

논문 2012-1-5

목적코드 화일의 유사도 측정 개선방안

이규대*

A Complimentary verification of the Object code file

Kyu-Tae Lee*

요 약

정보기기는 독자적인 기능의 구현과 성능개선을 위한 목적으로 특정한 프로세서와 특정 입출력 인터페이스를 갖는 하드웨어 시스템이다. 정보기기의 개발에서 분쟁이 발생한 경우, 저작원의 소유자를 검증하기 위한 자료로써 소스코드가 사용되어야 하나, 도용이 의심되는 측에서는 이를 은폐하기 위해 소스코드를 제공하지 않는 경우가 발생한다. 이 경우 실행코드 만으로 양측의 분쟁내용을 감정해야하며 역공학기법, 기능 실행의 시퀀스 비교 등의 비교기법을 적용한다. 이때 목적코드화일 데이터 자체를 비교하게 되거나, 역어셈블 방법으로 비교하게 되는 경우 컴파일의 선택에 따라 유사도의 편차가 크게 나타나는 기계어 번역의 특성으로 유사도 결과에 대한 객관성 부여가 쟁점이 된다. 본 연구에서는 이러한 실행코드 비교시 감정인이 규정하는 제한사항에 대해 다룬다.

Abstract

IT device is a hardware system which has special processor and interface modules for it's own function and performance upgrade. On property right verification of these system, there could be a mis-result when they have open source and general known program code by a interface module company. For verifying owner's copyright by object code only, it would be a difficult process to be justify. This paper shows the guideline to keep the property right of owner on the comparison only with object code verification.

한글키워드 : 목적코드, 유사도 측정, 가중치 설정

1. 서 론

정보기기는 프로세서를 기반으로 하는 입출력 하드웨어 회로보드와 운영체제를 기반으로 하는

응용소프트웨어가 FlashROM에 삽입되어 일체형으로 제작되고 있다. 정보기기의 제작자 관점에서는 독자적인 기능의 구현과 성능개선을 위한 목적으로 특정한 프로세서와 특정 입출력 인터페이스를 갖는 하드웨어를 설계하고 관련 인터페이스 프로그램을 제작, 구현한다. IT정보기기의 복제도 검증시, 객관적으로 고려 가능한 감정 체크리스트를 기준으로, 각 기능에 대한 공개된 기

* 공주대학교 정보통신공학부
(email: ktlee@kongju.ac.kr)
접수일자: 2012.5.15 수정완료: 2012.6.11

능을 조사하여, 가중치를 조정하는 방법으로 일반에게 공개되는 부분과 제작자가 독자적으로 구현한 사항을 객관화 할 수 있는 감정방법이 요구된다.[1]

이를 위해 정보기기의 구동방식, 입출력 동작 방식, 통신방식 등, 다양한 기능에 대한 공개회로와 개발된 프로그램에 대한 세부적인 차이점을 분석하고, 유사성을 검출하는 과정이 필요하다.

이때 소스코드 기준의 감정을 어렵게 하는 부분이 소스코드의 제공이 어려운 경우이다. 분쟁 당사자들의 자료제공이 선행되어야 하나, 복제 의심을 받는 측에서 소스코드를 거부하는 경우, 또는 감정의 특성상 정보기기만 확보되어, 제품의 내부에 저장된 실행코드만으로 검증해야 하는 상황들이 발생한다. 또한 소스코드가 확보되지 못하고, 목적코드의 실행화일로 제공된 경우, 유사성 비교에 있어 기능적 동작의 유사성 및 기술적 유사성에 대한 판단에 공지기술의 적용 범위를 검증한다.

본 연구는 실행코드에 대한 유사성비교에서의 과정 및 유사도의 객관성 확보에 대해 다룬다.

