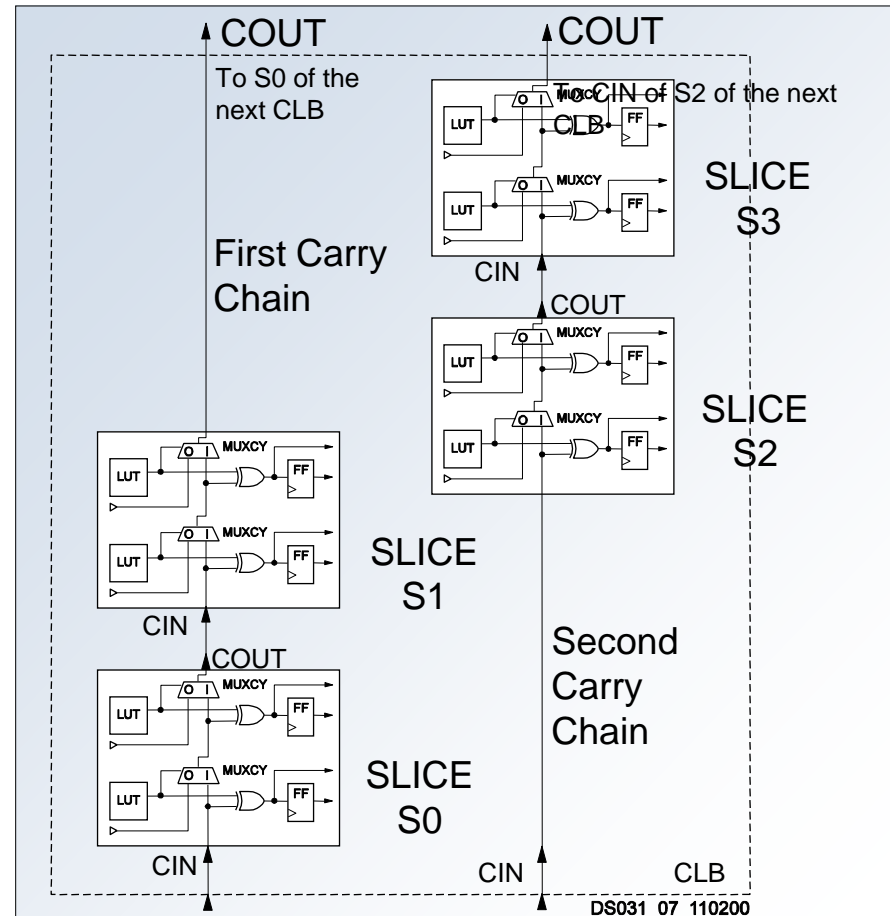
1. Multiplexores: conectan slices y LUTs



2. Carry Chains

Dedicated carry chains speeds up arithmetic operations

- Simple, fast, and complete arithmetic Logic
 - Dedicated XOR gate for single-level sum completion
 - Uses dedicated routing resources
 - All synthesis tools can infer carry logic



2. Cadena de Acarreo

- Recordamos las expresiones de la suma:

a	b	Cin	S	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$S_i = a_i \oplus b_i \oplus c_i$$

$$C_{i+1} = a_i b_i + (a_i \oplus b_i) c_i$$

(2) Adder de 4 bits en XC5200 (Xilinx)

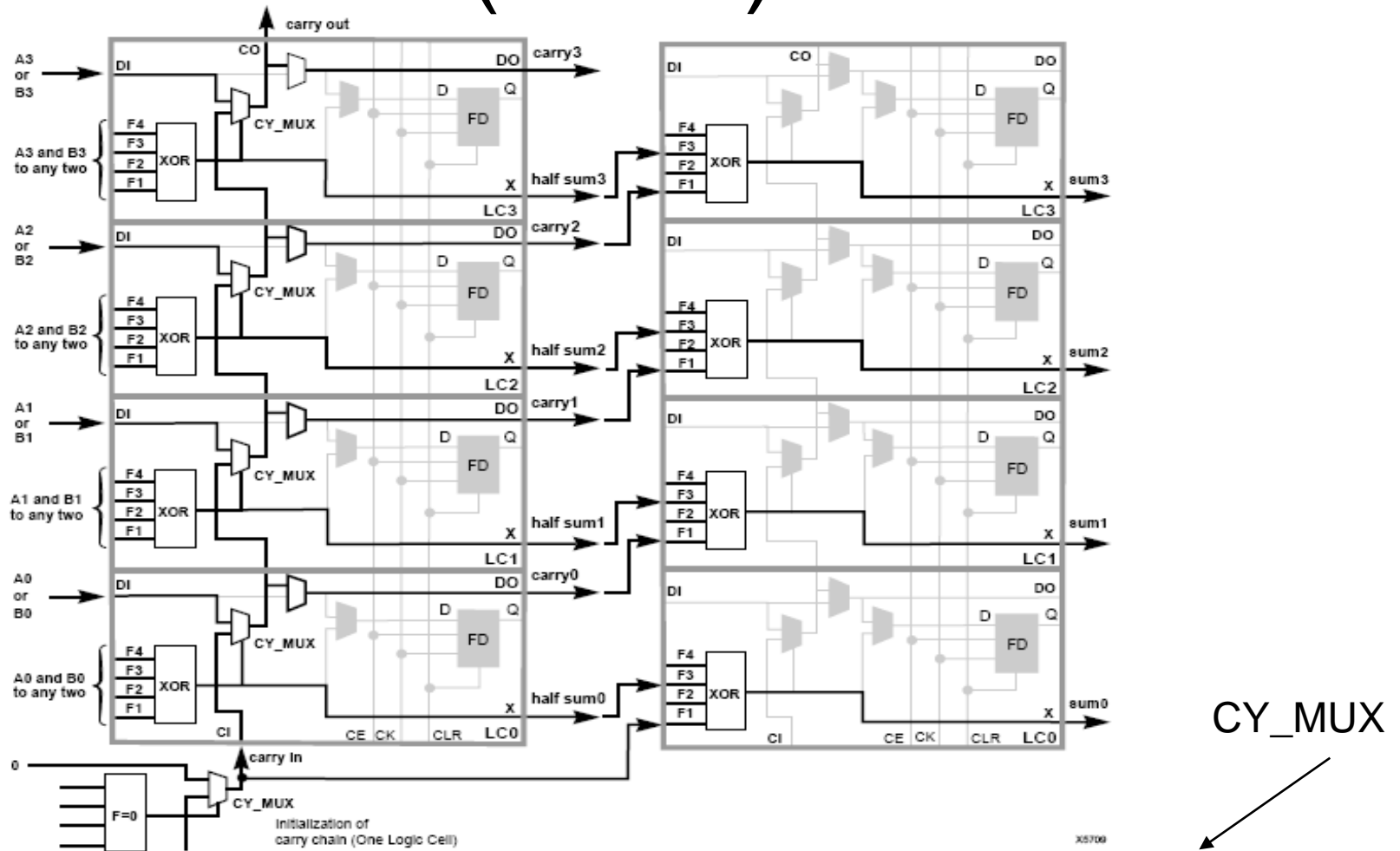
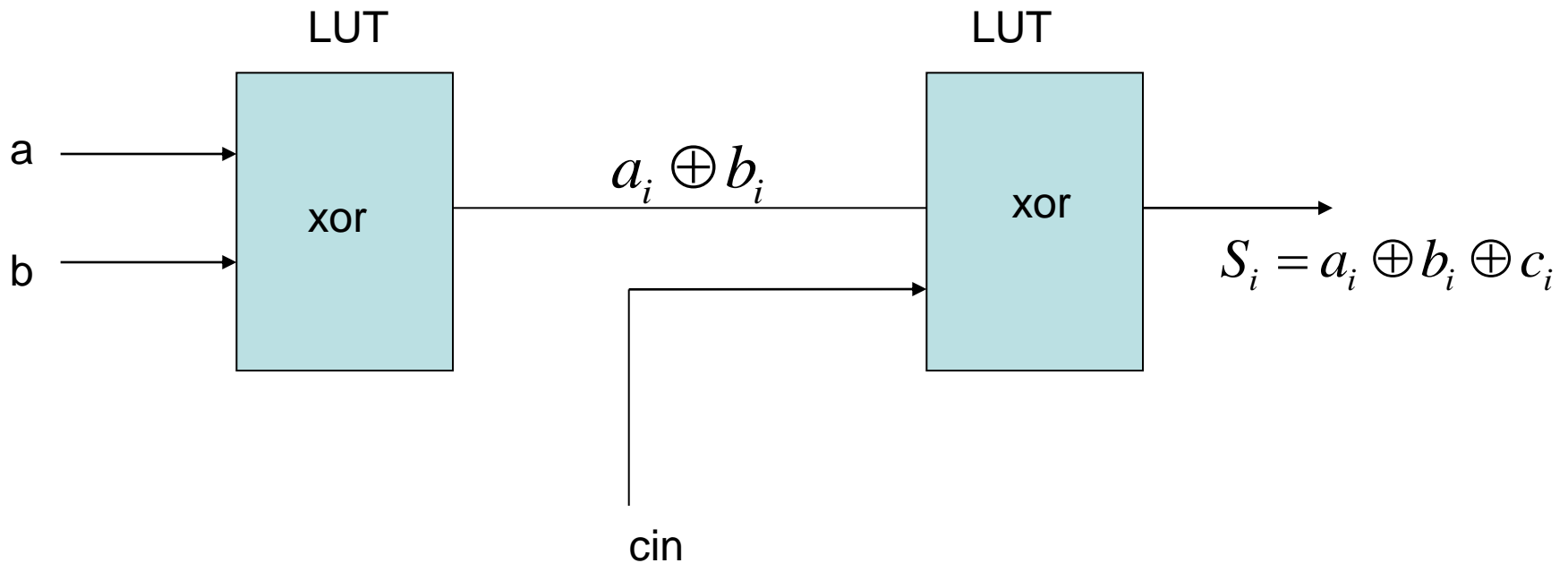


