Project Number 288008

D 9.5 – Public Project Report

Version 1.1
9 October 2014
Final

Public Distribution

The Open Group and Technical University of Denmark

Project Partners: AbsInt Angewandte Informatik, Eindhoven University of Technology, GMVIS Skysoft, Intecs, Technical University of Denmark, The Open Group, University of York, Vienna University of Technology

Every effort has been made to ensure that all statements and information contained herein are accurate, however the Partners accept no liability for any error or omission in the same.

DOCUMENT CONTROL

Version	Status	Date
1.0	Public Report full version	28 August 2014
1.1	Update with final evaluation results	9 October 2014

TABLE OF CONTENTS

1. Introduction 1

1.1 Industrial context 1

1.2 Project Objectives 2

2. Project Results 2

2.1 T-CREST Processor 2

2.2 T-CREST Interconnect 3

2.3 Memory Hierarchy 4

2.4 Compiler and WCET Analysis 4

3. Industry evaluations 5

3.1 Industrial applications 5

3.2 Performance achievements 6

4. Project results 7

4.1 Reports 7

4.2 Technologies 9

5. Market impact 10

6. Project partners 11

EXECUTIVE SUMMARY

Safety-critical systems are important parts of our daily life. These particular types of systems have to be reliable since our lives can depend on them, and many examples exist in society today such as systems that control an airplane, systems that manage the brakes in a car, or train signalling systems that prevent collisions. Safety-critical systems need to be certified and the maximum execution time needs to be bounded and known so that response times can be assured when critical actions are needed. Just using a faster processor will not provide sufficient assurance of time predictability as today's processors are complex and understanding precisely how that complexity affects response times and predictability for critical systems has been difficult, and with newer multicore processors, nearly impossible.

The mission of T-CREST was to develop tools and build a platform and tools that provides the level of time predictability required by safety-critical systems while harnessing the power of today and tomorrows advanced multicore processing. The T-CREST time-predictable system will simplify the safety argument with respect to maximum execution time, and has achieved important technological breakthroughs having delivered double the performance in terms of predictable execution times for 4 core processors, while also achieving 4 times the performance for 16 core processors over a standard processor in the same technology (e.g. FPGA). Thus the T-CREST system will lower costs for safety-relevant applications by reducing system complexity and at the same time providing faster time-predictable execution. This report describes the project technologies, how to get access to the technologies, and some of the achievements and impact the technologies will have on future development of critical systems in Europe.

1. INTRODUCTION

1.1 INDUSTRIAL CONTEXT

Safety-critical systems are important parts of our daily life. These particular types of systems have to be reliable since our lives can depend on them, and many examples exist in society today such as systems that control an airplane, systems that manage the brakes in a car, or train signalling systems that prevent collisions. Safety-critical systems need to be certified and the maximum execution time needs to be bounded and known so that response times can be assured when critical actions are needed. Just using a faster processor will not provide sufficient assurance of time predictability when key actions or tasks must be carried out by a safety-critical system since today's processors are complex. Understanding precisely how that complexity affects response times and predictability for critical systems has been difficult, and with newer multicore processors, nearly impossible.

Standard computer architecture is driven by the following paradigm:

Make the common case fast and the uncommon case correct.

This design approach leads to architectures where the average-case execution time is optimized at the expense of the worst-case execution time (WCET). Modelling the dynamic features of current processors, memories, and interconnects for WCET analysis often results in computationally infeasible problems. The bounds calculated by the analysis are thus overly conservative, which results in system designers having to over-allocate processing cycles to a critical application to be sure timing constraints are met, or not including desired functionality in an application, which is inefficient and costly.

T-CREST brings about a sea change and takes the constructive approach by delivering a computer architecture where predictable timing is a first-order design factor. For real-time systems the project has designed an architecture using a new paradigm:

Make the worst-case fast and the whole system easy to analyze.

Within T-CREST novel solutions have been developed for time-predictable multi-core and many-core system architectures. The resulting time-predictable resources (processor, interconnect, memories, etc.) have been proven through industrial evaluations to be a good target for WCET analysis and the WCET performance has been shown to be outstanding compared to current processors. Time-predictable caching and time-predictable chip-multiprocessing (CMP) from T-CREST provide a solution for the need of increased processing power in the real-time domain.

