Proceedings of the 6th WSEAS International Conference on Applied Informatics and Communications, Elounda, Greece, August 18-20, 2006 (pp353-358)

Software Qualificaiton of a Programmable Logic Controller for Nuclear Instrumentation and Control Applications

KYOUNGHO CHA, JANGYEOUL KIM, JANGSOO LEE, SEWOO CHEON, KEECHOON KWON
Instrumentation&Control and Human Factors Division
Korea Atomic Energy Research Institute
150 Deokjin-Dong, Yuseong-Gu, Daejeon 305-353
REPUBLIC OF KOREA

Abstract: Software qualification includes such activities as a software Verification and Validation (V&V), a software safety analysis, a software configuration management and a software quality assurance for the safety-critical applications in Nuclear Power Plant (NPP). This paper presents the software qualification of a safety grade Programmable Logic Controller (PLC) which is applied to a Reactor Protection System prototype. The software V&V is characterized by defining the inputs, tasks, and outputs for all the software life cycle phases defined in the software V&V plan, and the V&V techniques such as a checklist-based review and the Fagan Inspection, a traceability analysis, a formal verification, and a software test are applied to improve the software quality. The software safety analysis process, which employs the HAZard OPerability (HAZOP) methodology, has been developed and applied to improve the software safety. All the software documents and source codes are managed as software configuration items throughout the software life cycle under the control of a software quality assurance plan and procedure. Automated software tools and a 3rd part review also support the activities for the software qualification. Our experience shows that the software qualification is very efficient for systematically qualifying the safety-critical software of a PLC to be embedded in the safety-critical systems of a NPP, and they can be easily extended to other safety-critical applications such as in the railways, military, medicine, etc.

Key-Words: Nuclear software qualification, software verification and validation, Software safety analysis, Software configuration management, Embedded software, Programmable logic controller

1 Introduction

Recently, there has been growing interest in a software qualification for applying a Programmable Logic Controller (PLC) to nuclear Instrumentation and Control (I&C) systems including a Reactor Protection System, since digital I&C platforms have been prototyped and they embed not only system software such as a real-time operating system or real-time executives but also application software such as Function Block Diagrams (FBDs) for the I&C functions of a Nuclear Power Plant (NPP). It is important that the software should meet the nuclear codes and standards of a software verification and validation (V&V), a software safety analysis (SSA) and a software configuration management (SCM) if software is to be applied to the safety I&C systems in a NPP. The nuclear codes and standards for a software qualification are BTP HICB-14 of NUREG-0800 SRP [6], Regulatory Guide 1.168 [7], Regulatory Guide 1.152 [8], IEEE Std 7-4.3.2 [9], IEEE Std 1012 [10], etc. Digital I&C platforms such as Teleperm XS, Common Qalified (Common Q) Platform, and Triconex's TRICON have been evaluated for nuclear safety applications by the office of US Nuclear Regulation Council (USNRC) [1]. During the design of the digital I&C platforms, much effort has been given to a V&V including a software test.

The main objective of our research is to develop a software qualification where the POSCON Safety Grade PLC (called POSAFE-Q) software is rigorously qualified to meet the nuclear codes and standards. Software verification techniques such as the Fagan Inspection and a formal verificaiton are applied to the software requirements and design in support of the NuSEE [2] and Statemate MAGNUM Model Checker/Certifier tools. A component test, an integraiton test, and a system test are also performed to validate software codes and each test follows the the software testing life cycle (STLC) which consists of a test plan generation, a test design generation, a test case generation, a test procedure generation, and a test execution. HAZOP (HAZard OPerability) methodology is specially adopted to define the SSA processes for the POSAFE-Q software. All the

software documents and source codes are managed as software configuration items for the life cycle under the quality assurance, and the NuSCM integrated within the NuSEE tool supports the SCM tasks.

Through our experience for the nuclear qualification of the POSAFE-Q software, we affirm that a V&V, an SSA, and a SCM should be applied systematically to assure a software quality and improve a software safety and reliability for the nuclear safety I&C systems. This software qualification approach can be expanded systematically into other safety-critical applicaitons such as in the railways, military, medicine, and so on. We briefly describe and classify the POSAFE-Q software in section 2, detail the software qualification approach for the POSAFE-Q in section 3, show the experimental results in section 4, and conclude in section 5.

