V&V Plan for FPGA-based ESF-CCS Using System Engineering Approach.

Restu Maerani, Joyce Mayaka, Mohamed El Akrat, Jung Jae Cheon
Department of Nuclear Power Plant Engineering, KINGS Instrumentation and Control Lab, KEPCO International Nuclear Graduate School, Ulsan, 45014 Republic of Korea.

maerani@email.kings.co.kr

Abstract. Instrumentation and Control (I&C) systems play an important role in maintaining the safety of Nuclear Power Plant (NPP) operation. However, most current I&C safety systems are based on Programmable Logic Controller (PLC) hardware, which is difficult to verify and validate, and is susceptible to software common cause failure. Therefore, a plan for the replacement of the PLC-based safety systems, such as the Engineered Safety Feature – Component Control System (ESF-CCS), with Field Programmable Gate Arrays (FPGA) is needed. By using a systems engineering approach, which ensures traceability in every phase of the life cycle, from system requirements, design implementation to verification and validation, the system development is guaranteed to be in line with the regulatory requirements. The Verification process will ensure that the customer and stakeholder's needs are satisfied in a high quality, trustworthy, cost efficient and schedule compliant manner throughout a system's entire life cycle. The benefit of the V&V plan is to ensure that the FPGA based ESF-CCS is correctly built, and to ensure that the measurement of performance indicators has positive feedback that “do we do the right thing” during the re-engineering process of the FPGA based ESF-CCS.

Keywords: ESF-CCS, FPGA, Verification, Validation

1. Introduction
Advanced Power Reactor 1400 (APR1400) is equipped with multi-loop controllers based on microprocessors for the safety and non-safety control systems. Principles such as redundancy, diversity, and segmentation have been incorporated in order to achieve both the desired availability and reliability of these systems [1]. The purpose of the Verification and Validation (V&V) plan is to meet the reliability requirements for components for the Instrumentation and Control (I&C) systems in the Nuclear Power Plant (NPP) [2]. Programmable Logic Controllers (PLCs) have previously been used to implement real-time controllers in the Reactor Protection Systems (RPS). However, the increase in the complexity of I&C systems as well as high maintenance cost, require a more reliable application development platform that still meets the measures of effectiveness (MOE’s) of the I&C system. One of the technologies currently being looked at is the Field-Programmable Gate Array (FPGA) [3].

The design and implementation of an FPGA-based ESF-CCS must pass through several verification stages. Because of the FPGA's are dual nature, being both hardware and software, and its inherent complexity, we need a specific regulatory document for FPGA that would address issues such
as system safety assessment, design life cycle, verification and validation (V&V) and related documentation requirements for NPP safety system[4]. Since the specific standard of FPGA is not found on United States Nuclear Regulatory Commission (USNRC) so we can use the alternative document from International Electrotechnical Commission IEC 62566:2012 about “Nuclear Power Plants – Instrumentation and control important to safety- Development of HDL-programmed integrated circuits for systems performing category A functions”. While carrying out reverse engineering, V&V activities are required to ensure that the system is redesigned in accordance with the original design basis requirements and system design requirements [5]. Then for the re-engineering process the verification process will be conducted after completing the design and implementation process; which includes system, hardware, and software verification. Based on the review of IEEE Std. 15288-2008 [6], R Halligan [7] said that “a verification system is a system like any other system, but is also a subsystem of the bigger system, of which verification is a part of the solution”.

V&V is a means by which the product is checked, and by which its performance is demonstrated and assured to be a correct interpretation of the requirements. The V&V processes provides assurance that the specified functional and reliability requirements of the system are met [8]. V&V process can prove the success of a design, but cannot prove that the model is true and accurate for all possible conditions and applications. It can only provide evidence that the model is accurate enough and its output can be utilized. Therefore, the V&V process is complete when sufficiency is achieved [9].

To make certain that the V&V process is carried out correctly, it needs to be implemented using System Engineering approach to ensure the realization of a successful system. The problem statement should begin by determining the top level function, which the system will do, as well as explaining the mission and concepts to be performed. In addition, the possibility of deficiencies or barriers should be defined, as well as how to overcome them and trace the requirements document. Explain how the process of implementation and design simulation. All these formulations must also be in accordance with the client's request as they have the ease of operation in the future [10].

