

# Embedded OS용 인터넷 접속 지원 모듈의 개발

† 김세현<sup>o</sup> † 고정국 † 김길용  
 † 부산대학교 컴퓨터공학과  
 {sehkim, gykim}@hyowon.pusan.ac.kr  
 † 동명정보대학교 컴퓨터공학과  
 jgkoh@tmic.tit.ac.kr

## Development of Internet Connection Supporting Module for Embedded OS

† Se-Hyun Kim<sup>o</sup> † Jeong-Gook Koh † Gil-Yong Kim

† Dept. of Computer Engineering, Pusan National University

† Dept. of Computer Engineering, Tongmyong University of Information Technology

### 요 약

본 논문에서는 Embedding OS를 대상으로 한 인터넷 접속 지원 모듈을 개발하고 기능을 시험하였다. 개발된 인터넷 접속 지원 모듈은 PPP 전화접속네트워크와 네트워크 라이브러리, DHCP 클라이언트, TFTP 클라이언트/서버, 텔넷 클라이언트 프로그램으로 구성되어 있으며, Embedded OS에서 인터넷 접속을 위한 네트워크 환경 설정과 파일 송/수신, 원격 시스템 접속 기능 등을 제공하고 있다.

### 1. 서론

오늘날 인터넷을 이용한 비즈니스 거래 등이 확대되면서 인터넷을 기반으로 하는 디지털 경제의 중요성과 발전 가능성이 강조되고 있다. 인터넷 사용자는 개인과 기업 사용자를 합하여 이미 1억명을 넘어섰으며, 매년 증가율이 20%에 달하여 2000년 이후에는 4억명 이상이 될 것으로 예상된다[1,2].

한편, 정보 가전 산업에서는 다양한 우수하고 서비스의 제공 여부가 제품의 경쟁력 제고에 영향을 미치므로, 이를 위해서는 소프트웨어 기술 외에도 PDA 같은 내장 시스템용 실시간 운영체제(Real-time operating system)의 확보가 중요하다. 따라서, 일부 국가나 지역에서 독자적인 RTOS를 확보하고 있으며, 현재 우리나라도 이러한 추세에 따라 정보 가전용 RTOS를 개발하고 있다. 정보 가전용 RTOS는 PDA나 셋탑 박스, 디지털 TV 등에 사용되며, MPEG 지원 API, I-TV API, VBI (Vertical Blanking Interval) API와 함께 웹 브라우저, 멀티미디어 전자메일, 자바 응용을 위한 퍼스널 자바 등을 지원하는 통합 기술 개발이 진행되고 있다[15].

본 논문에서는 국산 RTOS용 인터넷 접속 기능 개발을 위해 그림 1과 같이 Embedded OS에서 구동되는 DHCP 클라이언트, TFTP 클라이언트/서버, Telnet 클라이언트를 개발하였으며, 응용 프로그램의 네트워크 사용을 지원하는 네트워크 라이브러리도 구현하였다. 본 논문의 구성은 다음과 같다. 2장에서는 국내외 통신 응용 프로토콜 개발 현황을 기술하며, 계속해서 인터넷 접속 지원 모듈의 개발 내용을 세부적으로 기술한다. 3장에서는 네트워크 라이브러리에

대해 기술한다. 4장에서는 DHCP 클라이언트를, 5장에서는 TFTP 클라이언트/서버를, 그리고 6장에서는 Telnet 클라이언트에 대해 기술한다. 마지막으로 7장에서는 결론 및 향후 연구 방향에 대해 기술한다.

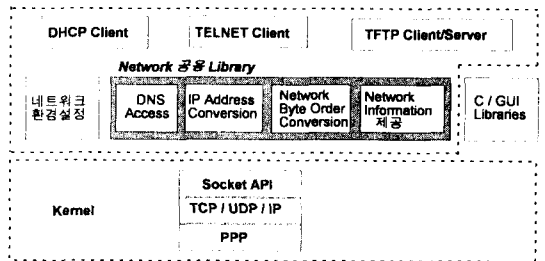


그림 1. 인터넷 접속 지원 모듈의 구조

### 2. 인터넷 접속 지원 모듈의 개발 현황

RTOS는 시스템 제어에 주로 활용되었기 때문에 범용 OS와는 다른 모습으로 발전해왔다. 그러나, 마이크로 프로세서의 성능이 향상되고 네트워크에 연결되면서 RTOS 분야도 변모하기 시작했다. 즉, GUI를 추가하고 TCP/IP 프로토콜 스택과 자바 가상 기계를 탑재하는 등 범용 OS 모습을 갖추기 시작하였다. 따라서, 개발중인 국산 RTOS도 인터넷 접속 지원 기능을 개발해야 한다.

