Australian Government
Civil Aviation Safety Authority

ADVISORY CIRCULAR
AC 21-50 v1.0

Approval of software and electronic hardware parts

Date	November 2022
File ref	D22/464706

Advisory circulars are intended to provide advice and guidance to illustrate a means, but not necessarily the only means, of complying with the Regulations, or to explain certain regulatory requirements by providing informative, interpretative and explanatory material.

Advisory circulars should always be read in conjunction with the relevant regulations.

Audience

This advisory circular (AC) applies to:

- registered operators
- persons authorised under Subpart 21.M of the Civil Aviation Safety Regulations 1998 (CASR)
- organisations authorised under Subpart 21.J of CASR.

Purpose

This AC provides guidance on the approval of non-integrated aircraft software and electronic hardware parts. For further information on integrated software and electronic hardware parts refer to Paragraph 5.1 of this AC.

For further information

For further information, contact CASA's Airworthiness Standards Branch (telephone 131 757).

Status

This version of the AC is approved by the Branch Manager Airworthiness & Engineering

Version	Date	Details
v1.1	November 2022	Administrative review only.
v1.0	May 2014	Initial AC.

Unless specified otherwise, all subregulations, regulations, Divisions, Subparts and Parts referenced in this AC are references to the *Civil Aviation Safety Regulations 1998 (CASR)*.

Contents

1 Reference material 3
1.1 Acronyms 3
1.2 Definitions 4
1.3 References 6
2 Software and electronic hardware standards 9
2.1 Software 9
2.2 Electronic hardware 9
2.3 Standards for software and electronic hardware 10
2.4 Design assurance level determination 10
3 RTCA/DO-178C 12
3.1 Overview 12
3.2 Breakdown of RTCA/DO-178C sections 12
3.3 CASA life cycle data requirements 14
3.4 RTCA/DO-178C supplements 17
3.5 COTS software 18
3.6 Reverse engineering 18
3.7 User-modifiable software 18
3.8 Media and software load control 19
3.9 Replication or duplication of approved software 19
4 RTCA/DO-254 21
4.1 Overview 21
4.2 Breakdown of RTCA/DO-254 sections 21
4.3 CASA life cycle data requirements 24
4.4 COTS graphics processors 26
4.5 Use of previously developed hardware 26
4.6 Single event upset 27
5 Software and electronic hardware approval process 28
5.1 Initial approval 28
5.2 Modification of approval 29
5.3 Approved design organisations and authorised persons 29

1 Reference material

1.1 Acronyms

The acronyms and abbreviations used in this AC are listed in the table below.

Acronym	Description
AC	Advisory Circular
ARINC	Aeronautical Radio, Incorporated
ASIC	Application-specific Integrated Circuit
ATSO	Australian Technical Standard Order
CAR	Civil Aviation Regulations 1988
CASA	Civil Aviation Safety Authority
CASR	Civil Aviation Safety Regulations 1998
CAST	Certification Authorities Software Team (of the FAA)
CD	Compact Disc
COTS	Commercial-off-the-shelf
DAL	Design Assurance Level
DD	Design Description
DO	Document (RTCA identifier)
EUROCAE	European Organisation for Civil Aviation Equipment
ETSO	European Technical Standard Order
FAA	Federal Aviation Administration
IFE	Inflight Entertainment
IOA	Instrument of Appointment
LRU	Line Replaceable Unit
MOS	Manual of Standards
PDS	Previously Developed Software
PDH	Previously Developed Hardware
PHAC	Plan for Hardware Aspects of Certification
PLD	Programmable Logic Devices
PSAC	Plan for Software Aspects of Certification
SAS	Software Accomplishment Summary
SCM	Software Configuration Management
SEU	Single Event Upset

Acronym	Description
SQA	Software Quality Assurance
TSO	Technical Standard Order
UMS	User Modifiable Software
USB	Universal Serial Bus

1.2 Definitions

Terms that have specific meaning within this AC are defined in the table below. Where definitions from the civil aviation legislation have been reproduced for ease of reference, these are identified by 'grey shading'. Should there be a discrepancy between a definition given in this AC and the civil aviation legislation, the definition in the legislation prevails.

Term	Definition
Application-specific Integrated Circuit	Integrated circuits that are developed to implement a function; including: but not limited to: gate arrays, standard cell and full custom components encompassing linear, digital and mixed mode technologies.
Aircraft Software	Software that is designed for installation as an integral part of an aeronautical product. Software used in an aeronautical product is also considered an aeronautical product in accordance with section 3 of the *Civil Aviation Act 1988*.
Commercial-off-the-shelf	Commercially available applications or hardware items sold by vendors through public catalogue listings.
Custom micro-coded component	A component that includes ASIC, Programmable Logic Devices, Field Programmable Gate Array and other similar electronic components used in the design of aircraft systems and equipment.
Electronic Hardware	Electronic hardware includes line replaceable units, circuit board assemblies, application specific integrated circuits, programmable logic devices and other similar technology. This guidance is applicable to current, new and emerging technologies. Aircraft electronic hardware is also considered an aeronautical product in accordance with section 3 of the *Civil Aviation Act 1988*.
Executable Object Code	A form of code that is directly usable by the processing unit of the target computer and is, therefore, a compiled, assembled, and linked binary image that is loaded in the target computing hardware.
Field Loadable Software	Software that is loaded without removal of the target line replaceable unit from the installation. Field Loadable Software can also refer to either executable code or data.
High-Level Requirements	Software requirements developed from analysis of system requirements, safety related requirements and system architecture.
Independence	Separation of responsibilities which ensures the accomplishment of objective evaluation. Independence is achieved when the verification activity is performed by a competent person(s) other than the developer of the item being verified.
Integrated Circuit	A circuit consisting of elements inseparably associated and formed in-situ on or within a single substrate to perform an electronic circuit function.

Term	Definition
Latent	A failure that is not detected and/or annunciated when it occurs.
Life cycle	The period of time between starting the design or modification of a hardware or software item and completing the design or modification up as far as transition to production.
Low-Level Requirements	Software requirements developed from high-level requirements, derived requirements and design requirements from which Source Code is directly implemented without further information.
Media	Devices or materials which act as a means of transferring or storing software. Examples of this are USB memory sticks, portable hard drives, CD/DVD disks or other types of memory storage devices.
Object Code	A low-level representation of the computer program not usually in a form directly usable by the target computer but in a form which includes relocation information in addition to the processor instruction information.
Partitioning	A technique for providing isolation between software or hardware components to contain and/or isolate faults.
Previously Developed Software	Software already developed for use. This encompasses a wide range of software including COTS, software developed to military standards, re-used software and software assessed to previous versions of RTCA/DO-178.
Product Service Experience	A period of time during which the hardware is operated within a known environment and during which successive failures are recorded.
Programmable Logic Device	A component that is purchased as an electronic component and altered to perform an application specific function. Examples include Programmable Array Logic components, General Array Logic components, Field Programmable Gate Array components and Erasable Programmable Logic Devices.
Reverse Engineering	Re-implementation of software code or hardware item by the study of its construction, function and performance within a particular environment.
Single event upset	A random bit flip in data that can occur in hardware. It occurs when a system, perturbed by a transient electrical event, ceases proper operation in accordance with its embedded software while suffering no apparent component or device damage.
Simple Hardware Item	A hardware item is considered simple if a comprehensive combination of deterministic tests and analyses can ensure correct functional performance under all foreseeable operating conditions with no anomalous behaviour.
Source Code	Code written in source languages, such as assembly language and/or high level language, in a machine-readable form for input to an assembler or a compiler.
Software tool	A computer program used to help develop, test, analyse, produce or modify another program or its documentation. A compiler is an example of a tool.
System safety assessment	Airworthiness standards that is based on fail safe design of systems. For further details see guidance material: FAA AC 23.1309-1E FAA AC 25.1309-1 FAA AC 27-1B (Subpart F – AC 27.1309) FAA AC 29-2C (Subpart F – AC 29.1309).

