Information Society Technologies

HiPEAC info 13

COMPILATION ARCHITECTURE

appears quarterly | January 2008

Network of Excellence on High Performance Embedded Architectures and Compilers

Welcome to HiPEAC 2008 Conference

www.HiPEAC.net

HiPEAC² starts on February 1, 2008

Message from the HiPEAC coordinator

Mateo Valero
Coordinator
UPC Barcelona
mateo@ac.upc.edu

Dear colleagues,
To give 2008 a good start, we have scheduled the HiPEAC Winter Conference in January in Göteborg, Sweden, where the HiPEAC consortium will be warmly welcoming you. This event has raised increasing interest among our community, and we hope it will be seen as one of the major conferences in our field.

As in previous years, the first HiPEAC Journal issue for 2008 will be devoted to top papers in the HiPEAC 2008 Conference. September 2007's issue of the HiPEAC Journal, consisting of three papers, is currently available on our website.

Last November, HiPEAC successfully passed its last review with the European Commission. Our events were highly appreciated by the reviewers, especially our industry workshops, a forum that brings together representatives from industry and academia. We will continue to strive to improve our Network by involving more industry participants in our activities.

Over 120 participants, more than half of whom were industry representatives, attended the fourth HiPEAC Industrial Workshop on November 26, 2007 at the ARM premises in Cambridge, UK. 12 papers and six posters were submitted to the five sessions and were presented by industry or academia representatives in the following areas: Optimization for Embedded Systems; Language and Tool Support for Multi-core Architectures; Dependable Computing; Modelling and Simulation; and a session on relevant EU projects. These presentations are now available on our website.

The last call for collaborations closed in mid-September, distributing funds of over one million euros. The call granted funds for clusters and fellowships, internships, 3-month travel grants and publications in top conferences. Another call for collaborations, which will be the last of this first HiPEAC phase, may take place in February; our website will keep you updated with the latest information.

We are already preparing for ACACES 2008, our 4th International Summer School on Advanced Computer Architecture and Compilation. The programme, involving as usual the participation of several prestigious professors and speakers, is announced in this newsletter.

HiPEAC is steadily growing: I welcome the new members and their research teams that joined our network during 2007. HiPEAC remains open to inviting new collaborators to run our research clusters until the end of the project this year, when we will be handing over to HiPEAC^2. If you intend to join us, now is always the best time!

Message from the project officer

Panos Tsarchopoulos
Panagiotis.Tsarchopoulos@ec.europa.eu

As I noted in the previous newsletter, we have started preparing the next Workprogramme for the "Computing Systems" objective, which will be published most probably at the end of 2008. In the last months of 2007, we organised three consultation meetings to discuss the research challenges in computing. The first meeting, discussing the overall challenges for computing systems, took place in Braga, Portugal, on 19th November. The second meeting, specifically discussing research challenges in reconfigurable computing, took place in Brussels, on 13th December. Finally, the third consultation meeting, discussing high-performance computing, took place in Brussels, on 17th December. The help of HiPEAC in organising these three meetings was crucial and very much appreciated. We plan to have the reports and the presentations from these meetings on our Cordis website over the next couple of weeks. Your input and comments will be very important for us.

For every research programme, and especially for collaborative research programmes that bring together industry and academia, it is important to keep an eye on how the market context will be evolving. Let's see some data on the microprocessor market.

According to Instat, the total revenue for microprocessor chips (including both embedded and general-purpose CPUs) was $54.7 billion in 2005 and it will reach $72.6 billion in 2011. It is very important to note that from 2005 to

Koen De Bosschere
HiPEAC² Coordinator

HiPEAC² Kicks off on January 30, 2008

As you have already read in a previous newsletter, our community succeeded in having a second Network of Excellence approved by the European Commission. This Network is also called HiPEAC – which now stands for High-Performance **and** Embedded Architecture and Compilation. This means that the scope of the Network is no longer limited to embedded architectures, but spans the complete range of computing systems. On top of that the Network also focuses on system software and tools. The new Network starts on February 1, 2008 which means that the two HiPEAC Networks will run in parallel for several months. In this transition period, gradually more and more tasks will be taken over by the new Network. We will work hard to make the transition between the two Networks as smooth as possible.

The new Network has 12 partners and is coordinated by Ghent University. The partners are ARM, RWTH Aachen, Chalmers University of Technology, Barcelona Supercomputing Center, FORTH, Ghent University, IBM, INRIA, NXP, ST, TU Delft, and the University of Edinburgh. These partners run the Network.

However, a much larger community of researchers in Europe can benefit from the Network by becoming a HiPEAC member. The main criterion to qualify as a HiPEAC member is that you contribute to the goals of the Network by doing excellent research, collaborating with other HiPEAC members, and attending the HiPEAC events. All existing HiPEAC members will automatically become members of the new HiPEAC network on February 1.

The HiPEAC² project consists of 4 work packages.

