IST-004527 ARTIST2 NoE   Year 3
Cluster: Execution Platforms, Compilers & TA   D15-EP-Y3
Activity: Resource Aware Design

IST-004527 ARTIST2
Network of Excellence
on Embedded Systems Design

Activity Progress Report for Year 3

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.*

IST-004527 ARTIST2 NoE   Year 3
Cluster: Execution Platforms, Compilers & TA   D15-EP-Y3
Activity: Resource Aware Design

Table of Contents

IST-004527 ARTIST2 NoE   Year 3
Cluster: Execution Platforms, Compilers & TA   D15-EP-Y3
Activity: Resource Aware Design

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

Prof. Dr. Tom Henzinger – EPFL (Switzerland)
Virtual machines for hard real-time computing, Embedded Machine; Schedule-Carrying Code

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. Reinhard Wilhelm - University of Saarbrücken (Germany)
Research on Timing Analysis and Timing Predictability

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

1.2 Affiliated Participants and Roles

Roberto Zafalon – STM (Italy)
Dynamically controlling power consumption in MPSoC platforms.

1.3 Starting Date, and Expected Ending Date

Starting date: September 1st, 2004.

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

Limited resources are a key characteristic of embedded systems. For example, no solution to the problem of very limited energy supply is in sight. Therefore, resource aware design will continue to play a key role well beyond the end of ARTIST2 funding.

1.4 Baseline

The importance of resource awareness in embedded systems is growing rapidly. The limited availability of computing resources is preventing the introduction of new products and applications, especially in areas where high-performance embedded systems are required (e.g. in telecom and consumer markets). Resources include energy, computational power and hardware components.

IST-004527 ARTIST2 NoE   Year 3
Cluster: Execution Platforms, Compilers & TA   D15-EP-Y3
Activity: Resource Aware Design

Minimization of the energy consumption plays a major role in the design of embedded systems. Limited availability of energy is the dominating constraint for many advanced embedded systems, in particular those involving multimedia or sensor technologies. At the start of the project, only a very limited number of energy models were available. Optimizing systems for minimized energy consumption at the software level was more an exception. Exploitation of memory hierarchies for minimizing the energy consumption was at an infant state.

Also, maximizing the computational power of embedded systems is becoming of increasing importance due to the wide-spread deployment of high-performance multimedia-enabled devices in the market. The increasing trend towards encrypted communication also increases the performance requirements. As a result the efficient use of available hardware components, most importantly of processors and memories, is mandatory. When the project was started, customized processors were in use, but the generation of tool chains for processors was a mainly manual process. Support for multiprocessor systems was almost completely lacking.

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 3

Due to the complexity of resource aware design, work on the problem continued in its third year. The specific focus of this activity was on linking research on execution platforms and on compilers. As a result of the work in the previous years, integrated tools from various research organizations were available. Year 3 is characterized by the use of the tools and tighter integrations on a large scale. This general approach has been applied by all the groups in order to tackle specific subproblems.

Links between research on compilers and architectures were used to solve the following subproblems:

- Memory systems are key consumers of the scarcest resource in embedded systems, electrical energy. The energy "consumption" (actially a conversion into heat) of memory systems can be reduced by making the compiler aware of the memory architecture. This leads to "memory architecture aware compilation". Not much is known about this area. We tackled the problem of designing proper memory representations and optimizations.

- The energy consumption can also be reduced by using voltage scaling. Previous approaches for voltage scaling are too static, i.e. they cannot exploit additional slack resulting from run-time variations.

- 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.

- A large design space exists for current state of the art systems, comprising several processors, buses and peripheral components. Tradeoffs exist between optimization criteria. Hence, techniques for design space exploration and multi-objective optimization are needed and proposed techniques had to be analysed.

- Separate mechanisms for the description of component interfaces existed, when this network was started. The problem of unifying different description techniques was continued.

IST-004527 ARTIST2 NoE   Year 3
Cluster: Execution Platforms, Compilers & TA   D15-EP-Y3
Activity: Resource Aware Design

- 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. Tradeoffs were analysed in the third year.

- In general, timing predictability was an issue looked at. Motivation and approaches are described in the report on the timing analysis activity. Providing real-time guarantees for resource-efficient systems is a problem tackled by the partners of these activities.

1.6 Comments From Year 2 Review

1.6.1 Reviewers' Comments

„ACCEPTED

General comments apply".

From the general comments sections:

„We found the tables describing the primary participants in each cluster in D2 very useful, and feel that the inclusion of a digital photo of the individual is very helpful, not just for the reviewers, but also for anyone outside of the core members who will invariably run into ARTIST2 members at workshops and conferences.

The consortium should put in place a quality process for deliverable. For example, a document from one cluster should be reviewed by independent people from other clusters.

The consortium should open itself to external views and additional industries. Today too many stakeholders are left over. We would like to see the number of affiliates growing. Following the recommendations from the last review meeting, we are glad to see that a procedure is in place for this on the website.

