

# ECE 720T5, Fall 2012: Cyber-Physical Systems

## General Information

**Instructor:** Rodolfo Pellizzoni, E5 4113, [rpellizz@uwaterloo.ca](mailto:rpellizz@uwaterloo.ca)

**Lecture Time:** Wednesday 11:30AM – 2:20PM

**Room:** EIT3141

**Office Hour:** Based on appointment

## Calendar Description

This course covers concepts, theory and research issues in safety-critical embedded systems and Cyber-Physical Systems (CPS), with a focus on the architectural design and analysis of the underlying hardware/software computing system.

Topics include: challenges in CPS design, predictable computer architectures, scheduling of hardware resources, operating system abstractions for CPS, timing and performance analysis, modeling and verification, CPS applications.

## Course Objectives

- Introduce the challenges in CPS design and how they correlate to the areas of computer architecture and system programming.
- Provide a multidisciplinary overview of key topics in CPS design.
- Appreciate the CPS requirements of timeliness, composability and predictability and be able to apply them to the design of computing architectures.
- Teach essential skills to perform research and development in the CPS field.

## Detailed Course Description

Cyber-Physical Systems (CPS) are systems featuring a tight integration of computation with physical processes. Applications of CPS include new generation automotive systems, high confidence medical devices, avionics, smart power infrastructure, process control, distributed robotics, and others. All such systems exhibit stringent real-time and safety requirements: computation must progress in parallel with physical processes in the environment, and system disruption can lead to catastrophic consequences. Compared to traditional embedded systems, the vision for CPS calls for open, interconnected systems rather than closed, "black box" devices.

The goal of this course is to introduce the students to the key challenges, design methodologies and research directions in the field of cyber-physical systems, with an emphasis on the design of the underlying computing architecture. In particular, we will show how CPS requirements of predictability and reliability lead to significant changes in the hardware architecture, and we will study state-of-the-art solutions in this area. Due to the multidisciplinary nature of CPS design, the course will also touch on related topics, including: predictable operating system abstractions, timing analysis, modeling and verification.

## Course Outline

The following topics will be covered in class; the number of hours per topic and the detailed content is approximate.

### **Introduction to CPS (2h)**

CPS as the next evolution of embedded systems. Key issues in CPS design: timing predictability, verification and certifiability, integration and composability, modeling and abstraction, COTS components and time-to-market. Overview of topics covered in the course. Research opportunities in CPS. Introduction to course project.

### **Introduction to Real-Time Systems (3h)**

Task model. Quality of service. Interplay between timing properties and digital control. Basic schedulability results. The worst-case execution problem. The end-to-end delay problem.

### **Applications of Cyber-Physical Systems (2h)**

Overview of CPS applications. IMA (Integrated Modular Avionics) design. Issues in ARINC 653. AUTOSAR (AUTomotive Open System ARchitecture). Further examples of medical systems, power grid control, monitoring applications.

### **Predictable Computer Architectures (11h)**

Impact of architectural features on predictability. Controllable pipelines. Cache partitioning strategies. Scratchpad memories. Bus scheduling. Network-on-chips and real-time. Predictable memory controllers. Heterogeneous systems (FPGA, GPGPU).

### **Predictable OS Abstractions (5h)**

Overview of Real-Time OS and Hypervisors. Interrupt scheduling. Task partitioning. Hierarchical and component-based OS. Predictable task execution.

### **Timing and Performance Analysis (7h)**

Overview of static analysis methodologies. Measurement-based techniques. Cache, bus and memory analyses. Real-time queuing theory. Network calculus. Real-time calculus.

### **Introduction to Models of Computation and Verification for CPS (3h)**

Hybrid models. Timing verification. Hardware run-time monitoring. Tools and architectural description languages.

### **Project Seminar (3h)**

Project presentations in class.

## Prerequisites

Undergraduate knowledge of computer architecture (ECE429) and operating systems (ECE354/254), or consent of instructor. Previous exposure to real-time embedded systems is welcomed but not required.

## Reference Material

There is no required textbook. Papers and lecture notes will be provided on the course website.

Students with no previous background in real-time systems may consider one of the following text:

- J. Liu: Real-Time Systems. Prentice Hall, 2000.
- G.C. Buttazzo: Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications. Springer, 2004.

