BOOKLET OF PRESENTATIONS

http://www.ada-europe.org/conference2024

In cooperation with

About this Booklet

This booklet contains short summaries of all the presentations included in the conference core program. The booklet groups presentations by session, prefixing their title with their type: '**RP**' for research presentations, '**IP**' for industrial presentations, '**WiP**' for work-in-progress presentations.

ELSEVIER

The proceedings of the research papers will appear in a dedicated Special Issue of Elsevier's Journal on Systems Architecture.

The proceedings of the industrial papers will appear in forthcoming issues of Ada-Europe's Ada User Journal, along with papers drawn from the Work-in-Progress presentations.

We invite you to use this booklet as "navigational tool" throughout the conference program. Enjoy the conference!

Core Conference Program

	Morning		Afternoon	
	Before Break	**After Break**	**Before Break**	**After Break**
Wednesday, June 12th	**Keynote Talk**	**Session 1:** *Fault Tolerance and Reliability in Heterogeneous Systems*	**Session 2:** *Software Verification and Code Generation*	**Session 3:** *Security and Safety in Embedded Systems*
Thursday, June 13th	**Panel Discussion**	**Session 4:** *Machine Learning and Optimization for Embedded Systems*	**Session 5:** *Real-Time Systems and Their Analysis*	**Session 6:** *Advancements in RTOS and Embedded Software Development*

Conference Sponsors

HUAWEI
EUROCITY
Collins Aerospace An RTX Business
AdaCore
acm-w
Rising STARS
BCN-W acm Chapter
OpenMP

EXCELENCIA SEVERO OCHOA
MINISTERIO DE CIENCIA, INNOVACIÓN Y UNIVERSIDADES · AGENCIA ESTATAL DE INVESTIGACIÓN
Financiado por la Unión Europea NextGenerationEU
GOBIERNO DE ESPAÑA · MINISTERIO PARA LA TRANSFORMACIÓN DIGITAL Y DE LA FUNCIÓN PÚBLICA
Plan de Recuperación, Transformación y Resiliencia

TABLE OF CONTENTS

SESSION 1: FAULT TOLERANCE AND RELIABILITY IN HETEROGENEOUS SYSTEMS

RP: HyFAR: a cost-effective Hypervisor-based Fault tolerance Approach for Heterogeneous Automotive Real-time Systems

Johannes Lex (speaker), Ralph Mader, Margull Ulrich, Dietmar Fey

Abstract

Fault tolerance is a key aspect for fully autonomous vehicles, as there is no human driver available to take control of the vehicle as a backup. Such autonomous vehicles incorporate signal-oriented and service-oriented hardware and software architectures within one heterogeneous real-time system. Fault tolerance is commonly achieved by adding redundant Electronic Control Units (ECUs) to the system. However, redundant ECUs increase the weight, cost and power consumption of the system. This paper presents a novel hypervisor-based fault tolerance approach for automotive real-time systems (HyFAR), which is based on the largely unexplored concept of migrating software in a highly heterogeneous real-time system using virtualization technology. It is shown, that the fault tolerance of an automotive vehicle can be enhanced in a cost-effective way without the need of additional hardware. The process of recovering critical service-oriented software using a signal-oriented hardware and vice versa is examined. This paper gives a detailed overview of the effects of emulation, virtualization, separation and the type of the hypervisor towards the recovery time and the freedom from interference of signal-oriented and service-oriented software. The results demonstrate that recovering critical service-oriented software using signal-oriented hardware is limited due to missing middleware and virtualization support and resource scarcity. However, recovering critical signal-oriented software using a service-oriented hardware is feasible, while a subset of the original service-oriented software can be continued on the same hardware. The resulting approach can be applied to a range of applications including thermal management or lane departure warning.

RP: Enhancing Scalability of Static Code Analysis through Graph Database and Pattern Matching

Quentin Dauprat (speaker), Paul Dorbec, Gaetan Richard, Jean-Pierre Rosen

Abstract

This study aims to enhance the scalability of static code analysis tools by combining graph databases and pattern matching queries. Traditional tools often struggle with large codebases, leading to increasingly long analysis times. A significant portion of this time is attributed to the necessity of accessing all information pertaining to an object, which may be dispersed across numerous files, and so, into multiples Abstract Syntax Trees (ASTs). The proposed method stores and queries code representations in a graph database, improving efficiency and scalability. The study focuses on Ada, a language with complex code structures, and benchmarks the method against established tools such as AdaControl and GNATcheck. The results show that the proposed method outperforms conventional tools, demonstrating its potential to enhance static code analysis. First results obtained show a clear improvement in terms of performance, being 326 times faster than AdaControl and outshining GNATcheck, ranging from 57 to 148 times faster respectively in mono-thread and multithreaded (32 cores) configurations.

