Verilog Course Plan

Session 1: Introduction to FPGA and Verilog HDL

i. What is FPGA? A slight insight of FPGA
ii. FPGA vs ASIC
iii. What are HDLs?
iv. A brief Introduction to Verilog HDL
v. Abstraction levels
vi. Programming Strategies
vii. Basic Data Types
viii. Number Representation
ix. Operators
x. Ports Declaration
xi. Program Body
xii. A Simple Example

Session 2: Basic Combinational Circuits Coding

i. Review of First Session
ii. Logic Gates
iii. Multiplexer
iv. Encoder
v. Decoder
vi. Ternary Operator (? :)
vii. Binary Adder
viii. Creating hierarchical designs
ix. Comments in Verilog HDL
x. Basic Testbench and Simulation setup
xi. Working with Modelsim
xii. Functions and Tasks in Testbenches

Session 3: Basic Sequential Circuits Coding

i. Review of Second Session
ii. Flip Flops
iii. Synchronous Counters (Up-down, Ring, One-hot)
iv. Shifters (Right, Left, Barrel)
v. ALUs
vi. Writing Testbenches for sequential circuits and simulating in Modelsim
vii. Introduction to Synthesis Tools
viii. Synthesis Flow and Pin Assignments
ix. Bitstream generation and downloading into FPGA
x. Hardware Testing

Session 4: More on Sequential Circuits Coding

i. Review of Third Session
ii. Coding memories (RAM, ROM and FIFO)
iii. Simulating Memories
iv. Introduction to System Verilog
v. Brief introduction to Constraints
vi. Introduction to Available hardware and Hardware Testing

Session 5: FSM Coding

i. Review of Fourth Session
ii. Introduction to FSM coding in Verilog (Mealy and Moore Machines)
iii. Some FSM Examples
iv. Simulating FSMs
v. Writing Generate Statement for Hardware Replication

Session 6: More on Hardware Testing and Debugging

i. Clock Domain Crossing (CDC) Concepts
ii. Timing Optimization and Timing Analysis
iii. Resource Utilization
iv. Post Synthesis and Post Implementation Analysis
v. Signal Tap or ILA for Hardware Debugging

Session 7: Final Session and Project

i. Review of the complete course
ii. Final Project Evaluation and Hardware Testing

PREREQUISITES:

This course assumes no prior experience to programming, but a sound knowledge of Digital Logic Design is required to understand this course. As well as, passion for hardware designing is a must.

NOTE:

1. Each session will be of 2-3 hours with 1 session per week on Saturday/Sunday.
2. The sessions will be online, so access to computer and good network connection is essential to have a seamless learning experience.
3. All links to required software and other resources will be provided as the course progresses.
4. Intel/Xilinx FPGA access will be provided over the internet to have hands-on experience with hardware.
5. Final Project will be decided based on the interest of the students and feasibility of the project.