2. IT 정보기기 특징

프로세서를 기반으로 제작되는 정보기기는 기존의 대형시스템을 기반으로 개발되는 소프트웨어 중심의 기능에서, 휴대 가능한 소형 시스템으로 구현되고 있다. 그리고 휴대가능한 정도의 크기로 제작되기 위해서는 전원사용의 시간과 내부 프로세서의 처리용량의 제한으로 기능의 제한이 따르는 단점이 있다. 그러나 사용자의 요구에 따라 기능을 제한하고, 사용시간을 최소화 하는 방법으로 다양한 정보기기가 제작되고, 기기에 응용프로그램을 탑재하여 다양한 정보처리가능한 유용한 기기로 일상에서의 사용범위가 확

대되고 있다.

휴대가능한 정보기기의 내부구성을 보면 그림 1과 같이 핵심기능을 담당하는 프로세서를 중심으로 프로그램의 임시저장소로 활용하는 RAM, 응용프로그램을 저장하는 FlashROM (EEPROM), 정보기기의 입출력용으로 사용되는 키보드, LCD 모니터(LCD 터치패드), 그리고 외부 통신이 가능하도록 하는 통신포트(USB, COM) 등이 연결되어 있다.

이러한 하드웨어적인 구성을 바탕으로 해당되는 프로그램언어를 사용하여, 응용프로그램을 제작하고, 완성된 프로그램 실행코드를 FlashROM에 저장하고, 전원을 입력하여 동작시킨다. 이때 응용프로그램은 프로세서의 각종 기능과 주변장치(key pad, LCD monitor, LCD touch screen, 통신포트)를 효과적으로 활용하여, 사용자에게 유용한 기능의 단말기기를 제공한다.

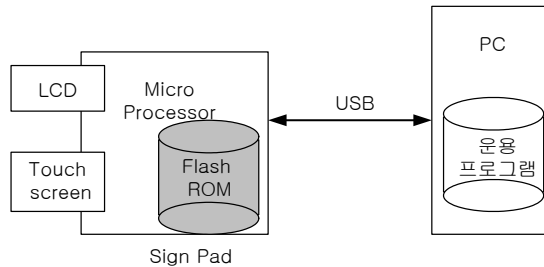


그림 1 마이크로 컨트롤러 인터페이스 구성

정보단말기의 중요정보는 USB등의 통신매체를 사용하여, PC와 연동이 가능한 구조로 제작된다. 프로그램 작성을 위해 사용되는 언어는 사용된 프로세서와 일치되는 언어로 작성되며, compile 과정을 통해 특정 프로세서에 일치하는 실행코드로 완성된다.

이상의 내부구조를 갖는 하드웨어는 미관을 고려하는 design 작업을 통해 그림2와 같이 사용하기 편하고, 디자인이 우수한 형태로 제작되어

구매력이 좋은 제품으로 완성된다.

단말기의 기능이 다양해지고, 핵심프로세서의 기능이 향상됨에 따라서, 단순한 기능의 단말기에 만족하지 못하는 소비자의 요구에 따라 기기의 기능은 데스크 탑 컴퓨터와 유사한 동작이 가능하도록 발전하고 있다. 이를 위해 임베디드 정보기기라는 구조로 개발된 정보기기에는 운영체제(OS)로 알려진 시스템관리용 프로그램이 탑재되어 제작된다. 운영체제는 데스크 탑 컴퓨터의 windows XP와 유사한 winCE, Android 등 주변기기, 메모리, 파일, 인터페이스 등 시스템의 모든 기능을 관리하는 능력을 가진 소프트웨어를 의미한다.

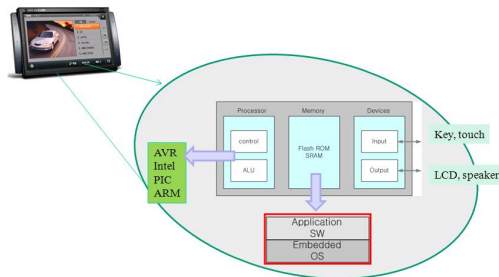


그림 2 임베디드 정보기기 구조

운영체제가(OS)가 탑재되어, 단말기를 실시간으로 동작시키는 기기를 임베디드 정보기기라 한다. 과거 OS가 포함되지 않은 단순기능의 정보기기와 비교할 때 기능면에서 전혀 다른 정보기기로 설계된다.