2 Software Description of POSAFE-Q

The POSAFE-Q has been designed and qualified for an IDiPS prototype, and the IDiPS software (i.e., IEC 61131-3 FBD for reactor protection logics) is embedded within the POSAFE-Q. Fig.1 shows the POSAFE-Q architecture which is highly reliable and fault-tolerant, highly qualified, highly maintainable, and so on [20].

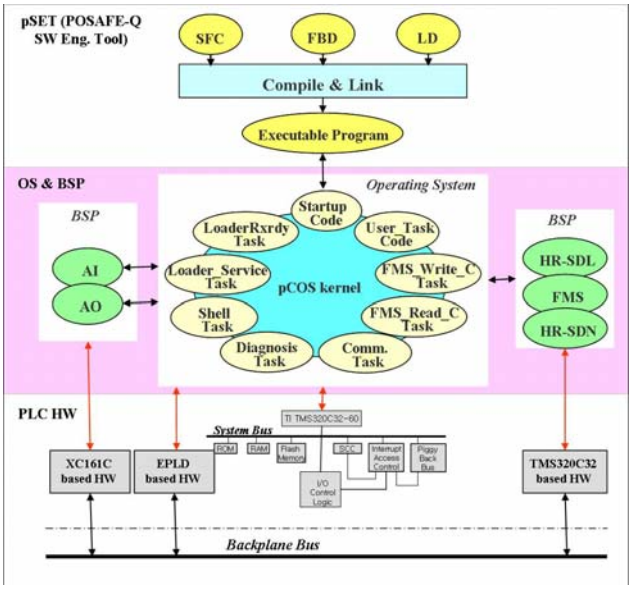


Fig. 1 POSAFE-Q architecture

The POSAFE-Q consists of a processor module, communication modules, and I/O modules. A pCOS pCOS kernel and system tasks are embedded in the processor module, pNMOS/pNDOS executives are embedded in the communication modules, and pIAOS/pOAOS execitives are embedded in the I/O modules. The IDiPS software (i.e., FBDs) are developed by using a POSAFE-Q software engineering tool (called pSET) and downloaded into memories embedded within the POSAFE-Q. All the software shown in Fig. 1 was classified as safety grade shown in Table 1.

Table 1 Software classification

HW modules	HW components	SW Components	SW Safety Grade
Processor Module	CPU	pCOS Kernel, BIHS	Safety Critical (SC)
		System Tasks	SC
Comm. Module	HR-SDL	pNMOS4, pNDOS4	SC
	HR-SDN	pNMOS6, pNDOS6	Safety Related (SR)
	Profibus-FMS	pNMOS1, pNDOS1	SC
I/O Module	Analog Input	pIAOS1	SC
	Analog Output	pOAOS1	SC
	RTD	pIAOS2	SC
	TC	pIAOS3	SC
	PC, Digital I/O	(TBD)	SC
pSET (SW tool for IEC 61131-1 FBD programming)		Compiler, Loader	SR
		Editor, Debugger	Non Safety (NS)
		Simulator	NS

All the software for the POSAFE-Q in Table 1 has been developed by following a well-defined software life cycle process such as a watefall model and a spiral model. A V&V, an SSA, and an SCM should be applied differently to each safety grade. The V&V, SSA, and SCM processes were defined and applied for the SC software and the SR does not require an SSA.

As the pCOS kernel is the central control component for an execution of the IDiPS applications, a task scgeduling is handled very carefully in the design and V&V. The IDiPS software are sheduled by the pCOS kernel, as shown in Fig.2.

