Peer Review in Google Sheet

RCCAC-ENG-G-900

Rev 1.0

Rockwell Collins CETC Avionics Co., Ltd.

Approval

	Name	Title	Approval	Date
Prepared by:	Di Zhang Judy Hu	Sr. Quality Engineer EPA	On File	10/18/2017
Reviewed by:	Ben Yan	System Manager	On File	10/18/2017
Reviewed by:	Yang Luo	Product Line Manager	On File	10/18/2017
Approved by:	Richard Hackett	CTO	On File	10/25/2017

Revision History

Revision	Originator	Description	Date
Rev 1.0	Di Zhang, Judy Hu	New Release	10/25/2017

Table of Contents

1 Scope

This document describes the RCCAC (Rockwell Collins CETC Avionics Co., Ltd.) engineering team Google sheet based internal peer review process. The Google sheet based peer review exists because RCCAC engineering team had not deployed any commercial peer review tool at the time this process document was under development. RCCAC engineering team shall follow this process document to plan and/or conduct their project specific peer reviews. Any deviation from this peer review process shall be communicated to quality focal and documented in the corresponding project plan.

2 References

2.1 RCCAC Documents

[1] Peer Review, RCCAC-ENG-G-601, Rev 1.2

2.2 Industry References

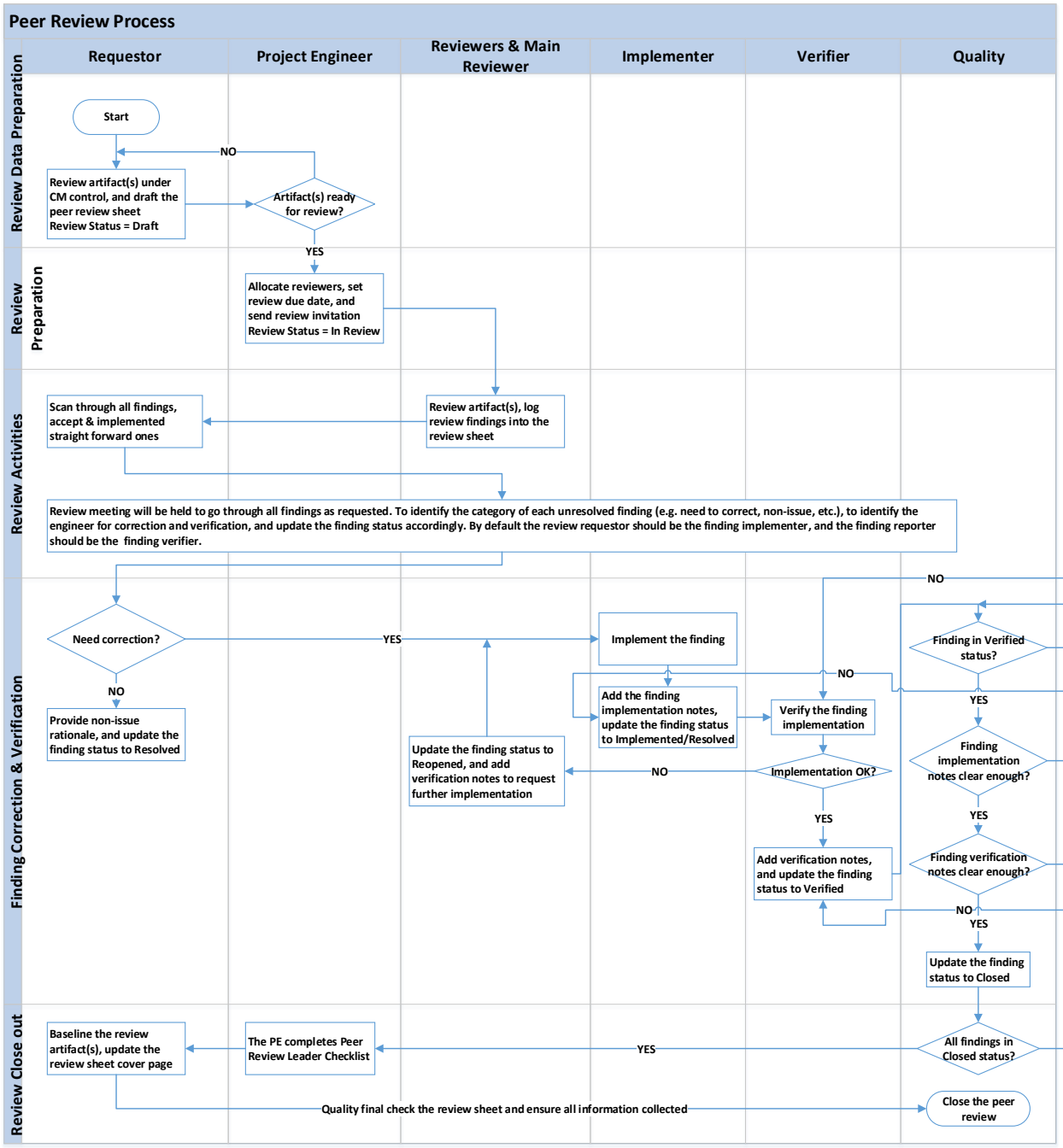
[2] Software Considerations in Airborne Systems and Equipment Certification, Document No. RTCA-178C, December 13 2011, RTCA, Washington, D.C.

3 Google Sheet Peer Review

3.1 Peer Review Process

According to [1] the RCCAC Google Sheet based peer review process is described in Figure 3-1.

Figure 3-1 RCCAC Google Sheet based Peer Review Process



3.1.1 Review Data Preparation

A Google Sheet based peer review request can be submitted by any engineer. To initiate a Google Sheet based peer review, the requestor shall:

1. Place the artifact(s) to be reviewed under RCCAC CM system;
2. Make a copy of the peer review spreadsheet template and input the peer review information, artifact information, proposed due date, etc.;
3. Register the peer review in the corresponding peer review master sheet located in cloud based SVN (shared13) under the Engineering Peer Review folder, and get a unique peer review ID assigned:

   https://apps.shared13-1corpjv.com/scm/svn/RCCAC_Engineering/Engineering%20Peer%20Review

   For example,

   - \Engineering Peer Review\RCCAC_System_Peer_Review_List.xlsx
   - \Engineering Peer Review\RCCAC_ProductLine_Peer_Review_List.xlsx
   - \Engineering Peer Review\ADF_Master_Peer_Review_List.xlsx
4. Copy appropriate checklist according to the type of the artifact to be reviewed. The reviewers should follow the checklist to perform the peer review. Table 3-1 shows sample checklists for different review artifact types. To use the review checklists is highly recommended but not mandatory. Engineering and quality may choose to use these checklists as review criteria for performing their review. If checklist is used in a review, then the checklist should be filled out before the peer review can be submitted to the quality focal for closure. It is project's responsibility to define appropriate checklists in the corresponding plan, e.g. in SyDP, HDP, SDP.

Table 3-1 Sample Peer Review Checklists for different artifacts

Artifact Type	Sample Checklist
Plan for Software Aspect Certification	Appendix C
Software Development Plan	Appendix D
Software Verification Plan	Appendix E
Software Accomplishment Summary	Appendix F
Software Verification Procedure and Result	Appendix G
Software Requirements (HLR/LLR)	Appendix H
Software Architecture and/or Design (Not Applicable)	Appendix I
C Source Code (Not Applicable)	Appendix J
Requirement based Test Case and Test Procedure	Appendix K
Structural Coverage Analysis (Not Applicable)	Appendix L
Test Result	Appendix M

3.1.2 Review Preparation

The peer review request shall be approved by the Project Engineer (PE). The PE shall ensure:

1. The artifact is ready for review;
2. The proposed due date allows enough time for the reviewers to perform the review.

