# ACADEMIC REGULATIONS
# COURSE STRUCTURE
# AND
# DETAILED SYLLABUS

# M.Tech.
# Real Time Systems

(Effective for the students admitted from the Academic Year 2007-08)

## Jawaharlal Nehru Technological University
## Hyderabad – 500 085

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

## COURSE STRUCTURE AND SYLLABUS

### I SEMESTER

| SUBJECT | T | P |
|---|---|---|
| Advanced Computer Architecture | 4 | - |
| Advanced Micro Controllers | 4 | - |
| Fundamentals of Real Time Systems | 4 | - |
| Design & Development of Real Time Systems | 4 | - |
| Elective – I | 4 | - |
| Elective – II | 4 | - |
| Micro Process and Programming Languages Lab | - | 3 |

### II SEMESTER

| SUBJECT | T | P |
|---|---|---|
| Real Time Systems Programming Languages | 4 | - |
| Embedded Systems Development | 4 | - |
| Real Time Operating Systems | 4 | - |
| Quality Assurance of Real Time Systems | 4 | - |
| Elective – III | 4 | - |
| Elective – IV | 4 | - |
| Embedded Systems Lab | - | 3 |

**Elective – I**

1 Digital Control Systems<br>
2 Distributed Operating Systems

**Elective – II**

1 Digital System Design<br>
2 Fault Tolerance Systems

**Elective – III**

1 Case studies in Real Time Operating Systems<br>
2 Low Power VLSI Design

**Elective – IV**

1 Case studies in Real Time Applications<br>
2 Mobile Computing

**III & IV SEMESTERS**

SEMINAR

PROJECT

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**I SEMESTER**

# Advanced Computer Architecture

**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.

**Unit – IV**

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 – V**

ILP software approach-compiler techniques-static branch protection-VLIW approach-H.W support for more ILP at compile time-H.W verses S. W solutions

**Unit – VI**

Multiprocessors and thread level parallelism-symmetric shared memory architectures-distributed shared memory-Synchronization-multi threading.

**Unit – VII**

Storage systems-Types-Buses-RAID-errors and failures-bench marking a storage device-designing a I/O system.

**Unit – VIII**

Inter connection networks and clusters-interconnection network media – practical issues in interconnecting networks-examples-clusters-designing a cluster.

**Text Books:**

1. "Computer organization and architecture", Williams Stallings, PHI of India, 1998.
2. Computer organization, Carl Hamachar, Zvonko Vranesic and Safwat Zaky, McGraw Hill International Edition.
3. Computer Architecture & Organization, John P. Hayes, TMH III Edition.
4. Computer Architecture A quantitative approach 3$^{rd}$ edition John L. Hennessy & David A. Patteson Morgan Kufmann(An Imprint of Elsevier)

**References:**

1. "Computer Architecture and parallel Processing" Kai Hwang and A. Briggs International Edition McGraw-Hill.
2. Advanced Computer Architecture, Dezso Sima, Terence Fountain, Peter Kacsuk, Pearson.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**I SEMESTER**

## Advanced Micro Controllers

**Unit - I**
INTRODUCTION TO EMBEDDED SYSTEMS
Review of Micro controllers and their Features. 8 & 16 Bit Micro Controller Families (of Intel 8051) Flash

**Unit - II**
Series, Motorola 68HC11; Micro Chip PIC 16C6X and Micro controller hardware. Embedded RISC Processor Architectures – ARM6TDMI(Advanced RISC Machines).

**Unit - III**
MICRO CONTROLLER INTERFACING
8051, 68HC11, PIC-16C6X and ATMEL External Memory Interfacing – Memory Management Unit, Instruction and data cache, memory controller.

**Unit - IV**
On Chip Counters, Timers, Serial I/O, Interrupts and their use. PWM, Watch dog, ISP, IAP features.

**Unit – V**
PROGRAMMING
Instruction sets and assembly language programme concepts and programming the 8051, 68HC11, PIC-
16C6X Micro controller ARM6TDMI Core (SOC) and PIC-IDE.

**Unit - VI**
Interrupt synchronization – Interrupt vectors & priority, external interrupt design. Serial I/O Devices – RS232 Specifications, RS422/Apple Talk/ RS 423/RS435 & other communication protocols. Serial communication controller.

