



M. Tech
VLSI and Embedded system Design

Curriculum and Syllabus - AR26

GMRIT Deemed to be University
Rajam, Andhra Pradesh



Institute Vision and Mission

Vision

- ❖ To be a preferred multidisciplinary university that advances knowledge, innovation, sustainability, and societal transformation through excellence in education, research, and industry-integrated learning.

Mission

- ❖ To provide flexible and multidisciplinary education that nurtures academic excellence, critical thinking, creativity and holistic development.
- ❖ To foster a vibrant ecosystem of research, innovation, entrepreneurship, intellectual property creation, and emerging technologies that address industry and societal challenges.
- ❖ To strengthen industry partnerships and global collaborations that enhances experiential learning, professional competence, employability, innovation, and international exposure.
- ❖ To develop ethically responsible, socially conscious, and future-ready graduates committed to sustainability, inclusiveness, leadership, and nation-building

School of Applied Engineering

Vision

- ❖ To be a leading school of engineering that advances sustainable technologies, innovation, and multidisciplinary engineering solutions through excellence in education, research, and industry engagement, contributing to societal progress and nation-building.

Mission

- ❖ To deliver high-quality, multidisciplinary core engineering education that builds strong fundamentals, critical thinking, problem-solving skills, creativity, and lifelong learning.
- ❖ To promote research, innovation, entrepreneurship, and sustainable engineering solutions addressing industrial, environmental, and societal needs.
- ❖ To strengthen collaborations with industry, research bodies, and global partners for experiential learning, professional competence, employability, and innovation.
- ❖ To nurture ethical, socially responsible, and future-ready engineers with leadership, inclusiveness, and commitment to sustainability and nation-building

Department Vision and Mission

VISION

To be a preferred department of learning for students and teachers alike, with a commitment towards academics & research, serving the students in an atmosphere of innovation, critical thinking and making them industry ready.

MISSION

- ❖ To provide adaptable education in a collaborative and innovative environment in skilling the graduates to solve real world problems in the field of Electronics and communication Engineering

- ❖ To prepare the students as critical thinking professionals with multidisciplinary research orientation and Innovation
- ❖ To instill ethical values and nurture the graduates who will be able to contribute to the society

Program Educational Objectives

- PEO1:** Employ logical and analytical skills in solving complex real-world engineering problems in the areas of Electronics and communication Engineering and allied fields
- PEO2:** Adaptable to emerging technologies with enhanced professional skills and ability towards continuous learning, facilitating higher studies and research
- PEO3:** Demonstrate professional ethics, leadership qualities and promote inclusive and collaborative growth with human values towards societal interest.

Program Outcomes

Engineering graduate will be able to

P01:	Apply knowledge of mathematics, natural science, computing, engineering fundamentals to develop to the solution of complex engineering problems. <i>(Engineering Knowledge)</i>
P02:	Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. <i>(Problem Analysis)</i>
P03:	Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. <i>(Design/Development of Solutions)</i>
P04:	Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. <i>(Conduct Investigations of Complex Problems)</i>
P05:	Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. <i>(Engineering Tool Usage)</i>
P06:	Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. <i>(The Engineer and the World)</i>
P07:	Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. <i>(Ethics)</i>
P08:	Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams. <i>(Individual and Collaborative Team)</i>
P09:	Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences <i>(Communication)</i>
P010:	Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to

	manage projects and in multidisciplinary environments. <i>(Project Management and Finance)</i>
PO11:	Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. <i>(Life-long learning)</i>
PS01:	Apply the knowledge of technological evolutions, model / characterize devices and design the integrated circuits to build analog and digital systems. <i>(Program Specific)</i>
PS02:	Understand and apply the fundamentals of communication and signal processing to develop systems wrapped with industry standard protocols and standards. <i>(Program Specific)</i>

Department of Electronics & Communication Engineering

Minimum Credits to be earned: (For Regular Students: 160, For Lateral entry students: 120)

S.No	Course Code	Course Name	L	T	P	C
Semester 1						
1	26MTVL101T	Analog and Digital IC Design	4	0	0	4
2	26MTVL102T	Advanced Microcontrollers and Processors	4	0	0	4
3		Major Core elective I (VLSI)	4	0	0	4
4		Major Core elective II (ES)	3	0	0	3
5	26MTUC101T	Research Methodology and IPR	2	0	0	2
6	26MTVL101L	Embedded Systems Lab	0	0	3	1.5
7	26MTVL102L	Analog and Digital IC Design Lab	0	0	3	1.5
		Total	15	0	6	20
Semester 2						
1	26MTVL109T	Physical Design Automation	4	0	0	4
2	26MTVL110T	IoT System Design	3	0	0	3
3		Major Core elective III (ES)	4	0	0	4
4		Major Core elective IV (VLSI)	3	0	0	3
5	26MTSC101T	Advanced Optimization Techniques ** Minor (open elective)	3	0	0	3
6		Audit Course	0	0	0	0
7	26MTVL103L	Physical Design Automation Lab	0	0	3	1.5
8	26MTVL104L	IoT System Design Lab	0	0	3	1.5
		Total	17	0	6	20
Semester 3						
1	26MTSC201T	Industrial Training (8 weeks)	0	0	0	10
2	26MTSC201P	Project 1 (8 weeks)	0	0	0	10
		Total	0	0	0	20
Semester 4						
1	26MTSC202P	Project 2 (Dissertation)	0	0	0	20
		Total	0	0	0	20

List of Major Core Elective

Major Core Elective I (VLSI)						
No	Course Code	Course	Periods			
			L	T	P	C
1	26MTVL103T	AIML Hardware accelerators	4	-	-	4
2	26MTVL104T	Digital Design through VERILOG	4	-	-	4
3	26MTVL105T	VLSI Signal Processing	4	-	-	4
Major Core Elective II (ES)						
1	26MTVL106T	Embedded Software Design	3	-	-	3
2	26MTVL107T	DSP Processors and Architectures	3	-	-	3
3	26MTVL108T	CPLD and FPGA Architectures & Applications	3	-	-	3
Major Core Elective III (ES)						
1	26MTVL111T	Embedded and Real Time Systems	4	-	-	4
2	26MTVL112T	Hardware Software Co-design	4	-	-	4
3	26MTVL113T	Embedded Networking	4	-	-	4
Major Core Elective IV (VLSI)						
1	26MTVL114T	Low Power VLSI Design	3	-	-	3
2	26MTVL115T	System Verilog Programming for Verification	3	-	-	3
3	26MTVL116T	Memory Architectures	3	-	-	3

26MTVL101T - Analog and Digital IC design

4 - 0 - 0 - 4

Course Outcomes:

At the end of the course, students will be able to

1. Understand MOSFET operation and characteristics in CMOS circuits.
2. Explain single-stage MOS amplifiers and their applications.
3. Analyze current mirrors and biasing circuits.
4. Describe CMOS combinational logic circuit design.
5. Explain latches, flip-flops, registers, and pipelined circuits.
6. Understand arithmetic circuits such as adders, multipliers, and shifters.

Unit 1: Basic MOS device

9 Hours

General consideration: MOSFET as a switch, MOSFET structure, MOS symbols, MOS I/V characteristic: threshold voltage, derivation of I/V characteristics, MOS Trans conductance, second order effects, MOS Device models: MOS Layout, MOS small signal model

Unit 2: Single Stage Amplifiers

12 Hours

Common Source stage: Common Source stage with resistor as a load, CS Stage with Diode-Connected Load, CS Stage with Current-Source Load, CS Stage with Active Load, CS Stage with Triode Load, CS Stage with Source Degeneration, Source Follower, Common Gate amplifier.

Unit 3: Current Mirrors and Biasing Techniques

11 Hours

Basic Current Mirrors, Cascode Current Mirrors, Active Current Mirrors, Biasing Techniques: CS Biasing, CG Biasing, Source Follower Biasing, Differential Pair Biasing.

Unit 4: Combinational Circuits

8 Hours

Introduction, Static CMOS Design: Complementary CMOS, Ratioed Logic, Pass-Transistor Logic, Dynamic CMOS Design: Dynamic Logic- Basic Principles, Speed and Power Dissipation of Dynamic Logic, Issues in Dynamic Design, Cascading Dynamic Gates

Unit 5: Sequential circuits

13 Hours

Introduction, Timing Metrics for Sequential Circuits, Classification of Memory Elements, Static Latches and Registers: The Bistability Principle, Multiplexer-Based Latches Master-Slave Edge-Triggered Register, Low-Voltage Static Latches, Static SR Flip-Flops—Writing Data by Pure Force, Dynamic Latches and Registers: Dynamic Transmission-Gate Edge-triggered Registers, Non-Bistable Sequential Circuits: The Schmitt Trigger, Monostable Sequential Circuits, Astable Circuits.

Unit 6: Designing arithmetic building Blocks

11 Hours

Introduction, datapaths in digital processor architectures, adder: the binary adder, the full adder, Multiplier, shifters: barrel shifter, logarithmic shifter.

Total 64 Hours

Textbooks:

1. David A Johns, Ken Martin, *Analog Integrated circuit Design*, John Wiley & Sons.1996.
2. Behzad Razavi, *Design of Analog CMOS Integrated Circuits*, TMH, 2003.
3. Ken Martin, *Digital Integrated Circuit Design*, Oxford University, 2000.
4. John F Wakerly, *Digital Design Principles & Practices*, Pearson Education & Xilinx Design Series, 3rd Edition, 2002.

