Technical Examinations Board, Gujarat State, Gandhinagar

VLSI Design using Verilog

Title	SEMI101:VLSI Design using Verilog
Level	Certificate Course
Course Duration	90 Hrs (60 Hrs Theory + 30 Hrs Practical) Fifteen Weeks (Part time) Three Weeks (Full Time)
Entry Qualification	B.E./B.Tech Sem-3 onwards (EC / IC / CE / Electrical /Mechanical/ IT / ICT or Similar Branch) / B.Sc./M.Sc. (with Physics/IT/Electronics) /BCA/MCA / Diploma

Teaching Scheme:

Sub Code	Subject Name	Teaching Scheme		Examination Scheme				Term Work Marks	Total Marks
		Theory	Practical	Theory Marks	Hrs.	Practical Marks	Hrs.		
SEMI101	VLSI Design using Verilog	4	2	100	3	50	2	25	175

Total Week	= 15	Theory	= 1 hour slot
Total Teaching slot/Week	= 05	Practical	= 2 hour slot
Theory Periods	= 60	Total teaching	
		6 hours/week (Part-time)	
		6 hours/day (Full time)	
Practical Periods	= 15		

VLSI Design using Verilog

Verilog is a hardware description language (HDL), which is getting standardized as IEEE 1364-2005 standard that is used by designers to design, verification and document electronic systems. The syntax of Verilog is very similar to the C programming language. Verilog is used to design computer chips, processors, CPUs, other peripherals like RAM and similar digital circuitry. Verilog is the most widely used HDL with many active designers using it for designing at various levels of abstraction.

Learning Verilog, a hardware description language commonly used in digital design and electronic design automation, can offer several benefits. Verilog can provide engineering students with a solid foundation in digital design, simulation, and implementation, along with valuable skills that are highly sought after in the electronics industry. It can open up a wide range of career opportunities and equip students with the knowledge and expertise needed to succeed in the field. This course contains introduction to semiconductors, diodes, transistors, MOSFETs their operations , introduction to Verilog, data types, operators, modelling techniques, FSM, coverage etc.

Course Objectives:

This course provides the student with the fundamental skills to understand the basic of electronics and Digital design including diodes, MOSFETS, transistors and their operations, Combinational and sequential logics, etc. This course also provides working knowledge of the Verilog. It explains design, test and implementation of digital hardware, the hierarchy and modelling of structures. It introduces syntax, lexical conventions, data types and memory, Behavioural and register transfer level modelling. It also explains how to write RTL Verilog code for synthesis, concept of delays, test benches, timing checks, etc. 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.

SEMI101:VLSI Design using Verilog	
Unit - 1	**Introduction to VLSI**
1.1	What is VLSI?
1.2	Moore's law
1.3	Scaling
1.4	Industrial Roadmap for Devices & System (IRDS)
1.5	Silicon
Unit - 2	**Semiconductors**
2.1	Insulators, Conductors, Semiconductors
2.2	Silicon Crystal, Band Diagram
2.3	Intrinsic Semiconductors
2.4	Two Types of Flow
2.5	Doping a semiconductor
2.6	Two type of Extrinsic Semiconductors
2.7	Unbiased Diode
2.8	Forward Bias
2.9	Reverse Bias
2.10	Breakdown
Unit - 3	**Bipolar Junction Transistor (BJT)**
3.1	BJT Operation
3.2	BJT voltages and currents
3.3	Common Emitter (CE), Common Base (CB), Common Collector (CC) characteristics
3.4	DC load line and bias point
3.5	Base bias
Unit - 4	**Transistor as switch**
Unit - 5	**Field Effect Transistor (FET) and its biasing**
5.1	Junction Gate FET (JFET)
5.2	Comparison of BJT and JFET
5.3	JFET Characteristics
5.4	Biasing in ohmic and active region
5.5	Transconductance
5.6	Amplification and switching
5.7	Metal Oxide Semiconductor FET (MOSFET)
5.8	CMOS introduction
Unit - 6	**MOSFET Operation and Characteristics**
6.1	MOS Cap Operation
6.2	MOSFET
6.3	NMOS/PMOS

