Implementation and STA

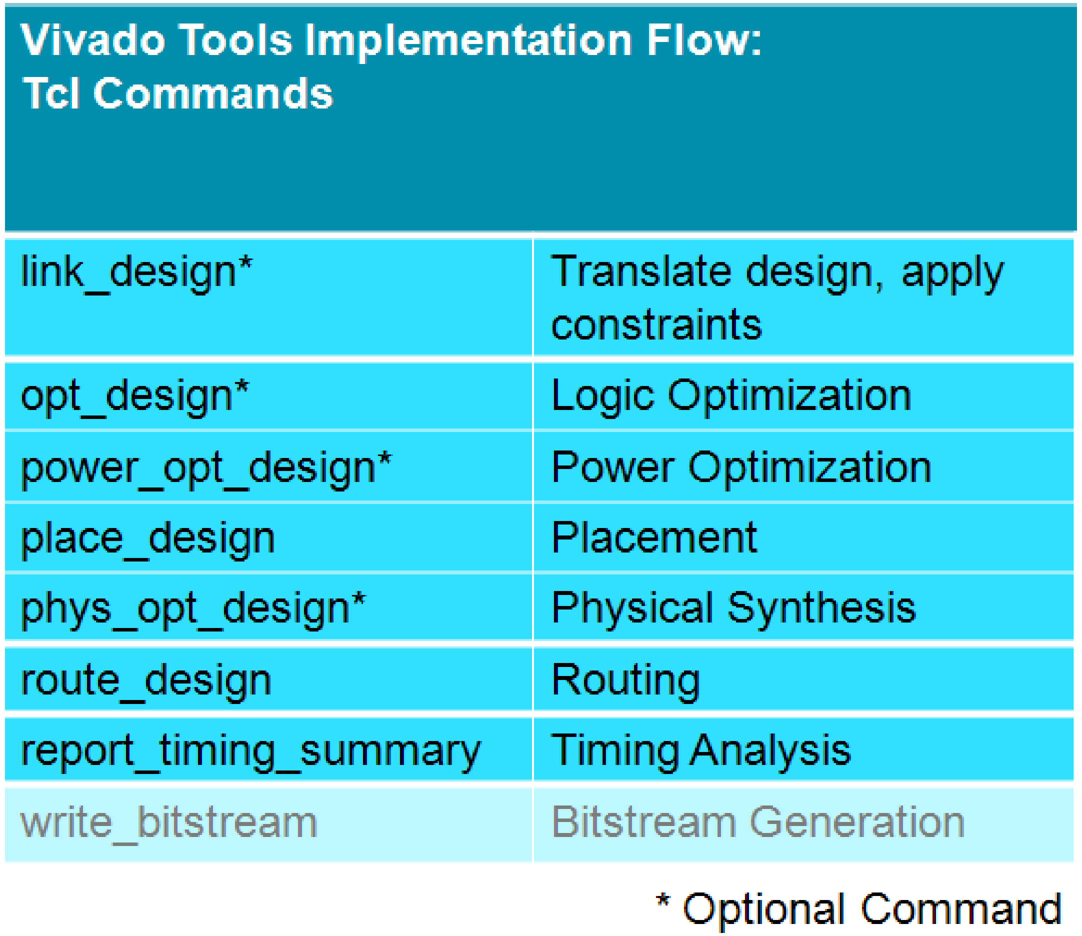
Objectives

- **After completing this module, you will be able to:**
  - Implement a design to completion
  - Generate useful implementation reports
  - Describe static timing analysis and static timing paths
  - Describe setup and hold checks
  - Understand the relationship between clocks and setup and hold checks
  - Generate a custom timing report to perform basic static timing analysis
  - Use the timing summary report to verify your timing constraints were met
  - Generate a bitstream and verify the functionality in hardware

Outline

- *Implementation*
- Reports
- Basic Static Timing Analysis
- Bitstream Generation and Verification in Hardware
- Summary

Vivado Implementation Phases

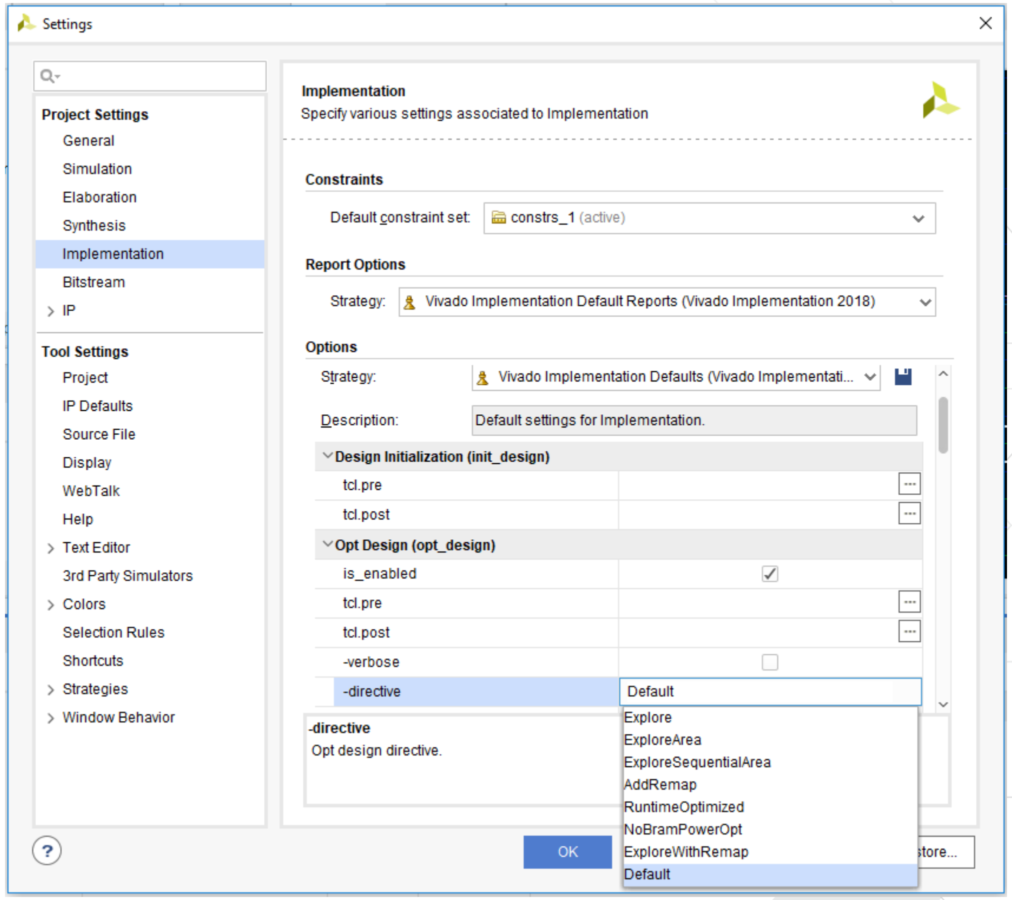


Vivado Tools Implementation Flow: Tcl Commands	
link_design*	Translate design, apply constraints
opt_design*	Logic Optimization
power_opt_design*	Power Optimization
place_design	Placement
phys_opt_design*	Physical Synthesis
route_design	Routing
report_timing_summary	Timing Analysis
write_bitstream	Bitstream Generation

* Optional Command

opt_design: Logic Optimization

- **Ensures optimal netlist for placement**
  - Further logic optimization on fully-assembled netlist built from synthesized RTL, IP blocks
    - Performs logic trimming on incoming netlist
    - Constant propagation – remove unnecessary static logic
    - LUT equation remapping
- **Optional in non-project batch flow (but recommended)**
  - Example: needed to trim unused bank cells in MIG IP (phaser/iodelay/….)
- **Automatically enabled in the project-based flow**



Implementation

Specify various settings associated to Implementation

Constraints

Default constraint set: constrs_1 (active)

Report Options

Strategy: Vivado Implementation Default Reports (Vivado Implementation 2018)

Options

Strategy: Vivado Implementation Defaults (Vivado Implementati...

Description: Default settings for Implementation.

Design Initialization (init_design)

tcl.pre

tcl.post

Opt Design (opt_design)

is_enabled ☑

tcl.pre

tcl.post

-verbose □

-directive: Default

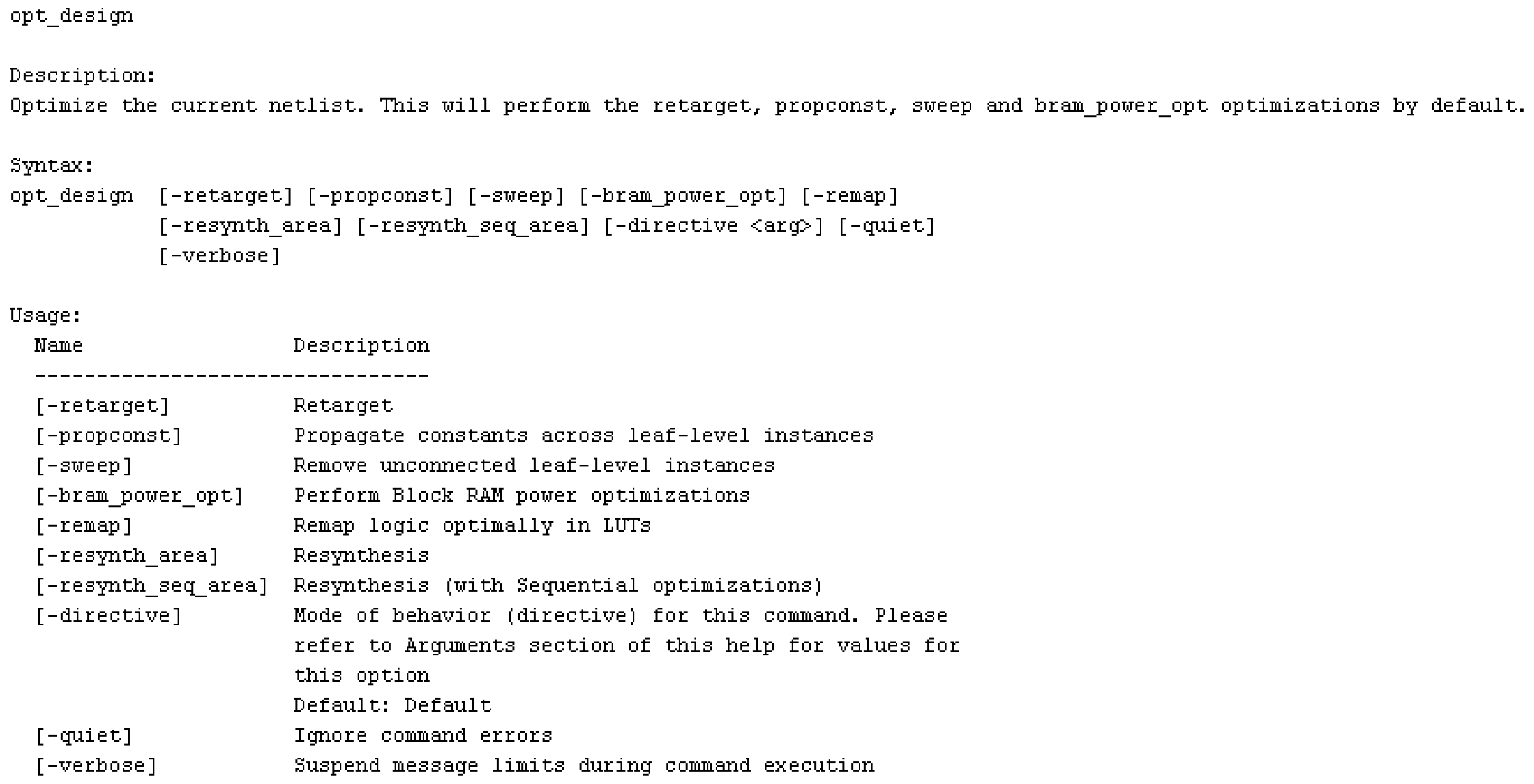
Explore
ExploreArea
ExploreSequentialArea
AddRemap
RuntimeOptimized
NoBramPowerOpt
ExploreWithRemap
Default

-directive

Opt design directive.

opt_design Options

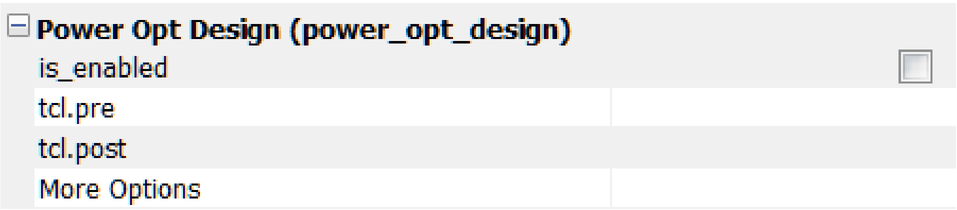
> **Non-project batch flow**
>> Can specify which optimizations to perform from a script



```
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
```

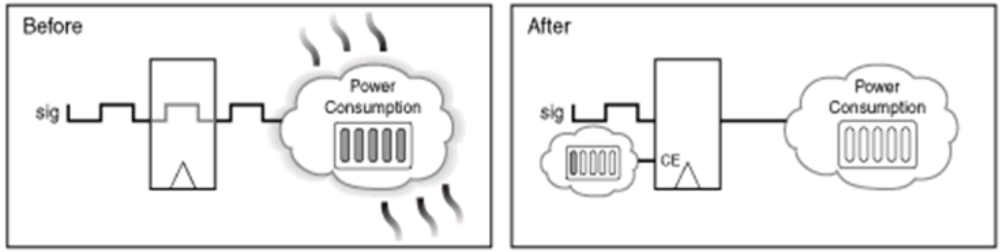
power_opt: Power Optimization

- Power optimization includes a fine-grained clock gating solution that can reduce dynamic power by up to 30%
- Intelligent clock gating optimizations are automatically performed on the entire design and will generate no changes to the existing logic or clocks
- Algorithm performs analysis on all portions of the design
  - Legacy and third-party IP blocks



