

무선 인터넷 망에서 임베디드 리눅스 기반 PDA를 이용한 영상보드 원격 제어 시스템 구현*

김성용** · 이상민***

〈 목 차 〉

I. 서론	3.2 영상뷰어를 이용한 영상송신보드 모니터링 및 제어
II. 임베디드 시스템	3.3 PDA를 이용한 스텝 모터 제어
2.1 임베디드 시스템	IV. 실험 결과 및 고찰
2.2 임베디드 리눅스 시스템 개발 환경	V. 결론
III. PDA를 이용한 영상보드의 원격 제어	참고문헌
3.1 임베디드 리눅스 기반 PDA 영상 스트리밍	Abstract

I. 서론

현대사회는 무선 인터넷 환경과 모바일 컴퓨팅 기술을 기반으로 한 유비쿼터스 시대를 맞이하고 있다(이동만, 2006; 차윤숙, 2007). 모바일 컴퓨팅기술, 무선인식기술과 임베디드 시스템 등이 유비쿼터스 컴퓨팅 환경을 구현하기 위한 핵심 기술로 활용되고 있다. 임베디드 시스템은 우리가 생활하는 곳곳에 자리잡고 있으며, 이미 유비쿼터스 시대를 맞이하고 있다. 임베디드 기술을 활용한 제품으로 유·무선 통신, 홈 네트워크, 휴대폰, PDA(personal digital assistant),

PMP(portable multimedia player), 네비게이션, 텔레마틱스 등이 있으며, 이들은 가정에서 많은 부분을 차지하고 있다(나종화 외, 2003; 박태규, 2004; 이연조, 2002; <http://embedded.com>; 이승현, 2006). 지금까지 고가의 장비로만 인식된 로봇 시스템들이 공장 및 의료 장비, 군 장비와 이동 로봇, 동물 로봇, 지능형 인공 청소 로봇 등의 다양한 형태로 사용되어지고 있다.

또한 인터넷을 통해 거리상의 제약 없이 원격에서 시스템을 제어하기 위한 연구가 많이 이루어지고 있다(강진범 외, 2003; 김성호 외, 2004; 김성용 외, 2005). 이와 같은 원격 제어 시

* 이 연구결과물은 2006학년도 위덕대학교 학술진흥연구비 지원에 의하여 이루어졌음.

** 위덕대학교 컴퓨터공학부 부교수, sykim@uu.ac.kr

*** 위덕대학교 컴퓨터공학과 박사과정, hunteri007i@daum.net

스텝은 공장 자동화, 군사 시스템, 우주 탐사, 홈 네트워크 그리고 의료 장비 등과 같은 광범위한 분야에서 쓰이고 있다. 이와 관련된 연구는 Mercury Project, Tele Garden, The Mechanical Gaze system 등이 있다(Luo & Kuo, 2003). 물론 인터넷을 통해 PDA를 이용한 연구도 꾸준히 진행되고 있지만 WinCE기반으로 연구되거나 임베디드 리눅스를 사용하면서 캡처된 이미지를 전송하는 정도의 연구만 진행되고 있다(한중희, 박태규 등, 2004; 김지훈, 2006)).

본 논문에서는 영상을 획득할 수 있는 영상 송신보드를 이동할 수 있도록 스텝 모터에 탑재시키고 무선 인터넷을 이용한 임베디드 리눅스 기반의 PDA를 통하여 영상송신보드를 원격 제어하고, 획득한 영상을 효과적으로 스트리밍하는 방법을 각각 제안하였다. 이를 위해 첫째, 임베디드 리눅스가 탑재된 PDA에서 리눅스 기반의 영상플레이어를 활용해 영상송신서버에서 영상을 획득하여 스트리밍하는 모듈을 개발하였다. 둘째, 퍼스널 컴퓨터에서 동작하는 뷰어가 영상송신보드에게 TCP/IP 프로토콜을 통해 영상보드의 프로세스를 제어 하는 기능과 각종 설정을 변경할 수 있는 모듈을 구현하였다. 그리고 셋째, 무선 네트워크를 통해 PDA가 방향을 제어하는 신호를 임베디드 리눅스가 탑재된 보드로 전달되고, 보드의 확장 I/O 포트를 통해 스텝 모터가 제어되는 모듈을 개발하였다.

II. 임베디드 시스템

2.1 임베디드 시스템

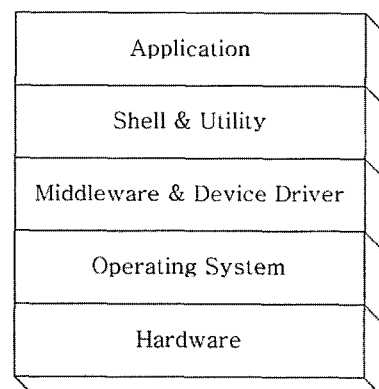
임베디드 시스템의 일반적인 의미로는 마이

크로프로세서 혹은 마이크로 컨트롤러를 내장하여 원래 제작자가 지정한 기능만을 수행하는 장치로 정의하고 있다. 임베디드 리눅스는 저성능의 프로세서와 소용량의 메모리를 가진 임베디드 시스템용으로 개발된 리눅스이다. 따라서 임베디드 시스템의 특징은 작은 크기와 경량화, 저전력, 복잡한 개발환경, 재구성 가능과 Operation /QoS 지원 등을 들 수 있다.

한편 실시간 시스템에서의 운영체제는 간단하고 단순한 순차적인 작업에 관련하고, 순차적인 프로그램으로 충분한 8bit, 16bit 마이크로프로세서 및 마이크로 컨트롤러를 사용하였다.

그러나 임베디드 시스템에서의 운영체제는 시스템 자체가 실시간 시스템에 비해 대형화되고 32bit RISC 마이크로프로세서를 일반적으로 사용하며, 네트워크 연결 및 GUI(graphical user interface)환경이 강화되었다. 또한 기능이 많아지고 복잡해짐에 따라 순차적인 프로그램 작성이 불가능하게 되어 WinCE, 임베디드 자바, 임베디드 리눅스 등의 운영체제가 도입되는 계기가 되었다.

임베디드 시스템 구조는 <그림 1>과 같고, 운영체제는 선점형 스케줄링과 세밀한 스케줄 단



<그림 1> 임베디드 시스템의 구조

위를 지원을 해야 하며, 디바이스 드라이버는 I/O 시스템과 연동될 수 있어야 한다.

