

RTOS를 이용한 MPC8260 하드웨어 환경에서 교육용 라우터 개발

김대구 정중수

안동대학교 정보통신공학과

shampu99@hanmail.net, jschung@andong.ac.kr

Development of Educational Router at MPC 8260 Hardware Environment using RTOS

Tae-gu Kim, Joong-Soo Chung

Andong National University

요 약

본 논문에서는 MPC8260 프로세서를 활용한 임베디드 시스템을 활용하여 교육용 라우터 개발을 수행하였다. 개발환경으로는 다양한 RTOS중 토네이도 2.0.2를 활용한 가장 안전한 vxworks를 사용하였다. 이때 호스트 컴퓨터를 서버로, MPC8260 프로세서를 활용한 임베디드 시스템을 클라이언트로 사용하여 서버/클라이언트간에는 RS-232-C로는 MPC8260 프로세서가 탑재된 보드의 상태 점검용으로 활용되고, 이더넷으로는 서버에서 클라이언트로 ftp로 컴파일된 소스의 다운로드 및 이더넷 상의 패킷 송, 수신시 기능 점검을 위해서 사용되었다.

1. 서 론

오늘날 인터넷의 백본망과 라우터의 급속한 발달과 더불어 수많은 응용 서비스들이 사용되고 있으며, 향후에도 더욱 더 다양한 망들을 통한 서비스가 요구되고 있다. 인터넷 서비스가 주로 유선망을 통해 전달되며, 이에따른 라우터가 폭발적으로 확장 추세이다.

따라서 본 논문에서는 라우터의 작동 원리를 대학에서 강의가 필요하며, 이를 위한 교육용 라우터를 개발을 수행하였다. 교육용 라우터를 개발은 MPC8260 프로세서를 활용한 임베디드 시스템(이후 MPC8260보드로 칭함)[1]을 활용하여 교육용 라우터 개발을 수행하였다. 개발환경으로는 다양한 RTOS중 토네이도 2.0.2[2]를 활용한 가장 안전한 vxworks[3]를 사용하였다. 이때 호스트 컴퓨터를 서버로, MPC8260 보드를 클라이언트로 사용하여 서버/클라이언트간에는 RS-232-C로는 MPC8260보드의 상태 점검 및 다운로드등의 명령수행용으로 활용되고, 이더넷으로는 서버에서 클라이언트로 ftp로 컴파일된 소스의 다운로드 및 이더넷 상의 라우팅 프로토콜에 관련된 패킷 송, 수신시 기능점검을 위해서 사용되었다.

개발된 소스의 기능 점검을 위해 이더넷 환경에서 네트워크 구성후 교육용 라우터 기능의 검증을 수행하였다. 이를 위한 교육용 라우터 기능을 C언어로 프로그래밍한 후 컴파일 된 .out파일을 VxWorks가 로딩 된 MPC8260보드에 로딩 하여 구동시켰다. 전체적인 시험 환경으로는 보드와 서버와 클라이언트를 허브를 통해 접속하고, 서버와 클라이언트 사이에 telnet이나 FTP 기능을 시험하면서 송, 수신되는 패킷의 상태를 점검하였다. 정상적인 프로그래밍 결과를 검증하기 위해서 라우터와 라우터간 프로토콜 분석장비[4,5]를 활용하여 패킷의 흐름을 분석할 수 있다. 교육용 라우터 기능 점검을 위해 RIP[6], OSPF[7], BGP[8] 기능의 소프트웨어를 교육환경에 적절하게 구현하였다.

2. 개발 플랫폼

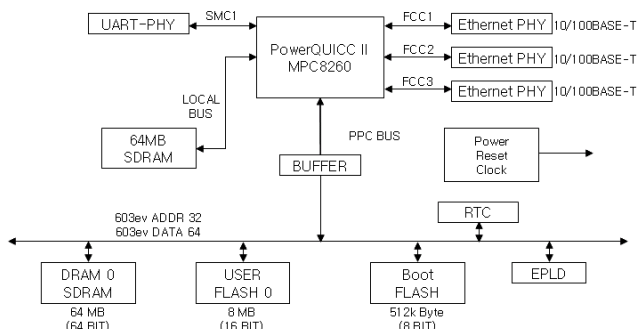
2.1 하드웨어 제원의 소개

하드웨어는 MPC8260보드를 사용하였으며, 이는 PowerPC 계열의 RISC 200MHz이상의 고성능 CPU인 MPC8260을 사용하고 있다. SDRAM은 주 메모리로 사용되며 두 개의 플래쉬 메모리를 사용하여 부팅 시 메모리에서 운영체제를 동작시키게 된다. 또한 3개의

패스트 이더넷(FCC)를 사용하며, 38400bps의 시리얼 콘솔을 통한 제어가 가능하다.

