Reliability and Safety of Embedded Systems
Correctness – Tools – Case Study
Part III: Certification Process

Jan Peleska
jp@verified.de

University of Bremen – Verified Systems International GmbH

Zuverlässigkeit und Sicherheit von Embedded Systems
2010-10-14

Objectives for Part III

- Illustrate software-related issues of the certification process by means of a practical examples
- Focus on the avionic domain where RTCA DO178B covers software-related certification issues
- Compare to certification in the railway domain, where CENELEC standards apply

Overview

- Background: overall aircraft certification process
- Certification of avionic systems according to RTCA DO178B
- Critical aspects of avionic systems certification
- Fundamental issues of the avionic systems verification process

Background: overall aircraft certification process

The overall certification process is driven by the following high-level standards

- **Air Transport Association (ATA) Chapters.** A systematic decomposition of a conceptual aircraft into aircraft systems. System descriptions induce the fundamental functions required in an aircraft

  Examples.
  - ATA-Chapter 21. Air Conditioning
  - ATA-Chapter 30. Ice and Rain Protection
  - ATA-Chapter 32. Landing Gear

  Note 1. The ATA description is "slightly outdated" from today's point of view since it already suggests a function $\leftrightarrow$ system association

  Note 2. Such a high-level description of essential system functions does not exist for railway control systems

Background: overall aircraft certification process

Technical standards such as **Radio Technical Commission for Aeronautics (RTCA)** and **Aeronautical Radio Incorporated (ARINC)** standards specify the (minimal) requirements for equipment implementing aircraft functions

Examples.

- ARINC 653: Avionics Application Software Standard Interface
- ARINC 664: Aircraft Data Network (Avionics Full Duplex Switched Ethernet (AFDX))
- RTCA DO-200A: Standards for Processing Aeronautical Data
- RTCA DO-185B: Minimum Operational Performance Standards for Traffic Alert and Collision Avoidance System II (TCAS II)
- RTCA DO-178B: Software Considerations in Airborne Systems and Equipment Certification

Background: overall aircraft certification process

- At the beginning of the certification process the applicable standards are identified
- The manufacturer's **verification and validation process** has the goal to obtain **certification credits** stating that
  - All necessary aircraft systems have been implemented
  - All systems comply with their applicable standards
  - The **development process is trustworthy**, that is,
    - The chain of development activities and supporting tools, as well as the methods used are suitable to prevent the production of erroneous airborne software
  - The **V&V process is trustworthy**, that is,
    - **traceable**: it can be confirmed by independent parties that no functions have been forgotten and that V&V covered all of these functions
    - **qualified**: it can be confirmed that the V&V methodology, procedures and tools have the necessary quality to trust its results

Background: overall aircraft certification process

Standard specialising on avionic software-based systems:

- **Radio Technical Commission for Aeronautics (RTCA).** Federal Advisory Committee – its recommendations are used by the Federal Aviation Administration (FAA)
- **RTCA DO-178B/ED-12B.** Software Considerations in Airborne Systems and Equipment Certification
- **System under test (SUT).** RTCA DO-178B applies to software in one controller or a (small) functionally cohesive network of controllers.
  Other standards apply to system integration tests and flight tests

Certification of avionic systems according to RTCA DO178B

RTCA DO178B defines the following software-related life cycle processes for avionic systems development

1. Software planning process
2. Software development process
3. Software verification process
4. Software configuration management process
5. Software quality assurance process
6. The **certification liaison process** manages the communication and understanding between certification applicant and certification authority

Certification of avionic systems according to RTCA DO178B

A specific view on **software requirements**

- **High-level software requirements** are derived from system requirements that are mapped onto software components
  - Functional software requirements as derived from system requirements
  - Freedom from negative impact on system safety
- **Low-level software requirements** are requirements from which source code can be implemented without further information
- **Derived software requirements** are requirements which are not directly traceable to higher-level requirements – typically derived from specific HW and SW design decisions

Certification of avionic systems according to RTCA DO178B

Details of the **certification liaison process**

- The software planning process applies and defines the certification liaison process
- The system-specific aspects of the certification liaison process are captured by the **plan for software aspects of certification**
- Outputs of the certification liaison process are
  - **Software accomplishment summary**
  - **Software configuration index**

Certification of avionic systems according to RTCA DO178B

Details of the **certification liaison process** . . . continued

- The main objectives of the certification liaison process are
  - Communication and understanding between the applicant and the certification authority is established
  - The means of compliance is proposed, and agreement with the plan for software aspects of certification is obtained
  - Compliance substantiation is provided

