IST-004527 ARTIST2
Network of Excellence
on Embedded Systems Design

Activity Progress Report for Year 4

JPRA-NoE Integration

Resource aware design

Clusters:

Execution Platforms
Compilers and Timing Analysis

Activity Leader:

Prof. Dr. Peter Marwedel, University of Dortmund
http://ls12-www.cs.uni-dortmund.de/~marwedel

Prof. Luca Benini, University of Bologna
http://www-micrel.deis.unibo.it/~benini/

Policy Objective (abstract)

Provide, through the integration of research activities of many participants, a viable path for resource-aware software and hardware development. The final objective is to achieve integration of research activities in concrete deliverables:

- *A set of tools that can interact and work together and demonstrate the achievable optimizations on a particular hardware platform.*
- *A methodology that enables the design of predictable embedded systems with a special focus on issues that cut several layers of abstraction, such as hardware and compiler design.*

Table of Contents

1. Overview of the Activity

1.1 ARTIST Participants and Roles

Prof. Dr. Luca Benini - University of Bologna (Italy)
Power modelling and OS integration

Prof. Dr. Petru Eles - University Linköping (Sweden)
Dynamic and leakage power optimisation for real-time system, accurate system-level power modelling for communication, predictable real-time systems, fault-tolerance

Prof. Dr. Rainer Leupers - RWTH Aachen (Germany)
Processor modelling tools

Prof. Dr. Jan Madsen, Technical University of Denmark (TUD)
Power modelling and resource aware design space exploration

Prof. Dr. Peter Marwedel - University of Dortmund (Germany)
Architecture-aware compilation, low-power code generation, Development of optimizations for worst case execution time (WCET) minimization.

Prof. Dr. Lothar Thiele – ETH Zürich (Switzerland)
Cooperating on concepts for real-time components (but not included in budget)

Prof. Dr Reinhard Wilhelm – Saarland University (Germany)

Cooperating on prectable and efficient embedded systems design

1.2 Affiliated Participants and Roles

Roberto Zafalon – STM (Italy)
Efficient multi-core programming for embedded platforms

1.3 Starting Date, and Expected Ending Date

Starting date: September 1st, 2004.

Expected ending date (within ARTIST2): End of the project

Resource aware design will continue to play a key role well beyond the end of ARTIST2 funding, as it is one of the key themes in embedded systems design.

1.4 Baseline

The importance of resource awareness in embedded systems growing. Resources include energy, computational power and hardware components. This activity is concerned with optimization of resource usage, while at the same time meeting application requirements. Energy efficiency, Reliability are typical requirement of embedded applications. Predictability is also an extremely important one.

With the growing software content in embedded systems and the diffusion of highly programmable and re-configurable platforms, software is given an unprecedented degree of control on resource utilization. This relation between hardware and software layers can be used to perform aggressive optimizations that can be achieved only by a synergistic approach that combines the advantages of static and dynamic techniques.

1.5 Problem Tackled in Year 4

Several activities have been carried out in Y4 following the main directions established in Y3. Integration has been pushed forward and applications with increased size and complexity have been tackled. More specifically, work has focused on the following topics.

- Real-time systems have to provide predictable upper bounds on the execution time. Existing approaches can predict execution times for single processors. Due to the increased use of multicore systems, there is now also the need to provide run-time guarantees for systems comprised of multiple processors. In the fourth year the problems of bus access optimisation and synthesis of bus controllers for predictable multiprocessor systems have been addressed.
- Due to the predicted increase in process variations, fault-tolerance is becoming an indispensable ingredient of complex embedded systems. Fault-tolerance of real-time systems is more difficult to achieve than fault-tolerance for other systems, since retries can violate time-constraints. The problems related to fault tolerance in mixed hard/soft real-time systems have been addressed in the fourth year.
- Memory systems are key consumers of the scarcest resource in embedded systems, electrical energy. Modeling memory hierarchies and exploring alternative memory organizations is an important step in execution platform analysis and optimization. This problem has been tackled in Y4 by integrating a multicore virtual platform environment developed by BOLOGNA with a memory hierarchy model developed by DORTMUND.
- The issue of generating code for parallel multicore architectures has become a critical one. The extraction of parallel execution threads from sequential code by compiler analysis and optimization is required to facilitate the exploitation of parallel processors without completely re-writing legacy applications. Partners AACHEN and BOLOGNA have worked in this area, by sharing a complex practical application and analyzing the potential for its compiler-based parallelization.
- ETHZ has been cooperating with BOLOGNA in order to set up an environment that enables the mapping of data flow graph onto a multi-processor system. To this end, ETHZ has send students to BOLOGNA and the MPARM simulation environment has been selected to be the target platform. Part of the envisioned environment is (a) the specification of the application, (b) the specification of the target platform, (c) the specification of the mapping. Based on this information, the run-time system is generated, the code is compiled and downloaded to the platform. The mapping is determined by a multi-objective optimization method. The underlying performance analysis uses the well known compositional method MPA-RTC (Modular Performance Analysis – Real-Time Calculus).
- In addition, ETHZ has been working on mapping algorithms to resource-constraint mult-processor platforms while taking into account (a) real-time constraints and (b) leakage and dynamic power.
- Saarland has worked on predictability of various memory architectures, in particulare focusing on the predictability of cache replacement strategies. Hard bound have been computed and it has been shown that the replacement strategy has a strong influence on the precision of any type of cache analysis.

2. Summary of Activity Progress

2.1 Previous Work in Year 1

Work achieved in the first 6 months:

Cooperation was established between the Universities of Bologna and Dortmund. The objective was to integrate the memory-aware compiler developed in Dortmund with the multi-processor platform simulator developed in Bologna. The first results were the definition of a standard format for the executable output of the compiler, as well as for the memory allocation information. This output format is supported by the platform simulator.

Cooperation was furthermore established between the Universities of Bologna and Aachen. The objective is to extend the modelling capabilities of the platform simulator developed in Bologna toward heterogeneous multi-core architectures, exploiting the Application-specific Processor development framework based on the LISA architecture description language developed in Aachen. The first result of this work was the definition of a standardized wrapping protocol which allows any SystemC core description generated by Aachen tools to be instantiated (multiple times) as a core in the platform simulator by Bologna.

Work achieved in months 6-12

The cooperations established in the first six months were continued and were significantly strengthened, as a significant amount of technical work was performed to sustain them. More specifically:

1. The cooperation between Bologna and Dortmund required the development of a new souce-level transformation tool for performing memory optimizations by Dortmund, and the development of compatible memory organization models by Bologna (including I and D caches as well as scratchpad memories).
2. The cooperation between Aachen and Bologna required extensive re-design of Bologna's core interfacing protocol within the platform simulator. On the other hand, Aachen has provided extensive technical support on Lisatek core wrapping architectures and toolsets.

An additional cooperation between Bologna and Saarland University was established. The objective of this cooperation is the exploitation of the platform simulator developed at Bologna, more specifically of the timing accurate core models incorporated in the simulator, as targets for the worst case execution analysis framework developed in Saarland University.

Milestones:

1. Established a working path between the memory-optimizing compiler developed at Dortmund and the platform simulator developed at Bologna. This tool interoperability path was validated during a visit of Dortmund's research staff to University of Bologna in July 2005
2. Development and extensive benchmarking of the LISATek-MPARM bridge: Several heterogeneous platforms were instantiated and tested. Performance analysis was carried out.

2.2 Previous Work in Year 2

Sorting of the following contributions is by the name of the city of the partner. Partners are mentioned in alphabetical order of the cities. The sequence of partner institutions has no significance with respect to the share of the partners in the workload.

