Hardware Platform Design Decisions in Embedded Systems
- A Systematic Teaching Approach -

Falk Salewski, Stefan Kowalewski
- WESE 2006 -
Motivation

- Embedded Systems = many different types of hardware platforms
  - CPU-based systems:
    - Microcontroller (MCU)
    - Digital Signal Processor (DSP)
    - Programmable Logic Controller (PLC)
    - ...
  - Programmable Logic Devices (PLDs):
    - Field Programmable Gate Arrays (FPGA)
    - Complex Programmable Logic Arrays (CPLD)
    - ...

- Platforms have several differences (internal structure, design process)
- Which platform is suited best? selection by expert in industry
- How we can provide students with a framework for a systematic hardware selection?
Lectures + exercises give an introduction in
- Hardware platforms
- Design processes

Lab course imparts practical knowledge and programming skills
- Different hardware platforms clarify (functional) differences

Remaining question: When to use which hardware platform?
Systematic hardware selection

- What we already teach
  - Properties of different hardware platforms
  - Requirements engineering for embedded systems
  - Design processes in embedded systems

- What we need in addition
  - Capability to systematically analyze both functional and non-functional features of hardware devices
  - Skills to map these features to the requirements of a given specified application
  - An approach including hardware and software issues
  - Consideration of the fact that available hardware devices are constantly changing over time
Teaching embedded systems (CS)

- Framework for systematic hardware selection at the end of the introductory course
- Next:
  - Presentation of our approach
  - Integration in the education
System requirements ⇔ hardware properties

- How to map hardware properties to system requirements?

- System Quality ⇐ HW attributes ⇐ HW properties (features)

- First step
  - Summarize all hardware attributes (functional and non-functional) which influence system’s qualities
  - Example hardware attribute: robustness of hardware platform

- Second step
  - Summarize all hardware properties which influence hardware attributes
  - Example hardware property: protective I/O circuits influence the attribute robustness of a hardware platform
First step: Hardware attribute tree

Hardware attributes influencing functional system properties:

- Dependability
  - Robustness
  - Reliability
  - Security
- Modifiability
  - Maintainability
  - Adaptability
  - Scalability
  - Configurability
- Reusability
- Testability
- Performance
- Functional range
- Marketability
- System qualities
  - Time-to-market
  - Cost
  - Mounting space
  - Power consumption
First step: Hardware attribute tree

Hardware attributes influencing non-functional system properties

Hardware attribute

- dependability
- modifiability
- reusability
- testability
- performance
- functional range
- marketability
- system qualities

Non-functional system properties

- robustness
- reliability
- security
- maintainability
- adaptability
- scalability
- configurability
- time-to-market
- cost
- mounting space
- power consumption
2nd step: Influencing HW properties

Hardware attribute

- dependability
  - robustness
  - reliability
  - security

- modifiability
  - maintainability
  - adaptability
  - scalability
  - configurability

- reusability
- testability
- performance
- functional range
- marketability

- system qualities
  - time-to-market
  - cost
  - mounting space
  - power consumption
HW properties $\rightarrow$ marketability (abstract)

- **Time-to-market**
  - General development effort
  - Expertise development team
  - External design support

- **Availability target HW**

- **Cost**
  - Cost target HW
  - Cost development environment
    - Cost compiler, simulator, debugging tools, ...
  - Cost programming and debugging hardware

Helps to identify how different platforms fulfill the requirement marketability for the desired application.
2nd step: Influencing HW properties

- Hardware attribute
  - dependability
    - robustness
    - reliability
    - security
  - modifiability
    - maintainability
    - adaptability
    - scalability
    - configurability
  - reusability
  - testability
  - performance
  - functional range
  - marketability
    - time-to-market
    - cost
    - mounting space
  - system qualities
    - power consumption
HW properties ➔ adaptability

- Adaptability (new reqs)
  - HW/SW dependencies
  - I/O capabilities
  - Integrated peripherals
<table>
<thead>
<tr>
<th>Factor influencing adaptability</th>
<th>CPU</th>
<th>PLD</th>
<th>Description, Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>HW/SW dependencies</td>
<td>X</td>
<td>X</td>
<td>Determines the effort necessary to transfer SW from one device to another (transfer is necessary if requirements cannot be fulfilled with the actual device)</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td></td>
<td><em>MCU families</em> ease migration from one MCU to another of the same family (often special migration notes available)</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td></td>
<td><em>Hardware abstraction/OS</em> could decrease dependencies between HW and higher SW layers</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td></td>
<td>If functionality is described in HDL (behavioral), a module can be transferred easily to any PLD suitable (new pin assignments necessary)</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td></td>
<td>If functionality is described in structural description, the module can be transferred easily only to hardware platforms with a similar structure (e.g. same basic elements)</td>
</tr>
</tbody>
</table>
# Details: I/O capabilities ➔ adaptability

