Embedded Reconfigurable Architectures

Stephan Wong
Delft University of Technology
The Netherlands
J.S.S.M.Wong@tudelft.nl

Luigi Carro
Universidade Federal do Rio Grande do Sul
Brazil
carro@inf.ufrgs.br

Stamatis Kavvadias
University of Siena
Italy
kavadias@dii.unisi.it

Georgios Keramidas
Industrial Systems Institute
Greece
keramidas@isi.gr

Francesco Papariello
ST Microelectronics
Italy
francesco.papariello@st.com

Claudio Scordino
Evidence s.r.l.
Italy
claudio@evidence.eu.com

Roberto Giorgi
University of Siena
Italy
giorgi@dii.unisi.it

Stefanos Kaxiras
Uppsala University
Sweden
stefanos.kaxiras@it.uu.se

ABSTRACT

In current-day embedded systems design, one is faced with cut-throat competition to deliver new functionalities in increasingly shorter time frames. This is now achieved by incorporating processor cores into embedded systems through (re-)programmability. However, this is not always beneficial for the performance or energy consumption. Therefore, adaptable embedded systems have been proposed to deal with these negative effects by reconfiguring the critical sections of an embedded system. In these proposals, we are clearly witnessing a trend that is moving from static configurations to dynamic (re)configurations.

Consequently, the proposed embedded systems can adapt their functionality at run-time to meet the application(s) requirements (e.g., performance) while operating in different environments (e.g., power and hardware resources). Besides processor cores, we have to deal with memory hierarchies and network-on-chips that should also be (dynamically) reconfigurable. Furthermore, the interplay of these components is increasing the design complexity that can be only alleviated if they can self-optimize.

In this tutorial, we will present and discuss several strategies to perform the mentioned dynamic reconfiguration of the processor, memory, and NoC components - together with their interaction. We will review and present the state-of-the-art for the design of each component that allows for a gradual selection of design points in the trade-off between performance and power. Finally, we will highlight an open-source project that incorporates many approaches for dynamic reconfiguration in both actual hardware and simulation accompanied by the necessary tools.

Categories and Subject Descriptors

C.3 [**Special-Purpose and Application-based Systems**]: Real-time and embedded systems

CASES'12, October 7-12, 2012, Tampere, Finland.
ACM 978-1-4503-1424-4/12/09.

General Terms

Design, Performance, Reliability

Keywords

Compilers, Dynamic Reconfiguration, Embedded Systems and Applications, Memory Hierarchies, NoC, Processor Design, Simulation

1. LIST OF TOPICS

In this tutorial, we will discuss the following topics:

- **Introduction: Need for (dynamic) adaptability?** This part will provide a general introduction to adaptability within embedded systems from the applications and hardware design perspectives. It will be argued that dynamic reconfigurability is the next stage in embedded systems design.
- **Heterogeneous behavior of the applications and systems** This part will show, using examples, how the behavior of even a single thread execution is heterogeneous, and how difficult it is to distribute heterogeneous tasks processes among the components in a SOC environment, reinforcing the need for adaptability. This is furthermore complicated when considering multiple threads and the existence of an embedded operating system.
- **Adaptive and reconfigurable processor architectures** This part will present an overview of adaptive and reconfigurable processors and their basic functioning. We will also discuss those which present some level of dynamic adaptability.
- **Reconfigurable memory hierarchies** This part will present the need for sophisticated memory hierarchies to deal with varying applications and present techniques how the organization of these memory components can be "morphed" when switching applications without much performance loss.

- **Communication architectures - NoCs** This part will discuss how important NoCs are for future embedded systems, which will have more heterogeneous applications being executed, and how the communication pattern might aggressively change, even with the same set of heterogeneous cores, from application to application.
- **Tools (compilers and simulators)** This part will discuss the need for a new breed of tools that are needed to support the earlier mentioned approaches.
- **Putting it all together** An analysis on the aforementioned techniques: how they can work together and what will be the impact of their use in future embedded systems. What is the price to pay for adaptability, and for which kind of applications it is well suited. This part will also present a working implementation of the software infrastructure to select and trigger the best reconfiguration of the architecture at run-time.
- **Conclusions** A summary of the research discussed and the road ahead.

2. ADDITIONAL AUTHOR INFORMATION

The following persons were involved in building this tutorial:

- Stephan Wong (Delft University of Technology - TUD) is an associate professor at Computer Engineering laboratory at TUD. He is the co-inventor of the MOLEN processor architecture and has considerable experience with reconfigurable architectures. He is coordinator of the ERA project (an European funded FP7 project) that is focusing on many topics described in this tutorial.
- Luigi Carro (Universidade Federal do Rio Grande do Sul - UFRGS) is a full professor at the Institute for Informatics at UFRGS. He has considerable experience with computer engineering with emphasis on hardware and software design for embedded systems focussing on: embedded electronic systems, processor architecture dedicated test, fault-tolerance, and multiplatform software development.
- Stamatis Kavvadias (University of Siena - UNISI) is a research associate at the Department of Information Engineering at UNISI. His main research area is memory hierarchy microarchitecture, communication, synchronization, cache coherence, and task scheduling in CMPs.
- Georgios Keramidas (Industrial Systems Institute - ISI) is a post-doctoral fellow at ISI and a visiting assistant professor at the University of Patras, Greece. His research interests include computer architecture, in particular, the design of memory subsystems of single core and multicore systems.
- Francesco Papariello (ST Microelectronics - STMICRO) is a researcher in the Advanced Systems Technology R&D group at STMICRO. He has been involved in the design and development of simulation and design space exploration tools, among them the Lx (ST2xx family) simulation models and the xSTreamISS, the simulation infrastructure for the xSTream platform (streaming multi-core heterogeneous system).
- Claudio Scordino (Evidence s.r.l. - EVI) is a project manager within EVI and his research activities include operating systems, real-time scheduling, energy saving and embedded devices.
- Roberto Giorgi (University of Siena - UNISI) is an associate professor at the Department of Information Engineering at UNISI. His main research area is in computer architecture with emphasis on multiprocessor/multicore issues (processor, coherence, and programmability).
- Stefanos Kaxiras (Uppsala University - UU) is a full professor at the Department of Information Technology at UU. He has considerable experience in memory systems (highly scalable cache coherence, cache management using reuse distances), power (decay), instruction-based prediction, network processors (IPstash IP-lookup memories), memory/processor integration (datascalar/distributed vector architectures).

3. ACKNOWLEDGMENTS

The research leading to the results presented in this tutorial has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement nr. 249059. The information presented is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability. The opinions expressed in the document are of the authors only and I no way reflect the European Commission's opinions.