

# 임베디드 리눅스를 이용한 IPv6 라우터의 설계에 관한 연구

류재훈\* · 김정태\* · 류광렬\*

\*목원대학교 IT공학과

## The IPv6 Router Design on Embedded Linux

Jae-Hoon Ryu\* · Jung-Tae Kim\* · Kwang-Ryol Ryu\*

\*Mokwon University

E-mail : ryol@mokwon.ac.kr

### 요 약

본 논문은 프로세서가 내장된 임베디드 리눅스 툴킷을 이용하여 IP 패킷을 기존의 IPv4 네트워크에서 IPv6 네트워크로 변환해주는 라우터 설계에 관한 연구이다. 주소 변환 플랫폼으로 프로세서를 이용하여 IPv6 모듈을 리눅스에 이식하였으며, IPv4 네트워크와 IPv6 네트워크를 구성하여 실험하였다. 시험망 구축을 위하여 기존 IPv4망과 IPv6망의 터널링 기법으로 구성된 결과, ICMP ping6을 이용 패킷 응답값은 2 홉스에 평균 2 $\mu$ sec정도이다.

### ABSTRACT

The design of router that converts IP packets from IPv4 network to IPv6 network using embedded Linux toolkit based on processor is presented. As an address transition platform, IPv6 module is transplanted to Linux using processor and the experiment was done with IPv4 and IPv6. In order to build the test network, it is constructed with Tunneling mechanism of IPv4 and IPv6 network. The packet value is obtained about 2 $\mu$ sec on average a 2 hops on the ICMP ping6.

### 키워드

Embedded Linux, IPv6, Router

### 1. 서 론

최근 인터넷의 활용이 극대화됨에 따라 IP주소가 고갈되었다. 따라서 현재의 IPv4 만으로는 급격히 발달하는 IT분야의 수요를 충족해주지 못한다. IPv6는 현재의 주소체계를 대체할 차세대 IP 주소로써 빠르게 전환될 전망이다[1]. 앞으로 각 가정의 가전제품에서부터 도시 전체에 이르기까지 IP주소가 할당되어 IPv6 주소 체계를 사용할 수밖에 없을 것이다. 이러한 IP 주소 체계의 전환을 준비하기 위해 세계 각국은 IPv6 도입 기술 및 망 구축 작업을 진행 중이다. 미국과 캐나다의 인터넷2 프로젝트, 캐나다 Canarie 중심의 CA\*net3 IPv6 프로젝트 일본의 WIDE, KAME 프로젝트, 유럽과 중국도 정부차원으로 IPv6 도입을 적극 추진 중에 있다. 우리나라도 2000년부터 MIC 주관의 “정보통신 선도기반 기술 개발 사업”중 “IPv4/IPv6 차세대 인터넷 주소변환기” 과제를 3년 동안 산업체와 공동

으로 수행 중에 있다.

이러한 새로운 주소체계를 사용하는 시스템은 기존 주소체계를 사용하는 시스템이 혼재된 인터넷 망과의 호환성 문제를 크나큰 부담으로 작용할 수 있다. 따라서 효율적이고 저렴한 주소체계 변환기가 필요하게 되고 이러한 플랫폼을 구성하기 위한 시스템 커널로 리눅스를 기반으로 하는 기술이 각광을 받고 있다[2]. 리눅스는 커널 소스가 개방되어 있어 누구나 자유롭게 접근할 수 있어 WinCe, VxWorks등 다른 임베디드 운영체제보다 큰 이점으로 작용하고 있다[3-4]. 이러한 상황 속에서 일반 리눅스 커널을 기반으로 Real-Time 기능을 커널에 추가하는 RTLinux 연구 등 다양한 형태의 변형을 시키는 연구가 진행 중이다[5].

본 논문에서는 경제성 있는 IPv4/IPv6 인터넷 주소 변환을 위한 Embedded Linux IPv6 라우터 구현에 관해 연구한다. 하드웨어는 MPC860을 사

용하였고, 이 보드에 리눅스 커널을 이식한 후 주소 변환 기능을 추가한 다음 실제 타겟 시스템에 대한 실험망을 구성하고 테스트 한다.