Integration of LISATek ISS models in SystemC and the MPARM virtual platform (Aachen, Bologna)

The issues arising from the integration of LISATek ISS models in SystemC and the MPARM virtual platform have been investigated in more detail [Ang05], especially concerning the interaction with level one (L1) memories. A new MPARM functional model was developed to handle the L1 memory. It was also useful to cluster other functionality within the same block. The end result is called a “processor tile”, comprising LISATek-generated SystemC model of the processor and the most tightly coupled components (see fig. 1).

The following component models were developed:

- a timer device,
- an emulated serial port,
- a simple interrupt controller.

The first component is vital if attempting to port an operating system. The second is very useful for debugging purposes; placing it next to IP cores, instead of in a shared location accessible to all system processors, has the advantage of allowing for independent input/output, and prevents debug traffic from spilling onto the system interconnect where it could pollute performance statistics. Finally, the interrupt controller is both a requirement of the other two devices and a crucial component to develop efficient synchronization mechanisms in multiprocessor systems. The controller is externally attached to a set of system-level wires which convey inter-core interrupts. On the IP core side, a simple interrupt handshaking protocol was implemented at Bologna. In this protocol, the value of interrupt registers is copied on some LISATek core pins which are polled every cycle by the core to take proper action. The interrupt controller is memory mapped to let the core reset the pending interrupt flags and configure the masking status.

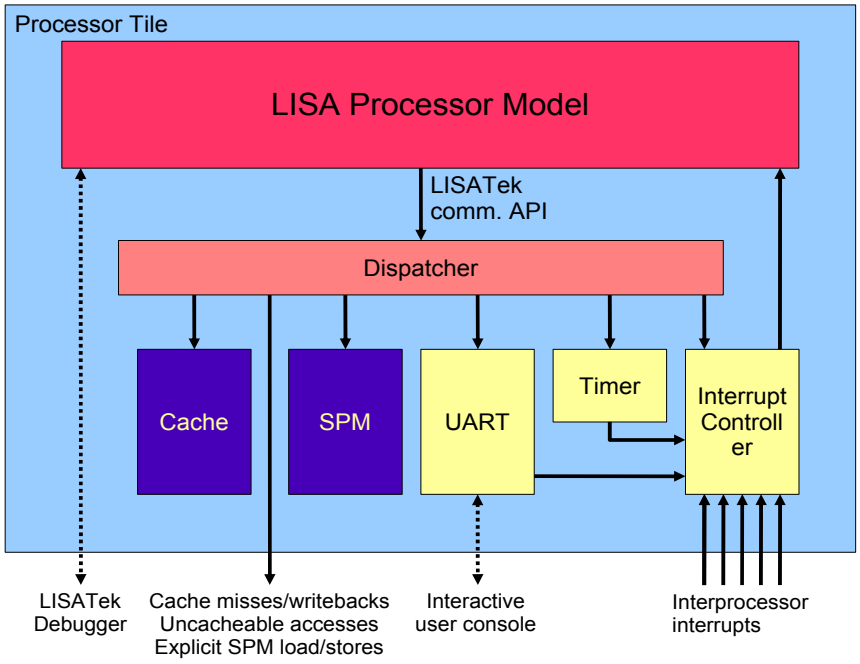

Figure 1 Processor Tile

Energy efficient time constrained systems (Bologna, Linköping)

Power models as well as a simulation environment for validation have resulted from cooperation of the University of Linköping with the Bologna group. As the first step, an approach for mono-processor systems has been elaborated, implemented and published [And05].

During the last six months of year 2, the efforts concentrated on an extension of this approach to multiprocessor systems. During the summer 2006, this work was performed as part of the ARTIST mobility action in cooperation with the Dortmund group and extended into the reporting period of year 3.

Predictability in Multiprocessor System on a Chip (MPSoC) architectures (Bologna, Braunschweig, Linköping)

Besides being energy efficient and having a high performance, for many applications it is required that multiprocessor SoC implementations are highly predictable with respect to their timing behaviour. This problem has been addressed by the Linköping group during this period. While this issue has been previously investigated in the context of mono-processor systems, available results are inapplicable to modern multiprocessor architectures in which, for example, due to the shared memory access and shared buses, the individual WCETs of tasks depend on the global system schedule. Providing WCET guarantees and reliable schedules in this context is extremely challenging. It involves issues related to bus protocols and control, WCET analysis, system level scheduling and optimizations. With regard to the "classical" aspect of WCET analysis the group is building on the Symta/P tool from the Braunschweig group (Ernst et al., a member of the (“execution platform” cluster). The Linköping group is also interacting with the Bologna group with regard to the issues of bus control.

This work is an effort started at the beginning of 2006. The overall concept has been elaborated, solutions have been developed and tools are under implementation. Publications and further results are expected in the following period.

Web site: http://www.ida.liu.se/~eslab/real-time.html

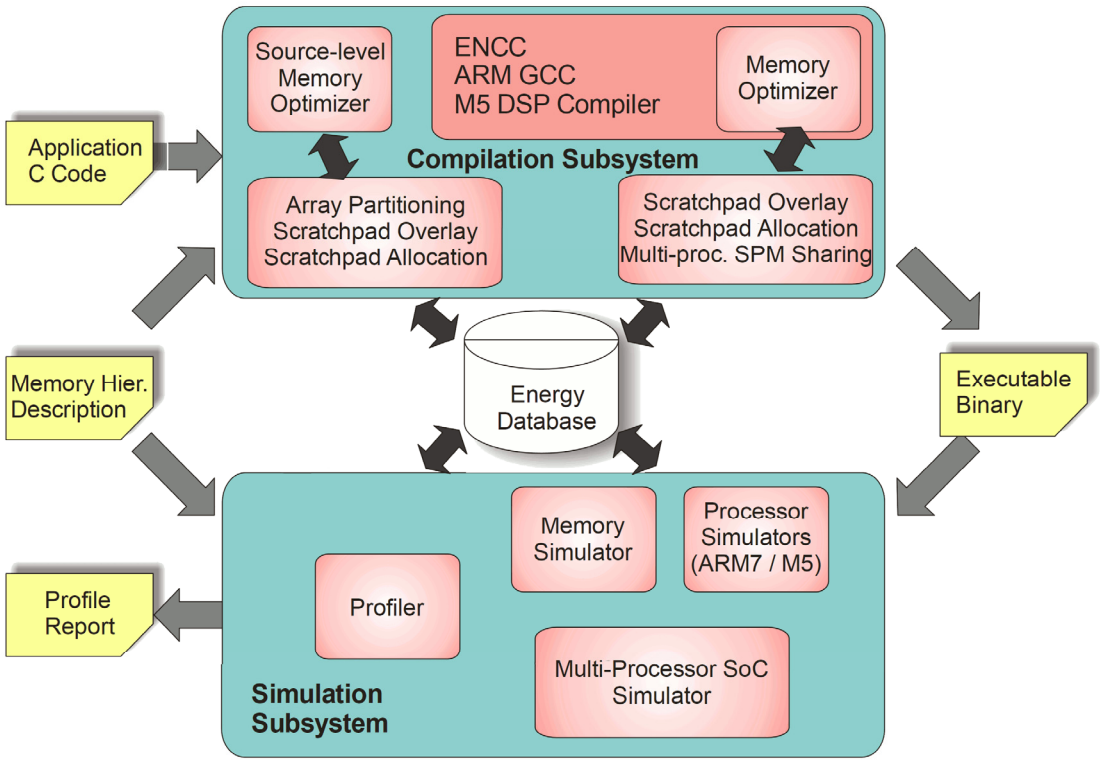

Figure 2: Memory Aware Compilation and Simulation Tool-Chain

Memory Aware Compilation and Simulation Tool-Chain for Energy Optimizations (Bologna, Dortmund)

