Energy consumption analysis of software controlled systems is crucial for minimizing the overall energy consumption of such systems both during the development phase and later in the field. ECA proposes an energy analysis that integrates both software and hardware together to assess the energy consumption of the system when executing. This analysis is adjustable in terms of both precision and the hardware used. Essentially, it enables the selection of the best software implementation for a given hardware configuration or vice versa.

The precise analysis is introduced for a high-level language that covers the essentials for control systems. Hardware is represented as a finite state machine, where transitions correspond to explicit function calls in the source code. These state changes are associated with energy consumption levels. Timing is added to finite state machines, and all transitions and states are annotated with energy consumptions, accounting for both time-dependent and incidental energy use.

A prototype of the ECA system has been developed and applied to a small case study.

### Panel discussion and Open problems

During the seminar, we had a panel discussion and an open problems session on resource bound analysis. Here's 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 is to efficiently design architectures that allow the 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 arise from heterogeneous systems including different kinds of architectures, e.g., mobile phones, wearable devices, data centers, etc.