MELTAC Platform Application Software Program Manual

Non-Proprietary

July 2017

Prepared: Tomonori Yamane — July 21, 2017
Tomonori Yamane, Manager
Control & Protection Systems Section
Date

Reviewed: Hitomi Sasaki — Jul. 21, 2017
Hitomi Sasaki, Manager
Control & Protection Systems Section
Date

Approved: Manabu Taniguchi — Jul. 4, 2017
Manabu Taniguchi, Senior Manager
Control & Protection Systems Section
Date

Approved: Hideki Matsui — Jul. 31. 2017
Hideki Matsui, QA Manager
Energy Systems Center
Date

Signature History

	Rev.0		
Prepared			
Reviewed			
Approved			

Revision History

Revision	Date	Page (section)	Description
0	July 2017	All	Initial Issue

Table of Contents

List of Tables

List of Figures

List of Acronyms

A/D	Analog/Digital
AOO	Anticipated Operational Occurrence
A-SPM	MELTAC Platform Application Software Program Manual
BTP	Branch Technical Position
CAD	Computer Aided Design
CAR	Corrective Action Report
CCB	Configuration Control Board
CCF	Common Cause Failure
FR	Code of Federal Regulations
CI	Configuration Item
COL	Combined Operating License
COLA	Combined License Application
COTS	Commercial-Off-the-Shelf
DC	Design Certification
DCD	Design Control Document
DT	Design Team
DTE	Design Team Engineer
DTM	Design Team Manager
ESF	Engineered Safety Features
ESFAS	Engineered Safety Features Actuation
FBD	Function Block Diagram
FD	Function Diagram
FMEA	Failure Modes and Effects Analysis
F-ROM	Flash Read Only Memory
FSAR	Final Safety Analysis Report
GBD	Graphic Block Diagram
HSI	Human-System Interface
I&C	Instrumentation and Control
IEEE	Institute of Electrical and Electronics Engineers
I/O	Input/Output
ISO	International Organization for Standardization
MELTAC	Mitsubishi Electric Total Advanced Controller
RG	Regulatory Guide

1.0 INTRODUCTION

1.1 Purpose

The purpose of this document is to describe the development lifecycle plans for the application software running on the MELTAC platform (MELTAC application software). Section 2 of this software program manual (A-SPM) provides an overview, and Section 3 provides the software development lifecycle plans which conform to the guidance of NUREG 0800, Branch Technical Position (BTP) 7-14, “Guidance on Software Reviews for Digital Computer-Based Instrumentation and Control Systems”.

This software program manual may be referenced in Application Licensing Documentation, which refers to specific licensing documentation for a group of plants or a single plant, such as:

- Design Certification Document (DCD)
- Combined Operating Licensing (COL) Application
- Final Safety Analysis Report (FSAR)
- License Amendment Request (LAR)

1.2 Scope

Plant safety or protection systems apply the MELTAC platform in a plant-specific manner. A plant-specific application means application software is developed, operated and maintained for one or more systems in one or more specific plants, using the MELTAC platform. The MELTAC platform itself is provided with basic software, which enables the hardware of the MELTAC platform to execute the application software. The software life cycle of the basic software shall be controlled under a separate software program manual, referred to as the “MELTAC Platform Software Program Manual” (JEXU-1041-1016).

The software life cycle of the application software shall be implemented, operated and maintained as described in this document (A-SPM) which is written to maintain consistency with “MELTAC Platform Software Program Manual” (JEXU-1041-1016) so that the application software and the basic software will execute correctly and remain under configuration management.

The Operations and Maintenance Phase of the application software development lifecycle is the responsibility of the nuclear plant maintenance or plant facility engineering organization. The nuclear plant organization shall maintain the application software in accordance with this A-SPM, and in accordance with their Quality Assurance Program (QAP).

The plans provided in this A-SPM are applicable to all application software development lifecycle projects that apply the MELTAC platform.

1.3 Definitions

The definitions of terminology in this A-SPM are contained in Appendix A.

2.0 PROGRAM OVERVIEW

Table 2.1-1 shows the application software life cycle plans corresponding to BTP 7-14 (Reference 1).

Table 2.1-1 Correspondence to BTP 7-14

BTP 7-14 Reference Plan	A-SPM Section
Software Management Plan (SMP)	3.1
Software Development Plan (SDP)	3.2
Software Quality Assurance Plan (SQAP)	3.3
Software Integration Plan (SIntP)	3.4
Software Installation Plan (SInstP)	3.5
Software Maintenance Plan (SMaintP)	3.6
Software Training Plan (STrngP)	3.7
Software Operations Plan (SOP)	3.8
Software Safety Plan (SSP)	3.9
Software Verification and Validation Plan (SVVP)	3.10
Software Configuration Management Plan (SCMP)	3.11
Software Test Plan (STP)	3.12

2.1 Organization and Responsibilities

2.1.1 Organization

The organizational structure to manage the application software life cycle plans in this A-SPM shall maintain the independence of the V&V team (VVT) from the Design Team (DT), as well as the Project organization and the QA organization, throughout a software life cycle.

Figure 2.1-1 shows the typical organizational structure from the Plant Requirement Phase to the Installation Phase. This structure shown in Figure 2.1-1 is provided on the basis that MELCO is responsible for documenting the system requirements for developing the application software for the target system(s) in the target plant(s) in accordance with a procurement specification, and confirming that all application software that is provided by MELCO and approved for installation meets these system requirements.

Mitsubishi Electric Power Products, Inc. (MEPPI) is responsible for providing a procurement specification to MELCO under MEPPI's 10 CFR 50 Appendix B QAP.

Figure 2.1-1 Organizational Structure to Execute the Application Software Life Cycle Process

2.1.2 Responsibilities

The roles and responsibilities for each organization are described as follows:

(1) Quality Assurance Manager (QAM) and Quality Assurance(QA) Section member (QAS)

The QAM is responsible for auditing activities that can affect the quality of items and services used in the application software applying the target plant system. The QAM is also responsible for planning, scheduling and conducting QA audits. The QAM assigns the QAS resources for performing QA audits. Audit findings are documented in Corrective Action Reports, and the QAM assigns them to the organization responsible for resolving the findings. The QASs are responsible to confirm that the application development process is performed under MELCO's 10 CFR 50 Appendix B QAP by performing QA audits. The QASs develop the associated QA audit plans and schedules, document the results of QA audits and report them to the QAM.

(2) Project Manager (PJM) and Project Management Team (PMT)

The PJM oversees project activities for the design and manufacturing organization, as well as the interfaces between the design and manufacturing organization and the VVT or QA organization. The PJM has no authority to plan, schedule, budget, or direct V&V or QA activities.

(3) Design Team Manager (DTM), Design Team Engineer (DTE) and Software Safety Analysis Engineer (SSE)

The DTM is responsible for ensuring adequate qualified staffing to execute all responsibilities of the design team, including the responsibilities for the Software Safety Plan (SSP) as described in Section 3.9. The DTM assigns DTE resources for the design activities of the application software and SSE resources for the software safety analysis. The DT conducts all design activities for hardware and software. Each DTE is directly responsible for the quality of the documents and configuration items that he/she produces, and shall not rely on the QA audits, V&V activities, or any other oversight or review activities for ensuring quality. The DTM assures that the DTE and the SSE correctly design and analyze for the application software based on technical requirements and the development process in accordance with the Software Quality Assurance Plan (SQAP) described in Section 3.3.

(4) V&V Manager (VVM), V&V Team Manager (VVTM) and Engineer (VVTE)

The V&V Manager is responsible for assignment of a VVTM, requesting budget and schedule of the V&V activities, instructing the commencement of the V&V activities and confirming the completion of V&V activities.
The VVTM is responsible for assignment of the VVTE resources, implementing independent V&V of the application software and ensuring that the V&V Team perform its activities in accordance with the Software V&V Plan (SVVP) described in Section 3.10 and Software Test Plan (STP) described in Section 3.12.
The VVTM shall confirm that the following qualifications and V&V independence criteria are met for the personnel selected for the V&V Team (VVT),

- Technical, managerial and financial independence as defined in IEEE std. 1012-2004. Digital control system and knowledge of target plant applying MELTAC platform and experience equal to or greater than personnel on the DT.

2.2 General Requirements

2.2.1 Overview of Life Cycle

The development of the application software is conducted according to the life cycle process that is formally defined in this A-SPM. The formally defined life cycle consists of seven (7) phases: Plant Requirements Phase, System Requirements Phase, Design Phase, Implementation Phase, Test Phase, Installation Phase, and Operation and Maintenance Phase.

Section 3.0 describes the key contents of each of the lifecycle plans, which are the same plans listed in BTP 7-14. Exceptions to specific BTP 7-14 guidance (or referenced Regulatory Guide or Standard) are explained at the end of each section of this A-SPM that describes a specific plan.

An overview of the software life cycle process is described below.

(1) Plant Requirements Phase

In this phase, the requirements and the key design aspects for the application software applying MELTAC platform that are critical to the plant design basis for safety, performance and maintainability shall be identified. Additionally, the industry regulations and standards that apply to the target system applying the MELTAC platform and its design process shall be identified. The Concept Phase and Requirement Phase described in IEEE std. 1012-2004 are combined into one phase as this phase.

(2) System Requirements Phase

A system requirement specification shall be developed for the application software. This specification provides the performance requirements, functional and Human-System Interface (HSI) requirements, and system interfaces requirements. All of these requirements are integrated and documented in the system requirements specification.

(3) Design Phase

This phase defines the specifications for hardware and software. The system design description for the application software is developed in accordance with the system requirements specification in this phase.

(4) Implementation Phase

During this phase, the hardware consisting of the MELTAC platform are manufactured and they are assembled to become the target system. The application software is also created, and it is integrated with the platform hardware and the basic software which are configured as the target system.

(5) Test Phase

During this phase a series of tests are performed to validate the design of the application software, first at the single system level and then at the level of combination of multiple

systems if necessary.

(6) Installation Phase

Activities in this phase are application software installation (or confirmation of software installation), inspection of the software/hardware configuration, and acceptance of the installed configuration through acceptance testing. Acceptance tests are conducted to ensure all equipment have not been damaged during installation or possible shipping, and that all interconnections are correct. Additional functional testing may be conducted as required by plant design.

(7) Operations and Maintenance Phase

During this phase, the target system is in operation. Also, software or hardware may be upgraded to accommodate new requirements, correct design errors or manage obsolescence. Retirement of application software is included in this phase.

2.3 Classification of Software

The application software applied for the target system shall be classified as software integrity level 4 in accordance with RG 1.168 Rev.2.

2.4 Documentation

Each application software life cycle plan provided in Section 3.0 of this A-SPM requires output documents. The output documents are defined in Table 3.2-1 through Table 3.2-4.

3.0 SOFTWARE DEVELOPMENT PROGRAM

This section describes the key contents of the 12 plans that govern the application software life cycle process.

(1) Software Management Plan (SMP)

a. Basic strategy and process for managing the application software life cycle

b. Method for monitoring progress against an application-specific project plan (as described in the SMP (Section 3.1))

c. Method for identifying any deviations from an application-specific project plan, or deviations from this A-SPM

d. Procedure for managing the application software

(2) Software Development Plan (SDP)

a. Technical aspects for the design and development activities of the application software

b. Phase activities in the application software life cycle for a project

c. Inputs to and outputs from each activity

(3) Software Quality Assurance Plan (SQAP)

a. Organizational responsibilities, security, quality assurance requirements, procedure, and methodology for application software

b. Metrics used to measure the specific quality

c. Reviews and audits in accordance with IEEE Std. 1028-2008

d. Problem reporting and corrective action

(4) Software Integration Plan (SIntP)

a. Procedures for software integration

1) Integrate Application Software Execution Data with the target CPU modules

2) Test the resulting integrated product

(5) Software Installation Plan (SInstP)

a. Procedures for software installation

1) Plan installation

2) Distribute software

3) Install software

4) Accept software in operational environment

(6) Software Maintenance Plan (SMaintP)

a. Processes for correcting faults and errors of the application software during plant operation

b. Activities for the maintenance of the application

1) Failure reporting

2) Fault correction

3) Re-release software

4) Configuration management system

(7) Software Training Plan (STrngP)

a. Metrics for the effectiveness of the training

b. Procedures for software training

1) Training activities

2) Software training manual and material

c. Specification of effective and sufficient training resources

(8) Software Operations Plan (SOP)

Operation of the application software during the Operation and Maintenance Phase

(9) Software Safety Plan (SSP)

a. Methodologies for software safety for all life cycle process of the application software

b. Definition of technical requirements and organizational responsibilities for specific software safety activities

c. Software Safety Management (SSM) in accordance with IEEE Std. 1228-1994 (Reference 8)

d. Software Safety Analysis (SSA)

1) Plant Requirement Phase SSA

2) System Requirement Phase SSA

3) Design and Implementation Phase SSA

4) Test Phase SSA

(10) Software Verification and Validation Plan (SVVP)

a. V&V activities during the application software life cycle phases

b. Procedures and methodologies for each V&V activity in accordance with IEEE Std. 1012-2004

1) System Requirement Phase V&V

2) Design Phase V&V

3) Implementation Phase V&V

4) Test Phase V&V

5) Installation V&V

6) Maintenance and operation V&V

c. V&V reporting requirements and V&V anomaly reporting and resolution

(11) Software Configuration Management Plan (SCMP)

a. Methods required for maintaining the project specific application software configuration items (CIs) in a controlled configuration

b. The six classes of information required by IEEE Std. 828-2005 (Reference 11)

1) Introduction

2) SCM management

3) SCM Activities

4) SCM Schedules

5) SCM Resources

6) SCM Maintenance

c. Procedure in each phase

(12) Software Test Plan (STP)

a. Methods for the following V&V test activities

1) Unit V&V Test

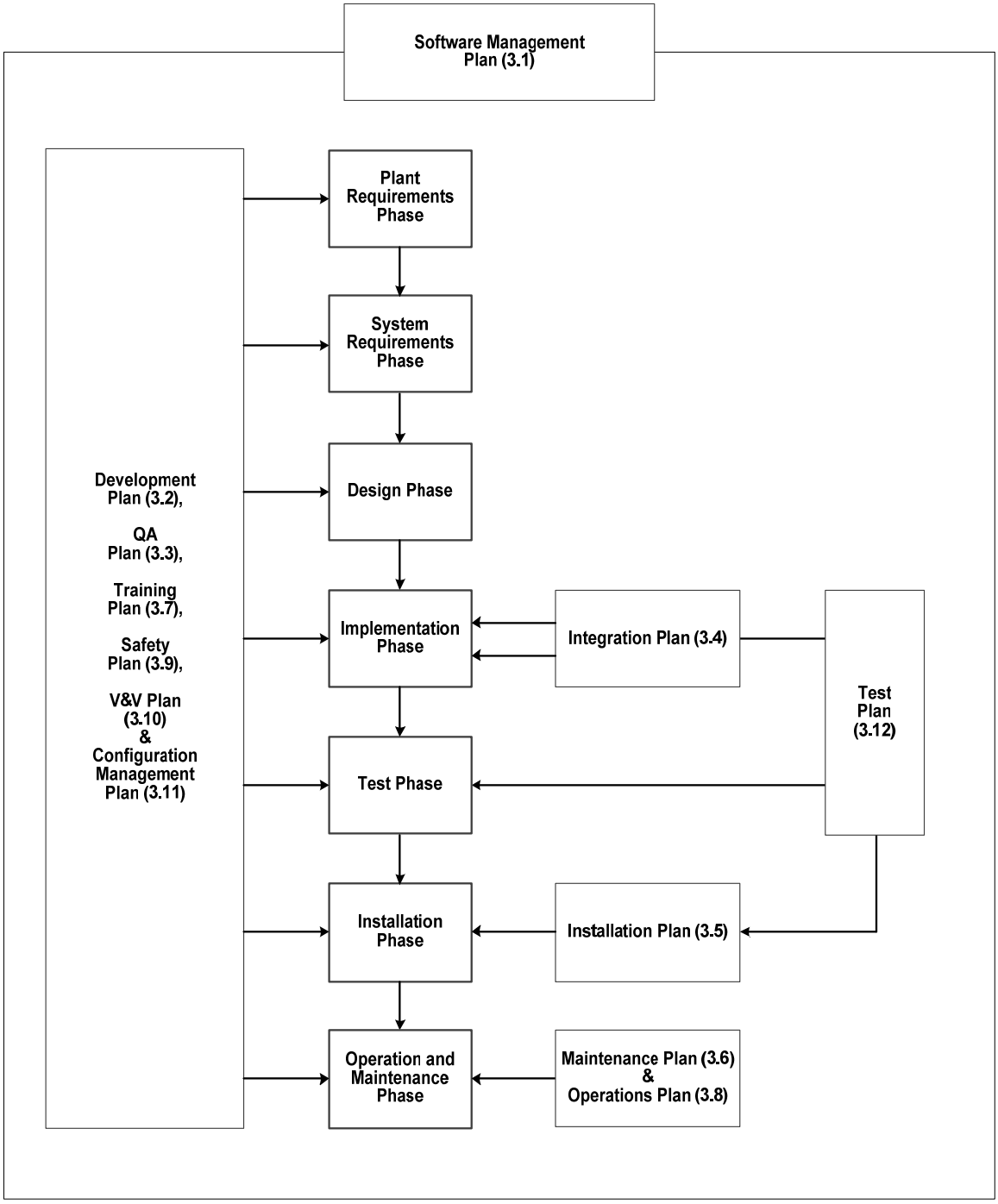
2) Integration V&V Test

3) System V&V Test

4) Acceptance V&V Test

b. Test documents in accordance with IEEE Std. 829-2008 (Reference 13)

The relationship of the software life cycle plan to each phase of the software life cycle is shown in Figure 3.0-1.



(*.*) means the section in this A-SPM.

Figure 3.0-1 Overview of Software Life Cycle Plan

3.1 Software Management Plan (SMP)

3.1.1 Purpose

The Software Management Plan (SMP) describes the overall management process for the application software life cycle. An overview and a description of the general requirements for the application software life cycle process are provided in Section 2.2 of this A-SPM.

The SMP also describes the general functions of the application software which are expected to be delivered by a project, and how each of these functions shall be traceable to the requirements identified in the Plant Requirements Phase output documents. In addition, the SMP describes the following items:

- An overview of where the application software will reside.
- General overview of an application software project.

3.1.1.1 Function

The application software governed by the SMP performs the functions of the target plant system, using the MELTAC platform, that are implemented in software.
The key functions of the application software are (For example):

- Process input signals and manual system level actuation signals for the reactor protection functions and the Engineered Safety Features (ESF) actuation functions.
- Process signals such as analog to digital (A/D) conversion, input signal and setpoint comparison, trip/actuation algorithm calculations, and a voting logic.
- Initiate reactor trip signal and engineered safety features actuation signals.
- Process input signals for post-accident monitoring and safe shutdown instrumentation.

Provide manual component level controls for credited operator actions for accident mitigation and for achieving and maintaining safe shutdown. For example:

- Initiate operating bypasses, maintenance bypasses, system level reset of automatic safety actuation signals and periodic surveillance testing.
- Provide safety-related HSI for the main control room and remote shutdown room to monitor, control, and test safety functions.

3.1.1.2 Overview

To implement the application software functions listed in Section 3.1.1.1, the application software utilizes the following configuration and features:

- The target plant system applying the MELTAC platform shall be described in application specific licensing documentation.

- The application software is subject to Independent V&V activities as described in the SVVP (Section 3.10).

- The design features to satisfy the reliability goals of the target plant system shall be described.

3.1.2 Organization/Responsibilities

All organizations involved in the application software life cycle process described in this A-SPM shall follow internal procedures that implement the requirements of the SMP and all other sections of this A-SPM.

Subcontractors and suppliers shall be managed in accordance with MELCO's 10 CFR 50 Appendix B QAP.

In addition, subcontractor management in the SSP (Section 3.9) and subcontractor/vendor control in the SCMP (Section 3.11) will be in accordance with Section 3.9.7.12 and 3.11.3.6 respectively.

An application-specific project plan for the target project shall include the following information, as a minimum:

- Project Purpose

- References to the latest applicable versions of the output documents from the Plant Requirements Phase (for compliance with applicable regulations, codes and standards)

- Responsible organizations

- Application-specific customer requirements

- Project schedule

- Project budget

Each application-specific project plan is developed by a Project Management Team (PMT) under the direction of a Project Manager (PJM). The PJM is responsible for reviewing and approving the application-specific project plan.

Section 2.1 of this A-SPM describes the software project organizational structure, the interfaces and boundaries between the project organization and other organizations, and the management reporting channels.

The organizational structure and independence between the organizations, as defined in Section 2.1 of this A-SPM, shall be maintained for each phase of the software life cycle process. The division of responsibility between companies for fulfilling a particular organizational role for a specific life cycle phase, or for fulfilling all organizational roles for a specific life cycle phase, shall be defined in each application-specific project plan.

Section 2.1 of this A-SPM describes the roles and responsibilities for personnel assigned to activities described in this A-SPM. Section 2.1 also provides a policy statement that the primary responsibility for assuring the quality of the application software during development is assigned to the personnel responsible for application software development.

Section 2.1 also describes the criteria and responsibilities for assuring independence of the QA organization (QAM and QAS) and the V&V organization (VVTM and VVTE) from the application software design organization (DTM, DTE and SSE). In particular, Section 2.1 requires VVT independence in accordance with Annex C of IEEE Std. 1012-2004 (Reference 9).

The organizational roles and responsibilities for the application software described in Section 2.1 satisfy the guidance in Section B.3.1.1 of BTP 7-14 (Reference 1) and Section 3.1.1 of NUREG/CR-6101 (Reference 23).

3.1.3 Oversight

3.1.3.1 Basic Strategy

In managing the application software life cycle process, the following basic management strategies shall be implemented to achieve high reliability and design quality of the application software.

- Ensure independence between the life cycle management organizations as described in Section 2.1 of this A-SPM.
- Ensure the application software life cycle activities are conducted in accordance with this A-SPM.
- Ensure that if any deviations from this A-SPM are detected and reported, corrective actions shall be initiated in accordance with MELCO's 10 CFR 50 Appendix B QAP.
- Ensure that the Design Team (DT) that produces each application software design related output has the primary responsibility for the quality of these outputs.
- Ensure that the DT understands that the V&V and QA activities are fully independent of the design activities and limited to confirmation that the design outputs products are of high quality.

3.1.3.2 Other Considerations

- The DTM identifies and controls precise milestones for design activities of the application software life cycle process in the project schedule defined in the application-specific project plan. Each application software product is developed in the order required by the schedule.
- Progress of the design activities shall be confirmed by regular DT meetings to monitor the design activity status and deviations.

- Any design activities that deviate from the project schedule are investigated to determine the cause of the deviation, and corrective actions shall be identified promptly.

- The VVTM independently identifies and controls milestones for each V&V activity of the application software life cycle, and performs the required V&V activities in the order required by the V&V schedule.

- Confirm progress and effectiveness of V&V activities through regular VVT meetings to check the V&V activity status and deviations.

- The MELTAC platform, including the life cycle process for the basic software, is described in “MELTAC Platform Software Program Manual” (JEXU-1041-1016) (Reference 27). The DT and the VVT shall ensure that the basic software is delivered according to the project schedule**.**

- The DT shall identify the activities being performed by its sub-vendor and ensure that these activities are conducted in accordance with this A-SPM.

3.1.4 Security

{[

Security-Related Information –Withhold Under 10 CFR 2.390

]

In addition to the general security requirements described above, security measures for specific application software life cycle processes are described in the other life cycle plans described in this A-SPM.

3.1.5 Measurement

The measures below shall be used to monitor and control the progress of the application software life cycle process.

- Progress status of documents in comparison to the project schedule. This may be expressed as a percentage (progress rate).
- The number of open items on the Problem List as described in Section 3.1.8.2.
- The number and severity level of V&V Anomaly Reports as described in the SVVP (Section 3.10).
- The number of design changes.

3.1.6 Procedures

3.1.6.1 Objective and Priorities

The application software projects are authorized and initiated by the PJM.

The DTM shall describe the objectives and priorities for management activities and specify the preliminary schedule, requirements, scope and budget for the application software development. The DTM will provide this information to the PJM for incorporation into the application-specific project plan.

The DTM shall include any necessary assumptions, dependencies and constraints in the information provided to the PJM. For example:

- assumption: degree of risk
- dependency: relationship between activities, or activity and milestone
- constraints: options regarding scope, staffing, and schedule

The VVTM has the sole authority to approve the release of the design outputs of the application software.

3.1.6.2 Risk Management

Risk management shall be performed in the application software life cycle process. The Project Manager shall evaluate potential risk areas and determine if specific risks should be more closely assessed in the course of the project.