If the peer review is approved, the PE shall select reviewers based on reviewers' domain knowledge and availability, and update cover page of peer review sheet with the reviewer information, and change the review status from DRAFT to IN REVIEW. The PE then sends out the data (artifacts and peer review spreadsheet) to the reviewers for review. The PE can request Engineering Project Assistant (EPA) to send out the data. It is recommended PE or EPA to send out the data, the review requestor can send out the data by the project's discretion, and the project specific peer review process shall be documented in the corresponding project plan.

There shall be at least three reviewers for each peer review. RCCAC engineering quality focal must be invited to all engineering peer reviews. Though it is up to the quality focal to decide whether to attend a specific peer review, the quality focal participation rate for a specific project must meet the requirement of the project SOW.

Depending on the size of the artifact to be reviewed, the reviewers should have at least three days to review the artifact prior to the peer review meeting.

3.1.3 Peer Review Meeting

After collecting the findings, a meeting should be held to discuss the findings. To improve the efficiency of the meeting, the producer of the artifact should review all findings before the meeting, and accept the straight-forward findings by updating the finding status to ASSIGNED.

During the review meeting, the findings that have not been accepted will be discussed one by one. Proposed solution, if there is one, will be logged into the peer review spreadsheet, and implementer and verifier will be assigned. If there is no immediate solution to a finding, an engineer will be assigned to perform the analysis. The finding status is then updated to ASSIGNED. The finding that has been proposed as non-issue and the proposal has not been accepted by the finding originator will be discussed, and the corresponding meeting decision will be recorded in the Meeting Notes.

If any review participant requests a meeting review when a desk review was planned, then a meeting review will be scheduled and conducted.

3.1.4 Correction and Verification

After the meeting, the assigned engineer should implement the finding to be corrected. When the finding is implemented, the Implementer should fill in the Implementation Notes with detailed information about the revised version in SVN, what and how the issue was implemented, and then update the finding status to RESOLVED. After that, the Verifier should verify the implementation. The finding status will be updated to VERIFIED if the implementation satisfies the verifier. Otherwise, Verifier will update the finding status to REOPENED and notify the Implementer through email. The REOPENED status is used only when the verifier is not satisfied with the implementation, which means the status will be reverted to previous ASSIGNED status and the finding needs further implementation. It is the verifier's responsibility to fill in the Verification Notes with detailed information about the revised version in SVN, how the issue implementation has been verified, and why the issue status is changed from RESOLVED to REOPENED.

The quality focal will change VERIFIED findings to CLOSED. When all findings are closed, the PE or allocated EPA will complete the Peer Review Leader Checklist, and notify quality focal to close the peer review.

The engineering quality focal should audit the process of the peer review, ensure that the peer review process (standard process and/or project specific process) is followed, all findings' implementation notes and verification notes are clear enough, all findings are closed properly, and that the peer review artifact(s) and finding sheet are under CM control.

3.1.5 Review Close Out

After all the findings are acknowledged to be closed by the engineering quality focal, the review requestor should update review sheet cover page to indicate the accepted SVN revision of the reviewed artifact(s). And the engineering quality focal should upload this closed peer review sheet to the corresponding SVN peer review folder.

3.2 RCCAC Peer Review Role Definitions

3.2.1 Requestor

The engineer(s) who submits the peer review request. Requestor is usually the producer of the artifact to be reviewed. However, engineer other than the producer or EPA may take on the role upon request by PE. If the requestor is not the producer of the artifact, the producer has to review finding list before peer review meeting and present in the meeting.

The requestor is responsible to:

- Prepare artifact, peer review sheet and register the peer review;
- Initiate peer review process by sending the peer review sheet to PE for review and approval;
- Set up peer review meeting and answer questions during the meeting;
- Baseline the artifact at the end of the peer review.

3.2.2 Project Engineer (PE)

The PE performs following activities in a peer review:

- Decide artifact's readiness for peer review;
- Decide review due date and ensure it allows enough time for the reviewers to perform the review;
- Choose reviewers & main reviewer for a peer review;
- Complete peer review leader checklist before the peer review is ready to be submitted to the quality for closure.

3.2.3 Reviewer & Main Reviewer

Reviewer could be manager, engineer, or any other RC or RCCAC employee who has good knowledge of the artifact to be reviewed.

The reviewer is responsible to:

- Review the artifact following the review standard and/or checklist;
- Log problems and findings clearly into the finding sheet;
- Participate in the peer review meeting.

The main reviewer is responsible to:

- Participate in the peer review as a Subject Matter Expert (SME);
- Complete the selected review checklist, if applicable, before the peer review is ready to be submitted to the quality for closure.

3.2.4 Implementer

The engineer assigned to implement the proposed solution.

The implementer is responsible to:

- Implement the proposed solution;
- Update the finding sheet with description of implementation notes, e.g. in which SVN revision the issue has been implemented, what issue has been implemented, and how the issue has been implemented, etc.

3.2.5 Verifier

The engineer who verifies the implemented solution. Normally the verifier for a finding is the reviewer who wrote the finding.

The verifier is responsible to:

- Verify the actual resolution addressed problems described in the finding, and match with the implementation notes;
- Update the finding sheet with description of verification notes, e.g. in which SVN revision the issue has been verified, what issue has been verified, and how the issue has been verified, etc.

3.2.6 Engineering Quality Focal

The engineering quality focal is responsible to:

- Review the artifact following the review standard and/or checklist;
- Close findings when they are in VERIFIED status;
- Ensure finding implementation notes & verification notes to be clearly described;
- Participate in the peer review meeting;
- Audit the peer review process and product.

3.3 Peer Review Template

RCCAC engineering team utilizes Google spreadsheet to record peer review findings.

RCCAC has created peer review google sheet template for engineering team to use. The peer review template contains “Cover page”, “Findings”, “Peer Review Leader Checklist”, artifact specific checklist, etc. pages for peer review. Please refer to Appendix B – Google Sheet Based Peer Review Page Layout for details.

3.3.1 Cover Page

Cover page is divided into 4 sections. The 1st section describes the peer review, including peer review title, review status, issue date, due date, actual closure date and the requestor who submitted the review request.

The 2nd section describes the artifact(s) to be reviewed, including artifact name, artifact location, status, version to be reviewed, and final reversion. The artifact location shall be cloud based SVN (shared13) hyperlink, and the artifact version shall be cloud based SVN (shared13) revision number. Any artifact location referring to shared13 drive will be rejected by engineering quality focal.

The 3rd section lists reviewers invited to the review, defining their roles within the peer review, and at least one main reviewer should be indicated. Before the review meeting, each reviewer should log the number of hours they spent in reviewing the artifact under the *Hours Spent* column.

The 4th section describes relevant reference materials, if there is any. Any reference materials location referring to shared13 drive will be rejected by engineering quality focal.

3.3.2 Findings Page

The Findings page is used to log findings against the artifact. For each finding, following attributes are defined for use:

Table 3-2 Attributes of Peer Review Finding

Attribute	Description
FID	Unique identifier of the finding
Reviewer	Reviewer who submitted the finding
Date Created	The time when the finding is created
Status	Current status of the finding could be one of the following: Draft, Assigned, Resolved, Verified, Closed, Reopened
Description	Free text to describe questions or problems found
Section	To indicate the location of the finding
Meeting Notes	Note down relevant information during the review meeting
Assigned to	Person who is assigned to implement the finding
Implementation Notes	Describe the details in which SVN revision the issue has been implemented, what issue has been implemented, how the issue has been implemented, etc.
Date Implemented	The time when the finding is implemented
Verified by	Person who verified the resolution
Verification Notes	Describe the details in which SVN revision the issue has been verified, what issue has been verified, how the issue has been verified, etc.
Date Verified	The time when the finding is verified

3.3.3 Peer Review Leader Checklist

By the end of the peer review, after all findings have been resolved and verified, the PE or allocated EPA shall complete the peer review leader checklist for quality auditing purpose. Questions must be answered accurately and honestly.

