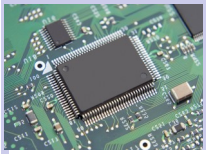

Digital Design Concepts LLC
Verilog Level 1 Online Training Description

Course Description, (doc version = 1.0.2)

The Verilog Level 1 Online Training Course is a 16 hour course that is taught with a live instructor. At the conclusion of the course, students will have the ability to design and simulate digital logic circuits using the Verilog language and commercial EDA tools. Approximately 80% of classroom time consists of teacher instruction, while the other 20% of classroom time is dedicated to exercises. At the conclusion of the course, students will be able to perform the following:

- write efficient code using the latest Verilog constructs, including features of Verilog-2001 and System Verilog
- create sophisticated synchronous designs using Verilog primitives, LSI components, and finite state machines.
- create hierarchical designs
- write testbenches and simulate designs using Modelsim simulator
- write code that follows good programming style

All students will receive a hard copy of the training materials and exercises several days prior to the start of the class. In addition, several reference designs will be available for download.

Course Topics

- Introduction to RTL languages
- The typical digital design workflow
- Verilog basics
- Data types
    - Wires, registers, vectors, integer, real
    - Enumerated types
    - Vectors
    - Arrays
- Combinational logic design
    - Continuous Assignments
    - Operators
- Procedural logic design
    - The always block
    - Sensitivity list
    - Blocking and non-blocking statements
    - Combinational logic using always blocks

- Sequential logic using always blocks
  - Flip Flops
  - If-else statements
  - Case, casex, and casez statements
- Modules and Ports
- Synthesis
  - Characteristics
  - The FPGA logic cell
  - Synthesis optimization techniques
- Case studies -- coding for basic LSI modules
  - Flip/Flop register with reset, clock enable
  - Multiplexer implementations
  - Shift registers
  - Counters
  - Comparators
- Hierarchical design
  - Advantages of hierarchical design
  - Instantiation methods
- Parameters
  - Local parameters
  - Passing parameters through hierarchy
  - Defparams
  - `define
- Finite State Machines
  - Mealy, Moore and mixed type machines
  - Two/Three Process formats
  - Encoding/synthesis strategies
- File I/O
- Testbench concepts
  - Initial and Begin blocks
  - Timescales
  - Tasks
  - Functions
  - $system tasks
- Functional Simulation
  - Compiling designs
  - Simulating designs
  - Verifying the results
- Good design practices
- Future Topics