

임베디드 인터넷에서 중계기 관리를 위한 Linux Kernel Module의 개발

금유환⁰ 고국희 정경호 안광선
경북대학교 컴퓨터 공학과
{ch3cooh⁰, gogooki, mccart, gsahn}@knu.ac.kr

Development of Linux Kernel Modules for the Cell Enhancer Management based on Embedded Internet

Yu-hwan Keum⁰ Goo-Ki Go Kuung-Ho Chung GWang-Sun Ahn
Dept. of Computer Engineering, Kyung-pook University

요 약

최근 하루가 다르게 증가하는 이동전화 가입자의 통화 품질을 보장하기 위해 CDMA중계기의 중요도가 증가하고 있으며 자동화된 모니터링 시스템의 필요성이 대두되었다. Embedded Internet하에 CDMA중계기를 임베디드 리눅스 시스템과 연동 하여 리눅스 커널 레벨에서의 모니터링 방안으로 EKMM(Embedded internet based Kernel level Monitoring Modules)시스템을 제시한다. 본 시스템은 SA-1110 마이크로프로세서로 구현된 하드웨어에 시리얼 디바이스 드라이버와 네트워크 디바이스 드라이버를 설계하였다. 또한 리눅스 커널을 재구성하며 동적 로딩 방식으로 EKMM이 탑재되어 기존의 중계기 임베디드 시스템 보다 기능 확장성이 뛰어나고 성능이 향상된 중계기 모니터링 시스템을 설계 구현하였다.

1. 서 론

임베디드 시스템은 정해진 특정 기능을 수행하기 위해 하드웨어에 소프트웨어가 내장된 전자 제어 시스템을 의미한다. 또한 단순한 회로만으로 구성된 장치가 아닌 마이크로프로세서가 내장되어 원하는 작업을 수행, 관리하는 프로그램이 포함된 시스템이다.[1] 이러한 임베디드 시스템은 정보 기술의 발달과 함께 기존의 전자 기기 들을 인터넷에 연결하고자 하는 욕구를 창출하게 되었다. 최근에는 Embedded Internet이라 하여 임베디드 시스템이 인터넷에 연결되어 센서와 구동장치 등이 네트워크로 상호 연동되어질 뿐만 아니라 원격지에서 각종 장비를 제어 및 감시 할 수 있게 되었다.[2]

현재(2001년 말 기준) 우리 나라의 이동전화 가입자수는 2,905만 명으로 집계되고 있고 이 수치는 인구 대비 보급률이 61.4%를 달성하는 등 이동 통신에 대한 비중이 점차 증가하고 있다.[3] 이러한 많은 이동전화가입자의 단말기를 기지국(Base station Transceiver Subsystem)이 모두 커버 할 수 없으며 기지국의 신호를 재 증폭해주는 중계기(Cell Enhancer)의 역할이 매우 중요하다. 뿐만 아니라 중계기의 상태에 따라 이동전화 가입자들의 통화 품질이 좌우되고 있으며 각 통신사들은 중계기의 효율적인 모니터링 방안을 모색하고 있다.[4]

본 논문에서는 효율적인 CDMA중계기 관리를 위해서 Embedded Internet에 연결되어진 임베디드 리눅스 시스템과 연동 하여 중계기의 제어 및 모니터링 할 수 있는 리눅스 커널 레벨의 방안을 제시한다. 이 방안은 기존 중계기 모니터링 시스템의 오버헤드와 같은 성능저하 문제를 해결하며 인터넷망에서 중계기 데이터를 모니터링

및 제어함으로써 타 시스템과 연동 할 수 있는 확장성을 가진다.

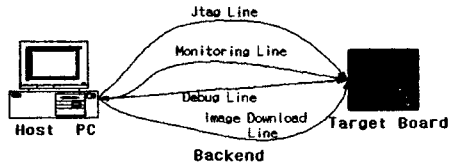
본 논문의 구성으로는 2절에서는 관련 연구로 임베디드 리눅스 시스템과 리눅스 커널 그리고 CDMA(Code Division Multiple Access) 중계기에 대해 설명하고 3절에서는 본 논문에서 제시하는 EKMM의 전체 시스템의 설계와 구현에 관하여 논하며 4절에서는 결론 및 향후 연구 과제에 대하여 기술한다.

