



---

## *Real-Time & Embedded Systems*

*Uwe R. Zimmer*

Remember: these questions are NOT covering the whole lecture, they are supposed to be training material, in order to get used to the style of questions to be expected. The question set for the actual exams will cover all major areas for each candidate.

- (1) What is the role of the operating system in a real-time system? – specifically with respect to the classical role of an operating system.
- (2) What are the criteria by which you would categorize real-time languages? – give examples!
- (3) Why are object oriented programming in the strong sense and real-time systems not going very well together?
- (4) What is ‘causality’ in a synchronous program?
- (5) Compare a pipelined-flash with an successive approximation shift register (SAR) converter!
- (6) How does a  $\Sigma$ - $\Delta$  converter operate?
- (7) Timing behaviours are often expressed in terms of differential equations in engineering domains. Why are differential equations rarely used as the general form of description in software designs?
- (8) What is going to happen, if the indication for an asynchronous transfer of control occurs, when the task, which is to be interrupted is currently not in an interruptible section?
- (9) What are the differences between semaphores and guards?
- (10) Can you call another monitor from within a monitor? – what do you need to consider?
- (11) Why is FPS much more frequently implemented and employed than EDF?
- (12) Explain the maximal blocking times in case of priority inheritance and ceiling priority protocols!
- (13) Which systems are not accessible or problematic for n-version programming methods?