The use of life cycle model is also recommended by the IAEA in “IAEA safety standards on Design of Instrumentation and Control Systems for Nuclear Power Plants” [5]. For safety critical I&C systems, it is very common to use the life cycle model for system that related to electronic system and instrumentation. In NPPs this practice has been applied for the documentation of the development process of I&C system to facilitate the process of verification of each activity and review of the final product by validation process, to ensure that the final product is suitable for use. It is important to control a process with the necessary management systems to achieve the objectives, provide the means to meet all requirements and provide the organization product to be identified, and its development should be planned, implemented, assessed and continuously improved.

This paper outlines how to verify and validate an FPGA-based ESF-CCS system for the APR1400. The paper does not discuss the overall development process of an FPGA-based ESF-CCS or give detailed information about ESF and FPGA, neither does it discuss the results of the V&V activities. These shall be developed in later work.

2. Definition

FPGA based EFS- CCS Operation

The digital Instrumentation and Control System (I&C) technology of nuclear power plants, is expected to have the ability to meet the safety criteria of nuclear reactors in order to avoid three problems that can occur: software common-cause failure, failure interaction between operator and Man Machine Interface System (MMIS) and the non-detectability of software failure [11].

In the recent past, FPGA technology has gained attention in the nuclear industry. A field-programmable gate array (FPGA) is an integrated circuit designed to be configured by a customer or a designer after manufacturing – hence "field-programmable". FPGAs contain an array of programmable logic blocks, and a hierarchy of reconfigurable interconnects that allow the blocks to be "wired together", like many logic gates that can be inter-wired in different configurations.
The ESF-CCS is composed of the electrical and mechanical devices and circuitry, from sensors to actuation device input terminals that are involved in generating those signals, which actuate the required ESF system components. This system consists of four independent channels (Channels A, B, C, and D) which are electrically and physically independent. The ESF-CCS configuration is currently based on a PLC platform. The ESF-CCS performs the functions of actuating Engineered Safety Features Actuation System (ESFAS) and executing component control by interfacing with ESFAS portion of Plant Protection System (PPS) as shown in Figure 1.

![Figure 1: EFS-CCS Operation](image)

**Design Life Cycle**

Based on the IAEA recommendations that I&C all the activities associated with the development, implementation and operation of the overall I&C architecture, individual I&C systems and I&C components should be carried out in the framework of a documented development life cycle [5]. Because this design is related to hardware for the FPGA Board also VHDL code as a software tools to generated signal for simulation of ESF CCS design, we also need to refer to IEEE 1074-2006: Standard for Developing a Software Project Life Cycle Process [12]. There are many life cycle models to use for the system engineering process of developing large-scale complex systems. One famous model is the V-Model [13]. What follows is a brief description of the lifecycle detailed in EPRI 1019181, which is also referenced by IAEA [14] and is consistent with the one suggested by IEC 62566 [15]. The V-model lifestyle is depicted in Figure 2.

![Figure 2: V Model for FPGA](image)
Verification and Validation (V&V)
Verification and validation is a process of checking the results of a design implementation to ensure the reliability of its performance and has been ensured in accordance with applicable requirements standards document [8]. The benefit of the V&V process here is to ensure the reliability of the I&C system developed using FPGA, appropriate design steps may require reiterating the design until the verification requirements are met. And verification steps can also had revision of the initial requirements so it need to update all the previous required design steps [4].

3. Methodology
NUREG /CR-7006 [4] recommends the life cycle illustrated in Figure 3. Based on this recommendation, the Y model in Figure 4 was developed with the aim of minimizing the system complexity.

