ADIUVO

ENGINEERING AND TRAINING, LTD.

Introduction to AMD Vivado™

AMD Tools and Frameworks

AMD Vitis™ AI

Enables the implementations of machine learning inference, using Tensor Flow, Caffe and PyTorch

*Requires a SoC/RFSoC/AMD Alveo™

AMD Vitis™

Embedded and accelerated SW development. Used to develop software solutions for MicroBlaze™, Arm® R5, A9, A53 and A72.

Petalinux

Embedded Linux solutions

PYNQ

Python framework for rapid prototyping on SoC/RFSoC/Alveo

AMD Vitis™ / Vivado HLS

High-Level Synthesis (HLS) tool supporting C/C++/OpenCL

AMD Vivado™

Design capture and implementation for the base platform

AMD Vivado™ Overview

- Vivado is the foundation of all design and higher level tools
- Vivado enables us to capture designs using VHDL or Verilog
- Large IP library to accelerate our designs
- Integrates Vivado / Vitis HLS IP cores

- ✓ Simulate designs using Vivado Simulator
- ✓ Synthesize, place and route the design
- ✓ Generate power estimations
- ✓ Create AMD support architecture

AMD Vivado™ Overview

FPGA Implementation Flow

Synthesis – Translates the HDL design into a series of logic equations which are then mapped onto the resources available in the target FPGA.

Place – The logic resources determined by the synthesis tool are placed at available locations within the target device.

Routing – The placed logic resources in the design are interconnected using routing and switch matrixes to implement the final application.

Bit File – The generation of the final programming file for the target FPGA.

We can control the flow implementation settings by using constraints (XDC file) and implementation strategies.

AMD Vivado™ IP Integrator

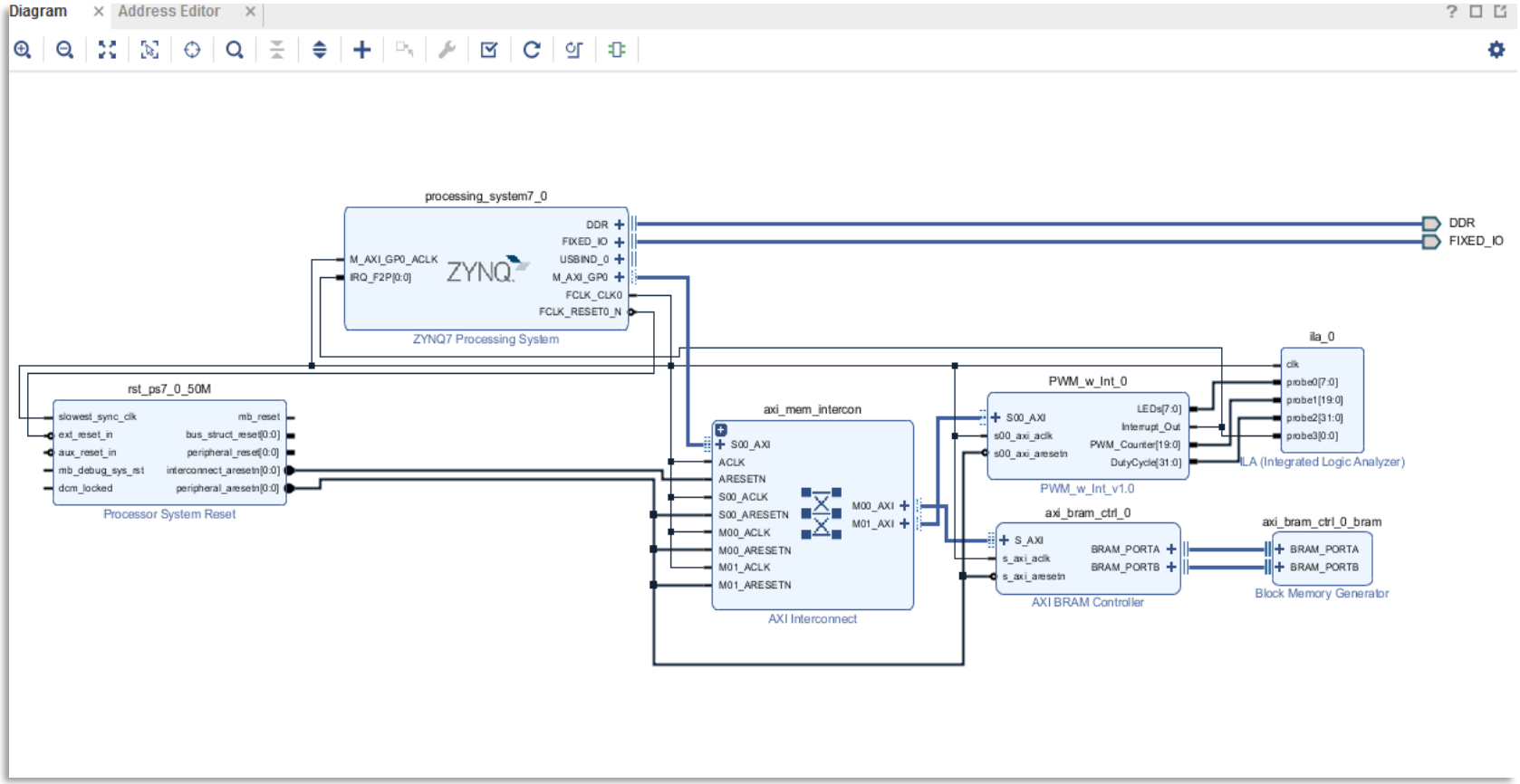
ADIUVO
ENGINEERING AND TRAINING, LTD.

AMD Vivado™ IP Integrator (IPI)

Create system-level designs

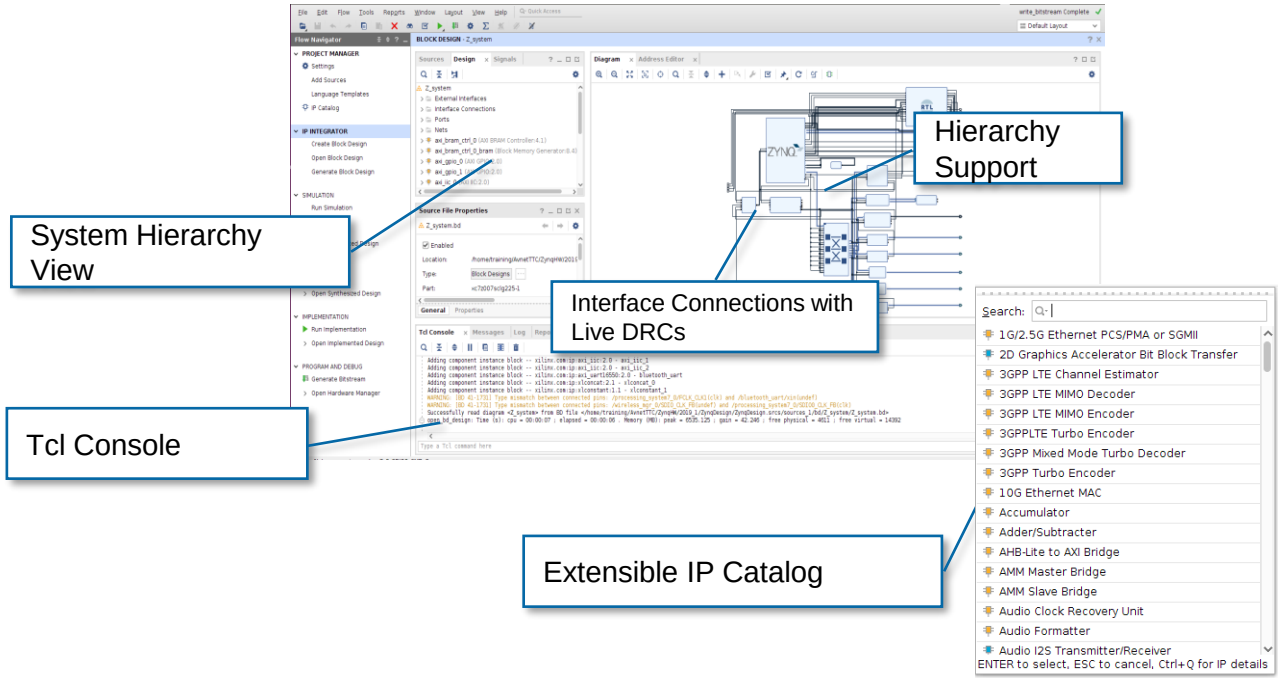
- Instantiate and interconnect IP cores

- IP-centric design flow
  - Plug-and-play IP
  - Vast IP catalog
- Accelerates
  - Integration
  - Productivity
- Example applications
  - Embedded
  - DSP
  - Video
  - Analog
  - Networking



