VIVEKANAND EDUCATION SOCIETY'S
Institute of Technology
(An Autonomous Institute Affiliated to University of Mumbai, Approved by A.I.C.T.E & Recognized by Govt. of Maharashtra)
Department of Electronics & Telecommunication Engineering

Department of Electronics and Telecommunication Engineering

Syllabus (NEP Scheme)

Dual Multidisciplinary Minor

Register Transfer Level VLSI Design

W.e.f A.Y. 2025-26

Dual Multidisciplinary Minor

Register Transfer Level VLSI Design

Teaching Scheme

Course Type	Semester	Course Name	Teaching scheme (Contact Hours)			Credits Assigned			
			Th	Pr	Tut	Th	Pr	Tut	Total
MDM- I	V	Advance Digital System Design	3	2	-	3	1	-	4
MDM- II	VI	Project Based Learning : Design with VERILOG	3	2	-	3	2	-	5
MDM- III	VII	Project based learning : Verification with System Verilog	3	2+2*	-	3	2	-	5
MDM- IV	VIII	NPTEL COURSE	4	-	-	4	-	-	4
Total Credits						13	5		18
* Self study : Mini Project slot									

Examination Scheme

Course Type	Semester	Course Name	Marks Scheme					
			Th	MT	CA	TW	PrOR	Total
MDM- I	V	Advance Digital System Design	60	20	20	25	-	125
MDM- II	VI	Project Based Learning : Design with VERILOG	60	20	20	25	25	150
MDM- III	VII	Project based learning : Verification with System Verilog	60	20	20	50	-	150
MDM- IV	VIII	NPTEL COURSE	60	20	20	-	-	100
Total Marks								425

COURSE NAME: Advance Digital System Design

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM- I	Advance Digital system Design	03	---	---	03	---	---	03

Advance Digital System Design (Theory)

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM- I	Advance Digital system Design (Theory)	03	---	---	03	--	---	03

Course Code	Course Name	Examination Scheme					
		Theory			Term Work	Practical & Oral	Total
		Internal Assessment		End Sem Exam			
		Mid-Term Test	CA				
MDM- I	Advance Digital system Design (Theory)	20	20	60	--	--	100

Course Prerequisite: Digital System Design, Computer Architecture.	
Course Objectives:	
1	Design and optimize sequential machines (Mealy vs. Moore FSMs).
2	Analyse and synthesize synchronous and asynchronous sequential circuits.
3	Understand the differences between hardwired and micro programmed control units.
4	Design and optimize finite state machines (FSMs) for control logic.
5	Implement micro programmed control units using microcode.
Course Outcomes: After successful completion of the course students will be able :	
1	To analyse, design and implement sequential logic circuits.
2	To develop a digital logic and apply it to solve real life problems.
3	To analyse and design clocked synchronous State Machines.
4	To develop a hardwired programmed processor.
5	To Design a Micro programmed controlled processor.
6	To address real world challenges through digital design.

Advance digital System design (Theory)

Module		Content	Hrs
1		**Fundamentals of Sequential Machines**	**8**
	1.1	Design of 4-bit adder, CLA Adder, ones complement adder, BCD adder , Latches, FF, Shift Register and counters	
	1.2	Finite State Machines (FSMs): Mealy vs. Moore models, State transition tables & diagrams, Synchronous vs. asynchronous sequential circuits, Timing considerations (setup/hold time, clock skew), Metastability and synchronization techniques.	
2		**Clocked Synchronous State Machine Analysis**	**8**
	2.1	**Clocked Synchronous State Machine Analysis:** State Machine Structure, Output logic, Characteristics equation, State Minimization techniques, state diagram, state diagram design and examples. State minimization techniques (Partitioning, Implication Tables). State encoding strategies (Binary, One-Hot, Gray Code), Flip-flop selection (D, T, JK) and excitation tables.	
	2.2	Analysis State Machine with DFF, Analysis State Machine with JK-FF.	
3		**Clocked Synchronous State Machine Design**	**8**
	3.1	State Table design Example, State assignment.	
	3.2	Synthesis using D-FF and JK-FF Design state machine using state diagrams.	
4		**ASM charts and Hazards**	**4**
	4.1	ASM charts, Hazards in sequential circuits (static/dynamic), Testability and fault detection in sequential logic	
5		**Hardwired Control Unit Design**	**4**
	5.1	Control unit basics and design approaches, Hardwired control: Finite State Machine (FSM) approach, Multi-level control logic implementation, Timing and performance considerations, Case study: Hardwired control in RISC processor	
		Micro programmed Control Unit Design	**7**

