

# 임베디드 리눅스 기반의 USB 네트워크 드라이버 모듈 설계 및 구현에 관한 연구<sup>1)</sup>

이상훈<sup>o</sup> 문승진

수원대학교 정보공학대학 컴퓨터학과

{vkfka1<sup>o</sup>, sjmoon}@mail.suwon.ac.kr

## A Study on Design and Implementation of a USB Network Driver Module based on Embedded Linux

Sang-Hoon Lee<sup>o</sup> Seung-Jin Moon

Dept. of Computer Science, The University of Suwon

### 요 약

임베디드 시스템의 발전으로 인하여 시스템에서 USB Network의 필요성 또한 빠르게 증가되어가고 있다. 본 논문에서는 임베디드 리눅스 기반의 단말기와 PC 사이에 USB 통신을 구현기 위해서 XScale 마이크로 프로세스 레퍼런스 보드에 리눅스 커널을 포팅하고 포팅 된 플랫폼을 기반으로 USB Network의 설계 및 구현에 관하여 논하도록 하였다.

### 1. 서 론

임베디드 시스템은 컴퓨터 하드웨어, 소프트웨어를 총칭하는 개념으로 범용이 아닌 특수용 시스템이라고 볼 수 있다.[1] 즉 미리 정해진 특정 기능을 수행하기 위해 만들어진 하드웨어와 소프트웨어를 함께 부르는 말이다. 즉 특별한 하드웨어를 제어하는 시스템으로, 다른 시스템에 의존하지 않고 내부에 마이크로프로세서가 있어 특정 작업과 기능만을 수행하도록 설계 되어 있는 시스템을 말한다.

초기 임베디드 시스템은 그 구성이 매우 단순하여서 운영체제 없이 동작하는 응용소프트웨어 정도의 사용으로도 여러 용도로 사용할 수 있었지만, 32bit 이상의 계산능력이 강화된 마이크로프로세서의 발전으로 인해서 사용 영역이 넓어지게 되었으며, 시스템 규모가 커짐에 따라 운영체제를 기반으로 하는 임베디드 시스템 구성이 일반화 되었다.

본 논문에서는 Intel Xscale PXA-255 Processor 을 사용하여 임베디드 리눅스 기반의 USB Network 를 구현하고자 한다

### 2. 관련 연구

#### 2.1 임베디드 리눅스

현재 개발된 임베디드 운영체제로는 Window CE, pSoS, VxWorks, uC/OS, QNX 등이 있지만 이러한 상용 운영체제들은 사이즈가 크고 커널이 재구성 되기가 어려우며 로열티가 비싼 편이다. 이에 임베디드 리눅스의 경우 커널을 개발 목적에 맞게 최적화 과정을 거쳐 크기를 줄일수 있으며, 공개된 운영체제로서 비용을 절감할 수 있는 장점이 있다.

이에 본 논문에서는 Embedded Linux를 본 연구의 기반 운영체제로 선택하였다.

#### 2.2 USB 하드웨어 및 소프트웨어

USB는 패킷 단위로 데이터를 주고받는다. 임베디드 시스템을 PC에 연결하려면, PC의 메인보드에는 USB 호스트 컨트롤러가 들어있어야 하고, 이에 대응해 임베디드 시스템에는 USB칩이 들어있어야 한다.

PC의 메인보드에서 사용되는 두 종류의 USB 호스트 컨트롤러는 UHCI (Universal Host Controller Interface) 와 OHCI(Open Host Controller Interface) 이며 두 종류의 호스트 컨트롤러가 제공하는 기능은 거의 비슷하다

임베디드 시스템에 들어가는 USB 칩은 필립스 등 많은 회사에서 개발하고 있다. 마이크로 컨트롤러에 USB 칩 기능을 추가한 컨트롤러도 선보이고 있다.

그 외에 PC와 임베디드 시스템을 연결할 USB 커넥트가 필요하다. 공식적으로 USB 커넥터는 A, B 두 종류가 있다.

