'2000 추계학술발표회 논문집
한국원자력학회

원전 소프트웨어의 안전등급과 확인 및 검증 활동의 차등 적용에 관한 연구

Study on Safety Classifications of Software used in Nuclear Power Plants and Distinct Applications of Verification and Validation Activities in Each Class

김복렬, 오성헌, 황희수, 김대일

한국원자력안전기술원
대전광역시 유성구 구성동 19

요 약

본 논문에서는 원전의 계측제어계통 및 소프트웨어에 대한 안전등급 분류내용과 소프트웨어에 관한 규제입장을 제시하고 소프트웨어 품질보증 확보를 위한 중요한 요소인 확인 및 검증 활동들을 소프트웨어 등급별로 차등 적용하는 방안을 제안한다. 즉 원전 안전성 계측제어계통을 안전등급 분류기준에 따라 안전등급 IC-1, IC-2, IC-3, 그리고 Non-IC로 분류하였으며, 소프트웨어는 안전중요도에 따라 안전성-필수 소프트웨어, 안전성-관련 소프트웨어, 그리고 비안전성 소프트웨어로 분류하였다. 이 같은 안전등급 분류기준에 근거하여 각 등급별 소프트웨어 확인 및 검증 활동의 정도를 서로 다르게 차등화 하였다. 또한 안전성 계측제어계통에 사용되는 소프트웨어를 설계 및 구현하는 관점에서 신개발 소프트웨어와 기성 소프트웨어로 분류하였고, 각 유형별의 소프트웨어에 대한 규제입장을 제시하였다.

Abstract

This paper describes the safety classification regarding instrumentation and control (I&C) systems and their software used in nuclear power plants, provides regulatory positions for software important to safety, and proposes verification and validation (V&V) activities applied differently in software classes which are important elements in ensuring software quality assurance. In other word, the I&C systems important to safety are classified into IC-1, IC-2, IC-3, and Non-IC and their software are classified into safety-critical, safety-related, and non-safety software. Based upon these safety classifications, the extent of software V&V activities in each class is differentiated each other. In addition, the paper presents that the software for use in I&C systems important to safety is divided into newly-developed and previously-developed software in terms of design and implementation, and provides the regulatory positions on each type of software.

1. 서 론

디지털 컴퓨터 및 정보처리 기술의 급속한 발전에 힘입어 산업계 전반에 걸쳐서 아날로그기술의 쇠퇴와 디지털기술로의 대전환이 이루어지고 있다. 그러나 컴퓨터기술의 빠른 확산에도 불구하고 원전에서의 컴퓨터기술 적용은 그 보수성을 고려할 때 안전성과 신뢰도 확보에 따른 많은 의문점을 제기하여 왔다. 즉 원전 계측제어계통의 설계를 구현하는 하드웨어와 소프트웨어가 주변 환경요인이나 설계 및 프로그램 오류에 취약하여 공통유형고장을 일으킬 가능성이 있는 것으로 지적되었고, 그것이 설계 및 규제에 관한 안전현안으로 대두되었다. 또한 컴퓨터기술은 각종 자원(예, CPU, 메모리, I/O 등)을 공유할 수 있는 장점이 있지만, 한편으로는 공통유형 소프트웨어오류로 인해 하드웨어로써 구현된 다중성 설계를 파기시킬 수 있다는 의견들이 제기된 바 있다. 이와 같은 안전성 현안들을 해결하기 위한 가장 설득력 있는 방안으로는 철저한 품질보증과 심층방어 및 다양성 설계기법을 채택하는 것으로 알려져 있다.[1,2]

본 논문에서는 원전의 안전성 계측제어계통에 사용된 소프트웨어에 대한 안전등급 분류내용과 소프트웨어에 관한 규제입장을 제시하고, 그리고 소프트웨어 품질보증 확보를 위한 중요한 활동인 확인 및 검증 활동들을 소프트웨어 등급별로 차등 적용하는 방안을 제안한다.

