Invariant-Based Embedded System Safety Monitor

Carnegie Mellon University

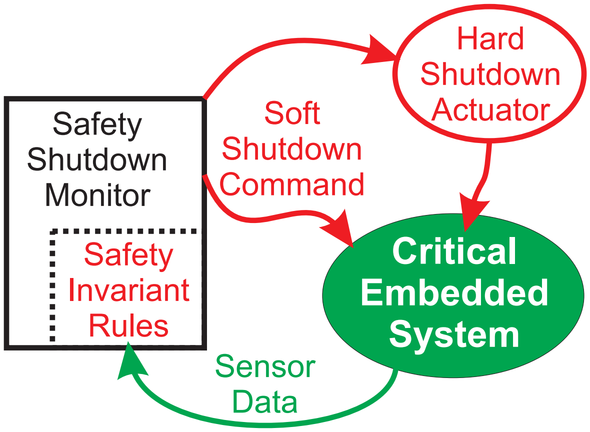
Aaron Kane, Prof. Phil Koopman

Can we create a simple, generic safety shutdown building block?

Motivation:

Many safety-critical or high dependability systems assume that the system or a key component will shut down when it fails or behaves in an unsafe manner. Unfortunately, most common approaches such as 2-of-2 systems don't detect software design or requirements defects.

We will combine our research group's strengths from two recent projects on system safety invariants and autonomous vehicle safety analysis to create a generalized approach to run-time safety monitors. The result will be a set of simple techniques, a generalized software architecture, and a generalized hardware architecture for building a safety shutdown system that accommodates the needs of complex, software-intensive embedded systems.



Key Challenges

- Devise a temporal logic-based invariant specification system that everyday engineers can understand (preferably, with little or no specialized math notation)
- Support orderly, non-zero-time shutdown sequences to avoid equipment damage due to abrupt stops, if safe to do so
- Support a hierarchy of warnings, soft-shutdowns, hard shut-downs, and instant shutdowns
- Create a generic approach that makes minimal assumptions about hardware and software platforms

Approach:

Safety Invariants:

assert(throttle XOR brake)
failed assertion provokes shutdown

+

Fault Tree Analysis

At least one assertion between root and each tree leaf

=

Time-Triggered
Safety Invariant
Rules & Monitoring

Overall Approach:

- Use a fault-tree approach to represent sources of safety violations
- Use simple run-time invariants to specify high level safety behaviors
  e.g. "commanding brake and throttle at same time is unsafe"

Building blocks from previous work:

- A safety shutdown approach for autonomous vehicles (in cooperation with National Robotics Engineering Consortium)
  - High level safety invariant ideas applied to an autonomous vehicle system (live operation on a test vehicle)
  - Used fault trees as a non-mathematical safety specification
- A decomposable approach to creating run-time safety invariants (Jen Black Ph.D. thesis, May 2009)
  - Structured approach to identifying control loop pathways
  - Temporal logic to specify invariants
  - Templates to help allocate safety behaviors to subsystems
  - Discovered new safety problems in industry prototype design
  - Demonstrated feasibility of high level safety invariants

Status:

New start for 2010

- Working on initial problem definition and feasibility demonstration

Timeline:

- Propose basic, generic, zero-time-delay safety shutdown invariant approach by end of 2010

Electrical & Computer ENGINEERING

May 2010 – koopman@cmu.edu – http://www.ece.cmu.edu/~koopman