

# 임베디드 소프트웨어의 평가 프로세스와 시험체계의 구축

양 해 술<sup>†</sup> · 신 석 규<sup>††</sup> · 정 혜 정<sup>†††</sup>

## 요 약

임베디드 소프트웨어의 품질시험을 통해 임베디드 시스템 구매자들의 요구에 부합되는 고품질의 시스템을 구현할 수 있다. 임베디드 소프트웨어의 경우, 아직까지 국내에서 활용할 수 있는 구체적인 품질시험 체계가 구축되어 있지 않은 실정이다. 최근 다양한 임베디드 시스템들이 개발되고 있으나 결함으로 인한 리콜 등 품질면에서 많은 문제점들이 노출되고 있다. 본 연구에서는 임베디드 소프트웨어 평가를 위해 ISO/IEC 12119를 근간으로 하여 품질시험을 수행할 수 있는 평가모듈과 품질검사표를 개발하여 임베디드 소프트웨어 평가에 적용할 수 있도록 하였다.

## Construction of Evaluation Process and Test System for Embedded Software

Hae-Sool Yang<sup>†</sup> · Seok-Kyoo Shin<sup>††</sup> · Hye-Jung Jung<sup>†††</sup>

### ABSTRACT

It is possible to implement high quality system which coincided with embedded system purchaser's requirements by quality test. The concrete quality test system applicable within our country for embedded software was not constructed yet. Recently, various embedded systems are developed, but they have many problems such as recall by defect. In this paper, we developed evaluation modules and quality test tables which can implement quality test based on ISO/IEC 12119 for embedded software evaluation.

키워드 : 임베디드 소프트웨어(Embedded Software), 품질 평가(Quality Evaluation), 평가모듈(Evaluation Module), 메트릭(Metric)

### 1. 서 론

컴퓨터 기술의 급격한 발전으로 오늘날 컴퓨터를 활용하지 않는 업무 분야가 거의 없을 정도로 많은 분야에서 컴퓨터의 활용도는 점점 높아지고 있을 뿐만 아니라 일반 개인 사용자들에게도 컴퓨터가 급속히 보급되고 있는 추세이다. 이러한 추세에 맞추어 컴퓨터를 어떤 용도로 사용하느냐에 따라 용도에 맞는 적절한 소프트웨어의 개발이 요구되고 있으며 이를 뒷받침할 수 있도록 소프트웨어 개발 업체에서는 다양한 종류의 소프트웨어들을 개발하고 있다.

이제 사용자는 다양한 유형의 소프트웨어들 중에서 자신이 컴퓨터를 사용하는 목적과 용도에 알맞은 소프트웨어를 선택할 수 있게 되었으며 이로 인해 올바른 선택 방법에 대한 중요성이 대두되고 있다.

더불어 소프트웨어 제품의 품질이 중요한 관건으로 대두된지 오래이며 소프트웨어 제품 품질에 대한 시험·평가의

필요성이 논의되고 있다. 특히, 최근 급격한 발전을 보이고 있는 정보통신 및 산업 분야에서 활용되고 있는 임베디드 소프트웨어에 대한 시험·평가의 필요성이 증대되고 있다 [7]. 임베디드 시스템은 범용의 프로세서가 처리하기 힘든 특수한 목적의 시스템으로서 하드웨어 부분과 소프트웨어의 특성을 고려하여 주어진 제약조건을 만족하면서 최소의 비용으로 시스템을 구현해야 한다는 특성을 가지고 있다 [8]. 전 세계에 존재하는 컴퓨터 시스템의 90% 이상이 임베디드 시스템이라는 보고가 있을 정도로 임베디드 소프트웨어의 활용분야는 매우 광범위한 뿐만 아니라 부가가치가 높은 분야이기도 하다[13].

소프트웨어의 품질향상을 위한 핵심적인 사항으로 소프트웨어를 올바르게 시험하는 체계의 구축이 급선무이며 패키지 소프트웨어 제품의 시험에 관한 국제표준인 ISO/IEC 12119를 적용하여 임베디드 S/W의 시험에 적합하도록 수정·보완할 수 있다.

임베디드 소프트웨어의 품질시험을 통해 임베디드 시스템 구매자들의 요구에 부합되는 고품질의 시스템을 구현할 수 있도록 지원할 수 있다. 지금까지 일반적인 SI 소프트웨어에 대한 품질평가 방법론이나 평가 기법, 평가 도구 등이

\* 본 연구는 대학 IT연구센터 육성·지원사업의 연구결과로 수행되었음.

† 중신회원 : 호서대학교 벤처전문대학원 교수

†† 정 회 원 : 정보통신기술협회 소프트웨어시험센터 센터장

††† 중신회원 : 평택대학교 정보통계학과 교수

논문접수 : 2003년 11월 12일, 심사완료 : 2004년 4월 14일

개발된 사례[13]가 있으나 임베디드 소프트웨어의 경우에는 아직까지 국내에서 활용할 수 있는 구체적인 체계가 구축되어 있지 않은 실정이다. 최근 다양한 임베디드 시스템들이 출시되어 소비자의 선택을 기다리고 있으나 소프트웨어의 결함으로 인해 불량 리콜되는 사례가 있으며, 1986년에서 1997년 사이 미국식품의약국(US. Food and Drug Administration)의 발표에 의하면 의료기기용 소프트웨어의 결함이 450건으로 집계되어 있으며 그 중 24건은 사망이나 심각한 상해를 초래한 것으로 보고하고 있으며 이것은 병산의 일각이라고 경고하는 등, 품질면에서 많은 문제점들이 노출되고 있다.

본 논문에서는 이러한 문제들을 해결하기 위해 본 연구에서는 이러한 문제들을 해결하기 위해 제2장에서는 관련 연구의 현황을 소개하였고 제3장에서는 임베디드 소프트웨어의 품질특성에 관한 모델과 품질평가 프로세스를 구축하였으며 제4장에서는 임베디드 소프트웨어의 품질시험을 위한 척도와 방법을 모듈화 한 시험모듈에 대해 기술하였고 제5장에서는 평가사례를 통해 평가방법을 예시하였으며 제6장에서 결론을 제시하였다.

## 2. 관련 연구의 현황

### 2.1 국외의 현황

국외에서는 소프트웨어의 프로세스와 제품에 관한 평가 기술이 활발히 연구되고 있으며 이미 실용화 단계에 접어들고 있다. 국외의 품질관련 연구 현황을 다음과 같이 요약하였다.

#### 2.1.1 미국의 현황

미국은 The Tolly Group(네트워크 장비의 성능시험 담당) 등 분야별로 민간 자율 형태로 운영되고 있으며 OSI 프로토콜 및 TCP/IP를 사용하는 모든 제품을 대상으로 산업체, 정부에 조달되는 제품들의 표준 규격인 산업/정보개방시스템 규격(IGOSS) 시험 프로그램을 운영하고 있다. 민간 IT 시험인증 전문기업 베리테스트(VeriTest)는 마이크로소프트와 제휴, 윈도2000 로고(Logo) 인증서비스를 실시. 또한 AT&T와 아이리스, BMC소프트웨어, IBM, 노벨(Novell), 팜(Palm) 등과 같은 주요 IT기업에 대한 인증 프로그램을 운영. VeriTest는 유명 SW시험업체인 데이터디멘션(Data Dimension Inc.)을 인수, 합병했으며 2001년 12월에는 한국정보통신기술협회(TTA) 산하 소프트웨어시험센터와 상호 인증계약을 체결하는 등 소프트웨어 품질인증분야의 위상을 강화하고 있다.

#### 2.1.2 일본의 현황

일본은 1992년 4월에 INTAP/ICTC를 OSI 제품에 대한 시험 기관으로 지정하여 적합성 시험과 상호운용성 시험 등을 수행하고 있다.

#### 2.1.3 독일의 현황

독일은 소프트웨어조합(GGS : Gutgemeinschaft Software)에서 패키지 소프트웨어의 제품 기술(description)이 제품과 일치하는지를 시험 인증하고 있으며 패키지 소프트웨어를 중심으로 매년 20개 안팎의 SW를 인증. GGS는 각 SW제품의 설명서를 기준으로 삼아 기능과 성능을 확인하는 형태로 인증을 실시하며 국가공인시험기관(TUV) 자격을 획득한 대형 SW업체들로 구성되어 있다.

#### 2.1.4 영국의 현황

영국표준원(BSI : British Standards Institution)에서 특정한 제품 품질요구사항을 만족하고 품질 시스템이 적합하면 PAS 마크를 부여하는 제품승인제도(PAS : Product Approval Scheme)를 구축하고 있다.