**Unit - VII**
Ethernet Protocol, SDMA, Channels and IDMA Simulation, CPM Interrupt controller and CPM Timers,

**Unit - VIII**
Power controls, External BUS Interface system Development and Debugging.
CASE STUDIES: Design of Embedded Systems using the micro controller – 8051/ARM6TDMI, for applications in the area of Communications, Automotives, industrial control.

**Text Books**

1. M.A. Mazadi & J.G. Mazidi, "The 8051 Micro Controller & Embedded Systems", Pearson Education. Asia (2000).
2. John B. Peatman, Designing with PIC Micro Controllers, Pearson Education.
3. Jonathan W. Valvano, Embedded Microcomputer systems, Real Time Interfacing, Brookes/Cole,
   Thomas learning, 1999.
4. Cathey May and Silha
5. (Ed)., "The Power PC Architecture", Morgan Kauffman Press (1998).

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**I SEMESTER**

## Fundamentals of Real Time Systems

**Unit – I**

Introduction, A Car-and-Driver Example, Issues in Real-Time Computing, Structure of a Real-Time System, Task Classes, Issues Covered in this Book, Characterizing Real-Time Systems and Tasks, Performance Measures for Real-Time Systems, Estimating Program Run Times, Suggestions for Further Reading.

**Unit – II**

Task Assignment and Scheduling

Introduction, Classical Uniprocessor Scheduling Algorithms, Uniprocessor Scheduling of IRIS Tasks, Task Assignment, Mode Changes, Fault-Tolerant Scheduling, Suggestions for Further Reading.

**Unit – III**

Programming Languages and Tools

Introduction, Desired Language Characteristics, Data Typing, Control Structures, Facilitating Hierarchical Decomposition, Packages, Run-Time Error(Exception) Handling, Overloading and Generics, Multitasking, Low-Level Programming, Task Scheduling, Timing Specifications, Some Experimental Languages, Programming Environments, Run-Time support, Suggestion for further reading.

**Unit – IV**

Real-Time Databases

Introduction, Basic Definitions, Real-Time vs. General-Purpose Databases, Main Memory Databases, Transaction Priorities, Transaction Aborts, Concurrency Control Issues, Disk Scheduling algorithms, A two-phase approach to improve predictability, Maintaining Serialization Consistency, Databases for Hard Real-Time systems, Suggestion for further reading.

**Unit – V**

Real-Time Communication

Introduction, Network Topologies, Protocols, Suggestions for Further Reading

**Unit – VI**

Fault-Tolerance Techniques

Introduction, What Causes Failures?, Fault Types, Fault Detection, Fault and Error Containment, Redundancy, Data Diversity, Reversal Checks, Malicious or Byzantine Failures, Integrated Failure handling, Suggestions for further reading.

**Unit – VII**

Reliability Evaluation Techniques

Introduction, Obtaining Parameter Values, Reliability Models for Hardware Redundancy, Software-Error Models, Taking Time into account, Suggestions for further reading

**Unit – VIII**

Clock Synchronization

Introduction, Clocks, A Nonfault-Tolerant Synchronization algorithm, Impact of faults, fault-tolerant synchronization in hardware, Synchronization in software, Suggestions for further reading

**Text Books:**

Real Time Systems, Krishna C. M. & Kand Shin G. Mc Graw Hill, 1997, L. R. Rabiner & R. W. Schafe

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**I SEMESTER**

## Design & Development of Real Time Systems

**Unit - I**

Introduction to Real-Time Systems

Historical background, Elements of a computer control system, Real-time systems – definition, Classification of real-time systems, Time constraints, Classification of programs, Summary

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**

Computer Hardware Requirements for Real-time applications

Introduction

General purpose computer, Single-chip microcomputers and microcontrollers, Specialized Processors, Process-related Interfaces, Data transfer techniques, Communications, Standard Interfaces, Summary

**Unit - III**

DDC Algorithms and Their Implementation

Introduction, Implementation of the Basic PID algorithm, Synchronization of the control loop,

**Unit - IV**

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

**Unit - V**

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 - VI**

Real-Time System Development Methodologies – 1

Introduction, Yourdon methodology, Requirements definition for drying oven, Ward and Mellor method, Hatley and Pirbhai method, Comments and the Yourdon Methodologies, Summary

**Unit – VII**

Real-Time System Development Methodologies – 2

MASCOT, Basic features of MASCOT, General design approach, Textual representations 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 summery, Summary

