Industrial Embedded Systems
- Design for Harsh Environment -

Dr. Alexander Walsch
alexander.walsch@ge.com

IN2244
Part VI – Safety Architectures
WS 2014/15

Technische Universität München
Fail-safe and Fail-operational Systems

Safety function

Protection Systems (on demand <1/a)
- safe=OFF
- de-energized
- fail-safe
- fail-off

Control and Monitoring Systems (continuous)
- safe=OFF
- degraded
- fully operational
- safe=ON
Besides providing a specific quality (failure rate) a safety function must be hosted by a specific architecture in context of IEC 61508.

Besides architecture constraints also specific fault detection mechanisms must be realized by the final design. This is expressed by the safe failure fraction (SFF).
Safe Failure Fraction (SFF)

- Failure (this is the same failure rate as in the reliability lecture) can happen in a safe or dangerous way. Detection mechanisms are software enabled in the context of complex systems (involving microcomputers).

- \[ SFF = 1 - \frac{\lambda_{du}}{\lambda_{total}}; \lambda_{total} = \lambda_{du} + \lambda_{dd} \lambda_{su} + \lambda_{sd} \]
1oo1 System

Source:
Goble, Safety instrumented systems verification: practical probabilistic calculation
1oo1 Software

- Reliability (random faults): see previous calculations
- Reliability (systematic faults): highly affected
- Safety: 1oo1 architecture, not used
1oo1 Software

- Still 1oo1.
- Provides some data and control flow checks (self-monitoring)
  - Internal watchdog, acceptance tests
- Use: not used in safety-related applications, reliability increase (depends on application)
1oo2 System

Source:
Goble, Safety instrumented systems verification: practical probabilistic calculation
2003 System

Source: Goble, Safety instrumented systems verification: practical probabilistic calculation
1o01D System

Source: Goble, Safety instrumented systems verification: practical probabilistic calculation
External Diagnostics (MooND Architectures)
1oo2D System

Source: Goble, Safety instrumented systems verification: practical probabilistic calculation
Architectures and Cost

Source: Goble, Safety instrumented systems verification: practical probabilistic calculation
Systematic Failures

- Architecture: common cause failures, dependency failures
  - Freedom from interference
  - Look at common cause failures in previous Markov diagrams
- Software: SIL for software renamed to systematic capability (SC) in IEC61508 Edition 2.0
  - SC N supports a safety function of SIL N