Term	Definition
Traceability	An identifiable association between software or hardware items, such as between a requirement and the source of the requirement, or between a verification method and its base requirement.
Validation	The process of determining that the requirements are correct and complete.
Verification	The evaluation of an implementation against the set of requirements, to determine that the requirements were addressed.

1.3 References

Legislation

Legislation is available on the Federal Register of Legislation website https://www.legislation.gov.au/

Document	Title
Civil Aviation Act 1988.	
Part 21 of CASR	Certification and airworthiness requirements for aircraft and parts.
Subpart 21.B of CASR	Supplemental type certificates.
Subpart 21.D	Changes to type certificates.
Subpart 21.E	Supplemental type certificates.
Subpart 21.J	Delegation option authorisation procedures.
Subpart 21.K	Approval of materials, parts, processes and appliances.
Subpart 21.M	Designs of modifications of, and repairs to, aircraft, aircraft engines, propellers and appliances.
CASR Dictionary.	
Regulation 42W of the *Civil Aviation Regulations 1988 (CAR).*	
Part 42 Manual of Standards (MOS)	

Advisory material

CASA's advisory materials are available at https://www.casa.gov.au/publications-and-resources/guidance-materials

Federal Aviation Administration (FAA) advisory material is available at http://rgl.faa.gov/Regulatory_and_Guidance_Library/rgAdvisoryCircular.nsf/MainFrame?OpenFrameset

Document	Title
AC 21-16	Approval of Materials, Parts, Processes and Appliances.
AC 21-27	Manufacturing Approval – Overview.
AC 21-601	Australian Technical Standard Order Authorisation.
CAAP 42W-2(5)	Authorised Release Certificate.
FAA AC 20-174	Development of Civil Aircraft and Systems.
FAA AC 20-115C	Airborne Software Assurance.
FAA AC 20-152	Design Assurance Guidance for Airborne Electronic Hardware.
FAA AC 23.1309-1E	System Safety Analysis and Assessment for Part 23 Airplanes.
FAA AC 25.1309-1A	System Design and Analysis.
FAA AC 27-1B	Certification of Normal Category Rotorcraft.
FAA AC 29-2C	Certification of Transport Category Rotorcraft.
FAA Order 8110.49	Software Approval Guidelines.
FAA Order 8110.105	Simple and Complex Electronic Hardware Approval Guideline.

Accepted Industry Standards

RTCA, Inc. standards are available at http://www.rtca.org/store_list.asp

EUROCAE standards are available at http://boutique.eurocae.net/catalog/

Document	Title
RTCA/DO-178C	Software Considerations in Airborne Systems and Equipment Certification. December 13, 2011.
EUROCAE/ED-12C	Software Considerations in Airborne Systems and Equipment Certification. January 2012.
RTCA/DO-254	Design Assurance Guidance for Airborne Electronic Hardware. April 19, 2000.
EUROCAE/ED-80	Design Assurance Guidance for Airborne Electronic Hardware. April 2000.

Reports

Rockwell Collins - Aeronautical Radio, Incorporated (Rockwell Collins –ARINC) reports are available at https://www.arinc.com/cf/store/index.cfm

FAA CAST position papers are available at
http://www.faa.gov/aircraft/air_cert/design_approvals/air_software/cast/cast_papers/

DOT/FAA/AR report is available at:

http://www.faa.gov/aircraft/air_cert/design_approvals/air_software/media/AR-95-31-CEH.pdf

Document	Title
Rockwell Collins – ARINC Report 665-3	Loadable Software Standards. August 2005.
Rockwell Collins – ARINC Report 667-1	Guidance for the Management of Field Loadable Software. November 2011.
DOT/FAA/AR-95/31	Design, Test, and Certification Issues for Complex Integrated Circuits.
FAA CAST Position Paper 28	Frequently Asked Questions (FAQs) on the Use of RTCA Document DO-254 and EUROCAE Document ED-80, Design Assurance Guidance for Airborne Electronic Hardware.
FAA CAST Position Paper 29	Use of COTS Graphical Processors (CGP) in Airborne Display Systems.

2 Software and electronic hardware standards

2.1 Software

2.1.1 Software is an intangible asset, having no physical presence, that is stored on a variety of media. The medium serves only as the container for the software. Software is considered to be an aeronautical product in accordance with the Civil Aviation Act 1988 (the Act).

2.1.2 Installed software is a subset of aircraft system hardware and is approved as an integral part of the parent equipment; however, software is fundamentally different to the physical components fitted to the aircraft. Software cannot be assured to the same reliability level as physical parts by continual testing. Physical parts can be tested to breaking point to ensure design and manufacturing flaws are not present, whereas mean time between failures and programmed replacements do not apply to software parts.

2.1.3 Software will only fail if there is a latent error, virus, design error or a single event upset (SEU). A software design error can exist for years without manifesting itself or causing a failure condition. The goal is to build quality into software by assuring the development and verification processes.

2.1.4 Aircraft software must meet the accepted design assurance standard rather than a software development standard. Software design assurance is aimed at ensuring that the software will perform its intended function[1] with a level of confidence similar to a physical part. RTCA/DO-178C is used to ensure that software meets design assurance requirements.

2.2 Electronic hardware

2.2.1 Aircraft equipment may contain programmable logic devices (PLDs). The functions of PLDs are determined by the embedded code configured following manufacture. PLDs installed in equipment are defined as electronic hardware and are considered an aeronautical product in accordance with the Act.

2.2.2 For the purpose of this AC, electronic hardware can include items such as:

- field programmable gate arrays
- PLDs
- programmable array logic devices
- graphics processors
- circuit card assemblies
- application-specific integrated circuits (ASICs).

2.2.3 This AC does not cover items such as:

- discrete components:
- resistors

[1] Functional requirements for software are identified during the aircraft system safety assessment.

- capacitors
- transistors
- diodes or other similar components
- analogue integrated circuits:
- power regulators
- amplifiers
- filters
- modulators
- mixers or other similar components
- digital or hybrid integrated circuits that do not contain embedded code:
- logic gates
- analogue-to-digital convertors
- digital-to-analogue convertors and other similar components.

2.2.4 Hardware design assurance is aimed at ensuring that the electronic hardware with embedded code, will perform its intended function with a level of design assurance. Conformance to RTCA/DO-254 ensures that electronic hardware meets aircraft system safety assessments.