Reference Books:

1. Ken Martin *Digital Integrated Circuit Design*, Oxford University, 2000.

2. John F Wakerly, *Digital Design Principles & Practices*, Pearson Education & Xilinx Design Series, 3rd Edition, 2002.
3. Samir Palnitkar, *Verylog HDL-A Guide to Digital Design and Synthesis*, Prentice Hall India 2002.

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL102T - Advanced Microcontrollers and Processors**4 - 0 - 0 - 4****Course Outcomes:****At the end of the course, students will be able to**

1. Understand the architecture, programming concepts, peripherals, and interfacing techniques of 8051/8052 microcontrollers.
2. Analyze Atmel microcontroller architecture and develop simple embedded applications using Atmel devices.
3. Apply programming and interfacing techniques using PIC and AVR microcontrollers for embedded applications.
4. Demonstrate programming and interfacing of ATMEGA microcontrollers and explain ARM processor fundamentals.
5. Evaluate ARM processor core architectures and implement embedded system solutions using ARM-based processors.
6. Analyze embedded processor architectures and embedded board concepts for designing embedded computing systems.

Unit 1: General Microcontrollers**11 Hours**

Introduction to 8051 & 8052 microcontrollers, features, architecture, memory organization, addressing modes, instruction set, assembly programming, software development tools, parallel I/O ports, interrupts, timers/counters, serial communication, data and control transfer operations, serial data transmission, programming and interfacing using 8051, Digital clock design

Unit 2: Atmel Microcontrollers and Applications**10 Hours**

Introduction to Atmel microcontrollers (89CXX and 89C20XX), architectural overview of Atmel 89C51 and 89C2051, pin descriptions, flash memory devices (89XX and 89C20XX), applications of MCS-51, Atmel 89C51 and 89C2051 microcontrollers, Digital clock design, Integrated development environment tools

Unit 3: PIC and AVR Microcontrollers**11 Hours**

Introduction to PIC microcontrollers, PIC 8 series and PIC16 series, PIC family (16C6X/7X, 16F84A, 12F50X, 16F8XX), architecture, instruction set, programming using Assembly and C languages, interfacing and applications of PIC microcontrollers. Introduction to AVR microcontrollers, AVR RISC architecture, instruction set, hardware design issues, hardware/software interfacing, communication links, AVR system development tools, Integrated Development Environment (IDE) tools, Introduction to Smart phone

Unit 4: ATMEGA Microcontrollers and ARM**11 Hours****Fundamentals**

Introduction to AVR family, ATMEGA8 and AT90S1200 microcontrollers, architecture, pin diagrams, programming using C and Assembly languages, interfacing ATMEGA8 to external modules. Introduction to ARM processors, ARM architecture, ARM instruction set, Thumb instruction set, design issues, C and Assembly programming, Basic concepts of Embedded boards

Unit 5: ARM Processor Cores and Embedded System Development**11 Hours**

ARM architectural support for system development, optimized primitives, exception and interrupt handling, cache memory, memory protection units, memory management units, embedded operating systems in ARM, introduction to smartphone architecture, ARM7TDMI and ARM9TDMI architectures, instruction sets, Assembly and C programming, interfacing and applications

Unit 6: ARM Processor Cores**10 Hours**

Introduction to embedded processors, ISA architecture models, internal processor design, processor performance, configurability features, processor architecture, instruction set, programming of embedded processors (PowerPC, MicroBlaze, Nios processors), interfacing with modules, basic concepts of embedded boards, Basic concepts of Embedded boards

Total 64 Hours

Textbooks:

1. *Ajay V Deshmukh, Microcontrollers-Theory and Applications, TMH Publication, 2005.*
2. *Dhananjay V Gadre, Programming and customising the AVR microcontroller, TMH Publications, 2000.*
3. *Stephen B Furber, ARM system – on chip Architecture, Pearson Publishers second edition, 2000.*
4. *Tammy Noergaard, Embedded systems architecture, Elsevier Publications, 2006.*

Reference Books:

1. *V. Udayashankara, M. S. Mallikarjuna Swamy 8051 Microcontroller-Hardware, Software and applications, TMH Publications, 2005.*
2. *lucio Bi Jasio ,PIC microcontrollers, Newnes Publishers, 2008.*

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL103T - AIML Hardware Accelerators
Major Core Elective I (VLSI)

4 - 0 - 0 - 4

Course Outcomes:

At the end of the course, students will be able to

1. Explain the basics of hardware accelerators for AI and machine learning.
2. Describe hardware accelerator architectures and optimization techniques for deep learning.
3. Explain the principles and architecture of quantum hardware accelerators.
4. Describe FPGA-based neural network accelerators and their optimization methods.
5. Explain GPU architectures and their role in deep learning acceleration.
6. Describe Neural Processing Unit (NPU) architectures for deep neural networks.

Unit 1: Introduction to hardware accelerator systems

9 Hours

Introduction to artificial intelligence and machine learning in hardware acceleration, Deep learning and neural network acceleration, RENO: A reconfigurable NoC accelerator, CNN accelerator architecture, Comparison of FPGA, CPU and GPU

Unit 2: Hardware accelerator systems for artificial intelligence and machine learning

12 Hours

Overview of convolutional neural networks, Quantization of weights and activations, Computational elements of hardware accelerators in deep neural networks, Hardware inference accelerators for deep neural networks, Architectures of hardware accelerators, hardware accelerator using a spatial architecture, UNPU and BIT FUSION: hardware accelerators using shift-based multiplier, Digital neuron: a multiplier-less massive parallel processor, Power saving strategies for hardware accelerators

Unit 3: Generic quantum hardware accelerators for conventional systems

11 Hours

Principles of computation, Need and foundation for quantum hardware accelerator design, Algorithms, Programming paradigm and languages, Compiler and runtime requirements, Quantum instruction set architecture (Q-ISA), Quantum microarchitecture, A generic quantum hardware accelerator (GQHA), Deciphering HQAP as a GQHA, Deconstructing GQHA.

Unit 4: FPGA based neural network accelerators

11 Hours

Deep neural network models and computations, Field programmable gate array, FPGA based acceleration systems, Challenges of FPGA based neural network acceleration, Algorithmic optimization, Pruning, Data quantization, Data encoding and sharing, Fast convolution algorithms, Accelerator architecture, Processing element, Vector architecture, Array architecture, Multi-FPGA architecture, Narrow bit-precision architecture

Unit 5: Deep learning with GPUs

13 Hours

Deep learning applications using GPU as accelerator, Overview of graphics processing unit, History and overview of GPU architecture, Structure of GPGPU applications, GPU microarchitecture, Evolution of GPUs, Deep learning acceleration in GPU hardware perspective, NVIDIA tensor core: Deep learning application-specific core, High-bandwidth memory, Multi-GPU system, Multiple-instance GPU, GPU software for accelerating deep learning, Deep learning framework for GPU, Software support specialized for deep learning, Software to optimize data communications on multi-node GPU.

Unit 6: Architecture of neural processing unit for deep neural networks

8 Hours

Introduction of Architecture of neural processing unit for deep neural networks, Considerations in hardware design, NPU architectures, NPU architectures for primitive neural networks, NPU architectures for DNN.

Total 64 Hours

Textbooks:

1. *Hardware Accelerator Systems for Artificial Intelligence and Machine Learning (Advances in Computers Book 122)* by Shiho Kim (Editor, Other Contributor), Ganesh Chandra Deka (Editor, Other Contributor), Pages 1-402 (2021)

Reference Books:

1. *Artificial Intelligence and Hardware Accelerators* by Ashutosh Mishra (Editor), Jaekwang Cha (Editor), Hyunbin Park (Editor), Shiho Kim (Editor). 2023

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL104T - Digital design through VERILOG

Major Core Elective I (VLSI)

4 - 0 - 0 - 4

Course Outcomes:

At the end of the course, students will be able to

1. Illustrate the importance of EDA tools and VLSI designs
2. Design and implement the fundamental digital logic circuits using Verilog HDL
3. Explain the system level design and related concepts.
4. Design and Implement the digital designs against timing parameters
5. Apply knowledge of drawing SM charts
6. Create the basic awareness on FPGA and CPLD architectures

Unit 1: Introduction to Verilog, Language Constructs & Conventions

10 Hours

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, Introduction, Keywords, Identifiers, White Space Characters, Comments, Numbers, Strings, Logic Values, Strengths, Data Types, Scalars and Vectors, Parameters, Memory, Operators, System Tasks, Exercises, VLSI Design, ASIC Design Flow

Unit 2: Gate Level Modeling

10 Hours

Introduction, AND Gate Primitive, Module Structure, Other Gate Primitives, Illustrative Examples, Tri-State Gates, Array of Instances of Primitives, Additional Examples, Design of Flip-Flops with Gate Primitives, Delays, Strengths and Contention Resolution, Net Types, Design of Basic Circuits, Exercises.

