Transactions of the Korean Nuclear Society Autumn Meeting
Gyeongju, Korea, October 27-28, 2016

Hazard Analysis of Software Requirements Specification for Process Module of FPGA-based Controllers in NPP

Sejin Jung [a], Eui-Sub Kim [a], Junbeom Yoo [a*], Jong Yong Keum[b] and Jang-Soo Lee[b]
[a]Division of Computer Science and Engineering, Konkuk University, 120 Neungdong-ro, Gwangjin-gu, Seoul, Republic of Korea
[b]Korea Atomic Energy Research Institute
[*]*Corresponding author: jbyoo@konkuk.ac.kr*

1. Introduction

FPGA (Field-Programmable Gate Array) has received much attention from nuclear industry as an alternative platform of PLC (Programmable Logic Controller) to develop digital I&C system. These systems should be identified that hazard or risk in systems are acceptably safe to operate. Hazard analysis is the process of identifying and evaluating the hazards of a system, and then either eliminating the hazard or reducing its risk to an acceptable level. Software hazard analysis *"... eliminates or controls software hazards and hazards related to interfaces between the software and the system (including hardware and human components). It includes analyzing the requirements, design, code, user interfaces and changes (NIST 1993)[1][2]."*

Software in PLC, FPGA which are used to develop I&C system also should be analyzed to hazards and risks before used. NUREG/CR-6430[2] proposes the method for performing software hazard analysis. It suggests analysis technique for software affected hazards and it reveals that software hazard analysis should be performed with the aspects of software life cycle such as requirements analysis, design, detailed design, implements. It also provides the guide phrases for applying software hazard analysis.

HAZOP (Hazard and operability analysis) is one of the analysis technique which is introduced in NUREG/CR-6430 and it is useful technique to use guide phrases. HAZOP is sometimes used to analyze the safety of software [7]. Analysis method of NUREG/CR-6430 had been used in Korea nuclear power plant software for PLC development [3][4]. Appropriate guide phrases and analysis process are selected to apply efficiently and NUREG/CR-6430 provides applicable methods for software hazard analysis is identified in these researches.

FPGA software also need to analyze its potential hazards and NUREG/CR-6430 is able to be useful methods. However, FPGA has a different development process from PLC, since it is a hardware-based platform. So software hazard analysis of FPGA software with NUREG/CR-6430 need to consider the applicability of methods. The safety analysis of FPGA software also had performed with several techniques [8], but hazard analysis of FPGA software with NUREG/CR-6430 is not before performed.

So there need to identify if the NUREG/CR-6430 is possible and useful to apply FPGA software requirements specification. In this paper, we performed the hazard analysis methods of NUREG/CR-6430 to DFLC-N[5] which is the prototype requirements specification of small modules in FPGA-based controllers. And we performed the HAZOP analysis with general guide words also. We analyze and compare these two approaches to identify the applicability of NUREG/CR-6430 methods to FPGA software requirement specification efficiently.

This paper organized as follows. Section 2 introduces the software hazard analysis methods of NUREG/CR-6430 and HAZOP as a background. Section 3 shows the result of the hazard analysis and discusses the results of analysis in section 4. Section 5 concludes the paper and provides remarks on future research extension and direction.

2. Software Hazard Analysis

2.1 NUREG/CR-6430

NUREG/CR-6430 is proposed by U.S.NRC (Nuclear Regulatory Commission) in order to suggest the software hazard analysis methods. Software hazard analysis in NUREG/CR-6430 is performed with the software life cycle aspects. <Figure. 1> shows the software hazard analysis scope and progress in NUREG/CR-6430. NUREG/CR-6430 does not fix the analysis techniques which are applied to each process (life cycle), although it recommends the HAZOP, FTA and FMEA.