In addition to the hardware, a compiler infrastructure has been developed in the project and WCET-aware optimization methods have been developed along with detailed timing models such that the compiler benefits from the known behaviour of the hardware. The well-known industrial WCET analysis tool aiT has also been adapted to support the developed hardware and guide the compilation and optimisations.

1.2 PROJECT OBJECTIVES

The main objectives that were established at the start of the T-CREST project were driven by the industrial needs of critical systems developers across Europe and focused on technological advances and innovations in four fundamental areas:

- Deliver a time-predictable multi-core platform so that reliable WCET analysis can be performed and tighter WCET bounds, which will enable a higher processor utilisation.
- Provide a global asynchronous local synchronous system to enable the implementation of large scale multicores. An asynchronous network on chip will connect synchronous processors.
- Design a memory network on chip and memory controller that will support time-predictable access to shared main memory to enable WCET analyses to include shared memory accesses.
- Provide a compiler and WCET analysis tool that is tightly integrated to support the T-CREST processor and provides WCET driven compiler optimisations.

The achievement of these objectives created a new T-CREST platform that will double the worst-case performance for 4 cores and be 4 times faster for 16 cores then a single processor, which will provide more processing power for future, more complex embedded real-time systems.

2. PROJECT RESULTS

T-CREST addressed technologies from the chip level (processor, memory, asynchronous network-on- chip), via compiler, single-path code generation, and WCET analysis tools, up to system level evaluations of two industry use cases. Figure 1 shows the main technology areas and their interrelationships.

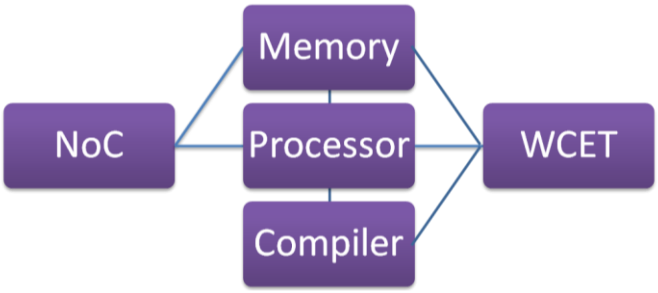

Figure 1: Core technologies from T-CREST and their interrelationships

Technology innovations in each of these areas are described in the following sections.

2.1 T-CREST PROCESSOR

The basis of a time-predictable system is a time-predictable processor. Within T-CREST a time-predictable processor, named Patmos, was developed to attack the complexity challenges of WCET analysis. Patmos is a statically scheduled, dual-issue RISC processor that is optimized for real-time systems. All instruction delays are visible at the instruction set architecture (ISA). This places more burdens on the compiler, but simplifies the WCET analysis tool. A major challenge for the WCET analysis is the memory hierarchy with multiple levels of caches, which was addressed in the project by caches that are especially designed for WCET analysis. For instructions we adopted the method cache, which operates on whole functions/methods and thus simplifies the modelling for WCET analysis. Furthermore, a split-cache architecture for data was utilised offering dedicated caches for the stack area, constants, static data, heap allocated objects, as well as a compiler and program managed scratchpad memory.

Accesses to the different types of data areas are explicitly encoded with the load and store instructions, referred to as typed load and store instructions, which direct the loads and stores to the relevant cache. This feature helps the WCET analysis to distinguish between the different data caches and makes it possible to detect earlier in the pipeline which cache will be accessed.

Patmos also supports predication of all instructions. This feature reduces the number of conditional branches and supports generation of single-path code. The industry reference and open source compiler LLVM has been extended with an optimization path that translates normal code into single-path code.

2.2 T-CREST INTERCONNECT

In order to build a chip-multiprocessor system out of Patmos processor cores an advanced interconnect – a network-on-chip (NoC) – was developed. The Patmos multi-processor platform uses non-coherent distributed memory and the address space is populated by smaller memory blocks within the different processor nodes and a large block of off-chip memory. The NoC supports time-predictable write transactions to memories in other processor nodes. The shared off-chip memory is supported by a specific memory NoC. The two NoCs are shared resources and support multiple concurrent read and write transactions.

