무선통신기반 열차제어시스템의 SIL 4 SW 모듈 검증에 관한 연구

Verification of SIL 4 SW Modules for Communication based Train Control System

김자영*†, 박기수*, 조동래*, 이태규*, 전종화*

Ja-Young Kim*†, Gie-Soo Park*, Dong-Rae Cho*, Tae-Gyu Lee*, Z.H. Quan*

Abstract As with the hardware and software errors, it is possible to generate the loss budget due to the failure of the project and enormous casualties, verification of safety is required. In the field of railway signaling system, reliability and safety of software has become a issue. In this paper, in accordance with international standards “Railway applications-Communications, signaling and processing systems-Software for railway control and protection systems” IEC62279, software modules of wireless communication-based train control system, is a study on the verification and validation method was applied in order to satisfy the Safety Integrity Level 4.

Keywords **:** , CBTC, SIL, Train Control System, SW Verification, Software Quality Assurance

초 록 하드웨어뿐만 아니라 소프트웨어의 오류는 엄청난 인명피해와 프로젝트 실패로 인한 예산 손실을 발생시킬 수 있어 안전성검증이 요구된다. 철도신호시스템 분야에서도 소프트웨어의 신뢰성과 안전성이 이슈로 떠오르고 있는 가운데 휴먼 에러와 같은 불확실성이 존재하는 소프트웨어를 열차제어시스템과 같은 Safety-Critical 시스템에 적용하기 위해 철저한 안전성 검증이 요구되고 있다. 본 논문은 무선통신기반 열차제어시스템의 소프트웨어 모듈을 IEC 62279 ‘철도용 전기 설비의 통신 및 신호 처리 시스템과 제어 및 보호 시스템에 관한 소프트웨어’ 국제 규격에 부합하며 가장 높은 안전성 수준인 Safety Integrity Level(SIL) 4를 만족하기 위해 적용한 검증 방안에 관한 연구이다.

주요어 : CBTC, SIL, 열차제어시스템, SW검증 , 소프트웨어 품질보증

1. 서 론

본 연구에서 소개하는 무선통신 기반 열차제어 차상 시스템은 지상과 열차간 무선을 이용하여 실시간으로 정확한 위치, 속도, 운행 방향과 제동거리 등의 정보를 교환하며 열차를 제어하는 시스템으로 무인운전을 목표로 하는 시스템이므로 Safety-Critical 시스템이다. 이러한 시스템의 안전성과 신뢰성을 보증하는 인증을 확보하기 위해 철도용 소프트웨어 국제 규격인 IEC 62279를 가이드라인으로 삼는다. 본 연구는 안전성 레벨 중 가장 높은 SIL 4 수준을 만족하도록 소프트웨어 개발 수명 주기 중 소프트웨어 모듈 디자인, 코드, 소프트웨어 모듈 테스팅 단계에서 적용된 검증 및 확인 활동에 대해 기술한다.[1]

† 교신저자: POSCO ICT 기술 연구소 철도교통기술팀(kjy121@poscoict.com)
* POSCO ICT 기술연구소 철도교통기술팀

2. 본 론

2.1 무선 통신기반 열차제어 차상 시스템 소프트웨어

2.1.1 무선 통신기반 열차제어 차상 시스템 소프트웨어 개요

무선통신기반 열차제어 차상 시스템은 자동열차방호(ATP) 기능과 자동열차운행(ATO) 기능으로 구성된다. 자동열차방호 기능은 열차의 과속 또는 해당 열차에 부여된 이동권한범위를 초과하여 운행하는 것을 방지함으로써 충돌 또는 기타 사고를 유발할 수 있는 위험을 예방하며 자동열차운행 기능은 자동열차방호 기능의 감시하에 무인운전을 가능하게 한다. ATP는 SIL 4, ATO는 SIL 0 수준으로 구성되므로 더 높은 수준의 안전 무결성 수준을 가진 ATP기능에 초점을 맞추어 기술한다. ATP의 소프트웨어 컴포넌트는 Fig. 1과 같으며 ATP 주요기능을 수행하는 ATP_Application, 외부 장치와 인터페이스하는 Network Application, RTOS(Real Time Operating System), BSP(Board Support Package), Software Interface, Firmware 로 구성된다. SW Component는 디자인 설계 단계를 통해 최종 하위 단위인 모듈 단위로 상세 디자인된다. 모든 설계 과정은 소프트웨어 요구 사양서를 기반으로 관련 없는 모듈이 추가되거나 요구사항을 만족하지 못하는 경우가 없도록 추적성을 확보한다.

