

Dr. Babasaheb Ambedkar Technological University, Lonere

Dr. Babasaheb Ambedkar Technological University
(Established as a University of Technology in the State of Maharashtra)
(Under Maharashtra Act No. XXIX of 2014)
P.O. Lonere, Dist. Raigad, Pin 402 103, Maharashtra
Telephone and Fax. 02140 - 275142
www.dbatu.ac.in



Undergraduate Programme

Bucket of Multi – Disciplinary Minor (MDM) Courses

In

Electronics Engineering (VLSI Design and Technology)

In line with National Education Policy 2020
(Effective from Academic year 2025-26 for University Affiliated
colleges only)

Department of Electronics Engineering

(VLSI Design and Technology)

Course Structure for Multi-Disciplinary Minor (MDM) Bucket in Department of Electronics Engineering (VLSI Design and Technology) Branches (UG) w.e.f. 2025-26 as per NEP-2020
(For Affiliated Institutes Only)

Semester	Course Code	Course Title	Teaching Scheme			Evaluation Scheme				Credit
			L	T	P	CA	MSE	ESE	Total	
III	25AF1378MD306	Digital System Design & Microprocessor	2	0	0	20	20	60	100	2
IV	25AF1378MD406	System Design using Verilog	2	0	0	20	20	60	100	2
V	26AF1378MD506	CMOS VLSI Design	3	0	0	20	20	60	100	3
VI	26AF1378MD606	SystemVerilog for Design and Verification	3	0	0	20	20	60	100	3
VII	27AF1378MD706A 27AF1378MD706B	A. Analog VLSI Design	2	0	0	20	20	60	100	2
		B. FPGA Architecture & VHDL Programming								
VIII	27AF1378MD806	Capstone Project	2	0	0	20	20	60	100	2

Multidisciplinary Minor Domains

Label	Multi-Disciplinary Minor Domains	SY		TY		B-Tech		Offered to students of B Tech Program
		MD1-1	MD2-2	MD3-3	MD4-4	MD5-5	MD5-6	
		Sem-III	Sem-IV	Sem-V	Sem-VI	Sem-VII	Sem-VIII	
MD	VLSI Design & Technology	Digital System Design & Microprocessor	System Design using Verilog	CMOS VLSI Design	SystemVerilog for Design and Verification	A. Analog VLSI Design B. FPGA Architecture & VHDL Programming	Capstone Project	ALL (Except EE (VLSI D & T) Students)

A student wishing to pursue a Minor program in B.Tech. (Electronics Engineering (VLSI Design and Technology)) needs to complete at least 14 credits from the list of courses.

Second Year (Semester III)

Digital System Design & Microprocessor

25AF1378MD306	Digital System Design & Microprocessor	MDM	2L- 0T - 0P	2 Credits
----------------------	---	------------	--------------------	------------------

Teaching Scheme	Examination Scheme
Lecture: 2 hrs./week Tutorial: 0 hr./week	Continuous Assessment: 20Marks Mid Semester Exam: 20 Marks End Semester Exam: 60 Marks

Course Objectives: The aim of this course is to:

1. To introduce fundamental digital concepts including number systems, binary codes, and the principles of Boolean algebra.
2. To teach systematic methods for simplifying Boolean functions using Karnaugh Maps and universal gates.
3. To design and analyze various standard combinational and sequential logic circuits.
4. To understand the architectural details, memory organization, and operating modes of the 8086 microprocessors.
5. To develop proficiency in 8086 assembly language programming, focusing on instruction sets and addressing modes.

Course Outcomes: After completion of the course, students will be able to:

CO1	Convert between different number systems, represent signed and floating-point numbers, and apply Boolean algebra theorems to simplify expressions.
CO2	Minimize Boolean expressions for three, four, and five variables using K-Maps and implement them using NAND/NOR logic.
CO3	Design and explain the operation of common combinational circuits (e.g., adders, decoders) and sequential elements (e.g., flip-flops, counters, shift registers).
CO4	Describe 8086 architecture, memory segmentation, flag register, and differentiate between its minimum and maximum operating modes.
CO5	Write and analyze 8086 assembly language programs, utilizing various addressing modes and control flow instructions for arithmetic and branching operations.

Course Contents:

Unit 1:

[8 hours]

Number systems and codes: Decimal, binary, octal and hexadecimal Number System, Number System conversions, Signed binary numbers: Sign Magnitude Approach, 1's complement Approach, and 2's Complement Approach, Binary codes: Weighted and Non-weighted codes, Alphanumeric codes, Reflective codes, digital logic gates, Boolean algebra, basic theorems & properties, Boolean functions, canonical and standard forms.