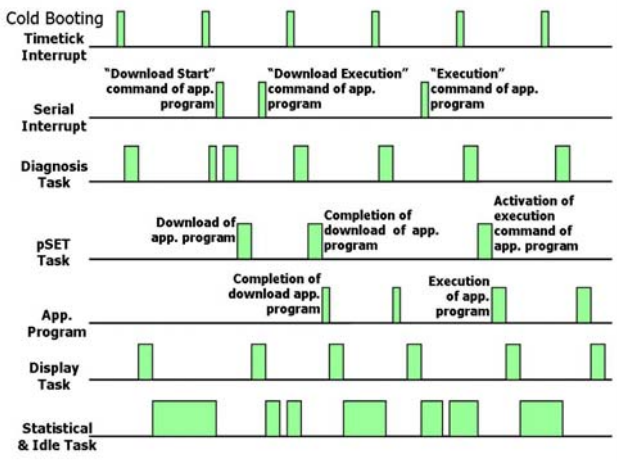


Fig. 2 Deterministic task scheduling by pCOS

3 Software Qualification

The POSAFE-Q has been qualified for the IDiPS. The software V&V plan, as a compliance of the BTP HICB-14 of NUREG-0800 SRP [6] and IEEE Std. 1012-1998 [10], was written for the life cycle software V&V of the POSAFE-Q. The SSA plan was prepared as a compliance of the IEEE Std. 1228-1994 [12] and the SCM plan was written as a compliance of the IEEE Std. 828-1998 [13]. Commercial-Off-The-Shelf (COTS) software were used for the communication modules of the POSAFE-Q and the COTS dedication plan and its procedure were developed and applied for the V&V of them. Fig. 3 illustrates the relationship of the activities among the software regulator, software developer, and software verifier/validator. In this paper, we limit the software qualification to the V&V, the SSA, and the SCM for product assurance.

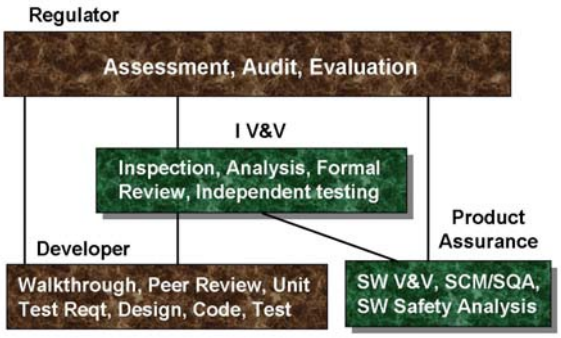


Fig. 3 Activities among nuclear software regulation, development, and qualification

3.1 Software V&V

The SC software should be rigorously verified and validated to meet the nuclear codes and standards. Thus, we defined the software V&V tasks for the life cycle in Fig. 4.

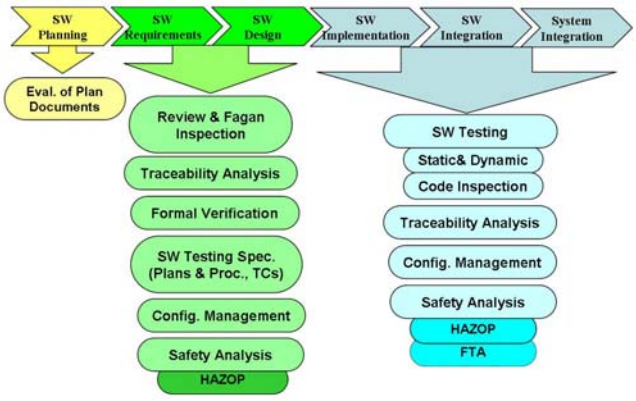


Fig. 4 V&V tasks for softwre life cycle

Checklists were developed and structured in the software V&V procedures. The V&V procedures were also developed and applied for each life cycle phase. They provided the required inputs and outputs, the specialized V&V procedures, the V&V methods and techniques including the V&V tools. Review and the Fagan Inspection were applied to verify overall outputs through the life cycle phases. Traceability analysis and inspection tasks were partially supported by a Software Inspection Support and Requirement Traceability (SIS-RT) integrated in NuSEE [2]. Fig. 5 shows a part of the traceability analysis of the software requirements for the pCOS kernel.

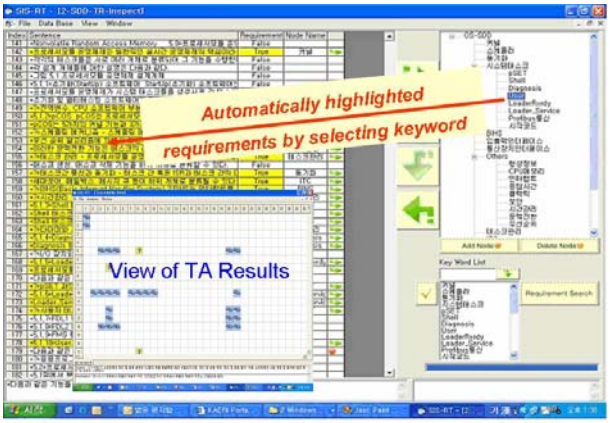
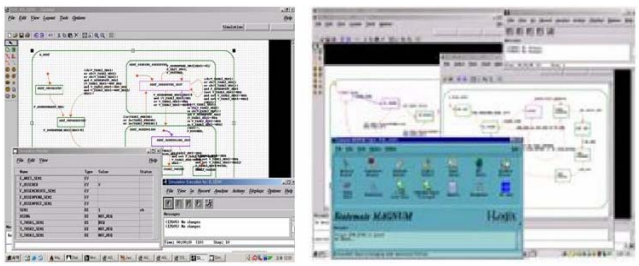


