

임베디드 시스템을 이용한 무선랜 통신에 관한 연구

A Study of Wireless LAN Communication using Embedded System

이 창 균*, 최 재 우*, 노 방 현*, 황 회 응*

(Chang Keun Lee, Jae Woo Choi, Bang Hyun Ro, Hee Yeung Hwang)

*호서대학교 전기정보통신학부 (전화:(041)548-5747, 팩스:(041)548-0965, E-mail : coba_lee@hanmail.net)

Abstract : In this paper, we designed the embedded system used for wireless LAN communication. Embedded system kernel is made from general linux kernel 2.4.18 by applying the ARM patch (2.4.18-rmk7) and the SA1100 patch(2.4.18-rmk7), then porting board level suitable to target system. The SA-1110 PCMCIA interface provides controls for one PCMCIA card slot with a PSKTSEL pin for support of a second slot. The embedded system requires external logic to complete the PCMCIA socket interface. For dual-voltage support, level shifting buffers are required for all SA-1110 input signals. Hot insertion capability requires that each socket be electrically isolated from each other, and from the remainder of the memory system. embedded system is for socket services approaching PCMCIA socket, detecting number of sockets, sensing insertion and removal, and applying power. It also provides interface with Card services. Embedded system supports Host driver for lucent chips that is installed orinoco driver cross compiled. The meaning can say that is doing wireless LAN communication through wireless LAN in imbedded system.

Keywords : Wireless LAN, porting, PCMCIA, Embedded,

1. 서론

최근 임베디드 시스템은 자동차, 우주, 항공, 군사, 의료장비와 공정 제어 등 산업용에서부터 가전제품, 휴대용 전화기 등 까지 영역을 확대하여 우리 생활에 밀접하게 다가서있다. 포스트 PC시대에 PC 시장의 성장 속도는 점차로 감소되고 있으며, 사람들은 이제 PC 수준과 동일한 기능을 하면서도 TV처럼 사용하기 쉽고 편리하며, 휴대하기 쉬운 장비를 원하고 있다. 포스트 PC는 사무용 오피스군의 애플리케이션에서부터 유무선 인터넷까지 개인 일정관리 및 이메일 송수신에서부터 회사 데이터베이스에 접근하여 업무를 처리하는 전 영역을 지원한다.

임베디드 시스템이란 정해진 특정 기능을 수행하기 위해 하드웨어와 소프트웨어가 내장되어 있는 시스템을 말하며, 마이크로프로세서가 내장되어 있으며 원하는 작업을 수행하기 위해 마이크로프로세서를 운영하는 프로그램이 포함된 시스템을 말한다. 임베디드 시스템에서 사용되는 리얼타임 운영 체제는 임베디드 리눅스, MS사의 Wince, VxWork, μ C/OS2, eCOS, pSOS 등이 있다. 임베디드 시스템에서 운영체제를 점점 더 많이 도입하는 이유는 좀더 많은 기능을 가진 복잡한 하드웨어가 사용되고 예전과는 비교 할 수 없는 많은 소프트웨어 기능이 요구되기 때문이다. 이제 최신의

32비트급이상의 RISC칩이 채용된 임베디드 시스템은 리얼타임OS를 통하여 멀티태스킹 기능을 충분히 소화해 낼 수 있는 수준으로 발전하였다. 임베디드 시스템의 장점은 적은 자원으로도 사용자가 요구하는 조건을 충분히 충족시킬 수 있다는 것이고, 적은 소비전력으로 설계가 가능하여 휴대성이 뛰어나다는 장점이 있다. 그리고 비록 소규모이긴 하지만 웹 서버나 파일 전송프로토콜 서버 등으로도 사용이 가능하게 되었다.

리얼타임 OS 중 리눅스는 1991년 리누스 토발즈에 의해 처음으로 세상에 등장했다. 그 이후 전 세계의 많은 개발자의 공헌에 힘입어 운영체제 커널에서부터 GUI(Graphic User Interface)를 포함한 응용 프로그램까지 모든 분야에 있어 발전을 거듭한 끝에 현재 인터넷의 서버로 가장 많이 사용되고 있는 운영체제로 자리 잡았다. 이제 리눅스는 임베디드 리눅스라는 이름으로 임베디드 시스템 분야에서 특히, 포스트 PC에서 사용될 가장 유력한 운영체제로 떠오르고 있다. 임베디드 리눅스가 주목받는 중요한 이유는 리눅스의 모든 소스는 공개 되어 있다는 점과 개발비용 및 사용료가 거의 들지 않는다는 것이다[1].