IP: FDN (Flexible Digital Network) for RTOS in the Automotive Industry

Ramon Llorca (speaker), Iñigo Alconada, Andreu Montiel

Abstract

This article introduces a novel FDN (Flexible Digital Network) concept based on the existing nPDU Gateway in AutoSAR. FDN has been developed in RTOS (Real-Time Operating Systems) for the Automotive industry. FDN is a bundle of technologies for the use in an automotive ethernet network for a flexible, scalable, and future proven communication concept. This development tries to demonstrate the importance of determinism in Zonal E/E architectures that are substituting the previous Domain-based E/E architectures that were not really scaling correctly with the proliferation of electronics and newer communication systems inside of a vehicle.
This architectural change and the necessity of this new communication concepts come together with a paradigm change in the development process of SDV (Software Defined Vehicle) Programs, in which is necessary to integrate in the most efficient way the more advanced computation resources for edge computing and HPC platform inside of a vehicle and combining this with cloud computing.
The level of complexity of the new SDV programs requires a Reference Architecture that can be used to consistently and robustly to validate the addition of new features in a car and its proper deployment in the field. By using the FDN concept developed by Technica, we can create a reference system architecture that can be flexibly adapted to any car architecture and used to ensure architecture scalability. Furthermore, this reference architecture should be used to validate the feasibility and life expectancy of the system architecture defined. This reference architecture also provides the KPIs required to perform the design and validation. In this way, the deployment of new features in the field can be sped up, meeting the customer expectations, and fulfilling the market needs.
Those technologies will be also integrated into the ASCENDER project from Barcelona Supercomputing Center (BSC) which aims to explore and further advance the utilization of edge computing applications within a vehicle.

WiP: Re-configurable and Scalable Honeynet for Cyber-Physical Systems

Luís Sousa, José Cecílio (speaker), Pedro Ferreira, Alan Oliveira

Abstract

Industrial Control Systems (ICS) constitute the backbone of contemporary industrial operations, ranging from modest heating, ventilation, and air conditioning systems to expansive national power grids. Given their pivotal role in critical infrastructure, there has been a concerted effort to enhance security measures and deepen our comprehension of potential cyber threats within this domain. To address these challenges, numerous implementations of Honeypots and Honeynets intended to detect and understand attacks have been employed for ICS. This approach diverges from conventional methods by focusing on making a scalable and reconfigurable honeynet for cyber-physical systems. It will also automatically generate attacks on the honeynet to test and validate it. With the development of a scalable and reconfigurable Honeynet and automatic attack generation tools, it is also expected that the system will serve as a basis for producing datasets for training algorithms for detecting and classifying attacks in cyber-physical honeynets.

SESSION 2: SOFTWARE VERIFICATION AND CODE GENERATION

RP: Modelling Task Priority in Symbolic Predictive Analysis for Embedded Software in Ada

Ranjani Krishnan, Ashutosh Gupta (speaker)

Abstract

The concurrent, embedded software in safety-critical systems is complex, and analysing all possible execution paths of such programs is challenging. Symbolic predictive analysis statically analyses a given concrete execution trace of a concurrent program to determine whether any feasible permutation of the given trace violates specified properties. The existing tools do not consider the priority of threads in their modelling of concurrent systems. In this work, we apply a novel static analysis technique based on symbolic predictive analysis for the formal verification of concurrent programs in embedded systems with pre-defined priorities and schedules for the tasks. We consider the given trace as a total order of events coupled with a partial order for the causal model, including several alternative interleavings. Additionally, we include the constraints for the order of event execution by considering the priority of the tasks. We develop a tool chain customized for Ada code, encode the constraints for priority and use it for verifying the onboard software of an aerospace launch vehicle. We accurately model the execution environment for this concurrent software and verify it, along with other benchmarks like concurrency litmus tests and mutual exclusion algorithms. Our experimental results demonstrate the correctness of our modelling and the suitability of our tool for verification of critical Ada software.

RP: A Framework for Static Analysis and Verification of Low-Level RTOS Code

Vignesh Manjunath (speaker), Marcel Baunach

Abstract

Modernembeddedsoftwaredevelopmentusesmodel-basedmethodstosupportlong-term maintenance, portability, and correctness. A growing trend is to use formal methods to create software models and verify their correctness against requirement specifications. However, modeling and verifying low-level Real-Time Operating Systems (RTOS) or Basic Software (BSW) code sequences remains a major challenge, as it requires correctness against the internal hardware behavior and timing. To ensure this correctness, we need formal models of the complex hardware architecture, and due to the increased model complexity, the verification can lead to a state space explosion. In this paper, we mitigate these challenges by using an existing static Worst-Case Execution Time (WCET) analysis tool, OTAWA, for microarchitecture analysis. We use the intermediate results of the WCET analysis as input to our process, which verifies the correctness of the low-level implementations against the runtime effects of the hardware (e.g., synchronization dependencies, memory race conditions) and analyzes the timing and performance of the low-level code with respect to the data hazards in the pipeline. After successful verification, the results can be used in a formal method environment to model and verify the low-level code for correctness against the timing and requirement specifications. We demonstrate the proposed framework by analyzing and verifying the low-level context switch sequence of a classic AUTOSAR-based RTOS and the kernel startup sequence of FreeRTOS for correctness against hardware effects in the AURIX TriCore architecture. In addition, we show an empirical evaluation of our framework to examine the scalability, performance, and state space.

