REAL-TIME SYSTEMS - 2014

Karl-Erik Årzén

Information:

- Home page:

  `http://www.control.lth.se/Education/EngineeringProgram/FRTN01.html`
- Email: karlerik@control.lth.se
- Phone: 046 2228782

Extra Occasion

This is an extra occasion of this course

From Fall 2014 the regular occasion for the course is VT LP1 + LP2

This is the last time this course fill be given during the Fall

Originally only intended for the students that are studying in their 5th year

Consequences:

- Not possible to do joint project with Predictive Control since that course is not running now
- Exams for you in January, June, August

Introduction

- Motivation
- Course Contents
- Lectures
- Material
- Exercises
- Laboratories
- Projects
- Exam
- Registration

A *real-time system* is a computer system that has to respond to externally generated events or inputs within a finite and specified time period

All control systems are real-time systems

Most real-time systems are *embedded systems*, i.e, the computer is an embedded, integrated part of some equipment or machinery

Embedded systems

- Embedded systems by far the largest computer sector by volume
- A large part of embedded systems are control systems with hard/soft real-time constraints
- Vehicles, telecom, process & manufacturing industri, intelligent building,
- Pervasive computing

Example - Car industry

- A Volvo S80 contains > 50 computers and several networks
- Most for various control applications
- 25-30% of the price
- Software largest part of the cost
- Strong connection between control and software
  - e.g. climate control system: 25.000 lines of C code

Example - Process Automation

- "Industrial IT" buzzword used by ABB
- Integration of automation and IT
  - software, distributed systems, WWW, e-commerce
- Focus on software rather than hardware

Real-Time Systems in Sweden

Real-Time and Embedded Systems have a very strong position in Sweden and in Lund

Research:

- LUCAS: Center for Applied Software Research at LTH
  - Computer Science and Automatic Control
- EASE: Industrial Excellence Center for Embedded Applications Software Engineering
- ELLIIT: The Lund-Linköping Initiative on IT and Mobile Communications

Industry:

- embedded systems and embedded control systems of vital importance to Swedish industry (Ericsson, ABB, Volvo, Scania, ...)

Course Aims

Study methods for design and implementation of computer control systems.

Focused on embedded control systems.

Two parts:

1. Real-Time Programming
2. Computer-Based Design and Implementation of Control Systems

Programming Languages

Java as the main programming language.

However, not a Java course.

We assume basic knowledge of

- Java
- object-oriented programming concepts

Code examples written Modula 2 (very similar to C, Pascal) will be shown.

One laboratory and some of the projects will use C

Relation to Real-Time Programming

The students who have taken the Java-version of the Real-Time Programming course ($\approx$ 50%) at Computer Science will recognize parts from the real-time programming lectures.

During the lectures we will also describe how real-time programming is performed with a conventional real-time programming language (Modula-2) and how a conventional real-time kernel (Stork) is implemented. (You do not have to program in Modula-2)

Deeper understanding and repetition.

Students that have taken the Real-Time Programming course will do a special version of Laboratory 1 in which LJRT is used.

Students that have taken (or are following) the Real-Time Programming course must do a control-oriented project.

Lecturers

Two lecturers:

Karl-Erik Årzén

Alessandro Vittorio Papadopoulos

- http://www.control.lth.se/Staff/AlessandroPapadopoulos.html

Lectures: Study Period 1

1. Introduction. Real-Time Systems.
2. Implementation Techniques. Concurrent Programming.
3. Concurrent Programming: semaphores, monitors, mailboxes.
4. Concurrent Programming: semaphores, monitors, mailboxes.
5. Concurrent programming: interrupts, time, real-time OS.
6. Sampling
7. Input-output models (Anton Cervin)
8. Approximations of continuous-time designs. PID
9. State-space design
10. Reference Signals (Anton Cervin)
11. Implementation aspects
12. Scheduling Theory
13. Project specifications
14. Discrete Event Control

Lectures: Study period 2

1. Integrated Control and Scheduling.
2. Control over networks
3. Project demonstrations (w.refreshments)

The order of some of the lectures may be changed.

Lectures

Study Period 1:

- Tuesdays 8.15 – 10.00 M:B (Weeks 1-7)
- Thursdays 10.15 – 12.00 M:B (Weeks 2-6)
- Fridays 08.15 – 10.00 Ma:Ma03 (Weeks 1-2)

Study Period 2:

- Tuesdays 15.15 – 17.00 M:B (Weeks 1,3)
- Thursday 15.15 – 17.00 M:B (Week 7)

Extra evening overview lecture on Java will be held on Wednesday 4 Sep, 17.15-19.15 (tomorrow!!) in M:B

Extra lecture at the end of Study Period 2 - Rehearsal for the exam

Material

- "Real-Time Control Systems" (RTCS), 2014, K-E Årzén, KF-Sigma
- "Computer-Control: An Overview", 2014 B. Wittenmark, K.J. Åström, K-E Årzén, KF-Sigma
- Exercises with solutions, 2014, KF-Sigma
- Copies of Lecture slides
  - handed out (6 slides/page)
  - all available on the home page
  - anyone wants to have 1 slide/page slides on the home page?