1. A **mobility work package** that aims to stimulate mobility among the different members of the Network. There are collaboration grants, sabbaticals, company internships, and cluster meetings. The cluster meetings are a continuation of the existing series of cluster meetings – three times per year: in fall, winter and spring. Other events will be co-located with these cluster meetings: the HiPEAC conference, the industrial workshops, general assemblies, and project meetings for other projects.
2. A **research work package**, which consists of nine predefined clusters, each coordinated by a HiPEAC partner. These clusters are thematic groups where the HiPEAC community meets, discusses research, and makes future research plans. Clusters have clear progress indicators, and everybody who contributes to these progress indicators is invited to apply for membership of the cluster. The nine cluster topics are: multi-core architecture, programming models and operating systems, adaptive compilation, interconnects, reconfigurable computing, design methodology and tools, binary translation and virtualization, simulation platform, and compilation platform.
3. A **spreading excellence work package** which consists of 11 tasks, most of which you are already familiar with: the HiPEAC conference, the ACACES Summer School, the HiPEAC journal, the HiPEAC newsletter, the roadmap, website, web seminars, industrial workshops, technical reports, promoting start-ups, and an award program for publication on top-conferences.
4. The **management work package**, dealing with the administration and the logistics of the network. The HiPEAC logistics staff is based in Ghent and supports the different partners in implementing their tasks.

The kick-off meeting for this HiPEAC Network takes place in Göteborg, Sweden, the day after the HiPEAC conference, January 30, 2008. It is followed by a first cluster meeting.

If you want to know more about the HiPEAC Network, please visit http://www.hipeac.net, or contact me at Koen.DeBosschere@UGent.be.

2011, Instat estimates that, although it will still be by far the biggest part, there will be a decrease of almost 5 percentage points of the total revenue share of microprocessors sold in the "Computer" market segment (i.e. mainly general-purpose microprocessors). At the same time, there will be an increase in the revenue share of microprocessors sold in the "Communications" (+3.3 percentage points) and in the "Automotive" (+1.6 percentage points) market segments. These are mostly embedded processors and in both of these market segments Europe has a very strong presence.

The actual number of microprocessors employed worldwide is exploding exponentially. Just for body electronics in the automotive market alone, VDC estimates that around 1.2 billion microprocessors will be sold in 2008. And since these are relatively less powerful microprocessors, handling simple tasks in the emerging category of In-Car Computing Systems, VDC estimates that around 90 million powerful, mainly 32-bit, embedded microprocessors will be sold in 2008; up from less than 45 million in 2005. (In-Car Computing Systems integrate wireless, voice and data technologies to provide services to the occupants onboard, including location-specific information, GPS navigation, communications, e-mail, voice recognition, and multimedia such as digital radio, CD, and DVD video players.)

This does not mean that PC sales (and therefore sales of PC microprocessors) will be declining. On the contrary, IDC estimates that around 300 million PCs will be sold worldwide in 2008, up from around 235 million in 2006. In 2011, IDC estimates that around 390 million PCs will be sold. A characteristic of the PC market is the very slow growth in the unit sales of desktops and servers compared to the explosion of laptop sales; in 2010, for the first time, there will be more laptops sold than desktops and servers.

Panos Tsarchopoulos
Project Officer

ACACES 2008: Fourth International summer school on advanced computer architecture and compilation for embedded systems

July 13 - 18, 2008, L'Aquila, Italy
We are proud to announce the fourth HiPEAC Summer School, which will take place during the third week of July in L'Aquila, a small town about 100 km northeast of Rome at the TILS Campus. We start on Sunday evening with an opening ceremony with a keynote talk by Yale Patt, University of Texas at Austin. On Monday the 12 courses start, spread over two morning slots and two afternoon slots. Per slot there are three parallel courses of which you can take one. The courses have been allocated to slots in such a way that it will be possible to create a summer school program that matches your research interests. The topics of this year's summer school will be presented by the world-class experts listed in the table below.

On Monday evening there will be an invited talk. On Wednesday afternoon, the participants will be given the opportunity to present their own work to the other participants during a huge poster session. On Friday evening there is a farewell dinner and party.

Students and lecturers will be accommodated in hotel-standard private rooms on campus, where they will stay for one week. This will provide plenty of opportunity to have discussions with the teachers and with the other participants in the relaxing surroundings of the TILS Campus. Long after-the-lecture discussions at the bar or the pool table are one of the major assets of this summer school. At the end of the event the participants will receive a certificate of attendance detailing the courses they took. You can arrange to be picked up in Rome on Sunday July 13, 2008 and to be taken back on Saturday July 19, 2008 to either Rome airport or downtown Rome (for those who want to spend some extra time in the eternal city).

Unfortunately, the number of participants will be limited. Therefore, we have an admission procedure to guarantee a fair distribution of the available places among all qualified applicants from the various countries and institutions. If you are a member of a HiPEAC institution you can ask for a grant that covers the registration fee. In this newsletter, you will find a summer school poster. Please post it at some visible place in your department. You can find more information about the summer school at http://www.hipeac.net/summerschool. We look forward to seeing you there!

