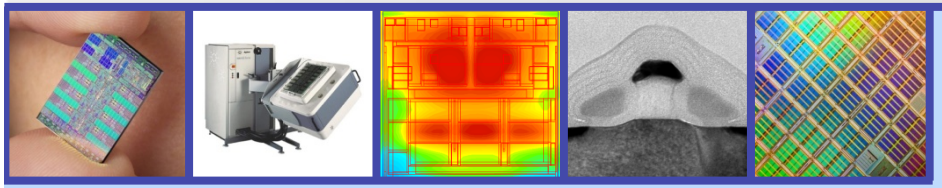
Practical Embedded Systems Engineering Syllabus for Graduate Students with Multidisciplinary Backgrounds

Bastian Haetzer
Gert Schley
Rauf Salimi Khaligh
Martin Radetzki

Embedded Systems Engineering / CSEE / U Stuttgart

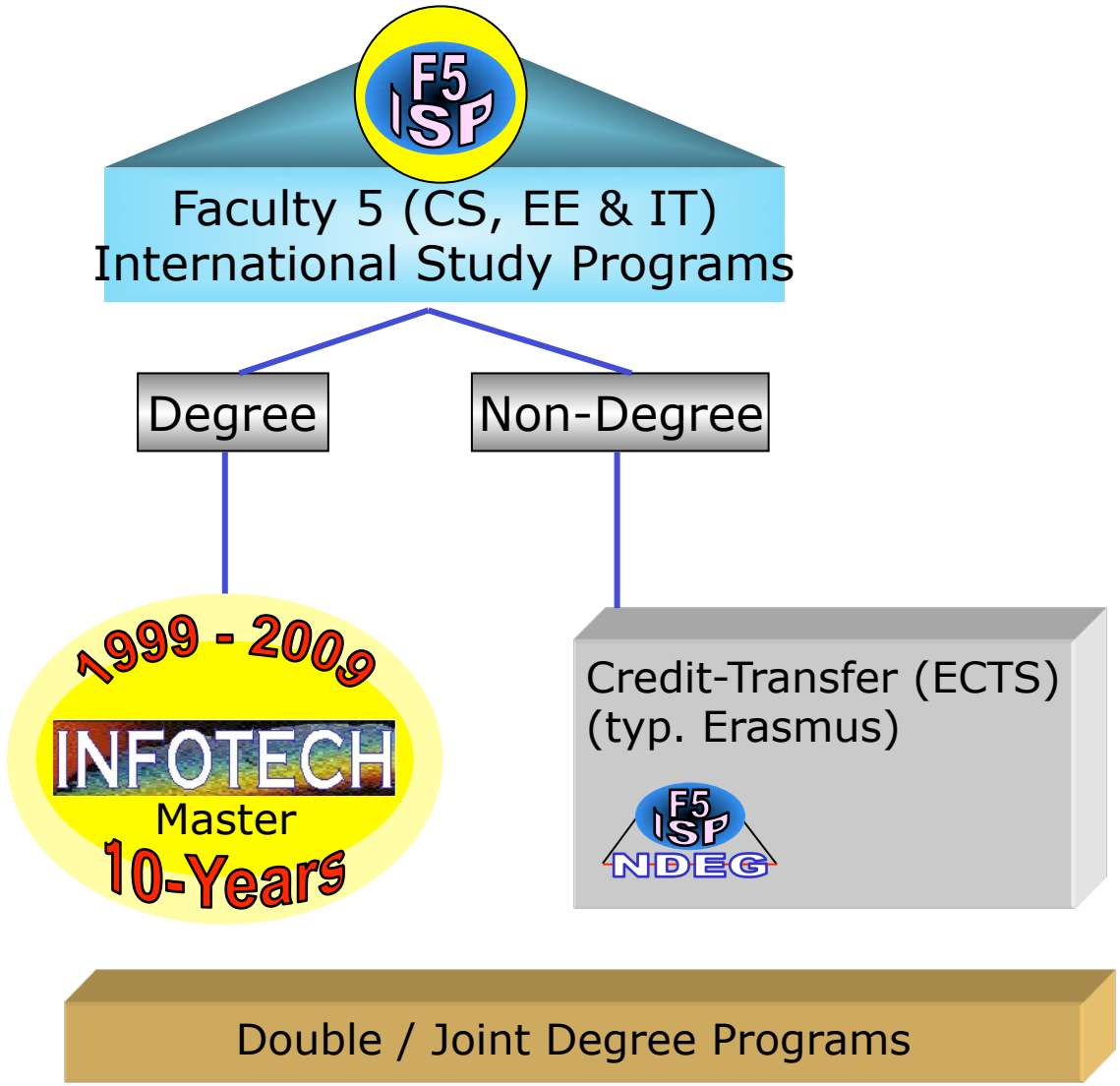


Universität Stuttgart

Overview

1. Context
2. Embedded systems courses
3. Practical (lab) course
4. Experiences
5. Conclusion

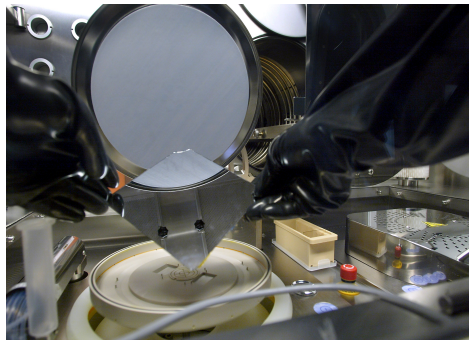
International Students



1999 - 2009 INFOTECH Master 10-Years

INFOTECH Master

- INFOTECH is a multidisciplinary (EE & CS) international master program
  - On top of a qualifying Bachelor degree
  - Leading to a Master of Science Degree (M.Sc.) in Information Technology
  - With a regular study time of 4 terms (3 teaching + 1 Master Thesis)
- Is truly international
- Is multidisciplinary
- 4 specializations (majors) can be selected
  - Communication Engineering and Media Technology
  - Embedded Systems
  - Micro- and Optoelectronics
  - Computer Hardware/Software Engineering



