A Ball Goes to School -
Our Experiences from a Cyber Physical Systems Design Experiment
Steffen Peter, Frank Vahid, Daniel Gajski, and Tony Givargis

Motivation/Approach
Develop an educational program to teach CPS design to graduate Computer Science students.

Improve understanding of CPS design and sharpen attention in crafting solutions by teaching:
- Typical design flows for CPS design
- Importance of models and their limitations
- Introduction to state-of-the-art simulation and modeling tools

Apply small examples, that are:
- Easy to understand
- Possible to design and evaluate using a variety of tools and methodologies
- Is implementable in the lab

The Falling Ball Example
- A camera should take a picture of a falling ball that is dropped from a variable height.
- Sensors mounted above the camera detect the ball.
- A program in the cyber part of the system estimates the approaching time.

Benefits:
- Easily understood
- Need for precise timing
- Physical process needing mathematical modeling
- No perfect precision in cyber part
- Can be build in the lab

Modeling and Implementation
- Students modeled or implemented the system applying a range of tools (one per student)
- Progress, advantages and problems of the selected tools are discussed in the group

Results:
Discovered and discussed challenges:
- Math and modeling the physical system
- Separation of physical and cyber part
- Design methodologies of graphical design tools
- Selection of an appropriate Model of Computation
- Zeno behavior and simulation time resolution issues

Conclusions:
- The Falling Ball example is a suitable use case to teach CPS design
- Simplicity of the example allows students to focus on the actual CPS design challenges
- In four weeks (10h/week) students learned how to use tools, model the system, run simulations, test the system and evaluate the results
- Discovered design challenges are good support for lecture