Summary

A real-time system is subject to important temporal constraints. This course is about understanding where processing time is spent and what a designer can do in order to achieve real-time processing systems. Some solutions are Multiprocessors, accelerators, custom instructions, specialized hardware.

Content

During this course, response time measurements of interrupts are studied in laboratories, such as for example: the influence of dynamic memories, cache memories, compilation flags. Interrupts response time measurements, task commutations and synchronizations primitives are carried out on an embedded system based on an FPGA.

The course includes the study of embedded systems management models through polling, interrupts and using a real-time kernel with its task management and synchronization primitives.

Specialized programmable interfaces are implemented in VHDL to help with these measurements. A real-time kernel is studied and used during the labs. An acquisition system is implemented and the gathered data is transmitted by a Web server. To ensure the real-time acquisition and reading by the Web server, a multiprocessor system is developed and implemented on an FPGA.

An Accelerator designed in VHDL makes it possible to facilitate the optimization of functions through hardware on an FPGA. Cross development tools are used.

Each topic is treated by a theoretical course and an associated laboratory. The laboratories are realized on an FPGA board including a hardcore multiprocessor. A real-time operating system is studied and used with the laboratories.

Keywords

Real Time, FPGA, SOC, microprocessor, hardware accelerator, custom instruction, Real Time OS

Learning Prerequisites

Required courses

Introduction to computing systems, Logic systems, Computer architecture

Recommended courses

Embedded Systems, Real time Programming

Important concepts to start the course

Programmable Logic Architecture (FPGA), Computer Architecture, VHDL, C programming, Real Times basic knowledge (semaphor, synchronization)

Learning Outcomes
By the end of the course, the student must be able to:

- Design a multiprocessor system on an FPGA
- Analyze the performance of a real time embedded system
- Use design tools for SOC conception on an FPGA
- Implement a complete real-time system based on a multiprocessor design on an FPGA
- Test the realized system
- Defend the choices during the design phases

Transversal skills

- Set objectives and design an action plan to reach those objectives.
- Communicate effectively, being understood, including across different languages and cultures.
- Continue to work through difficulties or initial failure to find optimal solutions.
- Make an oral presentation.
- Write a scientific or technical report.

Teaching methods

Ex cathedra, laboratories and a mini-project

Expected student activities

- 3 groups of laboratories on specific topics, with a report by group for each of them, 3-4 weeks/topic;
- A final mini-project to practically synthesize the content of the course, with the design of a multiprocessor system on an FPGA, including for example a Web-server, a camera controller, a specific algorithm to be implemented in an FPGA hardware accelerator, 3~4 weeks for this mini-project

Assessment methods

Continuous control with reports and oral presentation
all labs 50% + final mini-project 50%

Supervision

Office hours  No
Assistants  Yes
Forum  Yes

Resources

Virtual desktop infrastructure (VDI)
No

Bibliography
Teaching notes and suggested reading material.
Specialized datasheets (ie.ex. FPGA et specific microcontrollers) and standards.

Notes/Handbook
Slides and documents on moodle

Websites

- http://fpga4u.epfl.ch
Moodle Link