SystemVerilog Assertions (SVA) Training

by

Recognized Verilog & SystemVerilog Guru, Cliff Cummings of Paradigm Works, Inc.

Cliff Cummings is the only Verilog & SystemVerilog Trainer who helped develop every IEEE & Accellera Verilog, Verilog Synthesis and SystemVerilog Standard.

Course Summary

Course Description

The FIFO Assertions Lab used in this course is a student favorite. The lab is a 1-clock, 16-deep FIFO design with a self-checking testbench. Students are given the complete FIFO design with a self-checking testbench; there is only one problem: the FIFO has 8 subtle bugs, and the student will add 13 assertions to test and debug the FIFO.

The objective of the lab is *NOT* to see how good an engineer is at debugging a FIFO design. The objective is to gain experience using assertions and waveform tools to identify and fix design bugs. One or two of the bugs are straightforward, but the remaining six are rather subtle and would be difficult to debug without assertions.

Students are not allowed to fix any of the FIFO code until they have seen an assertion identify the bug. Again, the objective is not just to fix the FIFO design; it is to gain experience with assertion debugging.

Students should NOT get stuck while working on this lab, so at the back of the lab instructions are three helpful pages that: (1) identify the 8 bugs in the FIFO design, (2) show the corrected FIFO code if needed, and (3) show the assertions that Cliff uses to test this FIFO design.

Duration	6 hours
Breakdown	50% Lecture, 50% Lab
Level	Intermediate & Advanced
Prerequisites	SystemVerilog language knowledge
Motivation	In recent years, Cliff has been called on to conduct SystemVerilog Assertion (SVA) training for companies that had previously taken multi-day SVA training, not because the training they received was bad, but because the training they received was too much, and their engineering teams had a hard time remembering all of the SVA options and syntax possibilities. The problem is that engineers use SVA sporadically for a few months on one project, then go many months without needing it again.
Online Details	The course delivery can be in-person or virtual (virtual courses are convenient for both U.S. and non-U.S. engineers).

Course Syllabus

Half Day #1

(1) SVA – SystemVerilog Assertions

- This training details how the SystemVerilog Assertion (SVA) syntax works and how assertions can be used for design and verification. Special macro-techniques are shown to reduce assertion coding effort by up to 80%.
  - What is an assertion? / Who should add assertions?
  - Assertion benefits - bug detection efficiency
  - SystemVerilog assertion types
  - SystemVerilog immediate assertions
  - SystemVerilog concurrent assertions
  - Assert & cover properties & labels
  - Properties and assert property
  - Overlapping & non-overlapping implications
  - Edge testing functions
  - Sequences
  - Vacuous success
  - Property styles
  - Reduced assertion coding effort using macros
  - Macros with default arguments (SystemVerilog-2009 update)
  - Assertion coding style efficiency benchmarks
  - SystemVerilog assertion system functions
  - Sampled value functions
  - Assertion severity tasks
  - Assertion and coverage example of an FSM design (reference material)
  - Binding SVA to an existing model
  - Bind command details and guidelines
  - Immediate assertions for asynchronous control signal testing (reference materials)
  - Bindfiles using immediate assertions (reference materials)
  - LAB: SystemVerilog Assertions with synchronous FIFO design
- Standalone SVA -vs- SystemVerilog Fundamentals SVA Training
  - About 80% of this material is included in the 2-day SystemVerilog Fundamentals Training Course. This standalone SVA course gives students about an extra 20% of SVA material and more time to complete the full lab.