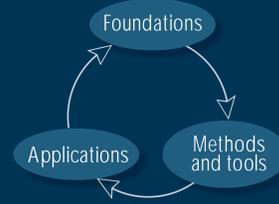
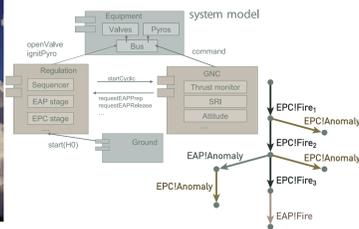


Methods and tools for building embedded systems of guaranteed quality

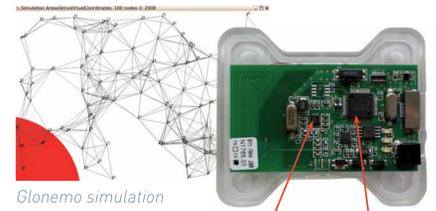


Ariane 5



«Correct start-up» property synthesised by IF and UML architecture model

→ Post-accident validation using Lustre and IF



Glono simulation

CC1100 MSP430

A sensor node

→ Simulation of a wireless sensor network detecting a pollution cloud for analysis of energy consumption



A380: software development based on SCADE

COMPUTER-AIDED SECURITY

- Computed-aided engineering methods for a systematic approach to security:
- Foundations of security
 - Computer-aided verification of cryptographic primitives
 - Software engineering for security
 - Tools for verification of the Common Criteria
 - Information flow and non-interference analysis
 - Testing and monitoring of security properties
 - Secure e-voting

SYSTEMS-ON-A-CHIP

- Virtual prototyping for systems-on-a-chip based on Transaction-Level-Modeling (TLM):
- Validation methods and tools for SystemC
 - Component-based design and transformation of Transaction Level models
 - Non-functional properties at the transaction level (time, energy)

WIRELESS SENSOR NETWORKS

- Providing formal models and virtual prototyping tools for studying energy consumption in wireless sensor networks:
- Distributed and fault tolerant approach to system design
 - Dedicated formal, executable and global models for energy consumption
 - Definition of notions of refinement and abstraction for energy models
 - Faithfulness and calibration of simulators

SOFTWARE VERIFICATION

- Developing theory and tools for scalable software verification:
- Multi-threading and dynamic recursive data structures
 - Verification tools for real-world C/C++/Java programs
 - Complexity analysis of verification problems
 - Assertion checking and termination proofs

VERIFICATION AND VALIDATION TECHNIQUES

- Verification and validation is crucial throughout the design cycle:
- Model checking and verification based on abstract interpretation
 - Simulation and early execution
 - Contract-based verification

LANGUAGE DESIGN FOR EMBEDDED SYSTEMS

- Coping with the growing complexity of embedded hardware and software requires high-level domain-specific languages and associated implementation methods:
- Languages and tool support for parallel programming
 - Non-deterministic languages for virtual prototyping and simulation
 - Aspects and components for synchronous languages, Lustre

IMPLEMENTATION OF EMBEDDED SYSTEMS

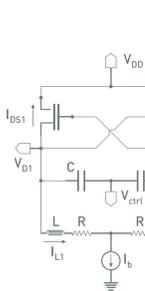
- Platform-dependent techniques:
- Support for correct-by-construction integration of components
 - Property-aware code generation for multi-processors
 - Time and space predictable dynamic memory management
 - Multi-threaded and distributed implementations of synchronous programs

HYBRID SYSTEMS

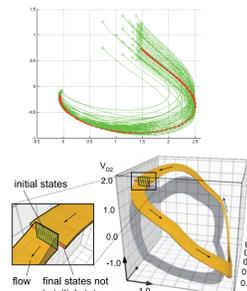
- Enriching the analysis toolbox for engineers and scientists in various domains:
- Theoretical and algorithmic foundations
 - Reachability analysis for continuous and hybrid systems
 - Scheduling and performance evaluation
 - Monitoring temporal properties
 - Systematic simulation and test generation
 - Applications: control, analogue circuits, multi-core computing, systems biology

COMPONENT-BASED DESIGN OF EMBEDDED SYSTEMS

- Modular component framework for Behavior - Interaction - Priority (BIP):
- Structural and compositional verification of programs
 - Component-based design of multi-core systems
 - Property enforcement and controller synthesis



Analog circuit

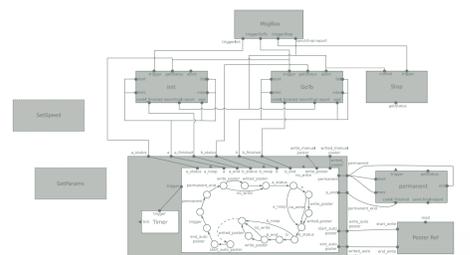


Trajectory based verification

→ Set based verification of oscillation properties



Dala robot



BIP component-model

→ Verification of a BIP model with D-finder, code generation with BIP, and integration into existing software

AWARDS

- 2007: Turing award, the highest recognition in Computer Science, to Joseph Sifakis, shared with Ed Clarke and Alan Emerson
- 2004: Michel Monpetit Prize of the French Academy of Science to Paul Caspi and Nicolas Halbwachs
- 2002: CNRS Silver Medal to Joseph Sifakis



Joseph Sifakis

