Course Title: Real Time Systems **Credit: 3**
Course No: CSIT.423.4 **Number of period per week: 3+3**
Nature of the Course: Theory+Lab **Total hours: 45+45**
Year: Fourth, Semester: Eighth
Level: B. Sc. CSIT

1. Course Introduction

This course introduces theory, mechanisms, and implementations of real-time computer systems. It introduces real-time systems, real-time scheduling, real-time synchronization, real-time operating system kernels, and real-time programming languages. It also covers design and analysis of real-time resource management algorithms (e.g., scheduling, synchronization), their implementations in production operating system kernels, experimental studies of those implementations, and real-time application development.

2. Objectives

Upon completion of this course students should be able to do the following things:

- To identify problems as hard, firm or soft real-time system and give justification
- To articulate and contrast different definitions in real-time systems
- To comprehend formal methods based design approaches and utilize design tools to model real-time systems formally or semi-formally;
- To understand the impact of hardware architectures for real-time performance;
- To analyze the scheduling feasibility of a set of independent tasks;
- To understand resource policies and system services for inter tasks communication and synchronization;
- To differentiate between various performance analysis techniques;
- To understand real-time software testing, verification and system integration.
- To be aware of performance optimization techniques.

3. Specific Objectives and Contents

Specific Objectives	Contents
• Understand overview of the nature of real - time systems • Understand brief historical review and acquaint the reader withpertinent terminology	**Unit I: Fundamentals of Real Time Systems (5Hrs)** 1.1. Definition and concept of real time systems 1.2. Design Challenges 1.3. Evolution of real time systems 1.4. Advancements on modern real time systems
• Understand the detailed review of central computer architecture • Understand the concepts from the perspective of the real - time systems designer	**Unit II: Hardware for real time systems (6Hrs)** 2.1. Processor architecture: Von Neumann architecture, instruction processing, interrupt considerations 2.2. Memory technologies: memory accessibility, class, and hierarchy 2.3. Architecture advancements and peripheral interfacing

• Understand the impact of advanced architectural features on real - time performance • Understand different memory technologies, input/output techniques, and peripheral support for embedded systems	2.4. Microprocessor versus microcontroller 2.5. Distributed real time architecture
• Understand the core of the text for those who are building practical real - time systems • Understand three principal real - time kernel services: scheduling/dispatching, intertask communication/ synchronization, and memory management	**Unit III: Real time operating system (8Hrs)** 3.1. Basics of RTOS 3.2. Scheduling Frameworks: Round-Robin, cyclic code, fixed priority, dynamic priority scheduling 3.3. System services for application 3.4. Memory management issues
• Understand specific language features desirable in good software engineering practice in general and real - time systems design in particular • Understand explicit criteria for rating a language ’ s ability to support real - time systems and to alert the user to the possible drawbacks of using each language in real - time applications	**Unit IV:Programming languages for real time systems (7Hrs)** 4.1. Coding of Real-Time Software: Fitness of a Programming Language for Real-Time Applications, Coding Standards for Real-Time Software 4.2. Assembly Language 4.3. Procedural Languages 4.4. Object-Oriented Languages: Synchronizing Objects and Garbage Collection, Cardelli’s Metrics and Object-Oriented Languages, Object-Oriented versus Procedural Languages
• Understand specific techniques in real - time system specification with illustrative examples • Understand structured and object - oriented methodologies are discussed as alternative paradigms for requirements writing	**Unit V: Requirements Engineering Methodology (6Hrs)** 5.1. Requirements Engineering for Real-Time Systems 5.2. Formal and Semiformal Methods in System Specification 5.3. The Requirements Document
• Understand design specification techniques used in both structured and object - oriented design	**Unit VI:Real time software design approaches (9Hrs)** 6.1. Qualities of Real-Time Software 6.2. Software Engineering Principles 6.3. Procedural Design Approach 6.4. Object-Oriented Design Approach 6.5. Life Cycle Models: Waterfall Model, V-Model, Spiral Model, Agile Methodologies
• Understand the future of real - time systems hardware,	**Unit VII: Future of Real time systems (4Hrs)** 7.1. Future of Real-Time Hardware, Real-Time

software, and applications	Operating Systems 7.2. Future of Real-Time Programming Languages: The UML++ as a Future "Programming Language" 7.3. Future of Real-Time Systems Engineering and Real-Time Applications

