

---

# 홈 네트워크를 위한 Embedded Ethernet Module 설계 및 구현

김판규\* · 황태문\*\* · 이종혁\*

Design and Implementation of Embedded Ethernet Module for Home Network

Pan-kyu Kim\* · Tae-Moon Hoang\*\* · Jong-hyeok Lee\*

---

이 논문은 2004년도 Brain Busan 21사업에 의하여 지원되었음.

---

## 요 약

초고속 인터넷의 보급, 디지털 정보장비의 확산 및 생활 패턴의 변화에 따라 홈 네트워크의 필요성이 점차 확대되고 있다. 이에 따라 본 연구에서는 홈 네트워크화를 위해서 8051 MCU, Hardware TCP/IP를 이용하여 임베디드 이더넷 모듈을 설계, 개발하였다. 타겟 머신에 간단한 웹 서버를 구축하고 이를 이용한 원격 제어 프로그램을 포팅하여 원격지에서도 인터넷으로 가전기기를 제어할 수 있도록 하였다. 본 연구에서 개발한 모듈을 이용하므로 저가적으로 홈 오토메이션을 구현할 수 있으리라 생각한다.

## ABSTRACT

The necessity of the home network have been enlarged gradually according to the distribution of ultra-high speed internet, the expansion of a digital information equipment and the change of a life pattern.

In this paper, we proposed the embedded ethernet module that could be operate on eight bit system, but it could control home appliance with full ethernet speed.

The embedded ethernet module consists of 8051 MCU and Hardware TCP/IP. In this module, we construct simple web server and port remote control program for I/O device control. We verified through internet that the developed embedded ethernet module could control and check home appliances anywhere and anytime.

We expect the developed ethernet module can build up niche market at home network. And it will be helpful to activate home network market.

## 키워드

홈 네트워크, 임베디드 이더넷, 하드웨어 TCP/IP, 가전제어

## I. 서론

우리나라는 초고속인터넷 보급이 1000만명을 돌파했고, 전 국민 63% 이상이 인터넷을 이용하고 있는 세

계적인 인터넷 강국으로 부상하였으며, 이를 바탕으로 황금알로 떠오르고 있는 홈 네트워킹 시장에서도 경쟁 우위를 가질 수 있으리라 전망한다. 홈 네트워크 표준화 작업, 유무선 통합 게이트웨이 등장, 유무선 통합

---

\* 경성대학교 컴퓨터 공학과

\*\* 대전정보통신고등학교

홈 서버, 광대역 통합 디지털 콘텐츠 보편화로 인해 홈 네트워크는 언제 어디서나 어느 기기로도 이용 가능토록 발전이 예상된다. 그러나 네트워크의 표준화가 아직 결정되지 않았고 표준 관련 기기를 개발해야 할 단계이다.[1][2]

임베디드 시스템에서 인터넷 기능을 구현하는 방법에는 OS에 포함되어 있는 Software TCP/IP Protocol Stack을 이용한 방법과 OS-less의 하드웨어 TCP/IP 솔루션을 이용한 두가지 방법이 있다.[3]

기존 임베디드 시장에 가장 많이 사용되고 있는 8비트 시스템에서 OS기반의 솔루션을 이용할 때는 네트워크의 안정성과 속도 등이 떨어진다. 이를 해결하기 위하여 OS-less의 Hardware TCP/IP 솔루션이 제안되었다.

따라서 본 연구에서는 홈 네트워크화를 위해서 8051 MCU, Hardware TCP/IP를 이용하여 가전기기를 인터넷으로 제어할 수 있는 임베디드 이더넷 모듈을 설계, 구현하고자 한다.

## II. 임베디드 인터넷

임베디드 인터넷이란 임베디드 시스템에 인터넷 기능을 구현한 것으로, 이를 구현하는 방법에는 두가지가 있다.

OS기반 솔루션의 경우 32-Bit 이상의 MCU를 사용하는 시스템에 적용 가능한 방법으로 기존 8-Bit, 16-Bit MCU 시스템에는 적용하기 어렵다. 굳이 적용하고자 한다면 MCU교체, 메모리확장, OS탑재 등 시스템 재설계가 불가피하다.[4]

반면, 하드웨어 TCP/IP 솔루션의 경우 별도의 OS없이 하드웨어로 TCP/IP 프로토콜 프로세싱을 전담함으로써 기존 Low-end 시스템에 간단히 추가함으로써 쉽게 구현이 가능하다

한편, 하드웨어 TCP/IP 솔루션이 최근 들어 이슈가 되고 있는 이유는 “MCU Offload” 라는 이슈 때문이다. OS기반 솔루션의 경우 MCU가 Main Job과 실시간 OS 처리 및 TCP/IP Protocol 프로세싱을 동시에 처리하기 때문에 실시간 멀티미디어 응용에서는 기대하는 만큼의 성능을 내지 못하고 있다. 특히 아래와 같은 TCP Protocol 프로세싱에 관련된 일에 MCU의 상당한 Com-

putation Power를 소모하고 있기 때문이다.

