EN50128 SIL4 소프트웨어 프로세스 ISA 인증 사례 연구

Study on ISA's assessment to software process for EN50128 SIL4

조치환† 강찬용* 황진호*
Chi-Hwan Cho Chan-Yong Kang Jin-Ho Hwang

ABSTRACT

It is inevitable to control the systematic failure to obtain the software safety integrity of embedded software installed in rolling stock. Because it is not possible to assess systematic failure integrity by quantitative methods, SILs are used to group documentation, methods, tools and techniques throughout software development lifecycle which, when used effectively, are considered to provide an appropriate level of confidence in the realization of a system to a stated integrity level. Normally, safety approval process is through generic product, generic application and specification application for . For safety approval on generic application of software based system, it is required to apply the certified software processes from the planning stage for the assigned SIL. As such, we will develop project specific application with high safety integrity within time limit of contractual delivery schedule through software assessment to the modified area with the re-use of certified software module and documentation. At this point, Hyundai Rotem has developed software processes applicable to support SIL 4 based on EN50128 which was assessed and certified by TUV SUD. This paper introduces the Hyundai Rotem's detailed approach and prospective action to achieve software safety integrity level.

1. 서론

철도차량에 탑재되는 임베디드 소프트웨어중 안전기능이 할당된 소프트웨어의 안전성 확보를 위해 해당 SIL에 해당되는 systematic failure가 제어되어야 한다. 소프트웨어는 Systematic 고장에 대해 정량적 안전 목표값을 할당할 수 없기 때문에 정성적인 SIL 등급을 부여하여 그에 상응하는 소프트웨어 생명 주기 단계별 문서화, 방법론, 툴, 각종 기법등을 효과적으로 적용하여 소프트웨어의 안전 무결성을 확보해야 한다. SIL 인증을 받고자 하는 제품은 Generic Product, Generic 및 Specific Application 등 그 목적에 따라 인증 과정을 거치게 되는데, Generic Application의 특정기능이 소프트웨어에 의해 구현되는 경우 EN50128규격에 따라 특정 SIL등급에 해당하는 소프트웨어 프로세스(개발 방법론)가 계획단계부터 적용된다면 향후 프로젝트별로 Specific Application을 구현할 때 소프트웨어 모듈 및 산출물을 재사용하여 변경된 기능 위주로만 평가를 받아 시행착오 없이 계약 납기내 안전 무결성이 확보된 소프트웨어를 개발/납품할 수 있는 가능성이 높아진다.

R사는 EN50128 기반 SIL4를 지원하는 소프트웨어 프로세스를 연구과제로 구축하여 TUV SUD로부터 인증을 획득하였고, 인증된 프로세스 표준을 웹기반 SIL 관리시스템에 탑재하였으며, SIL개념을 도입하여 소프트웨어 개발시 이번에 인증받은 프로세스를 적용 할 계획이다.

† 비회원, 현대로템(주) 시스템기술팀, 선임연구원
E-mail : onetop@hyundai-rotem.co.kr
TEL : (031)596-9331 FAX : (031)596-9759
* 비회원, 현대로템(주), 수석연구원
* 비회원, 비즈피어(주), 책임컨설턴트

이에 본 논문은 EN50128 기반 SIL4 기반을 지원할 수 있는 소프트웨어 프로세스 인증 추진배경, 프로세스 체계 및 구성, 인증절차, 웹기반 SIL 지원시스템 및 향후 실제 제품에 대한 적용 계획 등에 대해 간략히 소개하고자 한다.

2. 본론

2.1 EN50128 및 EN50126, EN50129

EN50128은 EN50126, EN50129와 함께, 산업 일반에 적용가능한 IEC61508의 Sector Specific Version(Railway)의 하나로서, 할당된 SIL target에 맞추어 Railway 관련 응용시스템을 개발할 때에 준수해야 하는 SW Engineering 요건이다.

EN50126에서는 Railway 관련 응용시스템의 개발 시 준수해야 하는 System Engineering 관점에서 RAMS 요건에 대하여 비교적 추상화가 높은 수준에서 기술하고 있다. EN50128은 System Engineering 요건의 일부로서의 SW Engineering 요건을 기술하고 있다. EN50126이 System Engineering의 관점에서 EN50128 및 EN50129를 커버하는 형태이므로 EN50128은 EN50126과 분리된 것으로 볼 수 없다. 그러므로 EN50128에서는 SW 라이프사이클 과정에서 EN50126 기반 System Engineering 과정과의 긴밀한 연계가 지속적으로 강조되고 있다. EN50129의 경우 Railway 응용시스템의 신뢰성과 안전성을 확보하기 위한 기술의 소개와 함께 Safety Case에 대한 가이드를 포함하고 있다.

EN50126의 경우 품질표준으로서 ISO9001의 준수를 요구하고 있으며, EN50128의 경우 ISO9000-3의 준수를 요구하고 있다. 하지만 ISO 9000-3은 현재 ISO/IEC 90003으로 개정된 상태이므로, 실질적으로 EN50128의 지원을 위해서는 ISO/IEC 90003의 채택이 합당하고, ISA도 그렇게 요구하고 있다.

2.2 EN50128 의 특징

2.2.1 SW 개발 라이프사이클 및 방법론

EN50128은 RAMS 기반 SW 개발과정에서 준수해야 할 활동요건들을 관련 산출물 요건들과 함께 기술하고 있다. 하지만 EN50128의 구성의 순차성이 특정 라이프사이클을 엄격히 따르는 것은 아니다. 그러므로 원론적으로는 EN50128에 기술된 요건들을 준수한다는 전제 하에서는 어떠한 라이프사이클의 구성도 허용된다.