Lectures on performance analysis will use material from:

- J.Y. Le Boudec, P. Thiran: Network Calculus. Springer, 2001.

## Evaluation

- |                         |     |
|-------------------------|-----|
| • Lecture Participation | 5%  |
| • Seminar Presentation  | 10% |
| • Project Milestones    | 25% |
| • Project Presentation  | 10% |
| • Final Exam            | 50% |

**Lecture participation:** lectures will consist of a mix of instructor's presentations and discussion of research articles. Lecture participation is measured by physical presence in the class room and participation in paper discussion. Students will be assigned 2-3 papers to read every week.

**Seminar presentation:** each student will be asked to provide a formal seminar presentation during lecture. Each seminar presentation will cover one or two research papers and shall provide an introduction followed by sufficient details for the class to understand the motivation, problems, and main solutions provided by the authors. Presentations will be evaluated based on the quality of presentation material, clarity of exposition, and ability to critique the paper content. The instructor will accommodate student preferences for seminar topic and schedule whenever possible.

**Course project:** the course includes a project by each student on a topic of his/her choice related to the course content. Project ideas will be presented at the beginning of the course, but PhD and MASc students are especially encouraged to formulate new ideas or apply existing ideas in the CPS field to their own research. Group projects are acceptable if the size of the project warrants it and responsibilities within the group are clearly defined. A more structured project option will be offered to MEng students, albeit they are free to choose the research option. Each project will be evaluated based on: 1) intermediate milestones during the course, including a project proposal and literature review for the research option; 2) an in-class final project presentation, to be held during the last course lecture. The final exam will also incorporate project topics.

The instructor reserves the right to grade exams on a curve.

## Relevant University Policies

**Academic Integrity:** In order to maintain a culture of academic integrity, members of the University of Waterloo community are expected to promote honesty, trust, fairness, respect and responsibility. [Check [www.uwaterloo.ca/academicintegrity/](http://www.uwaterloo.ca/academicintegrity/) for more information.]

**Grievance:** A student who believes that a decision affecting some aspect of his/her university life has been unfair or unreasonable may have grounds for initiating a grievance. Read Policy 70, Student Petitions and Grievances, Section 4, [www.adm.uwaterloo.ca/infosec/Policies/policy70.htm](http://www.adm.uwaterloo.ca/infosec/Policies/policy70.htm). When in doubt please be certain to contact the department's administrative assistant who will provide further assistance.

**Discipline:** A student is expected to know what constitutes academic integrity [check [www.uwaterloo.ca/academicintegrity/](http://www.uwaterloo.ca/academicintegrity/)] to avoid committing an academic offence, and to take responsibility for his/her actions. A student who is unsure whether an action constitutes an offence, or who needs help in learning how to avoid offences (e.g., plagiarism, cheating) or about "rules" for group work/collaboration should seek guidance from the course instructor, academic advisor, or the undergraduate Associate Dean. For information on categories of offences and types of penalties, students should refer to Policy 71, Student Discipline, [www.adm.uwaterloo.ca/infosec/Policies/policy71.htm](http://www.adm.uwaterloo.ca/infosec/Policies/policy71.htm). For typical penalties check Guidelines for the Assessment of Penalties, [www.adm.uwaterloo.ca/infosec/guidelines/penaltyguidelines.htm](http://www.adm.uwaterloo.ca/infosec/guidelines/penaltyguidelines.htm).

**Appeals:** A decision made or penalty imposed under Policy 70 (Student Petitions and Grievances) (other than a petition) or Policy 71 (Student Discipline) may be appealed if there is a ground. A student who believes he/she has a ground for an appeal should refer to Policy 72 (Student Appeals) [www.adm.uwaterloo.ca/infosec/Policies/policy72.htm](http://www.adm.uwaterloo.ca/infosec/Policies/policy72.htm).

**Note for Students with Disabilities:** The Office for Persons with Disabilities (OPD), located in Needles Hall, Room 1132, collaborates with all academic departments to arrange appropriate accommodations for students with disabilities without compromising the academic integrity of the curriculum. If you require academic accommodations to lessen the impact of your disability, please register with the OPD at the beginning of each academic term.