# M.Tech (VLSI & Embedded Systems)  
**PG Course Structure-R20**

## I SEMESTER

<table>
<thead>
<tr>
<th>S.No</th>
<th>Course code</th>
<th>Course Title</th>
<th>Contact hours/week</th>
<th>Credits</th>
<th>Scheme of Valuation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>20VE01</td>
<td>Digital VLSI System Design</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>20VE02</td>
<td>Embedded System Design</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Program Elective-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20VE03</td>
<td>Cryptography and Network Security</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>20VE04</td>
<td>Design for Internet of Things</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20VE05</td>
<td>High speed VLSI Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Program Elective-2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20VE06</td>
<td>Image and Video Processing</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>40</td>
</tr>
<tr>
<td>20VE07</td>
<td>System Modeling and Simulation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20VE08</td>
<td>VLSI Design Automation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>20RM01</td>
<td>Research Methodology and IPR</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>6</td>
<td>AC</td>
<td>Audit course-1</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>20VE61</td>
<td>Digital VLSI System Design Lab</td>
<td>4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>20VE62</td>
<td>Embedded System Design Lab</td>
<td>4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td><strong>Total</strong></td>
<td></td>
<td>16</td>
<td>0</td>
<td>8</td>
</tr>
</tbody>
</table>
## II SEMESTER

<table>
<thead>
<tr>
<th>S.No</th>
<th>Course code</th>
<th>Course Title</th>
<th>Contact hours/week</th>
<th>Credits</th>
<th>Scheme of Valuation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>20VE09</td>
<td>Analog VLSI Design</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>20VE10</td>
<td>Real Time Operating Systems</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Program Elective-3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>20VE11</td>
<td>CPLD and FPGA Architectures</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>20VE12</td>
<td>Embedded Software Design</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>20VE13</td>
<td>VLSI Testing and Verification</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>20VE14</td>
<td>Wireless Communications and Networks</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td>20VE15</td>
<td>ASIC Design</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>20VE16</td>
<td>VLSI Architecture for Signal Processing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>AC</td>
<td>Audit course-2</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>20VE63</td>
<td>Analog VLSI Design Lab</td>
<td>4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>20VE64</td>
<td>Real Time Operating Systems Lab</td>
<td>4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>8</td>
<td>20VE51</td>
<td>Mini Project</td>
<td>4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total</td>
<td>14</td>
<td>0</td>
<td>12</td>
</tr>
</tbody>
</table>

M.Tech. (VLSI & Embedded Systems)  R20 Regulations (w.e.f. 2020-21)  Page 2 of 60
### III SEMESTER

<table>
<thead>
<tr>
<th>S.No</th>
<th>Course code</th>
<th>Course Title</th>
<th>Contact hours/week</th>
<th>Credits</th>
<th>Scheme of Valuation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>Program Elective-5 / MOOCs</td>
<td></td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>20VE17 Embedded Linux</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>20VE18 MEMS Design and Fabrication</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>20VE19 Low Power VLSI Design</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Open Elective / MOOCs</td>
<td></td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>20CO81 Advanced Computer Architecture</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>20CO82 Python Programming</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>20CO83 Introduction to Machine Learning</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>20VE52 Internship</td>
<td></td>
<td>4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>20VE53 Project Work &amp; Dissertation (Phase-I)</td>
<td></td>
<td>16</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td></td>
<td>6</td>
<td>0</td>
<td>20</td>
</tr>
</tbody>
</table>

**Total Credits: 18 + 18 + 16 + 16 = 68**

### IV SEMESTER

<table>
<thead>
<tr>
<th>S.No</th>
<th>Course code</th>
<th>Course Title</th>
<th>Contact hours/week</th>
<th>Credits</th>
<th>Scheme of Valuation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L</td>
<td>T</td>
<td>P</td>
</tr>
<tr>
<td>1</td>
<td>20VE54 Project Work &amp; Dissertation (Phase-II)</td>
<td></td>
<td>32</td>
<td>32</td>
<td>16</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td></td>
<td>0</td>
<td>0</td>
<td>32</td>
</tr>
</tbody>
</table>

Total Credits: 18 + 18 + 16 + 16 = 68
List of Courses offered under Audit Course

<table>
<thead>
<tr>
<th>S.No</th>
<th>Code</th>
<th>Name of the Course</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20AC01</td>
<td>English for research paper writing</td>
</tr>
<tr>
<td>2</td>
<td>20AC02</td>
<td>Disaster Management</td>
</tr>
<tr>
<td>3</td>
<td>20AC03</td>
<td>Sanskrit for Technical Knowledge</td>
</tr>
<tr>
<td>4</td>
<td>20AC04</td>
<td>Value education</td>
</tr>
<tr>
<td>5</td>
<td>20AC05</td>
<td>Constitution of India</td>
</tr>
<tr>
<td>6</td>
<td>20AC06</td>
<td>Pedagogy Methods</td>
</tr>
<tr>
<td>7</td>
<td>20AC07</td>
<td>Stress Management by Yoga</td>
</tr>
<tr>
<td>8</td>
<td>20AC08</td>
<td>Personality Development through Life Enlightenment Skills.</td>
</tr>
</tbody>
</table>

List of Open Elective Courses offered to other Departments

<table>
<thead>
<tr>
<th>S.No</th>
<th>Code</th>
<th>Open Elective Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20VE81</td>
<td>CAD for VLSI Design</td>
</tr>
<tr>
<td>2</td>
<td>20VE82</td>
<td>Programmable Logic Devices</td>
</tr>
<tr>
<td>3</td>
<td>20VE83</td>
<td>VLSI Testing</td>
</tr>
</tbody>
</table>
M.Tech (VLSI & Embedded Systems) - I Semester

Detailed Syllabus
**Pre-Requisites:** STLD, CO

**COURSE EDUCATIONAL OBJECTIVES:**
In this course student will learn about the design of various combinational and sequential circuits using verilog HDL and VHDL, write the verilog tasks, functions, and various digital modules.

**COURSE OUTCOMES:** At the end of this course student will be able to
CO1: Design Combinational and Sequential circuits.
CO2: Understand Digital System Design flow using Verilog HDL.
CO3: Model Digital System Using Verilog HDL.
CO4: Write Verilog Tasks, Functions, UDPs for Digital modules.

**UNIT-I: REVIEW OF LOGIC DESIGN FUNDAMENTALS**

**UNIT-II: INTRODUCTION TO VERILOG**

**UNIT-III: DESIGN EXAMPLES**

**UNIT-IV: SM CHARTS AND MIRCOPROGRAMMING**
UNIT-V: ADDITIONAL TOPICS IN VERILOG & DESIGN OF A RISC MICROPROCESSOR


TEXT BOOK:


REFERENCE:

Pre-Requisites: Computer Organization, Microprocessors and Microcontrollers

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about the design methodologies of embedded systems, design the control unit and data-path unit for a given embedded system, interface between ARM processor, memory and co-processor etc.

COURSE OUTCOMES: At the end of this course student will be able to
CO1: Choose different design methodologies to implement given specifications.
CO2: Design control unit and Data path unit for given Embedded System.
CO3: Use ARM processor architecture for development of Embedded System.
CO4: Develop interface between ARM processor core, memory and co-processor.
CO5: Build frame work for embedded system using IC & design technology.

