FIRSTEDA
ENABLING DESIGN

Contact
+44 (1295) 201250
info@firsteda.com
www.firsteda.com

Advanced VHDL Testbenches & Verification (Remote online course - 10 days)

A thorough introduction and overview of advanced VHDL verification techniques and methodology using OSVVM - 50/50 labs and lecture

Overview

Want to learn the latest VHDL Verification techniques and methodologies for FPGAs and ASICs, including the Open Source VHDL Verification Methodology (OSVVM)?

Through this training, you will gain the knowledge needed to improve your verification productivity and create a VHDL testbench environment that is competitive with other verification languages, such as SystemVerilog UVM. Unlike UVM however, our methodology works with any simulator that supports VHDL-2008, removing the requirement to learn a complex new language or invest in additional costly tools.

Who should attend?

Suitable for digital (FPGA/ASIC/PLD) designers with a requirement to improve verification efficiency and effectiveness.

Prerequisites

Delegates should have a good working knowledge of digital circuits and prior experience of VHDL at work, or through previous VHDL training, such as our own Intermediate VHDL. As this course is delivered live online, each delegate will require access to a personal computer with a reliable and stable internet connection.

Content and structure

You will learn how to create structured transaction-based testbenches using either procedures or models (aka verification IP or transaction level models). Both methods facilitate creation of simple, powerful, and readable tests.

You will also learn about subprogram usage, libraries, file reading and writing, error reporting (Alerts and Affirmations), message handling (logs), abstractions for interface connectivity (records and resolution functions), model synchronisation methods (barrier synchronisation and others), verification data structures (scoreboards and FIFOs), directed, algorithmic, constrained random, coverage driven random test generation, self-checking (result, timing, protocol checking and error injection), functional coverage, representation of analogue values and periodic waveforms (such as triangle or sine waves) and test planning.

The course then progresses onto advanced topics including, modeling multi-threaded models (such as AXI4-Lite), advanced functional coverage, advanced randomisation, creating data structures using protected types and access types, timing and execution, configurations and modeling RAM.

This is exactly the same training and materials as our 5-day public Advanced VHDL Testbenches and Verification course but is instead delivered remotely online and is spread over 10 daily (Monday - Friday) 2.5 hour lecture sessions as outlined below:

Day 1
- Testbench overview
- Basic testbenches
- Transactions and subprograms
- **Lab:** UartTx using subprograms

Day 2
- Modelling for verification
- VHDL and OSVVM IO
- **Lab:** UartTx using subprograms (continued)

Day 3
- **Lab review:** testing w/ subprograms
- Transaction-based models
- Elements of a transaction-based model
- **Lab:** UartTx using transaction-based models

Day 4
- Generating and checking tests (part 1)
- OSVVM library
- **Labs:** Adding error injection to UartTx
  Adding a scoreboard to UartTx

Day 5
- **Lab review:** UartTx BFM
- Constrained random testing
- Functional coverage
- **Labs:** Adding functional coverage
  Adding constrained and intelligent random

Day 6
- Execution and timing issues
- Planning and reuse
- **Labs:** Reuse & subblock testing
  UartRx using models

Day 7
- Advanced modelling w/ AXI-Lite master
- Data structures and protected types
- **Labs:** AXI-Stream (part 1)

Day 8
- Advanced coverage techniques
- Simulation Management and configurations
- **Labs:** AXI-Stream (part 1 continued)

Day 9
- Advanced randomisation techniques
- Generating and checking tests (part 2)
- **Labs:** AXI-Stream (part 2)

Day 10
- Modelling RAM
- Modelling interrupt handling
- Validating self-checking models
- **Labs:** AXI-Stream (part 2 continued)

Approach and labs

This hands-on, how-to course is taught by an experienced verification engineer with specialist VHDL knowledge and practical expertise. Delivered online via live daily interactive group lecture sessions with our instructor. Training materials and software licenses are provided in advance to allow delegates to complete lab exercises between the lecture sessions. Real time one-on-one support and guidance from the trainer is available throughout.

The trainer

Developed and delivered by Jim Lewis, chair of the IEEE VHDL standards working group and chief architect of OSVVM, this training gives you the deep insight into the methodology that is only available directly from its developers.

OSVVM