Plan for Software Aspects of Certification (PSAC) for the Airspace Access & Services (AAS) Data Link Communications Application (DLCA)

Document Number XXX-XXXX-XX

Revision -

CAGE Code 0EFD00EFD00EFD0

Rockwell Collins

Contract Number None

NOTICE: The contents of this document are proprietary to Rockwell Collins and shall not be disclosed, disseminated, copied, or used except for purposes expressly authorized in writing by Rockwell Collins.

The technical data in this document (or file) is controlled for export under the Export Administration Regulations (EAR), 15 CFR Parts 730-774. Violations of these export laws may be subject to fines and penalties under the Export Administration Act

© 2022 Rockwell Collins.

|  |  |  |  |
| --- | --- | --- | --- |
|  | NAME | TITLE | APPROVAL |
| Prepared By: | Mohammed Musthafa | PreparerPreparerPreparer | N/A |
| Approved By: | Lori J Sipper | EngineeringEngineeringEngineering | On File |
| Approved By: | Mikael Molina Sandoval | Safety Engineer | On File |
| Approved By: | Eileen P Roberson | DAC EngineerDAC EngineerDAC Engineer | On File |
| Approved By: | Hatem Abu-Dagga | TPM | On File |

REVISION HISTORY

|  |  |  |  |
| --- | --- | --- | --- |
| REV | DESCRIPTION | DATE | APPROVED |
| - | Initial version of DLCA-6510 PSAC | 2022-08-10 | xxxxxxxxx |

Table of Contents

1 Scope 8

1.1 Purpose 8

1.2 Applicability 9

1.3 Project Overview 9

2 References 11

2.1 Rockwell Collins Documents 11

2.1.1 General Documents 11

2.1.2 Project-Specific Documents 12

2.2 External Documents 14

2.2.1 General Certification Documents 14

2.2.2 Project-Specific Certification Documents 15

2.3 Miscellaneous Documents 15

3 System Overview 16

3.1 System Functional Description 16

3.1.1 CCM System Functional Description 17

3.1.2 AFD-37X0 System Functional Description 17

3.2 System Safety Considerations 18

3.3 System Architecture 18

3.3.1 Hardware Architecture 18

3.3.1.1 CCM System Architecture 18

3.3.1.2 AFD-37X0 System Architecture 18

3.3.2 Software Architecture 19

3.3.3 Hardware/Software Interfaces 19

3.4 Allocation of System Functions 19

4 Software Overview 21

4.1 Software Functions 21

4.1.1 CCM Hardware– Software Configuration Part Numbers 21

4.1.2 AFD-37X0 Hardware - Software Configuration Part Numbers 22

4.2 Software Architecture 23

4.2.1 Application Specific Layer 24

4.2.1.1 HMI 24

4.2.1.1.1 CPDLC Page Objects A661 (HMI A661) 24

4.2.1.1.2 Message Requests and Responses 24

4.2.1.2 Core Library 24

4.2.1.2.1 Automatic Dependent Surveillance Contract (ADS) 24

4.2.1.2.2 Data Manager Client (DMC) 25

4.2.1.2.3 Dialogue Service Interface Client (DSI) 25

4.2.1.2.4 Managed Information Base (MIB) 25

4.2.1.2.5 ATN CPDLC Application Service Element (CPDLC ASE) 25

4.2.1.2.6 FANS CPDLC Application Service Element (CPDLC ASE) 26

4.2.1.2.7 ATN CM Control (CM Control) 26

4.2.1.2.8 ATN CM Application Service Element (CM ASE) 26

4.2.1.2.9 CPDLC Control 26

4.2.1.2.10 Advisory Controller (Advisory) 27

4.2.1.2.11 ACARS Compatible System (ACS) 27

4.2.1.2.12 ATS Facilities Notification (AFN) 27

4.2.1.2.13 Health and Status (H&S) 27

4.2.1.2.14 Dual Controller (Dual) 27

4.2.1.2.15 PM BOP Interface (BOP Interface) 27

4.2.1.2.16 I/O Interface 27

4.2.1.2.17 SysVars 27

4.2.1.2.18 Debug/Trace Controller 27

4.2.1.2.19 Printer Controller 27

4.2.1.2.20 NVM Controller 28

4.2.1.2.21 Datalink Recording 28

4.2.1.3 Message Library 28

4.2.1.3.1 Message Server 28

4.2.1.3.2 Message Processing 28

4.2.1.3.3 DM ASN.1 Codec 28

4.2.1.3.4 ATN/FANS ASN.1 Codec UPER (Codec) 28

4.2.1.3.5 Message Formatter 28

4.2.1.3.6 Message Repository 28

4.2.1.4 XML Files 28

4.2.1.4.1 XML I/O Configuration File 29

4.2.1.4.2 XML ATN Configuration File 29

4.2.1.4.3 XML DLCA General Configuration File 29

4.2.1.5 A661 Definition Files 29

4.2.2 Platform Specific Layer 29

4.2.3 External Libraries 30

4.2.3.1 Platform Libraries 30

4.2.3.1.1 LynxOS-178 Standard Libraries and Includes 30

4.2.3.1.2 Persistent Storage Library 30

4.2.3.1.3 Common I/O (CIO) Library 30

4.2.3.1.4 Reliable User Datagram Protocol (RUDP) 31

4.2.3.1.5 Error Logging and Watchdog Library 31

4.2.3.1.6 Protocol Manager Client library 31

4.2.3.1.7 Avionics Full Duplex Avionics System LAN (AFDX-ASL) 31

4.2.3.1.8 A661 API and A661 Common Library 31

4.3 Overview of Applied New Technology 32

4.4 Software Fault Management Techniques 32

4.5 Software Partitioning 32

4.6 Software Timing and Scheduling Strategies 32

5 Certification Considerations 33

5.1 Certification Basis and Proposed Means of Compliance 33

5.2 Software Criticality Level 33

5.3 Non-TSO Functionality 35

5.3.1 Non-TSO Function #1 36

5.3.2 Non-TSO Function #2 36

5.4 Outsource / Offshoring Entity and Risk Summary 36

6 Software Life Cycle 37

6.1 Description of the Software Life Cycle Processes 37

6.1.1 Planning 37

6.1.1.1 Planning Approach on Future Minor Changes 38

6.1.2 Software Development Process 38

6.1.2.1 Software Development Environment 38

6.1.2.2 Software Requirements Process 39

6.1.2.2.1 Requirements Configuration Management 39

6.1.2.2.2 Requirements Reviews 39

6.1.2.3 Software Design Process 40

6.1.2.3.1 Design Configuration Management 40

6.1.2.3.2 Design Reviews 40

6.1.2.4 Software Implementation/Coding Process 41

6.1.2.4.1 Implementation/Coding Configuration Management 41

6.1.2.4.2 Implementation/Coding Reviews 41

6.1.2.5 Software Integration Process 41

6.1.2.5.1 Integration Configuration Management 41

6.1.2.5.2 Integration Reviews 42

6.1.3 Software Verification Process 42

6.1.3.1 Test Case 42

6.1.3.1.1 Test Configuration Management 42

6.1.3.1.2 Test Case Reviews 42

6.1.3.2 Test Procedure 43

6.1.3.2.1 Test Procedure Configuration Management 43

6.1.3.2.2 Test Procedure Reviews 43

6.1.3.3 Verification Testing 43

6.1.3.3.1 Test Configuration Management 44

6.1.3.3.2 Test Reviews 44

6.2 Peer Review Process 44

6.3 Configuration Management Process 45

6.3.1 Problem Reporting / Change Request Process 45

6.4 Software Quality Assurance Process 45

6.5 Organizational Responsibilities 45

6.5.1 Staffing 46

6.5.2 Roles and Responsibilities 46

6.6 Certification Liaison 47

7 Software Life Cycle Data 49

8 Certification Schedule 53

9 Additional Considerations 54

9.1 Use of Previously Developed Software 54

9.2 Commercial Off The Shelf (COTS) Software 54

9.3 Product Service Experience 54

9.4 Tool Assessment and Qualification 54

9.4.1 Virtual Integrated Software Testbed for Avionics (VISTA) 60

9.4.2 VectorCAST Cover Tool Suite 61

9.5 Option Selectable Software 61

9.6 User Modifiable Software 62

9.7 Field Loadable Software 62

9.8 Multiple Version Dissimilar Software 62

9.9 Parameter Data Items 62

9.10 Model-Based Development (MBD) Methods and Tools 62

9.11 Multi-Core Processor Software 62

9.12 Object Code Coverage Analysis 63

9.13 Object Oriented Methods and Languages 63

9.14 Alternative Methods 63

10 Outsourcing/Offshoring & Supplier Oversight 64

10.1 Tasks and Responsibilities 64

10.2 Technical Oversight 65

10.3 Visibility of Regulations, Plans, and Standards 65

10.4 Integration Management 66

10.5 Problem Reporting and Resolution 67

10.6 Integration Verification Activity 67

10.7 Configuration Management 68

10.8 Design Assurance Oversight 68

10.9 Compliance Substantiation and Data Retention 68

10.10 Certification Oversight 69

Appendix A List of Acronyms 70

Appendix B Planning Change Impact Analysis 75

B.1 Software Component Changes 75

B.2 Change Description (3.1.2) 75

B.2.1 Documentation Updates 75

B.3 Verification Approach 76

**B.4** **Software Level Changes (3.1.3.1)** 76

B.5 Development or Verification Environment Changes (3.1.3.2) 76

**B.5.1** **Tool Changes (3.1.3.4)** 77

**B.6** **Software Process Changes** 77

**B.7** **Relevant Target Changes (3.1.3.5)** 77

**B.8** **Configuration Data Changes (3.1.3.6)** 77

**B.9** **Software Interface and Input/Output Impact (3.1.3.7)** 77

**B.10** **Requirements, Design, Architecture, and Source Code Impact (3.1.3.8)** 77

B.11 Supplier/Outsourcing/Offshoring Activities 77

B.12 IMA Compliance Impact 78

B.13 Environmental/EMI Compliance Impact of Software Changes 78

B.14 Proposed Stage of Involvement Reviews 78

B.15 Legacy Software Analysis 79

Appendix C DO-178C Matrix 82

Appendix D FAA Order 8110.49A Matrix 83

Appendix E Outsource/Offshoring of Regulatory Objectives 84

Appendix F Hardware Dependent Components 87

Appendix G Level of Involvement Self Assessment 88

List of Figures

Figure 3‑1 CCM High Level Architecture 17

Figure 3‑2 AFD-37X0 High Level Architecture 17

Figure 4‑1 CCM CPCI Configuration Part Numbers 22

Figure 4‑2 AFD-37X0 CPCI Configuration Part Numbers 23

Figure 4‑3 Detailed Software Architecture 24

Figure 8‑1 – Representative Figure 53

Figure 10‑1 – Legacy Software Certification Analysis (AC 20-115D) 81

List of Tables

Table 1‑1 – Applicable Equipment 9

Table 3‑1 Minimum Memory and Timing Requirements 20

Table 5‑1 DLCA-6510 Criticality Data 34

Table 7‑1 DLCA-6510 Life Cycle Data Items 49

Table 8‑1 – Representative Table 53

Table 9‑1 - DLCA-6510 S/W Tools 55

Table 9‑2 DLCA Options (Licenses Keys/Strapping) 61

Table 10‑1 DO-178C vs PSAC 82

Table 10‑2 FAA Order 8110.49A Matrix 83

Table 10‑3 Outsource/Offshoring Activities 84

Table 10‑4 -Hardware Dependent Components 87

# Scope

## Purpose

The purpose of this document is to fulfill the Plan for Software Aspects of Certification (PSAC) expectations as identified in RTCA/DO-178C [4] & EuroCAE/ED-12C [4] (hereafter referred to as DO-178C [4]) section 11.1 based on the guidance from FAA Advisory Circular (AC) 20-115D [3]. Clarification contained in RTCA/DO-248C [5] & EuroCAE/ED-94C [5] (hereafter referred to as DO-248C [5]) will also be used.

In addition to DO-178C [4], the guidance contained in FAA Order 8110.49A [10] will be applied to this project; refer to Appendix D for details.

The following identifies the applicability of supplements to this project:

* Applicable: RTCA/DO-330 (EuroCAE/ED-215) [6] Tool Qualification
* Not Applicable: RTCA/DO-331 (EuroCAE/ED-217) [7] Model-Based Development
* Applicable: RTCA/DO-332 (EuroCAE/ED-218) [8] Object-Oriented Technology
* Not Applicable: RTCA/DO-333 (EuroCAE/ED-216) [9] Formal Methods

The following discussion TAIL are currently being handled as aircraft certification Issue Papers or CRIs. It is optional for teams to include this information or not, however as many projects cover the issues it may be valuable to identify awareness of this TAIL list. This is primarily being provided in this template to raise project awareness to improve areas of increased project risk. TAIL identifies international and FAA regulatory concerns. In many cases these are concerns in which the FAA and international regulatory authorities do not know how to address yet and therefore what is required for one aircraft may change on the next aircraft. This list originally was intended for domestic Part 25 transport aircraft, but has been expanded and is being applied other aircraft including BRS class aircraft (fixed wing and rotary). This list is currently located at http://www.faa.gov/aircraft/air\_cert/design\_approvals/transport/media/rpttailistforpublicweb.pdf.

Regulatory considerations from the FAA Transport Airplane Issues List (TAIL) are also addressed in this document. Following are the TAIL Airborne Electronic Hardware (AEH) regulatory concerns which have been evaluated for applicability to this project:

* Database and Data Structure Assurance: Addressed in section 9.9.
* General Software Guidance: Addressed throughout this document.
* Host Environment Testing Validity: Addressed in section ???.
* Management of Open Problem Reports: Addressed in section ???.
* Model-Based Development (MBD) Methods and Tools: Addressed in section 9.10.
* Multi-Core Processors: Addressed in section 9.11.
* Object Code Coverage Analysis: Addressed in section 9.12.
* Object Oriented Methods and Languages: Addressed in section 9.13.
* Software Maturity Prior to Flight Test: Addressed in section ???.

Note: Many projects address new certification items by applying CAST papers. This is an acceptable approach in the absence of any other guidance; however teams must recognize that CAST papers are not an official regulatory position. CAST papers only identify discussions between civil aviation authorities. Thus applying CAST positions do not mitigate risk of guidance changes.

DO-178C [4] is a consideration of parts certification (FAR Part 21; TSO) and aircraft certification (FAR Part 23/25/27/29) approvals. DO-178C [4] expectations are equivalent between parts and aircraft certification approvals. This document focuses on DO-178C [4] activities that are common with all certification activities. Since this is a product intended to be used on several aircraft, any additional aircraft certification details will be addressed in separate supplement document(s). These supplement document(s) will be identified in the <Software Configuration Index Document/Field Loadable Software Configuration Index Document (SCID/FLSCID) [??]>. The intent of the <SCID/FLSCID [??]> is to identify the detailed software configuration and the associated software life cycle information.

## Applicability

This section identifies the equipment applicable to this document. This section also states whether this is an initial submittal or a modification and also identifies the baseline if applicable (it is possible for an initial development to have a baseline if this is a new Type Identifier but based on a different Type Identifier).

This document is applicable to the <initial development>~~<modification>~~ initial development of the Data Link Communications Application (DLCA-6510) software product, which will be hosted on the Common Computing Module (CCM) and the Adaptive Flight Display AFD-37X0. Throughout this document when there are differences between the CCM and the AFD-37X0, the differences will be noted. DLCA-6510 will be developed and verified in accordance with the objectives for DO-178C Level C software. Table 1 identifies the planned equipment for this specific project. However, a minor change typically does not require a new PSAC and therefore this PSAC is also expected to support potential future minor changes.

(For Lori: We need to decide the information to be placed here)

Table 1‑1 – Applicable Equipment

| **Baseline RCPN** | **New RCPN** | **Description** |
| --- | --- | --- |
| Enter “N/A, initial development” if this is not an LRU modification | RCPN#1 | Very high level description of the equipment. This is mostly valuable to help readers understand the basic functionality and differences when multiple RCPNs are listed.  <For Lori: We need to decide the information goes here> |
|  | RCPN#2 |  |

## Project Overview

Important Consideration to Project Teams: Teams must remember that a project needs to simultaneously address both TSO approval and aircraft certification. Sometimes the scope of the project is different between TSO and aircraft certification. To minimize issues during certification reviews, teams need to clearly identify the scope of both TSO and aircraft activities if different. For example, sometimes an aircraft certification includes more than one TSO certification change, this section needs to help certification personnel on both activities understand the scope pertinent to them.

This section provides a very high level summary of this project to help a reader understand the purpose of this project.

If this is a modification project, provide a textual description of the reason for the changes from the baseline identified in section 1.2 along with a very high level summary of the functional changes which is understandable to someone outside the project. DO NOT force a reader to leave this section to understand this summary. For example some projects just list CR numbers which forces a reader to search for additional information which immediately frustrates the intended audience by creating an abundance of questions.

If this is an initial development summarize the basic functionality, whether this is based on an existing product or not, identify significant reuse considerations, and any other significant certification considerations.

The DLCA-6510 provides the Link 2000+ implementation of Aeronautical Telecommunication Network (ATN) applications and FANS 1/A+ applications. These applications define the Data Link methods and messages that are exchanged between the aircraft and the ground services in support of Air Traffic Services (ATS) and Air Traffic Control (ATC). The application supports seamless transfer of bilingual aircraft transiting from FANS to ATN and ATN to FANS per DO-305A [73], Section 4 interoperability requirements. DLCA-6510 software described by this PSAC represents the next generation of datalink software to be hosted in the CCM/AFD. It will inherit from the DLCA-6500 generation, but will start a brand new baseline beginning with Block Point 1. DLCA-6500 Block Pint TBD software will be used as baseline for DLCA-6510 software. <For Lori: We need to decide the information goes here>

# References

This section identifies all specifications, standards, industry documents and other publications, which are referenced in the body of this document. There is no need to include documents that are not referenced in herein.

The configuration of the planning documents must be documented, under appropriate CC1/CC2 CM control, and identified in the PSAC submittal. CM control is typically an SCL release. To identify the configuration, each planning life cycle artifact must be identified by its identifier (typically RCPN) and version (typically revision letter). There are two typical approaches to address the planning configuration; (1) in the PSAC References section and (2) in a SCI (SCID, FLSCID, CID, etc) document (recommendation is #2, to create a SCID/FLSCID/CID during the planning phase and update it at each major delivery/submittal). A statement like the following should be provided if option #2 is applied: “Refer to <SCID/FLSCID/CID/etc> [?] to identify the revision applicable to this project.”

Titles must be exactly the same as the title on the reference cover page. If a typo exists on the title page then that typo should exist on the reference to that document. Confusion exists outside Rockwell Collins when titles are not aligned with the provided document. Differences create doubts to an external reader whether they have received the proper document or if the issue is only with the title. Note: PDM database contains titles which may be different from the actual title and therefore must be checked.

Some documents use a date as a version identifier, when a date is listed it should be formatted exactly as listed on the official document. Dates are written in a wide variety of formats, some projects attempt to use a uniform date format which can result in confusion and errors which is why it is preferred to use the exact formatting as listed on the document.

References in this template are listed by identifier (part number & version) followed by the title. This order was intentional as it is easier for an auditor or reader to cross-check and therefore is preferred. In addition this approach can reduce cross reference errors. However, this order of providing references is not required.

## Rockwell Collins Documents

It is undesirable to list reference documents which are not SCL released when the plans are complete. However, some customers insist on identifying documents planned to be created/modified during this project but are not yet released. Any document which is not released at the time the PHAC is submitted must be clearly identified as not SCL released. Some possible examples are using an RCPN with a “-TBD” extension or “Rev TBD” and explaining that these identify future documents which are not currently SCL released in section 2.

### General Documents

This section identifies the applicable Rockwell Collins documents which are applicable to multiple projects such as corporate or department level documents.

Note: The purpose of this document is to define the plan to address industry commercial certification considerations with a target audience of external regulatory officials. References to Rockwell Collins policies/procedures should be minimal in this document. Rockwell Collins policies/procedures address business expectations which are not aligned with industry commercial certification considerations. Corporate documents often exceed minimum certification expectations in some areas, and often do not fully meet regulatory configuration management expectations; projects are responsible to address regulatory expectations. Also, corporate documents typically do not meet FAA data retention expectations; thus if corporate documents are used to satisfy regulatory expectations the project needs address CM expectations (see RC-ENG-P-000 17 Aug 2016 section 7.2d).

1. Rockwell Collins Technical Consistent Process Version 3.2, RCPN 832-8716-009
2. Design Quality Assurance Plan for Hardware, Software and System Development, RCPN 946-5892-100
3. Software Configuration Management Plan, RCPN 832-2963-001
4. Risk Assessment and Oversight for Offshoring or Activities involving Civil Certification, HRC-ENG-P-016
5. Pro Line Fusion® Input Output Common Format Interface Definition Document (IOCF IDD) Process BRS-ENG-P-006
6. Change Request and CCB Process for the Commercial Systems Data Link Organization Using JIRA , RCPN 946-8189-002

### Project-Specific Documents

This section identifies the documents that are specific to the project (e.g. system level documents).

