Department of Electrical and Computer Engineering

University of Colorado at Boulder

ECEN5623 - Real Time Embedded Systems

Homework 1

Submitted by

Parth Thakkar

Submitted on January 24, 2024

Contents

List of Figures

List of Tables

PDF is clickable

1 Questions

1.1 Real-Time Embedded Systems

Q: Provide examples of real-time embedded systems you are familiar with and describe how these systems meet the common definition of real-time and embedded.

1.1.1 Answer

Answer: Definition
A real-time embedded system is designed for time-sensitive tasks. It operates under strict timing constraints, called deadlines. Real time systems manages time critical tasks, Scheduling them and managing resources and IO's for sensing and actuating to th real world.

Real-time Embedded Systems Examples
Automotive Control Systems: Modern vehicles use embedded systems for various functions like engine control, airbag deployment, and anti-lock braking systems. These systems are real-time because they must respond to inputs (like brake pedal pressure or wheel speed) within a guaranteed time to ensure safe operation.
Medical Devices: Devices like pacemakers or insulin pumps are real time embedded systems. They continuously monitor patient conditions and provide necessary adjustments in real-time. For example, a pacemaker must adjust heartbeats in response to the body's needs without any deadline misses.
Telecommunications Systems: Network routers and switches are embedded systems that must handle data packets in real-time. Delays or errors are accepted in this case, This was example of soft real time system where deadline miss will be accepted in some cases

1.2   Liu and Layland, Section 3

Q: Find the Liu and Layland paper and read through Section 3. Why do they make the assumption that all requests for services are periodic? Why might this be a problem with a real application?

1.2.1   Answer

Answer: Liu and Layland Paper: Assumption of Periodicity In Section 3, Liu and Layland assume that all requests for services are periodic, meaning they occur at regular, predictable intervals. This assumption is made for several reasons:

Simplification for Modeling: Assuming periodic requests simplifies the mathematical modeling of the scheduling problem, making it more easy to develop scheduling algorithms.

Reflects Many Real-World Scenarios: Many real-time applications, especially in process control and industrial automation, naturally exhibit periodic behavior (e.g., sensors providing regular updates).

This assumption can be problematic for real applications because:

Not All Systems are Periodic: Some real-time systems, like event-driven systems in multimedia applications, have aperiodic requests time. The assumption of periodicity doesn't accurately represent these systems.

Flexibility: Real-world scenarios often have unpredictable events. A system designed only for periodic requests might not handle unexpected or non-periodic tasks efficiently.

1.3 Hard and Soft Realtime Systems

Q: Define hard and soft real-time services and describe why and how they are different.

1.3.1 Answer

Answer: Hard and Soft Real-Time Services

1. Hard Real-Time Services:

   Definition: Hard real-time systems are those where it is critical that responses to inputs occur within a strict deadline. Failure to meet these deadlines can lead to catastrophic results. Example: An anti-lock braking system in a car where delayed response can cause accidents.

2. Soft Real-Time Services:

   Definition: Soft real-time systems are those where deadlines are important but not absolute. Performance degrades if deadlines are missed, but it doesn't lead to system failure. Example: Video streaming services where occasional delays might cause minor glitches but won't crash the system.

3. Differences:

   Consequence of Missing Deadlines: In hard real-time, missing a deadline can lead to system failure or dangerous situations, while in soft real-time, it results in reduced performance or quality. Applications: Hard real-time is common in safety-critical systems (e.g., aerospace, medical devices), while soft real-time is seen in consumer electronics and entertainment systems. Scheduling, Allocation: Hard real-time systems require more strict scheduling algorithms and resource allocation to ensure deadlines are always met, unlike soft real-time systems which can have some flexibility.