World Class Verilog & SystemVerilog Training

Sunburst Design - Accelerated Introduction To Verilog-2001 & Best Coding Practices

by Recognized Verilog & SystemVerilog Guru, Cliff Cummings of Sunburst Design, Inc.

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

1 Day
90% Lecture, 10% Lab
Accelerated Introduction Level

Course Objective

The primary objective of this course is to jump-start the Verilog knowledge of new users, or to provide Verilog fundamentals for self-taught Verilog users and experienced VHDL users. This course prepares skilled engineers to take Sunburst Design Advanced & Expert Verilog & SystemVerilog training.

Upon completion of this course, students will:

- Understand efficient Verilog-2001 RTL coding styles
  - includes broad understanding of important Verilog syntax concepts
  - includes good Verilog coding techniques
  - includes fundamental Verilog testbench techniques
  - includes Best Coding Practices for Verilog coding

Course Overview

Sunburst Design - Accelerated Introduction to Verilog-2001 & Best Coding Practices is a 1-day fast-paced intensive course on Verilog syntax, usage and best known coding styles.

A detailed 300+ page student guide and 49-page Verilog-2001 HDL Quick Reference Guide supplement the lecture and provide excellent resources for after-class reference.

Target Audience

Sunburst Design - Accelerated Introduction to Verilog-2001 & Best Coding Practices is a 1-day intensive course intended for design and verification engineers who plan to take advanced & expert Verilog & SystemVerilog training and who require an accelerated introduction to the Verilog-2001 RTL syntax & coding styles to meet advanced training prerequisites.

Prerequisites (mandatory)

No prior HDL experience required but knowledge of digital design concepts is strongly recommended

The Sunburst Design - Advantage

Who is teaching your "expert" and "advanced" classes? Most companies will not tell you because their instructors might not have much design experience or may never have participated on any of the Verilog Standards groups or presented at industry recognized conferences. Go to our web site and read about the Sunburst Design - Instructors - they are simply the best at what they do and they have the experience and qualifications to offer best-in-class training.

Sunburst Design Courses:

- *Sunburst Design - Advanced SystemVerilog for Design & Verification* - 4 days
  - *Sunburst Design - Advanced SystemVerilog for Design* - 3 days
  - *Sunburst Design - Advanced SystemVerilog for Verification* - 3 days
- *Sunburst Design - Expert Verilog-2001 for Synthesis & Verification* - 4 days
  - *Sunburst Design - Expert Verilog-2001 & Coding for RTL Design & Synthesis* - 2 days
  - *Sunburst Design - Expert Verilog-2001 Design, RTL Synthesis & Verification Techniques* - 2 days
- *Sunburst Design - Comprehensive Verilog-2001 Design & Best Coding Practices* - 4 days
  - *Sunburst Design - Introduction to Verilog-2001 & Best Coding Practices* - 2 days
  - *Sunburst Design - Advanced Verilog-2001 Knowledge & Design Practices* - 2 days
  - *Sunburst Design - Accelerated Introduction to Verilog-2001 & Best Known Coding Practices* - 1 day
- *Advanced Verilog PLI Courses* - (taught by a Sunburst Design training partner)

Course Customization? - Sunburst Design courses can be customized to include ***your*** company's coding guidelines or to modify the course for a different audience. Sections can be added or deleted from a course to meet you company's needs.

Course Syllabus

Overview of Verilog Resources

Introduction to Verilog Modeling
- An introduction and overview of major Verilog-2001 modeling basics.

- Modules
- Port and net declarations
- V2K1 ANSI-C style module headers
- Instantiation with positional and named ports
- Procedural blocks: initial & always
- Hierarchy
- Introduction to synthesis design flows
- Power-user guidelines (presented in section 1 - detailed in later sections)

Verilog HDL Syntax & Semantics
- Detailed instruction of important Verilog-2001 (V2K1) language syntax.

- Good formatting = fewer bugs & better documentation
- Comments
- V2K1 attributes
- Identifier names
- Name scopes
- Language tokens
- Numbers and logic values
- Net & variable types
- Scalars & vectors
- Multi-dimensional arrays
- Port declaration styles
- Parameters
- Verilog's 4+ logic values

Continuous Assignments & Operators, Verification & Running Sims
- Detailed discussion of continuous assignments with design examples, followed by an overview of Verilog-2001 operators, also with examples. An introduction to writing Verilog testbenches and running Verilog simulations.

- Continuous assignments
- Procedural continuous assignments (do not use these!)
- Required net declarations
- ?: conditional operator
- Strength basics
- Simulation times, delays, timescales and time formatting
- Writing Verilog testbenches
- Repeat-loop & forever loop

- Important compiler directives
- Display and formatting commands
- System tasks for simulation control
- Using multiple Verilog source files & Verilog command files
- Running a Verilog simulation*
- Lab: basic testbench development

* Course notes are printed with detailed instructions on how to use the major Verilog simulators (NC Verilog, VCS, ModelSim, Questa - note: per Cadence, Verilog-XL does not and will not support Verilog-2001 enhancements).

Continuous Assignments & Operators
- Detailed discussion of continuous assignments with design examples, followed by an overview of Verilog-2001 operators, also with examples.

- Continuous assignments
- Procedural continuous assignments (do not use these!)
- Required net declarations
- ?: conditional operator

Programming Statements & Timescales
- Detailed discussion of blocking and nonblocking assignments, followed by an overview of Verilog-2001 programming statements with examples. This section concludes with a discussion of Verilog timescales and their impact on simulation efficiency.

- Verilog operators - arithmetic, bitwise, logical, unary reduction, more
- Sequential & parallel statement groups
- Blocking & nonblocking assignments (introduced)
- Time delays
- Level-sensitive timing controls
- Edge-sensitive timing controls
- Sensitivity lists (V2K1)
- If-else & case statements
- For, while, repeat & forever loops
- Tasks, functions and automatic (V2K1)
- Rise, fall, min, max delays
- `timescale & $timeformat
- Lab: (optional) `timescale & $timeformat capability and efficiency

File I/O & Usage
- Description of Verilog file I/O commands and usage. Fundamentals of using Verilog file I/O within test environments.

- Verilog-2001 File I/O commands
- Line parsing
- Reading & writing files
- Fundamental self-checking testbench techniques
- Lab: (optional) Verilog-2001 File I/O experimentation

Basic RTL Modeling
- Behavioral & synthesizable coding styles for modeling combinational logic, sequential logic, and memory devices. Includes multiple Verilog-2001 enhancements.

- Sensitivity lists
- Continuous assignments
- Procedural combinational blocks
- V2K1 comma-separated and @* sensitivity lists
- Inertial & transport delays
- Correct methods for adding behavioral timing delays
- Flip-flops and latches
- Synchronous and asynchronous inputs
- Where to add timing delays
- Modeling memories
- Array declarations
- System tasks to load memories
- Preferred coding style for read operations
- Preferred coding style for write operations
- Testing bi-directional ports
- Basic timing constraints
- Lab: model and verify an 8-bit ALU
- Lab: model and verify a barrel shifter
- Lab: (optional) extra labs

Classroom Details
Training is generally conducted at your facilities. For maximum effectiveness, we recommend having one workstation or PC for every two students, with licenses for your preferred Verilog simulator (we often can help provide the simulator and temporary training licenses).

For more information, contact:

Cliff Cummings - cliffc@sunburst-design.com - Sunburst Design, Inc. - 503-641-8446