## II. IPv6 라우터 시스템 설계

### 2-1. 라우터의 설계

임베디드 리눅스 IPv6 라우터는 그림 1과 같은 과정으로 설계된다. 라우터를 설계할 타겟보드를 선정한다. 선정된 H/W에 리눅스 부트 스트랩 로더와 커널을 이식하였다. IPv6 커널소스를 설계 목적에 맞게 수정 및 컴파일 한다. 라우터가 동작할 수 있는 실험망을 구성 하고, 설계한 라우터의 동작 테스트를 한다. 라우터의 설계 과정을 순서도를 나타내면 그림 1과 같다.



그림 4. 라우터 설계 순서도.

### 2-2. 타겟보드 구성

라우터 설계에 사용한 타겟보드의 사양은 그림 2에서와 같이 칩의 내부에 통신장치를 구현하기 용이하도록 HDLC (High-level Data Link Control), SCC (Serial Communication Control) 등의 통신용 모듈을 많이 지원하는 PowerPC 계열의 MPC860SR, 16비트 4Mbyte Flash Memory, 32비트의 64Mbyte SDRAM을 사용하고, 통신을 위한 4개의 HDLC 및 이더넷 지원을 위한 통신포트 (SCC1 ~ SCC4), 2개의 RS-232 통신용 SMC포트로 구성 하였다. 타겟보드를 초기화 하기 위한 코드는 먼저 CS0 8M 16bit Flash FF80,0000~FFFF,FFFF 영역의 번지를 액세스하여 실행하기 시작한다. cpu의 레지스터들과 보드 관련 setting에 관련된 parameter 값들은 Flash memory의 FFFC,0000 번지에 저장된다.

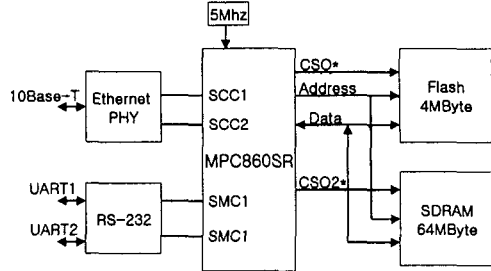


그림 5. 타겟보드 구성도

라우터 인터페이스 설계를 위해 Ethernet 포트는 타겟보드에서 제공하는 SCC1과 SCC2를 NI(Network Interface)로 사용한다. SCC2는 10/100Base-T 스위칭 허브를 연결하여 4개의 RJ45로 구성되었고, 단일 RJ45 잭으로 연결되는 SCC1을 사용한다. 그림 2는 SCC2를 나타내고, 표 1은 사용되는 포트 신호 및 코드이다.

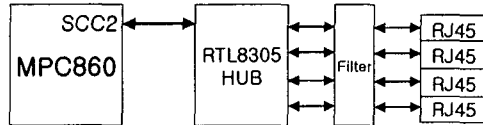


그림 6. SCC2 10/100Base-T 스위칭허브 구성도

표 1. SCC2 Port Signal

Signal	MPC860 Side	RTL8305
TX Data	PA12(TXD2)	TXD0
TX Clock	PA5(CLK3)	TXC
RX Data	PA12(RXD3)	RXD0
RX Clock	PA7(CLK1)	RXC
TX_Enable	PB18(RTS2*)	TXEN
RX_Enable	PC8(CD2*)	CRS
Collision	PC9(CTS2*)	COL

### 2-3. 리눅스 커널 및 부트 스트랩 로더 이식

타겟보드를 설계 목적에 맞게 선정된 후 부트 스트랩 로더와 리눅스 커널을 이식하기 위하여 그림 4와 같이 교차개발환경을 구축한다. 즉 타겟보드가 목표하는 동작을 하도록 부트 스트랩 로더의 개발, 리눅스 커널 컴파일, 파일 시스템 적재등의 작업을 호스트 컴퓨터에서 하고 이를 이식 및 퓨징 하기 위해 RS232와 네트워크 통신(TFTP, NFS 등)을 이용한다. 구성된 환경의 호스트에 개발 툴과 타겟보드의 BDM을 이용하여 PPCBOOT 부트 스트랩 로더를 이식 한다. PPCBOOT는 PowerPC계열 CPU에 사용되는 부트 스트랩 로더이며, 하드웨어 테스트와 초기화, 커널과 루트 파일시스템의 초기화, 커널 부팅에 필요한 각종 환경설정 등의 하드웨어, 소프트웨어를 사용하기위한 초기화를 수행한다.

