WCAP-18780-NP, Revision 0
"Advanced Logic System v2 Development Process Topical Report"
(Non-Proprietary)

Westinghouse Electric Company
1000 Westinghouse Drive
Cranberry Township, PA 16066

Westinghouse Non-Proprietary Class 3

WCAP-18780-NP
Revision 0

December 2022

Advanced Logic System® v2 Development Process Topical Report

Westinghouse

WCAP-18780-NP
Revision 0

Advanced Logic System® v2 Development Process Topical Report

Matthew A Shakun*

Licensing Engineering

December 2022

Reviewer: Warren R. Odess-Gillett*, Fellow Engineer
Licensing Engineering

Verifier: Thomas Tweedle*, Principal Engineer
Advanced Reactors and Engineering

Approved: Anthony J. Schoedel*, Manager
Advanced Reactors Licensing

James F. Mermigos*, Manager
Instrumentation & Monitoring Systems

*Electronically approved records are authenticated in the electronic document management system.

Westinghouse Electric Company LLC
1000 Westinghouse Drive
Cranberry Township, PA 16066, USA

TABLE OF CONTENTS

TABLE OF CONTENTS (cont.)

TABLE OF CONTENTS (cont.)

TABLE OF CONTENTS (cont.)

TABLE OF CONTENTS (cont.)

TABLE OF CONTENTS (cont.)

TABLE OF CONTENTS (cont.)

LIST OF TABLES

Table **Title** **Page**

LIST OF FIGURES

Table **Title** **Page**

ACRONYMS AND TRADEMARKS

The following lists acronyms used in this document that are not defined in 6002-00040, “ALS Terms and Abbreviations” (Reference 1) or WNA-PS-00016-GEN, “Standard Acronyms and Definitions” (Reference 2), or they are included below to ensure unambiguous understanding of their use within this document.

Acronym	**Definition**
ABTS	ALS Board Test System
ALS	Advanced Logic Systems
ALS v2	Advanced Logic Systems Version 2
ASME	American Society of Mechanical Engineers
ASU	ALS Service Unit
ATCT	ALS Test and Configuration Tool
BOM	Bill of Material
BTP	Branch Technical Position
CFR	Code of Federal Regulations
CGDP	Commercial Grade Dedication Program
CDI	Commercial Dedication Instructions
CHT	Cabinet Hardware Test
CLB	Core Logic Board
CM	Configuration Management
CMP	Configuration Management Plan
CMRR	Configuration Management Release Report
CPU	Central Processing Unit
CRC	Cyclic Redundancy Checks
CSA	Configuration Status Accounting
DT	Design Team
EDMS	Electronic Document Management System
ELM	Engineering Line Manager
EMC	Electro-Magnetic Compatibility
EQ	Equipment Qualification
ESFAS	Engineered Safety Features Actuation System
FA	Functional Audit
FAT	Factory Acceptance Test
FLD	Functional Logic Diagrams
FMEA	Failure Modes and Effects Analysis
FMEDA	Failure Modes and Effects Diagnostic Analysis
FOAK	First of a Kind
FPGA	Field Programmable Gate Array
FSM	Finite State Machine
HMI	Human Machine Interface
I&C	Instrumentation & Control

Acronym	**Definition**
I/O	Input/Output
IEEE	Institute of Electrical and Electronics Engineers
IV&V	Independent Verification & Validation
NPP	Nuclear Power Plant
NVM	Non-Volatile Memory
PAMS	Post-Accident Monitoring System
PCB	Printed Circuit Board
PM	Project Manager
PMP	Project Management Plan
QA	Quality Assurance
QAP	Quality Assurance Plan
QDS	Qualified Display System
QMS	Quality Management System
RFP	Request for Proposal
RLET	Reusable Logic Element
RLETD	Reusable Logic Element Document
RM	Requirements Management
RPS	Reactor Protection System
RTA	Requirements Traceability Analysis
RTL	Register Transfer Language
RTM	Requirements Traceability Matrix
SAT	Site Acceptance Testing
SCM	Software Configuration Management
SDE	Secure Development Environment
SDP	Software Development Plan
SecP	Security Plan
SHA	Software Hazards Analysis
SHT	Subassembly Hardware Test
SMP	Software Maintenance Plan
SOP	Software Operations Plan
SSP	Software Safety Plan
SVVP	Software Verification and Validation Plan
TL	Technical Lead
TP	Test Plan
V&V	Verification & Validation

All other product and corporate names used in this document may be trademarks or registered trademarks of other companies, and are used only for explanation and to the owners' benefit, without intent to infringe.

GLOSSARY OF TERMS

The terms used in this document that are not defined in 6002-00040, "ALS Terms and Abbreviations" (Reference 1) or WNA-PS-00016-GEN, "Standard Acronyms and Definitions" (Reference 2), or they are included below to ensure unambiguous understanding of their use within this document.

Term **Definition**

ALS Applications

The use of the ALS platform components in specific instrumentation and control systems, usually specified by a customer. Each application of the ALS platform is unique according to specific design requirements that make use of the ALS platform. Generally, applications involve custom field programmable gate array (FPGA) logic on the ALS-152 core logic board (CLB) and non-volatile memory (NVM) images installed and running on ALS platform boards.

ALS Platform

The standard set of hardware and software components that encompass the ALS product. ALS platform components are verified and validated for use in applications such as instrumentation and control systems. These components mainly consist of the ALS boards, backplane, chassis, FPGA, and non-volatile memory images.

Register Transfer Language

A standard language used to describe the behavior and structure of the logic of an FPGA design. The term Register Transfer Language is used interchangeably with Hardware Description Language. The most popular register transfer languages include Verilog and VHDL.

Register Transfer Level

A description of an FPGA design that is focused on the flow of signals between registers. The register transfer level design is translated into gate level description with the synthesis process. A register transfer language is used to describe the register transfer level design.

Software

The programs used to direct operations of a programmable digital device. Examples include computer programs and logic for programmable hardware devices, and data pertaining to its operation. (Definition from IEEE Std. 7-4.3.2-2016 [Reference 29]).

Xxx or XXX

The notation "xxx" is used to denote that a document exists for each of the boards in the platform and xxx will be substituted by the three-digit identifier for the document associated with a given board.

REFERENCES

Following is a list of references used throughout this document.

1. 6002-00040, Rev. 4, "ALS Terms and Abbreviations," Westinghouse Electric Company LLC.

2. WNA-PS-00016-GEN, Rev. 8, "Standard Acronyms and Definitions," Westinghouse Electric Company LLC.

3. WCAP-18762-P, Rev. 0, "Advanced Logic System v2 Platform Topical Report," Westinghouse Electric Company LLC.

4. "Westinghouse Quality Management System," Westinghouse Electric Company LLC.

5. WNA-PV-00129-GEN, Rev. 0, "Advanced Logic System v2 Software Verification and Validation Plan," Westinghouse Electric Company LLC.

6. 10 (Code of Federal Regulations) CFR Part 50, Appendix B, "Quality Assurance Criteria for Nuclear Power Plants and Fuel Reprocessing Plants," U.S. Nuclear Regulatory Commission.

7. IEEE Standard 1012-2004, "IEEE Standard for Software Verification and Validation," Institute of Electrical and Electronics Engineers, Inc., 2004.

8. IEEE Standard 603-1991, "IEEE Standard Criteria for Safety Systems for Nuclear Power Generating Stations," Institute of Electrical and Electronics Engineers, Inc., 1991.

9. IEEE Standard 7-4.3.2-2003, "IEEE Standard Criteria for Digital Computers in Safety Systems of Nuclear Power Generating Stations," Institute of Electrical and Electronics Engineers, Inc., 2003.

10. IEEE Standard 1074-2006, "IEEE Standard for Developing a Software Project Life Cycle Process," Institute of Electrical and Electronics Engineers, Inc., 2006.

11. NUREG-0800, Standard Review Plan (SRP) for the Review of Safety Analysis Reports for Nuclear Power Plants, Chapter 7, "USNRC Standard Review Plan for the Review of Safety Analysis Reports for Nuclear Power Plants," U.S. Nuclear Regulatory Commission, Revision 7, August 2016.

12. 6003-00010, Rev. 1, "ALS v2 Platform Requirements Specification," Westinghouse Electric Company LLC.

13. IEEE Standard 830-1998, "IEEE Recommended Practice for Software Requirements Specifications," Institute of Electrical and Electronics Engineers, Inc., 1998.

REFERENCES (cont.)

14. IEEE Standard 352-1987, “IEEE Guide for General Principals of Reliability Analysis of Nuclear Power Generating Station Safety Systems,” Institute of Electrical and Electronics Engineers Inc., 1987.

15. Regulatory Guide 1.170, Rev. 1, “Test Documentation for Digital Computer Software Used in Safety Systems of Nuclear Power Plants,” U.S. Nuclear Regulatory Commission.

16. IEEE Standard 730-1998, “Standard for Software Quality Assurance Plans,” Institute of Electrical and Electronics Engineers, Inc., 1998.

17. ISO 9001:2015, “Quality Management Systems – Requirements,” International Organization for Standardization.

18. ASME NQA-1- 2008/2009a, “Quality Assurance Program Requirements for Nuclear Facilities/with Addenda,” American Academy of Mechanical Engineers.

19. 10 CFR Part 21, “Reporting of Defects and Noncompliance,” U.S. Nuclear Regulatory Commission.

20. IEEE Standard 1028-2008, “IEEE Standard for Software Reviews and Audits,” Institute of Electrical and Electronics Engineers, Inc., 2008.

21. 6002-00002-P, Rev. 11, “ALS Configuration Management Plan,” Westinghouse Electric Company LLC.

22. 6002-00006, Rev. 3, “ALS Security Plan,” Westinghouse Electric Company LLC.

23. Branch Technical Position (BTP) Rev 6, 7-14, “Guidance on Software Reviews for Digital Computer-Based Instrumentation and Control Systems,” U.S. Nuclear Regulatory Commission.

24. NUREG/CR-6101, “Software Reliability and Safety in Nuclear Reactor Protection Systems,” Lawrence Livermore National Laboratory, 1993.

25. WCAP-17266-P, Rev. 1, “Common Q Generic Change Process,” Westinghouse Electric Company LLC

26. NUREG/CR-6430, “Software Safety Hazard Analysis,” 1995.

27. IEEE Standard 1228-1994 (Reaffirmed 2010), “IEEE Standard for Software Safety Plans,” Institute of Electrical and Electronics Engineers, Inc., 1994.

28. IEEE Standard 829-1998, “IEEE Standard for Software Test Documentation,” Institute of Electrical and Electronics Engineers, Inc., 1998.

REFERENCES (cont.)

29. IEEE Standard 7-4.3.2-2016, “IEEE Standard Criteria for Programmable Digital Devices in Safety Systems of Nuclear Power Generating Stations,” Institute of Electrical and Electronics Engineers, Inc., 2016.

30. 6002-00030, Rev. 13, “Advanced Logic Sytem Design Tools,” Westinghouse Electric Company LLC.

31. 6003-00011, Rev.1, “ALS v2 Platform Design Specification,” Westinghouse Electric Company LLC.

(Last Page of Front Matter)

1 PURPOSE

This purpose of this document is to define the overall life cycle development process and to identify the detailed supporting processes by which safety-related instrumentation and control (I&C) activities will be executed for Advanced Logic System (ALS) version 2 (v2) platform development and maintenance, and applications of the ALS v2 platform. The ALS v2 platform is described in WCAP-18762-P, “Advanced Logic System v2 Platform Topical Report” (Reference 3).

1.1 SCOPE

This process applies to ALS v2 based platform development and maintenance, and applications of the ALS v2 platform.

The ALS v2 platform is based on field programmable gate array (FPGA) technology and this process is targeted to the FPGA-related activities. The use of the ALS v2 platform for a specific application can require supporting components or subsystems that are not FPGA-based, such as an ALS service unit (ASU), which uses traditional central processing unit (CPU)-based software. The development, integration, and testing of these subsystems and components shall be addressed in the project planning documentation, generally following the procedures and plans identified in the Westinghouse Quality Management System (QMS) (Reference 4). These subsystems and components are outside the scope of this process and are not discussed further herein.