인터넷 접속 지원 기능의 핵심은 통신 응용 프로토콜이며, 기술의 적용 범위가 넓기 때문에 이미 국외에서 충분한 상용화 과정을 거친 상태이다. 대표적인 외국 기업은 FTP, HummingBird, Xlink, Weird Solutions, Computer Solutions, EHS Company, Incognito, Vicomsoft 등이며, 국내 기업으로는

퓨터, Aisoft, 이미지랜드, 넷사랑 등을 들 수 있고 업체 수는 계속 증가할 전망이다.

3. 네트워크 라이브러리의 개발

일반적으로 호스트는 주소 지정을 위해 32 비트의 IP 주소 방식을 사용하지만[3], 사용자들이 직접 IP 주소를 사용하거나 관리하기 어려워서 기호 주소 지정 방식이 대안으로 제시되었다.

네트워크 라이브러리는 기호 주소와 IP 주소간의 변환 기능 외에도 응용 프로그램상에서 네트워크 사용을 지원하는 루틴을 제공한다. 호스트의 기호 주소는 그림 2와 같이 resolv 라이브러리와 host 라이브러리를 이용하여 내부적으로 IP 주소로 변환된다.

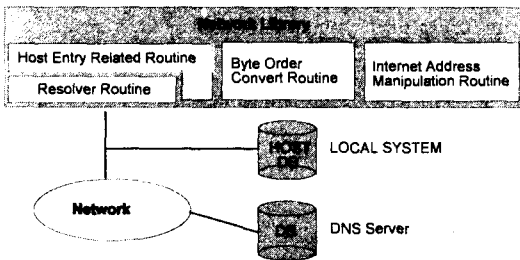


그림 2. 네트워크 라이브러리의 구조

구현된 네트워크 라이브러리는 다음 기능을 제공한다.

- DNS Access  
DNS 서버에 질의를 보내며, TCP/UDP 연결을 지원한다.
- IP 주소 변환  
IP 주소를 TCP/IP의 내부 자료 구조에 맞게 변환시킨다.
- 네트워크 바이트 순서 변환  
호스트 바이트와 네트워크 바이트의 순서를 바꾼다.
- 네트워크 호스트에 대한 정보 획득  
네트워크상의 호스트에 대한 정보를 얻어 낸다.
- 프로토콜에 대한 정보 획득  
네트워크 프로토콜에 대한 정보를 얻어 낸다.
- 네트워크 서비스와 네트워크에 대한 정보 획득

4. DHCP 클라이언트의 개발

DHCP는 그림 3과 같이 각 호스트가 DHCP 서버로부터 네트워크 설정 정보를 동적으로 획득하여 네트워크 설정에 이용하며, IP 주소의 중복 할당을 방지하여 네트워크 관리를 용이하게 한다[4,5,6]. 또한, 각 호스트에게 할당된 주소가 더 이상 사용되지 않으면 해당 주소를 다른 호스트에게 재할당한다. 따라서, IP 주소가 간헐적으로 사용될 때 DHCP 프로토콜은 매우 유용하며 설치가 간편하여 네트워크 관리 비용도 절감된다. 한편, 선택사항(option) 설정 기능과 동적 IP 주소 할당 기능이 지원되는 Bootstrap 프로토콜(BOOTP)의 논리적 확장 형태로서 UDP 프로토콜이 이용된다[7,8,9].

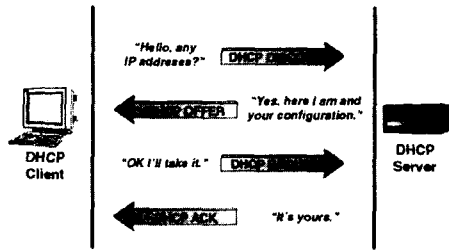


그림 3. DHCP 프로토콜의 역할

클라이언트에 대한 IP 주소 할당 과정은 다음과 같다. 우선, 클라이언트는 서버의 IP 할당 제안(DHCP OFFER)을 수집하기 위해 DHCPDISCOVER 메시지를 broadcast 한다. DHCP OFFER 메시지 수신 후 클라이언트는 특정 서버를 선택하여 주소 할당을 요청하며, DHCPINFORM 메시지를 서버에 전송하고 DHCPACK 메시지를 기다린다. 끝으로 메시지를 수신한 클라이언트가 네트워크 파라메타를 선정하여 환경 설정을 마치면 네트워크 설정 과정은 완료된다. 할당된 주소는 임대 기간동안 사용가능하며 이 기간이 끝나기 전에 서버로부터 주소를 재할당받아야 한다. IP 주소의 재할당 형태는 다음과 같다.