2. 관련 연구

2.1 Embedded Linux System

Windows CE, VxWorks, uC/OS등의 임베디드 시스템의 운영체제에 비해 리눅스가 주목받는 이유는 모든 소스가 공개되어 있으며 GNU(Gnu is Not Unix)개발 도구를 이용하여 커널을 포함한 다른 응용 프로그램 개발이 가능하고 이식성이 뛰어나기 때문이다. 이와 같이 다양한 점에서 주목은 받고 있으나 임베디드 시스템으로써는 주 메모리와 파일 시스템 기술, 휴대용 시스템에서의 전원문제 등의 제한을 가지고 있다. 이러한 부분은 내장형 시스템의 설계와 응용 프로그램 제작 시 많은 부분에서 고려되어야 한다.[5]

임베디드 리눅스 시스템을 구현하기 위해 선행되어야 할 과정은 Target Board의 자원 한계를 극복하기 위한 CDE(Cross Development Environment)를 구축해야 한다. CDE는 Boot loader와 유기적으로 작용하게 되며 이것은 하드웨어 초기화, 커널 이미지 다운로드 그리고 Flash 메모리 기록, 삭제, 수정 등의 기능을 TFTP(Trivial File Transfer Protocol)와 BOOTP(BOOT strap Protocol)와 연동 하여 처리하게 된다.



[그림 2-1] CDE(Cross Development Environment)

2.2 Linux Kernel

리눅스 커널은 자원 관리자로서 CPU, 메모리 등과 같은 물리적 자원과 태스크, 세그먼트와 같은 추상적인 자원을 관리하는 태스크 관리자, 메모리 관리자, 파일 시스템, 네트워크 관리자 그리고 디바이스 드라이버로 구성되며 서로 유기적으로 동작된다. 리눅스 커널은 하나의 커널에 필요한 모든 기능이 통합되어 있던 Monolithic Kernel 형태가 아닌 제한적으로 작은 서버 모듈로 나누어진 Micro Kernel 형태의 모듈을 사용하고 있다. 이러한 모듈은 로딩 방식에 따라 커널 부팅 시에 로딩 되는 정적 로딩(Static Loading) 방식과 커널 실행 도중에 로딩 가능하여 기능 확장성이 뛰어난 동적 로딩(Dynamic Loading) 방식으로 나누어진다.[6]

2.3 CDMA 중계기

기지국의 RF(Radio Frequency)신호를 제3의 전송 매체를 통하여 원하는 지역에 전송하고 다시 RF신호로 재생하여 기지국 신호를 재 증폭해 주는 중계기는 HPA(High Power Amplifier)출력상태, H/W 지연 상태, 중계기 출력 등의 요소를 고려하여야 하며 시스템 통화 용량과 기계 내부적인 고장 등의 원인에 의해 여러 형태의 문제가 발생할 수 있으며 잡음의 원인과 문제들의 대처 방안에 대해 많이 연구되어 왔다.[7]

본 논문에서는 세 부분으로 나누어진 중계기를 [표 2-1]와 같이 각 상태 Item으로 정의하고 이를 EKMM 시스템에서 모니터링 및 제어하게 된다.

[표 2-1] 중계기의 부분별 상태 ITEM

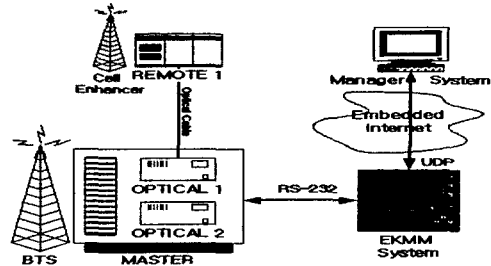
Part	Status Item
MASTER	장비업체, 종류, 버전, 전원, OPT구성, RF POWER Alarm, +5V V/I Sensing Signal 등
OPTICAL	상태, OPT Alarm, 환경상태, Forward & Reverse ATT, LD Current & Power 등
REMOTE	장비업체, 종류, 버전, 전원, HPA V Sensing Signal, Temperature, TX RF Power 등

MASTER : 전원의 전압이나 전류 등을 감시하는 상태를 정의
 OPTICAL : Optical Unit 자체를 감시하는 상태를 정의
 REMOTE : Remote 중계기 자체를 감시하는 상태를 정의

