# M. TECH (REAL TIME SYSTEMS) –R13 Regulations

## JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

*(Established by an Act No.30 of 2008 of A.P. State Legislature)*

Kukatpally, Hyderabad – 500 085, Andhra Pradesh (India)

## M. TECH (REAL TIME SYSTEMS)

### COURSE STRUCTURE AND SYLLABUS

#### I Year - I Semester

<table>
<thead>
<tr>
<th>Code</th>
<th>Group</th>
<th>Subject</th>
<th>L</th>
<th>T/P/D</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Advanced Computer Architecture</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Micro Controllers</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Real Time Systems and Design</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Embedded C</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Elective - I</td>
<td>Digital Systems Design</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Elective - I</td>
<td>Distributed Operating Systems</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Elective - I</td>
<td>Cloud Computing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Elective - II</td>
<td>Fault Tolerant Systems</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Elective - II</td>
<td>Advanced Computer Networks</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Elective - II</td>
<td>Parallel Algorithms</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lab</td>
<td></td>
<td>Embedded C Lab</td>
<td>0</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Seminar</td>
<td></td>
<td></td>
<td>-</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>18</td>
<td>3</td>
<td>22</td>
</tr>
</tbody>
</table>

#### I Year - II Semester

<table>
<thead>
<tr>
<th>Code</th>
<th>Group</th>
<th>Subject</th>
<th>L</th>
<th>T/P/D</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Embedded Real Time Operating System</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Embedded Systems Development</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Embedded Linux Programming</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Scripting Languages</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Elective - I</td>
<td>Case Studies in Real Time Applications</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Elective - I</td>
<td>Low Power VLSI Design</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Elective - I</td>
<td>Mobile Computing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Elective - II</td>
<td>Quality Assurance in RTS</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Elective - II</td>
<td>Neural Networks</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Elective - II</td>
<td>Information Security</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lab</td>
<td></td>
<td>Embedded Systems Lab</td>
<td>0</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Seminar</td>
<td></td>
<td></td>
<td>-</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>18</td>
<td>3</td>
<td>22</td>
</tr>
</tbody>
</table>

#### II Year - I Semester

<table>
<thead>
<tr>
<th>Code</th>
<th>Group</th>
<th>Subject</th>
<th>L</th>
<th>T/P/D</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Comprehensive Viva</td>
<td>-</td>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project Seminar</td>
<td>0</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project Work</td>
<td>-</td>
<td></td>
<td>18</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>-</td>
<td>3</td>
<td>22</td>
</tr>
</tbody>
</table>

#### II Year - II Semester

<table>
<thead>
<tr>
<th>Code</th>
<th>Group</th>
<th>Subject</th>
<th>L</th>
<th>T/P/D</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Project Work and Seminar</td>
<td>-</td>
<td></td>
<td>22</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
<td>-</td>
<td></td>
<td>22</td>
</tr>
</tbody>
</table>
ADVANCED COMPUTER ARCHITECTURE

Objectives:
- To understand concepts of parallel processing and design choices of implementing parallel execution within a single processor (pipeline, VLIW, and superscalar) and multiprocessor systems.
- To gain knowledge of the state of the art research topics on advanced computing systems.

UNIT I
Concept of instruction format and instruction set of a computer, types of operands and operations; addressing modes; processor organization, register organization and stack organization; instruction cycle; basic details of Pentium processor and power PC processor, RISC and CISC instruction set.

UNIT II
Memory devices; Semiconductor and ferrite core memory, main memory, cache memory, associative memory organization; concept of virtual memory; memory organization and mapping; partitioning, demand paging, segmentation; magnetic disk organization, introduction to magnetic tape and CDROM.

UNIT III
IO Devices, Programmed IO, interrupt driver IO, DMA IO modules, IO addressing; IO channel, IO Processor, DOT matrix printer, ink jet printer, laser printer. Advanced concepts; Horizontal and vertical instruction format, microprogramming, microinstruction sequencing and control; instruction pipeline; parallel processing; problems in parallel processing; data hazard, control hazard.

UNIT IV
ILP software approach-complier techniques-static branch protection-VLIW approach-H.W support for more ILP at compile time-H.W verses S.W solutions
Multiprocessors and thread level parallelism-symmetric shared memory architectures-distributed shared memory-Synchronization-multi threading.

UNIT V
Storage System - Types-Buses-RAID-errors and failures-bench marking a storage device designing a I/O system.
Inter connection networks and clusters-interconnection network media – practical issues in interconnecting networks-examples-clusters-designing a cluster

TEXT BOOKS:

REFERENCE BOOKS:
MICRO CONTROLLERS

Objectives:
- Functions of a microcontroller.
- Essential peripherals for a microcontroller.
- Difference between a microprocessor (like 8085) and a microcontroller.
- The difference between Princeton and Harvard architectures.
- Variations in MCS-51 family of microcontrollers.
- Importance of power management in microcontrollers.
- Different types of packaging used for microcontrollers.