6	6.1	Microprogramming concepts and terminology, Horizontal vs. vertical microcode, Microinstruction formats and encoding, Microprogram sequencers and control stores, Advantages and disadvantages of microprogramming, Case study: Microprogrammed control in CISC processor	
		Total	**39**

Textbooks:	
1	John F. Warkerly, "Digital Design Principles and Practices", Pearson Education, Fifth Edition (2018).
2	Morris Mano, Michael D. Ciletti, "Digital Design", Pearson Education, Fifth Edition (2013).
3	Carl Hancher, Zvonko Vranesic, Safawat Zaky, "Computer Organization", McGraw Hill, Fifth Edition-2002
Reference Books:	
1	Donald P. Leach / Albert Paul Malvino/Gautam Saha, "Digital Principles and Applications", The McGraw Hill, Eight Edition (2015).
2	Stephen Brown & Zvonko Vranesic, "Fundamentals of Digital Logic Design with VHDL", Second Edition, TMH (2009).
3	Frank Vahid, "Digital Design with RTL design, VHDL and VERILOG", John Wiley and Sons Publisher 2011.
NPTEL/Swayam Courses:	
1	https://cse15-iiith.vlabs.ac.in/List%20of%20experiments.html
2	https://da-iitb.vlabs.ac.in/List%20of%20experiments.html

Internal Assessment:

1) Assessment consists of one Mid Term Test of 20 marks and Continuous Assessment of 20 marks.
2) Mid Term test is to be conducted when approx. 50% syllabus is completed.
3) Duration of the midterm test shall be one hour.

Continuous Assessment:

Continuous Assessment is of **20 marks**. The rubrics for assessment will be considered on approval by the subject teachers. The rubrics can be any 2 or max 4 of the following:

Sr. No	Rubrics	Marks
1	*Certificate course for 4 weeks or more: NPTEL/ Coursera/ Udemy/any MOOC	10 marks
2	Wins in the event/competition/hackathon	10 marks
3	Content beyond syllabus presentation	10 marks
4	Creating Proof of concept	10 marks
5	Mini Project / Extra Experiments/ Virtual Lab	10 marks
6	GATE Based Assignment test/Tutorials etc	10 marks
7	Participation in event/workshop/talk / competition followed by small report and certificate of participation relevant to the subject (in other institutes)	05 marks
8.	Multiple Choice Questions (Quiz)	05 marks
*For sr.no.1, the date of the certification exam should be within the term and in case a student is unable to complete the certification, the grading has to be done accordingly.		

End Semester Theory Examination:	
1	Question paper will be of 60 marks
2	Question paper will have a total of five questions
3	All questions have equal weightage and carry 20 marks each
4	Any three questions out of five need to be solved.

COURSE NAME: Advance Digital System Design LAB

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM- I	Advance Digital System Design Lab	---	02	---	--	01	---	1

Advance Digital System Design LAB

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM- I	Advance Digital System Design Lab	---	02	---	---	01	---	01

Course Code	Course Name	Examination Scheme					
		Theory			Term Work	Practical & Oral	Total
		Internal Assessment		End Sem Exam			
		Mid-Term Test	CA				
MDM- I	Advance Digital System Design (LAB)	--	---	--	25		25

Course Prerequisite: Digital Design, Computer Organization	
Course Objectives:	
1	Develop practical skills in designing, simulating, and implementing digital circuits.
2	Understand the complete workflow from logic design to implementation and testing.

3	Apply theoretical concepts (Boolean algebra, FSM design, sequential logic) to real-world problems.
4	Learn debugging techniques for identifying and resolving issues in digital designs.
Course Outcomes: After successful completion of the course students will be able :	
1	Design Digital Circuits & Implement combinational logic.
2	Design Digital Circuits & Implement sequential logic.
3	Understand cascaded logic implementation with ICs
4	Debug and verify circuits with breadboards and GPPs.

