

# CS 353: Real-Time Systems

Instructor: Sherif Abdelwahed

## Introduction

Real-time systems are systems that require a timely response to external stimuli or they risk severe performance degradation and possibly complete system failure. Real-time systems are often safety-critical and/or part of an embedded system that requires communication and data exchange among the various components. Applications include flight controllers, sensor networks, financial databases, and telecommunication. Key problems in real-time system design include specifying the timing constraints and their importance, and verifying system performance with respect to those constraints.

The main goal of this class is to understand the critical issues involved in real-time system design and development. The course will provide a good overview of the issues related to real-time systems and provide some depth with respect to theoretical work in this field. Special attention is paid to computational models and languages, and analysis tools for real-time systems. An important goal is to give students hands-on experience with these languages and tools. The course will use material from the following references:

- Alan C. Shaw, *Real-time Systems and Software*, Second edition, Wiley, 2001.
- Albert M. K. Cheng, *Real-Time Systems: Scheduling, Analysis, and Verification*, Wiley-Interscience, 2002.
- Jane W. S. Liu, *Real-Time Systems*, Prentice Hall, 2000.

## Prerequisite

The course assumes familiarity with basic discrete mathematics and computer science concepts including relations, functions, graph theory, propositional and predicate logic, and finite-state machines.

## Syllabus

1. Introduction to real-time systems
  - Definitions, examples, characteristics of problem domain
  - Hard vs. soft real-time systems
  - Requirements: predictability, schedulability, correctness
  - Examples of real-time applications
2. Real-time scheduling techniques
  - Static scheduling: cyclic executives
  - Dynamic scheduling with static priority: RM, DM
  - Dynamic scheduling with dynamic priority: EDF, LLF
  - Feedback techniques: rate matching, GF
3. Real-time requirement specification and analysis
  - Timed automata and real-time temporal logic
  - Model checking real-time systems
  - Introduction to real-time model-checking tools: Koronus, Uppaal
4. The synchronous approach
  - Definitions and advantages
  - Synchronous languages: types and computational models
  - Introduction to Esterel
5. Real-time Technologies
  - Real-time OS: Synchronization and concurrency management
  - Real-time middleware
  - Real-time communications: protocols
  - Future visions for RT systems

## Grades

To encourage hands-on experience, there will be several assignments involving prototype-quality real-time design and analysis tools. In addition, each student will have to complete a research project to specify and design a practical case study. Reports on the projects will be written up, and results will be presented in class.

- Assignments 60%
- Project 30%
- Project presentation 10%

## Credit hours

Students will receive 3 hours of credit for this course.