MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE

(Deemed to be University)

MADANAPALLE

www.mits.ac.in

Estd 1998

MASTER OF TECHNOLOGY

VLSI Design and Embedded Systems

Course Structure

&

Detailed Syllabi

For the students admitted to

Master of Technology in VLSI Design and Embedded Systems from the Academic Year 2025 – 26 Batch onwards

M.Tech Regular Two Year P.G. Degree Course

MADANAPALLE INSTITUTE OF TECHNOLOGY & SCIENCE

(Deemed to be University)

MADANAPALLE

M. Tech Two Year Curriculum Structure

Branch: VLSI Design and Embedded Systems

Total Credits	80 Credits for 2025 Admitted Batch onwards

M. Tech VLSI Design and Embedded Systems

R25 - Curriculum Structure
I Year I Semester

S. No.	Category	Course Code	Course Title	Hours Per Week				Credits
				L	T	P	Total	
1	PC	25MBVESTC01	CMOS Digital IC Design	3	0	0	3	3
2	PC	25MBVESTC02	Advanced Microcontrollers and Signal Processors	3	0	0	3	3
3	MC	25MBCOMMC01	Research Methodology and IPR	2	0	0	2	2
4	DE		**Discipline Elective-I** *(Refer ANNEXURE - I)*	3	0	0	3	3
5	DE		**Discipline Elective-II** *(Refer ANNEXURE - I)*	3	0	0	3	3
6	PC	25MBVESLC01	CMOS Digital IC Design Laboratory	0	0	4	4	2
7	PC	25MBVESLC02	Advanced Microcontrollers and Signal Processors Laboratory	0	0	4	4	2
8	SEC	25MBVESSC01	Scientific Computing	1	0	2	3	2
9	MC	25MBCIVMC01	Disaster Management	2	0	0	2	0
Total				**17**	**0**	**10**	**27**	**20**

I Year II Semester

S. No.	Category	Course Code	Course Title	Hours Per Week				Credits
				L	T	P	Total	
1	PC	25MBVESTC03	CMOS Analog IC Design	3	0	0	3	3
2	PC	25MBVESTC04	Embedded System Design	3	0	0	3	3
3	DE		**Discipline Elective-III**	3	0	0	3	3
4	DE		**Discipline Elective-IV**	3	0	0	3	3
5	OE		**Open Elective-I** *(Refer ANNEXURE - II)*	2	0	0	2	2
6	PC	25MBVESLC03	CMOS Analog IC Design Laboratory	0	0	4	4	2
7	PC	25MBVESLC04	Embedded System Design Laboratory	0	0	4	4	2
8	SEC	25MBVESSC02	QNX Based RTOS	1	0	2	3	2
Total				**15**	**0**	**10**	**25**	**20**

(L = Lecture, T = Tutorial, P = Practical, C = Credit

M. Tech VLSI Design and Embedded Systems

II Year I Semester (Tentative Structure)

S. No.	Category	Course Code	Course Title	Hours Per Week				Credits
				L	T	P	Total	
1	PC	25MBVESTC05	Physical Design Automation	3	0	0	3	3
2	PE		**Discipline Elective – V** *(Refer ANNEXURE - I)*	3	0	0	3	3
3	OE		**Open Elective- II** *(Refer ANNEXURE - II)*	3	0	0	3	3
4	SEC		**Skill Enhancement Course**	1	0	2	3	2
5	PR	25MBVESIC01	Internship*	0	0	6	6	3
6	PR	25MBVESPC01	Dissertation Phase I	0	0	20	20	10
Total				**10**	**0**	**28**	**38**	**24**

* 6 Weeks Internship during I Year II Semester Summer Break and to be evaluated in II Year I Semester

II Year II Semester (Tentative Structure)

S. No.	Category	Course Code	Course Title	Hours Per Week				Credits
				L	T	P	Total	
1	PR	25MBVESPC02	Dissertation Phase II	0	0	32	32	16
Total				**0**	**0**	**32**	**32**	**16**

(L = Lecture, T = Tutorial, P = Practical, C = Credit

ANNEXURE - I

LIST OF DISCIPLINE ELECTIVE COURSES

Discipline Elective – I		
Sl. No.	**Course Code**	**Course Title**
1.	25MBVESDC01	Communication Buses and Interfaces
2.	25MBVESDC02	Data Acquisition System Design
3.	25MBVESDC03	FPGA Architectures and Applications
Any advanced courses can be appended in future.		

Discipline Elective – II		
Sl. No.	**Course Code**	**Course Title**
1.	25MBVESDC04	Low Power VLSI Design
2.	25MBVESDC05	Nano-materials and Nanotechnology
3.	25MBVESDC06	Network Security and Cryptography
Any advanced courses can be appended in future.		

Discipline Elective – III		
Sl. No.	**Course Code**	**Course Title**
1.	25MBVESDC07	Pattern Recognition and Machine Learning
2.	25MBVESDC08	Programming Languages for Embedded Software
3.	25MBVESDC09	RF IC Design
Any advanced courses can be appended in future.		

Discipline Elective – IV		
Sl. No.	**Course Code**	**Course Title**
1.	25MBVESDC10	SoC Architecture
2.	25MBVESDC11	System Design with Embedded Linux
3.	25MBVESDC12	High Speed VLSI Design
Any advanced courses can be appended in future.		

ANNEXURE - II

LIST OF OPEN ELECTIVE COURSES

Open Elective - I (To be offered under MOOC's Category from SWAYAM – NPTEL)		
Sl. No.	**Course Code**	**Course Title**
1.	25MBCIVOM01	Remote Sensing and GIS
2.	25MBCIVOM02	Sustainable Engineering Concepts and Life Cycle Analysis
3.	25MBMECOM01	Product Engineering and Design Thinking
4.	25MBMBAOM01	Innovation, Business Models and Entrepreneurship
Any other new Inter-Disciplinary Course offered by SWAYAM NPTEL which doesn't exist in the Curriculum can be appended in future.		

I Year I Semester

M. Tech I Year I Semester

25MBVESTC01 CMOS DIGITAL IC DESIGN

L	T	P	C
3	0	0	3

Pre-requisite Digital system design

Course Objectives:

This course enables students to

1. To understand the fundamentals of CMOS technology and its significance in digital circuit design.
2. To design and analyze basic and complex CMOS logic gates and circuits.
3. To teach fundamentals of CMOS Digital integrated circuit design such as importance of Pseudo logic, Combinational MOS logic circuits and Sequential MOS logic circuits.
4. To understand the trade-offs involved in optimizing power, performance, and area in CMOS digital circuits.
5. To gain knowledge of advanced topics in CMOS design, such as low power techniques, high-speed design, and emerging technologies in semi-conductor memories

UNIT I MOS DESIGN PSEUDO NMOS LOGIC 9 hours

Inverter, Inverter threshold voltage, Output high voltage, Output Low voltage, Gain at gate threshold voltage, Transient response, Rise time, Fall time, Pseudo NMOS logic gates, Transistor equivalency, CMOS Inverter logic.

UNIT II COMBINATIONAL MOS LOGIC CIRCUITS 9 hours

MOS logic circuits with NMOS loads, Primitive CMOS logic gates–NOR & NAND gate, Complex Logic circuits design–Realizing Boolean expressions using NMOS gates and CMOS gates, AOI and OIA gates, CMOS full adder, CMOS transmission gates, Designing with Transmission gates.

UNIT III SEQUENTIAL MOS LOGIC CIRCUITS 9 hours

Behavior of bistable elements, SR Latch, Clocked latch and flip flop circuits, CMOS D latch and edge triggered flip-flop

UNIT IV DYNAMIC LOGIC CIRCUITS 9 hours

Basic principle, Voltage Bootstrapping, Synchronous dynamic pass transistor circuits, Dynamic CMOS transmission gate logic, High performance Dynamic CMOS circuits.

UNIT V SEMICONDUCTOR MEMORIES 9 hours

Types, RAM array organization, DRAM – Types, Operation, Leakage currents in DRAM cell and refresh operation, SRAM operation Leakage currents in SRAM cells, Flash Memory-NOR flash and NAND flash.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Demonstrate advanced knowledge in Static and dynamic characteristics of CMOS,
CO2: Estimate Delay and Power of Adders circuits.
CO3: Classify different semiconductor memories.
CO4: Analyze, design and implement combinational and sequential MOS logic circuits.
CO5: Analyze complex engineering problems critically in the domain of digital IC design for conducting research.

Text Book(s)

1. Neil Weste, David Harris, "CMOS VLSI Design: A Circuits and Systems Perspective", 4th Edition, Pearson, 2010
2. Digital Integrated Circuit Design – Ken Martin, Oxford University Press, 2011.
3. CMOS Digital Integrated Circuits Analysis and Design – Sung-Mo Kang, Yusuf Leblebici, TMH, 3rd Edition, 2011.

Reference Books

1. Introduction to VLSI Systems: A Logic, Circuit and System Perspective – Ming-BO Lin, CRC Press, 2011
2. Digital Integrated Circuits – A Design Perspective, Jan M. Rabaey, Anantha Chandrakasan, Borivoje Nikolic, 2ndEdition, PHI.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

M. Tech I Year I Semester

25MBVESTC02 ADVANCED MICROCONTROLLERS AND SIGNAL PROCESSORS

L	T	P	C
3	0	0	3

Pre-requisite Microprocessor and Microcontroller

Course Objectives:

This course enables students to

1. To understand the architecture and applications of the ARM Cortex-Mx processor.
2. To comprehend the types, priority, and behaviour of exceptions and interrupts.
3. To study the architectural and advanced features of the LPC 17xx microcontroller.
4. To understand the architecture and features of programmable DSP processors.
5. To explore the architecture and programming considerations of the TMS320C62x, TMS320C64x, and TMS320C67x processors.

UNIT I **9 hours**

ARM Cortex-Mx Processor: Applications, Programming model – Registers, Operation - modes, Exceptions and Interrupts, Reset Sequence, Instruction Set (ARM and Thumb), Unified Assembler Language, Memory Maps, Memory Access Attributes, Permissions, Bit-Band Operations, Unaligned and Exclusive Transfers. Pipeline, Bus Interfaces.