UNIT I
INTRODUCTION TO EMBEDDED SYSTEMS

UNIT II
MICRO CONTROLLER INTERFACING
8051, 68HC11, PIC-16C6X and ATMEL External Memory Interfacing – Memory Management Unit, Instruction and data cache, memory controller. On Chip Counters, Timers, Serial I/O Interrupts and their use. PWM /Watch dog, ISP, IAP features.

UNIT III
PROGRAMMING
Instruction sets and assembly language programme concepts and programming the 8051, 68HC11, PIC-16C6X Micro Controller ARM6TDMI Core (SOC) and PIC-IDE.

UNIT IV

UNIT V
Power Controls, External BUS Interface system Development and Debugging. CASE STUDIES: Design and Embedded Systems using the Micro Controller – 8051/ARM6TDMI, for application in the area of Communications, Automotives, industrial control.

TEXT BOOKS:
2. John B. Peatman, Designing with PIC Micro Controllers, Pearson Education.
4. Cathey May and Silha

REFERENCE BOOKS:
REAL TIME SYSTEMS AND DESIGN

Objectives:

- To explain the concept of a real-time system and why these systems are usually implemented as concurrent processes
- To describe a design process for real-time systems
- To explain the role of a real-time operating system
- To introduce generic process architectures for monitoring and control and data acquisition systems

UNIT I

Concepts of Computer control: Introduction, Sequence control, Loop control (direct digital control), Supervisory control, Centralized computer control, Hierarchical systems, Distributed systems, Human-Computer Interface (HCI), the control engineer, Economics and benefits of computer control systems, Summary.

UNIT II

DDC Algorithms and Their Implementation
Introduction, Implementation for the Basic PID algorithm, Synchronization of the control loop.

UNIT III
Bumpless transfer, Saturation and integral action wind-up, Tuning Choice of sampling interval, Plant input and output, improved forms of algorithm for integral and derivative calculation, Implementation of controller designs based on plant models, Summary

Design of Real-Time Systems – General Introduction
Introduction, Specification document, Preliminary design, Single-program Approach Foreground/Background system, Multi-tasking approach, Mutual exclusion, Monitors, Rendezvous, Summary.

UNIT IV
Real-Time System Development Methodologies -2: MASCOT, Basic features of MASCOT, General design approach, Textual representation of MASCOT designs, Other features of MASCOT, Development facilities, The MASCOT Kernel, Summary of MASCOT, Formal Methods, The PAISLey system for real-time software development method, PAISLey summary, Summary.

UNIT V

TEXT BOOK:
1. Real Time Computer control, Stuart Bennett, 2nd edition, Pearson Education

REFERENCE BOOK:
1. Real-Time Systems Design and Analysis by Phillip A Laplante
EMBEDDED C

Objectives:
- To develop 88051 microcontrollers in embedded systems
- To understand techniques of reading and writing bytes in embedded systems.
- To explain Hardware delays in various timers.

UNIT – I
Programming Embedded Systems in C
Introduction, What is an embedded system, Which processor should you use, Which programming language should you use, Which operating system should you use, How do you develop embedded software, Conclusions

Introducing the 8051 Microcontroller Family
Introduction, What’s in a name, The external interface of the Standard 8051, Reset requirements, Clock frequency and performance, Memory issues, I/O pins, Timers, Interrupts, Serial interface, Power consumption, Conclusions

UNIT – II
Reading Switches
Introduction, Basic techniques for reading from port pins, Example: Reading and writing bytes, Example: Reading and writing bits (simple version), Example: Reading and writing bits (generic version), The need for pull-up resistors, Dealing with switch bounce, Example: Reading switch inputs (basic code), Example: Counting goats, Conclusions

UNIT – III
Adding Structure to your Code
Introduction, Object-oriented programming with C, The Project Header (MAIN.H), The Port Header (PORT.H), Example: Restructuring the ‘Hello Embedded World’ example, Example: Restructuring the goat-counting example, Further examples, Conclusions

UNIT – IV
Meeting Real-Time Constraints
Introduction, Creating ‘hardware delays’ using Timer 0 and Timer 1, Example: Generating a precise 50 ms delay, Example: Creating a portable hardware delay, Why not use Timer 2?, The need for ‘timeout’ mechanisms, Creating loop timeouts, Example: Testing loop timeouts, Example: A more reliable switch interface, Creating hardware timeouts, Example: Testing a hardware timeout, Conclusions

UNIT – V
Case Study: Intruder Alarm System
Introduction, The software architecture, Key software components used in this example, running the program, the software, Conclusions

TEXT BOOK:
- Embedded C by Michael J. Pont, A Pearson Education

REFERENCE BOOK:
- PICmicro MCU C-An introduction to programming, The Microchip PIC in CCS C By Nigel Gardner
DIGITAL SYSTEMS DESIGN
(ELECTIVE – I)

Objectives:
- To design a small digital system for the specified functionality.
- To use all basic building blocks of digital systems.
- To understand timing and design situations.