1. Software Development Plan (SDP) for DO-178C Data Link Products (RCPN TBD)
2. Coding Standards for C++ Language, RCPN 832-0536-005 (RCPN TBD)
3. Application Footprint Document for the Data Link Communications Application (DLCA 6500) 945-8964-(XXX) (RCPN TBD)
4. Application Footprint Document for the Airspace Access & Services (AAS) Data Link Communications Application (DLCA-6510) System
5. High-Level Software Requirements for the Airspace Access & Services (AAS) Data Link Communication Application (DLCA) (RCPN TBD)
6. Low-Level Software Requirements for the Airspace Access & Services (AAS) Data Link Communication Application (DLCA)
7. Software Requirements Specification (SRS) for the Data Link Communications Application (DLCA) Future Air Navigation System (FANS-1/A), RCPN 945-0516-(XXX) (RCPN TBD)
8. Software Requirement Specification (SRS) for Common System Services Data Link Communication Application (DLCA), RCPN 945-0592-(XXX) (RCPN TBD)
9. Software Requirements Specification (SRS) for Aeronautical Telecommunication Network Context Management (CM) And Controller Pilot Data Link Communication (CPDLC), RCPN 945-0591-(XXX) (RCPN TBD)
10. Software Requirement Specification (SRS) for DLCA-6500 Human Machine Interface (HMI), RCPN 945-0517-(XXX) (RCPN TBD)
11. Input/Output Common Format (IOCF) Interface Definition Document for Airspace Access & Services (AAS) Data Link Communication Application (DLCA) (RCPN TBD)
12. Software Design Document (SDD) for the Airspace Access & Services (AAS) Data Link Communication Application (DLCA) (RCPN TBD)
13. Computer Program Configuration Item for the Data Link Communication Application (DLCA-6500) with A661 for Human Machine Interface (HMI) RCPN 096-6363-(XXX) (RCPN TBD)
14. Computer Program Configuration Item (CPCI) for the Data Link Communication Application (DLCA-6500) Core Software Library, RCPN 096-2550-(XXX)
15. Computer Program Configuration (CPCI) for the Data Link Communication Application (DLCA-6500) Message Library, RCPN 096-6864-(XXX) (RCPN TBD)
16. Computer Program Configuration Item (CPCI) for the Airspace Access & Services (AAS) Data Link Communication Application (DLCA) XML I/O Configuration File (RCPN TBD)
17. Computer Program Configuration Item (CPCI) for the Airspace Access & Services (AAS) Data Link Communication Application (DLCA) XML Default ATN Addresses (RCPN TBD)
18. Computer Program Configuration Item (CPCI) for the Airspace Access & Services (AAS) Data Link Communication Application (DLCA) XML General Configuration File (RCPN TBD)
19. Software Verification Procedures and Results (SVPR) for the Airspace Access & Services (AAS) Data Link Communication Application (DLCA) (RCPN TBD)
20. Software Accomplishment Summary for the DLCA-6500, RCPN 945-0519-(XXX) (RCPN TBD)
21. Airspace Access & Services (AAS) Data Link Communication Application (DLCA) Software Top Level (FANS/ATN Dual Stack (RCPN TBD)
22. Data Link Products Peer Review Checklists, RCPN 963-9782-100 (RCPN TBD)
23. Peer Review Method Using PREP for the Commercial Systems Data Link Organization, RCPN 945-9104-(XXX) (RCPN TBD)
24. CoRE Platform Software Accomplishment Summary, RCPN 815-0524-(XXX) (RCPN TBD)
25. LynxOS-178 Software Accomplishment Summary, AAN-1161-02-(XXX) (RCPN TBD)
26. Software Accomplishment Summary (SAS) for the Protocol Manager Application (PMA-6000), RCPN 963-6390-(XXX) (RCPN TBD)
27. Software Accomplishment Summary (SAS) for the AFDX-ASL, RCPN 815-9675-(XXX) (RCPN TBD)
28. Plan for Software Aspect of Certification for the Pro Line Fusion ARINC-661 Graphics Server and ARINC 661 Application Programming Interface, RCPN 963-9111-008 (RCPN TBD)
29. Computer Program Configuration Item (CPCI) for the Support Files for the DLCA-6500, RCPN 096-9885-(XXX) (RCPN TBD)
30. Computer Program Configuration Item (CPCI) for the Airspace Access & Services (AAS) Data Link Communication Application (DLCA) A661 Definition Files (RCPN TBD)
31. Persistent Storage Design Description, RCPN 815-0020-(XXX) (RCPN TBD)
32. CoRE Common Input Output – CIO- SW Design Description, RCPN 815-0119-(XXX) (RCPN TBD)
33. Reliable User Datagram Protocol – RUDP Communication Library Design Description, RCPN 815-0686-(XXX) (RCPN TBD)
34. Reliable User Datagram Protocol – RUDP Connection Library Design Description, RCPN 815-0712-(XXX) (RCPN TBD)
35. Health Monitor SW Design Description, RCPN 815-0594-(XXX) (RCPN TBD)
36. Software Design Description (SDD) for PMA-6000, RCPN 815-9996-(XXX) (RCPN TBD)
37. AFDX Local Area Network – LAN – Design, RCPN 815-0533-(XXX) (RCPN TBD)
38. ARINC 661 Application Programming Interface (API) Software Design Document, RCPN 964-2609-(XXX) (RCPN TBD)
39. Software Accomplishment Summary for the Fusion ARINC 661 Application Programming Interface (A661 API), RCPN 964-9718-(XXX) (RCPN TBD)
40. Software Developers User's Guide for the Data Link Communication Application (DLCA) ARINC 661 Projects, RCPN 946-5683-(XXX) (RCPN TBD)
41. Software Verification User's Guide for the Airspace Access & Services (AAS) Data Link Communication Application (DLCA) ARINC 661 Projects (RCPN TBD)
42. CoRE Common I/O SW User’s Guide, RCPN 815-0080-(XXX)
43. Software Accomplishment Summary for AFD-37X0 Platform Software, RCPN 945-6685-(XXX)
44. Software Accomplishment Summary (SAS) for the Pro Line Fusion User Interface Data Items for Data Link Communications Application (DLCA), RCPN 946-0M08-(XXX)
45. Software Deliverable for the Data Link Communications Application, RCPN 811-5733-(XXX)
46. Computer Program Configuration Item for the Archive of DOORS Project for DLCA-6500, RCPN 096-2548-(XXX)
47. Computer Program Configuration Item (CPCI) for the Data Link Communication Application (DLCA-6500) Message Library Tester 096-9882-(XXX)
48. Computer Program Configuration Item (CPCI) for the Airspace Access & Services (AAS) Data Link Communication Application (DLCA) VAPS (Virtual Application Protocol Software) Model, RCPN TBD
49. Computer Program Configuration Item (CPCI) for the Data Link Communication Application (DLCA-6500) Electronic Nameplate, RCPN 096-2552-(XXX)
50. Software Deliverable for the Data Link Communications Application Message Library, RCPN 072-0951-(XXX)
51. C Computer Program Configuration Item (CPCI) for the Airspace Access & Services (AAS) Data Link Communication Application (DLCA) Software Verification Procedures and Results (SVPR) (RCPN TBD)
52. Software Configuration Index (SCI) for the Data Link Communications Application (DLCA-6500), RCPN 946-1T23-(XXX)
53. C Computer Program Configuration Item (CPCI) for the Airspace Access & Services (AAS) Data Link Communication Application Processor Configuration Table (PCT), (RCPN TBD)
54. Computer Program Configuration Item (CPCI) for the Airspace Access & Services (AAS) Data Link Communication Application Media Set - EDS ONLY (RCPN TBD)
55. Software Deliverable for the Data Link Communication Application (DLCA-6500) Media Set, RCPN 072-2266-(XXX)
56. DOORS Documentation Method for the Commercial Systems Data Link Organization, 945-9527-001
57. Software Accomplishment Summary for the Reliable User Datagram Protocol (RUDP) Libraries, RCPN 946-0MH2-(XXX)

## External Documents

### General Certification Documents

This section identifies the general certification documents specific to the project (e.g. industry standard documents).

1. FAA AC 20-115D, 07/21/2017; “Airborne Software Assurance”
2. RTCA/DO-178C (EuroCAE ED-12C), December 13, 2011; “Software Considerations in Airborne Systems and Equipment Certification”
3. RTCA/DO-248C (EuroCAE ED-94C), December 13, 2011; “Supporting Information for DO-178C and DO-278A”
4. RTCA/DO-330 (EuroCAE ED-215), December 13, 2011; “Software Tool Qualification Considerations”
5. RTCA/DO-331 (EuroCAE ED-218), December 13, 2011; “Model-Based Development and Verification Supplement to DO-178C and DO-278A”
6. RTCA/DO-332 (EuroCAE ED-217), December 13, 2011; “Object-Oriented Technology and Related Techniques Supplement to DO-178C and DO-278A”
7. RTCA/DO-333 (EuroCAE ED-216), December 13, 2011; “Formal Methods Supplement to DO-178C and DO-278A”
8. FAA Order 8110.49A; “Software Approval Guidelines” 3/29/2018
9. EASA CM – SWCEH – 002, Issue: 01, Revision: 01, 09th of March 2012; “Software Aspects of Certification”
10. FAA Order 8150.1D, 3/17/2017; “Technical Standard Order Program”
11. RTCA DO-258A Interoperability Requirements for ATS Applications Using ARINC 622 Data Communications (FANS 1/A Interop Standard), April 7, 2005
12. RTCA DO-306 Safety and Performance Standard for Air Traffic Data Link Services in Oceanic and Remote Airspace (Oceanic SPR Standard), October 11, 2007
13. RTCA DO-306 Change 1 Safety and Performance Standard for Air Traffic Data Link Services in Oceanic and Remote Airspace (Oceanic SPR Standard), March 17, 2011
14. FAA Order 8110.49A Software Approval Guidelines, March 29, 2018
15. ARINC 622-4 ATS Data Link Applications Over ACARS Air-Ground Network, Oct 12, 2001
16. EUROCONTROL SPECIFICATION on Data Link Services, EUROCONTROL SPECIFICATION, EUROCONTROL-SPEC-0116 Edition 2.1, January 28, 2009
17. INTEROPERABILITY REQUIREMENTS STANDARD FOR AERONAUTICAL TELECOMMUNICATION NETWORK BASELINE 1 (ATN B1 INTEROP STANDARD), EUROCAE ED-110B/RTCA DO-280B, December 2007
18. SAFETY AND PERFORMANCE REQUIREMENTS STANDARD FOR AIR TRAFFIC DATA LINK SERVICES IN CONTINENTAL AIRSPACE (CONTINENTAL SPR STANDARD), EUROCAE ED-120 with Changes 1 and 2, October 2007
19. MANUAL OF TECHNICAL PROVISIONS FOR THE AERONAUTICAL TELECOMMUNICATION NETWORK (ATN), Doc ICAO 9705AN/956, SECOND EDITION – 1999, plus PDR’s identified in EUROCONTROL-SPEC-0116
20. Certification Authorities Software Team (CAST) Position Paper CAST-8, Use of the C++ Programming Language, January, 2002
21. Guidelines for Design Approval of Aircraft Data Link Communications Systems Supporting Air Traffic Services (ATS), AC 20-140C
22. RTCA DO-305A Future Air Navigation System 1/A – Aeronautical Telecommunication Network Interoperability Standard (Fans 1/A – ATN B1 Interop Standard), March 21, 2012
23. ICAO Doc 4444, Procedures for Air Navigation Services, Air Traffic Management, 15th Edition, November 22, 2007
24. ICAO Doc 10037 Global Operational Data Link Document (GOLD) Manual, 1st Edition, 2017
25. Advisory Circular (AC) 20-115D, Airborne Software Development Assurance Using EUROCAE ED-12( ) and RTCA DO-178( ), 2017, July 12

### Project-Specific Certification Documents

This section identifies the certification documents that apply specifically to the project (e.g. TSO documents).

1. xxx

## Miscellaneous Documents

This section identifies any other document relevant to the project.

1. xxx

# System Overview

Certification Note: DO-178C §11.1a: This section provides an overview of the system, including a description of its functions and their allocation to the hardware and software, the architecture, processor(s) used, hardware/software interfaces, and safety features.

It is recommended that the following common wording be used for this paragraph in all PSACs:

This section provides a high-level overview of the system, with emphasis on the safety considerations that need attention in the design. This section includes the system overview of the DLCA-6510, which meets the System Overview objective of the Plan for Software Aspects of Certification found in DO‑178B [61], section 11.1.a. Descriptions are provided in terms of the functions and their allocation to the hardware/software, the architecture, processor(s) used, hardware and software interfaces, and safety features.

<For Lori: This section needs to be revisited/updated after the system architecture is finalized. Currently the information has been kept from DLCA-6500 PSAC>

## System Functional Description

This subsection provides a description of the system’s intended functionality. Also included is a functional block diagram that depicts how the listed functions relate to each other and to the external environment. This functional view provides the reader with a sense of which facilities/functions are involved in performing each of the functions.

This subsection also provides a brief description of the system installation variations that are to be covered by the submittal. The focus should be on installation considerations that may have potential safety related issued, such as single of dual installations. Actual interfaces should not be described here, as they should be contained in s the subsection on System interfaces below, however the associated equipment should be listed.

If security considerations are applicable to this product, they should be summarized in this section.

The DLCA-6510 provides the Link 2000+ implementation of Aeronautical Telecommunication Network (ATN) applications and FANS 1/A+ applications. These applications define the Data Link methods and messages that are exchanged between the aircraft and the ground services in support of Air Traffic Services (ATS) and Air Traffic Control (ATC). The application supports seamless transfer of bilingual aircraft transiting from FANS to ATN and ATN to FANS per DO-305A [73], Section 4 interoperability requirements.

The applications that make up FANS-1/A+ are:

1. ATS Facilities Notification (AFN),

2. Automatic Dependent Surveillance - Contract (ADS-C), and

3. Controller Pilot Data Link Communication (CPDLC)

The AFN, ADS-C, and CPDLC applications are defined by RTCA DO-258 Rev A [62], which is the interoperability requirement standard for FANS-1/A+ systems. The FANS-1/A+ application uses ARINC 622 [66] data communications interface. ARINC 622 provides a definition to convert bit-oriented messages to character-oriented messages that can be transmitted over the ACARS character oriented network. The reverse procedure is performed for receiving messages. This is also known as the ACARS Convergence Function (ACF).

The Link 2000+ ATN applications are:

1. Context Management (CM), and
2. Controller Pilot Data Link Communication (CPDLC)

The Link-2000 ATN applications are defined by RTCA DO-280B/EUROCAE ED-110B [68], and ICAO 9705[70] plus the PDRs as defined by EUROCONTROL-SPEC-0116[67].

Notable ATN differences from FANS 1/A+ includes:

* Full bit-oriented protocol stack
* Only able to transmit messages over VHF Data Link Mode 2 (VDLM2)
* Currently ATN is used only over European domestic airspace, while FANS is used in globally in remote and Oceanic environments.
* Only a subset (now full set in DLCA 6510?) of the ATN CPDLC messages as defined in the EUROCONTROL-SPEC-0116 [67] is supported. FANS supports the full CPDLC message set as defined in RTCA DO-258 Rev A [62].

A portion of the ATN CPDLC and FANS 1/A+ functionality provides the opportunity to harmonize the related functionality. The DLCA-6510, and the associated subsystems, will provide a harmonized interface to both applications. The CPDLC message display text was harmonized using guidance from ICAO Doc 4444 [74] and ICAO Doc 10037 GOLD Manual [75].

DLCA-6510 software can execute on multiple hardware platforms. This PSAC describes the two supported types of hardware:

1. The Common Computing Module (CCM)
2. The AFD-37X0

Figure 3‑1 CCM High Level Architecture and Figure 3‑2 AFD-37X0 High Level Architecture (See figures below) provides a high-level context diagram for the DLCA-6510, which illustrates the major interfaces and data flows to/from the DLCA-6510. The functional components of the DLCA-6510 and the other components are discussed in the subsequent sections.

### CCM System Functional Description

This section provides the high level architecture of Common Computing Module (CCM).

<For Lori: DLSS-18533 will be worked in the future sprint>

A picture containing text, furniture, table

Description automatically generated

Figure 3‑1 CCM High Level Architecture

### AFD-37X0 System Functional Description

This section provides the high level architecture of FUSION Adaptive Flight Display (AFD-37x0)

<For Lori: DLSS-18533 will be worked in the future sprint>

A picture containing text, furniture, table

Description automatically generated

Figure 3‑2 AFD-37X0 High Level Architecture

## System Safety Considerations

This subsection provides a brief description of the consequences of the failure of the equipment discussed within this PSAC and should emphasize architectural features implemented to mitigate system safety considerations. Significant failures and their classification from the relevant Functional Hazard Assessment and Preliminary System Safety Assessment should be included in this subsection.

The goal of this project is to develop and verify the DLCA-6510 application executes with the Platform Software (See Table 10‑1 -Hardware Dependent Components for appropriate citation). The development and verification process must ensure that this application provides sufficient integrity, reliability, and/or redundancy to assure that all requirements associated with avoidance of operational hazards are complied with when integrated with existing and new avionics on the target airplane.

The system safety and partitioning concepts, such as resource management, fault tolerance, and scheduling algorithms, are provided by the underlying platform software in the Integrated Modular Architecture (IMA) environment (see sections 4.4.1.4.3 and 4.4.3 below).

## System Architecture

This subsection provides a top-level description of the system architecture.

This subsection describes, from a systems perspective, what is known about the hardware architecture at the time of the preparation of this PSAC with emphasis on system safety.

The system is based on a modular and integrated architecture that combines multiple concentrated processing centers with localized point-to-point communications and a high-speed communications network between processing centers to minimize the installation requirements on the aircraft.

### Hardware Architecture

This subsection describes, from a systems perspective, what is known about the hardware architecture at the time of the preparation of this PSAC with emphasis on system safety.

Details of the Hardware Architecture can be found in the Platform Software Accomplishment Summary (SAS) (Appendix E Hardware Dependent Components).

#### CCM System Architecture

The DLCA-6510 software executes on a Common Computing Module (CCM) that provides the basic computing resource, operating environment, and communication network. The CCM utilizes the LynxOS-178 operating system that allows “hard” real time partitioning to ensure that no application can “starve” any other application of resources (i.e. time or memory resources).

Details of the Hardware Architecture can be found in the CoRE Platform Software Accomplishment Summary (SAS) [27].

#### AFD-37X0 System Architecture

The DLCA-6510 software executes on a FUSION Adaptive Flight Display (AFD-37x0) that provides the basic computing resource, operating environment, and communication network. The AFD-37x0 utilizes the LynxOS-178 operating system that allows “hard” real time partitioning to ensure that no application can “starve” any other application of resources (i.e. time or memory resources).

The DLCA-6510 application software interfaces with ARINC 661 Graphics Server (AGS) via binary definition files (BDFs) submitted under the User Interface Data Items SAS. These BDFs establish the graphical components used by the DLCA-6510 and their interfaces, allowing the software to modify the graphics at run-time. The BDFs and other configuration files submitted under the User Interface Data Items SAS [47] are loaded into the AGS at initialization and provide the AGS with the input required for generation of graphics services on the display.

Details of the Hardware Architecture can be found in the AFD-37x0 Platform Software Accomplishment Summary (SAS) [46].

### Software Architecture

This subsection describes, from a systems perspective, what is known about the software architecture at the time of the preparation of this PSAC with emphasis on system safety.

The DLCA-6510 software architecture is explained in section 4.4 Software Architecture. This section provides a graphical high level overview of the system along with a detailed description of each component

### Hardware/Software Interfaces

This subsection provides an overview of the interfaces between hardware and software. Interfaces between software and complex hardware items should be emphasized.

## Allocation of System Functions

This subsection tabulates the significant systems functions, the safety level of each of these functions, and the allocation of the functions to hardware and software.

If the use of complex hardware items is planned, the allocation description should identify which functions will be implemented by such devices (complex item to function, and function to complex item).

The DLCA-6510 code will be developed using the C++ programming language for a PowerPC target processor running on LynxOS-178 operating system [28].

The minimum timing and memory requirements allocated to DLCA-6510 are identified in Table 3‑1. The actual system allocation might be different per program setting. These values are the minimum values needed for DLCA to operate. These requirements will be validated throughout the development cycle and listed in the Footprint measurements taken at the completion of the project.

DLCA does not have direct access to persistent memory (FLASH or NVM) since it operates on an Integrated Modular Architecture (IMA). The system allocates NAND (FLASH) file system resources to store the executable, configuration (XML) files and the VAPS (BDF/TDF) files. The platform provides an abstraction layer to NVM and the system allocates resources for DLCA.

<For Lori: DLSS-18536 will be worked in the future sprint for finalization of below table>

Table 3‑1 Minimum Memory and Timing Requirements

| **Type** | **DLCA-6510 CCM Minimum Allocation** | **DLCA-6510 AFD-37X0 Minimum Allocation** |
| --- | --- | --- |
| **System RAM** | 20MB (+5 MB) | 20MB (+5 MB) |
| **Minor / Major Frame Time** | 5 ms / 50 ms (+3 Ticks) | 8 ms / 50ms (+3 Ticks) |
| **% Allocated of the Processor** | 10% | 16% |

# Software Overview

Certification Note: DO-178C §11.1b: This section briefly describes the software functions with emphasis on the proposed safety and partitioning concepts. Examples include resource sharing, redundancy, fault tolerance, mitigation of single event upset, and timing and scheduling strategies.

It is recommended that the following common wording be used for this paragraph in all PSACs:

The following subsections provide a brief overview of the software components included in this equipment. The descriptions include the identification of the software component, with emphasis on how safety considerations have been addressed, and a brief description of the approach taken for each consideration. Where the use of previously approved software impacts safety considerations, a description of the scope of reuse is provided, including references to the source of the previously developed software. The descriptions also address the use of new technologies, significant architectural features, and safety techniques such as fail safes, fault tolerance, redundancy, and partitioning.

<For Lori: This section needs to be revisited/updated after the software architecture is finalized. Currently the information has been kept from DLCA-6500 PSAC>

## Software Functions

This section identifies and defines significant operational functions addressed by software as known at the time of the preparation of this PSAC.

The DLCA-6510 Application (see figures below) is built using the HMI, Core Library, Message Library and Support files. In addition, the DLCA-6510 Application relies on a set of external libraries that are described in section 4.4.3 External Libraries. The external libraries are covered by their own SAS which is identified in section 4.4.3 for each external library.

XML files used to provide startup and configuration information for the DLCA-6510 Application, the XML files are described in section 4.4.1.4XML Files and are covered by this SAS.

The Top Level (T/L) 810-0315-(XXX) and Software Configuration Index (SCI) for the Data Link Communications Application (DLCA-6510) [55] will serve as a software configuration index for DLCA-6510.

The following terms are used in the subsection below and are defined here:

Computer Software Component (CSC) – Collection of more than 1 source file.