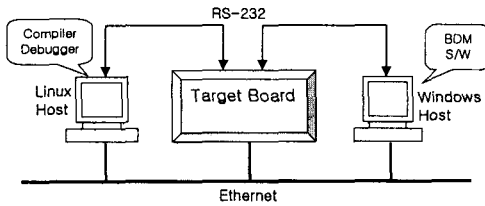


그림 7. 교차개발 구성도

타겟 시스템의 전원이 인가하면, 이식한 부트 스트랩 로더가 장치를 초기화 하며, 정상적으로 초기화 과정이 끝난 후 리눅스 커널(버전 2.4.2)을 tftp로부터 다운로드 받아 타겟보드에 이식하였다.

### 2.4. 수정 및 컴파일

이식한 리눅스 커널은 커널의 기능들을 다수의 모듈로 나누고 이 모듈들이 동적으로 삽입되거나 삭제될 수 있도록 하기 때문에 타겟의 요구에 따라 운영체제의 크기를 가변적으로 생성하고, 응용프로그램 구축까지 디스크 활용을 최대화해야 하므로 적절히 수정해야 한다. 리눅스 커널에서 소스에 대한 변경 및 컴파일은 ppchboot에 설정한 파일의 하부구조에 맞게 수정, 추가 한다. 크게 세분화 하면 arch/ppc/, drivers/, include/, 디렉토리에서 수정 및 추가를 하며 SCC2 이더넷을 지원, SCC1 및 SCC2 Port Assign 정보 설정, Ethernet 주소의 변경 등의 커널을 설정한다.

표 2. 초기화를 위한 소스코드 수정

Memory Map 구성	CS0, Flash, DRAM
include/config_MPC860.h	board parameter정의
board/MPC860/config.mk	커널이미지, initrd 수정
board/MPC860/MPC860.c	DRAM초기화
board/MPC860/flash.c	flash 초기화 코드
common/board.c	console 설정

부트 스트랩 로더 소스코드의 수정은 주로 arch/ppc/ 디렉토리를 주로 변경하며, 표 2. 는 소스코드 수정 의도와 그 디렉토리 경로이다. 임베디드 리눅스 라우터가 구성한 설정을 동작하기 위해 Ethernet Port와 Tunnel Interface, 그리고 IPv6 라우팅 테이블 엔트리를 설정 하였다. 설정을 위해 스크립트 파일을 수정하여 IPv6를 enable 시켰다. 실험 주소를 수정할 디렉토리의 파일 구성은 다음과 같다.

- ▶ /etc/sysconfig/network
- ▶ /etc/sysconfig/network-scripts/ifcfg-eth0
- ▶ /etc/sysconfig/network-scripts/ifcfg-eth1
- ▶ /etc/sysconfig/network-scripts/ifcfg-sit0
- ▶ /etc/sysconfig/network-scripts/ifcfg-sit1

▶ /etc/sysconfig/static-routes-ipv6

## III. 실험 및 고찰

### 3-1 실험망 구성 및 테스트

본 연구의 테스트 네트워크는 IPv4 망과 IPv6 Layer 2 구성한 후, 임베디드 라우터에서 Router1까지의 경로에 터널링 기법중 SIT를 사용하였으며, ICMP Ping6 실험으로 라우팅 성능을 확인하는 모델을 사용하였다.

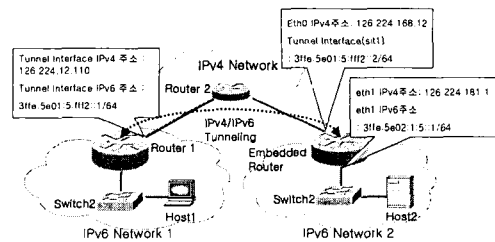


그림 8. IPv4/IPv6 실험망 구성도

Router1과 Router2는 Cisco 2600을, 스위칭허브는 Catalyst2950을 사용하였다. 터널링은 패킷을 해당 네트워크에서 운영되는 프로토콜 버전의 패킷으로 캡슐화하여 전송하는 기법이며, SIT는 IPv4 네트워크 안에서 IPv6를 지원하는 호스트 사이에서 이루어지는 Tunneling 기법이다. 그림 5에서와 같이 임베디드 라우터는 패킷을 IPv4 패킷으로 캡슐화하여 Router2의 IPv4 멀티캐스트를 이용하여 Router1에게 전송한다. Router1은 멀티캐스트 패킷을 받아 IPv6 패킷을 역캡슐화 하여 임베디드 라우터와 Router1 사이의 터널을 구성 하였다.

테스트 네트워크에 할당된 파라미터는 다음과 같다. 3ffe:5e02:15::/64 서브넷과 일반적으로 알려진 실험주소 3ffe::/16와 공식 주소 2000::/3에 대한 루트 정보를 추가 하였다. Longest Prefix Matching에 의해 위 실험주소 공식주소 서브넷에 매치하는 모든 주소가 sit1을 통해 라우팅 될 수 있도록 추가하였다. 라우팅 알고리즘은 Static routing을 사용하였으며, 마지막으로 IP 내의 ICMP인 Ping6을 통해 실험하였다. 세 개의 노드에 대해 Ping6으로 테스트를 하였으며, 그림 6은 Ping6의 실행 결과를 나타낸다.



(a)



(b)



(c)

- (a) 임베디드 라우터 자신의 IPv6주소(3ffe:5e01:5:fff2::2)로 실행
- (b) Router1의 IPv6주소(3ffe:5e01:5:fff2::1)로 실행
- (c) Router1에 터널 연결된 Host1의 IPv6주소(3ffe:5e01:7:fff2::3)를 실행

그림 6. 각 노드간 ping6 실행

(a)는 임베디드 라우터 자신의 주소로 ICMP 패킷을 전송해 SIT 주소의 인터페이스의 연결 상태를 확인 해주며, (b)는 터널로 연결된 Router1의 SIT주소의 인터페이스와 ping6 실행 결과 화면이다. (c)는 Router1이 관리하는 Host1에 패킷을 전송하여 실험 한 결과이다.

### 3-2 고찰

ping6 실험을 통해 임베디드 리눅스 IPv6 라우터는 ICMP 패킷을 IPv6 패킷 생성한 후 IPv4 패킷으로 캡슐화하여 멀티캐스팅 하는 걸 알수 있다. 10/100Mbps Ethernet 환경에서 IPv6망에서 IPv4 망을 거쳐 IPv6망으로 터널링 기법을 사용한 결과 Router1에 연결된 임의의 Host1에 ICMP ping6 테스트 수치는 2 hops에 평균 2.082μsec을 얻었다. 실험 수치는 일반적인 네트워크 망에서 직접 연결된 라우터들 사이에서 얻을 수 있는 값이다. 따라서 설계한 임베디드 라우터가 저비용의 IPv6라우터로 사용 가능하다.

## IV. 결론

본 논문에서는 타겟보드에 임베디드 리눅스를 이용하여 IPv4/IPv6 주소를 변환해주는 IPv6라우터를 설계하였다. 소규모 실험망에서 터널링 기법으로 구성하여 ICMP 패킷의 응답값을 얻어 동작을 테스트 하였다. 그리하여 기존의 IPv4라우터와 연동시켰을 때 정상 작동을 하고, 임베디드 리눅스를 사용하기 때문에 경제성을 제공한다. 앞으로 QoS, 필터링 정책, 동적 라우팅 알고리즘등을 적용한 IPv6 라우터의 설계 등 적합한 Internetworking 등의 연구가 요구된다.

## 참고 문헌

- [1] R. hinden, Nokia, S. Deering, "RFC 2373 - IP Version 6 Addressing Architecture" Cisco Systems, The Internet Society (1998), July 1998.
- [2] Nikolaos S. Voros "Hardware/Software Co-Design of Complex Embedded Systems" Design Automation for Embedded Systems, 2003 Kluwer Academic Publishers, 8, 5-49, 2003.
- [3] [http://kl에.org/KoreanDoc/html/2.4Kernel\\_Compile-KLD.html#toc22](http://kl에.org/KoreanDoc/html/2.4Kernel_Compile-KLD.html#toc22)
- [4] <http://www.bieringer.de/linux/IPv6/IPv6-HOWTO/>
- [5] 김홍남외 5명, "Embedded System Programming" ETRI, 생능출판사, 2003.