The consortium might consider addressing the issues related to fault-tolerance. Today it is treated by verification but not at software level and a real-time system cannot meet its deadline in presence of uncontrolled faults.

Each document should have a short conclusion (what are the results compared to expectation – what specific actions will be taken to enhance things, to get more local funding etc.. (for example))."

External funding figures should be given – it gives an idea of the effort in a particular theme.

Some more graphics in the reports may be welcome (to make reading more pleasant and interesting). PDFs delivered with systematically clickable links in all the reports would be good.

1.6.2 How These Have Been Addressed

Photos have been added to the cluster document.

In addition to the proofreading by the members of this activity, this document has also been proofread by a reviewer not involved in ARTIST2 (see section 4).

The network is now open for additional affiliates, as suggested by the reviewers.

DTU and the University of Linköping are cooperating on fault-tolerance issues.

A conclusion has been added for the description of the technical achievements.

Selected external funding figures are available at the cluster level.

IST-004527 ARTIST2 NoE   Year 3
Cluster: Execution Platforms, Compilers & TA   D15-EP-Y3
Activity: Resource Aware Design

The number of graphical elements has been increased with respect to last year's deliverable. Links are now clickable.

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.*

IST-004527 ARTIST2 NoE   Year 3
Cluster: Execution Platforms, Compilers & TA   D15-EP-Y3
Activity: Resource Aware Design

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.
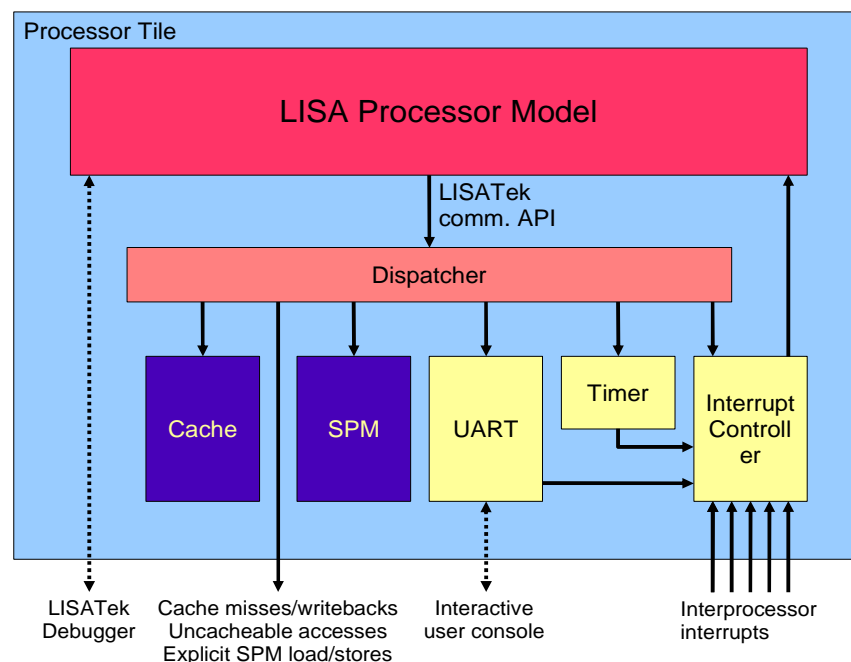
Figure 1 Processor Tile

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
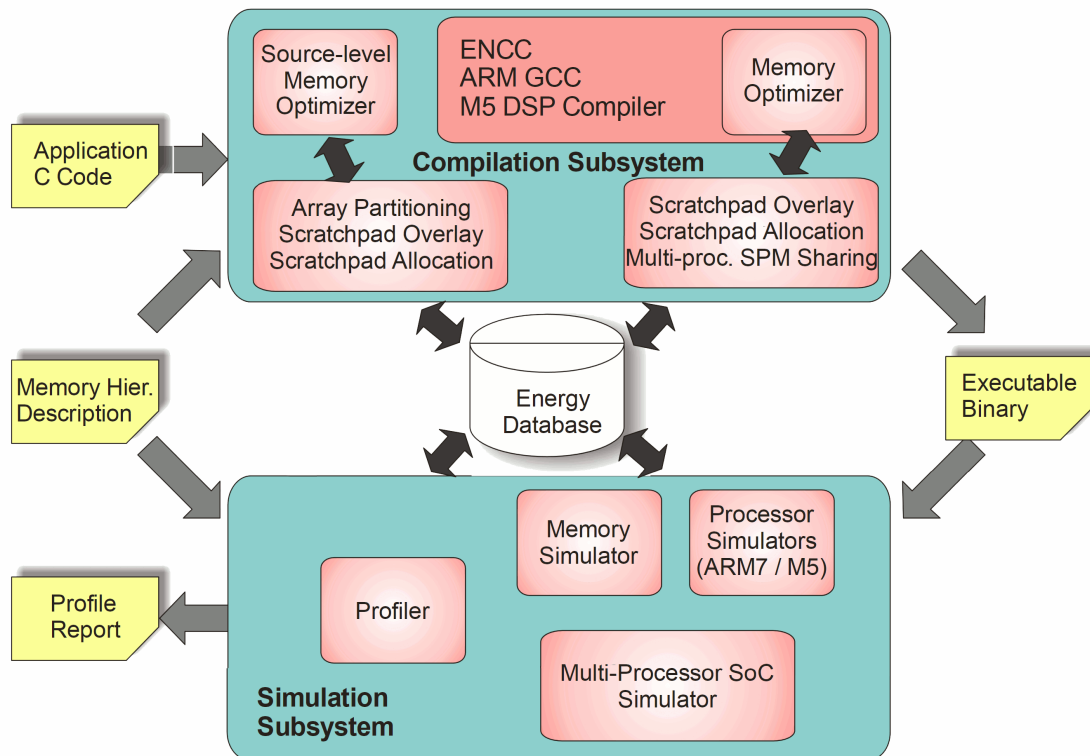
Figure 2: Memory Aware Compilation and Simulation Tool-Chain