2.3 Standards for software and electronic hardware

2.3.1 All applications for approval of software/electronic hardware in aircraft systems must be granted by CASA unless an engineering speciality has been granted under Subpart 21.M or Subpart 21.J.

2.3.2 RTCA/DO-178C and EUROCAE/ED-12C were released in 2012 and are technically equivalent documents.

2.3.3 RTCA/DO-254 and EUROCAE/ED-80 were released in 2000 and are technically equivalent documents.

2.3.4 RTCA/DO-178C is backwards compatible with RTCA/DO-178B. A summary of their differences is included in Paragraph 3 of Appendix A in RTCA/DO-178C. RTCA/DO-178C and EUROCAE/ED-12C were jointly released to address ambiguities in RTCA/DO-178B that could lead to an unacceptable means of compliance if interpreted incorrectly.

2.3.5 Software standards such as MIL-STD-498, MIL-STD-2167A, IEEE/EIA-12207, IEC 61508, and U.K. Defense Standard 0-55, deal with certain aspects of software development covered by RTCA/DO-178C. These software standards do not provide complete coverage of RTCA/DO-178C design assurance objectives and fail to address required system safety assessment.

2.3.6 Compliance with RTCA/DO-178C and RTCA/DO-254 is recommended and strongly encouraged by CASA and in some cases it will mandatory.

2.4 Design assurance level determination

2.4.1 A system safety assessment is required for aircraft equipment with software and/or electronic hardware to determine the software design assurance level (DAL) or

hardware DALs (see Figure 1). CASA should be consulted before any projects are commenced. The DAL may already be defined for critical equipment e.g., in a Technical Standard Order (TSO), ATSO or ETSO.

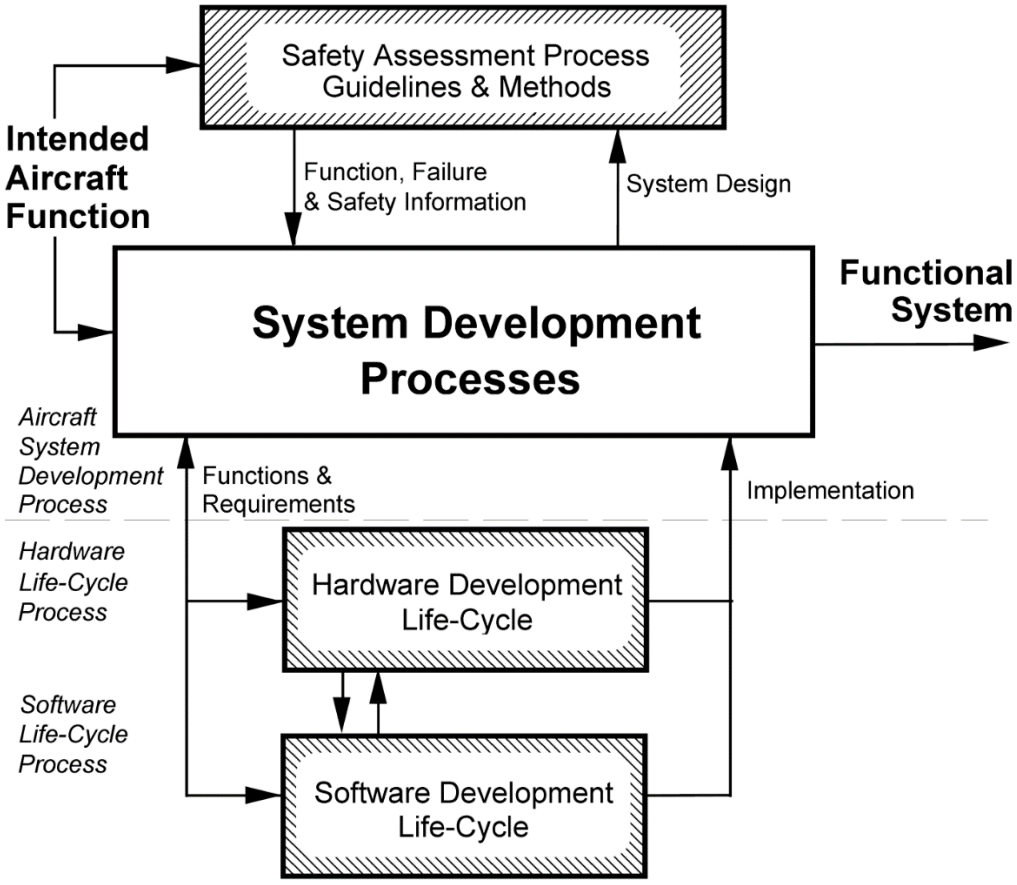

Figure 1 – Certification Guidance Documents

3 RTCA/DO-178C

3.1 Overview

3.1.1 Applicants are required to satisfy all applicable objectives listed in Annex A of RTCA/DO-178C, depending on the software DAL. There are five software DALs in RTCA/DO-178C[2]—DAL A requires the largest amount of software life cycle data and DAL E only requires CASA agreement.

3.2 Breakdown of RTCA/DO-178C sections

3.2.1 The relationship between the various processes in RTCA/DO-178C is illustrated in Figure 2. The software development process provides verification criteria and requirements for the integral process. The integral process provides feedback in the form of validation results that are used to refine the Software Development Process.

