Course Profile

Assertions in UVM

CVC Pvt. Ltd.

VTD Square, 2nd Floor, 183/3 Sarjapura Road, Dommasandra,
Bangalore – 562125. Ph. No. +91-9620209226
http://www.cvcblr.com, info@cvcblr.com

I. CONTENTS

Course Profile

1. Assertions in UVM Flow (ABV UVM)

SVA is an integral part of IEEE-1800 SystemVerilog language focusing on temporal aspect of specification, modeling and verification. SVA allows sophisticated, multi-cycle assertions and functional checks to be embedded in HDL code. SVA also allows such cycle accurate, protocol

checking to exist as independent entities that can be bound to a typical UVC and/or DUT. SVA allows simple HDL Boolean expressions to be composed into complex definitions of design behavior that can be used for assertions, constraints, functional coverage, and debug. Well-written assertions and assumptions can also be used in a formal verification flow with a Model Checker. To demonstrate, with examples, value of SVA in a UVM flow.

2. Class Details:

- Duration: 4-day full time
- Prerequisites: Hands-On UVM, Verification, SystemVerilog Assertions
- Enrolling for a class: Please refer to Registration section.

3. Trainers Profiles

A. Srinivasan Venkataramanan, CTO

http://www.linkedin.com/in/svenka3

- Over 20+ years of experience in VLSI Design & Verification
- Designed, verified and lead several multi-million ASICs in image processing, networking and communication domain
- Worked at **Philips, Intel,** and **Synopsys** in various capacities. Co-authored leading books in the Verification domain.
- Presented papers, tutorials in various conferences, publications and avenues.
- Conducted workshops and trainings on PSL, SVA, SV, VMM, E, ABV, CDV and OOP for Verification
- Holds M.Tech in VLSI Design from prestigious IIT, Delhi.

B. Ajeetha Kumari, CEO AND MD

http://www.linkedin.com/in/ajeetha

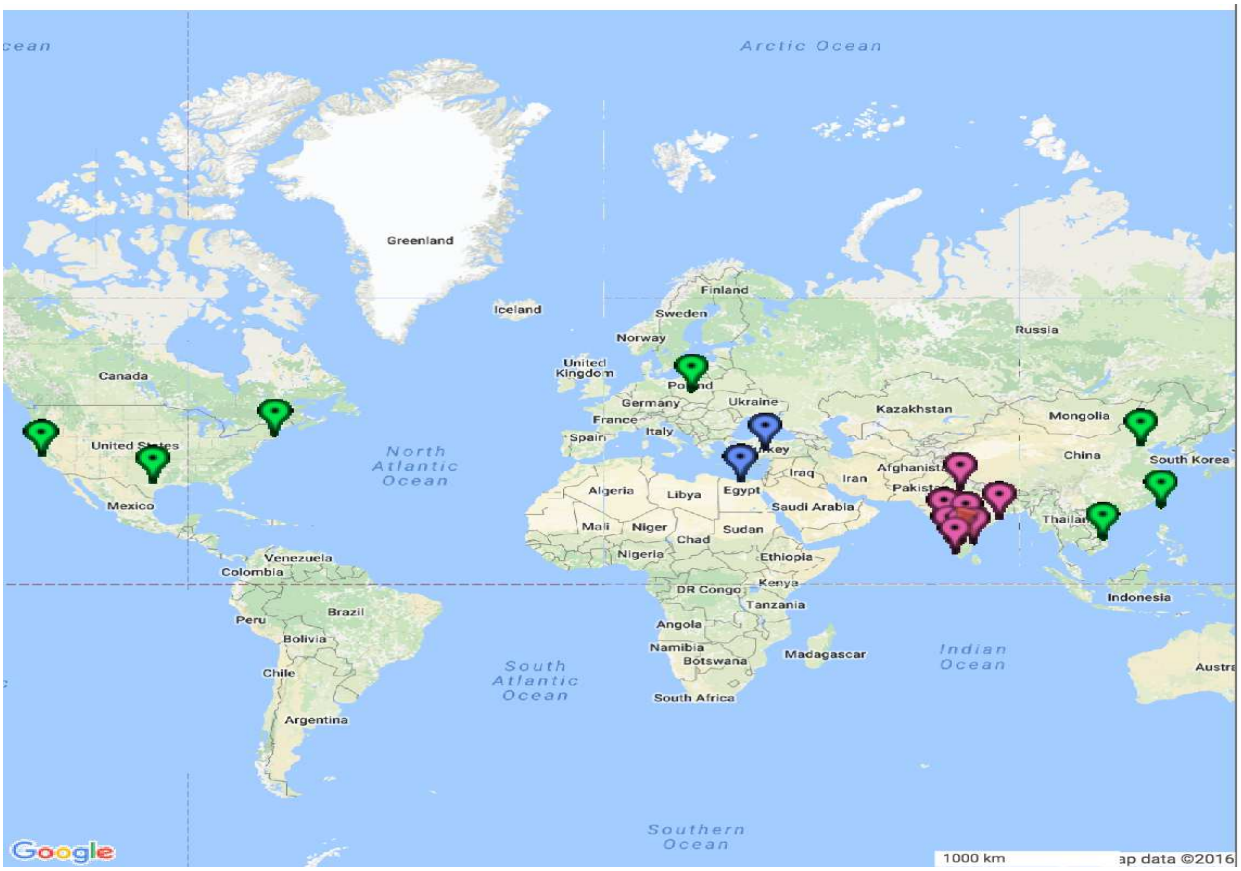
- Has 18+ years of experience in Verification
- Implemented, architected several verification environments for block & subsystems
- Co-authored leading books in the Verification domain.
- Presented papers, tutorials in various conferences, publications and avenues.