![Figure 3: FPGA Design Flow](image)

![Figure 4: Modification of Development Process from V Model to Y Model](image)

During development process for the FPGA-based ESF-CCS design, there are several activities that should be verified, referenced based on the design flow in Figure 3. The following tasks should be performed:

- Analyze the requirements document for the ESF CCS by referring:
  - 10 CRF 50
  - Design Document
  - Korea Utility Requirements Document
  - Standards
  - Regulatory Guides
  - Branch Technical Position
  - Standard Review Plant
- Develop Design Architecture of ESF-CCF based FPGA using Basys 3 board
• Create the formal modelling for the operating scenario of ESF CCS, make the code for generate the signal to FPGA board.
• Develop test case to verify the VHDL code
• Develop coverage test to verify the successful of the system
• Validate the design by reporting the technical performance.

4. Discussion

4.1 Requirement Analysis

This is a description of the system to be implemented. It should lay out the required functionality, interfaces and operational environment of the system. Additional requirements, such as critical timing, power, and board size, may be added to the requirements [4].

When we start to design a system, in this case I&C system, we must first submit the design that related to design requirement from the regulator to be verified. The regulator will determine whether the I&C system will meet the requirements of the design requirement from the regulator to be verified. The regulator will determine whether the I&C system will meet the requirements of standard regulation document [17].

I&C systems are described in Chapter 7 of the Standard Review Plant (SRP), specifically section 7.3 deals with the Engineered Safety Features Actuation System (ESFAS). In the SRP, reference is found to the Regulatory Guides (RG) which must be implemented in the design for the licensee to be in compliance with the established law of 10 Code Federal Regulations (CFR). These RGs in turn, reference standards which give a more detailed guide on the implementation of a given RG. The hierarchy of the requirements document is as shown in Figure 5.

![Figure 5: Requirements Hierarchy and Inter-relationship for ESF-CCS](image)
4.2 Formal Modelling

Elisabeth Hull [18] mentioned that “Those methods that use diagrammatic representations are usually referred to as ‘structural methods’, those that use object-orientation are referred to as ‘object-oriented method’ and those that use mathematics are referred to as ‘formal methods’. The benefit of using formal modelling for I&C system because the formal method relies on the use of unambiguous formalisms for specifying systems. With that methods we can develop a larger system with complex software systems for re-engineering program because formal modelling are easy to understand [16]. Therefore, the correct checking can be done perfectly to avoid human error, because formal modelling describes in detail about design requirements [18]. During the implementation process, the requirements allocated to the system elements are used to design, fabricate, code, or build each individual element using specified materials, processes, physical or logical arrangements, standards, technologies, and/or information flows outlined in detailed drawings or other design documentation [19].

4.3 Testing and Coverage Test

FPGA verification is performed by giving input which is will be simulated based on software. According to FPGA Basys3 board hardware features it has limited digital inputs/output ports (24 Input/Output/digital ports). According to system design there are more than 25 Input/output signals. Every VHDL implementation goes through extensive verification. In this case VHDL code will be verified using Xilinx Vivado Software where the logic design is created to be simulated with FPGA Board.

In this process, the following steps should be done to verify the code:
1. Use generated VHDL to simulate the VHDL code for verification of all logic gates and hardware functions.
2. Run the simulation under all possible scenarios
3. Check the output waveform and compare it with the designed truth table
4. Verify the possibility to generate bit stream file that can be deployed on FPGA
5. Simulation.

Figure 6 shows the steps to verify the code.

Figure 6: VHDL Code Verification Flow Chart.
4.3.1 Develop Test Case for VHDL Code

Test cases for this system were developed, based on[20] as follows:

1. Requirements based test case selection,
2. Test coverage analysis.

The test cases were developed according to [8]. This guide outlines different tests that ensure 100 percent coverage of a gate or decision point. Figure 7 was the test case from VHDL code from Vivado which is will be use for generate coverage test in Aldec software for Modified Condition/Decision Coverage (MC/DC) test.