포팅하려는 RTOS(본 논문에서는 VxWorks)를 플래쉬 메모리에 적재를 한 후에는 보드 가운데에 위치한 점퍼를 설정하는 핀에 점퍼를 꽂아 구동을 시키면 적재한 VxWorks가 로딩되어 실행이 된다.

MPC8260보드의 구동 상황과 제어를 사용하기 위해서 CRT 프로그램을 사용하여 화면상으로 MPC8260보드의 구동 상황을 지켜보았다.



<그림 1. MPC8260 블록 다이어그램>

MPC8260보드에 VxWorks를 포팅하기 위해서는 VxWorks 정보가 담긴 Board Support Package(BSP)가 필요하며 본 논문에서는 vxWorks.st_rom.bin 파일로 작성된 BSP를 이용하여 8260 보드에 VxWorks를 포팅하였다. VxWorks 포팅 후 VxWorks 환경에서 해당 서버로부터 컴파일된 .out 파일을 FTP로 vxWorks.st_rom.bin와 연결되어 처리되도록 하였다.

2.2 호스트와 보드와의 포팅 시나리오

토네이도가 설치된 호스트에서 MPC8260보드까지 로딩 방법은 다음과 같다. MPC 8260보드에 VxWorks를 포팅하는 작업을 수행해야 한다. 이 때 자신의 ip주소와 서버로 접속할 서버의 ip주소를 설정하는 작업을 수행한다. 통신을 위한 ip설정이 모두 끝난 후 FTP서버를 열어둔다.(본 논문에서는 Cisco TFTP Server 프로그램을 사용하였다.)

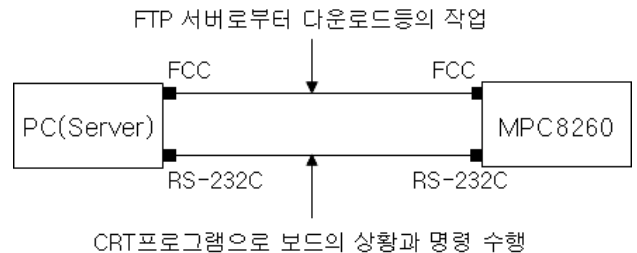
ip설정이 끝난 보드는 FTP 서버로부터 접속을 하여 VxWorks BSP를 다운받아 SDRAM에 적재하여 둔다. 그 후 erase명령어로 플래쉬 메모리를 모두 삭제 후 다운받은 BSP를 플래쉬 메모리에 설치하는 작업을 수행한다. 포팅 작업이 끝났으면 보드의 전원을 끄고 보드 가운데쯤에 위치한 점퍼 설정하는 핀에 점퍼를 꽂아 VxWorks가 포팅된 플래쉬 메모리만 로딩이 되도록 설

정한 뒤 이더넷 케이블을 그에 해당하는 FCC에 끼워둔다.

이렇게 VxWorks로 포팅된 보드에서 해당 서버와의 통신을 이용하기 위해 다시 FTP서버를 연다. 여기에서 여는 프로그램은 토네이도 설치 후 같이 들어 있는 토네이도 FTP서버 프로그램을 사용한다. 토네이도로 테스트에 쓰일 소스를 컴파일하여 나온 확장자 명 'out'파일을 FTP서버에 공유된 폴더에 복사하여 둔다.

보드의 상황과 제어를 위한 CRT 프로그램을 수행 후 보드에 전원을 켜다. VxWorks가 구동이 되면 설정 상태로 들어가 자신의 모듈의 IP와 접속하려는 서버의 IP를 설정하여 준다. 모든 설정이 끝난 후 재부팅을 하여 준다.

