2004 춘계학술발표회 논문집
한국원자력학회

PLC-기반 디지털 원자로안전계통 시제품을 위한 운영체제 소프트웨어의 요구사항 확인검증

Requirements Verification and Validation of Operating System Software for a PLC-based Plant Protection System Prototype

차경호, 이영준, 천세우, 손한성, 김장열, 이장수, 권기춘
한국원자력연구소
대전광역시 유성구 덕진동 150

요 약

본 논문은 원전계측제어시스템(Korea Nuclear Instrumentation and Control System: KNICS) 사업단에서 개발중인 안전등급 제어기기(Programmable Logic Controller: PLC)-기반 디지털 원자로안전계통 시제품의 운영체제 소프트웨어에 대한 요구사항 확인 및 검증(Requirements Verification and Validation)을 기술한다. 운영체제 소프트웨어는 신개발 소프트웨어(Newly Developing Software: NDS)로써 생명주기 확인검증(Lifecycle Phases V&V)이 적용되며, NUREG-0800 의 Software Review Plan(SRP)/BTP-14 와 IEEE Std. 7-4.3.2 이 요구하는 소프트웨어 확인검증 기준들과 IEEE Std. 1012 와 IEEE Std. 1028 의 소프트웨어 확인검증 요구들이 생명주기 각 단계마다 엄격하고 체계적으로 적용된다. 요구사항 확인검증은 체크리스트-기반 Fagan 인스팩션이 주로 적용되었으며, 그리고 모델체킹에 의한 정형검증과 HAZOP(HAZards and OPerability) 방법에 의한 안전 요구사항의 확인이 수행되어 왔다. 체크리스트로 구성된 검증절차가 체계적인 요구사항 검증에 매우 효과적인 방법이었으며, 요구사항 확인검증에 사용된 검증기법과 도구들은 체계적인 설계 확인검증에도 적용될 수 있다.

Abstract

This paper describes Requirements Verification and Validation(V&V) of operating system(OS) software to be developed for Programmable Logic Controller(PLC)-based digital Plant Protection System(PPS) prototype in Korea Nuclear Instrumentation and Control System (KNICS) project. The OS is being developed as newly developing software, lifecycle V&V is applied, and software V&V criteria and requirements in the Software Review Plan (SRP)/BTP-14, the IEEE Std. 7-4.3.2, the IEEE Std. 1012, and the IEEE Std. 1028 are applied systematically and strictly at each lifecycle phase. Checklist-based Fagan Inspection has mainly been applied for requirements V&V while model checking is applied for formal verification and HAZOP is applied for identification of safety requirements. Checklist-based V&V procedure was very effective for systematic requirements V&V of OS software, and the applied V&V techniques and their tools in requirements V&V can also be applied for systematic design V&V of OS software.

1. 서론

원전계측제어시스템(Nuclear Instrumentation and Control System: KNICS)의 발전소보호

계통(Plant Protection System: PPS) 시제품은 원자로보호계통(Reactor Protection System: RPS)과 공학적 안전설비-기기제어계통(Engineered Safety Features-Component Control System: ESF-CCS)으로 구성되며, 안전등급 제어기기(Programmable Logic Controller: PLC)를 개발하여 적용한다[1]. 안전등급 제어기기는 운영체제 소프트웨어와 응용 소프트웨어를 내장하여(embedded) RPS 의 Bistable Processor(BP)와 Coincidence Processor(CP), 그리고 ESF-CCS 의 Group Processor(GP) 등 발전소보호계통의 critical component 로써 설계에 적용된다.

운영체제 소프트웨어는 신-개발 소프트웨어(Newly Developing Software: NDS)에 해당하고, 응용 소프트웨어의 실행을 제어하기 때문에 안전 소프트웨어(safety software)로 분류되어 왔다[2]. 안전 소프트웨어는 규제기관이 채택하는 Software Review Plan(SRP)/BTP-14[3]와 IEEE Std. 7-4.3.2[4]의 소프트웨어 확인검증 기준, 그리고 IEEE Std. 1012[5] 및 IEEE Std. 1028[6]이 요구하는 소프트웨어 확인검증 요건을 만족해야 한다. 운영체제 소프트웨어의 생명주기 확인검증(Lifecycle V&V)은 요구사항 V&V, 설계 V&V, 구현 V&V, 그리고 시험 V&V 로 구성되며, 이들 각 단계마다 체계적인 확인검증 절차에 따라 엄격하고 철저하게 수행한다. 그리고, 검토 및 인스펙션, 시뮬레이션, 수학적 증명을 통한 정형검증(formal verification), 정적 및 동적시험(static and dynamic software testing), 그리고 안전성 분석을 위한 HAZOP(HAZards and OPerability) Methodology 와 FTA(Fault-Tree Analysis) 등 다양한 검증 기법들과 자동화된 도구가 활용된다.