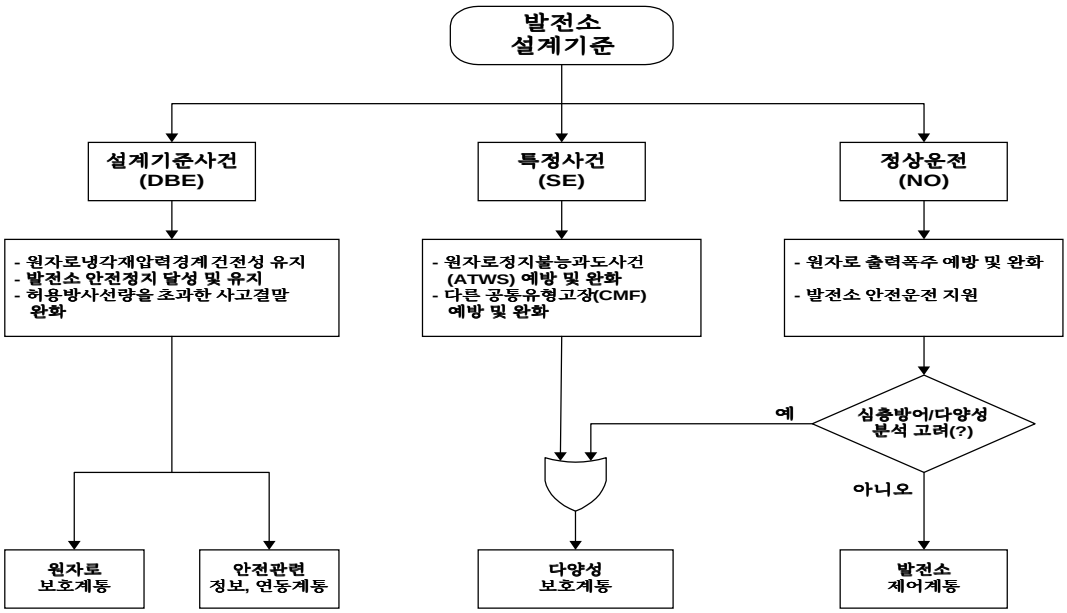
2. 본 론

2.1 원전 계측제어계통의 안전등급 분류

원자력발전소(이하 "원전") 계측제어계통의 소프트웨어를 포함한 모든 구조물, 계통 및 기기는 안전 기능과 중요도에 근거하여 안전등급이 부여되고, 그에 적합한 명세서, 설계, 검증, 품질보증, 제작, 설치, 보수 및 시험 등의 요건들을 만족하여야 한다. 원전 계측제어계통은 과학기술부고시 제94-10호, "원자력시설의 안전등급과 등급별 규격에 관한 규정"과 차세대원자로 상세안전요건(3.1.2.3)절, "안전등급 3 분류기준"에서 안전등급 3으로 분류되어 있다. 상기의 과학기술부고시와 상세안전요건의 분류는 원자로용기 압력경계를 기준으로 하였기 때문에, 계측제어계통의 고유한 특성을 고려한 실질적인 안전등급 분류 체계와 기준이 필요하다.

원전 계측제어계통의 안전등급 분류기준은 그림 1과 같이 발전소 설계기준(plant design bases)에 근거한다.[3] 원자로보호계통과 안전관련 정보·연동계통 등은 발전소 설계기준사건(DBE)에 대비하여 설계되며, 다양성 보호계통은 원자로정지불능과도사건(ATWS) 혹은 다른 원자로보호계통의 공통유형고장(CMF)과 같은 특정사건에 대비하여 설치된다. 발전소 제어계통은 발전소 정상운전을 위해 필요한 공정 설비를 계측 및 제어하는 계통들을 포함한다. 그림 1의 발전소 설계기준에 근거하여 계측제어계통의 안전등급을 분류하면 ① 안전에 중요한 계측제어계통[이하 "안전성 계측제어계통(I&C Systems Important to Safety)"]과 ② 안전에 중요치 않는 계측제어계통[이하 "비안전성 계측제어계통(I&C Systems not Important to Safety)"]으로 분류된다. 안전성 계측제어계통은 다시 안

전관련 계측제어계통(Safety-Related I&C Systems)과 비안전관련 계측제어계통(Non-Safety-Related I&C Systems)으로 분류된다.



<그림 1> 발전소 설계기준 및 계측제어계통의 안전등급 블록

안전관련 계측제어계통은 안전등급 IC-1, IC-2 및 IC-3으로 분류되고, 비안전관련 계측제어계통은 Non-IC로 분류된다.[4] 안전등급 IC-1과 IC-2는 발전소 설계기준사건에 대비하여 안전기능을 수행하고, 안전등급 IC-3은 특정사건(예, ATWS)에 대비한 안전관련기능을 수행하는 계측제어계통이다. 비안전등급(Non-IC)은 발전소 정상운전 중에 사용되는 계측제어계통으로써, 예상운전과도사건(AOO) 또는 사고 후에 안전관련기능을 직접 수행하지 않지만, 그것의 잘못된 기능 또는 고장이 발전소 안전에 심각한 영향을 주는 발전소 공정계통 및 설비들을 제어하는 계통이다. 이와 같은 계측제어계통의 안전등급체계와 각 등급에 속한 하부 계통들을 구체적으로 기술하면 다음과 같고 이를 그림 2에 나타냈었다.

(1) 안전등급 IC-1 계측제어계통

안전등급 IC-1은 원자력발전소 설계기준사건(DBE)에 대비하여 안전기능들을 수행하는 원자로 보호계통, 즉 원자로트립계통(RTS) 및 공학적안전설비작동계통(ESFAS)과 발전소 안전정지·냉각 기능을 수행하는 계측제어설비, 사고후감시(PAMI) 유형 A 변수 계측설비, 그리고 필수보조지원 계통을 포함한다.

