

The Quartus II Software Design Series: Foundation

Course Description

You will learn how to use the Quartus® II software v. 9.0 to develop an FPGA or CPLD. You will create a new project, enter in new or existing design files, and compile your design. You will also learn about timing constraints and analyze a design compiled with these constraints using the TimeQuest timing analyzer, the path-based static timing analysis tool included with the Quartus II software. You will learn techniques to help you plan your design. You will employ Quartus II features that can help you achieve design goals faster. You will also learn how to plan and manage I/O assignments for your target device.

The Quartus II Software Design Series: Timing Analysis

Course Description

You will learn how to constrain & analyze a design for timing using the TimeQuest timing analyzer in the Quartus® II software v. 9.0. This includes understanding FPGA timing parameters, writing Synopsys Design Constraint (SDC) files, generating various timing reports in the TimeQuest timing analyzer & applying this knowledge to an FPGA design. Besides learning the basic requirements to ensure that your design meets timing, you will see how the TimeQuest timing analyzer makes it easy to create timing constraints to help you meet those requirements.

The Quartus II Software Design Series: Verification

Course Description

You will learn features of the Quartus® II software v. 9.1 that will enable you to verify your FPGA design*. You will learn how to simulate Altera IP and megafunctions in other EDA simulation tools and how to use NativeLink to simulate directly in the Quartus II software from 3rd-party tools. You will also estimate FPGA power consumption using tools found in the Quartus II software. You will use debugging tools available in the Quartus II software, such as the SignalTap® II embedded logic analyzer, In-System Sources & Probes, & the Logic Analyzer Interface. You will learn to select the correct tool to effectively debug your design. *Some (not all) features examined by this course apply to CPLD designs

The Quartus II Software Design Series: Optimization

Course Description

You will learn advanced features of the Quartus® II design software v.9.1 that will enable you to shorten your design cycle as well as improve your design performance and utilization. You will use the incremental compilation flow and LogicLock™ regions in the Quartus II software to reduce compile times and preserve performance on selected regions of your designs. You will obtain your design goals in the area of performance, resource usage and power consumption by using design strategies, HDL coding styles and Quartus II software settings. You will also learn how to manage compile times effectively.

Designing with the Nios II Processor and SOPC Builder

Course Description

This course will teach you how to design in a soft core embedded processor with an Altera FPGA. This course is focused on the hands-on development of Nios II hardware and software using the Nios II Development Kit. You will learn how to integrate a Nios II 32-bit microprocessor and test it in an Altera FPGA. You will learn how to configure and compile designs using the Quartus II software v. 9.0 and SOPC Builder tool as well as how develop and run embedded software for Nios II in the Nios II IDE. You will participate in discussions about the features and capabilities of the development board along with how to create and test your own custom IP. After taking this course you should feel confident tackling your next SOPC design.

Developing Software for the Nios II Processor

Course Description

This course is targeted at Software Engineers or Developers. You will learn to develop and run embedded software for the Nios II processor in the Nios II IDE and on the Nios II Command Tools. You will also be exposed to a few hardware concepts including how a Nios II 32-bit microprocessor is configured and integrated into an Altera® FPGA using the Quartus® II software v. 8.0 and SOPC Builder design tools. This course utilizes the Nios II Development Kit so that you can download, run, and debug your code in an Altera FPGA. You will participate in discussions about the features and capabilities of the Nios II toolchain, and after taking this course you should feel confident tackling your next embedded programming task for the Nios II processor.

Introduction to VHDL

Course Description

This one-day course is a general introduction to the VHDL language and its use in programmable logic design. The emphasis is on the synthesis constructs of VHDL; however, you will also learn about the simulation constructs. You will gain a basic understanding of VHDL to enable you to begin creating your design file. In the hands-on laboratory sessions, you will put this knowledge to the test by writing simple but practical designs. You will also learn the basic instructions needed for operating both the synthesis and simulation tools of the Quartus® II software v. 7.2.

Introduction to Verilog HDL

Course Description

You will learn how to implement basic constructs and modeling structures in Verilog to create an optimal FPGA design. The emphasis is on the synthesis constructs of Verilog HDL; however, you will also learn about the simulation constructs. You will also learn how to take advantage of various features in Verilog HDL such as delays in programmable logic design. You will gain hands-on experience by implementing various simple but practical designs. You will gain a basic understanding of Verilog HDL to enable you to begin writing designs. In addition you will create a new project, enter in a new design, compile and simulate your design using the Quartus® II software v. 7.2 development tool.

Designing with DSP Builder

Course Description

Learn the FPGA design flow for implementing DSP designs. You will use DSP Builder which is an interface between the Quartus® II software v. 9.0 & Mathworks' Matlab Simulink tools. You will analyze, design, implement, & verify DSP systems using the DSP Builder blockset in Matlab & Simulink. You will increase simulation speed by co-simulating a design with a FPGA board using Hardware in the Loop feature. With a link for ModelSim® feature, you will co-simulate ModelSim RTL-level models from within Simulink. By exploiting Matlab & Simulink interoperability, you will parameterize & verify a DSP Algorithm from the system level. You will incorporate IP MegaCore® cores in your design.

Designing with DSP Builder Advanced Block Set

Course Description

Learn the timing-driven Simulink design flow to implement high speed DSP designs. We'll focus on hands-on development of highly optimized DSP algorithms using the advanced blockset capability of DSP Builder—an interface between Quartus® II software v. 9.0 & Mathwork's Matlab & Simulink tools. You'll analyze & design a DSP algorithm using the DSP Builder advanced blockset in Matlab & Simulink. According to system-level constraints, the timing-driven Simulink synthesis flow will automatically pipeline & time-share hardware. You'll explore architecture & performance tradeoffs with system-level constraints. You'll verify functionality & performance of generated hardware in ModelSim® & Quartus II software. You'll further speed up development time by incorporating ready made ModelIP cores.