Journal of KOSSE. (2018. 12) Vol. 14, No. 2 pp. 41-48
DOI: https://doi.org/10.14248/JKOSSE.2018.14.2.041
www.kosse.or.kr

ISSN (print) : 1738-480X
ISSN (online) : 2288-3592

Development Process of FPGA-based Departure from Nucleate Boiling Ratio Algorithm Using Systems Engineering Approach

In Sok Hwang, Jae Cheon Jung*

Department of NPP Engineering, KEPCO International Nuclear Graduate School

Abstract : This paper describes the systems engineering development process for the Departure from Nucleate Boiling Ratio (DNBR) algorithm using FPGA. Current Core Protection Calculator System (CPCS) requirement and DNBR logic are analyzed in the reverse engineering phase and the new FPGA based DNBR algorithm is designed in the re-engineering phase. FPGA based DNBR algorithm is developed by VHSIC Hardware Description Language (VHDL) in the implementation phase and VHDL DNBR software is verified in the software Verification & Validation phase. Test cases are developed to perform the software module test for VHDL software modules. The APR 1400 simulator is used to collect the inputs data in 100%, 75%, and 50% reactor power condition. Test input signals are injected to the software modules following test case tables and output signals are compared with the expected test value. Minimum DNBR value from developed DNBR algorithm is validated by KEPCO E&C CPCS development facility. This paper summarizes the process to develop the FPGA-based DNBR calculation algorithm using systems engineering approach.

Key Words : Core Protection Calculator System, Field Programmable Gate Array, Departure from Nucleate Boiling Ratio, VHSIC Hardware Description Language, systems engineering

Received: November 8, 2018 / **Revised**: December 14, 2018 / **Accepted**: December 27, 2018

* 교신저자 : Jae Cheon Jung, jcjung@kings.ac.kr

1. Introduction

It is widely accepted that the complexity of the development life cycle is increasing as technology evolves rapidly. The complexity of system development requires a high degree of understanding of needs, requirements, and makes manufacturing, operation, and maintenance processes more complex. This could increase risk and hinder innovation. The departure from nucleate boiling ratio (DNBR) algorithm in the Core Protection Calculator System (CPCS) is one of the most complex algorithms used in the nuclear power plant safety critical system. Applying systems engineering approach to DNBR algorithm development could increase the efficiency of life cycle and more thoroughly verify requirements.

The CPCS is to protect the reactor from exceeding the safety limit by monitoring reactor core conditions and to generate the DNBR trip signal and the local power density (LPD) trip signal during plant operation [1]. The DNBR is the ratio between the critical heat flux (CHF) and the maximum local heat flux. If the DNBR is lower than the safety limit, it would cause the formation of vapor layer, causing reduction in heat transfer capability between reactor coolant and fuel rods. The CPCSs are installed in Advanced Power Reactor 1400 (APR1400) such as Shin-Kori Nuclear Power Plants (NPPs) 3&4 and Barakah 1,2,3&4 NPPs in United Arab Emirates (UAE). These systems are developed based on the Programmable Logic Controller (PLC) platform.

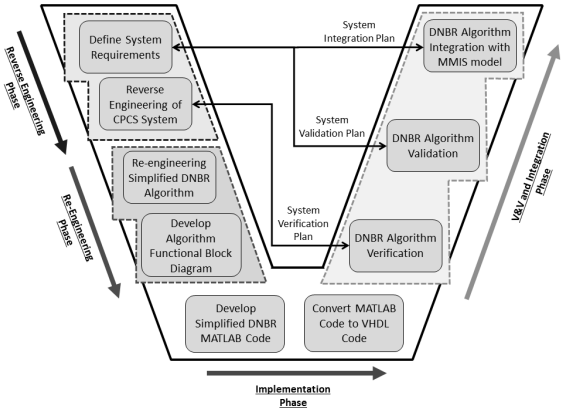
The Field Programmable Gate Array (FPGA) is Hardware Description Language (HDL) device, which are a type of large scale integrated circuit where the internal hardware architecture is configured for a specific application according to the user needs after production of the chip. FPGA based systems can be made simpler and easier to test and qualify for safety application comparing with PLC based systems. And FPGA supports longer lasting availability and faster response time, and reduces vulnerability to cybersecurity attacks. FPGAs are generally viewed by regulatory bodies as diverse from software based systems, and provide a higher level of diversity due to the different nature of their components [2].