1) 본 연구는 (주) 이디지콤와 제 12차 산학연전소사업 공동연구의 일부를 포함함

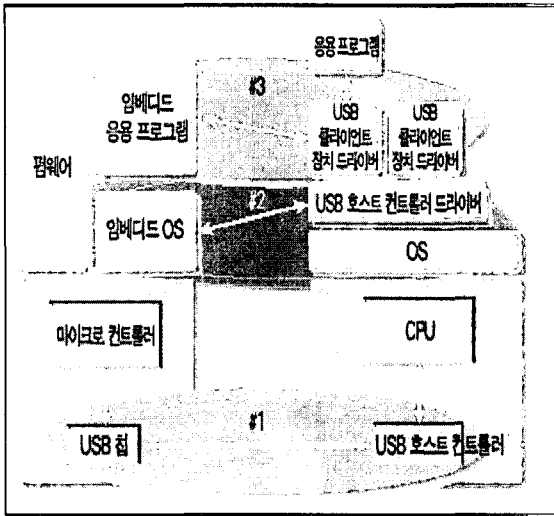


그림 1 USB 구조도

#1에서 물리적인 데이터 교환과 주고받는 데이터에 어려가 있는지 감지하며, 대부분 하드웨어 로직으로 처리된다. 호스트 컨트롤러부터 USB칩이 특정 데이터 처리 요청을 받으면, 인터럽트를 통해 마이크로 컨트롤러에게 알린다. 컨트롤러가 적절한 데이터를 메모리로부터 읽어와 USB칩 쪽으로 보내면 하드웨어 통신이 이루어진다. #2에서 임베디드 시스템이 USB를 통해 PC에 추가되어 물리적으로 통신 할 수 있게 되면, 임베디드 시스템은 USB 호스트 컨트롤러 드라이버와 여러 가지 정보를 교환한다. 이 때 교환하는 데이터에는 디스크립터도 포함되어 있다. 좀더 자세하게 말하면 핫드 드라이버가 보내는 셋업 패킷에 따라서 적절한 임베디드 시스템에 관한 데이터를 보내야 한다. 이 때 보내는 데이터를 디스크립터라 한다. 호스트 컨트롤러 드라이버는 받은 디스크립터를 모든 클라이언트 장치 드라이버에 전파하며, 0이 아닌 값을 리턴하는 클라이언트 장치 드라이버를 해당 임베디드 시스템의 드라이버로 사용한다. #3에서 PC의 응용 프로그램이 해당 임베디드 시스템을 사용하려고 할 때, 클라이언트 드라이버에 처리를 요청하면 하드웨어를 거쳐 마지막으로 임베디드 시스템의 펌웨어가 데이터를 받아 적절하게 처리한다. [2]

3. 임베디드 리눅스 기반의 USB 네트워크 모듈 설계 및 구현

3.1 개발 환경

3.1.1 개발 환경 및 도구

임베디드 보드는 하이버스(주) [3] 의 X-Hyper 255B 를 기반으로 다음과 같은 환경과 도구들을 사용하였다.

- RedHat Linux version 9.0
- linux-2.4.20.tar.gz
- GNU tools version 2.96 ( cross compiler )
- USB Slave Device Driver
- 

3.1.2 개발환경 구성도

개발 환경으로 타겟보드 보드와 호스트간의 Jtag 케이블 ,Ethernet cross cable, 시리얼 케이블로 연결되어진다.

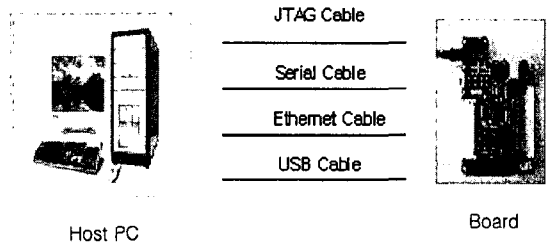


그림 2 개발환경 구성도

3.2 USB 네트워크 모듈 설계

Host PC의 리눅스 환경에서는 board와 usb network 를 할 수 있도록 usb driver를 board에 맞게 소스를 수정하고 커널 컴파일시에 usb network로 컴파일 옵션을 변경 시켜준다.

board에는 리눅스 소스를 arm용 및 그 board용으로 패치하고 JTag cable를 이용해 Bootloader를 올리고 ethernet을 통해서 패치한 커널 이미지를 board에 올린다.

Host PC와 board 각각 환경이 갖추어지면 minicom으로 board에 usbf에 ip를 할당해 준다.

그후 usb cable을 연결후 Host PC에 모듈이 설정되어지면, usb0에 ip를 할당하고 ping으로 통신이 이루어지는지 확인한다.