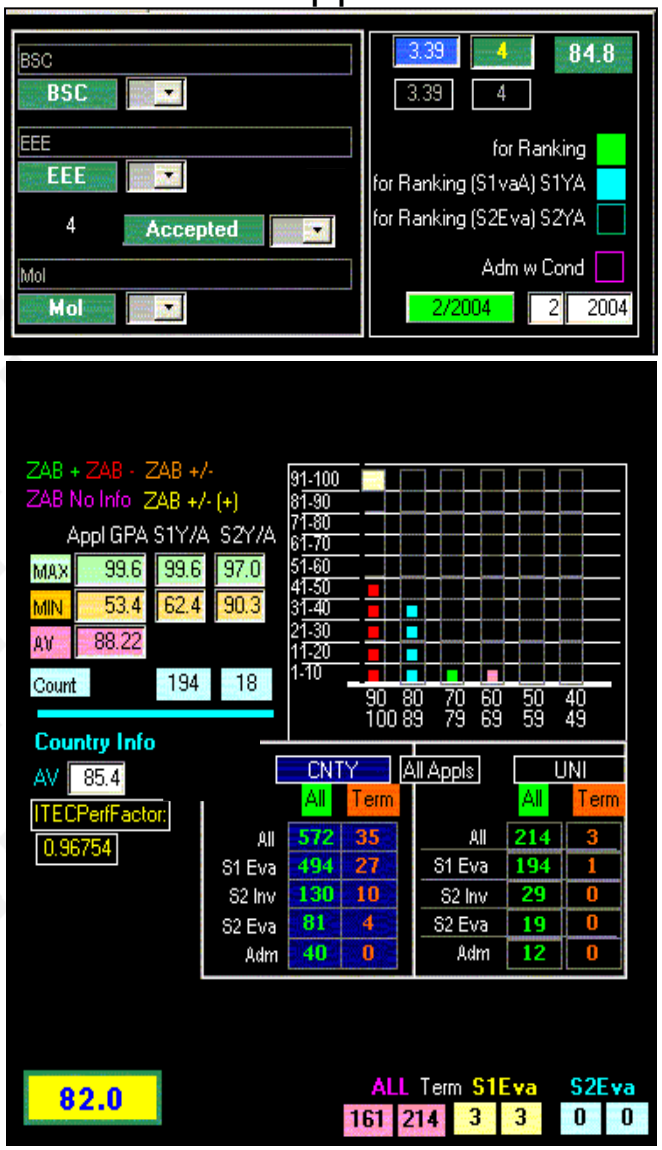
History and Facts

- has been set up in 1999 (is in its 12th year)
- Attracted some 9500 applications
  - Had applications from graduates of more than 1000 different universities
  - From 116 different countries
- Admitted some 600 students
- Conferred 310 Master Degrees
- Offers 4 majors
- Is supported by some 20 professors of the faculty (faculty total of 30), 10 professors from other faculties and some 5 industry lecturers
- Some 55 Lecture Courses
- Some 10 Lab Courses
- Some 10 Seminars
- Currently admits 80 students / yr from > 1200 applicants

Admission Selection

- Several “Competition Groups” (e.g. World, Partner, etc.)
- Selection mainly on GPA basis
  - Within group / University/ Country/ World
  - Based on comprehensive experience and reference data from previous applications
  - With weighted GPA (Itec Performance Factor)
  - And based on Ranking of applicant
  - And based on “Reputation of University” (Media, Own experience)