설정된 상태로 제대로 보드가 구동이 된 후 CRT의 커맨드 창에서 'ld < 가져올 파일이름' 명령을 사용하여 수행할 프로그램을 로딩하여 가져온다. 제대로 로딩이 되어 FTP서버로부터 파일을 수신받은 후에는 수행을 원하는 함수의 함수명을 그대로 역시 CRT의 커맨드 창에 타이핑 후 수행하면 프로그램이 구동이 된다.



<그림 2. 하드웨어 포팅 환경>

- 토네이도가 설치된 호스트에서 디버깅 방법

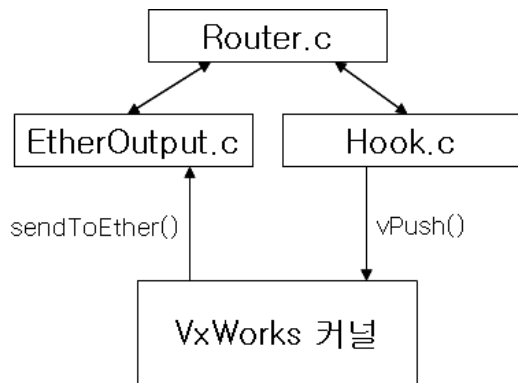
토네이도 환경에서 코딩이 된 소스를 컴파일 하여 나온 결과물인 .out파일을 보드에서 로딩 하여 구동한다. 보드와 인터넷이 연결된 일반 PC와 연결하여 테스트 환경을 구축한다. 테스트의 확인은 ping 패킷을 만들어 그것의 제대로 된 전달과 그 후의 작업 수행에 제대로 되는지를 시뮬레이션 하여 확인, 검증하였다. 버그나 에러의 발견 시 Tornado환경에서 수정 후 다시 보드에서 로딩 하여 테스트를 하였다.

3. 소프트웨어 개발

3.1. 소프트웨어 구조

본 논문에서 소개되는 교육용 라우터의 구동에 필요한

소스코드로는 그림 3과 같이 크게 3개가 있다.



<그림 3 교육용 라우터의 구동에 필요한 소프트웨어 구조>

- Hook.c : 이더넷으로 전송이 되는 패킷을 가져와 제어할 수 있도록 한다.
- EtherOutput.c : 이더넷으로 패킷을 보낼 시 사용이 된다.
- publicHeader.h : 공용으로 사용되는 변수들을 정리해두게 된다.
- Router.c: RIP, OSPF, BGP 기능의 소프트웨어를 교육환경에 적절하게 구현되었다.

이들 소프트웨어의 동작과정을 살펴보면 다음과 같다. 먼저 Router.c와 EtherOutput.c 및 Hook.c 는 sendToLan (char *buffer, int len)함수와 lanToReceive(char *buffer, int len)함수를 활용하고, EtherOutput.c 및 Hook.c와 VxWorks 커널과의 관계는 vPush(char *buffer, int len)와 sendToEther(char *buffer, int len)함수를 활용한다. 라우팅 프로토콜 관련 패킷 작성후 이를 이더넷으로 송신하기 위해서는 Router.c 파일의 sendToLan(char *buffer, int len)함수를 호출하고, 라우팅 프로토콜 관련 패킷을 이더넷으로부터 수신하는 Hook.c파일로부터 Router.c 파일의 lanToReceive(char *buffer, int len)함수를 호출한다. 작성후 이를 이더넷으로 송신하기 위해서는 EtherOutput.c 파일의 sendToLan(char *buffer, int len)함수를 호출한다. 이는 반드시 Hook.c나 EtherOutput.c의 vPush(char *buffer, int len)와 sendToEther(char *buffer, int len)함수를 통하여 vxWorks커널과 연결된다. 이더넷이나 함수호출시 매개변수 파라미터는 이더넷 프레임이나 라우팅관련 패

킷 정보 정보가 들어 있는 메모리리 시작주소를 *buffer 변수를 활용하고, 그 정보의 길이가 len이다.

