Mixed-Criticality Architecture for Networked Multi-Core Chips

- Cross-domain architectural style and models for MCS
- Modular certification and mixed-criticality product lines
- Platform with virtualization at chip and network level
- Adaptation strategies for mixed-criticality systems
- Development methodology, variability management and tools
- Mapping to different technology targets (e.g., Xilinx ZYNQ, PPC, Intel)
- Demonstration in avionic, healthcare and wind power use cases
**Project Timing**

**PHASE 1:** Requirements Analysis

**PHASE 2:** Definition of architectural style, development process and high-level design of virtualization methods

**PHASE 3:** Development of 1st version of models, virtualization and development methods, demonstrator designs

**PHASE 4:** Development of final models, virtualization techniques, development methods, demonstrator designs

**PHASE 5:** Demonstration, validation, training

---

**Phase 5 focuses on**

- Demonstrator support (WP1-5)
- Demonstration (WP6-8)
- Community building and standardization (WP9)
- Dissemination (e.g., book manuscript), exploitation and training (WP10)
Project Results (1)

WP1 Architecture

- WP2 Multicore Virtualization Technology
  - Firmware monitoring for virtualization of processor
  - Memory virtualization and interleaving
  - Execution environment based on XtratuM and KVM
  - On-chip network virtualization
  - Local resource management

WP3 Mixed-Criticality Network

- Timely Redirection of Messages
- Protocol Conversion
- Monitoring and Configuration

- Monitoring and Configuration
- Packet classification
- Serialization service
- Ingress and egress queuing
- MAC interfacing

WP4 Tooling, Scheduling and Analysis

- Offline adaptation strategies for MCS
  - Generation of platform configuration files
  - Tool integration
  - Explicit variability configuration

Major Results

<table>
<thead>
<tr>
<th>Architectural style</th>
<th>Application and platform Models</th>
<th>Memory virtualization and interleaving</th>
<th>Execution environment based on XtratuM and KVM</th>
<th>On-chip network virtualization</th>
<th>Local resource management</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Project Results (2)

<table>
<thead>
<tr>
<th>WP5 Mixed-Criticality Certification</th>
<th>Modular safety case</th>
<th>Simulation Framework</th>
<th>Cross Domain Mixed Criticality Patterns</th>
<th>Guidelines for Process / Tool Integration</th>
<th>Certification of Product Families</th>
</tr>
</thead>
<tbody>
<tr>
<td>WP6-8 Demonstrators</td>
<td>Avionic (FMS)</td>
<td>Windpower (e.g., safety system)</td>
<td>Healthcare (e.g., patient monitoring, entertainment)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>WP9 Community Building</td>
<td>Community Building</td>
<td>Road-mapping</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WP10 Dissemination, Exploitation and Training</td>
<td>Dissemination</td>
<td>Exploitation</td>
<td>Training</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
# Physical DREAMS Platforms

<table>
<thead>
<tr>
<th>Technology from WP2</th>
<th>Technology from WP3</th>
<th>Technology from WP7</th>
<th>HW platform</th>
<th>Leading WP</th>
</tr>
</thead>
<tbody>
<tr>
<td>XtratuM</td>
<td>Security Services</td>
<td>TTE</td>
<td>MemGuard</td>
<td>Linux/KVM</td>
</tr>
<tr>
<td>DRAL</td>
<td>TTE</td>
<td>XtratuM</td>
<td>Secure Monitor</td>
<td>Juno</td>
</tr>
<tr>
<td>STNoC</td>
<td>PCIe Integration</td>
<td>T4240</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ZYNQ (DHP)</td>
<td>Galileo</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Legend:
- Technology from WP2
- Technology from WP3
- Technology from WP7
- HW platform
- Leading WP

12.01.2017
Technology Pitch: Execution environments

Presenter: Javier Coronel
Execution environments - XtratuM

Execution environments based on

- DRAL
- Windows
- Linux
- LithOS
- RTEMS

XtratuM

- Bare Metal hypervisor for MCS
- Spatial and temporal isolation
- Fault contention and management
- Static resources allocation
- Robust IPC
- Partition/system management
- Interrupt, time, memory and CPU
- Hierarchical scheduling
- High-performance/scalability
Execution environments - KVM

- Turns the Linux kernel into a full-blown hypervisor
- Leverages infrastructure of Linux for scheduling and memory management
- Multiple architectures supported, including ARMv7/v8
- KVM features:
  - Support for hardware virtualization
  - High performance and scalability
  - Open Source community
  - Used in conjunction with various userspace tools
- Provides virtualization support for CPU, memory, interrupts, timers
KVM architecture overview

