Safety assessment of smart devices

Sofia Guerra
aslg@adelard.com

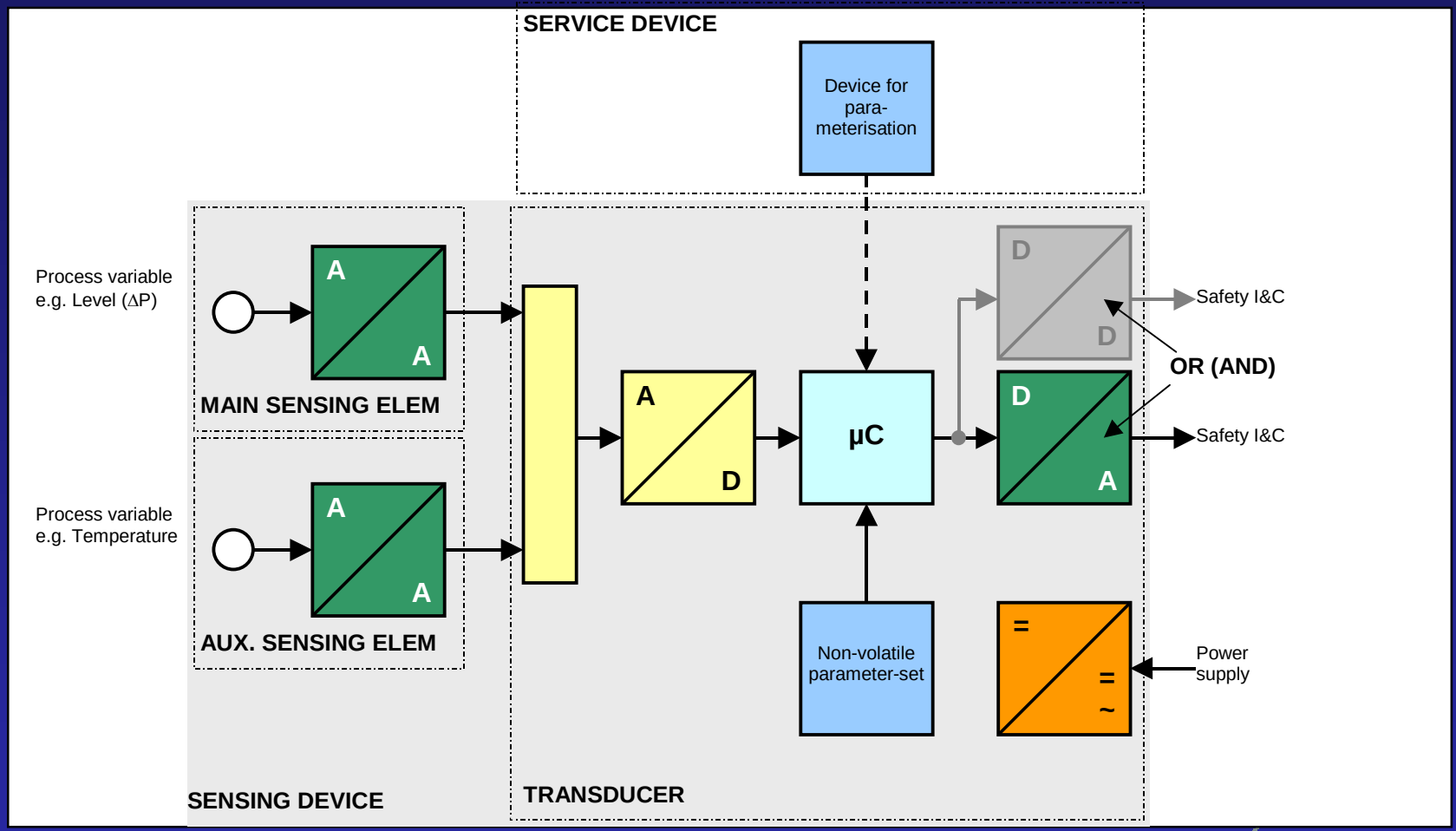
Overview

- Background
- Stakeholders
- Safety justification approaches
- Analysis of a smart device
- Conclusion

Background

- Sensors have been relatively simple analogue devices.
- Industry increasingly using “smart sensors”.
- Smarts have advantages:
  - better accuracy
  - better noise filtering
  - in-built linearisation
  - better on-line calibration
- Safety justification might require information about product and process and knowledge of internal structure.