Unit 3: Behavioural Modeling, Modeling at Data Flow Level and Switch Level Modeling

12 Hours

Introduction, Operations and Assignments, Functional Bifurcation, Initial Construct, Always Construct, Examples, Assignments with Delays, Wait Construct, Multiple Always Blocks, Designs at Behavioral Level, Blocking and Non-Blocking Assignments, The Case Statement, Simulation Flow, If and If-Else Constructs, Assign-Deassign Construct, Repeat Construct, For Loop, Disable Construct, While Loop, Forever Loop, Parallel Blocks, Force-Release Construct, Event, Introduction, Continuous Assignment Structures, Delays and Continuous Assignments, Assignment to Vectors, Operators, Design of Finite State Machines

Unit 4: System Tasks, Functions, Compiler Directives and Timing Parameters

10 Hours

Introduction, Parameters, Path Delays, Module Parameters, System Tasks and Functions, File-Based Tasks and Functions, Compiler Directives, Hierarchical Access, General Observations, Exercises

Unit 5: Functions, Tasks, User Defined Primitives and Digital Design with SM Charts

10 Hours

Introduction, Function, Tasks, User-Defined Primitives (UDP), FSM Design (Moore and Mealy Machines), State Machine Charts, Derivation of SM Charts, Realization of SM Charts, Implementation of the Dice Game, Alternative Realizations for SM Charts using Microprogramming, Linked State Machines, Design of Finite State Machines.

Unit 6: Designing with Programmable Gate Arrays and Complex Programmable Logic Devices, Verilog Models

12 Hours

Xilinx 3000 Series FPGAs, Designing with FPGAs, Using a One-Hot State Assignment, Altera Complex Programmable Logic Devices (CPLDs), Altera FLEX 10K Series CPLDs, Static RAM Memory, A

Simplified 486 Bus Model, Interfacing Memory to a Microprocessor Bus, UART Design, Design of Microcontroller CPU, Xilinx 4000 Series

Total 64 Hours

Textbooks:

1. *T.R. Padmanabhan and B. Bala Tripura Sundari, Design through Verilog HDL, WSE, 2004.*
2. *J. Bhaskar, A Verilog Primer, BSP, 2003.*

Reference Books:

1. *Stephen. Brown and Zvonko Vranesic, Fundamentals of Logic Design with Verilog, TMH, 2005.*
2. *Charles H Roth, Digital Systems Design using VHDL, Thomson Publications, 2004.*
3. *Michael D. Ciletti, Advanced Digital Design with Verilog HDL, PHI, 2005.*

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL105T - VLSI Signal Processing
Major Core Elective I (VLSI)

4 - 0 - 0 - 4

Course Outcomes:

At the end of the course, students will be able to

1. Design parallel processors in VLSI systems
2. Illustrate the register minimization using the retiming, unfolding & folding concepts.
3. Design systolic architecture using canonical mapping and generalized mapping
4. Design parallel bit circuits
5. Apply FFT for signal processing applications
6. Demonstrate the DSP processors

Unit 1: Introduction to VLSI Signal Processing

11 Hours

Typical Signal Processing Algorithms, Overview of VLSI Architectures, Representations of DSP Algorithms. Introduction, Data Flow Graph Representation, Loop Bound and Iteration Bound, Algorithms for Computing Iteration Bound, Parallel Processing, Pipelining and Parallel Processing for Low Power.

Unit 2: Pipelining, Retiming, Unfolding and Folding
Pipelining of FIR Filters

11 Hours

Definitions and Properties, Solving Systems of Inequalities, Retiming Techniques. Unfolding Algorithm, Properties of Unfolding, Critical Path, Unfolding and Retiming, Folding Transformation, Register Minimization Techniques, Register Minimization in Folded Architectures, Folding of Multirate Systems.

Unit 3: Systolic Architecture Design

10 Hours

Matrix Operations and 2D Systolic Array Design, Parallel Algorithm Expressions, Canonical Mapping Methodology, Generalized Mapping, Systolic Design for Space Representations Containing Delays.

Unit 4: Arithmetic Components and Parallel Bit Circuits

10 Hours

Parallel Bit Circuits: Carry-Look Ahead Addition, Prefix Computations, Carry-Save Addition, Multiplication.

Unit 5: Fast Convolution Algorithms

12 Hours

Introduction, Cook-Toom Algorithm, Winograd Algorithm, Iterated Convolution and Cyclic Convolution. Design of Fast Convolution Algorithm by Inspection.

Unit 6: Programmable Digital Signal Processors

10 Hours

Important Features, DSP Processors for Mobile and Wireless Communications, Processors for Multidimensional Signal Processing.

Total 64 Hours

Textbooks:

1. K. K. Parhi, *VLSI Digital Signal Processing Systems, Design and Implementation*, John Wiley, 1999.

Reference Books:

1. S.Y.Kung, *VLSI Array Processors*, Prentice-Hall, 1988

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL106T - Embedded Software Design

Major Core Elective I (ES)

3 - 0 - 0 - 3

Course Outcomes:

At the end of the course, students will be able to

1. Understand the embedded system design life cycle and the stages involved in developing embedded products.
2. Analyze embedded system selection criteria and design considerations for small embedded systems.
3. Apply hardware–software partitioning principles and explain embedded system execution environments and interrupt handling.
4. Demonstrate embedded software techniques and utilize debugging tools for embedded system development.
5. Evaluate advanced debugging, validation, and testing techniques for embedded software and hardware systems.
6. Analyze the complete embedded life cycle in designing of embedded systems

Unit 1: Embedded Design Life Cycle

8 Hours

Introduction to Embedded Design Life Cycle, Product Specification, Hardware/Software Partitioning, Iteration and Implementation, Detailed Hardware and Software Design, Hardware/Software Integration, Product Testing and Release, Maintaining and Upgrading Existing Products, Small Embedded system Design

Unit 2: Embedded System Selection Process and Small System

8 Hours

Selection Process: Packaging the Silicon, Adequate Performance, RTOS Availability, Toolchain Availability, other issues in Selection Process. Small Embedded System Design concepts and implementation approaches, Introduction to Interrupts

Unit 3: Hardware–Software Partitioning and Development Environment

8 Hours

Partitioning Decision, Hardware/Software Duality, Hardware Trends, ASICs and Revision Costs. Development Environment: Execution Environment, Memory Organization, System Startup. Introduction to Interrupts, Introduction to debugging concept

Unit 4: Embedded Software Techniques and Debugging Tools

8 Hours

Special Software Techniques: Manipulating Hardware, Interrupts and Interrupt Service Routines (ISRs), Watchdog Timers, Flash Memory, Design Methodology. Basic Tool Set: Host-Based Debugging, Remote Debuggers and Debug Kernels, ROM Emulator, Logic Analyzer, Advantages of buffers

Unit 5: Advanced Debugging, Testing, and Validation

8 Hours

Background Debug Mode (BDM), Joint Test Action Group (JTAG), Nexus. ICE–Integrated Solution: Bulletproof Run Control, Real-Time Trace, Hardware Breakpoints, Overlay Memory, Timing Constraints, Trigger Setup. Testing: Need for Testing, Test Planning, Test Case Selection, Embedded Software Testing, Performance Testing, Maintenance and Future Trends. Introduction to Debugging Concepts, Introduction to debugging concept

Unit 6: Embedded Software Development and Buffer Management

8 Hours

Writing Software for Embedded Systems: Compilation Process, Native vs Cross Compilers, Runtime Libraries, Writing and Using Libraries, Porting Kernels, C Extensions for Embedded Systems, Downloading, Emulation and Debugging Techniques. Buffering and Data Structures: Linear Buffers, Directional Buffers,

Double Buffering, Buffer Exchange, Linked Lists, FIFOs, Circular Buffers, Buffer Underflow/Overflow, Buffer Memory Allocation, Memory Leakage, Memory and Performance Trade-offs, Advantages of Buffers, Advantages of buffers

Total 48 Hours

Textbooks:

1. Arnold Berger, *Embedded Systems-An introduction to processes, tools and techniques*, Elsevier, New Delhi, 2010.
2. Barry B. Brey, *The Intel Microprocessors*, 4th Edition, PHI, 1997.

Reference Books:

1. Frank Vahid, Tony D. Givargis, *Embedded system Design: A Unified Hardware/Software Introduction*, John Wiley & Sons Inc.2002.
2. Rajkamal, *Embedded Systems: Architecture, Programming and Design*, TMH Publications, 2nd Ed, 2008.

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL107T - DSP Processors and Architectures

Major Core Elective I (ES)

3 - 0 - 0 - 3

Course Outcomes:

At the end of the course, students will be able to

1. Understand the fundamental concepts and mathematical operations used in Digital Signal Processing systems.
2. Analyze numerical representation methods and computational errors affecting DSP system performance.
3. Explain DSP processor architectural features and evaluate performance enhancement techniques used in DSP devices.
4. Apply programming and architectural concepts of TMS320C54XX DSP processors for signal processing applications.
5. Implement FFT algorithms and analyze spectral computation techniques using DSP processors.
6. Demonstrate DSP processor interfacing techniques using memory, I/O, DMA, and CODEC communication interfaces.