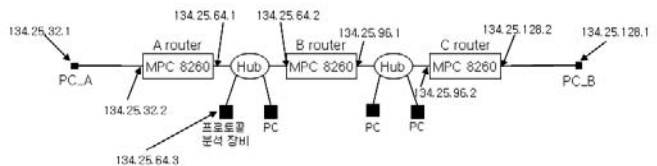
3.2. 외부 인터페이스

asTfpHook.c에서 사용자들이 사용할 수 있도록 이더넷에서 가져온 패킷을 넘겨준다. 그럼으로써 사용자들이 현재 수신된 패킷을 사용할 수 있도록 해준다. 사용자는 넘겨받은 패킷을 적절히 이용한 후 EtherOutput.c를 이용하여 이더넷으로 패킷을 전송할 수 있다.

4. 라우터의 네트워크 구성 및 기능 점검

4.1. 라우터의 네트워크 구성 및 점검

MPC8260 하드웨어는 3개의 이더넷 포트를 보유함으로 2개의 포트를 활용하여 그림 4와 같이 네트워크를 구축하여 기능 점검을 수행하였다.



<그림 4 라우터 기능 점검을 위한 네트워크 구축>

그림4에서 네트워크 구성시 클래스 B인 네트워크 주소 134.25.0.0 네트워크에 서브넷 마스크 3개를 할당하였다. 시험방법의 시나리오는 매우 다양하나 그중 한 경우를 생각하면, MPC8260 보드 세 개 모두 router.c에서 RIP를 구동하면 134.25.16.1인 PC_A에서 134.25.128.1로 ICMP를 활용한 ping 기능이 성공적으로 수행되어야 한다. 만일 한개의 MPC8260에는 router.c에서 RIP를 구동하고, 나머지는 ODPF로 구동되면 ping 기능이 성공적으로 수행되지 않아야 한다.

4.2 실습용 기능 점검

그림4와 같이 구축된 네트워크에서 RIP프로토콜 기능을 보유한 router.c가 컴파일된 결과를 MPC8260 보드에 로딩하여 사용자인 PC_A에서 PC_B로 ping 패킷을 전달하면 처리되도록 프로그램한 결과를 제시한후 실습생들은 그와 같은 결과를 도출하도록 router.c를 프로그래밍 하도록 하였다. 이때 정상적인 프로그래밍 결과를 검증하기 위해서 라우터와 라우터 간 프로토콜 분석장비를 활용하여 패킷의 흐름을 분

석할 수 있다. 즉, 보드 내에 Tornado에서 컴파일하여 나온 .out 파일을 로딩 시킨 후 인터넷과 연결하여 이더넷상으로 흐르는 패킷들을 제대로 잡아 Hook에서 가져와서 라우팅 기능을 제대로 처리하는지에 대한 테스트를 수행하였다. 그 결과는 4.1항과 동일하도록 프로그래밍하여야 한다.

5. 결론

본 논문에서는 MPC8260 프로세서가 탑재된 하드웨어 제원하에서 교육용 라우터 기능 점검을 위해 RIP, OSPF, BGP 기능의 소프트웨어를 교육환경에 적절하게 구현하였다. RTOS로는 vXworks를 개발환경으로 Torando2.2를 활용하였으며, 응용프로그램부와 vXworks 커널간은 Vpush, etherhook 두개의 프리미티브의 인터페이스로 router.c를 프로그래밍하였다.

기능 점검으로는 네트워크 구성시 클래스 B인 네트워크 주소 134.25.0.0 네트워크에 서브넷 마스크 3개를 할당하여 각 포트마다 ip 주소를 할당하여 수행하였다. 정상적인 프로그래밍 결과를 검증하기 위해서 라우터와 라우터간 프로토콜 분석장비를 활용하여 패킷의 흐름을 분석할 수 있었다.

향후에는 HMI(Human Machine Interface)기능을 보완하여 라우터 프로그램을 교육생이 작성시 보다 효율적이고 명확한 출력 결과가 디스플레이 되도록 보완적인 개발이 요구된다.

[참고 문헌]

- [1] MPC ELEX8260, Cuwave Company, 2002
- [2] Tornado 2.0.2
- [3] VxWork 5.4, "Windriver Company", 1984.
- [4] Sniffer Pro 4.50.04, 2000
- [5] Ethereal 0.10.10, 2005
- [6] rfc1058, "RIP", 1988
- [7] rfc1403, "OSPF", 1993
- [8] rfc1997, "BGP", 1996