AMD Vivado™ IP Integrator: Intelligent IP Integration

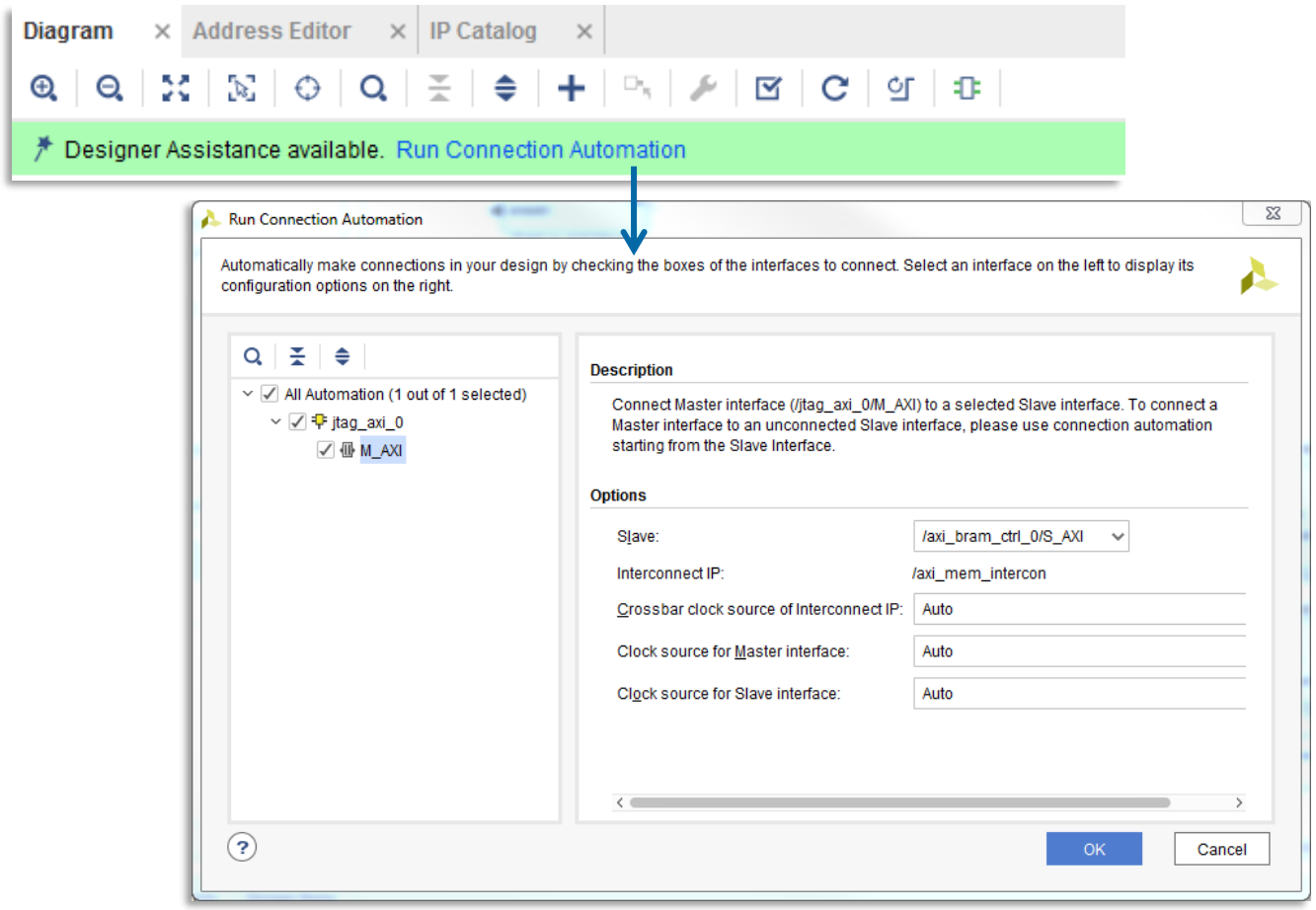
- Automated IP subsystems
- Block automation for rapid design creation
- One click IP customization
- Board aware
- Support all 7 Series FPGAs and AMD Zynq™ SoCs
- Built-in presets, accelerating design creation



AMD Vivado™ IP Integrator: Intelligent IP Integration

Correct-by-construction

- Interface level connections
- Extensible IP repository
- Real-time DRCs and parameter propagation / resolution
- Designer assistance



Demonstration

AMD Vivado™

ADIUVO
ENGINEERING AND TRAINING, LTD.

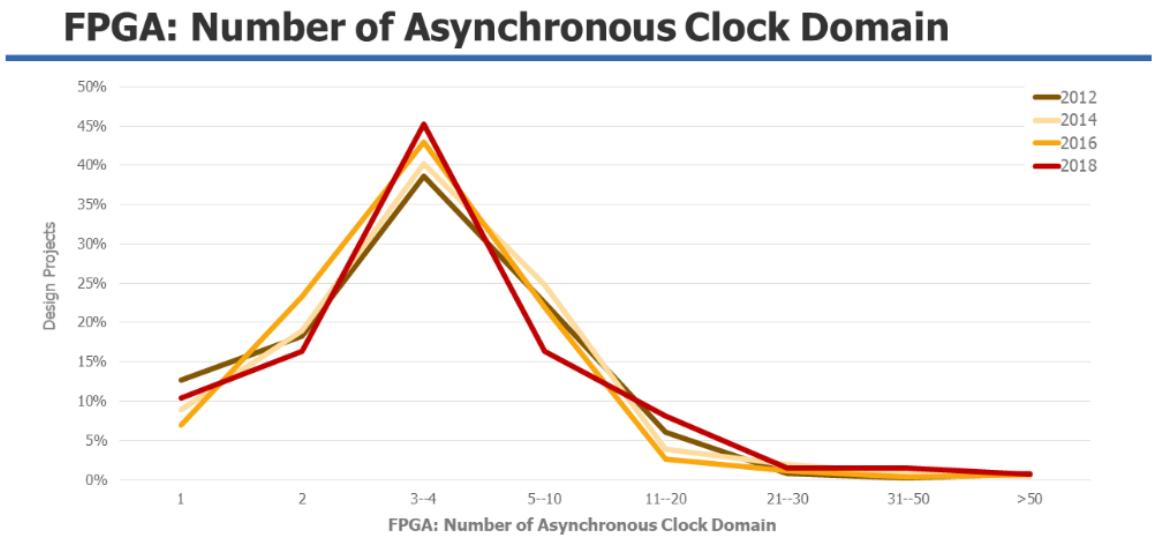
Clock Domain Crossing

ADIUVO
ENGINEERING AND TRAINING, LTD.

Clock Domain Crossing

Ideal solution uses one clock and the entire design is synchronous.