This process is applicable for the ALS v2 platform and projects. The terms “ALS” and “ALS projects” within this document refers to both ALS v2 application projects as well as ALS v2 platform development and maintenance.

1.2 SOFTWARE CLASSIFICATION AND CATEGORIZATION

This process shall apply to all development activities using the ALS. ALS equipment and FPGA classifications shall be determined per the following classes. Software tasks and responsibilities for each classification are defined in WNA-PV-00129-GEN, “Advanced Logic System v2 Software Verification and Validation Plan” (SVVP) (Reference 5).

- Protection (safety critical – critical performance of the system) – Safety critical software whose functionality is necessary to directly perform reactor protection system (RPS) control actions, engineered safety features actuation system (ESFAS) control actions, and safe shutdown control actions (meets 10 CFR Part 50 Appendix B [Reference 6] requirements).

- Important-to-safety – Safety-related software whose functionality is necessary to directly perform alternate protection system control actions, software that is relied on to monitor or test protection functions, or software that monitors plant critical safety functions (meets 10 CFR Part 50 Appendix B [Reference 6] requirements).

- Important-to-availability – Non-safety-related software whose functionality is relied on to directly perform alternate protection system control actions or software to maintain operation of plant systems and equipment that are critical to maintaining an operating plant.

- General purpose – Non-safety-related software whose functionality performs some purpose other than those described in the previous classifications. This software includes tools that are used to develop software in the other classifications but is not installed in an online plant system.

Procedures identify control requirements for equipment and software based on their safety-related function. All equipment and software shall be classified. The software integrity levels (classification) will determine the intensity and rigor of the verification and validation (V&V) activities per the SVVP (Reference 5).

Table 1.2-1 provides a mapping of the four software integrity levels defined in the Institute of Electrical and Electronics Engineers (IEEE) Standard 1012 (Reference 7) to the classifications described above.

Table 1.2-1: Classification Mapping

Software Classification	**IEEE Standard 1012 Integrity Level**
Protection	4
Important-to-safety	4*
Important-to-availability	2
General purpose	1

*Note: A software hazard analysis will not be performed for important-to-safety software. See Section 8.

The assignment of ALS software to the classifications described above is shown in Table 1.2-2. Specific functions in a single system may be assigned to different classes. Each system function must have an assigned class.

Table 1.2-2: Assignment of ALS Software to Classes

System	Sub-System Scope	Class
Plant Protection Reactor Protection Safety-Related Protection System Engineered Safety Features Actuation	Safety Critical Functions	Protection
	Intra-Divisional Communications	Protection
	Qualified Display System (QDS)	Important-to-Safety
	ASU (Maintenance)	Important-to-Availability
	Development Tools	General Purpose
Post-Accident Monitoring System (PAMS)	PAMS Functions	Important-to-Safety
	QDS	Important-to-Safety
	ASU (Maintenance)	Important-to-Availability
	Development Tools	General Purpose
Diesel Load Sequencer	Load Sequencing Functions	Protection
	QDS	Important-to-Safety
	ASU (Maintenance)	Important-to-Availability
	Development Tools	General Purpose
Diesel Control System	Diesel Control Functions	Important-to-Safety
	QDS	Important-to-Safety
	ASU (Maintenance)	Important-to-Availability
	Development Tools	General Purpose
Reactor Controls	Reactor Control Functions (Reactor Power Control, Load Following)	Important-to-Availability
	Interface and Test	Important-to-Availability
	QDS (Soft Control)	Important-to-Availability
	ASU (Maintenance)	Important-to-Availability
	Development Tools	General Purpose
Diverse Actuation System	Diverse Actuation Functions	Important-to-Availability
	Interface and Test	Important-to-Availability
	QDS (Soft Control)	Important-to-Availability

Table 1.2-2: Assignment of ALS Software to Classes

System	Sub-System Scope	Class
	ASU (Maintenance)	Important-to-availability
	Development Tools	General Purpose
Plant Controls	Secondary Side Control Functions (Secondary Side Control, Secondary Side Monitoring)	Important-to-Availability
	Interface and Test	Important-to-Availability
	QDS (Soft Control)	Important-to-Availability
	ASU (Maintenance)	Important-to-Availability
	Development Tools	General Purpose

(Last Page of Section 1)

2 ORGANIZATION AND RESPONSIBILITIES

The following is a high-level definition of the Westinghouse organization and responsibilities for a project. Detailed responsibility assignments are initially established in the concept/planning phase and updated as the project progresses as necessary. Assignment of responsibilities for each exit criterion shall be consistent with the applicable Westinghouse policies, procedures, and planning documents.

2.1 ENGINEERING LINE MANAGER

The engineering line manager (ELM) is responsible for:

1. Approving the project management plan (PMP)
2. Determining the need for design reviews
3. Reviewing the completion of exit criteria before declaring a phase complete

2.2 PROJECT MANAGER

The project managers (PMs) are responsible for:

1. The overall execution of the project
2. Developing and issuing the project management plan
3. Conducting management reviews as required
4. Ensuring that design reviews are conducted as scheduled
5. Reviewing the completion of exit criteria before declaring a phase complete

The PM is assigned to a particular ALS project and is responsible for overseeing the development, scheduling, and the financial and quality execution of the assigned project. The ALS platform lead may be responsible for these functions for internal ALS platform development activities. The ALS platform lead is responsible for the platform development meeting the continuing needs of the product family. PMs and platform leads may delegate the performance of necessary tasks to other persons but remain responsible for their execution.

2.3 TECHNICAL LEAD (TL)

Note: A technical lead (TL) is typically assigned for each major activity area, such as overall project design lead, functional requirements, software or hardware design, requirements management (RM), independent verification and validation (IV&V), testing, etc.

The TLs are responsible for:

1. Providing technical leadership on assigned projects and tasks
2. Ensuring that project engineering activities and design documentation comply with all regulatory and contract requirements

2.4 CONFIGURATION MANAGER

The configuration manager is responsible for:

Note: The PM and configuration manager may be the same person.

1. Generating the configuration management plan (CMP)
2. Oversight of the execution of the CMP
3. Generating configuration status accounting baselines and configuration management reports at the appropriate intervals

2.5 QUALITY ASSURANCE PERSONNEL

The quality assurance (QA) personnel are responsible for:

1. Verifying that the documented development processes and controls used by the project are adhered to and effective
2. Evaluating the development products (e.g., design data and documentation) for completeness and correctness

The QA personnel interact with the design team (DT) to function effectively and meet their goals. QA personnel must:

1. Participate in project meetings, as appropriate
2. Participate in the design review process, as required
3. Perform surveillances and audits of the various development activities
4. Report results of activities periodically

2.6 EQUIPMENT QUALIFICATION PERSONNEL

The equipment qualification (EQ) personnel responsible for:

1. Ensuring that safety-related equipment complies with all applicable seismic, environmental, and electro-magnetic compatibility (EMC) regulations. For applications, this includes customer requirements for a specific site.

2.7 INDEPENDENT VERIFICATION AND VALIDATION TEAM

The independent verification and validation (IV&V) team is responsible for:

1. Planning and executing product software verification and validation for safety-related products that are classified as protection and important-to-safety, as defined in Section 1.2. Included in IV&V activities is product testing, which is executed by an independent test team. The test team is part of the IV&V effort.

See Figure 2.7-1 for the minimum level of independence required for the design and IV&V teams.

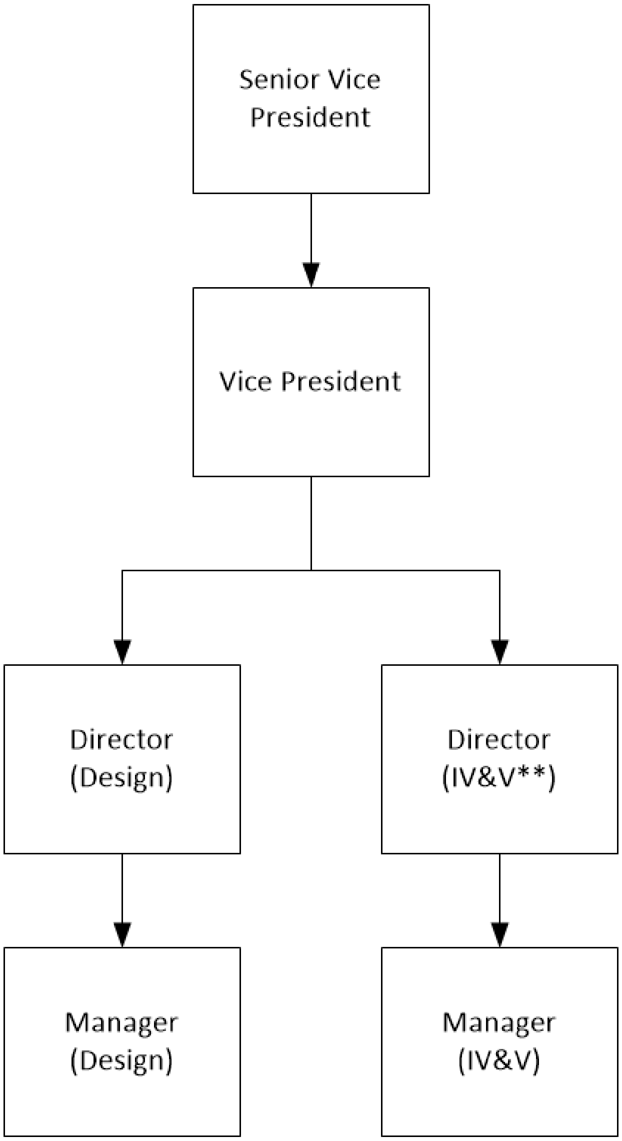

Figure 2.7-1: IV&V Team Organization

**Note: System-level validation testing can be performed by another group that meets the same level of independence as the IV&V group depicted in this organization chart.

(Last Page of Section 2)

3 LIFE CYLE PHASE ACTIVITIES

3.1 INTRODUCTION

This process defines the life cycle process for ALS-based nuclear safety equipment development. The exit criteria for each phase of the life cycle are provided. The process described in this document applies to both platform and application development. Section 4 provides a description of the platform developments activities while Section 5 provides a description of the application development activities.

IEEE Standard 603 (Reference 8) and IEEE Standard 7-4.3.2-2003 (Reference 9) support systems development. The former addresses computer and non-computer hardware elements while the latter addresses system-level issues for software.

Each phase of the life cycle is defined in a dedicated section of this document. The sections start with a description of the basic purpose of the phase, followed by subsections containing a description of phase products and tasks, and a list of the design completion criteria relative to those products and tasks.

Outputs produced by the IV&V team are outlined in WNA-PV-00129-GEN, "Advanced Logic System v2 Software Verification and Validation Plan" (Reference 5). When specific regulatory criteria and guidance exists for a phase output or task, the identification of such is included.

3.2 LIFE CYCLE DEFINITION

3.2.1 Phases

Software-projects shall follow a structured life cycle per IEEE Standard 1074 (Reference 10). Although FPGA-based products do not use software in a traditional sense, a similar structured life cycle shall be applied. IEEE Standard 1074 does not require one specific life cycle model and it recognizes that there are a variety of models that can be chosen for software development.

The life cycle model phases applied to ALS projects are as follows:

- Concept/Planning Phase
- Requirements Phase
- Design Phase
- Implementation Phase
- Integration (Test) Phase
- Installation Phase
- Operation & Maintenance Phase
- Retirement Phase

This process expands on the software-specific activities and outputs of the life cycle phases by inclusion of the overall activities and outputs for the development of integrated hardware/software platform or system. The phases identified above are similar to the life cycle depicted in BTP 7-14 of NUREG 0800 (Reference 11) for software development.

The activities of the life cycle phases may overlap, (i.e., work may be performed concurrently in more than one life cycle phase); however, a given phase can be declared "closed" only if its predecessor phases are closed. While not desirable, iterations of life cycle phases can occur due to changes in one or more products (as defined by the exit criteria) of a phase that was previously declared as being closed. A previously closed phase can be reopened with an analysis of the overall effects on that phase and all subsequent phases. Thus, reopening of any given phase will necessitate reopening of any closed subsequent phases since the products and activities of the subsequent phases will likely be affected. The design team along with the PM and ELM will assess which activities or products of the reopened life cycle phase and subsequent phases must be reperformed or revised. The IV&V team will determine their scope for regression activities. The configuration management plan (CMP) and Westinghouse QMS (Reference 4) procedures ensure appropriate control of changes. A phase can be reclosed if its predecessor phases are closed, and its completion is documented via a configuration management release report (CMRR). Additional information and requirements for phase closure are provided in the following section.