WiP: Algebraic Effects and Static Analysis for Safety-Critical Applications in Fuzion

Fridtjof Siebert (speaker), Michael Lill, Max Teufel

Abstract

This work-in-progress paper presents the introduction of algebraic effects to the Fuzion language and how algebraic effects can be used in the context of safety-critical systems.
Fuzion is a modern, general purpose programming language that unifies functional and object-oriented paradigms into a pure functional language. Algebraic effects are used to represent and manage non-functional aspects like I/O operations or mutable state. Static analysis is used extensively at several stages in the Fuzion toolchain to verify different correctness aspects of the application.
We start with a condensed overview of the Fuzion language to then describe how algebraic effects are used to represent non-functional aspects. The Fuzion toolchain will be explained and how static analysis is used to build and validate applications. Finally, it will be shown how algebraic effects can be used to model aspects relevant to safety-critical systems.

SESSION 3: SECURITY AND SAFETY IN EMBEDDED SYSTEMS

RP: Zero-Trust Design and Assurance Patterns for Cyber-Physical Systems

Saqib Hasan (speaker), Isaac Amundson, David Hardin

Abstract

Security is paramount in all mission-critical domains, including the aerospace industry. Cyber-attacks are increasing both in number and sophistication. Zero-trust is an emerging initiative that has proven very effective for enterprise systems in the Information Technology domain; however, research is lacking on applicable zero-trust mechanisms and their assurance for cyber-physical systems (CPS). We have already identified various zero-trust mechanisms in our previous work. In this paper, we present our zero-trust architecture design patterns and provide a methodology for the assurance of these mechanisms. Towards this objective, we have identified an initial set of assurance patterns covering individual zero-trust components in a system design. Our design and assurance patterns are made available to system engineers in pattern libraries. Engineers can model system architectures and utilize one or more of these patterns to provide design assurance based on individual zero-trust security requirements to improve the overall system cyber-security. To demonstrate our approach, we apply our assurance patterns to an unmanned aerial vehicle surveillance application. We discuss how our framework leverages the use of these patterns to develop zero-trust-enabled systems with different security requirements. Furthermore, our assurance patterns enable engineers to identify any design flaws and correct them during the initial system design phase, thus saving development time, effort, and cost. As a result, the overall approach can be utilized to design system models with specific zero-trust security requirements to improve the security posture of a CPS.

RP: Assuring the Safety of Rechargeable Energy Storage Systems in Electric Vehicles

Faiz Ul Muram, Muhammad Atif Javed, Paul Pop (speaker)

Abstract

Energy storage systems, especially lithium-ion batteries have gained significant attention and interest due to their potential in storing electrical energy and environmental sustainability. They play a crucial role in electric vehicles and significantly impact their performance, particularly in terms of electric driving range and quick acceleration. Despite their advantages, lithium-ion batteries also have limitations. These include the potential for thermal runaway, which can lead to safety hazards if not properly managed, such as outgassing, fire, and explosion that in turn cause significant property damage and fatalities. Published studies on road vehicles have not adequately considered the safety assurance of rechargeable energy storage systems in accordance with ISO 26262 standard. Accordingly in this paper, we focus on the safety assurance of a battery management system (BMS) that prevents thermal runaway and keeps lithium-ion batteries safe in electric vehicles. To this end, the safety life cycle process is performed. At first, the potential hazards that lead to thermal runaway impacting the functions of electric vehicles have been identified and safety goals related to means for preventing and controlling hazards are formulated. Next, the functional safety requirements are derived from each safety goal, and subsequently technical safety requirements are derived. To demonstrate the acceptable safety of electric vehicles using the BMS strategy, the safety cases are developed from the functional safety activities. The safety contracts are derived from battery specifications and chemistry and are associated with safety cases that provide the means for performing necessary adaptations at the operational phase. We leveraged a simulation for performing the verification and validation as well as fine tuning of the BMSstrategy. Simulation data is gathered, and the critical parameters are monitored to determine safety violations, control actions are triggered to resolve them, and safety cases are updated to reflect the current system safety.

IP: Implementing Unsafe Features on top of a Safe Intermediate Language

S. Tucker Taft (speaker)

Abstract

It is common to translate a programming language to an intermediate representation as part of interpreting or compiling a program written in the language. The question is what are the implications when we try to implement normally unsafe features, such as pointers into a heap with user-controlled deallocation, on top of an intermediate language that is inherently safe and provides no user-controlled deallocation. A similar question arises in the implementation of exceptions on top of an intermediate language that has no exceptions, but has lightweight threading. This talk will discuss how we are addressing these two challenges as part of building an Ada implementation on top of a parallel virtual machine, and evaluate the result.

