

# Proceso de Implementación y Análisis de Temporización

# Contenido

- **Realización del Proceso de Implementacion**
- **Generacion de Reportes de Implementacion**
- **Descripcion del analisis estatico de temporizacion y los caminos estaticos de temporizacion**
- **Descripcion de las verificaciones de tiempos de estabilizacion (setup) y retencion (hold)**
- **Analisis de la relacion entre la frecuencia de los relojes y la verificacion de tiempos**
- **Generacion de reportes para hacer analisis estatico de temporizacion**
- **Usu del reporte *Timing Summary* para verificar el cumplimiento de las restricciones temporales**
- **Generación del Archivo de Configuración de la FPGA y configuración de la misma**

# Temario

- **Proceso de Implementacion**
- Reportes
- Analisis estático de temporizacion
- Generacion del archivo de configuracion y configuración de la FPGA
- Resumen

# Proceso de Implementación

El proceso de Implementación se compone de 5 subprocessos

- **Opt\_design: Optimización de la lógica**
- **Power\_opt\_design: Optimización del consumo**
- **Place\_design: ubicación física de los elementos**
- **Phys\_opt\_design: optimización de las redes post-ubicación**
- **Route\_design: Ruteo del sistema**

## Subproceso opt\_design: Optimización de la Logica

- **Busca una netlist optima para ubicacion y ruteo**
  - Se hace una nueva optimizacion de la lógica en la netlist completa a partir del RTL sintetizado y la IP agregada
    - Realiza recortes de la lógica no utilizada en la netlist de entrada
    - Busca un tiempo de propagacion constante
    - Recalcula las ecuaciones de las LUTs
- **Es opcional en desarrollos independientes (aunque es recomendable)**
  - Por ejemplo: elimina la logica asociada a bancos no usados cuando se utiliza la IP *MIG* (Memory Interface Generator) (bloques phaser/iodelay/....)
- **Se ejecuta automaticamente en desarrollos basados en proyectos**

[-] <b>Opt Design (opt_design)</b>	
is_enabled	<input checked="" type="checkbox"/>
tcl.pre	
tcl.post	
-verbose	<input type="checkbox"/>
-directive	Default
More Options	Explore
[-] <b>Power Opt Design (power_opt_design)</b>	
is_enabled	ExploreArea
tcl.pre	ExploreSequentialArea
tcl.post	AddRemap
More Options	RuntimeOptimized
	NoBramPowerOpt
	Default

# Opciones de opt\_design

## ➤ Desarrollos independientes

– Se puede especificar cuales optimizaciones realizar en un script

opt\_design

Description:

Optimize the current netlist. This will perform the retarget, propconst, sweep and bram\_power\_opt optimizations by default.

Syntax:

```
opt_design [-retarget] [-propconst] [-sweep] [-bram_power_opt] [-remap]
           [-resynth_area] [-resynth_seq_area] [-directive <arg>] [-quiet]
           [-verbose]
```

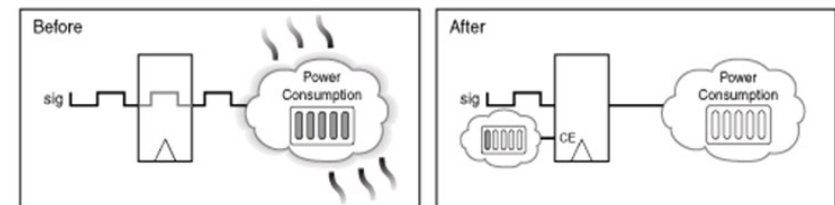
Usage:

Name	Description
-----	-----
[-retarget]	Retarget
[-propconst]	Propagate constants across leaf-level instances
[-sweep]	Remove unconnected leaf-level instances
[-bram_power_opt]	Perform Block RAM power optimizations
[-remap]	Remap logic optimally in LUTs
[-resynth_area]	Resynthesis
[-resynth_seq_area]	Resynthesis (with Sequential optimizations)
[-directive]	Mode of behavior (directive) for this command. Please refer to Arguments section of this help for values for this option
	Default: Default
[-quiet]	Ignore command errors
[-verbose]	Suspend message limits during command execution

## Subproceso power\_opt: Optimización del consumo

- La optimización del consumo incluye realizar habilitaciones de reloj (clock gating), de manera de lograr reducciones del consumo dinámico de hasta un 30%
- Las optimizaciones de habilitación de reloj se realizan a nivel del sistema completo y no realizan cambios en la logica o las señales de reloj definidas
- Esta técnica también se aplica sobre los bloques IP

Power Opt Design (power_opt_design)	
is_enabled	<input type="checkbox"/>
tcl.pre	
tcl.post	
More Options	



# Opciones para optimizacion de consumo

## ➤ Reduccion automatica del consumo

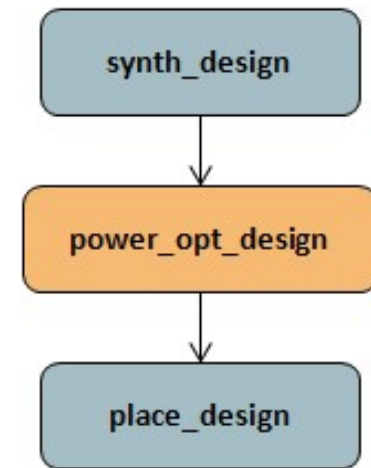
- Automaticamente se apagan las partes no usadas del sistema
- No requiere conocimientos detallados del sistema

## ➤ La interface grafica permite la optimizacion a nivel global y a nivel objetos

- Comando para optimizacion global: `power_opt_design`
- Optimizacion a nivel local a traves del comando SDC

`set_power_opt`

- Instancia: Incluir/excluir instancias de la optimizacion
- Dominio de Reloj:: Optimizar instancias sincronizadas por un reloj especifico
- Tipo de Celda: Block RAM, registros, SRL, DSP, etc





# Subproceso place\_design: ubicación física de los elementos

## ➤ Desarrollos basados en proyectos

- Se ejecuta en la etapa de implementación

## ➤ Desarrollos independientes

- Comando Tcl `place_design`

## ➤ Puede usar un archivo XDEF para ubicaciones iniciales

<input type="checkbox"/> <b>Place Design (place_design)</b>	
tcl.pre	
tcl.post	
<b>-directive</b>	<b>Default</b>
More Options	SSI_SpreadSLLs
<input type="checkbox"/> <b>Post-Place Power Opt Design (power_opt)</b>	SSI_BalanceSLLs
is_enabled	SSI_BalanceSLRs
tcl.pre	SSI_HighUtilSLRs
tcl.post	RuntimeOptimized
More Options	Quick
<input type="checkbox"/> <b>Post-Place Phys Opt Design (phys_opt_de)</b>	<b>Default</b>

place\_design

Description:

Automatically place ports and leaf-level instances

Syntax:

```
place_design [-directive <arg>] [-no_timing_driven] [-timing_summary]
             [-unplace] [-post_place_opt] [-quiet] [-verbose]
```

Usage:

Name	Description
-----	-----
[-directive]	Mode of behavior (directive) for this command. Please refer to Arguments section of this help for values for this option. Default: Default
[-no_timing_driven]	Do not run in timing driven mode
[-timing_summary]	Enable accurate post-placement timing summary.
[-unplace]	Unplace all the instances which are not locked by Constraints.
[-post_place_opt]	Run only the post commit optimizer
[-quiet]	Ignore command errors
[-verbose]	Suspend message limits during command execution

# Características del subproceso de Ubicacion

## ➤ Etapas de la Ubicacion

- Chequeo DRC Pre-ubicacion
  - Verifica conexiones no ruteables, restricciones fisicas validas y sobreutilizacion de recursos
- Ubicacion
  - Ubicacion de I/O y relojes

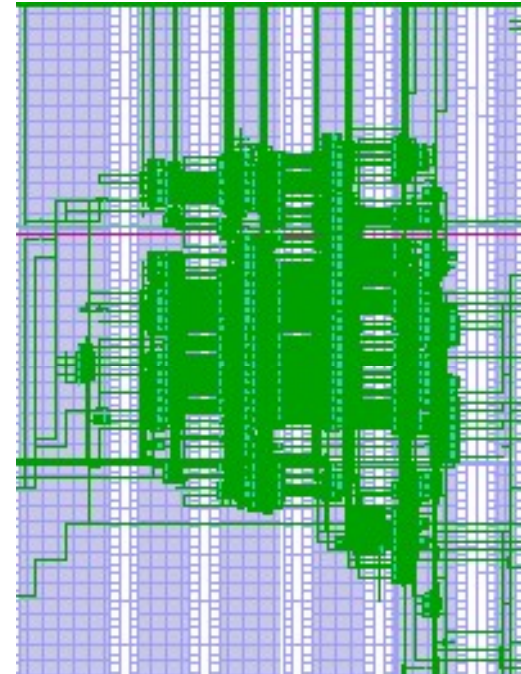
## ➤ Ubicacion de Macros y bloques basicos

