

arseam



arseam

T9S900SQAPB_Redacted.pdf

01/06/20 10:42 AM

Xerox® WorkCentre® 7855

Software Quality Assurance Plan

TRIGA Control System

NRAD Reactor Instrumentation and Control Console
Replacement Project
Contract No. 99551

T9S900SQAP

December 2010

ISSUE/RELEASE SUMMARY

T-1485 (REV. 10/99)

<input type="checkbox"/> R&D <input type="checkbox"/> DV&S <input checked="" type="checkbox"/> DESIGN <input type="checkbox"/> N/A	APPVL LEVEL 1	SAFE.CLS	QA LVL/CLS	DOC. TYPE	PROJECT	DOCUMENT NO. T9S900SQAP	REV. B
		EIS	2	P	01015		
		SYSTEM	SEIS. CLS.	SEIS. CAT.	E. CLASS		
		900	N/A	N/A	N/A		

TITLE:
Software Quality Assurance Plan - NRAD Reactor Instrumentation and Control Console

CM APPROVAL/ DATE	REV.	PREPARED BY	APPROVAL(S)			REVISION DESCRIPTION
			ENGINEERING	QA	PROJECT	
V. Dixon 11/10/10	A	H. Staines	H. Walker	M. Miller	R. Ray	Initial issue
<i>V. Dixon</i> 12/10/10	B	S. Babb <i>S. Babb</i> 12/09/10	H. Walker <i>H. Walker</i> 12/10/10	M. Miller <i>M. Miller</i> 12/10/10	R. Ray <i>R. Ray</i> 12/10/10	Incorporate customer comments

USE CONTINUATION SHEET		NEXT INDENTURED DOCUMENT (S)
* See List of Effective Pages		N/A
APPROVAL BY GENERAL ATOMICS		COMPUTER PROGRAM PINs N/A
APPROVAL BY CUSTOMER		

☐ NO QA PROPRIETARY INFORMATION

TABLE OF CONTENTS

1.	INTRODUCTION.....	4
1.1	PURPOSE.....	4
1.2	SCOPE.....	4
1.3	SYSTEM OVERVIEW.....	4
1.4	REGULATORY AND STANDARDS COMPLIANCE.....	5
2.	REFERENCE DOCUMENTS.....	5
2.1	GOVERNMENT REGULATIONS, GUIDANCE AND REPORTS.....	5
2.2	STANDARDS/PROCEDURES.....	5
2.3	TRIGA CONTROL SYSTEM PROJECT DOCUMENTS.....	6
2.4	DEFINITIONS AND ACRONYMS.....	6
2.4.1	<i>Definitions</i>	6
2.4.2	<i>Acronyms</i>	8
3.	MANAGEMENT OF THE SOFTWARE QA FUNCTION.....	8
3.1	ORGANIZATION.....	8
3.1.1	<i>TRIGA Control System Project Roles and Responsibilities (Internal Structure)</i>	9
3.2	TASKS, GENERAL DESCRIPTION.....	9
3.2.1	<i>Task: Evaluate the Project Planning and Control Process</i>	9
3.2.2	<i>Task: Evaluate Software Tools and Computer Programs</i>	9
3.2.3	<i>Task: Evaluate the System Requirements Process</i>	10
3.2.4	<i>Task: Evaluate the System Design Process</i>	10
3.2.5	<i>Task: Conduct Reviews of Software Products</i>	10
3.2.6	<i>Task: Evaluate the Software Requirements Process</i>	10
3.2.7	<i>Task: Evaluate the Software Design Process</i>	10
3.2.8	<i>Task: Evaluate the Unit Coding/Unit Testing Process</i>	11
3.2.9	<i>Task: Evaluate the Software Integration Testing Process</i>	11
3.2.10	<i>Task: Evaluate the System Testing Process</i>	11
3.2.11	<i>Task: Evaluate the Installation and Checkout Process</i>	12
3.2.12	<i>Task: Audit CyberSecurity</i>	12
3.2.13	<i>Task: Media Certification</i>	12
3.2.14	<i>Task: Non-Deliverable Software Certification</i>	12
3.2.15	<i>Task: Evaluate Storage and Handling Process</i>	12
3.2.16	<i>Task: Evaluate Subcontractor Control</i>	12
3.2.17	<i>Task: Evaluate Deviations and Waivers</i>	13
3.2.18	<i>Task: Evaluate Configuration Management Process</i>	13
3.3	ROLES AND RESPONSIBILITIES IN QUALITY.....	13
3.3.1	<i>GA-ESI Director of Quality</i>	13
3.3.2	<i>TRIGA Control System Software Quality</i>	13
3.3.3	<i>TRIGA Control System Verification and Validation Team</i>	14
3.4	SOFTWARE QUALITY RESOURCE REQUIREMENTS.....	14
3.4.1	<i>Equipment and Infrastructure</i>	14
3.4.2	<i>Personnel</i>	14
4.	DOCUMENTATION.....	14
4.1	MINIMUM DOCUMENTATION.....	14
4.2	USER DOCUMENTATION OTHER DOCUMENTATION.....	14
5.	STANDARDS, PRACTICES, CONVENTIONS AND METRICS.....	15
5.1.1	<i>Documentation Standards</i>	15
5.1.2	<i>Design Standards</i>	15
5.1.3	<i>Coding Standards</i>	15