WiP: Software-based Security Framework for Edge and Mobile IoT

José Cecílio (speaker), Alan Sa, André Souto

Abstract

With the proliferation of Internet of Things (IoT) devices, ensuring secure communications has become imperative. Due to their low cost and embedded nature, many of these devices operate with computational and energy constraints, neglecting the potential security vulnerabilities that they may bring. This work-in-progress is focused on designing secure communication among remote servers and embedded IoT devices to balance security robustness and energy efficiency. The proposed approach uses lightweight cryptography, optimizing device performance and security without overburdening their limited resources. Our architecture stands out for integrating Edge servers and a central Name Server, allowing secure and decentralized authentication and efficient connection transitions between different Edge servers. This architecture enhances the scalability of the IoT network and reduces the load on each server, distributing the responsibility for authentication and key management.

WiP: A Framework for Improving Portability and Ensuring Correctness of Operating System Kernels

Vignesh Manjunath (speaker), Marcel Baunach

Abstract

Traditional embedded Real-Time Operating Systems (RTOS) or Basic Software (BSW) implementations typically require manual porting to new hardware platforms. However, this approach can be time-consuming and error-prone, especially given the frequent introduction of new or upgraded hardware architectures. In addition, traditional testing methods may not fully capture the complexity and nuances of the system, making it difficult to ensure correctness and dependability. To address these challenges, we propose a comprehensive methodology that integrates formal methods, a WCET tool, and a code generation technique. We use formal methods to create models and verify their correctness against functional and non-functional specifications or properties such as safety, liveness, and timing. We use the WCET tool as a microarchitecture analyzer to analyze the low-level binary code. The intermediate results of the tool are used to verify the correctness of the software implementation against the runtime effects of the hardware, such as data/memory race conditions. Finally, the code is generated from the formal models. Our proposed framework simplifies the maintenance of RTOS or BSW implementations while ensuring their correctness and partially automating portability to new hardware architectures.

SESSION 4: MACHINE LEARNING AND OPTIMIZATION FOR EMBEDDED SYSTEMS

RP: MLino Bench: A Comprehensive Benchmarking Tool for Evaluating ML Models on Edge Devices

Vlad-Eusebiu Baciu (speaker), Johan Stiens, Bruno Da Silva

Abstract

In today's rapidly evolving technological landscape, Machine Learning (ML) has become an integral part of our daily lives, ranging from recommendation systems to advanced medical diagnostics and autonomous vehicles. As ML continues to advance, its applications extend beyond conventional boundaries. With the continuous refinement of the models and frameworks, the possibilities for leveraging these technologies into devices that traditionally lacked any form of computational autonomy are ever-expanding. This shift towards embedding ML capabilities directly into edge devices brings new challenges due to the stringent limitations these devices have in terms of memory, power consumption, and cost. The ML models implemented on such devices must find an equilibrium between memory footprint and performance, attaining a classification time that fulfills real-time demands and maintains a similar level of accuracy as the desktop version. Without automated assistance in managing these considerations, the complexity of evaluating multiple models can lead to suboptimal decisions. In this paper, we introduce MLino Bench, an open-source benchmarking tool tailored for assessing lightweight ML models on edge devices with limited resources and capabilities. The tool accommodates various models, frameworks, and platforms, presenting a meticulous design that enables a comprehensive evaluation directly on the target device. It encompasses crucial metrics such as on-target accuracy, classification time, and model size, providing a versatile framework that assists practitioners in decision-making when deploying models to such devices. The tool employs a fully streamlined benchmark flow involving training the ML model in a high-level interpreted language, porting, compiling, flashing, and finally benchmarking on the actual target. Our experimental evaluation of the tool highlights its flexibility in assessing multiple ML models across different model hyperparameters, frameworks, datasets, and embedded platforms. Furthermore, a distinctive advantage compared to state-of-the-art ML benchmarking tools is the inclusion of classical ML models, including Random Forests, Decision Trees, Support Vector Machines, Naive Bayes, and more. This sets our tool apart from others that predominantly emphasize only neural network models. Due to this inclusive approach, our tool facilitates the evaluation of ML models across a broad spectrum of devices, ranging from resource-constrained edge devices to those with medium and advanced computational capabilities.

RP: An Autoencoder Architecture for Network Intrusion Detection in Embedded Systems

Niccolò Borgioli (speaker), Federico Aromolo, Linh Thi Xuan Phan, Giorgio Buttazzo

Abstract

Nowadays, security threats are becoming an increasingly relevant concern in cyber-physical systems. Cyber attacks on these systems are not only common today but also increasingly sophisticated and constantly evolving. One way to secure the system against such threats is by using intrusion detection systems (IDSs) to detect suspicious or abnormal activities characteristic of potential attacks. State-of-the-art IDSs exploit both signature-based and anomaly-based strategies to detect network threats. However, existing solutions mainly focus on the analysis of statically defined features of the traffic flow, making them potentially less effective against new attacks that cannot be properly captured by analyzing such features.

