Course Summary
Real-Time Systems

2018

Remaining Lectures

- Thursday March 29, 15:15-17:00 (research topics)
- Monday May 14, 13:15-15:00
  - Demo lecture, meet in M:B first, then in lab
- Monday May 14, 15:15-17:00
  - Oral presentations
  - 10 minutes / group
  - Projectors will be available
  - Parallel sessions with multiple groups per session – a schedule will be distributed later

Exams

- Wednesday, April 11
  14:00 -19:00, Victoriastadion 1A-1B
- Thursday, May 31
  8:00 - 13:00, Sparta D
- Saturday, September 1
  8:00 - 13:00, MA 9A

Real-Time Systems

The most important parts!

or

with some luck

What you need to know to pass the exam

Lecture 1: Introduction

- Basic definitions (hard, soft, …)
- Timing parameters in continuous controllers (sampling latency, sampling interval, input-output latency)
- Different event types (periodic, aperiodic, sporadic)

Lecture 2: Concurrent programming

- Process vs threads
- Process’ internal states and state transitions
- The ReadyQueue
- Context switches
  - Save, restore
  - The role of the stackpointer
- Process representation
- The Schedule procedure in Stork
- Java threads:
  - Extend Thread versus implement Runnable
  - Thread priorities

Lecture 3: Process communication 1

- Non-reentrant code
- Race conditions
- Mutual exclusion
- Semaphores:
  - Use for mutual exclusion and synchronization
  - Logical semantics
  - Different types of semaphores (counting, binary)
  - Basic version vs alternative version
  - Stork implementation
  - Condition synchronization using semaphores
  - Java Class Semaphore

- Monitors:
  - Basic definitions
  - Condition variables
  - Monitors in Stork
    - Implementation
- Synchronization in Java
  - Synchronized methods
  - Synchronized blocks
  - Instance locks vs class locks
  - Condition synchronization in Java
  - Class ConditionVariable
- Producer-Consumer example
  - Using semaphores
  - Using synchronization
- Passing objects between threads

Lecture 4: Process communication 2

- Deadlock
  - Necessary conditions
  - Deadlock handling (prevention, avoidance, detection & recovery)
  - Hierarchical resource allocation
- Priority inversion
  - When does it occur?
  - Basic priority inheritance
  - Priority Ceiling
  - Immediate inheritance
- Message passing
  - Alternative schemes (asynchronous/synchronous, direct/indirect)

Lecture 5: Interrupts and time

- Interrupts and interrupt handling
- Clock interrupts
  - The actions performed in the clock interrupt handler
  - TimeQueue
- Tick-based vs event-based clock interrupts
- Foreground-background schedulers
- Time primitives (relative vs absolute)
- Implementation of periodic controller tasks:
  - Different alternatives and their problems
- Minimizing the input-output latency
  - CalculateOutput and UpdateState
  - Cascaded controllers
- Jitter

Lecture 6: Sampling of linear systems

- Sample and Hold
- Effects of sampling
- Aliasing
- ZOH sampling
- ZOH sampling of systems with input delays
- Calculating Φ and Γ
- Solution of system equations
- Stability regions
- Convolution
- From difference equations to state-space

Lecture 7: Input-output models

- Shift operators and z-transform
- Pulse transfer operator and Pulse transfer function
- Poles and zeros
- Input-output models
- Frequency response
- Transformation of poles
- Calculation of H(z)

Lecture 8: Approximations of analog controllers, PID control

- Different approximation methods
- Prewarping
- PID control
  - Textbook algorithm (P, I, and D part)
  - Absolute versus incremental form
  - Algorithm modifications
    - Setpoint weighting
    - Limitation of derivative gain
    - Derivative weighting
  - Windup and anti-windup
    - Tracking
  - Bumpless mode and parameter changes
  - Discretization
  - Code

Lecture 9: State feedback and observers

- State feedback
- Deadbeat
- Observers
  - Prediction form
  - Filter form (with direct term)
- Disturbance estimation & integral action

Lecture 10: Feedforward design

- Feedforward to reduce disturbances
- Feedforward to handle reference changes
  - Transfer function approach
  - State-space approach
  - Nonlinear reference generation

Lecture 11: Implementation aspects

- Sampling & Aliasing
- Choice of sampling interval
- Computational delay
- A-D and D-A quantization
- Pulse width modulation
- Fixed-point arithmetic
  - Q format
  - Two's complement representation
  - Fixed point operations (+, -, *, /) including C code
  - Overflow
  - Sensitivity towards coefficient roundoff

Lecture 12: Scheduling theory

- Execution time analysis
  - Measurements vs analysis
  - Basic problems
- CPU utilization
- Critical instant
- Static cyclic scheduling
  - Basic ideas
- Earliest Deadline First Scheduling
  - Draw diagrams
  - Sufficient schedulability condition
  - Overrun behaviour

- Fixed Priority Scheduling:
  - Priority assignment (rate monotonic, deadline monotonic)
  - Rate monotonic analysis
    - Approximate analysis (two formulas !!)
    - 69% rule of thumb
    - Exact analysis
      - Response-time calculations
  - Draw schedules
  - Overrun behaviour
- NOT:
  - Scheduling of aperiodic tasks
  - Alternative scheduling models

Lecture 13: Real-time networks and networked control systems

- The OSI protocol (stack) model
- Shortcomings of the OSI/IP stack for real-time communication
- CAN protocol
  - Basic notions and arbitration mechanism
- TTP
  - Basic notions

Lecture 14: Discrete-event control

- State machines
- Statecharts
- Grafcet
  - Firing rules
  - Action types
  - Be able to use Grafcet in problems and examples
- Petri Nets
  - Firing rules
  - Generalized PNs
  - Dijkstra’s problems
- Coding state machines in Java

Lecture 15: Project specifications

Lecture 16: Hot research topics

- NOT on the exam

Knowledge from the projects

- The use of Java in real-time programming
- The program structure from Lab 1
- Common problems and solutions
- Priorities, synchronization, …..

Typical Exam Problems

- PID implementation
- Discretization of continuous designs
- Synchronization (semaphores, monitors, deadlock)
- Scheduling theory
- Grafcet / Petri nets
- ZOH sampling
- Input-output models
- State feedback / observers / reference signals
- Fixed point arithmetic

Open Book Exam

- You may use the two text books during the exam
- You may NOT use the exercise book
- You may NOT use the slide copies
- No extra notes in the text books
- Problems were the solution can be directly taken from the text books will not be given