Multicore Avionics Operational Analysis

Lightning Talk 2:
User Needs and Requirements

Alex Bashara, Joe Dicklin, Hank Haldin, Anthony Manschula

Project Overview

- Multicore avionics systems
  - Meet the increasing compute demand of modern avionics software
  - Programs competing for resources between cores can introduce interference & affect execution timing behavior
    - Ability to examine and verify the effects of interference is critical for FAA certification
- Reference ARM-based SoC and bare-metal hypervisor
  - Run control applications with the system under extreme load
    - Collect and analyze data on worst-case execution time (WCET)

Problem Statement

- Our project addresses a need for a suite of open-source tools to characterize interference modes in multicore avionics systems
  - Identify potential interference channels on a multicore platform
    - "Control tests" as a baseline that target each channel for use in analysis
  - Set of tools to apply stress and contention to the identified subsystems in a controlled manner
  - Set of tools/methods to demonstrate mitigation of interference channels
  - Integrate testing and analysis tools into unified suite

User Needs

- Non-Technical Nancy
  - Needs technical support from other departments to cover issues
  - Needs a method to communicate with and understand the work of engineers
  - Needs to be able to communicate the value of their department's work to other departments
- Technical Timothy
  - Needs a comprehensive tool suite to validate Multi-core SoCs
  - Needs to satisfy FAA regulations and standards
  - Needs to complete work assigned by engineering managers by deadlines

Project Requirements

- Physical Requirements
  - *ARMv8 processor subsystem*
  - Form-factor: Single-board computer (Raspberry Pi, Pine64 family, etc.)
- Knowledge
  - Linux environments
  - Worst-case execution time and its influencing factors
  - Familiarity with multi-core computer architectures
  - *Documentation providing a sufficient level of detail to allow the user to learn any of the above at a high level*
- Functional/technical requirements
  - Properly and methodically stress the system
    - *Identify major points of resource contention (processor time, memory usage, IO bus usage)*
    - Accurately emulate potential worst-case scenarios (rogue process uses too much CPU time/memory/IO bandwidth)
  - *Demonstrate an upper bound on worst-case execution time for our platform*
  - Provide a user-friendly interface for managing and interpreting test results

Requirement: Hardware Platform

- ARMv8 Architecture
  - Instruction set architecture – defines how the hardware based on it must be implemented and behave
  - Performance and power characteristics
    - Good blend of performance and efficiency
  - Stipulated by our contacts at Boeing

Requirement: Documentation

- Make our tool as accessible as possible for other parties
  - Boeing would like to open-source this at the end of the project
    - Good documentation is critical for this hand-off
- Ensure that we are staying organized as we move through the project
  - Several aspects of this project have little outside resources
    - Ability to quickly replicate successes and trace back steps is key

Requirement: Interference Modes and Worst-Case Execution Time

- Identify major points of resource contention

  - Simultaneously target each point with a control and stress test to gather data on how the control program is affected under stress

- Demonstrate an upper bound on worst-case execution time (critical)

  - Essential for multicore avionics platforms

  - Quantitatively prove that a rogue program cannot reduce system performance beyond a certain point

Applicable Engineering Standards

- FAA: AC 20-193
- IEEE Code of Ethics
- CAST-32A
- SAE Aerospace Standards
- RCTA/DO-178C
- ASTM (American Society for Testing and Standards)
- POSIX (Portable Operating System Interface)
- ARINC 653
- FACE (Future Airborne Capability Environment)

For the Audience

- Conclusion
  - The team has many requirements to fulfill, but our choice of ISA, need for thorough and accesible documentation, as well as definitive proof of WCET are most critical.
- Questions?