5.1.4	Commentary Standards	15
5.1.5	Testing Standards and Practices	15
5.1.6	Software Product Quality Metrics	15
5.1.7	Software Process Quality Metrics	15
6.	SOFTWARE REVIEWS	16
6.1	SOFTWARE REVIEWS, GENERAL	16
6.2	MINIMUM REQUIREMENTS	16
6.2.1	Software Specification Review (SSR)	16
6.2.2	Architecture Design Review or Preliminary Design Review(PDR)	16
6.2.3	Detailed Design Review or Critical Design Review (PDR)	16
6.2.4	Verification & Validation Plan Review	16
6.2.5	Functional Configuration Audit (FCA)	17
6.2.6	Physical Configuration Audit	17
6.2.7	In-Process Audits	17
6.2.8	Managerial Reviews	17
6.2.9	Software Configuration Management Plan Review	17
6.2.10	Post-Implementation Review	17
6.3	OTHER REVIEWS AND AUDITS	17
7.	SOFTWARE TESTING	17
7.1	TEST AUTOMATION	18
8.	PROBLEM REPORTING AND CORRECTIVE ACTION	19
9.	TOOLS, TECHNIQUES AND METHODOLOGIES	19
9.1	TOOLS TO SUPPORT THE SOFTWARE QUALITY EFFORT	19
9.2	TECHNIQUES TO SUPPORT THE SOFTWARE QA EFFORT	19
10.	MEDIA CONTROL	19
11.	SUPPLIER CONTROL	19
12.	PROCURED SOFTWARE AND SOFTWARE SERVICES	20
13.	OTHERWISE ACQUIRED SOFTWARE	20
14.	RECORDS COLLECTION, MAINTENANCE AND RETENTION	20
15.	TRAINING	20
16.	TRIGA CONTROL SYSTEM RISK MANAGEMENT	20
17.	GLOSSARY	20
18.	SQAP CHANGE PROCEDURE AND HISTORY	20

TABLE OF FIGURES

FIGURE 1: PROJECT EXTERNAL INTERFACES	8
---------------------------------------------	---

LIST OF TABLES

Table 1	Project Responsibilities	9
Table 2	Project Control Tools	19

1. INTRODUCTION

1.1 PURPOSE

The purposes of this Software Quality Assurance Plan (SQAP) are to define:

- The Software Quality organization for TRIGA Control System General Atomics Electronic Systems Inc. (GA-ESI)
- Software Quality tasks and responsibilities
- The standards, practices and conventions used to perform Software Quality activities; and
- The tools, techniques, and methods to support Software Quality activities and Software Quality reporting.

The TRIGA Control System Project includes:

- The CSC computer unit in master mode, which functions as a radiation data processor

1.2 SCOPE

This SQAP articulates Software Quality activities (including those of software quality engineering, software quality assurance and software testing) performed throughout the software development life cycle of the TRIGA Control System. This plan will define Software Quality functions for this project and specify the reporting activities of Software Quality to the GA-ESI Quality directorate, with communications links to the TRIGA Control System Project Manager and Engineering Manager.

A key goal of the Software Quality function is to verify that all software and documentation to be delivered meet all technical requirements. The Software Quality tasks defined herein shall be used to examine deliverable software and documentation to determine compliance with technical and regulatory compliance requirements.

1.3 SYSTEM OVERVIEW

The TRIGA Control System shall include instrumentation for monitoring reactor parameters during all operational states and for recording all variables important to reactor operation. It also shall manage all control rod movements taking into account the choice of operating mode and interlocks. The TRIGA Control System is a computer-based system, but includes dedicated hardwired displays and controls so that safe operation can continue should the computers become unavailable. There are two major system components, the Control System Console (CSC), Data Acquisition and Control (DAC).

1.4 REGULATORY AND STANDARDS COMPLIANCE

This SQAP is written to comply with the Quality Assurance Requirements contained in Appendix D of document SOW-8330 which sites compliance to ASME NQA-1-2000. The TRIGA project will also comply with internal GA-ESI Quality procedures and RMS Engineering Operating procedures and Quality Manual.

Section C.2 of Nuclear Regulatory Commission document RG 1.152 describes:

- A waterfall software development model
- A specific software development life cycle (SDLC) consisting of:
 - Concept phase
 - Requirements phase
 - Design phase
 - Implementation phase
 - Testing phase
 - Installation and Checkout phase
 - Operation phase
 - Maintenance phase
 - Retirement phase

This SQAP is also written in accordance with the requirements of GA-ESI Quality Manual section 22 *Software Control* and GA-ESI Quality Assurance Procedure QAP 22-02 *Software Quality Assurance Plan Implementation*. It is applicable to the first six SDLC phases listed above, from the Concept phase through the Installation and Checkout phase.