This paper presents an anomaly-based IDS approach that leverages unsupervised neural models to learn the expected network traffic, enabling the detection of unknown novel attacks (as well as previously-known ones). The proposed solution uses an autoencoder to reconstruct the received packets and detect malicious packets based on the reconstruction error.
A careful optimization of the model architecture allowed improving detection accuracy while reducing detection time. The proposed solution has been implemented on a real embedded platform, showing that it can support modern high-performance communication interfaces, while significantly outperforming existing approaches in both detection accuracy, inference time, generalization capability, and robustness to poisoning (which is commonly ignored by state-of-the-art IDSs). Finally, a novel mechanism has been developed to explain the detection performed by the proposed IDS through an analysis of the reconstruction error.

RP: Gradient Descent Algorithm for the Optimization of Fixed Priorities in Real-Time Systems

Juan M. Rivas (speaker), J. Javier Gutiérrez, Ana Guasque, Patricia Balbastre

Abstract

This paper considers the offline assignment of fixed priorities in partitioned preemptive real-time systems where tasks have precedence constraints. This problem is crucial in this type of systems, as having a good fixed priority assignment allows for an efficient use of the processing resources while meeting all the deadlines. In the literature, we can find several proposals to solve this problem, which offer varying trade-offs between the quality of their results and their computational complexities. In this paper, we propose a new approach, leveraging existing algorithms that are widely exploited in the field of Machine Learning: Gradient Descent, the Adam Optimizer, and Gradient Noise. We show how to adapt these algorithms to the problem of fixed priority assignment in conjunction with existing worst-case response time analyses. We demonstrate the performance of our proposal on synthetic task-sets with different sizes. This evaluation shows that our proposal is able to find more schedulable solutions than previous heuristics, approximating optimal but intractable algorithms such as MILP or bruteforce, while requiring reasonable execution times.

IP: Software verification and Generative AI – Some practical examples and considerations

Maurizio Martignano (speaker), Andrea Damiani, Daniele Gui, Sabina Magalini and Leonardo Nucciarelli

Abstract

Software verification, that is requirements baseline analysis, technical specification analysis, design analysis and code and testing analysis, is a crucial aspect of software development, ensuring that the products of each development phase satisfy the conditions imposed at the start of that phase.
Traditional software verification techniques often rely on manual effort, which can be time-consuming and error prone. However, with recent advancements in Generative Artificial Intelligence (AI) and Large Language Models (LLMs), there is a growing opportunity to automate and improve software verification activities.
This paper describes how Generative AI, particularly LLMs, can facilitate software verification activities, including understanding of documentation, code analysis, bug detection and testing.
Benefits are presented together with the associated challenges and limitations, especially the potential risk of exposing sensitive and proprietary information.

SESSION 5: REAL-TIME SYSTEMS AND THEIR ANALYSIS

RP: Toward Linux-based safety-critical systems — Execution time variability analysis of Linux system calls

Markel Galarraga (speaker), Charles-Alexis Lefebvre, Jon Perez-Cerrolaza, Jose A. Pascual

Abstract

Latest transportation and industrial domain safety-critical applications, such as autonomous vehicles and collaborative robots, exhibit a combination of escalating software complexity and the need to integrate diverse software stacks and machine learning algorithms, consequently demanding complex high-performance hardware. Linux's extensive platform support and library ecosystem make it a valuable general purpose operating system for developing complex software systems. However, Linux has not been designed to comply with safety standards and does not provide execution path determinism and execution time guarantees. In this context, several research initiatives have studied the usage of Linux for developing complex safety-related systems, focusing on topics that include its development process, isolation architectures, or test coverage estimation. Nonetheless, execution-time analysis and providing temporal guarantees is still a challenge. This work extends the current test coverage analysis based on execution paths with the analysis of the Linux system calls' execution-time variability and proposes a method for estimating the worst-case execution time, forming a sound approach for an in-depth analysis of the Linux kernel execution paths and execution times for safety-related systems. The proposed method is applied to a representative use case that implements an Autonomous Emergency Brake application in an NVIDIA Jetson Nano board connected to the CARLA autonomous driving simulator.