Identified risks shall be placed on a Risk Matrix that describes each risk and the methods to be used for assessing and mitigating the risk. The Design Team Manager (DTM) shall initiate and maintain the Risk Matrix, including updates to account for new risk items identified through periodic review methods described in the SQAP (Section 3.3).

The Project Manager shall employ the following risk mitigation strategies for risks identified on the Risk Matrix:

- Analyze identified risks to determine their relative rank and priority.
- Develop contingency plans for high priority or safety significant risks.
- Immediately initiate corrective actions if an identified risk becomes an actual problem. Maintain a working environment that supports effective communication in order to correspond to emerging risks and problems.

3.1.6.3 Monitoring and Controlling

Application software life cycle activities shall be executed and controlled by procedures that implement each of the plans in this A-SPM.

- The Managers identified in Section 2.1 shall be responsible for the respective activities described in Sections 3.2 through 3.12.

- Meetings shall be held periodically to review and confirm the status of project activities to identify any issues to be addressed, and to initiate corrective actions if needed.

- The PMT shall monitor, review, and periodically report project progress to the PJM. The PMT shall also monitor, review, and periodically report the Problem List to the PJM.

Each organization involved in the application software life cycle shall develop internal procedures for the activities under their responsibility. The specific procedures used to implement the requirements shall be specified in the application-specific project plan. Should the division of responsibility for the application software life cycle process change or the procedures change, the application-specific project plan shall be revised to ensure appropriate procedures are in place during development activities.

Implementing procedures for activities under MELCO responsibility shall be controlled under MELCO's 10 CFR 50 Appendix B QAP.

The PJM is responsible for coordination of communications and information transfer between the following entities to ensure that organizational interfaces on the application software project are effective:

- The DT and the customer

- The DT and the QA Organization

- The DT and the Verification and Validation Team (VVT)

3.1.6.4 Staffing Plan

The DTM and the VVTM shall specify the number of personnel required to conduct the design activities and the V&V activities of each project and develop staffing plans in according with the project schedule. V&V activities, however, shall be controlled solely by the VVTM, and shall not be influenced by the DTM or the PJM.

3.1.6.5 Document Plan

All outputs described in this A-SPM, including software configuration items and documents, are listed in Table 3.2-1 through Table 3.2-4 with the responsible organization for each output.

3.1.6.6 Change Plan

A means of managing externally or internally driven changes to any of the application software and related documents is described in the SCMP (Section 3.11).

The organizations responsible for assessing potential changes to any of the application software and related documents in this A-SPM (Sections 3.1 through 3.12) are described in the SCMP.

3.1.7 Budget

Sufficient resources, such as financing, human resources and tools, shall be made available for each organization that are assigned activities described in this A-SPM.

The QAM/QAS and the VVTM/VVT shall be provided budgets that are independent from the DT budget. The required resources shall be identified before proceeding beyond the initiation of a project.
Resource utilization reports shall be periodically prepared and assessed by each responsible manager to ensure adequate resources are available throughout the application software life cycle process.

In addition, budget reports for each organization shall be periodically prepared and assessed by respective managers to ensure the availability of resources when needed for executing the requirements.

3.1.8 Methods/Tools

3.1.8.1 Methods

During each project phase, project management shall be executed in accordance with the basic process described in Section 3.1.6.

3.1.8.2 Tools

The following tools shall be used for the application software project management, as a minimum:

(1) Application-specific project plan

An application-specific project plan shall be initiated and maintained by the Project Manager for each development or changed activity that produces or affects an application software configuration item. Exceptions are editorial changes to documents that do not affect any functional, performance, safety or quality characteristics of the application software.

The following information shall be included in the application-specific project plan, as a minimum:

a. Scope Description
b. Summary Description of Systems, Components or configuration items to be produced or affected
c. A List of Nonconformance or Corrective Action Reports to be resolved by the project (if any)
d. Results of Configuration Control Board (CCB) activities (if required), as described in the SCMP (Section 3.11)
e. Project Schedule (with periodic updates for progress and schedule changes), including milestones, hold points and review schedules

(2) Master Test Plan (MTP)/Master Test Report (MTR)
An MTP summarizes overall test plans for a project. This document shall be prepared, revised and maintained during the project as needed. The MTP shall identify the test documents for all testing activities, describe the test overview, and provide the overall schedule of a project. In accordance with the MTP, a Level Test Plan (LTP) shall be prepared for each test level. Upon completion of all test activities described in the MTP, an MTR shall be issued. For specific contents of the MTP and MTR, see Section 3.12.5.

(3) Level Test Plan (LTP)/Level Test Report (LTR)
LTPs shall be prepared for each test level specified in the MTP. The LTP shall describe the test overview and provide a schedule for a specific testing activity. Upon completion of a test level specified in the LTP, an LTR shall be prepared. For specific contents of the LTP and LTR, see Section 3.12.5.

(4) Risk Matrix

The Project Manager shall evaluate potential risk areas and determine if any risks should be periodically assessed in the course of the project. Identified risks shall be placed on a Risk Matrix that describes each risk and the methods to be used for assessing and mitigating them. The DTM shall initiate and maintain the Risk Matrix, including updates to account for new risk items identified through periodic reviews as described in the SQAP (Section 3.3).

The following risk areas shall be considered for inclusion on the Risk Matrix:

- Previous project experience and lessons learned
- Operating experience
- Complexity of the proposed change, including impact on multiple configuration items
- Internal and external organizational interfaces
- Schedule pressure
- Availability of resources
- Need for specialized resources
- Results of previous QA Audits and Corrective Action Reports

- Previous V&V Anomaly Reports
- Previous Nonconformance Reports

(5) Problem List

Items on the Risk Matrix that become actual issues, or any other issues such as technical problems, significant design review comments, organizational conflicts, facility and environment problems, or resource problems that are identified in the course of a project shall be identified and reported on a Problem List.

The Problem List shall be maintained and updated by the Project Manager as problems are identified, addressed, and closed. Activities to mitigate and close each identified problem shall be described in the Problem List.

Problems that can impact a critical safety function as described in the SSP (Section 3.9) require the initiation of a Nonconformance Report as described in the SQAP (Section 3.3).

3.1.9 Personnel

The DTE, including individuals assigned to perform the software safety analyses activities described in the SSP (Section 3.9) shall be trained and qualified prior to performing any of the activities assigned to the DT as described in this A-SPM. The DTM is responsible for assuring that the DTE are trained and qualified for their assigned activities. Staff qualification and training for SSE is described in Section 3.9.7.3.

The VVTE, including individuals assigned to perform V&V of the software safety analyses outputs from the DT, shall be trained and qualified prior to performing any of the V&V activities assigned to the VVT as described in the SVVP and STP (Sections 3.10 and 3.12, respectively). The VVTM is responsible for assuring that the VVTE are trained and qualified for their assigned activities.

Training and qualification of the DTE and VVTE shall include technical competencies, software engineering competencies, and the application software life cycle process knowledge as determined by the DTM and the VVTM, respectively. The following process shall be included in the training and qualification.

- Define the competence requirements for the design or V&V activities, designate the personnel having the work experience and prepare designation records.
- Prepare training program, implement training and prepare training records.

3.1.10 Standards

The SMP complies with the following guidance and standards.

- Clause A.1.2.7 and of IEEE Std. 1074-2006 (Reference 4) which is endorsed by RG 1.173 (Reference 20)

- Section 3.1.1 of NUREG/CR-6101 (Reference 23)

3.2 Software Development Plan (SDP)

3.2.1 Purpose

This Software Development Plan (SDP) describes the design and development activities of the application software.

The purpose of this SDP is to:

- Describe the phase activities in the application software life cycle
- Describe the inputs to and outputs from each activity.

This SDP complies with the guidance and standards identified in Section 3.2.9.

3.2.2 Organization

3.2.2.1 Application Software Life Cycle Process

As described in Section 2.2.1, the application software life cycle process consists of the following seven phases. The life cycle model used for the application software is the waterfall model illustrated in Figure 3.2-1.

1. Plant Requirements
2. System Requirements
3. Design
4. Implementation
5. Test
6. Installation
7. Operations and Maintenance

The activities illustrated in Figure 3.2-1 shall be performed by the Project organization, the Design Team (DT), the Verification and Validation Team (VVT), and the Quality Assurance organization as described in Section 2.1. Figure 3.2-1 also provides the organizational interfaces and boundaries.

The activities required of each organization are listed in Tables 3.2-1 to 3.2-4, including the phase-specific activity inputs and outputs.

3.2.2.2 Comparison of Application Software Life Cycle Activities to IEEE Std. 1074-2006

The relationship of the application software life cycle to IEEE Std. 1074-2006 is mapped in Table 3.2-5, where it demonstrates that all mandatory activities are accounted for in the application software life cycle process.

3.2.3 Oversight

Project oversight activities and measures are described in Section 3.1.3.

3.2.4 Risks

Risk management activities and measures are described in Section 3.1.6.

3.2.5 Measurement

Measurements used to monitor and control the technical and quality aspects of the application software development process shall be performed as described in Section 3.3.4.

3.2.6 Procedures

This section describes the inputs, the activities and the outputs of each application software life cycle phase. An accompanying illustration is provided in Figure 3.2-1.

The body of this SDP annotates specific activities in parentheses, such as “(P-2),” for cross-reference to Figure 3.2-1. The first digit means the type of activity, such as “P” for “Project” and the second digit means the application software life cycle phase where it is performed as listed in Section 3.2.2.1. For example, “(P-2)” means a Project Management activity, as listed in Table 3.2-1, in the System Requirements Phase (Item 2 in the list provided in Section 3.2.2.1).

Software safety analyses shall be performed as described in the SSP (Section 3.9).

3.2.6.1 Plant Requirements Phase

The Plant Requirements Phase is defined as the activities that are conducted while developing plant system-level functional and performance requirements for the target plant system application. These activities are generally supported by Application Licensing Documentation, such as the FSAR for an operating plant or the Design Certification Document and Combined Operating Licensing (COL) Applications.
This phase defines the key design aspects for the target plant system. The Plant Requirements Phase consists of the following activities.

1. Develop/Maintain Platform (B-1)
2. Develop/Maintain Plant Requirements (D-1)

3.2.6.1.1 Develop/Maintain Platform

[

]

3.2.6.1.2 Develop/Maintain Plant Requirements

[

]

3.2.6.2 System Requirements Phase

The System Requirements Phase defines the requirements for the target plant system. These requirements include performance, functional and Human-System Interface (HSI) requirements, and system interface requirements.

The System Requirements Phase consists of the following activities:

1. Develop System Requirements (D-2)
2. System Requirements Phase V&V (V-2)

3.2.6.2.1 Develop System Requirements

[

]

3.2.6.2.2 System Requirements Phase V&V Activity

[

]

3.2.6.3 Design Phase

The Design Phase transforms the system requirements into the system design description. The Design Phase consists of the following activities:

1. Develop System Design (D-3)
2. Design Phase V&V (V-3)

3.2.6.3.1 Develop System Design

[

]

3.2.6.3.2 Design Phase V&V

[

]

3.2.6.4 Implementation Phase

The Implementation Phase consists of the following activities.

1. Manufacture Hardware
2. Develop Application Software (D-4A)
3. Integrate Software with Hardware (D-4B)
4. Implementation Phase V&V (V-4A)

3.2.6.4.1 Manufacture Hardware

[

]

3.2.6.4.2 Develop Application Software

[

]

3.2.6.4.3 Integrate Software with Hardware

[

]

3.2.6.4.4 Implementation Phase V&V

[

]

3.2.6.5 Test Phase

The Test Phase consists of the following activities:

1. Integration V&V Test (V-5A)
2. System V&V Test (V-5B)

3.2.6.5.1 Integration V&V Test

[

]

3.2.6.5.2 System V&V Test

[

]

3.2.6.6 Installation Phase

The Installation Phase consists of the following activities.

1. Install System (D-6)
2. Installation Phase V&V (V-6A)
3. Acceptance V&V Test (V-6B)

3.2.6.6.1 Install System

[

]

3.2.6.6.2 Installation Phase V&V

[

]

3.2.6.6.3 Acceptance V&V Test

[

]

3.2.6.7 Operations and Maintenance Phase

The Operations and Maintenance Phase consists of the following activities.

1. Operations and Maintenance Support (D-7)
2. Maintenance Phase V&V (V-7)

3.2.6.7.1 Operations and Maintenance Support

[

]

3.2.6.7.2 Maintenance Phase V&V

[

]

3.2.7 Schedule

The software development schedule including milestones, hold points and coordination with the QA organization is provided in the application-specific project plan, as described in the SMP (Section 3.1).

3.2.8 Methods/Tools

The following methods and techniques are used to develop the software.

3.2.8.1 Methods

3.2.8.1.1 Design Verification

The DT shall perform design verification for all design outputs described in 3.2.6, prior to the distribution of them, as described in the SQAP (Section 3.3). The DTM shall confirm that the qualification and independence criteria are met for the personnel selected for DT:

- A verifier should have and hold knowledge and experience related to both digital control system and the subject system to be updated, equal to or greater than the person who prepared the document or configuration item.
- A verifier should be a person different from the person who prepared the document or configuration item.

3.2.8.1.2 Software Quality Metrics

The DT shall develop and maintain the software quality metrics as described in the SQAP

(Section 3.3).

3.2.8.1.3 Equipment Qualification

The DT shall assure the qualification of the MELTAC platform is maintained as described in “Safety System Platform - MELTAC - Topical Report” (JEXU-1041-1008), to be in compliance with Clause 5.4 of IEEE Std. 7-4.3.2-2003 (Reference 3), as endorsed by RG 1.152 (Reference 15).

3.2.8.1.4 Software Review and QA Audit

Software Review and QA Audits are described in Section 3.3.5.2.

3.2.8.1.5 Software Training

Software Training is described in the STrngP (Section 3.7).

3.2.8.1.6 Software Safety Analysis

Software safety analysis is described in the SSP (Section 3.9).

3.2.8.1.7 Verification and Validation (V&V)

Independent V&V is described in the SVVP (Section 3.10).

3.2.8.1.8 Software Configuration Management

Software Configuration Management is described in the SCMP (Section 3.11).

3.2.8.2 Tools

3.2.8.2.1 Documented Checklists

The checklists shall be used to verify the design outputs. The Documented Checklists are described in the SQAP (Section 3.3).

3.2.8.2.2 Requirements Traceability Matrix

Traceability of documents between phases shall be identified and documented using a Requirements Traceability Matrix (RTM) for verifying the design outputs. The RTM is described in the SVVP (Section 3.10).

3.2.8.2.3 Application Software Development Tool

[

]

3.2.9 Standards

This SDP complies with the following guidance and standards.

- Clause 5.3 and 5.9 of IEEE Std. 603-1991 (Reference 2) which is endorsed by RG

1.153 (Reference 25)

- Clause 5 of IEEE Std. 7-4.3.2-2003 (Reference 3) which is endorsed by RG1.152 (Reference 15)

- IEEE Std. 1074-2006 (Reference 4) which is endorsed by RG 1.173 (Reference 21)

- IEEE Std. 830-1998 (Reference 5) endorsed by RG 1.172 (Reference 19), with the following exception:
  Clause 4.6 is not applicable to this SDP. As shown in Tables 3.2-1 to 3.2-4 and Figure 3.2-1, the lifecycle activities for the application software which are organized in a waterfall model include no activity to use prototyping.

- NUREG/CR-6101 (Reference 23)

- Section C of RG 1.152 Rev. 3 (Reference 15)

- Section C of RG 1.153 Rev. 1 (Reference 25)

- Section C of RG 1.172 Rev.1 (Reference 19)

- Section C of RG 1.173 Rev.1 (Reference 20)

3.2.10 Basic Software

The SDP for the basic software are described in Section 6.0 of JEXU-1041-1008.

Table 3.2-1 Project Organization Activities

Table 3.2-2 QA Organization Activities

Table 3.2-3 Design Team Activities (1/2)

Table 3.2-3 Design Team Activities (2/2)

Table 3.2-4 V&V Team Activities (1/2)

Table 3.2-4 V&V Team Activities (2/2)

Table 3.2-5 MELCO Application Software Life Cycle Activities Mapped to IEEE Std. 1074-2006 Endorsed by RG 1.173 (1/4)

Table 3.2-5 MELCO's Application Software Life Cycle Activities Mapped to IEEE Std. 1074-2006 Endorsed by RG 1.173 (2/4)

Table 3.2-5 MELCO’s Application Software Life Cycle Activities Mapped to IEEE Std. 1074-2006 Endorsed by RG 1.173 (3/4)

Table 3.2-5 MELCO Application Software Life Cycle Activities Mapped to IEEE Std. 1074-2006 Endorsed by RG 1.173 (4/4)

Table 3.2-6 Minimum Contents of SysRS (1/2)

		1

Table 3.2-6 Minimum Contents of SysRS (2/2)

Table 3.2-7 Required SysRS Functional Characteristics

Table 3.2-8 Required SysRS Process Characteristics

Figure 3.2-1 Overview of Application Software Life Cycle Process (1/2)

Figure 3.2-1 Overview of Application Software Life Cycle Process (2/2)

Figure 3.2-2 Development Process of the Application Software

3.3 Software Quality Assurance Plan (SQAP)

3.3.1 Purpose

This Software Quality Assurance Plan (SQAP) describes the organizational responsibilities, security, quality assurance requirements, techniques, procedures and methodologies for assuring high quality application software for use in the target plant system.

This SQAP is based on the software life cycle process described in the SDP (Section 3.2), and describes the quality methodology to be followed during development and maintenance of the application software.

This SQAP complies with the guidance and standards identified in Section 3.3.8.

The quality of the following application software life cycle outputs shall be assured by applying the QA methods described in this SQAP to activities described in this A-SPM, with particular emphasis on the following activities and sections:

(1) The application software design documents as described in the SDP (Section 3.2)
(2) The application software test documents as described in the STP (Section 3.12)
(3) The V&V documents as described in the SVVP (Section 3.10)
(4) The application software configuration items as described in the SCMP (Section 3.11).

3.3.2 Organization/Responsibilities

The organizations and responsibilities for assuring application software quality are described in Section 2.1.

3.3.3 Security

The QA organization shall conduct periodic audits to confirm the security of the application software development process is controlled in accordance with this A-SPM.
The security requirements described in this A-SPM shall be implemented in accordance with RG1.152 (Reference 15).

3.3.4 Measurement

General measures of the application software life cycle process are described in this section. Metrics used to measure the specific quality of the application software documents and configuration items as they emerge from the application software life cycle process are described in Section 3.3.5.1.

The number and age of open QA Audit findings reported by the QA organization (QAM/QAS) are recorded as QA data, and are treated as a measure of the quality of the application software life cycle process and the documents and configuration items that are produced. This measure is an important indicator for observing the extent to which the application software life cycle processes are performed in accordance with this A-SPM. Audit findings shall be tracked to closure by the QAM/QAS.

QA Audit findings that detect software hazards (anomalies), not already discovered and documented by either the DT or the VVT (including their independent reviewers), are considered to be an indication of a potential weakness in the application software life cycle process or effectiveness of the overall organization, and merit further investigation. For definition of “software hazards” and “anomalies”, refer to Appendix A of this A-SPM.

Other application software life cycle process measures include the following:

(1) Number of comments identified by Design Reviews (as described in the SDP (Section 3.2))
(2) Number and age of V&V Anomaly Reports (as described in the SVVP (Section 3.10))
(3) Number and age of Nonconformance Reports (NCR)
(4) Number and age of Corrective Action Reports (CAR)

3.3.5 Procedures

3.3.5.1 Metrics

Section 4.0 lists the regulatory guidance and industry standards to be used in this A-SPM. These standards address the topics of documentation standards, design standards, test standards and software quality assurance products.

The metrics described in this Section shall be used to measure the specific quality of the application software documents and configuration items as they emerge from the application software life cycle process. The metric data described below shall be collected periodically, monitored, and systematically analyzed for emerging trends.

(1) Correctness/completeness of the Plant Requirements Phase and the System Requirements Phase outputs
(2) Design Phase output compliance with requirements
(3) Implementation Phase output compliance with design specifications
(4) Test Phase output compliance with requirements documents
(5) Installation (on-site) functional compliance with requirements
(6) Operations and Maintenance Phase performance history

The DTM shall be responsible for collecting, monitoring and analyzing these metrics, and shall implement improvements to the application software life cycle process as necessary. Any adverse trends detected shall result in initiation of a Nonconformance Report under MELCO’s 10 CFR 50 Appendix B QAP.

Throughout the application software life cycle process, the following quality measures of application software configuration items shall also be collected, monitored and analyzed by the DTM. Any adverse trends detected shall result in initiation of a Nonconformance Report.

(1) Functional and Performance Characteristics

    a. Accuracy

    b. Functionality

c. Reliability

d. Robustness

e. Safety

f. Security

g. Timing

(2) Application Software Life Cycle Process Document Characteristics

a. Completeness

b. Consistency

c. Correctness

d. Style

e. Traceability

f. Unambiguity

g. Verifiability

3.3.5.2 Reviews and Audits

This section describes the reviews and audits required throughout the application software life cycle process.

The following activities shall be performed using the methods described in IEEE Std. 1028-2008 (Reference 6) which is endorsed by RG1.168 (Reference 16):

(1) Management Reviews

(2) Design Reviews

(3) Audits

(1) Management Reviews

The objective of the management review is to assess and determine periodically if required activities assigned to the DT and VVT are making progress, and are being performed in compliance with this A-SPM and implementing procedures.
The subjects of the management review are as follows (for example):

- Software Verification and Validation Plan Review
- Software Configuration Management Plan Review

Inputs to the management review include the application-specific project plan, Risk

Matrix, Problem List described in the SMP (Section 3.1) as a minimum.
The management review activities are conducted individually and periodically as follows.

a. The DTM and VVTM shall periodically assess and summarize the activities assigned to their teams to determine if any resource assignment changes are necessary, or to redirect their teams, if necessary, in order to maintain compliance with this SPM and implementing procedures.

b. The VVTM shall use the Management Review method to independently determine whether or not to proceed from one phase of the application software life cycle to the next phase. If any V&V anomalies detected in any application software life cycle phase potentially require a functional change described in an output document produced in the Plant Requirements Phase or the System Requirements Phase, the DTM shall convene a Configuration Control Board activity as described in the SCMP (Section 3.11) before proceeding.

c. The DTM and VVTM shall periodically assess and verify that the activities assigned to their teams comply with the requirements of this SPM and implementing procedures.

Management Review meetings shall accomplish the following objectives:

1) Evaluate project status and measures
2) Review the application-specific project plan, Risk Matrix, and Problem List
3) Review open V&V Anomaly Reports
4) Generate a list of action items
5) Document the meeting results

Management Reviews shall be performed using the method described in Section 4 of IEEE Std. 1028-2008 (Reference 6).

(2) Design Reviews

The objective of a Design Review is to evaluate application software configuration items (described in the SCMP (Section 3.11)) to determine their suitability for their intended use and identify discrepancies from specifications and standards.

Design Reviews provide assurance that:

a. Application software configuration items conform to specifications

b. Application software life cycle phase outputs adhere to the requirements of this A-SPM and its implementing procedures

c. Changes to application software configuration items are properly implemented and affect only those areas identified by the requested change.

Design reviews described in this SQAP correspond to the following reviews required

to be conducted in accordance with IEEE Std.730-1998.
- Software Requirement Review
- Preliminary Design Review
- Critical Design review

Design Reviews shall be performed using the method described in Section 5 of IEEE Std. 1028-2008 (Reference 6). Design Reviews shall be conducted by DT Engineers and those who have the same training and qualifications as the DT Engineer who prepared the application software configuration item or design document. The results of design review comments and activities shall be documented and reported to the responsible DTE and the DTM.

The detailed design review such as the design review conducted for the CIs of D-3 and D-4 phase are equivalent to the “Inspections” review method described in IEEE Std. 1028-2008 (Reference 6).
Walk-throughs described in IEEE Std. 1028-2008 (Reference 6) are not conducted for application software development activities, because the source code of application software is verified as the detailed design review described above and V&V activities as described in SVVP.

The DTM shall confirm that internal design reviews have been adequately performed for the design output documents described in the SDP (Section 3.2). The DTM shall attend design review meetings convened as necessary in the course of design activities, and assess design review reports for clarity, completeness, and timeliness.

(3) QA Audits

The objective of QA Audits is to provide an independent evaluation of conformance of the application software life cycle activities and outputs with this A-SPM and its implementing procedures.

QA Audits shall be conducted and reported under MELCO’s 10 CFR 50 Appendix B QAP using the following methods described in IEEE Std. 1028-2008 (Reference 6). QA member shall examine application software life cycle output products and document their observations, findings, and recommended corrective actions. QA member shall be free from bias and influences that reduce their ability to make independent and objective evaluations.