**Unit – VIII**

Dependability, Fault detection and fault tolerance

Introduction, Use of Redundancy, Fault tolerance in Mixed Hardware-Software systems, Fault detection measures, Fault detection mechanisms, Damage containment and assessment, Provision of fault tolerance, summary

**Test Books:**

Real time computer control, Stuart Bennett, 2$^{nd}$ edition Pearson education

**Reference Books**

Real-Time Systems Design and Analysis by Phillip A. Laplante

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**I SEMESTER**

## Digital Control Systems

**Elective - I**

**Unit – I**

SAMPLING AND RECONSTRUCTION

Introduction, Examples of Data control systems – Digital to Analog conversion and Analog to Digital conversion, sample and hold operations.

**Unit-II**

THE Z – TRANSFORMS

Introduction, Linear difference equations, pulse response, Z – transforms, Theorems of Z – Transforms, the inverse Z – transforms, Modified Z- Transforms.

**Unit - III**

Z-PLANE ANALYSIS OF DISCRETE-TIME CONTROL SYSTEM

Z-Transform method for solving difference equations; Pulse transforms function, block diagram analysis of sampled – data systems, mapping between s-plane and z-plane.

**Unit – IV**

STATE SPACE ANALYSIS

State Space Representation of discrete time systems, Pulse Transfer Function Matrix solving discrete time state space equations, State transition matrix and it's Properties, Methods for Computation of State Transition Matrix, Discretization of continuous time state – space equations

**Unit – V**

CONTROLLABILITY AND OBSERVABILITY

Concepts of Controllability and Observability, Tests for controllability and Observability. Duality between Controllability and Observability, Controllability and Observability conditions for Pulse Transfer Function.

**Unit – VI**

STABILITY ANALYSIS

Mapping between the S-Plane and the Z-Plane – Primary strips and Complementary Strips – Constant frequency loci, Constant damping ratio loci, Stability Analysis of closed loop systems in the Z-Plane. Jury stablility test – Stability Analysis by use of the Bilinear Transformation and Routh Stability criterion.

**Unit – VII**

DESIGN OF DISCRETE TIME CONTROL SYSTEM BY CONVENTIONAL METHODS

Transient and steady – State response Analysis – Design based on the frequency response method – Bilinear Transformation and Design procedure in the w-plane, Lead, Lag and Lead-Lag compensators and digital PID controllers.

**Unit – VIII**

STATE FEEDBACK CONTROLLERS AND OBSERVERS

Design of state feedback controller through pole placement – Necessary and sufficient conditions, Ackerman's formula. State Observers – Full order and Reduced order observers.

**Text books :**

1. Discrete-Time Control systems - K. Ogata, Pearson Education/PHI, 2nd Edition
2. Digital Control and State Variable Methods by M.Gopal, TMH

**References :**

1. Digital Control Systems, Kuo, Oxford University Press, 2nd Edition, 2003.
2. Digital Control Engineering, M.Gopal.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**I SEMESTER**

## Distributed Operating Systems

### Elective - I

**Unit – I**

Characterization of Distributed Systems, Design Issues, User requirement, Network Technologies and Protocols, IPC, Client-server Communication, Group communication, IPC in UNIX.

**Unit – II**

Remote Procedure Calling, Design issues, Implementation, Asynchronous RPC.

**Unit – III**

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

**Unit – IV**

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.

**Unit – V**

Concurrency control, Locks, Optimistic concurrency control, Timestamp ordering, Comparison of methods for concurrency control.

**Unit – VI**

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 – VII**

Cryptography, Authentication and key distribution, Logics of Authentication, Digital signatures.

**Unit – VIII**

Distributed shared memory, Design and Implementation issues, Sequential consistency and ivy, Release consistency and Munin, Overview of Distributed Operating systems Mach, Chorus.

**Text Books:**

Distributed Systems Concepts and Design, G Coulouris, J Dollimore and T Kindberg, Second Edition, Addison Wesley

**Reference Books:**

Advanced Concepts in Operating Systems, M Singhal, N G Shivarathri, Tata McGraw-Hill Edition

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**I SEMESTER**

## Digital System design

**Elective - II**

**Unit – I**

DESIGN OF DIGITAL SYSTEMS: ASM charts, Hardware description language and control sequence method, Reduction of state tables, state assignments.