Computer Software Unit (CSU) – A single source file.

Executable – A CPCI that is compiled, linked, and executes on the target CPU.

Library – CPCI to be used later during the creation of the DLCA-6510 Application

### CCM Hardware– Software Configuration Part Numbers

This section provides the DLCA-6510 software configuration part numbers for CCM Hardware environment (IPS Media Set). <For Lori: DLSS-18534 will be worked in the future sprint for finalization of below figure>



Figure 4‑1 CCM CPCI Configuration Part Numbers

### AFD-37X0 Hardware - Software Configuration Part Numbers

This section provides the DLCA-6510 software configuration part numbers for AFD-37X0 Hardware environment (EDS Media Set). <For Lori: DLSS-18534 will be worked in the future sprint for finalization of below figure>



Figure 4‑2 AFD-37X0 CPCI Configuration Part Numbers

## Software Architecture

This section includes a list of all major software components included in the system. High-level data flow and control flow between components should be clearly identified. The project should propose, and the Project TSO Compliance Representative (TCR) should concur, with the level of detail to be included in this section. <For Lori: DLSS-18535 will be worked in the future sprint for finalization of below figure>

A picture containing text, furniture, table

Description automatically generated

Figure 4‑3 Detailed Software Architecture

### Application Specific Layer

The DLCA-6510 application specific software layer is made up of software covered by this PSAC and includes libraries not covered by this PSAC.

The application specific software layer provides the following set of specific functions to support the capabilities described in section 3.1 System Functional Description. The following subsections address the software in Figure 4‑3 Detailed Software Architecture covered by this PSAC. Section 4.2.3 gives a description of the external libraries used by DLCA-6510 that are not covered by this PSAC.

#### HMI

The Human Machine Interface (HMI) is a collection CSC’s which provide support for page presentation and interface to the DLCA-6510. The following subsection describes in more detail the CSC’s that make up the HMI.

##### CPDLC Page Objects A661 (HMI A661)

The Controller Pilot Data Link Communication (CPDLC) Page Objects CSC is made up of CSU’s that define and manage the logical views (pages) for an ARINC 661 based HMI for crew interaction with DLCA-6510. These logical views are arranged to form a hierarchical menu structure of pages that allow the crew to:

* Initiate ATS Facilities Notification (AFN) or Context Management (CM) Logon contact with ground Air Traffic Services Units (ATSU).
* Categorically select and compose CPDLC downlink messages.
* View and respond to CPDLC uplink messages.
* Display message log of CPDLC uplink and downlink messages.
* Display Automatic Dependent Surveillance (ADS) connection and contract status.
* Display DLCA-6510 system health and status.
* Terminate CPDLC connection.

##### Message Requests and Responses

The Message Requests and Responses CSC is made up of CSU’s that provide the mechanism through which the HMI can generate downlink requests and obtain information about the operating environment. Message requests interact with the core message logic to generate downlink message. If several displays happen to be viewing and/or manipulating a particular type of request message, each of those displays will be communicating with one host for that message type. In this way, all the displays will be synchronized to the current contents of the message data as contained within that common host

#### Core Library

Sdfsfdsfds

##### Automatic Dependent Surveillance Contract (ADS)

The Automatic Dependent Surveillance (ADS) CSC is responsible for implementing the airborne ADS application as defined in DO-258A.[62] This CSC manages up to four connections with ground ATSU centers and provides air/ground contract management for the various types of ground-initiated ADS contracts (immediate, periodic, and event triggered). This CSC utilizes the Data Manager (DM) Client to establish local contracts with the on-board FMS to retrieve the required data elements at the required time to provide downlink responses for each air/ground contract. There is a 1:1 relationship between an air/ground contract and a local DM contract with the FMS. The ADS CSC has ultimate responsibility for providing air/ground contract responses within the required timeframe, and will do so using default data as described in DO-258A if the FMS and/or DM system becomes unresponsive/unavailable.

##### Data Manager Client (DMC)

The Data Manager Client CSC is responsible for establishing a link with one or more Flight Management System (FMS) Data Manager Servers to exchange data element transfers required by the CPDLC and ADS applications to support a fully integrated Data Link solution with the FMS. Data element exchanges are managed using contracts between the DM client and DM server. Contracts are used to manage the following message types:

* Input Message – Data to be loaded into the FMS
* Request Message – Data requested from the FMS
* Output Message – Response message to a request
* Status Message – Facilitates connection and contract state management

Only the DM client initiates a contract, and both the DM client and DM server maintain contract handlers (state machines) until the contract is fulfilled. The DM client will monitor connection status with the active FMS DM server and will re-establish the connections as necessary.

<For Lori: This will be replaced by new function in the new architecture. Currently the information has been kept from DLCA-6500 PSAC>

##### Dialogue Service Interface Client (DSI)

The Dialogue Service Interface (DSI) Client CSC is used to establish a communication channel via Bit Oriented Protocol (BOP) from DLCA-6510 to the DSI Provider to access the ATN stack. The DSI Provider and ATN stack are hosted on either the Radio Interface Unit (RIU) or the Communications Management Unit (CMU). The Protocol Manager API manages the communication channel between the DSI Provider (RIU/CMU) and the DSI Client (DLCA-6510).

##### Managed Information Base (MIB)

The Managed Information Base (MIB) CSC stores and retrieves application information including addresses, facility designator, application type and version, and aircraft address. The MIB provides an interface that is used by other components which need access to this information.

##### ATN CPDLC Application Service Element (CPDLC ASE)

The Aeronautical Telecommunications Network CPDLC ASE manages states that correspond to establishing an ATN CPDLC connection, accepting CPDLC messages, and closing the CPDLC dialogue.

The states relating to CPDLC connection can only be reached when the ground server sends an uplink requesting a connection to be established. Once the connection is established, the ATN CPDLC ASE will accept CPDLC messages. DLCA-6510 will not be able to send or receive CPDLC messages until this state is reached. The ASE will accept CPDLC messages from the ground server that established the connection until the dialogue is closed. When the CPDLC dialogue is closed, through an end indication or an abort, the DLCA-6510 will no longer accept messages from this ground server until a connection is established with it.

When indications (e.g. end, start, dialogue) are received from the DSI, the ATN CPDLC ASE is responsible for moving these messages along to the CPDLC Control to allow confirmation of these indications. CPDLC Control will send responses and messages to the ASE, and in turn the ASE will move the state as needed as well as send the messages and responses to the DSI client so that these can be sent down to the ground server.

##### FANS CPDLC Application Service Element (CPDLC ASE)

The Future Air Navigation System CPDLC ASE manages states that correspond to establishing an FANS CPDLC connection, accepting CPDLC messages, and closing the CPDLC dialogue.

The states relating to CPDLC connection can only be reached when the ground server sends an uplink requesting a connection to be established. Once the connection is established, the FANS CPDLC ASE will accept CPDLC messages. DLCA-6510 will not be able to send or receive CPDLC messages until this state is reached. The FANS CPDLC ASE will accept CPDLC messages from the ground server that established the connection until the dialogue is closed. When the CPDLC dialogue is closed, through an end indication or an abort, the DLCA-6510 will no longer accept messages from this ground server until a connection is established with it.

When indications (e.g. end, start, dialogue) are received from the ACS, the FANS CPDLC ASE is responsible for moving these messages along to the CPDLC Control to allow confirmation of these indications. CPDLC Control will send responses and messages to the FANS CPDLC ASE, and in turn the FANS CPDLC ASE will move the state as needed as well as send the messages and responses to the ACS so that these can be sent down to the ground server.

##### ATN CM Control (CM Control)

The ATN CM Control is responsible for processing all CM messages that are received from the ATN CM ASE. The ATN CM Control provides automatic responses including processing incoming connections and providing information via the ASE which ultimately will be received by the ground peer, such as the ATC capability of the aircraft. Control includes the capability to manage building and sending Logon Requests, or Aborts. Control also m**a**intains the current ATN logon status.

##### ATN CM Application Service Element (CM ASE)

The Aeronautical Telecommunications Network Context Management ASE accepts three types of uplink messages: Contact, Update and Logon responses. Contact and Update messages can be received by DLCA-6510 when DLCA-6510 is not currently logged on. The messages are used to update the Message Identification Number (MIN) by adding to or updating what is currently present. Logon response messages are received in reply to logon requests sent from DLCA-6510.

DLCA-6510 is responsible for sending logon requests to the ground server in order to begin communication. An ATN CPDLC connection cannot be established until DLCA-6510 is Logged On (i.e. a manually initiated CM Logon has successfully completed). DLCA-6510 will open a CM dialogue when it sends the logon request down and will close the dialogue, unless the ground server requests otherwise, once there is a response. If the dialogue remained open, DLCA-6510 will not close the dialogue until a termination is received from either DLCA-6510 or the ground server.

##### CPDLC Control

The CPDLC Control processes messages received from FANS CPDLC ASE or ATN CPDLC ASE, and when messages are requested to be sent by the HMI. CPDLC Control detects if errors are present within uplink messages received from the ground peer including verifying the included integrity check value. This ensures the message was received from the correct ground peer, was intended for this aircraft, and was not corrupted. CPDLC Control is responsible for generating automated downlink responses such as start or end is accepted or rejected, and CPDLC system messages such as logical acknowledgement, error, current data authority, not current data authority, etc.

CPDLC Control contains and uses message dialogue objects in order for messages that require a response, from either the DLCA-6510 or the ground peer to be monitored. If a CPDLC message does not require a response, the message dialogue will close once it is processed. Messages that do require a response will not close the message dialogue until a CPDLC message response is sent or received for the initial message, or the time allowed for a response to be sent or received expires.

In the interface between the CPDLC HMI and CPDLC Control, the CPDLC Control keeps track of information that deals with the ATC center. The CPDLC Control facilitates the communication between the ground peer and the aircraft; giving information to the HMI to display or transferring messages via the ASE to the ground peer. It provides information the CPDLC HMI requests in respect to connection maintenance such as the current data authority (CDA) and next data authority (NDA).

##### Advisory Controller (Advisory)

The Advisory Controller is responsible for setting/clearing bits in the output word(s) that are broadcast to the Engine Indicating Crew Alert System (EICAS) for aural and visual alerting. The controller is also responsible for building and maintaining a list of inactive and active advisories. It then will use this list to inform when and what advisory should be displayed or cleared.

##### ACARS Compatible System (ACS)

The ACARS Compatible System (ACS) implements the ARINC 622 [66] ACARS Convergence Function (ACF), which takes binary encoded downlink messages and converts them into character-oriented format for transport over the ACARS character-oriented network. Likewise, for uplink messages, it converts from character-oriented format to native binary encoded format.

##### ATS Facilities Notification (AFN)

The AFN CSU is responsible for establishing the initial logon connection with the ground. It is also responsible for handling autonomous ATS facility transfers from one center to the next (CDA to NDA).

##### Health and Status (H&S)

Health and Status provides reporting of application faults to the system Health Monitor. This may include detecting a failure of the peer DLCA-6510 application (in a dual installation) or detecting a loss of I/O with an external peripheral or application.

##### Dual Controller (Dual)

The Dual Controller is responsible for determining the active/standby mode for each peer DLCA-6510 in a dual installation. This determination is made on initial startup to ensure that one peer is active and the other is standby, based on a defined set of rules. Additionally, the Dual Controller monitors for switching conditions during runtime, such that the standby DLCA-6510 will become active (and vice versa) when conditions warrant a mode change.

##### PM BOP Interface (BOP Interface)

The PM BOP Interface provides a method to send and receive BOP messages via PM. The BOP interface also manages the flow control of uplink/downlink messages, CVR data, and printer messages and provides methods to maintain the link and disconnect when necessary.

##### I/O Interface

The I/O Interface uses the CIO library, or set of APIs, to communicate with other subsystems using AFDX. As depicted in Figure 4‑3, this interface is used by DLCA-6510 Health and Status and Dual Controller software modules. The interface abstracts the communication functionality. DLCA-6510 Data transfers include Cross talk data and status, trace data, chime alerts, and status words.

##### SysVars

SysVars is the manager and central repository for DLCA-6510 application system data.

##### Debug/Trace Controller

Debug/Trace Controller manages whether print statements are enable/disabled.

##### Printer Controller

Printer Controller manages the format and delivery of a message to the printer, the messages are those in a closed state. Printer Controller also manages whether the print prompt is displayed or not.

##### NVM Controller

The NVM Controller manages the DLCA-6510 application data that needs to be stored persistently. The NVM controller uses the API provided by the external Persistent Storage API library.

##### Datalink Recording

Provides a single point for formatting and delivering messages for Datalink Recording. Provides methods for sending ASCII copies of uplink/downlink messages to the CVR. Provides methods for sending messages detailing the status of the DLCA, state of CPDLC, and state of advisories to the CVR

#### Message Library

##### Message Server

The Message Server provides the interface to encode, decode, and build a formatted textual message for display on the AFD.

##### Message Processing

The Message Processing invokes the encoder/decoder and message formatter. It also checks if the message can be encoded/decoded using the codec(s).

##### DM ASN.1 Codec

The Data Manager (DM) Abstract Syntax Notation One (ASN.1) Codec is used to encode and decode all messages that are transferred between the Flight Management System’s Data Link Data Manager (DLDM) software and DLCA. The codec provides a uniform means to exchange data in a uniform manner that can be interpreted by different systems.

##### ATN/FANS ASN.1 Codec UPER (Codec)

The FANS/ATN Abstract Syntax Notation One (ASN.1) Codec Packed Encoding Rules, unaligned variant (UPER) is used to decode and encode all FANS and ATN CPDLC uplink/downlink messages, FANS ADS messages, and CM/AFN messages. This codec complies with the ASN.1 syntax defined in DO-258A and ED-110B. The codec provides a uniform means to exchange data in a uniform manner that can be interpreted by different systems.

##### Message Formatter

The message formatter is used to build a textual string to print or display using the raw data received from an uplink or downlink.

##### Message Repository

The Message Repository is the storage location for all uplink and downlink messages. This contains the message header information as well as the message attributes. It allows you to delete and add new messages.

#### XML Files

The DLCA-6510 relies on XML files for customization of program specific options and for I/O configuration information. The XML files the DLCA-6510 application uses are covered by this PSAC. Each XML file is a unique CPCI. Each XML file is used during verification testing of the DLCA-6510. This verification testing validates the XML file is properly configured for the program.

##### XML I/O Configuration File

The XML I/O Configuration File [19] is an XML text file that describes the external interfaces and data flows for the DLCA-6510 application. It contains the Well Known Names (WKN), Well Known Services (WKS) and the NDO IDs as defined in DLCA-6510’s IOCF document [14] for this platform.

This component is configuration controlled as its own entity in order to facilitate common reusable software on different platforms; however it is not intended to be individually loadable in the field. The content of this file will be verified together with the DLCA-6510 application and then released/fielded as a bundled load.

##### XML ATN Configuration File

The XML ATN Configuration File [20] contains the default list of ATN facilities and their respective CM addresses that are going to be pre-loaded in the DLCA-6510 application software. The ATN addresses contained in this file represents the stations the pilot can logon to in ATN service type. In addition to the ATN Addresses, other ATN configuration data will be stored in this file. This data includes information such as ATN Timers.

##### XML DLCA General Configuration File

The XML DLCA General Configuration File[21] is an XML file that contains parameters used to configure DLCA-6510 human machine interface and to determine if this is a CCM or AFD-37X0 environment. This file is to provide a method to modify configurable items within the HMI without requiring a change to the source code.

The XML General Configuration File [21] is an XML text file that provides the ARINC 661 Graphical Server (AGS) connection required for the DLCA-6510 application. It contains the Well Known Names (WKN), Well Known Services (WKS) that are used to communicate with AGS as defined in DLCA-6510’s IOCF document [14] for the platform that the DLCA is hosted. It also includes configuration settings required for the Configurable Inbox.

This component is configuration controlled as its own entity in order to facilitate common reusable software on different plaforms; however it is not intended to be individually loadable in the field. The content of this file will be verified together with the DLCA-6510 application and then released/fielded as a bundled load.

#### A661 Definition Files

The definition file will be used to inform the AGS of the widget data necessary to allocate the memory resources for graphics, as well as to establish a means for the DLCA-6510 to describe and update the user interface details. The definition file will be a binary file auto generated based on the DLCA-6510 Virtual Avionics Prototyping System (VAPS) widget layout.

The Binary Definition File is read by the AFDA-6510 Display Application as part of the process to create the AFD Configuration Table (AFDT) files. Refer to the PSAC for the Flight Display System Application for further detail regarding BDF loading, and validation checks.

The A661 Definition file also contains a Text Data File (TDF). This file is a human readable file of the BDF.

For CCM platform, the VAPS A661 Definition file is created and developed by the DLCA team and covered under the DLCA SAS[23]. For AFD-37X0 platform, the VAPS A661 Definition file is external to DLCA and covered under the User Interface Items SAS [47].

### Platform Specific Layer

The platform specific layer provides software components that allow the DLCA-6510 application to be hosted on the CCM-5110 platform or the AFD-37X0 platform. Some of these software components are listed below for the CCM-5110 platform:

* Boot.
* Kernel Download Image (KDI)
  + LynxOS-178
  + Hardware Support
  + System Applications
  + Static Device Drivers
* User File System (USEFS)
* Platform Software Libraries.
  + LynxOS-178 Standard Libraries and Includes
  + Persistent Storage
  + Common I/O (CIO)
  + Reliable User Datagram Protocol (RUDP) Communication
  + Reliable User Datagram Protocol (RUDP) Connection
* Device drivers – NAND, CRC, and IOC Mezzanine.

The collection of these device drivers and libraries are utilized by DLCA-6510, but are not part of the DLCA-6510 high-level design and are not covered by this PSAC

### External Libraries

In order to communicate with other applications within the IMA system, the DLCA-6510 application will statically link in external libraries that contain client APIs. These APIs establish a common path for multiple applications to communicate with the services provided by the serving application.

All client APIs/libraries linked into the DLCA-6510 software application are required to be separately maintained and verified by the applications providing the service. The development of these client APIs and their related documentation and verification is completely independent of the DLCA-6510, and is outside the scope of this PSAC. All of the client APIs that are linked into the DLCA-6510 application code will go through independent verification to DO-178C level C standards or higher.

#### Platform Libraries

##### LynxOS-178 Standard Libraries and Includes

The LynxOS-178 Standard Libraries & Includes API provides a set of functions used for accessing the operating system. This allows the DLCA-6510 to communicate with the hardware without having to understand the lower level hardware/software protocols. This library is covered by the Platform SAS (See Table 9‑1 -Hardware Dependent Components for appropriate citation).

##### Persistent Storage Library

The Persistent Storage API will provide the DLCA-6510 an interface to store its operational data persistently in Non Volatile Memory (NVM). The data includes CM ground facilities and their ATN address, and ATN settings such as timers, and strapping data.

This library is covered by the Platform SAS (See Table 10‑1 -Hardware Dependent Components for appropriate citation). See the Persistent Storage Design Description [34], section 5.1.1 for API definitions.

##### Common I/O (CIO) Library

The Common I/O library provides an API for an application to transfer information over the Avionics System LAN (ASL) to other applications.

This library is covered by the Platform SAS (See Table 9‑1 -Hardware Dependent Components for appropriate citation). See the CIO Design Description [35], section 3.3.5 for API definitions.

##### Reliable User Datagram Protocol (RUDP)

###### RUDP Communication Library

The RUDP Communication library provides the API to send and receive data over the ASL.

This library is covered by the SAS the Reliable User Datagram Protocol (RUDP) Libraries [60]. See the RUDP Communication Library Design Description [36], section 3.3.1 for API definitions.

###### RUDP Connection Library

The RUDP Connection library is used on the remote application’s side. The RUDP Connection library uses a connection based API to interface with the remote application software.

This library is covered by the SAS the Reliable User Datagram Protocol (RUDP) Libraries [60]. See the RUDP Connection Library Design Description [37], section 3.3.1 for API definitions.

##### Error Logging and Watchdog Library

The Error Logging and Watchdog library provides the APIs for the Health Monitor error logging, error status, and software watchdog functionality.

This library is covered by the Platform SAS (See Table 9‑1 -Hardware Dependent Components for appropriate citation). See the Health Monitor SW Design Description [38], section 4.1 for API definitions.

##### Protocol Manager Client library

The Protocol Manager Client library is used to interact with the Protocol Manager server which provides Bit-Oriented protocol interfaces. The Protocol Manager Client (PM) API software provides the applications access to standard ARINC protocol functions, such as ARINC 429 Williamsburg file transfer services. The DLCA-6510 Application uses the API provided by the library to access the ARINC 429 Williamsburg file transfer services to exchange messages with RIU/CMU.

This library is covered by the Protocol Manager SAS [29]. See the SDD for PMA-6000 [39], section 4.1 for API definitions.

##### Avionics Full Duplex Avionics System LAN (AFDX-ASL)

The software modules covered by the AFDX-ASL support the Avionics System LAN (ASL) network.

###### AFDX\_ASL WinSock2 API

Provides a WinSock2 based sockets API to provide applications an interface to the AFDX-ASL device driver.

The WinSock2 API is covered by the Avionics System LAN (ASL) SAS [30]. See AFDX – LAN Design [40], section 4.2 for API definitions.

###### AFDX-ASL ARINC 653 User API

Provides an ARINC 653 Inter-Partition API to provide applications an interface to the AFDX-ASL device driver.

The ARINC 653 User API is covered by the Avionics System LAN (ASL) SAS [30]. See AFDX – LAN Design [40], section 4.2 for API definitions.

##### A661 API and A661 Common Library

The ARINC 661 API is responsible for connecting and managing ARINC 661 communications to and from multiple AGS’s. The ARINC 661 common library functions include:

* Sending keep alive / heart beat messages to AGSs.
* Synchronizing DLCA-6510 widget states and data with the CDS's.
* Handling layer, widget, and exception events from AGS’s and Window Manger.
* Re-establishing and re-synchronizing broken connections from AGS’s.

