US Safety-Related

Toshiba Project Document No.	Rev. No.
FA10-0501-0024	1

NRW-FPGA-Based Safety-Related I&C System Qualification Project
Project Requirement Document

Title: Software Program Plan

Customer Name	None
Project Name	NRW-FPGA-Based I&C System Qualification Project
Item Name	None
Item Number	A10
Job Number	9P04482
Applicable Plant	None

Rev. No.	Issue Date	Description	Approved by	Reviewed by	Prepared by
1	Apr 27, 2012	See DCN-FA10-0501-0024-01	Y. Goto Apr 27, 2012	T. Miyazaki Apr 27, 2012	K.Sato Apr 27, 2012

Initial Issue Date	Issued by	Approved by	Reviewed by	Prepared by	Document filing No.
Nov 2, 2011	Monitoring System Engineering Group Instrumentation & Control Systems Design & Engineering Dept	Y.Goto Nov 2, 2011	T.Miyazaki Nov 2, 2011	K.Sato Nov 2, 2011	RS-5156851

Record of Revisions

Rev No.	Date	History	Approved by	Reviewed by	Prepared by
0	See Cover Page	First Issue	See Cover Page	See Cover Page	See Cover Page
1	See Cover Page	See DCN-FA10-0501-0024-01	See Cover Page	See Cover Page	See Cover Page

Table of Contents

Tables

Figures

1 Introduction

1.1 Purpose

This Software Program Plan (SPP) provides guidance and expectations for the design, development, implementation, safety analysis, review, testing, installation, and configuration management for software that Toshiba supplies for US nuclear plants. This SPP covers the complete software life cycle for the customer[1] units, from Conceptual design through System Retirement, and can be used by the customer as well as Toshiba. This Software Program Plan (SPP) will not be used directly for software design and development. Rather, this SPP is intended to provide the requirements, guidance, and expectations as explained in Section 1.2.

This SPP defines the approach, organization, responsibilities, and methodologies for the complete software life cycle for nuclear safety-related and nonsafety related (Groups 1 and 2) systems. The nonsafety groups are defined in Section 1.10.

1.2 Use of the Software Program Plan

Section 1 of this SPP shall be considered to be incorporated into each of Sections 2 through 13 in its entirety. Appendix C shall also be considered to be incorporated into the applicable portions of Sections 2 through 13 in its entirety.

1.2.1 Vendor Plan Use

This Software Program Plan (SPP) will not be used directly for software design and development. Rather, this SPP is intended to provide the requirements, guidance, and expectations for the following:

- Software-based systems provided by Toshiba and Toshiba's partners in plant engineering, construction, procurement, renovation, modification, or maintenance,
- Software-based systems subcontracted by Toshiba, and
- Long-term maintenance, modification, and replacement activities performed by the customer.

It is not the intent that the SPP would be used directly to guide software design and development. Rather, Toshiba and its subcontractors shall establish the procedures, guidelines, and instructions necessary to deliver their software-based systems and documentation in a consistent, safe, reliable, secure manner, in compliance with this SPP. However, each vendor should provide a single document that references the vendor document or documents that implement each element required by this SPP in their software life cycle.

[1] Specific customer does not exist in this project. This "customer" in this report means that the customer applies Toshiba NRW-FPGA-Based Safety-Related I&C system to the nuclear power plants.

Thus, this SPP is to be used as a basis to generate plans for individual systems or for logical groups of systems by Toshiba, Toshiba's contractors, and vendors who supply configured, customized, or custom developed software-based systems and equipment. This program plan describes and establishes requirements for the following activities in the software development life cycle:

- Project Management (Section 2)
- Development (Section 3)
- Verification and Validation (Section 4)
- Quality Assurance (Section 5)
- Safety (Section 6)
- Configuration Management (Section 7)
- Integration (Section 8)
- Testing (Section 9)
- Training (Section 10)
- Installation (Section 11)
- Operations (Section 12)
- Maintenance (Section 13)

Plans for these activities may be combined as deemed appropriate by the customer, Toshiba, subcontractors, and vendors as long as all aspects of the plans contained in this program plan are provided. This SPP encourages each set of software plans be constructed in a logical manner. One possible organization groups the plans in accordance with the independence requirements and management structure in a typical project (see Figure 1).

The content and scope of each of Toshiba plans are summarized in two tables within the body of this SPP (Table 9 on page 89 and Table 19 on page 141). Toshiba shall, and the customer staff, and subcontractors can use this SPP to establish software plans based on the following data:

- Software life cycle activities are summarized in Table 9 for safety, nonsafety Group 1, and nonsafety Group 2 systems. This table, along with the supporting text provided in the SPP section containing the table, defines the software design aspects of the software life cycle and the process applicability for each classification of software.
- Verification and validation (i.e., peer review and test) activities are summarized in Table 19. This table, along with the supporting text provided in the SPP section containing the table, defines the software verification and validation aspects of the software life cycle and the process applicability for each classification of software.

- Software quality assurance activities are summarized in Section 5. The text in this SPP section defines the methods to be used to verify compliance to plans, procedures, engineering instructions, and other activities associated with the SPP. Software quality assurance activities shall be performed for systems and equipment classified as Safety, Nonsafety Group 1, and Nonsafety Group 2.

At the discretion of the customer, the customer will hold audits and/or assessments to verify compliance with the software plans in use at Toshiba and Toshiba's contractors. These audits and inspections may include technical evaluations of work products, in addition to evaluation of software quality assurance and other software life cycle program effectiveness.

While certain documents are identified as quality records in this SPP, these documents are intended to supplement, and not to replace, those records defined in the project quality assurance plan as quality records.

The software plans associated with designs, including the Software Development Plan (SDP), Software Integration Plan (SIntP), and Software Configuration Management Plan (SCMP), could be grouped together in a single section. The plans associated with verification and validation, including the Software Validation and Verification Plan (SVVP) and Software Installation Plan (SInstP), could be grouped together. The group of plans associated with Project Management, including the Software Project Management Plan (SPMP), Software Safety Plan (SSP), Software Training Plan (STrngP), Software Maintenance Plan (SMaintP), and Software Operations Plan (SOP), could be grouped together for convenience. Alternatively, the SSP could be part of either the SDP or SVVP groups, or distributed between these two groups. The Software Quality Assurance Plan (SQAP) would be performed by an organization separate from Development and Verification and Validation (V&V), to retain the required independence from the remainder of the project organization.

1.2.2 Licensing Basis Documents

This SPP shall be used with the customer's Licensing Basic Documents, when it is required.

1.2.3 [Deleted]

1.3 Scope

This Software Program Plan (SPP) pertains to both safety and nonsafety structures, systems, and components (SSCs) for the customer's nuclear power plant. This SPP applies to those systems that are used to monitor, control, and protect the plant and plant equipment. This SPP applies to the development and procurement of software-based products intended for use in nuclear safety-related systems and equipment for which the requirements of Title 10 of the Code of Federal Regulation Part 50 (10 CFR 50), Appendix B apply. This SPP also includes guidance for systems and equipment classified as nonsafety related Groups 1 and 2, which invokes the quality requirements equivalent to those of International Organization for Standardization (ISO) 9001. The nonsafety Groups are defined in Section 1.10.

The requirements of this SPP can be applied to the hardware, software, and configuration of firewalls and other credited boundaries between the Plant Data Network (PDN) and Information Technology scope, as well as equipment that secures and protects the PDN.

Throughout this Software Program Plan, the term "subcontractor" is interpreted as a software, systems, component, or other equipment vendor who is contracted by Toshiba.

This plan shall not be applied to the plant reference Simulator, used for training operators.

This plan shall not be applied to Information Technology systems, as long as appropriate firewalls, one-way communication, and isolation from the plant control systems to the Information Technology systems are provided. Toshiba considers that the isolation configuration must be controlled by the customer plant controls staff. If firewalls, one-way communication, and isolation do not exist between the Information Technology system and the I&C system, then Toshiba considers that the requirements of this plan needs to be applied to all such Information Technology systems.

1.3.1 Use of Existing Software Plans and Processes

This Software Program Plan (SPP) allows Toshiba or one of their subcontractors to use their existing system software life cycle plans, programs, guidelines, engineering instructions, and other materials used in the software life cycle, instead of creating a new and unproven set of software life cycle plans. However, a summary of conformance to this SPP shall be documented by Toshiba (or their subcontractor) between their plans and this SPP. Each difference shall be explained in the documented summary of conformance. For example, the summary of conformance document could address differences between the regulation used by the existing software life cycle plan and this SPP. Each documented summary of conformance shall be reviewed and approved by the subcontractor (if applicable) and Toshiba, and provided to the customer for review and approval. Each summary of conformance shall be retained as quality records by the customer. This summary of conformance is intended be provided to the customer to reconcile the differences between the SPP and Toshiba or the subcontractor's software life cycle programs.

1.3.2 Nonsafety Plan Requirements

For nonsafety systems, the classification of the nonsafety system, the software technology applied in the system or equipment, or the vendor's software processes, may allow for relaxation of the requirements contained in this Software Program Plan (SPP). The plan contains a set of such relaxations for software classified as Nonsafety Group 1 or Nonsafety Group 2 (see Section 1.10). According to the custermer's request, any additional relaxations shall be defined explicitly in Toshiba or the subcontractor's software plans or in an additional document, and provided to the customer for review, approval, and long-term retention. This strategy meets the requirements of this plan.

No such relaxation is available for safety systems or equipment. The requirements, guidance, and expectations defined in this SPP shall be implemented for safety systems and equipment.

Some aspects of the SPP for nonsafety systems and equipment are not subject to requirements relaxation. The following documents are required. Other documents may be required by the customer on a case-by-case basis. Required documents and/or activities include:

- System Requirements Document (Section 3.8)
- Secure Development and Operational Environment (Appendix C) evaluations are required
- Cyber security except as allowed in the customer cyber security program plan.
- Software or System Validation Testing (Table 13) shall not be eliminated, but System Validation may be combined with factory acceptance testing, referred to as Platform Factory Test (PFT).
- Operations and Maintenance Manual content shall be defined by Toshiba
- Baseline Reviews (e.g., Section 3.3.7) shall be performed on the available documentation, and shall be used as a means of verifying that the assumptions made about reduced documentation are appropriate for the design provided.
- Requirements traceability based on Appendix D for those aspects that are explicitly required in Design Acceptance Criteria for nonsafety systems

For nonsafety systems and equipment, the V&V requirements can be satisfied by peer review and testing. For certain systems and equipment, the V&V requirements may be satisfied by testing alone.

Other documentation requirements may be relaxed, allowing less content, but providing sufficient information to assist in the review, comprehensive testing, long-term maintenance of the system, logical group of systems, or equipment. Examples of such documents where content requirement may be relaxed include:

- The Software Requirements Specification, System Architecture Description, Software Interfaces Document, and Data Communication Protocol and Architecture defined in Section 3.10.3
- The Intra-System Communication Protocol Specification defined in Section 3.11.3

Other documentation may be reduced or eliminated based on technology capabilities. As an example, the Toshiba TOSMAP controller's problem oriented programming language supports direct graphical implementation of logic drawings at a block-to-block level. With the obvious mapping from one logic block to one program block provided in a non-redundant TOSMAP controller, the Software Design Description (defined in Section 3.11.3) may be reduced to a map leading from logic drawing sheets to program modules and/or files. However, when redundant TOSMAP controllers are used, significant complexity is introduced into the application to deal with the redundancy, such as in the Reactor Recirculation Control System or in the Rod Control and Information System. With this complexity, the obvious one-to-one mapping between the logic drawing and the TOSMAP program is mostly eliminated. For redundant TOSMAP controllers, the Software Design Description becomes a requirement, clearly documenting how the logic drawings maps to the software. For several other technologies, such as ladder logic in programmable logic controllers, similar relaxation may be possible, as long as the mapping is clear to an experienced programmer other than the original developer. Even with these

relaxations, mapping between logic drawings and programming modules and files in which the programs are stored shall be documented and provided to the customer.

Subject to similar technology capabilities, this SPP allows relaxation of the following documents, based on the similar technology capabilities:

- Software Design Description (Section 3.11.3.1) and Code Review (Section 3.12.3.3) may be eliminated when logic drawings map to graphical programming at a block-to-block level. When simple mapping is not possible, as in the case of the redundant TOSMAP controller, at least those complex sections of code dealing with redundancy shall be reviewed.

- Formal, documented Unit and Module Testing as well as Integration Testing (Section 3.12.3.4) may be eliminated if code review was not required.

- Pre-definition of the Unit and Module Testing as well as Integration Testing (Section 3.12.3.4) may be eliminated, with testing documentation created while testing is being performed. This testing should be done by an engineer other than the engineer who designed or wrote the code. If testing is performed by the engineer who designed or wrote the code, sufficient testing procedures, acceptance criteria, and test case document shall be created to allow for independent review of the testing, and the reviewer shall be able to require additional testing.

1.3.3 Defining Software

For this Software Program Plan (SPP), the term "software" are defined to include all of the following:

- Software;

- Firmware, as defined in the Institute of Electrical and Electronics Engineers (IEEE) Standard (Std.) 7-4.3.2-2003 (**Reference** 11);

- Logic in Field Programmable Gate Array (FPGA) or Complex Programmable Logic Devices (CPLD) which cannot be 100% tested, as defined by the USNRC in Digital Instrumentation and Controls (DI&C) Interim Staff Guidance (ISG)-04 for FPGAs and CPLDs in Priority Logic Modules (**Reference** 34); and

- Other constructs having complex digital sequencing, such as sequencers built of discrete or integrated logic that cannot be 100% tested.

The primary focus of this SPP is the software provided in Instrumentation and Control systems and equipment, including data historians and other systems that support plant control and the main control room operators. This SPP can be applied to software (see preceding paragraph) in other plant systems and equipment, to include but not be limited to electrical systems (e.g., motor control centers, circuit breakers, variable frequency drives, and adjustable speed drives); heating, ventilation, and air conditioning (HVAC) systems; security systems; and communications networks, switches, routers, firewalls, and other network equipment.

1.4 Roles and Responsibilities

1.4.1 Organization

Toshiba and Toshiba's contractor supplying software-based systems, shall define and maintain a documented organizational structure, including roles, responsibilities, delegation, and authority of personnel involved in the development of software products. A suggested organization for the software life cycle defined in this Software Program Plan (SPP) is shown in Figure 1. Also shown in this figure are the various software plans and the personnel that typically have the responsibility for implementing.

Alternative organizations and methods of assigning the plans to personnel are acceptable, but require evaluation and documentation to justify that the expected and required levels of independence are maintained (see Section 1.4.2).

1.4.2 Independence

1.4.2.1 Safety Related Software

The independence of the Quality Assurance (QA) Manager and the Software Quality Assurance Lead from the Project Manager (PM) and the independence of the Software V&V and Software Safety Leads from the Software Development Lead shall be maintained in any proposed organization.

Each QA Manager shall report directly to the Executive Level, ensuring that each QA Manager has the authority and organizational freedom, including independence from cost and schedule considerations, sufficient to identify quality problems; initiate, recommend, or provide solutions; and verify implementation of solutions. Each Software Quality Assurance Lead ensures the requirements in this plan are met, and must be an independent, dedicated individual. Each Software Quality Assurance Lead reports to the QA Manager.

Each Software Safety Lead and their Software Safety Team shall have sufficient autonomy from the V&V and Development organizations to perform all necessary tasks without interference from or unacceptable control by the software Development or V&V organizations. IEEE Standard 1012-1998 (**Reference** 20) provides guidance for establishing financial, managerial, and technical independence for software V&V from the Development organization. The IEEE Standard 1012-1998 guidance for financial independence provides appropriate freedom with respect to the V&V organization's budget. The IEEE standard's guidance for managerial and project management independence provides appropriate freedom with respect to V&V schedules, as well as overall project cost and schedule considerations.

Audits and reviews shall be performed by individuals or groups other than those who performed the original design or work being audited or reviewed.

1.4.2.2 Nonsafety Software

For nonsafety software, audits, reviews, and V&V tasks shall be performed by individuals or groups other than those who performed the original design or work.

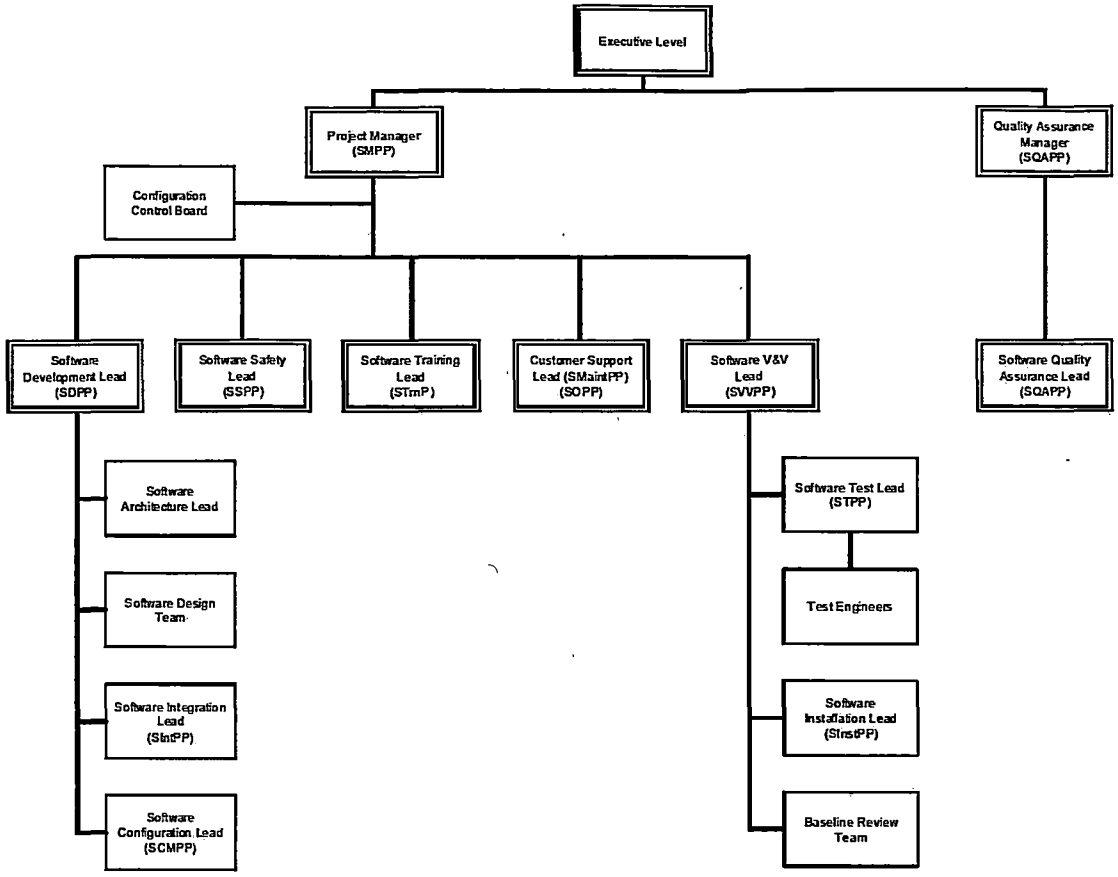

Figure 1. Example Software Development Organization for Safety Related Software

1.4.3 Responsibilities

Project Manager – The Project Manager (PM) shall be responsible for the managerial processes and technical direction of the software development activities as defined by the Software Program Management Program Plan (SPMPP) (Section 2 of this Software Program Plan (SPP)).

The project manager reports directly to their Executive Level, as does the Quality Assurance Manager, assuring the independence of the Quality Assurance (QA) function. Responsibilities and authority within the software development project can be delegated, as long as the PM retains overall authority, responsibility, and authority.

Software Development Lead – Each Software Development Lead shall ensure use of controlled processes throughout the software life cycle for all developed software within the scope of this SPP. This responsibility is divided into Architecture and Integration functions. Each Software Development Lead shall approve software specifications and requirements documents and reviews deliverables.

- Software Architecture Lead – Each Software Architecture Lead is the head of the Software Design Team. Each Software Architecture Lead shall ensure the software requirements are incorporated

into the design in a manner within the framework. This should be accomplished in coordination with their Software Integration Lead.

- Software Integration Lead – Each Software Integration Lead shall coordinate the integration of software modules and software into production hardware to complete the software builds and integration in preparation for testing.

- Software Configuration Lead – Each Software Configuration Lead shall be responsible for configuration management of software and hardware. Each Configuration Librarian reports to their Software Configuration Lead. Each Software Configuration Lead is responsible for releasing software, with initial releases provided by the appropriate Lead and re-releases at the additional authorization of the Configuration Control Board.

- Software Safety Lead – Each Software Safety Lead shall be responsible for carrying out the actions described in the Software Safety Program Plan (SSPP) for safety systems. Each Software Safety Lead shall have sufficient autonomy from the Software Development Lead and the Software Verification and Validation Lead to require compliance with findings resulting from software safety analysis. Software safety analyses should be performed in synchronization with the design activities, with independent verification and validation of the resulting analyses and design.

Software Training Lead – Each Software Training Lead shall ensure that end users are properly trained on use and maintenance, as appropriate, of the software and the system. Each Software Training Lead shall ensure that the personnel performing activities under their software life cycle processes are appropriately trained and qualified.

Software V&V Lead – Each Software Verification and Validation Lead shall be responsible for carrying out the verification and validation activities as defined by their Software Verification and Validation Program Plan (SVVPP). The following organizations are administratively aligned under the Software V&V Lead:

- Software Test Lead – Each Software Test Lead shall be responsible for defining the software and systems test plans, procedures, and cases. Each Software Test Lead shall be responsible for the overseeing the performance of testing and test engineers, working with the Software Development Lead to resolve test anomalies, and setting boundaries and requirements for retest activities.

- Software Installation Lead – Each Software Installation Lead shall be responsible for ensuring that software modules are built and hardware configurations are set, verified, and validated. These actions shall be performed in accordance with the software build instructions and hardware configured in accordance with the Operations and Maintenance Manual associated with their system or logical group of systems.

- Baseline Review Team – Each Baseline Review Team (BRT) is responsible for performing Baseline Reviews to ensure that activities are properly performed and documented at each phase in the software life cycle. Each Baseline Review Team is responsible for verifying all work products are completed, placed under configuration control, and records updated to reflect completion of a life cycle phase, working under the direction of the Quality Assurance Manager. Work products to be subject to baseline review shall be defined in the Software Configuration Management Program Plan. Each Baseline Review Team shall draft a Baseline Review Report at the conclusion of each review. The Software Configuration Lead shall be part of the Baseline Review Team.

Customer Support Lead – Each Customer Support Lead shall be responsible for the System Operations and Maintenance Programs associated with their system or logical group of systems. These items are critical to ensuring their systems are employed according to design and updates and patches are installed in a regimented manner. Unless specifically authorized and evaluated by each software plan, patches shall not be installed on systems classified as safety related and patches should not be installed on systems classified as nonsafety. Each Customer Support Lead shall also provide user feedback into the development cycle in order to improve functionality, safety, security, and human-system interface issues continually.

Quality Assurance Manager – Each Quality Organization shall be responsible for the Software Quality Assurance Program for their systems or logical groups of systems. The QA Manager appoints the Software Quality Assurance Lead, who is responsible for the implementation of development, maintenance, and implementation of QA procedures that ensure that the development of safety-related and nonsafety Group 1 and Group 2 software is performed in accordance with approved plans, procedures, and instructions. Each Software Quality Assurance Lead shall be knowledgeable in QA methodologies standard in the nuclear and software development industries. As necessary, each Software Quality Assurance Lead uses other staff to assist in the performance of QA activities. Each Software Quality Assurance Lead also designates the members of the Baseline Review Team. Each Software Quality Assurance Lead shall provide oversight to ensure that all activities in design, development, review, and test are performed in accordance with the appropriate procedures. The QA Manager is responsible for maintaining the QA program to comply with industry standards.

- Software Quality Assurance Lead – Each QA Manager appoints a Software Quality Assurance Lead to assist in the performance of QA activities for a particular project (for example, review of software life cycle documents or performance of audits or inspections) associated with their system or logical group of systems. Each Software Quality Assurance Lead and the staff working with that lead shall remain independent of the Software Development and V&V teams. Each Software Quality Assurance Lead shall be responsible for implementing the reviews and audits necessary to demonstrate adherence to plans, procedures, and instructions, assigned to them, working under the direction of their Quality Assurance Manager.

1.4.4 Qualifications and Training

All personnel in the software organizations defined in this Software Program Plan (SPP) shall be properly trained to carry out their assigned duties. Training records shall be maintained for each individual involved in the projects. As a minimum, each individual involved in any software development project or aspect shall have training in basic nuclear regulatory requirements, to include safety, compliance procedures, quality assurance, and regulatory guidance, and reporting of Defects and Noncompliances under 10 CFR Part 21, as well as appropriate technical and professional training associated with their position in the organization.

If certification is required, each applicable Lead and each Project Manager shall be responsible for ensuring that only properly certified staff is used.

The QA Manager shall be responsible for establishing requirements for training and indoctrination of all personnel associated with the software life cycle and for ensuring that all software training records are documented and archived. The Project Manager is responsible for verifying that all required training has been administered and, if recurring training is required, training status is current. The Project

Manager's approach to tracking the necessary training shall be documented (e.g., in a task specific Software Project Management Plan) and retained with the project records.

1.4.5 Organizational Interfaces

The Project Manager (PM) shall be the interface with the customer Project Organization. The PM shall control the official external project communications, specifically including those affecting the customer commitments and communication with regulatory agencies, contractual and technical requirements, cost, and schedule.

1.5 Terms and Definitions

The verbs *shall*, *must*, *will*, *should*, *may*, and *can* have specific meanings in this program plan. The meanings of these words are provided below, and duplicated in Appendix A. The definitions of these verbs are taken from the Institute of Electrical and Electronics Engineers (IEEE) "2007 IEEE Standards Style Manual," Section 5, Clause 13.1. These definitions are consistent with normal nuclear usage.

The word *shall* is used to indicate mandatory requirements strictly to be followed in order to conform and from which no deviation is permitted (*shall* equals *is required to*).

The word *must* is deprecated[2] and shall not be used when stating mandatory requirements; *must* is used only to describe unavoidable situations.

The word *will* is deprecated and shall not be used when stating mandatory requirements; *will* is only used in statements of fact.

The word *should* is used to indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain course of action is deprecated but not prohibited (*should* equals *is recommended that*).

The word *may* is used to indicate a course of action permissible within the limits of the standard (*may* equals *is permitted to*).

The word *can* is used for statements of possibility and capability, whether material, physical, or causal (*can* equals *is able to*).

All other verb forms (e.g., "is" and "are") represent informative sentences, devoid of requirements.

All terms and definitions are provided in Appendix A.

1.6 Acronyms

Acronyms used in this plan are provided in Appendix B.

[2] The word "deprecated" indicates a term or definition whose use is discouraged because such use is obsolete, misleading, or ambiguous. [IEEE 610.12-1990]

1.7 Secure Development and Operational Environment

Appendix C provides an overview of the Secure Development and Operational Environment (SDOE) as a systems design context. Appendix C to this Software Program Plan (SPP) is normative. All software plans shall document the methods used to ensure that the documentation from the cyber security as well as the documentation from the SDOE programs shall be treated as required in current USNRC guidance, which may include document classification and handling as Official Use Only (OUO).

1.8 Applicable Standards and References

Each normative and informative reference from the USNRC was considered as input in the development of this plan, and incorporated in the plan appropriately. As a result, the plan considered to meet and implement the intent of the USNRC Standard Review Plan, NUREG-0800, Revision 5, Chapter 7, Branch Technical Position 7-14 (**Reference** 3).

The list of applicable standards and references are listed below:

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

2. USNRC, NUREG-0800, "Standard Review Plan (SRP) for the Review of Safety Analysis Reports for Nuclear Power Plants," Chapter 7, "Instrumentation and Controls," Revision 5

3. USNRC, NUREG-0800, "Standard Review Plan (SRP) for the Review of Safety Analysis Reports for Nuclear Power Plants," Branch Technical Position (BTP) 7-14, Revision 5, "Guidance on Software Reviews for Digital Computer-Based Instrumentation and Control Systems"

4. USNRC, RG 1.168, Revision 1, "Verification, Validation, Reviews, and Audits For Digital Computer Software Used in Safety Systems of Nuclear Power Plants"

5. USNRC, RG 1.169, Revision 0, "Configuration Management Plans for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"

6. USNRC, RG 1.170, Revision 0, "Software Test Documentation for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"

7. USNRC, RG 1.171, Revision 0, "Software Unit Testing for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"

8. USNRC, RG 1.172, Revision 0, "Software Requirements Specifications for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"

9. USNRC, RG 1.173, Revision 0, "Developing Software Life Cycle Processes for Digital Computer Software Used in Safety Systems of Nuclear Power Plants"

10. USNRC, RG 1.152, Revision 3, "Criteria for Use of Computers in Safety Systems of Nuclear Power Plants"

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

12. USNRC, RG 1.153, Revision 1, "Criteria for Safety Systems"

13. IEEE Std. 603-1991, "IEEE Standard Criteria for Safety Systems for Nuclear Power Generating Stations," including correction sheet dated January 30, 1995

14. IEEE Std. 610.12-1990, "IEEE Standard Glossary of Software Engineering Terminology"

15. IEEE Std. 730-2002, "IEEE Standard for Software Quality Assurance Plans"

16. IEEE Std. 828-1990, "Standard for Software Configuration Management Plans"

17. IEEE Std. 829-1983, "Standard for Software Test Documentation"

18. IEEE Std. 830-1993, "IEEE Recommended Practice for Software Requirements Specifications"

19. IEEE Std. 1008-1987, "IEEE Standard for Software Unit Testing"

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

21. IEEE Std. 1016-1998, "IEEE Recommended Practice for Software Design Descriptions"

22. IEEE Std. 1028-1997, "Standard for Software Reviews"

23. IEEE Std. 1042-1987, "Guide to Software Configuration Management"

24. IEEE Std. 1058-1998, "IEEE Standard for Software Project Management Plans"

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

26. IEEE Std. 1219-1998, "IEEE Standard for Software Maintenance"

27. IEEE Std. 12207-1996, "IEEE/Electronic Industries Alliance (EIA) Standard for Software Life Cycle Processes"

28. Not Used

29. Not Used

30. Not Used

31. Electric Power Research Institute (EPRI) Technical Report 1011710, "Handbook for Evaluating Critical Digital Equipment and Systems," November 2005

32. EPRI TR-106439, "Guideline on Evaluation and Acceptance of Commercial Grade Digital Equipment for Nuclear Safety Applications," October 1996

33. USNRC, NUREG/CR-6463, Revision 1, "Review Guidelines for Software Languages for Use in Nuclear Power Plant Safety Systems," August 1997

34. USNRC Digital Instrumentation and Controls Interim Staff Guidance 4 (DI&C ISG-04), "Highly-Integrated Control Rooms – Communications Issues (HICRc)," Revision 1, March 06, 2009

35. Not Used

36. Not Used

37. Not Used

38. EPRI TR-107339, "Evaluating Commercial Digital Equipment for High Integrity Applications," December 1997

39. IEEE Std. 1233-1998 (Reaffirmed 2002), "IEEE Guide for Developing System Requirements Specifications"

40. Not Used

41. USNRC, NUREG/CR-6101, "Software Reliability and Safety in Nuclear Reactor Protection Systems," June 1993

USNRC documents (e.g., NUREG, RG) can be downloaded freely, when available, from the USNRC website at http://www.nrc.gov. The Code of Federal Regulation (CFR) can be downloaded in part or in whole from the USNRC web site, or from other free sites. IEEE documents can be purchased from http://standards.ieee.org. Older EPRI documents can be downloaded freely from the EPRI web site at http://www.epri.com.

Those IEEE standards with associated USNRC Regulatory Guides were incorporated as expressed in the Regulatory Guides. Additional IEEE standards referenced were used at least as guidance and often invoked as mandatory, as defined in the individual software program plans.

This Software Program Plan (SPP) provides one solution to resolving the small overlaps that exist among the IEEE software standards. Other solutions may be implemented in the software plans written by Toshiba and Toshiba's contractors, or to the customer (during and after commercial operation). According to the customer's request, documentation shall be provided to the customer for review and approval of any alternate solutions, and the documentation shall map that solution back to that of this SPP.

1.9 Software Life Cycle Overview

The process model specified by this SPP is consistent with the IEEE software life cycle model found in IEEE Standard 1074-1995, "IEEE Standard for Developing Software Life Cycle Processes" (Reference 25). The verification and validation model specified by this procedure is consistent with IEEE 1012-1998, "IEEE Standard for Software Verification and Validation" (Reference 20). Phases of this development, which can be performed in a sequential or iterative manner, include:

- Planning Phase - Project Planning and Concept Definition occur during this phase, in which initial project planning (e.g., defining the organization, interfaces, responsibilities) is carried out, the basic problem is defined, and possible solutions are explored.

- Requirements Phase - Requirements Definition occurs during this phase, in which system and software requirements that pertain to functionality, performance, design constraints, attributes, human-system interface, and system interfaces are documented. Planning for and verification of System and Acceptance Tests is performed during this phase.

- Design Phase – In the Design Phase, the system and software design is developed, documented, and reviewed based on the system requirements. Development of the detailed design may further refine the software specifications. Planning for and verification of Component or Unit and Integration Tests is performed during this phase. Tests are designed for Component, Integration, System, and Acceptance Testing.

- Implementation Phase – In the Implementation Phase, the system and software design is translated into code and individual software modules or logical groups of modules are tested by the developer. This process is referred to as "unit testing." During this phase, a code review (or code walkthrough) is performed. The code review includes tracing the source code to the design specifications to verify correctness, consistency, completeness, and accuracy. This phase also includes Integration, which is the process of assembling and testing software modules and hardware to create the final system. Test procedures and test cases are generated for the Component, Integration, System, and Acceptance Tests.

- Test Phase - Validation testing, in which the integrated system is exercised by testing, provides verification and validation of software functionality to assure adherence to the requirements, which shall include functionality, performance, and other attributes required for nuclear safety, process control, and human-system interface. In order to achieve an appropriate level of independence, the developer should not define, write, or perform test cases and testing. The developer may be used to verify the test performance results. Criteria for testing should be based on system requirements, with appropriate review. The test procedures should be designed and developed starting in the requirements phase and proceeding in parallel with system development. At the completion of this phase, all Component, Integration, and System Tests have been performed. Acceptance Testing may start during this phase for Factory Acceptance Testing.

- Installation Phase – In the Installation Phase the integrated system is installed, commissioned, and acceptance tested in the final operational environment. Normally, all activities from all previous phases are complete before this phase is initiated. However, this phase may also be performed in phases during a properly planned and phased life cycle. At the completion of this phase, the Acceptance Testing has been performed.

- Operation Phase – During this phase, the software is used and maintained. This phase starts after the units have been accepted by the customer. An error reporting process must be established to ensure that detected problems or errors are recorded, classified as to severity, and corrected as necessary. Any required software changes are performed during the Maintenance Phase.

- Maintenance Phase – During this phase, which may be entered and exited many times for each system, the software maintained. Maintenance includes both repair of design errors and

enhancements. An error reporting process must be established to ensure that detected problems or errors are corrected as necessary and the error closed out. Any required software changes must be done within the framework of the development process.

Each of these phases includes development and safety analysis activities as well as verification and validation activities. The relationship between development, safety analysis, and V&V personnel is defined in this SPP.

This SPP is generally based on the waterfall life cycle model to provide a logical structure and nomenclature for this Software Program Plan. However, this plan does not require rigid adherence to the waterfall life cycle model as this can cause severe difficulties in practice. For example, rigid compliance with the waterfall model requires it to be implemented in discrete design phases in which everything necessary to complete a given phase is finished before the next phase begins. Thus, all requirements must be specified, reviewed against the requirements in the previous phase, and approved before the design can begin. The design must be complete, reviewed, and approved before coding can begin.

Meeting this sequencing requirement is cumbersome or impractical to execute in all software development projects. For example, design decisions made early may be found to be impractical during later phases. Changes would require reopening various completed phases of the design, revising the design documents, repeating the review process, finalizing the documents again, reviewing and approving them, and continuing with the design process.

Because of the possible inefficiencies of the waterfall method, this SPP allows use of other life cycle models (for example, rapid prototyping, incremental builds, the spiral model, etc.) and design processes have been developed which produce the same levels of documentation, careful design, process compliance, peer review (or verification), and careful testing (or validation). Whatever life cycle models or design methods are used, the requirements are:

- Each phase shall be complete in its coverage
- Each phase shall have the connections and dependencies associated with its scope definition based on the waterfall model
- Each phase shall be closed out as soon as practical and not held open until the end of the project with all phases closed in a single roll-up

Rapid prototyping involves building a throwaway prototype for user interaction and comment. Multiple evolutionary prototypes may be required to iterate to an acceptable final solution. Where rapid prototyping is used in the customer software development effort, limited documentation of the throwaway or iterative solutions should be retained to document the information obtained and the bases for the decisions made in those prototypes. However, all the applicable requirements of this SPP shall be applied to any portion of the prototype software that is used in the final product.

Incremental builds may be used as an integration technique. In this method, the product is designed, implemented, documented, verified, integrated, and validated in a series of incremental builds. Where incremental builds are used in the customer software development effort, the associated documentation shall be updated in each build such that the final product fully complies with the applicable requirements of this SPP.

The spiral model defines a process with similar phases to the waterfall. The difference is that each phase only generates a small portion of the final design. The model is best visualized as a spiral, rooted in the fundamental system architectural design. The spiral is built from design, coding, verification, unit testing, integration, and validation activities, spiraling out from the center with increasing detail and system requirements coverage. The spiral methodology designs, assesses risk, verifies, codes, and validates manageable system pieces, integrating them into the required system. Verification and validation activities are performed in parallel with development process activities. Where the spiral model is used in the customer development effort, the associated documentation shall be updated in each phase of the spiral, such that the final product fully complies with the applicable requirements of this SPP.

For other life cycle software models or design methods, the Software Development Team shall document how their use and the data and information obtained complies with the SPP. Any deviations need to be submitted for review and approval to the customer.

The formats, contents, and names of the Toshiba and Toshiba's contractor software life cycle plans can be different. When differences exist, Toshiba and the Toshiba's contractor shall document the differences and shall provide a requirements traceability matrix between their plans and this SPP.

1.10 Software Classification

Toshiba has an internal standard to classify systems as safety-related In addition, the customer may require to classify a specific system as safety-relatd. All other systems are considered nonsafety related.

The current industry standards and USNRC regulations provided in the references (Section 1.8) allow for a graded approach to the activities in the software life cycle. Accordingly, the plan splits nonsafety related into two groups, Group 1 and Group 2. Group 1 systems have less formal life cycle plans than safety systems, but still use various design and review techniques to enhance reliability and availability. Group 2 systems have the least formal life cycle plans, and have the least effect on the customer reliability, availability, and safety.

A list of all the nuclear plant systems containing software and the classification of the software shall be developed and maintained current through plant retirement. The software classification shall separate software functions into Safety and Nonsafety. Within the classification of nonsafety software, there are two quality sub-classifications:

- Nonsafety Group 1 – Nonsafety functions with requirements for enhanced quality assurance
- Nonsafety Group 2 – General Nonsafety functions

The classification of software does not establish additional requirements for that software. Rather, the classification establishes the degree of rigor applied to the assurance that those requirements are met.

For this classification guidance, software shall include firmware, logic embedded in complex programmable devices, including but not limited to field programmable gate arrays (FPGAs) and other complex programmable logic devices (CPLDs), as specified in this SPP.

The initial classification of software functions and the quality classification of Nonsafety Group 1 and Nonsafety Group 2 shall be the responsibility of the plant system designer. The classifications list shall

be generated by the Toshiba and shall be provided to the customer for review and approval, to be retained and maintained as a quality record.

All structures, systems, and components (SSCs) containing software shall be subject to this classification, whether purchased or implemented by Toshiba, or Toshiba's contractors. This SPP provides guidance for the life cycle activities that are required for each of these three software classifications.

If software functions exist within a safety related SSC that embody both safety and nonsafety related functions, all software within that SSC shall be classified as safety-related. This requirement can be relaxed only if adequate, demonstrated, and documented isolation exists that protects the safety-related functions from any adverse effects resulting from faults and failures within the nonsafety functions. The isolation shall be in accordance with current safety to nonsafety isolation guidance found in the USNRC regulation, guidance, Interim Staff Guidance, and IEEE standards. The documented evaluation of that isolation shall be provided to the customer for review and approval. The documented basis for that isolation shall also be incorporated in the system, software, and hardware documentation (as appropriate), and shall be clearly identified as means for ensuring the isolation within that system, to ensure that future maintenance activities do not invalidate the isolation.

If software functions classified as Nonsafety Group 1 exist within an SSC that embodies both Group 1 and Group 2 Nonsafety functions, faults and failures in the Nonsafety Group 2 software functions shall not cause previously unknown faults and failures in the Nonsafety Group 1 software functions running within that SSC. One method of compliance with this requirement is applying the Nonsafety Group 1 life cycle for the Nonsafety Group 2 items. Another acceptable method is to perform a documented analysis of any identified or identifiable faults and failures in the Nonsafety Group 2 functions, including all Nonsafety Group 1 software and data accessible to the Group 2 software. The analyses shall demonstrate that any fault or failure in the lower classification software cannot propagate to the higher safety classification software and cause a fault or failure in the Nonsafety Group 1 function. This documented analysis shall be provided to the customer for review and approval according to the customer's request. The basis and the evaluation shall also be incorporated in the system, software, and hardware documentation as appropriate, and shall be clearly identified to ensure that future maintenance activities do not alter the functional separation and isolation.

For all systems, software tools do not require classification, unless the vendor intends to reduce the level of verification and validation that would otherwise be required for software at that classification level, based on the quality of the software tools.

Toshiba shall ensure that the documented evaluations of each of their vendor's scope of supply correctly evaluates the software classification for each SSC. The documented evaluations shall include review and evaluation of the appropriateness of each vendor's software life cycle (or cycles) plans, programs, procedures, processes, and adherence to the software life cycle for both the application and platform software within each packaged SSC, including both newly developed and previously developed software. The content of each review should follow that provided in EPRI Technical Report 1011710 (**Reference** 31), Chapter 4, Sections and 4.5, 4.6, and 4.7. Each review should be documented and provided to the customer. The customer will maintain all software classification records in a design basis document, in a location that the customer will establish, and which the customer staff will maintain for the life of the plant.

Documentation of each evaluation shall be supplied to the customer for review and approval according to the customer's request. When the customer requires it, the customer can review and approve the Toshiba's documented evaluation of that vendor and the vendor's products prior to contract award by Toshiba.

The following example lists of criteria shall be used to generate the initial software classification. The lists of example criteria are provided below, as alphabetic lists, under each of the classification groups. Classification requires evaluating each item, until a match or condition is found which either forces the software function into a given category or forces the software function out of that classification. Any software function that is not classified as Safety shall be assigned a Nonsafety classification. Any nonsafety software function not classified Nonsafety Group 1 shall be classified as Nonsafety Group 2.

1. Software classified as Safety Related resides in safety SSCs. Examples of criteria that invoke this classification are provided below:

    a. Safety related systems are defined by the United Stated Nuclear Regulatory Commission (USNRC) in 10 CFR 50.2, "Definitions," which states:

        > Safety related structures, systems and components means those structures, systems and components that are relied upon to remain functional during and following design basis events to assure:
        >
        > (1) The integrity of the reactor coolant pressure boundary;
        >
        > (2) The capability to shut down the reactor and maintain it in a safe shutdown condition; or
        >
        > (3) The capability to prevent or mitigate the consequences of accidents which could result in potential offsite exposures comparable to the applicable guideline exposures set forth in § 50.34(a)(1), § 50.67(b)(2), or § 100.11 of this chapter, as applicable."

    b. Safety related systems are those that are classified correctly as a safety system in the current revision of the final safety analysis report (FSAR).

    c. Safety related functions are defined in RG 1.97, "Instrumentation for Light-Water-Cooled Nuclear Power Plants to Assess Plant and Environs Conditions During and Following an Accident."

    d. Safety related functions shall be controlled only by software classified as safety-related, and implemented within safety related systems.

    e. Safety related software is installed in other safety related systems that support the safety functions, including but not limited to: Heating, Ventilating, and Air Conditioning (HVAC); chilled water systems; and both emergency and normal power supplies, including the power sequencing and distribution networks.

    f. Safety classified software does not necessarily include SSC where the safety function is not embodied in the software. Just because software is installed in a safety related SSC does not mean that the software must also be classified as safety related. As an example, the software

in a pressure transmitter that is considered part of the reactor coolant system pressure boundary does not have to be safety related if the function of that software is to provide pressure or flow data to a nonsafety control function, such as the reactor feedwater system for normal feedwater control.

2. Software classified as Nonsafety Group 1 performs critical functions within the plant. Critical functions are defined as functions required to achieve major plant performance requirements, or functions which, on failure, would degrade plant performance significantly or pose a safety hazard to plant personnel or the general public. Examples of Nonsafety Group 1 classification criteria include the following:

   a. Software functions that play a complimentary role to safety functions in the achievement or maintenance of plant safety, especially the functions required to operate after the controlled state has been achieved, to prevent a Postulated Initiating Events (PIE) from leading to unacceptable consequences, or mitigate the consequences of a PIE.

   b. Software functions that improve or complement the execution of safety functions in mitigating the consequences of a PIE, so that plant or equipment damage or release of radioactive material may be avoided or minimized but not be directly involved in mitigating the physical consequences of the accident, or functions necessary for beyond-design-basis accidents

   c. Software that implements reactivity control, to include control or monitor rod position, control or implement reactor recirculation, regulate reactor pressure including control of the main turbine/generator, and control reactor water level.

   d. Software whose faults[3] and failures[4] directly initiates accidents or incidents that result in activation of safety systems[5]

   e. For the new and spent fuel pools, software that protects the integrity of the fuel cladding and maintains reactivity within licensed limits

   f. Software that provides radiological monitoring inside the facility that is not classified as safety related

   g. Software that provides indication of loose parts, reactor internals vibration, or monitoring of the Nuclear Boiler System

[3] From IEEE 610.12-1990, Faults are defined as "(1) A defect in a hardware device or component; for example, a short circuit or broken wire. (2) An incorrect step, process, or data definition in a computer program. Note: This definition is used primarily by the fault tolerance discipline. In common usage, the terms 'error' and 'bug' are used to express this meaning."

[4] From IEEE 610.12-1990, Failures are defined as "The inability of a system or component to perform its required functions within specified performance requirements. Note: The fault tolerance discipline distinguishes between a human action (a mistake), its manifestation (a hardware or software fault), the result of the fault (a failure), and the amount by which the result is incorrect (the error)."

[5] The NRC summarizes items a through d as "those systems that can, through normal operation, system failure, or inadvertent operation, affect the performance of critical safety functions."

h. Software that is in the diversity and defense-in-depth analysis that provides backup to safety systems for displaying data, status, or actions; acquiring data; processing data; or controlling plant equipment. This includes the Anticipated Transient Without SCRAM (ATWS) System

i. Software that provides functions whose correct operation is credited in the Probabilistic Risk Assessment (PRA) as reducing the frequency of incidents or accidents

j. Software where faults and failures directly result in significant loss of efficiency or power generation capacity in accordance with the customer's technical specification.

k. Software where faults and failures directly result in radiological releases

l. Software that provides on-line computation of fuel parameters to ensure compliance with licensed limits

m. Software used to detect, mitigate, or suppress internal hazards, including but not limited to fire and flood

n. Software used in systems that provide critical support to the Nonsafety Group 1 systems, including but not limited to: Heating, Ventilating, and Air Conditioning; chilled water systems; and both normal and off-normal power supplies, including any power sequencing and the distribution networks

o. Application software that processes or displays Safety Parameter Display System (SPDS)

p. Application software that is credited as acquiring, processing, or displaying PAMS variables that are not classified as safety related (Category I) variables

q. Software that formats and communicates with plant data and status through the Emergency Response Data System (ERDS) for communication to the NRC

r. Software used in site physical security systems.

s. Software that supports cyber security for all the safety and nonsafety process and process communication systems, including the Plant Data Network (PDN), plant physical security, and emergency preparedness functions, including offsite communications

3. All remaining software within the scope of this SPP is classified as Nonsafety Group 2. Nonsafety Group 2 denotes functions that play a role in the achievement or maintenance of plant operational goals, including functions that have operational significance, but are not classified as Nonsafety Group 1. Nonsafety Group 2 functions can be part of the total response to PIE, but not be directly involved in mitigating the physical consequences of the accident, or functions necessary for beyond-design-basis accidents. No examples of this group are provided. Equipment attached to the "Z" network is not included in Nonsafety Group 2, excepting communication to the EOF and TSC and other equipment classified above.

1.11 General Policies for All Plans

All software plans shall include the general policies defined in Sections 1.11.1 through 1.11.9.

For each software plan, Appendices A, B, and C to this Software Program Plan are normative.

1.11.1 Use of IEEE Standards

Toshiba uses the IEEE software standards that are endorsed by the USNRC for all safety systems. Toshiba should use the IEEE software standards that the customer requires to use for Nonsafety Group 1 systems. Other IEEE standards may be used, but are not required. Where the applicable IEEE standards contain outlines for software plans, Toshiba and subcontractor's plans should comply with and shall provide the content of the outlines in the applicable IEEE standards. Deviations from the requirements and outlines in these plans shall be identified; and the devisations are submitted, with an explanation justifying the deviation, to the customer according to the customer's request.

1.11.2 Life Cycle Task Iteration Policy

Each software plan shall specify the procedures and criteria that shall be used to determine the extent to which tasks shall be repeated when an input is changed or any task procedure is changed. The criteria shall include, as a minimum, assessments of change, change significance, software integrity level, and effects on budget, schedule, and quality.

1.11.3 Deviation Policy

Each software plan shall define the procedures and criteria used to deviate from that software plan. The information required for deviations shall include task identification, rationale, and effect on software quality. The Design Team Lead, V&V Lead, Software Quality Assurance Manager, and Project Manager shall each be responsible for approving each deviation.

1.11.4 Control Procedures

Each software plan shall define control procedures that apply to that plan.

1.11.5 Standards, Policies, and Conventions

Each software plan shall define the standards, practices, and conventions that govern the performance of tasks under that plan.

Secure Development and Operational Environment and cyber security plan information shall not be combined with software life cycle information, since the cyber security information may contain safeguards or Official Use Only (OUO) data, which would unnecessarily restrict access to the software life cycle documentation and add personnel clearance issues.

1.11.6 Schedule

Each plan shall be scheduled in an integrated project schedule, which may be provided by the customer, and tied to the entire software life cycle.

Each plan shall be completed in sufficient time for review and approval prior to use by the customer, according to the customer's request.

Detailed schedules for each plan shall be created and maintained. The scheduled activities shall reflect the requirements of the SPP and each Section within the SPP.

During system development, requirements for each system specific project schedule covering the software life cycle through site Pre-Operational Testing shall be listed in the Software Project Management Plan (SPMP) (see Section 2) as required, and shall be integrated into the overall project's integrated project schedule.

1.11.7 Use of Designees

For this Software Program Plan (SPP), the position responsible is identified for each aspect of the software life cycle. The person holding the responsible position shall be able to designate staff for defined aspects of their roles. The person holding the responsible position shall always be responsible for the actions and decisions of their designated staff.

Toshiba and Toshiba's contractors document and maintain a list of responsible plan leads, project managers, and management and their designees. This document defines the roles and responsibilities for each identified lead role, with each role's identified designees clearly defined, including the roles of each designee. Toshiba as well as Toshiba's contractors shall treat this documentation as part of the permanent quality record for each system or logical group of systems.

1.11.8 Modifications to PDS and COTS

Any modifications required to Previously Developed Software (PDS) or to Commercial-Off-The-Shelf (COTS) software shall be made using the same software plans used to develop custom software. The modifications shall be made using the processes defined in this SPP as applicable to that classification of system or equipment (see Table 9, Table 19, and Section 5). The modification shall employ Configuration Management, Change Control, peer review, testing, verification, and validation. For software classified as safety, software safety analysis shall be employed to the extent practical.

1.11.9 Modifications to Configuration

Any hardware or software configuration changes shall be made using the same software plans used to develop custom software. The modifications shall be made using the processes defined in this SPP as applicable to that classification of system or equipment (see Table 9, Table 19, and Section 5). The modification shall employ Configuration Management, Change Control, peer review, testing, verification, and validation. For software classified as safety, software safety analysis shall be employed to the extent practical.

1.11.10 Use of Metrics

Metrics,as applicable to Toshiba's (or Toshiba's contractors') technology and processes should be applied as needed to control and measure quality for both the work products and the processes. Metrics shall be defined and should be used only when benefits can be ascribed to their use. Metrics should not be defined without some perceived value in measuring and controlling or improving work products or processes. Metrics shall be created and maintained implementing the software life cycle.

Metrics shall be trended as appropriate to the metric, with adverse trends investigated and resolved as necessary. Metrics that are demonstrated to be of no value should be abandoned, or replaced with metrics that have value to ascertaining process compliance.

Metrics that shall be maintained and trended include the total number of issues identified and the number of remaining open issues for each system or logical group of systems.

The following are examples of metrics that should be considered for each system or logical group of systems:

- Metrics on requirements traceability matrices might include the number of requirements that are unlinked, the number of untestable requirements, and the number of ambiguous words used in requirements.

- Metrics for design activities might include measures of code complexity, measures of software testing code coverage, the magnitude of change based on changes in system requirements, changes identified from verification and validation activities, and other measures of volatility across the software life cycle.

- Metrics for verification and validation activities might include the number of design errors found per lines of code during review as well as during testing.

- Metrics for software safety analysis might include the number of risks and hazards not identified in the appropriate life cycle phase as well as the number of risks and hazards that were not correctly verified and validated during appropriate life cycle phases.

- Metrics for testing might include the total count of as well as the unresolved number of 1) software design errors, 2) test procedure errors, 3) test case errors, 4) acceptance criteria errors.

Metrics should be used to support project management activities, by providing a set of tools to identify areas where processes are ineffective, to identify areas where improvements are required, and to assess the effectiveness of any process improvements. Absolute values of metrics may not be as useful as trends of individual metrics.

1.12 Software Plan Maintenance

This Software Program Plan (SPP) provides a template for the software plans to be written by Toshiba and Toshiba's contractors, for a system or logical group of systems. Toshiba internal organizations can require modification of this SPP to keep their software plans consistent with this SPP. If a Toshiba internal organization find any error in this SPP, the organization shall notify the error for correction.

Toshiba internal organiations and Toshiba's contractors shall review their software plans, procedures, and instructions against the revised SPP. Toshiba internal organiations and Toshiba's contractors shall also review the effectiveness of their plans to determine if changes are necessary, based on metrics associated with their plans. Any changes required in Toshiba internal organiations' software plans shall be written, the plans shall be reviewed and approved internally. The customer shall be notified of the changes for review and approval according to the customer's request.

The personnel responsible for implementing each software plan for each Toshiba internal organization as well as within Toshiba's contractors shall verify that the processes defined in their plan or plans are effective, adequate, suitable, sufficient, and implement the requirements and expectations provided in these software program plans. The responsible personnel shall be responsible for correcting and extending the plan as required to ensure meeting objectives. Each plan shall state the following as applicable:

- How changes to the plan are to be evaluated and approved,
- How changes to the plan are to be made and communicated,
- How staff will be retrained to use the updated plans,
- What backfit to the existing work products will be implemented, and
- How changes to the plan are to be evaluated by the customer.

