

Analyse alternativer Hardware-Architekturen eingebetteter Systeme im Bezug auf Sicherheit & Zuverlässigkeit

Aufgabenstellung

Eingebettete Systeme müssen eine Vielzahl von verschiedensten Aufgaben bewältigen. Einige dieser Aufgaben lassen sich dabei besonders gut mit CPU basierten Systemen (Mikrocontroller, DSP, etc.) lösen. Andere Aufgaben wiederum lassen sich besonders gut mit Systemen lösen, die aus rekonfigurierbarer Hardware bestehen (FPGA, CPLD).

HW/SW Codesign Ansätze beschäftigen sich damit, die Vorteile beider Hardware Plattformen in einem System zu integrieren.

In dieser Arbeit soll eine Funktionalität auf unterschiedlichen Architekturen implementiert werden. Dies beinhaltet Mikrocontroller, FPGAs und den sog. FPSLIC von ATMEL. Letzterer enthält sowohl einen Microcontroller als auch einen FPGA, wobei ersterer in C, letzterer in VHDL programmiert wird. Eine Co-Design Umgebung erlaubt gemeinsames Simulieren und Debuggen.

Im besonderen sollen in dieser Arbeit folgende Punkte geklärt werden:

- Welche Klassen von Anwendungen könnten von diesen Architekturen profitieren? Wie sehen die konkreten Vorteile aus (funktionale und nicht-funktionale Qualitäten)?
- Wie stark ist der Mehraufwand durch die Entwicklung einer Funktionalität auf zwei verschiedenen Hardwareplattformen (verschiedene Sprachen, Schnittstellen, etc)?
- Wie gut wird dieser Mehraufwand durch die Entwicklungsumgebung abgefangen (Co-Simulation etc)?

Beispielhaft soll dabei die (modifizierte) Aufgabenstellung des Praktikums implementiert werden.

Student

- Clemens Krämer

Betreuer

- [salewski](#)

Last update: 2009/06/11 13:13
lehre:abschlussarbeiten:analyse_alternative_hardware_architekturen https://embedded.rwth-aachen.de/doku.php?id=lehre:abschlussarbeiten:analyse_alternative_hardware_architekturen

From:
<https://embedded.rwth-aachen.de/> - **Informatik 11 - Embedded Software**

Permanent link:
https://embedded.rwth-aachen.de/doku.php?id=lehre:abschlussarbeiten:analyse_alternative_hardware_architekturen

Last update: **2009/06/11 13:13**

