6.13 ECA: Energy Consumption Analysis of software controlled systems

Marko van Eekelen (University of Nijmegen, NL)

Energy consumption analysis of software controlled systems can play a major role in minimising the overall energy consumption of such systems both during the development phase and later in the field. ECA proposes such an energy analysis, analysing both software and hardware together, to derive the energy consumption of the system when executing. The energy analysis has the property of being adjustable both to the required precision and concerning the hardware used. In principle, this creates the opportunity to analyse which is the best software implementation for given hardware, or the other way around: choose the best hardware for a given algorithm.

The precise analysis is introduced for a high level language, that covers the essentials for control systems. Hardware is modelled as a finite state machine, in which the transitions are function calls that are made explicit in the source code. In these model state changes correspond to energy consumption level. For that reason timing is added to the finite state machines. All the transitions and states in the hardware models are annotated with energy consumptions, to account both for time-dependent and for incidental energy consumptions. A prototype of the ECA system has been developed. It is applied to a small case study.

7 Panel discussion and Open problems

During the seminar we had a panel discussion and open problems session on resource bound analysis. Here is our summary of this session with contributions by Jan Reineke and Tom Reps.

Efficiently computed low and precise time bounds

The most fundamental open problem in the Timing-Analysis area connects the performance of architectures with the efficiency of timing-analysis methods and the precision of their results: how to design architectures that allow the efficient determination of precise and low execution-time bounds.

Multi-core architectures

While the timing-analysis problem for single-core architectures is solved, there is currently no practically usable timing-analysis method for multi-core processors.

Non-standard architectures

A completely open problem is how to determine safe execution-time bounds for GPUs, which are heavily used in computer-vision systems in cars, as well as in other types of accelerators. Similar challenges are given from heterogeneuos systems including different kinds of architectures, e.g. mobile phones, wearable devices, data centers, etc.