#### 2.1.5 덴마크의 현황

민간기업인 델타(DELTA)가 지난 1982년부터 소프트웨어 시험인증서비스를 실시하여 안정성을 확보해야 하는 소프트웨어 공정(프로세스) 통제와 실시간으로 소프트웨어 기능성을 평가하는데 주력하고 있으며 이에 맞춰 주로 사용자 매뉴얼, 설계문서 및 소스코드의 유지보수성을 평가하고 있다.

### 2.2 국내의 현황

국내에서는 연구소를 중심으로 소프트웨어에 대한 평가 기술 연구, 기술성 평가 등이 이루어지고 있으며, 공공기관에서 추진하는 정부차원의 품질인증 서비스가 초기단계로서 추진되고 있거나 시험 인증 서비스를 거친 후 본격적인 인증 서비스를 추진하고 있다.

ETRI 컴퓨터/소프트웨어기술연구소에서 1997년부터 1999년에 걸쳐 소프트웨어 전반에 대한 평가 기술을 연구한 바 있으며 ETRI 표준연구센터(ETRI PEC)에서 정보통신기기와 프로토콜에 대한 시험 서비스를 일부 제공하였다.

한국정보통신기술협회(TTA : Telecommunications Technology Association)에서는 네트워크 장비, 소프트웨어, 이동통신 단말기, 디지털 방송장비, 통신서비스 품질에 대한 시험인증 서비스를 실시하고 있고, 산자부 기술표준원에서는 자동화 설비 시험 평가 센터를 설립하여 운영하고 있으며 산업용 소프트웨어에 대한 인증체계 구축을 추진하고 있다.

국내에서는 소프트웨어 제품 인증에 대한 관련 기반이 아직 미흡한 실정이며 국가적인 시험인증체계가 초기 시험 단계를 거쳐 점진적으로 개선이 진행되고 있는 단계에 있다. 특히, 정보통신 분야의 시험인증 관련 법령 및 규정이 미흡하고 최근에 시험인증체계에 대해 기반을 구축해 나가고 있는 실정이다. 민간차원의 소프트웨어 시험인증 기관은 아직 없으며 일부 업체에서 외국 시험기관의 시험인증을 획득한 사례가 있으며 국내 시험인증 기관의 시험인증을 받으려는 시도가 점차 증가하고 있다.

### 2.3 임베디드 소프트웨어의 특성

임베디드 시스템에 대한 품질평가를 수행하기 위해서는 먼저 임베디드 시스템의 구성요소와 특징을 파악하고 국제 표준의 품질특성과 관련성을 파악하여 품질평가를 위한 제반사항을 정의하는 평가모듈을 구축하고 품질평가를 위한 수행 절차를 구축하여 적용할 필요가 있다. 이 절에서는 임베디드 시스템의 구성요소와 특징을 분석하여 품질특성 모델을 구축하였다.

#### 2.3.1 임베디드 소프트웨어의 구성요소

임베디드 소프트웨어의 구성요소에는 임베디드 소프트웨어가 만족시켜야 할 권고나 요구사항 또는 규제를 포함하고 있는 요구사항 문서와 소프트웨어 속성을 설명하는 문서로서 잠재적인 구매자가 제품 구입에 앞서 스스로 그 제품의 적합성을 평가할 수 있도록 하는 제품설명서, 제품의 사용을 위해 제공되며 제품 사용에 필요한 모든 정보가 포함되는 사용자 문서, 한가지 이상의 매체를 통해 제공되는 실행 프로그램과 관련 데이터가 있다.

#### 2.3.2 임베디드 시스템의 특징

임베디드 시스템은 너무나 다양해서 특징을 일일이 언급하기는 어렵지만, 앞에서 열거한 임베디드 시스템의 경향에 의거하여 몇 가지 특징을 요약하면 다음과 같다[13].

##### (1) 목적이 한정되어 있다

범용으로 만들어진 프로세서와 메모리는 어떤 프로그램이 수행될지 정확히 알 수 없기 때문에 일반적인 프로그램이 빠르게 실행될 수 있는 구조를 갖게 되며, 이렇게 설계된 후 고정된다. 이렇게 일단 설계된 프로세서의 성능은 결국, 운영체제 및 컴파일러 등 많은 구성 소프트웨어에 의해 평가된다. 이는 먼저 하드웨어가 개발되고 나서 그 구조에 가장 적합한 소프트웨어 설계라는 순차적인 설계 방법론을 의미한다.

컴퓨터 시스템이 어떤 특정 목적을 위해 설계될 때, 수행되는 기능이 거의 고정적이기 때문에, 범용 시스템의 설계시 만큼, 소프트웨어만을 최적화시켜 얻을 수 있는 성능 향상이 크지 않은 경우가 많다. 그래서, 임베디드 시스템은 빈번한 연산의 보다 빠른 수행을 위하여 때론 그 기능을 하드웨어로 구현해야될 필요가 있다. 즉, 소프트웨어와 하드웨어가 동시에 개발되면서, 어떤 기능이 하드웨어로 구현되어야 하는지의 문제를 먼저 고려해야 한다. 바꾸어 말하면, 이는 특정한 목적을 효과적으로 달성하기 위한 최소의 하드웨어와 최소의 소프트웨어를 이용한 시스템을 설계해야 한다는 문제를 임베디드 시스템 설계자가 해결해야 함을 의미한다. 현실적으로는 하나의 범용 프로세서가 개발되었을 때, 이 프로세서가 임베디드 시스템에 쓰이기 위해, 이 범용 프로세서를 기본으로 하는 여러 종류의 변종 프로세서가 상황에 맞추어 필요한 기능을 갖도록 특화되는 경우가 많다. 그 만큼 하나의 임베디드 시스템은 각각의 개성

을 갖게 되며, 다양하다는 말과는 일맥 상통한다고 보아야 한다. 혹자는 이를 "Each embedded system is unique."이란 말로 표현하는데 임베디드 시스템의 다양성을 잘 말해준다고 하겠다.

##### (2) 실시간 처리가 많다

범용 시스템의 처리는 일반적으로 주어진 자원을 최대한 효과적으로 활용하여, 되도록 빠르게 수행하는 것을 목적으로 한다. 이는 절대적인 처리 기한이 없이 주어진 시간 안에 되도록 많은 일을 수행하도록 설계되었음을 의미한다. 이에 반해 임베디드 시스템의 경우는 과제의 처리 기한이 주어지는 실시간 처리가 많다. 예를 들어, 목소리와 동영상 이 같이 멀티미디어로 실행되어야 하는 경우, 목소리와 동영상은 실행 기한(deadline)이 주어져 있어 그 실행 기한 안에 데이터가 처리되지 않으면, 동영상이나 인터넷 전화에서 정보 전달이 제대로 되지 않는다. 다중의 매체가 동시에 처리되는 멀티미디어 처리에서의 어려운 점은 각 매체의 종류마다, 실행 기한이 각 과제마다 다를 수 있으며, 데이터 처리의 정밀도 및 질(QoS : Quality of Service) 등이 모두 다르기 때문에 효과적인 실시간 처리는 매우 어려운 문제이다. 이런 실시간 처리는 실행 기한 내에서만 처리된다면, 무조건 빨리 처리되는 것이 바람직하다고 할 수 없기 때문에, 처리해야 하는 모든 작업들의 우선 순위를 효과적으로 조정하여 모든 작업이 각각의 제한 시간 안에 처리되도록 하는 것이 중요하다. 이는 임베디드 시스템의 주요 특징이다.

또한, 때로는 처리되어야 하는 속도가 외부의 환경적인 이벤트(event)에 적절히 반응해야 하는 리액티브(reactive)적인 성격이 강해서, 지나치게 비판적이지 않으면서도 주기적이거나 비주기적인 이벤트에 적절히 대응하기 위한 조건을 만족해야 한다. 실시간 처리는 실행 제한 시간을 절대로 어겨서는 안되는 경성 실시간(hard realtime) 시스템과, 어느 정도 어겨도 되는 연성 실시간(soft realtime) 시스템으로 크게 나뉜다.