현재, 운영체제 소프트웨어의 요구사항 확인검증이 수행되었으며, 체크리스트-기반 인스팩션을 요구사항 확인검증에 체계적으로 적용하였다. 운영체제 소프트웨어의 요구사항 명세에 대한 확인검증 경험을 통해, 운영체제 소프트웨어에 적용되는 생명주기 확인검증은 KNICS PPS 를 위한 소프트웨어 안전성과 품질 확보에 충분할 것으로 예측된다.

2. 운영체제 소프트웨어

KNICS PPS 시제품의 운영체제 소프트웨어는 안전등급제어기기(PLC)의 ROM 에 구현되고, 계통기능을 담당하는 응용 소프트웨어의 실행을 제어한다. 그림 1 은 운영체제 소프트웨어를 탑재하여 PPS 프로세서(즉, BP, CP, GC 등)로 사용되는 안전등급제어기기 시제품의 배치를 나타낸다.

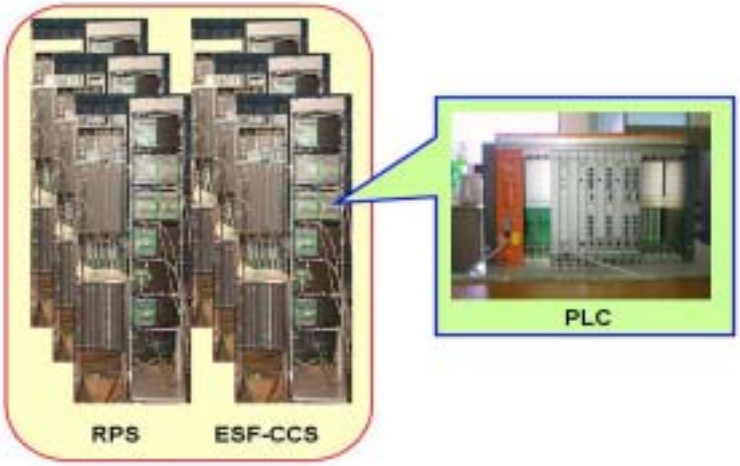

그림 1. KNICS PPS 를 위한 안전등급제어기기의 배치 [1]

2.1 계층적 소프트웨어 구조

운영체제 소프트웨어는 PLC 하드웨어와 응용 소프트웨어 계층 사이에 위치하며, 실시간 운영체제 커널과 시스템 소프트웨어(KNICS 에서는 에이젼트 태스크로 불림)를 포함한다. 이러한 운영체제 소프트웨어는 시스템의 시작 및 초기화, 응용 태스크의 제어, 자가진단, 디스플레이, 그리고 소프트웨어 개발 시스템(KNICS 에서는 공정 엔지니어링 스테이션이라 함)과의 통신-관련 기능을 수행한다. 그림 2 는 PPS 의 계층적 소프트웨어 구조를 나타낸다.

그림 2. KNICS PPS 의 계층적 소프트웨어 구조

2.2 운영체제 소프트웨어의 개발

운영체제 소프트웨어의 요구사항 및 설계명세는 자연어와 정형명세 언어를 사용하여 작성되고, 정형명세는 Statechart 로 명세된다. Statechart 명세는 Statemate MAGNUM(STM) 도구의 Activity-charts 를 이용하여 기능과 입출력을 명세하고, Statecharts 를 이용하여 운영체제 소프트웨어의 행위, 상태, 그리고 상태전이 조건을 명세한다. 그림 3 은 STM 을 이용한 정형명세의 예를 나타낸다.