Unit I: Embedded System Introduction: Embedded systems overview, design challenge, processor technology, IC technology, Design Technology, Trade-offs. Single purpose processors RT-level combinational logic, sequential logic(RT level), custom single purpose processor design(RT –level), optimizing custom single purpose processors.

Unit II: State Machine and Concurrent Process Models: Introduction, models Vs languages, finite state machines with data path model(FSMD), using state machines, program state machine model (PSM), concurrent process model, concurrent process model, concurrent processes, communication among processes, synchronization among processes, Implementation, data flow model, real-time systems.

Unit III: ARM Processor Architecture: The ARM programmer’s model, ARM development tools, ARM Assembly Language Programming: Data processing instructions, Data transfer instructions, Control flow instructions, writing simple assembly language programs,


TEXT BOOKS:

REFERENCES:
Programme Elective-I

20VE03– CRYPTOGRAPHY & NETWORK SECURITY

Pre-Requisites: Computer Networks

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about the various security attacks and their characteristics, analyze the cryptographic techniques, concepts of digital signatures and its applications.

COURSE OUTCOMES: At the end of this course student will be able to
CO1: Discuss various types of attacks and their characteristics.
CO2: Analyze various cryptographic techniques.
CO3: Design public key cryptographic algorithms using the concepts from number theory.
CO4: Describe the concept of digital signature and its applications.
CO5: Illustrate the concepts of IP and Web security.

UNIT-I
Introduction: Security Attacks- Passive Attacks, Active Attacks, Security Service-
Different Security Services, Security Mechanism- Different Security Mechanisms,
A Model for Network Security.
Classical Encryption Techniques: Simplified model of Conventional Encryption,
Model of Conventional Cryptosystem, Substitution Techniques- Caesar Cipher,
Monoalphabetic cipher, Play Fair Cipher, Transposition Techniques- Rail Fence Cipher,
Columnar Transposition, Stegnography and Techniques.

UNIT-II
Block Ciphers and the Data Encryption Standard: Block Cipher Principles,
Classical Feistel Network, Data Encryption Standard (DES) and DES Encryption,
Strength of DES, Block Cipher Design Principles, Block Cipher Modes of operation: ECB, CBC, CFB, OFB, CTR, Blowfish Algorithm, Random Number Generation.

Public Key Cryptography: Encryption, Authentication, Secrecy, Requirements of
Public Key Cryptography, RSA Algorithm (Rivest-Shamir-Adleman), Diffie-
Hellman Key Exchange.

UNIT-III
Number theory: Prime and Relatively prime numbers, Modular arithmetic,
Fermat’s and Euler’s theorems, Testing for primality, Euclid’s Algorithm, the
Chinese remainder theorem, Discrete logarithms.
Message authentication and Hash functions: Authentication Requirements,
UNIT-IV
Hash and Mac Algorithms: MD File, Message digest Algorithm, Secure Hash Algorithm, RIPEMD-160, and HMAC.

UNIT-V
Intruders, Viruses and Worms: Intruders, Viruses and Related threats.
Fire Walls: Fire wall Design Principles, Trusted systems.

Text Book:

References:
1. Principles of Network and Systems Administration, Mark Burgess, John Wiel
Programme Elective-I
M.Tech. (I Sem.)  
20VE04 – DESIGN FOR INTERNET OF THINGS (IoT)

<table>
<thead>
<tr>
<th>L</th>
<th>P</th>
<th>T</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>-</td>
<td>-</td>
<td>3</td>
</tr>
</tbody>
</table>

**Pre-Requisites:** Networking Knowledge

**COURSE EDUCATIONAL OBJECTIVES:**
In this course student will learn about the Technologies used in IoT, build platform for various IoTs, analyzing the IoT architecture, performance and security aspects.

**COURSE OUTCOMES:** At the end of this course student will be able to
CO1: Understand technologies involved in IoT Development.
CO3: Analyze IoT architecture for performance and security aspects.
CO4: Develop IoT application using various protocols.
CO5: Integrate Big Data and Visualization issues in IoT.

**UNIT I:**
The IoT Networking Core : Technologies involved in IoT Development: Internet/Web and Networking Basics OSI Model, Data transfer referred with OSI Model, IP Addressing, Point to Point Data transfer, Point to Multi Point Data transfer & Network Topologies, Sub-netting, Network Topologies referred with Web, Introduction to Web Servers, Introduction to Cloud Computing

**UNIT II:**

**UNIT III:**

**UNIT IV:**
UNIT V:

TEXT BOOKS:
1. LoWPAN: The Wireless Embedded Internet, Zach Shelby, Carsten Bormann, Wiley
2. Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems, Dr. OvidiuVermesan, Dr. Peter Friess, River Publishers
3. Interconnecting Smart Objects with IP: The Next Internet, Jean-Philippe Vasseur, Adam Dunkels, Morgan Kuffmann

REFERENCES:
1. The Internet of Things: From RFID to the Next-Generation Pervasive Networked Lu Yan, Yan Zhang, Laurence T. Yang, HuanshengNing
2. Internet of Things (A Hands-on-Approach) , Vijay Madisetti , ArshdeepBahga
3. Designing the Internet of Things , Adrian McEwen (Author), Hakim Cassimally
Programme Elective-I

M.Tech. (I Sem.) 20VE05– HIGH SPEED VLSI DESIGN

Pre-Requisites: VLSI Design

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about the frequency and time metrics, various power and clock distribution techniques for ICs, and various Latching strategies used in ICs.

COURSE OUTCOMES: At the end of this course student will be able to
CO1: Analyze need for High speed VLSI design using frequency, time metrics.
CO2: Use power distribution on-chip for high speed chips.
CO3: Design signaling convention and circuits for on-chip transmission lines.
CO4: Resolve timing convention and synchronization issues on-chip.
CO5: Develop clocked, non clocked logic and latching strategies for high speed design.

UNIT I:
Introduction to high speed digital design: Frequency, time and distance issues in digital VLSI design. Capacitance and inductance effects, high speed properties of logic gates, speed and power. Modeling of wires, geometry and electrical properties of wires, Electrical models of wires, transmission lines, lossless LC transmission lines, lossy RLC transmission lines and special transmission lines.

UNIT II:

UNIT III:
Signaling convention and circuits: Signaling modes for transmission lines, signaling over lumped transmission media, signaling over RC interconnect, driving lossy LC lines, simultaneous bi-directional signaling terminations, transmitter and receiver circuits.

UNIT IV:
Timing convention and synchronization: Timing fundamentals, timing properties of clocked storage elements, signals and events, open loop timing, level sensitive clocking, pipeline timing, closed loop timing, clock distribution, synchronization failure and meta-stability, clock distribution, clock skew and methods to reduce clock skew, controlling crosstalk in clock lines, delay adjustments, clock oscillators and clock jitter - PLL and DLL based clock aligners.
UNIT V:
Clocked & non clocked Logics:
Single-Rail Domino Logic, Dual-Rail Domino Structures, Latched Domino Structures, Clocked Pass Gate Logic, Static CMOS, DCVS Logic, Non-Clocked Pass Gate Families.