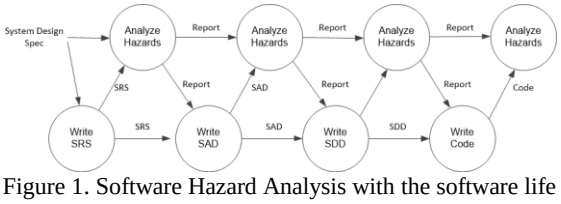

Figure 1. Software Hazard Analysis with the software life cycle (need modify)

Hazard analysis of software requirements is first started to identify the preliminary hazard and its

analysis. And identify the hazards which software is in any way responsible, next step is 'identify the software critical level' and 'match software requirement and system hazards (results of preliminary hazard analysis).' Finally, 'analyze each requirement using the guide phrases' and 'document the results of analysis' are proceed.

Table 1. Examples of guide phrases in NUREG/CR-6430

Quality	*Aspect*	*Phase*	*Guide Phrases*
Accuracy	Sensor	RADC	Stuck at all zeroes
		RADC	Stuck at all ones
		RADC	Stuck elsewhere
		RADC	Below minimum range
		RADC	Above maximum range
		RADC	Within range, but wrong
		RADC	Physical units are incorrect
		RADC	Wrong data type or data size
	Actuator	RADC	Stuck at all zeroes
		RADC	Stuck at all ones
		RADC	Stuck elsewhere
		RADC	Below minimum range
		RADC	Above maximum range
		RADC	Physical units are incorrect
		RADC	Wrong data type or data size

Guide phrases which are provided by the NUREG/CR-6430 consist of qualities, aspects, phase and contents. Qualities are sets of terms about addressed aspects of software in this report. <Table. 1> shows examples of the guide phrases about sensor, actuator accuracy in NUREG/CR-6430. Guide phrases also contains non-functional requirements such as 'security,' 'safety' etc. These guide phrases can be used to apply hazard analysis usefully.

2.2 HAZOP (Hazard and Operability Analysis)

HAZOP is a technique for identifying and analyzing hazards and operational concerns of a system. The HAZOP analysis utilizes key guide words and system diagrams to identify system hazards. Guide words such as no, more, less and part of are combined with process/system conditions in the hazard identification process. Generally, HAZOP is performed to use HAZOP worksheet which consists of table structure.

Table. 2 HAZOP worksheet example[6]

HAZOP Worksheet									
No.	Item	Function/ Purpose	Parameter	Guide word	Consequence	Cause	Hazard	Risk	Recommendation

HAZOP worksheet generally consists of 'Item,' 'Function/Purpose,' 'Parameter,' 'Guide word,' 'Consequence,' 'Cause,' 'Hazard,' and 'Recommendation.' Accordance with the worksheet table, HAZOP is performed to suppose the function/purpose and parameter are deviated by applying guide words and identify the consequence of deviation.

3. Application of Hazard Analysis Methods to DFLC-N

We use DFLC-N[5] which is the prototype version of software requirements used in a process module of FPGA-based controllers. It is a small part of the controllers. First, we identify the PHL (Preliminary Hazard List) of DFLC-N PM in order to perform hazard analysis. Identifying PHL is the process in NUREG/CR-6430. PHL is used to connect the software hazards to software related hazard in system/subsystem/component which level is in the target software.

We decide to PHL of FPGA hardware level, because the software requirements specification, which we use to analysis, is small FPGA software. PHL is classified 4 categories and each category has sub lists. <Table. 3> shows the PHL which we have identified to use hazard analysis of DFLC-N software.

Table.3 PHL of DFLC-N process module

No.	**Preliminary Hazard List – Process Module**
1	Power supply a. Loss of operating power b. Over current c. Overvoltage
2	Physical effects of internal/external a. Fire occurrence b. Physical impact c. Radioactivity
3	Operation error a. Operation error of application b. Memory error/failure c. Response time error(timing error, scan time) d. Error diagnosis function failure e. Lack of transmit capacity f. LED failure g. Disability of network
4	Operation failure a. Operation failure by operator (bypass)

3.1 Software Hazard Analysis with NUREG/CR-6430

HAZOP technique with analysis process of NUREG/CR-6430 and its guide phrases are used to perform software hazard analysis. Guide phrases are selected to apply analysis accordance with characteristics of FPGA. <Table. 4> shows a part of HAZOP results which are '*accuracy*' qualities and '*sensor*' aspects of guide phrases. It also contains PHL information which connects the SW and system, although we cannot write it limitation of pages.