---COMPONENT PROCESSING MODULE TEST CASES---

-- Test Case TC01--

MCR_Start <= '1'; MCR_Enable<=0'; MCR_Conf<='1'; RSR_Transfer <='0';
wait for Clock_period*10;

MCR_Start <= '1'; MCR_Enable<=1'; MCR_Conf<=0'; RSR_Transfer <='0';
wait for Clock_period*10;

MCR_Start <= '1'; MCR_Enable<=1'; MCR_Conf<='1'; RSR_Transfer <='1';
wait for Clock_period*10;

MCR_Start <= '1'; MCR_Enable<=1'; MCR_Conf<='1'; RSR_Transfer <='0';
wait for Clock_period*10;

MCR_Start <= '0'; MCR_Enable<=1'; MCR_Conf<=1'; RSR_Transfer <='0';
wait for Clock_period*10;

--Reset signals

MCR_Start <= '0'; MCR_Enable<=0'; MCR_Conf<=0'; RSR_Transfer <='0';
wait for Clock_period*10;

---

Figure 7: Applied Test Case in Test Bench Vivado
Table 1: Sample of Test Case

| Test Case ID | TC01 |
|-------------|------|
| Test Case   | The objective is to observe if the component start command is generated only when there are valid conditions at the MCR. |
| Summary     | 2.1.2.1.2 |
| Related     | 2.1.2.1, 2.1.2.3, 2.1.2.5, 2.1.2.6 |
| Requirements| 2.1.2.1, 2.1.2.3, 2.1.2.5, 2.1.2.6 |
| Test Procedure| Set the inputs of to the defined states and observe the component output. |
| Test Data   | Test No. | MCR | ESCM | START | ESCM ENABLE | CONF. SWITCH | RSR | TRANSFER |
|-------------|----------|-----|------|--------|--------------|--------------|-----|----------|
|             | 1        | 1   | 0    | 1      | 0            |              |     |          |
|             | 2        | 1   | 1    | 0      | 0            |              |     |          |
|             | 3        | 1   | 1    | 1      | 1            | 1            |     |          |
|             | 4        | 1   | 1    | 1      | 0            |              |     |          |
|             | 5        | 0   | 1    | 1      | 0            |              |     |          |
| Expected Result | Test No. | Component Start |
|                | 1        | 0               |
|                | 2        | 0               |
|                | 3        | 0               |
|                | 4        | 1               |
|                | 5        | 0               |
| Actual Result | Test No. | Component Start |
|               | 1        | 0               |
|               | 2        | 0               |
|               | 3        | 0               |
|               | 4        | 1               |
|               | 5        | 0               |
| Status        | Passed   |
| Created By    | Joyce Mayaka |
| Executed By   | Mohamed Elakrat |
| Date          | 5/12/17  |
| Test Environment | Vivado   |

4.3.2 Coverage Test

Coverage is a measure and not a method of testing. Definition of the coverage required for testing gives a measure of the adequacy of the verification. There are two measures of coverage depicted in Figure 8, requirements coverage and software structure coverage. Requirements coverage analysis determines how well the requirements based testing verified the implementation of the software requirements, and establishes traceability between the software requirements and the test cases. Structural coverage analysis determines how much of the code structure was executed by the requirements-based tests and establishes traceability between the code structure and the test cases. For type of structural coverage metric used was the Modified Condition/Decision Coverage (MC/DC).
5. Validation
After completing all verification activities, the last activity is to validate the system. Validation checks compliance of the whole system with the initial system requirements specification. The FPGA function validation method is different from general software. We will measure the performance by define the Measures of Performance (MOP) and Technical Performance Measures (TPMs).

6. Conclusion
The V&V activity plan are includes verification at the system level, component level and software level of the FPGA-based ESF-CCS. In this V&V plan, the stages of the V&V process are structured by applying a systems engineering approach. It aims to produce a system that can be justified both from the design concept and the design implementation before it is applied significantly to ESF-CCS based FPGA.

The goal of this V&V plan is to verify the development and implementation for the stage of reverse engineering program which it applied to elicit the requirements of the system as well as gain an understanding of the current system architecture, and the re-engineering program with the aim of improving the performance of ESF-CCS.

The result should prove whether FPGA has been able to surpass the performance of PLC which is already applied in I&C system. The benefit of V&V activity that it allows for the early identification of problems, any design failure or system malfunction during the design process, so that errors and modifications can be made as soon as possible. The hope that from the good structure of V&V plan it will produce a capable design and ready to apply in the I&C system of NPP.