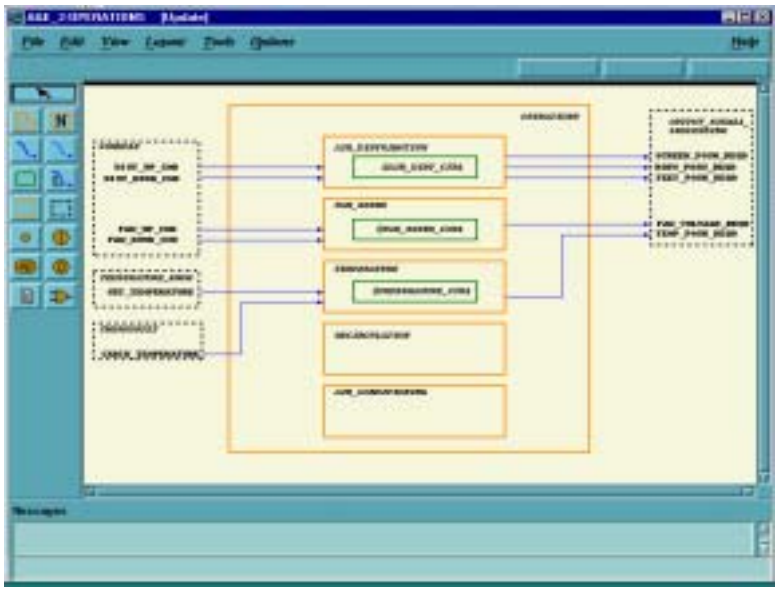

그림 3. STM 을 이용한 Statechart-기반 정형명세 예

운영체제 소프트웨어는 ANSI C 언어 및 TI Assembly 로 구현되고, 정적 및 동적시험을 거쳐 응용 소프트웨어와 통합되어 최종 실행코드가 안전등급제어기기의 ROM 에 내장된다.

2.3 운영체제 소프트웨어의 생명주기 확인검증

운영체제 소프트웨어의 생명주기 확인검증을 위해 그림 4 와 같은 체계적인 소프트웨어 확인검증 전략이 수립되었으며, 규제기준과 산업기술표준이 규정하는 안전 소프트웨어의 확인검증 기준과 요건들을 체크리스트(checklist) 형태로 구성하여 체계적인 검증절차를 마련하였다.

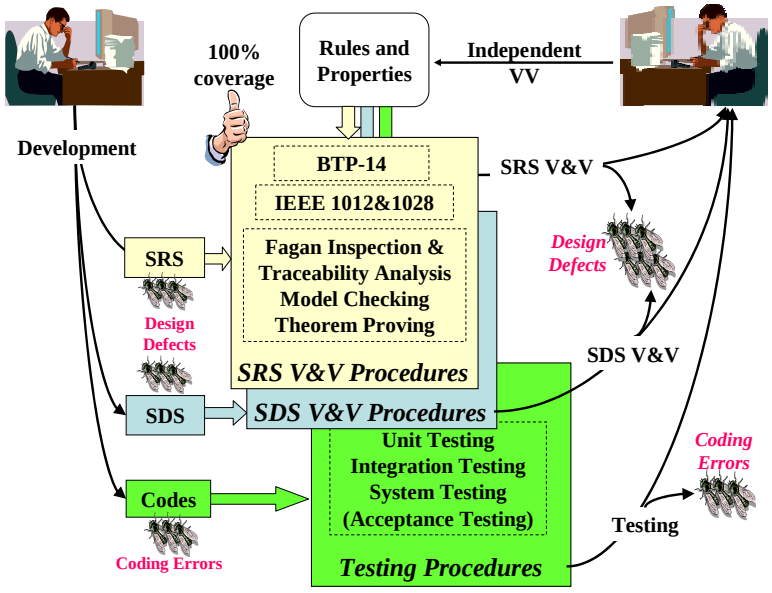

그림 4. 운영체제 소프트웨어의 생명주기 확인검증 절차 [2, 9]

운영체제 소프트웨어의 확인검증은 IEEE Std. 1012[5]의 생명주기별 확인검증 직무(tasks)를 근간으로 요구사항 확인검증, 설계 확인검증, 구현 확인검증, 통합 확인검증, 그리고 계통 확인검증을 위한 확인검증 절차서를 작성하였다.

