Achieving Safety-Critical Determinism with Multicore Processors

Embedded Tech Trends

January 24-25, 2019

Richard Jaenicke

richj@ghs.com

Multicore is Everywhere

Even in Most Regulated Industries

Except Safety-Critical Applications

Strict Determinism is Required for Flight Safety

DAL A is the Strictest Safety Level

- DAL A failure rate is 10^-9/h
  - 1 failure every 114,155 years of continuous operation
- No single HW failure can result in a catastrophic event

Design Assurance Level	Failure condition	Failure Rate
A	Catastrophic	10^{-9}/h
B	Hazardous	10^{-7}/h
C	Major	10^{-5}/h
D	Minor	10^{-3}/h
E	No Effect	n/a

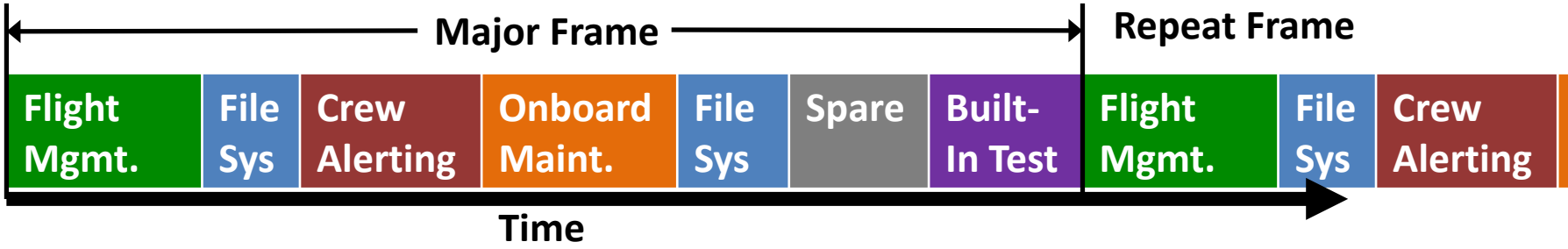
System Complexity Makes Failure Evaluation Difficult

“Lion Air pilots unable to correct for faulty sensor”

Achieving Determinism in a Single-Core World

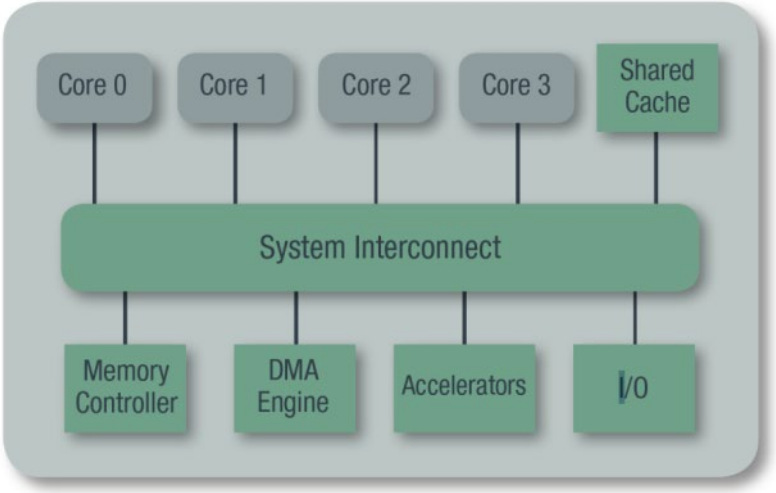
Avoid Interference between Applications through Partitioning of Space and Time (ARINC 653)

- **Memory Space Partitioning**
  - Enforced by CPU's Memory Management Unit (MMU)
- **Processor Time Partitioning**
  - RTOS gives each application a fixed length time window