RP: Using MAST for Modeling and Response-Time Analysis of Real-Time Applications with GPUs

Iosu Gomez (speaker), Unai Díaz de Cerio, Jorge Parra, Juan M. Rivas, J. Javier Gutiérrez, Michael González Harbour

Abstract

The ever increasing computing demands in embedded systems is driving the adoption of hardware accelerators such as GPUs, which offer powerful platforms that can compute parallel workloads efficiently. Relevant critical applications that benefit from such platforms, for instance autonomous driving, usually impose additional real-time requirements that must be met to guarantee the correctness of the systems. In this paper, we propose exploiting readily available and extensively validated techniques to model and analyze real-time systems with GPUs. Specifically, we propose a methodology to employ the MAST model to characterize such systems, and different variants of the Offset-Based Response-Time Analysis techniques to validate the real-time requirements. We verify our approach with a real industrial application sourced from the railway industry. Through a comprehensive evaluation involving synthetic and real task-sets, we characterize the applicability of the approach, and we also show how estimated worst-case response times are aligned with real measurements up to 87.2%.

IP: Using AdaCore's GNAT for CUDA for Safety Critical GPU Code Development and Verification

Dimitris Aspetakis, Matina Maria Trompouki (speaker), Leonidas Kosmidis, Jose Ruiz and Gabor Marosy

Abstract

In this short Industrial paper, we describe our experience using AdaCore's GNAT for CUDA toolchain which is currently in closed beta, in the context of the ESA-funded project "Formal Methods for GPU Software Development and Verification". The tool allows the development of Safety Critical GPU code for NVIDIA GPUs in Ada and SPARK instead of CUDA. In the case of SPARK, the absence of runtime errors can be proven, potentially allowing reaching up to platinum adoption level. All our project developments are released as open-source, serving as a demonstrator of the tool capabilities and as learning resources for future developments.

WiP: An Iterative Benchmark Configuration Method for Quantifying Multi-Core Interference

Sébastien Levieux, Frank Singhoff (speaker), Stéphane Rubini, Philippe Plasson, Pierre-Vincent Gouel, Lee-Roy Malac-Allain, Lucas Miné, Gabriel Brusq

Abstract

Interference within a multi-core architecture may have several origins. Understanding where interference comes from is essential for verification and certification purposes. Unfortunately, the complexity of current architectures makes it difficult to quantify such interference. In this paper, a new approach is introduced that enables benchmark configurations to isolate and quantify interference. An experiment with DMA interference is presented and shows a WCET overhead of up to 0.26% at 25 Mbit/s. This experiment was also able to identify other interference related to DMA, such as interruptive flow overhead, around 3% for 25 Mbit/s, or packet transmission memory access overhead, around 9% for 25 Mbit/s.

WiP: Task-to-Thread Mapping in OpenMP Using Fuzzy Decision Making

Mohammad Samadi (speaker), Tiago Carvalho, Luis Miguel Pinho, Sara Royuela

Abstract

The performance of shared-resource multi-core hardware platforms in complex cyber-physical systems (CPSs), e.g., automotive industry, can be improved using task-based parallelism through OpenMP. However, most CPS require certain level of predictability, which challenges the efficient implementation of the task-to-thread mapping process. This exploratory work build on the fact that existing mapping methods mostly use elementary or heuristic algorithms, and the idea that artificial intelligence (AI) algorithms can be used to enhance the efficiency of such processes. Accordingly, this paper (1) evaluates the suitability of AI-based techniques in improving the performance of task-to-thread mapping in the OpenMP framework, and (2) proposes a hypothesis to perform an intelligent mapping using fuzzy logic for multi-queue schedulers to improve the predictability of the system.

SESSION 6: ADVANCEMENTS IN RTOS AND EMBEDDED SOFTWARE DEVELOPMENT

RP: Unishyper: A Rust-based Unikernel Enhancing Reliability and Efficiency of Embedded Systems

Bo Jiang (speaker), Keyang Hu, Wang Huang, Lei Wang, Ce Mo, Yu Chen, Ju Ren

Abstract

Unikernels are simple, customizable, efficient, and small in code size, which makes them highly applicable to embedded scenarios. However, most existing unikernels are developed and optimized for cloud computing, and they do not fully meet the requirements of high reliability and platform customization in embedded environments. We propose Unishyper, a reliable and high-performance embedded unikernel in Rust. To support memory isolation between user applications as well as user code and kernel code, Unishyper proposes the Zone mechanism on top of Intel MPK. Unishyper further proposes a thread-level unwind strategy for safe fault handling while avoiding memory leakage. Finally, Unishyper supports fine-grained customization, has seamless integration with the Rust ecosystem, and proposes Unilib for function offloading to further reduce image size. Our evaluation results show that Unishyper achieves better performance than peer unikernels on major micro-benchmarks, can effectively stop illegal memory accesses across application boundaries, and has a minimal memory footprint of less than 100KB.