In addition to life cycle phases, IEEE Standard 1074 (Reference 10) defines the concept of the integral activities group. These are activities that are necessary to ensure the successful completion of a project but are considered as support activities rather than those that are directly oriented to the development effort. These integral activities occur throughout the development lifecycle; therefore, they are not defined as a specific phase within the lifecycle sequence. Integral activities consist of audits, reviews, tests, traceability matrices, V&V analyses, configuration management, etc., and are defined as activities or exit criteria for the applicable lifecycle phases.

3.2.2 Phase Closure

A listing of exit criteria for each phase of the life cycle is included at the end of the document section. The exit criteria includes the overall major design team deliverables and tasks that must be completed and documented before the phase is declared "closed." While the project life cycle phase activities may overlap or be performed iteratively, a given phase can be closed only if the predecessor phases have a closed status. Below are additional requirements and information related to phase closure:

Open items can exist for one or more exit criteria in a phase while not limiting the ability to perform tasks in subsequent phases, provided the effects of the open item on subsequent phases are evaluated and deemed low risk by the PM or ELM.

Phase exit criteria that is produced by the IV&V team (refer to WNA-PV-00129-GEN, Reference 5) is not required for the DT phase closure. DT phases are closed via issuance of the configuration status accounting (CSA) document, which defines the configuration management release.

3.2.3 Life Cycle Sequencing

Baselines capture unique sets of configuration items that are to be used as input or the basis for subsequent life cycle phases. The design team develops and releases their lifecycle products (or configuration items) against a baseline document, and they document their released design products in a CMRR. The CMRR is the document that releases a product (e.g., design document) for use by downstream activity groups. For ALS projects, the CSA document is the CMRR.

Large, complex projects can benefit from baseline-controlled life cycle sequencing. The life cycle sequencing concept allows work to be performed in more than one life cycle sequence simultaneously. For example, consider that the life cycle for a given I&C project had progressed through the design phase when a baseline revision necessitates that the project's requirements phase be reopened to implement a revision to the requirements documents. Since such changes are often a small portion of the overall scope, design activities that were occurring per the prior baseline can continue as unaffected portions of scope.

As the affected Requirements Phase documents are updated per the new baseline and the phase is subsequently reclosed with an update of the exit criteria and the CMRR, the changes are allocated to the downstream design activities via the normal processes and the Design Phase can then align with the new baseline.

3.2.4 Deviations to Life Cycle Model

Any deviations (i.e., phases, activities, reviews, deliverables, exit criteria, etc.) to the life cycle model defined in this process shall be explicitly documented and justified in the project management plan.

Reopening of any closed phase and its effect on the overall life cycle and the exit criteria, shall be documented in the configuration management report.

(Last Page of Section 3)

4 PLATFORM DEVELOPMENT PLAN

4.1 INTRODUCTION

This section provides an overview of the ALS platform development activities for the life cycle phases defined in Section 3.2.1 of this document. Platform development ends when a board is produced that is ready to use in future applications. Therefore, platform development ends with the Integration Phase. Figure 4.1-1 shows an outline of the ALS platform development process, excluding IV&V.

a,c

Figure 4.1-1: ALS Platform Development Process

4.2 [ ]^{a,c}

[
]^{a,c}

4.2.1 [ ]^{a,c}

[
]^{a,c}

4.2.2 [ ]^{a,c}

[
]^{a,c}

[

]^{a,c}

4.2.3 [ ]^{a,c}

[

]^{a,c}

4.2.4 [ ]^{a,c}

[

]^{a,c}

4.2.5 [ ]^{a,c}

[ ]^{a,c}

4.2.6 [ ]^{a,c}

[

]^{a,c}

[ ]^{a,c}

4.3 [ ]^{a,c}

[ ]^{a,c}

4.3.1 [ ]^{a,c}

[ ]^{a,c}

4.3.1.1 [ ]^{a,c}

[ ]^{a,c}

[

]^{a,c}

4.3.2 [ ]^{a,c}

[

]^{a,c}

4.3.3 [ ]^{a,c}

[

]^{a,c}

4.4 [ ]^{a,c}

[

]^{a,c}

4.4.1 [ ]^{a,c}

[

]^{a,c}

4.4.2 [ ]^{a,c}

[

]^{a,c}

4.4.3 [ ]^{a,c}

[

]^{a,c}

4.4.4 [ ]^{a,c}

[

]^{a,c}

4.4.5 [ ]^{a,c}

[

]^{a,c}

[

]^{a,c}

4.5 [ ]^{a,c}

[

]^{a,c}

4.5.1 [ ]^{a,c}

[

]^{a,c}

4.5.1.1 [ ]^{a,c}

[

]^{a,c}

4.5.1.1.1 [ ]^{a,c}

[

]^{a,c}

[

]^{a,c}

4.5.1.2 [ ]^{a,c}

[

]^{a,c}

[

]^{a,c}

4.5.1.3 [ ]^{a,c}

[

]^{a,c}

4.5.1.4 [ ]^{a,c}

[ ]^{a,c}

4.5.1.5 [ ]^{a,c}

[

]^{a,c}

4.5.2 [ ]^{a,c}

[

]^{a,c}

4.5.2.1 [ ]^{a,c}

[

]^{a,c}

[

]^{a,c}

4.5.2.2 [ ]^{a,c}

[

]^{a,c}

4.5.2.3 [ ]^{a,c}

[ ]^{a,c}

4.5.2.4 [ ]^{a,c}

[ ]^{a,c}

4.5.2.5 [ ]^{a,c}

[

]^{a,c}

4.5.2.6 [ ]^{a,c}

[

]^{a,c}

4.5.2.7 [ ]^{a,c}

[

]^{a,c}

4.5.3 [ ]^{a,c}

[

]^{a,c}

[

]^{a,c}

4.5.4 [ ]^{a,c}

[

]^{a,c}

4.6 [ ]^{a,c}

[ ]^{a,c}

(Last Page of Section 4)

5 APPLICATION DEVELOPMENT PLAN

5.1 INTRODUCTION

This section provides an overview of the ALS application development activities for the life cycle phases defined in subsection 3.2.1 of this document. Application development ends when the system is installed in the plant site. Figure 5.1-1 shows an outline of the ALS application development process.

a,c

Figure 5.1-1: ALS Application Development Process

5.2 [ ]a,c

[

]a,c

5.2.1 [ ]a,c

[
]a,c

5.2.2 [ ]a,c

[
]a,c

5.2.3 [ ]a,c

[
]a,c

5.2.4 [ ]a,c

[ ]a,c

5.2.5 [ ]a,c

[
]a,c

5.2.6 [ ]a,c

[ ]a,c

5.3 [ ]a,c

[

]a,c

5.3.1 [ ]^{a,c}

5.3.1.1 [ ]^{a,c}

[

]^{a,c}

[

]^{a,c}

5.3.1.2 [ ]^{a,c}

[

]^{a,c}

5.3.2 [ ]^{a,c}

[

]^{a,c}

[

]^{a,c}

5.3.3 [ ]^{a,c}

[
]^{a,c}

5.3.4 [ ]^{a,c}

[

]^{a,c}

5.4 [ ]^{a,c}

[

]^{a,c}

5.4.1 [ ]^{a,c}

[
]^{a,c}

[

]^{a,c}

5.4.2 [ ]^{a,c}

[
]^{a,c}

5.4.3 [ ]^{a,c}

[

]^{a,c}

5.4.4 [ ]^{a,c}

[

]^{a,c}

[

]^{a,c}

5.5 [ ]^{a,c}

[

]^{a,c}

5.5.1 [ ]^{a,c}

[

]^{a,c}

5.5.1.1 [ ]^{a,c}

[

]^{a,c}

[

]^{a,c}

5.5.1.1.1 [ ]^{a,c}

[

]^{a,c}

[
]a,c

5.5.1.2 [ ]a,c

[
]a,c

5.5.1.3 [ ]a,c

[
]a,c

5.5.1.4 [ ]a,c

[
]a,c

5.5.1.5 [ ]a,c

[
]a,c

5.5.1.6 [ ]^{a,c}

[

]^{a,c}

5.5.2 [ ]^{a,c}

[
]^{a,c}

5.5.2.1 [ ]^{a,c}

[

]^{a,c}

5.5.2.2 [ ]^{a,c}

[ ]^{a,c}

5.5.2.3 [ ]^{a,c}

[
]^{a,c}

5.5.2.4 [ ]^{a,c}

[

]^{a,c}

[

]^{a,c}

5.5.2.5 [ ]^{a,c}

[
]^{a,c}

5.5.3 [ ]^{a,c}

[

]^{a,c}

5.5.4 [ ]^{a,c}

[

]^{a,c}

[

]a,c

5.6 [ ]a,c

[ ]a,c

5.7 [ ]a,c

[

]a,c

5.8 [ ]a,c

[

]a,c

5.9 [ ]a,c

[

]a,c

(Last Page of Section 5)

6 REUSABLE LOGIC ELEMENT

A RLET can be developed during a platform or application development project. In the context of a FPGA, an element and module are equivalent. If a RLET is developed during an application development project, it is expected that the RLET is generically developed such that it can be used for other platform and application projects going forward.

The reusable logic element document (RLTED) provides the following categories of information:

- A RLET summary consisting for a general functional description of the element.
- Requirements specification
    - External interfaces
    - Functional requirements
    - Truth table
    - Exception handling
    - Performance requirements
    - Design constraints
- Design specification
    - RLET block diagram
    - RLET list of I/O
    - Critical timing diagrams
    - Configurable registers within the RLET
    - Synthesis and place and route directives

The requirements specification portion of the RLETD will be completed in conformance with IEEE 830 (Reference 13).

The RLET implementation follows the process outlined in subsection 4.5.1.3 and undergoes simulation testing in accordance with subsection 4.5.1.4.

(Last Page of Section 6)

7 QUALITY ASSURANCE PLAN

7.1 PURPOSE

The quality assurance plan (QAP) defines the techniques, procedures, and methodologies that Westinghouse will use to assure quality in the design and test developments of the ALS platform and applications. It was written using IEEE Standard 730, “Standard for Software Quality Assurance Plans” (Reference 16) as guidance. It covers the entire FPGA development process, which includes processes such as requirements specification, design, implementation, source/data control, reviews, change management, and configuration management.

The QAP is a platform and application QAP applied along with the 18-point criteria of 10 CFR Part 50, Appendix B, “Quality Assurance Criteria for Nuclear Power Plants and Fuel Reprocessing Plants” (Reference 6), the most recent edition of ISO 9001 prior to the required compliance date, “Quality Management Systems – Requirements” (Reference 17), and American Society of Mechanical Engineers (ASME) NQA-1-2008/2009a, “Quality Assurance Program Requirements for Nuclear Facilities/with Addenda” (Reference 18). It also includes the reporting requirements of 10 CFR Part 21, “Reporting of Defects and Noncompliance” (Reference 19).

Compliance with the Westinghouse QMS Level 1 (Reference 4) is achieved by implementing Westinghouse QMS Level 2 and Level 3 Procedures.

7.2 SCOPE

Westinghouse is responsible for supplying the ALS equipment, systems, and components for nuclear power generating facilities. ALS boards are generic boards that can be configured for different applications. Because of this, the ALS board life cycle spans the ALS board development and integration into an application-specific system, including Class-1E safety-related or mission-critical systems. Its objective is to develop and produce generic ALS boards for stock and then program them (e.g., CLB 152) and configure them (e.g., slave I/O boards and the CLB 152) later for integration into dedicated systems.

7.3 ALS PROJECT ORGANIZATION AND RESPONSIBILTIES

Section 2 of this document provides an overview of the ALS project organization and responsibilities.

7.4 TASKS

Section 4 of this document provides the description of the platform development tasks. Section 5 of this document provides the description of the application development tasks.