Figure 6: XC5200 CY_MUX Used for Adder Carry Propagate

$$C_{i+1} = a_i b_i + (a_i \oplus b_i) c_i$$

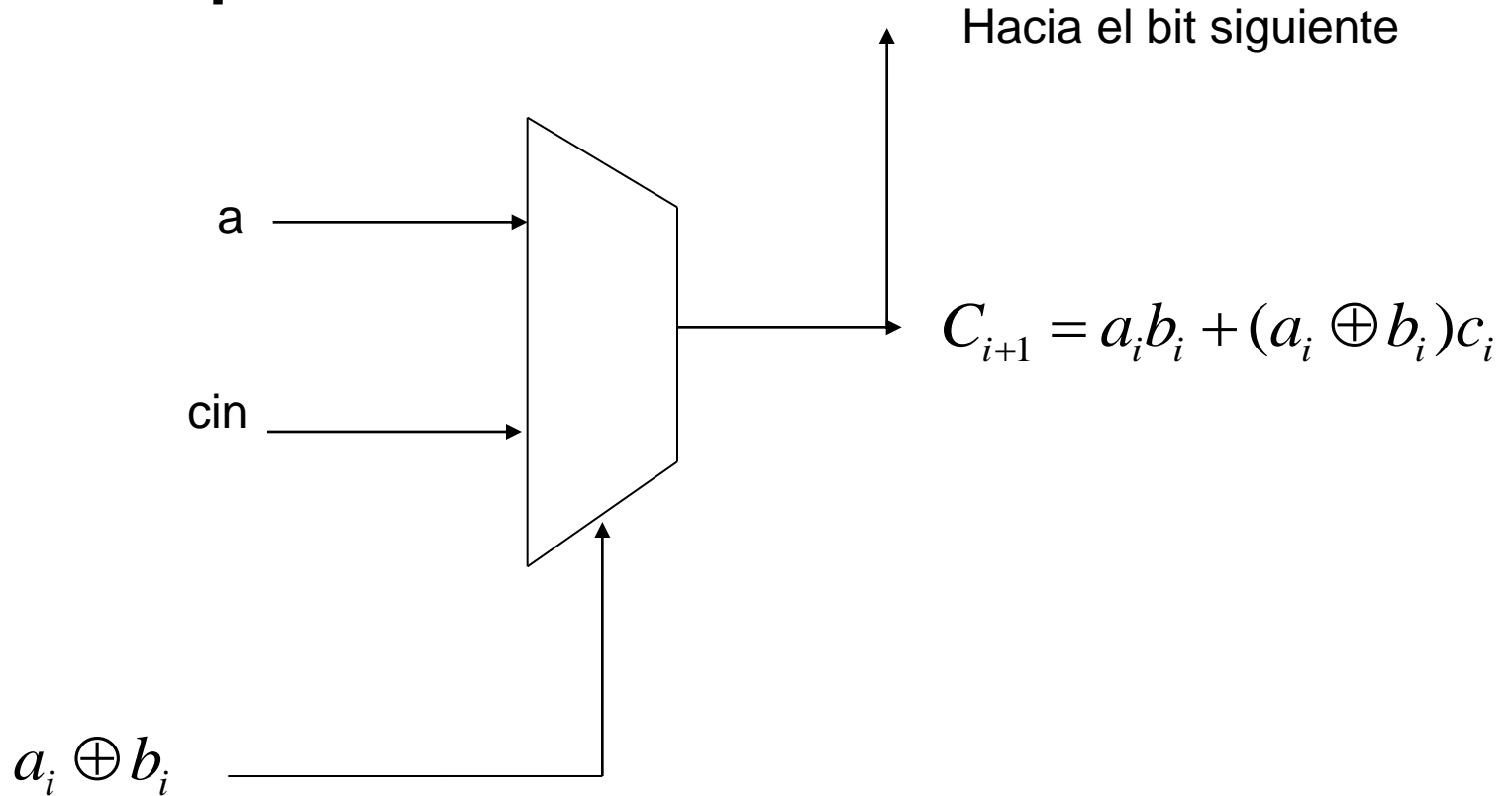
2. Cadenas de acarreo

Mapeo de funciones: S



2. Cadenas de acarreo

Mapeo de funciones: Cout

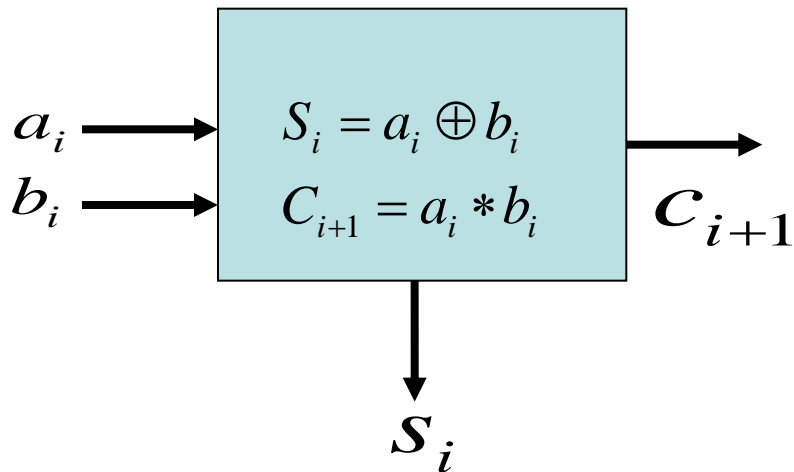


2. Cadenas de acarreo

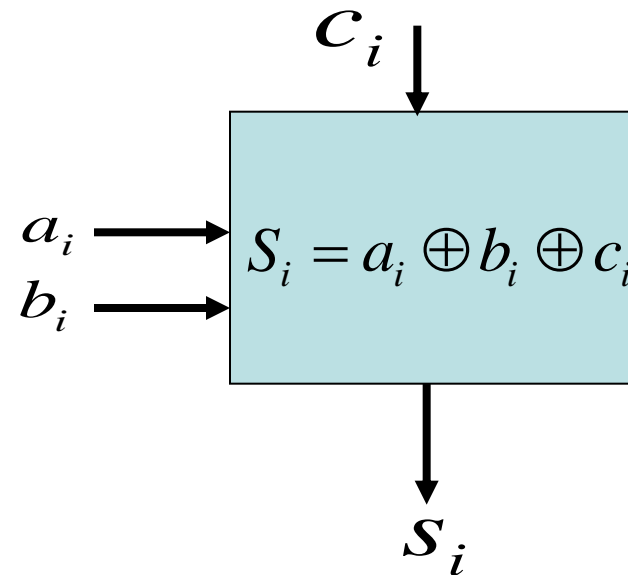
Adder de 4 bits en XC5200

- Se usan dos LC's. Uno implementa el sumador parcial. EL adyacente, obtiene la suma total.