3.3 USB 네트워크 모듈 구현

Host PC의 커널 컴파일 전 usbnet.c 파일의 소스를 보드에 맞게 변경하여 주고, 컴파일 한 후 Host PC와 board 의 환경이 갖추어지면 board에 ip를 할당하고 Host PC의 콘솔 창에서 log 파일을 검사하고, 시스템의 network를 ping으로 확인한다.

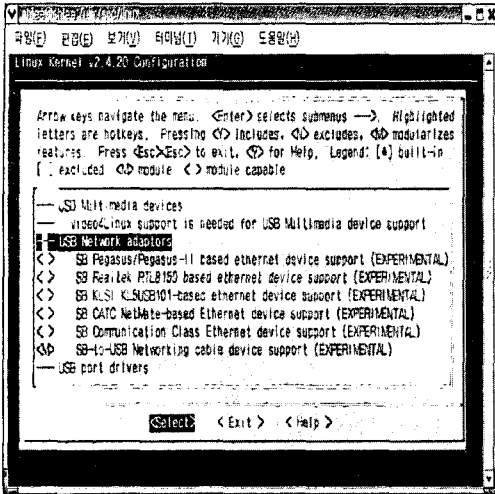


그림 3 Host PC 커널 컴파일

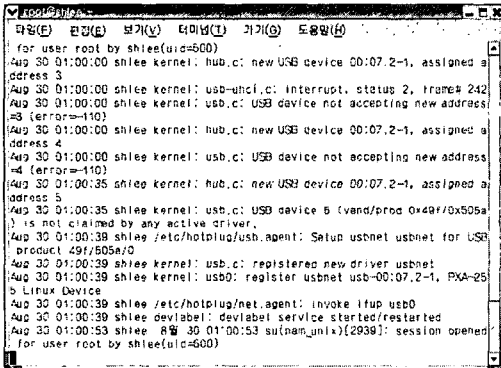


그림 4 USB 드라이버 연결

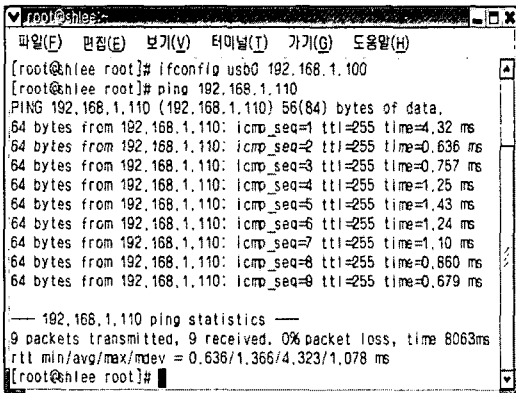


그림 5 USB Network

그림 3과 같이 Host PC의 커널을 board와 Usb 통신을 할 수 있도록 컴파일 하여주고, 그 전에 usb driver를 xscale에 맞게 변경하여 준다. 컴파일 후 Host PC에 driver 모듈이 생성이 되면 Host PC와 board 간에 통신을 할 수 있는 환경이 갖추어 진다. 그림 5와 같이 드라이버 연결이 확인되면 그림 4에서 Host PC와 board 사이에 network 연결 상태를 볼 수 있다.

#### 4. 결론 및 향후 연구과제

본 논문의 usb network 설계와 구현은 리눅스를 운영체제로 하는 PC와 xscale용 커널이 포팅된 board 사이에 usb로 통신이 되는 기능을 가지고 있다. 현재 board의 usb칩이 slave로 되어 있어, 단지 PC와의 통신으로 이루어져 있지만, 후에 master의 기능을 추가해 직접 board에서 다른 usb slave 장치와 연결하는 방법을 개발중이다

향후 제안된 모듈은 USB로 연결 가능한 임베디드 장비에 추가되어 휴대용 단말장비의 확대 발전이 기대되며 특히 데이터 전송의 속도를 이용하여 더 빠른 데이터 전송이 필요한 영상진단 장비로의 확대 발전으로 그 가치가 더욱 높아질 것으로 기대된다.

#### 참고문헌

- [1] 박재호, "임베디드 리눅스", 한빛미디어
- [2] 김병건, 오지성 "리눅스 USB 드라이버개발" 마이크로소프트웨어 2002. 6
- [3] 하이버스 (주) <http://www.hybus.net/>
- [4] USB, <http://www.usb.org>
- [5] Linux USB, <http://www.linux-usb.org/>
- [6] KESL, <http://www.kesl.org>
- [7] <http://www.arm.linux.org.uk/>
- [8] KLDP, <http://www.kldp.org>