Universal Verification Methodology (UVM) Tutorial

Course Title

Universal Verification Methodology (UVM) Tutorial

Course Description

This tutorial presents Universal Verification Methodology (UVM).

As chips grow in complexity and more highly integrated, functional verification requires a more systematic approach to ensure that chips are designed, verified and completed within a reasonable time frame.

UVM is an IEEE standard that defines methods to realize modular, scalable, configurable, generic verification environments. The ultimate goal is improvement of design and verification productivity especially for large complex chips.

The objective of this tutorial is to provide design and verification engineers with technical knowledge of the various features of UVM. It will also enable technical managers to gain a good understanding of UVM.

Target Audience

This course is targeted for those who are involved in digital design verification. This includes chip design engineers, verification engineers and technical managers.

Prerequisite

SystemVerilog, and in particular some knowledge of SystemVerilog classes.

Duration

1 day (9 a.m. - 5 p.m.)

Course Presentation

Each participant will receive a set of course material. Each participant will receive a Certificate of Attendance after completing the tutorial.

Training Venue

On-site training

Course Instructor

Chang Chee Keng, Sys-ASIC Designs Pte Ltd

C.K. Chang, B.S.E.E., M.S.E.E., is a chip designer with over 25 years of experience and has participated in a number of chip development projects. He was a technical staff of AMCC/JNI Corporation (Singapore) and participated in design, verification and architecture of 10Gbit Fibre Channel host adapter ASICs. Prior to that, he founded Digital Media Interconnect Pte Ltd to provide consulting and training services to companies in Singapore and the region. Previous to that, he was a Senior Staff Engineer at Silicon Systems (Singapore) Pte Ltd, a subsidiary of Texas Instruments Inc, and was involved in the design of Firewire storage integrated circuits. Prior to returning to Asia, he worked in Silicon Valley, California, USA, at ESS Technology, 3DO Company and Sun Microsystems in design, verification and architecture of various ASIC development projects.

Contact

Sys-ASIC Designs Pte Ltd
21 Bukit Batok Crescent
#09-79, WCEGA Tower
Singapore 658065
Email: ckchang@sys-asic.com
Website: www.sys-asic.com

Course Outline

SystemVerilog and UVM
History and Goals of UVM
Limitations of Verilog Verification
UVM Big Picture
UVM Class Hierarchy
Components and Transactions
Bridging UVM to Modules
UVM Testbench
UVM Component Topology
Quasi-static and Dynamic Objects
UVM Phase Concepts
Build, Connect, Run Phases, etc.
UVM Component Classes
Environment, Agent, Driver, Monitor, etc.
TLM Connections
Types of TLM Connections
Roles of Ports and Exports
Get and Put Implementations
Sequencer-Driver Connection
Analysis Ports
Multi-Port Scoreboard Connections
UVM Subscriber
UVM Transactions
Classes vs Structs
Transaction Methods
Using Field Automation Macros
User-defined do_methods()
UVM Sequences
Sequencer-Driver Interaction
Basic, Nested and Concurrent Sequences
Virtual Sequence
Virtual Sequencer
UVM Factory
Limitations of Constructor new()
Factory Registration and Methods
Factory Create and Override Methods
Override by Type and Instance
Factory Override Ordering
Configuration Database
Configuration Database Methods
Configuring UVM Components
Using Configuration Objects
Configuration Object Hierarchy
Configuring Sequences
UVM Test Component and Usage
Running Tests
Phase Objection
Objection by Components
Objection by Sequences
Drain Time and Timeout
UVM Reporting
Reporting Macros
Severity, Verbosity and Action
Configuring Verbosity Threshold
Configuring Report Actions
Command Line Report Control
Other UVM Features
UVM Register Layer Overview
UVM Features to be Avoided
General Discussions and Q&A

UVM Tutorial FAQ

What is the main objective of this tutorial?

- The objective of this tutorial is to provide design and verification engineers with technical knowledge of the various features of UVM. This will enable them to jump-start development work sooner and more productively.
- Many engineers start learning UVM from the UVM User's Guide and UVM Class Reference Manual, and soon realize that it is difficult to pick up UVM knowledge, even for those who are already familiar with verification using Verilog and SystemVerilog.
- There is a shortage of examples and they are mainly using code snippets. The explanations of various topics are quite fragmented and often unclear how they are related to one another from a system perspective. The volume of materials is overwhelming, and it is not clear what is for the users and what is predefined by the UVM base classes.
- The UVM User's Guide was written ten years ago when there was not much UVM experience in real chip projects. With more verification experiences in the last ten years, verification experts have come up with best practices and a more standardized approach when using UVM for different test scenarios. The User's Guide does not go into these.

This tutorial will enable engineers to learn the fundamentals of UVM in a much shorter time, without having to spend lots of time reading through the user guide and manuals.

What is the prerequisite to take this tutorial?

The prerequisite is SystemVerilog, and in particular some knowledge of SystemVerilog classes.

Does this tutorial have any lab exercises?

No. However, this tutorial will go through examples of UVM verification code.

I am planning to take a 4-day lab-based training for UVM soon. Is this tutorial still beneficial?

You will gain more from a full lab-based training if you have some upfront preparation prior to attending the training. If you are more prepared, you will be able to understand the presentations more, ask the trainer more in-depth questions, and will more likely to complete the labs. As a result, you will gain more from the full lab-based training.

This tutorial will help you to prepare upfront prior to taking a 4-day UVM training course. In this respect, this tutorial complements the full lab-based training course.

We need to get some UVM work done to meet project schedule, but the public UVM training is scheduled only two months from now. Is this tutorial beneficial?

In the situation where the full training is a few months away, this tutorial is even more relevant.

Instead of spending more time reading the UVM User's Guide and Reference Manuals and still unsure about how things work, it is better to jump-start with a quick knowledge-packed UVM tutorial. You will be able work more productively on the project sooner, while waiting for the public training course to be conducted.

My design team already has an in-house expert in UVM, and we go to this expert when we have questions. Is this tutorial still beneficial?

It is great that your team has an in-house expert. However, this in-house expert is usually someone who has project tasks to work on and a schedule to meet. In this case, you may not always have full access to this person all the time especially when he/she is busy.

This UVM tutorial is prepared with the aim of rapid knowledge transfer and will be presented in a systematic way. The tutorial also has some discussions on current best practices recommended by the verification community (based on discussions on the web and papers).

Therefore, this tutorial complements your in-house expert by providing faster and more systematic upgrading of the knowledge and capabilities of your design team.

My project team is currently using Verilog/VHDL and SystemVerilog for verification, and not using UVM. Is this tutorial still relevant?

As you are aware, more and more companies are using UVM in their chip projects. If you are wondering how UVM verification methodology is different from what you are currently doing, this short technical tutorial can provide the knowledge and answers.

The tutorial will also discuss the types of chip designs where UVM verification is most beneficial. With this knowledge, you will be more current with the latest verification methods, and will have a better idea whether to adopt UVM for your future projects.

Another benefit of understanding UVM is that it will give you some new ideas - borrowed from UVM - on some ways to improve your current verification environment.

I am a technical manager. Is this tutorial beneficial for me?

Even though you may not perform any coding now, this tutorial will provide you with new ideas and insights that will help you better manage your design projects. You will be able to make better decisions on whether to adopt UVM for your design projects. With some knowledge of UVM, you will also be able to communicate better with the engineers doing the technical work.

In addition, this tutorial is given for a group of participants on-site at your company. During the tutorial presentation, there will be opportunities to answer questions that are specific to your design projects.