Multicore is More Complex

Multicore must address contention for shared resources

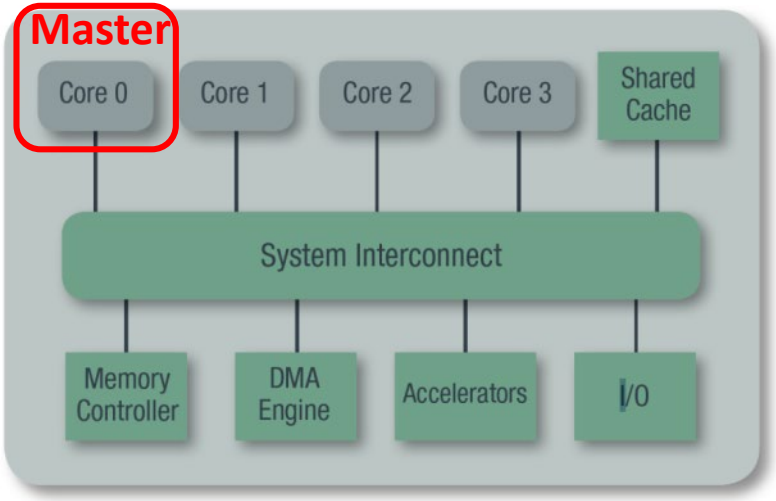


For flight safety, certification authority guidance is in CAST-32A

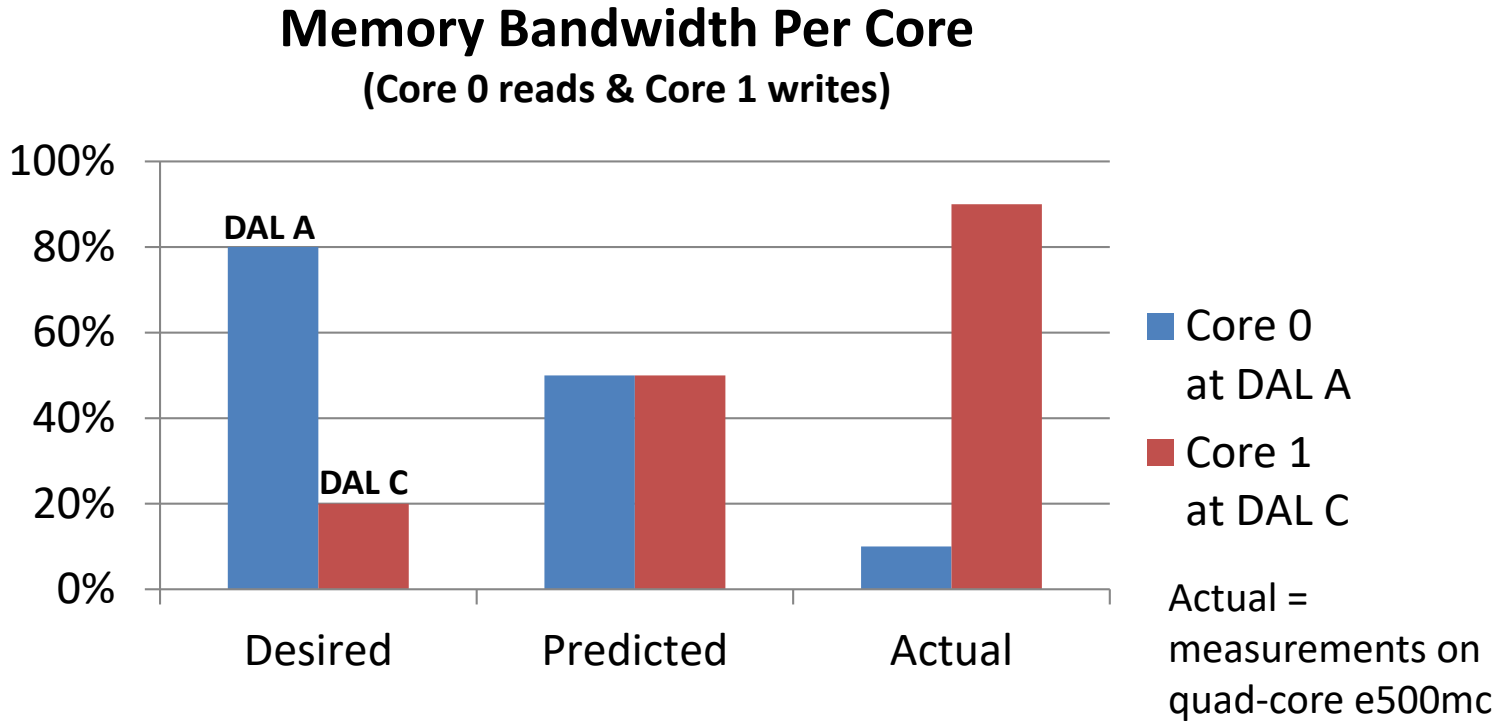
Simple Approach Doesn't Work Well

Try to have one core responsible for all shared resource access

- Possible for I/O, but results in vast under utilization of cores