1.13 [Deleted]

2 Software Project Management Program Plan (SPMPP)

2.1 Introduction

This Software Project Management Program Plan (SPMPP) describes the project management methods and approach used to control schedule, budget, resources, and processes associated with the development, review, test, and software quality assurance of software products. The resultant methods and approach help ensure that the work products meet the specified requirements and are appropriate for use in nuclear power plants.

2.1.1 Purpose

This SPMPP defines the requirements for each Software Project Management Plan (SPMP), which shall be written, reviewed, approved, and implemented for each software-based system or logical group of systems. Implementing each SPMP shall ensure that the work products fulfill the various regulatory requirements referenced in this Software Program Plan (SPP). Specifically, this plan shall:

- Define the managerial processes that ensure all software life cycle activities are properly controlled
- Ensure that the applicable plans define the technical plans, procedures, processes, engineering instructions, and other documented means used to define the software life cycle processes, including software design, development, implementation, safety analysis, review, test, verification, validation, configuration control, and change management
- Discuss the organizational strategy to maintain independence of the design, configuration management and the verification and validation organizations

2.1.2 Scope

This SPMPP applies to software project management plans for the equipment defined in Section 1.1. A SPMP shall be written for each system or logical group of systems defined in Section 1.1. Each SPMP, as well as all other software plans, shall be prepared, reviewed, approved, and retained as a quality record.

A SPMP shall be implemented by Toshiba and Toshiba's contractor supplying software-based equipment. Toshiba and each Toshiba's contractor responsible for a subcontractor shall provide sufficient oversight of their subcontractors to assure implementation of the requirements of this SPP. SQA oversight shall be provided by Toshiba and each Toshiba's contractor responsible for that subcontractor. Additional SQA oversight can be supplied by the customer QA and I&C organizations, with any additional support the customer staff deem necessary.

This SPMPP is based on IEEE Std. 1058-1998 (**Reference** 24).

2.1.3 [Deleted]

2.1.4 Relationship of the SPMPP to Other SPP Sections

In order to simplify the description of processes and ensure clarity in establishing roles and responsibilities and to align with Branch Technical Position 7-14 (**Reference** 3) guidance, the SPP has been split into several sections. Each section of this SPP describes the processes, roles, and responsibilities for the activities in that section. Each section either explicitly or implicitly refers to other sections as necessary to implement additional processes.

This SPMPP defines the methods to control, coordinate, and oversee the activities defined in Section 1, Sections 3 through 11, Section 13, and Appendix C of this SPP. The project management functions defined in Section 12 are implemented in the customer plans, procedures, and instructions necessary to operate and maintain the nuclear power plant.

Thus, for each of the sections, the SPMPP provides the project management oversight necessary to ensure that all life cycle activities specified are used to implement the technical requirements of that life cycle. The SPMPP oversight is defined by the separate sections of this SPP as follows:

- All requirements provided in Section 1 of the SPP apply to the SPMPP, including system and software organization, roles and responsibilities, classification of systems and software, and the life cycle for system and software development.

- The SPMPP controls and oversees the design, development, and implementation performed by the design organizations defined in accordance with the technical and process requirements listed in Section 3, Development.

- Different project management activities within the SPMPP, and possibly different project managers, oversee and control the review, test and other Verification and Validation (V&V) activities performed by the V&V organizations in accordance with the technical and process requirements listed in Section 4, Verification and Validation.

- Similarly, the SPMPP provides oversight and coordination of the software quality assurance activities performed by the quality assurance organization, in accordance with the technical and process requirements listed in Section 5, Software Quality Assurance.

- As with verification and validation, a different set of project management activities controls and coordinates the performance of system and software safety analyses by the system safety group in accordance with the technical and process requirements listed in SPP Section 6, Software Safety.

- The SPMPP activities ensure that change control and configuration management activities are performed in accordance with the technical and process requirements listed in Section 7, Configuration Management. Change control is invoked on each work product preferably after the work product has been released for V&V, and is invoked after timely completion of V&V on each work product.

- The SPMPP activities ensure that the system and software integration activities are controlled and performed in accordance with the technical and process requirements listed in SPP Section 8, Software Integration.

- The SPMPP activities ensure that the system and software testing activities are controlled and performed in accordance with the technical and process requirements listed in Section 9, Testing.

- The SPMPP activities ensure that appropriately trained personnel are used in all phases of the software and system life cycle, and that plant staff are provided with appropriate training and training materials in accordance with the technical and process requirements listed in Section 10, Training.

- The SPMPP activities ensure that software installation at the vendor site and at the nuclear plant are controlled and performed in accordance with the technical and process requirements listed in Section 11, Installation.

- There are no SPMPP activities or requirements associated with plant operations, as these will be controlled by the plans, procedures, and instructions provided by the customer for the normal, expected plant functions that include operation, surveillance, calibration, modifications to tunable constants and other configuration items, and troubleshooting, in accordance with the technical and process requirements listed in Section 12, Operations.

- The SPMPP will be invoked as required to implement the changes required to close anomaly reports and implement enhancements to plant systems in a controlled manner, in accordance with the technical and process requirements listed in Section 13, Maintenance.

- The SPMPP is responsible to ensure that software work is performed in accordance with the technical and process requirements listed in Appendix C, Secure Development and Operational Environment, for all safety systems and for other systems where cyber security requirements apply.

2.2 Project Organization

Each SPMP shall contain the organizational elements provided in this section of the SPMPP.

2.2.1 Process Model

The process model for each project organization shall define:

- The work covered by the SPMP,

- The relationships among the project functions and activities,

- The relationships between the customer organization; Toshiba; Toshiba's contractor, and their subcontractors, and

As a minimum, this shall include relating the timing of major milestones, design baselines, reviews, work products deliverables, and signoffs. Both textual and graphic descriptions may be used.

2.2.2 Organizational Structure

A typical organizational structure is illustrated in Figure 1. Other organizations are acceptable as discussed in and subject to the constraints of Section 1.4. For each SPMP, the organizational structure shall be expanded to include internal and external relationships with Toshiba and Toshiba's contractors.

2.2.3 Organizational Boundaries and Interfaces

Each SPMP shall define the interfaces with higher level plant design, construction, operation, and maintenance schedules (e.g., integrated project schedule during construction or the integrated outage schedule).

As stated in Section 1.4.5, the Project Manager (PM) shall be responsible for external communication control between the customer, customer's contractors, Toshiba, and Toshiba's contractors. For software that is not developed within the PM's immediate software organization, coordination and collaboration will likely exist at lower organizational levels. Each SPMP shall include restrictions on commitments in communications between lower organizations levels. These restrictions shall be implemented to ensure that express or implied commitments are not made at lower organizational levels. All commitments shall be made expressly by project management and the executive levels. Effective controls should include measures such as an autosignature disclaimer at the end of each email message.

2.2.4 Project Responsibilities

The Project Manager (PM) shall be responsible for ensuring independence of the Design, Verification and Validation (V&V), Quality Assurance, and Software Safety Analysis (as applicable) functions as specified in this SPP and their SPMP. Methods and controls to maintain this independence shall be incorporated into the system-specific plan (reporting relationships, budgetary separation, etc.).

2.3 Managerial Process

2.3.1 Management Objectives and Priorities

The Project Manager (PM) shall effectively govern project deliverable creation, ensuring they meet the regulatory expectations for safety and quality as well as the executive level expectations and the customer project requirements for cost and schedule. Each SPMP shall address each of the following critical elements of project delivery:

- Integrity – All aspects of project performance practice, including honesty, transparency, and consistency
- Quality – compliance with the Software Quality Assurance Program Plan (SQAPP), each Software Quality Assurance Plan (SQAP) (Section 5), and applicable industry codes and standards
- Occupational Safety – Compliance with corporate and government regulations, providing a safe work environment, safe working habits, and proper reporting of mishaps

- Project Standards – project deliverables meeting the quality, timeliness, and budgetary requirements as specified by the work packages

2.3.2 Assumptions, Dependencies, and Constraints

Toshiba assumes that management aspects of software development for the customer include the following:

- Software may be developed within Toshiba organization or subcontracted by Toshiba to external software subcontractors.

- Additional documented reviews by Toshiba organizations working under 10 CFR 50 Appendix B program shall be performed when purchasing safety systems that use previously developed software or commercial-off-the-shelf (COTS) software. Such reviews shall include verification that the previously developed software was developed under an acceptable life cycle and that COTS software for the purchased system was evaluated by Toshiba organizations working under Appendix B program. These reviews, and definition of any compensatory actions required to be completed by Toshiba working under Appendix B program for such systems and software, should be performed before contract award.

- Each SPMP shall document additional applicable assumptions, dependencies, and constraints as required.

2.3.3 Risk Management

Risk Management is the identification, analysis, and prioritization of risks followed by deliberate effort to monitor, minimize, and control the probability and/or impact of these risks. Each system-specific project shall document, mitigate, control, and track project risk management, both in terms of the project deliverables, and the schedule and cost. The risk management plan may be included directly in the SPMP, or by reference. Project risk management shall not be limited to risks to nuclear safety, but shall include the risks associated with any project. These risks include lack of realism in schedules or budgets, lack of trained personnel, lack of sufficient detail in the Work Breakdown Structure to proactively detect and recover from task overruns, continuously changing requirements, and lack of clear definitions for phase and project completion.

Section 6 of this document is the Software Safety Program Plan (SSPP) and contains the necessary requirements for software risk management in the area of nuclear safety, which shall be applied as required by the SSPP and by each Software Safety Plan, which shall be created as required for each system or logical group of systems.

Project Management shall minimize project risk by implementing control of changes, both within the individual Project Manager's scope and across Toshiba and Toshiba's contractors, using SPP Section 7.4.2.

Project Management risk management shall follow the applicable the customer as well as Toshiba, and Toshiba's contractors corporate procedures, as applied to all project types.

2.3.4 Monitoring and Controlling Mechanisms and Metrics

Project management occurs throughout the software life cycle. Toshiba and Toshiba's contractor are primarily involved in the development processes, and may be involved in the Maintenance Phase. Subcontractors to the customer will likely be involved in the plant Maintenance Phase.

In addition to the schedule, budget, and resource metrics that each PM shall generate, maintain, and use for process correction and control, the PM shall ensure that appropriate technical software metrics are generated, maintained, and used for appropriate process correction and control, in accordance with the requirements of Sections 1.11.10 and 5.5.

The project phases that the customer expects to see included in each SPMP are as follows:

- Initiation – The project begins with the award of a contract or an internal project is authorized.
- Planning and scheduling – Each project manager is responsible for defining, planning, scheduling, costing, and resourcing the project, with the assistance and data provided by the appropriate management and technical staff who are to be assigned to the project.
- Execution – Processes are performed in accordance with each SPMP.
- Closeout – Finalize the project or task and complete delivery in accordance with the applicable contract

Project monitoring and controls are important throughout the project life cycle. Each SPMP shall describe project control mechanisms applied in each phase. Specifically:

Initiation – Each Project Manager shall produce the preliminary schedule, which shall consider resource availability and allocate these resources according to the approved schedule and budget.

Planning and Scheduling – Each Project Manager shall document the process model, schedule, design inputs and outputs, deliverables, QA requirements and resource allocation in the SPMP.

Project Management tools (and applicable versions) used to accomplish these activities shall be listed within the SPMP. Project management tools include, but are not limited to office automation equipment, including computers and standard office automation software. Standard office software also includes project management software such as the standard Primavera P6 Enterprise Project Portfolio Management software.

Each Project Manager shall verify that all software tools used are listed, including the software version, and accepted for their intended use. This evaluation shall include the possible impact on the safety, design integrity, and quality, if the tool produces incorrect results as well as the ability of the verification and validation activities to detect such errors. These evaluations are especially important for tools written by Toshiba or a Toshiba's contractor, or other such non-commercial tools in limited use. The formality and depth of these evaluations shall be based on the impacts to nuclear safety, reliability, availability, and investment protection of the system where the tool is used.

Each SPMP shall require documentation and use of standard methods to build work packages, request changes, issue documents, purchase materials, and retain records.

Execution – Each SPMP shall define the requirements, content, approval process, use, maintenance, change process, and measurement techniques of at least the following critical project execution elements:

- Work Breakdown Structure (WBS) for each phase of the project, to a level of detail sufficient to detect and control adverse trends in schedule, budget, or resources
- Requests for services and materials
- Kick-off meeting and recurring project status meetings with all interfacing organizations
- Hardware and software development
- Interfaces between software and hardware, and interfaces between systems uses the processes documented in each SPP
- Software V&V and testing
- Software safety analysis, as required
- Implementation of the SQAP
- Assuring that all hardware and software deliverables and life cycle documentation are produced correctly
- Project progress shall be reported with appropriate metrics emphasizing critical path and near critical path activities. Progress shall be measured based on independently verifiable, objective evidence of document and work product status. Progress shall not be based on measurement of elapsed time or resources consumed. Earned value shall be considered equivalent to progress. Progress can be presented based on Cost Performance Index (CPI) and Schedule Performance Index (SPI). According to the customer's request, Toshiba and Toshiba's contractors shall provide CPI and SPI updates to the customer scheduling and controls staff on a schedule set by the customer scheduling and controls staff. CPI and SPI shall not be based on assumptions concerning percent complete based on cost, schedule, or resources consumed.
- Risk assessment and risk management activities
- Milestone dates for critical activities such as software safety analysis, internal V&V, independent V&V, and configuration control
- Resource and manpower levels

The Project Manager (PM) shall monitor and measure progress toward execution using standard tools specified in their SPMP. Each project manager shall schedule regular project reviews with team members and provide updates to the executive level at documented intervals. Progress reporting requirements from subcontractors and Toshiba organizations shall be documented. Project review frequency is at the discretion of each project manager, based on input from the executive level and language from regulatory commitments as appropriate. The PM is responsible for ensuring that appropriate metrics are maintained, in accordance with the requirements in Section 1.11.10 and 5.5.

Closeout – The PM shall ensure that all process requirements are complete, the system has been evaluated and accepted by the customer, deliveries and signoffs are complete, and final disposition of documentation is performed according to their SPMP as defined in the SPMPP (Section 2.4.2 below).

2.3.5 Staffing Plan

Each Project Manager shall ensure that the overall the customer Integrated Project Schedule is updated and maintained with the following data:

- Personnel requirements
- Start times and durations of need based on the Work Breakdown Structure
- Ties to other predecessor and successor activities

Each Project Manager shall ensure that their SPMP and associated software plans define the prerequisite skills and required training for all personnel.

2.4 Technical Process

2.4.1 Methods, Tools, and Techniques

Each SPMP shall specify the methods and techniques to be used in that Toshiba organization and Toshiba contractor's work, and in the work performed by each of their subcontractors.

Each SPMP shall specify technical tools to be used on a project-wide level, either directly or by reference for both project management activities and technical software activities. Requirements for software activities are defined more completely in Sections 3 through 13 and Appendix C of this SPP.

Software Development Tools

Software development tools used in the software life cycle shall be defined, including version. Each The Project Manager (PM) should ensure that a standard set of software development tools are used within systems being supplied by Toshiba and Toshiba's contractors, to the extent practical.

2.4.2 Software Documentation

Each SPMP shall contain, either directly or by reference, the documentation plan for the software project. The status of each software module shall be documented at the end of each life cycle stage.

2.4.3 Secure Development and Operational Environment and Cyber Security

Each SPMP shall include a process to ensure coordination with the project-specified cyber security requirements. Each SPMP shall include a process to ensure compliance with the Software Secure Development and Operational Environment (SDOE) regulatory expectations contained in Appendix C of this report. If the customer make cyber security requirements on Tohsiba, the requirements shall be

applied to all systems. The PM shall be responsible for ensuring that the customer's requirements are implemented.

The PM will be cleared for safeguards or Official Use Only documents as required and have access to information as necessary to support the SDOE and cyber security for their projects. Methods for clearance are outside the scope of this SPP.

2.4.4 Project Support Functions

Each SPMP shall contain, either directly or by reference, plans for software project support. Other sections of this Software Program Plans contain the program plans for each support function.

2.5 Work Packages, Schedule, and Budget

Each SPMP shall define each of the features defined in the following subsections and who is responsible for documenting each feature, and where the responsible party will store each document. Each SPMP shall also define the individual responsible for maintaining and disposing of each item.

2.5.1 Work Packages

Each SPMP shall require the creation of work packages, which provides the specification of the work that must be accomplished to complete a task. The work packages shall be sufficient to define the work to be performed, specific personnel work assignments, and interfaces to other equipment. The content of each work package shall be defined in each SPMP. Toshiba and Toshiba's contractors shall specify and describe a standard work package format and consistent numbering scheme in their SPMP.

2.5.2 Dependencies

The Project Manager shall be responsible for the identification, documentation, and control of dependencies, both internal and external, and ensure that those dependencies are documented appropriately in the Integrated Project Schedule. Each SPMP shall describe, directly or by reference, the processes to be used to control scheduling and prioritization of work products with dependencies on other work products.

2.5.3 Resource Requirements

Each SPMP shall provide estimates of the total resources required to complete the system or logical group of systems within Toshiba and Toshiba's contractors' scope. These resource requirements should be documented in the Integrated Project Schedule. These resources include the number and types of personnel, computer hardware and time, support software, office and laboratory facilities (as required), travel, and maintenance requirements. Each SPMP shall ensure that resource requirements are maintained in the Integrated Project Schedule.

2.5.4 Budget and Resource Allocation

The Project Manager (PM) shall be responsible for project financial and cost management. The PM shall allocate resources to each project organization, including Toshiba and Toshiba's contractor software development organizations. Resource usage (budget, labor, equipment, etc.) shall be tracked and records retained in the manner specified in their SPMP.

Each SPMP shall describe the maintenance of financial independence of the software development, software safety (as applicable), V&V, and quality assurance organizations. Specifically each SPMP shall explain how the Project Manager shall ensure that the software development organization does not reallocate funds from quality assurance, safety, and V&V budgets.

Each SPMP shall define the work breakdown structure so expenditures can be monitored at the task level. The work breakdown structure shall conform to the customer schedule requirements. Control of the budget for development, V&V, and quality assurance activities shall be assigned to managers that have the required level of fiduciary responsibility and independence (see Section 1.4.2). The Project Manager shall generate the work breakdown structure based on scheduled activities.

2.5.5 Schedule

Each SPMP shall specify a schedule of project activities by reference. This schedule shall be expressed in a format that can be summarized and used to update progress within the customer Integrated Project Schedule, in accordance with Section 1.11.6.

3 Software Development Program Plan (SDPP)

3.1 Introduction

The Software Development Program Plan (SDPP) outlines a plan for technical project development of software intended for use in control and monitoring of plant process systems in the customer's plant. This SDPP shall be used as a template for Toshiba and Toshiba's contractors to create Software Development Plans (SDPs) for each system or logical group of systems, and for the customer to support the installed systems.

3.1.1 Purpose

The purpose of the SDPP and each Software Development Plan (SDP) is to:

- Define an approach to the software development process, which increases the probability of detection of human errors and reduces overall risk.
- Define the activities performed in each phase of the development process.
- Describe the methods, tools, and techniques used for portions of the software life cycle processes applicable to this plan, and ensure that the software life cycle processes defined in the remainder of this SPP are compatible with the processes required for this section of the SPP.
- Describe the personnel or groups responsible for development, verification, and validation of various design outputs.

3.1.2 Scope

Each SDP is intended for use in the development of software to be used in nuclear safety related applications for which the requirements of 10 CFR 50, Appendix B apply as well as software that is classified as nonsafety Group 1 or Group 2. Each SDP shall be prepared, reviewed, approved, and retained as a quality record. The SDPP is written to follow the guidance of RG 1.173 (**Reference** 9) for safety systems and Institute of Electrical and Electronics Engineers (IEEE) Std. 1074 (**Reference** 25) for both safety and nonsafety systems.

This SDPP shall be implemented by Toshiba and Toshiba's contractors who supply software-based equipment. Additional oversight shall be supplied for subcontractors by Toshiba or a Toshiba's contractor responsible for that subcontractor. Additional SQA oversight can be supplied by the customer QA and I&C organizations, with additional support as deemed necessary by the customer.

3.1.3 [Deleted]

3.1.4 Relationship of the SDPP to Other SPP Sections

In order to simplify the description of processes and ensure clarity in establishing roles and responsibilities and to align with Branch Technical Position 7-14 (**Reference** 3) guidance, the SPP has

been split into several sections. Each section of this SPP describes the processes, roles, and responsibilities for the activities in that section. Each section either explicitly or implicitly refers to other sections as necessary to implement additional processes.

Thus, this SDPP defines the methods to be used to perform software design, development, and implementation activities, using other portions of the activities defined in Section 2, Sections 4 through 11, Section 13, and Appendix C of this SPP. The project management functions defined in Section 12 are implemented in the customer plans, procedures, and instructions necessary to operate and maintain the nuclear power plants.

Thus, for each of the sections, the SDPP provides the technical design activities necessary to implement the life cycle activities specified by the life cycle. The SDPP interfaces with the other sections of this SPP as follows:

- All requirements provided in Section 1 of the SPP apply to the SDPP, including system and software organization, roles and responsibilities, classification of systems and software, and the life cycle for system and software development.

- The SPMPP, Section 2, controls and oversees the design, development, and implementation performed by the design organizations defined in accordance with the technical and process requirements listed in this SDPP.

- The SDPP processes work cooperatively with the verification and validation organization to perform the necessary review, test, and other V&V activities performed by the V&V organizations in accordance with the technical and process requirements listed in Section 4, Verification and Validation.

- The SDPP works cooperatively with the software quality assurance organization, which ensures that the SDPP and V&V organizations work in accordance with their plans, procedures, and engineering instructions. The implementation of the SDPP is overseen by the software quality assurance activities performed by the quality assurance organization, in accordance with the technical and process requirements listed in Section 5, Software Quality Assurance.

- For safety related systems, the SDPP works cooperatively with the Software/System Safety organization to ensure that the design, development, implementation, review, test, and other V&V activities are performed in a manner that ensures that safety is maximized and that identified safety concerns are implemented correctly as well as reviewed and tested completely. The Software/System Safety work is performed in accordance with the plans, procedures, and engineering instructions in accordance with the technical and process requirements listed in SPP Section 6, Software Safety.

- Performance of all SDPP activities must be coordinated with the change control and configuration management activities. The change control and configuration management activities are performed in accordance with the technical and process requirements listed in Section 7, Configuration Management. Change control is invoked on each work product preferably after the work product has been released for V&V, and no later than after timely completion of V&V on each work product.

- The SDPP activities ensure that the system and software integration activities are controlled and performed in accordance with the technical and process requirements listed in SPP Section 8, Software Integration.

- The SDPP activities coordinate or perform (depending on the safety classification) system and software testing activities in accordance with the technical and process requirements listed in Section 9, Testing.

- The SDPP activities use appropriately trained personnel in all phases of the software and system life cycle, and coordinate with this activity to ensure that training materials and other manuals are created to ensure that plant staff can be trained, in accordance with the technical and process requirements listed in Section 10, Training.

- The SDPP activities ensure that software installation at the vendor site and at the nuclear plants have appropriate documentation and procedures such that the installations can be controlled and performed in accordance with the technical and process requirements listed in Section 11, Installation.

- There are no SDPP activities or requirements associated with plant operations, as these will be controlled by the plans, procedures, and instructions provided by the customer for the normal, expected plant functions that include operation, surveillance, calibration, modifications to tunable constants and other configuration items, and troubleshooting in accordance with the technical and process requirements listed in Section 12, Operations.

- The SDPP will be invoked as required to implement the changes required to close anomaly reports and implement enhancements to plant systems in a controlled manner, in accordance with the technical and process requirements listed in Section 13, Maintenance.

- The SDPP performs software work in accordance with the technical and process requirements listed in Appendix C, Secure Development and Operational Environment, to ensure that work performed does not compromise safety, for all safety systems and for other systems where cyber security requirements apply.

3.2 Organization of Software Life Cycle Processes

This SDPP utilizes a modified waterfall model for the software development life cycle process. The life cycle model is illustrated in Figure 2 (on page 92). As indicated in the introduction, each SDP written for a system or logical group of systems may define a different software life cycle model, as long as the work products, reviews, and testing produced address the requirements of this life cycle model defined in the SDPP. Similarly, the documentation produced from the life cycle model may be grouped differently, as long as each SDP provides a table showing the mapping between the requirements in this SDPP and documentation to be provided, as defined in each SDP.

3.3 Methods

Each SDP shall list the plans, procedures, processes, engineering instructions, and other software life cycle documents required to implement requirements of the SDP. The SDP shall describe how objective evidence of their application is maintained.

3.3.1 Schedule

All activities within this plan shall be planned and scheduled, in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

3.3.2 Configuration Management and Change Control

The design outputs from each phase of software development life cycle shall be controlled as Configuration Items (CI). All CI shall be controlled in accordance with each Software Configuration Management Plan (SCMP).

Any discrepancies or deficient conditions found in a CI shall be resolved in accordance with the Corrective Action process in Section 5.8 and the Change Control process detailed in the SCMPP (see Section 7).

3.3.3 Independent Verification and Validation

Independent verification and validation shall be performed on software design outputs from each phase of the development process, as required in Section 1.4.2 of this SPP.

For safety related software, Independent Verification and Validation (IV&V) shall be performed by an independent organization (see Section 1.4.2 for the definition of independence). The verification and validation process shall be performed in accordance with the Software Verification and Validation Program Plan (SVVPP) (see Section 4).

For software classified as nonsafety Group 1, verification and validation should be performed by an independent organization. If the verification and validation of nonsafety Group 1 software is not to be performed by an independent organization, it shall at least be performed by personnel in the design team who did not participate in or set major aspects of the design

For software classified as nonsafety Group 2, verification and validation shall be performed by personnel in the design team who did not participate in or set major aspects of the design.

3.3.4 Testing

Testing is intended to ensure the software product is free from faults in its specification, design, and implementation (see Sections 4 and 9). The organizations responsible for testing are listed in Table 1 through Table 9.

3.3.5 Software Safety Analysis

SSA activities shall be performed in a timely manner, in parallel with the design and development work to which the SSA analyses apply. A Software Safety Analysis (SSA) shall be performed on each software work product during the same life cycle phase in which the software work product is created, as defined in the Software Safety Program Plan (SSPP, see Section 6). The SSA shall be performed by an independent organization[6] working with the software development group. Preferably, the SSA team shall work with the Software Design team during design activities. The SSA work products shall be reviewed by the software Verification and Validation (V&V) group in accordance with the Software Safety Program Plan (Section 6). Software safety analysis shall be performed in a timely manner, in each phase of the software life cycle.

[6]Requirements for independence of the software safety analysts are stated in NUREG/CR-6101 (Reference 41).

3.3.6 Secure Development and Operational Environment Analysis

A Secure Development and Operational Environment (SDOE) Analysis is performed for each system or logical group of systems to ensure the security of digital assets. Each SDOE Analysis shall be performed, documented, reviewed, approved, and maintained in accordance with Appendix C and coordinated with the customer cyber security program plan. Guidance for the regulatory expectations for SDOE security programs in design activities, or where support for operations and maintenance shall be considered in design activities, is provided in informative Appendix C.

Each Software Development Lead shall ensure that that team members are cleared, as necessary, for access to cyber security information, which may require clearance for safeguards or Official Use Only documents and information,

3.3.7 Baseline Review

A Baseline Review (BR) shall be performed at the end of each software life cycle phase. The purpose of the baseline review is to ensure the following:

- The design activities and design outputs meet the requirements specified in the applicable software project management plans and software requirements specification.
- The SSA and Verification and Validation have been performed in accordance with the requirements of the SSPP (Section 6) and SVVPP (Section 4), respectively.

The requirements for the BR are provided in the SVVPP. The BR shall be documented in a Baseline Review Report (BRR). Any discrepancies in the BRR shall be tracked as open items. Once open items are resolved, the results shall be captured in a revision to the BRR. Each BRR for each system or logical group of systems shall be retained in the permanent software records.

Baseline Review and Baseline Review Reports are not required for Software Program Plans and other programmatic documents, including but not limited to plans, procedures, guidelines, and engineering instructions.

3.3.8 Incomplete Requirements

When the design document or portion of the design cannot be completed prior to release for verification and validation, such as a To Be Determined (TBD) requirement or an Assumption, conditional release of a document is permissible. The portions of the documents that cannot be verified shall be labeled with the words "Conditional Release" and controlled. Documents that have incomplete requirements shall be tracked, and, at the earliest opportunity, the document shall be completed, reviewed, and approved. Incomplete requirements shall be tracked in accordance with the Software Quality Assurance Program Plan, Section 5.6.1.1. All Conditional Releases shall be resolved prior to Platform Factory Test starts (see Section 9).

3.3.9 Use of Previously Developed or Purchased Software

Use of previously developed software (PDS) or commercial-off-the-shelf (COTS) software shall be evaluated prior to its incorporation into any deliverable from Toshiba and Toshiba's contractors. The method for the evaluations is provided in the Software Development Program Plan in Sections 3.11.3.5 and 3.11.3.6 as well as in the Software Verification and Validation Program Plan in Section 4.3.11.

Any PDS or COTS used in safety related systems shall be commercial grade dedicated in accordance with the requirements provided in SVVPP Section 4.3.11. The additional documentation required for commercial grade dedication of the digital content shall augment the documentation required for commercial grade dedication of the equipment that includes the digital content.

The Software Development Lead, the Software Verification and Validation Lead, and the Software Safety Lead, as appropriate, shall ensure completion, documentation, and maintenance of these evaluations.

Modifications to PDS or COTS shall be performed in accordance with Toshiba's or Toshiba's contractor's software plans, as defined by this SPP, based on the safety classification of the system or logical group of systems where the software will be installed. Modifications to PDS or COTS software tools shall be performed in accordance with Toshiba or Toshiba's contractor's software plans, with the appropriate classification.

3.4 Tools

Tools required for project execution, which include compilers, emulators, simulators, and hardware, shall be evaluated and documented as specified in Section 3.11.3.4.

3.5 Requirements Traceability Matrix

Requirements Traceability shall be performed for both safety related and nonsafety software design outputs. Requirements Traceability Matrices (RTMs) shall be updated at the end of each life cycle activity group. The RTM shall provide traceability, verification, and validation of requirements. The RTM will need to be prepared in accordance with a customer's program plan for I&C design requirements management.

3.6 Life Cycle Figures

Several figures illustrate the software life cycle (see Figure 2 on page 92) and the processes used in each life cycle phase (see Figure 3 on page 93 through Figure 9 on page 99) in the customer Software Program Plan. Descriptions of each life cycle phase as well as the phase inputs and outputs are provided in the following sections. Figure 9 provides the Legend for Figure 3 through Figure 9.

Each SDP shall provide similar graphics and sufficient text to define their software life cycle.

3.7 Life Cycle Phases

The development life cycle consists of eight distinct phases or activity groups as follows:

- Planning Phase
- Requirements Phase
- Design Phase
- Software Implementation

- Testing and Integration Phase
- Operations Phase
- Maintenance Phase
- Retirement Phase

RG 1.152 (**Reference** 10), RG 1.173 (**Reference** 9), and IEEE Std. 1074 (**Reference** 25) provide the basis for the software development phases provided in this SDPP. The Software Program Plan (SPP) is based on the Software Life Cycle Process Overview provided in Section 1.9 of this SPP.

Each of these phases is covered separately in the subsections below. For each phase, information is presented providing background and bases for the requirements, followed by the requirements themselves. Table 1 summarizes the design inputs. The outputs from each life cycle phase are summarized in Table 2 through Table 8. When applying Table 2 through Table 8, the requirements established in Table 9 for independence shall be applied. If no entry exists in Table 19, then that process does not have to be applied.

Each SDP shall define their software life cycle phases, and shall provide a matrix, as required in Appendix D, documenting how the SDP life cycle phase relates to the life cycle provided in this SDPP.

3.8 Plant-Level SPP Design Inputs

Design inputs for the Software Program Plans shall be created outside the scope of this Software Program Plan. The documents in Table 1 shall be provided as design inputs for the software systems. The responsible design organizations for the various design outputs are provided. Verification and validation of these phase outputs is to be performed in accordance with the SVVPP, Software Safety Program Plan, (SSPP), and CSPP.

The plant system-specific documents defined in Table 1 shall be translated and extended into a System Requirements Document (SyRD)[7] for each system or logical group of systems, based on the inputs provided in Table 1. The additional requirements necessary for software-based systems shall be added to the plant system requirements. This SyRD document shall be constructed to provide clear, concise, correct communication between the plant systems engineers defining the plant system behavior and the computer system engineers defining the control system behavior. Creation of this document shall ensure that both the plant systems and the computing systems engineers understand the other group's requirements.

The complete set of requirements in the SyRD shall include, but not be limited to, requirements for:

- Complete references to those plant documents that define the plant-level system requirements, as defined in Table 1.
- Concept of operations for the system, including the system interfaces with the rest of the plant, through all plant modes of operation (e.g., refueling, cold shutdown, hot standby, startup, shutdown, power operation, power maneuvering, etc.)

[7]One example of a SyRD is the Toshiba Equipment Design Specification (EDS). Since vendors use different names for this document, a generic name is provided in this SPP.

- Plant and system level architecture
- Hardware inputs, including input impedances, analog and digital filtering, accuracy, resolution, and drift
- Hardware outputs, including current and voltage requirements to include minimum, nominal, and maximum levels; filtering; accuracy; resolution; and drift
- Communication links between this and other systems in the Distributed Control and Information System, to include data transmission protocols, data transmission methods, maximum propagation delays, error detection and correction methods, message definitions, and protocols
- Automated controls capabilities, including timing requirements for data input and output, computing cycle time, use of data gathered by other systems or use of data generated by this system in other systems and timing requirements for data transfer, and tuning parameters
- Human-system interfaces, including requirements for the main control room watch-standing reactor operators for monitoring and control; and interfaces for maintenance staff for surveillance, calibration, and modifications
- Response of the system to faults and failures in hardware inputs, hardware outputs, communication links, automated controls capabilities, and human-system interface software and hardware errors

IEEE Std. 1233 (**Reference** 39), should be considered as guidance for creation of System Requirements Specifications.

The software life cycle may start before the SyRD is reviewed and approved.

The reference documents used to define the SyRD, as defined in Table 1, shall be included in all references to the SyRD.

Table 1. Plant-Level SPP Design Inputs

Plant Planning Phase Output Documents	Included in SyRD
Software Life Cycle Program Plans	No
Procurement Specification (as applicable)	Yes
System Design Documents (SDDs)	Yes
Piping and Instrumentation Diagrams (P&IDs)	Yes
Logic Drawings	Yes
Interlock Block Diagrams (IBDs)	Yes
Electrical One Line Diagrams	Yes
Instrument Equipment Diagrams (IEDs)	Yes
Modulating Control Logic Diagrams (MCLDs)	Yes
Human Factors Program Plans	No
System Requirements Documents (SyRD)	(This Document)

3.9 Planning Phase

3.9.1 Overview

The Planning Phase covers the following:

- Define the system or equipment scope and identify high level product functionality requirements,
- Determine the nuclear plant classification of the system as safety, nonsafety Group 1, or nonsafety Group 2,
- Develop, review, and approve software process management plans compliant to this SPP and Software Program Management Program Plan (SPMPP), and
- Ensure the team is trained to the Software Quality Assurance Program and software development requirements.

The Planning Phase inputs, outputs, and activities are illustrated in Figure 3.

Activities performed during the Planning Phase include:

- Development of the programmatic software management plans, including the Software Project Management Plan, Software Quality Assurance Plan, and Software Development Plan,
- Development of the high-level requirements for the system using such input as the system design document, Piping and Instrumentation Diagram (P&ID), electrical one line diagrams, and logic diagrams, as defined in Table 1,
- Development of the Human Factors Engineering (HFE) Plans, and
- Development of programmatic Platform Factory Test (PFT) Plan and Platforms Integration Test (PIT) Plan, with input from the Secure Development and Operational Environment (SDOE) and from the cyber security plan.

3.9.2 Planning Phase Inputs

The final safety analysis report (FSAR) in the operating stage of the plant life shall be the input into the Planning Phase. Regulation and regulatory guidance shall be input to the planning phase throughout the plant life cycle.

3.9.3 Planning Phase Outputs

The required design outputs from this phase are listed in Table 2. The responsible design organizations for the various design outputs are provided. Verification and validation of these phase outputs shall be performed in accordance with the SVVPP, Software Safety Program Plan, (SSPP), and cyber security program plan. The vendor is expected to use their standard vendor procedures for software generation, including any changes required to their standard procedures for compliance with this SPP and the vendor plans output from this phase.

Table 2. Planning Phase Outputs

Planning Phase Output Documents	Responsible Organization
Plans for the Software Design Process (e.g., Software Quality Assurance Plan, Software Project Management Plan, etc.)	Toshiba and Toshiba's Contractors
Platform Factory Test (PFT) Plan	Software Test Team
Platforms Integration Test (PIT) Plan	Integration and Integration Test Team
Software Safety Analysis Report	Software Safety Team (Perform) Software V&V Team (Review)
Secure Development and Operational Environment Analysis Report	Toshiba and Toshiba's Contractors
Planning Phase Requirements Traceability Matrix	Software Development Team (Perform) Software V&V Team (Review)
Configuration Management Assessment	Configuration Management Lead
Planning Phase Baseline Review Report for Documents that are not Programmatic	Baseline Review Team for safety and nonsafety Group 1 systems Software Development Lead for nonsafety Group 2 systems

3.9.3.1 Software Safety Analysis Report

Software safety analysis is only required for safety related software. The Software Safety Team shall perform and the Software V&V Team shall independently review the Software Safety Analysis (SSA) on the design outputs of the planning phase, as shown in Figure 3. The SSA shall be performed and documented in a Software Safety Analysis Report as specified in the Software Safety Program Plan (SSPP) (see Section 6).

3.9.3.2 Secure Development and Operational Environment Analysis Report

A Secure Development and Operational Environment Analysis shall be performed on design output documents of the Planning Phase, as shown in Figure 3. The SDOE Analysis shall be performed in accordance with Appendix C, and shall be coordinated with the cyber security program plan, and documented in the Secure Development and Operational Environment Analysis Phase Report.

3.9.3.3 Configuration Management Assessment

This task is conducted during or immediately before baseline review to ensure:

- Planning phase activities are completed on the required outputs
- Adequacy of quality assurance as defined in the SQAP
- Appropriate configuration controls (according to the SCMP) are in place to monitor design activities including document revision and track changes control.

3.9.3.4 Baseline Review Report

A baseline review shall be performed by the Baseline Review Team (BRT) at the end of the Planning Phase. The review shall be performed and documented in a BRR as specified in the SVVPP (see Section 4.2.6.6). Baseline Review and Baseline Review Reports are not required for Software Program Plans and other programmatic documents, including but not limited to plans, procedures, guidelines, and engineering instructions.

3.10 Requirements Phase

3.10.1 Overview

The Requirements Phase shall translate the high level functional design requirements into verifiable, traceable technical requirements that define the software required to operate the system. These requirements will be used directly in the design, verification, and validation of the software and system or equipment.

The Requirements Phase inputs, outputs, and activities are illustrated in Figure 4.

Activities performed during the Planning Phase include:

- Development of a Software Requirements Specification (SRS)
- Development of a Hardware Requirements Specification
- Development of Data Communication Protocol and Architecture to define the external system communication, which shall be developed jointly by the organizations responsible for the systems at each end of the communications link, and coordinated with the Cyber Security organization, as defined in the cyber security program plan.

3.10.2 Requirements Phase Inputs

The Requirements Phase input documents shall be the planning phase output documents listed in Table 2.

3.10.3 Requirements Phase Outputs

The required Requirements Phase output documents are listed in Table 3. The V&V shall be performed in accordance with the SVVPP, SSPP, and cyber security program plan.

Table 3. Requirements Phase Outputs

Output Document	Responsible Organization
Software Requirements Specification (SRS)	Software Development Team
Hardware Requirements Specification	Toshiba and Toshiba's Contractors
System Architecture Description (SAD)	Software Development Team
Software Interfaces Document (SID)	Software Development Team
Data Communication Protocol and Architecture	Software Development Team
Software Safety Analysis Report	Software Safety Team (Perform) Software V&V Team (Review)
Secure Development and Operational Environment Analysis Report	Toshiba and Toshiba's Contractors
Requirements Phase – RTM	Software Development Team (Perform) Software V&V Team (Review)
Configuration Management Assessment	Configuration Management Lead
Requirements Phase Baseline Review Report	Baseline Review Team for safety and nonsafety Group 1 systems Software Development Lead for nonsafety Group 2 systems

3.10.3.1 Software Requirements Specification

The Software Requirements Specification (SRS) documents the functional, safety, and performance requirements of the software product. The requirements in the SRS shall be traceable to the design documents and plans generated in the planning phase.

Except as discussed below, the SRS shall meet the requirements of RG 1.172 (**Reference** 8), which endorses IEEE Std. 830 (**Reference** 18). IEEE Std. 830 provides guidance for developing software requirements that are unambiguous, complete, consistent, traceable, and verifiable. IEEE Std. 830 provides a format that shall be used in preparation of all SRS, unless sound, documented reasons exist that another format is to be used.

One sound reason for not using this format is that extensive revision would be required to reformat an existing document. If an existing format is to be used, a documented traceability matrix shall be generated, showing how the existing document maps to the required format, and how the existing document contains all the required materials. If the existing document does not provide the required content, then additional content shall be generated and added to the existing document. The requirements traceability matrix shall be provided to the customer with system delivery, and shall be made available for the customer review.

The SRS shall include the following requirements, as applicable:

- Interface Requirements

- User interfaces, including alarms, displays, soft controls, hard controls, and other equivalent devices, as defined by the Human Factors Engineering Program
- Hardware interfaces
- Software interfaces
- Communication interfaces
- Interfaces between safety and nonsafety systems and equipment, between systems

- Functional Requirements
  - Inputs, both directly input (hardwired to the system) and read from other systems (communicated to this system)
  - Processing methods or formulas for input to output conversion
  - Variables monitored and variables controlled
  - Responses to abnormal conditions (e.g., overflow, communication faults and failures) which shall define error handling and recovery
  - Modes of operation
  - Sequence operations, including requirements for parallel operations
  - Initialization and termination of functions and the system status at termination
  - Prohibited functions and states are clearly defined
  - Outputs
  - Data retention or display requirements

- Design requirements and constraints
  - Input and output attributes, which should including ranges, units, accuracy, bounding ranges, data size, data type, and sampling intervals.
  - Hardware limitations (e.g., memory limitation, processing power, numerical precision)
  - Regulatory requirements
  - Security attributes used to protect the software from accidental or malicious access, use, modification, destruction, or disclosure.
  - Self diagnostics requirements
  - Reliability and availability requirements
  - Maintainability requirements

  - Requirements for testing, surveillance, and calibration incorporated
  - Portability
  - Safety requirements
  - Criticality of certain processes
  - Failure modes for each mode of operation
- Performance requirements
  - System response time
  - Size and type of information to be handled
  - Timing constraints in all modes of operations including anticipated failure conditions.
- Software quality assurance and testing criteria requirements
  - Acceptance criteria
  - Compatibility requirements

3.10.3.2 Hardware Requirements Specification

A Hardware Requirements Specification shall be developed to support the system or equipment development process. The Hardware Requirements Specification shall define at least the requirements for the platform used for running the software, external system interfaces, human-system interfaces, and the hardware-software interfaces. Hardware Requirements Specifications are not defined in this Software Program Plan.

3.10.3.3 System Architecture Description (SAD)

A plant level System Architecture Description (SAD) has been prepared in the FSAR. This SAD provides a high-level system design and describes the functions of the integrated Distributed Control and Information System (DCIS).

For all systems, Toshiba organizations' or Toshiba's contractor's Software Architecture Lead shall generate a SAD for the system or logical group of systems applicable to their scope of supply and document the functions within their scope of supply. Where the interface crosses Toshiba organizations' or Toshiba's contractors' boundaries, all involved parties shall cooperatively generate a single SAD document, with clear identification of each involved Toshiba organizations' and Toshiba's contractors' scope and responsibilities. System interfaces should include the interface between the sensors, transmitters, and actuators and the system hardware. The SAD shall describe the system interfaces without regard to whether the other systems are software-based.

For systems and equipment classified as safety related, the Software Safety Lead from each involved Toshiba organizations and Toshiba's contractors having responsibility for safety systems shall review the SAD and shall provide any additional text or diagrams necessary for the Software Safety Analysis of the system, in accordance with SPP Section 6.4.2.2.

Each SAD shall be provided to the customer Software Safety Lead for review and approval. The customer Software Safety Lead shall be responsible for ensuring that all functions described in the plant-level FSAR describing the DCIS architecture are covered appropriately in Toshiba and Toshiba's contractors analysis.

3.10.3.4 Software Interfaces Document (SID)

For all systems, Toshiba organizations' and Toshiba's contractors' Software Development Lead shall generate a Software Interfaces Document (SID) for the system or logical group of systems applicable to their scope of supply and document the functions within their scope of supply. Toshiba organizations' and Toshiba's contractors' Software Safety Lead shall review the SID and shall provide any additional text or diagrams necessary for the Software Safety Analysis of the system, in accordance with SPP Section 6.4.

Where the interface crosses Toshiba organizations or Toshiba's contractors' boundaries, all parties involved shall cooperatively generate a single SID document, with clear identification of Toshiba organizations' and Toshiba's contractors' scope and responsibilities. System interfaces should include the interface between the sensors, transmitters, and actuators and the system hardware. The SID shall describe the system interfaces without regard to whether the other systems are software-based.

Each SID shall be provided to the customer Software Safety Lead for review and approval. The customer Software Safety Lead shall be responsible for ensuring that the interfaces provided support all functions described in the plant-level FSAR describing the DCIS architecture, and that the interfaces support operation of the plant, including propagation delays between systems and timeliness of data presented to the control room operators.

3.10.3.5 Data Communications Architecture and Protocol Specification

A Data Communications Architecture and Protocol Specification shall be prepared to define the external system communication interfaces to the software product. The architecture shall identify and define each external communication interface, message structure, format, and sequence. Each external communication interface shall comply with the guidance provided in United States Nuclear Regulatory Commission Digital Instrumentation and Controls Interim Staff Guidance 4, (DI&C ISG-04), Revision 1, "Highly-Integrated Control Rooms – Communications Issues (HICRc)" (**Reference** 34).

The architecture specification shall:

- Identify and define communications between safety channels, between safety divisions, and between safety related software and nonsafety related software, to ensure communications independence, such that communication malfunctions will not interfere with the execution of the safety function.

- Define a response for loss and restoration of communication, which shall set data and associated status to a defined state after significant data loss. Communication fault sources include message corruption due to errors in communication processors, errors in buffer interfaces, errors in transmission media, or from external interference. These errors are defined in DI&C ISG-04.

- Incorporate the Secure Development and Operational Environment security requirements, based on Appendix C.

- Incorporate the cyber security requirements, based on the cyber security program plan.

3.10.3.6 Software Safety Analysis Report

Software safety analysis is only required for safety related software. The Development Team shall perform and the Software V&V team shall independently review the Software Safety Analysis on the design and design output documents, as shown in Figure 4. The SSA shall be performed and documented in a Software Safety Analysis Report in accordance with the SSPP (see Section 6).

3.10.3.7 Secure Development and Operational Environment Analysis Report

A Secure Development and Operational Environment Analysis shall be performed on design output documents, as shown in Figure 4. The SDOE Analysis shall be performed in accordance with Appendix C, coordinated with the cyber security program plan, and documented in the SDOE Analysis Phase Report.

3.10.3.8 Configuration Management Assessment

This task shall be conducted during or immediately before baseline review to ensure:

- Requirements phase activities are completed on the required outputs
- Adequacy of quality assurance as defined in SQAP
- Appropriate configuration controls (according to SCMP) are in place to monitor design activities including document revision and track changes control.

3.10.3.9 Baseline Review Report

A Baseline Review shall be performed by the Baseline Review Team at the end of the Requirements Phase. The review shall be performed and documented in a Baseline Review Report as specified in the SVVPP (see Section 4.2.6.6)).

3.11 Design Phase

3.11.1 Overview

The Design Phase shall perform the following:

- Translate the software requirements into a software design.
- Define the interfaces between software components.
- Define new or refine existing coding conventions to be used in implementation of the design.
- Evaluate suitability of previously developed and commercial-off-the-shelf software for use in the design, including other Nuclear Quality Aspects of commercial grade dedication, using the processes defined in EPRI technical report TR-106439 (**Reference** 32), and considering the guidance provided in EPRI TR-107339 (**Reference** 38) and 1011710 (**Reference** 31).
- Develop a plan for testing the integrated software.

The Design Phase inputs, outputs, and activities are illustrated in Figure 5.

The Design Phase activities are defined in Sections 3.11.3.2 through 3.11.3.8 of the SDPP.

3.11.2 Design Phase Inputs

The Design Phase input documents shall be the Requirements Phase output documents listed in Table 3.

3.11.3 Design Phase Outputs

The required Design Phase output documents are listed in Table 4. V&V is performed in accordance with the SQAPP and cyber security program plan.

Table 4. Design Phase Outputs

Output Document	Responsible Organization
Software Design Description	Software Development Team
Intra-System Communication Protocol Specification	Software Development Team
Software Coding Convention and Guideline Documents	Software Development Team
Software Tool Documentation Package	Software Development Team
PDS Evaluation Report and Documentation Package	Software Development Team
COTS Evaluation Report and Documentation Package	Software Development Team
Software Validation Test Plan and Test Cases Specification	Software Test Team for safety and nonsafety Group 1 systems Software Development Lead for nonsafety Group 2 systems
Software Safety Analysis Report	Software Safety Team (Perform) Software V&V Team (Review)
Secure Development and Operational Environment Analysis Report	Toshiba and Toshiba's Contractors
Design Phase – RTM	Software Development Team (Perform) Software V&V Team (Review)
Configuration Management Assessment	Configuration Management Lead
Design Phase Baseline Review Report	Baseline Review Team for safety and nonsafety Group 1 systems Software Development Lead for nonsafety Group 2 systems

3.11.3.1 Software Design Description

The Software Design Description (SwDD) shall be prepared to provide a representation of the software architecture that will be used to satisfy the requirements identified in the SRS, to provide sufficient detail to allow persons other than the author of the SwDD to create, review, and test the software. The SwDD supports the V&V process by providing a design to allow starting development of validation testing and factory acceptance testing plans. The SwDD shall translate the software requirements into a

description of the software and its structure, software components, interfaces, and data necessary for implementation.

The SwDD shall clearly identify components that have safety related functions or features identified as potential sources of risk by the Software Safety Analysis.

The SwDD shall identify COTS or PDS that will be incorporated into the design, as well as providing references to the appropriate COTS and PDS evaluations of suitability for use.

3.11.3.2 Intra System Communications and Protocol Specification

An Intra System Communications and Protocol Specification (ISCPS) shall be prepared to define each communication interface established between the software modules that make up the system within the system or equipment. As with Inter-System Communication in Section 3.10.3.5, the specification shall conform to the requirements of DI&C ISG-04, Revision 1 (**Reference** 34).

The ISCPS shall:

- Identify and define each communication interface, message structure, format, and sequence that was not included in the Inter-System documents created in Section 3.10.3.5.
- Identify communication interfaces between software modules at different safety classification levels (e.g., communication boundary between a safety related and a nonsafety software module or system) and ensure the safety related systems can perform the required safety function or functions when communication errors occur.
- Define response to data loss or communication failures.
- Include SDOE requirements (Appendix C).
- Include cyber security requirements

3.11.3.3 Software Coding Conventions and Guidelines Document

The Software Coding Conventions and Guidelines Document shall establish the software coding practices to be followed in the implementation of software design for all classes of systems and equipment to which this Software Program Plan applies. The guideline document shall provide coding practices that will result in readable, consistent, correct, maintainable, reliable, and efficient source code. The guidelines shall be programming language or development platform specific. Revision of an existing document is encouraged.

The guidelines provided shall include:

- Code formatting guidelines
- Commenting guidelines
- Techniques for declaration and naming of variables
- Technology specific coding practices to be applied as well as practices to be avoided

- Coding practices to include maintainability, readability, robustness, calculations, timing dependability, and traceability
- In-code comment documentation formatting and required content
- Code version tracking practices, including change identification within the code
- Architectural practices to be avoided
- Other coding guidance specific to each technology, based on guidance such as NUREG/CR-6463 (**Reference** 33)

3.11.3.4 Software Tool Evaluation Documentation

Software tools used in within each software life cycle shall be identified in the Design Phase. Each tool shall be evaluated for suitability of use for the safety criticality of the software to which the tool will be applied or the criticality of each tool's application. These tool evaluations shall be documented. All tool evaluation documentation shall be treated as quality records.

All tool evaluation documentation shall be placed under configuration management in accordance with the Software Configuration Management Plan (see Section 7.3.1).

The Software Development Lead shall be responsible for performing and documenting software tool evaluations. The Software Verification and Validation Lead shall be responsible for review of software tool evaluations. The staff that uses the tool shall be involved in selection and evaluation of software tools.

Tools supplied to the customer shall include those tools necessary to support all supplied databases. For those tools supplied to the customer or those tools recommended for purchase by the customer, tool evaluation documents shall be supplied to the customer with the software tools. The methods used to accept tools shall be documented.

Software tools shall be evaluated in a manner consistent with the guidance of IEEE Std. 7-4.3.2 (**Reference** 11). Since software tools are not perfect, and since purchased software tools cannot be credited with having been generated under a software life cycle process with the same or an equivalent high quality life cycle process as required for the software upon which the tool is being used, an alternative method must be used to accept the output of software tools. Therefore, the safety related output of the software tool shall be subject to V&V, to determine that the output of that tool meets the requirements established during the previous life cycle phase, and complies with the functional, performance, and interface requirements.

Tool operating experience should be an appropriate input into tool selection, as it provides additional confidence in the suitability of a tool. However, operating experience with the tool alone shall not be considered as sufficient to determine that the tool is the equivalent of a tool developed using a high quality, appropriate life cycle process, or that the tool will function as required in the development of applications.

For all software products, the criteria and results providing the documented basis to accept the tool as sufficient to credit for V&V activities shall be documented in a software tool report. The software tool report shall be reviewed and approved by the applicable Toshiba and Toshiba's contractors' Project Manager (PM).

The Software Tool Report shall contain:

- Unique tool identification (e.g., software name and version)
- Purpose of the support tool
- Acceptance criteria
- Verification methods
- An evaluation of the acceptability of the software tool for its intended use

The software tool report and other documentation on the support tool shall be included in the Software Tool Documentation Package.

For software products that are safety related and for software products that are classified as nonsafety related Group 1, the software tool report is required, shall be reviewed by the Software Verification and Validation Lead, and shall be approved by the Software Quality Assurance Lead. For software products that are classified as nonsafety related Group 2, the software tool report should be generated, should be reviewed by the Software Verification and Validation Lead, and should be approved by the Software Quality Assurance Lead.

3.11.3.5 PDS Evaluation Report and Documentation Package

Previously Developed Software (PDS) shall be evaluated to ensure it is suitable and qualified for the intended application. The rigor with which the PDS is evaluated shall be commensurate with the safety classification of the application. PDS should be evaluated in accordance with guidance such as EPRI technical report TR-106439 (**Reference** 32), EPRI TR-107339 (**Reference** 38), and EPRI 1011710 (**Reference** 31). Other commercial grade dedication activities shall be performed in accordance with Nuclear Quality Assurance programs. The additional documentation required for commercial grade dedication of the digital content shall augment the documentation required for commercial grade dedication of the equipment that includes the digital content.