Semi sumador

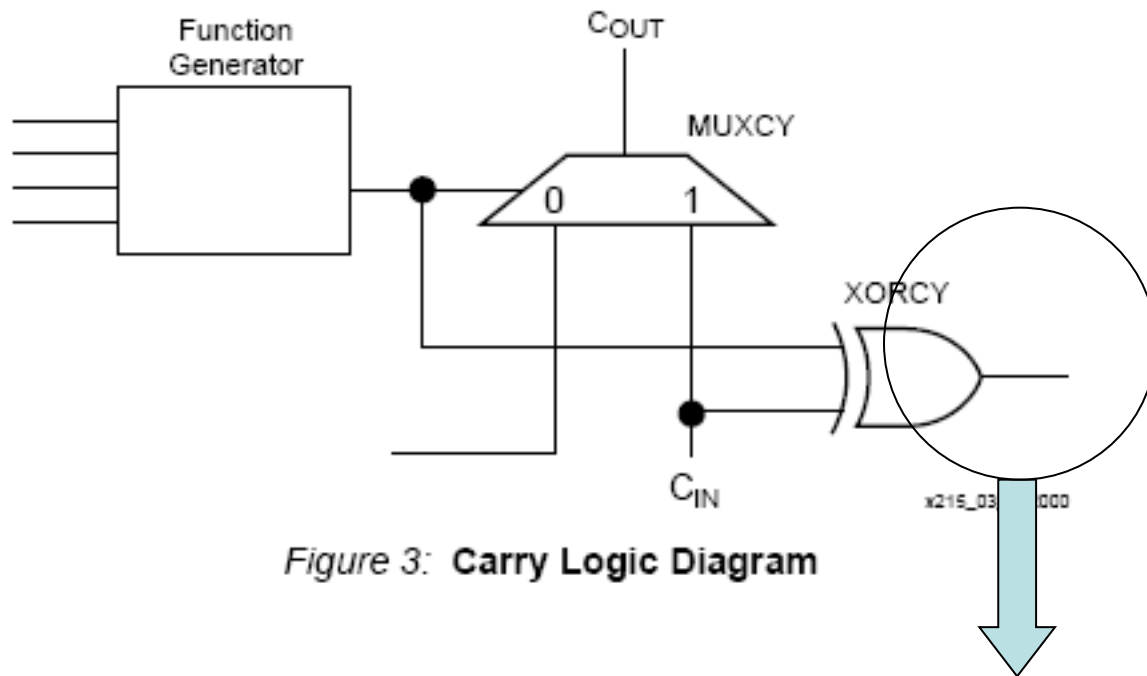


Sumador Total



$$C_{i+1} = a_i b_i + (a_i \oplus b_i) C_i$$

2. Cadenas de acarreo Slice en Virtex

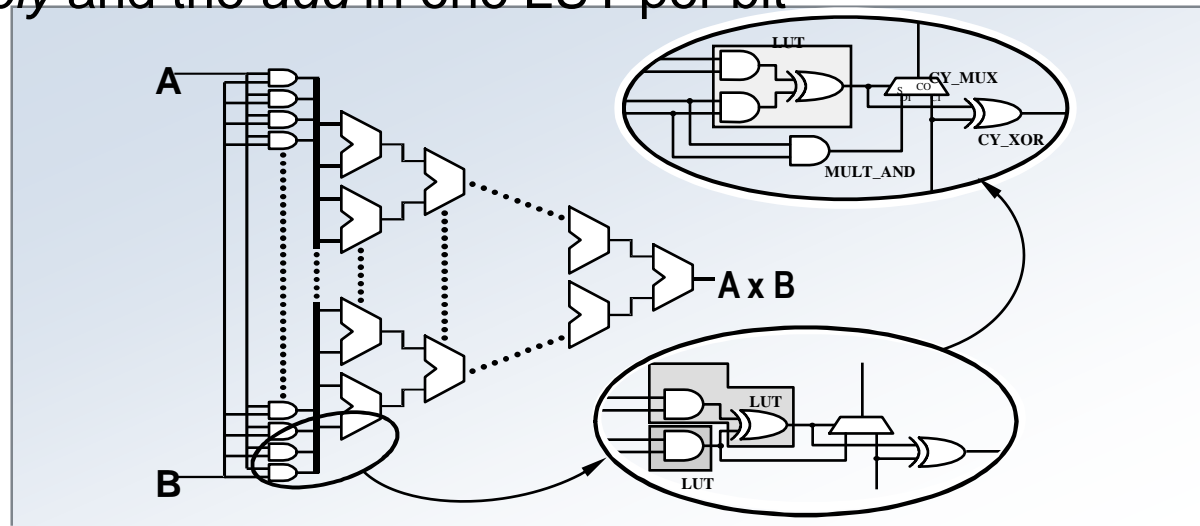


Se agrega la puerta XOR para implementar una suma total en un solo slice

3. Multiplier AND Gate

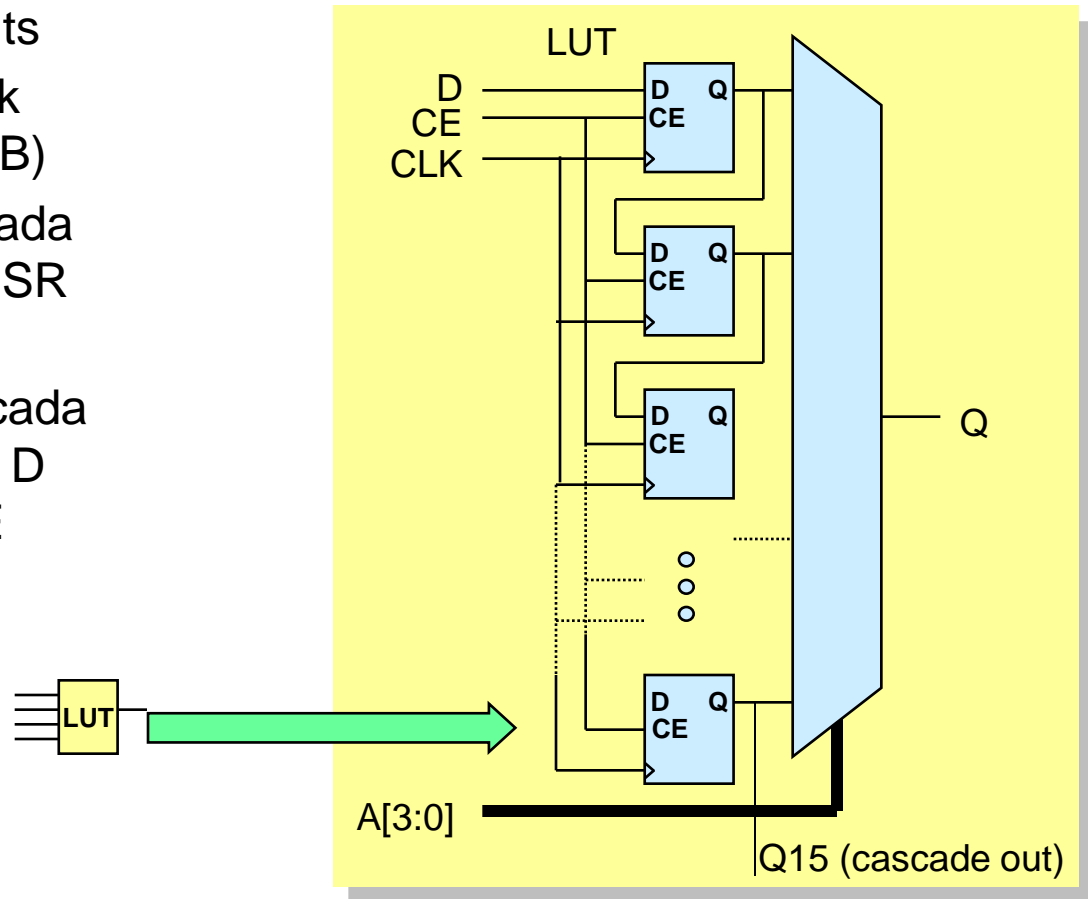
Speed up LUT-based multiplication

- Highly efficient multiply and add implementation
 - Earlier FPGA architectures require two LUTs per bit to perform the multiplication and addition
 - The MULT_AND gate enables an area reduction by performing the *multiply* and the *add* in one LUT per bit



4. Shift Register LUT (SRL16CE)

- Una LUT de 4 entradas se puede configurar como un SR de 16 bits