UNIT-2:

[8 hours]

Gate level minimizations, K-Map – two variable, three variable, four variable and five variables, SOP, POS simplifications, NAND and NOR implementation and other two-level implementation.

UNIT-3:**[8 hours]**

Combinational circuits for code converters, Binary adders: half adder, full adder, N-bit parallel binary adder, Binary subtractor: half subtractor, full subtractor, N-bit Parallel Binary Subtractor, Binary Multiplier (2x2), comparator (1 and 2 bit only), decoder, encoder, priority encoder, multiplexers, demultiplexers.

UNIT-4:**[8 hours]**

Memory Elements: latches and flipflops, Overview of clock.

Latches: SR latch, Gated SR latch, Gated D latch. Flip-flops: SR flip flop, JK flipflop, D flipflop, and T flipflop. Excitation Table for flip flops, flipflop conversion.

Counters: Asynchronous Counters: ripple, modulo, BCD or decade, and divide by N counter. Ring Counter, Twisted Tail Ring Counter.

Synchronous Counters: n-bit synchronous counter, synchronous counter as sequence generator. Shift Registers.

UNIT-5:**[8 hours]**

Introduction to 8086 Microprocessor: Features, Architecture and Register Organization, Memory Organization & Segmentation, 8086 flag register, Signal description of 8086 common function signals, operating modes, Addressing Modes of 8086, Overview of various types of instruction set.

Textbooks:

1. John Wakerly, "Digital Design: Principles and Practices", Pearson Education
2. D.V. Hall, "Digital Circuits and Systems", Tata McGraw Hill, 1989
3. K M Bhurchandi, A K Ray, Advanced microprocessors and Peripherals, McGraw Hill Education India, 2012, 3rd ed.
4. Ramesh S. Gaonkar, "Microprocessor Architecture, Programming, and Applications with the 8085/8086", Penram International Publishing

Reference Books:

1. R.P. Jain, "Modern digital Electronics", TataMcGrawHill,4thedition,2009.
2. Morris Mano and Michael Ciletti, "Digital Design", Pearson Education
3. Stephen Brown and Zvonko Vranesic, "Fundamentals of Digital Logic with VHDL Design", McGraw-Hill Education

Second Year (Semester IV)

System Design using Verilog

25AF1378MD406	System Design using Verilog	MDM	2L- 0T - 0P	2 Credits
---------------	-----------------------------	-----	-------------	-----------

Teaching Scheme	Examination Scheme
Lecture: 2 hrs./week Tutorial: 0 hr./week	Continuous Assessment: 20Marks Mid Semester Exam: 20 Marks End Semester Exam: 60 Marks

Course Objectives: The aim of this course is to:

1. To know the basic language features of Verilog HDL and the role of HDL in digital logic design.
2. To know the behavioural modelling of combinational and simple sequential circuits.
3. To know the behavioural modelling of algorithmic state machines.
4. To know the synthesis of combinational and sequential descriptions.
5. To know the architectural features of programmable logic devices.

Course Outcomes: After completion of the course, students will be able to:

CO1	Demonstrate knowledge on HDL design flow, digital circuits design, switch debouncing, metastability, memory devices applications
CO2	Design and develop the combinational and sequential circuits using behavioural modelling
CO3	Solving algorithmic state machines using hardware description language
CO4	Analyse the process of synthesizing the combinational and sequential descriptions
CO5	Memorizing the advantages of programmable logic devices and their description in Verilog.

Course Contents:

Unit 1:

[8 hours]

Introduction to Logic Design with Verilog: Structural models of combination logic, logic simulation, design verification, test methodology, propagation delay, truth table models of combinational and sequential logic with Verilog modules, ports, gate types, gate delays, dataflow modelling, continuous assignments delays, expressions, operators, operands, operator.

UNIT-2:

[8 hours]

Logic Design with Behavioral Models of Combinational And Sequential Logic: Behavioral modeling, data types for behavioral modeling, behavioral models of combinational logic, propagation delay and continuous assignments, latches and level sensitive circuits in Verilog, cyclic behavioral models of flip flops and latches, cyclic behavior and edge detection, a comparison of styles for behavioral modeling.

UNIT-3:

[8 hours]

Behavioral models of multiplexers, encoders and decoders data flow model of a LFSR machines with multicycle operations, algorithmic state machine charts for behavioral modeling, asmd charts, behavioral models of counters, shift registers and register files, switch debounce, metastability, synchronizers for asynchronous signals.

UNIT-4:**[8 hour]**

Introduction to synthesis: synthesis of combinational logic, synthesis of sequential logic with latches, synthesis of three state devices and bus interfaces, synthesis of sequential logic with flip flops, synthesis of explicit state machines registered logic. Synchronous Counters: n-bit synchronous counter, synchronous counter as sequence generator.