BUT!



Modern devices have multiple clocks to address different clock domains e.g. ADC / DAC clocks, source synchronous interfaces.

Brings with it the need to transfer data, and signals safely and reliability between the clock domains.

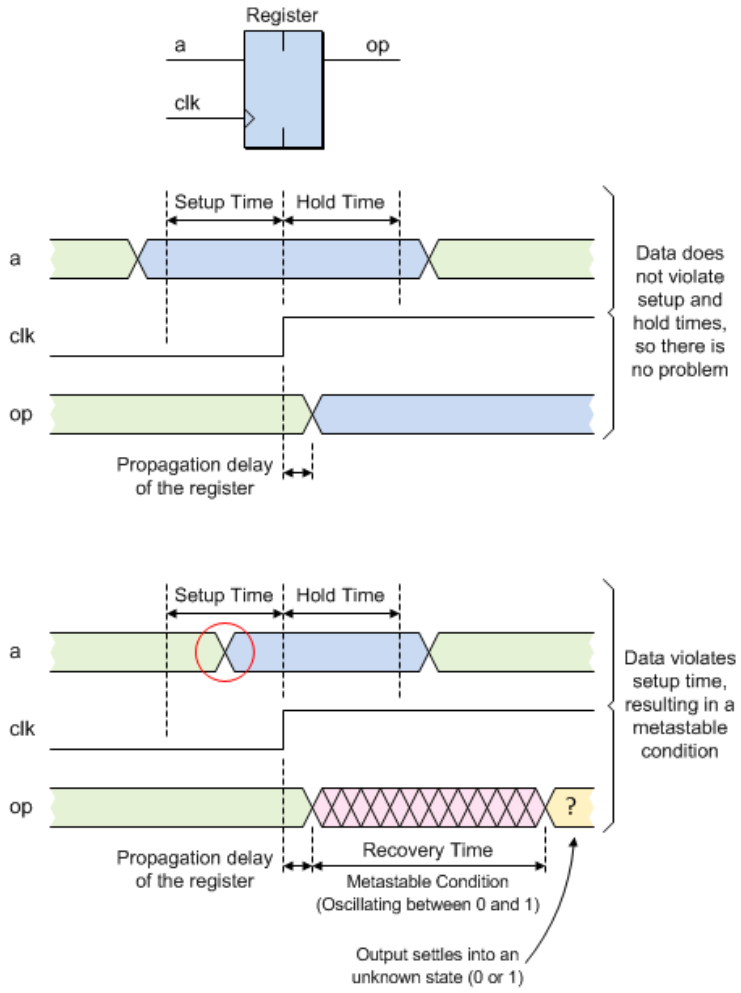
Metastability

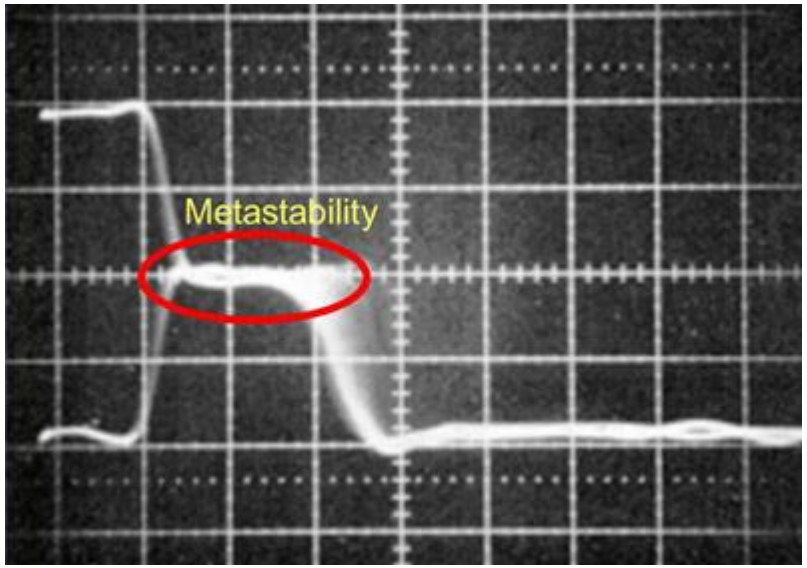
One issue which can arise with incorrect domain crossing is metastability.

This can lead to corruption of data or incorrect behaviour.

Occurs when a flip flops set up or hold time is violated.

Metastability





Clock Domain Crossing

Several techniques can be used depending upon what needs to be transferred:

- Two stage synchroniser – Ideal for single bit data
- Grey code synchroniser – Encodes data bus in grey code and transfer between domains – Ideal for counters as input to be converted to grey code can only decrement / increment by one from previous value
- Hand shake synchroniser – Transfers data bus between two clock domains using handshake signals
- Pulse synchroniser – transfer pulse from one clock domain to another
- Asynchronous FIFO – transfers data from one domain to another, useful for high throughput / burst transfers

Clock Domain Crossing

To support reset functionality across clock domains we may need the following synchronisers structures.

- Asynchronous Reset Synchroniser – enables asynchronous assertion and synchronous de-assertion.
- Synchronous Reset Synchroniser – synchronises a synchronous reset to another clock domain.

CDC in AMD

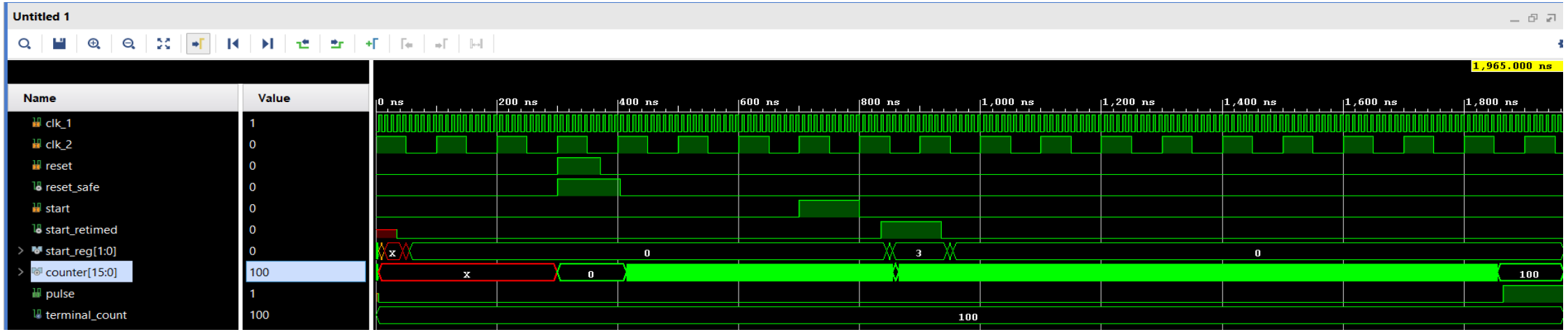
AMD Parameterised Macros (XPM) – provide CDC structures

