Annals of Nuclear Energy 94 (2016) 338–349

Contents lists available at ScienceDirect

Annals of Nuclear Energy

journal homepage: www.elsevier.com/locate/anucene

An evaluation and acceptance of COTS software for FPGA-based controllers in NPPS ☆

Sejin Jung [a], Eui-Sub Kim [a], Junbeom Yoo [a,*], Jang-Yeol Kim [b], Jong Gyun Choi [b]

[a] Konkuk University, Division of Computer Science and Engineering, 120 Neungdong-ro, Gwangjin-gu, Seoul 05029, Republic of Korea
[b] Korea Atomic Energy Research Institute, Man-Machine Interface System Team, 989-111 Deadeok-daero Yuseong-gu, Daejeon 34057, Republic of Korea

ARTICLE INFO

Article history:
Received 12 January 2016
Received in revised form 28 March 2016
Accepted 31 March 2016

Keywords:
Acceptance process
Evaluation criteria
COTS SW dedication
FPGA
Logic synthesis tools
Digital I&C

ABSTRACT

FPGA (Field-Programmable Gate Array) has received much attention from nuclear industry as an alternative platform of PLC (Programmable Logic Controller)-based digital I&C (Instrumentation & Control). Software aspect of FPGA development encompasses several commercial tools such as logic synthesis and P&R (Place & Route), which should be first dedicated in accordance with domestic standards based on EPRI NP-5652. Even if a state-of-the-art supplementary EPRI TR-1025243 makes an effort, the dedication of indirect COTS (Commercial Off-The-Shelf) SW such as FPGA logic synthesis tools has still caused a dispute. This paper proposes an acceptance process and evaluation criteria, specific to COTS SW, not commercial-grade direct items. It specifically incorporates indirect COTS SW and also provides categorized evaluation criteria for acceptance. It provides an explicit linkage between acceptance methods (Verification and Validation techniques) and evaluation criteria, too. We tried to perform the evaluation and acceptance process upon a commercial FPGA logic synthesis tool being used to develop a new FPGA-based digital I&C in Korea, and could confirm its applicability.

1. Introduction

PLC (Programmable Logic Controller) has been widely used to develop digital I&Cs in nuclear power plants. The sharply-rising maintenance cost, however, as well as several problems such as the lower computation power, cyber-security (Regulatory Guide 5.71, 2010; IEC62645, 2014) and common cause failure (CCF), have requested for alternative solutions, *i.e.*, FPGA (Field-Programmable Gate Array) (TR-1019181, 2009; TR-1022983, 2011; Survey of the CPLD/FPGA Technology, 2009). FPGA is able to provide the higher computation power than PLC with lower hardware cost and also provides the diversity of systems (Yoo and Seong, 2002; Kelly and Murphy, 1990). Many researches now try to use FPGA as an implementation platform of digital I&Cs (Yoo et al., 2013; Choi and Lee, 2012; Bakhmach et al., 2010; Hayashi et al., 2014; Clarkson, 2008; She and Jiang, 2009; sook Jang et al., 2008).

FPGA has a different development process from PLC (Yoo et al., 2013), since it is a hardware-based platform. An FPGA design (*i.e.*, software) is first designed with HDL (Hardware Description Language) such as Verilog (2001) and VHDL (2008), and then is subsequently synthesized into gate-level designs and physical layouts by FPGA logic synthesis and P&R tools, mechanically (Brown and Rose, 1996). Commercial FPGA logic synthesis tools (*e.g.*, '*Synplify Pro*' (Synopsys, 2015), '*Precision RTL*' (Mentor Graphics, 2015a) and '*Encounter RTL Compiler*' (Cadence, Encounter Conformal LEC, 2015) and FPGA EDAs[1] (*e.g.*, '*Xilinx ISE Design Suit*' (Xilinx ISE design suite, 2015), '*Altera Quartus 2*' (Altera Quartus 2, 2015) and '*Actel Microsemi Libero SoC*' (Microsemi Libero SoC, 2015) make the synthesis almost mechanical.

All software (also hardware) used directly as a safety-grade controller or indirectly to develop them should be developed and maintained under quality assurance programs such as 10CFR50 App.B (U.S. Code of Federal Regulations) or NQA-1 certification (The American Society of Mechanical Engineers, 2015), or should be certificated/selected by appropriate standards (Andryushin et al., 2014). If not, they should be dedicated through international reports/guidelines such as EPRI NP-5652 (2014) and TR-106439 (1996). The Korean regulatory also requires to comply with KINS/RG-17.12 (2011), which is based on them above. The FPGA logic synthesis tools mentioned above are widely recognized by many researchers and organizations (*e.g.*, SCC (Software Certification Consortium) (SCC) as an item which should be dedicated strictly,

☆ A preliminary version of this paper was presented in *ISET 2015* (Yoo et al., 2015).

* Corresponding author.
E-mail address: jbyoo@konkuk.ac.kr (J. Yoo).

[1] EDA: Electronic Design Automation.

http://dx.doi.org/10.1016/j.anucene.2016.03.026
0306-4549/

since they produce FPGA hardware designs mechanically without human intervention.

The problem we now face is that the reports/guidelines EPRI NP-5652/TR-106439 do not specifically incorporate the indirect COTS SW such as FPGA logic synthesis tools. They provide a well-defined process for dedication, but focus on commercial-grade (hardware) items, which directly compose safety-grade controllers. Even if an supplementary report EPRI TR-1025243 (2013) has recently been proposed to resolve the case of indirect software specifically, it still judges that these tools are not the subject of COTS SW dedication, contrary to expectations. The general consensus (SCC; Santhanam, 2002; Evaluation of Guidance for Tools, 2015) among the industry is that FPGA logic synthesis tools should be dedicated more strictly than other indirect COTS SW.

This paper proposes an acceptance process and evaluation criteria, specialized for the dedication of indirect COTS SW as well as direct ones. (Step 1) It first recognizes an indirect COTS SW as a target of dedication, unlike EPRI NP-5652/TR-106439. (Step 2) It then determines the safety category of the target SW and identifies detailed evaluation criteria for acceptance. We adopted and modified those from US.NRC NUREG/CR-6421 (1996). (Step 3) Acceptance methods and specific V&V techniques which can satisfy the evaluation criteria successfully are selected. It also can provide an explicit linkage from evaluation criteria identified in Step 2 to acceptance methods and V&V techniques selected in Step 3. The dedication with the acceptance methods then gets started. (Step 4) We can finally accept the target SW on the basis of the judgement whether the target SW satisfies its evaluation criteria or not, thorough various information and evidences which we can gather after getting through with the acceptance methods.

We also performed a case study with a commercial FPGA logic synthesis tool – '*Synopsys Synplify Pro*' which are being used to develop a prototype (Choi and Lee, 2012) of digital I&C in Korea. We tried to perform the COTS SW dedication according to the proposed evaluation criteria and acceptance process, and received positive response from the experts who have to prepare the COTS SW dedication before long.

The paper is organized as follows: Section 2 introduces the FPGA development and verification processes as background. Section 3 explains and compares the relevant standards and reports. The evaluation criteria and acceptance process for COTS SW is proposed in Section 4. The case study with an indirect COTS software is introduced in Section 5, and Section 6 surveys related work and the state-of-the-art on the field of COTS SW dedication. Section 7 concludes the paper and provides remarks on future research extension and direction.

2. FPGA development and V&V processes

The development life-cycle of FPGA-based digital I&Cs follows IEC 61513 (2011) basically. An FPGA-based system, however, has a specific feature that the part of development life cycle using HDL (Hardware Description Languages) is classified into software, while after downloading to chip is classified into hardware. FPGA, therefore, should be developed to comply with both IEC 60880 (2006) in terms of software and IEC 60987 (2007) in terms of hardware. Fig. 1 depicts the V-shaped life-cycle of FPGA development explained in IEC 62566 (2012), consisting of software and hardware aspects. The software aspect also has a typical development life-cycle (NUREG/CR-7006, 1996) presented on the left-hand side of the figure.