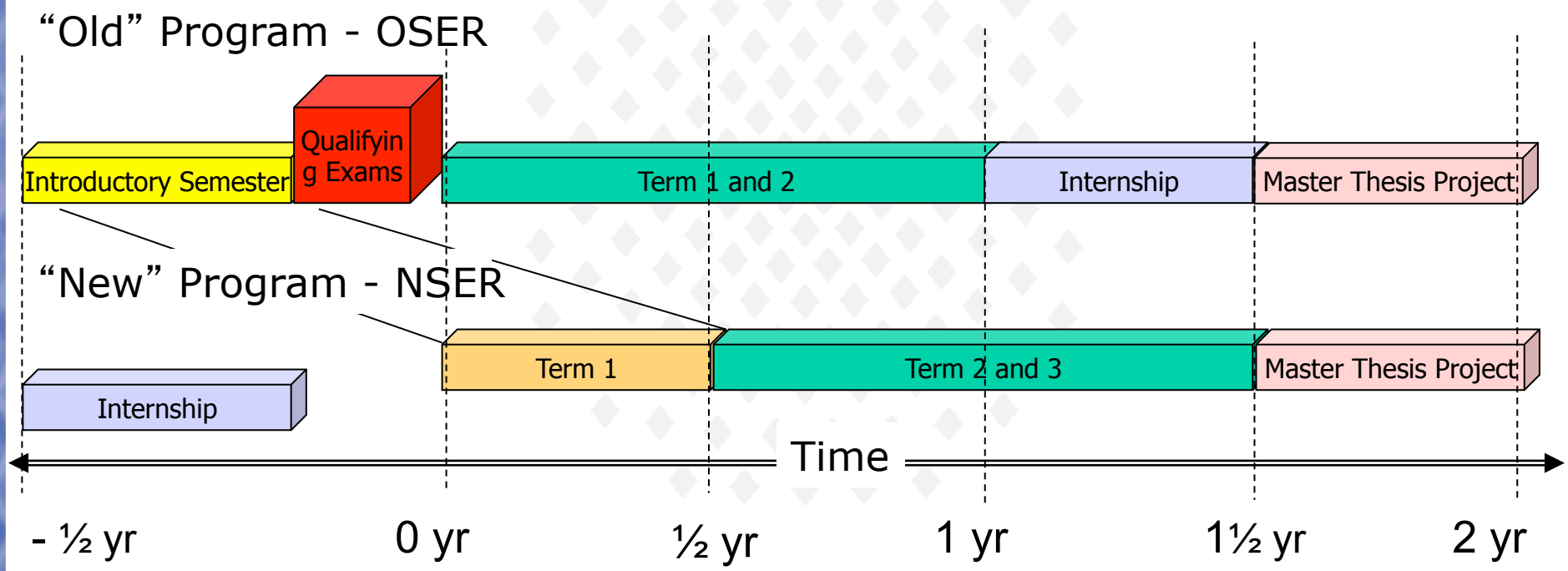
Part-view of Applicants Database



1999 - 2009 INFOTECH Master 10-Years

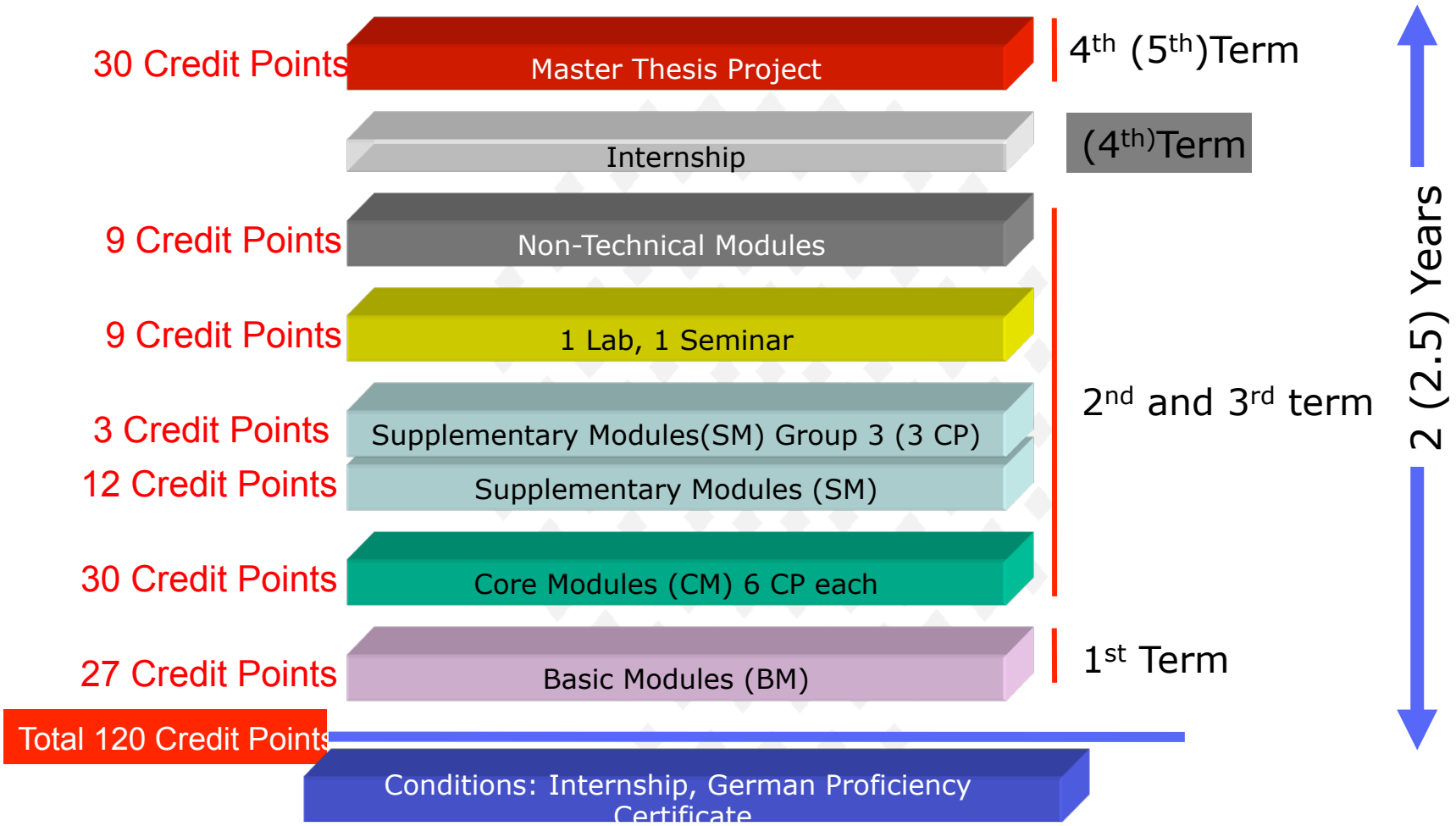
INFOTECH

the international Master's program in Information Technology



1999 - 2009 INFOTECH Master 10-Years

The Curriculum INFOTECH Master



1999 - 2009 INFOTECH Master 10-Years

Basic Modules (1st Term)

Basic Modules (BM)

- Advanced Higher Mathematics (AHM)
- "Container" Electronics and Communication (EC) with
  - Communications (Com)
  - System and Signal Theory (SST)
  - Radio Frequency Technology (RF)
  - Electronic Circuits (ElC)
- "Container" Computer Science (CS) with
  - Programming Languages (PL)
  - Computer Architecture and Organization (CAO)
  - Data Structures and Algorithms (DSA)
  - Operating Systems (OS)