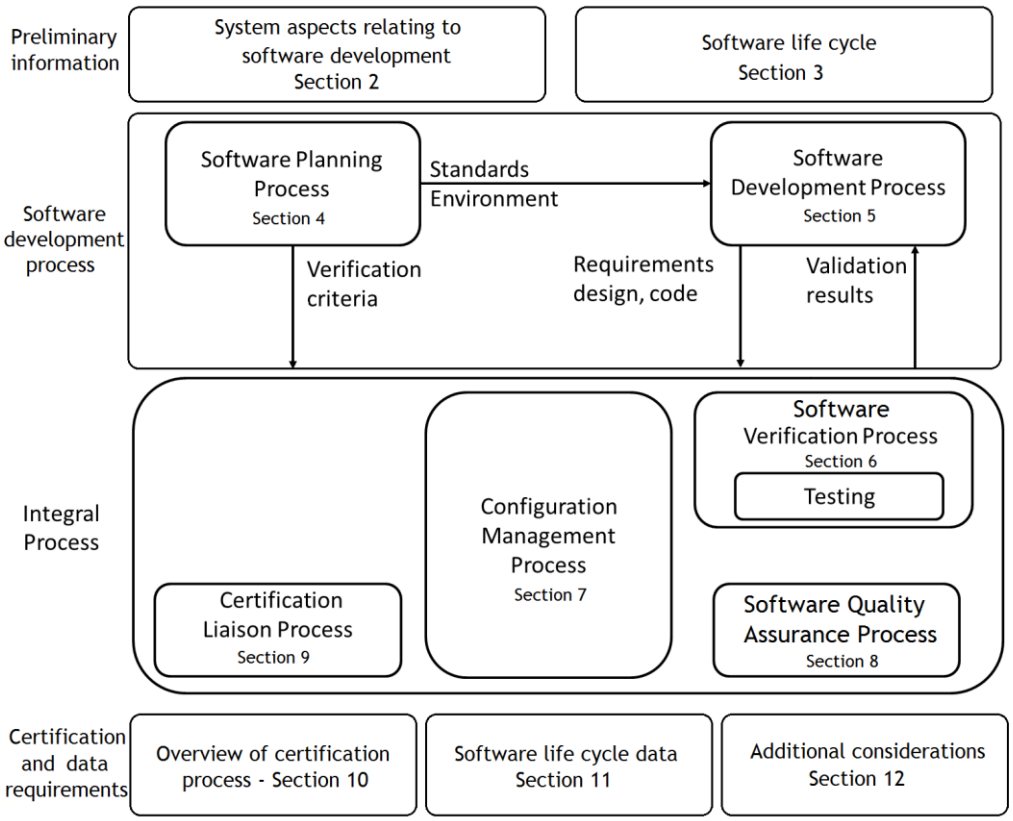

Figure 2 – RTCA/DO-178C Software Life Cycle

3.2.2 Preliminary information

3.2.2.1 Preliminary information includes:

– **System Aspects Relating to Software Development – Section 2.** This section discusses those aspects of the system life cycle processes necessary to

[2] Section 2.3.3 of RTCA/DO-178C details the requirements for each DAL.

understand the software life cycle processes. The software DAL and options for software are detailed in this section.

- **Software Life Cycle – Section 3.** This section discusses the software life cycle processes, software life cycle definition and transition criteria between software life cycle processes.

3.2.3 Software development process

3.2.3.1 Software life cycle processes include:

- **Software Planning Process – Section 4.** The Software Planning Process provides verification criteria for the integral process. This process produces the software plans and standards that direct the software development and integral processes. Table A-1 in RTCA/DO-178C summarises the objectives and outputs of the software planning process by DAL.
- **Software Development Process – Section 5.** The Software Development Process provides requirements to the Integral Process. The development of software architecture involves decisions made about the structure of the software.
  - Software Requirements Process – Section 5.1: Functional, performance, interface, and safety-related requirements are used to develop high-level requirements.
  - Software Design Process – Section 5.2: High-level requirements are used to develop the software architecture and the low-level requirements.
  - Software Coding Process – Section 5.3: The low-level requirements are used to develop the source code.
  - Integration Process – Section 5.4: The target computer and source code are used to compile, link and load data in the integral process to develop the integrated system or equipment.

3.2.4 Integral process

3.2.4.1 Integral processes are separated into:

- **Software Verification Process – Section 6.** Verification is a technical assessment of the outputs of the software planning process, software development processes and the software verification process.
- **Software Configuration Management (SCM) Process – Section 7.** Outputs of the SCM process are recorded in SCM records or in other software life cycle data.
- **Software Quality Assurance Process – Section 8.** This process assesses the software life cycle and its outputs to obtain assurance that:
  - objectives were satisfied
  - deficiencies were detected, evaluated, tracked and resolved
  - software product and software life cycle data conformed to certification requirements.
- **Certification Liaison Process – Section 9.** The liaison process establishes communication and understanding, gains agreement on the means of compliance through approval of the PSAC and provides compliance substantiation.

3.2.5 Certification and data requirements

3.2.5.1 Certification and data requirements sections include:

- **Overview of Certification Process – Section 10**. Software is certified as an integral part of an aircraft's airborne systems. Systems and equipment, including embedded software, should be 'approved' in order to be accepted as part of airborne system or equipment certification.
- **Software Life Cycle Data – Section 11.** Describes data produced during the software life cycle to plan, direct, explain, define, record or provide evidence of activities.
- **Additional Considerations –** Section 12. Includes guidance for the use of previously developed software, software tool qualification and the use of alternative methods for compliance.

3.3 CASA life cycle data requirements

3.3.1 CASA may request additional life cycle data for a project depending on the:

- DAL
- level of integration with aircraft systems
- applicant's experience with RTCA/DO-178C.

3.3.2 Minimum requirements

3.3.2.1 The following items comprise the minimum software life cycle data that should be presented to CASA for approval:

- PSAC
- SCI
- SAS
- Software Quality Assurance (SQA) Records.

Plan for software aspects of certification

3.3.2.2 The PSAC describes the software and is the primary means used by CASA to determine if the amount of rigor in the application matches the DAL. The PSAC and the other software planning process data (detailed in Section 4 of RTCA/DO-178C) must be submitted with the project application for CASA's approval.

3.3.2.3 At a minimum, the PSAC should contain the following information:

- system overview – hardware details and specification
- software overview, with emphasis on the proposed safety and partitioning concepts
- certification considerations, including proposed software level and justification
- software life cycle – how the objectives of each software life cycle are satisfied
- software life cycle data, to specify what data is produced and controlled
- schedule, including sufficient time for CASA to review the application
- additional considerations that may affect certification
- supplier oversight – details of supplier's compliance with software plans and standards

- details of any supplements used.

Software configuration index

3.3.2.4 The SCI serves as an overall account of the content of the final software product and is maintained through the configuration management process. The SCI requires CASA approval.

3.3.2.5 The SCI should identify:

- software product(s)
- executable object code and parameter data item files, if any
- each source code component
- previously developed software (PDS) in the software product, if used
- software life cycle data
- archive and release media
- instructions for building the executable object code and parameter data item files
- reference to the software life cycle environment configuration index
- data integrity checks for the executable object code, if used
- procedures, methods, and tools for making modifications to the user-modifiable software, if any
- procedures and methods for loading the software into the target hardware.

Software accomplishment summary

3.3.2.6 The primary purpose of the SAS is to show compliance with the plans and processes detailed in the PSAC. The SAS demonstrates that the objectives as set out in the planning documents were achieved and is submitted to CASA at the conclusion of a software development project.

3.3.2.7 An applicant can copy data directly from the PSAC to the SAS if there are no differences from the planning phase of development. Figure 3 shows the transition to an SAS in which the System Overview, Software Overview and Software Life Cycle have been revised from the original PSAC and new activities introduced.

Software Quality Assurance Records

3.3.2.8 The primary purpose of the SQA Records is to record activities from the SQA processes. This can include:

- reviews or audit reports
- meeting minutes
- records of authorised process deviations
- software conformity review records

PSAC		*SAS*
System Overview	→	System Overview (revised)
Software Overview	→	Software Overview (revised)
Certification Considerations	→	Certification Considerations
Software Life Cycle	→	Software Life Cycle (revised)
Software Life Cycle Data	→	Software Life Cycle Data
Schedule		
Additional Considerations	→	Additional Considerations
		Supplier Oversight (new)
		Software Identification (new)
		Software Characteristics (new)
		Software Status (new)
		Change History (new)
		Compliance Statement (new)

Figure 3 – PSAC and SAS differences

3.3.2.9 The additional data required for the SAS are:

- supplier oversight, used to describe how supplier processes and outputs comply with plan
- software identification by part number—CASA will use this part number on the software approval instrument
- software characteristics, such as:
  - the size of executable object code
  - timing and memory margins
  - resource limitations
- description of how each of these characteristics are measured
- change history – summary of all software changes
- software status – summary of any unresolved problems and potential impacts
- compliance statement – summary of methods used to demonstrate compliance
- a matrix that maps objectives and references to compliance, to ensure that all evidence needed to demonstrate compliance is provided.