Unit 1: Fundamentals of Digital Signal Processing

8 Hours

Introduction to Digital Signal Processing, Sampling Process, Discrete-Time Sequences, Discrete Fourier Transform (DFT), Fast Fourier Transform (FFT), Linear Time Invariant (LTI) Systems, Decimation and Interpolation, Finite word length effects in digital filters

Unit 2: Number Representation and DSP Computational

8 Hours

Issues

Number Formats for Signals and Coefficients in DSP Systems, Dynamic Range and Precision, Sources of Errors in DSP Implementations, DSP Computational Errors, Compensating Filters, Finite Word Length Effects in Digital Filters, Multiple access memory, multiported memory

Unit 3: DSP Processor Architecture and Performance

8 Hours

Features

Basic Architectural Features of DSP Devices, DSP Computational Building Blocks, Bus Architecture and Memory, Data Addressing Capabilities, Address Generation Unit, Programmability and Program Execution, Speed Issues, External Interfacing Features, Hardware Looping, Interrupts, Stacks, Relative Branch Support, Pipelining and Performance, Pipeline Programming Models, Multiple Access Memory, Multiported Memory, TMS320C54X Assembly language instructions

Unit 4: TMS320C54XX DSP Processor Architecture and

8 Hours

Programming

Introduction to TMS320C54XX DSPs, Data Addressing Modes, Memory Space Organization, Program Control, TMS320C54XX Instructions and Programming, On-Chip Peripherals, Types of Interrupts, Pipeline Operation, TMS320C54X Assembly Language Instructions, A CODEC Interface circuit

Unit 5: FFT Algorithm Implementation using

8 Hours

TMS320C54XX

FFT Algorithm for DFT Computation, Butterfly Computation, Overflow and Scaling, Bit-Reversed Index Generation, 8-Point FFT Implementation on TMS320C54XX, Computation of Signal Spectrum, TMS320C54X Assembly language instructions

Unit 6: DSP Interfacing Techniques and Peripheral

8 Hours

Communication

Memory Space Organization, External Bus Interfacing Signals, Memory Interface, Parallel I/O Interface, Programmed I/O, Interrupts and I/O, Direct Memory Access (DMA), CODEC Interface Circuit, A CODEC Interface circuit

Total 48 Hours**Textbooks:**

1. Avtar Singh and S. Srinivasan, Digital Signal Processing, Thomson Publications, 2004.
2. Lapsley, DSP Processor Fundamentals, Architectures & Features, S. Chand & Co, 2000.

Reference Books:

1. B. Venkata Ramani and M. Bhaskar, Digital Signal Processors, Architecture, Programming and Applications, TMH, 2004.
2. Jonatham Stein, Digital Signal Processing, John Wiley, 2005.

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL108T - CPLD and FPGA Architectures & Applications

Major Core Elective I (ES)

3 - 0 - 0 - 3

Course Outcomes:

At the end of the course, students will be able to

1. Understand the architecture, features, and implementation of programmable logic devices for digital system design.
2. Analyze different CPLD technologies, architectures, and programming methodologies used in programmable system design.
3. Apply FPGA design principles and finite state machine methodologies for programmable logic implementations.
4. Design finite state machine architectures using one-hot encoding, shift registers, and microprogramming techniques.
5. Evaluate advanced FSM modeling techniques using Petri nets, synchronization methods, and parallel controller designs.
6. Develop system-level digital designs using CPLDs, FPGAs, and EDA tools for controller and datapath implementations.

Unit 1: Programmable Logic Devices (PLDs) and CPLDs

8 Hours

ROM, PLA, PAL, CPLD, FPGA – Features, Architectures, Programming, Applications, and Implementation of MSI Circuits using Programmable Logic Devices, 370 Device technology, Lattice pLSI's architectures – 3000 series – Speed performance and in system programmability, Programming technologies

Unit 2: CPLD Technologies and Programming Methods

8 Hours

Complex Programmable Logic Devices: Altera MAX 3000/7000 series, Altera FLEX 10000 series, AMD CPLDs (Mach 1–5), Cypress FLASH 370 Device Technology, Lattice pLSI 3000 Series Architectures, Speed Performance, In-System Programmability, Programming Technologies, Linked state machines

Unit 3: FPGA Architectures and Finite State Machine Design

8 Hours

Field Programmable Gate Arrays (FPGAs): Logic Blocks, Routing Architecture, Design Flow, Technology Mapping, Case Studies – Xilinx XC4000, Altera FLEX 8000/10000, AT&T ORCA, ACTEL ACT-1,2,3. Top-Down Design, State Transition Tables, State Assignment for FPGAs, FSM Realization using PALs, Linked State Machines, Alternative Realization for State Machine Chart using Microprogramming.

Unit 4: FSM Architectures and One-Hot Design Methods

8 Hours

Architectures Centered around Non-Registered PLDs, State Machine Design using Shift Registers, One-Hot State Machines, One-Hot Design Method, Use of ASMs in One-Hot Design, Applications of One-Hot Design Method, Alternative Realization of State Machine Charts using Microprogramming, Extended Petri-nets for parallel controllers

Unit 5: Petri Nets, Synchronization and Complex FSM Design

8 Hours

Petri Nets for State Machines – Basic Concepts and Properties, Extended Petri Nets for Parallel Controllers, Metastability, Synchronization, Complex Designs using Shift Registers, Linked State Machines, Alternative Realization for State Machine Chart using Microprogramming.

Unit 6: System Level Design using CPLDs and FPGAs

8 Hours

Controller and Datapath Design, Functional Partitioning, Digital Front-End Design Tools for FPGAs & ASICs, System Level Design using Mentor Graphics EDA Tool, Design Considerations using CPLDs and FPGAs for Parallel Adder Cells, Sequential Circuits, Counters, Multiplexers, Parallel Controllers, Extended Petri-nets for parallel controllers

Total 48 Hours**Textbooks:**

1. S. Trimberger, Edr, Field Programmable Gate Array Technology, Kluwer Academic Publications, 1994.
2. Richard F. Tindler, Engineering Digital Design, 2nd Edition, Academic press., 2011
3. Fundamentals of logic design-Charles H. Roth, 4th Edition Jaico Publishing House, 2004

Reference Books:

1. P. K. Chan & S. Mourad, Digital Design Using Field Programmable Gate Array, Prentice Hall, 1994.
2. S. Brown, R. J. Francis, J. Rose, Z.G.Vranesic, Field programmable gate array, BSP, 2007.

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTUC101T - Research Methodology and IPR (Common to all branches of Engineering)

2 - 0 - 0 - 2

Course Outcomes:

At the end of the course, students will be able to

1. Identify the fundamentals of research problems.
2. Apply appropriate approaches for research investigation and data analysis.
3. Analyze literature, plagiarism, and research ethics.
4. Develop research proposals using effective technical writing.
5. Explain IPR, patents, and the patenting process.
6. Evaluate patent rights and emerging IPR issues.

Unit 1: Meaning of Research Problem

5 Hours

Meaning of research problem, Sources of research problem, Criteria and characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem.

Unit 2: Approaches of Investigation

5 Hours

Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, Necessary instrumentations.

Unit 3: Research Ethics

6 Hours

Effective literature studies approaches, analysis, Plagiarism, and Research ethics.

Unit 4: Research Proposal

5 Hours

Effective technical writing, how to write report, Paper, Developing a Research Proposal, Format of research proposal, A presentation and assessment by a review committee.

Unit 5: Patenting

5 Hours

Nature of Intellectual Property: Patents, Designs, Trade and Copyright, Process of Patenting and Development: technological research, innovation, patenting, development, International Scenario: International cooperation on Intellectual Property.

Unit 6: Patent Rights

6 Hours

Procedure for grants of patents, Patenting under PCT, Patent Rights: Scope of Patent Rights, Licensing and transfer of technology, Patent information and databases, Geographical Indications, New Developments in IPR: Administration of Patent System, New developments in IPR, IPR of Biological Systems, Computer Software etc., Traditional knowledge Case Studies, IPR and IITs.

Total 32 Hours

Textbooks:

1. *Stuart Melville and Wayne Goddard, Research Methodology: An Introduction for Science & Engineering Students, Juta & Co.*
2. *Ranjit Kumar, Research Methodology: A Step-by-Step Guide for Beginners, 2/e, SAGE Publications.*
3. *C. R. Kothari and Gaurav Garg, Research Methodology: Methods and Techniques, 4/e, New Age International Publishers, 2019.*

Reference Books:

1. *R. Panneerselvam, Research Methodology, 2/e, PHI Learning.*
2. *Wayne Goddard and Stuart Melville, Research Methodology: An Introduction, Juta & Co.*

WIPO, *WIPO Intellectual Property Handbook: Policy, Law and Use*, World Intellectual Property Organization. Glyn James, *Advanced Modern Engineering Mathematics*, 5/e, Pearson publishers, 2018.

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL101L – Embedded Systems Lab

0 - 0 - 3 - 1.5

Course Outcomes:

At the end of the course, students will be able to

1. Demonstrate the installation, configuration, and use of embedded development tools and implement basic GPIO-based applications using LEDs and digital I/O interfaces.
2. Design and develop embedded applications involving human-machine interfaces such as Seven Segment Displays, LCDs, and Matrix Keypads.
3. Implement serial communication protocols including UART/USART for reliable data transmission and reception in embedded systems.
4. Develop and verify embedded applications using I2C communication protocol, including master/slave operation, memory interfacing, and bus monitoring.
5. Design embedded systems employing advanced communication interfaces such as SPI and USB, including speed detection and protocol analysis.
6. Analyze, acquire, and synchronize real-time embedded data using digital I/O, event-driven techniques, and data capture mechanisms for embedded applications.