Koen De Bosschere
Summer school organizer

Teacher	Affiliation	Title
Luca Benini	University of Bologna	Communication-dominated architectures: toward Networks on Chip
Chaitali Chakrabarti	Arizona State University	Low Power System Design
Nikil Dutt	University of California, Irvine	Bus-based On-Chip Communication Architectures
Babak Falsafi	EPFL	Bridging the Processor/Memory Performance Gap
Paolo Ienne	EPFL	Automatic Customization of Embedded Processors
Christos Kozyrakis	Stanford University	Transactional Memory: Concepts, Implementations, and Opportunities
Wayne Luk	Imperial College	Reconfigurable Technology and Custom Computing
Olav Lysne	Simula research laboratory	Routing and Resource Utilization in Interconnection Networks
Mary Lou Soffa	University of Virginia	Model Driven and Dynamic Optimization
Josep Torrellas	University of Illinois at UC	Multiprocessor Architectures for Speculative Multithreading
Dean Tullsen	University of California, San Diego	Multi-core and multi-threaded processor architectures
Leendert Van Doorn	AMD, Austin	Virtualization Technologies

HiPEAC 2009 Conference Announcement

We are very pleased to announce that the 4th HiPEAC conference will take place in Paphos on the eastern Mediterranean island of Cyprus during January 25-28, 2009. Paphos is a UNESCO World Heritage city renowned for its exquisite mosaics, the underground Tombs of the Kings, and many other archaeological sites. Paphos' picturesque harbour with its medieval fort and seaside cafes is another major attraction for visitors to Cyprus.

The general co-chairs of the conference are Andre Seznec (Irisa, Inria) and Joel Emer (Intel). The program co-chairs are Michael O'Boyle (University of Edinburgh) and Margaret Martonosi (Princeton University). Deadline for paper, workshop and tutorial submissions as well as other conference-related information will be announced by mid-February 2008 on the conference website:

http://www.hipeac.net/conference

Topics of interest include:

- Processor architectures
- Architectures for mobile, networked and embedded systems
- Memory system optimization
- Power, performance and implementation efficient designs
- Interconnection networks, networks-on-chip, network interfaces and processors
- Security, dependability, and predictability support
- Application-specific processors and accelerators
- Reconfigurable architectures
- Simulation and methodology
- Compiler techniques for embedded processors
- Feedback-directed optimization
- Program characterization and analysis techniques
- Dynamic compilation, adaptive execution, and continuous profiling/optimization
- Back-end code generation
- Binary translation/optimization
- Code size/memory footprint optimizations

Transactions on High-performance Embedded Architectures and Compilers

The following papers were accepted for the third issue:

Christine Rochange and Pascal Sainrat,
A Context-Parameterized Model for Static Analysis of Execution Times.
Transactions on High-Performance Embedded Architectures and Compilers, 2(3):109-128, 2007.

Amit Golander and Shlomo Weiss,
Reexecution and Selective Reuse in Checkpoint Processors.
Transactions on High-Performance Embedded Architectures and Compilers, 2(3):129-152, 2007.

RWTH Aachen launches MAPS to tackle MPSoC programming challenge

In the past few years, it has become clear that MPSoC (Multiprocessor System-on-Chip) is the most promising way to keep on exploiting the high level of integration provided by the semiconductor technology and, at the same time, matching the constraints imposed by the embedded-system market in terms of performance and power consumption. A modern MPSoC usually combines dozens or even hundreds of specialized programmable processors, which provides both powerful processing power and flexibility. However, the MPSoC solution poses a great number of challenges that have to be addressed by the designers. The problem of programming such parallel architectures in an efficient way is considered the biggest one.

