DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Scheme of Instruction

and

Syllabi of

M.Tech (Embedded Systems and Computing)

2017-2018

UNIVERSITY COLLEGE OF ENGINEERING

(AUTONOMOUS)

OSMANIA UNIVERSITY

HYDERABAD – 500 007, TELANGANA

SCHEME OF INSTRUCTION
M.TECH (EMBEDDED SYSTEMS AND COMPUTING)
Proposed from the Academic year 2017-18

SEMESTER - I

S.No	Course Code	Course Title	Scheme of Instruction		Contact Hrs/Wk	Scheme of Examination		Credits
			L/T	P		CIE	SEE	
1.	# Core	Core	3	--	3	30	70	3
2.	# Core	Core	3	--	3	30	70	3
3.	# Core/ *Elective	Core / Elective	3	--	3	30	70	3
4.	# Core/ *Elective	Core / Elective	3	--	3	30	70	3
5.	*Elective	Elective	3	--	3	30	70	3
6.	*Elective	Elective	3	--	3	30	70	3
Departmental Requirements								
7.	CS 5321	Software Lab - I	--	3	3	--	--	2
8.	CS 5322	Seminar - I	--	3	3	--	--	2
		Total	**18**	**6**	**24**	**280**	**420**	**22**

SEMESTER - II

S.No	Course Code	Course Title	Scheme of Instruction		Contact Hrs/Wk	Scheme of Examination		Credits
			L/T	P		CIE	SEE	
1.	# Core	Core	3	--	3	30	70	3
2.	# Core	Core	3	--	3	30	70	3
3.	# Core/ *Elective	Core / Elective	3	--	3	30	70	3
4.	# Core/ *Elective	Core / Elective	3	--	3	30	70	3
5.	*Elective	Elective	3	--	3	30	70	3
6.	*Elective	Elective	3	--	3	30	70	3
Departmental Requirements								
7.	CS 5323	Software Lab - II	--	3	3	--	--	2
8.	CS5324	Seminar - II	--	3	3	--	--	2
		Total	**18**	**6**	**24**	**280**	**420**	**22**

SCHEME OF INSTRUCTION
M.TECH (EMBEDDED SYSTEMS AND COMPUTING)
Proposed from the Academic year 2017-18

SEMESTER III

S.No	Course Code	Course Title	Scheme of Instruction		Contact Hrs/Wk	Scheme of Examination		Credits
			L/T	P		CIE	SEE	
1.	CS5325	Project Seminar	--	4	4	100**	--	8
		Total	**--**	**4**	**4**	**100**		**8**

***Project Seminar Evaluation:** 50 marks to be awarded by Supervisor and 50 marks to be awarded by Viva-Voce committee comprising Head, Supervisor and an Examiner.*

SEMESTER – IV

S. No	Course Code	Course Title	Scheme of Instruction		Contact Hrs/Wk	Scheme of Examination		Credits
			L/T	P		CIE	SEE	
1.	CS5326	Dissertation	--	6	6	---	200	16
		Total	**--**	**6**	**6**	**---**	**200**	**16**

Note: Six Core subjects, Six Elective subjects, Two Laboratory Courses and Two Seminars must be offered in Semester I and II.

 # List of Core Subjects:

S.No	Course Code	Course Title
1	CS 5301	Embedded System Design
2	CS 5302	Digital System Design
3	CS 5303	Microcontrollers for Embedded Systems
4	CS 5304	Real Time Operating Systems
5	CS 5305	Simulation and Modeling
6	CS 5306	Hardware and Software Co-design

***List of Elective Subjects:**

S.No	Course Code	Course Title
1	CS5054	Multimedia Technologies
2	CS 5056	Network Security
3	CS 5069	Human Computer Interaction
4	CS5101	Advanced Algorithms
5	CS 5251	Advanced Computer Networks
6	CS 5254	Wireless Sensor Networks
7	CS5351	Advanced Computer Architecture
8	CS5352	Scripting Languages for Design Automation
9	CS5353	Software Engineering for Real Time Systems
10	CS5354	Embedded Programming
11	CS5355	Field Programmable Gate Arrays
12	CS5356	System On Chip Architecture
13	CS5357	Optimization Techniques
14	CS558	Product Design and Quality Management
15	CS5359	Design for Testability

16	CS5360	DSP Architecture
17	CS5361	Graph Theory and its Applications
18	CS5362	Low Power VLSI Design
19	CS5363	Reliability and Fault Tolerance
20	CS5364	Performance Evaluation of Computer Systems

CS 5301 *With effect from the Academic year 2017-2018*

Embedded System Design

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Introduction to Embedded Systems, Characteristics and quality attributes of Embedded Systems, Challenges in Embedded System Design, Application and Domain specific Embedded Systems.

UNIT-II

Embedded Hardware Design and Development: VLSI and Integrated Circuit Design, EDA tools, usage of EDA tools and PCB layout.

Embedded Firmware Design and Development: Embedded firmware Design approaches and Development languages. Examples of Embedded Systems, Design Metrics in Embedded System

UNIT-III

Embedded System Architecture: CISC and RISC Instruction set architecture. Basic Embedded processor/Architecture. Introduction to ARM processors.

UNIT-IV

Introduction to Real Time Operating Systems**:** Tasks and Task States, Tasks and Data, Semaphores, and Shared Data; Message Queues, Mailboxes and Pipes, Timer Functions, Events, Memory Management, Interrupt Routines in an RTOS Environment. OS security issues and Mobile OS.

UNIT-V

Embedded Systems Development Environment: IDE, Cross Compilation, Dissassembler, Simulators, Emulators and Debugging. Target Hardware Debugging, Boundary Scan, Product Enclosure Design and Development Tools. Embedded Product Development Life Cycle-Different phases and Approaches of EDLC. Trends in Embedded Industry.

Suggested Reading:

1. Embedded Systems – Architecture, Programming and Design RajKamal, 2nd Edition, 2008, TMH.
2. Embedded System Design, Frank Vahid, Tony Givargis, John Wiley, 2002
3. Embedded /Real Time Systems: Concepts, Design and Programming- Dr K.V.K.K. Prasad, DreamTech Press 2005.

CS 5302 *With effect from the Academic year 2017-2018*

Digital System Design

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Analysis & Design of Combinational Logic: Introduction to combinational circuits, code conversions, decoder, encoder, priority encoder, multiplexers as function generators, binary adder, subtractor, BCD adder, Binary comparator, arithmetic logic units.

UNIT-II

Design of Sequential Circuits- Derivation of State diagrams and tables, transition table, excitation table and equations. Analysis of simple synchronous sequential circuits, construction of state diagram, counter design with state equations, Registers , serial in serial out shift registers, tristate register, timing considerations.

UNIT-III

Minimization and Transformation of Sequential Machines: Simplification of incompletely specified machines-Flow table, State reduction, Merger graphs, Merger Tables, Compatible pairs, Minimal closed covers – Races, Cycles and Hazards.

UNIT-IV

Design options of Digital Systems: Programmable logic devices, programmable read only memory, programmable logic arrays and programmable array logic, Design using PLA, PAL, and Field Programmable Gate Arrays. **Synthesis**: Design flow of ASICs and FPGA based system.