UNIT I
DESIGN OF DIGITAL SYSTEMS: ASM charts, Hardware description language and control sequence method, Reduction of state tables, state assignments.
SEQUENTIAL CIRCUIT DESIGN: design of Iterative circuits, design of sequential circuits using ROMs and PLAs, sequential circuit design using CPLD, FPGAs.

UNIT II
FAULT MODELING: Fault classes and models – Stuck at faults, bridging faults, transition and intermittent faults.
TEST GENERATION: Fault diagnosis of Combinational circuits by conventional methods – Path Sensitization techniques, Boolean difference method, Kohavi algorithm.

UNIT III
TEST PATTERN GENERATION: D-Algorithm, PODEM, Random testing, Transition count testing, Signature analysis and testing for bridging faults.
FAULT DIAGNOSIS IN SEQUENTIAL CIRCUITS: State identification and fault detection experiment, Machine identification, Design of fault detection experiment.

UNIT IV
PROGRAMING LOGIC ARRAYS: Design using PLA's, PLA minimization and PLA folding.
PLA TESTING: Fault models, Test generation and Testable PLA design.

UNIT V
ASYNCHRONOUS SEQUENTIAL MACHINE: Fundamental mode model, flow table, state reduction, minimal closed covers, races, cycles and hazards.

TEXT BOOKS:
1. Z. Kohavi – “Switching & finite Automata Theory” (TMH)
2. N.N. Biswas – “Logic Design Theory (PHI)

REFERENCE BOOKS:
**DISTRIBUTED OPERATING SYSTEMS**

**ELECTIVE – I**

**Objectives:**
This course is designed to examine the fundamental principles of distributed systems, and provide students hands-on experience in developing distributed protocols. While we still look at issues in distributed operating systems, this course will address distributed systems in a broader sense. Emphasis will be placed on communication, process, naming, synchronization, consistency and replication, and fault tolerance.

**UNIT I**
Characteristics of Distributed Systems, Design Issues, User requirements, Network Technologies and Protocols, IPC, Client-server Communication, Group communications, IPC in UNIX, Remote Procedure calling Design issue, implementation, Asynchronous RPC

**UNIT II**
Distributed OS, its Kernel, Processor and Threads, Naming and Protection, Communication and Invocation, Virtual memory, File service components, Design issues, Interfaces, Implementation techniques, SUN Network File System.

**UNIT III**
SNS – a name service model, its design issues, Synchronizing physical clocks, Logical time and logical clocks, Distributed coordination, Replication and its architectural model, Consistency and request ordering, Conversation between a client and a server, Transactions, Nested Transactions, Concurrency control, Locks, Optimistic concurrency control, Timestamp ordering, Comparison of methods for concurrency control.

**UNIT IV**
Distributed Transactions and Nested Transactions Atomic commit protocols, Concurrency control in distributed transactions, distributed Deadlocks, Transactions with replicated data, Transaction recovery, Fault Tolerance, Hierarchical and group masking of faults

**UNIT V**
Cryptography, Authentication and Key distribution, Logics and Authentication, Digital Signatures, Distributed shared memory, Design and Implementation issues, sequential consistency and ivy, Release consistency and Munin, Overview of Distributed Operating systems Mach, Chorus.

**TEXT BOOK:**


**REFERENCE BOOK:**

Learning Objective:
- To explain the evolving computer model called cloud computing.
- To introduce the various levels of services that can be achieved by cloud.
- To describe the security aspects in cloud.

Outcomes:
- Ability to understand the virtualization and cloud computing concepts.

UNIT – I
Introduction to virtualization and virtual machine, Virtualization in cluster/grid context
Virtual network, Information model & data model for virtual machine, Software as a Service (SaaS),
SOA, On Demand Computing.

UNIT – II
Cloud computing: Introduction, What it is and What it isn’t, from Collaborations to Cloud, Cloud
application architectures, Value of cloud computing, Cloud Infrastructure models, Scaling a Cloud
Infrastructure, Capacity Planning, Cloud Scale.

UNIT – III
Data Center to Cloud: Move into the Cloud, Know Your Software Licenses, The Shift to a Cloud Cost
Model, Service Levels for Cloud Applications

UNIT – IV
Defining Clouds for the Enterprise- Storage-as-a-Service, Database-as-a-Service, Information-as-a-
Service, Process-as-a-Service, Application-as-a-Service, Platform-as-a-Service, Integration-as-a-
Infrastructure-as-a-Service

UNIT – V
Disaster Recovery, Disaster Recovery Planning, Cloud Disaster Management
Case study: Types of Clouds, Cloudcentres in detail, Comparing approaches, Xen OpenNEbula,
Eucalyptus, Amazon, Nimbus

TEXT BOOKS:
1. Cloud Computing – Web Based Applications That Change the way you Work and Collaborate
   Online – Michael Miller, Pearson Education.