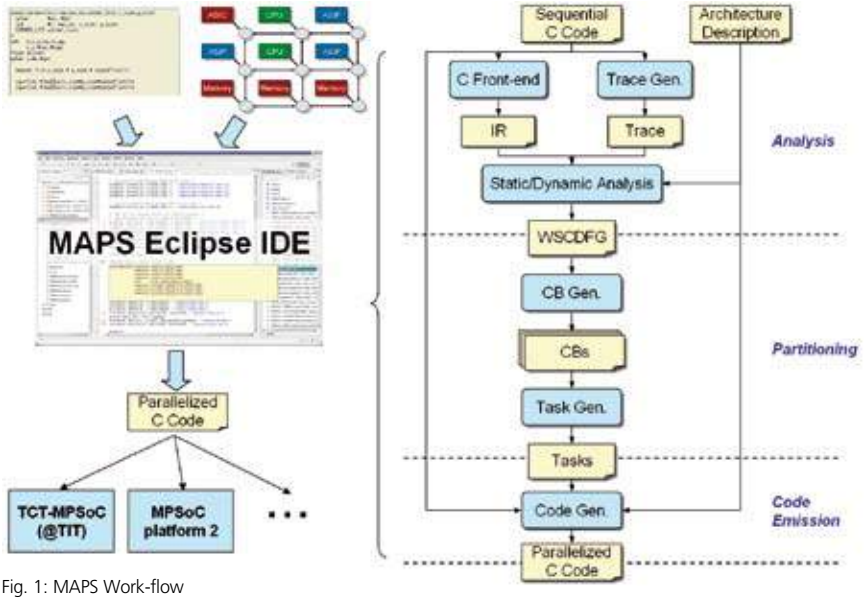


Fig. 1: MAPS Work-flow

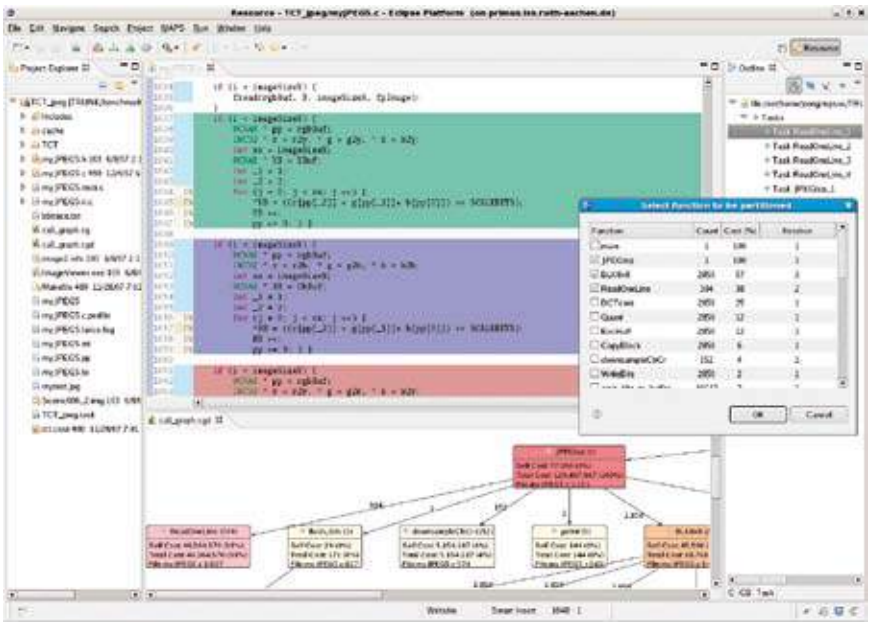


Fig. 2 MAPS Eclipse IDE

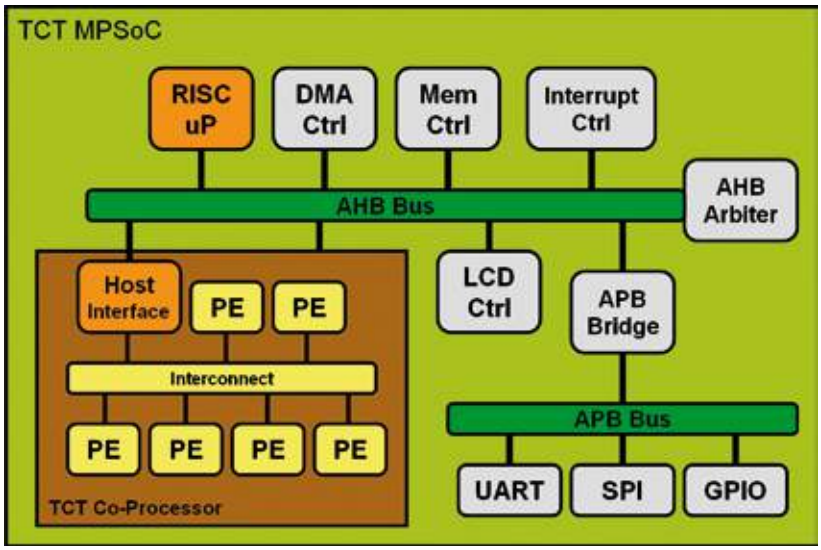
The ISS institute at RWTH Aachen University started looking into the challenge of MPSoC programming early this year and has been working on the tool-set, named MAPS (MPSoC Application Programming Studio), which aims to help algorithm/application developers to parallelize their sequential C programs for efficient execution in MPSoC platforms. Unlike a fully autonomous parallel compiler, MAPS provides a set of tools which guides the parallelization. The parallelization process is divided into three phases: analysis, partitioning and code emission (Fig. 1). The inputs to the MAPS workflow are two-fold: the application source code in C and the target MPSoC architecture description. As the first step, the MAPS analysis tool-set provides high-level profiling information to help the designer to understand the application execution. Both static and dynamic approaches are applied here to produce analysis results such as call-graph, source-level profiling, and control/data-dependency. With this information, the hotspots of the application and the functions that have potential to be partitioned are presented to the user for later partitioning. The second phase is to partition the C source code into coarse-grained tasks, which will be allocated to run parallel in the MPSoC platform. We have proposed a novel concept of C source code granularity level, called CB (Coupled Block), for this purpose. An iterative algorithm has been developed to extract CBs from the so-called WSCDFG (Weighted Statement Control/Data-Flow Graph), which is the compiler IR graph with detailed program-analysis information collected from the first phase. The extraction results, i.e. tasks, are delivered to the programmer in response to the suggestion of decomposing the application. With the a-priori knowledge of the MPSoC platform and the application, the programmer is also free to modify the suggested tasks or create some new tasks that the automatic approach does not find. The final task-model is then fed into the next phase.

Fig. 3 TCT-MPSoC Architecture

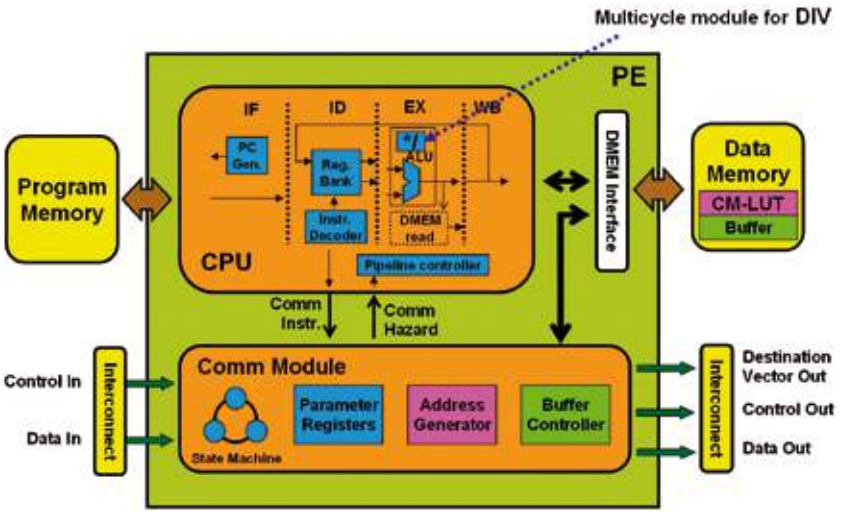
The code-emission phase generates the parallelized C code based on the task-model and the user is able to evaluate the task-partitioning by simulation. The MAPS tool-set is integrated into the open-source Eclipse IDE (Fig. 2) and works in a workbench style.