하지만 EN50128에서는 Formal Method 나 구조적 Method 와 함께 Structured Methodology의 적용을 강력하게 권장하고 있다. 이는 전통적으로 높은 수준의 신뢰성이 요구되는 시스템들이 대부분 Structured Methodology의 적용을 통해서 개발되어 온 역사적 사실과 무관하지 않을 것이다. 또한 SW의 신뢰성과 안전성이 중요한 상황이라면 잘 알려지고 쉽게 따를 수 있는 라이프사이클의 적용이 요구되는 경향이 있다. SIL 의 수준이 높아지면 높아질수록 다루어야 하는 문제의 개수가 많아지고 문제의 수준도 따라서 높아지기 때문이다. 이는 EN50128을 따라야 하는 경우 채택 가능한 라이프사이클 중의 하나가 폭포수 모델일 수 있다는 점을 시사한다. EN50128의 경우, 예로서 제시하는 SW 개발 라이프사이클 및 산출물은 EN50128의 Figure 4에 명시되어 있다.

EN50128에서는 객체지향 기법에 대하여 단지 "Subset of C or C++ with coding standards" 및 "Object Oriented Programming" 정도만 언급되어져 있다. 하지만 객체지향기법 적용 영역의 확산은 SW 업계의 세계적인 흐름 중 하나이다. 더구나 신뢰성과 안전성이 요구되는 분야에 있어서의 객체지향 기법의 기여는 최근 몇 년간 커다란 신장을 보이고 있으며, 객체지향 SW 시스템의 신뢰성, 안전성 관련 지식 또한 급속한 축적 속도를 보이고 있다.

그러므로 객체지향 기법을 사용하는 경우, EN50128의 권장사항을 객체지향기법의 관점에서 해석 및 적용하는 방법론의 수립은 당분간 향후 과제가 될 것이다.

2.2.2 특정 SIL 에 따른 개발 기법의 강제

EN50128에서는 특정 SIL에 따라서 SW 기법을 차별화하여 적용할 것을 규정하고 있다. 그러므로 SIL등급이 높아지면 적용되어야 하는 SW 기법의 개수가 많아지고 범위 또한 넓고 깊어지게 된다. 이는 개발조직에게는 상당한 부담이지만 신뢰성과 안전성 목표를 만족시키기 위한 피할 수 없는 선택이기도 하다.

EN50128에서 특정 SIL 에 따라 규정하는 SW 개발 기법은 그 개수와 난이도에서 일반적인 SW 개발 방법론의 범위를 넘어선다.

EN50128에서는 SW 개발 라이프사이클 단계별로 준수해야 하는 라이프사이클별 산출물, 품질 및 기법관련 항목들을 20개의 TECHNIQUE/MEASURE 테이블로 규정해두고 있다. 이들 테이블은 서로간에 독립적인 경우도 있고 의존적인 경우도 있으며, 테이블의 각 항목마다 특정 SIL 별로 준수여부를 결정하는 Criteria 가 'M(Mandatory)', 'HR(Highly Recommended)', 'R(Recommended)', 'NR(Not Recommended)' 및 '-(No Recommendation)'으로 구분되어 있다. 이들 Criteria 에 대한 설명은 EN50128의 Annex A에 명시되어 있다.

만약에 특정 SIL에서 규정된 기법을 적용하지 않는 경우, EN50128에서는 해당 기법이 적용되지 않아도 되는 이유가 적절히 설명되기를 요구한다. 실제적인 ISA(Independent Safety Assessor) 와의 관계 속에서 이러한 Tailoring 방안은 사실상의 '논증'활동이며, 만만치 않은 작업인 경우가 많다.

2.3 SIL4 기반 SW 프로세스 개발 방향

2.3.1 추진 배경

해외 프로젝트의 경우 SIL 인증 요구 및 관련 EN 규격에 대한 적용을 요구하는 경우가 증대되고 있으며, 이러한 경우 사전에 Generic Application에 대한 인증이 없이는 촉박한 납기내 SIL 요구사항을 만족하는 제품을 납품하기에는 현실적으로 어려운 점이 있다.

한편, 모 K업체의 소프트웨어 품질 감사시 K업체는 안전기능에 대한 Generic Application(플랫폼)에 대한 ISA SIL 인증서가 있었고, 소프트웨어는 인증된 SW 프로세스를 사용하여 프로젝트별 Specific Application에 대한 SIL 인증 요구에 대응하고 있었다.

이런 배경에서, 당사 전장품에 대한 Generic Application 인증의 필요성을 느끼게 되었고, 인증을 위한 선행활동으로 소프트웨어 프로세스를 구축하여 인증을 획득하고, 인증된 프로세스를 Generic Application 적용시점부터 적용하므로써, 체계적인 소프트웨어 개발이 가능하고, 시행착오 또한 줄일 수 있을 것으로 확신하여 본 연구과제를 자체적으로 실시하게 되었다.

2.3.2 SW 프로세스 개발 방향

2.3.2.1 EN50128 및 ISO/IEC 90003 기반 구조

EN50128에 의하면 SW 프로세스는 ISO/IEC 90003을 따라야 한다. ISO/IEC 90003은 ISO 9001 품질표준의 SW 부문에 대한 적용 가이드라인인 ISO 9000-3의 개량된 표준이다.

EN50128과 ISO/IEC90003가 동시에 적용되는 SW 개발 프로세스라는 것은 관련 규격의 해석에 따라서 서로 상이한 몇 가지 형태들이 가능하겠지만 본 연구과제는 R사의 SW 프로젝트 이력과 현행 업무관행을 고려하여 EN50128과 ISO/IEC 90003을 다음과 같이 해석 및 적용하기로 하였다.

개발될 SW 프로세스 체계는 크게 ISO 90003 품질 표준의 틀 속에 포함되며, EN50128 기반 SW 개발 방법론의 경우 라이프사이클, 적용기법 및 단계별 산출물은 Product Realization의 영역에 포함되는 것으로 볼 수 있다. 그러면 EN50128에서 라이프사이클과 단계별 산출물을 제외한 나머지 품질영역은 ISO/IEC 90003 Quality Management, Management Responsibility, Resource Management 및 Measurement, Analysis and Improvement 의 적용을 받아야 하는 영역이 된다.