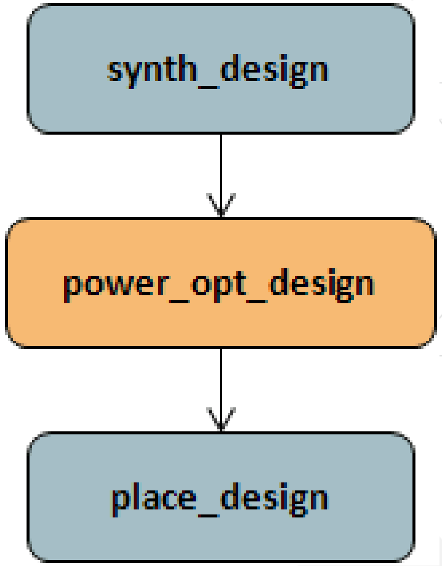
⊟ **Power Opt Design (power_opt_design)**

is_enabled	□
tcl.pre	
tcl.post	
More Options	



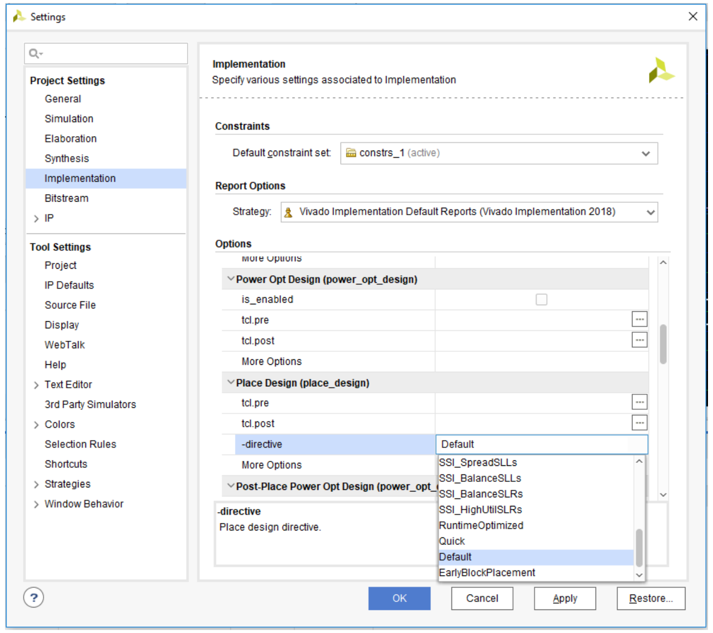
Power Optimization Commands

- **Automatic power reduction**
  - Automatically turns off unused portions of the design
  - Does not require deep system level knowledge
- **Vivado IDE provides optimization control at global and object level**
  - Global command for optimizing the design: `power_opt_design`
  - Local level control through SDC command: `set_power_opt`
    - Instance level: Include/exclude instances for power opt
    - Clock domain: Optimize instances clocked by the specified clock
    - Cell-type level: Block RAM, registers, SRL



place_design: placer

- **Project-based flow**
  - Included in implementation stage
- **Non-project batch flow**
  - `place_design`
- **Can use an input XDEF as a starting point for placement**



Implementation

Specify various settings associated to Implementation

Constraints

Default constraint set: constrs_1 (active)

Report Options

Strategy: Vivado Implementation Default Reports (Vivado Implementation 2018)

Options

More Options

Power Opt Design (power_opt_design)

is_enabled ☐

tcl.pre

tcl.post

More Options

Place Design (place_design)

tcl.pre

tcl.post

-directive: Default

SSI_SpreadSLLs
SSI_BalanceSLLs
SSI_BalanceSLRs
SSI_HighUtilSLRs
RuntimeOptimized
Quick
Default
EarlyBlockPlacement

More Options

Post-Place Power Opt Design (power_opt_

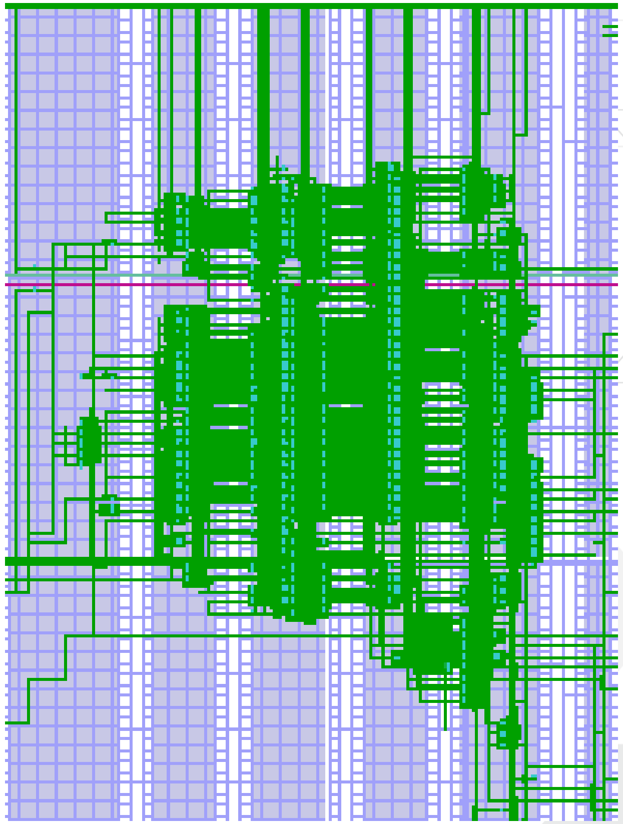
-directive

Place design directive.

```
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
```

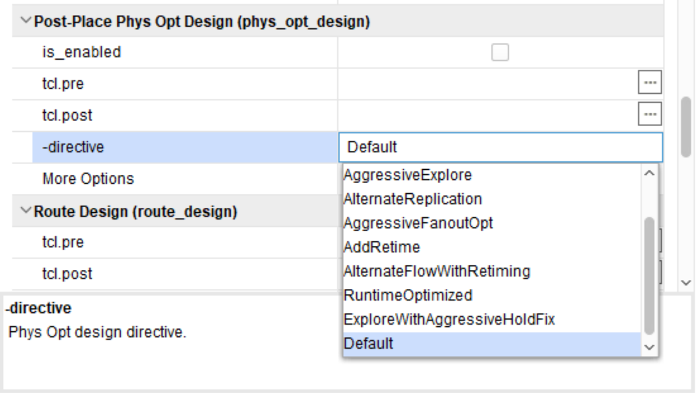
Placement

- Phases of a complete placement
  - Pre-placement DRC
    - Check for unroutable connections, valid physical constraints, overutilization
  - Placement
    - I/O and clock placement
- Macro and primitive placement
  - Timing-driven and wire length-driven
  - Congestion-aware
- Detailed placement
  - Refine locations of small "shapes," flip-flops, LUTs
  - Commit to location sites – pack into slices
- Post-commit optimizations



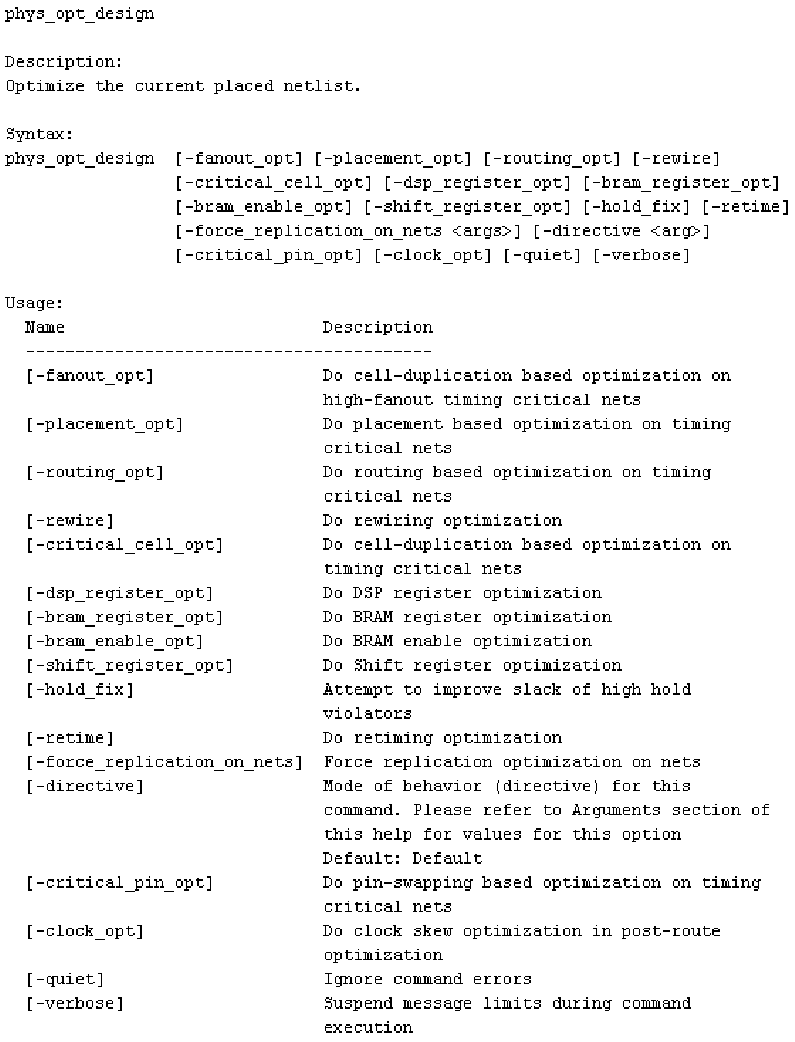
phys_opt_design: Physical Synthesis

- **Post-placement timing-driven optimization**
  - Replicates and places drivers of high fanout nets with negative slack
    - Replication only performed if it improves timing
    - Slack must be within critical range
      - Approximately 10% of worst negative slack (WNS)
- **Available in all flows and can be de-activated in the GUI**
  - `phys_opt_design`
  - Run between `place_design` and `route_design`



Post-Place Phys Opt Design (phys_opt_design)	
is_enabled	□
tcl.pre	
tcl.post	
-directive	Default
More Options	
Route Design (route_design)	
tcl.pre	
tcl.post	

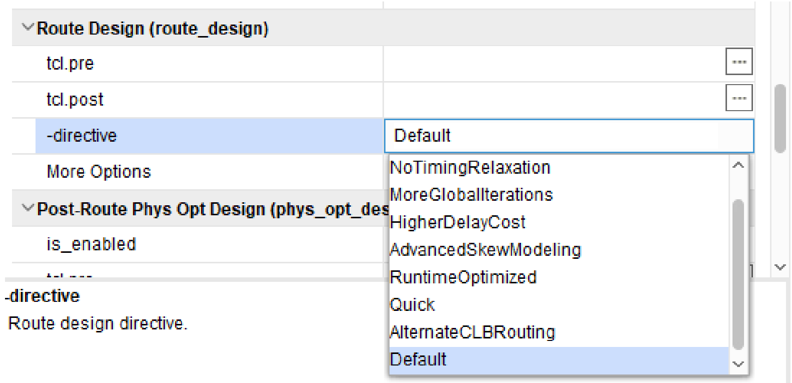
-directive
Phys Opt design directive.



```
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
```

route_design: Router

- > **Project-based flow**
  - >> Included in implementation stage
- > **Non-project batch flow**
  - >> `route_design`
- > **Router reporting**
  - >> `report_route_status` command
  - >> Check route status of individual nets
    - – Fully routed: lists routing resources
    - – Failed routes



Route Design (route_design)

tcl.pre

tcl.post

-directive: Default

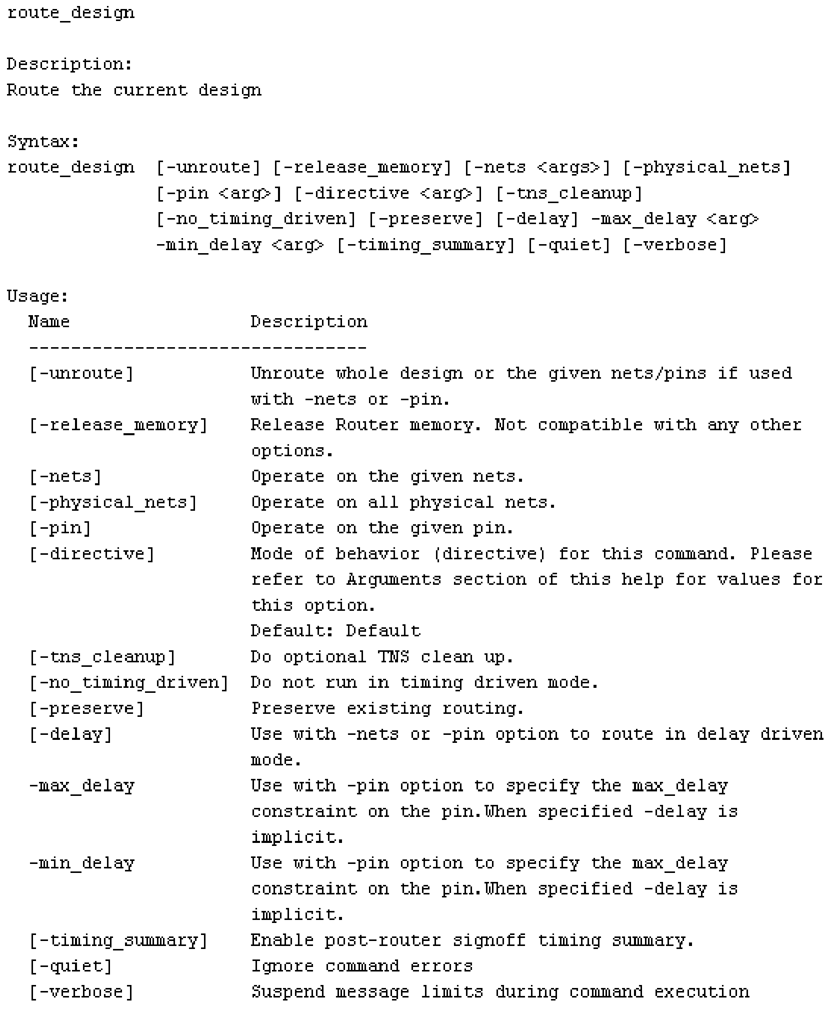
More Options

Post-Route Phys Opt Design (phys_opt_des

is_enabled

-directive

Route design directive.



```
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
```