따라서 소프트웨어 부분은 단순기능의 응용프로그램에 운영체제가 추가되어, 운영체제를 탑재하고, 운영체제에 실행 가능한 프로그램을 제작하고, 실행하게 된다.

3. 정보기기 기능비교 항목

정보단말기는 사용자의 요구에 의해 새로운 기능이 계속 추가되고 있으며, 특히 동일한 입력장치에서 다양한 기능을 구분하기 위해 시퀀스 지연기능을 사용하고 있다. 이 방법은 하드웨어적인 기술 및 소프트웨어적 기술이 연결된 기능으로, 유사도 감정시 응용 프로그램이나 하드웨어 모듈만의 비교방법에 포함되어야 할 항목이다.

여기서는 정보단말기의 하드웨어적인 기능과 프로그램에 의한 소프트웨어적인 기능을 구분하고, 비교항목을 도출해 본다.

3.1 하드웨어 기능

정보기기의 하드웨어는 그림1과 같이 프로세서를 중심으로 하는 입출력부분, 프로그램저장부분, 통신부분으로 구분할 수 있다. 최근에 출시되는 프로세서는 자체적으로 다양한 인터페이스 블록을 내장하고 있기 때문에 프로세서가 입출력 기능을 포함하는 경우가 많아지고 있다.

3.1.1 입출력부

1) 입력부

정보기기에서의 입력은 기본적으로 버튼을 사용하며, 최근들어 LCD touch pad가 안정화 되면서 대부분의 입력신호를 터치방식으로 인식하고 있다. 이 방식은 LCD 화면에 가상적인 버튼을 특정 위치에 표시하고, 입력을 기다리는 기능으로 버튼의 모양과 위치를 달리하는 방법으로 다양한 입력신호를 구분하여 입력할 수 있는 장점이 있다. 또한 이 기능은 일반화되어, 대부분의 프로세서에 LCD 터치패드용 입출력 기능 블록을 제공하고 있다. 따라서 세부기능의 복잡도는 프로그램에 의해 다기능화 된다.

2) 출력부

정보기기의 출력은 사용자에게 처리결과를 알려주는 부분으로, 주로 LCD 디스플레이, 소리, 진동 등이 사용된다. LCD는 프로세서에서 제공하는 인터페이스 블록을 소프트웨어적으로 동작하는 부분이고, 소리출력은 스피커를 통하여, 간단한 beep에서 음악이나, 언어문장이 사용된다. 또한 모터의 바이브레이션(진동)을 이용하여, 기기의 진동을 유발함으로써 사용자에게 출력 결과를 알려주는데 사용된다.

3.1.2 프로그램 저장부

정보기기의 프로그램과 데이터를 저장하는 부분으로, 프로세서에서 제공하는 내부 FlashROM, 데이터를 저장하기위한 외부 EEPROM, 외부 인터페이스를 사용하여 대용량의 데이터를 입출력하는 SD card, memory stick 등이 사용된다.

1) 프로세서 내부 플래시 메모리

프로세서 내부 메모리는 정보기기에 사용되는 프로세서가 지정되면, 해당 모델에 대한 규격에 의해 정해지는 부분으로, 프로세서의 모델이 알려지면 메모리 용량과 사용방법이 고정된다. 따라서 프로세서의 종류가 확인되면, 모델번호로 유사여부가 판정된다.

2) EEPROM

정보기기의 보안을 위한 정보나 사용자 접근을 제한하기 위한 정보, 또는 정보기기의 초기화 데이터를 저장하고, 프로세서가 초기 동작 시 참조하는 용도로 프로세서 외부 저장장치가 사용되는데, 이때 EEPROM이 사용된다. 이 메모리는 설계자의 목적에 따라 용량이 정해지며, 인터페이스 방식으로 설계자의 의도에 따라 달라진다. 따라서 동일한 모델과 동일한 용량의 메모리를

사용한 기기라면 복제 가능성에 가중치를 줄 수 있다.