UNIT-V

Introduction to Verilog HDLs –Modeling levels- Data types- Modules and ports- Instances – Basic Language concepts- Data flow modeling. **Design Examples**: Adders and Subtractors. Behavioral modeling, Gate-level modeling. Tasks and functions – Modeling techniques – logic synthesis with Verilog. **Design Examples**: Multiplication and Division Algorithms.

Suggested Reading:

1. Ming-Bo Lin., *Digital System Designs and Practices using Verilog HDL and FPGAs*, Wiley, 2nd Edition, 2015.
2. Michael D. Ciletti, *Advanced Digital Design with the Verilog HDL*, Prentice Hall India, 2nd Edition, 2010.
3. Samir Palnitkar, *Verilog HDL: A Guide to Digital Design and Synthesis*, Prentice Hall Professional, 2003.

CS 5303 *With effect from the Academic year 2017-2018*

Microcontrollers for Embedded Systems

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT–I

Introduction to Microcontrollers, Embedded Systems, 8051-Architecture: instruction set, addressing modes and programming using 8051 microcontroller.
Memory Organization, Program Memory, Data Memory, Interrupts, Peripherals: Timers, Serial Port, I/O Port, Addressing Modes, Instruction Set, Programming

UNIT – II

Comparison of various families of 8-bit micro controllers, Interfacing of LCD, ADC, DAC, Sensors and Keyboard using Microcontrollers , USB and RS232.

UNIT – III

Introduction: RISC/ARM Design Philosophy and Functional Block Diagram. Programmers Model: Data Types, Processor modes, Registers, General Purpose Registers, Program Status Register, CP15 Coprocessor, Memory and memory mapped I/O, Pipeline, Exceptions, Interrupts and Vector table, and ARM Processor Families.

UNIT – IV

ARM9 Microcontroller Architecture: Block Diagram, Features, Memory Mapping, Memory Controller (MC), External Bus Interface (EBI), Connections to Memory Devices System Timer (ST): Period Interval Timer (PIT), Watchdog Timer (WDT), Real- time Timer (RTT), Real Time Clock (RTC), and Parallel Input/Output Controller (PIO).

UNIT- V

Universal Synchronous Asynchronous Receiver Transceiver (USART): Block Diagram, Functional Description, Synchronous and Asynchronous Modes. Development & Debugging Tools for Microcontroller based Embedded Systems: Software and Hardware tools like Cross Assembler, Compiler, Debugger, and Simulator.

Suggested Reading:

1. Muhammad Ali Mazidi, Janice Gillispie Mazidi, Rolin D. McKinlay, "The 8051 Microcontroller and Embedded Systems using Assembly and C", 2nd Edition, Prentice Hall
2. David Seal "ARM Architecture Reference Manual", 2001 Addison Wesley, England, Morgan Kaufmann Publishers.

CS 5304 *With effect from the Academic year 2017-2018*

Real Time Operating Systems

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT I

Brief Review of Unix Operating Systems (Unix Kernel – File system, Concepts of – Process, Concurrent Execution & Interrupts. Process Management – forks & execution. Programming with system calls, Process Scheduling. Shell programming and filters).

Portable Operating System Interface (POSIX) – IEEE Standard 1003.13 & POSIX real time profile. POSIX versus traditional Unix signals, overheads and timing predictability.

UNIT II

Hard versus Soft Real-time systems – examples, Jobs & Processors, Hard and Soft timing constraints, Hard Real-time systems, Soft Real-time systems. Classical Uniprocessor Scheduling Algorithms – RMS, Preemptive EDF, Allowing for Preemptive and Exclusion Condition.

UNIT III

Concept of Embedded Operating Systems, Differences between Traditional OS and RTOS. Real-time System Concepts, RTOS Kernel & Issues in Multitasking – Task Assignment, Task Priorities, Scheduling, Intertask Communication & Synchronization – Definition of Context Switching, Foreground ISRs and Background Tasks. Critical Section – Reentrant Functions, Interprocess Communication (IPC) – IPC through Semaphores, Mutex, Mailboxes, Message Queues or Pipes and Event Flags.

UNIT IV

VxWorks – POSIX Real Time Extensions, timeout features, Task Creation, Semaphores (Binary, Counting), Mutex, Mailbox, Message Queues, Memory Management – Virtual to Physical Address Mapping.

UNIT V

Debugging Tools and Cross Development Environment – Software Logic Analyzers, ICEs.

Comparison of RTOS – VxWorks, µC/OS-II and RT Linux for Embedded Applications.

Suggested Reading:

1. Real Time Systems, Jane W.S.Liu, Pearson Education, Asia, 2001.
2. Betcnhof, D.R., Programming with POSIX threads, Addison - Wesley Longman, 1997.
3. Wind River Systems, VxWorks Programmers Guide, Wind River Systems Inc.1997.
4. Jean.J.Labrosse, MicroC/OS-II, The CMP Books.
5. Real Time Systems, C.M.Krishna and G.Shin, McGraw-Hill Companies Inc., McGraw Hill International Editions, 1997.

CS 5305 *With effect from the Academic year 2017-2018*

Simulation and Modelling

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Introduction to Simulation: Advantages and Disadvantages of simulation, Areas of applications, Systems and System Environment, Concept of a System, Discrete and continuous systems, Models, Types of models, Steps in a simulation study – examples, Discrete – event system simulation.

UNIT-II

Overview of Statistical Models and Queuing Systems Programming languages for Simulation: Continuous and discrete Simulation Languages – GPSS, SIMAN, SIMSCRIPT, MATLAB and SIMULINK

UNIT-III

Random Numbers: Generation, Properties of Random Numbers, Generation of Pseudo Random Numbers, Tests for random Numbers.
Random Variate: Generation, Inverse Transformation Technique, Uniform Distribution, Exponential Distribution, Weibul's Distributio, Triangular Distribution, Empirical Continuous Distributions, Discrete Distributions, Direct Transformation for the Normal Distribution, Convolution Method of Erlang distribution, Acceptance Rejection Techniques: Poisson Distribution, Gamma Distribution.

UNIT-IV

Input Data Analysis: Data Collection: Identify the distribution, Parameter & Estimation.
Goodness of fit tests: Chi Square Test – KS test; Multivariate and rime series input models, Verification and Validations of simulation models, Model building, Verification and Validation: Verification of Simulation Models: Calibration and Validation of Models Face Validity, Validation of Model Assumptions, Validation Input/Output Transformations, Input/Output Validation using Historical input data, Input/Output validation sing Turning Test.

UNIT-V

Output Data Analysis, Stochastic, Nature of Output Data, Types of Simulation with respect to Output Analysis, Measures of Performance and their Estimation, Output Analysis for Terminating Simulations, Output analysis for Steady – State Simulations.
Comparison and Evaluation of Alternative System Designs: Comparison of several System Designs, Statistical Models for Estimating the Effect of Design Alternatives.