List of Experiments:

1. Installation and configuration of Embedded Development Environment (IDE) and target board.
2. Basic data streaming and LED interfacing using GPIO.
3. Interfacing and control of Seven Segment Display.
4. Interfacing of Matrix Keypad and key detection.
5. Serial communication using USART/UART.
6. Interfacing and operation of LCD display module.
7. Programming GPIO-based Digital Input/Output applications.
8. Simulation and implementation of I2C Master device.
9. Simulation and implementation of I2C Slave device.
10. Programming and verification of I2C-based EEPROM/Memory device.
11. Passive real-time monitoring and analysis of I2C bus transactions.
12. SPI communication between embedded devices.
13. Real-time data capture and delayed-download data acquisition.
14. High-Speed USB Chirp Detection and analysis.
15. Automatic USB Speed Detection and enumeration.
16. Digital I/O synchronization and event-triggered control applications.

Total 32 Hours

Textbooks:

1. *Ajay V Deshmukh, Microcontrollers-Theory and Applications, TMH Publication, 2005.*
2. *Dhananjay V Gadre, Programming and customising the AVR microcontroller, TMH Publications, 2000.*
3. *Stephen B Furber, ARM system – on chip Architecture, Pearson Publishers second edition, 2000.*
4. *Tammy Noergaard, Embedded systems architecture, Elsevier Publications, 2006.*

Reference Books:

1. *V. Udayashankara, M. S. Mallikarjuna Swamy 8051 Microcontroller-Hardware, Software and applications, TMH Publications, 2005.*

2. *lucio Bi Jasio ,PIC microcontrollers, Newnes Publishers, 2008.*

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks
Continuous Assessment <i>(Preparation, Observation & Results – 10, Record – 5, Viva-Voce – 15, Internal Test – 10)</i>	40
End Semester Assessment (ESA) <i>(Procedure & Write-up – 10, Conduct of Experiment/Program – 15, Analysis & Results – 20, Record – 5, Viva-Voce – 10)</i>	60
Total	100

26MTVL102L - Analog and Digital IC Design Lab

0 - 0 - 3 - 1.5

Course Outcomes:

At the end of the course, students will be able to

1. Carry out the simulations of basic analog circuits
2. Carry out the simulations of MOSFET circuits
3. Implement the layout diagrams for CMOS circuits
4. Carry out the DRC and LVS in a given circuit implementation
5. Compute the effects of parasitic present in the various layers of the layouts
6. Demonstrate the ASIC implementation flow

List of Experiments:

Simulate a basic Common Source amplifier

1. Simulate a basic Common Source amplifier
2. Simulate a basic Common Drain amplifier
3. Simulate a basic Common Gate amplifier
4. Simulate a basic Current mirror
5. Verify the functionality of a CMOS inverter
6. Verify the functionality of a CMOS AND gate
7. Perform the Layout for the CMOS Inverter
8. Design the layout of a CMOS OR gate
9. Perform the LVS and DRC for the CMOS Inverter Layout
10. Perform the LVS and DRC for the CMOS OR gate Layout
11. Perform the RC extraction for the CMOS inverter
12. Compute the delay of the CMOS inverter after the RC extraction
13. Design and simulate the 2:1 MUX using transmission gate logic
14. Design and simulate the 1:2 De-MUX using transmission gate logic
15. Perform the design and ASIC Implementation of an 8-bit ripple carry adder
16. List of Mini-Projects2:

(I) Carry out the ASIC implementation of a universal shift register

(II) Carry out the full custom IC implementation of a ALU performing four arithmetic and four logical operations.

Total 32 Hours

Textbooks:

1. Kamran Eshraghian, Douglas A. Pucknell And Sholeh Eshraghian, *Essentials of VLSI Circuits and Systems*, Prentice-Hall of India Private Limited, 2005 Edition.
2. Behzad Razavi, *Design of Analog CMOS Integrated Circuits*, McGraw Hill, 2003
3. Jan M. Rabaey, Anantha Chandrakasan and Borivoje Nikolic, *Digital Integrated Circuits*, Pearson Education, 2nd edition, 2016.
4. Weste and Eshraghian, *Principles of CMOS VLSI Design*, Pearson Education, 3rd Edition, 1999

5. Michael John Sebastian Smith, *Application specification integrated circuits*, Addison Wesley, 1st edition, 1997

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks
Continuous Assessment <i>(Preparation, Observation & Results – 10, Record – 5, Viva-Voce – 15, Internal Test – 10)</i>	40
End Semester Assessment (ESA) <i>(Procedure & Write-up – 10, Conduct of Experiment/Program – 15, Analysis & Results – 20, Record – 5, Viva-Voce – 10)</i>	60
Total	100

26MTVL109T - Physical Design Automation

4 - 0 - 0 - 4

Course Outcomes:

At the end of the course, students will be able to

1. Explain design methodologies used in VLSI design.
2. Apply Backtracking, Branch and Bound Methods for combinational optimization
3. Illustrate layout, placement, routing, and simulation techniques.
4. Explain logic synthesis and high-level synthesis concepts.
5. Apply FPGA physical design automation techniques
6. Analyze the physical design flow of MCM's.

Unit 1: Design Methodologies

10 Hours

Introduction to Design Methodologies, Design Automation tools, Algorithmic Graph Theory, Computational Complexity, Tractable and Intractable Problems, Power Analysis

Unit 2: Combinational Optimization

11 Hours

Backtracking, Branch and Bound, Dynamic Programming, Integer Linear Programming, Local Search, Simulated Annealing, Tabu search, Genetic Algorithms, Graph Algorithms: Depth-First Search, Breadth-First Search

Unit 3: Layout & Simulation

11 Hours

Layout Compaction, Placement, Floorplanning and Routing Problems, Concepts and Algorithms. Gate Level Modelling and Simulation, Switch level modeling and simulation, Partitioning, Kernighan-Lin Partitioning Algorithm

Unit 4: Logic Synthesis and High Level Synthesis

12 Hours

Basic issues and Terminology, Binary, Decision diagram, Two- Level Logic Synthesis. Hardware Models, Internal representation of the input algorithm, Allocation, Assignment and Scheduling, Some Scheduling Algorithms, Some aspects of Assignment problem, High – level Transformations, Heuristic based on ROBDDs

Unit 5: FPGA Physical Design Automation

10 Hours

FPGA technologies, Physical Design cycle for FPGA's partitioning and Routing for segmented and staggered models, Routing in Programmable FPGA Modules

Unit 6: MCM Physical Design Automation

10 Hours

MCM technologies, MCM physical design cycle, Partitioning, Placement – Chip array based and full custom approaches, Routing – Maze routing, Multiple stage routing, Topologic routing, Integrated Pin – Distribution and routing, Routing in Programmable Multichip Modules

Total 64 Hours

Textbooks:

1. S. H. Gerez ,*Algorithms for VLSI Design Automation*, WILEY student Edition, John Wiley & Sons (Asia) Pvt. Ltd. 1999.
2. Naveed Sherwani, *Algorithms for VLSI Physical Design Automation*, 3rd Edition, Springer International Edition, 2005.

Reference Books:

1. Fredrick J. Hill and Gerald R. Peterson, *Computer Aided Logical Design with Emphasis on VLSI*, Wiley Publishers, 4th Edition, 1993.

2. Wayne Wolf, *Modern VLSI Design: Systems on silicon*, Pearson Education Asia, 2nd Edition, 1998.

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL110T - IoT System Design**3 - 0 - 0 - 3****Course Outcomes:****At the end of the course, students will be able to**

1. Explain the fundamental concepts of IoT, networking reference models, M2M communication, and IoT standardization frameworks
2. Analyze and compare various wired and wireless communication technologies and application-layer protocols used in IoT systems
3. Demonstrate the role of advanced IoT networking protocols, addressing mechanisms, and gateway-based data management techniques
4. Select and interface appropriate sensors, actuators, and development platforms for IoT-based applications
5. Evaluate IoT business models and examine the applications of IoT in industrial, automotive, and smart-grid domains
6. Apply cloud computing concepts and IoT platforms to design solutions for smart homes, smart cities, agriculture, and healthcare applications

Unit 1: Fundamentals of IoT and Networking Models**8 Hours**

Introduction to OSI and TCP/IP Reference Models, Overview of Internet of Things (IoT), IoT Conceptual Frameworks: Oracle, IBM, and Cisco, Introduction to Machine-to-Machine (M2M) Communication, IoT Design Standards: IETF, ITU-T, ETSI, and OGC, Modified OSI Models for IoT/M2M Systems proposed by IETF and ITU-T

Unit 2: IoT Communication Technologies and Protocols**8 Hours**

Wireless Communication Technologies: Zigbee, Bluetooth/BLE, Wired Communication Technologies: SPI, I2C, UART, Constrained Application Protocol (CoAP), Message Queue Telemetry Transport (MQTT) Service