Counseling Meetings

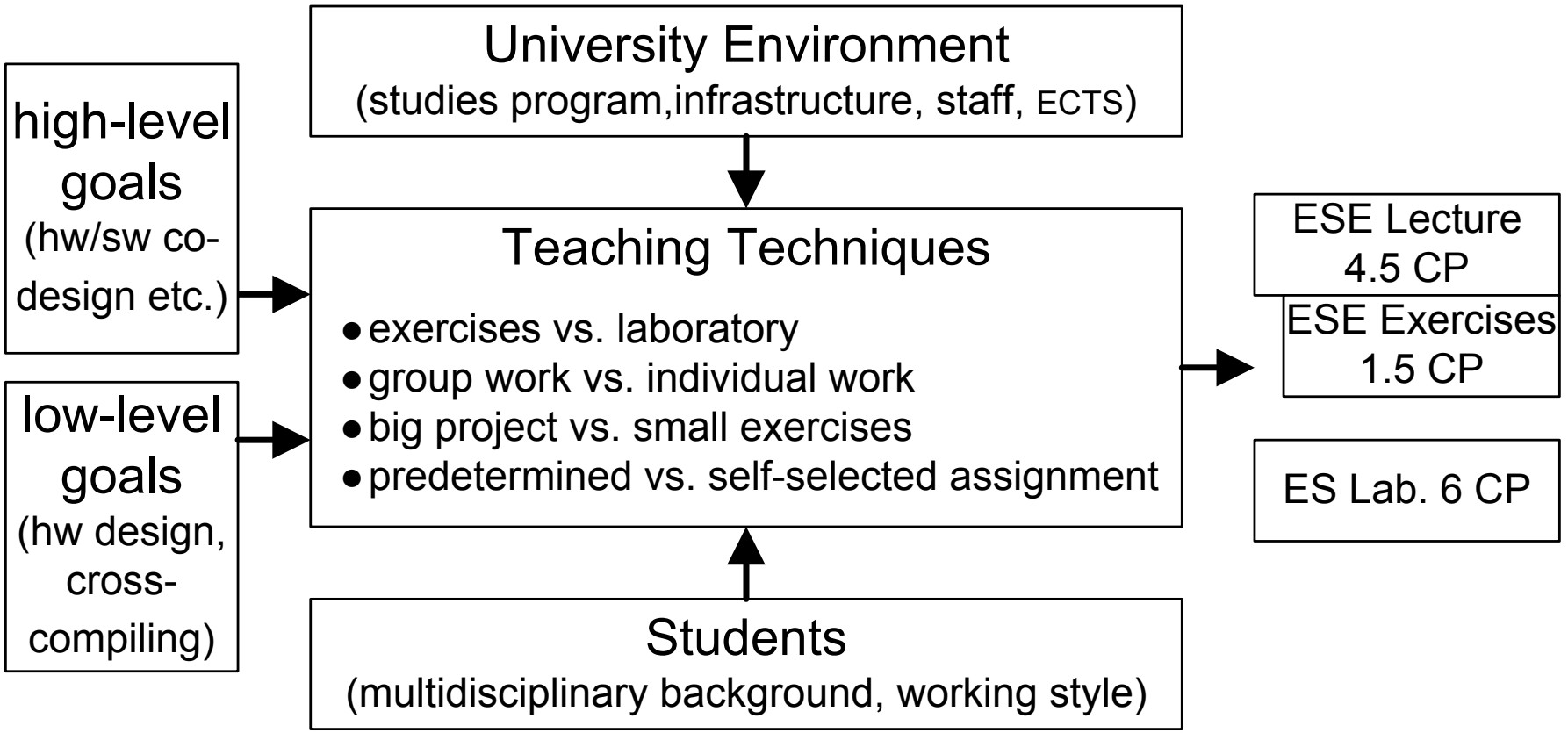
1999 - 2009 INFOTECH Master 10-Years

Embedded Systems Core Modules

Core Modules (CM): 5 out of 12

- Advanced Processor Architecture
- Communication Networks I
- Communications III
- Design and Test of Systems-on-a-Chip
- Distributed Systems
- **Embedded Systems Engineering**
- Industrial Automation Systems
- Intelligent Sensors and Actors
- *Modelling, Simulation and Specification*
- Real-Time Programming
- Software Engineering for Real-Time Systems
- Statistical and Adaptive Signal Processing

Syllabus design



ES syllabus focus areas

Systems-oriented
System specification
Modelling
Simulation
Performance estimation
Platform architecture

Lecture with exercises

Hardware-oriented
System specification
Modelling
Simulation
Performance estimation
Platform architecture

Lab

Software-oriented
Real-time analysis
RT programming, RTOS
Cross compilation
Remote debugging

Application-oriented
Multimedia | Automotive | Robotics | …

ESE lecture with exercises

Theory (exercises on paper, with ILP):
High-level synthesis: from algorithm to RTL,
Pipelined data paths and controllers,
Software scheduling: periodic and sporadic,
Software on bare CPU vs. RTOS, schedulability tests,
System architecture, storage and communication,
HW/SW integration

Practice (exercises in lab):

Task 1	Session 1	Processor-peripheral communication with memory-mapped I/O
Task 2	Session 2	Implementation of the interrupt handler
Task 3	Homework	Design of a frame-based scheduler
Task 4	Session 3	Implementation of a framed-based scheduler
	Session 4	