The evaluation process shall include the following tasks:

- Evaluate the SQAP and SSP requirements applied in the PDS development with the SQAP and SSP requirements established in the planning phase for this system or logical group of systems. The evaluation shall determine whether the requirements are met or additional V&V activities are needed.
- Review the requirements and V&V activities performed in development of PDS. Any additional requirements or verification activities needed for the intended application shall be identified.
- Review the operating history of the product.
- Identify and review any relevant problem reports and their disposition. Ensure there are no unresolved problems that may affect the safety function of the intended application.
- Review the development process documentation and identify differences between available documentation and that required for the application.

The results of the evaluation shall be documented in the PDS Evaluation Report. The report, at a minimum shall:

- Identify the PDS.
- Describe the function of the PDS and previous applications.
- Describe the evaluation processes and acceptance criteria applied.
- Any deviations or deficiencies found in the review process.

The PDS Evaluation Report shall be verified in accordance with the SVVP prior to the PDS software being incorporated into the design.

In the case that the evaluation acceptance criteria were not met, the PDS should be re-engineered. This requires that the applicable V&V and SSA activities to be performed and documented in accordance with the software program manual. The supplemental activities and documentation shall be verified in accordance with the SVVP.

The PDS Evaluation Report and any other applicable documentation, including plans, reviews, and other documentation associated with re-engineering, shall be included in the PDS Evaluation Package prior to being incorporated into the design, and shall filed in the permanent project files. The PDS Evaluation Package shall be provided to the customer.

3.11.3.6 COTS Software Evaluation Report and Commercial Grade Dedication

Commercial Off-The-Shelf software, equipment, and systems, for safety related use of commercial software, equipment, and systems developed outside a 10 CFR 50 Appendix B program as safety related equipment, shall be evaluated to ensure it meets the quality level required prior to incorporation into the design. COTS software used in nonsafety applications should be evaluated for use in a similar manner. The rigor with which the COTS software, equipment, and systems are evaluated shall be commensurate with the classification of the intended application. COTS software, equipment, and systems used in plant safety systems shall be evaluated in accordance with the EPRI guidance on commercial grade dedication of equipment with digital content in EPRI technical report TR-106439 (**Reference** 32). The requirements of TR-107339 (**Reference** 38) and 1011710 (**Reference** 31) shall be included in these evaluations. These three EPRI technical reports shall be applied to Nonsafety Group 1 applications, unless equivalent documentation is provided concerning the acceptability of the software, which shall be reviewed and approved by the customer according to the customer's request. Evaluations of COTS and PDS software in Nonsafety Group 2 should be performed in accordance with these three EPRI technical reports. The additional documentation required for commercial grade dedication of the digital content shall augment the documentation required for commercial grade dedication of the equipment that includes the digital content

The evaluation process shall include the following activities:

- Review of development process and its documentation
- Review of the software (e.g., adherence to accepted coding practices, internal consistency, and readability)
- Review of the qualification and experience of personnel involved in design and verification

- Review of Toshiba organizatino's and Toshiba's contractors' Software Quality Assurance program
- Review of Toshiba organizatino's and Toshiba's contractors' configuration control program
- Review of the product operating history
- Review of the reported problems and their dispositions to ensure they do not impact the safety function of the application

The result of the evaluation process shall be captured in COTS Software Evaluation Report. The report shall:

- Identify the COTS software, using configurations and version numbers as applicable
- Describe the COTS software function
- Describe the evaluation processes and acceptance criteria applied
- Any deviations or deficiencies found in the review process
- Intended application of the COTS software within the design

In the case that the evaluation acceptance criteria were not met, the COTS should be re-engineered. This requires that the applicable V&V and SSA activities to be performed and documented in accordance with the software program manual. The supplemental activities and documentation shall be verified in accordance with the SVVP.

The COTS Evaluation Report shall be generated by the staff performing the evaluation. The COTS Evaluation Report shall be verified in accordance with the SVVP and SSP for the applicable system or logical group of systems, prior to being incorporated into the design. The COTS Evaluation Report and any other applicable documentation, including plans, reviews, and other documentation associated with re-engineering, shall be included in the COTS Evaluation Package and shall be filed in the permanent project files. The COTS Software Evaluation Package shall be provided to the customer.

3.11.3.7 Software Validation Test Plan and Test Cases Specification

The Software Validation Test Plan and Test Cases Specifications shall outline the methodology of how various tests will be used to verify that the integrated software meets the requirements stated in the SRS. The test plan and specification shall identify environments, cases (to include inputs, procedures, outputs, and expected results), resources (to include tools, personnel, and equipment), methodologies, and acceptance criteria.

These activities shall be initiated in the Design Phase to ensure completion prior to the Test Phase. The Baseline Review on the Software Validation Test Plan and Test Cases Specifications should be performed in this Design phase. A Baseline Review shall be performed on the Software Validation Test Plan and Test Cases Specifications no later than the end of the Implementation Phase.

The Software Validation Test Plan and Test Cases Specifications shall be traceable to the SRS.

3.11.3.8 Test Procedure Development

The development of the following test procedures will be started no later than during the Design Phase:

- Platform factory test, including input from the Secure Development and Operational Environment (SDOE) and from the cyber security program, and
- Platforms integration test, and including input from the SDOE and from the cyber security program.

The requirements and responsible personnel for the tests listed above are described in the Software Test Program Plan in Section 9. A customer validation and integration test plan will describ additional requirements. These test procedures should be completed and baseline reviewed during this phase. These test procedures shall be completed and baseline reviewed prior to use.

3.11.3.9 Software Safety Analysis Report

The Development Team shall perform and the Software V&V team shall independently review the SSA on the design output documents as shown in Figure 5. The SSA shall be performed and documented in a Software Safety Analysis Report in accordance with the SSPP (see Section 6).

3.11.3.10 SDOE Analysis Report

SDOE Analysis shall be performed on design output documents, as shown in Figure 5. The SDOE Analysis shall be performed in accordance with Appendix C, coordinated with the cyber security program plan, and documented in the SDOE Analysis Phase Report.

3.11.3.11 Configuration Management Assessment

This task is conducted during or immediately before baseline review to ensure:

- Design phase activities are completed on the required outputs
- Adequacy of quality assurance as defined in SQAP
- Appropriate configuration controls (according to SCMP) are in place to monitor design activities including document revision and track changes control.

3.11.3.12 Baseline Review Report

A BR shall be performed by the Baseline Review Team at the end of the Design Phase. The review shall be performed and documented in a Baseline Review Report as specified in the SVVPP (see Section 4.2.6.6).

3.12 Implementation Phase

3.12.1 Overview

The Implementation Phase shall perform the following:

- Translate the design outlined in the SwDD into modular software source code or equivalent, appropriate for the platform to be applied
- Review, unit test, integrate and integration test, and approve a software release for validation testing

The Implementation Phase inputs, outputs, and activities are illustrated in Figure 6.

3.12.2 Implementation Phase Inputs

The Implementation Phase input documents shall be the Design Phase output documents listed in Table 4.

3.12.3 Implementation Phase Outputs

The required Implementation Phase output documents are listed in Table 5. Verification and validation shall be performed in accordance with the SVVPP and cyber security program plan.

Table 5. Implementation Phase Output

Output Document	Responsible Organization
Integrated Application / Source Code	Software Development Team
Software Functional Testing (Unit/Module Test Plan, Procedures, and Test Case Specifications)	Software Test Team for safety and nonsafety Group 1 systems Software Development Lead for nonsafety Group 2 systems
Software Validation Test Plan, Procedure, and Test Case Specification	Software Test Team for safety and nonsafety Group 1 systems Software Development Lead for nonsafety Group 2 systems
Software Functional Test Report	Software Test Team for safety and nonsafety Group 1 systems Software Development Lead for nonsafety Group 2 systems
Software Build Description	Software Development Team
Software Safety Analysis Report	Software Safety Team (Perform) Software V&V Team (Review)
SDOE Analysis Report	Toshiba and Toshiba's Contractors
Implementation Phase – RTM	Software Development Team (Perform) Software V&V Team (Review)
Configuration Management Assessment	Configuration Management Lead
Implementation Phase Baseline Review Report	Baseline Review Team for safety and nonsafety Group 1 systems Software Development Lead for nonsafety Group 2 systems

3.12.3.1 Software Coding Readiness Review

The software coding readiness review shall be performed by the software development team prior to start of coding. This ensures:

- The software development team is familiar with the Software Requirements Specification (described in Section 3.10.3.1), Data Communications Architecture and Protocol (described in Section 3.10.3.5), Intra-System Communications and Protocol (described in Section 3.11.3.2), Software Design Description (described in Section 3.11.3.1), and Software Coding Conventions and Guidelines Document (described in Section 3.12.3.2).
- Software tools needed for development are evaluated, approved, under configuration control, and available for use.
- PDS and COTS software, systems, and equipment are evaluated, approved, under configuration control, and available for use.

3.12.3.2 Software Coding

Software coding is the process of implementing the software design, described in the SwDD, into a format that software tools can translate into binary code that can be understood by a computer, or a set of configuration bits that establish the logic in programmable logic. The source code is written in human-readable language and translated into executable code through preexisting software. The source code shall be written in a manner consistent with the Software Coding Conventions and Guidelines Document, in Section 3.11.3.3, applicable to the system or equipment.

In addition, each component or unit of the code shall contain comments that describe the purpose of the software, version description, inputs to the unit, software classification, processing algorithm performed, and outputs from the unit. Meaningful comments shall be provided within the code to support review, test, and long-term maintenance.

3.12.3.3 Code Review

As determined during the design of each SVVP, designated software code units and/or software code units meeting criteria defined in each SVVP shall be subjected to code review to ensure the source code complies with the design outlined in the design input documents and fulfills the SRS functional requirements. Code reviewers shall be qualified engineers with sufficient understanding of the software language as well as understanding of the software design and intended application. Staff training and qualification requirements are provided in the Software Training Program Plan, Section 10.

For safety related software, the code reviewer or reviewers shall be a part of the V&V team.

For nonsafety related Group 1 software, the code reviewer or reviewers should be from the independent V&V team, but may be a part of the design team. If any code reviewer was part of the design team, that code reviewer shall not have been responsible for or an active participant in any portion of the code design and shall not have set major design features of the code design or software architecture.

For nonsafety related Group 2 software, the code reviewer or reviewers can be part of the design team but shall not be the same engineer or engineers responsible for the code design and none of the reviewers shall have set major design features of the code design or software architecture.

The code review process shall, at a minimum, evaluate the code for:

- Compliance with coding conventions defined in the Software Coding Conventions and Guidelines Document. Toshiba or Toshiba's contractor shall define a Software Conventions and Guidelines document applicable to the computer language used in each system or logical group of systems.
- Inclusion of sufficient comments to allow an experienced programmer other than the author to read, understand, and modify the code without extensive reverse engineering
- Proper syntax
- Readability
- Correctness or the extent to which program satisfies requirements
- Robustness

- Reliability
- Maintainability
- Testability

Acceptance criteria shall be specified and any discrepancies and reviewer comments shall be documented in individual Code Review Data Sheets. Each Code Review Data Sheet shall include:

- Software unit reviewed revision/version information, and classification
- Identity of code reviewer or reviewers and code designer
- Discrepancies found and reviewer's comments
- Resolution of each discrepancy

The Code Review Data Sheets will be incorporated into the Software Implementation Review Report.

3.12.3.4 Software Functional Testing

Software Functional Testing shall be performed during the Implementation Phase to ensure that each software unit and integrated units satisfy the requirements provided in the design documents.

The Software Functional Testing shall have the objectives to:

- Identify and correct code design errors prior to the Testing Phase
- Verify the software units interface properly

Unit tests shall be performed to test the correctness and robustness of the code. Unit tests shall include response to input data outside design limits, response time, and fault injection in addition to simple functionality tests and structured basis testing. Integration tests shall test the ability of the software units to interface correctly and perform the intended function.

Unit tests shall be performed before integration tests, although unit tests may include logical groups of software units.

Any failures in integration tests that require changing units that have already been unit tested shall require at least an evaluation of the unit tests and repetition of affected tests, and should repeat the complete unit test on modified units and all interfaces between unmodified and modified software units.

A summary of test activities, including a basis for determining the rigor of testing and the test results, shall be prepared and documented in the Software Functional Test Report. The Software Functional Test Reports shall be reviewed and approved.

If software modifications are necessary after this phase is completed, the functional tests shall be performed again and the results of the retest activities shall be added to the Software Implementation Review Report.

For safety software, this testing shall be performed by an independent Test Team from the V&V organization, based on the requirements for independence defined in Section 1.4.2.1. For nonsafety software, this testing shall be performed by an independent Test Team, based on the requirements for independence defined in Section 1.4.2.2.

3.12.3.5 Software Implementation Review Report

The Software Implementation Review Report shall combine the Software Functional Test Reports (see Section 3.12.3.4) and Code Review results (see Section 3.12.3.3). The Software Implementation Review Report will serve as an input into the Software Build Description (SBD). The Software Implementation Review Report shall be treated as a quality record and retained for the life of the system or logical group of systems.

3.12.3.6 Software Build Procedure and Report

The Software Build Procedure and Report (SBPR) shall include the procedure that shall be followed to create a software build and the results of that build. The SBPR shall also contain the Software Functional Test Report for the software modules used to construct the software. The SBPR shall document the names of the source code files and the exact process followed to compile the source code. The SBPR shall document the process required to load the software build into the production hardware and, if necessary into prototype hardware, for Validation Testing. The process shall be documented at a level of detail sufficient to allow someone other than the Toshiba or Toshiba's contractors' staff to configure the system in preparation for loading the software and to load the software. This ensures that the software build can be replicated and that the software is ready for Validation Testing. The build procedure shall ensure that the build does not contain any undocumented code or configuration.

3.12.3.7 Finalize Software Validation Test Plan and Test Cases Specification

The Software Validation Test Plan and Test Cases Specification defined in Section 3.11.3.7 shall be finalized no later than the completion of the Implementation Phase. Baseline Review shall be performed on the Software Validation Test Plan and Test Cases at the end of the Implementation Phase, unless BR has already been completed on these items.

3.12.3.8 Software Safety Analysis Report

The Development Team shall perform and the Software V&V team shall independently review the SSA on the design output documents as shown in Figure 6. Each SSA shall be performed and documented in a Software Safety Analysis Report in accordance with the SSPP (see Section 6).

3.12.3.9 SDOE Analysis Report

SDOE Analysis shall be performed on design output documents, as shown in Figure 6. The SDOE Analysis shall be performed in accordance with Appendix C, coordinated with the cyber security program plan, and documented in the SDOE Analysis Phase Report.

3.12.3.10 Configuration Management Assessment

This task is conducted during or immediately before baseline review to ensure:

- Implementation phase activities are completed on the required outputs
- Adequacy of quality assurance as defined in the SQAP
- Appropriate configuration controls (according to the SCMP) are in place to monitor design activities including document revision and track changes control.

3.12.3.11 Implementation Phase Baseline Review Report

A baseline review shall be performed by the Baseline Review Team at the end of the Implementation Phase. The review shall be performed and documented in a Baseline Review Report as specified in the SVVPP (see Section 4.2.6.6).

3.13 Testing and Integration Phase

3.13.1 Overview

The purpose of the Testing and Integration Phase is to demonstrate the software performs the intended function by performing validation of the software on the target or production hardware. The Test and Integration Phase inputs, outputs, and activities are illustrated in Figure 7.

Verification and validation ends when then the validation testing is complete. There is no intent or requirement for verification and validation to be part of the plant Installation, Pre-Operation, or Startup Testing. The Design Installation Verification (DIV[8]) process is included in the Plant Installation and Construction (i.e., Commissioning) Testing. However, modifications made to any software after the completion of validation testing shall still be analyzed for appropriate regression testing under the configuration management process. All required regression verification and validation testing must be performed before the modification can be completed and the CM open item closed.

3.13.2 Testing and Integration Phase Inputs

The Testing and Integration Phase design input documents shall be the Implementation Phase design output documents listed in Table 5.

3.13.3 Testing and Integration Phase Outputs

The required Testing and Integration Phase design output documents are listed in Table 6. Verification and validation shall be performed in accordance with the SVVPP (Section 4.3.7) and the cyber security program plan.

[8] DIV is defined as: "Verification that the installed configuration of safety related I&C equipment is bounded by the test configuration and test conditions or that an analysis exists which concludes that any differences will not affect the safety function of the I&C system."

Table 6. Testing and Integration Phase Output

Output Document	Responsible Organization
Software Validation Test Report	Software V&V Team for Safety Related Software Development Team for Nonsafety Related
Software Release Notes	Software Development Team
PFT Plan and Procedure	Software Test Team
Input from SDOE and Cyber Security for the PFT Plan and Procedure	Software Development Cyber Security Teams (perform) Software Verification and Validation Team (review)
PIT Plan and Procedure	Integration and Integration Test Team
Input from SDOE and Cyber Security for the PIT Plan and Procedure	Software Development, and Cyber Security Team (perform) Software Verification and Validation Team (review)
Software Safety Analysis Report	Software Safety Team (Perform) Software V&V Team (Review)
SDOE Analysis Report	Toshiba and Toshiba's Contractors
Testing and Integration Phase – RTM	Software Development Team (Perform) Software V&V Team (Review)
Configuration Management Assessment	Configuration Management Lead
Testing and Integration Phase Baseline Review Report	Baseline Review Team for safety and nonsafety Group 1 systems Software Development Lead for nonsafety Group 2 systems

3.13.3.1 Software Validation Testing

Software Validation Testing verifies proper functionality of the fully integrated software once installed on the production hardware. Software Validation Testing requirements and responsible personnel shall be described in detail in the SVVPP (see Section 4).

Software Validation Testing may use the same test procedures written for Platform Factory Test (PFT), as defined in Section 3.14.3.

The results of Software Validation Testing shall be documented in a Software Validation Test Report. The requirements for the Software Validation Test Report are described in the SVPP (see Section 4). The Software Validation Test Report shall be incorporated in the Testing and Integration Phase Baseline Review Report.

3.13.3.2 Production Release

Once a software build meets the requirements of the Software Validation Testing, it shall be released as Production Software through the Baseline Review process. A software quality assurance audit shall be performed on the software build as part of the Testing and Integration Phase Baseline Review.

3.13.3.3 Software Safety Analysis Report

The Development Team shall perform and the Software V&V team shall independently review the SSA on the design output documents as shown in Figure 7. Each SSA shall be performed and documented in a Software Safety Analysis Report in accordance with the SSPP (see Section 6).

3.13.3.4 SDOE Analysis Report

SDOE Analysis shall be performed on design output documents, as shown in Figure 7. The SDOE Analysis shall be performed in accordance with Appendix C, coordinated with the cyber security program plan, and documented in the SDOE Analysis Phase Report.

3.13.3.5 Configuration Management Assessment

This task is conducted during or immediately before baseline review to ensure:

- Testing and Integration phase activities are completed on the required outputs
- Adequacy of quality assurance as defined in the SQAP
- Appropriate configuration controls (according to the SCMP) are in place to monitor design activities including document revision and track changes control.

3.13.3.6 Testing and Integration Phase Baseline Review Report

A baseline review shall be performed by the Baseline Review Team at the end of the Testing and Implementation Phase when all verification, validation, and analysis activities are complete. The review shall be performed and documented in a Baseline Review Report as specified in the SVVPP (see Section 4.2.6.6).

3.14 Installation Phase

3.14.1 Overview

The Installation Phase shall ensure that the system performs the intended function when integrated with other systems as well as when installed in the plant.

All software installed in target hardware shall be under configuration management control to ensure continuity and change management processes are in place prior to PFT.

The installation phase is broken down into four phases:

- In Phase I, testing shall be performed to verify that a system or a logical group of systems function properly prior to shipment at the site where development occurred. This phase shall be referred to as a Platform Factory Test (PFT) for the customer. After successful completion of PFT, some or all of the system (or simulator for the communication interfaces) shall be delivered to a central location to be integrated for Platforms Integration Test (PIT) with the remainder shipped to the customer site.

- In Phase II, testing shall be performed to verify that individual systems or groups of systems tested in Phase I perform as required when integrated with other plant systems that have also completed PFT. This phase shall be referred to as PIT. After completion of PIT, the systems shall be either stored or delivered to the customer site for installation.

- In Phase III, each system or logical group of systems shall be installed in the plant, integrated with sensors, transmitters, actuators, human-system interfaces, local control panels, electrical and fluid systems, and tested for correct operation. This testing shall include verification that any changes made after completion of PFT and/or PIT is performed correctly. The specific content of this phase of testing shall be defined in the specific test plan and procedures to be defined for each system or logical group of systems. These system-specific plans and procedures shall be defined in the customer program plan. For the customer, the Design Installation Verification (DIV) is included in the installation and commissioning activities.

- In Phase IV, all the systems and the human-system interfaces shall be exercised under operating conditions as the plant is tested for correct operation. The specific content of this phase of testing shall be defined in the specific test plan and procedures to be defined for the plan. These plant-specific plans and procedures shall be defined in the customer program plan. For the customer, this phase is referred to as Pre-Operational Testing.

3.14.2 Installation Phase Inputs

The required Installation Phase input documents are listed below in Table 7.

Table 7. Installation Phase Input Documents

Installation Phase Input Documents	Responsible Organization
Phase I, PFT Plan and Procedures	Software Test Team
Phase I, Input from SDOE and Cyber Security for the PFT Plan and Procedures	Software Development Team and Cyber Security Team
Phase II, PIT Plan and Procedures	Integration and Integration Test Team
Phase II, Input from SDOE and Cyber Security Team for the PIT Plan and Procedures	Cyber Security Team
Phase III, DIV	Software Installation Team
Phase III, Cyber Security During DIV	Cyber Security Team
Phase IV, Pre-Operational Testing Plan and Procedures	Software Installation Team
Phase IV, Cyber Security Pre-Operational Testing Plan and Procedures	Cyber Security Team
Software portions of the System O&M Manual	Software Development Team
Software Training Manual	Software Development Team
SDOE Analysis Reports	Toshiba and Toshiba's Contractors
Configuration Management Assessment	Configuration Management Lead
Phase Baseline Review Reports	Baseline Review Team for safety and nonsafety Group 1 systems Software Development Lead for nonsafety Group 2 systems

3.14.3 Installation Phase Outputs and Activities

The Installation Phase output documents and activities are defined below.

- PFTs shall be executed for each system or logical group of systems to validate that the software and hardware has been correctly integrated into a system, configured, and calibrated. PFT shall be performed using the hardware and validated software to be delivered to the customer. The PFT and test documentation requirements are described in the SVVPP and the STPP.

  For Phase I, PFT, production software and hardware is the responsibility of Toshiba and a Toshiba's contractor, as applicable. The PFT Test Report shall be prepared by Toshiba and a Toshiba's contractor, as applicable. The SDOE Analysis Test Report shall be prepared by Toshiba and/or a Toshiba's contractor responsible for the system being tested. Testing is performed in accordance with the Software Test Program Plan (STPP). Verification and validation is performed in accordance with the SVVPP, STPP, and SDOE requirements (Appendix C). Review of the cyber security portions of the PFT are the responsibility of the Toshiba cyber security team, if otherwise required by the customer.

  For testing at PFT, the process shall ensure that the software is completely rebuilt and installed prior to starting PFT, where the technology allows. This rebuild and reinstall ensures that the build and installation procedures provided by the vendor are appropriate and complete. The intent of this

reinstallation is to ensure that the rebuilt software completely defines the configuration necessary to operate the system, and that no configuration items are not set in system memory (which would be unlikely to exist in new hardware installed as part of maintenance) are required, but which are not included in the installation procedures. This requirement would not apply to technologies like the Toshiba non-rewritable field programmable gate arrays used in the Reactor Trip and Isolation System and Neutron Monitoring System, ultra-violet erasable programmable read-only memory, or flash type technologies that cannot be programmed in place. However, this requirement would apply to flash or other electrically erasable memories that can be programmed in place. When such a clean rebuild and installation is not performed, a written rationale, including the compensatory measures taken to ensure correctness of the instructions and configuration data to be provided to the customer, shall be provided to the customer for review and approval according to the customer's request.

- The PIT shall be executed to validate that the individual systems or logical groups of systems tested in PFT performs as required when integrated with other plant systems that have also completed their PFTs. Applicable portions of the Plant Data Network shall be included in PFT and PIT as appropriate.

  For Phase II, PIT, the PIT Test Report are the responsibility of the Integration and Integration Test Team. The SDOE Analysis Test Report shall be prepared by the Integration and Integration Test Team. Testing is performed in accordance with the STPP. Verification and validation is performed in accordance with the SVVPP, STPP, and SDOE requirements in Appendix C. Review of the Cyber Security portions of the PFT are the responsibility of the Toshiba cyber security team, if otherwise required by the customer.

- The Design Installation and Verification (DIV) scope is included in the Installation and Commissioning Tests scope. DIV shall be performed to validate the performance of the system when installed in the plant and connected to the sensors, transmitters, actuated devices, human-system interfaces, and all other equipment attached to the Plant Data Network (PDN). DIV is defined in the validation and integration test plan.

  For Phase III, DIV Test records are to be maintained in accordance with the plans, procedures, and processes defined for site installation. The testing reports shall be reviewed Toshiba or Toshiba's contractor's inspection and test staff. The customer will review the reports. The Cyber Security Test Report is the responsibility of the customer Cyber Security staff. The Cyber Security Analysis Test Report shall be prepared in accordance with the cyber security program plan. Testing is performed in accordance with the STPP. Any verification and validation of modifications is performed in accordance with the SVVPP, STPP, and cyber security program plan.

- Pre-Operational Tests shall be performed to validate the performance of all systems and the plant, in preparation for commercial operation.

  For Phase IV, Pre-Operational Testing, records are to be maintained in accordance with the plans, procedures, and processes defined by the customer and/or Toshiba and Toshiba's contractors for Pre-Operational Testing. The testing reports shall be reviewed by the customer inspection and test staff. The Cyber Security Test Report is the responsibility of the customer Cyber Security staff. The Cyber Security Analysis Test Report shall be prepared in accordance with the cyber security program plan. Testing is performed in accordance with the Software Test Program Plan (STPP). Any verification and validation of modifications is performed in accordance with the SVVPP, STPP, and cyber security program plan.

3.14.3.1 System Operations and Maintenance Manuals

This plan requires the generation of System Operations and Maintenance Manuals, which combine the hardware, software, and support tools into a single, unified manual or set of manuals. These manuals shall satisfy the requirement for the Software Operations and Maintenance Manuals in the IEEE standards. These manuals shall be generated for an individual system, or for logical groups of systems. The requirements and content of the System Operations and Maintenance Manuals shall be described in requirements documents to be supplied later by the customer Instrumentation and Controls (I&C) Manager.

The System Operations and Maintenance Manuals shall provide hardware, software, and system installation instructions. The instructions shall provide sufficient information to install all software and configuration data into the system or equipment. The installation instructions shall consider the technology provided. As examples:

- The software for the Toshiba Field Programmable Gate Array (FPGA) safety systems can only be installed by replacing modules in those systems. Other equipment may have programmable read-only memory with software. Therefore, installation instructions shall document procedures for module replacement.

- For some equipment, the configuration data can be stored in nonvolatile re-writable memory, which can be modified in the field. Therefore, installation instructions shall document procedures for module configuration.

- The software for some equipment is saved in nonvolatile, re-writable memory, and can be reloaded in the field. In this case, the installation instructions shall document procedures for software loading, backup, and restoration.

- The software and/or configuration for some equipment are saved in memory that depends on external power, backed up by batteries or large capacitors. After a period without external power, this equipment will require reload of the software and/or configuration. In this case, the installation instructions shall document procedures for software loading, backup, and restoration.

- The software and/or configuration for some equipment are saved in memory that depends on external power, with no back up by batteries or large capacitors. After external power is lost, this equipment will require reload of the software and/or configuration. In this case, the installation instructions shall document procedures for software loading, backup, and restoration as well as the procedures for dealing with whatever media is used to allow automatic restart of such equipment after power failure.

3.14.3.2 System Training Manuals

The requirements and responsible personnel for the System Training Manuals shall be described in guidance to be provided by the customer I&C for content and format of the customer Operations and Maintenance Manuals. The System Training Manuals shall include training for support software, the base software supplied with the platform (as applicable), application software, and tools needed for calibration, surveillance, troubleshooting, maintenance, software modification, software reload, system backups, and other normally expected the customer functions.

3.14.3.3 SDOE Analysis Report

SDOE Analysis shall be performed in accordance with Appendix C, coordinated with the cyber security program plan, and documented in the SDOE Analysis Phase Report.

3.14.3.4 Configuration Management Assessment

This task is conducted during or immediately before baseline review to ensure:

- Integration phase activities are completed on the required outputs
- Adequacy of quality assurance as defined in the SQAP
- Appropriate configuration controls (according to the SCMP) are in place to monitor design activities including document revision and track changes control.

3.14.3.5 Installation Phase Baseline Review Report

This task is conducted prior to baseline review to ensure:

- Installation phase activities are completed on the required outputs
- Adequacy of quality assurance as defined in the SQAP
- Appropriate configuration controls (according to the SCMP) are in place to monitor design activities including document revision and track changes control.

3.14.3.6 Installation Phase Baseline Review Report

Baseline Reviews are performed for each phase in the installation phase. The reviews shall be performed and documented in separate Baseline Review Reports as specified in the SVVPP (see Section 4).

3.15 Operations Phase

3.15.1 Overview

The Operation Phase begins once Pre-Operational Testing is complete and the software products have been installed, commissioned, and pre-operational tested in the customer.

The Operations Phase proceeds for each system until a Corrective Action Program entry is required, documenting a concern, issue, or error. Then, the Maintenance Phase shall be entered. Evaluation of the Corrective Action Program entry is performed in the Operations Phase, with a resolution either to accept-as-is, which does not require any action other than documenting the rationale for accepting the issue as-is, or to modify the software. The Maintenance Phase remains in effect for this issue until the Corrective Action Program entry is resolved and closed. Then, the Operations Phase resumes.

3.15.2 Operations Phase Inputs

Since there are no software modifications, enhancements, or installations performed under this phase, this phase has no inputs or outputs associated with this Software Program Plan. Monitoring of system operation and cyber security shall be performed under the customer plans, procedures, and instructions.

3.15.3 Operations Phase Outputs

Outputs from the Operations Phase include:

- Corrective Action Program entries for resolution in the Maintenance Phase
- Records from the customer monitoring of system operation
- Historical data from the systems
- Cyber security evaluations, based on the customer cyber security program

3.16 Maintenance Phase

3.16.1 Overview

The Maintenance Phase may be entered at any time after the Pre-Operational Testing is complete and the software products have been installed, commissioned, and pre-operational tested in the customer.

The Operations Phase proceeds for each system until a Corrective Action Program entry is required, documenting a concern, issue, or error. Then, the Maintenance Phase shall be entered. Evaluation of the Corrective Action Program entry is performed, usually in the Operations Phase. If the Corrective Action Program determines that software modification is required, the Maintenance Phase is started and remains in effect for this issue until the Corrective Action Program entry is resolved and closed. Then, the Operations Phase resumes.

3.16.2 Maintenance Phase Inputs

The Maintenance Phase is initiated by the identification of a nonconformance in software operation. This issue shall be documented in a Corrective Action Report, which shall be used as an input document to the Change Control process documented in Section 7.4.2.

The applicable portions of the vendor and the customer plans shall be documented in the resolution for the Corrective Action Report, which shall result in modifications to existing documents created by these plans, as well as new, similar documents. Since no new types of documents are created by this plan, their descriptions are not repeated here.

3.16.3 Maintenance Phase Outputs

Table 8. Maintenance Phase Output

Maintenance Phase Output Documents	Responsible Design Organization
Revised documentation	The customer, may be delegated to Toshiba or a Toshiba's contractor
Revised Production Build	The customer, may be delegated to Toshiba or a Toshiba's contractor
Supplemental Software Validation Testing Report	The customer, may be delegated to Toshiba or a Toshiba's contractor
Software Safety Analysis Reports	The customer, may be delegated to Toshiba or a Toshiba's contractor
SDOE Analysis Report	The customer, may be delegated to Toshiba or a Toshiba's contractor
Supplemental Baseline Review Reports	The customer, may be delegated to Toshiba or a Toshiba's contractor

3.16.4 Maintenance Phase Activities

When software modification or enhancement is required, the customer is expected to follow the process described in this Software Program Plan (SPP). Modifications should be performed under either the original software plans associated with the system where the problem can be corrected, or a revised version of those software plans. At a minimum, the following items shall be performed:

- All affected documentation shall be revised,
- Review and testing activities shall be performed,
- Appropriate installation, commissioning, and testing activities shall be defined,
- Instructions for installation of the revised software and/or hardware shall be written, reviewed, and approved by the customer.

3.17 Retirement Phase

The scope and content of the Retirement Phase is dependent on the actions being taken by the customer. At present, only the base requirements are provided, with the expectation that the customer staff shall define and perform the actions required for replacement in accordance with the customer plans, procedures, and instructions, and that the software in a replacement system shall be designed, implemented, tested, and reviewed in accordance with this software life cycle.

3.17.1.1 Retirement Phase Activities

The Retirement Phase shall be initiated when a system upgrade, replacement, or modernization is initiated. The Retirement Phase includes the following activities:

- User notification
- Evaluation of affected systems
- Cyber Security processing for the retired system and documentation, in accordance with the cyber security program plan
- Comparison between the software and its upgrade
- Documentation of activities and archiving of records

The Change Control process shall be used to evaluate, approve, and implement changes that result in the retirement of the software product and controlled documents. The Change Control process is described Section 7.4.2.

Table 9. Development Activities Assigned to Each Software Life Cycle Phase

Life Cycle Process	Development																		Operation			Maintenance		
	Planning V&V Activity			Requirements V&V Activity			Design V&V Activity			Implementation V&V Activity			Test V&V Activity			Installation/ Checkout V&V Activity			Operation V&V Activity			Maintenance V&V Activity		
Software Classification	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2
Development Activities																								
Plans for the Software Design Process Plans (no baseline review required)	TC	TC	TC																					
Platform Factory Test (PFT) Plan	STT	STT	STT																					
Platforms Integration Test (PIT) Plan	TT	TT	TT																					
Software Safety Analysis Report	SST			SST			SST			SST			SST						CTM			CTM		
Secure Development and Operational Environment Analysis Report	CST	CST	CST	CST	CST	CST	CST	CST	CST	CST	CST	CST	CST	CST	CST	CST	CST	CST	CTM	CTM	CTM	CTM	CTM	CTM
Planning Phase Requirements Traceability Matrix	DT	DT	DT	DT	DT	DT	DT	DT	DT	DT	DT	DT	DT	DT	DT									
Configuration Management Assessment	CM	CM	CM	CM	CM	CM	CM	CM	CM	CM	CM	CM	CM	CM	CM	CM	CM	CM						
Baseline Review Report	BRT	BRT	DT	BRT	BRT	DT	BRT	BRT	DT	BRT	BRT	DT	BRT	BRT	DT	BRT	BRT	DT	CTM	CTM	CTM	CTM	CTM	CTM
Software Requirements Specification				DT	DT	DT																		
Hardware Requirements Specification				TC	TC	TC																		
System Architecture Description				DT	DT	DT																		
Software Interfaces Document				DT	DT	DT																		
Data Communication Protocol and Architecture				DT	DT	DT																		
Software Design Description							DT	DT	DT															
Intra-System Communication Protocol Specification							DT	DT	DT															

Table 9. Development Activities Assigned to Each Software Life Cycle Phase

Life Cycle Process	Development																		Operation			Maintenance		
	Planning V&V Activity			Requirements V&V Activity			Design V&V Activity			Implementation V&V Activity			Test V&V Activity			Installation/ Checkout V&V Activity			Operation V&V Activity			Maintenance V&V Activity		
Software Classification	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2
Development Activities																								
Software Coding Convention and Guideline Documents							DT	DT	DT															
Software Tool Documentation Package							DT	DT	DT															
PDS Evaluation Report and Documentation Package							DT	DT	DT															
COTS Evaluation Report and Documentation Package							DT	DT	DT															
Software Validation Test Plan and Test Cases Specification							STT	STT	DT															
Integrated Application / Source Code										DT	DT	DT												
Software Functional Test Plan, Procedures, and Test Cases										STT	STT	DT												
Software Validation Test Plan, Procedure, and Test Case Specification										STT	STT	DT												
Software Functional Test Report										STT	STT	DT												
Software Build Description										DT	DT	DT												
Software Validation Test Report													V&V	DT	DT									
Software Release Notes													DT	DT	DT									
PFT Plan and Procedure													STT	STT	STT									
PIT Plan and Procedure													TT	TT	TT									
Phase I, PFT Plan and Procedures																STT	STT	STT						

Table 9. Development Activities Assigned to Each Software Life Cycle Phase

Life Cycle Process	Development																		Operation			Maintenance		
	Planning V&V Activity			Requirements V&V Activity			Design V&V Activity			Implementation V&V Activity			Test V&V Activity			Installation/ Checkout V&V Activity			Operation V&V Activity			Maintenance V&V Activity		
Software Classification	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2
Development Activities																								
Phase II, PIT Plan and Procedures																TT	TT	TT						
Phase III, DIV																SIT	SIT	SIT						
Phase III, Cyber Security During DIV																CST	CST	CST						
Phase IV, Pre-Operational Testing Plan and Procedures																SIT	SIT	SIT						
Phase IV, Cyber Security Pre-Operational Testing Plan and Procedures																CST	CST	CST						
Software portions of the System O&M Manual																DT	DT	DT						
Software Training Manual																DT	DT	DT						
Revised documentation																			CTM	CTM	CTM	CTM	CTM	CTM
Revised Production Build																			CTM	CTM	CTM	CTM	CTM	CTM
Supplemental Software Validation Testing Report																			CTM	CTM	CTM	CTM	CTM	CTM

SR = Safety related G1 = Nonsafety Group 1 G2 = Nonsafety Group 2		BRT=Baseline Review Team CM=Configuration Management Lead CST= Cyber Security Team DT=Software Development Team TC= Toshiba and Toshiba's contractors SIT= System Installation Test Team	SST=Software Safety Team CTMr= the customer, may be delegated to Toshiba ro Toshiba's contractor STT=Software Test Team V&V=V&V Lead TT= Integration and Integration Test Team

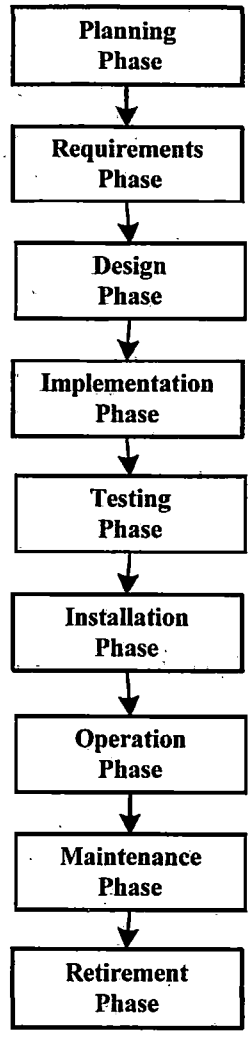

Figure 2. Software Life Cycle Process Overview

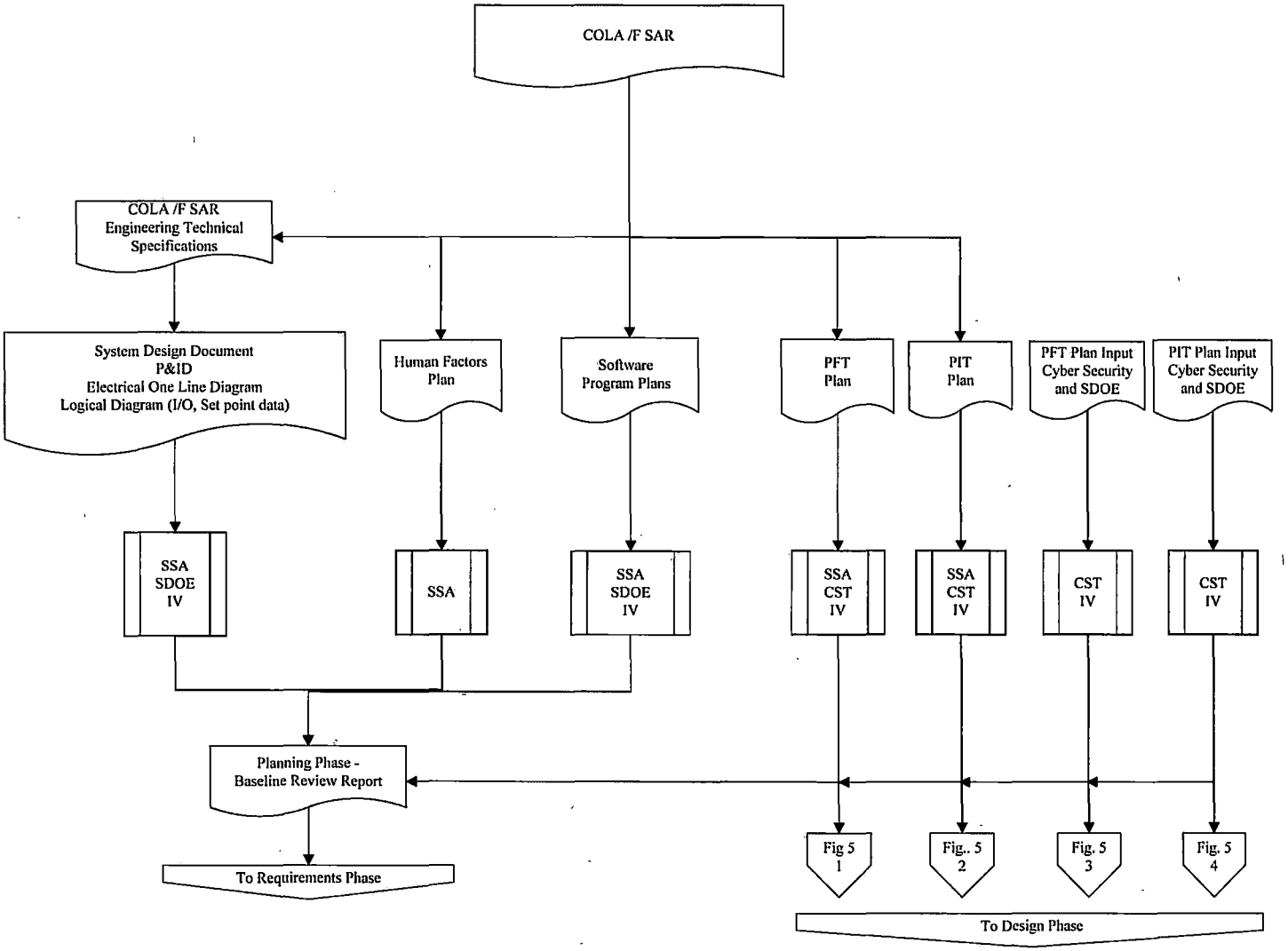

Figure 3. Software Life Cycle Process: Planning Phase

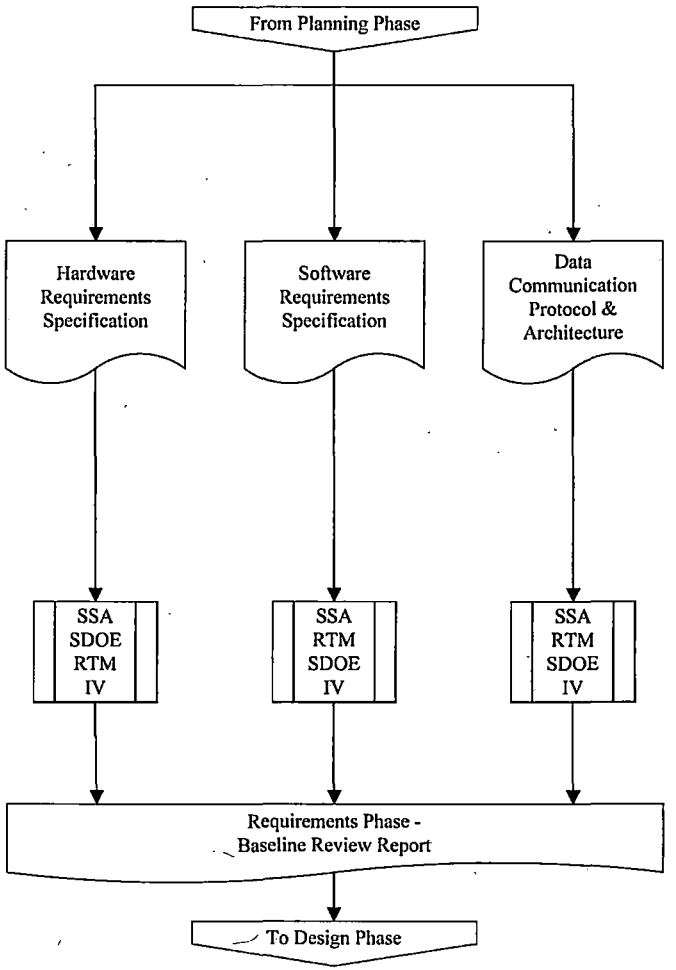

Figure 4. Software Life Cycle Process: Requirements Phase

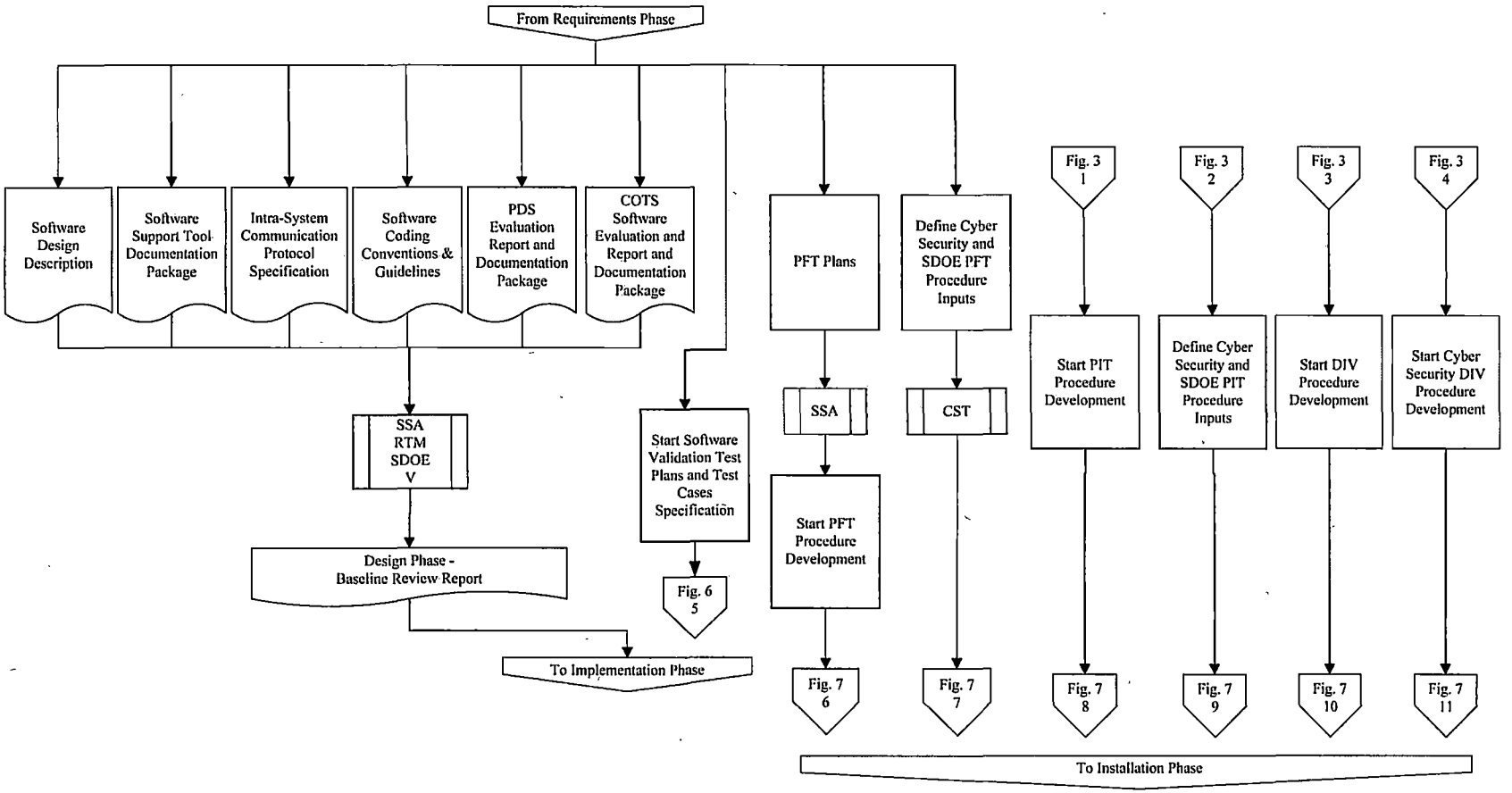

Figure 5. Software Life Cycle Process: Design Phase

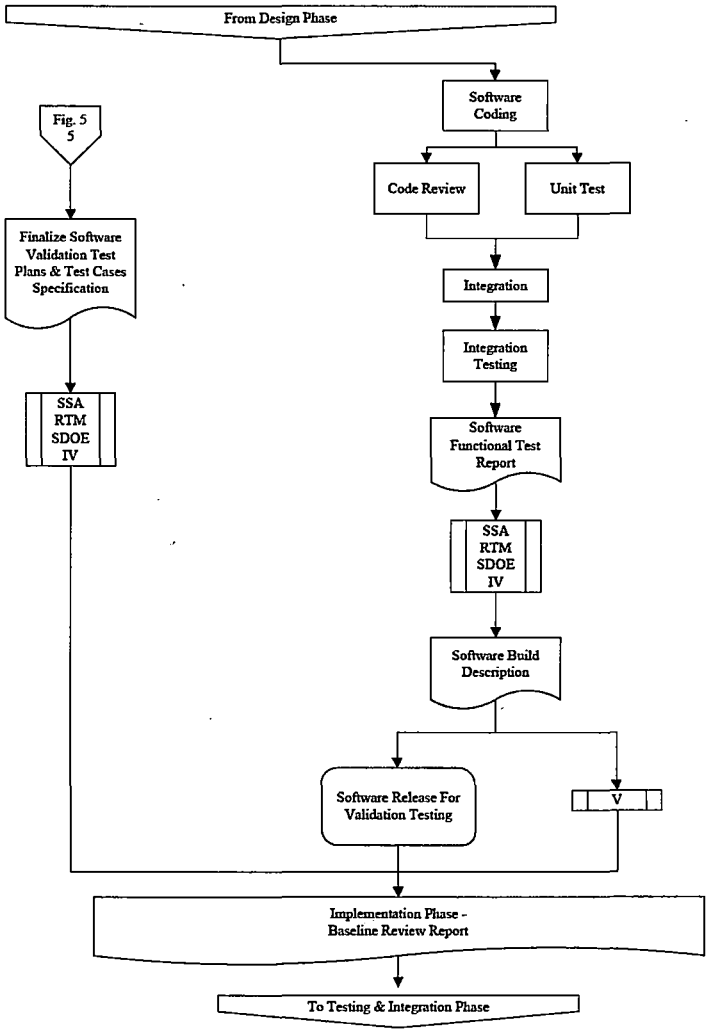

Figure 6. Software Life Cycle Process: Implementation Phase

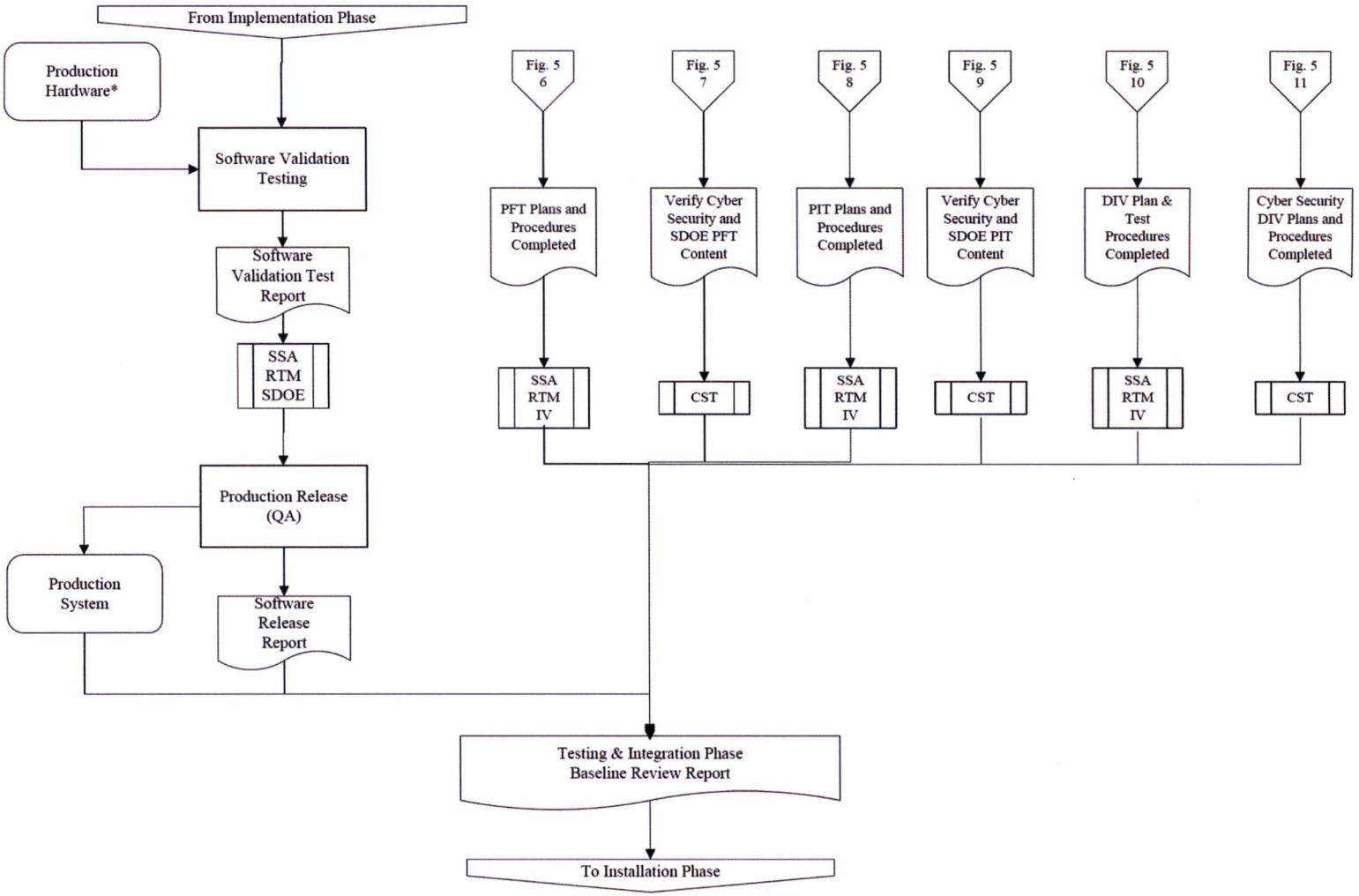

Figure 7. Software Life Cycle Process: Testing and Integration Phases

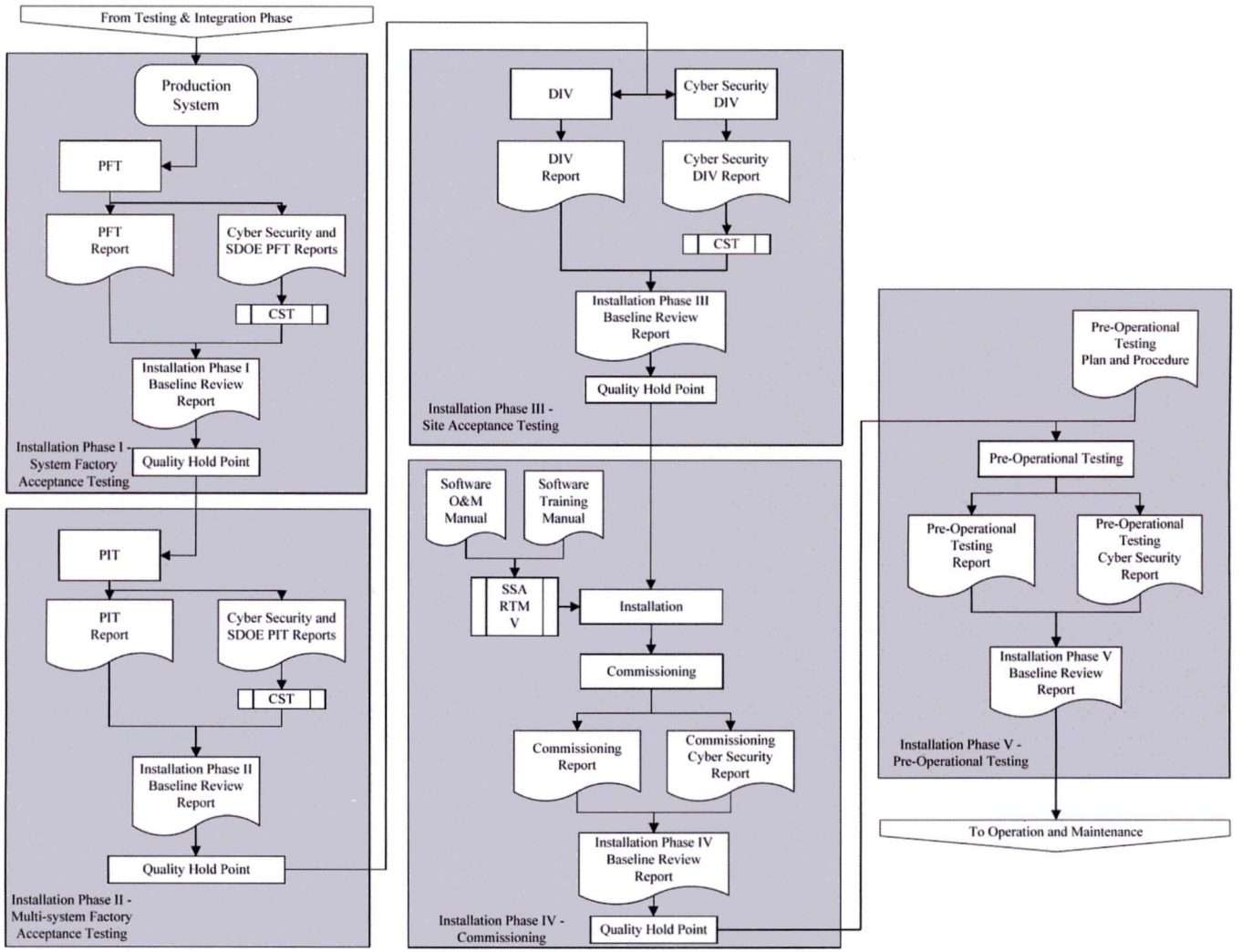

Figure 8. Software Life Cycle Process: Installation Phase

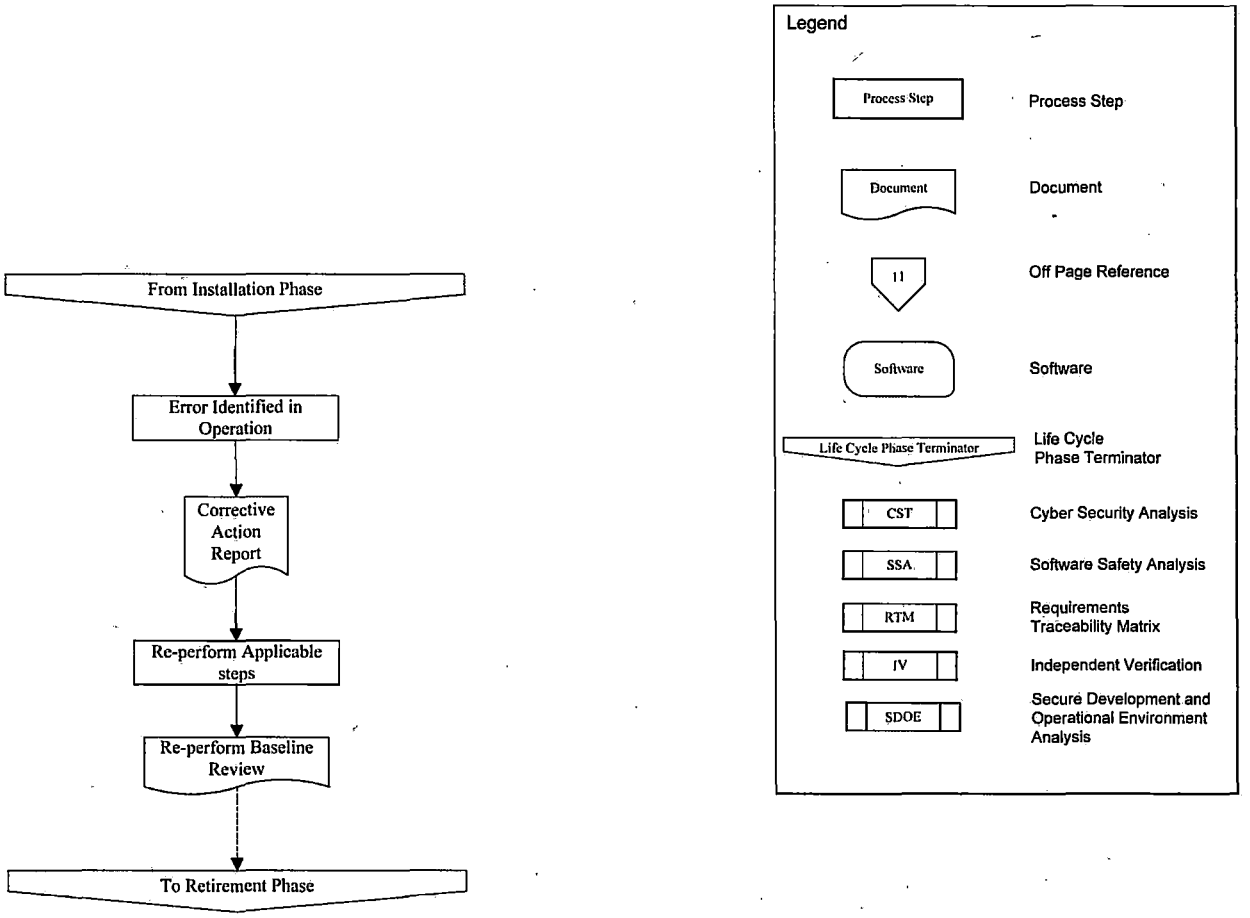

Figure 9. Software Life Cycle Process: Operation and Maintenance Phases

4 Software Verification and Validation Program Plan (SVVPP)

4.1 Introduction

The Software Verification and Validation Program Plan (SVVPP) defines the Verification and Validation (V&V) activities for all software. For safety systems, this SVVPP complies with the verification and validation program of IEEE Standard 1012, as endorsed by USNRC Regulatory Guide 1.168, specifically including the requirements of Section 1.4 of this SPP. This SVVPP provides the requirements for V&V activities for nonsafety systems.