REFERENCE BOOKS:
   David S. Linthicum Addison-Wesley Professional.
2. Enterprise Web 2.0 Fundamentals by Krishna Sankar; Susan A. Bouchard, Cisco Press
M. TECH (REAL TIME SYSTEMS) –R13 Regulations

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
M. Tech – I Year – I Sem. (RTS)

FAULT TOLERANT SYSTEMS
(ELECTIVE – II)

Objectives:

- To explain the relevance of the terms fault tolerance, reliability, and availability.
- To implementing methods for fault tolerance in an operating system.
- To explain how an operating system can continue functioning after a fault occurs.

UNIT I
Basic Concepts
Failure and Faults, reliability and failure rate, relation between eligibility and Mean-time between failures, maintainability and availability, reliability and series and parallel systems, Modeling of faults, stuck at, Bridging (short circuit), stuck open, transient and intermittent faults.
Fault diagnosis of digital systems, Test generation for combinational logic circuits conventional methods, Random testing, transition count testing and signature analysis

UNIT II
Basic concepts – static, dynamic, Hybrid, and self-purging redundancy, shift – over Modular Redundancy (SMR). Triplet Modular redundancy, SMR.
Reconfiguration, use of error correcting codes, Time redundancy, software redundancy, fail soft-operation examples of practical fault tolerant systems, Introduction to fault Tolerant Design of VLSI chips.

UNIT III
Design of Totally self-checking checkers, checkers using m-out of n codes, Berger codes and low cost residue code. Self-checking sequential Machines, Partially self checking circuits.

UNIT IV
Basic Concepts of test ability, controllability and observability. The read muller expansion technique, three level OR-AND-OR design, use of control logic and syndrome-testable design.

UNIT V
Design of Testable Sequential circuits the scan-path technique – level sensitive scan design (LSSD) and Random Access scan technique, built-in-test, built-in-test of VLSI chips, design for autonomous self-test, Designing Testability into logic Boards.

TEXT BOOK:

1. Fault Tolerant and Fault testable hardware design, Parag K.Lala PHI 1985

REFERENCES BOOKS:

1. Digital systems design using PLD’s LALA, PHI 1990
2. Logic Design theory, N.N. Biswas, PHI 1990
ADVANCED COMPUTER NETWORKS  
(ELECTIVE – II)

Objectives:

- The objective of this course is to build a solid foundation in computer networks concepts and design
- To understand computer network architectures, protocols, and interfaces.
- The OSI reference model and the Internet architecture. Network applications.
- The course will expose students to the concepts of traditional as well as modern day computer networks - wireless and mobile, multimedia-based.
- Students completing this course will understand the key concepts and practices employed in modern computer networking

Prerequisite: Computer Networks

Course description: This course will enable the student to refresh the fundamentals of Computer Networks in Unit I. Unit II describes the architecture, components, and operation of routers, and explains the principles of Routing and Routing protocols. Especially the Routing protocols need to be understood thoroughly with the help of any freely downloadable simulator tool. Through Unit III a student can learn the technologies and protocols needed to design and implement a converged switched network. This section explains how to configure a switch for basic functionality and how to implement Virtual LANs, VTP, and Inter-VLAN routing in a converged network. Students need to develop the necessary skills to implement a WLAN in a small-to-medium network. This course in Unit IV discusses the WAN technologies and network services required by converged applications in enterprise networks. Unit V makes the student to implement Networking using Java programs.


UNIT I: Review

UNIT II: Network Routing
Routing and its concepts: Structure of a Router, Basic Router Configuration, Building a Routing Table, Static Routing, Dynamic Routing – Distance Vector Routing Protocol (RIPv1, RIPv2, EIGRP), Link State Routing Protocols (OSPF).

UNIT III: LAN Switching
Switching and its concepts: Structure of a Switch, Basic Switch Configuration, Virtual LANs (VLANs), VLAN Trunking Protocol (VTP), Spanning Tree Protocol (STP), Inter-VLAN Routing.

UNIT IV: Wide Area Networks (WANs)
Introduction to WANs, Point-to-Point Protocol (PPP) concepts, Frame Relay concepts, Dynamic Host Configuration Protocol (DHCP), Network Address Translation (NAT), IPv6.

UNIT V: Network Programming using Java
TCP sockets, UDP sockets (datagram sockets), Server programs that can handle one connection at a time and multiple connections (using multithreaded server), Remote Method Invocation (Java RMI) - Basic RMI Process, Implementation details - Client-Server Application.

TEXT BOOKS:
2. Network Fundamentals, Mark Dye, Pearson Education.
4. LAN Switching & Wireless, Wayne Lewis, Pearson Education.
5. Accessing the WAN, Bob Vachon, Pearson Education.