(2) 안전등급 IC-2 계측제어계통

안전등급 IC-2는 안전관련 정보계통(PAMI 유형 B 변수 등), 안전관련 연동계통, 주제어실, 원

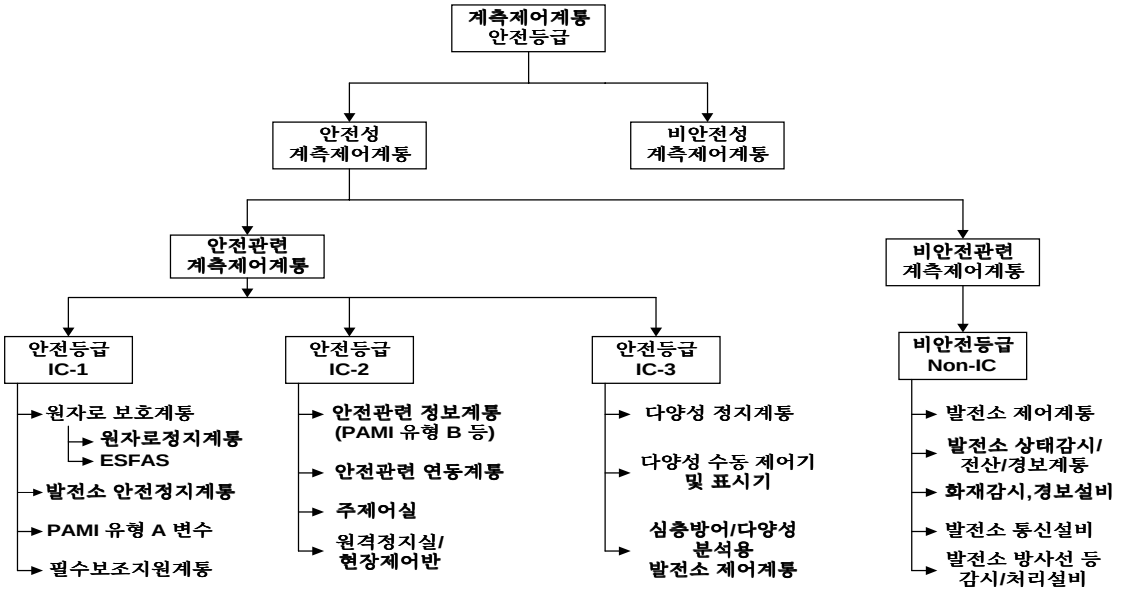
격제어실 및 현장제어반을 포함한다. 안전관련 정보계통은 발전소 정상운전, 예상운전과도사건, 그리고 사고 시에 안전한 운전을 위해 필요한 정보와 안전계통 수동 개시 및 제어를 위한 정보를 제공한다. 또한 안전관련 정보계통은 발전소 정상운전 동안에는 안전계통의 정상상태에 관한 정보뿐만 아니라 우회상태 및 작동불능상태에 관한 정보를 제공한다. 안전관련 연동계통은 특정사건의 발생 가능성을 줄이거나, 또는 사고 시에 안전계통을 가용한 상태로 유지하는 설비를 포함한다. 안전관련 연동계통은 안전관련 조치가 사고 이전 또는 사고 예방을 위해 취해진다는 점에서 원자로보호계통과는 다르다.

(3) 안전등급 IC-3 계측제어계통

안전등급 IC-3은 다양성 보호계통 또는 계측제어설비, 그리고 심층방어 및 다양성 분석용의 발전소 제어계통을 포함한다. 다양성 보호계통 또는 계측제어설비는 발전소 특정사건, 즉 원자로 정지불능과도사건(ATWS) 완화 또는 원자로보호계통의 공통유형고장에 대비하여 원자로보호계통과는 다른 방법으로써 안전기능을 수행하는 설비이다. 심층방어 및 다양성 분석용 제어계통은 원자로보호계통의 심층방어 및 다양성 분석에서 고려되는 높은 품질의 발전소 제어계통을 포함한다.

(4) 비안전등급 Non-IC 계측제어계통

비안전등급 Non-IC은 발전소 제어계통, 발전소 상태감시 및 진단설비, 전산설비 및 경보계통, 화재감지·경보설비, 통신설비, 방사성 폐기물처리 및 사용후 핵연료처리 등의 계측제어 설비들을 포함한다. 발전소 제어계통은 예상운전과도(AOO) 또는 사고 이후에 안전기능을 직접적으로 수행하지 않지만, 그 잘못된 기능 또는 고장이 발전소 안전에 심각한 영향을 주는 발전소 공정설비를 제어하는 계통들을 포함한다.



<그림 2> 계측제어계통의 안전등급 분류체계와 관련 하부계통

2.2 소프트웨어 안전등급 분류

소프트웨어는 계측제어계통의 일부 구성요소이므로, 안전성 계측제어계통에 사용되는 소프트웨어 안전등급은 수행되어야 할 기능 및 성능의 안전 중요도에 따라 3개 등급, 즉 안전성-필수 소프트웨어, 안전성-관련 소프트웨어, 그리고 비안전성 소프트웨어로 분류하였다.

(1) 안전성-필수 소프트웨어(safety-critical software)

안전성-필수 소프트웨어는 안전등급 IC-1 계측제어계통에 사용된 소프트웨어로서 가장 엄격한 소프트웨어 품질보증요건과 기술기준을 적용하며, 심층방어 및 다양성 분석결과에 따라 필요하다면 다양성 백업설비가 요구될 수 있다. 원자로보호계통의 소프트웨어 개발에 따른 안전성 분석에서는 소프트웨어 구성요소가 원자로보호계통에 미치는 위험요소 또는 위험도 분석들을 수행하여 공통유형고장 발생 가능성을 최소화해야 한다. 소프트웨어 확인 및 검증 활동들은 개발조직과는 기술, 조직, 책임, 권한 및 재정적으로 독립된 확인 및 검증조직에 의해 수행되어야 한다.[5,6]

(2) 안전성-관련 소프트웨어(safety-related software)

안전성-관련 소프트웨어는 안전등급 IC-2 및 IC-3 계측제어계통에 사용되는 소프트웨어로서 안전성-필수 소프트웨어보다는 완화된 품질보증요건과 기술기준을 적용 받게 된다. 안전성 분석에서는 안전성-관련 소프트웨어가 안전성-필수 소프트웨어에 미칠 수 있는 위험요소를 찾아서 시정하여야 하며, 확인 및 검증 활동은 개발조직과는 적절하게 독립된 확인 및 검증조직에 의해 수행되어야 한다.