3.3.3 Additional requirements for Type Certificate projects

3.3.3.1 The use of RTCA/DO-178C is integral to Type Certificates (Subparts 21.B and 21.D) and Supplemental Type Certificates (Subpart 21.E) that include software in their design. Software used in projects related to Supplemental Type Certificates and Type Certificates require additional life cycle data for substantiation, including:

- software requirements data (i.e., a definition of the high-level requirements including the derived requirements)
- design description (i.e., a definition of the software architecture and the low-level requirements that will satisfy the high-level requirements)

- source code (i.e., code written in source language(s))
- executable object code and parameter data item files, if any.[3]

3.4 RTCA/DO-178C supplements

3.4.1 RTCA/DO-178C recognised that new software development methodologies may result in new issues. Rather than constantly expanding the standard to account for all current software development methodologies, RTCA/DO-178C acknowledged that one or more technology supplements may be used in conjunction with RTCA/DO-178C to modify the guidance for specific technologies or methodologies.

3.4.2 The PSAC should reference any supplements used and specify how the objectives of RTCA/DO-178C are modified by their usage. Supplements allow developers to use:

- software tools to generate code, verify code or detect errors in code
- model-based software such as MATLAB or Simulink
- object-orientated languages such as C++, Java or Adacore.

3.4.3 The supplements to RTCA/DO-178C and RTCA/DO-278A, as shown in Figure 4 are:

- RTCA/DO-330—Software tool qualification considerations
- RTCA/DO-331—Model-based development and verification
- RTCA/DO-332—Object-oriented technology and related techniques
- RTCA/DO-333—Formal methods

Note: Tools used in RTCA/DO-178C will require qualification. Qualification of tools is required so that their usage does not introduce errors in coding.

3.4.4 RTCA/DO-248C supplies information common to both RTCA/DO-178C and 278A. It contains Frequently Asked Questions (FAQ) and discussion papers.

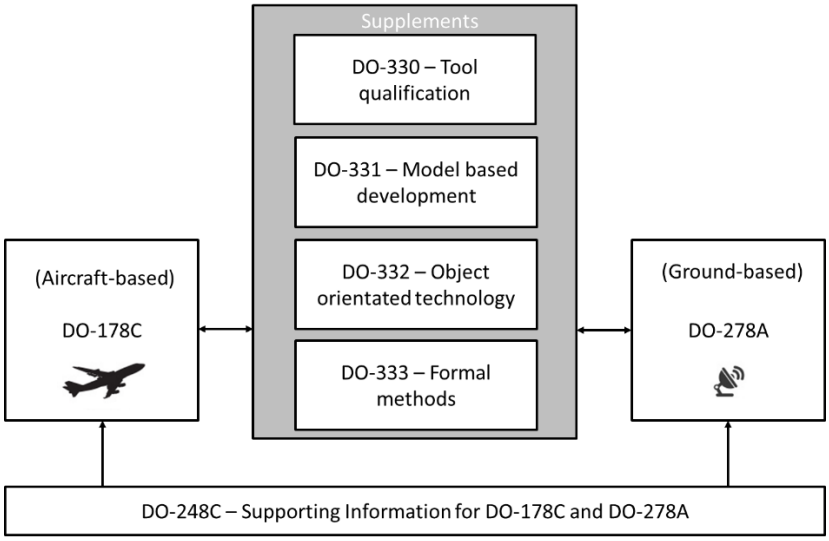

Figure 4 – RTCA/DO-178C supplements

[3] Executable object code is code directly loaded into the target LRU and parameter data item files consist of data that is directly usable by the processing unit of the target LRU. Parameter data item files influence the behaviour of executable object code but do not change the core program (e.g., a configuration file).

3.5 COTS software

3.5.1 COTS typically use proprietary, closed source software and are classified as PDS. It has usually been developed for non-aerospace or non-safety-critical purposes.

3.5.2 The use of closed source software makes traceability difficult, if not impossible, and may cause difficulty with software substantiation. If PDS is used, the applicant must ensure that it satisfies the objectives of RTCA/DO-178C.

3.5.3 CASA may accept the use of COTS in projects that are proven not to interfere with the safety of airworthiness systems (e.g., in-flight entertainment).

3.5.4 Proper application of DAL D objectives permits the use of COTS software that has been developed using military standards. Annex A to RTCA/DO-178C lists objectives for each DAL—there are only 26 objectives required for DAL D, as opposed to 62 required for DAL C. Applicants should carefully assess the use of COTS software on the basis of the objectives and activities detailed in Annex A to RTCA/DO-178C.

3.5.5 CASA's approval of COTS software is given on a case-by-case basis.

3.6 Reverse engineering

3.6.1 Except in well justified cases, CASA strongly discourages the practice of reverse engineering in new software development, as it may not address the traceability objectives of RTCA/DO-178C. CASA prefers that an applicant use custom-built software rather than reverse engineering of software.

3.6.2 Acceptable use of generating data may include:

- developing source code from the object code or executable code
- developing high-level requirements from low-level requirements.

3.6.3 If an applicant does reverse engineer software, they will be required to substantiate its use in the PSAC. CASA may also impose additional requirements on an applicant in order to satisfy the objectives of RTCA/DO-178C.

3.7 User-modifiable software

3.7.1 User-modifiable software (UMS) is customisable by the end-user without additional approval. UMS is designed within the constraints of the original software approval design. Any potential issues with the use of UMS should be identified via the system safety assessment conducted during the initial design stage.

3.7.2 Any UMS must not adversely affect:

- safety
- operational capabilities
- flight crew workload
- any non-modifiable components
- any software protection mechanisms used.

3.8 Media and software load control

3.8.1 Software is intangible by its design and does not fit the traditional definition of a part. Approval is not required for the media (e.g., USB memory stick, hard drive, disk or other memory device) used to contain the software prior to installation on the aircraft or aeronautical product. Duplication of approved software is not considered to be manufacturing.

3.8.2 CASA gives approval for the software to be installed on the target LRU and such approval is strictly related to the target LRU on a particular type of aircraft or aeronautical product.

3.8.3 Every software design project must include detail on how software load control is accomplished.[4] Methods of software load control may include:

- installation of factory pre-programmed memory devices
- *in situ* re-programming of the system

or

- equipment using a field loading device.

3.8.4 For *in situ re*-programming, any duplicated copies of the software must be verified against the master copy. Any copies that vary from the master copy should be considered unserviceable. Instructions of how *in situ* re-programming is performed must be documented.[5]

3.9 Replication or duplication of approved software

3.9.1 A registered operator may receive a single copy of an item of software which is uniquely identified by part number and is required to be distributed across the registered operator's fleet as field loadable software.

3.9.2 This replication is not considered to be manufacture of software. The software has been developed, packaged and approved by the vendors own processes and does not require further approvals provided no changes are made.

3.9.3 Suitable processes need to be in place to ensure that:

- the application and version that will be used for the replication is identified
- the proposed error checking methodology proposed to confirm the integrity of the copies being made is identified
- the individual copies are uniquely identified to facilitate the distribution and configuration management of the copies

Depending on the nature and criticality of the software, security may also need to be addressed.