- Machine model or device emulation is not provided by KVM but by userspace application
- KVM exposes a /dev/kvm interface for userspace tools
- Popular tools to use with KVM: QEMU, kvmtool
Resource Management

Gerhard Fohler (TUKL)
(input from ONERA, TRT, USIEGEN)
before DREAMS: from EU projects

global – local coordination

BETSY

fault handling, reconfiguration

XtratuM

resource mgt - multicore
Goals of resource management

- reconfiguration of a mixed-criticality system
  - upon foreseen and unforeseen changes in its operational and environmental conditions
- adaptability mechanisms for securely reconfiguring the system
  - without interrupting or interfering with execution
- secure, adaptive fault tolerance
coordination via separation of decisions

- local
- global (system wide)

offline computed configurations
- Hypervisor XtratuM
- DREAMS services on top of XtratuM
- Applications:
  - critical applications: Flight Management System (FMS), Display Management System (DMS), and Sensors Data Provider (SDP)
  - Best-effort applications are: In-Flight Entertainment (IFE) and panels (PAN)
DREAMS: adaptive, faults, reconfiguration

- GRM stores the global reconfiguration graph (LRMs must have complete symmetric local reconfiguration graph).
DREAMS: secure RM
with DREAMS: from EU projects

- fault handling
- secure res mgt
- virtualization
- offline configurations
- on/off chip networks

BETSY

global – local coordination

fault handling, reconfiguration

resource mgt - multicore

DREAMS

fault handling
secure res mgt
virtualization
offline configurations
on/off chip networks

STACCATO
Technology Pitch:
Modular Safety Cases (MSCs)

November 23rd, 2016
Imanol Martinez
IK4-IKERLAN
Certification & Modularity

**Certification**: “Procedure in which an accredited or authorized person or body assesses and verifies the requirements of a system in accordance with established requirements or standards”

**Safety Certification**: “assess the compliance of a system to the requirements of a safety-related standard (E.g., IEC 61508)”

*Traditional approach to certification relies on the certification of the whole system, where if a safety aspect of the system changes, the re-certification of the entire system is required.*

**Modularity** - “is a complexity management technique that subdivides the system into smaller parts (modules) that are independently generated and re-used to compose a system”. [Kopetz08]
**Safety Case** – “A documented body of evidence that provides a convincing and valid argument that a system is adequately safe for a given application in a specific environment (such as automotive, railway, lift and etc.).” [Bishop98]

**Modular Safety Cases (MSCs)** – “Safety cases that enable the reusability of predefined modules, reducing the overall complexity (simplification strategy) and limiting the impacts of changes to specific modules or areas.”
Modular Safety Cases for Mixed-Criticality Systems

D5.1.1 A Modular Safety Case for an IEC 61508 compliant Generic Hypervisor

D5.1.1 A Modular Safety Case for an IEC 61508 compliant Generic Partition

D5.1.2 A Modular Safety Case for an IEC 61508 compliant Generic COTS Multi-Core Device

D5.1.3 A Modular Safety Case for an IEC 61508 compliant Generic Mixed-Criticality Network
Modular Safety Cases and Linking Analysis

D5.1.1
A Modular Safety Case for an IEC 61508 compliant Generic Hypervisor

D5.1.1
A Modular Safety Case for an IEC 61508 compliant Generic Partition

D5.1.2
A Modular Safety Case for an IEC 61508 compliant Generic COTS Multi-Core Device

D5.1.3
A Modular Safety Case for an IEC 61508 compliant Generic Mixed-Criticality Network

Xtratum

EtherCAT®

Universität Siegen
Technology Pitch
Model-Based Development and Toolchain

November 22nd, 2016
Simon Barner
fortiss GmbH
DREAMS Model-Based Development and Toolchain

- Development Methodology and Integrated Toolchain
  - Variability Exploration
  - Design-Space Exploration
  - Resource Allocation and Scheduling
  - Reconfiguration
  - Safety Checker and Argumentation Synthesis
  - Platform Configuration Generation

- Modeling Mixed-criticality Systems
  - Applications (Architecture, Timing Requirements)
  - Safety Requirements and Properties
  - Hierarchical Platforms (DREAMS: Cluster, Node, Tile, Processor & Hypervisor Level)
  - Deployments and Resource Allocations
1. Basic Scheduling Configuration
   - Mapping of Application to Computation and Communication Resources
   - Offline Scheduling
   - Configuration Generation

2. Scheduling Configuration with Resource Management
   - Extends Use Case 1
   - Global and Local Reconfiguration Strategies
   - Compensates Core Failures and Deadline Overruns