Suggested Experiments: Students are required to complete at least 10 experiments.	
Sr. No.	**Name of the Experiment**
1.	Debugging technique with breadboard and multi meter.
2.	Implementation 4-bit adder and cascaded adder using 7483.
3.	Implementation of CLA adder using gates and ICs
4.	Implementation of Counter using 7490.
5.	Implementation of Mod counter using 7492.
6.	Testing of FF and Latches with ICs
7.	Testing of Static Hazards
8.	Testing of Dynamic Hazard
9.	Implementation of FSM circuit with FF, Latches and Gates
10.	4-5 Experiments with Virtual lab

Term Work:	
1	Term work should consist of 8 to 10 experiments.
2	Journal may include assignments.

3	The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.
4	Total 25 Marks (Experiments: 15-marks, Attendance Theory & Practical: 05-marks, Assignments/Quiz/mock viva/activity: 05-marks)

COURSE NAME: Project Based Learning with Verilog

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM- II	Project Based Learning with Verilog	03	---	---	03	---	---	03

Project Based Learning with Verilog (Theory)

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM- II	Project Based Learning with Verilog (Theory)	03	---	---	03	--	---	03

Course Code	Course Name	Examination Scheme					
		Theory			Term Work	Practical & Oral	Total
		Internal Assessment		End Sem Exam			
		Mid-Term Test	CA				
MDM- II	Project Based Learning with Verilog (Theory)	20	20	60	--	--	100

Course Prerequisite: Digital System Design, Advance Digital System design	
Course Objectives:	
1	Write synthesizable Verilog code for combinational and sequential circuits.
2	Simulate and verify designs using testbenches.

3	Implement designs on FPGAs
4	Debug and optimize Verilog-based digital systems.
5	Design and implement VERILOG based project
Course Outcomes: After successful completion of the course students will be able :	
1	Understand Verilog HDL syntax, data types, and modeling styles.
2	Differentiate between simulation and synthesis in digital design workflows.
3	Design combinational and sequential circuits (e.g., ALUs, FSMs, counters) using Verilog.
4	Debug Verilog code using waveform analysis tools
5	Assess timing constraints and critical paths in FPGA-based implementations.
6	Develop a complete FPGA project using veroilog and Demonstrate hardware-software co-verification techniques.

Project Based Learning with Verilog (Theory)

Module		Content	Hrs
1		**Introduction to VERILOG HDL**	**8**
	1.1	Verilog as HDL, Levels of Design Description, Concurrency, Simulation and Synthesis, Functional Verification, System Tasks, Programming Language Interface (PLI), Module, Simulation and Synthesis Tools, Test Benches.	
	1.2	introduction, Keywords, Identifiers, White Space Characters, Comments, Numbers, Strings, Logic Values, Strengths, Data Types, Scalars and Vectors, Parameters, Memory, Operators, System Tasks: $display, $monitor, $time	
2		**Structural and Dataflow Modeling**	**8**
	2.1	**Dataflow**: Introduction, Continuous Assignment Structures, Delays and Continuous Assignments, Assignment to Vectors, Operators. Example: gates, full adder	
	2.2	**Structural:** Instantiating Modules, Port Mapping (Positional and Named), Hierarchical Naming and Scope, Gate-level Modeling (Basic Logic Gates), Parameter, generate block, Examples: Multiplexer, decoder, CLA adder, 4 bit adder	

3		**Behavioral Modeling**	**6**
	3.1	**Behavioral:** initial and always Blocks, Procedural Assignments, Control Flow Statements: if, case, for, while, Blocking vs. Non-blocking Assignments, Delay Modeling. Sequential circuits (Flip-flops, counters, shift registers)	
	3.2	Tasks and Functions	
4		**Testbenches & Verification**	**5**
	4.1	Testbench structure ($display, $monitor), Clock generation & reset strategies, Stimulus generation (random, file-based inputs), Waveform analysis. Self-checking Testbenches	
5		**RTL Modeling**	**5**
	5.1	Finite State Machine (FSM) design (Mealy & Moore machines), Memory modeling (RAM, ROM).	
6		**FPGA Implementation & Advanced Topics**	**7**
	6.1	Synthesis vs. simulation differences, FPGA architecture overview (LUTs, CLBs, IOBs), Timing constraints & critical path analysis, Optimization techniques (pipelining, resource sharing), Mini-project (UART, PWM, or simple CPU design).	
		Total	**39**