QA Audits shall be performed as described below.

a. Functional Audits

A functional audit shall be performed in the Test Phase, by the QA organization, prior to exiting the Test Phase. The functional audit shall independently confirm that all requirements specified in the output documents from the Plant Requirements Phase and the System Requirements Phase have been met, based on the Requirements Traceability Matrix (RTM). The RTM is described in the SVVP (Section 3.10).

b. Physical Audits

A physical audit shall be performed by the QA organization in the Installation Phase to independently confirm that the application software configuration items and documents are complete and controlled in accordance with the SCMP (Section 3.11). The physical audit shall also confirm that all change requests and V&V Anomaly Reports are fully dispositioned and closed.

c. In-Process Audits

In-process audits shall be performed by the QA organization to confirm that System Requirements Phase, Design Phase, and Implementation Phase activities and outputs are being performed in accordance with this SPM and its implementing procedures.

The basic QA Audit process is as follows:

1) Management preparation
2) Planning the audit
3) Opening meeting
4) Examination (evidence collection and closing meeting)
5) Reporting

QA Audit Reports shall contain the following information, as a minimum:

1) Purpose and scope
2) Observations (strengths, or weaknesses that do not constitute a Nonconformance)
3) Findings (Nonconformance)
4) Summary Results
5) Recommendations

A QA Audit shall be considered complete when the QA Audit Report has been submitted and all actions included in the scope of the audit have been performed, reviewed and approved.

3.3.5.3 Software V&V

Independent V&V of the application software is performed to ensure that it meets the specified requirements for the application software. The scope of V&V is the application software configuration items, documents and all aspects of the target plant system that relate to the application software. V&V activities are described in the SVVP (Section 3.10).

3.3.5.4 Problem Reporting and Corrective Action

Application software hazards, problems and issues identified by management reviews, design reviews, independent V&V, QA audits and external sources (such as customer reports) shall be promptly acted upon in accordance with this SQAP and its implementing procedures.

Problem reporting and corrective action procedures shall span the entire application software life cycle described in this A-SPM. Identified application software hazards, problems and issues that constitute a condition adverse to quality, that have the potential to adversely affect safety functions and related performance characteristics listed in item (1) of Subsection 3.3.5.1, shall immediately result in initiation of a Nonconformance Report under MELCO's 10 CFR 50 Appendix B QAP.

The DTM shall analyze and report the cause of such conditions, corrective actions and preventive actions to be taken. The DT shall conduct the prescribed actions. Changes to the application software shall be controlled in accordance with this SPM. V&V activities shall be initiated in response to changes due to reported problems as described in the SVVP (Section 3.10). The QAM shall independently confirm that the required corrective and preventive actions have been implemented satisfactorily.

The QA organization is responsible for assuring compliance with this SQAP and its implementing procedures.

The QA organization shall closely monitor the disposition of application software-related QA Audit findings and Nonconformance Reports to ensure that identified findings, hazards, problems and issues are promptly and properly resolved.

3.3.5.5 Test

Application software test activities shall cover all functional and performance requirements specified in the System Requirements Phase of the application software life cycle, as described in the SVVP and the STP (Sections 3.10 and 3.12, respectively).

3.3.5.6 Code and Media Control

The application software configuration items, including media, shall be controlled in accordance with the SCMP (Section 3.11).

3.3.5.7 Training

Training activities shall be planned and conducted as described in the STrngP (Section 3.7). The roles and responsibilities of each organization are described in this A-SPM. Training objectives shall be developed in a manner that assures high quality and reliability of the application software. Training records shall be documented and maintained.

3.3.5.8 Risk Management

Potential risks associated with the development, maintenance and assurance of high quality application software configuration items and life cycle process outputs shall be identified and managed. Risk management methods and tools are described in the SMP (Section 3.1).

3.3.5.9 Documentation

Table 3.2-1 through Table 3.2-4 lists the output documents created during the development, V&V, use and maintenance of application software. These documents are to

be checked for adequacy through the review and audits described in 3.3.5.2.

3.3.6 Record Keeping

Controlled documents and QA records produced by the activities described in this A-SPM shall be controlled under MELCO's 10 CFR 50 Appendix B QAP.

3.3.7 Methods/Tools

The following tools are used for executing the application software life cycle activities described in this A-SPM:

(1) Documented Checklists

Checklists shall be used by the DT to ensure the completeness of the design outputs from each phase of the application software life cycle process assigned to the DT. Checklists shall contain the following information, as a minimum:

Functional Characteristics

a. Accuracy

b. Functionality

c. Reliability

d. Robustness

e. Safety

f. Security

g. Timing

Process Characteristics

a. Completeness

b. Consistency

c. Correctness

d. Style

e. Traceability

f. Unambiguity

g. Verifiability

(2) Requirements Traceability Matrix (RTM)

Traceability between design documents, test documents and configuration items shall be documented by the VVT using the RTM as described in the SVVP (Section 3.10). The RTM is updated at the end of the each application software life cycle phase.

3.3.8 Standards

This SQAP complies with the following guidance and standards.

- Clause 5.3.1 of IEEE Std. 7-4.3.2-2003 (Reference 3) which is endorsed by RG 1.152 (Reference 15)
- IEEE Std. 1028-2008 (Reference 6) which is endorsed by RG 1.168 (Reference 16))
- IEEE Std. 730-1998 (Reference 7) which is referenced by IEEE Std. 7-4.3.2-2003 (Reference 4)
- Section 3.1.2 of NUREG/CR-6101 (Reference 23).
- Section C of RG 1.152 Rev. 3 (Reference 15)
- Section C of RG 1.168 Rev. 2 (Reference 16)

3.3.9 Supplier Control

All of the application software can be provided by MELCO. MELCO may purchase some services regarding preparing design outputs from a sub-vendor. The sub-vendor shall be controlled as an approved supplier of safety-related items and services, subject to the provisions of 10 CFR 50 Appendix B and 10 CFR 21.

3.4 Software Integration Plan (SIntP)

3.4.1 Purpose

This Software Integration Plan (SIntP) describes how developed application software configuration items are integrated, and how the fully integrated application software is integrated with the MELTAC platform hardware and basic software. The SIntP is performed during the Implementation Phase of the application software lifecycle to allow a complete system to be achieved for testing in the Test Phase.

System V&V Testing, described in the SVVP and STP (Sections 3.10 and 3.12, respectively), demonstrates that the integrated system correctly performs all requirements in the System Requirements Specification (SysRS), as described in the SDP (Section 3.2).

This SIntP complies with the guidance and standards identified in Section 3.4.6.

3.4.2 Organization/Responsibilities

The organization structure is described in Section 2.1.

The Design Team is responsible for the following:

(1) Integrate the application software configuration items, tested via Unit V&V Tests (as described in the SVVP and STP), together to form one Application Software Execution Data for each target CPU module in the system.

(2) Integrate the Application Software Execution Data from Step (1) into the target CPU modules.

The V&V Team is responsible for executing the Integration V&V Test Procedure as described in the SVVP and STP.

3.4.3 Measurement

The Design Team shall perform the following tasks, using the MELTAC engineering tool, before releasing the integrated system to the VVT:

(1) Confirm that all Application Software Execution Data are properly installed in the target CPU modules using the MELTAC engineering tool.

(2) Confirm that the installed Application Software Execution Data is identical to the Application Software Execution Data saved in the MELTAC engineering tool as described in Section 4.1.4.1 of “Safety System Digital Platform - MELTAC - ” (JEXU-1041-1008).

If any anomalies occur during these steps, the DT shall identify the cause and implement corrective actions.

If the anomaly indicates a software safety hazard, as described in the SSP (Section 3.9), a Nonconformance Report shall be promptly initiated as described in accordance with the

SQAP (Section 3.3).

If an application software change is determined to be required to resolve the integration anomaly, the DT shall initiate and process a Software Change Request (SCR) as described in the SCMP (Section 3.11).

3.4.4 Procedures

[

]

3.4.5 Methods/Tools

The MELTAC engineering tool shall be used for the Integration and Test activities described in this SIntP, the STP, and the SVVP.

3.4.6 Standards

This SIntP complies with the following guidance and standards.

- Sections A.1.2.8 and A.3.3.3 of IEEE Std. 1074-2006 (Reference 4) which are endorsed by RG 1.173 (Reference 20)
- Section 3.1.8 of NUREG/CR-6101 (Reference 23)
- Section C of RG 1.173 Rev. 1 (Reference 20)

3.5 Software Installation Plan (SInstP)

3.5.1 Purpose

This Software Installation Plan (SInstP) describes the methods used for installing the application software in a target system. If a software hazard arises as described in the SSP (Section 3.9), it shall be identified and reported, and necessary corrective actions shall be developed and implemented as described in the SQAP (Section 3.3).

The following activities shall be performed as described in this SInstP:

- Application software installation
- On site Inspection of the software configuration installed in the target plant system
- Acceptance of software Installation

The SInstP complies with the guidance and standards identified in Section 3.5.6.

3.5.2 Organization/Responsibilities

The organization structure is described in Section 2.1.

(1) Design Team (DT)

The DT shall perform application software installation activities as described in the SIntP (Section 3.4). The DT shall document the results of the installation activity in a report to the DTM.

(2) V&V Team (VVT)

The VVT is responsible for the Implementation and Installation Phase V&V activities, including executing the Integration V&V Test, the System V&V Test, and the Acceptance V&V Test Procedures described in the SVVP and the STP. The VVT shall provide the V&V Test Reports to the VVTM.

(3) QA organization Responsibilities

The QA organization performs QA Audits as described in the SQAP (Section 3.3) to assure DT and the VVT activities are performed as described in this SInstP.

3.5.3 Measurement

The following data shall be collected and analyzed to determine the success or failure of the installation effort of the application software:

- Status of installation activities in comparison to the schedule.
- The number of Non-conformances
- Number of installation related items on the Problem List

- The number of V&V Anomaly Reports.
- The number of Software Change Requests (SCR) initiated during installation.

3.5.4 Procedures

The necessary steps, methods and tools required for installing the application software in the development/test environment prior to Integration and System V&V Tests is described in the Software Integration Plan (SIntP). The following steps are for software installation steps that may occur in the Installation Phase of a target project. If there is no need to change the application software between the System V&V Test and the Installation Phase, it is acceptable to proceed to Section 3.5.4.4.

3.5.4.1 Installation Documents

The DT shall identify and analyze the plant installation environment of the application software for a target system which is described in the System Requirements Specification (SysRS), and provide the plant installation procedures for the application software.

3.5.4.1.1 Installation Documents

The DT shall provide the following installation documents for the application software:

(1) Installation Procedure
(2) Installation Reports

3.5.4.1.2 Installation Procedure

The Installation Procedure shall include the following information, as a minimum:

(1) Overall installation strategy.

(2) Software installation procedure.

(3) Hardware and software integration procedure

(4) System installation procedure.

(5) All MELTAC controllers are functional.

(6) The correct software versions are installed in the correct controllers.

(7) All communication links and networks are functional for all interfaced devices.

(8) Software check and test procedure, including self-testing, for after installation

Acceptance Criteria for determining the success or failure of the installation effort of the application software shall be provided in the Installation Procedure.

Installation of new or modified application software should only be performed when all functions affected by the software have been declared inoperable according to the target plant technical specifications.
When application software is involved, particularly for distributed software architectures, the determination of affected functions can depend on extremely subtle considerations, which is application specific. As a minimum, all functions performed, in part, by a given software executable should be declared inoperable if the software executable, its configuration, or its operating platform is to be altered. Interconnections of all types with other software, hardware, or human elements should also be examined. Before affected functions may be declared operable, the currently approved software, under the SCMP, must be installed according to the procedures specified in Section 3.5.4. This ensures that the intended software is installed.

3.5.4.2 Distribute Software

The validated application software stored on media as described in the SCMP (Section 3.11) and the CPU module including the basic software, shall be distributed, with the Installation Procedure, to the target plant system for installation. Note that basic software installation occurs during production of the CPU module, and is controlled under MELCO's 10 CFR 50 Appendix B QAP.

3.5.4.3 Install Software

The packaged application software, shall be installed in the target plant system according to the installation procedures.

The installation report shall document the installation results along with any problems if the DT encountered during the installation.

The Installation Reports shall include the following items:

(1) Software version

(2) Installation check results

3.5.4.4 Accept Software in Operational Environment

After the installed application software configuration items are inspected as required by the "Inspect Installation Configuration" task in the SVVP (Section 3.10) in the Installation Phase, the Acceptance V&V Test Procedure shall be executed by the VVT, as described in the SVVP and the STP.

3.5.5 Methods/Tools

Software is installed in a MELTAC controller using the MELTAC engineering tool. The method used in this Installation Phase is the same as the Implementation Phase described in Section 3.4.4.

3.5.6 Standards

This SInstP complies with the following guidance and standards.

- Clause A.1.2.4 and A.4.1 of IEEE Std. 1074-2006 (Reference 4) which is endorsed by RG 1.173 (Reference 20), with following exceptions:
  - Temporary Work-Around of Clause A.4.1 is not applicable to the application software because temporary changes are not made otherwise that would allow continuation of installation activities or test of the affected parts of the software.
  - Tailoring software of Clause A.4.1.2 is not applicable to the application software because there is no tailored software used in the application software.

- Section 3.1.8 of NUREG/CR-6101 (Reference 23)

- Section C of RG 1.173 Rev. 1 (Reference 20)

3.6 Software Maintenance Plan (SMaintP)

3.6.1 Purpose

This Software Maintenance Plan (SMaintP) describes the processes for correcting defects in the application software that are discovered during plant operations and maintenance activities. This SMaintP does not include the changes to the software derived from functional design changes, which are considered to be the development process described in the SMP (Section 3.2). Defects and non-compliance shall be reported in accordance with 10 CFR 21.

The application maintenance activities are conducted for the resolution of software defects. The software life cycle process is continued and executed, thereby treating the maintenance process as iterations of development.

This SMaintP complies with the guidance and standards identified in Section 3.6.8.

3.6.2 Organization/Responsibilities

If a defect of the target plant system is identified during the Operations and Maintenance Phase of the application software life cycle, the following organizations shall have the following responsibilities (details are described in Section 3.6.6):

(1) Plant Facility Engineering Organization

The plant facility engineering organization, trained and qualified to perform operability determination and evaluations, is responsible for evaluating operability of affected Systems, Structures and Components as described in the facility Technical Specifications, and for initiating internal and external reporting as described in plant facility procedures. Application software maintenance activities performed by the plant facility engineering organization shall be reported, evaluated and initiated in the same manner as the steps described in this SMaintP.

(2) Design Team (DT)

The DT is responsible for receiving the report described above, promptly initiating internal and external reports, and tracking identified corrective actions to closure as described in the the SQAP (Section 3.3).

The DT is responsible for responding to internal reports of an application software failures, defects or other nonconforming conditions, determining the root cause, extent of condition, and corrective actions, and implementing the identified corrective actions. This includes initiating a Notice of Defect report, if required, as described in the SQAP (Section 3.3).

(3) Verification and Validation Team (VVT)

The VVT is responsible for Maintenance Phase V&V activities as described in the SVVP (Section 3.10).

3.6.3 Risks

The DT shall assess the risks associated with maintenance related software changes as described in the SMP (Section 3.1).

3.6.4 Security

Throughout the Maintenance Phase, security management shall be performed in accordance with Section 3.1.4, which complies with RG 1.152 (Reference 15). These security controls ensure that unauthorized changes cannot be introduced during maintenance activities.

3.6.5 Measurement

Errors found during software maintenance activities should be collected, recorded and analyzed to determine the quality of the software maintenance program.

3.6.6 Procedures

The application maintenance plan describe three primary activities:
1. reporting of failures that were detected during operation,
2. correction of the defects that caused those failures, and
3. release of new versions of the application product.

The following activities shall be performed for the maintenance of the application:

3.6.6.1 Activity: Application Software Defect Detection and Reporting

The DT shall provide the methodology for identifying, assessing, and recording application software defects discovered in the target plant system during plant operation in an Operation and Maintenance Manual to be developed before the Operations and Maintenance Phase. As a minimum, the methodology in the Operation and Maintenance Manual shall include requirements for identifying the date and time of a software defect discovered in the target plant system, a brief description of the defect (including the state of the system at the time of discovery), information retrieved using the MELTAC engineering tool, and a description of immediate corrective actions taken by the operation personnel and plant facility engineering organization.

(1) Plant Facility Engineering Organization

Upon discovery of an application software defect in the target plant system, the plant facility engineering organization, which may include design engineers, system engineers, maintenance engineers or training engineers, shall identify, assess, and record defect data as described in the Operation and Maintenance Manual. This data shall be attached to an internal report to be initiated in accordance with their internal procedures for Issue Reporting and Corrective Actions.

If a fault or error of the target plant system is likely due to an application software defect or nonconforming condition, the plant facility engineering organization should promptly report the condition to MELCO. If a nonconforming condition is determined by the plant facility engineering organization to be a defect as described in 10 CFR 21, a Notice of Defect report shall be initiated in accordance with the internal procedures of the plant facility engineering organization. If an application software defect is discovered by MELCO during the Operations

and Maintenance Phase of an installed system, MELCO shall initiate a Notice of Defect report in accordance with MELCO procedures for compliance with 10 CFR 21.

The plant facility engineering organization shall also evaluate operability of affected Systems, Structures and Components in accordance with the facility Technical Specifications.

(2) Design Team (DT)

In response to the plant facility engineering organization report provided in Step (1), the DT shall promptly initiate a Nonconformance Report under MELCO's 10 CFR 50 Appendix B QAP. The DT shall have the responsibility of determining the root cause, extent of condition, and corrective actions (described in Section 2.1).

3.6.6.2 Activity: Fault Correction

The DT shall collect and analyze the report and any related operational failure data to be provided by the plant facility engineering organization, which may include design engineers, system engineers, maintenance engineers, or training engineers, as described in Section 3.6.6.1.

(1) Evaluation

The DT shall determine if any reported failures are caused by nonconforming conditions or defects in the application software. The DT shall determine the root cause, extent of condition, and corrective actions necessary to correct the nonconforming condition or defect under MELCO's 10 CFR 50 Appendix B QAP.

(2) Corrective Actions

The DT shall initiate the corrective actions identified in Step (1), above. If the corrective action requires an application software change, a Software Change Request (SCR) shall be promptly initiated as described in the SCMP (Section 3.11), and the necessary change activities shall be performed as described in this A-SPM.

The VVT shall perform the Maintenance Phase V&V activities described in the SVVP (Section 3.10), including regression analysis for the proposed application software change to determine the necessary V&V activities and tasks for the proposed change.

3.6.6.3 Activity: Release and Installation

After completion of all required application software change activities and V&V tasks through the Test Phase as described in this A-SPM, the DT shall release the new version of the application software for installation.

Installation activities shall be performed as described in the SInstP, and Application V&V Test activities shall be performed as described in the SVVP and STP (Sections 3.10 and 3.12, respectively).

3.6.6.4 Maintenance of Commercial Dedication

There are no commercial grade items used in the application software applying to the target plant system. All systems, structures and components in the target plant system use the qualified MELTAC platform including the basic software as described in “Safety System Digital Platform - MELTAC - Topical Report” (JEXU-1041-1008), are produced and maintained as basic components to comply with 10 CFR 50 Appendix B. Therefore, there is no maintenance of commercial dedication activities applicable to the target plant system.

3.6.6.5 Configuration Management

Configuration management of the application software shall be performed as described in the SCMP (Section 3.11).

3.6.7 Methods/Tools

The MELTAC engineering tool shall be used for retrieving operational data as described in Section 3.6.6.1, and for the application software change and V&V activities as described in the SVVP (Section 3.10).

3.6.8 Standards

This SMaintP complies with the following guidance and standards.

- Clause A.4.3 of IEEE Std. 1074-2006 (Reference 4) which is endorsed by RG 1.173 (Reference 20)
- Section 3.1.9 “Software Maintenance Plan” of NUREG/CR-6101 (Reference 23)
- Section C of RG 1.173 Rev. 1 (Reference 20)

Clause 5.4.2.3 of IEEE Std. 7-4.3.2-2003 (Reference 3), which is endorsed by RG1.152 (Reference 15), listed in B.3.1.6 of BTP 7-14 (Reference 1), is not applicable to this SMaintP for the reason described in Section 3.6.6.4 of this SMaintP.

3.7 Software Training Plan (STrngP)

3.7.1 Purpose

The operation and maintenance of quality software products is largely dependent upon knowledgeable and skilled personnel, including MELCO technical personnel and management as well as the potential for the plant facility personnel to be qualified to install, operate and maintain the software. Training is therefore essential for plant facility personnel. This StrngP describes plant facility personnel training for the MELTAC platform and the application software.

This STrngP complies with the guidance and standards identified in Section 3.7.6.

3.7.2 Organization/Responsibilities

There are two sets of organizations responsible for being trained and qualified for performing the application software lifecycle process described in this A-SPM.

(1) DT and VVT

Training is required for the Design Team (DT) and the V&V Team (VVT) personnel who are responsible for development, maintenance and V&V activities. Such training is the responsibility of the manager of each organization and team as described in Section 2.1.2.

(2) Plant Facility Personnel

Training is required for a target plant personnel, including operators, engineers and technicians who are engaged in technical support, operations, and maintenance activities for the target plant system in the Operations and Maintenance Phase. Specific training procedures for a target plant are post-development activities and are the responsibility of the plant facility personnel.

The DT shall provide plant personnel training for the application software using the training materials described in Section 3.7.4.1.1 and for the MELTAC platform as described in Section 3.7.4.1.2.

Plant facility personnel, including operators, engineers and technicians selected by the facility, shall be trained in accordance with the training program described in the facility procedures.

3.7.3 Measurement

Training effectiveness shall be measured in accordance with the plant personnel training program as described in the facility procedures.

3.7.4 Procedures

3.7.4.1 Training Activities

The following activities shall be performed:

3.7.4.1.1 Training for Application Software

(1) Develop Training Materials

The DT shall develop and maintain the training materials to be used for training plant facility personnel. Training materials shall contain information for performing technical support and the Operations and Maintenance activities described in the Operations and Maintenance Manual, which is to be delivered to the plant personnel as described in the SMaintP (Section 3.6). Training materials shall contain the following information as a minimum:

a. Purpose

b. Learning Objectives

c. Application Level Content

- Overview of the target Plant
- System Description
- Functional Overview
- Maintenance Methods
- Troubleshooting Methods

d. Suggested Test Questions (against the Learning Objectives)

(2) Train the Trainer for the Plant Personnel

The trainers for the plant personnel shall be trained using the Systematic Approach to Training methods developed by the National Academy for Training (INPO), and the materials developed in Step (1), above.

(3) Implement the Training Program

The trainers for the plant personnel qualified in accordance with the facility procedures, shall implement the training of the plant personnel, including operators, engineers and technicians selected by the facility, in accordance with this STrngP, using the training materials provided in Step (1), above.

3.7.4.1.2 Training for the MELTAC Platform

[

]

3.7.5 Resources

The application software level training is executed with training materials described in Section 3.7.4.1.1.

The equipment level training of the target plant system applying MELTAC platform, including use of the MELTAC engineering tool, is described in Section 3.7.4.1.2.

3.7.5.1 Methods and Tools

Methods and tools used to perform the application software training shall be defined in accordance with the plant personnel training program as described in the facility procedures.

3.7.5.2 Training Facilities

Operator training, qualification and licensing shall be performed in the facilities required and described by the plant personnel training program described in the facility procedures.

3.7.6 Standards

This STrngP complies with the following guidance and standards.

- Section A.5.4 of IEEE Std. 1074-2006 (Reference 4) which is endorsed by RG 1.173 (Reference 20)
- Section 3.1.10 “Software Training Plan” of NUREG/CR-6101 (Reference 23).
- Section C of RG 1.173 Rev. 1 (Reference 20).

3.8 Software Operations Plan (SOP)

3.8.1 Purpose

The purpose of the Software Operations Plan (SOP) is to define the process of operating the application software during the Operation and Maintenance Phase.

This SOP complies with the guidance and standards identified in Section 3.8.7.

3.8.2 Organization/Responsibilities

(1) Design Team (DT)

The DT is responsible for providing Operation and Maintenance Manuals.

Problem reporting and corrective actions during the Operations and Maintenance Phase shall be performed in accordance with the SMaintP (Section 3.6).

3.8.3 Security

Throughout the Operations Phase, security management shall be performed in accordance with Section 3.1.4, which complies with RG 1.152 (Reference 15). These security controls ensure that unauthorized changes cannot be introduced during operation activities.

3.8.4 Measurement

The following data shall be collected and analyzed to determine the reliability of the target plant system applying the MELTAC platform.