7.5 DOCUMENTATION

Section 4 of this document provides a description of the platform development documentation. Section 5 of this document provides a description of the application development documentation. Documentation structure, format, and archival shall be in accordance with the Westinghouse QMS (Reference 4).

7.6 STANDARDS, PRACTICES, CONVENTIONS, AND METRICS

7.6.1 Purpose

This section defines the applicable standards, practices, conventions, and metrics to be applied to the ALS platform and application projects. Compliance with each item will be monitored and assured through training, as defined in the project management plan. The project management plan may define additional items, as applicable.

7.6.2 Documentation Standards

The project management plan and the supporting plans referenced therein (see Section 15.1), define the standards applicable to the overall development effort. The overall life cycle activities, products, and documentation are defined in Section 4 for the platform development, and Section 5 for application development.

7.6.3 Coding Standards

ALS project FPGA (software) development will be performed in accordance with approved coding standards.

The coding standards include commentary standards.

7.6.4 Nonconformances

During the requirements, design, implementation, and test phases, all discrepancies shall be identified, tracked, and closed using an anomaly reporting system. Identification and resolution of hardware nonconformances are handled per the Westinghouse QMS (Reference 4).

7.6.5 Testing Standards and Practices

Testing is performed in accordance with the Test Plan in Section 9 of this document. IV&V performs verification and validation testing per WNA-PV-00129-GEN (Reference 5).

7.6.6 Process Metrics

Quality metrics are tracked using the error reporting system throughout the duration of the project life cycle. The PM tracks metrics, performs trend analysis, and establishes corrective actions, as needed. These metrics are reviewed at periodic management reviews, as defined by the project management plan.

7.7 REVIEWS AND AUDITS

7.7.1 Purpose

This section defines the minimum technical and managerial reviews and audits to be performed. The project management plan can include requirements for additional reviews and audits.

The QA team has the authority and freedom to perform planned and unplanned monitoring activities (for example, audits, assessments, or reviews) at any time during the ALS project life cycle.

IEEE Standard 1028, “IEEE Standard for Software Reviews and Audits” (Reference 20) lists five types of software reviews. The reviews defined in IEEE Standard 1028 (Reference 20) are addressed as follows:

1. Managerial reviews – See subsection 7.7.9
2. Technical reviews – Design reviews per Westinghouse QMS (Reference 4) and see subsections 7.7.3 and 7.7.4
3. Inspections – Performed by IV&V in accordance with WNA-PV-00129-GEN (Reference 5)
4. Walk-throughs – Design reviews per Westinghouse QMS (Reference 4) and IV&V plan, WNA-PV-00129-GEN (Reference 5)
5. Audits – See subsection 7.7.8

7.7.2 Software Requirements Review

As discussed in Sections 4 and 5, ALS projects produce or revise requirements and design specification documents. These documents and the code releases are reviewed by the FPGA design team in accordance with the Westinghouse QMS (Reference 4). The reviews are performed and documented using a checklist.

7.7.3 Design Review

As discussed in Sections 4 and 5, FPGA requirements documentation will undergo a design review in accordance with the Westinghouse QMS (Reference 4).

7.7.4 Critical Design Review

The critical design review is equivalent to the final design review per the “Final Design Review” definition in the Westinghouse QMS (Reference 4). The final design review is conducted in accordance with the requirements of the Westinghouse QMS (Reference 4). At the PM’s and ELM’s discretion, an intermediate design review may be performed in addition to the final design review.

7.7.5 Software Verification and Validation Plan Review

Verification and validation of the FPGA is performed in accordance with WNA-PV-00129-GEN (Reference 5). This V&V plan is reviewed and approved in accordance with the Westinghouse QMS (Reference 4).

7.7.6 Functional Audit

Functional audits (FAs) verify that the configuration item meets the functional requirements including performance. The FA helps confirm that there are no unintended functional characteristics. IV&V conducts this audit via the requirements traceability analyses defined in WNA-PV-00129-GEN (Reference 5).

FAs may be conducted early in the prototype stage to ensure that the design will meet the requirements. The final FA is conducted after testing is complete and documented data is available to demonstrate that the platform or application meets the requirements. The final FA must be conducted before the physical audit if there is any possibility that amendments will arise as a result of the final FA. FA criteria are met through the execution of the test plan and WNA-PV-00129-GEN (Reference 5). IV&V satisfies this audit via the requirements traceability analysis defined in WNA-PV-00129-GEN (Reference 5), which includes testing.

7.7.7 Physical Audits

IV&V will verify that the FPGA (software) and its documentation are consistent and are ready for delivery. The IV&V summary report will document the reviews of the FPGA (software) and its documentation.

7.7.8 In-Process Audits

In-process audits verify design consistency, including the following, consistent with the scope of the project:

1. Software versus design documentation
2. Interface specifications (hardware and software)
3. Design implementation versus functional requirements
4. Functional requirements versus test descriptions

The QA team performs oversight of the ALS platform and application development and related processes. The QA team conducts in-process audits per the Westinghouse QMS (Reference 4). The scope of in-process audits will be determined on a project-specific basis and documented in the project management plan.

7.7.9 Managerial Review

Managerial reviews are performed in accordance with the project management plan and WNA-PV-00129-GEN (Reference 5). Managerial reviews assess the execution of the actions and items identified in the QAP.

7.7.10 Software Configuration Management Plan Review

IV&V review of the plan 6002-00002-P, "ALS Configuration Management Plan" (Reference 21) is conducted during the planning phase of the life cycle. IV&V performs an assessment to verify that the methods defined in the plan are complete and adequate.

7.7.11 Post-Mortem Review

The post-mortem reviews are conducted per the project closeout plan in the project management plan. This review is held at the conclusion of the project to assess the development activities implemented on that project and to provide recommendations for appropriate actions.

7.8 TESTS

Tests, including V&V tests, are performed per the test plan in Section 9.

7.9 PROBLEM REPORTING AND CORRECTIVE ACTIONS

Measures taken to resolve problems, mitigate their consequences, address the extent of condition, the extent of cause, and to minimize the potential for event recurrences are addressed per the corrective action process in the Westinghouse QMS (Reference 4). Issues are recorded and tracked to resolution.

Audit findings and results are also documented in the Westinghouse corrective action system.

Software and software document issues are handled per subsection 7.6.4 of the QAP.

7.10 TOOLS, TECHNIQUES, AND METHODOLOGIES

The tools used for the ALS project and their applicable activities are defined by a design tool document. IV&V tools are specified in WNA-PV-00129-GEN (Reference 5). These tools are used for design activities and software quality assurance (i.e., IV&V) activities. Tools credited for qualification and acceptance testing are validated and approved before use.

Standard commercial software packages (e.g., Microsoft Excel®, Microsoft Word®, and Adobe Acrobat®) are used to create reports and other documentation. The packages are not used to create customer delivered software.

7.11 CODE CONTROL

Software and design data are maintained in a separate project repository managed by a version control system. Requirements for the control of software are described in 6002-00002-P, “ALS Configuration Management Plan” (Reference 21).

7.12 MEDIA CONTROL

The methods used for media control are identified in the Westinghouse QMS (Reference 4) and are also addressed by 6002-00006, “ALS Security Plan” (Reference 22).

The secure development environment (SDE) is described in 6002-00006, “ALS Security Plan” (Reference 22).

7.13 SUPPLIER CONTROL

Supplier control is implemented in accordance with the Westinghouse QMS (Reference 4).

7.14 RECORDS COLLECTION, MAINTENANCE, AND RETENTION

All records are maintained in accordance with the Westinghouse QMS (Reference 4).

7.15 TRAINING

Training for ALS project personnel is planned and executed per the Westinghouse QMS (Reference 4). Training curricula and personnel transcripts are maintained in an online learning system. Additional details are provided in the project management plan.

7.16 RISK MANAGEMENT

Project risk management is conducted per the Westinghouse QMS (Reference 4) as supplemented by the project management plan. Risks are managed and discussed during regular project meetings and maintained by the PM.

(Last Page of Section 7)

8 SOFTWARE SAFETY PLAN

8.1 INTRODUCTION

8.1.1 PURPOSE

The goal of the safety plan is to enable the development of safety critical software for ALS projects that has reasonable assurance that software defects do not present severe consequences to public health and safety.

8.1.2 SCOPE

The safety objective of this plan is to provide procedures and methodologies for the development, procurement, maintenance, and retirement processes of ALS safety-critical software to mitigate the potential of a software defect jeopardizing the health and safety of the public.

Any acceptable risks and safety objectives specific to a project shall be defined in the specific project management plan for a given system implementation.

This plan is prepared in accordance with Branch Technical Position (BTP) 7-14, “Guidance on Software Reviews for Digital Computer-Based Instrumentation and Control Systems” (Reference 23), and NUREG/CR-6101, “Software Reliability and Safety in Nuclear Reactor Protection Systems,” (Reference 24). It applies to all ALS safety critical software whose failure could result in severe consequences to public health and safety. For ALS applications, safety critical software is defined as software belonging to the protection class as defined in Section 1.2.

This plan does not apply for the development of the ALS platform. NUREG/CR-6101 (Reference 24) states: “The Software Safety Plan is required for safety critical applications, such as reactor protection systems, to make sure that system safety concerns are properly considered during the software development.”

As ALS v2 is a platform, and not an application, many software safety activities, such as safety analyses, are more appropriate within the scope of ALS v2 application projects (see Section 5). However, it should be noted that several planning documents contain attributes of a software safety plan.

8.2 DEFINITIONS, ACRONYMS, ABBREVIATIONS, AND REFERENCES

See the front matter of this document for definitions, acronyms, abbreviations, and references.

8.3 SOFTWARE SAFETY MANAGEMENT

In compliance with NUREG/CR-6101 (Reference 24), this section provides a description of the software safety organization and the management of software safety activities and safety analysis requirements.

8.3.1 Organization and Responsibilities

Section 2 defines the organization that is responsible for design and implementation of ALS protection software.

The mechanism for communicating safety concerns raised by project staff to software safety personnel is defined in the Westinghouse QMS (Reference 4).

8.3.2 Resources

Management shall develop an early understanding of the resources required to develop protection-class software so that these resources are put in place when they are required. The PM and the IV&V team leader shall determine the resources required to implement an ALS application. ELMs shall assign the appropriate resources to the PM and IV&V team leader. The following resources are considered for both the design and IV&V team:

- Personnel
- Test materials and data
- Computers and other equipment
- Equipment support
- Tools
- Financial and schedule

The PM shall maintain an up-to-date resource plan and assure that the resources are made available when required.

Project schedules and resource allocations are established via the project management plan.

8.3.3 Staff Qualifications and Training

The qualifications and training requirements for those personnel performing software safety functions are primarily the same as those for performing the software design.

Table 9.6-1 identifies the personnel that will perform the tasks identified in NUREG/CR-6101 (Reference 24), subsection 3.1.5:

Table 8.3-1: Software Safety Task Assignments

Task	Assignee
Define safety requirements	Design Team
Design and implement safety-critical portions of the system	Design Team
Perform software safety analysis tasks	IV&V Team
Test safety-critical features	IV&V Team
Audit software safety plan implementation	Quality organization
Perform process certification	Quality organization (subsection 8.3.13)

One of the most important factors in developing reliable software is the development and use of qualified staff. In assessing the training requirements, the ELM considers that:

- Training needs vary by individual
- Training and retraining may be needed at various project phases
- Staff qualification and training need to be periodically reassessed

In addition to the above, the IV&V team shall be trained in the tools, techniques, and methodologies described in WNA-PV-00129-GEN (Reference 5).

The ELMs assure that all personnel participating in the design, implementation, testing, and verification of software are qualified to perform their assigned tasks. Since there is currently no industry sanctioned certification program for protection and important-to-safety class software personnel, the ELM assesses the capabilities of candidates and selects appropriately qualified personnel based on the manager's experience.

In determining whether any candidate is qualified, the ELM considers whether the candidate:

- Understands the system and potentially hazardous effects, as described in Section 8.4Understands the job to be performed
- Has (or can obtain) working knowledge of system software and tools required to do the job
- Possesses the combination of skills and knowledge to perform the job through a proper level of formal education, supplemental training, and experience
- Understands the related quality assurance, configuration management, and verification and validation plans
- Can produce reliable software and good documentation, and can implement required quality assurance practices

Throughout a project, requirements and tasks may change. The ELM shall periodically reassess the qualifications of all personnel working on protection-class software, particularly when specific changes to the project become known. The ELM may direct additional training before the changes are effective to ensure a fully qualified project team.