유무선 통신구현 기술은 유비쿼터스(Ubiquitous)컴퓨팅을 위한 핵심기술로서 자리매김하고 있으며 현재 PDA, 웹패드, 화상전화기, 차세대 이동단말장치 등 언제 어디서나 인터넷에 접속할 수 있는 기술들이 계속

해서 발표되고 있다.

본 논문에서는 Intel社의 SA1110이라는 32bit ARM 프로세서를 응용하여 임베디드 시스템을 구성하고, 무선통신을 하기 위해 리눅스를 포팅하여 OS차원에서 시스템을 감시하여 자원에 효율성과 안정성을 보장하기 위해 사용하였으며, 무선랜카드는 IEEE에서 2.4GHz 대역 11Mbps를 지원하는 802.11b 규격의 무선랜을 사용하였다. 소켓 규격은 PCMCIA(Personal Computer Memory Card International Association)의 규약을 사용하였다. 또 임베디드 시스템과의 무선랜을 인터페이스하기 위해 버퍼회로와 전원인가회로를 추가 설계하여 현재 판매되는 모든 무선랜카드의 지원을 하게 하였다. 최종적으로 무선랜통신의 검증을 위하여 TCP/IP 응용서비스 즉 ftp나 telnet, 웹서버 프로그램을 수행해보았다.

II. PCMCIA 와 무선랜

1. PCMCIA 구조

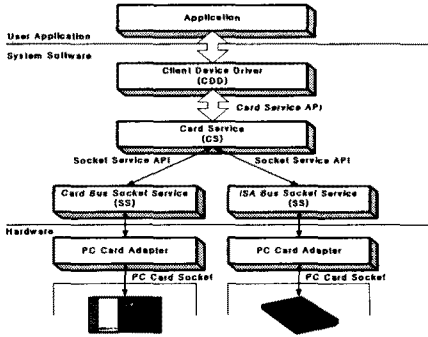


그림 1. PCMCIA 기본 구조.

Fig. 1. PCMCIA general architecture.

PCMCIA 기본 구조는 그림 1과 같다. PCMCIA의 구조는 하드웨어 측면과 소프트웨어 측면으로 나누어 지게 된다. 하드웨어 측면은 16bit 방식과 물리적소켓, HBA(Host Bus Adapter)로 구분되며, 소프트웨어 측면은 Socket Services, HBA에 대한 정보 및 초기화로 나누어진다. 또 Card Services와 Card Enablers로 구분되어진다. Card Services는 첫째 프로그래머가 카드로 액세스 할 수 있도록 하고, 둘째 설정 요구사항을 결정하고, 셋째 그 필요한 시스템 자원을 요청하는 기능을 수행하는 고수준 함수들로 구성된 소프트웨어 계층을 제공한다. 특히 인에이블러(Enabler)를 위해 주된 기능중의 하나가 시스템 자원 할당이다. 구체적으로 그 역할을 살펴보면, 카드서비스는 pc카드에 할당 가능한 시스템 자원의 데이터베이스를 가지고 있다. 그래서 인에이블러가 카드의 CIS(Card Information Structure)를 읽어 카드의 설정 요구사항을 결정할 후

이들 자원을 할당한다. 그러나 자원이 이미 다른 곳에서 사용되고 있다면, 할당 실패를 리턴하고 인에이블러는 CIS에서 다른 설정 선택 사항을 읽어서 다시 시도한다. 이 과정은 카드의 설정 선택사항이 요청을 만족 시킬 때까지 계속되거나 모든 선택사항을 다 소비하여 설정되어 질수 없을 때 까지 계속된다. 또한 카드서비스는 카드의 삽입 및 제거, 그리고 다른 상태 이벤트를 파악하여 인에이블러에게 알려준다. 인에이블러란 pc 카드를 설정하기 위해 인에이블러라는 소프트웨어가 필요하다. 이 소프트웨어는 pc 카드가 설치되었는지 체크하고, 만약 설치되었다면 카드의 CIS를 읽어 이 카드 장치의 타입, 필요한 시스템자원, 가능한 설정 선택사항 등을 결정하여 시스템 내에서 그 카드에 액세스 할 수 있도록 프로그래밍 하는 기능을 수행한다. 인에이블러의 이러한 기능을 쉽게 그리고 신뢰성 있게 구현하기 위해서 카드 서비스라는 소프트웨어 계층이 존재한다[2].