(1) Error rate reported by self-diagnostics

(2) Module failure rate

3.8.5 Procedures

3.8.5.1 Operations and Maintenance Manual

An Operations and Maintenance manual shall be developed by the DT. This manual shall include the following information, as a minimum:

(1) Startup and reset of MELTAC controllers

(2) De-energization of MELTAC controllers

(3) Response to failure alarms and indications

(4) Initiating and removing maintenance bypasses

(5) Periodic surveillance tests and calibration

(6) Periodic performance monitoring

(7) Periodic equipment maintenance or replacement

(8) Security access and controls

(9) Removing and installing MELTAC modules

(10) Failure reporting and corrective actions

(11) Backup procedure of all program files, including data and code

(12) Backup Interval

(13) Instruction manual of MELTAC engineering tool

3.8.5.2 Problem Reporting

Problem reporting shall be performed in accordance with the SQAP (Section 3.3).

3.8.6 Methods/Tools

The safety VDU (S-VDU) is the primary operator interfaces for operating the target plant system, and its operator interfaces will be described in Application Licensing Documentation, such as the FSAR for an operating plant or the Design Certification Document and Combined Operating Licensing (COL) Applications for new plants.

The MELTAC engineering tool is the interface for operation and maintenance activities related to routine and corrective maintenance, including retrieving and assessing diagnostic information in response to an alarm of the target system. Instructions for use of the MELTAC engineering tool shall be provided in the Operation and Maintenance Manual.

3.8.7 Standards

This SOP complies with the following guidance and standards.

- Clause 4.2 of IEEE Std. 1074-2006 (Reference 4), which is endorsed by RG 1.173 (Reference 20)
- Section C of RG 1.173 Rev. 1 (Reference 20)

3.9 Software Safety Plan (SSP)

3.9.1 Purpose

The purpose of the Software Safety Plan (SSP) is to describe methodologies to minimize the potential of a software defect jeopardizing the health and safety of the public.

The SSP ensures that critical plant requirements, such as reactor trip functions, ESFAS functions, response times and fail-safe modes, etc., are identified. These critical requirements and functions are assured through implementation of this SSP throughout the application software life cycle. The SSP assures that precautions are defined for all life cycle phases to prevent software hazards that could result in failure of these critical requirements and functions. The SSP then assures these precautions are followed in the design and Implementation Phases and for any changes to the application software during the Operations and Maintenance Phases.

The scope of this SSP is the application software and all aspects of the target plant system that relate to the application software (i.e., the plant-specific configuration of standard MELTAC platform, which is unique to a specific application software). Each life cycle process within this A-SPM, including this SSP, considers the interaction between the application configuration of the application software and the MELTAC platform, as a completely integrated system. Life cycle process activities of the MELTAC platform are defined in Section 6.0 of “Safety System Digital Platform - MELTAC - Topial Report” (JEXU-1041-1008). The Design Team (DT) shall create the application software and associated system configuration in accordance with the critical safety requirements.

This SSP defines technical methods and organizational responsibilities for specific activities which are known to enhance software safety. The aggregate of these activities is described in Sections 3.9.7 and 3.9.8 as Software Safety Management (SSM) and Software Safety Analyses (SSA). Throughout this SSP, the entire system is evaluated to determine the influence of a potential failure.

The SSA ensures the following in accordance with Section C.3 of RG 1.173:

(1) All system safety requirements are correctly described in the System Requirements Specification (SysRS), which includes the hardware requirements specification, the software requirements specification and the interface requirements specification. Requirements that are fulfilled by the MELTAC platform shall be uniquely identified.

(2) The SysRS should cover all safety requirements in the Plant requirements (for example, FSAR Chapter 7 “Instrumentation and Controls”, Chapter 15 “Transient and Accident Analyses”, and Chapter 19 “Probabilistic Risk Assessment and Severe Accident Evaluation”).

(3) No additional hazards have been introduced in subsequent life cycle process.

(4) Other software elements that may affect safety are identified.

(5) Software and system elements that affect safety are protected from adverse

influence from software and system elements that do not directly affect safety.

(6) Software hazards and their resolutions from each phase identified in the SSA are documented.

The purpose of the SSA is to ensure compliance with safety goals established for the final safety application software of the target plant. The principal safety goals associated with the target plant system are identified in the applicable licensing documentation.

This SSP complies with the guidance and standards identified in Section 3.9.9.

This SSP is conducted in accordance with Section B.3.1.9 of BTP 7-14 (Reference 1), Section of RG 1.173 (Reference 20), Sections 3.1.5 and 4.1.5 of NUREG/CR-6101 (Reference 23), and IEEE Std. 1228-1994 (Reference 8).

3.9.2 Organization/Responsibilities

Section 2.1 describes the organizational responsibilities in supporting the SSM and SSA activities. Various parts of the project organization perform the SSM described in Section 3.9.7 and the SSA listed in Section 3.9.8. The Design Team Manager (DTM) ensures that these analyses are completed in accordance with the SSP and the DT has the responsibilities listed below for the completion of the SSM and SSA activities.

(1) Obtain and allocate resources to ensure effective implementation of the SSM and the SSA within the DT scope of responsibility.

(2) Coordinate safety task planning with other organizational activities, such as development, system safety, software QA, software reliability, software configuration management, V&V, and software testing.

(3) Coordinate software safety tasks described in the SSP within the overall context of the activities described in the SPM.

(4) Coordinate technical issues related to software safety with other components of the development and support organization, within the project sponsor, or with the licensee described in the SCMP (Section 3.11).

(5) Ensure that required records are kept in accordance with the SCMP (Section 3.11) to document the conduct of the SSP.

(6) Participate in audits regarding the all life cycle activities implementing the SSP.

(7) Ensure training of safety and other appropriate personnel on the SSM and SSA methods, tools, and techniques described in the SSP.

The VVT independent from the DT, is responsible for assuring the requirements of the SSP are followed throughout the application software life cycle process. The V&V Team Engineer (VVTE), having knowledge of the safety implications of hardware, software and interfaces between them, is responsible for performing verification and validation of software safety activities performed by the DT as described in the SSP. Software V&V activities are described in the SVVP (Section 3.10).

The VVTE confirms that system documents provided as design output, such as functional requirements, functional diagrams etc., define critical software functions and software hazards that can prevent the functions, and precautions to prevent these software hazards. If the application software deficiencies are detected, the VVTE will initiate a V&V anomaly report as described in the SVVP (Section 3.10), and the Design Team Engineer (DTE) will implement corrective actions in accordance with MELCO's 10 CFR 50 Appendix B QAP.
[

] The safety officer has the authority to ultimately reject the application software if the application software cannot be shown to be safe.

3.9.3 Risks

The SSP shall be executed by the DT and VVT staff in accordance with approved procedures that implements the roles, responsibilities, activities and tasks described herein.

The general risk management requirements are described in Section 3.1.6.2, and specific risks for related requirements for the SSP are described below.

In order to identify and resolve software hazards as early as possible in the project schedule, the SSA shall be performed during each life cycle process of the application software, and shall document the result of the SSA appropriately in accordance with this A-SPM. The SSA shall assure complete and clear documentation of:

(1) Critical safety functions

(2) Potential software hazards that may adversely affect the critical safety functions, including abnormal events, conditions and malicious modifications

(3) Mitigating design features or defensive measures to reduce the hazard potential

(4) Special tests to ensure the hazard potential has been minimized

The target plant system consists of a complete integration of hardware and software, designed specifically for nuclear safety applications, and is configured using the MELTAC platform, described in "Safety System Digital Platform - MELTAC - Topical Report" (JEXU-1041-1008).

The MELTAC platform uses a comprehensive set of self-diagnostic functions to monitor system performance for internal software hazards, including software hazards due to random hardware failures and software hazards due to software errors. MELTAC's substantial nuclear operating experience helps to minimize the potential for hidden design defects in the MELTAC platform hardware or basic software.
[

]

The software safety methodology for the application software is based on the use of the MELTAC platform and the MELTAC engineering tool as described in “Safety System Digital Platform - MELTAC - Topical Report” (JEXU-1041-1008).

Section 3.9.8 describes the SSA shall be performed for the target plant system to ensure that the application software satisfies the design basis safety requirements. These activities ensure high reliability for the application software.

When combined these SSA conform to the requirements of Section C.3 “Software Safety Analyses” of RG 1.173, Sections 3.1.5 and 4.1.5 of NUREG/CR-6101 and Section 4 “Software Safety Analyses” of IEEE Std. 1228-1994.

3.9.4 Measurement

Metrics shall be maintained throughout the entire life cycle process for safety analysis deficiencies that have been discovered by the VVT. The deficiency metrics related to software safety shall be recorded and maintained together with all other deficiencies. However, metrics related to the functions of the application software shall be specifically identified. Deficiency metrics shall be periodically reported in the V&V phase summary reports.

The measurement of the success of the SSP is the passing of the Acceptance Test with all previously discovered software safety hazards corrected and closed.

3.9.5 Procedures

The life cycle process developed by the DT, and identified in Sections 3.9.7 and 3.9.8 as being pertinent to the SSM and the SSA, shall include safety requirements that are described in the SysRS.

The VVT shall confirm the adequacy of the DT outputs, with regard to all SSM and SSA requirements, as an integral part of their V&V tasks as described in the SVVP (Section 3.10). By the V&V activities and tasks described in the SVVP (Section 3.10), hazards caused by software and hazards which software is expected to control shall be identified.

As a part of the SSM and the SSA, the VVT shall ensure that all requirements imposed by plant safety analysis, system safety analysis and security vulnerability assessments can be traced to DT outputs. Traceability shall be documented using the Requirements Traceability Matrix (RTM) as described in the SVVP (Section 3.10). These tasks shall be conducted throughout the application software life cycle process.

Problems encountered in implementing the SSP or any SSM and SSA deficiencies which are identified by the VVT at any time after document or software release from the DT shall be recorded in a V&V anomary report and tracked to closure as described in the SVVP (Section 3.10).

Problems that relate to the critical safety functions of the target plant system including the application software shall be specifically identified so they can be resolved expeditiously. If the application software requires a change, the effect on all life cycle process outputs, including the critical software safety functions shall be re-evaluated by the DT and VVT. Any safety hazard discovered after completion of appropriate SSA activities is documented in accordance with the SQAP (Section 3.3) and corrected as described in Section 3.9.8.5.

The SSA summary document shall be prepared to summarize the SSA activity. The SSA summary is documented as a quality record in accordance with the SQAP (Section 3.3).

3.9.6 Methods/Tools

The following tools shall be used:

(1) Documented checklist

Checklists shall be used to identify hazards and perform software safety analysis. It shall be documented, reviewed and approved appropriately in accordance with the SMP (Section 3.2). .

(2) Traceability check sheet

Traceability shall be documented using the RTM as described in the SVVP (Section 3.10). It shall be documented, reviewed and approved appropriately in accordance with the SVVP (Section 3.10).

3.9.7 Software Safety Management (SSM)

Sections 3.9.7.1 through 3.9.7.13 describe the organization, schedule, resources, responsibilities, tools, techniques and methodologies used in the development of the application software. The SSM activities are conducted by the DT and the VVT in accordance with the standards described in Section 3.9.9.

3.9.7.1 Organization and Responsibilities

The overall organization and responsibilities plan is described in Section 2.1 and Section 3.9.2.

3.9.7.2 Resources

The resources plan is described in Section 2.1.

3.9.7.3 Staff Qualifications and Training

The staff qualifications and training plan is described in the STrngP (Section 3.7). This plan shall specify the qualifications for personnel who will perform, at a minimum, the following

tasks:

a) Define safety requirements
b) Design and implement safety-critical portions of the system
c) Perform software safety analysis tasks
d) Test safety-critical features
e) Audit software safety plan implementation
f) Perform process certification

3.9.7.4 Software Life Cycle

The software life cycle plan is described in all sections of this A-SPM.

3.9.7.5 Documentation Requirements

The documentation requirements plan is described in the SQAP (Section 3.3).

3.9.7.6 Software Safety Program Records

The application software program records plan is described in the SQAP (Section 3.3) and the SVVP (Section 3.10).

3.9.7.7 Software Configuration Management Activities

The software configuration management activities plan is described in the SCMP (Section 3.11).

3.9.7.8 Software Quality Assurance Activities

The software quality assurance activities plan is described in the SQAP (Section 3.3) and the SVVP (Section 3.10).

3.9.7.9 Software Verification and Validation Activities

The software quality assurance activities plan is described in the SVVP (Section 3.10).

3.9.7.10 Tool Support and Approval

The tool support and approval plan is described in the SDP (Section 3.2).

3.9.7.11 Previously Developed or Purchased Software

The target plant system consists of a complete integration of hardware and software, designed specifically for nuclear safety applications and is configured using the MELTAC platform as described in “Safety System Digital Platform - MELTAC - Topical Report” (JEXU-1041-1008). There are no other previously purchased or developed basic software and/or application software configuration items applicable to the target plant system.
[

]
The application software to be installed in the target plant system shall be developed and controlled by this A-SPM. The basic software to be installed in the target plant system is previously developed software as described in “Safety System Digital Platform - MELTAC - Topical Report” (JEXU-1041-1008). The DT and VVT shall confirm that the basic software installed in the target plant system has been approved through implementation of the following process.

(1) Determine the interfaces to and functionality of the previously developed software.

(2) Identify relevant documents (e.g., product specification, design documents, usage documents) that are available to the obtaining organization and determine their status.

(3) Determine the conformance of the previously developed software to published specifications.

(4) Identify the capabilities and limitations of the previously developed software with respect to the project’s requirements.

(5) Following an approved test plan, test the safety-critical features of the previously developed software independent of the project’s software.

(6) Following an approved test plan, test the safety-critical features of the previously developed software with the project’s software.

(7) Perform a risk assessment to determine if the use of the previously developed software will result in undertaking an unacceptable level of risk.

3.9.7.12 Subcontractor Management

The subcontractor management plan complies with Section 3.1.5-11 of NUREG/CR-6101 and Section 4.3.12 of IEEE Std. 1228-1994.

3.9.7.13 Process Certification

The process certification is fulfilled by performing the processes described in the SQAP (Section 3.3) and the SVVP (Section 3.10).

3.9.8 Software Safety Analysis (SSA)

Sections 3.9.8.1 through 3.9.8.5 describe the SSA for the application software life cycle process, including the SSA applied for any changes during the Operations and Maintenance Phase. When combined together, these SSA activities conform to the requirements of NUREG/CR-6101 and IEEE Std. 1228-1994 for a software hazards analysis.

Phase-specific SSA activities described in Sections 3.9.8.1 through 3.9.8.6 shall be conducted and documented within each phase in accordance with RG 1.173 (Reference 20) per the following steps:

(1) Identify Required Input Information
- Regulatory requirements and guidance
- Information reported for the system safety analysis
- Information from previous phases for the SSA
- The design information from previous and current system and software phases activities

(2) Perform Required Analyses
The analyses must ensure that:
- System safety requirements have been correctly addressed,
- No new hazards have been introduced,
- Software elements that can affect safety are identified,
- There is evidence that other software elements do not affect safety, and
- Safety problems and resolutions identified in these analyses are documented.

(3) Document Required Output Information
Information for the current phase activities shall be reported in the SSA. This information should be used for the design activities of the current life cycle phase, subsequent SSA activities, the software configuration management process, and the verification and validation process.

3.9.8.1 Plant Requirements Phase SSA

The purpose of the SSA conducted during the Plant Requirements Phase is to identify any errors or deficiencies that could contribute to a hazard and to identify system safety considerations that are not addressed in the software requirements specified in the System Requirements Specification (SysRS), which will be developed during the System Requirements Phase as described in Section 3.9.8.2. The SSA conducted during the Plant Requirements Phase, which is an application specific item, is to establish the fundamental plant critical safety characteristics as they affect the design and implementation of application software used in the target plant system.

The Plant Requirements Phase SSA is referred to as the Software Safety Analysis Preparation, as defined in Section 4.4.1 of IEEE Std. 1228-1994.

The DT shall review the plant requirements of the target plant system as the inputs of the System Requirement Phase, as described in Sections 3.9.8.1.1 through 3.9.8.1.5, to identify the following items:

(1) Scope of safety functions which will be performed by software.
(2) Interfaces between the software and the rest of the safety system.
(3) MELTAC platform design changes.

The results of the above review shall be documented in the SSA reports by the DT and shall be independently verified by the VVT.

If the scope, interfaces or the MELTAC platform design is changed, the following Plant Requirements Phase SSA shall be performed for the changed portion and the results of each SSA shall be revised and documented.

The following SSA reports, including necessary changes based on the above review by the DT, are output documents of the Plant Requirements Phase, and these documents are inputs for the SSA during the System Requirements Phase in the application software life cycle process.

- Report which describes the results of Preliminary Hazard Analysis
- Report which describes the results of Response Time Analysis
- Report which describes the results of Criticality Analysis
- Report which describes the results of Diversity and Defense-in-Depth Analysis
- Report which describes the results of FMEA and Reliability Analysis

3.9.8.1.1 Preliminary Hazard Analysis

A preliminary hazard analysis is intended to address preparatory activities associated with high-level system design, and interfaces between the application software and the rest of the design to identify hazardous system states or actions that can cause the system to enter a hazardous state as follows:

(1) An analysis performed on the entire system or any portion of the system that identifies

a. Hazardous system states

b. Sequences of actions that can cause the system to enter a hazardous state

c. Sequences of actions intended to return the system from a hazardous state to nonhazardous state

Hazardous states are the states that would prevent the target plant system from performing actions intended to mitigate the consequences of plant accidents.

(2) An evaluation of the high-level system design to identify those functions that will be performed by software and specifying the software-related actions that will be required of the software to prevent the system from entering a hazardous state, or to move the system from a hazardous state to a nonhazardous state.

(3) The interface between the application software and the rest of the target plant system.

The target plant system is implemented through the software and hardware of the MELTAC platform and other connected components such as sensors and reactor trip breakers. The MELTAC platform hazard analysis described in “MELTAC Platform Safety Analysis” (JEXU-1041-1030) establishes the preliminary hazards analysis for the MELTAC platform. This analysis confirms that the MELTAC platform can prevent hazardous system states due to any condition within the platform, including software and hardware, and the inter-division communication design. The analysis also confirms that internal hardware or software failures that result in hazardous system states can be either automatically or manually detected.
Detection allows correction before the concurrence of hazardous states in multiple divisions. The system level preliminary hazard analysis for the target plant system, including the overall system configuration, the redundancies and the interfaces among the target plant system, including interfaces between the MELTAC platform and other

components (process and actuating devices), are described in Sections 3.9.8.1.2 through 3.9.8.1.5.

3.9.8.1.2 Response Time Analysis

The response time analysis ensures that the target plant system satisfies the safety analysis performance requirements including response time allocations and assumptions described in the requirements of the target plant system.

Timing and sizing analyses of the application software and hardware requirements has been performed and all requirements and results shall be described in the requirements of the target plant system.

3.9.8.1.3 Criticality Analysis

The criticality analysis determines the functionality of each software subsystem and assigns a safety classification for each software subsystem in accordance with the safety functions. As described in Section 2.3.2, all application software shall be designed to Software Integrity Level 4, which is the highest software integrity level. Therefore, no additional functional classification analysis will be performed.

3.9.8.1.4 Diversity and Defense-in-Depth Analysis

The Diversity and Defense-in-Depth (D3) analysis is performed to (1) assess the diversity afforded within the target plant system to ensure that adequate defense-in-depth has been provided in the design, (2) verify that any credited diverse actuation systems or components, including the indications and manual controls for credited manual actions and for control of critical safety functions are diverse from the MELTAC platform used in the reactor protection and the ESF actuation functions of the target plant system, and (3) confirm that the functions provided by diverse actuation systems or components, including functions initiated by manual actions, are sufficient to mitigate anticipated operational occurrences (AOO) and postulated accidents (PA) concurrent with Common Cause Failure (CCF) of the target plant system. The results of this analysis are incorporated into succeeding phases of the software development. The application software development methodology established in this A-SPM is intended to minimize the potential for software CCF.

The D3 analysis shall be documented in the plant requirements.

3.9.8.1.5 FMEA and Reliability Analysis

The failure modes and effects analysis (FMEA) ensures that the single failure requirements associated with the system safety analysis requirements and assumptions are satisfied. The reliability analysis ensures that the safety system reliability, including consideration of software and hardware CCF, is sufficient to satisfy the safety goals of the target plant system described in the plant requirements.

The FMEA for the target plant system shall be performed under the project-specific method to establish conformance with the requirements of IEEE Std. 603-1991, specifically the single failure criterion as stated in clause 5.1, as endorsed by RG 1.153. This FMEA demonstrates that the adverse effect of any credible single failure is limited to a single division. The remaining divisions are sufficient to perform the safety functions. The FMEA

also demonstrates that all single failures are detectable; therefore, all single failures can be corrected before multiple failures would coexist in multiple divisions.

The reliability analysis of the target plant system shall be performed under the project-specific method and the summary of the result of them shall be documented in the plant requirements. The reliability of the target plant system may be credited in the PRA Fault Tree Analysis (FTA) and the PRA sensitivity analysis, which are documented in the plant requirements related to probabilistic risk assessment.

3.9.8.2 System Requirements Phase SSA

The purpose of the SSA performed during the System Requirements Phase is to evaluate potential errors and deficiencies in the requirements that may contribute to a hazard.

The SSA performed during this phase shall be referred to as the requirements safety analysis in Sections 3.2.2 and 4.2.2 of NUREG/CR-6101 and the software safety requirements analysis in Section 4.4.2 of IEEE Std. 1228-1994.

During this phase, the DT shall review the results of the Plant Requirements Phase SSA to write the System Requirements Specification (SysRS) for the software and hardware of the target plant system. The SysRS shall define key testing requirements to confirm the SSA, to the extent practical. To ensure traceability of all SSA requirements from the Plant Requirement Phase, the SysRS shall be independently verified by the VVT as described in the SVVP (Section 3.10). Traceability of the requirement shall be documented in accordance with the SVVP (Section 3.10).

Outputs of the following SSA of this phase shall be inputs for the Design Phase SSA.

3.9.8.2.1 Preliminary Hazard Analysis

The results of the MELTAC platform level and the system level preliminary hazard analyses shall be described in the SSA reports, which will be documented at the Plant Requirements Phase. These SSA reports shall account for additional system level hazards not already addressed at the MELTAC platform level (as described in Section 3.9.8.1.1) by taking into account the overall system configuration, system redundancies, and system functions. These SSA reports shall identify the basic MELTAC platform design, the high level system design and versions to identify included components, process and actuating devices to determine what the hazard analysis is done on and if it has to be re-evaluated.

The DT shall analyze all results of the preliminary hazard analysis, and provide the SysRS to comply with this analysis.

The SysRS shall be independently verified by the VVT to ensure traceability of all related requirements from the preliminary hazard analysis at the Plant Requirements Phase in accordance with the SVVP (Section 3.10).

3.9.8.2.2 Response Time Analysis

The results of the response time analysis shall be described in the SSA report, which will be documented at the Plant Requirements Phase.

The DT shall analyze all results of the response time analysis, and provide the SysRS to comply with this analysis.

The SysRS shall be independently verified by the VVT to ensure traceability of all response time requirements at the Plant Requirements Phase in accordance with the SVVP (Section 3.10).

3.9.8.2.3 Criticality Analysis

The results of the criticality analysis for any interdivision communication interfaces of the target plant system shall be described in the SSA report which will be documented during the Plant Requirements Phase.

The DT shall analyze all communication independence requirements, and provide the SysRS to comply with this analysis. The SysRS shall also reflect that all functions of the target plant system are software integrity level 4.

To ensure traceability of all communications independence requirements at the Plant Requirement Phase, the SysRS shall be independently verified by the VVT in accordance with the SVVP (Section 3.10).

3.9.8.2.4 Diversity and Defense-in-Depth Analysis

The results of the diversity and defense-in-depth analysis shall be described in the SSA report which will be documented at the Plant Requirements Phase.

The DT shall analyze all results of the diversity and defense-in-depth analysis, and provide the SysRS to comply with this analysis.

To ensure traceability of D3 requirements at the Plant Requirement Phase, the SysRS shall be independently verified by the VVT in accordance with the SVVP (Section 3.10).

3.9.8.2.5 FMEA and Reliability Analysis

The results of the FMEA and reliability analysis of the target plant system shall be described in the SSA report which will be documented at the Plant Requirements Phase.

The DT shall analyze all FMEA and reliability requirements, and provide the SysRS to comply with this analysis.

The SysRS shall be independently verified by the VVT to ensure traceability of all FMEA and reliability requirements at the Plant Requirements Phase in accordance with the SVVP (Section 3.10).

3.9.8.2.6 Application Software Specification Analysis

Safety critical requirements for the application software that have been analyzed and evaluated against the key safety qualities will have been described in the plant requirements. The DT shall review these safety critical application software requirements, confirm their applicability, and include the results in the SysRS. The SysRS describes Functional Requirements and the Functional Diagram (FD) for the functions of the target plant system. Following development of the SysRS, the SSA activities performed in the System

Requirements Phase shall identify any system safety considerations not identified in the SysRS.