**Unit – II**

SEQUENTIAL CIRCUIT DESIGN: design of Iterative circuits, design of sequential circuits using ROMs and PLAs, sequential circuit design using CPLD, FPGAs.

**Unit – III**

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 technique, Boolean difference method, Kohavi algorithm.

**Unit – IV**

TEST PATTERN GENERATION: D – algorithm, PODEM, Random testing, transition count testing, Signature analysis and testing for bridging faults.

**Unit – V**

FAULT DIAGNOSIS IN SEQUENTIAL CIRCUITS: State identification and fault detection experiment. Machine identification, Design of fault detection experiment.

**Unit – VI**

PROGRAMMING LOGIC ARRAYS: Design using PLA's, PLA minimization and PLA folding.

**Unit – VII**

PLA TESTING: Fault models, Test generation and Testable PLA design.

**Unit – VIII**

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)
3. Nolman Balabanian, Bradley Calson – "Digital Logic Design Principles" – Wily Student Edition 2004.

**Refrence books:**

1. M. Abramovici, M. A. Breues, A. D. Friedman – "Digital System Testing and Testable Design", Jaico Publications
2. Charles H. Roth Jr. – "Fundamentals of Logic Design".
   Frederick. J. Hill & Peterson – "Computer Aided Logic Design" – Wiley 4th Edition

**I SEMESTER**

## Fault Tolerant Systems

### Elective - II

**Unit – I**
Basic Concepts
Failure and Faults, reliability and failurerate, relation between eligibility and Mean-time – between failures, maintainability and availability, reliability of series and parallel systems, Modeling of faults, stuc at, Bridging (short- circuit), stuck open, transient and intermittent faults.

**Unit – II**
Fault diagnosis of digital systems, Test generation for combinational logic circuits – conventional methods, Random testing, transition count testing and signature analysis

**Unit – III**
Basic concepts – static, dynamic, Hybrid, and self-purging redundancy, shift –over Modular Redundancy (SMR). Tripte Modular redundancy, SMR.

**Unit – IV**
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 – V**
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 – VI**
Strongly fault secure circuits, failsafe Design of sequential circuits using partition theory and Berger codes totally self-checking PLA design

**Unit – VII**
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 – VIII**
Design of Testable Sequential circuits the scan-path technique – leel 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 Books:**
Fault Tolerant and fault testable hardware design Parag K. Lala PHI 1985.

**Reference Books:**
1. Digital systems design using PLD's LALA, PHI 1990
2. Logic Design theory, N. N. Biswas, PHI 1990.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**I SEMESTER**

## Microprocessors and Programming Language Lab

1. 8259 – Interrupt Controller
2. 8279 – Keyboard display
3. 8255 – PPI
4. 8251 – USART
5. Reading and Writing on a parallel port.
6. Timer in different modes.
7. Serial communication implementation
8. Excersice on C++, prolog lisp and similar

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**II SEMESTER**

## Real-Time Systems and Programming Languages

**Unit - I**

Introduction to Real-Time Systems, Definition of a real-time system, Examples of real-time systems, Characteristics of real-time systems, Summary

Designing Real-Time Systems

Levels of notation, Requirement specification, Design activities, Design methods, *Implementation*, *Testing*, Prototyping, Human computer interaction, Managing design Summary

**Unit - II**

Programming in the Small

Overview of Ada, Java, C and occam2, Lexical conventions, Overall style, Data types, Control structures, Subprograms, Summary

Programming in the Large

Information hiding, Separate compilation, Abstract data types, Object-oriented programming, Reusability, Summary

**Unit - III**

Reliability and Fault Tolerance

Reliability, failure and faults, Failure modes, Fault prevention and fault tolerance, N-Version programming, Software dynamic redundancy, The recovery block approach to software fault tolerance, A comparison between N-version programming and recovery blocks, Dynamic redundancy and exceptions, Measuring and predicting the reliability of software, Safety, reliability and dependability, Summary

Exceptions and Exception Handling

Exception handling in older real-time languages, Modern exception handling, Exception Handling in Ada, Java and C, Exception handling in other languages, Recovery blocks and exceptions, Summary

**Unit - IV**

Concurrent Programming

The notion of process, Concurrent execution, Process representation, A simple embedded system, Summary

