

ASTM을 적용한 A/V와 Mobile 시스템 소프트웨어 품질 향상에 관한 연구

한상섭¹⁾, 전경빈²⁾, 김정희³⁾, 박상득⁴⁾

삼성전자 CS 경영 센터 원류품질 혁신 팀 전문기술그룹

A Study on the Improvement of Quality for A/V and Mobile System Software Applied to ASTM (Automated Software Test Methods)

Sang Seop Han, Kyung Bin Chun, Kristin Chung, Sang Deuk Park

Advanced Technology Group, CS Management Center Samsung Electronics.Co.,Ltd

Abstract

As the function of digital products gets diverse and complex, more than 30~40% job of software developing and testing group within its development cycle, concentrates on

- 1) SPICE 국제 심사원, 소프트웨어 신뢰성 및 자동화 테스트 연구
- 2) 한국 과학 기술원 기계 공학 박사, SPICE 국제 심사원, 소프트웨어 품질 보증
- 3) SPICE 국제 심사원, CSQE, 소프트웨어 품질 보증
- 4) 시스템 신뢰성, 삼성전자 CS 경영센터 전문기술그룹 그룹장

the software testing[1]. To cope with consumers' quickly changing needs, development cycle time gets short, and the number of model and test items increase steadily. This is why automating software basic function and UI(User Interface) verification is needed [4][5][6].

This paper draws many strength and weakness defines Input/Output through integrating mobile phone, DVDP, PDA, and Black-Box (Intrusive (mobile) or Non-Intrusive(DVDP, PDA)) testing systematically, which brought above the 64% average of automation rate, and ensure 80% of test coverage [4][5][6].

1. 서론

오늘날 Embedded System의 품질은 소프트웨어 품질로 대변될 정도로 소프트웨어 품질에 대한 중요성이 대두되고 있다. 실제로 개발 단계의 많은 시간을 소프트웨어 테스트에 투자하고 있다. 디지털 제품의 소프트웨어는 제품의 모든 기능을 제어하는 중요한 역할을 담당하기 때문에 소프트웨어의 결함은 곧 시스템 결함으로 소비자에게 많은 불편함과 회사 브랜드에 많은 손상을 입힐 수 있게 된다 [1][2]. 하드웨어와 다른 점은 하드웨어의 경우 많은 결함이 개발 단계에서 테스트를 통해 결함을 재현하거나 하드웨어 자체에 테스트 백터를 내장하여 자체 진단하는 기능까지 구현되어 있다. 하지만, Embedded System Software는 한정된 메모리 크기에 최적화된 코드로 그 기능을 구현해야 하기 때문에 하드웨어와 같은 테스트는 자체적으로 해결하기는 어렵다. 또한, 소프트웨어의 비가시적인 특성 때문에 잠재적인 결함을 발견하기 어려우며 시장에서 발견된 결함을 재현하기 또한 쉽지는 않다. 이러한 디지털 제품의 소프트웨어 품질을 어떻게 하면 향상시킬 수 있을까? 본 논문에서는 소프트웨어 자동화 테스트 기법을 디지털 제품에 적용하기 위해 다음과 같은 몇 가지 관점에서 결론을 얻고자 접근했으며 본 논문의 본론 및 결론과 같이 DVDP, Cell Phone, PDA에 대해 아래와 같은 부분에 대한 실험 Data를 얻을 수 있었다[4][5][6].

- 1) 개발단계 소프트웨어 테스트 노력의 감소
- 2) 소프트웨어 인력의 단순 반복 시험 자동화
- 3) 납기 전 소프트웨어의 잦은 사양 변경에 따른 결함 파급 억제
- 4) 소프트웨어 테스트 자동화율 향상, 테스트 커버리지 확보

디지털 제품의 소프트웨어 품질 향상을 위해 현재처럼 단순히 테스트를 반복하거나 많은 시간과 인력을 투자한다고 해서 소프트웨어 품질이 완성되는 것은 아니다. 소프트웨어 품질은 일반 응용 소프트웨어뿐만 아니라 Embedded System Software 또한 CMM⁵⁾, SPICE⁶⁾와

5 CMM : Capability Maturity Model

6 SPICE : Software Process Improvement and Capability dEtermination

같은 SW Process를 통해 조직의 Software Process 성숙도가 측정되어야 한다. 하지만, 본 논문에서는 소프트웨어 자동화 테스트 기법 적용을 통해 관리적인 요소를 제외한 공학적인 부분에 대한 품질만을 다루고자 한다. 따라서, A/V 제품(DVDP)과 Mobile 제품(Cell Phone, PDA) 사례 연구를 통해 자동화 테스트를 위한 업무 프로세스, 테스트 환경 구성 등 소프트웨어 품질을 향상하기 위한 여러 가지 노력들을 본문에서 소개하고자 한다.