RP: The MATERIAL Framework: Modeling and AuTomatic Code Generation of Edge Real-TIme AppLications under the QNX RTOS

Matthias Becker (speaker), Daniel Casini

Abstract

Modern edge real-time automotive applications are becoming more complex, dynamic, and distributed, moving away from conventional static operating environments to support advanced driving assistance and autonomous driving functionalities. This shift necessitates formulating more complex task models to represent the evolving nature of these applications aptly. Modeling of real-time automotive systems is typically performed leveraging Architectural Languages (ALs) such as Amalthea, which are commonly used by the industry to describe the characteristics of processing platforms, operating systems, and tasks. However, these architectural languages are originally derived for classical automotive applications and need to evolve to meet the needs of next-generation applications. This paper proposes an automatic framework for the modeling and automatic code generation of dynamic automotive applications under the QNX RTOS. To this end, we extend Amalthea to describe chains of communicating tasks with multiple operating modes and to consider the QNX's reservation based scheduler, called APS, which allows providing temporal isolation between applications colocated on the same hardware platform. Finally, an evaluation is presented to compare different implementation alternatives under QNX that are automatically generated by our code generation framework.

IP: HiRTOS: A Multi-Core RTOS writen in SPARK Ada

J. German Rivera (speaker)

Abstract

This industrial presentation will describe the design of HiRTOS (High-Integrity RTOS), a simple real-time operating system kernel and separation kernel written in SPARK Ada. HiRTOS targets safety-critical and security-sensitive embedded software applications that run in multi-core microcontrollers. HiRTOS was designed using the Z notation, as a methodical way to capture correctness assumptions that can be expressed as programming contracts in SPARK Ada. Z is a software modeling notation based on discrete mathematics structures (such as sets, relations and functions) and predicate logic.
Although there are several popular RTOSes for embedded applications that run on small microcontrollers, most of them are not designed with high-integrity applications in mind, and as such are written in C, a notoriously unsafe language. So, it would be desirable to have an RTOS specifically designed for high-integrity applications, and written in a safer language, like Ada or its subset SPARK Ada, even if application code is written in C/C++. Modern versions of Ada and SPARK Ada have programming-by-contract constructs built-in in the language, which allows the programmer to express correctness assumptions (contracts) as part of the code. One challenge when doing programming-by-contract is to be aware of all the correctness assumptions that can be checked in programming contracts. Describing software design in a formal notation, such as the Z notation, can help identify/elicit correctness assumptions in a more methodical way than just thinking of them as we write code.
Z is a software modeling notation based on discrete mathematics structures (such as sets, relations and functions) and predicate logic. With Z, data structures can be specified in terms of mathematical structures and their state invariants can be specified using mathematical predicates. Specifying (designing) data structures at a higher-level of abstraction using discrete-math structures can lead to simpler and more elegant code. The pre-conditions and post-conditions of the operations that manipulate the data structures can also be specified using predicates. Using Z for this purpose encourages a rigorous and methodical thought process to elicit correctness properties, in a systematic way.
The code of HiRTOS is written in SPARK Ada, a high integrity subset of the Ada programming language. HiRTOS data types were modeled in Z at a level of abstraction that can be mapped directly to corresponding data types in SPARK Ada.

WiP: Supporting Ada in the ROSE compiler

Peter Pirkelbauer (speaker), Chunhua Liao, Pei-Hung Lin, David Wright, Charles Reynolds, Daniel Quinlan

Abstract

The Ada programming language has been used for the development of many embedded and safety critical applications. With the evolving Ada programming language, the software community needs better tool support to aid in the rejuvenation of large legacy code bases.
This work discusses our progress of adding Ada support to ROSE, a mature source-to-source translation infrastructure. The paper will discuss the design of ROSE, the extensions required for adding Ada, difficulties we encountered with processing existing code bases, and several prototype analysis and translation tools enabled by the new Ada support in ROSE.

WiP: Improving availability in a robotic application without loss of safety

Gema Rincon (speaker), Carlos-F. Nicolas, Tomaso Poggi

Abstract

In our automated and industrialized world, ensuring safety in human-robot interaction is essential, a complex engineering task especially in dynamic environments. The widespread adoption of collaborative and autonomous robots across various sectors underscores the critical need for robust safety measures. This article examines the current state of safety in collaborative robotics and proposes a strategy for assessing the safety of the robot task against the indications of existing standards. If the task is not considered safe in the current environment, a new task is sought for the robot, which increases its availability. These addresses dynamic environments where robots and humans coexist, allowing the autonomous robot to make task decisions based on safety considerations.