6.4	I-V Characteristics
6.5	Non Ideal Effects
6.6	Threshold Voltage
6.7	Switch
6.8	Capacitance in MOSFET
Unit - 7	**CMOS inverter**
7.1	Inverter
7.2	Layout
7.3	Noise Margin
7.4	R-C Equation of NMOS/PMOS
7.5	CMOS Logic gates
Unit - 8	**Gates**
8.1	CMOS Logic gates
8.2	Logical Effort
8.3	Pass Transistor Logic
Unit - 9	**What is Digital and Analog Systems?**
9.1	Introduction to Digital Electronics and Applications
9.2	Number System and Code Conversion
9.3	BCD, Excess-3, Gray Code
9.4	Complements
9.5	Logic Gates, Boolean Algebra, SOP, POS
9.6	K-Maps
Unit - 10	**Combinational Logic Design**
10.1	Adders & Subtractors
10.2	Comparator
10.3	Multiplexers & De-multiplexers
10.4	Encoder & Decoder
10.5	Parity Checker
Unit - 11	**Sequential Logic Design**
11.1	Difference between Combinational and Sequential circuits
11.2	Flip-Flop & Latch
11.3	Trigger Types - Level Triggered & Edge Triggered
11.4	Setup & Hold Time, Max Delay, Min Delay
11.5	Clocked S-R Flip Flops
11.6	J-K Flip Flops
11.7	D-Flip Flops
11.8	T-Flip Flops
11.9	Master Slave Flip Flops

Unit - 12	**Shift Registers**
12.1	Serial In Serial Out (SISO)
12.2	Serial In Parallel Out (SIPO)
12.3	Parallel In Serial Out (PISO)
12.4	Parallel In Parallel Out (PIPO)
12.5	Universal Shift Register (USR)
Unit - 13	**Counters**
13.1	Asynchronous
13.2	Synchronous
13.3	Mod
13.4	Ring counter
13.5	Johnson (Twisted Ring counter)
Unit - 14	**Finite State Machine (FSM)**
14.1	Mealy & Moore
14.2	Overlapping and Non-Overlapping
Unit - 15	**Memory**
15.1	Random Access Memory (RAM)
15.2	First In First Out (FIFO)
15.3	Read & Write Operations
15.4	Programmable Array Logic (PAL), Programmable Logic Array (PLA), Field Programmable Gate Array (FPGA)
Unit - 16	**Introduction to Verilog**
16.1	Introduction to Verilog, History and evolution of Verilog, modules, ports, & data types, Verilog design flow
16.2	What is HDL?
16.3	Most Commonly used HDLs
16.4	What is Verilog?
16.5	Difference between VHDL and Verilog
16.6	What is usage of Verilog?
16.7	Application Specific IC (ASIC) vs Field Programmable Gate Array (FPGA)
16.8	RTL Design Methodologies
Unit - 17	**Overview of Digital Design with Verilog HDL**
17.1	Hierarchical Modelling Concepts, Top-Down Design, Bottom Down Design, Example, Four Bit Adder (Ripple Carry Counter), Verilog Modules, Instance, Testing Block (Stimulus), Levels of Abstraction
17.2	Basic Concepts, Lexical Conventions, Whitespace, Comments, Operators, Number Specification, Sized numbers, Unsized numbers, X or Z values, Negative Numbers, Identifiers and Keywords, Strings, Identifiers and key words
17.3	Data Types, Value Set, Nets
17.4	Registers, Vectors, Vector Part Select, Variable Vector Part Select

17.5	Integer, Real, and Time Register Data Types, Integer, Real, Time, Arrays, Memories, Parameters, Strings
17.6	System Tasks, displaying information, monitoring information, Stopping and finishing in a simulation
17.7	Compiler Directives, Define, include
Unit - 18	**Modules and Ports**
18.1	Modules
18.2	Ports, List of Ports, Port Declaration
18.3	Port Connection Rules, Inputs, Outputs, Inouts, Width matching, Unconnected ports, Example of illegal port connection
18.4	Connecting Ports to External Signals, connecting by ordered list, Connecting ports by name
18.5	Hierarchical Names
18.6	Gate-Level Modelling, Gate Types, And / Or Gates, Buf / Not Gates, Bufif / notif
18.7	Array of Instances, Examples
18.8	Gate Delays, Rise, Fall, and Tum-off Delays, Rise delay, Fall delay, Tum-off delay
18.9	Min/Typ/Max Values, Min value, Typ val, Max value
Unit - 19	**Data Flow Modelling**
19.1	Continuous Assignments, Implicit Continuous Assignment, Implicit Net Declaration
19.2	Delays, Regular assignment delay, Implicit continuous Assignment delay, Net declaration delay
19.3	Expressions, Operators and Operands, Expressions, Operators, Operands
19.4	Operator types, Arithmetic Operators, Binary operators, Unary operators, Logical operators, Relational operators, Equality operators, Bitwise operators, Reduction operators, Shift operators, Concatenation operators, Replication operators, Condition operators, Operator precedence
Unit - 20	**Behavioural Modelling**
20.1	Structure procedures, Initial statement, Combined variable declaration & initialization, Combined port/data declaration & initialization, always Statement
20.2	Procedural Assignments, Blocking Assignments, Non-blocking Assignments, Application of non-blocking assignments
20.3	Timing controls and event scheduling, Delay-Based Timing Control, Regular delay control, Intra-assignment delay control Zero delay control, zero delay control, Event- Based Timing Control, Regular event control, named event control, Event OR Control, Level-Sensitive Timing Control
20.4	Conditional Statements, Multiway Branching, Case statement, Casex, casez, keywords
20.5	Loops, While loop, For loop, Repeat loop, forever loop
20.6	Sequential and parallel Blocks, Block types, Sequential blocks, Parallel blocks
20.7	Special feature of blocks, Nested blocks, Named blocks, Disabling named blocks
20.8	Generate blocks, Generate loop, Generate conditional, Generate case
Unit - 21	**Tasks and Functions**
21.1	Differences between Tasks and Functions, Task Declaration & Invocation