2.2 임베디드 리눅스 시스템 개발 환경

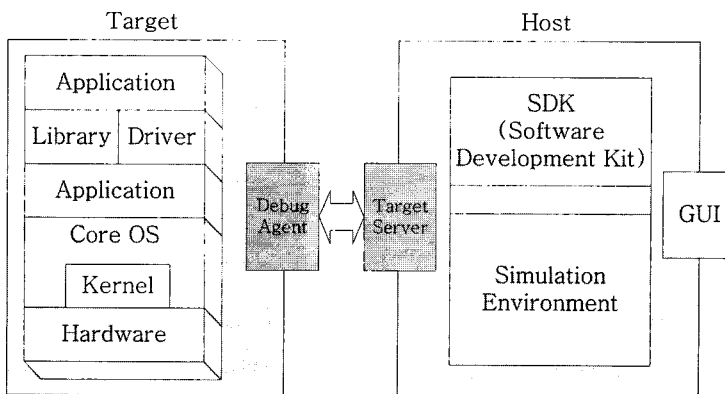
일반적인 임베디드 리눅스 시스템의 개발 순서는 다음과 같다.

- ① 기능과 용도에 따른 전체적인 시스템 설계
- ② 하드웨어 설계 및 구현
- ③ 개발 환경 구축
- ④ 리눅스 커널 포팅
- ⑤ 모니터 프로그램 포팅 및 구현
- ⑥ 디바이스 드라이버 포팅 및 구현
- ⑦ 응용 프로그램 개발

여기서 ①과 ②의 하드웨어가 설계 및 구축된 뒤 개발 환경 구축, 리눅스 커널 조정, 디바이스 드라이버 조정을 한 후 응용 프로그램 개발 등의 절차가 일반적이다. 하지만 임베디드 리눅스 시스템에서 사용되는 CPU와 개발되어지는 퍼스널 컴퓨터의 CPU는 다르기 때문에 하드웨어 설계가 이루어지면 개발되어지는 퍼스널 컴퓨터의 개발 환경을 구축해야 한다. 개발

환경에 따른 개념도는 <그림 2>와 같다.

또한 타겟(임베디드 리눅스가 탑재된 개발 보드)과 호스트는 여러 통신을 하면서 개발되는데 주로 세 가지의 통신이 이루어진다. 즉 JTAG, 직렬 포트와 LAN을 사용하며 이들의 사용 용도는 각각 다르다. 첫째, JTAG은 부트로더를 포팅하기 위해 사용된다. 둘째, 직렬 포트는 호스트와 타겟의 정보 교환 및 상태 파악을 위한 연결로서 타겟에 없는 콘솔을 호스트를 통해 사용할 수 있게 한다. 또한 호스트를 이용하여 타겟에 명령을 수행하며 그 타겟 보드의 수행 결과를 호스트에서 모니터할 수 있게 한다. 셋째, LAN은 TFTP 등과 같은 프로토콜을 이용하여 고속의 데이터 통신이 가능하게 하고 커널이나 파일 시스템 이미지 등 대량의 데이터를 전송하는데 사용한다. 그리고 NFS(network file system)를 사용하여 개발된 프로그램을 타겟에 직접 포팅하지 않고, 디렉터리를 공유함으로써 호스트의 저장된 개발 프로그램을 타겟에서 바로 수행할 수 있어 응용 소프트웨어의 개발을 용이하게 한다.



<그림 2> 임베디드 시스템 개발 환경

Ⅲ. PDA를 이용한 영상보드의 원격 제어

임베디드 리눅스가 포팅되어 있는 PDA를 사용하여 영상송신보드를 원격으로 제어하고, 영상송신보드를 탑재할 수 있는 스텝 모터를 설계하고 영상송신보드의 상태 및 스텝 모터를 PDA로 제어할 수 있는 시스템을 구축하였다. 즉, 본 논문에서는 각 기능을 크게 영상 획득, 프로세스 제어, 스텝 모터 제어 세 가지로 구분 할 수 있으며, 이러한 각각의 기능을 무선 인터넷망에서 수행할 수 있는 임베디드 리눅스 시스템을 제안하였다.

3.1 임베디드 리눅스 기반 PDA 영상 스트리밍

PDA는 주기억 장치인 RAM과 보조기억장치인 플래시 메모리는 하드 디스크와 같은 보조기억장치의 역할을 하면서 동시에 ROM과 같은 주기억장치의 역할도 한다. 그러나 영상을 전송하는 서버에서 전송되는 영상 크기와 네트워크 상태에 따라 PDA에서 영상을 처리하기란 쉽지 않다. 이러한 멀티미디어 데이터처럼 부피가 큰 데이터를 처리하기 위해서는 데이터를 저장

할 수 있는 공간이 필요하고, 저장된 데이터를 처리하는 속도가 매우 중요한 요소가 된다. 이를 해결하기 위해 무선 네트워크를 이용하여 임베디드 리눅스 PDA에서의 영상 스트리밍을 제안하였다. 즉, 영상 전송 서버를 리눅스 서버로 구축하고 AP(access point)를 통하여 PDA에게 IP를 할당한다. IP를 할당 받은 PDA에서는 NFS를 사용한 서버와 연결한 후 IEEE802.11b표준을 지원하고 11Mbps의 데이터 전송 속도를 유지하는 네트워크를 통해 PDA로 각종 동영상을 재생할 수 있는 모듈을 제안한다.

본 논문에서는 Compaq사의 iPAQ h3870모델의 PDA을 사용하였고, 이 모델은 WinCE를 기반으로한 Microsoft Pocket PC는 멀티미디어와 네트워크 등의 기능들을 제공하고 있다.

임베디드 리눅스 기반 PDA 영상 스트리밍을 수행하기 위해 임베디드 리눅스를 PDA에 포팅하기 위한 방법으로 StrongARM에서 지원하는 Jtag포트를 이용하여 bootloader를 적재하는 방법이 있고, 직렬 포트를 사용하는 방법, 외부 저장 공간, PPP 등의 여러 방법이 있다. 본 논문에서는 기본적으로 WinCE기반이기 때문에 WinCE기반으로 만들어진 BootBlaster-1.19를 설치하기 위해 Microsoft사에서 출시된 ActiveSync 프로그램으로 이를 PDA로 전송하고 실행시킨

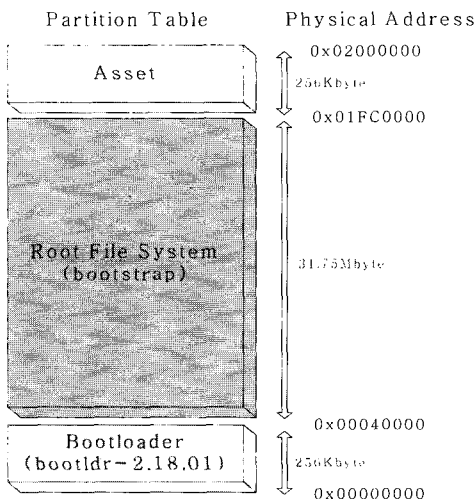
<표 1> PDA와 PC간의 터미널 통신 설정

Properties	Value
Bit/sec	115200
Data bit	8
Parity	none
Stop bit	1
Flow control	none

후 BootBlaster를 이용해서 bootloader인 bootldr-2.18.01을 PDA에 설치하였다. 다음으로 jffs2 파일 시스템을 지원하는 bootstrap을 RS-232C를 사용하여 호스트 PC와 연결시키고 <표 1>과 같이 포트를 설정 후 bootldr에서 지원하고 있는 각종 명령어중 load [partition]을 이용하여 커널을 포팅하였다.

bootstrap은 리눅스의 기본적인 명령어들과 제공되고 있는 서비스들을 모두 포함 시켜 놓은 이미지 파일로서 개발자들을 위해 널리 배포되고 있는 패키지의 일종이다.

