MNEMEE: Memory management technology for adaptive and efficient design of embedded systems

Stylianos Mamagkakis[1], Peter Lemmens[1], Dimitrios Soudris[2], Twan Basten[3], Peter Marwedel[4], Dimitrios Kritharidis[5] and Gwenael Guilmin[6]

[1] IMEC vzw, Kapeldreef 75, B-3001 Leuven, Belgium, mamagka@imec.be, lemmensp@imec.be
[2] Institute of Communications and Computer Systems - ICCS, 9, Iroon Polytechniou Str., 15773, Athens, Greece, dsoudris@microlab.ntua.gr
[3] Technische Universiteit Eindhoven - TUE, Department of Electrical Engineering P.O. Box 513, 5600 Eindhoven, The Netherlands, a.a.basten@tue.nl
[4] Informatik Centrum Dortmund e.V. ICD, Joseph-von-Fraunhofer-Str. 20, 44227 Dortmund, Germany, peter.marwedel@udo.edu
[5] Intracom S.A. Telecom Solutions, 19.7 Km Markopoulou Avenue, 190 02 Peania, Athens, Greece, dkri@intracom.gr
[6] Thales Communications - TCF, 146 Boulevard de Valmy, BP 82, 92704 Colombes Cedex, France, Gwenael.GUILMIN@fr.thalesgroup.com

***Abstract*— The theme of intelligent ubiquitous devices will dominate future embedded system designs and speed up the integration of multimedia and communication applications, thus creating very complex, dynamic source code. Today it is increasingly impossible for designers to map applications cost-efficiently to any platform without significant optimization of the initial source code. The MNEMEE project will address this key challenge with innovative source-to-source optimizations for data management. Novel features include: i) multi-objective explorations that allow trade-offs which designers highly need to rightly position their product in the huge search space, ii) a combination of design-time and run-time techniques to boost the cost-efficiency and performance, and iii) automated optimizations that are applied once to reduce design effort and can handle very complex code. These optimizations and automation support will result in reduced exploration design time by at least a factor of 2, decreased memory footprint and memory bandwidth requirements of 30%, and improved energy and power efficiency by a factor of 2. The project will enable customized solutions for programmable MPSoC platforms involving the necessary system trade-offs, while meeting all real-time specifications.**

MNEMEE PROJECT OVERVIEW

The key characteristic of future applications will be the intensive data transfer and storage and the need for efficient memory management [1, 2]. This memory management happens at several levels of abstraction: from virtual-dynamic memory management of the application, over physical memory management (being still at the platform-independent source code level), to detailed memory management (at memory hierarchy/ architecture/ platform dependent level). Hence, inefficient memory management support will lead to decreased system performance and increased cost in energy consumption and memory footprint due to larger memory needs. Unfortunately, it is well known today by designers that it is impossible to cost-efficiently map these applications on any platform without significant optimization of the initial application source code. This problem gets even more difficult in Multi-Processor System on Chip (MPSoC) platforms, which will dominate future designs for these nomadic embedded systems [3]. The embedded system designer community needs optimization methodologies and tools, which do not change the input-output functionality of the software applications, its real-time behavior or the design of the underlying hardware platform. These tools should be able to optimize both statically and dynamically allocated data, in order to cope with design-time needs and adapt to run-time memory needs (scalable multimedia input changing at run-time, unpredictable network traffic, etc.) in the most efficient way. Finally, there is a definite need to deliver trade-offs based on multi-objective cost functions to enable product customization at design time and adaptation at run-time.

The MNEMEE project is an effort to address these challenges and to deliver all the necessary design methodologies, heuristics and prototype tools to enable the fast exploration of the huge dynamic and static design space. More specifically the MNEMEE project will:

1) Introduce a novel source-to-source optimization design layer and framework for static and dynamic data management in MPSoC embedded systems between the state-of-the-art optimizations at the application and the compiler design layer.

2. Analyze embedded software applications and highlight the source-to-source optimization opportunities. This will enable the modeling of the data access and storage behavior and identify the memory management bottlenecks.

3. Develop a set of prototype tools that will assist the embedded system designer in the source-to-source optimization process and reduce considerably the total design effort needed and handle multi-objective optimization decisions.

4. Provide data memory-hierarchy aware assignment and scheduling methodologies, which will take advantage of memory hierarchy configurations of the underlying MPSoC platform, exploiting multi-layered memory configurations of on-chip (e.g. scratchpads, caches, etc.) and off-chip memories (e.g. SDRAMs).