UNIT-5:**[8 hours]**

Programmable logic devices, storage devices, programmable logic array programmable array logic, programmability of PLDs CPLDs.

TEXTBOOKS:

1. Michael D Ciletti - Advanced Digital Design with the VERILOG HDL, 2ND Edition, PHI, 2009.
2. Samir Palnitkar - Verilog HDL, 2nd edition, Pearson Education, 2003.

REFERENCE:

1. Stephen Brown and Zvonko Vranesic - Fundamentals of Digital Logic with Verilog, 2nd Edition, TMH, 2008.
2. Z Navabi - Verilog Digital System Design, 2nd Edition, McGraw Hill, 2005.

Third Year (Semester V)

CMOS VLSI Design

26AF1378MD506	CMOS VLSI Design	MDM	3L- 0T - 0P	3 Credits
---------------	------------------	-----	-------------	-----------

Teaching Scheme	Examination Scheme
Lecture: 2 hrs./week Tutorial: 0 hr./week	Continuous Assessment: 20Marks Mid Semester Exam: 20 Marks End Semester Exam: 60 Marks

Course Objectives: The aim of this course is to:

1. Explain the operational physics, small-geometry effects, and fabrication layout rules of MOS transistors.
2. Illustrate the static voltage transfer characteristics and transient delay metrics of the complementary CMOS inverter.
3. Apply complementary CMOS, pass-transistor, and transmission gate architectures to realize complex Boolean logic.
4. Calculate the optimized delay paths in standard logic networks utilizing logical effort methodologies.
5. Examine the signal degradation effects caused by resistive, capacitive, and inductive interconnect parasitics.

Course Outcomes: After completion of the course, students will be able to:

CO1	Explain the VLSI Design flow, current-voltage parameters and device physics of MOSFET devices.
CO2	Compare the noise margins, threshold voltages, and power dissipation metrics of CMOS inverters.
CO3	Construct optimized combinational logic circuits using standard Ratioless and transmission gate styles.
CO4	Apply power & delay optimization techniques for CMOS logic
CO5	Design CMOS logic layouts and stick diagrams adhering to standard fabrication technologies.

UNIT-I MOS TRANSISTOR BASICS

[08 Hours]

Overview: VLSI Design Flow: Full custom and Semicustom IC design flow.

Device Physics: The MOS Structure, MOSFET Current-Voltage characteristics, threshold voltage, body effect. Operational regions, MOSFET capacitances. Small-Geometry Effects, Small Signal Model.

UNIT II: MOSFET Inverters

[08 Hours]

Introduction to MOS inverters: Active and passive load nMOS inverters, NMOS inferior pull up device, PMOS inferior pull up device.

Static Characteristics of CMOS Inverter: CMOS Inverter, Voltage Transfer Characteristics (VTC), Inverter threshold voltage, Calculation of Critical Voltages, noise margins.

Transient & Power Analysis of CMOS Inverter: Rise time, fall time, delay estimations. Power Dissipation (dynamic, static, short-circuit), latch up.

UNIT III: BASIC COMBINATIONAL LOGIC DESIGN

[08 hours]

Combinational CMOS Design: Primitive logic gates (NOR & NAND) & realizing complex Boolean expressions using CMOS logic, AOI and OAI gates, CMOS Full Adder Circuit.

Pass-Transistor & Transmission Gate Logic: Pass Transistor logic, CMOS Transmission Gate logic, Basic gates and MUX realization using pass transistor and transmission gate logic.

UNIT IV: Power & Delay Economy & Logical Efforts in CMOS logic [08 Hours]

Power & Delay Economy: Ways to reduce power in CMOS logic, Estimation of Interconnect resistance, Estimation of interconnect capacitance, RC delay model: lumped model, Ways for Reduction of RC delay, Introduction to BiCMOS logic.

Logical Efforts: Basics of logical efforts, calculation of logical efforts for logic gates, electrical effort, and optimizing delay in multistage networks.

UNIT V: CMOS Fabrication & Layout Design Rules [08 Hours]

CMOS Fabrication Flow: CMOS fabrication process flow & technologies.

CMOS Design Layout: Layout Design Rules: Lambda and Micron rules, Stick Diagram, Optimization of Stick Diagram using Euler Graph, Stick Diagram for simple and complex CMOS logic circuits.

Technology Related CAD Issues: Design Rule Checking, Layout Versus Schematic Check.

TEXT BOOKS:

1. Sung-Mo Kang, Yusuf Leblebici “CMOS Digital Integrated Circuits Analysis and Design”, TMH, 3rd Ed., 2011.
2. John P. Uyemura, “CMOS LOGIC CIRCUIT DESIGN”, KLUWER ACADEMIC PUBLISHERS, 2002
3. Ken Martin, “Digital Integrated Circuit Design”, Oxford Press, 2000.
4. Neil H.E. Weste and Harris, “CMOS VLSI Design: A Circuits and Systems Perspective”.