Use registers optimised for CDC in the fabric

- » Registers located close together and have small set up and hold windows

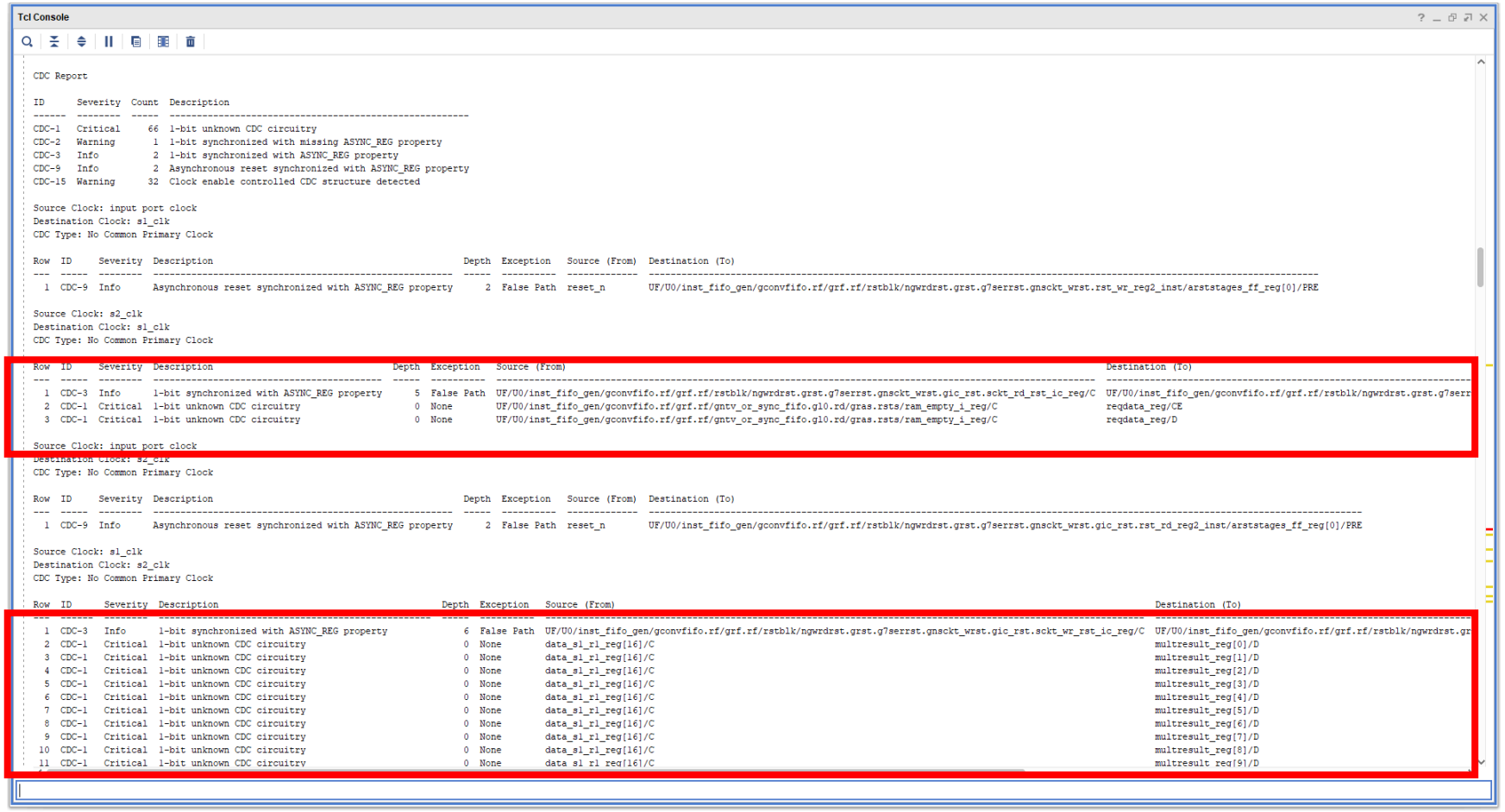
Described within UG953 Libraries Guide

Described within UG 974 UltraScale™ Architecture Libraries

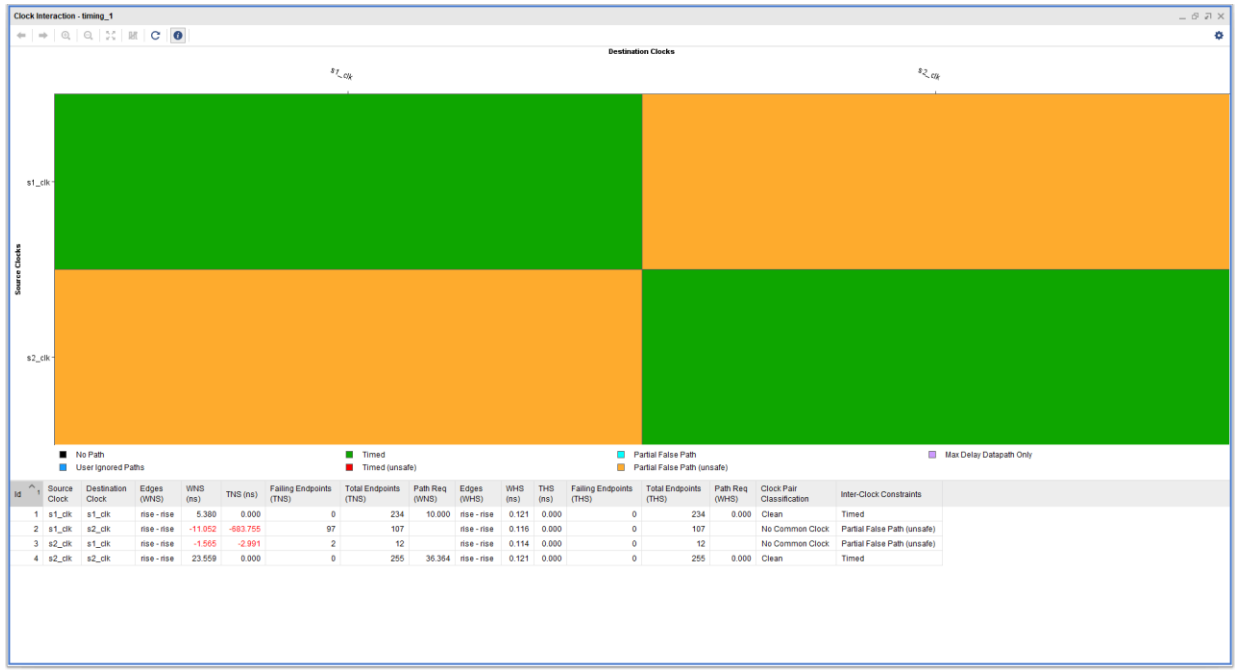


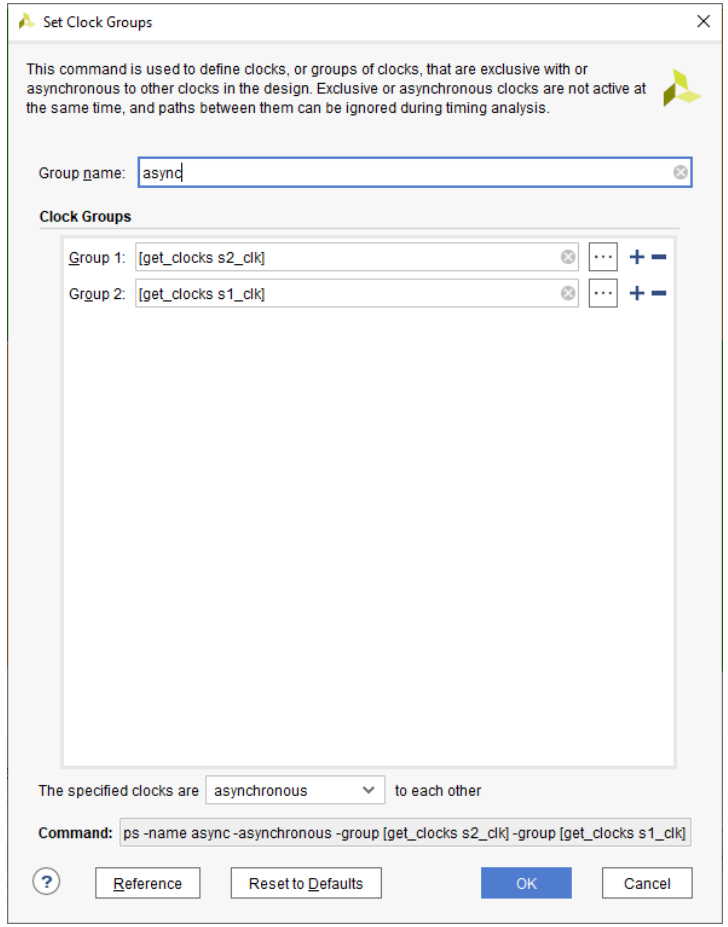
AMD Vivado™ CDC Report

Following Synthesis – in TCL window run the command report_cdc



AMD Vivado™ Timing Analysis – Clock Interaction

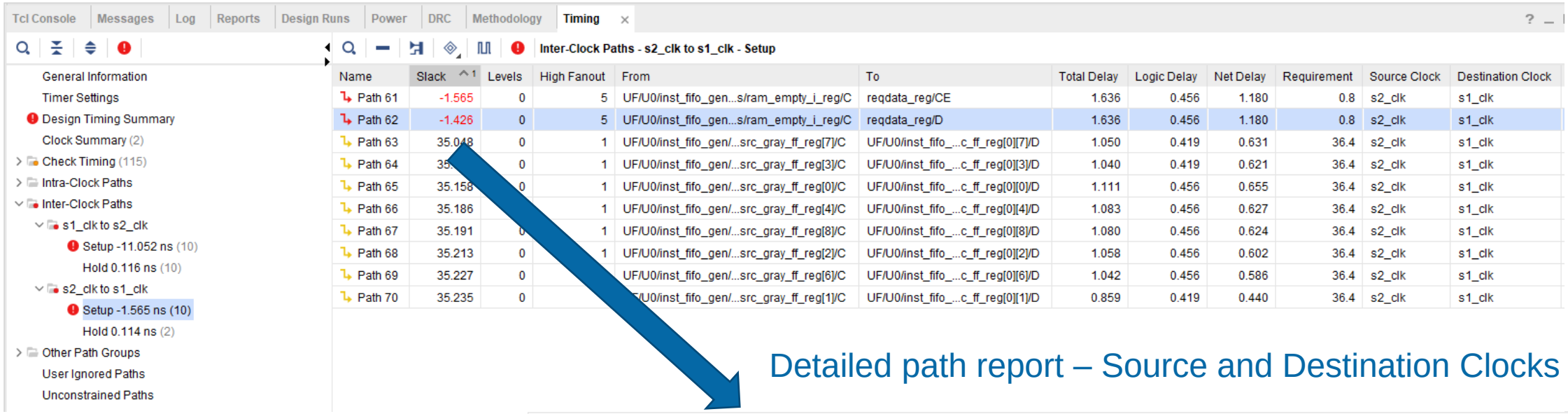




Generated from Flow Navigator when implementation is open.

Yellow show unrelated clock – Indicates CDC issues as well. (Means we need to define constraints.)

AMD Vivado™ Timing Analysis – Inter clock Issues



Name	Slack	Levels	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock	Destination Clock
Path 61	-1.565	0	5	UF/U0/inst_fifo_gen...s/ram_empty_i_reg/C	reqdata_reg/CE	1.636	0.456	1.180	0.8	s2_clk	s1_clk
Path 62	-1.426	0	5	UF/U0/inst_fifo_gen...s/ram_empty_i_reg/C	reqdata_reg/D	1.636	0.456	1.180	0.8	s2_clk	s1_clk