이와 같은 bootstrap이 모두 PDA상에 적재 되었다면 총 세 개의 파티션으로 Flash ROM이 <그림 3>과 같이 구성된다.



<그림 3> Flash ROM의 메모리 맵

Bootloader 부분은 Boot Parameter Partition 을 포함하고 있으며, parameter는 부팅 시 필요한 설정 값들을 모아 놓은 집합이 된다. 만약 parameter partition이 NULL이면 /boot/params 파

일을 참조하여 커널이 로딩하게 된다.

Root File System은 JFFS2(journaling flash file system version two)를 사용하였으며, 여기에는 kernel을 포함한 각종 설정 값 및 사용자 어플리케이션이 모두 포함되고 있다. 본 논문에서 사용하는 커널버전은 2.4.19-rmk6-pxa1 이다.

Asset 부분은 최초 WinCE가 설치된 PDA를 사용하였기 때문에 리눅스 커널을 사용하다가 WinCE로 복구하기 위해서 주요 정보를 저장해 놓은 공간이 된다.

무선 네트워크를 이용하기 위해서는 PDA에 CF Type 무선 네트워크 카드를 장착하여 네트워크를 구성해야 된다. 물론 선행되어야 할 작업으로 AP의 설정과 서버의 구성이 이루어져야 하는데 이러한 AP 설정 및 리눅스서버 구축은 생략한다. 먼저 무선 네트워크에 사용되는 이더넷카드는 Buffalo WLI-CF-S11G를 사용하였다.

PDA에 무선 이더넷카드를 인식시키기 위해 linux-wlan-ng_0.2.0-2.4.19-rmk6-arm을 PDA에 직렬 포트를 이용하여 전송시킨다.

디바이스 드라이버 설치가 모두 완료되면 /etc/pcmcia/hostap_cs.conf 파일, /etc/pcmcia/wireless 파일과 /etc/pcmcia/wireless.opt 파일이 생성된다.

이 세 가지 파일의 내용을 생성, 추가 설정하고 파일을 수정한 후 service 명령어로 네트워크를 재실행 시켜 wlan0 디바이스를 인식하고 AP를 통하여 IP를 할당 받도록 하였다(김성용 외, 2005).

물론 AP는 정상적으로 작동해야 하며, 본 논문에서 사용된 AP장비는 Buffalo-WAL-L11G를 사용하였다(http://www.buffalotech.co.kr/down/access.html). AP 설정은 정상적으로 AP에게 전원을 공급하고 퍼스널 컴퓨터상의 AP

를 설정할 수 있는 air station 응용 프로그램을 설치한 후 AP를 설정하였다. 여기서 AP는 하나의 IP를 가지고 있어야 한다.

영상전송 서버에서 PDA까지 네트워크 구성은 <그림 4>와 같다. 영상전송서버와 PDA사이의 무선 네트워크가 가능하게 된다면 스트리밍할 수 있는 재생기가 필요하다. 동영상 재생하는 플레이어들은 상당히 많이 존재하고 그 활용도 면에서도 상당하다. 하지만 리눅스 OS 기반에서 사용되고 있는 플레이어는 Window OS 기반보다 상당히 미약하다는 것이 사실이다.

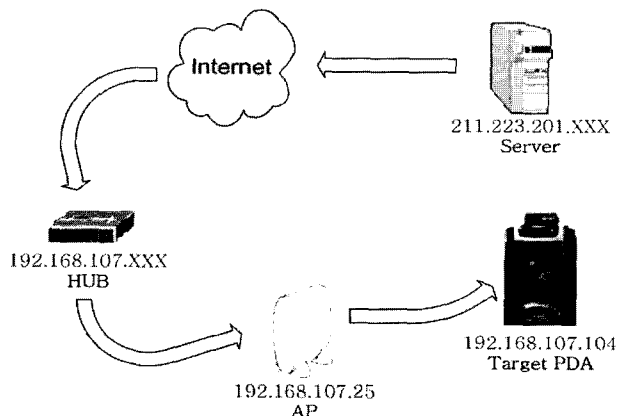
리눅스에서 많이 사용되고 있는 플레이어를 들자면 MPlayer, Real, Xine 등이 있다 (<http://www.mplayerhq.hu/homepage/design7/dload.html>; ISO/IEC 1998). 본 논문에서 사용한 플레이어는 MPlayer-1.0 버전을 사용하였으며, 코덱 또한 MPlayer를 다운로드 받으면 대부분의 정보가 포함되어 있다. 이러한 MPlayer는 VCD, DVD, Divx, Mpeg1/ Mpeg2/Mpeg4, AVI, ASF 등 다양한 동영상 파일을 재생시킬 수 있는 강력한 동영상 플레이어이다. 또한 광범위한 출력 드라이버를 지원하고 있으며 자막(Subtitle)을 지원하고

있어 한글 폰트의 설치로 한글 자막을 볼 수 있도록 하였다.