We have co-operated with Prof. Isshiki's group at Tokyo Institute of Technology and used their TCT-MPSoC platform (Fig. 3) as the first target hardware backend. Its target PE (Processing Element) is a RISC processor inside an MPSoC where 6 PEs are connected with a full crossbar and also connects with its host-CPU through an AHB bus. Each PE has a dedicated module for high speed communication (4-6 cycles setup, 4Bytes/cycle burst transfer, 1 cycle PE-to-PE latency). TCT-MPSoC also has the SW toolkit which includes compiler, trace scheduler and simulator. MAPS uses this SW toolkit in the work-flow to examine the parallel execution and the speedup. Other important characteristics, such as thread schedule, are also available. Based on the simulation result, the programmer can decide whether further improvement is necessary to explore the potential of the parallel platform.

JPEG encoder is one of the first few applications that we have successfully parallelized using the MAPS framework. After the first milestone, ISS research will move forward to improve our tools, e.g. smarter algorithms to discover more forms of parallelism, MPSoC architectural modelling, targeting more back-ends and fast virtual MPSoC platform simulation. For more details, please contact Prof. Rainer Leupers, leupers@iss.rwth-aachen.de.



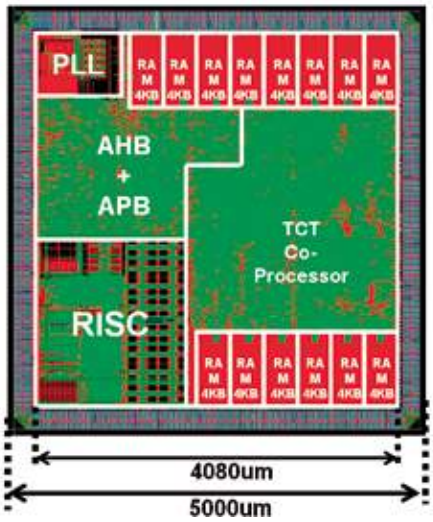
(a) TCT-MPSoC



(b) TCT Processor Core

About ISS

The Institute for Integrated Signal Processing Systems at RWTH Aachen University of Technology focuses on the design of wireless communication systems. A number of successful design automation tools for application specific systems have been developed at ISS, with LISATek (now available from CoWare Inc.) being the most recent example. Current research activities concentrate on multi-processor SoC design tools as well as compilers for embedded processors. Further information can be found at www.iss.rwth-aachen.de.



(c) TCT-MPSoC Implementation (TSMC 0.18um)

FP7-STREP Project: MERASA - Multi-Core Execution of Hard Real-Time Applications Supporting Analysability

The EC FP-7 MERASA project (http://www.merasa.org/), which started Nov. 1, 2007, targets analysable hardware architectures, system software, and WCET (worst case execution time) analysis tools for multi-core execution of hard real-time applications. Most of the partners met at the HiPEAC Cluster on Multithreaded Real-Time, namely the partners of the University of Augsburg, Barcelona Supercomputing Center, University Paul Sabatier of Toulouse, and Rapita Systems Ltd., expanded by Honeywell spol. s.r.o. of the Czech Republic as an application company.

A higher performance than what today's embedded processors can deliver will increase safety, comfort, and services, as well as lower the emissions of current and future automotive, aerospace, space, and construction-machinery systems. However, at the same time safety-related real-time embedded systems are developed, there is a need to prove that the timing requirements are met.

Higher performance is typically achieved by increasing the clock frequency of the processor, which also increases power consumption, leads to higher electromagnetic interference (EMI), requires a very stable power supply, and worsens the bottleneck in accessing platform resources, such as flash memory.

A drastic clock frequency increase is therefore unsuitable for embedded processors.

Multi-core processors offer a solution. Instead of scaling performance by improving single-core performance, performance is now scaled by putting multiple cores on a single chip, effectively integrating a complete multiprocessor on one chip. Multi-cores offer reduced average execution times for parallel workloads and a better performance/Watt ratio than a single core solution with similar performance. However, mainstream multi-core processor designs result in processors with unpredictable and unanalysable (or extremely pessimistic) worst case behaviour that deems them unusable in the domain of safety-related real-time embedded systems.

The overall objective of the MERASA project is to achieve a breakthrough in techniques for embedded multi-core processors, hard real-time support in system software for multi-cores, and WCET analysis tools, which will enable the requirements for high-performance to be combined with the time-predictable execution of single or multiple threads on a multi-core processor. As a result, we will achieve the societal benefits of increased safety and improved control of emissions into the environment in the areas of automotive, avionic and construction machinery.

