

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.

Note that one of the prerequisites of *Designing for Performance* is the completion of the HDL coding style modules listed below. Go to www.xilinx.com/education and click the Recorded e-Learning link to view these recorded modules.

Level – Intermediate

Course Duration – 2 days

Price – \$1200 or 12 Xilinx Training Credits

Course Part Number – FPGA23000-10-ILT

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

Prerequisites

- *Fundamentals 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
- 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)

Software Tools

- ISE Foundation™ software 10.1 with the ISE Simulator
- ChipScope™ Pro software
- Synplicity Synplify Pro software 9.2

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

- Describe a flow for obtaining timing closure
- Describe 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
- 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 *Fundamentals 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
- Power Estimation (Optional)
- **Lab 7:** FPGA Editor Demo (Optional)
- ChipScope Pro Software (Optional)
- **Lab 8:** ChipScope Pro Software (Optional)

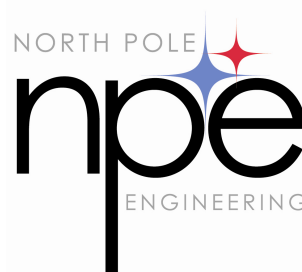
Lab Descriptions

- **Lab 1:** CORE Generator Software System – Create a core, instantiate the core into VHDL or Verilog source code, and run behavioral simulation.
- **Lab 2:** Designing Clock Resources – Use the Clocking Wizard to configure DCMs and global clock buffer resources.
- **Lab 3:** Synthesis Techniques – Experiment with different synthesis options and view the results. Versions of this lab are available for Synplicity Synplify Pro, Precision RTL, and Xilinx XST 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 – Use the FPGA Editor to view a design and add a probe to an internal net.
- **Lab 8:** ChipScope Pro Software – Add an internal logic analyzer to a design to perform real-time debugging.

Register Today

NPE delivers public and private courses in locations throughout the central US region; including Iowa, Illinois, Kansas, Minnesota, Missouri, North Dakota, South Dakota and Wisconsin.

Visit www.npe-inc.com/home/training, for full course schedule and training information.



You must have your tuition payment information available when you enroll. We accept credit cards (Visa, MasterCard) as well as purchase orders and training credits.