요구사항 확인검증 단계에서는 추적성 분석, 요구사항 평가, 안전 요구사항 확인, 형상관리 평가, 그리고 시스템 시험 계획 작성을 들 수 있다. 이러한 확인검증 직무의 효율적인 수행을 위해 다양한 소프트웨어 확인검증 기법들과 도구들이 사용된다.

2.4 확인검증절차서 (SVVP)

운영체제 소프트웨어의 확인검증을 위해 체크리스트를 근간으로 생명주기 확인검증 직무(V&V tasks)에 대한 확인검증절차서(Software V&V Procedure: SVVP)가 체계적으로 작성되었다. 그림 5 는 운영체제 소프트웨어의 요구사항 확인검증을 위한 확인검증절차서의 목차(TOC)를 나타낸다.

그림 5. 운영체제 소프트웨어의 요구사항명세 확인검증절차서 목차 [7]

3. 요구사항 확인검증

요구사항단계 확인검증은 요구사항 추적성분석, 요구사항 검토, 요구사항 상세검증(정확성, 완전성, 일관성), 안전 요구사항 확인, 형상관리 평가, 그리고 시스템 시험계획 작성을 주요 확인검증 직무(V&V tasks)로 들 수 있다. 이러한 확인검증 직무의 효율적인 수행을 위해 다양한 소프트웨어 확인검증 기법들과 도구들이 그림 6 과 같이 적용된다.

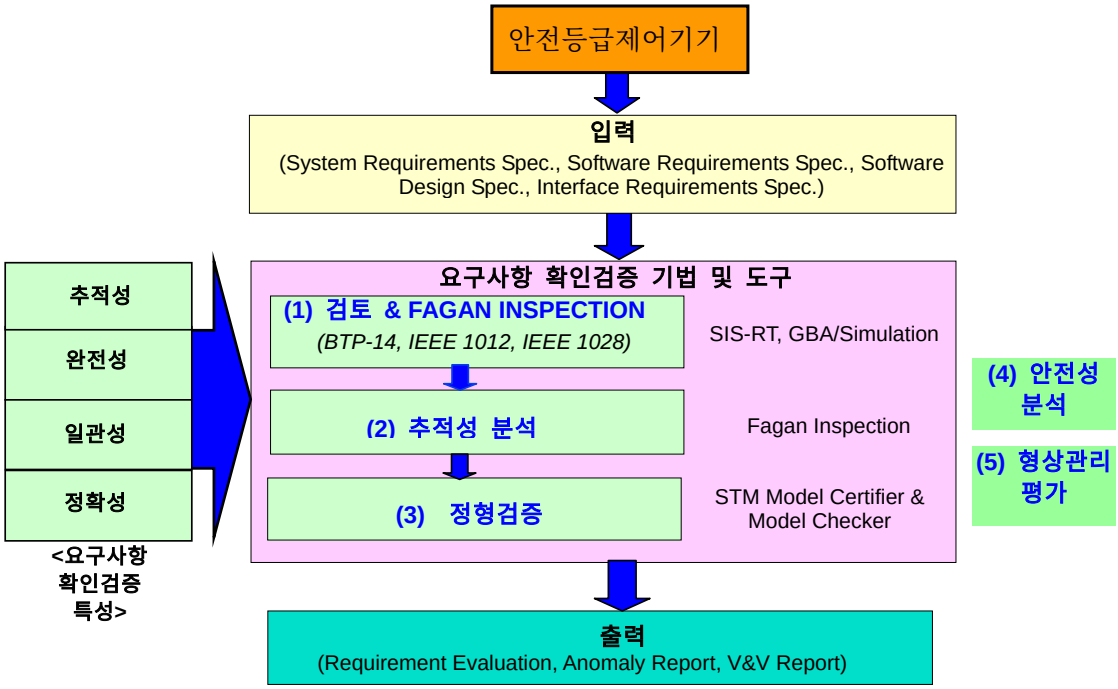

그림 6. 운영체제 소프트웨어의 요구사항명세 확인검증

3.1 요구사항 추적성 분석

운영체제 소프트웨어의 요구사항을 계통 요구사항에 대하여 추적하고, 계통

요구사항을 요구사항에 대하여 추적한다(현재, 계통 요구사항은 RPS 및 ESF-CCS 의 기능요건에 준하여 수행되었으며, 추후 PLC 계통의 기능요건에 대하여 적용되어야 한다.). 추적성 분석의 특성(properties)은 정확성, 일관성, 그리고 완전성이다. 이러한 추적성 분석의 특성에 대하여 체크리스트가 절차로 이용된다. 추적성 분석에는 자연어로 작성된 요구사항에 대하여 KAIST 에서 개발된 추적성 분석 도구(SIS-RT)[8]가 그림 7 과 같이 활용되었다.