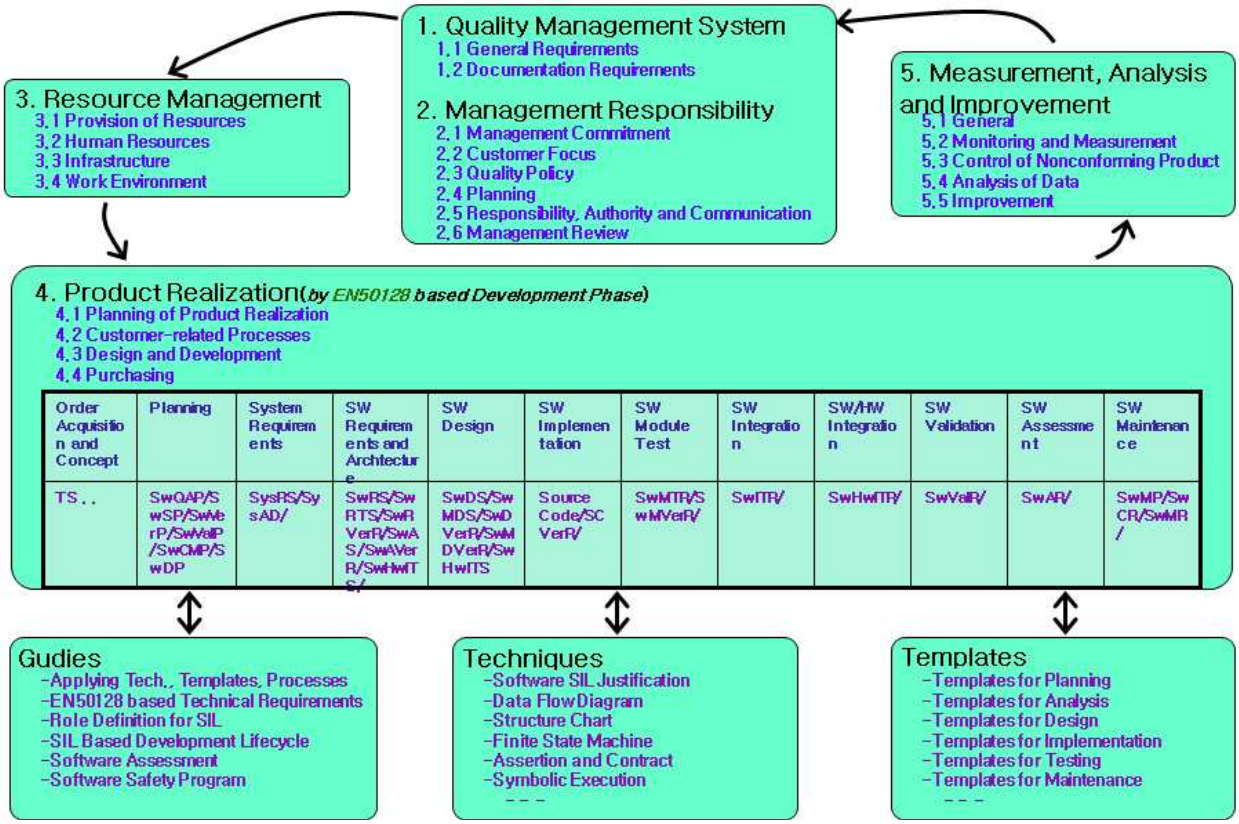

그림1. EN50128 및 ISO/IEC90003 기반 SW 프로세스 체계

2.3.2.2 ISO/IEC90003 요건 충족을 위한 CMMI Level3 프로세스의 보완

R사는 다양한 해외 발주자들로부터의 SW 관련 요구사항에 대한 대응 경험을 보유한 CMMI Level 3 획득 조직이다. 그런데 CMMI Level3와 ISO/IEC90003의 요건 사이에는 공통적인 부문이 존재하고, R사는 해외 프로젝트 업무시 발주자의 엄격한 SW 요구사항에 대하여 지속적으로 대응해야 하는 상황이므로 R사의 SW 프로세스는 몇 가지 부문에서 이미 ISO/IEC 90003의 요건들을 함께 충족시키고 있었다. 그리하여 R사에서 현재 CMMI 기반으로 내재화되어 있는 SW 프로세스에 대하여 ISO/IEC 90003의 관점에서 갭분석을 실시한 후, 현행 CMMI Level 3 기반 프로세스를 ISO/IEC 90003 의 관점에서 보완하기로 하였다.

2.3.2.3 EN50128 기반 TECHNIQUE/MEASURE

EN50128의 또다른 중요한 특징은 안전성을 요구하는 SW 의 개발시 준수되어야 하는 기술적 요구사항이 'TECHNIQUE/MEASURE'라는 명칭으로 SIL 별로 규정되어 있다는 것이다. 그러므로 EN50128 기반 SW 프로세스를 지원하기 위해서는 ISO/IEC 90003에서 규정된 요구사항에 추가하여 이들 'TECHNIQUE/MEASURE'를 만족시켜야 한다.

이들 기술적 요구사항은 크게 보아 SW 개발 방법론과 SW 개발 라이프사이클상의 특정 단계에 적용되어야 하는 SW 기법 및 Assessment 관련 요구사항으로 구분된다. 이들 기술적 요구사항의 가장 큰 특징은 SIL이 결정되는 순간, 해당 SIL 에 요구되는 SW 기술도 함께 결정된다는 것이다. 만약 어떤 기술의 적용을 피하고자 한다면, 해당 기술을 적용하지 않아도 되는 이유를 ISA 에게 논리적으로 제시해야 한다는 절차적 특징은 전술한 바와 같다.

R사는 현재 구조적 개발방법론 및 폭포수형 라이프사이클을 따르고 있다. 그리하여 본 프로젝트에서는 R사에 이미 내재화되어 있는 SW 개발체계를 고려하여 'TECHNIQUE/MEASURE' 요구사항 중에서 구조적 방법론 및 폭포수 모델의 구현수단으로서 EN50128 SIL4 요구 기술에 대해서 설명하는 기술설

명서(Techniques), 기술과 절차를 함께 설명하거나 기술 적용방안을 기술하는 가이드(Guides) 및 산출물 작성지원을 위한 템플릿(Templates)을 구비하기로 하였다.