Latching Strategies:
Basic Latch Design, and Latching single-ended logic and Differential Logic, Race Free Latches for Pre-charged Logic Asynchronous Latch Techniques

TEXT BOOKS:

REFERENCE BOOKS:
Pre-Requisites: Digital Image Processing

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about the various compression, enhancement and segmentation techniques, image transformation techniques, and algorithms for 2D video signals.

COURSE OUTCOMES: At the end of this course student will be able to
CO1: Examine different image transformation techniques. 
CO2: Evaluate various image enhancement and segmentation techniques. 
CO3: Analyze different compression techniques for images. 
CO4: Infer time-varying image formation models in video. 
CO5: Compare 2-D motion estimation algorithms for video signals.

UNIT I

UNIT II
Image Processing Techniques: 
Image Segmentation: Segmentation concepts, Point, Line and Edge Detection, Thresholding, Region Based segmentation.

UNIT III
Image Compression: Image compression fundamentals - Coding Redundancy, Spatial and Temporal redundancy, Compression models: Lossy & Lossless, Huffman coding, Arithmetic coding, LZW coding, Run length coding, Bit plane coding, Transform coding, Predictive coding, Wavelet coding, JPEG Standards.

UNIT IV

UNIT V 2-D
Motion Estimation: Optical flow, General Methodologies, Pixel Based Motion Estimation, Block- Matching Algorithm, and Mesh based Motion Estimation, Global Motion Estimation, Region based Motion Estimation, Multi resolution motion estimation, Application of motion estimation in Video coding.
Text Books:

References:
Programme Elective-II
M.Tech. (I Sem.) 20VE07– SYSTEM MODELING AND SIMULATION

<table>
<thead>
<tr>
<th>L</th>
<th>P</th>
<th>T</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>-</td>
<td>-</td>
<td>3</td>
</tr>
</tbody>
</table>

**Pre-Requisites:** Random variables

**COURSE EDUCATIONAL OBJECTIVES:**
In this course student will learn about the various Simulation and Programming Models, Optimization methodologies.

**COURSE OUTCOMES:** At the end of this course student will be able to
CO1: Understand basic simulation models, characterizing and simulation diagrams.
CO2: Model time driven systems and stochastic processes.
CO3: Know how to simulate any discrete systems using queuing systems.
CO4: Discuss system optimization, modeling and simulation methodology.
CO5: Build simulation models with programming languages.

**Unit – I**

**Unit – II**

**Unit – III**

**Unit – IV**
**System Optimization:** System Identification, Searches, Alpha / Beta trackers, Multidimensional Optimization, Modeling and Simulation Methodology.

**Unit – V**
**Simulation Software and Building Simulation Models:**
Comparison of Simulation Packages with Programming Languages, Classification of Simulation Software, Desirable software features, General Purpose Simulation Packages-Arena, Extend; Guide lines for determining the level of Model detail, Techniques for increasing Model Viability and credibility.
TEXT BOOKS:

REFERENCES:
1. Systems Simulation - Geoffery Gordan, PHI
Pre-Requisites: VLSI Design

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about the design cycles, various techniques on Partitioning, Placement and Routing and addressing their problems.

COURSE OUTCOMES: At the end of this course student will be able to
CO1: Understand need for VLSI physical design automation.
CO2: Analyze VLSI automation algorithms for partitioning.
CO3: Formulate placement , floor planning and pin assignment problems and simulate.
CO4: Resolve routing issues using various algorithms.
CO5: Illustrate physical design cycle for FPGAs.

UNIT I:
VLSI Physical Design Automation: Introduction, VLSI Design cycle, new trends in VLSI design cycle, new trends in Physical design cycle, Design styles, full custom Basic terminology, complex issues, basic algorithms, Basic data structures, and algorithms.

UNIT II:

UNIT III:
Placement, Floor Planning & Pin Assignment: problem formulation, simulation base placement algorithms, other placement algorithms, constraint based floor planning, floor planning algorithms for mixed block & cell design. General & channel pin assignment

UNIT IV:
Global Routing: Problem formulation, classification of global routing algorithms, Maze routing algorithm, line probe algorithm, Steiner Tree based algorithms, ILP based approaches
Detailed Routing: Problem formulation, classification of routing algorithms, single layer routing algorithms, two layer channel routing algorithms, three layer channel routing algorithms, and switchbox routing algorithms.
Over The Cell Routing & Via Minimization: two layers over the cell routers, constrained & unconstrained via minimization

UNIT V:
Physical design Automation of FPGAs:
Introduction, FPGA Technologies, Physical design cycle for FPGAs, Partitioning, Routing, Routing algorithms for the non-segmented model, Routing algorithms for segmented model, routing algorithms for staggered model.
TEXT BOOK:

REFERENCE BOOKS:
Pre-requisites: Knowledge in Engineering, English

Course Objective: To understand the research problem, to know the literature studies, plagiarism and ethics, to get the knowledge about technical writing, to analyse the nature of intellectual property rights and new developments and research related information and to know the patent rights

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

CO1 Analyze the research problem and its formulation.
CO2 Analyze the significance of research ethics
CO3 Apply the information technology for better tomorrow and to develop creativity.
CO4 Identify the importance of intellectual property rights to be promoted among students in general & engineering in particular
CO5 Describe the IPR protection for new and better products, and in turn brings about, economic growth and social benefits.

UNIT I - RESEARCH PROBLEM AND SCOPE FOR SOLUTION
Meaning of research problem, Sources of research problem, Criteria Characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem. Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, Necessary instrumentations

UNIT II- FORMAT
Effective literature studies approaches, analysis, Plagiarism, Research ethics. 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 III- PROCESS AND DEVELOPMENT

UNIT IV- PATENT RIGHTS

UNIT V- NEW DEVELOPMENTS IN IPR
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.
REFERENCES

   engineering students”
2. Wayne Goddard and Stuart Melville, “Research Methodology: An Introduction”
   beginners”
   Hall, 1962.
Course Educational Objective: This course gives knowledge on research paper writing skills. This course also describes the different section of the research paper. This course also provides skills that are needed for publication which includes paper preparation, plagiarism check and submission process.

Course Outcomes (COs): At the end of the course, students will be able to
CO1: Understand research article readability and writing skills.
CO2: Identify essential parameters of each section of research articles.
CO3: Apply the knowledge for writing and submit research paper for publication.
CO4: Develop skills that are required to maintain quality of research paper.

UNIT-I: Research paper Writing Skills Part-I
Planning and Preparation, Word Order, Breaking up long sentences, Structuring, Paragraphs and Sentences, Being Concise, and Removing Redundancy

UNIT-II: Research paper Writing Skills Part-II
Avoiding Ambiguity and Vagueness, Clarifying Who Did What, Highlighting Your Findings, Hedging and Criticising, Paraphrasing and Plagiarism

UNIT-III: Parts of research Paper
Title, Abstract, Introduction, Review of the Literature, Methods, Results, Discussion, Conclusions, the Final Check.

