Do you want to reduce the design time for your next Lattice FPGA? Learn how to apply the power of the Verilog language in the Lattice design flow. Make the transition from basic digital design concepts to Verilog constructs. Numerous examples will cover various modeling styles for synthesis and verification, enhancing performance and fine tuning the FPGA. This comprehensive course is appropriate for beginner to intermediate designers.

- Hands-on Structure: 50% Lecture & 50% Lab
- Learn which Verilog constructs to use for coding scenarios
- Understand nuances of syntax and semantics
- Envision the synthesized logic expected from the HDL code
- Take designs through the entire Lattice FPGA flow
- Use advanced Lattice features: clock constraints, floor planning
- Write Verilog code to meet timing and utilization constraints

Day 1:
Verilog Design Flow with ispLEVER
- How HDLs Work
- Simulation and synthesis tools
- Lattice FPGA implementation tools
- FPGA flow, Verilog code to bit file

Elements of Verilog Syntax
- Literal Bit Vectors
- Data Types: Variables vs Nets
- Assignments and Expressions
- Parameters: Symbolic Names

Verilog Operators
- Concatenate and Replicate
- Bitwise Operators
- Arithmetic Operators
- Shift and Rotate Operations

Dataflow Coding
- Anatomy of continuous assign
- Describing glue logic
- More complex dataflow code

Day 2:
Structural Verilog Coding
- Instantiating sub-modules
- Using Lattice IPexpress
- Data types for declaring ports
- Probing via hierarchical pathnames

Writing and Running Testbenches
- Instantiate a design under verification
- Effective stimulus generation code
- Monitoring response via waveforms
- Event-driven simulation

RTL Coding for Combinational Logic
- Algorithmic description of hardware
- Anatomy of an always construct
- Controlling execution of algorithms
- Nuances of if-else, case(z), for, while
- Understanding priority in if-else and case

RTL Coding for Sequential Logic
- Edge Triggered Code
- Non-blocking Syntax
- Resets (asynchronous vs. synchronous)

Day 3:
Enhancing Performance
- Constraining the Clock
- Controlling Clock Enables
- Using IO FFs
- Coding for Speed
- Controlling Fanout

Fine Tuning FPGA Device
- Speed Grades
- I/O Placement
- Hold Time Constraints
- Floorplanning Groups
- Timing Closure Strategies

TM Associates has specialized in HDL training for over 12 years and has trained over 5,000 people. To register for the April 1-3 class in San Jose, contact Tom Wille at TM Associates: