

## ARM 프로세서를 이용한 PMP 구현

김희수, 한경호  
단국대학교 전기공학과

### Implementation of PMP using ARM processor

Hee-Su Kim, Kyoung-Ho Han  
Department of Electrical Eng. Dankook University

**Abstract** - 본 논문에서는 인텔사의 PXA 255 프로세서에 내장형 운영 시스템을 실장하여 미디어 플레이어의 기능을 구현하고, Wireless LAN(IEEE 802.11b)망을 통한 다양한 멀티미디어 콘텐츠를 다운로드할 수 있는 시스템 구현을 보였다. 이를 위하여 리눅스 운영 시스템에 미디어 플레이어와 코덱을 크로스 컴파일하여 보드에 실장하고, 사용자의 편의를 위하여 USB 디바이스 지원과 저장 용량 확대를 위한 Compact Flash 메모리 디바이스 지원을 보였다. 또한 사용자 인터페이스를 위하여 Qt/E를 이용하여 어플리케이션을 제작함으로써 사용자의 편의성을 도모하였다.

#### 1. 서 론

현재 주요 휴대형 멀티미디어 기기들은 차세대 PC로써 정보기기 기능을 가진 PDA, 휴대형 오디오 플레이어인 MP3 플레이어, 동영상 재생 및 네비게이션 등 멀티미디어 기능을 가진 PMP 등이 있다.

최근 출시되고 있는 PMP들은 동영상, MP3, 이미지 뷰어의 기능만을 가지고 있으며, 무선랜을 통한 웹접속이나 사용자 편의를 위한 USB 디바이스의 기능을 지원하지 않는다. 그러나 시간이 지날수록 대부분의 기기들은 원래 가진 기능 이외에도 인터넷을 비롯한 다양한 부가기능들을 그 안에 담게 될 것이고, 멀티기능을 통합적으로 제공하는 디바이스인 PMP를 중심으로 통합될 것으로 전망된다[1].

이에 본 논문에서는 최근 저 전력 고성능의 프로세서로 각광받고 있는 ARM 계열의 프로세서 보드와 모든 소스가 공개되어 있고, 안정적이며 기능이 검증되어 있는 리눅스 OS를 이용하여 개인용 멀티미디어 단말기 모듈 개발을 목적으로 하였다

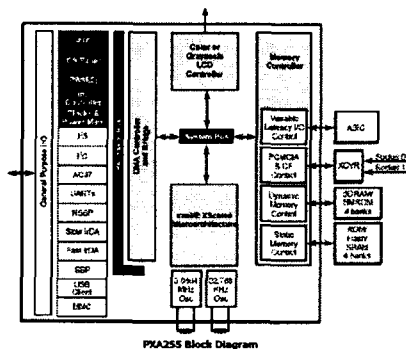
#### 2. 본 론

##### 2.1 구현 시스템의 특성

ARM 프로세서 보드는 고성능 프로세서인 Intel PXA255 Processor (400MHz)를 탑재한 보드로 OS는 Embedded Linux를 사용하였다. 이 보드는 인터페이스를 통한 자유로운 확장이 가능하며 PXA255 peripherals 대부분의 기능을 테스트 할 수 있다. 임베디드 리눅스는 FAT에서부터 NTFS에 이르기까지 다양한 파일 시스템을 지원하며 TCP/IP상에서 현존하는 거의 모든 프로토콜을 지원한다. 또한 마이크로소프트사에서 개발한 윈도우 운영 체제에 버금가는 하드웨어 호환성을 지니므로 임베디드 시스템에 적합한 운영체제이다[2].

##### 2.1.1 ARM9 프로세서의 특성

ARM의 가장 큰 특징은 다른 프로세서에 비해 전력을 훨씬 적게 소모한다는 것이다. 이런 이유 때문에, ARM 프로세서는 상대적으로 전력이 중요하게 여겨지는 휴대용 단말기에 많이 채택되고 있다. 더불어 다양한 장점들 때문에 모바일 영역뿐만 아니라 내장형 시스템 시장까지 급속히 확장되고 있다. 그림 1에 보인 Intel PXA255 프로세서(ARM)는 타겟에 맞는 다양한 peripheral들을 지원 한다



<그림 1> PXA255 Block Diagram

. 예를 들어 ISC core는 소형의 저전력, multimedia application 성능의 향상을 위해 lcd controller, multi-media controller, 외부 memory 인터페이스 등과 통합 가능하다. [3]