UNIT II **9 hours**

Exceptions and Interrupts: Exceptions, Types, Priority, Vector Tables, Interrupt Inputs and Pending behaviour, Fault Exceptions, Supervisor and Pendable Service Call, Nested Vectored Interrupt Controller, Basic Configuration, SYSTICK Timer, Interrupt Sequences, Exits, Tail Chaining, Interrupt Latency.

UNIT III **9 hours**

LPC 17xx microcontroller: Internal memory, GPIOs, Timers, ADC, UART and other serial interfaces, PWM, RTC, WDT.

UNIT IV **9 hours**

Programmable DSP (P-DSP) Processors: Harvard architecture, Multi port memory, architectural structure of P-DSP- MAC unit, Barrel shifters, Introduction to TI DSP processor family.

UNIT V **9 hours**

VLIW architecture and TMS320C6000 series, architecture study, data paths, cross paths,
Introduction to Instruction level architecture of C6000 family, Assembly Instructions memory addressing, for arithmetic, logical operations.

TMS320C55x – Architecture overview, Addressing modes, Instruction set, Programming considerations, system issues.

TMS320C62x and TMS320C64x: Architecture overview, Memory systems, External memory addressing, Instruction set, Programming considerations, system issues.

TMS320C67X – Architecture overview, Instruction set, Pipeline Architecture, Programming considerations, Realtime implementations

Course Outcomes:

Upon the completion of the course, the student is able to

CO1: Explain the architecture, applications, and programming model of the ARM Cortex-Mx processor

CO2: Explain and configure different types of exceptions and interrupts.

CO3: Describe the architecture and peripheral interfaces of the LPC 17xx microcontroller and its advance features.

CO4: Explain the architecture and key features of programmable DSP processors.

CO5: Describe the architecture, instruction set, and programming considerations of various TI DSP processors.

Text Book(s)

1. N. Senthil Kumar and M Saravanan, "Microprocessors and Microcontrollers" 2nd Edition, Oxford university press, 2016.
   Joseph Yiu, "The definitive guide to ARM Cortex-M3", Elsevier, 2nd Edition
2. Venkatramani B. and Bhaskar M. "Digital Signal Processors: Architecture, Programming and Applications", TMH, 2nd Edition.
3. TMS Manual on TMS320C62XX, TMS320C64XX and TMS320C67XX.

Reference Books

1. Sloss Andrew N, Symes Dominic, Wright Chris, "ARM System Developer's Guide: Designing and Optimizing", Morgan Kaufman Publication.
2. Steve furber, "ARM System-on-Chip Architecture", Pearson Education
3. Frank Vahid and Tony Givargis, "Embedded System Design", Wiley
4. Technical references and user manuals on www.arm.com, NXP Semiconductor www.nxp.com and Texas Instruments www.ti.com

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

M. Tech I Year I Semester

25MBCOMMC01 RESEARCH METHODOLOGY AND IPR

L	T	P	C
2	0	0	2

Pre-requisite Nil

Course Description:

This course aims to provide students with a comprehensive understanding of research methodology and the principles and practices of intellectual property rights (IPR). The course will equip students with the skills needed to design, conduct, and evaluate research effectively while also understanding the legal and ethical considerations surrounding intellectual property.

Course Objectives:

To impart knowledge on

1. Formulation of research problems, design of experiment, collection of data, interpretation and presentation of result
2. Intellectual property rights, patenting and licensing

UNIT I RESEARCH PROBLEM FORMULATION 9 hours

Objectives of research, types of research, research process, approaches to research; conducting literature review- information sources, information retrieval, tools for identifying literature, Indexing and abstracting services, Citation indexes, summarizing the review, critical review, identifying research gap, conceptualizing and hypothesizing the research gap

UNIT II RESEARCH DESIGN AND DATA COLLECTION 9 hours

Statistical design of experiments- types and principles; data types & classification; data collection - methods and tools

UNIT III DATA ANALYSIS, INTERPRETATION AND REPORTING 9 hours

Sampling, sampling error, measures of central tendency and variation,; test of hypothesis- concepts; data presentation- types of tables and illustrations; guidelines for writing the abstract, introduction, methodology, results and discussion, conclusion sections of a manuscript; guidelines for writing thesis, research proposal; References – Styles and methods, Citation and listing system of documents; plagiarism, ethical considerations in research

UNIT IV INTELLECTUAL PROPERTY RIGHTS 9 hours

Concept of IPR, types of IPR – Patent, Designs, Trademarks and Trade secrets, Geographical indications, Copy rights, applicability of these IPR, IPR & biodiversity; IPR development process, role of WIPO and WTO in IPR establishments, common rules of IPR practices, types and features of IPR agreement, functions of UNESCO in IPR maintenance.

UNIT V PATENTS 9 hours

Patents – objectives and benefits of patent, concept, features of patent, inventive steps, specifications, types of patent application; patenting process - patent filling, examination of patent, grant of patent, revocation; equitable assignments; Licenses, licensing of patents; patent agents, registration of patent agents.

Course Outcomes:

Upon completion of the course, the student can

CO1: Describe different types of research; identify, review and define the research problem
CO2: Select suitable design of experiment s; describe types of data and the tools for collection of data
CO3: Explain the process of data analysis; interpret and present the result in suitable form
CO4: Explain about Intellectual property rights, types and procedures
CO5: Execute patent filing and licensing

Text Book(s)

1. Cooper Donald R, Schindler Pamela S and Sharma JK, "Business Research Methods", Tata McGraw Hill Education, 11e (2012).
2. Soumitro Banerjee, "Research methodology for natural sciences", IISc Press, Kolkata, 2022.
3. Catherine J. Holland, "Intellectual property: Patents, Trademarks, Copyrights, Trade Secrets", Entrepreneur Press, 2007.

Reference Books

1. David Hunt, Long Nguyen, Matthew Rodgers, "Patent searching: tools & techniques", Wiley, 2007.
2. The Institute of Company Secretaries of India, Statutory body under an Act of parliament, "Professional Programme Intellectual Property Rights, Law and practice", September 2013.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

M. Tech I Year I Semester

25MBVESLC01 CMOS DIGITAL IC DESIGN LABORATORY

L	T	P	C
0	0	4	2

Pre-requisite Digital Logic Design

Course Objectives:

This course enables students to

1. To explain the VLSI Design Methodologies using any VLSI design tool.
2. To grasp the significance of various design logic Circuits in full-custom IC Design.
3. To explain the Physical Verification in Layout Extraction.
4. To fully appreciate the design and analyze of CMOS Digital Circuits.
5. To grasp the Significance of Pre-Layout Simulation and Post-Layout Simulation.

List of Experiments

The students are required to design and implement the Circuit and Layout of any TEN Experiments using CMOS 130nm Technology.

1. Inverter Characteristics.
2. NAND and NOR Gate
3. XOR and XNOR Gate
4. 2:1 Multiplexer
5. Full Adder
6. RS-Latch
7. Clock Divider
8. JK-Flip Flop
9. Synchronous Counter
10. Asynchronous Counter
11. Static RAM Cell
12. Dynamic Logic Circuits
13. Linear Feedback Shift Register

Lab Requirements:

Software:
Mentor Graphics Tool/ Cadence/ Synopsys/Industry Equivalent Standard Software
Hardware:
Personal Computer with necessary peripherals, configuration and operating System.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Explain the VLSI Design Methodologies using any VLSI design tool.
CO2: Grasp the significance of various design logic Circuits in full-custom IC Design.
CO3: Explain the Physical Verification in Layout Extraction.
CO4: Fully appreciate the design and analyze of CMOS Digital Circuits.
CO5: Grasp the Significance of Pre-Layout Simulation and Post-Layout Simulation.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination.

M. Tech I Year I Semester

25MBVESLC02 ADVANCED MICROCONTROLLERS AND SIGNAL PROCESSORS LABORATORY

L	T	P	C
0	0	4	2

Pre-requisite Microprocessor and Microcontroller

Course Objectives:

This course enables students to

1. To write the ARM 'C' programming for applications
2. To understand the interfacing of various modules with ARM 7/ ARM Cortex-M3
3. To develop assembly and C Programming for DSP processors

List of Experiments

Part A) Experiments to be carried out on Cortex-Mx development boards and using GNU tool-chain

1. Blink an LED with software delay, delay generated using the SysTick timer.
2. System clock real time alteration using the PLL modules.
3. Control intensity of an LED using PWM implemented in software and hardware.
4. Control an LED using switch by polling method, by interrupt method and flash the LED once every five switch presses.
5. UART Echo Test.
6. Take analog readings on rotation of rotary potentiometer connected to an ADC channel.
7. Temperature indication on an RGB LED.
8. Mimic light intensity sensed by the light sensor by varying the blinking rate of an LED.
9. Evaluate the various sleep modes by putting core in sleep and deep sleep modes.
10. System reset using watchdog timer in case something goes wrong.
11. Sample sound using a microphone and display sound levels on LEDs.

Part B) Experiments to be carried out on DSP C6713 evaluation kits and using Code Composer Studio (CCS)

12. To develop an assembly code and C code to compute Euclidian distance between any two points
13. To develop assembly code and study the impact of parallel, serial and mixed execution
14. To develop assembly and C code for implementation of convolution operation
15. To design and implement filters in C to enhance the features of given input sequence/signal

Software Requirements:

Keil for ARM, Code Composer Studio

Hardware Requirements:

ARM Cortex Mx Development Boards, TI TMS C6713 evaluation kit

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Install, configure and utilize tool sets for developing applications based on ARM processor core.

CO2: Design and developtheARM7 based embedded systems for various applications.

CO3: Develop application programs on ARM and DSP development boards both in assembly and C

CO4: Design and implement the digital filters on DSP6713 processor.

CO5: Analyze the hardware and software interaction and integration.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination.

M. Tech I Year I Semester

25MBVESSC01 SCIENTIFIC COMPUTING

L	T	P	C
1	0	2	2

Pre-requisite MATLAB

Course Objectives:

This course enables students to

1. To introduce the use of MATLAB as a modeling and simulation tool for VLSI and embedded system applications.
2. To provide knowledge of digital filter design, image processing, communication systems, and data compression algorithms and their hardware realization.
3. To develop skills in designing and verifying digital arithmetic units and VLSI architectures prior to FPGA implementation.
4. To enable students to design and test embedded system algorithms, including high-level control, data acquisition, and hardware-in-the-loop testing.
5. To build the ability to integrate simulation results with FPGA/embedded hardware for performance evaluation and optimization in real-time applications.