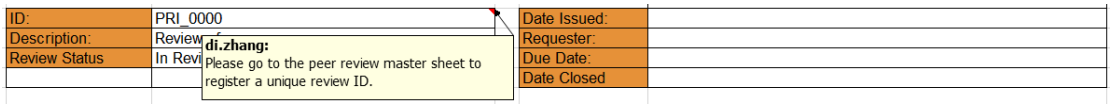
Appendix A – Acronyms, Abbreviations, and Definitions

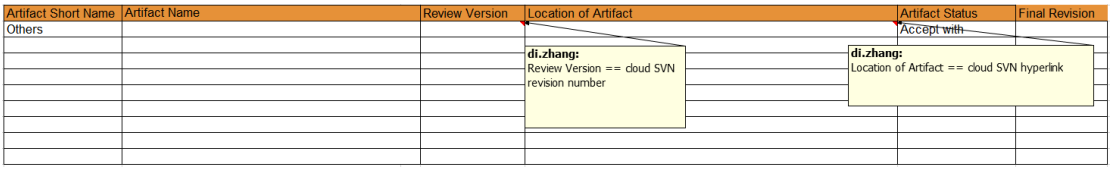
Acronyms and Abbreviations

ACRONYM	DEFINITION
CM	Configuration Management
COTS	Commercial Off the Shelf Software
CTO	Chief Technical Officer
CR	Change Request
DOORS	Dynamic Object Oriented Requirements System
EPA	Engineering Project Assistant
FAA	Federal Aviation Agency
HLR	High Level Requirement
LLR	Low Level Requirement
PE	Project Engineer
PSAC	Plan for Software Aspects of Certification
RC	Rockwell Collins
RCCAC	Rockwell Collins CETC Avionics Co., Ltd.
RCCACPN	RCCAC Part Number
SAS	Software Accomplishment Summary
SCM	Software Configuration Management
SDP	Software Development Plan
SME	Subject Matter Expert
SQA	Software Quality Assurance
SVN	Subversion
SVPR	Software Verification Procedures and Results

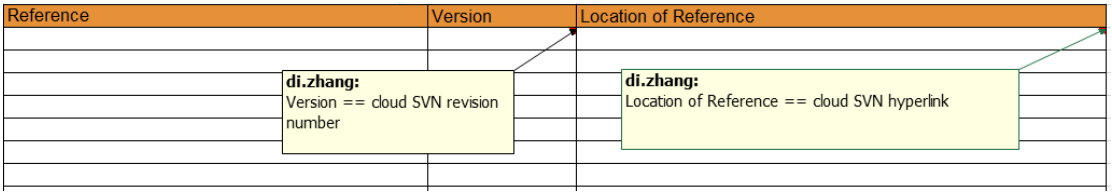
Appendix B – Google Sheet Based Peer Review Page Layout

Peer Review Sheet - Cover Page

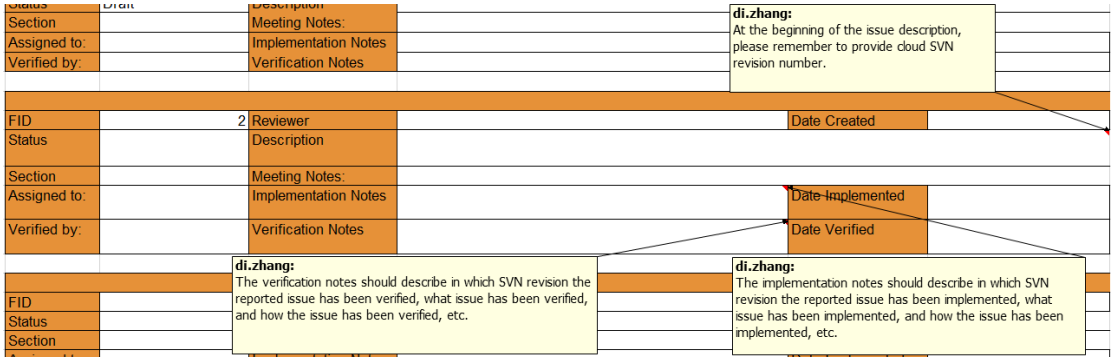




Invitee Role	Invitee Name	Hours Reviewed
Leader		
Main Reviewer		
Systems		
Systems		
Systems		
Systems		
Systems		
Quality		



Peer Review Sheet – Findings Page



Peer Review Sheet – Leader Checklist Page

Peer Review Leader Checklist

Title					
Date					
Completed By					
#	Peer Review Guide line	Question	Yes	No	Notes/References
1	4.1.1	Has the artifact been under configurationally control?			
2	4.1.1	Has the developer prepared the artifact to be reviewed and reference materials to adequately review the artifact?			
3	4.1.2	Do invitees include representatives from all stakeholder?			
4	4.1.2	Do the minimum number of attendees attend peer review? In Notes/References cell F9, please clarify it is a meeting review or desk review.			
5	4.1.3	Has the review schedule been arranged properly so that time for preparation is allowed?			
6	4.1.3	Has the reviewer spent appropriate of time on the peer review?			
7	4.1.4	Has the Review Spreadsheet been used correctly?			
8	4.1.4	Is the applicable information enough for the review?			
9	4.2	Have review findings and dispositions been recorded and can be traced?			
10	4.3	Have all findings been assigned?			
11	4.4	Have all findings been completed and verified?			

di.zhang:
To answer this question, please clarify it is a meeting review or desk review in the Notes/References column.

Appendix C – Plan for Software Aspects of Certification Checklist

No.	DO-178C[2] Ref	Question	Answer Options
1	NA	Does the document meet the current RCCAC document standard? (RCJV-ENG-I-001, PDI Section for Documents)	Yes, No, N/A
2	NA	Does the document meet the requirements of any specified template? (e.g., RCCAC-ENG-T-301)	Yes, No, N/A
3	NA	Is the Revision History up-to-date?	Yes, No, N/A
4	NA	Have Change Drivers been listed and are they at the appropriate status?	Yes, No, N/A
5	NA	Have references been reviewed for correct title, part number, and release status?	Yes, No, N/A
6	11.1 a A-1: 1	Does the plan include a system overview section which provides an overview of the system, including a description of its functions and their allocation to the hardware and software, the architecture, the processor(s) used, hardware/software interfaces, and safety features?	Yes, No, N/A
7	11.1 b A-1: 1	Does the plan include a software overview section which briefly describes the software functions with emphasis on the proposed safety and partitioning concepts, for example, resource sharing, redundancy, multiple-version dissimilar software, fault tolerance, and timing and scheduling strategies?	Yes, No, N/A
8	11.1 c A-1: 1	Does the plan include a certification considerations section which provides a summary of the certification basis, including the means of compliance as relating to the software aspects of certification?	Yes, No, N/A
9	11.1 c A-1: 1 9.1	Does the plan include a certification considerations section which states the proposed software level(s) and summarizes the justification provided by the system safety assessment process, including potential software contributions to failure conditions?	Yes, No, N/A
10	11.1 d A-1: 1	Does the plan include a software life cycle section which defines the software life cycle to be used and includes a summary of each software life cycle and its processes for which detailed information is defined in their respective software plans?	Yes, No, N/A
11	11.1 d A-1: 1	Does the software life cycle section explain how the objectives of each software life cycle process will be satisfied?	Yes, No, N/A

