

CPS Education

(formerly known as the)

Breakout Session

Magnus Egerstedt, Rajesh Gupta, Peter Marwedel
(and Chris Gill)

The Charge

- CPS[r] has matured
 - No longer unclear what is CPS
 - A community has crystallized
- Time has come for CPS[e]!
- Can go Large or Small (Marwedel)
 - Small: Minor tweaks to existing courses/curriculum
 - **Large: CPS curriculum**

Absolutely Not

- *What is CPS?*

Questions/Discussion

- Should we do this?
 - EE/ME/CS degree with some other exposure, or a CPS degree
- How to do this?
 - Teach architecture principles (broadly) – reducible vs. irreducible complexity
 - Continuing education as well as the traditional undergrad through grad pipeline
 - Lifetime education, skills upgrade and essential part of this
 - Asymmetry of CS students taking EE and vice versa matters and needs to be addressed
 - Tailoring to origin disciplines rather than being entirely general-purpose
 - E.g., calculus is taught in Engineering schools for engineers
 - In some places (e.g., Lund) CS and EE students take more similar curricula, which helps (real-time, concurrency, controls, scheduling)
 - Hands-on projects are at the heart of CPS and motivate students from different disciplines working together on shared projects involving >1 discipline
- What should the components be?
 - Lone ranger (academic tendency?) vs. team development (industry tendency)
 - T-shaped education (breadth at the top, and depth)
 - Tetris model of educational “shapes”

Questions/Discussion

- What should the components be?
 - We have been teaching CPS based on elements selected from existing disciplines
 - CPS applications are now emerging that are not well served by the existing collection of elements
 - CPS education can help fill relevance/preparation gap
 - Long term objectives vs. transition period
 - Long term goal is to have experts educated specifically in CPS
 - How do we transition to that? Harvey Mudd style T-shaped education in CPS
 - How can ECE departments' experience help us with that?

Questions/Discussion

- What should the components be?
 - Freshmen come in with FIRST robotics experience
 - How can we leverage that? Students are driving the progression themselves (e.g., in capstone courses) based on their perceptions of employability
 - Can we shift ME design focus towards other disciplines?
 - Would making it more CPS help with that?
- How ensure relevance as opposed to shallow exposure to lots of stuff/disciplines?
 - How can we set up incentives for joint programs?
 - Can small companies/industry play a part?
 - Can we achieve this in academic settings?

Questions/Discussion

- How to do this?
 - Wrest control of calculus and teach it in your department in support of CPS education?
- Funding/dissemination mechanisms?
 - CPS VO as a sharing portal for syllabus, materials
 - Follow up with how to work with ACM/IEEE curricular standards (sub-cultural tensions/goals) CRA & ABET, too
 - Are there additional courses we need to offer, e.g., to support CPS concentrations (bottom-up from depts.)?
 - Shifts of emphasis in departmental courses (X for CPS)

Questions/Discussion

- How to do this?
 - Faculty advisors, CPS cohort, to guide students through existing departments, curricula, etc. (education *aspects*)
 - Teach signals and systems early?
 - Incentives and cost sharing for industry involvement
 - Regular interaction between industry and students
 - Industry intake of students is higher at the masters level than at the PhD level
- How ensure relevance as opposed to shallow exposure to lots of stuff/disciplines?
 - Engineering has become increasingly specialized, which in part poses the CPS education challenges we face
 - General engineering education requires modularity

Questions/Discussion

- How to do this?
 - Have students, faculty, industry, etc. gather to discuss “how are these courses relevant to me”
 - Combinations of topics to make a course (composition)
 - Modularity even within a syllabus (decomposition)
 - How do we integrate across disciplines? Everyone wants to teach their own courses – horizontal integration via common platforms used across courses by the students
- How ensure relevance as opposed to shallow exposure to lots of stuff/disciplines?
 - Invite people from diverse relevant companies to attend CPS Week workshop and let us know what we are and are not doing well in educating the students who will be heading their way

Questions/Discussion

- How to do this?
 - How to deal with disjoint backgrounds? Do it anyway
 - 200 students per year are coming through a combined program, so the reality is different than the perception of insurmountable barriers
 - Shared experience helps people learn from each other which at least makes them aware, helps them talk/work together from shared design/engineering perspective
- What should the components be?
 - Industrially relevant problems can help drive the curriculum transformation needed for CPS
 - Submit your projects to the VO?

Questions/Discussion

- What should the components be?
 - Multiple abstractions at different levels, each person can manage particular abstractions well, collaborate on others: choosing the abstractions well is necessary
 - Encapsulation of complexity, specialization of expertise, but perhaps along a different decomposition for CPS
 - How to modularize/abstract for CPS?
- How ensure relevance as opposed to shallow exposure to lots of stuff/disciplines?
 - Hopkins Engineering Design Teams, required diversity of years in school (seniors through freshmen) together
 - Students *seek out* courses to help their teams succeed

Questions/Discussion

- How to do this?
 - How to scale up ability of students to self-direct with specific objectives/trajectories in mind?
 - CPS oriented students take courses aimed at different career objectives, so the message they get about the entrepreneurial potential of CPS is essential
 - Combine with maker movement, CPS App Store
- What should the components be?
 - Need to address system maintenance as well as development, what can we do in teaching and research to address that consideration?
 - Entrepreneurship is key even at highly established places