The ARINC 661 common library provides the API interface for the application to communicate with ARINC 661 Graphics server using ARINC 661 protocol. This library is used to exchange messages between DLCA application and the AGS for the DLCA-6510 A661 HMI pages. Using A661 library application updates the information that is display and presented to the user by AGS.

The ARINC 661 API is covered by the ARINC 661 PSAC [31] and the ARINC 661 SAS[42]. See ARINC 661 SDD [41], for API definitions

## Overview of Applied New Technology

This subsection identifies and describes methods, or items which implement or use techniques or technology that have not been previously approved to the assurance level defined for the equipment in this PSAC.

## Software Fault Management Techniques

This subsection summarizes features, functions, or techniques used to mitigate failure effects. If applicable, descriptions of safety techniques such as fail safe, fault tolerance, redundancy management, or other techniques used to manage faults should be briefly discussed.

## Software Partitioning

This section describes features, functions, and techniques used to achieve partitioning of software functions to a level appropriate to the system safety issues. Hardware mechanisms used to support partitioning should be described, and the interaction (both control and data flow) of software and the hardware used to support partitioning should be described sufficiently for a reviewer to understand the mechanisms involved in controlling and monitoring the partitioning mechanism. Safety aspects, including software support for detection and management of partition violations, of the scheme proposed should be provided, discussed, and rationalized.

## Software Timing and Scheduling Strategies

This section describes timing and scheduling issues and strategies for management of these issues. Interactions between hardware and software should be described if applicable.

# Certification Considerations

Certification Note: DO-178C §11.1c: This section provides a summary of the certification basis, including the means of compliance, as related to the software aspects of certification. This section also states the proposed software level(s) and summarizes the justification provided by the system safety assessment process, including potential software contributions to failure conditions.

It is recommended that the following common wording be used for this paragraph in all PSACs:

This section describes the certification basis, proposed means of compliance, and the software criticality level of each function implemented in software and covered by this PSAC. It also provides the justification for the software criticality level assignment based on a safety assessment of the software and its use within the airborne system, including a description of the potential software failure conditions.

## Certification Basis and Proposed Means of Compliance

The Certification Basis is the applicable components of federal law, issue papers, Notices, or Advisories Circulars, or TSOs which apply to the equipment or system being described in this PSAC. Identification of corresponding foreign Certification Authority requirements should also be included when approval of a foreign Certification Authority (either directly or indirectly) is anticipated.

For each item listed in the Certification Basis, a corresponding means of compliance (showing that the item has been properly addressed) must be proposed in this section.

Typically a table listing the applicable TSO’s and corresponding MPS for the product(s) covered by this document is included in this section.

The certification basis for the DLCA-6510 software is DO-178C [61]. The proposed means of compliance are methods, activities, and the software life cycle data defined in this document and the associated Software Development Plan [6]. The SDP will be written in compliance with objectives defined in DO-178C [61] and the Rockwell Collins Technical Consistent Process (RC-TCP) [1] to demonstrate that the objectives defined in DO-178C [61] have been met.

The DLCA-6510 itself is not a TSO function, nor is it part of another functional TSO. The DLCA-6510 will be certified as part of the Type Certification for the target aircraft. The proposed means of compliance will be documented verification results demonstrating interoperability and compliance to applicable industry standards. AC 20-140C [72] provides interoperability, safety, and performance criteria and identifies requirements specified in industry standards that may be used as a means of compliance. For certifications through European Aviation Safety Administration (EASA) an Acceptable Means of Compliance (AMC) document is not yet available. As such, EASA currently provides a Certification Review Item (CRI) document that is unique to a specific certification. The CRI provides interoperability, safety, and performance criteria, again including references to requirements in industry standards. See section 3.1 System Functional Description for information on FANS and ATN industry standards

## Software Criticality Level

This section briefly describes the software criticality level or levels that are applicable to the software. If more than one criticality level is applicable, identify the software functions and related modules that are associated with each criticality level. The rationale for the criticality level choices should be provided. Application of all criticality levels should be justified by an applicable safety analysis assessment summary performed following the guidance of DO-178C paragraph 2.2 and a discussion of the adequacy of the resultant criticality levels provided.

The DO-178C DAL for the DLCA-6510 software will be developed and certified to Level C. The rationale for Level C is based on the assessment of the operational hazards for FANS and ATN use.

FANS operational hazards are defined in RTCA DO-306 Safety and Performance Standard for Air Traffic Data Link Services in Oceanic and Remote Airspace (Oceanic SPR Standard), October 11, 2007[63] and augmented by RTCA DO-306 Change 1 Safety and Performance Standard for Air Traffic Data Link Services in Oceanic and Remote Airspace (Oceanic SPR Standard), March 17, 2011[64].

ATN operational hazards are defined in SAFETY AND PERFORMANCE REQUIREMENTS STANDARD FOR AIR TRAFFIC DATA LINK SERVICES IN CONTINENTAL AIRSPACE (CONTINENTAL SPR STANDARD), EUROCAE ED-120 with Changes 1 and 2, October 2007 [69]

The highest FANS and ATN failure condition is classified as Major, which DO-178C [61], section 2.2.2 correlates with Level C.

Table 5‑1 DLCA-6510 Criticality Data below summarizes the Major hazards from the noted FANS and ATN industry standards.

<For Lori: This needs confirmation from safety team – DLSS-18537 has been created for future sprints. Currently the information has been kept from DLCA-6500 PSAC>

Table 5‑1 DLCA-6510 Criticality Data

| **Ref#1** | **Hazard Description** | **Classification** |
| --- | --- | --- |
| **FANS Operational Hazards** | | |
| H-CRD-7  H-IER-7  H-PR-7 | Undetected late or expired message | Major |
| H-CRD-8  H-IER-8  H-PR-8 | Undetected misdirection of a message | Major |
| H-CRD-9  H-IER-9  H-PR-9 | Undetected corruption of a message | Major |
| H-CRD-10  H-IER-10  H-PR-10 | Undetected spurious/inadvertent message delivery | Major |
| **ATN Operational Hazards** | | |
| H-ACL-3 | Undetected early delivery of a message used for separation | 3 (Major) |
| H-ACL-6 | Undetected late or expired  message used for separation | 3 (Major) |
| H-ACL-9 | Undetected misdirection of a message used for separation | 3 (Major) |
| H-ACL-12 | Undetected corruption of a message used for separation | 3 (Major) |
| H-ACL-15 | Undetected out of sequence CPDLC messages used for separation | 3 (Major) |

[[1]](#footnote-2)Hazard reference identifiers are based on the associated Data Link services, where…

ACL = ATC Clearance

CRD = Clearance Request and Delivery service  
IER = Information Exchange and Reporting service  
PR = Position Reporting service

## Non-TSO Functionality

Certification Information: This subsection identifies functionality included in the equipment which is not part of the functionality identified in the TSO’s applicable to the products covered by this document. Since this functionality is not covered by the TSO approval, it therefore is required to be covered as part of aircraft TC/STC certification activities. Additional information regarding non-TSO functionality is provided in FAA Order 8150.1D Chapter 8.

All in-flight non-TSO functionality must be identified in this section; non-TSO functionality may be items such as antenna switching or rebroadcasting data to be used by other equipment. Refer to FAA Order 8150.1D Chapter 8 for additional information including the definition of non-TSO functionality. For example functions such as Dataload and Central Maintenance Computer interfaces are not specified by the TSO MPS but are also not considered non-TSO functions because they technically need to be implemented within the TSO article.

It is typically best to identify all non-TSO functionality and then identify if software is involved in each function; this approach is recommended to avoid multiple questions from TCR’s and aircraft OEMs. However, since this section is in the PSAC, only non-TSO functions supported by software must be identified.

Non-TSO functionality should be identified in System documentation, but may not be easy to identify. Thus LRU hardware/software engineers should work with systems engineers to identify Non-TSO functionality. Identify the software considerations associated with each non-TSO function.

The following text provides examples to assist project teams develop the text applicable to their project:

The equipment addressed in this document does not contain non-TSO functionality as described in FAA Order 8150.1D [12] chapter 8-1 as follows:

A non-TSO function is a function that is not covered by a TSO-approved MPS, does not support or affect the hosting article’s TSO function(s), and could technically be implemented outside of the TSO article.

All software will be developed, verified, and evaluated in the same manner as described in this document.

OR

The equipment addressed by this document includes non-TSO functionality which is identified in the following sub-paragraphs. Non-TSO functionality is identified based on the definition from FAA Order 8150.1D [12] chapter 8-1 as follows:

A non-TSO function is a function that is not covered by a TSO-approved MPS, does not support or affect the hosting article’s TSO function(s), and could technically be implemented outside of the TSO article.

All TSO and non-TSO software will be developed, verified, and evaluated in the same manner as described in this document.

DLCA-6510 itself is not a TSO function, nor is it part of another functional TSO. The DLCA-6510 will be certified as part of the Type Certification for the target aircraft.

<For Lori: This needs confirmation from Lori/Hatem>

### Non-TSO Function #1

<Details go here>

### Non-TSO Function #2

<Details go here>

…

## Outsource / Offshoring Entity and Risk Summary

This section should identify the entity/entities to which engineering will be outsourcing / offshoring work. This should include company name, city / country, and summarize the OORA score for that entity. This section should reference where the full OORA scorecard is archived. In addition, this subsection should identify and describe at a very high level any outsourcing / offshoring activities performed on this project or state that no outsourcing/offshoring activities were performed on this project. If outsourcing/offshoring was performed a reference to see Section 10 for additional details should be provided.

The DLCA-6510 team will include offshore and outsourced partners to meet the Rockwell Collins business and program/project goals.

The DLCA-6510 team will use the GETC-India as an offshore partner. GETC-I is a Rockwell Collins subsidiary based in India with locations in Hyderabad and Bangalore. Off-shore partners’ responsibilities included requirements definition, software development and verification. An Outsourcing/Offshoring Risk Assessment (OORA) was completed for GETC-India in accordance with Risk Assessment and Oversight for Outsourcing or Offshoring Activities involving Civil Certification, HRC-ENG-P-016 [4]. The total score of the assessment indicates a Low Risk, and this risk is applicable to all projects which call out this PSAC, and which may use GETC-I Offshore personnel. The recommendations and guidance called out for a Low Risk project in HRC-ENG-P-016 [RC\_OORA] will be followed. The full OORA scorecard for GETC-I is included below:

<SVN Path: TBD>

<For Lori: Currently OORA is shared here. After the approval, this will be moved to SVN and the path will be updated>

<https://rtxusers.sharepoint.us/:f:/r/sites/GS-DataLinkTeamDrive-COL/Shared%20Documents/DLCA-6510/Planning%20and%20Process%20Documents?csf=1&web=1&e=zMScGW>

The DLCA-6510 team will also use onsite contractors from HCL Technologies, an Indian engineering firm. The onsite HCL contractors were utilized for development and verification.

The Rockwell Collins USA team has primary responsibility for meeting all the objectives of the software planning, requirements definition, software development, verification, and developmental configuration management processes.

Please refer section 10 Outsourcing/Offshoring & Supplier Oversight for more details.

# Software Life Cycle

Certification Note: DO-178C §11.1d: This section defines the software life cycle to be used and includes a summary of each of the software life cycle processes for which detailed information is defined in their respective software plans. The summary explains how the objectives of each software life cycle process will be satisfied, and specifies the organizations to be involved, the organizational responsibilities, and the system life cycle processes and certification liaison process responsibilities..

It is recommended that the following common wording be used for this paragraph:

This section provides a summary of the software life cycle processes that will be used for this project. The summary identifies how the objectives of each life cycle process will be satisfied. The summary also specifies the organizations to be involved, the organizational responsibilities, the system life cycle processes, and certification liaison process responsibilities.

As a product line application, the DLCA-6510 is planned for reuse across multiple aircraft platforms. Future versions will be developed by adding new requirements, and/or modifying, and/or removing existing requirements from the previous baseline version. For each planned deliverable version, an iterative software process model will be followed, where a series of labeled builds will be developed, such that each successive build is closer to satisfying the final requirements for the planned deliverable version. Each of these intermediate builds will include activities from various life cycle processes. Not all life cycle processes will be performed for each intermediate build. However, all life cycle processes will be performed prior to certification application and/or the final production build for a given deliverable version. Regression analysis may be used if necessary to provide assurance that all appropriate life cycle activities have been performed.

The specific details of the software life cycle for DLCA-6510 may be found in the Data Link Products Software Development Plan (SDP) [6].

There will be significant differences in this section for new developments or major modifications versus derivative, maintenance, or other minor developments.

## Description of the Software Life Cycle Processes

This section describes the elements of the software life cycle process.

The entire software life cycle for the DLCA-6510 software will conform to the process documented in DO-178C [61] as well as the Rockwell Collins Technical Consistent Process [1]. The specific processes followed on this project are documented in more detail in the SDP [6].

There will be multiple internal/intermediate builds that will be part of the iterative development process. These intermediate builds will be used to support internal development and verification.

The software development will be performed using both host and target platforms. The host platform will be a Windows PC that utilizes a host compiler to generate host builds and a cross compiler to generate builds for the target hardware. The host environment also includes a simulation tool that enables the host build to run in the host environment.

<For Lori: DLSS-18372 planned for future sprints. This is to include possibility of system rig as test environment>

### Planning

This section describes planning activities performed on this project.

The software development plan and the supporting integral activities for the DLCA-6510 are defined in this PSAC, the Software Development Plan [6], the SCM plan [3], and the Design Quality Assurance Plan for Hardware, Software and System Development [2]. The transition criteria, inter-relationships, and sequencing among these processes are defined in the SDPs [6] (see section 5.1).

Project plans are developed to meet the following objectives:

* Provide process activity definitions.
* Define transition criteria, inter-relationships, and sequencing among processes.
* Define the software life cycle.
* Define software development standards.

Reviews of the project plans are completed per the SDP [6]. Any significant changes to the software plans will result in the updating of one or more of the software planning documents discussed in this section. DAC Representative will review these changes to ensure they are coordinated. If the PSAC is revised it will be resubmitted for approval by the certification authorities

#### Planning Approach on Future Minor Changes

Recommend the following be included:

The planning phase for minor software changes to the product(s) covered by this PSAC will typically be addressed with the creation of a Planning Change Impact Analysis which is submitted to the Rockwell Collins Certification Department for review to evaluate if the planned changes meet the criteria for a minor TSO change. The Planning CIA addresses the criteria identified in AC 00-69.

### Software Development Process

Software Development Plan [6] define the software development processes to:

* Analyze system/subsystem requirements to define software requirements.
* Analyze software requirements to define the software architecture and low-level requirements.
* Design and code the software to implement the software architecture and low-level requirements.
* Verify the outputs of the software requirements process, design process, and coding process by having peer reviews.
* Integrate the software on the host platform and target hardware.

The processes will produce development artifacts that are accurate and consistent, verifiable, traceable to the system requirements, and compatible with the target computer.

Process Deviations/Additions contains a list of known deviations/additions to some of the detailed process steps in the SDP [6] for which this project intends to follow. It is expected that a future version of the SDP will incorporate these changes. The Software Accomplishment Summary will document any changes to this approach.

#### Software Development Environment

The Software Development Plan [6] provides a detailed description of the software development environment and the planned tools to be used for each of the life cycle processes. The Software Developers User’s Guide [43] provides a description on using the tools called out at each life cycle process.

Reference the SDP [6] Section 4.3 for a complete listing of the planned software development tools for PPC-based targets (DLCA-6510 is targeted for the PPC) and the planned software verification tools for IMA products (DLCA-6510 is an IMA product).

The following tools will **not** be used for DLCA-6510:

* LDRA coverage analyses tool
* Data Link Tester

DO-178C [61] Section 12.2.1 (Determining if Tool Qualification is Needed) provides guidance for when a development or verification tool is required to be qualified.

“Qualification of a tool is needed when processes of this document are eliminated, reduced or automated by the use of a software tool without its output being verified…”

Tools used for this development that meet the above criteria:

* VISTA is a multi-process simulation environment suitable for use in the development and testing of avionics software.
* VectorCAST is a tool used for Structural Coverage Analysis.
* Vision Framework is a tool used for testing of avionics HMI.

#### Software Requirements Process

This section describes the requirements capture activities on this project and describes the requirements changes from the baseline, if applicable.

The development of software high-level requirements (HLRs) will be performed in accordance with the process defined in the Software Development Plan [6] section 6.1.

The DLCA-6510 software high level requirements will be captured in the High Level Software Requirements Specification (SRS) for the Pro Line Fusion DLCA-6510 Data Link Communications Application [9].

Note that the DLCA-6510 product is largely driven by detailed industry interoperability specifications, where these specifications target specific aircraft software applications (e.g. CPDLC) and contain detailed requirements that are already decomposed to govern specific application behavior. In these cases the industry specifications, while likely having the detail required to be treated as high-level (or even low-level) software requirements, are not considered requirements but instead are used to justify certain derived requirements. Tracing may exist between DLCA requirements and industry requirements, but this tracing is in place only to show compliance to industry specifications.

##### Requirements Configuration Management

This section primarily discusses Configuration Management considerations during development activities. However it should also mention the planned formal CM, typically SCL release. The planned formal CM should identify the format of the data released to SCL, typically in PDF document format. A reference to the general Configuration Management process, section 6.3, is expected to avoid duplication of information.

The Software Configuration Management process is described in detail in the Software Development Plan [6], section 8.

##### Requirements Reviews

This section describes the requirements and associated traceability analysis activities. A reference to the general peer review process, such as section 6.2, is expected; typically a common peer-review process is applicable. However, some projects use different processes for different artifacts. Items expected to be identified include, but are not limited to: will reviews be performed only from the native tool (JAMA/DOORS) or are exported documents reviewed or both; if reviews are performed from the native tool how will the exported document be evaluated for completeness and accuracy, if no export is performed how will data meet the FAA data retrieval expectations, etc. Typically only the Software Requirements to System Requirements is evaluated during a Requirements review as Design/Code and Tests may not be available. Explain how bi-directional traceability expectations are planned to be addressed.

The purpose of this activity is to verify the software high-level requirements have been developed in accordance with the process defined in the SDP [5] section 7.4.1 and in accordance with the objectives defined in DO-178C.

The verification method used for this activity will be the Peer Review method [11].

Software high-level requirements verification may begin once the relevant system requirements allocated to software have been reviewed and approved, and the software high-level requirements to be verified have been placed under developmental configuration control.

For small scope change requests, this activity may be done in conjunction with verifying the relevant system requirements.

The following is a summary of activity tasks that will be performed for this process. The Software Development Plan provides the detailed information for each of these activity tasks.

* Task 1: Conduct Peer Review

This activity is considered complete when all findings in the peer review have been closed and the peer review itself is closed.

#### Software Design Process

This section describes the design activities on this project and describes the design changes from the baseline, if applicable. This section should also describe verification/validation activities on the design and associated traceability for this project.

The development of the software architecture, detailed design, and low-level requirements (LLRs) will be performed in accordance with the process defined in the Software Development Plan [6] section 6.2. Unified Modeling Language (UML) will be used as applicable to aid in documenting the design. UML will not be used for anything other than documentation purposes.

The software architecture and detailed design will be documented in the Software Design Document for DLCA [15] and will include descriptions of the high-level software architecture, Input/Output interfaces, data flow and control, and pertinent design decisions with associated rationale.

The software low-level requirements will be captured in the project Software Requirements Specifications [10][11][12][13].

The IOCF [5] will also be captured as part of the design process.

The SDD [15] contains a thorough description, design data, of the class definitions. This can be found in the header files for the classes. All low level requirements will trace to entries in this document. An accompanying SDD [15], in a Microsoft word document, covers the following topics:

* Software Product Design Decisions; and
* Software Architecture and High-Level Design; and
* Software Detailed Design.

The SDD [15] describes each software unit that satisfies both the high and low-level software requirements.

##### Design Configuration Management

This section primarily discusses Configuration Management considerations during development activities. However it should also mention the planned formal CM, typically SCL release. The planned formal CM should identify the format of the data released to SCL, typically in PDF document format. A reference to the general Configuration Management process, section 6.3, is expected to avoid duplication of information.

The Software Configuration Management process is described in detail in the Software Development Plan [6], section 8.

##### Design Reviews

This section describes the design and associated traceability analysis activities. A reference to the general peer review process, such as section 6.2, is expected as typically a common peer-review process is performed. However, some projects use different processes for different artifacts. Items expected to be identified include but are not limited to: will reviews be performed only from the native tool (JAMA/DOORS) or are exported documents reviewed or both; if reviews are performed from the native tool how will the exported document be evaluated for completeness and accuracy, if no export is performed how will data meet the FAA data retrieval expectations, etc. Explain how bi-directional traceability expectations are planned to be addressed.

The software architecture and low-level requirements will be verified to meet the required objectives of DO-178B using the Peer Review method as described in the Software Development Plan [6] section 7.4.2.

#### Software Implementation/Coding Process

This section describes the implementation/coding activities on this project and describes the extent of the changes from the baseline, if applicable.

The development of the software source code will be performed in accordance with the process defined in the Software Development Plan [6] section 6.3. The DLCA-6510 software will be implemented using the C++ programming language.

The software source code will be formally released in the DLCA-6510 CPCIs [16],[17], and [18]

##### Implementation/Coding Configuration Management

This section primarily discusses Configuration Management considerations will be addressed during development activities. However it should also identify the planned formal CM, typically SCL release. The planned formal CM should identify the format of the data released to SCL, typically in CPCI documents with executables identified in Software Deliverable drawings. A reference to the general Configuration Management process, section 6.3, is expected to avoid duplication of information.

The Software Configuration Management process is described in detail in the Software Development Plan [6], section 8.

##### Implementation/Coding Reviews

This section describes the implementation and associated traceability analysis activities. Often a one-to-one relationship exists between code and design and therefore the requirements to design traceability is equivalent to code to requirements. A reference to the general peer review process, such as section 6.2, is expected as typically a common peer-review process is performed. However, some projects use different processes for different artifacts. Items expected to be identified include but are not limited to: will reviews be manual or will tools be used, etc. Explain how bi-directional traceability expectations are planned to be addressed.

