



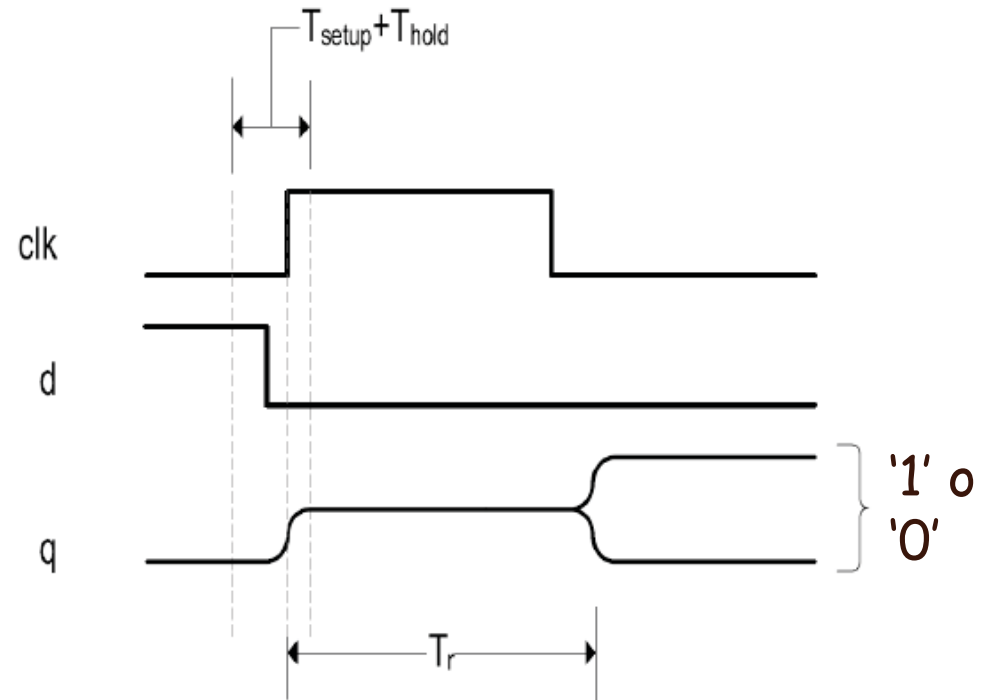
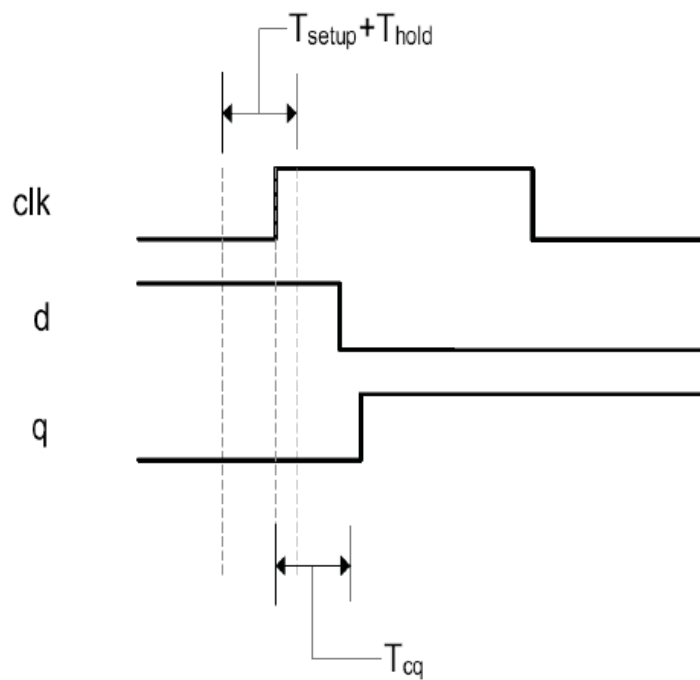
FPGA Design Techniques

Cristian Sisterna, MSc
Universidad Nacional de San Juan
Argentina

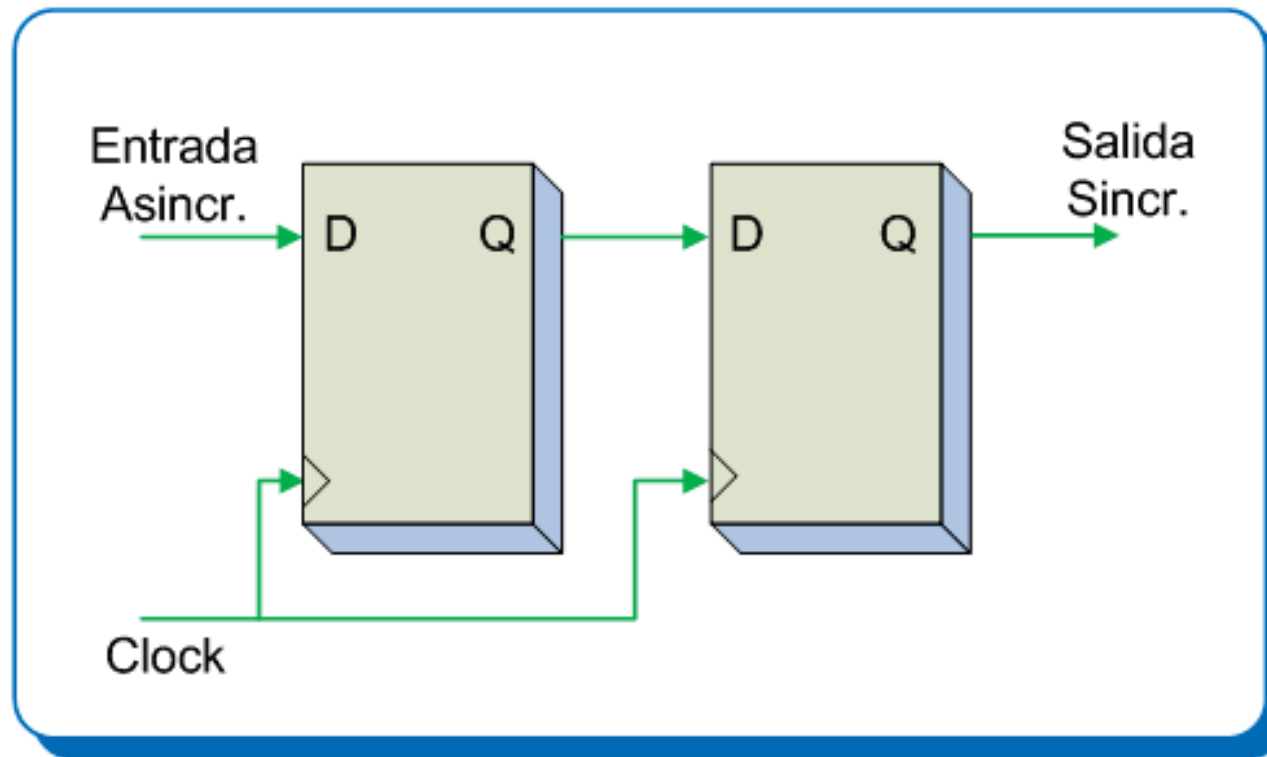


DESIGN WITH DIFFERENT CLOCK DOMAINS

Metaestability



Basic Synchronizer





Synchronization Circuit

- Synchronize an asynchronous input with the system (FPGA) clock
- There is not circuit that **can prevent** metaestability
- Synchronizer provides a time for resolving the metaestability



Synchronizer Failures

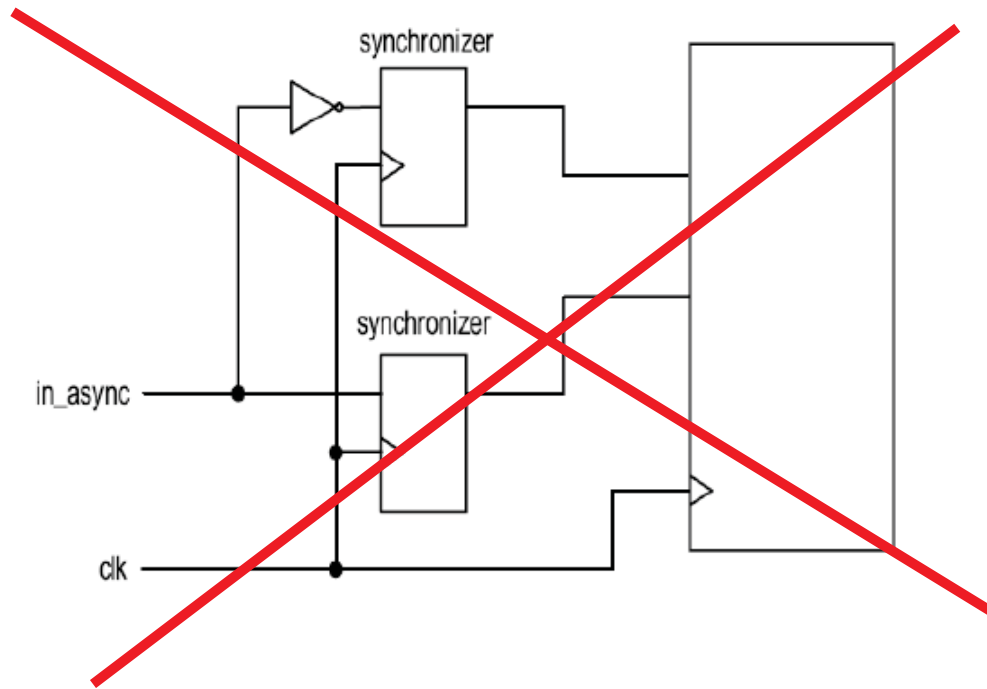
- Given the same parameters for a synchronizer the Medium Time Between Failures (MTBF) is:
 - One flip-flop synchronizer: 1 año
 - Two flip-flops synchronizer: 3000 años
 - Three flip-flops synchronizer : 6 billones de años



Correct Use of the Synchronizer

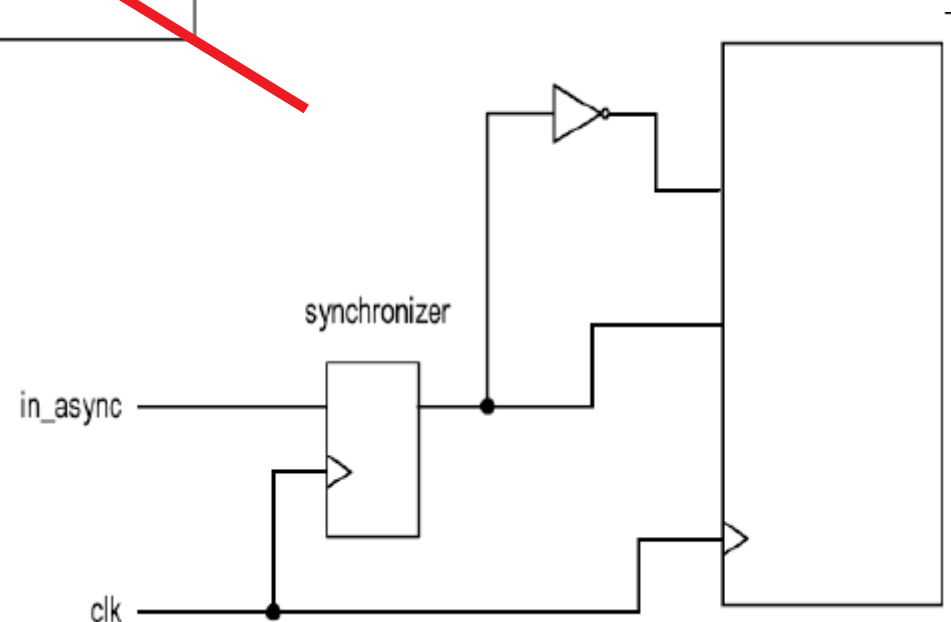
- Use en stable input signal (glitch-free)
- Synchronize the signal in only one place
- Avoid the synchronization of multiple related signals
- Re-analyze the synchronizer after every change in your design