The FPGA software development is fully automated by FPGA logic synthesis tools and commercial EDAs of FPGA vendors. After programming an RTL (Register-Transfer Level) design with HDLs, the design is transformed into a gate-level design (*i.e.,* netlist) by

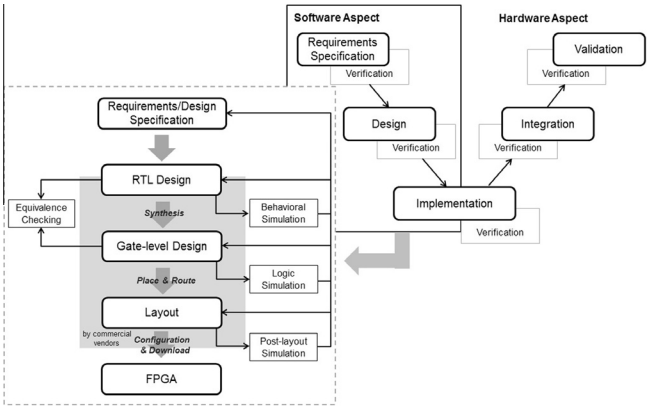

Fig. 1. The V-shaped life-cycle of FPGA development.

synthesis software such as '*Synopsys Synplify Pro*', '*Precision RTL*' and '*Encounter RTL Compiler*'. The EDAs of FPGA vendors such as '*Xilinx ISE Design Suit*', '*Altera Quartus 2*' and '*Microsemi Libero SoC*' perform P&R to physically place and map all netlist elements and prepare a downloadable file through configuration.

At each step of the FPGA software development life-cycle, designers perform '*simulation-based verification*' in order to confirm that each artifact satisfies its requirements specification. The first simulation on RTL designs is called '*behavioral simulation*' and aims to confirm that all requirements are implemented into the RTL design correctly. As most designers develop RTL designs manually, it takes much time. After logic synthesis from RTL to Gate-level design, designers perform '*logic simulation*' in order to confirm that all functionalities were preserved during the synthesis. After P&R, they can validate the layout via '*post-layout simulation*' to check that the layout meets all timing requirements. Simulators such as '*ModelSim*' (Mentor Graphics, 2015b) and '*Questa Simulator*' (Mentor Graphics, 2015c) are widely used for the '*simulation-based verification*'. Every simulation-based verification at each step is performed individually and independently, and it is considered as one of key factors for efficient FPGA development.

The V&V process of the FPGA development includes *equivalence checking* (Huang and Cheng, 1998; Huang et al., 2000; Kuehlmann and Krohm, 1997; Kuehlmann et al., 1995; Burch et al., 1994) as well as the *simulation* techniques. The equivalence checking can prove that two given designs have the same functionality, *i.e.,* "*whether they show the same behavior for all possible input sequences.*" For example, it can prove that an RTL design and the gate-level design synthesized from the RTL design always show the same behavior. As the synthesis and optimization of EDA tools becomes increasingly sophisticated, we may encounter various unintended and unexpected behavior of FPGA designs. The equivalence checking can help us ensure that the synthesis or optimization worked correctly.

3. The COTS SW dedication

All software which were not developed in accordance with 10CFR50 App.B or NQA-1 certification should be first dedicated by EPRI NP-5652/TR-106439 to be used directly in digital I&Cs or indirectly to develop (*e.g.,* compile, synthesis, testing, et. al.) other directly-used software. They try to verify the software through appropriate processes and methods, and demonstrate that the software can be regarded as the same one developed in accordance with 10CFR50. EPRI NP-5652 set up dedication guidelines about commercial mechanical/electrical items, and EPRI TR-106439 added guidelines for software-based digital equipments (*e.g.,* PLC). These EPRI reports, however, do not distinguish direct and

indirect software specifically. TR-1025243 has been recently proposed to supplement them with the judgment criteria for indirect COTS SW. NUREG/CR-6421 is a guideline for COTS software in nuclear power plants and considers indirect software in details, but it has no common position yet.

3.1. EPRI NP-5652/TR-106439

Fig. 2 overviews the dedication process of NP-5652. It first identifies the item procured and checks whether it performs a safety function or not. If the item does, then it checks if the item should have been procured as a basic item, which already completed the dedication process. If not, the item is a commercial grade item and should be dedicated according to its critical characteristics. EPRI TR-106439 (NP-5652) calls software among commercial grade items as COTS software. The process has 3 categories of critical characteristics, *Physical*, *Performance* and *Dependability*, and each has attributes to be verified/demonstrated by '*Acceptance Method(s)*'. We need to select appropriate methods which can verify the critical characteristics efficiently and sufficiently, especially for Method 1 (Special Tests and Inspection).

The process is, however, not suitable for the dedication of indirect COTS SW such as FPGA logic synthesis tools, which are pertinent to our discussion. Such software do not perform a safety function (IEC, 1997; IEC 61513, 2011) and then takes the No* flow. That is one of motivations that this paper proposes an extended and refined dedication process for indirect COTS software. If we interpret the criterion ("*Does item perform a safety function?*") as it is, we do not have to do the COTS SW dedication for FPGA logic synthesis tools.

If we had performed the next steps (virtually), the *Physical* characteristic is not applicable for COTS SW. *Accuracy* and *Functionality* in *Performance* and all attributes in *Dependability* would be applicable. Method 1 would be selected for the *Performance*, and special tests such as equivalence checking, model checking and testing techniques would be used. We need to carefully determine appropriate special tests techniques for the dedication target – indirect COTS SW. Method 2 and 4 would be selected for the *Dependability* characteristic. Quality assurance programs of supplier, V&V processes, configuration management processes, design reviews, test reports, requirements traceability, bug & error reporting and tracking, and etc. would be the scope of these methods. The reports, however, do not provide any precise criteria how we can reach a specific level of sufficient dedication. It provides no obvious linkage from *Critical Characteristics* to *Acceptance Methods*, which is an important precondition to be evaluated as "*Accepted*".

3.2. EPRI TR-1025243

EPRI TR-1025243 focuses on computer programs (*i.e.*, not hardware items) and adopts '*Functional Safety Classification*' to determine whether a computer program should be dedicated or not. Fig. 3 summarizes the classification process.

If the target computer program is classified into *direct/indirect safety-related (1,1-1)*, the application of dedication process of NP-5652/TR-106439 is required. All direct COTS SW performing safety-related SCC will be classified into *Direct safety-related (1)*, while a '*pipe stress calculation and analysis SW*' will be an example of *Indirect safety-related (1-1)*. If the software is not integral to a safety-related SSC (System, Structure or Component), it will be *Non-safety-related (2)* or *Non-safety-related but augmented quality (3)*. Since a '*seismic analysis SW*' is used to design or analyze a safety-related SSC and its result can be verified independently, it is classified into *Non-safety related but augmented quality*. An '*inventory management system*' is an example of *Non-safety-related*, presented by EPRI TR-1025243. We can find that FPGA logic synthesis tools, which this paper is interested in, will be classified into *Non-safety related but augmented quality*, contrary to expectations.

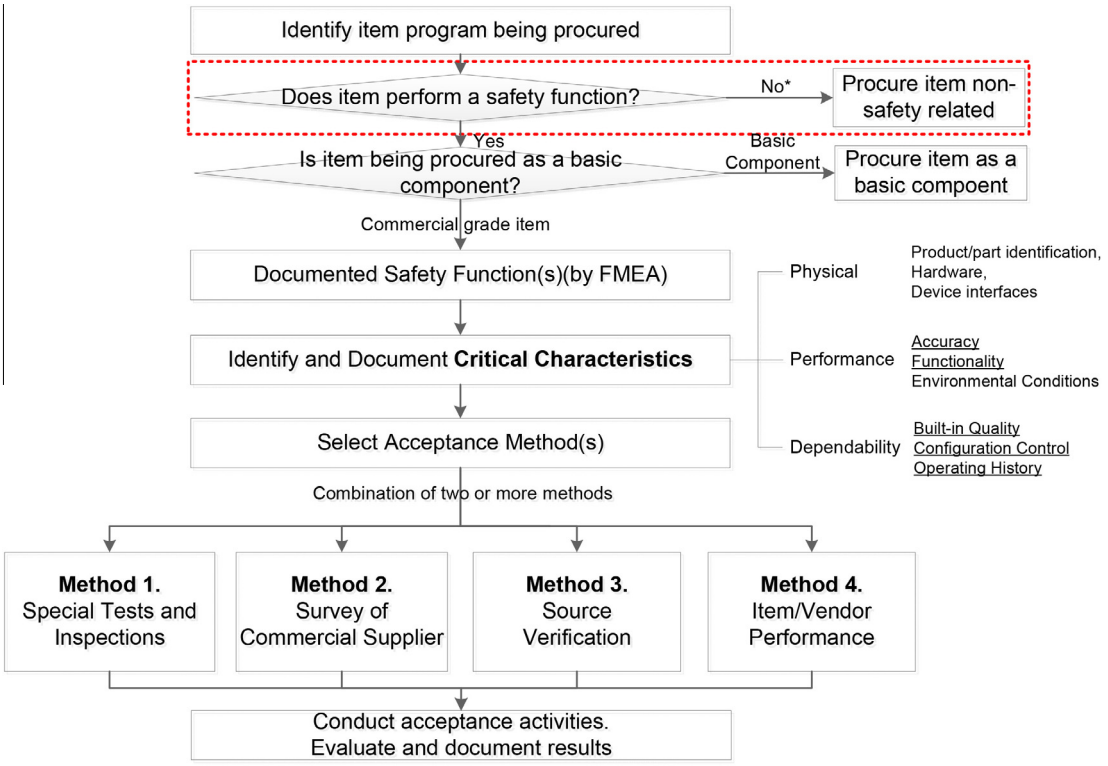

Fig. 2. The dedication process of EPRI NP-5652/TR-106439.