2. 무선랜(Wireless LAN)

무선랜은 케이블을 사용하지 않고 전파를 이용하여 네트워크 상에서 통신을 하는 방식이다. 무선랜의 등장은 케이블링으로 인한 설치, 유지보수, 이동의 어려움을 해소하기 위한 대안으로 대두 되었으며, 이동사용자의 증가로 인해 그 필요성이 점점 늘어나고 있는 추세이다. 무선랜 제품 제조업체들이 모두 이 표준에 의거하여 제품을 생산하고 있으며, 서로 다른 제품간의 상호 호환 여부는 Wi-Fi 라는 무선 인증 유무를 기준으로 삼고 있다. 무선랜의 구성은 AP(Access Point)와 무선랜카드로 이루어진다. AP는 전송거리 이내의 무선랜 사용자들이 인터넷 접속 및 네트워크를 할 수 있도록 전파를 보내는 장비로서 휴대폰의 기지국 또는 유선 네트워크의 허브 역할을 한다. 무선랜의 망 구성은 크게 2가지 형태로 나누어진다. 첫 번째는 Infrastructure Wireless LANs 방식으로 그림 2와 같다.

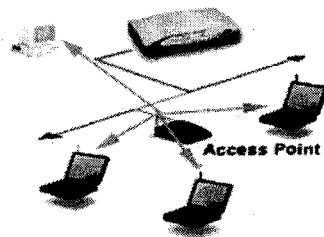


그림 2. Infrastructure 무선랜 방식

Fig. 2. Infrastructure wireless LANs

Infrastructure 방식의 네트워크는 무선 스테이션과 하나 이상의 AP로 구성된다. 일반적으로 액세스 포인트는 한쪽에는 무선 네트워크를 한쪽에는 이더넷과 연

결하여 유, 무선 네트워크를 연결시켜주는 브릿지 역할을 한다. 이러한 브릿지 역할 외에 AP는 무선 스테이션이 특정 AP의 셀을 벗어날 경우 다른 액세스 포인트를 검색하여 베이스 스테이션으로 사용할 수 있는 로밍(roaming)과 무선 스테이션이 자신과 다른 AP를 사용하고 있는 무선 스테이션과 통신 할 때 프레임을 다른 AP로 전달하는 포워딩 기능을 담당한다. 둘째는 Ad-Hoc wireless LANs 방식으로 그림 3과 같다.

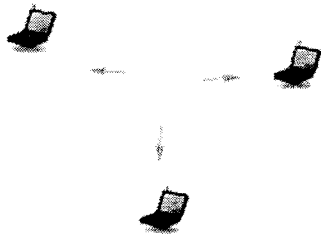


그림3. Ad-Hoc 무선랜 방식
Fig 3. Ad-Hoc Wireless LANs

Ad-Hoc 방식의 네트워크는 단순히 무선랜만으로 구성된 네트워크로 무선랜 상호간 통신(peer-to-peer)만을 할 수 있는 로컬 네트워크를 말한다. 이러한 네트워크 구성을 IEEE 802.11 표준에서는 IBSS(Independent Basic Service Set)로 정의하고 있다. 각 스테이션은 AP없이 상호간의 통신이 이루어지고 각 스테이션의 구분은 BSSID(Basic Service Set Identifier)로 이루어진다. 그러나 무선랜은 MAC(Media Access Control) 레벨에서 각 프레임의 목적지를 결정하므로 스테이션과 펌웨어와 드라이버를 브릿지(bridges)기능을 하도록 구현하며 AP 없이도 유선 네트워크와의 통신이 가능하다. 또 무선랜은 DSSS(Direct Sequence Spread Spectrum)의 전송 방식을 따른다. DSSS는 신호를 더욱 넓은 주파수 대역에 걸쳐 확산시킴으로써 과도한 출력의 집중을 방지하며, data의 각 bit는 송신기에 의해 칩이라는 형태로 변화 된다. 목적지에 선 칩이 다시 bit 단위로 변화되어 원래 data로 재현되고, 송신기와 수신기는 정상 작동을 위하여 동기화가 필요하게 된다[3].