UNIT-IV: Preparation of manuscript
Key skills are needed when writing a Title, an Abstract, an Introduction, a Review of the Literature, the Methods, the Results, the Discussion, Conclusions, preparing the tables and figures

UNIT-V: Publishing the Paper
Rights and Permission, How to Submit the Manuscript, The Review Process (How to Deal with Editors), The Publishing Process (How to Deal with Proofs) and After Publication.

Text Books:
Pre-Requisites: ECAD Lab

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about the VHDL Programming of various Digital Circuits, design and implementation of various CMOS IC circuits using Mentor Graphics Tools.

COURSE OUTCOMES: At the end of this course student will be able to
CO1: Design CMOS Logic gates using Pyxis Schematic Editor.
CO2: Model digital modules using VHDL/Verilog and Simulate.

Syllabus:
- Minimum of 10 programs from Part –I and 2 programs from Part -II are to be conducted.
- Design and implementation of the following CMOS digital circuits using Mentor Graphics /Xilinx CAD tools. The design shall include Gate-level design, Transistor-level design, Hierarchical design, Verilog HDL/VHDL design, Logic synthesis, Simulation and verification.

Part –I: VLSI Front End Design programs:
Programming can be done using any compiler. Down load the programs on FPGA/CPLD boards and performance testing may be done using pattern generator (32 channels) and logic analyzer apart from verification by simulation with any of the front end tools.

1. Design of 8 input priority encoder and 3 to 8 Decoder
2. Design of 8x1Mux and 1x8 Demux
3. Design of 4 bit adder
4. Design 4 bit Magnitude Comparator
5. Design of BCD adder
6. Design of ALU
7. Design of D,T,JK Flip Flops
8. Design of Shift Registers
9. Design of Counters
10. Design of FSM( Moore’s, Mealy Machines)
11. Design of Booth multiplier
12. Design of Electronic Dice game
Part -II: VLSI Back End Design programs:

- Design and implementation of the following CMOS digital/analog circuits using Cadence / Mentor Graphics / Synopsys / Equivalent CAD tools.
- The design shall include Gate-level design/Transistor level design/Hierarchical design/Verilog HDL or VHDL design, Logic synthesis, Simulation and verification, Scaling of CMOS Inverter for different technologies, study of secondary effects (temperature, power supply and process corners), Circuit optimization with respect to area, performance and/or power, Layout, Extraction of parasitics and back annotation, modifications in circuit parameters and layout consumption, DC/transient analysis, Verification of layouts (DRC, LVS).

1. Introduction to layout design rules

2. Layout, physical verification, placement & route for complex design, static timing analysis, IR drop analysis and crosstalk analysis of the following:
   - Basic logic gates
   - CMOS inverter
   - CMOS NOR/ NAND gates
   - CMOS XOR and MUX gates
   - CMOS 1-bit full adder
   - Static / Dynamic logic circuit (register cell)
   - Latch
   - Pass transistor

3. Layout of any combinational circuit (complex CMOS logic gate)- Learning about data paths
Pre-Requisites: Microprocessors and Microcontrollers Lab

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about the programming in Assembly language programs on ARM processor kits, programming on CPLD and FPGA kits using EDK Tools.

COURSE OUTCOMES:
At the end of this course student will be able to
CO1: Develop the Assembly Language Programs for ARM processors.
CO2: Develop the interfacing programs for ARM processors and I/O devices.
CO3: Design an Embedded System using the FPGA and EDK tools.

Syllabus:

Part I
- The following programs are to be implemented on ARM based Processors/Equivalent.
- Minimum of 10 programs are to be conducted.

1. ARM Assembly Language Programming-I
2. ARM Assembly Language Programming-II
3. Program to Interface 8 Bit LED
4. Program to demonstrate Time delay program using built in Timer/Counter feature
5. Program to Displaying a message in a 2 line x 16 Characters LCD display and verify the result in debug terminal.
6. Generation of PWM Signal
7. Serial Communication
8. Traffic light Controller
9. Stepper motor Controller
10. Basic Audio Processing on IDE environment.
11. Program to demonstrate I2C Interface on IDE environment
12. Program to implement Buzzer Interface on IDE environment

Part II

The following programs are to be implemented on Xilinx FPGA Zynq 7000 series/Equivalent.

- Minimum of 2 programs are to be conducted.

14. Design dual processor based System on chip using Xilinx EDK Tools and Zynq 7000 series FPGA.
15. Hardware Software co-design using Xilinx EDK Tools and Advanced FPGA Board Zynq 7000 series.
M.Tech(VLSI & ES)-II Semester

Detailed Syllabus
**PRE-REQUISITES:** VLSI Design

**COURSE EDUCATIONAL OBJECTIVES:**
In this course student will learn about the operation of MOS devices, design of Oscillators, Phase Locked Loops, Reference Generators and Data Converters.

**COURSE OUTCOMES:**
At the end of the course, student will be able to

CO1: Understand the basic physics and operation of MOS devices.
CO2: Develop efficient analytical tools for quantifying the behavior of basic circuits by inspection.
CO3: Design of Oscillators and Phase Locked Loops.
CO4: Develop reference generators.
CO5: Implement efficient data converters.

**UNIT I:**
**Basic MOS Device Physics:** General considerations, MOS I/V Characteristics, second order effects, MOS device models.
**Single stage Amplifier:** CS stage with resistance load, divide connected load, current source load, triode load, CS stage with source degeneration, source follower, common-gate stage, cascade stage, choice of device models.

**UNIT II:**
**Frequency response of CS stage:** source follower, Common gate stage, Cascade stage and Difference pair. Noise in CS stage, C- G stage, source follower, cascade stage, differential pair.
**Differential Amplifiers & Current Mirrors:** Basic difference pair, common mode response, Differential pair with MOS loads, Gilbert cell. Basic current mirrors, Cascade mirrors, active current mirrors.

**UNIT III:**
**Oscillators and Phase Locked Loops:** Ring Oscillators, LC Oscillators, VCO, Mathematical Model of VCO. Simple PLL, Charge pump PLL, Non-ideal effects in PLL, Delay locked loops and applications.

**UNIT IV:**
**Band gap References** and Switched capacitor Circuits: General Considerations, Supply Independent biasing, PTAT Current Generation, Constant Gm Biasing, Sampling Switches, Switched Capacitor Amplifiers.

**UNIT V:**
**Data Converter Architectures:** DAC & ADC Specifications, Resistor String DAC, R-2R Ladder Network, Current Steering DAC, Charge Scaling DAC, Cyclic DAC, Pipeline DAC, Flash ADC, Pipeline ADC, Integrating ADC, Successive Approximation ADC.
TEXT BOOK:

REFERENCE BOOK:
PRE-REQUISITES: Embedded System Design

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about fundamental concepts of real time operating systems, operating system objects, services and I/O concepts, various interrupts and timers.

COURSE OUTCOMES:
At the end of the course, student will be able to

CO1 Understand the basic set of commands and utilities in Linux/UNIX systems.
CO2 Explain the fundamental concepts of real-time operating systems.
CO3 Analyze real-time operating systems objects, services and I/O concepts.
CO4 Evaluate various Interrupts and Timers.
CO5 Design real time embedded systems using the concepts of RTOS.

