
**World Class SystemVerilog & UVM Training**

# ***Sunburst Design - Expert SystemVerilog Design & Synthesis***

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.*

**2 Days** ***- (with optional 3rd day)***
**70% Lecture, 30% Lab**
**Advanced Level**

## Course Selection

There are two versions of this course. If engineers need to also learn SystemVerilog, they should choose the 4-day Sunburst Design - Comprehensive SystemVerilog Design & Synthesis course, or they should first take the 2-day Sunburst Design - SystemVerilog Fundamentals course.

## Course Objective

Simply stated, to give engineers *world class* SystemVerilog language, advanced design & synthesis training using award winning materials developed by renowned Verilog & SystemVerilog Guru, Cliff Cummings

**Upon completion of this course, students will:**

- Write efficient synthesizable SystemVerilog RTL models
  - includes new SystemVerilog data types and capabilities
  - includes new SystemVerilog RTL and abstraction capabilities
  - includes six different FSM coding styles
  - includes multi-clock and FIFO design techniques
- Gain exposure to new SystemVerilog modeling capabilities
  - includes use of dynamic types and arrays for behavioral modeling
  - includes inclusion of C-models using the new SystemVerilog DPI

## Course Overview

*Sunburst Design - Expert SystemVerilog Design & Synthesis* is a 2-day fast-paced intensive course that focuses on proven and new SystemVerilog features for design and synthesis. Efficient and proven coding styles are combined with frequent exercises and synthesis labs to demonstrate the capabilities of new SystemVerilog features.

**For more information, contact:**
Cliff Cummings - cliffc@sunburst-design.com - Sunburst Design, Inc. - 801-960-1996

This SystemVerilog training was developed and is frequently updated by the renowned SystemVerilog guru and IEEE SystemVerilog committee member, Cliff Cummings. Cliff has presented at numerous SystemVerilog seminars and training classes world wide, including the 2003-2004 SystemVerilog NOW! Seminars, the 2010 ModelSim SystemVerilog Assertion Based Verification Seminars, and multiple Verification Academy DAC seminars.

The 900+ page binder and 140+ page lab guide for this 2-day course covers all of the important SystemVerilog coding styles for RTL & behavioral design. These materials are constantly being updated with the latest clarifications and corrections passed by the IEEE SystemVerilog committee, of which Cliff is an active participant. Numerous proven usage guidelines are taught and explained.

**Target Audience**

*Sunburst Design - Expert SystemVerilog Design & Synthesis* is intended for design engineers who require in-depth knowledge on the IEEE SystemVerilog standard with an emphasis on the new RTL, design & synthesis capabilities.

**Prerequisites (mandatory)**

***This is a very advanced SystemVerilog class that assumes engineers already have a good working knowledge of the Verilog language.***

This course assumes that students have a practical working knowledge of Verilog and SystemVerilog.

**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 or SystemVerilog Standards groups or presented at industry recognized conferences. Go to our web site and read about the Sunburst Design - Instructors - they are the best and they have the experience and qualifications to offer best-in-class training.

**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.

**Classroom Details**

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

**For more information, contact:**
Cliff Cummings - cliffc@sunburst-design.com - Sunburst Design, Inc. - 801-960-1996

## Course Syllabus

### Day One

**(1a) SystemVerilog Enhancements & Methodology Overview**
- Includes a quick review of SystemVerilog resources available to design & verification engineers.

- Verilog & SystemVerilog Keywords
- SystemVerilog Books & Resources
- SystemVerilog Enhancements Strategy & High-Level Methodology

**(1b) Configurations** ***(Optional - included for engineering teams that want to include advanced configuration training)***
- Important Verilog basics that are not generally understood by most Verilog users. Configurations introduced by Verilog-2001. Common useful command lines switches, project directory structures, and commands for conditionally compiled SystemVerilog designs and testbenches.

- Essential Verilog basics
- Working with project directory structures
- Verilog/SystemVerilog command switches
- Conditional compilation for design and verification
- Configuration files

**(2) SystemVerilog Operators, Loops, Jumps. New Logic-Specific Processes. Enhanced functions & tasks. New timing controls** ***(Included in SystemVerilog Fundamentals Training)***
- Includes a review of SystemVerilog enhancements used in RTL synthesis New always_type blocks show design intent and help ensure construction of proper hardware designs. Useful enhancements were added to tasks and functions for RTL designs.

- Logic specific processes (always_type blocks) document designer intent
- always_comb
- always_latch
- always_ff
- Added design checks using always_type blocks
- always @* -vs- always_comb
- void functions
- always_comb & void functions
- Combinational sensitivity
- Design encapsulation through void functions
- Review of SystemVerilog enhancements to tasks & functions
- LABS: simple SystemVerilog combinational and sequential logic labs

**For more information, contact:**
Cliff Cummings - cliffc@sunburst-design.com - Sunburst Design, Inc. - 801-960-1996

**(3) Latches & Priority Encoders. Unique & Priority**
- Detailed description of two synthesis problem areas: latches and priority encoders. Detailed description of the synthesis directives "full_case" and "parallel_case", and why they should generally be avoided. Unique and priority are SystemVerilog replacements for the dangerous "Evil Twins," full_case parallel_case.

- always_latch
- always blocks & sensitivity lists
- Generating latches
- Generating priority encoders
- Latch & priority encoder guidelines
- full_case parallel_case, "the Evil Twins"
- What is full_case? What is parallel_case?
- unique & priority case
- unique & priority if
- unique0 (SystemVerilog-2009 enhancement)
- One example using case modifiers (+ 2 more reference examples)

**(4) Combinational Logic I**
- RTL coding styles for combinational logic, including problems and inefficiencies that arise from poor coding styles. Numerous combinational labs demonstrate many potential problem areas related to common combinational coding styles.