Personnel performing software safety reviews shall meet the qualifications for an independent reviewer, as defined in the Westinghouse QMS (Reference 4).

8.3.4 Software Life Cycle

The software life cycle to be implemented for ALS application development activities, including IV&V, is defined in Subsection 3.2.1. Section 8.4 describes the relationship among specific software safety analysis tasks and the associated activities for each phase of the software life cycle.

8.3.5 Documentation Requirements

The documentation for ALS software shall be prepared in accordance with the requirements in Sections 4 and 5, and incorporates the software safety documentation requirements. The change and approval process for the protection-class software portions of project documentation is the same as for other documentation as specified in Section 7.7.

8.3.5.1 Software Project Management

A project management plan shall be developed that will coordinate both the system development, software safety, and quality assurance activities to identify the prescribed procedures and provide the adequate, allocated resources for their proper execution.

8.3.5.2 Software Configuration Management

6002-00002-P, “ALS Configuration Management Plan” (Reference 21) contains the requirements for software configuration management. Any deviations to these requirements shall be documented in the project specific project management plan. 6002-00002-P (Reference 21) defines specific software configuration management (SCM) responsibilities for an ALS project and covers each phase of the software life cycle.

8.3.5.3 Software Quality Assurance

Section 7 is the QAP that describes the requirements and methodology to be followed in developing, acquiring, using, and maintaining safety-critical software. This QAP follows the guidance in IEEE Standard 730 (Reference 16).

8.3.5.4 Software Safety Requirements

The system requirements documentation specifies the safety requirements to be met by the software to avoid or control system hazards.

8.3.5.5 Application FPGA Design Specification

The application FPGA design specification includes descriptions of the software design elements that satisfy the software safety requirements.

8.3.5.6 Software Development Methodology, Standards, Practices, Metrics, and Conventions

The standards, practices, metrics, and conventions to be applied to the ALS project are defined in Section 7.6.

8.3.5.7 Test Documentation

Test documentation includes test plans, test procedures, and test reports. Test procedures incorporate test design and test cases.

8.3.5.7.1 Test Plans

The test plans provide a high-level description of tests that will be conducted for the ALS project. The plan will contain the method for defining the requirements to be tested, the method for establishing the acceptance criteria, and how it will be documented. It also defines the methodology for the disposition of test exceptions (errors). This document is verified against the outputs generated from the requirements phase of IV&V for completeness. All prerequisites for testing shall also be identified in the detailed test sections.

8.3.5.8 Test Procedures

The test procedures are the instructions for the actual tests conducted on the ALS software. They include test setup, precautions and limitations, prerequisites, and the test cases used to validate proper operation. The test procedures are verified against both the test plan and outputs generated from the requirements phase of IV&V.

8.3.5.9 Test Reports

The test reports document the execution of the test procedures. In addition to attaching the signed and checked off test results, the test reports provide an overall summary of the test results and the resulting exception reports generated during the test. The system configuration at the time of test execution is also documented in the test reports.

8.3.5.10 Software Verification and Validation

The software IV&V documentation is described in WNA-PV-00129-GEN (Reference xiii5).

8.3.5.11 Reporting Safety Verification and Validation

IV&V reporting is described in WNA-PV-00129-GEN (Reference 5).

8.3.5.12 Software User Documentation

User documentation (i.e., technical manual) is described in subsection 5.5.3.

8.3.5.13 Results of Software Safety Requirements Analysis

The results of the software safety requirements analysis, as described in subsection 8.4.2 below, shall be documented in the Requirements Phase section of the IV&V Report.

8.3.5.14 Results of Software Safety Design Analysis

The results of the software safety design analysis, as described in subsection 8.4.3 below, shall be documented in the Design Phase section of the IV&V Report.

8.3.5.15 Results of Software Safety Code Analysis

The results of the software safety code analysis, as defined in subsection 8.4.4, shall be found in the IV&V report for the Implementation Phase of the software life cycle. Any changes will be documented in either IV&V discrepancy reports or as suggestions in the IV&V report.

8.3.5.16 Results of Software Safety Test Analysis

The results of the software safety test analysis, as defined in subsection 8.4.6, shall be found in the IV&V report for the Testing Phase of the software life cycle.

8.3.5.17 Results of Software Safety Change Analysis

The results of the software safety change analysis, as defined in subsection 8.4.8, shall be found in the IV&V report. For each software life cycle that is revisited by the design team, the IV&V team will analyze the impact on the previous life cycle phase as well as the phase it is analyzing. The results of each phase’s analysis will be found in the IV&V report for that software life cycle phase.

8.3.6 Software Safety Program Records

Records generation and maintenance procedures required for ALS software are described throughout this document.

Before the requirements phase is completed and after the overall system design is known, an evaluation is made to determine the safety critical hazards posed by the system through its interfaces. The analysis assumes that a worst-case scenario of possible errors (hardware or software) has occurred in the system. Based on this assumption, the analysis results in an identification of system malfunctions that are injurious to public health and safety.

For each hazard identified above, the analysis further determines whether a software malfunction could produce the hazardous condition. These software hazards are identified in the software hazards analysis report as described in subsection 8.4.1. Each software producible hazard is evaluated during each phase of development of the safety critical software. The software hazards analysis report is issued by the design team and is an input to the IV&V team.

Results of IV&V analyses performed on requirements, design, code, testing, and other technical documentation are documented in the IV&V phase summary reports and the final IV&V report. Information on suspected or confirmed safety problems in the prerelease or installed system is recorded in the final IV&V report. Results of audits performed on software safety program tasks are documented in the Quality organization’s audit report. Results of safety tests conducted on all or any part of the entire system are documented in the test report. Training records are maintained by management per the Westinghouse QMS (Reference 4). Software safety certification is documented in the code certificate as described in WNA-PV-00129-GEN (Reference 5).

Retention of software safety program records is in accordance with the Westinghouse QMS (Reference 4). The initiation and completion criteria for software safety program tasks for each phase in the software life cycle are defined in Sections 4 and 5.

The tracking system used to confirm that hazards and their status are tracked throughout the software life cycle through retirement is the requirements traceability analysis (RTA) and RTM.

8.3.7 Software Configuration Management Activities

A key factor in developing reliable software is strict and detailed configuration management. Software configuration management activities for ALS software are described in 6002-00002-P, “ALS Configuration Management Plan” (Reference 21).

8.3.8 Software Quality Assurance Activities

Software quality assurance activities for ALS software are described in Section 7.

8.3.9 Software Verification and Validation Activities

Software verification and validation activities for ALS software are described in WNA-PV-00129-GEN (Reference 5).

8.3.10 Tool Support and Approval

Section 7.10 describes the use of software tools that are used in development of ALS platform and applications. Tools may produce better program structure and more reliable software through the automation of repetitive or time-consuming tasks. The EPM and IV&V team leader approve the use of any tool. This approval is based on an evaluation of the tool's readiness for use on a project involving protection-class software. This evaluation considers:

- The tool's past performance
- The extent of tool validation already performed
- The consistency of tool design with planned use
- The use of tool upgrades
- The retirement of tools
- The restrictions on the use of the tool due to limitations

The inadvertent introduction of software hazards by project tools is mitigated by the proper use of techniques for software configuration management, software quality assurance, and IV&V as described in this document.

8.3.11 Previously Developed or Purchased Software

Existing commercial software may be used in protection and important-to-safety applications if it is qualified using a commercial grade dedication program (CGDP), such as the one described in the Westinghouse QMS (Reference 4).

Existing nuclear power plant (NPP) non-commercial software that has been actively used in a nuclear power plant maybe used for the same class of software under this document, provided it has been maintained under an acceptable quality plan with an active program for problem and corrective action reporting. This software shall also have adequate design documentation, user documentation, and well-commented source code. This software shall have been verified and validated under another program that is judged by the IV&V team to be acceptable.

The Westinghouse QMS (Reference 4) describes the change analysis for previously developed software to preserve the safety integrity. This process is modeled after the process described in WCAP-17266-P, "Common Q Platform Generic Change Process," (Reference 25).

8.3.12 Subcontract Management

Section 7.13 specifies the provisions for ensuring that subcontractor software meets established software safety program requirements.

8.3.13 Process Certification

An audit report from an in-process audit described in subsection 7.7.8 is prepared by the QA team to document that the software-related activities were performed in accordance with the Westinghouse QMS (Reference 4) and its implementing procedures.

8.4 SOFTWARE SAFETY ANALYSES

8.4.1 Software Safety Analyses Preparation

It is vitally important to understand the ways that a system could potentially present hazards to public health and safety. The system design and review techniques described in this document are used to avoid, preclude, or mitigate the impact of potential software hazards in systems built using the ALS platform. Systems that include both protection- and important-to-safety-class software need to postulate in the software hazards analysis potential software hazards in the important-to-safety-class software and the impact on protection-class software.

A software hazards analysis (SHA) will identify the following:

- Hazardous system states – Before the requirements phase is completed and after the overall system design is known, an evaluation is made to determine the safety hazards posed by the system through its interfaces could be injurious to public health and safety. The plant safety analysis defines the safety-critical hazards (accidents) posed by the plant that may be injurious to public health and safety. The failure modes and effects analysis performed for the specific ALS application analyzes the vulnerability to single failures at the hardware module level, including existing compensating provisions (hazard controls) within the design of each system. These two sources form the design bases for software-safety requirements for the ALS application.

- Sequences of actions that can cause the system to enter a hazardous state – For each identified hazard, the analysis determines whether a software malfunction could produce the hazardous condition or if the hazard could affect software operability. These hazards are identified in the software hazards analysis report. Each software-related hazard is evaluated during each phase of development of the protection-class software. NUREG/CR-6430, “Software Safety Hazard Analysis” (Reference 26) shall be used as a guide in performing this analysis.

- Sequences of actions intended to return the system from a hazardous state to a non-hazardous state – For each hazardous state, the system design must account for returning the system to a non-hazardous state. In preparing the application FPGA requirements specification, the software developer considers techniques that can avoid a hazardous condition or return the system to a non-hazardous state. The result of the requirements phase may be a set of required or forbidden design, coding, or testing techniques. The requirements phase may also identify specific tests to be performed or the implementation of certain hazard recovery techniques.

The application requirements and application design specification provide the high-level system design as required in subsection 4.4.1 b) of IEEE Standard 1228-1994, “IEEE Standard for Software Safety Plans” (Reference 27). The interfaces between the software and the rest of the system are defined in the software requirements specification.

8.4.2 Software Safety Requirements Analysis

In preparing the application FPGA requirements, the software developer considers techniques that can avoid a hazardous condition. The result of the requirements phase may be a set of required or forbidden design, coding, or testing techniques. The requirements phase may also identify specific tests to be performed or the implementation of certain hazard recovery techniques.

Refer to WNA-PV-00129-GEN (Reference 5) for a description of the software safety requirements analyses performed. These activities provide reasonable assurance that each system safety requirement is satisfied by the software safety requirements.

8.4.3 Software Safety Design Analysis

Refer to WNA-PV-00129-GEN (Reference 5) for a description of the software safety design analyses performed. These activities provide reasonable assurance that each software safety requirement is satisfied by the software safety design.

8.4.4 Software Safety Code Analysis

Refer to WNA-PV-00129-GEN (Reference 5) for a description of the software safety code analyses performed. These activities provide reasonable assurance that each software safety design element is satisfied by the software safety code.

8.4.5 Software Integration Safety Analysis

The software integration safety analysis is performed as part of the software safety test analysis. See subsection 8.4.6 for the software safety test analysis.

8.4.6 Software Safety Test Analysis

See Section 9, for a description of the software safety test analyses performed for system level testing. These activities provide reasonable assurance that each system and software safety requirement is tested.

8.4.7 Software Installation Safety Analysis

WNA-PV-00129-GEN (Reference 5) fulfills the requirements for a software installation safety analysis. This final safety analysis verifies that the installed system operates correctly.