 - Retardo máximo de 16 clock cycles por LUT (128 por CLB)
 - Se puede construir en cascada con otras LUTs obteniendo SR mas
 - Hay una conexión dedicada desde Q15 a la entrada D del siguiente SRL16CE
 - Solo puede cargarse serialmente. Cuando un dato entra, otro sale.
 - No hay set ni reset



4. SRL16CE

1 Shift Chain
in CLB

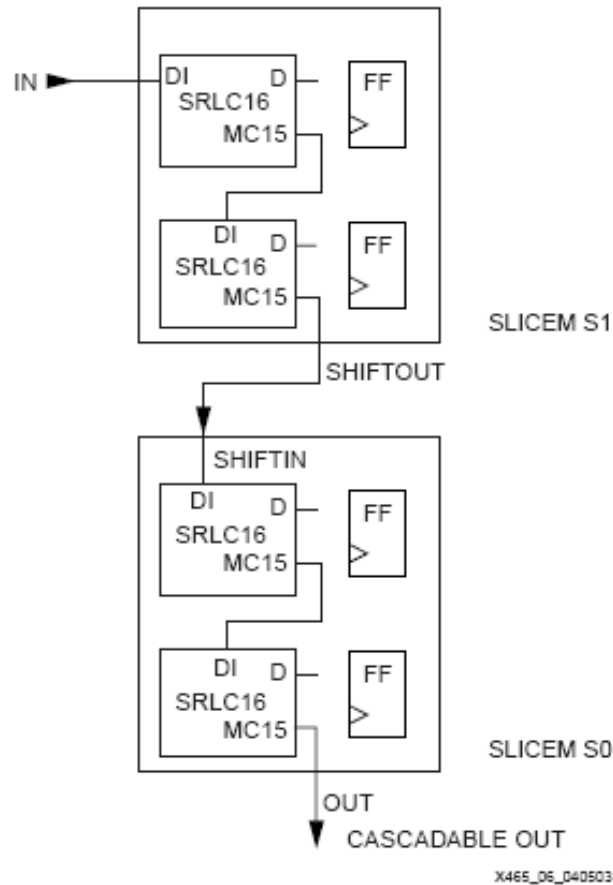


Figure 6: Cascading Shift Register LUTs in a CLB

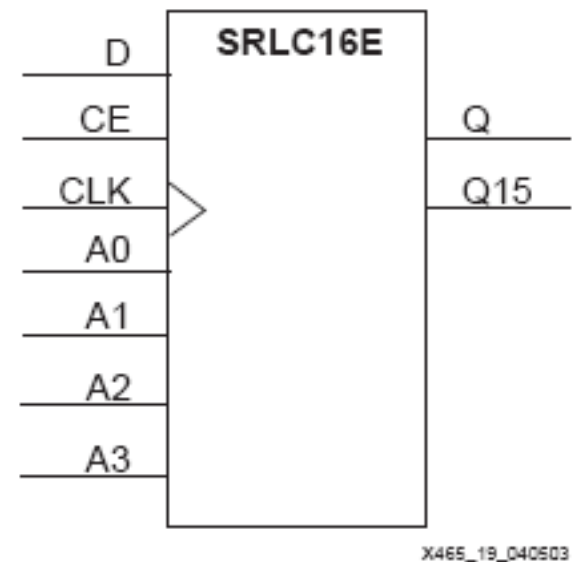


Figure 8: SRLC16E Primitive

4. SRL16

Verilog Inference Code

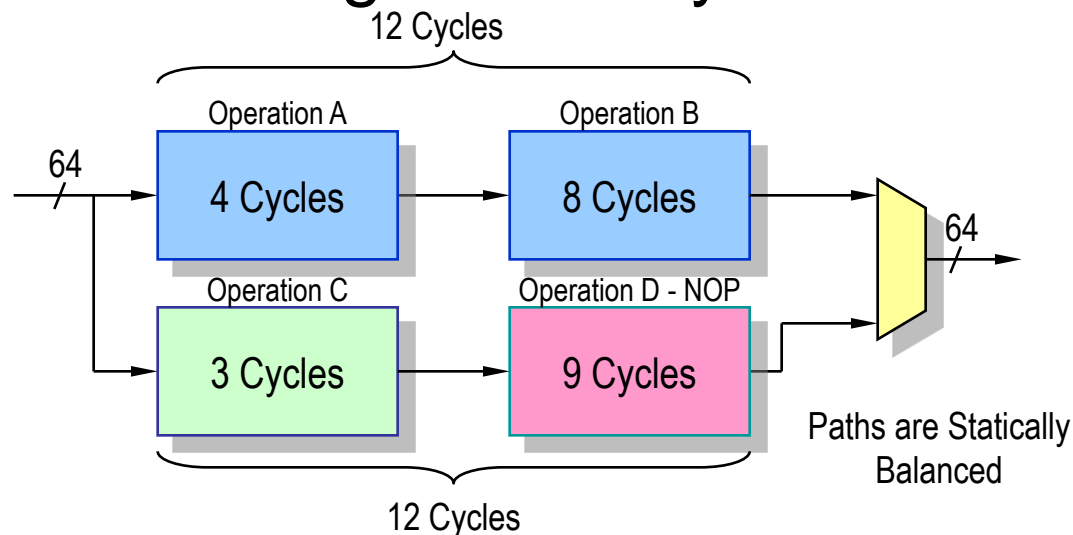
The following code infers an SRL16 in Verilog.

```
always @ (posedge C)
begin
    Q_INT <- {Q_INT[14:0],D};
end

always @(Q_INT)
begin
    Q <- Q_INT[15];
end
```