Fig. 5 Traceability analysis using the SIS-RT

Formal verification (FV) was applied to find a missing requirement or an ill-specified requirement in the software design. Statemate MAGNUM's ModelChecker was used to detect syntactic errors and Statemate MAGNUM's ModelCertifier was adopted to verify semantic errors in the formal specifications of the statecharts-based software requirements and the design. Fig. 6 shows a process of a formal verification for the pCOS kernel.



(a) Model checking (b) Model verification

Fig. 6 Formal verification using the Statemate MAGNUM's Model Checker/Model Certifier

The 3rd party review by ISTec was processed for qualifying the POSAFE-Q pCOS kernel and system tasks in accordance with the IEC and IEEE standards such as IEC 60880, IEEE 1012-1998, and so on [19]. Fig. 7 shows the software type test procedure used for the 3rd review.

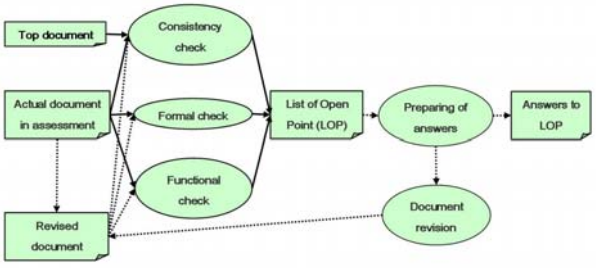


Fig. 7 Software type test procedure by ISTec

3.2 Software Testing

Software test consisted of a component test, an integration test, a system integration test, and an acceptance test. Software test for the life cycle (STLC) consisted of a test plan generation, a test design generation, a test case generation, a test procedure generation, and a test execution generation. The STLC was applied to each test of the SC software. McCabe TEST, Cantata++, and Tessy tools were utilized to automate or support the STLC tasks. McCabe TEST and Cantata++ tools were also utilized to inspect the source codes written by C Language. A CPLD/FPGA design is to be simulated and verified by using ModelSim tool. Fig. 8 illustrates the test environment for the POSAFE-Q software.

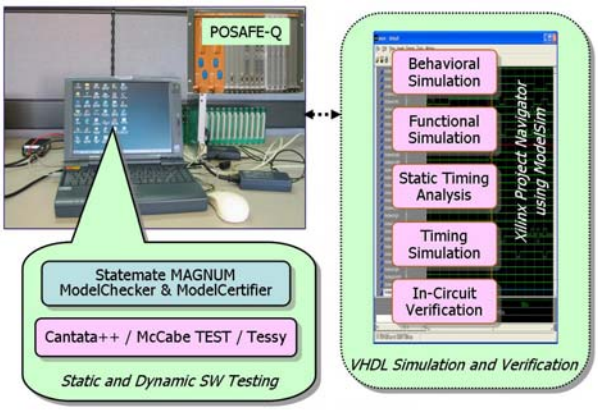


Fig. 8 Test environment

Component tests were performed for all of the SC software. The STLC tasks for the component tests were automated by using the McCabe TEST and Cantata++ tools. Fig. 9 shows a part of the process of the component test for pCOS kernel [3].

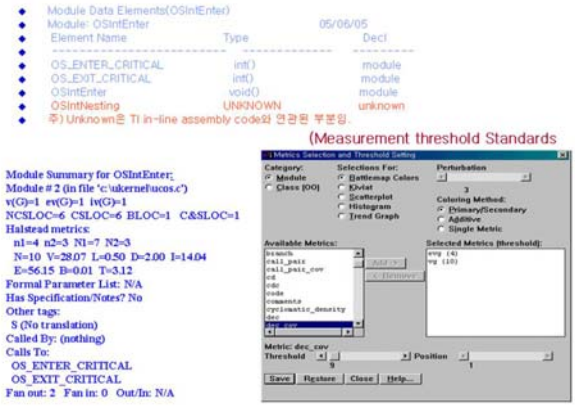


Fig. 9 Component test by using the McCabe TEST

3.3 SSA

The SC software should be analized for a software safety. We developed an SSA process and it was applied to the life cycle of the SC software [5]. The SSA process uses a HAZOP and Fig. 10 shows it.

