Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institut für Technische Informatik
und Kommunikationsnetze
Computer Engineering and
Networks Laboratory

Prof. L. Thiele

Master Thesis:

Analysis and Optimization of Frequency Governors

The Background: Modern MpSoCs have energy consumption as a first level design constraint. Therefore, these platform support various hardware mechanisms to reduce power dissipation during low/no load. These mechanisms include Dynamic Voltage and Frequency Scaling (DVFS), and clock/power gating. These hardware mechanisms are typically utilized by frequency governors [1] to minimize energy subject to some response time goals. For-example, the Conservative Governor increases frequency in response to a persistent load; and generally prefers low frequencies. Due to this behavior, the conservative governor decreases energy consumption; consequently increasing the battery life of a battery powered device. On the flip side, the slow reactiveness, results in high response times. A complementary governor design is the Interactive Governor. In contrast to the Conservative Governor, the Interactive Governor responds quickly to increase in workload. This design results in low response time at the cost of higher energy consumption. As [1] states, there are several different governors; each with different energy, latency characteristics.

However, the governors have been designed basically based on heuristics. A theoretical analysis of the impact of governor design on the latency/energy consumption has not been conducted. The primary goal of this thesis is to fill this gap. We will study the impact of frequency on response-time and power dissipation of a set of applications running on a smartphone-like platform. The developed theoretical models will then be used to design *optimal* governors.

The Thesis: In this thesis, you will primarily be working with Dragonboard 810 development kit. As part of this thesis, you will understand how governors are designed and how the decisions made by the governors impact system and application behavior. Specifically, you will develop models for both governors (henceforth called Governor model) and applications (henceforth called application model). The governor model, along with the application model will be able to predict the response time and energy consumption of an application. As a last step, you will design a governor that attempts to optimize a certain metric (e.g. minimize energy) subject to certain constraints which can be specified (e.g. response time of application $\leq$ T).

What you will get? You will get yourself familiar with the advanced power management and application profiling techniques. Furthermore, you will have the opportunity to advance state-of-the-art governor design; leading to better system utility.

Requirements:
Courses: Embedded System Programming: C/C++, Python/Matlab

Interested? Please have a look at http://www.tec.ethz.ch/research.html and contact us!

Contacts

- Rehan Ahmed: rehan.ahmed@tik.ee.ethz.ch, ETZ G76
- Philipp Miedl: philipp.miedl@tik.ee.ethz.ch, ETZ G76

References

[1] CPU Governors, Hotplug drivers and GPU governors (accessed on 24.08.2017)
https://androidmodguide.blogspot.ch/p/blog-page.html