##### (3) 대량 생산을 목적으로 한다

우리의 생활 속에서 쉽게 발견되는 몇 가지 임베디드 시스템의 예를 보아도, 임베디드 시스템의 얼마나 우리의 생활과 밀접하게 연관되어 있는지 알 수 있다. 예를 들어, 세탁기, 냉장고, 셋-탑 박스, 게임기 등의 기기들을 보아도, 대량 생산을 목적으로 최소 비용에 최대 효과를 내야하는 기기 등임을 쉽게 알 수 있다. 즉, 임베디드 시스템의 성능 평가는 최종 제품의 성능에 의해 평가되지, 그 안에 쓰이는 하드웨어와 소프트웨어의 성능에 의해 평가되지는 않는다. 그러므로 비용의 최소화와 소비자의 다양한 욕구를 최대한 충족시켜야 하는 것이 중요하다. 예를 들어, 세탁기를 설계함에 있어 인텔 텐티엄 프로세서를 쓴다면, 그 회사는 아무리 고성능 세탁기를 만든다 하더라도, 경쟁에서 살아남기

힘들 것이다. 그런 의미에서 임베디드 시스템의 설계 철학은 상당부분 마케팅에 의존하며, 범용 컴퓨터 시스템의 설계와는 많이 달라야 하고 또 다르다.

(4) 거친 환경에 강한 내구성을 가져야 한다

많은 임베디드 시스템은 고온이나 다습한 환경, 또는 충격이 가해지거나, 일부 기능에 이상이 있어도 기본 기능은 계속 동작하도록 요구되는 경우가 많다. 예를 들어, 자동차에 쓰이는 ABS의 경우, 50도 이상의 고온과 -30도 이하의 저온에서도 고장이 없는 내구성을 보여야 한다. 이는 많은 임베디드 시스템이 험난한 환경에서 동작함과 동시에 절대적으로 오작을 허락하지 않는 미사일 제어 시스템이라든가 ABS 등의 아주 중요한 작업을 수행하는 경우가 빈번하기 때문에 더욱 더 중요하다고 할 수 있다.

2.3.3 임베디드 소프트웨어의 적용 분야

임베디드 소프트웨어의 적용 분야는 제어, 가전, 단말, 통신장비 등 크게 네 가지로 나눌 수 있다.

- ① 제어 : 제어 분야로는 공정제어, 자동차 제어, 로봇 제어 등이 있으며 실시간 요구가 강한 특징이 있다.
- ② 가전 : 가전 분야로는 세탁기, 전자레인지 등의 단순제어와 인터넷에 접속하는 TV 등의 GUI(Graphical User Interface)가 중요시되는 분야가 있다.
- ③ 단말 : 단말 분야로는 핸드폰, PDA 등의 모바일(mobile) 기기 등이 있으며 소비전력의 최소화가 필요하고 작은 크기의 디스플레이를 요구하는 특징이 있다.
- ④ 통신장비 : 통신장비 분야로는 직렬, 병렬, 이더넷, USB 등의 다양한 통신포트를 요구하는 부문이 있다.

각 응용들은 실시간성의 강약, 입출력 장치의 다양성, GUI의 중요도, CPU 처리요구의 강약, 저소비전력 요구의 강약 등에서 차이를 보이며 이런 점들이 마이크로 프로세서의 선택이나 하드웨어의 구성 등에 영향을 미친다. <표 1>에서는 각 응용분야와 그들의 특징들을 비교 분석하고 있다.

<표 1> 응용과 특성 비교

구 분	실시간성	입출력	GUI	CPU 처리	소비전력 요구
제 어	크다	다 양	불필요	보통	보통
가 전	작다	단 순	대형LCD	높다	작다
단 말	작다	단 순	중소형LCD	보통	크다
통신장비	보통	다 양	불필요	보통	보통

3. 임베디드 소프트웨어의 품질평가 체계 구축

임베디드 소프트웨어를 평가하기 위해 임베디드 소프트웨어의 특성에 따른 품질모델과 평가 프로세스를 구축하였다.

3.1 품질특성 모델의 구축

임베디드 소프트웨어의 유형은 응용 분야에 따라 제어, 가전, 단말, 통신장비 등, 네 가지로 분류할 수 있다.

제어응용은 공장이나 자동차의 제어, 로봇의 제어 등의 응용을 말하며 실시간 요구가 매우 강한 특성을 가지며 오류 발생으로 인한 큰 피해가 발생할 수 있고 입출력 방식이 다양하며 GUI(Graphical User Interface)는 불필요하다. 따라서, 이러한 특성을 고려하여 품질특성과 관련시켜 보면 실시간 요구가 강하다는 점에서 효율성의 측면이 강조되며, 오류 발생으로 인한 피해가 크고 공장이나 자동차, 로봇을 제어하는 다소 거친 환경이라는 면에서 신뢰성 또한 중요한 면이 될 수 있다. GUI가 불필요하므로 사용성 측면은 약하다고 볼 수 있다.

가전응용은 과거의 세탁기나 전자레인지와 같이 단순 제어만을 제공하던 것에서 벗어나 인터넷에 접속하는 냉장고, TV 등과 같이 GUI가 강화되는 방향으로, 또한 인터넷에 접속되는 방향으로 발전하고 있다. 가전응용 분야는 실시간성이 약하므로 효율성 측면의 중요도가 떨어질 수 있으나 멀티미디어 정보를 많이 다루기 때문에 고성능의 시스템 성능을 요구하므로 효율성 측면이 매우 중요한 요소로 부각되어야 한다. 또한, 대형 LCD를 필요로 하는 일반 사용자를 위주로 하므로 편리한 사용환경이 요구되어 사용성 측면이 강조되어야 한다.

단말응용은 핸드폰이나 PDA, 혹은 그에 준하는 모바일 기기들을 말하는데 소비 전력의 최소화, 작은 크기의 디스플레이 등을 요구한다. 단말응용의 경우는 중 소형 LCD를 요구하며 상당히 진보된 사용자 인터페이스를 지원하는 등 사용성 측면이 중시된다. 실시간성이 떨어지고 특별히 고성능의 CPU를 필요로 하지 않으므로 시간적 효율성 측면은 다소 중요도가 떨어진다고 볼 수 있다. 그러나 소비전력에 대한 요구가 크므로(대체로 작은 크기로 인해 전력 소모는 적으나 휴대용 기기라는 측면과 작은 배터리 용량 등을 고려할 때 전력 요구가 크다고 할 수 있다) 자원 효율성의 측면은 강조된다. 단말응용의 경우는 특히 제품의 생명주기가 짧으며 빠른 신제품 개발과 출시가 중요하므로 유지보수성의 측면이 매우 중요하다.

통신장비 응용은 직렬, 병렬, 이더넷, USB 등의 다양한 통신포트를 요구한다는 점이 특징이다. 통신장비응용은 실시간성이 다소 강조되므로 시간 효율성의 측면이 중요하고 GUI가 불필요하므로 사용성의 측면은 중요도가 떨어진다.

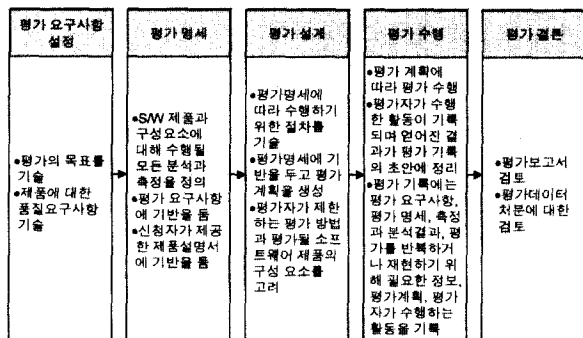
상기한 임베디드 소프트웨어의 다양한 특성들을 고려하여 소프트웨어 품질특성에 관한 국제표준인 ISO/IEC 9126 과 패키지 소프트웨어 품질시험을 위한 국제표준인 ISO/IEC 12119의 품질특성을 검토하여 <표 2>와 같은 임베디드 소프트웨어 품질특성을 설정하였다.

〈표 2〉 임베디드 소프트웨어의 품질특성

품질특성	내 용
상호 운영성	• 주변 하드웨어와의 높은 결합성 요구
신뢰성	• 오류 발생의 최소화(오작동 및 작동중지가 발생되지 않아야 함) • 거친 환경에서 강한 내구성을 가지는 수준 • 오류 발생 수준(신뢰성 수준)에 대해 명시하고 명시된 규정을 준수
사용성	• 사용자가 소프트웨어의 기능을 쉽게 재구성할 수 있도록 하여 사용 편리성 제공 • 멀티미디어 서비스 제공을 통한 편리한 사용 환경 제공
실시간성	• 하드웨어로 구현한 부분에 대한 적절성 여부(고성능을 요구하는 기능이 하드웨어로 구성) • 신속한 응답, 정해진 제한 시간 내에 처리 • 하드웨어와 결합된 최종 제품의 성능 우수성
융통성	• 소프트웨어의 변경이 손쉽고 간단하며 기능 확장 용이 (업그레이드 용이) • 기존의 소프트웨어에 대한 높은 재사용 수준 • 향후, 업그레이드가 필요한 부분이 소프트웨어로 구성됨 • 하드웨어 플랫폼이 쉽게 변경될 수 있으므로 다양한 플랫폼을 지원할 수 있어야 함 • 기존의 임베디드 소프트웨어에 대한 호환성 • IP Block과의 연결 용이

3.2 품질평가 프로세스의 구축

이 절에서는 소프트웨어 품질평가 프로세스에 관한 국제 표준인 ISO/IEC 14598의 부분5인 평가자를 위한 제품 평가 절차를 근간으로 하여 제3자 기관의 관점에서 평가할 수 있는 소프트웨어 평가 프로세스에 대해 구축하고 세부 활동을 정의하였다. 소프트웨어의 품질평가를 위한 프로세스는 (그림 1)과 같다.