Synchronization Example

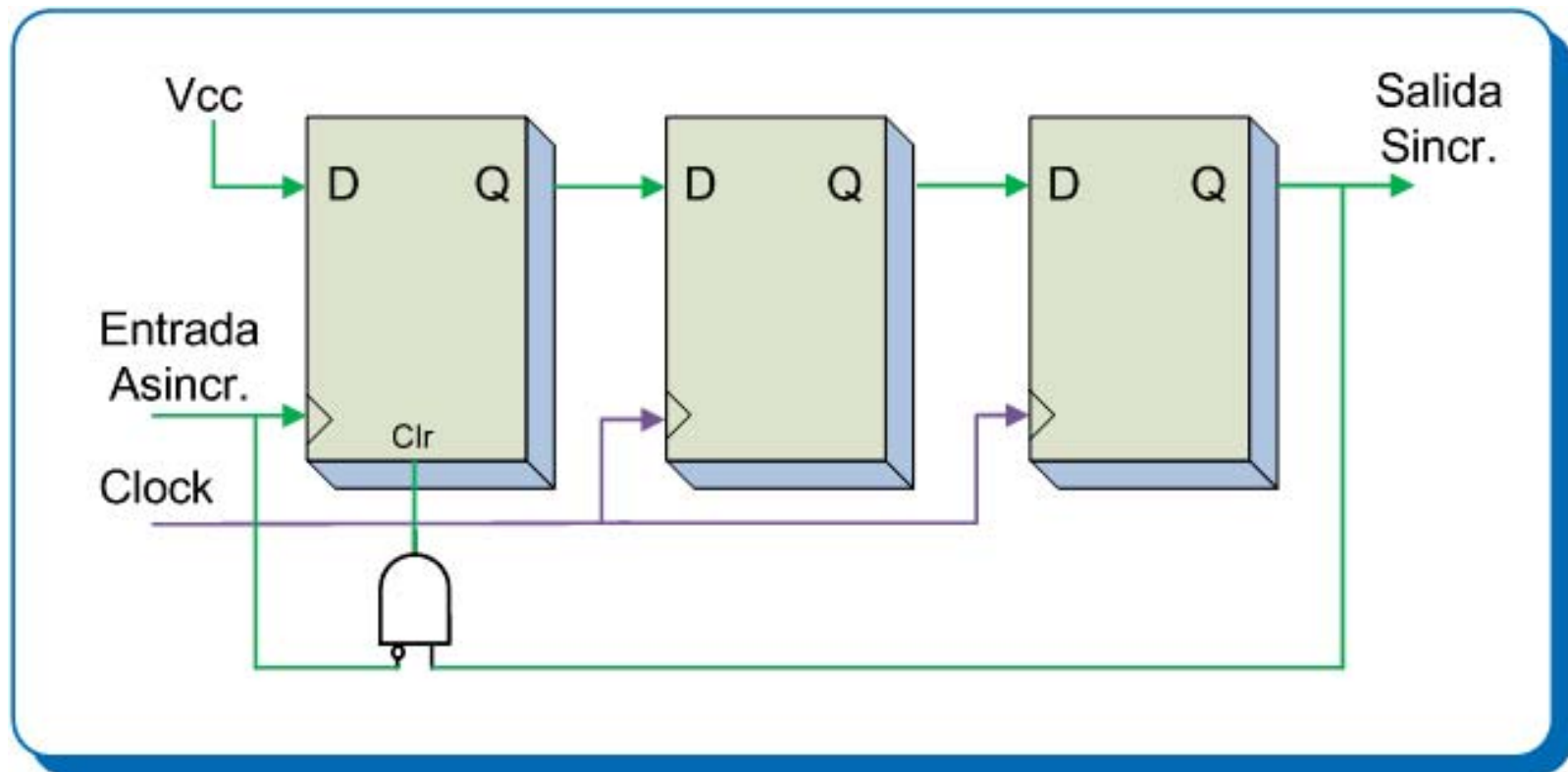


Synchronization
in two places

Synchronization
in one place

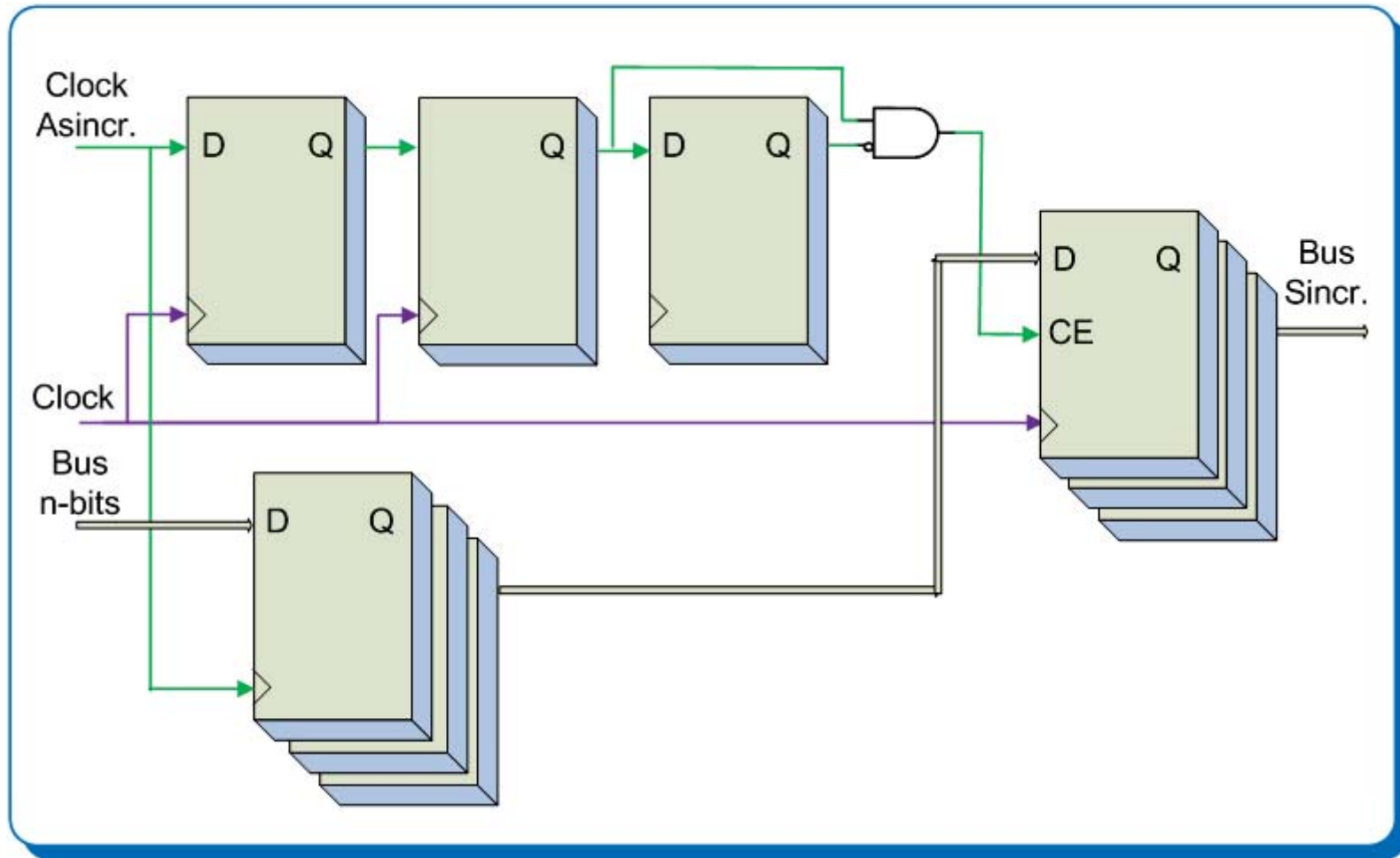


Synchronizer for Narrow Input Signal



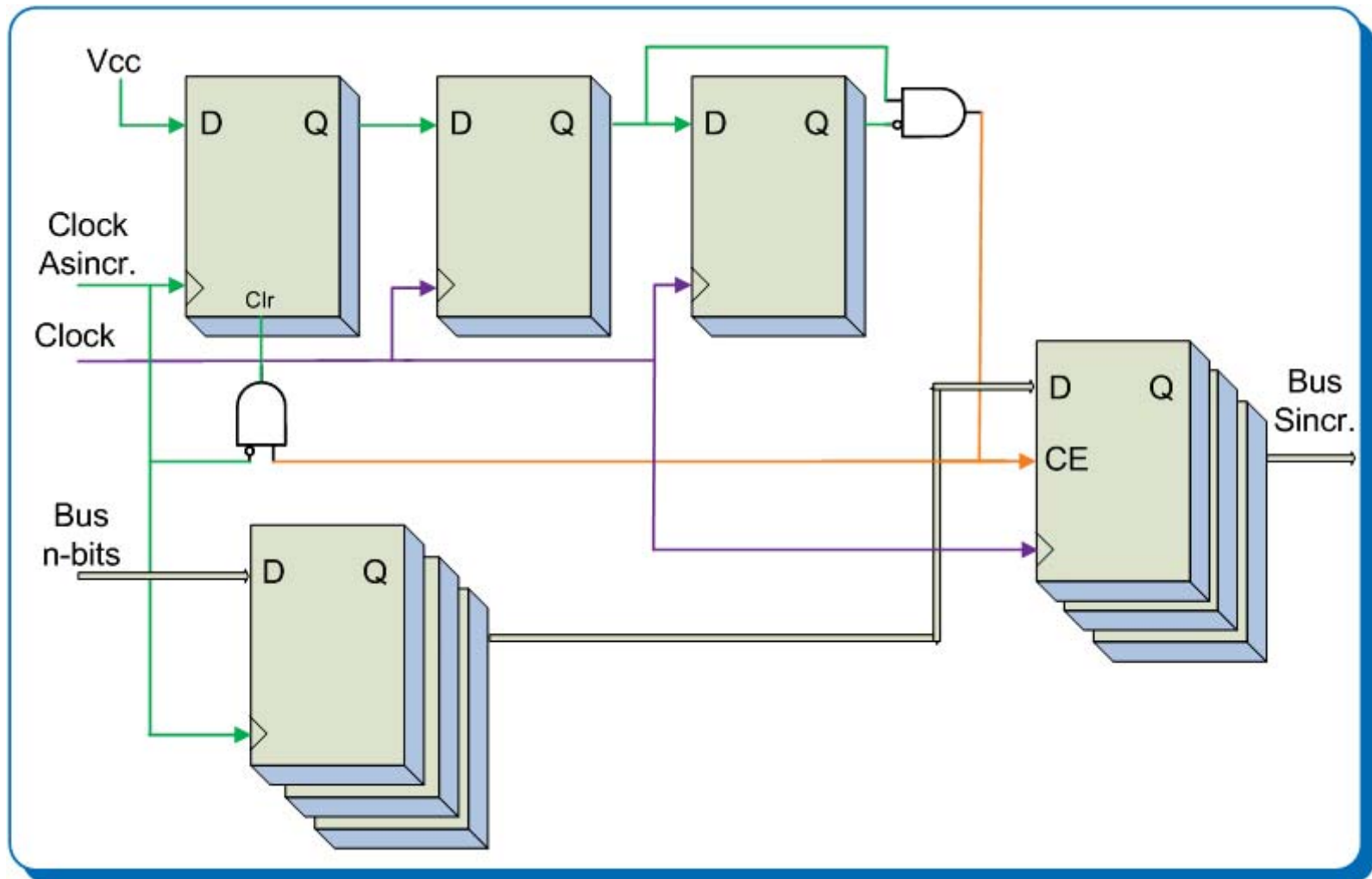
Basic Bus Synchronizer

- From a low frequency to a high frequency



Bus Synchronizer

- From a high frequency to a low frequency

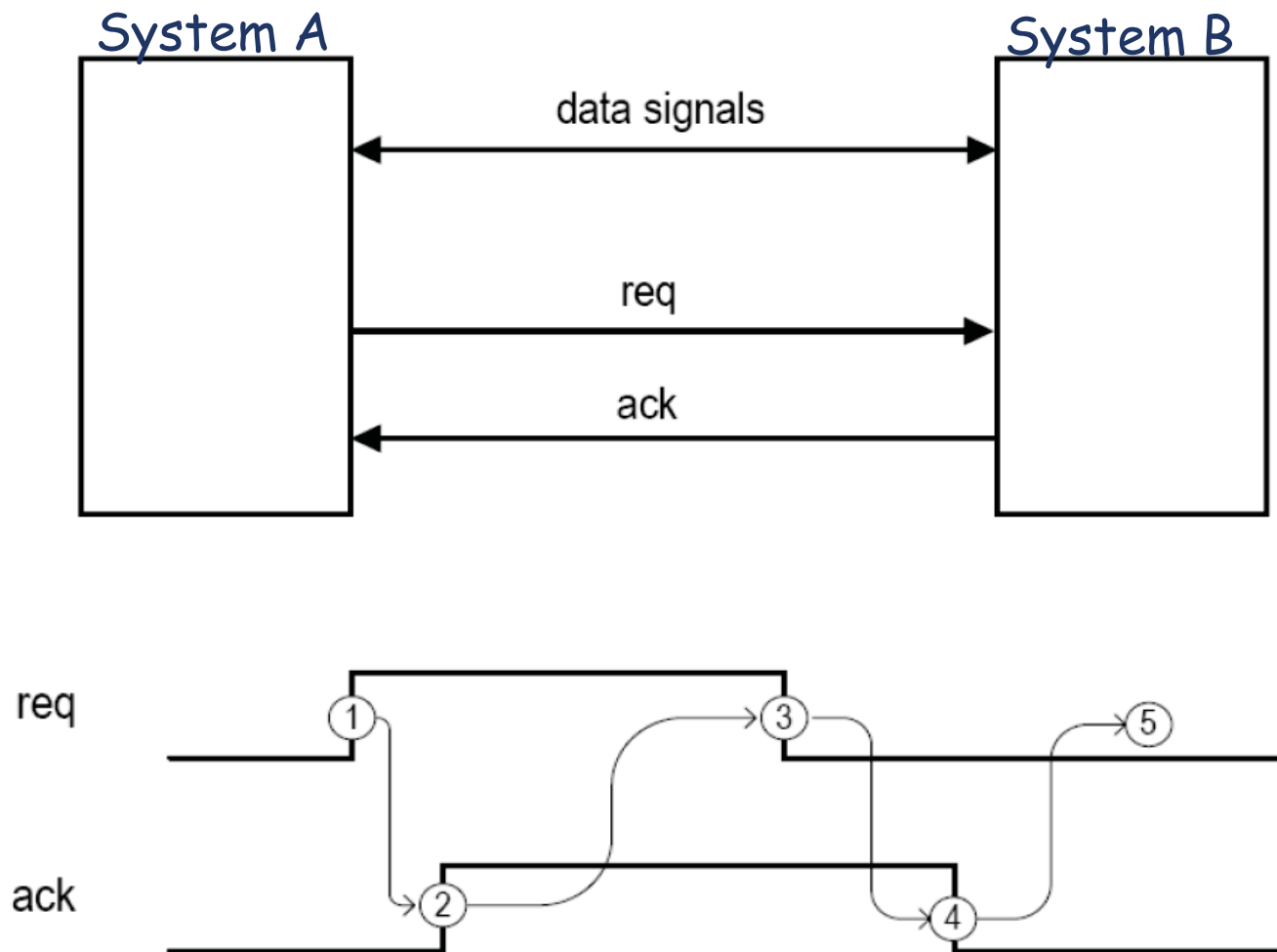




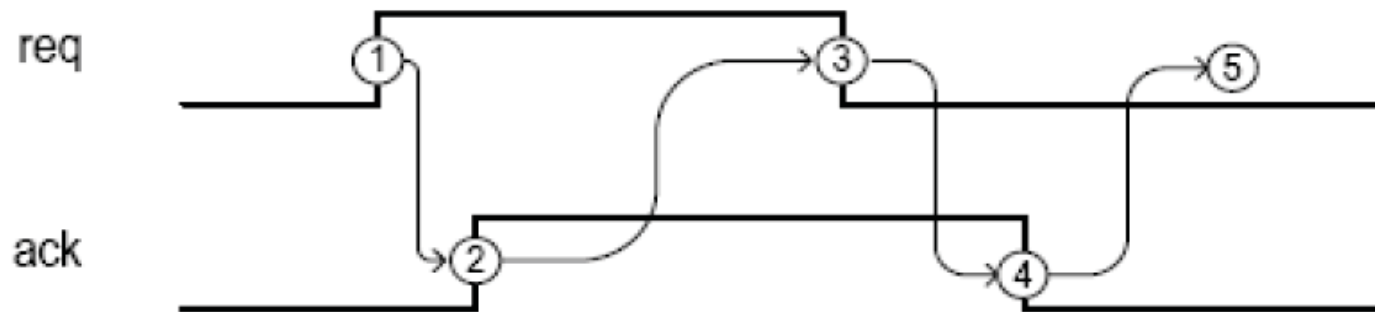
Tx/Rx Protocol Between Systems

- Problemas to face:
 - Asynchronous systems
 - No relation between the systems
- There are several scheme, one of the most used is the ‘For Phases’”

Four Phases Protocol

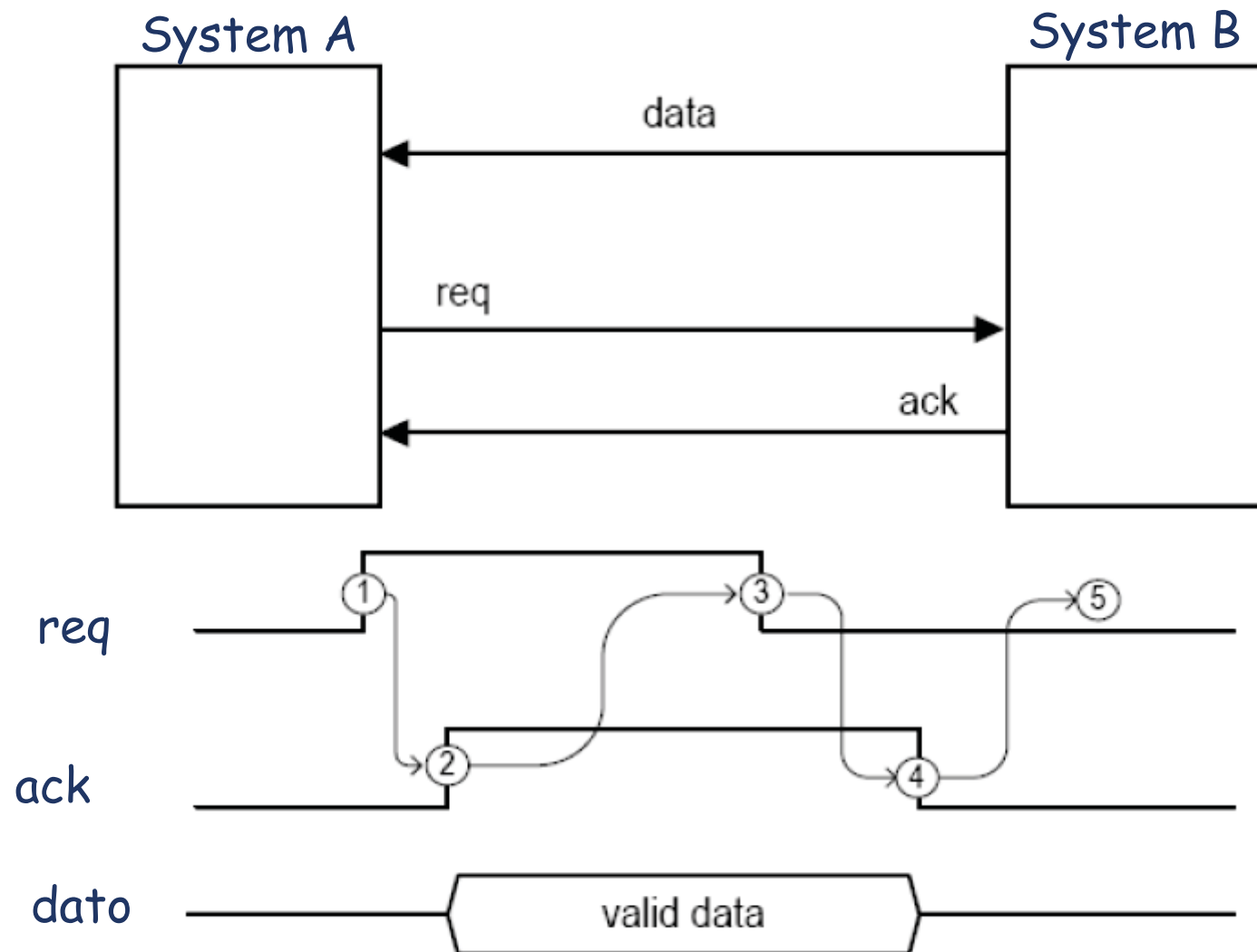


Four Phases Protocol



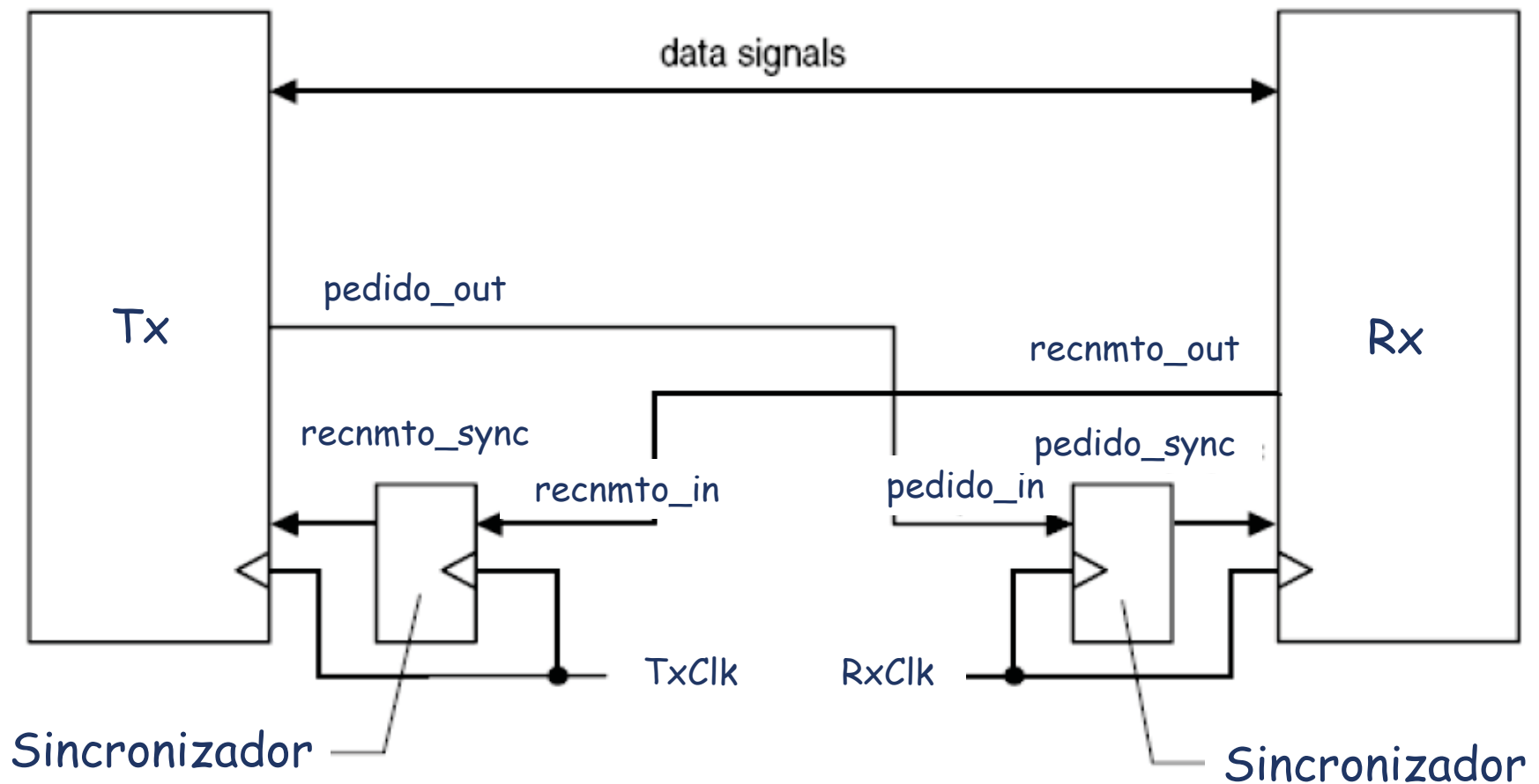
- Phase 1: Tx activate the request
- Phase 2: Rx activate the acknowledge of the request
- Phase 3: Tx de-activate the request
- Phase 4: Rx deactivate the acknowledge
- Tx can start a new request

Four Phases Protocol



Four Phases Protocol

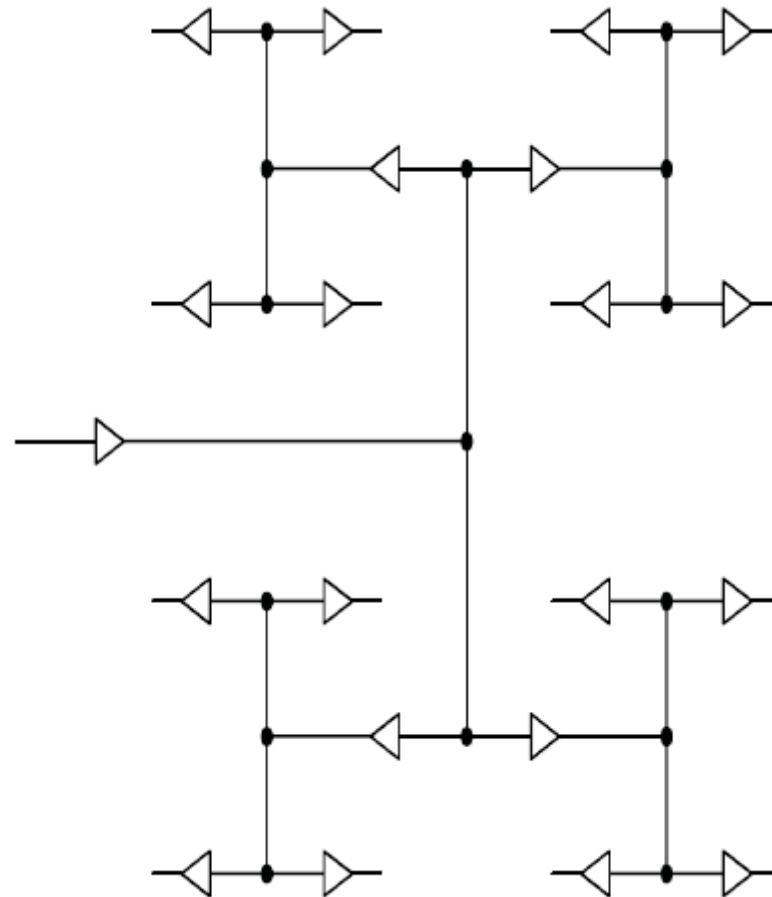
Importan: If Tx and Rx are in different clock domains, use synchronizers



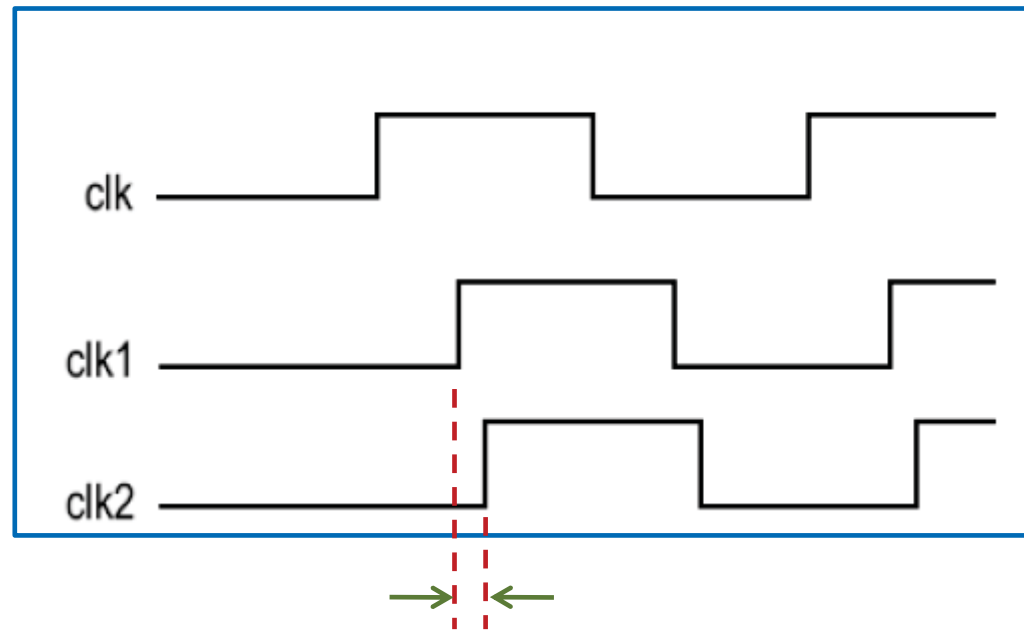
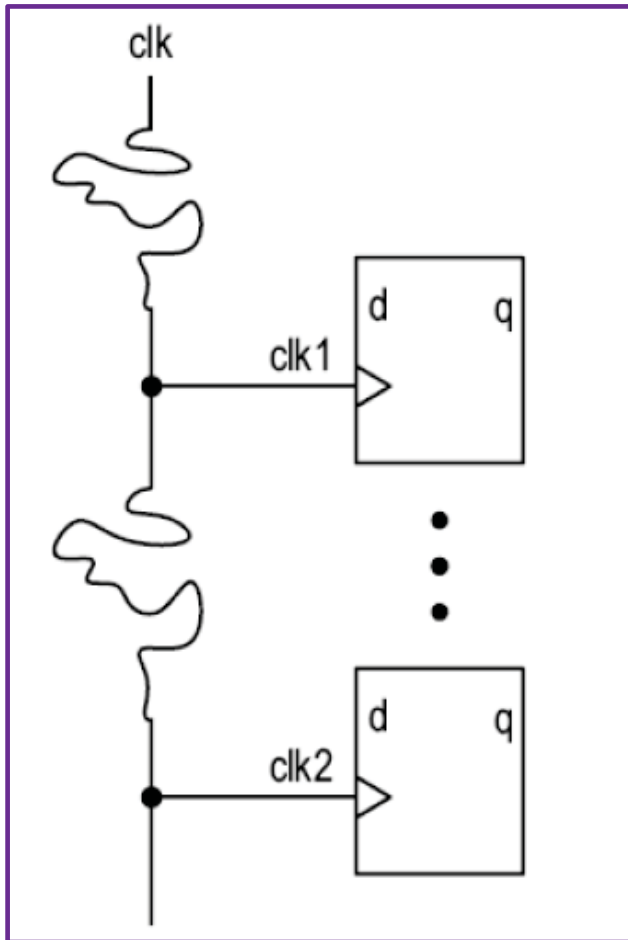