Textbooks:	
1	Samir Palnitkar, "Verilog HDL A guide to Digital Design and Synthesis", 2nd Edition, Pearson Education, (2009)
2	Stephen Brown & Zvonko Vranesic, "Fundamentals of Digital Logic with Verilog Design", Third Edition, MGH (2014).
3	Frank Vahid, "Digital Design with RTL design, VHDL and VERILOG", John Wiley and Sons Publisher 2011.
NPTEL/Swayam Courses:	
1	https://onlinecourses.nptel.ac.in/noc24_cs61/preview
2	https://archive.nptel.ac.in/courses/106/105/106105165/

Internal Assessment:

1) Assessment consists of one Mid Term Test of 20 marks and Continuous Assessment of 20 marks.
2) Mid Term test is to be conducted when approx. 50% syllabus is completed.
3) Duration of the midterm test shall be one hour.

Continuous Assessment:

Continuous Assessment is of **20 marks**. The rubrics for assessment will be considered on approval by the subject teachers. The rubrics can be any 2 or max 4 of the following:

Sr. No	Rubrics	Marks
1	*Certificate course for 4 weeks or more: NPTEL/ Coursera/ Udemy/any MOOC	10 marks
2	Wins in the event/competition/hackathon	10 marks
3	Content beyond syllabus presentation	10 marks
4	Creating Proof of concept	10 marks
5	Extra Experiments/ Virtual Lab	10 marks
6	GATE Based Assignment test/Tutorials etc	10 marks
7	Participation in event/workshop/talk / competition followed by small report and certificate of participation relevant to the subject (in other institutes)	05 marks
8.	Multiple Choice Questions (Quiz)	05 marks
*For sr.no.1, the date of the certification exam should be within the term and in case a student is unable to complete the certification, the grading has to be done accordingly.		

End Semester Theory Examination:	
1	Question paper will be of 60 marks
2	Question paper will have a total of five questions
3	All questions have equal weightage and carry 20 marks each
4	Any three questions out of five need to be solved.

COURSE NAME: Project Based Learning with Verilog (LAB)

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM-II	Project Based Learning with Verilog (LAB)	---	02	---	--	02	---	02

Project Based Learning with Verilog (LAB)

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM-II	Project Based Learning with Verilog (LAB)	---	02	---	---	02	---	02

Course Code	Course Name	Examination Scheme					
		Theory			Term Work	Practical & Oral	Total
		Internal Assessment		End Sem Exam			
		Mid-Term Test	CA				
MDM- II	Project Based Learning with Verilog(LAB)	--	---	--	25	25	50

Course Prerequisite: Digital Design, Computer Organization.	
Course Objectives:	
1	Develop Proficiency in Verilog Coding
2	Master Simulation and Verification
3	Optimize Digital Circuits

4	Debug and Troubleshoot Effectively
5	Adopt Industry Best Practices
Course Outcomes: After successful completion of the course students will be able :	
1	Understand Verilog HDL Fundamentals
2	Design and Simulate Digital Circuits
3	Synthesize and Optimize RTL Designs
4	Debug and Verify Hardware Functionality
5	Implement FPGA-Based Projects
6	Work with Industry-Standard Tools

Suggested Experiments: Students are required to complete at least 10 experiments.
Suggested Tool : AMD Xilinx Vivado, Intel Quartus, EDA Playground
Suggested FPGA Boards : Boolean FPGA Board, Zynq Boards

Sr. No.	Name of the Experiment
1.	Implement and verify using test bench Data Flow code for different logic gates and Full adders using VERILOG
2.	Implement and verify using test bench Behavioural code for 4-bit adder
3.	Implement and verify Generic adder using VERILOG
4.	Implement and verify using test bench Behavioural code for mux and encoder using VERILOG
5.	Implement and verify using test bench Behavioural code for demux and decoder using VERILOG
6.	Implement FF's, Counter using VERILOG
7.	Implement traffic signal FSM and simulate using VERILOG
8.	State machine for one's counter using VERILOG
9.	Implement Multiplier using VERILOG
10.	Implement RAM using VERILOG