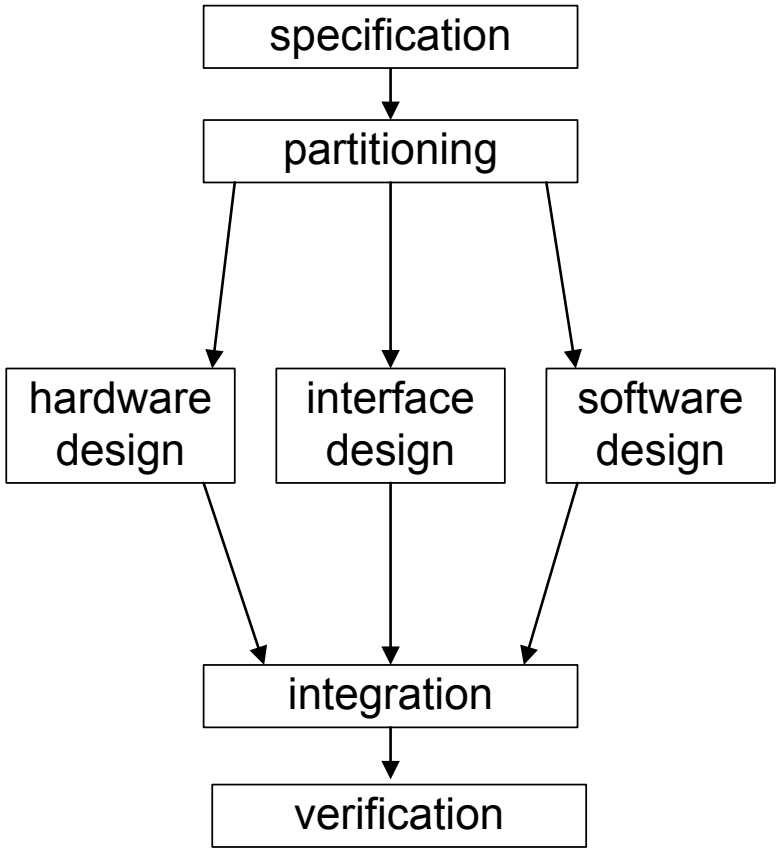
Embedded Systems Lab

Lab design trade-offs

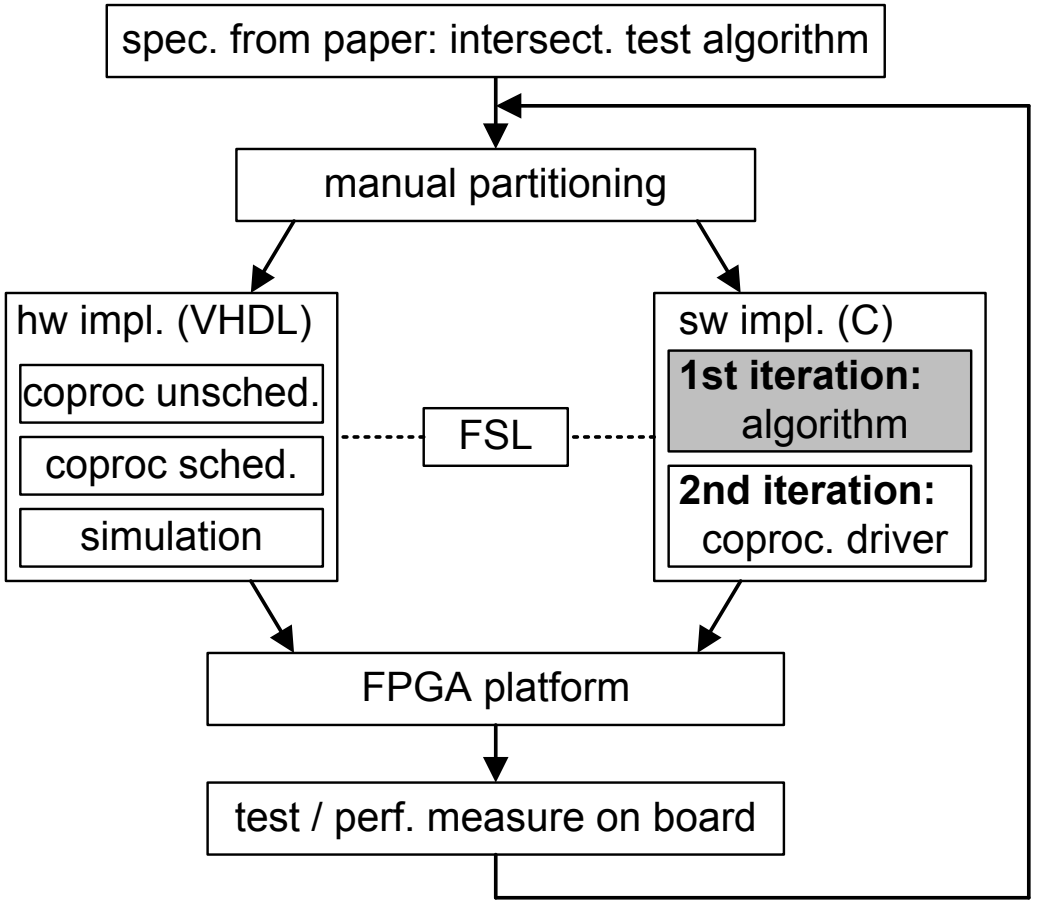
Lab experiments + guidance + focus – no big picture **Ex**	**Structured lab assignment** **Lab**	**Large project** (+) motivation + overview – getting lost
Working individually + grading + focus + individual advice **Lab**	**Working in groups** **Ex** + learn from each other – one does all the work + higher admission possible	**Project team** + real-world + soft skills – teaching costs
Fixed assignment + costs + focus – invites plagiarism **both**	**Individual assignments** + adaptation to interests – arbitration necessary – teaching costs	**Self-assignment** (+) motivation – risk – teaching costs