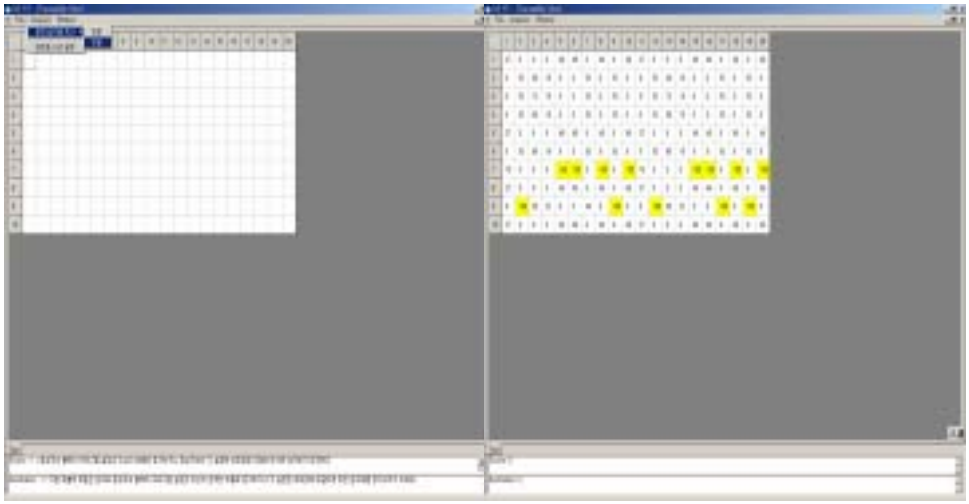

그림 7. SIS-RT 를 이용한 추적성 분석 화면 예

3.2 요구사항 검토 및 검증

운영체제 소프트웨어의 요구사항은 BTP-14[3]의 기능특성 및 공정특성에 대하여 검토된다. 그림 8 은 이러한 검토과정을 나타낸다.

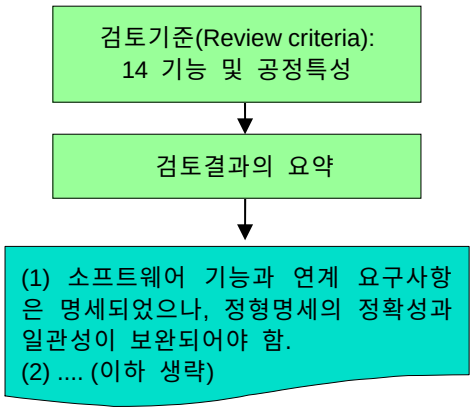

그림 8. BTP-14 의 기능특성 및 공정특성에 대한 검토

요구사항 검증기준은 정확성, 일관성, 그리고 완전성이다. 요구사항 검증의 입력은 SRS 와 계통요건이며, 출력은 요구사항 이상상태 보고서이다. Fagan Inspection 은 소프트웨어 품질을 검증하기 위한 소프트웨어 검토(review) 방법 중의 하나로써 1970 년대 중반 IBM 의 Fagan 에 의해 제안되었으며[9], 운영체제 소프트웨어의 요구사항 검증에 체크리스트-기반 Fagan Inspection 기법이 사용된다. 표 1 은 체크리스트-기반 Fagan Inspection 에 의한 운영체제 소프트웨어의 요구사항

검증의 일부를 나타낸다.