The SysRS shall be independently verified by the VVT to ensure traceability of all safety critical requirements from the Plant Requirements Phase as described in the SVVP (Section 3.10).

3.9.8.3 Design and Implementation Phase SSA

Design and Implementation Phase SSA are performed twice during the software life cycle process: (1) during the Design Phase of the hardware and application software of the target plant system and (2) during the Implementation of the hardware and software of the target plant system. These SSA confirm that the safety critical functional requirements from the System Requirements Phase for the application software correctly implement the actual application without introducing new hazards.

The SSA of these phases shall be referred to as the software design safety analysis in Sections 3.3.3 and 4.3.3 of NUREG/CR-6101, the code safety analysis in Sections 3.4.1 and 4.4.2 of NUREG/CR-6101, the software safety design analysis in Section 4.4.3 of IEEE Std 1228-1994 and the software safety code analysis in Section 4.4.4 of IEEE Std 1228-1994.

The intent of the Design and Implementation Phase SSA is to provide adequate information for the DT to develop the detailed hardware and software configuration of the target plant system. Based on the Design Phase SSA described in the sections below, the DT shall generate the System Design Description (SysDD), including the hardware design description, the software design description and the interface design description.
[

]
Outputs of the following Design Phase SSA are inputs for the Test Phase SSA. Specific analyses performed during the SSA listed below.

3.9.8.3.1 Functional Analysis

Functional analysis ensures that each application software requirement and function is translated and covered correctly in the Design Phase. All safety critical functional requirements for the target plant system in the SysRS have been analyzed and evaluated by the DT, and the DT shall provide the SysDD to comply with this analysis.

The SysDD shall be independently verified by the VVT to ensure traceability of all functional requirements during the Design Phase as described in the SVVP (Section 3.10).

3.9.8.3.2 Logic Analysis

[

]

3.9.8.3.3 Response Time Analysis

The response time analysis conducted initially during the System Requirements Phase should be revised and refined, as necessary, during the Design Phase. Revisions may not be needed if the detailed design configuration is bounded by the analysis conducted during the System Requirements Phase.

The results of the Design Phase response time analysis shall be independently verified during the Design Phase to ensure traceability of all response time requirements as described in the SVVP (Section 3.10).

3.9.8.3.4 Test Specification Evaluation

The validation test specifications are developed in this phase of the application software life cycle process.

The VVT shall provide validation test specifications based on the System Requirements Phase outputs to confirm that:

(1) The fully integrated software and hardware will operate as intended.

(2) Any task, or subtask, which cannot be tested by the validation test, is documented and it has been clearly identified that there is no impact on safety, as demonstrated by other tests or analyses.

The DT may provide input to the validation test specifications for the VVT. The validation test specifications shall be independently reviewed to ensure traceability of all safety requirements from the System Requirements Phase as described in the SVVP (Section 3.10).

3.9.8.3.5 Software Safety Code Analysis

The software safety code analysis is performed during the Design Phase of the application software life cycle process to confirm that the safety-critical portions of the software design are correctly implemented in the software code, and the software coding introduces no new hazards. The Design and Implementation Phase SSA is referred to as the code safety analysis in Sections 3.4.1 and 4.4.2 of NUREG/CR-6101 and the software safety code

analysis in Section 4.4.4 of IEEE Std. 1228-1994.
[

]
The VVT will review that the outputs of this analysis comply with all the input requirements from the SysDD, including the FBD, as described in the SVVP (Section 3.10). Traceability of the requirements shall be documented using the RTM as described in the SVVP (Section 3.10).

3.9.8.4 Test Phase SSA

The Test Phase SSA is performed to confirm that the safety-critical portions of the application software design are correctly implemented in the actual application software, and the application software introduces no new hazards.

The Test Phase SSA is referred to as the integration safety analysis in Sections 3.5.2 and 4.5.2 of NUREG/CR-6101, the validation safety analysis in Section 3.6.1 and 4.6.1 of NUREG/CR-6101, the installation safety analysis in Section 3.7.5 and 4.7.1 of NUREG/CR-6101 and the software safety test analysis in Section 4.4.5 of IEEE Std. 1228-1994.

The DT and VVT shall review the SysRS and the SysDD, including the validation test specifications, etc., to perform the SSA described below during the Test Phase of the application software life cycle process. To ensure traceability of all SSA requirements from the System Requirement Phase, the outputs of the Test Phase shall be independently verified by the VVT. Traceability of the requirements shall be documented using the RTM as described in the SVVP (Section 3.10).

Test results, including safety problems and their resolution, shall be documented on a project basis. Upon completion of the SSA conducted for the Test Phase, an SSA summary document shall be prepared to summarize the software safety analysis activities.

3.9.8.4.1 Integration Test Analysis

The integration test analysis ensures the comprehensiveness of the integration testing effort. The integration testing should ensure that the hardware and the correct version of both basic and application software is installed and operating with no errors reported by the MELTAC

self-diagnostics. Manual tests shall be conducted to confirm operability of portions of the system that are not automatically tested, such as system binary/analog inputs and outputs. The integration test report for the target plant system summarizes the findings and the corrective actions resulting from the integration testing.

The input document for the integration test analysis is the SysRS, which includes the hardware and Software Requirement Specification (SRS). The DT provides the integration test specifications, procedures and reports as output documents.

The VVT will verify the integration test outputs to ensure traceability and compliance with all input requirements in accordance with the SVVP (Section 3.10).

3.9.8.4.2 Acceptance Test Analysis

The acceptance test analysis ensures that all requirements of the SysRS are validated. The acceptance testing is performed to validate that the functionality of the fully integrated system meets the design and licensee requirements. The acceptance test reports summarize the findings and the corrective actions of the system validation testing.

The input document for the acceptance test analysis is the SysRS including the hardware and software requirement specifications. The VVT provides the acceptance test specifications, procedures and reports as the output documents.

The VVT will verify the acceptance test outputs to ensure traceability and compliance with all input requirements in accordance with the SVVP (Section 3.10).

3.9.8.5 Design Change SSA

The design change SSA are performed to identify the safety critical design elements of the application software that are affected directly or indirectly by a design change request as described in the SCMP (Section 3.11). The need for design changes is typically identified during the Operations and Maintenance Phase of the software life cycle. However, any design change needed after completion of any application software life cycle process shall be included in the design change SSA.

The design change SSA shall examine the impact of the change on previously completed SSA. Previous SSA shall be repeated as necessary based on a regression analysis as described in the SCMP (Section 3.11).

This design change SSA is referred to as the change safety analysis in Section 3.8 of NUREG/CR-6101 and the software change analysis in Section 4.4.6 of IEEE Std. 1228-1994.

The design change SSA shall include the following activities:

(1) Specify the means for determining the impact of each change on safety.

(2) Specify the techniques used to determine which safety-critical software design elements (if any) are affected by changes.

(3) Specify the documentation to be revised to accurately reflect all software safety

changes.

(4) Identify the SSA that must be repeated whenever the system or its environment is modified.

(5) Specify the extent to which regression testing in Test Phase SSA of this SSP is to be performed as a consequence of modifications to the system.

The DT shall perform the design change SSA and all results of the design change SSA shall be documented as described in the SCMP (Section 3.11). All results of the design change SSA shall be independently reviewed and verified by the VVT as described in the SVVP (Section 3.10).

3.9.8.6 Post Development

The purpose of the SSA conducted during the post development phase is to define the requirements for training, deployment, monitoring, maintenance, and retirement of the safety-critical software that are required to ensure the continued safety of the system after deployment and until its orderly retirement.

3.9.8.6.1 Training

Training shall be provided in accordance with the systematic approach to training to assure safe operation of the software for a target plant. The software training plan of the post development phase is described in the STrngP (Section 3.7).

3.9.8.6.2 Deployment

(1) Installation

All SSA tasks described in Section 3.9.8 serve to assure installation of the software safety product consistent with the results of the SSA of the Plant Requirements Phase. The installation plan of the post development phase is described in the SInstP (Section 3.5).

(2) Startup and Transition

All requirements for safely starting the new system, and, if an old system is to be replaced, for making a safe transition from old system to the new system, will be described in the operation and instruction manuals of the target plant system and the MELTAC platform which will be provided for the target plant system. The operation and instruction manuals shall include, as a minimum, the following items:

a. Fallback modes for the new system

b. Startup of backup components and subsystems

c. Startup of the new systems

d. Parallel operation with backups

e. Parallel operation of the old system and the new system

f. Subsystem vs. full system operation

g. Switchover to full system operation

h. Validation of results from the new system

i. Cross validation of results between the old system and the new system

j. Fallback in the case of failure of the new system, including fallback to an old system if one exists

(3) Operation Support

Operation Support shall be provided in accordance with the systematic approach to support plan to assure safe operation of the application software for the target plant. The operation support plan of the post development phase is described in the SOP (Section 3.8).

3.9.8.6.3 Monitoring

All requirements for monitoring the safe operation of the application software are described in the plant requirements. The operation and instruction manuals for the target plant system shall include procedures for verifying the integrity of the application software after its deployment. These manuals will be provided for the target plant system. All requirements for documenting and reporting all safety concerns that are detected during the target plant operations and Instruction Phase will be described in the operation and instruction manuals.

3.9.8.6.4 Maintenance

All requirements for the maintenance of the application software will be described in the operation and instruction manuals of the target plant system and the MELTAC platform, which is provided as the unique configuration for the application software. The operation and instruction manuals of the target plant system and the MELTAC platform shall include maintenance procedures for verifying the integrity of the safety-critical software after its deployment.

The software maintenance is described in the SMaintP (Section 3.6)

3.9.8.6.5 Retirement and Notification

The retirement and notification plan is described in the SDP (Section 3.2).

3.9.9 Standards

This SSP complies with the following guidance and standards.

- Section C of RG 1.173 Rev.1 (Reference 20)
- Sections 3.1.5, 3.2.2, 3.3.3, 3.4.1, 3.5.2, 3.6.1, 3.7.5, 3.8 of NUREG/CR-6101

(Reference 23)

- All sections of IEEE Std 1228-1994 (Reference 8)

3.10 Software Verification and Validation Plan (SVVP)

3.10.1 Purpose

The Software Verification and Validation Plan (SVVP) describes the verification and validation (V&V) activities of the application software including the methodologies for each of the V&V steps.
This SVVP complies with the guidance and standards identified in Section 3.10.8.

3.10.1.1 Scope

The scope of the SVVP is the application software and all aspects of the target plant system that relate to the application software, including the project-specific configuration of the MELTAC platform for a target plant system. Throughout the application software lifecycle, including the SVVP described in this A-SPM, the DT shall consider the interaction between the project-specific application configuration of the target plant system and the MELTAC platform, as a completely integrated system. All life cycle phases of the MELTAC platform including the SVVP for the basic software, are described in Section 6.0 of “Safety System Digital Platform –MELTAC - Topical Report“ (JEXU-1041-1008).

3.10.1.2 General Description of V&V Process

This SVVP complies with the guidance and standards identified in Section 3.10.8.

The Software Integrity of the application software controlled by this A-SPM shall be assigned level 4 in accordance with position C.1 of RG 1.168 (Reference 16) unless otherwise specified.

The V&V activities for the application software shall be performed throughout the application software life cycle model illustrated in Figure 3.10-2. Each V&V activity is consistent with each phase of the life cycle, with the exception of the plant requirements phase. The Plant Requirements Phase is concluded when the activities described in Section 3.2.6.1 of this SPM are completed.

The V&V activities begin from the System Requirement Phase of the application software life cycle, which shall be regarded as the starting point of project-specific design with receiving the outputs developed during the Plant Requirement Phase.
The V&V activities are concluded at the end of the Installation Phase. The V&V activities are also initiated in response to reported problems or requested changes in the Operations and Maintenance Phase.

Each V&V activity is made up of V&V tasks as described in this SVVP. There is one V&V activity for each phase of the application software life cycle (with the exception of the Plant Requirements Phase, as described above), and there are multiple V&V tasks for each V&V activity. Each V&V activity, as shown in Figure 3.10-2, is described in a specific section of this SVVP. The inputs of each V&V tasks and V&V outputs, as well as the roles and responsibilities of the V&V individuals (by title) responsible for each V&V task are described in Figure 3.10-2. The Acquisition Phase and Supply Phase activities required by IEEE Std. 1012-2004 are involved in the Plant Requirements Phase. All V&V activities are conducted independently from design activities as described in Annex C of IEEE Std. 1012-2004.

The V&V activities are planned, scheduled, and directed by an independent V&V Team Manager (VVTM). Additional oversight is performed by the V&V Manager (VVT) and the QA Manager (QAM). V&V activities are not considered complete until the VVTM is satisfied that all tasks are complete, documented, and all identified V&V Anomaly Reports are properly disposed.

There is no project-specific or plant-specific SVVP. This SVVP serves the purpose of the V&V planning activity guided by BTP 7-14 (Reference 1), and it demonstrates how the requirements of IEEE Std. 1012-2004 are to be carried out.

3.10.2 Organization/Responsibilities

3.10.2.1 Organization

Section 2.1 describes the organization responsibilities in supporting the V&V activities. V&V activities are performed by an independent V&V Team (VVT).

The VVT shall be technically independent, managerially independent, and financially independent in accordance with IEEE Std. 1012-2004 as endorsed by RG 1.168.

3.10.2.2 Responsibilities

The VVTM is responsible for all Independent V&V activities and tasks described in this SVVP and in the SSP (Section 3.9).

The VVT is also responsible for configuration management activities on V&V output documents and verification activities on design documents as described in the SCMP (Section 3.11).

The VVTM is responsible for the initiation of V&V activities, development of Master Test Plan (MTP) and Task Manuals, management of V&V tasks, and the final review and approval of V&V reports and Master Test Report (MTR). In the System Requirement Phase, the VVTM organizes the V&V Team (VVT) and requires the VVT to prepare the MTP. After initiating the MTP, the VVT and VVTM perform the following steps for each V&V phase activity (see numbered items in Figure 3.10-1):

[

]
The MTP is maintained and improved (if necessary) by the VVTM while V&V activities are performed.
In the Installation Phase, the VVTM reviews and approves the Final V&V report and Master Test Report prepared by the VVT.

The VVTM shall confirm that the qualifications and V&V independence criteria described in 3.3.2 are met for the personnel selected for the VVT.

Figure 3.10-1 V&V Activity Flow

3.10.3 Management and Oversight of V&V Activities

Management of V&V spans all life cycle phases.

The QAM shall perform periodic assessments of the V&V process in the area of technical accomplishments, resource utilization, future planning, identified risks and lessons learned. VVT prepares V&V output documents, V&V anomaly reports and V&V phase summary reports. The summary reports are sent to the QAM. The QAM performs the review and/or audits in accordance with the SQAP (Section 3.3). V&V output documents and V&V phase summary reports shall be evaluated by the QAM to determine if decisions to proceed to the next life cycle phase are appropriate, the QAM also recommends changes if V&V activities or tasks to be improved are discovered. All V&V process improvements shall be performed in accordance with the SCMP (Section 3.11).

The application-specific project plan, as described in the SMP (Section 3.1) identifies DT technical reviews and project milestones. The costs and the resources for performing V&V activities shall be identified by the VVTM and written into the application-specific project plan at the start of initial software life cycle.

3.10.4 Risks

The risks of the project related to VVT and DT are described in Section 3.1.6.2.

V&V activities are integrated into each life cycle phase. Experience has shown that the earlier an anomaly is discovered, the easier it is to resolve. Anomalies that are detected by the VVT in each phase of the life cycle process require the issuance and disposition of a formal V&V anomaly report as described in Section 3.10.6.5.1.
[

]
The potential risks of V&V activities shall be documented by the VVTM via the V&V Task Manuals prepared for each V&V phase activity. These risks shall be based on industry experience, the target plant’s operating experience, QA audit findings, V&V Anomaly Reports, the Problem List, and project experience, and may include system risk, mechanical risk, hardware risk, size risk, complexity risk, pre-developed software risk, schedule risk, technical risks, and risks associated with program interfaces (project management, maintenance, users, etc.) risks. The Software Safety Plan (SSP) described in Section 3.9 specifically addresses risks and activities associated with critical safety functions.

The VVTM shall identify contingency plans in the V&V Task Manuals, commensurate with

identified risks, and report these contingency plans to the PJM. Contingency plans shall identify which organization is responsible for managing the risk, and the potential magnitude of any issues or problems that can emerge if not managed correctly.

3.10.5 Measurement

The VVT shall measure the effectiveness of the software development activities and describe how these metrics support the V&V objectives. These metrics should conform to the requirements in IEEE Std. 7-4.3.2-2003 Clause 5.3.1.1 (Reference 3). A key measure is the number and severity of anomalies identified by the VVT during V&V activities. V&V anomalies shall be measured, recorded, analyzed and reported. V&V anomaly severity levels shall be classified as follows:

(1) Severe (could have an impact on one or more critical safety functions as described in the SSP (Section 3.9)

(2) Major (could affect one or more non-critical functions)

(3) Minor (no effect on any critical or non-critical functions)

V&V phase activities are considered complete only if all V&V Anomaly Reports are resolved, and V&V output documents are approved by the VVTM. Evaluation criteria for the V&V tasks associated with each V&V phase activity are described in Section 3.10.6.1 to Section 3.10.6.4.

3.10.6 Procedures

3.10.6.1 Scope

The scope of the PSMS application software V&V includes activities, tasks, and V&V output documents produced in the following life cycle phases as illustrated in Figure 3.10-2:

(1) System requirements

(2) Design

(3) Implementation

(4) Test

(5) Installation

(6) Operation and Maintenance

In the course of the Plant Requirements Phase illustrated in Figure 3.10-2, there are no V&V activities, tasks or outputs for the Plant Requirements Phase as described in 3.10.1.2.

3.10.6.2 Software Integrity Level (SIL)

As described in 3.10.1.2, the Software Integrity of the application software controlled by this A-SPM shall be assigned level 4 in accordance with C.1 of RG 1.168 (Reference 16).

3.10.6.3 V&V Activities

The following task items shall be performed during V&V activities associated with the application software.

These task items have been formulated in accordance with the items required for SIL 4 software as described in IEEE Std. 1012-2004.

3.10.6.3.1 Process: Management

The management process is comprised of the following generic activities and tasks that are applied to each V&V activity as described in this SVVP.

(1) Preparing the plans for the V&V processes (satisfied by this SVVP)

(2) Initiating the implementation of the plan (Section 3.10.2)

(3) Monitoring the execution of the plan (Sections 3.10.3 through 3.10.5)

(4) Analyzing problems discovered during the execution of the plan (Sections 3.10.2 through 3.10.5)

(5) Reporting progress of the V&V processes (Section 3.10.5)

(6) Ensuring products satisfy requirements (Section 3.10.6)

(7) Assessing evaluation results (Section 3.10.6)

(8) Determining whether a task is complete (Section 3.10.2)

(9) Checking the results for completeness (Section 3.10.6)

(10) Checking processes for efficiency and effectiveness (Section 3.10.5)

(11) Reviewing project quality (Sections 3.10.3 through 3.10.5)

(12) Reviewing project risks (Section 3.10.4)

(13) Reviewing project measures (Section 3.10.5)

3.10.6.3.1.1 Activity: Management of the V&V Effort

This activity comprises continual examination of V&V outputs, and any revisions of this SVVP that may be determined necessary by the VVTM. Figure 3.10-2 provides an overall illustration of the relationship between design activities and V&V activities in the application software life cycle.

Figure 3.10-2 Overview of Application Software V&V Activities, Tasks and Outputs

3.10.6.3.1.1.1 Tasks

(1) Prepare SVVP

a. Develop the V&V plan for all life cycle processes of the target application software. This step is completed as evidenced by this SVVP.

(2) Proposed change assessment

a. Proposed software changes, including changes to design documents shall be evaluated by the DT and the VVT. Changes can arise from proposed modifications, enhancements, and additions as a result of anomaly corrections or requirement changes. The change evaluation shall determine the effects on the reference system designs, this SVVP, and previously completed V&V activities.

b. The change assessment shall reiterate recurring tasks or initiate a revision to this SVVP to address changes to SVVP activities or tasks as required.

c. Changes to the reference design or a plant-specific design shall be verified and validated in accordance with this SVVP.

(3) Management review of V&V activities

a. The VVTM shall periodically assess and summarize V&V activities to determine if any V&V task changes are necessary, or to redirect VVT members on any specific V&V tasks.

b. The VVTM shall recommend whether to proceed to the next phase of the development life cycle and associated V&V tasks, provide V&V outputs, including V&V Anomaly Reports, and the V&V Phase Summary Report to the organizations identified in Figure 3.10-1 of this SVVP.

c. The VVTM shall verify that all V&V tasks conform to task requirements defined in the SVVP.

d. The VVTM shall verify that V&V task results have a basis of evidence supporting the results.

e. The VVTM shall assess all V&V results and provide recommendations for software product acceptance and certification. This assessment shall be an input to and described in the V&V Final Report.

The management reviews of V&V uses the review methodology in accordance with the SQAP (Section 3.3).

(4) Project management and technical review support

a. Project management support tasks:

The PJM shall check whether the release and updating of the design team

documents subject to V&V have been performed in the proper sequence.

1) Task criterion: Release dates of the design output documents subject to V&V shall be later than the release dates of design input documents.

b. Technical review tasks:

1) The DTM shall confirm that internal design reviews have been adequately performed for the design output documents subject to V&V.

2) As necessary, the DTM shall attend technical evaluation review meetings convened in the course of design activities, and assess design review reports for clarity, completeness, and timeliness.

The technical reviews use the review methodology described in the SQAP (Section 3.3).

3.10.6.3.2 Process: Development (Initial and Changes)

The development process phases are illustrated in Figure 3.10-2. V&V activities are performed to verify and validate the software configuration items and documents produced during these life cycle phases.

3.10.6.3.2.1 Activity: System Requirements Phase V&V

The system requirements V&V activity addresses software requirements analysis. The objective of this V&V activity is to assure the appropriateness, completeness, correctness, testability, and consistency of the specified requirements.

The eight V&V topics listed in Clause 7.5.1 of IEEE Std. 1012-2004 are described below:

(1) System Requirements Phase V&V tasks

a. Prepare V&V Task Manual

b. System Requirements Phase traceability analysis

c. System requirements evaluation

d. Interface analysis

e. System Requirements Phase Software Safety Analysis V&V

f. Prepare System V&V Test Specification

g. Prepare Acceptance V&V Test Specification

h. System Requirements Phase V&V Anomaly Reports

i. System Requirements Phase V&V Summary Report

(2) System Requirements Phase V&V methods and procedures

a. Prepare V&V Task Manual

The VVTM shall prepare a Task Manual for this activity as described in Section 3.10.2. The VVTM shall use the application-specific project plan and the V&V input documents described in Figure 3.10-2 to develop the Task Manual.

b. System Requirements Phase traceability analysis

The VVT shall record the Requirements Traceability Matrix (RTM) as described in Section 3.3.

The VVT shall trace the requirements specified in the system requirements specification to the Plant requirements that are used as design inputs of the System Requirements Phase, and analyze the identified relationships for correctness, consistency, completeness, and accuracy:

c. System requirements evaluation

The VVT shall evaluate the software requirements specified in the SysRS and verify that they are correct, complete, accurate, readable, unambiguous, and can be verified or validated in later life cycle phases. The examples of the software requirements specified in the SysRS are described as follows.

- Functional
- Performance
- Safety
- Security
- Human factors
- Data definitions
- User documentation
- Installation and acceptance
- User operation
- User maintenance

d. Interface analysis

The VVT shall evaluate the interface requirements specified in the SysRS and verify that they are correct, complete, accurate, readable, unambiguous, and can be verified or validated in later life cycle phases. The examples of the interface requirements specified in the SysRS are described as follows.

- Hardware
- Operator
- Maintenance
- Interface type
- Interface characteristics
- Safety
- Security

e. System Requirements Phase Software Safety Analysis V&V

The VVT shall evaluate the outputs of the SSA activities described in the SSP (Section 3.9) and verify that they are correct, complete, accurate, readable, unambiguous, and can be verified or validated in later life cycle phases.

f. Prepare System V&V Test Specification

The VVT shall develop a System V&V Test Specification as described in Section 3.12 “STP” for validating software requirements in the Test Phase. The system V&V test design shall describe the specific system-level test activities, target hardware system, input conditions and constraints, expected results and acceptance criteria. The System V&V Test Specification shall enable tracing of requirements specified in the system requirements specification to system level test designs, test cases, test procedures and test reports.