2. REFERENCE DOCUMENTS

2.1 GOVERNMENT REGULATIONS, GUIDANCE AND REPORTS

ISSUED BY	DOCUMENT IDENTITY	TITLE
NRC	10 CFR 50, Appendix B	Quality Assurance Criteria for Nuclear Plants and Fuel Reprocessing Plants
NRC	RG 1.152, Rev 2	Criteria for Digital Computers in Safety Systems of Nuclear Power Plants
NRC	RG 1.168, Rev. 1	Verification, Validation, Reviews, and Audits for Digital Computer Software Used in Safety Systems of Nuclear Power Plants
NRC	RG 1.170	Software Test Documentation for Digital Computer Software Used in Safety Systems of Nuclear Power Plants
NRC	RG 1.171	Software Unit Testing for Digital Computer Software Used in Safety Systems of Nuclear Power Plants
NRC	RG 1.172	Software Requirements Specifications for Digital Computer Software used in Safety Systems of Nuclear Power Plants
NRC	RG 1.173	Development Software Life Cycle Processes for Digital Computer Software Used in Safety Systems of Nuclear Power Plants

2.2 STANDARDS/PROCEDURES

ISSUED BY	DOCUMENT IDENTITY	TITLE
GA-ESI	S00008001	Quality Manual Rev. T
GA-ESI	S00008004	Software Standards and Procedures Manual
GA-ESI	SQAI1-7,9,10	Software Quality Assurance Instructions
GA-ESI	QAP-16-01	Corrective and Preventive Actions
GA-ESI	QAP- 22-02	Software Quality Assurance Plan Implementation

IEEE	IEEE STD 7.4.3.2 Section 5.3	Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations
IEEE	IEEE Std 610.12-1990	IEEE Glossary of Software Engineering Terminology
IEEE	IEEE Std 829-1998	IEEE Standard for Software Test Documentation
IEEE	IEEE Std 1008-1987	IEEE Standard for Software Unit Testing
IEEE	IEEE Std 1016-1998	IEEE Recommended Practice for Software Design Descriptions
IEEE	IEEE Std 730-2002	IEEE Guide to Software Quality Assurance Plans
IEEE	IEEE Std 16085-2006	Systems and Software Quality Engineering—Life Cycle Processes—Risk Management
GA-ESI	OP-4.0-160	Authorization of Deviations Procedure
ASME	NQA-1-2000 Subpart 2.7, Requirement 2, 3,11,13,18	Quality Assurance Requirements for Computer Software for Nuclear Facility Applications

2.3 TRIGA CONTROL SYSTEM PROJECT DOCUMENTS

ISSUED BY	DOCUMENT IDENTITY	TITLE
GA-ESI	TRP-1029-05	Technical Requirements & Approach
BEA/INL	SOW-8330, Rev 0	Replacement of the NRAD Instrumentation and Control Console
GA-ESI	T9S900SQAP, Rev. B	Software Quality Assurance Plan
GA-ESI	GATR-E-03, rev C	Validation Program
GA-ESI	T9S900CMP, Rev. A	Configuration Management Plan

2.4 DEFINITIONS AND ACRONYMS

2.4.1 Definitions

Code	(1) In software engineering, computer instructions and data definitions expressed in a programming language or in a form output by an assembler, compiler, or other translator. (2) To express a computer program in a programming language. [IEEE 610.12-1990]
Component	One of the parts that make up a system. A component may be hardware or software and may be subdivided into other components. Note: The terms "module", "component", and "unit" are often used interchangeably or defined to be sub-elements of one another in different ways depending upon the context. The relationship of these terms is not yet standardized. [IEEE 610.12-1990]
Configuration Management	A discipline applying technical and administrative direction and surveillance to: identify and document the functional and physical characteristics of a configuration item, control changes to those characteristics, record and report change processing and implementation status, and verify compliance with specified requirements. [IEEE 610.12-1990]
Design	(1) The process of defining the architecture, components, interfaces, and other characteristics of a system or component, (2) The result of the process in (1). [IEEE 610.12-1990]

Design specification (or design description)	A document that describes the design of a system or component. Typical contents include system or component architecture, control logic, data structures, input/output formats, interface descriptions, and algorithms. [IEEE 610.12-1990]
Engineering change	In configuration management, an alteration in the configuration of a configuration item or other designated item after formal establishment of its configuration identification. [IEEE 610.12-1990]
Functional configuration audit	An audit conducted to verify that the development of a configuration item has been completed satisfactorily, that the item has achieved the performance and functional characteristics specified in the functional or allocated configuration identification, and that its operational and support documents are complete and satisfactory. [IEEE 610.12-1990]
Interface	(1) A shared boundary across which information is passed. (2) A hardware or software component that connects two or more other components for the purpose of passing information from one to the other. [IEEE 610.12-1990]
Module	(1) A program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading; for example, the input to, or output from, an assembler, compiler, linkage editor, or executive routine. (2) A logically separable part of a program. [IEEE 610.12-1990]
Physical configuration audit	An audit conducted to verify that a configuration item, as built, conforms to the technical documentation that defines it. [IEEE 610.12-1990]
Project plan	A document that describes the technical and management approach to be followed for a project. The plan typically describes the work to be done, the resources required, the methods to be used, the procedures to be followed, the schedules to be met, and the way that the project will be organized. [IEEE 610.12-1990]
Requirements specification	A document that specifies the requirements for a system or component. Typically included are functional requirements, performance requirements, interface requirements, design requirements, and development standards. [IEEE 610.12-1990]
Software design description or SDD	A representation of software created to facilitate analysis, planning implementation, and decision making. The software design description is used as a medium for communicating software design information, and may be thought of as a blueprint or model of the system. [IEEE 610.12-1990]
Software development folder or SDF	A collection of material pertinent to the development of a given software unit or set of related units. Contents typically include the requirements, design, technical reports, code listings, test plans, test results, problem reports, schedules, and notes for the units. [IEEE 610.12-1990]
Software library	A controlled collection of software and related documentation designed to aid in software development, use, or maintenance. Types include master library, production library, software development library, software repository, system library. [IEEE 610.12-1990]
Test plan	A document describing the scope, approach, resources, and schedule of intended testing activities. It identifies test items, the features to be tested, the testing tasks, who will do each task, and any risks requiring contingency planning. [IEEE 829-1998]
Unit	(1) A separately testable element specified in the design of a computer software component. (2) A logically separable part of a computer program. (3) A software component that is not subdivided into other components. [IEEE 610.12-1990]