Router

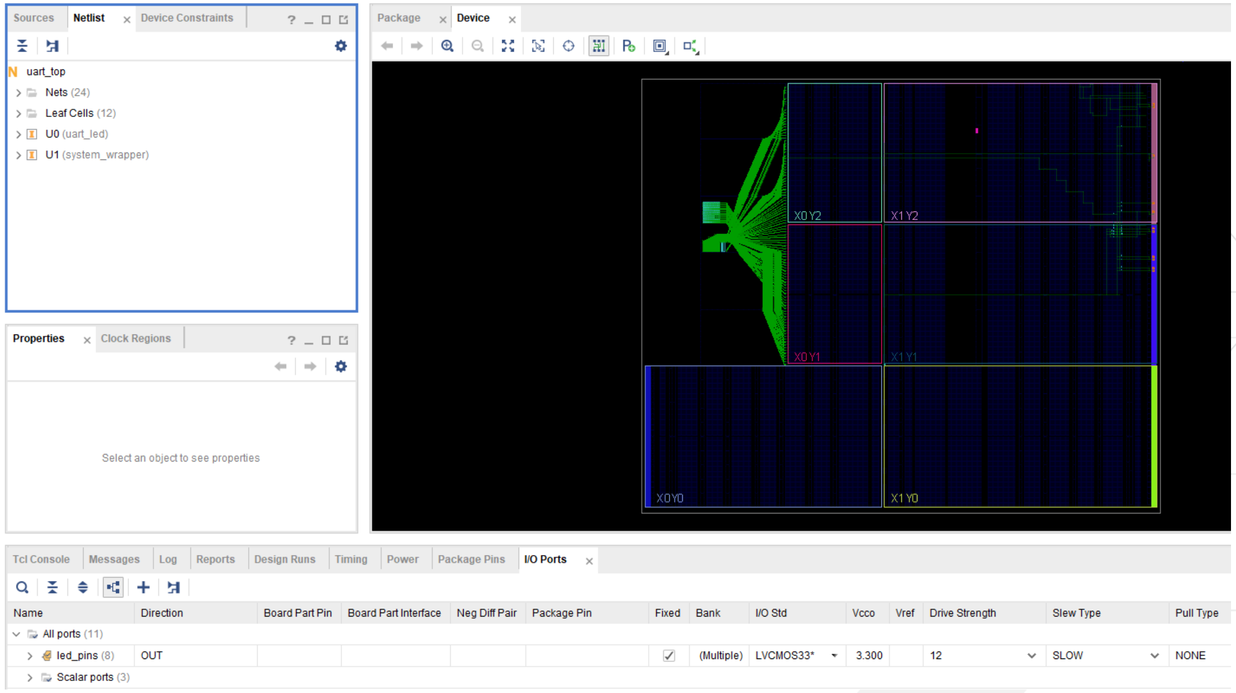
- **Phases of a complete route**
  - Special net and clock routing
  - Timing-driven routing
    - Prioritized by setup/hold path criticality
    - Swap LUT inputs pin to improve critical paths
    - Fix reasonable hold time violations
- **Two modes**
  - Normal (default): Router starts with a placed design and attempts to route all nets
  - Re-Entrant (non-project batch only): Router can route/unroute as well as lock/unlock specific nets

Reports

XILINX

After Implementation

- **Sources and Netlist tabs do not change**
  - As each resources is selected, it will show the exact placement of the resource on the die
- **Timing results have to be generated with the Report Timing Summary**
- **As each path is selected, the placement of the logic and its connections is shown in the Device view**
  - This is the cross-probing feature that helps with static timing analysis



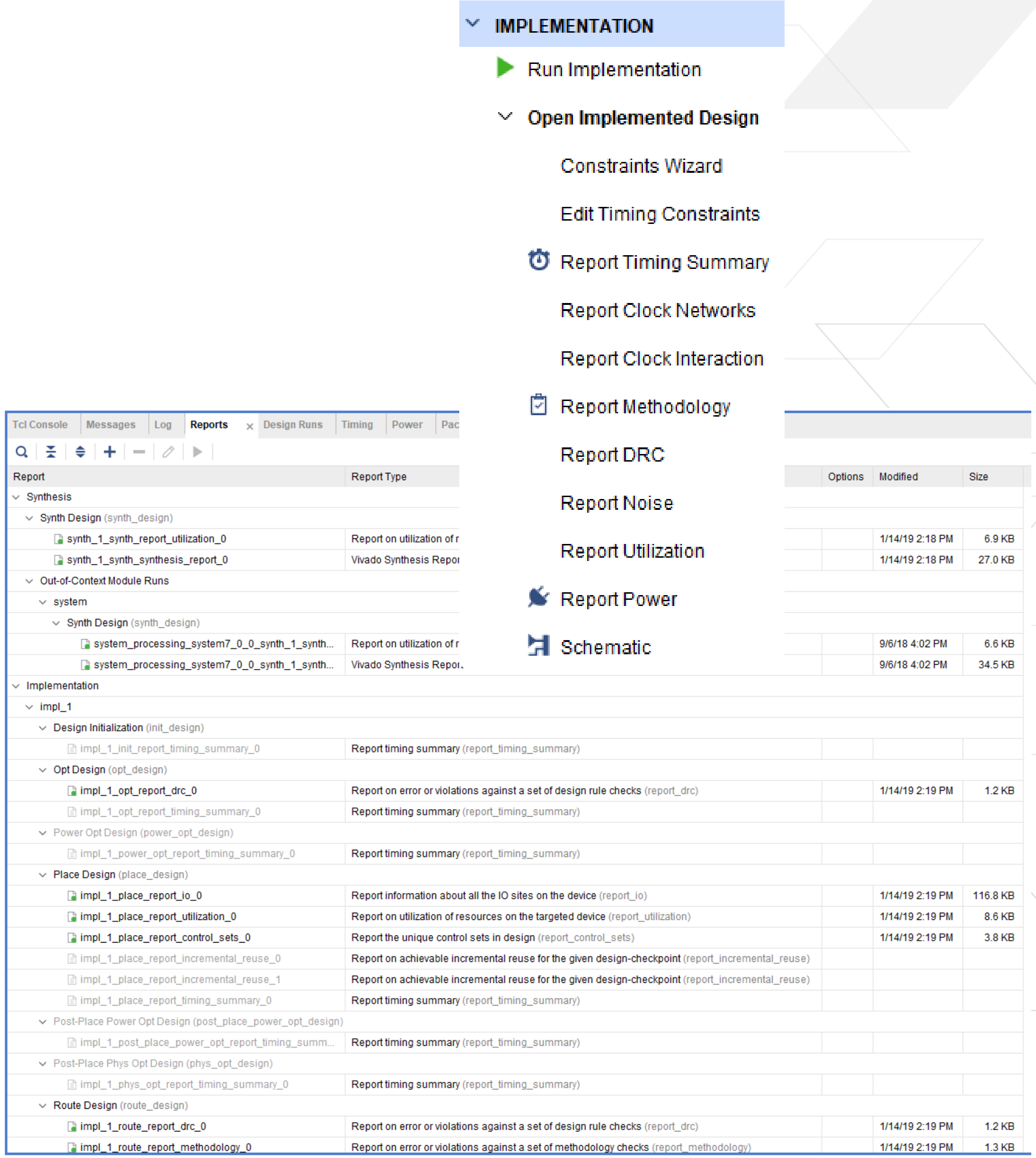
uart_top
- Nets (24)
- Leaf Cells (12)
- U0 (uart_led)
- U1 (system_wrapper)

Select an object to see properties

Name	Direction	Board Part Pin	Board Part Interface	Neg Diff Pair	Package Pin	Fixed	Bank	I/O Std	Vcco	Vref	Drive Strength	Slew Type	Pull Type
All ports (11)													
led_pins (8)	OUT					✓	(Multiple)	LVCMOS33*	3.300		12	SLOW	NONE
Scalar ports (3)													

Implementation Reports

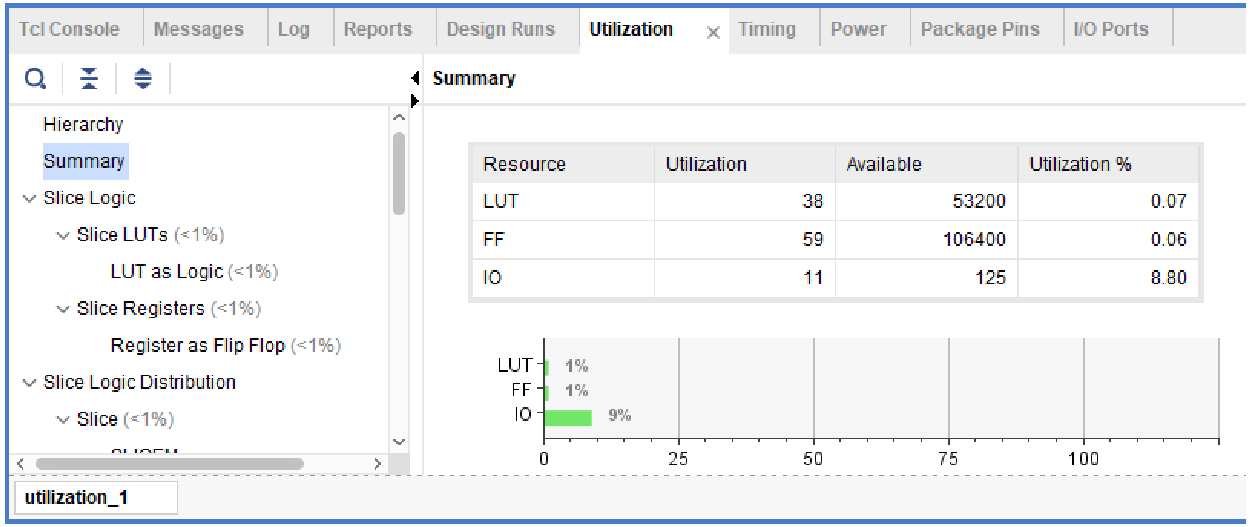
- While the Flow Navigator points to the most important reports, the Reports tab contains several other useful reports
  - Post Optimization DRC Report: Lists the I/O DRC checks that were completed
  - Post Power Optimization DRC Report: Lists the power DRC checks that were completed
  - Place and Route Log: Describes the implementation process and any issues it encountered
  - IO Report: Lists the final pinout for your design
  - Clock Utilization Report: Describes the clock resources used and the clock utilization resource on a region-by-region basis
  - Utilization Report: Describes the amount of FPGA resources used in a text format
  - Control Sets Report: describes how your control signals were grouped



Report	Report Type	Options	Modified	Size
Synthesis				
Synth Design (synth_design)				
synth_1_synth_report_utilization_0	Report on utilization of r		1/14/19 2:18 PM	6.9 KB
synth_1_synth_synthesis_report_0	Vivado Synthesis Repor		1/14/19 2:18 PM	27.0 KB
Out-of-Context Module Runs				
system				
Synth Design (synth_design)				
system_processing_system7_0_0_synth_1_synth...	Report on utilization of r		9/6/18 4:02 PM	6.6 KB
system_processing_system7_0_0_synth_1_synth...	Vivado Synthesis Repor.		9/6/18 4:02 PM	34.5 KB
Implementation				
impl_1				
Design Initialization (init_design)				
impl_1_init_report_timing_summary_0	Report timing summary (report_timing_summary)			
Opt Design (opt_design)				
impl_1_opt_report_drc_0	Report on error or violations against a set of design rule checks (report_drc)		1/14/19 2:19 PM	1.2 KB
impl_1_opt_report_timing_summary_0	Report timing summary (report_timing_summary)			
Power Opt Design (power_opt_design)				
impl_1_power_opt_report_timing_summary_0	Report timing summary (report_timing_summary)			
Place Design (place_design)				
impl_1_place_report_io_0	Report information about all the IO sites on the device (report_io)		1/14/19 2:19 PM	116.8 KB
impl_1_place_report_utilization_0	Report on utilization of resources on the targeted device (report_utilization)		1/14/19 2:19 PM	8.6 KB
impl_1_place_report_control_sets_0	Report the unique control sets in design (report_control_sets)		1/14/19 2:19 PM	3.8 KB
impl_1_place_report_incremental_reuse_0	Report on achievable incremental reuse for the given design-checkpoint (report_incremental_reuse)			
impl_1_place_report_incremental_reuse_1	Report on achievable incremental reuse for the given design-checkpoint (report_incremental_reuse)			
impl_1_place_report_timing_summary_0	Report timing summary (report_timing_summary)			
Post-Place Power Opt Design (post_place_power_opt_design)				
impl_1_post_place_power_opt_report_timing_summ...	Report timing summary (report_timing_summary)			
Post-Place Phys Opt Design (phys_opt_design)				
impl_1_phys_opt_report_timing_summary_0	Report timing summary (report_timing_summary)			
Route Design (route_design)				
impl_1_route_report_drc_0	Report on error or violations against a set of design rule checks (report_drc)		1/14/19 2:19 PM	1.2 KB
impl_1_route_report_methodology_0	Report on error or violations against a set of methodology checks (report_methodology)		1/14/19 2:19 PM	1.3 KB

Utilization Reports

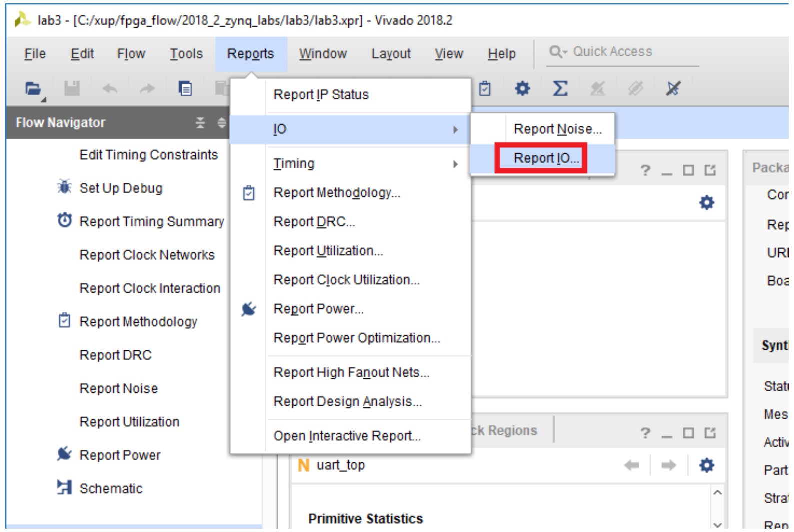
- Double-click in the Reports tab to view in text form
- Click on Report Utilization under Implementation Result in the Flow Navigator pane to view in chart and table format

