

안드로이드 플랫폼 기반 모바일 방송 수신 시스템 개발

이정은, 최은창
한국전자통신연구원
e-mail: {jeleel, ecchoi}@etri.re.kr

Development of DMB Receiving System Based on Android Platform

Jung-Eun Lee, Eun-Chang Choi
Electronics and Telecommunication Research Institute

요 약

스마트폰의 급속한 확산으로 모바일 디바이스를 위한 개방형 플랫폼인 안드로이드의 시장 점유율이 지속적으로 증가하고 있다. 안드로이드는 미디어 프레임워크, 카메라, Wi-Fi, GPS 등에 필요한 하위 하드웨어 드라이버만 개발하면 보다 상위 계층은 기본적으로 제공되는 라이브러리와 프레임워크를 기반으로 짧은 시간 내에 어플리케이션 개발이 가능하다. 본 논문에서는 스마트폰의 주요 서비스인 모바일 방송 어플리케이션 개발을 위한 시스템을 제안한다. 제안한 시스템은 안드로이드 소프트웨어의 계층적 스택 구조로 되어 있으며, 이를 활용한 차세대 모바일 방송 규격인 AT-DMB 수신기를 설계 및 구현하였다. 또한, 연구개발 결과물의 수신 기능을 검증하였다.

1. 서론

급속한 스마트폰의 확산과 모바일 CPU의 성능 향상으로 다양한 형태의 모바일 디바이스가 출시되고 있다. 최근 구글에서는 모바일 디바이스를 위한 오픈 소스 기반의 소프트웨어 스택인 안드로이드를 공개하였다. 안드로이드는 리눅스 커널 기반의 운영체제와 개발 플랫폼을 포함하고 있으며, 통화 기능, SMS, 웹 브라우저, 계산기, 전자 달력, 미디어 플레이어 등 기본 어플리케이션들을 제공하기 때문에 모바일 디바이스 운영체제 시장에서 강세를 보이고 있다.

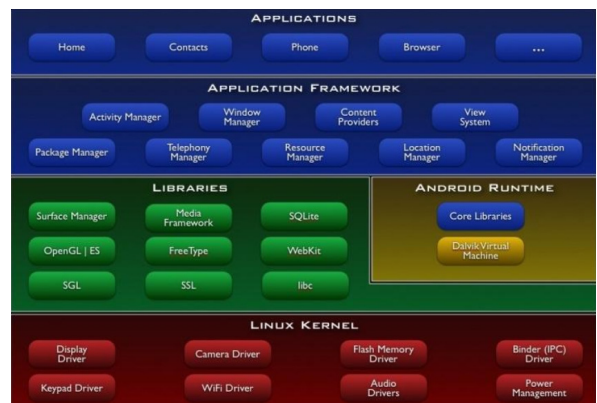
본 논문의 구성은 다음과 같다. 2장에서 안드로이드 플랫폼 구조를 설명하고, 3장에서는 안드로이드 플랫폼 기반의 제안하는 시스템에 대해 설명한다. 4장에서는 제안한 시스템 기반의 AT-DMB 수신기의 설계 및 구현 결과에 대해 기술하고, 5장에서는 제안한 시스템에 대한 수신 시험결과를 분석한 후 마지막으로 6장에서 결론을 맺는다.

2. 안드로이드 플랫폼

안드로이드 플랫폼은 그림 1에서 나타낸 것과 같이 소프트웨어 스택으로 구성되어 있다.

2.1 응용프로그램

안드로이드는 이메일, SMS, 전자달력, 지도, 브라우저, 연락처 등을 포함한 핵심 응용프로그램을 기본적으로 제공한다. 모든 응용프로그램은 자바 프로그래밍 언어를 사용하여 개발되었다 [1].



[그림 1] 안드로이드 소프트웨어 스택 구조

2.2 응용프로그램 프레임워크

응용프로그램 프레임워크는 자바 기반 응용프로그램 실행을 위한 API 인터페이스를 제공한다. 또한, 응용프로그램 구조는 컴포넌트들의 재사용이 용이하도록 설계되었다.