During the last reporting period, the need for a coherent tool chain for energy optimizations and for exploration of memory hierarchies across different system architectures was recognized. Therefore, a memory aware tool-chain supporting uni-processor ARM, multiprocessor ARM and M5 DSP based systems was developed (see fig. 2) at Dortmund. Both the simulation and compilation subsystems are configured from a single memory hierarchy description. In addition, a common energy database is used by the memory optimizers in the compilation subsystem as well as by the memory and multi-processor SoC simulators in the simulation subsystem. The developed tool-chain optimizes input application code for a given memory hierarchy [Ver06d, Weh06] and also evaluates the optimization by simulating the optimized executable on the same memory hierarchy. The tool-chain is developed due to the cooperation between University of Dortmund and University of Bologna, as the simulation subsystem includes the multi-processor SoC simulation from Bologna while the compilation subsystem is developed at Dortmund. Moreover, both partners have agreed on a common memory hierarchy description format, which will be used for developing future optimizations [Ver06c].

Web site: http://ls12-www.cs.uni-dortmund.de/research/macc

Design-Time Memory Allocation Techniques for Multi-Process Applications with Aperiodic Processes (Bologna, Dortmund)

Previous work at Dortmund proposed compile-time or design-time memory allocation approaches to share the scratchpad memory among the periodic processes of a multi-process application. The current work extends the previous work and proposes memory allocation approaches for applications consisting of aperiodic tasks. This significantly increases the complexity of the memory allocator as the arrival times of the processes are completely

unknown at design time. Therefore, the memory allocator is divided into an intelligent design-time component and a simple run-time component.

The design-time component of the memory allocator works in the following stepwise manner. First, it identifies memory objects, *i.e.* code segments and data variables, which on scratchpad allocation lead to reduction in the energy consumption of the system. Second, it processes the application code to enable the movement of memory objects at runtime. Finally, it inserts blocking statements in the application code to prevent unsafe movement of memory objects. The runtime component, depending upon the current set of active processes and the current state of the scheduled process, allocates (de-allocates) memory objects to (from) the scratchpad memory. Experiments report that a two-phased memory allocator minimizes the energy consumption due to applications with aperiodic tasks [Ver06a, Ver06b].

Web site: http://ls12-www.cs.uni-dortmund.de/research/macc

Operating System Support for Online Allocation of Scratchpad Memories (Bologna, Dortmund)

The goal of this work at Dortmund is to develop a runtime memory allocator which keeps track of the execution behaviour of the application and allocates scratchpad memory with memory objects (code segments and data variables) at runtime. The runtime allocator of this approach is more complex than the design-time memory allocator described above. At compile time, attributes such as access counts and the size are computed for each memory object. These attributes are then supplied as input to the memory allocator. The allocator based upon the input attributes, the scratchpad memory utilization and the current execution pattern swaps memory objects in and out of the scratchpad memory. Several heuristics as well as analytical approaches have been proposed for the online allocation of the scratchpad memory. The proposed approaches have been integrated into the RTEMS operating system. Experiments demonstrate that for highly dynamic applications, significant energy savings can be achieved.

Web site: http://ls12-www.cs.uni-dortmund.de/research/macc

Resource awareness in sensor networks (Bologna, ETH Zürich)

The University of Bologna cooperated with ETH Zürich on resource awareness in sensor networks. For a full description please refer to the report on progress within the execution platform cluster for year 2.

Resource aware design space exploration (DTU, ETH Zürich)

The Technical University of Denmark (DTU) has developed a multi-objective design space exploration environment based on the PISA environment for multi-objective optimization from the group of Lothar Thiele, ETH Zurich. The exploration is based on a genetic algorithm to solve the problem of mapping a set of task graphs onto a heterogeneous multiprocessor platform. The objective is to meet all real-time deadlines subject to minimizing system cost and power consumption, while staying within bounds on local memory sizes and interface buffer sizes. The approach allows for mapping onto a fixed platform or onto a flexible platform where architectural changes are explored during the mapping. This work will be continued. A paper was published at DIPES 2006 [Mad06].

FET Open Call project proposal (Dortmund, ETH Zürich, Saarbrücken)

A consortium from within ARTIST2 consisting of the Universities of Saarbrücken, Zürich, Bologna, Pisa and Dortmund as well as AbsInt has applied for a project on “Reconciling Performance with Predictability” in the FET Open Call. Both short and long proposals have passed all thresholds.

Analysis of cache predictability (Saarbrücken, AbsInt)

First quantitative results have been obtained on the predictability of different cache architectures. A paper is in preparation.

Improvement of timing analysis by integration with code synthesis (Saarbrücken, AbsInt)

The University of Saarbrücken and AbsInt (an industrial member of the compiler cluster) have cooperated with ETAS (an external company located at Stuttgart, see http://www.etas.com) on the integration of the ASCET-SD model-based design tool with the AbsInt timing analyzer aiT. This work is continuing. A paper was published by Ferdinand et al. [Fer06].

Web site: http://en.etasgroup.com/about/tradeshows/documents/2006-03-15_AutomotiveSoftwareWorkshop_ASCET_Paper_Renz.pdf

Interfaces for real-time components (ETH Zürich, EPFL)

Between members of the group of Tom Henzinger (EPFL) and Lothar Thiele (ETHZ) there have been intensive discussions on interface based design of embedded systems. There were common meetings and presentations. The main concept is to extend the common idea of static types towards resource types that talk about the use of various resources by a component, e.g. power, time, computing resources. As a result, the concept of interface-based design (by Tom Henzinger) has been successfully applied to real-time systems and associated publications have been written [Hen06, Thi06, Cha06].

Web site: http://chess.eecs.berkeley.edu/pubs/92.html

2.3 Previous Work in Year 3

Predictability for Multiprocessor SoC Architectures (Bologna, Braunschweig, Linköping)

The very first steps for this work have been done during the first and second reporting period. The work has continued during the third year and first results are available and have been published.

In multiprocessor systems, the traffic on the bus does not solely originate from data transfers due to data dependencies between tasks, but is also affected by memory transfers as result of cache misses. This has a huge impact on worst-case execution time (WCET) analysis and, in general, on the predictability of real-time applications implemented on such systems. As opposed to the WCET analysis performed for a single processor system, where the cache miss penalty is considered constant, in a multiprocessor system each cache miss has a variable penalty, depending on the bus contention. This affects the tasks' WCET which, however, is needed in order to perform system scheduling. At the same time, the WCET depends on the system schedule due to the bus interference. We have developed an approach to worst-case execution time analysis and system scheduling for real-time applications implemented on multiprocessor SoC architectures. An important aspect of the problem is the bus scheduling policy and its optimization, which is of huge importance for the performance of such a predictable multiprocessor application. What concerns the "classical" aspect of WCET analysis we are building on the Symta/P tool from the Braunschweig group. The design of appropriate bus controllers to support the proposed approach is done in cooperation with the group in Bologna. A master student from Bologna is visiting Linköping for a period of seven months starting with June 2007.

Energy efficient time constrained systems (Bologna, Dortmund, Linköping)

This work has been started in the previous reporting period and has been performed at Linköping in cooperation with the groups at Dortmund and Bologna.