3. EKMM(Embedded internet based Kernel level Monitoring Modules)

3.1 전체 시스템 구조

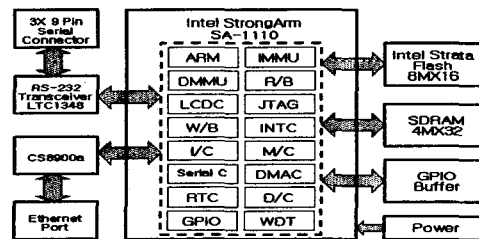
본 논문에서 제안하는 시스템의 전체 시스템 구조는



[그림 3-1] EKMM System의 구성도

[그림3-1]과 같이 나타낼 수 있다. 기지국의 MASTER내의 OPTICAL과 REMOTE가 1:1로 대응되어 있다. 이 MASTER에서 통합적인 상태 정보를 가지며 EKMM 시스템과 연결되게 된다. 이러한 구성은 Embedded Internet에 있는 모니터링 관리 시스템에서 모니터링 할 수 있는 구조로 구성된다. Embedded Internet 하에서 EKMM 시스템과 모니터링 관리 시스템 사이의 데이터는 ID, TYPE, LEN 필드로서 6 Byte의 헤더와 모니터링 데이터 또는 제어 명령어로 구성된다. 이것은 최대 256 Byte의 특성화 된 데이터와 명령어 집합으로 이루어진 비 연결형 프로토콜인 UDP(User Datagram Protocol)로 구현함으로써 최소한의 오버헤드만을 허용하게 하였다.

3.2 EKMM을 위한 하드웨어 구성



[그림 3-2] EKMM을 위한 하드웨어 구성

본 논문에서 사용되어진 임베디드 시스템의 하드웨어 구성은 [그림3-2]와 같이 중계기와 연결될 시리얼 포트와 Embedded Internet과 연결되는 이더넷 포트, 그리고 SA-1110[8]의 기능 확장 용도의 GPIO(General Purpose I/O) 인터페이스를 가진다.

SA-1110의 시리얼 포트는 16550 프로토콜을 지원하는 Full Duplex의 UART(Universal Asynchronous Receiver Transmitter)이며, 56.25bps에서 230.4kbps의 전송 속도를 지원한다. 또한 이더넷 컨트롤러는 CIRRUS LOGIC

[표 3-1] Flash Memory의 메모리 맵

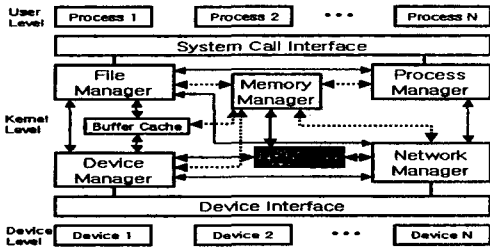
물리 주소	가상 주소	용도
0x00000000	0x50000000	Boot Loader
0x00030000	0x50030000	Boot Script
0x00040000	0x50040000	Kernel Image
0x00100000	0x50200000	Root File System Image

의 Full Duplex인 CS8900칩을 사용한다.

EKMM 시스템의 마이크로 프로세서인 SA-1110에 맞게 리눅스 커널의 하드웨어 의존적인 부분을 처리하고 리눅스 커널을 최소화하여 재구성해 주었으며 커널이 올라갈 최소한의 파일시스템을 위해 [표 3-1]과 같이 커널 내의 플래쉬 메모리 맵을 구성하였다.

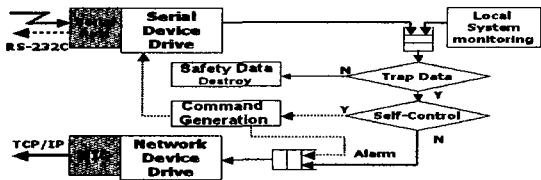
3.3 리눅스 커널 내의 EKMM

본 논문에서 구현하는 EKMM은 리눅스 커널 모듈 형태로 구현되었다. 커널 영역에서 동작함으로써 유저 영역에서 구동하는 타 시스템에 비해 모드 체인지로 발생하는 오버헤드로 인한 지연이 발생하지 않아 빠른 모니터링과 제어가 이루어진다.

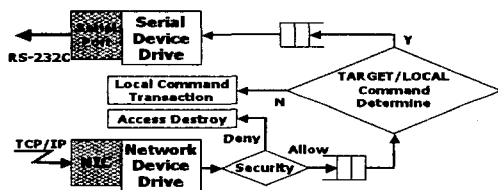


[그림 3-3] EKMM이 로딩된 상태의 커널 구조

EKMM은 커널 부팅 시나 커널 실행 도중에 로딩이 가능한 동적 로딩 방식의 모듈로 커널 코드의 확장성과 재사용성의 한계를 극복하였다. [그림3-3]은 EKMM이 커널에 로딩되어 디바이스 관리자와 네트워크 관리자 그리고 메모리 관리자간의 유기적인 상태를 나타낸다.