1.1. 소프트웨어 자동화 테스트 프로세스

Embedded System Software의 소프트웨어 자동화 테스트 방법은 해당 디지털 제품의 입출력 정의에 따라 Black-Box test로써 구현 될 수 있으며 보다 높은 자동화 율 및 Test coverage를 구현하기 위해서는 Intrusive System Integration 작업이 필요함을 알 수 있었다. 실제로 Mobile Phone의 경우 제품의 입출력 부분인 LCD Driver와 Key Pad Driver에 대한 연구 후 <그림 2> 와 같은 시험 환경을 구성하여 초기 시험 때 보다 30% 이상의 자동화 율을 높일 수 있었다. 시험 환경 구성만으로 소프트웨어 자동화 테스트의 효율을 높일 수는 없다. <표 1>은 디지털 제품의 소프트웨어 자동화 테스트 작업 절차를 나타낸 것이다. 소프트웨어 자동화 테스트를 위해 해당 제품의 입출력 관계 및 인터페이스 관계를 분석하고 개발 제품 및 모델의 특성을 파악하여 테스트 중점 사항을 도출하며 시스템의 제약 조건 및 소프트웨어의 기능 요구사항 그리고 구조를 분석해야 한다.

소프트웨어 자동화 테스트에서 가장 중요한 소프트웨어 테스트 Spec.은 각 제품 및 모델 별로 테스트 의뢰자가 소프트웨어 요구사항에 맞게 제출해야 하며 이 Test Spec.은 시험 Pass/fail의 기준이 된다. 이러한 시험 사전 작업을 거친 후 해당 제품의 입출력을 정의하여 시험 환경을 구성한다. 이때 정의된 입출력 관계가 시험 환경에 적합 하지 못할 경우 <그림 2>와 같은 tailoring 작업이 필요하게 된다.

자동화 스크립트 작성은 항상 처음과 끝이 동일해야 하며 해당 소프트웨어 구조를 완벽히 분석하여 모든 기능을 활성화 시키고 검증해야 한다. 이때 소프트웨어 전원 On/Off 와 같은 반복적인 작업은 소프트웨어 신뢰성 시험 항목으로 구성하여 소프트웨어뿐만 아니라 시스템 시험으로 발전시킬 수 있다.

소프트웨어 자동화 시험을 위한 모든 사전 작업을 완료한 후 테스트 스크립트를 Operating 시키고 결과 Log를 분석하면 시험을 완료 하게 된다. <표 1>의 소프트웨어 자동화 테스트 프로세스는 Mobile Phone, DVDP, PDA 제품을 시험 하면서 정착되었으며 자동화 율 및 Test Coverage 측정의 결과 기준은 제품별 특성에 따라 Test Spec.을 기준으로 측정되었다.

<표 1> 소프트웨어 자동화 테스트 프로세스

Process	Output
1) 시스템 분석	시험 계획
2) 소프트웨어 분석	시험 요구사항 소프트웨어 Menu Tree 소프트웨어 중요 고려사항
3) Test Spec. 의 분석	Test Scenario의 설계 시험 결과의 표준화
4) 시험 환경 구축	시험 시스템 설계 시스템 입출력 재정의
5) 스크립트 작성	시험 수행
6) 소프트웨어 신뢰성 시험	Software Stress, Repeated, Aging test Power On/Off Testing
7) 결과 분석	결합 검출률, Test coverage etc.,

1.2. 소프트웨어 자동화 테스트를 위한 선결 조건

Embedded system software 자동화 test를 위한 선결 조건은 두 가지로 나누어 설명 할 수 있다.

- 1) Test Specification 및 Checklist는 필수적으로 준비되어 있어야 한다.
- 2) 시험대상 시스템의 입출력이 명확히 정의되어 있어야 한다.