The software source code will be verified to meet the required objectives of DO-178B using the Peer Review method as described in the Software Development Plan [6] section 7.4.3.

#### Software Integration Process

<For Lori: This subsection added based on DLCA-6500 PSAC and it was not part of the template>

The software integration will be performed in accordance with the process defined in the Software Development Plan [6] section 6.4.

The DLCA-6510 Executable Object Code [48] will be formally released to the Software Control Library (SCL).

##### Integration Configuration Management

This section primarily discusses Configuration Management considerations will be addressed during development activities. However it should also identify the planned formal CM, typically SCL release. The planned formal CM should identify the format of the data released to SCL, typically in CPCI documents with executables identified in Software Deliverable drawings. A reference to the general Configuration Management process, section 6.3, is expected to avoid duplication of information.

The Software Configuration Management process is described in detail in the Software Development Plan [6], section 8.

##### Integration Reviews

This section describes the implementation and associated traceability analysis activities. Often a one-to-one relationship exists between code and design and therefore the requirements to design traceability is equivalent to code to requirements. A reference to the general peer review process, such as section 6.2, is expected as typically a common peer-review process is performed. However, some projects use different processes for different artifacts. Items expected to be identified include but are not limited to: will reviews be manual or will tools be used, etc. Explain how bi-directional traceability expectations are planned to be addressed.

The software integration process will be verified to meet the required objectives of DO-178C using the Test method as described in the Software Development Plan [6] section 7.4.4.

### Software Verification Process

This section provides a summary of the software verification processes used to verify the results of the software development processes. The Software Verification User’s Guide [44] provides additional information on the verification environment.

#### Test Case

This section describes the test activities on this project and describes the test case and procedure changes from the baseline, if applicable.

The purpose of this activity is to develop the software verification test cases. The software verification test cases describe, at a high level, the actions (analysis, inspection, or test) that will be performed to verify that the software satisfies the software high-level and low-level requirements. To accomplish this, the test cases will be developed from the software low-level requirements and the associated software high-level requirements.

This activity may begin once the relevant software high-level and low-level requirements are sufficiently understood. Normally, this means the relevant software high-level and low-level requirements have been documented, placed under developmental configuration control, and are ready for peer review. Activity tasks started prior to this are at risk of rework and should be coordinated with the Project Engineer.

The development of software test cases will be performed in accordance with the process defined in the Software Development Plan [6] section 7.4.5.

##### Test Configuration Management

This section primarily discusses Configuration Management considerations will be addressed during development activities. However it should also identify the planned formal CM, which may be an SCL release CPCI(s) or test information remains in developmental CM. If test artifacts remain in developmental CM, information should be provided to explain how this data is being protected from accidental deletion or modification for the life of the product. Archive of test cases and procedures is most critical, test results could be recreated if requested (although this creates a costly and time consuming risk), but test cases and procedures need to be protected. Developmental CM does not provide protection against human error. A reference to the general Configuration Management process, section 6.3, may be appropriate to avoid duplication of information.

The Software Configuration Management process is described in detail in the Software Development Plan [6], section 8.

##### Test Case Reviews

This section describes the tests and associated traceability analysis activities performed on requirements. A reference to the general peer review process, such as section 6.2, is expected as typically a common peer-review process is performed. Explain how is bi-directional traceability expectations are planned to be addressed.

The purpose of this activity is to verify the software test cases have been developed in accordance with the process defined in the SDP and in accordance with the objectives defined in DO-178C.

The software verification test cases will be verified to meet the required objectives of DO-178B [61] using the Peer Review method and the Requirements Based Test Coverage Analysis method as described in the Software Development Plan [6], section 7.4.6.

This activity is considered complete when all findings in the peer review have been closed and the peer review itself is closed. Generally, test case and test procedure (section 6.1.3.2) will be reviewed together instead of reviewing them separately.

#### Test Procedure

This section describes the test activities on this project and describes the test case and procedure changes from the baseline, if applicable.

The objective of this activity is to provide step-by-step instructions for how each test case is to be set up and executed, how the test results are evaluated, and the configuration of the test environment used to execute the tests.

This activity may begin once the relevant test cases are sufficiently understood. Normally, this means the relevant test cases have been documented, placed under developmental configuration control, and are ready for peer review. Activity tasks started prior to this are at risk of rework and should be coordinated with the Project Engineer.

This activity is considered complete when all of the activity tasks have been successfully completed and the corresponding outputs and/or portions thereof have been placed under developmental configuration control.

The development of software test procedures will be performed in accordance with the process defined in the Software Development Plan [6] section 7.4.7.

##### Test Procedure Configuration Management

This section primarily discusses Configuration Management considerations will be addressed during development activities. However it should also identify the planned formal CM, which may be an SCL release CPCI(s) or test information remains in developmental CM. If test artifacts remain in developmental CM, information should be provided to explain how this data is being protected from accidental deletion or modification for the life of the product. Archive of test cases and procedures is most critical, test results could be recreated if requested (although this creates a costly and time consuming risk), but test cases and procedures need to be protected. Developmental CM does not provide protection against human error. A reference to the general Configuration Management process, section 6.3, may be appropriate to avoid duplication of information.

The Software Configuration Management process is described in detail in the Software Development Plan [6], section 8.

##### Test Procedure Reviews

This section describes the tests and associated traceability analysis activities performed on requirements. A reference to the general peer review process, such as section 6.2, is expected as typically a common peer-review process is performed. Explain how is bi-directional traceability expectations are planned to be addressed.

The software verification test procedures will be verified to meet the required objectives of DO-178B using the Peer Review method as described in the Software Development Plan [6] section 7.4.8.

#### Verification Testing

This section describes the test activities on this project and describes the test case and procedure changes from the baseline, if applicable.

The objectives of this activity are to demonstrate that the software satisfies its high-level and low-level requirements and to demonstrate with a high level of confidence that errors which could lead to unacceptable failure conditions have been removed.

Software verification testing is performed to demonstrate functional correctness of the software product using requirements based testing.

Software testing activities may begin once the applicable inputs, or portions thereof, have been placed under developmental configuration control, reviewed and approved.

This activity is considered complete when all outputs and/or portions thereof have been placed under developmental and/or production configuration control.

Control coupling analysis will be performed in conjunction with Structural Coverage Analysis (SCA), where statements that were not covered will be analyzed to ensure that no adverse effects on the logical control of execution would occur if the statements were to execute. Likewise, data coupling analysis will be performed in conjunction with SCA, where statements that were not covered will be analyzed to ensure that no adverse effects on shared data would occur if the statements were to execute.

The software verification testing (requirements based testing) will be performed in accordance with the process defined in the Software Development Plan [6] section 7.4.9.

<For Lori: DLSS-18372 has been created to allow IPS or EDS execution for verification instead of repeating the execution on both environment every time>

##### Test Configuration Management

This section primarily discusses Configuration Management considerations will be addressed during development activities. However it should also identify the planned formal CM, which may be an SCL release CPCI(s) or test information remains in developmental CM. If test artifacts remain in developmental CM, information should be provided to explain how this data is being protected from accidental deletion or modification for the life of the product. Archive of test cases and procedures is most critical, test results could be recreated if requested (although this creates a costly and time consuming risk), but test cases and procedures need to be protected. Developmental CM does not provide protection against human error. A reference to the general Configuration Management process, section 6.3, may be appropriate to avoid duplication of information.

The Software Configuration Management process is described in detail in the Software Development Plan [6], section 8.

##### Test Reviews

This section describes the tests and associated traceability analysis activities performed on requirements. A reference to the general peer review process, such as section 6.2, is expected as typically a common peer-review process is performed. Explain how is bi-directional traceability expectations are planned to be addressed.

The software verification test results will be verified to meet the required objectives of DO-178B [61] using the Peer Review method and the Structural Coverage Analysis method as described in the Software Development Plan [6] section 7.4.10

## Peer Review Process

This section describes the general peer-review process and any differences if more than one approach is used on the project.

All work done will be performed per the SDP [6] and will be reviewed per the Peer Review process [26] using the Data Link Products Peer Review Checklists [13].

## Configuration Management Process

This section describes both the general developmental configuration management processes and also summarizes the general final configuration management process applicable to this project. In addition. When formal Configuration Management is planned to occur should be identified. Please note that developmental Configuration Management typically only meets CC2 criteria described in DO-178B/C Table 7-1; providing artifacts out Rockwell Collins may require additional activities such as SCL snapshot releases. This section should also identify configuration management for software tools.

In general, all project personnel that contribute to the development of formal project artifacts are responsible for configuration management of those artifacts. In this context, a formal project artifact is one that is a deliverable to a consumer outside the Data Link organization, and/or is a required artifact as defined by the processes in the SDP.

The Software Control Library (SCL) provides production release configuration management as described in the Software Configuration Management Plan [2].

Software Quality Assurance personnel are responsible for auditing the SCM process.

Subversion will be used to maintain developmental configuration control of all project artifacts, except for requirements, which are configuration controlled within the Jama tool. Refer to the *Requirements Standards and Processes for the Commercial Systems Data Link Organization Using Jama, RCPN 946-7012-001* [11] for further details on Jama process.

JIRA will be used to record and control additions and changes to project artifacts by use of Change Requests (CRs). Refer to Change Request and CCB Process for the Commercial Systems Data Link Organization Using JIRA , RCPN 946-8189-002 [6] for further details.

The Software Configuration Management process is described in detail in the Software Development Plan [6], section 8.

### Problem Reporting / Change Request Process

This section summarizes the problem reporting / change request process. As DO-178B/C refer to Problem Reports and Rockwell Collins uses the term Change Requests, it is recommended to explain the renaming in this section.

DO-178C Problem Reporting (PR) process is called a Change Request (CR) process by Rockwell Collins and on this project. The JIRA tool will be used to record and track Change Requests for any item subject to configuration management. Problem Reporting will be managed by creating Change Requests. Refer to Change Request and CCB Process for the Commercial Systems Data Link Organization Using JIRA , RCPN 946-8189-002 [6] for details on generating and tracking Change Requests.

All software project personnel will have access to JIRA. Change Requests are entered as they are discovered throughout the product’s life cycle. Change requests are controlled and closed by the project Change Control Board (CCB).

Open/deferred problem reports or change requests will be categorized based on the guidance from DO-248C [5] DP#9 which is equivalent to EASA CM-SWCEH-002 [11] chapter 16.

## Software Quality Assurance Process

This section summarizes the software quality assurance plan applicable to this project and references the detailed SQAP. A summary of the planned Software Conformity Inspection is to be included in this section.

The Software Quality Assurance Plan is described in the Software Development Plan [6], section 9 and in the Design Quality Assurance Plan for Hardware, Software and System Development [2].

## Organizational Responsibilities

This section should describe the organization used for development, testing, configuration management, and quality assurance. Roles and responsibilities should be clearly defined.

Recommended wording for this section for the Quality Assurance role is as follows:

### Staffing

This section summarizes the problem reporting / change request process. As DO-178B/C refer to Problem Reports and Rockwell Collins uses the term Change Requests, it is recommended to explain the renaming in this section.

To optimize efficiency, the Data Link team as a whole is comprised of a multi-disciplinary integrated product development team, composed of systems, software, and verification engineers. Over time, the size and composition of the teams will change to meet the needs of the various Data Link projects. Team members will be expected to be flexible and fill multiple roles as required. A high level of cross training within the team will be used to achieve this flexibility. Note that independence for reviews and testing will be an important factor when team member roles are adjusted within the team.

Each team will accept full responsibility for all aspects of the product being developed by the team, including certification artifacts, and will have broad authority to manage that development. Teams will be encouraged to proactively identify potential issues before they arise and manage them independently. Issues that impact other teams will be raised to either the software leadership team or project leadership team as necessary.

As need dictates, the Data Link team may utilize contract engineers to meet the project staffing needs. In such cases, these contract engineers will be full team members with access to the same tools, files, documentation, and information necessary to accomplish their assigned tasks

### Roles and Responsibilities

Data Link team has primary responsibility for meeting all the objectives of the software planning, development, verification, and development configuration management processes. The team is also responsible for providing support during the Software Quality Assurance (SQA) process activities.

The Rockwell Collins Design Assurance Center (DAC) for Quality is responsible for the Software Quality Assurance process and is independent from the Data Link team. DAC representatives are also responsible for monitoring the activities of the Data Link team during their life cycle process activities. Refer to the *Commercial Systems Software Quality Engineering Assurance Plan* [7].

Software Quality Assurance role is fulfilled by engineers under Design Assurance Center (DAC) management, which is organizationally independent from product development engineering. The DAC is part of the Engineering Infrastructure & Integrity (EI&I) directorate in the Engineering & Information Technology organization.

DAC Engineers with support from DAC Representatives fulfill the Software Quality Assurance objectives as described in DO-178()

The following roles and responsibilities are provided to further define the Data Link team.

**Life Cycle Value Stream Manager (LCVSM)** – The LCVSM has the overall life cycle responsibility for a given Data Link product with respect to the product’s cost and schedule in all aspects including development, manufacture, and deployment into service. The LCVSM is responsible to the customer, senior management, and the program team for overall cost, schedule, and quality performance of the Data Link products.

**Technical Product Manager (TPM)** – The TPM has the overall responsibility for the technical performance of a given Data Link product. The TPM functions as the technical advisor on Data Link products by providing technical direction as required and overall responsibility for development activities.

**Group Manager / Engineering Manager** – The Group Manager / Engineering Manager is responsible for providing resources, establishing priorities, resolving conflicts, and providing career development guidance. The Group Manager / Engineering Manager works closely with Project Engineers to ensure that projects are progressing according to plan, within budget, and on schedule.

**Systems Engineer** – The Systems Engineer has the primary responsibility for defining and allocating the system requirements, creating and maintaining the IOCF document, developing and running System Integration tests on the Systems Integration Rig, and is responsible for tracking cost, schedule, risk and status for the system engineering functions.

**Integration and Test Engineer** – The Integration and Test Engineer has responsibility for the system integration and performance testing. This is a System Engineering function.

**Hardware Engineer** – The Hardware Engineer defines the hardware requirements, hardware design, and hardware specification for the various hardware components being designed in the system. Depending on the project size and complexity related to hardware design, there may be a lead hardware engineer for each major component of the system.

**Project Engineer (PE)** – The PE is responsible for software project detailed planning, including development of the project schedule and work packages that will formulate the project’s baseline for measuring earned value. Throughout the development cycle, the PE tracks earned value against the plan and adjusts resources as necessary to maintain scope, schedule, and cost of all work packages. Depending on the project size and complexity, there may be individual domain PE’s (e.g. System, Hardware, Software) for a single project.

**Software Engineer** – The Software Engineer defines the software high-level requirements, software architecture, detailed design, low-level requirements, and source code. The software engineer performs informal unit testing of code units, integrates the software within a software build for the target hardware, and either assists or leads in the investigation and resolution of problem reports. The software engineer may be called upon to assist the verification engineer to help identify test methods, and leads the SCA process by generating instrumented software builds, performs uncovered code analysis, and generates summary coverage reports.

**Design Assurance Center (DAC) Representative** – The DAC is comprised of DAC Engineers, Subject-Matter-Experts (SMEs), DAC Auditors, and Technical Leads. The primary responsibility of the DAC representatives is to assure that the project is aware of, and compliant with, the quality standards of the operating organizations and the customer. The DAC is independent of the project team and does not participate in the actual software development. The DAC’s primary means of gaining assurance is by auditing the process to verify the output artifacts of the various development phases.

**Engineering Project Assistant (EPA)** – The EPA assists the development team and the PE by facilitating project meetings, peer reviews, and performs general maintenance of engineering data, including administrative control of engineering repositories and process tools, maintaining project management artifacts and generating reports related to scheduling, budgeting, staffing, risk management tracking, etc.

**Software Architect** – The software architect provides the framework and foundation for software engineering to design and develop the various Data Link software products. The software architect leads or participates in activities such as trade studies, white papers, market trends, competitive landscape, customer proposals, and product roadmaps to steer the development of Data Link software products.

**Verification Engineer** – The verification engineer performs the verification activities including the development of high-level and low-level requirements-based test cases and test procedures, dry run test execution, formal test execution, and development of the formal Software Verification Procedures and Results (SVPR) document.

NOTE: A member of the project team may fulfill more than one role on the project.

## Certification Liaison

The objective of the certification liaison process is to establish communication and understanding between the applicant and the certification authority throughout the software life cycle to assist the certification process.

This subsection should describe the expected interaction between the applicant and the certification authority including the proposed use of FAA delegated representatives. Following is the guidance from prior templates which is acceptable as a starting point:

The Certification and Regulatory Coordinator is responsible for Certification Liaison process activities. The Certification and Regulatory Coordinator provides advice to the project and evaluates the preliminary documents. The Certification and Regulatory Coordinator is also responsible for coordinating qualification activities with the regulatory authority and the project. The Certification and Regulatory Coordinator is involved in TSO Compliance Representative (TCR) assignments. TCR selection is based on individual TCR capabilities, experience, availability/workload, and accessibility to the design team/artifacts.

A summary of the SOI process should be included. When discussing the SOI process it is recommended that both the USA and international terms be used for completeness: Planning Review (SOI-1), Development Review (SOI-2), Verification Review (SOI-3) and Final Review (SOI-4).

This PSAC will be submitted to the applicable Program Office for transmittal to the OEM for approval.

SOI activity will be performed throughout the initial project development and verification processes. Following initial equipment approval, subsequent SOI activities will be focused on changed areas only. For example, SOI 1 will only be repeated for follow-on projects if the planning documents previously reviewed are changed. Likewise, SOI 2 and SOI 3 will only examine the changed areas of the design and previous SOI 2 and SOI 3 reviews will remain valid.

FAA Order 8110.49A, Chapter 2 “Software Review Processes” [65] provides objectives of the software review process and guidance on the certification authority involvement.

FAA Order 8110.49A, Appendix A [65] provides a Level of Involvement worksheet. Provided in Appendix G of this document are the completed self-assessment worksheets for DLCA. The Total Score Result (TSR) was 142. Based on the TSR score result of 142 and DLCA at DAL C Software level, the level of certification authority involvement in the DLCA software is determined at LOW level of required involvement.

<For Lori: This needs to be revisited if the number gets impacted as we are going to certify for DO-178C. Above number is based on DLCA-6500 DO-178B certification>

# Software Life Cycle Data

Certification Note: DO-178C §11.1e: This section specifies the software life cycle data that will be produced and controlled by the software life cycle processes. This section also describes the relationship of the data to each other or to other data defining the system, the software life cycle data to be submitted to the certification authority, the form of the data, and the means by which the data will be made available to the certification authority.

This section identifies the software life cycle data that will be produced and controlled by the software life cycle processes. This section further describes the relationship of the data to each other or to other data defining the system, the software life cycle data to be submitted to the certification authority, the form of the data, and the means by which this life cycle data will be made available to the certification authority. If applicable, data to be submitted by or from a sub tier supplier in support of certification should also be identified, and the manner in which this data will be used in support of the overall certification effort defined.

The objective of this section is to provide a summary of the software life cycle data to be produced and controlled during the Product Line DLCA-6510 software development, which meets the Software Life Cycle Data objectives of the Plan for Software Aspects of Certification found in DO‑178C [61], section 11.1.e.

Table 7‑1 DLCA-6510 Life Cycle Data Items (below) identifies the software life cycle data items that will be generated as part of the DLCA-6510 development. All software life cycle data items will be kept under configuration control at Rockwell Collins. The numbers in the DO-178C column are the numbers of the equivalent software life cycle data items as outlined in section RTCA DO-178C [61].

The “Submit” columns lists items that will be submitted as part of this certification package or available for review at a Rockwell Collins facility.