4.1.1 Purpose

The purpose of the SVVPP is to specify the V&V activities for the software life cycle. V&V activities are used to ensure that the development of software products meets the specified requirements. The SVVPP shall be applied to all software intended for use in nuclear safety-related, nonsafety related Group 1, and nonsafety related Group 2 system applications (see Section 1.10). This SVVPP shall be used as a guide for the creation of Software Verification and Validation Plans (SVVP), which shall be written for each system or logical group of systems. Each SVVP shall be prepared, reviewed, approved, and retained as a quality record.

This SVVPP specifies the V&V activities required to produce robust, safe software products that accomplish the following:

- Implements the safety function or functions assigned to that system correctly
- Satisfies its specified requirements, user needs, industry standards, and regulatory requirements and expectations
- Performs all intended functions within the predetermined design and provides a high level of assurance that it does not perform unintended and undesirable functions
- Is complete, correct, accurate, testable, traceable, and consistent with the provided requirements
- The system provided matches the design basis documents

The software V&V activities shall be performed throughout software development as required by this SVVPP and each SVVP.

4.1.2 Scope

The V&V activities and tasks covered by the SVVPP are specified in Section 4.3 and are organized by life cycle phase. The SVVPP covers V&V activities for plant software as defined by this Software Program Plan (SPP) as well as V&V activities used to verify and validate the software at its hardware and user interfaces.

Verification and validation ends with the completion of Platform Factory Test and Platform Integrated Test. Site testing is not considered part of the verification and validation activities. However, changes made and tested during site testing shall be processed under this SVVPP and each system's or logical group of systems' SVVP. After the system is installed, V&V of modifications shall credit the testing performed at site, including regression testing.

Each SVVP shall be applied to the entire software life cycle for software products intended for use in nuclear safety related, nonsafety related Group 1, and nonsafety related Group 2 plant systems in the customer that are:

- Prepared and maintained by Toshiba and Toshiba' contractors up through construction
- Prepared and maintained by the customer and its subcontractors for the life of the plan
- Purchased as commercial-off-the-shelf (COTS) software, Commercial Grade Dedicated, and used for safety-related applications in the customer
- Purchased as COTS and used in Nonsafety Group 1 or Group 2 plant applications in the customer
- Purchased or applied as Previously Developed Software (PDS) and used in plant applications in the customer

Commercial grade dedication, evaluation and qualification of equipment containing COTS and PDS software in safety related applications shall be performed in accordance with EPRI TR-106439 (**Reference** 32). The requirements of EPRI TR-107339 (**Reference** 38) and 1011710 (**Reference** 31) shall be included in these evaluations. These three EPRI technical reports shall be applied to Nonsafety Group 1 applications, unless equivalent documentation is provided concerning the acceptability of the software, which shall be reviewed and approved by the customer according to the customer's request. Any evaluations of COTS and PDS software in nonsafety Group 2 could be performed in accordance with the same three EPRI references.

Verification and validation of safety systems is based on the Software Integrity Level (SIL) 4 rating provided in IEEE Std. 1012 (**Reference** 20) and RG 1.168, Regulatory Position 1 (**Reference** 4). The SVVP shall discuss the approach used to ensure that V&V activities for Nonsafety Group 1 or Group 2 software are appropriate for the system technology and the safety significance of the software.

This SVVPP shall be implemented by Toshiba and Toshiba's contractors, who supply software-based systems and/or equipment. Additional oversight shall be supplied for subcontractors by Toshiba or a Toshiba's contractor responsible for that subcontractor. Additional SQA oversight shall be supplied by the customer QA and I&C organizations, with additional support as deemed necessary by the customer.

4.1.3 [Deleted]

4.1.4 Relationship of the SVVPP to Other SPP Sections

In order to simplify the description of processes and ensure clarity in establishing roles and responsibilities and to align with the Branch Technical Position 7-14 (**Reference** 3) guidance, the SPP has been split into several sections. Each section of this SPP describes the processes, roles,

and responsibilities for the activities in that section. Each section either explicitly or implicitly refers to other sections as necessary to implement additional processes.

Thus, this SVVPP defines the methods to be used to verify and validate that the software design, development, and implementation activities, using other portions of the activities defined in Section 2, Section 3, Sections 5 through 11, Section 13, and Appendix C of this SPP. The project management functions defined in Section 12 are implemented in the customer plans, procedures, and instructions necessary to operate and maintain the nuclear power plants.

Thus, for each of the sections, the SVVPP provides the technical review and test activities necessary to implement the life cycle activities specified by the life cycle. The SVVPP interfaces with the other sections of this SPP as follows:

- All requirements provided in Section 1 of the SPP apply to the SVVPP, including system and software organization, roles and responsibilities, classification of systems and software, and the life cycle for system and software development.

- The SPMPP, Section 2, controls and oversees the design, development, and implementation performed by the V&V organization defined in accordance with the technical and process requirements listed in Section 3, Development.

- The SDPP, Section 3, processes work cooperatively with this SVVPP, as required to perform the necessary review, test, and other Verification and Validation (V&V) activities in accordance with the technical and process requirements of this section.

- The SVVPP works cooperatively with the software quality assurance program, which ensures that the V&V organization works in accordance with their plans, procedures, and engineering instructions. The implementation of the SDPP and SVVPP is overseen by the software quality assurance activities performed by the quality assurance organization, in accordance with the technical and process requirements listed in Section 5, Software Quality Assurance.

- The SVVPP works cooperatively with the Software/System Safety organization to ensure that the design, development, implementation, review, test, and other V&V activities are performed in a manner that ensures that safety is maximized and that identified safety concerns are implemented correctly as well as reviewed and tested completely. The Software/System Safety work is performed in accordance with the plans, procedures, and engineering instructions in accordance with the technical and process requirements listed in SPP Section 6, Software Safety.

- The SVVPP activities coordinate with the change control and configuration management activities. The change control and configuration management activities are performed in accordance with the technical and process requirements listed in Section 7, Configuration Management. Change control will be invoked on each work product preferably after the work product has been released for V&V, and is invoked after timely completion of V&V on each work product.

- The SVVPP activities ensure that the system and software integration activities are controlled and performed in accordance with the technical and process requirements listed in SPP Section 8, Software Integration.

- The SVVPP activities coordinate or perform (depending on the safety classification) system and software testing activities in accordance with the technical and process requirements listed in Section 9, Testing.

- The SVVPP activities use appropriately trained personnel in all phases of the software and system life cycle, and coordinate with this activity to ensure that training materials and other manuals are reviewed to ensure that plant staff can be trained appropriately, in accordance with the technical and process requirements listed in Section 10, Training.

- The SVVPP activities ensure that software installation at the vendor site and at the nuclear plant have appropriate documentation and procedures such that the installations can be controlled and performed in accordance with the technical and process requirements listed in Section 11, Installation.

- There are no SVVPP activities or requirements associated with plant operations, as these will be controlled by the plans, procedures, and instructions provided by the customer for the normal, expected plant functions that include operation, surveillance, calibration, modifications to tunable constants and other configuration items, and troubleshooting in accordance with the technical and process requirements listed in Section 12, Operations.

- The SVVPP is invoked as required to implement the changes required to close anomaly reports and implement enhancements to plant systems in a controlled manner, in accordance with the technical and process requirements listed in Section 13, Maintenance.

- The SVVPP verifies compliance, testability, and other V&V actions in software work, in accordance with the technical and process requirements listed in Appendix C, Secure Development and Operational Environment, to ensure that work performed does not compromise safety, for all safety systems and for other systems where cyber security requirements apply.

4.2 Verification and Validation Overview

Verification consists of reviews performed on the results of each development phase to ensure the phase was completed appropriately and correctly. Validation is used to ensure that the final product satisfies the user requirements. Validation shall be performed on the final product, although validation may be necessary or performed prior to the final code being produced. The overall V&V process includes inspections, evaluations, and tests of intermediate and final software products produced throughout the life cycle.

4.2.1 Organization

A typical organizational structure is illustrated in Figure 1. Other organizations are acceptable as discussed in and subject to the constraints of Section 1.4.

The V&V effort shall be led by the Software V&V Lead. The V&V organization shall have sufficient autonomy and fiduciary independence to perform all V&V tasks without interference from or unacceptable control by the software development organization. The SVVP shall describe the system specific V&V organization and its independence, as defined in Section 1.4.2. The independence of the Software V&V Lead is defined in Section 1.4.2.

4.2.2 Schedule

Activities within this plan shall be planned and scheduled, in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

4.2.3 Resource Summary

The SVVP shall list the personnel fulfilling the roles defined in the V&V organization defined in Section 4.2.4 as well as identify other personnel participating in V&V activities. The SVVP shall also list any facilities or equipment essential to the completion of the V&V activities.

4.2.4 Roles and Responsibilities

The software V&V effort shall be managed and led by the Software V&V Lead. A Software V&V Team works with and for the Software V&V Lead to accomplish V&V tasks and activities.

Software V&V Lead – The Software V&V Lead for each system or logical group of systems shall be responsible for implementation of all V&V activities as specified in this SVVPP. Each Software V&V Lead shall have the authority and responsibility to select or reject personnel assisting in the V&V effort. Each Software V&V Lead shall have the responsibility of identifying and requesting additional staff from the appropriate Project Manager, as necessary. Each Software V&V Lead shall have the responsibility of oversight of the V&V Team and of work performed by the Development Team, to ensure efforts are being completed correctly.

Each Software V&V Lead shall be responsible for the following:

- Drafting each SVVP in accordance with the requirements of this SVVPP,
- Recommending to the Project Manager responsible for V&V or to corporate executive management the size and makeup of the V&V Team,
- Filling the positions in the V&V Team with qualified, trained personnel,
- Implementing and completing the V&V efforts in accordance with each SVVP,
- Approving V&V outputs and documentation,
- Coordinating the V&V activities schedule with design activities,
- Coordinating with the Software Development Lead to get the proper design documentation in a timely manner,
- Ensuring each design meets specified requirements, user needs, and industry and regulatory standards,
- Ensuring all software performs all intended functions with a high level of assurance that it does not perform unintended and undesirable functions,
- Identifying nonconformances for resolution if the designs are incomplete, incorrect, inaccurate, not testable, not traceable, and not consistent with the given requirements,

- Assigning tasks to the V&V Team, assisting with V&V effort, and overseeing the V&V Team,
- Interfacing with the QA Manager to assist in quality activities, to include audits,
- Coordinating with the Baseline Review Team (BRT) to conduct the baseline review when the software life cycle is at the appropriate point,
- Preparing and implementing test plans and procedures according to each Software Test Plan (STP), which can be designated to the Software Test Lead,
- Maintaining configuration control of software for V&V according to each Software Configuration Management Plan (SCMP), which can be designated to the Software Configuration Lead,
- Ensuring that SDOE is implemented,
- Ensuring that team members are cleared for access, as necessary, or coordinate with the appropriately cleared Cyber Security Team who already has access to cyber security information, which may require clearance for safeguards or Official Use Only documents and information, and
- Overseeing various configuration management activities according to each Software Installation Plan (SInstP), which can be designated to Software Installation Lead.

The roles and responsibilities for the BRT and the QA Manager are described in the Software Quality Assurance Program Plan (SQAPP) (Section 5). The roles and responsibilities for the Software Test Lead are described in the Software Test Program Plan (STPP) (Section 9.1.3). The roles and responsibilities for the Software Configuration Lead are described in the Software Confirmation Management Program Plan (SCMPP) (Section 7.2.2). The roles and responsibilities for the Software Installation Lead are described in the Software Installation Program Plan (SInstPP) (Section 11.2.2).

4.2.5 Qualifications

The personnel performing V&V tasks and efforts shall be qualified to the extent necessary for the performance of their V&V duties. All personnel performing V&V tasks shall be aware of and comply with the requirement for independence from the development organization. All personnel performing V&V tasks shall be at least as qualified as the Development team to write similar software. All personnel performing V&V tasks shall be qualified to review the software and other work products assigned to them, and shall be responsible for raising issues of their own qualification to the V&V Team Lead as necessary.

4.2.6 Tools, Techniques, and Methodologies

Each SVVP shall describe the tools, techniques, and methodologies that will be used during the V&V activities. **Reference** shall be provided to the procedures, processes, engineering instructions, and other documented methods used in verification and validation.

Tools shall be evaluated for suitability and documented in accordance with each Software Development Plan (SDP, see Section 3.11.3.4). Configuration control of software tools is managed in accordance with the requirements of the SCMP (see Section 7.3.1).

Tools may be credited as part of the verification and validation activities, which requires a documented basis to accept the tool as sufficient to credit for V&V activities. For tools used only as aids to review, a documented basis to accept the tool as sufficient to credit for V&V activities of the tool is not required, although evaluation and documentation shall be performed. Where tools, techniques, or methods require a documented basis to accept the tool as sufficient to credit for V&V activities (for example, automated tests or static code analysis) each SVVP that invokes those tools shall define how the this acceptance and evaluation will be accomplished and how the tools, techniques, or methods will be used, to include any limitations on use.

This section lists some examples of common techniques and methods to be used for verification and validation, based on the needs and requirements of the software being evaluated.

4.2.6.1 Verification

Verification shall be performed in all phases of the software life cycle, to determine if design outputs of the phase are completed appropriately and correctly and if those design outputs fulfill the established requirements of the phase. Verification may be conducted through design review. The verifier shall be knowledgeable of the design yet be separate from those actually responsible for the design. Results of the verification shall be documented. The SVVP shall specify procedures to be followed when performing and documenting verification.

4.2.6.2 Code Review and Walk-through

During code reviews and walk-through, the source code shall be traced to the design specifications to verify correctness, consistency, completeness, and accuracy. Source code shall also be reviewed for compliance with coding standards and for sufficient commenting. The code review and walk-through shall be performed by an individual knowledgeable of the design yet separate from those actually responsible for the design. Results of the code review and walk-through shall be documented. The SVVP shall specify procedures to be followed when performing and documenting code review and walk-through.

4.2.6.3 Software Unit Test

Software unit testing shall be required for each software module or logical group of software modules for safety systems and for nonsafety Group 1 systems. Software unit testing should be performed for each software module or logical group of software modules for nonsafety Group 2 systems. Unit testing shall be performed to ensure that each software module operates as specified in the design documents. Unit testing shall be performed by the V&V team with independent review by the design staff. Results of the software unit tests shall be documented. Each SVVP shall specify procedures to be followed when performing and documenting software unit tests.

4.2.6.4 Software Validation Test

Software validation testing shall be completed in accordance with written test procedures to demonstrate that the software performs all intended functions within the predetermined design, and provides a high level of assurance that the software does not perform unintended and

undesirable functions. The Software V&V Lead shall be responsible for the execution of software validation testing. Results of the software validation tests shall be documented. Each SVVP shall specify procedures to be followed when performing and documenting software validation tests.

4.2.6.5 Requirements Traceability Matrix

Requirements Traceability Matrices (RTMs) shall be generated by the Software Development Team and reviewed by the Software V&V Team to ensure the software has completely, accurately, correctly, and consistently addressed the requirements for safety and nonsafety Group 1 systems. The RTM should be performed for nonsafety Group 2 systems. The results of the requirements traceability analysis are captured in an RTM electronic database or in a paper document, either of which can be placed under configuration control and credited as design basis documents. The RTM shall trace the V&V activities (e.g., test cases and acceptance criteria) to the design elements (e.g., specifications and code) to the software requirements (e.g., SRS) to ensure all system requirements are addressed properly. Each SVVP shall specify the procedures to be followed and tools to be used when generating an RTM.

4.2.6.6 Baseline Reviews

The Baseline Review Team shall perform Baseline Reviews at the conclusion of each phase in the software life cycle to ensure that the required activities during that phase were completed in accordance with the Software Program Plan (SPP) and Toshiba and Toshiba's contractors' approved software life cycle plans. All software life cycle activities for a given phase shall be complete prior to initiating a baseline review. Any anomalies (including but not limited to correctness, consistency, completeness, traceability, ambiguity, and verifiability) found in baseline reviews shall be resolved through the software life cycle processes. The staff performing baseline reviews shall perform functional audits (see Section 5.6.2.1) as part of the Baseline Review, which shall include evaluation of review elements to include, but not be limited to, an evaluation of complete, consistent, correct, traceable, testable, verifiable, unambiguous requirements. Baseline Review and Baseline Review Reports are not required on programmatic documents.

The Baseline Review Team shall be comprised of members of the Software V&V Team appointed by the Software Verification and Validation Lead. At least one member of the BRT shall have sufficient clearance to access safeguards information.

The BRT shall prepare a report at the conclusion of each baseline review, which shall be retained as part of the project quality records. The QA Manager shall approve the Baseline Review Report. The Baseline Review Report shall:

- Describe the review scope,
- Identify the reviewers,
- Identify the persons contacted during the review,
- Document the outputs and versions reviewed,
- Contain a summary of the review results, and

- Describe recommendations and findings.

A separate Corrective Action Report is used for each finding. Since these are procedural compliance issues, each shall be assigned to the organization identified in the CAR, and should be tracked to completion by the QA organization.

4.2.6.7 Audit Support

Audits are conducted under the SQAPP. A description can be found in (Section 5.6.2).

4.3 Life Cycle Verification and Validation

Requirements for V&V activities supporting each life cycle phase are defined in Table 10 through Table 18. Each table provides requirements for:

- Tasks to be performed,
- Inputs for the phase,
- Methods and Procedures,
- Outputs from the phase, and
- The person responsible for ensuring that these actions are performed.

When applying Table 10 through Table 18, the requirements established in Table 19 for independence shall be applied. If no entry exists in Table 19, then that process does not have to be applied. In application of Table 10 through Table 19, both the written text and the table content shall be used.

Table 19 shall be applied to COTS software, PDS, and to configuration activities based on the classification of the system or equipment.

When the tables refer to the System Requirements Document (SyRD), the reference includes all other documents defined in Table 1.

4.3.1 Management of V&V Activities

Management of the V&V process is performed throughout all life cycle phases of the software. The Management activities do not constitute a separate life cycle phase, but rather an umbrella covering all phases. Management activities are defined in the Software Project Management Program Plan (SPMPP). Based on the SPMPP, individual Software Project Management Plans (SPMPs) shall be created for each system or logical group of systems. In support of each SVVP, each SPMP shall:

- Monitor the execution of the SVVP and analyze problems associated with the execution,
- Report progress of the various V&V activities,
- Ensure software being produced fulfills requirements,

- Evaluate testing results and check for completeness,
- Monitor V&V outputs and determine when a task is complete,
- Assess proposed changes to the software to identify affected requirements and any new hazards or risks as well as changing and re-performing V&V tasks as necessary to address the changes, and
- Determine when changes or updates to the SVVP are necessary.

The requirements for V&V Management activities are defined in Table 10.

Note that the SPMP must include definition of the required level of independence between the project manager responsible for design and development and the project manager responsible for V&V, if such independence is required.

4.3.2 Planning Phase V&V Activities

The objectives of the Planning Phase V&V Activities are to verify that all system requirements have been allocated and validate the selected implantation solution. Each SVVP shall address Planning Phase activities for each system or logical group of systems. The Planning Phases V&V Activities that shall be performed in each Planning Phase are defined in Table 11.

4.3.3 Requirements Phase V & V Activities

The objectives of the Requirements Phase V&V activities are to ensure correctness, completeness, accuracy, testability, and consistency of the software and interface requirements. Each SVVP shall address Requirements Phase activities for each system or logical group of systems. The Requirements Phase V&V activities that shall be performed in each Requirements Phase are defined in Table 12.

4.3.4 Design Phase V & V Activities

The objective of the Design Phase V&V activities are to demonstrate that the design correctly, accurately, and completely represent the software requirements without introducing any unintended features. Each SVVP shall address Design Phase activities for each system or logical group of systems. The Design Phase V&V Activities that shall be performed in each Design Phase are defined in Table 13.

4.3.5 Implementation Phase V & V Activities

The objectives of the Implementation Phase V&V Activities are to verify and validate that the code, database structures, and state machines are correct, accurate, and complete. Each SVVP shall address Implementation Phase activities for each system or logical group of systems. The Implementation Phase V&V activities that shall be performed in each Implementation Phase are defined in Table 14.

4.3.6 Testing and Integration Phase V & V Activities

The objectives of the Testing and Integration Phase V&V Activities are to ensure that the software requirements are satisfied by execution of integration, system, and acceptance tests. Each SVVP shall address Testing and Integration Phase activities for each system or logical group of systems. The Testing and Integration phase V&V activities that shall be performed in each Testing and Integration Phase are defined in Table 15. Note that PFT and PIT are considered part of V&V, but that all tests conducted on site at the customer are not considered part of the V&V program.

4.3.7 Installation Phase V & V Activities

The objectives of the Installation Phase and V&V Activities are to verify and validate the correctness of the software installation into the target environment. Each SVVP shall address Installation Phase activities for each system or logical group of systems. The Installation phase V&V activities that shall be performed in each Installation Phase are defined in Table 16.

Installation may occur several times; therefore, variations on these activities can occur at various times of the life cycle. These variations shall be controlled, documented, reviewed, and approved prior to their use. Note that installation of the system at the customer is not considered part of the verification and validation process. V&V does not include design installation verification (DIV). However, any changes to systems, hardware, or software that are tested as part of the DIV, installation, commissioning, pre-operational, or startup tests shall be processed in accordance with this SVVPP and each system's or logical group of systems' SVVP.

For PFT, software re-installation activities shall verify that the instructions provided are sufficient to re-install the software in the system. For site activities, DIV activities may require re-installation of the software, as mandated by an appricable cyber security program, along with testing that the system functions correctly in the plant environment, with all sensors and actuators, as well as interfaces to other systems.

4.3.8 Operation Phase V & V Activities

No software modifications, updates, enhancements, or other activities that change software configuration are planned for the Operation Phase. Therefore, no V&V activities are associated with this life cycle phase.

4.3.9 Maintenance Phase V & V Activities

The objectives of the Maintenance Phase Activities are to evaluate new constraints of the system, assess proposed changes, evaluate operating procedures, evaluate anomalies discovered during operation, assess migration and retirement requirements, and re-perform V&V tasks. When the Maintenance Phase is entered, a customized SVVP shall be written, as necessary, to address software life cycle activities for each system or logical group of systems. When anomalies are to be corrected, the Maintenance Phase shall invoke the required, documented portions of the software life cycle for the affected system. The Maintenance phase V&V activities that shall be performed in each Maintenance Phase are defined in Table 18.

4.3.10 Summary of V&V Activities

The V&V activities are listed in Table 19 along with designated individual responsibility assigned for those activities.

Within the table:

- Blank cells means no V&V activities are applicable.
- Entries within each cell designate the individuals that shall be responsible for the V&V activity within that life cycle, for that classification of system, which are defined as safety related (SR), Nonsafety Group 1 (G1), and Nonsafety Group 2 (G2).
- Entries are abbreviated, as Baseline Review Team (BRT), Configuration Control Board (CCB), Configuration Management Lead (CM), Design Team (DT), Human Factors Engineering (HFE), Project Manager (PM), Software Safety Lead (SSL), Software Safety Team (SST), and Verification and Validation Lead.

Table 10. Management of V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Software Verification and Validation Plan (SVVP) Update	SVVP (previous update) Contract Final Safety Analysis Report (FSAR) Engineering Technical Specifications System Requirements Document (SyRD) Human Factors Program Plans Development plans, including resource and schedule requirements	Update each SVVP throughout the life cycle Update milestones, tasks, technical reviews, schedules, resource requirements, and other documents need to control the project	SVVP (updated)	Software V&V Lead
Baseline Change Assessment	SVVP Proposed changes Safety Analysis Reports Risk Analysis Reports	Evaluate proposed software changes for effects on V&V tasks already complete. Plan reevaluation of task or create new V&V task to address changes Assess whether proposed changes are compliant with requirements and that changes do not produce new hazards or risks or affect already resolved hazards or risks.	Management Task Report: Baseline Change Assessment Anomaly report	Software V&V Lead
Management Review	SVVP Development plans, resources, and schedules V&V outputs for various tasks	Review V&V effort to define changes to or redirect V&V effort Determine when it is appropriate to move to the next set of V&V and development life cycle activities. Place V&V outputs under configuration control according the requirements in SCMPP Assess level of completion of V&V activities and verify tasks comply with requirements defined in SVVPP Verify V&V task results have a basis of evidence and based on these results provide recommendations for program acceptance as input to V&V final report	Management Task Report: Recommendations to V&V Activity Summary Report and Final Report, issued when each phase is completed, updated as necessary at the completion of any re-opened phase	Software V&V Lead

Table 10. Management of V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Management and Technical Review Support	V&V outputs for various tasks Materials for various reviews	Support management and technical reviews (e.g. design reviews) by attending reviews and assessing review materials and providing task and anomaly reports Verify delivery of all software products and documents according to schedule	Management Task Report: Review Results Anomaly reports	Project Manager
Organizational and Supporting Processes Interfaces	Software Test Plan (STP) Software Safety Plan (SSP) Software Quality Assurance Plan (SQAP) SVVP Software Installation Plan (SInstP) Software Configuration Management Plan (SCMP)	Coordinate V&V efforts development, safety, and Quality Assurance (QA) Leads. Support performance of baseline reviews and audits (see SQAPP, Section 5.6.1.1)	SVVP	Software V&V Lead

Table 11. Planning Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Software Verification and Validation Plan Generation	Contract FSAR Engineering Technical Specifications System Requirements Document (SyRD) Human Factors Program Plans Development plans, resources, and schedules	Generate a SVVP for the system or logical group of systems to cover the entire software life cycle to establish a baseline prior to Requirements V&V for non-programmatic documents Establish milestones, tasks, technical reviews, schedules, resource requirements, and other documents need to control the project	SVVP	Software V&V Lead
Concept Documentation Evaluation	FSAR Engineering Technical Specifications System Requirements Document (SyRD) Human Factors Program Plans Development plans, resources, and schedules User Needs	Verify the concept documentation satisfies the user needs/requirements and constraints and limitations of proposed approach and they are captured in the RTM. Verify that system requirements satisfy: • System functions • End-to-end performance • Feasibility and testability of functional requirements • System architecture design • Operation and maintenance requirements	Planning Task Report: Concept Documentation Analysis Anomaly Report RTM	Software V&V Lead
Program Plan Evaluation	Codes and standards Software Program Plan	Verify the various plans that comprise the software plan satisfy the user needs/requirements and capture the requirements of the codes, standards, and guidance. Verify these requirements are captured in the RTM.	Planning Task Report: Software Plan Evaluation Anomaly Report RTM	Software V&V Lead

Table 11. Planning Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Platform Factory Test (PFT) and Platforms Integration Test (PIT) Plans Generation	FSAR Engineering Technical Specifications System Requirements Document (SyRD) Human Factors Program Plans System Test Plan	Plan V&V system acceptance testing to validate software correctly implements requirements in expected operational environment. Validate PFT and PIT test plan has: • Test coverage of system requirements • Conformance to expected results • Feasibility and testability of operation and maintenance Plan tracing of PFT and PIT requirements to and documentation for test designs/cases/procedures/execution PFT and PIT test plans shall address compliance with acceptance requirements in the expected operational environment and adequacy of user documentation Verify PFT and PIT test plans conform to format defined in System Test Plan	PFT Test Plan PIT Test Plan Anomaly Report	Integration and Integration Test Team
Planning Traceability Analysis	FSAR Engineering Technical Specifications System Requirements Document (SyRD) Human Factors Program Plans RTM	Identify all system requirements implemented completely or partially by the software and ensure all of them are incorporated into the RTM.	RTM Anomaly Report	Software Development Lead (Prepare) Software V&V Lead (Review)
Safety Analysis	See SSPP (Section 6.4.2.1)	Analyze potential system hazard caused by software as specified in the SSPP (Section 6.4.2.1)	See SSPP (Section 6.4.2.1)	Software Safety Lead
Risk Analysis	Schedule	Identify and provide ways to reduce, mitigate, or eliminate technical and management risks	Planning Task Report: Risk Analysis	Project Manager
Phase Summary Report	All recorded V&V work completed for phase	Summarize tasks of the Planning phase, retained as quality records	Planning Phase V&V Summary Report	Software V&V Lead

Table 12. Requirements Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures	Required Outputs	Responsibility
Requirements Traceability Matrix	RTM Hardware Requirements Specification (HRS) Software Requirements Specification (SRS) Data Communication Protocol & Architecture (DCP&A)	Trace software requirements to system requirements and vice versa, to analyze for correctness, consistency, completeness, and accuracy. Ensure they are captured in the RTM. • Correctness – Verify relationship between software and system requirements is correct • Consistency – Verify relationship between software and system requirements are specified to a consistent level of detail • Completeness – Verify software requirements trace to system requirements with sufficient detail and all system requirements dealing with software trace to a software requirement • Accuracy – Software requirements accurately system and operating performance	RTM Anomaly Report	Software Development Lead (Prepare) Software V&V Lead (Review)
Software Requirements Evaluation	FSAR Engineering Technical Specifications System Requirements Document (SyRD) Human Factors Program Plans SRS DCP&A	Verify and validate requirements of SRS and DCP&A for correctness, consistency, completeness, accuracy, readability, and testability to ensure: • Software requirements satisfy system requirements within the constraints of the system • Software requirements comply with standards, regulations, and policies. • Sequences of states, state changes, and flow of date satisfy functional and performance requirements • Data usage and format appropriate • Consistency within software requirements • The following elements are defined o Functionality (including but not limited to: algorithms, reporting, and logging), o Interfaces (software/hardware), o Performance criteria (including but not limited to: timing sizing, speed, capacity, precision, safety, and security), and	Requirements Task Report: Software Requirements Evaluation Anomaly Report	Software V&V Lead for safety systems Software Development Lead for nonsafety systems

Table 12. Requirements Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures	Required Outputs	Responsibility
		○ System/device/software control (including but not limited to: initialization, transaction and state monitoring, and self-testing) Documentation is legible, understandable, and unambiguous and defines all acronyms, abbreviations, terms, and symbols. All requirements have objective acceptance criteria for validating.		
Software Requirements Interface Analysis	FSAR Engineering Technical Specifications System Requirements Document (SyRD) Human Factors Program Plans SRS HRS DCP&A	Verify and validate the requirements for software interfaces with hardware, user, and other systems are: • Correct – System and software interface requirements are correct • Consistent – Consistency between SRS and IRS interface descriptions • Complete – Each interface is described and includes data format and performance requirements (e.g., timing, bandwidth, accuracy, safety, and security) • Accurate – Each interface provides information with the required accuracy • Testable – There are objective acceptance criteria for interface requirements.	Requirements Task Report: Software Requirements Interface Analysis Anomaly Report	Software V&V Lead for safety systems Software Development Lead for nonsafety systems
Configuration Management Assessment	All inputs of this phase SCMP	Verify configuration management process is complete and adequate • Complete – This is a process for describing software functionality, tracking program version, and managing changes • Adequate – The configuration process is adequate for software complexity, size, integrity level, and user need.	Requirements Task Report: Configuration Management Assessment Anomaly Report	Software Development Team
Safety Analysis	See SSPP (Section 6.4.2.1)	Analyze potential system hazard caused by software as specified in the SSPP (Section 6.4.2.1)	See SSPP (Section 6.4.2.1)	Software Safety Lead

Table 12. Requirements Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures	Required Outputs	Responsibility
Risk Analysis	Schedule SRS DCP&A HRS	Identify and provide ways to reduce, mitigate, or eliminate technical and management risks	Requirements Task Report: Risk Analysis Anomaly Report	Project Manager
Phase Summary Report	All recorded V&V work completed for phase	Summarize tasks of the Requirements Phase, retained as quality records	Requirements Phase V&V Summary Report	Software V&V Lead

Table 13. Design Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Design Traceability analysis	SRS DCP&A Software Design Description (SwDD) Intra-System Communication Protocol Specification (ICPS) RTM	Trace software design elements to software requirements and vice versa, to analyze for correctness, consistency, and completeness. Ensure they are captured in the RTM. • Correctness – Validate relationship between design elements and requirements • Consistency – Ensure on consistent level of detail for specification relationship between design and requirement • Completeness – Verify all design elements are traceable to requirements and all requirements are traceable to design elements	RTM Anomaly Report	Software Development Lead (Prepare) Software V&V Lead (Review)

Table 13. Design Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Software Design Evaluation	SRS DCP&A SwDD ICPS Software Coding Conventions and Guidelines	Verify and validate design elements for correctness, consistency, completeness, accuracy, readability, and testability to ensure: • Source code satisfies the software design and complies with standards, regulations, and policies • Sequences of state, state changes, and flow of data satisfy functionality and performance requirements • Data usage and format appropriate • Appropriate coding methods and standards • Consistency in documentation of code terms/concepts and source code components • The following elements are in the design documentation within the constraints of the system: o Functionality (including but not limited to: algorithms, reporting, and logging), o Interfaces descriptions (software/hardware), o Performance criteria (including but not limited to: timing sizing, speed, capacity, precision, safety, and security), and o System/device/software control (including but not limited to initialization, transaction and state monitoring, and self-testing) • Documentation is legible, understandable, and unambiguous and defines all acronyms, abbreviations, terms, symbols, and design language. • There are objective acceptance criteria for validating software design elements and the design elements are testable to the criteria.	Design Task Report: Software Design Evaluation Anomaly Report	Software V&V Lead for safety systems Software Development Lead for nonsafety systems

Table 13. Design Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Software Design Interface Analysis	SRS DCP&A SwDD ICPS Software Coding Conventions and Guidelines System Design Documents P&IDs Procurement Specification	Verify and validate the design interfaces with hardware, user, and other systems are: • Correct –External and internal software interface design are correct • Consistent – Consistency between SwDD, ICPS, and DCP&A interface designs • Complete – Each interface is described and includes data format and performance requirements (including but not limited to: timing, bandwidth, accuracy, safety, and security) • Accurate – Each interface provides information with the required accuracy • Testable – There are objective acceptance criteria for interface design.	Design Task Report: Software Design Interface Analysis Anomaly Report	Software V&V Lead for safety systems Software Development Lead for nonsafety systems

Table 13. Design Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Unit Test Plan Generation	SRS SwDD DCP&A ICPS Software Test Plan	Plan unit V&V testing to validate software components (i.e., modules or logical groups of modules). Validate Unit Test Plan satisfies: • Traceable to software requirements and design • Consistency with software requirements and design and unit requirements • Test coverage of requirements • Feasibility of software integration and testing • Feasibility of operation and maintenance requirements Plan tracing of design requirements to and documentation for test designs/cases/procedures/results Unit Test Plan shall address compliance with design requirements, assessment of timing/sizing/accuracy, measures of software reliability and maintainability, and performance at boundaries and under stress conditions Verify Unit Test Plan conforms to format defined in each Software Test Plan.	Unit Test Plan Anomaly Report	Software Test Lead for safety systems Software Development Lead for nonsafety systems

Table 13. Design Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Integration Test Plan Generation	SRS SwDD DCP&A ICPS Software Test Plan	Plan integration V&V testing to validate software correctly implements software requirements and design as units are integrated. Validate Integration Test Plan satisfies: • Traceable to software requirements • Consistency with requirements • Test coverage of requirements • Appropriate test standards and methods • Conformance to expected results • Feasibility of software tool acceptance • Feasibility of operation and maintenance Plan tracing of requirements to and documentation for test designs/cases/procedures/results Integration Test Plan shall address compliance with requirements, assessment of timing/sizing/accuracy, measures of software reliability, and performance at boundaries and under stress conditions Verify Integration Test Plan conforms to format defined in each Software Test Plan.	Integration Test Plan Anomaly reports	Software Test Lead for safety systems Software Development Lead for nonsafety systems

Table 13. Design Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
System Validation Test (SVT) Plan Generation	FSAR Engineering Technical Specifications System Requirements Document (SyRD) Human Factors Program Plans SRS DCP&A Preliminary User Documentation Software Test Plan	Plan system validation V&V testing to validate software requirements. Validate SVT Test Plan has: • Test coverage of system requirements • Appropriateness of test methods • Conformance to expected results • Feasibility of system qualification testing • Feasibility and testability of operation and maintenance requirements • Coverage for each input, each output, each logic path, each soft connection in the Human-System Interface, each alarm point, each control room display • Ability to backup, restore, and reload the software (as applicable) Plan tracing of system requirements to and documentation for test designs/cases/procedures/results SVT Test Plan shall address compliance with system requirements as complete software end item, adequacy of user documentation, and performance at boundaries and under stress conditions Validate SVT Test Plan conforms to format defined in each Software Test Plan.	System Validation Test Plan Anomaly Report	Software Test Lead

Table 13. Design Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
SVT, PFT, and PIT Test Case Generation	System Requirements Document (SyRD) SRS DCP&A SwDD ICPS Preliminary User Documentation Software Test Plan Test Plans	Develop Test Cases for component, integration, system, and system integration testing Continue tracing requirements as specified in Test Plans Verify Test Cases comply with the test documentation specified in the Software Test Plan and criteria identified for defined test plans (above)	SVT Test Cases PFT Test Cases PIT Test Cases Anomaly Report	SVT: Software Test Lead PFT and PIT: Integration and Integration Test Team
Safety Analysis	See SSPP (Section 6.4.2.1)	Analyze potential system hazard caused by software as specified in SSPP (Section 6.4.2.1)	See SSPP (Section 6.4.2.1)	Software Safety Lead
Risk Analysis	Schedule SwDD ICPS	Identify and provide ways to reduce, mitigate, or eliminate risks	Design Task Report: Risk Analysis Anomaly Report	Project Manager
Phase Summary Report	All recorded V&V work completed for phase	Summarize tasks of the Design Phase, retained as quality records	Design Phase V&V Summary Report	Software V&V Lead

Table 14. Implementation Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Source Code Traceability analysis	SwDD ICPS Source Code RTM Procurement Specification	Trace software source code to design specifications and vice versa to analyze for correctness, consistency, and completeness. Ensure they are captured in the RTM. • Correctness – Validate relationship between design elements and source code • Consistency – Ensure on consistent level of detail for specification of relationship between design and source code • Completeness – Verify all design elements are traceable to source code units and all source code units are traceable to design elements	RTM Anomaly Report	Prepare: Software Development Lead Review: Software V&V Lead for safety systems Software Development Lead for nonsafety systems

Table 14. Implementation Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Source Code Evaluation	SwDD ICPS Software Coding Conventions & Guidelines Source Code User documentation	Verify and validate source code units for correctness, consistency, completeness, accuracy, readability, and testability to ensure: • Source code satisfies the software design and complies with standards, regulations, and policies • Sequences of state, state changes, and flow of data satisfy functionality and performance requirements • Data usage and format appropriate • Appropriate coding methods and standards • Consistency in documentation of code terms/concepts and source code components • The following elements are in the source code within the constraints of the system: o Functionality (including but not limited to: algorithms, reporting, and logging), o Interface descriptions (software/hardware), o Performance criteria (including but not limited to: timing sizing, speed, capacity, precision, safety, and security), and o System/device/software control (including but not limited to: initialization, transaction and state monitoring, and self-testing) • Documentation is legible, understandable, and unambiguous and defines all acronyms, abbreviations, terms, symbols, and design language. • There are objective acceptance criteria for validating source code units and the source code units are testable to the criteria.	Implementation Task Report: Source Code Evaluation Anomaly Report	Software V&V Lead for safety systems Software Development Lead for nonsafety systems

Table 14. Implementation Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Source Code Interface Analysis	SwDD ICPS Source code FSAR Engineering Technical Specifications System Requirements Document (SyRD) Procurement Specification Human Factors Program Plans User documentation	Verify and validate the source code interfaces with hardware, user, and other systems are: • Correct –External and internal software interface design are correct • Consistent – Source code units are consistent • Complete – Each interface is described and includes data format and performance requirements (including but not limited to: timing, bandwidth, accuracy, safety, and security) • Accurate – Each interface provides information with the required accuracy • Testable – There are objective acceptance criteria for interface code.	Implementation Task Report: Source Code Interface Analysis Anomaly Report	Software V&V Lead for safety systems Software Development Lead for nonsafety systems
Unit and Integration Test Case Generation	SRS DCP&A SwDD ICPS User documentation Software Test Plan Test Plans Procurement Specifications	Develop Test Cases for component, integration, system, and acceptance testing Continue tracing requirements as specified in Test Plans Verify Test Cases comply with the test documentation specified in each Software Test Plan and criteria identified for test plans (above)	Unit Test Cases Integration Test Cases Anomaly Reports	Software Test Lead for safety systems Software Development Lead for nonsafety systems

Table 14. Implementation Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Unit, Integration, and SVT Test Procedure Generation	SRS DCP&A SwDD ICPS User documentation Software Test Plan Test Plans and Cases Procurement Specifications	Develop Test Procedures for unit, integration, and system validation testing Continue tracing requirements as specified in Test Plans Verify Test Procedures comply with the test documentation specified in each Software Test Plan and criteria identified for test plans (above)	Unit Test Procedures Integration Test Procedures System Validation Test Procedures Anomaly Reports	Software Test Lead for safety systems Software Development Lead for nonsafety systems
Operation and Maintenance Manual Review	Operation and Maintenance Manuals Customer Instrumentation and Controls (I&C) guidance for manual content	Evaluate the operation and maintenance manuals for completeness, correctness, and consistency with respect to requirements for user interface and for any functionality that can be invoked by the user. The review of the manuals for readability and effectiveness should include representative end users who are unfamiliar with the software	Operation and Maintenance Manuals	Software V&V Lead for safety systems Software Development Lead for nonsafety systems

Table 14. Implementation Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Unit and Integration Test Execution	Source Code Executable Code SwDD ICPS SRS DCP&A Test Plans, Cases, Procedures Procurement Specifications	Perform V&V unit testing and document the results as specified in the applicable test plan Analyze test results to show: • For Unit Testing: software correctly implements the design • For Integration Testing: components integrated correctly Validate that test results trace to test criteria established by test traceability Validate that test results satisfy acceptance criteria. Document discrepancies between actual and expected test results	Unit Test Report Integration Test Report Anomaly Report	Software Test Lead for safety systems and Group 1 nonsafety systems Software Development Lead for Group 2 nonsafety systems
Software Release Report	Hardware Configuration Item List Software Configuration Item List Build Instructions	Review hardware configuration item list for completeness, including manufacturer, model, revision, settings for all jumpers, switches, etc. Review software configuration item list for completeness, including, but not limited to, the other items defined in Section 7.4.1.1 All instructions necessary to install all software tools, third party software, and third party libraries; verify correct installation of tools and libraries; and methods to rebuild the software completely	Reviewed and accepted Software Release Report	Software V&V Lead
Safety Analysis	See SSPP (Section 6.4.5)	Analyze potential system hazard caused by software as specified in SSPP (Section 6.4.5)	See SSPP (Section 6.4.5)	Software Safety Lead
Risk Analysis	Source code Schedules V&V task results	Identify and provide ways to reduce, mitigate, or eliminate risks	Implementation Task Report: Risk Analysis Anomaly Report	Project Manager

Table 14. Implementation Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Phase Summary Report	All recorded V&V work completed for phase	Summarize tasks of the Implementation Phase, retained as quality records	Implementation Phase V&V Summary Report	Software V&V Lead

Table 15. Testing and Integration Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Test Traceability analysis	Completed Test Plans, Designs, and Procedures	Analyze relationships in test plans, designs, procedures, and cases for correctness and completeness. • Correctness – Valid relationship between test plans, designs, cases, and procedures • Completeness – Verify all test procedures are traceable to test plans	Testing and Integration Task Report: Traceability Analysis Anomaly Report	Prepare: Software Development Lead Review: Software V&V Lead for safety systems Software Development Lead for nonsafety systems
PFT and PIT Procedure Generation	SwDD ICPS Source Code User documentation Software Test Plan Test Plans and Cases Procurement Specifications	Develop Test Procedures for PFT and PIT Continue tracing requirements as specified in Test Plans Verify Test Procedures comply with the test documentation specified in each Software Test Plan and criteria identified for applicable test plans (above)	PFT Test Procedures PIT Test Procedures Anomaly Reports	PFT: Software Test Lead PIT: Integration and Integration Test Team

Table 15. Testing and Integration Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
SVT Execution	Source Code Executable Code Test plan, cases, and procedures	Perform system validation testing and document the results as specified in the applicable test plan Analyze test results to show software satisfies system requirements Validate test results trace to test criteria established by test traceability and satisfy acceptance criteria. Document discrepancies between actual and expected test results	System Validation Test Report Anomaly Report	Software Test Lead for safety systems Software Development Lead for nonsafety systems
Safety Analysis	See SSPP (Sections 6.4.6 and 6.4.7)	Verify the test instrumentation does not introduce new hazards as specified in SSPP and that those hazards identified in the SSPP (Sections 6.4.6 and 6.4.7) are completely tested	See SSPP (Sections 6.4.6 and 6.4.7)	Software Safety Lead
Risk Analysis	Source code Schedules V&V task results	Identify and provide ways to reduce, mitigate, or eliminate risks	Testing and Integration Task Report: Risk Analysis Anomaly Report	Project Manager
Phase Summary Report	All recorded V&V work completed for phase	Summarize tasks of the Testing and Integration Phase, retained as quality records	Testing and Integration Phase V&V Summary Report	Software V&V Lead

Table 16. Installation Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
PFT and PIT Execution	Source Code Executable Code User documentation Test plan, cases, and procedures	Perform system validation testing and document the results as specified in the applicable test plan Analyze test results to show software satisfies system requirements Validate test results trace to test criteria established by test traceability and satisfy acceptance criteria. Document discrepancies between actual and expected test results	PFT Report PIT Report Anomaly Report	PFT: Software Test Lead PIT: Integration and Integration Test Team
Installation Configuration Audit	System O&M Manual System Training manual Installation Procedure Source and Executable Code User documentation Design documents and requirements specifications	Verify all elements of the software needed to correctly install and operate the software are present in the installation package. Validate all site parameters to verify supplied values are correct Verify coding items (CI) are correct. Verify installation procedure and safety requirements are correct, complete, and accurate.	Installation Task Report: Installation Configuration Audit Anomaly Report	Software V&V Lead, and Software QA Manager for safety systems Software Development Lead for nonsafety systems

Table 16. Installation Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Installation Checkout	User documentation System O&M Manual System Training manual Installation Procedure Source and Executable Code Design documents and requirements specifications	Conduct tests to verify the installed software corresponds with the software that underwent V&V. Verify the software operates (initialization, execution, termination) correctly. Validate the software can be removed from the system (e.g., transition from one version of software to the next) with affecting the functionality of the rest of the system Verify the requirements for continuous operation and service during the transition	Installation Task Report: Installation Checkout Anomaly Report	Software V&V Lead for safety systems Software Development Lead for nonsafety systems
Safety Analysis	See SSPP (Sections 6.4.6 and 6.4.7)	Verify the test instrumentation does not introduce new hazards as specified the SSPP (Sections 6.4.6 and 6.4.7).	See SSPP (Sections 6.4.6 and 6.4.7)	Software Safety Lead
Risk Analysis	Schedules V&V task results	Identify and provide ways to reduce, mitigate, or eliminate risks	Installation Task Report: Risk Analysis Anomaly Report	Project Manager
Final V&V Report Generation	V&V Task Summary Reports	Summarize the V&V activities, tasks, and results, including status and disposition of anomalies for the final report Provide an assessment of the overall software quality and provide recommendations	V&V Final Report	Software V&V Lead for safety and nonsafety Group 1 systems Software Development Lead for nonsafety Group 2 systems

Table 16. Installation Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Phase Summary Report	All recorded V&V work completed for phase	Summarize tasks of the Installation Phase, retained as quality records	Installation Phase V&V Summary Report	Software V&V Lead

Table 17. Operations Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Evaluation of New Constraints	SVVP New constraints	Evaluate new constraints (e.g., operational requirements, platform characteristics, operating environment) on the software requirements to verify the applicability of the SVVP.	O&M Task Report: Evaluation of New constraints Updated SRS or other design documents as necessary	Customer
Operating Procedures Evaluation	System O&M Manual User documentation Software Design Document FSAR Engineering Technical Specifications[9] System Requirements Document (SyRD)	Verify that the operating procedures are consistent with the user documentation and conform to the requirements	O&M Task Report: Proposed Change Assessment Anomaly Report	Software V&V Lead for safety systems Software Development Lead for nonsafety systems
Safety Analysis	See SSPP (Section 6.4.2.1)	Analyze potential system hazard caused by software modifications as specified in the SSPP (Section 6.4.2.1)	See SSPP (Section 6.4.2.1)	Software Safety Lead
Risk Analysis	Schedules V&V task results	Identify and provide ways to reduce, mitigate, or eliminate risks	Design Task Report: Risk Analysis Anomaly Report	Project Manager

[9]With the turnover of the constructed plant to the customer, the Engineering Technical Specifications document, which defined the relationship between the Engineering, Construction, and Procurement Team and the customer is completed. However, the document has value as an historical document, especially as it sets design direction and requirements for the plant design. This document is included solely in the role of providing useful insights into design decisions made in the plant design. This document shall not be considered as design basis information.