Version	(1) An initial release or re-release of a computer software configuration item, associated with a complete compilation or recompilation of the computer software configuration item. (2) An initial release or complete re-release of a document, as opposed to a revision resulting from issuing change pages to a previous release. [IEEE 610.12-1990]
---------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

2.4.2 Acronyms

CDR	Critical Design Review
DIU	Detector Interface Unit
FCA	Functional Configuration Audit
GA-ESI	General Atomics Electronic Systems Inc.
PCA	Physical Configuration Audit
PDR	Preliminary Design Review
QA	Quality Assurance
QAP	Quality Assurance Procedure
SCMP	Software Configuration Management Plan
SDLC	Software Development Life Cycle
SPMP	Software Project Management Plan
SQAI	Software Quality Instructions
SQAP	Software Quality Assurance Plan
SSR	Software Specification Review
V&V	Verification & Validation

3. MANAGEMENT OF THE SOFTWARE QA FUNCTION

3.1 ORGANIZATION

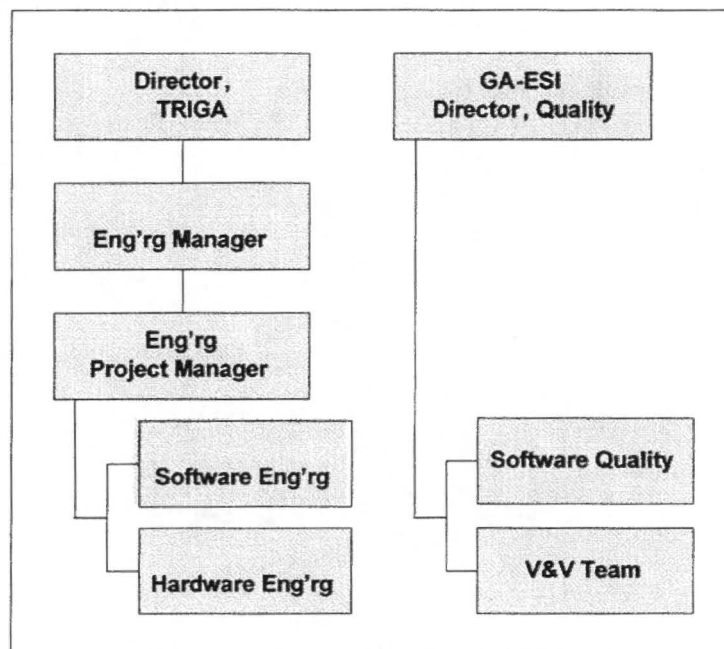


Figure 1: Project External Interfaces

3.1.1 TRIGA Control System Project Roles and Responsibilities (Internal Structure)

Position	Role/Responsibility
Project Manager	Performs project management through the SDLC; manages project scope, schedule and resources; manages organizational, developmental and supporting processes.
Engineering Manager	Provides software engineering and hardware engineering resources to the project. Reports to the project manager.
Software Quality	Provides quality engineering, quality assurance and testing services to the project. Reports to the GA-ESI Director of Quality
Software V&V Team	Provides verification and validation services to the project. Reports to the GA-ESI Director of Quality

Table 1: Project Responsibilities

Members of the TRIGA Control System development team shall have the ability to evaluate and/or monitor the quality of the software being developed. Also, the status and disposition of each item in the project anomaly database shall be available to members of the development team. This SQAP is authored and maintained by the TRIGA Control System Software Quality Engineer.

3.2 TASKS, GENERAL DESCRIPTION

Software Quality tasks are performed on the software development process and on software development products.

Software Quality tasks are performed in relationship to the software development activities taking place and several Software Quality tasks may be performed concurrently. A task is considered complete when the required actions and report(s) are satisfactorily completed and/or action items have been closed. General task descriptions for Software Quality are provided in the subsections which follow.

The listed tasks and subtasks address the requirements set forth in the GA-ESI Quality Instructions and Procedures Volume 1 (Quality Manual Rev T), GA-ESI Software Quality Assurance Instructions and provides adherence to contractually prescribed software quality requirements.

3.2.1 Task: Evaluate the Project Planning and Control Process

Project planning and control includes the creation of project management plans, measuring planned-versus-actual performance, and managing projects to successful conclusion. Software Quality will identify appropriate standards, guidelines, and other documents to support successful project execution. Software Quality will use a checklist to evaluate the Software Project Management Plan and report the results using a standard GA-ESI report form or a standard GA-ESI memorandum.