UNIT I **6 hours**

Design and analysis of FIR and IIR filters using MATLAB; windowing and bilinear transform methods; frequency response, stability, and performance evaluation; comparison of MATLAB designs with FPGA hardware implementations. Fundamentals of image processing; spatial and frequency domain filtering; image compression techniques; Discrete Cosine Transform (DCT) based compression; MATLAB modeling of filtering and compression algorithms; hardware realization of image processing functions on FPGA.

- Digital Filter Design & Analysis:
  Design FIR and IIR filters using MATLAB and then analyze their performance and compare it w hardware implementations on an FPGA.
- Image Processing Algorithms:
  Implement algorithms for image compression (e.g., DCT) or filtering in MATLAB to model the behavior of a hardware implementation on an FPGA

UNIT II **6 hours**

Modeling and simulation of communication systems using MATLAB; spread-spectrum techniques and CDMA modem design; Bluetooth voice transmission modeling and performance verification; comparison of MATLAB simulation with hardware realization. Fundamentals of data compression; entropy, redundancy, and coding principles; design and analysis of Huffman coding and other compression algorithms using MATLAB; evaluation of efficiency and feasibility for FPGA implementation.

- Communication Systems:
  Model and simulate complex systems like CDMA modems or Bluetooth voice transmission in MATLAB to verify their functionality before hardware implementation.
- Data Compression Algorithms:
  Design and analyze Huffman coding or other data compression algorithms using MATLAB.

UNIT III **6 hours**

Simulation and verification of digital arithmetic units including ripple carry adder, carry skip adder, and floating-point ALU using MATLAB; FPGA implementation and performance analysis on Xilinx platforms. Development and testing of high-level control algorithms for embedded systems; applications in automotive and robotics; MATLAB-based modeling and validation of control strategies prior to hardware deployment.

- Digital Arithmetic Units:
  Simulate and verify the functionality of adders (ripple carry, carry skip), or a floating-point ALU, which can then be implemented on a Xilinx FPGA.
- High-Level Control Algorithm Development:
  Develop and test high-level control algorithms for embedded systems, such as those in automotive or robotics, using MATLAB.

UNIT IV **6 hours**

Data acquisition from embedded systems and sensors; interfacing and data transfer to PC; MATLAB-based data processing and high-level algorithm development. Design of MATLAB-based GUIs for real-time control of embedded devices; applications such as voice-operated fire extinguishers and remote-controlled robotic arms; integration of GUI with hardware for monitoring and control.

- Data Acquisition & Analysis:
  Collect data from an embedded system (e.g., sensors) and transfer it to a PC to use MATLAB for high-level algorithm development and analysis.
- GUI-based Control:
  Design a MATLAB-based GUI to control an embedded device, such as a voice-operated fire extinguisher or a remote-controlled robotic arm.

UNIT V SEMICONDUCTOR MEMORIES **6 hours**

Principles of Hardware-in-the-Loop (HIL) simulation; modeling and simulating environments in MATLAB; generating real-time test inputs for embedded systems; integration of MATLAB with embedded hardware; performance evaluation and validation without full deployment.

- Hardware-in-the-Loop (HIL) Testing:
  Use MATLAB to simulate the environment and generate test inputs for a real embedded system, enabling testing without full deployment.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Apply MATLAB for modeling, simulation, and analysis of digital systems, including filters, image processing, and communication systems.

CO2: Design and verify VLSI architectures such as digital filters, arithmetic units, and compression algorithms before FPGA implementation.

CO3: Develop and test embedded system algorithms for control, data acquisition, and hardware-in-the-loop applications using MATLAB.

CO4: Integrate MATLAB with FPGA/embedded hardware to validate system functionality through simulation and real-time testing.

CO5: Demonstrate problem-solving, design optimization, and performance evaluation skills in implementing real-world VLSI and embedded system applications.

Text Book(s)

1. Wayne Wolf, *Modern VLSI Design: System-on-Chip Design*, Pearson, 4th Edition, 2014.

2. Stephen Brown and Zvonko Vranesic, Fundamentals of Digital Logic with VHDL Design, McGraw Hill, 3rd Edition, 2013.
3. Amos Gilat, MATLAB: An Introduction with Applications, Wiley, 6th Edition, 2021

Reference Books

1. Charles H. Roth and Larry L. Kinney, Fundamentals of Logic Design, Cengage, 7th Edition, 2013.
2. M. Morris Mano and Michael D. Ciletti, Digital Design with an Introduction to the Verilog HDL, Pearson, 6th Edition, 2017.
3. Sanjay Churiwala, Designing with Xilinx FPGAs, Springer, 2017.

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

M. Tech I Year I Semester

25MBCIVMC01 DISASTER MANAGEMENT

L	T	P	C
2	0	0	0

Course Objectives:

Upon the completion of subject student will be able to-

1. Learn to demonstrate a critical understanding of key concepts in disaster risk reduction and humanitarian response.
2. Critically evaluate disaster risk reduction and humanitarian response policy and practice from multiple perspectives.
3. Develop an understanding of standards of humanitarian response and practical relevance in specific types of disasters and conflict situations.
4. Critically understand the strengths and weaknesses of disaster management approaches,
5. Planning and programming in different countries, particularly their home country or the countries they work in

UNIT-I DISASTER CLASSIFICATION 6 hours

Disaster: definition, factors and significance; difference between hazard and Disaster; natural disaster: earthquakes, volcanisms, cyclones, tsunamis, floods, droughts and famines, landslides and avalanches; man-made disasters: nuclear reactor meltdown, industrial accidents, oil slicks and spills, outbreaks of disease and epidemics, war and conflicts

UNIT-II REPERCUSSIONS OF DISASTERS 6 hours

Economic damage, loss of human and animal life, destruction of ecosystem. **Disaster Prone Areas in India:** Study of seismic zones; areas prone to floods and droughts, landslides and Avalanches; areas prone to cyclonic and coastal hazards with special reference to tsunami.

UNIT-III DISASTER PREPAREDNESS AND MANAGEMENT 6 hours

Preparedness: monitoring of phenomena triggering a disaster or hazard; Evaluation of risk: application of remote sensing, data from meteorological and Other agencies, media reports: governmental and community preparedness.

UNIT-IV RISK ASSESSMENT 6 hours

Disaster risk: concept and elements, disaster risk reduction, global and national disaster risk situation. Techniques of risk assessment, global co-operation in risk assessment and warning.

UNIT-V DISASTER MITIGATION 6 hours

Meaning, concept and strategies of disaster mitigation, emerging trends in Mitigation. Structural mitigation and non-structural mitigation, programs of Disaster mitigation in India.

Course outcomes:

After the completion of the subject following outcomes can be achieved-

CO1: Students will be able to understand disaster and its types in general.

CO2: They will understand the post disaster damage in terms of both like and commodity.

CO3: They will have clear picture of disaster-prone zones.

CO4: They will be able to understand the pre and post disaster preparedness needed to mitigate the disaster impact in large scale.

CO5: Student will also understand to quantify the risk in terms of monetary for both commodity and life.

CO6: Student will also learn the structural and non-structural measures for risk mitigation

Reference Books:

1. Ghosh G.K., 2006, Disaster Management, APH Publishing Corporation
2. R. Nishith, Singh AK, "Disaster Management in India: Perspectives, issues and strategies "'New Royal book Company.
3. Sahni, PardeepEt.Al. (Eds.)," Disaster Mitigation Experiences and Reflections", Prentice Hall Of India, New Delhi.
4. Goel S. L., Disaster Administration And Management Text and Case Studies" ,Deep&Deep Publication Pvt. Ltd., New Delhi

Mode of Evaluation: Assignments, Mid Term Tests

I Year II Semester

M. Tech I Year II Semester

25MBVESTC03 CMOS ANALOG IC DESIGN

L	T	P	C
3	0	0	3

Pre-requisite Semiconductor physics and devices/ Analog circuits

Course Objectives:

1. This course focuses on theory, analysis and design of analog integrated circuits in both Bipolar and Metal-Oxide-Silicon (MOS) technologies.
2. Basic design concepts, issues and tradeoffs involved in analog IC design are explored.
3. Intuitive understanding and real-life applications are emphasized throughout the course.
4. To learn about Design of CMOS Op Amps, Compensation of Op Amps, Design of Two-Stage Op Amps, Power Supply Rejection Ratio of Two-Stage Op Amps, Cascade Op Amps, Measurement Techniques of OP Amp.
5. To know about Characterization of Comparator, Two-Stage, Open-Loop Comparators, Improving the Performance of Open-Loop Comparators, Discrete-Time Comparators etc.

UNIT I **9 hours**

Basic MOS Device Physics: General Considerations, MOS I/V Characteristics, Second Order effects, MOS Device models and MOS Capacitor. Short Channel Effects and Device Models. Single Stage Amplifiers – Basic Concepts, Common Source Stage, Source Follower, Common Gate Stage, Cascode Stage.

UNIT II **9 hours**

Differential Amplifiers: Single Ended and Differential Operation, Basic Differential Pair, Common Mode Response, Differential Pair with MOS loads, Gilbert Cell. Passive and Active Current Mirrors – Basic Current Mirrors, Cascode Current Mirrors, Active Current Mirrors. Current Steering Circuit

UNIT III **9 hours**

Frequency Response of Amplifiers: General Considerations, Common Source Stage, Source Followers, Common Gate Stage, Cascode Stage, Differential Pair. Noise – Types of Noise, Representation of Noise in circuits, Noise in single stage amplifiers, Noise in Differential Pairs.

UNIT IV **9 hours**

Feedback Amplifiers: General Considerations, Feedback Topologies, Effect of Loading. Operational Amplifiers – General Considerations, One Stage Op Amps, Two Stage Op Amps, Gain Boosting, Common – Mode Feedback, Input Range limitations, Slew Rate, Power Supply Rejection, Noise in Op Amps, Stability and Frequency Compensation.

UNIT V **9 hours**

Comparators: Characterization of comparator, Two-Stage, Open-Loop comparators, Other Open-Loop Comparators, Improving the Performance of Open-Loop Comparators, Discrete-Time Comparators.