21.2	Task Examples, Use of input and output arguments, Asymmetric Sequence Generator
21.3	Functions, Function Declaration & Invocation, Function Examples, Parity calculation, Left/right shifter
21.4	Automatic (Recursive) Functions, Constant Functions, Signed Functions
21.5	Useful modelling technique, Procedural Continuous Assignments, assign and de-assign, force and release, force and release on registers, force and release on nets
21.6	Overriding Parameters, defparam Statement, Module Instance Parameter Value
21.7	Conditional Compilation and Execution, Conditional Compilation, Conditional Execution
21.8	Useful system tasks, File output, Opening a file, Writing to files, Closing files
Unit - 22	**FSM, Advanced Verilog & Coding Style**
22.1	Basic FSM structure, Moore Vs Mealy, Common FSM coding styles, Registered outputs
22.2	Advanced Verilog for Verification, System Tasks, Compiler directives, Internal variable monitoring, File input and output
22.3	Synthesis Coding Style, Unwanted latches, Synthesizable operators, RTL coding styles, Synthesis errors
Unit - 23	**Coverage**
23.1	What is Coverage in Verification?
23.2	Why is Coverage Important?
23.3	Code Coverage
23.4	Different types of code coverage, Statement/Branch Coverage, Expression/Condition Coverage, Path Coverage, Toggle Coverage, FSM Coverage
23.5	Functional Coverage
23.6	Exploring Functional Coverage Metrics
23.7	Coverage Analysis
23.8	Merits & Demerits of CC & FC
23.9	Assertion-Based Verification
23.10	Constrained-Random Testing
23.11	Formal Verification
Unit - 24	**Verilog for Synthesis and Verification**
24.1	Verilog Synthesis and Optimization, Overview of synthesis process, Writing synthesizable Verilog code, Optimizing Verilog code for area, speed, and power
24.2	Verification Techniques in Verilog, Introduction to verification methodologies
24.3	Case Studies and Projects, Real-world VLSI design projects using Verilog

Suggested List of Practical's

Sr. No	Practical Name
1	Implementation of Boolean function using logic gates
2	Design and implementation of half and full adders and subtractors
3	Design and implementation of multiplexer and Boolean function using multiplexer

4	Design and implementation of an encoder and decoder
5	Write down a Verilog code example using data types like reg, wire, integer, time, real, arrays, strings. Understand the difference between scalar and vector
6	Write down a Verilog code for 4-bit ripple carry adder using FA and HA with 3 different modelling techniques.
7	Write down a Verilog code using various operators like Arithmetic, Relational, Equality, Logical, Bitwise, Shift Operators.
8	Other based on syllabus & student

Reference books:

- Millman & Halkias, Electronic Devices and Circuits, McGraw Hill
- David A. Bell, Electronic Devices and Circuits, Oxford University press
- Boylestead & Nashelsky, Electronic Devices and Circuit Theory, PHI
- David Harris and Neil Weste, CMOS VLSI Design: A circuits and systems perspective
- Sung-Mo Kang and Yusuf Leblebici, CMOS Digital Integrated Circuits
- M. Morris Mano, Digital logic and Computer Design, Pearson Education India.
- A. Anand Kumar, Fundamentals of Digital Circuits, Prentice Hall India.
- Samir Palnitkar, Verilog® HDL: A Guide to Digital Design and Synthesis, Prentice Hall PTR.
- Charles H. Roth, Jr. Lizy Kurian and John Byeong Kil Lee, Digital Systems Design using Verilog, Cengage Learning India, 2016.
- Frank Vahid and Roman Lysecky, Digital Design with RTL Design, Verilog and VHDL

Software/ Tool List:

- LTSpice
- Various Digital Electronics Kits
- Xilinx Vivado
- EDA Playground

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