##### 2.1.2 운영체제 및 소프트웨어

임베디드 리눅스는 기본적으로 특정 임베디드 어플리케이션에 맞도록 리눅스 커널의 크기와 성능을 최소화, 최적화 시켜 만들어낸 커널을 의미하며 여기에 C 라이브러리나 유틸리티들은 필요에 따라 추가되거나 교체되어 하나의 시스템을 만드는 형태가 된다. 이런 커널의 최소화, 최적화에는 커널의 내부구조 변화도 가능하며 이를 극대화시키면 일반적인 사용 운영체제의 성능을 뛰어넘는 것도 가능하다. 결국 임베디드 리눅스는 개발자에게 시스템 개발에 있어서 금전적인 제약과 개발 환경에 적응 시간, 그리고 제품의 신뢰성에 대한 장벽을 넘을 수 있게 해주는 운영체제라 할 수 있다. 그리고 Qt는 Trolltech사에서 만든 Cross Platform 툴킷이며, Cross Platform에서 알 수 있듯이, 여러 운영체제에서 동일한 코드로 컴파일만 새로하면 동일한 결과를 얻을수 있어 개발 환경 시간을 단축 할 수가 있는 이점이 있다. 현재 많은 임베디드 리눅스 기반의 GUI로 사용되고 있다.

#### 2.2 PMP 구현

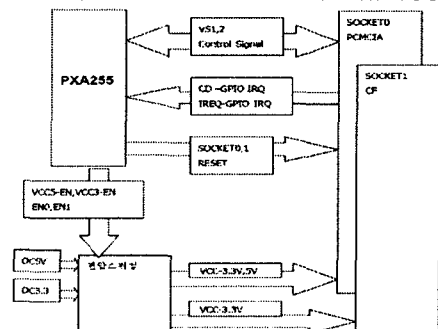
본 논문에서는 ARM 프로세서 보드와 임베디드 리눅스 OS를 기반으로 하여 호스트 PC에서 개발된 프로그램을 타겟보드로 적재할 수 있는 개발환경을 구축하고, 타겟보드에서 사용될 리눅스 미디어 플레이어인 mplayer를 구현하여 타겟보드로 적재됨을 보였다. 또한 무선 인터넷 환경이 구축되어 있는 곳에서는 멀티미디어 콘텐츠를 웹상에서 다운로드 할 수 있도록 Wireless LAN을 구현하여 보았다. 필요에 따라 추가적인 메모리 확장이 가능하도록 Compact Flash 메모리를 추가하여 저장 용량을 확대하였고, 다양한 USB 디바이스를 지원함으로써 사용자의 편리성을 높였다.

##### 2.2.1 호스트 PC에서 타겟보드로 mplayer 컴파일

타겟보드에 mplayer를 올리기 위해서는 동영상 플레이어와 코덱이 필요하다. 논문에서 사용된 코덱은 libmad-0.151b를, mplayer 소스는 mplayer\_0.90rc1-arm을 사용하였다. mplayer를 설치하기 위해서는 소스를 직접 크로스 컴파일 하는 방법과, ARM용으로 나온 바이너리를 tarpt보드에 복사하는 두 가지 방법이 있다. 바이너리 설치 방법은 해당 파일을 복사만 해주면 되나, 실행에 있어서 문제점을 가지고 있다. 따라서 코덱은 바이너리를 통해서 설치하고 mplayer는 소스를 크로스 컴파일 하여 타겟보드에 적재하였다.

##### 2.2.2 호스트 PC와 타겟보드의 무선랜 설정

무선 인터넷은 우리가 접해왔던 유선 네트워크 개념을 무선 매체를 통하여 확장한 인터넷 서비스 제반 기반 및 운용 시스템들을 말한다. 무선 LAN은 근거리 네트워크 내 호스트들간의 연결을 유선이 아닌 무선으로 구성한 것이다. 이는 기존의 유선 LAN에서 발생했던 비용, 확장성, 이동성 등의 문제들을 해결하였다.[4]



<그림 2> 무선랜을 사용하기 위한 커널 재구성

무선랜 카드를 시스템에서 사용하기 위해서는 커널을 그림 2와 같이 재구성하여야 한다. 커널의 menuconfig에서 사용하고자 하는 device에 대한 세부설정을 하고 디바이스 파일들을 커널에 포함시키면, system init에서 cardmgr를 실행시켜준다. 따라서 별도의 초기화없이 카드를 삽입함으로써 자동인식이 된다.