* S – Submitted as a part of the certification package
* A – Available for review at a Rockwell Collins Facility

Table 7‑1 DLCA-6510 Life Cycle Data Items

| DO-178C Life Cycle Data | Rockwell Collins Equivalent | CCM Submit | AFD-37X0 Submit |
| --- | --- | --- | --- |
| 11.1 Plan for Software Aspects of Certification | Plan for Software Aspects of Certification for the Data Link Communications Application (DLCA-6510) (this document) | S | S |
| 11.2 Software Development Plan | Software Development Plan for the Commercial Systems Data Link Products [6] | A | A |
| 11.3 Software Verification Plan | Software Development Plan for the Commercial Systems Data Link Products [6] | A | A |
| 11.4 Software Configuration Management Plan | Software Configuration Management Plan [3] | A | A |
| 11.5 Software Quality Assurance Plan | Design Quality Assurance Plan for Hardware, Software and System Development [2] | A | A |
| 11.6 Software Requirements Standards  11.7 Software Design Standards  11.8 Software Code Standards | Software Development Plan for the Commercial Systems Data Link Products [6]  C++ Coding Standards [7] | A  A | A  A |
| 11.9 Software Requirements Data | High Level SRS for the Pro Line Fusion DLCA-6510 Data Link Communications Application [9]  CPCI for DOORs Archive [49] | A  A | A  A |
| 11.10 Design Description | SRS for the Data Link Communications Application (DLCA) Future Air Navigation System (FANS-1/A) [10]  SRS for Common DLCA System Services[11]  SRS for Aeronautical Telecommunication Network Context Management (CM) And Controller Pilot Data Link Communication (CPDLC) [12]  Software Requirement Specification (SRS) for HMI [13]  Software Design Document (SDD) for the DLCA-6510 Data Link Software Component Design [15]  CPCI for DOORs Archive [49] | A  A  A  A  A  A | A  A  A  A  A  A |
| 11.11 Source Code | CPCI for the Human Machine Interface (HMI) DLCA-6510[16]  CPCI for the DLCA-6510 Core Software Library [17]  CPCI for the DLCA-6510 Message Library [18]  CPCI for the DLCA-6510 XML I/O Configuration File [19]  CPCI for the DLCA-6510 XML ATN Default Addresses [20]  CPCI for the DLCA-6510 XML General Config [21]  CPCI for the Support Files [32] | A  A  A    A  A  A  A  A | A  A  A    A  A  A  A  A |
| CPCI for PCT Data [56]  CPCI for the DLCA-6510 Media Set [57]  CPCI for the DLCA-6510 Electronic Nameplate [52] | N/A | A  A  A |
| CPCI for the A661 VAPS Model [51]  CPCI for the A661 Definition Files [33] | A  A | N/A |
| 11.12 Executable Object Code | Software Deliverable for the DLCA-6510 Message Library [53]  Software Deliverable for the DLCA-6510 [48] | A  A | A  A |
| Software Deliverable for DLCA-6510 Media Set [58] | N/A | A |
| 11.13 Software Verification Cases and Procedures | SVPR for the DLCA-6510 [22]  CPCI for the DLCA-6510 SVPR [54] | S  S | S  S |
| 11.14 Software Verification Results | Software Verification Procedures and Results (SVPR) for the DLCA-6510 [22]  CPCI for the DLCA-6510 SVPR [54] | S  S | S  S |
| 11.15 Software Life Cycle Environment Configuration Index  a. Identify the software life cycle environment hardware and its operating system software.  b. Identify the software development tools  c. Identify the test environment used to verify the software product  d. Identify qualified tools and their associated tool qualification data. | Plan for Software Aspects of Certification for the Data Link Communications Application (DLCA-6510) (this document). Covers RTCA DO-178C [61] objectives 11.15.a, 11.15.b, 11.15.c, 11.15.d.  Software Development Plan [6]. Covers RTCA DO-178C [61] objectives 11.15.b, 11.15.c.  Software Verification Procedures and Results for the DLCA-6510 [22]. Covers RTCA DO-178C [61] objectives 11.15.c.  CPCI’s for the DLCA-6510 [16], [17], [18], [19], [20], [21] [32], [33]. Covers RTCA DO-178C [61] objectives 11.15.a, 11.15.b. | A  A  S  A | A  A  S  A |
| 11.16 Software Configuration Index | Software Configuration Index (SCI) for the Data Link Communications Application (DLCA-6510) [55]  Top Level Drawing [24] | S  A | S  A |
| 11.17 Problem Reports | Problem Reports1 | A | A |
| 11.18 Software Configuration Management Records | SCM Records2 | A | A |
| 11.19 Software Quality Assurance Records | SQA Records3 | A | A |
| 11.20 Software Accomplishment Summary | DLCA Software Accomplishment Summary for the DLCA-6510 [23]  DLCA-6510 Footprint Document [8] | S  S | S  N/A |
| 11.21 Trace Data |  | A | A |
| 11.22 Parameter Data Item File |  |  |  |

1 Change Request (CR) records will be archived in a change tracking tool database.

[[2]](#footnote-3) Software Configuration Management (SCM) records will be archived in the Enterprise Product Data Management database.

[[3]](#footnote-4) Software Quality Assurance (SQA) records will be archived in an audit tool database.

# Certification Schedule

Certification Note: DO-178C §11.1f: This section describes the means the applicantwill use to provide the certification authority with visibility of the activities of the software life cycle processes so reviews can be planned.

The objective of this section is to provide a schedule of the software development activities for the Product Line DLCA-6510 software development, which meets the Schedule objective of the Plan for Software Aspects of Certification found in DO‑178C [61], section 11.1.f.

The DLCA-6510 will follow its schedule that will correspond to certification millstones. Several key milestones include SOI 1, 2, 3, 4. Schedule details are specified in the project specific CIA. Schedule changes during project execution will be discussed during program reviews.

This section identifies the certification significant program milestones and the dates when design life cycle data will be submitted to the certification authority.

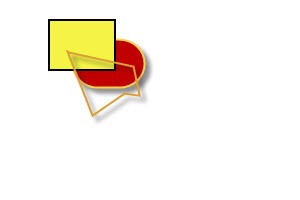


Figure 8‑1 – Representative Figure

Table 8‑1 – Representative Table

|  |  |  |
| --- | --- | --- |
| Heading 1 | Heading 2 | Heading 3 |
| Data | Data | Data |

# Additional Considerations

Certification Note: DO-178C §11.1g: This section describes specific considerations that may affect the certification process. Examples include alternative methods of compliance, tool qualification, previously developed software, option-selectable software, user-modifiable software, deactivated code, COTS software, field-loadable software, parameter data items, multiple-version dissimilar software, and product service history.

## Use of Previously Developed Software

This section should provide a brief discussion of how previously developed software will be used on this project with emphasis on differences between the proposed usage and previously approved usage. Special attention should be given to new applications that have a higher safety criticality than previous applications. Topics of interest include modifications to previously developed software, change of aircraft installation, change of application environment, change of design environment, and upgrading a design baseline. See AC 20-115D when updating an existing project following a DO-178 process other than DO-178C with Previously Developed Software.

According to AC 20-115D [76], paragraph 9, Previously Approved Software is software that was approved using ED-12/DO-178, ED-12A/DO-178A, or ED-12B/DO-178B. DLCA-6510 does not have any Previously Approved software that was approved using ED-12/DO-178 or ED-12A/DO-178A. The CIA will list the baseline for previously approved software that was developed to ED-12B/DO-178B.

DLCA-6510 reused life cycle artifact of the baseline (Refer to the CIA for the baseline which will be used). The baseline modifications were driven by new requirements and software enhancements, which were developed and verified in accordance with the applicable objectives for DO-178C, resulting in this version of the DLCA-6510 software. There were no changes to any of the existing DAL levels.

Certification credits will be taken for all artifacts that did not change from the baseline version. For artifacts that changed from the baseline version, partial credit will be taken for the unmodified portions of those artifacts, whereas re-verification was performed on the changed and affected portions. The re-verification effort will include requirements and functional based testing as well as structural coverage of the changes.

The DLCA-6510 SAS [23] will identify the final set of artifacts changed.

<For Lori: Will need more information here from Lori after the corresponding task completed>

<For Lori: Also DLSS-18372 has been created to allow IPS or EDS execution for verification instead of repeating the execution on both environment every time>

## Commercial Off The Shelf (COTS) Software

N/A – there is no Commercial off the Shelf Software (COTS) in the DLCA-6510.

## Product Service Experience

This section should briefly describe the type of product experience for which approval credit is being requested. In addition, the plan(s) for obtaining collecting, verifying, and maintaining such experience over the equipment life cycle should be described in detail.

No product service history credit will be taken for the DLCA-6510 product.

## Tool Assessment and Qualification

This section should list all tools along with an assessment as to why they do or do not need to be qualified.

Qualification is required for all software tools which:

* Eliminate, reduce or automate a process of DO-178C by the use of the tool.  
  OR
* Can insert an error into the airborne hardware or fail to detect an existing error in the hardware within the scope of the intended use of the tool AND will not have the output of the tool verified.

For each software tool that meets the preceding criteria, the following information should be provided:

* The name of the tool
* A brief functional description of the purpose of the tool
* The DO-178C process or processes eliminated, reduced or automated by the tool identified
* An assessment in accordance with the criteria in DO-178C section 12.2 and DO-330. Tool Qualification Level (TQL) should be identified and justified.
* An indication of whether the tool was purchased or internally developed
* A summary of the process to be used for tool qualification provided
* A list of the data to be submitted in support of tool qualification identified.

Additional information on software tool qualification is provided in DO-330 “Software Tool Qualification Considerations”.

DO-178C [19] Section 12.2 Tool Qualification provides the following guidance for when a development or verification tool is required to be qualified.

*Qualification of a tool is needed when processes of this document are eliminated, reduced or automated by the use of a software tool without its output being verified…*

All tools selected for qualification will be qualified in accordance with DO-178C [RTCA\_DO178C] / DO-330 [RTCA\_DO330] unless otherwise specified. Tools may be upgraded during development. If an upgrade occurs, the PE will evaluate the impact to the tool and determine whether re-qualification of the tool is required.

Table 9‑1 - DLCA-6510 S/W Tools list all the tools used in the development of the life cycle data for DLCA-6510 along with an assessment as to why they do or do not need to be qualified. Qualification is required for all software tools which:

1. Can insert an error into the airborne software or fail to detect an existing error in the software within the scope of the intended use of the tool; AND
2. Will not have the output of the tool verified as specified in Section 6 of DO-178B [61]; AND
3. Eliminate, reduce or automate a process of DO-178B by the use of the tool.

Applying this assessment to the tools planned for use in the DLCA-6510 resulted in below mentioned tools requiring qualification. They are discussed in the subsections below.

Table 9‑1 - DLCA-6510 S/W Tools

| Life Cycle | Tool Capability | Tool Used | DO-178B Qualification Required? | Rationale/Comments |
| --- | --- | --- | --- | --- |
| Support | Documentation | Google Application Suite | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Support | Documentation | DOORS | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Support | Documentation | Microsoft Office Word | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Support | Documentation | Microsoft Office VISIO | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Support | Documentation | Microsoft Office PowerPoint | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Support | Schedule Management | Microsoft Project | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Support | Schedule Management | SAP | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Support | Problem Report Tracking | JIRA | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Support | Configuration Management | Subversion | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Support | Peer Review Tool | Rockwell Collins PREP | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Support | Traceability | DOORS | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Requirements | Specification | DOORS | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Requirements | Specification | Microsoft Word | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Requirements | Modeling | Microsoft Office Visio | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Design | Modeling | Microsoft Word | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Design | Modeling | DOORS | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Design | Design graphical layout | VAPS XT | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool. Also, the tool output has been reviewed. |
| Design | IDE | Eclipse | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Coding  (Intel based target) | Source Code Analyzer | Gimpel PC Lint for C/C++ | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Coding  (Intel based target) | Source Code Analyzer | Dr.Memory² | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool. This tool helps the developer to check any memory leaks Manual Review is performed on the Source Code to ensure Software Coding standard. |
| Coding  (Intel based target) | Source Code Analyzer | Understand¹ | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool. Manual Review is performed on the Source Code to ensure Software Coding standard |
| Coding  (Intel based target) | IDE | Eclipse CDT | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Coding (PPC based target) | C/C++ Compiler | LynxOS-178 CDK GNU gcc PPC cross compiler for Windows | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Coding (PPC based target) | Source Code Analyzer | Gimpel PC Lint for C/C++ | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Coding (PPC based target) | Source Code Analyzer | Understand¹ | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool. Manual Review is performed on the Source Code to ensure Software Coding standard. |
| Software Verification | Coverage Analysis | Vector Cast | Yes | Detailed discussed about the qualification provided in subsection below. |
| Software Verification | Test Simulation | AGPS | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Software Verification | Test Simulation | ATC Ground Station | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Software Verification | Test Simulation | Airtel ATN Router | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Software Verification | Test Simulation | Message Library Tester [50] | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Software Verification | Test Simulation | Trace Tool | No | Does not eliminate, reduce or automate a process of DO-178C by the use of the tool |
| Software Verification | Test Simulation | Vision Framework Tool | Yes | Detailed discussed about the qualification provided in subsection below. |
| Software Verification | Test Simulation | VISTA | Yes | Detailed discussed about the qualification provided in subsection below. |
| Software Verification Procedures & Results | Documentation | MS Word | No | Does not eliminate, reduce or automate a process of DO-178C by use of the tool. |
| Software Verification Procedures & Results | Documentation | MS Excel | No | Does not eliminate, reduce or automate a process of DO-178C by use of the tool. |

Note 1: Understand tool is an Aid tool to help the developer. Manual Review is performed on the Source Code to ensure Software Coding standard.

Note 2: Dr.Memory is an Aid tool to the developer to check any memory leaks. Manual Review is performed on the Source Code to ensure Software Coding standard.

### Virtual Integrated Software Testbed for Avionics (VISTA)

**Type:** Verification Tool, TQL 5

**Source:** Developed by Rockwell Collins

**Functional Description:** VISTA is a software tool used throughout the development and verification processes of the DLCA-6510 product. This tool facilitates code execution, debugging, and verification testing in both target hardware and host-based environments. Selected features of this tool are used to automate the gathering of verification test results. Therefore, these features of the tool will be qualified. VISTA was developed by Rockwell Collins. The qualification of this tool will be performed by the VISTA development group.

**DO-178C process or processes eliminated, reduced, or automated by the tool:** The tool aids in the satisfaction of DO-178C [RTCA\_DO178C] Table A-7 objective 2. The data logging and file compare capabilities of VISTA eliminate the need for manual logging of results and reduce the amount of peer review effort required.

**Qualification Plan and Data:** A qualified version of the tool will be used. This tool was qualified in accordance with DO-178C [RTCA\_DO178C] / DO-330 [RTCA\_DO330] at TQL 5. If updates are made to the qualified tool, re-qualification will be completed. Only commands for logging and comparing test results are qualified. The tool qualification accomplishments summary [VISTA\_TAS] will be referenced as part of the software accomplishments summary [SAS].

### VectorCAST Cover Tool Suite

**Type:** Verification Tool, TQL TBD

**Source:** Developed by VectorCAST

**Functional Description:** VectorCAST Cover Tool Suite is developed by VectorCAST and is a software tool used to automate the collection and reporting of structural coverage results, whereby the source code is instrumented with tags that record statement coverage during functional requirements based testing. The reports are used to assist engineering in performing Structural Coverage Analysis (SCA).

**DO-178C process or processes eliminated, reduced, or automated by the tool:** The tool aids in the satisfaction of DO-178C [RTCA\_DO178C] Table A-7 objective 7 (Statement Coverage).

**Qualification Plan and Data:** VectorCAST offers a Tool Qualification Kit, and enables clients and certification authorities to audit the VectorCAST Tool Suite for use in projects. Rockwell Collins will coordinate with VectorCAST to obtain the Tool Qualification Kit, and perform the qualification testing.

## Option Selectable Software

This subsection should describe any option selectable features, the means by which such selections are made and the impact of improper selection on system functionality and safety. The means used to prevent improper selection should be described.

<For Lori: This section needs to be revisited based on input from Lori – Task DLSS-18538 planned for upcoming sprints>

Option selectable software is synonymous with deactivated software, and represents functional code segments that can be enabled/disabled by an external configuration item – such as a license key or strap. For all such code segments, the software verification test procedures will test the function when enabled *and* will test that the function will not execute when disabled.

The following option selectable software exists in the DLCA-6510:

* Dual DLCA Key: Enables active/standby mode determination logic with a peer DLCA-6510.
* FANS Key: Enables the FANS applications within the DLCA-6510.
* ATN Key: Enables the ATN applications within the DLCA-6510.

Table 9‑2 DLCA Options (Licenses Keys/Strapping)

| **Valid Keys/Strapping** | **Explanation of Options** |
| --- | --- |
| None | No DLCA active |
| 2 | Single DLCA with FANS |
| 3 | Single DLCA with ATN |
| 2,3 | Single DLCA with FANS & ATN |
| 1,2 | Dual\* DLCA with FANS |
| 1,3 | Dual\* DLCA with ATN |
| 1,2,3 | Dual\* DLCA with FANS and ATN |

Note: \* indicates Active/Standby Dual Installation

The following option selectable code exists in DLCA:

* Inbox: The DLCA software can enable the Configurable Inbox, the Non-Configurable Inbox or deactivate the inbox.  This is managed in the XML DLCA General Configuration File [21].
* Configurable Inbox Style: The Configurable Inbox can use a Message Log or operate without one.  This is managed in the XML DLCA General Configuration File [21].
* IOC Flows: The ICAO Address and Data Link Interlock Status word can use a different number of flows on the CCM hardware versus the AFD-37X0 hardware.  With the CCM hardware, the DLCA uses 4 flows and with the AFD-37X0 hardware, the DLCA uses 2 flows. This is managed in the XML I/O Configuration File [19].
* Top Level Menuing: The DLCA software can enable the drop down menu logic or tab menu logic. This is managed in the XML DLCA General Configuration File [21].

For those items that correspond to an enabled/disabled option, the requirement base software verification will test the function when enabled and will test that the function will not execute when disabled

## User Modifiable Software

This subsection should describe any user modifiable components, the means of modification, and the potential impact on system safety if such modification is incorrectly performed. See AC 20-115D (Guidance for User-Modifiable Software (UMS)) for guidance on obtaining approval of User Modifiable Software.

N/A – there is no user-modifiable software in the DLCA-6510 product.

## Field Loadable Software

See AC 20-115D (Guidance of Field-Loadable Software (FLS)) for guidance on obtaining approval of Field Loadable software.

## Multiple Version Dissimilar Software

This subsection should identify if a multiple version dissimilar software approach is being used or not on this project (typically not used). If used this section is to provide details on the multiple version dissimilar software approach(s) used on this project. See DO-178C Section 12.3.3 for guidelines on such verification.

## Parameter Data Items

This section should identify parameter data items added or changed by this project and the certification approach used on this project as described in DO-178C.

<For Lori: Assumption is this is not applicable. DLSS-18371 has been created for future sprints>

## Model-Based Development (MBD) Methods and Tools

This section should identify the use of MBD methods or tools applicable to this project.

<For Lori: Assumption is this is not applicable. DLSS-18371 has been created for future sprints>

## Multi-Core Processor Software

Although multi-core processors is considered a Systems issue with both hardware and software considerations, the software considerations should be described in this section. Official regulatory guidance regarding multi-core processors continues to evolve at the time this template was developed. Project teams will need to research the current guidance based on the way they intend to use the multi-core processor(s). Note: CAST papers are not official regulatory guidance and may not be acceptable. Aircraft Issue Papers and CRIs are official regulatory guidance which typically are the artifacts to use.

## Object Code Coverage Analysis

This section should identify if structural coverage is being applied at the source code or object code level. If structural coverage is evaluated at the object code level additional details need to be provided to justify it is being used in a manner which will address Issue Papers and CRIs when the unit is installed in an aircraft. Object code analysis is a stable regulatory area and CAST-17 typically provides useful guidance.

## Object Oriented Methods and Languages

This section should identify the use of OOT methods or tools applicable to this project.

<For Lori: DLSS-18371 has been created for future sprints>

## Alternative Methods

This section is to identify if any Alternative Methods of compliance which are planned to be applied on this project, refer to DO-178C §12.3 for additional information. Typically only DO-178C guidance is applied to a project and therefore this section typically only states “No alternative methods of compliance are planned.”

Any changes to this section from that submitted in the Plan for Software Aspects of Certification must be identified and discussed.

No alternative Methods of compliance was applied to this project.

No alternative Methods of compliance are planned.

# Outsourcing/Offshoring & Supplier Oversight

Certification Note: DO-178C §11.1h: This section describes the means of ensuring that supplier processes and outputs will comply with approved software plans and standards.

Note: Refer to Outsourcing/Offshoring ALM pages for additional guidance (<http://alm.rockwellcollins.com/wiki/display/CERTW/Certification+Planning+for+Offshoring+and+Outsourcing>) and RC-ENG-P-016.

This section should summarize the outsourcing / offshoring activities to be performed on this project and the oversight that the USA based Rockwell Collins personnel will perform on these outsourced / offshored activities. The oversight must summarize Engineering, DAC, and Certification Liaison activities.

Per RC-ENG-P-016, this section should include:

* Project Name
* Project Description
* Offshore/outsourced company and location
* Brief description of what was performed by Rockwell Collins in the U.S. and what was outsourced/offshored including identification of any safety or security functionality
* Restatement of the initial OORA results and what was outsourced/offshored (may just be a copy of what was in the PHAC)
* Any differences from what was planned versus what actually occurred
* Summary of what went well
* Summary of challenges encountered and how those were addressed
* Summary of the planned process improvements and when they plan to be implemented
* Summary of QA issues noted during the project
* Summary of DER/TCR issues noted during the project

## Tasks and Responsibilities

This section should identify all supplier tasks for the project, as defined in Table 5-4 of RC-ENG-P-016. If multiple suppliers are used, provide sub-paragraphs (i.e. 9.1.1) as necessary. Project focal point roles should be identified for both Rockwell Collins and the supplier. It should also clearly define areas of responsibility, coordination, and communication. This should also include responsibilities for approvals and recommendation for approvals of software and hardware life cycle data.

Systems engineering for DLCA-6510 will be performed by system engineers located in the United States of America (USA) and Global Engineering Technology Center-India (GETC-I) will be used as needed.

Software requirements, design, and development are planned to be done in the USA by RC personnel, contractors located within the USA and India, and the GETC-I will be performed per the SDP [6] and will be reviewed per the Peer Review process [26].

DLCA-6510 plans to utilize Global Engineering Technology Center-India (hereafter referred to as GETC-I), to perform all verification activities. GETC-I will be used to develop Acceptance Test Procedures (ATPs), verify those ATPs provide full coverage of the DLCA-6510 software requirements, run requirements-based test procedures to produce the data used for structural coverage analysis, run test procedures for a “dry-run-for-score”, and participate in a formal “run-for-score” at Rockwell Collins facilities.

As needed HCL Technologies (HCLT), an Indian engineering firm will be utilized for development and verification. All HCLT personnel used are planned to be located onsite at the RCI facilities in the USA. This would be done to help train and manage the GETC-I staff. HCLT personnel are experienced DLCA verification engineers.

The processes utilized by DLCA in previous verification efforts with off-shore verification will be repeated for this project. A verification lead will be assigned that is located in the USA. This person will be responsible for coordination with our off-shore partners. A verification lead at each of our off-shore partners will also be assigned to this project which will coordinate activities at the off-shore facilities and ensure enforcement of the DLCA processes.

All dry run and formal Run For Score (RFS) testing on the target system will be performed in the USA and India at Rockwell Collins facilities by GETC-I. Structural Coverage Analysis (SCA) testing is obtained from requirements based test procedures and is planned to occur at the off-shore and at USA Rockwell Collins facilities. SCA will be executed at off shore facilities because it is planned to be executed on the host environment and not on the target system. GETC-I will have the full host solution at their facilities. The target solution will only exist at Rockwell facilities in the USA. GETC-I facility can be used for RFS on target system as needed once the target system is setup and the required process are followed for conformity inspection.

As needed to augment staff, domestic partners will be used. All staff utilized by domestic partners will reside in the U.S. No offshoring/outsourcing outside the U.S. by the domestic partners is planned.

