



Designing FPGAs Using the Vivado Design Suite 3

FPGA-VDES3 (v1.0)

Course Specification

Course Description

Learn how to effectively employ timing closure techniques.

This course includes:

- Demonstrating timing closure techniques such as baselining, pipelining, and synchronization circuits
- Showing optimum HDL coding techniques that help with design timing closure
- Illustrating the advanced capabilities of the Vivado® logic analyzer to debug a design

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

What's New for 2022.1

- Added information on clock domain crossing (CDC)
- Added lab and module: Intelligent Design Runs (IDR)
- Added lab: Increasing Design Performance Using Report QoR
- All labs have been updated to the latest software versions

Level – FPGA 3

Course Details

- 2 days
 - 21 lectures
 - 14 labs
 - 4 demos

Price –

Course Part Number – FPGA-VDES3

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 1* course (recommended)
- *Designing FPGAs Using the Vivado Design Suite 2* course (recommended)

Optional Videos

- Basic HDL Coding Techniques*
- Power Estimation*

Software Tools

- Vivado Design Suite 2022.1

Hardware

- Architecture: UltraScale™ FPGAs and Versal® ACAPs**
- Demo board: Zynq® UltraScale+™ ZCU104 board

* Go to www.xilinx.com/training and click the Online Training tab to view these videos.

** This course focuses on the UltraScale and Versal architectures. Check with your local Authorized Training Provider for specifics or other customizations.

After completing this comprehensive training, you will have the necessary skills to:

- Employ good alternative design practices to improve design reliability
- 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
- Build a more reliable design that is less vulnerable to metastability problems and requires less design debugging later in the development cycle
- Implement Intelligent Design Runs (IDR) to automate analysis and timing closure for complex designs
- Perform quality of results (QoR) assessments at different stages to improve the QoR score
- Increase performance by utilizing FPGA design techniques
- Use Vivado Design Suite reports and utilities to full advantage, especially the Clock Interaction report
- Describe how to enable remote debug

Course Outline

Day 1

UltraFast Design Methodology (UFDm)

- **UltraFast Design Methodology: Implementation**
Introduces the methodology guidelines covered in this course. {Lecture}

Vivado Tool Flow

- **Vivado Design Suite Non-Project Mode**
Create a design in the Vivado Design Suite non-project mode. {Lecture}

Simulation

- **Timing Simulation**
Simulate the design post-implementation to verify that a design works properly on hardware. {Lecture, Lab}

Design Techniques

- **Baselining**
Use recommended baselining procedures to progressively meet timing closure. {Lecture, Demo, Lab}
- **Pipelining**
Use pipelining to improve design performance. {Lecture, Lab}
- **Inference**
Infer AMD-Xilinx dedicated hardware resources by writing appropriate HDL code. {Lecture, Lab}
- **Sampling and Capturing Data in Multiple Clock Domains**
Overview of debugging a design with multiple clock domains that require multiple ILAs. {Lecture, Lab}
- **Clock Domain Crossing (CDC) and Synchronization Circuits**
Use synchronization circuits for clock domain crossings. {Lecture, Lab, Case Study}

Day 2

Design Runs

- **Intelligent Design Runs (IDR)**
Introduces Intelligent Design Runs (IDR), which are special types of implementation runs that use a complex flow to attempt to close timing. {Lecture, Lab}

Design Analysis

- **Report Clock Interaction**
Use the clock interaction report to identify interactions between clock domains. {Lecture, Demo}



Designing FPGAs Using the Vivado Design Suite 3

FPGA-VDES3 (v1.0)

Course Specification

- **Report Datasheet**
Use the datasheet report to find the optimal setup and hold margin for an I/O interface. {Lecture, Demo}
- **QoR Reports Overview**
Describes what quality of result (QoR) is and how to analyze the QoR reports generated by the Vivado IDE. {Lecture, Lab}

Version Control System

- **Revision Control Systems in the Vivado Design Suite**
Use version control systems with Vivado design flows. {Lecture, Lab}

Power

- **Dynamic Power Estimation Using Vivado Report Power**
Use an SAIF (switching activity interface format) file to determine accurate power consumption for a design. {Lecture, Lab}
- **Versal ACAP: Power and Thermal Solutions**
Discusses the power domains in the Versal ACAP as well as power optimization and analysis techniques. Thermal design challenges are also covered. {Lecture}

Configuration

- **Configuration Modes**
Understand various configuration modes and select the suitable mode for a design. {Lecture}

Debugging

- **Netlist Insertion Debug Probing Flow**
Covers the netlist insertion flow of the debug using the Vivado logic analyzer. {Lecture, Lab}
- **JTAG to AXI Master Core**
Use this debug core to write/read data to/from a peripheral connected to an AXI interface in a system that is running in hardware. {Lecture, Demo}
- **Debug Flow in an IP Integrator Block Design**
Insert the debug cores into IP integrator block designs. {Lecture, Lab}
- **Remote Debugging Using the Vivado Logic Analyzer**
Use the Vivado logic analyzer to configure an FPGA, set up triggering, and view the sampled data from a remote location. {Lecture, Lab}

Tcl Commands

- **Manipulating Design Properties Using Tcl**
Query your design and make pin assignments by using various Tcl commands. {Lecture, Lab}