REFERENCE BOOKS:

PARALLEL ALGORITHMS
(ELECTIVE –II)

Objectives:
- To understand the role of computation models in parallel computation;
- To understand the circuit and comparison network models;
- To design Parallel Matrix Transportation and Multiplication Algorithm;
- To understand the PRAM and BSP models and their theoretical foundations;

UNIT-I
Sequential model, need of alternative model, parallel computational models such as PRAM, LMCC, Hypercube, Cube Connected Cycle, Butterfly, Perfect Shuffle Computers, Tree model, Pyramid model, Fully Connected model, PRAM-CREW, EREW models, simulation of one model from another one.

UNIT-II
Performance Measures of Parallel Algorithms, speed-up and efficiency of PA, Cost optimality, Example to illustrate Cost-optimal algorithms such as summation, Min/Max on various models.

UNIT-III
Parallel Sorting Networks, Parallel Merging Algorithms on CREW/EREW/MCC/, Parallel Sorting Networks on CREW/EREW/MCC/, linear array

UNIT-IV
Parallel Searching Algorithm, Kth element, Kth element in X+Y on PRAM, Parallel Matrix Transportation and Multiplication Algorithm on PRAM, MCC, Vector-Matrix Multiplication, Solution of Linear Equation, Root finding.

UNIT-V
Graph Algorithms - Connected Graphs, search and traversal, Combinatorial Algorithms- Permutation, Combinations, Derrangements.

TEXT BOOK:

REFERENCE BOOK:
1. Introduction to Parallel algorithms by Jaja from Pearson, 1992.
EMBEDDED C LAB

List of Programs:

Write a simple program to print “hello word”
Write a simple program to show a delay.
Write a loop application to copy values from P1 to P2
Write a c program for counting the no of times that a switch is pressed & released.
Illustrate the use of port header file (port M) using an interface consisting of a keypad and liquid crystal display.
Write a program to create a portable hardware delay.
Write a c program to test loop time outs.
Write a c program to test hardware based timeout loops.
Develop a simple EOS showing traffic light sequencing.
Write a program to display elapsed time over RS-232 link.
Write a program to drive SEOS using Timer 0.
Develop software for milk pasteurization system.

Mini Project

Develop & implement a program for intruder alarm system.
EMBEDDED REAL TIME OPERATING SYSTEMS

Objectives:

- To provide a timely response to real world events. Events occurring in the real world can have deadlines before which the real time / embedded system must respond and the RTOS scheduling policy must ensure these deadlines are met.

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
Real Time Operating Systems

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
Exceptions, Interrupts and Timers
Exceptions, Interrupts, Applications, Processing of Exceptions and Spurious Interrupts, Real Time Clocks, Programmable Timers, Timer Interrupt Service Routines (ISR), Soft Timers, Operations.

UNIT - V
Case Studies of RTOS
RT Linux, MicroC/OS-II, Vx Works, Embedded Linux, and Tiny OS.

TEXT BOOK:


REFERENCE BOOKS:

EMBEDDED SYSTEMS DEVELOPMENT

Objectives:
- Design embedded computer system hardware
- Design, implement, and debug multi-threaded application software that operates under real-time constraints on embedded computer systems
- Use and describe the implementation of a real-time operating system on an embedded computer system
- Formulate an embedded computer system design problem including multiple constraints, create a design that satisfies the constraints.

UNIT I

UNIT II

UNIT III

UNIT IV

UNIT V
Introduction to advanced architecture: ARM and SHARC, processor and memory organization and Instruction level parallelism; Networked embedded systems; Bus protocols, I²C bus and CAN bus;

TEXT BOOKS:
1. Computers and Components, Wayne Wolf, Elsevier
3. An Embedded Software Primer by David E. Simon, Pearson Education
REFERENCE BOOKS:

1. Embedded Systems building blocks, Labrosse, via CMP publishers
2. Embedded Systems, Raj Kamal, TMH
3. Micro Controllers, Ajay V. Deshmukhi, TMH
4. Embedded system Design, Frank Vahid, Tony Givargis, John Wiley
5. Microcontrollers, Rajkamal, Pearson Education.
EMBEDDED LINUX PROGRAMMING

Objectives:
- To explain various embedded system applications and design requirements.
- To construct embedded system hardware.
- To develop software programs to control embedded system.
- To generate product specification for embedded system.

UNIT-I

UNIT-II
Introducing gdb, Local Debugging, Remote Debugging, Network-Mounting the root, File system, Configuring the NFS Server Configuring the Target Kernels Debugging Techniques: Design support in the Kernel, debugging by printing, querying, watching, system faults.

UNIT-III
Introduction to the Kernel – Important Data Structures, Main Algorithms, Implementation of system calls.
Memory Management: Architecture Independent memory models, Virtual address space of a process, block device caching, Paging under Linux.
Allocating memory – Kmalloc, lookaside caches, get free page and friends, vmalloc and friends, per-CPU variables, obtaining large Buffers.