Shift Register LUT Example

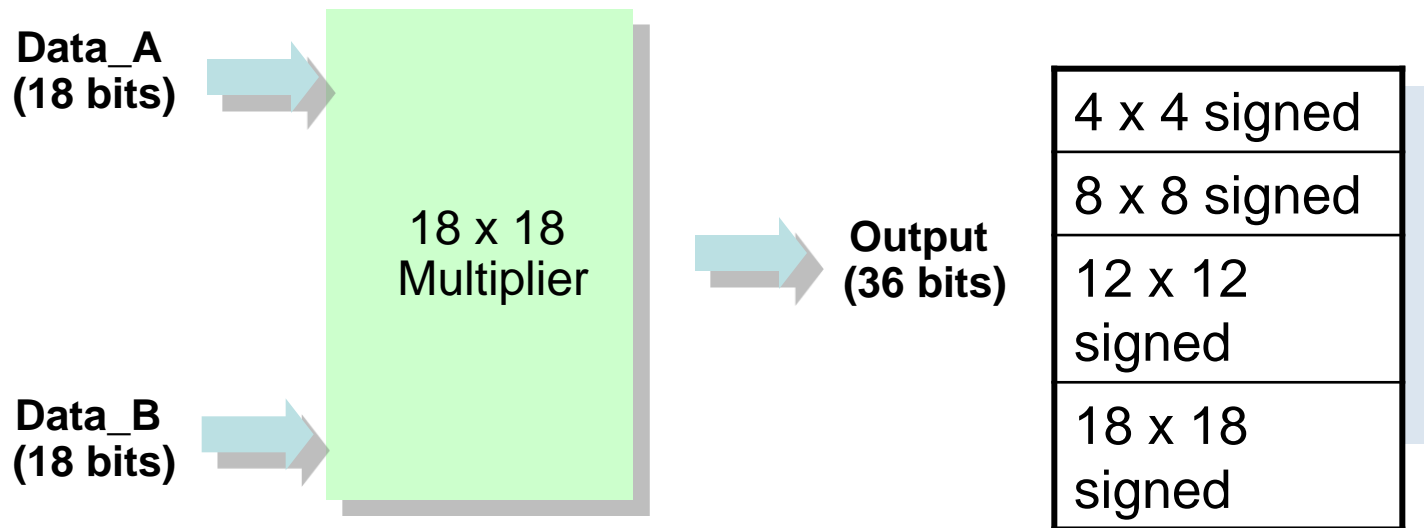
- The SRL can be used to create a No Operation (NOP)
 - This example uses 64 LUTs (8 CLBs) to replace 576 flip-flops (72 CLBs) and associated routing and delays



5. Embedded Multiplier Blocks

Saves from having to use LUTs to implement multiplications and increases performance

- 18-bit twos complement signed operation
- Optimized to implement Multiply and Accumulate functions
- Multipliers are physically located next to block SelectRAM™ memory

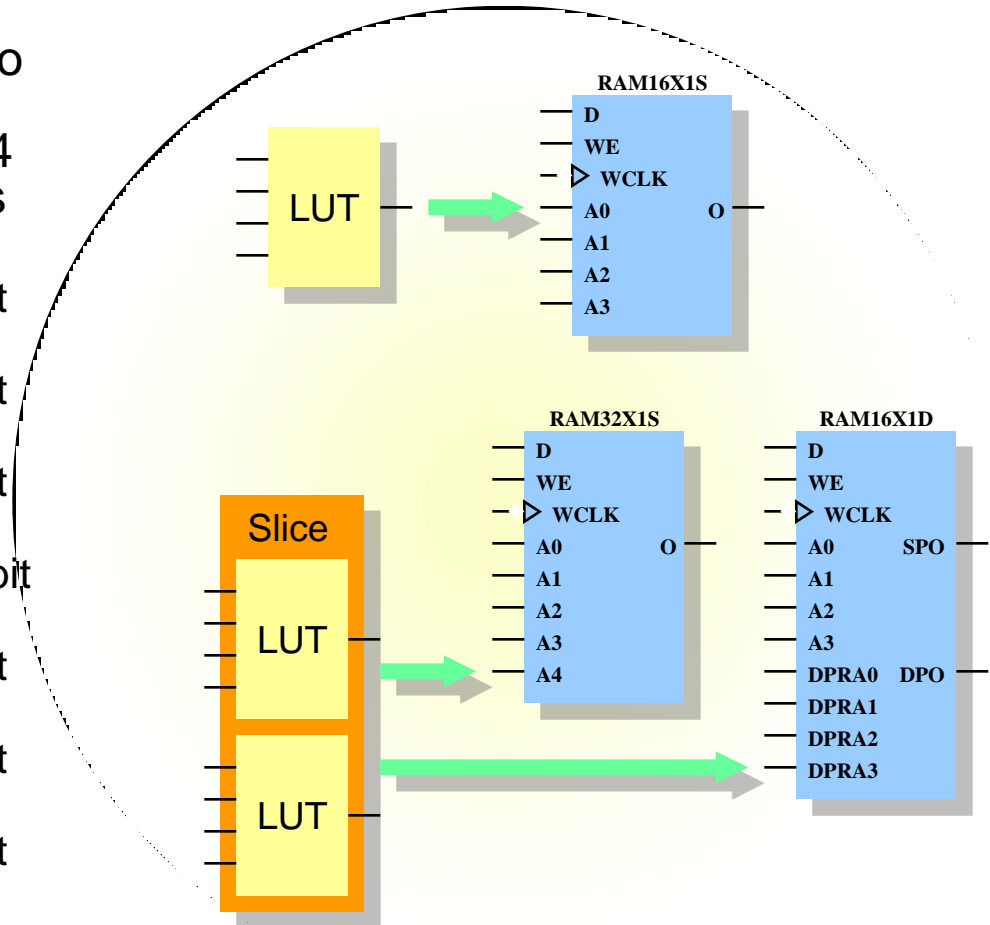


Elementos Secuenciales

- Pueden ser flip-flops o latches
- Dos en cada slice; ocho en cada CLB
- Las entradas provienen de las LUTs o de otro CLB
- Controles de Set y Reset independientes y
 - Pueden ser síncronos o asíncronos
 - Son compartidos dentro del slice, pueden ser invertidos localmente.