CLOCK: RESOURCES AND ROUTING

FPGA Clock Routing



FPGA Clock Skew



Skew between Clk1 and Clk2



Proyecto con Múltiples relojes

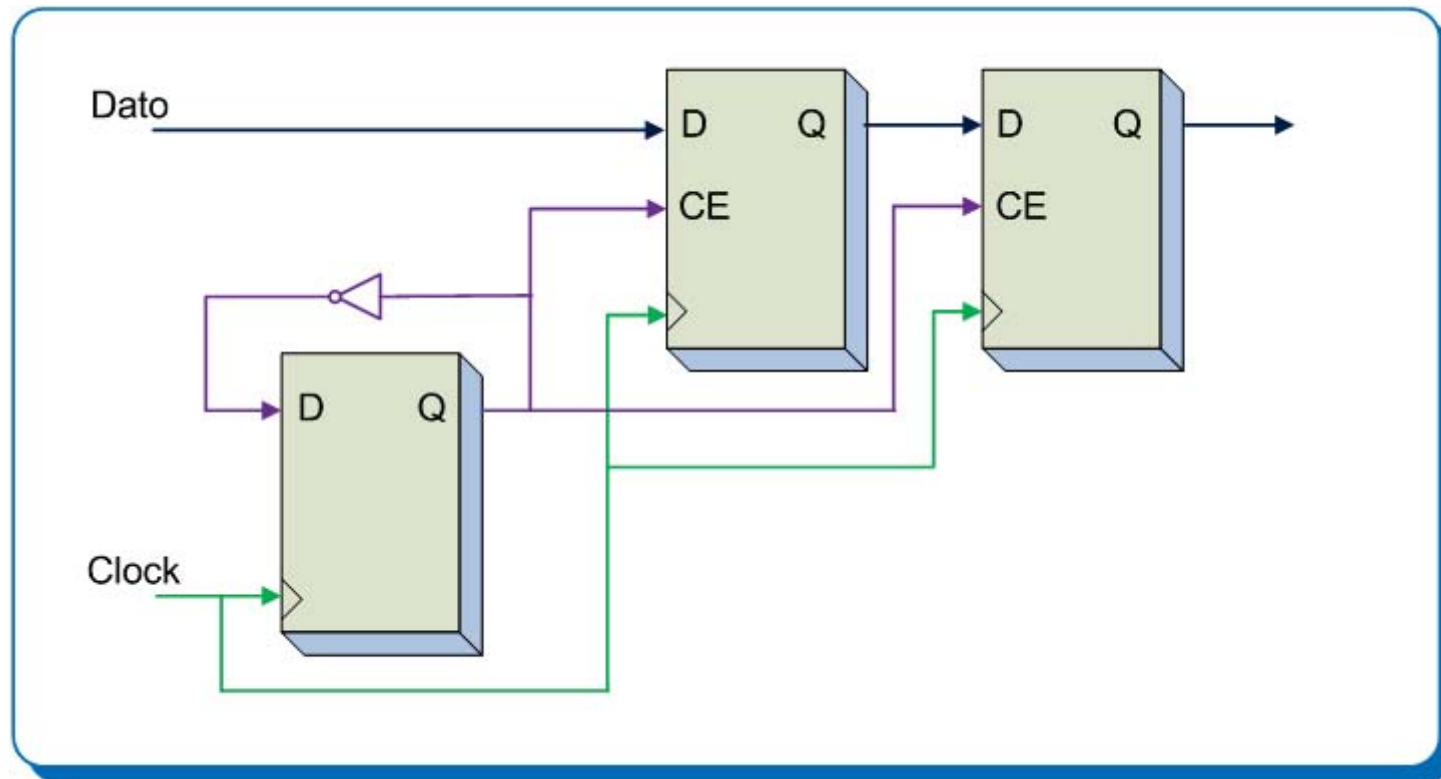
- Interface con otros componentes externos al FPGA
 - Comunicación
- Complejidad del proyecto
 - Desvío de reloj se incrementa con el números de FF
 - FPGA con 16-bit 20 MHz processor, 1-bit 100 MHz serial interface, 1 MHz I/O controller
- Consumo de Potencia



Relojes Derivados vs Independientes

- Relojes independientes:
 - Relación entre relojes es desconocida
- Relojes derivados
 - El reloj es derivado desde otra señal de reloj
 - La relación entre relojes es conocida
 - La lógica para el reloj derivado debería ser separada de la lógica regular y usar opciones de síntesis y P&R específicos
 - El sistema que usa el reloj derivado debe ser tratado como un sistema '*sincrónico*'

Uso de Divisor de Frecuencia - CE





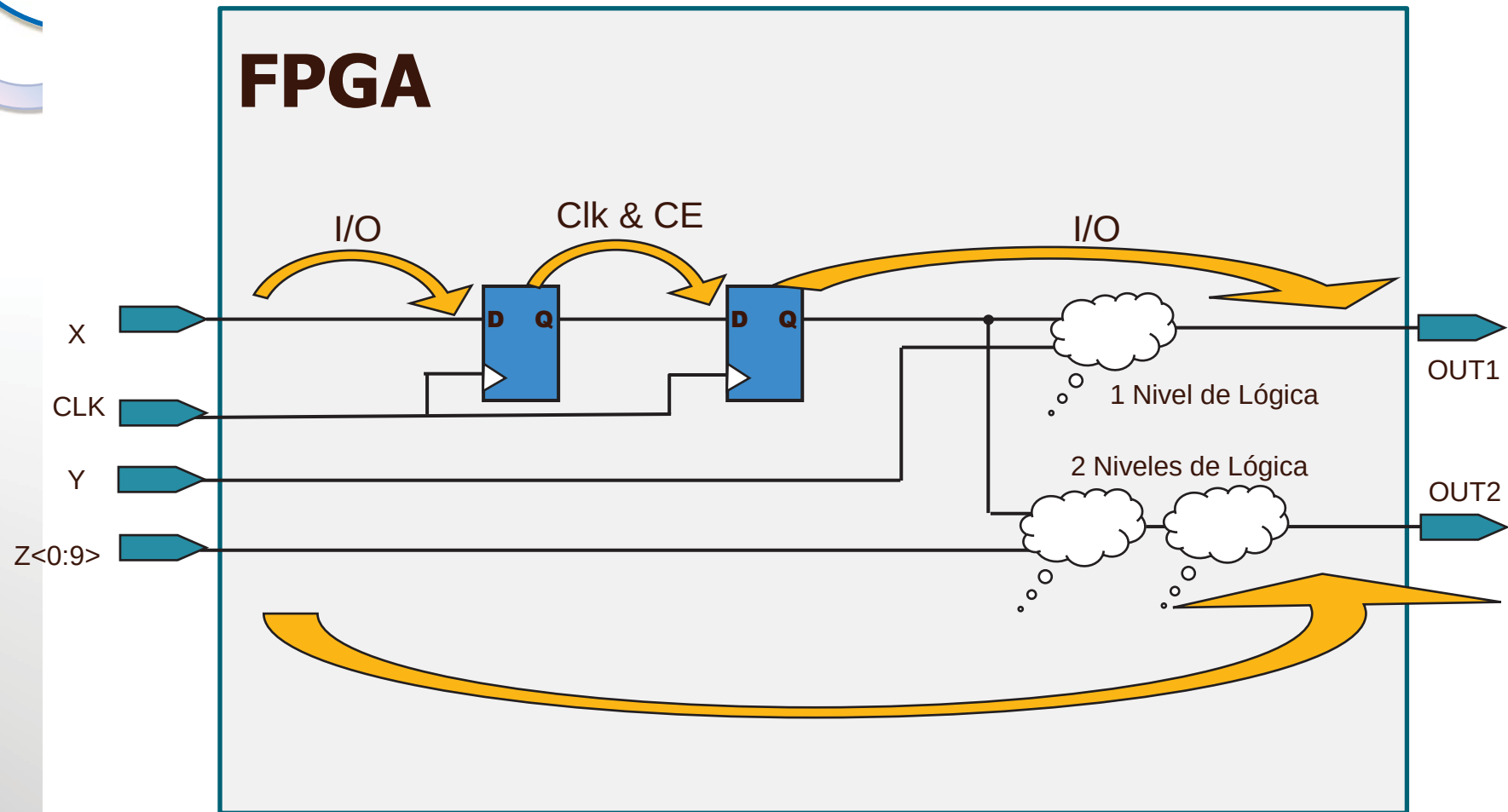
° **RESTRICCIONES DE IMPLEMENTACIÓN**



Restricciones de Implementación

- Restricciones de Tiempo
- Restricciones de Colocación
- Restricciones de Hard-IP

Restricciones de Tiempo—Que? Como?





Restricciones de Tiempo Típicas

- Frecuencia/Periodo
 - *Cada reloj dentro del FPGA debe tener su propia restricción*
- Restricción de tiempo de establecimiento de entradas
- Restricción de tiempo de salida de las salidas
- Bloqueo de una <net> en particular
- Usar restricciones de multi-ciclo de reloj en caso de ser necesario
- Configurar la restricción MAXDELAY para lógica combinacional
- Seleccione la opción 'Check Unconstrained Paths' en la herramienta de Colocación y Ruteo (P&R)
- Restricciones de E/S del FPGA que interactúan con otros integrados

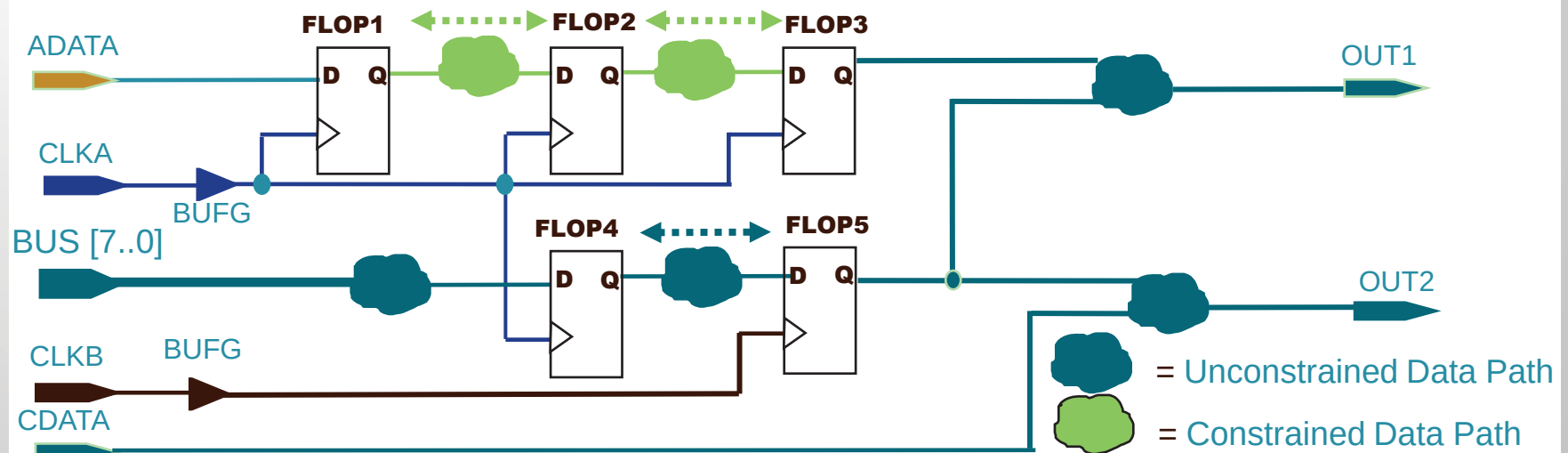


Señal del Reloj en el FPGA

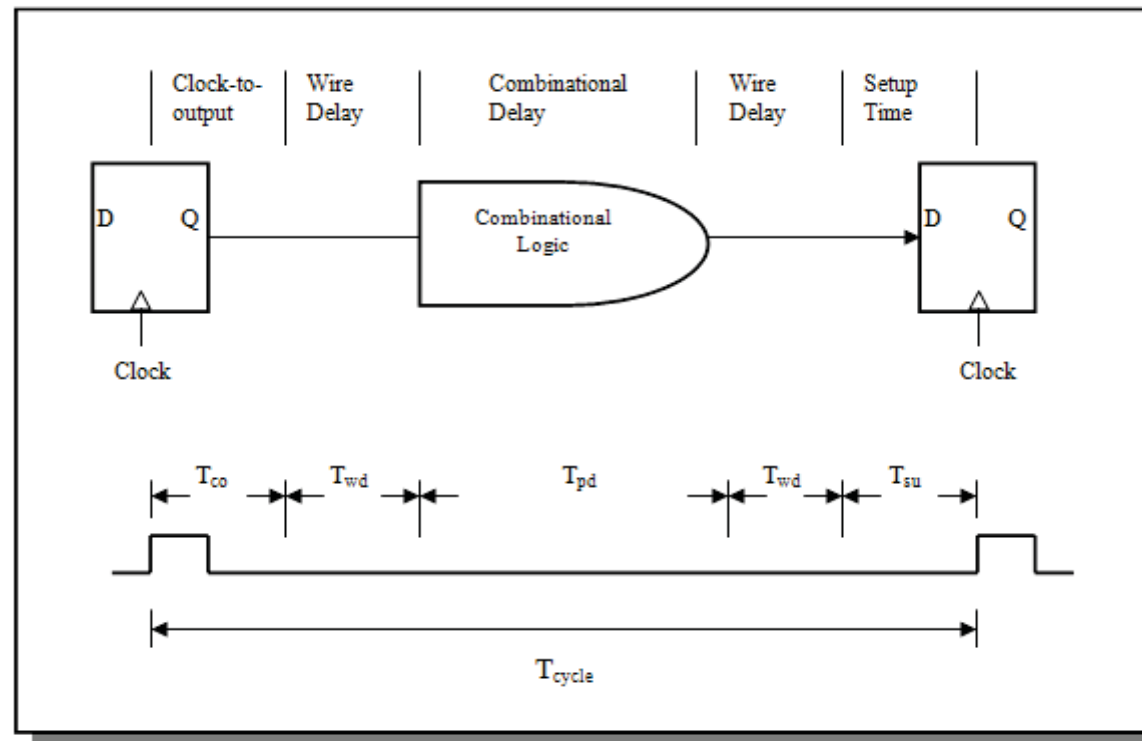
- No existe ningún reloj con retardo cero, desfasaje cero, y corrimiento cero.
- Los FPGAs tienen ruteo dedicado de bajo desfasaje y alta capacidad de señales (fan-out)
 - Opciones con VHDL/Verilog
 - Instanciar los componentes específicos del fabricante del FPGA
 - Dejar a la herramienta de síntesis que identifique las señales de alto fan-out y automáticamente infiera la lógica necesaria

Restricción de Periodo de Reloj

- La restricción de periodo de reloj:
 - SOLO cubre los caminos entre elementos sincrónicos controlados por el reloj bajo restricción
 - NO analiza:
 - Caminos directos entre entrada y salida (combinacionales)
 - Caminos entre las entradas y los registros
 - Caminos entre los registros y las salidas del FPGA
 - Caminos entre relojes no relacionados

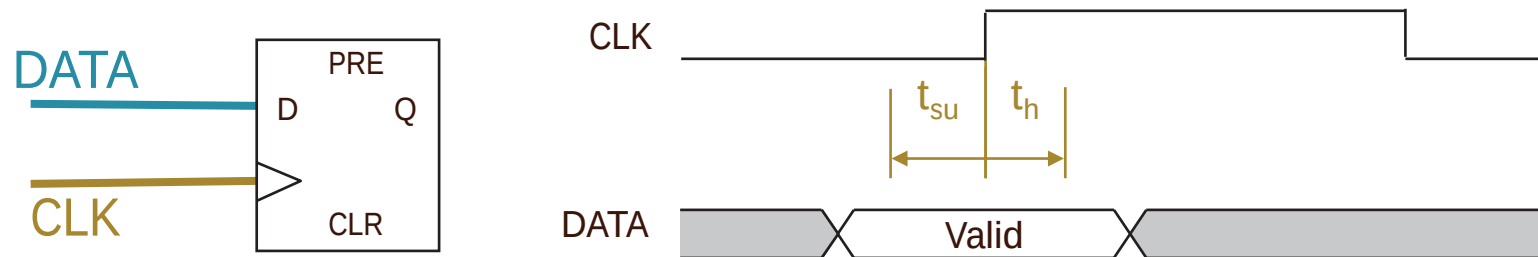


Fórmula General Periodo Mínimo



$$T_{CO} + T_{wd1} + T_{pd} + T_{wd2} + T_{SU} + T_m < T_{cycle}$$

Tiempo de Establecimiento y de Sostenimiento



Tiempo de Establecimiento

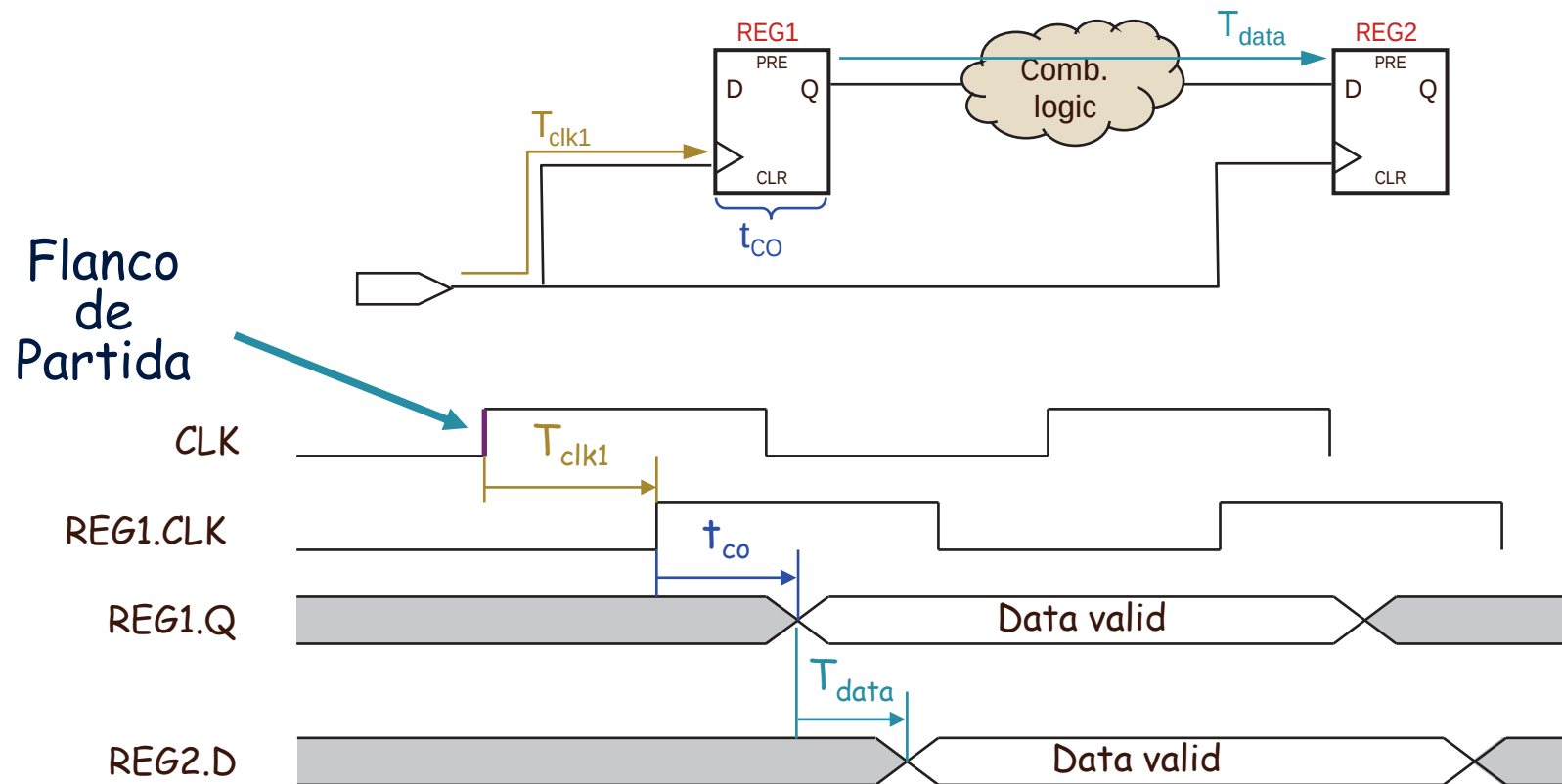
El tiempo mínimo que la señal Dato debe ser estable ***antes*** del flanco positivo del reloj