Suggested Reading:

1. Jabey Banks, John S. Cansen and Barry L. Nelson, *Discrete – Event System Simulation*, PHI, 2001.
2. Nursing Deo, *System Simulation with Digital Computer*, PHI, 1979.
3. Anerill M. Law and W. David Kelton, *Simulation Modelling and Analysis*, McGraw Hill, 2001.
4. Agam kumar tyagi, MATLAB and Simulink for Engineers, Oxford Publishers, 2011

CS 5306 *With effect from the Academic year 2017-2018*

Hardware and Software Co-design

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Nature of Hardware and Software: Introduction to Hardware/Software Codesign, Issues in co-designs, Driving factors in Hardware/Software Codesign,
Data Flow Modelling and Implementation: Need for Concurrent Models, Analyzing Synchronous Data Flow Graphs, Software and Hardware Implementation of Data Flow.
Analysis of Control and Data Flow-Implementing Data and Control Edges and Construction of Data Flow Graphs and Applications.

UNIT-II

Design Space of Custom Architectures: Finite State Machine with Data Path- Cycle based Bit-parallel Hardware, Hardware Modules, Finite Sate machines with data path, Simulation and RTL Synthesis of FSMD, Limitations of Finite State Machines.
Micro programmed Architectures: Microprogrammed Control, Encoding, Datapath. Implementing Microprogrammed Machine, Interpreters and Pipelining.

UNIT-III

General-Purpose Embedded Cores: Processors, RISC Pipeline, Program Organization and Analysis of quality of Compiled Code.
System On Chip: Concept and Design Principles in SoC Architectures.

UNIT-IV

Hardware/Software Interfaces: On-Chip Busses-Connecting Hardware and Software, OnChip Bus Systems, Bus Transfers, Multimaster Bus Systems, OnChip Networks.
Hardware/Software Interfaces: Synchronization Schemes, Memory-mapped Interfaces, Coprocessor Interfaces and Custom-Instruction Interfaces.

UNIT-V

Co Processor Control Shell Design: CoProcessor Control Shell, Data Design, Control Design, Programmers Model, AES encryption coprocessor.
Case Study: Trivium Crpto-Coprocessor and CORDIC Coprocessor**.**

Suggested Reading:

1. Schaumont, Patric R, *A Practical Introduction to Hardware/Software Codesign*, 2nd Edition, Springer publishers, 2013.
2 Jargen Staunstrup, Wayne Wolf, *Hardware/Software Co-Design, Principles and Practice,* Kluwer Academic Publishers, 1997.

CS 5321 *With effect from the Academic year 2017-2018*

SOFTWARE LAB-I

Credits: 2

Instruction: (3L) hrs per week
CIE: 50 marks

Documentation Using LATEX: Introduction to Linux Commands, Introduction to LateX, Creating & Editing Document, Formatting Document, Auto-text, Autocorrect, Spelling and Grammar tool, Page Formatting, Single/Multi column, Pictures/Objects, Drawing, Hyperlinks, Header/Footer, and Tables.

I. Implement the following using 8051, ARM7/ARM9 and Embedded C:

1) Experiments based on I/O Port, Timer, Serial & Interrupt Program using Keil (or equivalent) IDE.
2) Experiments on:

   i. Digital Interfaces
   ii. LCD Display interfaces
   iii. Analog Interfaces
   iv. Keyboard Interfaces
   v. PC –Interface with RS232, Ethernet etc.,
   vi. Stepper motor, traffic light controller, sensor devices etc.,

II. Verilog HDL modeling, Simulation, Synthesis, Timing Analysis and implementation on FPGA/CPLD target devices.

i. Combinational Circuits
ii. Sequential Circuits and FSMs
iii. Case study (Complete FPGA design flow including on-chip debugging)

Suggested Tools: Xilinx ISE/Altera Quartus, Modelsim/Active HDL and Target boards.

Note: The students have to submit a report using LateX at the end of the semester.

Suggested Reading:

1. Leslie Lamport**,** *Latex: A Document Preparation System,* 2nd Edition, Pearson Education India, 1994.

2. Stefan Kottwitz, *LaTeX Beginner's Guide***,** Shroff/Packt Publishers, First Edition, 2012.

Note: The students have to submit a report at the end of the semester.

SEMINAR - I

Credits: 2

Instruction: (3L) hrs per week
CIE: 50 marks

Oral presentation is an important aspect of engineering education. The objective of the seminar is to prepare the student for systematic independent study of state of the art topics in broad area of his/her specialization.

Seminar topics can be choosen by the students with the advice from the faculty members. Students are to be exposed to following aspects of seminar presentations.
Literature survey
Organization of material
Preparation of Power point Presentation slides
Technical writing

Each student is required to

1. Submit one page of synopsis of the seminar talk two days before for display on notice board.
2. Give 20 minutes presentation through MS-PowerPoint Presentation Slides followed by 10 minutes discussion.
3. Submit a report on the seminar topic with a list of references and slides used within a week.

Seminars are to be scheduled from the 3rd week of the last week of the semester and any change in schedule should be discouraged.
The CIE marks will be awarded to the students by atleast 2 faculty members on the basis of oral presentation and report as well as their involvement in the discussion.

SOFTWARE LAB – II

Credits: 2

Instruction: (3L) hrs per week
CIE: 50 marks

Programs based on usage of UNIX commands, System Calls, Shell programming.

1. Program that makes a copy of a file using standard I/O and system calls.(using command line arguments)
2 Program to implement **'cat'** command using system calls

3 Program to implement **'ls'** command using system calls

4 Program that takes one or more file/directory names as command line input and reports the following information on the file.
   A. File type.
   B. Number of links.
   C. Time of last access.
   D. Read, Write and Execute permissions.
5. Program to create a child process and allow the parent to display "parent" and the child to display child" on the screen.
6. Write a program to create a Zombie process.
7. Write a program that illustrates how an orphan is created.
8. Write a program that illustrates how to execute two commands concurrently with a command pipe.
9. Write a program that illustrates suspending and resuming processes using signals.
10 Write a program to implement IPC using unnamed pipes (anonymous pipe)
11 Write a program to implement half duplex communication between two unrelated processes using named pipe(FIFO)
12 Write programs to demonstrate message queue IPC
13 Write a program that illustrates two processes communicating using shared memory.
14 Write a Program that demonstrate semaphores

15. Write a C program to demonstrate multi threading

Note: The students have to submit a report at the end of the semester.

SEMINAR –II

Credits: 2

Instruction: (3L) hrs per week
CIE: 50 marks

Oral presentation is an important aspects of engineering education . The objective of the seminar is to prepare the student for systematic independent study of state of the art topics in broad are his/her specialization.

Seminar topics can be chosen by the students with the advice from the faculty members.

Students are to be exposed to following aspects of seminar presentation.

Literature Survey

Organization of material

Preparation of Power point Presentation slides and Technical Writing.

Each Student is required to:

1. Submit one page of synopsis of the seminar talk two days before for display on notice board.
2. Give 20 minutes presentation through MS-Power Point presentation slides followed by 10 minutes discussion.
3. Submit a report on the seminar topic with a list of references and slides used within a week

Seminar are to be scheduled from the 3rd week to the last week of the semester and any change in schedule should be discouraged.

The CIE marks will be awarded to the students by atleast 2 faculty members on the basis of oral and a written presentation as well as their involvement in the discussion.