3.2.2 Task: Evaluate Software Tools and Computer Programs

Software Quality shall evaluate software tools and computer programs used during development or formal verification and validation of the TRIGA Control System Project software. Software tools and computer programs shall be assessed based on criteria described in ASME NQA-1-2000 Requirement 3, paragraph 401 and 402, Requirement 11, paragraph 400 and Part II Subpart 2.7 paragraph 601 and 602.

3.2.3 Task: Evaluate the System Requirements Process

The system requirements process includes: (1) Defining system requirements, (2) Analyzing requirements, and (3) Allocating functions to hardware and software. Software Quality shall:

- Verify that appropriate participants are specified for this task.
- Verify that system software requirements definition criteria are adequately specified.
- Verify that system software requirements review criteria are adequately specified.
- Verify that processes for defining, analyzing, and allocating system software requirements are followed.
- Verify that system software requirements traceability is established and maintained.
- Verify content, format and compliance of the SyRS.

3.2.4 Task: Evaluate the System Design Process

The system design process includes: (1) Defining system architecture, (2) Partitioning the system into subsystems and components, and (3) Allocating functions to hardware and software. Software Quality shall:

- Verify that appropriate participants are specified for this task.
- Verify that system software design criteria are adequately specified.
- Verify that system software design review criteria are adequately specified.
- Verify compliance of the system software design review process.
- Verify that the system software design change process is adequately specified.
- Verify that system software design traceability is established and maintained.
- Verify content, format and compliance of the system software design elements in the SyRS
- Review prototypes for content, format and compliance.

3.2.5 Task: Conduct Reviews of Software Products

TRIGA Control System software products shall be reviewed using GA-ESI SQAI -2 *Coding Standards and Conventions*, SQAI-7 *Coding/Design Walk-through Certification* and SQAI-10 *Review of Documentation Pending Release*.

3.2.6 Task: Evaluate the Software Requirements Process

The software requirements process includes: (1) Defining requirements, (2) Analyzing requirements, and (3) Allocating functions to hardware and software. Software Quality shall:

- Verify that appropriate participants are specified for this task.
- Verify that requirements definition criteria are adequately specified.
- Verify that requirements review criteria are adequately specified.
- Verify that processes for defining, analyzing, and allocating requirements are followed.
- Verify that requirements traceability is established and maintained.

3.2.7 Task: Evaluate the Software Design Process

The software design process includes: (1) Defining software architecture at the system level, and (2) Partitioning the software into subsystems, components, etc. The software design process is sometimes segregated into: (1) Top-level design and (2) Detailed design. Software Quality shall:

- Verify that appropriate participants are specified for this task.

- Verify that design criteria are adequately specified.
- Verify that Software Development Folders (SDFs) are created.
- Verify that design review criteria are adequately specified.
- Verify compliance of the design review process.
- Verify that the design change process is adequately specified.
- Verify that software safety and hazards have been adequately addressed.
- Verify that design traceability is established and maintained.

3.2.8 Task: Evaluate the Unit Coding/Unit Testing Process

Software implementation is synonymous with "programming" or "coding." The Implementation Phase is the activity in the SDLC in which the design is coded and implemented. Unit (or module) coding normally includes unit testing tasks by programmers. Software Quality shall:

- Verify that a complete set of testing documents exist for Component Testing.
- Review all testing documents for content, format and compliance.
- Verify that testing is performed in accordance with the Component test plan, test designs, test cases, test procedures and applicable standards.
- Perform periodic surveillance of testing, witness tests, and certify test results.
- Verify that defects found during testing are captured, verified, logged and developers are supported to resolve the defects.
- Verify that test logs are established and maintained.
- Verify that Software Development Folders (SDFs) are maintained.
- Verify that code traceability is established and maintained.

3.2.9 Task: Evaluate the Software Integration Testing Process

During this process software elements are combined and then tested to ensure that the combined software elements work together correctly. First, units/modules are incrementally-integrated and integration testing is repeatedly performed on incrementally-integrated units/modules. Eventually, integrated units/modules become software components. Software components are then incrementally combined and testing is performed on the incrementally-integrated software components. Integrated components eventually become software subsystems, and testing is performed on the incrementally-integrated subsystems. Subsystems eventually are combined to produce a system and the system testing is then possible. Software Quality shall:

- Verify that complete sets of Integration testing documents exist.
- Review all testing documents for content, format and compliance.
- Verify that testing is performed in accordance with the Integration test plan, test designs, test cases, test procedures and applicable standards.
- Perform periodic surveillance of testing, witness tests, and certify test results.
- Verify that defects found during testing are captured, verified, logged and developers are supported to resolve the defects.
- Verify that test logs are established and maintained.

3.2.10 Task: Evaluate the System Testing Process

During this process, software is tested as an integrated system. In the event that all but one subsystem is complete, the remaining subsystems are integrated, the missing subsystem is simulated and testing is performed as though the entire system is present. Software Quality

shall:

- Verify that complete sets of System testing documents exist.
- Review all testing documents for content, format and compliance.
- Verify that testing is performed in accordance with the System test plan, test designs, test cases, test procedures and applicable standards.
- Perform periodic surveillance of testing, witness tests, and certify test results.
- Verify that defects found during testing are captured, verified, logged and developers are supported to resolve the defects.
- Verify that test logs are established and maintained.