Table 18. Maintenance Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Proposed Change Assessment	Proposed Changes System O&M Manual System Training manual Installation Procedure Source and Executable Code Design documents and requirements specifications Software Modification Request	Assess proposed changes (e.g., modifications, enhancements, or additions) to determine the effect of the changes and to the extent to which the V&V tasks would be iterated	Software Modification Analysis	Configuration Control Board (CCB)
SVVP Revision	SVVP Approved changes	Revise the SVVP to comply with approved changes, reviewing all other plans to ensure that the software plans are still consistent	Updated SVVP	Software V&V Lead
Anomaly Evaluation	Anomaly Report	Evaluate the effect of software operation anomalies	O&M Task Report: Anomaly Evaluation	Software V&V Lead for safety systems Software Development Lead for nonsafety systems
Migration Assessment	System O&M Manual Installation Procedure Source and Executable Code Design documents and requirements specifications Approved changes	Assess whether the software requirements and implementation address: • Migration requirements and tools • Conversion of software products and data • Software archiving • Support for the prior environment • User notification	O&M Task Report: Migration Assessment	Software V&V Lead for safety systems Software Development Lead for nonsafety systems

Table 18. Maintenance Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Retirement Assessment	System O&M Manual System Training manual Installation Procedure Source and Executable Code User documentation Approved changes	For retirement, assess whether the installation package addresses: • Software support • Impact on existing systems • Software archiving • Transition to a new software product • User notification	O&M Task Report: Retirement Assessment	Software V&V Lead for safety systems Software Development Lead for nonsafety systems
Task Iteration	Approved changes System O&M Manual System Training manual Installation Procedure Source and Executable Code User documentation Design documents and requirements specifications	Perform V&V tasks as needed to ensure: • Planned changes are implemented correctly • Documentation is complete and correct • Changes do not cause unacceptable or unintended system behaviors	O&M Task Report: Task Iteration Anomaly Reports	Software V&V Lead for safety systems Software Development Lead for nonsafety systems
Safety Analysis	See SSPP (Section 6.4.2.1)	Analyze potential system hazard caused by software modifications as specified in the SSPP (Section 6.4.2.1)	See SSPP (Section 6.4.2.1)	Software Safety Lead
Risk Analysis	Schedules V&V task results	Identify and provide ways to reduce, mitigate, or eliminate risks	Design Task Report: Risk Analysis Anomaly Report	Project Manager

Table 18. Maintenance Phase V&V Activities

Required Tasks	Required Inputs	Methods and Procedures Requirements	Required Outputs	Responsibility
Regression Analysis	System O&M Manual System Training manual Installation Procedure Source and Executable Code User documentation Design documents and requirements specifications Proposed changes	Regression analysis shall be done to determine the extent of testing to be repeated if proposed changes occur to: • Assess side effects and impacts of change to software • Rerun test cases based on changes to detect errors associated with modification	O&M Task report: Regression Analysis	Software V&V Lead for safety systems Software Development Lead for nonsafety systems

Table 19. V&V Activities Assigned to Each Software Life Cycle Phase

Life Cycle Process	Development																Operation		Maintenance					
	Planning V&V	Requirements V&V				Design V&V		Implementation V&V			Test V&V			Installation/ Checkout V&V			Operation V&V		Maintenance V&V					
Software Classification	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	
V&V Activities																								
Management Review of V&V	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM
SVVP Generation	V&V	V&V	V&V																					
Concept Documentation Evaluation	V&V	V&V	V&V																					
Program Plan Evaluation	V&V	V&V	V&V																					
PFT Plan Generation	TT	TT	TT																					
PIT Plan Generation	TT	TT	TT																					
Planning Traceability Analysis	V&V	V&V	V&V																					
Safety Analysis	SST			SST			SST			SST			SST			SST			SST			SST		
Risk Analysis	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM	PM
Phase Summary Report	V&V	V&V	V&V	V&V	V&V	V&V	V&V	V&V	V&V	V&V	V&V	V&V	V&V	V&V	V&V	V&V	V&V	V&V						
Baseline Reviews	BRT	BRT	DT	BRT	BRT	DT	BRT	BRT	DT	BRT	BRT	DT	BRT	BRT	DT	BRT	BRT	DT						
Requirements Traceability Matrix				V&V	V&V	V&V																		
Software Requirements Evaluation				V&V	DT	DT																		
Software Requirements Interface Analysis				V&V	DT	DT																		
Configuration Management Assessment				DT	DT																			
Design Traceability Analysis							V&V	V&V																
Software Design Evaluation							V&V	DT	DT															
Software Design Interface Analysis							V&V	DT	DT															
Unit Test Plan Generation							STT	DT	DT															
Integration Test Plan Generation							STT	DT	DT															
System Validation Test Plan Generation							STT	STT	STT															
SVT Test Case Generation							STT	STT	STT															
PFT Test Case Generation							TT	TT	TT															
PIT Test Case Generation							TT	TT	TT															
Source Code Traceability Analysis										V&V	DT													
Source Code Evaluation										V&V	DT	DT												
Source Code Interface Analysis										V&V	DT	DT												
Unit and Integration Test Case Generation										STT	DT	DT												
Unit, Integration, and SVT Test Procedure Generation										STT	DT	DT												
Operation and Maintenance Manual Review										V&V	DT	DT												
Unit and Integration Test Execution										STT	STT	DT												
Software Release Report										V&V	DT	DT												
Test Traceability Analysis													V&V	DT										
PFT Procedure Generation													STT	STT	STT									
PIT Procedure Generation													TT	TT	TT									
SVT Execution													STT	DT	DT									
PFT Execution																STT	STT	STT						
PIT Execution																TT	TT	TT						
Installation Configuration Audit																V&V	DT							
Installation Checkout																V&V	DT	DT						
V&V Final Report Generation																V&V	V&V	DT						
Evaluation Of New Constraints																			I&C	I&C	I&C			
Operation Procedures Evaluation																			V&V	DT	DT			
Proposed Change Assessment																						CCB	CCB	CCB
SVVP Revision																						V&V	V&V	V&V

Table 19. V&V Activities Assigned to Each Software Life Cycle Phase

Life Cycle Process	Development																Operation		Maintenance					
	Planning V&V	Requirements V&V			Design V&V		Implementation V&V			Test V&V			Installation/ Checkout V&V			Operation V&V		Maintenance V&V					
Software Classification	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	G2	SR	G1	
V&V Activities																								
Anomaly Evaluation																						V&V	DT	DT
Migration Assessment																						V&V	DT	DT
Retirement Assessment																						V&V	DT	DT
Task Iteration																						V&V	DT	DT
Regression Analysis																						V&V	DT	DT

SR = Safety related
G1 = Nonsafety Group 1
G2 = Nonsafety Group 2

empty cell = process is not applied

BRT=Baseline Review Team
CCB=Configuration Control Board
CM=Configuration Management Lead
DT=Software Development Team
TT= Integration and Integration Test Team

I&C=Customer
PM=Project Manager, may be two separate PM, one responsible for design/development and responsible for V&V
SSL=Software Safety Lead
SST=Software Safety Team
V&V=V&V Lead
SIT=System Installation Test Team
STT=

4.3.11 Previously Developed or Purchased Software

The SVVP shall identify the methods used to verify and document that previously developed or purchased software is of appropriate quality for use, based on the evaluations performed in Sections 3.11.3.5 and 3.11.3.6. Methods defined in the Electric Power Research Institute (EPRI) technical report TR-106439 (**Reference** 32) shall be employed for the evaluation of previously developed, purchased, or commercial-off-the-shelf software in safety related applications.

Additional commercial grade dedication guidance is required beyond EPRI TR-106439 (**Reference** 32), which provides guidance accepted by the NRC, which shall be applied to safety grade software. The additional documentation required for commercial grade dedication of the digital content shall augment the documentation required for commercial grade dedication of the equipment that includes the digital content. The requirements of EPRI TR-107339 (**Reference** 38) and EPRI 1011710 (**Reference** 31) shall be included in these evaluations. These three EPRI technical reports shall be applied to Nonsafety Group 1 applications, unless equivalent documentation is provided concerning the acceptability of the software, which shall be reviewed and approved by the customer according to the customer's request. Evaluations of COTS and PDS software in nonsafety Group 2 should be performed in accordance with these three EPRI technical reports.

These evaluations shall be performed by the Software Development Lead and the Software Verification and Validation Lead. The Software Safety Lead shall participate in the evaluations for software to be used in systems included in the scope of the Software Safety Plan (Sections 6.1.2 and 6.3.12).

As a minimum, the approval process shall include verification that the evaluations performed in Section 3:

- Determine the interfaces to and functionality of the previously developed or purchased software.
- Identify relevant documents (e.g., product specifications, design documents, usage documents) that are available to the obtaining organization and determine their status.
- Determine the conformance of the previously developed or purchased software to published specifications.
- Identify the capabilities and limitations of the previously developed or purchased software with respect to the requirements for the system.
- Following an approved test plan, test the safety-critical features of the previously developed or purchased software independent of the system's software.
- Following an approved test plan, test the safety-critical features of the previously developed or purchased software with the system's software.
- Perform a risk assessment to determine if the use of the previously developed or purchased software will result in undertaking an unacceptable level of risk.

Only previously developed or purchased commercial-off-the-shelf software that 1) can be adequately tested, 2) presents minimal risk as documented in a formal report which shall be submitted to the customer Software Safety Lead for review and approval, or 3) remains safe in the context of its planned

use shall be used. The inability to determine if these criteria are satisfied adequately shall be sufficient justification for rejecting the use of the previously developed or purchased software.

4.4 V & V Reporting and Administrative Requirements

4.4.1 Reporting For Each System or Logical Group of Systems

The required output documents from the V&V activities are defined in Table 10 through Table 18.

A Final V&V Report shall be prepared, reviewed, and approved to summarize the V&V activities, including:

- V&V activities completed, including V&V methods used
- Any deviation from the each SVVP or from the SVVPP shall be documented and that documentation included in the final V&V report. Deviations include process, document formats, and requirements.
- Summary of results
- Compliance demonstration for each requirement
- Summary of anomalies, resolutions, and current anomaly status
- System/software configuration tested
- Recommendations

The PM shall approve the Final V&V Summary Report. The PM will submit the Final V&V Summary Report to the customer for review and approval. The Final V&V Summary Report shall confirm that the vendor performed all required reviews and testing and that no outstanding anomalies or issues remain. The Project Manager shall not release any software for plant use until the V&V activities and this summary report are completed and approved. This Final V&V Summary Report shall document that software development is complete, and that the vendor performed all work in accordance with this SPP and with the vendor's software plans, programs, procedures, and instructions. The Final V&V Summary Report shall provide references to permanent records that evaluate each noncompliance with the software life cycle plans and the resolution of each noncompliance, as well as summarizing each noncompliance and the resolution of that noncompliance.

4.4.2 Anomaly Reporting and Resolution

The SQAPP and each SQAP defines the requirements for anomaly reporting and corrective action (see Section 5.8). Each SVVP shall include these requirements.

5 Software Quality Assurance Program Plan (SQAPP)

5.1 Introduction

5.1.1 Purpose

This Software Quality Assurance Program Plan (SQAPP) describes the Toshiba's approach, management, organization, responsibilities, and methodologies used to ensure the development of software products meets the specified customer requirements .

5.1.2 Scope

This SQAPP shall be applied to the entire software life cycle for software products intended for use in nuclear safety-related system and equipment and defines the subset of requirements that shall also be applied to systems and equipment classified as nonsafety related (Groups 1 and 2). This SQAPP shall be used as a basis to generate a Software Quality Assurance Plan (SQAP) for Toshiba organizations' or Toshbia's contractors' individual systems or for logical groups of systems, and ensuring inclusion of the customer requirements in each SQAP. Each SQAP shall be prepared, reviewed, approved, and retained as a quality record.

This SQAPP shall be implemented by a Toshiba organization or a Toshiba's contractor supplying software-based equipment. Additional oversight shall be supplied for subcontractors by Toshiba or a Toshiba's contractor responsible for that subcontractor. Additional SQA oversight shall be supplied by the customer QA and customer I&C organizations with additional support as deemed necessary by the customer staff.

This SQAPP and each SQAP operate under the customer, Toshiba, or Toshiba's contractor's 10 CFR 50 Appendix B program for safety related systems, structures, and components. This SQAPP and each SQAP augment those Appendix B programs. All criteria of the Appendix B programs (including, for example, minimum quality records requirements, electronic records, and electronic signatures) are provided from the upper tier Appendix B programs and are not repeated in this SQAPP or in each SQAP.

This SQAPP and each SQAP operate under the customer, Toshiba, or Toshiba's contractor's International Organization for Standardization (ISO) 9001 program for nonsafety related systems, structures, and components. This SQAPP and each SQAP augment those ISO 9001 programs. All criteria of the ISO 9001 programs (including, for example, minimum quality records requirements, electronic records, and electronic signatures) are provided from the upper tier ISO 9001 programs and are not repeated in this SQAPP or in each SQAP.

5.1.3 [Deleted]

5.1.4 Relationship of the SQAPP to Other SPP Sections

In order to simplify the description of processes and ensure clarity in establishing roles and responsibilities and to align with Branch Technical Position 7-14 (**Reference** 3) guidance, the SPP has been split into several sections. Each section of this SPP describes the processes, roles, and responsibilities for the activities in that section. Each section either explicitly or implicitly refers to other sections as necessary to implement additional processes.

Thus, this SQAPP defines the methods to be used to assure compliance to plans, programs, and instructions during the software life cycle. This activity will oversee the operation of the activities defined in Sections 2 through 4, Sections 6 through 11, Section 13, and Appendix C of this SPP.

The project management functions defined in Section 12 are implemented in the customer plans, procedures, and instructions necessary to operate and maintain the nuclear power plants.

Thus, for each of these sections, the SQAPP provides the compliance oversight of all activities necessary to implement the life cycle activities specified by the life cycle. The SQAPP interfaces with the other sections of this SPP as follows:

- All requirements provided in Section 1 of the SPP apply to the SQAPP, including system and software organization, roles and responsibilities, classification of systems and software, and the life cycle for system and software development.

- The SPMPP, Section 2, controls and oversees the design, development, and implementation performed by the design and V&V organizations defined in accordance with the technical and process requirements of the SPP, with the oversight of those activities defined in this Section 5, Quality Assurance.

- The SDPP, Section 3, performs design and development activities. The SQAPP provides the oversight of the design organization to ensure procedure compliance, in accordance with the technical and process requirements of this section.

- The software quality assurance organization provides oversight of the SVVPP organization. The software quality assurance organization ensures that the V&V organization works in accordance with their plans, procedures, and engineering instructions. The SDPP implementation and SVVPP implementation are overseen by the software quality assurance activities performed by the quality assurance organization, in accordance with the technical and process requirements listed in Section 4, Verification and Validation.

- The software quality assurance organization provides oversight of the Software/System Safety organization to provide additional assurance that safety is maximized, and that the design, development, implementation, analysis, review, test, other V&V activities, and maintenance activities are performed in a manner such that identified safety concerns are implemented correctly as well as reviewed and tested completely. The Software/System Safety work is performed in accordance with the plans, procedures, and engineering instructions in accordance with the technical and process requirements listed in SPP Section 6, Software Safety.

- The software quality assurance organization activities coordinate with the change control and configuration management activities. The SQAPP activities ensure that change control and configuration management activities are performed in accordance with the technical and process requirements listed in Section 7, Configuration Management.

- The SQAPP activities provides oversight to the Software Integration Organization and additional assurance that the system and software integration activities are controlled and performed in accordance with the technical and process requirements listed in SPP Section 8, Software Integration.

- The software quality assurance organization activities provide oversight of the system and software testing activities and additional assurance of compliance with the plans, procedures, and instructions created for testing, in accordance with the technical and process requirements listed in Section 9, Testing.

- The software quality assurance organization activities provide oversight of the assignment of appropriately trained personnel in all phases of the software and system life cycle. The software quality assurance organization provides oversight of the creation and review of appropriate training materials and other manuals such that plant staff can be trained appropriately, in accordance with the technical and process requirements listed in Section 10, Training.

- The software quality assurance organization activities provide oversight of compliance with the requirements for other organizations to create the instructions and any required procedures for software installation at the vendor site and at the nuclear plants, such that the installations can be controlled and performed in accordance with the technical and process requirements listed in Section 11, Installation.

- Software quality assurance activities or requirements continue during plant operation to ensure that configuration management is performed appropriately. The software quality assurance organization has no involvement with day-to-day plant operation. The day-to-day operation will be controlled by the plans, procedures, and instructions provided by the customer for the normal, expected plant functions that include operation, surveillance, calibration, modifications to tunable constants and other configuration items, and troubleshooting in accordance with the technical and process requirements listed in Section 12, Operations.

- The software quality assurance organization will oversee the implementation of any changes required to close anomaly reports and implement enhancements to plant systems in a controlled manner, in accordance with the technical and process requirements listed in Section 13, Maintenance.

- The software quality assurance organization verifies compliance with the plans, procedures, and instructions required to perform software work in accordance with the technical and process requirements listed in Appendix C, Secure Development and Operational Environment, to ensure that work performed does not compromise safety, for all safety systems and for other systems where cyber security requirements apply.

5.2 Reference Documents

The reference documents for this SQAPP are listed in Section 1.8.

5.3 Management

5.3.1 Organization

A typical organization for the Software Quality Assurance activity is illustrated in Figure 1. Other organizations are acceptable as discussed in and subject to the constraints of Section 1.4.

5.3.2 Tasks

Each SQAP shall perform the following tasks throughout the software life cycle, as described in Section 1.9:

- Review and approval of software life cycle documents to ensure compliance with relevant sections of this Software Program Plan
- Oversight of all software life cycle activities to verify compliance with plans, programs, procedures, and instructions, to include both the digital and traditional aspects of commercial grade dedication
- Performance of Baseline Reviews to ensure that activities are properly performed and documented at each phase in the software life cycle
- Performance of periodic audits to ensure that all software life cycle activities performed in accordance with relevant sections of this Software Program Plan
- Controlling the corrective action program at Toshiba and Toshiba's contractors, including problem reporting, tracking, and resolution

All activities within this plan shall be planned and scheduled in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

5.3.3 Roles and Responsibilities

Roles and responsibilities for the Quality Assurance Manager, Software Quality Assurance Lead, and Baseline Review Team are defined in Section 1.4.3.

The Executive level (see Figure 1) shall provide the QA Manager and the Software Quality Assurance Lead with adequate, appropriately independent staff to perform the work assigned in a timely manner.

5.4 Documentation

The documentation governing the development, verification and validation, use, and maintenance of software are identified in the SDPP (Section 3), SVVPP (Section 4), SOPP (Section 12), and SMaintPP (Section 13), respectively. This documentation shall be reviewed during Baseline Reviews. Baseline Reviews and Baseline Review Reports are not required for programmatic documents.

Additionally, the following documentation (created as required by the Software Program Plans and the classification of each system) shall require reviews for adequacy[10] by the Verification and Validation Team, with oversight from the Software Quality Assurance Lead:

- Software Requirements Description
- Software Design Description
- Verification and Validation Plans
- Software Safety Plan
- Verification and Validation Final Report
- Operation and Maintenance Manuals
- Software Configuration Management Plan
- Commercial Grade Dedication Plans

5.5 Standards, Practices, Conventions, and Metrics

Since the applicable standards, practices, conventions, and metrics may differ depending upon software platform or classification, they shall be identified by the Project Manager during the planning phase of the software life cycle for each software project (see Section 2.4.1). The Software Quality Assurance Lead shall be responsible for collecting and analyzing metrics data for safety related and nonsafety software.

Standards for documentation, logic structure, coding, commenting, and testing along with testing practices should be considered in each SQAP.

As required by Section 1.11.10, metrics appropriate for Toshiba or a Toshiba's contractors' technology, and processes shall be applied as needed to control and measure quality for both the work products and the processes. The Software Quality Assurance Lead shall be responsible for metrics associated with this program, and shall define and implement appropriate metrics. The Software Quality Assurance Lead shall also be responsible for oversight of metrics use and application in the overall software life cycle. Metrics that are demonstrated to be of no value shall be abandoned or replaced with metrics that have value to ascertaining process compliance.

[10]From IEEE Std. 730-2002 [Reference 15]

5.6 Reviews and Audits

5.6.1 Reviews

5.6.1.1 Baseline Reviews

Baseline Reviews shall be performed as defined in Section 4.2.6.6. Baseline review activities do not require nuclear quality assurance audits.

5.6.1.2 Technical Reviews

The Software Quality Assurance staff shall participate in Technical Reviews conducted by the Software Verification and Validation Team. The Software Quality Assurance staff is not required to participate in every Technical Review. The Software Quality Assurance staff should choose appropriate technical reviews based criteria such as performance issues in the Development or Verification and Validation (V&V) organizations, corrective action reports, a sample of the reviews to verify that processes and procedures are observed and technical issues are tracked and resolved. These Technical Reviews shall be performed by a qualified individual or team of individuals to evaluate the accuracy, thoroughness, and suitability of key documents generated during the software life cycle process. Team members shall be independent, as specified in Section 1.4.1. Specific reviews and reviewers will be determined by the Project Manager during the Planning Phase and shall, at a minimum, include the following:

- Software Requirements Review
- Preliminary Design Review
- Critical Design Review
- Software Verification and Validation Plan Review

A report shall be prepared at the conclusion of each review describing the resulting follow-up actions and the disposition of the review comments.

5.6.1.3 Managerial Review

The QA Manager, with responsibility and authority delegated to the Software Quality Assurance Lead, shall be responsible for implementing each SQAP shall verify that schedule and resources are adequate to ensure the required SQAPP evaluations can be provided throughout the life cycle. The Software Quality Assurance Lead responsible for the staff implementing each SQAP shall also verify that the processes defined in their SQAP or SQAPs is effective, adequate, suitable, and sufficient, correcting and extending the SQAP as required to ensure that the SQAPP objectives are met.

5.6.2 Audits and Inspections

Audits and inspections shall be performed in accordance with each organization's QA program, and coordinated through the Software Quality Assurance Lead. The staff assigned to the audit or inspection shall exhibit the same independence from the Design and Verification and Validation staff as are required for the Software Quality Assurance Lead. Audits shall be performed as deemed necessary by

the QA Manager. While audits may be performed at the end of a life cycle phase, audits are not required at the end of each life cycle phase.

The audits shall be performed using written checklists, which shall be based on the requirements of the Software Program Plan applicable to the life cycle phase and activity being audited. An assessment of the adequacy of the criteria currently contained in each set of software plants is performed as part of the audit process. Revisions to the various software plans shall be recommended as necessary at the conclusion of each audit. Results of each audit shall be documented in Audit Reports, which shall be archived as part of the Baseline Review Report. The audit report shall:

- Describe the audit scope,
- Identify the auditors,
- Identify the persons contacted during the audit,
- Contain a summary of the audit results, and
- Describe recommendations and findings. A separate Corrective Action Report (CAR) is used for each finding.

The Audit Reports, Inspection Reports, and CARs shall be submitted to the Software Quality Assurance Lead for review. These shall be controlled in accordance with the organization's QA program. Follow-up action, including re-audit of deficient areas, shall be taken where indicated. Each CAR shall be assigned to the appropriate Lead for resolution, and completion of each CAR tracked by the Software Quality Assurance Lead, with final approval by the QA Manager.

5.6.2.1 Functional Audit

The Baseline Review Team (BRT) shall perform technical Functional Audits by reviewing the Requirements Traceability Matrix (RTM) during Baseline Reviews to ensure that the software requirements specified in the previous phase are adequately addressed in work products created or modified in the current phase. The results of these audits shall be documented in the Baseline Review Report. These audits are not required to be formal Nuclear Quality Assurance audits. Additionally, RTM and functional audits are not required for programmatic documents, except as noted in Section 1.3.

5.6.2.2 Physical Audit

The BRT shall perform technical Physical Audits during the Test Phase Baseline Review to verify the accuracy and thoroughness of the Software Build Description, to assure that sufficient detail exists to create a duplicate version of the executable code. The results of this audit shall be documented in the Test Phase Baseline Review Report. These audits are not required to be formal Nuclear Quality Assurance audits.

5.6.2.3 In-Process Audits and Inspections

The Software Quality Assurance Lead shall schedule formal Quality Assurance In-Process Audits and Inspections as necessary to evaluate and ensure that software life cycle activities are being performed in accordance with this Software Program Plan. Staff performing these audits shall be independent of the work being audited, to the extent required in Section 1.4.1. These audits are performed at the discretion

of the Software Quality Assurance Lead. These audits shall be performed, for example, when the Software Quality Assurance Lead questions procedural compliance or configuration management, or in response to significant changes in the software plans, or if the team performing the work is inexperienced. The results of this audit shall be documented in an audit report that summarizes the audit activities, results, and recommended corrective actions. These audit reports shall be archived with an appropriate Baseline Review Report.

5.7 Test

All safety-related tests that will be required on the software shall be included in the Software Verification and Validation Plan (see Section 4).

5.8 Anomaly Reporting and Corrective Actions

Processing of anomalies and corrective actions shall include evaluation of any software metrics in use, to determine if the anomaly or corrective action is indicative of a single event, or of a developing adverse trend, in accordance with the requirements of Section 1.11.10.

5.8.1 Anomaly Reporting

Anomalies and deviations found in any work product after release of that work product for review, test,or other use by someone other than the author, during any phase of the software life cycle shall be formally documented. Safety critical anomalies discovered post-release or after Commercial Grade Dedication is complete shall be evaluated to determine if notification under 10 CFR 21 is required and perform appropriate notifications. The anomaly documentation shall provide a description with sufficient detail to permit an evaluation of the causes and impact of the problem and shall include the name of the person who identified the potential problem. The SQAP shall identify the requirements and format to be used for all Anomaly Reports.

5.8.2 Corrective Action

Corrective Action Reports shall be generated, during day-to-day performance of work or during the course of a review or audit, whenever any of the following conditions are noted:

- When a deviation from the any software plan, procedures, or engineering instructions is noted or required,
- When any other conditions are noted that require corrections to previously approved work products
- When any other conditions are noted that could be considered adverse to quality,

The deviation and recommended corrective action shall be determined in conjunction with the QA Manager and shall be documented. The cause of the problem shall also be identified to the degree it can be determined.

The preparer of the Corrective Action Report shall determine whether the condition adverse to quality is considered significant based on factors such as:

- Impact on health and safety of the public,

- Impact on the software safety analyses,
- Impact on reliability, availability, or maintainability of equipment,
- Importance in meeting regulatory commitments, and the consequences of recurrence, and
- Extent to which the condition may apply to other items or activities and have greater impact.

If the condition is considered significant, a plan to prevent recurrence will be identified and consideration shall be provided to issuing a stop work order.

The Corrective Action Report shall be signed, dated, and submitted to the Software Quality Assurance Lead and QA Manager. An evaluation of the Corrective Action Report shall be performed by the Software Quality Assurance Lead and QA Manager. The QA Manager approves appropriate corrective action based on the conditions observed, including the need for immediate corrective action. The Software Quality Assurance Lead and QA Manager jointly review the determination of the significance of the condition and the plan to prevent recurrence, if applicable. The QA Manager shall review the deviation for possible notification in accordance with 10 CFR 21.

Corrective actions that necessitate changes to approved work products, including documentation, reviews, testing, or code, shall require management approval and re-planning as necessary prior to starting work on the correction. This process is managed under the Software Modification Process described in Section 5.9.

The SQAP shall identify the requirements and format to be used for all Corrective Action Reports.

5.9 Software Modification Process

The SCMPP describes the process for generating and implementing Software Modification Requests (SMR) (see Section 7.4.2). The SQA organization shall ensure inclusion of all applicable SMRs in the baseline reviews at the completion of each life cycle phase.

5.10 Tools, Techniques, and Methodologies

Tools, Techniques, and Methodologies used to support the activities defined in each Software Quality Assurance Plan shall be identified and documented in accordance with each Software Development Plan (SDP, see Section 3.11.3.4) during the planning phase of the software life cycle for each software project.

5.11 Code Control

Code Control is described in the Software Configuration Management Program Plan (see Section 7).

5.12 Media Control

Media Control is described in the Software Configuration Management Program Plan (see Section 7).

5.13 Subcontractor and Vendor Control

Toshiba organizations and Toshiba's contractors shall be evaluated to ensure that software provided will meet established project requirements. Software purchased from vendors is the responsibility of the Toshiba organization or Toshiba's contractor purchasing such software. An audit of nuclear safety related equipment suppliers or a survey of suppliers provided equipment that is commercial grade dedicated shall be performed to determine compliance with the requirements of 10 CFR 50, Appendix B and of this Software Program Plan (SPP). Appropriate audits, inspections, and surveys based on the requirements of this SPP and quality requirements similar to ISO 9001 shall be performed for Toshiba organizations, Toshiba organizations' subcontractors, and vendors supplying equipment classified as nonsafety Group 1. Appropriate surveys to this SPP shall be performed for systems and equipment classified as nonsafety Group 2.

For systems and equipment subject to Software Safety Analysis, the audits and surveys shall be performed concurrently with the requirements in Section 6.3.13 by the Toshiba organization and/or Toshiba's contractor purchasing such software.

The QA Manager is responsible for appointing an audit or survey team and for ensuring that the audit or survey is performed as necessary to qualify suppliers.

The audit team shall prepare an audit plan. The inspection team shall prepare an inspection plan. The survey team shall prepare a plan for the survey to be performed. The plan shall identify the scope, requirements, and schedule of the audit or survey. Audit and survey plans shall be reviewed and approved by the QA Manager and forwarded to the Toshiba organizations, Toshiba organizations' subcontractor, and the vendor as appropriate.

Each audit, inspection, and survey shall be performed using written checklists, which shall be based on the pertinent requirements of 10 CFR 50, Appendix B, or, for systems not classified as safety related, on ISO 9001. The audit, inspection, or survey team shall inform the Toshiba organizations, Toshiba organizations' subcontractor, and the vendor, as appropriate, of problems and issues as they are discovered.

The results of each audit, inspection, and survey shall be documented in an Audit Report, Inspection Report, or a Survey Report, which is sent to the Toshiba organizations, Toshiba organizations' subcontractor, and the vendor, as appropriate. The requirements for the Audit Report, Inspection Report, or Survey Report are described in Section 5.6.2. The requirements for the Corrective Action Reports are described in Section 5.8.2.

If corrective action is required, clear roles and responsibilities shall be documented as part of the plan for resolution and closure of the corrective action. When corrective action is completed, each audit and survey team will indicate closure by letter to the supplier. Follow-up action, including re-audit or re-survey of deficient areas, shall be taken where indicated. Completion of follow-up actions shall be verified and documented to close out each audit and survey.

The QA Manager shall maintain a file of all audit, inspection, and survey reports, relevant close out correspondence, and a list of approved Toshiba organizations, Toshiba organizations', subcontractors, and vendors.

5.14 Previously Developed or Purchased (COTS) Software

The methods used to assure the suitability of previously developed or purchased software, including commercial grade dedication, are described in the Software Verification and Validation Program Plan (see Section 4.3.11).

5.15 Records Collection, Maintenance, and Retention

Records collection, maintenance, and retention are described in the Software Configuration Management Program Plan (see Section 7).

5.16 Training

All personnel supporting Software Quality Assurance shall be trained prior to personnel performing actions for which they are not trained. Software Quality Assurance staff shall be trained in the software plans applicable to Toshiba and Toshiba's contractors. Software Quality Assurance staff shall be competent in the appropriate technical and quality activities required in the design, development, review, test, and modification of software products. Training shall be by methods identified in the SQAP. The QA Manager shall retain training and experience records for each person supporting Software Quality Assurance activities.

5.17 Risk Management

The Risk Management process is described in Software Project Management Program Plan (see Section 2).

6 Software Safety Program Plan (SSPP)

6.1 Introduction

6.1.1 Purpose

This Software Safety Program Plan (SSPP) establishes the processes and activities intended to ensure that the nuclear safety concerns for the software products classified as safety related are properly considered during the applicable software development life cycle phases. This SSPP ensures that each safety related system or logical group of systems is designed, developed, implemented, tested, reviewed, installed, operated, and maintained in a manner consistent with the defined safety analyses for each system or logical group of systems, and consistent with the overall the customer regulatory commitments. Each SSPP for safety related systems or logical groups of safety related systems shall be prepared, reviewed, approved, and retained as a quality record.

This SSPP is consistent with the software safety analysis requirements provided in Regulatory Guide (RG) 1.173 (**Reference** 9). This SSPP meets the regulatory expectations specified in Chapter 7 of NUREG 0800 (**Reference** 2).

6.1.2 Scope

Software Safety Analysis (SSA) shall be performed on all software that is classified as safety related. Software Safety Analysis is not required for software that is classified as nonsafety related.

While NUREG/CR-6101 (**Reference** 41) recommends performance of Criticality Analysis to determine which requirements are associated with nuclear safety, this plan does not perform such an analysis. For safety systems, all of the system normally installed in the customer that could affect the safety function shall be treated as safety related. Permanently attached equipment shall require safety classification. Safety classification is not required for maintenance equipment, equipment brought in for calibration and surveillance, and other equipment that is only installed when the system is not credited with performing its safety function. Such equipment does not require software safety analysis beyond the nonsafety classified equipment's possible impact on the safety system, as long as the requirements of IEEE Std. 7-4.3.2 (**Reference** 11) and IEEE Std. 603 (**Reference** 13) are met.

For a nuclear power plant, software safety analysis activities are based on work performed initially by other engineering organizations, which are not controlled by this Software Program Plan. This work includes:

- Determining which systems are safety related,
- Designing plant systems to ensure nuclear safety,
- Defining fluid and mechanical systems for safe and reliable operation,
- Determining the conditions that lead to and consequences of accidents, and
- Ensuring that safety systems are reviewed and tested appropriately.

The Software Safety Analyses are responsible for ensuring that the digital systems provided are as safe as reasonably achievable, and that there are no new and unidentified means of entering accident conditions not currently evaluated in the accident analyses in the plant's final safety analysis report (FSAR).

Diversity and defense-in-depth (D3) analyses are not included in the Software Safety Program Plan, as D3 analyses will be included in the plant final safety analysis report, eliminating the need for additional analysis of the nuclear safety impacts from common cause or common mode failures.

This SSPP shall be implemented by each Toshiba organization and Toshiba' contractor supplying software-based equipment that is classified as safety related. Additional oversight shall be supplied for subcontractors by the Toshiba organization and Toshiba' contractor responsible for that subcontractor. Additional SQA oversight shall be supplied by the customer QA and I&C organizations, with additional support as deemed necessary by the customer.

6.1.3 [Deleted]

6.1.4 Relationship of the SSPP to Other SPP Sections

In order to simplify the description of processes and ensure clarity in establishing roles and responsibilities and to align with Branch Technical Position 7-14 (**Reference** 3) guidance, the SPP has been split into several sections. Each section of this SPP describes the processes, roles, and responsibilities for the activities in that section. Each section either explicitly or implicitly refers to other sections as necessary to implement additional processes.

Thus, this SSPP defines the methods to be used to evaluate the software design, development, and implementation activities, using other portions of the activities defined in Section 2 through 5, Sections 7 through 11, Section 13, and Appendix C of this SPP. The project management functions defined in Section 12 are implemented in the customer plans, procedures, and instructions necessary to operate and maintain the nuclear power plants.

Thus, for each of the sections, the SSPP provides the technical evaluation activities necessary to maximize safety of the safety related systems, throughout the life cycle activities specified by the life cycle. The SSPP interfaces with the other sections of this SPP as follows:

- All requirements provided in Section 1 of the SPP apply to the SSPP, including system and software organization, roles and responsibilities, classification of systems and software, and the life cycle for system and software development.

- The SPMPP, Section 2, controls and oversees the design, development, and implementation performed by the design organizations defined in accordance with the technical and process requirements listed in this Section 3, Development. The operation of the Software/System Safety organization is coordinated with organizations defined in this SPP in accordance with the technical and process requirements listed in Section 2, Project Management.

- The Systems Safety organization works cooperatively with the design organization to maximize the safety built into the safety related systems. The operation of the Software/System Safety organization is coordinated with organizations defined in this SPP in accordance with the technical and process requirements listed in Section 3, Design.

- The Software/System Safety organization works cooperatively with the V&V organization to ensure the completeness of the review, test, and other Verification and Validation (V&V) activities performed by the V&V organizations in accordance with the technical and process requirements listed in Section 4, Verification and Validation.

- The Software/System Safety organization works cooperatively with the software quality assurance organization, which ensures that the SDPP and V&V organizations work in accordance with their plans, procedures, and engineering instructions.

- The Software/System Safety activities coordinate with the change control and configuration management activities. The change control and configuration management activities are performed in accordance with the technical and process requirements listed in Section 7, Configuration Management. The Software/System Safety organization ensures that proposed changes are evaluated prior to implementation, as part of the technical and process requirements listed in SPP Section 7, Change Control.

- The Software/System Safety activities ensure that performance of the system and software integration activities do not introduce safety concerns and that the integration activities address any safety concerns during the integration, in accordance with the technical and process requirements listed in SPP Section 8, Software Integration.

- The Software/System Safety activities coordinate or perform (depending on the safety classification) system and software testing activities in accordance with the technical and process requirements listed in Section 9, Testing.

- The Software/System Safety activities use appropriately trained personnel in all phases of the software and system life cycle, and coordinate with this activity to ensure that training materials and other manuals are created to ensure that plant staff can be trained, in accordance with the technical and process requirements listed in Section 10, Training.

- The Software/System Safety activities ensure that software installation at the vendor site and at the nuclear plants have appropriate documentation and procedures such that the installations can be controlled and performed in accordance with the technical and process requirements listed in Section 11, Installation.

- There are no Software/System Safety activities or requirements associated with plant operations, as these will be controlled by the plans, procedures, and instructions provided by the customer for the normal, expected plant functions that include operation, surveillance, calibration, modifications to tunable constants and other configuration items, and troubleshooting in accordance with the technical and process requirements listed in Section 12, Operations. This is possible since modifications to tunable constants and other configuration items will have been evaluated as part of the system design, and acceptable ranges that do not affect Software/System Safety are created and documented in the procedures or other plant documentation supporting such modifications.

- The Software/System Safety program will be invoked as required to implement the changes required to close anomaly reports and implement enhancements to plant systems in a controlled manner, in accordance with the technical and process requirements listed in Section 13, Maintenance.

- The Software/System Safety performs software work in accordance with the technical and process requirements listed in Appendix C, Secure Development and Operational Environment to ensure

that this activity does not compromise safety, for all safety systems and for other systems where cyber security requirements apply.

6.2 Reference Documents

The reference documents for the SSPP are listed in Section 1.8.

6.3 Software Safety Management

Toshiba organizations supplying safety systems shall define a Software Safety Lead. Each Software Safety Lead shall ensure that a Software Safety Plan (SSP) is prepared for each system or logical group of systems for which they are responsible, during the project planning stage.

Each Software Safety Lead shall ensure that their subcontractors are aware of plant requirements and requirements for their systems, including the requirements derived from the final safety analysis report (FSAR).

Each Software Safety Lead shall be responsible for ensuring compliance of each of their subcontractor's Software Safety Plans to the both the SSPP and the Toshiba oragnization's Software Safety Plan.

Each Software Safety Plan shall describe the following:

- Organization,
- Schedule,
- Resources,
- Responsibilities,
- Activities,
- Tools,
- Techniques, and
- Methodologies used in the development of safety-related software.

6.3.1 Organization and Responsibilities

Each SSP shall document the software safety activities to be performed on the software project. Each SSP shall include the following activities:

- Prepare the Software Safety Analysis Plan during each Planning Phase, each of which shall be retained as a quality record
- Allocate resources to implement the Plan

- Ensure that the Secure Development and Operational Environment (SDOE) implementation is appropriate, and that the SDOE implementation does not adversely affect system safety, reliability, availability, maintainability, or the ability to test, calibrate or perform surveillance on the system
- Ensure that team members are cleared for access, as necessary, or coordinate with the appropriately cleared Cyber Security Team who already has access to cyber security information, which may require clearance for safeguards or Official Use Only documents and information
- Coordinate activities within each Software Safety Analysis Plan with other software development functions including development, system safety, software quality assurance, software reliability, software configuration management, verification and validation (V&V), and software testing, system testing, and plant testing
- Coordinate software safety tasks within the overall context of the customer software system safety program
- Coordinate technical issues related to software safety with other components of software systems development and support organization, with Toshiba and with the customer, as necessary
- Ensure adequate records are maintained to document software safety activities
- Participate in audits of the software safety plan implementation
- Ensure personnel assigned to software safety activities have appropriate training in the methods, tools, and techniques used to accomplish software safety tasks.

Each Software Safety Plan shall define the how software safety activities interface with other elements of the software development plan in use for the system or systems, including:

- Software safety activities relative to overall organization of the software development effort,
- Relationships between software safety and system safety program tasks.

The Software Safety Lead shall assign individuals responsible for activities and tasks in each Software Safety Plan. Most activities will be performed by software development personnel and reviewed by software V&V personnel. Each plan shall clearly define which group shall be performing which activities, and the requirements for independence among the participating individuals and groups.

6.3.2 Resources

Each SSP shall specify resource allocation requirements for safety tasks within the plan and monitoring of resources during the software safety program. Resources shall include financial, schedule, safety personnel, personnel involved in other software life cycle activities, computers, equipment, and tools.

6.3.3 Schedule

All activities within this plan shall be planned and scheduled in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

6.3.4 Qualifications and Training

Each SSP shall ensure that qualification and training for staff implementing the SSP is performed, documented, and maintained. Each SSP shall specify personnel qualifications for performance of the following tasks:

- Definition of Safety Requirements
- Design and implementation of safety-critical portions of the system
- Performance of software safety analysis tasks
- Testing of safety-critical features
- Auditing of software safety plan implementation
- Performance of process certification

Each SSP shall define on-going training requirements and methods of accomplishing training objectives for personnel with software safety-related responsibilities.

6.3.5 Software Life Cycle

The software safety process shall be integrated into the software life cycle process defined in the each Software Development Plan, which shall be written to comply with the Software Development Program Plan (Section 3). Each SSP shall define the phase of the software life cycle model in which specific tasks and activities are to be performed, and clearly define appropriate reviews and approvals needed for each activity.

6.3.6 Documentation Requirements

Each SSP shall be prepared by the Software Safety Lead. Contents of each SSP shall be reviewed by the Software Development Task Lead and Software V&V Task Lead. Each SSP shall be approved by the Project Manager, and shall be reviewed and approved by the customer Instrumentation and Controls (I&C) Manager. Each SSP shall be under configuration control. Changes to each SSP shall be evaluated by the Software Safety Lead, Software Development Task Lead, and Software V&V Task Lead. The purpose of this evaluation is to review the impact that changes to the SSP may have on other software or system documents. Revisions shall also be reviewed and approved by the customer.

Software Safety reports may be combined with other reports or be prepared as stand-alone reports. Each SSP shall define the approach to be used for the project. In either case, the Software Safety report content shall, as a minimum, satisfy the following requirements for the software to which it is applied:

- *Software project management.* Software Safety Program activities shall be defined relative to the life cycle model for the software project. Implementation, integration, and management of these software safety activities shall be clearly defined in each Software Project Management Plan.
- *Software configuration management.* Information regarding the configuration management of software modules and documents shall be prepared (see Section 6.3.8).

- *Software quality assurance.* Information regarding the quality assurance of software modules and documents shall be prepared (see Section 6.3.9).

- *Software safety requirements.* Information regarding specification of software safety requirements to avoid or control system hazards shall be prepared and documented in the Software Requirements Specification (SRS).

- *Software safety design.* Descriptions of the software design elements that satisfy the software safety requirements shall be prepared and documented in the Software Design Description (SwDD).

- *Software development methodology, standards, practices, metrics, and conventions.* Approved and controlled practices that are essential to satisfy system and software safety objectives and requirements shall be specified. These elements shall be defined in the Software Coding Conventions and Guidelines.

- *Test documentation.* Software safety-related testing documentation including test planning, test design, test cases, test procedures, and test reports shall be prepared as defined in the Software Test Program Plan, which includes system tests.

- *Software Verification and Validation.* Information regarding how software safety will be verified and validated is defined in the Software Verification and Validation (V&V) Program Plan. The software safety analyses are specified in the SSP. Requirements Traceability Matrices (RTMs) shall be used to ensure traceability of software safety requirements to the software requirement specifications, software design, source code, and software safety test cases.

- *Reporting safety verification and validation.* Information documenting the result of software Safety-Related Verification and Validation activities is specified in Section 4.

- *Software user documentation.* Information necessary for the safe installation, use, maintenance, and retirement of the system or logical group of systems shall be prepared as defined the requirements for Operational and Maintenance Manuals, which shall be prepared and provided by the customer.

- Reports on the results of software safety requirements analysis, Section 6.4.3.

- Reports on the results of software safety design analysis, Section 6.4.4.

- Reports on the results of software safety code analysis, Section 6.4.5.

- Reports on the results of software safety test analysis, Section 6.4.6.

- Reports on the results of software safety installation analysis, Section 6.4.7.

- Reports on the results of software safety change analysis, Section 6.4.8.

6.3.7 Software Safety Program Records

The Software Safety Lead shall ensure that following software safety program records are prepared, maintained current, and retained. These records shall be maintained in the Configuration Management system documented in Section 7. These records shall include:

- Reports on the results of analyses, including V&V, performed on requirements, design, code, test, and changes
- Information on suspected or confirmed safety problems in the prerelease or installed system shall be tracked in the Corrective Action and Problem Reporting System, as defined in the Software Quality Assurance Program Plan (SQAPP), Section 5.8
- Reports containing the results of audits performed on software safety program tasks shall be retained, and issues uncovered during those audits shall be tracked and resolved based on the Corrective Action and Problem Reporting System
- Reports on the results of safety tests conducted on all or any part of the entire system shall be retained, and issues uncovered during those audits shall be tracked and resolved based on the Corrective Action and Problem Reporting System
- A record of training provided to software safety program personnel, to include the results of training and any personnel certifications from the training

These records shall be maintained and retained in accordance with the Software Configuration Management Program Plan (see Section 7.3.2).

The Software Safety Lead shall prepare a summary software safety report for each system, or logical group of systems. This summary report shall provide evidence that the software safety program has been properly carried out during every phase of the software life cycle.

Hazards for each system or logical group of systems identified during software safety activities shall be tracked throughout the software life cycle to ensure that their status is tracked through retirement. This requires that all individual hazards shall be provided to the customer in an electronic database, which shall include the hazards resolved prior to shipment to the customer as well as clear identification of those hazards which are not completely resolved. This data shall be provided to the customer as part of the safety analysis reporting provided in Section 6.4. The software safety plan shall specify the following:

- The traceability requirements to be met by the hazards tracking system
- The hazards tracking system to be used
- The criteria to determine the applicability of the hazards tracking system, including the hazards to be tracked

6.3.8 Software Configuration Management Activities

Software configuration management shall be in force during all phases of the software life cycle and shall be accomplished in accordance with the Software Configuration Management Plan (see Section 7).

6.3.9 Software Quality Assurance Activities

Software Quality Assurance activities, as described in the SQAPP (see Section 5), shall assure proper performance of software safety program activities.

6.3.10 Software Verification and Validation Activities

Software Verification and Validation activities are specified in the SVVPP (see Section 4). To comply with this SSP, verification and validation activities shall ensure that:

- All system safety requirements have been satisfied by the life cycle phases
- No additional hazards have been introduced by the work done during the life cycle activity

6.3.11 Tool Support and Approval

Software tools used in Toshiba and Toshiba's contractors software life cycle for systems included in the scope of each SSP shall be evaluated for suitability as specified in each Software Development Plan (SDP, see Section 3.11.3.4). This evaluation shall include evaluations of the potential adverse effects of tools on software safety. Configuration control of software tools is managed in accordance with the requirements of the SCMP (see Section 7.3.1). To lessen the possibility of inadvertent introduction of software hazards by project tools, the following areas shall be addressed:

- Tool approval for use on the project
- Installation of upgrades to previously approved tools
- Withdrawal of a previously approved tool
- Identification of limitations that may be imposed on tool use

6.3.12 Previously Developed or Purchased (COTS) Software

The Software Safety Lead shall participate in developing criteria for approving previously developed or purchased software for use in systems to which the SSPP applies, along with the Software Development Lead and the Software Verification and Validation Lead (see Section 4.3.11). The Software Safety Lead shall evaluate the commercial grade dedication to ensure that system and software safety is considered and incorporated as required.

This same methodology shall be applied to software safety change analysis to all previously developed or purchased software.

6.3.13 Subcontract Management

Subcontractors for safety-critical software (safety-related software or software to be dedicated as safety-related software) shall be managed in accordance with the requirements of Toshiba and Toshiba's contractors' set of software plans for the applicable system or logical group of systems and this SSPP. Any software activities performed by a subcontractor will either be performed in accordance with Toshiba's plans or an alternative plan, reviewed and approved by Toshiba, as well as being reviewed

and approved by the customer according to the customer's request, before any work on the associated activity.

The inspections performed by the Software Safety Team should be coordinated with the audits and surveys performed by the Software Quality Assurance organization (see Section 5.13).

Toshiba organization's Software Safety Lead shall be responsible for ensuring that hazards identified by the subcontractor are communicated to the Software Safety Lead. If issues span to work done by a Toshiba's contractor, then both Toshiba and Toshiba's contractor Software Safety Leads shall be involved in the resolution of the identified hazard. Such issues shall be communicated to the customer for information only.

6.3.14 Process Certification

Baseline reviews shall be performed to certify that the software safety work products developed in each phase are acceptable, complete, and performed in accordance with each SSP and each SQAP, as described in the SQAPP (see Section 5.6.1.1).

6.4 Software Safety Analyses

Software safety analysis activities shall be performed to identify hazards associated with system operation; estimate and evaluate the associated risks; eliminate the risks, if possible, or at least mitigate the risks; ensure that risks and hazards are monitored and tracked throughout the software life cycle; ensure that the conditions leading to those risks are reviewed and tested; ensure that appropriate metrics are maintained, evaluated, and used throughout the software life cycle (see Section 1.11.10); and monitor the risk control measures.

Toshiba and Toshiba's contractors shall be responsible for software safety analysis activities as defined in Section 6.4, with coordination through the customer.

Toshiba and Toshiba's contractors Software Safety Lead will coordinate and support the customer for software safety analysis activities, as necessary.

The process includes the following activities:

- Analysis Preparation;
- Requirements analysis;
- Design analysis;
- Code analysis;
- Test analysis;
- Installation Analysis; and
- Change analysis.

The Software Safety Lead shall be responsible for planning specific software safety activities throughout the development process, and identifying these in the Software Safety Plan. The results of software safety analysis activities shall be recorded in the software safety project file for that system or logical group of systems. Each of these files shall be provided to the customer as quality records with delivery of the system or logical group of systems.

During the life cycle phases described in Figure 3 (page 93) through Figure 9 (page 99), several different software safety analyses shall be conducted, as described below.

6.4.1 Preparatory Analyses

Toshiba and Toshiba's contractor's Software Safety Team shall perform a safety assessment on the design inputs provided in Table 1 (page 57). The information in these documents shall be assessed to determine if sufficient information is provided to perform the software safety analysis required for each safety related system. This shall include an evaluation of the functional requirements for the system as well as verification that the hazards identified in the document cover the hazards inherent in the system design.

Toshiba and Toshiba's contractor's Software Safety Team shall perform a safety assessment on system design documentation to ensure that results from the human factors program are addressed in the system design documentation, with the intent of minimizing operator errors.

Toshiba and Toshiba's contractor's Software Safety Team shall review the Software Plans written for each system and ensure that the plans adequately address the software safety aspects from the Software Program Plan.

Toshiba and Toshiba's contractor's Software Safety Team shall review the Platform Factory Test (PFT) Plan and Platforms Integration Test (PIT) Plan to determine if the plans adequately address the system hazards identified, including adequate demonstration of hazards resolution and mitigation.

6.4.2 Software Safety Requirements Analysis Preparation

6.4.2.1 Preliminary Hazard Analysis

A Preliminary Hazard Analysis (PHA) shall be prepared by the Software Safety Lead. The Safety Analysis Reports are the primary outputs of the Software Safety Analysis process, including this PHA. The PHA shall contain the following information:

- Hazardous systems states
- Sequences of actions that can cause the system or plant to enter a hazardous state, including actions from nonsafety systems, which could be a result of component failures, software failures, and use errors
- Sequences of actions intended to return the system or plant, possibly by actions within another system, from a hazardous state to a non-hazardous state
- Actions intended to mitigate the consequence of accidents

The Software Safety Analysis Team shall update the Preliminary Hazard Analysis (PHA) throughout each software life cycle. As updates occur, the PHA then evolves into the Final Hazards Analysis.

6.4.2.2 System Architecture Description

The plant-level System Architecture Description (SAD) exists in the plant final safety analysis report (FSAR), as documented in Section 3.10.3.3. The FSAR provides a high-level system design and describes the functions performed by the Distributed Control and Information System (DCIS).

As documented in SSP Section 3.10.3.3, Toshiba organizations' and Toshiba's contractors' Software Safety Lead or Leads shall work cooperatively with the Software Development Team to generate a SAD specific to their scope of supply. If the SAD interfaces with another Toshiba organization's or Toshiba contractor's scope of supply, their Software Safety Leads shall work together to define the scope, roles, and responsibilities for Software Safety Analysis across the Toshiba organizations and Toshiba contractors. The SSA documents created by each of the Toshiba organizations and Toshiba's contractors shall define the scope, roles, and responsibilities for each party, and shall clearly indicate the methods used to perform software safety analysis across the scope of both Toshiba organizations' and Toshiba's contractors.

Each Toshiba organizations' and Toshiba's contractors' Software Safety Lead shall review the SAD specific to their scope of supply and shall provide any additional text or diagrams necessary for the Software Safety Analysis of the complete Distributed Control and Information System (DCIS).

Each SAD shall be provided to the customer Software Safety Lead for review and approval according to the customer's request. The customer Software Safety Lead shall be responsible for ensuring that all functions described in the plant-level FSAR for the DCIS architecture are covered appropriately in each Toshiba organizations' and Toshiba's contractors' analysis.