Course Outcomes:

CO1: Design MOSFET based analog integrated circuits.
CO2: Analyze analog circuits at least to the first order.
CO3: Appreciate the trade-offs involved in analog integrated circuit design.
CO4: Understand and appreciate the importance of noise and distortion in analog circuits.
CO5: Analyze complex engineering problems critically in the domain of analog IC design for conducting research.

Text Book(s)

1. B. Razavi, "Design of Analog CMOS Integrated Circuits", 2nd Edition, McGraw Hill, 2016.
2. Paul. R. Gray& Robert G. Meyer, "Analysis and Design of Analog Integrated Circuits", Wiley, 5th Edition, 2009.

Reference Books

1. T. C. Carusone, D. A. Johns & K. Martin, "Analog Integrated Circuit Design", 2ndEdition, Wiley, 2012.
2. P. E. Allen & D. R. Holberg, "CMOS Analog Circuit Design", 3rd Edition, Oxford University Press, 2011.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

M. Tech I Year II Semester

25MBVESTC04 EMBEDDED SYSTEM DESIGN

L	T	P	C
3	0	0	3

Pre-requisite Microcontrollers and Microprocessors, Digital Logic Design

Course Objectives:

This course enables students to

1. Understand the basics of an embedded system.
2. Analyse the typical components of an embedded system.
3. Understand different Embedded Firmware design approaches.
4. Learn the design process of embedded system applications.
5. Analyse the RTOS and inter-process communication.

UNIT I INTRODUCTION TO EMBEDDED SYSTEMS 9 hours

Definition of Embedded System, Embedded Systems Vs General Computing Systems, History of Embedded Systems, Classification, Major Application Areas, Purpose of Embedded Systems, Characteristics and Quality Attributes of Embedded Systems.

UNIT II TYPICAL EMBEDDED SYSTEM 9 hours

Core of the Embedded System: General Purpose and Domain Specific Processors, ASICs, PLDs, Commercial Off-The-Shelf Components (COTS), Memory: ROM, RAM, Memory according to the type of Interface, Memory Shadowing, Memory selection for Embedded Systems, Sensors and Actuators, Communication Interface: Onboard and External Communication Interfaces. DDR, Flash, NVRAM

UNIT III EMBEDDED FIRMWARE 9 hours

Reset Circuit, Brown-out Protection Circuit, Oscillator Unit, Real Time Clock, Watchdog Timer, **Embedded Firmware Design Approaches and Development Languages**:

Embedded firmware design approaches-super loop-based approach, operating system-based approach; embedded firmware development languages-assembly language-based development, high-level language-based development.

UNIT IV RTOS Based Embedded System Design 9 hours

Operating System Basics, Types of Operating Systems, Tasks, Processes and Threads, Multiprocessing and Multitasking, and Task Scheduling: Pre-emptive and Non-Premptive Scheduling, Interrupt Routines of RTOS.

UNIT V TASK COMMUNICATION 9 hours

Shared Memory, Message Passing, Semaphores, Remote Procedure Call and Sockets, Task Synchronization: Task Communication/Synchronization Issues, Task Synchronization Techniques, Device Drivers, How to Choose an RTOS.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the basics of an embedded system.
CO2: Analyse the typical components of an embedded system.
CO3: Understand different Embedded Firmware design approaches.
CO4: Learn the design process of embedded system applications.
CO5: Analyse the RTOS and inter-process communication.

Text Book(s)

1. Introduction to Embedded Systems - Shibu K.V, Mc Graw Hill.
2. Embedded Systems - Raj Kamal, TMH.

Reference Books

1. Embedded System Design - Frank Vahid, Tony Givargis, John Wiley.
2. An Embedded Software Primer - David E. Simon, Pearson Education.
3. Embedded Systems – Lyla, Pearson, 2013

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

M. Tech I Year II Semester

25MBVESLC03 CMOS ANALOG IC DESIGN LABORATORY

L	T	P	C
0	0	4	2

Pre-requisite Analog circuits

Course Objectives:

This course enables students to

1. To explain the VLSI Design Methodologies using VLSI design tool.
2. To grasp the significance of various CMOS analog circuits in full-custom IC Design flow
3. To explain the Physical Verification in Layout Design
4. To fully appreciate the design and analyze of analog and mixed signal simulation
5. To grasp the Significance of Pre-Layout Simulation and Post-Layout Simulation

List of Experiments

The students are required to design and implement any TEN Experiments using CMOS 130nm Technology.

The students are required to implement LAYOUTS of any SIX Experiments using CMOS 130nm Technology and Compare the results with Pre-Layout Simulation.

1. MOS Device Characterization and parametric analysis
2. Common Source Amplifier
3. Common Source Amplifier with source degeneration
4. Cascode amplifier
5. Simple current mirror
6. Cascode current mirror.
7. Wilson current mirror.
8. Differential Amplifier
9. Operational Amplifier
10. Sample and Hold Circuit
11. Direct-conversion ADC
12. R-2R Ladder Type DAC

Lab Requirements:

Software: Mentor Graphics – Pyxis Schematic, IC Station, Calibre, ELDO Simulator
Hardware: Personal Computer with necessary peripherals, configuration and operating System.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Explain the VLSI Design Methodologies using VLSI design tool.
CO2: Grasp the significance of various CMOS analog circuits in full-custom IC Design flow
CO3: Explain the Physical Verification in Layout Design
CO4: Fully appreciate the design and analyze of analog and mixed signal simulation
CO5: Grasp the Significance of Pre-Layout Simulation and Post-Layout Simulation

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination.

M. Tech I Year II Semester

25MBVESLC04 EMBEDDED SYSTEM DESIGN LABORATORY

L	T	P	C
0	0	4	2

Pre-requisite Embedded System

Course Objectives:

This course enables students to

1. To familiarize with embedded systems programming concepts
2. To implement different embedded communication and interfacing protocols

List of Experiments

1. Functional Testing of Devices
   Flashing the OS on to the device into a stable functional state by porting desktop environment withnecessary packages.
2. Exporting Display on to other Systems
   Making use of available laptop/desktop displays as a display for the device using SSH client & X11display server.
3. GPIO Programming
   Programming of available GPIO pins of the corresponding device using native programminglanguage. Interfacing of I/O devices like LED/Switch etc., and testing the functionality.
4. Interfacing Chronos eZ430
   Chronos device is a programmable Texas Instruments watch which can be used for multiple purposes like PPT control, Mouse operations etc., Exploit the features of the device by interfacingwith devices.
5. ON/OFF Control Based On Light Intensity
   Using the light sensors, monitor the surrounding light intensity & automatically turn ON/OFF thehigh intensity LED's by taking some pre-defined threshold light intensity value.
6. Battery Voltage Range Indicator
   Monitor the voltage level of the battery and indicating the same using multiple LED's (for ex: for 3V battery and 3 LEDs, turn on 3 LED s for 2-3V, 2 LEDs for 1-2V, 1 LED for 0.1-1V & turn offall for 0V)
7. Dice Game Simulation
   Instead of using the conventional dice, generate a random value similar to dice value and display thesame using a 16X2 LCD. A possible extension could be to provide the user with option of selecting single or double dice game.
8. Displaying RSS News Feed On Display Interface
   Displaying the RSS news feed headlines on a LCD display connected to device. This can be adaptedto other websites like twitter or other information websites. Python can be used to acquire data fromthe internet.
9. Porting Open w.r.t the Device
   Attempt to use the device while connecting to a WiFi network using a USB dongle and at the sametime providing a wireless access point to the dongle.
10. Hosting a website on Board
    Building and hosting a simple website(static/dynamic) on the device and make it accessible online.There is a need to install server (eg: Apache) and thereby host the website.
11. Webcam Server

Interfacing the regular USB webcam with the device and turn it into fully functional IP webcam &test the functionality.

12. FM Transmission
    Transforming the device into a regular FM transmitter capable of transmitting audio at desiredfrequency (generally 88-108 Mhz)

Software Requirements:
Keil / Python
Hardware Requirements:
Arduino/Raspbery Pi/Beaglebone

Course Outcomes:
Upon successful completion of the course, students will be able to
CO1: Understand the architecture and programming of embedded systems.
CO2: Apply programming skills to develop embedded system applications.
CO3: Design and implement hardware-software integration for embedded systems.
CO4: Analyze and debug embedded system applications using simulation and debugging tools.
CO5: Develop real-time embedded system projects using modern development platforms.

Mode of Evaluation: Continuous Internal Evaluation and End Semester Examination.

M. Tech I Year II Semester

25MBVESSC02 QNX BASED RTOS

L	T	P	C
1	0	2	2

Pre-Requisite Nil

Course Description:
This laboratory course emphasizes to the students to understand the concepts of QNX Embedded Real time operating systems. This course covers the Micro Kernal OS, different types of policies, Process, Multi-threading multi-resource services for real time environment. It also covers the High availability and Reliability Design with commercial QNX Momentics IDE.

Course Objectives
By the end of this course, students will:

1. Understand the architecture and core concepts of the QNX RTOS and Learn to develop and debug applications using the QNX Momentics IDE. (L2)
2. Apply the knowledge of process and thread management, including synchronization techniques. (L3)
3. Construct inter-process communication (IPC) methods and their applications in QNX. (L3)
4. Understand hardware programming concepts, including interrupt handling and memory access. (L2)
5. Develop to build and configure QNX boot/OS images for embedded systems. (L6)

UNIT I INTRODUCTION TO QNX OS ARCHITECTURE 6 hours
Brief history of Embedded Systems and Real Time Systems, Overview of QNX OS architecture: microkernel, process manager, and standards, Protected address spaces, process/thread model, and scheduling, Introduction to inter-process communication (IPC) and synchronization, Resource managers and shared objects.

- Study of QNX Micro Kernal Operating System and QNX Momentics IDE
- Installation of QNX Momentics IDE 8.0, and Installation of QNX Operating System in Raspberry Pi 4
- Write a program for “Hello World” and using QNX Momentics IDE, execute on x86 and ARMx64 architecture.

UNIT II PROCESSES, THREADS, AND SYNCHRONIZATION 6 hours
Process management: creation, termination, and memory protection, Thread management: creation, termination, and synchronization, Synchronization techniques: mutexes, semaphores, and condition variables, Hands-on exercises: process/thread creation and synchronization.