3.2.11 Task: Evaluate the Installation and Checkout Process

During the Installation and Checkout phase the software is installed into the target system, a checkout process is performed and acceptance testing is conducted. Software Quality shall:

- Verify that complete sets of acceptance testing documents exist for the TRIGA Control System Project.
- Review all testing documents for content, format and compliance.
- Verify that testing is performed in accordance with the Acceptance test plan, test designs, test cases, test procedures and applicable standards.
- Perform periodic surveillance of testing, witness tests, and certify test results.
- Verify that defects found during testing are captured, verified, logged and developers are supported to resolve the defects.
- Verify that test logs are established and maintained.

3.2.12 Task: Audit CyberSecurity

Software Quality shall conduct periodic audits to determine the effectiveness of the digital safety system security procedures for TRIGA Control System products as set discussed in RG 1.152.

3.2.13 Task: Media Certification

Software Quality shall verify that the media used for source code and object code are identified correctly, that the media is readable, and (if required) that the media autoloads properly.

3.2.14 Task: Non-Deliverable Software Certification

The project may use non-deliverable software in the development of deliverable software as long as the operation and support of the deliverable software after delivery does not depend on the non-deliverable software. Software Quality shall certify that the use of deliverable software is not dependent on non-deliverable software to execute. Software Quality shall verify that all non-deliverable software used on the project performs its intended functions.

3.2.15 Task: Evaluate Storage and Handling Process

Software Quality shall verify that there is a plan or set of procedures for storage and handling of media. Software Quality shall evaluate the plan or procedures for adequacy.

3.2.16 Task: Evaluate Subcontractor Control

Software Quality shall ensure that software development processes and products for deliverable software from subcontractors conform to contract requirements.

3.2.17 Task: Evaluate Deviations and Waivers

Software Quality shall be included in the review cycle of requests for deviations and waivers and support the processing of same in accordance with TRIGA Control System OP-4.0.160 *Authorization of Deviations*.

3.2.18 Task: Evaluate Configuration Management Process

Software Configuration Management is separately addressed in the TRIGA Control System Software Configuration Management Plan (SCMP). Software Quality shall:

- Ensure configuration baselines are established as planned and that Configuration Items include, but are not limited to:
 - Project Documentation
 - Computer programs (source code, etc.)
 - support software
- Verify that an adequate set of configuration management documents are available to support the TRIGA Control System Project. These documents shall address:
 - Configuration Identification
 - Configuration Status Accounting
 - Configuration Control
 - Configuration Audits
- Review configuration management documents for content, format and compliance.
- Verify that the performance of configuration management complies with configuration management documents, including operation of the software development library.
- Perform periodic configuration audits
- Perform periodic surveillance of the configuration management function to ensure the change control process includes:
 - initiation, evaluation and disposition of change requests
 - control and review/approval of changes prior to implementation or release.
 - requirements for re-test and acceptance of test results.

3.3 ROLES AND RESPONSIBILITIES IN QUALITY

3.3.1 GA-ESI Director of Quality

The GA-ESI Director of Quality is responsible for:

- Reviewing and approving the TRIGA Control System SQAP.
- Supporting the implementation of the quality program defined in the TRIGA Control System SQAP.
- Providing adequate resources for the performance of TRIGA Control System Software Quality and Software Testing functions.
- Representing the Quality directorate at the executive level at TRIGA Control System SDLC phase reviews

3.3.2 TRIGA Control System Software Quality

3.3.2.1 TRIGA Control System Software Quality

TRIGA Control System software quality is responsible for:

- providing objective insight into the maturity and quality of the software processes and associated work products on the TRIGA Control System Project
- Recording, monitoring and facilitating the resolution of TRIGA Control System Project quality issues.
- Ensuring compliance with the contents of this SQAP
- Conducting training in software quality and software testing, as necessary.
- Witnessing software testing and certifying the quality of deliverable software

3.3.3 TRIGA Control System Verification and Validation Team

The TRIGA Control System Verification and Validation (V&V) Team is responsible for the performance of TRIGA Control System verification and validation tasks in accordance with the TRIGA Control System SVVP.

3.3.3.1 TRIGA Control System Software Testing

The TRIGA Control System Software Testing group is responsible for:

- Composing testing documents for component testing, integration testing, system testing and acceptance testing in accordance with GATR-E-03 Instrumentation System Firmware Validation Program.
- Executing testing of deliverable software and associated documentation in accordance with TRIGA Control System testing documents
- Documenting the performance of testing
- Identifying and logging defects found during the execution of testing
- Supporting software developers in the defect resolution process

3.4 SOFTWARE QUALITY RESOURCE REQUIREMENTS

3.4.1 Equipment and Infrastructure

The GA-ESI Director of Quality shall ensure that adequate resources (work area, computing hardware, computing software, connectivity, etc.) to perform software quality and software testing tasks and functions.

3.4.2 Personnel

The GA-ESI Director of Quality shall ensure that adequate staff is available to perform software quality and software testing functions. Labor hour estimates for software quality and software testing, per task or per document, are enumerated in the previously-discussed appendices to the TRIGA Control System SPMP.