12	11.1 d A-1: 1	Does the software life cycle section specify the organizations to be involved and their organizational responsibilities? *This includes a description of subcontractor responsibilities if applicable.*	Yes, No, N/A
13	11.1 d A-1: 1	Does the software life cycle section describe the system life cycle processes?	Yes, No, N/A
14	11.1 d A-1: 1	Does the software life cycle section describe the certification liaison process responsibilities?	Yes, No, N/A
15	11.1 e A-1: 1	Does the plan include a software life cycle data section which specifies the software life cycle data that will be produced and controlled by the software life cycle processes?	Yes, No, N/A
16	11.1 e A-1: 1	Does the software life cycle data section describe the relationship of the data to each other or to other data defining the system?	Yes, No, N/A
17	11.1 e A-1: 1	Does the software life cycle data section describe the software life cycle data to be submitted to the certification authority, the form of the data and the means by which software life cycle data will be made available to the certification authority?	Yes, No, N/A
18	11.1 f A-1: 1	Does the schedule section describe the means the applicant will use to provide the certification authority with visibility of the activities of the software life cycle processes so reviews can be planned?	Yes, No, N/A
19	11.1 g A-1: 1, 4	Does the additional considerations section describe specific features that may affect the certification process, for example: • Alternative methods of compliance • Tool Assessment and Qualification • Previously developed software • Option-selectable software • User-modifiable software • COTS software • Field-loadable software • Multiple-version dissimilar software • Product service history • Deactivated Code *Typical questions may include:* *Have* sections *5, 6, 7, 8, and 9 of FAA N8110.49 notice been addressed?*	Yes, No, N/A
20	11.1 h	Does the supplier oversight section describes that the supplier process and task output will comply with the approved software plan and standard?	Yes, No, N/A

21	4.4.1 f	Does the known tool problems and limitation which can adversely affect airborne software are assessed and described?	Yes, No, N/A
22	11.1 A-1: 1	Is the proposed software life cycle in the PSAC commensurate with the rigor required for the level of software being developed?	Yes, No, N/A

Appendix D – Software Development Plan Checklist

No.	DO-178C[2] Ref	Question	Answer Options
1	NA	Does the document meet the current RCCAC document standard? (RCJV-ENG-I-001, PDI Section for Documents)	Yes, No, N/A
2	NA	Does the document meet the requirements of any specified template? (e.g., RCJV-ENG-T-305)	Yes, No, N/A
3	NA	Is the Revision History up-to-date?	Yes, No, N/A
4	NA	Have Change Drivers been listed and are they at the appropriate status?	Yes, No, N/A
5	NA	Have references been reviewed for correct title, part number, and release status?	Yes, No, N/A
6	11.2 a A-1: 5 5.1.1 a 5.1.2 e A-2: 1	Does the plan identify the Software Requirements Standards and method for the project?	Yes, No, N/A
7	11.2 a A-1: 5 A-2: 3 5.2.2,a	Does the plan identify the Software Design Standards and method for the project?	Yes, No, N/A
8	11.2 a A-1: 5 5.3.2.b A-2:6	Does the plan identify the Software Code Standards and method for the project?	Yes, No, N/A
9	11.2 a A-1: 4, 5	Does the plan include references to the standards for previously developed software, including COTS software, if those standards are different?	Yes, No, N/A
10	11.2 b A-1: 1	Does the plan include a description of the software life cycle processes to be used to form the specific software life cycle(s) to be used on the project?	Yes, No, N/A
11	5.1.1 a 5.1.2 a A-2: 1	Does the plan address that the system functional and interface requirements allocated to software will be analyzed for ambiguities, inconsistencies and undefined conditions?	Yes, No, N/A
12	5.1.1 a 5.1.2 b A-2: 1	Does the plan address that inputs to the software requirements process, detected as inadequate or incorrect, will be reported as feedback to the input source for clarification or correction?	Yes, No, N/A

13	5.1.1 a 5.1.2 c A-2: 1	Does the plan address that each system requirement, allocated to software, will be specified in the high-level requirements?	Yes, No, N/A
14	5.1.1 a 5.1.2 h A-2: 1	Does the plan address that each system requirement allocated to software should be traceable to one or more software high-level requirements?	Yes, No, N/A
15	5.1.1 a 5.1.2 i A-2: 1	Does the plan address that each high-level requirement should be traceable to one or more system requirements, except for derived requirements?	Yes, No, N/A
16	5.1.1 a 5.1.2 d A-2: 1	Does the plan address that high-level requirements will be defined to address system safety requirements?	Yes, No, N/A
17	5.1.1 b 5.1.2 i A-2: 2	Does the process address that derived high-level requirements should be provided to the system safety assessment process? *Note: engineering must include the following required text in the SDP:* "Derived software requirements are requirements that are not directly traceable to a higher level requirement. High and Low Level derived requirements shall be captured and received into the system safety assessment process. This process step assures compliance with DO-178C Sections 5.0, 5.1.1b, 5.1.2i, 5.2.1b, and 5.2.2c."	Yes, No, N/A
18	11.2 b A-1: 2	Does the plan include the transition criteria for the software development processes? *This includes:* • *Inputs to the process, including feedback from other processes.* • *Any integral process activities that may be required to act on these inputs.* • *Availability of tools, methods, plans and procedures.*	Yes, No, N/A
19	11.2 b A-1: 1	Is the software life cycle description distinct from the summary provided in the Plan for Software Aspects of Certification, in that it provides the detail necessary to ensure proper implementation of the software life cycle processes?	Yes, No, N/A
20	11.2 c1 A-1: 3	Does the description of the environment include the chosen requirements development method(s) and tools to be used?	Yes, No, N/A
21	11.2 c2 A-1: 3	Does the description of the environment include the chosen design method(s) and tools to be used?	Yes, No, N/A

22	11.2 c3 A-1: 3	Does the description of the environment includes the coding method, programming language(s), coding tools to be used and when applicable, options and constraints of autocode generators to be used?	Yes, No, N/A
23	11.2 c4	Does the description of the environment include the compilers, linker, editors and loader to be used?	Yes, No, N/A
24	11.2 c5 A-1: 3	Does the description of the environment include the hardware platforms for the tools to be used?	Yes, No, N/A

Appendix E – Software Verification Plan Checklist

No.	DO-178C[2] Ref	Question	Answer Options
1	11.3 a A-1: 1	Does the plan define organizational responsibilities within the software verification process and interfaces with the other software life cycle processes?	Yes, No, N/A
2	11.3 b A-1: 1	Does the plan include a description of the methods for establishing verification independence, when required? *Independence guidelines are shown in DO-178C Annex A.*	Yes, No, N/A
3	11.3 c1 A-1: 1	Does the plan include a description of the verification review methods including checklists or other aids? *Typical questions may include:* *Are the review methods and/or checklists defined or referenced for the high-level requirements, low-level requirements, software architecture, source code, test cases, test procedures and test results?* *Do the checklists include the objectives from DO-178C section 6.3 and Tables A-3, A-4, A-5, A-6, A-7?*	Yes, No, N/A
4	11.3 c2 A-1: 1	Does the plan include a description of the verification analysis methods including traceability and coverage analysis? *Typical questions may include:* • *Is there a method for resolving structural coverage analysis issues per DO-178C section 6.4.4.3?*	Yes, No, N/A
5	11.3 c3 A-1: 1	Does the plan include a description of the verification testing methods including guidelines that establish the test case selection process, the test procedures to be used, and the test data to be produced? *Typical questions may include:* • *Is a method for recording and reporting anomalies defined?*	Yes, No, N/A
6	11.3 d A-1: 1	Does the plan include a description of the equipment for testing, the testing and analysis tools, and the guidelines for applying these tools and hardware test equipment?	Yes, No, N/A