Shared Variable-Based Synchronization and Communication

Mutual exclusion and condition synchronization, Busy waiting, Suspend and resume, Semaphores, Conditional critical regions, Monitors, Protected objects, Synchronized methods Summary

**Unit - V**

Message-Based Synchronization and Communication

Process synchronization, Process naming and message structure, Message-passing semantics of Ada and occam2, Selective waiting, POSIX messages, The CHILL language, Remote procedure call, Summary

Atomic Actions, Concurrent Processes and Reliability

Atomic actions, Atomic actions in concurrent languages, Atomic actions and backward error recovery, Atomic actions and forward error recovery, Asynchronous notification, POSIX signals, Asynchronous event handling in Real-Time, Java, Asynchronous transfer of control in Ada, Asynchronous transfer of control in Real-Time Java, Summary

**Unit - VI**

Resource Control

Resource control and atomic actions, Resource management, Expressive power and ease of use, The requeue facility, Asymmetric naming and security, Resource usage, Deadlock, Summary

Real-Time Facilities

The notion of time, Access to a clock, Delaying a process, Programming timeouts, Specifying timing requirements, Temporal scopes, Language support for temporal scopes, Fault tolerance, Summary

**Unit - VII**

Scheduling

Simple process model, The cyclic executive approach, Process-based scheduling, Utilization-based schedulability tests, Response time analysis for FPS, Response time analysis for EDF, Worst-case execution time, Sporadic and aperiodic processes, Process systems with D < T, Process interactions and blocking, Priority ceiling protocols, An extendible process model, Dynamic systems and on-line analysis, Programming priority-based systems, Summary

Distributed Systems

Distributed system definition, Overview of issues, Language support, Distributed programming systems and environments, Reliability, Distributed algorithms, Deadline scheduling in a distributed environment, Summary

**Unit - VIII**

Low-Level Programming

Hardware input/output mechanisms, Language requirements, Modula-1, Ada, Real-Time Java, Occam2, C and older real-time languages, Scheduling device drivers, Memory management, Summary

The Execution Environment

The role of the execution environment, Tailoring the execution environment, Scheduling models, Hardware support, Summary

A Case Study in Ada

Mine drainage, The HRT-HOOD design method, The logical architecture design, The physical architecture design, Translation to Ada, Fault tolerance and distribution, Summary, Conclusions

**Text Books:**

Real-Time Systems and Programming Languages - A. Burns, A. J Wellings Addison Wesley 2001

**II SEMESTER**

## Embedded Systems Development

**Unit – I**
An Overview of Embedded Software, Applications of Embedded Systems, Hardware Architectures for Embedded Systems, Developing for Embedded Systems, Embedded Software Development Environments, Hardware Fundamentals for the Software Engineer: Terminology, Gates, Timing Diagrams and Memory.

**Unit – II**
Advanced Hardware Fundamentals, Microprocessors, Buses, Direct Memroy Access, interrupts, Other Common Ports, Conventions Used in Schematics. Interrupts, Microprocessor Architecture, Interrupt Basics, The Shared-Data Problem, Interrupt Latency.

**Unit – III**
Operating Systems Basics, Round-Robin, Round-Robin with Interrupts, Function-Queue-Scheduling Architecture, Real-Time Operating System Architecture. Introduction to Real-Time Operating Systems, Tasks and Task States, Tasks and Data, Semaphores and Shared Data.

**Unit – IV**
More Operating System Services, Message Queues, Mailboxes, and Pipes, Time Functions, Events, Memory Management, Interrupt Routines in an RTOS Environment, Basic Design Using a Real-Time Operating Systems, Encapsulating Semaphores and Queues, Hard Real-time Scheduling Considerations, Saving Memory Space and Power.

**Unit – V**
Embedded Software Development Tools, Host and Target Machines, Linker/Loaders for Embedded Software, Getting Embedded Software into the Target System, Debugging Techniques, Testing on Your Host Machine, Instruction Set Simulators, Using Laboratory Tools, Serial Communication Programming.

**Unit – VI**
Introduction to Real – Time Operating Systems: Tasks and task States, Tasks and Data, Semaphores, and Shared Data, Message Queues, Mailboxes and Pipes, Time functions, Events, Memory Management, Interrupt Routines in an RTOS Environment.