Questions/Discussion

- What should the components be?
 - CPS in home health care, mobile sensing and actuation, robotics, etc. open up a wide range of new opportunities
 - Need to make sure people are educated in the full vertical abstraction stack for CPS
 - Students need to cultivate domain knowledge in the area of their interest (as well as breadth of understanding across CPS)

Rajesh Gupta

CSE 30 & CSE 125: “Initiation” Exercises

Basic I/O

P1: PushButton
& LED/LCD

P2: Potentiometer
& LED/LCD

Bus Control

P3: “Hello Chip!”
(I2C Bus)

P4: Event Capture
& Count/Measure

Touch

P5: Capsensing
& Display
& Count

P6: Music Synthesis
& Bar graph

USB

P7: Joystick Mouse!

P8: Logic Design

P9: Process Controller

Wireless and Motion

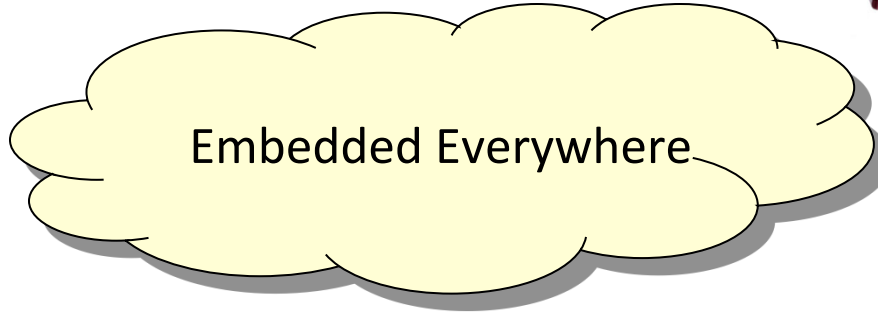
P10: Transceivers

P11: Motion Control

Rajev Alur

Education @ PRECISE

CPS PI Meeting, October 2012



Goal: To train next generation of engineers in fundamenals of embedded system design and implementation

Embedded Systems Masters Program

Intersection of Computer Science, Computer Engineering, and Control Systems

Industry:

Avionics, Automobile, Defense, Medical devices, Robotics

Topics

- Hardware-software co-design

- Real-time operating systems

- Integration of control, computation, and communication

- Safety-critical systems

- Model-based design, Specification, and Verification

Curriculum

Required Courses (4)

CIS541: Embedded Software for life-critical applications (Fall I)

CIS540: Principles of Embedded Computation (Spring I)

CIS 542: Embedded Systems Programming (Spring I)

ESE 519: Real-Time and Embedded Systems (Fall II)

Electives (6)

Embedded Systems Project / Masters Thesis

500-level CIS courses

CIS 501: Computer Architecture

CIS 505: Software Systems

CIS 553: Networked Systems

CIS 573: Software Engineering

500-level ESE courses

ESE 505: Control Systems

ESE 531: Digital Signal Processing

ESE 535: Electronic Design Automation

EMBS Status Report

- ❑ Launched in 2010
- ❑ First batch of 12 students graduated in Summer 2012
 - Majority found jobs at Embedded Systems Industry
- ❑ For Fall 2012:
 - Number of applicants: 240
 - Class size: 20
- ❑ Positive lessons
 - Students tightly integrated with PRECISE CPS research
 - Positive feedback on curriculum
- ❑ Challenges
 - Student demand primarily international
 - Generic software jobs far outnumber CPS industry jobs

Principles of Embedded Computation

- ❑ One-semester course for theory /science of CPS
- ❑ Aimed at upper-level undergraduate or Masters students with Computer Science / Computer Engg / Elec. Engg. Background
- ❑ Goal: Introduce formal models of computation, requirements, analysis techniques
 - Balance between depth and breadth
 - Emphasize rigorous mathematical thinking and reasoning
 - Maintain connection to real-world problems

PEC Course Structure

1. Synchronous Models
2. Safety Verification
3. Asynchronous distributed computation
4. Temporal logic and model checking
5. Dynamical systems
6. Timed systems
7. Real-time scheduling
8. Hybrid systems

Homeworks with theory problems +

Modeling projects in SPIN, NuSMV, Matlab, Uppaal, Simulink

See <http://www.seas.upenn.edu/~cis540/>

Magnus Egerstedt

Georgia Tech Robotics PhD

- 2 Colleges (CoC, CoE), 5 Schools (IC, ECE, AE, ME, ISyE)
- 5 Core Areas: Controls, Mechanics, Perception, AI, Autonomy
 - Largely based on existing courses
 - Students chose 3 out of 5
- Two new course packages
 - Introduction to Robotics (Boot camp)
 - Multi-disciplinary research course sequence

CPS-ED workshop at CPSWEEK

Jensen, Lee, Gupta, Egerstedt

- Focus: Concrete examples, instantiations, ideas pertaining to these topics
 - Build educational bridges across domains
 - Identify core concepts in relevant domains to establish CPS competency
 - Challenges (bureaucratic, intellectual, student backgrounds, ...) to actually pulling off a CPS curriculum