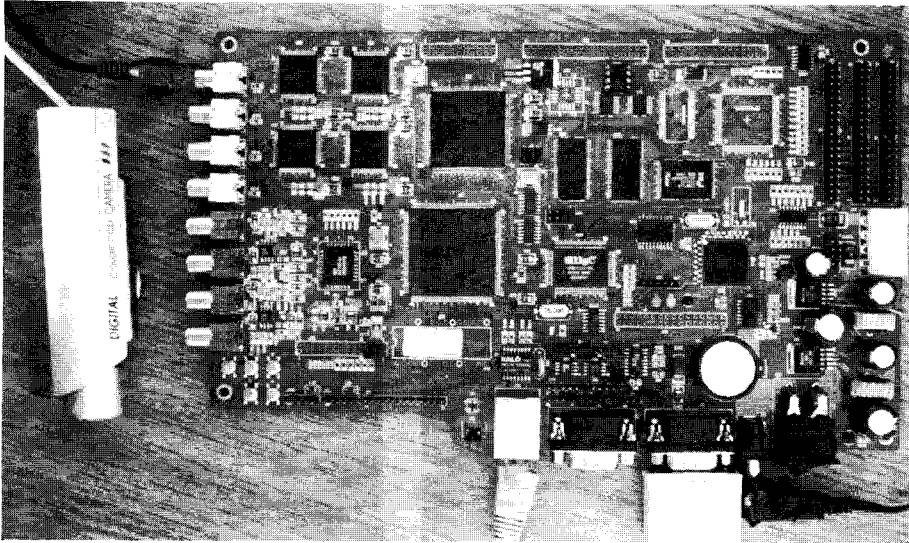
3.2 영상뷰어를 이용한 영상송신보드 모니터링 및 제어

최근 원격지에서 획득한 영상을 실시간으로 영상정보가 필요한 지역으로 영상을 전송할 수 있고 원격지 시스템을 무선으로 제어할 수 있는 영상송수신 시스템이 군사 및 소수의 산업분야에 개발이 진행 중에 있다. 따라서 원격장비에서 획득한 영상을 원격지의 타 지역에 영상을 제공할 수 있고 여러 장소에서 동시에 고급 영상정보를 획득할 수 있는 영상송신시스템 개발의 필요성을 가지게 된다.

이러한 영상송수신부를 구축하기 위해 임베디드 리눅스 시스템을 이용한 방법으로 기존에 출시된 개발 보드를 사용하였다. 영상송신보드는 IME6400 Chip을 사용하고 있고, 임베디드 시스템 개발 키트로서 실시간으로 네트워크를 통하여 영상/음성을 전달하는 특징으로 멀티미디어를 사용하는 제품을 개발할 수 있다. 영상송



<그림 4> 영상 스트리밍을 위한 네트워크 구성도



<그림 5> 영상송신보드와 영상을 획득하는 카메라

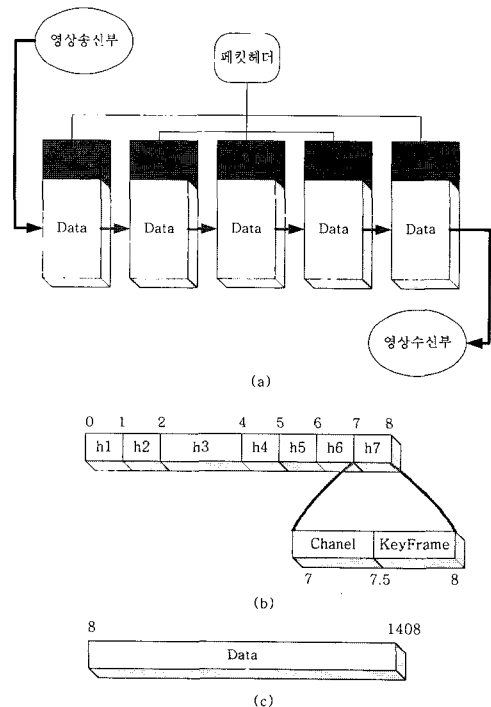
신보드는 리눅스 커널 2.4.5를 기반으로 무인 비행체등에서 받아 온 영상을 TCP/IP 통신을 통하여 각각의 인증을 획득한 클라이언트에게 영상을 송신한다. 영상수신부에서는 영상송신보드에게 Encoding Type, Video Picture Size, Video Frame Rate, Picture Encoding Type, Quantization level, I Picture Interval, BitRate를 제어할 수 있다.

영상송신부의 OS는 Linux 2.4.5 이며(김성용 외, 2004), <그림 5>는 영상송신부와 카메라이다. 영상송신부의 소프트웨어 구성은 커널 2.4.5를 사용하였고 카메라에서 받은 영상은 크게 두 가지의 명령어(실행파일)에 의해 달라진다. 한 가지는 실시간으로 네트워크로 패킷화된 영상데이터를 영상수신부에게 보내는 경우와 다른 한 가지는 AVI 포맷으로 로컬 하드디스크로 저장되어 웹을 통해 저장된 영상 데이터를 읽는 방식이다.

본 논문에서는 후자의 방법은 고려하지 않고 전자의 방법인 네트워크로 보내어지는 패킷의 헤더 부분에 각종 제어 신호를 추가시켜 영상

송신보드를 제어한다.

네트워크로 보내어지는 패킷을 도식화하면



<그림 6> 영상송신부에서 영상수신부로 전달되는 패킷의 형식

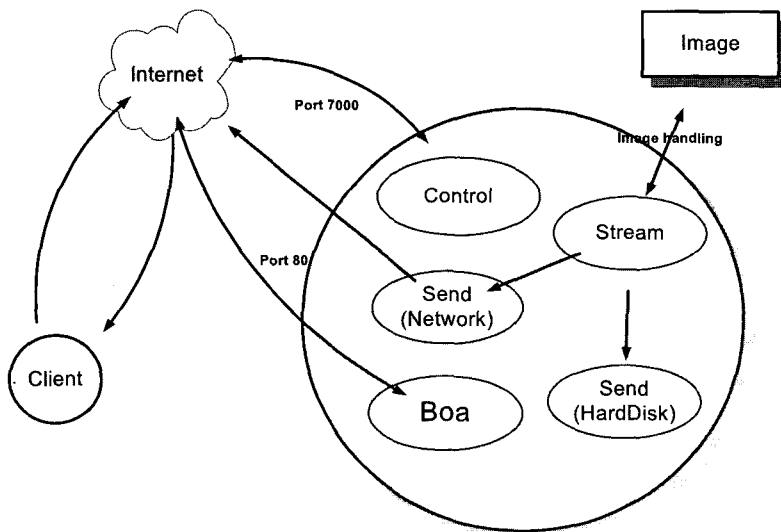
<그림 6>과 같다. 영상송신부는 장치로부터 하나의 프레임을 읽어 들인 후, 데이터 사이즈에 맞게 프레임을 자르고 헤드를 붙인 다음 수신부로 보낸다. 따라서 영상수신부 측에서도 실시간으로 전송되어 오는 데이터를 저장하기 위해서는 전송받은 패킷에서 헤드를 분리한 후 데이터만 얻는다.

<그림 6>의 (a)는 영상송신부에서 영상수신부로 헤드와 데이터가 전체적으로 전달되는 그림이고 하나의 헤더와 하나의 데이터는 한 개의 패킷이 된다. (b)는 헤더부분을 상세화한 그림인데, 전체 크기는 8바이트이다. (c)는 영상 데이터를 의미하며 크기는 1400바이트이다.