Certification of avionic systems according to RTCA DO178B

Details of the **certification liaison process** . . . continued

- **Software accomplishment summary**
  - System and software overview
  - Certification considerations
  - Software characteristics (size, timing , memory utilisation, . . .)
  - Software life cycle data
  - Additional considerations with potential impact on certification
  - Software identification
  - Change history
  - Software status (problem reports and their status, functional limitations)
  - Compliance status

Critical aspects of avionic systems certification

The following items should be addressed in the plan for software aspects of certification and approved by the certification authorities at an early stage:

- **Design considerations:**
  - Partitioning
  - Safety monitors
  - Multi-version dissimilar software
  - Field-loadable software
  - User-modifiable software
  - Option-selectable (selection by pin programming or software user interface)
  - Utilisation of Commercial Off-The-Shelf software
  - Robustness

Critical aspects of avionic systems certification

- **Tool-related considerations:**
  - Development tool qualification
  - Verification tool qualification
- **Verification considerations:**
  - Correctness of HW/SW integration
  - Effectiveness of partitioning and resulting system robustness and system safety
  - Verification of the verification results
    - Verification coverage
    - Accuracy of the verification activities
    - Traceability

Critical aspects of avionic systems certification

Tool qualification – which tools have typically to be qualified?

- Compilers
- Model-based code generators
- Libraries
- Linkers – "board support packages"
- Operating systems
- Verification tools
  - Test automation tools, including coverage analysers
  - Formal functional verification
  - Absence of run-time errors
    - WCET analysis
    - Stack analysis
    - Safe memory access (arrays, pointers,. . .)
    - . . .

Critical aspects of avionic systems certification

Principles of tool qualification

- Qualification is required if the tool automates some of the processes controlled by RTCA DO178B, **without verification of the outputs**
- Examples
  - Object code
  - Executable binary image
  - Automatically generated test data
  - Automatically generated executable test procedure
  - Automatically generated test verdict
  - Verification result of model checker or proof tool for algorithms
- Only **deterministic tools** can be qualified

Critical aspects of avionic systems certification

Principles of tool qualification . . . continued

The effort of tool qualification corresponds to the criticality of the tasks that are automated by the tool

- Software development tools (Compilers, model-based code generators, linkers, libraries):
  - may introduce errors into the airborne software
  - have to be developed according to the same criticality as the on-board software whose source and object code they generate, **if their output is not verified** according to the requirements of RTCA DO178B

Critical aspects of avionic systems certification

Principles of tool qualification . . . continued

- Operating systems have the same criticality as the most critical application that runs on the associated controller
- Verification tools cannot introduce errors, but may fail to detect them
- Verification tools whose outputs are not verified have to be shown to be consistent with respect to their operational requirements under normal operational requirements

Critical aspects of avionic systems certification

Principles of tool qualification . . . continued

- $\Rightarrow$ Tool qualification credit is only granted with respect to a specific development target (on-board software and controller hardware), because the qualification requirements depend on the target's criticality

Critical aspects of avionic systems certification

Tool qualification data

- Tool qualification plan (only for software development tools)
- Tool purpose and general description in plan for software aspects of certification
- Tool operational requirements
- Tool accomplishment summary (only for software development tools)
- Data controlled as control category 1 for software development tools and control category 2 for software verification tools

Fundamental issues of the avionic systems verification process

Test types

- **Functional testing.** Test cases are developed in order to check the implementation of high-level, low-level and derived requirements
- **Structural testing.** Test cases are developed in order to cover code structure, interfaces, data and control coupling
  - **All structural aspects must be traceable to requirements!**
- **Non-functional tests**
  - Robustness
  - Absence of runtime errors
  - Safety
  - Usability
  - ...

Fundamental issues of the avionic systems verification process

Test types . . . continued

- **Inter-operability tests** check that controllers in aircraft network communicate as specified
- **End-to-end tests** verify chains of actions involving several avionic control systems, together offering a comprehensive high-level service as specified by the ATA Chapters
- **Bare module tests** verify that HW, operating system and drivers operate correctly and fulfil their configuration requirements
- **Configured module tests** verify that HW, operating system and drivers and specific configuration provide all resources and communication means as required by the application layer and by other controllers in the network

Fundamental issues of the avionic systems verification process

Test levels

- Testing is performed on different levels:
  - System integration
  - HW/SW integration
  - SW integration
  - Unit (= SW module)
- **Higher-level tests cancel the necessity for lower-level tests**, if functional and structural and non-functional coverage is obtained already on the higher level

Fundamental issues of the avionic systems verification process

Test coverage

- DO178-B requires:
  - For all levels: 100% coverage of high-level requirements
  - For all levels: 100% coverage of low-level requirements
  - For all levels: 100% coverage of derived requirements
  - Level C: 100% Statement Coverage
  - Level B: 100% Branch Coverage
  - Level A: 100% Modified Condition/Decision Coverage
  - For Levels A,B,C: 100% coverage of control and data coupling