3.9.4 ARINC Report 667-1 provides guidance that an Operator may use as a basis for the development of internal processes.

[4] See 7.4 in RTCA/DO-178C
[5] For further information on software integrity see ARINC Report 665-3.

3.9.5 In summary, registered operators intending to replicate approved software delivered by OEM/approved vendors are not manufacturing the software but need to have processes in place to control replication and distribution.

4 RTCA/DO-254

4.1 Overview

4.1.1 RTCA/DO-254 provides acceptable guidance for applicants seeking approval of electronic hardware for aircraft systems. The layout and intent of RTCA/DO-254 is similar to that of RTCA/DO-178C for software design assurance.

4.1.2 Applicants are required to satisfy all applicable objectives for each DAL listed in Appendix A of RTCA/DO-254. There are five levels of hardware DAL in RTCA/DO-254— DAL A requires the largest amount of hardware life cycle data and DAL E only requires CASA agreement. Section 2.3 of RTCA/DO-254 details how the DAL is established.

4.2 Breakdown of RTCA/DO-254 sections

4.2.1 The relationship between the various processes in RTCA/DO-254 is shown in Figure 5. There are 3 main processes:

- hardware planning process (Section 4)
- hardware design process (Section 5)
- supporting process (Sections 6 to 9).

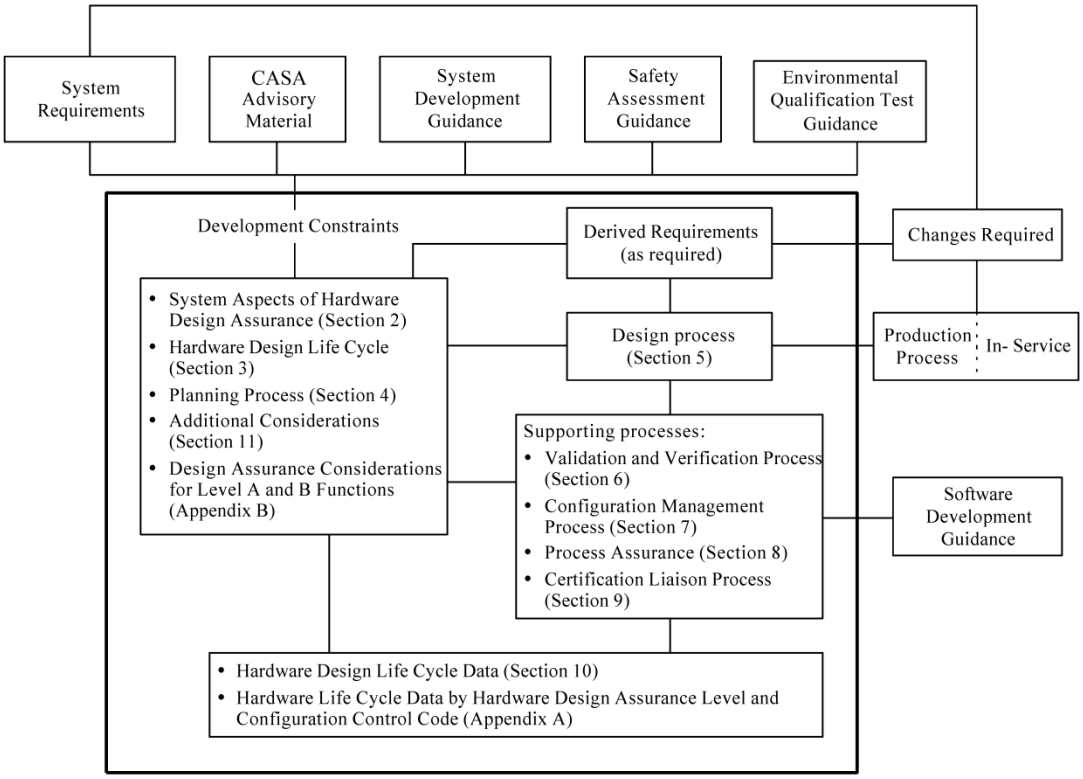

Figure 5 - Overview of RTCA/DO-254

4.2.2 Preliminary information

4.2.2.1 Preliminary information includes:

- **System Aspects of Hardware Design Assurance – Section 2.** This section discusses those aspects of the system life cycle processes necessary to understand the hardware life cycle processes. The hardware DAL and options for hardware are detailed in this section. CASA recommends that all new applicants read this section as it contains important background information about electronic hardware development processes in the context of system processes.
- **Hardware Design Life Cycle – Section 3.** This section discusses the hardware life cycle processes and the criteria for transition between hardware life cycle processes.

4.2.3 Hardware planning and design processes

4.2.3.1 The hardware planning and design processes comprise:

- **Planning Process – Section 4.** This section describes the hardware planning process used to control the development of the hardware item. The planning process is further expanded by objectives and activities.
- **Design Process – Section 5.** The hardware design processes produce a hardware item that fulfils the requirements allocated to hardware from the system requirements.
- As shown in Figure 6, the design process is broken down into the following subsections:
- **Requirements Capture Process – Subsection 5.1.** This process is used to identify and record hardware items. The process may be iterative if additional requirements are identified during design.
- **Conceptual Design Process – Subsection 5.2.** This process produces a high-level design concept. The concept may be produced using functional block diagrams, design and architectural descriptions, circuit card assembly descriptions, circuit card assembly outlines and chassis sketches.
- **Detailed Design Process – Subsection 5.3.** This process produces detailed design data using hardware item requirements and conceptual design data from Subsections 5.1 and 5.2. Any derived requirements are fed back into to the Conceptual Design Process for refinement.
- **Implementation Process – Subsection 5.4.** The implementation process uses the detailed design data from Subsection 5.3 to produce the hardware item that is an input to the testing activity.
- **Production Transition Process – Subsection 5.5.** The production transition process uses the outputs from the implementation process (Subsection 5.4) and verification process to move the product into production.
- **Acceptance Test – Subsection 5.6.** An acceptance test demonstrates that the manufactured, modified or repaired product performs in compliance with the attributes of the unit on which certification is based.

– **Series Production – Subsection 5.7.** Series production is outside of the scope of RTCA/DO-254, but considerations include the management of change or production processes and updating of all documentation related to changes.

