INSPECTION PROCEDURE 35710

QUALITY ASSURANCE INSPECTION OF SOFTWARE USED IN NUCLEAR APPLICATIONS

Effective Date: 01/30/2018

PROGRAM APPLICABILITY: 2502, 2507, 2508

35710-01 INSPECTION OBJECTIVES

01.01 To verify that safety-related software used for Digital Instrument and Control (DI&C) and Design and Analysis applications is developed in accordance with a Quality Assurance Program (QAP) that complies with the requirements of Appendix B to Title 10 of the *Code of Federal Regulations* (10 CFR) Part 50. Specifically, but not limited to:

Criterion II, “Quality Assurance Program,” as it relates to unique aspects of software project management and organizational processes; software quality assurance (QA) processes; software verification and validation processes; and software configuration management processes.

Criterion III, “Design Control,” as it relates to activities that should be specified in design documents, and the design control measures for verifying or checking the adequacy of the design that are unique to software;

Criterion V, “Instructions, Procedures, and Drawings,” as it relates to guidance for I&C activities affecting quality that may warrant specific documented procedures and guidance on the control of I&C documents that prescribe activities affecting quality specific to software;

Criterion XI, “Test Control,” as it relates to aspects of the testing program specific to software; and

Criterion XVI, “Corrective Action,” as it relates to identifying and correcting failures, malfunctions, and anomalies throughout the software lifecycle.

01.02 To verify the basic activities; which include translating a conceptual design into system requirements, developing a system design, implementing the design into hardware and software functions, testing the functions to ensure system requirements have been implemented correctly, installing the system and performing acceptance testing, operating and maintaining the system, and retiring the system upon the end of use, are integrated with organizational and management processes throughout a prescribed software lifecycle.

01.03 This procedure is to be used in combination with the following inspection procedures when applicable: Inspection Procedure (IP) 35017 “Quality Assurance Program Implementation During Construction and Pre-Construction Activities,” and IP 43004, “Inspection of Commercial-Grade Dedication Programs” and IP 65001 Attachment 65001.22, “Inspection of Digital Instrumentation and Control (DI&C) System/Software Design Acceptance Criteria (DAC) – Related to ITAAC [Inspections, Tests, Analyses and Acceptance Criteria].”

35710-02 INSPECTION REQUIREMENTS

02.01 Assess whether the activities related to the software development coincide with the specified software lifecycle.

02.02 Assess whether the activities documented in the software life cycle adequately address the following minimum basic activities/phases:

- Requirements
- Design
- Implementation
- Integration
- Testing
- Installation
- Operation
- Maintenance

02.03 Review the implementing instructions, procedures, plans, and policies in place for the QA, verification and validation, and configuration management controls for the development of software to be used in DI&C safety systems.

35710-03 INSPECTION GUIDANCE

03.01 Requirements: The requirements phase consists of developing a description of what the software/DI&C system must accomplish. Assess the requirements activities of the software lifecycle by performing the following:

a. Verify that the software design requirements are documented and that they incorporate applicable regulatory requirements, standards and codes.

b. Verify that the requirements documentation specifies the operating system, functionality, performance characteristics, interfaces, installations considerations, design constraints, and security constraints.

c. Verify that a formal process is documented and implemented to ensure changes to software requirements are evaluated, reviewed, approved and documented.

d. Select a sample of requirements to verify the implementation of controls for traceability by performing the following:

i. Verify that design bases are adequately translated into requirements that are documented within specifications, drawings, procedures, or instructions, and each requirement has a unique identifier.

ii. Verify that requirements can be traced from the bases document that they were derived from, to the software code implementing them, and to the test case that tests them.

iii. Verify that changes to the software requirements and software design maintain traceability throughout subsequent documentation.

03.02 Design: The design phase consists of translating requirements into a hardware/software architecture. Evaluate the design activities of the software life cycle by performing the following:

a. Verify that procedures are implemented to ensure design requirement documentation is reviewed, approved, baselined, updated as necessary, and placed under configuration control.

b. Verify that a process is implemented to establish a software baseline at the completion of each design activity.

c. Verify that procedures are implemented to ensure that changes made to the software are evaluated, reviewed, approved, and documented. Verify the documentation includes provisions for documenting a description of the change, rationale for the change, identification of the software baseline affected by the change, and status of the change throughout the implementation process.

03.03 Implementation: The implementation phase consists of translating the completed software design into code. Assess the implementation activities of the software lifecycle by performing the following:

a. Verify that implementation activities, such as the creation of an executable code, development of operation documentation, software unit testing, and management of software releases are completed in accordance with a documented implementation plan.

b. Verify that procedures are established and implemented for compliance with coding rules, methods, and standards.