The MERASA project investigates analysable high-performance features and core-interconnects of embedded multi-core architectures with 2 to 16 cores hand-in-hand with system-level software with predictable timing performance and timing analysis tools for hard real-time applications. We investigate architectural solutions and tools for WCET (worst case execution time) analysis. We complement the strongly analysable hard real-time demand with a weakly-hard real-time demand that guarantees that nearly all RETs (real execution times) are within a WCET boundary. The project will address both static WCET analysis tools (the OTAWA toolset of Université Paul Sabatier, Toulouse, France) as well as hybrid measurement-based tools (RapiTime of Rapita Systems Ltd., UK) and their interoperability. To constrain production costs and technology integration risks, we investigate hardware-based real-time scheduling solutions that empower the same multi-core processor to handle hard, soft, and non real-time tasks on different cores. The developed hardware/software techniques will be evaluated by application studies of safety-relevant real-time embedded systems performed by selected industrial partners.

The project is highly research focused, but significant efforts have been put into setting up links to industry end customers (Honeywell spol. s.r.o., Czech Republic) and the "glue" of using an SME (Rapita Systems Ltd.) to support the technology transfer beyond the scope of the project. A key feature of this project has been the introduction of an Industrial Advisory Board made up of key players in the EU, notably European Space Agency ESA, Airbus, Bauer Maschinen GmbH, Infineon

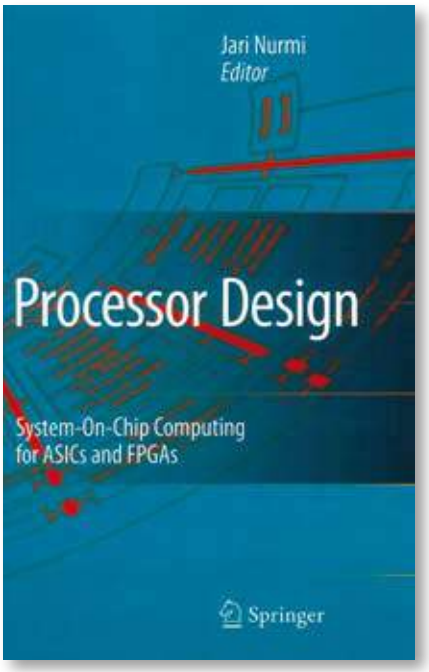
Drilling rig of Bauer Maschinen

Technologies, and NXP. Their contribution is to build a strong industrial focus in the project, not by bringing them in as full partners, but as advisors and observers of the project. The establishment of an Industrial Advisory Board is essential in providing a path to exploitation of these activities by providing industry requirements early in the project and reference case studies at the end of the project. We believe this is an excellent balance and a very cost-effective mechanism to ensure a clear research challenge of the project, but at the same time providing a clear exploitation path of the results of the project.

To reach its objectives, the MERASA project has been conceived to be 36 months in duration, comprising five work packages, structured in three main phases that correspond roughly to a spiral development model. The five work packages are WP1: Management; WP2: Design Space Exploration (WP leader: Francisco J. Cazorla from Barcelona Supercomputing Center); WP3: WCET Analysis and Tools (WP leader: Pascal Sainrat from Université Paul Sabatier); WP4: System-level Software (WP leader: Theo Ungerer from University of Augsburg); WP5: Pilot Studies, Dissemination and Exploration (WP leader: Guillem Bernat from Rapita Systems Ltd). The three phases of the project concern (1) Design Space Exploration, (2) Architectural Refinement, and (3) Prototyping and Pilot Studies.

Project Co-ordinator:
Theo Ungerer;
Department of Computer Science,
University of Augsburg,
86159 Augsburg, Germany
ungerer@informatik.uni-augsburg.de

New Book: Jari Nurmi (Ed.), Processor Design: System-on-Chip Computing for ASICs and FPGAs. Springer, 2007.



Processor Design addresses the design of different types of embedded, firmware-programmable computation engines. Because the design and customization of embedded processors has become a mainstream task in the development of complex SoCs (Systems-on-Chip), ASIC and SoC designers must master the integration and development of processor hardware as an integral part of their job. Even contemporary FPGA devices can now accommodate several programmable processors. There are many different kinds of embedded processor cores available, suiting different kinds of tasks and applications.

Processor Design provides insight into a number of different flavors of processor architectures and their design, software tool generation, implementation, and verification. After a brief introduction to processor architectures and how processor designers have sometimes failed to deliver what was expected, the authors introduce a generic flow for embedded on-chip processor design and start to explore the vast design space of on-chip processing. The types of processor cores covered include general purpose RISC cores, traditional DSP, a VLIW approach to signal processing, processor cores that can be customized for specific applications, reconfigurable processors, protocol processors, Java engines, and stream processors. Co-processor and multi-core design approaches that deliver application-specific performance over and above that which is available from single-core designs are also described.

The special design requirements for processors targeted for FPGA implementation, clock generation and distribution in microprocessor circuits, and clockless realization of processors are addressed. Tools and methodologies for application-specific embedded processor design are covered, together with processor modeling and early estimation techniques, and programming tool support for custom processors. The book concludes with a glance at the future of embedded on-chip processors.