UNIT – I:
Introduction: Introduction to UNIX/LINUX, Overview of Commands, File I/O (open, create, close, lseek, read, write), Process Control (fork, vfork, exit, wait, waitpid, exec).

UNIT - II:

UNIT - III:
Objects, Services and I/O Pipes, Event Registers, Signals, Other Building Blocks, Component Configuration, Basic I/O Concepts, I/O Subsystem

UNIT - IV:

UNIT V:
RT Linux, MicroC/OS-II, Vx Works, Embedded Linux, Tiny OS, and Basic Concepts of Android OS.
TEXT BOOKS:

REFERENCE BOOKS:
1. Embedded Systems- Architecture, Programming and Design by Rajkamal, 2007, TMH.
2. Advanced UNIX Programming, Richard Stevens
3. Embedded Linux: Hardware, Software and Interfacing – Dr. Craig Hollabaugh
M.Tech. (II Sem.)  

**Programme Elective-III**

**20VE11 - CPLD AND FPGA ARCHITECTURES**

<table>
<thead>
<tr>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>-</td>
<td>-</td>
<td>3</td>
</tr>
</tbody>
</table>

**Pre-Requisites:** Basics on CPLD and FPGA

**COURSE EDUCATIONAL OBJECTIVES:**
In this course student will learn about the complex programmable logic devices, field programmable gate arrays, architecture of SRAM programmable and anti-fuse programmed FPGAs.

**COURSE OUTCOMES:** At the end of this course student will be able to

- **CO1** Analyze different types of Complex Programmable Logic Devices.
- **CO2** Understand different types of Field Programmable Gate Arrays.
- **CO3** Evaluate architecture of SRAM Programmable FPGAs.
- **CO4** Explain the device Architecture of Anti-Fuse Programmed FPGAs.
- **CO5** Design the application for Combinational and Sequential Circuits.

**UNIT - I:**

**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.

**UNIT – II:**

**UNIT – III:**

**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 – IV:**

**SRAM Programmable FPGAs:**

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

**UNIT – V:**

**Design Applications:**
General Design Issues, Counter Examples, A Fast Video Controller, A Fast DMA Controller, Designing Counters with ACT devices, Designing Adders and Accumulators with the ACT Architecture.
TEXTBOOKS

REFERENCES:
1. Field Programmable Gate Arrays by John V. Oldfield, Richard C. Dorf, Wiley India.
3. Digital Systems Design with FPGAs and CPLDs by Ian Grout, Elsevier, Newnes.
Programme Elective-III

M.Tech. (II Sem.) 20VE12- EMBEDDED SOFTWARE DESIGN

<table>
<thead>
<tr>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>-</td>
<td>-</td>
<td>3</td>
</tr>
</tbody>
</table>

PRE-REQUISITES: Embedded Systems and Operating Systems

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about Operating systems concepts, programming concepts, programming in Linux and RT Linux.

COURSE OUTCOMES:
At the end of the course, student will be able to

CO1 **Understand** the basic concepts of Operating Systems.
CO2 **Explain** the Programming concepts of RTOS.
CO3 **Analyze** various Case Studies for practical applications.
CO4 **Create** Target Image for Windows XP Embedded & **write** Programming in Linux.
CO5 **Write** Programming in RT Linux.

UNIT-I: Introduction


UNIT-II: RTOS Programming

Basic Functions and Types of RTOS for Embedded Systems, RTOS mCOS-II, RTOS Vx Works, Programming concepts of above RTOS with relevant Examples, Programming concepts of RTOS Windows CE, RTOS OSEK, RTOS Linux 2.6.x and RTOS RT Linux.

UNIT-III: Program Modeling – Case Studies

Case study of embedded system design and coding for an Automatic Chocolate Vending Machine (ACVM) Using Mucos RTOS, case study of digital camera hardware and software architecture, case study of coding for sending application layer byte streams on a TCP/IP Network Using RTOS Vx Works, Case Study of Embedded System for an Adaptive Cruise Control (ACC) System in Car, Case Study of Embedded System for a Smart Card, Case Study of Embedded System of Mobile Phone Software for Key Inputs.
UNIT-IV: Target Image Creation & Programming in Linux


UNIT-V: Programming in RT Linux

Overview of RT Linux, Core RT Linux API, Program to display a message periodically, semaphore management, Mutex, Management, Case Study of Appliance Control by RT Linux System.

TEXT BOOKS:


REFERENCES:


M.Tech. (I Sem.) 20VE13 – VLSI TESTING AND VERIFICATION

**PRE-REQUISITES:** None

**COURSE EDUCATIONAL OBJECTIVES:**
In this course student will learn about testable design, test generation algorithms for combinational and sequential circuits, design verification and verification tools, timing and physical design verification.

**COURSE OUTCOMES:**
At the end of the course, student will be able to

- CO1 Identify the significance of testable design
- CO2 Implement combinational and sequential circuit test generation algorithms
- CO3 Understand the importance of Design verification.
- CO4 Learn verification tools.
- CO5 Analyze the static timing verification and physical design verification.

**UNIT I:**

**Test Generation for Combinational Logic Circuits:** Fault Diagnosis of Digital Circuits, Test Generation Techniques for Combinational Circuits, Detection of Multiple Faults in Combinational Logic Circuits.

**UNIT II:**
**Design of Testable Sequential Circuits:** Controllability and Observability, Ad Hoc Design Rules for Improving Testability, Design of Dignosable Sequential Circuits, The Scan-Path Technique for Testable Sequential Circuit Design, Level-Sensitive Scan Design, Random Access Scan Technique, Partial Scan, Testable Sequential Circuit Design Using Nonscan Techniques, CrossCheck, Boundry Scan. Built-In Self Test: Test Pattern Generation for BIST, Output Response Analysis, Circular BIST, BIST Architectures.

**UNIT III:**
**Testable Memory Design:** RAM Fault Models, Test Algorithms for RAMs, Detection of Pattern Sensitive Faults, BIST Techniques for Ram Chips, Test Generation and BIST for Embedded RAMs.

**Importance of Design Verification:** What is verification? What is attest bench? The importance of verification, Reconvergence model, Formal verification, Equivalence checking, Model checking, Functional verification.[Ref4- Chapter1]
UNIT IV:
**Verification Tools:** Linting tools: Limitations of linting tools, linting verilog source code, linting VHDL source code, linting OpenVera and esource code, code reviews. Simulators: Stimulus and response, Event based simulation, cycle based simulation, Co-simulators, verification intellectual property: hardware modelers, waveform viewers.[Ref4-Chapter2]

**The verification plan:** The role of verification plan: specifying the verification plan, defining the first success. Levels of verification: unit level verification, reusable components verification, ASIC and FPGA verification, system level verification, board level verification, verifying strategies, verifying responses. [Ref4-Chapter3]

UNIT V:
**Static Timing Verification:** Concept of static timing analysis. Cross talk and noise. Limitations of STA. slew of a wave form, Skew between the signals, Timing arcs and unateness, Min and Max timing paths, clock domains, operating conditions, critical path analysis, falsepaths, Timing models. [Ref5 Chapter 1, 2, 3, 8]

**Physical Design Verification:** Layout rule checks and electrical rule checks. Parasitic extraction. Antenna, Crosstalk and Noise: Cross talk glitch analysis, crosstalk delay analysis, timing verification [Ref6 Chapter 8]

**TEXT BOOKS:**

**REFERENCE BOOKS:**
PRE-REQUISITES: Communication Systems

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about wireless components, wireless networks, wireless system architecture, protocols, Mobile IP, Adhoc networks, digital audio and video broadcasting for high speed internet access.