The main objective of this research is to develop the FPGA-based DNBR algorithm using VHSIC Hardware Description Language (VHDL). An in-depth analysis of the current CPCS is performed to identify the whole CPCS algorithm, and new FPGA based DNBR algorithm is designed. 100%, 75%, and 50% reactor power data are obtained using KEPCO International Nuclear Graduate School (KINGS) APR1400 simulator and used to test the DNBR algorithm. Minimum DNBR value from developed DNBR algorithm is verified by KEPCO E&C CPCS development facility.

2. Methodology

2.1 Vee model for development process

The popular systems engineering Vee diagram provides a view of life cycle development with explicit relationships shown between requirements and systems definition and the developed and validated product [3]. The Vee model describes a software systems engineering process from software concept development to software integration and test. In order to design the



[Figure 1] Vee model for FPGA-based CPCS DNBR algorithm

development process of FPGA-based DNBR algorithm, the Vee model is designed as shown in figure 1. First, current Advanced Power Reactor 1400 (APR1400) CPCS requirements are identified to understand the system and CPCS specification and software requirement documents are analyzed in reverse engineering process. Current software algorithms are analyzed using CPCS simulator tool. This process corresponds to the software concept development and analysis, which decompose the current CPCS software to develop the new one. Second, new simplified DNBR algorithm is designed based on the current DNBR software in re-engineering process. System requirements of DNBR algorithm are defined and the simplified DNBR Enhanced Functional Flow Block Diagrams (EFFBDs) are developed using Core 9 software tool. This process corresponds to the software design process. Third, in implementation process, simplified DNBR MATLAB codes are developed based on the EFFBDs. The MATLAB software tool is used to develop and simulate each module of DNBR software algorithm. And MATLAB software codes are converted to the VHDL codes by HDL Coder in MATLAB. This process corresponds to the software coding process. Fourth, DNBR VHDL codes are verified by test cases and validated with system requirements in V&V and integration process.

2.2 Systems Engineering Management Plan

The Systems Engineering Management Plan (SEMP) is the top-level plan for managing the Systems Engineering (SE) effort. The creation of the SEMP involves defining the SE processes, functional analysis approaches, what trade studies will be included in the project, schedule, and organizational roles and responsibilities, to name a few of the more important aspects of the plan [4]. Early in the project, the SEMP should be provided to show how the project will be managed and handled.

In this project, the SEMP defines how the FPGA-based DNBR algorithm project will be organized, structured, conducted, and controlled. It covers purpose, scope, applicable documents and tools, organization structure, responsibilities, task deliverable, schedule, and work breakdown structure (WBS). Through the project, WBS progress is updated and the SEMP is revised. The WBS is shown in Figure 2.

2.3 Reverse Engineering Process

Software reverse engineering is the process of analyzing a software system to identify the systems components and their inter relationships and create representations of the system in another form or at a higher level of abstraction [5].

Current APR1400 CPCS requirements are analyzed and listed in reverse engineering process. The CPCS software consists of six interdependent programs: Coolant Mass Flow

WBS STRUCTURE		Progress 0-100%
Preparation and Problem Definitions		
	Design of SU6	
	WBS	
	Creation of SEMP	
Develop Simplified DNBR VHDL Code		
	Design and Specialty Engineering	
	Develop MATLAB file	
	Simulate MATLAB file	
	Convert to VHDL Code	
	Management Engineering	
	Revise SEMP/WBS	
Software Requirements Analysis / Test cases		
	Software requirement analysis	
	Functional requirement	
	Develop test cases	
	Get input data from KINGS APR1400 Simulator	
	Develop software module test cases	
	Get DNBR and intermediate values - KEPCO E&C CPCS development facility	
Software testing		
	Software Module #1 Test	
	Software Module #2 Test	
	Software Module #3 Test	
	Software Module #4 Test	
	Software Module #5 Test	
	Software Module #6 Test	
	Software Module #7 Test	
	Modify MATLAB Code	
	Modify VHDL Code	
	Upate WBS / Write activity report	

