Real-Time Systems

Lecturer: Prof. Dr. Jörg Friedrich, joerg.friedrich@hs-esslingen.de, Room F 01.316

Exam aids: Two double sided DIN A4 sheets and a pocket calculator

Course Objectives: Gain Knowledge

- What constitutes a real-time system?
- How are real-time systems being developed?
- How are real-time systems being modeled?
- Understand concurrency, synchronization and communication.
- Understand real-time operating system primitives.
- Understand real-time systems scheduling.

Course Objectives: Acquire Skills

- Learn how to develop a real real-time system in a real project.
- Learn how to develop software to control hardware.
- Learn how to model real-time systems with UML state charts and activity diagrams.
- Learn how to create real software from these models.
- Learn how to verify real-time software without real-time hardware.
- Learn how to work with project management and change management tools.

Literature for this lecture

Text books for this lecture

[1.1] Laplante, P.: Real-Time Systems Design and Analysis, IEEE Press, 2004
[1.2] Kopetz, H.: Distributed Real-Time Systems, Springer 2008
[1.3] Kienzle, E., Friedrich, J.: Echtzeitsysteme, Hanser 2008
[1.4] Buttazzo, G.: Hard Real-Time Computing Systems, Springer 2005

Complementing texts

Some books complementing the material treated in this lecture

[2.1] Liu, J.S.: Real-Time Systems, Prentice Hall 2000
[2.2] Williams, R.: Real-Time Systems Development, Elsevier 2006
[2.3] Halbwachs, N.: Synchronous Programming of Reactive Systems, Kluwer 1993

Journal Articles and Web Documents

Original journal articles and documents pertaining to this lecture

[3.0] Harel, D. et al.: STATEMATE: A Working Environment for the Development of Complex Reactive Systems; IEEE Trans. Software Eng., Vol. 16, No. 4, April 1990
[3.1] Liu, C.L.; Layland, J.W.: Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment; Journal of the ACM, Vol. 20, No. 1, January 1973
[3.2] Goodenough, J.B.; Sha, L.: The Priority Ceiling Protocol: A Method for Minimizing the Blocking of High Priority Ada Tasks; Ada Letters, Vo. VIII, No. 7; 1988
[3.3] Sha, L.; Rajkumar, R.; Lehoczky, J.P.: Priority Inheritance Protocols: An Approach to Real-Time Synchronization; IEEE Trans. Computers, Vol. 39, No.9, September 1990

Chapter 1: The Real-Time Environment

- Definition of a real-time system
- Simple model with operator, computer system, and controlled object
- Introduction of distributed real-time systems
- Hard real-time systems and soft real-time systems
- Functional, temporal, and dependability requirements
- Sphere of control
- Event-triggered versus time-triggered systems

Chapter 2: The System Development Process

- The V-Model as an example for a real-time system development process
- Project management, controlling deadlines, budgets, and resources
- Configuration management, managing varieties, managing changes
- Requirements engineering
- First glance at system design
- Introduction to system validation: faults, failures, and bugs

Chapter 3: Hardware Considerations

- Hardware interfacing, basic input and output
- Raw data, measured data, and agreed data
- Sampling and polling
- Interrupts
- Sensors and actuators
- Overview on real-time communication systems

Chapter 4: Modeling Real-Time Systems

- Introduction of a conceptual model for real-time systems
- Tasks, nodes, fault-tolerant units, clusters
- Simple and complex tasks
- Interface placement and interface layout
- Temporal control and logical control
- The history state
- UML state charts and activity diagrams

Chapter 5: The Software Production Process

- Implementing simple real-time systems
- Interrupt-driven systems
- Hints for programming in the C language
- Directory layout and software layering
- Software development without the target hardware
- Cyclic executives for bare hardware
- Implementing finite state machines

Chapter 6: Real-Time Entities and Images

- Real-time entities
- Observations, state and event observations
- Real-time images as current picture of real-time entity, and real-time objects
- Temporal accuracy and state estimation to improve real-time image accuracy
- Permanence in case of race conditions and idempotency with replicated messages
- Replica determinism to implement fault-tolerance by active redundancy

Chapter 7: Concurrency

- Concurrency with real-time operating systems
- Processes, threads, and multitasking
- Real-time operating systems
- Task management

Chapter 8: Synchronization and Communication

- Problems due to concurrency
- Synchronization and coordination with semaphores
- Event communication with event flags
- Message queues and mailboxes
- Signals

Summary

Chapter 9: Real-Time Scheduling

- The scheduling problem
- The adversary problem
- Dynamic and static scheduling
- Aperiodic scheduling
- Periodic scheduling

Chapter 10: Scheduling Hybrid Task Sets

- Hybrid task sets
- Background scheduling of aperiodic tasks
- Polling server
- Sporadic server

Chapter 11: Resource Access Protocols

- Scheduling dependent task sets
- Priority Inheritance Protocol
- Priority Ceiling Protocol
- Scheduling anomalies in dependent task sets