

## Course Description

Attending the *Designing for Performance* class will help you create more efficient designs. This course can help you fit your design into a smaller FPGA or a lower speed grade for reducing system costs. In addition, by mastering the tools and the design methodologies presented in this course, you will be able to create your design faster, shorten your development time, and lower development costs.

**Level** – FPGA 3

**Course Duration** – 2 days

**Price** – \$1400 or 14 training credits

**Course Part Number** – FPGA23000-13-ILT

**Who Should Attend?** – FPGA designers with intermediate knowledge of HDL and some experience with the Xilinx ISE® software tools

### Prerequisites

- *Essentials of FPGA Design* course or equivalent knowledge of FPGA architecture features; the Xilinx implementation software flow and implementation options; reading timing reports; basic FPGA design techniques; global timing constraints and the Constraints Editor
- Intermediate HDL knowledge (VHDL or Verilog)
- Solid digital design background

### Recommended RELs

- Basic HDL Coding Techniques REL (parts 1 and 2)
- Virtex-6 & Spartan-6 FPGA HDL Coding Techniques REL (parts 1 and 2)
- Power Estimation REL

### Software Tools

- ISE Design Suite: Logic or System Edition 13.1

### Hardware

- Architecture: Spartan®-6 and Virtex-6 FPGAs\*
- Demo board: Spartan-6 FPGA SP605 board\*

\* This course focuses on the Spartan-6 and Virtex-6 architectures. Check with your local Authorized Training Provider for the specifics of the in-class lab board or other customizations.

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

- Describe the architectural features of the Virtex-6 FPGA and Spartan-6 FPGAs
- Identify the different members of the 7 series families
- Create and integrate cores into your design flow by using the CORE Generator™ interface
- Describe the clocking features of the Virtex-6 and Spartan-6 FPGAs and how they can be used to improve performance
- Increase performance by duplicating registers and pipelining
- Increase system reliability by adding an appropriate synchronization circuit
- Describe different synthesis options and how they can improve performance
- Describe a flow for obtaining timing closure
- Pinpoint design bottlenecks by using Timing Analyzer reports
- Apply advanced timing constraints to meet your performance goals
- Use advanced implementation options to increase design performance

## Course Outline

### Day 1

- Review of *Essentials of FPGA Design*
- Designing with FPGA Resources

- CORE Generator Software System
- Basic FPGA Clock Resources
- Virtex-6 and Spartan-6 FPGA Clock Resources
- **Lab 1:** Designing With FPGA Resources
- FPGA Design Techniques
- Synthesis Techniques
- **Lab 2:** Synthesis Techniques

### Day 2

- Achieving Timing Closure
- **Lab 3:** Review of Global Timing Constraints
- Path-Specific Timing Constraints, Part 1
- Path-Specific Timing Constraints, Part 2
- **Lab 4:** Achieving Timing Closure
- Advanced Implementation Options
- **Lab 5:** Designing for Performance
- **Lab 6:** FPGA Editor Demo (optional)
- ChipScope Pro Software (optional)
- **Lab 7:** ChipScope Pro Software (optional)

## Lab Descriptions

- **Lab 1:** Designing with FPGA Resources – Create block RAM and clocking FPGA cores using the CORE Generator™ tool. Instantiate these cores and other clock resources and implement the design.
- **Lab 2:** Synthesis Techniques – Experiment with different synthesis options (including timing constraints, resource sharing, synthesis optimization effort, and register balancing) and view the results.
- **Lab 3:** Review of Global Timing Constraints – Use the Constraints Editor to enter global timing constraints.
- **Lab 4:** Achieving Timing Closure – Review timing reports and enter path-specific timing constraints to fully describe your performance requirements.
- **Lab 5:** Designing for Performance – Improve performance and maximize results solely with implementation options and SmartXplorer.
- **Lab 6:** FPGA Editor Demo (optional) – Use the FPGA Editor to view a design and add a probe to an internal net.
- **Lab 7:** ChipScope Pro Software (optional) – Add an internal logic analyzer to a design to perform real-time debugging.

## Register Today

VAI Logic, the Authorized Training Provider (ATP) for Indiana, Michigan, Ohio, Kentucky, and western Pennsylvania offers public and private training.

Please visit [www.vaitechnology.com](http://www.vaitechnology.com) for more information, to view the current training schedule, or to register.

Please send inquiries to [info@vaitechnology.com](mailto:info@vaitechnology.com), or contact the registrar at (440) 832-7637.