- **Impossible for memory controller** without running only one core at a time

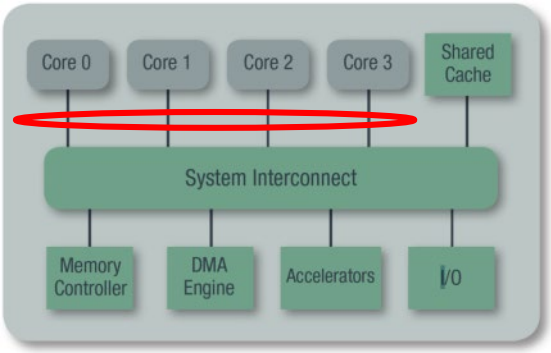


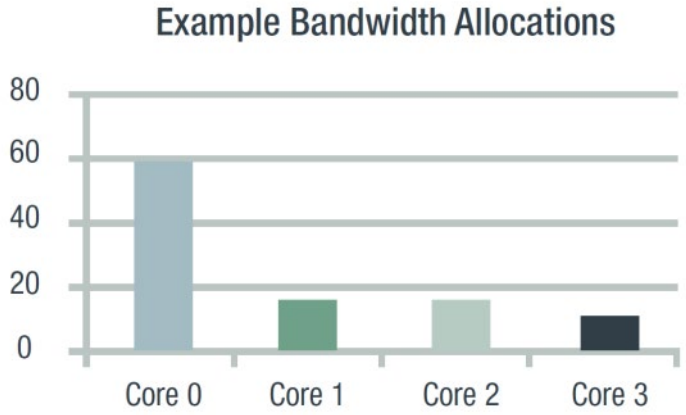
Memory Access can be Very Unfair



General Solution is to Enforce QoS

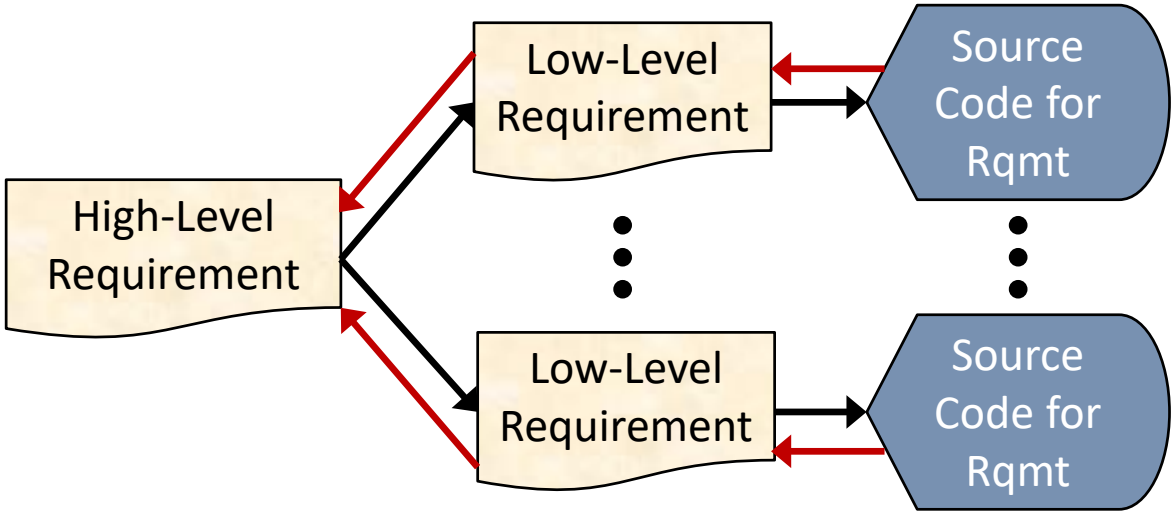
- All shared access goes through on-chip interconnect, so can enforce it there
- Set access thresholds for each time window for each core, enforced by the OS