Several potential hazards, which may be happen by the situation (applying guide phrases), is expressed in <Table. 4>. These hazards are almost concerned with section 9.2. The contents which are expressed in above table are related with sensor failures. When a sensor

fails to operate, software cannot control all of these failures accordance with the requirements specification.

Table 4. A part of analysis results about accuracy-sensor

Item	Function/ Purpose	Parameter	Guide Phrases	Consequence	Hazard
9.2 Operating voltage monitoring function	Read and output the signal	Read the operating voltage state value	Stuck at all zeroes	Receive 0 regardless of the current state Change the state to err when zero value continues with ten cycles	Display the normal state when operating voltage has normal value
			Stuck at all ones	Receive 1 regardless of the current state This stuck makes unreached error value	Display the error state to a normal state for abnormal operating voltage
			Stuck elsewhere	Making opposite state value is possible	Display the opposite state to current
			Below minimum range	Do not occur	X
			Above minimum range	Do not occur	X
			Within range, but wrong	Making opposite state value is possible	Display the opposite state to current
			Physical units are incorrect	Do not receive any state value by operating power monitor	Cannot operate normally with absence value
			Wrong data type or data size	Do not occur	X

3.2 Software Hazard Analysis with HAZOP

We also apply HAZOP without NUREG/CR-6430 methods and guide phrases to DFLC-N. We use guide words which are selected by before research for developing template of hazard analysis. It consists of 8 kinds of guide words, they are '*No*,' '*Reverse*,' '*Also*,' '*Early*,' '*Late*,' '*Part of*,' '*Before/After*,' '*Inadvertent*.' They are commonly used as a guide words introduced by [6][7].

<Table. 5> shows the HAZOP analysis results about section 9.2 'Operating voltage monitoring function' in requiremetns specification. There are 3 kinds of hazards when states applying with guide word is occurred, and we think that sensor, circuit or memory failures are one of the causes to hazards. Hazards which are appeared in <Table. 5> are derived from applying guide words of no, reverse, less and so on.

Table 5. A part of analysis results about 9.2 section

Item	Function /Purpose	Parameter	Guide Words	Consequence	Cause	Hazard
9.2 Operating voltage monitoring function	Read and output the signal	Make output err value when P33GD variable has error value	No(fail)	Cannot change state to err when operating voltage has strange	Counter failure Output circuit error Sensor failure	Circuit/function errors caused by Overvoltage
			Reverse	Make output to error value while current voltage operates normal	P33GD save memory failure Output circuit failure	Unintended init operation Display voltage error state
			Also	-	-	-
			Early	-	-	-
			Late	Change the state value is too late	Circuit or sensor failure	Checking voltage failure is done lately
			Part of	-	-	-
			Before/ After	-	-	-
			Inadvertent	-	-	-

Like this, two approaches of hazard analysis makes different results of analysis aspects. We show the difference and advantages by comparing and analyzing in the next section.

4. Discussion of the results of hazard analysis with comparison

The results of each approach which we explain and use above have some different aspects to analyze. These differences appear well in the guide phrases. Guide phrases of NUREG/CR-6430 have points (aspects) which have potential hazards in function of requirements and engineer supposes the situation to deviate the function of each points (aspects). On the other hand, guide words which we use in general HAZOP technique, identify the hazards while guide words are occurrence in specific function.

Table. 6 Comparison of Analysis Aspects with requirements point

Requirements Point	Analysis Aspects	
	NUREG/ CR-6430	HAZOP (General GW)
Sensor	Analysis of deviation	Cause
Input/output	Analysis of deviation	Cause
Timing	Analysis of deviation	Cause Analysis of deviation
Function	Analysis of deviation	Analysis of deviation
Circuit	Analysis of deviation	Cause
Security	Analysis of deviation	-
Memory	Cause	Cause Analysis of deviation
Data bus	(Analysis of deviation)	Cause Analysis of deviation
Network	(Analysis of deviation)	Cause Analysis of deviation

<Table. 6> shows the comparison results of analysis aspects about each point in requirements. This difference was brought from the perspective of differences in the application of the guide words. For instance, the requirement about sensor is analyzed which hazardous state can be occurred by deviating the sensor in NUREG/CR-6430, however HAZOP analyzes the sensor causes of the other hazards.