To enable time-predictable usage of a shared resource the resource arbitration has been made time-predictable. Statically scheduled time division multiplexing (TDM) has been used as a time-predictable solution for the NoC, where the static schedule is repeated and the length of the schedule is called the period. Like tasks in real-time systems, the communication is also organized in periods. One optimization point of the T-CREST design was minimizing the period in order to minimize the latency of delivering data and the size of the schedule tables. The T-CREST NoC uses TDM from end to end, including the network interface, which results in an efficient implementation of the network interface.

In the field of embedded systems, multi-processor platforms are typically optimized for a given application or application domain. The NoC structure and/or the routing schedules are then optimized and are then application-specific. In the T-CREST project we developed a general-purpose platform – a platform that can be configured to optimize the performance of the system, or a platform which can be used as is without any configuration.

Different types of data are transferred on the NoC such as message passing data between cores, cache fills from main memory, synchronization operations such as compare-and-swap. The NoC that has been designed and developed in the T-CREST project uses asynchronous routers and a novel network interface design. In combination this avoids all buffering and flow-control and most clock-domain synchronization, and this has reduced the area by a factor of 2-3 compared to previous NoC designs.

2.3 MEMORY HIERARCHY

The only memory layer that is under direct control of the compiler is the register file. Other levels of the memory hierarchy are usually not visible – they are not part of the ISA abstraction. The placing of data in the different layers is automatically performed. While caches are managed by the hardware, virtual memory is managed by the operating system (OS). The access time for a word that is in a memory block paged out by the OS is several orders of magnitude higher than a first level cache hit. Even the difference between a first level cache access and a main memory access is in the order of two magnitudes.

Caches in general, and particularly data caches, are usually hard to statically analyze. Therefore, T-CREST developed caches that are organized to speed-up execution time and provide tight WCET bounds. A split cache architecture was utilised consisting of: (1) an instruction cache for full methods, (2) a stack cache, (3) a cache for static data, constants, and type information, and (4) a small, fully associative buffer for heap access. Furthermore, the project also addressed the integration of program- or compiler-managed scratchpad memory for data storage and inter-processor communication to tighten bounds for hard to analyze memory access patterns.

The shared memory abstraction that is prevalent in (embedded) systems introduces contention on the shared memories. Especially as off-chip SDRAM and Flash memories, which are performance bottlenecks, are heavily shared between multiple processors. Cache misses or scratchpad memories prefetches therefore have a highly variable response time due to contention between processors. Worse, even without sharing, response times are variable due to effects such as different read/write latencies, and bank open/close effects. In T-CREST we created a time-predictable external SDRAM memory controller, as this is an essential ingredient in every embedded system.

The combination of the TDM based NoC and the time-predictable memory controller allows, even on a chip-multiprocessing (CMP) system, upper bounds on memory transactions to be determined. This upper bound enables WCET analysis of individual tasks executing on a CMP system.

2.4 COMPILER AND WCET ANALYSIS

The performance of the dual-issue processor depends on statically scheduled instructions. The T-CREST approach is to expose all architectural features of a processor to the compiler in order to generate time-predictable code. Within T-CREST the industry reference open source LLVM compiler framework was adapted to target Patmos. Furthermore, compiler optimizations were developed for the WCET instead of the average case execution time.

The processor is intended as a platform to explore various time-predictable design trade-offs and their interaction with WCET analysis techniques as well as WCET-aware compilation. A co-design approach of time-predictable processor features coupled with the commercial WCET analysis tool aiT from AbsInt has been utilised. Only features where a static program analysis can be provided have been added to the processor. This includes time-predictable caching mechanisms, chip-multiprocessing (CMP), as well as novel pipeline organizations.

The aiT WCET analysis tool has also been adapted to support the very long instruction word (VLIW) Patmos processor. It is also the tool used for exploration of time-predictable processor features. The WCET oriented optimization in the compiler is tightly integrated with the WCET analysis tool, and the WCET tool provides information on the worst-case path and basic block timings to guide the optimization process.