(3) 비안전성 소프트웨어(non-safety software)

비안전성 소프트웨어는 비안전등급 Non-IC 계측제어계통에 사용되는 소프트웨어이며, 발전소 안전운전에 적합한 수준의 품질보증요건과 기술기준에 맞게 개발되어야 한다. 만약 비안전성 소프트웨어가 안전성-필수 또는 안전성-관련 소프트웨어와 함께 동일한 컴퓨터-기반 시스템에 적재(loading)되어 실행된다면 그 소프트웨어가 다른 상위의 소프트웨어들에 미칠 수 있는 위험요소를 분석하고 필요하다면 적절한 보완(예, 격리)을 하여야 한다.

2.3 안전성 소프트웨어에 관한 규제입장

안전성 소프트웨어는 상기 2.2절에 기술된 바와 같이 3개의 등급으로 분류되었다. 소프트웨어는 계측제어계통의 일부 구성요소이므로, 그 계통에 적용되는 품질보증요건과 기술기준을 그대로 적용 받게 된다. 다시 말하면, 안전성 계측제어계통에 사용되는 소프트웨어를 포함한 모든 구조물, 계통 및 기기는 수행되어야 할 안전 기능과 중요도에 근거하여 안전등급이 부여되고 그에 적합한 명세서, 설계, 검증, 품질보증, 제작, 설치, 보수 및 시험 등의 요건들을 만족하여야 한다. 따라서 소프트웨어 안전등급별로 서로 다른 규제요건과 기술기준을 적용해야 한다.

그러면 서로 다른 규제요건과 기술기준을 적용함에 있어서 소프트웨어 안전등급별로 어떻게 차등적으로 적용할 것인가 하는 문제이다. 소프트웨어 설계특성은 하드웨어 특성처럼 가시적이지 못하고 설계공정 결함들이 일반적으로 최종결과물에서 확인될 수 있다는 점 때문에 소프트웨어 확인 및 검증과 같은 품질활동들을 보다 강화하고 있다. 소프트웨어 안전등급별 규제요건 및 기술기준 적용의 차등화는 소프트웨어 생명주기에 따른 안전성분석활동, 확인 및 검증활동, 형상관리활동, 그리고 품질보증활동에서 찾을 수 있다. 그 차등화는 안전등급의 취지에 맞게 원전 사업자 또는 설계자가 원칙적으로 결정해야 할 사항이며 안전규제관점에서 인허가 승인을 받아야 한다.

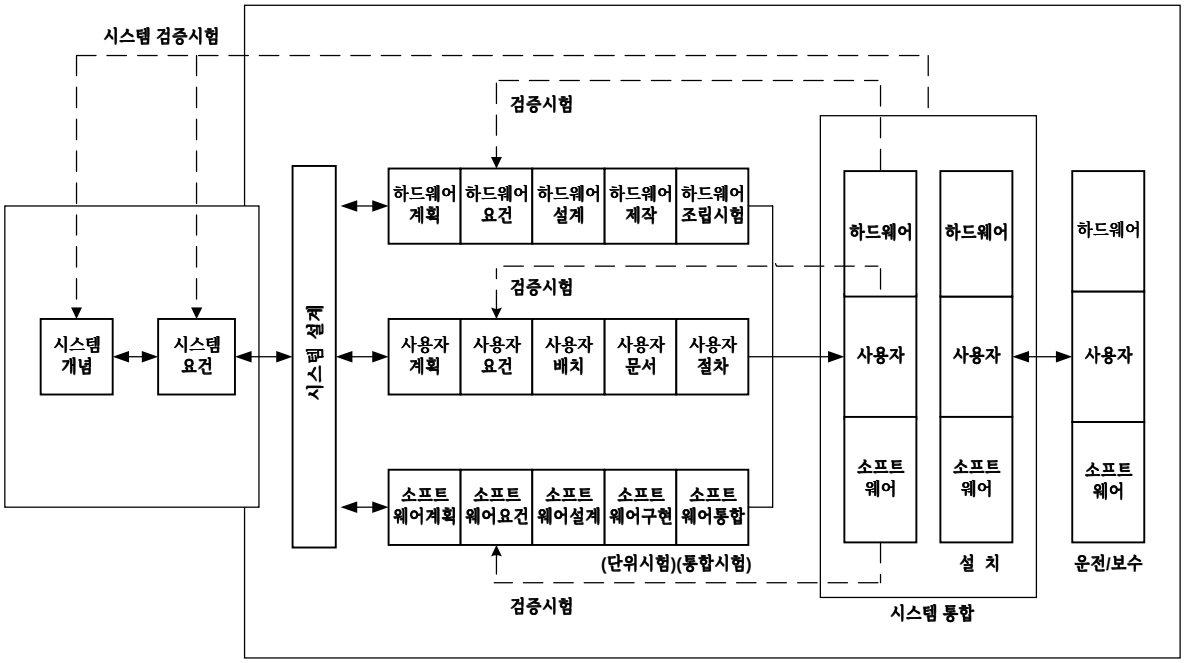
한편 소프트웨어 개발관점에서 분류되는 신개발 소프트웨어와 기성 소프트웨어는 원칙적으로 그것이 적용되는 계측제어계통의 규제요건과 기술기준을 그대로 적용 받게 된다. 이와 관련된 세부적인 규제입장은 아래와 같다.