CS 5054 *With effect from the Academic year 2017-2018*

Multimedia Technologies

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Media and Data Streams: Properties of multimedia systems, Data streams characteristics: Digital representation of audio, numeric instruments digital interface Bark concepts, Devices, Messages, Timing Standards Speech generation, analysis and transmission.

UNIT-II

Digital Image: Analysis, recognition, transmission, **Video**: Representation, Digitalization transmission **Animations**: Basic concepts, animation languages, animations control transmission

UNIT-III

Data Compression Standards: JPEG, H-261, MPEG DVI

Optical storage devices and Standards: WORHS, CDDA, CDROM, CDWO, CDMO.

Real Time Multimedia, Multimedia file System.

UNIT-IV

Multimedia Communication System: Collaborative computing session management, transport subsystem, QOS, resource management.

Multimedia Databases: Characteristics, data structures, operation, integration in a database model. **A Synchronization**: Issues, presentation requirements, reference to multimedia synchronization, MHEG

UNIT-V

Multimedia Application: Media preparation, Composition, integration communication, consumption, entertainment.

Suggested Reading:

1. Ralf Steninmetz, Klara Hahrstedt, *Multimedia: Computing, Communication and Applications*, PHI PTR Innovative Technology Series.
2. John F.Koegel Bufford, *Multimedia System*, Addison Wesley, 1994.
3. Mark Elsom Cook, *Principles of Interactive Multimedia* , Tata Mc-Graw Hill, 2001.
4. Judith Jefcoate, *Multimedia in Practice: Technology and Application* , PHI 1998.

CS 5056 *With effect from the Academic year 2017-2018*

NETWORK SECURITY

Credits: 3

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Introduction: Attributes of Security, Integrity, Authenticity, Non-repudiation, Confidentiality Authorization, Anonymity, Types of Attacks, DoS, IP Spoofing, Replay, Man-in-the-Middle attacks General Threats to Computer Network, Worms, Viruses, -Trojans

UNIT-II

Secret Key Cryptography : DES, Triple DES, AES, Key distribution, Attacks

Public Key Cryptography: RSA, ECC, Key Exchange (Diffie-Hellman), Java Cryptography Extensions, Attacks

UNIT-III

Integrity, Authentication and Non-Repudiation : Hash Function (MD5, SHA5), Message Authentication Code (MAC), Digital Signature (RSA, DSA Signatures), Biometric Authentication.

UNIT-IV

PKI Interface: Digital Certificates, Certifying Authorities, POP Key Interface, System Security using Firewalls and VPN's.

Smart Cards: Application Security using Smart Cards, Zero Knowledge Protocols and their use in Smart Cards, Attacks on Smart Cards

UNIT-V

Applications: Kerberos, Web Security Protocols (SSL), IPSec, Electronic Payments, E-cash, Secure Electronic Transaction (SET), Micro Payments, Case Studies of Enterprise Security (.NET and J2EE)

Suggested Reading:

1. William Stallings, *Cryptography and Network Security*, 4th Edition. Pearson,. 2009.

2. Behrouz A Forouzan, *Cryptography and Network Security*, TMH, 2009

3. Joseph Migga Kizza, *A Guide to Computer Network Security*, Springer, 2010

4. Dario Cataiano, *Contemporary Cryptology*, Springer, 2010.

CS 5069 *With effect from the Academic year 2017-2018*

Human Computer Interaction

Credits: 3

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT- I

Interaction Paradigms: Computing Environments, Analyzing Interaction Paradigms, Interaction Paradigms
Interaction Frameworks and Styles: Frameworks for Understanding Interaction, Coping with Complexity, Interaction Styles.

UNIT- II

Interaction Design Process: Iterative Design, User-Centered Design, Interaction Design Models, Overview of Interaction Design Models
Discovery: Discovery Phase Framework, Collection, Interpretation , Documentation
Design: Conceptual Design, Physical Design, Evaluation, Interface Design Standards, Designing the Facets of the Interface.

UNIT- III

Design Principles: Principles of Interaction Design, Comprehensibility, Learnability, Effectiveness/Usefulness, Efficiency/Usability, Grouping, Stimulus Intensity , Proportion , Screen Complexity, Resolution/Closure, Usability Goals
Interaction Design Models: Model Human Processor, Keyboard Level Model, GOMS, Modeling Structure, Modeling Dynamics, Physical Models
Usability Testing: Usability, Usability Test, Design the Test, Prepare for the Test, Perform the Test, Process the Data

UNIT- IV
Interface Components: The WIMP Interface, Other Components
Icons: Human Issues Concerning Icons, Using Icons in Interaction Design, Technical Issues Concerning Icons
Color: The Human Perceptual System, Using Color in Interaction Design, Color Concerns for Interaction Design, Technical Issues Concerning Color

UNIT- V

Text: Human Issues Concerning Text, Using Text in Interaction Design, Technical Issues Concerning Text
Speech and Hearing: The Human Perceptual System, Using Sound in Interaction Design, Technical Issues Concerning Sound
Touch and Movement: The Human Perceptual System, Using Haptics in Interaction Design, Technical Issues Concerning Haptics

Suggested Reading:

1. Steven Heim, *The Resonant Interface: HCI Foundations for Interaction Design*, Addison-Wesley, 2007
2. J. Preece, Y. Rogers, and H. Sharp, *Interaction Design: Beyond Human-Computer Interaction*, Wiley & Sons, 2nd Edition, 2007
3. Ben Shneiderman , Catherine Plaisant, *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, Addison-Wesley, 5th Edition, 2009.

CS 5101 *With effect from the Academic year 2017-2018*

Advanced Algorithms

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Algorithm Analysis: Asymptotic Notation, Amortization.

Basic Data Structures: Stacks and Queues, Vectors, Lists and Sequences, Trees, Priority Queues, Heaps, Dictionaries and Hash Tables.

Search Trees: Ordered Dictionaries and Binary Search Trees, AVL Trees, Bounded-Depth Search Trees, and Splay Trees.

UNIT-II

Fundamental Techniques: The Greedy Method, Divide-and-Conquer, and Dynamic Programming.

Graphs: The Graph Abstract Data Type, Data Structures for Graphs, Graph Traversal, Directed Graphs.

UNIT-III

Weighted Graphs: Single-Source Shortest Paths, All-Pairs Shortest Paths, Minimum Spanning Trees.

Network Flow and Matching: Flows and Cuts, Maximum Bipartite Matching, Minimum-Cost Flow.

UNIT-IV

Text Processing: Strings and Pattern Matching Algorithms, Tries, Text Compression, Text Similarity Testing.

Number Theory and Cryptography: Fundamental Algorithms involving numbers, Cryptographic Computations, Information Security Algorithms and Protocols.

UNIT-V

Computational Geometry: Range Trees, Priority Search Trees, Quad Trees and k-D Trees, Convex Hulls.