ORGANIZERS

Conference Chair
Sara Royuela
BSC, Spain

Journal track Co-Chairs
Björn Andersson
Software Engineering Institute - Carnegie Mellon University, USA

Luis Miguel Pinho
ISEP & INESC TEC, Portugal

Industrial track Co-Chairs
Luciana Provenzano
Mälardalen University, Sweden

Michael Pressler
Robert Bosch GmbH, Germany

Work-in-Progress track Co-Chairs
Alejandro R. Mosteo
CUD Zaragoza, Spain

Ruben Martins
Carnegie Mellon University, USA

Tutorial & Education Chair
Maria A. Serrano
NearbyComputing, Spain

Workshop Chair
Sergio Saez
Universitat Politècnica de València, Spain

Exhibition & Sponsorship Chair
Ahlan Marriott
White Elephant GmbH, Switzerland

Publicity Chair
Dirk Craeynest
Ada-Belgium & KU Leuven, Belgium

Local Chair
Nuria Sirvent
BSC, Spain

Web Master
Hai Nam Tran
University of Brest, France

JOURNAL-TRACK COMMITTEE

Adejokun (Peter) Ademola (Lockheed Martin, USA), Al Mok (University of Texas at Austin, USA), Alejandro Mosteo (CUD Zaragoza, Spain), Alwyn Godloe (NASA, USA), Andrea Marongiu (University of Modena and Reggio Emilia, Italy), António Casimiro (University of Lisbon, Portugal), Arne Boralv (Prover Technology, Sweden), Barbara Gallina (Mälardalen University, Sweden), Bernd Burgstaller (Yonsei University, South Korea), C. Michael Holloway (NASA, USA), Cristina Seceleanu (Mälardalen University, Sweden), Doug Schmidt (Vanderbilt University, USA), Frank Singhoff (University of Brest, France), George Lima (Universidade Federal da Bahia, Brazil), Hector Perez Tijero (University of Cantabria, Spain), Isaac Amundson (Rockwell Collins, USA), Johann Blieberger (Vienna University of Technology, Austria), John B Goodenough (CMU, USA), Jérôme Hugues (CMU/SEI, USA), José Cruz (Lockheed Martin, USA), Kristoffer Nyborg Gregertsen (SINTEF Digital, Norway), Laurent Pautet (Telecom ParisTech, France), Leonidas Kosmidis (Barcelona Supercomputing Center, Spain), Mario Aldea Rivas (University of Cantabria, Spain), Matthias Becker (KTH - Royal Institute of Technology, Sweden), Michael González Harbour (University of Cantabria, Spain), Patricia López Martínez (University of Cantabria, Spain), Risat Pathan (Chalmers University, Sweden), Sara Royuela (Barcelona Supercomputing Center, Spain), Sergio Iserte (Barcelona Supercomputing Center, Spain), Sergio Sáez (Universitat Politècnica de València, Spain), Shige Wang (Motional, USA), S. Tucker Taft (AdaCore, USA), Tullio Vardanega (University of Padua, Italy), Xiaotian Dai (University of York, England)

INDUSTRIAL-TRACK COMMITTEE

Aida Causevic (Alstom, Sweden), Alexander Viehl (FZI Research Center for Information Technology, Germany), Ana Rodríguez (Silver Atena, Spain), Aurora Agar (NATO, Netherlands), Behnaz Pourmohseni (Robert Bosch GmbH, Germany), Claire Dross (AdaCore, France), Elena Lisova (Volvo CE, Sweden), Enricco Mezzetti (Barcelona Supercomputing Center, Spain), Federico Aromolo (Scuola Superiore Sant'Anna, Italy), Helder Silva (Edisoft, Portugal), Hugo Torres Vieira (Evidence Srl, Italy), Irune Agirre (Ikerlan, Spain), Jordi Cardona (Rapita Systems, Spain), José Ruiz (AdaCore, France), Joyce Tokar (Raytheon, USA), Luciana Alvite (Alstom, Germany), Marco Panunzio (Thales Alenia Space, France), Patricia Balbastre Betoret (Valencia Polytechnic University, Spain), Philippe Waroquiers (Eurocontrol NMD, Belgium), Raúl de la Cruz (Collins Aerospace, Ireland), Santiago Urueña (GMV, Spain), Stef Van Vlierberghe (Eurocontrol NMD, Belgium)

WORK-IN-PROGRESS-TRACK COMMITTEE

Alan Oliveira (University of Lisbon, Portugal), J. Javier Gutiérrez (University of Cantabria, Spain), Jérémie Guiochet (LAAS-CNRS, France), José Cecílio (University of Lisbon, Portugal), Kalinka Branco (University of São Paulo, Brazil), Katherine Kosaian (University of Iowa, USA), Kevin Cheang (AWS, USA), Kristin Yvonne Rozier (Iowa State University, USA), Leandro Buss Becker (University of Manchester, UK), Li-Pin Chang (National Yang Ming Chiao Tung University, Taiwan), Mathias Preiner (Stanford University, USA), Raffaele Romagnoli (Carnegie Mellon University, USA), Robert Kaiser (RheinMain University of Applied Sciences, Germany), Sara Abbaspour (Mälardalen University, Sweden), Sergi Alcaide (Barcelona Supercomputing Center, Spain), Simona Bernardi (Unizar, Spain), Stefan Mitsch (School of Computing at DePaul University, USA), Teresa Lázaro (Aragon's Institute of Technology, Spain), Tiago Carvalho (ISEP & INESC TEC, Portugal), Yannick Moy (AdaCore, France)