RTA: Timing Constraints and Worst Case Execution Time

Lee Pike leepike@galois.com

May 15, 2012

RTA Needs Time

Ed Lee's manifesto, *Computing needs time*[1]

- Many abstraction layers.
- Time is not treated as a first-class property.

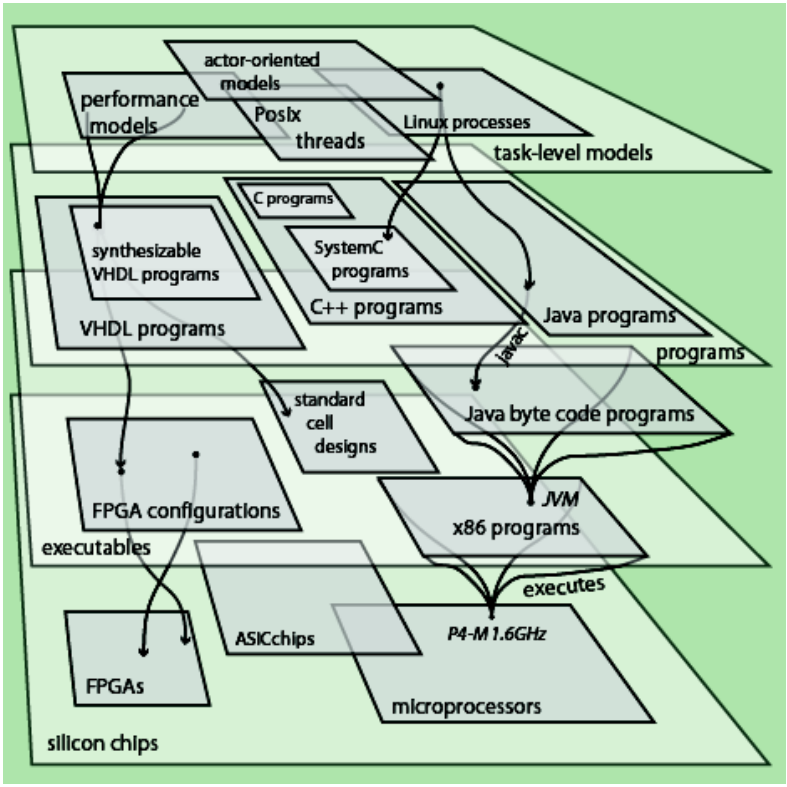

Figure: src: *Computing needs time*

[1] Communications of the ACM, 52(5), 2009.

galois

RTA Needs Time

Exacerbated in RTA:

- Complex-controller computational complexity

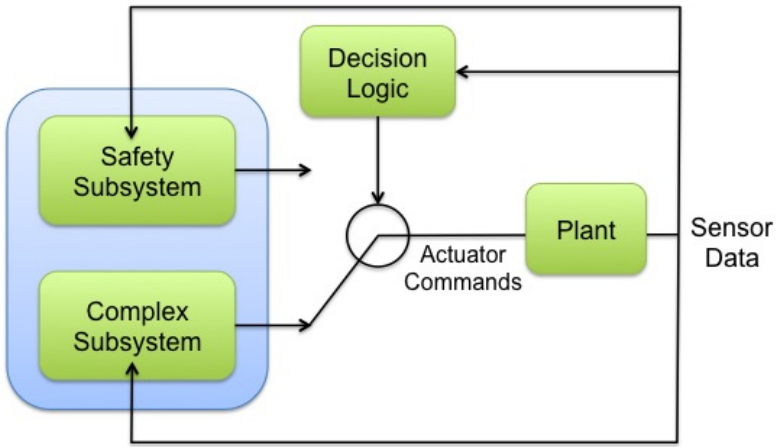

Figure: src: Bak *et al.* The system-level simplex architecture for improved real-time embedded system safety, *RTETAS*, 2009.

galois

RTA Needs Time

Exacerbated in RTA:

- Complex-controller computational complexity
- Monitor complexity

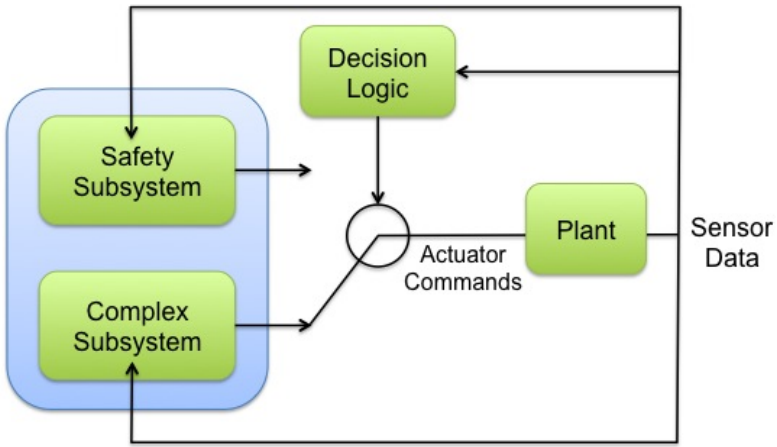

Figure: src: Bak *et al.* The system-level simplex architecture for improved real-time embedded system safety, *RTETAS*, 2009.