검증특성	검증기준	체크리스트		검증결과
완전성	안전등급제어 기기의 SRS 검증절차서(KNICS-SED-SVP121 (Rev. 00))	Agent 의 기능 요구사항이 완전하게 기술하였는가?	각 기능 요구사항의 시작을 기술하고 있는가?	7.0 장에서 시스템 에이젼트의 인터페이스와 일반기능이 기술되었고, 9.0 장에서 시스템 에이젼트 각각에 대한 요구사항의 시작을 명세하였슴.
			각 기능 요구사항이 그 기능을 수행하기 위해 요구되는 작업 순서, 행위 및 사건들을 명시하고 있는가?	기능 요구사항은 시스템 에이젼트의 작업(task), 행위, 그리고 사건에 대하여 명세되었으나(정형명세 포함), 각 요건에 그 기능과 상태이름을 명시하여 완전한 명세로 보완될 필요가 있슴.
			운전 중에 사용자에 의해 수정될 수 있는 소프트웨어의 변수가 있는가?	운전 중에 사용자가 수정하는 변수는 없슴.
	….	….	….	….
일관성	….	….	….	….
정확성	….	….	….	….

표 1. 체크리스트-기반 요구사항 Fagan Inspection

3.3 요구사항 정형검증

운영체제 소프트웨어의 정형검증은 Statechart-기반 정형명세에 대하여 수행되고, 정형검증의 검증특성(properties)은 완전성, 정확성, 그리고 일관성이다. 이러한 정형검증에는 STM 의 모델체커(Model Checker)와 시뮬레이션(Graphical Back Animation: GBA)이 이용되었다 [그림 9]. STM 을 이용하여 명세된 Statechart 의 완전성과 정확성 검증은 모델체커를 이용함으로써 거의 자동으로 검증된다. 모델체킹 및 GBA 를 통해 운영체제 소프트웨어의 요구사항 명세에 대한 non-determinism 을 포함하는 요구사항 명세의 오류들을 발견하여 요구사항 명세의 완전성, 일관성, 그리고 정확성 향상으로 반영되었다.

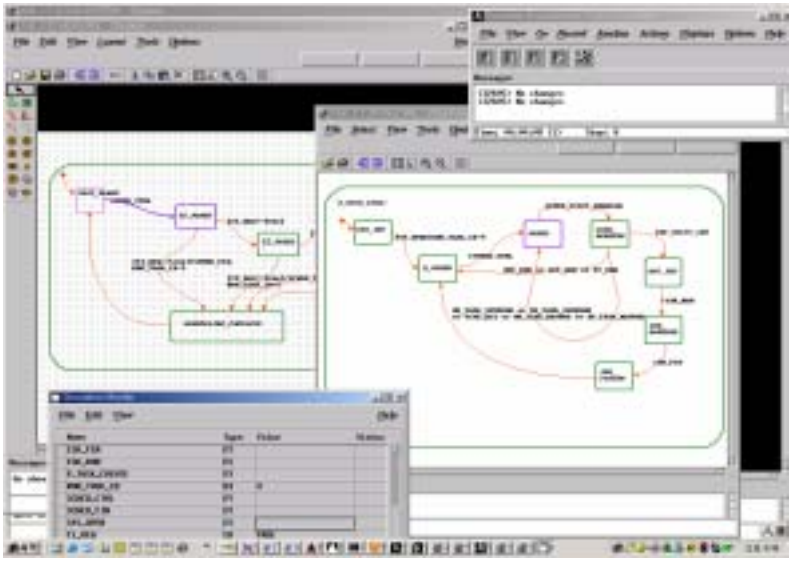

그림 9. STM Simulator 를 이용한 GBA 과정 예

3.4 안전 요구사항 확인 (HAZOP 방법)

운영체제 소프트웨어의 안전성 분석은 원칙적으로 계통 수준에서 수행하는 것이 타당한 것으로 판단된다. 그러나, 운영체제 소프트웨어가 안전등급제어기기 측면에서 매우 중요한 기능을 수행하기 때문에, 계통의 안전에 영향을 미치는 요인을 확인하는 직무는 요구사항 확인검증 단계에서도 체크리스트로 절차를 제공하여 검토한다. 체크리스트는 HAZOP 방법[10]의 Guide phrases 를 참고하여 작성되었으며, 표 2 는 체크리스트를 이용하여 운영체제 소프트웨어의 안전 요구사항을 확인하는 일부를 나타낸다.