3) 외부 SD card/메모리 스틱

대용량 데이터의 외부접근을 위해 사용되는 방법으로 정보기기외부에 해당 인터페이스 터미널을 제공하여, 일반적으로 사용되는 메모리카드를 연결하여 프로세서가 접근할 수 있도록 한 방법이다.

인터페이스 방식은 표준화되어 공개된 방식이고, 해당 프로세서에서 인터페이스 블록을 제공하는 경우가 대부분이어서, 이 기능을 사용하고 있는 상황에 대한 비교가 가능하다.

3.1.3 통신부

정보기기의 통신기능은 외부와의 자료교환을 목적으로 사용되며, 인터페이스 단자수를 줄이기 위해 병렬통신방식에서 직렬통신방식으로 변화하였다. 또한 사용자의 편의성을 고려하여 무선 통신방식을 채용하고, 무선인터넷기능을 추가하는 방향으로 발전하고 있다.

통신기능은 표준화된 기술을 사용하는 기능으로 통신방식이 유사한부분이 대부분이다. 기능비교 시에는 각 통신방식의 사용여부, 응용프로그램에서의 인터페이스 부분을 조사함으로써 유사성 검토가 가능하다.

1) 직렬통신부

직렬통신방식으로 많이 사용되는 기술은 RS232 통신으로 최소 3개의 선으로 외부와의 통신이 가능하도록 표준화되어있는 기능이다. 현재 대부분의 프로세서에 USART라는 이름으로 인터페이스 블록을 제공하고 있어서, 응용프로그램 제작시에 간단한 부프로그램을 작성하는 방법으로 통신기능 삽입이 가능하도록 되어있다. 따라

서 이 부분은 프로세서의 유사성이 확인되면 사용방법이 동일하게 나타난다.

최근에는 직렬통신기능에 전원공급을 가능하게 하는 이러한 USB가 직렬인터페이스의 표준으로 대체되고 있다. USB는 초기 1.0 버전에서 기능이 확대되어, 속도가 개선되고, 무선USB도 가능하도록 발전하고 있다. 이 방식도 프로세서 내부에 인터페이스 블록이 탑재되어 있어, 응용 프로그램을 통해 기능설정이 가능하도록 되어있다. 따라서 이방식도 프로세서의 사용 모델에 따라 정해진다.

이상의 정보기기의 유선통신방식은 프로세서 고유방식으로 차별화되기 때문에, 비교가 간단해질 수 있다.

2) 무선통신부

무선통신부는 인터페이스의 편리성을 위해 사용되는 기능으로 사용자는 특별한 인터페이스 선을 사용하지 않고, 무선중계기 근처에서 바로 통신이 가능한 기능이다. 이전에 비해 속도도 빨라지고, 인터넷과의 접속이 가능하게 되어 많은 정보기기에서 채용하고 있는 기능이다.

이 기능도 유선 인터페이스와 같이 표준화가 되어있는 기술로 많이 사용되는 것으로, Bluetooth, Zigbee, WiFi, wireless USB 등이 있다. 이 기능을 탑재하기 위해서는 특정한 인터페이스 칩을 사용해야 하므로, 정보기기의 비교 시 무선인터페이스를 사용하고 있다면, 사용되는 인터페이스 칩과 응용프로그램에서 구현된 부프로그램의 코드를 분석함으로써 유사성을 도출할 수 있다.

이상의 정보기기 하드웨어의 기능에 따른 유사성 비교는 표준화된 방식에 의한 부분을 제외하고, 응용프로그램에서 작성된 프로그램을 비교하고, 특정 인터페이스 칩이 사용되는 경우를 고

려하여 판단하는 것이 타당하다.