Scheduling in High Performance Buffered Crossbar Switches

By Lotfi Mhamdi
(lotfi@ce.et.tudelft.nl)
Prof. Dr. Stamatis Vassiliadis
TU Delft, The Netherlands
October 19, 2007

Most high performance routers built today use crossbars and a centralized scheduler. Due to their scheduling complexity, crossbar-based routers are not scalable. This dissertation studies a buffered crossbar switching architecture. A scalable buffered crossbar switch design is proposed, using simple embedded unicast scheduling. Furthermore, an efficient buffered crossbar-based switching architecture is described for multicast traffic support as well as the integration of unicast and multicast traffic flows. Finally, a partially buffered crossbar switching architecture is shown to exhibit high performance, comparable to that of fully buffered crossbars and a low cost, comparable to that of unbuffered crossbars.

Arithmetic Soft-Core Accelerators

by Humberto Calderon
(hcalderon@ce.et.tudelft.nl)
Prof.dr. K.G.W. Goossens
TU Delft, The Netherlands
November 27, 2007

In this dissertation, we designed a universal arithmetic unit able to carry out multiple complex arithmetic operations while maximally reutilizing the shared hardware resources. First, we proposed an arithmetic unit for collapsed Sum-of-Absolute Differences (SAD) and Multiplication operations (AUSM) with a hardware reutilization level of up to 75% and a performance comparable to the fastest standalone designs supporting the individual operations. We also collapsed fixed-point dense and sparse matrix-vector multiplication in one unit and achieved 21GOPS on a Virtex2P FPGA device. Finally, we proposed an arithmetic unit for universal addition, which supports addition/subtraction in binary and Binary Coded Decimal (BCD) representations in various sign notations with a hardware reutilization level of 40% and a performance of more than 82MOPS.

Predictable Mapping of Streaming Applications on Multiprocessor

By Sander Stuijk (s.stuijk@tue.nl)
Prof. H. Corporaal,
Prof. J. van Meerbergen, dr. T. Basten
University of Technology Eindhoven,
The Netherlands
October 25, 2007

To manage the design complexity of modern consumer electronics devices, a predictable design flow is needed. The result should be a system that guarantees that an application can perform its own tasks within strict timing deadlines, independent of other applications running on the system. This requires that the timing behaviour of the hardware, the software, as well as their interaction can be predicted. This thesis presents techniques needed for developing systems with a predictable timing behaviour. These techniques are embedded into a coherent design flow. This makes it the first complete design flow that maps an application modelled with a time-constrained Synchronous Dataflow Graph to a Network-on-Chip-based Multi-Processor-System-on-Chip while providing throughput guarantees.

System-level design and configuration management for run-time reconfigurable devices.

By Yang Qu (yang.qu@vtt.fi),
Prof. Jari Nurmi
Tampere University of Technology,
Finland
November 2007

In this thesis, system-level supports for design of Dynamically Reconfigurable Hardware (DRHW) and various configuration management approaches for reducing the impact of configuration overhead are presented. In order to efficiently apply the multi-tasking feature of DRHW, we have developed three static task scheduling techniques and a run-time scheduling technique: a list-based heuristic approach, an optimal approach based on constraint programming and a guided random-search approach using a genetic algorithm. The run-time scheduling uses a novel configuration-locking technique. In addition, we present two novel techniques to reduce the configuration overhead. The first is configuration parallelism (loading tasks in parallel), the second is dynamic voltage scaling in the reconfiguration process to reduce the configuration energy.

Secure Computing on Reconfigurable Systems

By Ricardo Chaves
(rjfc@sips.inesc-id.pt)
Prof. Stamatis Vassiliadis (TU Delft),
Prof. Leonel Sousa (TU Lisbon),
Prof. Kees Goossens (TU Delft),
TU Delft, The Netherlands,
December 11, 2007

This thesis proposes a Secure Computing (SC) module for reconfigurable computing systems to avoid malicious attacks or tampering against the system. The SC provides a protected and reliable computational environment, based on encryption algorithms and on the attestation of the executed functions. Hardware cryptographic units were designed for both symmetric encryption and hash functions. Implementation results on a VIRTEX II Pro FPGA, in particular for the AES algorithm, show improvements of more than 500%. A novel method is also proposed to attest dynamically reconfigured hardware structures, which allows the reconfiguration bitstreams to be stored in unsecured mediums. By implementing the proposed SC module on a VIRTEX II Pro FPGA, throughputs above 1 Gbit/s can be achieved, with a low circuit usage.

Reconfigurable Network Processing Platforms

by Christoforos Kachris
(kachris@ce.et.tudelft.nl)
Prof.dr. K.G.W. Goossens
dr. J.S.S.M. Wong
TU Delft, The Netherlands
December 11, 2007

This dissertation describes a reconfigurable network processing platform (RNPP) that can be adapted to diverse network processing requirements and traffic fluctuations. An access/edge router, that can be adapted to network traffic fluctuations, is mapped and evaluated using the RNPP. Furthermore, a content switch is implemented and compared with a network processor-based implementation. The RNPP also incorporates a reconfigurable queue scheduler that can be adapted to the number of active queues. Finally, a configurable transactional memory controller is proposed that can facilitate the multi-processor programming of the network applications. The performance evaluation shows that RNPPs are most suitable for edge, access and enterprise network devices due to their performance, power, and flexibility.