- 임대 기간의 1/2 경과 후 DHCPREQUEST 메시지를 서버에 전송하여 설정 내용을 갱신하며, DHCPACK 메시지를 통해 서버는 임대 기간을 재설정한다.
- 임대 기간의 7/8이 경과되면 DHCPREQUEST 메시지를 broadcast 한다. 주소 할당 서버가 메시지를 수신하면 주소 임대 정보를 갱신하며, 다른 서버가 수신하면 주소의 가용 여부를 판단하여 요청을 수락하거나 주소 할당 과정을 다시 시작한다.

DHCP 클라이언트는 그림 4와 같이 패킷 필터링 기능을 이용하여 라이브러리나 유한 상태 기계(FSM) 형태로 구현된다. 기능이 제한적인 라이브러리 형태에 비해 FSM 방식은 다양한 기능을 제공할 수 있기 때문에 본 논문에서는 FSM 방식으로 DHCP 클라이언트를 구현하였다.

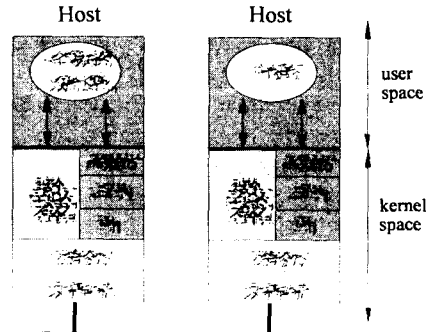


그림 4. DHCP 시스템

5. TFTP 클라이언트 / 서버

TFTP(Trivial File Transfer Protocol)는 FTP만큼 다양한 기능들을 제공하지는 않지만 호스트간에 파일 송,수신 기능을 제공하며, UDP 프로토콜을 사용한다[14]. 또한, DHCP나 BOOTP 클라이언트처럼 저장 장치가 없는 워크스테이션에 사용될 경우 TFTP 서버로부터 초기화 코드

이미지를 다운로드한다.

데이터 전송은 파일에 대한 읽기/쓰기 요청으로 시작된다. 서버가 요청을 승인하면 두 호스트간에 연결이 설정되며, 512 바이트 크기의 블록을 이용하여 파일이 전송된다. 데이터 패킷은 1개의 데이터 블록으로 구성되며 ACK 패킷에 의해 승인이 이루어진다. 결국, 패킷 크기가 512 바이트보다 작다면 마지막 패킷이 된다.

만약 전송 도중 패킷이 소실되면 송신측에서 타임아웃이 발생하여 패킷을 재전송해야 하므로 패킷 재전송에 대비해 마지막 전송 패킷을 버퍼에 유지한다. 또한, 전송중 에러가 발생하면 송,수신자간의 연결이 끊어지며 에러 패킷을 통해 에러 발생을 통지한다.

TFTP에서 사용되는 패킷들은 RRQ(Read Request), WRQ(Write Request), DATA, ACK, ERROR로 구분되며, 세션 설정 방식은 2가지로 구분된다. 즉, WRQ 패킷을 이용하여 서버에 데이터를 업로드하며, RRQ 패킷을 이용하여 서버로부터 데이터를 다운로드한다.

구현된 TFTP 패키지는 그림 5의 tftpsubs 모듈을 공통으로 사용하며, 이 모듈은 패킷 헤더에 대한 인코딩과 디코딩을 담당한다. 또한, 전달 모드(ascii/binary)에 따라 CR/LF의 처리 방법을 결정한다. 구현된 TFTP 패키지는 다음 기능을 제공한다.