3.2 소프트웨어 기능

소프트웨어의 기능은 정보기기 하드웨어를 활성화 시키는 것으로 프로그램작업에 의해 세부 인터페이스 기능이 동작하는 역할을 하는 핵심적 부분이다. 실행환경을 관리하는 운영체제부분과 사용자와의 인터페이스부분을 담당하는 응용프로그램부분을 나누어지며, 정보기기 제작사에서 설정한 운영체제에 의해 기능비교가 달라진다. 운영체제는 정보기기의 전원을 동작하였을 때 최초 화면표시와 입력대기 기능에 의해 일차적인 유사성을 판단할 수 있다. 즉 사용된 운영체제에 의해 대부분의 기능이 결정되기 때문에 운영체제의 동일 여부가 비교의 큰 차이로 나타난다. 운영체제의 종류, 버전, 용량 등이 비교대상으로 포함될 수 있다.

응용프로그램은 사용자와의 직접적인 인터페이스를 제공하고, 운영체제를 통해 기기의 하드웨어를 연결해 사용자가 필요로 하는 기능을 구현하는 부분이다. 따라서 제작자의 고유기능과 아이디어가 포함되는 부분으로 사용되는 응용프로그램의 개수만큼 유사성 비교가 가능하다.

3.2.1 운영체제 환경

운영체제는 정보기기의 운영환경을 설정하고, 응용프로그램이 하드웨어와 인터페이스를 위한 접속기능을 제공하는 부분으로 프로세서와 사용자를 연결하는 부분이다. 또한 메모리관리, 프로세스관리, 인터페이스관리 등을 담당하기 때문에 정보기기에 어떤 운영체제를 탑재했는가에 따라, 같은 하드웨어 기기라도 운영환경이 달라진다.

운영체제의 비교항목은 운영체제의 종류, 모델이다. 예를 들어 Linux OS와 Window OS,

Apple OS 등의 운영체제가 각각 다른 기능을 제공한다. 따라서 운영체제의 구분으로 소프트웨어의 비교가 결정된다.

운영체제가 같은 경우는 초기화면의 표시, 입력정보 대기화면의 표시 등으로 비교항목을 작성할 수 있다.

3.2.2 응용프로그램

응용프로그램은 사용자인터페이스부분으로 사용자에게 편리한 입력기능과 출력기능을 제공하여, 정보기기 사용자의 기기활용도를 높이는 부분이다. 사용목적에 따라 기기의 하드웨어 부분과 접속되며, 운영체제의 기능을 효과적으로 사용한다.

정보기기의 기능은 이 응용프로그램의 동작에 의해 결정되는 것으로 제작자는 사용자의 요구에 적합한 응용프로그램은 많이 제작해서 보급하고 있다. 따라서 응용프로그램은 제작자 고유의 기능과 아이디어가 포함된 부분으로 응용프로그램의 기능비교를 통해 기기의 유사성 여부를 도출할 수 있다.

운영체제, 응용프로그램에 대한 기능비교가 가능하며, 통신부분의 프로그램은 하드웨어부의 기능비교항목에서 다루지 않은 부분을 중심으로 비교항목을 다룬다.

4. 실행화일의 비교 검토

프로세서 인터페이스를 참조하여 작성되는 소스코드(원시 프로그램)은 해당 프로세서가 인식하는 기계어(object code)로 번역되어야 한다. 이 과정을 담당하는 프로그램을 컴파일러라고 하며, 일반적으로 C 언어 형태의 프로그램언어가 사용되고 있다. 컴파일러는 해당프로세서의 모델, 프

로그램 코드 저장 위치, 클럭 속도 등을 고려하여 프로세서가 인식하는 기계어를 생성한다. 작성된 기계어는 프로세서내의 프로그램메모리에 복사되어야 하는데, 이 과정을 다운로드(download) 한다고 한다.