2.3 런타임

안드로이드 런타임은 응용프로그램을 실행하기 위한 환경을 의미하며, Dalvik VM과 핵심 라이브러리로 구성된다. 자바는 스택 기반 아키텍처인데 반해 Dalvik은 레지스터 기반이며, 모바일 디바이스에 최적화 되었다. Dalvik은 다중 Instance를 제공 가능하도록 설계되었고 리눅스 커널 기반의 스레드와 메모리 관리기능을 제공한다. 확장자가 .dex인 실행파일은 최소 메모리 사용을 통해 모바일 환경에 최적화 되어 있다.

2.4 HAL(Hardware Abstraction Layer)

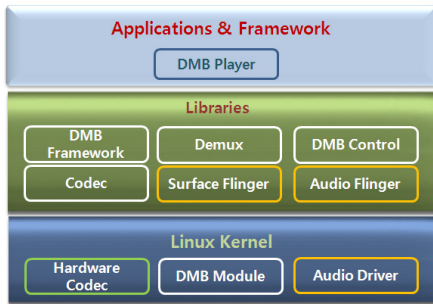
HAL은 안드로이드 플랫폼과 하드웨어를 연결해주는 계층으로 응용프로그램이 다양한 하드웨어 장치에 접근할 수 있는 인터페이스를 제공한다.

2.5. 리눅스 커널

안드로이드의 보안, 메모리 관리, 프로세스 관리, 네트 워크 스택, 드라이버 모델과 같은 핵심 시스템 서비스는 리눅스 커널 버전 2.6에 기반하고 있다 [1].

3. 안드로이드 기반 모바일 방송 수신 시스템

본 장에서는 안드로이드 기반 모바일 방송 수신 시스템을 제안한다. 안드로이드는 하드웨어와 밀접한 리눅스 커널 영역, 응용프로그램의 라이브러리, API 기반의 인터페이스를 제공하는 프레임워크 등으로 구성되며, 전체 구성 영역을 기준으로 제안하는 방식을 설명한다. 그림 2는 제안하는 모바일 방송 시스템의 소프트웨어 구성도를 안드로이드 플랫폼의 계층별로 도식한 것이다. 흰색 블록은 제안한 시스템 개발을 위해 추가로 구현한 부분이고, 노란색 블록은 안드로이드에서 제공하는 것을 활용한 부분이다. 초록색 블록은 칩 제조사에서 제공하는 인터페이스를 활용하여 개발한 부분이다.