7	11.3 e A-1: 2	Does the plan include a description of the transition criteria for entering the software verification process defined in the plan? *This includes:* • *The inputs to the process, including feedback from other processes.* • *Any integral process activities that may be required to act on these inputs.* • *Availability of tools methods, plans and procedures.*	Yes, No, N/A
8	11.3 f A-1: 4	If partitioning is used, does the plan include the methods used to verify the integrity of the partitioning?	Yes, No, N/A
9	11.3 g A-1: 4	Is a description of the assumptions made by the applicant about the correctness of the compiler, linkage editor or loader included in the plan?	Yes, No, N/A
10	11.3 h A-1: 4	For software modifications, has a description of the methods for identifying the affected areas of the software and the changed parts of the Executable Object Code been included? *The re-verification should ensure that previously reported errors or classes of errors have been eliminated.* *Typical questions may include:* • *Is the regression analysis process defined?*	Yes, No, N/A
11	11.3 i A-1: 4	For previously developed software, if the initial compliance baseline for the verification process does not comply with DO-178C, then has a description of the methods to satisfy the objectives of this document been included in the plan?	Yes, No, N/A
12	11.3 j A-1: 4	If multiple-version dissimilar software is used, then has a description of the software verification process activities been included in the plan and are they in conformance with DO-178C (paragraph 12.3.2)?	Yes, No, N/A
13	NA	Does the Software Verification Plan include a description of the verification procedures to satisfy the software verification objectives? *Typical questions may include:* • *Will the planned verification process satisfy the objectives of:* *6.1, 6.3.1, 6.3.2, 6.3.3, 6.3.4, 6.3.5, 6.3.6, 6.4.2.1, 6.4.2.2, 6.4.4.1, 6.4.4.2,* *Table A-3, Table A-4, Table A-5, Table A-6, Table A-7?*	Yes, No, N/A
14	NA	Are the activities defined for the software verification processes and integral processes of the software life cycle that will address the system requirements and software level(s)?	Yes, No, N/A

15	NA	Have the Software Plans been developed at a point in time in the software life cycle that provides timely direction to the personnel performing the software development process and integral processes?	Yes, No, N/A
16	NA	Does the plan include a description of the inter-relationships between the processes, their sequencing, feedback mechanisms, and transition criteria?	Yes, No, N/A
17	NA	Has the software verification environment been selected, including the methods and tools to be used for the activities of each software life cycle process?	Yes, No, N/A
18	NA	Have methods and tools been chosen that provide error prevention in the software verification processes?	Yes, No, N/A
19	NA	Does the software planning process provide coordination between the software verification and integral processes to provide consistency among strategies in the software plans?	Yes, No, N/A
20	NA	When multiple-version dissimilar software is used in a system, does the software planning process choose the methods and tools to achieve the error avoidance or detection necessary to satisfy the system safety objectives?	Yes, No, N/A
21	NA	If deactivated code is planned, does the software verification planning process describe how the deactivated code will be verified to achieve system safety objectives?	Yes, No, N/A
		SCM and SQA checklist	
22	11.4	Does the SCM section establish the method to be used to achieve the objectives of the SCM process per DO-178C?	Yes, No, N/A
23	11.5	Does the SQA section establish the method to be used to achieve the objectives of the SQA process per DO-178C?	Yes, No, N/A

Appendix F – Software Accomplishment Summary Checklist

No.	DO-178C[2] Ref	Question	Answer Options
1	11.20 a	Does the SAS include an overview of the system, including a description of its functions and their allocation to hardware and software, the architecture, the processor(s) used, the hardware/software interfaces, and safety features?	Yes, No, N/A
2	11.20 a	If the system overview description is functionally different than the description in the PSAC, are those differences specifically described in the SAS?	Yes, No, N/A
3	11.20 b	Does the SAS include a software overview which briefly describes the software functions with emphasis on the safety and partitioning concepts used?	Yes, No, N/A
4	11.20 b	If the software overview description is functionally different than the description in the PSAC, are those differences specifically described in the SAS?	Yes, No, N/A
5	11.20 c	Is certification considerations restated as described in the PSAC and are any differences described?	Yes, No, N/A
6	11.20 d	Are software characteristics described which states the Executable Object Code size, timing and memory margins, resource limitations, and the means of measuring each characteristic?	Yes, No, N/A
7	11.20 e	Does the SAS summarize the actual software life cycle(s) and explains differences from the software life cycle and software life cycle processes proposed in the PSAC?	Yes, No, N/A
8	11.20 f	Is the software life cycle data referenced which is produced by the software development processes and integral processes?	Yes, No, N/A
9	11.20 f	Is a description included of the relationship of the data to each other and to other data defining the system, and the means by which software life cycle data will be made available to the certification authority?	Yes, No, N/A
10	11.20 f	Is a description included of the differences between the actual software life cycle data and the Plan for Software Aspects of Certification?	Yes, No, N/A
11	11.20 g	Does the SAS include a section describing additional considerations which summarizes certification issues that may warrant the attention of the certification authority and references data items applicable to these issues?	Yes, No, N/A

12	N/A	Does the SAS include a section describing how Link Analysis, Memory Analysis and Stack Analysis were performed?	Yes, No, N/A
13	N/A	Does the SAS include a section to describe report/result of the Link Analysis, Memory Analysis and Stack Analysis?	Yes, No, N/A
14	N/A	Does the result of the Link, Memory and Stack analysis is peer reviewed? Note: The result of the review can be reviewed separately or part of the SAS review.	Yes, No, N/A
15	11.20 h	Is the software configuration identified by part number and version?	Yes, No, N/A
16	11.20 i	Does the SAS include a Change History section which includes a summary of software changes with attention to changes made due to failures affecting safety, and which identifies changes from the software life cycle processes since the previous certification?	Yes, No, N/A
17	11.20 j	Does the SAS include a software status section which contains a summary or problem reports unresolved at the time of certification, including a statement of functional limitations?	Yes, No, N/A
18	11.20 k	Does the SAS include a compliance statement section which includes a statement of compliance with DO-178C?	Yes, No, N/A
19	11.20 k	Are additional rulings and deviations from the software plans, standards, and DO-178C addressed?	Yes, No, N/A
20	N/A	For software updates or minor changes, does the SAS clearly identify the baseline?	Yes, No, N/A
21	N/A	Is the top-level RCCACPN of the LRU to be certified included?	Yes, No, N/A
22	N/A	Are all the part numbers full 10-digit part numbers, including the revision letter, if applicable?	Yes, No, N/A
23	N/A	For software updates or minor changes, does the SAS describe the processes followed since the previous certification?	Yes, No, N/A
24	N/A	Does the SAS summarize the software verification results, describing any open problem reports, and explaining why open problem reports do not affect the certifiability of the aircraft?	Yes, No, N/A
25	11.1 h	Does the supplier oversight section describes that the supplier process and task output complies with the approved software plan and standard?	Yes, No, N/A

Appendix G – Software Verification Procedure and Result Checklist

No.	DO-178C[2] Ref	Question	Answer Options
1	11.13 a	Does the software verification case and procedures data include details, supplementary to the description in the Software Verification Plan, which describes the scope and depth of the review or analysis methods to be used?	Yes, No, N/A
2	11.13 b	Does the test case data include the purpose of each test case, set of inputs, conditions, expected results to achieve the required coverage criteria, and the pass/fail criteria?	Yes, No, N/A
3	11.13 c	Do the software test procedures data include the step-by-step instructions for how each test case is to be set up and executed how the test results are evaluated, and the test environment to be used?	Yes, No, N/A
4	11.14 a	Does the SVPR include review and analyses records, or references thereof, for reviews against the following life cycle data: • Software Plans • Software High-Level Requirements • Software Low-Level Requirements • Software Architecture • Software Code • Software Integration • Test Cases and Procedures • Test Results	Yes, No, N/A
5	11.14 a	For each analysis is an indication included that it passed or failed during the activities and the final pass/fail results?	Yes, No, N/A
6	11.14 a	For each analysis failure (if there is any), is the failure corrected or captured in a CR with the justification?	Yes, No, N/A
7	11.14 a	For each test, is an indication included that it passed or failed during the activities and the final pass/fail results?	Yes, No, N/A
8	11.14 a	For each test failure (if there is any), is the failure corrected or captured in a CR with the justification?	Yes, No, N/A
9	11.14 b	For items recorded as reviewed, analyzed, or tested in the SVPR, are the items correctly identified using configuration controlled versions?	Yes, No, N/A

10	11.14 c	Does the SVPR include the results of tests, reviews, and analyses, including coverage analyses and traceability analyses? *If the peer review results are captured and stored in a paper or electronic project file, the SVPR should summarize the review process performed and describe where the peer review records can be found.*	Yes, No, N/A

Appendix H – Software Requirements Checklist

High Level Requirement Checklist:

No.	DO-178C[2] Ref	Question	Answer Options
1	A-3:1	Do the software high-level requirements comply with the system requirements? *The objective is to ensure that the system functions to be performed by the software are defined, that the functional, performance, and safety-related requirements of the system are satisfied by the software high-level requirements, and that derived requirements and the reason for their existence are correctly defined.*	Yes, No, N/A
2	A-3:2	Are the software high-level requirements accurate and consistent? *The objective is to ensure that each high-level requirement is accurate, unambiguous, and sufficiently detailed and that the requirements do not conflict with each other. Also the requirement is free of conflicts with the target computer, especially system response time and input/output hardware.*	Yes, No, N/A
3	A-3:4	Are the software high-level requirements verifiable? *The objective is to ensure that each high-level requirement can be verified. In this context, "verifiable" means that a requirements-based verification procedure (e.g. review, analysis, or test) can be developed to verify that the software implementation satisfies the high-level requirement.*	Yes, No, N/A

4	A-3:5	Do the software high-level requirements conform to standards? Have the required DOORS attributes been completed?
		• _Req=True
		• _Derived=True or False
		  ○ If Derived=True, _Rationale filled out with reason why requirement is necessary
		• Link By Attribute Traceability
		• _Safety=True or False
		  ○ If derived requirement has _Safety=True, requirement must maintain _Safety=True
		• _Verification Method = Test, Analysis, etc.
		Are derived requirements identified along with rationale?
		The objective is to ensure that the Software Requirements Standards were followed during the software requirements process and that deviations from the standards are justified.
			Yes, No, N/A

5	A-3:6	Are the software high-level requirements traceable to system requirements? *The objective is to ensure that the functional, performance, and safety-related requirements of the system that are allocated to software were developed into the software high-level requirements. Is each non-derived high-level requirement traced to a system level requirement?*	Yes, No, N/A
6	A-3:1	Are all aspects of the system requirement that the software requirements traces to satisfied by the software requirements?	Yes, No, N/A
7	11.21.a	Are the system requirements traceable to software high-level requirements? *The objective is to ensure that the functional, performance, and safety-related requirements of the system are completely decomposed in the high-level requirement. This ensures the bi-directional traces between the system-level and high level requirement established.*	Yes, No, N/A
8	A-3:6	Are all aspects of the software requirements traced to at least one system requirement? If the Software requirement has additional aspects that do not trace, is the software requirement marked as derived?	Yes, No, N/A
9	A-3:7	Are the algorithms specified in the software high-level requirements accurate? *The objective is to ensure the accuracy and behavior of the proposed algorithms, especially in the area of discontinuities*.	Yes, No, N/A

10	A-2:1,2 11.9 a	Do the software high-level requirements contain a description of the allocation of system requirements to software, with attention to safety related requirements and potential failure conditions? *Are the software high-level requirements truly a decomposition of the system requirements? Do all safety related high-level requirements trace back to a safety-related system requirement?*	Yes, No, N/A
11	A-2:1,2 11.9 b	Do the software high-level requirements include functional and operational requirements under each mode of operation?	Yes, No, N/A
12	A-2:1,2 11.9 c	Do the software high-level requirements include performance criteria, for example, precision and accuracy?	Yes, No, N/A
13	A-2:1,2 11.9 d	Do the software high-level requirements include timing requirements and constraints?	Yes, No, N/A
14	A-2:1,2 11.9 e	Do the software high-level requirements include memory size constraints?	Yes, No, N/A
15	A-2:1,2 11.9 f	Do the software high-level requirements include hardware and software interfaces, for example, protocols, formats, frequency of inputs and frequency of outputs?	Yes, No, N/A
16	A-2:1,2 11.9 g	Do the software high-level requirements include failure detection and safety monitoring requirements?	Yes, No, N/A
17	A-2:1,2 11.9 h	If partitioning (protection) requirements are allocated to software, has it been defined how the partitioned software components interact with each other, and the software level of each partition?	Yes, No, N/A
18	5.1.2 h	Does each derived software high-level requirement include a rationale?	Yes, No, N/A

Low Level Requirement Checklist:

No.	**DO-178C Ref**	**Question**	**Answer Options**

1	A-4:1	Do the software low-level requirements comply with the high-level requirements? *The objective is to ensure that the software low-level requirements satisfy the software high-level requirements and that derived requirements and the design basis for their existence are correctly defined.*	Yes, No, N/A
2	A-4:2	Are the software low-level requirements accurate and consistent? *The objective is to ensure that each low-level requirement is accurate and unambiguous and that the low-level requirements do not conflict with each other.*	Yes, No, N/A

3	A-4:5	Do the software low-level requirements conform to standards? Have the required DOORS attributes been completed?
		• _Req=True
		• _Derived=True or False
		  ○ If Derived=True, _Rationale filled out with reason why requirement is necessary
		• Link By Attribute Traceability
		• _Safety=True or False
		  ○ If derived requirement has _Safety=True, requirement must maintain _Safety=True
		• _Verification Method = Test, Analysis, etc.
		The objective is to ensure that the Software Requirements Standards were followed during the software requirements process and that deviations from the standards are justified.
		Are derived requirements identified along with rationale?
			Yes, No, N/A

4	A-4:6	Are the software low-level requirements traceable to high-level requirements? *The objective is to ensure that the high-level requirements were completely decomposed and developed into the software low-level requirements. Is each non-derived low-level requirement traced to a high-level requirement?*	Yes, No, N/A
5	11.21.b	Are the software high-level requirements traceable to low-level requirements? *The objective is to ensure that the high-level requirements were completely decomposed and developed into the software low-level requirements. This ensures the bi-directional traces between the high-level and the low- level requirement established.*	Yes, No, N/A

6	A-4:1	Are all aspects of the software high level requirement that the software low level requirements traces to satisfy by the software low level requirements?	Yes, No, N/A
7	A-4:6	Are all aspects of the software low level requirements traced to at least one software high level requirement? If the Software low level requirement has additional aspects that do not trace to high level requirement, is the software low level requirement marked as derived?	Yes, No, N/A
8	A-4:7	Are the algorithms specified in the software low-level requirements accurate? *The objective is to ensure the accuracy and behavior of the proposed algorithms, especially in the area of discontinuities.*	Yes, No, N/A
9	5.2.1 b	Are the low-level derived requirements provided to the system safety assessment process?	Yes, No, N/A
10	N/A	Where the new requirement affects code that is linked to other requirements, have all requirements been examined to ensure that the new requirement is consistent and correct in combination with the other requirement?	Yes, No, N/A
11	A-2:4,5 11.10 a	Does the design description data include a detailed description of how the software satisfies the specified software high-level requirements, including algorithms, data structures, and how software high-level requirements are allocated to processors and tasks? *Do the software low-level requirements represent a complete decomposition of the allocated software high-level requirements?*	Yes, No, N/A
12	A-2:4,5 11.10 e	Does the design description data define applicable resource limitations, such as timing and memory margins? *Do the software low-level requirements include resource requirements?*	Yes, No, N/A
13	A-2:3,4,5 11.10 j	Do the software low-level requirements include derived requirements from the software design process? *While DO-178C expects that low-level requirements will contain derived requirements, it is quite possible that every low-level requirement in a given requirements document traces to a high-level requirement. If such is the case, answer "No" and simply state in the Comment field that "All requirements trace to high-level requirements".*	Yes, No, N/A