Suggested Reading:

1. M T Goodrich, R Tomassia. *Algorithm Design – Foundations, Analysis, and Internet Algorithms*, Wiley, 2006.
2. E Horowitz S Sahani, S Rajasekaran, C*omputer Algorithms*, Silicon Press, 2nd Edition, 2007.
3. Aho, A V Hopcraft, Ullman J D, *The Design and Analysis of Computer Algorithms*, Pearson Education, 2007.
4. Hari Mohan Pandey, *Design Analysis and Algorithms*, Firewall Media, 2008.
5. Cormen, Lieserson, Rivest, *Introduction to Algorithms*, MIT Press, 2nd Edition, 2009.

CS 5251 *With effect from the Academic year 2017-2018*

Advanced Computer Networks

Credits:3

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT I

History of Computer Networks and the Internet: Protocol Layers and Their Service Models Review of OSI and TCP/IP Delay, Loss, and Throughput in Packet-Switched Networks

UNIT II

Wireless and Mobile Networks: Introduction, Wireless Links and Network Characteristics, WiFi:802.11 Wireless LANs, Cellular Internet Access, Mobility Management: Principles Managing Mobility in Cellular Networks, Wireless and Mobility: Impact on Higher-layer Protocols, Bluetooth, Securing Wireless LANs

UNIT III

The Network Layer: Virtual Circuit and Datagram Networks, **The Internet Protocol** (IP): Forwarding and Addressing in the Internet Routing in the Internet Broadcast and Multicast Routing, Congestion Control QOS Label Switching and MPLS, Mobile IP, Voice over IP, IPv6 , **Network-LayerSecurity**:IPsec

UNIT IV

Transport Layer: Introduction and Transport-Layer Services, Multiplexing and Demultiplexing, Connectionless Transport: UDP, Principles of Reliable Data Transfer, Connection-Oriented Transport: TCP, Principles of Congestion Control, TCP Congestion Control, Securing TCP Connections: SSL,

Application Layer: Principles of Network Application, The Web and HTTP, HTTPS, File Transfer: FTP, Electronic Mail in the Internet, Securing E-mail

DNS—The Internet's Directory Service, Peer-to-Peer Applications

UNIT V

Network Management: The Infrastructure for Network Management, The Internet-Standard Management Framework, ASN.1, Multimedia Networking, Multimedia Networking Applications , Streaming Stored Audio and Video , Making the Best of the Best-Effort Service, Protocols for Real-Time Interactive Applications, Providing Multiple Classes of Service, Providing (QOS) Quality of Service Guarantees.

Suggested Reading:

1. E James, Keith W. Ross and F. Kurose, *Computer Networking: A Top-Down Approach*, Addison-Wesley, 4th Edition, 2008.
2. Andrew S Tanenbaum, *Computer Networks*, Prentice Hall PTR, 4th Edition, 2003

CS 5254 *With effect from the Academic year 2017-2018*

Wireless Sensor Networks

Credits:3

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UN1T-I

Mobile Ad-Hoc Networking with a View of 4G Wireless: Imperatives and Challenges, Off-the-Shelf Enables of Ad-Hoc Networks, IEEE 802.11 in Ad Hoc Networks : Protocols, Performance and Open Issues; Sctternet Formation in Bluetooth Networks, Antenna Beamforming and Power Control for Ad Hoc Networks.

UNIT-II

Topology Control in Wireless Ad Hoc Networks Broadcasting and Activity Scheduling in Ad Hoc Networks, Location Discovery, Mobile Ad Hoc Networks (MANETSs): Routing Technology for Dynamic, Wireless Networking, Routing Approaches in Mobile Ad Hoc Networks.

UNIT-III

Energy-Efficient Communication in Ad Hoc Wireless Networks, Ad Hoc Networks Security, Self-Organized and Cooperative Ad Hoc Networking, Simulation and Modeling of Wireless, Mobile, and Ad Hoc Networks, Modeling Cross-Layering Interaction Using Inverse Optimization, Algorithmic Challenges in Ad Hoc Networks.

UNIT-IV

Introduction and Overview of Wireless Sensor Networks: Application of Wireless Sensor Networks, Examples of Category 1 WSN Applications, Another Taxonomy of WSN Technology.

Basic Wireless Sensor Technology: Sensor Node Technology, Sensor Taxonomy, WN Operating Environment, WN Trends.

UNIT-V

Wireless Transmission Technology and Systems: Radio Technology Primer, Available Wireless Technologies. Medium Access Control Protocols for Wireless Sensor Networks: Fundamentals of MAC Protocols, MAC Protocols for WSNs, Sensor-MAC Case Study, IEEE 802.15.4 LR-WPANS Standard Case Study.

Suggested Reading:

1. Stefano Basagni, Silvia Giordano, Ivan Stojmenvic, *Mobile Ad Hoc Networking,* A John Wiley & Sons, Inc, Publication 2004.

2. Kazem Sohraby, Daniel Minoli, Taieb Znati, *Wireless Sensor Networks*, A John Wiley & Sons, Inc, Publication 2007.

CS 5351 *With effect from the Academic year 2017-2018*

Advanced Computer Architecture

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Measuring Performance and cost: Performance measurement, Enhancements to Uni processsor models, Benchmarks, Basic model of advanced computer architectures.

UNIT-II

Pipelining and superscalar techniques: Basic pipelining, data and control hazards, Dynamic instruction scheduling, Branch prediction techniques, Performance evaluation, case study- Sun Microsystems -Microprocessor.

UNIT-III

Vector Processors**:** Vector Processor Models**,** Vector architecture and Design, performance evaluation, Programming Vector processors.

UNIT-IV

Array Processors: parallel array processor model, memory organization, interconnection networks: performance measures, static and dynamic topologies.

UNIT-V

Multiprocessors and Multi computers: Multiprocessor models, Shared-memory and distributed memory architectures, memory organization, Cache Coherence and Synchronization Mechanisms, parallel computer, performance models.

Suggested Reading:

1. John L. Hennessey and David A. Patterson, *Computer Architecture, A Quantitative Approach*, Elsevier, 4th Edition, 2007.
2. Sajjan G. Shiva, *Advance Computer Architecture*, Taylor Series Group, CRC press, 2006.
3. Kai Hwang, *Advanced Computer Architecture*, Mc Graw Hill, 1999.

CS 5352 *With effect from the Academic year 2017-2018*

Scripting Languages for Design Automation

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT I

Introduction to Python Programming: Program Development Cycle, Input, Processing, and Output, Variables, Performing Calculations (Operators, Type conversions, Expressions),

Decision Structures and Boolean Logic: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables.

Repetition Structures: Introduction, while loop, for loop, Input Validation Loops, Nested Loops.

UNIT II

Functions: Introduction, Defining and Calling a Void Function, Designing a Program to Use Functions, Local Variables, Passing Arguments to Functions, Global Variables and Global Constants, Value-Returning Functions.

Lists and Tuples: Sequences, Introduction to Lists, List slicing, Finding Items in Lists with the in Operator, List Methods and Useful Built-in Functions, Copying Lists, Processing Lists, Two-Dimensional Lists, Tuples.

UNIT III

File and Exceptions: Introduction to File Input and Output, Using Loops to Process Files, Processing Records, Exceptions.

Strings: Basic String Operations, String Slicing, Testing, Searching, and Manipulating Strings

Dictionaries and Sets: Dictionaries, Sets, Serializing Objects.