COURSE OUTCOMES:
At the end of the course, student will be able to

CO 1: Discuss about wireless components and wireless networks.

CO 2: Describe System architecture and medium access control protocols.

CO 3: Explain Mobile IP and mobile Adhoc networks.

CO 4: Compare different transport layer protocols.

CO 5: Analyze requirements of digital audio and video broadcasting for high-speed internet access.

UNIT I

Wireless LAN: Infra redVs radio transmission, Infrastructure and Ad-hoc Network.


UNIT II:

(Wireless) Medium Access Control: Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA.

UNIT III:
Mobile Network Layer:
Mobile IP: Goals, assumptions, entities and terminology, IP packet delivery, agent advertisement and discovery, registration, tunneling and encapsulation, optimizations, Dynamic Host Configuration Protocol (DHCP).
Mobile Ad hoc Networks (MANETs): Overview, Properties of a MANET, spectrum of MANET applications, routing and various routing algorithms.

UNIT IV
Mobile Transport Layer: Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit/fast recovery, Transmission /time-out freezing, Selective retransmission, Transaction oriented TCP.

UNIT V

TEXT BOOK:

REFERENCE BOOKS:
PRE-REQUISITES: CPLD & FPGA

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about ASIC design flow, architecture of ASIC library design, programmable ASIC logic cells, logic synthesis and physical design phases.

COURSE OUTCOMES:
At the end of the course, student will be able to

CO1: Acquire Knowledge on ASIC design flow
CO2: Analyze the architecture of ASIC library designs
CO3: Understanding the programmable ASIC logic cells
CO4: Analyzing the Logic synthesis in HDLs
CO5: Comprehend the physical design phases

Unit I: Types of ASICs – Design flow – Economics of ASICs – ASIC cell libraries – CMOS logic: CMOS Transistors, The CMOS process, CMOS design rules, Combinational logic cells, sequential logic cells, data path logic cells – I/O cells – cell compilers.

Unit II: ASIC Library design: Transistors as resistors – parasitic capacitance – logical effort, library cell design, library architecture, gate array design, standard cell design, data path cell design programmable ASICs: The Anti-fuse, Static RAM, EPROM and EEPROM Technology, practical issues, specifications.


Unit IV: Logic synthesis in Verilog and logic synthesis in VHDL, Finite –State machine synthesis, Memory synthesis, performance-Driven synthesis, Simulation: Types of simulation, The Comparator/MUX example, logic systems, how logic simulation works, cell models, delay models, Static Timing Analysis, formal verification, switch-level simulation, transistor level simulation.

TEXT BOOKS:


REFERENCES:

2. Introduction to VLSI System,C. Mead & L. Canway, Addison Wesley Pub
3. Introduction to NMOS & VLSI System Design, A. Mukharjee, Prentice Hall,
PRE-REQUISITES: DSP

COURSE EDUCATIONAL OBJECTIVES:

In this course student will learn about the fundamentals of VLSI signal processing and expose them to examples of applications. Design and optimize VLSI architectures for basic DSP algorithms.

COURSE OUTCOMES:

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

1. Understand VLSI design methodology for signal processing systems.
2. Differentiate between folding and unfolding architectures.
3. Elaborate various Systolic Array structures.
4. Analyze the VLSI algorithms and architectures for DSP.
5. Implement basic architectures for DSP using CAD tools.

UNIT -I:
Introduction to DSP: Typical DSP algorithms, DSP algorithms benefits, Representation of DSP algorithms

UNIT –II:

UNIT -III:

UNIT -IV:
UNIT -V:

TEXT BOOKS:

REFERENCE BOOKS:
AUDIT COURSE
20AC05 – CONSTITUTION OF INDIA

Pre-requisites: Nil

Course Educational Objectives
• To enable the student to understand the importance of constitution.
• To understand the structure of Executive, Legislature and Judiciary.
• To understand Philosophy of fundamental rights and duties.
• To understand the autonomous nature of constitution bodies like Supreme Court and High Court Controller and Auditor General of India and Election Commission of India.
• To understand the Central and State relation, financial and administrative.

Course Outcomes: At the end of the course, the student shall be able to
CO1: Understand history and philosophy of constitution with reference to Preamble, Fundamental Rights and Duties (Understand – L2).
CO2: Understand the concept of Unitary and Federal Government along with the role of President, Prime Minister and Judicial System (Understand – L2).
CO3: Understand the structure of the state government, Secretariat, Governor and Chief Minister and their functions (Understand – L2).
CO4: Learn local administration viz. Panchayat, Block, Municipality and Corporation (Understand – L2).
CO5: Learn about Election Commission and the process and about SC, ST, OBC and women (Understand – L2).

UNIT – I:

UNIT – II:
Union Government and its Administration Structure of the Indian Union: Federalism Centre – State relationship, President: Role, Power and Position. Prime Minister (PM) and Council of Ministers, Cabinet and Central Secretariat, Lok Sabha, Rajya Sabha. The Supreme Court and High Court: Powers and Functions.

UNIT – III:
State Government and its Administration Governor – Role and Position – Chief Minister (CM) and Council of Ministers. State Secretariat: Organization, Structure and Functions.

UNIT – IV:
A Local Administration -- Role and Importance, Municipalities – Mayor and Role of Elected Representative, Panchayati Raj: Functions of Panchayati Raj Institution, Zilla Panchayat, Elected Officials and their roles, Village level – Role of Elected and Appointed officials.
UNIT – V:

Reference Books:
3. J. A. Siwach, Dynamics of Indian Government and Politics.

E-Resources:
1. nptel.ac.in/courses/109104074/8.
2. nptel.ac.in/courses/109104045.
3. nptel.ac.in/courses/101104065.


PRE-REQUISITES: PSPICE knowledge

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about the design of basic building blocks of analog VLSI chips, cascade amplifiers and operational trans-conductance amplifier using CMOS technology.

COURSE OUTCOMES:

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

CO1: Design basic building blocks of analog VLSI Chips
CO2: Design cascade amplifier using CMOS Technology
CO3: Design operational Trans-conductance amplifier (OTA) using CMOS Technology

Minimum of 10 experiments are to be conducted using Cadence/Mentor Graphics/Synopsys/Equivalent CAD Tools

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Trans conductance plots (voltage bias, current bias and technology bias).</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Design of basic amplifier.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>Design of cascode amplifier.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>Design of basic current sink.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.</td>
<td>Design current sink by using negative feed back resistor.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.</td>
<td>Design of cascode current sink.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.</td>
<td>Design of positive feed back boot strap current sink.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.</td>
<td>Design of regulated cascode current sink.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.</td>
<td>Design of cascode current mirror.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.</td>
<td>Design of wilson current mirror.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12.</td>
<td>Design of widlar current mirror.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
PRE-REQUISITES: Knowledge on Operating Systems

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about multitasking applications using RTOS, porting Linux on FPGA and create applications using ARM’s Real Time Operating system.