- Write a program that creates multiple child processes using fork() and print different messages in parent and child process. Terminate the parent after 5 seconds and print the pid from the child.
- Implement a multi-threaded application using POSIX threads (pthread_create). Each thread should process a different part of an array and the main thread should wait for all threads to complete using pthread_join.

- Write a program to create a process with 4 threads that update the portin of array of size 1000 bytes by updating 250 bytes each. Make the main thread to join on the 4 threads and print the completion. Use mutex to prevent data corruption while each thread is updating the array.
- Implementing a thread-safe bounded buffer (also known as a circular queue) that is shared between multiple producer threads and multiple consumer threads. The buffer has a fixed size (N slots). Producers add items to the buffer, and consumers remove items from the buffer.

UNIT III INTER-PROCESS COMMUNICATION (IPC) 6 hours

Overview of IPC methods in QNX: message passing, pulses, and shared memory, Comparing IPC methods: advantages and disadvantages, Practical implementation of IPC in QNX, Hands-on exercises: message passing and shared memory.

- Write a program to create a basic Client and Server model to send / receive data. If *MsgReply()* was removed from server what happens?, if the server's *MsgReceive()* returns a failure, should the program exit?
- Write a program to create a pulse and send from a process to another by keeping *pid* and *cid* in the pulse. Verify the exchange of messages and delivery of pulses
- Write a program to create a pulse and send from a process to another by use name_attach(), and name_attach() creates a channel with several channel flags turned on, your server should expect to receive kernel pulses on a disconnect pulse, release the scoid by calling ConnectDetach()
- Write a program to creates shared memory objects and handles, reads and updates the shared memory, finally release message and disconnect pulse with server client model.

UNIT IV HARDWARE PROGRAMMING AND TIMING 6 hours

Hardware access methods: IO-mapped and memory-mapped IO, Interrupt handling and DMA-safe memory allocation, Timing architecture: periodic timing, one-shot timing, and timeouts, Hands-on exercises: interrupt handling and timing mechanisms.

- Write a Program to Simple standard interrupt handling - Interrupt 1, works only on VM ware virtual machine
- Write a program to a timer function, it will wake up 5 seconds from the time it runs and then every 1500 milliseconds after that it will wake up by receiving a pulse.
- Implement two processes that communicate via shared memory using shm_open and mmap. One process writes data, and the other reads and displays it.

UNIT V BUILDING AND CONFIGURING QNX BOOT/OS IMAGES 6 hours

Overview of QNX boot/OS image structure, Components of a boot image: startup code, kernel, drivers, and scripts, Building and loading boot images onto target hardware, Introduction to resource managers and their implementation.

- Create a new virtual machine image with a larger boot disk from Momentics IDE and VSCode and build the image directory on the target (x86_64 virtualized)
- Create the Boot image using QNX BSP

Course Outcomes:

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

CO1: Describe the QNX OS architecture and its microkernel-based design and Develop and debug QNX-based applications using appropriate tools.

CO2: Apply process/thread management and synchronization techniques in QNX.

CO3: Implement inter-process communication methods for real-time systems.

CO4: Configure and build QNX boot/OS images for specific hardware platforms.

CO5: Develop strong knowledge on the POSIX standards that help in System Application Development

Text Book(s)

1. Amos, Brian. "*Hands-On RTOS with Microcontrollers: Building real-time embedded systems using FreeRTOS, STM32 MCUs, and SEGGER debug tools"*. Packt Publishing Ltd, 2020.
2. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, "*Operating System Concepts"*, 9th Edition, Wiley, 2018
3. Barr, Michael, and Anthony Massa. *Programming embedded systems: with C and GNU development tools*. " O'Reilly Media, Inc.", 2006.

Reference Book(s)

1. Kerrisk, Michael. *The Linux programming interface: a Linux and UNIX system programming handbook*. No Starch Press, 2010.
2. Tanenbaum, A. S., and A. S. Woodhull. "The minix book–operating systems." (2006).
3. Online Resources:
   (a) QNX Neutrino RTOS User's Guide, QNX Software Systems.
   (b) https://blackberry .qnx.com/en/products/qnx-everywhere
   (c) https://gitlab.com/qnx/projects

Mode of Evaluation: Assignments, Mid Term Tests and End Semester Examination

Discipline Electives

Discipline Elective – I

25MBVESDC01 COMMUNICATION BUSES AND INTERFACES

L	T	P	C
3	0	0	3

Pre-requisite Computer Communication Networks

Course Objectives:

This course enables students to

1. Understand the concepts of different types of serial buses.
2. Learn about CAN Architecture and its standards,
3. Familiarize with PCI Express Technology and its protocols
4. Analyse USB architecture, its transfer types, and device drivers.
5. Learn about data streaming using serial communication protocols

UNIT I SERIAL BUSSES **9 hours**

Cables, Serial busses, serial versus parallel, Data and Control Signal- data frame, data rate, features, Limitations and applications of RS232, RS485, I2C, SPI

UNIT II CAN ARCHITECTURE **9 hours**

ISO 11898-2, ISO 11898-3, Data Transmission- ID allocation, Bit timing, Layers- Application layers, Object layer, Transfer layer, Physical layer, Frame formats- Data frame, Remote frame, Error frame, Over load frame, Ack slot, Inter frame spacing, Bit spacing, Applications.

UNIT III PCIe **9 hours**

Revision, Configuration space- configuration mechanism, Standardized registers, Bus enumeration, Hardware and Software implementation, Hardware protocols, Applications.

UNIT IV USB **9 hours**

Transfer Types- Control transfers, Bulk transfer, Interrupt transfer, Isochronous transfer. Enumeration- Device detection, Default state, Addressed state, Configured state, enumeration sequencing. Descriptor types and contents- Device descriptor, configuration descriptor, Interface descriptor, Endpoint descriptor, String descriptor. Device driver.

UNIT V DATA STREAMING SERIAL COMMUNICATION PROTOCOL **9 hours**

Serial Front Panel Data Port(SFPDP) configurations, Flow control, serial FPDP transmission frames, fiber frames and copper cable

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the concepts of different types of serial buses.

CO2: Learn about CAN Architecture and its standards.

CO3: Familiarize with PCI Express Technology and its protocols.

CO4: Analyse USB architecture, its transfer types, and device drivers.

CO5: Learn about data streaming using serial communication protocols.

Text Book(s)

1. A Comprehensive Guide to controller Area Network – Wilfried Voss, Copperhill Media Corporation, 2nd Ed., 2005.
2. Serial Port Complete-COM Ports, USB Virtual Com Portsand Ports for Embedded Systems-Jan Axelson, Lakeview Research, 2nd Ed.,

Reference Books

1. USB Complete – Jan Axelson, Penram Publications.
2. PCI Express Technology – Mike Jackson, Ravi Budruk, Mindshare Press.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – I

25MBVESDC02 DATA ACQUISITION SYSTEM DESIGN

L	T	P	C
3	0	0	3

Pre-requisite Computer Networks

Course Objectives:

This course enables students to

1. To understand the different types of communication interface buses.
2. To familiarize different methods of ADC's and DAC characteristics, specifications
3. To study the software tools to develop the code and implementation for data acquisition system

UNIT I **9 hours**

Fundamentals of Data Acquisition Systems, Sensors and Transducers, Signal conditioning Introduction, Types of signal conditioning, Classes of signal conditioning, DAQ Hardware, DAQ Software, Communications Cabling, Parameters of a DAQ System.

UNIT II **9 hours**

Data acquisition system configuration, Computer plug in I/O, Distributed I/O, Stand-alone or distributed loggers/controllers- Introduction, Methods of operation, Stand-alone logger/controller hardware, firmware & software design, Communications hardware interface, Host software, Considerations, internal systems, USB overall structure, PCMCIA card. Application Layer Protocols.

UNIT III **9 hours**

Data Acquisition Systems: Hardware-Introduction, Plug-in DAQ Systems, Converters A/D, Converters D/A, Amplifier, Multiplexer/De-multiplexer, Power Management, Timing System, Filtering, Memory Board, Bus Interface

UNIT IV **9 hours**

Communication Bus-Bus and FireWire, Serial Communications, Wireless, Ethernet and Bluetooth, GSM for Data Acquisition System, PCI and PCI Express, Standard VME. High speed data acquisition and interface, SPI read/write protocol, RTC interfacing and programming.

UNIT V **9 hours**

Design of Data Acquisition System: Introduction to the Design, Functional Design of high-Speed Computers-Based DAS, Portable DAS, Design Guidelines for High-Performance Multichannel. Software for Data Acquisition Systems, Introduction to LabVIEW, Android for DAQ, Design of Firmware, Example of Implementation of a Software.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the fundamental concepts and components of data acquisition systems.

CO2: Apply knowledge of analog-to-digital (ADC) and digital-to-analog (DAC) converters in data acquisition.

CO3: Design and implement hardware for data acquisition systems.

CO4: Analyze the performance of data acquisition systems, including resolution, accuracy, and speed.

CO5: Develop and implement software solutions for data collection and processing.

Text Book(s)

1. Maurizio Di Paolo Emilio "Data acquisition systems-from fundamentals to applied design" springer, 2013.
2. John Park and Steve Mackay "Practical Data acquisition for instrumentation and control systems" Elsevier, 2003.

Reference Books

1. Robert H King, "Introduction to Data Acquisition with LabVIEW", 2nd edition, 2012, McGraw

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – I

25MBVESDC03 FPGA ARCHITECTURES AND APPLICATIONS

L	T	P	C
3	0	0	3

Pre-requisite Digital System Design, ASIC Design

Course Objectives:

This course enables students to

1. To acquire knowledge about various architectures and device technologies of PLD's.
2. To comprehend FPGA Architectures.
3. To analyze System level Design and their application for Combinational and Sequential Circuits.
4. To familiarize with Anti-Fuse Programmed FPGAs.
5. To apply knowledge of this subject for various design applications.

UNIT I **9 hours**

Introduction to Programmable Logic Devices: Introduction, Simple Programmable Logic Devices – Read Only Memories, Programmable Logic Arrays, Programmable Array Logic, Programmable Logic Devices/Generic Array Logic; Complex Programmable Logic Devices–Architecture of Xilinx Cool Runner XCR3064XL CPLD, CPLD Implementation of a Parallel Adder with Accumulation.