4.1 소스코드 변형에 따른 실행코드 변화

컴파일러가 기계어로 번역하는 과정은 컴파일러 선택사양에 따라 변화될 수 있다. 따라서 동일한 프로그램이라도 컴파일러의 옵션 파라미터가 달라지면 다른 기계어가 생성될 수 있다. 이러한 특성으로 기계어만으로 두 개의 프로그램을 비교하고, 유사도를 도출하는데 오류가 발생할 수 있는 부분이다.

동일한 기능의 프로그램의 경우라도, 기능에 관계없는 일부 코드가 삽입된 경우, 컴파일을 수행하여 생성된 기계어는 다른 모습으로 변형되어 유사도에 영향을 준다. 다음의 예는 동일기능의 프로그램에 i, j 변수를 사용하여 기능에 변화 없이 소스코드에 변형을 주고 기계어를 생성한 후 두 코드를 비교한 결과를 보인다.

소스코드의 주석부분은 컴파일 과정에서 무시되기 때문에 변형에 대한 변화가 없으나, 명령어 기능을 가지는 텍스트(변수)에 변형이 가해지면 실행코드에 일부 또는 전체적인 변화를 줄 수 있다.

```
while(1) {
    FORTE=0x06; //FND 0
    PORTC=Disp[sec1];
    Delay(10);
    i=i+1;
    FORTE=0x05; //FND 1
    PORTC=Disp[sec2];
    Delay(10);
    k_index++;
    if(k_index>50) {
        k_index=0;
        sec1++;
        if(sec1>9) {sec1=0; sec2++;}
        if(sec2>9) {sec2=0;}
    } //end k delay for count
} //end while
```

```

while(1) {
    PORTE=0x06; //FND 0
    PORTC=Displsec1;
    Delay(10);
    i=1;
    PORTE=0x05; //FND 1
    PORTC=Displsec2;
    Delay(10);

    k_index++;

    if(k_index>50) {
        k_index=0;
        //sec1=
        sec1++;
        if(sec1>9) {sec1=0; sec2++;}
        if(sec2>9) {sec2=0;}
    } //end k delay for count
    } //end while
    
```

그림 4 (j=1;) 코드의 추가(a), 삭제 예

이진코드의 비교는 Beyond compare 비교툴을 가지고 실행되었으며, 실행화면은 그림과 같다. 기계어 비교 파일을 원본과 비교본을 대상으로 비교하여 출력된 결과는 다음과 같이 83.9%의 유사도 결과를 보인다.

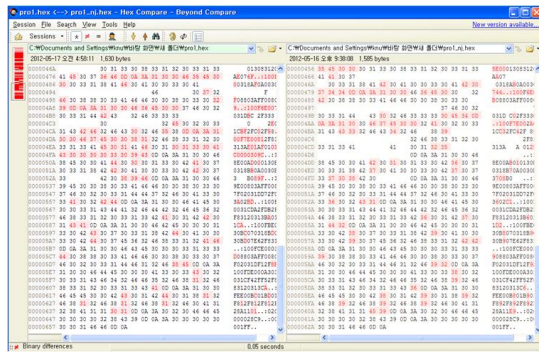


그림 5 실행코드 비교과정

비교도구의 실행후 리포트 내용은 완전히 다른 부분과 원본대비 상이한 부분, 비교본 대비 상이한 부분으로 구분하여 나타난다.

1601bytes(j) 1585bytes(nj)

```

2012-05-16 오후 9:43:05                      Hex Compare
Mode: All
Left file: C:\Documents and Settings\knu\바탕 화면\새 폴더\pr01_j.hex
Right file: C:\Documents and Settings\knu\바탕 화면\새 폴더\pr01_nj.hex
1344 same byte(s)
107 left orphan byte(s)
91 right orphan byte(s)
150 difference byte(s)
    
```

유사도 = 83.9%

다음은 i 변수를 추가하고, 컴파일과정으로 생성된 기계어의 비교결과를 확인한 결과는 81.2%의 유사성을 보였다.