2.3.2.3.1 SIL4 TECHNIQUE/MEASURE 충족을 위한 기술목록 작성

우선 EN50128에서 SIL4의 경우에 요구하는 목표와 기술을 조사 및 분석하고, R사에 적합한 SIL4 요구기술 목록을 작성하였다. EN50128의 문장은 대부분의 유사 명세와 마찬가지로 추상화의 수준이 높은 편이며, How(어떻게?) 보다는 What(무엇을?) 을 중심으로 구성되어 있는 경우가 많다. 그러므로 적절한 해석이 가해져야만 구체적인 사항(How, 어떻게?)이 식별되는 기술 요구사항에 대해서는 관련 문헌 및 현재적, 역사적 SW 사례를 조사하여 구체적인 기술 요구사항을 식별하였다. EN50128은 EN50126 을 상위개념으로 하고 있으므로, EN50126과의 연관관계 속에서 구체적인 기술 요구사항이 식별되는 경우도 있었으며, 필요에 따라서는 EN5012x 씨리즈 이외의 다른 문헌에 대한 조사도 함께 수행되었다. 때로는 산업 여건, R사의 여건 등 환경적인 여건을 고려하여 EN50128의 요건을 벗어나지 않는 한도 내에서 대안적인 기술을 모색하기도 하였다.

기술목록은 기술간 상호관계에 영향을 받을 수 있기 때문에, 기술간 상호관계에서 새로운 사항이 식별되는 경우 기술목록 자체가 업데이트되기도 한다.

2.3.2.3.2 SIL4 TECHNIQUE/MEASURE 충족을 위한 가이드라인 목록 작성

기술과 절차를 함께 설명하거나 기술 적용방안을 설명하는 항목의 경우 가이드라인으로 분리하기로 한 방침에 따라 TECHNIQUE/MEASURE 충족 기술목록을 고려한 가이드라인 목록이 작성되었다.

기술목록, 템플릿 목록 및 관련 내용의 형상변화는 가이드라인의 목록이나 내용의 형상변화를 수반한다. 그러므로 기술, 템플릿 목록 및 관련내용의 업데이트에 따라서 가이드 목록 및 내용의 업데이트가 함께 수반된다. 필요에 따라서는 ISO/IEC 90003 기반 프로세스가 변경되기도 한다.

2.3.2.3.3 SIL4 TECHNIQUE/MEASURE 충족을 위한 템플릿 목록 작성

기술 및 가이드에 따라 작성되는 SW 라이프사이클 단계별 산출물의 작성은 경험이 부족한 조직의 경우, 상당한 노력이 투입되는 작업이다. EN50128을 따르는 경우 요구사항의 정의 및 분석과 함께 상세설계서의 작성이 특히 그러하며, 나머지 산출물들도 일반적인 해외프로젝트용 SW 산출물의 평균적인 투입공수를 훨씬 상회한다. 그러므로 SW 산출물을 위한 템플릿의 활용은 개발 생산성에 커다란 영향을 줄 수 있다. 이 경우 템플릿은 EN50128 요구사항을 전적으로 반영해야만 한다. 그러므로 SW 산출물의 효율적인 작성을 지원하기 위하여 SW 라이프사이클 단계별 산출물에 대한 템플릿을 구비하기로 하였다.

EN50128의 여러 요건들 중 비교적 일반적인 사항으로 볼 수 있는 요건들은 대부분 SW 라이프사이클 산출물과 관련된 IEEE 표준에 의해서 비교적 만족스럽게 지원된다. 그러므로 SW 산출물 템플릿은 기본적으로 IEEE의 SW 표준을 적용하여 개발하기로 하였다. IEEE STD 의 경우 글로벌 SW 산업 분야에서의 De Facto Standard(사실상의 시장에서의 기준) 이기도 하다.

여기에 더하여 안전성, 신뢰성과 관련된 EN50128에 고유한 SW 관련 요건들이 EN50128 및 관련 문헌들로부터 추출된 후, SW 산출물 템플릿별로 반영하기로 하였다. R사의의 SW 개발 경험 및 기술적 제약 등이 고려된 가정(assumption) 또한 SW 템플릿에 반영되었다. 향후에는 실제 이들 템플릿을 실제로 적용하여, Best Practice 기반 산출물 샘플을 구비하는 것이 추후 과제가 될 수 있다.

2.3.2.3.4 SIL4 TECHNIQUE/MEASURE 충족을 위한 개발 Infra Structure

SW 시스템의 개발에는 다양한 도구가 적용될 수 있다. ISO/IEC 90003에서는 SW 개발 관련도구에 대한 사항들을 Infra Structure 라는 명칭으로 분류하고 있다.

-SW Testing 도구

EN50128 SIL 목표를 달성하기 위한 SW 시스템의 개발에는 SW 에 대한 다양한 검증 및 확인이 필수적이다. 본 프로젝트에서는 Source Code에 대하여 MISRA C 기반의 검증을 수행하는 Static Source Code Inspection 과 동적 소프트웨어 모듈 시험이 가능한 SW Testing Tool 을 구비하여 철저한 검증을 통한 Error Free 소프트웨어를 구현 하고자 하였다.

-소스코드 형상관리 도구

소스코드의 형상관리를 위하여 웹기반 차량 형상 변경 통제 시스템을 일부 수정하여 소스코드 형상관리가 가능하도록 하였고, 시스템 사용에 대한 지침서를 작성하였다.

-요구사항 관리 도구

I사의 요구사항 관리 툴을 도입하여 적용해본 결과 운용 및 관리에 대한 M/H 소요가 많아 현업의 반대가 있어 Spread Sheet 기반의 요구사항 추적 매트릭스 양식을 도입하였다.