Appendix I – Architecture and Design Checklist

No.	DO-178C[2] Ref	Question	Answer Options
1	A-4:8 6.3.3 a	Is the software architecture compatible with high-level requirements? *The objective is to ensure that the software architecture does not conflict with the high-level software requirements, especially functions that ensure system integrity (e.g. partitioning schemes, task rate and order).*	Yes, No, N/A
2	A-4:9 6.3.3 b	Is the software architecture consistent? *The objective is to ensure that a correct relationship exists between the components of the software architecture. This relationship exists via data flow and control flow.*	Yes, No, N/A
3	A-4:10 6.3.3 c	Does the software architecture and/or detail design data include a description on interrupts and/or exception handling? (if any)	Yes, No, N/A
4	A-4:12 6.3.3 e	Does the software architecture conform to design standards? *The objective is to ensure that the Software Design Standards were followed during the software design process and that deviations from the standards are justified, especially complexity restrictions and design constructs that would not comply with the system safety objectives. Reference the project specific Software Development Plan for a description of the Software Design Standards.*	Yes, No, N/A
5	A-4:13 6.3.3 f 11.10 h	Do the software architecture and/or detailed design data include a description of the partitioning methods and the means of preventing partition breaches? *The objective is to ensure that partitioning breaches are prevented.*	Yes, No, N/A
6	A-2:3 11.10 b	Does the design description data define the overall software structure for implementing the software requirements? *Does the software architecture provide a level of decomposition that identifies the significant software components/processes within the system?*	Yes, No, N/A
7	A-2:3 11.10 c	Do the software architecture and/or detailed design data include input/output descriptions, for example, a data dictionary, both internally and externally throughout the software design?	Yes, No, N/A
8	A-2:3 11.10 d	Do the software architecture and/or detailed design data include the data flow and control flow of the design? *Does the software architecture provide data flow diagrams to show how external data is processed?*	Yes, No, N/A

		Does the software architecture provide control flow diagrams to show overall tasking and threading flows?	
9	A-2:3 11.10 e	Does the design description data define applicable resource limitations, such as timing and memory margins? *Do the software architecture and/or detailed design data include resource allocations and corresponding design margins?*	Yes, No, N/A
10	A-2:3 11.10 f	Do the software architecture and/or detailed design data include a description of the scheduling procedures and inter-processor/inter-task communication mechanisms? *Does the software architecture document task's execution sequence/order, rate and conditional execution? (if any)*	Yes, No, N/A
11	A-2:3 11.10 g	Do the software architecture and/or detailed design data include a description of the design methods and details for their implementation, for example, software data loading, user-modifiable software, or multiple-version dissimilar software?	Yes, No, N/A
12	A-2:3 11.10 i	Do the software architecture and/or detailed design data include descriptions of the software components, whether they are new or previously developed, and, if previously developed, reference to the baseline from which they were taken?	Yes, No, N/A
13	A-2:3 11.10 k	If the system contains deactivated code, does the design description data include a description of the deactivated function(s) and the means to ensure that the deactivated code cannot be enabled in the target computer?	Yes, No, N/A
14	A-2:3 11.10 l	Do the software architecture and/or detailed design data include rationale for design decisions, especially if those decisions impact safety-related system requirements?	Yes, No, N/A

Appendix J – C Source Code Checklist

No.	DO-178C[2] Ref	Question	Answer Options
1	A-5:1	Does the source code comply with low-level requirements? *The objective is to ensure that the source code is accurate and complete with respect to the software low-level requirements, and that no source code implements an undocumented function.*	Yes, No, N/A
2	A-5:2	Does the source code comply with the software architecture? *The objective is to ensure that the source code matches the data flow and control flow defined in the software architecture.*	Yes, No, N/A
3	A-5:4	Does the source code conform to standards? *The objective is to ensure that the Software Coding Standards were followed during the development of the code and that deviations to the standards are justified.*	Yes, No, N/A
4	A-5:5	Are the source codes traceable to the software low-level requirements? *The objective is to ensure that the software low-level requirements were developed into source code.* *Is the function name and/or header file name that describes the source code captured as individual objects in a DOORS proxy module and is traceability established from those objects to software low-level requirements?*	Yes, No, N/A
5	11.21.c	Are the low-level requirements traceable to the source code? *The objective is to ensure that the software low-level requirements were developed into source code. This ensures the bi-directional traces between the low-level requirements and the source code.*	Yes, No, N/A
6	A-5:6	Is the source code accurate and consistent? *This objective will be satisfied by answering items 6, 7, 8, 9, 10, 11, 12 and 19.*	Yes, No, N/A
7	6.3.4 f	Is exception handling correct and consistent?	Yes, No, N/A
8	6.3.4 f	Is handling of stack usage correct and consistent?	Yes, No, N/A
9	6.3.4 f	Is handling of resource contention correct and consistent?	Yes, No, N/A
10	6.3.4 f	Is handling of worst-case execution timing correct and consistent?	Yes, No, N/A
11	6.3.4 f	Are all variables initialized before use?	Yes, No, N/A
12	6.3.4 f	Has data corruption due to task or interrupt conflicts been removed?	Yes, No, N/A
13	6.3.4 f	Have all unused variables or constants been removed?	Yes, No, N/A
14	N/A	Data dictionary and code agree and are both complete?	Yes, No, N/A

15	N/A	Unexpected floating point values are prevented or detected?	Yes, No, N/A
16	N/A	Are global variables updated or modified as expected by the design?	Yes, No, N/A
17	A-5:7	Has the review and/or analysis ensured that the results of the integration process are complete and correct? *This could be performed by a detailed examination of the linking and loading data and memory map. The topics should include:* *a. Incorrect hardware addresses.* *b. Memory overlaps.* *c. Missing software components.*	Yes, No, N/A
18	N/A	Are all array index (subscript) values bounded and correct (no off-by-one errors)?	Yes, No, N/A
19	N/A	Are string indices bounded when indexing into strings?	Yes, No, N/A
20	N/A	Data is typed correctly and consistently?	Yes, No, N/A
21	N/A	Are parameters to functions or procedures are passed and returned properly?	Yes, No, N/A
22	N/A	Are data items that are order-dependent in the correct sequence?	Yes, No, N/A
		Control flow	
23	N/A	Are all iterative sections of code (loops, etc.) guaranteed to terminate?	Yes, No, N/A
24	6.3.4 f	Is handling of worst-case execution timing correct and consistent?	Yes, No, N/A
25	N/A	In tasking, will the program, module or procedure eventually terminate?	Yes, No, N/A
26	N/A	In a control loop index, are there no “off by one” errors, either one too many or one too few? (e.g. watch for starting indexes with 0 vs. 1)	Yes, No, N/A
27	N/A	Are there no instances of in-exhaustive/open-ended decisions? (e.g. if an input parameter’s expected values are 1, 2, or 3, does the logic assume that must be 3 if it’s not 1 or 2?) If so, is this assumption correct?)	Yes, No, N/A
28	N/A	Are there no instances of dead code? *Provide rationale for the justification of the dead code.*	Yes, No, N/A
		Compiler	
29	N/A	Have all affected files been compiled error-free?	Yes, No, N/A
30	N/A	Have any compiler/linker warnings been reviewed to verify that they will not adversely impact execution?	Yes, No, N/A
31	N/A	Have all warnings or errors justified?	Yes, No, N/A