- Comandadas por temporizacion y por recorrido de señales
- Busca evitar el congestionamiento en el ruteo

## ➤ Ubicacion detallada

- Establece la ubicacion de flip-flops, LUTs, etc.
- Empaqueta los bloques en Slices

## ➤ Optimizaciones Post-ubicacion detallada



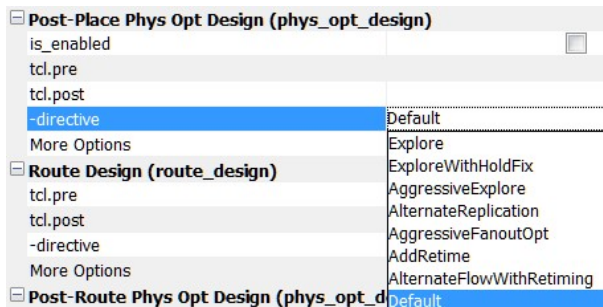
# Subproceso phys\_opt\_design: Optimización de las redes post-ubicación

## ➤ Optimización Post-ubicación comandada por temporización

- Copia y ubica amplificadores para redes con alto fanout y retardo negativo
  - La copia solo se hace si mejora la temporización
  - El retardo debe estar dentro de un límite específico
    - Alrededor del 10% del worst negative slack (WNS)

## ➤ Disponible en la interface grafica

- phys\_opt\_design
- Se ejecuta entre *place\_design* y *route\_design*



```
phys_opt_design

Description:
Optimize the current placed netlist.

Syntax:
phys_opt_design [-fanout_opt] [-placement_opt] [-routing_opt] [-rewire]
                [-critical_cell_opt] [-dsp_register_opt] [-bram_register_opt]
                [-bram_enable_opt] [-shift_register_opt] [-hold_fix] [-retime]
                [-force_replication_on_nets <args>] [-directive <arg>]
                [-critical_pin_opt] [-clock_opt] [-quiet] [-verbose]

Usage:
Name                Description
-----
[-fanout_opt]       Do cell-duplication based optimization on
                    high-fanout timing critical nets
[-placement_opt]   Do placement based optimization on timing
                    critical nets
[-routing_opt]     Do routing based optimization on timing
                    critical nets
[-rewire]          Do rewiring optimization
[-critical_cell_opt] Do cell-duplication based optimization on
                    timing critical nets
[-dsp_register_opt] Do DSP register optimization
[-bram_register_opt] Do BRAM register optimization
[-bram_enable_opt]  Do BRAM enable optimization
[-shift_register_opt] Do Shift register optimization
[-hold_fix]        Attempt to improve slack of high hold
                    violators
[-retime]          Do retiming optimization
[-force_replication_on_nets] Force replication optimization on nets
[-directive]       Mode of behavior (directive) for this
                    command. Please refer to Arguments section of
                    this help for values for this option
                    Default: Default
[-critical_pin_opt] Do pin-swapping based optimization on timing
                    critical nets
[-clock_opt]       Do clock skew optimization in post-route
                    optimization
[-quiet]           Ignore command errors
[-verbose]         Suspend message limits during command
                    execution
```

# Subproceso route\_design: Ruteo del sistema

## ➤ Desarrollos basados en proyectos

– Se incluye en la etapa de implementacion

## ➤ Desarrollos independientes

– Comando Tcl `route_design`

## ➤ Reporte de ruteo

– Comando Tcl `report_route_status`

– Verifica el ruteo de redes individuales

- Totalmente ruteadas: lista los recursos de ruteo
- Listado de Ruteos fallidos

Route Design (route_design)	
tcl.pre	
tcl.post	
-directive	Default
More Options	Explore
Post-Route Phys Opt Design (phys_opt_d	NoTimingRelaxation
is_enabled	MoreGlobalIterations
tcl.pre	HigherDelayCost
tcl.post	AdvancedSkewModeling
-directive	RuntimeOptimized
More Options	Quick
	Default

route\_design

Description:

Route the current design

Syntax:

```
route_design [-unroute] [-release_memory] [-nets <args>] [-physical_nets]
[-pin <arg>] [-directive <arg>] [-tns_cleanup]
[-no_timing_driven] [-preserve] [-delay] -max_delay <arg>
-min_delay <arg> [-timing_summary] [-quiet] [-verbose]
```

Usage:

Name	Description
-----	-----
[-unroute]	Unroute whole design or the given nets/pins if used with -nets or -pin.
[-release_memory]	Release Router memory. Not compatible with any other options.
[-nets]	Operate on the given nets.
[-physical_nets]	Operate on all physical nets.
[-pin]	Operate on the given pin.
[-directive]	Mode of behavior (directive) for this command. Please refer to Arguments section of this help for values for this option. Default: Default
[-tns_cleanup]	Do optional TNS clean up.
[-no_timing_driven]	Do not run in timing driven mode.
[-preserve]	Preserve existing routing.
[-delay]	Use with -nets or -pin option to route in delay driven mode.
-max_delay	Use with -pin option to specify the max_delay constraint on the pin. When specified -delay is implicit.
-min_delay	Use with -pin option to specify the max_delay constraint on the pin. When specified -delay is implicit.
[-timing_summary]	Enable post-router signoff timing summary.
[-quiet]	Ignore command errors
[-verbose]	Suspend message limits during command execution

# Características del proceso de Ruteo

## ➤ Etapas del ruteo

- Ruteo de redes especiales y señales de reloj
- Ruteo comandado por temporización
  - Priorizado por tiempos de setup/hold
  - Intercambio de entradas en las LUTs para mejorar los caminos criticos
  - Corrige dentro de ciertos limites las violaciones a los tiempos de setup/hold violations

## ➤ Dos modos

- Normal (por defecto): El ruteo comienza con los componentes ubicados y se intenta rutear todas las redes
- Re-Entrante (solo en diseños independientes): se puede rerutear y mantener fijo el ruteo de redes especificas

# Temario

- Proceso de Implementacion
- **Reportes**
- Analisis estático de temporizacion
- Generacion del archivo de configuracion y configuración de la FPGA
- Resumen

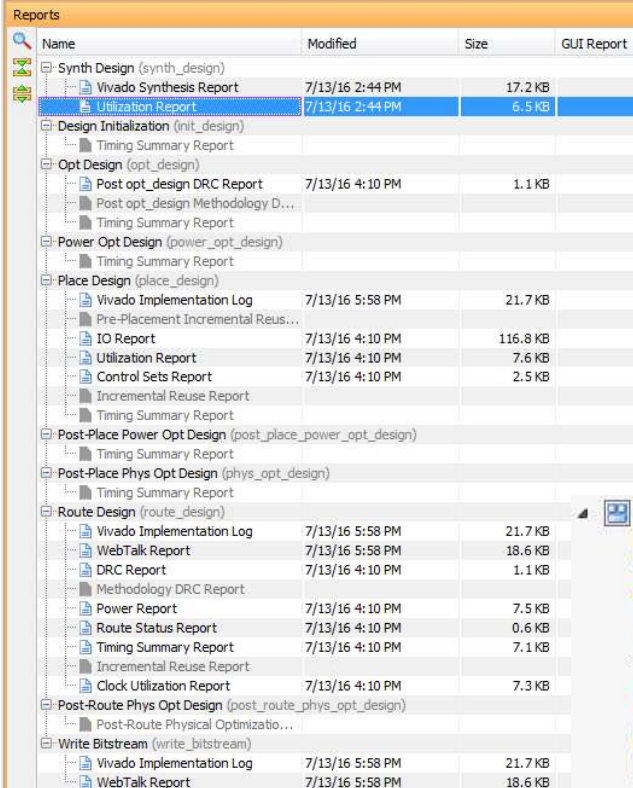
## Luego de ejecutarse el proceso de Implementacion:

- Las pestañas de Fuentes (Sources) y Redes (Netlist) no cambian
  - A medida que se selecciona un recurso, se muestra su ubicacion en el dispositivo físico (FPGA)
- Los resultados de temporizacion se pueden ver en el reporte *Timing Summary*
- Al seleccionar cada red, la ubicacion de la lógica y el ruteo se muestra en la vista del dispositivo
  - Esta forma de ver la ubicacion ayuda cuando es necesario reubicar para mejorar el analisis de temporizacion

Name	Constraints	WNS	TNS	WHS	THS	TPWS	Failed Routes	LUT	FF	BRAM	DSP	Start	Elapsed	Status	Progress	Strategy
synth_1	constrs_1							0.23	0.08	0.00	0.00	7/21/14 4:14 PM	00:07:59	Synthesis Out-of-date	100%	Vivado Synthesis Defaults (V)
impl_1	constrs_1													Not started	0%	Vivado Implementation Defau
synth_2	constrs_1							4.34	2.28	2.86	0.00	8/6/14 6:30 PM	00:00:48	Synthesis Out-of-date	100%	Vivado Synthesis Defaults* (
impl_2	constrs_1	2.45	0.00	0.04	0.00	0.00	0	7.69	5.22	7.50	0.00	8/6/14 6:38 PM	00:05:42	Implementation Out-of-date	100%	Vivado Implementation Defau
synth_3 (active)	constrs_1							4.34	2.28	2.86	0.00	8/7/14 12:06 PM	00:00:48	synth_design Complete!	100%	Vivado Synthesis Defaul
impl_3 (active)	constrs_1	2.32	0.00	0.02	0.00	0.00	0	8.17	5.49	7.86	0.00	8/7/14 12:32 PM	00:07:11	write_bitstream Complete!	100%	Vivado Implementation I

# Reportes de Implementacion

- En el menu *Flow Navigator* se muestran los reportes mas usados, mientras que la pestaña *Reports* contiene otros reportes mas detallados
  - Reporte *Post Optimization DRC*: Enumera los chequeos de DRC en las I/O que fueron exitosos
  - Reporte *Post Power Optimization DRC*: Enumera los chequeos de DRC de consumo que fueron exitosos
  - Reporte *Place and Route Log*: Describe el proceso de implementacion y los resultados obtenidos
  - Reporte *IO*: Enumera el pinout final del sistema
  - Reporte *Clock Utilization*: Describe los recursos de reloj utilizados y la utilizacion de recursos en cada dominio de reloj
  - Reporte *Utilization*: Describe los recursos de la FPGA utilizados en formato texto
  - Reporte *Control Sets*: describe como se agruparon las señales de control



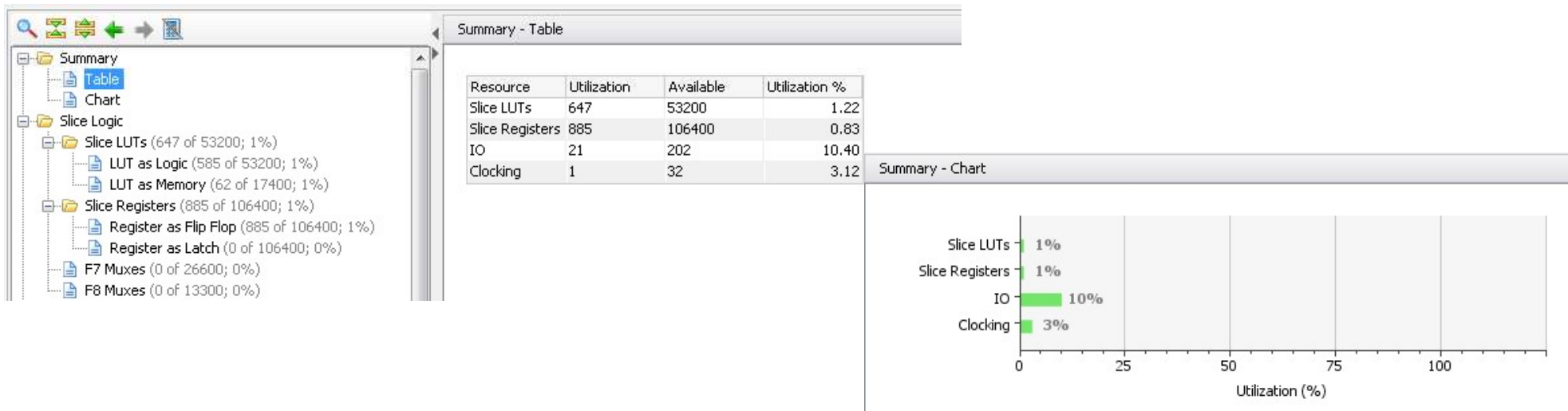
Name	Modified	Size	GUI Report
Synth Design (synth_design)			
Vivado Synthesis Report	7/13/16 2:44 PM	17.2 KB	
Utilization Report	7/13/16 2:44 PM	6.5 KB	
Design Initialization (init_design)			
Timing Summary Report			
Opt Design (opt_design)			
Post opt_design DRC Report	7/13/16 4:10 PM	1.1 KB	
Post opt_design Methodology D...			
Timing Summary Report			
Power Opt Design (power_opt_design)			
Timing Summary Report			
Place Design (place_design)			
Vivado Implementation Log	7/13/16 5:58 PM	21.7 KB	
Pre-Placement Incremental Reus...			
IO Report	7/13/16 4:10 PM	116.8 KB	
Utilization Report	7/13/16 4:10 PM	7.6 KB	
Control Sets Report	7/13/16 4:10 PM	2.5 KB	
Incremental Reuse Report			
Timing Summary Report			
Post-Place Power Opt Design (post_place_power_opt_design)			
Timing Summary Report			
Post-Place Phys Opt Design (phys_opt_design)			
Timing Summary Report			
Route Design (route_design)			
Vivado Implementation Log	7/13/16 5:58 PM	21.7 KB	
WebTalk Report	7/13/16 5:58 PM	18.6 KB	
DRC Report	7/13/16 4:10 PM	1.1 KB	
Methodology DRC Report			
Power Report	7/13/16 4:10 PM	7.5 KB	
Route Status Report	7/13/16 4:10 PM	0.6 KB	
Timing Summary Report	7/13/16 4:10 PM	7.1 KB	
Incremental Reuse Report			
Clock Utilization Report	7/13/16 4:10 PM	7.3 KB	
Post-Route Phys Opt Design (post_route_phys_opt_design)			
Post-Route Physical Optimizatio...			
Write Bitstream (write_bitstream)			
Vivado Implementation Log	7/13/16 5:58 PM	21.7 KB	
WebTalk Report	7/13/16 5:58 PM	18.6 KB	

- Implemented Design
  - Constraints Wizard
  - Edit Timing Constraints
  - Report Timing Summary
  - Report Clock Networks
  - Report Clock Interaction
  - Report DRC
  - Report Noise
  - Report Utilization
  - Report Power



# Reporte Utilization

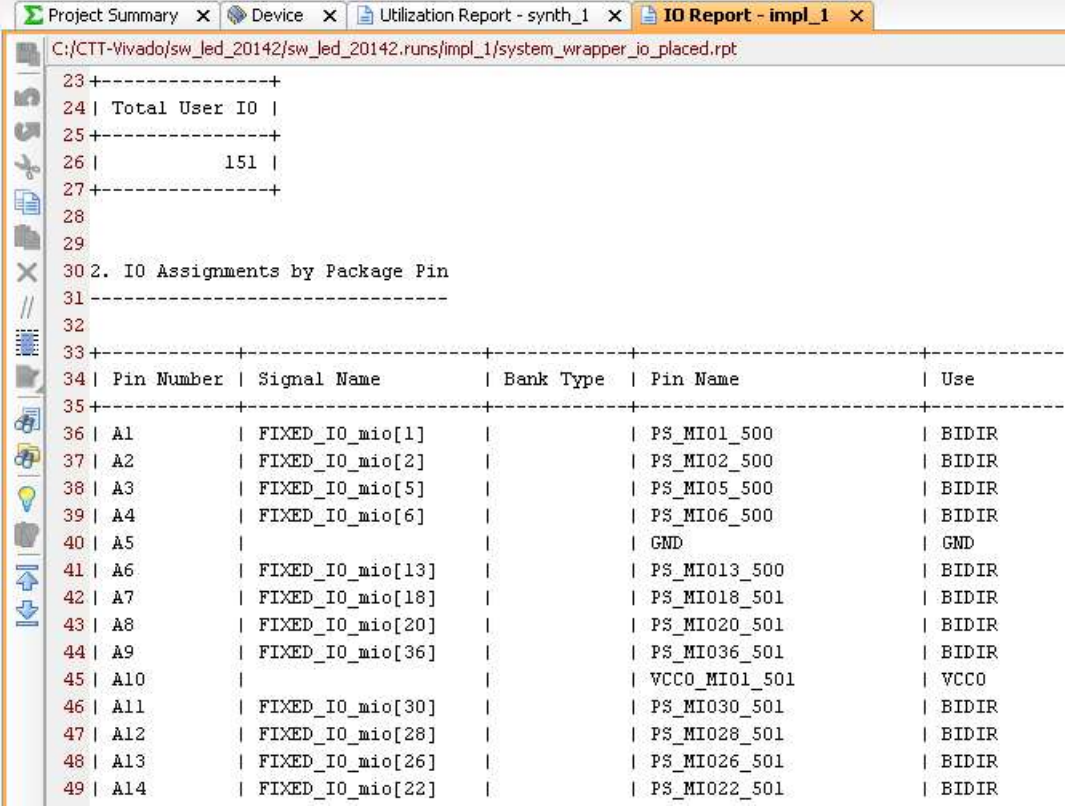
- Presionar dos veces en la pestaña *Reports* para verlo en formato texto
- En el menu *Flow Navigator*, en la seccion *Implementation*, Presionar en el reporte **Utilization** para ver los resultados en formato tabular