2.4 SIL4 기반 프로세스 개발과정

SIL4 기반 프로세스의 개발과정에는 전형적인 SIL 인증 프로세스가 적용되었다. ISA 는 SIL4 기반 프로세스 개발 산출물 하나하나에 대해서 심층분석 및 평가를 수행하였으며, ISA의 평가결과는 개발중인 프로세스에 대한 보완 요구 피드백으로 이어지는 절차가 반복되었다. 본 연구과제 책임자는 ISA 의 피드백을 수용하여 프로세스를 수정 및 보완하였고, 필요에 따라서는 프로세스 문서 목록 자체가 변화하기도 하였으며, 경우에 따라서는 ISO/IEC 90003 기반 프로세스에 변형이 가해지기도 하였다. 이러한 ISA 의 피드백과 개발자의 보완은 수개월간 총 5회에 걸쳐 반복되었다. 이러한 일련의 피드백 및 프로세스 보완의 사이클이 마무리된 후에는 R사의 연구과제 책임자에 대한 심층 인터뷰를 포함하는 ISA 의 최종 감사가 R사에서 2일에 걸쳐 수행되었다. 그리고 모든 과정이 마무리된 후 ISA는 EN50128 SIL기반 소프트웨어 프로세스에 대해 SIL4 인증서를 수여하였다.

2.5 SIL4 기반 SW 프로세스 개발 결과

R사의 EN50128 SIL4 기반 SW 프로세스 개발 결과는 다음과 같다.

2.5.1 ISO/IEC 90003 지원 프로세스 분야 산출물

CMMI Level 3 프로세스 중 ISO 90003을 지원할 수 있는 일부 프로세스를 도입하여 인증에 대응하였다.

2.5.2 Techniques

Technique 정의서는 19종이 산출되었다.

많은 Technique들이 SW 개발 라이프사이클 단계에서 반복적으로 서로 다른 수준의 시스템에 적용된다.

Assertion and Contract 기술은 높은 수준의 안전성과 신뢰도를 요구하는 시스템에서 필수적인 Safety Concept의 표현 및 구현과 관련된 기술이다. 본 프로세스에서는 안전 요구사항 명세와 이 기술의 연관성을 강조하였으며, 소스코드 수준에서의 구현법을 기준으로 구성되어 있다.

Data Flow Diagram의 경우 구조적 방법론 요구사항이다. 시스템의 수준 별로 하향 분할하는 것이 원칙이며, SW 시스템 분석 단계에서 주로 사용되지만 HW 시스템의 데이터 흐름 분석에도 사용할 수 있다.

Finite State Machine, Sequence Diagram 및 Activity Diagram의 경우 분석 단계와 설계 단계에서 모두 사용 가능하다. 이들 기법은 흔히 UML과 관련하여 객체지향 방법론 분야에서 알려져 있지만 구조적 방법론과도 배치되지 않는다. 이들 기법은 SW 의 설계를 지원할 뿐만 아니라, SW 의 안전성을 분석하는 데에도 유용하게 사용될 수 있다.

도표 1. R 사를 위한 EN50128 SIL4 기반 프로세스 지원 Techniques

1	Technique-Data Flow Diagram	11	Technique-Transform Analysis
2	Technique-Equivalence Classes and Boundary Value Testing	12	Technique-Assertion and Contract
3	Technique-Finite State Machine	13	Technique-Software FMEA
4	Technique-Interface Testing	14	Technique-Software FTA
5	Technique-Modular Programming in C	15	Technique-Inspection
6	Technique-Performance Requirements Testing and Stress Testing	16	Technique - Walkthrough
7	Technique-Sequence Diagram	17	Technique - Technical Review
8	Technique-Structure Chart	18	Technique - Activity Diagram
9	Technique-Symbolic Execution	19	Technique - Software Fault Detection and Diagnosis
10	Technique-Transaction Analysis	20	

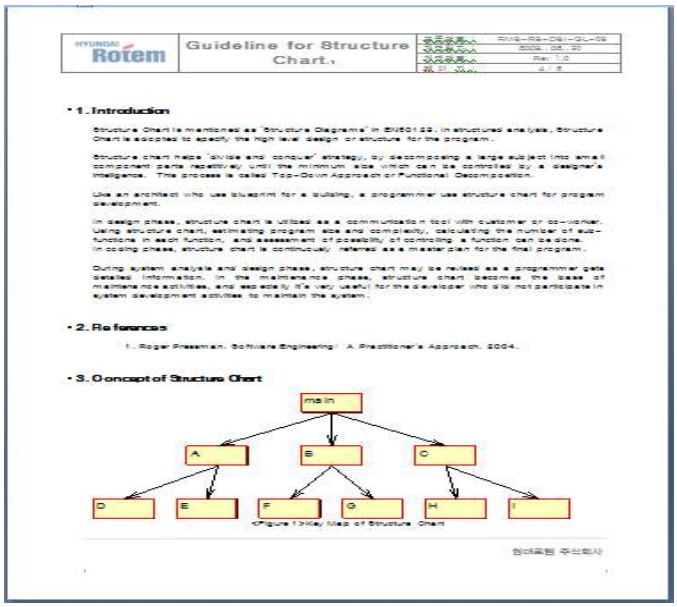
Guideline for Structure Chart.

그림2. SW Technique 의 예(SW Structure Chart)

SW FMEA와 SW FTA의 경우, 본 프로세스에서는 분석단계와 설계 단계에서 모두에서 SW FMEA와 FTA를 수행하는 것이 유용함이 강조되고 있으나, 실제로는 시스템 레벨에서 수행되고 있다. FMEA와 FTA의 시스템 수준별 반복 수행 횟수는 요구되는 SW 시스템의 안전무결성 수준에 따른다.

Inspection, Walkthrough 및 Technical Review는 검증 활동으로서 이론가들의 관점에 따라서 조금씩 다르게 이해될 수 있는 부분이다. 본 프로세스에서는 ISA의 권고를 준수하여 이들 3가지 개념을 구분하였으며, 고도의 안전성과 신뢰성이 요구되는 SW 시스템의 개발을 지원하기 위하여, 검증 프로세스의 공식화가 강조되고 있다.