Table. 7 Comparison of PHL aspects

PHL	NUREG/CR-6430	HAZOP (General GW)
Operation error		
a. Operation error of application	O	O
b. Memory error/failure	N/A	O
c. Response time error	O	O
d. Error diagnosis function failure	O	O
e. Lack of transmit capacity	N/A	N/A
f. LED failure	O	O
g. Disability of network	N/A	N/A
Operation failure		
a. Operation failure by operator (bypass)	O	O

Difference of applying methods of guide phrases (words) makes difference to the result, we analyze these difference with comparing about PHL aspects first. We perform that identifying potential hazards which are founded by aspects of each approaches. <Table. 7> shows the results of analyze. HAZOP with general GW finds one more hazard compared with NUREG/CR-6430 about '*Memory error/failure*'

It does not means that HAZOP with general guide words is more than useful rather than NUREG/CR-6430, it just shows the difference about analysis aspects of two approaches. The reason why these phenomenon occurs that guide phrases about memory is not contained in NUREG/CR-6430. Software hazard analysis does not concerns about hardware characteristics and failures generally. On the other hand, software hazard analysis of FPGA concerns hardware characteristics, FPGA is hardware-based platform.

Additionally, NUREG/CR-6430 provides the additional guide phrases about non-functional requirements like security, safety, and so on. These guide phrases make possible to identify that requirements specification considers or defines contents related with these phrases. This point may effect an advantage to perform software hazard analysis. Providing the guide phrases which have points of potential hazards can help to apply easy rather than HAZOP directly.

Likewise, we perform the software hazard analysis of requirements specification level with two approaches and compare the results of each approach. Each approaches have different points of analysis aspects and portion. NUREG/CR-6430 method is enough to apply for FPGA software, despite it has some supplement points about hardware characteristics of FPGA.

5. Conclusion and Future Work

We perform software hazard analysis of FPGA software requirements specification with two approaches which are NUREG/CR-6430 and HAZOP with using general GW. We also perform the comparative analysis with them. NUREG/CR-6430 approach has several pros and cons comparing with the HAZOP with general guide words and approach. It is enough applicable to analyze the software requirements specification of FPGA.

We are now planning to supplement the guide phrases in NUREG/CR-6430 to apply FPGA SW requirements specification efficiently. We also compensate our hazard analysis results using the supplemented guide phrases.

Acknowledgements

This research was supported by the Ministry of Science, ICT & Future Planning.

REFERENCES

[1] NIST 1993, Review of Software Hazard Analysis, National Institutes of Standards and Technology, Draft June 1993.
[2] U.S. Nuclear Regulatory Commission, Software Safety Hazard Analysis, 1995.
[3] Korea Atomic Energy Research Institute, Safety Analysis Report of Reactor Protection System software requirements specification, KNICS-RPS-SVR122 Rev.01, 2007.
[4] Gee-Yong Park, Jang-Soo Lee, Se-Woo Cheon, Kee-Choon Kwon, Eunkyoung Jee and Kwang Yong Koh, Safety Analysis of Safety-Critical Software for Nuclear Digital Protection System, International Conference on Computer Safety, Reliability, and Security, pp.148-161, September 18-21, Germany,2007.
[5] Korea Atomic Energy Research Institute, NTIP-FLC-SRS201.
[6] Clifton A. Ericson, Hazard Analysis Techniques for System Safety, Wiley interscience, 2005.
[7] UK Ministry of Defence, Defence Standard 00-58: HAZOP Studies on Systems Containing Programmable Electronics, 1996.
[8] Martin Remnant, The Application of Sneak Analysis To Safety Critical FPGAs, MSc Safety Critical Systems Engineering, The University of York, 2009.