Design flow

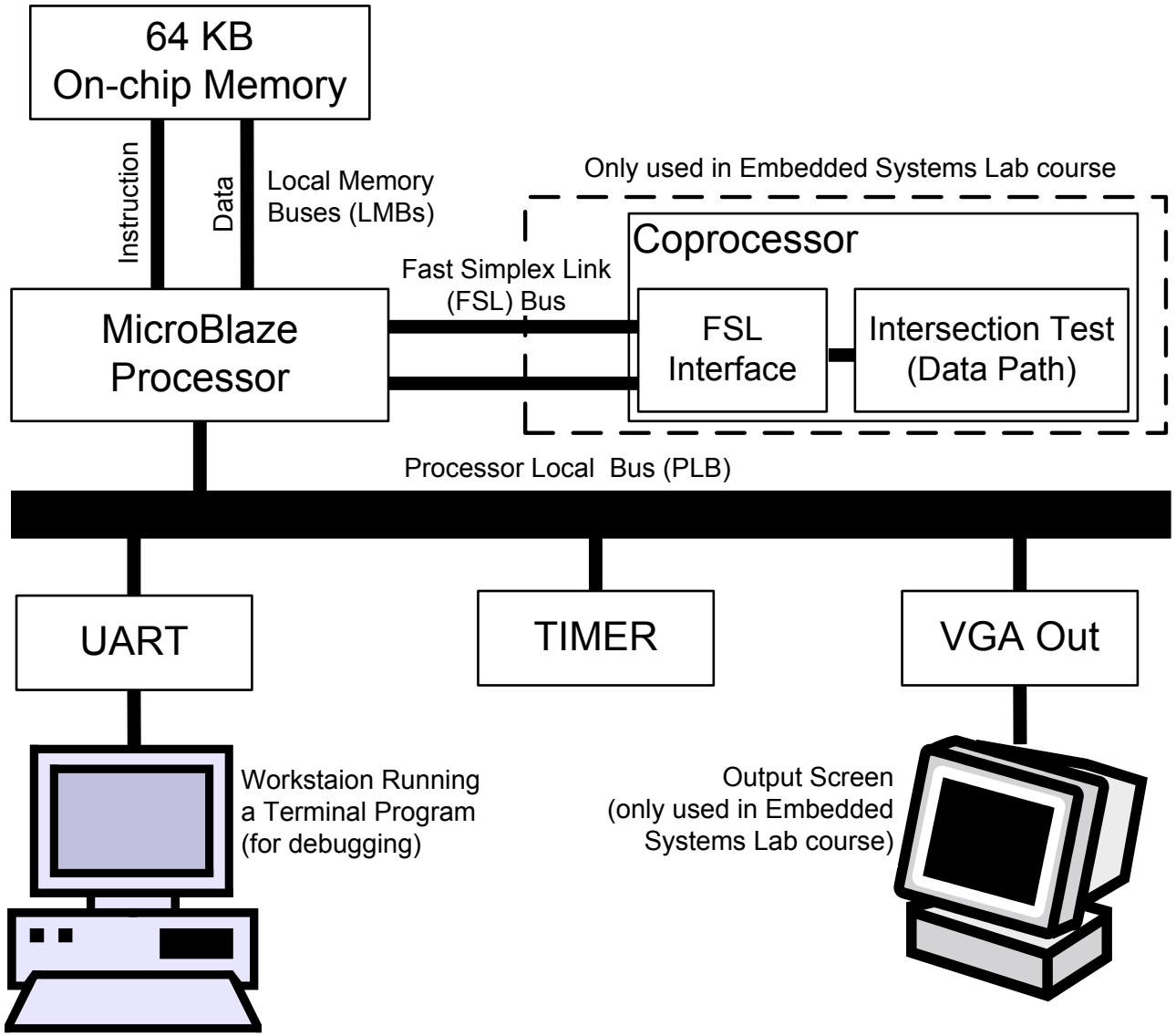
Canonical embedded systems design flow



Design flow in lab



FPGA platform



Initial try:
ARM 7/9 +
FPGA

Xilinx Univ.
Programme
boards

old: XUPV2P
Virtex-II

new: XUPV5
Virtex-5

Lab assignment (old)

Part 1	Session 1	Introduction to XPS + platform building and basic I/O
	Session 2,3	Software Implementation of algorithm + performance measurement
Part 2	Session 4-11	• Dataflow graph of algorithm • Unscheduled implementation of operators • Scheduling of operators • Datapath with scheduled operators • FSL-interface design • Integration of coprocessor into platform • Driver implementation • Performance measurement

Many students got lost in part 2 due to
- lack of overview, unstructured working style
- lack of basic knowledge, in particular in VHDL

Lab assignment (new)