The VVT shall verify that the System V&V Test Specification conforms to the requirements of the STP (Section 3.12).

g. Prepare Acceptance V&V Test Specification

For recurring, plant-specific initial development or change projects, the VVT shall develop an Acceptance V&V Test Specification as described in the STP (Section 3.12) for validating software requirements in the Test Phase. The Acceptance V&V Test Specification is used to validate that the software correctly implements system and software requirements in an operational environment.

The Acceptance V&V Test Specification shall describe the specific test activities, target hardware system, input conditions and constraints, expected results and acceptance criteria. The Acceptance V&V Test Specification shall enable tracing of requirements specified in the system requirements specification to system level test designs, test cases, test procedures and test reports.

The VVT shall verify that the Acceptance V&V Test Specification conforms to the requirements described in the STP (Section 3.12).

h. System Requirements Phase V&V Anomaly Reports

If the VVT detects any anomalies, they shall initiate V&V Anomaly Reports as described in Section 3.10.6.5.1.

i. System Requirements Phase V&V Summary Report

The VVTM shall prepare and issue a System Requirements Phase V&V Summary Report that describes the V&V inputs, V&V tasks, V&V outputs, and disposition of V&V Anomaly Reports.

(3) System Requirements Phase V&V inputs

The inputs of this phase are listed in Table 3.2-4 (See the description of System Requirement Phase V&V activities).

(4) System Requirements Phase V&V outputs

The outputs of this phase are listed in Table 3.2-4 (See the description of System Requirement Phase V&V activities).

(5) System Requirements Phase V&V schedule

The V&V schedule including milestones, hold points, and review schedules for each task is addressed in the Task Manual, which is described in Sections 3.10.2 and 3.10.6.6 as one of the V&V output.

(6) System Requirements Phase V&V resources

The resources for the performance of the V&V task including staffing, equipment, facilities, travel, and training are addressed in the Task Manual, which is described in Sections 3.10.2 and 3.10.6.6 as one of the V&V output.

(7) System Requirements Phase V&V risk and assumptions

The Risks (Section 3.10.4) and assumptions of each task are addressed in the Task Manual, which is described in Sections 3.10.2 and 3.10.6.6 as one of the V&V output.

(8) System Requirements Phase V&V roles and responsibilities

The roles and responsibilities of each task are addressed in the Task Manual, which is described in Sections 3.10.2 and 3.10.6.6 as one of the V&V output.

3.10.6.3.2.2 Activity: Design Phase V&V

The Design Phase V&V activities address the software architectural design and the software detailed design. The objective of this V&V activity is to demonstrate that the application software design is correct, accurate, and is a complete transformation of the software requirements and that no unintended features are introduced.

The eight V&V topics listed in Clause 7.5.1 of IEEE Std. 1012-2004 are described below:

(1) Design Phase V&V Tasks

a. Prepare V&V Task Manual

b. Design Phase traceability analysis

c. Software design evaluation

d. Interface analysis

e. Design Phase Software Safety Analysis V&V

f. Prepare Unit V&V Test Specification

g. Prepare Unit V&V Test Design

h. Prepare Integration V&V Test Specification

i. Prepare Integration V&V Test Design

j. Prepare System V&V Test Design

k. Prepare Acceptance V&V Test Design

l. Design Phase V&V Anomaly Reports

m. Design Phase V&V Summary Report

(2) Design Phase V&V Methods and procedures

a. Prepare V&V Task Manual

The VVTM shall prepare a Task Manual for this activity as described in Section 3.10.2. The VVTM shall use the application-specific project plan and the V&V input documents listed above to inform the development of the Task Manual.

b. Design Phase Traceability Analysis

The VVT shall update the Requirements Traceability Matrix (RTM) as described in Section 3.3.

The VVT shall trace the software design characteristics described in the SysDD (where FBD are included) to the SysRS (where FD are included) and analyze the identified relationships for correctness, consistency, completeness, and accuracy. The VVT shall verify that there are no characteristics described or shown in the SysDD that are not specified in the SysRS.

The VVT shall also verify that all of the requirements specified in the SysRS are fully and completely translated into the SysDD.

c. Software design evaluation

The VVT shall evaluate the software design characteristics described in the SysDD (e.g., functional, performance, safety, security, human factors, data definitions,) and verify that they are correct, complete, accurate, readable, unambiguous, and can be verified or validated in later life cycle phases.

d. Interface analysis

The VVT shall verify that the SysDD (an I/O List is included) is correct, complete, accurate, readable, unambiguous, and can be verified or validated in later life cycle phases.

e. Design Phase Software Safety Analysis V&V

The VVT shall evaluate the outputs of the Design Phase SSA activities described in Section
3.9.8.3 of the SSP (Section 3.9) and verify that they are correct, complete, accurate, readable, unambiguous, and can be verified or validated in later life cycle phases.

f. Prepare Unit V&V Test Specification

The VVT shall develop a Unit V&V Test Specification as described in the STP (Section 3.12) for validating the requirements to the software configuration items in the Implementation Phase. The Unit V&V Test Specification shall describe the specific component level test activities, the target software configuration item, input conditions and constraints, expected results and acceptance criteria. The Unit V&V Test Specification shall enable tracing of requirements specified in the Software Requirements Specification to component level test designs, test cases, test procedures and test reports.

The VVT shall verify that the Unit V&V Test Specification conforms to the requirements of the STP (Section 3.12).

g. Prepare Unit V&V Test Design

The VVT shall develop a Unit V&V Test Design as described in the STP (Section 3.12) for implementing the Unit V&V Test Specification in the Implementation Phase. The Unit V&V Test Design shall enable tracing of requirements specified in the Software Requirements Specification to component level test cases, test procedures and test reports.

The VVT shall verify that the Unit V&V Test Specification conforms to the requirements of the STP (Section 3.12).

h. Prepare Integration V&V Test Specification

The VVT shall develop an Integration V&V Test Specification as described in the STP (Section 3.12) for validating the application software requirements as specified in the SRS on a target hardware system in the Test Phase. The Integration V&V Test Specification shall describe the specific integration test activities, target hardware modules or sub-systems, input conditions and constraints, expected results and acceptance criteria. The Integration V&V Test Specification shall describe the methods, tools, resources

constraints and applicable procedures required for integration testing.

The VVT shall verify that the Integration V&V Test Specification conforms to the requirements of the STP (Section 3.12).

i. Prepare Integration V&V Test Design

The VVT shall develop an Integration V&V Test Design as described in the STP (Section 3.12) for implementing the Integration V&V Test Specification on target hardware modules or sub-systems in the Test Phase. The Integration V&V Test Design shall enable tracing of requirements specified in the Software Requirements Specification and Integration V&V Test Specification to test cases, test procedures, and test reports.

The VVT shall verify that the Integration V&V Test Specification conforms to the requirements of the STP (Section 3.12).

j. Prepare System V&V Test Design

The VVT shall develop a System V&V Test Design as described in the STP (Section 3.12) for implementing the System V&V Test Specification on a target hardware system. The System V&V Test Design shall describe the specific system-level test activities, target system, input conditions and constraints, expected results and acceptance criteria. The System V&V Test Design shall enable tracing of requirements specified in the System Requirements Specification and System V&V Test Specification to system-level test cases, test procedures and test reports.

The VVT shall verify that the System V&V Test Design conforms to the requirements of the STP (Section 3.12).

k. Prepare Acceptance V&V Test Design

For recurring, plant-specific initial development or change projects, the VVT shall develop an Acceptance V&V Test Design as described in the STP (Section 3.12) for validating software requirements in the Test Phase. The Acceptance V&V Test Design is used to validate that the software correctly implements system and software requirements in an operational environment.

The Acceptance V&V Test Design shall enable tracing of requirements specified in the System Requirements Specification to system level test cases, test procedures and test reports.

The VVT shall verify that the Acceptance V&V Test Design conforms to the requirements of the STP (Section 3.12).

l. Design Phase V&V Anomaly Reports

If the VVT detects any anomalies, they shall initiate V&V Anomaly Reports as described in Section 3.10.6.5.1.

m. Design Phase V&V Summary Report

The VVTM shall prepare and issue a Design Phase V&V Summary Report that describes the V&V inputs, V&V tasks, V&V outputs, and disposition of V&V Anomaly Reports.

(3) Design Phase V&V inputs

The inputs of this phase are listed in Table 3.2-4 (See the description of Design Phase V&V activities).

(4) Design Phase V&V outputs

The outputs of this phase are listed in Table 3.2-4 (See the description of Design Phase V&V activities).

(5) Design Phase V&V schedule

The V&V schedule including milestones, hold points, and review schedules for each task is addressed in the Task Manual, which is described in Sections 3.10.2, 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(6) Design Phase V&V resources

The resources for the performance of the V&V task including staffing, equipment, facilities, travel, and training for each task are addressed in the Task Manual, which is described in Sections 3.10.2, 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(7) Design Phase V&V risk and assumptions

The Risks (Section 3.10.4) and assumptions for each task are addressed in the Task Manual, which is described in Sections 3.10.2, 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(8) Design Phase V&V roles and responsibilities

The roles and responsibilities for each task are addressed in the Task Manual, which is described in Sections 3.10.2, 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

3.10.6.3.2.3 Activity: Implementation Phase V&V

The Implementation Phase V&V activities address the application source code and execution modules that run on the MELTAC platform. The objective of this V&V activity is to demonstrate that the source code and execution modules are correct, accurate, and are a complete transformation of the software design with no unintended features introduced.

The eight V&V topics listed in Clause 7.5.1 of IEEE Std. 1012-2004 are described below.

(1) Implementation Phase V&V tasks

a. Prepare V&V Task Manual

b. Implementation Phase traceability analysis

c. Code evaluation

d. Interface analysis

e. Prepare Unit V&V Test Cases

f. Prepare Integration V&V Test Cases

g. Prepare System V&V Test Cases

h. Prepare Acceptance V&V Test Cases

i. Prepare Unit V&V Test Procedures

j. Prepare Integration V&V Test Procedures

k. Prepare System V&V Test Procedures

l. Execute Unit V&V Tests

m. Implementation Phase V&V Anomaly Reports

n. Implementation Phase V&V Summary Report

(2) Implementation Phase V&V methods and procedures

a. Prepare V&V Task Manual

The VVTM shall prepare a Task Manual for this activity as described in Section 3.10.2. The VVTM shall use the application-specific project plan and the V&V Input documents listed above to inform the development of the Task Manual.

b. Implementation Phase traceability analysis

[

]

d. Interface analysis

The VVT shall verify that the software execution module interfaces with hardware, users, operators, and other systems are correct, consistent, complete, accurate, and can be validated.

e. Prepare Unit V&V Test Cases

The VVT shall develop Unit V&V Test Cases as described in Section 3.12. The Test Cases shall implement the Unit V&V Test Design, and enable tracing to component-level test procedures and test reports.

The VVT shall verify that the Unit V&V Test Cases conform to the requirements of the STP (Section 3.12).

f. Prepare Integration V&V Test Cases

The VVT shall develop Integration V&V Test Cases. The Test Cases shall implement the Integration V&V Test Design, and enable tracing to integration test procedures and test reports.

The VVT shall verify that the Integration V&V Test Cases conform to the requirements of the STP (Section 3.12).

g. Prepare System V&V Test Cases

The VVT shall develop System V&V Test Cases as described in the STP (Section 3.12). The Test Cases shall implement the System V&V Test Design, and enable tracing to system-level test procedures and test reports.

The VVT shall verify that the System V&V Test Cases conform to the requirements of the STP (Section 3.12).

h. Prepare Acceptance V&V Test Cases

For plant-specific projects, the VVT shall develop Acceptance V&V Test Cases as described in the STP (Section 3.12). The Test Cases shall implement the Acceptance V&V Test Design, and enable tracing to system-level test procedures and test reports.

The VVT shall verify that the Acceptance V&V Test Cases conform to the requirements of the STP (Section 3.12).

i. Prepare Unit V&V Test Procedures

The VVT shall develop Unit V&V Test Procedures as described in the STP (Section 3.12). The Test Procedures shall implement the Unit V&V Test Design and Test Cases, and enable tracing to component-level test reports.

The VVT shall verify that the Unit V&V Test Procedures conform to the requirements of the STP (Section 3.12).

j. Prepare Integration V&V Test Procedures

The VVT shall develop Integration V&V Test Procedures as described in the STP (Section 3.12). The Test Procedures shall implement the Integration V&V Test Design and Test Cases, and enable tracing to integration test reports.

The VVT shall verify that the Integration V&V Test Procedures conform to the requirements of the STP (Section 3.12).

k. Prepare System V&V Test Procedures

The VVT shall develop System V&V Test Procedures as described in the STP (Section 3.12). The Test Procedures shall implement the System V&V Test Design and Test Cases, and enable tracing to system-level test reports.

The VVT shall verify that the System V&V Test Procedures conform to the requirements of the STP (Section 3.12).

l. Execute Unit V&V Tests

The VVT shall execute the Unit V&V Tests and record the results in accordance with the Unit V&V Test Procedures.

The VVT shall validate that the test results demonstrate that the target Application Software Execution Data correctly implement the design requirements.

The VVT shall validate that the test results are traceable to the Unit V&V Test Specification acceptance criteria, and that the test results meet the acceptance criteria.

Any discrepancies between the actual and expected test results shall be documented by way of a V&V Anomaly Report.

The VVT shall prepare a Unit V&V Test Report as described in the STP (Section 3.12).

m. Implementation Phase V&V Anomaly Reports

If the VVT detects any anomalies, they shall initiate V&V Anomaly Reports as described in Section 3.10.6.5.1.

n. Implementation Phase V&V Summary Report

The VVTM shall prepare and issue an Implementation Phase V&V Summary Report that describes the V&V inputs, V&V tasks, V&V outputs, and disposition of V&V Anomaly

Reports

(3) Implementation Phase V&V inputs

The inputs of this phase are listed in Table 3.2-4.

(4) Implementation Phase V&V outputs

The outputs of this phase are listed in Table 3.2-4.

(5) Implementation Phase V&V schedule

The V&V schedule including milestones, hold points, and review schedule for each task is addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(6) Implementation Phase V&V resources

The resources for the performance of the V&V task including staffing, equipment, facilities, travel, and training for each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(7) Implementation Phase V&V risk and assumptions

The Risks (Section 3.10.4) and assumptions of each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(8) Implementation Phase V&V roles and responsibilities

The roles and responsibilities of each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

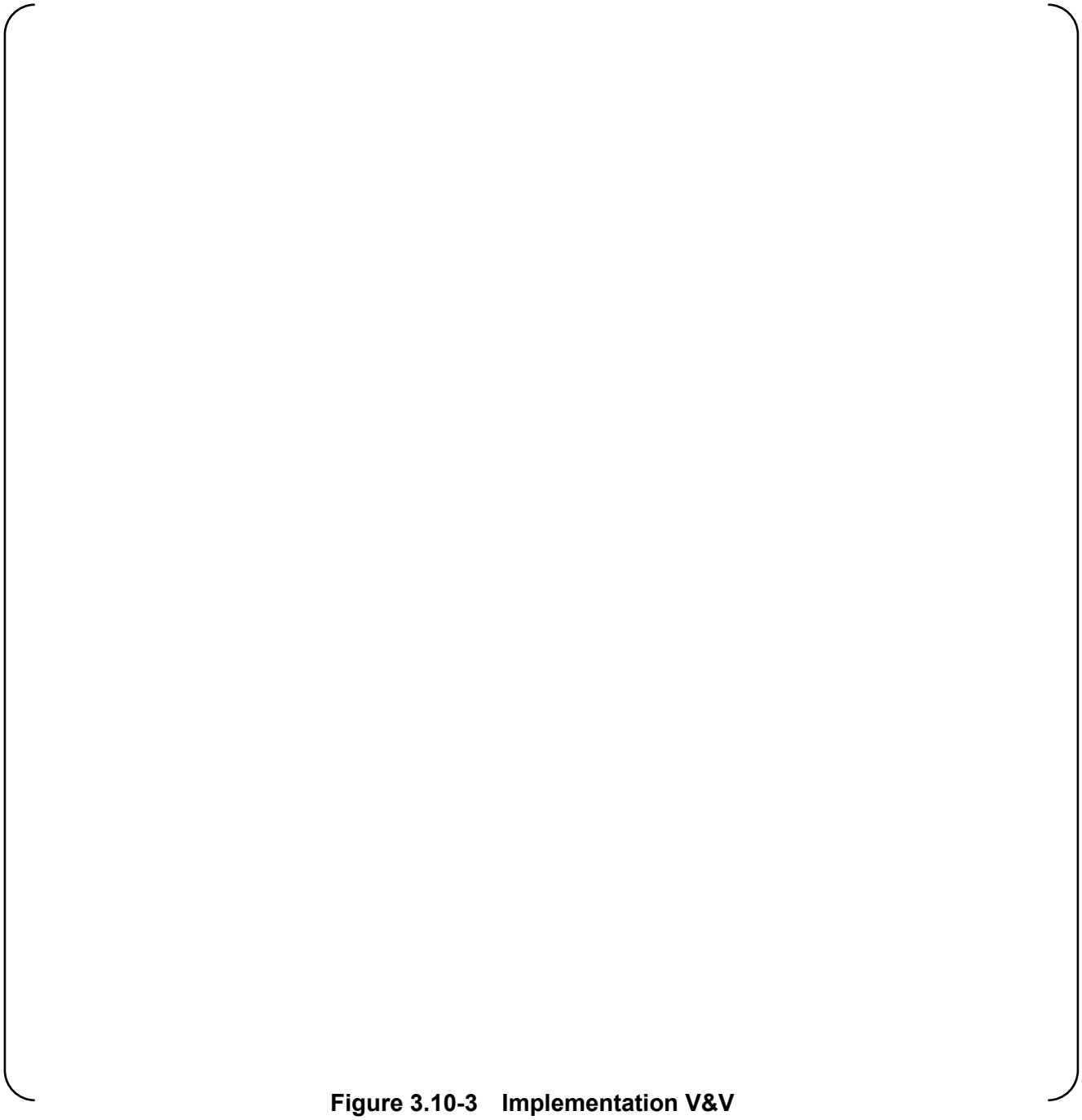


Figure 3.10-3 Implementation V&V

3.10.6.3.2.4 Activity: Test Phase V&V

The Test Phase V&V activities assure that the software requirements and systems requirements allocated to software are properly designed and implemented by execution of integration, system, and acceptance test activities on the application source code and execution modules running on the MELTAC platform. This V&V activity requires the execution of system and acceptance test specifications, test designs, test cases and test procedures prepared in earlier life cycle phases.

The eight V&V topics listed in Clause 7.5.1 of IEEE Std. 1012-2004 are described below.

(1) Test Phase V&V tasks

a. Prepare V&V Task Manual

b. Test Phase traceability analysis

c. Test Phase Software Safety Analysis V&V

d. Execute integration V&V test

e. Execute System V&V Test

f. Prepare Acceptance V&V Test Procedure

g. Test Phase V&V Anomaly Reports

h. Test Phase V&V Summary Report

(2) Test Phase V&V Methods and procedures

a. Prepare V&V Task Manual

The VVTM shall prepare a Task Manual for this activity as described in Section 3.10.2. The VVTM shall use the application-specific project plan and the V&V Input documents listed above to inform the development of the Task Manual.

b. Test Phase Traceability Analysis

The VVT shall update the Requirements Traceability Matrix (RTM) as described in Section 3.3.

1) Integration Test Traceability

The VVT shall trace the integration test characteristics in the Integration Test Cases and Procedures to the Integration V&V Test Specification and the Integration V&V Test Design, and analyze the identified relationships for correctness, consistency, completeness, and accuracy. The VVT shall verify that there are no characteristics described or shown in the Integration Test Cases and Procedures that are not specified or described in the Integration V&V Test Specification and the Integration V&V Test

Design.

The VVT shall also verify that all of the test requirements and design characteristics described in the Integration V&V Test Specification and the Integration V&V Test Design are described in the Integration Test Cases and Procedures.

2) System Test Traceability

The VVT shall trace the system test characteristics in the System Test Cases and Procedures to the System V&V Test Specification and the System V&V Test Design, and analyze the identified relationships for correctness, consistency, completeness, and accuracy. The VVT shall verify that there are no characteristics described or shown in the System Test Cases and Procedures that are not specified or described in the System V&V Test Specification and the System V&V Test Design.

The VVT shall also verify that all of the test requirements and design characteristics described in the System V&V Test Specification and the System V&V Test Design are described in the System Test Cases and Procedures.

3) Acceptance Test Traceability

The VVT shall trace the acceptance test characteristics in the Acceptance Test Cases to the Acceptance V&V Test Specification and the Acceptance V&V Test Design, and analyze the identified relationships for correctness, consistency, completeness, and accuracy. The VVT shall verify that there are no characteristics described or shown in the Acceptance Test Cases that are not specified or described in the Acceptance V&V Test Specification and the Acceptance V&V Test Design.

The VVT shall also verify that all of the test requirements and design characteristics described in the Acceptance V&V Test Specification and the Acceptance V&V Test Design are described in the Acceptance Test Cases.

c. Test Phase Software Safety Analysis V&V

The VVT shall evaluate the outputs of the SSA activities described in Section 3.9.8.4 and verify that they are correct, complete, accurate, readable, unambiguous, and can be verified or validated in later life cycle phases.

d. Execute Integration V&V Test Procedures

The VVT shall execute the Integration V&V Tests and record the results in accordance with the Integration V&V Test Procedures.

The VVT shall validate that the integration test results demonstrate that the integrated system correctly implements the Software Requirements Specification and the Integration V&V Test Specification.

The VVT shall validate that the integration test results are traceable to the Integration V&V Test Specification acceptance criteria, and that the test results meet the acceptance criteria.

Any discrepancies between the actual and expected test results shall be documented by way of a V&V Anomaly Report.

The VVT shall prepare an Integration V&V Test Report as described in the STP (Section 3.12).

e. Execute System V&V Test

The VVT shall execute the System V&V Tests and record the results in accordance with the System V&V Test Procedures. The System V&V Test is conducted in the factory environment, and may be witnessed by customer.
The VVT shall validate that the system test results demonstrate that the integrated system correctly implements the System Requirements Specification and the System V&V Test Specification.
The VVT shall validate that the system test results are traceable to the System V&V Test Specification acceptance criteria, and that the test results meet the acceptance criteria.

Any discrepancies between the actual and expected test results shall be documented by way of a V&V Anomaly Report.

The VVT shall prepare a System V&V Test Report as described in the STP (Section 3.12).

f. Prepare Acceptance V&V Test Procedure

For plant-specific initial development or change projects, the VVT shall develop Acceptance V&V Test Procedures as described in the STP (Section 3.12). The Acceptance V&V Test Procedures shall implement the Acceptance V&V Test Design and Test Cases, and enable tracing to Acceptance Test Reports.

The VVT shall verify that the Acceptance V&V Test Procedures conform to the test procedure requirements described in the STP (Section 3.12).

g. Test Phase V&V Anomaly Reports

If the VVT detects any other anomalies, they shall initiate V&V Anomaly Reports as described in Section 3.10.6.5.1.

h. Test Phase V&V Summary Report

The VVTM shall prepare and issue a Test Phase V&V Summary Report that describes the V&V inputs, V&V tasks, V&V outputs, and disposition of V&V Anomaly Reports.

(3) Test Phase V&V Inputs

The inputs of this phase are listed in Table 3.2-4.

(4) Test Phase V&V Outputs

The outputs of this phase are listed in Table 3.2-4.

(5) Test Phase V&V Schedule

The V&V schedule including milestones, hold points, and review schedules for each task is addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(6) Test Phase V&V Resources

The resources for the performance of the V&V task including staffing, equipment, facilities, travel, and training are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(7) Test Phase V&V Risk and assumptions

The Risks (Section 3.10.4) and assumptions of each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(8) Test Phase V&V Roles and responsibilities

The roles and responsibilities of each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

3.10.6.3.2.5 Activity: Installation Phase V&V

The Installation V&V effort supports the system installation and software acceptance activities.

The objective of the Installation V&V activity is to verify and validate the correctness of the software installation in the target system environment.

The eight V&V topics listed in Clause 7.5.1 of IEEE Std. 1012-2004 are described below:

(1) V&V Tasks

a. Prepare V&V Task Manual

b. Installation Configuration Inspection

c. Execute Acceptance V&V Test

d. Installation Phase V&V Anomaly Reports

e. Prepare Installation Phase V&V Summary Report

f. Prepare Final V&V Report

(2) Methods and procedures

a. Prepare V&V Task Manual

The VVTM shall prepare a Task Manual for this activity as described in Section 3.10.2. The VVTM shall use the application-specific project plan and the V&V Input documents listed

above to inform the development of the Task Manual.

b. Installation Configuration Inspection

The VVT shall inspect the installed system and its configuration items and verify they match the Installation Configuration Listing. The VVT shall confirm there are no hardware or software configuration items in the installed systems that are unlisted in the Installation Configuration Listing.

c. Execute Acceptance V&V Test

For plant-specific initial development or change projects, the VVT shall execute the Acceptance V&V Tests and record the results in accordance with the Acceptance V&V Test Procedures. The Acceptance V&V Test constitutes customer acceptance.