	MINI Project: Suggested List
1.	PWM Generator ● Concepts: Duty cycle control, counters. ● Application: LED dimming, motor speed control.
2.	ALU (4 Operations: Add, Sub, AND, OR) ● Concepts: Multiplexers, RTL design. ● Extension: Add shift operations.
3.	FIFO Buffer ● Concepts: Memory modeling, read/write pointers. ● Challenge: Add overflow/underflow flags.
4.	UART (Serial Communication) ● Concepts: Baud rate generation, start/stop bits. ● Test: Send/receive data between FPGA and PC.
5.	VGA Signal Generator (Display Patterns) ● Concepts: Clock domain crossing, metastability. ● Application: Reliable input for FSMs.
6.	Debounce Circuit for Pushbuttons ● Concepts: Clock domain crossing, metastability. ● Application: Reliable input for FSMs.
7.	SPI Interface (Master/Slave) ● Concepts: Serial communication, clock synchronization. ● Extension: Connect to an ADC (e.g., MCP3008).
8.	**RISC-V Single-Cycle CPU Core** ● Concepts: ISA implementation, control unit design. ● Minimal: Support 5-10 instructions (ADD, LW, SW, BEQ).
9.	**CNN Accelerator (Fixed-Point Multiplier)** ● Concepts: Pipelining, parallel processing. ● Simplified: 3x3 convolution for image edge detection.
10.	**Cache Memory Simulator** ● Concepts: Direct-mapped/set-associative caching, LRU policy.

	• Input: Trace files of memory accesses.

Term Work:	
1	Term work should consist of 8 to 10 experiments.
2	Compulsory Mini project [10 Marks]: 1. RTL project must be design and implemented using VERILOG. 2. Simulated with testbench and verified on tool. 3. Synthesized with EDA tool and implemented on FPGA. 4. Small 5-10 pages report to be produced.
3	The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.
4	Suggested TW Mark scheme: Total 25 Marks (Experiments: 10-marks, Attendance Theory & Practical: 05-marks, Mini Project: 10-marks)

COURSE NAME: System Verilog with UVM

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM-III	System Verilog with UVM	03	---	---	03	---	---	03

System Verilog with UVM (Theory)

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM-III	System Verilog with UVM (Theory)	03	---	---	03	--	---	03

Course Code	Course Name	Examination Scheme					
		Theory			Term Work	Practical & Oral	Total
		Internal Assessment		End Sem Exam			
		Mid-Term Test	CA				
MDM- III	System Verilog with UVM (Theory)	20	20	60	--	--	100

Course Prerequisite: Advance Digital System design, Design with Verilog	
Course Objectives:	
1	Understand the need for SystemVerilog and its enhancements over Verilog
2	Apply object-oriented programming (OOP) concepts for verification.
3	Design reusable testbenches with constrained-random stimulus.

4	Implement FPGA-based projects using SystemVerilog.
5	Understand Universal Verification Methodology (UVM) fundamentals.
Course Outcomes: After successful completion of the course students will be able :	
1	Write synthesizable and simulation-optimized SystemVerilog code.
2	Differentiate between Verilog and SystemVerilog constructs and apply them appropriately.
3	Design testbenches using constrained randomization, assertions, and coverage.
4	Implement classes, inheritance, and interfaces in verification scenarios.
5	Perform functional verification with improved automation and efficiency.
6	Plan verification using Universal Verification method (UVM)

System Verilog with UVM (Theory)

Module No.	Unit No.	Topics	Hrs.
		Prerequisite: Advance Digital system design, Design with Verilog Programming	
1	**Introduction to System Verilog**		4
		VERIFICATION GUIDELINES: Evolution from Verilog to SystemVerilog, The Verification Process, Basic Testbench Functionality, Role of re-use in verification, Directed Testing, Methodology Basics, Constrained-Random Stimulus, What Should You Randomize?, Maximum Code Reuse, Layered Testbench, Building a Layered Testbench, Simulation Environment Phases.	
2	**Datatypes and interface**		8
	2.1	User defined types, Enumeration, Casting, Parameterized types Dynamic Arrays, Associative Arrays, Queues/Linked Lists, Structures System Verilog Scheduler, Program Control- fork/join, structures, Packages, Tasks & Functions, subroutines, Dynamic Processes Control Interposes Sync & Communication, Semaphore, mailbox., Choosing a Storage Type, Creating User-Defined Structures, Creating New Types with typedef.	
	2.2	Interface : interface , ports , interface methods, clocking block, virtual interface, system verilog testbench and verilog DUT	
3	**Classes**		7
		Constructors, encapsulation,Inheritance, Virtual methods, Protection, Parameterized classes, Polymorphism, Virtual Classes Interfaces: Interface, Virtual Interface, object allocation, deallocation, static vs global variables, public vs local, Class Methods, Defining Methods Outside of the Class, Scoping Rules, structures and unions	
4	**Functional Coverage and assertions**		5

