

Advanced Verilog HDL Design Techniques



Course Description

In this course, you will learn efficient coding techniques for writing synthesizable Verilog. You will gain experience in behavioral and structural coding while learning how to effectively write common logic functions including registered, memory and arithmetic functions. You will learn how to use Verilog constructs to parameterize your design, increasing their flexibility and reusability. While the concepts presented will mainly be targeting Intel® FPGA devices using the Quartus® Prime software, many can be applied to other synthesis tools as well. You will be introduced to testbenches and Verilog constructs used when building them. The hands-on exercises will use the Quartus Prime software to synthesize Verilog code and the ModelSim®-Intel tool for simulation.

Skills Developed

- Implementing synthesizable sequential and combinatorial RTL code
- Implementing finite state machines using multiple encoding schemes
- Debugging RTL code for common errors
- Developing simple testbenches for verification
- Using the Quartus Prime software to synthesis and verify results
- Running functional simulations in the ModelSim-Intel software

Prerequisites

We recommend completing one of the following courses:

- Introduction to Verilog HDL
- Verilog HDL Basics

Skills Required

- Background in digital logic design
- Understanding of synthesis and simulation processes

Exercises

- How to use IF-ELSE efficiently
- Create a 16-bit up/down counter with a modulus
- State Machine Encoding
- Operator Balancing, Resource Sharing & Pipelining
- Create a testbench to simulate a multiplier design
- Create a self checking testbench to simulate a multiplier design
- Writing Parameterized Code

Course Length	1 day
Language	Presentation in German or English and documentation in English
Platform	PC Windows 10
Pricing	Public: 800,- EUR / attendee In-House: On Request
Dates	See schedule at elcamino.de