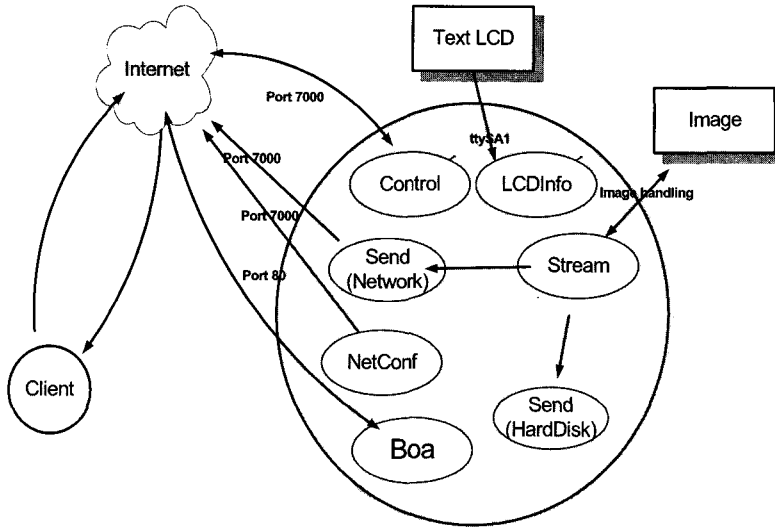
최초 영상송신부에 탑재되어 있는 응용프로그램은 카메라에서 영상을 획득 후 MPEG4 인코딩 칩을 통해 커널에게 데이터를 전달 받는다. 이것을 네트워크로 영상을 전송하기 위해서 3개의 프로세스가 존재하는데 Stream 프로세스는 영상을 Send 프로세스에게 전달한다. Send 프로세스는 받아들인 영상을 버퍼에 가지고 있다

가 영상수신부의 요구에 의해 전달되거나 하드디스크에 저장을 한다. 그리고 Control 프로세스는 Client(영상수신부)의 네트워크 연결 관리를 담당한다. 영상송신보드의 기본 프로그램은 <그림 7>과 같다.

<그림 7>의 프로그램 구조는 영상송신부의 상태를 영상수신부에게 실시간으로 전달하기 위해서는 송수신되는 패킷정보를 수정해야 한다. 이를 위해서는 새로운 프로세스를 만들어서 패킷을 재구성해야 된다. 그 이유는 네트워크 전송을 위한 Send 프로세스나 커널에게 받은 영상데이터를 일시적으로 가지고 있는 Stream 프로세서에서 패킷의 재구성이 이루어지면 각각의 프로세서는 무리한 작업이 따르게 되고 이에 따라 동기화가 맞지 않을 수도 있기 때문이다. 즉, 교착상태(deadlock)에 빠질 수 있고 올바른 데이터가 전달되지 않을 가능성이 높다. 이를 해결하기 위한 개선된 프로그램은 <그림 8>과 같다.



<그림 7> 영상송신보드의 기존 프로그램 구조



<그림 8> 영상송신보드의 개선된 프로그램 구조

영상송수신부에 추가되는 기능인 LCD에 시스템 상태(CPU, Network, Memory) 표시와 영상수신부의 요구에 따라 설정 변경을 하기 위해 두 가지 프로세스를 <그림 8>과 같이 추가하였다.

LCDInfo 프로세스는 Text LCD를 사용할 수 있도록 디바이스 드라이버를 구현하고 커널에게 동적으로 적재를 시킨다. Text LCD를 적재한 디바이스 드라이버를 연결시키기 위한 포트는 StrongARM CPU에서 사용하는 직렬 포트의 하나인 ttySA1을 통하여 LCD에 출력 시키는데 출력되는 내용은 네트워크 이상 유무, CPU 사용량, 메모리 사용량, 현재 프레임 수 등이다.

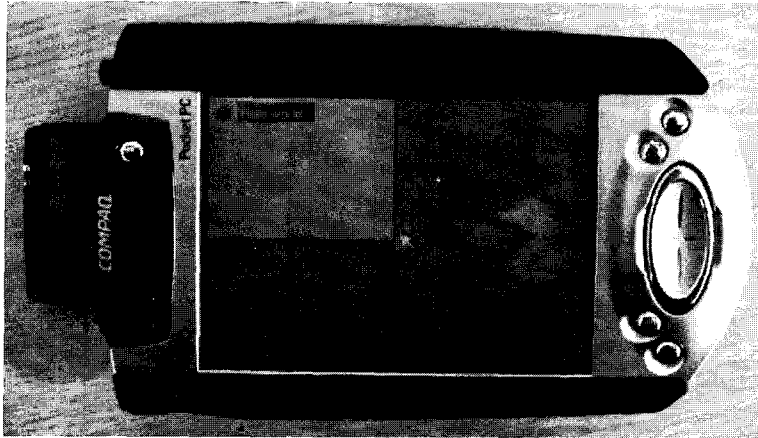
NETConf 프로세스는 영상송신보드의 상태를 패킷 헤더에 실어서 영상수신부에 전송되며, 영상수신부는 헤더부분을 읽어 영상송신부의 상태를 실시간 체크할 수 있다. 영상수신부에서 영상송신부의 제어는 설정 값을 가지고 있는 패킷을 영상송신부에 전달되며 영상송신부는

영상수신부에게 받은 패킷을 분석하여 설정 값들을 적용하고 영상송신 소프트웨어가 재가동된다.

3.3 PDA를 이용한 스텝 모터 제어

본 논문에서는 임베디드 리눅스가 탑재된 PDA를 GUI 기반으로 스텝 모터 원격 제어를 제안한다. 3.1절에서 소개한 임베디드 리눅스가 탑재된 PDA를 사용하며, 추가적으로 GUI 인터페이스를 표현하기 위해 리눅스에서 대표적으로 GTK와 Qt 중 본 논문에서는 Toolkit으로 qt-embedded-free-3.3.3을 사용하였다(<http://www.trolltech.com>). 먼저 호스트에 Qt(<http://www.trolltech.com>)를 크로스컴파일 한 후 환경 구축이 모두 완료된 상태이면, PDA에 Qt를 사용하기 위해 크로스 컴파일된 라이브러리를 전송한다.

<그림 9>는 PDA에서 Qt를 이용한 프로그램



<그림 9> Qt를 이용한 “Hello World” 출력 프로그램

으로 “Hello world”의 출력을 나타낸다. 물론 전 문에서는 직렬 포트에 연결하여 Zmodem을 통 송할 때는 여러 방법을 이용하면 되지만 본 논 하여 라이브러리를 호스트에서 타겟(PDA)으로

```
PATH="/usr/local/bin:/usr/bin:/bin"

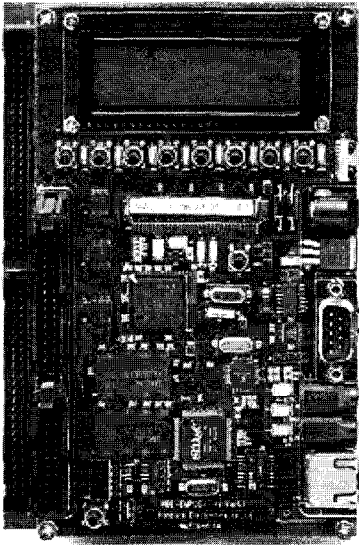
if [ " id -u " -eq 0 ]; then
    PATH=$PATH:/usr/local/sbin:/usr/sbin:/sbin:
fi

if [ "$PS1" ]; then
    PS1='\u@\h:\w\$ '
fi

if [ -d /etc/profile.d ]; then
    for i in ls /etc/profile.d/; do
        . /etc/profile.d/$i
    done
    unset i
fi

export PATH PS1 OPIEDIR QPEDIR QTDIR
export QWS_MOUSE_PROTO=linuxtp:/dev/touchscreen/0
//PDA에게 실질적으로 디바이스를 사용한다는 의미가 된다.
umask 022
```