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   Current Results

2.3.1 Technical Achievements

The common vision of the partners on resource-aware design led to several joint research proposals.

For example, Bologna, ETH Zürich, Dortmund, Saarbrücken, AbsInt and one partner from another cluster (Scuola Superiore Sant' Anna di Pisa) defined the Collaborative Project PREDATOR. The vision of PREDATOR is that of reconciling performance and predictability requirements across several levels of abstraction. PREDATOR will start in 2008.

A second example is that of the STREP proposal MNEMEE, which will involve IMEC, TU Eindhoven (active in other clusters) and ICD, a spin-off of the University of Dortmund. Efficient use of the memory hierarchy is the focus of MNEMEE.

According to the current technical annex, the goal for year 3 was to generate results from the tools integrated in year 2. A number of cooperations demonstrate that this really took place.

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 integration between the LISATek flow and the MPARM virtual platform has continued with a focus on advanced processor architectures. 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.

IST-004527 ARTIST2 NoE   Year 3
Cluster: Execution Platforms, Compilers & TA   D15-EP-Y3
Activity: Resource Aware Design

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 as depicted in fig. 3.
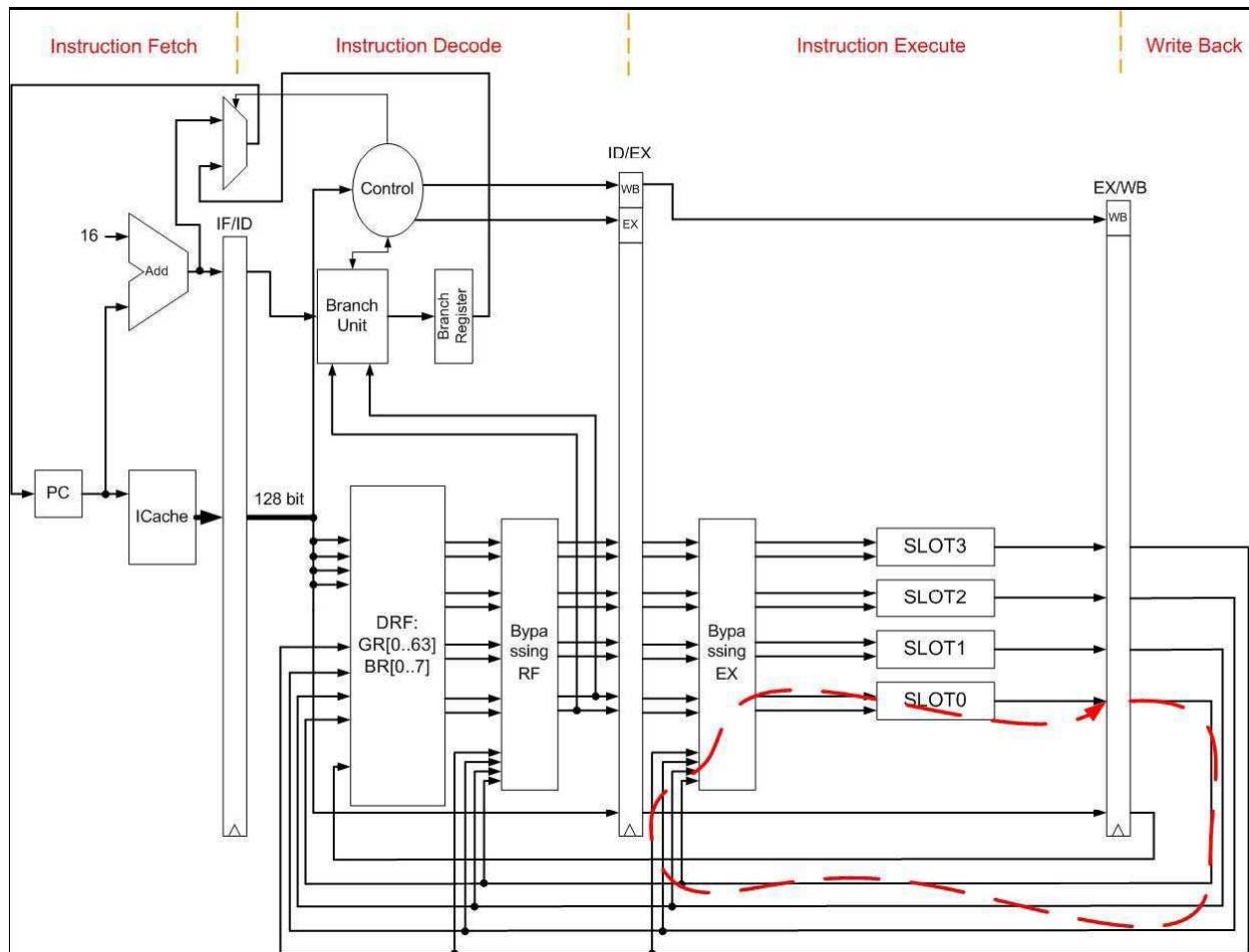