- Introduction to synthesis design flows
- Continuous assignments
- always blocks
- V2K1 @* and comma-separated sensitivity lists
- always_comb
- Instantiated library elements
- Instantiated primitives
- Labs: SystemVerilog combinational labs I

**For more information, contact:**
Cliff Cummings - cliffc@sunburst-design.com - Sunburst Design, Inc. - 801-960-1996

**Day Two**

**(5) Combinational Labs Review I**
- An in-depth review of all the coding styles used in the combinational labs and the synthesized results. Conclusions are drawn about which coding styles infer the most efficient logic implementations.

- Review of Day-One labs

**(6) Nonblocking Assignments, Race Conditions & SystemVerilog Event Scheduling**
- SystemVerilog is fully backward compatible with Verilog-2001 (it is also fully race backward compatible!) This section describes in detail how the new SystemVerilog event scheduling works and how it will reduce race conditions between RTL designs and verification suites.

- Verillog-2001 Event Scheduling
- 8 guidelines for RTL coding & nonblocking assignments
- SystemVerilog enhanced scheduling - includes IEEE 1800 latest updates
- Verilog -vs- SystemVerilog race conditions
- Scheduling of new SystemVerilog commands
- * Blocking & Nonblocking Assignment Details *(* reference materials)*
- * Mixed RTL & Gate simulations *(* reference materials)*

**(7) Combinational Logic II**
- RTL coding styles for combinational logic, including problems and inefficiencies that arise from poor coding styles. Includes Verilog-2001 (V2K1) combinational logic enhancements. Numerous combinational labs demonstrate many potential problem areas related to common combinational coding styles. Additional RTL coding styles for combinational logic, including more problems and inefficiencies that occur from poor coding styles. Verilog-2001 (V2K1) enhancements are discussed including reasons to avoid over-usage of generate statements. Example of poor usage includes I/O pad instantiation (use the more concise and better supported Array of Instance). More combinational labs demonstrate many potential problem areas related to common combinational coding styles.

- Synthesizable and non-synthesizable Verilog constructs
- Bitwise -vs- logical operators
- Tasks & functions
- Tri-state drivers
- Bi-directional busses
- Instantiating ASIC/FPGA library primitives
- Guidelines

**For more information, contact:**
Cliff Cummings - cliffc@sunburst-design.com - Sunburst Design, Inc. - 801-960-1996

**(8) IEEE Verilog 2001 Enhancements**
- A concise summary of the important IEEE Verilog-2001 design enhancements.

- The V2K1 top-5 enhancement requests
- V2K1 Multi-dimensional arrays
- Array of Instances (AOI) & V2K1 generate statements
- V2K1 reentrant tasks and functions
- V2K1 configurations
- New V2K1 port and parameter styles
- V2K1 sensitivity lists
- V2K1 RTL enhancements
- Guidelines
- Labs: SystemVerilog combinational labs II

**(9) Sequential Logic**
- This section covers coding styles for sequential logic. Inferring efficient designs using adders and other large resources is also detailed. Also discusses and includes advantages and disadvantages of instantiation.

- always_ff
- Edge-sensitive sensitivity list
- Basic asynchronous & synchronous resets
- Additional flip-flop coding styles
- Simulation/synthesis differences
- Simulation efficiency
- Register banks
- Memories
- Instantiating Blocks
- Resource sharing

**(10) Synchronous & Asynchronous Reset Design**
- Detailed material for selection and usage of synchronous and asynchronous reset design taken from actual design experiences.

- Synchronous vs. asynchronous resets
- Reset removal metastability
- Asynchronous reset synchronizer circuitry
- Reset distribution trees and techniques
- Multiple clock domains reset synchronization
- LABS: SystemVerilog sequential logic labs

**For more information, contact:**
Cliff Cummings - cliffc@sunburst-design.com - Sunburst Design, Inc. - 801-960-1996

***Day Three (Optional)***

**SystemVerilog FSM Design Techniques** ***(Included in SystemVerilog Fundamentals Training but can be added as a half-day, Day-3 topic)***
- Six different FSM coding styles, enhanced with new SystemVerilog constructs, are detailed and compared for coding and synthesis efficiency. Multiple FSM designs are benchmarked for coding style efficiency.

- FSM coding goals
- Moore & Mealy
- Binary & Onehot
- ASIC -vs- FPGA FSM design
- Review proven FSM coding styles
- One always block - avoid this
- Two always blocks - recommended
- Three always blocks - recommended
- Onehot case(1'b1) - recommended
- Onehot parameters - avoid this
- Output encoded - recommended
- Coding & synthesis efficiency
- Verilog-2001 FSM enhancements
- SystemVerilog FSM enhancements
- Advanced enumerated types
- LABS: SystemVerilog FSM design labs

**Multi-clock Clock Domain Crossing (CDC) & FIFO Design Techniques using SystemVerilog** ***(Included in Expert Clock Domain Crossing (CDC) & FIFO Design Techniques using SystemVerilog Training class, but can be partially added as a half-day, Day-3 topic)***
- Very advanced design techniques from Cliff's award-winning presentations on the efficient implementation of multi-clock CDC & FIFO designs. These materials are not specific to SystemVerilog but solutions are shown using SystemVerilog syntax (advanced techniques that all design engineers should know - the stuff you did not learn in college).

See syllabus at:
**XXX**

**For more information, contact:**
Cliff Cummings - cliffc@sunburst-design.com - Sunburst Design, Inc. - 801-960-1996