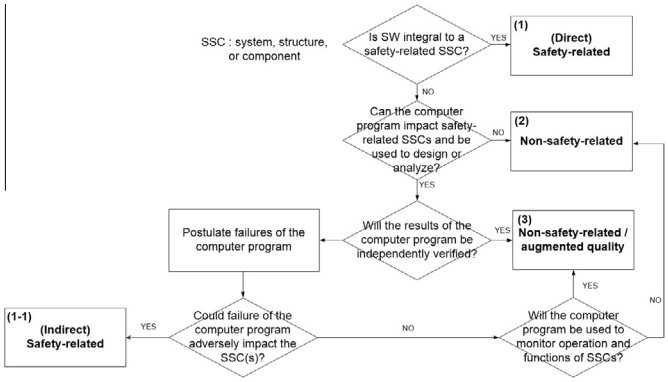

Fig. 3. The functional safety classification process of TR-1025243.

3.3. NUREG/CR-6421

Fig. 4 explains the COTS SW dedication process of NUREG/CR-6421. It aims for the dedication of COTS SW, both used directly and indirectly. It identifies safety functions of COTS software as well as its target system, and then determines the '*Safety Category*' of the target system and '*Usage Category*' of the COTS software in order to determine the *Safety category* of the COTS software at last.

Safety category uses the categorization of IEC 61226 (2009), *i.e., A*, *B*, *C* and *Unclassified*. Safety-level systems such as RPS (Reactor Protection System) and ESFAS (Engineered Safety Features Actuation System) are the examples for the category *A*. *Usage category* of the COTS SW is determined as *Direct*, *Indirect*, *Support* and *Unrelated*. Finally, the safety category of the COTS software is determined as explained in Table 1.

The detailed criteria for determining safety category of COTS software is as follows. The safety category of *direct* COTS software is the same with that of its target system. For example, a COTS software used directly in an RPS will have the same category with it target system – '*A*'. However, in case of *indirect* COTS software, if the result of the COTS software can be verified, the safety category of the indirect COTS software is identified as a lower category than its target system. For example, an FPGA logic synthesis tool such as '*Synopsys Synplify Pro*' is used to generate a preliminary program (*i.e.,* netlists) of RPS, which can be verified through testing, simulation and formal verification. We can identify the safety category of the synthesis tool as '*B*'.

Fig. 5 shows a summarized information about the detailed dedication criteria for the COTS software of safety category '*B*'. Up to our best knowledge, the safety category '*B*' is the highest one which any indirect COTS software for FPGA-based digital I&Cs can be categorized. Dedication criteria of the '*B*' category consists of 6 steps (B5–B10) and concern with *SQA (Software Quality Assurance) plan*, *quality of software*, and *operating experience in similar applications*. As shown in the illustrated example above, it provides more detailed criteria than EPRI NP-5652/TR-106439.

Table 1
Usage and safety categories of COTS SW.

Usage category	Description	Safety category of COTS SW
Direct	Directly used in A, B or C application	A, B, C
Indirect	Directly produces executable modules that are used in A, B, or C application (compiler, linker)	A, B, C, Unclassified
Support	CASE systems, or other support systems that indirectly assist the production of applications	Unclassified
Unrelated	Software that has no impact A, B, or C applications	Unclassified

3.4. Comparative analysis on the standards and reports

Table 2 summarizes our comparative analysis on three groups of the standard and reports for COTS software dedication. They have both different and common points. First of all, the dedication targets of NP-5652/TR-10643 are different from others. While NP-5652/TR-106439 aim at commercial-grade items, the target of NUREG/CR-6421 and TR-1025243 are COTS SW only. The usage of COTS items are also different. NP-5652/TR-106439 consider direct items only, but others consider direct/indirect items both. NP-5652/TR-106439/TR-1025243 do not provide grading of safety and categorization of criteria, but NUREG/CR-6421 provides several levels of dedication criteria for each category. Therefore, NUREG/CR-6421 provides more detailed and obvious linkage from criteria to acceptance methods. NP-5652/TR-106439 and TR-1025243 refer to the available before-dedication-records, but NUREG/CR-6421 does not.

They also have something in common. Demonstrating critical characteristics of TR-106439 is very similar with the dedication criteria of NUREG/CR-6421. For example, the part concerning to identifying software quality assurance program/plan is the same with each other. Review of operating history of a target software is also performed by all approaches.

4. The acceptance process and evaluation criteria for indirect COTS SW

This paper proposes an acceptance process and evaluation criteria, *i.e.,* '*dedication process*', for not only direct COTS SW but also indirect ones such as FPGA logic synthesis tools which are being used to develop FPGA-based digital I&Cs. It keeps pace with EPRI NP-5652/TR-106439. Detailed acceptance criteria and acceptance methods will be selected as the reports, but the acceptance criteria are now strengthened with the ones based on '*safety category*' which we adopted and modified from NUREG/CR-6421. Since NP-5652/TR-106439 uses critical characteristics (*e.g., physical*, *performance* and *dependability*) as the information source of acceptance criteria, it provides only ambiguous guidelines to determine the acceptance. We now have specific and clear criteria for each

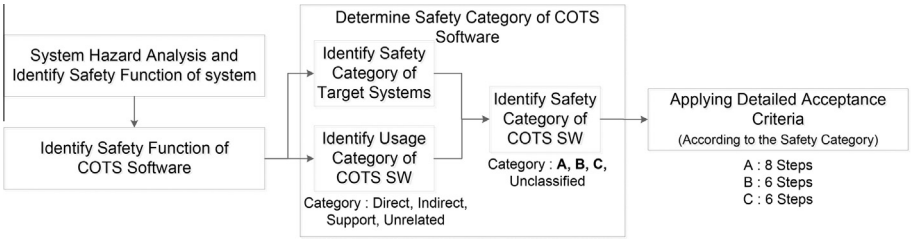

Fig. 4. The dedication process of NUREG/CR-6421.

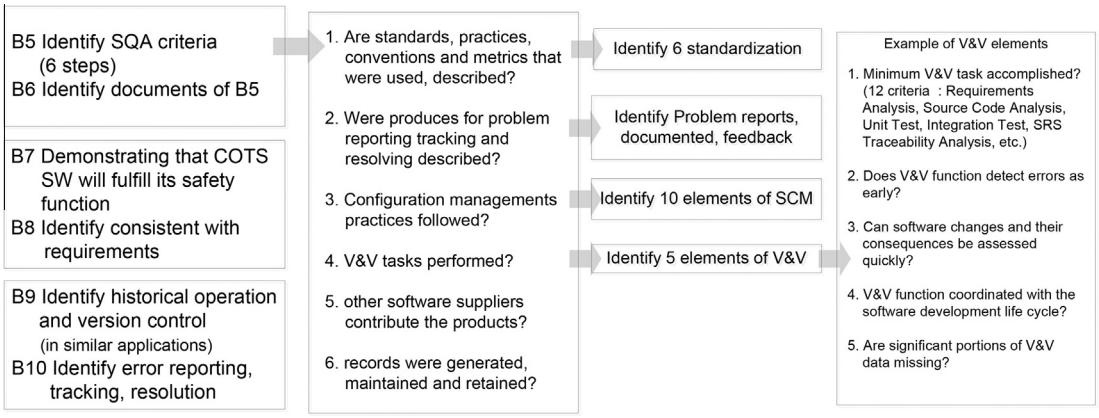

Fig. 5. An illustrated example of a chain of dedication criteria for the '*B*' category.

Table 2
Comparison between NP-5652/TR-106439/TR-1025243 and NUREG/CR-6421 for COTS SW.