Fig. 3 VLIW processor with bypass

The VLIW has 4 issue slots and a double bypass stage to achieve better parallelism and reduce the number of empty slots. 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.

Web site: http://www-micrel.deis.unibo.it/sitonew/research/mparm.html

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

The very first steps for this work have been done during the previous reporting period. The work has continued during this year and first results are available and have been published. Linköping takes the lead for this work.

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

IST-004527 ARTIST2 NoE   Year 3
Cluster: Execution Platforms, Compilers & TA   D15-EP-Y3
Activity: Resource Aware Design

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 [Jov06]. Results for a journal submission have been generated.

Web site: http://ls12-www.cs.uni-dortmund.de/~marwedel/ artist-mobility.html

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.

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. This runtime library can be coupled with a parallelizing compiler to obtain a fully automated tool flow, as shown in figure 4.
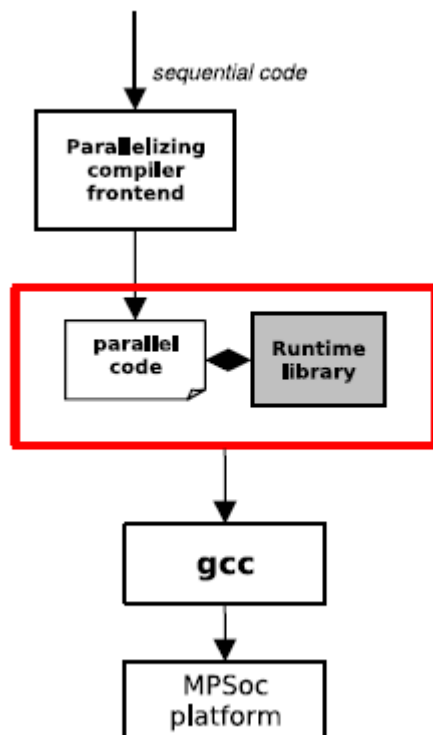
Fig. 4 Approach to parallelization

The parallelizing front-end performs source-to-source code transformations, inserting calls to the parallelization function in the original sequential source code and modifying loop iterators accordingly. The runtime library implemented in this activity can then be linked to the code which is then compiled with the target compiler. The main difference with similar approaches for general purpose computing is that the parallelization function does not assume a shared memory for code nor run-time data-structures. Hence, it can be used for architectures where there is no common shared memory kernel. Moreover, the implementation has been carefully optimized for minimum run-time overhead.

Integration with a parallelizing compiler frontend has been tested with promising results, as detailed in publications [Ben06, Ben07].

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. The required profile data and final runtime results are generated by the MPARM simulation platform developed at the University of Bologna. Fig. 5, comparing different scratchpad allocation strategies, shows an example of the results generated with this approach [Pyk07]. Web site: http://ls12-www.cs.uni-dortmund.de/research/macc
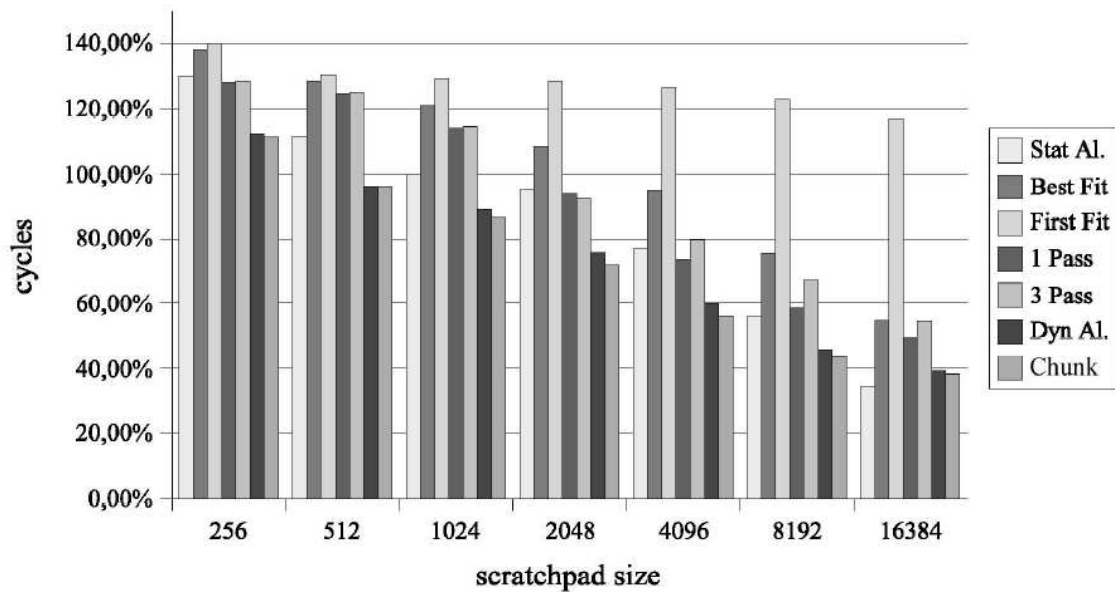
Fig. 5 Runtime for Media+ benchmark for various scratchpad allocation strategies