UNIT-IV
Concurrency and race Conditions: Pitfalls in scull, concurrency & its management, semaphores and mutexes, completions, spin locks, loading traps, alternatives to Locking.
Time, Delays, Deferred Work: Measuring time lapses, Knowing current time, delaying execution, kernel timers, tasklets, workqueues.

UNIT-V
Interrupt handling: Preparing the parallel port, installing an Interrupt handler, implementing a handler, Top and bottom Halves, Interrupt Sharing, interrupt driven I/O.
Communicating with H/W: I/O ports and I/O Memory, Using I/O ports, An I/O port example, using I/O memory, Data types in Kernel: Uses of structured C types, assigning an explicit size to data items, interface specific types, other portability issues, linked lists.

TEXT BOOKS:
1. Embedded Linux: Hardware, Software and Interfacing, Dr. Craig Hollabaugh
2. Building Embedded Linux Systems By Karim Yaghmour Publisher: O’Reilly Media

REFERENCE BOOKS:
4. An Embedded Software Primer, David E. Simon, Pearson Education.
5. Micro Controllers, Ajay V Deshmukhi, TMH.
7. Introduction to Embedded Systems,Shibu K.V,TMH.
JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
M. Tech – I Year – II Sem. (RTS)

SCRIPTING LANGUAGES

Learning Objective:
The course demonstrates an in depth understanding of the tools and the scripting languages necessary for design and development of applications dealing with Bio-information/ Bio-data. The instructor is advised to discuss examples in the context of Bio-data/ Bio-information application development.

UNIT I
Introduction to PERL and Scripting
Scripts and Programs, Origin of Scripting , Scripting Today, Characteristics of Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines, advance perl - finer points of looping, pack and unpack, file system, eval, data structures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.

UNIT II
PHP Basics

UNIT III
Advanced PHP Programming
Php and Web Forms, Files, PHP Authentication and Methodologies -Hard Coded, File Based, Database Based, IP Based, Login Administration, Uploading Files with PHP, Sending Email using PHP, PHP Encryption Functions, the Mcrypt package, Building Web sites for the World – Translating Websites- Updating Web sites Scripts, Creating the Localization Repository, Translating Files, text, Generate Binary Files, Set the desired language within your scripts, Localizing Dates, Numbers and Times.

UNIT IV
TCL – Tk
TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures , strings , patterns, files, Advance TCL- eval, source, exec and uplevel commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, C Interface. Tk-Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding , Perl-Tk.

UNIT V
Python

TEXT BOOKS:

1. The World of Scripting Languages, David Barron, Wiley Publications.
REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux, Apache, MySQL, Perl and PHP, J. Lee and B. Ware (Addison Wesley) Pearson Education.
2. Programming Python, M. Lutz, SPD.
4. PHP 5.1, I. Bayross and S. Shah, The X Team, SPD.
5. Core Python Programming, Chun, Pearson Education.
7. Perl by Example, E. Quigley, Pearson Education.
8. Programming Perl, Larry Wall, T. Christiansen and J. Orwant, O'Reilly, SPD.
9. Tcl and the Tk Tool Kit, Ousterhout, Pearson Education.
10. PHP and MySQL by Example, E. Quigley, Prentice Hall (Pearson).
12. PHP Programming Solutions, V. Vaswani, TMH.
CASE STUDIES IN REAL TIME APPLICATIONS
(ELECTIVE-III)

Objectives:
- To understand the priority scheduling
- To learn about the specification and verification in timed CSP.
- To design the DC and PDC policies

UNIT I
Time and Real-time
Introduction, Real-time computing, Requirements, specification and implementation. The mine pump.
Fixed priority Scheduling – A Simple Model
Introduction, Computational model, Static Scheduling, Scheduling with priorities, Simple methods of analysis, Exact analysis, Extending the analysis

UNIT II
Advanced Fixed Priority Scheduling
Introduction, Computational model, advanced scheduling analysis, Introduction to Ada. The mini pump
Dynamic Priority Scheduling
Introduction, Programming dynamic real-time systems, Issues in dynamic scheduling Dynamic priority assignment, Dynamic best-effort approaches, Dynamic planning-based approaches, Practical considerations in dynamic scheduling

UNIT III
Assertional Specification and Verification
Introduction, Basic framework, The mine pump, Communication between parallel components, Parallel decomposition of the sump control, Programming language, The mine pump example: Final implementation, Further work, Historical background.

UNIT IV
Specification and Verification in Timed CSP
Introduction, the language of real-time CSP, Observations and processes, Specification, Verification, Case study: the mine pump, Historical background
Specification and Verification in DC
Introduction, Modeling real-time systems, Requirements, Assumptions, Design, the basic duration calculus (DC), the mine pump.

UNIT V
Specification of scheduling policies, Probabilities duration calculus (PDC), Historical background, Further work

TEXT BOOK:
1. Real-Time systems Specification, Verification and Analysis – Mathai Joseph
LOW POWER VLSI DESIGN
(ELECTIVE III)

Objectives:
- To study the concepts on different levels of power estimation and optimization Techniques.
- To design chips used for battery-powered systems and high-performance circuits not Exceeding power limits.