Transaction/Transform Analysis는 구조적 방법론 요건으로서 분석단계의 산출물을 설계단계의 입력물로 변환시키는 일련의 기술적 과정이다.

Structure Chart(그림2 기술 가이드라인 참조)는 구조적 설계의 요건이다. 구조적 방법론에서 설계의 요건은 최하위 단위 function의 알고리즘에 대한 기술을 플로우차트나 의사코드 등으로 표현하는 수준

까지를 포함한다. 그러므로 이는 SIL을 위한 요건이 된다.

Symbolic Execution은 특정 변수가 가질 수 있는 가능한 값의 범위를 모두 조사하는 고도의 소스코드 검증기법이다. 매우 높은 수준의 안전성이나 신뢰성이 요구되는 경우에 고려될 수 있는 기법으로서 본 프로세스에 포함되어 있다. Symbolic Execution의 자동화에 대한 연구는 현재 진행형이며, 최근 부분적인 완성도를 보이는 시제품들이 나오기 시작하였다.

C 언어 기반 모듈화 프로그래밍을 위해서는 키워드 'static'의 준수가 무엇보다도 요구된다. 그러므로 'Modular Programming in C'에서는 static 키워드를 이용하여 객체지향 언어와 유사하게 모듈화하여 프로그래밍 할 수 있는 기법을 중요하게 다루고 있다. 기타 EN50128에서 요구되는 다양한 코딩 규칙들은 코딩 규칙 준수여부를 검사할 수 있는 SW Static Testing 도구를 이용하는 것으로 구성되어 있다.

Software Fault Detection and Diagnosis 기술은 역시 Safety Concept의 구현과 관련된 기술로서 SW의 fault의 탐지와 해결방안을 중심으로 구성되어 있다.

Equivalence Class, Boundary Value 및 Interface 테스팅은 단위 테스트, 통합 테스트 및 시스템 테스트 등의 모든 테스트의 경우에 적용 가능하다. Performance Requirements Testing 및 Stress Testing의 경우 시스템 개발 후 적용가능한 테스트로서 검증활동의 일부이며, 검증목적은 대부분의 경우 안전성 및 신뢰성 목표 달성 여부의 측정이다. 본 프로세스에서는 이러한 테스트의 필요성과 절차를 중심으로 구성되어 있다.

2.5.3 Guidelines

Guideline은 표2와 같이 9종이 산출되었다.

도표 2. R 사를 위한 EN50128 SIL4 기반 프로세스 지원 Guidelines

1	Guideline for EN50128 based technical requirements
2	Guideline for Software Assessment
3	Guideline for Software Safety Program
4	Guideline for SIL Based Development Lifecycle
5	Guideline for Software SIL Allocation
6	Guideline for Role Definition of SIL based Project
7	Guideline for Coding Standard of "C"
8	Guideline for Software Quality Assurance
9	Guideline for Applying Techniques & Templates

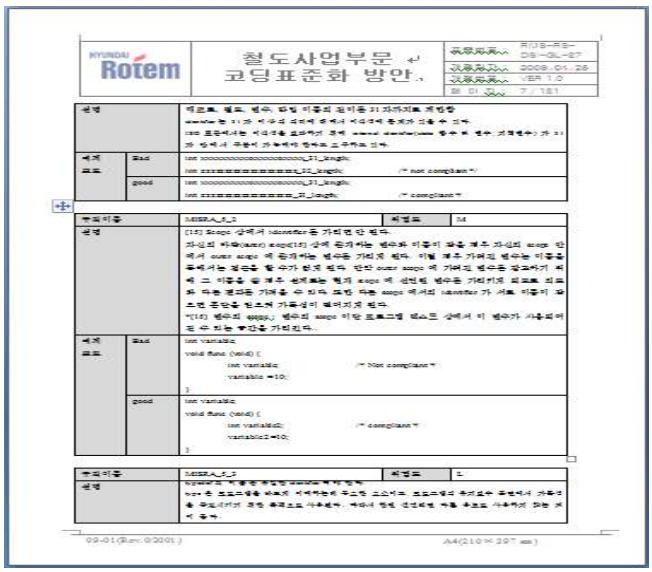

그림3. SW Guideline 의 예(MISRA C 기반 코딩 표준)

EN50128 Based Technical Requirements 가이드에서는 EN50128 SIL4 급에 해당되는 TECHNIQUE/MEASURE 요건을 구현하는 기술적 방안들이 R사의 SW 개발 라이프사이클 단계별로 구성되어 있다.

Software Assessment 가이드에서는 R사가 자체적으로 SW 를 평가 혹은 감사할 경우 따라야 할 방법론이 표현되어 있다.

SW Safety Program 가이드에서는 Safety Function 관련 SIL에 대한 설명으로부터 시작하는 SW Safety Lifecycle 단계별 활동을 표현하고, System Safety Lifecycle 과의 관계가 나타나 있다. Safety Lifecycle에 대한 표현을 위해서는 EN50128 외에 IEC61508이 함께 참조되었다.

SIL Based Development Lifecycle 가이드에서는 EN50128 기반 SIL4 급 SW 시스템의 개발시 R사에서 따라야 하는 라이프사이클을 기술하고, 각 단계별 Entry Criteria, Input, Task, Output, Exit Criteria 들을 설명하고 있다.

Software SIL Allocation 가이드에서는 SIL 을 할당하기 위한 방법이 설명되어 있다. 이 문서의 포함은 전체 프로세스의 완전성을 기하고자 하였던 ISA 의 권고에 따른 것이다.

Role Definition of SIL based Project 가이드에서는 SIL 기반 프로세스를 따를 때, 프로젝트 수행 주체들의 역할정의가 기술되어 있다. 그러므로, 이 가이드는 SIL 기반 개발 프로젝트의 경우 프로젝트 팀원들이 필수적으로 준수해야 하는 역할 정의서이다.