Olivera Jovanovic, a master student from Dortmund visited Linkoping for 7 months. The work has aimed at extending a dynamic, on-line, voltage scaling approach so that it can be applied to multiprocessor systems. Another extension concerns taking into consideration the voltage/frequency switching overheads at energy optimization. An approach for dynamic and leakage energy reduction via combined supply voltage scaling and body biasing in real-time multiprocessor systems has been developed. Discrete voltage modes and intra-task scaling have also been considered. The mapping and scheduling of the task sets were assumed to be already given. The main optimization target for this approach is to achieve energy efficiency by exploiting dynamic slack, which results at runtime, for example when the tasks do not execute their worst case number of clock cycles. Dynamic slack is exploited by using online voltage reduction techniques. Since these algorithms are executed online, after each of the tasks finishes, they must have a low complexity. Moreover the energy and time overhead for changing the supply and body bias voltage is also considered. Results for a journal submission have been generated.

For the experimental validation of the approach the MPARM simulation platform from Bologna has been used. A publication reporting the research results is in the final refinement steps.

Fault-tolerant embedded systems design (DTU, Linköping)

The Technical University of Denmark (DTU) and Linköping University started a collaboration on safety-critical embedded systems. Safety-critical applications have to function correctly and meet their timing constraints even in the presence of faults. Such faults can be permanent (i.e., damaged microcontrollers or communication links), transient (e.g. caused by electromagnetic interference), or intermittent (appear and disappear repeatedly). The transient faults are the most common, and their number is increasing due to the increasing level of integration in semiconductors.

Linköping has proposed a list scheduling-based heuristic for the generation of fault-tolerant schedules, and have used a tabu-search meta-heuristic on top of list scheduling to optimize the assignment of fault-tolerance policies (i.e., re-execution vs. active replication) in order to reduce the fault-tolerance overheads. Such heuristics are able to produce good quality solutions in a reasonable time.

Researchers have used constraint logic programming (CLP) in the context of system-level design. The advantages of a CLP approach are: it produces optimal solutions, can capture complex design constraints and trade-offs, it is flexible, more general and easy to extend. However, none of the proposed CLP approaches take into account fault-tolerance aspects. Hence, DTU has proposed a CLP framework that produces the fault-tolerant schedules such that the application is schedulable in the presence of transient faults, and the constraints and tradeoffs imposed by the designer are satisfied.

DTU have modelled the application as a fault-tolerant process graph, where the fault occurrence information is represented as conditional edges, and they have proposed an algorithm for the derivation of such graphs. The proposed CLP framework can be used to easily capture design optimization problems such as mapping and fault-tolerance policy assignment. In addition, the CLP framework can be used to reason about the effects of voltage scaling on reliability. Then, the system can be optimized for energy minimization under limited resources and strict timing and reliability constraints.

DTU have compared their CLP scheduling approach with the list-scheduling proposed by Linköping, and the CLP performs 25% better on average. By carefully optimizing the system implementation they are able to provide fault-tolerance under limited resources. The cooperation with Linköping has been in terms of reciprocal visits (Paul Pop, DTU has visited Linköping several times in 2007 and Viacheslav Izosimov, Linköping has visited DTU during 2006), exchange of tools and case studies.

Integration of LISATek ISS models in SystemC and the MPARM virtual platform (Aachen, Bologna)

The integration between the LISATek flow and the MPARM virtual platform has completed in year 3. As many embedded systems today deploy multiple instantiations of very-long-instruction-word (VLIW) processors for dataprocessing, integration efforts have aimed at developing a VLIW core suitable for multi-instantiation in the MPARM virtual platform. A VLIW architecuture compatible with the VEX instruction set (a simplified version of the STMicroelectronics' ST230 ISA) has been developed in LISA.

Significant effort has been devoted to exploiting the capabilitiy of the LISATek tools to generate synthesizable VHDL, if a suitable sub-set of the LISA syntax is utilized for the processor description. A 4-stage pipelined architecture has been described.

A SystemC model for virtual platform integration and VHDL model for synthesis have been automatically generated using LISATek tools. The VHDL version was synthesized from standard cells using Synopsys Design Compiler. In a first phase, UMC 0.13 µm technology with Design Compiler version 2004.12-SP2 was used and in a second phase there was a migration to a newer TSMC 90nm technology with the newer Synopsys Physical Compiler Y-2006.06.

Memory Aware Compilation and Simulation Tool-Chain for Energy Optimizations (Bologna, Dortmund)

In this activity Bologna has focused on developing a software infrastructure for compiler-based parallelization for MPSoC platforms. One of the key components of any compiler-parallelized code is barrier instructions which are used to perform global synchronization across parallel processors. As compared to programmer-parallelized codes, compiler-parallelized codes can contain larger number of barriers, mainly because a compiler has to be conservative in parallelizing an application (to preserve the original sequential semantics of the program), and this means, in most cases, inserting extra barrier instructions in the code.

Bologna has worked towards the implementation of MPSoC-suitable lightweight runtime synchronization facilities used by a parallelizing compiler frontend, with particular emphasis on barrier implementation. In order to avoid overheads due to multiple software layers the approach does not require OS support.

Operating System Integrated Energy Aware Scratchpad Allocation Strategies for Multiprocess Applications (Bologna, Dortmund)

Various scratchpad allocation strategies have been developed in the past. Most of them target the reduction of energy consumption. These approaches share the necessity of having direct access to the scratchpad memory. In earlier embedded systems this was always true, but with the increasing complexity of tasks systems have to perform, an additional operating system layer between the hardware and the application is becoming mandatory. This work presents an approach to integrate a scratchpad memory manager into the operating system. The goal is to minimize energy consumption. In contrast to previous work, compile time knowledge about the application's behavior is taken into account. A set of fast heuristic allocation methods is proposed in this work. An in-depth study and comparison of achieved energy savings and cycle reductions was performed.

Energy Efficient Cooperative Scheduling and Memory Allocation Techniques for Multiprocess Systems (Bologna, Dortmund)

The increasing amount of functionality in contemporary embedded systems implies the usage of complex software where execution of multiple processes is the common case. Usually the processes are interrupted at an arbitrary point in time. In such a scenario the energy savings achieved by utilization of small and therefore fast and energy efficient scratchpad memories could easily be diminished by excessive copy overhead on each context switch. Therefore this

work tackles this problem by using compile time knowledge and profiling result to define energy and runtime efficient points in the code, where a context switch could be performed with least overhead. The work presented here applies source-level transformations to the code through insertion of context switch points. Basically it provides cooperative scheduling at source-level under the constraint of a preferred time-slice length, guarantied maximum deviation from this time-slice length and energy efficient placement of context switch points. The source-level compile-time transformations have been developed at the University of Dortmund. A new lightweight scheduling layer has been implemented for the MPARM simulation platform from the University of Bologna. This setup has been used for gathering required profile data and final runtime results.

Worst-case execution-time aware compilation (Dortmund, AbsInt)

The two partners cooperated on establishing a link between the two subdomains of this cluster. The integration of tools from the two domains led to first results, which were published [Fal07, Lok07]. Details are described in the compiler cluster report.

Resource aware design space exploration (DTU, ETH Zürich)

The Technical University of Denmark (DTU) has focussed its activities in resource aware design space exploration on run-time resource optimization. Based on an extension of our ARTS multiprocessor simulation framework which allows for handling dynamic refonfiguration, which accounts for both communication and reconfiguration overhead, DTU have conducted a set of experiments aimed at gaining a better understanding of the dynamic behavior of coprocessor-coupled reconfigurable systems. The first study has been based on an MP3 decoder application and a simple ”worst case” resource management algorithm which enforces many run-time reallocations of subsets of the application and, hence, many reconfigurations. The study has focused on coprocessor-coupled architectures where the architecture is partitioned into a homogeneous array of reconfigurable unites (RUs). DTU have studied the impact of different numbers and sizes of RUs, as well as the number of reconfiguration contexts on each RU and the granularity of the RU, i.e. fine or coarse grained, on the run-time behavior of the system. The conclusion from this study [WuMa07] showed that it is possible to gain performance from such architectures. Based on these experiments, DTU has explored various run-time resource management polices and how they impact the system performance. The results of these experiments [Wu07] have been submitted to the International Conference on Field-Programmable Technology 2007. This work will be continued.