첫 번째, 해당 제품의 소프트웨어 테스트 Specification과 결과 기준이 명확해야 하며 시험에 필요한 시스템 Specification, software specification을 바탕으로 정확한 Test Specification이 필요하게 된다. 두 번째, 정확한 입출력 정의를 통해 시험 환경을 구성 할 수 있다. 소프트웨어 자동화 테스트의 가장 중요한 부분으로써 해당 시스템의 입출력 정의는 Test Coverage 및 자동화에 많은 영향을 준다. 실제로 mobile phone의 경우 전체 Test process의 45%이상을 Test Environment 구성에 투자하였고 83%의 가장 좋은 자동화 율과 90%이상의 Test Coverage를 확보 할 수 있었다.

2. 본 론

2.1.소프트웨어 자동화 테스트 기법

2.1.1.시스템 분석

보다 높은 Test coverage와 자동화 율의 결과를 얻기 위해서는 test를 하기 전에 해당 제품에 대한 연구가 필요하다. 해당 제품의 소프트웨어 기능 특성은 무엇이며 어떤 특징을 갖는지? 하드웨어와의 인터페이스는 어떤지? 소프트웨어 기능 구조는 어떤지? 소프트웨어의 어떤 기능이 시스템 전체 기능에 영향을 미치는지? 에 대한 연구가 선결되어야 하며 이를 바탕으로 test의 resolution이 결정된다. 본 논문의 mobile phone, DVDP, PDA는 <표 2>과 같은 특성 및 특징과 기능을 갖고 있다.

<표 2> Defined I/O of Products

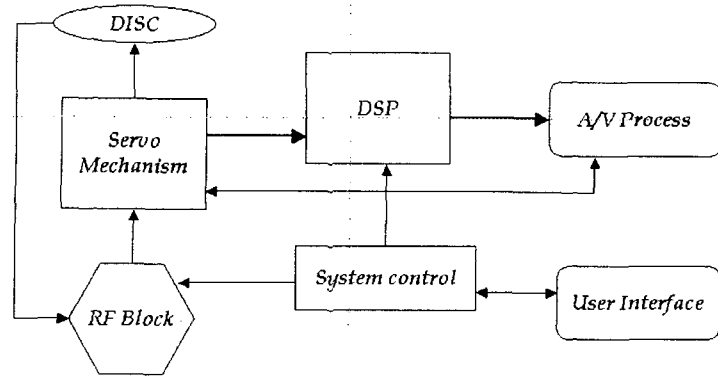
Products	I/O	Type of Test
Mobile phone	Keypad/LCD	Intrusive
DVD Player	Remote Control/S-Video	Non-Intrusive
PDA	Touch Screen, USB etc.,	Non-Intrusive

2.1.2.소프트웨어 분석

소프트웨어 자동화 테스트는 Black-Box Test기법을 사용하므로 소프트웨어 내부 프로그램 자체 내용은 중요하지 않다. 단지, 시험 대상 시스템의 소프트웨어가 어떤 구조로 구성되어 있으며 어떤 상태 천이를 갖고 어떤 제어 논리를 구현하는지에 대한 정보 분석을 통해 해당 제품의 menu tree diagram이나 <그림 1>과 같은 state diagram이 직접 test script 구성에 유용하게 사용되어진다.

2.1.3.소프트웨어 시험 Specification 의 분석

Test specification은 제품별로 그 형태나 type이 매우 다양하다. DVDP의 경우 표준 Navigation Disc.(NF002)의 NTSC 및 PAL 방식에 따라 표준 Test Navigation Disc.를 표준 reference로 Test를 하였으며 mobile phone 및 PDA는 소프트웨어 Menu Tree에 따라 그 영역을 한정 지어 Test Specification을 구성하였다.



<그림1> DVDP의 소프트웨어 구조

2.1.4. 시험 환경의 구성

Mobile phone LCD의 경우 2 gray, 4 gray로 발전하면서 현재 256 Color, 65000 color 그리고 TFT-LCD에 이르기까지 급변하고 있다 이런 LCD는 소프트웨어 자동화 테스트 환경을 구축하는데 많은 걸림돌이 된다. <그림 2>는 이러한 LCD의 system Integration 사례를 보여주고 있다. Integration 작업은 크게 입력 부분의 Keypad와 출력 부분의 LCD 부분으로 나누어 keypad driver, LCD driver에 직접 wiring을 통해 RS232C 통신으로 신호를 호스트 PC에서 제어 할 수 있도록 구성 되어있다.

