디지털 원자로 보호계통의 소프트웨어 확인 및 검증

Software Verification & Validation for Digital Reactor Protection System

박기용, 권기춘
(Gee-Yong Park and Kee-Choon Kwon)

Abstract - The reactor protection system is the most important function for the safe operation of nuclear power plants (NPPs) in that such system protects a nuclear reactor core whose damage can cause an enormous disaster to the nuclear facility and the public. A digital reactor protection system (DRPS) is being developed in KAERI for use in the newly-constructed NPPs and also for replacing the existing analog-type reactor protection systems. In this paper, an software verification and validation (V&V) activities for DRPS, which are independent of the DRPS development processes, are described according to the software development life cycle. The main activities of DRPS V&V processes are the software planning documentations, the verification of software requirements specification (SRS) and software design specification (SDS), the verification of codes, the tests of the integrated software and system. Moreover, the software safety analysis and the software configuration management are involved in the DRPS V&V processes. All of the V&V activities are described, in detail, in this paper.

Key Words : Digital reactor protection system, Software verification and validation

1. Introduction

The existing protection systems of nuclear power plants in Korea are mostly implemented based on the outdating analog system. The modernization of instrumentation and control (I&C) system in nuclear field is now an important subject because of the obsolescence and the lack of spare parts of analog components. According to the many advantages of digitalization such as the availability improvement and the self-diagnosis function, it is being developed that a digital plant protection system and a safety-grade programmable logic controller(PLC) for the safety critical nuclear I&C systems through the Korea Nuclear I&C System (KNICS) projects. The PLC is being designed to provide various communication networks, a strict real time performance, and a high reliability. Thus, it is expected that DRPS and PLC being developed through the KNICS project can be applied to existing NPPs for the replacement or to new NPPs for the installation[1].

As is well known to conventional and also nuclear I&C systems, a computer system has a key role in a digitalized system. The software (S/W) in a computer system has to, therefore, be developed carefully based on the standard/regulatory guide for software development, e.g., Ref.[2] and Ref.[3], and verified rigorously for ensuring the safety and reliability of the S/W to be used in the safety-critical system such as a nuclear reactor protection system.

This paper presents an independent V&V process according to the software development life cycle for DRPS. The main activities in DRPS software V&V are the preparation of S/W V&V planning, the verification of S/W requirements specification (SRS) and S/W design specification (SDS), the verification of implementation codes, and the tests of the integrated software and integrated system. Also, they include the S/W safety analysis and the S/W configuration management(CM). The SRS V&V of DRPS are technical evaluation, licensing suitability evaluation, inspection and traceability analysis, formal verification, S/W safety analysis, and S/W configuration management. The SDS V&V of RPS are technical evaluation, licensing suitability evaluation, inspection and traceability analysis, formal verification, integrated S/W test plan, S/W safety analysis, and S/W configuration management. The code V&V are traceability analysis, source code inspection, test case and test procedure generation, S/W safety analysis, and S/W configuration management. Testing is a major V&V activity of software integration and system integration phase. The S/W safety analysis at SRS phase uses Hazard Operability (HAZOP) method, at SDS phase, it uses both HAZOP and Fault Tree Analysis (FTA), and, at implementation phase, it uses FTA.

2. Overview of DRPS

The DRPS has to protect the reactor core by dropping control rods into nuclear fuel assembly and shutting down the core when a certain severe event is occurred during the reactor operation. The DRPS has four channels and each channel is configured in electrically and physically isolated rooms. The DRPS generates the reactor trip and Engineered Safety Features (ESF) actuation signals automatically whenever the monitored variables reach the predefined setpoints. As is shown in Fig.1, each DRPS channel has two Bistable Processors (BPs), two Coincidence Processors (CPs), Automatic Test and Interface Processor (ATIP), and Cabinet Operator Module (COM). Signals comparison with corresponding set points, coincidence logic, and other operations are implemented in the safety-grade PLC (Programmable Logic Controller) that is developed specifically for DRPS.