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

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

During previous reporting periods, 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. The target was to provide a single point configuration for both the simulation and compilation subsystems. In addition, a common energy database is used. Both, the memory optimizers in the compilation subsystem as well as the memory and multi-processor SoC simulators in the simulation subsystem, can access this database. On top of this system an optimizing tool-chain has been developed. It is capable of performing optimizations for a given 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 pursuit further

development of a common memory hierarchy description format, which will be used for developing future optimizations. Web site: http://ls12-www.cs.uni-dortmund.de/research/macc

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.

Fault-tolerant embedded systems design (DTU, Linköping)

The Technical University of Denmark (DTU) and Linköping University started 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 [Pop2-07]. 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, as DTU and Linköping have shown in [Pou07]. In addition, the CLP framework can be used to reason about the effects of voltage scaling on reliability [Pop07]. 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 and joint publications, [Pou07], [Pop07] and [Pop2-07].

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.

Conclusion

Using MPARM as a key simulation platform for generating results became widespread in year 3. In addition, several bilateral research partnerships led to research results. Resource awareness is a common theme for these partnerships. Resource consumption is evaluated in terms of evaluation metrics. Evaluation metrics used by the partners include energy, time, and

memory footprint. These are the criteria which are very much appropriate in embedded systems design. Several techniques improving embedded systems in terms of these criteria have been developed and published. Goals as described in section 6.4.3 of the Technical Annex are met.

2.3.2 Individual Publications Resulting from these Achievements

Bologna

[Ben06] L. Benini, A. Marongiu, M. Kandemir, "Lightweight Barrier-Based Parallelization Support for Non-Cache-Coherent MPSoC Platforms", *CASES – International Conference on Compiler and Architectural Support for Embedded Systems,* 2006.

[Ben07] F. Poletti, A. Poggiali, D: Bertozzi, L. Benini, P. Marchal, M. Loghi, M. Poncino, "Energy-Efficient Multiprocessor Systems-on-Chip for Embedded Computing: Exploring Programming Models and Their Architectural Support", *IEEE Transactions on Computers Vol. 56, no. 5*, pp. 606-621, 2007

Dortmund

[Fal06] Heiko Falk, Jens Wagner, André Schaefer: Use of a Bit-true Data Flow Analysis for Processor-Specific Source Code Optimization, *4th IEEE Workshop on Embedded Systems for Real-Time Multimedia (ESTIMedia), Seoul/Korea*, Oct 2006

[Gr07] Nina Grau: Energy Efficient Cooperative Scheduling and Memory Allocation Techniques for Multiprocess Systems, *Dortmund University, Master thesis*, November 2006

[Jov06] Olivera Jovanovic: Dynamic voltage selection for energy efficient real-time mutliprocessors, *Dortmund University, Master thesis*, November 2006

[Mar07] Peter Marwedel: Eingebettete Systeme, *Springer*, 2007 (German edition of embedded systems textbook)

[Mar07a] Peter Marwedel: Embedded System Design, *Science Publishing Co.*, 2007 (Chinese edition of embedded systems textbook)

[Pyk07] Robert Pyka, Christoph Faßbach, Manish Verma, Heiko Falk, and Peter Marwedel: Operating system integrated energy aware scratchpad allocation strategies for multiprocess applications, *SCOPES'07*, Nice/France, Apr 2007

[Ver06a] Manish Verma, Lars Wehmeyer and Peter Marwedel: *Cache-Aware Scratchpad-Allocation Algorithms for Energy-Constrained Embedded Systems*, *IEEE Trans. on CAD of Integrated Circuits and System (TCAD), vol. 25, no. 10*, October 2006, pages 2035-2051

DTU

[Mad06] Jan Madsen, Thomas K. Stidsen, Peter Kjærulf, Shankar Mahadevan: Multi-Objective Design Space Exploration of Embedded System Platforms, *In: From Model-Driven Design to Resource Management for Distributed Embedded Systems, Edited by B. Kleinjohann, L. Kleinjohann, R.J. Machado, C. Pereira and P.S. Thiagarajan*, pages 185-194, Springer 2006.