8.4.8 Software Safety Change Analysis

WNA-PV-00129-GEN (Reference 5) fulfills the requirements for a software safety change analysis. These activities provide reasonable assurance that changes to safety-critical software do not create a new hazard, impact a previously resolved hazard, exacerbate a currently existing hazard, or adversely affect any safety-critical software design elements.

8.5 POST DEVELOPMENT

The recipient of the software does not have access to modify FPGA logic, which provides protection against inappropriate maintenance.

Software personnel shall be trained in the procedures in the Westinghouse QMS (Reference 4) involving exception reporting and correction.

8.5.1 Training

ALS customers are responsible for providing safety training for the users, operators, and maintenance and management personnel, as appropriate.

Westinghouse personnel assigned to work on any activity in the software life cycle process must complete training on this document in accordance with the Westinghouse QMS (Reference 4).

8.5.2 Deployment

8.5.2.1 Installation

Since the FPGA logic is installed in the factory, there is no FPGA installation after the system is delivered.

8.5.2.2 Startup and Transition

Changes to installed systems may be disruptive to operations, particularly if problems occur or the resulting system operates differently. A technical manual will be prepared addressing the following (as appropriate to the configuration of the system being installed):

- Fallback modes for the new system
- Startup of backup components and subsystems
- Startup of the new system
- Parallel operation with backups
- Parallel operation of the old system and the new system
- Subsystem vs. full system operation
- Switchover to full system operation
- Validation of results from the new system

- Cross validation of results between the old system and the new system
- Fallback in the case of failure of the new system, including fallback to an old system if one exists

8.5.2.3 Operations Support

Documentation of the system and its software is supplied as described in Sections 4 and 5. This documentation includes design documents, technical manuals, and instructions for maintenance expected by plant personnel.

8.5.3 Monitoring

Problem reporting and corrective action contains requirements for monitoring the use of delivered software and associated exception reporting.

In addition, FPGA binary images cannot be modified outside of the factory, protecting against unauthorized modification of the code or data.

8.5.4 Maintenance

Software changes during all software life cycles are executed according to the Configuration Management Plan in 6002-00002-P (Reference 21) and the Software Maintenance Plan in Section 12.

8.5.5 Retirement and Notification

6002-00002-P (Reference 21) describes the retirement of software and associated notification to current users.

(Last Page of Section 8)

9 TEST PLAN

9.1 INTRODUCTION

9.1.1 Purpose

The ALS test plan covers the test planning for the ALS v2 platform and applications. This document describes testing activity scope, approach, resources, and scheduling. It identifies:

- The features to be tested
- The testing tasks to be performed
- The personnel responsible for each task
- The risk associated with the plan

Supporting configuration management and quality plans are defined in the project management plan.

9.1.2 Scope

The ALS test plan covers ALS v2 platform integration testing and ALS application testing. The information presented in this plan provides the prescribed details for a testing program.

9.1.3 Standards

IEEE Standard 829-1998, “IEEE Standard for Software Test Documentation” (Reference 28) is used as guidance for the generation of ALS project test documentation, which contains FPGA designs (software). Test case specifications may be incorporated within the body of a larger test document, generally within the corresponding test procedure or V&V simulation specification. This approach is consistent with Regulatory Guide 1.170, “Software Test Documentation for Digital Computer Software Used in Safety Systems of Nuclear Power Plants” (Reference 15), where it is acceptable for individual documents to be incorporated into larger test documents, provided the identity of each component document is retained.

9.1.4 OBJECTIVE

The ALS application testing process validates the functional requirements of the ALS safety systems applied to a specific project and/or a component being developed for the ALS platform. This plan is intended to guide a qualified test team to prepare detailed test procedures that test the ALS project.

This section serves as a generic test plan outline. It is expected that each ALS project creates a project-specific test plan to capture project specific testing requirements.

9.2 TEST ITEMS

The project-specific test plan will define the test items for the ALS project.

9.3 FEATURES TO BE TESTED

Testing will be conducted to ensure the test items function in accordance with the requirement and design specifications for the ALS project. All testable requirements for ALS safety system features and functions shall be tested with explicit acceptance criteria. The requirements shall be derived from the requirements traceability process. Each testable feature and function identified within the RTM shall be tested with a procedure that is traceable to the item within the RTM. Maintaining the RTM shall provide evidence of complete test coverage of ALS safety system features and functions.

The project-specific test plan will define the requirements and design specifications applicable for the ALS project.

9.3.1 Test Design Specification

Each test technique has its own test design specification, which describes the test cases unique to that particular test technique.

The project-specific test plan will define the test design specifications applicable for the ALS project.

9.4 FEATURES NOT BEING TESTED

This plan covers the ALS platform and application testing.

The project-specific test plan will define features not being tested for the ALS project if any, in accordance with IEEE Standard 829 (Reference 28).

9.5 ALS PLATFORM TESTING APPROACH

This section identifies the high-level approach for ALS platform testing.

The design team or test team will perform the following testing activities, and the IV&V team will review the testing results:

- FPGA simulation

IV&V performs the following testing activities:

- Integration-level testing

The FPGA simulation engineer from the design team will not have been involved in the creation of the design and will meet the independence requirements from 10 CFR 50, Appendix B, Criteria III (Reference 6).

Organizational independence between the design and IV&V teams is defined in Section 2 of this document.

The project-specific test plan will define additional details on the testing approach for the ALS project.

9.5.1 ALS Simulation Testing

The ALS platform RTL that is generated during the FPGA design process is placed in a software simulation environment. Test cases are typically written in the form of random, constrained random, and directed to cover design requirements, particularly the FPGA requirements. The FPGA design simulation is part of the overall test strategy for the platform, but FPGA design simulation testing will follow its own test plan. The test plan, test specifications, and procedures shall comply to IEEE Standard 829-1998 (Reference 28).

9.5.2 Integration Testing

Integration testing involves designing and executing a comprehensive test set – through various test techniques – that reasonably ensures that the ALS platform functions correctly, as defined by the product requirements.

ALS platform integration testing uses the following test techniques:

- ALS board-level testing
- Board hardware circuit testing

9.5.2.1 ABTS Board-Level Testing

The objectives for ALS board test system (ABTS) board-level testing are:

1. Test ALS requirements and circuits, as appropriate
2. Provide a test environment to leverage for future manufacturing tests

Board-level test cases are implemented using the ABTS. Board-level testing detects a wide range of design defects, performance issues, and manufacturing defects related to both the FPGA and non-FPGA requirements. Board-level tests are defined in the test design specification.

9.5.2.2 Board Hardware Circuit Testing (Manual Test)

Hardware circuit tests are a procedure set used to test functionality and performance (e.g., analog circuit accuracy across temperature, power supply performance, reset circuit behavior, and effects of component shorts on self-test circuitry) that board-level testing is unable to test. Manual tests are defined in the test design specification.

9.5.3 INSPECTION, REVIEW, OR ANALYSIS

If a requirement for a test item cannot be tested via one of the other identified test techniques, an inspection, review, or analysis may be conducted with proper documentation and justification as to how the requirement has been verified to be correctly implemented and functioning as intended.

9.5.4 TESTING COMPREHENSIVENESS & COVERAGE

Test item requirements and features must be tested at least once using one or more test techniques. To demonstrate test comprehensiveness and coverage, RTM will be developed to document that each requirement is tested, and which test techniques are used to fulfill the testing coverage.

9.5.5 IV&V Review of Testing Results

The IV&V team will review the testing results and document their review in accordance with WNA-PV-00129-GEN (Reference 5).

9.6 ALS APPLICATION TESTING APPROACH

This section identifies the high-level approach for ALS application testing.

9.6.1 Application Integration Tests

Once the ALS platform is standardized and reaches certification and can be used on projects, a project will create and load application-specific FPGA and NVM images onto the 152 CLB(s). The slave boards are configured with a generic platform FPGA image and application-specific NVM images. The ALS-152 CLB is configured with application-specific FPGA and NVM images. Overall, any platform boards with applications-specific FPGA images shall be placed through some or all of the ALS board design verification testing. Additional tests or test cases may be necessary to fulfill test coverage. The project-specific test plan shall cover the scope of the application integration tests.

Additional integration-type of tests are listed below and are only discussed here as a reference. A project-specific test plan will be developed to cover ALS as integrated into a larger system where one or more of these test techniques will be used. These tests may be run one or more times as needed for Nth of kind systems.

9.6.1.1 Subassembly Hardware Test

A subassembly hardware test (SHT) shall confirm proper operation of hardware subassemblies following manufacture and prior to installation in a safety system cabinet. An SHT shall verify the requirements specified for the product.

9.6.1.2 Cabinet Hardware Test

A cabinet hardware test (CHT) is performed after the SHT. A CHT shall check the cabinet hardware following final assembly and verify that the AC and DC power distribution within the cabinet is correct. It might be necessary to use the ALS during the test or include it as a target of CHT.

With the electronic modules removed and the cabinet de-energized, visual inspection and standard electrical tests (i.e., resistance measurements) shall be performed to verify adequate power, neutral, and ground wiring insulation. With the cabinet energized, standard electrical tests (i.e., voltage measurements) shall be performed to verify that all AC and DC voltage levels are within required tolerances.
With the electronic modules installed and the cabinet energized, the standard electrical tests shall be repeated to verify that all AC and DC voltage levels remain within required tolerance.

9.6.1.3 Integration Test

The integration test shall be executed prior to completion of the system validation test or factory acceptance testing (FAT). The integration test shall verify integration of the released application software (e.g., ASU/display) and ALS 152 CLB images with the production hardware or with a system that is functionally equivalent. A functionally equivalent system can be a test bed or an equivalent set of production hardware (e.g., a unit of the same design for a different site deliverable system). A test bed shall be configured with hardware that provides functionally equivalent configuration to the production hardware for the testing performed.

An integration test shall address the safety system requirements documented in the application requirements specification.

Integration testing is used as part of system validation testing when validating the design and as part of the FAT testing to demonstrate the deliverable system has been properly integrated.

The integration test can be segregated into tests that are performed on a cabinet level, on a division or channel level, or on a system level. For tests on a channel level, cabinets within a safety-system division shall be interconnected and integrated for this test. Functions implemented in a single cabinet within a division or across multiple cabinets within a division shall be tested. Communications between cabinets within a division, data flow, control functions, signal loops, redundancy, interdivisional voting logic, and fault tolerance shall be tested. Functions implemented across multiple divisions shall be tested with the system fully integrated during the system validation test or FAT. Functions shall be tested by confirming the correct relationship between test input and output signals. Each input signal shall be exercised to verify mapping with expected outputs.

9.6.1.4 System Validation Test

The system validation test shall be completed before site acceptance testing is completed.

The system validation test is a set of tests developed to validate the hardware design, software design, and the system integration at the functional level. The system validation test shall address the safety system requirements documented in the application requirements specification.

Aspects of system validation testing can be performed on a single division to show compliance of functions that are contained within one division. The system validation testing is also performed on multiple divisions to show compliance of functions that require communication with other divisions.

The system validation test shall test the integration of the cabinets in the safety system as defined by the project-specific test plan.

The system validation test shall verify that the cabinets in the safety system divisions (as defined the project-specific test plan) satisfy system-level functional and performance requirements. The test shall verify correct communications between cabinets in different divisions.

System validation functional testing shall focus on system-level functional requirements, requiring cabinet interaction both within the division and across divisions.

Testing shall verify system boundaries to other I&C systems, communications between divisions (including interface loading), data flow, control functions, signal loops, redundancy, interdivisional voting logic, and fault tolerance incorporated in the system's design. Overall system time response shall be verified.

The following test items shall be included in the system validation test:

- Safety functions
- Communications
- Displays
- Diagnostics
- Performance
- Error handling – potential errors shall be handled with known consequences
- Communications – all defined outputs shall be broadcast and received correctly within the channel
- Redundancy – all shared inputs shall produce the same output from redundant processors
- Diversity – all functionally diverse signals shall be verified for correctness in termination

Table 9.6-1 provides a description of the tests performed during system validation testing and FAT.

For a system validation test to be credited as FAT, it must be performed on the delivered equipment.
As an alternative to functional testing with production hardware, a system validation test can be performed with a test bed. This test bed shall be a functionally equivalent configuration to the production hardware. Alternatively, system validation testing can be performed on any FOAK deliverable system. As design changes are introduced, a regression analysis needs to be performed to determine what tests need to be repeated or introduced to maintain the level of system validation achieved during the FOAK test

program. The system validation tests required by the regression analysis may be performed on the deliverable equipment as a separate section of the FAT or on surrogate equipment consistent with the regression testing methods described in Section 9.10.