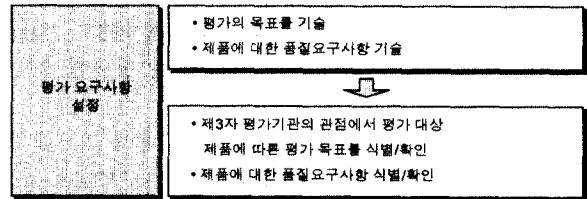
(그림 1) 임베디드 소프트웨어의 품질평가 프로세스

3.2.1 품질평가 활동의 수정

ISO/IEC 14598-5의 품질평가 프로세스는 품질 평가자를 위한 프로세스이지만 제3자의 입장에서 다수의 소프트웨어에 대해 동일한 기준을 적용한 평가로 보기는 어렵다. 즉, 개발자의 목표를 점검하기 위해 요구사항을 정의하고 요구사항을 검증할 수 있는 측정 방법을 명세하여 평가를 수행한다는 관점이다.

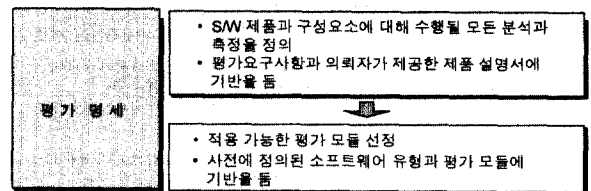
그러나 제3자 평가기관의 관점에서는 사전에 설정된 기준을 평가 대상 소프트웨어에 대해 동일하게 적용해야 하

므로 (그림 2)와 같이 “평가 요구사항 설정” 과정은 평가 대상 제품의 유형에 따라 사전에 정의된 평가 목표를 평가 의뢰자에게 식별 및 확인시키고 평가대상 제품에 대한 품질 요구사항의 식별 및 확인 과정이 이루어져야 한다.



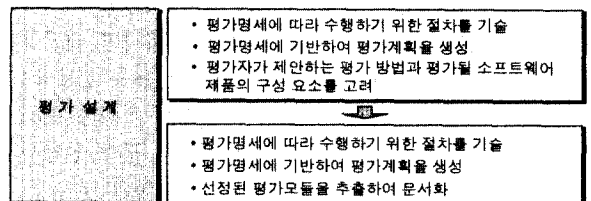
(그림 2) 평가 요구사항 설정 단계의 수정

“평가명세” 과정도 마찬가지로 (그림 3)과 같이 소프트웨어 품질평가를 위해 사전에 정의된 평가모듈 중에서 평가 의뢰된 임베디드 소프트웨어에 적용 가능한 평가모듈을 선정하는 과정으로 수정하였다.



(그림 3) 평가명세 단계의 수정

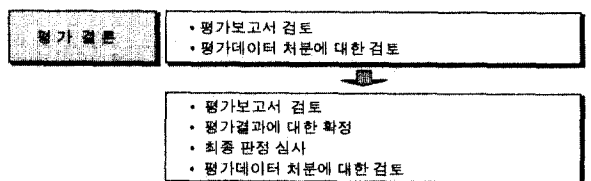
“평가설계” 과정에서는 평가 절차를 기술하고 평가명세에 따른 평가계획을 생성한다. 평가방법은 평가모듈에 정의되어 있으므로 (그림 4)와 같이 선정된 평가모듈을 추출하여 문서화하는 과정으로 수정하였다.



(그림 4) 평가설계 단계의 수정

“평가수행” 과정은 제3자 품질평가의 관점에서 보더라도 동일한 활동을 수행하는 것으로 적합하다고 판단된다.

“평가결론” 과정은 (그림 5)와 같이 제3자 품질평가의 관점에서 평가결과에 대해서 확정하고 평가위원회를 통한 최종 검토과정을 포함하였다.



(그림 5) 평가결론 단계의 수정

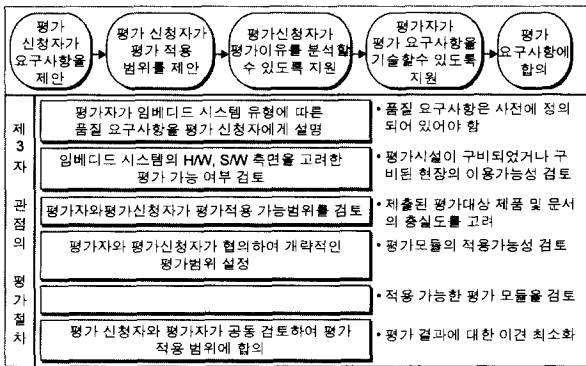
3.2.2 품질평가 단계별 상세 활동

(1) 평가 요구사항 설정

평가의 목표를 기술하고 제품에 대한 품질 요구사항을 기술하는 단계로 (그림 6)의 하단부와 같은 활동을 수행한다.

(그림 6)은 상단 부분과 같은 ISO/IEC 14598의 평가 요구사항 설정 과정에서 수행하는 활동에 대해 제3자 기관의 관점에서 임베디드 소프트웨어를 시험할 때 활용할 수 있도록 하단 부분과 같이 수정한 것이다.

먼저, 소프트웨어를 시험하는 제3자 기관에서 평가를 수행하는 경우, 시험기관에서 규정하고 있는 요구사항에 따라 시험 과정이 진행될 것이므로 평가 신청자가 요구사항을 제안하는 경우는 생각할 수 없다.



(그림 6) 평가 요구사항 설정

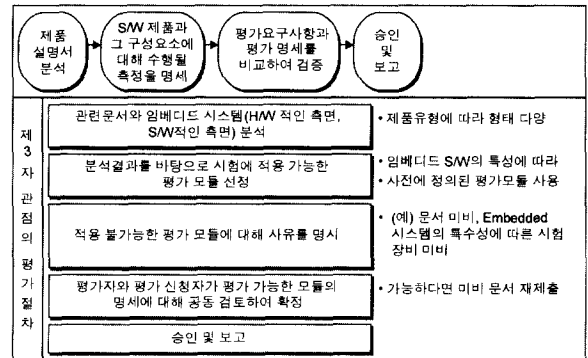
따라서, 평가자가 소프트웨어의 유형에 따른 품질 요구사항을 평가 신청자에게 설명하는 과정으로 대체하였다. 두 번째로, 임베디드 시스템은 소프트웨어와 하드웨어가 밀접하게 결합된 시스템인 만큼 통합된 형태의 시험이 가능하도록 평가기관의 시설이 구비되어 있는지 아니면 개발기관의 시설을 현장방문하여 활용할 수 있는지 등을 검토할 필요가 있다. 세 번째로, 평가 신청자가 평가 적용범위를 제안하는 과정도 적절하지 않으므로 평가자와 평가 신청자가 토의를 거쳐 평가 신청자가 제출한 대상 제품의 구성 요소를 검토하고 평가모듈을 이용하여 평가가 가능한 범위를 협의하는 과정으로 변경하였다. 네 번째로, 평가자가 평가 요구사항 정의 과정에 참여하는 부분을 제거하고 평가자와 평가 신청자가 협의하여 개략적인 평가범위를 설정하는 과정을 추가하였다. 마지막으로, 평가자가 평가 적용범위를 설정하고 평가 신청자와 공동으로 검토하여 평가 적용범위에 대해 합의하는 과정이 필요하다.

(2) 평가 명세

평가 대상 제품에 대해 수행될 평가와 측정의 범위를 정의하는 단계로 (그림 7)과 같은 활동을 수행한다. ISO/IEC 14598-5의 평가 명세 과정은 평가 신청자 중심이라는 특징이 있다. 즉, 평가 신청자는 평가를 통해 자신들이 개발한 소프트웨어의 문제점을 파악하고 평가 결과를 바탕으로 수

정·보완함으로써 품질향상을 꾀하고자 하는 것이 목적이다. 따라서 측정을 명세하고 그것이 평가 요구사항의 목적에 부응하는가를 비교 검증하는 과정을 거치게 된다.