Coding Standard of C 는 C 언어 기반 코딩표준으로서 MISRA C를 기반 표준으로 사용하고 중요도가 높은 표준으로 "High" 로 정의하여 적용토록 하고 있고, 코딩 스타일과 함께 Language Subset을 구성한다.

Software Quality Assurance 가이드에서는 SW 품질보증 활동과 SW 검증/확인 활동을 분리하고, SW 개발시 체크되고 모니터링 되어야 하는 세부적 항목들을 SW 개발 라이프사이클 단계별로 표현하였다.

Applying Techniques & Templates 가이드에는 본 연구과제를 통해서 산출된 Technique 정의서와 템플릿을 적용하는 방법이 SW 개발 라이프사이클 단계별로 구체적으로 기술되어 있다.

2.5.4 Templates

템플릿은 39종이 산출되었다.

대부분의 템플릿은 IEEE 요건을 준수하고 있다. 또한 모든 템플릿은 EN50128 및 ISO 90003 의 요건들을 문장별로 분석하여 충족시키고 있다. EN50128에서 특정 템플릿을 위한 요구사항을 만족시키지 못하는 경우 대안적인 표준을 따르고 ISA 의 승인을 받았다.

템플릿의 경우 수출용 SW 시스템 개발의 일반적인 항목인 SW Quality Assurance, SW Configuration Management, SW Verification and Validation 에 대한 계획서를 포함한다. 템플릿이 여러 개라 하더라도 필요에 따라서 문서는 통합될 수 있다. 예를 들어 A사의 SIL2 프로젝트의 경우 발주자의 요구에 따라 SW Verification 계획과 SW Validation 계획을 통합하였으며, ISA도 이에 동의한 사례가 있다.

SW Requirements에서 가장 중요한 부분은 시스템 수준의 Safety Concept을 SW 수준의 Safety Concept으로 사상시키는 과정이다. 이러한 사상과정에서 발생할 수 있는 Safety Concept 의 붕괴에 대비하기 위하여 EN51028에서는 SW Requirements Spec. 에 더하여 SW Requirements 에 대한 테스트와 검증 보고서를 따로 분리시키고 있다. SW Verification 의 개념에 대해서는 많은 문헌과 규격에서 조금씩 상이한 관점을 취하고 있으나 본 프로젝트에서는 검증활동을 각종 문서 산출물에 대한 inspection 및 walkthrough 를 모두 포함하는 리뷰 활동에 테스트 활동까지 포함하는 활동으로 가정하였다.

도표 3. R사를 위한 EN50128 SIL4 기반 프로세스 지원 Templates

1	Template-Software Assessment Report	20	Template-Software Source Code Verfication Report
2	Template-Data Preparation Plan	21	Template-Software Validation Plan
3	Template-Software Quality Assurance Plan	22	Template-Software Validation Report
4	Template-Software Change Record	23	Template-Software Verification Plan
5	Template-Software Configuration Management Plan	24	Template-Software Design Specification
6	Template-Data Test Report	25	Template-Software Module Design Specification
7	Template-Software Hardware Integration Test Report	26	Template-Software Maintenance Plan
8	Template-Software Module Test Report	27	Template-Software Project Management Plan
9	Template-Application Requirement Specification	28	Template-Software Architecture Specification
10	Template-Software Requirement Specification	29	Template-Software Maintenance Record
11	Template-Data Test Plan	30	Template-Software Source Code Headers
12	Template-HW SW Integration Test Plan	31	Template-Software Test Execution Trace Report
13	Template-Software Architecture and Design Verification Report	32	Template-Software Test Incident Report
14	Template-Software Integration Test Plan	33	Template-Software Test Log
15	Template-Software Integration Test Report	34	Template-Software Test Procedure Specification
16	Template-Software Module Test Specification	35	Template-Software Test Summary Report
17	Template-Software Module Verification Report	36	Template-Software Verification Report
18	Template-Software Requirements Test Specification	37	Template-Software Bug Tracking Log
19	Template-Software Requirements Verification Report	38	Template-Software Version Description Document

SW Architecture와 Design Spec.은 서로 연결되는 것으로 보일 수도 있으나, Architecture 단계에서는 상위수준 System과의 관련성 속에서 Safety Concept이 구현된 모습이 표현되어야 한다는 특징이 있다. 그러므로 처음부터 안전하고 신뢰성이 높았던 시스템이 아니라면, SW Architecture Spec.의 구성을 위해서는 대부분 심각한 설계 혹은 설계변경 회의를 수 차례 반복하게 된다.

SW Design Spec.의 경우 IEEE 1016을 충분히 준수한다면 Module Design Spec.을 따로 작성할 필요가 없을 정도가 된다. 하지만, 국내 현장의 업무환경을 고려한다면 소위 예비설계라는 것과 상세설계라는 것이 분리되는 것이 현실적이다. 상세 설계 단계에서는 취하위 수준 function의 순차/선택/반복을 포함하는 알고리즘까지 기술해주어야 구조적 방법론 요건을 만족시킬 수 있음을 아울러 밝힌다.