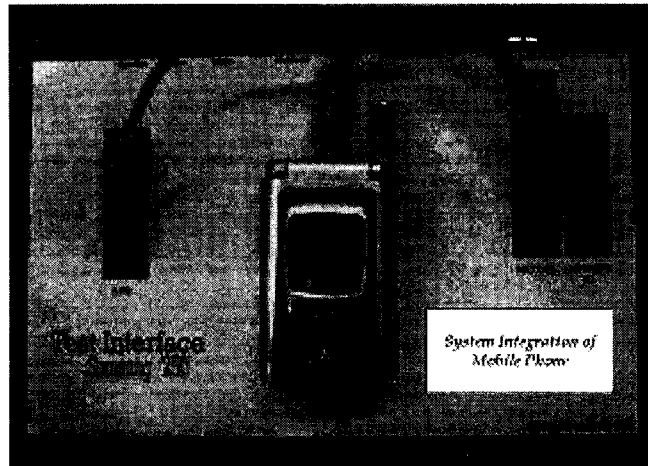
이러한 Intrusive 방법은 보다 높은 자동화율과 Test coverage를 확보하는데 큰 도움을 얻을 수 있었다.

2.1.5. 자동화 스크립트의 작성

<그림3>은 Mobile phone의 자동화 테스트 scripts의 실례이다. 자동화 스크립트의 구성 원칙 및 동작 요구사항은 몇 가지로 나눌 수 있다.

- 1) 자동화 Test scripts의 처음과 끝이 일정 해야 한다.
- 2) Test scenario에 대한 Pass/Fail Log에 따라 결함 지점을 검색하는데 어려움이 없어야 한다.
- 3) 사용자정의 함수 및 단일 Test 기능 함수들이 random하게 동작 할 수 있어야 한다.
- 4) 자동화Test scripts 동작은 시스템 timing 문제에 민감하므로 정확한 실시간 제약 조건을 수행 하도록 한다.

위와 같은 사항들은 자동화 테스트 툴을 실제 제품에 적용하여 얻어낸 정보들이며 Embedded system에서 자동화 테스트를 구현할 경우 꼭 고려해야 할 사항들이다.

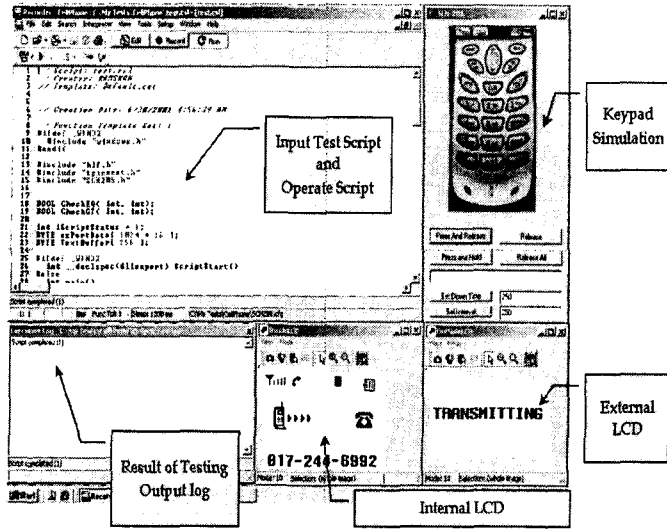


<그림 2> 시험 환경 구성

2.1.6. 소프트웨어 신뢰성 시험 사례

소프트웨어 신뢰성 시험은 대부분 단순 반복적인 시험 기능을 무한 반복 수행하여 소프트웨어 결함보다는 시스템 결함을 찾아내는데 그 목적이 있다. 우리는 mobile phone, DVDP, PDA의 Power on/off 및 기타 단순 반복 기능에 대해 무한 반복 Test를 구현하여 시스템이 Down 되거나 소프트웨어의 특정 기능에 결함이 발생하는 현상을 관찰 할 수 있었다. 그 이유는 Embedded System의 경우 하드웨어와 소프트웨어의 기능이 논리적으로 완전히 분리된 것이 아니라 상호 연동(Interlocking)되어 최종 기능을 발휘하기 때문에 긴밀한 관계 Interface 상태에 있다는 것을 간과하지 말아야 한다. 다음은 간단한 Power on/off 예제이다.

```
int Power_On_Off_Testing_DVDP()
{
    for(;;)
    {
        ir_SendSignal(Power);
        verify;
        delay(time);
        ir_SendSignal(Power);
        verify;
        delay(time);
    }
    return 0;
}
```