(1) 신개발(newly-developed) 소프트웨어에 대한 규제입장

원전 안전성 소프트웨어의 개발에 따른 생명주기는 잘 정의된 생명주기 모델을 기반으로 선정되어야 한다. 그 모델은 개발사업 규모, 생명주기 활동 및 수행되는 순서에 따라서 조금씩 달라질 수 있다. 따라서 소프트웨어 개발자는 관련기술기준(예, IEEE 1074)에 근거하여 적절한 소프트웨어 생명주기를 선정하고, 그에 따라 생산되어야 할 설계문서 또는 제품들을 규정하여야 한다.[7]

그림 3은 안전성 디지털 계측제어계통의 개발에 따른 생명주기 참조모델이다.[8] 이 모델은 안전규제 관점에서 설정된 것이며, 시스템 개념, 요구사항, 설계, 통합 및 검증, 그리고 운전 및 보수 단계들로 구성되어 있다. 시스템 설계단계를 더 세분하면 계획작성, 요구사항, 설계, 구현, 그리고 통합으로 구분된다. 소프트웨어 생명주기는 시스템 개념 및 요건으로부터 할당된 소프트웨어 요구사항을 만족할 수 있는 소프트웨어 계획단계에서부터 운전 및 보수단계까지를 의미한다. 그러나 만약 소프트웨어 개발자가 그림 3의 참조모델과는 다른 생명주기를 채택하는 경우에는 그 채택된 생명주기가 참조모델과 비교하여 유사한 단계와 활동들을 포함하고, 그리고 설계문서와 제품들을 생산한다면 새로 채택된 모델의 사용이 가능하다. 그러한 경우 참조모델의 각 단계와 활동에 대한 관련 설계문서 또는 제품을 상호-비교할 수 있는 도표를 작성하여 그 적합성을 입증해야 한다.

신개발 소프트웨어에 대한 규제입장은 첫째, 소프트웨어 개발공정에 따른 활동들을 관리하는 소프트웨어 공정계획들이 작성되었는지를 확인하고, 둘째 그 소프트웨어 공정계획이 소프트웨어 생명주기 공정이행 활동들에서 준수되었는지를 확인하며, 셋째 소프트웨어 생명주기 공정활동을 통해 생산되는 공정설계 결과물, 즉 설계문서와 제품들의 적합성을 평가한다.



<그림 3> 안전성 계측제어계통의 개발 수명주기(NUREG-0800 발췌)

(2) 기성(previously-developed) 소프트웨어에 대한 규제입장

기성 소프트웨어는 어떤 특정한 응용분야에 맞게 전용으로 개발되지 않고 일반시장의 수요를 충족할 수 있도록 범용으로 개발되고 상용화된 소프트웨어(COTS) 또는 특정한 설비를 대상으로 개발되어 지적 소유권(proprietary)이 보장된 소프트웨어를 의미한다. 원전 안전성 계측제어계통은 신개발 소프트웨어와 기성 소프트웨어를 사용할 수 있다. 이러한 경우 신개발 소프트웨어는 상기 (1)절의 규제입장을 만족하면 되지만, 기성 소프트웨어는 그 제품의 설계활동과 소스 코드 등을 포함한 수집된 각종 설계문서를 신개발 소프트웨어의 생명주기에 따른 활동과 설계문서로 서로 연관해서 매핑(mapping)을 해야 한다. 즉 그림 3의 생명주기에 따른 활동과 설계문서를 확인할 수 없는 기성 소프트웨어는 누락된 개발공정 또는 부적절한 설계문서를 보상할 인자(factor)들을 고려해야 한다. 그리고 신개발 및 기성 소프트웨어들이 통합된 시점부터는 신개발 소프트웨어의 생명주기에 따라 검증되어야 한다. 그러나 만약 기성 소프트웨어의 일부분이 시스템 요구사항을 만족하기 위하여 수정 또는 추가되는 경우에는 신개발 소프트웨어 생명주기에 따라 역 공학(reverse engineering) 검증이 이루어져야 한다.

기성 소프트웨어가 신개발 소프트웨어와 통합되기 전까지 검증은 다음과 같은 사항들을 고려하여야 한다.[9,10]

① 기성 소프트웨어가 적용되는 특정시스템[이하 “모(mother) 계통”]의 요구사항을 정의하고, 그 소프트웨어가 정의된 요구사항을 만족할 수 있는 기능을 갖고 있는지를 파악하고, 그리고 그 기능이 안전성에 미치는 영향을 평가하여야 한다.

② 기성 소프트웨어는 모(mother) 계통의 안전등급에 맞게 검증되고, 신개발 소프트웨어와 동등한

수준의 평가가 수행되어야 한다. 만약 필요하다면, 그 소프트웨어의 모든 명세서가 평가될 수 있도록 역 공학(reverse engineering) 검증이 이루어져야 한다.

③ 기성 소프트웨어의 일부 모듈이나 코드체계가 수정되어야 한다면 그 소프트웨어에 관한 설계 문서와 소스 코드가 필요하며, 그리고 신개발 소프트웨어의 생명주기에 따른 역 공학 검증이 이루어져야 한다.

④ 기성 소프트웨어는 그 본래의 사용 취지에 적합한 소프트웨어 공학관례와 품질보증 기술기준에 따라 개발되었고 유지되어야 한다. 기성 소프트웨어의 품질과 개발공정을 평가할 때 필요한 정보는 그 소프트웨어를 요구되는 수준의 품질까지 평가할 수 있을 만큼 충분하여야 한다.

⑤ 기성 소프트웨어의 기능들은 모(mother) 계통의 요구사항 명세서 또는 소프트웨어에 관한 다른 모든 요구사항을 만족하여야 한다.

