

## 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** – 1300\$ or 13 Training Credits

**Course Part Number** – FPGA23000-11-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; reading timing reports; basic FPGA design techniques; global timing constraints; and use of the Xilinx Constraints Editor
- Intermediate HDL knowledge (VHDL or Verilog)
- Solid digital design background

#### Recommended RELs

- Basic HDL Coding Techniques REL (parts 1 and 2)
- Spartan-3 FPGA HDL Coding Techniques REL (parts 1 and 2)
- Virtex-5 FPGA HDL Coding Techniques REL (parts 1 and 2)
- Power Estimation REL

#### Software Tools

- ISE Design Suite: System Edition 11.1
- Synplicity Synplify Pro software C-2009.03

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

- Describe a flow for obtaining timing closure
- Describe the architectural features of the Virtex®-5 FPGA
- Describe the features of the Digital Clock Manager (DCM) and Phase-Locked Loop (PLL) 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
- Create and integrate cores into your design flow by using the CORE Generator™ software system
- Run behavioral simulation on an FPGA design that contains cores
- Pinpoint design bottlenecks by using the 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 Virtex-5 FPGA Resources
- CORE Generator Software System
- **Lab 1:** CORE Generator Software System
- Designing Clock Resources

- **Lab 2:** Designing Clock Resources
- FPGA Design Techniques
- Synthesis Techniques
- **Lab 3:** Synthesis Techniques

### Day 2

- Achieving Timing Closure
- **Lab 4:** Review of Global Timing Constraints
- Timing Groups and OFFSET Constraints
- Path-Specific Timing Constraints
- **Lab 5:** Achieving Timing Closure
- Advanced Implementation Options
- **Lab 6:** Designing for Performance
- **Lab 7:** FPGA Editor Demo (optional)
- ChipScope Pro Software (optional)
- **Lab 8:** ChipScope Pro Software (optional)

### Lab Descriptions

- **Lab 1:** CORE Generator Software System – Create a core, instantiate it into VHDL or Verilog source code, and implement the design.
- **Lab 2:** Designing Clock Resources – Use the Clocking Wizard to configure DCMs and global clock buffer resources. Instantiate these resources and implement the design.
- **Lab 3:** Synthesis Techniques – Experiment with different synthesis options (including timing constraints, resource sharing, synthesis optimization effort, and register balancing) and view the results. Versions of this lab are available for the Xilinx XST and Synplify Pro software.
- **Lab 4:** Review of Global Timing Constraints – Use the Constraints Editor to enter global timing constraints.
- **Lab 5:** Achieving Timing Closure – Review timing reports and enter path-specific timing constraints to meet performance goals.
- **Lab 6:** Designing for Performance – Improve performance and maximize results solely with implementation options.
- **Lab 7:** FPGA Editor Demo (optional) – Use the FPGA Editor to view a design and add a probe to an internal net.
- **Lab 8:** ChipScope Pro Software (optional) – Add an internal logic analyzer to a design to perform real-time debugging.

## Register Today

Hardent, the Authorized Training Provider (ATP) for Canada (excluding British Columbia), New England (Connecticut, Maine, Massachusetts, New Hampshire, Rhode Island and Vermont) and the Southeastern United States (Alabama, Florida, Georgia, Mississippi, North Carolina, South Carolina and Tennessee) delivers Xilinx public and private courses in your region. Visit [www.hardent.com/training](http://www.hardent.com/training) or contact Hardent's Training Coordinator for more information, to register for a class or to schedule a private course.

Email: [training@hardent.com](mailto:training@hardent.com)  
Telephone: 514-999-3880