UNIT IV

Recursion: Introduction, Problem Solving with Recursion, Examples of Recursive Algorithms.

Object-Oriented Programming: Procedural and Object-Oriented Programming, Classes, Working with Instances, Techniques for Designing Classes, Inheritance, Polymorphism.

UNIT V

GUI Programming: Graphical User Interfaces, Using the tkinter Module, Display text with Label Widgets, Organizing Widgets with Frames, Button Widgets and Info Dialog Boxes, Getting Input with Entry Widget, Using Labels as Output Fields, Radio Buttons, Check Buttons.

Suggested Reading:

1. Tony Gaddis, *Starting out With Python* , Pearson College Division, 3rd Edition, 2014.
2. John V Guttag, *Introduction to Computation and Programming using Python*, MIT Press, 3rd Edition, 2016.

CS 5353 *With effect from the Academic year 2017-2018*

Software Engineering for Real Time Systems

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Introduction: Review of Software Engineering Concepts, Characteristics of Real Time Systems, Importance of including Time Factor, The Real Time System Life Cycle: Requirement Specifications, State Charts.

UNIT-II

Structured Design Approaches: Event Based Model, Process-Based Structured Design, Graph-Based Theoretical Model, Petri Net Models: Stochastic Petri Net (SPN) Model Analysis, Annotated Petri Nets, Time-Augmented Petri Nets, Assessment of Petri Net Methods.

UNIT-III

Axiomatic Approaches: Weakest Precondition Analysis, Real Time Logic, Time Related History variables, State Machines and Real-Time Temporal Logic.

UNIT-IV

Language Support Restrictions: Real-Time Programming Descipline, Real-Time Programming Languages, Schedulability Analysis.

UNIT-V

Verification and Validation of Real-Time Software: Testing Real Time Properties, Simulation as Verification Tool, Testing Control and Data Flow, Proof Systems, Operational Approach.

Suggested Reading:

1. Shem – Tow Levi and Ashok K. Agarwal, *Real Time System Design*, McGraw Hill International Editions, 1999.
2. Cooling J.E. Jim Cooling, *Software Engineering for Real Time Systems*, Addison Wesly, 2002.

CS 5354 *With effect from the Academic year 2017-2018*

Embedded Programming

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT- I

Embedded OS Fundamentals (Linux): Introduction, Operating System fundamentals, General and Unix OS Architecture Embedded Linux. Booting Process in Linux GNU Tools: gcc, Conditional Compilation, Preprocessor directives, Command line arguments, Make files

UNIT-II

Embedded C Programming, Review of data types - Scalar types-Primitive types-Enumerated types- Subranges, Structure types-character strings -arrays- Functions Introduction to Embedded C-Introduction, Data types Bit manipulation, Interfacing C with Assembly. Embedded programming issues - Reentrancy, Portability, Optimizing, and testing embedded C programs.

UNIT-III

Embedded Applications using Data structures , Linear data structures- Stacks and Queues Implementation of stacks and Queues- Linked List - Implementation of linked list, Sorting, Searching, Insertion and Deletion, and non-linear structures.

UNIT-IV

Introduction to Object Oriented Concepts, Core Java/Java, Core- Java buzzwords, Overview of Java programming, Data types, variables and arrays, Operators, and Control statements.

UNIT –V

Embedded Java - Understanding J2ME, CDC (Connected Device configuration), CLDC (Connected Limited device configuration), MIDP applications.

Introduction to Programming and App Inventor: Introducing App Inventor, Getting Hands-On with App, Working with Media: Displaying Images, Duplicating Blocks and Using Dropdowns, Sounds, Color Blocks, Layout Components, Input, Variables, and Calculations: The Text Box Component, Performing Calculations, Storing Data with Variables , Creating Blocks with Type blocking, Math Functions. Decision Blocks and Boolean: Introduction to Operators, and Control statements.

Suggested Reading:

1. Jones, M Tim, *GNU/Linux Application Programming*, 2nd Edition, Course Technology PTR, 2008.

2. Prasad K.V.K.K, *Embedded /Real-Time Systems: Concepts, Design and Programming*, Dreamtech Press, 2003.

3. Sing Li and Jonathan Knudsen, *Beginning J2ME-From Novice to Professional*, Dreamtech Press, 3rd Edition, 2000.

4. Herbert Schildt, *The Complete Reference Java2*, Tata Mc GrawHill, 9th Edition, 2014.

5. Tony Gaddis, Rebecca Halsey, *Starting Out with App Inventor for Android (1e)*

CS 5355 *With effect from the Academic year 2017-2018*

Field programmable Gate Arrays

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Introduction to ASIC: Types of ASIC's, ASIC design flow, Economics of ASIC's, Programmable ASIC's: CPLD and FPGA. Commercially available CPLD's and FPGA's: XILINX, ALTERA, ACTEL. FPGA Design cycle, Implementation tools: Simulation and synthesis, and Programming technologies.

UNIT-II

FPGA logic cell for XILINX, ALTERA and ACTEL ACT, Technology trends, AC/DC IO Cells, clock and power inputs, FPGA interconnect: Routing resources, Elmore's constant, RC delay and parasitic capacitance FPGA design flow, and Low-level design entry.

UNIT-III

FPGA physical design, CAD tools, Power dissipation, FPGA Partitioning, Partitioning methods.Floor planning: Goals and objectives, I/O, Power and clock planning, and Floor Planning tools.

UNIT-IV

Placement: Goals and objectives, Placement algorithms: Min-cut based placement, Iterative Improvement, and simulated annealing.

Routing: Goals and objectives, Global routing methods, Back-annotation. Detailed Routing: Goals and objectives, Channel density, Segmented channel routing, Maze routing, Clock and power routing, Circuit extraction, and DRC.

UNIT-V

Verification and Testing: Verification, Logic simulation, Design validation, Timing verification. **Testing Concepts:** Failures, Mechanism and faults, Fault coverage, ATPG methods, Design for testability, Scan Path Design, Boundary Scan design, BIST Design guidelines, and Design of a Testing machine.

Suggested Reading:

1. Pak and Chan, Samiha Mourad, *Digital Design using Field Programmable Gate Arrays*, Pearson Education, 1st edition, 2009.
2. Michael John Sebastian Smith, *Application Specific Integrated Circuits*, Pearson Education Asia, 3rd edition 2001.
3. S. Brown, R.J.Francis, J.Rose, Z.G.Vranesic, *Field programmable Gate Array*, BSP, 2007.

CS 5356 *With effect from the Academic year 2017-2018*

System On Chip Architecture

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Introduction to Processor Design, Abstraction in Hardware Design, Processor design Tradeoffs, Design for Low power consumption, ARM processor as System-On-Chip: Acorn RISC Machine – Architecture inheritance – ARM programming model – ARM development tools – 3 and 5 stage pipeline ARM Organization – ARM instruction execution and implementation – ARM Co-processor interface.

UNIT-II

ARM Assembly Language programming: ARM instruction types – data transfer, data processing and control flow instructions.ARM instruction set- Co-processor instructions. Architectural support for High-level language.

UNIT-III

Memory Hierarchy: Memory size and speed – On-chip memory Caches –Cache design- an example- memory management.