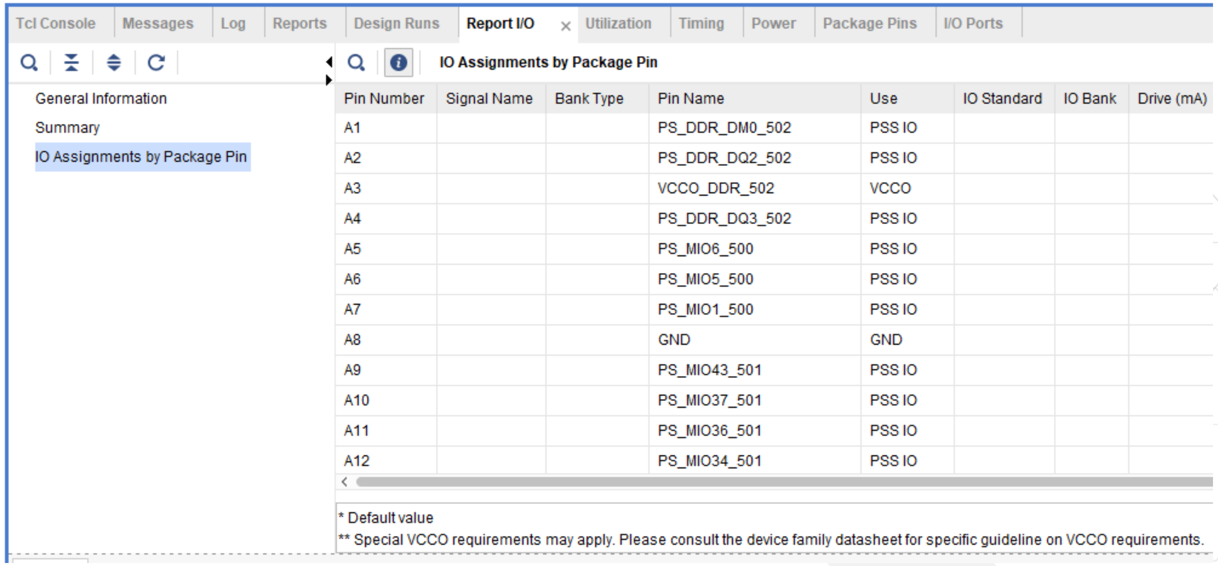


Resource	Utilization	Available	Utilization %
LUT	38	53200	0.07
FF	59	106400	0.06
IO	11	125	8.80

I/O Report

- This report provides a table that lists every signal, its attributes, and its final location
  - It is always important to double-check pin assignments before implementing because the tools can move any pin that is unassigned





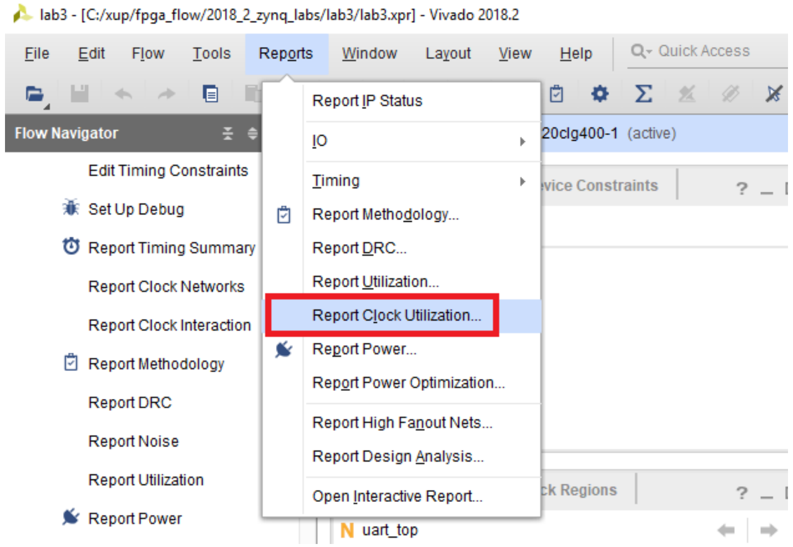
IO Assignments by Package Pin

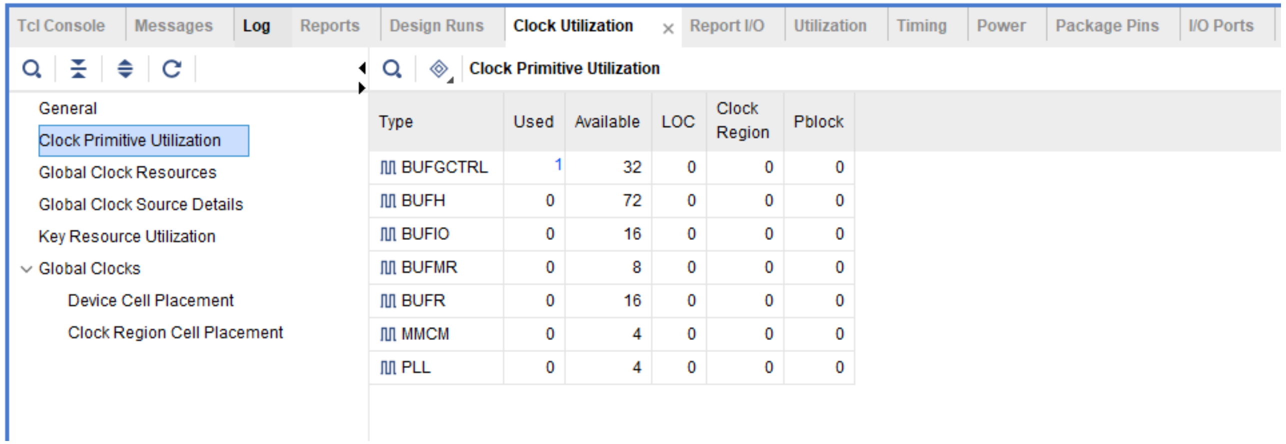
Pin Number	Signal Name	Bank Type	Pin Name	Use	IO Standard	IO Bank	Drive (mA)
A1			PS_DDR_DM0_502	PSS IO			
A2			PS_DDR_DQ2_502	PSS IO			
A3			VCCO_DDR_502	VCCO			
A4			PS_DDR_DQ3_502	PSS IO			
A5			PS_MIO6_500	PSS IO			
A6			PS_MIO5_500	PSS IO			
A7			PS_MIO1_500	PSS IO			
A8			GND	GND			
A9			PS_MIO43_501	PSS IO			
A10			PS_MIO37_501	PSS IO			
A11			PS_MIO36_501	PSS IO			
A12			PS_MIO34_501	PSS IO			

* Default value
** Special VCCO requirements may apply. Please consult the device family datasheet for specific guideline on VCCO requirements.

Clock Utilization Report

- This report describes the clocking resources used in the design
  - BUFG, BUFH, BUFHCE, MMCM, and a clock region analysis



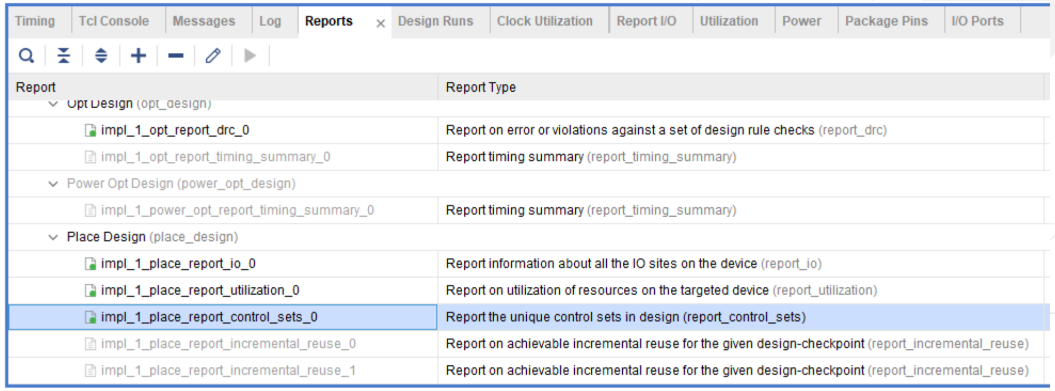


Clock Primitive Utilization

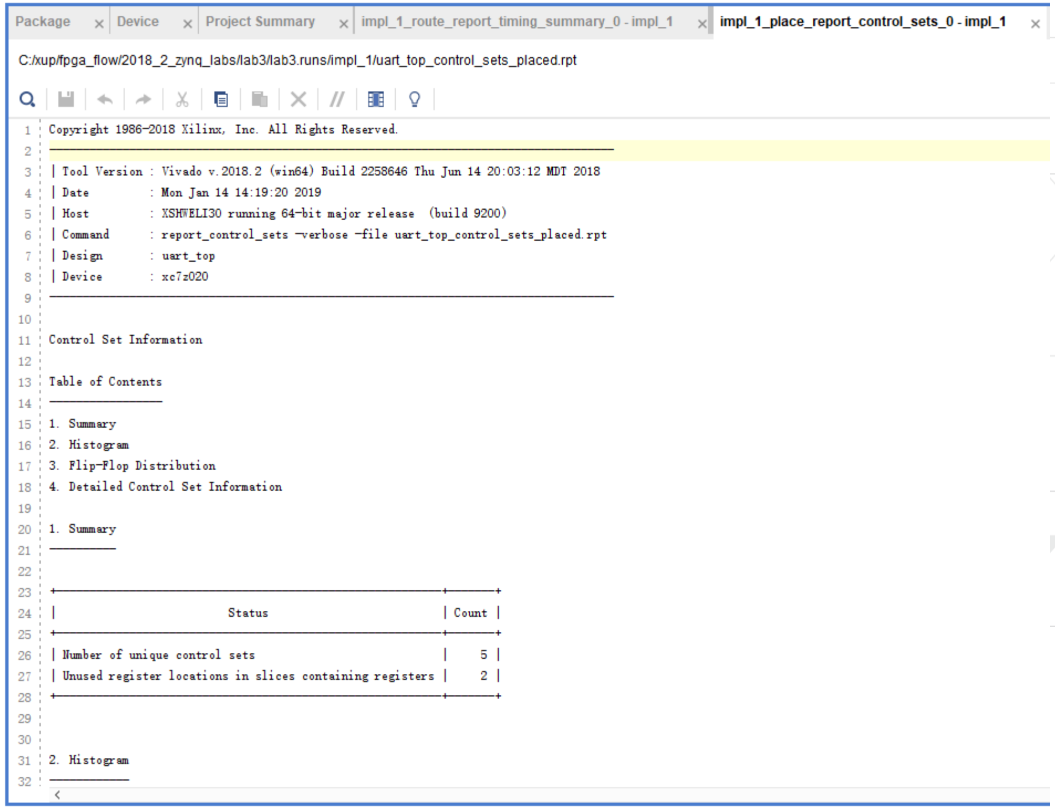
Type	Used	Available	LOC	Clock Region	Pblock
BUFGCTRL	1	32	0	0	0
BUFH	0	72	0	0	0
BUFIO	0	16	0	0	0
BUFMR	0	8	0	0	0
BUFR	0	16	0	0	0
MMCM	0	4	0	0	0
PLL	0	4	0	0	0

Control Sets Report

- This report describes the number of unique control sets in the design (ideally this will be as small as possible)
- Number of control sets describes how control signals were grouped
  - This determines the ability of the tools to reach high device utilization
  - Number of controls signals in the design is determined by the designer's inference of sets, resets, and clock enable signals
  - Number of control signals can be reduced if the designer attempts to share controls signals throughout the design as much as possible



Report	Report Type
Opt Design (opt_design)	
impl_1_opt_report_drc_0	Report on error or violations against a set of design rule checks (report_drc)
impl_1_opt_report_timing_summary_0	Report timing summary (report_timing_summary)
Power Opt Design (power_opt_design)	
impl_1_power_opt_report_timing_summary_0	Report timing summary (report_timing_summary)
Place Design (place_design)	
impl_1_place_report_io_0	Report information about all the IO sites on the device (report_io)
impl_1_place_report_utilization_0	Report on utilization of resources on the targeted device (report_utilization)
impl_1_place_report_control_sets_0	Report the unique control sets in design (report_control_sets)
impl_1_place_report_incremental_reuse_0	Report on achievable incremental reuse for the given design-checkpoint (report_incremental_reuse)
impl_1_place_report_incremental_reuse_1	Report on achievable incremental reuse for the given design-checkpoint (report_incremental_reuse)


Package
Device
Project Summary
impl_1_route_report_timing_summary_0 - impl_1
impl_1_place_report_control_sets_0 - impl_1

C:/xup/fpga_flow/2018_2_zynq_labs/lab3/lab3.runs/impl_1/uart_top_control_sets_placed.rpt

```
Copyright 1986-2018 Xilinx, Inc. All Rights Reserved.
--------------------------------------------------------------------------------
| Tool Version : Vivado v.2018.2 (win64) Build 2258646 Thu Jun 14 20:03:12 MDT 2018
| Date         : Mon Jan 14 14:19:20 2019
| Host         : XSHWELI30 running 64-bit major release  (build 9200)
| Command      : report_control_sets -verbose -file uart_top_control_sets_placed.rpt
| Design       : uart_top
| Device       : xc7z020
--------------------------------------------------------------------------------

Control Set Information

Table of Contents
-----------------
1. Summary
2. Histogram
3. Flip-Flop Distribution
4. Detailed Control Set Information

1. Summary
----------


+----------------------------------------------------------+-------+
|                          Status                          | Count |
+----------------------------------------------------------+-------+
| Number of unique control sets                            |     5 |
| Unused register locations in slices containing registers |     2 |
+----------------------------------------------------------+-------+



2. Histogram
------------
```

power_opt: Report

- `report_power_opt` **command details the gating performed on the design by the power_opt program**
  - Can be run before power optimization to create a report detailing all the user-gated logic in the design for a before and after picture of your design

Basic Static Timing Analysis

XILINX

Static Timing Analysis (STA)

- **A design is an interconnected set of cells and nets**
- **The functionality of a design is determined by the RTL design sources**
  - The functionality can be verified by a simulation tool
- **The performance of a device is determined by the delays of the cells that comprise the design**
  - This is verified by static timing analysis
- **In STA the functionality of the components of the design are not important**
  - Only the performance of the components
  - Cells need only be classified as combinatorial or sequential

Component Delays

- **Each component takes time to perform its function**
  - A LUT has propagation delay from its inputs to its outputs
  - A net has propagation delay from the driver to the receiver(s)
  - A flip-flop requires stable data for a required time around its sample point
- **These delays are dependent on a several factors**
  - Some are determined by the composition of the FPGA and the implementation of the design
    - The physical characteristics of the element (how it is constructed)
    - The location of the object (where it is placed with respect to other objects)
  - Some are determined by environmental factors (PVT)
    - The process variation of the device
    - The voltage applied to the cells
    - The temperature of the cells