galois

RTA Needs Time

Exacerbated in RTA:

- Complex-controller computational complexity
- Monitor complexity
- Switching logic complexity

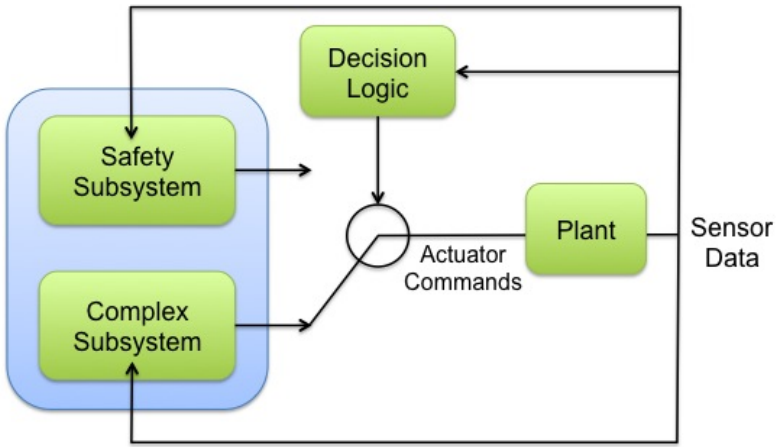

Figure: src: Bak *et al.* The system-level simplex architecture for improved real-time embedded system safety, *RTETAS*, 2009.

galois

RTA Needs Time

Exacerbated in RTA:

- Complex-controller computational complexity
- Monitor complexity
- Switching logic complexity
- Software implementations

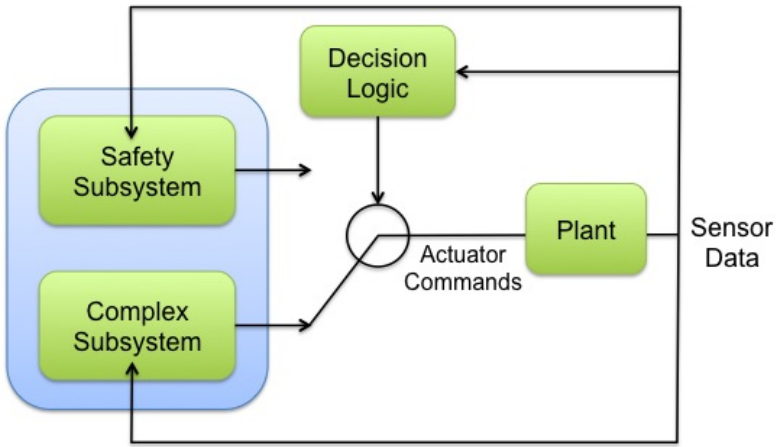

Figure: src: Bak *et al.* The system-level simplex architecture for improved real-time embedded system safety, *RTETAS*, 2009.

galois

RTA Needs Time

Exacerbated in RTA:

- Complex-controller computational complexity
- Monitor complexity
- Switching logic complexity
- Software implementations
- Hardware nondeterminism

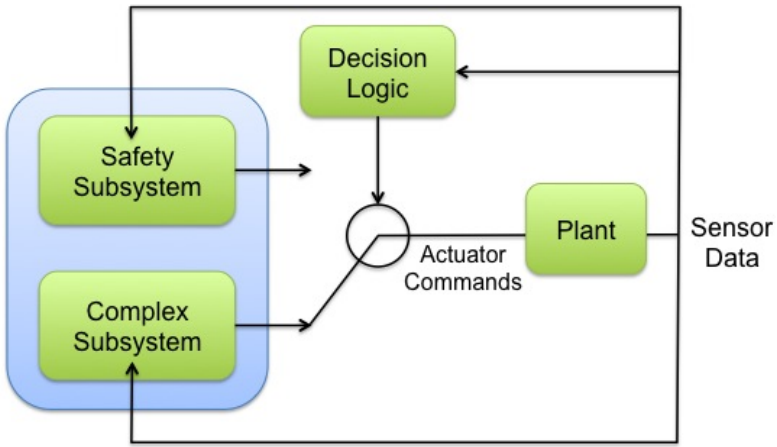

Figure: src: Bak *et al.* The system-level simplex architecture for improved real-time embedded system safety, *RTETAS*, 2009.