Smart sensor



Aims

- Exploring the degree of co-operation that can be expected from manufacturers including confidentiality constraints, costs and liability, availability of information, domain expertise.

- Exploring the development of a safety case for smart sensors.

Relationship with stakeholders

- Manufacturers
  - What is required from the manufacturers and what is in this for them?
- End-users
  - What are they expected to do for different levels of assurance?
  - Co-operation within industry
- Regulators
  - Defining the expectations

Adelard

Manufacturers

- Obtained the code of smart devices by two major manufacturers.

- Level of co-operation across organisation varies according to company specific characteristics.
  - Non-disclosure agreements and modifications to these.
  - Domain experts replies.
  - Divulge the results.

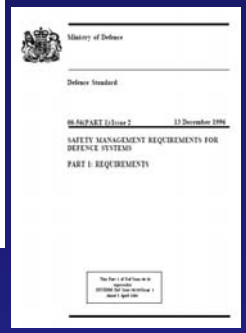
Manufacturers - long-term issues

- Concerns about effort and cost needed for routine justifications.
- In favour of an “assurance package” of additional information that is paid for by the customer and common across industry.

Adelard

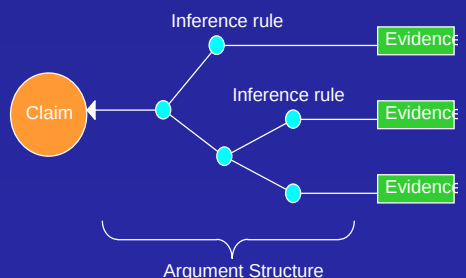
Safety justification approaches

Industry regulations and guidelines



Claim-based approach

Concerns on black-box assessment



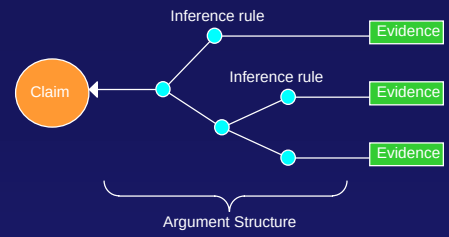
Adelard

Black-box assessment

- adequacy of functional testing
- housekeeping code
- infeasible paths
- fault detection code
- fault tolerance
- time based events
- counters
- malicious code
- unwanted requirements
- software security vulnerabilities
- complexity
- non-predictability

Adelard

Claim-based approach



- More flexible - focus on the safety requirements
- Can relate to other safety standards
- Similar to Cemsis approach

- Justification in two parts:
  - claims that the component “does what it says on the tin”
  - claims that the component is suitable for the application context

Adelard

Evidence to support claims

- Obtain an “evidence profile” to support claims in the safety justification.
  - Different types of evidence available for different sensors, e.g. process IEC 61508 compliant vs. field experience
- Where claims are not supported, it is necessary to:
  - obtain further evidence
  - address the issue in the application specific safety justification

Smart sensor claims

Ref	Sensor property
1	Measurement accuracy better than 0.1% under stated operating conditions
2	Sample rate < 0.128 secs, step response < 1.024 secs
3	Safe failure fraction >0.8
4	MTBF > 25 years
5	Configuration and calibration errors are minimised
6	Smart sensor behaviour is predictable
7	The smart sensor will be supported for another 10 years

Evidence to support the claims

Adelard

Analysis of a smart device

- ~6 kloc of assembler

- Different analyses:
  - Software criticality (SCA) and integrity static analysis
  - Software architecture
  - Supported by simulator/debugger tool on assembler source

Adelard

Software Criticality Analysis

The primary objective

- to assess the importance to safety of software components.

Secondary objectives

- traceability
- failure mode analysis
- understanding of the software structure and quality

"Software Criticality Analysis of COTS/SOUP"
P Bishop, R E Bloomfield, T Clement and S Guerra
Reliability Engineering and System Safety 81 (2003)

Adelard

Integrity static analysis

- Static analysis = Assessing software without executing it.
- Attempts to demonstrate program properties for all possible program executions.

Contrast with testing!