3. Variability and Design-Space Exploration
   - Extends Use Case 1 to MCS Product-Lines
   - Business variability: Which Features and Requirements?
   - Technical variability: How are Features implemented?
Modeling Mixed Criticality Systems
Variability & Product-lines

November 23rd, 2016
Franck CHAUVEL
SINTEF ICT
Reuse Beyond one System

- Each customer is different

- Why Product-lines?
  - Lower Costs
  - Higher quality
  - by Reusing across products

- Certification?
Variability Management in DREAMS

TECHNICAL VARIABILITY

- Variability Model
- Reusable Artefacts
- Product Sampling
- Resolution
- Realisation
- Annotated System models

BUSINESS VARIABILITY

- Safety Case Generation
- Solution
- Evolutionary Optimisation
  - Energy Verification
  - Timing Verification
  - Safety Verification

Solution
Memory/Network Bandwidth Regulation & Virtualization

ST Microelectronics, TEI & VOSYS
Porting Genuine MemGuard/Linux

- MemGuard regulates memory bw per core
  - working implementation as kernel module on ARM v7 (Zedboard) & v8 (Juno)
  - ARM v8 (Dragonboard 410c) issues, e.g. kernel readjusts perf-event update rates
- Genuine MemGuard already used with KVM hypervisor
  - control memory bw of VMs mapped to different cores
Extended MemGuard

- Extended MemGuard supports Violation-Free mode, improved BW reservation and reclaiming, EWMA
  - HW prototype (Zedboard FPGA)
    - backpressure to avoid deadlock
    - improvements vs Genuine MemGuard HW
  - Linux implementation on Intel CPU & ARM v7
    - optimization & exploration
Netguard

- Linux NetGuard Extension on Intel CPU & ARM v7
  - network bw regulation for video streams (WP8)
  - ARM v8 implementation & further extensions
- Linux scheduler policy for regulation per process group
- Examine interactions with STNoC QoS policies
Video Streaming Demo

root@linaro-ubuntu-desktop:~/netguard_driver# echo "1500 50 300 1200" > /sys/kernel/debug/netguard/netguard_config
root@linaro-ubuntu-desktop:~/netguard_driver# echo "1500 50 1200 300" > /sys/kernel/debug/netguard/netguard_config
root@linaro-ubuntu-desktop:~/netguard_driver#
DREAMS Architectural Style
TTEthernet Technology Maturation
Scheduling in the Real-Time IoT
Mixed Criticality in Healthcare

Marcello Coppola
Thirty years ago, health care technologists realized a simple truth: monitoring patients improves outcomes. However, hospital error is still a leading cause of death:

- the Institute of Medicine named it the third leading cause of death after heart disease and cancer.
- Thousands and thousands of errors occur in hospitals every day.
- Many of these errors are caused by false alarms, slow responses, and inaccurate treatment delivery.
Main Objective: new technology spreading through patient care

- By networking devices, alarms can become smart,
  - Only sounding when multiple devices indicate errant physiological parameters.
- By connecting measurements to treatment, smart drug delivery systems can react to patient conditions much faster and more reliably than busy hospital staff.
- By tracking patients around the hospital and connecting them to the hospital server, efficiency of care can be dramatically improved.
The BG is a wearable, battery-operated device intended for use as a part of a multi-parameter analysis system: it acquires, digitalizes, stores and periodically transmits via a Bluetooth.

It is based on our bestselling STM32 product.

**Key features**
- Heart-rate detection
- Physical-activity estimation
- Breathing-rate measurement
- Body position

**Applications**
- Chronic cardiac-disease monitoring
- Home monitoring for the elderly
- Event monitoring
System Architecture Overview
From PoC to ST Products

- Body Gateway Product
  - Started to be used in Hospitals, and Medical Services companies

- NoC supporting mixed criticalities used in real products
  - Space, Multimedia Applications, Automotive

- Applications (eg Video Streaming) on STM32
Conclusions

DREAMS enables ST to add extra value to our products increasing our sales and growing our customer base in different market segments

Thank you!
EC DREAMS Success Story: VOSYSmonitor, a low latency monitor firmware for mixed-criticality systems

Kevin Chappuis
2016-11-22
Introduction

- An important trend in the design of embedded systems is the integration of applications with different levels of criticality.