⑥ 모(mother) 계통의 요구사항 명세서에서 요구하지 않는 기성 소프트웨어의 기능들은 유발(invoke)되어서는 안되며, 잘못된 입력, 인터럽트 그리고 잘못된 사용으로 인해 안전성 기능에 악영향을 미쳐서는 안된다. 특히, 사용자 또는 다른 소프트웨어가 기성 소프트웨어의 특정한 모듈을 호출할 때에 연관된 인터페이스는 정확하게 확인되고 철저하게 검증되어야 한다.

⑦ 기성 소프트웨어의 무의도된 기능 또는 의도된 기능이 실패한 경우 모(mother) 계통의 품질 및 안전성에 미치는 영향을 위험요소분석 또는 위험도분석을 통해 확인하여야 한다.

⑧ 만약 기성 소프트웨어 개발공정에서 생산되는 일부 설계문서가 없거나 또는 소스 코드의 입수가 곤란하여 운전경험을 보상 인자로 고려한 경우, 운전이력 및 고장율에 관한 충분한 정보가 제시되어야 한다. 운전경험은 운전시간, 오류보고서, 또는 시스템 운전에 사용된 배포(release) 이력을 기반으로 하여 적절히 평가되어야 한다. 운전경험의 반영에서는 특히 기성 소프트웨어가 유사한 운전조건에서 사용되었다는 것을 전제로 하여야 한다.

⑨ 운전경험의 반영에는 다음과 같은 각 인자를 적절하게 평가하여야 한다.

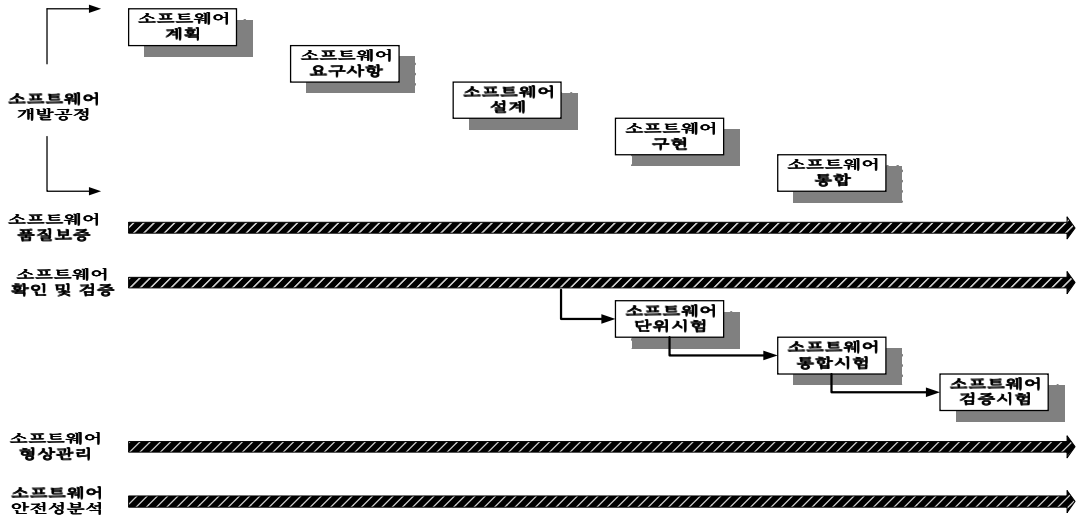
- 기성 소프트웨어가 산업계에서 얼마나 많이 사용되고 있는지를, 특히 사용자 및 버전 관점에서 확인한다.
- 기성 소프트웨어를 종전에 사용했던 시스템과 그 소프트웨어를 새로 적용하려는 모(mother) 계통이 어느 정도로 유사한지를 평가한다.
- 기성 소프트웨어가 얼마나 많이 모(mother) 계통에서 사용되는지를 평가한다.
- 기성 소프트웨어가 얼마나 많은 운전시간을 갖고 있는지를 평가한다. 운전시간에 대한 기록은 연속운전시간, 총 운전년수, 그리고 하루 운전시간을 보여야 한다.
- 고장 이력과 어떤 결함들이 있었는지, 그리고 시정 결과를 평가한다.
- 종전의 시스템에서 발생했던 문제점이 새롭게 적용하려는 모(mother) 계통에서도 유사하게 발생할 가능성이 있는지를 평가한다.

⑩ 만약 기성 소프트웨어의 개발공정 문서 또는 소스 코드 입수가 거의 곤란하고, 그리고 상기 ⑧항의 운전경험 평가를 위한 충분한 정보가 없는 기성 소프트웨어는 안전성-필수 또는 안전성-관련 소프트웨어로서 사용될 수 없다. 만약 이와 같은 기성 소프트웨어를 비안전성 소프트웨어에 적용하려면, 기성 소프트웨어의 고장에 따른 안전성 영향분석을 수행하여야 한다.

⑪ 기성 소프트웨어의 검증과정에서 발견된 오류들은 철저히 분석되고 수락 절차에 반영되어야 한다.