[그림 3-4] EKMM의 Trap Flow



[그림 3-5] EKMM의 Command Flow

EKMM 시스템의 내부적인 알고리즘은 [그림3-4]와 [그림3-5]에서와 같이 Trap과 Command 흐름으로 도식화하였다. Trap 흐름에서 Self-Control은 내부에서 중계기의 Trap 상태이긴 하나 자체 처리 가능한 상태의 데이터인 경우에는 Self-Control로 흐름을 바꾸어 그에 따른 제어 명령을 발생하여 중계기를 제어한다. 이 Self-Control은 선 처리, 후 보고 방식으로 구동하게 된

다. 이러한 방식은 자체 Command에 의한 처리와 적절한 보고 방법인 Alarm정보를 거의 동시에 중계기와 모니터링 관리 시스템에 전송하여 현재 문제 상황을 처리 및 보고하여 신속한 상황대처 능력을 가진다.

Command 흐름에서는 모니터링 관리 시스템에서 특정 명령 또는 현재 중계기 상태를 Embedded Internet으로 전송 요청 및 제어 명령을 보내어 EKMM 시스템과 중계기를 제어한다. 또한 Embedded Internet망에 연결된 EKMM에 불법적인 접근을 막기 위한 방안으로 네트워크 디바이스로 들어오는 신뢰성 없는 접근에 대해 차단하도록 하였다.

SA-1110 마이크로프로세서의 UART와 연결된 문자 디바이스인 시리얼 디바이스와 CS8900과 연결된 네트워크 디바이스의 데이터 손실을 최대한 막기 위해 일정 시간 후에 작업을 실행할 수 있도록 하는 후반부 작업(Bottom Halves)으로 스케줄 하였다. 또한 임베디드 시스템의 자원 사용의 제약을 고려하여 각 디바이스에서 일출력되는 데이터의 저장을 위한 공간으로, 동기화가 필요한 유한 버퍼(Bounded-Queue)형태의 원형 큐로 구현함으로써 메모리를 효율적으로 사용하였다.

4. 결론 및 향후 연구 과제

본 논문에서는 많은 이동전화 가입자들의 통화 품질과 밀접한 관계가 있는 중계기를 Embedded Internet하에 임베디드 시스템과 연동 하여 커널 영역에서 고속의 모니터링 방안을 제시하였다. 기존의 중계기 모니터링 시스템의 성능을 산출한 데이터가 없어 비교 할 수는 없지만 기존 시스템이 유저 영역에서 구현되어 커널 영역의 데이터가 유저 영역으로 모드 체인지 할 때 오버헤드가 발생하는데 비해 본 논문의 EKMM 시스템은 커널 영역에서 구현되어 모드 체인지로 인한 오버헤드를 줄였다. 또한 모듈화 된 동적 로드 방식을 취하므로 기능 확장성이 뛰어나다. 또한 임베디드 시스템의 제약점 중 하나인 메모리 사용측면에서 커널 내에서 Bounded-Queue인 원형 큐를 사용함으로써 임베디드 시스템에서 효율적인 메모리 사용방안을 제시한다. 뿐만 아니라 인터넷 환경 즉, Embedded Internet에 연결되어 모니터링과 제어를 함으로서 타 시스템과 연동 할 수 있는 확장성을 가진다.

향후 연구 과제로는 인터넷망에 EKMM 시스템이 노출된 상태이므로 본 EKMM의 소극적인 보안 대책인 단순한 접근 거부 방식의 보안이 아닌 좀더 적극적인 보안에 대한 연구가 필요하다.

5. 참고 문헌

- [1] Michael Barr, "Programming Embedded System in C and C++", O'Reilly, 1997.
- [2] Deborah Estrin, Ramesh Govindan, and John Heidemann. Embedding the Internet. Communications of the ACM, 43 (5), pp. 39-41, May, 2000.
- [3] 산업자원부, 2001수출입 실적(확정), 2002.2.28
- [4] Sang Gun. Lee, Hyp Chang. Bang, "CDMA Wireless Technique", Saehwa, 2000.
- [5] Linux@Work Magazine, "Embedded Linux System" pp114~150, May 2001.
- [6] Gary Nutt, "Kernel projects for Linux", Addison Wesley Longman, 2001.
- [7] Jay jayspalan and Mike Buke, "Cellular Data Service Architecture and Signalling, IEEE Personal Communications", 1994.
- [8] Intel Corporation, "Intel StrongARM SA-1110 MicroProcessor Developer's Manual", 2000.