- Such concept brings new challenges to the industry:
  - Multi-OS support and integration
  - Efficient shared use of SoC resources (e.g., peripheral, memory, etc)
  - Separation of functions and ensure the isolation of safety critical systems
Software systems consolidation

- Last multi-cores architectures (e.g., ARMv8-A) are bringing new features to hardware platforms.
  - Computing performance is increasing
  - Power consumption is decreasing
  - New hardware extensions (e.g., security, virtualization, etc)

- The goal is to use the computing performance and hardware capacities to embed more functionalities with different levels of criticality in the same platform in to decrease the number of hardware resources needed.
VOSYSmonitor description

- VOSYSmonitor, developed by Virtual Open Systems, enables the co-execution of virtualized systems along with a safety critical application on the same ARMv8-A platform.
- Safety critical OS isolation using ARM TrustZone
- GPOS virtualization extensions (KVM) enabled
- Ability to safely exchange data between RTOS/GPOS
- Certifiable firmware
- High priority to the critical applications to meet timing constraints.
- Power management coordination
VOSYSmonitor specification

- VOSYSmonitor design is based on the following requirements in order to integrate mixed-criticality systems without compromising safety applications.
  - VOSYSmonitor setup impact less than 1% on the total Safety critical OS boot process.
  - Minimize the interrupt latency impact – GPOS / Safety critical OS context switching time must be lower than 1us.
  - Support complete safety critical OS resources (e.g., Memory, Peripherals, etc) isolation from GPOS illegal access.
  - Standard compliances (e.g., PSCI, SMCCC).

- VOSYSmonitor supports several ARMv8 platforms:
  - ARM Fast Models AEMv8A (virtual platform)
  - ARM Juno Development Board
  - Renesas R-Car H3 (ISO 26262 – ASIL B compliant)
  - Nvidia Jetson TX1
Application fields

- **Drones**: Run a safety software (e.g., landing, overflight, etc) in case of the main control OS fails.

- **Automotive**: Consolidation of the Infotainment and cluster dashboard systems on a common hardware.

- **Healthcare**: Monitor critical signals (e.g., ECG) and infotainment video streaming

- **Mobiles**: Execute Android OS and secure applications (e.g., online payment, DRM, fingerprint, etc)

All possible use-cases with mixed-criticality systems.
Alstom Wind Exploitation story

Barcelona, November 22nd, 2016
Ton Trapman
Planned Product Overview

- Input / Output Control Unit (Galileo Platform)
- Protection Unit (Dreams Harmonized Platform)
- Safe Input / Output
- PCIe
Value proposition

- Compliance with customer demands for:
  - Increase in product safety for Health, Safety and Environment (HSE)
  - Higher flexibility towards customisations
  - Compliant with product certification requests demonstrating product reliability and resilience

- Marketing:
  - Competitive value proposition for offering SIL 3 safety in wind turbines

- Product evolution in:
  - Following the tendency of using more standard hardware and industrial controllers
  - Compliance with stricter industry related standards (IEC 61400)
  - Product updates for lifetime extension
  - Cost reduction by using hypervisor technologies
THALES Exploitation Story

Before DREAMS

Technologies
- Multi-processor
- Single-core
- Federated systems
- Redundancy
- Partitioning (IMA)

Products
- Space
- Defence
- Aerospace
- Transportation

Future with DREAMS

Framework (models, tools)
- Multi-core
- Distributed systems
- MC Fault Mgmt, Redundancy
- Certification
- Bounded Partitioning
- Resource Management

Next-gen Products
- Transportation
- Space
- Aerospace
- Defence

DREAMS towards
Before DREAMS

XtratuM

- x86 and LEON processors
- Para-virtualization
- Basic Comm. services

Space
Industrial

After DREAMS

XtratuM

- Multi-processor
- X86, LEON, ARM, PPC processors
- Para-virtualization & Full virtualization
- Certification Templates (Safety)
- Multi-processor Multi-node
- Comm. and Network services

Space (LEON + ARM)
Avionics (PPC)
Automotive (ARM)
Industrial (ARM, x86)
TÜV Rheinland is a service provider for:

- Testing
  - Inspection
  - Certification

project helps:

- to increase the competencies regarding the use of multi-core processor systems in mixed criticality applications
- to stay in business and beyond to be able to assess the increasingly complex safety applications using increasingly powerful and complex processing engines
- to evaluate components and systems, which are not sufficiently taken into account in today’s safety standards
Exploitation Story: RTaW

Before DREAMS

Scheduling Configuration
Timing Analysis Algorithms

- Off-chip communication
- Event driven scheduling

Design & Verification Tool
Engineering, R&D

- Avionics
- Automotive
- Industry

After DREAMS

- Off-chip & on-chip communication
- Event & Time triggered scheduling
- Task scheduling
- Timing Decomposition