# Reporte I/O

➤ En este reporte se ve una tabla con el listado de todas las señales, sus propiedades y su ubicacion

– Es importante verificar la asignacion de pines, ya que las herramientas reubican todos los pines que no tienen señales asignadas. Por ejemplo, si se define un bus de 16 bits en I/O, pero solo se utilizan 12 en el sistema; los 4 no utilizados pueden ser reubicados en lugares fisicos no relacionados con los 12 utilizados.



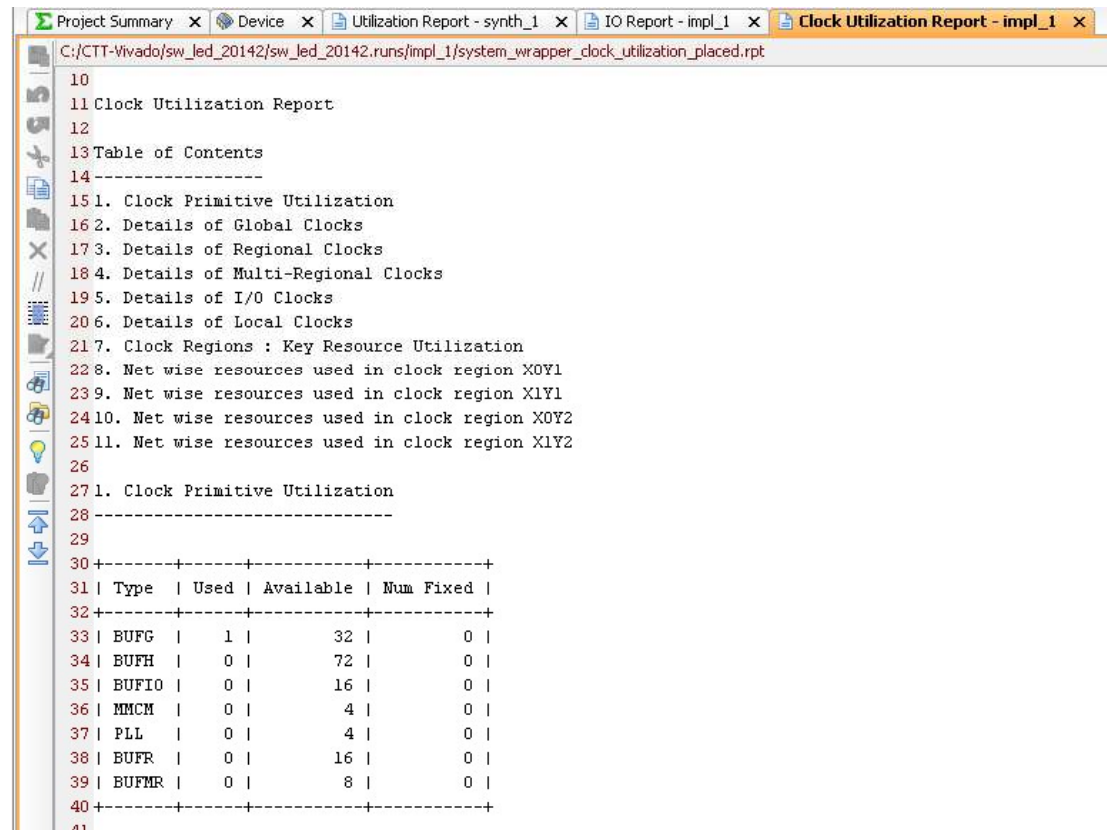
The screenshot shows the Vivado IO Report window. The title bar includes 'Project Summary', 'Device', 'Utilization Report - synth\_1', and 'IO Report - impl\_1'. The main content area displays the following text:

```
C:/CTT-Vivado/sw_led_20142/sw_led_20142.runs/impl_1/system_wrapper_io_placed.rpt
23 +-----+
24 | Total User IO |
25 +-----+
26 |           151 |
27 +-----+
28
29
30 2. IO Assignments by Package Pin
31 -----
32
33 +-----+-----+-----+-----+-----+
34 | Pin Number | Signal Name | Bank Type | Pin Name | Use |
35 +-----+-----+-----+-----+-----+
36 | A1 | FIXED_IO_mio[1] | | PS_MI01_500 | BIDIR |
37 | A2 | FIXED_IO_mio[2] | | PS_MI02_500 | BIDIR |
38 | A3 | FIXED_IO_mio[5] | | PS_MI05_500 | BIDIR |
39 | A4 | FIXED_IO_mio[6] | | PS_MI06_500 | BIDIR |
40 | A5 | | | GND | GND |
41 | A6 | FIXED_IO_mio[13] | | PS_MI013_500 | BIDIR |
42 | A7 | FIXED_IO_mio[18] | | PS_MI018_501 | BIDIR |
43 | A8 | FIXED_IO_mio[20] | | PS_MI020_501 | BIDIR |
44 | A9 | FIXED_IO_mio[36] | | PS_MI036_501 | BIDIR |
45 | A10 | | | VCC0_MI01_501 | VCC0 |
46 | A11 | FIXED_IO_mio[30] | | PS_MI030_501 | BIDIR |
47 | A12 | FIXED_IO_mio[28] | | PS_MI028_501 | BIDIR |
48 | A13 | FIXED_IO_mio[26] | | PS_MI026_501 | BIDIR |
49 | A14 | FIXED_IO_mio[22] | | PS_MI022_501 | BIDIR |
```

# Reporte Clock Utilization

➤ Este reporte describe los recursos de reloj utilizados en el sistema

- Listado de BUFG, BUFH, BUFHCE, MMCM, y un analisis por dominio de reloj



```
C:/CTT-Vivado/sw_led_20142/sw_led_20142.runs/impl_1/system_wrapper_clock_utilization_placed.rpt
10
11 Clock Utilization Report
12
13 Table of Contents
14 -----
15 1. Clock Primitive Utilization
16 2. Details of Global Clocks
17 3. Details of Regional Clocks
18 4. Details of Multi-Regional Clocks
19 5. Details of I/O Clocks
20 6. Details of Local Clocks
21 7. Clock Regions : Key Resource Utilization
22 8. Net wise resources used in clock region XOY1
23 9. Net wise resources used in clock region XIY1
24 10. Net wise resources used in clock region XOY2
25 11. Net wise resources used in clock region XIY2
26
27 1. Clock Primitive Utilization
28 -----
29
30 +-----+-----+-----+-----+
31 | Type | Used | Available | Num Fixed |
32 +-----+-----+-----+-----+
33 | BUFG | 1 | 32 | 0 |
34 | BUFH | 0 | 72 | 0 |
35 | BUFIO | 0 | 16 | 0 |
36 | MMCM | 0 | 4 | 0 |
37 | PLL | 0 | 4 | 0 |
38 | BUFR | 0 | 16 | 0 |
39 | BUFMR | 0 | 8 | 0 |
40 +-----+-----+-----+-----+
41
```