	4.1	Cover group, Cover point, Cross Coverage methods. Coverage bins, explicit bin creations, transition bins, coverage options.	
	4.2	SV-Assertions: Introduction, event simulation, Immediate assertions, Concurrent assertions, Boolean Expressions, Sequences, Property Block, Verification Directives, Local Data values.	
5	**Randomization**		8
	5.1	Randomization & Constraints: Stimulus Generation techniques, Constraint blocks, Randomize, Random sequences What to Randomize, Randomization in SystemVerilog, constraint Details, Solution Probabilities, Controlling Multiple Constraint Blocks,Valid Constraints, In-line Constraints, The pre_randomize and post_randomize Functions, Random Number Functions , Constraints Tips and Techniques, Common Randomization Problems, Iterative and Array Constraints, Atomic Stimulus Generation vs. Scenario Generation, Random Control, Random Number Generators, Random Device Configuration.	
6	**UVM**		7
		Specification, Feature extraction, Stimulation Generation Plan, Coverage Plan, Verification Environment, Scoreboard.	
		Total	**39**

Textbooks:	
1.	Srikanth Vijayaraghavan, Meyyappan Ramanathan, "System Verilog Assertions", Publisher: Springer. IEEE 1800-2012 SV LRM
2.	Chris Spear, Greg Tumbush, "System Verilog for Verification: A Guide to Learning the Test Bench Language Features"Springer, Second Edition.
3.	Vanessa R. Cooper," Getting Started with UVM: A Beginner's Guide", Kindle Edition
4.	John Aynsley, David Long, Doug Smit,"Doulos UVM Golden Reference Guide", Kindle Edition.
5.	P. Moorby, Stuart Sutherland, Simon Davidmann, "System Verilog for Design Second Edition: A Guide to Using System Verilog for Hardware Design and Modelling Hardcover"

Internal Assessment:

1) Assessment consists of one Mid Term Test of 20 marks and Continuous Assessment of 20 marks.
2) Mid Term test is to be conducted when approx. 50% syllabus is completed.
3) Duration of the midterm test shall be one hour.

Continuous Assessment:
Continuous Assessment is of **20 marks**. The rubrics for assessment will be considered on approval by the subject teachers. The rubrics can be any 2 or max 4 of the following:

Sr. No	Rubrics	Marks
1	*Certificate course for 4 weeks or more: NPTEL/ Coursera/ Udemy/any MOOC	10 marks
2	Wins in the event/competition/hackathon	10 marks
3	Content beyond syllabus presentation	10 marks
4	Creating Proof of concept	10 marks
5	Extra Experiments/ Virtual Lab	10 marks
6	GATE Based Assignment test/Tutorials etc	10 marks
7	Participation in event/workshop/talk / competition followed by small report and certificate of participation relevant to the subject (in other institutes)	05 marks
8.	Multiple Choice Questions (Quiz)	05 marks
*For sr.no.1, the date of the certification exam should be within the term and in case a student is unable to complete the certification, the grading has to be done accordingly.		

End Semester Theory Examination:	
1	Question paper will be of 60 marks
2	Question paper will have a total of five questions
3	All questions have equal weightage and carry 20 marks each
4	Any three questions out of five need to be solved.

