EE-390(a)

Lab on hardware-software digital systems codesign

Atienza Alonso David, Peon Quiros Miguel

Cursus	Sem.	Type
Electrical and Electronical Engineering	BA6	Opt.
HES - EL	E	Obl.

Language of teaching	English
Credits	3
Withdrawal	Unauthorized
Session	Summer
Semester	Spring
Exam	During the semester
Workload	90h
Weeks	14
Hours	**3 weekly**
Practical work	3 weekly
Number of positions	

Il n'est pas autorisé de se retirer de cette matière après le délai d'inscription.

Summary

This course explores hardware-software co-design techniques to develop heterogeneous multi-core embedded systems running Linux on FPGAs. The course explores high-level synthesis tools (HLS) to design hardware accelerators that reduce total execution time and energy consumption for complex tasks.

Content

Architecture of embedded systems. SW-HW co-design
Architecture and SW-HW co-design of embedded systems. Coherence between custom HW modules and processor memory hierarchy. Integration of custom HW in the Linux operating system. Address translation between physical address space and virtual memory. Device drivers.
Tools and design flows
Division of tasks between SW and HW. Integration of HW components in a multi-core system. Simulation and on-chip debugging at the system level.
High-level synthesis (HLS) as a productivity booster for the design of HW accelerators. Fine-grained optimizations in HLS: loops, arrays, memory accesses and scheduling. Coarse-grained optimizations in HLS: dataflow model, tasks and scheduling into parallel workers.
Measurement of system performance, speed-up. Characterization of whole-system energy savings via dedicated hardware.

Keywords

Co-design SW-HW, embedded system, high-level synthesis, FPGA, Linux, device drivers.

Learning Prerequisites

Required courses

- Systèmes embarqués microprogrammés (EE-310)
- Systèmes numériques (EE-334)

Important concepts to start the course

- Architecture of embedded devices (microprocessors, peripherals, interrupts, DMA, etc.).
- Design of digital systems in VHDL, Verilog or SystemVerilog.
- Programming in C (arrays, pointers, memory management).

- Basic operating system (Linux) concepts.

Learning Outcomes

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

- Compose un système embarqué complexe
- Develop des composants matériels personnalisés et les intégrer
- Synthesize une spécification et une architecture à partir d'une idée abstraite
- Explain les fondations et les principes gouvernant le matériel d'un système digital embarqué

Teaching methods

Work on small projects that build up on learnt concepts. During the course, we introduce the required concepts to build step-by-step a platform composed of SW and HW components. Among other projects, we use HLS to build HW accelerators for convolutions that are later integrated into complete CNNs running on Linux.

Expected student activities

Participation in class sessions and delivery of weekly practical assignments.

Assessment methods

During the semester, continuous evaluation of weekly assignments will represent a 30% of the final grade. During class hours, the students will be asked (individually) to explain their exercises, either to the teachers or to the whole class. Over the last weeks of the course, a guided project covering all the concepts of the course will be completed in groups; the evaluation of this group project will account for 70% of the final grade. The code of all the assignments will be delivered via a git repository.

Resources

Moodle Link

- https://go.epfl.ch/EE-390_a