- Avionics
- Industry
- Space
- Automotive
Exploitation Story: RTaW

- RTaW offering, before DREAMS:
  - Software Tools
    - Worst-Case Timing Analysis and Simulation of event-driven networks: Switched Ethernet, CAN, DITS
    - Corresponding optimization of scheduling parameters
  - Competences (consultancy, R&D)
    - Event-driven communication protocols:
      - Optimal configuration
      - Verification through analysis and simulation
  - Application domains
    - Automotive, Aerospace
Exploitation Story: RTaW

- RTaW offering, after DREAMS:
  - Gained competences (consultancy, R&D, tooling):
    - Mixed criticality systems
    - Layered time triggered and event triggered scheduling
    - Hierarchical task/partition scheduling with mode changes
    - NoC technologies and scheduling
    - Model driven configuration file generation
  - Extensions of tools:
    - Timing decomposition
    - Scheduling Configuration: NoC
    - Timing analysis: STNoC, TTEthernet, cyclic partition/task scheduling
Exploitation Story: RTaW

- New opportunities
  - Satellites launch vehicles
  - Multi/many-core in automotive
  - High performance embedded computing: many core with NoC
  - All industries using time-triggered control systems
Dissemination

Roman Obermaisser, Univ. of Siegen
Barcelona, Nov. 2016
DREAMS Dissemination Policy and Goals

 Achieve the widest possible awareness for scientific & technical concepts and integrated technologies in DREAMS

1. Broadcast research results to stakeholder community
2. National and international exhibitions and fairs
3. Focused training
4. Academic and industrial clusters and networks
5. Public Awareness
DREAMS Timeline and Dissemination in Year 3

**PHASE 1:** Requirements Analysis

**PHASE 2:** Definition of architectural style, development process and high-level design of virtualization methods

**PHASE 3:** Development of 1st version of models, virtualization and development methods, demonstrator designs

**PHASE 4:** Development of final models, virtualization techniques, development methods, demonstrator designs

**PHASE 5:** Demonstration, validation, training

- M1 ... M6
- M7 .. M10
- M11..M18
- M20..M34
- M36..M48

**Networking Session at ICT 2015**
Innovate, Connect, Transform, October 2015, Lisbon

**HIPEAC Conference January 2016,**
Prague, Czech Republic

**IEEE Int. Symposium on Industrial Embedded Systems, Session on Mixed-Criticality Systems, June 2016**

**Advanced Computing and CPS Workshop, Brussels, June 2016**
Target Groups

- Application Domains
  - Avionics
  - Industrial stakeholders
  - Healthcare
  - Further domains (e.g., railway, automotive)
- Stakeholders along Supply Chains
  - System integrators
  - Tool developers
  - Hardware platform developers
  - Software developers
  - Consulting
- Public
Overview of Dissemination: DREAMS Year 3

- Scientific: 79%
- Industrial: 15%
- Public: 6%

Total Number of Publications in Year 3: 44

- Supply Chains – Semiconductor Industry: 22%
- Supply Chains – Software developers: 37%
- Supply Chains – Consulting: 7%
- Avionic: 8%
- Windpower: 4%
- Healthcare: 7%
- Further Domains: 15%
Overview of Dissemination: DREAMS Year 3

- Architecture: 16%
- Real-time and scheduling: 14%
- Dependability: 7%
- Security: 2%
- Model-based development: 5%
- Multicore chips: 23%
- Networks: 12%
- Simulation: 14%
- OS and hypervisor: 2%
- Certification and safety: 5%

European Project “DREAMS” Integrates Novel Technological Building Blocks towards a Reference Platform for Mixed-Criticality Systems

The FP7-ICT project DREAMS (Distributed REal-time Architecture for Mixed criticality Systems) has announced its progress to develop a methodology, architecture and a reference platform for complex, mixed-criticality systems across a broad range of application areas, including healthcare, avionics, and energy systems.

DREAMS was launched in October 2013 to realize the tremendous economic benefits of reduced maintenance and installation efforts, hardware cost, weight, size and energy consumption that could be generated from the reduction of discrete devices and cables of mixed-criticality systems.
Social Media (Linkedin, Twitter)

DREAMS Architectural Style Introduced and other news in the first DREAMS newsletter
The first DREAMS newsletter is out! Read it at the DREAMS Website (under "Publications")
- Architectural Style introduced
- Intermediate Integration Nearing Completion
and more...

DREAMS Newsletter #1 - May 2015
The FPT-ICT integrated project DREAMS (Distributed REal-Time Architecture for Mixed Criticality Systems) with project start in October 2013, provides first results on the cross...

European Project “DREAMS” Integrates Novel Technological Building Blocks towards a Reference Platform for Mixed-Criticality Systems