- Approach:
  - Internal and intra-component integrity of the code.
  - Resource locking, stack allocation, pointer and array use, uninitialised variables, other analyses

"Integrity Static Analysis of COTS/SOUP"
P Bishop, R E Bloomfield, T Clement, S Guerra and C Jones
Safecomp 2003

Adelard

SCA and static analysis

- Use of SPIN for control flow - SCA
- Perl script for dead code software anomalies
  - directly on assembler code
- Translation from assembler to compilable C
- Analysis of translated files
  - integrity analysis with Safer C
  - control flow analysis with CodeSurfer
- Software metrics using Safer C

Another device originally written in C!

Adelard

Analysis of architecture

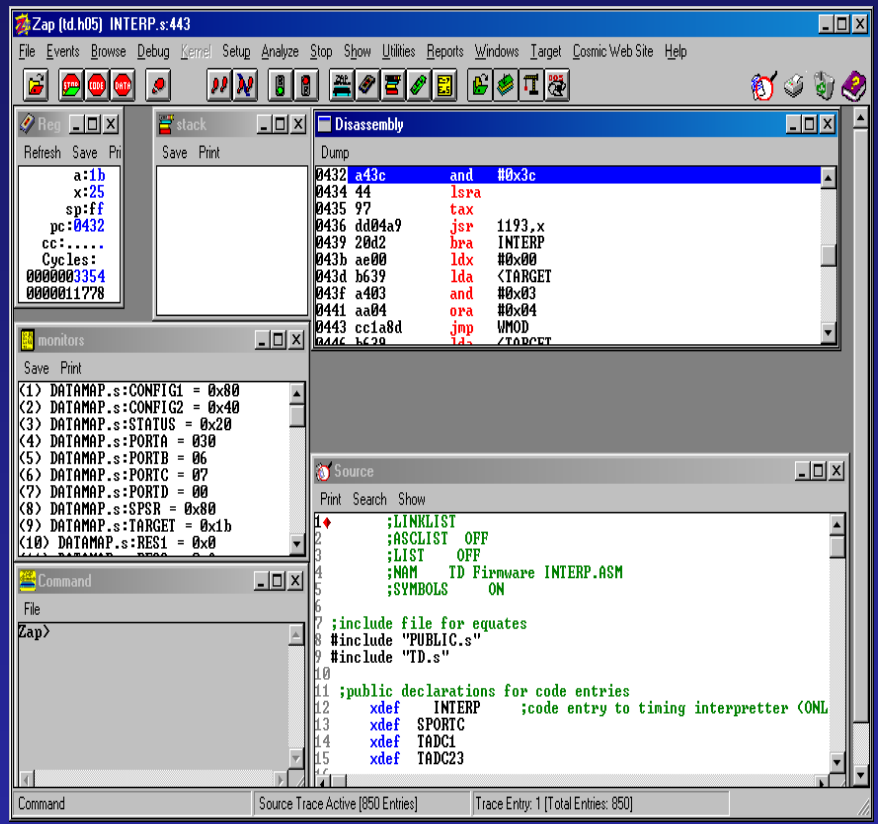
- Elicitation of software architecture to support other analyses and testing.

- Structure elicited by:
  - review of the documentation
  - inspection of the source code
- Analyses:
  - concurrency analysis: data exchange between threads
  - redundant software analysis
  - failure integrity analysis: failsafe response to smart sensor failure e.g. RAM start-up test

Adelard

Simulation

- Used COSMIC assembler and simulator/debugger tool on assembler source.
- Some format changes to source code needed to get it through the tool.
- Binary output is identical to original binary.
- Can simulate device input by reading from data file.
- Can record computed outputs to a data file.