Unit 3: Advanced IoT Networking and Device Management**8 Hours**

6LoWPAN Protocol, LoRaWAN Protocol, IP Addressing in IoT (IPv4 and IPv6), Media Access Control (MAC) Addressing, Data Enrichment, Data Consolidation, Device Management at Gateway

Unit 4: Sensors, Actuators and IoT Development Platforms**8 Hours**

Sensors for IoT Applications: Temperature Sensor, Humidity and Temperature Sensor, Light Sensor, Touch Sensor, Smoke Detector, Rain Detector, Ultrasonic Sensor, Soil Moisture Sensor, RFID, Actuators: Solenoid Valve, DC Motor, Stepper Motor, Servo Motor, Relay, Development Boards: Arduino, ESP32, NodeMCU, Raspberry Pi

Unit 5: IoT Business Models and Industrial IoT**8 Hours**

Business Models in IoT, Business Scenarios in IoT, Industrial Internet of Things (IIoT), Automotive IoT, IoT for Smart Grid

Unit 6: Cloud Computing and IoT Applications**8 Hours**

Cloud Platforms for IoT, Virtualization Concepts and Cloud Architecture, Cloud Computing Fundamentals and Benefits, Cloud Service Models: SaaS, PaaS, IaaS, Cloud Providers and Offerings, IoT Applications: Smart Home, Smart City, Precision Agriculture, Healthcare

Total 48 Hours

Textbooks:

1. . Raj Kamal, *"Internet of Things: Architecture and Design Principles"*. TMH Publications, 2017.
2. Yogesh Misra, *Programming and Interfacing with Arduino*, CRC Press, 1st Edition, 2021

Reference Books:

1. Vijay Madiseti and Arshdeep Bahga, *"Internet of Things (A Hands-on-Approach)"*, 1st Edition, VPT, 2014
2. Behrouz A. Forouzan, *Data Communications and Networking*, 5th Edition, Tata McGraw Hill Higher Education, 2013
3. Boyle, *From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence*, 1st Edition, Academic Press, 2014. 2M Communications, ISBN: 978-1-118-47347-

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL111T - Embedded and Real Time Systems
Major Core Elective III (ES)

4 - 0 - 0 - 4

Course Outcomes:

At the end of the course, students will be able to

1. Understand embedded system fundamentals, processor technologies, and design methodologies used in embedded applications.
2. Analyze processor architectures including general-purpose processors, ASIPs, microcontrollers, and DSPs for embedded system development..
3. Apply embedded system modeling techniques and communication interfaces in concurrent and real-time embedded applications..
4. Evaluate real-time system scheduling techniques and programming language features for embedded software development..
5. Demonstrate embedded programming practices and analyze database management concepts in real-time systems.
6. Develop embedded design solutions using synthesis, verification, simulation, and hardware – software co-design methodologies.

Unit 1: Fundamentals of Embedded Systems and Processor Technologies

10 Hours

Embedded Systems Overview, Design Challenges, Processor Technology, Design Technology, Design Trade-offs. Single Purpose Processors, RT-Level Combinational Logic, Sequential Logic (RT-Level), Custom Purpose Processor Design, Optimization of Custom Single Purpose Processors, Application of Embedded systems design

Unit 2: General Purpose Processors and Embedded Processing Platforms

11 Hours

Introduction, models Vs Languages, finite state machines with data path model (FSMD), program state machine model (PSM), concurrent process model, concurrent processes, communication among processes, synchronization among processes, Implementation, data flow model, real-time systems. Communication Processes – Need for communication interfaces, RS232/UART, RS422/RS485, USB, Infrared, IEEE1394 Firewire, Ethernet, IEEE 802.11, Blue tooth, Existed embedded project implementation with the models.

Unit 3: Embedded System Models, Concurrent Processes and Communication Interfaces

11 Hours

Introduction to Models and Languages, Finite State Machine with Datapath (FSMD) Model, Program State Machine (PSM) Model, Concurrent Process Model, Process Communication and Synchronization, Data Flow Model, Real-Time Systems. Communication Processes: Need for Communication Interfaces, RS232/UART, RS422/RS485, USB, Infrared, IEEE1394 FireWire, Ethernet, IEEE 802.11, Bluetooth. Existing Embedded Project Implementations using Models. C and C++ Programming with Visual C basic concepts.

Unit 4: Real-Time Systems, Scheduling and Programming Languages

11 Hours

Introduction to Real-Time Systems, Issues in Real-Time Computing, Structure of Real-Time Systems, Task Classes, Performance Measures, Program Run-Time Estimation. Task Assignment and Scheduling: Classical Uniprocessor Scheduling Algorithms, IRIS Task Scheduling, Mode Changes, Fault-Tolerant Scheduling. Programming Languages and Tools: Language Characteristics, Data Typing, Control Structures, Hierarchical Decomposition, Packages, Exception Handling, Overloading, Generics, Multitasking, Low-Level Programming, Timing Specifications, Programming Environments, Run-Time Support, DBMS concept, Simulation tools

Unit 5: Embedded Programming Tools and Real-Time Databases

13 Hours

C and C++ Programming Concepts using Visual C, Programming Environments and Tool Support. Real-Time Databases: Definitions, Real-Time vs General Purpose Databases, Main Memory Databases, Transaction Priorities, Transaction Aborts, Concurrency Control, Disk Scheduling Algorithms, Predictability Approaches, Serialization Consistency, Databases for Hard Real-Time Systems, C and C++ Programming with Visual C basic concepts

Unit 6: Embedded Design Technology and Hardware-Software Co-Design

8 Hours

Automation, Synthesis, Parallel Evolution of Compilation and Synthesis, Logic Synthesis, RT Synthesis, Behavioral Synthesis, System Synthesis, Hardware/Software Co-Design, Verification, Hardware/Software Co-Simulation, Reuse of Intellectual Property (IP) Cores, DBMS Concepts, Simulation Tools, DBMS concept, Simulation tools

Total 64 Hours

Textbooks:

1. Frank Vahid, Tony D. Givargis, John Wiley & Sons, *Embedded System Design-A Unified Hardware/Software Introduction*, Wiley, Inc. 2002.
2. KVKK prased, *Embedded/Real Time Systems*, Dreamtech press, 2005.
3. Krishna. C. M, Kang. G, Shin, *Real Time Systems*, McGraw Hill, 2003.

Reference Books:

1. Herma. K, *Real Time Systems-Design for distributed Embedded Applications*, Kluwer Academic, 2002.
2. Charles Crowley, *Operating Systems-A Design Oriented approach*, McGraw Hill, 2004.
3. Raymond J.A. Bhur, Donald L. Bailey, *"An Introduction to Real Time Systems"*, PHI 2002.

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL112T - Hardware Software Co-design
Major Core Elective III (ES)

4 - 0 - 0 - 4

Course Outcomes:

At the end of the course, students will be able to

1. Understand the principles, models, and methodologies of hardware–software co-design for embedded systems.
2. Analyze prototyping, emulation techniques, and embedded architectures used in co-design environments.
3. Apply system design methodologies, software development concepts, and concurrent computation techniques in embedded co-design.
4. Evaluate verification methodologies, system-level specification approaches, and design tools for hardware–software co-design.
5. Analyze firmware design and performance modeling techniques for embedded hardware–software systems.
6. Develop embedded co-design solutions using high-level synthesis techniques and cross-compilation tools.

Unit 1: Fundamentals of Hardware–Software Co-Design

11 Hours

Introduction to Co-Design, Co-Design Models, Architectures, Languages, Generic Co-Design Methodology, Hardware–Software Synthesis Algorithms, Hardware–Software Partitioning, Distributed System Co-Synthesis, Waterfall Model of Co-Design, Waterfall model of co Design

Unit 2: Prototyping, Emulation and Embedded Architectures

11 Hours

Prototyping and Emulation Techniques, Prototyping and Emulation Environments, Future Developments in Emulation and Prototyping, Architecture Specialization Techniques, System Communication Infrastructure, Target Architectures, Application System Classes, Architectures for Control-Dominated and Data-Dominated Systems, Integrated Development Environment

Unit 3: System Design and Embedded Software Development

10 Hours

Modern Embedded Architectures, Embedded Software Development Needs, Compilation Technologies, Compiler Development Environment, Design Specification and Verification, Co-Design Computational Model, Concurrency, Coordinating Concurrent Computations, Interfacing Components, Integrated Development Environment (IDE), Programming languages

Unit 4: Design Verification and System-Level Design Tools

11 Hours

Design Verification, Implementation Verification, Verification Tools, Interface Verification, System-Level Specification, Design Representation for System-Level Synthesis, System-Level Specification Languages, Heterogeneous Specifications, Multi-Language Co-Simulation, COSYMA System and LYCOS System, Programming Languages, Cross Compiler

Unit 5: Firmware Design and Performance Modeling

11 Hours

Firmware Design, System-Level Performance Modeling vs Low-Level Performance Modeling, Modeling of Execution Speed (System Latency), Energy Consumption Modeling for Hardware and Software, Estimation of Memory Requirements, Programming languages

Unit 6: High-Level Synthesis and Cross Compilation

10 Hours

High-Level Synthesis, Behavioral Specification of Hardware, Module Set Allocation, Resource Binding, Operation Scheduling, Controller Synthesis, Cross Compiler Concepts and Applications, Cross Compiler

Total 64 Hours

Textbooks:

1. *Kluwer, Hardware /Software Co-Design Principles and Practice, Academic Publishers, 2nd printing 2000.*

Reference Books:

1. *Arnold S. Berger, Embedded System Design, CMP books, USA 2002.*
Wayne Wolf, Computers as Components: Principles of Embedded Computer Systems Design, Morgan Kaufman Publishers, 2005.

