

ARM 기반의 SOC MCU를 이용한 SOHO 라우터 용의 임베디드 리눅스 시스템의 개발

김운기⁰ 최승현 김홍석 강성봉 장동은
삼성전자, 시스템LSI, Sys. Solution
{foriver⁰,sh428.choi, kimhans.kim, osbkang, chirsd.chang}@samsung.com
윤희용
성균관대학교, 정보통신 공학부
youn@ece.skku.ac.kr

Development of Embedded Linux System on ARM SOC MCU for SOHO Router

Woon Gi Kim, Seung Hyun Choi, Hans Kim, Sung Bong Kang, Chris Chang
System LSI Business, Samsung Electronics Co.,Ltd
Hee Yong Youn
School of Information and Communication Engineering
Sung kyun kwan University

요 약

본 논문에서는 SOHO 라우터와 같은 네트워크 응용 목적의 임베디드 시스템에 리눅스를 성공적으로 적용하였다. 본 논문에서는 임베디드 리눅스 시스템의 개발에 있어 S3C2510과 같은 ARM 계열의 SOC MCU의 특성을 충분히 이용하고, 리눅스의 강력한 네트워크 프로토콜을 적절히 활용하였다. 본 논문에서는 리눅스와 다른 상용 RTOS의 네트워크 라우팅 성능을 비교 측정함으로써 리눅스의 성능을 검증하였다.

1. 서 론

임베디드 OS 시장에서 리눅스는 점점 그 시장을 확대하고 있다. 임베디드 시스템에서 리눅스를 사용하게 되는 주된 이유는 리눅스의 안정성 이외에도 다양한 프로세서에 대해 이미 이식이 완료되어 있으며, GPL 및 LGPL등과 같은 간편한 라이선스와 소스가 공개되어 있어서 문제 해결이 용이하다는 점, 그리고 강력한 네트워킹 프로토콜의 제공, 모듈 기능의 지원을 통해 소프트웨어를 효율적으로 구성할 수 있고, 그래픽 처리 등의 많은 API(Application Program Interface)를 제공하고 있는 점이다. 또한 다양한 Open Source Community에 의한 공동 개발 프로젝트를 통해 소스를 구하고 문제 해결을 할 수 있다는 점 또한 강점이다. 그리고 리눅스를 사용하여 독자적인 소프트웨어를 확보함으로써 독점적인 RTOS 공급자로부터 독립하고자 하는 개발 업체들의 노력도 리눅스가 임베디드 시장에서 점점 많이 채택되고 있는 이유이다.

특히, 네트워크 용의 임베디드 시스템에서는 리눅스의 강력하고 안정적인 네트워크 프로토콜을 이용할 수 있다는 점에서 더욱 강점이 있다. 이것은 상용 임베디드 OS에 비해 비교적 열세인 그래픽 관련 응용이 없고

강력한 리눅스의 기능적인 측면을 이용할 수 있기 때문이다.

본 논문에서는 리눅스를 SOHO (Small Office Home Office)용의 라우터 개발에 적용하였다. 특히, 리눅스의 강력한 네트워크 프로토콜들을 이용하고 응용 프로그램은 GPL을 따르는 리눅스 상의 공개된 응용 프로그램들을 사용하였다.

본 논문에서 개발에 사용한 시스템은 삼성의 SMDK2510이다. SMDK2510은 S3C2510 SOC MCU와 네트워크 응용에 필요한 주변 장치들로 구성된 개발 키트(Kit)이다. 삼성의 S3C2510 MCU는 ARM940T를 기반으로 Ethernet MAC등을 내장하고 기타 주변 장치들을 갖추어서 SOHO 라우터에 적합하도록 개발된 네트워크용 마이크로 프로세서이다.

본 논문에서는 S3C2510을 이용하여 SOHO 라우터를 개발함에 있어서 MMU가 없는 CPU를 대상으로 변화한 리눅스의 일종인 uClinux를 사용하였다. uClinux를 이식함에 있어서 S3C2510에 내장된 각 기능 블록을 적절히 사용하였다. 그리고 임베디드 시스템에서의 리눅스의 성능을 확인하기 위하여 라우팅 Throughput 등의 네트워크 성능을 측정하였다.