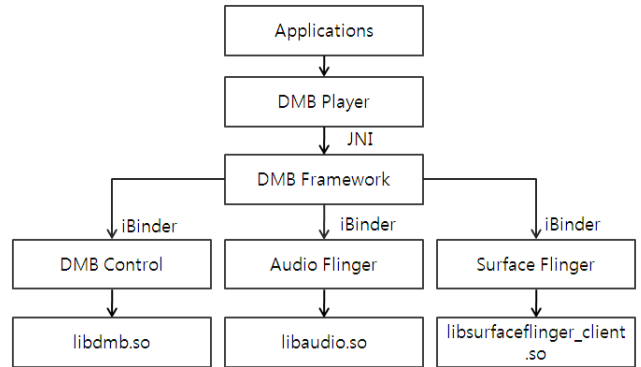
[그림 2] 모바일 방송 수신 시스템 소프트웨어 구성도

DMB Player는 모바일 방송 서비스를 위한 실행기능을 제공한다. DMB Framework는 DMB Player에서 필요로 하는 기능을 제공하는 라이브러리로 핵심 기능인 Demux, DMB Control 등의 라이브러리와 인터페이스한다. Demux는 전송 스트림을 Demultiplex하는 부분으로 모바일 방송 규격에 따라 상이하게 구현되지만, DMB Framework와 인터페이스는 동일하게 구현하여 방송규격이 바뀌어도 DMB Framework, DMB Player 그리고 응용프로그램들에게 영향을 주지 않아야 한다. 이것은 코덱에도 동일하게 적용된다. 다양한 모바일 방송규격들은 그 규격에 적합한 비디오, 오디오, 데이터 규격을 채택하고 있다. 예를 들면, 오디오의 경우 지상파 DMB는 BSAC을 사용하고, DVB는 AAC를 사용한다. 그러므로 복수의 모바일 방송을 지원하기 위해서 비디오는 비디오 코덱들을 위한 공통 인터페이스를 설계해야 한다. 이것은 오디오도 마찬가지다. Surface Flinger는 디스플레이를 위한 라이브러리이고, Audio

Flinger는 오디오 출력을 위해 안드로이드에서 제공하는 라이브러리이다. DMB Control은 리눅스 커널의 디바이스 드라이버와 연동을 위한 공통 인터페이스를 설계하고, 응용프로그램에서 디바이스 활용이 용이하도록 상위 계층을 위한 공통 인터페이스를 설계해야 한다. 리눅스 커널영역의 DMB Module은 모바일 방송 수신 RF모듈의 드라이버이다.

3.1 DMB 프레임워크

DMB 프레임워크는 모바일 방송과 관련된 기능을 제공하는 인터페이스 역할을 한다. 그림 3은 응용프로그램에서 리눅스 커널 이전 단계까지의 DMB 프레임워크의 동작흐름을 나타낸 것이다.



[그림 3] 모바일 방송 수신 프레임워크

DMB Player는 자바 언어로 개발되어 Dalvik VM에서 실행된다. 자바의 핵심 라이브러리 외에도, DMB 프레임워크와 관련된 기능들은 JNI(Java Native Interface)를 통해 C/C++ 라이브러리를 연결하여 사용한다. DMB 프레임워크를 포함한 하위계층은 모두 Native 라이브러리이다. DMB 프레임워크는 DMB Control, Audio Flinger, Surface Flinger를 바인딩하여 실행한다. DMB Control은 DMB 수신 모듈과 인터페이스되며, 제어 및 데이터를 수신하는 libdmb.so를 동적으로 구동한다. Audio Flinger는 오디오 출력을 위해 오디오 드라이버와 인터페이스되는 libaudio.so를 동적으로 구동한다.

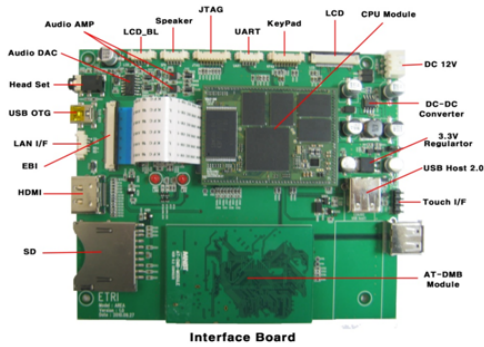
4. 구현

본 장에서는 제안한 안드로이드 기반 모바일 방송 수신 시스템을 적용한 AT-DMB 수신기를 설계 및 구현하였다. 표 1은 구현을 위한 개발 환경이다.

[표 1] 시스템 개발환경

구성요소	규격
CPU	SAMSUNG S5PV210 (1Ghz)
Memory	DDR2 SDRAM(512MB)
Flash	NAND Flash(256MB)
DMB 수신	AT-DMB 수신 모듈(ETRI)
USB	USB 2.0, USB OTG, Host 지원
개발 OS	Ubuntu 8.10
	Linux Kernel v2.6.32.9
개발 플랫폼	Android Froyo 2.2

AT-DMB 수신모듈은 아직 상용제품이 출시되지 않아, ETRI에서 개발한 모듈을 사용하였다. 그림 4는 시스템 개발에 사용한 개발 보드이다. 하단 중앙의 모듈이 AT-DMB 수신모듈이다.