	NP-5652/TR-106439	TR-1025243	NUREG/CR-6421
Dedication target	Commercial-grade item (COTS HW + COTS SW)	Computer program (COTS SW)	COTS SW
Usage	Direct	Direct/Indirect	Direct/Indirect
Grading of safety, Categorization of criteria	X	X	O
Direct linkage from criteria to acceptance	X	X	O
Use of available before-dedication-records	O (for the last 3 years)	O (for the last 3 years)	X
Identification of SW QA plans	O	O	O
Review of operating history	O	O	O

safety category like *A*, *B* and *C*, and the determination of acceptance will be more objective and verifiable. Fig. 6 overviews the extended and refined dedication process, consisting of four parts: *1. Basic Analysis*, *2. Identifying Acceptance Criteria*, *3. Determining Acceptance Methods*, and *4. Dedication*.

(1) *Basic analysis* The process starts with identifying an item, *i.e.,* COTS SW, to dedicate (1.1). If the software was supposed to be procured as a basic item, which should have passed certification, we don't have to do the dedication process for it (1.2). Then it checks whether it performs a safety function or not (1.3). If not, it tries to confirm whether it is an indirect COTS SW such as compilers and synthesis tools. In case of such software which produces outputs/results, it regards it as the one performing the safety functions of its target system, and identifies as a target of COTS SW dedication. On the other hand, NP-5652/TR-106439 do not consider such software as an item to dedicate.
(2) *Identifying acceptance criteria* We then determine the safety category of the direct/indirect COTS SW (2.1). It first identifies the safety category of its target system (2.1.1) and the usage category of the COTS SW (2.1.2). The safety category of the COTS SW is then determined (2.1.3). We then can identify a list of dedication criteria for each category (2.2), which will be used as dedication criteria later. We organized the acceptance criteria as shown in Table 3. They consist of '*Functionality*', '*SQA*' and '*Operating History*' categories and detailed criteria (questions) to satisfy them are also provided. The acceptance criteria require that the higher level of safety category should satisfy more criteria than the lower ones. It is worth to note that the safety category A and B might be an indicator of direct/indirect COTS SW. Even if it can be classified into the category A COTS SW, because of its safety and importance, the accessability to source codes and development/verification environments might determine the category as B.
(3) *Determining acceptance methods* The process then refines each acceptance criterion in terms of the target item (3.1). Obvious meaning of each acceptance criterion should be defined, and then acceptance methods to demonstrate the satisfaction of acceptance criteria will be selected (3.2). We have 4 kinds of acceptance methods as EPRI NP-5652, *i.e.,* (Method 1) special tests and inspections, (Method 2) survey of supplier, (Method 3) source verification and (Method 4) item/vendor performance. Finding appropriate V&V techniques to confirm/check/verify or demonstrate acceptance criteria for the required safety category is the next process (3.3). Table 4 summaries the selectable V&V techniques, which we extracted and summarized from various international standards of functional safety and dedication.
(4) *Dedication* We now apply the selected acceptance methods/techniques to the target item – the direct/indirect COTS SW (4.1). All methods try to demonstrate/prove that the software is sufficiently satisfied with each acceptance criterion, and the (refined) criteria from (2.2) help us judge if the criteria are actually satisfied or not. Finally, we can determine whether the COTS SW is to be acceptable or not (4.2).

In summary, the proposed evaluation criteria and acceptance process for direct/indirect COTS SW uses the dedication process of NP-5652/TR-106439 as a baseline, whereas adopted and refined the safety categorization and acceptance/evaluation criteria of NUREG/CR-6421 to complement and clarify the final decision. It can apply to indirect COTS SW such as compilers and FPGA logic synthesis tools, unlike NP-5652/TR-106439. The grading/categorization approach will support appropriate selection of acceptance methods and V&V techniques, and also will be used as a firm basis of the final determination, since the process provides an explicit linkage between selected acceptance methods (V&V techniques) and evaluation criteria to be satisfied with. The process also provides a detailed guideline for selectable V&V techniques.

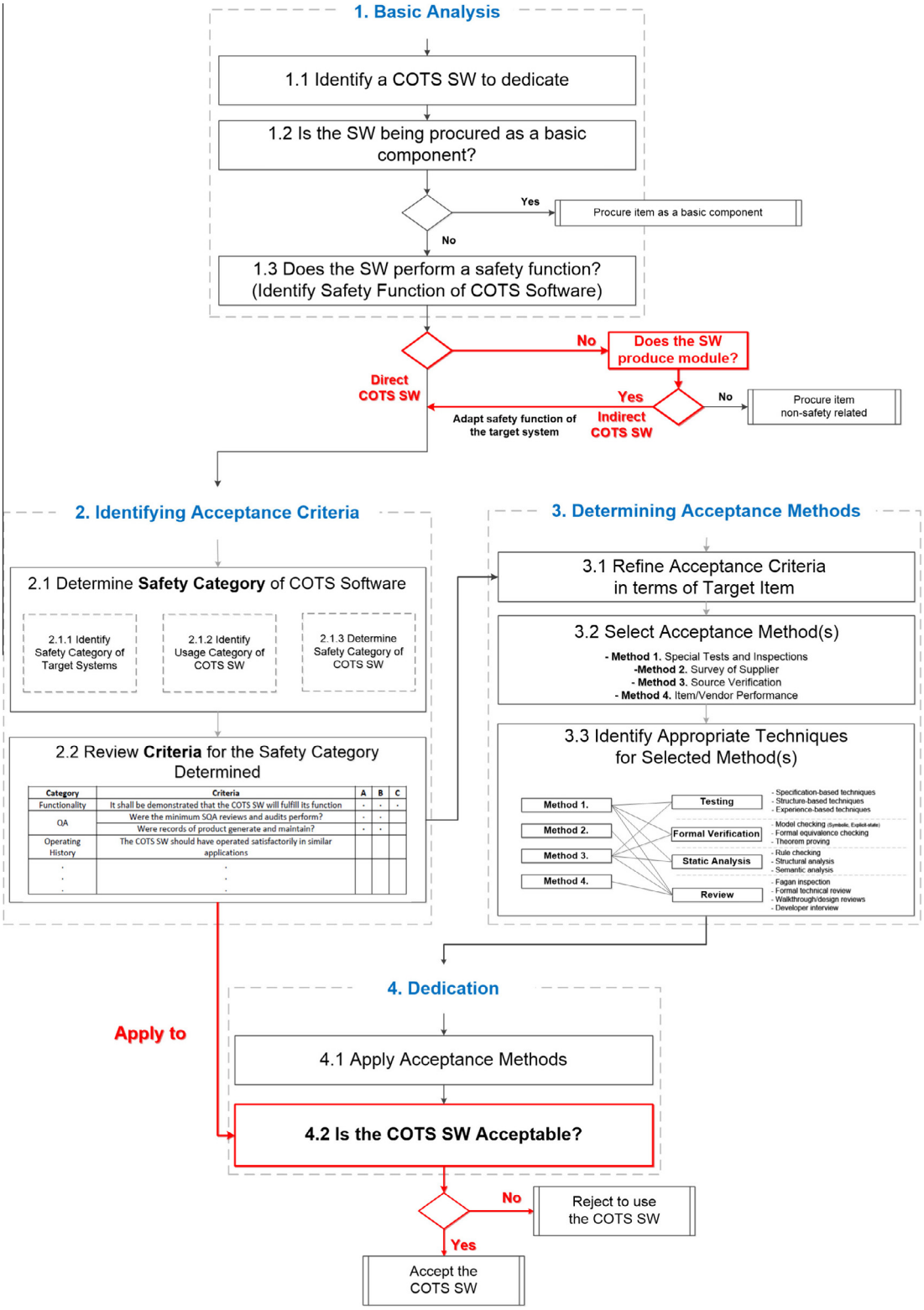

Fig. 6. The acceptance process and evaluation criteria for indirect COTS SW.

It is worth to note that this paper applies the dedication process to indirect COTS SW only, even if we can apply it to direct COTS SW without difficulty. It is due to the fact that there is no case/example of direct COTS SW used to develop digital I&Cs in Korea, to the best of our knowledge. This paper wants to collect enough evidences before insisting it.

5. Case study

This section performs a case study with an indirect COTS SW, which are widely used to develop a new FPGA-based digital I&C in Korea (Choi and Lee, 2012) and also should be dedicated before long. The indirect COTS SW to dedicate is a FPGA logic synthesis

Table 3
Acceptance criteria for indirect COTS SW.