- Has worked with all leading edge simulators and formal verification (Model Checking) tools.
- Conducted workshops and trainings on PSL, SVA, SV, OVM, E, ABV, CDV and OOP for Verification
- Holds M.S.E.E. from prestigious IIT, Madras.

4. Why CVC?

Factor / Vendor	CVC	XYZ training company	EDA Vendor
Training Delivery	World renowned experts	Part timers, in bet'n job engineers	Tool support Engineer
Focus	Verification	Language	EDA tools
Topics covered	User/Verification perspective	Language perspective	Based on the tools strength
How Recently Updated	Last week	Months Back	As old as language was standardized
Verification Expertise	Yes	Depends on the trainer	No
Can I run labs across tools	Yes	Yes	No
Is Content Tool independent	Yes	No/Yes (Typically only one tool)	No
Global Footprint	Yes	No	Yes
Publications	Yes	No	No
Post training support	Yes	No	No
Online Technical Evaluation	Yes	No	No
Customization	Yes	No	No
Online Blogs	Yes	No	No
Extended Hands on	Yes	No	No
Code review	Yes	No	No
Architecture Review	Yes	No	No
Productivity Tools	Yes	No	No
Cost	Low	<Unknown>	Expensive

5. Our Global Footprint



6. Other Relevant Courses

- Art of debugging with UVM
- UVM Level 1
- UVM Level 2
- UVM Level 3
- ABV-UVM
- Go2UVM
- Graph Based Verification
- Formal Verification

CVC Pvt. Ltd.

VTD Square, 2nd Floor, 183/3 Sarjapura Road, Dommasandra,
Bangalore – 562125. Ph. No. +91-9620209226
http://www.cvcblr.com, info@cvcblr.com

7. Customer set (sub-set)

CVC Pvt. Ltd.

VTD Square, 2nd Floor, 183/3 Sarjapura Road, Dommasandra,
Bangalore – 562125. Ph. No. +91-9620209226
http://www.cvcblr.com, info@cvcblr.com

8. Course Content

What Is SystemVerilog Assertions?

SVA is an integral part of IEEE-1800 SystemVerilog language focusing on temporal aspect of specification, modeling and verification. SVA allows sophisticated, multi-cycle assertions and functional checks to be embedded in HDL code. SVA also allows such cycle accurate, protocol checking to exist as independent entities that can be bound to a typical UVC and/or DUT. SVA allows simple HDL Boolean expressions to be composed into complex definitions of design behavior that can be used for assertions, constraints, functional coverage, and debug. Well-written assertions and assumptions can also be used in a formal verification flow with a Model Checker.

What Is UVM?

Universal Verification Methodology (UVM) is the industry standard Verification methodology for Verification using SystemVerilog (SV). UVM provides a mean of doing verification in a well-defined and structured way. It is a culmination of well-known ideas, thoughts and best practices. It is also supported by a standard set of base classes to help building structured verification environment faster. More details about the standard can be found at: http://www.go2uvm.org

Agenda

Session 1: Introduction to SystemVerilog and UVM

- What is Universal Verification Methodology? (UVM)
- Simple UVM via Go2UVM package
- Creating test cases using UVM (*uvm_test* and Go2UVM)
- Basic SVA recap (optional)
- Introduction to SystemVerilog Assertion
  - What is an Assertion?
  - What are the types of Assertions?
- Basic Assertion Concepts
  - What are Properties?
  - What are Assertions?
  - What are Cover properties?

- What is Bind directive?
- What are Sequences?

LAB 1 – Simulation of a basic SVA constructs

Session 2: Use Model

- Tunneling SVA messages through UVM logger
- Assertions to verify Driver BFM (APB, AHB-Lite)
- Assertions to verify DUT clock cycle accurate behaviors
- Assertions at block level interfaces for portable checks (VME example)
- Cover properties to capture functional coverage (Cover every bit)
- Immediate asserts in various UVM components
- Gotchas with assert around randomize()
- Assertions for UVM RAL models for post-reset checks
- Using simple UVM test to validate SVAs (via Go2UVM)

LAB 2 – Different Use models

Session 3: Case Study

- AMBA AHB-Lite addresses the requirement of high-performance synthesizable design. It is a bus interface that supports a single bus master and provide high bandwidth operation. In this case study we are going to verify particular behavior of the AHB lite using SystemVerilog assertions.

9. Registration

Send us the following details:

- Name, Email, Contact number of all attendees
- A coordinator name (In case of multiple attendees)
- Training module you are looking for
- Onsite or at CVC premises
- Tentative schedule – month & week (Indicate when your team is available to attend the training)

You may email the details to training@cvcblr.com or

Visit Us: http://www.cvcblr.com or

Call Us: +91-9620209223

CVC Pvt. Ltd.

VTD Square, 2nd Floor, 183/3 Sarjapura Road, Dommasandra,
Bangalore – 562125. Ph. No. +91-9620209226
http://www.cvcblr.com, info@cvcblr.com