The VVT shall validate that the acceptance test results demonstrate that the system to be delivered correctly implements the System Requirements Specification and the Acceptance V&V Test Specification.

The VVT shall validate that the system test results are traceable to the System V&V Test Specification acceptance criteria, and that the test results meet the acceptance criteria.

Any discrepancies between the actual and expected test results shall be documented by way of a V&V Anomaly Report.

The VVT shall prepare an Acceptance V&V Test Report as described in the STP (Section 3.12).

d. Installation Phase V&V Anomaly Reports

If the VVT detects any other anomalies, they shall initiate V&V Anomaly Reports as described in Section 3.10.6.5.1.

e. .Prepare Installation Phase V&V Summary Report

The VVTM shall prepare and issue an Installation Phase V&V Summary Report that describes the V&V inputs, V&V tasks, V&V outputs, and disposition of V&V Anomaly Reports

f. Prepare Final V&V Report

The VVTM shall prepare and issue a Final V&V Report that describes the V&V phase-specific activities, results, disposition of V&V Anomaly Reports, and lessons learned.

The Final V&V Report shall also provide an assessment of the overall software life cycle and recommendations, if needed, for updating this SVVP, the SPM, or implementing procedures.

(3) Inputs

The inputs of this phase are listed in Table 3.2-4.

(4) Outputs

The outputs of this phase are listed in Table 3.2-4.

(5) Installation Phase V&V Schedule

The V&V schedule including milestones, hold points, and review schedules for each task is addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(6) Installation Phase V&V Resources

The resources for the performance of the V&V task including staffing, equipment, facilities, travel, and training for each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(7) Installation Phase V&V Risk and assumptions

The Risks (Section 3.10.4) and assumptions for each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(8) Installation Phase V&V Roles and responsibilities

The roles and responsibilities for each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

3.10.6.3.3 Process: Operation and Maintenance

3.10.6.3.3.1 Activity: Maintenance Phase V&V

The Maintenance Phase V&V activity encompasses problem reporting and resolution, change analysis, change initiation, maintenance review/acceptance, migration, and software disposal. The objectives of the Maintenance V&V activity are to:

(1) Assess proposed changes and their impact on the software

(2) Evaluate anomalies discovered during operation

(3) Assess migration requirements and disposal requirements

(4) Initiate V&V activities

The eight V&V topics listed in Clause 7.5.1 of IEEE Std. 1012-2004 are described below:

(1) V&V Tasks

a. Prepare V&V Task Manual

b. Perform Change Evaluation

c. Anomaly Evaluation

d. Regression Analysis

(2) Methods and procedures

a. Prepare V&V Task Manual

The VVTM shall prepare a Task Manual for this activity as described in Section 3.10.2. The VVTM shall use the application-specific project plan and the V&V Input documents listed above in the development of the Task Manual.

b. Perform Change Evaluation

The VVT shall assess the proposed changes to determine the potential impact on the system requirements and design.

c. Anomaly Evaluation

If the proposed change is a result of a reported problem, the VVT shall initiate a V&V Anomaly Report as described in Section 3.10.6.5.1.

The VVT shall evaluate the reported problem and determine if any changes are necessary for this SVVP or this SPM.

The VVT shall evaluate the V&V Anomaly Report disposition proposed by the responsible department and determine if any changes are necessary for this SVVP or this SPM.

d. Regression Analysis

The VVT shall perform a Regression Analysis for the proposed change and determine the extent to which V&V activities, tasks, and testing as described in this SVVP should be performed.

The VVTM shall assess the results of the Regression Analysis and determine the required resources, budget, and other needs, and shall provide them to the PJM for inclusion in the application-specific project plan associated with the proposed change.

(3) Inputs

The inputs of this phase are described in Table 3.2-4 (See the description of Operations and Maintenance Phase V&V activities).

(4) Outputs

The inputs of this phase are described in Table 3.2-4 (See the description of Operations and Maintenance Phase V&V activities).

(5) Schedule

The V&V schedule including milestones, hold points, and review schedules for each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(6) Resources

The resources for the performance of the V&V task including staffing, equipment, facilities, travel, and training for each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(7) Risk and assumptions

The Risks (Section 3.10.4) and assumptions of each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 as one of the V&V outputs.

(8) Roles and responsibilities

The roles and responsibilities of each task are addressed in the Task Manual, which is described in Section 3.10.2, Section 3.10.6.6 and Table 3.2-4 one of the V&V outputs.

3.10.6.4 V&V Reporting Requirements

V&V activities, tasks, and results shall be documented in accordance with Clause 6.1 of IEEE Std. 1012-2004.

V&V reports shall be prepared at the conclusion of each V&V task or activity as described within this SVVP.

The V&V reports to be prepared shall consist of the following five reports:

(1) Task-specific documents and reports

a. As called out in Sections 3.10.6.3.2 and 3.10.6.3.3.

b. Any specific task identified in this SVVP that does not describe a task-specific report (e.g., an Interface Analysis) shall be described in the V&V Phase Summary report.

(2) V&V Phase Summary Reports

a. As described in Sections 3.10.6.3.2 and 3.10.6.3.3

V&V Summary Reports shall contain the following items, as a minimum:

1) Summary report number and date

2) Project name, number, and applicable phase

3) List of input documents reviewed (title, number, and revision number)

4) Phase-specific deviations from SPM as noted in the application-specific project plan

5) Description of specific V&V activities, tasks, analyses, and results

6) List of V&V Phase output documents

7) Summary of reported anomalies and their dispositions

8) Summary of lessons learned

(3) V&V Anomaly Reports

a. As described in Section 3.10.6.5.1

(4) V&V Final Report

a. As described in Section 3.10.6.3.2.5

V&V Final Reports shall contain the following items, as a minimum:

1) Summary of V&V activities at all phases

2) Summary of V&V task results

3) Summary of V&V anomalies and resolutions, including the number and severity of anomalies

4) Assessment of overall software project

5) Lessons learned/best practices

6) Recommendations

(5) V&V Optional Reports

a. Any specific V&V studies conducted on V&V activities are reported as special studies report.

3.10.6.5 V&V Administrative Requirements

The Verification and Validation Administrative Requirements used in conjunction with the V&V activities described in this SVVP are outlined in the following Subsections.

3.10.6.5.1 V&V Anomaly Reporting and Resolution

A V&V anomaly is anything observed in the documentation or operation of the software that deviates from expectations based on this SVVP, V&V reference documents (i.e., the documents to which V&V Inputs are compared), or previous technical experience and/or calculations.

The VVTM shall provide V&V Anomaly Reports to the DTM or other responsible manager for

evaluation, resolution, and disposition. The VVTM shall review the final disposition of each V&V Anomaly Report and determine if it is complete, correct, and appropriate for the identified V&V anomaly.

Any detected V&V anomalies shall be documented by way of one or more V&V Anomaly Reports in each phase of the life cycle. V&V Anomaly Reports shall contain the following information, as a minimum:

(1) Anomaly Report number

(2) Project name, number and applicable phase

(3) The date the anomaly was detected

(4) The name of the V&V engineer that detected the anomaly

(5) The V&V Activity and Task that were underway when the anomaly was detected

(6) The V&V Input document that was being verified or validated

(7) The V&V reference document that was being used for the V&V task

(8) A detailed description and summary of the anomaly

(9) The severity level of the V&V anomaly (See Section 3.10.5)

(10) The impact of the V&V anomaly

(11) The date the V&V Anomaly Report was sent to the responsible manager for resolution

(12) The final disposition of the anomaly, including documents and/or software that were affected, and the V&V activities and tasks that were performed

(13) The date when the VVTM accepts the final disposition

3.10.6.5.2 V&V Task Iterations

If any revisions or changes are made to any Design Outputs and/or V&V inputs, including documents and software configuration items, the VVTM shall determine which V&V activities and tasks must be performed again, and direct the VVT accordingly.

The VVTM shall update Task Manuals if necessary.

Identified V&V Activities and Tasks shall be repeated until the VVTM confirms all V&V Anomaly Reports are fully disposition and closed.

3.10.6.5.3 Deviations

When a deviation from the application-specific project plan, SPM, or implementing procedures is considered necessary for a given project or life cycle phase activity, the Design Team

Manager shall prepare a Deviation Request and present it to the PJM and the VVTM. Deviation Reports shall include the following:

(1) Identification of the design activity/task and V&V activity/task to be amended or deleted

(2) The basis for the requested deviation

(3) An assessment of the impact on software quality

(4) Indication of VVTM acceptance or rejection of the Deviation Request

The VVTM shall evaluate the Deviation Request and after discussion with the Design Team Manager, accept or reject the request. If any Deviation Requests are initiated, they shall be described in the V&V Phase Summary Report and the V&V Final Report.

3.10.6.5.4 Record Retention

All reports and records of V&V activities shall be retained in accordance with MELCO's 10 CFR 50 Appendix B QAP.

3.10.6.6 V&V Documentation Requirements

A Task Manual shall be contained the following items as a minimum:

(1) Specific tasks
(2) Role and responsibility of V&V members who are engaged in the specific tasks.
(3) Applicable procedures and methods
(4) Required inputs
(5) Required outputs
(6) Schedule for performing the identified tasks (including milestones, hold points and review schedule)
(7) Acceptance criteria to proceed to the next step listed in the schedule
(8) Required resources
(9) Risks involved in the identified tasks and assumptions

The test documents shall be composed of the purpose, format and contents as described in the STP (Section 3.12).

3.10.7 Methods/Tools

The basic tasks performed in each stage of the V&V process in the software life cycle are described below. The specific tasks are described for each life cycle phase in this SVVP.

In each task of the V&V process, consistency between the upstream document and the downstream document at each life cycle phase shall be verified or validated.

(1) Checking of basic software and MELTAC engineering tool

[

]

(2) V&V Procedures

The procedures that implement the requirements of this SVVP shall include a check sheet, including check results against acceptance criteria. Check sheets and results shall be documented in the associated V&V output document or V&V Phase Summary Report.
V&V results include these check sheets and results. V&V procedures shall also list the interfacing procedures for record retention and V&V anomaly reporting.

3.10.8 Standards

This SVVP complies with the following guidance and standards.

- Clause 5.3 of IEEE Std. 7-4.3.2-2003 (Reference 3) which is endorsed by RG 1.152 (Reference 15)
- IEEE Std. 1012-2004 (Reference 9) which is endorsed by RG 1.168 (Reference 16)
- Section 3.1.4 of NUREG/CR6101-1993 (Reference 23)
- Section C of RG 1.152 Rev. 3 (Reference 15)
- Section C of RG 1.168 Rev. 2 (Reference 16)
- Section C of RG 1.173 Rev. 1 (Reference 20)

3.11 Software Configuration Management Plan (SCMP)

This software configuration management plan (SCMP) describes the methods for maintaining generic and project-specific application software configuration items (CIs) in a controlled configuration.

This SCMP addresses the six classes of information required by IEEE Std. 828-2005 (Reference 11), as listed in Table 3.11-1. The referenced sections of the SCMP provide the detailed descriptions for each class of information.

Table 3.11-1 IEEE Std. 828-2005 vs. SCMP Section

Class of Information	Description	Section in IEEE Std 828-2005	Section in this SCMP
Introduction	Describes the SCMP purpose, scope of application, key terms, and references	3.1	3.11.1
SCM Management	Identifies the responsibilities and authorities for accomplishing the planned activities	3.2	3.11.2
SCM Activities	Identifies all activities to be performed in applying to the project	3.3	3.11.3
SCM Schedules	Identifies the required coordination of SCM activities with the other activities in the project	3.4	3.11.4
SCM Resources	Identifies tools and physical and human resources required for execution of the SCMP	3.5	3.11.5
SCMP Maintenance	Identifies how the SCMP will be kept current while in effect	3.6	3.11.6

3.11.1 Purpose, Scope, and Applicability

3.11.1.1 Purpose

The purpose of this SCMP is to describe the methods required for maintaining the application software CIs in a controlled configuration.

This SCMP describes the process for management of the application software CIs throughout its life cycle, which includes identifying the CIs, developing and maintaining the CI list, controlling its implementation and changes documentation, and recording and report its status. This SCMP is intended to be utilized throughout the application software life cycle.

The following minimum set of SCM activities shall be performed throughout the application software life cycle:

(1) Identification and control of the application software designs and code

(2) Identification and control of the application software design functional data (e.g., data templates and databases)

(3) Identification and control of the application software design interfaces

(4) Control of the application software design changes

(5) Control of the application software documentation (user, operating, and maintenance documentation)

(6) Control and retrieval of qualification information associated with the application software designs and code

(7) The application software configuration audits

(8) Status accounting

This SCMP complies with the guidance and standards identified in Section 3.11.12.

3.11.1.2 Scope

The CIs for the application software to which this SCMP shall be applied includes the following application software items, associated documentation, and databases. See Section 3.11.3.1 for more detail.

- System Requirement Specification (SysRS)
- System Design Description (SysDD)
- V&V related documents
- Application software

[

]

Execution of changes after software development, software V&V, software release, software test, or other activities described in this A-SPM that could impact the cost, schedule, or ability to perform defined SCM activities shall be identified using the application-specific plan, Risk Matrix, or Problem List tools described in the SMP (Section 3.1).

3.11.1.3 Key Terms

Key terms are defined here, as they apply to this SCMP in order to establish a common terminology.

The specific terms and these definitions as used within this SCMP (i.e., control point, release) are described in Appendix A. Additional terms that are relevant are defined in IEEE Std. 610.12-1990 (Reference 22), which are also described in Appendix A, and are as follows:

- Baseline
- Component
- Configuration
- Configuration Audit
- Configuration Control
- Configuration Control Board
- Configuration Identification
- Configuration Item
- Configuration Management
- Configuration Status Accounting
- Interface
- Interface Control
- Software
- Software Library
- Software Life Cycle
- Unit
- Version

3.11.2 SCM Management

This section describes the allocation of responsibilities and authorities for SCM activities. Organizations and individuals within those organizations are responsible for the application software life cycle activities and configuration items. This section includes three topics:

(1) The organizations responsible for SCM activities described in this SCMP

(2) The specific SCM responsibilities of these organizations

(3) SCM policies and directives that apply to the application software

3.11.2.1 Organizations

All organizations that participate in or are responsible for any SCM activities and relationships between organizations for the application software project are described in Section 2.1 and Figure 2.2-1.

The functional roles of these organizational units within the application software life cycle activities are also described in Section 2.1.

3.11.2.2 SCM Responsibilities

Table 3.11-2 provides a matrix that relates the organizations defined above to the SCM functions, activities, and tasks as described in this SCMP.

The DT is responsible for SCM activities on design outputs (described in the SDP; Section 3.2) that are generated by DTE.

The DTM is responsible for all SCM activities related to DT inputs and outputs described in the SDP. All SCM activities, with the exception of V&V and QA activities, shall be documented by the DT and approved by the DTM.

The DTM has the following specific responsibilities for the application software SCM:

(1) After the completion of each design activity in the application software life cycle process, the DTM releases design outputs for independent V&V as described in the SVVP.

(2) Upon completion of Implementation Phase design activities, the DTM releases the application software configuration items for Implementation Phase and Test Phase V&V activities described in the SVVP.

(3) The DTM shall release the final version of the application software after successful completion of Installation Phase V&V activities described in the SVVP.

All SCM activities assigned to the DT shall be verified by the VVT. The VVT is responsible for SCM on the specific V&V activities that they generate.

The QA organization shall perform QA audits of SCM activities to ensure adherence to this SCMP and its implementing procedures. QA audits shall be performed as described in the SQAP (Section 3.3), and shall be coordinated with the PJM, DTM, and VVTM.

The PJM and the DTM have final approval of change requests; however, the DTM is responsible for convening Change Control Board (CCB) activities as described in Section 3.11.2.2.1.

Table 3.11-2 Matrix of SCM Responsibilities

Organization / Individual	SCM Function, Activities, and Tasks
DT	- SCM for design activities and design outputs
DTM	- Coordination with QA audits - Overall responsibility of SCMP implementation of DT - Documentation and review of SCM activities - Release of the application software - Chairman and member of CCB - Final approval of change requests
VVT	- SCM on V&V activities
VVTM	- Coordination of QA audits - Overall responsibility of SCMP implementation of VVT - CCB member
QA	- Audit of SCM activities
QAM	- Overall responsibility of QA audit - CCB member
PJM	- Coordination of QA audits - Contact to Customer - CCB member

3.11.2.2.1 Configuration Control Boards (CCB)

CCBs shall be utilized prior to development of or changes to the application software under the following conditions:

- Proposed changes that affect functional or performance requirements defined in the Plant Requirements or System Requirements Phases.
- Any application software changes required during the Test Phase that affect design outputs from the System Requirements or Design Phase.

The application software CCB shall function as described in IEEE Std. 1042-1987 (i.e., software CCB focused on technical issues). The purpose of the software CCB is to control major changes, such as changes to system functions and overall configuration of the application software, before proceeding with detailed change activities in the application software life cycle phases. The application software CCB members shall be the PJM, DTM, VVTM, and QAM as a minimum. As needed, representatives from engineers their respective organizations, or other organizations, may be included on the CCB.

The DTM is the CCB chairman and calls CCB meetings when required.

The definition of a minor change is a change that does not affect functional or performance requirements, or a design modification, or changes to the application documents. Examples of these minor changes are input/output format changes, clarifications, correction of typos, etc. A CCB meeting is not required for minor changes, but minor changes are reviewed and approved by the DT and are a V&V activity. Changes to the application software are initiated and controlled using an SCR. This approach is acceptable because these changes (i.e., input/output format changes, clarifications, correction of typos, etc.) are limited by the existing functional requirements. All such changes shall be reviewed and approved as described in the SDP and SQAP (Sections 3.2 and 3.3, respectively), and require independent V&V as described in the SVVP (Section 3.10).

The application software CCB has the authority to approve or disapprove proposed change requests that require CCB activity. The CCB may also define required changes to application software CIs.

3.11.2.3 Applicable Policies, Directives, and Procedures

Applicable policies, directives, and procedures related to this SPM are described in Section 1.3.

3.11.3 SCM Activities

The SCM activities described in this SCMP are grouped into four basic functions:

(1) Configuration Identification

(2) Configuration Control

(3) Status Accounting

(4) Configuration Audits and Reviews

The minimum information requirements for each function are described in Sections 3.11.3.1 through 3.11.3.4. The requirements for interface control and subcontractor/vendor control activities are identified separately in 3.11.3.5 and 3.11.3.6.

3.11.3.1 Configuration Identification

The application software configuration items shall be identified by a unique identifier (e.g., name or number), and their physical, functional or performance characteristics shall be described in a design output document (e.g., SysRS, SysDD per the SDP).

Sections 3.11.3.1.1 and 3.11.3.1.3 describe the minimum information required for configuration identification of the items listed in Section 3.11.1.2 (2).

3.11.3.1.1 Identifying Configuration Items

A CI List for all CIs that are to be delivered and maintained for the application software shall be developed and maintained by the DT.

The CI List shall be controlled and stored in the same manner as any other application software document, and it shall retain the revision history of each CI so that it may be retrieved and so that the latest revision of each CI may be easily identified.

The DT is responsible for identification of all separately identifiable modules comprising the software CIs, along with the design output documents described in the SDP (Section 3.2).

An application software baseline shall be established at the end of Implementation Phase in the software life cycle. Approved changes that are created subsequent to a baseline shall be added to the baseline. The application software baseline is described in the SDP (Section 3.2).

3.11.3.1.2 Naming Configuration Items

The application software documents described in this A-SPM shall be uniquely identified by name or number, with revision levels, and they shall be controlled and stored in accordance with the document control and record keeping requirements described in MELCO’s 10 CFR 50 Appendix B QAP.

The application software CIs stored on media shall be uniquely identified and labeled when they are released by the DT. The labeling of application software CIs and media shall include unique identification of each CI, and revision and/or date time stamps for each CI. Configuration management sheets shall record the unique identifiers and versions of the released configuration items to be stored.

3.11.3.1.3 Control of Configuration Items

The format, location, and documentation requirements of the application software CIs as described in this SCMP shall be controlled under implementing procedures. Access control procedures shall specify the storage locations and storage requirements of documents and

media, including the physical marking and labeling of items.

Archived copies of each baseline of the application software used or created for the target projects shall be kept in a fire-proof software library, and additional back-up copies shall be created and stored in a separate location for disaster recovery. Storage media shall be clearly and indelibly marked for easy and unambiguous identification. The versions of the basic software and development tools for application software used on each release of the application software shall be controlled and recorded.

The DT shall have the control of software libraries. The software libraries shall be stored in a specific, secure, and controlled storage area.

The status accounting of the application software CIs is described in Section 3.11.3.3.

3.11.3.2 Configuration Changes

An application software baseline shall be established at the Implementation Phase in the software life cycle. Approved changes that are created subsequent to a baseline shall be added to the baseline. The application software baseline is described in the SDP (Section 3.2).

Changes to the application software shall be formally documented and approved as described in this SCMP. The documentation shall include the reason for the change, identification of the affected application software CIs, and the impact of the change on the plant design and operation. Additionally, the documentation associated with an application change shall describe the plan for implementing the change in the plant (e.g., immediately implementing the change, or scheduling the change for a future version).

Changes to CIs shall be initiated by a Software Change Request (SCR) as described below.

3.11.3.2.1 SCR Initiation

The DT shall be responsible for receiving and processing SCRs, which may be initiated by any organization described in this A-SPM.

The minimum information required in an SCR is as follows:

(1) Originator’s name and organization

(2) Date of request

(3) The name(s) and version(s) of the affected application software CIs

(4) SysDD (Functional Block Diagram, etc.) affected

(5) The need for the change

(6) Description of the requested change

(7) Associated V&V anomaly reports or nonconformance reports (if any)

(8) Requested completion date

3.11.3.2.2 SCR Evaluation

The PJM, DTM, and VVTM shall each evaluate the requested change and independently determine the potential impact of the proposed change. If the proposed change requires CCB review and approval, the DTM shall call a CCB meeting as described in Section 3.11.2.2.1.

3.11.3.2.3 SCR Approval or Disapproval

The PJM is responsible for approving or disapproving SCRs with the exception of those that require CCB approval.

3.11.3.2.4 SCR Implementation

If an SCR is approved, the PJM shall prepare an application-specific plan for new projects to implement of SCR as described in the SMP (Section 3.1). If the SCR is associated with an active project, the PJM shall update the application-specific plan as necessary.

3.11.3.3 Configuration Status Accounting

The application software CIs, including documents, shall be recorded on Configuration Management Sheets that include the following information for each CI, as a minimum:

- Unique identifier and current version number
- Current status (under development, under test, or released)
- Last release date
- Associated design and test documents
- Associated V&V anomaly reports (if any)
- Associated nonconformance reports (if any)

The Configuration Management Sheets for design activities are the responsibility of the DT, and shall be verified by the VVT. The Configuration Management Sheets for V&V activities are the responsibility of the VVT. Configuration Management Sheets shall be controlled under MELCO's 10 CFR 50 Appendix B QAP.

3.11.3.4 Design Reviews and QA Audits

Design Reviews and QA Audits shall be performed as described in the SQAP (Section 3.3), to confirm that CIs conform to their required physical and functional characteristics.

3.11.3.5 Interface Control

The following interface controls describe the methods for coordinating changes to the application software CIs that may be driven by activities that are outside the scope of this SCMP. The external items which are examined for potential interfacing effects on the

application software include outputs from the Plant Requirements Phase and the basic software (controlled under Section 6.0 of JEXU-1041-1008 “Safety System Digital Platform - MELTAC - Topical Report”).

(1) Interface with Plant Requirements

The application software CIs shall conform to the requirements produced in the Plant Requirements Phase as described in the SVVP (Section 3.10). Any proposed changes to the application software that do not fully and completely meet Plant Requirements shall not proceed until the associated Plant Requirements Phase documents are changed and approved in accordance with NRC regulations.

(2) Interface with Basic Software

[

]

3.11.3.6 Subcontractor / Vendor Control

Subcontractor/vendor control is to comply with Section 3.3.6 of IEEE Std. 828-2005.

3.11.4 SCM Schedules

The application SCM activities described in this SCMP shall be performed in accordance with the schedule described in the application-specific plan.

3.11.5 SCMP Resources

The tools and procedures, equipment, personnel, and training necessary for the implementation of the SCM activities in each phase are described in Sections 3.11.9 and 3.11.11. Personnel assigned to work on the application software development projects are trained in the requirements of the SQAP and the SCMP and skilled in the use of the tools as required by their individual job functions.