Tiempo de Mantenimiento

El tiempo mínimo que la señal de Dato debe ser estable ***después*** del flanco positivo del reloj

Tiempo de Arribo de Dato

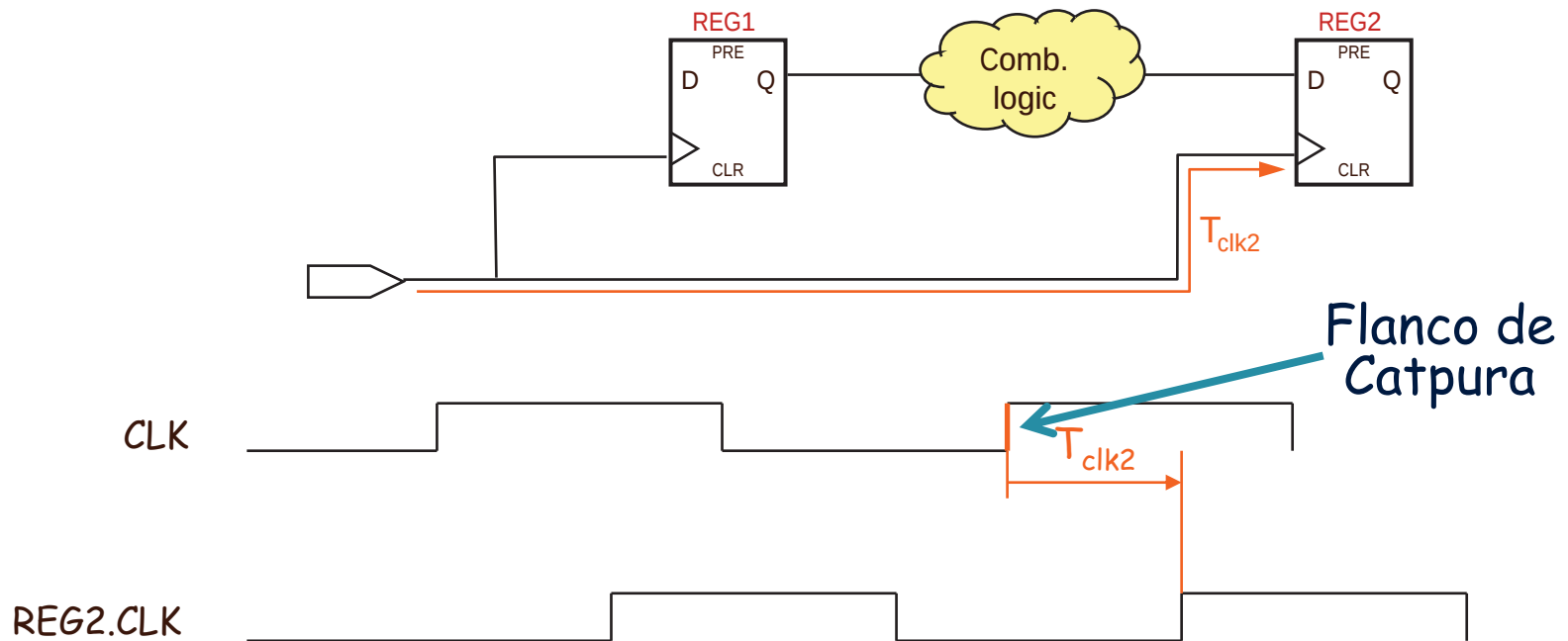
- El tiempo para que el dato llegue a la entrada D del registrro de destino



$$\text{Tiempo de Arribo de Dato} = \text{Flanco de Partida} + T_{clk1} + t_{co} + T_{data}$$

Tiempo de Arribo de Reloj

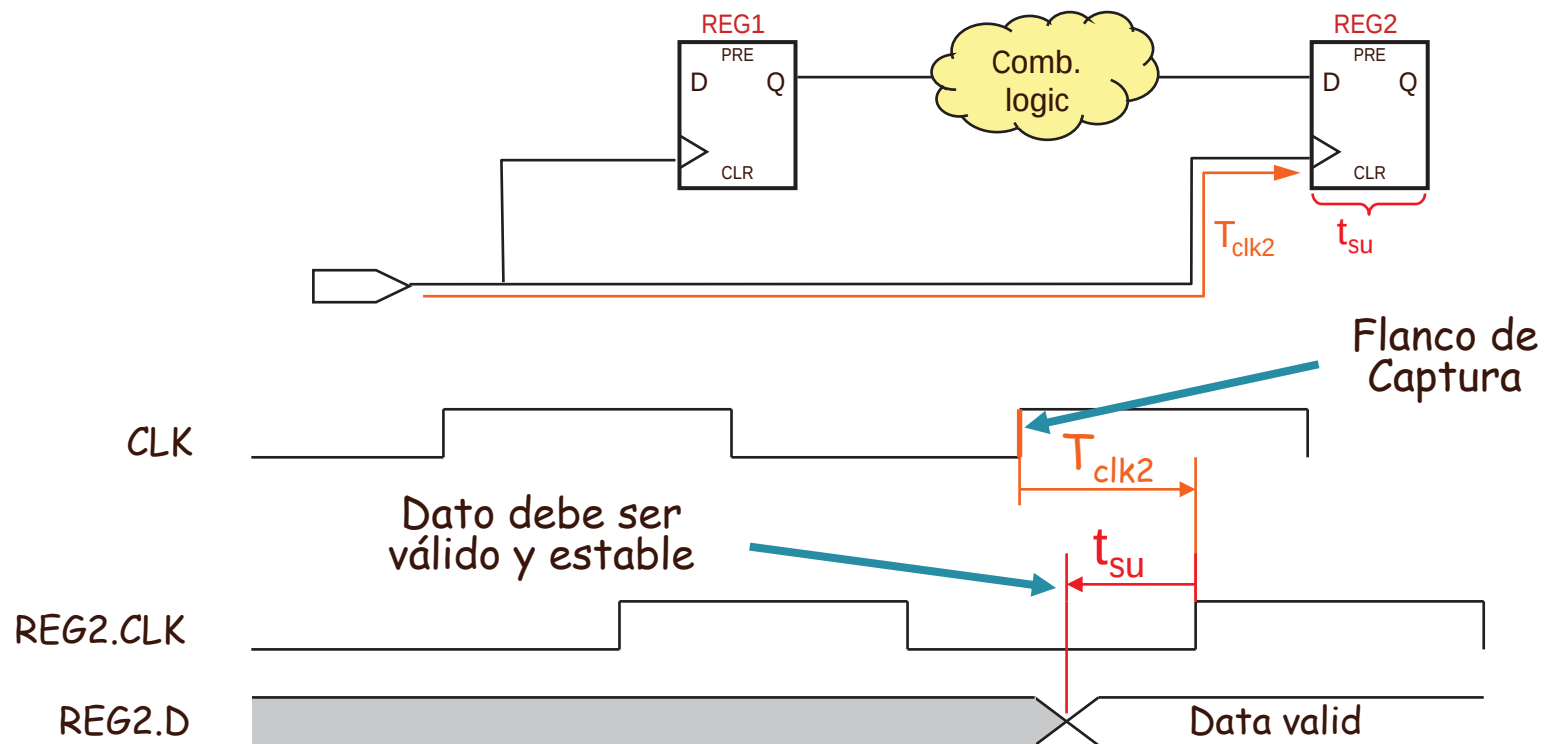
- El tiempo de reloj para arribar a la entrada de reloj del registro de destino



$$\text{Tiempo de Arribo de Reloj} = \text{Flanco de Captura} + T_{clk2}$$

Tiempo Requerido de Dato (setup)

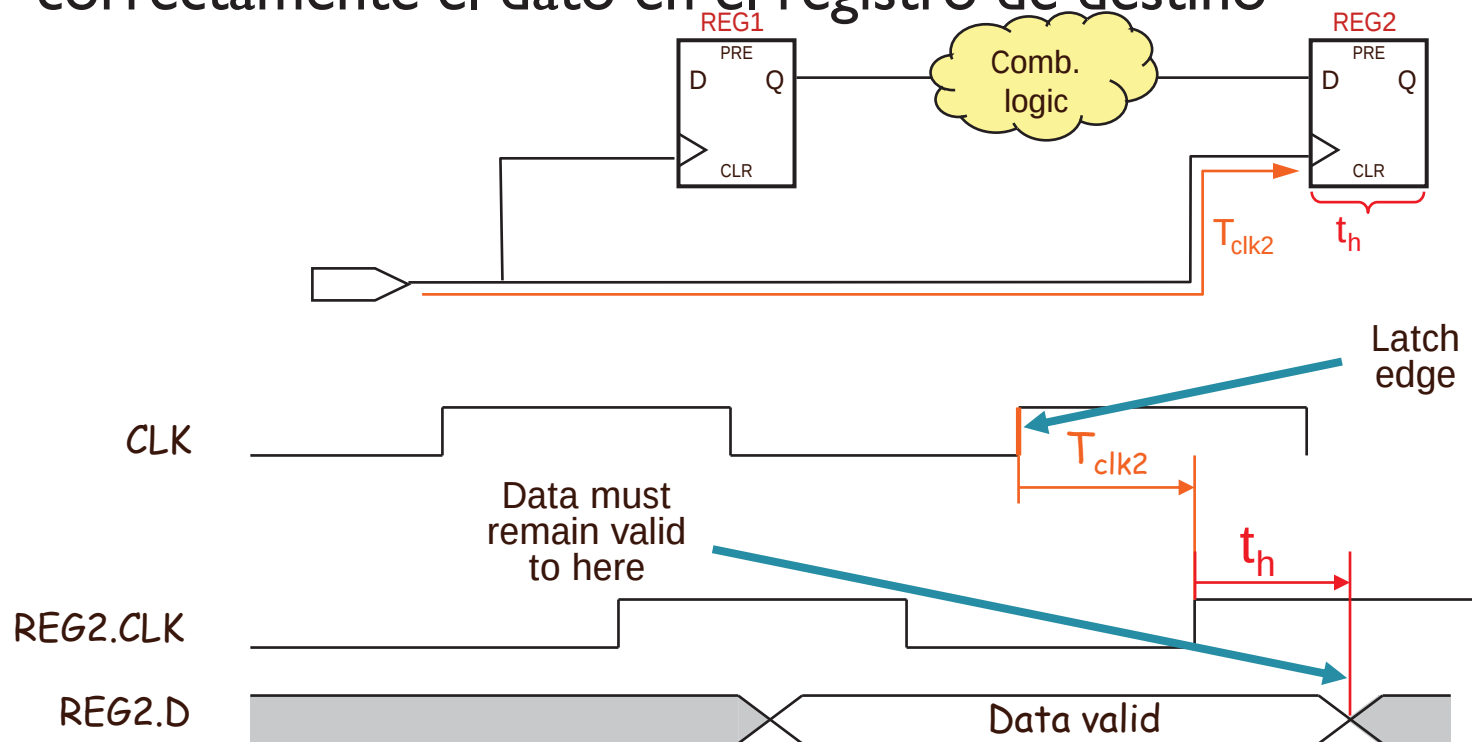
- El tiempo mínimo requerido antes del flanco de de captura del reloj para que el dato sea capturado en el registro de destino



$$\text{Tpo Requerido de Dato (Setup)} = \text{Tpo Arribo Reloj} - t_{su} - \text{Margen Setup}$$

Tiempo Requerido de Dato (Hold)

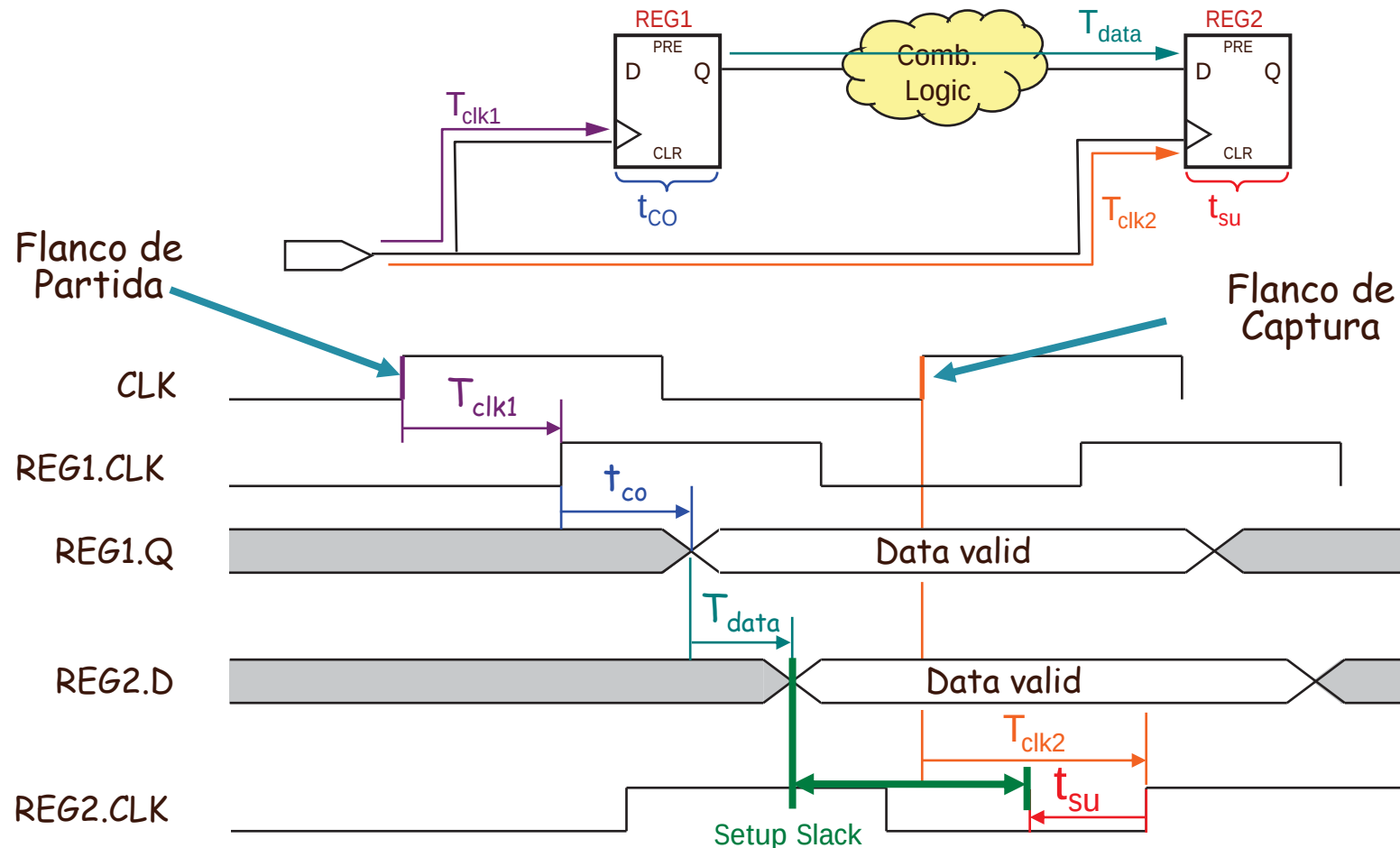
- El tiempo minimo requerido despues del flanco de reloj para que dato permanezca estable y vlaido para capturar correctamente el dato en el registro de destino



$$\text{Data Required Time (Hold)} = \text{Clock Arrival Time} + t_h + \text{Hold Uncertainty}$$

Margen de Setup (Setup Slack)

- Margen por el cual el tiempo de establecimiento es satisfecho. Asegura por ese tiempo que el dato generado por el flanco de partida arriba al registro de captura en tiempo para cumplir el tiempo de establecimiento del mismo.





Margen de Setup

*Margen de Setup = Mínimo tiempo de Dato Requerido(Setup)
- Máximo Tiempo de Arribo de Dato*

Margen Positivo

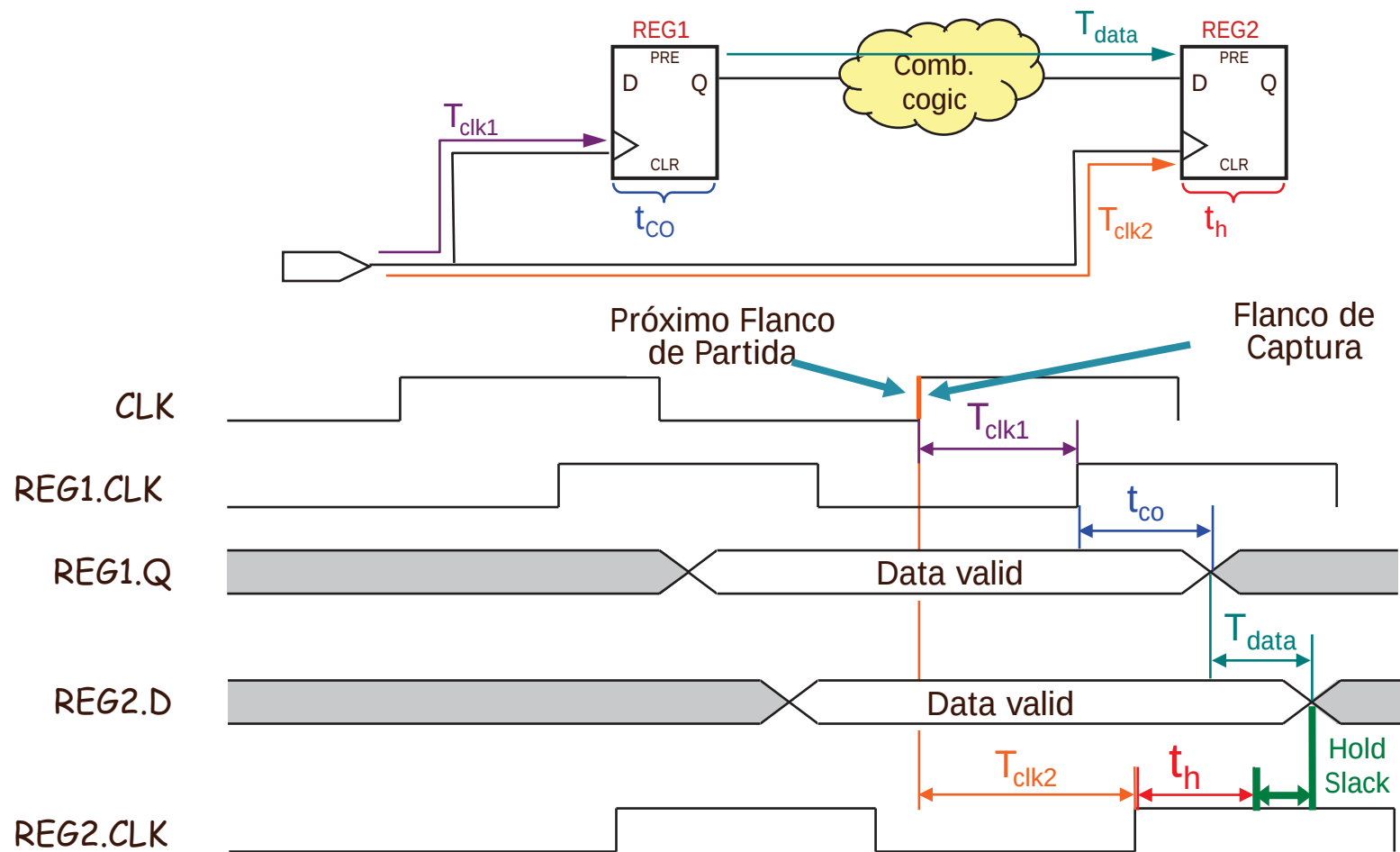
- Requerimientos de Tiempo Satisfechos

Margen Negativo

- Requerimientos de Tiempo no Satisfechos

Margen de Mantenimiento

- Margen por el cual el requerimiento de tiempo de mantenimiento es satisfecho.