All TCR and safety activities will be done by personnel located in the USA. This effort will not be done through offshoring.

Appendix E provides a summary of DO-178C objectives which will be performed by USA-based Rockwell Collins team members and GETC-I team members.

## Technical Oversight

This section should identify what supplier oversight will be provided by USA based Rockwell Collins personnel for the project. USA based Rockwell Collins engineering has the ultimate responsibility for showing compliance to the FAA TSOs and associated MOPS, including the accuracy of project and certification artifacts.

Oversight of any outsourced/offshored functionality with safety or security considerations needs to be identified.

The preferred method for providing engineering oversight is for the supplier to use the Rockwell Collins development or verification plans. These plans should identify the artifacts (policies, standards, etc.) that are required to be used to perform the tasks and show compliance to the regulations.

Cedar Rapids Engineering Leads will provide oversight to ensure offsite development and verification engineers comply with Data Link’s processes currently in place.

All activities performed; regardless of what partner is conducting them; will be performed using the same tools, plans and processes used by the Rockwell Collins USA team.

The plan is 30% of artifacts modified or newly generated by our offshore partners will be reviewed by an experienced member of the development team and 5% by the Rockwell Collins USA DAC for Quality. The tools and access to version control repositories that are necessary to accomplish these tasks will be made available to our off-shore partners and domestic partners. Some restrictions on network usage for the off-shore partners will be enforced by Rockwell Collins USA team to fully comply with export control regulations.

To maintain synchronization, weekly and monthly status reports and teleconferences will be scheduled between the Rockwell Collins USA and off-shore partners team members. The final responsibility is with Rockwell Collins USA team to make sure that the artifact complies with the procedures that are required for DO-178C certification.

## Visibility of Regulations, Plans, and Standards

This activity should identify how Rockwell Collins will provide a supplier with visibility to artifacts required for the supplier to perform development or verification tasks. This should include how Rockwell Collins will ensure that all applicable regulations, policies, guidance material, OEM standards, Rockwell Collins standards, plans, and any other agreements are conveyed to, coordinated with and complied with by suppliers (including sub-tier suppliers) to Rockwell Collins.

All activities, regardless of which partner is conducting them, will be performed using the same tools, plans and processes used by the Rockwell Collins USA team. All offshore/outsource partners will have access to the applicable project plans and standards, either through direct access to the project repositories located in Cedar Rapids, Iowa, or through an agreed upon method of delivery.

The tools and access to version control repositories that are necessary to accomplish these tasks will be made available to both our offshore partners and domestic partners. Some restrictions on network usage for the offshore partners will be enforced by the Rockwell Collins USA team to fully comply with export control regulations.

## Integration Management

If the supplier is performing development activities, this section should identify how the system components will be integrated, and who will be responsible for validating and verifying the hardware and the integrated system..

All life cycle artifacts generated by offshore/outsource partners, either in whole or part, will be maintained in Rockwell Collins repositories, databases, and network drives located in Cedar Rapids, Iowa. The development, management, and validation of such artifacts will be completed in accordance with the defined processes in the SDP [5].

It is planned that an experienced member of the Rockwell Collins USA development team will review TBD% of the artifacts modified or newly generated by our offshore partners, and that a Rockwell Collins USA software quality engineer will review TBD% of the artifacts modified or newly generated by our offshore partners. Jama Review Center for requirements and Crucible and A3T tools for source code will be used to track and manage peer reviews in accordance with the peer review methods defined in the SDP [5].

The final integration and verification activities will be performed by Rockwell Collins USA personnel in Cedar Rapids, Iowa.

* 1. Requirements Management

All requirement changes regardless of who performs them or where they are performed are governed by the change request and peer review processes.

Offshoring/Outsourcing personnel may be involved in the creation and review of requirements. Artifacts produced by the Offshoring/Outsourcing personnel will be peer reviewed by USA-based Rockwell Collins personnel or Designated Reviewers from Offshore/Outsourcing teams. USA-based Rockwell Collins personnel are required to participate in peer reviews at or above the levels defined in HRC-ENG-P-016 [RC\_OORA] for the risk level identified by OORA score for the Outsourced/Offshore supplier.

All requirements are maintained in the same requirements management tool and configuration management repositories that the USA-based Rockwell Collins personnel utilize.

See Section TBD Organizational Responsibilities

See Section TBD General Supplier Oversight

* 1. Design Control and Approval

All design changes regardless of who performs them or where they are performed are governed by the change request and peer review processes.

Offshoring/Outsourcing personnel may be involved in the creation and review of design changes. Artifacts produced by the Offshoring/Outsourcing personnel will be peer reviewed by USA-based Rockwell Collins personnel or Designated Reviewers from Offshore/Outsourcing teams. USA-based Rockwell Collins personnel are required to participate in peer reviews at or above the levels defined in HRC-ENG-P-016 [RC\_OORA] for the risk level identified by OORA score for the Outsourced/Offshore supplier.

All design changes are maintained in the same requirements management tool and configuration management repositories that the USA-based Rockwell Collins personnel utilize.

See Section 6.5 Organizational Responsibilities

See Section TBD General Supplier Oversight

* 1. Integration and Verification Environment Control

The integration test environment is specified by the USA-based Rockwell Collins personnel. The project engineer (or a delegate) identifies the integration test environment, and communicates this information to the team. The test environment is also captured in the Software Verification Procedures and Results document at the end of the program.

* 1. Software Build/Release Process

The software build/release process is specified by the USA-based Rockwell Collins personnel. The project engineer (or a delegate) identifies the software build/release process to the team. The FDSA software will be released by the USA-based Rockwell Collins personnel to the Software Control Library. The software build identification is captured in the Computer Program Configuration Item document at the end of the program

See Section TBD General Supplier Oversight

## Problem Reporting and Resolution

This section should include mechanisms to identify problems and determine how problems are reported to Rockwell Collins from domestic and off-shore suppliers (including sub-tier suppliers). It should also include descriptions of tools used, if any, to aid problem reporting across software and hardware life cycle data. Rockwell Collins should be notified of all problem reports. Engineering should review supplier problem reports for concurrence with all dispositions.

The preferred method to provide compliance for the problem reporting and resolution aspects of DO-160() or DO-254() is for the supplier to use Rockwell Collins policies, procedures and tools that are identified in the PHAC. The problem reporting artifacts should be maintained at Rockwell Collins.

Problem Reporting and Resolution includes a series of steps during the design and development process. A build will be created per the schedule and the state of the development. The build will undergo subsystem testing to ensure that the delivered Change Requests (CR) are implemented correctly and function properly on the system rig. A Change Request will be generated for any issues found in any DLCA-6510 artifact (i.e. documents, code, etc.).This Change Request (CR) will go through the Change Control Board (CCB), and the necessary actions will be taken.

When a build is delivered to the Verification Team, a similar process will be followed. Upon any bug or deviation from the requirement, a Change Request will be generated. The CR(s) will go through CCB to determine the actions to be taken on the Change Request.

Note: The term Change Request (CR) is used generically to describe a change driver created in the product’s problem tracking database.

## Integration Verification Activity

This section should address who will be responsible for ensuring that all integration verification activities between all levels of suppliers will comply with the applicable guidance.

Rockwell Collins USA personnel will have responsibility for all integration verification activities in accordance with the SDP [5].

## Configuration Management

This section should show how configuration control will be maintained with Rockwell Collins across all suppliers (including sub-tier suppliers). It should also include descriptions of tools used, if any, to aid configuration management across software and hardware life cycle data. The methods used to pass artifacts between Rockwell Collins and the supplier should also be discussed. It should also show how project artifacts will be partitioned from artifacts of other projects.

The preferred method to provide compliance for the configuration management aspects is for the supplier to use Rockwell Collins policies, procedures and tools and identify them in the PHAC. The artifacts under configuration control should be maintained at Rockwell Collins.

All development artifacts will reside in Rockwell Collins repositories, databases, and network drives located in Cedar Rapids, Iowa, and will be subject to configuration management processes in accordance with the SDP [5]

## Design Assurance Oversight

This section should show how quality assurance will be maintained with Rockwell Collins across all suppliers (including sub-tier suppliers). It should include descriptions of tools used, if any, to aid quality assurance across software and hardware life cycle data. Reviews and audits used to find compliance should also be considered as well as how quality assurance deficiencies will be addressed. In addition, Quality Assurance oversight activities should be identified.

The preferred method to provide compliance for the quality assurance aspects of DO-178() is for the supplier to use Rockwell Collins policies, procedures and tools and identify them in the PSAC. All supplier DAC records of review activities are to be maintained at Rockwell Collins.

Members of Rockwell Collins USA team travelled to India to train the GETC-I/HCL engineers assigned to the DLCA-6510 project. The training covered an overview of DO-178B and a wide range of topics concerning the software development and verification environment, processes, tools, standards, management, etc. The training sessions and training materials included relevant portions of the applicable Software Development Plan [6].

Rockwell Collins USA leadership will assess the needs for additional trainings based upon progress and performance quality. Based upon the assessment, Rockwell Collins USA leadership may choose to dispatch Rockwell Collins USA team members to India for additional training or to provide training through Rockwell Collins USA approved media.

Rockwell Collins USA leadership and coordination between the GETC-I/HCL team and Rockwell Collins USA team, including the DAC for Quality and development team, will assure that the GETC-I/HCL will perform to Rockwell Collins USA processes and standards.

The GETC-I will also have DAC Representative on site that will assure that the team performs to Rockwell Collins USA processes and standards outlined in Software Development Plan [6].

## Compliance Substantiation and Data Retention

This section should address how Rockwell Collins will ensure that all supplier and (if applicable) sub-tier supplier compliance findings are substantiated and retained for the program.

All artifacts pertaining to compliance substantiation and data retention will be kept in project repositories and archived databases located in Cedar Rapids, Iowa. Peer Review records will be maintained in the project’s Crucible, Jama and A3T’s tool databases. Change Request records will be archived in the change tracking tool database. SCM records will be archived in the Enterprise Product Data Management database. DAC Quality Assurance records will be archived in an audit tool database.

## Certification Oversight

This section should show how certification oversight will be performed for the suppliers. Any planned use of supplier certification experts or outside certification reviews (e.g., contract DERs) should be documented here. If the plan is for Rockwell Collins TCRs to perform oversight, that should be documented here, including any plans for TCR on-site visits. (This may need to be discussed with the TCR beforehand.) This section should be sure to address how SOI audits will be performed for the offshored/outsourced activity. See RC-ENG-P-016 for guidelines on TCR oversight.

1. List of Acronyms

AC Airspace Access & Service

AC Advisory Circular

ACARS Aircraft Communications Addressing and Reporting System

ACS ACARS Compatible System

ADS Automatic Dependent Surveillance

AFDX Avionics Full-Duplex Switched Ethernet

AFN ATS Facilities Notification

AGPS Air Ground Protocol Stack

AGS ARINC-661 Graphics Server

ALM Application License Manager

AMC Acceptable Means of Compliance

API Application Programming Interface

ARINC Aeronautical Radio Incorporated

ARTCC Air-Route Traffic Control Center

ASE Application Service Element

ASL Avionics System LAN

ASN.1 Abstract Syntax Notation One

ATC Air Traffic Control

ATN Aeronautical Telecommunications Network

ATP Acceptance Test Procedures

ATS Air Traffic Services

ATSU Air Traffic Services Unit

BER Binary Encoding Rules

BOP Bit Oriented Protocol

BMS Binary Message Server

BRS Business and Regional Systems

CAST Certification Authorities Software Team

CCB Change Control Board

CCM Common Computing Module

CDA Commanders Digital Assistant

CIO Common Input/Output

CM Context Management

CMU Communications Management Unit

CNS Communications, Navigation, and Surveillance

CODEC Coder/Decoder

CORE Computing on Redundant Elements

COTS Commercial off the Shelf

CPCI Computer Program Configuration Item

CPDLC Controller Pilot Data Link Communication

CPN Collins Part Number

CR Change Request

CRC Cycle Redundancy Check

CRI Certification Review Item

CS Computer Software

CSC Computer Software Component

CSU Computer Software Unit

DAC Design Assurance Center

DAL Design Assurance Level

DLCA Data Link Communications Application

DLCS Data Link Communications System

DM Data Manager

DO-178C FAA Software Development Standard

DOORS Dynamic Object Oriented Requirements System

DSI Dialogue Service Interface

EASA European Aviation Safety Administration

EER Engineering Estimate Request

EICAS Engine Indicating Crew Alert System

EMOD Engineering Modification

FAA Federal Aviation Association

FANS Future Air Navigation System

FAR Federal Air Regulations

FDOR Flight Deck Operational Requirements

FHA Flight Hazards Analysis

FMF Flight Management Function

FMS Flight Management System

GETC-I Global Engineering Technology Center-India

GPS Global Positioning System

HCLT HCL Technologies Limited

HL High Level

HLR High Level Requirements

HMI Human Machine Interface

ICAO International Civil Aviation Organization

IDE Integrated Development Environment

ID Identification

IMA Integrated Modular Avionics

IOC Input/Output Concentrator

IOCF Input Output Common Format

IPC Integrated Processing Cabinet

IPS Integrated Platform Software

JAR Joint Airworthiness Requirements

KDI Kernel Downloadable Image

LAN Local Area Network

LCVSM Lifecycle Value Stream Manager

LLR Low Level Requirements

LTC Lead Technical Contact

MBD Model Based Development

MIB Managed Information Base

MIN Message Identification Number

MRN Message Reference Number

NAND Not And

NDA Next Data Authority

NDO Network Data Object

NVM Non-Volatile Memory

OEM Original Equipment Manufacturer

OORA Offshoring/Outsourcing Risk Assessment

OOT Object-Oriented Technology

OS Operating System

PC Personal Computer

PCT Partition Configuration Table

PDI Parameter Data Item

PDR Preliminary Design Review

PDU Protocol Data Unit

PE Project Engineer

PM Protocol Manager

POSIX Portable Operating System Interface

PSAC Plan for Software Aspects of Certification

Systems PSAC Program Specific Plan for System Integration Aspects of Certification

RAM Random Access Memory

RBT Requirements Based Test

RBTCA Requirements Based Test Coverage Analysis

RCI Rockwell Collins India

RCPN Rockwell Collins Part Number

REV Revision

RIU Radio Interface Unit

RFS Run For Score

RTCA Radio Technical Commission for Aeronautics

RTTI Run-Time Type Identification

RUDP Reliable User Datagram Protocol

SAP Financial Analysis Software

SARD System Architecture Requirements Document

SARPS Standards and Recommended Practices

SAS Software Accomplishment Summary

SCA Structural Coverage Analysis

SCL Software Control Library

SCM Software Configuration Management

SDD Software Design Document

SDP Software Development Plan

SIL Service Information Letter

SOI Stages of Involvement

SQA Software Quality Assurance

SQAP Software Quality Assurance Plan

SRD Subsystem Requirements Document

SRS Software Requirements Specification

SSA System Safety Analysis

STP Software Test procedure

SVPR Software Verification Procedure and Results

TBD To Be Defined

TCR TSO Compliance Representative

TPM Technical Project Manager

TSO Technical Standard Order

UPER Unaligned Packed Encoding Rules

VAPS Virtual Applications Prototyping System

VCT Virtual Machine Configuration Table

VDD Version Description Document

VDLM2 VHF Digital Link Mode 2

VHF Very High Frequency

WDI Well Defined Interface

WKN Well Known Names

WKS Well Known Services

WP Work Package

XML eXtensible Markup Language

1. Planning Change Impact Analysis

Note 1: If this is an initial (Conformance Statement) submittal with no change considerations, then this section can contain “This is the initial certification of the <equipment type> and no change considerations are applicable.” Project teams need to recognize that reuse from other certification approved equipment may be considered a change, consult your TCR if uncertainty exists.

Note 2: Guidance for performing the Change Impact Analysis is documented on ALM Note 4: The CIA format in this template should be checked against the latest Equipment CIA Template (<http://alm.rockwellcollins.com/wiki/display/CERTW/Equipment+Change+Impact+Analysis+Overview>). The criteria for determining major/minor change and the impact to part numbering is documented in RC ENG-P-002.

Note 3: This section contains recommended text which is appropriate only if change information actually exists in the references. This template intended to provide adequate guidance so changes are documented in the body of this document, but it is the responsibility of the project team to ensure the references are accurate and complete.

Note 4: The CIA format in this template should be checked against the latest Equipment CIA Template (<http://alm.rockwellcollins.com/wiki/display/CERTW/Equipment+Change+Impact+Analysis+Overview>) the CIA version in this template is modelled from the CIA “Template Rev 01/18/2019” listed in the footers of the template.

Include the following note in this document:

Note: All references in the headings of this section are references to specific sections in FAA AC 00-69.

For each of these subsections, “N/A” by itself is not an acceptable answer. Each section needs some positive indication of what the changes are, or a confirmation that the topic was evaluated and that the planned changes do not affect the section.

* 1. Software Component Changes

A software component is software which has separate DO-178 life cycle data from the equipment it is used on. Examples of this include the ARINC 661 Graphics Server (AGS) and FlexIO. See additional discussion on software components on ALM.

If software component change plans were submitted, reference them here by name and the CIA package number in Enovia. If all component changes are included in this discussion, mention the component names here and state that all component changes are included in this CIA.

Software that does not include components with separate DO-178 life cycle data (this includes most traditional LRUs) should complete this section this way:

No software components which are separately evaluated for certification compliance are used on this equipment, all software is evaluated as part of this CIA.

* 1. Change Description (3.1.2)

Section 1.3 summarizes the changes for this project.

* + 1. Documentation Updates

The purpose of this section is to identify which baseline life cycle documents are planned to be changed and also identify any documents planned to be added by this project.

Baseline documents needs to be identified by full RCPN and Rev Letter. New documents only need to identify if a new RCPN dash number or a new Rev Letter is planned. In cases where the first 7 digits are planned to change, explain why this approach is appropriate.

Note: Showing the baseline information with planned changes typically is not typically shown in another section within the PSAC. However, it is acceptable to point to a section containing this information if contained in another section or document (such as a preliminary SCID/FLSCID).

Note: The objective of this section is to identify which documents are planned to change along with a high level of how the new documents will be identified to evaluate how the plan addresses configuration management considerations. However, teams should avoid providing excessive information which could create avoidable deviation/changes from plans (example guessing that a Rev Letter will roll from Rev – to Rev A, but at the end of the project it actually is Rev B or higher)

| **Software Artifacts** | | |
| --- | --- | --- |
| **Artifact Description** | **Baseline Part Number and Rev Letter** | **“New Dash Number” or “New Rev Letter” or “New Part Number”** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

* 1. Verification Approach

Section 6.1.5 summarizes the changes for this project.

Describe the planned verification activities. Will the full test suite be run for score? Will a regression analysis be performed to substantiate running a more limited set of tests? Will all the analyses be re-performed, or does the project intend to justify reusing prior analysis results?

If section 6.1.5 does not adequately describe the verification activities performed on this specific project, then the details need to be provided here.

* 1. **Software Level Changes (3.1.3.1)**

No software certification considerations, including criticality levels, were changed on this project.

Identify any changes to the software certification basis either changes in criticality or design assurance levels or to the DO-178 basis (such as changing from DO-178B to DO-178C).

* 1. Development or Verification Environment Changes (3.1.3.2)

No change to development or verification environments from the baseline.

Summarize any changes to the development and verification environments for this update. Describe the resulting impact and the activities to satisfy DO-178B/C and continue to satisfy requirements for safe operation. For significant changes, other software process document updates may also be necessary. When in doubt, contact your TCR to discuss your particular situation.

* + 1. **Tool Changes (3.1.3.4)**

Discuss any changes to tool use in the development and verification processes, including new tools, new versions of existing tools, and changing how an existing tool is used. Describe the resulting impact and the activities to satisfy DO-178B/C and continue to satisfy requirements for safe operation. Reference section 9.4 if applicable to reduce duplication of information.

* 1. **Software Process Changes**

Identify any process changes made since the baseline certification. Describe the resulting impact and the activities to satisfy DO-178B/C and continue to satisfy requirements for safe operation. For significant process changes, an SOI-1 review may be required. When it doubt, contact your TCR to discuss your particular situation.

* 1. **Relevant Target Changes (3.1.3.5)**

Identify any changes to the platform software, processor, or other platform components and interfaces associated with the software changes. Describe the resulting impact and the activities to satisfy DO-178B/C and continue to satisfy requirements for safe operation.

* 1. **Configuration Data Changes (3.1.3.6)**

Identify any expected changes to configuration data, paying special attention to any that is activating or deactivating functions. Describe the resulting impact and the activities to satisfy DO-178B/C and continue to satisfy requirements for safe operation.

* 1. **Software Interface and Input/Output Impact (3.1.3.7)**

Identify any changes to the application’s software interfaces and I/O. For example, consider the following questions: Are new buses being activated/deactivated? Are there any changes to bus speeds (from LS to HS, etc.) or bus protocols (from ARINC 429 to CSDB)? Are new labels/bits being used to support new/modified functionality?

If component CIAs were submitted, the impact of those components should be included here so that this section provides a full picture of I/O impact to the application.

* 1. **Requirements, Design, Architecture, and Source Code Impact (3.1.3.8)**

Identify at a high level the changes to software high-level and low-level requirements, design, architecture, and source code. This may be expressed in an estimated percentage of new/changed requirements objects, lines of code, etc. A reference to relevant sections is sufficient if this information is contained in the body of this PSAC.

* 1. Supplier/Outsourcing/Offshoring Activities

Section 10 summarizes the planned Supplier/Outsourcing/Offshoring Activities for this project.

Identify the planned use of suppliers, outsourced teams, or offshored teams on this project. Also identify any supplier/outsourcing/offshoring changes since the last development. (If the team will you’ll be doing anything different than the previous released PSAC, this is the place to say so.) If you’re doing it exactly as described in the PSAC, say “Outsourcing/Offshoring will be performed as described in the PSAC.”

The baseline Offshoring/Outsourcing Risk Assessment (reference RC-ENG-P-016) should be reviewed to determine if it is still accurate. If it is still accurate, say so here. If not, provide the updated risk assessment and any planned changes in supplier oversight in this section. This may drive additional discussions and/or process updates after TCR review.

