Designing FPGAs Using the Vivado Design Suite 3

Course Description

You will learn to effectively employ timing closure techniques by applying basic timing constraints, demonstrating timing closure techniques such as baselining, pipelining, and synchronization circuits, showing optimum HDL coding techniques that help with design timing closure, using synchronous design techniques, Reviewing, and analyzing timing reports for a design, employing advanced implementation options, using Tcl scripting in non-project batch flows.

This course builds further on the previous Designing FPGAs Using the Vivado Design Suite 2.

Level: FPGA 3

Training Duration: 3 days

Who Should Attend? – FPGA designers with intermediate knowledge of HDL and FPGA architecture and some experience with the Vivado Design Suite.

Prerequisites:

- Intermediate HDL knowledge (VHDL or Verilog)
- Solid digital design background
- Designing FPGAs Using the Vivado Design Suite 2 course (recommended

Software Tools

- Vivado Design Suite 2021.1

Skills Gained: **After completing this training, you will have the necessary skills to:**

- Apply timing exception constraints in a design as part of the Baselining procedure to fine tune the design
- Apply clock and I/O timing constraints and perform timing analysis. design modifications for source-synchronous and system-synchronous interfaces
- Use the Schematic and Hierarchy viewers to analyze and cross-probe a design
- Define a properly constrained design
- Apply baseline constraints to determine if internal timing paths meet design timing objectives
- Optimize HDL code to maximize the FPGA resources that are inferred and meet performance goals
- Identify synchronous design techniques
- Build resets into your system for optimum reliability and design speed
- Increase performance by utilizing FPGA design techniques
- Use Vivado Design Suite reports and utilities to full advantage, especially the Clock Interaction report
- Perform clocking and static timing analysis (STA)
- Analyze a timing report to identify how to center the clock in the data eye

Course Outline:

1. **Vivado Design Suite Non-Project Based Mode**
   Describes the design flow using non-project batch mode, including using design analysis commands and how constraints are managed in non-project mode
2. **Vivado Design Suite Non-Project Mode**
   Create a design in the Vivado Design Suite non-project mode**.**
3. **Introduction to Clock Constraints**
4. **Generated Clocks**
   Demonstrates using the report clock networks report to determine if there are any generated clocks in a design
5. **I/O Constraints and Virtual Clocks**
6. **Timing Constraints Wizard**

Lab Description

– Basic Design Analysis in the Vivado IDE
Outlines the various design analysis features in the Vivado Design Suite

– Introduction to Clock Constraints
Shows how to apply clock constraints and perform timing analysis.

– I/O Constraints and Virtual Clocks
Covers applying I/O constraints and performing timing analysis

– Timing Constraints Wizard
Reviews how use the Timing Constraints Wizard to apply missing timing constraints in a design

– Introduction to Timing Exceptions
Introduces timing exception constraints and applying them to fine tune design timing

7. **Static Timing Analysis (STA)**
   Describes the clock and its attributes, basics of clock gating, and static timing analysis (STA).
8. **Setup and Hold Violation Analysis**
   Covers what setup and hold slack are and describes how to perform input/output setup and hold analysis
9. **I/O Timing Scenarios**
   Overview of various I/O timing scenarios, such as source- and system-synchronous, direct/MMCM capture, and edge/center aligned data.
10. **System-Synchronous I/O Timing**
    Apply I/O delay constraints and perform static timing analysis for a system-synchronous input interface.
11. **Source-Synchronous I/O Timing**
12. **Timing Constraints Priority**
    Identify the priority of timing constraints.
13. **Timing Closure Using Physical Optimization Techniques**
14. **Case Analysis**
    Understand how to analyze timing when using multiplexed clocks in a design
15. **Synchronous Design Techniques**
    Introduces synchronous design techniques used in an FPGA design.
16. **Resets**
17. **Register Duplication**
    Use register duplication to reduce high fanout nets in a design.
18. **Design Techniques – Baselining**
19. **Design Techniques – Pipelining**
20. **Clock Domain Crossing (CDC) and Synchronization Circuits**
21. **Report Clock Interaction**
    Use the clock interaction report to identify interactions between clock domains
22. **Report Datasheet**
    Use the datasheet report to find the optimal setup and hold margin for an I/O interface
23. **QoR Reports Overview**
24. **Timing Constraints Editor**
    Introduces the timing constraints editor tool to create timing constraints.
25. **Report Clock Networks**
    Use report clock networks to view the primary and generated clocks in a design.
26. **Timing Summary Report**
    Use the post-implementation timing summary report to sign-off criteria for timing closure.
27. **Clock Group Constraints**
    Apply clock group constraints for asynchronous clock domains.
28. **Introduction to Timing Exceptions**
29. **Manipulating Design Properties Using Tcl**
30. **Congestion**
    Identifies congestion and addresses congestion issues

– Source-Synchronous I/O Timing
Apply I/O delay constraints and perform static timing analysis for a source-synchronous, double data rate (DDR) interface.

– Timing Closure Using Physical Optimization Techniques
Use physical optimization techniques for timing closure

– Resets
Investigates the impact of using asynchronous resets in a design.

– Design Techniques – Baselining
Use recommended baselining procedures to progressively meet timing closure

– Design Techniques – Pipelining
Use pipelining to improve design performance

– Clock Domain Crossing (CDC) and Synchronization Circuits
Use synchronization circuits for clock domain crossings

– QoR Reports Overview
Describes what quality of result (QoR) is and how to analyze the QoR reports generated by the Vivado IDE

– Manipulating Design Properties Using Tcl
Query your design and make pin assignments by using various Tcl commands.