The BP1 and BP2 in Fig.1 perform the same task by determining the trip state by comparing the measured process variable with the predefined trip set point. The CP1 and CP2 generate a trip signal by a two out of four voting logic. When a channel is bypassed, the trip signal is determined by a two out of three voting. Two processors in BP and CP are for facilitating the diversity. The ATIP performs the tests of the operational integrity of BPs and CPs and the interface with other processors and other channels. The COM comprises of two parts, i.e., a computer based part that provides the status information regarding the overall DRPS equipments and a hardware based part that performs protection-related controls such as a channel bypass and an initiation circuit reset. In the DRPS, there are three different networks such as ICN (Intra-Channel Network), SDL (Safety Data Link), and

ICDN(Inter-Channel Data Network). And All networks are configured via dual network structure for improving reliability.

The S/W development process for the DRPS is composed of several phases: plan, requirement, design, implementation, and test phases. The plan phase documents the plan and various analyses for development, management, and quality of S/W. In the requirement phase, the DRPS S/W requirements are specified in NuSCR, a formal software specification method [4]. The NuSCR has a formal syntax and semantics unlike natural languages and thus allows us to systematically generate PLC programs from the NuSCR requirements specification, which have the same behavioral aspects as the NuSCR specification. The systematic generation of PLC programs from NuSCR specification can reduce the possible errors occurring in the manual design specification, and also the software development cost and time. In the design phase, the operation contents are specified in more detail before the implementation. The S/W of control systems in NPPs is usually the embedded S/W. In the implementation phase, the code is generated by a aid of the pSET tool [1] that provides the environment for code generation, compilation, and loading into corresponding PLC. The test has an increasing step-by-step procedures: S/W module test, integrated S/W test, and integrated system test.

3. Preparation of Verification & Validation

According to BTP HICB-14 of NUREG-0800 Standard Review Plan (SRP) [2][5], the information to be reviewed is subdivided into three topic areas: Software Life Cycle Process (SLCP) planning, SLCP implementation, and SLCP design outputs. Due to the importance of planning, the following documents are decided to be prepared: the S/W V&V plan, the S/W quality assurance (QA) plan, and the S/W safety plan, all from V&V activities, and the S/W management plan, the S/W development plan, the S/W integration plan, the installation plan, and the S/W CM plan. The S/W V&V plan is strongly related to the S/W development plan in that software V&V activity depends on which process a software development project follows. The software V&V plan addresses the issues of team organization, master schedule, software integrity level, management of V&V, life cycle V&V activities, V&V reporting and documentations, etc.

The V&V process provides an objective assessment of software products and processes throughout the software life cycle. We have developed V&V procedures for software requirement, design, implementation, and integration phases. The V&V for DRPS software is performed based on the V&V procedures in each phase. The V&V procedures provide specialized checklists for the life cycle V&V tasks, and define V&V methods and their supporting tools, and inputs and outputs. It also provides procedures for formal verification. Test procedures are also included within the V&V procedures because regulatory position regards software tests as a V&V activity.

4. Verification & Validation Activities for DRPS S/W

One of the main purposes of DRPS software V&V is to acquire a license from the nuclear regulatory authority. Thus it is crucial for the V&V process to meet regulatory requirements as well as design goals. To meet nuclear regulatory requirements and design goals, software V&V criteria and requirements are based mainly on Ref.[5] to Ref.[12] and the others.