Designs & Algorithms for Packet and Content Inspection

by Ioannis Sourdis
(sourdis@ce.et.tudelft.nl)
Prof.dr. S. Vassiliadis
Prof.dr. K.G.W. Goossens
TU Delft, The Netherlands
December 18, 2007

This dissertation deals with essential issues pertaining to high performance processing for network security and deep packet inspection. The proposed solutions keep pace with the increasing number and complexity of known attacks providing multi-Gbps processing rates. We advocate the use of reconfigurable hardware to provide flexibility, hardware speed, and parallelism in packet and content inspection functions. We consider high speed scanning and analyzing packet payload to detect hazardous contents, which are described in either static patterns or regular expression format. In addition, packet prefiltering is introduced to offload the overall processing of a packet inspection engine. Partially matching descriptions of malicious traffic avoids further processing of the majority of the attack descriptions per packet.

Diversity for software protection

By Bertrand Anckaert
(bertrand.anckaert@elis.ugent.be)
Prof. Koen De Bosschere
Ghent University, Belgium
January 2008

The openness of the PC has left software vulnerable to attacks by inspection or modification. Diversity is a technique that, rather than protecting against the first attack, reduces the ease with which an attack can be reused: information learned from one copy may no longer be applicable to other copies. Furthermore, diversity can be used to hide critical differences between versions. Consider the example of a security update – here, the difference between the patched and the original version may reveal a vulnerability which can be exploited on systems that are not kept up-to-date. Through diversity, we can hide the original differences within a large number of artificial differences. This thesis describes a number of techniques to generate syntactically different versions of the same program. Furthermore, we have set up a practical attack in which we automatically try to detect similarities and differences between versions.

Profiling Techniques for Performance Analysis and Optimization of Java Applications

By Dries Buytaert
(dries.buytaert@elis.ugent.be)
Prof Koen De Bosschere,
Prof Lieven Eeckhout
Ghent University, Belgium
January 2008

As Java applications become more complex and as virtualization becomes more important, we need to investigate profiling techniques that combine information captured at different layers of the execution stack. In this PhD dissertation, we present three novel profiling techniques. The central attribute of these contributions is that they link information gathered at different levels of the execution stack, and that this information is used to produce more complete profiles. By collecting information at the micro-processor level, we have profile information that is more accurate, faster to obtain, or that was otherwise not available. We show that these advances in profiling techniques lead to better understanding of program behaviour and faster executing Java applications.

Upcoming events

11th Workshop on Computer Architecture Evaluation using Commercial Workloads (CAECW-11),
Salt Lake City, Utah, February 16, 2008, http://www.cs.utah.edu/hpca08/caecw/

HPCA-14, 14th International Symposium on High-Performance Computer Architecture,
Salt Lake City, Utah, February 16 - 20, 2008, http://www.ece.arizona.edu/hpca08/

PPoPP'08 13th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming,
Salt Lake City, Utah, February 20 - 23, 2008, http://research.ihost.com/ppopp08/

DATE, Design, Automation and Test in Europe,
ICM, Munich, Germany, March 10 - 14, 2008, http://www.date-conference.com/

SAC 2008, 23rd ACM Symposium on Applied Computing,
Fortaleza, Brazil, March 16 - 20, 2008, http://www.acm.org/conferences/sac/sac2008

CC 2008 - International Conference on Compiler Construction,
Budapest, Hungary, March 29 - April 6, 2008, http://www.sable.mcgill.ca/~hendren/CC2008/

CGO-2008, 2008 International Symposium on Code Generation and Optimization,
Boston, Massachusetts, April 6 - 9, 2008, http://www.cgo.org/

ODES-6: 6th Workshop on Optimizations for DSP and Embedded Systems (co-located with CGO-2007),
Boston, Massachusetts, April 6 - 9, 2008, http://www.imec.be/odes/

ISPASS-2008, 2008 IEEE International Symposium on Performance Analysis of Systems and Software,
Austin, Texas, April 20 - 22, 2008, http://ispass.org/ispass2008/

ACM International Conference on Computing Frontiers,
Ischia, Italy, May 5 - 7, 2008, http://www.computingfrontiers.org/

SIES'2008, Third International Symposium on Industrial Embedded Systems,
Hotel Mercure, Montpellier - La Grande Motte, June 11 - 13, 2008, http://www.lirmm.fr/SIES2008/

ACACES 2008, Fourth HiPEAC Summer School,
L'Aquila, Italy, July 13 - 19, 2008

SAMOS VII: International Symposium on Systems, Architectures, Modeling and Simulation,
Samos, Greece, July 21 - 24, 2008, http://samos.et.tudelft.nl/samos_viii/

International Symposium on System-on-Chip 2008,
Tampere, Finland, November 5 - 6, 2008, http://soc.cs.tut.fi/

HiPEAC 2009 Conference,
Paphos, Cyprus, January 25-28, 2009

Contributions

If you are a HiPEAC member and you want to contribute to this newsletter, please contact Sylvie Detournay at **sylvie.detournay@elis.ugent.be**

HiPEAC info 13

HiPEAC Info is a quarterly newsletter published by the HiPEAC network of excellence.
Funded by the 6th European Framework Programme (FP6), under contract no. IST-004408.
Website : http://www.HiPEAC.net
Subscriptions: http://www.HiPEAC.net/newsletter

Registration for ACACES 2008 now open at http:/www.HiPEAC.net/summerschool