<table>
<thead>
<tr>
<th>Factor influencing adaptability</th>
<th>CPU</th>
<th>PLD</th>
<th>Description, Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>I/O capabilities</td>
<td>X</td>
<td>X</td>
<td>Determines how easy new requirements with respect to I/O pins could be realized</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td></td>
<td>Usually, certain I/O functionality is mapped to a particular I/O pin. Some I/O functionalities can be mapped to different I/O pins. Few MCUs offer a free mapping of functionalities to I/O pins</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td></td>
<td>External busses (system bus, SPI, I2C, ...) ease the integration of additional external peripherals/memory in the system</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td></td>
<td>Usually, all I/O pins have the same properties/options (e.g. termination, voltage level, ...) Clock signals should be fed into the device via dedicated I/O pins</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td></td>
<td>Mapping of functionalities to I/O pins is done by SW. Pin assignment could influence amount of chip area used for the design</td>
</tr>
</tbody>
</table>
## Details: Integrated peripherals → adaptability

<table>
<thead>
<tr>
<th>Factor influencing adaptability</th>
<th>CPU</th>
<th>PLD</th>
<th>Description, Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Integrated peripherals</td>
<td>X</td>
<td>X</td>
<td>Multi purpose integrated peripherals increase adaptability</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td></td>
<td>Functionality of integrated peripherals can be determined via dedicated registers</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Integrated peripherals with a high number of options increase adaptability (and complexity)</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td></td>
<td>Almost all (digital) functionality is determined via software. Predesigned modules are available for common functions, written in HDLs (soft cores)</td>
</tr>
<tr>
<td></td>
<td>X</td>
<td></td>
<td>Hard wired peripherals (e.g. clock divider) can be used if available. Integrated peripherals usually do not include analog-to-digital</td>
</tr>
</tbody>
</table>
Selection process

- Requirements engineering process
  - Determine functional and non-functional requirements of the overall system
  - e.g.: robustness: ++ (very important)
        adaptability: +/- (nice to have)

- Analysis of different hardware platforms
  - How do certain HW platforms fulfill system requirements?
Selection process (Analysis)

- Analysis:
  - Analyse hardware properties of different platforms
  - Use this information to rate the corresponding hardware attribute for each platform
Selection process (Analysis)

- Analysis:

  - Robustness
  - Silicon structure
  - I/O capabilities
  - Package
  - Adaptability (new reqs)
  - H/W/SW dependencies
  - I/O capabilities
  - Integrated peripherals

 ➤ Analyse & rate different hardware platforms
Selection process (cont.)

- Requirements engineering process
  - Determine functional and non-functional requirements of the overall system
  - e.g.: robustness: ++ (very important)
  - adaptability: +/- (nice to have)

- Analysis (two step approach)
  - How do certain HW platforms fulfill system requirements?
    - Rate hardware attributes for each hardware platform
    - e.g.: HW1: robustness: ++ adaptability: +
    - HW2: robustness: +/- adaptability: ++

- Selection on basis of rating
  - Usually, trade offs are necessary (example in the paper)
Integration into ES education

- Integration at the end of the introductory course
  - Present hardware attribute tree
  - Develop structure „influencing hardware properties“ with students

- Advantages:
  - Development with students allows repetition/survey of lecture contents
  - Students learn a systematic selection process
Conclusions

- Need for educating systematic hardware platform selection
- 2-step approach for analysis
  - Hardware attribute tree
  - Hardware properties influencing these attributes
- This approach allows
  - Students with little experience in hardware platforms to understand and realize systematic hardware selection
    (use completed approach / use it as framework)
  - Repetition/survey of lecture contents if the development of the structures is done together with the students
    (develop approach)
- The approach is not device dependent and thus flexible for future devices
Future Work

- Evaluate and improve our approach in education
- Realize a web system which allows
  - An intuitive representation of the contents presented
  - Feedback from users
    - Students
    - Other institutes/universities
    - Industry
  - Integration of knowledge from companies developing/selling embedded hardware platforms
Hardware Platform Design Decisions in Embedded Systems
- A Systematic Teaching Approach -

Falk Salewski, Stefan Kowalewski
- WESE 2006 -