2.4 안전성 소프트웨어에 대한 확인 및 검증 활동의 차등 적용

일반적으로 소프트웨어 개발사업은 소프트웨어 개발에 따른 개발공정(development process), 즉 소프트웨어 생명주기 공정들과 그 공정들이 관련된 요구사항, 법규, 또는 기술기준을 준수하는지를 확인 및 검증하는 보증공정(assurance process)으로 구분된다. 소프트웨어 보증공정은 개발공정을 계획·관리하는 공정이며 사업관리, 소프트웨어 품질보증, 소프트웨어 확인 및 검증, 소프트웨어 형상관리, 그리고 소프트웨어 안전성분석 활동들을 포함한다. 소프트웨어 보증공정은 소프트웨어 개발공정과는 별도로 동시에 수행되며, 소프트웨어 개발공정을 반복적으로 요구할 수 있다. 다시 말하면 소프트웨어 보증공정은 소프트웨어 개발공정과 그 출력물에서 문제점을 찾아내고, 소프트웨어가 관련 명세서를 만족하는지를 확인해서 부적합한 경우 시정 또는 해결방안을 강구하도록 요구한다. 소프트웨어 보증공정들은 그림 4에서와 같이 서로 별개의 활동들이고 동시에 수행되며, 소프트웨어 개발공정에 직접적인 영향을 미치게 된다. 그림 4는 안전성-필수 소프트웨어를 대상으로 하여 개발공정과 보증공정의 관계를 나타낸 것이다.



<그림 4> 소프트웨어 개발공정 및 보증공정의 관계

그러면 소프트웨어 안전등급별로 보증공정 활동에서 품질보증요건 또는 기술기준들을 어떻게 차등적으로 적용할 것인가 하는 규제입장을 기술하면 다음과 같다.

첫째는 소프트웨어 개발조직과 품질보증, 또는 확인 및 검증조직의 독립성 보장이다. 안전성-필수 소프트웨어의 경우 소프트웨어 품질보증, 확인 및 검증 조직들은 개발조직과는 기술, 조직, 책임, 권한 및 재정적으로 독립되어야 한다. 안전성-관련 소프트웨어의 확인 및 검증 조직은 개발조직과는 기술, 조직, 또는 재정적 관점에서 적절히 독립되어야 한다. 비안전성 소프트웨어는 개발조직이 확인 및 검증 업무들을 동시에 수행할 수 있다.

둘째는 안전성 분석활동이다. 안전성-필수 소프트웨어의 경우 위험요소 또는 위험도 분석활동

을 반드시 수행하여 안전등급 IC-1 계통에 미치는 영향을 확인하여야 한다. 반면에 안전성-관련 또는 비안전성 소프트웨어에 대해서는 안전성 분석활동이 요구되지는 않으나, 이들 소프트웨어가 안전성-필수 소프트웨어와 함께 동일한 컴퓨터에서 돌아간다면 안전성-필수 소프트웨어에 미칠 수 있는 위험요소를 찾아서 보완하고, 필요하다면 적절한 격리를 하여야 한다.

셋째는 소프트웨어 생명주기에 따른 확인 및 검증 활동에 있어서 각 등급의 소프트웨어마다 서로 다른 확인 및 검증 업무와 기법을 적용한다. 본 논문에서는 대표적으로 소프트웨어 요구사항 단계의 확인 활동과 기법을 제시한다. 소프트웨어 요건사항에 대한 확인 요소는 표 1에 요약되어 있다.[6] 즉 소프트웨어 요구사항 단계의 확인 업무에 대해 각 소프트웨어 안전등급별로 적용되는 권고사항과 선택사항을 제시하였다. 소프트웨어 확인 및 검증 활동에서 적용될 수 있는 여러 가지 확인 및 검증 기법들을 특성별로 4 가지로 구분하고 표 2에 제시하였다. 표 2에 의하면 여러 가지 기법들이 각 공정마다 적용되고 있으나, 반드시 준수되어야 할 의무사항들은 아니고 각 공정마다 확인 및 검증 활동에서 공정 특성에 맞게 선택적으로 관련 기법들을 적용할 수 있다.

<표 1> 소프트웨어 요구사항에 대한 확인 활동 및 기법

입력 문서	출력 문서
• 소프트웨어 요구사항 명세서(SRS) • 인터페이스 요구사항 명세서(IRS) • 전단계 필수성분석보고서 • 전단계 위험요소 및 위험도 분석보고서	• 소프트웨어 요구사항 확인보고서 • 시스템 및 수락(acceptance) 시험계획서 • 필수성분석보고서 개정 • 위험요소 및 위험도 분석보고서 개정 • 부적합보고서(anomaly report) • 단계요약보고서
확인 기준문서: 시스템 개념문서 시스템 설계명세서(SDS) 소프트웨어 개발공정계획서	

확인 업무	확인 기법 (표 2)	소프트웨어 안전등급(*)		
		SC	SR	NS
• 소프트웨어 요구사항 추적성 분석	I.2.10	R	R	O
• 소프트웨어 요구사항 검토 및 분석(**)	I.1.1-I.1.6	R	R	R
	I.2.1/3/4/5/6/8/9/11/12	R	O	O
	I.2.13	O	O	O
• 소프트웨어 요구사항 인터페이스 (하드웨어, 사용자 등) 분석	I.2.6	R	R	R
• 시스템/수락 시험계획서 작성 및 검토(**)	I.1.2-I.1.5	R	R	R
• 형상관리 평가(**)	I.1.2-I.1.5	R	R	O
• 필수성 분석	I.4.3	R	R	O
• 위험요소 및 위험도 분석(**)	I.4.4-I.4.5	R	O	O

(*) SC: 안전성-필수 소프트웨어, SR: 안전성-관련 소프트웨어, NS: 비안전성 소프트웨어
R - 권고사항, O - 선택사항
(**) 적어도 한 가지 이상의 확인기법들을 활용하여야 한다.