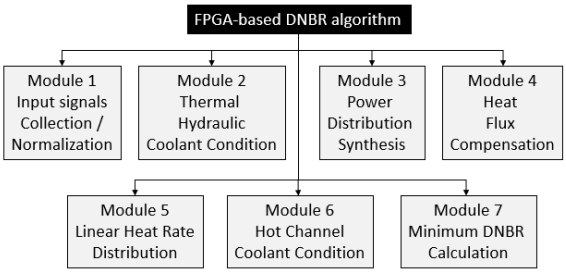
[Figure 2] WBS Structure

Program (FLOW), DNBR and Power Density Update Program (UPDATE), Power Distribution Program (POWER), Static DNBR and Power Density Program (STATIC), Trip Sequence Program (TRIPSEQ), and CEAC Penalty Factor Program [6]. Requirements of each program are used to understand the function, data flow, input, local, and output signals. In order to analyze the software algorithm of CPCS, the Sim Core Simulator is used, which is developed by KHNP to simulate the APR1400 CPCS software.

2.4 Re-Engineering Process

Software re-engineering is the examination and alteration of a subject system to reconstitute it in a new form and the subsequent implementation of the new form [5].

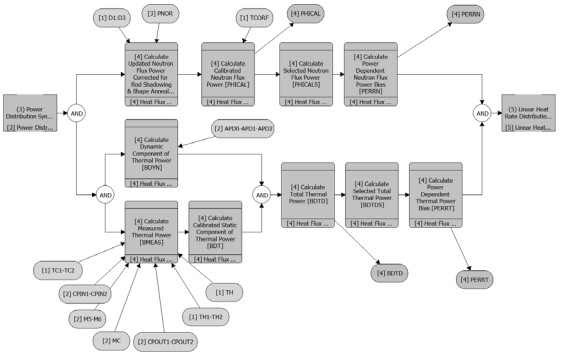
The new simplified DNBR algorithm is designed in re-engineering process. Based on the analysis of current APR1400 CPCS software,



[Figure 3] Functional architecture

new requirements for simplified DNBR algorithm are defined. System requirements are necessary to design, implement, and verify the new DNBR algorithm. High LPD, CEA Withdrawal Prohibit (CWP), Asymmetric Steam Generator Transient (ASGT) trip, and Variable Overpower Trip (VOPT) are not considered in this paper.

Figure 3 shows the functional architecture of simplifid DNBR algorithm which consists of seven modules. The Module 1 receives all input signals and normalizes them to use in this algorithm. Pressurizer pressure, reactor coolant pump speed, ex-core detector, cold leg temperature, hot leg temperature, and CEA positions are transmitted from instrumentation to the module 1. The normalized input signals are used for all modules. The Module 2 calculates thermal hydraulic coolant condition. It generates enthalpy-temperature ratio of water feeding steam generator 1 & 2, pumps legs 1 & 2, and normalized average cold leg specific volume. The module 3 receives CEA position and normalized ex-core detector power and generates relative power in axial nodes of pseudo hot pin and average of hot pin power distribution. The module 4 performs heat flux compensation and calculates maximum compensated core average power (FCALC1). The module 5 calculates hot pin heat flux distribution



[Figure 4] EFFBD of Module 4

and linear heat rate distribution. The module 6 receives hot pin heat flux distribution and linear heat rate distribution and calculates corrected mass flux for hot channel and enthalpy for hot channel. Final module 7 calculates all DNBR values and selects minimum DNBR value.

In re-engineering process, EFFBDs are developed for seven modules of new DNBR algorithm using Core 9 software. It represents the flow of control through sequencing of functions and constructs as well as the data interactions overlaid to present a more complete picture [7]. Figure 4 shows an EFFBD of Module 4 "Heat flux compensation" which describes input signals, output signals, functions, and flow controls.