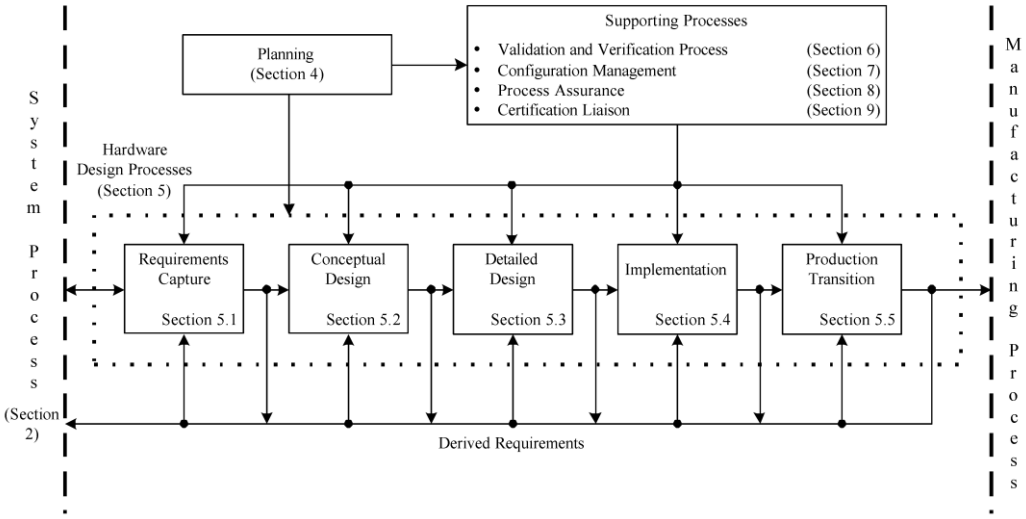

Figure 6 – Components of the hardware planning process

4.2.4 Supporting Processes

4.2.4.1 Supporting processes include:

– **Validation and Verification Process – Section 6.** Validation is the assurance that any hardware item-derived requirements are correct and complete with respect to the system requirements allocated to the hardware item. Verification provides assurance that the hardware item implementation meets all the of the hardware requirements, including any derived requirements.

– **Configuration Management Process – Section 7.** The configuration management process provides the ability to:
  - o consistently replicate the configuration item
  - o regenerate the information (if necessary)
  - o modify the configuration item in a controlled fashion, if modification is necessary and allowable.

  There are 2 types of data control categories for life cycle data, termed Hardware Control Category 1 and 2 (HC1 & HC2). HC1 requires all configuration management activities to be performed for a particular life cycle data objective. HC1 applies to hardware requirements for all DALs; HC2 applies to hardware review and analysis for all DALs.

– **Process Assurance – Section 8.** Process assurance ensures that the life cycle process objectives are met, and activities have been completed as outlined in the plans, or that any deviations have been addressed. Process assurance activities sometimes require separation of responsibility from the design process in order to objectively assess the life cycle process, identify deviations and ensure corrective

action. Separation of responsibility is termed independence and an independent person is anyone who is not the designer but can understand the processes.

- **Certification Liaison Process – Section 9.** The liaison process establishes communication and understanding between the applicant and CASA. The plan for hardware aspects of certification (PHAC) provides a means to prove hardware design assurance when the applicant seeks CASA's approval.

4.2.5 Supporting information

4.2.5.1 Supporting information includes:

- **Hardware Design Life Cycle Data – Section 10.** This section describes the hardware design life cycle data items that may be produced for providing evidence of design assurance in order to achieve approval.
- **Additional Considerations – Section 11**. This section details:
  - alternative means of compliance
  - use for previously developed hardware
  - COTS component usage
  - product service experience
  - tool qualification
  - tool assessment.

4.3 CASA life cycle data requirements

4.3.1 CASA may request additional life cycle data for a project depending on the:

- DAL
- level of integration with aircraft systems
- applicant's experience with RTCA/DO-254.

4.3.2 Minimum requirements

4.3.2.1 The following items comprise the minimum hardware life cycle data that should be presented to CASA for approval:

- PHAC
- hardware verification plan (HVP)
- top-level drawings
- hardware accomplishment summary (HAS)

Plan for hardware aspects of certification

4.3.2.2 The PHAC defines the processes, procedures, methods and standards used to achieve the life cycle data objectives of RTCA/DO-254.

4.3.2.3 The PHAC and other hardware planning process data (as detailed in Section 4 of RTCA/DO-254) should be submitted with the application for CASA's approval.

4.3.2.4 CASA will review the PHAC and associated data and notify the applicant of the outcome along with any potential impacts. The approval of a PHAC by CASA represents the start of an agreement between the applicant and CASA.

4.3.2.5 At a minimum, the PHAC should contain the following information:

- system overview – hardware details and specification
- hardware overview, with emphasis on the proposed fail-safe and partitioning concepts
- certification considerations, including proposed hardware assurance level and justification
- hardware design life cycle – procedures, methods and standards to meet objectives
- hardware design life cycle data, to specify what data is developed and submitted
- additional considerations, including applicable reused data, COTS usage and tool qualification
- alternative methods proposed for use in lieu of RTCA/DO-254 and any potential impacts
- certification schedule, including details of milestones and allowing sufficient time for CASA to review the application.

Hardware verification plan

4.3.2.6 The HVP describes the procedures, processes, activities, methods and standards to be applied to hardware in order meet RTCA/DO-254. The HVP may be included in the PHAC. It should include:

- verification methods, including any COTS and unused functions
- verification data – identification and description of the evidence produced for verification
- verification independence – the means how independence is assured for certain life cycle data
- verification environment – list of analysis tools, verification tools, test equipment and test setup diagrams
- organisational responsibilities – identification of the organisation responsible for implementing verification.

Top-level drawings

4.3.2.7 Top-level drawings uniquely identify the hardware item and all assemblies, subassemblies, components and relevant documentation.[6]

Hardware accomplishment summary

4.3.2.8 The HAS is the primary means to show compliance with the PHAC and demonstrate to CASA that the objectives of RTCA/DO-254 have been achieved.

4.3.2.9 An applicant can copy data directly from the PHAC to the HAS if there are no differences from the planning phase of development. Figure 7 show the transition to an HAS in which the System Overview, Hardware Overview and Hardware Life Cycle have been revised from the original PHAC and new activities introduced.

[6] For further information on top-level drawings see FAA Certification Authorities Software Team (CAST) Position Paper 28.

4.3.2.10 In addition to the PHAC, the following 4 activities need to be addressed for CASA approval:

- hardware Identification by part number: CASA will use this part number on the hardware approval instrument
- change history: a summary of hardware changes
- hardware status: summary of unresolved problems and an analysis of potential impacts
- compliance statement: statement of compliance with agreed RTCA/DO-254 design assurance objectives.

PHAC		*HAS*
System Overview	→	System Overview (revised)
Hardware Overview	→	Hardware Overview (revised)
Certification Considerations	→	Certification Considerations
Hardware Design Life Cycle	→	Hardware Design Life Cycle (revised)
Hardware Design Life Cycle Data	→	Hardware Design Life Cycle Data (revised)
Additional Considerations	→	Additional Considerations
Alternative Methods	→	Alternative Methods
Certification Schedule		
		Hardware Identification (new)
		Change History (new)
		Hardware Status (new)
		Compliance Statement (new)

Figure 7 - PHAC and HAS differences

4.4 COTS graphics processors

4.4.1 COTS graphics processors were originally designed for the non-aerospace market. They are therefore not certified to accepted standards, such as FAA TSO-C113a or ETSO-C113a, that are required for multipurpose electronic displays such as electronic flight instrument systems.

4.4.2 The use of COTS graphics processors for RTCA/DO-254 verification activities is problematic and most likely impractical. COTS graphics processors have rapid life cycles, allowing design errors or variations to occur during production life. COTS graphics processors may also contain software drivers or libraries that do not meet the design assurance objectives of RTCA/DO-178C.[7]

4.5 Use of previously developed hardware

4.5.1 Section 11.1 of RTCA/DO-254 details the use of PDH. There are additional configuration management considerations, including traceability and change control processes for the hardware.