The work on multi-objective design space exploration environment based on the PISA environment for multi-objective optimization from the group of Lothar Thiele, ETH Zurich, was completed with an invited talk at DIPES 2006, October 2006, Braga, Portugal and the publication [Mad06].

Interfaces for real-time components (EPFL Lausanne, ETH Zürich)
According to the workplan, there have been major activities in the area of interfaces for real-time components. EPFL and ETH Zürich have continued working on developing interface formalisms and algorithms for interface compatibility checking for interfaces that expose timing and resource constraints of components. Concretely, the partners hope to understand better the differences and commonalities between their interface formalisms, in order to combine or generalize them. There have been visits from ETH Zurich to EPFL Lausanne and a presentation of the concept of Modular Performance Analysis with Interfaces by Nikolay Stoimenov of ETH Zurich. Particular results of the cooperation are described in the publications listed in section 2.3.2. In partcicular, we have been able to formally describe the interface algebra that has been used at ETH Zurich in terms of the notation introduced by Henzinger and D'Alfaro. Finally, the new concepts could be applied to interface-based rate analysis of embedded systems.

As a result of these discussions, there is a joint participation of both groups in the FP7 project COMBEST (lead by Joseph Sifakis) which shows the achieved degree of integration.

Timing Analysis and Timing Predictability (USaar and AbsInt)
First hard analytical results have been obtained about the predictability of architectural features, in this case cache replacement strategies. These show that the replacement strategy has a strong influence on the precision of any type of cache analysis.

The formal derivation of abstract processor timing models has been mostly implemented. This process starts from a specification of the hardware architecture in VHDL and proceeds by a series of analyses and transformations. Analyses of such models for several kinds of proerties will be possible once formally derived abstract architectural models are available.

Preemptive scheduling of hard real-time tasks requires precise estimations of context-switch costs. These are largely dependent on the cache-refill costs caused by pre-empting tasks. An approach has been developed and implemented that estimates and even minimizes the cache interference of tasks. The latter optimization uses the memory allocation to define the cache mapping.

An integration of AbsInt's aiT timing-analysis tool with the ASCET specification and synthesis tool of ETAS has been realized, and experimental results about the effect have been produced.

2.4 Current Results

2.4.1 Technical Achievements

Fault-tolerant Embedded Systems Design (DTU, Linköping)

As part of the collaboration between Linköping University and DTU and as a continuation of the work in the previous years, an approach to the synthesis of fault-tolerant schedules for embedded applications with soft and hard real-time constraints has been developed. The goal is to guarantee the deadlines for the hard processes even in the case of faults, while maximizing the overall utility. Time/utility functions are used to capture the utility of soft processes. Process re-execution is employed to recover from multiple faults. A single static schedule computed off-line is not fault tolerant and is pessimistic in terms of utility, while a purely online approach, which computes a new schedule every time a process fails or completes, incurs an unacceptable overhead. Thus, a quasi-static scheduling strategy is used, where a set of schedules is synthesized off-line and, at run time, the scheduler will select the right schedule based on the occurrence of faults and the actual execution times of processes. Moreover, a preemption technique is elaborated as a method to generate flexible schedules that maximize the overall utility for the average case while guarantee timing constraints in the worst case. The scheduling algorithm determines off-line when to preempt and when to resurrect processes. Prof. Paul Pop from DTU has visited Linköping with several occasions during this period. The following joint publications have been generated: [Izo08], [Ele08], [Izo2-08].

Resource analysis using price timed automata (DTU)

DTU has done initial work aimed at using the formalism of price timed automata to model and analyze resources of simple embedded systems. Analysis of resource consumption of embedded systems is a major challenge. DTU has focused on the analysis of timing and memory access costs of those models, using UPPAAL Cora. Although the experiments performed are on small and rather basic models, we have demonstrated [OBH08] that priced timed automata can be used to reason about resource consumption of embedded systems.

Run-time resource management (DTU)

DTU has continued its work on run-time resource optimization. The experiments on dynamic reconfiguration on a coprocessor-coupled architecture, based on an extension of our ARTS multiprocessor simulation framework, have been continued in year 4. Experiments with various run-time resource management polices have shown that it is possible to gain performance from such architectures and have suggested some general guidelines for obtaining efficient run-time resource management [WuHaMa08].

Predictability for Multiprocessor SoC Architectures (Bologna, Braunschweig, Linköping)

One of the major issues in the context of predictability for multiprocessor systems is the shared communication infrastructure. The traffic on the bus does not solely originate from data transfers due to data dependencies between tasks, but is also affected by memory transfers as result of cache misses. A bus access policy and bus access schedule have to be developed which (1) guarantee predictability and (2) provide efficiency in terms of system performance. In the previous year we have developed an overall strategy and framework for predictable multiprocessor applications. Now, in year four, we have addressed the issues of bus access optimization and bus controller design. Bus access optimization is crucial in achieving predictability while, at the same time, maintaining efficiency in terms of performance. In order to demonstrate the practicality of the approach, we have designed and synthesised adequate bus controllers fro the proposed protocols.

A master student from Bologna, Paolo Burgio, has spent the period June 2007 – April 2008 at Linköping working, in particular, on controller design. The Symta/P tool for WCET analysis, from Braunschweig, has been used in this project.

Publications in the last period: [And 08], [Ros 07]. Joint publication is under way.

Integration of MPARM and MEMSIM into one simulation platform (Dortmund, Bologna)

MPARM is a multiprocessor simulation platform developed at the University of Bologna. The simulator offers a cycle-true simulation of a ARM7 based multiprocessor SoC. The simulation includes a configurable number of processors and associated memories. Furthermore, additional hardware devices can be simulated. To some extend the memory hierarchy of each processor can be configured. Presence of a caches, as well as size and number of scratchpads can be configured, but there are obvious limitations to the reconfigurability of the whole system. For example, only an one-level cache hierarchy is available. Another restriction forces every processor to implement the same memory hierarchy.

Complementary, at TU Dortmund a versatile memory hierarchy simulator has been developed. It offers the simulation of a free configurable multi-level memory hierarchy. The user provides a memory hierarchy description and an processor-based memory access sequence to MEMSIM, which computes the accumulated timing and energy values for that input. Limitations of that approach are the simulation of a single execution trace (i. e. single processor system) and the missing simulation of the processing unit.

An integration of both simulation platforms has been accomplished in the context of Artist. The resulting MPARM/MEMSIM simulator is capable of simulating a ARM-based multiprocessor SoC with various memory hierarchies. Each processing unit may have a different memory hierarchy and different cycle and energy values. The most challenging task of merge both simulation platforms while ensuring a cycle true execution in the simulation environment has

been successfully performed. The resulting simulator offers a shared memory platform with one AMBA bus interconnect. Up to 32 processing units with MEMSIM-defined local memory hierarchies (i. e. local caches, scratchpads etc.) can be attached to this bus. The bus-attached main memory of the original MPARM has been replaced with another MEMSIM component, which allows for defining memory hierarchies instead of plain SRAMs. Since this versatile configurable memory hierarchy is not covered by the access values provided in MPARM, an interface to CACTI has been integrated to automatically generate reliable values for a particular memory hierarchy.