Appendix K – Requirement Based Test Case and Test Procedure Checklist

No.	DO-178C[2] Ref	Question	Answer Options
1	11.13 b	Does each test case include a purpose, set of inputs, conditions, expected results to achieve the required coverage criteria, and the pass/fail criteria?	Yes, No, N/A
2	A-6:1,3 6.4.2.1 a	Normal Range Test Cases: Are real and integer input variables exercised using valid equivalence classes and boundary values?	Yes, No, N/A
3	A-6:1,3 6.4.2.1 b	Normal Range Test Cases: For time-related functions, such as filters, integrators and delays, have multiple iterations of the code been performed to check the characteristics of the function in context?	Yes, No, N/A
4	A-6:1,3 6.4.2.1 c	Normal Range Test Cases: For state transitions, are test cases developed to exercise the transitions possible during normal operation?	Yes, No, N/A
5	A-6:1,3 6.4.2.1 d	Normal Range Test Cases: For software requirements expressed by logic equations, are normal range test cases developed to verify the variable usage and the Boolean operators?	Yes, No, N/A
6	A-6:2,4 6.4.2.2 a	Robustness Test Cases: Are real and integer input variables exercised using equivalence class selection of invalid values?	Yes, No, N/A
7	A-6:2,4 6.4.2.2 b	Robustness Test Cases: Has system initialization been exercised during abnormal conditions?	Yes, No, N/A
8	A-6:2,4 6.4.2.2 c	Robustness Test Cases: Have the possible failure modes of incoming data been determined, especially complex, digital data strings from an external system?	Yes, No, N/A
9	A-6:2,4 6.4.2.2 d	Robustness Test Cases: For loops where the loop count is a computed value, have test cases been developed to attempt to compute out-of-range loop count values?	Yes, No, N/A
10	A-6:2,4 6.4.2.2 e	Robustness Test Cases: Have test cases been developed to test the protection mechanisms for exceeded frame times respond correctly?	Yes, No, N/A
11	A-6:2,4 6.4.2.2 f	Robustness Test Cases: For time-related functions, such as filters, integrators and delays, have test cases been developed for arithmetic overflow protection mechanisms?	Yes, No, N/A
12	A-6:2,4 6.4.2.2 g	Robustness Test Cases: For state transitions, are test cases developed to provoke transitions that are not allowed by the software requirements?	Yes, No, N/A

13	N/A	Is the standard test case template format followed?	Yes, No, N/A
14	11.21.d	Are the test cases traceable to High-level Requirements?	Yes, No, N/A
15	11.21.d	Are the test cases traceable to Low-level Requirements?	Yes, No, N/A
16	11.21.d	Are the High-Level requirements traceable to test cases? *This ensures the bi-directional traces between the high-level requirements and test cases established.*	Yes, No, N/A
17	11.21.d	Are the Low-Level requirements traceable to test cases? *This ensures the bi-directional traces between the low-level requirements and test cases established.*	Yes, No, N/A
18	A-7:3,4 6.4.4.1	Do the results of the requirements traceability analysis show that test cases exist for each software high-level and low-level requirement? *Have the test cases been linked to all the applicable High-level and low-level requirement(s), and has the analysis been completed to ensure the test cases also meet the high-level and low-level requirement(s)?*	Yes, No, N/A
19	6.4.2 c A-7:1	Is test procedure generated from test case?	Yes, No, N/A
20	11.21.e	Are the test procedures traceable to test cases?	Yes, No, N/A
21	11.21.e	Are the test cases traceable to test procedures? *This ensures the bi-directional traces between the test procedures and test cases are established.*	Yes, No, N/A
22	NA	If the requirement is verifiable by Analysis, does the Analysis follow the template?	Yes, No, N/A
23	NA	If the requirement is verifiable by Analysis, does the Analysis ensure that the code is comply with requirements?	Yes, No, N/A
24	NA	If the requirement is verifiable by Analysis, does the Analysis ensure that the pass/fail criterion is identified properly?	Yes, No, N/A
25	N/A	Has the procedure been dry run and dry run test results are attached as a reference?	Yes, No, N/A

Appendix L – Structural Coverage Analysis Checklist

No.	DO-178C[2] Ref	Question	Answer Options

1	A-7:7	Has the test coverage of software structure (statement coverage) been achieved? *N/A for Level D or E.* *In order to answer Yes, all uncovered statements must be analyzed to provide an acceptable reason code for not being covered. If an acceptable reason code does not apply, you must answer No and supply a Change Request that documents the deficiency. Reference the SDP for a list of acceptable reason codes.*	Yes, No, N/A
2	A-7: 7	Has the test analysis indicate that test coverage of software structure (statement coverage) been achieved?	Yes, No, N/A
3	N/A	If the missing coverage is for Defensive Code, does the analysis explain: Why the code can't be executed? What would happen if the code was executed? Why can't the code be covered by a requirements based test?	Yes, No, N/A
4	N/A	If the missing coverage is determined to be missing test cases, have the test cases been updated to achieve the needed coverage?	Yes, No, N/A
5	N/A	If the missing coverage is determined to be missing requirements, have the requirements and/or test cases been updated to achieve the missing coverage?	Yes, No, N/A
6	N/A	If the missing coverage is for Category I Deactivated Code as specified in the SDP, has a test or an analysis been created for the deactivation method and there is a requirement(s) for that method?	Yes, No, N/A
7	N/A	If the missing coverage is for Category I or II Deactivated Code as specified in the SDP, does the analysis explain: Why the code can't be executed? What would happen if the code was executed? Why can't the code be covered by a requirements based test (if analysis is provided)?	Yes, No, N/A
8	N/A	If the missing coverage is for Category II Deactivated Code as specified in the SDP, has a requirements based test been created to cover the non-covered source code?	
9	N/A	If the missing coverage is determined to be Extraneous code or Dead Code, has the code been removed?	Yes, No, N/A
10	N/A	If the missing coverage is analyzed for test coverage by manual analysis, does the analysis list the requirement, the test case and the individual test steps in the test case to explain the test coverage?	Yes, No, N/A
11	N/A	Does each missing statement only have an analysis Code and have the applicable analysis write-up as specified in the SDP?	Yes, No, N/A

12	A-7: 7	Has the coverage analysis indicate that test coverage of software structure (statement coverage) been achieved?	Yes, No, N/A
13	NA	Do the functional test results match (i.e. identical pass/fails) the structural test results?	Yes, No, N/A
14	NA	For each log file that is listed, does the configuration data match what is currently in Subversion? This ensures that the coverage report was generated from the most recent run of each test.	Yes, No, N/A

Appendix M – Test Result Checklist

No.	DO-178C[2] Ref	Question	Answer Options
1	A-7:2 6.3.6 c	Are the test results correct and have discrepancies between actual and expected results been explained? *If there are discrepancies, has a Change Request been written and submitted to document the test failure(s)?*	Yes, No, N/A
2	A-2:7 A-6:5 5.4.1 a 6.4.3 a	Is the executable object code compatible with the target computer? *Has the Executable Object Code been successfully loaded into the target hardware for the hardware/software integration?*	Yes, No, N/A
3	A-6:1 6.4.2.1 6.4.3	Does the executable object code comply with high-level requirements? *Did all normal test cases related to testing high-level requirements pass? If the normal test cases do not pass, indicate "No" and list the CR related to the test failure in the Comments field.*	Yes, No, N/A
4	A-6:2 6.4.2.2 6.4.3	Is the executable object code robust with high-level requirements? *Did all robustness test cases related to testing high-level requirements pass? If the robustness test cases do not pass, indicate "No" and list the CR related to the test failure in the Comments field.*	Yes, No, N/A
5	A-6:3 6.4.2.1 6.4.3	Does the executable object code comply with low-level requirements? *Did all normal test cases related to testing low-level requirements pass? If the normal test cases do not pass, indicate "No" and list the CR related to the test failure in the Comments field.*	Yes, No, N/A
6	A-6:4 6.4.2.2 6.4.3	Is the executable object code robust with low-level requirements? *Did all robustness test cases related to testing low-level requirements pass? If the robustness test cases do not pass, indicate "No" and list the CR related to the test failure in the Comments field.*	Yes, No, N/A
7	N/A	Were the test(s) and all supporting files required for the test under configuration control prior to the test being executed and was this data recorded?	Yes, No, N/A
8	N/A	If a test has failures, is each test step that failed listed in the results along with a CR reference as well as with the proper justification why it's ok to not correct the test failure?	Yes, No, N/A

9	N/A	Does the test result file conform to the test results defined in the verification artifact document? • Target Hardware where the test was executed • Test execution date and time • Verifier name • Total number of conditions tested • Total number of conditions passed • Total number of conditions failed • Expected and actual outputs • Overall Pass/Fail status	Yes, No, N/A
10	11.21.f	Are the test results traceable to test procedures?	Yes, No, N/A
11	11.21.f	Are the test procedures traceable to test results? *This ensures the bi-directional traces between the test procedures and test cases are established.*	Yes, No, N/A