6.4.2.3 Software Interfaces Document

As documented in SSP Section 3.10.3.4, Toshiba organizations' and Toshiba's contractors' Software Safety Lead shall work cooperatively with the Software Development Team to generate a Software Interfaces Document (SID) specific to their scope of supply. If the SID interfaces with Toshiba organization's or Toshiba contractor's scope of supply, all involved Software Safety Leads shall work together to define the scope, roles, and responsibilities for Software Safety Analysis across the Toshiba organizations' and Toshiba contractors' boundaries. The SSA documents created by each of Toshiba organizations and Toshiba contractors shall define the scope, roles, and responsibilities for each party, and shall clearly indicate the methods used to perform software safety analysis across the scope of both Toshiba organizations' and Toshiba's contractors.

Toshiba organizations' and Toshiba's contractors' Software Safety Leads shall review the SID specific to their scope of supply and shall provide any additional text or diagrams necessary for the Software Safety Analysis of the system.

All SIDs shall be provided to the customer Software Safety Lead for review and approval. The customer Software Safety Lead shall be responsible for ensuring that the interfaces provided support all functions described in the plant FSAR for the DCIS architecture, and that the interfaces support operation of the plant, including propagation delays between systems and timeliness of data presented to the control room operators.

6.4.3 Software Safety Requirements Analysis

The Software Safety Requirements Analysis shall be performed as part of the software requirements phase of development. This analysis shall determine that all safety requirements assigned to software have been captured in the Software Requirements Specification. This analysis shall also verify that the safety requirements are written as clear, concise, unambiguous, testable, understandable statements.

6.4.4 Software Safety Design Analysis

6.4.4.1 Resolution and Mitigation for Safety-Critical Functions

During the design phase of the project, a safety assessment shall be performed on the Preliminary Hazards Analysis to confirm that potential hazards are adequately resolved to provide adequate safety levels. The mitigation features associated with the achieving adequate safety for a particular hazard shall be added to the information contained in Preliminary Hazards Analysis, which then becomes the Final Hazards Analysis. Additionally, the mitigation features shall be added to the appropriate requirements specification and to the traceability analyses, to ensure these safety requirements are traceable throughout the software development process.

6.4.4.2 Analysis Techniques for Safety-Critical Functions

Three sets of information are essential to a software and systems safety analysis. They are the functional requirements, the hazards (or failure effects), and the causes. The quality of the software safety analysis is largely determined by the completeness of these sets of information. The quality is also dependent on the quality and completeness of the hardware analyses performed to support the software safety analyses. The functional requirements shall be an input to the analysis; the hazards and causes shall be outputs.

- The functional requirements establish what the system or equipment shall do. Each key function shall be carefully examined. As necessary, functions should be subdivided to examine the specific components that support the function.

- The hazards are the undesirable effects or consequences of a system's failure to meet its functional requirements. Built-in safety features, alarms, or procedural controls, shall be considered in assessing the hazard.

- The causes shall be the different ways that the hazards might occur.

The Software Safety Lead is responsible for defining the scope and methods used in the risk analysis and documenting these methods in each SSP. This decision shall be made with consideration of the complexity and safety-critical elements of the system. This decision shall consider that the use of both top-down and bottom-up analyses methods tends to identify faults and failures that either method by itself may fail to identify. Thus, this SSPP strongly encourages application of both top-down and bottom-up analyses for safety-critical applications.

Several methods exist for identifying hazards and causes in a systematic manner, including fault tree analysis (FTA) and failure modes and effects analysis (FMEA).

Fault Tree Analysis

A top-down method such as fault tree analysis looks at the system functional requirements and the systems, components, user actions, etc., needed to support those functions. The first step shall be to review available information to identify the functional requirements. Then, the potential hazards that could adversely affect the required functions shall be identified. Finally, the potential causes of identified hazards, such as component failures, shall be identified. This top-down approach is useful in the early stages of risk analysis and when comparing several design alternatives.

Failure Modes and Effects Analysis

Working only from the top down using fault tree analysis can be difficult, as seeing fault relationships and the complete set of potential faults from the top can be hard. When detailed system or equipment designs exist, a bottom-up method such as FMEA shall be used. The FMEA method works by starting with identified failure modes, finding causes and the likelihood of the failure, and finally determining the effects of the failure.

Both methods should be used together and results from both should be identified in a single hazards analysis table. The benefit of using both methods early in the design phase is that with the top-down approach, important hazards can be identified at a high level before the details of a design are available. At the same time, using the bottom-up method, insights into system design and operation can be obtained that will improve the understanding of the inter-relationships of components.

6.4.4.3 Method to Classify Design Elements

During the design process, a hazard analysis shall be performed to identify risks requiring additional mitigation and to evaluate the effectiveness of such mitigations. This analysis is performed using the Preliminary Hazards Analysis document as an input. Each hazard is reviewed along with its associated risk. Design controls that mitigate the risk and reduce it to negligible levels shall be identified for all non-negligible risks.

The Final Safety Analysis Report shall be prepared by each Toshiba organizations' and Toshiba's contractors' Software Safety Lead. The Final Safety Analysis Report shall document the result of this evaluation and identify design controls that shall be implemented in the design phase of the project. The risk analysis shall be used by the developers and reviewers of the design to ensure that all necessary steps have been taken to ensure that the software meets acceptable levels of safety.

The customer Software Safety Lead shall integrate the individual analyses into an analysis for the entire nuclear power plant. The customer Software Safety Lead shall evaluate the integrated analysis and determine if the individual analyses adequately cover all requirements and resolve all hazards.

The Final Safety Analysis shall:

- Document the intended use of the system and any reasonably foreseeable misuse.
- Identify and document those qualitative and quantitative characteristics of the system or logical group of systems that could affect safety, with defined limits as appropriate.
- Identify and document known and foreseeable hazards associated with the system in both normal and fault conditions, ensuring that the hazards are reflected in the FSAR.

- Estimate and document the risk(s) for each hazardous situation.
- Identify risk controls that mitigate risks.
- Evaluate acceptability of residual risks.

During design verification and/or validation, the Final Safety Analysis document shall be reviewed and updated as necessary to evaluate the effectiveness of risk mitigations that were incorporated in the design and to establish the level of residual risk.

6.4.4.4 Analyses Performed on Each Design Element

The Hazards Analysis shall provide summary information regarding postulated failure modes and the resulting hazards for the system. The Software Architecture Lead from the Software Development organization shall decompose the system software into software design elements in the Software (or System) Architectural Description (SAD), such that individual software features and hardware features can be traced back to system requirements (see Sections 3.10.3.3, 3.10.3.4, and 6.4.2.3). To accomplish this, the relationship between the system hazards and the software design elements shall be established and documented. This relationship between system hazards and design elements shall be determined by decomposing the software for a system into software items and software units. The SAD shall be augmented as required by the Software Safety Lead. The SAD provides an overview of the software and hardware for the system, including the description of redundancy and separation features requirements for the software within a system. The use of the term software unit is sufficiently generic, meaning that a software unit can comprise many software design elements, or a software unit can equal one software design element. This provides means to summarize the decomposition of the software system into software items and units and elements. Typically, a software item is a module level entity such as an executable program.

Once the software system is decomposed through appropriate documentation in the SAD, the software system shall be evaluated to determine design compliance with the system safety requirements. This shall be accomplished by reviewing the software requirements specification against the software units defined in the SAD to ensure that each requirement in the SRS has a corresponding software unit in the Software Architecture Description and that each requirement is adequately addressed in the Software Design Description.

Software safety design analysis shall include a review of required testing for integration of software subsystems (software items) and systems to ensure that the software shall be capable of meeting the system safety goals. Documentation of this review shall be accomplished in the test reviews for software unit, software item, and software system level tests.

Additional analyses on the software design shall be performed using the following analysis methods:

- Functional Analysis
- Logic Analysis
- Data Analysis and Structure
- Internal Interface Analysis
- Constraint Analysis

- Non-Critical Code Analysis
- Software Element Analysis
- Timing and Sizing Analysis
- Test Procedure Evaluation
- Reliability and Availability Assessments

The Software Safety Lead shall provide project specific guidance for these analyses in each SSP.

The customer Software Safety Lead will integrate the individual analyses into an analysis for the entire nuclear power plant. The customer Software Safety Lead will evaluate the integrated analysis and determine if the individual analyses adequately cover all requirements and resolve all hazards.

6.4.5 Software Safety Code Analysis

Systems classified as safety systems shall be subject to 100% code inspection. Code inspection should be performed on systems classified as nonsafety Group 1. Code inspections shall be performed by qualified personnel and shall use the Toshiba organizations' or Toshiba contractor's coding standards and guidelines. The following code analyses shall be performed to verify that the source code implementation is traceable back to the Software Design Description (SwDD):

- Equations, algorithms, and control logic
- Program break points and return points
- Data structure and usage
- Timing Constraints
- Unit Level testing for correct execution of elements
- Interface review for compatibility between modules
- Software operation within requirement constraints
- Standard software practices, such as code size limits, code initialization, etc.)

Any tools used in the performance of code inspection shall be documented.

The code inspection reports shall document the results of the code inspections and include recommendation for code and/or design changes. In addition, detailed test requirements should be provided to ensure that adequate test coverage is provided by the test procedures (included in the Verification and Validation Plan) for the software.

The customer Software Safety Lead shall review and approve the individual analyses. The customer Software Safety Lead shall evaluate set of individual analyses, determine if the individual analyses adequately cover all requirements, and resolve all hazards.

6.4.6 Software Safety Integration and Validation Test Analyses

The Software V&V Lead shall have overall responsibility for software testing. See Sections 4 and 9 for the Software Verification and Validation Program Plan (SVVPP) and Software Test Program Plan (STPP), respectively, for details. The Software Safety Lead shall be responsible for the software safety analysis of testing for each system or logical group of systems, for all test scope identified in the Software Test Program Plan (see Section 9). The Software Safety Lead shall ensure that these analyses are performed for each Integration and Validation Test for each system or logical group of systems.

The Software Safety Test Analysis shall be performed by each Toshiba organization's and Toshiba contractors' Software Safety Lead. The test analysis performed by the software safety team shall ensure that requirements added for software safety have been implemented and that testing has proven that the implementation has successfully maintained required levels of system safety. These analyses shall be performed for the Integration Tests as well as for the System Validation Tests on each system or logical group of systems.

The customer Software Safety Lead shall review and approve the individual analyses. The customer Software Safety Lead shall evaluate set of individual analyses, determine if the individual analyses adequately cover all requirements, and resolve or at least minimize all hazards.

6.4.7 Software Safety Installation Analysis

The results of analysis within each SSP shall define the installation, commissioning, Design Installation Verification (DIV), and pre-operational testing requirements, to be demonstrated during these tests, to demonstrate features deemed important by the software safety analyses. Each Software Safety Lead shall be responsible for ensuring that appropriate testing is provided in installation, commissioning, and pre-operational testing.

The customer Software Safety Lead will be responsible for ensuring that the individual analyses performed by Toshiba and Toshiba's contractors' Software Safety Leads, individually and as an aggregate group, provide sufficient software safety analysis and demonstration of the Installation, Commissioning, and Pre-Operational activities.

6.4.8 Software Safety Change Analysis

The change management process shall be performed as defined in Section 7.4.2, Anomaly Reporting, Corrective Action, and Change Control. The Software Safety Lead shall review changes to software throughout the development life cycle to ensure that changes do not affect system safety.

After the system or logical group of systems has been transferred to the customer control, the customer Software Safety Lead shall become solely responsible. This responsibility includes ensuring that changes to any system's software or hardware do not affect the software safety analyses, and ensuring that the software safety analyses are re-performed, updated, and maintained throughout the life of the plant. Changes made by the customer shall be performed under the same SSP under which the system or logical group of systems was developed, with any required revisions to be reviewed and approved by the customer Software Safety Lead, prior to work being performed.

6.5 Post Development

The post development and installation activities will be performed to ensure the continued safety of the software system as it is integrated at the Toshiba or Toshiba's contractor's site, integrated with other systems for Platforms Integration Test (PIT), and sent to the customer for installation and test, until the system is retired. These responsibilities shall include those documented in Section 6.4.8.

6.5.1 Training

Training is described in the Software Training Plan (STrngPP) (see Section 10).

6.5.2 Deployment

6.5.2.1 Installation at the customer

The customer Software Safety Lead must be responsible for verifying that the Installation, Construction, and DIV Testing includes appropriate demonstration of the identified risks and hazards.

The Software Program Plan (SSP) shall define the appropriate software safety activities to verify that software safety is demonstrated appropriately for the identified risks and hazards, during those installation and test activities defined in Table 6.

Installation of the software shall be described in the Software Installation Plan (Section 8).

This SPP is written with the expectation that, in keeping with normal nuclear plant practices, installation and commissioning activities at the customer's site shall be described in a separate program plan being written by Toshiba or another contractor and approved by the customer. Testing scope within this SPP may also be described in separate Program Plans, such as DIV, and are further described in Section 9.3.

6.5.2.2 Startup and Transition at Customer

Prior to starting up the newly installed software product, the discrepancy log shall be reviewed and evaluated, DIV tests shall be conducted to demonstrate that the installed software product operates as intended in the plant environment; and the required set points (e.g., trip and alarm) will be tested. The Pre-Operational Test shall be conducted in accordance with test plans and procedures reviewed and approved by the customer.

The customer Software Safety Lead shall be responsible for verifying that the DIV and Pre-Operational Testing includes appropriate demonstration of the identified risks and hazards.

The scope of DIV and Pre-Operational Testing needs to be defined in a separate program plan. That program plan shall identify the scope, roles, and responsibilities of Toshiba and the customer staff. The testing shall be supported by qualified engineers who, as a group, are knowledgeable in the installed software product and plant operation, including software safety analysis.

The Startup Procedure shall address the requirements for safely starting the new systems and, if an existing system is to be replaced, for making a safe transition from the existing system to the replacement system. At a minimum, the following shall be addressed:

- Fallback modes for the new system

- Startup of backup components and subsystems
- Startup of the new system
- Parallel operation with backups, if possible
- Parallel operation of the old system and the new system, if possible
- Subsystem vs. full system operation
- Switchover to full system operation
- Validation of results from the new system
- Cross validation of results between the old system and the new system
- Fallback in the case of failure of the new system, including fallback to an old system if one exists

6.5.2.3 Operations Support

System Operation and Maintenance (O&M) Manuals shall be supplied for all systems or logical groups of systems in Toshiba or Toshiba's contractor's scope. Requirements for content of the System O&M Manual and User Interface Specification shall be prepared and provided by the customer.

6.5.2.4 Monitoring

The customer shall be responsible for monitoring the operation of the software systems. Safety concerns that are detected during operation shall be documented and reported in accordance with the plant's corrective action and problem reporting procedures. Since historical data is often useful in troubleshooting, Toshiba and each Toshiba's contractors shall supply their corrective action and problem reporting databases electronically to the customer for use in long-term support activities.

If any individual or company identifies a condition that could have potential safety implications for safety after shipment of their system to the customer, Toshiba, Toshiba's contractors, or Toshiba's vendors shall notify the customer and shall follow the directions and requirements of 10 CFR 21 for Notification of the USNRC, as necessary.

6.5.2.5 Maintenance

Methods and processes for software maintenance are specified in the System O&M Manual. The System O&M Manual is described in documentation to be provided by the customer. The processes and requirements of this SSPP, as implemented in each SSP associated with an individual system or logical group of systems, shall be followed for any software maintenance that could affect the operation of safety systems.

6.5.2.6 Retirement and Notification

Retirement and notification shall be performed as necessary to support design and installation of a new system and evaluations of the effects of system replacement on the surrounding systems, which shall be performed in accordance with the Maintenance processes defined in Section 6.5.2.5.

6.6 Plan Approval

This SSPP shall be reviewed and approved, and accepted by the customer, as part of the overall Software Program Plan, as stated in Section 1 of this SPP. Each SSP written for a system or logical group of systems shall also be reviewed and approved by the customer.

6.7 Software Safety Analysis Reporting

A Software Safety Analysis Report shall be completed and provided to the customer for each safety related system or logical group of systems. Each report shall be reviewed and approved by the customer Software Safety Lead prior to acceptance of Platform Factory Test (PFT) and Platforms Integration Test (PIT), and shall be updated when the SSP for each system or logical group of systems is completed.

Each Software Safety Analysis Report for a system or logical group of systems shall include the following information:

- Name, Description, and Version of the Software Evaluated
- System
- Software Classification
- Purpose and Scope
- **Reference** Inputs
- Software Safety Analysis Body of Report
- Anomalies Noted
- Conclusion
- Responsible Engineer
- Approving Authority

The report shall be placed under configuration control as described in the Software Configuration Management Plan, see Section 7. All of these reports shall be provided to the customer with system turnover.

7 Software Configuration Management Program Plan (SCMPP)

7.1 Introduction

This plan provides a framework for Software Configuration Management activities for the development or configuration of digital systems and equipment for the customer. This plan applies to the entire life cycle of such projects, including those developed or maintained by plant staff, or purchased from or maintained by external vendors.

7.1.1 Purpose

This program plan provides guidance for the preparation and content of system or multi-system software configuration management plans (SCMPs). Each SCMP shall conform to this program plan, and shall provide system-specific details that assure and verify that each SCMP fulfills the requirements of this program plan. Each SCMP shall be prepared, reviewed, approved, and retained as a quality record. Each SCMP may provide additional, supplemental requirements as appropriate, including technology-specific details or established configuration management procedures in system and hardware design.

7.1.2 Scope

This program plan establishes a formal set of standards and methodologies used to administer and control the configurations of all process software. This program plan shall remain in effect throughout the plant life cycle. Each SCMP shall remain in effect throughout the system's life cycle for which the SCMP was written.

This program plan and system-specific SCMPs written to conform to this program plan satisfy the requirements of Regulatory Guide (RG) 1.169, "Configuration Management Plans for Digital Computer Software Used in Safety Systems of Nuclear Power Plants." RG 1.169, Revision 0, endorses Institute of Electrical and Electronics Engineers (IEEE) Standard (Std.) 828-1990, "IEEE Standard for SCM Plans," to which this program plan and each SCMP shall comply.

Specifically, the scope of this program plan includes:

- Software Configuration Management organization, including the organization, responsibilities, and applicable policies,
- Software Configuration Management activities, including the identification of configuration items, the control of configuration items, and the accounting of configuration control performance and status,
- Software Configuration Management scheduling, which establishes the sequence and coordination for the identified activities and for all events affecting the implementation,
- Software Configuration Management resources, which identifies the software tools, techniques, plans, programs, procedures, engineering instructions, equipment, personnel, and training necessary for the implementation of the specified activities, and

- Software Configuration Management plan maintenance, which identifies the activities and necessary to ensure continued configuration management during the life cycle of the project.

This SCMPP shall be implemented by Toshiba and Toshiba's contractors supplying software-based equipment. Additional oversight shall be supplied for subcontractors by Toshiba or a Toshiba's contractor responsible for that subcontractor. Additional SQA oversight shall be supplied by the customer QA and I&C organizations, with additional support as deemed necessary by the customer.

7.1.3 [Deleted]

7.1.4 Relationship of the SCMPP to Other SPP Sections

In order to simplify the description of processes and ensure clarity in establishing roles and responsibilities and to align with Branch Technical Position 7-14 (**Reference** 3) guidance, the SPP has been split into several sections. Each section of this SPP describes the processes, roles, and responsibilities for the activities in that section. Each section either explicitly or implicitly refers to other sections as necessary to implement additional processes.

Thus, this SCMPP defines the methods to be used to control the work products created during the software life cycle, as well as the methods to be used to control change to those work products, throughout the software life cycle. This SCMPP uses the activities defined in Section 2 through 6, Sections 8 through 11, Section 13, and Appendix C of this SPP. The project management functions defined in Section 12 are implemented in the customer plans, procedures, and instructions necessary to operate and maintain the nuclear power plants.

Thus, for each of the sections, the SCMPP provides the change control and configuration management processes necessary to control the life cycle activities specified by the life cycle. The SCMPP interfaces with the other sections of this SPP as follows:

- All requirements provided in Section 1 of the SPP apply to the SCMPP, including system and software organization, roles and responsibilities, classification of systems and software, and the life cycle for system and software development.

- The SPMPP, Section 2, controls and oversees the design, development, and implementation performed by the design organizations defined in accordance with the technical and process requirements listed in this Section 3, Development.

- The SDPP, Section 3, processes work cooperatively with the SCMPP to perform configuration management and change control activities in accordance with the technical and process requirements listed in Section 7, Configuration Management.

- The organization implementing Verification and Validation, Section 4, works cooperatively with the configuration management organization. This ensures that work products and changes to those V&V work products are controlled. This also ensures that verification and validation work products are managed appropriately. The implementation of the SCMPP in the V&V organization is overseen by the SPMPP.

- For safety related systems, the SCMPP works cooperatively with the Software/System Safety organization to ensure that the analyses are performed on current work products and that the analyses themselves are placed under configuration management. Notification that changes are required to completed Software/System Safety work products or that input work products are changing is provided by the Configuration Management group, in accordance with the technical and process requirements listed in SPP Section 6, Software Safety.

- The SCMPP activities ensure that the work products resulting from the system and software integration activities are controlled in accordance with the technical and process requirements listed in SPP Section 8, Software Integration.

- The SCMPP activities ensure that the work products resulting from the system and software testing activities are controlled in accordance with the technical and process requirements listed in Section 9, Testing.

- The SCMPP activities require appropriately trained personnel, and training staff coordinate with the configuration management activities to ensure that training materials and other manuals are created to ensure that plant staff can be trained, in accordance with the technical and process requirements listed in Section 10, Training.

- The SCMPP activities ensure that software installation at the vendor site and at the nuclear plants have appropriate, controlled documentation and procedures such that the installations can be controlled and performed in accordance with the technical and process requirements listed in Section 11, Installation. The SCMPP activities also ensure that the software provided for installation at site is the correct, tested version, and that all changes implemented on site during installation, commissioning, through plant startup are also controlled and managed in accordance with the technical and process requirements listed in Section 11, Installation.

- Configuration management and change control activities and requirements continue during plant operations, to ensure that configuration is controlled as setpoint values and other configuration constants are changed. There is no configuration management organization involvement with day-to-day plant operations, as these will be controlled by the plans, procedures, and instructions provided by the customer for the normal, expected plant functions that include operation, surveillance, calibration, modifications to tunable constants and other configuration items, and troubleshooting in accordance with the technical and process requirements listed in Section 12, Operations.

- The SCMPP will be invoked as required to implement the changes required in the closure anomaly reports and implementation of enhancements to plant systems in a controlled manner, in accordance with the technical and process requirements listed in Section 13, Maintenance.

- The SCMPP ensures that all changes to work products are configuration managed and change controlled in accordance with the technical and process requirements listed in Appendix C, Secure Development and Operational Environment, to ensure that work performed does not compromise safety, for all safety systems and for other systems where cyber security requirements apply.

7.2 Software Configuration Management Overview

7.2.1 Organization

A typical organization for software projects is illustrated in Figure 1. Other organizations are acceptable as discussed in and subject to the constraints of Section 1.4.

The roles associated with Software Configuration Management are defined in Section 7.2.2.

Consistent with the discussion in Section 1.4, it is acceptable for the Project Manager (PM), Software Development Lead, and the Software Configuration Lead roles to be performed by one, two, or three engineers. The Software Quality Assurance Lead ensures the requirements in this plan are met, and must be an independent, dedicated individual, as defined in the Software Quality Assurance Program Plan (SQAPP) and each Software Quality Assurance Plan (SQAP).

7.2.2 Responsibilities

Personnel responsibilities for Software Configuration Management activities are as follows:

- Each Project Manager shall be responsible for approving all change requests to any Configuration Items.
- Each Project Manager shall be responsible for designating the Software Configuration Lead.
- Each Software Configuration Lead shall have the overall responsibility and authority for configuration management and ensuring that the activities described in this document and the project-specific SCMP are performed.
- Each Software Configuration Lead shall manage the activities of the Software Configuration Librarian. The Software Configuration Librarian shall be responsible for Configuration Item control for hardware, software, and system configurations.
- Each Software Configuration Lead will be responsible for ensuring that the Software Configuration Librarian and staff working for that librarian are cleared for access, as necessary, or coordinate to cyber security information, which may require clearance for safeguards or Official Use Only documents and information.
- Each Software Development Lead shall have overall responsibility for identifying and preparing software configuration items.
- Each Software Verification &Validation (V&V) Lead shall be responsible for reviewing the software release report (SRR) and ensuring that all configuration items are identified and properly controlled.
- Each Software Quality Assurance Lead, reporting to the Quality Assurance Manager, shall be responsible for ensuring that the requirements described in this document and the project-specific SCMP are fulfilled.

7.2.3 SCMP Program Records

7.2.3.1 System-Specific SCMP

Toshiba, each Toshiba's contractor, and each of their subcontractors supplying a digital system, or logical group of systems, shall prepare a SCMP for each system or logical group of systems. Each SCMP shall conform to the guidance of this program plan, and shall provide system-specific details that provide clear requirements for criteria to verify that the SCMP fulfills this SCMPP. Each SCMP may provide additional supplemental requirements for established configuration management procedures in system and hardware design.

Each system specific SCMP will comply with the following sequence of sections. The list includes references to the sections of this document that provide guidance for the content of that section:

- Introduction (including: overview, purpose, scope, definitions, and references)
- SCM Management (Section 7.2)
- SCM Activities (Section 7.4)
- SCM Schedules (Section 7.6)
- SCM Resources (Section 7.3)
- SCM Plan Maintenance (Section 7.8)

All records shall be subject to the requirements of cyber security and the Secure Development and Operational Environment, as defined in Appendix C.

7.2.3.2 Software Release Report

Each release of software associated with a digital system, logical group of systems, or release of equipment configuration with digital content shall be accompanied by a Software Release Report (SRR), to document the Configuration Items related to that release. It shall contain at least:

- Identification of all Hardware Configuration Items on which the build will operate. The hardware Configuration Items are identified using the equipment supplier's designation or model number, supplemented as necessary to identify clearly any necessary version information.
- Identification of all Hardware and Software Configuration Items required to build the current release. The software Configuration Items are identified using the scheme defined in Section 7.4.1.1. All Configuration Items are identified to define the build and target environment for the software.
- Instructions necessary to create the build environment. The instructions are used to define the build environment and should include at a minimum (as applicable):
    - Installation instructions for compilers
    - Installation instructions of tools or third-party software on the build machine
    - Methods to verify correct installation

7.2.4 Applicable Policies, Directives, and Procedures

This plan provides complementary guidance to Toshiba, each Toshiba's contractor, or their subcontractors, configuration management procedures for system and hardware, by providing enhanced requirements specific to software configuration management. Unless otherwise approved, the system configuration, including hardware and software, shall be maintained in an automated, electronic configuration management system. Any exceptions or deviations to the requirements in this program plan shall be documented and justified in each SCMP, and reviewed and approved by the customer.

7.2.5 Schedule

All activities within this plan shall be planned and scheduled, in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

7.3 Software Configuration Management Resources

Software Configuration Management resource information identifies the software tools, techniques, equipment, personnel, and training necessary for the implementation of the specified activities. For each type of configuration management activity identified, the SCMP shall specify what tools, techniques, equipment, personnel, and training are required and how each resource will be provided, obtained, and maintained.

7.3.1 Tools

Any tool used during testing shall be evaluated by the processes defined in the Software Development Plan (Section 3.11.3.4).

For each software tool used in the each Toshiba's organization, each Toshiba's contractor, and each of their subcontractors software life cycles, whether developed specifically for the project, previously developed, or purchased, the software tool evaluation shall describe or reference the software tool's functions and shall identify the configuration controls to be placed on the tool. At a minimum, the SCMP shall detail the type, location, and use of the master file repository, as described in the following section. The SCMP shall also identify the methods used to determine that each tool is adequate for its intended use and the safety significance of the system or systems on which it is applied.

7.3.2 Master File Repository

A master file shall be created and maintained for the duration of the each Toshiba's organization, each Toshiba's contractor, and their subcontractors' responsibility for any aspect of their scope of supply, including but not limited to system, software, hardware, source code, reports, Corrective Action Reports, and other items placed under configuration control by this Software Program Plan. Each Master File shall be an electronic repository for Configuration Items, including source code. Each Master File shall be separate from local copies used by Lead Engineers or other project engineers during systems and software development.

The form and location of each Master File shall be detailed in each SCMP. The Master File shall be used in accordance with the following guidelines:

- The Software Configuration Lead shall be responsible for each Master File.

- The Software Configuration Librarian shall be responsible for maintenance of the Master File.
- The Master File shall contain the master copies of the current and past revisions of all source code.
- Lead Engineers or other project engineers shall check-in source code into the Master File at a frequency defined by the Project Manager. This should be at least weekly, and shall be performed at the creation of a new software component/unit, at the addition of a new feature, at the release of a new version of the software component/unit, or after a significant modification.
- The Master File shall contain all releases of Software Items including at a minimum the following items (as applicable):
    - Software Release Report
    - Software Requirements Specification
    - Software Design Description
    - Data communication protocol documents
    - Source code
    - Supporting software items
    - Compiled executables
- The master file shall be on electronic media that is periodically backed up to a separate electronic media and stored in a location separate from the normal location of the Master File.
- The master file shall be retained throughout the life cycle of the system, and maintained until the customer, Toshiba, Toshiba's contractors, and the vendors no longer have responsibility for any aspect of activities to include, but not be limited to, design, implementation, review, test, maintenance, or retirement of the system, hardware, software, configuration, troubleshooting, or repair.

7.3.3 Training

For each type of configuration management activity identified, the SCMP shall specify what personnel and training are required and how training will be provided or obtained. This training shall be provided to the applicable Toshiba and Toshiba' contractor's staff to the extent that responsibilities are transferred to the customer staff. This training shall be provided to the customer staff according to th ecustomer's request. Training needs are provided in the Software Training Program Plan.

7.4 Software Configuration Management Activities

This section identifies functions and tasks required to manage the configuration of the software system. Each SCMP will restate the requirements of this generic plan and provide more detail as appropriate.

As in IEEE Std. 828-1990, Software Configuration Activities are grouped into four functions: configuration identification, configuration control, status accounting, and configuration audits and reviews. The information requirements for each function are identified in Sections 7.4.1 through 7.4.4.

7.4.1 Configuration Identification

Configuration identification activities shall identify, name, and describe the documented physical and functional characteristics of the hardware, code, specifications, design, verification and validation reports, test reports, data elements, and other software process artifacts to include reports and the corrective action program database. These documents shall be controlled and placed under configuration control. Controlled items shall also include intermediate and final outputs (such as executable code, source code, user documentation, program listings, databases, test cases, test plans, specifications, and management plans) and elements of the support environment (such as compilers, operating systems, programming tools, and test beds).

Each SCMP shall require the generation of software release reports to identify the project configuration items (CI). This document shall be reviewed and updated with any additional CIs for each development phase. Each SCMP shall state how each CI and its versions are to be uniquely named. Each SCMP shall describe the activities performed to define, track, store, revise, maintain, and retrieve CIs. Information required for configuration identification is specified in Sections 7.4.1.1 through 7.4.1.3.

7.4.1.1 Identifying Configuration Items

Each SCMP shall require that the software release report identify all CIs to be controlled, as they evolve or are selected. The software release report shall be created and maintained as work progresses, and shall be finalized upon software release. Each SCMP shall also describe how the list of items and the structures are to be maintained for the system or systems. CIs shall include all applicable items from the following non-inclusive list:

- Custom Software, including custom created software and supporting software items
    - Software Requirements Specification
    - Software Design Description
    - Data communication protocol documents
    - Source code (this can include: classes, types, controls, libraries, or other software components/units)
    - Executables
    - Databases
    - Configuration files
    - Data files
- Third-party Software, including software delivered with the system and client supplied software items

  - Operating systems
  - Libraries
  - Drivers
  - Third party software components
- Tools, including software tools used to create any custom software
  - Compilers
  - Assemblers
  - Device Programmers for Complex Programmable Logic Devices (CPLD), Field Programmable Gate Arrays (FPGA), flash memory, etc.
  - Bootloaders
  - Other utilities
- Hardware items, including information related to the target environment
  - Firmware revision levels
  - Personal Computer (PC) Basic Input/Output System (BIOS) settings and equivalent
  - Circuit board models and revision levels
  - Switch and jumper settings
- Documentation, including any documentation that describes the software or system, including Operations and Maintenance Manuals as well as User's Manuals
- Software plans, procedures, and instructions revision and items written to comply with the software plans, to include:
  - System or Software Requirements Specification revision
  - Verification and Validation Plan revision
  - Design Document revision
  - Requirements Traceability Matrices and their revisions
  - User Manual sections of the Operations and Maintenance (O&M) Manual as well as the remainder of the O&M Manual, and their revision(s)
  - Test Procedures, Test Cases, Test Reports, their revisions

Appropriate baselines shall be defined at control points within the project life cycle in terms of the following:

- The event that creates the baseline;
- The items that are to be controlled in the baseline;
- The procedures used to establish and change the baseline; and
- The authority required to approve changes to the approved baselined documents.

At a minimum, baselines shall be created at the end of each major phase of the software life cycle on those documents where baseline evaluations are required.

A means of identifying changes and associating them with the affected CIs and the related baseline shall be specified.

7.4.1.2 Naming Configuration Items

Each SCMP shall specify an identification system for assigning unique identifiers to each item to be controlled. Each SCMP shall also specify how different versions of each CI are to be uniquely identified. Identification methods could include naming conventions and version numbers and letters.

Each SCMP shall describe the methods for naming controlled items for purposes of storage, retrieval, tracking, reproduction, maintenance, modification, and distribution. Activities may include version marking, labeling of documentation and executable software, serialization, and altered item marking for executable code or data embedded on a microchip, and identification of physical packaging.

Included in the methods for naming controlled items shall be subcontracted software, proprietary software, and support software such as software tools, operating systems, software required for maintenance and calibration, and other equivalent items may require special identification schemes and labeling.

The following scheme should be used to identify all software Configuration Items (custom and third party) uniquely during development and/or at each release. If this scheme is not used, Toshiba and Toshiba's contractor shall propose their standard method in each SCMP, for review and approval by the customer.

Custom Software

Custom software version numbering can be defined by the following:

- Major – a designation of major revisions of the software. The major number shall be 1 for the first release to the client
- Minor – a designation of minor revisions of the software that occur during development or during change control of a release
- Unique ID – is a unique build number, date, or other unique identification code

Procured Software

Procured software version numbering can be defined by the following:

- Title – the title of the procured software item
- Manufacturer – the software manufacturer
- Unique Designator – the applicable version, release date, patch number, upgrade designation, etc., which may be specific to each manufacturer

7.4.1.3 Acquiring Configuration Items

Each SCMP shall identify the controlled software libraries for the project and describe how the code, documentation, and data of the identified baselines are to be physically placed under control in the appropriate library, including the master file repository for that system or logical group of systems. For each library, the format, location, documentation requirements, receiving and inspection requirements, and access control procedures shall be specified.

Each SCMP shall specify procedures for the physical storage of documents and permanent storage, including the physical marking and labeling of items. Data retention periods and disaster prevention and recovery procedures may also be described.

Procedures shall describe how to retrieve and reproduce controlled items from library storage. These activities shall include verification of marking and labeling, tracking of controlled copies, and protection of proprietary and security information.

7.4.2 Anomaly Reporting, Corrective Action, and Change Control

Anomaly reporting, corrective action, and change control activities shall be invoked to request, evaluate, approve or disapprove, and implement changes to baselined CIs, including resolution of errors identified in software tools. Changes shall include both error correction and enhancement. The degree of formality necessary for the change process depends on the project baseline affected, on the impact of the change within the configuration structure, and of the safety significance of the system or systems affected.

Until each system or logical group of systems is turned over to the customer, Toshiba or the Toshiba's contractor remains responsible for this process for all items supplied by that team member or their subcontractors. Once each system or logical group of systems is turned over to the customer, the customer is responsible for ensuring that appropriate arrangements have been made with the system, application, and configuration suppliers to implement anomaly reporting, corrective action, and change control.

For each identified project Configuration Item, each SCMP shall describe the change controls imposed on the baselined CIs. The SCMP shall define the following sequence of specific steps:

- Identification and documentation of the need for a change;
- Analysis and evaluation of the impact of the change request;

- Approval or disapproval of the request; and
- Verification, implementation, and release of a change to those portions of the software life cycle, including verification and validation, which were identified in the change release.

Each SCMP shall identify and update the records to be used for tracking and documenting this process for each change. Any differences in handling changes based on the origin of the request shall be explicitly documented.

7.4.2.1 Requesting Changes

Each SCMP shall specify the procedures for initiating a Software Modification Request (SMR), which documents a change request to a baselined CI and the information to be documented for the request.

The intent of the SMR process is to provide a standard method to:

- Collect all pertinent information regarding a change to software requirements, design, implementation, or testing;
- Track the decisions made throughout the SMR processing;
- Support the Software Configuration Management Process; and
- Provide sufficient data for the Configuration Control Board to assess the impact of the requested change sufficiently to approve or disapprove the change

As a minimum, the information recorded for a proposed change shall contain the following:

- Unique change request tracking number;
- The name(s) and version(s) of the CIs where the anomaly appears;
- Originator's name and organization;
- Detailed information concerning the life cycle phase and activity in progress when the proposed change was identified, sufficient to allow recreation of test anomalies, etc.;
- Date of request;
- Indication of urgency;
- How the anomaly manifests, and enough data to allow repeating the anomaly, or notation that the anomaly cannot be recreated;
- The need for the change;
- Description of the requested change;
- Initial estimate of the impact of the requested change on documentation, code, review, and testing, at least by listing the life cycle phases and primary documents affected;

- Status of the change request shall be added to each record in the Evaluation process (see Section 7.4.2.2);
- Priority and classification of the change request will be incorporated into each record as needed; and
- Disposition of the change request shall be added to each record in the Approving or Disapproving Changes process (see Section 7.4.2.3).

7.4.2.2 Evaluating Changes

Each SCMP shall require a Software Modification Analysis (SMA) for each change request to determine the impact of the proposed change and the procedures for analysis of the impact of the change.

Each SMA shall include input from the Software Development Lead, Software Verification and Validation Lead, Software Safety Analysis Lead, Software Test Lead, Software Training Lead, Software Installation Lead, Cyber Security Lead, and their teams. Additional input shall be solicited from site installation and testing organizations if the change impacts previously completed installation or testing instructions. Additional input shall be solicited from the Human Factors Engineering organization if the human-system interface design is affected.

Changes shall be evaluated according to their effect on the deliverable, their impact on project resources, and the safety significance of the affected system or systems.

Each SMA shall determine the software life cycle phases to be re-entered, and the review and testing requirements necessary to evaluate the proposed change. The SMA shall consider and document the expected impact on work products, to include, but not be limited to, software requirements, software detailed design, operation and maintenance manuals, test plans, test procedures, test cases, code review reports, V&V reports, factory acceptance tests, DIV, installation, commissioning, and pre-operational testing, and cost and schedule issues.

7.4.2.3 Approving or Disapproving Changes

An I&C Configuration Control Board (CCB) will be created as necessary to evaluate and approve/disapprove each Software Modification Request and Analysis for Toshiba and Toshiba's contractors. This I&C CCB will be developed, maintained, and led by the customer. This CCB will have voting senior staff management and technical members that represent the customer I&C organization Toshiba, and Toshiba's contractors. Additional members of this CCB will be added as deemed necessary and appropriate by the customer. Documents will be evaluated and approved by the Cyber Security Team prior to their approval by the I&C CCB. This CCB will control changes at the project level. Toshiba, each Toshiba's contractor, and their subcontractors are responsible for implementing the changes approved by the I&C CCB.

Each SCMP shall identify an appropriate Configuration Control Board (CCB) and the CCB's authority for approving proposed changes through the associated SMR and SMA. The Project Manager (PM) responsible for the software life cycle activities shall present the change to the CCB. Multiple levels of CCBs may be specified, depending upon the degree of system or project complexity, safety significance of the system or systems, and the project baseline involved. When multiple CCBs are used, each SCMP shall specify how the proper level is determined for a change request, including any variations during the project life cycle. For any CCB utilized, the SCMP shall indicate its level of authority and its responsibilities as defined in Section 7.2.2.

Each CCB shall control the introduction of changes into the plant and system life cycles to minimize the impacts of technical and project risks introduced by changes within a controlled software process.

The CCB should generate and/or evaluate metrics associated with process compliance and software quality, to determine if process changes, staff training, or other quality enhancing measures are required. The CCB should review any software metrics associated with the proposed change, to ensure that Toshiba and Toshiba's contractors are maintaining the expected quality level.

Changes that the CCB disapproves shall document the rationale for their disapproval in the change request.

As Configuration Items, complete change requests shall be retained. Each SCMP shall document the requirements for signatures of staff involved in the creation, evaluation, approval, and closure of change requests.

7.4.2.4 Implementing Changes

Each SCMP shall specify the activities for verifying and implementing an approved change. The information recorded for the completion of a change shall contain the following as a minimum:

- The associated change request(s);
- The names and versions of the affected items;
- Determine of the re-entry points into the software life cycle processes and work products to be revised;
- Verification date and responsible party;
- Release or installation date and responsible party; and
- The identifier of the new version.

Additional information, such as software fault metrics or identification of the supporting software used to implement the change, may be included. Each SCMP shall also specify activities for release planning and control, i.e., coordinating multiple changes, reconfiguring the CIs, and delivering a new baseline.

Each SCMP shall define how all SMRs are logged, tracked, and documented throughout the various affected life-cycle phases.

7.4.3 Configuration Status Accounting

Configuration status accounting activities record and report the status of project CIs. Each SCMP shall include information on the following:

- What data elements are to be tracked and reported for baselines and changes;
- What types of status accounting reports are to be generated and their frequency;
- How information is to be collected, stored, processed, and reported; and

- How access to the status data is to be controlled.

If an automated system is used for any status accounting activity, the automated system's function shall be described or referenced. The following minimum data elements shall be tracked and reported for each CI:

- The initial approved version of the CI;
- The status of requested changes; and
- The implementation status of approved changes.

The level of detail and specific data required may vary according to the information needs of the project and the safety significance of the system.

7.4.4 Configuration Audits and Baseline Reviews

Configuration audits shall be performed to determine to what extent the actual CI reflects the required physical and functional characteristics, as well as to ensure that the list of CI is complete and complies with this SCMPP. Configuration reviews are management tools for establishing a baseline. Each SCMP shall identify the configuration audits and reviews to be held for the project. At a minimum, a configuration audit shall be performed on a CI prior to its release for integration.

Configuration audits shall be performed internally by the Software Quality Assurance, Development, and Verification and Validation organizations. External configuration audits should be performed on their subcontractors by Toshiba or the Toshiba's contractor at points throughout the software life cycle. Equivalent external configuration audits of Toshiba, Toshiba's contractors, and their subcontractors should be performed by the customer Nuclear Quality Assurance and I&C, at points selected by the customer staff throughout the software life cycles for the systems.

For each planned configuration audit or review, each SCMP shall define the following:

- The audit or review objective;
- The CIs under audit or review;
- The schedule of audit or review tasks;
- The procedures for conducting the audit or review;
- The participants by job title;
- Documentation required to be available for review or to support the audit or review;
- The procedure for recording any deficiencies and reporting corrective actions; and
- The approval criteria and the specific action(s) to occur upon approval.

7.4.5 Interface Control

Interface control activities coordinate changes to the project CIs with changes to interfacing items outside the scope of the digital system software product. Hardware, system software and support software, as well as other projects and deliverables, shall be examined for potential interfacing effects on the project.

Each SCMP shall identify the external items to which the project software interfaces. For each interface, each SCMP shall define the following:

- The nature of the interface;
- The affected organizations;
- How the interface code, documentation, and data are to be controlled; and
- How the interface control documents are approved and released into a specified baseline.

7.4.6 Subcontractor and Vendor Control

Subcontractor control activities incorporate items developed by subcontractors to Toshiba and Toshiba's contractor prior to commercial operation. These subcontractors generate CIs for the plant prior to turnover to the customer. After commercial operations, the customer will likely employ subcontractors during the Operation and Maintenance of the nuclear power plant which are also likely to add, modify, and delete CIs.

Vendor control activities shall include items purchased by a subcontractor, Toshiba, a Toshiba's contractor, or the customer.

Software developed by contract and software acquired in its finished form, from subcontractors and vendors other than those belonging to Toshiba Nuclear Energy Systems and Services Division (NED) or NED's contractor, shall be included in this classification. Special attention should be directed to these SCM activities due to the added organizational and legal relationships.

For both subcontracted and acquired software, Toshiba NED or NED's contractors shall either verify that the subcontractor's or vendor's configuration management program complies with the expectations of this SCMPP, or generate an SCMP for that subcontractor's or vendor's activities. In either case, the SCMP shall define the activities to incorporate the externally developed items into the customer CIs and to coordinate changes to these items with the external development organizations.

For subcontracted software, each SCMP invoking software subcontracting shall describe the following:

- What configuration management requirements are to be part of the subcontractor's agreement;
- How the subcontractor will be monitored for compliance;
- What configuration audits and reviews of subcontractor items will be held;
- How external code, documentation, and data will be tested, verified, accepted, and merged with the plant software;

- How proprietary items will be handled for security of information and traceability of ownership (e.g., copyright and royalties);
- How the subcontractor will comply with other plant-level plans, programs, and processes associated with software, including Secure Development and Operational Environment, Cyber Security, and requirements management; and
- How changes are to be processed, including the subcontractor's participation.

For purchased software, each SCMP shall describe how the acquired system, software, hardware, user's manuals, design documentation, and other software work products and artifacts will be received, reviewed, tested, and placed under configuration management; how changes to the supplier's software are to be processed; and whether and how the supplier will participate in the plant change management process.

Acquired software can come from a vendor, a subcontractor, a customer, another project, or other source.

Each SCMP shall describe how these CIs will be transferred to the customer.

7.5 Software Release Report

The software release process contained in each SCMP shall ensure evaluation of software system verification and validation results to determine whether the system is suitable for delivery or return to operability and service, and verifies that identified anomalies have been resolved adequately, or deferred with rationale for not resolving them.

Prior to release, the Software V&V Lead shall verify that all software document reviews and test activities are complete and shall prepare a software V&V summary to document the completion status.

The Software Development Lead shall prepare the Software Release Report, which describes:

- Any remaining anomalies (defects) and the rationale for not resolving them (e.g., evaluation that they do not contribute to non-negligible risk);
- The released version of the software, listing the version number(s) of all software components and a list of all software tools and their versions necessary to control or rebuild the software, which shall be based on the revision data which shall be provided in the source code, if source code is available;
- The identification and version numbers of other configuration items; and
- The environment used to build the software product.

The complete, released software in source and executable forms shall be electronically archived so that it can be restored in each released configuration.

7.6 Software Configuration Management Schedule

Software configuration management schedule information establishes the sequence and coordination for the identified activities and for all events affecting each SCMP's implementation.

Each SCMP shall state the sequence and dependencies among all software configuration management activities. Each SCMP shall state the relationship of key activities to project milestones or events. The schedule shall cover the duration of the SCMP. The schedule shall contain all major milestones for the system or logical group of systems, related to software configuration management activities. Milestones shall include establishment of a configuration baseline, implementation of change control procedures, and the start and completion dates for a configuration audit.

All activities within this plan shall be planned and scheduled in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

7.7 Software Configuration Management Process Requirements

Each SCMP shall identify the software configuration management resource information, to include identification of the software tools, techniques, equipment, personnel, and training necessary for the implementation of the specified activities. Software configuration management can be performed by a combination of automated software tools and manual procedures. Procedures shall be written to define use of the automated software tools for each system or logical group of systems.

Tools can be specific to software configuration management or embedded in general project aids, and can be specific to a given system or set of system. Tools can be standard organizational resources or specially acquired or built for this project. Tools can be applied to library structure and access control; documentation development and tracking; code control; baseline system generation; change processing, communication, and authorization; change/anomaly tracking and status reporting; archiving, retention, and retrieval of controlled items; or the software configuration management planning process itself.

For each type of software configuration management activity identified, each SCMP shall specify what tools, techniques, equipment, procedures, personnel, and training are required and how each resource will be provided or obtained.

For each software tool, whether developed for this or another project or purchased, each SCMP shall describe or reference the tool functions and shall identify the configuration controls to be placed on the tool.

Each SCMP shall also specify how these tools, techniques, equipment, procedures, and training shall be transferred to the customer on delivery of their system.

7.8 Software Configuration Management Plan Maintenance

Software configuration management plan maintenance information identifies the activities and responsibilities necessary to ensure continued software configuration management planning during the life cycle of the project. Each SCMP shall state the following:

- Who is responsible for monitoring this SCMP;
- How frequently updates are to be performed;
- How changes to this SCMP are to be evaluated by the customer;
- How changes to this SCMP are to be evaluated and approved;

- How changes to this SCMP are to be made and communicated.

Each SCMP should be reviewed at the start of each project software phase, changed accordingly, and approved and distributed to the appropriate staff for Toshiba, Toshiba's contractors. If each SCMP has been constructed with detailed procedures documented elsewhere in appendixes or references, different maintenance mechanisms for those procedures may be appropriate.

The software configuration management schedule that establishes the sequence and tasks shall be specified in the Integrated Project Schedule, and assigned to the individuals responsible for the software configuration management tasks.

8 Software Integration Program Plan (SIntPP)

8.1 Introduction

8.1.1 Purpose

This program plan defines the preparation and content of Software Integration Plans (SIntPs) for each system or logical group of systems. Each SIntP shall be prepared, reviewed, approved, and retained as a quality record.

This program plan addresses both the integration of software into hardware to form a system or logical group of systems as well as the integration of systems into the larger plant digital monitoring and control system.

8.1.2 Scope

This document provides general planning requirements to be included in each SIntP. Specifically, each SIntP shall include:

- The roles and responsibilities for the management of software integration
- The activities that comprise software and hardware integration and planning for integration
- The relationship between software and hardware integration and software development
- The relationship between software and hardware integration and software Verification and Validation (V&V)

This plan satisfies the requirements of Regulatory Guide 1.173, which endorses Institute of Electrical and Electronics Engineers (IEEE) Standard (Std.) 1074 (Reference 27).

This SIntPP shall be implemented by Toshiba, each Toshiba's contractor supplying software-based equipment. Additional oversight shall be supplied for subcontractors by Toshiba or a Toshiba's contractor responsible for that subcontractor. Additional SQA oversight will be supplied by the customer QA and I&C organizations, with additional support as deemed necessary by the customer.

8.1.3 [Deleted]

8.1.4 Relationship of the SIntPP to Other SPP Sections

In order to simplify the description of processes and ensure clarity in establishing roles and responsibilities and to align with Branch Technical Position 7-14 (**Reference** 3) guidance, the SPP has been split into several sections. Each section of this SPP describes the processes, roles, and responsibilities for the activities in that section. Each section either explicitly or implicitly refers to other sections as necessary to implement additional processes.

Thus, this SIntPP defines the methods to be used to integrate and interface with the plans to verify and validate that the software and hardware provide the required functionality and that the configuration as tested can be reproduced through records retained during the integration, using other portions of the activities defined in Section 2 through 7, Sections 9 through 11, Section 13, and Appendix C of this SPP. The project management functions defined in Section 12 are implemented in the customer plans, procedures, and instructions necessary to operate and maintain the nuclear power plants.

Thus, for each of the sections, the SIntPP provides the technical activities necessary to integrate individual systems and the integrated plant, as specified by the life cycle. The SIntPP interfaces with the other sections of this SPP as follows:

- All requirements provided in Section 1 of the SPP apply to the SIntPP, including system and software organization, roles and responsibilities, classification of systems and software, and the life cycle for system and software development.

- The SPMPP, Section 2, controls and oversees the activities performed by the integration staff in accordance with the technical and process requirements listed in this Section, Integration.

- The SDPP, Section 3, works cooperatively with this SIntPP, as required to perform the necessary activities in accordance with the technical and process requirements of Section 3, Development, and Section 8, Integration.

- The SVVPP, Section 4, works cooperatively with the integration organization, to ensure that the V&V activities required for integration are performed and documented based on the V&V organization's plans, procedures, and engineering instructions.

- The implementation of the SIntPP and SVVPP activities is overseen by the software quality assurance activities performed by the quality assurance organization, in accordance with the technical and process requirements listed in Section 5, Software Quality Assurance.

- The SIntPP works cooperatively with the Software/System Safety organization to ensure that the implementation activities are performed in a manner such that identified safety concerns are verified and validated. The Software/System Safety work is performed in accordance with the plans, procedures, and engineering instructions in accordance with the technical and process requirements listed in Section 6, Software Safety.

- The SIntPP activities coordinate with the change control and configuration management activities. The change control and configuration management activities are performed in accordance with the technical and process requirements listed in Section 7, Configuration Management.

- The SIntPP activities coordinate or perform (depending on the safety classification) system and software integration activities to prepare for testing activities, in accordance with the technical and process requirements listed in Section 9, Testing.

- The SIntPP activities use appropriately trained personnel during integration activities. The SIntPP staff coordinate with generation of such materials to ensure that training materials and other manuals are reviewed such that plant staff can be trained appropriately, in accordance with the technical and process requirements listed in Section 10, Training.

- The SIntPP activities ensure that software installation at the vendor site and at the nuclear plant have appropriate documentation and procedures such that the installations can be controlled and

performed in accordance with the technical and process requirements listed in Section 11, Installation.

- There are no SIntPP activities or requirements associated with plant operations, as these will be controlled by the plans, procedures, and instructions provided by the customer for the normal, expected plant functions that include operation, surveillance, calibration, modifications to tunable constants and other configuration items, and troubleshooting in accordance with the technical and process requirements listed in Section 12, Operations.

- The SIntPP will be invoked as required to integrate the changes required to close anomaly reports and implement enhancements to plant systems in a controlled manner, in accordance with the technical and process requirements listed in Section 13, Maintenance.

- The SIntPP ensures that the integration activities are performed such that the technical and process requirements listed in Appendix C, Secure Development and Operational Environment are maintained and not compromised, for all safety systems and for other systems where cyber security requirements apply.

8.2 Organization

A typical organization for software projects is illustrated in Figure 1. Other organizations are acceptable as discussed in and subject to the constraints of Section 1.4.

The Project Manager, the Software Development Lead, the Software Integration Lead, the Software V&V Lead, and the Software Quality Assurance Lead shall be responsible for software and system integration. While these roles should be fulfilled by separate staff, it is acceptable to combine any of these roles except the V&V Lead and the Software Quality Assurance Lead, which shall each be independent, dedicated individuals.

8.2.1 Responsibilities

Personnel responsibilities for Software Integration activities are as follows:

- Each Project Manager shall be responsible for designating and identifying the Software Integration Lead in a quality record.

- Each Software Integration Lead shall have overall responsibility for planning and execution of Software Integration, providing input for Integration testing, and turnover to Integration testing.

- Each Software Integration Lead shall have the responsibility for ensuring that team members are cleared for access, as necessary, or coordinate with the appropriately cleared Cyber Security Team who already has access to cyber security information, which may require clearance for safeguards or Official Use Only documents and information

- Each Software V&V Lead shall be responsible for the preparation and execution of Software Integration testing, and ensuring such testing demonstrates that no defects were introduced during the integration process, that defects in the software design are uncovered, and that errors or defects associated with the interfaces between software or hardware units are found.

- Each Software Quality Assurance Lead shall be responsible for ensuring that the requirements described in the SIntPP and each SIntP are fulfilled.