2.5 Implementation Processs

Based on the functional block diagram and modeling diagram which are generated in re-engineering process, all seven modules are developed using Matlab software tool. Input and output signals are defined in codes and function is developed in each module. Output signals of each module are verified by comparing with Sim Core simulator data.

After developing matlab software codes, these matlab codes are converted to the VHDL using HDL coder of matlab software. Test bench is needed to convert from matlab code to VHDL code. These VHDL codes can be implemented in FPGA board.

2.6 V&V and Integration Process

This process is to verify and validate the implementation of CPCS DNBR algorithm based FPGA for the software level as the safety critical software for this system, to specify systematic approach to be used as a tools for the software engineering process of digital computer based I&C system. Software verification and validation is recommended by IEEE 1012-2012 and software V&V is associated with software life cycle [8]. DNBR algorithm verification process is performed by system verification plan and DNBR algorithm validation by system validation plan.

3. Verification

Verification activities are performed as follow sequence.

1) Get input data from KINGS APR1400 simulator
2) Develop test cases for seven modules
3) Perform software module test for seven VHDL modules
4) Compare test outputs with KEPCO-ENC CPCS development facility output

3.1 Software Module Test Cases

The KINGS APR1400 simulator is used to collect the inputs data and the final minimum DNBR at 100%, 75%, and 50% of reactor power. Four reactor coolant pump speed, cold leg temperature, hot leg temperature, primary

<Table 1> Software module test case

Module 2 Thermal Hydraulic Coolant Condition			
Input signal		Output signal	
Name	Value (hexa)	Name	Value (hexa)
TCMAX	22B.5	HIN	22B.0
PR	8CD.6	HFG	19F.D
TH1	266.A	HF	2BD.A
TH2	266.C	VIN	0.058
TC1	22B.4	GIN	2F1.0
TC2	22B.5	M5	1.06E
TH	266.B	M6	1.06E
N1	1	MC	1.06E
N2	1	XL	FFFF.A3E1
N3	1	CPOUT1	1.086
N4	1	CPOUT2	1.086
TC	22B.5	CPIN1	0.0FF0
		CPIN2	0.0FF0
		VC	0.F9B8

pressure, ex-core neutron flux, and CEA subgroups position are selected to develop the software module test cases. These input data are inserted to predeveloped Matlab algorithm to check the outputs for each module for verifying the results from VIVADO. Each module has different inputs and outputs at each power level. One of software module test cases is described in table 1. The input signals are inserted to predeveloped Matlab algorithm to check the outputs for each modules to be the references for verifying the results from VIVADO.

3.2 Software Simulation

Module test case inputs are inserted to VIVADO simulator to verify the outputs with Matlab results. The input values are converted to hexadecimal equivalent representation according to the designed word length and fraction length. When input values are entered to the VHDL software modules in hexadecimal by test case, hexadecimal output values are displayed in VIVADO simulation and written in the test result report.

<Table 2> Comparison minimum DNBR with KEPCO E&C facility

	100% Power	75% Power	50% Power
KEPCO E&C Facility	1.6500	2.4368	3.8574
FPGA Algorithm	1.6548	2.4256	3.747
Percentage Change	+0.29%	−0.46%	−2.86%
Sim Core Simulator	1.5718	2.5853	3.7004
KINGS APR1400	1.7120	2.1467	2.6997
Low DNBR Trip Setpoint	1.2800	1.2800	1.2800

3.3 Comparison with KEPCO E&C CPCS development facility

The KEPCO E&C CPCS development facility has been used to develop, verify and validate the CPCS software for Shin-kori 3&4 and Barakah Nuclear Power Plants 1,2,3&4. Input and output information of KEPCO E&C CPCS development facility could be used to compare with input and output signals of FPGA based DNBR algorithm. Final minimum DNBR values are verified comparing with CPCS development facility's DNBR outputs.