Margen de Sostenimiento =
Minimo Tpo de Arribo de Dato -
Maximo Tpo de Dato Requerido (Hold)

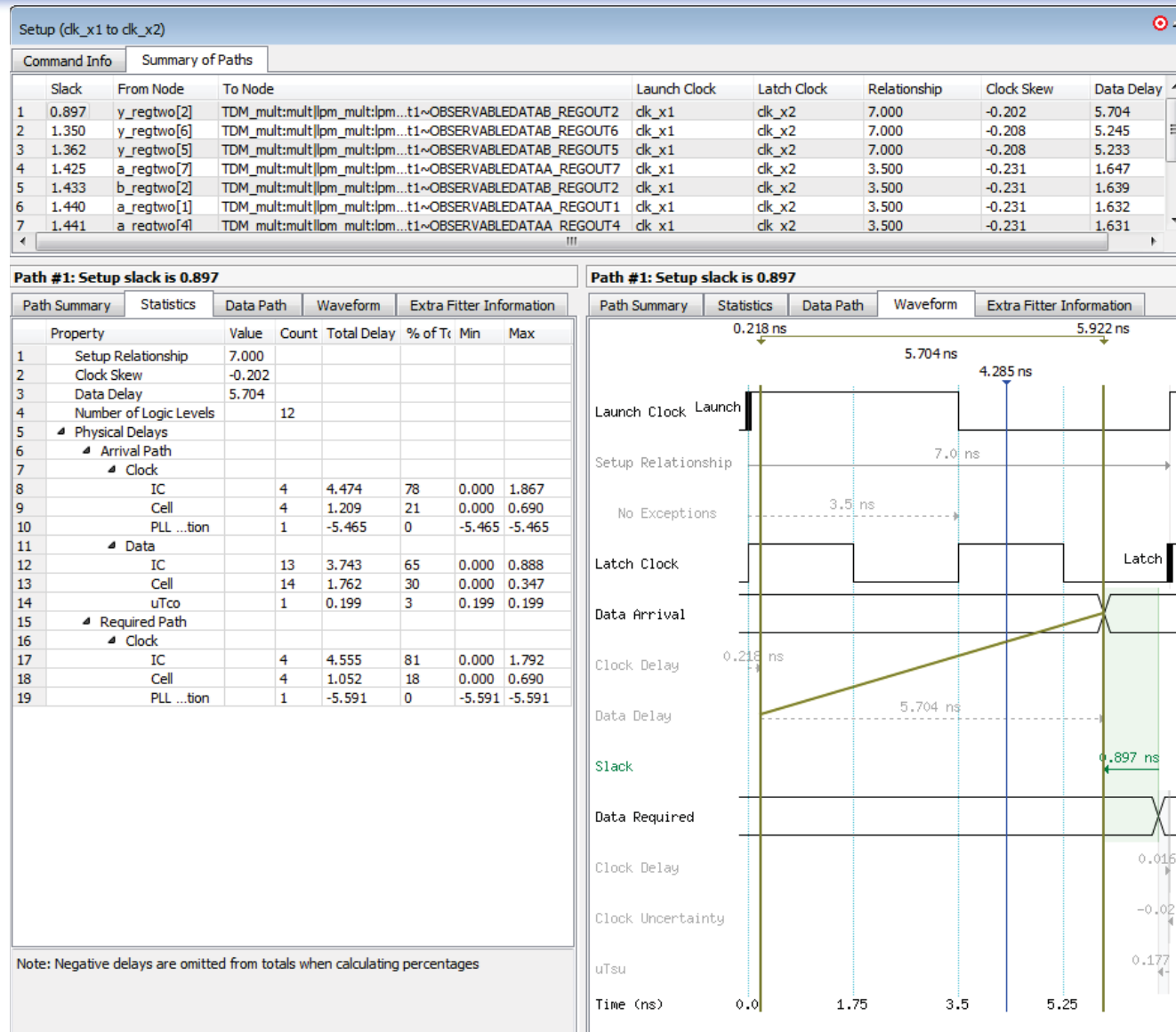
Margen Positivo

- Requerimiento de tiempo satisfechos

Margen Negativo

- Requerimiento de tiempo no satisfechos

Ejemplo de Reporte



Información de los tiempos de retardos lógicos y de ruteo

Ejemplo de Reporte (Cont.)

```
report_timing -from_clock clk_x1 -to_clock clk_x2 -setup -npaths 10 \
  -detail full_path -panel_name "Setup (clk_x1 to clk_x2)"
```

Path #1: Setup slack is 0.970

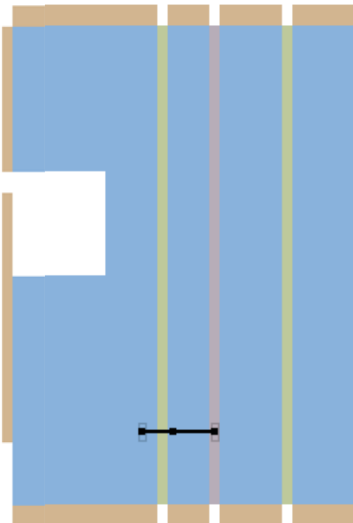
Path Summary Statistics Data Path Waveform **Extra Filter Information**

Extra information
about data path cells

Extra Filter Information [hide details]

Type	Location	Element	Partition	Bounding Box	Location Constraint Sources	Routing to Node is Constrained	Constrained Placement
1 CELL	FF_X13_Y4_N25	a_regtwo[2]q	Top	n/a	n/a	n/a	no
2 CELL	LCCOMB_X16_Y4_N2	mux_axlpm_mux_component auto_generated result_node[2]~2 combout	Top	n/a	n/a	no	no
3 CELL	DSPMULT_X20_Y4_N0	TDM_mult:multipm_mult:lpm_mult_component multipm_mult:auto_generated mac_mult1~OBSERVABLEDATA_A_REGOUT2	Top	n/a	n/a	no	no

Graphical Data Path [hide details]



The thumbnail view shows a quick visual representation of the extra filter information related to this path.

The **path connections** appear as heavy black lines. **Netlist nodes** along the path and **routing drivers** along the routing path appear as black dots. The **routing connections** appear as thin black lines. No directional information is drawn for path or routing connections.

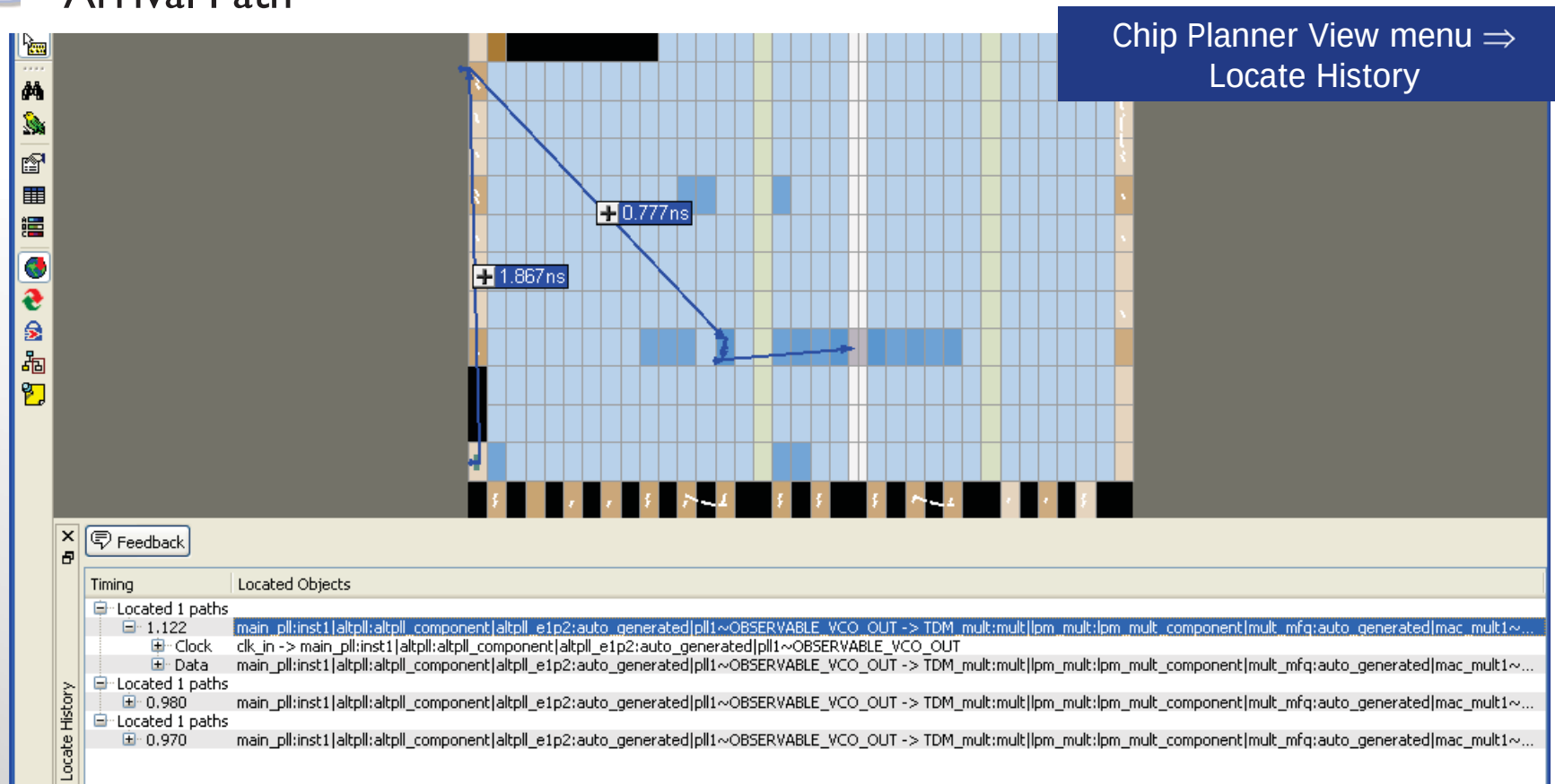
Any **bounding boxes** listed in Extra Filter Information appear as shaded blue rectangles.

Mini Chip Planner:
Visualización del camino y
colocación

Factores que pueden restringir la
ubicación de la lógica en el FPGA
(definidos por el usuario)

Localización Camino Crítico

- Keep track of paths analyzed in Chip Planner
- Quickly highlight or re-highlight whole, clock, or data portions of Data Arrival Path



Reporte de Periodo Mínimo

Slack: 18.108ns (requirement - (data path - clock skew))
Source: [ff 0](#) (FF)
Destination: [ff 0](#) (FF)
Requirement: 20.000ns
Data Path Delay: 1.892ns (Levels of Logic = 1)
Clock Skew: 0.000ns
Source Clock: clk20g rising at 0.000ns
Destination Clock: clk20g rising at 20.000ns

Data Path: [ff 0](#) to [ff 0](#)

Delay type	Delay(ns)	Logical Resource(s)
Tcko	0.568	ff 0
net (fanout=7)	0.514	ff 0
Tilo	0.439	N 147 i
net (fanout=1)	0.001	N 147 i
Tdxck	0.370	ff 0
Total	1.892ns	(1.377ns logic, 0.515ns route) (72.8% logic, 27.2% route)

Ecuación de Margen de Setup

Registros: partida/destino

Niveles de Lógica

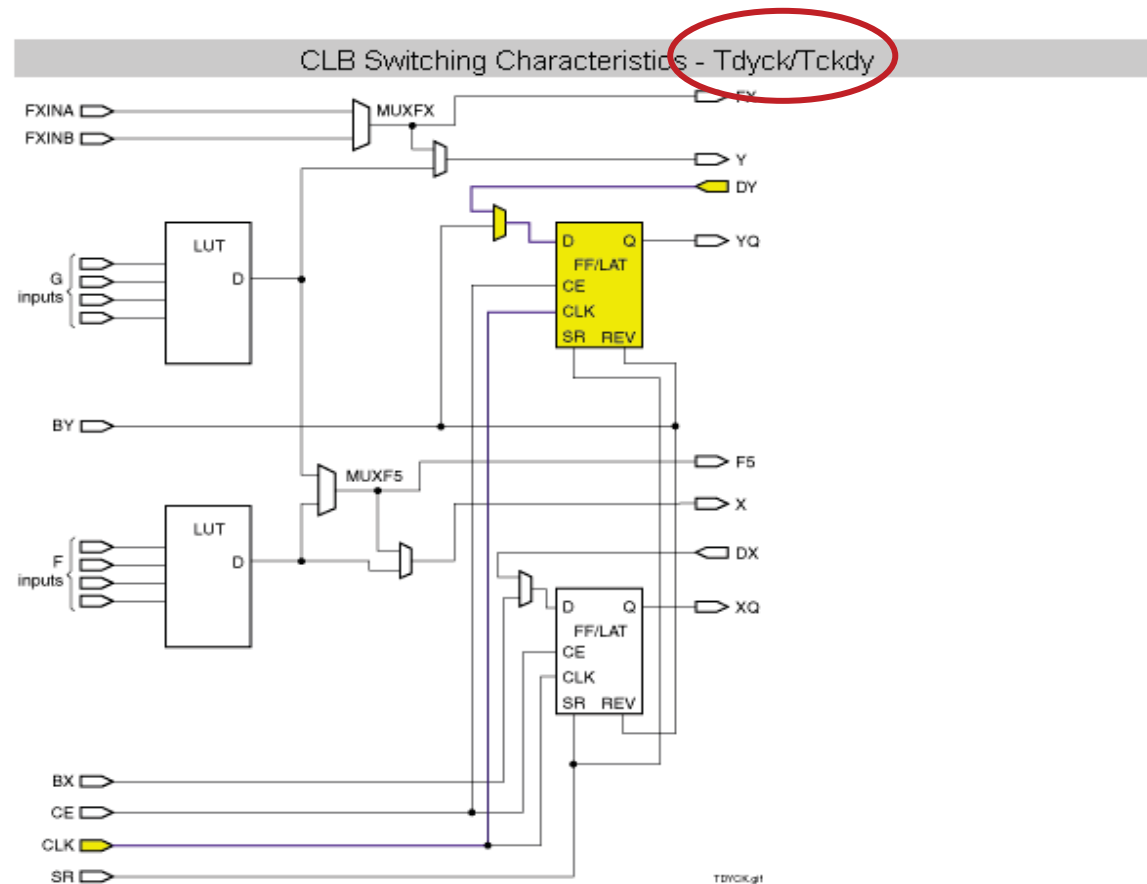
Nombre del Reloj y flanco activo del mismo

Camino del Dato con retardo lógico y de ruteo. Link a herramientas dedicadas

Link al circuito mencionado en retardo

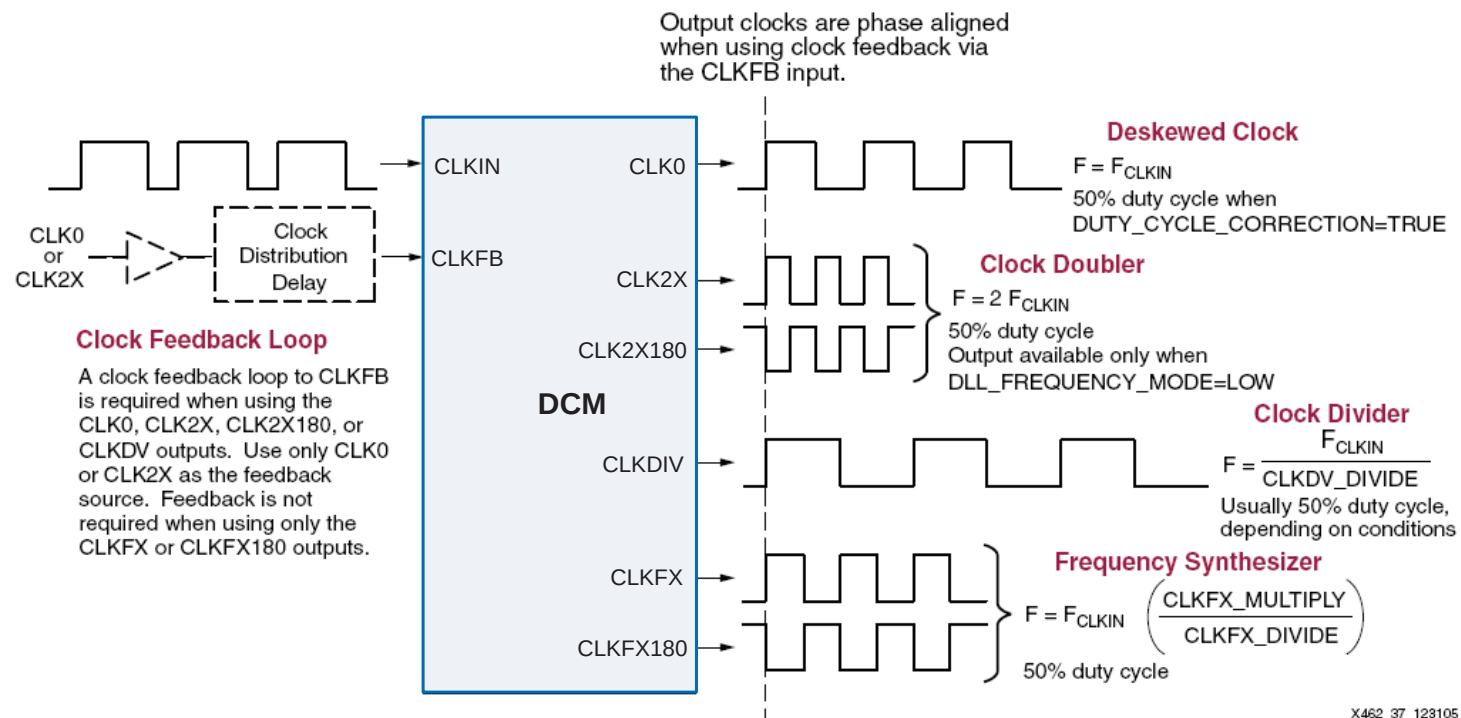
Ejemplo de Información de Componente

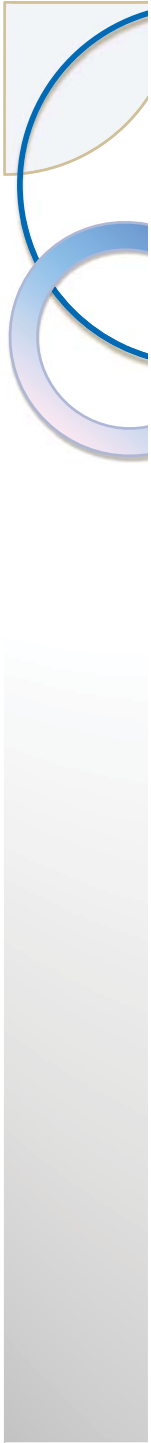
Show Contents



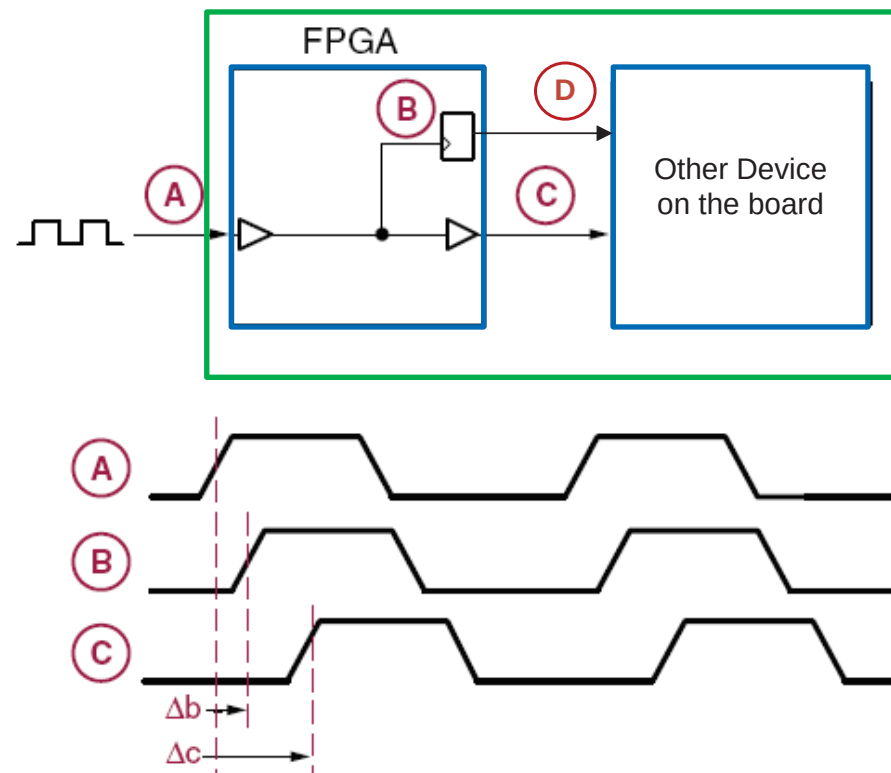
Description: Setup and hold times for DY inputs, relative to clock CLK