[WuMa07] Kehuai Wu, Jan Madsen: COSMOS: A System-Level Modelling and Simulation Framework for Coprocessor-Coupled Reconfigurable Systems*, In: Proceedings of Embedded Computer Systems: Architectures, MOdeling, and Simulation (SAMOS'07)*, July 16-19 2007, Samos, Greece.

[Wu07] Kehuai Wu, Esben Rosenlund, Jan Madsen: Towards Understanding and Managing the Dynamic Behavior of Run-Time Reconfigurable Architectures*. Submitted to the International Conference on Field-Programmable Technology* 2007 (ICFPT'07).

[Pou07] K. H. Poulsen, P. Pop, V. Izosimov: A Constraint Logic Programming Framework for the Synthesis of Fault-Tolerant Schedules for Distributed Embedded Systems, *IEEE Conference on Emerging Technologies and Factory Automation*, 2007

Linköping

J. Rosen, A. Andrei, P. Eles, and Z. 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 2007.

USaar

M. Schlickling, M. Pister*: A Framework for Static Analysis of VHDL Code*, *WCET Workshop*, 2007

S. Wilhelm, B. Wachter: Towards Symbolic State Traversal for Efficient WCET Analysis of Abstract Pipeline and Cache Models*, WCET Workshop,* 2007-08-28

Jan Reineke, Daniel Grund, Christoph Berg, and Reinhard Wilhelm: Timing Predictability of Cache Replacement Policies, *Real-Time Systems*, 2007.

Christian Ferdinand, Reinhold Heckmann, Hans-Joerg Wolff, Christian Renz, Oleg Parshin, and Reinhard Wilhelm: Towards Model-Driven Development of Hard Real-Time Systems - Integrating ASCET-MD and aiT/StackAnalyzer, In: *Proceedings of Automotive Software Workshop in San Diego*, 2006

Jan Reineke, Björn Wachter, Stephan Thesing, Reinhard Wilhelm, Ilia Polian, Jochen Eisinger, and Bernd Becker: A Definition and Classification of Timing Anomalies, *In: Proceedings of 6th International Workshop on Worst-Case Execution Time (WCET) Analysis*, July 2006

Sebastian Altmeyer and Gernot Gebhard: Optimal Task Placement to Improve Cache Performance, *EMSOFT 2007, Salzburg.*

ETH Zürich

Samarjit Chakraborty, Yanhong Liu, Nikolay Stoimenov, Lothar Thiele, Ernesto Wandeler: Interface-Based Rate Analysis of Embedded Systems, *7th IEEE International Real-Time Systems Symposium, (RTSS 06)*, Rio de Janeiro, Brasil, pages 25-34, 2006.

Lothar Thiele, Ernesto Wandeler, Nikolay Stoimenov: Real-time interfaces for composing real-time systems, *International Conference On Embedded Software EMSOFT*, 06, Seoul, Korea, pages 34-43, 2006.

2.3.3 Interaction and Building Excellence between Partners

The activities of Bologna have required interaction with Aachen and Dortmund. Interaction with Aachen was required on the use of the LISATek tools both for behavioural description and for RTL generation of the VLIW processors. Students of Bologna and Aachen met at the DATE conference and discussed technical issues. Moreover email communication was very frequent.

Interaction between Bologna and Dortmund was needed to discuss compiler architecture and the two-step compilation approach (source-to-source parallelization followed by code generation) was discussed and agreed between Bologna and Dortmund based on previous experience in Dortmund using a similar approach for memor-aware compilation.

ETH Zurich has been giving a PhD course at DTU on formal methods for embedded systems, especially for resource-aware design. The course has been taken place on June 4-12 in Denmark. The title "System Level Performance Anlysis Using MPA-RTC: Models, Methods and Scenarios" already shows, that the contents of the workshop has been largely influenced by the work done within ARTIST2 and the interaction with ARTIST2 research groups, in particular

IST-004527 ARTIST2 NoE   Year 3
Cluster: Execution Platforms, Compilers & TA   D15-EP-Y3
Activity: Resource Aware Design

EPFL (Tom Henzinger) and Joseph Sifakis. Details about this activity can be found at http://www.artist-embedded.org/artist/ARTIST2-PhD-Course-on-Automated.html.

In addition, there was much interaction during the workshops organized and co-organized by ETH Zurich, see section 2.3.5.

Additional cooperations were described in the section on the results achieved.

2.3.4 Joint Publications Resulting from these Achievements

[Fal07] Heiko Falk, Sascha Plazar and Henrik Theiling. Compile-Time Decided Instruction Cache Locking Using Worst-Case Execution Paths. *In: Proceedings of "The International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS)*, Salzburg, Austria, October 2007.