[그림 4] 개발한 시스템 보드

4.1 DMB Framework 구현

AT-DMB는 지상파 DMB와 역호환성을 보장하며, 계층 변조를 통해 데이터 전송량을 증대시킨 차세대 모바일 방송 규격이다. AT-DMB 기본계층은 지상파 DMB 서비스를 지원하는 계층으로 지상파 DMB 채널을 의미하며, 지상파 DMB 단말기와 AT-DMB 단말기로 수신 가능하다. AT-DMB 향상계층은 지상파 DMB의 전송량 증대를 위해 추가한 계층으로 향상계층 신호는 AT-DMB 단말기뿐만 아니라 수신할 수 있다 [2]. 본 논문에서는 향상계층을 추가 서비스 채널로 사용하는 AT-DMB의 다채널 서비스[3]를 구현하였다.

AT-DMB의 비디오 규격은 H.264를, 오디오는 BSAC을 사용한다. S5PV210은 하드웨어 기반 코덱으로 H.264를 지원하기 때문에, 비디오는 프로세서에 제공하는 하드웨어 코덱을 사용한다. 또한, BSAC는 S5PV210에서 제공하지 않으므로 소프트웨어 기반의 코덱을 사용하였다. 비디오, 오디오는 별도의 라이브러리로 구현하였으며, 데이터 서비스를 위한 MOT도 DMB 프레임워크와 연결을 위한 인터페이스를 포함하여 라이브러리 형태로 구현하였다. 또한, AT-DMB의 Demux와 DMB Control도 DMB 프레임워크에 연결하였고, AT-DMB에서 향상계층과 기본계층의 정보를 제공하는 FIC Parser는 DMB Control에 구현하였다.

5. 실험 결과

보다 신뢰성 있는 시험환경을 구성하기 위해 다음과 같은 조건에서 실험을 진행하였다. AT-DMB 휴대용 송출기를 이용하여 AT-DMB 신호를 송출하고, 결과물로 송출되는 방송신호를 수신하여 송출된 스트림을 정상적으로 수신하는지를 시험하였다. 기본계층과 향상계층에 비디오 서비스와 데이터 서비스 각각 1개씩의 스트림이 송출되도록 하였다.

첫 번째 실험은 AT-DMB 기본계층과 향상계층에서 방송신호의 정상 수신여부를 실험하고, 두 번째는 지상파 DMB와 호환성 검증실험을 하였다. 첫 번째 실험 결과는 기본 계층의 비디오 서비스, 데이터 서비스 모두 정상 수신하였고,

향상 계층의 비디오, 데이터 모두 정상 수신하였다. 그림 5는 향상 계층의 비디오와 데이터 서비스를 동시에 수신하고 있는 화면이다.



[그림 5] AT-DMB 향상계층 수신 화면



[그림 6] 지상파 DMB 수신 화면

두 번째 실험은 지상파 DMB와의 호환성 실험이다. 그림 6은 송출되는 지상파 DMB 방송을 수신하고 있는 화면이다.

6. 결론

본 논문에서는 모바일 디바이스를 위한 개방형 플랫폼인 안드로이드 기반의 모바일 방송 수신 시스템을 개발하였다. 또한, 제안한 시스템을 AT-DMB 규격으로 설계 및 구현하였으며, AT-DMB 송출기를 활용하여 AT-DMB의 계층별 수신과 지상파 DMB와의 역호환성도 검증하였다. 제안한 시스템은 ATSC-MH, DVB 등 타 모바일 방송 규격의 모바일 방송 수신기 개발에 활용이 가능하며, AT-DMB, 지상파 DMB의 추가 서비스 개발을 위한 시험 플랫폼에도 활용될 수 있다.

참고문헌

- [1] Google Android, <http://developer.android.com/>
- [2] "Specification of the Advanced Terrestrial Digital Multimedia Broadcasting(AT-DMB) to mobile, portable and fixed receivers," TTAS.KO-07.0070, TTA, 2009.
- [3] 배병준, 김광용, 윤정일, 이지봉, 임종수 "차세대 지상파 DMB(AT-DMB) 기술," 한국방송공학회 학회지, 제14권, pp. 4~15. 2009년 3월.