EDAN85 (earlier coded EDA385) Embedded Systems Design - Continuation (Advanced)

Flavius Gruian
flavius.gruian@cs.lth.se
Tel: 0462224518
Office: 2125b, E house

Lecture 1 Contents

- Course organization
  - admission, purpose, goals
  - schedule, rooms
  - passing requirements
  - bibliography, pointers
- Getting started
  - project suggestions
  - available hardware and software

Admission

- Prerequisite: EDAN15 (Embedded Systems Design)
  - passed labs absolutely necessary
  - exchange students: **equivalent skills**
- Rule: **3 students/group**
- Practical constraint: admittance is subject to available resources (boards, computers)
  - Using an external (your own) board is OK (FPGA!)

Purpose

- **Theoretical (lectures)**: reveal more advanced notions and specialized subdomains in embedded systems
- **Practical (project work in the laboratory)**:
  - show the challenges of non-trivial projects, from idea to prototype
  - gain experience with toolchains, teamwork, report writing, presentation skills

Concrete Goals

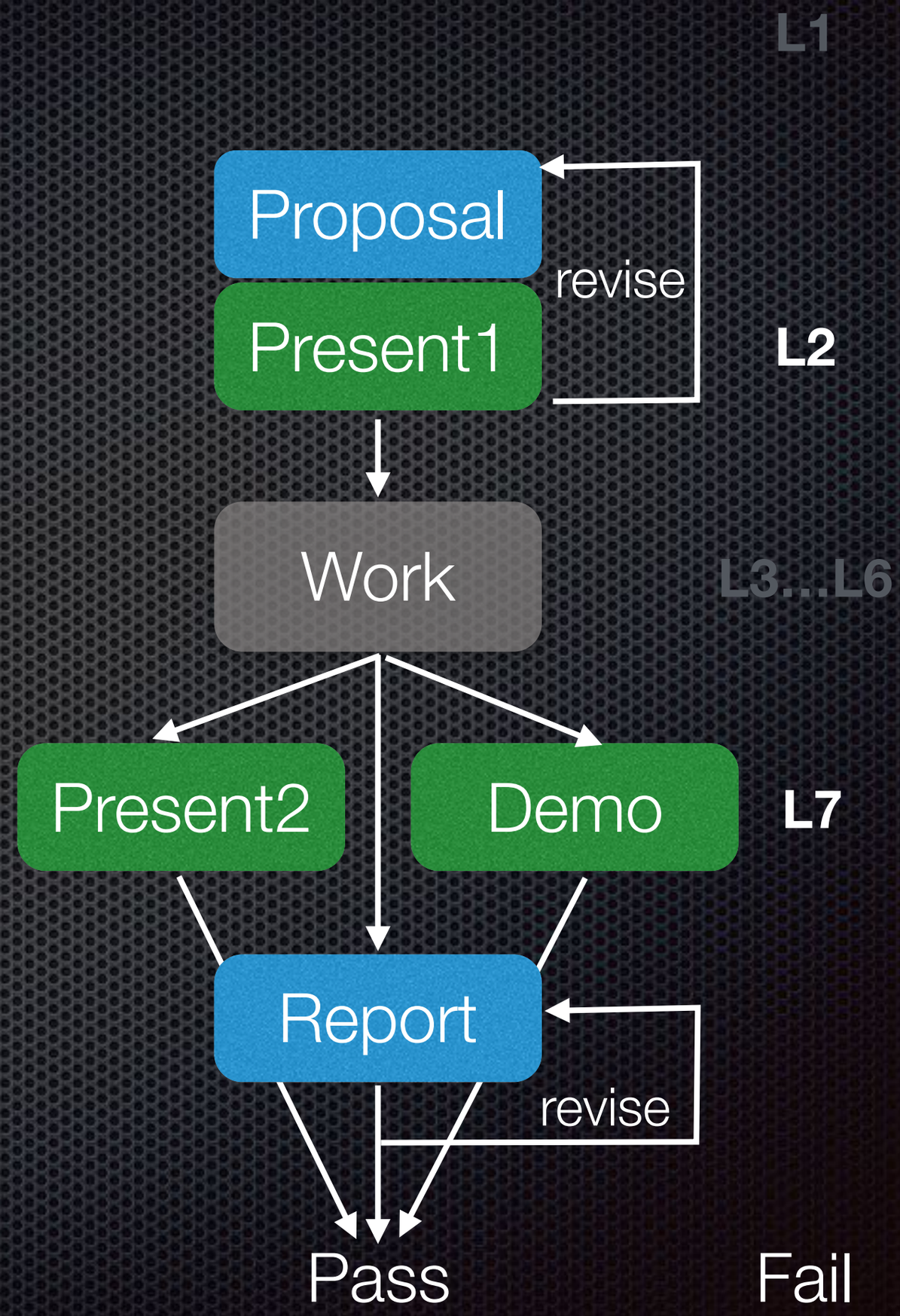
1. a fully working embedded system prototype
2. a report describing your work
3. experience with:
   - FPGA based prototyping (focus on Xilinx)
   - system design (Hw, Sw, integration)
   - selected standards/formats/protocols