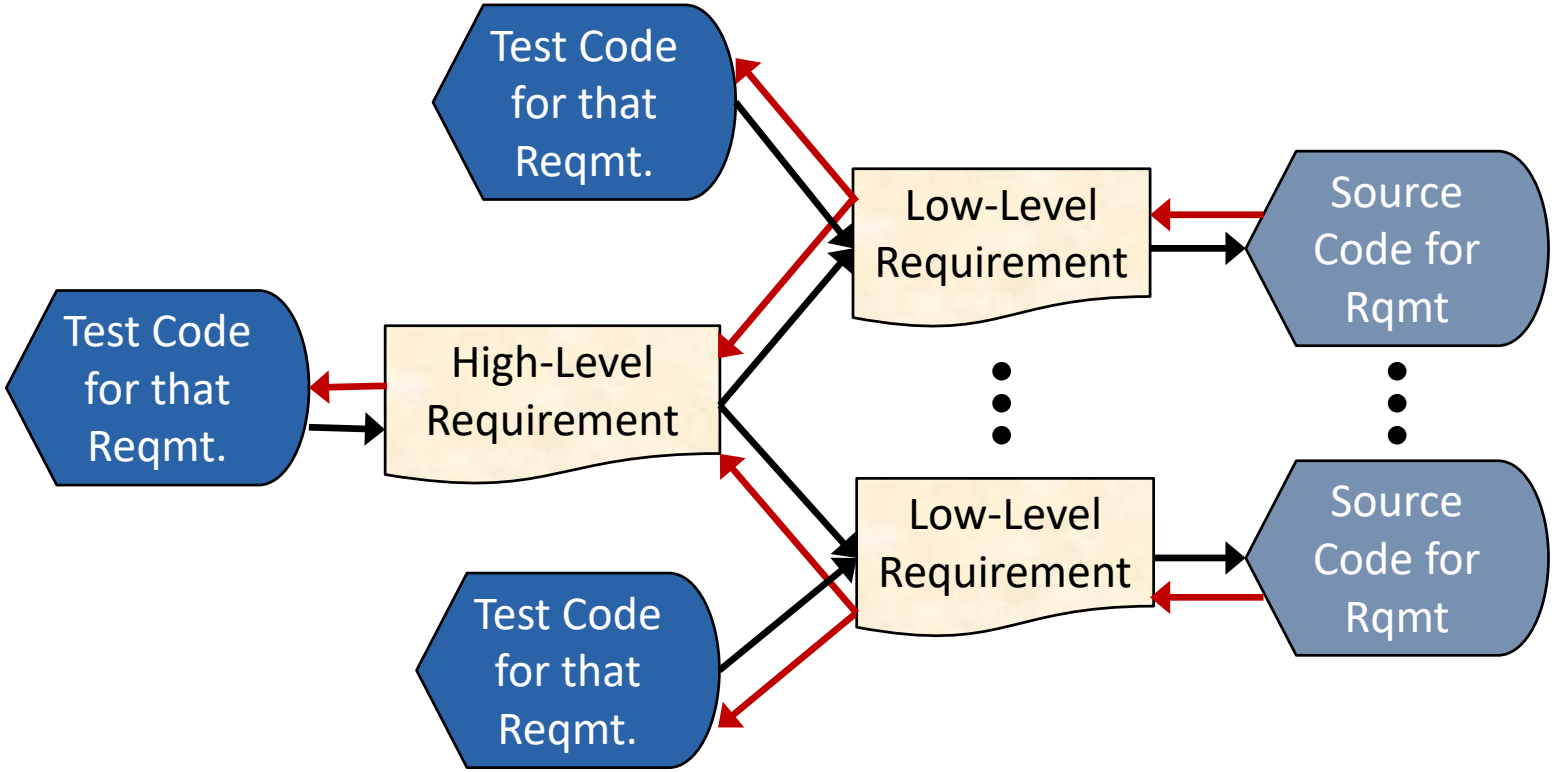


Multiple Layers for Safety Certification

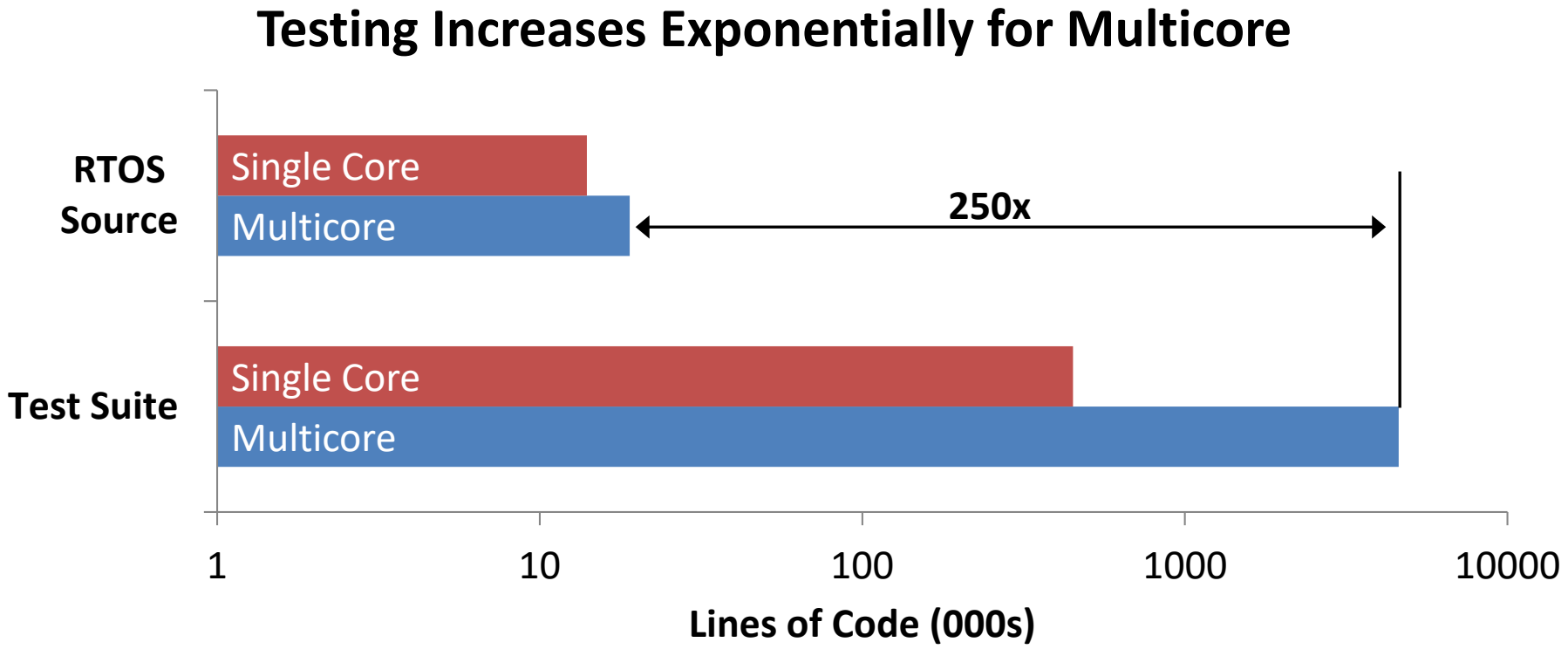
DAL-A Requires Complete Traceability



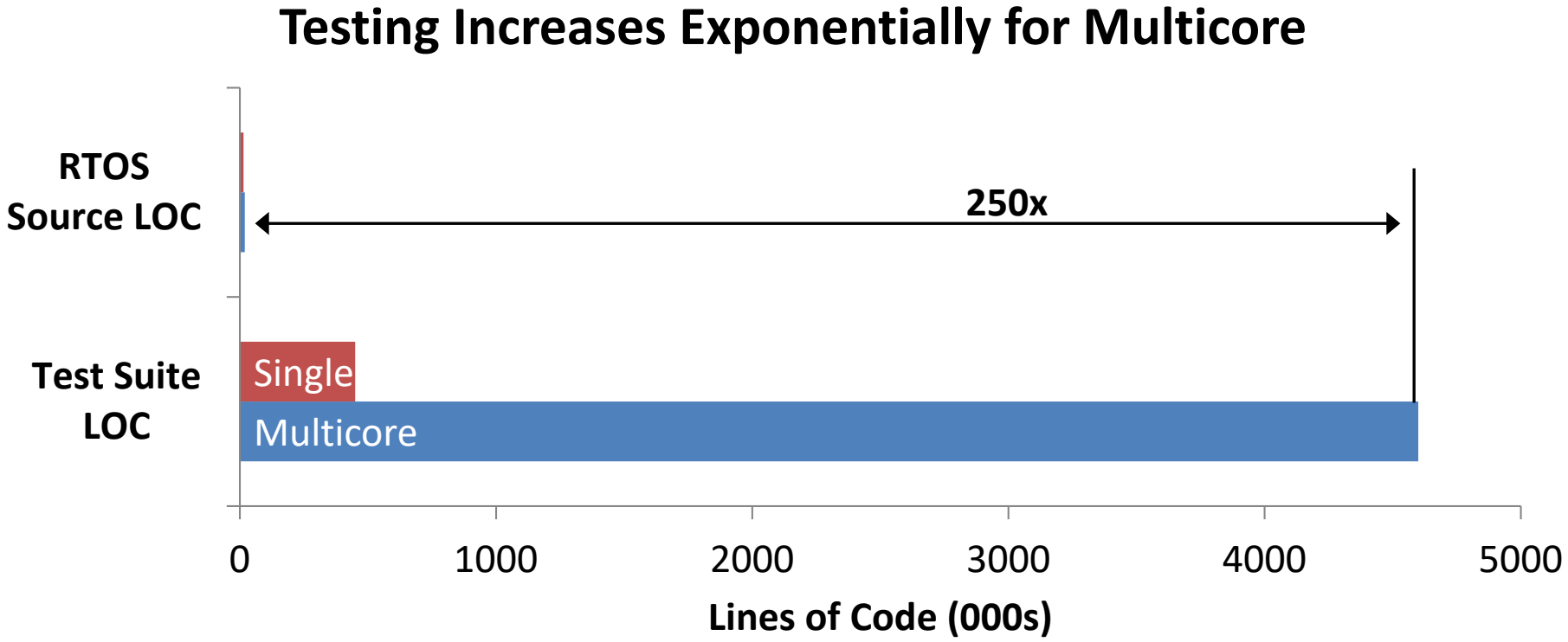
DAL-A Requires Huge Testing



Test Suites Can Be Huge for Multicore



Test Suites Can Be Huge for Multicore



Summary

Deterministic multicore is hard, but achievable

- Contention for shared resources causes unpredictable delays
- Must enforce QoS, such as via bandwidth allocation
- Testing and validation are exponentially harder for multicore

Almost Impossible for System Integrators to do Themselves

- Use suppliers with the most extensive support for multicore interference mitigation, testing, and validation suites

See Also

- FAA Position Paper: CAST 32A Multicore processors
  https://www.faa.gov/aircraft/air_cert/design_approvals/air_software/cast/cast_papers/media/cast-32A.pdf
- Whitepaper: Optimal Multicore Processing for Safety-Critical Applications
  https://www.curtisswrightds.com/infocenter/white-papers/optimal-multicore-processing-for-safety-critical-applications.html
- Website: GHS solutions for Aerospace and Defense
  https://www.ghs.com/AerospaceDefense.html