- Checksum Calculation
- Connection Management
- Setup and Teardown
- Congestion Control Algorithm
- Sliding-window Flow Control
- Out of Order Support
- Fragmentation
- Timer Management
- Proxying
- Data Buffering
- Address Filtering

따라서, 별도의 하드웨어 Logic이 이러한 일들을 처리해준다면 MCU는 더 많은 Computing Power를 Main Job에 쓸 수 있게 되어 자연스레 전체 시스템의 성능이 올라가게 된다. 또한 하드웨어로 프로토콜 프로세싱이 이루어지기 때문에 OS기반 솔루션에서 발생하는 처리지연이나 지터 (Jitter)와 같은 문제가 해결되어 QoS(Quality of Service)가 보장되고, 하드웨어 특유의 뛰어난 안정성을 제공하며, 요구되는 MCU의 성능을 down grade할 수 있어 저급의 MCU만으로도 기대하는 수준의 성능을 얻을 수 있게 된다.

## III. 임베디드 이더넷 모듈 설계

### 3.1 임베디드 이더넷 모듈의 구성

임베디드 이더넷 모듈의 전체 Block Diagram은 그림 1과 같다.

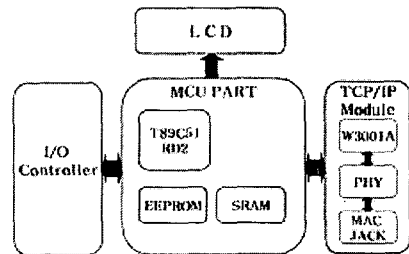


그림 1. 임베디드 이더넷 모듈 Block Diagram

3.1.1 MCU 하드웨어

개발 모듈 MCU의 Main Clock은 16MHz를 사용하고, X2 Mode로 동작한다. MCU는 T89C51R D2이며 이 MCU의 특징으로는 8051시리즈의 구조와 주변장치를 모두 가지고 있고, 인 시스템 프로그램 (ISP:In System Programming) 기능을 제공한다. 이를 그림 2에 나타내었다. 인시스템 프로그램이란 T89C51RD2 칩이 보드에 납땜된 상태에서 T89C51RD2 내부의 플래쉬 메모리를 프로그램 하는 것으로 이를 사용하기 위해서는 PC와 T89C51RD2을 RS-232c 케이블로 연결하고, PC상에서 FLIP이라고 부르는 인 시스템 프로그램을 실행해야 한다.[5]

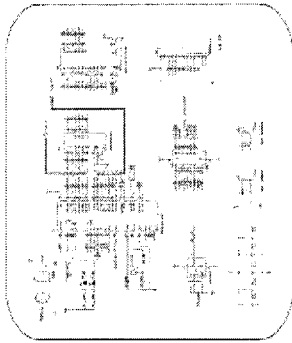


그림 2. MCU 회로도

3.1.2 이더넷 제어 설계

본 모듈에서는 인터넷 구현을 위해서 Ethernet의 MAC layer와 TCP/IP layer의 일부를 구현해 놓은 하드웨어 TCP/IP칩 W3100A와 Ethernet의 PHY layer를 구현해 놓은 RTL8201를 사용하였다. 하드웨어 TCP/IP칩은 그림 3에서 보는 바와 같이 기존 Software TCP/IP 프로토콜과 Ethernet MAC을 하나의 칩 속에 구현하고 송수신 데이터를 위한 버퍼로 DPRAM을 내장하고 있어 별도의 OS 없이 하드웨어로 TCP/IP 프로토콜을 처리할 수 있도록 하여 이더넷에 들어오는 데이터를 Line Speed 가깝게 실시간으로 처리한다.

상위 Layer에 해당하는 MCU 인터페이스는 버스 인터페이스나 I2C 인터페이스를 통해 메모리 핸들링 방식으로 쉽게 TCP/IP칩을 제어할 수 있고, 하위 Layer인 Ethernet PHY와는 표준 MII인터페이스를 통해 Seamless하게 데이터를 주고 받는다.

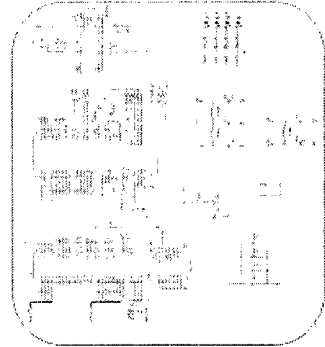


그림 3. TCP/IP 모듈 회로도

3.1.3 I/O 컨트롤러

본 모듈의 Web Server를 활용하여 모듈과 연결된 Device들의 전원을 제어할 수 있다.