[Lok07] Paul Lokuciejewski, Heiko Falk, Martin Schwarzer, Peter Marwedel and Henrik Theiling. Influence of Procedure Cloning on WCET Prediction. *In: Proceedings of The International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS)*, Salzburg, Austria, October 2007.

[Pop07] P. Pop, K. H. Poulsen, V. Izosimov, and P. Eles: Scheduling and Voltage Scaling for Energy / Reliability Trade-offs in Fault-Tolerant Time-Triggered Embedded Systems, *ACM/IEEE International Conference on Hardware-Software Codesign and System Synthesis*, 2007

[Pop2-07] P. Pop, V. Izosimov, P. Eles, and Z. Peng: Design Optimization of Time- and Cost-Constrained Fault-Tolerant Embedded Systems with Checkpointing and Replication, *submitted to IEEE Transactions on VLSI Systems*, 2007

[Ver06] Manish Verma, Lars Wehmeyer, Robert Pyka, Peter Marwedel, Luca Benini: Compilation and Simulation Tool Chain for Memory Aware Energy Optimizations, *6th SAMOS International Workshop*, 2006, p. 279-288.

[Wil07] Reinhard Wilhelm, Jakob Engblom, Andreas Ermedahl, Niklas Holsti, Stephan Thesing, David Whalley, Guillem Bernat, Christian Ferdinand, Reinhold Heckmann, Frank Mueller, Isabelle Puaut, Peter Puschner, Jan Staschulat, and Per Stenström: The Determination of Worst-Case Execution Times-Overview of the Methods and Survey of Tools, *accepted for ACM Transactions on Embedded Computing Systems (TECS)*, 2007.

2.3.5 Keynotes, Workshops, Tutorials

Luca Benini: Tutorial: NoC Middleware – OS, Platform Services, Resource Management

Design Automation and Test in Europe, Nice, France April 2007.

The tutorial covered issues related to the software environment required to efficiently support MPSoC/NoC-based platforms. Middleware services and abstractions were discussed in details.

Luca Benini: Panel: 10 or 90? The Share of the Infrastructure in Future SoCs

Workshop on Diagnostic Services in Network-on-Chips Test, Debug, and On-Line Monitoring, Nice, France April 2007

Peter Marwedel: Tutorial: Memory architecture aware compilation

Advanced Digital Systems Design, Lausanne, Sept. 2006, http://www.artist-embedded.org/artist/Overview,299.html

Peter Marwedel: Opening tutorial: Embedded Systems: Overview and research issues

1st Summer School on Ubiquitous Computing, Dortmund, Sept. 2006

Peter Marwedel: Workshop (Chairman)

Workshop on Compiler Assisted SoC Assembly (CASA), Seoul, Oct. 2006
http://ls12-www.cs.uni-dortmund.de/~marwedel/CASA_2006.html

Peter Marwedel: Tutorial: Memory architecture aware compilation

CASTNESS Workshop and School, Rome, Jan. 2007;
http://shapes.atmelroma.it/ twiki/bin/view/ShapesPublic/CastNess07

Peter Marwedel: Keynote: Performance and Predictability Improvement by Memory Architecture Aware Compilation

Infineon Workshop on Performance Modeling, Munich, Jan. 2007

Peter Marwedel: Keynote: Compiler Challenges for Embedded Design (in German)

Gesellschaft für Informatik, SIG of University Professors, April 2007, http://ira.informatik.uni-freiburg.de/gibu/jahrestreffen2007-programm.html

Peter Marwedel, Heiko Falk: Workshop (Chairmen)

10th Int. Workshop on Software and Compilers for Embedded Systems (SCOPES), Nice, April 2007, http://www.scopesconf.org/scopes-07/

Peter Marwedel: Tutorial: Memory architecture aware compilation

3rd Intern. Summer School on Advanced Computer Architecture and Compilation for Embedded Systems (ACACES), L'Aquila, July 2007, http://www.hipeac.net/acaces2007/

Lothar Thiele: Workshop: Foundations and Applications of Component-based Design

EMSOFT, October 26th 2006, Seoul

The workshop was organized by Lothar Thiele and Joseph Sifakis and brought together experts from various disciplines related to emebdded system design. One of the focus areas has been resource-awareness: Discuss recent results on component-based design with emphasis on design frameworks for real-time systems encompassing heterogeneous composition and models of computation. Especially frameworks for handling non-functional and resource constraints, design under conflicting dependability criteria, trade-offs between average performance and predictability.

http://www.esweek.org/, http://www.artist-embedded.org/artist/Overview,29.html

Lothar Thiele: Workshop: MoCC - Models of Computation and Communication

November 16-17, 2006, Zurich, Switzerland

This workshop took place at ETH Zurich. It has been recognised for long that the embedded systems domain is a multidisciplinary one which raises problems of communication and cooperation between several disciplines: software and hardware primarily but also computer science and engineering, real-time and distributed systems, telecommunication, control and signal processing etc. Each of these worlds have their own notion of such basic concepts as computation and communication which makes it difficult for designers to cooperate and achieve correct and efficient designs.