Delays

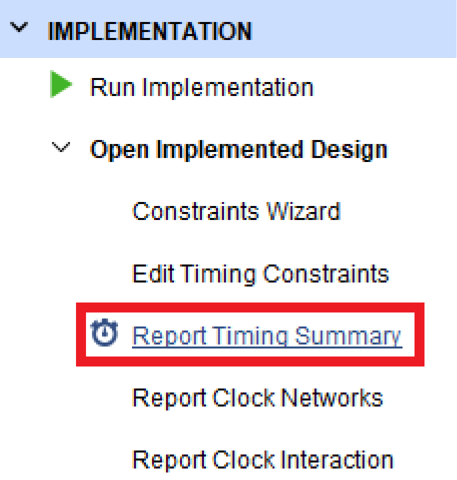
- **Component and net delays are provided by Xilinx, and are extracted by careful characterization of production devices**
  - Timing is extracted over the allowed operating range of the device
    - Process is within a specific range
      - Different ranges are used for different speed grades (-1, -2, -3)
    - Voltage is between the minimum and maximum allowed for the device
      - Different speed grades may allow different voltages (i.e. -1L)
    - Temperature is between the maximum and minimum specified
      - Commercial and industrial parts allow different temperature ranges
  - These characterized delays may be extracted at various process corners
  - Fastest PVT, slowest PVT
  - The characterized delays at the appropriate corner are used by the tools during STA

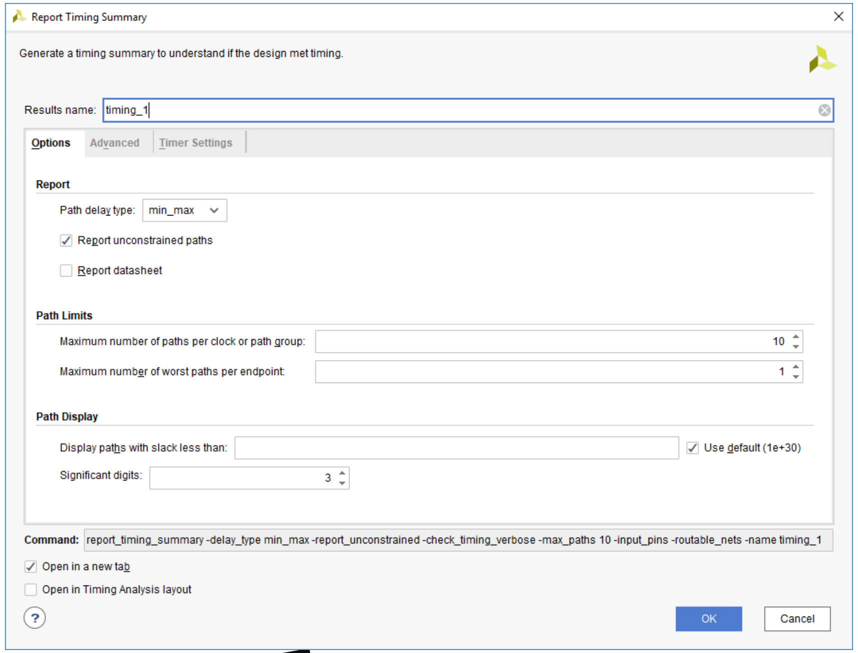
Why Do We Need STA?

- **Many of the processes in FPGA implementation are timing driven**
  - Synthesis for circuit construction
  - Placer for optimal cell locations
  - Router for choosing routing elements
- **Tools must have constraints to determine the desired performance goals**
- **STA is used during the processes, and afterwards for generating reports**
- **Ultimately, STA determines if a design will provide the desired performance**

report_timing_summary: Timing Report Summary

- Timing reports with true timing information can be generated from the Flow Navigator after implementation has been completed





Generate a timing summary to understand if the design met timing.

Results name: timing_1

Report

Path delay type: min_max

☑ Report unconstrained paths

□ Report datasheet

Path Limits

Maximum number of paths per clock or path group: 10

Maximum number of worst paths per endpoint: 1

Path Display

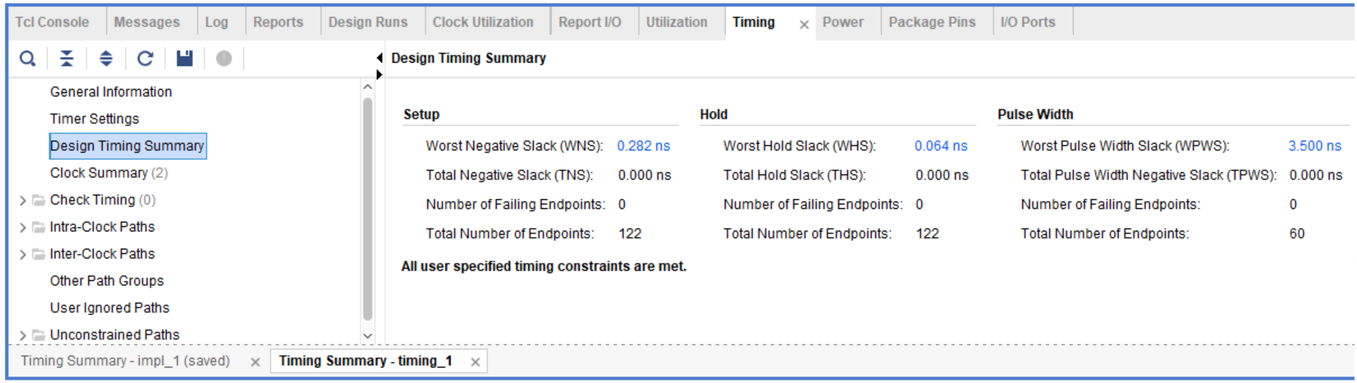
Display paths with slack less than: ☑ Use default (1e+30)

Significant digits: 3

Command: `report_timing_summary -delay_type min_max -report_unconstrained -check_timing_verbose -max_paths 10 -input_pins -routable_nets -name timing_1`

☑ Open in a new tab

□ Open in Timing Analysis layout



Design Timing Summary

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	0.282 ns	Worst Hold Slack (WHS):	0.064 ns	Worst Pulse Width Slack (WPWS):	3.500 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:	122	Total Number of Endpoints:	122	Total Number of Endpoints:	60

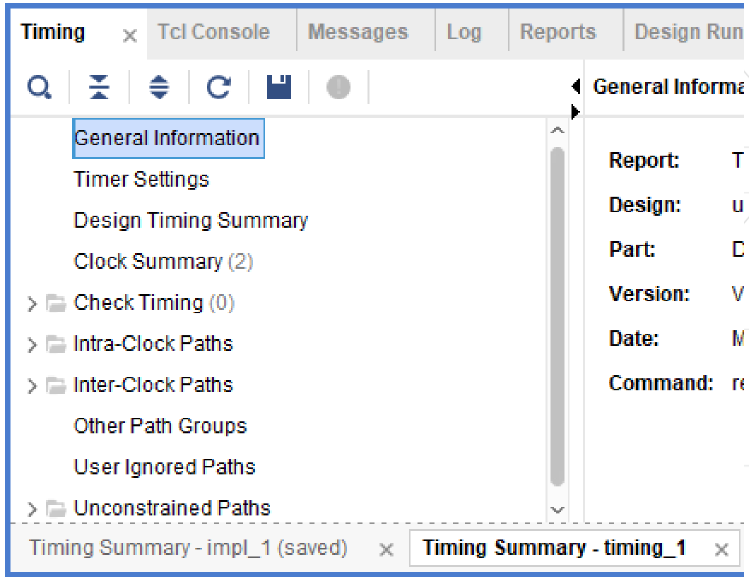
All user specified timing constraints are met.

Timing Summary

- **Summary reports Setup, Hold, and Pulse Width related results**
  - Setup
    - Worst Negative Slack (WNS): The worst slack of all the timing paths for max delay analysis. It can be positive or negative; positive means no violation
    - Total Negative Slack (TNS): The sum of all WNS violations, when considering only the worst violation of each timing path endpoint- 0ns when all timing constraints are met, Negative when there are some violations
    - Number of Failing Endpoints: The total number of endpoints with a violation (WNS<0ns)
  - Hold
    - Worst Hold Slack (WHS): Corresponds to the worst slack of all the timing paths for min delay analysis. It can be positive or negative
  - Pulse Width
    - Worst Pulse Width Slack (WPWS): Corresponds to the worst slack of all the timing checks listed above when using both min and max delays

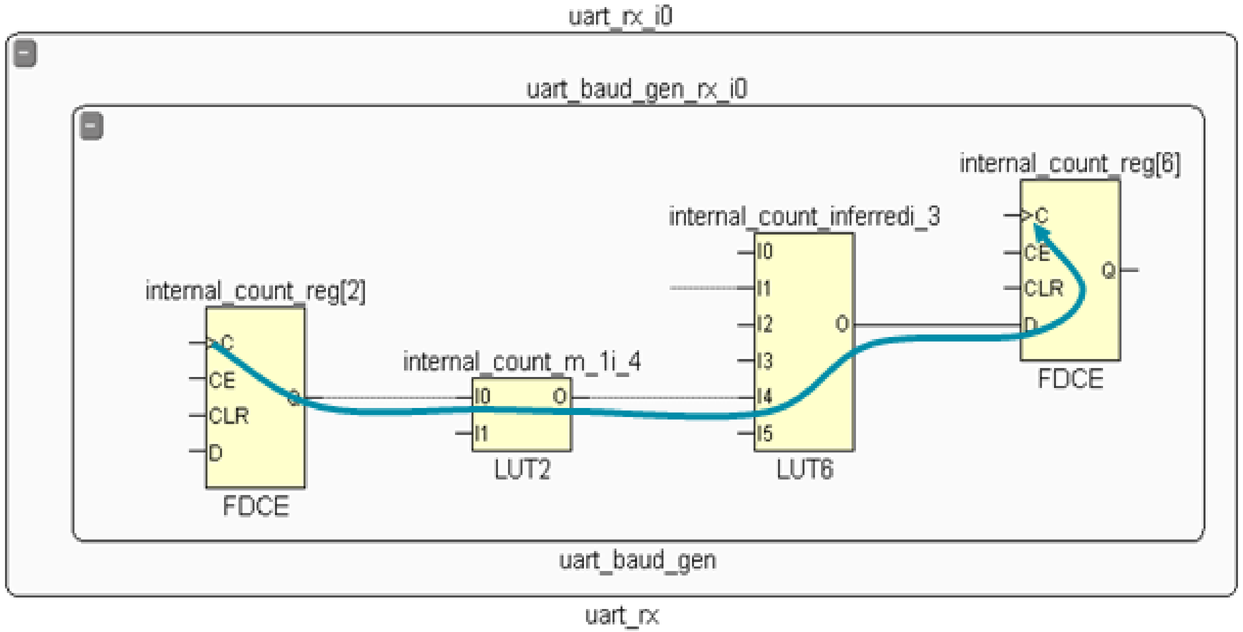
The Timing Summary Table

- **The Timing Summary table shows**
  - General Information provides design name, device, package, speed grades etc.
  - Timer Settings provides timing analysis engine settings
  - Design Timing Summary provides a summary of all the timing reports
  - Clock Summary includes information similar to that produced by report_clocks
  - Check Timing includes information about missing timing constraints or paths with constraints issues that need to be reviewed
  - Intra-Clock Paths includes summary of the worst slack and total violations of the timing paths with the same source and destination clocks
  - Inter-Clock Paths includes summary of the worst slack and total violations of timing paths with different source and destination clocks
  - Other Path Groups displays paths not covered above, including user-defined path groups
  - User Ignored Paths are paths that are ignored during timing analysis
  - Unconstrained Paths contain paths that were not covered by the XDC constraints
  - Violations are displayed in red



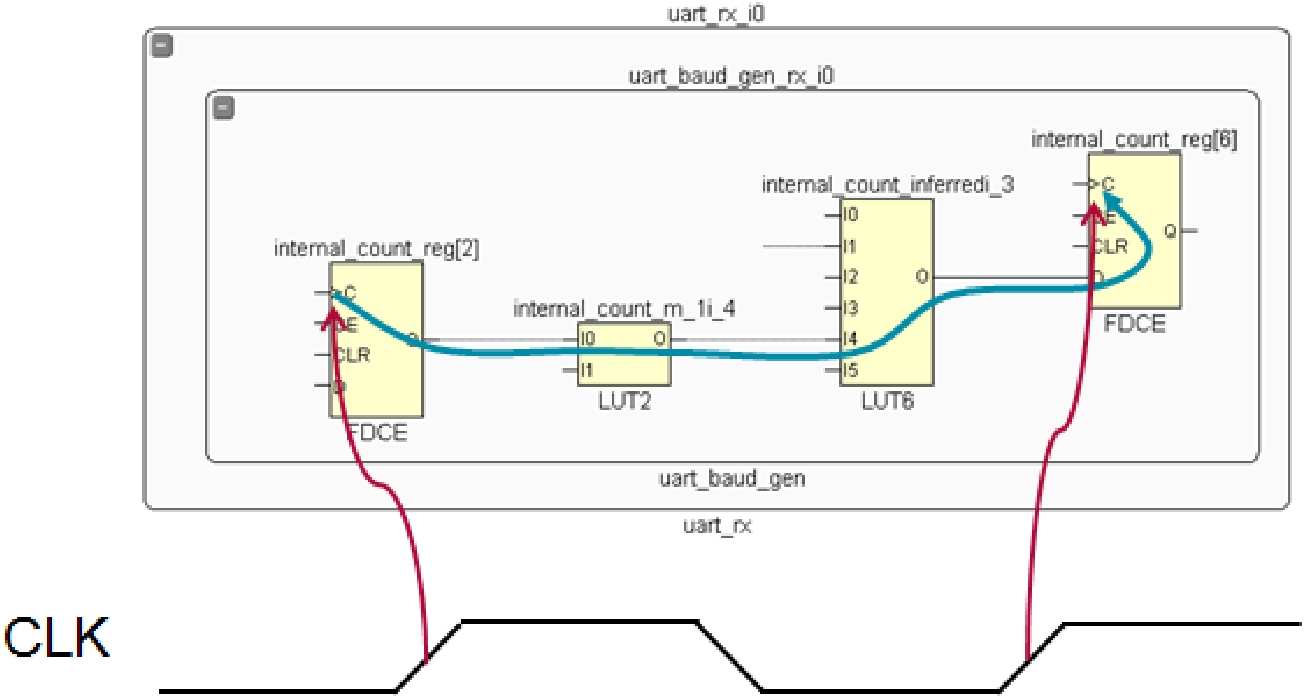
Static Timing Paths

- A static timing path is a path that
  - Starts at a clocked element
  - Propagates through any number of combinatorial elements and the nets that interconnect them
  - Ends at a clocked element
- Clocked elements include flip-flops, block RAMs, DSP cells, …
- Combinatorial elements include LUTs, wide MUXes, carry chains, …



