Ultra-Low Power: An EDA Challenge

Christoph Grimm, Javier Moreno, Xiao Pan
TU Kaiserslautern, Faculty of Computer Science
[grimm|moreno]@cs.uni-kl.de

***Abstract*—Visions such as the internet of things require vast amount of sensors distributed in our environment that strongly rely on circuits that are energy autonomous. However, design of such circuits is a challenge that is currently done by experts only. The challenge is to cope with circuit level design and even technology while designing an application. Unfortunately, tools and methods that support cross-layer and cross-domain optimizations are missing.** [1]

Keywords—ultra-low power, cross-layer optimization

I. Power Optimization: a Cross-Layer Challenge

Power (energy) aware design is a key challenge for mobile or autonomous systems. Approaches to reduce power consumption range from technology to system level design including sofware- and application development.

At technology level, we have a number of options to reduce power consumption at the cost of speed. Options include sub-threshold circuits, adjustable voltages for substrate and V_{DD}, or technologies/devices with lower power consumption (e.g. High-K). At architecture level, we have to aim at providing infrastructure for adjusting the infrastructure to adapt the "service" of a circuit to the needs of an application. Options include power gating, dynamic frequency/voltage scaling (DVFS), clock gating. In particular power gating is known to be very efficient by also reducing static power. At system level, we must match strategies such as "Run Fast Then Stop" (RFTS) or DVFS with timing of all components in order to enable power gating to switch off components as long as possible.

Power optimization becomes a cross-layer and cross-domain issue because of the contrary impact of the different layers: a (slow) low power technology could in the end lead to increased need for energy if it leads to shorter power down for components. Hence, system design requires careful evaluation of all options considering impact across layers and domains.

II. Approaches for Low Power Design

To achieve low power, power management infrastructure at device- and architecture level and run-time management are interactively tuned.

A. Virtual Prototyping and Power Profiling at Design Time

For power optimization, virtual prototypes are used to analyze a very limited number of usage scenarios. Adding power estimation e.g. by power state machines to the virtual prototype of the overall system increases awareness of overall power and energy efficiency during co-design of hardware and software (e.g. [1]). Furthermore, power profiling based on scenarios helps finding tradeoffs between different available design parameters and to fine-tune power management and firmware.

B. Power Optimization at Run Time

Power optimization at run time is done at different layers of software. Best places to manage power are operating system, drivers of peripheral components, and application software. Manual optimization relies on scenarios assumed during design time; more dynamic techniques are discussed in [2]. Still an issue is safety/security and dependability: Applications must be able to announce need for resources, while not being allowed to drain too much power. Dynamically changing scenarios, and peripherals in evolving systems are challenges not yet solved sufficiently.

III. Challenges for EDA

Efficient use of existing technologies at higher layers offers tremendous potential for improvements that is not yet used sufficiently. Current tools offer means to evaluate and manually tune designs, but do not offer automatic optimizations that cross layers of implementation or domains, e.g. to include also RF/analog design.

Tools for design automation focus on too isolated design issues and rely too much on modeling/simulation as only mean to cross-domains and layers of implementation. A new generation of EDA tools is needed that allows designers to automatically optimize systems in a more holistic way, combining options from device and architecture level up to system and application level, including in particular also mixed-domain (analog, RF, micro-mechanic harvesters), and also considering less main-stream design alternatives such as re-configurable processors instead of software.

References

[1] J. Moreno, J.G.O. Wenninger, J. Haase, C. Grimm, Energy Profiling Technique for Network-Level Energy Optimization, in Proceedings of the 10th IEEE Africon 2011, 2011.

[2] K. Kluest, V. Handziski, C. Lu, A. Wolisz, D. Culler, D. Gay, P. Levis: Integrating Concurrency Control and Energy Management in Device Drivers. Proceedings of SOSP07

[1] 978-3-9815370-0-0/DATE13/©2013 EDAA