그러나 평가 명세 과정을 제3자 시험·평가의 한 과정으로 파악하는 관점에서는 먼저, 제품설명서, 사용자 문서 등의 관련 문서와 하드웨어적인 측면과 소프트웨어적인 측면을 포함하는 임베디드 시스템에 대한 분석이 이루어지고, 사전에 정의된 평가모듈과 분석 결과를 비교하여 적용 가능한 평가모듈을 선정하게 되며 적용 불가능한 평가모듈에 대해서는 그 사유를 구체적으로 명시하는 과정이 필요하다.

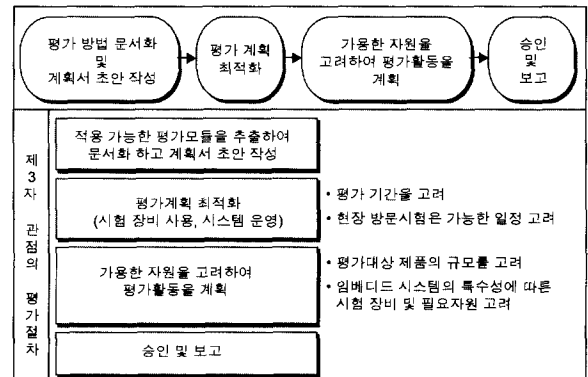


(그림 7) 평가 명세

그 다음 평가자와 평가 신청자가 공동으로 검토하여 평가 가능한 모듈의 명세에 대해 확정해야 하며 이 과정에서 만일 가능하다면 평가 불가능한 모듈과 관련된 평가 가능한 추가제품 및 문서를 평가 신청자가 첨부할 수 있도록 할 수 있다. 공동 검토 후 확정되면 승인 및 보고 과정을 거쳐 평가 명세 과정이 완료된다.

(3) 평가 설계

측정을 수행하는데 사용하는 절차를 문서화하는 단계로 (그림 8)에 평가 설계 과정에서 수행하는 활동을 나타내었다.



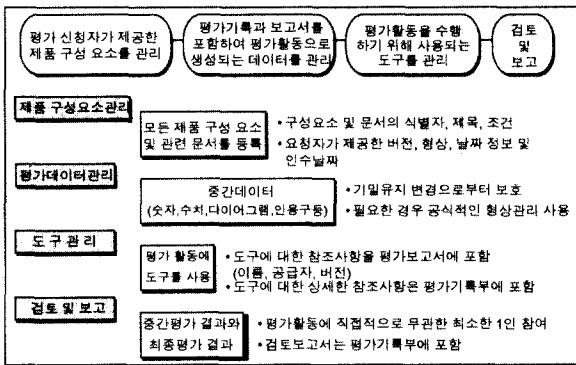
(그림 8) 평가 설계

제3자 시험·평가의 관점에서는 이미 평가모듈의 형식으로 평가 방법이 문서화되어 있으므로 “평가 방법 문서화

및 계획서 초안 작성” 과정은 이미 문서화된 평가모듈에서 “적용 가능한 평가모듈을 추출하여 별도로 문서화하고 계획서에 대한 초안을 작성”하는 과정으로 수정하였다. “평가 계획 최적화” 과정에서는 임베디드 시스템의 평가라는 점을 고려하여 임베디드 시스템을 시험하기 위한 장비의 사용 가능 기간과 책정된 평가 기간을 고려하고 개발업체에 대한 현장 방문의 경우에는 방문 가능 일정 등도 고려하여 계획을 수립한다. 기타, “가용한 자원을 고려하여 평가활동을 계획”하는 과정 및 “승인 및 보고” 과정은 동일한 절차에 따르면 된다.

(4) 평가 수행

평가 요구사항에 따라 소프트웨어 제품을 측정/검증하는 활동을 수행하여 결과를 도출하는 단계로 (그림 9)에 평가 수행 과정의 활동을 나타내었다.

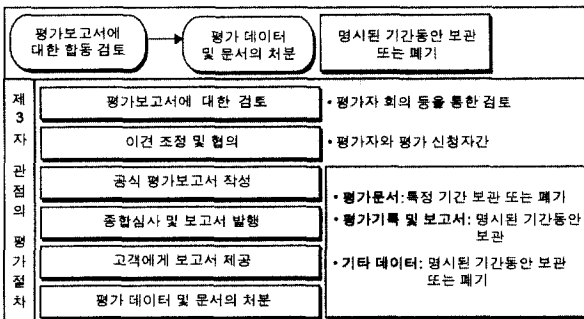


(그림 9) 평가 수행

평가 수행 과정의 세부 활동은 ISO/IEC 14598-5의 평가 수행 과정과 마찬가지로 “제품 구성요소 관리”, “평가 데이터 관리”, “도구 관리”, “검토 및 보고”의 과정을 거친다.

(5) 평가 결론

평가 보고서 검토 및 평가 데이터 처분에 대해 검토하는 단계로 (그림 10)은 평가 결론 단계에서 수행하는 활동을 나타내고 있다.



(그림 10) 평가 결론

평가 결론 과정에서는 ISO/IEC14598-5에서는 “평가 보고

서에 대한 합동 검토”와 “평가 데이터 및 문서의 처분”에 대해서만 다루고 있으나 시험 평가센터의 실제적인 과정을 고려하여 평가 신청자와의 “이견 조정 및 협의” 과정과 “공식 평가보고서 작성”, “최종 심사 및 보고서 발행”, “보고서 제공 및 문서의 처분” 과정으로 세분하였다.

4. 임베디드 소프트웨어 시험모듈

시험모듈은 품질평가를 위한 평가 매트릭에 대해 ISO/IEC 14598-6의 형식에 의거하여 평가를 위한 제반 사항을 문서로서 정의한 것이다. 임베디드 소프트웨어의 시험을 위한 모듈에 대해 기본적인 사항을 정리하면 다음과 같다.

4.1 측정 유형과 시험 유형의 종류

측정 유형이란 매트릭의 계산식을 구성하는 측정값들이 가질 수 있는 값의 형태를 의미하며, 시험 유형이란 매트릭의 결과값이 가질 수 있는 값의 형태를 말한다. 본 시험모듈에서 사용하는 측정 유형의 종류는 <표 3>과 같다.

<표 3> 측정 유형의 종류

측정유형	측정단위	표시기호
측정유형 1	Y: 만족함 N: 만족하지 않음 NA: 적용 불가능	(Y/N/NA)
측정유형 2	비율	Scale
측정유형 3	숫자	Number
측정유형 4	시간	Time

측정유형 1은 측정값이 Y/N의 형태로 판정되는 경우이며 NA(Not Applicable)는 측정 대상이 미비하거나 적용하기 곤란한 경우를 의미한다. 측정유형 2는 0부터 1사이의 비율값으로 나타나는 경우이며, 측정유형 3은 개수를 측정하는 경우와 같이 정수값의 형태로 측정값이 나타나는 경우이다. 측정유형 4는 정해진 시간에 일어나는 사건(오류발생, 결합의 복구 등)에 대해 측정하는 매트릭의 경우에 필요한 시간값을 나타낸다.

본 시험모듈에서 사용하는 시험 유형의 종류는 <표 4>와 같다.

<표 4> 시험 유형의 종류

시험유형	측정단위	표시기호
시험유형 1	Y: 만족함 N: 만족하지 않음 NA: 적용 불가능	(Y/N/NA)
시험유형 2	비율	Scale

시험유형 1은 매트릭에 대한 결과가 Y/N의 형태인 경우이며 NA인 경우는 매트릭을 적용할 대상이 미비하여 적용할 수 없는 경우이다. 시험유형 2는 매트릭의 결과가 비율

형태의 0과 1사이의 값으로 나타나는 경우이다.

4.2 시험모듈의 체계의 개발 내역

4.2.1 시험모듈의 체계

시험모듈은 품질시험에 관한 전반적인 사항을 정리하여 문서화한 것으로 시험의 개요, 기법, 메트릭에 대한 상세 내용, 적용 절차, 결과에 대한 해석 등을 포함하고 있으며 품질평가 프로세스에 관한 국제표준인 ISO/IEC 14598의 <부분 6>인 평가모듈의 형식에 근거하여 작성하였다. 본 연구에서 구축한 품질시험 모듈의 체계는 시험모듈에 대한 개요, 적용 범위, 참조문서, 메트릭에 대한 상세 설명, 시험모듈의 적용절차, 결과에 대한 해석 및 보고사항으로 구성된다.