Path 63	35.048	0	1	UF/U0/inst_fifo_gen/...src_gray_ff_reg[7]/C	UF/U0/inst_fifo_...c_ff_reg[0][7]/D	1.050	0.419	0.631	36.4	s2_clk	s1_clk
Path 64	35.	0	1	UF/U0/inst_fifo_gen/...src_gray_ff_reg[3]/C	UF/U0/inst_fifo_...c_ff_reg[0][3]/D	1.040	0.419	0.621	36.4	s2_clk	s1_clk
Path 65	35.158	0	1	UF/U0/inst_fifo_gen/...src_gray_ff_reg[0]/C	UF/U0/inst_fifo_...c_ff_reg[0][0]/D	1.111	0.456	0.655	36.4	s2_clk	s1_clk
Path 66	35.186		1	UF/U0/inst_fifo_gen/...src_gray_ff_reg[4]/C	UF/U0/inst_fifo_...c_ff_reg[0][4]/D	1.083	0.456	0.627	36.4	s2_clk	s1_clk
Path 67	35.191	0	1	UF/U0/inst_fifo_gen/...src_gray_ff_reg[8]/C	UF/U0/inst_fifo_...c_ff_reg[0][8]/D	1.080	0.456	0.624	36.4	s2_clk	s1_clk
Path 68	35.213	0	1	UF/U0/inst_fifo_gen/...src_gray_ff_reg[2]/C	UF/U0/inst_fifo_...c_ff_reg[0][2]/D	1.058	0.456	0.602	36.4	s2_clk	s1_clk
Path 69	35.227	0		UF/U0/inst_fifo_gen/...src_gray_ff_reg[6]/C	UF/U0/inst_fifo_...c_ff_reg[0][6]/D	1.042	0.456	0.586	36.4	s2_clk	s1_clk
Path 70	35.235	0		/U0/inst_fifo_gen/...src_gray_ff_reg[1]/C	UF/U0/inst_fifo_...c_ff_reg[0][1]/D	0.859	0.419	0.440	36.4	s2_clk	s1_clk

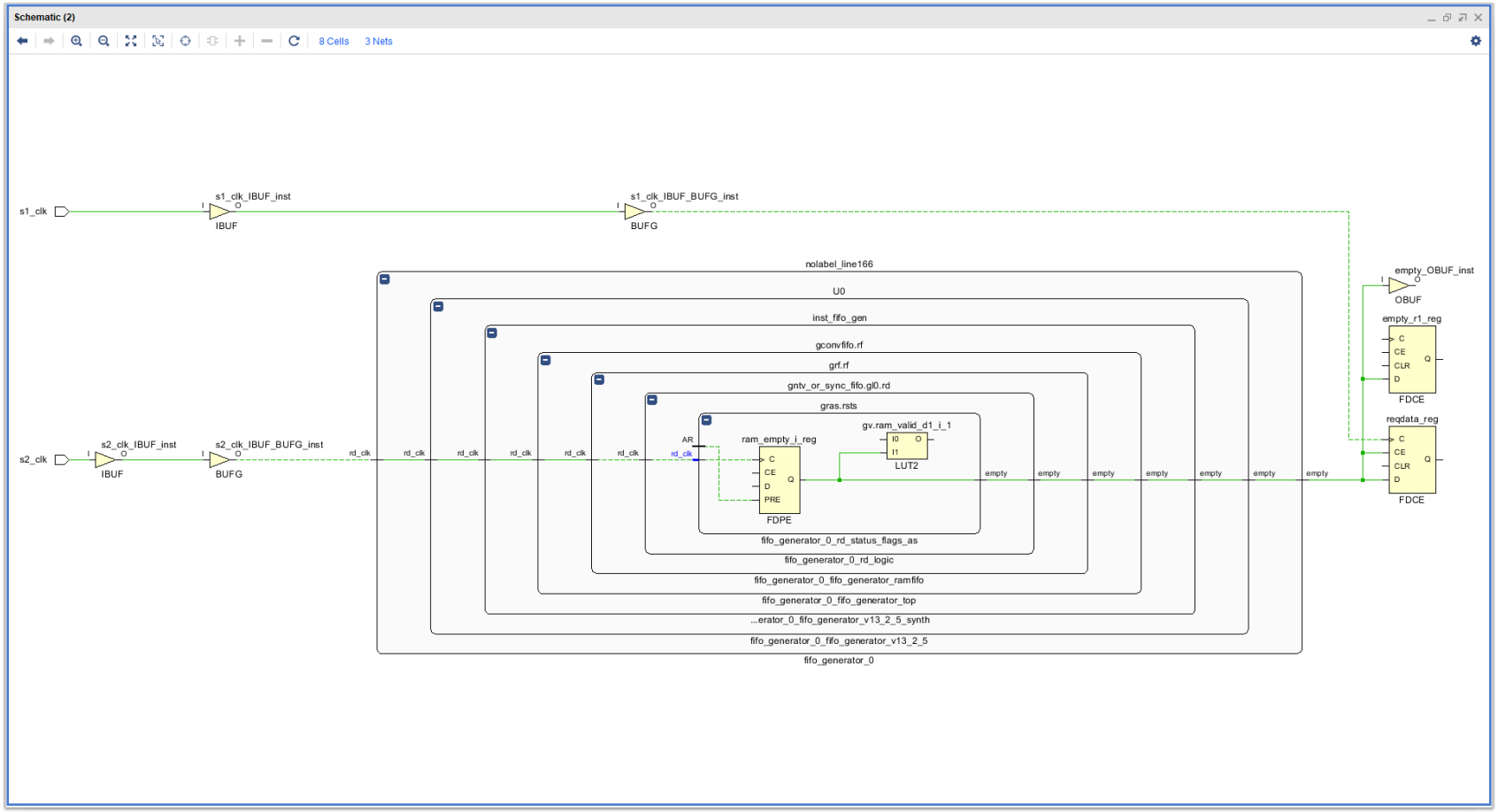
Detailed path report – Source and Destination Clocks different