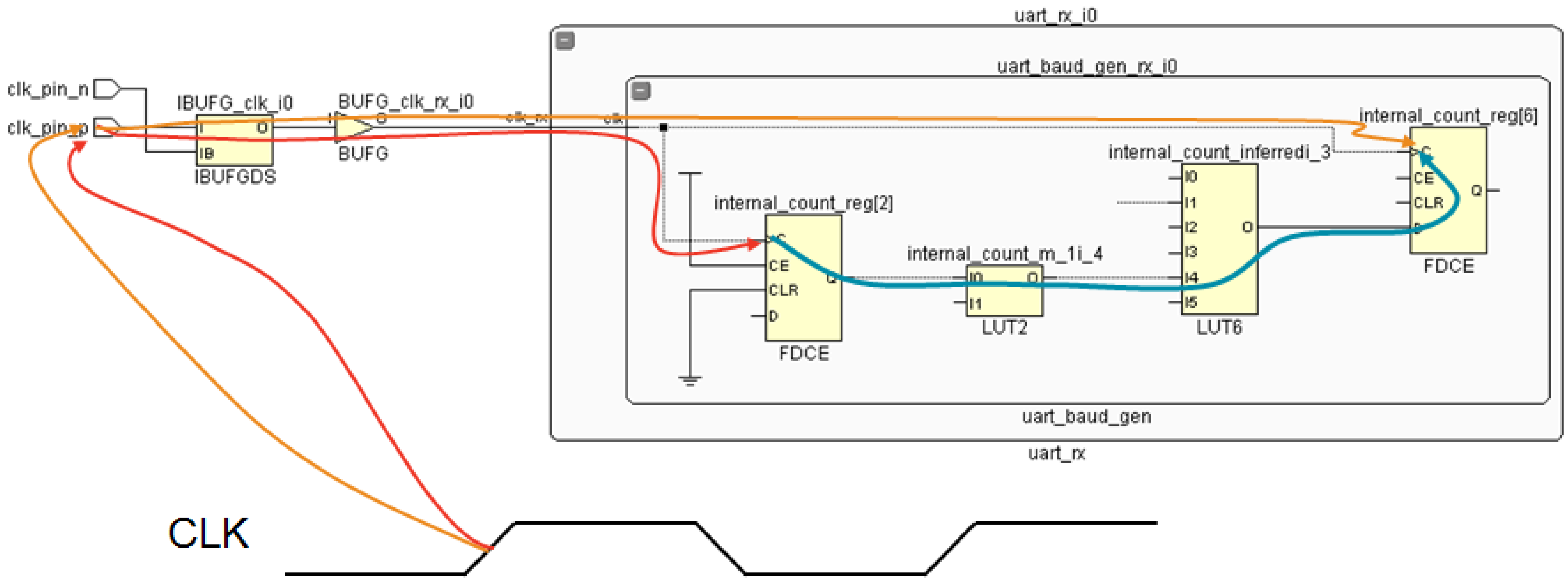
Setup Check

- **Checks that a change in a clocked element has time to propagate to other clocked elements before the next clock event**
  - That is, from the rising edge of the clock to the next rising edge of the clock
  - Checked for all static timing paths



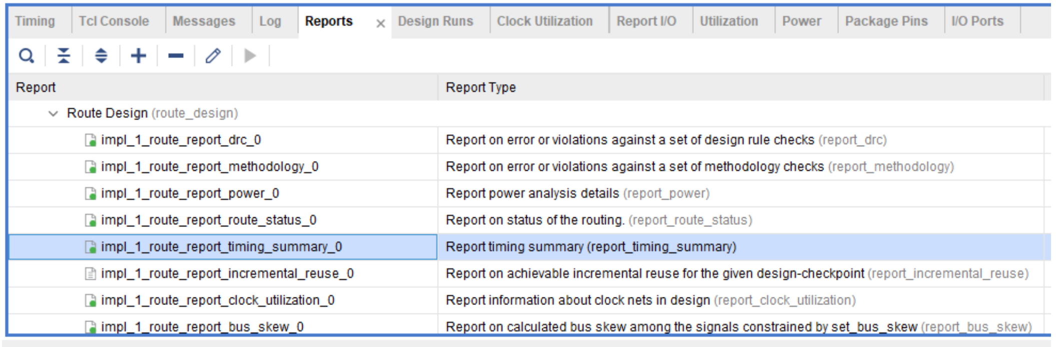
Hold Check

- **Checks that a change in a clocked element caused by a clock event does not propagate to a destination clocked element before the same clock event arrives at the destination element**
  - Usually from the rising edge of the clock to the same edge of the clock
  - Checked for all static timing paths
- **The shortest delay is used for Source Clock and Data Path Delay, and the longest delay is used for Destination Clock Delay**

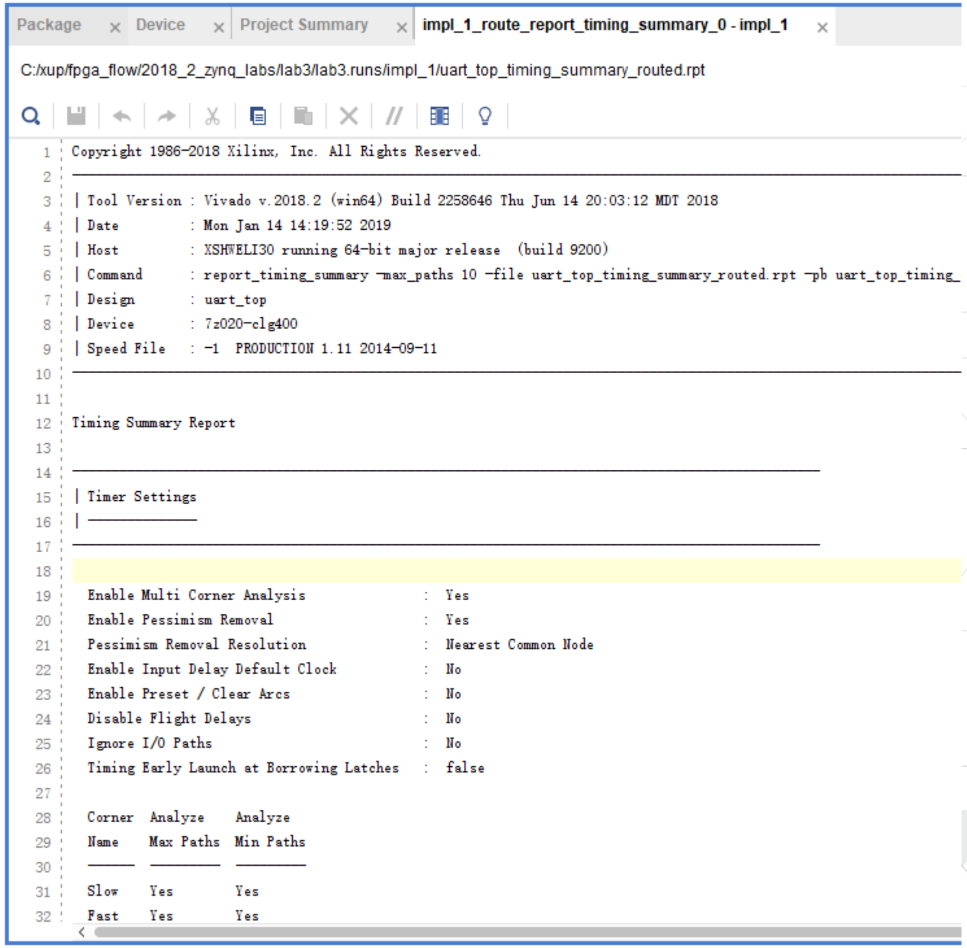


report_timing_summary: TRCE Like Report

- `report_timing_summary` **command produces a comprehensive TRCE-like timing report**
- **This report contains these sections**
  - Timer Settings
  - check_timing report
  - Design Timing Summary
  - Clock Definitions
  - Intra Clock table
  - Inter Clock table
  - Path Group table



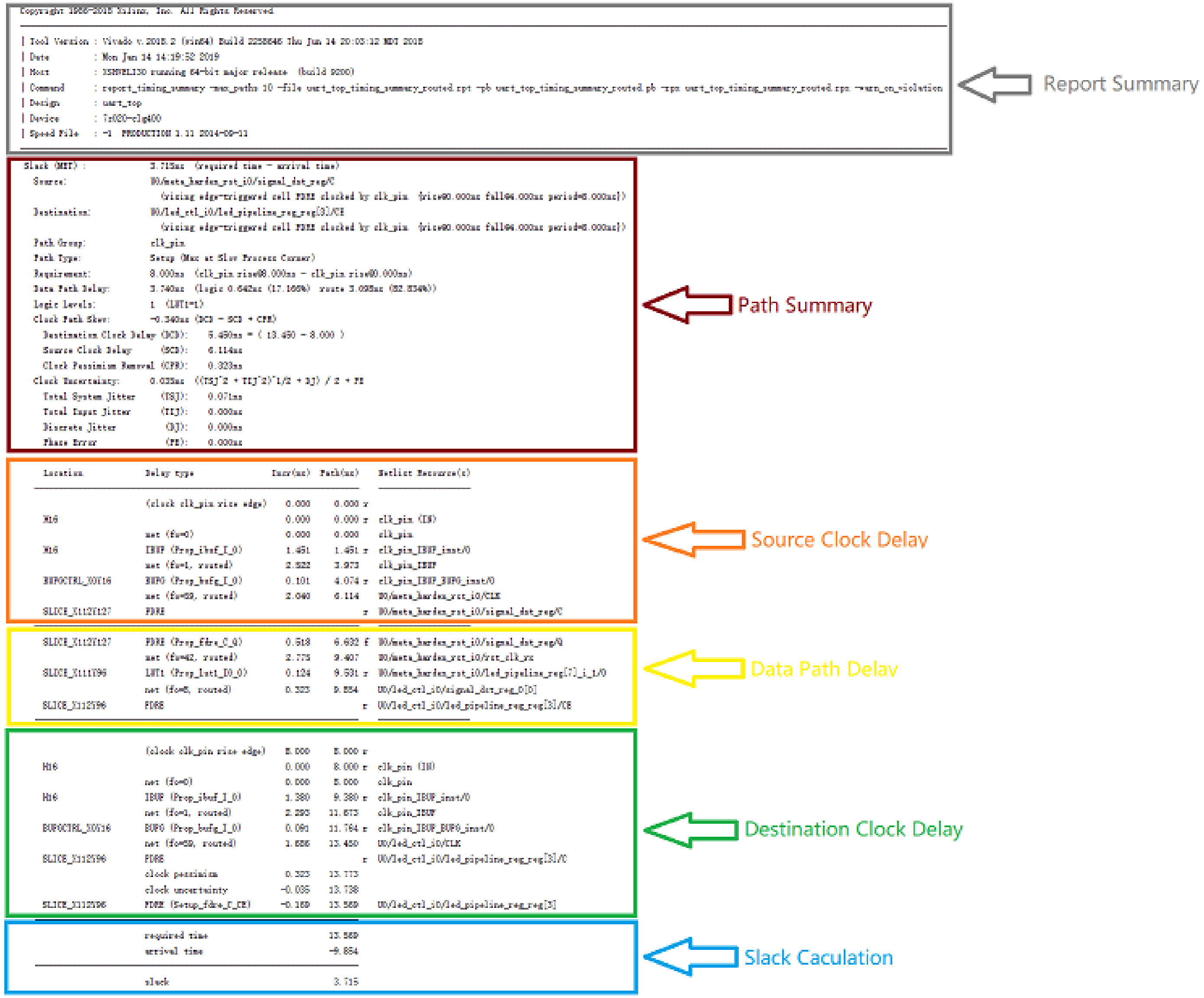
Report	Report Type
Route Design (route_design)	
impl_1_route_report_drc_0	Report on error or violations against a set of design rule checks (report_drc)
impl_1_route_report_methodology_0	Report on error or violations against a set of methodology checks (report_methodology)
impl_1_route_report_power_0	Report power analysis details (report_power)
impl_1_route_report_route_status_0	Report on status of the routing. (report_route_status)
impl_1_route_report_timing_summary_0	Report timing summary (report_timing_summary)
impl_1_route_report_incremental_reuse_0	Report on achievable incremental reuse for the given design-checkpoint (report_incremental_reuse)
impl_1_route_report_clock_utilization_0	Report information about clock nets in design (report_clock_utilization)
impl_1_route_report_bus_skew_0	Report on calculated bus skew among the signals constrained by set_bus_skew (report_bus_skew)



C:/xup/fpga_flow/2018_2_zynq_labs/lab3/lab3.runs/impl_1/uart_top_timing_summary_routed.rpt

```
Copyright 1986-2018 Xilinx, Inc. All Rights Reserved.
-------------------------------------------------------------------------------------------
| Tool Version : Vivado v.2018.2 (win64) Build 2258646 Thu Jun 14 20:03:12 MDT 2018
| Date         : Mon Jan 14 14:19:52 2019
| Host         : XSHWELI30 running 64-bit major release  (build 9200)
| Command      : report_timing_summary -max_paths 10 -file uart_top_timing_summary_routed.rpt -pb uart_top_timing_
| Design       : uart_top
| Device       : 7z020-clg400
| Speed File   : -1  PRODUCTION 1.11 2014-09-11
-------------------------------------------------------------------------------------------


Timing Summary Report

------------------------------------------------------------------------------
| Timer Settings
| --------------
------------------------------------------------------------------------------

  Enable Multi Corner Analysis                  :  Yes
  Enable Pessimism Removal                      :  Yes
  Pessimism Removal Resolution                  :  Nearest Common Node
  Enable Input Delay Default Clock              :  No
  Enable Preset / Clear Arcs                    :  No
  Disable Flight Delays                         :  No
  Ignore I/O Paths                              :  No
  Timing Early Launch at Borrowing Latches      :  false

  Corner  Analyze    Analyze
  Name    Max Paths  Min Paths
  ------  ---------  ---------
  Slow    Yes        Yes
  Fast    Yes        Yes
```