[7] For further information on use of COTS Graphics Processors see FAA CAST Position Paper 29.

4.5.2 An applicant should state their intention to use PDH in the PHAC; the PDH must still meet the objectives of RTCA/DO-254.

4.5.3 RTCA/DO-254 defines four scenarios when it may be acceptable to use PDH:

- modifications to PDH
- change of aircraft installation
- change of application or design environment
- upgrading a design baseline.

4.5.4 The nature of the PDH implementation may influence the requirement to modify life cycle data and possibly increase the DAL. No additional effort will be required if the DAL is the same or less stringent for new aircraft installations than previous installations.

4.6 Single event upset

4.6.1 Charged particles can travel through hardware devices and alter the logic state of any electronic hardware device, resulting in an SEU. Figure 8 illustrates how a charged particle can have an adverse effect as it travels through an electronic hardware device.

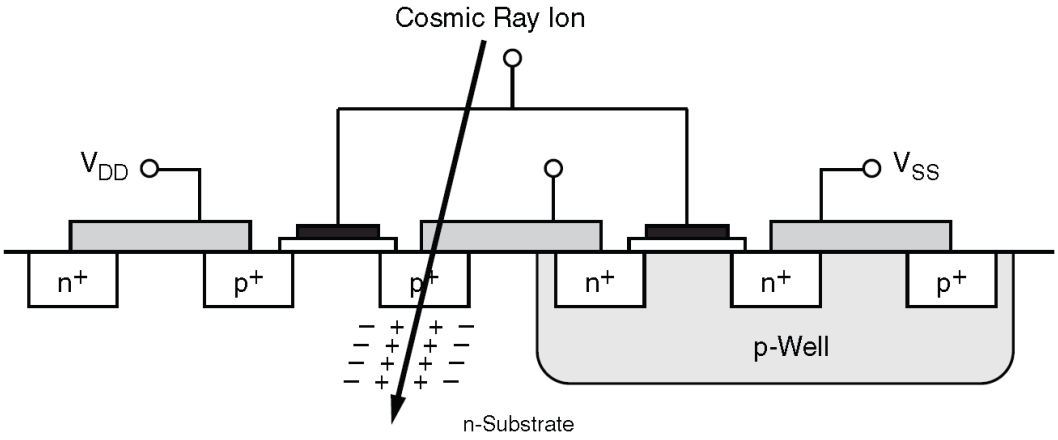

Figure 8 – Example of a single event upset

4.6.2 Such events are likely to become more of an issue as devices increase in complexity and density. Studies have found that SEU occurrence doubles at mid or high altitudes and polar latitudes. Any projects that are flight critical (DAL A and B) must take into account SEU.

4.6.3 Techniques that have proven effective in minimising the effects of SEU include[8]:

- use of radiation-hardened components, which are heavier, larger, and more expensive
- use of error detection and correction techniques
- incorporation of device redundancy
- shielding of devices.

[8] DOT/FAA/AR-95/31 has further information on SEU and RTCA/DO-254 issues.

5 Software and electronic hardware approval process

5.1 Initial approval

5.1.1 Approval of software and electronic hardware in Australia has been mainly limited to approval of non-integrated applications. This AC provides guidance on software and electronic hardware approvals. For systems beyond the scope of this AC, the following guidance documents are accepted:

- FAA Order 8110.49—Software Approval Guidelines
- FAA Order 8110.105—Simple and Complex Electronic Hardware Approval Guideline
- FAA AC 20-115C—Airborne Software Assurance
- FAA AC 20-152—Design Assurance Guidance for Airborne Electronic Hardware.

5.1.2 The requirements for the software/electronic hardware are identified during the system safety assessment. Applicants are required to submit life cycle data to meet the objectives in either:

- Annex A to RTCA/DO-178C
- Appendix A of RTCA/DO-254.

5.1.3 Applicants are required to submit the software/electronic hardware life cycle data to CASA for approval. Applicants are encouraged to send more life cycle data than the minimum requirements of RTCA/DO-178C or RTCA/DO-254. An approval will be granted by CASA when the applicant satisfies the life cycle data objectives of RTCA/DO-178C or RTCA/DO-254.

5.1.4 Applicants submitting a DAL E application will require prior agreement and approval by CASA that the system safety assessment supports a case that the design would cause no detrimental effect on aircraft operational capability or pilot workload. If CASA agrees to a DAL E, there are no life cycle data objectives from either RTCA/DO-178C or RTCA/DO-254 to be completed.

Note: An example of DAL E software could be independent in-flight entertainment systems.

5.1.5 Approval of software or electronic hardware will be based on evidence of compliance with the objectives detailed in RTCA/DO-178C or RTCA/DO-254, as appropriate. Alternative standards may be considered.

5.1.6 As software is considered an aeronautical product, it is required to have an authorised release certificate in accordance with either regulation 42.420 of CASR or regulation 42W of CAR.[9]

Software and electronic hardware are approved as an integral part of an aircraft system's development. Approval does not occur in isolation – it is granted when all aircraft compliance requirements are met.

[9] For further information on authorised release certificates see CAAP 42W-2(5) or the Part 42 MOS.

5.2 Modification of approval

5.2.1 Any change to software/electronic hardware life cycle data or LRU details, as listed on the current approval, will require re-approval. The applicant should forward details of any proposed changes to CASA to initiate the re-process.

5.2.2 Any approved change to life cycle data that affects the software approval will render obsolete the previous approval, due to the fact that a change in life cycle data or system requirements requires a re-assessment of the system safety assessment.

5.2.3 If the applicant proposes a change to embedded code contained in an electronic hardware component, which does not affect the hardware life cycle data that is contained in the PHAC, then the applicant can follow RTCA/DO-178C design assurance processes.

5.3 Approved design organisations and authorised persons

5.3.1 CASA may authorise an approved design organisation under Subpart 21.J, or an individual under regulation 201.001, to approve designs, modifications and technical data for software life cycle data or electronic hardware life cycle data under Part 21. Such authorisations may be subject to limitations specified by CASA.

5.3.2 Organisations and individuals seeking such authorisations can contact CASA's Permission Applicant Centre or by email to aircraft.certification@casa.gov.au. CASA may consider one or more of the following criteria when assessing an application to appoint an authorised person:

- completion of RTCA/DO-178C or RTCA/DO-254 course
- experience with RTCA/DO-178B or RTCA/DO-254
- relevance of Authorised Persons experience speciality in the requested technical speciality
- a history of successful software or electronic hardware certification projects
- ability to consistently produce RTCA/DO-178C or RTCA/DO-254 life cycle data
- the applicants documented procedures for carrying out the relevant activities.

5.3.3 Once satisfied of an applicant's suitability, CASA will grant an authorisation with either or both of the following specialities:

- software – in accordance with RTCA/DO-178C and limited to certain DALs and applications
- electronic hardware – in accordance with RTCA/DO-254 and limited to certain DALs and applications.