DLL: Ejemplo

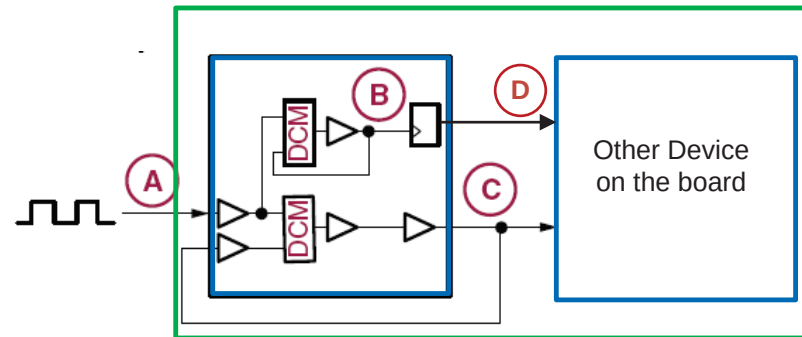




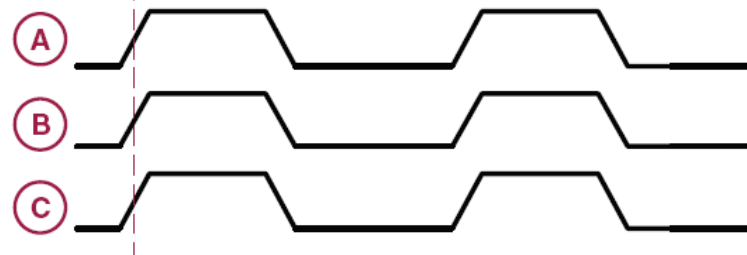
Ejemplo: Eliminación Sesgo de Reloj



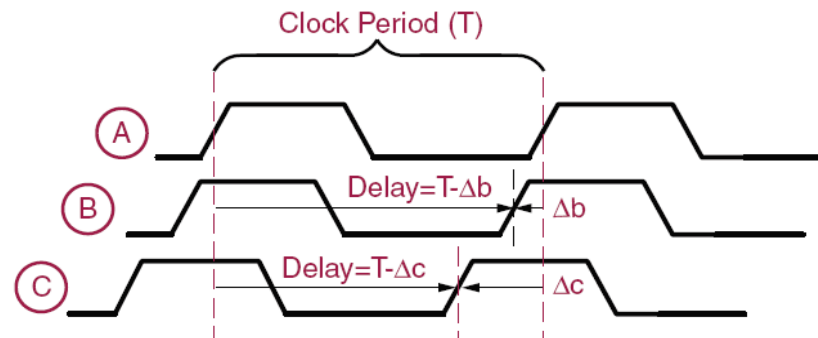
PLL/DLL: Ejemplo de Uso I



A) Esquema de configuración de los DCMs

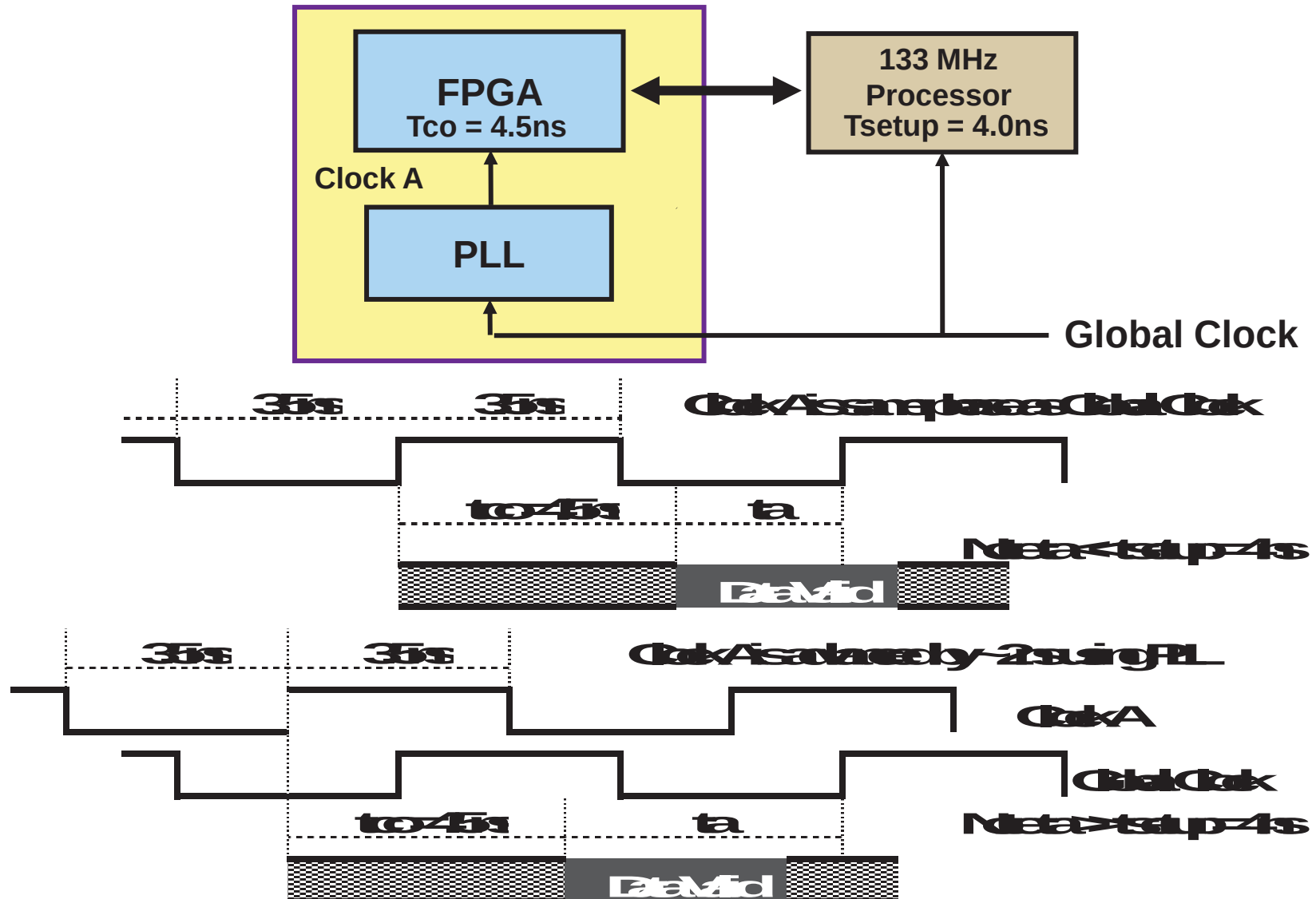


B) Alineamiento ideal del reloj



C) Retardando el reloj, parece un adelantamiento

PLL/DLL: Ejemplo de Uso 2





SOLUCIONES PARA PROBLEMAS DE TIMING



Resolución de Fallas de Tiempo

Análisis del fallas de tiempo mas comunes y sus posibles soluciones.

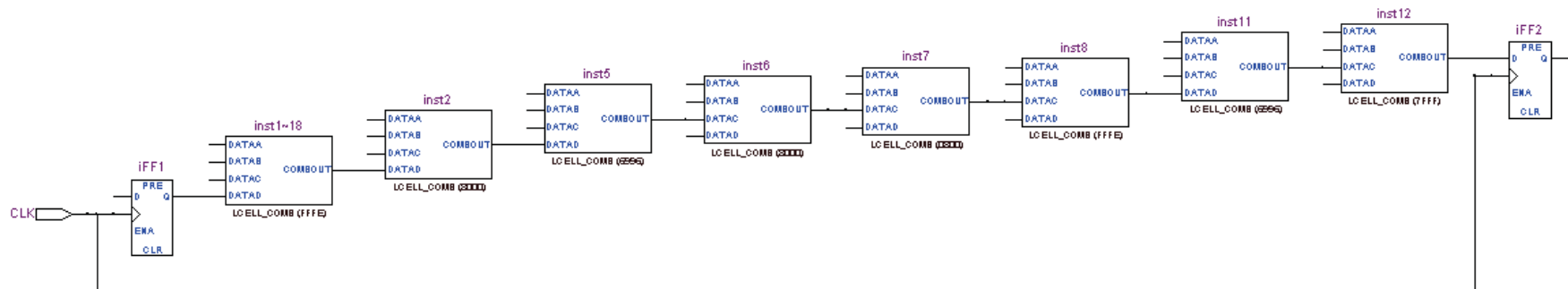
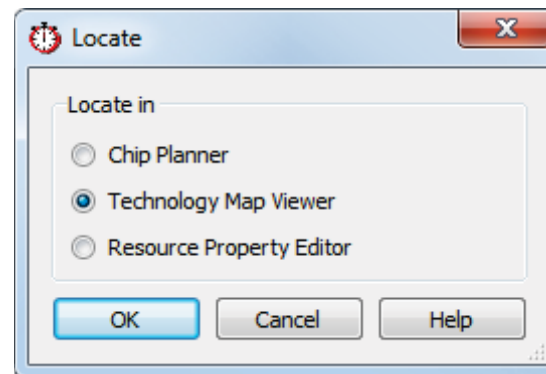
1. Demasiados Niveles Lógicos
2. Alto fan-out de señales
3. Conflicto de requerimientos de tiempo
4. Requerimientos de Tiempo demasiados exigentes



Caso I – Demasiados Niveles Lógicos

- Incrementa el retardo de T_{data} , por lo que incrementa el tiempo de arribo
- Verificación
 - Ver el esquemático correspondiente
 - Analisis con la herramienta STA correspondiente (TimeQuest o Timing Analyzer)

Caso I – Cont.



El camino de Data tiene 8 niveles Lógicos

Caso I – Time Quest Análisis

Report Timing (Worst-Case Path)

Command Info Summary of Paths

	Slack	From Node	To Node	Launch Clock	Latch Clock	Relationship	Clock Skew	Data Delay
1	-2.466	PACKET_CHECK:IPACKET_CHECK[last_data[1]	PACKET_CHECK:IPACKET_CHECK[parity_error	SCLK	SCLK	5.000	-0.070	7.394

Path #1: Setup slack is -2.466 (VIOLATED)

Path Summary Statistics Data Path Waveform Extra Fitter Information

Property	Value	Count	Total Delay
1 Setup Relationship	5.000		
2 Clock Skew	-0.070		
3 Data Delay	7.394		
4 Number of Logic Levels		17	
5 Physical Delays			
6 Arrival Path			
7 Clock			
8 Clock Network (Lumped)		1	2.651
9 Data			
10 IC		18	4.571
11 Cell		19	2.591
12 uTco		1	0.232
13 Required Path			
14 Clock			
15 Clock Network (Lumped)		1	2.558

Note: Negative delays are omitted from totals when calculating percentages

Path #1: Setup slack is -2.466 (VIOLATED)

Path Summary Statistics Data Path Waveform Extra Fitter Information

Data Arrival Path

	Total	Incr	RF	Type	Fanout	Location	Element
1	0.000	0.000					launch edge time
2	2.651	2.651					clock path
3	2.651	2.651	R				clock network delay
4	10.045	7.394					data path
5	2.883	0.232		uTco	1	FF_X11_Y13_N3	PACKET_CHECK
6	2.883	0.000	FF	CELL	1	FF_X11_Y13_N3	IPACKET_CHECK
7	3.209	0.326	FF	IC	1	LCCOMB_X11_Y13_N16	IPACKET_CHECK
8	3.359	0.150	FR	CELL	1	LCCOMB_X11_Y13_N16	IPACKET_CHECK
9	3.563	0.204	RR	IC	1	LCCOMB_X11_Y13_N6	IPACKET_CHECK
10	3.718	0.155	RR	CELL	1	LCCOMB_X11_Y13_N6	IPACKET_CHECK
11	4.095	0.377	RR	IC	1	LCCOMB_X10_Y13_N30	IPACKET_CHECK
12	4.250	0.155	RR	CELL	1	LCCOMB_X10_Y13_N30	IPACKET_CHECK
13	4.619	0.369	RR	IC	1	LCCOMB_X10_Y13_N18	IPACKET_CHECK
14	4.758	0.139	RF	CELL	1	LCCOMB_X10_Y13_N18	IPACKET_CHECK
15	4.985	0.227	FF	IC	1	LCCOMB_X10_Y13_N16	IPACKET_CHECK
16	5.135	0.150	FR	CELL	1	LCCOMB_X10_Y13_N16	IPACKET_CHECK
17	5.339	0.204	RR	IC	1	LCCOMB_X10_Y13_N26	IPACKET_CHECK
18	5.494	0.155	RR	CELL	1	LCCOMB_X10_Y13_N26	IPACKET_CHECK
19	5.866	0.372	RR	IC	1	LCCOMB_X10_Y13_N24	IPACKET_CHECK

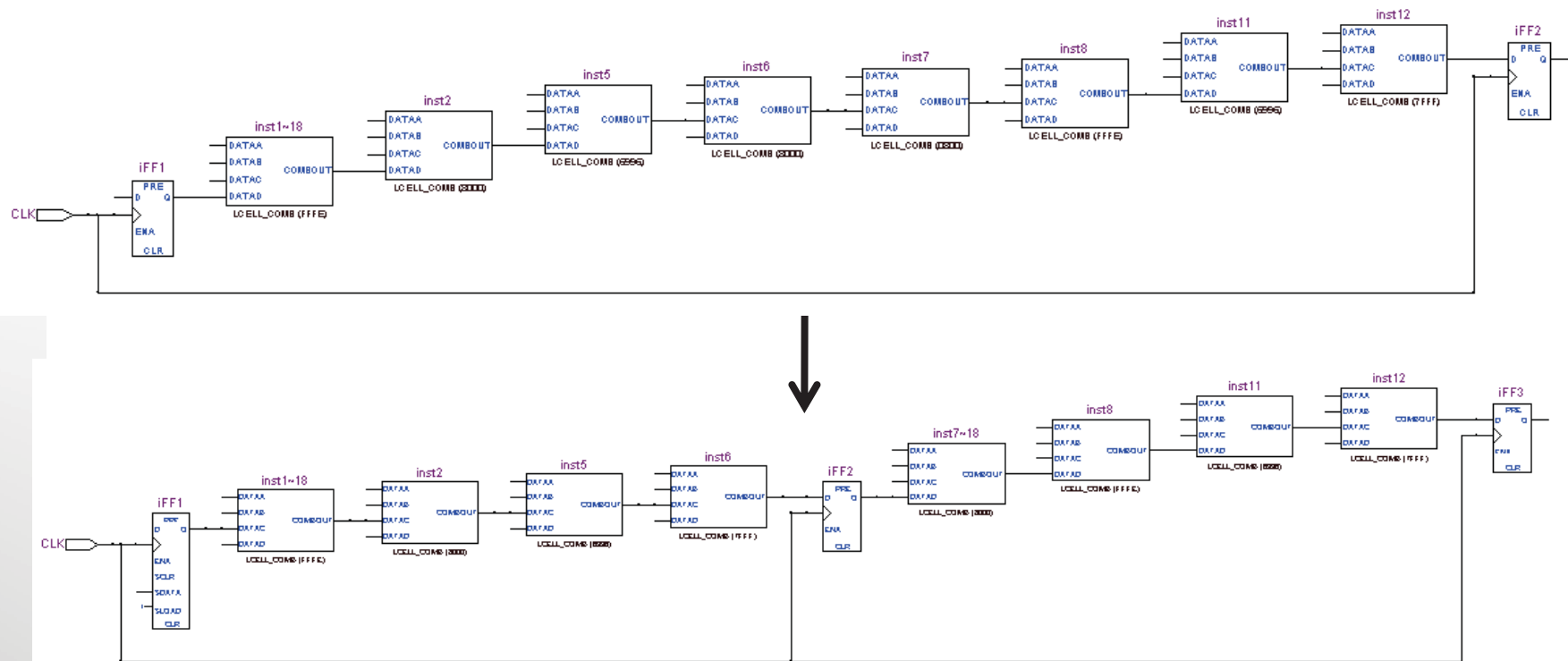
Data Required Path

	Total	Incr	RF	Type	Fanout	Location	Element
1	5.000	5.000					latch edge time
2	7.581	2.581					clock path
3	7.558	2.558	R				clock network delay

Número de Niveles Lógicos en el camino de arribo

Caso I–Solución Agregando Registros

- Add pipeline registers to reduce T_{data}



Caso I – Consejo I

- No use *IF* anidados. Use *CASE*

VHDL

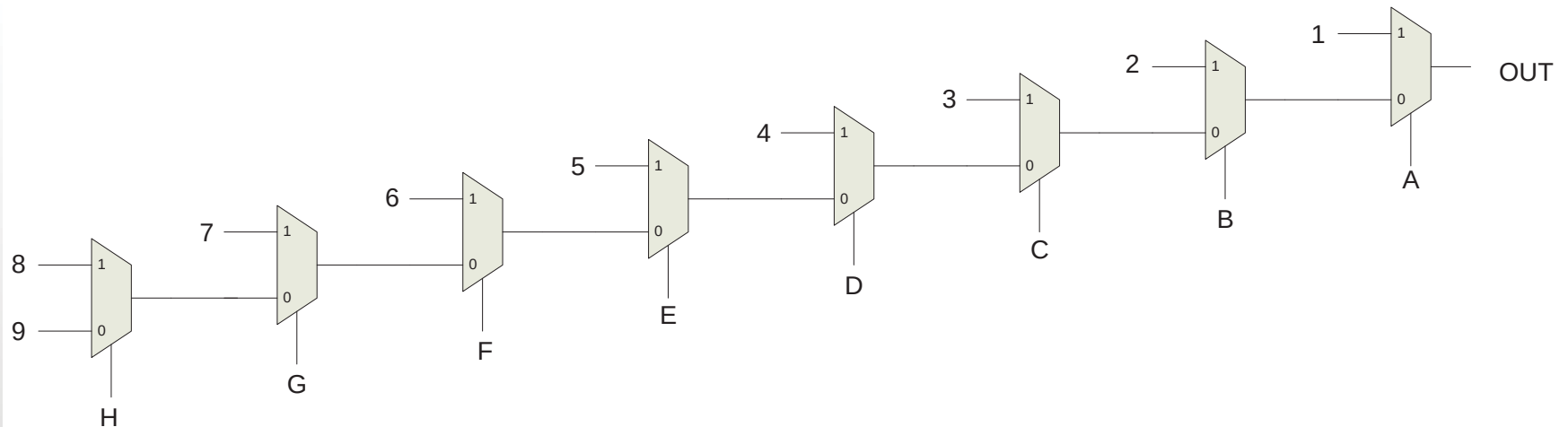
```
-- Too many embedded IF statements
process(A, B, C, D, E, F, G, H)
begin
    if A = '1' then
        sig_out <= 1;
    elsif B = '1' then
        sig_out <= 2;
    elsif C = '1' then
        sig_out <= 3;
    elsif D = '1' then
        sig_out <= 4;
    elsif E = '1' then
        sig_out <= 5;
    elsif F = '1' then
        sig_out <= 6;
    elsif G = '1' then
        sig_out <= 7;
    elsif H = '1' then
        sig_out <= 8;
    else
        sig_out <= 9;
    end if;
end process;
```

Verilog

```
// Too many embedded IF statements
always @(*)
begin
    if (A)
        sig_out <= 1;
    else if (B)
        sig_out <= 2;
    else if (C)
        sig_out <= 3;
    else if (D)
        sig_out <= 4;
    else if (E)
        sig_out <= 5;
    else if (F)
        sig_out <= 6;
    else if (G)
        sig_out <= 7;
    else if (H)
        sig_out <= 8;
    else
        sig_out <= 9;
end
```


Caso I – Solucion I.a

- Resulting hardware interpretation



Re-Temporizado (retiming)

- Mueva registros a través de lógica combinacional para balancear el timing
- Intercambia caminos críticos con no-críticos
- No cambia la funcionalidad de la lógica



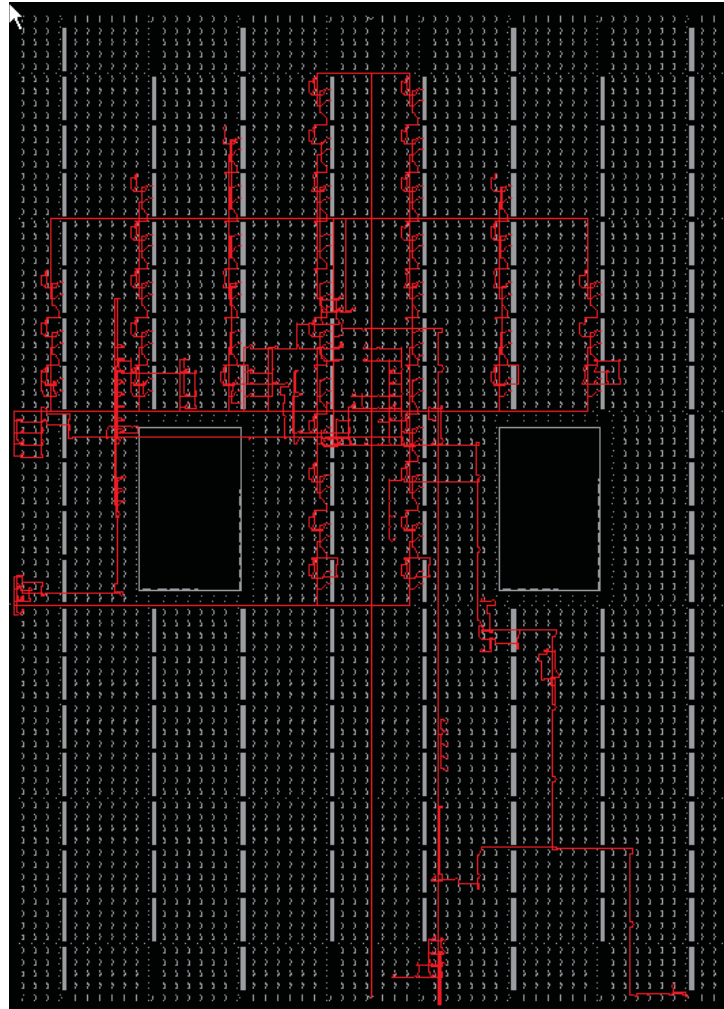
Señales de Alto FAN-OUT

The following nets have been assigned to a chip global resource:

Fanout	Type	Name
4378	INT_NET	Net : SysClk Driver: Instance_Clk_M1Rst_Gen/P11_25Mhz_0/Core Source: ESSENTIAL
2277	INT_NET	Net : SysRstN_c_c Driver: Instance_SG_FEDK/CortexM1Top_0/CortexM1Top_II/U_CLKSRC Source: NETLIST
1861	SET/RESET_NET	Net : Instance_SG_FEDK/CMACIO1III_RNI964 Driver: Instance_SG_FEDK/CORE10100_AHBAPB_0/CMAC00010I/CMACII11I/CMACIO1III_RNI964/U_CLKSRC Source: NETLIST
776	INT_NET	Net : Instance_SG_FEDK/CortexM1Top_0/DBGRESETn Driver: Instance_SG_FEDK/CortexM1Top_0/genblk1085.genblk1087.CortexM1Top_01/U_CLKSRC Source: NETLIST
322	CLK_NET	Net : Instance_SG_FEDK/CLKINT_1_Y Driver: Instance_SG_FEDK/CLKINT_1/U_CLKSRC Source: NETLIST
273	CLK_NET	Net : Instance_SG_FEDK/CLKINT_0_Y Driver: Instance_SG_FEDK/CLKINT_0/U_CLKSRC Source: NETLIST

1						
	-----+-----+-----+-----+-----+-----+					
	Clock Net Resource Locked Fanout Net Skew(ns) Max Delay(ns)					
	-----+-----+-----+-----+-----+-----+					
	fpga_clk_250 BUFGMUX3S No 126 0.280 1.410					
	-----+-----+-----+-----+-----+-----+					
	baud_gen/rs232_clk_i BUFGMUX7S No 72 0.307 1.410					
	-----+-----+-----+-----+-----+-----+					
	sys_clk_100_BUFGP BUFGMUXOP No 34 0.210 1.322					
	-----+-----+-----+-----+-----+-----+					

Señales de Alto FAN-OUT – P&R





Caso 2 – Cont.