http://www.artist-embedded.org/artist/MoCC-06.html

Lothar Thiele: Dagstuhl Workshop: Quantitative Aspects of Embedded Systems

04.03.2007-09.03.2007, Dagstuhl, Germany

The workshop has been organized by B. Haverkort (Univ. of Twente, NL), J.-P. Katoen (RWTH Aachen, DE) and L. Thiele (ETH Zürich, CH). The goal of this Dagstuhl seminar was to bring together experts in the areas of embedded software design and implementation, model-based analysis of quantitative system aspects, and researchers working on extending all kinds of formal (design and analysis) methods with quantitative system aspects. These three areas are clearly well-related in the context of embedded systems, but have not been addressed as such in the past, as they have been worked upon in different communities.

Web-Page: http://kathrin.dagstuhl.de/07101/

3. Future Work and Evolution

3.1 Problem to be Tackled over the next 12 months (Sept 2007 – Aug 2008)

Aachen will work on the modelling of processors using LISATek tools and integrating them into the MPARM platform. Work will also include various backends.

Bologna will continue to work on the parallelization of code for MPSoCs. Cooperation with the affiliated partner STM will continue. Future work is aimed at increasing the level of compiler-library interaction.

Further collaboration of DTU and Linköping will focus on adaptivity-related aspects, which will allow system reconfiguration in case of failures or changes in the environment. Linköping will address quasi-static fault-tolerant scheduling for mixed soft/hard time-triggered systems, where the total utility of the application has to be maximized for soft processes, while the deadlines for hard processes have to be satisfied. DTU will address schedulability analysis for mixed hard/soft applications mapped on event-driven real-time systems. They will compare and integrate the two approaches.

At Linköping, work on "Predictability for Multiprocessor SoC Architectures" will be continued. Main goals include further optimization of the bus access and Controller design and synthesis.

Dortmund will work on memory architecture aware compilation. Emphasis will be on more dynamic approaches. A key challenge of dynamic approaches is the desire to maintain the excellent timing properties of static allocation approaches. Also, efforts will be made to supply the resulting tools for as many architectures as feasible. Cooperation with IMEC on resource aware design (now performed exclusively within the compiler and timing analysis cluster) will be extended in the European project MNEMEE.

At Lausanne and ETH Zürich work on component modelling will continue and is expected to lead to a tighter integration of the models.

Bologna, Dortmund, USaar and ETH Zürich will extend their cooperation in the European project PREDATOR, involving an additional partner from another ARTIST2 cluster.

IMEC and Dortmund's spin-off ICD will extend their cooperation in the European project MNEMEE.

3.2 Current and Future Milestones

Milestone defined for Year 3: "Use of the interacting tools, analysis of their potential and conclusions for future design methodologies; this includes, in particular, the use of the MPARM platform and integrated tools designed at the Universities of Aachen and Dortmund".

This milestone is met: Year 3 can be characterized by the massive use of tools (integrated during the previous years) on a day to day basis. A number of smaller groups have tackled specific problems described in section 1.5. Integrated tools have been used for generating results and conclusions have been drawn. Good results have been achieved in all the cases, as is evident from the list of publications. Hence, the corresponding milestone has been reached and the criteria for success are met.

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"

Only first steps into this direction can be expected to be the result of ARTIST2 funding for year 4, since ARTIST2 does not support fundamental research. However, the partners will cooperate in other projects (such as PREDATOR and MNEMEE) and these projects are methodology-oriented and will help reaching the goal defined by the milestone.

3.3 Indicators for Integration

Aachen-Bologna: use of the LISATek tools for specification of advanced processors, and integration of the generated processor model (VLIW) in MPARM virtual platform.

Bologna-Dortmund: Library support for parallelizing compiler based on the two-step compilation architecture proposed by Dortmund.

Bologna-STM: extensive use of STM power models for estimating energy efficiency of barrier syncronization.

Dortmund-IMEC: Use of integrated memory allocation tools.

DTU-Linköping: research collaboration on design and optimization of fault-tolerant embedded systems with the aim to consider efficient inclusion of fault tolerance given the tight resource constraints. Two joint publications have already been published and one is under review.

Additional indicators are included already in section 2.3.

3.4 Main Funding

Other sources of funding include:

- Bologna: STMicroelectronics, direct industrial grant, Freescale semiconductors, direct industrial grant
- Linköping: Swedish Foundation for Strategic Research (SSF)
- Dortmund: Commission of the European Union, project MORE; BMBF; Deutsche Forschungsgemeinschaft (DFG)
- ETH Zürich: Swiss National Science Foundation (Estimation of System Properties)

This list is not exhaustive.

4. Internal Reviewers for this Deliverable

Prof. Gernot Fink, Pattern Recognition Group, FB Informatik, Universität Dortmund