**Unit – VII**
Basic Design Using a Real – Time Operating System: Principles, Semaphores and Queues, hard Real-Time Scheduling considerations, Saving Memory and Power, An example RTOS like uC-OS (Open Source); Embedded Software Development Tools; Hot and Target machines, Linker/Locators for Embedded Software, Getting Embedded Software into the Target System: Debugging Techniques: Testing on Host Machine, Using Laboratory Tools, an Example System.

**Unit – VIII**
Introduction to advanced architectures: ARM and SHARC, Processor and memory organization and Instruction level parallelism; Networked embedded systems; Bus protocols, $I^2C$ bus and CAN bus; Internet-Enabled Systems, Design Example-Elevator Controller.

**Text Books:**

1. Computers and Components, Wayne Wolf, Elseveir.
2. The 8051 Microcontroller, Third Edition, Kenneth J. Ayala, Thomson.
3. An Embedded Software Primer by David E. Simon, Pearson Education.
4. Programming for Embedded Systems by Dreamtech Software Team, Wiley Publishing Inc.

**References:**

1. Embedding 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.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**II SEMESTER**

## Real Time Operating Systems

**Unit – I**

Typical Real-Time Applications

Digital Control, High-Level controls, Signal processing, Other Real-Time Applications, Summary

Hard Versus Soft Real-Time Systems

Jobs and Processors, Release Times, Deadlines, and Timing Constraints, Hard and Soft Timing Constraints, Hard Real-Time Systems, Soft Real-Time Systems, Summary

A Reference Model of Real-Time Systems

Processors and Resources, Temporal Parameters of Real-Time Workload, Periodic Task Model, Precedence Constraints and Data Dependency, Other Types of Dependencies, Functional Parameters, Resource Paramenters of jobs and parameters of resources, Scheduling Hierarchy, Summary

**Unit – II**

Commonly Used Approaches to Real-Time Scheduling

Clock-Driven Approach, Weighted Round-Robin Approach, Priority-Driven Approach, Dynamic versus Static Systems, Effective Release Times and deadlines, Optimality of the EDF and LST algorithms, Nonoptimality of the EDF and the LST algorithms, Challenges in validating timing constraints in priority-driven systems Off-Line versus On-Line scheduling, Summary

**Unit – III**

Clock-Driven Scheduling

Notations and Assumptions, Static, Timer-Driven scheduler, General structure of Cyclic schedules, Cyclic Executives, Improving the average response Time of aperiodic jobs, Scheduling Sporadic Jobs, Practical Considerations and Generalizations, Algorithm for Constructing Static schedules, Pros and Cons of clock-driven scheduling, Summary

Priority-Driven Scheduling of Periodic Tasks

Static assumption, Fixed-priority versus dynamic-priority algorithms, Maximum schedulable utilization, Optimality of the RM and DM algorithms, A schedulability test for fixed-priority tasks with short response times, Schedulability test for fixed-priority tasks with arbitrary response times, Sufficient schedulability conditions for the RM and DM algorithms, Practical Factors, Summary

### Unit – IV

Scheduling Aperiodic and Sporadic Jobs in Priority-Driven Systems
Assumptions and approaches, Deferrable Servers, Sporadic servers, Constant Utilization, Total bandwidth, and Weighted fair-queueing servers, Stack stealing in deadline-driven systems, Stack stealing in fixed-priority systems, Scheduling of sporadic jobs, Real-Time performance for jobs with soft timing constraints, A two-level scheme for integrated scheduling, Summary

### Unit – V

Assumptions on Resources and their usage, Effects of resource contention and resource access control, Nonpreemptive critical sections, Basic priority-inheritance protocol, Basic priority-ceiling protocol, Stack-based, priority-ceiling(Ceiling-priority – Protocol, Use of priority-ceiling protocol in dynamic-priority systems, Preemption-ceiling protocol, Controlling Accesses to multiple-unit resources, Controlling concurrent accesses to data objects, Summary

### Unit – VI

Multiprocessor Scheduling, Resource Access Control, and Synchronization
Model of multiprocessor and distributed systems, Task assignment, Multiprocessor priority-ceiling protocol, Elements of scheduling algorithms for end-to-end periodic tasks, End-to-end tasks in heterogeneous systems, Predictability and validation of dynamic multiprocessor systems, Summary

Scheduling Flexible Computations and Tasks With Temporal Distance Constraints
Flexible applications, Tasks with temporal distance constraints, Summary

