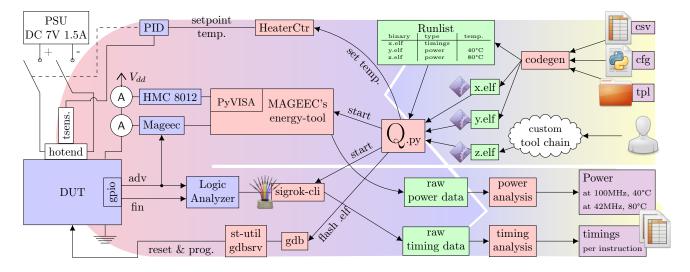
## Timing and power measurement infrastructure for embedded processors

Mark Kettner, Frank Oppenheimer Hardware/Software Design Methodology OFFIS – Institute for Information Technology Oldenburg, Germany mark.kettner@offis.de



Keywords—Embedded System, Power Management, Power and Timing Measurement

## I. Introduction

We present a framework that significantly reduces the effort for creating accurate energy/timing models for embedded processors covering different conditions. It supports the systematic collection of large amount of timing and power data needed to cover the complete microprocessors' ISA in different working conditions. Since manual measurements are tedious and error-prone we present an automated approach.

## II. The Demonstrator

Power consumption is measured at the target processors voltage supply using a MAGEEC board with modified firmware. It is operated by a patched version of pyenergy (MAGEECs own host software) sampling at 2.3kHz. Our patch enables pyenergy to also measure power consumption using a industry grade ammeter (HMC 8012). Further a PID controller is used to keep the target processor at a stable temperature above room temperature. Using the MAGEEC Board for timing measurements showed to be not reliable, hence timings are measured using a logic analyzer with a 4MHz sampling. But unfortunately this results in a small time frame of about 33s until the logic analyzer runs out of memory.

The software can be separated into three objectives: *code generation*, *experiment orchestration* and *power and timing analysis*. The code generation setup generates specially tailored binaries. The small time frame forced us to generate different binaries for either power (long repeating instruction sequences) or timings (short loop executions). These (and other) binaries can be queued into the experiment orchestration script (big Q.py) for execution on the target processor. It prepares the temperature and starts the desired measurement method (power or timing). The last step generates statistics for power and timings from the recorded data.

## III. Futher Use

This framework made power and timing measurements effortless and first tests produced sound results. We'll use this for verification and characterization of power models for embedded processors. Another use case might be verification of cycle accuracy since it helped us to further our understanding on which occasions stall cycles occur.

We would like to thank the German Federal Ministry for Education and Research (BMBF) for funding this research work within the project ACME 4.0 ("Acoustic Condition Monitoring Electronic Platform for Industry 4.0", reference code 16ES04).