UNIT II **9 hours**

Field Programmable Gate Arrays: Organization of FPGAs, FPGA Programming Technologies, Programmable Logic Block Architectures, Programmable Interconnects, and Programmable I/O blocks in FPGAs, Dedicated Specialized Components of FPGAs, and Applications of FPGAs.

UNIT III **9 hours**

SRAM Programmable FPGAs: Introduction, Programming Technology, Device Architecture, the Xilinx XC2000, XC3000 and XC4000 Architectures.

UNIT IV **9 hours**

Anti-Fuse Programmed FPGAs: Introduction, Programming Technology, Device Architecture, The Actel ACT1, ACT2 and ACT3 Architectures.

UNIT V **9 hours**

Design Applications: General Design Issues, Counter Examples, A Fast Video Controller, A Position Tracker for a Robot Manipulator, A Fast DMA Controller, Designing Counters with ACT devices, Designing Adders and Accumulators with the ACT Architecture.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Acquire knowledge about various architectures and device technologies of PLD's.

CO2: Comprehend FPGA Architectures.

CO3: Analyze System level Design and their application for Combinational and Sequential Circuits.

CO4: Familiarize with Anti-Fuse Programmed FPGAs.

CO5: Apply knowledge of this subject for various design applications.

Text Book(s)

1. Field Programmable Gate Array Technology - Stephen M. Trimberger, Springer International Edition.
2. Digital Systems Design - Charles H. Roth Jr, Lizy Kurian John, Cengage Learning.

Reference Books

1. Field Programmable Gate Arrays-John V.Oldfield, Richard C.Dorf, Wiley India.
2. Digital Design Using Field Programmable Gate Arrays - Pak K. Chan/SamihaMourad, Pearson Low Price Edition.
3. Digital Systems Design with FPGAs and CPLDs-Ian Grout, Elsevier, Newnes.
4. FPGA based System Design-Wayne Wolf, Prentice Hall Modern Semiconductor Design Series.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – II

25MBVESDC04 LOW POWER VLSI DESIGN

L	T	P	C
3	0	0	3

Pre-requisite VLSI

Course Objectives:

This course enables students to

1. To understand the concepts of velocity saturation, Impact Ionization and Hot Electron Effect.
2. To implement Low power design approaches for system level and circuit level measures.
3. To design low power adders for efficient design of systems.
4. To design low power multipliers for efficient design of systems.
5. To design memory circuits with low power dissipation.

UNIT I INTRODUCTION 9 hours

Need for Low Power Circuit Design, Sources of Power Dissipation – Static and Dynamic Power Dissipation, Short Circuit Power Dissipation, Glitching Power Dissipation, Short Channel Effects – Drain Induced Barrier Lowering and Punch Through, Surface Scattering, Velocity Saturation, Impact Ionization, Hot Electron Effect.

UNIT II Low-Power Design Approaches 9 hours

Low-Power Design through Voltage Scaling – VTCMOS circuits, MTCMOS circuits, Architectural Level Approach –Pipelining and Parallel Processing Approaches. Switched Capacitance Minimization Approaches: System Level Measures, Circuit Level Measures, Mask level Measures.

UNIT III Low-Voltage Low-Power Adders 9 hours

Introduction, Standard Adder Cells, CMOS Adder's Architectures – Ripple Carry Adders, Carry Look Ahead Adders, Carry Select Adders, Carry Save Adders, Low-Voltage Low-Power Design Techniques – Trends of Technology and Power Supply Voltage, Low-Voltage Low-Power Logic Styles.

UNIT IV Low-Voltage Low-Power Multipliers 9 hours

Introduction, Overview of Multiplication, Types of Multiplier Architectures, Braun Multiplier, Baugh Wooley Multiplier, Booth Multiplier, Introduction to Wallace Tree Multiplier.

UNIT V Low-Voltage Low-Power Memories 9 hours

Basics of ROM, Low-Power ROM Technology, Future Trend and Development of ROMs, Basics of SRAM, Memory Cell, Precharge and Equalization Circuit, Low-Power SRAM Technologies, Basics of DRAM, Self-Refresh Circuit, Future Trend and Development of DRAM.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand concepts of velocity saturation, Impact Ionization and Hot Electron Effect.
CO2: Implement Low power design approaches for system level and circuit level measures.
CO3: Design low power adders for efficient design of systems.
CO4: Design low power multipliers for efficient design of systems.
CO5: Design memory circuits with low power dissipation.

Text Book(s)

1. CMOS Digital Integrated Circuits – Analysis and Design – Sung-Mo Kang, Yusuf Leblebici, TMH, 2011.
2. Low-Voltage, Low-Power VLSI Subsystems – Kiat-Seng Yeo, Kaushik Roy, TMH Professional Engineering.

Reference Books

1. Introduction to VLSI Systems: A Logic, Circuit and System Perspective – Ming-BO Lin, CRC Press, 2011.
2. Low Power CMOS Design – Anantha Chandrakasan, IEEE Press/Wiley International, 1998.
3. Low Power CMOS VLSI Circuit Design – Kaushik Roy, Sharat C. Prasad, John Wiley & Sons, 2000

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – II

25MBVESDC05 NANO-MATERIALS AND NANOTECHNOLOGY

L	T	P	C
3	0	0	3

Pre-requisite Material science, Basic chemistry

Course Objectives:

This course enables students to

1. To understand the basic idea behind the design and fabrication of nano scale systems.
2. To understand and formulate new engineering solutions for current problems and technologies for future applications.
3. To acquire knowledge on the operation of fabrication and characterization devices to achieve precisely designed systems.

UNIT I **9 hours**

Introduction of nano materials and nanotechnologies, Features of nanostructures, Applications of nano materials and technologies. Nano dimensional Materials 0D, 1D, 2D structures – Size Effects – Fraction of Surface Atoms –Specific Surface Energy and Surface Stress – Effect on the Lattice Parameter – Phonon Density of States – the General Methods available for the Synthesis of Nanostructures – precipitate – reactive– hydrothermal/solvo thermal methods – suitability of such methods for scaling – potential Uses.

UNIT II **9 hours**

Fundamentals of nanomaterials, Classification, Zero-dimensional nanomaterials, One-dimensional nanomaterials, Two-dimensional nano materials, three dimensional nanomaterials. Low Dimensional Nanomaterials and its Applications, Synthesis, Properties and applications of Low Dimensional Carbon-Related Nanomaterials.

UNIT III **9 hours**

Micro- and Nanolithography Techniques, Emerging Applications, Introduction to Micro electromechanical Systems (MEMS), Advantages and Challenges of MEMS, Fabrication Technologies, Surface Micromachining, Bulk Micromachining, Molding. Introduction to Nano Phonics.

UNIT IV **9 hours**

Introduction, Synthesis of CNTs - Arc-discharge, Laser-ablation, Catalytic growth, Growth mechanisms of CNT‟s - Multi-walled nanotubes, Single-walled nano tubes Optical properties of CNT‟s, Electrical transport in perfect nanotubes, Applications as case studies. Synthesis and Applications of CNTs.

UNIT V **9 hours**

Ferroelectric materials, coating, molecular electronics and Nano electronics, biological and environmental, membrane based application, polymer based application.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the fundamental principles and concepts of nanotechnology and nanomaterials.
CO2: Analyze the synthesis methods of nanomaterials, including top-down and bottom-up approaches.
CO3: Evaluate the physical, chemical, and mechanical properties of nanomaterials.
CO4: Gather detailed knowledge of the operation of fabrication and characterization devices to achieve precisely designed systems.
CO5: Understand the applications of nanomaterials in various fields, including electronics, energy, medicine, and environmental science.

Text Book(s)

1. Kenneth J. Klabunde and Ryan M. Richards, "Nanoscale Materials in Chemistry", 2ndedition, John Wiley and Sons, 2009.
2. I Gusev and A Rempel, "Nanocrystalline Materials", Cambridge International Science Publishing, 1stIndian edition by Viva Books Pvt. Ltd. 2008.
3. B. S. Murty, P. Shankar, Baldev Raj, B. B. Rath, James Murday, "Nanoscience and Nanotechnology", Tata McGraw Hill Education 2012.

Reference Books

1. Digital Integrated Circuit Design – Ken Martin, Oxford University Press, 2011.
2. Digital Integrated Circuits - A Design Perspective, Jan M.Rabaey, Anant Chandrakasan, Borvivoje Nikolic, 2nd Edition, PHI.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – II

25MBVESDC06 NETWORK SECURITY AND CRYPTOGRAPHY

L	T	P	C
3	0	0	3

Pre-requisite Computer Networks

Course Objectives:

This course enables students to

1. To identify and utilize different forms of cryptography techniques.
2. To incorporate authentication and security in the network applications.
3. To distinguish among different types of threats to the system and handle the same.

UNIT I SECURITY **9 hours**

Need of security, security services, Attacks, OSI Security Architecture, one-time passwords, Model for Network security, Classical Encryption Techniques like substitution ciphers, Transposition ciphers, Cryptanalysis of Classical Encryption Techniques

UNIT II NUMBER THEORY **9 hours**

Introduction, Fermat's and Euler's Theorem, The Chinese Remainder Theorem, Euclidean Algorithm, Extended Euclidean Algorithm, and Modular Arithmetic

UNIT III PRIVATE- KEY(SYMMETRIC) CRYTOGRAPHY: **9 hours**

Block Ciphers, Stream Ciphers, RC4 Stream cipher, Data Encryption Standard (DES), Advanced Encryption Standard (AES), Triple DES, RC5, IDEA, Linear and Differential Cryptanalysis.

UNIT IV PUBLIC- KEY(ASYMMETRIC) CRYTOGRAPHY: **9 hours**

RSA, Key Distribution and Management, Diffie-Hellman Key Exchange, Elliptic Curve Cryptography, Message Authentication Code, hash functions, message digest algorithms: MD4 MD5, Secure Hash algorithm, RIPEMD-160, HMAC

UNIT V AUTHENTICATION AND SYSTEM SECURITY: **9 hours**

IP and Web Security Digital Signatures, Digital Signature Standards, Authentication Protocols, Kerberos,

IP security Architecture, Encapsulating Security Payload, Key Management, Web Security Considerations, Secure Socket Layer, Secure Electronic Transaction Intruders, Intrusion Detection, Password Management, Worms, viruses, Trojans, Virus Countermeasures, Firewalls, Trusted Systems.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the fundamental principles of network security and cryptography.
CO2: Analyze various cryptographic algorithms, including symmetric and asymmetric encryption.
CO3: Apply cryptographic techniques for securing data transmission over networks.
CO4: Understand and implement security protocols for network communication.
CO5: Analyze potential vulnerabilities and threats in computer networks.