This research was supported by the 2017 Research Fund of the KEPCO International Nuclear Graduate School(KINGS), Republic of Korea.
7. References

[1] H. Kim, “The Design Characteristics of Advanced Power Reactor 1400,” vol. IAEA-CN-16.

[2] Y. Wu, X. Shui, Y. Cai, J. Zhou, Z. Wu, and J. Zheng, “Development, verification and validation of an FPGA-based core heat removal protection system for a PWR,” Nucl. Eng. Des., 2016.

[3] J. Yoo, J. H. Lee, and J. S. Lee, “A research on seamless platform change of reactor protection system from PLC to FPGA,” Nucl. Eng. Technol., vol. 45, no. 4, pp. 477–488, 2013.

[4] U.S.NRC, “NUREG/CR-7006, Review Guidelines for Field-Programmable Gate Arrays in Nuclear Power Plant Safety Systems,” 2009.

[5] IAEA, “Design of Instrumentation and Control Systems for Nuclear Power Plants,” in IAEA Safety Standards, no. SSG-39 Specific Safety Guide, 2016.

[6] IEEE, “Systems and software engineering - System Life Cycle Processes,” in IEEE Std 15288-2008, 2008.

[7] R. Halligan, “Application Guidance for : ISO / IEC 15288 ( IEEE Std 15288 2008 ) Second Edition 2008 02 01 Systems and software engineering – System Life Cycle Processes by Robert Halligan , FIE Aust,” vol. 15288, pp. 1–14, 2009.

[8] IAEA Technical Report Series 384, “Verification and Validation of Software Related to Nuclear Power Plant Instrumentation and Control,” 1999.

[9] B. Thacker, S. Doebling, F. Hemez, M. Anderson, J. Pepin, and A. Edward, “Concepts of Model Verification and Validation,” Natl. Nucl. Secur. Agency (NNSA), Los Alamos Natl. Laboratory, vol. LA-14167-M, no. October 2004, 2004.

[10] “What is Systems Engineering.” [Online]. Available: http://www.incose.org/AboutSE/WhatIsSE. [Accessed: 27-Jun-2017].

[11] H.-W. Huang, C. Shih, S. Yih, and M.-H. Chen, “Integrated software safety analysis method for digital I&C systems,” Ann. Nucl. Energy, vol. 35, no. 8, pp. 1471–1483, 2008.

[12] IEEE, IEEE 1074-2006: IEEE Standard for Developing a Software Project Life Cycle Process, vol. 2006, no. July. 2006.

[13] A. Sanders and J. Klein, “Systems engineering framework for integrated product and industrial design including trade study optimization,” Procedia Comput. Sci., vol. 8, pp. 413–419, 2012.

[14] IAEA, “Application of Field Programmable Gate Arrays in Instrumentation and Control Systems of Nuclear Power Plants,” Iaea Nucl. Energy Ser. Publ. NP-I-3.17, vol. 1995–7807, no. Viena, 2016.

[15] EPRI, “Guidelines on the Use of Field Programmable Gate Arrays (FPGAs) in Nuclear Power Plant I&C Systems,” 2009.

[16] J. Jung, H.-S. Chang, and H.-B. Kim, “‘3+3 process’ for safety critical software for I&C system in nuclear power plants,” Nucl. Eng. Technol., vol. 41, no. 1, pp. 1–8, 2009.

[17] S. Yih and C.-F. Fan, “Analyzing the decision making process of certifying digital control systems of nuclear power plants,” Nucl. Eng. Des., vol. 242, pp. 379–388, 2012.

[18] E. Hull, Requirements Engineering, vol. 41, no. 4. 2009.

[19] IEEE, “System and Software Engineering-System Life Cycle Process,” in IEEE Std 15288-2008, vol. Second edi, 2012.

[20] IEEE, “Software life cycle processes,” in IEEE Std 12207-2008, 2008, p. Software & Systems Engineering Standards Committe.

[21] K. J. Hayhurst, D. S. Veerhusen, J. J. Chilenski, and L. K. Rierson, “A Practical Tutorial Decision Coverage on Modified Condition,” 2001.