### 2.2.3 Compact Flash 메모리 포팅

미국의 SanDisk사가 개발했으며 널리 사용되는 메모리 규격중에서 큰 편에 속한다. 36\*43\*3.3mm 크기 때문에 MP3 플레이어와 같은 초소형 기기에는 어울리지 않으며 PMP, 디지털 카메라, 노트북용 저장 장치 등 고용량이 요구되는 곳에 주로 사용되고 있다. CF 메모리를 사용하기 위해서는 해당 디바이스 드라이버와 커널을 시스템에 맞게 변경 시켜야 한다. 커널의 환경설정은 아래와 같이 한다.

```

General setup -->
  [*] Support hot-pluggable devices
PCMCIA/CardBus support -->
  [*] PCMCIA/CardBus support
  [*] PXA250/210 support
  
```

<그림 3> Compact Flash 메모리를 위한 커널 설정

커널을 재 컴파일 후 시스템에 다시 적재시킨후 리눅스의 mount 명령어를 사용하여 CF 메모리 디바이스를 인식할 수 있다.

### 2.2.4 USB 인터페이스 구현

USB란 컴퓨터와 외부로 연결되는 주변 기기들의 연결 장치를 일정한 규격으로 통일해 사용하는 새로운 형태의 연결 장치이다. USB 시스템은 크게 3부분으로 구성된다. 먼저 호스트(Host)가 있어야 된다. 이는 USB 시스템의 root가 되며 임베디드 시스템에서는 시스템 본체가 호스트가 된다. 두 번째로 허브(Hub)라는 장치가 필요하다. 이는 Network에서 사용되는 허브와 동일한 개념이다. 마지막으로 USB 주변 장치가 필요하다. 이는 키보드, 마우스, 스캐너, 프린터, 모뎀, CD-ROM 드라이브, 조이스틱 등이 있다. 보통 사용하는 USB 디바이스라는 용어는 허브와 USB 주변 장치를 이르는 것이다. USB 시스템은 Hardware, Firmware, Software의 복합으로 구성된다. 호스트쪽에서는 커널이 USB가 동작을 잘하도록 칩셋을 초기화 해야 하고, 호스트의 운영체제에서는 각 디바이스의 USB 드라이버가 잘 동작해야 한다. 한편 허브쪽에서는 내부 controller와 repeater, 기타 여러 요소들이 사용하는 프로토콜과 전송형태(eTDM)에 맞게 잘 동작되도록 hardware와 firmware가 구비되어야 한다. 본 논문에서 구현한 시스템은 USB를 사용하기 위해서 Hybus사의 USB.tar 파일을 사용하였다. 이 파일은 USB 드라이버 Module과 Source로 구성되어 있다. 해당 파일들을 커널에 위치시키고 커널의 환경 설정을 변경한다. 커널의 환경설정은 아래와 같이 한다.

```

Menuconfig
USB Support -->
  (<*) Support for USB
    (<*) USB Mass Storage support
    (<*) USB Human Interface Device support
SCSI support -->
  (<*) SCSI support?
  (<*) SCSI disk support
Input core support -->
  (<*) Keyboard support
  (<*) Mouse support
  
```

<그림 4> USB 인터페이스를 위한 커널 설정

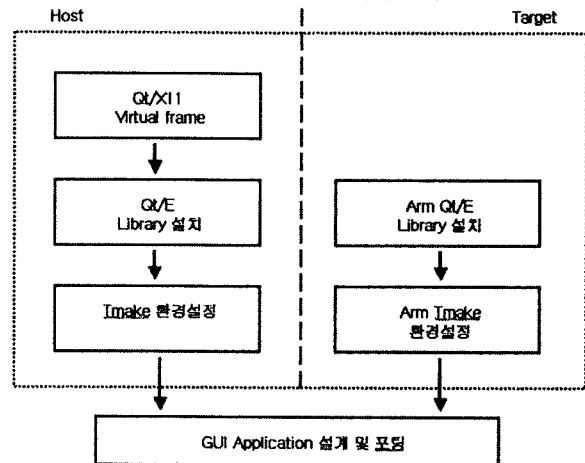
커널을 재 컴파일 후 시스템에 다시 적재시키면 USB 드라이버가 정상적으로 커널에 추가되었다는 메시지를 볼 수 있다.