Text Book(s)

1. William Stallings, "Cryptography and Network Security, Principles and Practices", Pearson Education, 3rd Edition.
2. Charlie Kaufman, Radia Perlman and Mike Speciner, "Network Security, Private Communication in a Public World", Prentice Hall, 2ND Edition. .

Reference Books

1. Christopher M. King, Ertem Osmanoglu, Curtis Dalton, "Security Architecture, Design Deployment and Operations", RSA Pres,
2. Stephen Northcutt, Leny Zeltser, Scott Winters, Karen Kent, and Ronald W. Ritchey, "Inside Network Perimeter Security", Pearson Education, 2 nd Edition
3 Richard Bejtlich, "The Practice of Network Security Monitoring: Understanding Incident Detection and Response", William Pollock Publisher, 2013

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – III

25MBVESDC07 PATTERN RECOGNITION AND MACHINE LEARNING

L	T	P	C
3	0	0	3

Pre-requisite Statistics and Probability Theory

Course Objectives:

This course enables students to

1. To understand the mathematical formulation of patterns.
2. To comprehend the principles and mathematical foundations of linear models.
3. To explore various kernel-based algorithms and their applications.
4. To study different types of graphical models and Markov random fields.
5. To understand the theory behind mixture models the EM algorithms

UNIT I **9 hours**

Introduction to Pattern recognition: Mathematical Formulation and Basic Functional Equation, Reduction of Dimensionality, Experiments in Pattern Classification, Backward Procedure for Both Feature Ordering- and Pattern Classification, Suboptimal Sequential Pattern Recognition, Nonparametric Design of Sequential Pattern Classifiers, Analysis of Optimal Performance and a Multiclass Generalization

UNIT II **9 hours**

Linear Models: Linear Basis Function Models -Maximum likelihood and least squares, Geometry of least squares , Sequential learning, Regularized least squares, Multiple outputs , The Bias-Variance Decomposition, Bayesian Linear Regression -Parameter distribution, Predictive, Equivalent, Bayesian Model Comparison, Probabilistic Generative Models-Continuous inputs , Maximum likelihood solution, Discrete features, Exponential family, Probabilistic Discriminative Models - Fixed basis functions, Logistic regression, Iterative reweighted least squares, Multiclass logistic regression, Probit regression, Canonical link functions

UNIT III **9 hours**

Kernel Methods: Constructing Kernels, Radial Basis Function Networks - Nadaraya-Watson model, Gaussian Processes -Linear regression revisited, Gaussian processes for regression, Learning the hyper parameters, Automatic relevance determination, Gaussian processes for classification, Laplace approximation, Connection to neural networks, Sparse Kernel Machines- Maximum Margin Classifiers, Overlapping class distributions, Relation to logistic regression, Multiclass SVMs, SVMs for regression, Computational learning theory, Relevance Vector Machines- RVM for regression, Analysis of sparsity, RVM for classification

UNIT IV **9 hours**

Graphical Models: Bayesian Networks, Example: Polynomial regression, Generative models, Discrete variables, Linear-Gaussian models, Conditional Independence- Three example graphs,
Markov Random Fields -Conditional independence properties, Factorization properties, Illustration: Image de-noising, Relation to directed graphs, Inference in Graphical Models- Inference on a chain, Trees, Factor graphs, The sum-product algorithm, The max-sum algorithm, Exact inference in general graphs, Loopy belief propagation, Learning the graph structure.

UNIT V **9 hours**

Mixture Models and EM algorithm: Image segmentation and compression - K-means Clustering, Mixtures of Gaussians-Maximum likelihood, EM for Gaussian mixtures, An Alternative View of EM Gaussian mixtures revisited, Relation to K-means, Mixtures of Bernoulli distributions, EM for Bayesian linear regression, The EM Algorithm in General, Combining Models- Tree-based Models, Conditional Mixture Models- Mixtures of linear regression models, Mixtures of logistic models, Mixtures of experts.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Explain the basic concepts and importance of pattern recognition in machine learning.
CO2: Evaluate and interpret the performance of linear models on different datasets.
CO3: Apply kernel-based algorithms to solve non-linear pattern recognition problems
CO4: Construct and interpret different types of graphical models for various applications.
CO5: Apply mixture models and the EM algorithm to practical problems and datasets.

Text Book(s)

1. Sequential methods in Pattern Recognition and Machine Learning-K. S. Fu, Academic Press, volume no. 52.
2. Pattern Recognition and Machine Learning- C. Bishop-Springer, 2006.

Reference Books

1. Pattern Classification- Richard o. Duda, Peter E. hart, David G. Stork, John Wiley& Sons, 2nd Ed., 2001.
2. The elements of Statistical Learning- Trevor Hastie, Robert Tibshirani, Jerome H. Friedman, Springer, 2nd Ed., 2009

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – III

25MBVESDC08 PROGRAMMING LANGUAGES FOR EMBEDDED SOFTWARE

L	T	P	C
3	0	0	3

Pre-requisite Microcontrollers and Microprocessors:

Course Objectives:

This course enables students to

1. To introduce students to various programming languages like C, C++, Java script, PERL, etc.
2. To distinguish between Procedural and OOP language, Introduce features of OOPs etc.
3. To demonstrate the development of some typical applications using different Programming languages.

UNIT I **9 hours**

Embedded 'C' Programming: Bitwise operations, Dynamic memory allocation, OS services, linked stack and queue, Sparse matrices, Binary tree, Interrupt handling in C, Code optimization issues, Writing LCD drives, LED drivers, Drivers for serial port communication, Embedded Software Development Cycle and Methods (Waterfall, Agile).

UNIT II **9 hours**

Object Oriented Programming: Introduction to procedural, modular, object oriented and generic programming techniques, Limitations of procedural programming, objects, classes, data members, methods, data encapsulation, data
Abstraction and information hiding, inheritance, polymorphism.

UNIT III **9 hours**

CPP Programming: 'cin', 'cout', formatting and I/O manipulators, new and delete operators, Defining a class, data members and methods, 'this' pointer, constructors, destructors, friend function,
dynamic memory allocation.

UNIT IV **9 hours**

Overloading and Inheritance: Need of operator overloading, overloading the assignment, overloading using friends, type conversions, single inheritance, base and derived classes, friend classes, types of inheritance, hybrid inheritance,
multiple inheritance.
Templates: Function template and class template, member function templates and template arguments

UNIT V **9 hours**

Exception Handling: Syntax for exception handling code: try-catch-throw, Multiple Exceptions.
Scripting Languages: Overview of Scripting Languages – PERL, CGI, VB Script, Java Script. PERL: Operators, Statements Pattern Matching etc. Data Structures, Modules, Objects, Tied Variables, Inter process Communication Threads, Compilation & Line Interfacing.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Introduce students to various programming languages like C, C++, Java script, PERL, etc.

CO2: Distinguish between Procedural and OOP language, Introduce features of OOPs etc.

CO3: Demonstrate the development of some typical applications using different Programming languages.

CO4: Analyze the trade-offs between various embedded programming languages and platforms.

CO5: Integrate hardware and software to design complete embedded system projects..

Text Book(s)

1. Michael J. Pont, "Embedded C", Pearson Education, 2nd Edition, 2008.
2. Robert Sedgewick, "Algorithms in C++", Addison Wesley Publishing Company, 1999.

Reference Books

1. Randal L. Schwartz, "Learning Perl", O'Reilly Publications, 6th Edition 2011.
2. Michael Berman, "Data structures via C++", Oxford University Press, 2002

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – III

25MBVESDC09 RF IC DESIGN

L	T	P	C
3	0	0	3

Pre-requisite Microwave, Electromagnetics

Course Objectives:

This course enables students to

1. To introduce students the concept of tuned circuit, matching network, reflection coefficients, transmission lines and MOS high frequency behavior etc.
2. To demonstrate design of High Frequency Amplifiers.
3. To introduce various types of Power Amplifiers and PLLs

UNIT I **9 hours**

RF Tuned Circuits: RF systems – Basic architectures, Maximum Power Transfer, Passive RLC Networks, Parallel RLC tank, Q, Series RLC networks, matching, Pi match, T match, Passive components in IC: Resistors, capacitors, Inductors, Transceiver Architectures.

UNIT II **9 hours**

Nonlinearity and Time Variance of system, sensitivity and dynamic range, Review of MOS Device Physics, MOS device review, Distributed Systems, Transmission lines, reflection coefficient, the wave equation Lossy transmission lines Smith charts – plotting gamma, Noise in FET: Thermal noise, flicker noise review

UNIT III **9 hours**

High Frequency Amplifier Design: Bandwidth estimation using open-circuit time constants, Bandwidth estimation using short-circuit time constants, Rise-time, delay and bandwidth, Zeros to enhance bandwidth, Shunt- series amplifiers, tuned amplifiers Cascaded amplifiers, Noise figure, Intrinsic MOS noise parameters, LNA Design, Power match versus noise match

UNIT IV **9 hours**

RF Power Amplifiers: Multiplier based mixers, Subsampling mixers & Mixer Design, RF Power Large signal performance Amplifiers, Class A, AB, B, C amplifiers, Class D, E, F amplifiers, RF Power amplifier design issues. LC Oscillators, Voltage Controlled Oscillators, Ring oscillators, Delay Cells, tuning range in ring oscillators, Tuning in LC oscillators, Tuning sensitivity, Phase Noise in oscillators, sources of phase noise

UNIT V **9 hours**

PLL: Voltage controlled oscillators, Resonators, Negative resistance oscillators, Phase locked loops, Linearized PLL models, Phase detectors, charge pumps, Loop filters, PLL design examples, Frequency synthesis and oscillator Frequency division, integer-N synthesis, Phase noise

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand relation between automation algorithms and constraints posed by VLSI technology.