3.11.6 SCMP Maintenance

This SCMP is the only means used to control the configuration of the application software. The DTM and the DT has the overall responsibility for maintaining the SCMP.

3.11.7 Security

All application software documents and software CI under configuration control shall be protected against the secure development/operational environment threats. Each organization described in this A-SPM shall be responsible for ensuring that the security related configuration controls and restrictions are maintained in accordance with Section

3.1.4, which describes how security controls comply with RG 1.152 (Reference 15).

3.11.8 Measurement

SCM activities shall be measured and trended as follows:

- Number of SCRs
- Number of V&V anomaly reports related to SCM activities
- Security level of V&V anomaly reports related to SCM activities
- Number of nonconformance reports related to SCM activities

The DTM and VVTM shall periodically review these measures, and if an adverse trend is detected, shall initiate additional corrective actions.

3.11.9 Procedures

In addition to the SCM activities described above, phase-specific SCM activities are described in Sections 3.11.9.1 to 3.11.9.5.

3.11.9.1 Plant Requirements and System Requirements Phases

(1) The input documents and output documents of these phases shall be controlled under MELCO's 10 CFR 50 Appendix B QAP.

(2) Independent V&V during these phases shall be performed and documented as described in the SVVP. V&V documents shall be controlled under MELCO's 10 CFR 50 Appendix B QAP.

(3) V&V anomaly reports shall be dispositional, including changes to affected design documents, as described in the SVVP.

3.11.9.2 Design and Implementation Phases

(1) Design and implementation documents shall be controlled under MELCO's 10 CFR 50 Appendix B QAP.

(2) The application software CIs shall be controlled as described in Section 3.11.3.3.

(3) Independent V&V of design outputs from these phases shall be performed and documented as described in the SVVP. V&V documents shall be controlled under MELCO's 10 CFR 50 Appendix B QAP.

(4) V&V anomaly reports shall be dispositional, including changes to affected design documents, as described in the SVVP.

3.11.9.3 Test Phase

(1) All software/hardware configurations and design documents shall be before entering

the Test Phase.

(2) Test Phase V&V test documents shall be controlled under MELCO's 10 CFR 50 Appendix B QAP.

(3) The final as-tested application software configuration shall be documented in the V&V test reports.

(4) V&V anomaly reports shall be dispositional, including changes to affected design outputs, as described in the SVVP.

(5) SCR documents shall be controlled and used to track software changes or required enhancements. An SCR may be used to disposition more than one V&V anomaly report.

3.11.9.4 Installation Phase

(1) Ensure that all as-built documentation is under configuration control.

(2) Acceptance test specifications, procedures and reports shall be controlled under MELCO's 10 CFR 50 Appendix B QAP.

(3) V&V anomaly reports shall be dispositional, including changes to affected design outputs, as described in the SVVP.

(4) SCR documents shall be controlled and used to track software changes or required enhancements. An SCR may be used to disposition more than one V&V anomaly report.

3.11.9.5 Operations and Maintenance Phase

(1) The DTM releases the final version of the application software and related documentation to the customer after successful completion of the acceptance V&V test as described in the SVVP and the STP (Sections 3.10 and 3.12).

(2) Nonconformance reports shall be initiated in response to the application software problems reported by customers or other outside organizations. Nonconformance reports shall be assigned to a responsible organization, evaluated, and dispositioned under MELCO's 10 CFR 50 Appendix B QAP.

(3) SCRs shall be initiated in response to customer requests for changes to the application software as described in Section 3.11.3.2.

(4) The DT shall maintain the configuration status accounting of the installed the application software as described in Section 3.11.3.3.

3.11.10 Record Keeping

All records and records of SCM activities shall be prepared and retained as QA records under MELCO's 10 CFR 50 Appendix QAP.

3.11.11 Methods/Tools

The following methods/tools shall be used:

(1) Configuration Item List

A CI List shall be developed and maintained by the DT and VVT as described in Section 3.11.3.1.

(2) Configuration Management Sheet

Configuration management sheets shall be developed and maintained by the DT as described in Section 3.11.3.3.

(3) Backups

Software backups of all program files, including tools, shall be initiated by the DT when a baseline is determined and shall be updated regularly. Backup methods shall be established and maintained by the DTM. Backup files shall be kept in a separate area.

3.11.12 Standards

This SCMP complies with the following guidance and standards.

- IEEE Std. 828-2005 (Reference 11) which is endorsed by RG 1.169 Rev.1 (Reference 21).
- IEEE Std 1042-1987 (Reference 12) which is endorsed by RG 1.169 Rev.0
- Clause A.1.2.2 and A.5.2 of IEEE Std 1074-2006 (Reference 4) which is endorsed by RG 1.173 (Reference 20)
- Clause 5.3.5 of IEEE Std 7.4.3-2-2003 (Reference 3) which is endorsed by RG 1.152 (Reference 15)
- Sections 3.1.3 and 4.1.3 of NUREG/CR-6101 (Reference 23)
- Section C of RG 1.152 Rev. 3 (Reference 15)
- Section C of RG 1.169 Rev. 1 (Reference 21)
- Section C of RG 1.173 Rev. 1 (Reference 20)

Clause 5.4.2.1.3 of IEEE Std. 7-4.3.2-2003, listed in B.3.1.11 of BTP 7-14 (Reference 1) is not applicable to this SCMP. Commercial-off-the-shelf (COTS) items are not applied to the application software.

3.12 Software Test Plan (STP)

3.12.1 Purpose

This Software Test Plan (STP) complements the SVVP (Section 3.10), and provides additional details and minimum information requirements for the following V&V test activities:

(1) Unit V&V Test

(2) Integration V&V Test

(3) System V&V Test

(4) Acceptance V&V Test

All tests of the application software are executed as V&V tests. In addition, for approved changes, regression analysis tasks are performed by the VVT to determine the extent to which these four test activities may be repeated as described in the SVVP (Section 3.10).

3.12.2 Organization/Responsibilities

The VVT shall perform all test activities described in this STP and the SVVP. The VVTM is responsible for all test activities.

The organization and responsibilities of the VVT are described in Section 2.1 and 3.10.2.

The V&V team members prepare the test scripts and execute tests, and are personnel who are not involved in the software development.

3.12.3 Security

The secure development/operational environment for the testing activities described in this STP and the SVVP shall be maintained in accordance with RG 1.152 (Reference 15).

3.12.4 Measurement

Measurements of test activities are described in Section 3.10.5.

3.12.5 Procedures

Alignment with IEEE Std. 1012-2004 (Reference 9) testing activities is described in Table 3.12-1

Table 3.12-1 Alignment with IEEE Std. 1012-2004 Testing Activities

IEEE Std. 1012-2004 Testing activity	Testing activity for the application software
Component Testing	Unit V&V Test (Sections 3.10.6.3 and 3.12.5.1 (1))
Integration Testing	Integration V&V Test (Sections 3.10.6.3 and 3.12.5.1 (2))
System Testing	System V&V Test (Sections 3.10.6.3 and 3.12.5.1 (3))
Acceptance Testing	Acceptance V&V Test (Sections 3.10.6.3 and 3.12.5.1 (4))

The test documents that the VVT shall prepare and review for each test activity are described in this STP and the SVVP. The minimum required information for each test document type is described in Section 3.12.5.2. The VVTM is responsible for approving the test documents.

3.12.5.1 Testing Activities

The following tests shall be demonstrated by the requirement of clause 5.4.1 of IEEE Std. 7-4.3.2-2003.

(1) Unit V&V Test

[

]

(2) Integration V&V Test

[

]

(3) System V&V Test

[

]

(4) Acceptance V&V Test

[

]

3.12.5.2 Test Documents

The test documents for all application software tests described in this STP and the SVVP shall be prepared as described in the SVVP and this STP, via implementing procedures, in accordance with IEEE Std. 829-2008 (Reference 13) and IEEE Std. 1012-2004 (Reference 9). Test documents are listed in Tables 3.12-2, 4-1, 4-2, 4-3, and 4-4.

Table 3.12-2 Alignment with IEEE Std. 829-2008 Test Documents

IEEE Std. 829-2008 Test Document	Application Software Test Document
Master Test Plan	Master Test Plan - Section 3.12.5.2 (1)
Level Test Plan	Test Specifications - Section 3.12.5.2 (2)
Level Test design	Test Designs - Section 3.12.5.2 (3)
Level Test case	Test Cases - Section 3.12.5.2 (4)
Level Test procedure	Test Procedures - Section 3.12.5.2 (5)
Level Test reports	Test Reports - Section 3.12.5.2 (6)
Level Test log	
Anomaly reports	
Master Test report	Master Test Report - Section 3.12.5.2 (7)

(1) Master Test Plan (MTP):

The MTP summarizes overall test plans for a project. The MTP shall contain the following items as a minimum:

a. Scope
b. References
c. System overview and key features
d. Organization
e. Master test schedule
f. Resources
g. Responsibilities
h. Tools, techniques, methods, and metrics
i. Life cycle processes
j. Test documentation requirements
k. Test administration requirements

(2) Test Specifications:

Test specification documents shall contain the following information, as a minimum, as described in IEEE Std. 829-2008 and this information shall be defined using the requirements of IEEE Std. 1008-1987 (Reference 14):

a. Test specification identifier
b. Introduction
c. Test items
d. Features to be tested
e. Features not to be tested
f. Approach
g. Item pass/fail criteria
h. Suspension criteria and resumption requirements
i. Test deliverables
j. Testing tasks
k. Environmental needs, including required tools and equipment
l. Responsibilities
m. Staffing and training needs

n. Schedule
o. Risks and contingencies
p. Approval

(3) Test Designs:

As described in IEEE Std. 829-2008, test design documents shall contain, at a minimum, the following information:

a. Test design identifier
b. Features to be tested
c. Approach refinements
d. Test identification
e. Feature pass/fail criteria

(4) Test Cases:

As described in IEEE Std. 829-2008, test case documents shall contain, at a minimum, the following information:

a. Test case identifier
b. Related Test Specifications and Test Designs
c. Input specifications
d. Output specifications
e. Environmental needs
f. Special procedural requirements
g. Interface dependencies

(5) Test Procedures:

As described in IEEE Std. 829-2008, test procedures shall contain, at a minimum, the following information:

a. Test procedure identifier
b. Purpose
c. Test Cases to be executed
d. Special requirements
e. Procedure steps

(6) Test Reports:

As described in IEEE Std. 829-2008, test reports shall contain, at a minimum, the following information:

a. Test report identifier
b. Summary
c. Variances
d. Comprehensive assessment
e. Summary of results
f. List of V&V Anomaly Reports
g. Evaluation

h. Summary of activities
i. Test Log
   (1) Test log identifier
   (2) List of tools and equipment used
   (3) Description
   (4) Activity and event entries
q. Approvals

As described in the SVVP (Section 3.10), V&V Anomaly Reports shall be prepared separately from Test Reports.

(7) Master Test Reports:

Master test reports shall summarize all test results and evaluations through the application software lifecycle. The MTR shall contain the following items as a minimum:

a. Scope
b. References
c. Overview of all aggregate test results
d. Rationale for decisions
e. Conclusions and recommendations

3.12.6 Record Keeping

All test documents described in the SVVP and this STP shall be prepared and retained as QA records under MELCO's 10 CFR 50 Appendix B QAP.

3.12.7 Methods/Tools

V&V Testing shall be executed in accordance with the procedures described in Section 3.12.5.

The MELTAC engineering tool shall be used for the test activities described in this STP and the SVVP and be required to confirm that it is suitable for use through a test tool validation program in accordance with the clause 5.3.2 of IEEE Std. 7-4.3.2-2003.

The MELTAC engineering tool is used in the integration test, the system test, and the acceptance test, in order to establish input conditions (Target Application software) and to monitor the results of the tests.

All test cases are executed manually. There is no automatic test function in the MELTAC engineering tool.

3.12.8 Standards

This STP shall be executed in conjunction with the SVVP, through use of implementing procedures, in accordance with the following standards:

(1) IEEE Std. 829-2008 (Reference 13), as endorsed by RG 1.170 (Reference 17)

(2) IEEE Std. 1008-1987 (Reference 14), as endorsed by RG 1.171 (Reference 18)

(3) IEEE Std. 1012-2004 (Reference 9), as endorsed by RG 1.168 (Reference 16)

(4) IEEE Std. 7-4.3.2-2003 (Reference 3), as endorsed by RG 1.152 (Reference 15)

(5) IEEE Std. 1074-2006 (Reference 4), as endorsed by RG 1.173 (Reference 20)

(6) Section C of RG 1.152 Rev. 3 (Reference 15)

(7) Section C of RG 1.168 Rev. 2 (Reference 16)

(8) Section C of RG 1.170 Rev. 1 (Reference 17)

(9) Section C of RG 1.171 Rev. 1 (Reference 18)

(10) Section C of RG 1.173 Rev. 1 (Reference 20)

4.0 REFERENCES

In this section, specific references referred in this SPM are provided.

Other general applicable codes and regulatory guidance are described in JEXU-1041-1008 “Safety System Digital Platform - MELTAC - Topical Report”.

1. NUREG-0800, BTP 7-14 Revision 6, “Guidance on Software Reviews for Digital Computer-Based Instrumentation and Control System”, August 2016.
   Note) Although “Safety System Digital Platform - MELTAC - Topical Report” (JEXU-1041-1008) refers to NUREG-0800, BTP 7-14 Revision 5, there is no impact on “Safety System Digital Platform - MELTAC - Topical Report” (JEXU-1041-1008).

2. IEEE Std. 603-1991, “IEEE Standard Criteria for Safety System for Nuclear Power Generating Stations”.

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

4. IEEE Std. 1074-2006, “IEEE Standard for Developing Software Life Cycle Processes”.

5. IEEE Std 830-1998, “IEEE Recommended Practice for Software Requirements Specifications”.

6. IEEE Std. 1028-2008, “IEEE Standard for Software Reviews and Audits”.

7. IEEE Std.730-1998,”IEEE Standard for Software Quality Assurance Plans”.

8. IEEE Std. 1228-1994, “IEEE Standard for Software Safety Plan”.

9. IEEE Std. 1012-2004, “IEEE Standard for Software Verification and Validation”.

10. Regulatory Guide 1.169 Revision 1 “Configuration Management Plans for Digital Computer Software Used in Safety System of Nuclear Power Plants”, September 1997.

11. IEEE Std. 828-2005, “IEEE Standard for Configuration Management Plans”.

12. IEEE Std. 1042-1987, “IEEE Guide for Software Configuration Management”.

13. IEEE Std. 829-2008, “IEEE Standard for Software Test Documentation”.

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

15. Regulatory Guide 1.152, “Criteria for Use of Computers in Safety Systems of Nuclear Power Plants” (Rev.3, July 2011).

16. Regulatory Guide 1.168, “Verification, Validation, Reviews, and Audits for Digital Computer Software Used in Safety Systems of Nuclear Power Plants” (Rev.2,

February 2004).

17. Regulatory Guide 1.170, “Software Test Documentation for Digital Computer Software Used in Safety Systems of Nuclear Power Plants”, (Rev.1, September 1997).

18. Regulatory Guide 1.171, “Software Unit Testing for Digital Computer Software Used in Safety Systems of Nuclear Power Plants”, (Rev.1, September 1997).

19. Regulatory Guide 1.172, “Software Requirements Specifications for Digital Computer Software Used in Safety Systems of Nuclear Power Plants”, (Rev.1, September 1997).

20. Regulatory Guide 1.173, “Developing Software Life Cycle Processes for Digital Computer Software Used in Safety Systems of Nuclear Power Plants”, (Rev.1, September 1997).

21. Regulatory Guide 1.169, “Configuration Management Plans for Digital Computer Software Used in Safety Systems of Nuclear Power Plants”, (Rev.1 September 1997).

22. IEEE Std. 610.12-1990, “IEEE Standard Glossary of Software Engineering Terminology”.

23. NUREG/CR-6101, “Software Reliability and Safety in Nuclear Reactor Protection Systems”, 1993.

24. ANSI/ASME NQA-1-1994, “Quality Assurance Program Requirements for Nuclear Facilities”.

25. Regulatory Guide 1.153, “Criteria for Safety Systems”, (Rev.1, June 1996).

26. “Safety System Digital Platform - MELTAC - Topical Report”, Rev.2.

27. “MELTAC Platform Software Program Manual”.

Appendix A Definitions

Acceptance Testing [IEEE Std. 610.12-1990]:
Formal testing conducted in an operational environment 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. Formal testing conducted to enable a user, customer, or other authorized entity to determine whether to accept a system or component.

Accident:
An unplanned event or series of events that result in death, injury, illness, environmental damage to or loss of equipment or property.

Anomaly [IEEE Std. 610.12-1990]:
Any condition that deviates from the expected condition based on requirements, specification, design, documents, user documents standards, or from someone's perceptions or experiences.
Anomalies may be found as a result of the review, test, analysis, compilation, or use of software products or applicable documentation.

Application Software:
The application software provides the plant specific functionality of the Mitsubishi Electric Total Advanced Controller (MELTAC) I&C system. It is documented and generated by the MELTAC engineering tool. The platform system software (i.e., basic software) uses this configuration data to carry out the application specific functionality of the I&C system.

Application Software Execution Data:
The installable application software data into the CPU module of the MELTAC platform, or the application software data installed in the the CPU module of the MELTAC platform.
This data is generated from the Application Source Codes by using the MELTAC engineering tool.

Application Source Code Listings:
The list of the application software outputted from the MELTAC engineering tool. Specifically, it is Graphic Block Diagram (GBD) currently written by Problem Oriented Language (POL).

Baseline [IEEE Std. 610.12-1990]:
A specification or product that has been formally reviewed and agreed upon, that thereafter serves as the basis for further development, and that can be changed only through formal change control procedures. Formal review and agreement means that design team management has reviewed and approved a baseline. Baselines are subject to change control.

Code:
Computer instructions and data definitions expressed in a programming language or in a form that is outputted by an assembler, compiler, or another translator.

Component [IEEE Std. 610.12-1990]:
One of the parts that make up a system. A component may be hardware or software and may be subdivided into other components.

Component Testing [IEEE Std. 610.12-1990]:
Testing of individual hardware or software components or groups of related components

Configuration [IEEE Std. 610.12-1990]:
The arrangement of a computer system or component as defined by the number, nature, and interconnections of its constituent parts. In configuration management, the functional and physical characteristics of hardware or software as set forth in technical documentation or achieved in a product.

Configuration Audit [IEEE Std. 610.12-1990]:
Functional Configuration Audit (FCA). An audit conducted to verify that the development of a configuration item has been completed satisfactorily, that the item has achieved the performance and functional characteristics specified in the functional or allocated configuration identification, and that its operational and support documents are complete and satisfactory.

Configuration Control [IEEE Std. 610.12-1990]:
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.

Configuration Control Board (CCB) [IEEE Std. 610.12-1990]:
A group of people responsible for evaluating and approving or disapproving proposed changes to configuration items, and for ensuring implementation of approved changes.

Configuration Identification [IEEE Std. 610.12-1990]:
(1) An element of configuration management, consisting of selecting the configuration items for a system and recording their functional and physical characteristics in technical documentation.
(2) The current approved technical documentation for a configuration item as set forth in specifications, drawings, associated lists, and documents referenced therein.

Configuration Item (CI) [IEEE Std. 610.12-1990]:
An aggregation of hardware, software, or both, that is designated for configuration management and treated as a single entity in the configuration management process.

Configuration Management (CM) [IEEE Std. 610.12-1990]:
A discipline applying technical and administrative direction and surveillance to:
-Identify and document the functional and physical characteristics of a configuration item
-Control changes to those characteristics
-Record and report change processing and implementation status
-Verify compliance with specified requirements

Configuration Status Accounting [IEEE Std. 610.12-1990]:
An element of configuration management, consisting of the recording and reporting of information needed to manage a configuration effectively. This information includes a listing of the approved configuration identification, the status of proposed changes to the configuration, and the implementation status of approved changes.

Control Point [IEEE Std. 828-1990]:
It is a point at which controls are be applied to manage configuration. A project agreed on point in time or times when specified agreements or controls are applied to the software configuration items being developed, e.g., an approved baseline or release of a specified document/code.

Criticality:
A subjective description of the intended use and application of the system. Software criticality properties may include safety, security, complexity, reliability, performance, or other characteristics.

Criticality Analysis:
A structured evaluation of the software characteristics (e.g., safety, security, complexity, performance) for severity of impact of system failure, system degradation, or failure to meet software requirements or system objectives.

Customer:
The individual or organization that specifies and accepts the project deliverables. The customer may be internal or external to the parent organization of the project, and may or may not be the end user of the software product. A financial transaction between customer and developer is not necessarily implied.

Design Review [IEEE Std. 610.12-1990]:
A process or meeting during which a system, hardware, or software design is presented to project personnel, managers, users, customers, or other interested parties for comment or approval. Types include critical design review, preliminary.

Flash Read Only Memory (F-ROM):
One of the nonvolatile semiconductor memories where data does not disappear after powering off.

Functional Testing [IEEE Std. 610.12-1990]:
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. Testing conducted to evaluate the compliance of a system or component with specified functional requirements.

Hazard:
A source of potential harm or a situation with a potential for harm in terms of human injury, damage to health, property, or the environment, or some combination of these.

Installation Configuration Listings:
Listing that includes all of the functional characteristics of the software application. This is equivalent to the “Installation Configuration Table” referenced in NUREG 0800, Branch Technical Position, BTP 7-14.

Installed System:
The system which the Installation Phase completed used for an operation system.

Integration Testing [IEEE Std. 610.12-1990]:
Testing in which software components, hardware components, or both are combined and tested to demonstrate correct interaction between them.

Interface Control [IEEE Std. 610.12-1990]:
(1) In configuration management, the process of:
(a) Identifying all functional and physical characteristics relevant to the interfacing of two or

more configuration items provided by one or more organizations, and (b) ensuring that proposed changes to these characteristics are evaluated and approved prior to implementation.
(2) In configuration management, the administrative and technical procedures and documentation necessary to identify functional and physical characteristics between and within configuration items provided by different developers, and to resolve problems concerning the specified interfaces

Interface Requirements Specification (IRS):
Documentation that specifies requirements for interfaces between systems or components. These requirements include constraints on formats and timing.

Maintenance [IEEE Std. 610.12-1990]:
The process of modifying a software system or component after delivery to correct faults, improve performance (or other attributes), or adapt to a changed environment. The process of retaining or restoring a hardware system or component in a state in which it can perform its required functions.

Pass/fail Criteria:
Decision rules used to determine whether a software configuration item or a software feature passes or fails a test.

Problem Oriented Language (POL):
Application software is described in a graphically symbolized manner, using the problem oriented language (POL), so that functions can be easily understood.

Release:
The formal notification and distribution of an approved version.

Required Inputs:
The set of items necessary to perform the minimum V&V tasks mandated within any life cycle activity.

Required Output:
The set of items produced as a result of performing the minimum V&V tasks mandated within any life cycle activity.

Requirement Traceability Matrix (RTM) [BTP 7-14]
An RTM shows every requirement, broken down in to sub-requirements as necessary, and what portion of the software requirement, software design description, actual code, and test requirement addresses that system requirement.

Software Design Description (SDD) [IEEE Std. 610.12-1990]:
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 model of the system.

Software Failure:
A distinguishing characteristic of a software item (for example, performance, portability, or functionality).

Software Hazard [Based on IEEE Std. 610.12-1990]:
A software error that could result in the failure of functions or unintended operation including abnormal events, conditions and malicious modifications.

Software Library [IEEE Std. 610.12-1990]:
A controlled collection of software and related documentation designed to aid in software development, use, or maintenance. Types include master library, production library, software development library, software repository, and system library.

Software Life Cycle [IEEE Std. 610.12-1990]:
The period of time that begins when a software product is conceived and ends when the software is no longer available for use.

System:
A combination of more than one process, hardware, software, equipment, and humans designed to provide the ability to meet specific requirements.

Traceability Matrix [IEEE Std. 610.12-1990]:
A matrix that records the relationship of verifiable characteristics between two or more products of the development process.

Unit [IEEE Std. 610.12-1990]:
A separately testable element specified in the design of a computer software component.
A logically separable part of a computer program. A software component that is not subdivided into other components.

Unit Testing [IEEE Std. 610.12-1990]:
Testing of individual hardware or software units or groups of related units.

Validation [IEEE Std. 610.12-1990]:
The process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements.

Verification [IEEE Std. 610.12-1990]:
The process of evaluating a system or component to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase. Contrast with: validation.

Verification and Validation (V&V) [IEEE Std. 610.12-1990]:
The process of determining 1) whether the requirements for a system or component are complete and correct, 2) whether the products of each development phase fulfill the requirements or conditions imposed by the previous phase, and 3) whether the final system or component complies with specified requirements.