UNIT-IV

Architectural Support for System Development: Advanced Microcontroller bus Architecture- ARM memory interface – ARM reference peripheral specification – Hardware system prototyping tools.

UNIT-V

Architectural Support for Operating System: An introduction to Operating Systems – ARM system control coprocessor- CP15 protection unit registers – ARM protection unit.- ARM MMU registers – ARM MMU Architecture – Synchronization – Context Switching input and Output.

Suggested Reading:

1. Steve Furber, *ARM System – on Chip Architecture*, Addison Wesley, 2nd Edition, 2000.
2. Ricardo Reis, *Design of System on a Chip: Devices and Component*, Springer publishers, 1st Edition, 2004.
3. Prakash Rashinkar, Peter Paterson and Leena Singh L, *System on Chip Verification Methodologies and Techniques*, Kluwer Academic Publishers, 2001

CS 5357 *With effect from the Academic year 2017-2018*

Optimization Techniques

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Use of optimization methods. Introduction to classical optimization techniques, motivation to the simplex method, simplex algorithm, sensitivity analysis.

UNIT-II

Search methods - Unrestricted search, exhaustive search, Fibonocci method, Golden section method, Direct search method, Random search methods, Univariate method, simplex method, Pattern search method.

UNIT-III

Descent methods, Gradient of function, steepest decent method, conjugate gradient method.

Characteristics of constrained problem, Direct methods, The complex method, cutting plane method.

UNIT-IV

Review of a global optimization techniques such as Monte Carlo method, Simulated annealing and Tunneling algorithm.

UNIT V

Generic algorithm - Selection process, Crossover, Mutation, Schema theorem, comparison between binary and floating point implementation.

Suggested Reading:

1. SS Rao, *Engineering Optimization: Theory and Practice*, John Wiley & Sons, 1996.
2. Zhigmiew Michelewicz, *Genetic Algorithms + Data Structures = Evaluation Programs*, Springer-Verlag Berlin Heidelberg, 1992.

CS 558 *With effect from the Academic year 2017-2018*

Product Design and Quality Management

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Product Design and Development, Development processes, Identifying customer needs, Establishing product specifications, Concept generation, Concept selection, Product architecture, and Industrial design.

UNIT- II

Product Design and Development Design for Manufacturing, Prototyping, Robust Design, Patents and Intellectual property, Product Development Economics, and Managing Product Development Projects.

UNIT-III

Total Quality Management Principles and Practices: Definition of quality, Customer satisfaction and Continuous improvement.

UNIT -IV

Total Quality Management, Tools and Techniques: Statistical Process Control, Quality Systems, Bench Marking.

UNIT -V

Total Quality Management , Quality Function Deployment, Product Liability, Failure Mode and Effect Analysis, Management Tools.

Suggested Reading:

1. T Dale H. Besterfield , *Total Quality Management*, Pearson Education Asia, Pearson Education India, 2nd Edition, 2011.
2. Karl T Ulrich & Steven D Eppinger, *Product Design & Development*; McGraw-Hill Education, 5th Edition, 2011.

CS 5359 *With effect from the Academic year 2017-2018*

Design for Testability

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Introduction to Testing: Testing Philosophy, Role of Testing, Digital and Analog VLSI Testing, VLSI Technology Trends affecting Testing, Types of Testing, Fault Modeling: Defects, Errors and Faults, Functional Versus Structural Testing, Levels of Fault Models, Single Stuck-at Fault.

UNIT –II

Logic and Fault Simulation: Simulation for Design Verification and Test Evaluation, Modeling Circuits for Simulation, Algorithms for True-value Simulation, Algorithms for Fault Simulation, ATPG.

UNIT –III

Testability Measures: SCOAP Controllability and Observability, High Level Testability Measures, Digital DFT and Scan Design: Ad-Hoc DFT Methods, Scan Design, Partial-Scan Design, Variations of Scan.

UNIT –IV

Built-In Self-Test: The Economic Case for BIST, Random Logic BIST: Definitions, BIST Process, Pattern Generation, Response Compaction, Built-In Logic Block Observers, Test-Per-Clock, Test-Per-Scan BIST Systems, Circular Self Test Path System, Memory BIST, Delay Fault BIST.

UNIT –V

Boundary Scan Standard: Motivation, System Configuration with Boundary Scan: TAP Controller and Port, Boundary Scan Test Instructions, Pin Constraints of the Standard, Boundary Scan Description Language: BDSL Description Components, Pin Descriptions.

Suggested Reading:

1. M. Abramovici, M.A.Breuer and A.D Friedman, *Digital Systems and Testable Design*, Wiley, 1994.
2. P.K. Lala, *Digital Circuits Testing and Testability*, Academic Press, 1997.

CS 5360 *With effect from the Academic year 2017-2018*

DSP Architecture

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT I

Introduction to DSP Processors: Differences between DSP and other μp architectures, their comparison and need for special ASPs , RISC & CISC CPUs .

UNIT II

Overview of DSP processor design: fixed point DSPs – Architecture of TMS 320C 5X, C54X Processors, addressing modes, Assembly instructions, Pipelining and on-chip peripherals.

UNIT III

Floating point DSPs: Architecture of TMS 320 – IX- Data formats, Floating Point operations, addressing modes, instructions, pipelining and peripherals.

UNIT IV

DSP interfacing & software development tools: I/O interfacing with A/D converters, PCs, Dual port RAMs, EPGAs, DSP tools – Assembler, debugger, c-compiler, linker, editor, code composer studio.

UNIT V

Applications using DSP adaptive filtering, spectrum analysis, Echo cancellation modems, voice synthesis and recognition. Brief ideas of AD, Motorola DSP CPUs and their comparison with TI CPUs.

Suggested Reading:

1. C. Marren & G. Ewess, *A Simple Approach to Digital Signal Processing*, WILEY Inter-science, 1996.
2. K. Shin, *DSP Applications with TMS 320 Family*, Prentice Hall, 1987.
3. B. Ventakaramani, M. Bhaskar, *Digital Signal Processes, Architecture Processing and Applications*, Tata Mc Graw Hill, 2002.

CS 5361 *With effect from the Academic year 2017-2018*

Graph Theory and its Applications

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT I

Preliminaries: Graphs, isomorphism, subgraphs, matrix representations, degree, operations on graphs, degree sequences

Connected graphs and shortest paths: Walks, trails, paths, connected graphs, distance, cut-vertices, cut-edges, blocks, connectivity, weighted graphs, shortest path algorithms

T**rees:** Characterizations, number of trees, minimum spanning trees.