- Encuentre las señales de fan-out como posibles causas de:
 - Use TimeQuest/Timing Analysis para el análisis de caminos críticos
- Localice la senales con alto fan-out signals as possible causes
- Use Chip Planner/FPGA Editor para verificar la locación de la lógica/ruteo

Caso 2 – Análisis Camino Crítico

Report Timing (Worst-Case Path)								
Command Info		Summary of Paths						
	Slack	From Node	To Node	Launch Clock	Latch Clock	Relationship	Clock Skew	Data Delay
1	-1.716	sig	my_regv[1095]	clk	clk	5.000	4.083	9.667

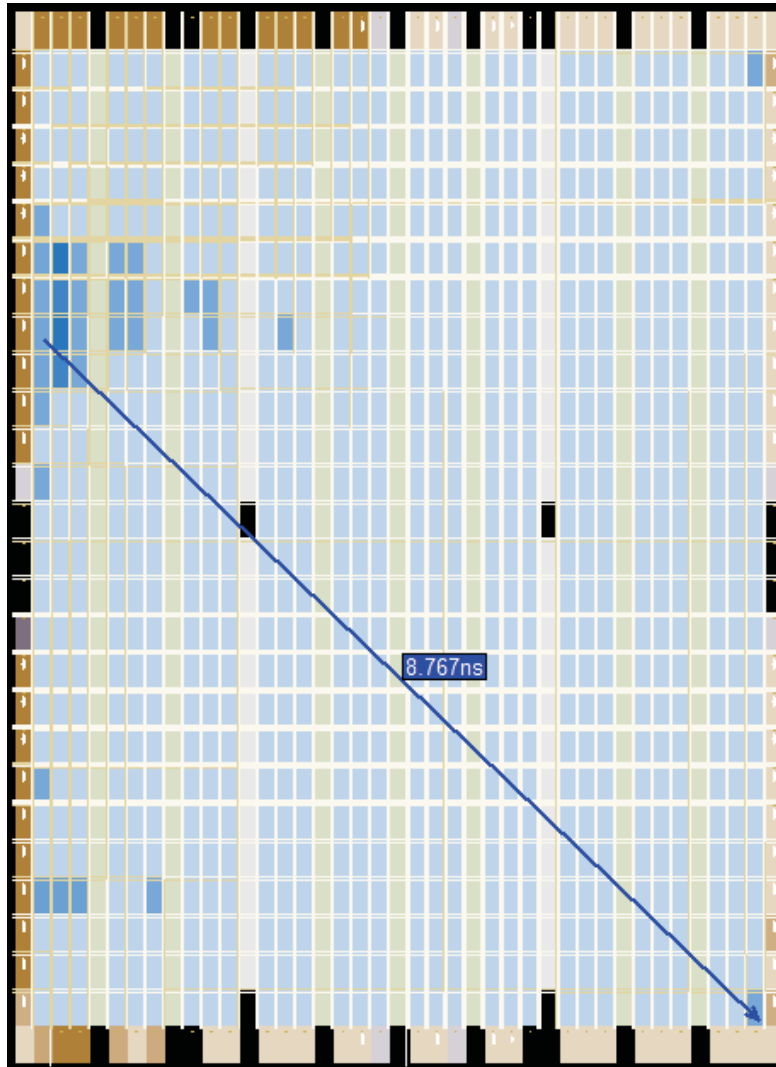
Path #1: Setup slack is -1.716 (VIOLATED)

Path Summary		Statistics	Data Path	Waveform	Extra Fitter Information			
Data Arrival Path								
	Total	Incr	RF	Type	Fanout	Location	Element	
1	0.000	0.000					launch edge time	
2	0.000	0.000					clock path	
3	0.000	0.000	R				clock network delay	
4	1.000	1.000	R	iExt	1	PIN_T34	sig	
5	10.667	9.667					data path	
6	1.000	0.000	RR	IC	1	IOIBUF_X0_Y55_N32	sig~input i	
7	1.782	0.782	RR	CELL	4001	IOIBUF_X0_Y55_N32	sig~input o	
8	9.945	8.163	RR	IC	1	MLABCELL_X92_Y47_N38	my_regv[1095]~feeder datab	
9	10.539	0.594	RR	CELL	1	MLABCELL_X92_Y47_N38	my_regv[1095]~feeder combout	
10	10.539	0.000	RR	IC	1	FF_X92_Y47_N39	my_regv[1095] d	
11	10.667	0.128	RR	CELL	1	FF_X92_Y47_N39	my_regv[1095]	

Fanout de 4001
retardo de ruteo
de 8.163 ns

Data Required Path								
	Total	Incr	RF	Type	Fanout	Location	Element	
1	5.000	5.000					latch edge time	
2	9.083	4.083					clock path	
4	8.951	-0.132		uTsu	1	FF_X92_Y47_N39	my_regv[1095]	

Caso 3 - Floorplanning



With setup issues, flops are usually too far apart

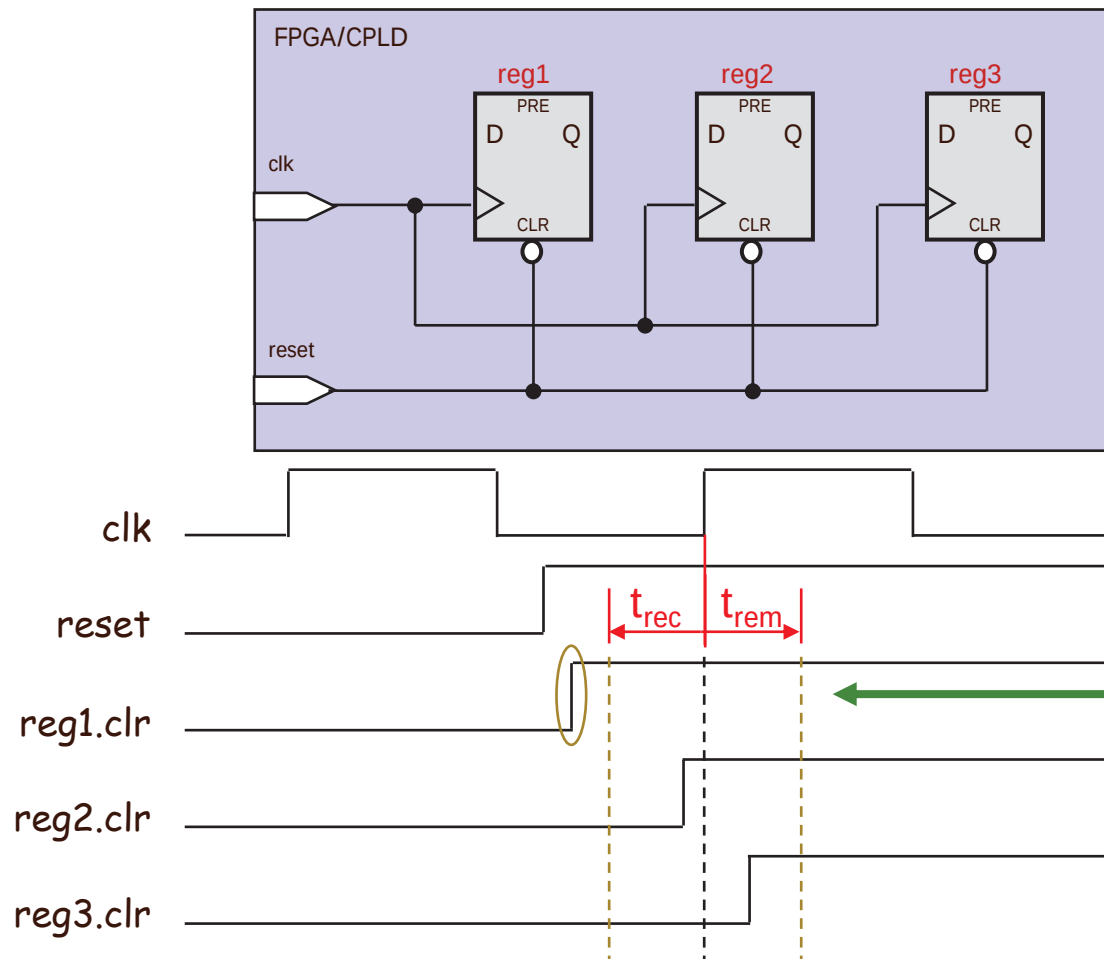
Porqué la lógica está colocada tan separada?



RESET SINCRÓNICO O ASINCRÓNICO

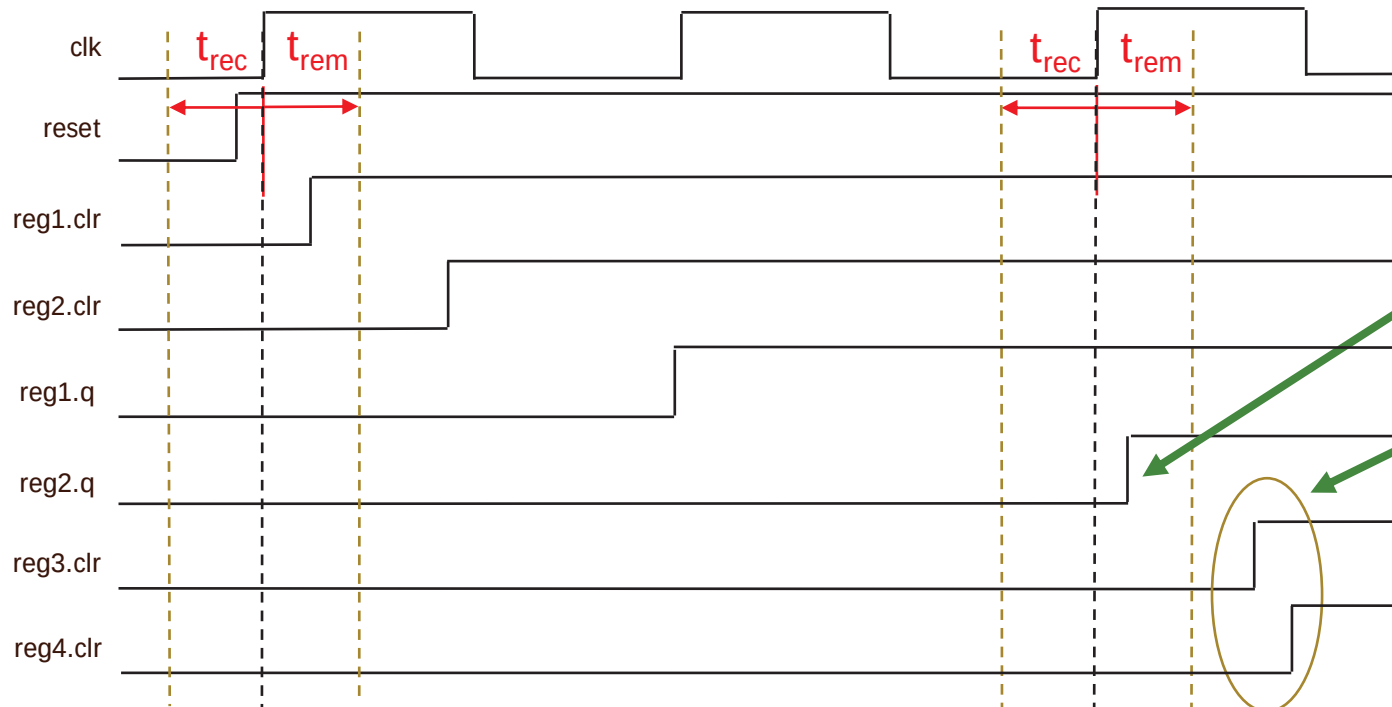
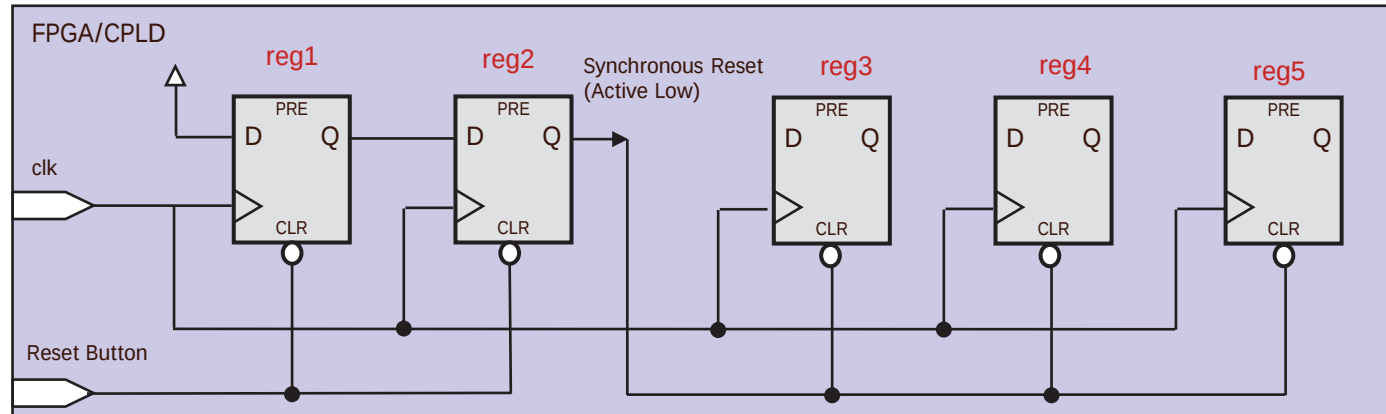
Ejemplo Mal Uso Reset Asincrónico

- For example, these state machine registers should all be de-asserted together, but...



Debido a retardo de ruteo de la señal de reset, SOLO *reg1* sale de reset correctamente. Los otros registros son reseteados en el otro ciclo de reloj. Esto puede hacer que se comience en un estado erróneo o incluso un estado ilegal.

Reset Asincronico – Uso Correcto



La señal de reset es activada asincrónicamente y desactivada sincrónicamente. Evitando violar los tiempos de recuperación y remoción del FF.



ENFOQUES PARA UN MÁXIMO RENDIMIENTO



Estrategias para FMax

- Establezca correctamente las restricciones de tiempo
 - Ni demasiado estrictas, ni demasiado sueltas
- Ejecute el Ruteo y Colocación la primera vez con sus valores por defecto como primer paso para obtener el mejor rendimiento
- Analice cuidadosamente el reporte de la herramienta de análisis de tiempo estático (60/40)
- Ajuste las restricciones de tiempo adecuadamente
- Ajuste los distintos parámetros de la herramienta de Ruteo y Colocación
- Floorplan su diseño
 - Use grupos/regiones
 - Use restricciones de colocación



Reglas para Diseños Sincrónicos

- Use uno y solo un flanco activo de reloj (no mezcle los flancos!)
- Particione jerárquicamente los módulos por funciones
- Registre las salidas de cada modulo
- Trate de usar habilitación de reloj (clock enable) en lugar de tener muchos dominios de reloj
- Siempre sincronice señales asincrónicas
- No use:
 - Reloj generados por compuertas lógicas
 - Reloj divididos (cuidado con los retardos)
 - Set/reset generados localmente



Regla del 60/40

Identifique cuanto tiempo del camino más lento corresponde a retardo lógico y cuanto tiempo corresponde a retardo de ruteo.