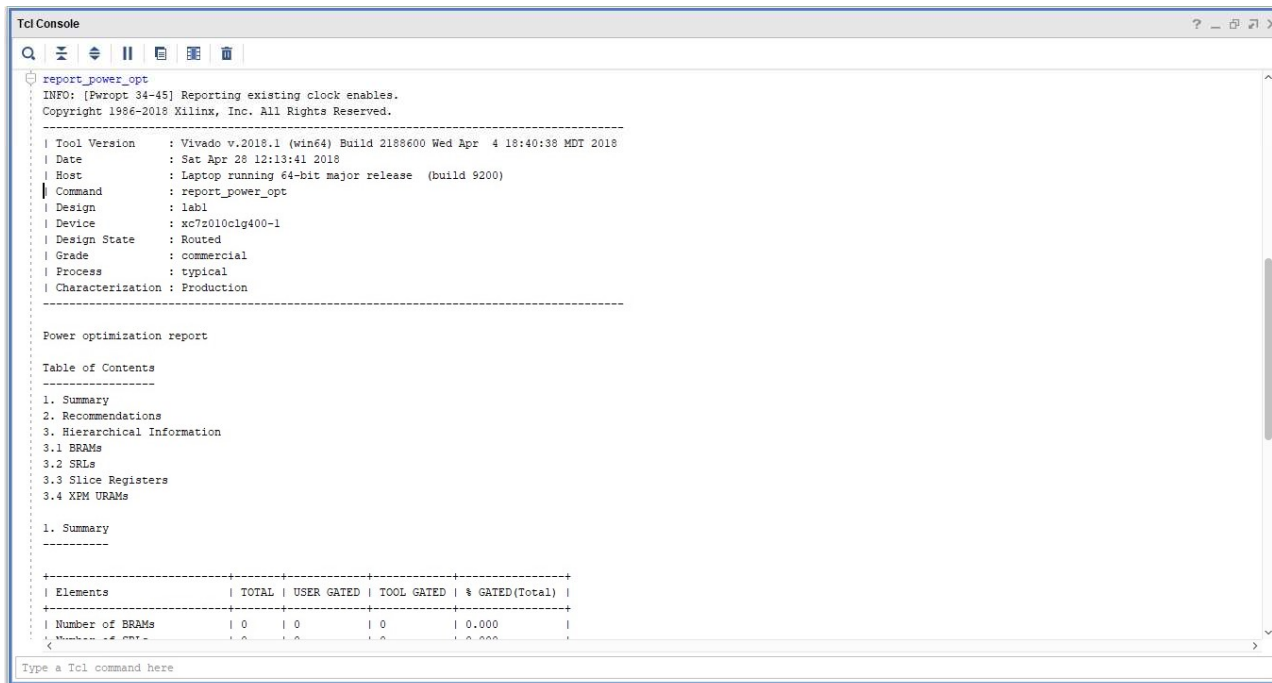
# Reporte Control Sets

- Este reporte describe los grupos de señales de control en el sistema (idealmente debería ser los menos posibles)
- El numero de grupos describe como se agrupan las señales de control
  - Esto determina la habilidad de las herramientas de lograr una alta utilizacion del dispositivo fisico
  - El numero de señales de control esta determinado por la inferencia de señales de set/reset y habilitacion de reloj
  - El numero de señales de control se puede reducir si estas se comparten en el sistema

```
C:/xup/fpga_flow/2016_2_ZYNQ_labs/zed/lab4/lab4.runs/impl_1/wave_gen_top_control_sets_placed.rpt
4 | Date       : Wed Jul 20 09:36:14 2016
5 | Host       : XSJNFPURUSH030 running 64-bit Service Pack 1 (build 7601)
6 | Command    : report_control_sets -verbose -file wave_gen_top_control_sets_placed.rpt
7 | Design     : wave_gen_top
8 | Device     : xc7z020
9
10
11 Control Set Information
12
13 Table of Contents
14 -----
15 1. Summary
16 2. Flip-Flop Distribution
17 3. Detailed Control Set Information
18
19 1. Summary
20 -----
21
22 -----+-----+
23 |                               | Status | Count |
24 |-----+-----+-----+
25 | Number of unique control sets | 43 |
26 | Unused register locations in slices containing registers | 130 |
27 |-----+-----+-----+
28
29
30 2. Flip-Flop Distribution
31 -----
32
33 -----+-----+-----+-----+-----+
34 | Clock Enable | Synchronous Set/Reset | Asynchronous Set/Reset | Total Registers | Total Slices |
35 |-----+-----+-----+-----+-----+
36 | No           | No                    | No                    | 27 | 17 |
37 | No           | No                    | Yes                   | 107 | 32 |
38 | No           | Yes                   | No                    | 134 | 47 |
39 | Yes          | No                    | No                    | 43 | 11 |
40 | Yes          | No                    | Yes                   | 50 | 10 |
41 | Yes          | Yes                   | No                    | 229 | 64 |
42 |-----+-----+-----+-----+-----+
```

# Reporte power\_opt:

- Se puede invocar con el comando `Tcl report_power_opt`
  - Se puede ejecutar antes y después de la optimización de consumo, para analizar las mejoras obtenidas en el sistema



```
report_power_opt
INFO: [PwOpt 34-45] Reporting existing clock enables.
Copyright 1986-2018 Xilinx, Inc. All Rights Reserved.
-----
| Tool Version   : Vivado v.2018.1 (win64) Build 2188600 Wed Apr  4 18:40:38 MDT 2018
| Date          : Sat Apr 28 12:13:41 2018
| Host          : Laptop running 64-bit major release (build 9200)
| Command       : report_power_opt
| Design        : lab1
| Device        : xc7z010clg400-1
| Design State  : Routed
| Grade         : commercial
| Process       : typical
| Characterization : Production
-----

Power optimization report

Table of Contents
-----
1. Summary
2. Recommendations
3. Hierarchical Information
  3.1 BRAMs
  3.2 SRLs
  3.3 Slice Registers
  3.4 XPM URAMs
1. Summary
-----

-----+-----+-----+-----+-----+
| Elements      | TOTAL | USER GATED | TOOL GATED | % GATED(Total) |
-----+-----+-----+-----+-----+
| Number of BRAMs | 1 0   | 1 0       | 1 0       | 0.000          |
-----+-----+-----+-----+-----+

Type a Tcl command here
```

# Temario

- Proceso de Implementacion
- Reportes
- **Analisis estático de temporizacion**
- Generacion del archivo de configuracion y configuración de la FPGA
- Resumen

# Analisis estático de temporizacion (Static Timing Analysis – STA)

- **Un sistema es basicamente un conjunto de bloques lógicos interconectados**
- **La funcionalidad del sistema esta definida en el codigo fuente RTL**
  - Esta funcionalidad se verifica mediante simulacion y archivos de prueba
- **El desempeño del sistema esta determinado por los retardos en los bloques lógicos**
  - Esto se verifica mediante un analisis estático de temporizacion (STA)
- **En un analisis STA la funcionalidad de los bloques lógicos no es importante**
  - Solo es importante el desempeño de cada componente
  - De esta manera, solo es necesario clasificar los bloques lógicos en combinatorios o secuenciales

# Retardo de los componentes

## ➤ Cada componente tiene un retardo asociado

- Una LUT tiene un retardo de propagación desde la entrada a la salida
- El ruteo tiene un retardo de propagación desde el amplificador (buffer/driver) hasta el/los receptores
- Un flip-flop necesita que los datos estén estables un cierto tiempo antes y después del momento en que se almacenan

## ➤ Estos retardos dependen de distintos factores

- Algunos dependen del dispositivo (FPGA) y de la implementación del sistema
  - Las características físicas del componente (como y con que está construido)
  - La ubicación del componente (su ubicación absoluta y relativa a otros componentes)
- Otras características están determinadas por factores ambientales (PVT)
  - Las variaciones propias del proceso de fabricación del dispositivo (**P**)
  - La tensión de alimentación aplicada (**V**)
  - La temperatura del componente (**T**)



# Retardos

## ➤ Los retardos de componentes y ruteo son provistos por el fabricante de la FPGA, mediante caracterizacion de la implementacion fisica de los dispositivos

– La temporizacion se calcula sobre todo el rango de operacion del dispositivo

- Los retardos debidos al proceso de fabricacion (P) estan dentro de un intervalo

– Cada modelo tiene distintos intervalos(speed grades: -1, -2, -3, etc)

- Los retardos debidos a la alimentacion (V) se calculan dentro del maximo y el minimo que admite el dispositivo

– Dentro de un mismo modelo puede haber distintos rangos de alimentacion (p.ej. -1 y -1L)

- El retardo debido a la temperatura (T) tambien se calcula dentro del maximo y minimo que admite el dispositivo

– Un mismo modelo puede tener un rango Comercial (0° a 85°) o Industrial (-40° a 100°)

– Estos rangos de retardo (PVT) se utilizan por las herramientas para el analisis STA

## Aplicacion del analisis STA

- **Los procesos de implementacion del sistema son comandados por los requisitos de temporizacion**
  - Sintesis (Synthesis) para la construccion de la netlist
  - Ubicacion (Placer) para la ubicacion fisica de los componentes
  - Ruteo (Router) para la interconexion de los componentes
- **El analisis STA se utiliza antes y despues de cada proceso para obtener reportes de la temporizacion del sistema a medida que el mismo es implementado**
- **De esta manera, el analisis STA determina si el sistema cumple los requerimientos de temporizacion establecidos**

# Reporte report\_timing\_summary:

- El reporte de temporizacion se genera en el menu *Flow Navigator* luego de realizar el proceso de implementacion

The image shows two screenshots from a design tool. The top screenshot shows the 'Flow Navigator' menu with 'Report Timing Summary' highlighted in a red box. The bottom screenshot shows the 'Report Timing Summary' dialog box and the 'Design Timing Summary' report window.