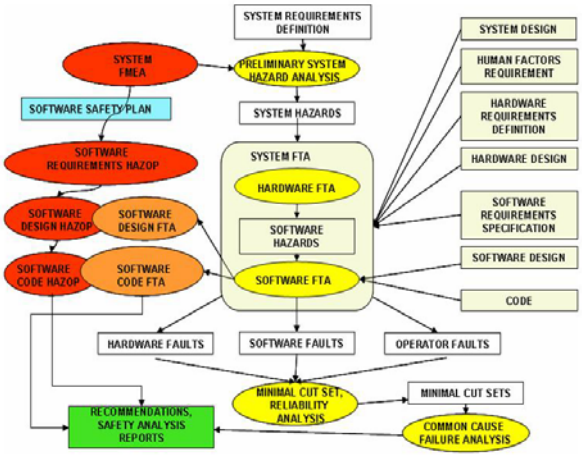


Fig. 10 SSA Process

Specialized checklists consisted of guide phrases and the guide phrases were developed for the SSA process. Table 2 shows an example of the checklists for the SSA.

Table 2 Example of the HAZOP checklist

Guide Phrases	Deviation checklists	Cause	...	Risk level	Recommendation
Timer can't operate	*Which type of risk occurs if timer can't operate?*	Initiation failure	...	Very high (4)	...
...	...	...	...	...	...

3.4 SCM

SCM configures the software configuration items such as the software documents, drawings and source codes throughout the life cycle. The software configuration items were managed by the NuSCM tool, which was developed for the SCM of the POSAFE-Q. Fig. 11 shows a part of the SCM process.

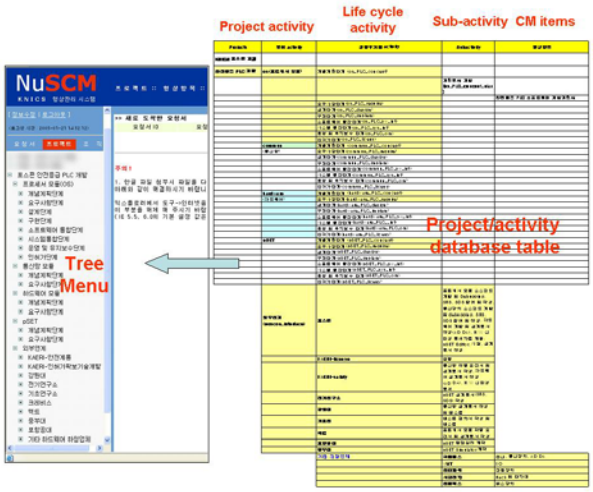


Fig. 11 SCM by using the NuSCM tool

4 Results

We found and corrected many errors during the software qualification of the POSAFE-Q. These faults and errors are summarized as follows.

4.1 Software V&V

Software requirements were revised due to their incorrect, incomplete, and inconsistent specifications. Software requirements were correctly refined for deadlock management of the pCOS kernel, while the requirements of the pIAOS1 and pOAOS1 software were revised because the software classification and security requirements were missing. The source programs were also modified because coding errors were found for the parameter names and variable values by the component and integration tests. Regression test was applied to validate the modified source programs. Inconsistency between the test plans and the test procedures were reported as an anomaly status.

4.2 SSA

SSA was applied to the safety features or proprties of the SC software. Some of them were selected from the results of the V&V and the selected ones were analized for their risks and hazards to the safey functions of the IDiPS. Besides the selected safety features found from the results of the V&V, a few additioanl safety features were found and analized by the HAZOP. One of them was a weak function for a self-diagnosis.

4.3 SCM

SCM, in cooperation with quality assurance, was performed during the life cycle phases. Inconsistency among the SCM items were found to be the date and revision number of software documents. Some of the reported anomalies should be resolved throughout the revision of software documents.