4.1 Software Requirement Phase

The first V&V activity in software requirement phase is the licensing suitability review. The purpose of the licensing suitability review determines whether the S/W requirements for the DRPS are suitable for the code & standard and technological viewpoint. According to Ref.[2] and Ref.[9], The DRPS S/W requirements must satisfy both the functionality characteristics (safety, robustness, security) and the process characteristics (completeness, consistency, correctness, readness, traceability, verifiability). The second V&V activity in this phase is Fagan inspection [13]. The Fagan inspection is a formal review process developed by Michael Fagan at IBM in the 1970s. The inspection technique is being applied for overall phases of software life cycle due to its applicability to coding. The inspection process consists of seven steps: planning, overview, preparation, inspection meeting, rework, and follow up. Inspection team is composed of a moderator, recorder, reader, author, and inspector. The third activity is the traceability analysis. Throughout the software life cycle, a software requirements traceability analysis will be performed and a requirements traceability matrix will be maintained for the DRPS software. We are performing the inspection and traceability analysis focused on the completeness, correctness and consistency. The Fagan inspection is supported by the Software Inspection Support-Requirement Traceability (SIS-RT) tool that was developed in the KNICS project. The fourth activity of requirement phase V&V is formal verification. As mentioned before, DRPS software requirements are specified in a formal specification method of NuSCR. The NuSCR makes it possible to easily perform formal verification such as theorem proving and model checking. Model checking is a method for formally verifying finite-state concurrent systems. Specifications about a software system are expressed as temporal logic formulas. Efficient symbolic algorithms are used to traverse the model defined by the system and check if the specification holds or not. Theorem proving specifies the system, a required property, the assumptions, and necessary background theories as formulas in a single logic. It proves that background, assumptions and system are a model of the property. The NuSCR specification can be mechanically translated into input languages of a theorem prover and a model checker. We are performing theorem proving using Prototype Verification System (PVS) and model checking using Symbolic Model Verifier (SMV).

4.2 Software Design Phase

The V&V activities in software design phase are almost the same as in section 4.1. Formal verification is a little different because the DRPS S/W development process does not introduce NuSCR into S/W design specification. Instead it adopts function block diagrams to specify software design. Thus we have developed a technique to translate function block diagrams into the input languages of model checkers. Using the technique, we are performing model checking in software design phase.

4.3 Software Implementation Phase

In implementation phase, we are also performing the licensing suitability review, Fagan inspection, and traceability analysis. However, in this phase, the main activity is the test. Fig.2 shows the S/W test life cycle including component test, integration test, system test, and acceptance test. Each test follows the S/W test life cycle tasks (e.g., generations of test plan, test design, test case, test procedure, and test execution). The component test of the DRPS S/W is challengeable because test techniques for PLC S/W are not mature yet.

4.4 Software Integration Phase

The integration phase can be divided into the software integration phase and the system integration phase. The integration test, the main test in this phase, is an orderly progression of tests for incremental pieces of S/W systems where S/W elements are combined and tested until the S/W has been integrated to show a compliance with the S/W design requirements. In system integration phase, the main V&V activities are the system test. The system test is the activities of testing an integrated hardware and software system to verify and validate whether the system meets its original objectives. Boundary value test and equivalence partitioning techniques are applied to the DRPS S/W system test.

4.5 Safety Analysis

The V&V activities include the S/W safety analysis (SA) to improve S/W safety as well as quality. Since the licensing criteria requires the SA of the product from each phase of the S/W life cycle, the software SA procedure for each phase has been developed. The procedures include HAZOP procedures for the requirement and design phases, and FTA procedures for the design and implementation phases. The HAZOP has been suggested for SA in the S/W requirement phase. HAZOP is a powerful hazard analysis technique which has a long history in process industries. As the use of digital systems for nuclear power plants becomes more common, it is clear that there is a need for a HAZOP method which can be used effectively with such systems. We develop the guide phrases, checklists, and the software HAZOP procedure for DRPS.

The FTA is one of the most widely used methods in system reliability analysis. One of the advantages of it is that safety engineers are already familiar with it. FTA was primarily used for safety analysis of hardware systems. The software FTA was proposed almost twenty years ago and since that time, the technique has been refined for analyzing the safety of software designs. We have proposed an approach of the FTA for the function block diagram (FBD). The conditions that can contribute to an undesired system state in FBD designs must be analyzed to prevent and control/eliminate those conditions. The proposed safety analysis technique identifies the safety aspects of the FBD-based specifications with a combined template-based fault tree analysis approach.