개요 부분에서는 평가모듈의 기본 개념, 측정 목적, 메트릭의 범주, 메트릭의 이해를 위한 관련 용어의 설명을 포함하고 있으며 적용 범위에서는 적용 대상 및 필요 자원, 적용할 수 있는 시험 기법, 평가모듈 적용시 고려해야 할 관련 정보 등을 포함한다.

참조 문서에서는 메트릭이 도출된 관련 문서를 기술하고 메트릭 부분에서는 측정 항목 및 방법, 측정 항목을 이용한 계산식을 정의한다. 적용 절차 부분에서는 시험을 수행하는 구체적인 절차와 방법에 대한 기술하며 결과 해석 및 보고 부분에서는 메트릭 결과에 대한 판정과 측정 결과의 해석 방법, 측정 결과에 대해 문서로서 보고해야 할 사항 등을 명시한다.

이상의 내용을 <표 5>에 나타내었다.

<표 5> 시험모듈의 구성

항 목	세부항목	개 념
개 요	메트릭의 개념	평가 모듈의 기본 개념
	측정목적	평가 모듈의 측정을 통해 무엇을 얻고자 하는가를 기술
	메트릭의 범주	메트릭이 속하는 소속
	용어설명	메트릭의 개념과 측정 목적의 기술에 포함된 용어의 해설
적 용	적용 대상 및 필요자원	메트릭을 적용할 문서나 프로그램 등의 대상물 기술
	기법	적용할 수 있는 시험 기법
	적용시 고려 사항	평가모듈 적용시 고려해야 할 관련 정보
참조 문서	메트릭이 도출된 관련 문서	
메 트 릿	측정항목	메트릭의 계산식을 구성하는 측정할 데이터 항목
	측정방법	측정 항목에 대한 구체적인 측정 방법
	계산식	측정 항목을 이용한 계산 방법
적용 절차	시험을 수행하는 구체적인 절차와 방법의 정의	
결 과 해 석	측정치외의 매핑	메트릭 결과에 대한 판정
	측정 결과 해석	측정 결과에 대한 해석 방법의 지침을 제시
	보고사항	측정 결과에 대해 문서로 보고해야 할 사항에 대한 명시

4.2.2 시험모듈 개발 내역

본 연구를 통해 <표 6>과 같이 상호운영성, 신뢰성, 사용성, 실시간성, 융통성에 대한 부특성 19개에 대해 총 51개의 메트릭을 개발하였다.

<표 6> 시험모듈 내역

특 성	부 특 성	시 험 모 들 수	계
상 호 운영성	연결성	<연결가능성> 외 2개	5
	상호동작성	<상호동작부결성> 외 1개	
신뢰성	성숙성	<문제해결이력정보제공> 외 2개	9
	결합허용성	<다운회피율> 외 2개	
	회복성	<데이터회복정보제공> 외 2개	
사 용 성	이해가능성	<예비지식정보제공> 외 5개	14
	학습성	<기능학습용이성> 외 1개	
	운영성	<오류복구용이성> 외 3개	
	신호도	<인터페이스조절가능성> 외 1개	
실시간성	시간효율성	<처리기한명세> 외 2개	6
	준수성	<실시간준수정보제공> 외 2개	
융 통 성	변경성	<환경설정변경정보제공> 외 1개	17
	시험가능성	<내장형시험기능정보제공> 외 1개	
	적용성	<이식편리성>	
	대체성	<데이터지속정보제공> 외 1개	
	공존성	<공존가능정보제공> 외 1개	
계	19		51

4.3 시험기준

품질시험을 수행하기 위해서는 시험결과에 대해 어떻게 판정할 것인가를 결정해야 한다. 시험 항목, 시험할 특성별로 시험을 수행한 후에 시험의 성공 여부를 판단하는 기준을 <표 7>과 같이 설정하였다.

<표 7> 시험 기준 설정의 예

시험 대상	시험할 특성 (메트릭)	합 격 기 준
상 호 운영성	연결가능성	연결 가능한 기기의 90% 이상과 연결됨
	상호동작부결성	연결 기기와의 데이터 전송 및 연동 동작에 이상이 없음
	∴	∴
신뢰성	결합회피율	운용 시간 중 발견된 결합의 수가 규정된 값을 넘지 않음
	다운 회피율	결합이 시스템의 다운을 가져오는 경우가 10% 미만
	고장 회피율	고장을 발생 시키는 정도의 심각한 결합은 전체 결합 중 20% 미만
	∴	∴
융 통 성	이식편리성	자신의 환경에 쉽게 적용시킬 수 있는 수준이 규정값을 준수함
	이식표준준수율	이식과 관련된 표준, 규약 등의 정보에 따라 90% 이상 구현
	∴	∴



일반적으로 품질시험 결과의 평균을 취하여 합격 기준 점수에 도달하는가에 따라 최종 판정 여부를 결정할 수 있으나 평균값이 기준에 도달했다라도 주요 시험모듈에서 문제가 발견되었을 경우 품질에 치명적인 결과를 가져올 수 있다는 문제가 있다. 따라서 최종 판정을 결정할 때 평가 결과의 평균을 고려함과 아울러 주요 시험모듈을 선정하여 반드시 도달해야 하는 기준을 설정하고 기준에 도달하는 경우에만 최종 판정 절차로 전이할 수 있도록 해야 한다.

4.4 품질검사표

품질검사표는 시험모듈에 정의된 매트릭을 기준으로 실제 품질 시험을 수행하는 과정에서 편리하게 활용할 수 있도록 필요한 핵심적인 사항들을 추출하여 정리한 표로서 매트릭명과 개념, 측정항목, 매트릭의 계산식, 결과의 영역, 결과값, 문제점 기술 부분 등으로 구성되어 있다. 이러한 품질검사표의 예를 <표 8>에 나타내었다.

품질검사표에는 기본적으로 매트릭명과 매트릭이 측정하고자 하는 내용에 대한 문장이 포함되어 있다. 측정항목은 계산식을 통해 매트릭을 구성하는 요소로 1개 또는 그 이상의 개수로 구성되며 항목 개요와 측정 방법에 대한 기술을 포함한다.

결과 영역은 계산식에 의해 산출되는 값이 나타날 수 있는 영역으로 매트릭들은 전체적으로 0과 1사이의 값으로 사상될 수 있도록 정의하였다.

<표 8> 품질검사표의 예

매트릭명	임베디드 시스템에 소프트웨어적인 기능과 하드웨어적인 기능분배성		
기능분배성	기능이 적절히 분배되었는가?		
측정 항목	A	임베디드 시스템의 전체 기능의 수 - 하드웨어적인 기능과 소프트웨어적인 기능의 총수	
	B	검토 결과, 적절하다고 판단된 기능의 수 - 빈번한 연산이 하드웨어로 구현되었는가/ - 고성능을 요구하는 기능이 하드웨어로 구현되었는가? - 업그레이드가 필요한 부분이 소프트웨어로 구현되었는가? ...	
계산식	$FD = \frac{A}{B}$ $FD = \frac{\sum_{i=1}^n Success\_TC_i}{Total\_TC_i}$ - Success_TC : 하드웨어적인 기능과 소프트웨어적인 기능의 총수 - Total_TC : 기능을 검토한 결과 소프트웨어, 또는 하드웨어로 구현한 것이 적절하다고 판단된 기능의 수		
결과 영역	0 ≤ 기능분배성 (FD) ≤ 1	결과값	
문제점			

4.5 점검표

점검표는 품질검사표를 이용하여 측정항목에 대한 측정을 수행하기 위해 작성된 테스트 케이스의 시험 목록이다. 예를 들어 <표 8>의 품질검사표에 있는 "기능분배성" 매트릭과 관련된 측정항목의 점검표는 <표 9>와 같이 작성될

수 있다.

<표 9> 점검표의 예

순번	기능명	2.1		2.2		비고
		실시간성		정확성		
		시간 효율성	기능 분배	기능구현 정확성 정보제공	기능구현 정확성	
F1	Multi-Codec	Y	Y	Y	Y	
F2	Firmware Upgrade	N/A	Y	Y	Y	
F3	다국어 지원	N/A	Y	Y	Y	
F4	Equalizer	N/A	Y	Y	Y	
F5	Play list 생성	Y	Y	Y	Y	
F6	Direct Encoding	Y	Y	Y	Y	
F7	광출력	N/A	Y	Y	Y	
F8	USB 인터페이스	N/A	Y	Y	Y	
...	...	...	...	...	...	
Y(D)의 갯수						
N의 갯수						
결과						

4.6 테스트 케이스

테스트 케이스는 점검표의 점검 항목을 테스트하기 위해 사례를 작성하여 시험함으로써 점검 항목에 대한 검토 결과를 획득하기 위해 작성된다. 예를 들어 <표 10>은 네트워크의 정상작동여부를 검사하는 기능에 대한 테스트 케이스를 나타내고 있다.

