

# Verification Using SystemVerilog

## What is SystemVerilog?

SystemVerilog is a major extension to Verilog-2001, adding significant new features to Verilog for verification, design and synthesis. Enhancements range from simple enhancements to existing constructs, addition of new language constructs to the inclusion of a complete Object-Oriented paradigm features. There are also considerable improvements in the usability of Verilog for RTL design.

## Overview

CVC's *Verification Using SystemVerilog* course gives you an in-depth introduction to the main enhancements that SystemVerilog offers for testbench development, discussing the benefits and issues with the new features. It also demonstrates how verification is more efficiently and effectively done using SystemVerilog constructs. The course explores in depth verification enhancements such as object-oriented design, constraint random generation, and functional coverage.

## Objectives

- ❖ To explore the new features of SystemVerilog for verification and demonstrate the improvements in verification environment efficiency from their use.
- ❖ To explain key features for verification, such as classes, OOP, randomization, and functional coverage and illustrate how to exploit these features for more efficient verification and testbench development.

## Duration


10 days with labs. We can also offer customized versions of this training onsite or at the location of your choice.

## Prerequisites

Attendees must be familiar with Verilog and ideally, but not essentially, Verilog2001. No prior knowledge of SystemVerilog is required. If you have queries on these prerequisites, please contact CVC.

**1777/42/1, Ground Floor, Gangothri, 26<sup>th</sup> 'B' Main, 40<sup>th</sup> 'A' Cross,  
Jayanagar 9<sup>th</sup> Block, Bangalore - 560069.**

**+91-9880706399, +91-80-26591356**

** [training@noveldv.com](mailto:training@noveldv.com), [cvc.training@gmail.com](mailto:cvc.training@gmail.com)**

1

## ***Enrolling for a class***

Send an email to [training@noveldv.com](mailto:training@noveldv.com), [cvc.training@gmail.com](mailto:cvc.training@gmail.com) with details such as:

Name:

Company Name:

Official Email ID:

Contact Number:

Preferred dates:

If you are coordinating for an entire team, kindly mention how many attendees are expected.


# **Table of Contents**

## **Session 1**

- Verification Introduction
  - ◆ Challenge
  - ◆ Testbench
  - ◆ Types of testbenches
- Elements of a good TB
- Introduction to SystemVerilog
  - ◆ Language evolution
  - ◆ SV Design
  - ◆ SV Assertions
  - ◆ SV testbench
  - ◆ DPI
  - ◆ API
- SV Assertions
  - ◆ Introduction to ABV
  - ◆ Structure of SVA
  - ◆ Compare it to a HDL assertion
  - ◆ Types of assertions
  - ◆ Use model
  - ◆ Checker lib

**1777/42/1, Ground Floor, Gangothri, 26<sup>th</sup> 'B' Main, 40<sup>th</sup> 'A' Cross,  
Jayanagar 9<sup>th</sup> Block, Bangalore - 560069.**

**+91-9880706399, +91-80-26591356**

** [training@noveldv.com](mailto:training@noveldv.com), [cvc.training@gmail.com](mailto:cvc.training@gmail.com)**

2

## **Session 2**


- Abstract modeling constructs
  - ◆ Data types, type checking, type cast
  - ◆ Structure and union
  - ◆ Packages
  - ◆ Enhanced always, case/if... else, loop, flow
  - ◆ Operators
- Arrays and its operators
- SV scheduling semantics

## **Session 3**

- DUT description
- Interface
  - ◆ Grouping signals
  - ◆ Modport
  - ◆ Clocking block, skews
  - ◆ Tasks, functions
  - ◆ Transaction Level Modeling (TLM)
- Class
  - ◆ OOP
  - ◆ cast
  - ◆ inheritance
  - ◆ polymorphism
  - ◆ parameterization
  - ◆ new constructor
  - ◆ Automatic garbage collection
  - ◆ Virtual interface
  - ◆ task and function
    - automatic and static
    - void
    - extern
    - Argument pass by value/reference
- Program construct
- Final block
- Enhanced Concurrency modeling
  - ◆ Threads – variants of fork .. join
  - ◆ Disable fork, terminate
- Inter process communication

**1777/42/1, Ground Floor, Gangothri, 26<sup>th</sup> 'B' Main, 40<sup>th</sup> 'A' Cross,  
Jayanagar 9<sup>th</sup> Block, Bangalore - 560069.**

**+91-9880706399, +91-80-26591356**

** training@noveldv.com, cvc.training@gmail.com**

3

- ◆ semaphore
- ◆ mailboxes
- ◆ queues

## **Session 4**

- Random vs. directed testing
- Need for random testing
  - ◆ Constraints in SVTB
  - ◆ Class constraint
  - ◆ Randomize success / fail
  - ◆ Inheritance
  - ◆ Randomize.with()
  - ◆ Distribution
  - ◆ Function calls in constraints
  - ◆ Array constraints
  - ◆ Pre / post randomize
- Functional coverage
  - ◆ Motivation
  - ◆ Introduction
  - ◆ Types of coverage
  - ◆ Functional coverage process
  - ◆ Covergroup
  - ◆ Coverpoint
  - ◆ Concept of binning
  - ◆ Cross
  - ◆ Sampling event
- DPI
  - ◆ Import
  - ◆ Export
  - ◆ Context
- DPI vs. VPI/PLI