**Report Timing Summary Dialog Box:**

- Results name: timing\_1
- Options: Advanced, Timer Settings
- Report: Path delay type: min\_max, Report unconstrained paths (checked), Report datasheet (unchecked)
- Path Limits: Maximum number of paths per clock or path group: 10, Maximum number of worst paths per endpoint: 1
- Path Display: Display paths with slack less than: (empty), Use default (1e+30) (checked), Significant digits: 3
- Command: report\_timing\_summary -delay\_type min\_max -report\_unconstrained -chec
- Open in a new tab (checked), Open in Timing Analysis layout (unchecked)

**Design Timing Summary Report:**

Setup	Hold	Pulse Width
Worst Negative Slack (WNS): 4.059 ns	Worst Hold Slack (WHS): 0.030 ns	Worst Pulse Width Slack (WPWS): 4.020 ns
Total Negative Slack (TNS): 0.000 ns	Total Hold Slack (THS): 0.000 ns	Total Pulse Width Negative Slack (TPWS): 0.000 ns
Number of Failing Endpoints: 0	Number of Failing Endpoints: 0	Number of Failing Endpoints: 0
Total Number of Endpoints: 1814	Total Number of Endpoints: 1814	Total Number of Endpoints: 956

**All user specified timing constraints are met.**

# Reporte Timing Summary

## ➤ Este reporte muestra los resultados de temporizacion para *Setup, Hold, y Pulse Width*

### – Setup

- Worst Negative Slack (WNS): El peor retardo en cada camino de señal, se utiliza para el calculo del maximo retardo. Puede ser positivo o negativo. Si es positivo significa que se cumplen los requisitos de temporizacion aun para el camino de mayor retardo. Si es negativo, significa que hay caminos de señal cuyo retardo es mayor que el requerido.
- Total Negative Slack (TNS): La suma de todos los WNS negativos asociados a un punto del circuito (endpoint)
- Number of Failing Endpoints: La cantidad de puntos del circuito que no cumplen los requerimientos de temporizacion

### – Hold

- Worst Hold Slack (WHS): El peor retardo de estabilizacion para cada camino de señal, se utiliza para calcular el retardo minimo del sistema

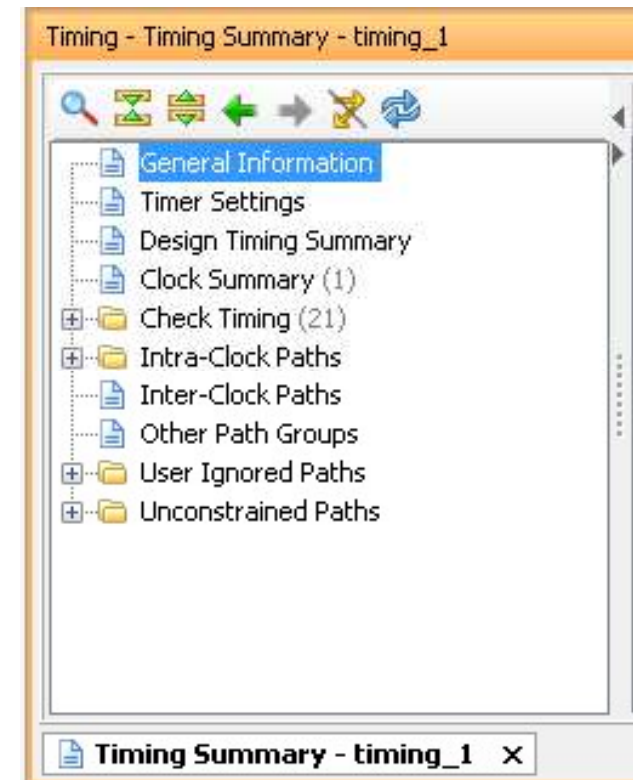
### – Pulse Width

- Worst Pulse Width Slack (WPWS): Es el peor retardo del sistema, calculado con los valores de Setup y Hold

# Reporte Timing Summary en formato de tabla

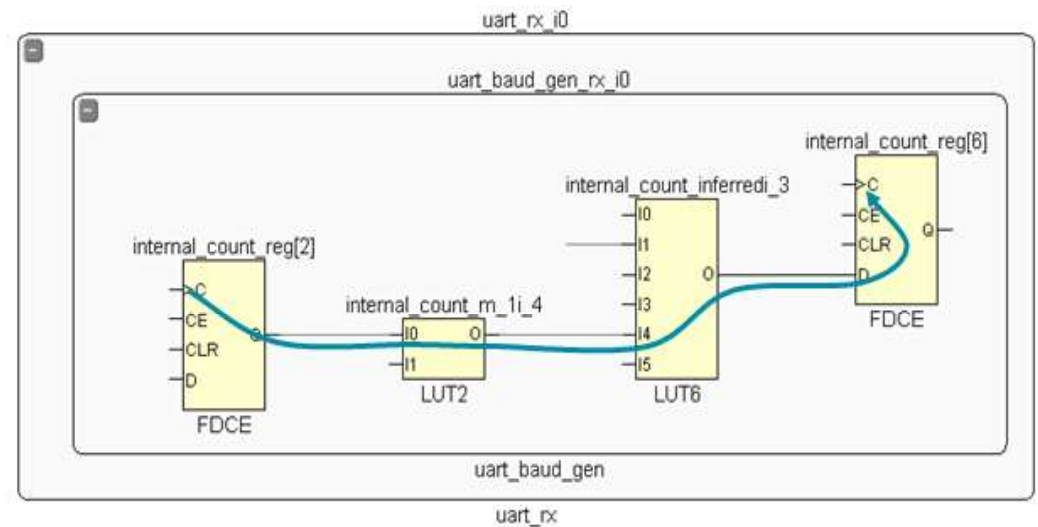
## ➤ La tabla Timing Summary muestra:

- General Information: nombre del proyecto, dispositivo, encapsulado, speed grade etc.
- Timer Settings: configuracion de la herramienta de analisis de temporizacion
- Design Timing Summary: resumen de todos los reportes de temporizacion
- Clock Summary: informacion resumida del reporte *report\_clocks*
- Check Timing: informacion sobre restricciones de temporizacion que no se cumplen y/o caminos de señal que deben revisarse
- Intra-Clock Paths: informacion de los valores de worst slack para caminos de señal que tienen el mismo reloj
- Inter-Clock Paths: informacion de los valores de worst slack para caminos de señal que tienen distintos relojes
- Other Path Groups: muestra los caminos de señal que no estan en los otros reportes
- User Ignored Paths: caminos de señal que no fueron tenidos en cuenta para el analisis de temporizacion
- Unconstrained Paths: caminos de señal que no tienen restricciones asociadas