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL113T - Embedded Networking
Major Core Elective I (ES)

4 - 0 - 0 - 4

Course Outcomes:

At the end of the course, students will be able to

1. Understand embedded networking concepts and the fundamental architecture of CANopen communication systems
2. Analyze CANopen communication services, messaging protocols, and network management operations.
3. Apply CANopen configuration, deployment, and testing methodologies in embedded network applications.
4. Demonstrate the implementation and simulation of CAN-based embedded communication systems.
5. Evaluate Micro CANopen architecture, source code structure, and conformance testing procedures.
6. Develop embedded networking solutions using Micro CANopen concepts, node management, and advanced communication objects.

Unit 1: Fundamentals of Embedded Networking and CANopen

11 Hours

Embedded Networking: Code Requirements, Communication Requirements, Introduction to CANopen, CANopen Standards, Object Dictionary, Electronic Data Sheets (EDS) and Device Configuration Files, Pseudo code of CAN

Unit 2: CANopen Communication Services and Network Management

11 Hours

Service Data Objects (SDO), Network Management (NMT), CANopen Messages, Device Profile Encoder, Pseudo Code of CAN, Communication Mechanisms in CANopen Networks, Simple application using CAN

Unit 3: CANopen Configuration and System Deployment

10 Hours

Evaluating System Requirements, Choosing Devices and Development Tools, Configuring Single Devices, Overall Network Configuration, Network Simulation, Network Commissioning, Advanced Features and Testing, Simple Applications using CAN, Simulation using CAN

Unit 4: Controller Area Network (CAN) Technology and Implementation

11 Hours

Underlying CAN Technology, CAN Overview, Selection of CAN Controllers, CAN Development Tools, Implementing CANopen Communication Layout and Requirements, Comparison of Implementation Methods, Simulation using CAN, Exploring micro CAN open

Unit 5: Micro CANopen Architecture and Source Code Concepts

11 Hours

Micro CANopen, CANopen Source Code, Conformance Testing, Entire Design Life Cycle, Physical Layer, Data Types, Object Dictionary, Simulation using CAN

Unit 6: Advanced Micro CANopen Concepts and Node Management

10 Hours

Communication Object Identifiers (COB-ID), Emerging Objects, Node States, Advanced CANopen Features, Exploring Micro CANopen Applications and Practical Design Considerations, Exploring micro CAN open

Total 64 Hours

Textbooks:

1. *Glaf P. Feiffer, Andrew Ayre and Christian Keyold, Embedded Networking with CAN and CAN open Embedded System Academy 2005.*

Reference Books:

1. *Gregory J. Pottie, William J. Kaiser, Principles of Embedded Networked Systems Design, Cambridge University Press, 2nd Edition, 2005.*

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL114T - Low Power VLSI Design

Major Core elective IV (VLSI)

3 - 0 - 0 - 3

Course Outcomes:

At the end of the course, students will be able to

1. Design and estimate the power of different circuits
2. Apply Bi-CMOS and advanced Bi-CMOS circuits for different applications
3. Apply different isolation techniques in chip design
4. Design low power memory elements
5. Demonstrate the Latches and Flip-Flops in detail
6. Demonstrate ESD free BiCMOS circuits

Unit 1: Introduction to Low Power VLSI Design

8 Hours

Sources of power dissipation – Physics of power dissipation in MOSFET devices: The MIS structure, long channel MOSFET, Submicron MOSFET, gate induced drain leakage– Power dissipation in CMOS : short circuit dissipation, dynamic dissipation, load capacitance– Low power VLSI design: Limits – principles of low power design.

Unit 2: Design Of Low Power Circuits

8 Hours

Transistor and Gate Sizing : Sizing an Inverter Chain, Transistor and Gate Sizing for Dynamic Power Reduction, Transistor Sizing for Leakage Power Reduction - Network Restructuring and Reorganization : Transistor Network Restructuring, Transistor Network Partitioning and Reorganization - Special Latches and Flip-flops : Self-gating Flip-flop, Varieties of Boolean Functions, Adjustable Device Threshold Voltage.

Unit 3: MOS/BiCMOS Process Technology and ESD-Free Circuits

8 Hours

Conventional CMOS and BiCMOS logic gates, Performance evaluation, Comparison of advanced BiCMOS digital circuits, ESD-free BiCMOS digital circuit. Bootstrapped-Type BiCMOS.

Unit 4: Design and Test of Low Voltage CMOS Circuits

8 Hours

Introduction, Circuit design style, Leakage current in deep submicrometer transistors, Deep submicrometer device design issues, Low voltage circuit design issues. Testing deep submicrometer ICs with elevated intrinsic leakage.

Unit 5: Low Power Static RAM Architectures

8 Hours

Introduction, Organization of a Static RAM, MOS Static RAM Memory Cell, Banked Organization of SRAMs, Reducing Voltage Swings on Bit Lines, Reducing Power in the Write Driver Circuits, Reducing Power in Sense Amplifier Circuits.

Unit 6: Evolution of Latches and Low Energy Computing Using Energy Recovery Techniques

8 Hours

Evolution of Latches and Flip-Flops, Quality Measures for Latches and Flip-Flops, Design Perspective. Energy Dissipation in Transistor Channel Using an RC Model, Energy Recovery Circuit Design, Quality measures of latches and flipflops

Total 48 Hours

Textbooks:

1. Yeo Rofail/ Gohl (3 Authors), *CMOS/BiCMOS ULSI low voltage, low power*, Pearson Education Asia 1st Indian reprint, 2002
2. Kaushik Roy, Sharat Prasad, *Low Power CMOS VLSI Circuit Design*, Wiley publishers 2000.

Reference Books:

1. Gary Yeap, Practical VLSI Design, Springer publishers, 1997.

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL115T - System Verilog Programming for Verification

Major Core Elective IV (VLSI)

3 - 0 - 0 - 3

Course Outcomes:**At the end of the course, students will be able to**

1. Interpret the verification guidelines and data types
2. Execute the programs using Procedural Statements and Routines.
3. Demonstrate the System Verilog constructs through simulations
4. Explain the basic OOPs concepts.
5. Organize the design modules in SV test bench.
6. Execute the ASIC verification using SV with help of EDA tools

Unit 1: Introduction to Verification**8 Hours**

Introduction, The Verification Process, The Verification Plan, White-Box Verification, Grey-Box Verification, Black-Box Verification, The Verification Methodology, Basic Test-bench Functionality, Directed Testing, Methodology Basics, Constrained-Random Stimulus, Functional Coverage, Test-bench Components, Layered Test-bench, Simulation Environment Phases, Maximum Code Reuse.

Unit 2: Data Types in System Verilog**8 Hours**

Introduction to Data Types, Built-in Data Types, Fixed-Size Arrays, Dynamic Arrays, Queues, Associative Arrays, Linked Lists, Array Methods, Creating New Types with typedef, Creating User-Defined Structures, Enumerated Types, Constants, Strings, Net Types, String Methods, Operators

Unit 3: Procedural Statements, Tasks and Functions**8 Hours**

Introduction, Procedural Statements, Task and Function Overview, Tasks, Functions and Void Functions, Routine Arguments, Returning from a Routine, Local Data Storage, Time Values, Enumerated Types, Repeat and Forever.

Unit 4: Basics of OOP for System Verilog**8 Hours**

Introduction, OOP Terminology, Creating New Objects, Object Deallocation, Using Objects, Static Variables vs Global Variables, Class Routines, Defining Routines Outside of the Class, Scoping Rules, Using One Class Inside Another, Understanding Dynamic Objects, Copying Objects, Building a Test-bench, Nested-Inner Class, Anonymous Classes, Generic Class Types.

Unit 5: Integration of Testbench and Design**8 Hours**

Introduction, Separating the Test-bench and Design, The Interface Construct, Stimulus Timing, Interface Driving and Sampling, Top-Level Scope, Program-Module Interactions, SystemVerilog Assertions, The Four-Port ATM Router, Interprocess Communication: Semaphore, Mailboxes and Event.

Unit 6: ASIC Verification Using EDA Tools**8 Hours**

Current Trends in Testing and Verification, Advanced Verification Methodologies: UVM Basic Levels, EDA Development Environments – Case Study

Total 48 Hours**Textbooks:**

1. Chris Spear, "System Verilog for Verification: A Guide to Learning the Test bench Language Features", Springer-Verlag New York, Inc. Secaucus, NJ, USA, 2006
2. Donald Thomas, "Logic Design and Verification Using System Verilog", Create Space Independent Publishing Platform, 2014.
3. Bergeron, J. "Writing Testbenches Using SystemVerilog Springer." 0-387-29221-7, BusinessMedia (2006).