Category		Criteria	A	B	C
Functionality	Functionality	It shall be demonstrated that the COTS SW will fulfill its safety function.	•	•	•
		The COTS SW should be consistent with system requirements.	•	•	•
		The interfaces between the COTS SW and systems shall be identified, clearly defined, and under CM.	•		
SQA	QA	Does the plan describe responsibilities, authority, and relations between SQA teams and software development teams?	•		
		Were the minimum SQA reviews and audits performed? (A-8)	•		
		Standards, practices, conventions, and metrics that were used, described?	•	•	
		Were procedures for problem reporting, tracking, and resolving described?	•	•	
		Well-managed other supplier? If exists	•	•	
		Were records of product generate and maintained?	•	•	
	V&V	Is the organizational structure of the V&V and software development independently?	•		
		25 kinds of V&V tasks are performed? (NUREG/CR-6421 App. A 12)	•	•	•
		Do V&V function detect errors early as possible?	•	•	
		Can software change be assessed quickly?	•		
		Are V&V function coordinated with the development?	•	•	
	SCM	Does the plan describe responsibilities, authority, and relations between CM and development?	•	•	
		At least one configuration control board is required.	•	•	
		Does the configuration management operation provide the following required functions?	•	•	
		CM is founded upon the establishment of "configuration baselines" for each version of each product?	•	•	
		Is the level of authority required for change described?	•	•	
		Dose status accounting include?	•	•	
		Software products under control for each supplier?	•	•	
		All Records to be maintained and identified?	•	•	
Operating history	History	The COTS SW shall have significant (greater than 1 year) operating time, with severe error-free operating experience.	•		
		The COTS SW should have operated satisfactorily in similar applications.		•	•
		Configuration management and update should provide traceability.	•	•	
	Error Tracking	The version and release have no major unresolved problems and bug list should be available to COTS purchaser as a support option.	•	•	•
		Error reporting, tracking and resolution should be consistent and correctly attributable to version and release is well managed.	•	•	

Table 4
Selectable V&V techniques.

V&V techniques			Applicable acceptance method (s)
Testing	Specification-based Techniques (ISO/IEC/IEEE 29119-4, 2015)	Equivalence partitioning	Method 1
		Classification tree method	
		Boundary value analysis	
		Syntax testing	
		Combinatorial testing	
		Decision table testing	
		Cause-effect graphing	
		State transition testing	
		Scenario testing	
		Random testing	
	Structure-based techniques (ISO/IEC/IEEE 29119-4, 2015)	Statement testing	Method 1
		Branch testing	Method 3
		Decision testing	
		Branch condition testing	
		Modified condition decision testing	
		Data flow testing	
	Experience-based techniques (ISO/IEC/IEEE 29119-4, 2015)	Error guessing (IEC 61508-7, 1997)	
Formal verification		Model checking (Clarke et al., 1999)	Method 1
		Formal equivalence checking (Huang and Cheng, 1998)	Method 3
		Theorem proving (Duffy, 1991)	
Static analysis		Rule checking (Laski and Stanley, 2009)	Method 1
		Structural analysis (Bush et al., 2000)	Method 3
		Semantic analysis (Laski and Stanley, 2009)	
Review		Fagan inspection (Fagan, 1986)	Method 1
		Management review (IEEE 1028, 2008)	Method 2
		Formal technical review (Collofello)	Method 3
		Walkthrough/design review (IEC 61508-7, 1997)	Method 4
		Developer interview (IEEE 1012, 2012)	

tool '*Synopsys Synplify Pro* (Synopsys, 2015)' used embedded in '*Actel Libero SoC*' EDA (Microsemi Libero SoC, 2015) as shown in Fig. 7.

The FPGA logic synthesis software plays an important role in the FPGA development as explained in Fig. 1, since it translates an RTL design to an equivalent gate-level design without human

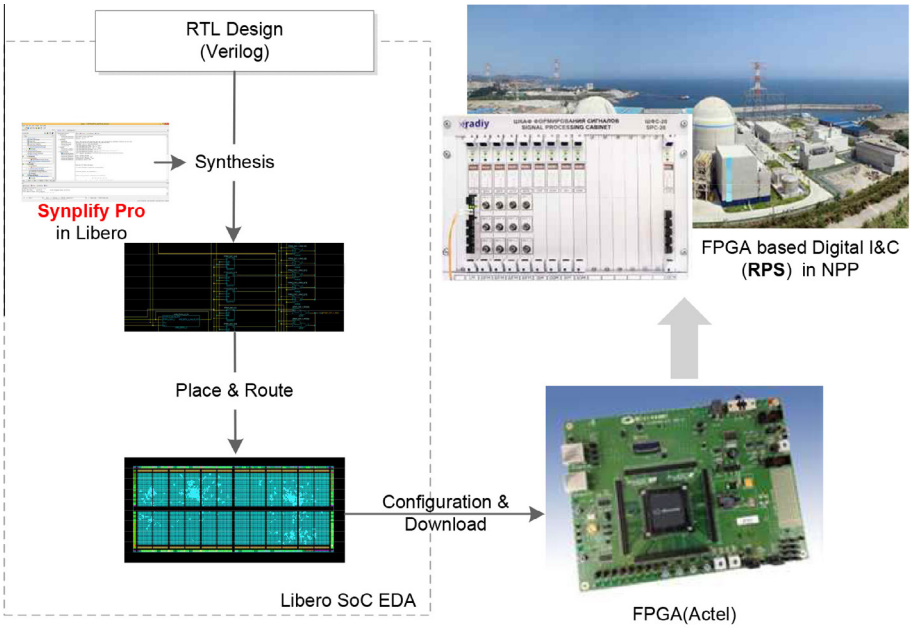

Fig. 7. The indirect COTS SW, EDA and its target system.

Table 5
Refined acceptance criteria for '*Synopsys Synplify Pro*'

Category		Refined definitions for '*Synopsys Synplify Pro*'	Selected methods	Selected V&V techniques
Functionality	Functionality	*The software should produce behaviorally-equivalent outputs from inputs as a compiler.* *The software should synthesize RTL design to gate-level design correctly.*	Method 1	Testing, Formal verification
SQA	QA V&V SCM	The same as before The same as before The same as before	Method 2	Review
Operating history	History Error tracking	The same as before The same as before	Method 4	Review

intervention. Its correct operation is a preliminary prerequisite for the safety of FPGA-based digital I&Cs. We tried to dedicate the indirect COTS SW in accordance with the proposed evaluation criteria and acceptance process in the order named.

5.1. Basic analysis

5.1.1. Identify a COTS SW to dedicate

The item this case study tries to dedicate is '*Synopsys Synplify Pro*'. It reads an RTL design programmed with HDLs, and is expected to produce a behaviorally equivalent gate-level design of netlists.

5.1.2. Is the SW being procured as a basic component?

No, to the best of our knowledge, there is no such tool certified in nuclear industry. '*Synopsys Synplify Pro*' does not get NQA-1 certification and was not developed under 10CFR50 App.B.

5.1.3. Does the SW perform a safety function?

No, it is a kind of compiler and does not perform any safety function defined by IEC 60880 (2006). It, however, produces a module which will be a system performing safety functions, and we regards it as an '*indirect COTS SW*', which performs the safety function of its target module (*e.g.,* RPS and ESF-CCS).

⇒ '*Synopsys Synplify Pro*' is determined to be an item which requires of the COTS SW dedication.

5.2. Identifying acceptance criteria

5.2.1. Determine safety category of COTS SW

The safety category of '*Synopsys Synplify Pro*' is determined in three steps. The safety category of its target system is first identified (2.1.1). The target system is a reactor protection system of the category '*A*'. The usage category of the COTS software is then identified (2.1.2) as '*indirect*'.

⇒ The safety category of the software is determined (2.1.3) as '*B*', since it is an indirect COTS SW with no information on source codes and its output could be verified by V&V techniques.

5.2.2. Review criteria for the safety category determined

We then review the acceptance criteria for the category '*B*' software. The '*B*' category includes 21 criteria as shown in Table 3.

5.3. Determining acceptance methods

5.3.1. Refine acceptance criteria in terms of target item

We now need to refine the acceptance criteria in terms of the target item – '*Synopsys Synplify Pro*' as shown in Table 5. For example, the acceptance criterion in *Functionality* – "*It shall be demonstrated that the COTS SW will fulfill its safety function.*" will be refined as "*The software should synthesize an RTL design to a gate-level design correctly.*" The acceptance criteria in *SQA* and *Operating History* are refined in terms of '*Synopsys Synplify Pro*', but they tend

to be refined slightly. "*The software should have appropriate quality*" and "*Supplier should manage the software configuration well*" are the examples of refined acceptance criteria of *SQA*.