03.04 Integration: The integration phase consists of combining software components and hardware components into a single system. Assess the integration activities of the software lifecycle by performing the following:

a. Verify that the plans and methods for integrating function divisions of software (units) are adequately documented. The plan should include a schedule, resource and staffing estimates, and criteria for the commencement of software integration. The software integration plan should also identify what is being integrated, define the integration environment, discuss the management of interfaces, define the integration sequence, and discuss the qualification testing to verify integration has been completed satisfactorily.

b. Verify that there are provisions in procedures to ensure the complete integration of all software units and comprised software modules or any other division of functional parts.

c. Verify that software integration test activities and tasks; primary test methods and standards; test cases; test coverage; and acceptance criteria are documented in accordance with Section 3.05.

03.05 Testing: The software testing phase consists of unit testing, integration testing, validation testing, and installation (acceptance) testing. For DI&C systems, the testing phase includes software testing, software integration testing, software qualification testing, system integration testing, and system qualification testing. Evaluate the testing activities of the software life cycle by performing the following:

a. Verify that there are provisions documented in procedures to ensure that all software requirements are covered by acceptance testing.

b. Verify that documentation supporting software testing includes the following:

i. Qualifications, duties, responsibilities, and skills required of persons and organizations assigned to testing activities

ii. Special conditions and controls, equipment, tools, and instrumentation needed for the accomplishment of testing

iii. Test instructions and procedures that incorporate the requirements and acceptance limits in applicable design documents

iv. Test prerequisites and the criteria for meeting these requirements and acceptance limits

v. Test items and the approach taken by the testing program

vi. Test logs, test data, and test results

vii. Acceptance criteria

viii. Test records that indicate the identity of the tester, the type of observation made, the results and acceptability, and the action taken in connection with any deficiencies

ix. Test plans, test activities and task, test cases, and test coverage test methods and standards

c. Verify that the results of testing are documented, reviewed, analyzed and approved, by a qualified individual to ensure test requirements have been fulfilled.

d. Verify that there is a documented method to identify and resolve discrepancies between actual and expected integration test results.

e. Assess whether the process established to incorporate changes to the software due to test results, is adequate to ensure that all test anomalies are documented and resolved.

f. Anomalies discovered during testing may impact system and software requirements. Verify the actions taken to address testing anomalies include revision to system and software requirement documentation and subsequent design documentation as necessary.

g. Verify that DI&C system testing is conducted on a completely integrated system, in which all hardware and software functionality has successfully passed integration testing and have been combined into one final system.

03.06 Installation: The installation phase consists of installing and testing the software in its operational environment. Assess the system testing activities of the software lifecycle by performing the following:

a. Verify that there are provisions documented in procedures for modifications to the software made during installation.

b. Verify that procedures are established and implemented for the performance of acceptance testing to demonstrate the installed system will perform its intended safety function as described in the system design basis.

c. Verify that acceptance test activities and tasks; primary test methods and standards; test cases; test coverage; and acceptance criteria are documented in accordance with Section 3.05.

d. Verify that procedures are implemented to document and resolve conditions that deviate from expectations based on requirements specifications, design documents, user documents, or standards prior to placing the system into operation.

03.07 Operation and Maintenance: The operation and maintenance phase consists of ensuring the continued use and operation of the software as designed. Assess the operation activities of the software lifecycle by performing the following:

a. Verify that documentation for the methods, plan, and deployment of the software or DI&C system, at minimum, include the following:

i. Documentation to support the operations, including user manuals, configuration control documents, instructions, procedures, and other associated documentation

ii. A description of the functions that the system is to perform and general discussion of the means to carry out those functions

iii. The controls needed over operation activities to prevent unauthorized changes to hardware, software, and system parameters

iv. Specification of the monitoring activities needed to detect unauthorized access to the system

v. Contingency plans needed to ensure appropriate response to control of access issues

vi. A description of the facilities used to operate the software

vii. A description of the procedures for executing the software in all operating modes and procedures for ensuring the software state is consistent with the plant operating mode at all times

viii. A description of the backup procedures for data and code and the intervals at which back up should occur

ix. A comprehensive list of the error messages, a description of the error indication, the probable interpretation of the error indication, and steps to be taken to resolve the error

x. Controls for continuously monitoring I&C safety system performance to ensure it is consistent with pre-established system performance measures

b. Verify that procedures have been established for monitoring the system's performance, recording problems for analysis, taking corrective and preventative actions, and confirming restored capability after servicing. Verify that procedures include instructions for documenting, evaluating, correcting, and reporting software errors. Evaluation should include how an error impacts previous use of the software and the development process.

c. Verify that there are provisions included in procedures to prohibit changes made to the software during maintenance that improve the performance or other attributes or adapt the design outputs to a modified environment. These changes are considered design changes and should be done in accordance with the software configuration management plan. Verify that maintenance is limited to the process of modifying a software design output to repair nonconforming items or implementing pre-planned actions necessary to maintain performance.