Reference Books:

1. Language Reference Manual for System Verilog

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL116T - Memory Architectures

Major Core Elective IV (VLSI)

3 - 0 - 0 - 3

Course Outcomes:

At the end of the course, students will be able to

1. Illustrate the RAM technologies, the architecture of SRAM
2. Interpret the architectures of DRAM, DRAM controllers, and their applications
3. Demonstrate the applications of SRAMs and DRAMs
4. Demonstrate the operation of Non-Volatile Memories
5. Abstract the Semiconductor Memory Reliability and Radiation Effects
6. Abstract the process and design issues involved in Semiconductor Memories

Unit 1: Introduction

8 Hours

Band diagrams; threshold voltage; body bias effect; drain current and gate current characteristics; subthreshold slope; hot electron effect; various leakages in a MOSFET; tunneling phenomenon; direct tunneling; direct band to band tunneling; SOI MOSFET; PDSOI; FDSOI; classification of memories.

Unit 2: SRAM Technologies and Architectures

8 Hours

Static Random-Access Memories (SRAMs), SRAM Cell Structures, MOS SRAM Architecture, MOS SRAM Cell and Peripheral Circuit, Bipolar SRAM, Advanced SRAM Architectures, Application Specific SRAMs.

Unit 3: DRAM Technologies and Memory Controllers

8 Hours

DRAMs, MOS DRAM Cell, BiCMOS DRAM, Error Failures in DRAM, Advanced DRAM Design and Architecture, Application Specific DRAMs, SRAM and DRAM Memory Controllers.

Unit 4: Non-Volatile Memory Technologies

8 Hours

Masked ROMs, PROMs, Bipolar & CMOS PROM, EEPROMs, Floating Gate EPROM Cell, OTP EPROM, EEPROMs, Non-Volatile SRAM, Flash Memories.

Unit 5: Semiconductor Memory Reliability and Radiation Effects

8 Hours

Semiconductor Memory Reliability and Radiation Effects, General Reliability Issues, RAM Failure Modes and Mechanisms, Non-Volatile Memory Reliability, Radiation Effects, SEP, Radiation Hardening Techniques.

Unit 6: Process and Design Issues in Semiconductor Memories

8 Hours

Process and Design Issues, Radiation Hardened Memory Characteristics, Radiation Hardness Assurance, Testing, Memory Repair.

Total 48 Hours

Textbooks:

1. Ashok K Sharma, "Advanced Semiconductor Memories: Architectures, Designs and Applications", Wiley Inter science
2. KiyooItoh, "VLSI memory chip design", Springer International Edition
3. Wang, Laung-Terng, Cheng-Wen Wu, and Xiaoqing Wen. "VLSI test principles and architectures: Design for testability", Elsevier, 2006.

Reference Books:

1. Ashok K Sharma, "Semiconductor Memories: Technology, Testing and Reliability", PHI

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks	Final Weightage
Continuous Assessment (CA1-40, CA2-40)	40 (Computed as 80% of the higher Assessment score + 20% of the lower Assessment score)	30 Marks
Assignment Test	30	10 Marks
End Semester Assessment (ESA)	60	60 Marks
Total	130	100 Marks

26MTVL103L – Physical Design Automation Lab**0 - 0 - 3 - 1.5****Course Outcomes:****At the end of the course, students will be able to**

1. 1. Design and implement the fundamental digital logic circuits using Verilog HDL
2. 2. Make use of system level design
3. 3. Create the technology schematics based on the design
4. 4. Illustrate the design rule checks and timing parameters
5. 5. Explain the resources consumed by the design on FPGA
6. 6. Illustrate the Interconnections during place and route

List of Experiments:

1. Design and simulate basic logic gates (AND, OR, NOT, NAND, NOR, XOR, XNOR) using Verilog HDL
2. Design and simulate Half Adder and Full Adder using Verilog HDL
3. Design and simulate Half Subtractor and Full Subtractor using Verilog HDL
4. Design and simulate 4:1 Multiplexer and 1:4 Demultiplexer using Verilog HDL
5. Design and simulate Encoder and Decoder circuits using Verilog HDL
6. Design and simulate a 4-bit Magnitude Comparator using Verilog HDL
7. Design and simulate a 4-bit Ripple Carry Adder using structural modeling
8. Design and simulate an Arithmetic Logic Unit (ALU) using Verilog HDL
9. Design and simulate SR, JK, D, and T Flip-Flops using Verilog HDL
10. Design and simulate Shift Registers (SISO, SIPO, PISO, PIPO) using Verilog HDL
11. Design and simulate Synchronous and Asynchronous Counters using Verilog HDL
12. Generate RTL schematic and technology schematic for a 4-bit ALU
13. Perform Design Rule Check (DRC), timing analysis, and verify setup/hold time reports
14. Synthesize combinational and sequential circuits and analyze FPGA resource utilization (LUTs, FFs, I/O Blocks)
15. Implement and verify a Traffic Light Controller or Vending Machine Controller using FSM design
16. Perform Place and Route (PAR) for a digital design and analyze routing/interconnection reports on FPGA

Total 32 Hours**Textbooks:**

1. *T.R. Padmanabhan and B. Bala Tripura Sundari, Design through Verilog HDL, WSE, 2004.*
2. *Stephen. Brown and Zvonko Vranesic, Fundamentals of Logic Design with Verilog, TMH, 2005.*
3. *Michael D. Ciletti, Advanced Digital Design with Verilog HDL, PHI, 2005.*
4. *J. Bhaskar, A Verilog Primer, BSP, 2003.*

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks
Continuous Assessment <i>(Preparation, Observation & Results – 10, Record – 5, Viva-Voce – 15, Internal Test – 10)</i>	40
End Semester Assessment (ESA) <i>(Procedure & Write-up – 10, Conduct of Experiment/Program – 15, Analysis & Results – 20, Record – 5, Viva-Voce – 10)</i>	60
Total	100

26MTVL104L - IoT System Design Lab**0 - 0 - 3 - 1.5****Course Outcomes:****At the end of the course, students will be able to**

1. Install, configure, and develop programs using an embedded target board IDE, and interface basic input/output peripherals such as keypads and LCDs for embedded applications.
2. Interface and control actuators, including DC/Stepper motors and relays, using embedded target boards and appropriate control techniques such as PWM.
3. Design embedded systems by interfacing sensors and communication modules, including temperature, humidity, Bluetooth, and camera modules, for data acquisition and basic processing applications.
4. Configure cloud-based IoT platforms and develop applications for real-time monitoring of sensor data using Things peak cloud services.
5. Implement IoT-based remote monitoring and control systems using Blynk cloud platform and MQTT-based publisher-subscriber communication protocols.
6. Design, develop, and integrate complete IoT solutions by combining embedded hardware, sensors, actuators, communication technologies, cloud platforms, and application software for real-world applications.

List of Experiments:

Simulate a basic Common Source amplifier

1. Installation and setup of embedded target board IDE
2. Interfacing of keypad with embedded target board
3. Interfacing of LCD with embedded target board
4. Interfacing of DC/Stepper motor with embedded target board and controlling it's speed using PWM Technique
5. Interfacing of high voltage device using relay with embedded target board
6. Interfacing a temperature and humidity sensor with embedded target board and controls it using Bluetooth
7. Interfacing a Camera Module with embedded target board for image capture and basic image processing.
8. Configure Things peak cloud platform
9. Interface a temperature sensor and monitor real-time temperature value on Things peak cloud platform
10. Interface a ultrasonic sensor and monitor real-time value of the obstacle on Things peak cloud platform
11. Interface a pressure sensor with embedded target board and monitor real-time value of the pressure on Things peak cloud platform
12. Configure Blynk cloud platform
13. Interface actuator with embedded target board and control it's operation by using Blynk cloud platform

14. MQTT Communication for IoT Applications Implementing publisher-subscriber communication between NodeMCU, Raspberry Pi, and MQTT broker.
15. Mini Project – Integrated Embedded IoT System Design and implementation of a Smart Agriculture Monitoring complete IoT application
16. Mini Project – Integrated Embedded IoT System Design and implementation of a Smart Home Automation complete IoT application

Total 32 Hours

Textbook (s)

1. Raj Kamal, "Internet of Things: Architecture and Design Principles". TMH Publications, 2017.
2. Yogesh Misra, Programming and Interfacing with Arduino, CRC Press, 1st Edition, 2021

Reference (s)

1. Vijay Madiseti and Arshdeep Bahga, "Internet of Things (A Hands-on-Approach)", 1st Edition, VPT, 2014
2. Behrouz A. Forouzan, Data Communications and Networking, 5th Edition, Tata McGraw Hill Higher Education, 2013
3. Boyle, From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence, 1st Edition, Academic Press, 2014. 2M Communications, ISBN: 978-1-118-47347-

Evaluation Pattern: 40:60

Assessment Component	Maximum Marks
Continuous Assessment <i>(Preparation, Observation & Results – 10, Record – 5, Viva-Voce – 15, Internal Test – 10)</i>	40
End Semester Assessment (ESA) <i>(Procedure & Write-up – 10, Conduct of Experiment/Program – 15, Analysis & Results – 20, Record – 5, Viva-Voce – 10)</i>	60
Total	100