<표 2> Touch Panel 을 사용하기 위한 profile 수정 내용



<그림 10> Tiny 보드

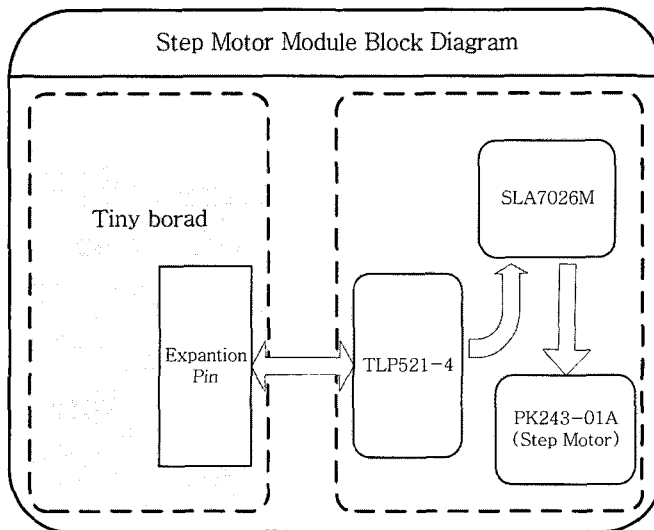
전송하였다.

다음으로 터치스크린을 구동하기 위해서는 몇 가지 과정을 거쳐야 한다. 먼저 Qt 컴파일 시 `-DQT_QWS_IPAQ` 의 옵션으로 컴파일 해야 하며 `lsmod` 명령으로 터치스크린 디바이스 드라이버는 이미 올라가 있는 것을 확인하고

<표 2>와 같이 `/etc/profile` 파일 수정한다. 스텝 모터 구현부는 임베디드 리눅스가 탑재된 H사에서 상용화 되고 있는 Tiny 보드를 이용하여 Tiny 보드의 확장 I/O 포트를 통해 스텝 모터를 제어하였다.

<그림 10>은 Tiny 보드는 다양한 디바이스 제어 및 임베디드 시스템 개발을 위한 보드로서 커널 패치 및 컴파일, Ramdisk 생성 및 수정, 웹서버 포팅, LED I/O 및 각종 디바이스 설계를 실습할 수 있는 장비이다(http://www.hanback.co.kr/htm/sub2_3.htm).

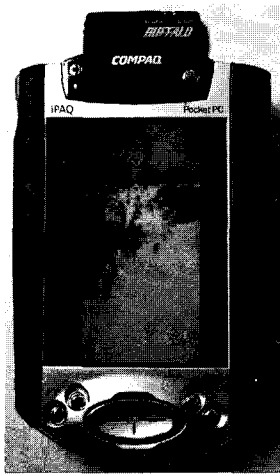
스텝 모터를 동작시키기 위해서는 5V, 12V의 신호가 필요하다. 따라서 Tiny 보드는 3.3V로 동작하기 때문에 외부에서 전원을 공급해 주어야 한다. 또한 확장 I/O 포트에서 나오는 3.3V를 스텝 모터 입력 신호인 5V로 전환시키기 위하여 TLP521-4 디바이스를 사용하였으며 전환된 신호 값은 스텝 모터 디바이스 드라이버 IC인 SLA7036M으로 전달된다. 스텝 모터의 전반적인 블록 다이어그램은 <그림 11>과 같다.



<그림 11> 스텝 모터의 동작 구조를 나타내는 다이어그램

IV. 실험 결과 및 고찰

임베디드 리눅스 기반 PDA 영상 스트리밍에 서는 영상을 획득하기 위해 영상전송서버와 NFS 프로토콜을 사용하였으며 영상을 스트리밍하고 있는 결과 화면은 <그림 12>와 같다.



<그림 12> PDA 영상 스트리밍 결과 화면

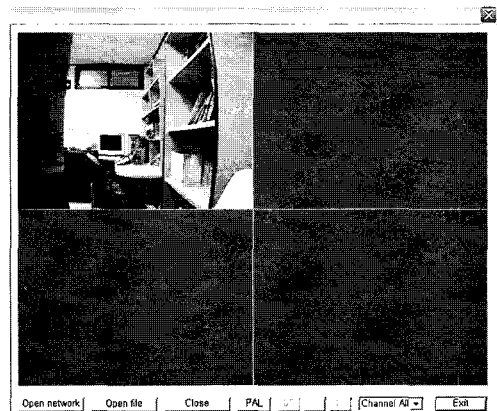
<그림 12>에서 보고 있는 결과 화면은 PDA가 지원하고 있는 240*320 크기로 구현하였으며 영상송신서버에서 전송하고 있는 원본 영상 파일은 240*352의 크기를 가지고 초당 30프레임을 전송하는 MPEG-4 압축 파일이다. 임베디드 리눅스 기반 PDA 영상 스트리밍에서는 MPlayer 자체가 스트리밍을 위한 플레이어가 아니다. 다시 말해, 원본 영상의 크기가 PDA의 최대 해상도보다 클 경우 버퍼의 크기를 넘어 서게 된다. 따라서 버퍼링을 하는 시간을 초과하여 프레임을 제대로 화면에 전달하지 못하게 된다. 이것은 단순히 버퍼의 크기를 늘리는 방법으로 수정하거나 이중 버퍼링 방식으로 이러

한 문제점을 해결할 수 있다. 단, 임베디드 시스템의 메모리는 매우 한정되어 있음을 고려해야 한다.

PDA를 이용한 스텝 모터 제어에서는 영상 스트리밍에 사용되는 PDA를 사용하였으며 네 가지 방향을 전달하기 위한 버튼과 종료 버튼으로 구성하였다.

<그림 13>은 원격 영상송신보드의 영상을 개인용 컴퓨터의 모니터에 나타내고 원격 영상 송신보드의 매개변수 및 패킷 등을 제어하기 위한 영상뷰어를 나타낸다. 이 영상뷰어를 이용한 영상송신보드 모니터링 및 제어에서 개인용 컴퓨터 기반 영상수신부의 화면은 <그림 13>과 같다. 아래의 결과 화면은 Open network 버튼을 클릭하여 영상송신부의 IP를 입력하고 영상 송신부의 접속 허락을 수락한 후 영상을 받아 오는 화면이 된다. 또한 NTSC 방식과 PAL 방식을 모두 지원하고 있고 4채널 각각의 영상을 획득 할 수 있다.