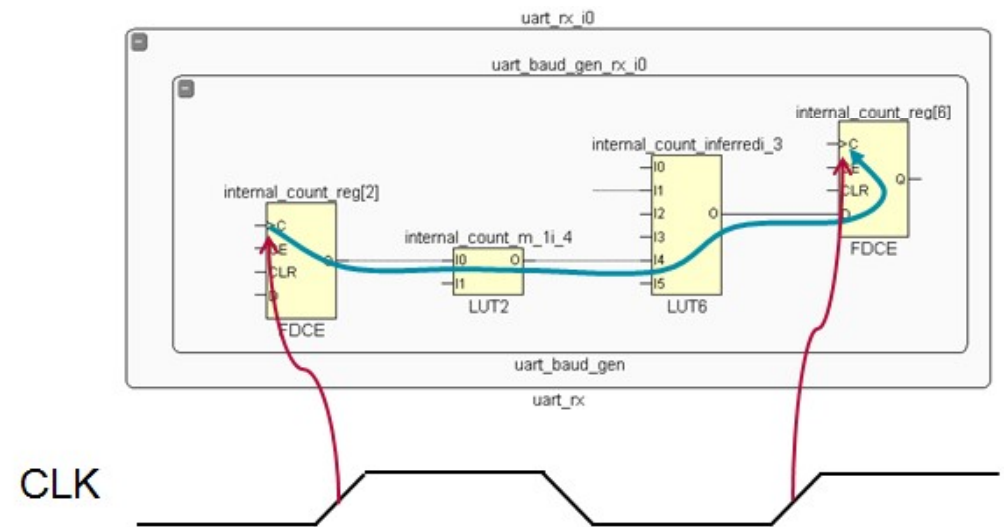
## Camino de señal utilizados para el análisis STA

- **Un camino de señal es un camino que:**
  - Comienza en un elemento sincronizado
  - Se propaga a través de elementos combinatoriales y las redes que los interconectan
  - Finaliza en un elemento sincronizado
- **Los elementos sincronizados son los flip-flops, block RAMs, bloque DSP, etc**
- **Los elementos combinatoriales son las LUTs, los MUXes, carry chains, etc.**



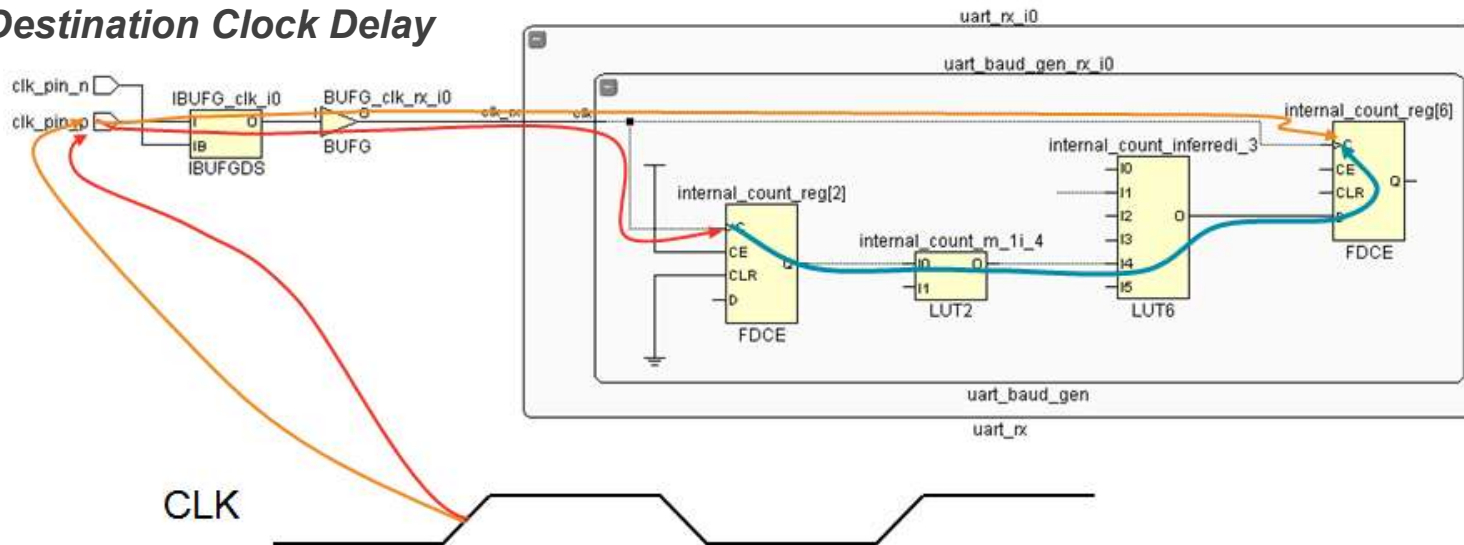
## Verificación del tiempo de Setup

- **Verifica que un cambio en un elemento sincronizado tiene tiempo de propagarse hasta los otros elementos sincronizados antes del siguiente cambio de reloj**
  - Este tiempo se verifica desde el flanco ascendente de la señal de reloj hasta el siguiente flanco ascendente
  - Se verifica para cada camino de señal



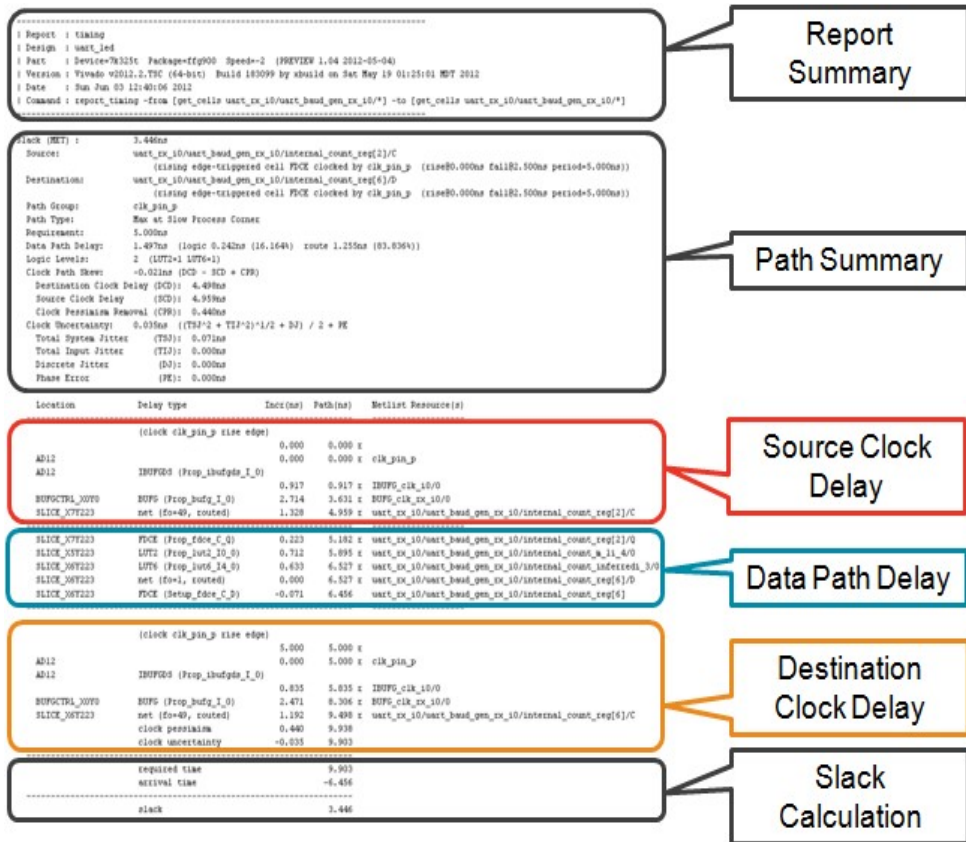
# Verificacion del tiempo de Hold

- Verifica que un cambio en un elemento sincronizado causado por un cambio de reloj no se propague al siguiente elemento sincronizado antes de que el mismo cambio de reloj llegue al siguiente elemento sincronizado
  - Usualmente se calcula del flanco ascendente del reloj al siguiente flanco ascendente
  - Se verifica para todos los caminos de señal
- El menor retardo se usa para *Source Clock* y *Data Path Delay*, y el mayor retardo se usa para *Destination Clock Delay*





# Estructura del reporte Timing Report



# Secciones del Reporte

## ➤ Report summary

```
C:/xup/fpga_flow/2016_2_ZYNQ_labs/zed/lab4/lab4.runs/impl_1/wave_gen_top_timing_summary_routed.rpt
1 Copyright 1986-2016 Xilinx, Inc. All Rights Reserved.
2 -----
3 Tool Version : Vivado v.2016.2 (win64) Build 1577090 Thu Jun  2 16:32:40 MDT 2016
4 Date       : Wed Jul 20 09:36:50 2016
5 Host      : XSUNFURUSHG30 running 64-bit Service Pack 1 (build 7601)
6 Command   : report_timing_summary -warn_on_violation -max_paths 10 -file wave_gen_top_timing_summary_routed.rpt
7 Design    : wave_gen_top
8 Device    : 7z020-clg484
9 Speed File : -1 PRODUCTION 1.11 2014-09-11
10 -----
```