galois

RTA Needs Time

Exacerbated in RTA:

- Complex-controller computational complexity
- Monitor complexity
- Switching logic complexity
- Software implementations
- Hardware nondeterminism
- Faults

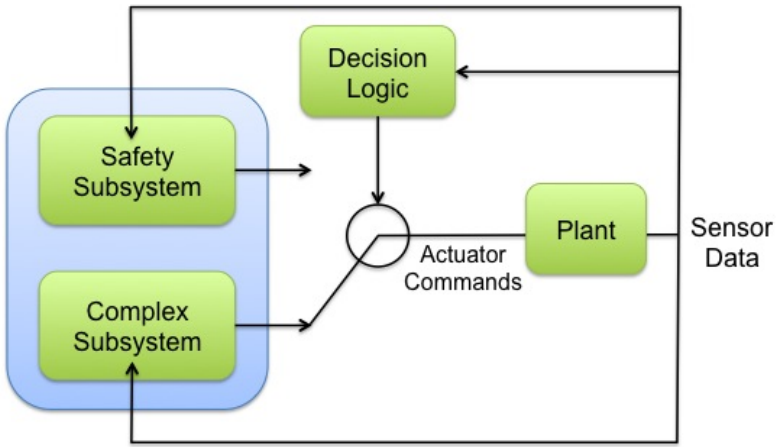

Figure: src: Bak *et al.* The system-level simplex architecture for improved real-time embedded system safety, *RTETAS*, 2009.

galois

Time-Aware RTA

Two classes:

1. RV for monitoring timing properties.
2. RV approaches for controlling temporal overhead.
    1. Time-triggered monitoring.
    2. Networking RTA.
    3. Probabilistic monitoring.

galois

Time-Triggered Monitoring

- Idea: sample observable state frequently enough.
- Approaches:
  - Control-flow (Waterloo): Borzoo Bonakdarpour, Samaneh Navabpour, and Sebastian Fischmeister. Sampling-based runtime verification. In 17th Intl. Symposium on Formal Methods (FM), 2011.
  - Data-flow (Copilot): Lee Pike, Sebastian Niller, and Nis Wegmann. Runtime verification for ultra-critical systems. In Proceedings of the 2nd Intl. Conference on Runtime Verification, LNCS. Springer, September 2011.
- Strengths/Weaknesses:
  - \+ Predictable timing.
  - – Difficult to monitor control-flow properties.

galois

Networking RTA

- Idea: put the monitor on separate hardware and watch transmitted messages.
- Approaches:
  - R. Pellizzoni, P. Meredith, M. Caccamo, and G. Rosu. Hardware runtime monitoring for dependable COTS-based real-time embedded systems. In Proceedings of the 29th Real-Time Systems Symposium (RTSS09), pages 481491, December 2008.
  - Copilot/data-flow if hardware is synchronized.
- Strengths/Weaknesses:
  - \+ No overhead.
  - – Only transmitted messages are observable.

galois

Probabilistic Monitoring

- Idea: statistical measure of the likelihood of satisfying a property.
- Approach—state-estimation using Hidden Markov Models: Scott D. Stoller, Ezio Bartocci, Radu Grosu, Havelund Klaus, Scott A. Smolka, Seyster Justin, and Erez Zadok. Runtime Verification with State Estimation. In In Proc. of RV 2011: 2nd International Conference on Runtime Verification, 2011.
- Strengths/Weaknesses:
  - \+ Hidden state is handled implicitly.
  - – Depends on the fidelity of the learned HMM.

galois

Comparing Approaches

	Property proof	Requires no code instrumentation	Control-flow properties
Time-triggered	depends	depends	✓
Networked	depends	✓	χ
Probabilistic	χ	✓	✓

Figure: Comparing three time-aware RTA paradigms.

galois

Future Research Recommendations

- New Metrics

galois

Future Research Recommendations

- New Metrics
- Multi-Core RTA

galois

Future Research Recommendations

- New Metrics
- Multi-Core RTA
- Empirical Studies

galois

Future Research Recommendations

- New Metrics
- Multi-Core RTA
- Empirical Studies

- WCET & RTA

galois

Future Research Recommendations

- New Metrics
- Multi-Core RTA
- Empirical Studies

- WCET & RTA
- Assurance, Certification, and RTA

galois

Future Research Recommendations

- New Metrics
- Multi-Core RTA
- Empirical Studies

- WCET & RTA
- Assurance, Certification, and RTA
- Scaling-up and System Integration

galois

ACRONYMS

RTA – Run Time Assurance

RV – Run Time Verification

HMM – Hidden Markov Model

WCET – Worst Case Execution Time