품질 특성	Guide Phrases	Deviation Checklist	원인	공통 원인	위험영향(결과)	위험도	설계 권고사항	비고
안전성	Software causes system to move to a hazardous state	소프트웨어가 계통을 위험한 상태로 옮기는 원인으로 작용할 때 발생 가능한 결과는 무엇인가?		(배제 못함)	운영체제 소프트웨어 시스템 자체는 계통을 안전한 상태로 운전할 수 없음.		Critical function 은 하드웨어,다중화, 또는 diverse 기능 설계할 것.	
	….	….			….		….	
	….	….			….		….	
	….	….			….		….	

표 2. HAZOP 에 의한 안전 요구사항 확인

3.5 계통 시험계획 생성

운영체제 소프트웨어의 구현(즉, 코딩)에 대한 시험은 그림 10 와 같이 IEEE Std. 1012[5]의 생명주기 소프트웨어 시험업무의 정의를 적용한다.

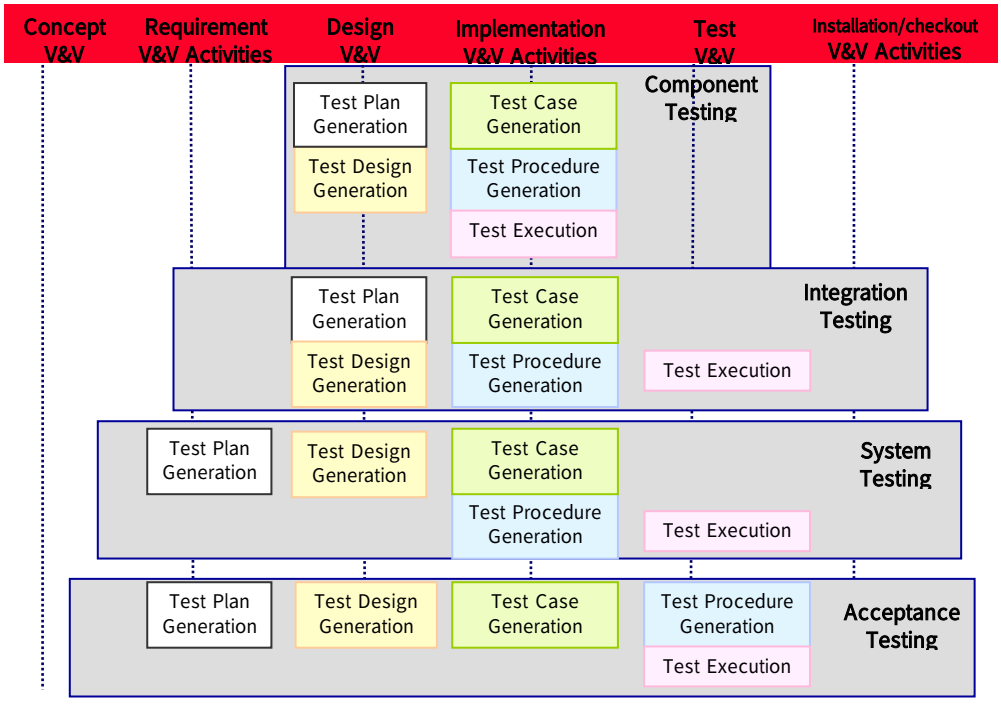

그림 10. 생명주기 시험업무 [5]

그림 10 에서 요구사항 확인검증 활동의 시험업무는 계통시험계획서와 Acceptance 시험계획서 생성이다. KNICS PPS 시제품은 적용 발전소가 정해지지 않은 관계로 계통시험계획서만 생성하였다.

3.6 형상관리 평가

운영체제 소프트웨어의 요구사항 확인검증 활동 중 소프트웨어 형상관리 평가는 설계문서와 데이터를 포함하는 형상관리 항목(CM items)에 대하여 형상관리 절차에 따라 평가하고, 형상관리 공정이 완전하고 적절한가를 확인한다. 형상관리 평가를 돕기 위하여 형상관리 도구(NuSCM)[11]가 개발되어 적용되어 왔다.