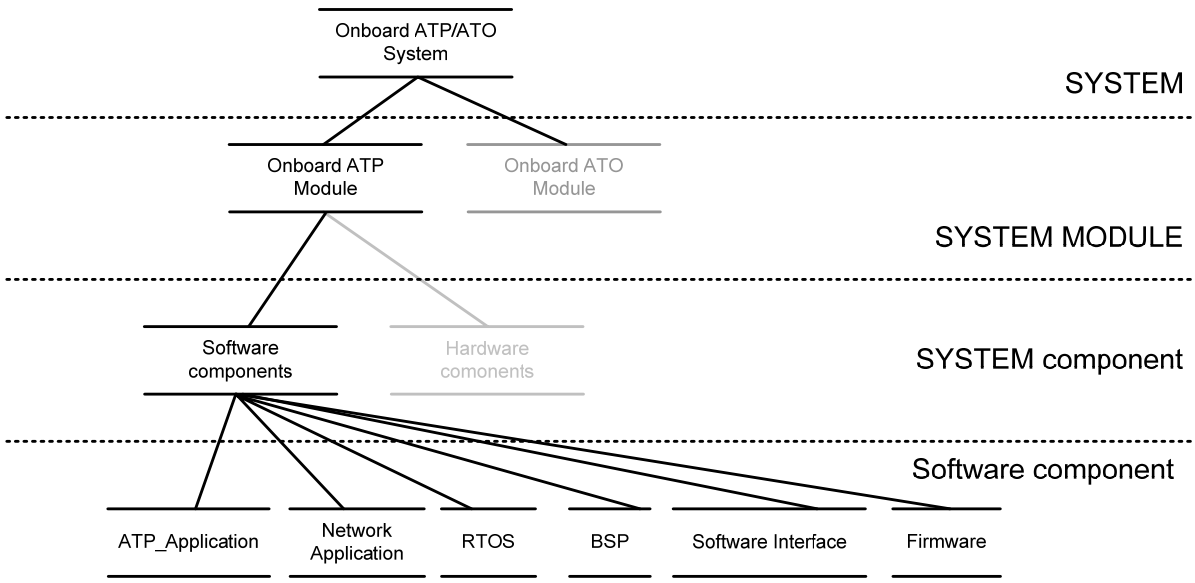


Fig. 1 ATP Software Component

2.2 소프트웨어 확인 및 검증 활동

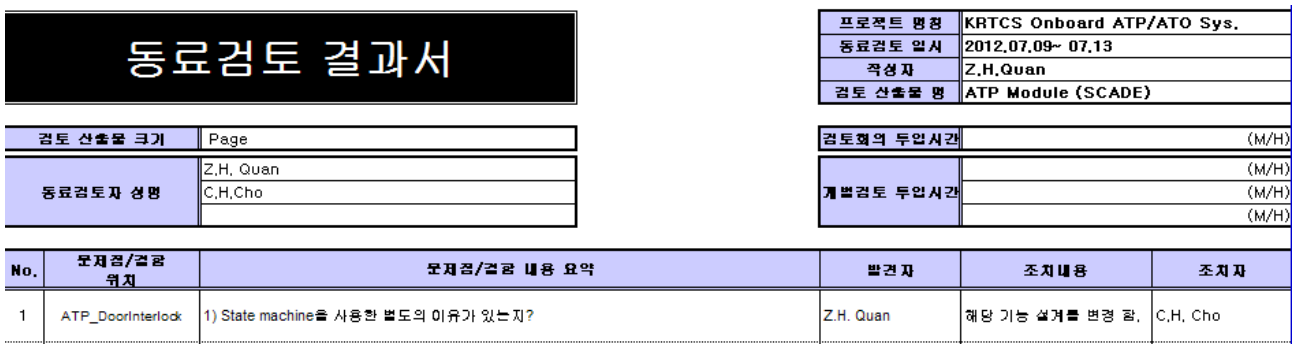
2.2.1 소프트웨어 모듈 디자인 단계

소프트웨어 모듈은 소프트웨어 요구사항 명세서를 만족하도록 소프트웨어 모듈 디자인 명세서로 기술된다. 소프트웨어 모듈 디자인을 위해 선택한 SIL 4 수준의 모델링 기법은 Table 1과 같다.

Table 1 Modeling

TECHINIQUE/MEASURE	SWSIL 0	SWSIL 1	SWSIL 2	SWSIL 3	SWSIL 4
Finite state machine	-	HR	HR	HR	HR
Formal methods	-	R	R	HR	HR
Structure diagrams	-	R	R	HR	HR

소프트웨어 모듈 디자인 명세서의 검증으로 Peer Review를 수행한다. Peer Review를 통해 소프트웨어 요구사항의 추적성과 소프트웨어 모듈 디자인의 적합성 및 일관성, 무결성을 확인하며 Fig. 2와 같은 산출물을 생성한다.