<그림 3> 소프트웨어 자동화 시험 스크립트

2.1.7. 결과의 분석

자동화 테스트 결과에 대한 분석은 다음과 같은 지표로 완성 될 수 있다. 우리는 mobile phone, DVDP, PDA에 대한 자동화 율 및 Test coverage 그리고 결함 검출 률에 대한 지표를 분석하여 자동화 테스트 효율을 도출하였다.

식1),2),3)은 제품별 혹은 모델별로 그 factor가 변할 수 있으며 factor가 세분화 될수록 보다 정확한 통계 지표로 활용 할 수 있다.7)

식 1) 자동화 율 = (자동화 가능 건수/Test 건수)*100

<표 3> 자동화 율

Products	Test 건수	자동화 건수	자동화 율
Mobile Phone	1580	1311	82.97%
DVDP	168	92	54.76%

식 2) Test Coverage = (Test 건수/Test Specification)*100

7 본 논문의 지표 자료는 삼성전자 기술 자료 보안상 일련의 데이터를 임의로 재 설정한 것입니다. 삼성전자의 기술 자료 보안 규칙상 본 논문의 실 데이터는 임의 작성되었습니다. 실 데이터에 대한 문의는 ramshan@samsung.co.kr로 문의해 주시기 바랍니다.

<표 4> Test Coverage

Products	Test Spec. 건수	Test 건수	Test Coverage
mobile phone	1755	1580	90%
DVDP	184	168	91%

식 3) 결함 검출률 = (결함건수/Test 건수)*100

<표 5> 결함 검출률

Products	Test 건수	결함 검출 수	결함 검출률
mobile phone	1580	47	2.97%
DVDP	168	8	4.76%

3. 결 론

Embedded system software 의 품질 향상을 위한 소프트웨어 자동화 테스트 기법의 적용은 <표 6>과 같은 시간 단축 률의 정량적 결과뿐만 아니라, 개발 단계 중 소프트웨어 버전 변경에 따른 에러 파급 영향의 잠금(Locking) 장치로써 많은 효과를 볼 수 있었다. 그 외 다른 정량적인 실험 결과는 본론에서 소개되었다.

<표 6> 테스트 효율 측정

Division	비 자동화	자동화	시간 단축 율
Computer	2700hr	1260hr	53%
DVDP	1120hr	422hr	62%
Visual Media	320hr	210hr	34%

또한 이러한 자동화 테스트 기법의 적용은 테스트 인력의 효율적 활용과 테스트 기술의 고급화라는 비가시적 효과를 얻을 수 있었다. 소프트웨어 자동화 테스트 기법을 적용하면서 만일 제품에 정의된 입출력을 상용 틀에서 커버하지 못하면 테스트 효율 및 시간에 대한 노력이 많이 소요된다. 또한 Test reference의 입력 시간이 많이 소요되기 때문에 개선이 필요했다. 현재 우리는 보다 많은 디지털 제품에 소프트웨어 자동화 테스트 기법을 적용하기 위해 많은 tailoring 작업을 시도하고 있으며 보다 많은 기능 결함들을 검출하기 위해 비정상 테스트 케이스들을 수집, 분석하고 있다.

참고문헌

- [1] Elefriede Dustin, Jeff Rashka, John Paul (1999), *Automated Software Testing*, Addison-Wesley, ISBN 0-201-43287-0
- [2] Cem Kaner, Jack Falk, Hung Quoc Nguyen (1999), *Testing Computer Software*, Wiley ISBN 04-7135-8460 .
- [3] Ron Patton (2001), *Software Testing*, SAMS.
- [4] Han sang seop (2001), Report of Automated Software Test for Mobile Phone in Samsung Electronics.
- [5] Han sang seop (2000), Report of Automated software test for DVDP in Samsung Electronics.
- [6] Han sang seop (2002), Report of Automated software test for PDA in Samsung Electronics.