Memory hierarchy design-space-exploration for the MPARM/MEMSIM Simulator (Dortmunt, Bologna)

Based on this new simulation platform, an approach for automatic design space exploration has been developed. The approaches utilizes SPEA2 – an advanced genetic algorithm – for optimizing the memory hierarchy for a particular application. The memory hierarchy can be optimized according to user defined weights in the 3-dimensional space: cycles - energy - die-size. The algorithm produces Pareto-optimal solutions, where from according to the wights the most efficient is selected. During the optimization run thousands of simulation runs with various memory hierarchies may become necessary for the estimation of the fitness of individuals in the genetic pool. For this task the MPARM/MEMSIM simulator has been successfully utilized. The results show that the approach was capable of finding a well suited, efficient and reasonable memory hierarchy for a particular application.

Parallelization of MIMO benchmark using MAPS tools (Aachen, Bologna)

MAPS (MPSoC Application Programming Studio) tools have been developed at RWTH Aachen aiming to help the developers efficiently parallelize the software for embedded parallel architectures. In this activity the MIMO benchmark from Bologna has been evaluated and parallelized using the MAPS compiler. Unlike a fully autonomous parallel compiler, MAPS provides a set of tools and technologies which helps the parallelization process, for example, advanced source-level profiling and data-flow driven partitioning. A novel concept, called CB (Coupled Block), is proposed to capture the coarse-grained task parallelism on the compiler IR level. TCT (Tightly-Coupled Thread) MPSoC, from Aachen's partner TokyoTech, has been used in the framework (shown in the Fig. 1) as a backend to evaluate the parallelization result. The experiment showed that MAPS could quickly spot the parallelism opportunities in the sequential application and raise the partitioning suggestions to the designer as good starting points. Together with the array analysis capability for the critical loops, the designer could understand the dependency flow easily, which is usually time-consuming in the manual parallelization process. Overall the parallelization productivity is improved. More details of the MAPS tools can be found in [1] and [2].

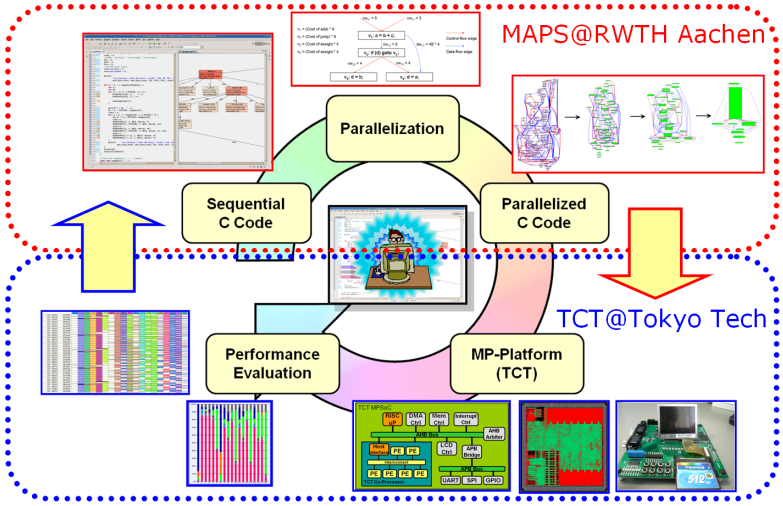

Fig. 1 MAPS-TCT Framework

Resource Aware Design Space Exploration and Mapping (ETHZ, BOLOGNA)

ETHZ has been working together with BOLOGNA in order to establish a joint methodology in order to map algorithms onto multi-processor platforms under resource constraints. In particular, the MPARM environment from BOLOGNA will be used as a simulation backend. The concrete results are as follows:

- Specification of the application using a process network, including an XML based specification syntax.
- Specification of the target architecture, again using a XML description.
- Specification of the mapping while taking different scheduling disciplines into account. A corresponding XML format has been defined.
- Mutli-objective design space exploration using a component-based analysis methodology named MPA-RTC (Modular Performance Analysis – Real-Time Calculus).
- Joined work on the generation of the necessary hardware-dependent software and the compilation towards the target platform has started.

Formal Performance Analysis and Resource-Aware Mapping (ETHZ, EPF Lausanne)

ETHZ has been extending the modular performance analysis method using results from interface theory as provided by EPF Lausanne. This way, it is possible to define resource-aware component interfaces. In addition, ETHZ has been developing a method that allows to map dynamic applications optimally to multi-processor platforms (MPSoC) under static (leakage) and dynamic energy requirements. This method has been intensively discussed with BOLOGNA and comparisons have been made with the methodology based on Benders decomposition (ILP and constraint programming).

Timing Analysis and Timing Predictability (USaar and AbsInt)

The notion of predictability of cache architectures has been clarified. This is the first precise notion of predictability found in the literature. It turns out that the cache-replacement strategy is the decisive characteristic for the predictability of architecture. 4 different replacement strategies were compared, and the LRU strategy was found to be optimal.

The PREDATOR project in the 7th Framework Programme attempts to reconcile performance and predictability. It has identified the PROMPT (PREdictability Of Mulit-processor Timing) design rules for predictable multi-processor design. The first principles are to avoid interference on shared resources in the architecture and to allow the application designer the mapping of applications to target architecture without the introduction of new interferences that were not present in the application.

Parametric timing analysis (USaar and Mälardalen)
Timing analyses require that information such as bounds on the maximum numbers of loop iterations are known statically, i.e., during design time. Parametric timing analysis softens these requirements: it yields symbolic formulas instead of single numeric values representing the upper bound on the task's execution time. So, some input parameters to the program can remain unknown until the final use of the task. The developed analysis determines the parameters of the program, constructs parametric loop bounds, takes processor behaviour into account and attains a formula automatically.

Synergy between Code Synthesis and Timing Analysis (USaar and AbsInt)
One of the problems to be solved synergetically by code synthesis compiling, and timing-analysis is to support mode-specific timing analyses. Many embedded control systems have several operating modes with different timing requirements. Some operating modes are not explicitly specified on the model level or in comments on the C level. Saarland University in cooperation with Bosch (within the PREDATOR project), attempts to semi-automatically identify such operating modes. This would allow timing analysis to implement mode-specific execution-time bounds which can be used to improve schedulability of task sets.

WCET Analysis for Systems with preemptive scheduling (USaar and Absint)
Derivation of timing guarantees was extended in order to cope with the systems with preemptive scheduling. A new method to compute valid upper bounds on a task's worst case execution time (WCET) under preemption was proposed.. This method approximates an optimal memory layout such that the set of possibly evicted cache-entries during preemption is minimized. This set then delivers information to bound the execution time of tasks under preemption in an adopted WCET analysis.

2.4.2 Individual Publications Resulting from these Achievements

Linköping
List of Publications

[And 08] Alexandru Andrei, Petru Eles, Zebo Peng, Jakob Rosén: Predictable Implementation of Real-Time Applications on Multiprocessor Systems on Chip, 21st Intl. Conference on VLSI Design, January 4-8, 2008, Hyderabad, India, pp. 103-110.

[Ros 07] Jakob Rosén, Alexandru Andrei, Petru Eles, Zebo Peng: Bus Access Optimization for Predictable Implementation of Real-Time Applications on Multiprocessor Systems-on-Chip, 28th IEEE Real-Time Systems Symposium (RTSS'07), December 3-6, 2007, Tucson, Arizona, USA, pp. 49-60.