8.2.2 Schedule

All activities within this plan shall be planned and scheduled in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

8.3 Process

8.3.1 Process Background

During the software design phase, software requirements are translated into a documented software design. An early step of the design process develops the software architecture, which decomposes the design into multiple design units and defines the interfaces in between the units. The decomposition is often performed in multiple stages, from a logical group of systems to individual systems, then from each system to subsystems, then to software components and modules. The appropriate scope of the lowest level of individual software units is system-dependent, but is often selected so that (a) the unit is of appropriate complexity for a single developer or group of developers, and (b) the unit is associated with (e.g., runs on, or interfaces with) a single hardware item. Software and hardware integration is the process of recomposing the software units into a complete system or group of systems, including the hardware on which it runs.

8.3.2 Integration

A Software Integration Plan (SIntP) shall be written for each system or logical group of systems. Each SIntP shall address the performance of these high-level activities. Each SIntP shall specify the following:

- Software units shall not be integrated until the development of that unit is complete and any planned unit verification has been performed.

- Integration shall be performed to a written plan. The plan shall define the software and hardware to be integrated. The plan shall define the order in which integration is to be performed, and the test procedures and test cases to be used to verify interfaces between software units and between software and hardware.

- Software integration should be performed as a staged process. For example, software units that run on a common hardware processor, or a software unit whose primary purpose is to interface with a single hardware unit, are often integrated early in the integration process.

8.3.3 Program Plan Generation

A Software Integration Plan (SIntP) shall be prepared for each system or logical group of systems. Each SIntP shall be based on the System Architecture Description and the Software Detailed Design Document or Documents appropriate to the system or logical group of systems being integrated.

Each SIntP shall require preparation, review, and approval of those SIntP procedures and tests cases necessary to verify and validate that the software and hardware have been appropriately integrated and to detect errors in the integration.

8.3.4 Scheduling and Planning

The SIntP will detail the specific sequence of integration activities, including integration testing activities. This shall include any partial integration, such as the integration of a subset of software or hardware components that are available earlier in the development cycle for intermediate integration testing, to be ultimately integrated into a final product. The Software Integration Lead shall coordinate with the Project Manager and Software Development Lead to ensure an efficient sequence is realized.

8.3.5 Resources

The SIntP shall describe any resources that will be required to support the software and hardware integration sequence and integration testing.

Any tools used during the integration and integration testing shall be evaluated by the processes defined in the Software Development Plan (Section 3.11.3.4).

8.3.6 Training

The SIntP shall define any training requirements for any personnel participating in software integration, and shall provide a plan for ensuring that personnel are trained according to those requirements in accordance with each Software Training Plan. Training records shall be retained as quality records.

8.3.7 Reviews

The SIntP, the test procedures, and test cases required for integration will be peer reviewed using independent staff as defined in the SVVP (see Section 4) and approved by the Project Manager.

8.3.8 Software Integration Activities

All software integration documents shall be retained in accordance with each Software Configuration Management Plan (see Section 7).

8.3.8.1 Prepare the Software Integration Plan

Each software integration plan and the SIntP Procedures and Test Cases necessary to implement each SIntP shall define appropriate aspects of software integration, integration of the software with the hardware, and integration of the system into the customer. Each set of SIntP plans, procedures, and test cases shall reference the complete set of software design documentation for definition of the software items. Each set of SIntP plans, procedures, and test cases shall specifically describe:

- The plan for the integration activity (referencing software development planning documents as appropriate), including:
    - The integration stages, and which software components or units shall be combined with which hardware components at each stage,

  - Any personnel and supporting hardware and software that shall be available for each integration stage, and
  - Any considerations relating to interfaces with software or hardware units that are not yet available for integration, such as emulation or software stubs.

- The plan, procedures, and tests cases for integration testing (referencing software verification and validation planning documents, as appropriate), including:
  - Any unit verification activities that shall be complete before a unit is considered ready for integration, and
  - The integration test activities that are performed on the result of any integration stage.

- If a group of systems is incorporated in the plan, the plan shall explicitly define the process for integrating the systems, or shall document the means used to ensure that execution of one system does not have any effect on operation of any other system within the logical group of systems. If any of the systems in the logical group of systems can adversely impact the operation of another system within that logical group of systems, the interaction between those systems shall be documented, and explicit instructions provided in the plan for those interactions.

The software integration plan should reference hardware and system planning and scheduling documents to coordinate the availability of any required personnel, software tools, or hardware components. The software integration plan shall also reference the software verification and validation plan for details on specific integration verification activities.

The input to the development of the software integration sequence is the software requirements specification and design documentation such as the software architecture description (see Section 3.10.3.3) for the system, which describes all software units and specifies the interfaces between software units. The integration plan shall be peer reviewed, controlled as a quality document, and identified as a configuration item for software configuration management.

8.3.8.2 Integration Testing

Integration testing is distinguished from other types of testing by its purpose. While unit testing verifies the design and implementation of the unit, and system testing validates the final design of the system, an integration test is a test specifically designed to uncover two types of flaws:

- Flaws associated with the interface between two or more software and hardware units. For example, errors are likely if the developers of two separate units with a common interface did not interpret the designed meaning of a data item on the interface definition the same way.
- Flaws resulting from an interaction between two units. For example, acceptable latency on a communication channel cannot usually be tested until both ends of the channel are active and communicating.

Integration testing may be performed as distinct testing activities, or may be combined with other testing activities. If integration testing is performed as part of another testing activity, then those testing activities shall be identified in the SIntP, and review of that testing activity shall confirm that the test fulfills the two purposes of integration testing given above.

Integration testing activities shall be planned and documented in each SIntP. The preparation of the integration test procedures is the responsibility of the Software V&V Lead, and shall be performed and executed in accordance with the guidance and requirements in Section 3.8. Integration testing activities shall be carried out by test engineers having the independence required in Section 1.4.2, with oversight by the Software Verification and Validation Lead.

8.3.8.3 Software Integration Execution

The software integration activities shall be carried out by software and hardware engineers under the oversight of the Software Development Lead, in accordance with the instructions provided in each SIntP.

All outputs of the Integration phase, including as appropriate, the project source, the project binaries, and the installation package, shall be controlled as quality documents in the Master File repository (Section 7.3.2).

Any issues discovered during integration shall be entered into the Corrective Action Program (see Section 5.8). If a change is required to resolve the issue, a Software Modification Request (SMR) shall be issued (see Section 7.4.2). If the Software Modification Analysis (SMA) determines that an issue requires redesign or reimplementation, that issue shall be referred to the appropriate developers, who are tasked with performing the appropriate activities, and updating the design documentation following the requirements provided by the SMA. When ready, the modules are resubmitted for verification and validation, and then integration.

8.3.8.4 Software Integration Reporting

Reporting of the integration activity and integration testing shall be as follows:

- SIntP documents indicate that the integration activity has been appropriately planned and that integration testing activities have been appropriately selected.
- Integration test procedures (or integration test portions of other test procedures) document that the software unit interfaces and interactions are verified adequately.
- Integration test reports (or integration test portions of other test reports) document that the software unit interfaces and interactions were tested in accordance with the procedure. Earlier procedure reviews verified that sufficient objective evidence would be gathered and retained to demonstrate software safety and quality.
- Corrective Action Program Reports resulting from integration testing document all anomalies found during integration. Resolution of these issues documents the successful execution of integration and integration testing.

All these documents shall be signed, reviewed, approved, and retained in accordance with the requirements of each SIntP and of each Software Configuration Management Plan.

9 Software Test Program Plan (STPP)

9.1 Introduction

This Software Test Program Plan (STPP) defines the processes and activities used to test the software during the module testing through Pre-Operational testing phases. Each Software Test Plan written to test systems or logical groups of systems will define the scope, approach, resources, and schedule of the testing activities.

9.1.1 Purpose

For Toshiba and each Toshiba's contractor supplying systems containing off-the-shelf, commercial, previously developed, and/or custom software to the customer's plant, a Software Test Plan shall be created in accordance with this program plan. Systems within the scope of this Software Program Plan, whether classified as safety related, nonsafety Group 1, or nonsafety Group 2 shall have Software Test Plans generated in accordance with this STPP. Each Software Test Plan shall comply with the requirements of this program plan. Exceptions shall be defined and documented, and provided with the Software Test Plan to the customer for review and approval according to the customer's request. Each Software Test Plan shall be prepared, reviewed, approved, and retained as a quality record.

Subcontractors to Toshiba or a Toshiba's contractor supplying software-based products shall be subject to the same requirements as Toshiba and the Toshiba's contractors. For subcontractors, Toshiba or the Toshiba's contractor initiating the subcontract shall be responsible for ensuring compliance with this plan. However, as with systems supplied directly from Toshiba and Toshiba's contractors, each plan and any documented exceptions will be supplied to the customer for review and approval.

This Software Program Plan (SPP) is based on the premise that systems to be installed in the plant should be tested at Toshiba's or Toshiba's contractor's site or at their subcontractor's site by an appropriate Platform Factory Test (PFT) before being shipped to the customer for installation and use. If the platform is part of the Platforms Integration Test (PIT), then that testing should be complete as well before the system is shipped to the customer for installation and use.

9.1.2 Scope

The scope of this program plan is to describe the upper level activities required to test:

- Individual software modules or unit
- Integration of software modules and units
- Integration of hardware and software into systems
- Initial integration tests for systems
- Platform Factory Test (PFT)
- Platforms Integration Testing (PIT)

- Installation, Commissioning, and Design Installation Verification (DIV) Testing[11]
- Pre-Operational testing

The set of tests performed on individual systems, the integrated system of systems, and the integrated system of systems in the plant shall demonstrate the ability of the individual systems and the integrated system to perform the following:

- Maintain or restore the plant to safe conditions;
- Control the plant in safe operating conditions;
- Operate within the performance requirements;
- Operate within data propagation requirements (e.g., from operator request to action in the plant, from plant event to alarm in the main control room, or from operator request to new screen displayed with current values);
- Operate when stressed by the environment; and
- Provide usable information to the operators to facilitate safe, reliable, highly available operation.

Each of these items to be demonstrated shall be considered throughout the test program, and should be tested at the earliest reasonable point during the test program. For some issues, testing may not be possible after certain development phases, which shall be documented in the test plans or procedures.

Further responsibilities for specific tests and documentation are described throughout each Software Test Plan.

This STPP shall be implemented by each Toshiba organization, each Toshiba's contractor, or their subcontractors who supply software-based equipment. Additional oversight shall be supplied for subcontractors by the Toshiba organization or Toshiba's contractor responsible for that subcontractor. Additional SQA oversight will be supplied by the customer QA and I&C organizations, with additional support as deemed necessary by the customer.

9.1.3 [Deleted]

9.1.4 Relationship of the STPP to Other SPP Sections

In order to simplify the description of processes and ensure clarity in establishing roles and responsibilities and to align with Branch Technical Position 7-14 (**Reference** 3) guidance, the SPP has been split into several sections. Each section of this SPP describes the processes, roles, and responsibilities for the activities in that section. Each section either explicitly or implicitly refers to other sections as necessary to implement additional processes.

[11] Commissioning is referred to as Construction Testing in the VITP.

Thus, this STPP defines the methods to be used to test the integrated systems and interfaces to ensure that the software and hardware provide the required functionality and that the testing itself can be reproduced based on records retained from the testing, using other portions of the activities defined in Section 2 through 8, Sections 10 and 11, Section 13, and Appendix C of this SPP. The project management functions defined in Section 12 are implemented in the customer plans, procedures, and instructions necessary to operate and maintain the nuclear power plants.

Thus, for each of the sections, the STPP provides the technical activities necessary to test individual systems and the integrated plant, as specified by the life cycle. The STPP interfaces with the other sections of this SPP as follows:

- All requirements provided in Section 1 of the SPP apply to the STPP, including system and software organization, roles and responsibilities, classification of systems and software, and the life cycle for system and software development.

- The SPMPP, Section 2, controls and oversees the activities performed by the test staff in accordance with the technical and process requirements listed in Section 9, Testing.

- The SDPP, Section 3, works cooperatively with this STPP, as required to perform the necessary activities in accordance with the technical and process requirements of Section 3, Development, and Section 9, Testing.

- The SVVPP, Section 4, works cooperatively with the Testing organization, to ensure that the V&V activities required for Testing are performed and documented based on the V&V organization's plans, procedures, and engineering instructions.

- The implementation of the STPP and SVVPP activities is overseen by the software quality assurance activities performed by the quality assurance organization, in accordance with the technical and process requirements listed in Section 5, Software Quality Assurance.

- The STPP works cooperatively with the Software/System Safety organization to ensure that the testing activities are performed in a manner that ensures that safety is not compromised that all identified safety concerns are verified and validated, and that all safety aspects are tested appropriately. The Software/System Safety work is performed in accordance with the plans, procedures, and engineering instructions, Section 6, Software Safety.

- The STPP activities coordinate with the change control and configuration management activities. The change control and configuration management activities are performed in accordance with the technical and process requirements listed in Section 7, Configuration Management.

- The STPP activities are based on the work products resulting from the system and software integration activities, which is controlled in accordance with the technical and process requirements listed in SPP Section 8, Software Integration.

- The STPP activities use appropriately trained personnel during testing activities, and coordinate with generation of such materials to ensure that training materials and other manuals are reviewed to ensure that plant staff can be trained appropriately, in accordance with the technical and process requirements listed in Section 10, Training.

- The STPP activities ensure that software testing at the vendor site and at the nuclear plant are based on appropriate documentation and procedures from integration, such that the testing can be

controlled and performed in accordance with the technical and process requirements provided from activities performed in Section 11, Installation.

- There are no STPP activities or requirements associated with plant operations, as these will be controlled by the plans, procedures, and instructions provided by the customer for the normal, expected plant functions that include operation, surveillance, calibration, modifications to tunable constants and other configuration items, and troubleshooting in accordance with the technical and process requirements listed in Section 12, Operations.

- The STPP will be invoked as required to test any changes required to close anomaly reports and implement enhancements to plant systems in a controlled manner, in accordance with the technical and process requirements listed in Section 13, Maintenance.

- The STPP ensures that all aspects affecting security are tested and that no testing activity compromises the system security, in accordance with the technical and process requirements listed in Appendix C, Secure Development and Operational Environment, for all safety systems and for other systems where cyber security requirements apply.

9.1.5 Organization, Management and Responsibilities

A typical organization for software projects is illustrated in Figure 1. Other organizations are acceptable as discussed in and subject to the constraints of Section 1.4. Responsibilities are addressed in Section 2.2.4.

Further responsibilities for specific tests and documentation are described throughout the STPP.

Each Software Test Lead will have the responsibility for ensuring that team members are cleared for access, as necessary, or coordinate to cyber security information, which may require clearance for safeguards or Official Use Only documents and information.

The Software Verification and Validation Lead shall be responsible for ensuring that all Test Team Members have the independence required in Section 1.4.2, with oversight by the Software Verification and Validation Lead. For safety systems, the Test Team Members shall be part of the Verification and Validation organization.

9.1.6 Schedule

All activities within this plan shall be planned and scheduled, in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

9.2 Software Test Overview

9.2.1 Test Structure

The hierarchy of tests to be conducted on the software products is as follows:

- Software Functional Tests, which covers:
    - Tests at the module or unit level, defined in Section 8.

  - Tests through software integration, defined in Section 8.
  - Software Validation Test (SVT) at the system level
- Platform Factory Test (PFT))
- Platforms Integration Test (PIT))
- Installation, Commissioning, and Design Installation Verification (DIV) tests
- Pre-Operational Tests

9.2.2 Test Responsibilities

Responsibilities for test generation and test performance are defined as follows:

- Software Functional Tests responsibilities are defined in the SVVPP.
- Software Validation Test responsibilities are defined in the Validation and Integration Test Plan (VITP).
- PFT responsibilities are defined in the VITP.
- PIT responsibilities are defined in the VITP.
- Installation and Commissioning Tests are defined in the applicable plans, procedures, and programs.
- DIV Test responsibilities are defined in the VITP.
- Pre-Operational Tests responsibilities are defined by the applicable Toshiba documents.

9.2.3 Test General Activities

All testing shall be performed to written test procedures and test cases. The formality of testing shall reflect the ability of the system to affect nuclear and personnel safety, to challenge safety systems, and to provide safe, reliable plant operation. That is to say, a simple nonsafety system that has no ability to challenge safety systems does not require the documentation formality of test procedures and test cases that are used for safety systems, but shall have the minimum documentation formality established by the customer for systems and software testing.

The following set of test activities shall be implemented for each type of testing.

Test preparation shall ensure that the test activities demonstrate that requirements are met, that software functions correctly. Test Preparation shall ensure that the appropriate resources are made available within the defined schedule to support the test implementation. As a minimum, test preparation shall include the following tasks:

- Define the test scope and identify the items to be tested
- Identify the reference documentation, if any, for the test

- Create a test schedule that supports, and does not conflict with, the Integrated Project Schedule
- Resolve any conflicts between the Integrated Project Schedule activities associated with the test and the planning for the test
- Identify equipment, tools, and instrumentation needed to support or perform each test
- Identify any special equipment, such as closed loop simulation and modeling, required to support each test
- Specify the test environment requirements, including security and environmental conditions. The test facility will be access controlled and limited to the test personnel and with active roles and responsibilities for the software and systems being tested, as defined by Toshiba, Toshiba's contractors, and the customer Management
- When multiple tests are being performed at the same time, a boundary shall be set up to assure that the test equipment, tools, documentation, and production hardware are not damaged and that test control is maintained
- Test planning and execution shall ensure that final, deliverable hardware is maintained in a state suitable for delivery, including maintenance of safety qualification
- Identify and specify the required qualifications and responsibilities of the individuals and organizations assigned to the test activities
- Identify and define record-keeping assignments, procedures, and forms, and test deliverables
- The change notification process or processes associated with each system or logical group of systems shall be identified and access to that procedure shall be verified
- Identify the procedures for change notification and determine responsibility as being the customer staff, Toshiba, Toshiba's contractor, subcontractor, or vendor
- Identify the procedures for correcting discrepancies and errors, and determine responsibility as being the customer staff, Toshiba, Toshiba's contractor, subcontractor, or vendor
- Prepare test plan

Test design shall identify the test strategies to verify completeness of the test and adequate coverage of testing. As a minimum, test design shall include the following tasks:

- Identify the state and configuration of the system or systems
- Based on the test purpose, determine the types of features to be tested
- Identify features and combinations of features that are not to be tested and state the reason they will not be tested
- For safety related software, each feature is formally tested under at least one test design (e.g., interfacing entities, boundary conditions, timing and response, etc.), with additional testing

encouraged. For timing tests, a statistically valid number of tests may be required to determine a valid response range

- Identify the test techniques and test approaches (i.e., performance test, interface test, etc.)
- For tests where testing may be suspended and restarted, the test design shall clearly identify points in the procedures where activities can be successfully suspended and resumed, and any test assumptions which need to be set into the equipment under test prior to restarting
- Create test cases, acceptance criteria, and test procedures to specify how each feature is tested, and define the format of the test log used to record test data
- Define acceptance criteria for each test
- Features to be tested include the following:
    - Interfaces between hardware and software
    - Communication interfaces within the system
    - Communication interfaces outside the system to other systems and equipment
    - Operation, including failure and restoration, of Foundation Fieldbus-, HART-, and Profibus-based equipment
    - The Human-System Interface, for operators, maintainers, and engineers
    - Appropriate error handling for inputs, outputs, communication links
    - Boundary conditions within functions, including but not limited to alarm limits, values at which operation changes (including reactor mode changes), values where logic changes state, upper and lower limits for engineering unit values, and upper and lower limits for current loops
    - Timing and response time determination, including but not limited to response to operator soft and hard commands, propagation time from input to output in systems where such timing is critical, control system response to transients, protection system response time from detected events to protective actions (including neutron flux instabilities in the reactor core), and control system stability

Test execution shall determine, based on the acceptance criteria, whether the feature or combination of features passed or failed. Testing shall be performed in accordance with the test procedures and test cases written, reviewed, and approved. To ensure completeness of the test coverage, test results and anomalies/errors encountered during testing shall be documented as specified in each program plan and in accordance with the SQAPP (see Section 5). As a minimum, test execution shall include the following:

- The testing team is responsible for documenting and reporting anomalies.
- The Software Quality Assurance, Design Team, and V&V Organization are responsible for analyzing, managing, and resolving identified anomalies.

- As part of the analysis of anomalies, the testing boundary and areas of the system impacted by any change shall be analyzed. The re-test shall contain at least those areas impacted by the change, and any areas that might have been impacted by the change. Interfaces shall be tested appropriately. The test that produced the anomaly shall be repeated.

- The testing phase shall continue until all anomalies are resolved and all pass/fail criteria are met.

- Electronically generated test records will be printed and attached to the test report, or filed as an electronic record and a reference provided to the file location in the test report. If electronic records are generated, copies of those electronic records shall be provided to the customer with the test report.

Test summary shall summarize the results of the testing activities and, based on those results, provide assessments. As a minimum, test summaries shall include the following tasks:

- Summarize the test activities

- Identify all resolved and unresolved anomalies and test incidents

- Summarize the related resolutions and the testing boundaries

- Summarize the test results

9.2.4 Test Review and Approval

No credit shall be taken for any testing or test activities that started before the test plan, procedures, and test cases have been reviewed and approved.

Pen and ink changes to test procedures and test cases shall be performed during testing, where obvious typographic errors were detected.

Test errors that reflect issues of understanding system purpose shall require review and approval before the results of that testing shall be considered complete, but testing should continue while such review and approval is gained.

Major changes to test procedures and test cases, including deletion of test cases, shall be reviewed and approved prior before testing shall be considered complete.

The PFT, PIT, DIV and Pre-Operational Tests shall be submitted to appropriate the customer staff for review and approval prior to running any of these tests.

Toshiba, Toshiba's contractors and/or the customer engineering staff may choose to witness tests and the customer Nuclear Quality Assurance staff may choose to witness selected tests, especially those for safety systems.

Test submittals shall include all test plans, procedures, and test cases. These submittals will be transmitted, received, and approved in accordance with processes established within Toshiba and Toshiba's contractors by the customer.

9.2.5 Test Submittal

Test submittals shall be certified by signatures of the preparer, reviewer, and approver. Test results shall also contain signatures of the testing staff and witnesses.

Test submittals shall include test documentation and test items. These submittals shall be transmitted, received, and approved in accordance with processes established within Toshiba and Toshiba's contractors by the customer.

9.2.6 Test Tools

Tools used during testing shall be evaluated by the processes defined in the Software Development Plan (Section 3.11.3.4).

9.3 Test Descriptions

Tests identified in Section 9.2.1 shall follow the order and contain the activities defined in Section 9.2.3.

Documentation for tests identified in Section 9.2.1 shall:

- Use the requirements described in Section 9.4
- Be reviewed and approved by the QA Manager prior to release
- Be placed under the control of the Software Configuration Lead

Test descriptions, detail for specific test activities, and responsibilities for each test shall be defined in the following sections.

Sections 9.3.1 through 9.3.4 define the latest point in the testing process when the defined testing shall be performed and completed. Testing elements may be distributed to earlier test phases, and credited as being completed, or partially completed, during one or more earlier test phases. Sufficient overlap shall be included and documented in the testing design documentation, to allow review by others to evaluate that the content and test expectations are fulfilled. In either case, the required testing or credit for earlier testing must be completed or credited as part of the test phase defined below, at the latest.

9.3.1 Software Validation Testing (SVT)

Software Validation Testing shall be executed to validate that the software, when installed in the target hardware, performs as specified in the requirements documents.

The SVT shall be performed to verify accuracy, correctness, robustness, safety, security, timing, and closed-loop control capability, as required. This testing shall include, but not be limited to:

- Performance Tests
- Interface Tests
- Regression Analyses and/or Tests, as applicable

- Stress Tests
- Human-System Interface Tests
- Secure Development and Operational Environment tests, as defined in Appendix C
- Cyber Security Tests

Responsibilities for Software Validation Testing are defined in the Software Verification and Validation Program Plan (SVVPP, Section 4).

9.3.2 Platform Factory Test (PFT)

System factory acceptance testing is called Platform Factory Tests (PFT). PFT shall be executed for each system or logical group of systems to validate that the software and hardware has been correctly integrated into a system, configured, and calibrated. PFT shall be performed using the hardware and validated software to be delivered to the customer. PFT shall include all system functions, human-system interface tests, with input from SDOE and cyber security.

PFT shall:

- Test the software and hardware to be used in actual operation
- Test diagnostics, to the extent practical, to be delivered for use in the customer
- Demonstrate that all performance requirements have been met
- Demonstrate that all safety functions are implemented correctly
- Demonstrate that all control functions are implemented correctly and are capable of controlling the plant equipment appropriately, in normal and transient conditions
- Exercise those sections of the system whose operation or failure could block or challenge safety functions

Scope, responsibilities, requirements, review, approval, quality assurance, and overall test design for PFT shall be defined in a separate program plan being written by the Toshiba organization or Toshiba's contractor, and are not repeated in this program plan.

9.3.3 Platforms Integration Test (PIT)

Certain systems will be integrated and tested during a platforms integration test (PIT). PIT will be executed to validate that each individual system or logical group of systems tested in PFT perform as required when integrated with other plant systems that have also completed their PFTs. PIT tests will include communication and interfaces, network stress, human-system interfaces, with inputs from SDOE and cyber security that have not already been completed or credited in a previous test phase. PIT should be performed on a completely integrated system, or as a series of tests on those available portions of the total Distributed Control and Information System (DCIS). However, the goal for PIT shall be to test the integration of the systems validated during PFT through testing of the digital communication links, hardwired links, and interfaces between those systems. PIT should test as much

of the DCIS as can be tested as an integrated whole, and should not be performed as a series of separate, independent overlap tests. Note that this intent does not preclude or dissuade the crediting of previous integration testing completed in previous PFT to support demonstration of overall system integration.

The test setup for PIT will include a representative set, as defined by a validation and integration test plan (VITP) as needed, of the following equipment types:

- At least one division of safety systems, with the other divisions simulated, and safety related human-system interfaces, with the preference of testing all divisions
- The portion of the Plant Data Network (PDN), including network switches, firewalls, and other equipment not already demonstrated in a previous PFT. The PDN testing shall include the scope defined by the VITP and its sub-tier plans and programs. The PDN testing will include any additional requirements from cyber security plans as needed, where these requirements have not been included in previous PFT scope. When these tests are complete, network loading shall have been demonstrated to be acceptable, based on worst case loading of the network.
- Data historians and other support equipment from the less-protected portions of the PDN
- Those systems classified as nonsafety Group 1
- The main safety and nonsafety portions of the Distributed Control and Information System, specifically including a simulation of the main control room, specifically:
    - The Large Display Panel
    - Actual human-system interfaces and simulation of the Main Control Console and its surrounding work area for the Reactor Operators
    - Actual human-system interfaces and simulation of the work area for the Senior Reactor Operator
    - Those portions of the DCIS associated with data historian functions, RG 1.97 displays, and post-accident monitoring
- Controllers, or alternatively high fidelity simulation of the communications interface for those controllers, for the packaged systems

Scope, responsibilities, requirements, review, approval, quality assurance, and overall test design for PIT shall be defined in a separate program plan being written by the Toshiba organization or Toshiba's contractor, and are not repeated in this program plan.

9.3.4 Commissioning and Pre-Operational Testing

Installation and commissioning tests, including cyber security and Design Installation Verification (DIV), verifies that each installed system functions correctly in the plant environment. This testing will include, but not be limited to, such aspects of system operation as verifying correction operation of the following:

- Inputs from sensors and transmitters in the plant

- Outputs to actuated equipment in the plant
- Human-system interfaces locally in the plant and in the main control room
- Interfaces to the Plant Data Network are functional and operational
- System and data communication into the functionality of the main control room, plant data historian, Technical Support Center and Emergency Operations Center
- Cyber security and functionality of the communications interfaces to the Information Technology and commercial network

DIV tests will be executed to validate the as-installed performance of the system in the plant. DIV tests will be defined as required, which may include tests of individual systems, and then of logical groups of systems. Cyber security testing will be defined and performed as required by cyber security plans and Program.

As commissioning continues, tests shall be written and performed that verify correct operation of the plant. These Pre-Operational tests shall be defined, reviewed, and approved to demonstrate successful operation of the plant.

DIV and Pre-Operational testing scope, responsibilities, requirements, review, approval, quality assurance, and overall test design shall be defined.

If changes have been made to the software, the completion of verification and validation and software safety activities on the modified software and/or configuration should be confirmed.

9.4 Test Documentation

Testing associated with PFT, PIT, and site testing including Design Installation Verification (DIV) need to comply with the content and format guidance provided in the referenced documents and the customer requirements.

Test documentation for testing defined in this plan shall meet the requirements specified in Regulatory Guide (RG) 1.170, Revision 0, "Software Test Documentation for Digital Computer Software Used in Safety Systems of Nuclear Power Plants." RG 1.170 endorses Institute of Electrical and Electronics Engineers (IEEE) Standard (Std.) 829-1983, "IEEE Standard for Software Test Documentation," (**Reference** 17). The IEEE standard recommends that each Software Test Plan consist of certain test documents. This STPP is based on recommendations in IEEE Std. 829. Each Software Test Plan shall require generation of the following documents, which are listed as:

- Test Plan
- Test Design Specification
- Test Case Specification
- Test Procedure Specification
- Test Logs

- Test Item Transmittal Reports
- Test Incident Reports
- Test Summary Reports

The test documents listed above do not have to be generated as individual documents; however, the information required shall be included in at least one of the test documents, which shall conform to the requirements of IEEE 829-1983.

The following sections describe the test documentation required to support the tests described in Section 9.3.

If there is no information pertinent to a topic or section in any of the following documents, the following phrase, "This topic or section is not applicable to this test plan," shall be provided in the test document, along with an appropriate reason for not providing material in that section.

9.4.1 Test Plan

A test plan shall be prepared for each system or logical group of systems, and shall assure the completeness of the testing process.

At a minimum, the test plan shall encompass the test preparation and design activities described in the Section 9.2.3, and the test plan shall include:

- The engineer or engineers who wrote the test plan (names, titles, signatures, and dates), including evidence that each of the engineers concludes that the test plan is ready for review and approval
- The individuals responsible for approving and reviewing the test plan (names, titles, signatures, and dates), including any required reviews and approvals from the Toshiba organization, Toshiba's contractor, and from the customer, including evidence that the reviews and approvals were performed and the test plan accepted
- Each member of the testing staff (names, titles, signatures, and dates)
- Test item version/revision number, including record of revisions for the test plan
- Prerequisite tasks to be performed
- Definition of the testing requirements, including features to be tested, features not to be tested and a justification for each, and development of acceptance criteria for each test case and element in each test case
- Definition of the methods to be used to log test results and observations
- Definition of the methods and forms to be used for reporting of test anomalies
- Definition of the methods to be used to make minor changes to test procedures and test cases (e.g., pen-and-ink changes), as well as the methods to be used to make more significant changes to test procedures and test cases, including requirements for review and approval of such changes

- Methods to be used to start, hold, repeat, or restart testing within the test procedures, including setup requirements for such testing
- **References** to the following test item documentation, as applicable:
    - System Design Specification
    - Software Requirements Specification
    - Software Design Description
    - User Interface Specification
    - Hardware and Software Installation Procedures
    - System Operations and Maintenance Manual
    - Test considerations or needs
    - Test setup and staging requirements
    - Test environment needs, including facilities, power, communications, HVAC, tools, hardcopy documentation, special training, and qualified personnel
    - Self-diagnostics during system startup
    - Setup or staging requirements
    - Environment needs
- Methods for evaluating and accepting changes to the test, acceptance criteria, or requirements
- Identification of risks, including contingency plan for each

Multi-level test plans may be prepared, if necessary for complex software, to define the required test process. Lower-level plans shall reference the next higher-level test plan. Higher-level test plans should reference lower-level plans.

Test plans shall be placed under configuration management, in accordance with the Software Configuration Management Program Plan and each Software Configuration Management Plan.

9.4.2 Test Case and Test Procedure Specification

Test cases and test procedures shall be designed to ensure that the function requirements are satisfied, the performance requirements are achieved, and the user interface and other requirements are met. The test cases and test procedures shall be traceable to the requirements and acceptance criteria.

As a minimum, the test case and test procedure shall include the following:

- The software being tested along with the applicable version and revision data

- The hardware being tested, along with the applicable version and revision data
- Hardware and Software configuration used for the testing to be conducted
- Acceptance criteria
- User interface
- Facility identification
- Support software
- Each test case shall contain detailed information, including:
    - Purpose of the test case and the associated procedure(s)
    - Association, if applicable, with other test cases

The following resources shall be identified when required to execute the test case, including the appropriate references, if applicable:

- Documents
- Characteristics and configurations of hardware
- System application software and support software
- Special procedural requirements and constraints on the test procedure
- Test inputs, test cases, expected test output and results, and acceptance criteria for each test output and results
- Test acceptance criteria for signals having ranges of values shall provide an acceptable range of values, which shall be expressed in engineering units

Each procedure shall contain detailed information, including:

- Instructions on preparing and initiating testing
- Actions necessary during test execution, including:
    - Steps to shut down or suspend the test during a procedure, if applicable
    - Steps to deal with anomalous events that may occur during the procedure
    - Steps necessary to restart the procedure at each of restart points
    - Instructions for test measurements
- Pass/fail criteria

- Means to report and resolve errors
- Traceability to the requirements specifications

Test procedures and test cases shall be placed under configuration management, in accordance with the Software Configuration Management Program Plan and each Software Configuration Management Plan.

9.4.3 Test Report

The test report shall be prepared by the appropriate personnel to capture the test execution activities and test results.

At a minimum, the test report shall include:

- Summary of the test
- Software name and version/revision numbers
- The identity of (name, title, signature, and date):
    - The responsible tester or testers
    - All personnel present during test execution
    - The reviewers and approver of the report
- The daily test log to document the chronological events relevant to the test being conducted, which shall include:
    - Test Lead, Date and time testing is started.
    - Test case number for each test case performed.
    - Test Lead, date and time testing is halted.
    - Report of any significant events, equipment malfunctions, or personnel issues that affect test execution
    - Each event logged should include the event's occurrence start and end date and time.
    - Record of the visually observed test results in the designated location
    - Error messages generated
    - Requests for operator action
    - Anomalies or incidents observed
    - Report of differences in the test items from the test plan, test designs, or test procedures
    - Justification for each difference shall be included

- Evaluation of the completeness of the testing process against the criteria specified in the test plan
- Justification for each feature included in the test procedure but not tested

9.5 Test Incident Reporting

A Test Incident Report shall contain pertinent data associated with an identified problem, anomaly, or test correction that occurs during testing.

Test Incident Reports:

- Describe and evaluate the reported incident
- Describe the resolution for the reported incident
- Report the changes that are required in the software, test procedure, or test case, called Corrective Actions

Test Incident Reports shall contain:

- Report Identifier
- Summary of the incident
- Description of the incident, including related activities and observations that may help isolate and correct the cause of the incident
- Impact on test plans, test design specifications, test procedures, or test case specifications
- Test incident reporting shall be organized in accordance with the structure specified in IEEE Std. 829

Test incidents in all phases of testing considered in both the SVVPP and this program plan shall be recorded in the Corrective Action Program and resolved based on procedures found in SQAPP Section 5.

9.5.1 Corrective Actions

Corrective actions shall follow the requirements in the SQAPP in Section 5.

10 Software Training Program Plan (STrngPP)

10.1 Introduction

Training is a vital aspect of a software organization. Adequate training is necessary to support the development of safe, reliable, high quality software products and to support the long-term use and maintenance of those software products. This Software Training Program Plan (STrngPP) describes the software training activities to be carried out for staff responsible for design, development, review, and test of systems, as well as for plant staff responsible for operating, maintaining, calibrating, and surveillance testing the systems.

10.1.1 Purpose

This STrngPP provides a framework for software and system training activities to support the development and use of digital instrumentation and control systems. The Software Training Program Plan addresses the management, implementation, and resource characteristics required for a test program compliant to the customer requirements.

Each organization generating or purchasing software shall write a Software Training Plan (STrngP) specific to the system or equipment, or logical group of systems or equipment. Each STrngP shall be prepared, reviewed, approved, and retained as a quality record. Each Software Training Plan shall specify the following:

- The personnel roles and responsibilities for the training program as well as the training and qualification of the trainers and the staff generating the training materials;
- Activities to identify training needs of appropriate software development staff, including managers, developers, reviewers, testers, software quality assurance, maintainers, and other software staff not otherwise identified;
- Activities to identify training needs to enable use of supplied software tools, including training on software tool documentation
- Activities to identify training needs of plant staff, including Operations, Maintenance, and Engineering staff;
- Activities to document and maintain general training records; and
- Activities to ensure that, for a given software project, personnel are adequately trained for their roles in the software development process, recurring training is defined appropriately, untrained staff shall not be used for software activities, and training is documented.

10.1.2 Scope

Each STrngP applies to training activities specific to the development life cycle of software and system projects, general activities independent of a software development project, activities of installing and commissioning software systems in the nuclear power plant, and the operation and maintenance of

software once installed in the nuclear power plant. The plan applies to all personnel involved in the safe development, verification, use, and maintenance of software products.

This plan satisfies the requirements of Regulatory Guide 1.173, which endorses Institute of Electrical and Electronics Engineers (IEEE) Standard (Std.) 1074 Reference XX. IEEE Std. 1074 Clause A.1.2.6, "Plan Training," provides guidance for the implementation of a software training plan, which shall be extended to training on the systems to be supplied.

Each STrngP shall include the following.

- Software and Systems Training organization, including the organization and responsibilities for the training program,
- Software and Systems Training activities, including the identification of training needs, documentation of training records, and review of appropriate training, and
- Software and Systems Training resources, which addresses the utilization of tools, techniques, equipment, and personnel necessary for the implementation of the specified activities.

This STrngPP shall be implemented by each Toshiba organization and Toshiba's contractor supplying software-based equipment. Additional oversight shall be supplied for subcontractors by the Toshiba organization or Toshiba's contractor responsible for that subcontractor. Additional SQA oversight shall be supplied by the customer QA and I&C organizations, with additional support as deemed necessary by the customer.

10.1.3 [Deleted]

10.1.4 Relationship of the STrngPP to Other SPP Sections

In order to simplify the description of processes and ensure clarity in establishing roles and responsibilities and to align with Branch Technical Position 7-14 (**Reference** 3) guidance, the SPP has been split into several sections. Each section of this SPP describes the processes, roles, and responsibilities for the activities in that section. Each section either explicitly or implicitly refers to other sections as necessary to implement additional processes.

Thus, this STrngPP defines the methods to be used to train all staff involved in software life cycle activities, to ensure that the staff understand their roles and responsibilities and are able to implement their roles and responsibilities. The STrngPP uses other portions of the activities defined in Section 2 through 9, Section 11, Section 13, and Appendix C of this SPP. The project management functions defined in Section 12 are implemented in the customer plans, procedures, and instructions necessary to operate and maintain the nuclear power plants.

Thus, for each of the sections, the STrngPP ensures that the staff performing technical and management activities understands the requirements of their positions. The STrngPP interfaces with the other sections of this SPP as follows:

- All requirements provided in Section 1 of the SPP apply to the STrngPP, including system and software organization, roles and responsibilities, classification of systems and software, and the life cycle for system and software development.

- The SPMPP, Section 2, ensures that training and training records are current, correct, and that staff implementing various technical and management processes understand their responsibilities, including project management staff. This work is performed in accordance with the technical and process requirements listed in Section 10, Training.

- The SDPP, Section 3, makes use of trained staff from this STrngPP. Trained staff is required to perform the necessary activities in accordance with the technical and process requirements of Section 3, Development.

- The SVVPP, Section 4, also makes use of trained staff from the STrngPP. Trained staff is required to perform the V&V activities, based on the V&V organization's plans, procedures, and engineering instructions.

- The implementation of the training activities and the retention of appropriate training records based on plans, procedures, and engineering instructions is overseen by the software quality assurance organization, in accordance with the technical and process requirements listed in Section 5, Software Quality Assurance.

- The Software/System Safety organization also makes use of trained staff, based on the STrngPP. Training is necessary to ensure that the Software/System Safety work is performed in accordance with the plans, procedures, and engineering instructions defined in the technical and process requirements listed in Section 6, Software Safety.

- The change control and configuration management activities require trained staff to make use of the databases and other software used to maintain configuration. Staff training for change control and configuration management is performed in accordance with the STrngPP. The change control and configuration management activities are performed in accordance with the technical and process requirements listed in Section 7, Configuration Management.

- The STrngPP ensures that trained staff is provided to perform integration activities. Training is necessary to ensure that the integration work is performed correctly and completely, in accordance with the technical and process requirements listed in SPP Section 8, Software Integration.

- The STPP activities require use of trained personnel, based on the STrngPP. Training, and usually certification, is necessary to ensure that the testing work is performed correctly and completely, in accordance with the technical and process requirements listed in SPP Section 9, Testing.

- The STrngPP activities ensure that software installation at the vendor site and at the nuclear plant have appropriate documentation and procedures such that the installations can be controlled and performed in accordance with the technical and process requirements listed in Section 11, Installation.

- The customer Training Program ensures that appropriately trained staff is available at the customer for the normal, expected plant functions that include operation, surveillance, calibration, modifications to tunable constants and other configuration items, and troubleshooting in accordance with the technical and process requirements listed in Section 12, Operations. This training includes the aspects necessary from the STrngPP for software.

- The STrngPP will provide trained staff for the software life cycle phases required for any changes required to close anomaly reports and implement enhancements to plant systems in a controlled manner, in accordance with the technical and process requirements listed in Section 13, Maintenance.

- The STrngPP provides trained staff capable of performing the technical and process requirements listed in Appendix C, Secure Development and Operational Environment, for all safety systems and for other systems where cyber security requirements apply.

10.2 Software Training Overview

10.2.1 Organization

A typical organization for software projects is illustrated in Figure 1. Other organizations are acceptable as discussed in and subject to the constraints of Section 1.4.

The roles associated with Software Training are defined below.

10.2.2 Responsibilities

The personnel responsibilities relating to Software Training activities are as follows:

- Each Quality Assurance (QA) Manager shall be responsible for establishing requirements and planning for training and indoctrination of all personnel associated with the software life cycle (e.g., developers, testers, operators, etc.) and for ensuring that all training records are documented and archived. Responsible organizations, information sources, and the intended audiences shall be defined for each type of training. Training tools, techniques, and methodologies shall be specified. The planning shall include developing schedules, estimating resources, identifying special resources, staffing, and establishing exit or acceptance criteria.

- Each Project Manager is responsible for verifying that all required training has been administered and, if recurring training is required, training status is current. The Project Manager's approach to tracking the necessary training shall be documented (e.g., in a task specific Software Project Management Plan) and retained with the project records.

- Each Project Manager shall be responsible for ensuring that all project software personnel, whether a customer employee or a subcontractor, are adequately trained for their roles, or that training is included in the Software Project Management Plan (see Section 2) to provide an appropriate training level.

- Each Software Development Lead shall review the training record for each staff member to ensure that all development personnel are adequately trained any role assigned each staff member.

- Each Software Verification and Validation (V&V) Lead shall review training records to ensure that all verification and validation personnel are adequately trained for their V&V activities (e.g., testing, code review, etc.).

- The responsibilities of each Software Test Lead are defined in Section 1.4.3.

- Each Training Lead shall be responsible for ensuring that team members are cleared for access, as necessary, or coordinate with the appropriately cleared Cyber Security Team who already has access to cyber security information, which may require clearance for safeguards or Official Use Only documents and information.

- For all customer employees and contract staff, their administrative supervisor shall be responsible for ensuring that training goals set by management are met, and ensuring that training records are submitted to the appropriate the customer records-keeping authority. Each administrative supervisor shall be responsible for ensuring that only trained, qualified staff work on these systems. The customer Training Department shall be responsible for performing training, or for certifying that training offered by external suppliers is of acceptable quality to allow the customer Training Department to certify that people are trained adequately.

10.2.3 Schedule

Software training shall be scheduled and completed in time to meet project needs. Training shall be included in the specific project schedule for each system or logical group of systems.

All activities within this plan shall be planned and scheduled, in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

10.3 Training Activities

10.3.1 General Training Activities

Personnel shall maintain their proficiency by participating in technical seminars and classes that cover:

- Engineering and design work in progress that is of a new or unique nature. The primary purpose of this type of information is to pass on to all customer personnel lessons learned in the solution of engineering and design problems encountered in the design, construction, and operation of the nuclear facility. During these presentations, pertinent areas of Codes, Standards, and USNRC regulations and guidance, as they apply to the customer work, are discussed.

- All personnel responsible for the operation or maintenance of software products shall receive training in that system, to include the software aspects, with respect for their particular responsibility.

- Other appropriate technical and quality assurance topics.

Appropriate organizations, information sources, and the intended audiences shall be defined and used for each type of training. Responsibilities for ensuring completion of training shall be documented in each STrngP.

Each STrngP shall specify training tools, techniques, and methodologies.

The planning shall include developing schedules, estimating resources, identifying special resources, staffing, and establishing exit or acceptance criteria.

Records of both the planned training, and the completed training, including any performance measurements such as test scores, shall be reviewed by the appropriate administrative supervisor and submitted to the appropriate records retention process for archival.

The STrngP shall provide for recurrent training.

The STrngP shall support certification of those staff requiring certification.

10.3.2 Project Training Activities

Training shall include, but not be limited to, the following skills, as applicable to the job functions being performed:

- Software Development
- Software Management
- Software Operation
- Software Maintenance
- Design and Code Inspections
- Software testing

Training shall include, but not be limited to, following capabilities, as applicable to the job functions being performed:

- Any software tools to be used on the software project, including tools for troubleshooting, backups, restoring backups, and restarting parts of or the whole system
- Any specific software skills or technologies applicable to the software project.

If the Project Manager determines that additional training is necessary, provisions to acquire the additional training shall be added to each STrngP.

10.4 Methods and Tools

Training for applicable staff of the customer, EPC Team Members, and their subcontractors shall be provided for the tools selected in the Software Development Plan (Section 3.11.3.4).

Methods and tools used to perform software training shall be defined in the Software Development Plan (Section 3.11.3.4) for all project training as required for each software system or logical group of software systems. The responsible trainer shall determine the content and methods for each training course.

10.5 Training Facilities

Effective training requires effective training facilities to fulfill the training objectives. When preparing a training course, the trainer shall determine the type of training facility that provides effective training, and shall ensure that such facilities are used. Examples of effective training facilities are:

- Dedicated classroom space
- Instructor-led classroom software lab facilities
- Self-study computer lab facilities
- A remote training access tool (e.g., presentation tools) which allow training at a remote training workstation
- Control room simulator
- Equipment similar to that in the plant, configured for training

The training program should incorporate quizzes or practical exams based on course objectives relevant to the task responsibilities.

The training program should incorporate self-study for training.

10.6 Measurement and Metrics

Metrics provide a basis for determining the effectiveness of the training program. Metrics shall be selected during the development of the training program. Metrics shall reflect measurable results. Metrics shall be based on the nature of the training program being offered. For example, a training course providing a one-day overview session would utilize a different set of metrics than a four-week course that utilizes extensive use of simulation training tools.

Examples of training tool metrics are:

- Instructor Assessment, where the instructor queries trainees during class session and grades the daily performance of the trainees
- Certification exams
- Computer software lab tests
- Student performance during plant scenarios in a training simulator

The test results or training results obtained at the end of the training activities shall be recorded, analyzed, and reported.

Software staff shall be considered qualified based on their training and experience. As qualification is subjective, the consideration of a staff member as qualified for their job shall be written and signed by the Lead to which the individual reports, with review by the Training Lead, and approval by the Project Manager. Qualification records shall be quality records and retained for the life of the system.

11 Software Installation Program Plan (SInstPP)

11.1 Introduction

Software installation is the process of putting a completed and validated software product onto a hardware component so that it can be executed. This software installation can be completed for testing prior to delivery, or can be required for installation at the plant or after modification of the software after the system has been installed in the customer.

For each system or logical group of systems, a Software Installation Plan (SInstP) shall be written to address the performance of these high-level activities.

11.1.1 Purpose

This document shall define the preparation and content of project-specific Software Installation Plans (SInstPs). This Software Installation Program Plan (SInstPP) shall define methods and requirements for the preparation of the SInstP. Each SInstP shall be prepared, reviewed, approved, and retained as a quality record.

11.1.2 Scope

Each SInstP shall address software installation strategy and techniques. Each SInstP shall also define the activities and procedures for the creation of documentation necessary to install software in the systems.

This plan satisfies the requirements of Regulatory Guide 1.173, which endorses Institute of Electrical and Electronics Engineers (IEEE) Standard (Std.) 1074 (Reference 25). This plan provides the requirements from the customer for those systems where software can be installed in the field, including those systems where software replacement is based on replacing hardware modules.

This SInstPP shall be implemented by a Toshiba organization or Toshiba's contractor supplying software-based systems or equipment. Additional oversight shall be supplied for subcontractors by the Toshiba organization or Toshiba's contractor responsible for that subcontractor. Additional SQA oversight shall be supplied by the customer QA and I&C organizations, with additional support as deemed necessary by the customer.

11.1.3 [Deleted]

11.1.4 Relationship of the SInstPP to Other SPP Sections

In order to simplify the description of processes and ensure clarity in establishing roles and responsibilities and to align with Branch Technical Position 7-14 (**Reference** 3) guidance, the SPP has been split into several sections. Each section of this SPP describes the processes, roles, and

responsibilities for the activities in that section. Each section either explicitly or implicitly refers to other sections as necessary to implement additional processes.

Thus, this SInstPP defines the methods to be used to install the software into the hardware, at vendor locations, integrated test facility, and at site. This controlled installation process ensures that the software will be installed in a manner that provides the required functionality and the reproducible results, using other portions of the activities defined in Section 2 through 10, Section 13, and Appendix C of this SPP. The project management functions defined in Section 12 are implemented in the customer plans, procedures, and instructions necessary to operate and maintain the nuclear power plants.

Thus, for each of the sections, the SInstPP provides the technical activities necessary to install software in the individual systems and the integrated plant, as specified by the life cycle. The SInstPP interfaces with the other sections of this SPP as follows:

- All requirements provided in Section 1 of the SPP apply to the SInstPP, including system and software organization, roles and responsibilities, classification of systems and software, and the life cycle for system and software development.

- The SPMPP, Section 2, controls and oversees the activities performed by the installation staff in accordance with the technical and process requirements listed in this Section 2, Project Management.

- The SDPP, Section 3, works cooperatively with this SInstPP, as required to perform the necessary activities in accordance with the technical and process requirements of Section 3, Development, and Section 11, Installation.

- The SVVPP, Section 4, works cooperatively with the Installation organization, to ensure that the Installation instructions are clear, correct, and understandable as part of the V&V activities, based on the V&V organization's plans, procedures, and engineering instructions.

- The implementation of the SInstPP and SVVPP activities is overseen by the software quality assurance activities performed by the quality assurance organization, in accordance with the technical and process requirements listed in Section 5, Software Quality Assurance.

- The SInstPP works cooperatively with the Software/System Safety organization to ensure that the installation activities are performed in a manner that maximizes safety, ensures that no new safety concerns are added, and ensures that all actions performed have been verified and validated, and that all safety aspects can be tested appropriately. The Software/System Safety work is performed in accordance with the plans, procedures, and engineering instructions in accordance with the technical and process requirements listed in Section 6, Software Safety.

- The SInstPP activities coordinate with the change control and configuration management activities. Any changes required based on review or implementation of the installation instructions will be processed exactly like any other change in work products. The change control and configuration management activities are performed in accordance with the technical and process requirements listed in Section 7, Configuration Management.

- The SInstPP activities are based on the work products resulting from the system and software life cycle activities, which is controlled in accordance with the technical and process requirements listed in SPP Section 8, Software Integration.

- The SInstPP activities coordinate or perform (depending on the safety classification) system and software integration activities to prepare for testing activities, in accordance with the technical and process requirements listed in Section 9, Testing.

- The SInstPP activities use appropriately trained personnel during installation activities, and coordinate with generation of such materials to ensure that training materials and other manuals are reviewed to ensure that plant staff can be trained appropriately, in accordance with the technical and process requirements listed in Section 10, Training.

- There are no SInstPP activities or requirements associated with plant operations, as these will be controlled by the plans, procedures, and instructions provided by the customer for the normal, expected plant functions that include operation, surveillance, calibration, modifications to tunable constants and other configuration items, and troubleshooting in accordance with the technical and process requirements listed in Section 12, Operations. Reloading software or any other changes to software, including tunable parameters and setpoint values, will be performed based on plant work orders and procedures written for each system.

- The SInstPP will be invoked as required to reinstall any changes required to close anomaly reports and implement enhancements to plant systems in a controlled manner, in accordance with the technical and process requirements listed in Section 13, Maintenance.

- The SInstPP ensures that installation activities do not compromise security or safety and that these activities are performed in accordance with the technical and process requirements listed in Appendix C, Secure Development and Operational Environment, for all safety systems and for other systems where cyber security requirements apply.

11.2 Software Installation Overview

11.2.1 Organization

A typical organization for software projects is illustrated in Figure 1. Other organizations are acceptable as discussed in and subject to the constraints of Section 1.4.

11.2.2 Roles and Responsibilities

Personnel responsibilities for Software Installation activities are as follows:

- Each Project Manager shall be responsible for designating and identifying their Software Installation Lead in permanent project records.

- Each Software Installation Lead shall have overall responsibility for the planning and execution of all Software Installation activities.

- Each Software Installation Lead shall be responsible for ensuring that team members are cleared for access, as necessary, or coordinate with the appropriately cleared Cyber Security Team who already has access to cyber security information, which may require clearance for safeguards or Official Use Only documents and information.

- Each Software Verification and Validation (V&V) Lead shall be responsible for ensuring that installation activities are adequately verified by system testing and/or acceptance testing. The Software V&V Lead shall also be responsible for ensuring that all V&V activities are complete before installation or use at the customer.

- The Software Quality Assurance Engineer shall be responsible for ensuring that the requirements described in this document and the project-specific SInstP are fulfilled. If no Software Quality Assurance Engineer exists, the Project Manager shall be responsible for ensuring that the requirements described in this document and the project-specific SInstP are fulfilled.

- While several of the roles above can be combined, the V&V Lead, Software Safety Lead, and the Software Quality Assurance Engineer shall be independent, as defined Section 1.4.2 to ensure that the requirements in this plan are met. The Project Manager for Design and the Project Manager for V&V shall be independent as defined in Section 1.4.2.

11.2.3 Program Records

A project-specific SInstP shall be prepared that conforms to this program planning document for each system or logical group of systems. Each SInstP shall be prepared, reviewed, approved, and retained as a quality record.

The planning, installation reports, installation configuration records, and other work products resulting from performance of each SInstP shall be prepared, reviewed, approved, and retained as quality records.

11.2.4 Scheduling and Planning

All activities within this plan shall be planned and scheduled, in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

11.2.5 Resources

Each SInstP shall define any resources that will be required to support the software installation activities and installation validation.

11.2.6 Training

Each SInstP shall state any training requirements for any personnel participating in software installation, and will provide a plan for ensuring personnel are trained according to those requirements in accordance with the Software Training Program Plan (see Section 10).

11.2.7 Reviews

The SInstP will be peer reviewed before being approved by the Project Manager.