2. SOHO 라우터를 위한 리눅스의 네트워크 프로토콜
본 논문에서 구현하고자 하는 SOHO 라우터를 위해서는
다음과 같은 네트워크 프로토콜이 필요하다.

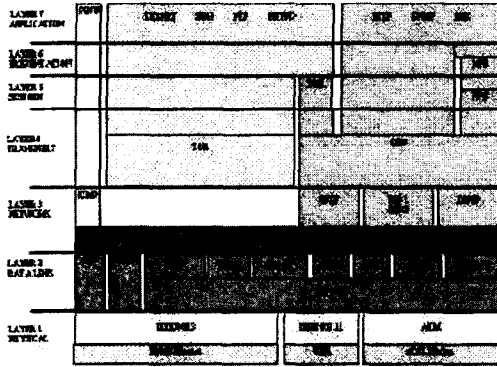


그림 1. SOHO 라우터의 프로토콜 스택

3. SOHO 라우터 시스템의 개발

삼성의 S3C2510은 ARM940T를 기반으로 저비용, 고효율을 지향하는 RISC 마이크로 컨트롤러이다. Ethernet 등의 내장된 네트워크용 주변 장치들은 간단하게 SOHO 라우터, 인터넷 게이트웨이 등을 구현할 수 있게 하고, 역시 내장된 PCI/PCMCIA/CardBus 등의 확장 가능한 버스 기능을 제공함으로써 WLAN AP (Access Point) 등을 구현 가능하도록 지원하고 있다. 또한 USB host 기능을 제공함으로써 네트워크 프린터 서버 등의 다양한 솔루션을 구축할 수 있다. 다양한 네트워크 응용에 대해 빠른 성능을 지원할 수 있도록 내장된 4Kbyte D-Cache 및 4Kbyte I-Cache, 메모리 컨트롤러, 6 채널의 DMA 기능 등을 제공하며, 최대 CPU 동작 속도 166MHz, 버스 동작 속도 133MHz의 고성능 RISC 마이크로 컨트롤러이다.

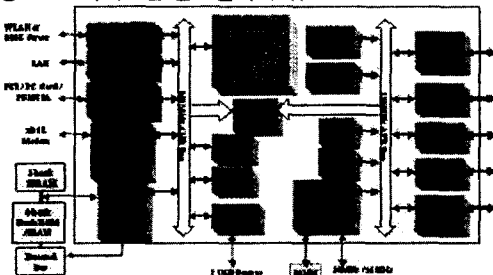


그림 2. S3C2510의 구성

본 논문에서는 S3C2510을 이용하여 제작한 SMDK2510 개발 키트를 사용하였다. SMDK2510은 삼성의 네트워크 분야의 응용에 대비한 프로세서인 S3C2510을 이용한 제품 개발용 보드이다. S3C2510 마이크로 컨트롤러는 ARM940T 코어를 사용하였고 SOHO router, Internet gateway, WLAN AP 등과 같은 Ethernet 기반의 응용 제품 개발에 유용한 기능들을 다수 포함하고 있다.

개발자는 이 보드를 참조하여 S3C2510을 이용한 하드웨어 디자인 방법을 습득할 수 있으며 제공되는 소스를 응용하여 보다 빨리 프로그램을 작성할 수 있다. SMDK2510 보드는 S3C2510, Boot ROM, SDRAM, USB, Ethernet, UART, JTAG 등으로 구성되어 있다.

4. 시스템 메모리의 구성

SMDK2510 개발 보드에는 2M의 Flash 메모리가 0x0번지에 있고, 32M의 SDRAM이 0x80000000번지에 할당되어 있다. 시스템에 전원이 들어오면 S3C2510 CPU는 Flash 메모리의 0x0번지에 저장되어 있는 Bootloader를 수행함으로써 부팅을 시작한다. Bootloader는 수행 속도의 향상을 위해 Flash 메모리로부터 SDRAM으로 실행 코드를 복사하고 메모리 Remap을 통해 SDRAM을 0x0번지에 할당함으로써 SDRAM에서 동작한다. 그 이후 Flash 메모리의 CRAMFS 이미지를 해석하여 리눅스 커널을 SDRAM에 복사하고 리눅스를 동작시킨다. Flash 메모리 상의 이미지는 계속 존재하며 응용 프로그램과 관련 설정들을 보유하여 리눅스의 루트 디렉토리로 마운트 된다.