Bologna

[Srini08] Srinivasan, S.; Li, L.; Ruggiero, M.; Angiolini, F.; Vijaykrishnan, N.; Benini, L.; " Exploring architectural solutions for energy optimisations in bus-based system-on-chip", Computers & Digital Techniques, IET Volume 2, Issue 5, pp. 347-354, 2008.

[Khat08] I. Al Khatib, F. Poletti, D. Bertozzi, L. Benini, M. Bechara, H. Khalifeh, A. Jantsch, R. Nabiev, "A multiprocessor system-on-chip for real-time biomedical monitoring and analysis: ECG prototype architectural design space exploration", ACM Trans. Des. Autom. Electron. Syst., Volume 13, No. 2, pp.1-21, 2008.

Dortmund

[Verm07] Manish Verma, Peter Marwedel: Advanced Memory Optimization Techniques for Low-Power Embedded Processors, Springer, 2007

[Marw08] Peter Marwedel: MIMOLA - A fully synthesizable language, in: Prabhat Mishra, Nikil Dutt (ed.): Processor Description Languages - Applications and Methodologies, Morgan Kaufman, 2008

Aachen

[Ceng08] Ceng, J., Castrillon, J., Sheng, W., Scharwächter, H., Leupers, R., Ascheid, G. and H. Meyr, RWTH Aachen University, Isshiki, T. and H. Kunieda, Tokyo Institute of Technology. "MAPS: An Integrated Framework for MPSoC Application Parallelization". In 45th Design Automation Conference (DAC '08), Anaheim, CA, USA, June 2008

DTU

[WuHaMa08] Kehuai Wu, Esben Rosenlund Hansen, Jan Madsen, Towards Understanding and Managing the Dynamic Behavior of Run-Time Reconfigurable Architectures, to appear in the proceedings of the Conference on Engineering of Reconfigurable Systems and Algorithms (ERSA'08), July 2008.

[OBH08] T. Ovatman, A. Brekling, M.R. Hansen, Cost Analysis for Embedded Systems: Experiments with Priced Timed Automata, In proceedings of Formal Foundations of Embedded Software and Component-Based Software Architectures (FESCA) March 2008.

Linköping

[And 08] Alexandru Andrei, Petru Eles, Zebo Peng, Jakob Rosén: Predictable Implementation of Real-Time Applications on Multiprocessor Systems on Chip, 21st Intl. Conference on VLSI Design, January 4-8, 2008, Hyderabad, India, pp. 103-110.

[Ros 07] Jakob Rosén, Alexandru Andrei, Petru Eles, Zebo Peng: Bus Access Optimization for Predictable Implementation of Real-Time Applications on Multiprocessor Systems-on-Chip, 28th IEEE Real-Time Systems Symposium (RTSS'07), December 3-6, 2007, Tucson, Arizona, USA, pp. 49-60.

Saarland

[Rein07] Jan Reineke, Daniel Grund, Christoph Berg, and Reinhard Wilhelm. Timing Predictability of Cache Replacement Policies. *Real-Time Systems*, 37(2):99-122, November 2007.

[Rein07] Jan Reineke and Daniel Grund. Relative Competitive Analysis of Cache Replacement Policies. In *LCTES '08: Proceedings of the 2008 ACM SIGPLAN-SIGBED conference on*

Languages, compilers, and tools for embedded systems, pages 51-60, New York, NY, USA, June 2008. ACM.

ETH Zürich

[HT07] Wolfgang Haid, Lothar Thiele: Complex Task Activation Schemes in System Level Performance Analysis. Proc. 5th Intl Conf. on Hardware/Software Codesign and System Synthesis (CODES+ISSS 2007), ACM Press, Salzburg, Austria, pages 173-178, October, 2007.

[PCTT07] Linh Phan, Samarjit Chakraborty, P. S. Thiagarajan, Lothar Thiele: Composing Functional and State-based Performance Models for Analyzing Heterogeneous Real-Time Systems. 28th IEEE Real-Time Systems Symposium (RTSS), Tucson, Arizona, US, December, 2007.

2.4.3 Interaction and Building Excellence between Partners

Interaction between Linköping and DTU has been in the area of fault tolerant embedded systems. Solutions have been developed in cooperation and publications have been written together. During the fourth year Prof. Paul Pop from DTU has visited Linköping.

Interaction between Linköping and Bologna has been in the area of predictable multiprocessor systems. Paolo Burgio has visited Linköping for 10 months and has worked on the elaboration of bus controllers. In this period he has also written his Master Thesis. Experiments at Linköping has been run using the MPARM tool from Bologna.

The activities of Bologna have required interaction with Aachen and Dortmund. Interaction with Aachen was required on documenting and delivering a complex, computationally intensive benchmark (MIMO decoding) for compiler-based parallelization by AACHEN TOOLS. Students and Post-docs of AACHEN and BOLOGNA interacted several times by email, skype and in person during the DATE conference.

Interaction between BOLOGNA and DORTMUND was focused on the joint development of multi-core simulation platforms, with special emphasis on accurate memory system simulation. Several conference calls and an half-day meeting, held during the DATE conference were needed to discuss tool integration.

There was close interaction between BOLOGNA, USAAR and ETHZ on (a) the integration of performance analysis and mapping with the MPARM simulation environment. 2 Phd students from ETHZ visited Bologna for a weak in order to larify all necessary technical details. In addition, ETHZ and USAAR cooperated in the area of resource aware scheduling; in particular, several meetings took place including a visit of a PhD student of USAAR in Zurich. Main area of cooperation are new ways to extend the method of abstract interpretation beyond WCET analysis, i.e. towards scheduling analysis.

2.4.4 Joint Publications Resulting from these Achievements

[Izo08] Viacheslav Izosimov, Paul Pop, Petru Eles, Zebo Peng: Synthesis of Flexible Fault-Tolerant Schedules with Preemption for Mixed Soft and Hard Real-Time Systems, 11th EUROMICRO CONFERENCE on DIGITAL SYSTEM DESIGN (DSD 2008), Parma, Italy, September 3-5, 2008.

[Ele08] Petru Eles, Viacheslav Izosimov, Paul Pop, Zebo Peng: Synthesis of Fault-Tolerant Embedded Systems, Design, Automation, and Test in Europe (DATE 2008), Munich, Germany, March 10-14, 2008, pp. 1117-1122.

[Izo2-08] Viacheslav Izosimov, Paul Pop, Petru Eles, Zebo Peng: Scheduling of Fault-Tolerant Embedded Systems with Soft and Hard Timing Constraints, Design, Automation, and Test in Europe (DATE 2008), Munich, Germany, March 10-14, 2008, 915-920.

[Mah-08] S. Mahadevan, F. Angiolini, J. Sparso, L. Benini, J. Madsen, "A Reactive and Cycle-True IP Emulator for MPSoC Exploration, " IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Volume 27, Issue 1, pp 109-122, 2008.

[JPTY08] Bengt Jonsson, Simon Perathoner, Lothar Thiele, Wang Yi: Cyclic Dependencies in Modular Performance Analysis ACM & IEEE International Conference on Embedded Software (EMSOFT), ACM Press, Atlanta, Georgia, USA, October, 2008.

[Per07] Simon Perathoner, Ernesto Wandeler, Lothar Thiele, Arne Hamann, Simon Schliecker, Rafik Henia, Razvan Racu, Rolf Ernst, Michael González Harbour: Influence of Different System Abstractions on the Performance Analysis of Distributed Real-Time Systems. ACM & IEEE International Conference on Embedded Software (EMSOFT), ACM Press, Salzburg, Austria, pages 193-202, October, 2007.