COURSE OUTCOMES:
At the end of the course, student will be able to
CO1: Develop multitasking applications using RTOS
CO2: Porting Linux on Zynq FPGA to develop embedded applications
CO3: Create applications using ARM’s Real Time Operating System RTX

Minimum of 10 experiments are to be conducted using Xilinx EDK/MicroVision IDE Keil

Write the following programs to understand the use of RTOS with ARM Processor on IDE Environment using ARM Tool chain and Library:

1. Create an application that creates two tasks that wait on a timer whilst the main task loops.
2. Write an application that creates a task which is scheduled when a button is pressed, which illustrates the use of an event set between an ISR and a task
3. Write an application that demonstrates the interruptible ISRs (Requires timer to have higher priority than external interrupt button)
4. a). Write an application to Test message queues and memory blocks. b). Write an application to Test byte queues
5. Write an application that creates two tasks of the same priority and sets the time slice period to illustrate time slicing.

Interfacing Programs:
6. Write an application that creates a two task to Blinking two different LEDs at different timings
7. Write an application that creates a two task displaying two different messages in LCD display in two lines.
8. Sending messages to mailbox by one task and reading the message from mailbox by another task.

9. Sending message to PC through serial port by three different tasks on priority Basis.

10. Porting Linux and developing simple application on Xilinx Zed board

11. Developing image processing application with Linux OS on Xilinx Zynq FPGA

12. Simulating a stepper-motor driver

13. Write simple applications using RTX (ARM Keil’s real time operating system, RTOS).
**PRE-REQUISITES:** Operating Systems & Linux

**COURSE EDUCATIONAL OBJECTIVES:**
In this course student will learn about embedded Linux, desktop Linux, embedded drivers for embedded applications, operating system porting, Linux real time programming.

**COURSE OUTCOMES:**
At the end of the course, student will be able to

- CO1: Compare Embedded Linux, desktop Linux and Embedded Linux distributions
- CO2: Develop board support packages for optimized embedded storage space
- CO3: Choose embedded drivers for typical embedded application
- CO4: Analyze application porting and operating system porting
- CO5: Describe Real-Time programming in linux /Hard Real-Time Linux

**UNIT – I:**
**Introduction:** History of Embedded Linux, Embedded Linux versus Desktop Linux, Embedded Linux Distributions, Architecture of Embedded Linux, Linux Kernel Architecture, Linux Start-Up Sequence, GNU Cross-p\Platform Tool chain.

**UNIT – II:**
**Board Support Package:** Inserting BSP in Kernel Build Procedure, Boot Loader Interface, Memory Map, Interrupt Management, PCI Subsystem, Timers, UART, and Power Management.

**Embedded Storage:** Flash Map, MTD—Memory Technology Device, MTD Architecture, Flash-Mapping Drivers, MTD Block and Character devices, Embedded File systems, Optimizing Storage Space.

**UNIT – III:**
**Embedded Drivers:** Linux Serial Driver, Ethernet Driver, I2C subsystem on Linux, USB Gadgets, Watchdog Timer, and Kernel Modules.

**UNIT-IV:**
**Porting Applications:** Architectural Comparison, Application Porting Road Map, Programming with Pthreads, Operating System Porting Layer (OSPL), Kernel API Driver.

**Unit-V:**
**Real-Time Linux:** Linux and Real-Time, Real-Time Programming in Linux, Hard Real-Time Linux.

**TEXT BOOK:**

**REFERENCE BOOK:**
1. Embedded Linux – Hardware, Software and Interfacing
PRE-REQUISITES: None

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about Micro-systems, Micro-fabrication. Mechanics for MEMS design, Electrostatic actuators and Electromagnetic actuators, various sensor structures.

COURSE OUTCOMES:
At the end of the course, student will be able to

CO1: Describe MEMS, Microsystems and Micro-fabrication
CO2: Analyze mechanics for MEMS Design
CO3: Distinguish Electrostatic actuators and Electromagnetic actuators
CO4: Model various circuits for MEMS Design
CO5: Choose Micro-bridge gas sensors, Piezo resistive Pressure Sensor and Bio-Chemical Sensors

UNIT – I:
INTRODUCTION TO MEMS

UNIT – II:
MECHANICS FOR MEMS DESIGN
Elasticity, stress, strain and material properties, Bending of thin plates, Spring configurations, torsional deflection, Mechanical vibration, Resonance - Thermo mechanics – Actuators, force and response time, Fracture and thin film mechanics, material, Physical Vapor Deposition (PVD), Chemical Mechanical Polishing (CMP).

UNIT – III:
ELECTROSTATIC DESIGN
Electrostatics:- basic theory, electro static instability, Surface tension, gap and finger pull up - Electro static actuators - Comb generators - Gap closers - Rotary motors - Inch worms - Electromagnetic actuators - Bistable actuators.

UNIT – IV:
CIRCUIT MODELING OF MEMS
UNIT – V:

CASE STUDIES

TEXT BOOK:

REFERENCE BOOKS:
PRE-REQUISITES: VLSI Design

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about concepts of power consumption, power analysis, low power design, and low power architectural level methodologies.

COURSE OUTCOMES:
At the end of the course, student will be able to
CO1: Understand the concepts of power consumptions
CO2: Analyze the power analysis
CO3: Create low power design circuits
CO4: Comprehend the low power systems
CO5: Evaluate the Architectural Level Methodologies

UNIT I:
Introduction: Need for low power VLSI chips, Sources of power dissipation on Digital Integrated circuits. Emerging Low power approaches, Physics of power dissipation in CMOS devices.

UNIT II:
Power estimation, Simulation Power analysis: SPICE circuit simulators, gate level logic simulation, capacitive power estimation, static state power, gate level capacitance estimation, architecture level analysis, data correlation analysis in DSP systems, Monte Carlo simulation.
Probabilistic power analysis: Random logic signals, probability & frequency, probabilistic power analysis techniques, signal entropy.

UNIT III:
Low Power Design Circuit level: Power consumption in circuits. Flip Flops & Latches design, high capacitance nodes, low power digital cells library.
Logic level: Gate reorganization, signal gating, logic encoding, state machine encoding, pre-computation logic.

UNIT IV:
Low power Architecture & Systems: Power & performance management, switching activity reduction, parallel architecture with voltage reduction, flow graph transformation, low power arithmetic components, low power memory design.
Low power Clock Distribution: Power dissipation in clock distribution, single driver Vs distributed buffers, Zero skew Vs tolerable skew, chip & package co design of clock network.
UNIT V:

TEXT BOOK:

REFERENCE BOOKS:
Pre-Requisites: Computer Organization & Architecture

COURSE EDUCATIONAL OBJECTIVES:
In this course student will learn about the classification of parallel architectures, mechanisms, general pipelining, vector processing and its architecture, multiprocessors and its interconnected networks, VLSI architectures and Data flow computers and their architectures.