3. INDUSTRY EVALUATIONS

3.1 INDUSTRIAL APPLICATIONS

T-CREST technologies have been evaluated using applications from two industrial partners from the Avionics and Railways domains. Project partner GMV has ported to the T-CREST platform a set of real-world applications used for avionics systems with extreme safety requirements. Partner Intecs has ported a specific railways communications processing application.

The avionics evaluations utilised three applications, one of which is intended for the highest criticality and therefore needs rigorous WCET and scheduling analysis. The demonstrator system is composed of three applications, all compliant to ARINC 653:

- Airline Operational Centre (AOC) is the on-board component of an Air Traffic and Trajectory Management System. The application was developed according to DO-178B, it has about 30 LoC and as such is a complex embedded application.
- Crew Alert System that is a typical highly critical on-board application found on civil aircraft of all major manufacturers. The GMV implementation is based on real industrial requirements and is used as a study prototype by GMV customers.
- Control Application is a typical closed-loop control application that was developed for training and demonstration purposes. It has strong real-time requirements and provides an interesting subject for further studies on thread-level parallelism.

The railways evaluations utilised an application within the European Railway Traffic Management System (ERTMS). The ERTMS has been conceived by the European railways and by the supplier industry supported by the European Commission to meet the future needs of the European rail transport network. The evaluations focused on the monitoring system of the GSM-R radio-link called GRIDES (GSM-R Integrity Detection System). The GRIDES system performs acquisition and analysis of the GSM-R radio signal within the allocated bandwidth (both in the uplink and downlink directions) in proximity of one (or more) HS/HC railway line(s). GRIDES can receive, process, and analyze both the useful GSM-R signal and the radio interference sources eventually encountered by the HS/HC train during its run, logging information about the quality of the radio link.

Intecs has ported the GRIDES implementation to the time-predictable T-CREST platform and utilised the supporting tools from the project for analysis and optimizations. The GRIDES system has specific timing constraints and requirements for predictability. It is also computationally intensive where increased parallelism will provide substantial benefits and enable additional system capabilities foreseen in the original conception of ERTMS, but not yet possible due to current limitations in platform architectures and complexities in implementing a time-predictable multi-threaded adaptation of GRIDES and other ERTMS components.

3.2 PERFORMANCE ACHIEVEMENTS

The platform used for evaluation contained 4, 9 or 16 Patmos cores, depending on the applications, communicating through the configurable NoC that interconnects local communication scratchpad memories belonging to each of the cores. Additionally, it provided a serial interface for outputting debug information and an Ethernet interface for communication. Several support tools, including the compiler tool chain and the aiT WCET tool were used to complete the evaluation platform.

All use cases were adapted to exploit the specific features provided by the T-CREST platform. To provide better coverage of these features and to address the different industrial usage models, the railways applications focused on parallelization, whereas the avionics applications focus was on achieving a higher degree of integration (i.e. concentration of applications and processor utilisation) than currently available in commercial platforms. Accordingly, the most complex railways demonstrator exploits the multi-processor characteristics of the T-CREST platform to improve the performance of a Fast Fourier Transform based application by parallelizing it, while the most complex avionics demonstrator deploys three independent and unrelated applications over the platform, validating that their temporal behaviour is unaffected.

For the railways use case the platform adaptation process encompassed the parallelization of the application, removal of floating point operations and addition of support for message passing using the network-on-chip. The avionics applications adaptations included the port of a real-time operating system to the platform (RTEMS), the development of a second stage bootloader to support asymmetric multiprocessing mappings and the modification of ARINC 653 message passing APIs to make use of the network-on-chip communication.

Once the applications were ported, the first validation of the platform was to verify if they could comply with their original requirements and execute accordingly with the expected behaviour. This was the case for all the use cases from railways and avionics. Next, the behaviour of the platform with regard to the worst case execution time was assessed. The worst-case execution time for selected tasks in all the applications was estimated using the aiT tool. It was possible to obtain estimations for every application, in any given configuration with regard to the number of cores.