서버프로그램의 실행 및 영상전송 상태를 영상뷰어 화면에서 나타내기 위해 영상송신보드의 직렬 포트에 연결하여 나타낸 결과 화면은



<그림 13> 영상수신부의 결과 화면

```

Success
REG-IncTime = 000003e9
REG-AUDIO = 10000080
REG-CBR = 0f0f0f0f
REG-MD = 00000000
REG-CH1CBR = 0c000400
REG-CH2CBR = 0c000400
REG-CH3CBR = 0c000400
REG-CH4CBR = 0c000400
REG-DC = 0000000c
PicSize = 320 x 240
Ch1 = on, Ch2 = off, Ch3 = off, Ch4 = off
REG-COM = 1a800009
0x2000 -> 0x1a800009
0x2004 -> 0x00000000
0x2008 -> 0x000003e9
0x200c -> 0x10000080
0x2010 -> 0x0f0f0f0f
0x2014 -> 0x00000000
0x2028 -> 0x0c000400
0x202c -> 0x0c000400
0x2030 -> 0x0c000400
0x2034 -> 0x0c000400
117 frames / 0:00:05: secs [ 28.6 frames/sec ] ( 13.6kbps)_
    
```

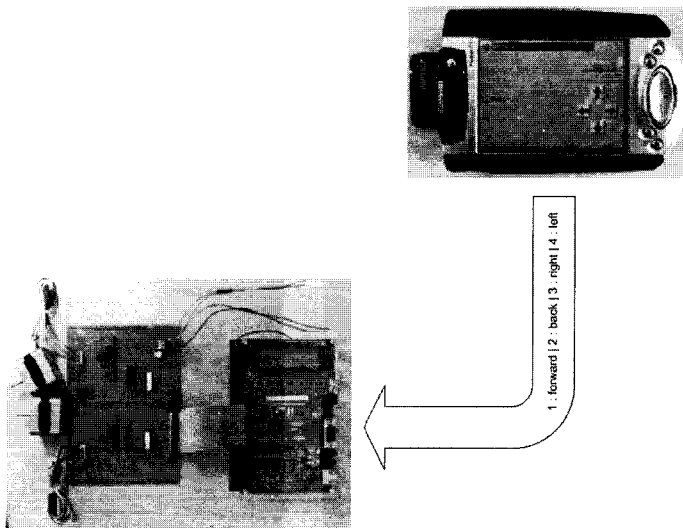
<그림 14> 영상송신보드의 터미널 모니터링 화면

<그림 14>와 같다. 이 결과 화면에서는 채널에 따른 기본 설정 값들과 영상의 크기, 활성화된 채널의 상태, 총 전송된 프레임 수, 전송된 시간, 초당 전달되는 프레임 수와 전송된 데이터 크기 등을 표시하고 있다.

<그림 15>는 PDA의 응용프로그램과 Direct Signal(1:forward, 2:back, 3:left, 4:right)이 전송

되는 과정을 보여주는 결과 화면이다. PDA에서 방향 버튼을 선택하면 UDP(user datagram protocol) 프로토콜을 통해 Tiny 보드에게 전송되고 다시 Tiny 보드에서는 그 선택된 값에 의해 스텝모터를 제어하게 된다.

PDA를 이용한 스텝 모터 제어에서는 4바이트의 신호 값을 사용하기 때문에 네트워크 상



<그림 15> PDA를 이용한 스텝 모터 제어 결과 화면

태에 따라 신호 값의 오류를 찾을 수 없다. 또한 스텝 모터의 속도를 `usleep()` 함수를 사용하기 때문에 프로세스의 관리가 제대로 이루어지지 않을 가능성이 있다. 이것은 스텝 모터 구현부에 다른 마이크로컨트롤러를 설계하여 스텝 모터를 안전하게 제어할 수 있도록 하면 된다. 또한 무선 이더넷 장치를 사용하고 외부 전원을 설계한다면 이동식 로봇을 구현할 수 있다.

영상뷰어를 이용한 영상송신 보드 모니터링 및 제어에서 영상송신보드의 IME6400 압축 칩에는 CBR과 VBR 기능이 추가되어 있는데, 이것은 각각 고정비트(constant bitrate)와 가변비트(variable bitrate)의 의미로, 고정비트는 영상의 각 프레임이 똑같은 비트 수를 가지지만, 가변비트는 파일 크기는 줄이면서 동일한 품질을 유지하기 위해 사용된다. 예컨대 움직임이 빠른 장면은 더 많은 비트 수가 할당하고, 움직임이 느린 장면은 더 적은 비트수가 할당하는 것이 가변비트이다. DVD는 고정비트를 사용하고, DIVX나 XVID 등은 가변비트를 사용한다. 하지만 IME6400의 칩 버그로 CBR 기능을 완전히 구동하지 않는다. 즉, CBR 형식으로 영상을 전송할 때 네트워크의 상태를 고려하면서 5프레임 정도의 오차를 보이고 있다.

또한 개발한 소프트웨어는 영상 스트림을 전달하기 위해서 Semaphore 방식을 사용하여 Blocking 방식으로 구현하였다. 즉, 이것은 영상의 동기화를 맞추기 위해 사용하는 방식이다. 그러나 Blocking 방식으로 스트림을 전달하면 원하고자 하는 상황에서 프로세스를 제어하지 못하는 단점을 가지고 있다. 때문에 영상이 들어오다가 영상신호가 없을 경우에 교착상태

(deadlock)에 빠질 수가 있다. 따라서 영상 스트림을 전달하기 위해서는 Nonblocking 방식을 채택하여 영상 스트림을 안정적으로 프로세스 제어할 수 있도록 해야 한다.

V. 결론

본 논문은 영상송신보드를 이동할 수 있도록 스텝모터에 탑재시키고, 획득한 영상을 임베디드 리눅스 기반의 PDA를 통하여 스트리밍 및 스텝 모터를 제어하는 모듈을 각각 개발하였다.