Lo ideal es la siguiente relación:

60% Retardo Logico
40% Retardo de Ruteo



Periodo de Reloj



Regla del 60/40

Use la herramienta de STA apropiada para analizar el camino de mayor retardo. Analice cual es el porcentaje del ciclo de reloj usado por el retardo lógico. Si ese valor esta en:

- **Debajo del 60%:** Seguramente el diseño va satisfacer los requerimientos de tiempo del diseño impuestos.
- **60-80%:** El diseño puede llegar a cumplir los requerimientos impuestos usando configuración avanzada de la herramienta de P&R
- **Sobre el 80%:** -> **no hay chances** ! Re-codifique el diseño o utilice opciones avanzadas en la herramienta de síntesis

Regla del 60/40

```
=====
Timing constraint: TS_fpga_clk_250 = PERIOD TIMEGRP "fpga_clk_250" 4 ns HIGH 50%;
```

```
1412 items analyzed, 0 timing errors detected. (0 setup errors, 0 hold errors)
Minimum period is 3.998ns.
```

```
-----
Slack: 0.002ns (requirement - (data path - clock path skew + uncertainty))
Source: inst ddr rx/abus ddr re sync i 6 (FF)
Destination: inst ddr rx/abus aligned0 6 (FF)
Requirement: 4.000ns
Data Path Delay: 3.955ns (Levels of Logic = 0)
Clock Path Skew: -0.043ns
Source Clock: fpga_clk_250 rising at 0.000ns
Destination Clock: fpga_clk_250 rising at 4.000ns
Clock Uncertainty: 0.000ns
```

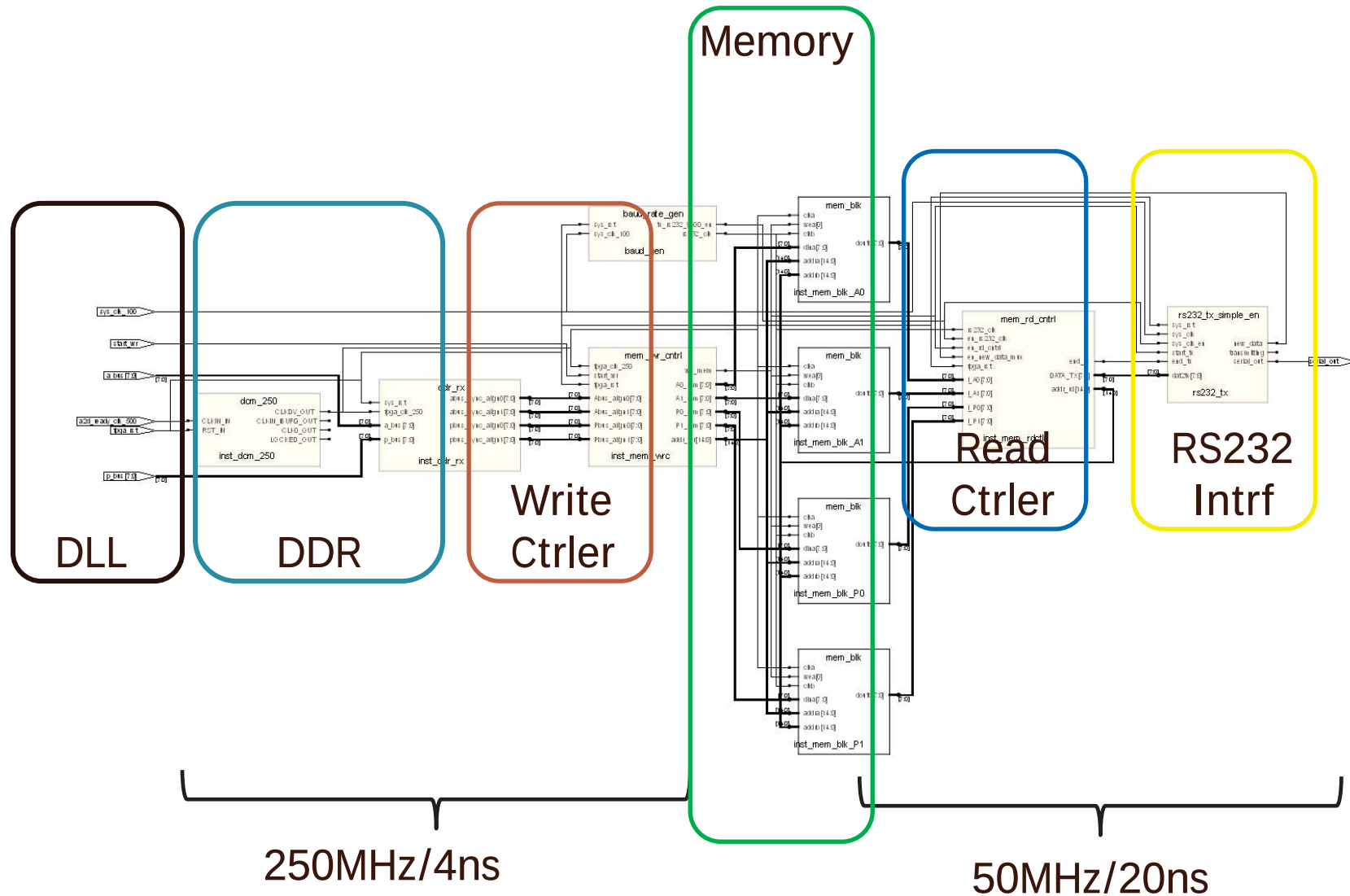
```
-----
Data Path Slack: 0.036ns (requirement - (data path - clock path skew + uncertainty))
Data Path Source: inst mem wrb/cl/BU2/U0/q i 11 (FF)
Data Path Destination: inst mem wrb/cl/BU2/U0/q i 12 (FF)
Data Path Requirement: 4.000ns
Data Path Data Path Delay: 3.964ns (Levels of Logic = 3)
Data Path Clock Path Skew: 0.000ns
Data Path Source Clock: fpga_clk_250 rising at 0.000ns
Data Path Destination Clock: fpga_clk_250 rising at 4.000ns
Data Path Clock Uncertainty: 0.000ns
```

[Data Path: inst mem wrb/cl/BU2/U0/q i 11 to inst mem wrb/cl/BU2/U0/q i 12](#)

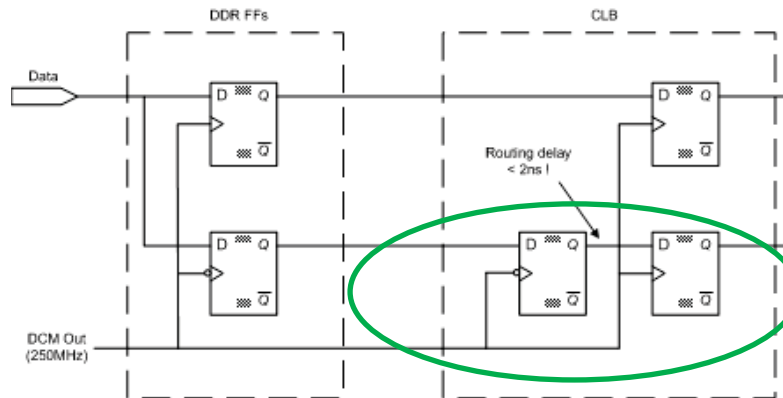
Delay type	Delay(ns)	Logical Resource(s)
Tcko	0.374	inst mem wrb/cl/BU2/U0/q i 11
net (fanout=46)	0.910	addr wr i<11>
Tilo	0.313	inst mem wrb/cl/BU2/U0/thresh0 i cmp eq000045
net (fanout=1)	0.260	inst mem wrb/cl/BU2/U0/thresh0 i cmp eq000045
Tilo	0.313	inst mem wrb/cl/BU2/U0/thresh0 i cmp eq000059
net (fanout=2)	0.053	inst mem wrb/cl/BU2/U0/thresh0 i
Tilo	0.288	inst mem wrb/cl/BU2/U0/q i or000001
net (fanout=8)	0.870	inst mem wrb/cl/BU2/U0/q i or00000
Tsrck	0.583	inst mem wrb/cl/BU2/U0/q i 12

Total	3.964ns	(1.871ns logic, 2.093ns route) (47.2% logic, 52.8% route)

Ejemplo de Diseño



Restricciones de Locación



Routing delay < 2ns !

Necesarios Constraints
 1 - CLB close to the DDR FFs
 2- RLOC para forzar FF_A sea adyacente a FF_B
 (en adyacente slice)

```
#####
# Rx RLOCs CONSTRAINTS #
#####

# The pairs of falling-edge to rising-edge flip-flops are grouped together
# in pairs. These are relatively placed with the rising-edge flip-flop in
# the slice above the falling-edge flip-flop since the relative Y-coordinate
# is 1 greater than that of its partner.

INST "rxl_reg" RLOC = X0Y0;
INST "rxo_reg" RLOC = X0Y1;
```

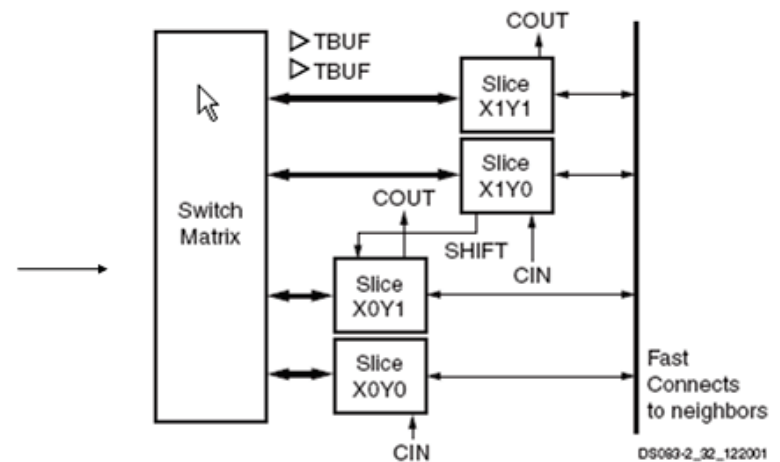


Figure 32: Virtex-II Pro CLB Element



Restricciones del Proyecto

```
# ----- #
# Set Internal clock from DCM to 250MHz
# ----- #
NET "fpga_clk_250" TNM_NET = "fpga_clk_250";
TIMESPEC "TS_fpga_clk_250" = PERIOD "fpga_clk_250" 4 ns HIGH 50 %;
# The pairs of falling-edge to rising-edge flip-flops are grouped together
# in pairs. These are relatively placed with the rising-edge flip-flop in
# the slice above the falling-edge flip-flop since the relative Y-coordinate
# is 1 greater than that of its partner.
#
# ABUS
INST *abus_fe2re_gen[7].abus_fe2re/data_i RLOC = X0Y0;
INST *abus_fe2re_gen[7].abus_fe2re/out_data RLOC = X0Y1;

# ----- ABus & PBus Constraints ----- #
# Route from falling edge to rising edge flip-flops
# for the CLB pair of FFs
NET *data_i* MAXDELAY= 960 ps;
```



Reporte de Tiempo Después de P&R

=====

Timing constraint: TS_fpga_clk_250 = PERIOD TIMEGRP "fpga_clk_250" 4 ns HIGH 50%;

1412 paths analyzed, 932 endpoints analyzed, 0 failing endpoints

0 timing errors detected. (0 setup errors, 0 hold errors)

Minimum period is 3.998ns.

TIMING ANALYSIS POST D&D

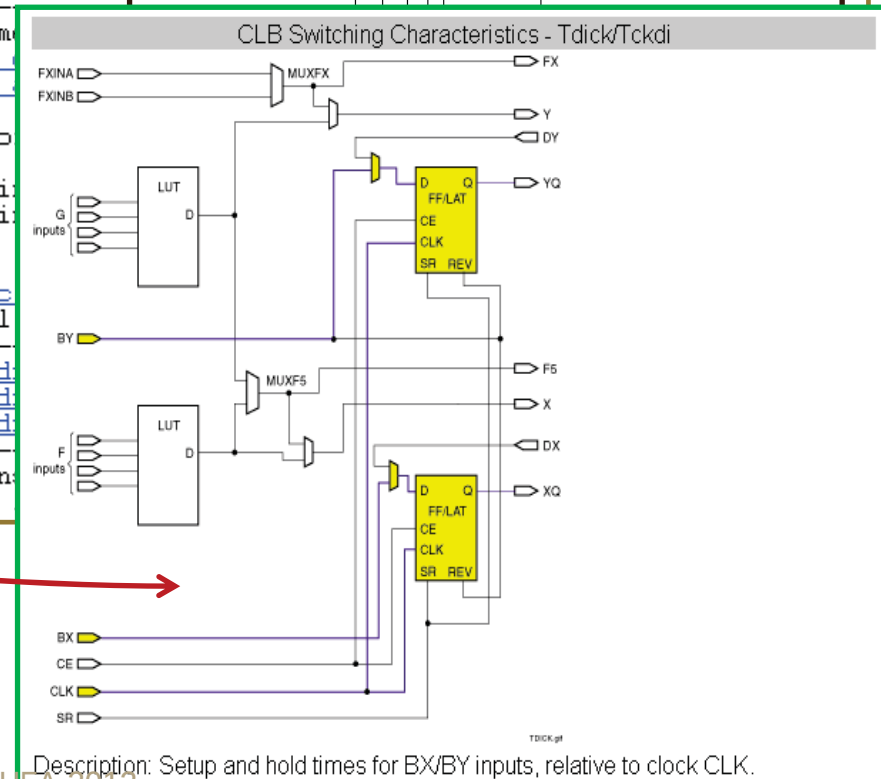
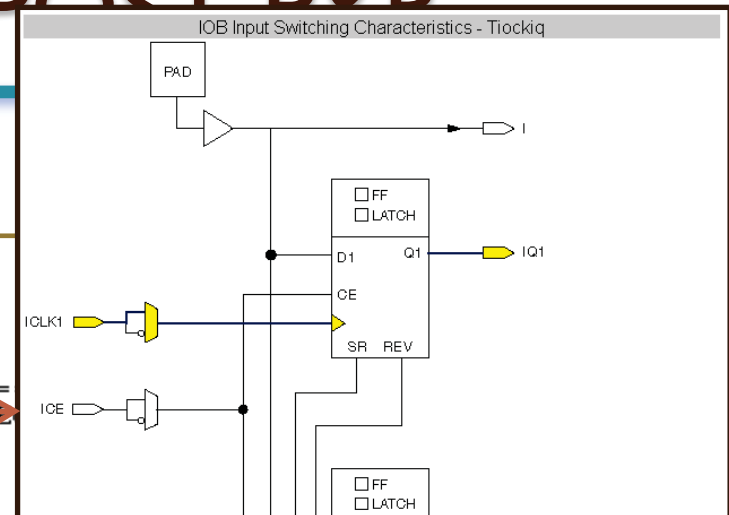
Timing constraint: TS_fpga_clk_250 = PERIOD TIME

1412 items analyzed, 0 timing errors detected.
Minimum period is 3.998ns.

Slack: 0.002ns (requirement)
Source: [inst_dds_rx/abus](#)
Destination: [inst_dds_rx/abus](#)
Requirement: 4.000ns
Data Path Delay: 3.955ns (Levels of delay)
Clock Path Skew: -0.043ns
Source Clock: fpga_clk_250 rising edge
Destination Clock: fpga_clk_250 rising edge
Clock Uncertainty: 0.000ns

Data Path: [inst_dds_rx/abus_dds_rx_sync](#)

Delay type	Delay(ns)	Logical
Tiockig	0.446	inst_dds
net (fanout=1)	3.307	inst_dds
Ttick	0.202	inst_dds
Total	3.955ns (0.648ns)	(16.4%)



Description: Setup and hold times for BX/BY inputs, relative to clock CLK.



FLOORPLANNING



Flooplanning

- Herramienta grafica usada para mostrar y *editar* el diseño dentro del FPGA
 - Un mal 'floorplainning' puede decrementar el rendimiento del diseño, tal como un buen 'floorplainning' puede mejorarlo
- Usos
 - Incrementar la productividad y rendimiento
 - Ver el diseño colocado en el FPGA
 - Realizar modificaciones de colocación
 - Crear areas y grupos de colocación



Floorplanning – Requisitos

- Conocer bien
 - El diseño
 - La arquitectura del FPGA
 - El software que use (Quartus/ISE)
- Antes de usar floorplanner use:
 - Restricciones de tiempo
 - Incremente el esfuerzo de P&R
 - Use pipeline en caminos criticos
 - Use ruteo re-entrante o multiple-paso P&R

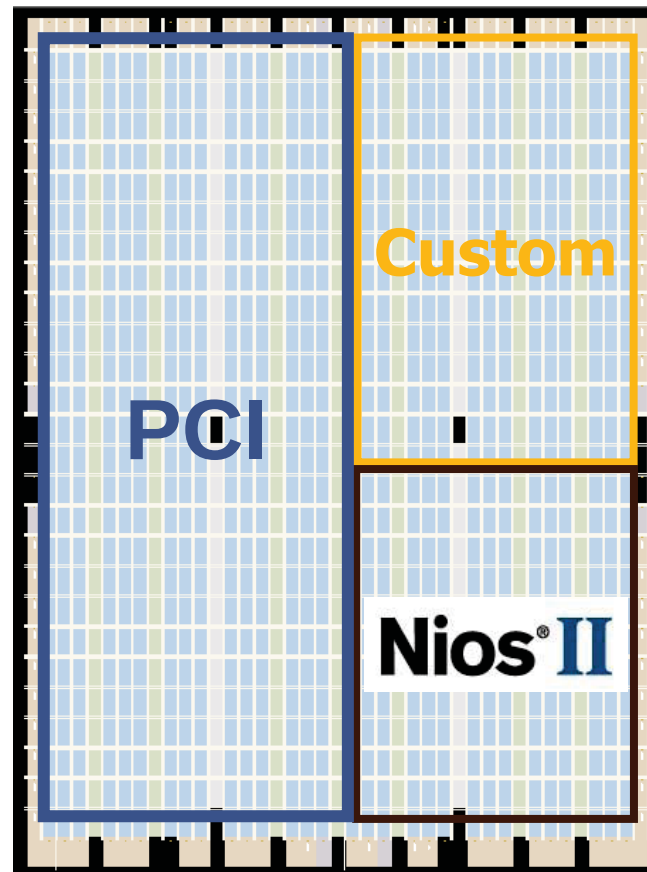


Qué es una Región?

- Área del FPGA en la cual se puede asignar lógica
- Tamaño y Locación controlada por el software (automáticamente) o por el usuario (manualmente)
- Región
 - Rectangular
 - No-rectangular

Partición Física del FPGA - Regiones

- Asignación de funciones lógicas a regiones específicas del FPGA

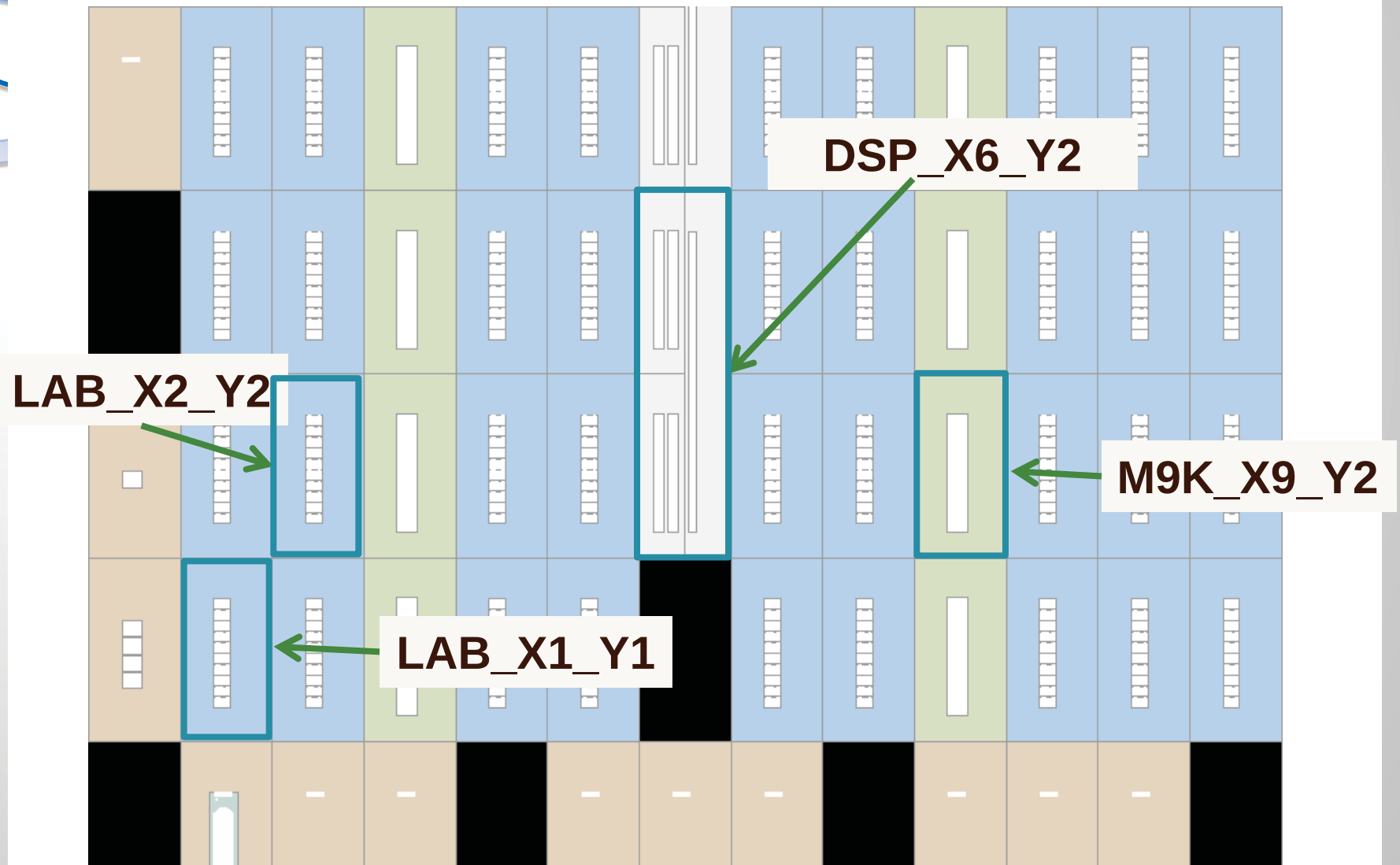




Definición de las Regiones Lógicas

- Origen de la región:
 - Indica la esquina inferior izquierda del bloque de la región
- Ancho y alto determinan el tamaño

Ejemplo de Coordenada de Regiones





ESTILO DEL CÓDIGO VHDL



Estilo de Código Para Rendimiento

- Pipeline
- Use instrucción *case* en lugar de *if*
- Registre *todas* las salidas de cada módulo
- Evite *if-then-else* anidados
- Particione jerárquicamente el diseño
- Use one-hot para codificar las MEF
- En las MEF considere usar un proceso para la lógica de próximo estado y otro para la lógica de salida



Estilo de Código para Confiabilidad

- Evite generación de *latches* indeseados
- Evite la generación de reloj generados con lógica del FPGA
- Evite la generación de reset asincrónicos con lógica del FPGA
- Use buffer globales para las señales de alto fan-out
- Use *clock enable*, no use clocks divididos internamente



Opciones de Herramienta Síntesis

- Síntesis orientada a max rendimiento
- Editor de restricciones de tiempo
- Extracción de MEF
- Re-temporizado
- Duplicación de registros
- Visor de esquemático(s)
- Comprobación cruzada (sch-VHDL)
- STA

- Preguntas ?





- **Contacto:**
 - cristian@unsj.edu.ar
 - cristian@c7t-hdl.com
- **Mas información (cursos, notas de aplicación, etc):**
 - <http://www.c7t-hdl.com>
- **Blog**
 - <http://hdl-fpga.blogspot.com.ar/>