5.3.2. Select acceptance methods

We then select acceptance methods for acceptance criteria in order to confirm that they are satisfied or not. Table 5 also includes the methods selected. We decided to use (Method 1) for the *Functionality* criteria, while (Method 2) and (Method 4) for the *SQA* and *Operating History*. As we cannot access to source codes of '*Synopsys Synplify Pro*', (Method 3) was out of our selection.

5.3.3. Identify Appropriate Techniques for Selected Method(s)

We have selected the (Method 1) to check the accurate and correct functioning of '*Synopsys Synplify Pro*' (*i.e.,* meaning "*correct synthesis from an RTL design into a gate-level design*"). However, we cannot use compiler verification techniques (Tony, 2003) which require detailed analysis on source codes. We, therefore, had to try the '*indirect verification*' to claim that.

> As the input and synthesized output are behaviorally-equivalent, the compiler works correctly at least for the input which will be used to be synthesized and downloaded into the new FPGA-based digital I&C (Yoo et al., 2015; Kim et al., 2014).

Table 5 also contains the selected techniques. We decided to use the equivalence checking (Huang and Cheng, 1998; Kim et al., 2016) in formal verification techniques to perform the indirect verification, and also selected a software testing technique (Kim et al., 2015) to complement it. In case of (Method 2) and (Method 4), we could look into technical documents provided by '*Synopsys*' in public by performing review techniques.

5.4. Dedication

5.4.1. Apply acceptance methods

[*Method 1. Special tests and inspections*] We used the '*CVEC (Customized VIS-based Equivalence Checking)*' (Kim et al., 2014; Kim et al., 2016) for the indirect verification (*i.e.,* equivalence checking), and also used '*IST-FPGA (Integrated Software Testing framework for FPGA)*' (Kim et al., 2015) for software testing as shown in Fig. 8. If the formal verification and software testing succeed, we can claim that '*Synopsys Synplify Pro*' worked correctly at least for the Verilog program (Choi and Lee, 2012) used.

The detailed explanation on the result of applying two verification techniques is out of the scope of this paper, but (Kim et al., 2016; Kim et al., 2015) explain them in details.

⇒ They could demonstrate that the target software '*Synopsys Synplify Pro*' worked correctly at least for the Verilog program (Choi and Lee, 2012) used.

[*Method 2. Survey of supplier*] We tried to survey the suppliers '*Synopsys*' and '*Microsemi*' to collect information regarding quality assurance plans and configuration management. We could find only the record of certification about ISO9001 and AS9100C (Microsemi, 2015), but the customer (*i.e.,* the host of COTS SW dedication) would be possible to access to all required information.

[*Method 4. Item/vendor performance*] We found records that '*Synopsys Synplify Pro*' was used to develop *Kozloduy* Nuclear Power Plants (Kozloduy Nuclear Power Plant, 2015) as *Radiy* platform (Radiy, 2015). It was also used for an alternative platform of ESFAS (Bachmach et al., 2010) for diversity. We also found a few history of update release, but could not find an error tracking report. It is worth to note that our survey does not indicate the absence of these information, but '*we*' could not find it on web-site.

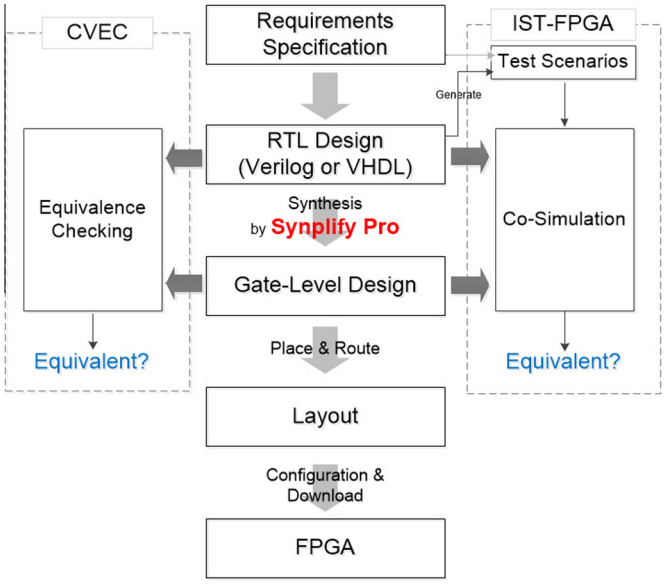

Fig. 8. CVEC and IST-FPGA for (Method 1).

5.4.2. Is the COTS SW acceptable?

We are finally to judge whether '*Synopsys Synplify Pro*' is acceptable or not on the basis of the acceptance criteria identified from (2.2). Table 6 lists the whole acceptance criteria for the category '*B*' COTS SW and also explains how they are satisfied or not by the acceptance methods performed at (4.1).

The dedication result of the target indirect COTS SW '*Synopsys Synplify Pro*' is out of the scope of this paper. We just tried to do the dedication from the view point of the 3rd party, in order to demonstrate the availability of the evaluation criteria and acceptance process which this paper proposes. The case study shows that the dedication process for COTS software can be usefully used to dedicate the indirect COTS software such as FPGA logic synthesis tools too, with no distortion from the existing standards and reports.

6. Related work

Table 7 reviews additional standards and technical reports for CGI (Commercial Grade Item) dedication. EPRI TR-017218 (1999) provides the guidelines on sampling COTS HW. It can be used to determine the number of samples to apply special tests. EPRI NP-6406 (1989) and TR-1008256 (2006) are supplementary guidelines for technical evaluation, and they consist of the first half of NP-5652. They also provide examples about performing technical evaluation for direct COTS SW and HW. TR-112579 (2000) provides seismic critical characteristics and assurance means to verify them. Its main target is direct COTS HW like PLC-based systems.

TR-107330 (1996) provides requirements for qualification of commercial PLC. It uses the criteria of NUREG/CR-6421. EPRI TR-104159 (1995) includes experience about dedicating COTS SW for PLC-based digital systems such as DAFAS (Diverse Auxiliary Feedwater Actuation System) and Emergency Diesel Bus Load Sequencer in accordance with EPRI NP-5652/TR-106439. It used checklists to demonstrate software quality, which were developed by domain experts. The examples used in TR-104159 are also digital systems using direct COTS SW. TR-1009659 (2005) provides examples about dedicating and qualifying direct commercial digital equipments like temperature controller and digital valve positioner.

In addition, there are also a few researches about applying dedication processes into COTS SW. Kim et al. (2000) proposed a COTS

Table 6
Acceptance criteria and results of applying acceptance methods

Category		Criteria (Refined)	Acceptance methods (V&V techniques)	Acceptance results
Functionality		It shall be demonstrated that the COTS SW will fulfill its safety function (→ *The software should produce behaviorally-equivalent outputs from inputs as a compiler)*	Method 1 (Formal verification – equivalence checking)	Indirectly Verifiable
		The COTS SW should be consistent with system requirements (→ *The software should synthesize RTL design to gate-level design correctly)*	Method 1 (Testing – random testing)	
SQA	QA	Are standards, practices, conventions, and metrics that were used, suggested described? Problem reporting, tracking and resolving described? Were Well-managed other supplier? If exists Were records of product generate and maintained? 25 kinds of V&V Tasks are performed?	Method 2 (Review – technical review, management review)	N/A (But, it may be provided by ISO9001 and AS9100C)
	V&V	Do V&V function detect errors early as possible? Are V&V function coordinated with the development?		
	CM	Does the plan describe responsibilities, authority, and relations between CM and development? At least one configuration control board is required Does the configuration management operation provide the following required functions? CM is founded upon the establishment of "configuration baselines" for each version of each product? Is the level of authority required for change described? Dose status accounting include? Software products under control for each supplier? All Records to be maintained and identified?		
Operating history	History	The COTS SW should have operated satisfactorily in similar applications	Method 4 (Review – technical review, management review)	Verifiable
		Configuration management and update should provide traceability		Verifiable
	Error Tracking	Error reporting, tracking and resolution should be consistent and correctly attributable to version and release is well managed		N/A
		The version and release have no major unresolved problems and bug list should be available to COTS purchaser as a support option		N/A

Table 7
Several standards and technical reports for CGI dedication.