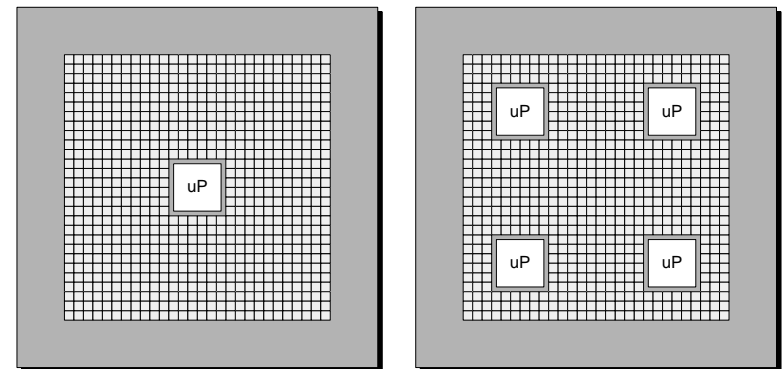
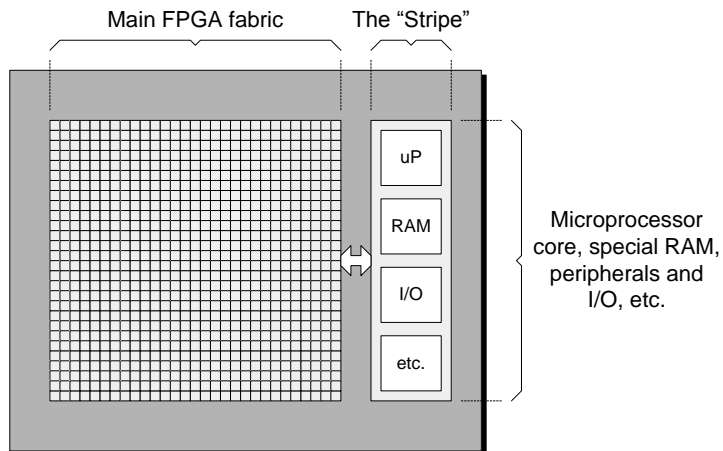
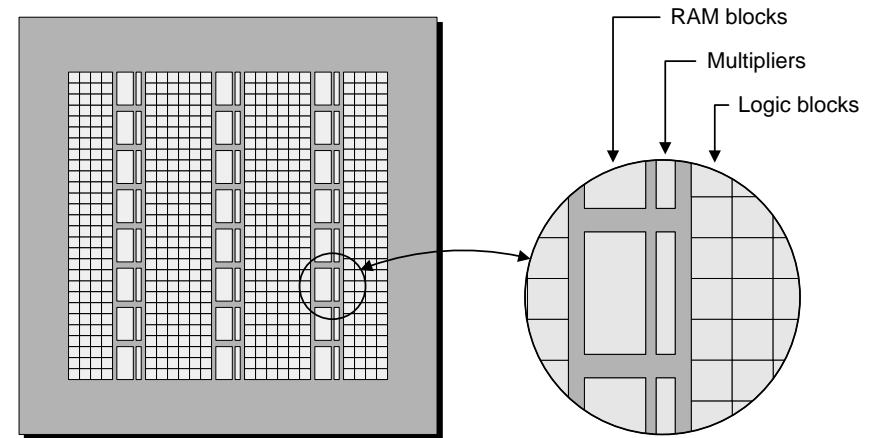
RAM Distribuida (Xilinx)

- Cada 4 bit LUT puede ser usado como una RAM de 16 x 1 bit → todas las LUTs de un CLB (de 4 slices) pueden ser configuradas juntas para implementar:
 - Single Port RAM 16 x 8 bit
 - Single Port RAM 32 x 4 bit
 - Single Port RAM 64 x 2 bit
 - Single Port RAM 128 x 1 bit
 - Dual Port RAM 16 x 4 bit
 - Dual Port RAM 32 x 2 bit
 - Dual Port RAM 64 x 1 bit



Bloques Embebidos en FPGA

- Memoria (Block RAM)
- Multiplicadores
- Sumadores
- MAC
- Cores de Procesadores, hard o soft