동료검토 결과서

프로젝트 명칭	KRTCS Onboard ATP/ATO Sys.
동료검토 일시	2012.07.09~ 07.13
작성자	Z.H.Quan
검토 산출물 명	ATP Module (SCADE)

검토 산출물 크기	Page	검토회의 투입시간	(M/H)
동료검토자 성명	Z.H. Quan	개별검토 투입시간	(M/H)
	C.H.Cho		(M/H)
			(M/H)

No.	문제점/결함 위치	문제점/결함 내용 요약	발견자	조치내용	조치자
1	ATP_DoorInterlock	1) State machine을 사용한 법도의 이유가 있는지?	Z.H. Quan	해당 기능 설계를 변경 함.	C.H. Cho

Fig. 2 SW Module Peer Review

2.2.2 소프트웨어 코드 단계

소프트웨어 코드 단계의 검증은 소프트웨어 안전 무결성 레벨(SIL)에 따라 소스 코드가 소프트웨어 모듈 설계 사양서와 소프트웨어 품질 보증 계획서의 만족 여부와 지정한 코딩 표준이 정확하게 적용되었는지 확인하기 위한 점검을 포함한다. 적합한 도구, 설계방법, 사용 언어 그리고 컴파일러의 선택이 SIL 4 요구사항을 만족해야 한다. 사용된 주요 언어는 C언어이며, 코딩 룰(Coding Standard)는 MISRA-C를 사용하였다. 해당 룰은 높은 신뢰성을 요구하는 자동차, 항공, 통신, 철도 등의 분야에서 많이 사용되는 코딩 룰이다. FPGA(Field Programmable Gate Array)를 위한 VHDL(VHSIC Hardware Description Language) 코딩에 있어서는 DO-254 user group 회원이며 항공 및 DO-254 전문 EDA 회사인 Mentor Graphics에서 제공하는 DO-254 Rule-set을 코딩 규칙을 적용한다. Static Analysis를 위해 선택한 SIL 4 Technique은 Table 2와 같다. 코드 단계를 검증하는 Evidence가 인증기관에 제공되어야 하며 코딩 룰의 Evidence는 Fig.3과 같다. 코드 단계의 검증 최종 산출물은 소프트웨어 코드 확인 보고서이다.

Table 2 Static Analysis

TECHINIQUE/MEASURE	SWSIL 0	SWSIL 1	SWSIL 2	SWSIL 3	SWSIL 4
Boundary value analysis	-	R	R	HR	HR
Control flow analysis	-	HR	HR	HR	HR
Data flow analysis		HR	HR	HR	HR
Walkthrough/design reviews	HR	HR	HR	HR	HR

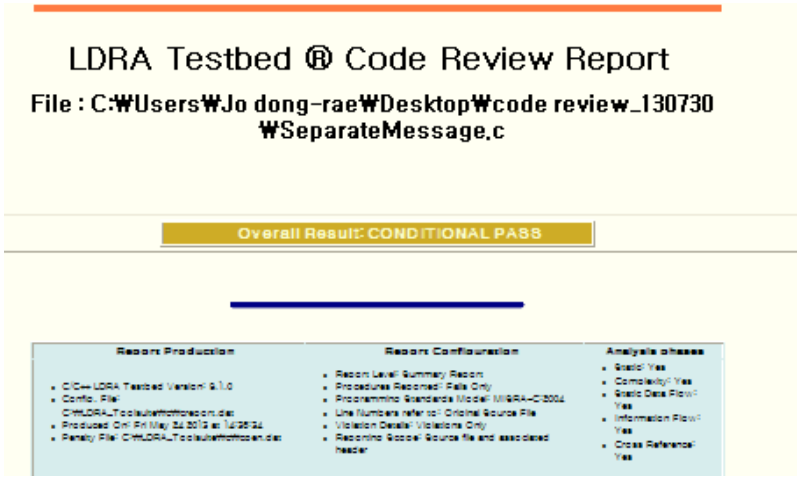


Fig. 3 Coding Rule Evidence

2.2.3 소프트웨어 모듈 시험 단계

소프트웨어 모듈 시험은 하나의 소프트웨어 단위가 정상적으로 기능을 수행하는지 여부를 테스트하는 것으로 소프트웨어 모듈의 다른 부분과의 연계성을 고려하지 않고 격리하여 테스트한다. 소프트웨어 모듈 시험은 가능한 많은 단위 내 결함을 제거하는 것에 포커스를 맞추며 SIL 4 수준에 맞는 소프트웨어 모듈의 시험을 위해 선택한 Technique은 Table 3, Table 4와 같다.