Standards/TR	Subject	Contents
TR-017218 (NP-7218)	Sampling guidelines	Sampling guidelines for CGI dedication about COTS HW
NP-6406 (1989), TR-1008256 (2006)	Guidelines for technical evaluation	Explaining technical evaluation in the dedication process of NP-5652
TR-112579 (2000)	Guidelines for seismically sensitive items	Providing information about seismic critical characteristics and assurance means of verification
TR-107330 (1996)	Guidelines for qualification of commercial PLC	Providing and explaining qualification requirements about commercial PLC
TR-104159 (1995)	Empirical study of dedicating commercial PLC	Providing experience about progressing dedication process of commercial PLC-based systems
TR-1009659 (2005)	Lesson learned about dedication of digital items	Providing experience and information about dedicating several COTS digital equipments

SW dedication process based on NUREG/CR-6421 and used methods of TR-106439 to apply SQA identification. However, it does not mention about indirect SW and uses the method '*Survey of supplier*' only. Kim et al. (2010) performed the direct COTS SW dedication of *QNX* RTOS (Real-Time Operating System). It focused on the method 1 of NP-5652/TR-106439. Kim et al. (2007) also performed the direct COTS SW dedication of *PROFIBUS FMS-Driver* on the basis of NP-5652/TR-106439. '*TRICONEX*' (Triconex Approved Topical Report, 2012) of 'Invensys', a PLC-based system, was dedicated successfully in accordance with TR-106439. It used NUREG-0800 branched technical plan 7–18 (NUREG-0800 STR BTP 7-18, 2007), which uses criteria information of NUREG/CR-6421 and TR-107330, as our approach.

7. Conclusion and future work

This paper proposes an acceptance process and evaluation criteria, which are specific to direct/indirect COTS SW, not commercial-grade direct items. It provides categorized evaluation criteria for acceptance, and specifically incorporates indirect COTS SW. The grading/categorization provided will support appropriate selection of acceptance methods and V&V techniques, and also will be used as a firm basis of the final determination, since the process provides an explicit linkage between selected acceptance methods (V&V techniques) and evaluation criteria to be satisfied with. The process also provides a detailed guideline for selectable V&V techniques. The case study showed that it can be used usefully to dedicate the indirect COTS software such as FPGA logic synthesis tools, with no distortion but complementing existing reports/guidelines.

The indirect COTS SW dedication has an arguable issue that standards and organizations have different points of view in dedication targets such as V&V tools. NUREG/CR-6421 classifies software tools like testing, model checking and simulation as '*Unclassified*', while EPRI NP-5652/TR-1025243 classify some of them as '*Non-safety-related but augmented quality*'. On the other hands, international standards on functional safety such as

IEC-61508 and IEC-60880 require levels of safety demonstration of such supplementary software tools, too. We are now trying to analyze the proposed dedication process and evaluation criteria in terms of functional safety standards. We are also planning to extend the process from a security point of view.

Acknowledgements

This research was supported by a grant from the Korea ministry of science, ICT and future planning under the I&C Safety Conformance Assessment Platform. It was also supported by a grant from the Korea Atomic Energy Research Institute, under the development of the core software technologies of the integrated development environment for FPGA-based controllers.

References

Altera quartus 2, 2015. Altera, http://www.altera.com/products/software/ (2015.8)
Andryushin, A., Durnev, V., Chernyaev, A., 2014. Issues of operating systems usage for nuclear power plants. Ann. Nucl. Energy 70, 87–89.
Bachmach, E., Siora, O., Tokarev, V., Reshetytsky, S., Kharchenko, V., Bezsalyi, V., 2010. FPGA-based technology and systems for I&C of existing and advanced reactors. RADIY. IAEA-CN-164-7S04.
Bakhmach, I., Kharchenko, V., Siora, A., Sklyar, V., Andrashov, A., 2010. Experience of I&C Systems Modernization Using FPGA Technology. In: International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human-Machine Interface Technologies (NPIC & HMIT 2010): Proceeding of the 7th International Topical Meeting.–Las Vegas, Nevada, USA, pp. 1345–1352.
Brown, Stephen, Rose, Jonathan, 1996. FPGA and CPLD architectures: a tutorial. IEEE Des. Test Comput. 2, 42–57.
Burch, J.R., Clarke, E.M., Long, D.E., McMillan, K.L., Dill, D.L., 1994. Symbolic model checking for sequential circuit verification. IEEE Trans. Comput.-Aided Des. Integ. Circuits Syst. 13 (4), 401–424.
Bush, W.R., Pincus, J.D., Sielaff, D.J., 2000. A static analyzer for finding dynamic programming errors. Software-Practice Exp. 30 (7), 775–802.
Cadence, 2015. Encounter Conformal LEC. http://www.cadence.com/products/ld/equivalencechecker/pages/default.aspx/ (2015.10).
Choi, J.-G., Lee, D.Y., 2012. Development of RPS trip logic based on PLD technology. Nucl. Eng. Technol. 44 (6), 697–708.
Clarke, E.M., Grumberg, O., Peled, D., 1999. Model Checking. MIT Press.
Clarkson, G., 2008. FPGA based safety related I&C wolf creek generating station. In: 1st IAEA Workshop on Applications of Fieldprogrammable Gate Arrays in Nuclear Power Plants, France.
Collofello, J.S., The Software Technical Review Process.
Critical Characteristics for Acceptance of Seismically Sensitive Items (TR-112579), Tech. Rep. Electronic Power Research Institute.
Duffy, D.A., 1991. Principles of Automated Theorem Proving. John Wiley & Sons Inc.
Evaluating Commercial Digital Equipment for High-Integrity Applications (TR-106439), 1996 Tech. Rep. Electronic Power Research Institute.
Evaluation of Guidance for Tools Used to Develop Safety-Related Digital Instrumentation and Control Software for Nuclear Power Plants, Task 3 Report: Technical Basis for Regulatory Guidance, 2015 Tech. Rep. U.S. Nulcear Regulatory Research.
Experience with the Use of Programable Logic Controllers in Nuclear Safety Applications (TR-104159), 1995 Tech. Rep. Electronic Power Research Institute.
Fagan, M.E., 1986. Advances in software inspections. IEEE Trans. Software Eng. (7), 744–751
International Electrotechnical Commission (IEC), 1997. Functional safety of electrical/electronic/programmable electronic safety-related systems: Part 1. Generic requirements (IEC 61508-1), Tech. Rep.
General Qualification and Dedicatio of Digital Compoents – Project Status and Lessons Learned (TR-1009659), 2005 Tech. Rep. Electronic Power Research Institute.
Generic Requirements Specification for Qualifying a Commercially Available PLC for Safety-Related Applications in Nuclear Power Plants (TR-107330), 1996 Tech. Rep. Electronic Power Research Institute.
Guidance on the Use of Programmable Logic Controllers in Digital Computer-Based Instrumentatio and Control Systems (NUREG-0800 STR BTP 7-18), 2007. Tech. rep., U.S. Nuclear Regulatory Commision (NRC).
Guideline for Sampling in the Commercial-Grade Item Acceptance Process, 1999 Tech. Rep. Electronic Power Research Institute.
Guidelines for the Technical Evaluation of Replacement Items in Nuclear Power Plants (NP-6406), 1989 Tech. Rep. Electronic Power Research Institute.
Guidelines on the Use of Field Programmable Gate Arrays in Nuclear Power Plant I&C Systems (TR-1019181), 2009 Tech. Rep. Electronic Power Research Institute.
Hayashi, T., Kojima, A., Miyazaki, T., Oda, N., Wakita, K., Furusawa, T., 2014. Application of FPGA to nuclear power plant I&C systems. In: Progress of Nuclear Safety for Symbiosis and Sustainability. Springer, pp. 41–47.
Huang, S.-Y., Cheng, K.-T., 1998. Formal Equivalence Checking and Design Debugging. Kliwer Academic Publishers (Chapter 4).
Huang, S.-Y., Cheng, K.-T., Chen, K.-C., Huang, C.-Y., Brewer, F., 2000. AQUILA: An equivalence checking system for large sequential designs. IEEE Trans. Comput. 49 (5), 443–464.
IEEE Standard for Software Revies and Audits (IEEE 1028), Tech. Rep. 2008, Institute of Electrical and Electronics Engineers (IEEE).
IEEE Standard for System and Software Verification and Validation (IEEE 1012), Tech. Rep. Institute of Electrical and Electronics Engineers (IEEE).
Institute of Electrical and Electronics Engineers, 2001 Tech. Rep. IEEE Standard Verilog Hardware Description Languages.
Institute of Electrical and Electronics Engineers, 2008 Tech. Rep. IEEE Standard VHDL Language Reference Manual.
Functional safety of electrical/electronic/programmable electronic safety-related systems: Part 7. Overview of techniques and measures (IEC 61508-7), 1997 Tech. Rep. International Electrotechnical Commission (IEC).
Kelly, J.P.J., Murphy, S.C., 1990. Achieving dependability throughout the development process: a distributed software experiment. IEEE Trans. Software Eng. 16 (2), 153–165.
Kim, J.-Y., Lee, J.-S., Cheon, S.-W., Lee, K.-Y., Park, J.G., 2000. Commercial off-the-shelf software dedication process based on the commercial grade survey of supplier. In: Korean Nuclear Society, Taejon (Korea, Republic of) autumn meeting, 2000, in Korean.
Kim, J.-Y., Lee, Y.J., Cha, K.H., Cheon, S.-W., Lee, J.-S., Kwon, K.C., 2007. Experience on the COTS Software Dedication of the PROFIBUS FMS-Driver. In: Transactions of the Korean Nuclear Society Spring Meeting Jeju, Korea, May 10–11.
Kim, J.-Y., Lee, Y.J., Cheon, S.-W., Lee, J.-S., Kwon, K.C., 2010. A commercial-off-the-shelf (COTS) dedication of a QNX real time operating system (RTOS). In: 2010 2nd International Conference on Reliability, Safety and Hazard (ICRESH). IEEE, pp. 123–126.
Kim, E.-S., Yoo, J., Choi, J.-G., Kim, J.-Y., 2014. A correctness verification technique for commercial FPGA synthesis tools. In: Transactions of the Korean Nuclear Society Autumn Meeting, pp. 1986–1988.
KimJ., Kim, E.-S., Yoo, J., Lee, Y.J., Choi, J.-G., 2015. An integrated software testing framework for FPGA-based controllers in nuclear power plants. In: Nuclear Engineering and Technology, (Accepted).
Kim, E.-S., Yoo, J., Kim, J.-Y., 2016. CVEC: A customized VIS-based equivalence checking technique for commercial FPGA logic synthesis. In: Software Quality Journal, (submitted).
Kuehlmann, A., Krohm, F., 1997. Equivalence checking using cuts and heaps. In: Proceedings of the 34th annual Design Automation Conference. ACM, pp. 263–268.
Kuehlmann, A., Srinivasan, A., LaPotin, D.P., 1995. Verity – A formal verification program for custom CMOS circuits. IBM J. Res. Dev. 39 (1.2), 149–165.
Laski, J., Stanley, W., 2009. Software Verification and Analysis: An Integrated, Hands-on Approach. Springer Science & Business Media.
Mentor Graphics, 2015. Precision RTL. http://www.mentor.com/products/fpga/synthesis/precisionrtl/ (2015.10).
Mentor Graphics, 2015. ModelSim. http://www.mentor.com/products/fv/modelsim/ (2015.8).
Mentor Graphics, 2015. questa. htt://www.mentor.com/products/fv/questa/ (2015.8).
Microsemi, 2015. Microsemi-Certifications. http://www.microsemi.com/company/quality/certifications (2015.6).
Microsemi Libero SoC, Microsemi, http://www.microsemi.com/products/fpga-soc/design-resources/design-software/libero-soc (2015.10)
Nuclear power plants – Instrumentation and control systems important to safety – Software aspects for computer-based systems performing category A functions (IEC 60880), 2006 Tech. Rep. International Electrotechnical Commission (IEC).
Nuclear power plants – Instrumentation and control important to safety – Hardware design requirements for computer-based systems (IEC 60987), Tech. Rep. International Electrotechnical Commission (IEC).
Nuclear power plants – Instrumentation and control important to safety – Classification of instrumentation and control functions (IEC 61226), 2009 Tech. Rep. International Electrotechnical Commission (IEC).
Nuclear power plants – Instrumentation and control important to safety – General requirements for systems (IEC 61513), 2011 Tech. Rep. International Electrotechnical Commission (IEC).
Nuclear power plants – Instrumentation and control important to safety – Development of HDL-programmed integrated circuits for systems performing category A functions (IEC 62566), 2012 Tech. Rep. International Electrotechnical Commission (IEC).
Nuclear power plants – Instrumentation and control systems – Requirements for security programmes for computer-based systems (IEC62645), 2014 Tech. Rep. International Electrotechnical Commission (IEC).
NUREG/CR-6421: A Proposed Acceptance Process for Commercial Off-the-Shelf (COTS) Software in Reactor Applications, 1996 Tech. Rep. United States Nuclear Regulatory Commission (NRC).
NUREG/CR-7006: Review Guidelines for Field-Programmable Gate Arrays in Nuclear Power Plant Safety Systems, 1996 Tech. Rep. Nuclear Regulatory Commision (NRC).
Plant Engineering: Guideline for the Acceptance of Commercial-Grade Design and Analysis Computer Programs Used in Nuclear Safety-Related Applications (TR-1025243), 2013 Tech. Rep. Electronic Power Research Institute.
Plant Engineering: Guideline for the Acceptance of Commercial-Grade Items in Nuclear Safety-Related Applications (NP-5652), 2014 Tech. Rep. Electronic Power Research Institute.
Plant Support Engineering: Guidelines for the Technical Evaluation of Replacement Items in Nuclear Power Plants (TR-1008256), 2006 Tech. Rep. Electronic Power Research Institute.