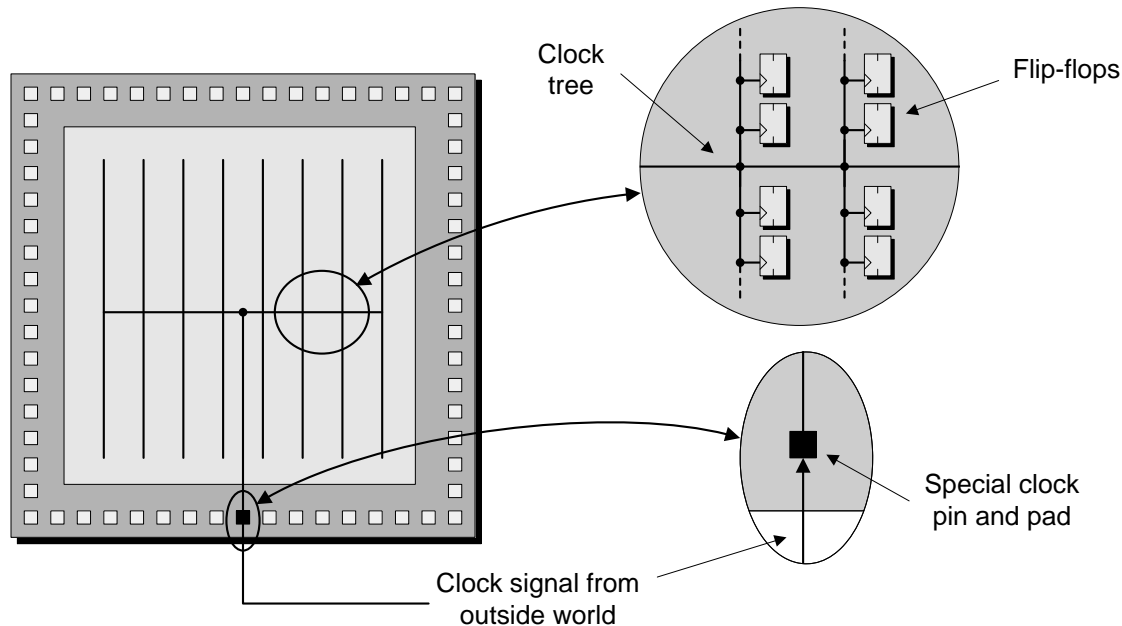


(a) One embedded core

(b) Four embedded cores

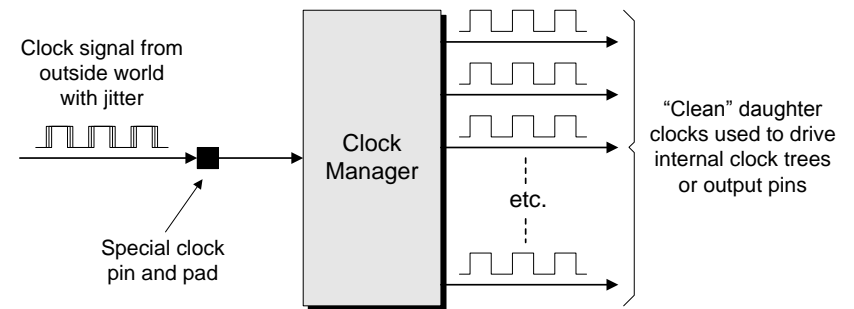
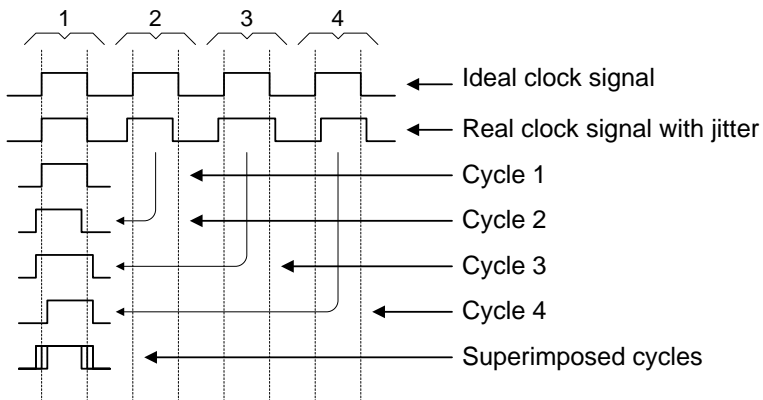
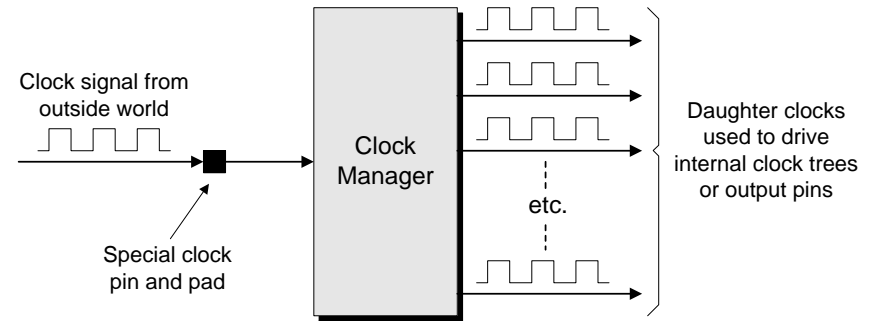
Relojes

- Todos los elementos síncronos necesitan un reloj.
- El reloj es una señal externa al FPGA, entra por un pin especial y es ruteada por el interior del dispositivo.
- Para poder acceder a todos los registros, la señal de reloj se va dividiendo como ramas de un árbol, donde las hojas son los flip-flops.
- Esto es así pues si la señal de reloj fuera una única y larga pista, entonces los flip-flops mas cercanos al pin del reloj verían la señal antes que otro situado al final de la cadena. (skew).
- El árbol del reloj se implementa con pistas especiales, separadas de la interconexión programable general.



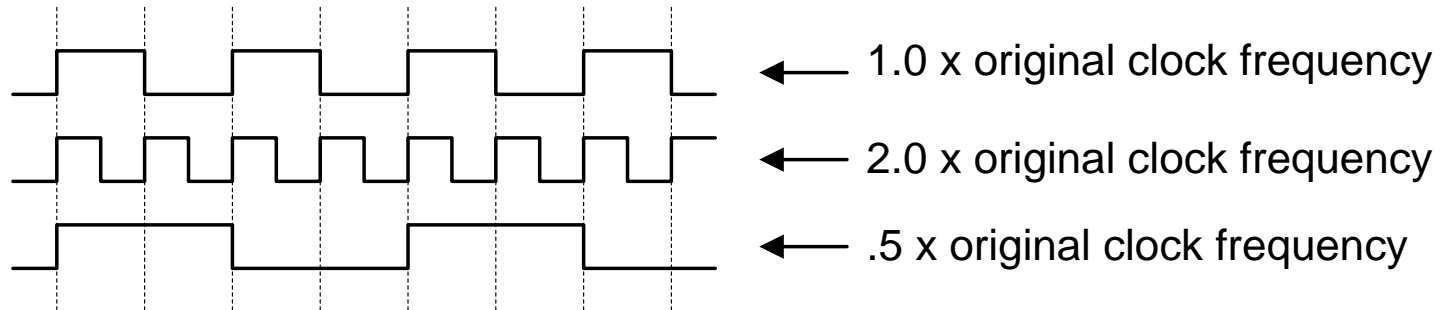
Clock Managers

- Un DCM (digital clock manager) es una función tal que recibe una señal de reloj externa y genera un número de señales hijas.
 - Se utiliza para: detectar y corregir el fenómeno conocido como “jitter”



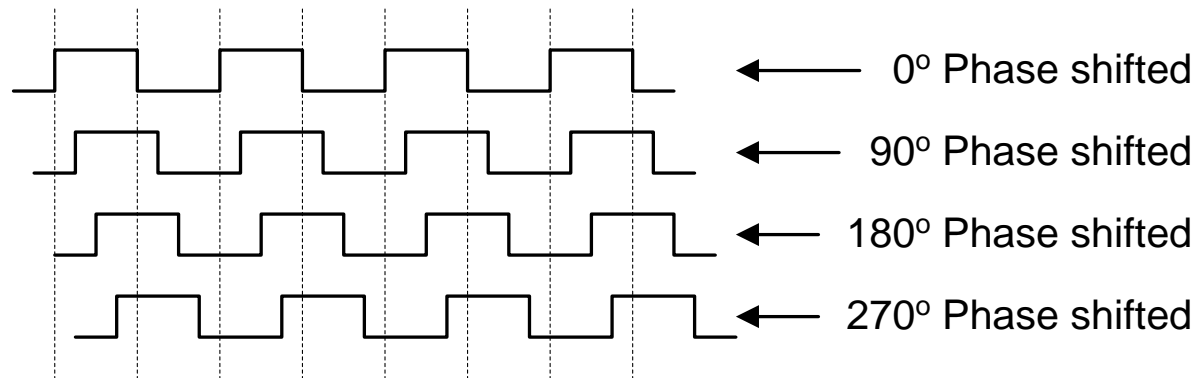
Clock Managers

- Otra aplicación:
 - síntesis de frecuencia: cuando la frecuencia que viene del exterior no es la frecuencia necesaria, el clock manager se usa para generar las frecuencias adecuadas (multiplicando o dividiendo la frecuencia externa)



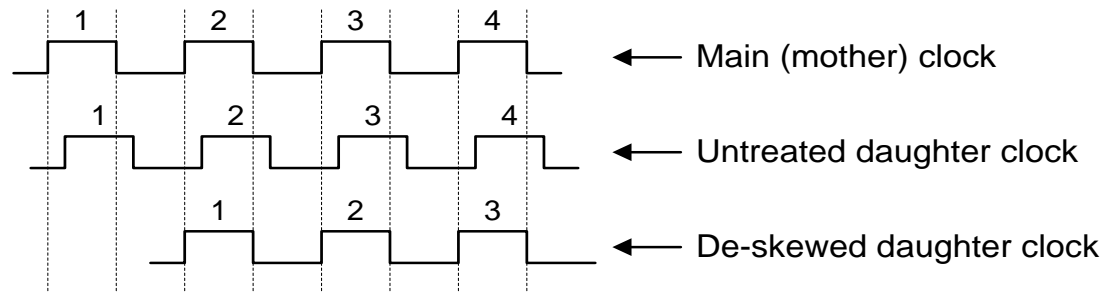
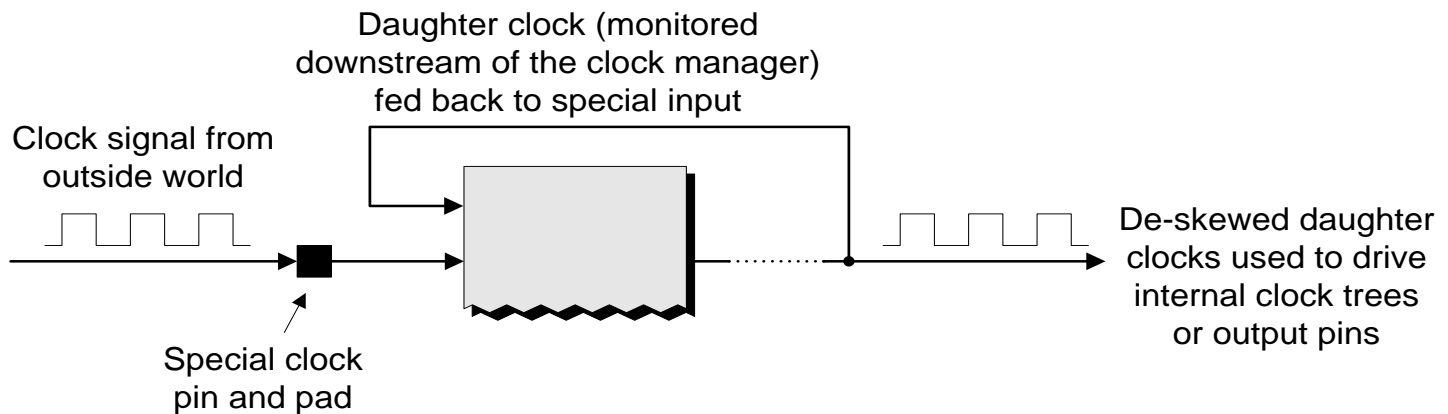
Clock Managers