## ➤ Design timing summary

```
120 -----
121 | Design Timing Summary
122 | -----
123 -----
124
125      WNS(ns)      TNS(ns)  TNS Failing Endpoints  TNS Total Endpoints      WHS(ns)      THS(ns)
126 -----
127      0.810      0.000                0                105          0.175      0.000
128
129
130 All user specified timing constraints are met.
```

## ➤ Clock summary

```
133 -----
134 | Clock Summary
135 | -----
136 -----
137
138 Clock  Waveform(ns)      Period(ns)      Frequency(MHz)
139 -----
140 clk_pin {0.000 5.000}  10.000          100.000
141
```

# Secciones del Reporte

## ➤ Intra clock table

```
143 -----
144 | Intra Clock Table
145 | -----
146 -----
147
148 Clock          WNS(ns)      TNS(ns)  TNS Failing Endpoints  TNS Total Endpoints  WHS(ns)
149 -----
150 clk_pin        0.810       0.000   0                      105                  0.175
151
```

## ➤ Maximum delay path

```
187 Max Delay Paths
188 -----
189 Slack (MET) : 0.810ns (required time - arrival time)
190 Source: led_ctl_i0/led_o_reg[0]/C
191 (rising edge-triggered cell FDRE clocked by clk_pin {rise@0.000ns fall@5.000ns period=10.000ns})
192 Destination: led_pins[0]
193 (output port clocked by clk_pin {rise@0.000ns fall@5.000ns period=10.000ns})
194 Path Group: clk_pin
195 Path Type: Max at Slow Process Corner
196 Requirement: 10.000ns (clk_pin rise@10.000ns - clk_pin rise@0.000ns)
197 Data Path Delay: 3.442ns (logic 3.441ns (99.971%) route 0.001ns (0.029%))
198 Logic Levels: 1 (OBUF=1)
199 Output Delay: 0.000ns
200 Clock Path Skew: -5.713ns (DCD - SCD + CPR)
201 Destination Clock Delay (DCD): 0.000ns = ( 10.000 - 10.000 )
202 Source Clock Delay (SCD): 5.713ns
203 Clock Pessimism Removal (CPR): 0.000ns
204 Clock Uncertainty: 0.035ns ((ISJ^2 + TIJ^2)^1/2 + DJ) / 2 + PE
205 Total System Jitter (TSJ): 0.071ns
206 Total Input Jitter (TIJ): 0.000ns
207 Discrete Jitter (DJ): 0.000ns
208 Phase Error (PE): 0.000ns
209
210 Location          Delay type          Incr(ns)  Path(ns)  Netlist Resource(s)
211 -----
212 (clock clk_pin rise edge)  0.000  0.000 r
213 R4                 net (fo=0)         0.000  0.000 r  clk_pin (IN)
214 R4                 IBUF (Prop_ibuf_I_O)  1.475  1.475 r  clk_pin_IBUF_inst/O
215 R4                 net (fo=1, routed)  2.114  3.589 r  clk_pin_IBUF
216
```

# Secciones del Reporte

## ➤ Delay path

210	Location	Delay type	Incr(ns)	Path(ns)	Netlist Resource(s)
211	-----				
212		(clock clk_pin rise edge)	0.000	0.000	r
213	R4		0.000	0.000	r clk_pin (IN)
214		net (fo=0)	0.000	0.000	clk_pin
215	R4	IBUF (Prop_ibuf_I_O)	1.475	1.475	r clk_pin_IBUF_inst/O
216		net (fo=1, routed)	2.114	3.589	clk_pin_IBUF
217	BUFGCTRL_X0Y0	BUFG (Prop_bufg_I_O)	0.096	3.685	r clk_pin_IBUF_BUFG_inst/O
218		net (fo=48, routed)	2.029	5.713	led_ctl_i0/CLK
219	OLOGIC_X0Y90	FDRE			r led_ctl_i0/led_o_reg[0]/C
220	-----				
221	OLOGIC_X0Y90	FDRE (Prop_fdre_C_Q)	0.472	6.185	r led_ctl_i0/led_o_reg[0]/Q
222		net (fo=1, routed)	0.001	6.186	led_pins_OBUF[0]
223	Y13	OBUF (Prop_obuf_I_O)	2.969	9.155	r led_pins_OBUF[0]_inst/O
224		net (fo=0)	0.000	9.155	led_pins[0]
225	Y13				r led_pins[0] (OUT)
226	-----				
227					
228		(clock clk_pin rise edge)	10.000	10.000	r
229		clock pessimism	0.000	10.000	
230		clock uncertainty	-0.035	9.965	
231		output delay	-0.000	9.965	
232	-----				
233		required time		9.965	
234		arrival time		-9.155	
235	-----				
236		slack		0.810	

# Temario

- Proceso de Implementacion
- Reportes
- Analisis estático de temporizacion
- **Generacion del archivo de configuracion y configuración de la FPGA**
- Resumen

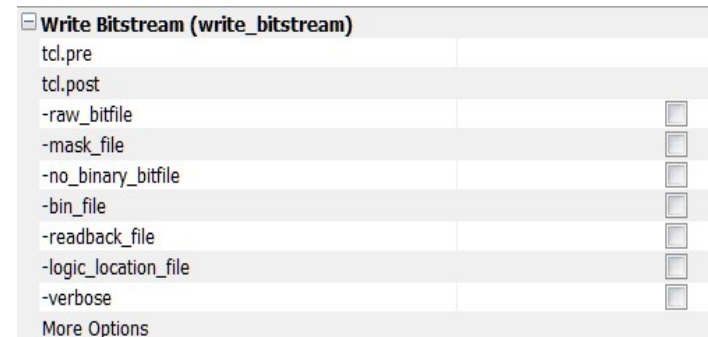
## Generacion del archivo de configuracion (Bitstream)

- **Genera el archivo de configuracion para el dispositivo seleccionado para el proyecto actual**
- **Se ejecuta sobre un sistema implementado**
- **Se ejecuta nuevamente el proceso de implementacion si hubo modificaciones**
- **Desarrollo basado en proyectos o desarrollos independientes**
  - IDE: Menu *Generate bitstream*
  - Comando Tcl: `launch_runs impl_1 -to_step write_bitstream`



# Configuración del proceso de generación del archivo de configuración

- Por defecto se usa el formato binario
- **-raw\_bitfile**: el archivo de configuración se genera en formato ascii y extensión (.rbt).
- **-mask\_file**: Generación de un archivo de máscara para verificación de un archivo de configuración cargado
- **-no\_binary\_bitfile**: No crea el archivo de configuración (.bit)
  - Se usa esta opción para generar un archivo de configuración ASCII o un archivo de máscara sin generar el archivo de configuración binario
- **-bin\_file**: Crea un archivo de configuración binario (.bin) solo con la información de configuración
- **-logic\_location\_file**: Genera un archivo de extensión (.ll) con la ubicación de las LUTs, BRAM, flip-flops, latches, I/O block, entradas y salidas



# Administrador de Hardware (Hardware Manager)

- **Los pasos para conectarse al hardware y configurar la FPGA son:**
  - Abrir el administrador de hardware
  - Abrir un dispositivo de hardware, administrado por un servidor de hardware. Esto puede ser en forma local o remota
  - Asociar el archivo de configuración al dispositivo FPGA
  - Configurar el dispositivo.
  - Opcionalmente, abrir la vista *hardware analyzer* para depuración





# Temario

- Proceso de Implementacion
- Reportes
- Analisis estático de temporizacion
- Generacion del archivo de configuracion y configuración de la FPGA
- **Resumen**

## Resumen

- La implementación utiliza los subprocesos *opt\_design*, *power\_opt*, *place\_design*, *phys\_opt\_design*, y *route\_design*
- Hay distintos tipos de reportes que permiten analizar el sistema que se está implementando
- Los caminos de señal utilizados en el análisis STA comienzan en un elemento sincronizado y terminan en otro elemento sincronizado
- Los caminos de señal se analizan para verificar los tiempos de setup y hold. Este análisis incluye la verificación de los tiempos de propagación de la señal de reloj
- El reporte *report\_timing\_summary* se utiliza para verificar el cumplimiento de las restricciones de temporización del sistema implementado
- El reporte *report\_timing* se usa para un análisis detallado de la temporización luego de los procesos de síntesis y de implementación