4.6 Configuration Management

The Software Configuration Management (SCM) is an activity which configures the product form of a system (documents, programs, and hardware) and systematically manages and controls modifications during software development and maintenance. This process improves the quality of the software and is highly correlated with the reliable software development. A software configuration management tool, NuSCM, has been designed for software life cycle V&V management of the KNICS DRPS software. NuSCM is based on the systematic management of software design documents and source codes based on projects and activities. Since NuSCM is based on the internet, it provides an user with the easy accessibility that maximizes the efficiency in carrying out tasks.

5. Conclusions

This paper has discussed the KNICS approach to the S/W life cycle V&V for DRPS software. Through the KNICS project, the V&V process for NPP safety software is being established. The KNICS approach involves structured checklists, V&V procedures, specialized V&V techniques and their tools. Representative V&V techniques include licensing suitability review, Fagan inspection, traceability analysis, model checking, theorem proving, and various tests. The main features of the KNICS software V&V process are summarized as follows.

- Strict compliance with the related codes & standards.
- Various V&V activities in each S/W development phase.
- Combined approach between informal and formal verification methods.
- New and self-developed V&V and test techniques.

Through these V&V activities, we can achieve the functionality, performance, reliability, and safety that are V&V objectives of nuclear safety software in KNICS project.

References

[1] J. B. Han, et al., "Development of a Safety Grade System for KNICS," The 25th KAIF-JAIF Seminar on Nuclear Technology, Seoul, Korea, October 20-21, 2003.

[2] USNRC, Branch Technical Position HICB-14, Guidance on Software Reviews for Digital Computer-Based Instrumentation and Control Systems. June 1997.

[3] IEEE Std. 1074, IEEE Standard for Developing Software Life Cycle Processes, 1997.

[4] J. B. Yoo, et al., "A Formal Software Requirements Specification Method for Digital Nuclear Plants Protection Systems," J. of Systems and Software, accepted.

[5] USNRC, NUREG-0800, Standard Review Plan, Chapter 7, July 1997.

[6] USNRC, Reg. Guide 1.170, Software Test Documentation for Digital Computer Software Used in Safety Systems of Nuclear Power Plants, 1997.

[7] USNRC, Reg. Guide 1.171, Software Unit Testing for Digital Computer Software Used in Safety Systems of Nuclear Power Plants, 1997.

[8] IEEE Std. 1012, Standard for Software Verification and Validation Plans, 1997.

[9] IEEE Std. 7-4.3.2, IEEE Standard for Digital Computers in Safety Systems of Nuclear Power Generating Stations, 1993.

[10] IEEE Std. 1008, Standard for Software Unit Testing, 1987.

[11] IEEE Std. 829, Standard for Software Test Documentation, 1983.

[12] IEEE Std. 1028, Standard for Software Reviews and Audits, 1998.

[13] M. E. Fagan, "Design and Code Inspections to Reduce Errors in Program Development," IBM Systems Journal, Vol. 15, No. 3, 1976.

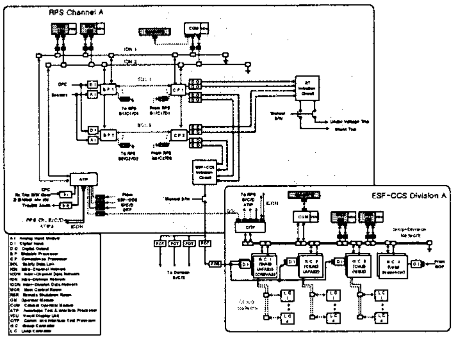

Fig. 1 Architecture of KNICS DRPS

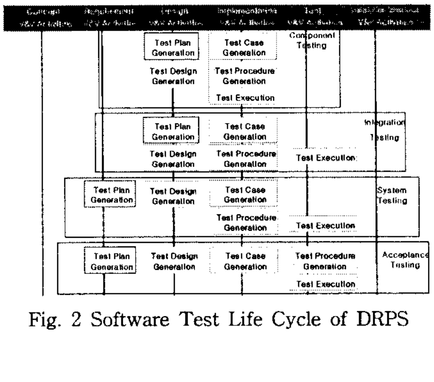

Fig. 2 Software Test Life Cycle of DRPS