COURSE OUTCOMES:
At the end of the course the student will be able to
CO1: Identify the differences between serial processing and parallel processing architectures and their applications.
CO2: Describe the functionality of pipelining and vector processor structures, hazards in various pipelining structures.
CO3: Develop the computer systems using SIMD processors with various interconnection networks.
CO4: Design the multi-processor systems with different memory configurations.
CO5: Analyze the working of Instruction flow computers and Data flow computers for VLSI computations.

UNIT – I: INTRODUCTION TO PARALLEL PROCESSING

UNIT – II: PRINCIPLES OF PIPELINING AND VECTOR PROCESSING
Principles of Linear Pipelining, General Pipelines and Reservation Tables, Interleaved Memory Organizations, Design of Pipelined Instruction Units, Instruction Prefetch and Branch Handling, Data Buffering and Busing Structures, Internal Forwarding and Register Tagging, Hazard Detection and Resolution, Job Sequencing and Collision Prevention, Characteristics of Vector Processing, Pipelined Vector Processing Methods.

UNIT – III: STRUCTURES AND ALGORITHMS FOR ARRAY PROCESSORS
SIMD Array Processors, SIMD Computer Organizations, Inter PE Communications, SIMD Interconnection Networks, Parallel Algorithms for Array Processors, SIMD Matrix Multiplication, Parallel Sorting on Array Processors, Associative Memory Organizations, Associative Processors.

UNIT – IV: MULTIPROCESSOR ARCHITECTURES
Functional Structures, Interconnection Networks, Parallel Memory Organizations, Interleaved Memory Configurations, Multi Cache Problems and Solutions, Multiprocessor Operating Systems, Exploiting Concurrency for Multiprocessing, Language Features to Exploit Parallelism, Detection of Parallelism in Programs.
UNIT – V: DATA FLOW COMPUTERS AND VLSI COMPUTATIONS

TEXT BOOK

REFERENCES
Pre-requisites: Problem solving skills

Course Objectives:
The objective of the course is to provide Basic knowledge of Python. Python programming is intended for software engineers, system analysts, program managers and user support personnel who wish to learn the Python programming language for Problem solving and programming capability.

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

<table>
<thead>
<tr>
<th>CO#</th>
<th>Course Outcome</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Identify the basic building blocks of python to solve mathematical problems</td>
</tr>
<tr>
<td>2</td>
<td>Apply the in-built data structures like list, tuple, set and dictionary for solving problems</td>
</tr>
<tr>
<td>3</td>
<td>Use exception handling mechanism to catch run-time errors</td>
</tr>
<tr>
<td>4</td>
<td>Demonstrate compelling concepts about string manipulation and regular expressions, File handling</td>
</tr>
<tr>
<td>5</td>
<td>Demonstrate object oriented programming principles of python</td>
</tr>
</tbody>
</table>

UNIT-I: Introduction to python, Operators, Control Structures

Introduction to Python: History of Python, Usage of Python interpreter, Python Shell, Indentation, Python Built-in types, Variables, Assignment, Input-Output Statements, Simple Programs, Identifiers and keywords, Literals. (Text Book-1,2)

Operators: Arithmetic operators, Relational operators, Logical operators, Assignment operators, Bitwise operators, Python Membership Operators (in & not in), Python Identity Operators (is & is not), Operator precedence. (Text Book-1,2)

Control Structures: Conditional Statements - if, if-else, Nested if-else. Jumping Statements - continue, break and pass. Python Loops - while, for, Nested loops with Programming Examples, Mathematical Functions and Constants (import math), Random Number Functions. (Text Book-1,2)

UNIT-II: Python Lists, Tuples, Set, Dictionaries

Python Lists: Concept, Creating and Accessing Elements, Updating & Deleting Lists, basic List Operations, Reverse, Indexing, Slicing and Matrices, Built-in List Functions. (Text Book-1,2)


Sets: Concept, Operations. (Text Book-1,2)

Dictionaries: Introduction, Accessing values in dictionaries, working with dictionaries, Properties, Functions. (Text Book-1,2)
UNIT-III: Functions, Modular Design, Exception Handling

**Functions:** Defining a Function, Calling a Function, Types of Functions, Function Arguments, Anonymous functions, Global and Local Variables, Recursion *(Text Book-1,2)*

**Modular Design:** Creating modules, import statement, from, Date and Time Module. *(Text Book-1,2)*

**Exception Handling:** Exception, Exception Handling, except clause, Try, finally clause, User Defined Exceptions. *(Text Book-1,2)*

UNIT-IV: Python Strings, Regular Expression Operations

**Python strings:** Concept, Slicing, Escape characters, String Special Operations, String formatting Operator, Triple Quotes, Raw String, Unicode Strings, and Built-in String methods. *(Text Book-1,2)*

**Regular Expression Operations:** Using Special Characters, Regular Expression Methods, Named Groups in Python Regular Expressions, Regular Expression with glob Module. *(Text Book-2)*

UNIT-V: Object Oriented Programming in Python

**Object Oriented Programming OOP in Python:** Classes, 'self-variable', Methods, Constructor Method, Inheritance, Overriding Methods, Data hiding. *(Text Book-1,2)*

**Text Books:**
- T2 Python for Everybody : Exploring Data in Python3 by Dr. Charles Russell Severance, Sue Blumenberg.

**Reference Books:**
- R3 Y.Daniel Liang, “Revel for Introduction to Python Programming and Data Structures”, Pearson publication.

**E-Books and Online Course Materials:**
1. www.python.org
2. www.geeksforgeeks.com

**Online Courses and Video Lectures:**
1. Durga Software Solutions. Online classes, video lectures.
2. The joy Of Computing using Python By NPTEL – [www.swayam.gov.in](http://www.swayam.gov.in)
3. [www.youtube.com/python/sundeepsaradhi](http://www.youtube.com/python/sundeepsaradhi)
**Pre-requisites:** Fundamentals of Data Mining

**Course Educational Objective:** This course will

- Develop an appreciation for what is involved in learning from data
- Demonstrate a wide variety of learning algorithms.
- Demonstrate how to apply a variety of learning algorithms to data.
- Demonstrate how to perform evaluation of learning algorithms and model selection.

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

**CO1:** Understand the basic concepts of learning and decision trees.

**CO2:** Able to solve real world problems using Neural Networks and Genetic Algorithms.

**CO3:** Demonstrate on Bayesian and Computational Learning.

**CO4:** Analyze different types of learning and learning set of rules such as case based reasoning and learning first order rules.

**CO5:** Summarize various concepts of analytical learning and reinforcement learning in terms of FOCL algorithm and Q learning.

**UNIT – I INTRODUCTION, CONCEPT LEARNING AND DECISION TREES**


**UNIT – II NEURAL NETWORKS AND GENETIC ALGORITHMS**


**UNIT – III BAYESIAN AND COMPUTATIONAL LEARNING**


**UNIT – IV INSTANT BASED LEARNING AND LEARNING SET OF RULES**


**UNIT – V ANALYTICAL LEARNING AND REINFORCED LEARNING**


**TEXTBOOKS**