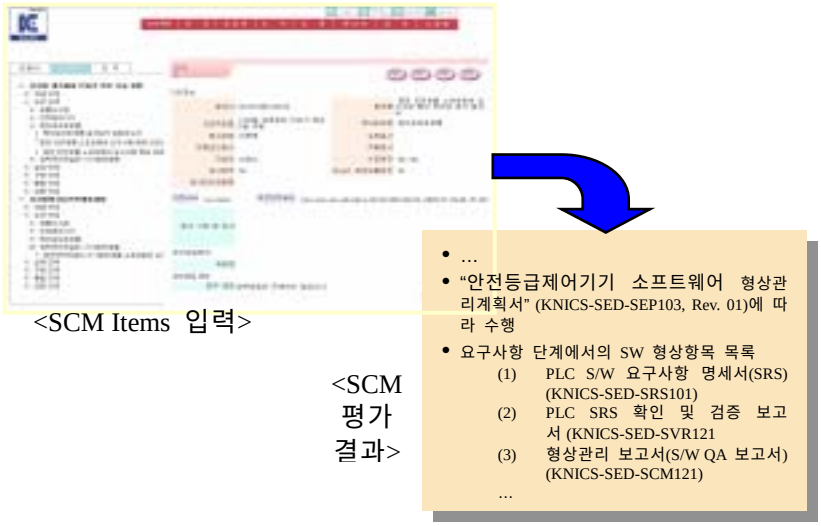

그림 11. NuSCM 을 이용한 형상관리 평가 예

4. 결론

운영체제 소프트웨어가 탑재되는 PLC 를 안전계통 개발에 적용하기 위해서는 운영체제 소프트웨어가 응용 소프트웨어(즉, 계통 기능)의 실행을 제어하기 때문에 운영체제 소프트웨어 그 자체도 응용 소프트웨어 수준의 안전성과 품질이 확보되어야 한다.

KNICS 원자로보호계통과 공학적안전설비-기기제어계통 시제품의 운영체제 소프트웨어에 대한 요구사항 확인검증을 통해, 체크리스트로 구성된 검증절차가 체계적인 요구사항 검증에 매우 효과적인 방법임을 확인할 수 있었으며, 정형검증 기법과 자동화 도구의 사용이 효율적임을 경험하였다.

요구사항 확인검증에 사용된 검증기법과 도구들은 체계적인 설계 확인검증에도 반복 적용할 수 있으며, 체크리스트로 구성되는 확인검증절차는 생명주기 확인검증 업무에 매우 효과적이고 효율적인 것으로 판단된다.

알림

이 연구는 원자력기반기술분야 원전계측제어시스템(KNICS) 개발 사업의 디지털 계측제어 인허가 확보기술 개발 과제에서 수행되었음.

참고문헌

[1] Han, Jae Bok, “Application of PLC for PPS and ESF-CCS,” Korea-Russia Joint Workshop on Nuclear I&C, Seoul, Feb. 19-20, 2004.

[2] 차경호 외, “The KNICS Approach to Verification and Validation of Safety-Critical Software for RPS Prototype,” 2003 춘계학술발표회논문집, 한국원자력학회, 2003 년 5 월 30 일.

[3] USNRC, “BTP-14: Guidance on Software Reviews for Digital Computer-based I&C Systems,” July 1997.

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

[5] IEEE Std. 1012-1989, “IEEE Standard for Software Verification and Validation Plans.”

[6] IEEE Std. 1028-1997, “IEEE Standard for Software Reviews and Audits.”

[7] KAERI, “SVVP for SRS of POSAFE-Q PLC,” KNICS-SED-SVP121 (Rev.00), Jan. 14, 2002. (In Korean)

[8] Seo Ryong Koo, et al., “Development of Software Requirement Analysis Tool for NPP Software Fields Based on Software Inspection and Formal Method,” Proceedings of International Symposium on the Future I&C for NPP, Seoul, Korea, Nov.7-8, 2002, pp.159-164.

[9] 손한성 외, “KNICS 원자로보호계통 소프트웨어 검증을 위한 체크리스트 개발,” 2003 춘계학술발표회 논문집, 한국원자력학회, 2003 년 5 월 30 일.

[10] 이장수 외, “소프트웨어 요구명세 안전성 분석을 위한 HAZOP 방안,” 2003 춘계학술발표회 논문집, 한국원자력학회, 2003 년 5 월 30 일.

[11] Bo-Gyun Byoun, et al., “A Design of Project-based Software Configuration Management System,” Proceedings of International Symposium on the Future I&C for NPP, Seoul, Korea, Nov.7-8, 2002, pp.165-168.