<표 10> 테스트 케이스의 예

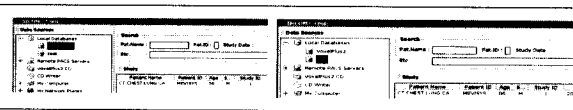
기능번호 : F5		기능명 : Remote PACS Server				
기능유형 : 관련 기능 수행						
테스트 내용 : Remote PACS Server 관련 기능의 정상 수행 확인						
테스트 케이스 번호	테스트케이스	기능 유형	입력자료	예상결과	실시결과	비고
Case-05-01	트리탐색기에서 "Remote PACS Server" 아이콘을 마우스 오른쪽 번으로 클릭하여 "Add Remote Server" 선택한 후 정상 작동 확인	선택	Title Address, description	설정된 Title로 Remote Database 생성	Yes	
Case-05-02	트리탐색기에서 "Remote PACS Server" 아이콘을 마우스 오른쪽 버튼으로 클릭하여 "Network Test All" 선택한 후 정상 작동 확인	선택	N/A	테스트 성공 메시지 Display	Yes	
Case-05-03	각 Remote 데이터베이스를 마우스 오른쪽 버튼으로 클릭하여 "Remove Server" 를 선택후 정상작동 확인	선택	N/A	Remote Server폴더 삭제됨	Yes	

4.7 문제점 기록서

문제점 기록서는 <표 11>과 같이 시험과정에서 발견된 문제에 대해 문제가 무엇인지에 대한 간단한 기술과 상서

기술 및 문제의 심각도가 어느 정도인지, 어떤 테스트 과정에서 발생된 문제인지를 기술한다.

<표 11> 문제점 기록서의 예

번호		문제점 발견자	
기능명	DB 폴더의 생성	문제점 등록일	
테스트 케이스 번호		문제 정도	결함 수정 권고
테스트 내역	로컬 데이터베이스에 대해 마우스 오른쪽 버튼을 누르로 "Add Existing Database"를 선택하여 기존의 DB 폴더에 연결되어 있는 DB 파일을 선택하고 연결함		
문제점	이미 폴더에 연결된 DB를 다른 폴더에 다시 연결할 수 있으므로 동일한 DB를 참조하는 다른 폴더의 변경에 영향을 받을 수 있음		
문제가 발행한 화면			
			
문제점 상세 기록			
같은 DB에 연결된 폴더들 중 하나를 선택한 후, 나타난 DB 목록중 일부를 삭제하면 DB를 공유하는 다른 폴더에서도 DB가 삭제되므로 원하지 않는 결과가 발생할 수 있으므로 이미 폴더에 연결되어 있는 DB에 대해 중복하여 연결할 수 없도록 할 필요가 있다고 생각됨			

4.8 시험결과서

점검표의 테스트 케이스를 사용하여 품질검사표에 대한 측정이 수행되면 각 메트릭별 측정 결과가 산출될 수 있다. 이 결과들을 품질특성, 부특성에 대한 메트릭별로 <표 12>와 같은 시험결과서로 정리된다.

<표 12> 시험결과서의 예

제품설명서 및 사용자 문서			
품질특성	부 특성	메트릭	측정값
상호운용성	연 결 성	제품정보제공(PIP)	0.80
		상호동작성	
신뢰성	성숙성	문제해결이력정보제공(PSI)	0.7
		문제해결률(PSR)	0.75
		결함회피율(DAR)	0.85
	결함허용성	다운회피율(DWR)	0.93
		고장회피율(FAR)	1.00
		...	...
∥	∥	∥	∥
용통성	분석성	진단기능정보(DFI)	Y
		진단기능지원률(DFS)	0.45
		...	...
	변경성	환경설정변경정보제공(CCI)	0.73
		변경가능률(CAR)	0.00
		...	...

4.9 시험성적서

시험성적서에서는 <표 13>과 같이 문제점 기록서 및 시험결과서를 바탕으로 하여 발견된 문제점에 대해 품질특성의 관점에서 전반적인 문제점을 제시한다.

<표 13> 시험성적서의 예

시험 항목별 결과 내역	
시험대상 : 제품설명서, 사용자매뉴얼, 임베디드 시스템	
품질특성	결 과
상 호 운 용 성	• 상호운용 가능한 다른 기기와 정상적으로 동작
신뢰성	• 데이터 회복에 관한 사항이 명시되어 있지 않음
사 용 성	• 사용자 매뉴얼을 명확히 이해할 수 있도록 하기 위한 예비지식(운영체제 관련 사항, 전문용어 등)에 대해 명세하고 있지 않음 • 온라인 도움말이 구비되어 있지 않아 기능 수행 중 특정 상황에서 필요한 도움말을 신속하게 찾아볼 수 없음
실시간성	• 의료 영상 생성과 같은 많은 시간을 필요로 하는 기능의 수행에 대해 성능 수준을 언급하고 있지 않음
용 통 성	• 사용 가능한 운영체제가 한정되어 있고 최신 버전의 서비스팩을 설치해야 하므로 설치에 어려움을 줄 수 있음

4.10 종합 판정

시험결과를 바탕으로 작성된 시험보고서는 심사위원회로 전달되고 심사위원들은 시험보고서와 다양한 심사 요인들을 검토하여 종합판정을 수행한다.

객관적인 심사를 위해서는 심사 요인들을 설정하여 명세화하고 적용방법에 대한 문서화를 명확히 할 필요가 있다.

5. 품질 측정과 평가 사례

본 평가 사례에서는 B사의 소규모 임베디드 소프트웨어를 대상으로 하여 평가를 수행하였으며 품질을 측정하고 평가한 사례를 통해 문제점과 개선 방안에 대하여 기술하였다.

5.1 메트릭에 대한 측정 결과

평가는 상호운용성, 신뢰성, 사용성, 실시간성, 용통성에 대해 수행하였으며 <표 14>에서는 사용성에 대한 측정 결과를 나타내었다. 측정 결과를 통해 각 메트릭에 대한 결과를 알 수 있고 상대적으로 취약한 특성을 파악할 수 있다.

경계값치리율, 인터페이스 표준준수율 등이 우수한 결과를 나타내고 있으며 데이터교환성, 접근통제가능성, 접근감시가능성 등이 낮은 값으로 나타났고 Y/N로 결과를 판정하는 메트릭의 경우 접근통제정보제공 메트릭의 경우 관련 정보를 제공하지 않는 것으로 나타났다.

〈표 14〉 사용성에 대한 측정 사례

특 성	부 특 성	메 트 릭	결 과
사용성	이해가능성	예비지식정보제공	Y
		기능이해도	0.87
		인터페이스이해도	0.93
		입/출력데이터이해도	0.88
		내용일관성	0.84
		사용자안내성	0.78
	학 습 성	기능학습용이성	0.81
		도움말접근용이성	0.75
	운 영 성	오류복구용이성	0.86
		메시지이해용이성	0.92
		운영절차조정가능성	0.88
		진행상태파악가능성	0.93
		인터페이스조정가능성	0.89
	인터페이스선호도	0.86	

5.2 품질부특성과 품질특성의 결과 집계

〈표 15〉는 품질부특성에 대한 집계 결과를 나타낸 것이다. 품질부특성의 집계는 〈표 14〉의 메트릭 결과로부터 각 부특성에 대한 메트릭값의 합계를 평균한 것이다. 이때 Y/N로 측정되는 메트릭은 Y를 1로, N을 0으로 하였으며 NA의 경우는 문서의 미비에 기인하는 것이므로 0으로 하여 계산하였다. 결과를 통해 각 품질특성별로 취약한 결과를 보이고 있는 부특성들을 확인할 수 있다.

〈표 15〉 품질부특성에 대한 집계표

특 성	부 특 성	결 과	특 성	부 특 성	결 과
상호 운영성	연결성	0.91	신뢰성	성숙성	0.83
	상호동작성	0.85		결합허용성	0.82
				회복성	0.87
사용성	이해가능성	0.88	실 시 성	시간효율성	0.92
	학습성	0.78		준수성	0.83
	운영성	0.89			
융통성	변경성	0.78	융통성	대체성	0.79
	시험가능성	0.83		공존성	0.82
	적용성	0.88			

〈표 16〉은 〈표 15〉를 정리하여 품질특성에 대해 백분율로 나타낸 결과이다.