2.5.5 SW 분석 및 설계의 예

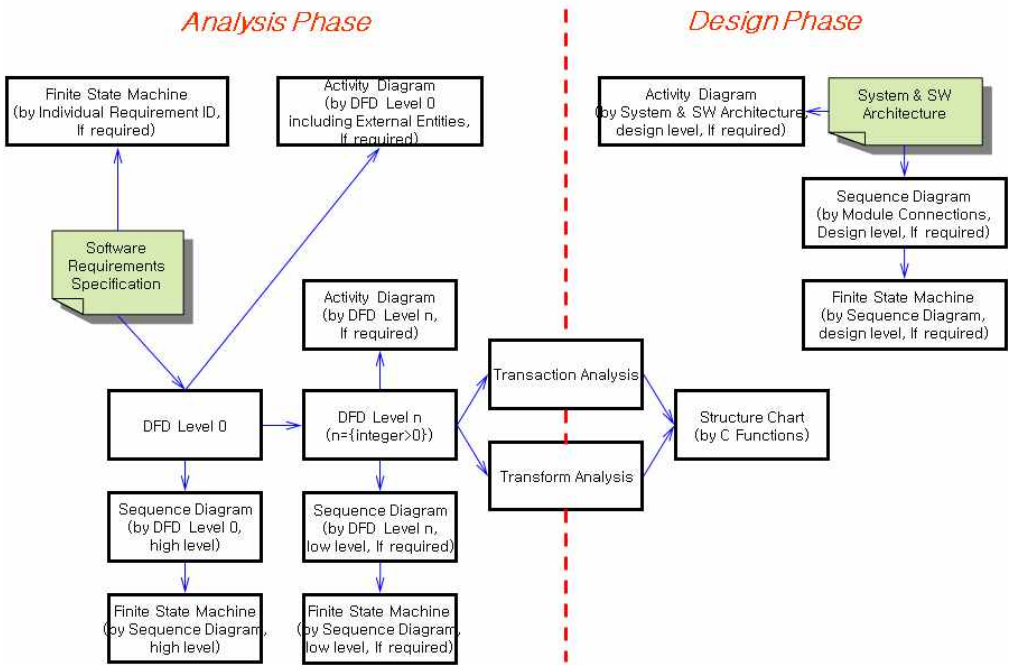

그림 4. SW 분석 및 설계 예

본 절에서는 R사의 SIL4 기반 구조적 방법론에 기반하여 SW 시스템을 분석 및 설계하는 예를 그림 4에 제시한다. 여기서 예로서 제시하는 단계는 SW Requiremetns Spec. 이 완료되었음을 가정한다.

분석단계의 초기에는 개별 요구사항으로부터 Finite State Machine 의 작성이 가능하다. 만약 Data Flow Diagram(이하 DFD)이 작성된 후라면 DFD Level 0로부터 Sequence Diagram의 작성이 가능하며, 이 Sequence Diagram 으로부터 Finite State Machines의 작성이 가능하다.

분석단계에서 만약 DFD가 하향분할된 경우라면 이들 하향분할된 DFD로부터 low level Sequence Diagram의 작성이 가능하며, 이 Sequence Diagram으로부터 low level Finite State Machine의 작성이 가능하다. 또한 하향분할된 DFD로부터는 Activity Diagram 의 작성이 가능하다.

하향분할된 Data Flow Diagram은 이후 Transaction/Transform Analysis를 거쳐 Structure Chart로 전환된다.

또한 설계단계 초기의 경우 System 과 SW의 Architecture로부터 Activity Diagram 과 Sequence Diagram의 작성이 가능하며, 이 Sequence Diagram으로부터 역시 Finite State Machine의 작성이 가능하다.

2.6 SIL 관리지원시스템

EN50128의 모든 기술적/절차적 요건을 ISO/IEC 90003 요건과 통합하여 SW 개발 라이프사이클 단계별로 기술적 산출물들을 생성하고 관련된 SW 품질활동을 진행한다는 것은 그리 용이한 일이 아니며, 관련된 시간과 노력 또한 만만치 않다.

그러므로 본 프로젝트에서는 SIL 표준 Spec을 준수하는 시스템 개발 활동 수행 시 필요한 프로세스, 기술요건, 템플릿, 산출물 등의 검색, 접근, 활용을 용이하게 하는 WEB 기반 시스템을 함께 개발하였다.

본 온라인 시스템에는 모든 SIL 4 기반 기술서, 템플릿 및 가이드가 ISO/IEC90003 기반 품질활동 및 프로세스 활동을 지원하는 문서, 양식, 기술서, 절차서 등과 함께 통합되어 있다.

본 시스템은 EN50128의 SIL 관련 명세를 체계적으로 구조화하여, 프로젝트 이해관계자들이 자신들이 요구하는 기술 및 프로세스 항목에 SW 라이프사이클 단계별로 쉽게 접근할 수 있도록 하였으며, 사

용자가 원하는 산출물 템플릿이나 작업 수행 절차 등을 최단 시간에 참조, 활용할 수 있도록 다양한 검색 메커니즘을 구현하였다.

그러므로 프로젝트 이해관계자들에게, SIL 기반 프로젝트의 수행에 있어 EN50128 이나 ISO/IEC 90003에 대한 조사활동을 최소화하면서도, SW 개발 라이프사이클 단계별로 요구되는 기술, 템플릿, 특정 명세, 샘플, 절차 등에 보다 용이하게 접근할 수 있는 환경을 제공할 수 있게 되었다.

그림 5. SIL 기반 프로세스 지원 시스템 메인화면

3. 결론

본 연구과제는 EN50128에서 제시하는 요건을 어떻게 소프트웨어 프로세스로서 지원 및 표현할 수 있는지를 인식하게 된 좋은 계기가 되었고, 어떤 부분을 ISA가 초점을 가지고 평가를 하는지 알게 되는 좋은 기회가 되었다.

인증을 득한 EN50128 SIL4 기반 소프트웨어 프로세스를 조속히 전장품에 적용하여 Generic Application에 적용토록 하고, 실제 적용시 보완해야될 항목이 있을 경우 프로세스를 보완할 예정이며, 본 연구 과제를 통하여 철도차량 제작사로서 소프트웨어 SIL 관리능력을 시행청에 입증하는 계기가 될 것으로 생각이 들면서, 본 논고를 마치고자 한다.

참고문헌

1. EN 50126. Railway applications. The specification and demonstration of reliability, availability, maintainability and safety (RAMS)

2. EN 50128. Railway applications. Communications, signalling and processing systems. Software for railway control and protection systems

3. IEC 61508. Functional Safety of electrical/electronic/programmable electronic safety-related systems

4. Ian Sommerville. Software Engineering. 8th Edition. Addison-Wesley. 2005.

5. Roger S. Pressman. Software_Engineering - A Practitioner's Approach. 5th Edition. McGraw_Hill. 2001.