9.6.1.5 Factory Acceptance Test

The FAT is to be executed on a deliverable system and must be completed and meet its approved requirements before the customer accepts the system. The FAT is typically performed in the factory but some portion of the test can be performed at site if agreed to with the customer. When performed on a deliverable system, the System Validation Test can fulfill the role of the Factory Acceptance Test. The FAT (if required) shall be executed and must be satisfactorily completed prior to shipment of the safety system to the customer. The FAT, as an integrated test covering multiple systems, is based on contractual obligations. External system interface testing shall be specified in the FAT procedure. The intent of the FAT can be accomplished by a combination of the test types listed above.

The purpose of the FAT is to demonstrate that the complete system is integrated and functional. To this end, the optimum scenario is to perform this test in the manufacturing facility. Prior to acceptance of equipment by the customer, a FAT is performed as a manufacturing test to provide evidence to the customer that the system meets its requirements and provides confidence that the site installation and integration activities will be successful. FAT includes tests that are performed on the deliverable system for each deliverable system. The FAT test, together with the documentation of the prior V&V activities demonstrate full compliance to the requirements.

FAT is performed to:

- Demonstrate that the system has been manufactured correctly and is acceptable to the customer
- Demonstrate (in conjunction with V&V) compliance to requirements for customer acceptance
- Reduce the risk associated with deferring compliance demonstration to the site activities (e.g., site acceptance testing [SAT], preoperational testing, etc.)
- Demonstrate aspects of the design that would not be practical once full integration is achieved due to limitations on interfaces that are connected in the plant.

The completeness of the FAT is demonstrated by a combination of the tests performed and reference to prior tests on the first application system that remain valid because the design is identical in all relevant aspects. Such references must be specific as to procedures and test cases or a reference trail. The results of these reference tests must be kept under configuration management, and any open items arising from the test must be either resolved or carried forward to the follow-on system.

The following test items shall be included or demonstrated in the FAT:

- Safety functions
- Communications
- Operability of displays
- Diagnostics associated with hardware specific inputs (e.g., door alarms, temperature alarms, breaker status, etc.)
- Performance (i.e., accuracy, time response, etc.)

Table 9.6-1 provides a description of the tests performed during system validation testing and FAT.

Table 9.6-1: Comparison of System Validation Test and FAT

Test Item	**Design Aspect**	**System Validation Test** **(First Application)**	**FAT** **(Nth Application)**
FPGA and NVM Integration	FPGA and NVM Integration	FPGA and NVM images are loaded into the target hardware. Cyclic redundancy checks (CRC) confirm images on the target hardware are consistent with design documentation.	Same as for first application.
Safety Functions	Safety Functions	Each safety function demonstrated to properly respond to each input per requirements or any functional logic diagrams (FLDs); component actuation outputs respond to system-level actuations appropriately; manual actuations at system and component level are effective.	Each safety function demonstrated to properly respond to each input per FLDs; component actuation outputs respond to system-level actuations appropriately; manual actuations at system and component level are effective. (Demonstrates trips and actuations are functioning but does not need to retest the software logic that has been previously verified in the first application. Time response testing can be used to demonstrate trips and actuations).
	Voting Logic	All combinations including bypasses and forced trips.	Subset of combinations demonstrating that each input to voting logic is effective. (Time response testing can be used to demonstrate voting logic inputs from each division.)
Communications	Intra-Cabinet Communications	Each signal shown to connect to every intended destination.	Links confirmed to be operational through diagnostics; no signal tracing.
	Inter-Cabinet Communications (within channel and between channels)	Each signal shown to connect to every intended destination.	Links confirmed to be operational through diagnostics; representative signals are traced.

Table 9.6-1: Comparison of System Validation Test and FAT

Test Item	**Design Aspect**	**System Validation Test** **(First Application)**	**FAT** **(Nth Application)**
Displays	Display Navigation	All designed displays loaded and accessible through various navigation means.	All designed displays are loaded.
	Signal Value Display	Each display shows values correctly formatted over signal range including display of abnormal conditions; trend functions demonstrated.	Single value for representative sample of signals is displayed (background displayed and foreground display elements operating as expected).
	Soft Operator Controls	All soft controls demonstrated to be effective, including operator dialog sequences, and test sequences.	Sampling of soft controls for plant operations (not maintenance) demonstrated to be effective per display.
Diagnostics	System Health Diagnostics	Abnormal conditions simulated to demonstrate correct operation of status signals and alarms.	No unexpected off-normal conditions created (health displays used to confirm normal system status).
	Error Handling	Random hardware failures; for example, single sensor, single power supply. Errors shall be handled with known consequences.	Hardware operability such as sensor input checks.
Performance	FPGA Functionality of Other Functions	Comprehensive logic and functional algorithm testing at the system level; testing shows connection of each input and output signal to function algorithm.	Tested only as it relates to operability of the hardware. This testing to be determined by V&V organization based on the need for the test to demonstrate variability that is possible from the assembly or manufacturing of the hardware. Examples may include hardware interlocks, hardware setpoints that have software interfaces, or functionality that is dependent on hardware configuration.

Table 9.6-1: Comparison of System Validation Test and FAT

Test Item	**Design Aspect**	**System Validation Test** **(First Application)**	**FAT** **(Nth Application)**
	Signal Redundancy	Shared inputs produce the same output from redundant processing elements in the system.	Sampling of the redundancy to the extent that indicates that the redundancy is effective in selection.
	I/O Redundancy	Testing shows connection of each input and output signal to function algorithm.	Confirmed as part of safety functional and response time tests and in the hardware tests in combination of V&V testing of software and system.

9.6.2 Site Acceptance Test

The SAT is a two-part test verifying correct functionality and performance after the system is installed at the customer's site. The site test personnel shall define and control the test. The primary intent of this test shall be to validate that the equipment was not damaged during shipment or installation. External system interface testing shall be specified in the SAT procedure.

9.7 REUSABLE LOGIC ELEMENT TESTING

Logic elements or pieces of code may be designed and built into a module that can easily be implemented and used in creating an FPGA image. They are then reused again across different ALS boards without changing the functionality of the RLET or its inputs or outputs. Typically, a library of these elements is kept. These RLETs can be officially entered into the library and freely used multiple times once they have been through the IV&V and testing process. The test involves ensuring the functional logic of a RLET meets its requirements and specifications. See Section 6 for a general description of the types of requirements and specifications are in an RLET (e.g., truth tables, interfaces, and functional logic). A one-time-type test will target just the RLET but might have to be set into a larger test environment such as the simulation discussed in subsection 9.5.1, or it may be additionally installed on hardware and tested (i.e., FPGA image is installed on a ALS card). On each use of an RLET in a design, testing is performed to ensure the RLET is used properly and interfaces (inputs and outputs) of the RLETS are connected and operational. Testing of the functionality does not need to be repeated since that was already completed.

9.8 ITEM PASS/FAIL CRITERIA

Each test technique's test specification defines pass/fail criteria for their respective testing. The comparison of test data to the pass/fail criteria will be documented in test reports for each test case. Overall pass or fail results will be documented in test summary reports.

9.9 SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS

The following criteria apply to an ALS project. Future ALS projects may define project-specific information to determine criteria for test suspension and test resumption in the project-specific test plan.

9.9.1 Test Suspension

Test activities may be suspended, if one of the following conditions occurs:

- Quality assurance (QA) program breach
- A trend toward a series of failures on a specific test item type
- Environmental issues that call for testing to stop.

Testing may be suspended by management, project management, or the test team members depending on the reason for suspension, including:

- Management for unqualified participants
- PM for changes to customer contracts
- Test engineering for failing conditions

9.9.2 Test Resumption

When testing is resumed after test suspension, the following actions should occur:

- Rerun of tests that were included in or the reason for the test suspension
- Rerun of tests that are substantially related to the reason for the test suspension
- Rerun any test affected by any component that was reworked

9.10 REGRESSION TESTING

Safety system changes can occur for several reasons. For example, changes can be made at the customer's direction or because of problems discovered during testing. It is normal for hardware and software modifications to be required during the system test period. All changes shall be formally documented and controlled according to the project management plan and configuration management plan.

Any time a problem is found and corrected, or a change is made in the system, a regression analysis is performed and documented. Once it is determined which subsystems and elements are affected, a review of the appropriate test procedure shall be performed to determine the changes in testing.

Original tests are performed on deliverable or surrogate hardware, as defined in the safety system test procedures. The deliverable hardware may not be available once the original tests have been completed. In this case, regression testing on surrogate equipment is permitted to be performed. Surrogate equipment performance and interface loading must be equivalent to the deliverable equipment for the level of testing performed.

9.11 TEST DELIVERABLES

The purpose of this section is to define the purpose, format, and content of required test documentation. The test documentation as a whole shall fulfill the requirements of IEEE Standard 829 (Reference 28) and Regulatory Guide 1.170 (Reference 15).

Additional details on test deliverables will be included in the project-specific test plan.

9.11.1 Test Plan

While this section serves as a generic outline for the test plan, the project-specific test plan documents the scope, approach, resources, and schedule for the testing activities of the project. It identifies the test items, the method for identifying the specific requirements to be tested, the testing tasks, and the required resources to perform these tasks.

9.11.2 Test Procedure Documentation

The following test procedure documentation may be separate documents or combined into a single document. These test procedure documents shall comply with the requirements of IEEE Standard 829, Section 7 (Reference 28).

9.11.2.1 Test-Design Specification

This portion of the test procedure specifies the details of the test approach for a software requirement or combination of requirements and identifies the associated tests.

9.11.2.2 Test-Case Specification

This portion of the test procedure specifies the inputs, predicted results, and a set of conditions for executing the test case.

9.11.2.3 Test-Procedure Specification

This portion of the test procedure specifies a sequence of actions for the execution of a test.

9.11.3 Test Anomaly Reports

All test anomalies will be reported and tracked in an anomaly reporting system.

9.11.4 Test Report

The test report summarizes the testing activities and documents the results. It also contains an evaluation of the corresponding test items. Typically, the test procedure document containing the handwritten entries by the tester becomes a part of the document.

The test report also contains an anomaly report log. Together, these identify the status of outstanding test exceptions reported during testing. The test reports shall comply with the requirements of IEEE Standard 829, Section 11 (Reference 28).

9.12 TESTING TASKS

9.12.1 Test Development

As part of the test planning, ALS requirements and design documents will be reviewed to determine the scope of testing. Test Phase RTMs identify each requirement’s test coverage.

For each test technique, the test cases and test procedure used to test the requirement are included in the RTMs.

9.12.2 Test Tool Development

Custom test tools were developed as part of the ALS platform. The test tool updates will follow the same process as that of the original platform:

1. Update test tool design specifications
2. Update the custom test tools
3. Validate that the test tools are fit for their purpose

Test tool development falls under the general-purpose software requirements defined in Section 1.2.

9.12.2.1 Test Tool Documentation

Custom test tools have design specification documents that describe the design and operation.

9.12.2.2 Tool Validation

All test tools developed for the verification and validation of the test items will be validated to ensure they can adequately test the test items. The validation of the test tools will be based on the requirement and design specifications for the test tools.

The tool validation reports will document any tool limitations.

9.12.3 Test Execution

Once developed, tests will be executed according to procedures developed for each test technique.

9.12.4 Burn-In

Burn-in will be completed before ABTS board-level testing and board hardware circuit testing. This process allows precision analog components to stabilize before performing further tests.

9.12.5 Test Reporting

At the completion of test execution, the results of all testing will be documented in test reports. Test reports will contain the test data, or reference the storage location for the test data, the results of the comparisons of the test data to the pass/fail criteria, and final analysis of the test results to determine if the test item has been built to and functions according to the requirements defined by the ALS requirements specifications and the test item's respective design specifications. Anomalies will be tracked and included only as needed for further informational purposes in related test summary reports.

9.13 ENVIRONMENTAL NEEDS

9.13.1 Test Environment

The project-specific test plan will define the test environment applicable for that ALS project.