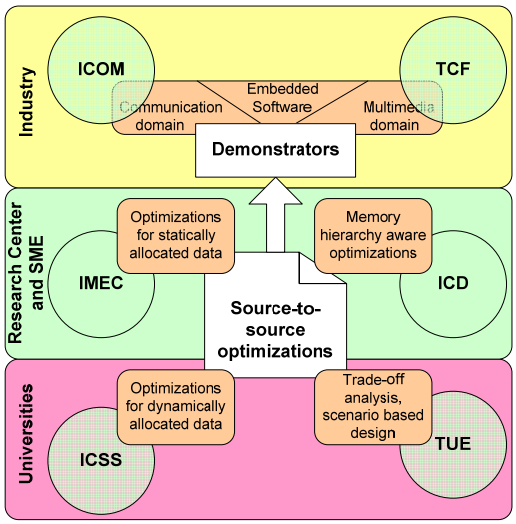

Figure 1. Complimentarity and expertise of MNEMEE partners

As can be seen in Figure 1, the individual source-to-source optimizations will leverage on the existing, extensive contributions to the state-of-the-art of each MNEMEE partner. The individual optimizations will be unified in a single design optimization flow, as can be seen in Figure 2. The industrial partners will introduce the optimizations in their design flows in their respective application domains and integrate them and develop a demonstrator for the multimedia and communication application domain respectively.

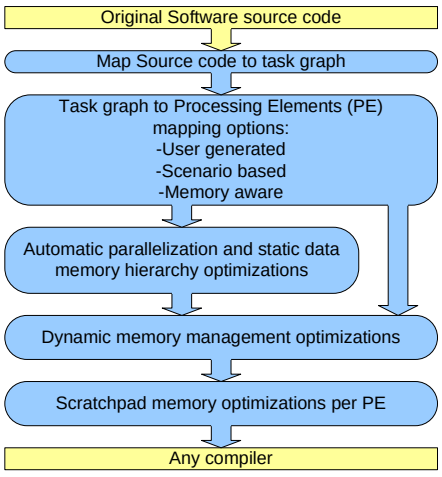

Figure 2. Proposed MNEMEE design flow

On the one hand, ICCS and TUE focus in more explorative, mixed design-time and run-time memory management methodologies and tools (based on [7, 8] and [9, 10], respectively). On the other hand IMEC and ICD focus in more mature, design-time memory mapping methodologies and tools (based on [4, 5, 6] and [11], respectively). Finally, ICOM and TCF will develop demonstrators that will be used to evaluate the aforementioned methodologies and tools (based on [12] and [13], respectively).

CONCLUSIONS

To conclude, the proposed optimizations and automation support target in reduced exploration design time by at least a factor of 2, decreased memory footprint and memory bandwidth requirements of 30%, and improved energy and power efficiency by a factor of 2. The project will enable customized solutions for programmable MPSoC platforms involving the necessary system trade-offs, while meeting all real-time specifications.

ACKNOWLEDGMENT

This work was supported by the EC through the FP7 IST project 216224 MNEMEE.

REFERENCES

[1] A Theme for research and development under the specific programme “Cooperation” implementing the Seventh Framework Programme (2007-2013) of the European Community for research, technological development and demonstration activities - 2007-08 Work Programme, www.cordis.lu/ist

[2] P. Marwedel, ‘Memory architecture aware compilation’, ACACES tutorial, July 2007, L'Aquila, Italy

[3] Hugo de Man, "Ambient Intelligence: Gigascale Dreams and Nanoscale Realities" Keynote speech in Solid-State Circuits Conference - ISSCC 2005 IEEE International, Feb. 6, 2005.

[4] ATOMIUM – Data Transfer and Storage Exploration optimization tool: http://www.imec.be/design/atomium/

[5] SPRINT source code parallelization tool: http://www.imec.be/design/sprint/

[6] MPSoC Design Time Application Mapping http://www.imec.be/design/mpsoc/mpsoc_dtm.shtml

[7] D. Atienza, J. M. Mendias, S. Mamagkakis, D. Soudris, F. Catthoor: Systematic dynamic memory management design methodology for reduced memory footprint. ACM Trans. Design Autom. Electr. Syst. 11(2): 465-489 (2006)

[8] S. Mamagkakis, A. Bartzas, G. Pouiklis, D. Atienza, F. Catthoor, D. Soudris, A. Thanailakis: Systematic methodology for exploration of performance - Energy trade-offs in network applications using Dynamic Data Type refinement. Journal of Systems Architecture (2007)

[9] M.C.W. Geilen, T. Basten. A Calculator for Pareto Points. In Design, Automation and Test in Europe, DATE 2007, pages 285-291. IEEE CS, 2007.

[10] Scenario-based design method: http://www.es.ele.tue.nl/scenarios

[11] ICD-C compiler framework: http://www.icd.de/es/icd-c/icd-c.html

[12] P802.16Rev2/D2 (December 2007) (Revision of IEEE Std 802.16-2004 and consolidates material from IEEE Std 802.16e-2005, IEEE Std 802.16-2004/Cor1-2005, IEEE Std 802.16f-2005 and IEEE Std802.16g-2007)

[13] “The 600 bits/s, 1200 bits/s and 2400 bits/s NATO interoperable narrow band voice coder”, NATO Standard STANAG No. 4591 Edition Y (amendment W) Ratification Draft 1, January 2006.

[14] MNEMEE website: http://www.mnemee.org