COURSE NAME: System Verilog with UVM (LAB)

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM-III	System Verilog with UVM (LAB)	---	02	---	--	02	---	02

System Verilog with UVM (LAB)

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM-III	System Verilog with UVM (LAB)	---	02+2*	---	---	02	---	02

* Self-study : Mini Project slot

Course Code	Course Name	Examination Scheme					
		Theory			Term Work	Practical & Oral	Total
		Internal Assessment		End Sem Exam			
		Mid-Term Test	CA				
MDM- III	System Verilog with UVM(LAB)	--	---	--	50	---	50

Course Prerequisite: Digital Design, Computer Organization.	
Course Objectives:	
1	Develop Proficiency in System Verilog Coding
2	Master Simulation and Verification Planning
4	Debug and Troubleshoot Effectively using System Verilog
5	Adopt Industry Best Practices like UVM

Course Outcomes: After successful completion of the course students will be able :	
1	Understand System Verilog HDL Fundamentals
2	Design verification plan using assertion-based System Verilog technique and basic constructs
3	Write constraint random verification testbenches
4	Debug and Verify Hardware Functionality using OOP in System Verilog
5	Check functional coverage using System Verilog
6	Write verification plan with UVM

Suggested Experiments: Students are required to complete at least 10 experiments. Suggested Tool: Intel Questa sim, EDA Playground	
Sr. No.	**Name of the Experiment**
1.	Test System Verilog constructs: Array/ queue, linked lists, User defined types
2.	Program Control blocks: Fork and Join
3.	Programming with Subroutines: tasks and functions
4.	Programming with Mailboxes and Semaphores
5.	Creation of interface: system Verilog testbench with interface
6.	Class: object, inheritance, polymorphism
7.	Class: structs and unions
8.	Creation of bins
9.	DPI in C
10.	Concurrent and sequential assertion
	MINI Project:
1.	Verifcation plan using UVM
2.	Verifcation plan using Assertion based technique

3.	Verifcation plan using constraint random verification

Term Work:	
1	Term work should consist of 8 to 10 experiments.[25 Marks]
2	Compulsory Mini project [25 Marks]: 1. Project must be design and implemented using VERILOG DUT and system Verilog verification technique. 2. Use any technique Randomization, UVM or assertion based 3. Small 5-10 pages report to be produced.
3	The final certification and acceptance of term work ensures satisfactory performance of laboratory work and minimum passing marks in term work.
4	Suggested TW Mark scheme: Labs (25M) + Mini Project (25 M)

COURSE NAME: NPTEL Course(MOOC)

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM-IV	NPTEL COURSE	04(12 week)	---	---	04	---	---	04

NPTEL Course

Course Code	Course Name	Teaching Scheme (Teaching Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	TW/ PR	Tut	Total
MDM-IV	NPTEL COURSE	04(12 week)	---	---	04	--	---	04

Course Code	Course Name	Examination Scheme					
		Theory			Term Work	Practical & Oral	Total
		Internal Assessment		End Sem Exam			
		Mid-Term Test	CA				
MDM-IV	NPTEL COURSE						

Instructions:

1. Student has to complete 1one of the following 2 courses as per their choice or availability on portal at that time
2. Certification exam can be given through NPTEL/SWAYAM portal

VLSI Design Flow: RTL to GDS (Theory)

ABOUT THE COURSE:
This course covers the entire RTL to GDS VLSI design flow, going through various stages of logic synthesis, verification, physical design, and testing. Besides covering the fundamentals of various design tasks, this course will develop skills in modern chip design with the help of activities and demonstrations on freely available CAD tools. This course will enhance the employability of the students and will make them ready to undertake careers in the semiconductor industry.

INDUSTRY SUPPORT: The course develops skills to use design automation tools for chip designing. The course will be valued by companies working on semiconductors, such as Qualcomm, Intel, Texas Instruments, NXP, ST Microelectronics, Micron, IBM, Cadence, Synopsys, Siemens, ARM, AMD, NVidia, Apple, and Google.

Course Link : https://onlinecourses.nptel.ac.in/noc23_ee137/preview

VLSI Physical Design with Timing Analysis

ABOUT THE COURSE:
The course covers all the steps of VLSI Physical design flow needed for VLSI chip design. It includes all the steps of VLSI Physical design such as partitioning, chip planning, placement, Routing, and finally Clock routing. As the timing of digital circuits is important, three weeks will be completely dedicated to Static Timing Analysis (STA). A demo of several Open-source tools such as Qflow, Yosys, OpenSTA, and OpenROAD is also included in the course.

INDUSTRY SUPPORT: All VLSI industries, For example: Intel, AMD, TI, Qualcomm, Analog Devices, ST-micro-electronics and many more.

Course Link : https://onlinecourses.nptel.ac.in/noc25_ee83/preview