1601bytes(j) 1630bytes(ij)

```

2012-05-16 오후 9:56:04                      Hex Compare
Mode: All
Left file: C:\Documents and Settings\knu\바탕 화면\새 폴더\pr01_j.hex
Right file: C:\Documents and Settings\knu\바탕 화면\새 폴더\pr01_ij.hex
1301 same byte(s)
193 left orphan byte(s)
222 right orphan byte(s)
107 difference byte(s)
    
```

유사도 = 81.2%

이상과 같이 소스코드의 기능을 유지하면서 소스코드에 불필요한 코드를 부가하고, 기계어를 생성하여 유사제품을 제작하는 경우, 기계어만의 감정 수행시 유사도에 영향을 미칠 수 있음을 알 수 있다.

4.2 실행코드 유사도 사례

다음은 분쟁 당사자가 실행화일만 제공한 상태에서 실행화일 4개의 상호 유사성을 도출한 내용으로 감정결과에 나타난 것처럼, 유사성 결과가 단순 비교상황을 보여주는 이상의 저작권관점의 도용 여부는 판단할 수 없는 결과를 나타낸다. 따라서 저작권 비교는 소스코드를 기준으로 했을 때 의미를 가지는 특징이 있다. 그러나 복제 의심을 받는 쪽에서 소스코드의 제공을 거부하는 분쟁의 경우, 최소한의 도용근거를 확보하는 차원에서 실행코드의 비교방법에 대한 연구는 계속되어야 할 것으로 보인다.

표 1. 실행코드 감정에

감정요청사항	감정결과
최초 제출된 기기의 롬데이터 HEX값과 현상에서 압수된 기기에서 추출한 롬데이터 HEX값을 비교하여 동일성 여부 확인	- 감정대상 4개화일 모두 최초 제출된 A파일과는 다르며, - B2, B3는 약 90%, B1, B4는 약 12%의 변형이 있음을 보인다. - 그러나 변형된 세부내용은 판별할 수 없다.

5. 결 론

정보기기의 인터페이스기능이 복잡해지면서, 유사한 프로그램소스를 사용하는 현상이 증가하고 있다. 이러한 정보기기 분쟁에서 소스코드 수준의 감정을 어렵게 하는 부분이 프로그램 소스의 불성실한 제공이다. 감정인이 정확한 유사성 도출을 위해서는 분쟁 당사자들의 자료제공이 선행되어야 하나, 복제 의심을 받는 측에서 소스코드를 거부하거나 감정의 특성상 정보기기만 확보되어, 제품의 내부에 저장된 실행코드만으로 검증해야 하는 상황들이 발생한다.

본 연구결과에서 보듯이 실행코드 만의 유사성비교는 유사도 결과에 대한 객관성을 확보가 미흡한 부분이 있다. 따라서 프로그램의 감정은 소스코드 중심의 유사도 검출이 될 수 있도록 분쟁 당사자 간의 자료제출에 소스코드 미 제출에 따른 불이익이 부가되도록 하는 제도보완이 요구된다.

참 고 자 료

- [1] Joseph Yiu, ARM Cortex-M3 guide, ITC, 2011
- [2] 전병태, "프로그램 복제도 감정기법 및 감정보 산출에 관한 연구" 프로그램심의조정위원회 결과보고서 2002.
- [3] 박선호, 오명환, ARM Cortex-M3 시스템 프로그래밍, D&W WAVE, 2010
- [4] www.st.com
- [5] 이규대, "임베디드시스템의 이진코드 추출 및 분석", 한국소프트웨어감정평가학회 논문지, 5권1호, pp27-38, 2009.5

저 자 소 개



이규대(Kyu-Tae Lee)

1984 고려대 전자공학과 졸업
 1986 고려대 전자공학과 석사
 1991 고려대 전자공학과 박사
 2001 미 조지아텍 교환 교수
 2006 미 일리노이주립대 교환 교수
 2007~2009: 한국전자통신연구원 이동통신연구
 구소 초빙연구원
 1992.3~현재 : 공주대 정보통신공학부 교수

<관심분야> 회로 및 시스템, 신호처리, VLC