〈표 16〉 품질특성에 대한 집계표

품질특성	상호운영성	신뢰성	사용성	실시간성	융통성
결과값	88	84	85	88	82
평 균	85				

5.3 문제점의 제시

품질평가는 결과를 산출하는 것뿐만 아니라 결과에 대한 문제점을 분석하여 개발자에게 제시함으로써 품질을 향상시키는 것이 목적이므로 〈표 17〉과 같이 품질특성 수준에서 평가 대상 소프트웨어에 나타난 문제점을 예시하였다.

〈표 17〉 문제점 예시의 일부

시 험 결 과 내 역	
시험대상 : 제품설명서 및 사용자 문서	
상호운영성	<ul style="list-style-type: none"> <li>타 기기와의 상호 동작 방법에 관한 설명 부족</li> <li>데이터 전송에 관한 오류 발생</li> </ul>
신뢰성	<ul style="list-style-type: none"> <li>문제점 해결 방안 제공 필요</li> <li>문제점에 대한 정보 부족</li> </ul>
사 용 성	<ul style="list-style-type: none"> <li>제품 사용을 위한 사전 숙지 정보 부재</li> <li>사용자에게 주어지는 피드백에 대한 설명 부족</li> </ul>
실시간성	<ul style="list-style-type: none"> <li>환경 설정 효과에 대한 정보 부족</li> </ul>
융 통 성	<ul style="list-style-type: none"> <li>환경 설정방법 상세 설명 필요</li> <li>문제점 해결 방법에 대한 정보 부족</li> <li>설치와 관련된 상세한 설명 부족</li> <li>최적의 운영환경 및 환경설정 방법 미비</li> </ul>

6. 결 론

일반 패키지 소프트웨어의 경우에는 몇몇 소프트웨어 개발 선진국이 개인용 컴퓨터에서 사용하는 소프트웨어의 대부분을 공급하고 있으며 개발 기술에서 우위를 점하고 있으나 임베디드 시스템 분야에서는 각 시스템마다 고유한 특성과 개성을 갖게 되는 만큼 특정 업체에서 독점할 수 없는 특징을 가지고 있다.

또한, 임베디드 시스템 산업의 급격한 발전과 높은 부가 가치로 인해 경쟁 우위를 점하기 위한 품질 확보의 필요성이 요구됨에 따라 임베디드 소프트웨어에 대한 품질평가 모델의 구축에 대한 연구의 필요성이 고조되고 있다.

본 논문에서는 ISO/IEC 12119의 품질특성을 참조하여 임베디드 소프트웨어 평가를 위한 평가모델을 개발하고 평가 과정에서 활용할 수 있는 품질검사표를 개발하였다. 최근 소프트웨어 제품의 품질향상에 대한 관심이 높아지고 있으며 국내 정보통신 분야에서 패키지 소프트웨어와 더불어 임베디드 소프트웨어의 품질평가 분야로 업무를 확대하기 위한 연구를 진행하고 있다. 본 연구 결과를 임베디드 소프트웨어의 품질평가 체계 구축에 활용할 수 있을 것이라 사료된다.

향후 연구 과제로는 본 연구를 기반으로 품질시험을 자동화 할 수 있는 도구의 개발이 필요하며 지속적인 시험을 통한 객관성 확보가 관건이 되리라 본다.

본 연구를 통해 임베디드 소프트웨어에 대한 품질 시험 체계 구축과 측정 기준의 개발 및 향후 실질적인 활용을 통해 고품질 임베디드 소프트웨어의 개발을 촉진하고 국제 경쟁력을 향상시킬 수 있을 것으로 기대한다.

### 참 고 문 헌

[1] ISO/IEC 9126-1, 2, 3, 4, "Information Technology-Software Product Quality," 2000.

[2] ISO/IEC 14598-1~6, "Information Technology-Software product evaluation," 1999.

[3] ISO/IEC 12119, "Information Technology-Software Package-Quality requirement and testing," 1990.

[4] N. F. Schneidewind, "Methodology for Validating Software Metrics," IEEE Trans. on SE., Vol.18, No.5, May, 1992.

[5] Moller, K. H. and Paulish, D. J., "Software Metrics," Chapman & Hall(IEEE Press), 1993.

[6] Wallmuller, E., "Software Quality Assurance A practical approach," Prentice Hall, 1994.

[7] 양해술, "Embedded S/W의 품질평가 모델 개발 연구", 한국정보통신기술협회 위탁과제, 1차년도 최종보고서, Nov., 2002.

[8] Ralf Niemann, Peter Marwedel, "Hardware/Software Partitioning using Integer Programming," European Design & Test Conference(ED & TC), pp.473-479, 1996.

[9] J. Boegh, S. De Panfilis, B. A. Kitchenham, A. Pasquini : A Method for software Quality Planning, Control, and Evaluation, IEEE Software, Vol.16, No.2, Mar.,~Apr., 1999.

[10] 水野幸男, "ソフトウェアの総合的品質管理", 日科技連出版, 1993.

[11] 吉澤. 東. 片山, "ソフトウェアの品質管理と生産技術", 日本規格協會, May, 1990.

[12] IAN Douglas Agranat, "Engineering Web Technologies for Embedded Applications," IEEE Internet Computing, May, 1998.

[13] 정기석, 김태환, "내장형 시스템 설계 : 개론", 한국정보과학회지, Vol.20, No.7, pp.5-13, July, 2002.

[14] 양해술, 이하용, "설계단계에서의 품질평가 톨킷(ESCORT-D)의 설계 및 구현", 정보과학회논문지(C), Vol.3, No.3, June, 1997.

[15] 양해술, 편용범, 이정립, 유규하, "의료용 소프트웨어의 평가기준 개발에 관한 연구", 정보처리학회논문지D, Vol. 10-D, No.5, Aug., 2003.

[16] 이하용, 황석형, 양해술, "패키지 소프트웨어 시험 프로세스와 평가모델의 개발", 정보처리학회논문지D, Vol.10-D, No.5, Aug., 2003.

[17] 양해술, 최민용, 박인수, "산업용 소프트웨어 국제표준 적합성 인증 시스템의 설계 및 구현", 정보처리학회논문지D, Vol.10-D, No.5, Aug., 2003.

### 양 해 술

e-mail : hsyang@office.hoseo.ac.kr  
 1975년 홍익대학교 전기공학과(학사)  
 1878년 성균관대학교 정보처리학과(석사)  
 1991년 日本 오사카대학 정보공학과 S/W 공학전공(공학박사)  
 1975년~1979년 육군중앙경리단 전산실 시스템분석 장교  
 1980년~1995년 강원대학교 전자계산학과 교수  
 1986년~1987년 日本 오사카대학교 객원연구원  
 1994년~1995년 한국정보처리학회 총무이사, 논문편집위원장  
 1995년~2002년 한국S/W품질연구소 소장  
 2001년~현재 한국정보처리학회 부회장  
 2003년~현재 미국 ACIS 학회 Vice President  
 1999년~현재 호서대학교 벤처전문대학원 교수  
 관심분야 : 소프트웨어공학(특히, S/W품질보증과 평가, 품질관리와 컨설팅, OOA/ OOD/OOP, CASE, SI), 프로젝트 관리, CBD기반기술, IT품질경영

### 신 석 규

e-mail : skshin@tta.or.kr  
 1991년 서울산업대학교 재료공학과(학사)  
 1998년 충남대학교 대학원 전산학과 석사 과정 수료  
 2001년~현재 한국정보통신기술협회 IT 시험연구소 소프트웨어시험센터 센터장  
 관심분야 : 소프트웨어공학(특히, S/W 품질시험과 품질평가, 품질관리와 BMT)

### 정 해 정

e-mail : jhjung@ptuniv.ac.kr  
 1988년 경북대학교 통계학과(이학사)  
 1991년 경북대학교 통계학과(이학석사)  
 1994년 경북대학교 통계학과(이학박사)  
 1995년~현재 평택대학교 정보통계학과 부교수  
 2001년~현재 경기도정보화추진위원, 한국산업표준심의회 S/W 공학분야전문위원(ISO/IEC JTC1/SC7/WG6 국제표준 전문가 활동), 평택시정보화추진위원, 한국정보통신기술협회 인증심의위원 등  
 관심분야 : 소프트웨어공학, 소프트웨어 모형결정, 소프트웨어 품질특성에 따른 평가, 소프트웨어 신뢰도 측정, 영상 처리