Anatomy of Timing Report



```
Copyright 1986-2018 Xilinx, Inc. All Rights Reserved.
------------------------------------------------------------------------------------
| Tool Version : Vivado v.2018.2 (win64) Build 2258646 Thu Jun 14 20:03:12 MDT 2018
| Date         : Mon Jan 14 14:19:52 2019
| Host         : XSMVELI30 running 64-bit major release  (build 9200)
| Command      : report_timing_summary -max_paths 10 -file uart_top_timing_summary_routed.rpt -pb uart_top_timing_summary_routed.pb -rpx uart_top_timing_summary_routed.rpx -warn_on_violation
| Design       : uart_top
| Device       : 7z020-clg400
| Speed File   : -1  PRODUCTION 1.11 2014-09-11
------------------------------------------------------------------------------------

Slack (MET) :             3.715ns  (required time - arrival time)
  Source:                 U0/meta_harden_rst_i0/signal_dst_reg/C
                            (rising edge-triggered cell FDRE clocked by clk_pin  {rise@0.000ns fall@4.000ns period=8.000ns})
  Destination:            U0/led_ctl_i0/led_pipeline_reg_reg[3]/CE
                            (rising edge-triggered cell FDRE clocked by clk_pin  {rise@0.000ns fall@4.000ns period=8.000ns})
  Path Group:             clk_pin
  Path Type:              Setup (Max at Slow Process Corner)
  Requirement:            8.000ns  (clk_pin rise@8.000ns - clk_pin rise@0.000ns)
  Data Path Delay:        3.740ns  (logic 0.642ns (17.166%)  route 3.098ns (82.834%))
  Logic Levels:           1  (LUT1=1)
  Clock Path Skew:        -0.340ns (DCD - SCD + CPR)
    Destination Clock Delay (DCD):    5.450ns = ( 13.450 - 8.000 )
    Source Clock Delay      (SCD):    6.114ns
    Clock Pessimism Removal (CPR):    0.323ns
  Clock Uncertainty:      0.035ns  ((TSJ^2 + TIJ^2)^1/2 + DJ) / 2 + PE
    Total System Jitter     (TSJ):    0.071ns
    Total Input Jitter      (TIJ):    0.000ns
    Discrete Jitter          (DJ):    0.000ns
    Phase Error              (PE):    0.000ns

    Location             Delay type                Incr(ns)  Path(ns)    Netlist Resource(s)
  -------------------------------------------------------------------    -------------------
                         (clock clk_pin rise edge)    0.000     0.000 r
    H16                                               0.000     0.000 r  clk_pin (IN)
                         net (fo=0)                   0.000     0.000    clk_pin
    H16                  IBUF (Prop_ibuf_I_O)         1.451     1.451 r  clk_pin_IBUF_inst/O
                         net (fo=1, routed)           2.522     3.973    clk_pin_IBUF
    BUFGCTRL_X0Y16       BUFG (Prop_bufg_I_O)         0.101     4.074 r  clk_pin_IBUF_BUFG_inst/O
                         net (fo=59, routed)          2.040     6.114    U0/meta_harden_rst_i0/CLK
    SLICE_X112Y127       FDRE                                         r  U0/meta_harden_rst_i0/signal_dst_reg/C
  -------------------------------------------------------------------    -------------------
    SLICE_X112Y127       FDRE (Prop_fdre_C_Q)         0.518     6.632 f  U0/meta_harden_rst_i0/signal_dst_reg/Q
                         net (fo=42, routed)          2.775     9.407    U0/meta_harden_rst_i0/rst_clk_rx
    SLICE_X111Y96        LUT1 (Prop_lut1_I0_O)        0.124     9.531 r  U0/meta_harden_rst_i0/led_pipeline_reg[7]_i_1/O
                         net (fo=8, routed)           0.323     9.854    U0/led_ctl_i0/signal_dst_reg_0[0]
    SLICE_X112Y96        FDRE                                         r  U0/led_ctl_i0/led_pipeline_reg_reg[3]/CE
  -------------------------------------------------------------------    -------------------

                         (clock clk_pin rise edge)    8.000     8.000 r
    H16                                               0.000     8.000 r  clk_pin (IN)
                         net (fo=0)                   0.000     8.000    clk_pin
    H16                  IBUF (Prop_ibuf_I_O)         1.380     9.380 r  clk_pin_IBUF_inst/O
                         net (fo=1, routed)           2.293    11.673    clk_pin_IBUF
    BUFGCTRL_X0Y16       BUFG (Prop_bufg_I_O)         0.091    11.764 r  clk_pin_IBUF_BUFG_inst/O
                         net (fo=59, routed)          1.686    13.450    U0/led_ctl_i0/CLK
    SLICE_X112Y96        FDRE                                         r  U0/led_ctl_i0/led_pipeline_reg_reg[3]/C
                         clock pessimism              0.323    13.773
                         clock uncertainty           -0.035    13.738
    SLICE_X112Y96        FDRE (Setup_fdre_C_CE)      -0.169    13.569    U0/led_ctl_i0/led_pipeline_reg_reg[3]
  -------------------------------------------------------------------
                         required time                         13.569
                         arrival time                          -9.854
  -------------------------------------------------------------------
                         slack                                  3.715
```

Report Summary

Path Summary

Source Clock Delay

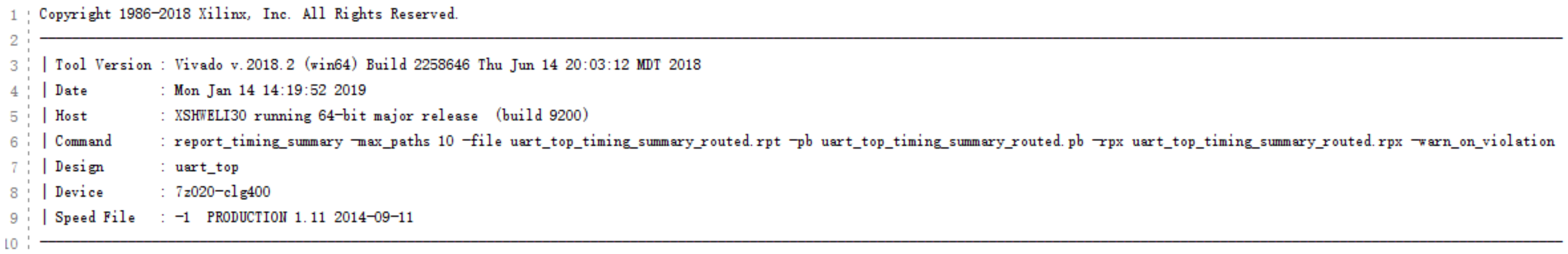
Data Path Delay

Destination Clock Delay

Slack Caculation

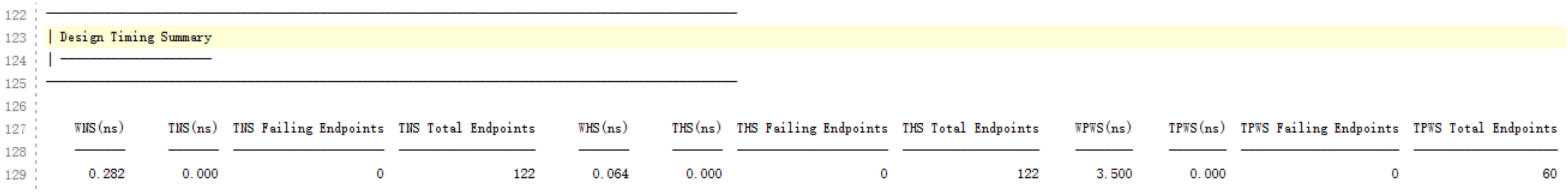
Report Sections

> **Report summary**



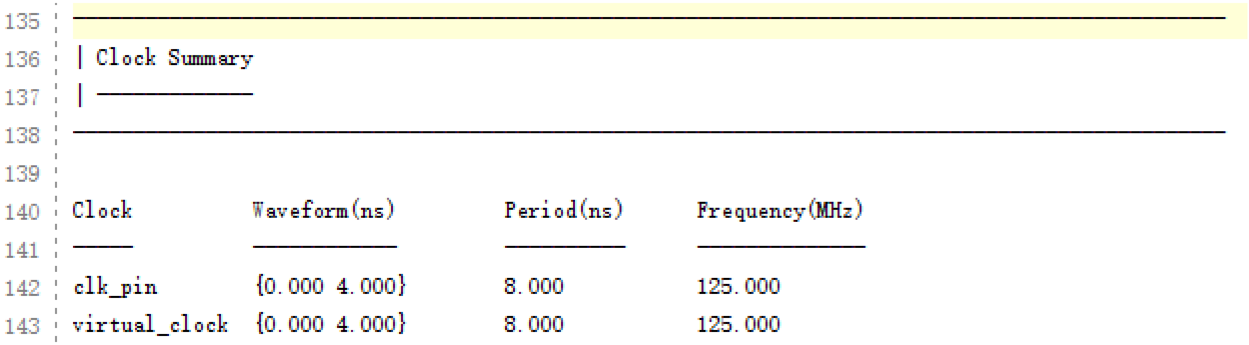
```
Copyright 1986-2018 Xilinx, Inc. All Rights Reserved.
------------------------------------------------------------------------------------------------------------------------------------------
| Tool Version : Vivado v.2018.2 (win64) Build 2258646 Thu Jun 14 20:03:12 MDT 2018
| Date         : Mon Jan 14 14:19:52 2019
| Host         : XSHWELI30 running 64-bit major release  (build 9200)
| Command      : report_timing_summary -max_paths 10 -file uart_top_timing_summary_routed.rpt -pb uart_top_timing_summary_routed.pb -rpx uart_top_timing_summary_routed.rpx -warn_on_violation
| Design       : uart_top
| Device       : 7z020-clg400
| Speed File   : -1  PRODUCTION 1.11 2014-09-11
------------------------------------------------------------------------------------------------------------------------------------------
```

> **Design timing summary**



```
------------------------------------------------------------------------------------------------
| Design Timing Summary
| ---------------------
------------------------------------------------------------------------------------------------

    WNS(ns)      TNS(ns)  TNS Failing Endpoints  TNS Total Endpoints      WHS(ns)      THS(ns)  THS Failing Endpoints  THS Total Endpoints     WPWS(ns)     TPWS(ns)  TPWS Failing Endpoints  TPWS Total Endpoints
    -------      -------  ---------------------  -------------------      -------      -------  ---------------------  -------------------     --------     --------  ----------------------  --------------------
      0.282        0.000                      0                  122        0.064        0.000                      0                  122        3.500        0.000                       0                    60
```

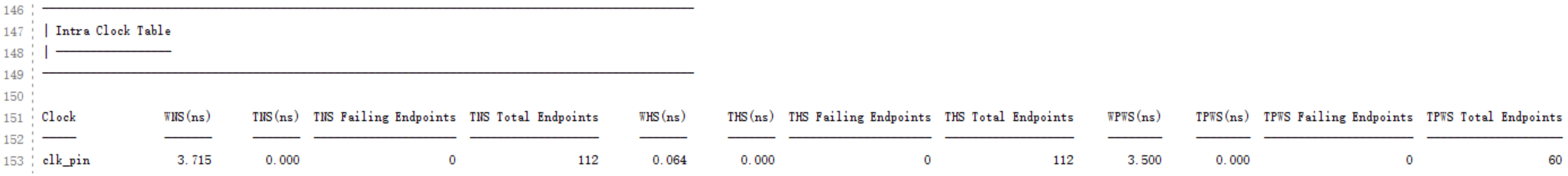
> **Clock summary**



```
------------------------------------------------------------------------------------------------
| Clock Summary
| -------------
------------------------------------------------------------------------------------------------

Clock          Waveform(ns)       Period(ns)      Frequency(MHz)
-----          ------------       ----------      --------------
clk_pin        {0.000 4.000}      8.000           125.000
virtual_clock  {0.000 4.000}      8.000           125.000
```

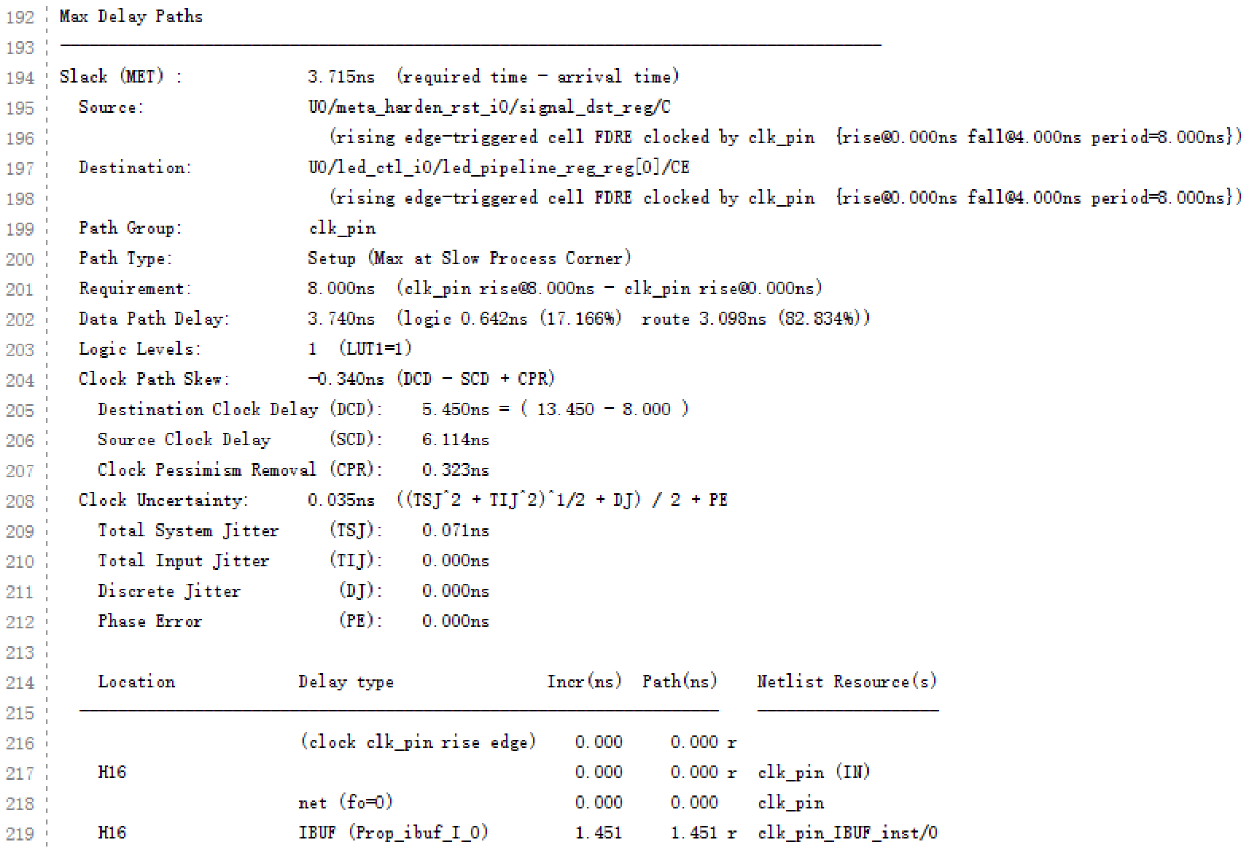
Report Sections

> **Intra clock table**



```
-----------------------------------------------------------------------------------------------
| Intra Clock Table
| -----------------
-----------------------------------------------------------------------------------------------

Clock            WNS(ns)    TNS(ns)  TNS Failing Endpoints  TNS Total Endpoints    WHS(ns)    THS(ns)  THS Failing Endpoints  THS Total Endpoints    WPWS(ns)    TPWS(ns)  TPWS Failing Endpoints  TPWS Total Endpoints
-----            -------    -------  ---------------------  -------------------    -------    -------  ---------------------  -------------------    --------    --------  ----------------------  --------------------
clk_pin            3.715      0.000                      0                  112      0.064      0.000                      0                  112       3.500       0.000                       0                    60
```