CO2: Introduce students the concept of tuned circuit, matching network, reflection coefficients, transmission lines and MOS high frequency behavior etc.

CO3: Demonstrate design of High Frequency Amplifiers.

CO4: Introduce various types of Power Amplifiers.

CO5: Apply frequency division techniques in oscillators and PLLs, and understand their role in frequency synthesis.

Text Book(s)

1. Thomas H. Lee, "The Design of CMOS Radio-Frequency Integrated Circuits", Cambridge University Press, 2004.
2. Behzad Razavi, "RF Microelectronics", Prentice Hall, 1997.

Reference Books

1. Abidi, P.R. Gray, and R.G. Meyer, eds., "Integrated Circuits for Wireless Communications", New York: IEEE Press, 1999.
2. R. Ludwig and P. Bretchko, "RF Circuit Design, Theory and Applications", Pearson, 2000.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – IV

25MBVESDC10 SoC ARCHITECTURE

L	T	P	C
3	0	0	3

Pre-requisite Computer architecture, Digital logic design

Course Objectives:

This course enables students to

1. To understand the basics related to SoC architecture and different approaches related to SoC Design.
2. To select an appropriate robust processor for SoC Design
3. To select an appropriate memory for SoC Design.
4. To realize real time case studies

UNIT I **9 hours**

Introduction to the System Approach: System Architecture, Components of the system, Hardware & Software, Processor Architectures, Memory &Addressing. System level interconnection, An approach for SOC Design, System Architecture and Complexity.

UNIT II **9 hours**

Processors: Introduction, Processor Selection for SOC, Basic concepts in Processor Architecture, Basic concepts in Processor Microarchitecture, Basic elements in Instruction handling.
Buffers: minimizing Pipeline Delays, Branches, More Robust Processors, Vector Processors and Vector Instruction extensions, VLIW Processors, Superscalar Processors

UNIT III **9 hours**

Memory Design for SOC: Overview: SOC external memory, SOC Internal Memory, Size, Scratchpads and Cache memory, Cache Organization, Cache data, Write Policies, Strategies for line replacement at miss time, Other Types of Cache, Split – I, and D – Caches, Multilevel Caches, SOC Memory System, Models of Simple Processor – memory interaction.

UNIT IV **9 hours**

Interconnect, Customization and Configurability: Interconnect Architectures, Bus: Basic Architectures, SOC Standard Buses, Analytic Bus Models, Using the Bus model, Effects of Bus transactions and contention time.
SOC Customization: An overview, Customizing Instruction Processor, Reconfigurable Technologies, Mapping design onto Reconfigurable devices, Instance- Specific design, Customizable Soft Processor, Reconfiguration - overhead analysis and trade-off analysis on Reconfigurable Parallelism.

UNIT V **9 hours**

Application Studies / Case Studies: SOC Design approach; AES-algorithms, Design and evaluation; Image compression–JPEG compression.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the basics related to SoC architecture and different approaches related to SoC Design.
CO2: Select an appropriated robust processor for SoC Design
CO3: Select an appropriate memory for SoC Design.
CO4: Design SoC
CO5: Realize real time case studies

Text Book(s)

1. Computer System Design System-on-Chip - Michael J. Flynn and Wayne Luk, Wiely India Pvt. Ltd.
2. ARM System on Chip Architecture – Steve Furber, 2ndEdition, 2000, Addison Wesley Professional.

Reference Books

1. Design of System on a Chip: Devices and Components – Ricardo Reis, 1st Ed., 2004, Springer.
2. Co-Verification of Hardware and Software for ARM System on Chip Design (Embedded Technology) – Jason Andrews – Newnes, BK and CDROM.
3. System on Chip Verification – Methodologies and Techniques –Prakash Rashinkar, Peter Paterson and Leena Singh L, 2001, Kluwer Academic Publishers

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – IV

25MBVESDC11 SYSTEM DESIGN WITH EMBEDDED LINUX

L	T	P	C
3	0	0	3

Pre-requisite Embedded Systems, Computer architecture

Course Objectives:

This course enables students to

1. To understand the importance of Embedded Linux in system design
2. To analyze the architecture of embedded Linux in detail
3. To explain the Linux BSP for a hardware platform
4. To develop and Debug the drivers in Embedded Linux

UNIT I **9 hours**

Introduction: Need of Embedded Linux, Embedded Linux versus Desktop Linux, Embedded Linux Distributions Embedded Linux Architecture, Kernel Architecture: Hardware Abstraction Layer (HAL), Memory Manager, Scheduler, File System, IO Subsystem, Networking Subsystems, IPC; User Space, Linux Start-Up Sequence.

UNIT II **9 hours**

Board Support Package: Inserting BSP in Kernel Build Procedure, the Boot Loader Interface, Memory Map, Interrupt Management, the PCI Subsystem, Timers, UART, and Power Management. Embedded Storage: Flash Map, Memory Technology Device, MTD Architecture, Embedded File Systems.

UNIT III **9 hours**

Embedded Drivers: Linux Serial Driver, Ethernet Driver, and I2C Subsystem on Linux, USB Gadgets, Watchdog Timer, and Kernel Modules. Porting Applications: Architectural Comparison, Application Porting Roadmap.

UNIT IV **9 hours**

Real-Time Linux: Linux and Real-Time: Building and Debugging: Building the Kernel, Building the Root File System, Integrated Development Environment, Elementary Concepts of Debugging. Embedded Graphics: Graphics System, Introduction to Display Hardware.

UNIT V **9 hours**

uClinux: Linux on MMU - Less Systems, Program Load and Execution, Memory Management, File/ Memory Mapping.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Understand the importance of Embedded Linux in system design
CO2: Explain the Linux BSP for a hardware platform
CO3: Implement device drivers for embedded hardware peripherals.
CO4: Develop and deploy embedded applications on Linux-based platforms.
CO5: Explore features of uClinux-based embedded systems.

Text Book(s)

1. Derek Molloy, "Exploring Beagle Bone: Tools and Techniques for Building with Embedded Linux", Wiley, 1st Edition, 2014.
2. Christopher Hallinan, "Embedded Linux Primer: A Practical Real-World Approach", Prentice Hall,2nd Edition, 2010.

Reference Books

1. P Raghvan, Amol Lad, Sriram Neelakandan, "Embedded Linux System Design and Development",Auerbach Publications, 2005.
2. Karim Yaghmour, "Building Linux Systems", O'Reilly & Associates,2008.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.

Discipline Elective – IV

25MBVESDC12 HIGH SPEED VLSI DESIGN

L	T	P	C
3	0	0	3

Pre-requisite Digital Logic Design, CMOS circuits, VLSI Design fundamentals

Course Objectives:

This course enables students to

1. To introduce the need and challenges of designing high-speed digital integrated circuits.
2. To provide knowledge of interconnect effects, delay modeling, power optimization, and clocking techniques.
3. To analyze circuit design techniques for achieving high performance in CMOS-based VLSI systems.
4. To develop skills for modeling, simulation, and verification of high-speed VLSI circuits using EDA tools.
5. To enable students to evaluate trade-offs between speed, power, and area in high-performance system-on-chip (SoC) design.

UNIT I **9 hours**

Need for high-speed VLSI circuits, emphasizing performance metrics such as delay, power, throughput, and area. It covers the impact of technology scaling, short-channel effects, and device limitations on speed. The role of CAD tools in modeling, simulation, and verification of high-speed digital designs is also discussed.

UNIT II **9 hours**

Different sources of delay in digital circuits, focusing on RC delay modeling, Elmore delay, and logical effort. Interconnect parasitics and their influence on circuit performance are analyzed, along with wire modeling and scaling effects. Signal integrity issues such as crosstalk, reflection, ringing, and ground bounce are introduced with methods to mitigate them.

UNIT III **9 hours**

Design techniques for achieving high speed in CMOS circuits. It includes static and dynamic CMOS design, domino logic, pass-transistor logic, and current-mode logic. Pipelining, retiming, and parallelism are introduced as techniques for enhancing system performance. Low-power design strategies and trade-offs between speed, power, and area are also highlighted.

UNIT IV **9 hours**

Clock distribution and synchronization challenges in high-speed designs. Topics include clock distribution networks, clock skew, PLLs, DLLs, and techniques to address on-chip variation. The concepts of metastability and synchronization methods are discussed, along with the design of high-speed latches and flip-flops for reliable operation.

UNIT V **9 hours**

Principles of high-speed design to practical systems. It includes high-speed I/O circuits, DDR memories, serializer/deserializer (SerDes) architectures, and on-chip interconnect design. Case studies on processor and FPGA-based high-speed designs are explored, along with current industry trends in high-performance VLSI systems.

Course Outcomes:

Upon successful completion of the course, students will be able to

CO1: Explain the principles, challenges, and performance requirements of high-speed VLSI systems.

CO2: Analyze interconnect delay, crosstalk, and signal integrity issues in deep-submicron technologies.

CO3: Design and optimize CMOS circuits and interconnects for high-speed performance.

CO4: Apply advanced clocking, pipelining, and power optimization techniques in VLSI design.

CO5: Simulate, verify, and evaluate high-speed VLSI circuits using industry-standard tools, demonstrating trade-offs in speed, power, and area.

Text Book(s)

1. Jan M. Rabaey, Anantha Chandrakasan, and Borivoje Nikolic, Digital Integrated Circuits: A Design Perspective, Pearson, 2nd Edition, 2016.
2. Sung-Mo Kang and Yusuf Leblebici, CMOS Digital Integrated Circuits: Analysis and Design, McGraw Hill, 4th Edition, 2014.
3. William J. Dally and John W. Poulton, Digital Systems Engineering, Cambridge University Press, 1st Edition, 1998.

Reference Books

1. David Harris and Sarah Harris, CMOS VLSI Design: A Circuits and Systems Perspective, Pearson, 4th Edition, 2012.
2. Neil H.E. Weste and David Money Harris, CMOS VLSI Design: A Circuits and Systems Perspective, Pearson, 4th Edition, 2010.
3. Charles E. Stroud, Basic Concepts in Digital Design with VHDL, Elsevier, Latest Edition.
4. Wayne Wolf, Modern VLSI Design: IP-Based Design, Pearson, 4th Edition, 2009.

Mode of Evaluation: Assignments, Mid Term Tests, End Semester Examination.