이상과 같은 동작의 지원을 위해 시스템의 메모리 다음과 같이 구성된다.

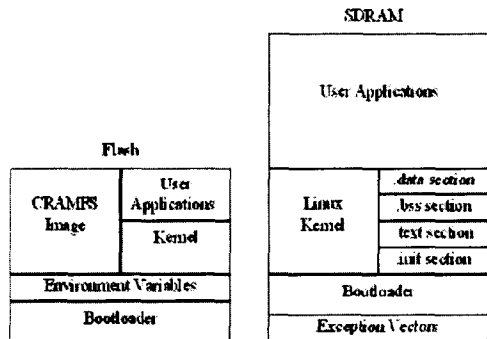


그림 3. 시스템 메모리 구성

5. uClinux 커널의 특징

본 논문에서 사용한 S3C2510 CPU는 MMU가 없으므로 PC 환경에서의 리눅스를 이식할 수 없다. 그래서 MMU 없는 CPU에 대한 리눅스의 변형 버전인 uClinux를 사용하였다. MMU를 사용할 수 없다는 것은 가상메모리의 사용이 불가능하다는 것을 의미한다. 이로 인해서 uClinux에서는 다음과 같은 제약사항이 있다. 가상 메모리를 사용할 수 없으므로 모든 응용 프로그램은 실행을 위한 자신의 메모리 로딩 주소를 알 수 없으므로 PIC(Position Independent Code)로 컴파일 되어야 한다. 새로운 프로세스를 생성할 때 fork 함수는 자신을 복제할 때 동일한 메모리 주소로 복제하므로 fork 함수를 사용할 수 없다. 대신 vfork 함수를 사용해야 한다. 즉, fork에서와 같이 자신과 같은 메모리 주소를 갖는 프로세스를 복제할 수 없고, 자식 프로세스를 만들고 부모 프로세스를 봉쇄하며 서로 메모리를 공유하는 vfork를 사용해야 한다. 또한 실행 시 링크하는 공유 라이브러리를 사용할 수 없다.

그리고 스택의 크기를 동적으로 증가시킬 수 없으며 메모리 보호를 받을 수 없다.

uClinux는 모토롤라 m68K 계열의 CPU를 대상으로 개발된 후 현재는 다양한 프로세서에 이식되어 사용되고 있고 특히, ARM 프로세서에 그 적용이 많다.

6. 성능 측정

SOHO 라우터로서의 성능 확인을 위해 네트워크 라우팅 성능을 측정하였다. 라우팅은 S3C2510의 eth0와 eth1에 각각 연결된 PC간의 Routing에 관한 측정이다. 본 논문에서는 Smartbit 장비를 이용하여 S3C2510의 routing 성능을 측정하였다. 즉 eth0로 들어온 패킷이 라우팅 과정을 거쳐서 eth1으로 전송되는 속도를 측정하였다. Smartbit 장비를 UDP 전송 프로토콜로 설정하고 loss 없는 상태의 속도를 측정하였다.

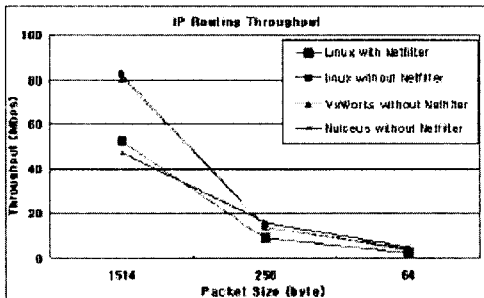


그림 4. IP 라우팅 성능 측정 결과

본 논문의 Target 시스템의 용도는 SOHO 라우터이다. 라우터로서의 가장 중요한 성능인 IP 라우팅 성능을 측정하였다. 리눅스를 사용한 경우의 성능을 확인하기 위해서 다른 상용 OS를 사용한 경우와 결과를 비교하였다.