UNIT I
LOW POWER DESIGN, AN OVER VIEW:
Introduction to low-voltage low power design, limitations, Silicon-on-Insulator.
MOS/BICMOS PROCESSES:

UNIT II
LOW-VOLTAGE/LOW POWER CMOS/ BICMOS PROCESSES:
Deep submicron processes, SOI CMOS, lateral BJT on SOI, future trends and directions of CMOS/BICMOS processes.

UNIT III
DEVICE BEHAVIOR AND MODELING:
Advanced MOSFET model,s, limitations of MOSFET models, Bipolar models.
Analytical and experimental characterization of sub-half micron MOS devices, MOSFET in a Hybrid-mode environmental

UNIT IV
CMOS AND Bi-CMOS LOGIC GATES:
Conventional CMOS and BiCMOS logic gates. Performance evaluation

UNIT V
LOW POWER LATCHES AND FLIP FLOPS:
Evolution of Latches and Flip flops-quality measures for latches and Flip flops, Design perspective.

TEXT BOOK:
1. CMOS/BiCMOS ULSI low voltage, low power by yeo Rofail / Fohl (3 Authors)- Pearson Education Asia 1st Indian reprint, 2002

REFERENCE BOOKS:
2. CMOS Digital ICs sung-moKang and yusuf leblebici 3rd edition TMH 2003 (chapter 11)
3. VLSI DSP systems, Parhi, John Wiley & sons, 2003 (chapter 17)
Objectives:

- To make the student understand the concept of mobile computing paradigm, its novel applications and limitations.
- To understand the typical mobile networking infrastructure through a popular GSM protocol.
- To understand the issues and solutions of various layers of mobile networks, namely MAC layer, Network Layer & Transport Layer.
- To understand the database issues in mobile environments & data delivery models.
- To understand the ad hoc networks and related concepts.
- To understand the platforms and protocols used in mobile environment.

Outcomes:

- Able to think and develop new mobile application.
- Able to take any new technical issue related to this new paradigm and come up with a solution(s).
- Able to develop new ad hoc network applications and/or algorithms/protocols.
- Able to understand & develop any existing or new protocol related to mobile environment.

UNIT I
Introduction to Mobile Communications and Computing: Mobile Computing (MC):
Introduction to MC, novel applications, limitations, and architecture.
GSM: Mobile services, System architecture, Radio interface, Protocols, Localization and calling, handover, Security, and New data services.
(Wireless) medium Access Control: Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA

UNIT II
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)

UNIT III
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 IV
Database Issues: Hoarding techniques, Caching invalidation mechanisms, client server computing with adaptation, power-aware and context-aware computing, Transactional models, query processing, recovery and quality of service issues.
Data Dissemination: Communications asymmetry, classification of new data delivery mechanisms, push-based mechanisms, hybrid mechanisms, selective tuning (indexing) techniques.

UNIT V
Mobile Ad hoc Networks (MANETs): Overview, Properties of a MANET, spectrum of MANET applications, routing and various routing algorithms, security in MANETs.
Protocols and Tools: Wireless Application Protocol-WAP, (Introduction, protocol architecture, and treatment of protocols of all layers), Bluetooth 9User scenarios, physical layer, MAC layer, networking, security link management and J2ME.

TEXT BOOKS:
REFERENCE BOOKS:
QUALITY ASSURANCE IN RTS
(ELECTIVE IV)

Objectives:
- To understand the concepts of data and distribution.
- To develop the parallel systems and redundancy concepts.
- To understand different types of distribution.

UNIT I
Discrete random variables, Binomial distribution, Poisson distribution, Acceptance testing
Continuous random variable, Normal exponential, and Weibull distributions

UNIT II
Data and distributions, Goodness-of-fit
Reliability and rates of failure, Constant failure rate model, Time-dependent failure rates
Redundancy, Active parallel systems, High and low-level redundancy

UNIT III
Redundancy, Active parallel systems, High and low level redundancy

UNIT IV
Maintained Systems, Ideal and imperfect preventive maintenance, Availability, Maintainability
Failure Interactions, Markov analysis

UNIT V
Real-Time Systems and Fault-tolerance
Introduction, Assertions and correctness formulae, Formalizing a failure hypothesis, A proof rule for failure prone processes, Reliability of the mine pump, Soundness and completeness of the new proof rule, Historical background

TEXT BOOKS:
2. Real-Time Systems Specifications, Verification and Analysis - Mathai Joseph
NEURAL NETWORKS
(ELECTIVE IV)

Objectives:
- To survey of attractive applications of artificial neural networks.
- To practical approach for using artificial neural networks in various technical, organizational and economic applications.

