

GUJARAT TECHNOLOGICAL UNIVERSITY

Digital VLSI Design & Verification - I Frontend

Subject Code: 3715204

Semester I

Type of course: Elective

Prerequisite:

1. Digital VLSI system design

Rationale:

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks				Total Marks
L	T	P		Theory Marks		Practical Marks		
			ESE (E)	PA (M)	PA (V) ESE	PA (I)		
4	2	0	5	70	30	30	20	150

L- Lectures; T- Tutorial/Teacher Guided Student Activity; P- Practical; C- Credit; ESE- End Semester Examination; PA- Progressive Assessment;

Content:

Sr. No.	Content	Total Hrs	% Weightage
1	Basics of Semiconductor Devices Recap of semiconductor theory. P-N junction & BJT structure and principle of operation.	6	10
2	Introduction - Levels of IC design – Digital circuit fundamentals Concept of IC. IC structure, components, applications. History and evolution of the IC industry. Moore’s Law. Design economics - Nonrecurring engineering costs (NRE) & Recurring costs, Yield & Technology scaling. System level Design. Top down design. Bottom up design. Back end design. Design abstraction levels. Behavior Level. Register-Transfer Level (RTL). Logic Level. Circuit Level. Component level. Examples of Domains and its Abstraction Levels. CMOS logic gates, Combinational circuits, Sequential circuits, FSM	5	10
3	Design Flow Design flow: Frontend – Marketing Requirements, Design specification, Verification plan, RTL description, Functional verification, Synthesis. Backend – Partitioning, Floor planning, Placement, CTS, Routing & layout generation, layout verification, Tapeout. Foundry – Fabrication Introduction to Testing – Need for testing, manufacturing test, design for test, chip-level test, system-level test, test generation & fault models Introduction to System on chip (Soc) design - Soft cores, Firm cores, Hard cores. Study of AMBA AXI4 protocol.	5	10
4	IC Design techniques & options, CAD design, Design challenges Structured design techniques: hierarchy, regularity, modularity, locality.	7	20

	<p>Design options: Programmable logic Design, sea of gates and gate array design, standard cell design, full-custom design. Study of FPGA tools. Using of CAD tools: Behavioral synthesis tool, RTL synthesis tool , logic optimization tool, structural to layout synthesis tool, layout synthesis tool, design capture tools, design verification tools, circuit extractor, design rule checker (DRC), electrical rule checker (ERC), layout vs. schematic.</p> <p>Design challenges – Microscopic issues: ultra-high speeds, power dissipation, supply rail drop, importance of inter connect, noise, crosstalk, reliability, manufacturability, clock distribution</p> <p>Macroscopic issues: time to market, design complexity, high levels of abstractions, reuse, IP, portability, tool interoperability. Sub-nm technologies: technology scaling, switching power reduction, leakage power control, process variations, die to die frequency variation, temperature variation.</p>		
5	<p>V Digital Design and Synthesis VHDL and Verilog HDL. Concepts of Design planning, partitioning, RTL design for Sequential & combinational circuits, State machines design & encoding. Simulation, waveform generation, coverage reports. Synthesis concepts, Technology libraries, Constraints, Netlist generation & optimization</p>	5	10
6	<p>System Verilog understanding Data types, Attributes, Procedural Statement, Control flow, Operators & Expressions Process, Task & Functions, Classes & OOP concepts, Random generation, constrains, Inter process communication, Clocking blocks, Programming blocks, Assertions, Its use in Design and Verification, Functional Coverage, Direct Programming Interface, Verification Using System Verilog</p>	5	10

Reference Books:

1. Silicon Processing for the VLSI Era, S. Wolf, and R.N. Tauber, ISBN0-9616721-6-1, Lattice Press, Sunset Beach (2000)
2. Synopsys recommended course material and lectures.
3. N. Weste, K. Eshragyan. Principles of CMOS VLSI Design. Adisson Wesley, 1993
4. J.F. Wakerly. Digital Design - Principles & Practices, Prentice Hall, 2001
5. Digital logic design by Morris Mano
6. Verilog HDL: A guide to digital design & synthesis, 2e by Samir Palnitkar
7. IEEE Std 1364™-2005 - Verilog LRM
8. System Verilog LRM.

Course Outcome:

1. After learning the course the students should be able to:
2. Understand about the basics of semiconductor devices.
3. Analyse the levels of IC Design with its history and evolution.
4. Evaluate the design flow in physical design along with partitioning, floor planning, placement and routing.
5. Analyse the IC design techniques, CAD Designs and study about FPGA.
6. Evaluate about the Verilog and VHDL synthesis and simulation.

List of Experiments: (with Open Ended Problems)

1. Write a VHDL code to describe behaviourally the operation of a 2 input mux.
2. Write a VHDL code to describe structurally the operation of a 4 input mux.
3. Write a VHDL code to describe behaviourally the operation of a two output demux.
4. Write a VHDL code to describe behaviourally the operation of an eight input priority encoder.
5. Write a VHDL code to describe behaviourally the operation of a three input Decoder.
6. Write a VHDL code to describe behaviourally the operation of a Full adder.
7. Write a VHDL code to describe structurally the operation of a Full adder
8. Write a VHDL code for the 8-3 binary encoder using case statement
9. Write a VHDL code for the 8-3 binary encoder using if-else statement.
10. Write the VHDL code for the 3-8 Decoder using a) case statement b) if-else
11. Write a VHDL code to describe behaviourally the operation of an eight input priority encoder.
12. Write a verilog code for the 1 bit full adder using DATA FLOW MODELING.
13. Write a verilog code for the 4 bit full adder using DATA FLOW MODELING.
14. Write a verilog code for the 1 bit full SUBTRACTOR using DATA FLOW MODELING
15. Write a verilog code for the 4 bit full adder WITH CARRY LOOKAHEAD using DATA FLOW modeling.
16. Write the Verilog code for the 8 bit magnitude comparator using dataflow style of modeling.
17. Write the verilog code the 4-bit parity Generator using dataflow style of modeling.
18. Write the verilog code for the 2:1 mux using Data flow modeling.
19. Write the verilog code for the 2:1 mux using Conditional Operator.
20. Write the verilog code for the 4:1 mux using Data Flow Modeling.
21. Write the verilog code for the 4:1 mux using Conditional Operator

Major Equipments:

1. FPGA boards.

List of Open Source Software/learning website:

1. Mentor graphics
2. Xilinx ISE
3. Synopsys VCS