Technical Examinations Board, Gujarat State, Gandhinagar

Mastering VLSI Verification

Title	SEMI108: Mastering VLSI Verification
Level	Certificate Course
Course Duration	140 Hrs (60 Hrs Theory + 40 Hrs Practical + 40 Hrs Project) Twenty Weeks (Part time) Seven Weeks (Full Time)
Entry Qualification	B.E./B.Tech. Sem-5 onwards (EC / IC / CE / Electrical /Mechanical/ IT / ICT or Similar Branch) / B.Sc./M.Sc. (with Physics/IT/ Electronics) / BCA/MCA / Diploma with knowledge of Basic Ele, Digital Ele, Verilog or VHDL

Teaching Scheme:

Sub Code	Subject Name	Teaching Scheme		Examination Scheme				Term Work Marks	Total Marks
		Theory	Practical	Theory Marks	Hrs.	Practical Marks	Hrs.		
SEMI108	Mastering VLSI Verification	3	4 (With Project)	50	2	100	3	25	175

Total Week	= 20	Theory	= 1 hour slot
Total Teaching slot/Week	= 05	Practical	= 2 hour slot
Theory Periods	= 60	Total teaching 7 hours/week (Part-time) 4 hours/day (Full time)	
Practical Periods	= 40		

VLSI Verification

It is standard verification course which includes basic fundamentals of Electronics and Digital Design. The VLSI verification process is a meticulous examination of a chip's design before it's physically manufactured. It's like a rigorous final exam, ensuring the millions of transistors on the chip collaborate flawlessly. This multi-step process involves creating a virtual model of the chip and feeding it various test scenarios. Engineers analyze the model's behaviour to identify and rectify any errors or bugs in the design logic. Verification techniques like simulations and formal proofs guarantee the chip functions according to its intended specifications, catching potential issues early on to prevent costly mistakes and ensure a reliable final product. This course contains introduction to verification using System Verilog and Universal Verification Methodology which includes concepts of OOPs, polymorphism, random constraint, assertion, sequence, coverage etc.

Course Objectives:

This course aims to providing detailed knowledge in functional verification process which includes Synthesis & Simulation, Verification, TestBench, literals, tasks & functions, interfaces, etc. In this process the student will understand the entire logic process and will be able to take on the challenges posed by the even demanding VLSI verification industry. Student will have exposure to simulation tools being used in industry, will get hands on experience and will be prepared for industry for verification in VLSI.

SEMI108: Mastering VLSI Verification	
Unit - 1	**Introduction to System Verilog**
1.1	ASIC Design System Level Languages, Why SV?, SV Methodology Overview and LRM detail, SV a supper set of Verilog-2001, SV Verification flow overview, TB Architecture, TB Components (Driver, Monitor and Scoreboard)
Unit - 2	**SV Data Types and usage in Design and Verification**
2.1	2/4 state data types, Typedef, unresolved and resolved, SV Array and union
Unit - 3	**Overview of Procedural Block and Control operational flow**
3.1	Procedural Block Flow, Fork Join inter process synchronization, Control Block Flow
Unit - 4	**Overview of Object-Oriented Programming**
4.1	Object Orientation and SV, Syntax and basic example with SV object, Class coding and constructor and, Object (Class instance and usage) with "this", Parameterised class, Simple OOPs based Example
Unit - 5	**SV Class type and extension (Inheritance) with example**
5.1	Static Class and Method, Extend Class – purpose and usage, Scope Resolution Operators
Unit - 6	**SV Polymorphism and other OOPS features**
6.1	Virtual function in base class, Class extended for Polymorphism with example, Override members i.e. override object of sub-class with parent class, Supper, Casting and chaining Data encapsulation etc., Constant and Abstract class, Call Backs (Inserting Call backs, Registering Call backs)
Unit - 7	**SV Random Constraint**
7.1	Constraint Block, Randomise methods, Disable Random constraint, In line constraint random variable, Dynamic Constraints, Scope variable and randomization within class, Random Number generation (urandom, urandom_range, etc.), Seeding and weighed case, Random Sequence
Unit - 8	**SV Interface and Clocking Block**
8.1	interface, modports, Specify block, Parameterised and Virtual interface, Interface object, Example with standard clock generation and data checks, Clocking block, Clocking block basic and input sampling, Example of Clocking blocks with program and interface
Unit - 9	**SV Assertions**
9.1	Immediate/ In-line assertions, Concurrent assertions, Constant Expressions and sequences, System Functions $onehot, $isunknown, etc., Declare Property, Expect sequence
Unit - 10	**Complete SV Verification example code**
10.1	Explaining one code of complete verification using SV
Unit - 11	**UVM Overview**
11.1	UVM hierarchy, UVM Packages, MACRO, Include and SV Interface, Example, Component and environment and test, Binding/Combining UVM Objects
Unit - 12	**UVM Transaction, Analysis and Reporting**
12.1	Transactions, Analysis, Reporting
Unit - 13	**UVM Sequences**
13.1	Basic Sequence, Connecting Sequence to test environment