### 2.2.5 QT/E를 이용한 사용자 인터페이스 구현

QT/Embedded(QT/E)란 Trolltech사에서 개발한 QT 라이브러리를 기본으로 하여 개발된 임베디드 시스템용 GUI(Graphic User Interface) 라이브러리(Library)이다. X-Window 없이 커널에서 제공하고 있는 프레임 버퍼를 이용하여 그래픽 장치에 접근을 하는 방식을 사용하고 있다. 이와 같은 접근 방식은 QT 라이브러리와 소스 호환성을 가지고 있으므로 KDE 프로젝트의 많은 응용 프로그램들을 QT/E를 사용하여 임베디드용으로 바꿀 수 있다. [5]

Qt/E 프로그래밍 환경은 개발용 리눅스 PC 환경과 타겟 보드 개발환경, 2가지 환경으로 구성된다. 먼저 타겟보드에서 Qt/E 프로그램을 하기 전에 개발용 리눅스 PC환경에서 개발자가 설계한 GUI 관련 프로그램을 Qt/E와 연관하여 수행 시킴으로서 설계된 GUI 프로그램을 확신 할 수 있다. Qt/E에서는 리눅스 PC에서 Qt/E 가상 프레임 버퍼의 Qt/E 프로그램 시뮬레이션 환경을 제공한다. 그러나

최종적으로 타겟 보드의 6.5인치 TFT칼라 LCD에 디스플레이 하고 터치 스크린 입력을 받기 위해서는 타겟 보드의 CPU 및 관련 Qt/E 환경에 적합한 개발 환경이 요구된다. 리눅스에서 Qt/E GUI 개발 환경 설치하는 그림 3과 같은 순서를 따라야 한다.



<그림 5> Qt/E GUI 개발환경 설치

즉 호스트 컴퓨터에서 시뮬레이션을 하기 위해 Qt/X11 Virtual framebuffer를 설치하고 Qt/E 라이브러리를 설치한다. 다음으로 메이크 파일 생성하기 위한 툴인 Tmake를 설치하면 된다.

## 3. 결 론

본 논문에서는 ARM 프로세서 보드에 임베디드 리눅스 OS를 실장하여 무선 기반의 PMP 시스템을 구현하였다. 구현된 시스템은 상대적으로 전력의 중요하게 여겨지는 휴대용 단말기에 많이 채택되고 있는 ARM 프로세서를 사용하였고, 운영 시스템은 지속적인 개발과 유지보수를 위한 기술지원 비용이 다른 운영 시스템보다 저렴하며, 소프트웨어의 확장성에 관한 시장의 요구를 경쟁력있게 만족시키는 개방적인 운영체제인 임베디드 리눅스를 사용하였다.

리눅스 미디어 플레이어를 타겟시스템에 맞도록 크로스 컴파일하여 보드 실장하였으며, 무선 랜망을 통한 다양한 멀티미디어 컨텐츠를 다운로드할 수 있는 시스템을 구현해 보았다. 또한 사용자 인터페이스를 위하여 Qt/E를 이용하여 응용 프로그램을 구현하였으며, 구현된 시스템에 다양한 응용프로그램을 호스트 PC에서 컴파일하여 포팅할 수 있음을 보였다. 휴대용 장비의 추가적인 저장장치의 문제를 해결하기 위하여 크기는 다른 메모리보다 크지만 고용량으로 사용할 수 있는 Compact Flash 메모리를 추가장착할 수 있고, 사용자의 추가적인 디바이스 확장을 위하여 USB 인터페이스를 구현해 봄으로써 차후 다양기기들과 통신을 할 수 있도록 구현하였다.

리눅스 운영 시스템은 소프트웨어를 계층화하여 운영 시스템을 사용하는 시스템이므로 개발 후 소프트웨어의 업그레이드도 손쉽게 할 수 있으며 더 나아가 리눅스 운영 시스템의 이점을 살려 GUI 응용 프로그램 개발로 사용자에게 더 쉬운 인터페이스를 제공할 수 있을 것이라 예상된다.

### [참 고 문 헌]

- [1] 이은민, "PMP의 특징 및 시장전망", 정보통신정책, 17권, pp2, 2005년
- [2] 박재호, "임베디드 리눅스", pp65, 2003년
- [3] (주)하이비스, "X-Hyper255B-TKUi Developer's Manual", pp8, 2003년
- [4] 김한수, "무선 인터넷 기술을 이용한 리눅스 시스템 원격관리", 호남대학교 정보통신 연구소 논문집, 11권, pp207, 2003년
- [5] (주)한백전자, "임베디드 리눅스 시스템 HBE-EMPOSI", pp1165, 2004년