Summary	
Name	Path 62
Slack	-1.426ns
Source	UF/U0/inst_fifo_gen/gconvfifo.rf/grf.rf/gntv_or_sync_fifo.gl0.rd/gras.rsts/ram_empty_i_reg/C (rising edge-triggered cell FDPE clocked by s2_clk {rise@0.000ns fall@18.182ns period=36.364ns})
Destination	reqdata_reg/D (rising edge-triggered cell FDCE clocked by s1_clk {rise@0.000ns fall@5.000ns period=10.000ns})
Path Group	s1_clk
Path Type	Setup (Max at Slow Process Corner)
Requirement	0.812ns (s1_clk rise@9710.000ns - s2_clk rise@9709.188ns)
Data Path Delay	1.636ns (logic 0.456ns (27.876%) route 1.180ns (72.124%))
Logic Levels	0
Clock Path Skew	-0.537ns
Clock Uncertainty	0.035ns
Clock Dom...Crossing	Inter clock paths are considered valid unless explicitly excluded by timing constraints such as set_clock_groups or set_false_path.

If you forget to report_cdc following synthesis. CDC issues will be apparent in the timing report if we have not correctly addressed the constraints.

AMD Vivado™ Flow - Schematic

Engineers can use information provided in the text report to navigate to the schematic to understand where CDC might be.



Strategies and Reports

ADIUVO
ENGINEERING AND TRAINING, LTD.

AMD Vivado™ – Strategies

- Strategies are a defined set of Vivado implementation feature options that control the implementation results.
- These strategies can be used to explore:
  - Timing Performance (e.g., Performance_Explore)
  - Congestion - Strategies to reduce routing congestion in areas of the design
  - Area - Optimize for area
  - Power – Optimize for power
  - Quick Flow – Reduced implementation time

We should always try to achieve timing closure.

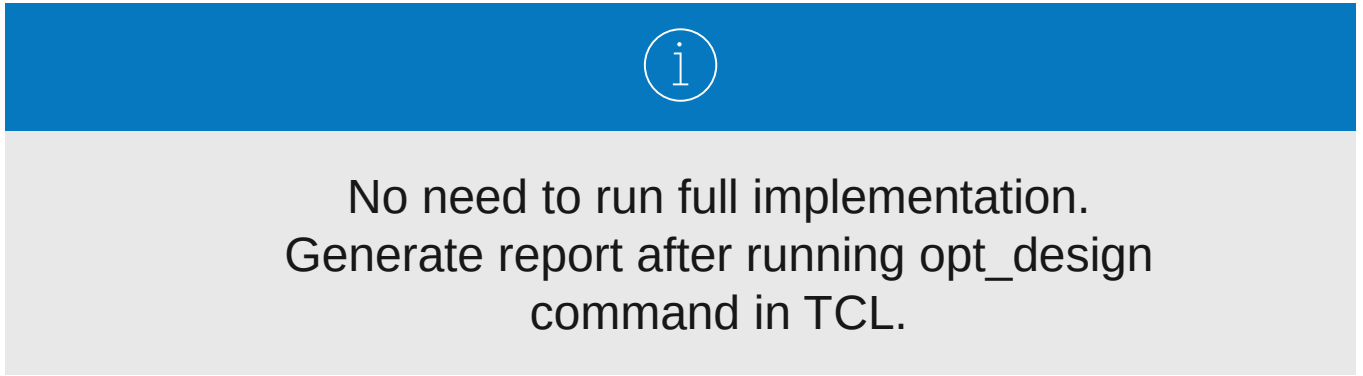
AMD Vivado™ – Reports

- Vivado provides several reports which can be used to help focus in on performance issues in the design:
  - Design Analysis Report – Provides information on design timing, congestion, and complexity of design.
  - Quality of Result Report – Provides overall design assessment and methodology check – QOR can also make suggestions to fix issues in the design.
- Both are very useful to achieve timing closure of the design.

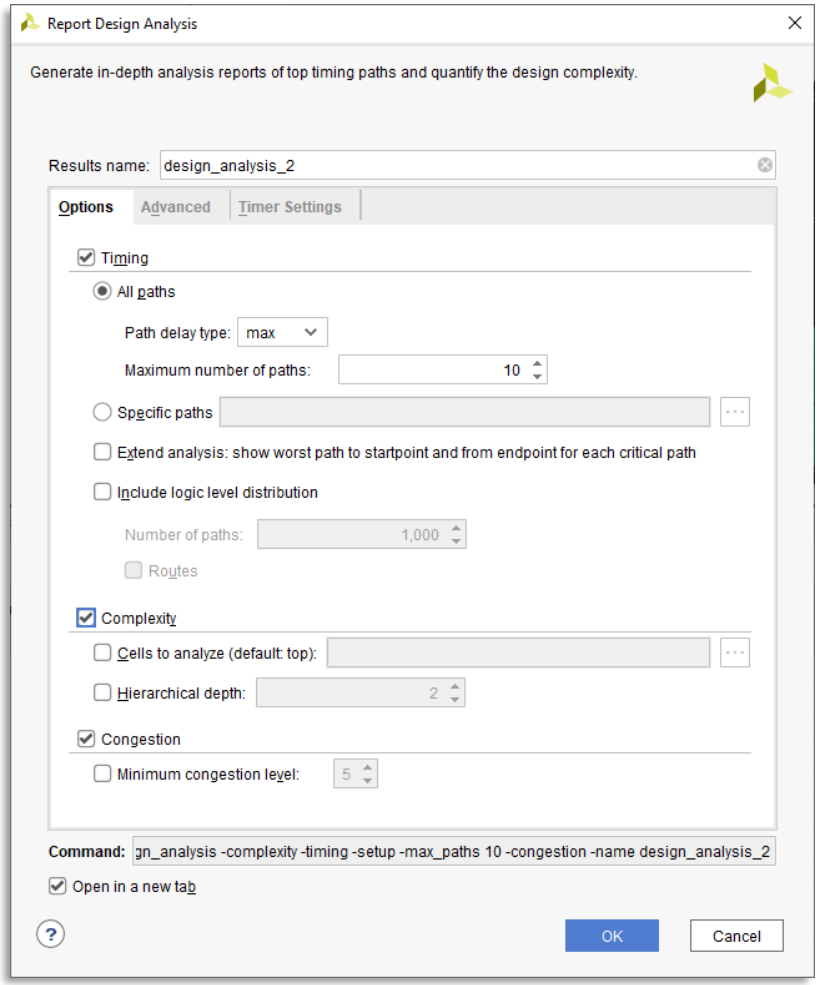
AMD Vivado™ – Design Analysis Report

Design Analysis Report provides information on:

- Timing – Provides information on the timing and physical characteristics of timing paths.
- Complexity – Provides information on routing complexity and LUT distribution.
- Congestion – Provides information on routing congestion.