이를 구현하기 위한 첫 번째 모듈은 무선 네트워크를 통해 PDA의 Direct Signal을 임베디드 리눅스가 탑재된 보드로 전달되고 보드의 확장 I/O 포트를 통해 스텝 모터가 제어할 수 있는 모듈을, 두 번째 모듈은 임베디드 리눅스가 탑재된 PDA에서 실시간 영상 스트리밍 하는 모듈, 그리고 세 번째 모듈은 퍼스널 컴퓨터에서 영상송신보드를 TCP/IP 통신을 이용한 프로세스 제어 모듈을 개발하였다. 또한 PDA에서는 영상을 수신하면 전달 받는 데이터 헤더 부분을 체크하여 영상송신보드의 상태를 체크하고 영상송신보드에게 스텝 모터를 제어할 수 있는 신호를 전송할 수 있게 구현하였다. 하나의 PDA를 이용해 각각의 모듈을 통합해서 운용할 수 있는 시스템을 구축한다면 영상보드를 스텝 모터 운용시스템에 탑재시켜 원격으로 영상 송신 보드를 제어할 수 있는 시스템 통합이 가능하리라 판단된다.

이러한 임베디드 리눅스 시스템은 각각의 다른 CPU 및 디바이스를 사용하고 있으며 개발된 프로그램도 각각 다르다. 하지만 리눅스는

이식성이 높은 장점을 지니고 있으므로, 영상송신보드로 통합시 큰 문제를 일으키지는 않는다. 그렇기 때문에 사용되는 모듈들을 하나로 통합하고, 스트리밍 전송기술과 압축방식 등을 고려하여 임베디드 리눅스 기반 PDA에서 패킷손실과 지터 등이 없는 스트리밍 서비스를 안정적으로 제공하는 안정적인 알고리즘 개선 등의 다양한 연구가 요구된다.

참고문헌

강진범, 김정인, "PDA를 통한 Linux 서버 관리 시스템 구현", 한국 멀티미디어학회 춘계학술발표논문집, 2003, pp. 683-686.

김성호 외, "AANN을 이용한 웹-모니터링 시스템 설계에 관한 연구", *Journal of Control, Automation and Systems*, Vol. 10, No 6, 2004, pp. 450-550.

김성용 외, "임베디드 리눅스 기반 네트워크를 통한 영상 송신 시스템 제어", 한국정보과학회 학술발표논문집, 제12권, 제1호, 2004, pp. 168-173.

김성용 외, "무선 네트워크를 이용하여 임베디드 리눅스 기반 PDA에서의 영상 스트리밍 설계 및 구현", 한국멀티미디어학회 춘계학술발표논문집, 제 8권, 제1호, 2005, pp. 158-160.

김지훈, "임베디드 시스템을 위한 실시간 스케줄링 알고리즘 기법", 영남대학교 석사학위논문, 2006.

나종화 외, *Embedded System Programming*, 생능출판사, 2003.

박태규, "인터넷을 이용한 임베디드 리눅스 기반의 PDA형 원격 제어 시스템 개발", 성균관대학교 석사학위논문, 2004.

박태현 외, "인터넷을 이용한 이동 로봇의 원격 운용 시스템", 제어·자동화·시스템공학 논문지 제8권, 제3호, 2002, pp. 270-274.

이동만 외, "u-비즈니스를 위한 RFID 시스템 도입에 관한 실증 연구", 정보시스템연구, 제15권, 제4호, 2006, pp. 225-244.

이승현, "OBS 망에서 임베디드 시스템 기반의 MEMS 스위치 제어 장치 설계 및 구현", 전북대학교 석사학위논문, 2006.

이연조, 임베디드 리눅스 프로그래밍, PCBOOK, 2002.

차윤숙 외, "유비쿼터스 특성요인이 모바일 서비스의 사용의도에 미치는 영향", 정보시스템연구, 제16권, 제2호, 2007, pp. 69-90.

한중희, "PDA를 이용한 원격 로봇 제어 시스템", 홍익대학교 석사학위논문, 2004.

Bovet, Daniel P. 외, 리눅스 커널의 이해, 한빛미디어, 2003.

Goldberg, K. and Mascha, M., "Desktop Teleoperation via the World Wide Web", 1995 *IEEE Int'. Conf. Robotics and Automation*, Vol. 1, 1995, pp. 654-659.

Luo, R.C., Su, K.L., Shen, S.H., and Tsai, K.H., "Networked Intelligent Robots Through the InterNet: Issues and Opportunities", *Proceedings of the IEEE*, Vol. 9, No. 3, 2003, pp. 371-382.

Paulos, E. and Canny, J., "Delivering Real Reality

to the World Wide Webvia Telerobotics",
in Proc, 1996 *IEEE Int. Conf. Robotics
and Automation*, Vol. 2, 1996, pp.
1694-1699.

Stevens, R., *Advanced Programming in the Unix
Environment*, Addison-Wesley, 1993.

Stevens, R., *Unix Network Programming*,
Prentice-Hall, 1991.

<http://www.buffalotech.co.kr/down/access.html>.

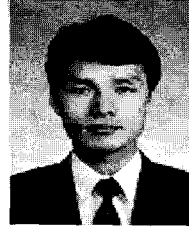
<http://www.embedded.com>.

http://www.hanback.co.kr/htr/sub2_3.htm.

[http://www.mplayerhq.hu/homepage/design7/dload.
html](http://www.mplayerhq.hu/homepage/design7/dload.html).

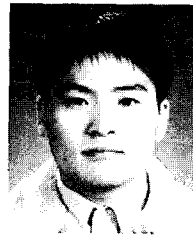
<http://www.trolltech.com>.

김성용(Kim, Sung-Yong)



경북대학교 전자공학과에
서 학사, 석사, 박사학위를 취
득하였다. 현재 위덕대학교
컴퓨터공학부 부교수, 공학교
육혁신센터장으로 재직 중이
며, AIN Technology 대표이사
로 있다. 주요 관심분야는 임
베디드 리눅스, 미들웨어, 모
바일 컴퓨팅 등이다.

이상민(Lee, Sang-Min)



위덕대학교 컴퓨터공학과
에서 학사, 석사학위를 취득
하였고, 현재 박사과정에 재
학 중이다. 주요 관심분야는
임베디드 리눅스, 임베디드
시스템, 모바일 컴퓨팅 등이
다.

<Abstract>

Implementation of an Image Board Remote Control System using PDA based on Embedded Linux in Wireless Internet

Sung-Yong Kim · Sang-Min Lee

This thesis proposed a method that connecting step motor to image send board which can acquire image to move and remote controlling via streaming image board of PDA(personal digital assistant) based on embedded Linux which is using wireless network. There are three embedded Linux system to embody movable image send board.

First, though the wireless network a signal of PDA is transmitted to the board which has embedded Linux and a system which is controlled by the expansion I/O port of the board.

Second, it's a system streaming realtime image at a PDA which has embedded Linux.

The last is a system which controls a process of image board using TCP/IP communication and image send board at PC.

These are the system which can use industrial settings and homes. It can also make use of an embodiment method about travelling image robot.

Keywords: PDA, Image Streaming, Embedded Linux, Image Board Remote Control

* 이 논문은 2008년 2월 14일 접수하여 2차 수정을 거쳐 2008년 3월 10일 게재 확정되었습니다.