Exercises

Study Period 1:

- Five computer exercises
  - Weeks 2-6.
  - + One extra Java computer exercise during week 1
- Four problem-solving exercises
  - Weeks 3-6
  - + One extra Matlab/Simulink intro exercise during week 2
- Department of Automatic Control, Lab A

Study Period 2:

- Two problem solving exercises
- Weeks 1-2

Exercises: Study Period 1

Four groups:

1. Jerker Nordh: Tue 10-12 (w 2-6), Wed 15-17 (w 3-6)
2. Gabriel Ingesson: Wed 8-10 (w 2-6), Thu 15-17 (w 3-6)
3. Fredrik Bagge Carlson: Wed 10-12 (w 2-6), Fri 8-10 (w 3-6)
4. Victor Millnert: Tue 15-17 (w 2-6), Thu 8-10 (w 3-6)

First computer exercise then problem-solving exercise

Registration needed.

Exercises not mandatory, but strongly recommended

You may also do the computer exercises on your own, at home or in some computer lab.

Exercises: Study Period 2

1. Jerker Nordh: Tue 8-10
2. Gabriel Ingesson: Wed 15-17
3. Fredrik Bagge Carlson: Thu 10-12
4. Victor Millnert: Fri 10-12

Laboratories

Laboratory 1:

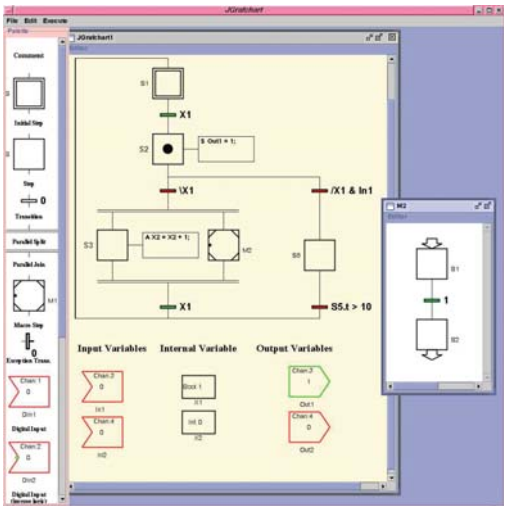
- Implementation of a control system for the ball & beam process
  - Java or Java/LJRT with Swing-based GUI
  - Prepared during the exercises
  - Weeks 6-7 in study period 1

Laboratory 2:

- Sequence control of a bead-sorter process
  - JGrafchart, Java-based Grafcet editor and run-time system
  - Weeks 1-2 in study period 2

Laboratory 3:

- Fix point implementation of a DC-servo controller
  - C on ATMEL AVR Mega16
  - Weeks 2-3 in study period 2



Projects

4 persons group

Around 30 different projects to chose among:

- control of ball and beam process
- control of inverted pendulum
- control of helicopter process
- real-time kernel projects
- embedded system projects using ATMEL AVR and C
- Lego Mindstorm NXT projects
- etc

Joint projects with Predictive Control

Students that have taken (or are following) the Real-Time Programming course must do a control-oriented project

Projects

Projects descriptions handed out around week 6.

Project selection deadline, end study period 1.

Before you are allowed to start coding, a detailed solution suggestion must be approved.

Project work during study period 2.

Project presentations, end study period 2

- oral presentation
- demonstration
- written report
  - Swedish or English
  - 10 - 15 pages

Exam

Written exam

Grades: 3, 4, 5

Open book (but no lecture slide copies and not the exercises)

Book price to those with grade 5

In the December exam period 2014

In the May exam period 2015

In the August exam period 2015

Registration

You must register for the course by signing the form available upfront during the break (will be passed around also during the 2nd hour)

If your name is not in the form please fill in an empty column

LADOK registration will be done immediately

If you decide to abort/skip the course within three weeks from today you should inform me and then the LADOK registration will be removed

Course Requirements

Laboratories

Project

- finished
- report
- presentation

Exam

Changes From Last Year

Most students were happy with the course last year (+64 in CEQ last year, +51 in 2012, +37 in 2011) –> no major changes

Course History

-71-72 Control of LKAB iron ore crusher over modem, PDP 15
-73 “Computers in Control Systems”, PDP 15, assembler
-79 “Computers in Control Systems 2”, LSI-11, Concurrent Pascal
-81 Pascal + real-time kernel
-83 “Applied Real-Time Programming”, IBM PC, Modula 2
-86-87 CS course on real-time programming. Focus on robotics.
-89 “Computer Implementation of Control Systems”, VME 68020
-93 “Real-Time Systems”. CS course no longer a prerequisite.
-96 Windows NT, Pentium, InTouch
-98 PowerPC, Migration to Java started
-00 Only Java, Linux, PC
-03 ATMEL AVR microprocessors introduced
-07 Stronger focus on digital and embedded control