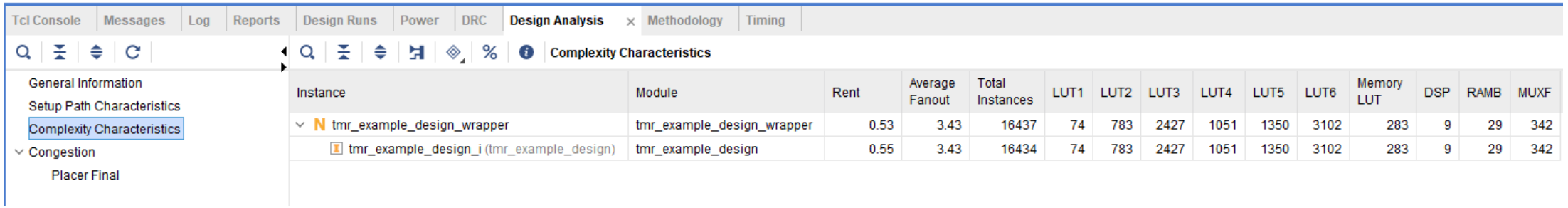


No need to run full implementation.
Generate report after running opt_design command in TCL.



AMD Vivado™ – Design Analysis Report

- Along with timing information, DAR can provide information on design complexity including indicating design risk for implementation.
- Low Risk Rent Analysis <0.65 and Fan Out <4.
- High Risk Rent Analysis >0.65 <0.85 and Fan Out >4 % <5 – May be difficult to place without congestion.



Instance	Module	Rent	Average Fanout	Total Instances	LUT1	LUT2	LUT3	LUT4	LUT5	LUT6	Memory LUT	DSP	RAMB	MUXF
N tmr_example_design_wrapper	tmr_example_design_wrapper	0.53	3.43	16437	74	783	2427	1051	1350	3102	283	9	29	342
tmr_example_design_i (tmr_example_design)	tmr_example_design	0.55	3.43	16434	74	783	2427	1051	1350	3102	283	9	29	342

AMD Vivado™ – Quality of Result

- Quality of Result Assessment (QoRA) and Quality of Result Suggestions (QoRS) since both provide information that can be used to achieve timing closure.
- Like Design Analysis Report – Run initially after doing the Opt.

1. Overall Assessment Summary

QoR Assessment Score	2 - Implementation may complete. Timing will not meet
Report Methodology Severity	Critical warnings
ML Strategy Compatible	Yes
Incremental Compatible	No
Next Recommended Flow Stage	Review methodology warnings and fix or waive them

3. Methodology Check Details

ID	Description	Criticality	Number of Violations
TIMING-6	No common primary clock between related clocks	Critical Warning	2
TIMING-7	No common node between related clocks	Critical Warning	2
TIMING-8	No common period between related clocks	Critical Warning	2
TIMING-9	Unknown CDC Logic	Warning	1

AMD Vivado™ – Quality of Results

SCORE	MEANING	CORRECTIVE ACTION
1	Design will not implement	Redesign RTL / HLS modules
2	Design will implement timing problems	Review constraints & RTL HLS
3	Design Runs have a small chance of success	Use QoR suggestions, review clocking, ML strategies
4	Design should meet timing if directives used	Use QoR suggestions, ML strategies
5	Design will implement without timing issues	Run Implementation

AMD Vivado™ – Quality of Results

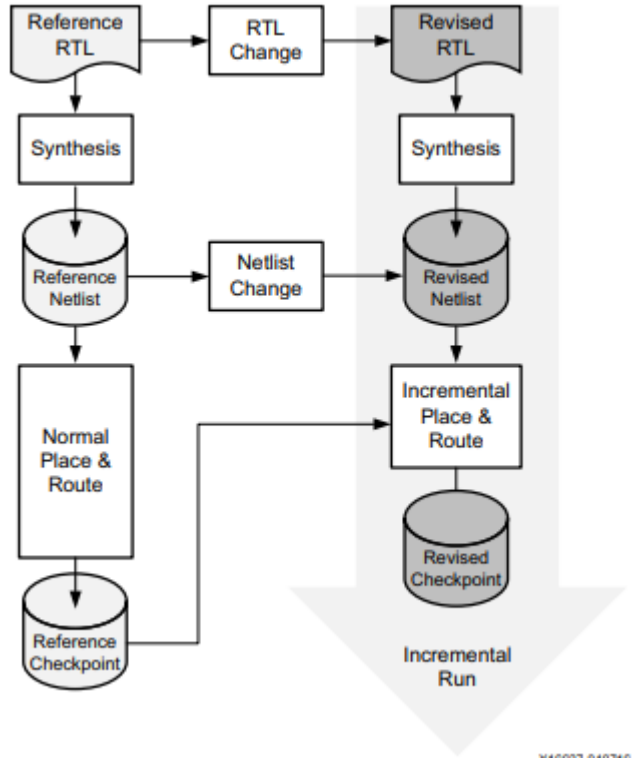
```
Copyright 1986-2020 Xilinx, Inc. All Rights Reserved.
------------------------------------------------------------------------------------
| Tool Version : Vivado v.2020.1 (win64) Build 2902540 Wed May 27 19:54:49 MDT 2020
| Date         : Sat Aug 29 14:13:22 2020
| Host         : DESKTOP-L30MJC1 running 64-bit major release  (build 9200)
| Command      : report_qor_suggestions -file QOR_Suggestions.txt
| Design       : design_1_wrapper
| Device       : xczu9eg
| Design State : Fully Placed
| ML Models    : v2019.2.0
------------------------------------------------------------------------------------

Report QoR Suggestions

Table of Contents
-----------------
1. QoR Suggestions Report Summary
2. ML Strategies
3. QoR Suggestions - XDC

1. QoR Suggestions Report Summary
---------------------------------

+-------------+-----------+-----------+-------------+---------------+-----------+--------------------+-----------------------------------------------------------------------------------------------------+-------------+
|     Name    |     Id    |   Status  | Generated At | Applicable For | Automatic | Incremental Friendly |                                                Description                                          |    Source   |
+-------------+-----------+-----------+-------------+---------------+-----------+--------------------+-----------------------------------------------------------------------------------------------------+-------------+
| RQS_XDC-3-1 | RQS_XDC-3 | Generated | place_design | synth_design   | No        | No                 | Tight constraints for given unsafe paths. Fix unsafe paths by amending the design or                | Current Run |
|             |           |           |             |               |           |                    | adding false path, datapath only, or clock group constraints.                                       |             |
+-------------+-----------+-----------+-------------+---------------+-----------+--------------------+-----------------------------------------------------------------------------------------------------+-------------+
* By default the number of failing paths is limited to 100. Use the -max_paths options to override.
** The design checks report may change until design is completely implemented/routed


2. ML Strategies
----------------

+---+----+---------+---------+
| # | Id | Command | Options |
+---+----+---------+---------+
* ML Strategies are available only in default/explore at successfully routed design.
```

Reducing Run Time

ADIUVO
ENGINEERING AND TRAINING, LTD.