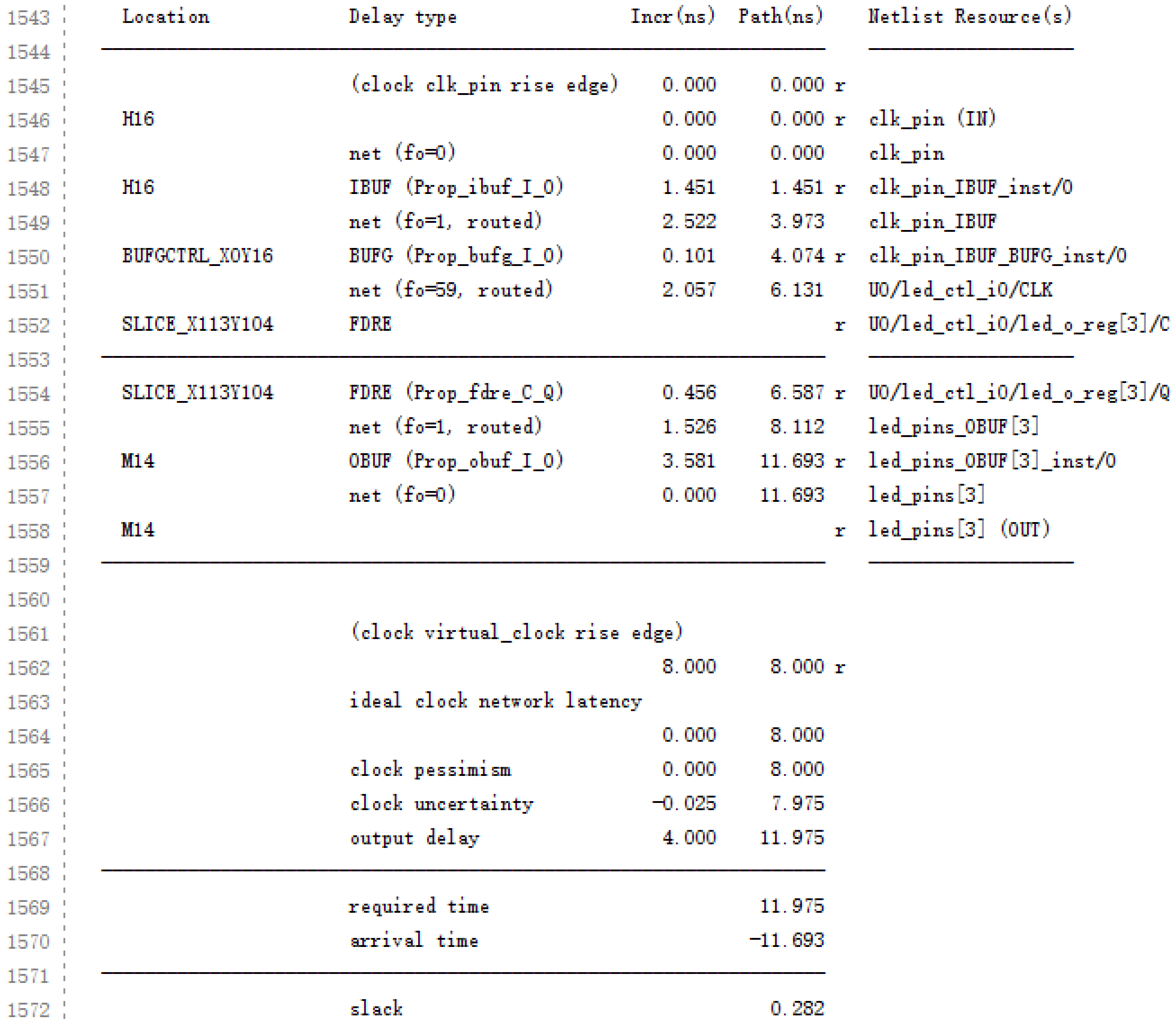
> **Maximum delay path**



```
Max Delay Paths
--------------------------------------------------------------------------------------
Slack (MET) :             3.715ns  (required time - arrival time)
  Source:                 U0/meta_harden_rst_i0/signal_dst_reg/C
                            (rising edge-triggered cell FDRE clocked by clk_pin  {rise@0.000ns fall@4.000ns period=8.000ns})
  Destination:            U0/led_ctl_i0/led_pipeline_reg_reg[0]/CE
                            (rising edge-triggered cell FDRE clocked by clk_pin  {rise@0.000ns fall@4.000ns period=8.000ns})
  Path Group:             clk_pin
  Path Type:              Setup (Max at Slow Process Corner)
  Requirement:            8.000ns  (clk_pin rise@8.000ns - clk_pin rise@0.000ns)
  Data Path Delay:        3.740ns  (logic 0.642ns (17.166%)  route 3.098ns (82.834%))
  Logic Levels:           1  (LUT1=1)
  Clock Path Skew:        -0.340ns (DCD - SCD + CPR)
    Destination Clock Delay (DCD):    5.450ns = ( 13.450 - 8.000 )
    Source Clock Delay      (SCD):    6.114ns
    Clock Pessimism Removal (CPR):    0.323ns
  Clock Uncertainty:      0.035ns  ((TSJ^2 + TIJ^2)^1/2 + DJ) / 2 + PE
    Total System Jitter     (TSJ):    0.071ns
    Total Input Jitter      (TIJ):    0.000ns
    Discrete Jitter          (DJ):    0.000ns
    Phase Error              (PE):    0.000ns

    Location             Delay type                Incr(ns)  Path(ns)    Netlist Resource(s)
  -------------------------------------------------------------------    -------------------
                         (clock clk_pin rise edge)    0.000     0.000 r
    H16                                               0.000     0.000 r  clk_pin (IN)
                         net (fo=0)                   0.000     0.000    clk_pin
    H16                  IBUF (Prop_ibuf_I_O)         1.451     1.451 r  clk_pin_IBUF_inst/O
```

Report Sections

> Delay path

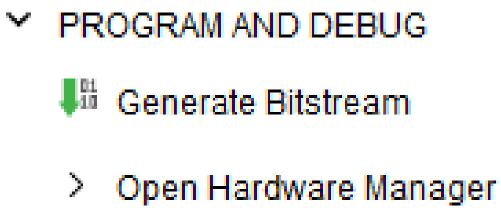


```
  Location             Delay type                Incr(ns)  Path(ns)    Netlist Resource(s)
  -------------------------------------------------------------------    -------------------
                       (clock clk_pin rise edge)    0.000     0.000 r
  H16                                               0.000     0.000 r  clk_pin (IN)
                       net (fo=0)                   0.000     0.000    clk_pin
  H16                  IBUF (Prop_ibuf_I_O)         1.451     1.451 r  clk_pin_IBUF_inst/O
                       net (fo=1, routed)           2.522     3.973    clk_pin_IBUF
  BUFGCTRL_X0Y16       BUFG (Prop_bufg_I_O)         0.101     4.074 r  clk_pin_IBUF_BUFG_inst/O
                       net (fo=59, routed)          2.057     6.131    U0/led_ctl_i0/CLK
  SLICE_X113Y104       FDRE                                         r  U0/led_ctl_i0/led_o_reg[3]/C
  -------------------------------------------------------------------    -------------------
  SLICE_X113Y104       FDRE (Prop_fdre_C_Q)         0.456     6.587 r  U0/led_ctl_i0/led_o_reg[3]/Q
                       net (fo=1, routed)           1.526     8.112    led_pins_OBUF[3]
  M14                  OBUF (Prop_obuf_I_O)         3.581    11.693 r  led_pins_OBUF[3]_inst/O
                       net (fo=0)                   0.000    11.693    led_pins[3]
  M14                                                               r  led_pins[3] (OUT)
  -------------------------------------------------------------------    -------------------

                       (clock virtual_clock rise edge)
                                                    8.000     8.000 r
                       ideal clock network latency
                                                    0.000     8.000
                       clock pessimism              0.000     8.000
                       clock uncertainty           -0.025     7.975
                       output delay                 4.000    11.975
  -------------------------------------------------------------------
                       required time                         11.975
                       arrival time                         -11.693
  -------------------------------------------------------------------
                       slack                                  0.282
```

Bitstream Generation and Verification in Hardware

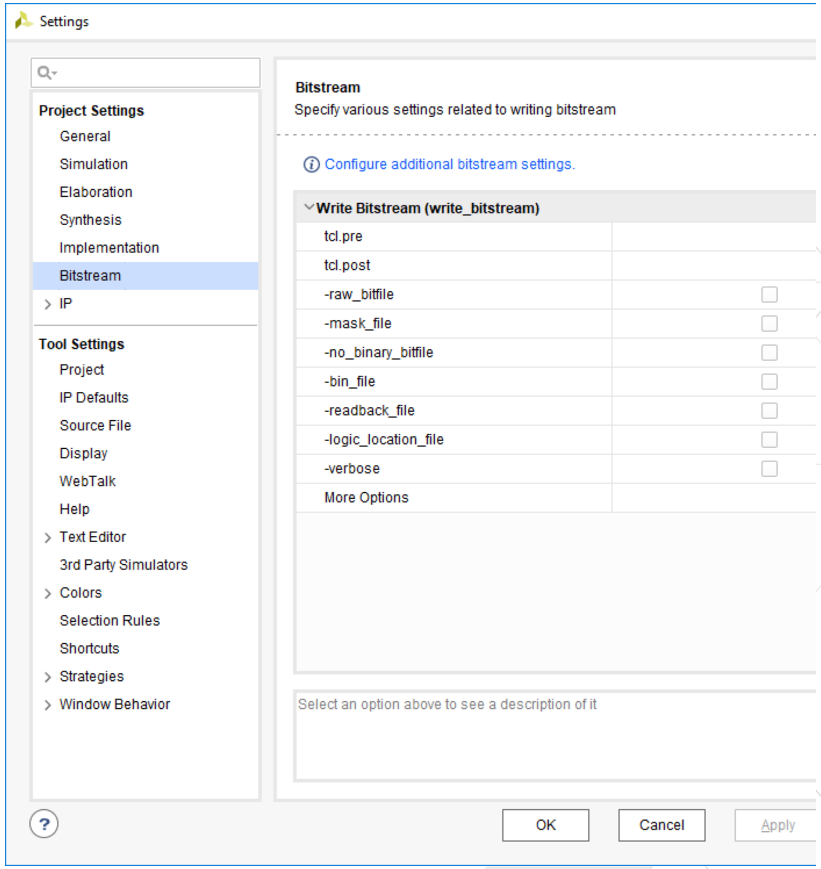
Bitstream Generation

- Generates bitstream for the device chosen for the current project
- Runs on an implemented design
- Uses the pull model to regenerate implemented design if design is out of date
- Project-based flow
  - IDE: Generate bitstream
  - Tcl: `launch_runs impl_1 -to_step write_bitstream`



Bitstream Generation Settings

- **By default binary bitstream format is used**
- **-raw_bitfile: Causes write_bitstream to write a raw bit file (.rbt) containing the same information in ASCII format**
- **-mask_file: Generates a mask file that masks out dynamic bits in the device fabric**
- **-no_binary_bitfile: Do not write the binary bitstream file (.bit)**
  - Use this command to generate the ASCII bitstream or mask file, or to generate a bitstream report, without generating the binary bitstream file
- **-bin_file: Creates a binary file (.bin) containing only device programming data, without the header information**
- **-logic_location_file: Generates a (.ll) file that contains the location of LUTs, BRAM, flip-flops, latches, I/O block inputs and outputs**



Bitstream

Specify various settings related to writing bitstream

Configure additional bitstream settings.

Write Bitstream (write_bitstream)

tcl.pre

tcl.post

-raw_bitfile □

-mask_file □

-no_binary_bitfile □

-bin_file □

-readback_file □

-logic_location_file □

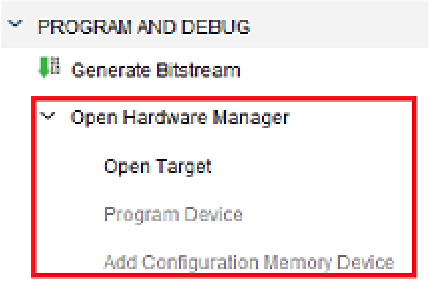
-verbose □

More Options

Select an option above to see a description of it

Hardware Manager

- **The steps to connect to hardware and programing the target FPGA device**
  - Open a hardware manager
    - Uses Target Communication Framework (TCF) Agent, `hw_server`
  - Open a hardware target that is managed by a hardware server running on a host computer
  - Associate the bitstream data programming file with the appropriate FPGA device
  - Program or download the programming file into the hardware device
  - Opens the hardware analyzer view for debugging



Summary

XILINX

Summary

- Implementation is made up of the programs link, opt_design, power_opt, place_design, phys_opt_design, and route_design
- There are several implementation reports available to help designers better manage their FPGA designs
- Static timing paths start at clocked elements and end at clocked elements
- Static timing paths are analyzed for setup and hold violations at both fastest and slowest process corners
- Setup and hold checks include the analysis of the clock propagation paths
- report_timing_summary is used as post-implementation sign-off
- report_timing is used for interactive and detailed timing analysis after synthesis or implementation