UNIT I
INTRODUCTION - What is a neural network? Human Brain, Models of a Neuron, Neural networks viewed as Directed Graphs, Network Architectures, Knowledge Representation, Artificial Intelligence and Neural Networks (p. no’s 1 –49)
LEARNING PROCESS – Error Correction learning, Memory based learning, Hebbian learning

UNIT II
Competitive, Boltzmann learning, Credit Assignment Problem, Memory, Adaption, Statistical nature of the learning process, (p. no’s 50 –116)
SINGLE LAYER PERCEPTRONS – Adaptive filtering problem, Unconstrained Organization Techniques, Linear least square filters, least mean square algorithm, learning curves, Learning rate annealing techniques, perceptron –convergence theorem, Relation between perceptron and Bayes classifier for a Gaussian Environment (p. no’s 117 –155)

UNIT III
MULTILAYER PERCEPTRON – Back propagation algorithm XOR problem, Heuristics, Output representation and decision rule, Computer experiment, feature detection, (p. no’s 156 –201)
BACK PROPAGATION - back propagation and differentiation, Hessian matrix, Generalization, Cross validation, Network pruning Techniques, Virtues and limitations of back propagation learning, Accelerated convergence, supervised learning. (p. no’s 202 –234)

UNIT IV
SELF ORGANIZATION MAPS – Two basic feature mapping models, Self organization map, SOM algorithm, properties of feature map, computer simulations, learning vector quantization, Adaptive patter classification, Hierechel Vector quantilizer, contexmel Maps (p. no’s 443 –469, 9.1 –9.8 )

UNIT V
NEURO DYNAMICS – Dynamical systems, stability of equilibrium states, attractors, neurodynamical models, manipulation of attractors as a recurrent network paradigm (p. no’s 664 –680, 14.1 –14.6 )
HOPFIELD MODELS – Hopfield models, computer experiment I (p. no’s 680-701, 14.7 –14.8)

TEXT BOOK:

REFERENCE BOOKS:
2. Neural networks in Computer intelligence, Li Min Fu, TMH 2003
M. TECH (REAL TIME SYSTEMS) –R13 Regulations

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
M. Tech – I Year – II Sem. (RTS)

INFORMATION SECURITY
(ELECTIVE-IV)

Objectives:
- Explain the objectives of information security
- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand various cryptographic algorithms.
- Understand the basic categories of threats to computers and networks
- Describe public-key cryptosystem.
- Describe the enhancements made to IPv4 by IPSec
- Understand Intrusions and intrusion detection
- Discuss the fundamental ideas of public-key cryptography.
- Generate and distribute a PGP key pair and use the PGP package to send an encrypted e-mail message.
- Discuss Web security and Firewalls

UNIT – I

Cryptography: Concepts and Techniques: Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, steganography, key range and key size, possible types of attacks.

UNIT – II
Symmetric key Ciphers: Block Cipher principles & Algorithms (DES, AES, Blowfish), Differential and Linear Cryptanalysis, Block cipher modes of operation, Stream ciphers, RC4, Location and placement of encryption function, Key distribution

Asymmetric key Ciphers: Principles of public key cryptosystems, Algorithms (RSA, Diffie-Hellman, ECC), Key Distribution

UNIT – III

Authentication Applications: Kerberos, X.509 Authentication Service, Public – Key Infrastructure, Biometric Authentication

UNIT – IV
E-Mail Security: Pretty Good Privacy, S/MIME


UNIT – V

Intruders, Virus and Firewalls: Intruders, Intrusion detection, password management, Virus and related threats, Countermeasures, Firewall design principles, Types of firewalls

Case Studies on Cryptography and security: Secure Inter-branch Payment Transactions, Cross site Scripting Vulnerability, Virtual Elections

TEXT BOOKS:

REFERENCE BOOKS:

Information Security, Principles and Practice : Mark Stamp, Wiley India.
Introduction to Network Security: Neal Krawetz, CENGAGE Learning
Network Security and Cryptography: Bernard Menezes, CENGAGE Learning
EMBEDDED SYSTEMS LAB

Objectives:

- To implement the file command concepts
- To implement fork in embedded systems.
- To develop the system call in embedded system.

1. Write a 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

2. Write a C Program that illustrates how to execute two commands concurrently with a command pipe.

3. Write a C program that illustrates the creation of child process using fork, systems call.

4. Write a C program that displays the real time of a day every 60 seconds

5. Write a C program that implements a producer-consumer system with two processes (using semaphores)

6. Write a C program that illustrates inter process communication using shared memory system calls.

7. Write a C program that illustrates the following
   a. Creating a message queue
   b. Writing to a message queue
   c. Reading from a message queue

8. Programs in Window CE.
   a. Building application for writing and debugging application and target systems and using platforms builder.
   b. Building application for writing and debugging application and target system and using platforms builder.
   c. Building and application using ACTIVE X and COM
   d. Building and application using object store and database
   e. Building .NET applications

9. Real Time application using Linux