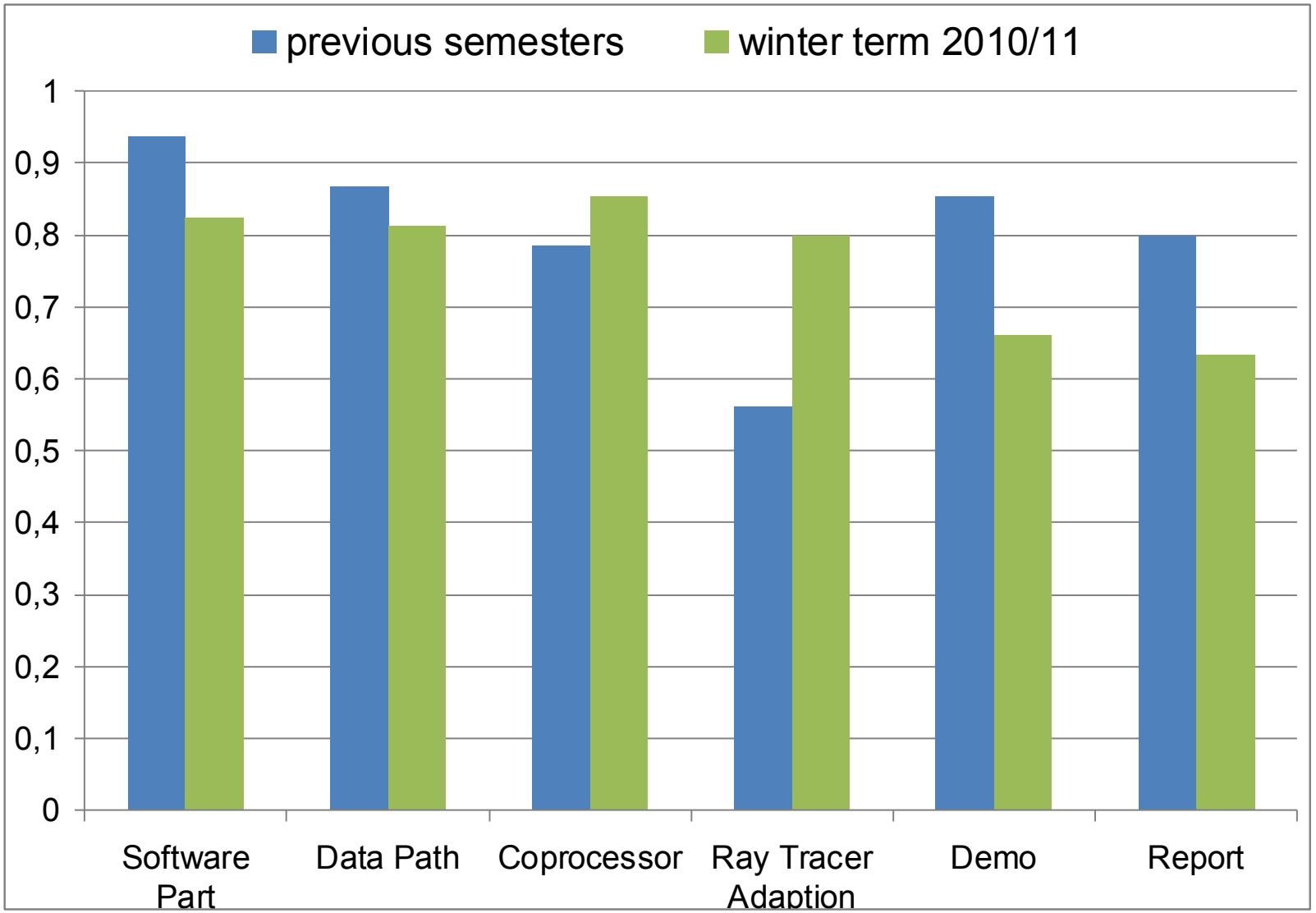
Admission requires entry test in C and VHDL

Part 1	Session 1	XPS Tutorial (Platform building + basic I/O)
	Session 2	Software impl. of algorithm + performance measurement
	Session 3	VHDL crash course (simple traffic light)
Part 2	Session 4,5	Dataflow graph of algorithm, unscheduled implementation + datapath testbench
	Session 6	Scheduling of operators e.g. cross product
	Session 7	Datapath with scheduled operators, simulation
	Session 8,9	FSL-interface design (state machine) + integration of coprocessor into platform
	Session 10	Driver implementation + performance measurement
Part 3	Session 11	System simulation + software enhancement (assembler)
	Session 12	Pipelining of FSL communication