4. Results

Table 2 shows the minimum DNBR values of FPGA based DNBR algorithm, KEPCO E&C facility, percentage change, and low DNBR trip setpoint.

The final DNBR calculation deviations between CPCS development facility in KEPCO E&C and FPGA-based DNBR algorithm are +0.29% at 100% power, -0.46% at 75% power, and -2.86% at 50% power. And minimum DNBR values of FGPA based DNBR algorithm are more closed to the KEPCO E&C CPCS development facility outputs than Simcore simulator and KINGS APR1400 simulator outputs.

5. Conclusion

This paper shows the development process of FPGA based DNBR algorithm using systems engineering approach. Applying systems engineering approach to DNBR algorithm development increases the efficiency of life cycle and verifies software requirements more thoroughly.

The Vee model is used to develop the DNBR algorithm, which shows the software systems engineering process including reverse engineering and re-engineering approach. The SEMP defines how the FPGA-based DNBR algorithm project will be organized, structured, conducted, and controlled. Current APR1400 CPCS requirement and DNBR algorithm are analyzed in the reverse engineering process and FPGA based DNBR algorithm design and development are done in the re-engineering process. After design and development, DNBR algorithm is converted to VHDL language in the implementation process. And DNBR algorithm is verified and validated in the V&V process.

Test cases are developed to perform the software module test for seven modules using matlab and input signals and expected output signals are defined in each test cases. The APR1400 simulator is used to collect the input data in 100%, 75%, and 50% reactor power condition. Test input signals are entered to the VHDL software modules following test case tables and output signals are compared with the expected test value.

In 100%, 75%, and 50% reactor power cases, minimum DNBR values of FGPA based DNBR algorithm are more closed to the KEPCO E&C CPCS development facility outputs than Simcore program and KINGS APR1400 simulator outputs. It means FPGA based DNBR algorithm is closed to the current APR1400 CPCS DNBR algorithm.

Further work is planned to develop the LPD algorithm and integrate with DNBR algorithm to make the FPGA-based CPCS.

References

1. NPP Systems - Core Protection Calculator System, Text Book for BNPP NPP, Revision 2, p.1.
2. IAEA Nuclear Energy Series NP-T-3.17 Application of Field Programmable Gate Arrays in Instrumentation and Control Systems of Nuclear Power Plants, p.3~43.
3. Systems Engineering Principles and Practice, Alexander Kossiakoff, Second edition, p36-37, 355-407.
4. INCOSE Systems Engineering Handbook Fourth edition, p106-108.
5. Elliot J. Chikofsky, James H. Cross II, Reverse Engineering and Design Recovery: A Taxonomy, IEEE, January 1990, p.15~16
6. KEPCO & KHNP, Functional Design Requirements for a Core Protection Calculator System for APR1400, Revision 0, 2014.
7. Vitech corporation homepage, http://www.

vitechcorp.com/resources/core/onlinehelp/desktop/Views/Enhanced_Function_Flow_Block_Diagram_(EFBD).
8. IEEE 1012, IEEE standard for system and software verification and validation, 2012.
9. Donghwi Jeon, Jaecheon Jung, Design architecture of simplified core protection calculator system algorithm, Nuclear Engineering and Design, 2017.
10. Restu Maerani, Software verification process and methodology for the development of FPGA-based engineered safety features system, Nuclear Engineering and Design, 2017.
11. IAEA Nuclear Energy Series No. NP-T-1.5, Protecting against Common Cause Failures in Digital I&C Systems of Nuclear Power Plants.
12. Peter L. Hung, Core Protection Calculator System: Past, Present, and Future, ICONE18-29001.
13. Hyen-Jeong Kim, Field Programmable Gate Array-based I&C Safety System, ISOFIC/ISSNP 2014.
14. IEC 62566, Nuclear power plants - Instrumentation and control important to safety - Development of HDL-programmed integrated circuits for systems performing category A functions, 2012.
15. NUREG/CR-7006, Review Guidelines for Field-Programmable gate Arrays in Nuclear Power Plant Safety Systems, 2009.