Smartbit 장비를 이용하여 리눅스와 VxWorks 그리고 Nucleus OS에 대해 패킷의 크기를 64 byte, 256 byte 및 1514byte로 고정하고 각 OS 별 라우팅 성능을 측정하였다. 결과 리눅스의 네트워크 라우팅 성능이 다른 OS에 비해 비슷하거나 우수한 것으로 나타났다.

7. 결론 및 향후 계획

임베디드 OS 시장에서 리눅스는 GPL로 표현되는 간편한 라이선스와 안정적이고 강력한 성능을 바탕으로 점점 그 적용 범위를 넓혀 가고 있다. 본 논문에서는 SOHO 라우터와 같은 네트워크 응용 목적의 임베디드 시스템에 리눅스를 성공적으로 적용하였다. 본 논문에서는 임베디드 리눅스 시스템의 개발에 있어 S3C2510과 같은 ARM 계열의 SOC MCU의 특성을 충분히 이용하고, 리눅스의 강력한 네트워크 프로토콜을 적절히 활용하였다.

본 논문에서는 네트워크 라우터의 개발에 있어서 MMU가 없는 CPU를 대상으로 수정된 리눅스의 일종인 uClinux를 적용하였다. 임베디드 시스템은 리소스의 제한과 특정 응용에 특화되는 특징이 있다. 이러한 환경에서 uClinux는

성공적으로 S3C2510과 같은 ARM SOC MCU에 이식되었고 SOHO 라우터로서의 충분한 성능을 보여 주었다. 본 논문에서는 리눅스와 다른 상용 RTOS의 네트워크 라우팅 성능을 비교 측정함으로써 리눅스의 성능을 검증하였다.

앞으로 계속해서 최적화 작업을 진행하고 동시에 본 임베디드 리눅스 시스템에 대하여 다양한 기능과 응용 프로그램들을 평가할 계획이다.

7. REFERENCES

- [1] S3C2510 User Manual
Samsung Electronics
<http://www.samsungelectronics.com/semiconductors/>
- [2] ARM 940T manual
ARM
<http://www.arm.com/>
- [3] uClinux kernel
<http://www.uclinux.org>
- [4] gcc cross tool chains
<http://gcc.gnu.org/install/>
- [5] Issues for making Linux predictable
Nakajima, T.; Iwasaki, M.; Ochiai, S.
Applications and the Internet (SAINT) Workshops, 2002. Proceedings. 2002 Symposium on, 2002
Page(s): 8 -14
- [6] Analysis and evaluation of the TCP/IP protocol stack of Linux
Guo Chuanxiang; Zheng Shaoren
Communication Technology Proceedings, 2000. WCC - ICCT 2000. International Conference on , Volume: 1, 2000
Page(s): 444 -453 vol.1
- [7] <http://www.linuxdevices.com>
- [8] Enhancing the real-time capability of the Linux kernel
Yu-Chung Wang; Kwei-Jay Lin
Real-Time Computing Systems and Applications, 1998. Proceedings. Fifth International Conference on , 1998
- [9] <http://www.mvista.com>
- [10] <http://www.aero.polimi.it/~rtai/>
- [11] Linux Device Driver 2nd.
Alessandro Rubini, Jonathan Corbet.
O'Reilly.
- [12] <http://www.arm.linux.org.uk/>
- [13] How to really measure real-time
Vik Sohal
Embedded Systems Conference
<http://www.esconline.com/>
- [14] Performance Analysis of VxWorks and RTLinux
Benjamin Ip
COMS W4995-2, Languages of Embedded Systems
- [15] Adaptive Domain Environment for Operating Systems
Karim Yaghmour.
Opersys inc.
<http://www.opersys.com>
- [16] Linux Trace Toolkit
<http://www.opersys.com/LTT/>
- [17] Resource Management of the OS Network Subsystem
Sourav Ghosh and Raganathan (Raj) Rajkumar
Real-time and Multimedia Systems Laboratory
Department of Electrical and Computer Engineering
Carnegie Mellon University