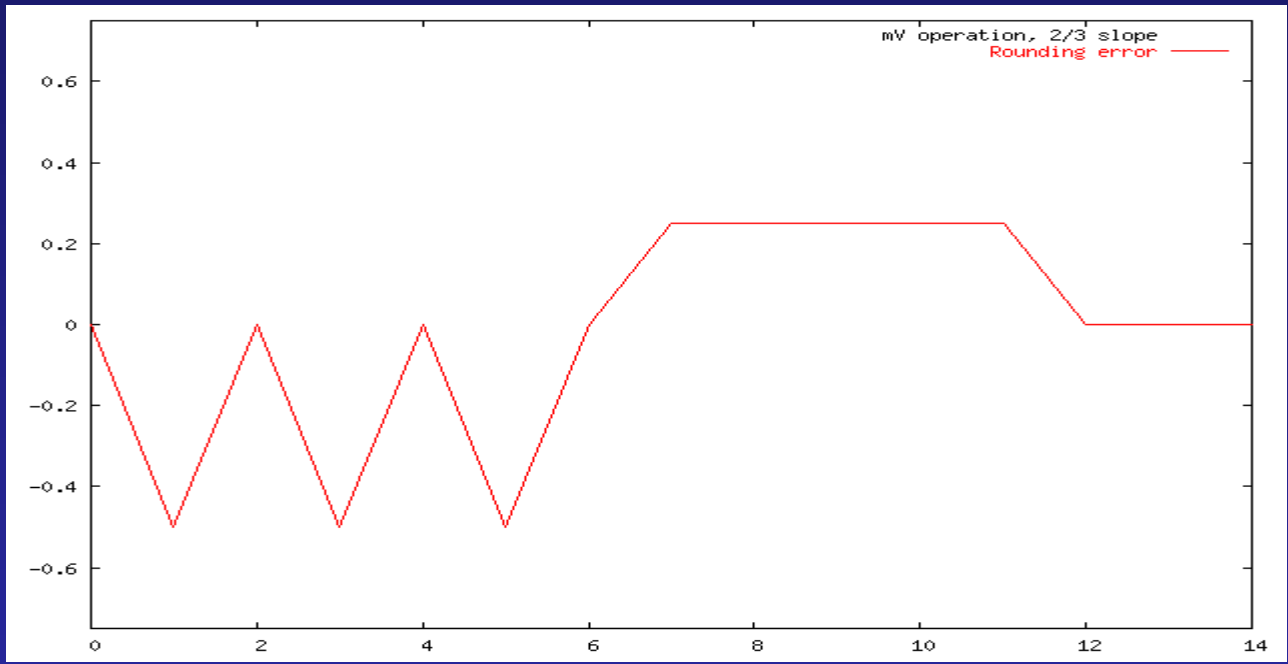


Applications

- Testing of specific functions
  - Accuracy of software conversion routines - plotted input value vs. output value to check conversion accuracy is within spec.
- Timing analysis
- Code coverage
  - can get a execution count for each line of code