Evaluation System

Undergraduate Programs							
External Evaluation	**Marks**	**Internal Evaluation**	**Weight age**	**Marks**	**Practical**	**Weight age**	**Mark**
End semester examination	60	Assignments	20%	20	Practical Report copy	25%	20
(Details are given in the separate table at the end)		Quizzes	10%		Viva	25%	
		Attendance	20%		Practical Exam	50%	
		Internal Exams	50%				
Total External	60	Total Internal	100%	20		100%	20
Full Marks 60+20+20 = 100							

External evaluation

1. **End semester examination:**
   It is a written examination at the end of the semester. The questions will beasked covering all the units of the course. The question model, full marks, time and others will be as per the following grid.

2. **External Practical Evaluation:**
   After completing the end semester theoretical examination, practical examination will be held. External examiner will conduct the practical examination according to the above mentioned evaluation. There will be an internal examiner to assist the external examiner. Three hours time will be given for the practical examination. In this examination Students must demonstrate the knowledge of the subject matter.

Full Marks: 100, Pass Marks: 45, Time: 3 Hrs

Nature of question	Total questions to be asked	Total questions to be answered	Total marks	Weightage
Group A: multiple choice*	20	20	20×1 = 20	60%
Group B: Short answer type questions	7	6	6×8 = 48	60%
Group C: Long answer type questions	3	2	2×16 =32	60%
			100	100%

Each student must secure at least 50% marks in internal evaluation in order to appear in the end semester examination. Failed student will not be eligible to appear in the end semester examinations.

Internal evaluation

Assignment: Each student must submit the assignment individually. The stipulated time for submission of the assignment will be seriously taken.

Quizzes: Unannounced and announced quizzes/tests will be taken by the respective subject teachers. Such quizzes/tests will be conducted twice per semester. The students will be evaluated accordingly.

Attendance in class: Students should regularly attend and participate in class discussion. Eighty percent class attendance is mandatory for the students to enable them to appear in the end semester examination. Below 80% attendance in the class will signify NOT QUALIFIED (NQ) to attend the end semester examination.

Presentation: Students will be divided into groups and each group will be provided with a topic for presentation. It will be evaluated individually as well as group-wise. Individual students have to make presentations on the given topics.

Mid-term examination: It is a written examination and the questions will be asked covering all the topics in the session of the course.

Discussion and participation: Students will be evaluated on the basis of their active participation in the classroom discussions.

Instructional Techniques: All topics are discussed with emphasis on real-world application. List of instructional techniques is as follows:

- Lecture and Discussion
- Group work and Individual work
- Assignments
- Presentation by Students
- Quizzes
- Guest Lecture

Students are advised to attend all the classes and complete all the assignments within the specified time period. If a student does not attend the class(es), it is his/her sole responsibility to cover the topic(s) taught during that period. If a student fails to attend a formal exam/quiz/test, there won't be any provision for re-exam. Unless and until the student clears one semester he/she will not be allowed to study in the following semesters.

Laboratory Work

Student should write programs and prepare lab sheet for major units in the syllabus. They should practice design and implementation of real time systems that demonstrates different concepts discussed is class. However, nature of programming can be decided by the instructor. The lab work should be practiced for minimum of 3 lab hours per week.

Prescribed Text

- Phillip A. Laplante, Seppo J. Ovaska, Real Time Systems Design and Analysis, 4th Edition, Wiley-IEEE Press; (2011)

References

- Jane W. S. Lui, Real Time Systems, First Edition, Pearson Education, 2000
- Elecia White, Making Embedded Systems: Design Patterns for Great Software, 1st Edition(2011)
- Cooling J.E., Software Design for Real-Time Systems, International Thompson Computer Press, London, England, 1991