III. 연구 내용

ARM 프로세서인 intel社의 SA1110 마이크로 컨트롤러를 이용하여 무선랜카드로 무선랜통신을 하기위해 PCMCIA 소켓 규격을 통하여 회로를 구성하였다. 또한, 무선랜을 인터페이스 하기 위해 CPU 내부에 PCMCIA 컨트롤러를 사용하여 추가 회로를 구성하여 인터페이스 하였다. Ethernet(cs8900a)는 커널이미지나 램디스크 이미지를 다운로드 하기 위해 사용되었으며, 기존에 시리얼 다운로드 방식은 다운로드하는데 시간이 오래거리는 단점으로 10Mbps의 이더넷 컨트롤러를

추가하여 다운로드하는 개발시간을 단축 시켰다. 시스템 구성은 SDRAM 16Mbyte 모듈을 2개 32bit로 인터페이스 했으며, 플래쉬 메모리는 Intel사의 Strata 16Mbyte를 16bit로 인터페이스 하였다. 플래쉬 메모리에는 부트로더, 커널컴파일한 이미지 그리고 램디스크 이미지가 저장되게 된다. UART(Universal Asynchronous Receiver/Transmitter)를 추가 설계하여 커널부팅 및 시스템 디버깅(debugging)을 할 수 있도록 하였다. 기타 디버깅 LED를 장착하여 커널부팅시나 시스템 오류시 LED로써 확인할 수 있도록 하였다. 또 JTAG 포트를 통해 최초 플래쉬 메모리에 부트로더를 쓸 수 있게 하였다.

PCMCIA에 각각의 소켓은, 시스템 전원이 들어온 상태에서도 PC카드가 삽입 및 제거 될 수 있도록, 확장 버스로부터 분리되어야한다. 그래서 데이터 버퍼를 추가 설계하여 PC카드가 소켓에 삽입되거나 소켓으로 부터 제거 될 때 확장 버스 신호선에 과도 신호가 발생하는 것을 방지하였다. 또 동작전원이 다른 무선랜 카드를 지원하기 위해서는 전원공급회로를 추가해 주워야 한다. 본 논문에서는 MIC2563A-1 컨트롤러를 추가하여 설계하였다. 그림 4는 본 논문에서 구현한 임베디드 시스템의 블록도를 보여준다. 리얼 타임 OS로는 로열티가 없는 임베디드 리눅스를 포팅하여 사용하였다.

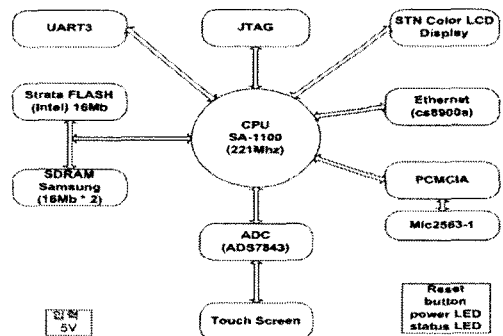


그림4. 임베디드 시스템 블록도
Fig 4. Embedded System block diagram

리눅스 포팅(porting)은 386에서 구동 되는 리눅스를 해당 타겟 보드(SA1110)에 구동될 수 있게 소스코드를 수정하는 것을 말한다. 타겟 보드에 리눅스를 포팅 하기 위해서는 먼저 크로스 컴파일러와 개발 환경이 구축되어야만 한다. 본 논문에서 사용한 커널 소스의 버전은 2.4.18 이다. 그리고 필요한 파일이 ARM 패치 파일과 SA1110 패치 파일이다. 패치(patch)파일이란 두 파일 간에 차이점을 기록한 파일이다. 패치 파일에는 커널에서 수정된 부분에 대한 파일들만 기록되어 있다. ARM 패치파일은 Russell King 이라는 사람이 작성하여 무료로 배포 하고 있다. 또 SA1110 패치 파일 SA1110 패치 파일은 Nicolas Pitre 가 패치를 적용하여

무료로 배포 하고 있다. 각각의 파일 이름은 ARM패치는 patch-2.4.18-rmk7.gz 이고 SA1110은 diff-2.4.18-rmk7.gz 이다. 여기서 중요한 것은 ARM패치나 SA1110 패치에서 많은 것을 SA1110프로세서에 맞게 수정되어 있지만 모든 SA1110 프로세서에서 동작 되지는 않는다. 왜냐하면 메모리 사이즈나 기타 하드웨어적인 요소들이 하드웨어 엔지니어에 따라 다르게 설계되어 있기 때문이다. 따라서 이부분에 대한 패치 파일이 필요하게 된다. 보드패치 라고 하는 이 파일은 타겟보드에 맞게 수정해주어야 한다. 파일 이름은 diff-2.4.18-rmk7-np1-lee.gz이다 이것은 사용자가 자신에 타겟 보드에 맞게 수정하여 만들어서 사용하여야 한다.

임베디드 리눅스에서의 무선랜 운영 방법은 Infrastructure 모드를 이용하여, 외부망에 접속하기 위해서 AP를 허브에 장착하여 IP를 할당하였으며, 임베디드 시스템에는 무선랜이 해당 AP와 통신하기 위해서 암호화와 BSS(Basic Service Set)에 참여할 수 있는 ID를 지정하였다. 또 AP설정시 나 임베디드 시스템에서 무선랜 설정시 SSID를 동일하게 구성하여야 한다. 그림5는 리눅스가 포팅된 임베디드 시스템이다.

