Challenges in Designing Embedded Systems Courses

Tulika Mitra

Department of Computer Science

National University of Singapore

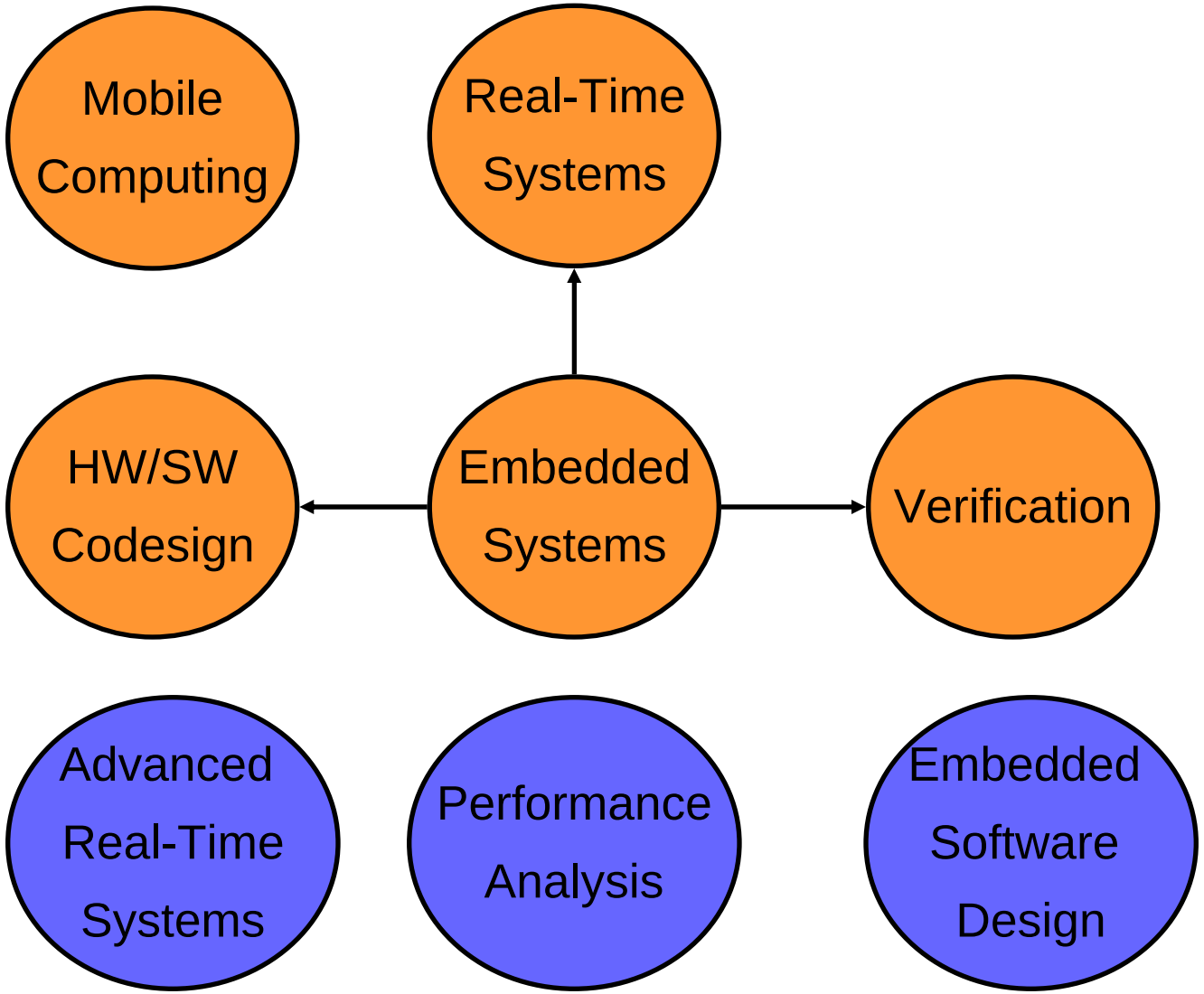
Computer Engineering Program

- Bachelor of Computing in Computer Engineering
  - Four year undergraduate program
  - Offered by department of computer science
  - Established in 2000
- Focus: Embedded Systems
- Motivation
  - Growing embedded systems industry in Singapore
- Goal
  - Graduate students with an integrated view of hardware-software design

Embedded Systems Curriculum

- Essential computer science modules
  - Programming languages
  - Data structures
  - Computer organization
  - Operating system
  - Computer architecture
  - Compiler
  - Networking
  - Databases
- Core and elective embedded systems modules

Embedded Systems Modules



Undergraduate Module

Embedded Systems

Breadth versus Depth

- A gentle introduction to embedded systems
- Pre-requisite for many other modules
- Depth: Overlap with follow-on modules
- Breadth: Shallow knowledge of each topic
- Syllabus
  - Hardware design with FPGAs
  - Processor, peripherals and interfacing
  - Programming with ARM
  - Real-time systems
  - System-level design

Learning curve

- Pre-dominantly CS students
- Little background in digital logic
- How to introduce hardware design?
  - Build on the strength of programming background
  - Use familiar language --- Handel-C
  - Make learning fun via real implementation --- FPGA
- Example lab exercises
  - Build a stack-based processor
  - Video game

Benefits

- Popular module for students
- Somewhat different from other CS modules
  - Lab exercises
- Provides opportunity to put the various standalone concepts (such as computer architecture, compiler, operating system, networking etc.) together and expose the big picture
- Future inclusion in CS curriculum?

Graduate Module

Embedded Software Design

Embedded Software Design

- Advanced module focusing on embedded software aspects
- Comprehensive understanding of unique design issues for embedded software
- Content
  - Embedded software development with ARM
  - Resource constrained compilation
    - Timing, power, area
  - Compilers for hardware acceleration

Research versus Industry Students

- Research students
  - More enthusiastic about reading papers
  - Lack of interest in hands-on exercises
- Industry students
  - Very enthusiastic with hands-on exercises
  - Difficulty in reading papers
- How to bridge the gap?
  - Convince each group about the importance of the hands-on exercises and reading research papers
  - Choose a middle ground with combinations of both

Projects and Exercises

- Hands-on exercise for each covered topic
  - Example: Code compaction on ARM/Thumb
- Diverse projects
  - Hands-on: Build cool applications
  - Research: Understand state-of-the-art and build on that

CS versus EE background

- Diverse background of the students
- Very little CS background for EE students
  - Compilation techniques are hard to explain
- No easy fix

Summary

- A worthwhile experience
- Graduated 300 students with embedded systems knowledge
- Teaching focus was instrumental in establishing the research group
  - 5 faculty members
  - 1 post-doc
  - 25 graduate students
  - SGD 2.5 million in funding

Current and Future Plans

- Education beyond modules: Special Interest Group in embedded systems
  - A group of undergraduates enthusiastic about embedded systems
  - Meets about once a week to discuss papers
  - Arrange seminars by industry experts
  - Participate in design contests
  - Participate in research projects

Wish list

- Inter-disciplinary nature of embedded systems poses unique challenges in designing curriculum
  - Exploit technological advances that raise the design complexity of higher abstraction layer whenever appropriate, e.g., C to hardware
- Standardization of curriculum across universities
  - Textbook
  - Exercises
  - Platforms and tool chains
- Lab infrastructure development
  - Amortize cost and effort across a set of modules