UNIT II

Special classes of graphs: Bipartite graphs, line graphs, chordal graphs

Eulerian graphs: Characterization, Fleury's algorithm, chinese-postman-problem

UNIT III

Hamilton graphs: Necessary conditions and sufficient conditions

Independent sets, coverings, matchings: Basic equations, matchings in bipartite graphs, perfect matchings, greedy and approximation algorithms

UNIT IV

Vertex colorings: Chromatic number and cliques, greedy coloring algorithm, coloring of chordal graphs, Brook's theorem

Edge colorings: Gupta-Vizing theorem, Class-1 graphs and class-2 graphs, equitable edge-coloring**.**

UNIT V

Planar graphs: Basic concepts, Eulers formula, polyhedrons and planar graphs, charactrizations, planarity testing, 5-color-theorem

Directed graphs: Out-degree, in-degree, connectivity, orientation, Eulerian directed graphs, Hamilton directed graphs, tournaments

Suggested Reading:

1 F.Harry, Graph theory, Narosa Publications, 1988.

2.C.Berge: Graphs and Hypergraphs, North Holland/Elsevier, 1973

3. J A Bondy and U.S. R Murthy, *Graph Theory with Applications*, Elsevier Science Ltd, 1976.

4. Douglas B West, *Introduction to Graph Theory*, Prentice Hall, 2004

CS 5362 *With effect from the Academic year 2017-2018*

Low power VLSI Design

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

MOS Transistor – Nanometer Transistor and its model – body effect, Channel Length Modulation and short channel effects – velocity saturation, sub-threshold conduction, threshold voltage control, drain induced barrier lowering, gate induced drain leakage, Complete MOS Transistor Model and large and small signal models of BJTs and MOSFETs.

CMOS Inverter: Static and Dynamic behavior and Power, Energy and Energy-Delay analysis of CMOS Inverter

UNIT-II

Designing Combinational Logic Gates in CMOS: Introduction, Static CMOS Design, Dynamic CMOS Design.

Designing Sequential Logic circuits: Introduction, Static Latches and Registers, Dynamic Latches and Registers

UNIT-III

Power Estimation Techniques: Circuit Level – Modeling of Signals, Signal Probability Calculations, Statistical techniques; High Level Power Analysis – RTL Power Estimation, Fast Synthesis, Analytical Approaches, Architectural Power Estimation.

UNIT-IV

Power Optimization Techniques – I: Dynamic Power Reduction – Dynamic Power Component, Circuit Parallelization, Voltage Scaling Based Circuit Techniques, Circuit Technology – Independent Power Reduction, Circuit Technology Dependent Power Reduction; Leakage Power Reduction – Leakage Components, Design Time Reduction Techniques, Run-time Stand-by Reduction Techniques, Run-time Active Reduction Techniques Reduction in Cache Memories.

UNIT-V

Power Optimization Techniques – II: Low Power Very Fast Dynamic Logic Circuits, Low Power Arithmetic Operators, Energy Recovery Circuit Design, Adiabatic – Charging Principle and its implementation issues.

Suggested Reading:

1. Jan M Rabaey, A Chandrakasan, Borvioje N "Digital Integrated Circuits Design Perspective" PHI-2nd edition,2005.
2. Kaushik Roy and Sharat Prasad, *Low-Power CMOS VLSI Circuit Design*, Wiley Inter-science Publications, 2000.
3. Christian Piguet, *Low Power CMOS Circuits Technology, Logic Design and CAD Tools*, 1st Indian Reprint, CRC Press, 2010.

CS 5363 *With effect from the Academic year 2017-2018*

Reliability and Fault Tolerance

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Introduction to Reliability Engineering: Reliability, Repairable and Non-repairable Systems, Maintainability and Availability, Desighning, Reliability, Repairable and Non-repairable Systems, MTBF MTBF, MTTF MDT, k out of in systems.

UNIT-II

Software Reliability: Software Reliability, Software Reliability Vs Hardware Reliability, Failures and Faults, Classification of Failures, Counting, System configuration, Components and Operational Models, Concurrent Systems, Sequential Systems, Standby Redundant Systems.
Software Reliability Approaches: Fault Avoidance, Passive Fault Detection, Active Fault Detection, Fault Tolerance, Fault Recovery, Fault Treatment.

UNIT-III

Software Reliability Modeling: Introduction to Software Reliability Modeling, Parameter Determination and Estimation, Model Selection, Markovian Models, Finite and Infinite failure category Models, Comparison of Models, Calendar Time Modeling.

UNIT-IV

Fault Tolerant Computers: General Purpose Commercial Systems, Fault Tolerant Multiprocessor and VLSI based Communication Architecture. Design – N – Version programming Recovery Block, Acceptance Tests, Fault Trees, Validation of Fault Tolerant Systems.

UNIT-V

Fault Types: Fault Detection and Containment, Redundancy, Data Diversity, Reversal, Reversal Checks, Obtaining Parameter Values, Reliability Models for Hardware Redundancy, Software Error Models, Checks, Fault /Tolerant Synchronization, Synchronization in Software.

Suggested Reading:

1. John D. Musa, *Software Reliability*, McGraw Hill, 1995.
2. Patric D. T. O. Concor, *Practical Reliability Engineering*, John Wesly & Sons, 4th Edition, 2003.
3. C.M. Krishna, Kang G. Shin, *Real Time Systems*, McGraw Hill, 1997

CS 5364 *With effect from the Academic year 2017-2018*

Performance Evaluation of Computer Systems

Credits: 3:

Instruction : 3L hrs per week *Duration of SEE : 3 hours*

CIE : 30 Marks *SEE : 70 Marks*

UNIT-I

Fundamental Concepts and Performance Measures: Time, Events, Measurements, Intervals, Response, Independence, Randomness, Workloads, Problems Encountered in Model Development. A Case Study. General Measurement Principles, Scheduling Algorithms and Workloads.

UNIT-II

Probability: Random Variables, Jointly Distributed Random Variables, Probability Distributions, Densities, Expectation, Some Example Probability Distributions.
Stochastic Processes: Basic Definitions, Poisson Process, Birth-Death Process, Markov Process.

UNIT-III

Queuing Theory: Networks of Queues, Estimating Parameters and Distributions. Computational Methods for Queuing Network Solutions.

UNIT-IV

Petri Nets: Basic Notation, Classical Petri Nets, Times Petri Nets, priority-Based Petri Nets, Colored Petri Nets, Generalized Petri Nets.
Hardware Test beds, Instrumentation, Measurement, Data Extraction, and Analysis: Derivation of Performance Evaluation parameters, Network performance tests, General Methods of Data Extraction, Tested and Model Workloads, Experimental Design, Data Presentation.
System Performance Evaluation Tool Selection and Use: Validation of Results, Conducting Experiments, Performance Metrics, Evaluation.

UNIT-V

Analysis of Computer Architectures:
Case I: Central Server Computer System
Case II: Multiple Server Computer System
Case III: Petri Net Example
Analysis of Operating System Components: System Architectures, Workloads, Experimental Design and Simulation, Experimental Analysis and Conclusion.
Database Systems Performance Analysis: The Testbed Systems, The Database Systems Tested Performance Analysis Testing, The Results.
Analysis of Computer Network Components: Analytical Modeling Examples, Simulation Modeling of Local Area Networks.

Suggested Reading:

1. Paul J. Fortier and Howard E. Michel, *Computer System Performance Evaluation and Prediction*, Digital Press, 2003.
2. Raj Jain, *The art of Computer Systems Performance Analysis, techniques for experimental design, measurement and modelling*, Wiley Publishers, 2015.
3. Neil J. Gunther, *Analyzing Computer System Performance with PERL:: PDQ*, Springer-Verlag Berlin Heidelberg, 2011.