4. DOCUMENTATION

4.1 MINIMUM DOCUMENTATION

The following are minimum software project documentation requirements at the first formal software release:

- Software Requirements Specification
- Software Design Description
- Software Configuration Management Plan
- Software V&V Plan
- Software V&V Report

4.2 USER DOCUMENTATION OTHER DOCUMENTATION

Other Software Quality documents, either required by regulation or regulatory guide by the U.S. Nuclear Regulatory Commission, or as derived requirements, are also listed in an appendix to the SPMP, and not duplicated herein.

5. STANDARDS, PRACTICES, CONVENTIONS AND METRICS

5.1.1 Documentation Standards

TRIGA Control System Documentation will follow the existing GA-ESI engineering document review-and-approval process, and configuration-controlled versions will reside on a designated Document Control Center network directory.

5.1.2 Design Standards

Unless otherwise specified or disclosed, IEEE Std 1016-1998 is the guideline used for software design descriptions.

5.1.3 Coding Standards

The TRIGA Control System Project shall use GSA-ESI procedure S00008004 *Software Standards and Procedures Manual* as a guide for software code development.

5.1.4 Commentary Standards

The commentary standard for the TRIGA Control System Project is:

- Commentary shall follow the content and format of the respective standard for which the commentary is targeted.
- Commentary shall be written in U.S. English

5.1.5 Testing Standards and Practices

Unless otherwise specified or disclosed, NRC RG 1.171 is the standard for unit testing, and NRC RG 1.168 is the standard for integration, system and acceptance testing for the TRIGA Control System Project. Testing shall also conform to test requirements in ASME NQA-1-2000 Part 1 Requirement 3, 11 and subpart 2.7.

5.1.6 Software Product Quality Metrics

Unless otherwise specified or disclosed, IEEE Std 982.1-1988 and in IEEE Std 982.2-1988 are guidelines for software product quality metrics for the TRIGA Control System. Software product metrics are contained within six (6) subcategories:

- Errors, Faults, and Failures
- Mean-time-to-failure
- Completeness & Consistency
- Complexity
- Reliability Growth & Projection
- Remaining Products faults

Metrics from these subcategories will be selected and will be tailored, as needed, for the TRIGA Control System Project during project execution.

5.1.7 Software Process Quality Metrics

Unless otherwise specified or disclosed, IEEE Std 982.1-1988 and in IEEE Std 982.2-1988 are guidelines for software process quality metrics for the TRIGA Control System. Software process metrics are contained within (3) subcategories:

- Management Control
- Coverage Measures

- Risk, Benefits, and Cost Evaluation

Metrics from these subcategories will be selected and will be tailored, as needed, for the TRIGA Control System Project during project execution.

6. SOFTWARE REVIEWS

6.1 SOFTWARE REVIEWS, GENERAL

Unless otherwise specified or disclosed, IEEE Std 1028-1997 and ASME NQA-1-2000 Part II Subpart 2.7 paragraph 202 are the guidelines for software reviews for the TRIGA Control System Project. Planned software reviews are listed in the appendices to the SPMP and are scheduled from phase-to-phase during the SDLC. The standards specify:

- The purpose for each type of review
- Attendees, roles and responsibilities
- Review preparation
- Review procedures
- Entry criteria
- Review processes
- Exit criteria
- Outputs

Issues and action items resulting from reviews shall be recorded and tracked to closure. Meeting minutes, including issues and action items shall be documented and placed in the project folder.

6.2 MINIMUM REQUIREMENTS

6.2.1 Software Specification Review (SSR)

While alternate nomenclature from that of IEEE Std 730-2002 may be used to identify a review, a review is planned for several TRIGA Control System Project documents including:

- System Requirements Specification
- Software Requirements Specification
- Software Design Description

6.2.2 Architecture Design Review or Preliminary Design Review(PDR)

An Architecture Design Review is planned for the TRIGA Control System Software Design Description. This review may be alternatively described in the TRIGA Control System Project as a Preliminary Design Review (PDR). This review is cited in Appendices to the SPMP.

6.2.3 Detailed Design Review or Critical Design Review (PDR)

A Detailed Design Review is planned for TRIGA Control System Project Software Design Descriptions. This review may be alternatively described in the TRIGA Control System Project as a Critical Design Review (CDR). This review is cited in Appendices to the SPMP.

6.2.4 Verification & Validation Plan Review

A review is planned for the TRIGA Control System Project Software Validation & Verification Plan (SVVP). This review is cited in Appendix 1 to the SPMP.

6.2.5 Functional Configuration Audit (FCA)

Functional configuration audits are planned for TRIGA Control System Project software. The audits shall follow SQAI-3 Configuration Identification Audit procedure.

6.2.6 Physical Configuration Audit

A physical configuration audit (PCA) is planned for TRIGA Control System Project software. The audit shall follow SQAI-3 Configuration Identification Audit procedure.

6.2.7 In-Process Audits

The following in-process audits are planned for TRIGA Control System Software.

- Code vs. design documents
- Interface specifications (hardware and software)
- Functional requirements vs. design
- Functional requirements vs. test documents

These audits are cited in the appendices to the SPMP.

6.2.8 Managerial Reviews

SDLC phase reviews shall be presented as managerial reviews at the end of each development phase in accordance with IEEE 1028-1997. As the TRIGA Control System Project progresses, additional managerial reviews may be desired.

6.2.9 Software Configuration Management Plan Review

A TRIGA Control System Project Software Configuration Management Plan review is intended. This review is cited in Appendix 1 to the SPMP.