9.13.2 ALS Platform Test Tools

ALS project testing requires custom tools as discussed in subsection 9.12.2.

The project-specific test plan will define the test tools applicable for that ALS project.

9.14 RESPONSIBILITIES

See Section 2 of this document for a discussion of the organization and responsibilities.

9.15 STAFFING AND TRAINING NEEDS

9.15.1 Staffing

Staffing needs are addressed as described in the project management plan.

9.15.2 Training

Training and qualification follow the Westinghouse QMS (Reference 4). Project-specific training needs are maintained by the PM, as described the project management plan. Function-specific training is reviewed periodically by the appropriate resource manager for updates and reviews of required training in accordance with the Westinghouse QMS (Reference 4).

9.16 SCHEDULE

Test resources provide schedule input, in terms of tasks and durations, to the PM, who maintains the master schedule as described in the project management plan.

9.17 RISK AND CONTINGENCIES

ALS platform test risks and associated contingencies will be identified and managed using the risk management plan defined in the project management plan.

9.18 APPROVALS

See this document’s cover page for plan approval.

The project specific test plan’s cover page will specify the approvals.

(Last Page of Section 9)

10 SOFTWARE INSTALLATION PLAN

10.1 PURPOSE

The following procedure is only performed at the factory and not at the plant site.

The following procedure describes how to program and configure an ALS printed circuit board assembly with the following information:

- FPGA binary image
- NVM memory image

10.2 SCOPE

This procedure is used to load FPGA and NVM images onto ALS boards.

10.3 EQUIPMENT LIST

The following equipment is required for the programming environment:

- Secure development environment (SDE) workstation capable of running all required software
- Software version control system
- FPGA programming software
- ALS test backplane
- ALS test and calibrate tool (ATCT)
- Power supply

10.4 FPGA AND NVM PROGRAMMING PROCESS

This section contains the instructions for loading the FPGA and NVM files onto ALS boards.

10.4.1 ALS Board Power

The ALS board is connected to the ALS test backplane and powered on as indicated by an illuminated green power LED on the ALS board.

10.4.2 Board FPGA Flashing

The SDE workstation and the FPGA programming software are used to program the ALS boards. To begin the programming process, the FPGA programming software will erase the FPGA configuration on the ALS board and then load the new FPGA configuration. The new FPGA configuration is then verified by comparing the checksum from the programming software and the software release record.

10.4.3 NVM Programming

This section describes how to program and configure the NVM for an ALS board. The following instructions assume that the ALS board is already plugged into the ALS extender board and powered per subsection 10.4.1.

The SDE workstation and the ATCT are used to program the ALS NVM. Refer to Section 3.6 of WCAP-18762 (Reference 3) for a description of the ATCT. Using the programming interface of the ATCT, the appropriate NVM files are located from the software version control system and then are written to the board NVM. CRC checksums are used to verify the correct NVM files are loaded. The test and configuration interface of the ATCT is then used to confirm configuration data, the NVM part number, and revision.

10.4.4 ALS Board Programming Completion

The board is disconnected from the ALS board test backplane to complete the programming process.

(Last Page of Section 10)

11 INTEGRATION PLAN

This integration plan addresses the integration of the software (FPGA) and the hardware. Integration consists of three major phases. First, various software modules are integrated into a single FPGA design. This process is discussed in Sections 4 and 5 of this document. The output of this task is a completed FPGA image.

The second phase integrates the FPGA image into the hardware by programming the FPGA. Programming the FPGA onto the target hardware is covered by the software installation plan in Section 10.

The third phase requires testing of the integrated product as defined by the test plan. The test plan is discussed in Section 9.

(Last Page of Section 11)

12 MAINTENANCE PLAN

12.1 INTRODUCTION

12.1.1 Purpose

Software maintenance is the process of correcting faults in the software product that led to failures during operation.

The maintenance phase includes all post-delivery product support. If problems are identified with the design or individual hardware components during operation, the problems are evaluated and corrected as needed. Relative to software, the maintenance plan describes three primary activities: reporting of failures that were detected during operation, correction of the faults that caused those failures, and release of new versions of the software product.

12.1.2 Scope

This plan applies to the maintenance of the ALS platform and applications. As described in NUREG/CR-6101 (Reference 24): “There is a related activity, sometimes termed “enhancement,” which is the process of adding functionality to a software product. That is not considered here. Enhancement of a reactor protection system should repeat all of the development steps described in this report.”

12.2 FAILURE REPORTING

12.2.1 Failure Detection

Since the ALS system is operated at a customer’s nuclear facility, it is the responsibility of the customer to detect and document any system failure in accordance with their plant procedures.

12.2.2 Failure Reporting

Since the ALS system is operated at a customer’s nuclear facility, it is the responsibility of the customer to notify Westinghouse of any system failure in accordance with their plant procedures.

12.2.3 Failure Tracking

Errors found externally (i.e., by a customer) may be reported to Westinghouse in any form. All external errors shall be documented and evaluated in Westinghouse’s Corrective Action Program. When the error impacts protection and/or important-to-safety class software or protection-system designs using the software, then the user is responsible for documenting appropriate action as necessary, including 10 CFR Part 21 evaluations. Technical bulletins are issued as required by the Westinghouse QMS (Reference 4).

In addition, all FPGA errors are documented in the anomaly reporting system as described in subsection 7.6.4, and all hardware errors are documented in the nonconformance system as described in subsection 7.6.4.

12.3 FAULT CORRECTION

The design team will analyze the system error to determine the cause and will take any necessary actions to correct the error. Depending on the source of the error, the starting life cycle phase will be determined. For example, if the source of the error is a faulty requirement, then regression activities will begin at the requirement phase. All impacted activities in subsequent phases, including the requirements phase, are then updated as described in this document, including IV&V activities. The FPGA is rereleased in accordance with the requirements of the CM plan. Similarly, this process would be applied if the source of the error is in the design or implementation phase.

(Last Page of Section 12)

13 CONFIGURATION MANAGEMENT PLAN

The ALS configuration management plan is defined in 6002-00002-P, “ALS Configuration Management Plan” (Reference 21).

(Last Page of Section 13)

14 SECURITY PLAN

The ALS Security Plan is defined in 6002-00006, “ALS Security Plan” (Reference 22).

(Last Page of Section 14)

15 PROJECT SPECIFIC PLANNING DOCUMENTS

The following planning documents are created for each ALS project:

- Project management plan
- Software V&V plan
- Test plan
- Software [customer] training plan
- Software operations plan
- Requirements management plan
- Equipment qualification plan

15.1 PROJECT MANAGEMENT PLAN

The PM initiates the PMP during the Concept/Planning Phase, typically after a contract is placed for an application or after approval for the development of a platform. The PMP includes the content necessary to satisfy the intent of a SMP. Westinghouse QMS (Reference 4) provides requirements for the PMP; however, the regulatory criteria shall take precedence in the event of a conflict.

The PMP shall include sections on purpose, organization and project team, oversight, responsibilities, security, milestones, metrics, project life cycle, training, planning documents, project file repositories, management reviews, risk management, project management procedures and tools, estimations and assumptions, budget, and other items as contractually required. The PMP shall include a high-level schedule and milestone list with life cycle phases identified.

The project life cycle shall be consistent with that defined in this process unless justification is provided in the PMP. The PMP shall define the tasks that are a part of each life cycle phase. The PMP shall define life cycle phase inputs and outputs, including review, verification, and validation of those outputs. Any other deviations to this document shall be described and justified in the PMP.

The PMP shall define any international, national, industry, and company standards and guidelines, including regulatory guides that are different from those defined in this process.

The team training defined in the PMP shall be in accordance with the Westinghouse QMS (Reference 4).

The PMP shall also define the set of plans to be produced or standard ALS plans to be utilized for the project including those identified in this process. The plans can be contained in individual documents or multiple plans can be contained in a single document.

The project file repository for FPGA and NVM design files shall be defined in the PMP. The PMP shall define any additional repositories necessary for execution of the project.

Risk management shall be accomplished using the risk analysis technique in the Westinghouse QMS (Reference 4). Tools and techniques for a given project shall be defined in the PMP. The project team shall update the risk management tool through the project's life cycle and take concrete steps to reduce the probability and/or impact of each risk until safely retired. FOAK risks shall be considered during the risk analysis.

The PMP shall specify tools and methods for tracking and reporting issues and anomalies.

The PMP shall include any contractually specified standards, methodologies and tools for configuration management, quality assurance, verification and validation, training, documentation, design, development, and tests. The output of this task shall be consistent with the regulatory basis defined in this process unless overarching customer requirements exist. Any deviations or exceptions to the regulatory basis defined in this process shall be explicitly justified in a project specific document.

The PMP references a project quality plan, which identifies Westinghouse QMS (Reference 4) applicable procedures with clarifications, and any alternatives and justifications.

Generally, any of the above items may either be included in the body of the management plan or by referring to supporting documents.

15.2 SOFTWARE V&V PLAN

An SVVP is created for each project following the outline in WNA-PV-00129-GEN (Reference 5).

15.3 TEST PLAN

A test plan is created for each project following the outline in Section 9.

15.4 SOFTWARE TRAINING PLAN

If in the project scope of supply, a training plan will be created that describes the procedures used to train the operators of the system. In this case, reactor operators will need to be trained in use of the protection system software. The training plan should also cover training for managers and maintenance personnel.

The training plan only applies to the end user of the system and therefore is not applicable for the platform development team. Platform engineers are trained and qualified in accordance with the Westinghouse QMS (Reference 4) and the PMP.

15.5 SOFTWARE OPERATIONS PLAN

The operations plan shall address the effort necessary to ensure that the appropriate plant personnel are able to operate the software. This is typically included in the system's technical manual.

The operations plan is not applicable to platform. An operation plan is applicable to plant personnel operating the software in system. An operation plan relative to the platform development team in the scope of this plan.

15.6 REQUIREMENTS MANAGEMENT PLAN

The detailed requirements management (RM) plan for the project shall be developed as required by the Westinghouse QMS (Reference 4). The RM plan shall describe how the requirements management process applies to the ALS project, identify source documents, how to document objective evidence the requirement was fulfilled, and any RM tools that will be used. The RM plan can be included in the PMP at the discretion of the PM.

15.6.1 Requirements Traceability Matrix

Associating requirements with the documentation and software that satisfy them creates the RTM. The system is verified to show that all applicable requirements have been met. A unique number should identify each requirement. The association between requirements, design, code, and tests can be made using document and section references, test identification numbers, software code identification numbers, etc.

The RTM can be either a table of information prepared manually or a report generated from a requirements database. It is recommended that the RTM be kept in a database format for ease of updating. The RTM shall be a living document to be used throughout each phase of the design life cycle process. After each life cycle phase, the design team shall complete the RTM for that phase to verify that all requirements have been properly addressed in that phase. In other words, the design team shall confirm that all lower-level requirements and design features are derived from higher-level requirements, and that all higher-level requirements are allocated to lower-level requirements and design features.

Traceability analysis verifies completeness, that all lower-level requirements and design features are derived from higher-level requirements, and that all higher-level requirements are allocated to lower-level requirements, design features, and tests. Traceability analysis is also used in managing change and provides the basis for test planning.

The traceability analysis also provides a method to cross-reference each software requirement against all the documents and other software items in which it is addressed. Requirements entered in the matrix are organized into successive lower-level requirements, as described in each document. The purpose of this analysis is to verify that the design team addresses every requirement throughout the design life cycle process. The life cycle phases that shall be analyzed are requirements, design, implementation, testing, and installation/checkout.

15.7 EQUIPMENT QUALIFICATION PLAN (or METHODOLOGY)

The EQ plan or methodology document contains the overall guidelines for equipment qualification criteria, qualification methods, applicable codes and standards, plant environmental parameters, plant seismic parameters, electro-magnetic compatibility (EMC) requirements, and the general qualification requirements for the project.

(Last Page of Section 15)

16 ALS TOPICAL REPORT CHANGE PROCESS

Section 6 of WCAP-18762 (Reference 3) describes the ALS topical report change process. Changes to supporting documents to this document will be reviewed in accordance with the ALS topical report change process.

Any alternatives to this process will be documented in the project management plan. See Section 15.1.

(Last Page of Section 16)