5 Conclusions

We presented the software qualification which a software V&V, a software safety analysis, and a software configuration management were applied to a safety grade PLC for nuclear I&C applications. The presented software qualificaiton could be applied efficiently and sufficiently to a combined software lifecycle process by following the Waterfall and Spiral models of the safety grade POSAFE-Q for the IDiPS prototype. Software V&V was ssuccessfully performed as a compliance with the nuclear codes and standards such as the regulatory guides, IEEEs and IECs. It was important to apply a checklist-based review and the Fagan Inspection because many errors were found by them. Especially, the 3rd part review was also valuable because the ISTec evaluated technically with the experience of Teleperm XS. In conclusion, our experience shows that the applied techniques and supporting tools used in this study are very efficient to qualify the PLC software for the safety-critical instrumentation and control systems in nuclear power plants, and the approach can be extended easily to other safety-critical applications such as in the railways, military, medicine, etc.

Acknowledgement

The work, performed for “development of licensing technology of digital I&C” as part of Korea Nuclear Instrumentation and Control (KNICS) project, is being supported by the Ministry of Commerce, Industry and Energy in Korea since the Ministry of Science and Technology had funded for the work from 2002 to 2004.

References:

[1] J. NASER, Qualification, Acceptance, and Implementation of Programmable Logic Controller- Based Platforms for Safety-Related Applications in Nuclear Power Plants, *Proceedings on NPIC&HMIT 2000,* Washington, DC, 2000.

[2] S.R. Koo, H.S., Son, P.H. Seong, J.B. Yoo, S.D. Cha, D.S. Son, and S.S. Choi, Development of Software Requirement Analysis Tool for NPP Software Fields Based on Software Inspection and Formal Method, *Proceedings of the 3rd Annual Conference of the International Symposium on the Future I&C for NPP (ISOFIC 2002)*, Nov. 7-8, 2002, pp.159-164.

[3] J.Y. Kim, S.W. Cheon, J.S. Lee, Y.J. Lee, K.H. Cha, and K.C. Kwon, Software V&V Methods for a Safety Grade Programmable Logic Controller, *Proceedings of the International Conference on Reliability, Safety and Hazards-2005*, Dec. 1, 2005.

[4] S.W. Cheon, J.S. Lee, K.C. Kwon, D.H. Kim, and H. Kim, The Software Verification and Validation Process for a PLC-based Engineered Safety Features-Component Control System in Nuclear Power Plants, *Proceedings of the 3rd Annual Conference of the IEEE Industrial Electronics Society*, Nov. 2-6, 2004, pp. 827-831.

[5] J.S. Lee, J.Y. Kim, H.S. Son, Y.J. Lee, S.W. Cheon, K.H. Cha, and K.C. Kwon, Safety Analysis Process for KNICS Safety Software, *Proceedings of the 3rd KNS-KIEE Joint Workshop on I&C Technology*, Nov. 14, 2003, pp.317-328.

[6] NUREG-0800, Standard Review Plan (Chapter 7), *USNRC*, 1997.

[7] Regulatory Guide 1.168, Verification, Validation, Reviews, and Audits for Digital Computers used in Safety Systems of Nuclear Power Plants, *USNRC*, 2004.

[8] Regulatory Guide 1.152, Criteria for Programmable Digital Computers System Software in Safety Related Systems of Nuclear Power Plants, *USNRC*, 1996.

[9] IEEE Std. 7-4.3.2, IEEE Standard for Digital Computers in Safety Systems of Nuclear Power Generating Stations, 2003.

[10] IEEE Std. 1012, IEEE Standard for Software Verification and Validation Plans, 1998.

[11] IEEE Std. 1028, IEEE Standard for Software Review and Audits, 1997.

[12] IEEE Std. 1228, IEEE Standard for Software Safety Plans, 1994.

[13] IEEE Std. 828, IEEE Standard for Software Configuration Management Plans, 1998.

[14] IEEE Std. 830, IEEE Recommended Practice for Software Requirements Specifications, 1998.

[15] IEEE Std. 1016, IEEE Recommended Practice for Software Design Descriptions, 1998.

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

[17] IEEE Std. 829, IEEE Standard for Software Test Documentation, 1998.

[18] J.J. Labrosse, *MicroC/OS-II: The Real-Time Kernel (Korean Edition)*, CMP Media, Inc., 2002.

[19] E. Hoffmann and A. Linder H. Miedl, Qualification of the POSAFE-Q RTOS (Phase A), ISTec-A-905, 2004.

[20] POSCON, Design Requirements of Safety Grade POSAFE-Q (in Korean), KNICS-PLC-DS301, 2005.