For the railways applications, it was possible to obtain an improvement in the worst case execution time, as estimated by the aiT analysis tool, with the parallelization of the application. When compared to the single-core version, it was possible to achieve an improvement of 178% for the tri-core case and 567% for the fifteen-core case in terms of worst case execution time. This provided verification that the T-CREST technology innovations achieved the targeted improvements of:

> *"A speedup of WCET bounds for parallelizable applications roughly logarithmic to the number of processors: double performance for 4 cores, three times for 8 cores, 4 times for 16 cores".*

For the avionics applications, the objective was to validate that different applications are temporally independent despite sharing an underlying multi-core platform. It was possible to verify that the worst case execution time of a given application deployed on the T-CREST platform depends solely on the number of cores used and, therefore, only on the configuration of the memory tree. To further validate this hypothesis, the execution time of selected applications was measured using the cycle accurate timer provided by the processor in different multi-core configurations. This test showed that the execution time only differs substantially when the number of cores is varied; executions maintaining the number of cores but varying the applications executing concurrently yield a constant temporal behaviour.

In addition, WCET bounds from the Patmos processor were compared to WCET bounds from a LEON processor for the same frequency and memory latency values. This was used to validate the T-CREST technology innovations achieved key improvements of:

> *"The T-CREST Patmos processor will perform, with respect to the WCET bound, at least 4 times faster than a standard processor in the same technology (i.e. FPGA)".*

The industrial evaluations and analyses have confirmed the T-CREST platform developed in the project fulfils the technical objectives and requirements for industry and have even over-achieved in some areas with regard to desired performance improvements.

4. PROJECT RESULTS

Nearly all of the project reports describing the technology advances developed in the project are publicly available. In addition, the majority of the T-CREST technologies are distributed in open source format and available online. All of the public project results can be accessed via the project website: www.t-crest.org. Details concerning the available public reports and open source technologies are described in the following sections.

4.1 REPORTS

The following public reports that were submitted to the European Commission as formal project deliverables for the T-CREST project are available for download:

Processor related reports

- D2.1 Software simulator of Patmos
- D2.2 Concepts for interrupt virtualization
- D2.3 Hardware implementation of Patmos
- D2.4 Design and implementation of interrupt virtualization hardware and APIs
- D2.5 Report on architecture evaluation and WCET Analysis
- D2.6 Integration and report
- D2.7 Evaluation report on interrupt virtualization

Network on Chip (NoC) related reports

- D3.1 Survey of time-predictable and asynchronous NoCs, and their WCET analysis
- D3.2 Simulation model of the self-timed NoC
- D3.3 Hardware implementation of the self-timed NoC
- D3.4 Report documenting hardware implementation of the NoC
- D3.5 Report on impact of asynchronicity on predictability of the NoC
- D3.6 FPGA implementation of self-timed NoC
- D3.7 Analysis report on FPGA implementation of self-timed NoC
- D3.8 Integration report of the full system implemented in an FPGA

Memory hierarchy related reports

- D4.1 Reconfigurable memory controller concepts
- D4.2 Reconfigurable memory controller design and implementation
- D4.3 Dynamic memory controller concepts
- D4.4 Dynamic memory controller design and implementation
- D4.5 Integration of combined memory controller
- D4.6 Final benchmarking and report of memory controller

Compiler related reports

- D5.1 Report on ISA and architectural support for generating time-predictable code
- D5.2 Initial compiler ersion
- D5.3 Report on compilation for time-predictability
- D5.4 Report on WCET-oriented optimizations
- D5.5 Report on coding policies for time-predictability
- D5.6 Full compiler version
- D5.7 Report on compiler evaluation

Dissemination related reports

- D8.2 Project white paper
- D8.4 Workshop report
- D8.5 Open source reference implementation

Copyright in all of the above documents remains vested with the T-CREST project partners.