Design Check Points

- AMD Vivado™ uses a physical design database to store placement and routing information.
- Design checkpoint files (.dcp) allow you to save and restore this physical database at key points in the design flow.



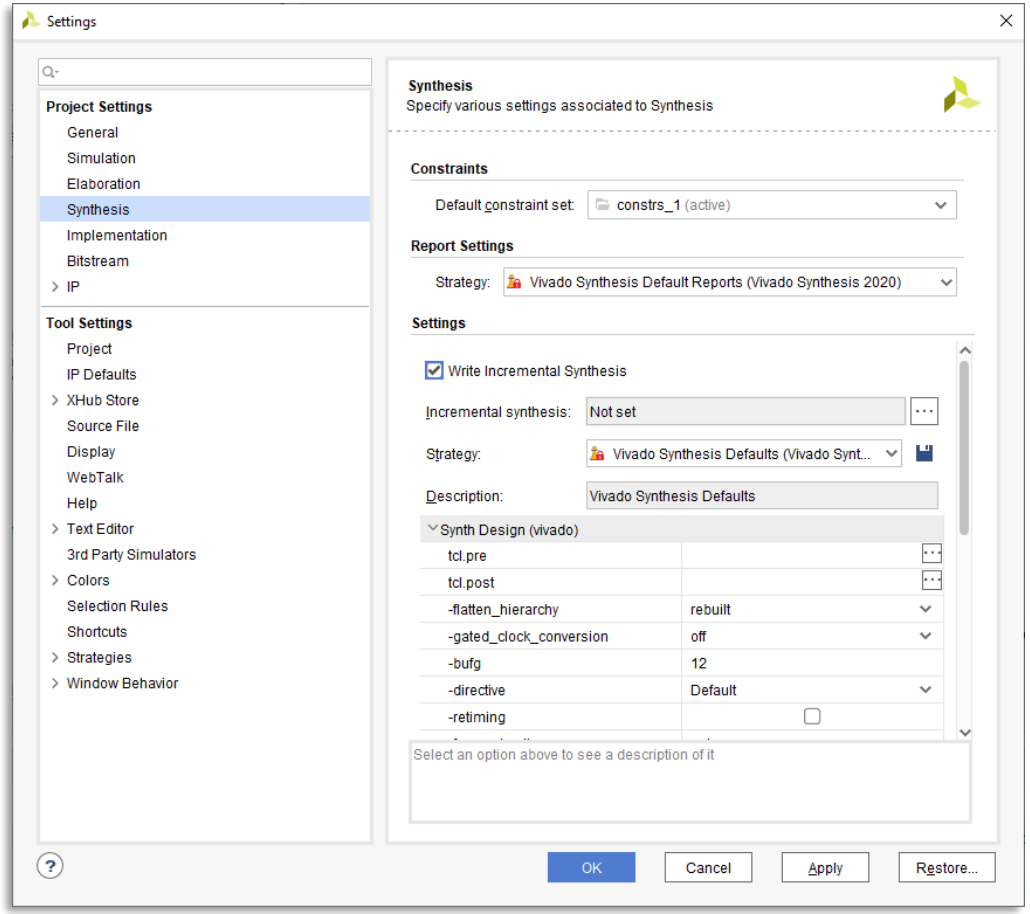
AMD Vivado™ – Reducing Compile Time

- Implementing a FPGA design can take a considerable time > 1 Hour.
- Iterating the design can therefore be an issue, there are several options which can reduce the implementation time both in synthesis and place and route.

Synthesis	
Global	Performs a traditional top-down synthesis of the entire design. Selecting this option takes the longest time because you need to re-run the entire synthesis every time you make a change.
Out of Context Per IP	Runs synthesis and creates a Design Check Point (DCP) for every individual IP block within your design. These check points are then collected into a black-box at the top-level implementation. Using this option means that only the blocks you change need to be re-synthesized, which saves time. OOC-IP also creates an IP customization file (XCI) for each IP block, allowing for customization and OOC XDC files. OOC-IP is the default setting for synthesis within Vivado. This option applies to all IP within the block diagram.
Out of Context Per Block Diagram	Like the OOC-IP option however this option allows you to define the entire block diagram as OOC.

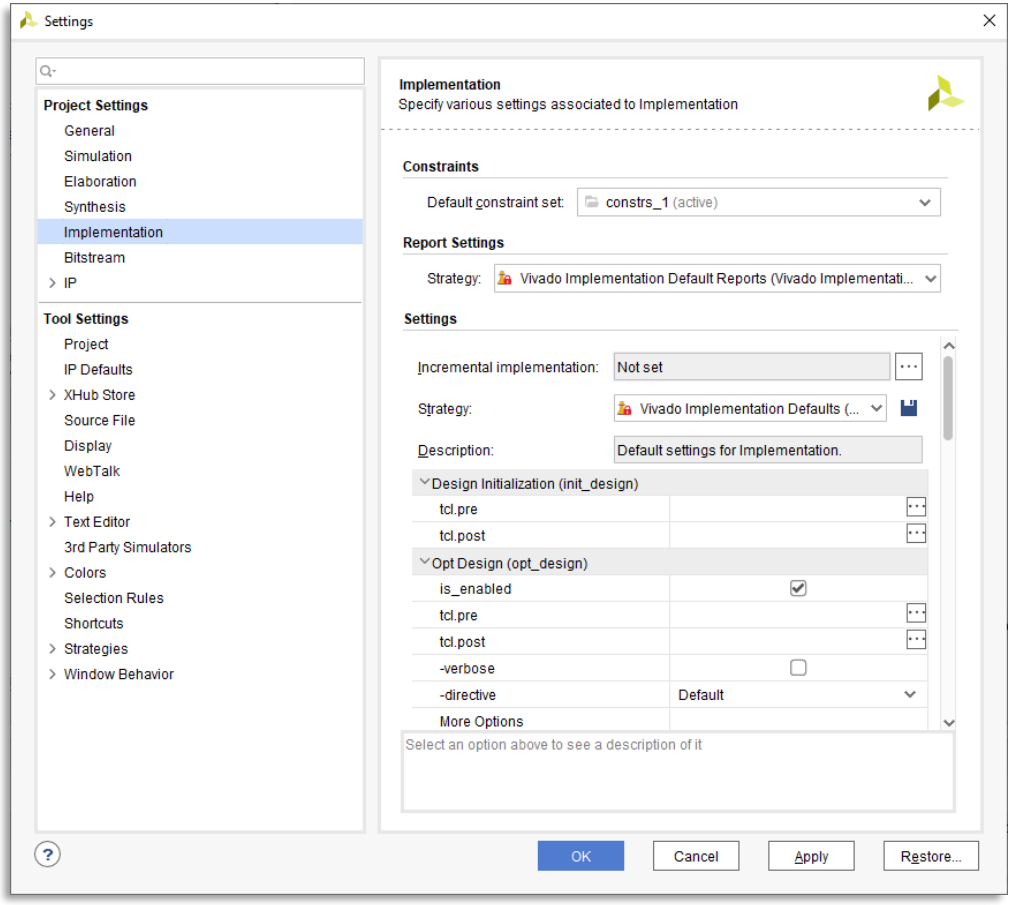
AMD Vivado™ – Reducing Compile Time

- Incremental synthesis can also be used when the changes are small.
- Write out incremental synthesis to the post synthesis design check point.
- Selecting incremental synthesis then provides two options:
  - Automatically use previous DCP
  - Use a defined DCP



AMD Vivado™ – Reducing Compile Time

- Incremental implementation allows use of Design Check Point as the starting point.
- Preserves QoR predictability by reusing prior placement and routing from a reference design.
- Speeds up place and route run time or attempts last mile timing closure.



ADIUVO

ENGINEERING AND TRAINING, LTD.

www.adiuvoengineering.com

adam@adiuvoengineering.com