6.2.10 Post-Implementation Review

A post-implementation review shall be held concurrently with the SDLC phase review for the Installation & Checkout Phase, and the agenda for that review shall encompass the content for both the respective phase review and for the post-implementation review.

6.3 OTHER REVIEWS AND AUDITS

In the event that other reviews and/or audits are deemed necessary, each additional review/audit shall be enumerated in the appropriate appendix to the SPMP, scheduled and performed.

7. SOFTWARE TESTING

During the TRIGA Control System Project four types of software testing shall be conducted and comply with the requirements defined in NRC RG 1.168 and ASME NQA-1-2000 Part II Subpart 2.7 paragraph 402 and 402.1:

- Component Testing
- Integration Testing
- System Testing
- Acceptance Testing

Software Quality shall:

- Verify that complete sets of testing documents for each category of testing are composed in accordance with applicable standards.
- Review all testing documents for content, format and compliance.
- Verify that testing is performed in accordance with each test plan, test design specification, test case, test procedure and standards.
- Perform continuous surveillance of testing, witness tests, and certify test results.
- Verify that defects found during testing are captured, verified, logged and developers are supported to resolve the defects.
- Verify that test logs are established and maintained.

7.1 TEST AUTOMATION

To effect cost reduction and improve schedule performance in component testing and in regression testing the software testing group shall establish and sustain a test automation environment. This shall include the acquisition, installation, configuration and operation of automated test tools.

8. PROBLEM REPORTING AND CORRECTIVE ACTION

Problem Reporting and corrective actions shall follow GA-ESI OP-6.6-210 *Software Discrepancy Reporting* procedure for all software product related issues in development. Software process issues shall follow GA-ESI QAP-16-01 *Corrective and Preventive Action* procedure.

9. TOOLS, TECHNIQUES AND METHODOLOGIES

9.1 TOOLS TO SUPPORT THE SOFTWARE QUALITY EFFORT

<i>Tool</i>	<i>Purpose(s)</i>
Microsoft Excel	Budgeting
Microsoft Outlook	eMail Communications
Microsoft PowerPoint	Briefings
Microsoft Project	Schedule Tracking
Microsoft Word	Issue Tracking
Microsoft Word	Software Defect Tracking
Microsoft Word	Team Collaboration/Groupware
PolyPVCS	Software Configuration Management
Microsoft Word	Reporting
Visual	Labor Tracking

Table 2: Project Control Tools

9.2 TECHNIQUES TO SUPPORT THE SOFTWARE QA EFFORT

The following techniques have been defined to support Software Quality:

- Placing TRIGA Control System Project management tools on the network to facilitate project visibility, accountability and transparency of project execution
- Conducting periodic management reviews, performed as phase reviews, at the end of each development phase to provide management with project status at defined milestones
- Weekly activity reports submitted by each member of the TRIGA Control System development project
- Routine project status briefings in which members of the TRIGA Control System Project provide status regarding their efforts toward TRIGA Control System Project completion

10. MEDIA CONTROL

Media control includes:

- Regularly scheduled backup of media
- Labeling and inventories of stored media in accordance with media storage procedures
- Adequate protection from unauthorized access.

TRIGA Control System Project software shall be located in a Microsoft *Visual SourceSafe* repository in TRIGA Engineering. The TRIGA software CM librarian shall perform routine backups of the repository to DVD media. Software Quality will conduct periodic checks of media control to verify adequacy of the performance of the backups.

11. SUPPLIER CONTROL

For those non-developmental items which are acquired to support the TRIGA Control System development project, supplier control shall follow GA-ESI's existing purchasing/materials

management operating procedures.

12. PROCURED SOFTWARE AND SOFTWARE SERVICES

There is no planned or intended procurement of software or software services for the TRIGA Control System project.

13. OTHERWISE ACQUIRED SOFTWARE

Software that has not been previously approved under a program consistent the ASME NQA-1-2000 standard for use in its intended application shall be evaluated using the criteria defined in Part II, Subpart 2.7 paragraph 302 of the standard.

14. RECORDS COLLECTION, MAINTENANCE AND RETENTION

Software Quality activities are documented by records and reports throughout the software development life cycle. Metrics collected will be periodically reviewed for trends and process improvement opportunities. Software Quality records will be collected and maintained in a designated Document Center network directory for a minimum of 10 years. Guidance regarding records collection and retention is contained in GA-ESI operational procedures for the operation of the GA-ESI Document Center.

15. TRAINING

Training will be provided to the staff assigned to perform software quality functions for the TRIGA Control System Project. Some training will be performed in the classroom; some as on-the-job-training. The training schedule will be compatible with the project schedule.

16. TRIGA CONTROL SYSTEM RISK MANAGEMENT

Risk management shall be conducted to satisfy the requirements of IEEE Std 1012-2004, generally following the content of IEEE Std 16085-2006, as elaborated in the TRIGA Control System Risk Management Plan. Risk analyses shall be performed and presented at development phase reviews.

17. GLOSSARY

See Section 2.4.2 herein.

18. SQAP CHANGE PROCEDURE AND HISTORY

The SQAP is a configuration-controlled document. As such, changes to the SQAP after its initial release shall follow the GA-ESI formal change process.

Revision	Section	Description of Change
B	Global	Updated to address corrective actions identified by INL