4.2 TECHNOLOGIES

The following technologies are available for download from the project source forge located at www.github.com/t-crest:

Core components

- **Patmos processor** - the Patmos repository contains the hardware description of the processor, a software simulator, and a small assembler. Furthermore, the repository contains two memory controllers for SRAM and SSRAM memories as they are available on the Altera DE2 boards.
- **Compiler, Linker, and Library** - the T-CREST compiler, linker, and libraries are distributed over several repositories.
  - Port of the LLVM compiler infrastructure to the time-predictable processor Patmos
  - Port of the clang front-end to the time-predictable processor Patmos
  - Port of the LLVM compiler library
  - Port of the BINUTILS gold linker to Patmos
  - Newlib port for Patmos
- **RTEMS** – Port of the RTEMS real-time operating system to the time-predictable processor Patmos
- **Argo NoC** – Argo is a time-division multiplexing network-on-chip with synchronous and asynchronous routers and an area-efficient network interface.
- **Memory Tree** – the memory tree is the memory interconnect designed for the T-CREST project such that multiple Patmos processors have a time-predictable way to access memory.
- **S4NOC** – a statically-scheduled TDM network-on-chip for real-time systems. This is the initial version of the TCREST NoC including a minimalistic network interface.
- **Benchmarks** – a collection of embedded and WCET benchmarks and tests for the Patmos processor and compiler.
- **Aegean** – a configuration framework for the T-CREST platform.
- **SDRAM controller model** – RTMemController is a tool for evaluating the worst-case and average-case execution time of memory transactions of the dynamic memory controller.

Secondary components

- **SDRAM controller** – a simple and static SDRAM controller for time-predictable main memory access.
- **Ospat** – a real-time operating system for Patmos.
- **Poseidon** – a static NoC TDM scheduler.
- **Debugger** – a port of the LLDB debugger for Patmos.
- **OTAWA** files – the adaptation of OTAWA analysis tool for WCET for Patmos.
- **RTEMP GUI** – a graphical configuration tool for the RTEMP project.

Most of the open source technologies from the T-CREST are made available under the industry friendly BSD license. Many of the secondary components are additional contributions made to the T-CREST technologies, which are often the results of a Master's thesis.

5. MARKET IMPACT

Embedded systems are taking over control in ever more demanding environments, including safety- and security-critical systems. The robustness and safety of systems is therefore an ever-growing competitiveness factor. The ability to produce robust predictable systems at a competitive price will be essential for keeping European companies at the cutting edge of the embedded system market.

The potential impact for European industry in being able to develop critical applications on a new generation of processors made possible by the T-CREST technologies is enormous. For a typical avionics system hardware costs are largely determined by the number of processor cards utilised. At similar volumes, a processor board supplier would be able to produce a T-CREST based processor at nearly the same manufacturing costs as that of a current COTS processor board. A comparison of the hardware costs of a typical Integrated Modular Avionics (IMA) based system running the same applications using COTS based processors versus new T-CREST based processors, which provide lower WCET bounds and greater application integration, is shown in Table 1.

	COTS IMA System	T-CREST IMA System
Hardware backplane	€16,000	€16,000
Networking backplane	€30,000	€30,000
Single core SBCs	16 x LEON-3 SBCs	5 x T-CREST SBCs
SBC Costs	16 x €10,000 = €160,000	6 x €10,000 = €60,000
Network Interface Card	16 x €3,000 = €48,000	6 x €3,000 = €18,000
Total Hardware Costs	€254,000	€124,000
Savings from T-CREST technologies		**51%**

Table 1: Comparison of IMA hardware exploiting T-CREST improved WCET bounds

Commercial installations of IMA avionics systems address reliability and safety by using redundancy for the processors within the IMA system so the cost savings would be further amplified through a commensurate reduction in redundant processor boards needed when using T-CREST technologies.

The economic advantages of utilising T-CREST technologies for a single IMA deployment are clearly compelling. Considering the number of IMA systems to be deployed in the coming years across European airplane production lines, the potential impact from the T-CREST project innovations reaches many millions of Euros. T-CREST technologies will help European industry to build reliable systems, not only in the areas of air and ground transportation, but also in many other areas where robustness, availability, and safety are important requirements for embedded systems.

6. PROJECT PARTNERS

The T-CREST project consortium consisted of the following industrial, academic, technology, and standardisation organizations:

AbsInt Angewandte Informatik

Eindhoven University of Technology

GMVIS Skysoft

Intecs

Technical University of Denmark

The Open Group

University of York

Vienna University of Technology

The T-CREST partners are always pleased to collaborate with other organisations and projects involved in related computing systems technologies. Contact details can be found by visiting the project website www.t-crest.org.