Unit - 14	**Coverage**
14.1	What is coverage?, Cover points, cross coverage, Bins range in bins, Bins and transaction coverage, Automatic bins, Wildcard in bins values, Ignore bins and illegal bins, Cross Coverage, Binsof and intersect feature, Predefine coverage method, system task and functions, Adding coverage to SV/UVM Example
Unit - 15	**Overview of UVM Test bench Architecture**
Unit - 16	**Advance UVM sequences**
16.1	Case and feeding of sequences, Layered sequence and other sequence
Unit - 17	**Factory and configuration**
Unit 1- 8	**Introduction to TLM**
Unit - 19	**Writing and adding multiple tests**
Unit - 20	**Register Modelling**
Project (40 Hrs)	

Suggested List of Practical's

Sr. No	Practical Name
1	SV Write example code using each of the System Verilog Data Types
2	SV Write example code for a multi-dimensional array, Packed & Unpacked array, dynamic array and associative array
3	SV Write example code for Queue, structure
4	SV Design class for Half adder, Full adder. Make use of class object
5	UVM Simple Memory Testbench
6	UVM sequencer - Driver
7	U VM - DUT Interface, Generator, Driver, Monitor & Scoreboard
8	UVM - Coverage Example
9	Other based on syllabus and Projects

Reference books:

- Chris Spears, System Verilog for Verification, 2nd Edition, Springer, 2008.
- Vijayaraghavan, Srikanth, and Meyyappan Ramanathan. A practical guide for SystemVerilog assertions, Springer Science & Business Media, 2006.
- Bergeron, Janick. Writing testbenches using SystemVerilog, 1st Edition,Springer Science & Business Media, 2007.
- SystemVerilog For Verification: A Guide to Learning the Testbench Language Features by Chris Spear & Greg Tumbush (3rd Edition)
- A Practical Guide For SystemVerilog Assertions by Srikanth Vijayaraghavan & Meyyappan Ramanathan
- SystemVerilog Assertions and Functional Coverage: Guide to Language, Methodology and Applications by Ashok B. Mehta
- SystemVerilog Assertions Handbook by Ben Cohen
- A Practical Guide to Adopting Universal Verification Methodology (UVM) by Sharon Rosenberg & Kathleen A Meade (2nd Edition)
- The UVM Primer: A Step-by-Step Introduction to the Universal Verification Methodology by Ray Salemi

- Getting Started with UVM: A Beginner's Guide by Vanessa R. Copper
- UVM Cookbook by Ray Salemi
- UVM Essentials by Chris Spear and Srinivasan Venkataramanan

Software/Tool list:

- Xilinx Vivado
- EDA Playground
- Cadence xcelium

Subject Course Committee

Prof. C. H. Vithalani, Prof. P. J. Brahmbhatt, Prof. T. P. Chanpura, Prof. M. S. Dave, Prof. P. B. Bhatt, Prof. J. A. Dhumale, Prof. A. S. Patel, Prof. A. K. Konkani