If the use of the supplier or the activities that will be performed by the supplier have changed, the PSAC should be updated. If the change in supplier usage is to reduce the scope of supplier work, a CIA will be sufficient. If the change is to identify a different or additional supplier, or to outsource/offshore additional tasks, the PSAC should be updated and resubmitted to the FAA.

If component CIAs were submitted, any discussion of changes to their supplier/outsourcing/offshoring usage does not need to be replicated here.

* 1. IMA Compliance Impact

For traditional LRUs, this section will be marked “N/A – This is a traditional LRU, IMA considerations do not apply”.

If the software is hosted on an IMA platform or is part of the platform software, will the change impact compliance to DO-297 guidance? For example, consider if any of the following will be negatively impacted: resources or allocations, health monitoring, fault detection, automatic configuration management, partitioning, etc. For software that is part of the platform, explain the impact of the change on the hosted applications as well.

* 1. Environmental/EMI Compliance Impact of Software Changes

If an Environmental SOI-1 package has been submitted, list the package number here.

Otherwise:

Identify any changes that may impact Environmental compliance (Temperature, air flow, etc.).

Identify any changes that may impact EMI control, susceptibility, consistency of emissions, or impact the functional allocation between the hardware/software interfaces.

Examples include changes to software for programmable receiver/transmitter waveforms, power management, input/output control processes/characteristics, and internal processing (error correction, DAC and ADC process, digital filtering, Built In Test (BIT), data monitoring, process sequencing, clock frequencies, loop time, timers & interrupts, graceful shutdown, and internal routing changes).

Identify if production software or “special” qualification test software is used during equipment environmental qualification testing.

For additional information, refer to RC-ENG-G-620 (Environmental Certification Guideline.) or contact an Environmental TCR to discuss your updates and possible impact to environmental and MOPS compliance.

If the software changes have effects on environmental performance, you must complete the “Requalification Matrix Table” in Appendix A of this template.

If component CIAs were submitted, the impact of those components should be included here so that this section provides a full picture of environmental/EMI impact to the application.

* 1. Proposed Stage of Involvement Reviews

For each of the stages of involvement, indicate whether the SOI is proposed to be performed separately or deferred / combined into a later SOI. (Refer to Combining SOIs Guidance on ALM for guidance on SOI reviews and when it is appropriate to defer or combine them.) These proposals will be reviewed by the TCR and may need to be updated based on TCR request.

If component CIAs were submitted, the SOI plans for those components does not need to be replicated here.

The following Stage of Involvement reviews will be performed:

**SOI-1 (Planning):**

If no planning changes are anticipated, include this statement:

Planning documents have previously been reviewed and approved; no plan changes are anticipated for this update.

Otherwise, include this statement:

Planning documents will be updated as indicated in section 1.1.5.2 and will be submitted for an SOI-1 review.

If the SOI-1 is planned to be combined with other SOI(s), identify and justify that plan.

**SOI-2 (Requirements/Design/Code):**

Identify if SOI-2 will be performed separately or justify combining it with other SOIs per the guidance on Combining SOIs.

**SOI-3 (Verification):**

Identify if SOI-3 will be performed separately or justify combining it with other SOIs per the guidance on Combining SOIs.

**SOI-4 (Accomplishment):**

SOI-4 will be performed prior to submitting a request for TSO authorization.

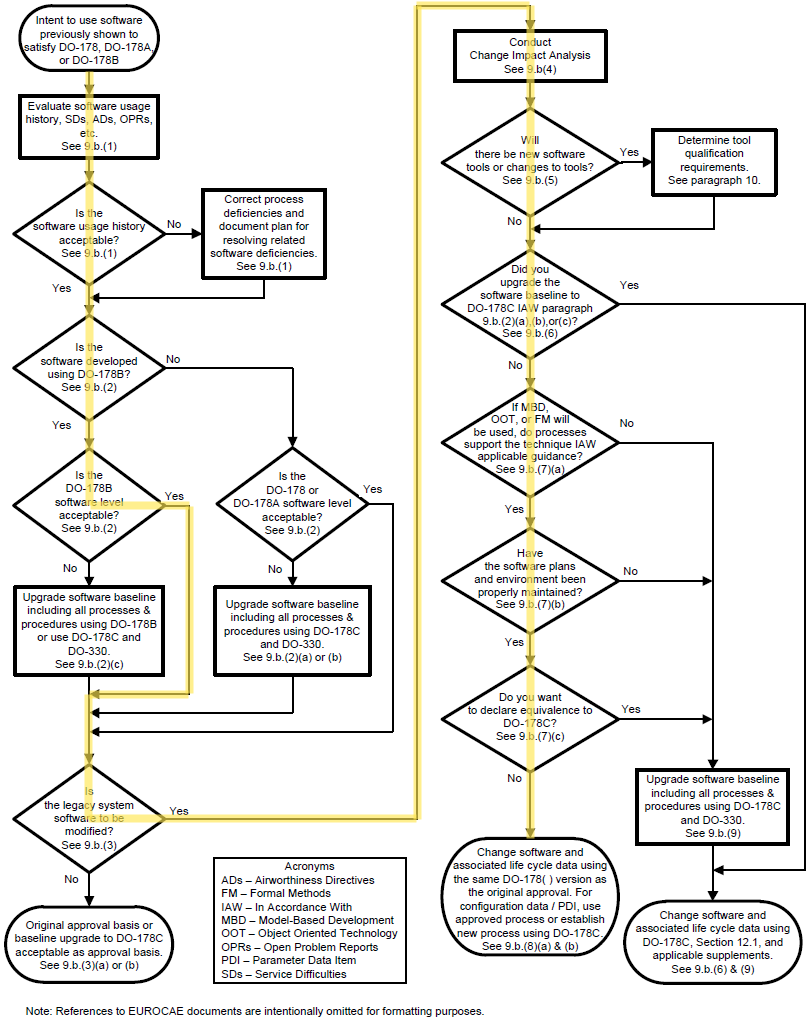
* 1. Legacy Software Analysis

Typically this should be “Not applicable, DO-178C applied to all software.”

Evaluate the planned software changes in light of Advisory Circular 20-115D and determine whether it is acceptable to remain at the prior DO-178x basis or if a transition to DO-178C is required. If a transition is required, update software planning documents to comply with DO-178C. If it is not, justify the decision to not transition in this section. A couple of the commonly-used paths through the flowchart are provided here. Use (and retain) the flowchart that represents the project plans (or provide an updated one if some other path applies). Delete the others.

If component CIAs were submitted, the DO-178C transition details for the components should be summarized here.

The following figures are only applicable if DO-178B software considerations exist and therefore should be removed for most DO-178C projects.



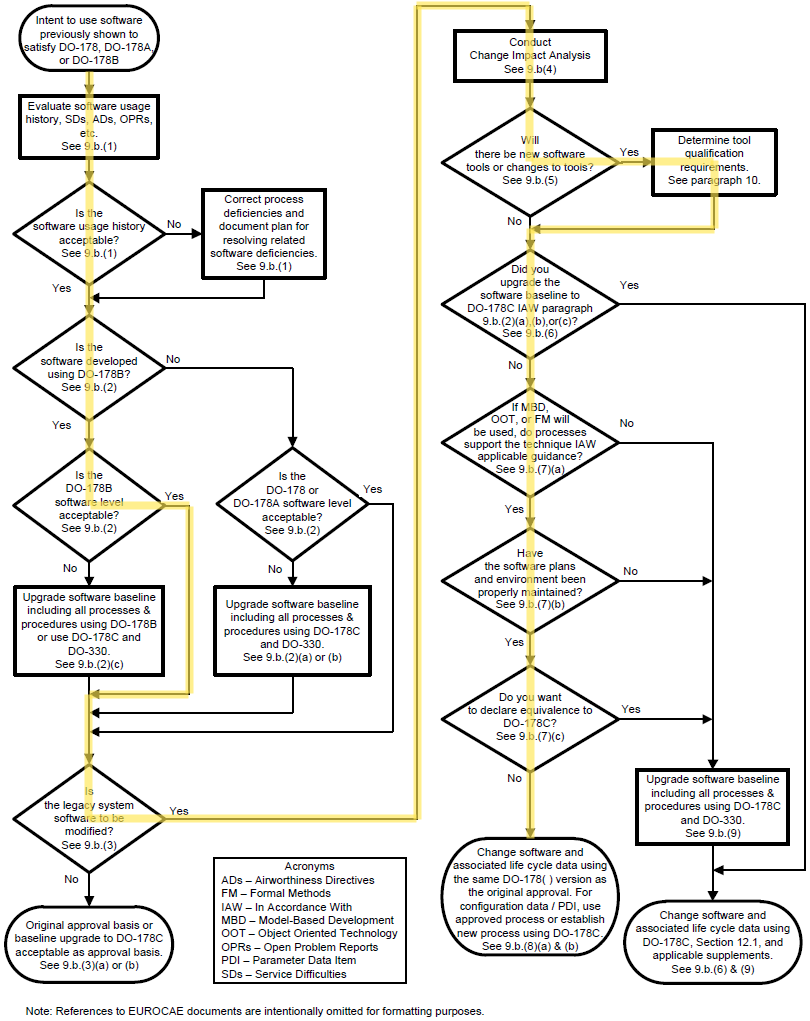


Figure 10‑1 – Legacy Software Certification Analysis (AC 20-115D)

1. DO-178C Matrix

Table 10‑1 DO-178C vs PSAC

|  |  |  |
| --- | --- | --- |
|  | **DO-178C Guideline** | **PSAC Section** |
| System Overview | 11.1.a | 3 |
| Software Overview | 11.1.b | 4 |
| Certification Considerations | 11.1.c | 5 |
| Software Life Cycle | 11.1.d | 6 |
| Software Life Cycle Data | 11.1.e | 7 |
| Certification Schedule | 11.1.f | 8 |
| Additional Considerations | 11.1.g | 9 |
| Supplier Oversight | 11.1.h | 10 |

1. FAA Order 8110.49A Matrix

The following table identifies where the issues typically are addressed, but coverage of many topics vary depending on project details. Project teams are expected to ensure each of these topics are addressed and the traceability in this table is accurate. This table is required by several, but not all, aircraft OEMs. Although compliance to these considerations is required for TSO approval, the Collins Cert Office position is that this table is not required for TSO approval.

FAA Order 8110.49A [10] has been changing to reduce the level of prescriptive guidance and therefore allow aircraft manufacturers more control over how regulatory concerns are addressed; often as FAA Issue Papers. However, the lack of information regarding FAA expectations creates an issue for product line avionics as all aircraft specific guidance may not be available during development. The product(s) addressed in this document will be developed to the latest available FAA information to minimize issues when addressing individual aircraft Issue Papers, refer to the following table for applicable mapping.

Table 10‑2 FAA Order 8110.49A Matrix

| **8110.49A Title/Topic** | **8110.49A Chapter** | **PSAC Section** |
| --- | --- | --- |
| Introduction | 1 | N/A |
| Software Review Process - Certification Liaison | 2.1 | 6.6 |
| Software Review Process – Objectives of the Software Review Process | 2.2 | 6.6 |
| Reserved | 3 | N/A |
| Software Conformity Inspection - General | 4.1 | 6.4 |
| Software Conformity Inspection - Discussion | 4.2 | 6.4 |
| Software Conformity Inspection - Software Part Conformity Inspection | 4.3 | 6.4 |
| Software Conformity Inspection - Software Installation Conformity Inspection. | 4.4 | 6.4 |
| Software Conformity Inspection - Summary | 4.5 | 6.4 |



1. Outsource/Offshoring of Regulatory Objectives

This section is required by the Wichita ACO for any project which has Outsourcing or Offshoring considerations: Refer to RC-ENG-P-016 “Risk Assessment and Oversight for Outsourcing or Offshoring Activities involving Civil Certification” for additional information.

The following table identifies the organizations addressing the applicable DO-178C objectives:

The following acronyms are used in the table:

USA-based Rockwell Collins – USA-RC

Global Engineering Technology Center - India (GETC-I)

Table 10‑3 Outsource/Offshoring Activities

| **DO-178C Objective** | **Entity Performing the Work** |  | **DO-178C Objective** | **Entity Performing the Work** |
| --- | --- | --- | --- | --- |
| A-1.1 | USA-RC, GETC-I |  | A-5.1 | USA-RC, GETC-I |
| A-1.2 | USA-RC, GETC-I |  | A-5.2 | USA-RC, GETC-I |
| A-1.3 | USA-RC, GETC-I |  | A-5.3 | USA-RC, GETC-I |
| A-1.4 | USA-RC, GETC-I |  | A-5.4 | USA-RC, GETC-I |
| A-1.5 | USA-RC, GETC-I |  | A-5.5 | USA-RC, GETC-I |
| A-1.6 | USA-RC, GETC-I |  | A-5.6 | USA-RC, GETC-I |
| A-1.7 | USA-RC, GETC-I |  | A-5.7 | USA-RC, GETC-I |
|  |  |  | A-5.8 | USA-RC, GETC-I |
| A-2.1 | USA-RC, GETC-I |  | A-5.9 | USA-RC, GETC-I |
| A-2.2 | USA-RC, GETC-I |  |  |  |
| A-2.3 | USA-RC, GETC-I |  | A-6.1 | USA-RC, GETC-I |
| A-2.4 | USA-RC, GETC-I |  | A-6.2 | USA-RC, GETC-I |
| A-2.5 | USA-RC, GETC-I |  | A-6.3 | USA-RC, GETC-I |
| A-2.6 | USA-RC, GETC-I |  | A-6.4 | USA-RC, GETC-I |
| A-2.7 | USA-RC, GETC-I |  | A-6.5 | USA-RC, GETC-I |
|  |  |  |  |  |
| A-3.1 | USA-RC, GETC-I |  | A-7.1 | USA-RC, GETC-I |
| A-3.2 | USA-RC, GETC-I |  | A-7.2 | USA-RC, GETC-I |
| A-3.3 | USA-RC, GETC-I |  | A-7.3 | USA-RC, GETC-I |
| A-3.4 | USA-RC, GETC-I |  | A-7.4 | USA-RC, GETC-I |
| A-3.5 | USA-RC, GETC-I |  | A-7.5 | USA-RC, GETC-I |
| A-3.6 | USA-RC, GETC-I |  | A-7.6 | USA-RC, GETC-I |
| A-3.7 | USA-RC, GETC-I |  | A-7.7 | USA-RC, GETC-I |
|  |  |  | A-7.8 | USA-RC, GETC-I |
| A-4.1 | USA-RC, GETC-I |  | A-7.9 | USA-RC, GETC-I |
| A-4.2 | USA-RC, GETC-I |  |  |  |
| A-4.3 | USA-RC, GETC-I |  | A-8.1 | USA-RC |
| A-4.4 | USA-RC, GETC-I |  | A-8.2 | USA-RC |
| A-4.5 | USA-RC, GETC-I |  | A-8.3 | USA-RC |
| A-4.6 | USA-RC, GETC-I |  | A-8.4 | USA-RC |
| A-4.7 | USA-RC, GETC-I |  | A-8.5 | USA-RC |
| A-4.8 | USA-RC, GETC-I |  | A-8.6 | USA-RC |
| A-4.9 | USA-RC, GETC-I |  |  |  |
| A-4.10 | USA-RC, GETC-I |  | A-9.1 | USA-RC |
| A-4.11 | USA-RC, GETC-I |  | A-9.2 | USA-RC |
| A-4.12 | USA-RC, GETC-I |  | A-9.3 | USA-RC |
| A-4.13 | USA-RC, GETC-I |  | A-9.4 | USA-RC |
|  |  |  | A-9.5 | USA-RC |
|  |  |  |  |  |
|  |  |  | A-10.1 | USA-RC |
|  |  |  | A-10.2 | USA-RC |
|  |  |  | A-10.3 | USA-RC |

1. Hardware Dependent Components

This appendix documents the different SAS’s used by the applicable software components. The DLCA software is able operate on multiple hardware platforms.

Table 10‑4 -Hardware Dependent Components

|  | **Applicable SAS** | |
| --- | --- | --- |
| **Component** | **AFD-37X0** | **CCM** |
| Platform Software | [46] | [27] |

1. Level of Involvement Self Assessment

<For Lori: This is not part of the template. I have added based on the discussion outcome>

FAA 8110.49A, Appendix A[65] contains three worksheets that may be used to help the certification authority or designee determine an appropriate level of involvement in software projects. The worksheets are provided as examples only; may contain criteria that are not applicable to all projects; and their use, individually or in combination, is not mandatory. Worksheet 1 indicates a level of involvement based on the software level of the project. Worksheet 2 allows for additional refinement of involvement based on more specific criteria. Worksheet 3 uses the total score result from Worksheet 2 to indicate a level of involvement.

The following tables contains DLCA’s self-assessment. The TSR score was 132. Based on a TSR score result of 132 and DLCA having a DAL C Software level, the recommended level of involvement is LOW per the worksheet tables provided in FAA8110.49A[65].

Worksheet 1: Level of Involvement Based on Software Level

|  |  |
| --- | --- |
| **RTCA/DO-178B/C Software Level** | **Level of Involvement** |
| D | LOW |
| C | LOW or MEDIUM |
| B | MEDIUM or HIGH |
| A | MEDIUM or HIGH |

Worksheet 2: Level of Involvement Based on Other Relevant Project Criteria

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Criteria** | **Scale MIN**. **MAX**. | **Score** |
| **1.** | **Applicant/Developer Software Certification Experience** | |  |
| 1.1 | Experience with civil aircraft or engine certification. | Scale: 0 5 10  # projects: 0 3-5 6+ | 10 |
| 1.2 | Experience with RTCA/DO-178B/C. | Scale: 0 5 10  # projects: 0 2-4 5+ | 5 |
| 1.3 | Experience with  RTCA/DO-178 or RTCA/DO-178A. | Scale: 0 3 5  # projects: 0 4-6 7+ | 0 |
| 1.4 | Experience with other software standards (other than RTCA/DO-  178 [ ]). | Scale: 0 2 4  # projects: 0 4-6 7+ | 4 |
| **2.** | **Applicant/Developer Demonstrated Software Development Capability** | |  |
| 2.1 | Ability to consistently produce RTCA/DO- 178B/C software products. | Scale: 0 5 10  Ability: Low Med High | 5 |
| 2.2 | Cooperation, openness, and resource  commitments. | Scale: 0 5 10  Ability: Low Med High | 10 |
| 2.3 | Ability to manage software development and sub-contractors. | Scale: 0 5 10  Ability: Low Med High | 10 |
| 2.4 | Capability assessments (for example, Software Engineering Institute  Capability Maturity Model, ISO 9001[]). | Scale: 0 2 4  Ability: Low Med High | 2 |
| 2.5 | Development team average based on relevant software development experience. | Scale: 0 5 10  Ability: < 2 yrs 2-4 yrs > 4 yrs | 5 |
| **3.** | **Applicant/Developer Software Service History** | |  |
| 3.1 | Incidents of software- related problems (as a  % of affected products). | Scale: 0 5 10  Incidents: > 25% > 10% None | 5 |
| 3.2 | Company  management’s support of designees. | Scale: 0 5 10  Quality: Low Med High | 10 |
| 3.3 | Company software quality assurance organization and configuration management process. | Scale: 0 5 10  Quality: Low Med High | 10 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Criteria** | **Scale** | **MIN**. |  | **MAX**. | **Score** |
| 3.4 | Company stability and commitment to safety. | Scale: Stability: | 0  Low | 3  Med | 6  High | 6 |
| 3.5 | Success of past company certification efforts. | Scale: Success: | 0  None | 3  > 50% | 6  All | 6 |
| **4.** | **The Current System and Software Application** | | | | |  |
| 4.1 | Complexity of the system architecture, functions, and  interfaces. | Scale: Complex: | 0  High | 5  Med | 10  Low | 0 |
| 4.2 | Complexity and size of the software and safety features. | Scale: Complex: | 0  High | 5  Med | 10  Low | 5 |
| 4.3 | Novelty of design and use of new technology. | Scale: Newness: | 0  Much | 5  Some | 10  None | 5 |
| 4.4 | Software development and verification environment. | Scale: Environ: | 0  None | 3  Older | 6  Modern | 3 |
| 4.5 | Use of alternative methods or additional considerations. | Scale: Standard: | 0  Much | 3  Little | 6  None | 3 |
| **5.** | **Designee Capabilities** |  | | | |  |
| 5.1 | Experience of  designee(s) with RTCA/DO-178B/C. | Scale: Projects: | 0  < 5 | 5  5-10 | 10  > 10 | 5 |
| 5.2 | Designee authority, autonomy, and  independence. | Scale: Autonomy: | 0  None | 5  Self-starter | 10  Outgoing | 5 |
| 5.3 | Designee cooperation, openness, and issue resolution effectiveness. | Scale: 0 5 10  Effectiveness: Non-Responsive Responsive Cooperative/Open | | | | 5 |
| 5.4 | Relevance of assigned designees’ experience. | Scale: Related: | 0  None | 5  Somewhat | 10  Exact | 5 |
| 5.5 | Designees’ current workload. | Scale: Workload: | 0  High | 5  Medium | 10  Low | 5 |
| 5.6 | Experience of designees with other software standards (other than RTCA/DO-  178[]). | Scale: Projects: | 0  < 5 | 3  5-10 | 5  > 10 | 3 |

Total Score Result (TSR): 132

Worksheet 3: Level of Involvement Combining Results of Worksheet 2 with Software Level

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Total Score Result (TSR)** | **Software Level A** | **Software Level B** | **Software Level C** | **Software Level D** |
| TSR < 80 | HIGH | HIGH | MEDIUM | LOW |
| 80 < TSR < 130 | HIGH | MEDIUM | MEDIUM | LOW |
| 130 < TSR | MEDIUM | MEDIUM | LOW | LOW |

1. [↑](#footnote-ref-2)
2. [↑](#footnote-ref-3)
3. [↑](#footnote-ref-4)