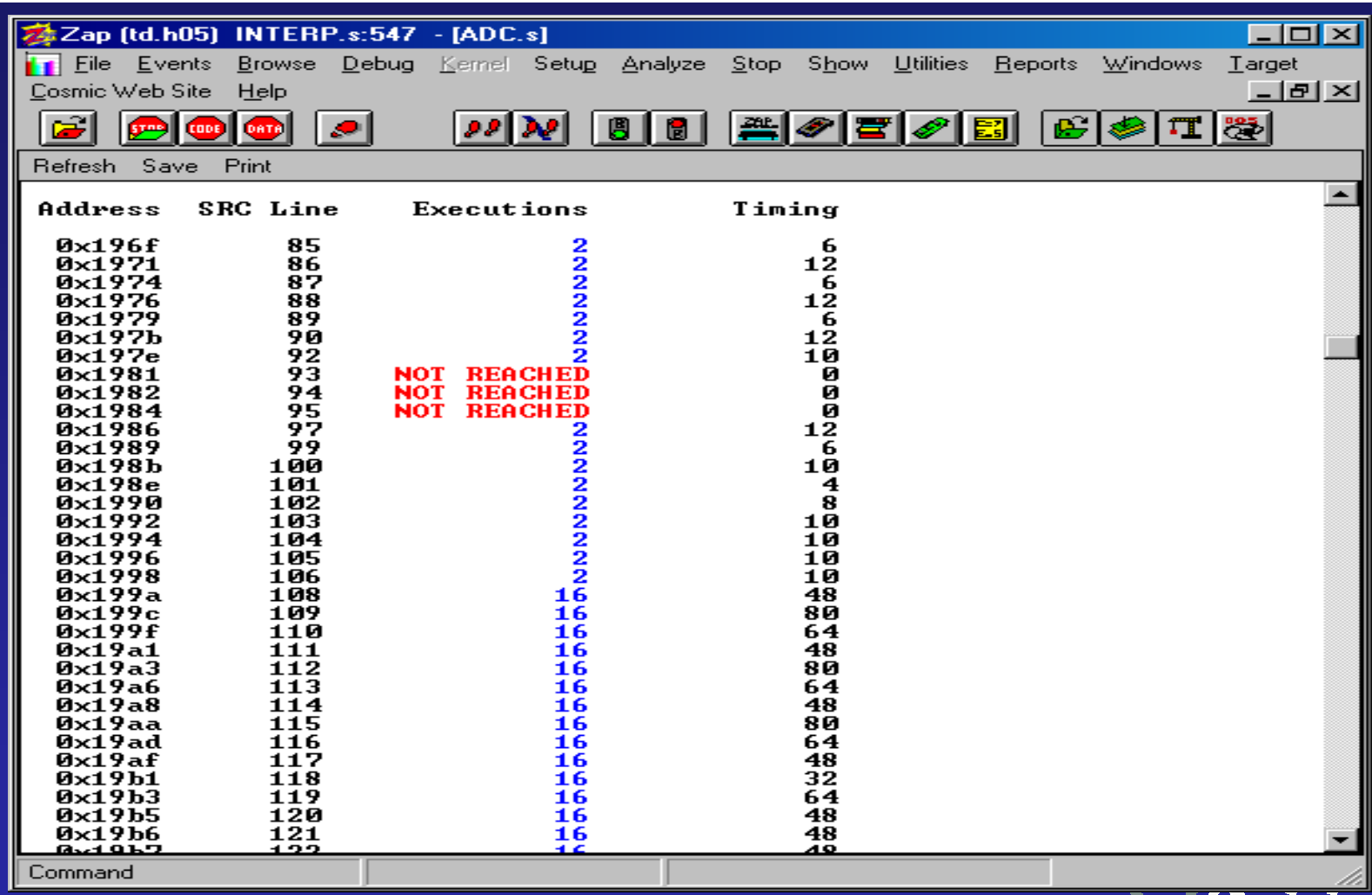
Numerical accuracy

- Output to DAC within +/-0.5 of computed value



Output range is ~4000 units => accuracy in the order of +/-0.002mA

Coverage of ADC module



Address	SRC Line	Executions	Timing
0x196f	85	2	6
0x1971	86	2	12
0x1974	87	2	6
0x1976	88	2	12
0x1979	89	2	6
0x197b	90	2	12
0x197e	92	2	10
0x1981	93	NOT REACHED	0
0x1982	94	NOT REACHED	0
0x1984	95	NOT REACHED	0
0x1986	97	2	12
0x1989	99	2	6
0x198b	100	2	10
0x198e	101	2	4
0x1990	102	2	8
0x1992	103	2	10
0x1994	104	2	10
0x1996	105	2	10
0x1998	106	2	10
0x199a	108	16	48
0x199c	109	16	80
0x199f	110	16	64
0x19a1	111	16	48
0x19a3	112	16	80
0x19a6	113	16	64
0x19a8	114	16	48
0x19aa	115	16	80
0x19ad	116	16	64
0x19af	117	16	48
0x19b1	118	16	32
0x19b3	119	16	64
0x19b5	120	16	48
0x19b6	121	16	48

Adelard

Claims and example evidence

Ref	Claim	Example Evidence
1	Measurement accuracy better than 0.1% under stated operating conditions	Independent SIREP tests. Accuracy analysis System design documentation. Analysis of maths routines.
2	95% response to step change < 0.5 secs	Customer qualification timing test. Analysis of averaging algorithm. Worst case timing analysis.
3	Fail-danger fraction <0.2	Customer product problem register Independent FMEDA report. Analysis of fault detection features in the software.
4	MTBF > 25 years	Customer sales and warranty returns Hardware reliability analysis. Software modification history.
5	Configuration errors are minimised	Human factors review of configuration interface Analysis of software design
6	The smart sensor behaviour is predictable	Software function analysis. Software architecture analysis. Concurrency analysis. Software timing analysis. Simulated execution of the software.
7	The smart sensor will be supported for another 10 years	Supplier support policy. Supplier support documentation.

Conclusions - manufacturers

- Successfully obtained genuine smart sensor software.
- Different level of co-operation across the organisations' hierarchy and different organisations.
- Manufacturers would like:
  - assurance package paid by the end-user
  - procedures accepted across industry
  - credit for using good processes, e.g. IEC 61508 certification

Conclusions - analyses

- Feasible to address assessment concerns by a combination of analysis, review and test.
- Analysis of assembler code is possible.
- Useful to have more than the code
  - access to experts
  - good design documentation supports the analysis of the code
  - other data will be needed to complete the safety case