[Spo07] Thomas Sporer, Andreas Franck, Iuliana Bacivarov, Michael Beckinger, Wolfgang Haid, Kai Huang, Lothar Thiele, Pier Paolucci, Piergiovanni Bazzana, Piero Vicini, Jianjiang Ceng, Stefan Kraemer, Rainer Leupers: SHAPES - a Scalable Parallel HW/SW Architecture Applied to Wave Field Synthesis. Proc. 32nd Intl Audio Engineering Society (AES) Conference, Audio Engineering Society, Hillerod, Denmark, pages 175-187, September, 2007.

[Altm07] Sebastian Altmeyer, Christian Hümbert, Björn Lisper, and Reinhard Wilhelm: Parametric timing analysis for complex architectures. Proc. 14th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA 08), Kaohsiung, Taiwan, Aug. 2008

[Altm08] Sebastian Altmeyer and Gernot Gebhard. WCET Analysis for Preemptive Systems. In Raimund Kirner, editor, *Proceedings of the 8th International Workshop on Worst-Case Execution Time (WCET) Analysis*, pages 105-112, Prague, Czech Republic, July 2008. OCG.

2.4.5 Keynotes, Workshops, Tutorials

Invited Lecture: Luca Benini - Optimization-centric embedded systems design: addressing the multi-core challenge ETHZ Computational Optimization mini-symposium, May 2008. The tutorial covered issues related to optimal, resource-aware allocation and scheduling for multi-core platforms

Tutorial: Lothar Thiele - Complex Embedded System Design. Lausanne, Sept. 10, 2007.

Tutorial: Lother Thiele - Analysis of Distributed Embedded Systems. CASTENESS WORKSHOP ROMA, Jan. 15-18, 2008.

Invited Talk: Lothar Thiele - Workshop Mapping Algorithms onto MPSOC. Germany, June 16-17. 2008.

Invited Talk: Lothar Thiele - MPSOC Conference. Aachen, Germany, June 23-27, 2008.

Summer School: Lothar Thiele - Embedded Systems, Florianopolis, Brasil, August 25-2, 2008.

Summer School: Lothar Thiele - Embedded Systems, Grenoble, France, Sept 8-12, 2008.

Invited talk: Petru Eles - Synthesis of Fault-Tolerant Embedded Systems DATE 2008 Conference, Munich, Germany - March 10, 2008 As part of the special day on Dependable Embedded Systems: With this occasion several results obtained in the ARTIST context have been made accessible to an international audience. They are related, in particular, to fault tolerance aspects of distributed real-time systems like those used in automotive applications.

Tutorial: Rainer Leupers, Peter Marwedel: Retargetable Compilation, (1 week course), ALARI, Feb. 11-15, 2008. This course is part of ALARI's master programmes for degrees in Embedded System Design (see www.alari.ch).

Tutorial: Peter Marwedel, Embedded Systems in a Nutshell (full day tutorial), Spring School on Knowledge Discovery in Ubiquitous Systems, March 2, 2008, Porto, www.kdubiq.org. This tutorial provided a brief overview over specification techniques, hardware, scheduling and optimization of embedded systems for a community without any pre-existing knowledge on embedded systems.

Workshop: Heiko Falk (Workshop Chair), Peter Marwedel (Workshop Publicity): SCOPES, March 13-14, 2008, Munich, www.scopesconf.org SCOPES is a specialized workshop on compilation and other software generation tools for embedded systems.

Tutorial: Peter Marwedel, Heiko Falk: Embedded Systems -with Emphasis on the Exploitation of the Memory Hierarchy- (Full week tutorial), Advanced Institute of Information Technology, Aug. 11-15, 2008, Seoul, ttt.aiit.or.kr/. The tutorial consisted of two parts: in the first part, an overview over specification techniques and techniques for mapping of applications to multiprocessor systems was provided. In the second part, the tutorial focused on compilation techniques exploiting descriptions of the memory architecture.

Tutorial: Peter Marwedel:, Memory architecture aware compilation for Embedded Systems (10 h tutorial), Artist South American Summer School, Aug. 25.-29., 2008, Florianopolis, Brazil, www.artist-embedded.org, The tutorial focused on compilation techniques exploiting descriptions of the memory architecture.

Invited talk: Peter Marwedel, Heiko Falk: Memory architecture aware compilation, Artist2 Summer School on Embedded Systems, Sept. 10, 2008, Autrans, www.artist-embedded.org. This was a 1 hour talk on the topic of the title.

Mini Keynote: Jan Madsen - Adaptive Embedded Systems Challenges of Run-Time Resource Management. 8th International Forum on Application-Specific Multi-Processor SoC, 23 – 27 June 2008, Château St. Gerlach http://www.chateauhotels.nl/ENG/index1.htm Valkenburg a.d. Geul, the Netherlands. This presentation addressed some of the challenges of run-time resource management in adaptive embedded systems. The challenges come from the often very complicated interplay between the application, the application mapping, and the underlying hardware architecture.

Keynote: Jan Madsen - On Line, Real Time Darwinism - Challenges of the Future. DIAG Conference 2008, September 4-5, Skjoldenæsholm, Jystrup, Denmark. This presentation

gave an overview of the evolution of adaptive embedded systems platforms, with a focus on resource management such as performance, low power, and reliability.

3. Future Work and Evolution

3.1 Milestones

Milestone defined for Year 4: "A methodology for resource-aware design of embedded systems; this methodology will include mechanisms for handling optimizations across different levels of abstraction"

Year 4 demonstrated several steps toward the definition of a joint methodology for resource-aware design and optimizations. Examples of integrations are the joint work on application parallelization carried out by AACHEN and BOLOGNA and the integration of memory model and design space exploration carried out by BOLOGNA and DORTMUND. Furthermore, the partners are now involed in a number of 7FP projects (PREDATOR, COMBEST, MNEMEE) where the preliminary integration work performed in ARTIST is demonstrating its seminal value.

3.2 Indicators for Integration

Dortmund, ETHZ, Saarland and Bologna are partners of the project ICT PREDATOR, started in January 2008. This STREP project originated from joint ARTIST2 work on resource aware design. The project is concerned with embedded systems that are characterized by efficiency requirements on the one hand and critical constraints on the other. Such systems occur in application domains such as automotive and avionics where the increase in the number of functions can no longer be coped with by adding one Electronic Control Unit or processor board per function. Instead the trend in both application domains goes towards the integration of many functions on powerful hardware platforms. The goal of the project is to define and develop predictable and efficient platforms for avionics and automotive. Industrial partners AIRBUS, BOSH and ABSINT are involved. The PREDATOR project is a clear example of tangible integration indicator, as it involves not only research partners from the ARTIST resource-aware-design cluster, but alsto large industrial partners interested in the uptake of this research in industrial methods and architectures (http://www.predator-project.eu/)

Mnemee (Memory management technology for adaptive and efficient design of embedded systems) is a small and medium scale focused research project involving IMEC, the ICD technology transfer center at Dortmund, TU Eindhoven, Thales Communications, Intracom (Greece) and the National Technical University University of Athens. The focus is on the exploitation of the memory system (http://www.mnemee.org)

3.3 Main Funding

Other sources of funding include:

- Linköping: Swedish Foundation for Strategic Research (SSF)
- Bologna: STMicroelectronics, direct industrial grant
- Bologna, ETHZ, DORTMUND, SAARLAND: ICT-PREDATOR STREP project (7FP).
- ETHZ: ICT-Project COMBEST (FP7)
- ETHZ: ICT-Project SHAPES

4. Internal Reviewers for this Deliverable

Prof. Michela Milano, Constraint Programming and Optimization Group, Dipartimento di Elettronica, Informatica e Sistemistica, Università di Bologna