- Otra aplicación:
 - desplazamiento de fase: el diseño puede requerir trabajar con frecuencias desplazadas unas respecto a las otras. Por ejemplo, se pueden desplazar tanto como uno quiera o bien valores fijos (120° y 240° , o bien 90° , 180° y 270°) dependiendo del clock manager



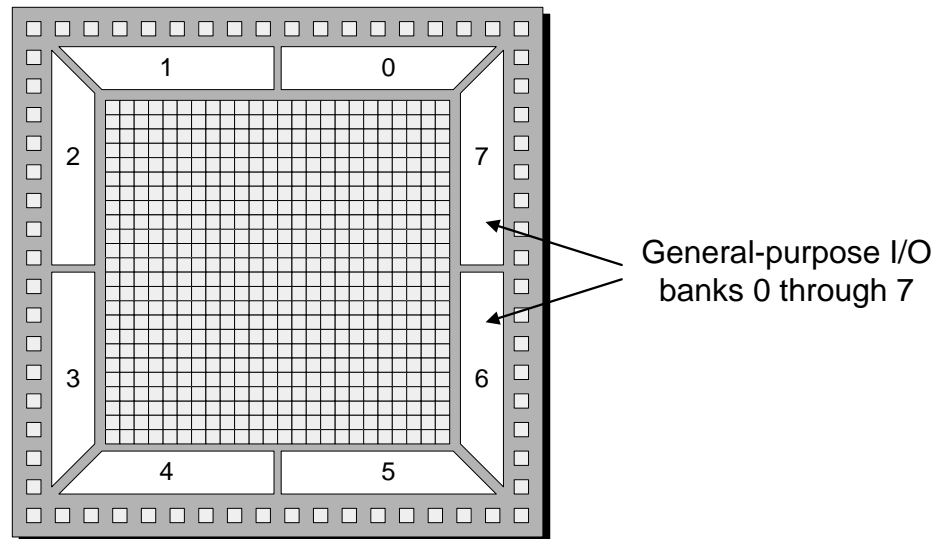
Clock Managers

- Otra aplicación:
 - corrección auto-skew: aunque el reloj hijo tenga la misma frecuencia que el externo, sufrirá retardos respecto a la señal original (skew). Esto es necesario corregirlo. Se realiza comparando la señal hija con el reloj externo.



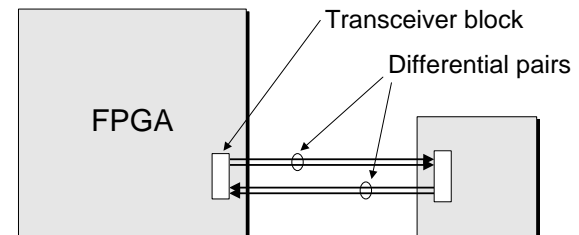
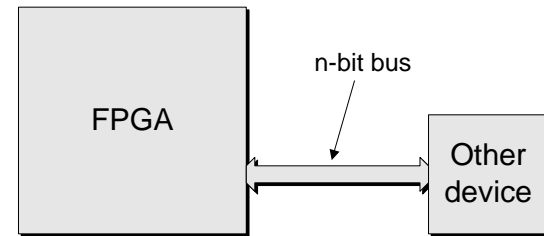
I/O Configurables

- Cada banco puede ser configurado para soportar un standard particular de I/O.
- Conforme la tecnología hace mas pequeños los transistores, es necesario una menor tensión... se pasa de 5 Volts (350nm) a los actuales 1.2 Volts (130nm). Esta tensión se denomina core voltage.
- Diferentes Standards de I/O pueden usar tensiones con voltajes diferentes al core voltage del FPGA.



Gigabits Transceivers

- Tradicionalmente la comunicación entre dispositivos se realiza a través de buses. Con el tiempo, la transferencia de grandes volúmenes de datos hizo necesario que los buses crezcan de 8, a 16, 32, 64 bits.... Y así sucesivamente. El problema es que esto requiere muchísimos pines, y pistas. Uno de los problemas es el ruteo de esas pistas para que todas tengan la misma longitud e impedancia. Se hace difícil, además, manejar la integridad de la señal.
- Los modernos FPGA no se comunican mediante buses sino mediante unos bloques especiales llamados gigabit transceivers. Estos bloques usan un par de señales diferenciales para transmitir (TX) y otro par para recibir (RX). Estos transceivers operan a velocidades altísimas. (a Gigabits por segundo). Cada bloque, además, soporta varios de estos transceivers



Hard IP, Soft IP, Firm IP

- IP: Intellectual Property. Son bloques funcionales diseñados previamente. Pueden ser tan complejos como un micro.
- Hard IP: microprocessor cores, gigabit interfaces, multiplicadores, funciones MAC, etc. Diseñados para ser eficientes en cuanto a : bajo consumo, area ocupada, rendimiento. Cada FPGA tiene una combinación de estos bloques combinados con los bloques de lógica programable.
- Soft IP: son librerías de funciones (Verilog, VHDL o RTL) que pueden ser incluídas en los diseños.
- Firm IP: son librerías de funciones pero que ya están mapeadas, ubicadas y ruteadas de forma óptima en un grupo de bloques programables.

Actores (vendedores)

- FPGA

- Actel Corp. www.actel.com
- Altera Corp. www.altera.com
- Anadigm Inc. www.anadigm.com
- Atmel Corp. www.atmel.com
- Lattice Semiconductor Corp. www.latticesemi.com
- Leopard Logic Inc. www.leopardlogic.com
- QuickLogic Corp. www.quicklogic.com
- Xilinx Inc www.xilinx.com

- FPNA

- Elixent Ltd www.elixent.com
- Ipflex Inc www.ipflex.com
- Motorola www.motorola.com
- PACT XPP Technologies AG www.Pactxpp.com
- Pico Chip Designs Ltd. www.picochip.com
- QuickSilver Technologie Inc www.Qstech.com

Equivalencias con Altera

- Logic Cell (LC) de Xilinx \leftrightarrow Logic Element (LE) de Altera
- Slice de Xilinx \leftrightarrow no hay equivalente
- CLB de Xilinx \leftrightarrow LABs (Logic Array Block) de Altera
- Cada LAB está formado por 10 LE.

LE (Logic Element) de Altera