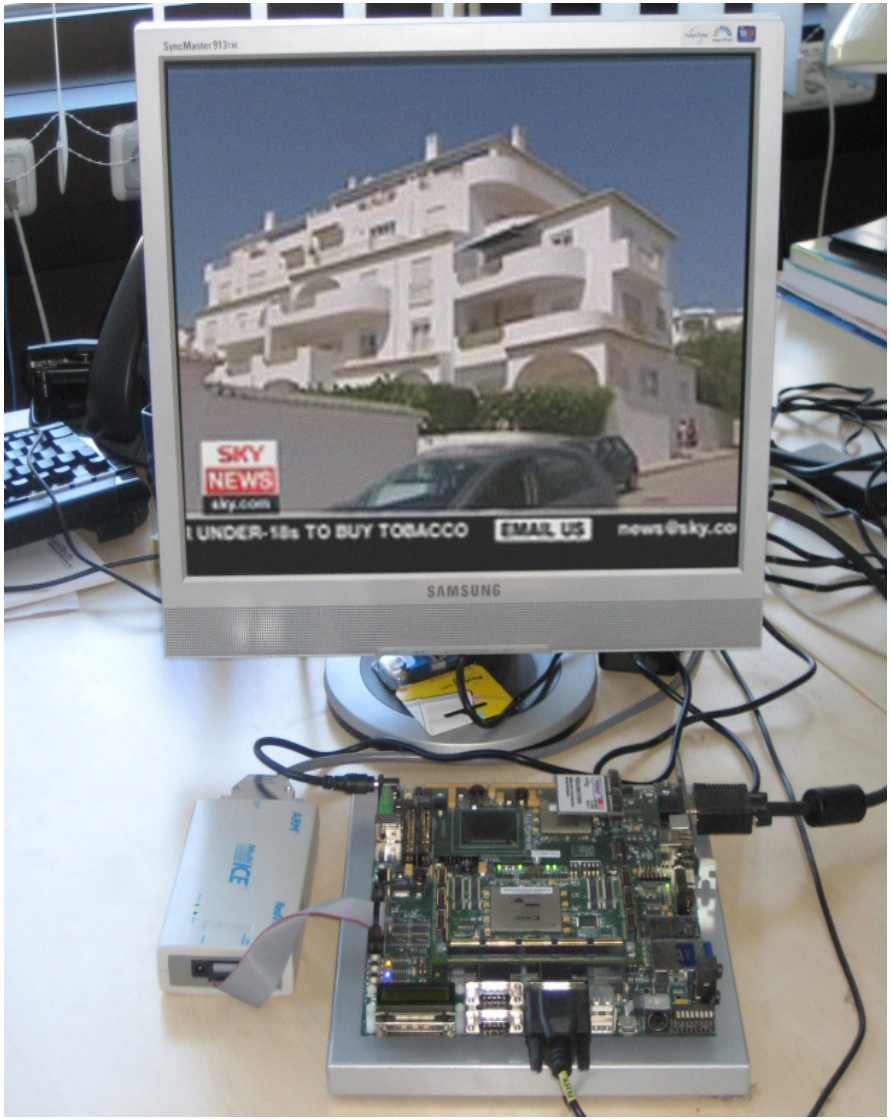
Deliverables

- Software code, structured and with comments
- VHDL code, structured and with comments
- Demo
  - “sales presentation“ of the results: speedup, FPGA area
  - verbal examination, questioning of design decisions
- Written documentation (datasheet)
  - data path documentation: IFs, structure, schedule
  - controller documentation: IFs, FSM diagram
  - SW integration documentation: FSL interface, register map
  - measurements documentation

Fraction of points



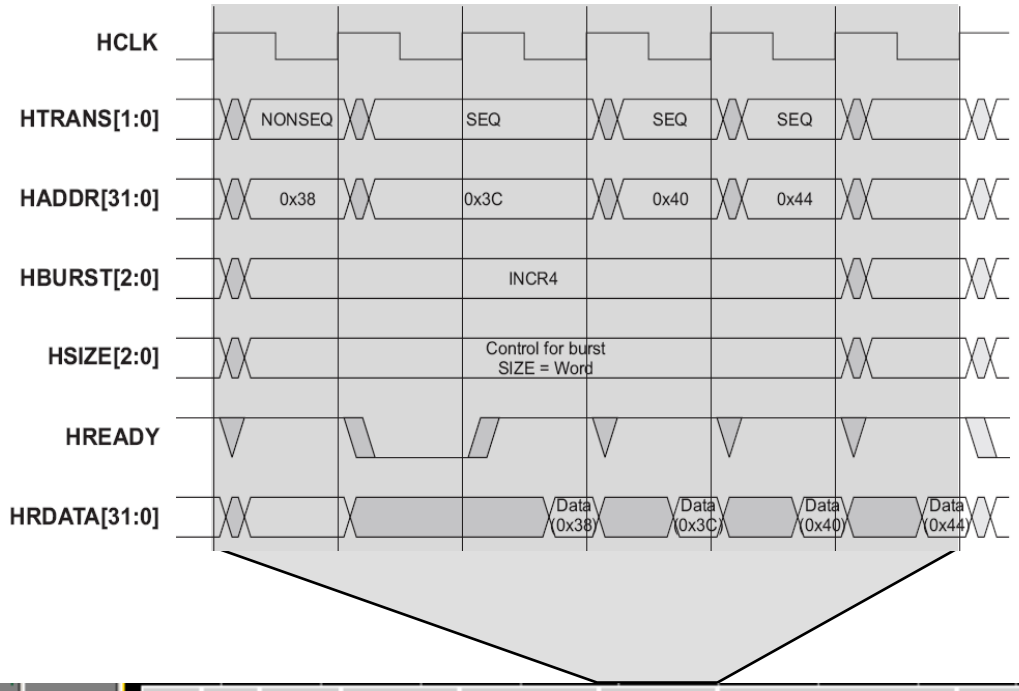
Research: embedded architectures

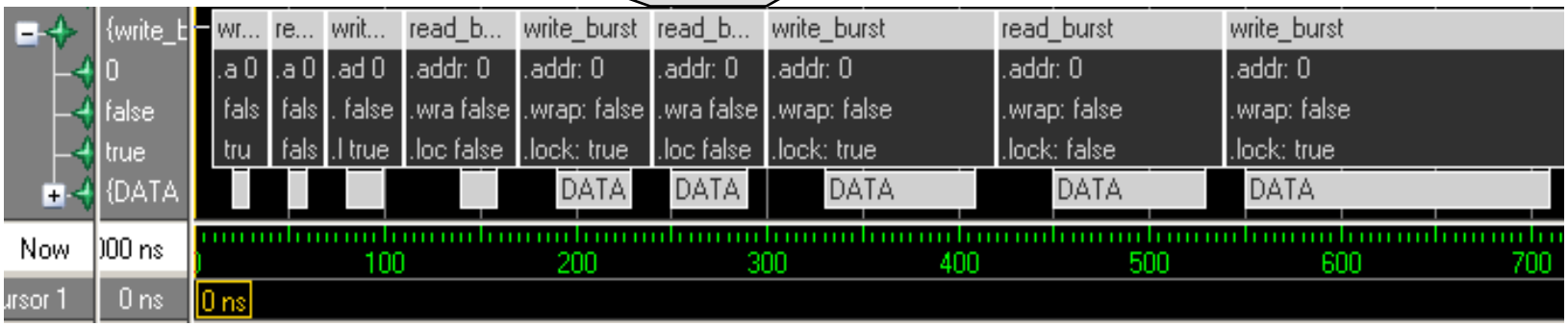


- Hardware/software architecture
- Many-core computing systems
- System communication
  - Bus subsystems
  - Networks-on-Chip
- Reconfigurable architectures
- Fault-tolerant architectures

Sample design (left): MPEG-4 H.264 video codec with ARM9 CPU and Xilinx Virtex5 FPGA to implement hardware accelerators

Research: design methods





- Modelling and simulation
- System-level fault simulation
- Analysis of properties such as reliability and robustness

Conclusion

- Embedded HW/SW design fundamentals, valid for any application
- Heterogeneity of students´ capabilites and backgrounds
- Monitoring results by qualitative and quantitative evaluation
- Monitoring the embedded market
- Continuous improvement of syllabus, courses