Table 3 Software Verification and Testing

TECHINIQUE/MEASURE	SWSIL 0	SWSIL 1	SWSIL 2	SWSIL 3	SWSIL 4
Static Analysis	-	HR	HR	HR	HR
Dynamic Analysis and Testing	-	HR	HR	HR	HR

Table 4 Dynamic Analysis and Testing

TECHINIQUE/MEASURE	SWSIL 0	SWSIL 1	SWSIL 2	SWSIL 3	SWSIL 4
Test case execution from boundary value analysis	-	HR	HR	HR	HR
Equivalence classes and input partition testing	-	R	R	HR	HR
Structure based testing	-	R	R	HR	HR

소프트웨어 요구사항을 모두 커버리지 하도록 테스트 케이스를 추적 관리하며 소프트웨어 모듈의 기능을 충분히 시험하도록 비정상 상황(Abnormal)을 고려하여 테스트 케이스를 구성한다. Structure based Testing은 구조적 결함이 없음과 시험의 충분도를 위한 것으로 결정 커버리지(Decision Coverage)를 선택하여 100% 커버리지 수행한다. 결정 커버리지 100% 수행을 통해 소프트웨어 구문이 최소한 한 번 실행되고 각 결정문이 모든 가능한 결과값을 최소한 한번 가지는 것을 확인한다. Fig. 4는 소프트웨어 모듈의 결정 커버리지 100%로 Complete(C) 상태임을 입증하는 Evidence이다.

```
3. Coverage Information

3.1. Synthesis
Columns details:
 - Tested: coverage cases covered by tests
 - Justified: coverage cases covered by justification
 - Total: covered coverage cases / number of coverage cases for the whole of instantiated operators or per operator
 - Status: 'NC' when the coverage is Not Complete; 'C' when the coverage is Complete
Status is 'C' when all the operators have been activated and when there is no not instantiated operators neither excluded operators.

+-----------------------------------------------------------------------------------------------------------+--------+-----------+--------+--------+
| Overall coverage                                                                                          | Tested | Justified | Total  | Status |
+-----------------------------------------------------------------------------------------------------------+--------+-----------+--------+--------+
| Instantiated operators                                                                                    | 6      | 0         | 6/6    | C      |
+-----------------------------------------------------------------------------------------------------------+--------+-----------+--------+--------+
| 150 not instantiated operators                                                                            | N/A    | N/A       | N/A    | NC     |
+-----------------------------------------------------------------------------------------------------------+--------+-----------+--------+--------+
| 0 excluded operators                                                                                      | N/A    | N/A       | N/A    | C      |
+-----------------------------------------------------------------------------------------------------------+--------+-----------+--------+--------+
```

Fig. 4 Structure based testing Evidence

3. 결 론

Safety-Critical 시스템인 무선통신기반 열차제어 차상 시스템의 소프트웨어 안전성을 SIL 4 수준으로 인증 받기 위해 IEC 62279를 가이드라인으로 하였다. 본 논문은 소프트웨어 개발 수명 주기 중 모듈 설계, 코드, 모듈 시험 단계의 확인 및 검증에 초점을 맞추어 구체적 내용을 제시하였다. 인증을 획득하기 위해서는 엄격하고 형식화된 프로세스와 Technique, 검증된 툴을 사용하여 명확한 Evidence를 제공해야 하므로 시스템에 적합한 Technique과 측정 방법, 도구를 선별하는 것이 중요하다. 소프트웨어의 개발은 실제적으로 계속적인 변경이 발생하므로 형상관리가 엄격히 이루어져야 하며, 확인 및 검증을 수행하는 Verifier는 별도의 조직으로 독립 구성하여 소프트웨어 검증 과정에서 개발조직으로부터 영향을 받지 않도록 해야 한다.

참고문헌

[1] IEC 62278 (2002) Railway Applications – The Specification and Demonstration of Reliability, Availability, Maintainability and Safety(RAMS)

[2] IEC 62279 (2002) Railway Applications – Software for Railway Control and Protection Systems

[3] EN 50128 Railway applications - Communications, signaling and processing systems – Software for railway control and protection systems

[4] Z.H. Quan (2011) Development of CBTC Car-borne Software with Model-Based Design and Its Applications, *The Korean Society for Railway*, pp. 910-917

[5] Kyudon. Shim (2010) Software Quality Assurance Activities of Automatic Train Control System to meet Requirements of the IEC 62279 Standard, *The Korean Society for Railway*, pp. 412-418