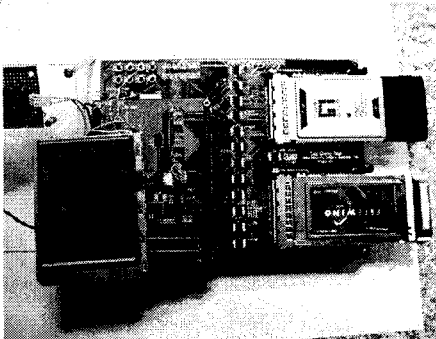


그림5. 임베디드 시스템
Fig5. Embedded System

SA1110의 PCMCIA의 메모리맵 영역은 무선랜카드가 슬롯0에 삽입된 경우에 0x2000 0000부터 0x2fff ffff 까지 이고 슬롯1에 삽입된 경우에는 0x30000 0000부터 0x3fff ffff번지 까지 접근한다. PCMCIA 메모리 맵은 그림 6과 같다[4].

0h3C00 0000	Socket 1 Memory Space
0h3800 0000	Socket 1 Attribute Space
0h3400 0000	Reserved
0h3000 0000	Socket 1 I/O Space
0h2C000 0000	Socket 0 Memory Space
0h2800 0000	Socket 0 Attribute Space
0h2400 0000	Reserved
0h2000 0000	Socket 0 I/O Space

그림6. PCMCIA 메모리 맵
Fig 6. PCMCIA Memory Map

실제 리눅스에서는 가상 번지로 매핑되어서 사용되어지며, PSKTSEL핀이 무선랜 카드가 삽입되어 있는 슬롯을 선택하게 된다. CD1#, CD2#핀은 카드가 삽입되거나 제거되었을때 체크하게 된다. VS1#, VS2#핀은 전압 감지 핀으로 특정 소프트웨어가 CIS 정보를 읽어 카드에 알맞은 전압을 공급시키는 역할을 하고 있다. 무선랜카드에 대한 디바이스 드라이버는 orinoco드라이버를 사용하여 구성하였다 [5].

무선랜카드에 대한 무선통신 테스트는 ftp, telnet, 기타 소켓 프로그램을 작성하여 그림 7과 같이 실험하였다. 또 보아 웹 서버를 임베디드 시스템에 올려 다른 클라이언트 PC에서 임베디드 시스템에 무선으로 접속되는 것을 알 수 있었다.

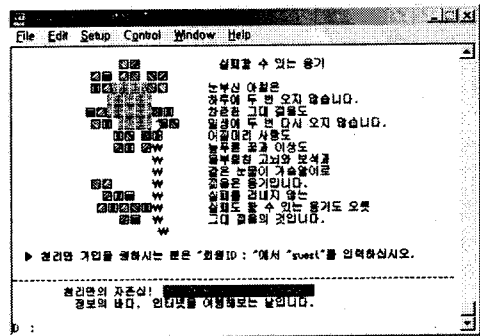


그림7. telnet에 접속한 화면
Fig7. Contacted screen by telnet

IV. 결론

본 논문은 최근 발전하고 있는 임베디드 리눅스를 ARM프로세서 SA1110에 포팅하여 무선랜을 통하여 리눅스에서의 무선랜 통신과 회로를 구성하여 임베디드 시스템에서의 무선랜통신 검증을 수행하였다. 현재 무선랜은 54Mbps 까지 지원되어 출시되고 있는 실정이며, 더욱 발전할 계획이다. 그리고 각종 모바일과 ebook, webpad, 홈네트워킹, etc 등 임베디드 시스템에서 무선랜을 통하여 무선인터넷을 가능하도록 하는 데에 그 의의가 있다고 할 수 있다.

참고문헌

- [1] 신영석, 박동선, 주수중, *Embedded Linux 이론과 실습*, 도서출판 홍릉과학출판사, 서울, 2003.
- [2] 권수호 *Linux Programming Bible*, 도서출판 글로벨, 서울, 2003.
- [3] (주)palmpalm, "타이눅스 박스 II 무선랜 사용자 가이드" 매뉴얼 pp. 5-6, 2001.
- [4] Intel, intel strongARM SA-1110 microprocessor developer's Manual pp. 10-55 ~ 10-63, 2000.
- [5] Device driver web page, <http://www.ozlabs.com/>.