REFERENCES:

1. Jan M. Rabaey, Anantha Chandrakasan, Borivoje Nikolic, “Digital Integrated Circuits – A Design Perspective”, 2nd Ed., PHI.
2. Neil H.E. weste and Kamran Eshraghian, “Principles of CMOS vlsi design: A system Perspective”.
3. Ming-BO Lin, “Introduction to VLSI Systems: A Logic, Circuit and System Perspective”, CRC Press, 2011
4. John Wakerly, “Digital Design – Principles and Practices”, Pearson.

Third Year (Semester VI)

SystemVerilog for Design and Verification

26AF1378MD606	SystemVerilog for Design and Verification	MDM	3L- 0T - 0P	3 Credits
---------------	---	-----	-------------	-----------

Teaching Scheme	Examination Scheme
Lecture: 3 hrs./week Tutorial: 0 hr./week	Continuous Assessment: 20Marks Mid Semester Exam: 20 Marks End Semester Exam: 60 Marks

Course Objectives:

The aim of this course is to:

1. Introduce the foundational guidelines of modern hardware verification methodologies, layered testbenches, and advanced System Verilog data types.
2. Develop the ability to write procedural routines, implement System Verilog assertions, and use interface constructs to effectively connect the testbench to the design under test.
3. Impart the core principles of Object-Oriented Programming (OOP) to help students build reusable, modular, and scalable verification environments.
4. Equip students with the concepts and techniques of constrained-random stimulus generation to efficiently explore and test the state space of complex digital designs.
5. Demonstrate the use of interprocess communication mechanisms (events, semaphores, mailboxes) and functional coverage strategies to measure and analyze verification completeness.

Course Outcomes:

On completion of the course, students will be able to:

CO1	Understand the concepts of verification methodologies and data types.
CO2	Summarize the concepts of procedural statements, routines and assertions.
CO3	Illustrate the concepts of OOP terminology.
CO4	Demonstrate the randomization in System Verilog.
CO5	Analyze the concepts of functional coverage.

Course Contents:

UNIT-I: Verification guidelines and Data types

[08 Hours]

Verification guidelines: Verification Process, Basic Test bench functionality, directed testing, Methodology basics, Constrained-Random stimulus, Functional coverage, Test bench components, Layered test bench, Building layered test bench, Simulation environment phases, Maximum code reuse, Test bench performance.

Data types: Built-in data types, Fixed-size arrays, Dynamic arrays, Queues, Associative Arrays, Linked lists, Array methods, choosing a storage type, creating new types with typedef, Creating user-defined structures, Type conversion, Enumerated types, Constants, strings, expression width.

UNIT-II: Routines and Connecting the test bench & design

[08 Hours]

Procedural statements and routines: Procedural statements, tasks, functions and void Functions, Routine arguments, returning from routine, local data storage, Time values.

Connecting the test bench and design: Separating the test bench and design, Interface constructs, Stimulus timing, Interface driving and sampling, connecting it all together, Top-level scope, Program – Module interactions, System Verilog assertions.

UNIT-III: Basic OOP**[08 Hours]**

Introduction, first class, define a class, OOP (Object Oriented Programming) terminology, Creating new objects, Object de-allocation, Using objects, Static variables vs. Global variables, Class methods, Defining methods outside of the class, Scoping rules, Using one class inside another, Understanding dynamic objects, Copying objects, Public vs. private, Straying off course, building a test bench.

UNIT-IV: Randomization**[08 Hours]**

Introduction, randomization, Randomization in System Verilog, Constraint details, solution probabilities, Controlling multiple constraint blocks, Valid constraints, In-line constraints, The pre_randomize and post_randomize functions, Constraints tips and techniques, common randomization problems.

UNIT-V: Interprocess communication and Functional Coverage**[08 Hours]**

Interprocess Communication, Events, Semaphores, Mailboxes, Coverage Types, Functional Coverage Strategies, Simple Functional Coverage Example, Anatomy of a Cover Group, Triggering a Cover Group, Data Sampling, Cross Coverage, Generic Cover Groups, Coverage Options, Analyzing Coverage Data, Measuring Coverage Statistics During Simulation.

TEXT BOOKS:

1. Chris Spears, System Verilog for Verification, 2nd Edition, Springer, 2008.

REFERENCES:

1. Vijayaraghavan, Srikanth, and Meyyappan Ramanathan. A practical guide for SystemVerilog assertions , Springer Science & Business Media, 2006.
2. Bergeron, Janick. Writing testbenches using SystemVerilog , 1st Edition, Springer Science & Business Media, 2007.