11.3 Software Installation Activities

11.3.1 Prepare the Software Installation Plan

A software installation procedure shall be produced. The software installation procedure may address a single logical, complete software entity (e.g., a software package), or may address the complete system or logical group of systems. A combined procedure may be produced for multiple packages within a single system, but each system or logical group of systems should have its own installation procedure.

As a separate document or as a part of each Operations and Maintenance (O&M) Manual, the initial installation procedure for each individual software package and for each system or logical group of systems shall be defined and documented. The installation procedure shall include:

- **Reference** to the Configuration Item list for the system, as a source of verification for hardware and software configuration for the system or logical group of systems
- The hardware installation procedure
- The procedure or procedures required to configure the system hardware to support system operation, including either reference to or the actual data
- The hardware installation verification and installation testing procedure or procedures
- A description of any hardware steps necessary to prepare for and/or enable software installation, as well as steps necessary when software installation is complete
- A description of any procedures and the procedure or procedures required to load or reload platform software along with configuration data for that software
- A description of the software installation procedure and the procedure or procedures
- Software installation methods and procedures
- Criteria used to determine the success or failure of the installation effort
- A checklist or sequence of steps that can be used to confirm that correct software is installed in the specific systems in accordance with the system design documents. The following is a sample list of items to be considered as part of the checklist:
    - Affected functions are inoperable and in a safe condition according to the plant's technical specifications and operational requirements before proceeding with installation.
    - The computer system is functional.
    - The sensors and actuators are functional.
    - All cards are present and installed in the correct slots.
    - The communication system is correctly installed.
    - The correct software versions are installed on the correct computers.

- Appropriate return-to-service testing has been successfully conducted before declaring the modified function operable.
- Installation configuration tables are complete.
- Special tools, methods, or techniques used to accomplish the installation function shall be identified.
- Installation tools shall be qualified with a degree of rigor and level of detail appropriate to the safety significance of the software utilizing the installation tools.
- Security provisions have been satisfied.
- Precautions to ensure personnel and plant safety have been identified.

11.3.2 Software Installation Reporting

A software installation report shall be produced for each software installation procedure upon the completion of the installation effort. A combined report may be produced for multiple packages within a single system. However, each system or logical group of systems should have its own installation report.

The installation report shall include the following installation activities as a minimum:

- Serial numbers or other identification for the hardware platform on which the software is installed
- Software revisions
- Circuit board revisions
- Any Cyclic Redundancy Code or checksum that may be displayed by the installed software
- Test results
- **References** to the reports and other documents that provide the results of testing
- Anomalies and other discrepancies discovered during installation and their resolution
- Any associated data sheets generated during the installation
- Installation test summary
- **Reference** to the verification of all user configurable parameter values, including default setting
- Completed Checklist

11.3.3 Installation Configuration Tables

Installation configuration tables shall be produced for each system or logical group of systems.

The configuration tables shall include the following items:

- Hardware configuration tables shall include all information necessary for the correct operation of the system and its associated plant functions. This shall include any as-supplied or as-purchased default settings along with the settings revised for this application used to test and accept the initial configuration.

- Software configuration tables shall include all information necessary for the correct operation of the system and its associated plant functions. This includes any as-supplied or as-purchased default settings along with the settings revised for this application used to test and accept the initial configuration.

- Installation configuration tables shall be consistent with the hardware specifications and the software specifications as described in the SRS, SwDD, software code, and software build documents.

- Safety-related software shall be required to provide traceability for each installed program element backward to the integrated software elements that created that installed program element.

- Each user configurable function shall be defined, along with each configurable mode. Each configurable mode shall include the function, safety, and security of the overall application. Alternatively, reference shall be provided to the separate document containing this information.

- The tables shall include all of the functional characteristics defined in the procedure section of the SInstP to ensure the software is correctly configured for the system. Alternatively, reference shall be provided to the separate document containing this information.

11.3.4 Operations and Maintenance Manuals

The system Operations and Maintenance (O&M) Manuals shall be produced for each system or logical group of systems. System O&M Manuals shall include installation details necessary to enable the end user to install the software on the system and to replace hardware components as needed.

Operations and Maintenance Manuals are described in Sections 12 and 13.

11.3.5 Training Manuals

The software system training manuals for each system or logical group of systems shall be produced. The software system training manuals are based on design documents and O&M manuals.

Training manuals are described in Section 3.14.3.2.

11.4 Methods And Tools

11.4.1 Installation Methods and Tools

Installation methods, tools, software, and hardware required to install the system and reinstall both hardware and software shall be defined in each system installation manual.

These tools shall be evaluated by the processes defined in the Software Development Plan (Section 3.11.3.4).

11.4.2 Software Archive Retrieval

Software Configuration Management shall meet the requirements of the Software Configuration Management Program Plan (SCMPP) and each Software Configuration Management Plan (SCMP) written for each system or logical group of systems. Where applicable, specific backup and recovery procedures shall be included in the maintenance section of the O&M manual.

11.4.3 Software Installation Test

Accompanying the system as a deliverable shall be a set of software installation test procedures and test cases. These may be provided as a separate test procedure or as part of the installation procedures for each software package. This set of tests, procedures, and tests cases shall be designed, documented, and performed in accordance with the Software Test Program Plan (STPP) and each Software Test Plan developed for each system or logical group of systems.

These procedures shall be retained by the customer in the maintenance procedures for each system, as necessary.

For some systems, software installation is not possible in the field. For these systems, software installation test procedures and test cases shall not be required.

11.4.4 Installation Documentation and Problem Reporting

Problems or issues encountered during the installation process shall be documented in an installation report and shall be entered into the Corrective Action Program (see Section 5.8).

11.4.5 Verification and Validation Methods

The installation phase outputs shall be verified in accordance with the Software Verification and Validation Program Plan (SVVPP) and each Software Verification and Validation Plan (SVVP) developed for each system or logical group of systems.

12 Software Operations Program Plan (SOPP)

12.1 Introduction

The Software Operations Program Plan (SOPP) describes the activities for software-based systems during operation of the customer. During the Operations phase, the behavior of all systems is monitored for faults, failures, and poor performance. Additional procedures and guidance for these activities are provided in the Operations portion of the O&M Manual, which should include User's Manuals and other required documentation from purchased off-the-shelf platforms or system components. Methods for monitoring and assessing these systems are part of the administrative procedure system, and include implementing the Maintenance Rule, Operator Rounds, and other normal administrative procedures in a US nuclear power plant.

During the Operations Phase, no software corrections or enhancements are performed. Corrections are made during the Maintenance phase, which may be entered at any time, controlled by the customer management and administrative procedures. This program plan defines the requirements for documenting the software and system level processes and activities used to operate the system or logical group of systems during plant operation in its intended operational environment. These processes and activities shall be described and defined in the Users' Manuals, which shall be contained in the System Operations and Maintenance (O&M) Manuals (see Section 3.14.3.1).

12.1.1 Purpose

This program plan defines guidance, requirements, and considerations for the content and for developing the system-, technology-, or vendor-specific System Operations and Maintenance (O&M) Manual (see Section 13 for Maintenance phase guidance, requirements, and considerations. Installation procedures are also included in the manual).

The Operations portion of the O&M Manual lists the general functions that the Operations, Maintenance, and Engineering groups will be expected to perform, and provides general information on troubleshooting faults and failures, as well as reporting and handling faults and failures.

For logical groups of systems, such as several systems provided in a common platform, a single O&M Manual encompassing all of the systems shall be provided, rather than individual manuals for each of the systems with large quantities of common content.

O&M Manuals may be provided as a single volume, or as a set of volumes. O&M Manuals should be provided in both paper and electronic form, for long-term maintenance and configuration control.

Each of the system-, technology-, or vendor-specific O&M Manuals shall conform to the guidance of this program plan, and shall provide system-, technology-, or vendor-specific details that provide sufficient data to verify that each O&M Manual fulfills the requirements of this program plan.

Each O&M Manual shall provide additional supplemental guidance for technology-specific details as necessary to support operation, surveillance, calibration, backup, and other Operations phase activities appropriate to the system's or systems' design and requirements.

The O&M Manual shall not be limited to software, firmware, and FPGA issues, but shall address all aspects of the system, to include, but not be limited to hardware, human-system interfaces, power supplies, software tools used for system maintenance and backup, and other normal Operational interactions with the system, in all plant and system modes.

This program plan provides the basis and requirements for establishing a set of formal plans, programs, procedures, and instructions for the nuclear power plant that will provide a basis for safe, reliable, uneventful operation of both power plants, based on the content of the set of O&M Manuals for all plant systems. This program plan shall remain in effect for the life of the nuclear power plant.

12.1.2 Scope

This program plan defines the requirements for documenting the software and system level processes and activities used to operate the system or logical group of systems during plant operation in its intended operational environment. These processes and activities shall be described and defined in the Users' Manuals, which shall be contained in the System Operations and Maintenance (O&M) Manuals (see Section 3.14.3.1).

This SOPP shall be implemented by each Toshiba organization or Toshiba's contractor supplying software-based equipment. Additional oversight shall be supplied for subcontractors by the Toshiba organization or Toshiba's contractor responsible for that subcontractor. Additional SQA oversight shall be supplied by the customer QA and I&C organizations, with additional support as deemed necessary by customer.

12.1.3 [Deleted]

12.1.4 Relationship of the SOPP to Other SPP Sections

In order to simplify the description of processes and ensure clarity in establishing roles and responsibilities and to align with Branch Technical Position 7-14 (**Reference** 3) guidance, the SPP has been split into several sections. Each section of this SPP describes the processes, roles, and responsibilities for the activities in that section. Each section either explicitly or implicitly refers to other sections as necessary to implement additional processes.

Thus, the SOPP provides the technical review and test activities necessary to implement the life cycle activities specified by the life cycle. The SOPP interfaces with the other sections of this SPP as follows:

- All requirements provided in Section 1 of the SPP apply to the SOPP, including system and software organization, roles and responsibilities, classification of systems and software, and the life cycle for system and software development.

- The Section 2, Project Management; Section 3, Development; Section 4, Verification and Validation; Section 6, Software Safety; Section 8, Software Integration; Section 9, Testing; Section 11, Installation; Section 13, Maintenance; and Appendix C, Secure Development and Operational Environment, are not invoked during the Operations phase.

- The Software Quality Assurance function verifies procedure compliance for configuration management and change control during plant operations, associated with changes of tunable

constants during plant operation in accordance with the technical and process requirements listed in Section 5, Software Quality Assurance. The function may also audit internal the customer as well as external vendor compliance with the required procedures.

- The SOPP activities do require use of the change control and configuration management activities to control changes to tunable constants during plant operation. The change control and configuration management activities are performed in accordance with the technical and process requirements listed in Section 7, Configuration Management. These activities are controlled by maintenance procedures that define the methods and processes to be used for these configuration changes.

- The SOPP activities use appropriately trained personnel while operating the plant, to ensure that training materials and other manuals are reviewed to ensure that plant staff can be trained appropriately, in accordance with the technical and process requirements listed in Section 10, Training.

12.2 Organization, Management, and Responsibilities

A typical organization of a software organization that supports plant operation is illustrated in Figure 1. Other organizations are acceptable as discussed in and subject to the constraints of Section 1.4 and staff responsibilities in Section 1.4.3.

12.3 Schedule

All activities within this plan shall be planned and scheduled, in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

12.4 Manual

Each O&M Manual shall be developed in accordance with the requirements outlined in this program plan, requirements and guidelines from the human factors engineering program, and the customer requirements.

Each O&M Manual shall be consistent with other system documentation, including the system design documentation, FSAR, and known properties of the operational environment where the system is required to operate. Consistent terminology, notation, and definitions shall be used throughout the O&M Manual. All supplied manuals shall conform to the same requirements.

All O&M Manual instructions and content shall be consistent, and shall be free of contradictions.

Abbreviations in each O&M Manual shall comply with the customer standard abbreviations.

The Operation Phase sections of each O&M Manual for each system or logical group of systems shall describe the:

- System functions

- Interface with other systems

- User interfaces and use of those interfaces for each category of user

- Hardware, associated documentation, and additional software required to operate the system
- Operational environment in which the system shall operate
- Variables in the physical environment that the software shall monitor, control, display, alarm, annunciate, and record
- Precautions and limitations that shall be observed during use of the system, for all categories of users, to avoid, or at least minimize, exposing the public, plant personnel, or plant equipment to hazards or security vulnerabilities

Each O&M Manual for each system or logical group of systems shall be informed by and reference the Technical Specification requirements for surveillance intervals and requirements.

12.4.1 Procedures

Procedures for activities shall be defined in each O&M Manual for the system. All procedures, including operation, maintenance, surveillance, calibration, and test, shall be evaluated for correctness, usability, and compliance with the intended use of the system.

During operation, all procedures shall identify the:

- Responsibilities of the operator, maintainer, and engineering staff
- Cyber security requirements for operating the system, including:
    - Controls needed to prevent unauthorized changes to hardware, software, and system parameters
    - Monitoring activities needed to detect penetration or attempted penetration of the system
    - Associated contingency plans and procedures shall be created to respond to penetrations
    - Controls need to prevent installation of unauthorized software systems such as PCs
    - Implementation of the customer cyber security program plan.
- Success or failure of an operating procedure by measuring, recording, analyzing, and reporting the error rate found during operation

At a minimum, the System Operations Manual shall include the following procedures to:

- Operate the system in all modes, which includes powering up, starting, operating, calibrating, surveillance, testing, stopping, and powering down the system
- Ensure that the system is operating correctly and calibrated
- Ensure the software state is consistent with the plant operating mode at all times
- Prepare and use the system to monitor critical plant parameters and perform the system's safety functions for safety related systems

- Prepare and use the system to control and monitor plant operations for nonsafety Group 1 and 2 systems

- Perform periodic or on-demand backups of the software, configuration, or historical plant data as recommended by the system vendor

The procedures shall be linked to a comprehensive list of operating modes with descriptions.

12.4.2 Problem Reporting

Problem reporting, recording, and recovery will be in accordance with a customer standard corrective action program. During activities performed by the Toshiba, Toshiba's contractors, or their subcontractors after to shipment of equipment to the customer, corrective actions at each of the Toshiba, Toshiba's contractor, or subcontractor sites will use the same problem reporting and corrective action program used during the activities prior to shipment of equipment to the customer, as defined in Section 5.8.

To support problem resolution, a comprehensive list of error messages, descriptions, fault or failure interpretation and possible means of resolving the error shall be provided in each O&M Manual, which may be separated by each application and for the platform, as appropriate. The list of platform error messages can be a comprehensive list of references to the platform-specific sections in the Operations and Maintenance Manual where platform error messages are provided.

12.5 Cyber Security

Each O&M Manual shall contain or reference external protected documents that contain sufficient information to ensure long-term means are provided and used to mitigate cyber security issues and ensure that the system has not been compromised.

Cyber Security issues are defined in a cyber security program plan. Detailed cyber security requirements shall be provided in each system's or logic group of systems' cyber security plan, which is likely to contain Safeguards or Official Use Only information. Safeguards and Official Use Only information shall not be incorporated into the Operations and Maintenance Manual.

The cyber security measures shall be appropriate to each system design and tailored based on the safety and risk associated with compromising the system.

During the Operations Phase, the customer shall verify that each system has not been compromised, by techniques that should include periodic testing and monitoring, review of system logs, and real-time monitoring as the customer determines to be necessary in their cyber security program plan, which might include:

- Analyze new and developing cyber security risks

- Determine appropriate defensive measures for applicable risks

- Analyze proposed system changes to ensure new cyber security vulnerabilities are not added to the plant

- Evaluate operating, maintenance, and engineering procedures associated with plant systems for correctness and usability

- Evaluate document storage and other administrative processes for vulnerabilities that might expose documents or software code, revealing system vulnerabilities.

Each Toshiba organization, Toshiba's contractor, their subcontractor, and their vendors shall be responsible for identifying and ensuring the cyber security of design documents, records, code, manuals, and other materials that could be used to compromise systems supplied to the customer.

12.6 Verification and Validation Methods

There are no verification or validation activities performed in this life cycle phase.

12.7 Measurement and Metrics

Based on data provided in each O&M Manual, the customer will determine appropriate metrics for the Operations Phase.

13 Software Maintenance Program Plan (SMaintPP)

13.1 Introduction

Maintenance is the process of maintaining and enhancing system performance after installation and acceptance in the customer. Maintenance includes repairing, managing, and implementing pre-planned solutions for nonconforming items and enhancements to software and systems for improving performance or controllability issues. All Maintenance phase activities use the software life cycles under which the systems were implemented.

13.1.1 Purpose

This program plan defines guidance, requirements, and considerations for developing the content and for developing the system-, technology-, or vendor-specific System O&M Manual (see Section 12 for Operations phase guidance, requirements, and considerations).

13.1.2 Scope

This program plan describes the activities for the system to modify, enhance, and maintain the systems and software once installed in the plant. The procedures and guidance for these activities is provided in the Maintenance portions of the O&M Manual.

The considerations provided in Section 12.1.1 for Operations also apply to the Maintenance content of the O&M Manual.

This SMaintPP shall be implemented by each Toshiba organization, Toshiba's contractor supplying software-based equipment. Additional oversight shall be supplied for subcontractors by the Toshiba organization or Toshiba's contractor responsible for that subcontractor. Additional SQA oversight shall be supplied by the customer QA and I&C organizations, with additional support as deemed necessary by the customer.

13.1.3 [Deleted]

13.1.4 Relationship of the SMaintPP to Other SPP Sections

Since the SMaintPP uses the entire SPP to perform the actions appropriate for software changes, no additional mapping to the SPP is provided.

13.2 Organization, Management, and Responsibilities

A typical organization for an organization maintaining software is provided as Figure 1. Other organizations are acceptable as discussed in and subject to the constraints of Section 1.4 and staff responsibilities in Section 1.4.3. Roles and responsibilities for the customer staff are in accordance with the customer programmatic plans, procedures, and instructions.

13.3 Schedule

All activities within this plan shall be planned and scheduled, in accordance with Section 1.11.6. The schedule shall be maintained in accordance with project requirements.

13.4 Manual

The base requirements for each O&M Manual are provided in Section 12.4.

The Maintenance Phase sections of each O&M Manual for each system or logical group of systems shall describe:

- Managing nonconforming items encountered in the field
- Modifying or installing revised software to repair nonconforming items
- Implementing pre-planned actions necessary to maintain performance
- Performing troubleshooting and repair activities for the hardware
- Calibration of hardware
- Surveillance and restoration from surveillance
- Precautions for surveillance and calibration to avoid undesired actions by other systems interfaced to this system
- Limitations and cautions for maintenance, surveillance, and calibration activities

The Maintenance portions of the O&M Manual shall, at a minimum:

- Provide procedures to update setpoint values, change alarm limits, and other similar activities that result in configuration changes
- Provide procedures for backup, restoration as well as modification, and installation of software updates and enhancements
- Provide procedures for troubleshooting
- Provide procedures for configuring and replacing modules, power supplies, and other field replaceable units
- List the general functions that the software maintenance organization will be expected to perform
- Provide specific procedures to accept work done on the system
- Describe the processes required to maintain the software
- Describe the configuration control required to maintain the delivered software

- Describe the facility required to maintain the software
- Describe the hardware, associated documentation, and additional software required to maintain the software
- Require periodic analysis and reporting of problems and their resolutions along with recommendations for improving operation
- Actions taken regarding recommendations must be reported
- Require as-built records to reflect any accepted deviations and justifications for acceptance

13.4.1 Procedures

Procedures for activities are defined in the Maintenance portion of the O&M Manual to be performed on the system, including maintenance of calibration and setpoint values.

Changes to setpoint values and other calibration data may be controlled by simpler procedures than used to control changes to software, as long as these data changes are performed to an appropriate maintenance procedure that provides an appropriate level of detail for making, verifying, and validating the changes.

All procedures shall identify the:

- Required cyber security, including the Secure Development and Operational Environment activities, for each maintenance activity
- Controls needed for maintenance activities and test equipment to prevent unauthorized changes to hardware, software, and system parameters
- Potential situations or actions that may introduce unauthorized changes must be addressed
- Requirements for data retention for all data associated with the maintenance effort along with analyses to determine the effectiveness of the maintenance effort
- Criteria to judge the success or failure of the maintenance effort by measuring, recording, analyzing, and reporting the error rate during maintenance activities
- Precautions and limitations during maintenance to avoid exposing personnel or the plant to hazards or security vulnerabilities

Each O&M Manual shall, at a minimum, include procedures for:

- Maintaining the hardware
- Maintaining the software
- Systems and software risk management during maintenance, with particular attention to risks that have the potential for compromising safety, functionality, reliability, availability, and cyber security
- Problem reporting, tracking, and handling (See Section 13.4.2)

- Installing the software, including test steps to confirm the software is installed correctly and that no errors have been introduced from any modifications
- Restoring previous software versions
- Backing up software
- Preventing, responding to, handling, and resolving security incidents such as contamination viruses, Trojan horses, or other nefarious additions

The procedures shall address appropriate loss scenarios and undesirable operations of plant systems. Procedures should include contingencies to ensure minimal disruption to critical services.

During maintenance, the same plans, procedures, processes, and activities shall be used for software corrections and for software enhancements as were used for system design and development. If the customer staff perform modifications, their training and qualifications shall be provided and maintained by the customer, based on the customer administrative procedures and the customer Training Department, and their work shall be performed under appropriately modified versions of the original plans, procedures, processes, and activities.

13.4.2 Problem Reporting and Handling

Problem reporting, recording, and recovery shall be in accordance with a customer standard corrective action program. Corrective actions at each of the Toshiba organizations, Toshiba's contractors, their subcontractor's, or their vendor's sites shall use the same problem reporting and corrective action program used during the vendor activities prior to shipment of equipment to the customer, as defined in Section 5.8.

Reported problems shall be evaluated to identify nonconforming items and the performance of corrective actions as described in Section XV and XVI of 10 CFR Part 50, Appendix B for safety systems.

Problems include, but are not limited to, software failures, misunderstandings by the operator, mistakes in documents, poor human factors design, and anything that causes the potential or actual failure of systems or operator actions.

Effective troubleshooting requires documentation of the following:

- State of the system at the beginning of the occurrence
- Time and date of occurrence
- Description of the problem
- Description of what was done to correct the problem.

the customer procedures shall be in place prior to shipment of equipment to site for handling nonconforming items. This shall include the identification, documentation, and evaluation, segregation where practical, and disposition of nonconforming items. Steps for the notification of affected organizations shall be included when applicable.

Evaluation of nonconforming items and corrective actions shall follow the requirements in 10 CFR 50 and 10 CFR Part 21.

13.5 Cyber Security

Each O&M Manual needs tocontain or provide references to external documents that provide sufficient information to ensure long-term means are provided and used to mitigate cyber security issues and ensure that the system has not been compromised.

Cyber Security issues will be defined in each Cyber Security Plan written for each system or logical group of systems. Each Toshiba organizations and Toshiba's contractgors shall ensure that each of their vendors are cognizant of, understand, and comply with these requirements. The Toshiba organization or Toshiba's contractor shall ensure that each of their vendor's programs implements the Secure Development and Operational Environment requirements provided in Appendix C. Detailed cyber security requirements shall be provided in each system's or logic group of systems' cyber security plan, which is likely to contain Safeguards or Official Use Only information. Each organization shall segregate Safeguards and Official Use Only information from all other system documentation, which will minimize the amount of safeguards information to maintain.

During the Maintenance Phases, software corrections or enhancements are performed. During the maintenance phase, the same life cycle plans, procedures, processes, and activities shall be used for software corrections and for software enhancements as were used to design, develop, review, test, and install the system or logical group of systems. These plans shall maintain cyber security and ensure that modifications do not significantly decrease or compromise cyber security, including the aspects defined in the Secure Development and Operational Environment.

The customer may migrate the system to new hardware, or replace the system with a system from a different supplier or from the original supplier. The migration and replacement activities may be performed with assistance from subcontractors and suppliers. During these activities, the customer shall ensure that elements of the system removed are destroyed or otherwise processed in a manner that would not reveal vulnerabilities for other users of similar systems. Migration and replacement activities shall be treated as new system design activities, and elements of this Software Program Plan shall be updated and applied to the new system design, development, review, test, installation, and commissioning.

13.6 Verification and Validation Methods

Configuration changes shall be verified and validated in accordance with the processes provided in each O&M Manual.

Changes to software shall be verified and validated for installation and operation in the plant in accordance with either the original or a revised Software Verification and Validation Plan specific to the system or logical group of systems and verified to address the actions to be performed in the plant.

13.7 Measurement and Metrics

Measurements and metrics shall be developed in accordance with the customer's maintenance rule. Systems not subject to the Maintenance Rule may also apply similar metrics. the customer should consider whether metrics for the Maintenance Phase are the same as metrics for the Operations Phase.

A Terms and Definitions

Acceptance Criteria – The criteria that a system or component must satisfy in order to be accepted by a user, customer, or other authorized entity. [IEEE Std. 610.12]

Acceptance Testing – Formal testing conducted to determine whether or not a system satisfies its acceptance criteria and to enable the customer to determine whether or not to accept the system. [IEEE Std. 610.12]

Algorithm – A finite set of well-defined rules for the solution of a problem in a finite number of steps. [IEEE Std. 610.12]

Anomaly – Anything observed in the documentation or operation of software that deviates from expectations based on previously verified software products or reference documents. [IEEE Std. 610.12]

Application Software – Software designed to fulfill specific needs of a user. [IEEE Std. 610.12]

Architectural Design – The process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system. [IEEE Std. 610.12]

Architecture – The organization structure of a system or component. [IEEE Std. 610.12]

Baseline – Items that have been formally reviewed and agreed upon, that thereafter serve as the basis for further development, and that can be changed only through formal change control procedures. [IEEE Std. 610.12]

Branch Testing – Testing designed to execute each outcome of each decision point in a computer program. [IEEE Std. 610.12]

Build – An operational version of a system or component that incorporates a specified sub set of the capabilities that the final product will provide. [IEEE Std. 610.12]

Can – is used for statements of possibility and capability, whether material, physical, or causal (*can* equals *is able to*). [IEEE "2007 IEEE Standards Style Manual, Section 5, Clause 13.1]

Certification – A written guarantee that a system or component complies with its specified requirements and is acceptable for operational use. [IEEE Std. 610.12]

Code – Computer instructions and data definitions expressed in a programming language or in a form output by an assembler, compiler, or other translator. [Definition (1) from IEEE 610.12]

Code Review – A meeting at which software code is presented to project personnel, managers, users, customers, or other interested parties for comment or approval. [IEEE Std. 610.12]

Coding – The process of expressing a computer program in a programming language. [IEEE Std. 610.12]

Compiler – A computer program that translates program expressed in a high order language into their machine language equivalents. [IEEE Std. 610.12]

Component – One of the parts that make up a system. A component may be hardware of software and may be subdivided into other components. [IEEE Std. 610.12]

Computer Language – A language designed to enable humans to communicate with computers. [IEEE Std. 610.12]

Configuration Control – An element of configuration management, consisting of the evaluation, coordination, approval or disapproval, and implementation of changes to configuration items after formal establishment of their configuration identification. [IEEE Std. 610.12]

Configuration Item – An aggregation of hardware, software, design documents or procedures that is designated for configuration management and treated as a single entity in the configuration management process. [IEEE Std. 610.12]

Commercial-Off-The-Shelf – In this Software Program Plan, this term (COTS) is defined to be software purchased from a vendor, which is not necessarily modified to support plant requirements, but may be configured to support plant requirements. COTS may also be modified to support plant requirements. This definition does not vary for safety or nonsafety life cycles. [This Software Program Plan, based on the definition of a commercial grade item from 10 CFR 21.3]

Completeness – Those attributes of the planning documents, implementation process documents, and design outputs that provide full implementation of the functions required of the software. The functions which the software is required to perform are derived from the general functional requirements of the safety system, and the assignment of functional requirements to the software in the overall system design. [BTP 7-14, Section A.3.2.2]

As an example for one type of document: A Software Requirements Specification is complete if, and only if, it includes the following elements:

a) All significant requirements, whether relating to functionality, performance, design constraints, attributes, or external interfaces. In particular, any external requirements imposed by a system specification should be acknowledged and treated.

b) Definition of the responses of the software to all realizable classes of input data in all realizable classes of situations. Note that it is important to specify the responses to both valid and invalid input values.

c) Full labels and references to all figures, tables, and diagrams in the Software Requirements Specification and definition of all terms and units of measure. [IEEE Std. 830-1998, Clause 4.3.3]

Consistency – The degree of freedom from contradiction within a single document or among the different documents and components. There are two aspects to consistency. Internal consistency denotes the consistency within the different parts of a component for example, a software design output is internally consistent if no set of design elements are mutually contradictory. External consistency denotes the consistency between one component and another for example, software requirements and the resulting code are consistent with one another if there are no contradictions between the requirements and the code. [BTP 7-14, Section A.3.2.2]

As an example for one type of document: A Software Requirements Specification is internally consistent if, and only if, no subset of individual requirements described in it conflict.

The three types of likely conflict in a Software Requirements Specification are as follows:

a) The specified characteristics of real-world objects may conflict. For example,

   1) The format of an output report may be described in one requirement as tabular but in another as textual.

   2) One requirement may state that all lights shall be green while another may state that all lights shall be blue.

b) There may be logical or temporal conflict between two specified actions. For example,

   1) One requirement may specify that the program will add two inputs and another may specify that the program will multiply them.

   2) One requirement may state that "A" must always follow "B," while another may require that "A and B" occur simultaneously.

c) Two or more requirements may describe the same real-world object but use different terms for that object. For example, a program's request for a user input may be called a "prompt: in one requirement and a "cue" in another. The use of standard terminology and definitions promotes consistency. [IEEE Std. 830-1998, Clause 4.3.4.1]

Correctness – The degree to which a design output is free from faults in its specification, design, and implementation. There is considerable overlap between correctness properties and properties of other characteristics such as accuracy and completeness. [BTP 7-14, Section A.3.2.2]

As an example for one type of document: A Software Requirements Specification is correct if, and only if, every requirement stated therein is one that the software shall meet. [IEEE Std. 830-1998, Clause 4.1]

Deprecated – Indicates a term or definition whose use is discouraged because such use is obsolete, misleading, or ambiguous. [IEEE 610.12-1990]

Design Phase – The phase in the software life cycle during which the designs for architecture, software components, interfaces, and data are created, documented, and verified to satisfy requirements. [IEEE Std. 610.12]

Documentation – A collection of documents on a given subject. [IEEE Std. 610.12]

Error – An incorrect step, process, or data definition. [IEEE Std. 610.12]

Firmware – The combination of a hardware device, software, and data that are incorporated into the hardware device. Firmware shall be treated as software in a programmable device. [draft IEEE Std. P7-4.3.2]

Functional Testing – Testing conducted to evaluate the compliance of a system or component with specified functional requirements. Such testing that ignores the internal mechanism of a system or

component and focuses solely on the outputs generated in response to selected inputs and execution conditions. [IEEE Std. 610.12]

Hazard – A condition that is a prerequisite to an accident. [IEEE Std. 1228-1994]

Implementation Phase – The period of time in the software life cycle during which a software product is created from design documentation and debugged. [IEEE Std. 610.12]

Independent Verification and Validation (IV&V) – Verification and Validation performed by an organization that is technically, managerially, and financially independent of the organization. [IEEE Std. 610.12]

Informative – Informative text is provided for information only and is therefore not officially part of the standard. [IEEE Standards Style Manual, January 2007]

Installation Phase – The period of time in the software life cycle during which a software product is integrated into its operational environment and tested in this environment to ensure that it performs as required. [IEEE Std. 610.12]

Integration Testing – Testing in which software components, hardware components, or both are combined and tested to evaluate the interaction between them. [IEEE Std. 610.12]. Integration testing is testing for the purpose of finding (a) flaws associated with the interface between two or more software and hardware units, or (b) flaws resulting from an interaction between two units. [This Software Program Plan]

Interface – A shared boundary across which information is passed. [IEEE Std. 610.12]

Interface Requirement – A requirement that specifies and external item with which a system or system component must interact, or that sets forth constraints on formats, timing, or other factors caused by such an interaction. [IEEE Std. 610.12]

May – is used to indicate a course of action permissible within the limits of the standard (*may* equals *is permitted to*). [IEEE "2007 IEEE Standards Style Manual, Section 5, Clause 13.1]

Metric – A quantitative measure of the degree to which a system, component, or process possesses a given attribute. [IEEE Std. 610.12]

Module – 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. [IEEE Std. 610.12]

Must – is deprecated and shall not be used when stating mandatory requirements; *must* is used only to describe unavoidable situations. [IEEE "2007 IEEE Standards Style Manual, Section 5, Clause 13.1]

Normative – Normative text means information that is required to implement the standard and is therefore officially part of the standard. [IEEE Standards Style Manual, January 2007]

Operations and Maintenance Phase – The period of time in the software life cycle during which a software product is employed in its operational environment, monitored for satisfactory performance, and modified as necessary to correct problems or to respond to changing requirements. [IEEE Std. 610.12]

Package – A separately compliable software component consisting of related data types, data objects and subprograms. [IEEE Std. 610.12]

Path Testing – Testing designed to execute all or selected paths through a computer program. [IEEE Std. 610.12]

Previously Developed Software - In this Software Program Plan, this term (PDS) is defined to be software that a vendor wrote, or purchased from another vendor, at an earlier date, which might be used as-is, or more likely will be modified to support plant requirements. PDS can be implemented under an Appendix B program or a commercial software program. This definition does not vary for safety or nonsafety life cycles. Commercial-off-the-shelf (COTS) software is a subset of PDS. [This Software Program Plan]

Procedure – A written description of a course of action to be taken to perform a given task. [IEEE Std. 610.12]

Process – A sequence of steps performed for a given purpose, e.g., the software development process. [IEEE Std. 610.12]

Programmatic – Of, having, advocating, resembling, or following a plan, policy or program [Webster's Unabridged Dictionary]

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 Std. 610.12]

Regression Testing – Selective retesting of a system or component to verify that modifications have not caused unintended effects and that the system or component still complies with its specified requirements. [IEEE Std. 610.12]

Release –The formal notification and distribution of an approved version for use. [This Software Program Plan]

Requirement – A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard specification, or other formally imposed documents. [IEEE Std. 610.12]

Requirements Phase – The period of time in the software life cycle during which the requirements for a software product are defined and documented. [IEEE Std. 610.12]

Requirements Analysis – The process of studying user needs to arrive at a definition of system, hardware, or software requirements. [IEEE Std. 610.12]

Requirements Traceability – The process of following the relationship between two or more products of the development process; for example, a requirements traceability matrix that records the relationship between the requirements and the design of a given software component. [IEEE Std. 610.12]

Retirement – Permanent removal of a system or component from its operational environment. [IEEE Std. 610.12]

Retirement Phase – The period of time in the software life cycle during which support for a software product is terminated. [IEEE Std. 610.12]

Risk – A measure that combines both the likelihood that a system hazard will cause an accident and the severity of that accident. [IEEE Std. 1228-1994]

Shall – is used to indicate mandatory requirements strictly to be followed in order to conform and from which no deviation is permitted (*shall* equals *is required to*). [IEEE "2007 IEEE Standards Style Manual," Section 5, Clause 13.1]

Should – is used to indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain course of action is deprecated but not prohibited (*should* equals *is recommended that*). [IEEE "2007 IEEE Standards Style Manual," Section 5, Clause 13.1]

Simulation – A model that behaves or operates like a given system when provided a set of controlled inputs. [IEEE Std. 610.12]

Software – Computer programs, procedures, and possibly associated documentation and data pertaining to the operation of a computer system. [IEEE Std. 610.12] Software includes firmware. [IEEE Std. 7-4.3.2] Complex logic embedded in devices such as Field Programmable Gate Arrays (FPGAs) and Complex Programmable Logic Devices (CPLDs) shall use the processes defined for the software life cycle for logic development, unless the devices can be 100% tested. As defined in this Software Program Plan, devices implementing stored program sequencers shall also use the processes defined for the software life cycle for logic development. [This Software Program Plan]

Software Design Description – 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 Std. 610.12]

Software Development Cycle – The period of time that begins with the decision to develop a software product and ends when the software is delivered. This cycle typically includes a requirements phase design phase, implementation phase, test phase, and sometimes, installation and checkout phase. [IEEE Std. 610.12]

Software Development Process – The process by which user needs are translated into a software product. The process involves translating user needs into software requirements, transforming the software requirements into design, implementing the design in code, testing the code, and sometimes, installing and checking out the software for operational use. [IEEE Std. 610.12]

Software Installation – The process of putting a completed and validated software product onto a hardware component so that it can be executed. This may refer to the installation of software onto hardware for the integration activity during development, or may refer to the installation of the software system into the plant. [This Software Program Plan]

Software Item – Source code, object code, job control code, control data, or a collection of these items [IEEE Std. 610.12] or, of similar items associated with firmware or logic as defined in "Software" above. [This Software Program Plan]

Software Life Cycle – The period of time that begins when a software product is conceived and ends when the software is no longer available for use. [IEEE Std. 610.12]

Software Requirements Specification – Documentation of the essential requirements (functions, performance, design constraints, and attributes) of the software and its external interfaces. [IEEE Std. 610.12]

Statement Testing – Testing designed to execute each statement or a computer program. [IEEE Std. 610.12]

Stress Testing – Testing conducted to evaluate a system or component at or beyond the limits of its specified requirements. [IEEE Std. 610.12]

Style – The form and structure of a planning document, implementation process document or design output. Document style refers to the structure and form of a document. This has connotations of understandability, readability, and modifiability. Programming style refers to the programming language characteristics of the software and programming techniques which are mandated, encouraged, discouraged, or prohibited in a given implementation. [BTP 7-14, Section A.3.2.2]

Subcontractor – In this Software Program Plan, this term is defined to be a software, systems, component, or other equipment vendor who is contracted by a Toshiba organization, or Toshiba's contractor, or by the customer (during and after commercial operation) to perform a specific scope for the customer project. All organizations within Toshiba supplying systems are defined as subcontractors. [This Software Program Plan]

Support Software – Software that aids in the development or maintenance of other software; for example, compilers, loaders, and other utilities. [IEEE Std. 610.12]

System or Logical Group of Systems – In this Software Program Plan, this term is taken to mean the largest possible grouping of systems from a single vendor that can logically be made part of a single set of software plans. [This Software Program Plan]

System Testing – Testing conducted on a complete, integrated system to evaluate the system's compliance with its specified requirements. [IEEE Std. 610.12]

Test Case – A set of test inputs, execution conditions, and expected results developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement. [IEEE Std. 610.12]

Test Item – A software item which is an object of testing. [IEEE Std. 610.12]

Test Log – A chronological record of all relevant details about the execution of a test. [IEEE Std. 610.12]

Test Objective – An identified set of software features to be measured under specified conditions by comparing actual behavior with the required behavior described in the software documentation. [IEEE Std. 610.12]

Test Phase – The period of time in the software life cycle during which the components of a software product are evaluated and integrated, and the software product is evaluated to determine whether or not requirements have been satisfied. [IEEE Std. 610.12]

Test Plan – A document describing the scope, approach, resources, and schedule of intended test activities. It identifies test items, the features to be tested, the testing tasks, who will do such task, and any risks requiring contingency planning. [IEEE Std. 610.12]

Traceability – The degree to which each element of work products created or transformed during a given phase of the life cycle can be traced forward to one or more elements of a successor life cycle work product, and can be traced backwards to one or more elements of a predecessor life cycle work product. Traceability is central to the production of complex systems to ensure all requirements are implemented, checked, and tested. [Adapted from BTP 7-14, Section A.3.2.2]

As an example for one type of document: A Software Requirements Specification is traceable if the origin of each of its requirements is clear and if it facilitates the referencing of each requirement in future development or enhancement documentation. The following two types of traceability are recommended:

a) Backward traceability (i.e., to previous stages of development). This depends upon each requirement explicitly referencing its source in earlier documents.

b) Forward traceability (i.e., to all documents spawned by the Software Requirements Specification). This depends upon each requirement in the SRS having a unique name or reference number. [IEEE Std. 830-1998, Clause 4.8]

Traceability Matrix – A matrix that records the relationship between two or more products of the development process; for example, a matrix that records the relationship between the requirements and the design of a given software component. [IEEE Std. 610.12]

Unambiguity – The degree to which each element of a product, and of all elements taken together, have only one interpretation. [BTP 7-14, Section A.3.2.2]

As an example for one type of document: A Software Requirements Specification is unambiguous if, and only if, every requirement stated therein has only one interpretation. As a minimum, this requires that each characteristic of the final product be described using a single unique term. In cases where a term used in a particular context could have multiple meanings, the term should be included in a glossary where its meaning is made more specific. [IEEE Std. 830-1998, Clause 4.3.2]

Unit Testing – Testing of individual hardware or software units or groups of related units. [IEEE Std. 610.12]

User Interface – An interface that enables information to be passed between a human user and hardware or software components of a computer system. [IEEE Std. 610.12]

Vendor – In this Software Program Plan, this term means someone other that the customer, a Toshiba organization, a Toshiba's contractor, a subcontractor to a subcontractor to a Toshiba's contractor, or an organization within Toshiba. [This Software Program Plan]

Verifiability – The degree to which a software planning document, implementation process document or design output is stated or provided in such a way as to facilitate the establishment of verification criteria and the performance of analyses, reviews, or tests to determine whether those criteria have been met. [BTP 7-14, Section A.3.2.2]

As an example for one type of document: A Software Requirements Specification is verifiable if, and only if, every requirement stated therein is verifiable. A requirement is verifiable if, and only if, there exists some finite cost-effective process with which a person or machine can check that the software product meets the requirement. In general, any ambiguous requirement is not verifiable. Non-verifiable requirements include statements such as "works well," "good human interface," and "shall usually happen." These requirements cannot be verified because it is impossible to define the terms "good," "well," or "usually." The statement that "the program shall never enter an infinite loop" is non-verifiable because the testing of this quality is theoretically impossible. [IEEE Std. 830-1998, Clause 4.3.6]

Verification and Validation – The process of determining whether the requirements for a system or component are complete and correct, the products of each development phase fulfill the requirements or conditions imposed by the previous phase, and the final system or component complies with specified requirements. [IEEE Std. 610.12]

Will – is deprecated and shall not be used when stating mandatory requirements; *will* is only used in statements of fact. [IEEE "2007 IEEE Standards Style Manual," Section 5, Clause 13.1]

Work Package – A specification of the work that must be accomplished to complete a work task. A work package should have a unique name and identifier, preconditions for initiating the work, staffing requirements, other needed resources, work products to be generated, estimated duration, risks factors, predecessor and successor work tasks, any special considerations for the work, and the completion criteria for the work package – including quality criteria for the work products to be generated. [IEEE Std. 1058-1998]

B Acronyms

ABWR – Advanced Boiling Water Reactor

BIOS – Basic Input/Output System

BR – Baseline Review

BRR – Baseline Review Report

BRT – Baseline Review Team

BTP – Branch Technical Position

CCB – Configuration Control Board

CFR – Code of Federal Regulations

CI – Configuration Item

COL – Combined License

COLA – Combined License Application

COTS – Commercial-Off-The-Shelf

CPI – Cost Performance Index

CPLD – Complex Programmable Logic Device

CSA – Cyber Security Analysis

CST – Cyber Security Team

CSPP – Cyber Security Program Plan

D3 – Diversity and Defense-In-Depth

DCIS – Distributed Control and Information System

DI&C – Digital Instrumentation and Controls

DIV – Design Installation Verification

EDS – Engineering Design Specification

EPRI – Electrical Power Research Institute

FMEA – Failure Modes and Effects Analysis

FPGA – Field Programmable Gate Array

FSAR – Final Safety Analysis Report

FT – Fault Tree

HFE – Human Factors Engineering

HICRc – Highly-Integrated Control Rooms – Communications Issues

HVAC – Heating, Ventilation, and Air Conditioning

I&C – Instrumentation and Controls

IBR – Incorporated by **Reference**

I/O – Inputs and Outputs

IBD – Interlock Block Diagram

IED – Instrument Equipment Diagram

IEEE – Institute of Electrical and Electronics Engineers

INPO – Institute of Nuclear Power Operations

ISCPS – Intra System Communication Protocol Specification

ISO –International Organization for Standardization

IV – Independent Verification (and Validation)

MCLD – Modulating Control Logic Diagram

NEI – Nuclear Energy Institute

NRW – Non Re-writable

NSSS – Nuclear Steam Supply System

OEM – Original Equipment Manufacturer

OUO – Official Use Only

O&M – Operation and Maintenance

P&ID – Piping and Instrumentation Diagram

PC – Personal Computer

PDN – Plant Data Network

PDS – Previously Developed Software

PFT – Platform Factory Test

PHA – Preliminary Hazards Analysis

PIE – Postulated Initiating Events

PIT – Platforms Integration Test

PM – Project Manager

QA – Quality Assurance

RG – Regulatory Guide

RTM – Requirements Traceability Matrix

SAD – System Architecture Description

SBPR – Software Build Procedure and Report

SCM – Software Configuration Management

SCMP – Software Configuration Management Plan

SCMPP – Software Configuration Management Program Plan

SDD – System Design Document

SDOE – Secure Development and Operational Environment

SDOEA – Secure Development and Operational Environment Analysis

SwDD– Software Design Description

SDP – Software Development Plan

SDPP – Software Development Program Plan

SER – Safety Evaluation Report

SIL – Software Integrity Level

SInstP – Software Installation Plan

SInstPP – Software Installation Program Plan

SIntP – Software Integration Plan

SIntPP – Software Integration Program Plan

SID – Software Interfaces Document

SMaintP – Software Maintenance Plan

SMaintPP – Software Maintenance Program Plan

SMA – Software Modification Analysis

SPMP – Software Project Management Plan

SPMPP – Software Project Management Program Plan

SMR – Software Modification Request

SOP – Software Operations Plan

SOPP – Software Operations Program Plan

SPI – Schedule Performance Index

SPP – Software Program Plan

SQAP – Software Quality Assurance Plan

SQAPP – Software Quality Assurance Program Plan

SRR – Software Release Report

SRS – Software Requirements Specification

SSA – Software Safety Analysis

SSP – Software Safety Plan

SSPP – Software Safety Program Plan

SSL – Software Safety Lead

SSA – Software Safety Analysis

SST – Software Safety Team

STPP – Software Test Program Plan

STrngP – Software Training Plan

STrngPP – Software Training Program Plan

SVT – Software Validation Testing

SVVP – Software Validation and Verification Plan

SVVPP – Software Validation and Verification Program Plan

SyRD – System Requirements Specification

TBD – To Be Determined

TBV – To Be Verified

TR – Technical Report

TT – Integration and Integration Test Team

USNRC – United States Nuclear Regulatory Commission

V&V – Verification and Validation

VITP – Validation and Integration Test Plan

WBS – Work Breakdown Structure

C Secure Development and Operational Environment

This normative appendix provides guidance for the Secure Development and Operational Environment (SDOE) considerations that shall be applied to all safety, nonsafety Group 1, and nonsafety Group 2systems, without regard to their attachment to the Plant Data Network. This Appendix also applies to equipment that attaches to plant equipment, such as Information Technology equipment, which has interfaces that might be vulnerable to cyber security penetration. This appendix provides the regulatory requirements expressed in Regulatory Guide (RG) 1.152, Regulatory Positions 2.1 through 2.5, Revision 3, as modified by the United States Nuclear Regulatory Commission (USNRC) in Digital Instrumentation and Controls Interim Staff Guidance 4 (DI&C ISG-04) for communications (**Reference** 34).

All vendors shall implement these requirements for SDOE. Implementation of appropriate and necessary SDOE plans and programs at each vendor's location has the goal of maximizing reliability, eliminating security vulnerabilities, and thus creating systems, structures, and components (SSCs) containing software that support the long-term requirements of the customer cyber security plan. Thus, each SSC containing software shall be implemented to support the customer cyber security program without compensatory measures. Previously developed or commercial-off-the-shelf software shall minimize the compensatory measures required for compliance with the customer cyber security program, and preferably eliminate such compensatory measures. The vendor's security process shall address the SSC's complete life cycle, including those portions for which the customer is responsible, including Installation, Operation, Maintenance, and Retirement.

The digital safety system development process should address potential security vulnerabilities in each phase of the digital safety system life cycle, to ensure reliable operation of each SSC containing software, and thus reliable operation of the complete nuclear power plant. While this SPP uses a waterfall model, that model is provided only as a framework for describing computer security requirements. Security requirements specific to the life cycle phase should be commensurate with the risk and magnitude of the harm resulting from unauthorized and inappropriate access, use, disclosure, disruption, or destruction of the digital safety system.

When incorporating security information into documents, details shall not be included that result in classifying software documents as safeguards or Official Use Only information, based on the information contained in the document.

This Appendix provides the design engineering requirements that provide a customer secure development and operational environment and a customer cyber security program.

Throughout the digital SSC's life cycle, the vendor shall have an active digital security program. The vendor's security policies, standards, and procedures shall ensure that the vendor's design basis information shall be protected in a manner that shall not compromise the security of the digital system, other systems, or the plant. The vendor shall perform a security assessment, which includes a risk assessment, to identify the potential security vulnerabilities caused by installation of the digital SSC, both as an individual SSC and for any security impact that SSC would have on the nuclear power plant. The risk assessment shall include an evaluation of new security constraints in their networks and operations; an assessment of any proposed changes and the impact of such changes on security; and an evaluation of the vendor's operating and design procedures for correctness and usability. The results of

these assessments shall provide a technical basis for establishing and confirming security levels for the SSCs and the plant.

The vendor's security program and the SSC containing software delivered by that vendor shall be designed, documented, implemented, tested, and reviewed to ensure that the SSC support the customer cyber security program through the Installation, Operation, Maintenance, and Retirement Phases of each SSC's life cycle. The requirements for each phase are listed below.

1. Planning Phase

While system conceptual designs are being determined, the customer and the SSC suppliers shall perform security assessments to identify potential security vulnerabilities in SSC designs throughout the life cycle. These assessments shall evaluate the potential susceptibility to security vulnerabilities against the effects of inadvertent access and undesirable behavior from connected SSC over the course of the system's life cycle that could degrade the SSC's reliable operation. These assessments shall identify the potential operational security vulnerabilities of the digital SSC and the vulnerabilities to the SSC's development life cycle phases. The results of the analysis shall be used to establish security requirements for the SSC's hardware, software and integrated design and protective measures for the development environment. the customer and the system suppliers shall identify security capabilities that shall be implemented.

Remote access to the safety SSCs shall not be implemented. Remote access to the nonsafety SSCs should not exist. Communication to other equipment outside the vital and protected areas shall be minimized, and each such communication link analyzed, with appropriate protection defined, documented, and implemented.

Digital SSC shall transfer data to nonsafety SSC through one-way communication pathways, unless requirements dictate otherwise, such as the special purpose data link between the nonsafety related PICS and the safety related NMS. Data pathways from nonsafety SSC to safety SSC shall be designed in accordance with the requirements provided in DI&C ISG-04 (**Reference** 34). The procedures for calibration and set point changes shall require a documented process for all such changes, with independent verification for both safety and nonsafety SSCs.

2. Requirements Phase

2.1. System Features

The customer and system developers shall define the security functional performance requirements and SSC configuration; interfaces external to the SSC; and the requirements for qualification, human factors engineering, data definitions, documentation for the software and hardware, installation and acceptance, operation and execution, and maintenance.

The security requirements shall be part of the overall system requirements where the digital SSC is incorporated, as these requirements are intended to ensure reliable operation of the SSC. Therefore, the V&V process of the overall system shall ensure the correctness, completeness, accuracy, testability, and consistency of the system security requirements.

Requirements specifying the use of commercial-off-the-shelf and pre-developed software and systems, including reuse and modification of software, shall address the vulnerability and reliability of the system

(e.g., by using pre-developed software functions that have been tested and are supported by operating experience).

2.2. Development Activities

The process of developing and documenting requirements shall ensure that the system does not contain undocumented code (e.g., back door coding), unwanted functions or applications, and any other code that could adversely impact the integrity or reliability of the digital SSC.

3. Design Phase

3.1. System Features

The system cyber security requirements identified in the system requirements specification shall be translated into specific design configuration items in the system design description. The system security design configuration items shall address control over the following:

- Physical and logical access to the system functions,
- Use of system services, and
- Data communication with other systems.

Design configuration items incorporating pre-developed software (PDS) and commercial-off-the-shelf (COTS) software shall address security vulnerabilities of the resulting system.

Physical and logical access control shall be based on the results of all security assessments performed. The results of the assessments may identify the need for more complex access control, such as a combination of knowledge (e.g., password), property (e.g., key, smart-card) or personal features (e.g., fingerprints), rather than just a password. The implementation of such changes shall be assessed, documented, and resulting changes implemented as necessary.

3.2. Development Activities

The developer shall delineate the standards and procedures that will conform with the applicable security policies to ensure the system, hardware, and software design products do not contain undocumented code, for example back door coding unwanted functions or applications, and any other code that could adversely impact the integrity or reliability of the digital SSC.

4. Implementation Phase

In the implementation phase for the SSC, the system design is transformed into code, database structures, and related machine executable representations. The implementation activity shall address the following:

- Hardware configuration and setup;
- Software coding and testing;

- Communication configuration and set-up; and
- Incorporation of PDS and COTS products.

4.1. System Features

The developer shall ensure that the security design configuration item transformations from the system design specification are correct, accurate, and complete. This shall be verified by the Verification and Validation Team, with oversight from the Cyber Security Team.

4.2. Development Activities

The developer shall implement security procedures and standards to minimize and mitigate tampering with the developed system. The developer's standards and procedures shall include review and testing to address undocumented codes or malicious functions that might allow unauthorized access or use of the system or cause systems to behave outside the system requirements.

The developer shall account for hidden functions and vulnerable features embedded in the code, and their purpose and impact on the system. If possible, these functions and features shall be disabled, removed, or at least addressed to prevent any unauthorized access or impact to the reliability of the system. Any remaining functions or features shall be addressed, at least, in the failure modes and affects analysis of the application code and/or system.

COTS systems are usually proprietary and may be unavailable for review. With no review, it may be difficult or impossible to evaluate the security of a COTS item, which may eliminate that COTS item from use.

There are no reliable methods to determine security vulnerabilities for operating systems and other callable code libraries. In such cases, unless operating systems and callable code libraries are modified by the application developer, the security effort should be limited to ensuring that the features within the system do not compromise the security and reliability requirements of the system, and the security functions should not be compromised by the other system functions. These and any other compensatory measures shall be documented, and such documentation provided to the customer. Since these features, compensatory measures, and vulnerabilities may need to be protected, this documentation should be provided under the customer cyber security plan.

5. Testing Phase

The objective of testing security functions is to ensure that the system security requirements are validated by execution of integration, system, and acceptance tests where practical and necessary. Appropriate security testing shall be included during system hardware configuration (including all external connectivity), software integration testing, system integration testing, equipment qualification testing, system validation testing, platform factory testing, platforms integration testing, and design installation verification testing.

5.1. System Features

The security requirements and configuration items intended to ensure reliable operation shall be considered part of validation of the overall system requirements and design configuration items.

Therefore, security design configuration items are just one element of the overall system validation. Each system security feature shall be validated to verify that the implemented feature achieves its intended function to protect against inadvertent access and the effects of undesirable behavior of connected systems and does not reduce the reliability of the system's functions.

5.2. Development Activities

The developer shall configure, document, and enable the designed security features correctly. The developer shall also test the system hardware architecture, external communication devices, and configurations for unauthorized pathways and system integrity. Attention shall be focused on built-in OEM features.

D Deleted [N/A]