



# IL2212 Embedded Software 7.5 credits

Programvara för inbyggda system

This is a translation of the Swedish, legally binding, course syllabus.

If the course is discontinued, students may request to be examined during the following two academic years

## Establishment

Course syllabus for IL2212 valid from Spring 2019

## Grading scale

A, B, C, D, E, FX, F

## Education cycle

Second cycle

## Main field of study

Computer Science and Engineering, Electrical Engineering

## Specific prerequisites

The course assumes basic knowledge in the design of Embedded Systems, the equivalent course IL2206 embedded systems.

## Language of instruction

The language of instruction is specified in the course offering information in the course catalogue.

## Intended learning outcomes

On completion of the course, the student should be able to

- point out the special nature of software for embedded systems with regard to non-functional requirements
- explain how the predictability of the software performance depends on the underlying hardware architecture
- model a software application so that it can be analysed with regard to real time behavior
- use different technologies for real time system analysis
- model and analyse systems with divided resources
- identify limitations of scheduling algorithms
- point out limitations of idealized models and be able to bring practical factors into the analysis
- understand the basic mechanisms in a real time operating system and be able to use it for the development of software for embedded systems
- give an overview of parallel computing models and their fundamental properties
- evaluate different programming languages and methods for design of software for embedded systems
- outline a construction flow for embedded systems software.

## Course contents

- Non-functional requirements of software for embedded systems.
- Hard and soft real time systems as well as systems where applications with different requirements divide the same platform.
- Task model for real time system.
- Static and dynamic scheduling mechanisms for periodic, aperiodic, and sporadic tasks.
- Protocols for access of divided resources.
- Parallel computing models: Synchronous, data flow, and time based computing models.
- Analysis of real time system.
- Multi-processor real time systems.
- Support for periodic task model and parallel computing models through programming language and operating system.
- Construction flow for software for embedded systems: System modeling, systems analysis and system synthesis.

## Course literature

The reading list consists of current research articles and selected book chapters that are granted four weeks before start of the course. The following recommended books give a good overview of different parts of the course:

- Giorgio Buttazzo, Hard Real-Time Computing System, Run, ISBN 978-1-4614-0676-1, 2011.

- Edward A. Lee and Sanjit A. Seshia, Introduction to Embedded System, A Cyber-Physical the Approach of System, Second Edition, MIT Press, ISBN 978-0-262-53381-2, 2017.

## Equipment

Own laptop computer

## Examination

- LABA - Laboratory Work, 3.0 credits, grading scale: P, F
- TENA - Examination, 4.5 credits, grading scale: A, B, C, D, E, FX, F

Based on recommendation from KTH's coordinator for disabilities, the examiner will decide how to adapt an examination for students with documented disability.

The examiner may apply another examination format when re-examining individual students.

## Ethical approach

- All members of a group are responsible for the group's work.
- In any assessment, every student shall honestly disclose any help received and sources used.
- In an oral assessment, every student shall be able to present and answer questions about the entire assignment and solution.