<표 2> 소프트웨어 생명주기 및 확인·검증 기법의 적용

구 분	생명주기 / 확인/검증기법	시스템 개념	시스템 요건	SW 요건	SW 설계	SW 구현	단위 시험	통합 시험	검증 시험	설치 시험	운전 보수	비고
검 토	관리 검토	○	○	○	○	○	○	○	○	○	○	I.1.1
	기술 검토	○	○	○	○	○	○	○	○	○	○	I.1.2
	검 사			○	○	○	○	○	○	○	○	I.1.3
	워크스루			○	○	○	○	○	○	○	○	I.1.4
	감 사			○	○	○	○	○	○	○	○	I.1.5
	데스크 점검			○	○	○						I.1.6
설계분석	알고리즘 분석			○	○	○	○				○	I.2.1
	경계 값 분석					○	○	○	○		○	I.2.2
	제어 흐름 분석			○	○	○						I.2.3
	데이터베이스 분석			○	○	○					○	I.2.4
	데이터 흐름 분석			○	○	○					○	I.2.5
	인터페이스 분석		○	○	○	○					○	I.2.6
	크기 및 타이밍 분석				○	○	○	○	○	○		I.2.7
	회귀 분석 및 시험			○	○	○	○	○	○	○	○	I.2.8
	시뮬레이션 분석			○	○	○	○			○		I.2.9
	추적성 분석	○	○	○	○	○	○	○	○	○	○	I.2.10
	기호 실행			○	○	○						I.2.11
	시제품 실행			○	○	○	○	○	○			I.2.12
	정형 방법			○	○	○						I.2.13
시험전략	구조 시험						○					I.3.1
	기능 시험						○	○	○	○	○	I.3.2
	성능 시험						○	○	○	○	○	I.3.3
	인터페이스 시험						○	○	○	○	○	I.3.4
	부하 시험						○	○	○		○	I.3.5
	통계 시험								○			I.3.6
	시험 인증						○	○	○	○	○	I.3.7
안전성 분석	소프트웨어 고장모드 및 영향분석			○	○	○					○	I.4.1
	소프트웨어 고장수목분석			○	○	○					○	I.4.2
	필수성분석			○	○	○	○	○			○	I.4.3
	소프트웨어 위험요소분석			○	○	○	○	○	○	○	○	I.4.4
	위험도분석			○	○	○	○	○	○	○	○	I.4.5

3. 결 론

본 논문에서는 원전의 계측제어계통 및 소프트웨어에 대한 안전등급 분류내용과 소프트웨어에 관한 규제입장을 제시하고 소프트웨어 품질보증 확보를 위한 중요한 활동인 확인 및 검증 활동들을 소프트웨어 등급별로 차등 적용하는 방안을 제안하였다. 즉 원전 안전성 계측제어계통을 안전등급 분류기준에 따라 안전등급 IC-1, IC-2, IC-3 및 Non-IC로 분류하였으며, 안전성 계측제어계통에 사용된 소프트웨어가 수행해야 할 안전기능의 중요도에 따라 안전성-필수 소프트웨어, 안전성-관련 소프트웨어, 그리고 비안전성 소프트웨어로 분류하였다. 이 같은 안전등급 분류기준에 근거하여 각 등급별 소프트웨어 확인 및 검증 활동의 정도를 서로 다르게 차등화 하였다. 또한 계측제어계통에 사용된 소프트웨어를 설계 및 구현하는 관점에서 신개발 소프트웨어와 기성 소프트웨어로 분류하였고, 각 유형별의 소프트웨어에 대한 규제입장을 제시하였다. 이것은 원전 안전성 디지

털-기반 계측제어계통의 건설 및 운영허가를 위한 심사에 대비하고, 소프트웨어 안전등급별 차등화된 확인 및 검증 활동에 대한 규제입장을 제시함으로써 규제자 및 피규제자가 서로 안정되고 일관된 인허가 업무를 수행할 수 있을 것으로 기대된다.

참고문헌

[1] SECY-91-292, “Digital Computer Systems for Advanced Light Water Reactors,” U.S. NRC, September 16, 1991.

[2] SECY-93-087, “Policy, Technical, and Licensing Issues Pertaining to Evolutionary and Advanced Light-Water Reactors Designs,” U.S. NRC, April 2, 1993.

[3] KINS/GR-196, “Development of Safety and Regulatory Requirements and Guides for Korean Next Generation Reactor,” Annual Report(Phase III-1), KINS, February, 2000.

[4] KINS/GR-192, “차세대원자로 상세안전요건 개발(III-1),” 3단계 1차년도, KINS, 2000. 2.

[5] Appendix 7-A, BTP ICSB 14, “Guidance on Software Reviews for Digital Computer-Based Instrumentation and Control Systems”, NUREG-0800, Rev. 4, U.S. NRC, June, 1997.

[6] IEEE Std. 1012-1998, “IEEE Standard for Software Verification and Validation.”

[7] IEEE Std. 1074-1995, “Standard for Developing Software Life Cycle Processes.”

[8] NUREG-0800, “Standard Review Plan,” Rev. 4, U.S. NRC, June, 1997.

[9] IEEE Std. 7-4.3.2-1993, “IEEE Standard for Digital Computers in Safety Systems of Nuclear Power Generating Station Class 1E System.”

[10] Safety Standards No. NS-G-1.1, “Software for Computer Based Systems Important to Safety in Nuclear Power Plants,” International Atomic Energy Agency, Vienna, 1999.