

Registration No:

--	--	--	--	--	--	--	--	--	--	--

Total Number
of Pages: 02

B. Tech
RCS6D001

6th Semester Reg/Back Examination: 2024-25

SUBJECT: Real Time System

BRANCH(S): CSE, CSEAI, IT

Time: 3 Hour

Max Marks: 100

Q. Code: S179

Answer Question No.1 (Part-1) which is compulsory, any eight from Part-II and any two from Part-III.
The figures in the right-hand margin indicate marks.

Part-I

Q1 Answer the following questions:

(2 x 10)

- Differentiate between static and dynamic task scheduling in real-time system.
- What is a real-time clock, and how does it contribute to RTOS functioning?
- Define interrupt latency and its impact on real-time performance
- Explain the role of task states in an RTOS scheduler (e.g., Ready, Running, Blocked).
- What is priority inversion? Provide one real-world example where it may occur.
- How do semaphores prevent race conditions?
- Define temporal consistency in real-time databases.
- Why are watchdog timers critical in safety-critical real-time systems?
- What is the main scheduling challenge when using backup overloading?
- Differentiate between transient and permanent faults with examples.

Part-II

Q2 Only Focused-Short Answer Type Questions- (Answer Any Eight out of Twelve)

(6 x 8)

- What are the defining characteristics of a real-time system? Discuss the Rate Monotonic Algorithm (RMA). Figure out the issues associated with this algorithm.
- A real-time system handles four tasks with the following parameters: Draw the Gantt chart for Earliest Deadline First (EDF) scheduling. Determine if all tasks meet their deadlines.

Task Arrival Time (ms) Execution Time (ms) Deadline (ms)

A	0	3	7
B	1	2	5
C	2	1	3
D	3	2	6

- Explain how interrupt handling is managed in a real-time operating system. Discuss interrupt latency, context switching, and nested interrupts. Give a diagrammatic view.
- Describe inter-process communication (IPC) mechanisms used in real-time systems. Include message queues, semaphores, and shared memory with examples.
- Discuss and compare the following concurrency control protocols used in real-time databases:
i. Priority Inheritance Protocol (PIP). ii. Priority Ceiling Protocol (PCP).
- Explain how task synchronization is achieved using semaphores and event-flags. Include practical scenarios where each method is appropriate.
- Describe the architectural modifications and features introduced in UNIX-based RTOS variants (e.g., RTLinux) to support real-time capabilities.

- h) Discuss different types of priority inversion under Priority Ceiling Protocol (PCP).
- i) Explain how the overloading of backup tasks works without compromising fault isolation. What are the criteria for scheduling backups on the same processor.
- j) Explain with an example how a real-time task can be implemented and scheduled in a POSIX-compliant UNIX system using real-time APIs.
- k) Explain the challenges involved in scheduling real-time tasks in a distributed system. Consider network latency, clock synchronization, task migration, and fault tolerance.
- l) Describe and compare the following scheduling strategies used in distributed real-time systems: (i). Static vs. Dynamic Scheduling (ii). Centralized vs. Distributed Scheduling

Part-III

Only Long Answer Type Questions (Answer Any Two out of Four)

- Q3** A real-time system has three periodic tasks that are to be scheduled using a clock-driven scheduler. Their parameters are as follows: **(16)**

Task	Period (ms)	Execution Time (ms)	Offset (ms)
T1	4	1	0
T2	6	2	0
T3	12	3	0

Assumptions are: (i) All tasks are ready to execute at their offsets, Preemption is allowed, (ii) the schedule is to be built for the least common multiple (LCM) of the periods, i.e., hyperperiod, (iii) the system tick is 1 ms.

Questions: a) Compute the hyperperiod of the system. (b) Construct a static time schedule showing task execution over the hyperperiod using a Gantt chart. (c) Comment on the CPU utilization and any idle time in the schedule. (d) Discuss advantages and limitations of using clock-driven scheduling for this task set.

- Q4** Describe how conflict resolution and deadlock prevention are handled in real-time concurrency control. (i) How are missed deadlines handled? (ii) What role does transaction abort/restart play in RTDB systems? **(16)**

- Q5** Given the following set of transactions, analyze which protocol (PIP or PCP) would be more appropriate, and justify your reasoning: **(16)**

Transaction	Start Time (ms)	Execution Time (ms)	Deadline (ms)	Data Item Accessed	Priority
T1	0	5	20	X (Read)	High
T2	1	4	15	X (Write)	Medium
T3	2	6	30	Y (Read)	Low

Assume resource locking is required and preemption is allowed. Draw a Gantt chart showing transaction execution and blocking under your chosen protocol.

- Q6** Writ short notes on following **(16)**
- a) Memory management in RTOS
 - b) Interrupt handling in RTOS
 - c) Real-time clock and its importance
 - d) Real-time communication over packet switched network