SUTHERLAND HDL

22805 SW 92nd Place., Tualatin, Oregon 97062 • Phone: +1-503-692-0898 • URL: www.sutherland-hdl.com

SystemVerilog Assertions for Design and Verification Engineers

Overview

SystemVerilog Assertions for Design and Verification Engineers is an advanced workshop covering the IEEE 1800 SystemVerilog Assertions (SVA). SVA enables engineers to verify extremely complex logic using a concise, portable methodology. SystemVerilog Assertions offer improvements at every stage of design and verification process. This workshop provides a thorough examination of SVA and assertion-based verification methodologies. Both immediate and concurrent assertions are presented, with discussion on the appropriate usage of each type of assertion. SVA sequence and property blocks are covered in great detail, with a focus on the semantics and proper usage of the many sequence and property operators. The presentation materials and training guide are filled with practical examples of writing assertions for various types of hardware logic. Topics presented in this comprehensive study on SVA include the use of local variables, property and sequence arguments, multiple thread termination and uniqueness, assertion-based system functions, and using assertions with multi-clock designs and clock domain crossing. Several labs reinforce the principles presented, with forty percent of the class time devoted to hands-on experience.

Course Length: 2-days on-site, 3-days *eTutored™ live*, 5 to 30 days *eTutored™ self-paced*.

Intended Audience and Objectives

This workshop is targeted towards both digital design engineers and digital verification engineers. The workshop is for experienced Verilog engineers. This workshop will enable design and verification engineers to immediately be productive with assertion-based verification methodologies and to write assertions and sequences that describe and verify complex design functionality.

Prerequisite Knowledge (essential)

A working knowledge of SystemVerilog is essential in order to fully benefit from this workshop. In order to fully understand and utilize the concepts presented in this course, students should have first completed the Sutherland HDL ***SystemVerilog Object-Oriented Verification*** course or equivalent.

Included Materials

- Full-color training binder with copies of all lecture slides, lab instructions, and supplemental information. (*eTutored™ self-paced* courses include an eBook instead of a training binder.)
- Lab files, including example solutions that illustrate proper and efficient coding styles.

Software Tools Used

The Aldec *Riviera-Pro™*, Cadence *Incisive™*, Synopsys *VCS™* or the Mentor Graphics *Questa™* simulator can be used for labs.

Workshop Locations

This workshop can be presented on-site at your facilities or as an *eTutored™ live* online class. We also offer several public *eTutored™ live* workshops throughout the year. For more information, please visit www.sutherland-hdl.com, or call us at +1-503-692-0898.

Licensed Training Materials

Sutherland HDL's training materials can be licensed for use in internal training programs. Licensed materials include presentation PowerPoint files, printable PDF files, and lab files. Train-the-trainer services are also provided.

SUTHERLAND HDL

22805 SW 92nd Place., Tualatin, Oregon 97062 • Phone: +1-503-692-0898 • URL: www.sutherland-hdl.com

Syllabus — SystemVerilog Assertions for Design and Verification Engineers

Introduction to SystemVerilog Assertions (SVA)
- A first look at SystemVerilog Assertions
- The traditional design process
- Using SVA in the definition of designs
- Using SVA in the definition of verification
- Using SVA to facilitate coverage metrics
- Naming conventions
- Lab: Running simulations with SVA

Overview of SVA Properties and Sequences
- Immediate and concurrent assertions
- The SVA property construct
- The SVA sequence construct
- When to use properties versus sequences
- Antecedent, consequent and threads
- Assertion, assumption and verification directives
- Lab

Understanding Sequences
- Sequence operators and built-in functions
- Capturing temporal behavior
- Implication operators
- First match operator
- Repetition operators
- Sequence composition operators
- Sequence methods
- Lab

Understanding Properties
- Property declaration syntax
- Using formal arguments
- Local variables in properties
- Clocking events
- Disabling condition
- Property expressions
- Property operators
- Lab

Advanced Properties and Sequences
- Data types in properties and sequences
- Proper use of assertion overlapping
- Chaining implication operators
- Multiple thread termination
- Unbounded ranges in properties
- Lab

SVA System Functions and System Tasks
- Using the $sampled system function
- Using the $past, $fell and $stable system functions
- Vector analysis system functions
- Severity level system functions
- Assertion control system tasks
- Lab

Clocked and Multi-clocked Assertions
- Clock specification for properties and sequences
- Clock resolution
- Using a default clock
- Multiple clocked sequences
- Multiple clocked properties
- Lab

Verification Directives & Verification-based Coverage
- The assert, assume and cover directives
- SVA coverage
- Coverage metrics
- Lab

Binding SVA to Design Blocks
- The SVA bind construct
- Binding to all instances of a module or interface
- Binding to a single instance of a module or interface
- Verifying VHDL models using SVA
- Lab

Assertion Verification Plans
- What goes into an assertion verification plan
- Planning the who, what and where
- Analyzing the design specification
- Final project: Define assertions for a small DSP design