Quality Verification of Commercial-Grade Items for replacing Safety-Related Items (KINS/RG-17.12), 2011 Tech. Rep. Korea Insitute of Nuclear Safety, in Korean.
Radiy, 2015. Radiy – Intelligent safety solutions proven in us. http://radiy.com/index.php/en/ (2015.6).
Recommended Approaches and Design Criteria for Application of Field Programmable Gate Arrays in Nuclear Power Plant Instrumentation and Control Systems (TR-1022983), 2011 Tech. Rep. Electronic Power Research Institute.
Regulatory Guide 5.71: Cyber Security Programs for Nuclear Facilities, 2010 Tech. Rep. United States Nuclear Regulatory Commission (NRC).
Santhanam, V., 2002. The anatomy of an FAA-qualifiable Ada subset compiler. ACM SIGAda Ada Letters, vol. 23. ACM, pp. 40–43.
SCC, Software Certification Consortium, http://cps-vo.org/group/scc.
She, J., Jiang, J., 2009. Application of FPGA to shutdown system No. 1 in CANDU. In: 6th American Nuclear Society International Topical Meeting on NPIC and HMIT, pp. 5–9.
Software and system engineering software testing Part 4: Test techniques (ISO/IEC/IEEE 29119-4), 2015. Tech. Rep., ISO/IEC/IEEE.
sook Jang, G., hyun Seong, D., yong Keum, J., youn Park, H., Kim, Y.-K., 2008. The design characteristics of an advanced alarm system for SMART. Annals of nuclear energy 35 (6), 1006–1015.
Survey of the CPLD/FPGA Technology for Application to NPP Digital I&C System, Tech. Rep. Korea Atomic Energy Research Institute.
Synopsys, 2015. Synopsys Synplify Pro. http://www.synopsys.com/Tools/ (2015.10).
The American Society of Mechanical Engineers, 2015. Nuclear Quality Assurance (NQA-1) Certification, https://www.asme.org (2015.10).
Tony, H., 2003. The Verifying Compiler: A Grand Challenge for Computing Research 50 (1), 63–69.
Triconex Approved Topical Report (7286-545-1-A)), 2012. Tech. rep., Invensys.
U.S. Code of Federal Regulations, Title 10, Part 50 App B: Quality Assurance Criteria for Nuclear Power Plants and Fuel Reprocessing Plants, Tech. Rep., U.S. Government Printing Office.
Wekipedia – Kozloduy Nuclear Power Plant, 2015. http://en.wikipedia.org/wiki/KozloduyNuclearPowerPlant (2015.6).
Xilinx ISE design suite, 2015. Xilinx. http://www.xilinx.com/products/ (2015.8).
Yoo, J., Kim, E.-S., Jung, S., 2015. Verification Techniques for COTS Dedication of Commercial FPGA Tools. In: The 10th International Symposium on Embedded Technology (ISET 2015), pp. 150–151.
Yoo, C.S., Seong, P.H., 2002. Experimental analysis of specification language diversity impact on NPP software diversity. J. Syst. Software 62 (2), 111–122.
Yoo, J., Lee, J.-H., Lee, J.-S., 2013. A research on seamless platform change of reactor protection system from PLC to FPGA. Nucl. Eng. Technol. 45 (4), 477–488.