Milestones

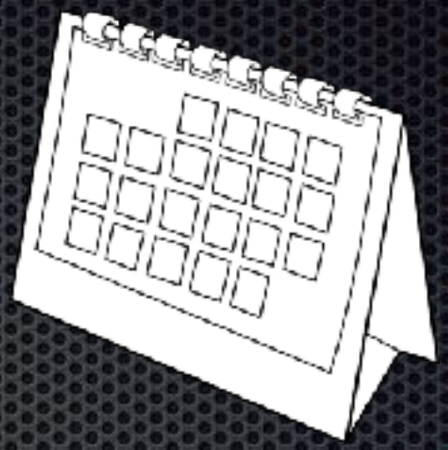
- project proposal (presented and revised)

work (during lab hours but mostly outside)

- final presentation
- demonstration
- final report (revised)



Schedule



- **Lectures**: Tuesdays 15-17, E:2116
  (5 lectures + 2 project presentations - other time for the last)
- **Labs**: Fridays 13-15, E:4116 (note the schema exceptions!)
  (no assignments, all is project work!)
- **Important Dates** (attendance compulsory):
  - proposal presentation: Tue, Sep. 5 (2nd lecture)
  - revised proposal handin: Mon, Sep. 11 (23:59)
  - final presentation: Fri, Oct. 13 (13:00-15:00)
  - demonstration: Fri, Oct. 13 (15:00-17:00)
  - final report handin: Sun, Oct. 22 (group based feedback, corrections later)

Lectures' Content

- L1: introduction, getting started
- L2: **you present** (+ partly on building whole systems)
- L3: real-life design trade-offs
- L4: software in embedded systems. testing and debugging
- L5: safety-critical systems
- L6: TBD (probably on IoT)
- L7: **your final presentation**

Grading

Pass/fail only! To pass:

1. proposal (presentation + revised) = 1hp
2. final presentation
3. demonstration
4. final report (with correction)

Total credits: 7.5 hp

Question: Who wants to be the student representative (“kursombud” D-line)?

Project Work

- groups of three (3) students (rare exceptions)
- separate projects (may be related)
- reasonable collaboration between groups is OK
  (exchange ideas, hardware IPs, software modules, etc.)
- EDAN15 environment, tools, architecture, lab
- To expect:
  - most work outside labs/lectures
  - a lot of individual reading/study
  - **7.5hp = 200h work = 4h+ of work/day !**

Support

- Boards:
  —> EIT labs (E:4116) has Nexys4 (Artix7) boards
  OR you (group) can borrow my boards full time!
  —> 10x Nexys3 (spartan 6) NOT SUPPORTED by Xilinx!
  OR use your own board if you wish (FPGA)
- Extension modules: borrow the ones I have (see website)
- Questions:
  Only during lectures, labs, and office hours (Wednesday, before lunch) … or by email anytime!

Pointers

- course web page (check it often!)
  http://cs.lth.se/edan85
- Xilinx page: http://www.xilinx.com
- recommended readings (not course books!)
  - Embedded Systems Design: A Unified Hardware/Software Introduction,
    Vahid, Givaris; John Wiley & Sons Inc., ISBN 0-471-38678-2
  - Embedded Systems: A Contemporary Design Tool,
    Peckol; John Wiley & Sons Inc., ISBN 978-0-471-72180-2

Lecture 1 Contents

- Course organization
  - admission, purpose, goals
  - schedule, rooms
  - passing requirements
  - bibliography, pointers
- Getting started
  - project suggestions
  - next steps

Questions?

The Project (= Labs)

1. Form groups (3 students)
2. Select the application (suggestions later on)
3. Decide the architecture (Proc, Peripherals, Custom IPs)
4. Model the system, if needed
5. Write software and hardware (use unit tests!)
6. Integrate, Test & Debug, go back to 5 (even 4, 3, …)

Project suggestions (I)

More challenging…

- accelerate a genetic algorithm/heuristic search: implement operators, fitness/cost function in Hw
- design and evaluate a Vector ALU for Microblaze: vector/matrix operations, compare to a Microblaze only system
- design a fault-tolerant Microblaze system: target transient faults, use dual/triple modular redundancy, checkpointing, etc.

Project suggestions (II)

- Arcade Video Game: Ping-pong, Tetris, Space Invaders, Guitar Hero, Bomber-man, Pac-man

  - outputs: custom VGA controller, sound controller
  - inputs: keyboard, mouse, gyroscope, microphone
  - software of varying complexity (interrupts, multi-threading,..)
- Sound processing: Echo, Visualizer, Graphic Equalizer, Synthesizer, etc.
  - microphone, speakers, display

Project suggestions (III)

- Head-to-head games: 5-in-line, Othello, ...
  - board-to-board communication (RF/USB/Ethernet)
  - simple AI/heuristic for gameplay
- Fine-grain process simulation: cloth/liquid dynamics, AI-life (flocks/schools/swarms) = hardware accelerated
- Security applications: encryption/decryption boxes, network traffic/bus/RF monitoring

Project suggestions (IV)

- an IoT edge device (and connect to a cloud service)
- Implement/evaluate/speed up useful libraries:
  - (partial) Message Passing Interface (MPI)
  - math functions, signal processing, vector operations
- Education helpers: remote FPGA configuration/ interaction for Nexys4 (web-based?)
- Your own ideas! (discuss them with me first)

Next steps…

- form your group
- brainstorm for project ideas and choose the best
- divide the work!
- include in the (written) proposal:
  - a work schedule
  - division of work (who does what, when)
- make the presentation (3-4 slides)
- start working!

Good Luck!