### Unit – VII

Real-Time Communication
Model of Real-Time communication, Priority-based service disciplines for switched networks, Weighted Round-Robin service disciplines, Medium access-control protocols of broadcast networks, Internet and resource reservation protocols, Real-Time protocol, Communication in multicomputer systems, Summary

### Unit – VIII

Operating systems
Overview, Time Services and scheduling mechanisms, Other basic operating system functions, Processor reserves and resource kernel, Open system architecture, Capabilities of commercial real-time operating systems, Predictability of general-purpose operating systems, Summary

**Text Books:**
Real-Time Systems – Jane W.S. LIU

**II SEMESTER**

## Quality Assurance of Real – Time systems

**Unit - 1**
Discrete random variables, Binomial distribution, Poisson distribution, Acceptance testing

**Unit - II**
Continuous random variables, Normal, exponential, and Weibull distributions

**Unit - III**
Data and distributions, Goodness-of-fit

**Unit - VI**
Reliability and rates of failure, Constant failure rate model, Time-dependent failure rates

**Unit - V**
Redundancy, Active parallel systems, High- and low-level redundancy

**Unit - VI**
Maintained systems, Ideal and imperfect preventive maintenance, Availability, Maintainability

**Unit - VII**
Failure Interactions, Markov analysis

**Unit - VIII**
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

**Textbooks:**

1. E. Lewis, "Introduction to Reliability Engineering," 2nd edition, Wiley
2. Real-time Systems Specification, Verification and Analysis - Mathai Joseph

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**II SEMESTER**

## Low Power VLSI Design

**Elective - III**

**Unit - I**

LOW POWER DESIGN, AN OVER VIEW:

Introduction to low- voltage low power design, limitations, Silicon-on-Insulator.

**Unit - II**

MOS/BiCMOS PROCESSES :

Bi CMOS processes, Integration and Isolation considerations, Integrated Analog/Digital CMOS Process.

**Unit - III**

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 - IV**

DEVICE BEHAVIOR AND MODELING:

Advanced MOSFET models, limitations of MOSFET models, Bipolar models.

**Unit - V**

Analytical and Experimental characterization of sub-half micron MOS devices, MOSFET in a Hybrid-mode environment

**Unit - VI**

CMOS AND Bi-CMOS LOGIC GATES:

Conventional CMOS and BiCMOS logic gates. Performance evaluation

**Unit - VII**

LOW-VOLTAGE LOW POWER LOGIC CIRCUITS:

Comparison of advanced BiCMOS Digital circuits. ESD-free Bi CMOS, Digital circuit operation and comparative Evaluation.

**Unit - VIII**

LOW POWER LATCHES AND FLIP FLOPS:

Evolution of Latches and Flip flops-quality measures for latches and Flip flops, Design perspective.

**Text Books**

CMOS/BiCMOS ULSI low voltage, low power by Yeo Rofail/ Gohl(3 Authors)-Pearson Education Asia 1st Indian reprint,2002

**References**

1. Digital Integrated circuits , J.Rabaey PH. N.J 1996
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)
4. IEEE Trans Electron Devices, IEEE J.Solid State Circuits, and other National and International Conferences and Symposia.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD
**M.TECH.(REAL TIME SYSTEMS)** w.e.f. 2007-08

**II SEMESTER**

## Case studies in Real – Time Operating systems
### Elective – III

**Unit – I**
System Architecture: Requirements and design goals, Operating system models, Architecture View, Portability, Symmetric Multiprocessing, Windows NT workstation vs Windows NT Server, Key System components, Environment subsystems and subsystems DLLs, NTDLL.DLL Executive, Trap Dispatching, Object Manager, Synchronization.

**Unit – II**
Process and Threads, Process Internals, Flow of Create Process, Thread Internals, Thread scheduling, Memory management, Service the memory manager, Digging in to the memory manager, address space layout, address translation, page fault handling, virtual address descriptors, page from data base, Security, protecting objects, security auditing

**Unit – III**
I/O systems, I/O structure and model , Device drivers, Data structures, I//O processing, Cache Manger, Windows NT file system.