03.08 Verification and Validation: Assess the verification and validation activities of the software lifecycle by performing the following:

a. Verify that procedures are established and implemented for performing design reviews, alternate calculations, or testing to verify the adequacy of the software design. Verify the verification of the software design is performed by qualified individuals who were not responsible for or involved in the software design.

b. Verify that procedures are established and implemented for management reviews, technical reviews, inspections, walkthroughs, and audits.

c. Verify that procedures are established for the documentation and resolution of all non-conformances identified during the software development lifecycle.

d. Verify that procedures are established for problem identification, extent of condition, and risk mitigation for issues that have the potential to significantly impact the system quality.

e. Verify that measures are established for conducting reviews which ensure conformance of the software to design requirements and satisfactory completion of the software development activities/phases.

03.09 Configuration Management Processes: Assess the configuration management activities of the software lifecycle by performing the following:

a. Verify that procedures are established and implemented for the control of appropriate records of software development activities which include:

i. Identification and control of all software designs and code

ii. Identification and control of all software design functional data (e.g., data templates and data bases)

iii. Identification and control of all software design interfaces

iv. Control of all software design changes

v. Control of software documentation (e.g., user, operating, and maintenance documentation)

vi. Control of software vendor development activities for the supplied safety system software

vii. Control and retrieval of qualification information associated with software designs and code

viii. Software configuration audits

ix. Status accounting

b. Verify that provisions are included in procedures ensure software tools used to support system development and verification and validation processes are controlled under configuration management.

35710-04 RESOURCE ESTIMATE

Inspection resources necessary to complete this inspection procedure are estimated to be 160 hours of direct inspection per facility.

35710-05 REFERENCES

1. 10 CFR Part 50, Appendix B, "Quality Assurance Criteria for Nuclear Power Plants and Fuel Reprocessing Plants."

2. ANSI/ASME NQA-1, "Quality Assurance Program Requirements for Nuclear Facility Applications"

3. Manual Chapter 2502, “Construction inspection Program: Pre-Combined License (Pre-COL) Phase”

4. Manual Chapter 2507, “Vendor Inspections”

5. Manual Chapter 2508, “Construction Inspection Program: Design Certification.”

6. Regulatory Guide 1.152,"Criteria for Use of Computers in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission, 2006 (ML053070150)

7. Regulatory Guide 1.168, "Verification, Validation, Reviews and Audits for Digital Computer Software Used in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission

8. Regulatory Guide 1.169. "Configuration Management Plans for Digital Computer Software Used in Safety Systems of Nuclear Power Plants." Office of Nuclear Regulatory Research, U.S. Nuclear Regulatory Commission

9. Regulatory Guide 1.170, “Software Test Documentation for Digital Computer Software Used in Safety Systems of Nuclear Power Plants”

10. Regulatory Guide 1.171, “Software Unit Testing for Digital Computer Software Used in Safety Systems of Nuclear Power Plants”

11. Regulatory Guide 1.172, “Software Requirements Specifications for Digital Computer Software Used in Safety Systems of Nuclear Power Plants”

12. Regulatory Guide 1.173, “Developing Software Life Cycle Processes for Digital Computer Software Used in Safety Systems of Nuclear Power Plants”

13. Information Notice 86-77, “Computer Program Error Report Handling,” issued August 28, 1986

14. IEEE Std. 7-4.3.2-2003, "IEEE Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations"

15. IEEE Std. 603-1991, "IEEE Standard Criteria for Safety Systems for Nuclear Power Generating Stations"

16. IEEE Std. 730-2002, “IEEE Standard Criteria for Software Quality Assurance Plans”

17. IEEE Std. 828-1990, “IEEE Standard for Configuration Management Plans”

18. IEEE Std. 829-1983, “IEEE Standard for Software Test Documentation”

19. IEEE Std. 830-1993, “IEEE Recommended Practice for Software Requirements Specifications”

20. IEEE Std. 1008-1987, “IEEE Standard for Software Unit Testing”

21. IEEE Std. 1012-1998, "IEEE Standard for Software Verification and Validation Plans"

22. IEEE Std. 1028-1997, “IEEE Guide to Software Configuration Management”

23. IEEE Std. 1074-1995, "IEEE Standard for Developing Software Life Cycle Processes"

END

Attachment:
Revision History Table

Attachment 1: Revision History for IP 35710

Commitment Tracking Number	Accession Number Issue Date Change Notice	Description of Change	Description of Training Required and Completion Date	Comment Resolution and Closed Feedback Form Accession Number (Pre-Decisional, Non-Public Information)
N/A	ML17278AA510 01/30/18 CN 18-002	Initial issue to verify that safety-related software used for Digital Instrument and Control (DI&C) and Design and Analysis applications is developed in accordance with a Quality Assurance Program (QAP) that complies with the requirements of Appendix B to Title 10 of the *Code of Federal* Regulations (10 CFR) Part 50.	N/A	ML17278A511