Course Summary Real-Time Systems

2013

Remaining Lectures

- Thursday 12 Dec 15:15-17:00
  - Demo lecture
  - We meet in M:B first, then to the Lab
- Thursday 12 Dec 17:15-19:00
  - Oral presentations
  - 10-15 minutes / group
  - Projectors will be available
  - 5 parallel sessions with 5-6 groups per session
  - More information later on the home page

Exams

- Tuesday 17 Dec, 8-13, Victoria
- Tuesday 7 Jan, 14-19, MA:10
- Wednesday 23 Apr 8-13, M:Q

Handout Material

- Slide copies
- Formula sheet 2010
  - Can be used at the exam
- Course requirements 2013

Real-Time Systems

The most important parts!

or (with some luck)

What you need to know to pass the exam

Lecture 1: Real-Time Systems

- 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: Communication Pt 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: Communication Pt 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 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: Z-transform, poles and zeros

- 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: From Analog to Digital

- 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

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

Error

- Error in the handout slides and in IFAC Professsional Brief
- The observer example for the double integrator

The Gamma*u(k) term is forgotten

Resulting observer

$$\hat{x}_1(k+1) = \hat{x}_1(k) + h\hat{x}_2(k) + 2\big(y(k) - \hat{x}_1(k)\big)$$

$$\hat{x}_2(k+1) = \hat{x}_2(k) + \frac{1}{h}\big(y(k) - \hat{x}_1(k)\big)$$

Lecture 10: Reference Generation

- Reference generation:
  - 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 11: Scheduling

- 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 14: Discrete 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: Integrated Control and Scheduling

- Input-output latency
  - Delay margin
- Why is jitter bad?
- Know the basic idea of the Jitter Margin (not the theory)
- TrueTime and Jitterbug
  - What can they do and how are they used?

Lecture 16: Real-Time Networks

- 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

The projects

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

Typical Exam Problems

- PID implementation
- Discretization of Continuous Designs
- Synchronization (semaphores, monitors, deadlock)
- Scheduling
- 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

Courses in Spring 2013

- VT LP1
  - Nonlinear Control
  - System Identification
  - Reglerteori
  - ~~International Project Course in Control~~
- VT LP2
  - Project Course in Control
  - Market Driven Control Systems

Courses 2014/2015 →

- HT LP1
  - Multivariable
  - System Identification
- HT LP2
  - Nonlinear
  - Project in Automatic Control
- VT LP1
  - Real-Time Systems (1+2)
  - Predictive Control (1+2)
- VT LP2
  - Market-Driven Systems
  - Network Dynamics (new course)