**Unit – VI**
Developing Embedded applications for mobile computing using windows XP

**Unit – V**
Applications and service development, Application – end user communication services and networking core OS services
Revise Management
File system and data store fonts

**Unit – VI**
Graphics and multimedia technologies, international internet client services, security shell and user interface, voice over IP phase services

Board support packages, production – quality OAL
Production – Quality BSPS

Production – quality drivers Device Manager Handling race condition I/O resource manager

**Unit – VII**
Configuration the workstation, first test program, directories, configuration files, make file, Target files, resources

Debugging embedded software, the target group, GPB, debugging a sample program, the jost of a debug environment, adding programmable set point and limit, Kernel modules and device drivers, Kernel modules, what's a device driver anyway, Linux device driver, Internal

drives structure, the hardware, the target version of thermostat, Debugging Kernel code, building your driver into the Kernel, an aleternate – U C Linux, resources

**Unit – VIII**

Embedded Networking, Sockets, a simple example, a remote thermostat, embedded web servers, resources

Linux & Real Time, Why Linux isn't real time, two approaches, resources – obtaining real time Linux implementation.

**Text Books:**

1. Linux for Embedded and Real Time applications by Doug Abbott
2. "Inside Windows NT " by David A. Solomon

**Reference Website:** www.windowsembeddedtoolkit.com
www.microsoft.com/embedded

**II SEMESTER**

## Case studies in Real Time applications

### Elective – IV

**Unit - I**
Time and Real-time
Introduction, Real-time computing, Requirements, specification and implementation, The mine pump,

**Unit - II**
Fixed Priority Scheduling – A Simple Model
Introduction, Computational model, Static scheduling, Scheduling with priorities, Simple methods of analysis, Exact analysis, Extending the analysis

**Unit - III**
Advanced Fixed Priority Scheduling
Introduction, Computational model, Advanced scheduling analysis, Introduction to Ada, The mine pump

**Unit - IV**
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 - V**
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 - VI**
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

**Unit - VII**
Specification and Verification in DC
Introduction, Modelling real-time systems, Requirements, Assumptions, Design, The basic duration calculus (DC), The mine pump,

**Unit - VIII**
Specification of scheduling policies, Probabilistic duration calculus (PDC), Historical background, Further work

**Textbooks:**

Real-time Systems Specification, Verification and Analysis - Mathai Joseph

**II SEMESTER**

## Mobile Computing

### Elective - IV

**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.

**Unit – II**

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

**Unit – III**

Mobile Network Layer: Mobile IP(Goals, assumptions, entities and terminology, IP packet delivery, agent advertisement and discovery, registration, tunneling and encapsulation, optimizations), Dynamic Host Configuration Protocol(DHCP).

**Unit – IV**

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

**Unit – V**

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.

**Unit – VI**

Data Dissemination: Communications asymmetry, classification of new data delivery mechanisms, push-based mechanisms, hybrid mechanisms, selective tuning (indexing) techniques.

**Unit – VII**

Mobile Ad hoc Networks (MANETs): Overview, Properties of a MANET, spectrum of MANET applications, routing and various routing algorithms, security in MANETs.

**Unit – VIII**

Protocols and Tools: Wireless Application Protolcol-WAP. (Introduction, protocol architecture, and treatment of protocols of all layers), Bluetooth (User scenarious, physical layer, MAC) layer, networking, security, link management) and J2ME.

**Text Books:**

1. Jochen Schiller, "Mobile Communications", Addison-Wesley, (Chapters 4, 7, 9, 10, 11), second edition, 2004.
2. Stojmenovic and Cacute. "Handbook of Wireless Networks and Mobile Computing", Wiley, 2002, ISBN 0471419028.(chapters 11, 15, 17, 26 and 27)

**References:**

1. Reza Behravafar, "Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML", ISBN: 0521817331, Cambridge University Press, October 2004.
2. Adelstein, Frank, Gupta, Sandeep KS, Richard III, Golden, Schwiebert, Loren, "Fundamentals of Mobile and Pervasive Computing", ISBN: 0071412379, McGraw-Hil Professioonal, 2005.
3. Hansmann, Merk, Nicklous, Stober, "Principles of Mobile Computing", Springer, second edition, 2003.
4. Martyn Mallick, "Mobile and Wireless Design Essentials:, Wiley Dream Tech, 2003.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY, HYDERABAD

M.TECH.(REAL TIME SYSTEMS) w.e.f. 2007-08

**II SEMESTER**

# EMBEDDED SYSTEMS LAB

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) Road, 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 system 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 SDK to run in XP environment.
   b. Building application for writing and debugging application and target system and using platform 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 applications using Linux