- TFTP 클라이언트-서버간의 연결 설정 기능
- TFTP 클라이언트-서버간의 파일 송,수신 기능
- 패킷 트레이스 설정 기능

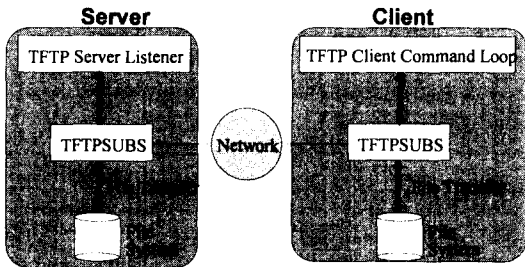


그림 5. TFTP 클라이언트-서버간의 상호 작용

### 6. Telnet 클라이언트

텔넷 서버는 대화를 주도하며 네트워크 가상 터미널(NVT)에 서비스들을 제공한다[13]. 텔넷 클라이언트는 그림 6의 텔넷 세션을 통해 다른 호스트로의 원격 로그인인 가능하도록 해준다. 즉, 사용자가 실제 사용하는 장치로부터 명령을 받아들인 후 NVT로 변환하며, 서버가 NVT에 제공하는 응답을 받아들여 실제 사용자의 장치가 받아들일 수 있는 응답으로 변환하여 보낸다. 텔넷 소프트웨어는 클라이언트 태스크와 서버 태스크간의 TCP 연결을 통해 로그인 세션을 구현한다.[17,19,20,21,22]. 구현된 텔넷 클라이언트는 다음 기능을 제공한다.

- 양방향 통신 세션 설정 기능
- NVT 설정 기능
- 4.2 BSD 용 Telnet 표준 명령어들 지원
- 표준제어 함수 지원

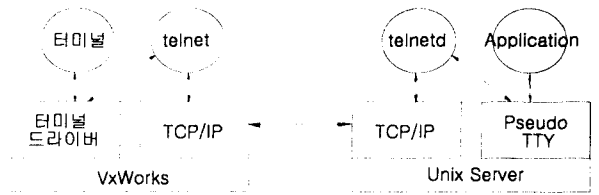


그림 6. 텔넷 세션

### 7. 결론

본 논문은 국산 RTOS 용 인터넷 접속 지원 모듈 개발을 위해 Embedded OS 상에서 수행된 인터넷 접속 지원 모듈 개발 내역을 기술하였다. 이 모듈은 네트워크 라이브러리와 DHCP 클라이언트, TFTP 클라이언트/서버, Telnet 클라이언트 프로그램으로 구성되어 있으며, 네트워크 환경 설정과 파일 송,수신, 원격 로그인 기능 등을 제공한다.

향후 연구 방향은 EmbeddedOS 내에서 개발된 인터넷 접속 지원 모듈들을 사용자가 편리하게 사용하도록 그래픽 사용자 인터페이스를 제공하고 기능을 보강할 예정이다.

### 참고 문헌

- [1] OVUM, International Market Forecasts, OVUM, 1998.
- [2] 서동일, 이종협, 강훈, "차세대 인터넷의 연구동향," 주간기술동향, 한국전자통신연구원, 1999년 1월.
- [3] R. Droms and K. Fong, "Netware/IP Domain Name and Information," RFC 2242, November 1997.
- [4] D. Provan, "DHCP Options for Novell Directory Services," RFC 2241, Novell Inc., November 1997.
- [5] S. Alexander and R. Droms, "DHCP Options and BOOTP Vendor Extensions," RFC 2132, March 1997.
- [6] R. Droms, "Dynamic Host Configuration Protocol," RFC 2131, Bucknell University, March 1997.
- [7] W. Wimer, "Clarifications and Extensions for Bootstrap Protocol," RFC 1542, CMU, October 1993.
- [8] R. Droms, "Interoperation Between DHCP and BOOTP," RFC 1534, Bucknell University, October 1993.
- [9] S. Alexander, R. Droms, "DHCP Options and BOOTP Vendor Extensions," RFC 1533, Bucknell University, October 1993.
- [10] R. Droms, "Dynamic Host Configuration Protocol," RFC 1531, Bucknell University, October 1993.
- [11] James Martin with JOE LEBEN, TCP/IP 네트워킹 구조, 관리, 프로그래밍.
- [12] 신재수, 권병희, 임강진, Network Bible, 영진출판사.
- [13] Jose Nabelesky, Anita P. Skelton, "A Virtual Terminal Management Model", RFC 782.
- [14] K. Sollins, "The TFTP Protocol (Revision 2)," RFC 783, NIC, June 1981.
- [15] 특별기획 "임베디드 시스템을 위한 RTOS 와 개발툴 퍼레이드," 마이크로소프트웨어 9월호, 1999.
- [17] J. Postel, J.K. Reynolds. "Telnet Extended Options: List Option.", RFC 861, May 1983.
- [18] J. Postel, J.K. Reynolds. "Telnet Timing Mark Option.", RFC 860, May 1983.
- [19] J. Postel, J.K. Reynolds. "Telnet Status Option.", RFC 859, May 1983.
- [20] J. Postel, J.K. Reynolds. "Telnet Binary Transmission.", RFC 856, May 1983
- [21] J. Postel, J.K. Reynolds. "Telnet Option Specifications.", RFC 855, May 1983
- [22] J. Postel, J.K. Reynolds. "Telnet Protocol Specification.", RFC 854, May 1983.