I/O Interface Board는 3개의 Device들의 전원 제어와 상태 정보를 확인 할 수 있게 하기위해 Device의 전원 제어는 MCU측에서 출력을 받아 LATCH에 의해 상태를 유지 시킨다.

Device의 동작상태 확인을 위하여 포토 카플러 등을 이용하여 만들어진 Device의 동작 상태 정보를 Buffer로 보낸다.

P1는 DC 5V와 MCU I/O Port로 Data를 입력과 출력을 받는 역할을 하며, P2는 Buffer, TranSceiver, Latch를 제어 한다. 그림 4는 원격제어를 위한 I/O Control Board의 블록도이다.

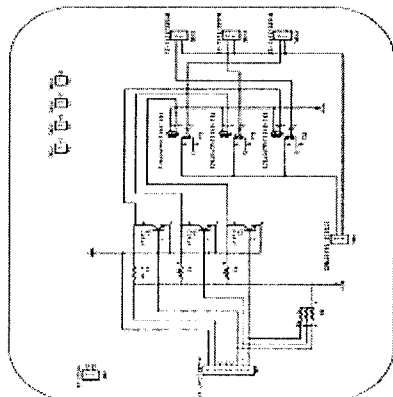


그림 4. I/O Controller 회로도

3.2 보드 제작

회로도 설계가 끝난 후 작성된 회로도를 가지고 보드 제작에 들어갔다. 보드 제작을 하기 위한 중간 단계로 OrCAD 9.2를 이용해 Layout 작업을 수행하였다. Layout 작업은 우선 선택되어진 보드에 칩들을 알맞은 위치에 배치하고 난 후 Routing 방법으로 진행될 수 있는데 자동 Routing 또는 수동 Routing 방법으로 진행될 수 있는데 자동 Routing은 컴퓨터가 판단하여 Routing을 수행하는 방법으로 Routing은 간단하게 이루어지지만 전체적인 Layout 모양이 좋지 않고 Routing이 이루어지지 않는 곳이 많이 생겨서 모든 Routing은 수동 Routing으로 수행하였다. 수동 작업 시 보드 전체에 전원이 안정적으로 공급이 될 수 있도록 하였고 각 칩의 전원부에 커패시터를 연결하여 노이즈 제거와 전원공급의 안정화를 기했다. 이것을 가지고 최종적으로 보드를 제작하였는데 Layout이 완성된 모습을 그림 5에 나타내었다.

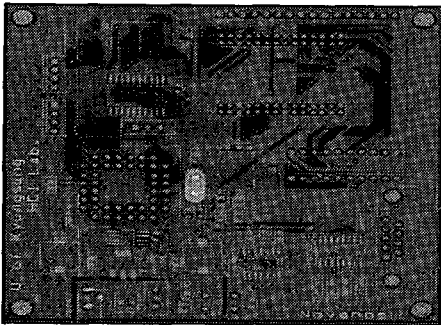


그림 5. 임베디드 이더넷 모듈 레이아웃

3.2 소프트웨어 설계

W3100A Socket API는 W3100A의 초기화, Socket 생성, TCP 연결 설정 및 대기, Data 통신, Socket Monitoring, Socket 해제를 지원한다.[6]

W3100A의 초기화 과정은 그림 6과 같다.



그림 6. W3100A의 초기화 과정

W3100A Socket API로 Sever Program을 구현 할 경우는, 동일 Listen Port를 사용하는 Server Socket을 최대 4개까지 사용할 수 있으며, 만약 동일 Listen Port의 Server Socket이 하나 이상 존재할 경우, Client의 접속 요구 시 Server Socket들 중 Channel Number가 가장 낮은 Socket부터 접속을 받아들이고, Client 통신 Socket으로 전환된다. Server Socket은 Client Socket과 일대일로 대응되며, 이의 흐름도는 그림 7과 같다.

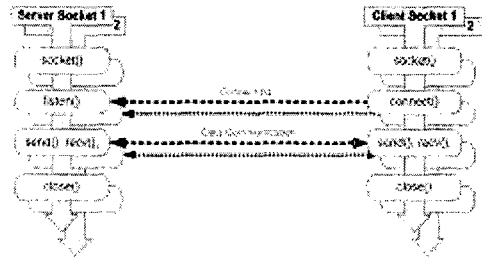


그림 7. Server/Client 흐름도

Web Server는 Web Browser로부터 들어온 HTTP Request의 Method와 URI(Uniform Resource Identifier)를 분석하고 URI가 WEB Page 요구시 해당 Page를 찾아 전송하면 CGI(Common Gateway Interface)와 같은 Action을 요구하는 경우에는 이에 해당하는 Action을 취한뒤 그 결과를 Web Page 형태로 알려준다.[7]

그림 8은 웹서버 MAIN()함수의 flowchart이다.

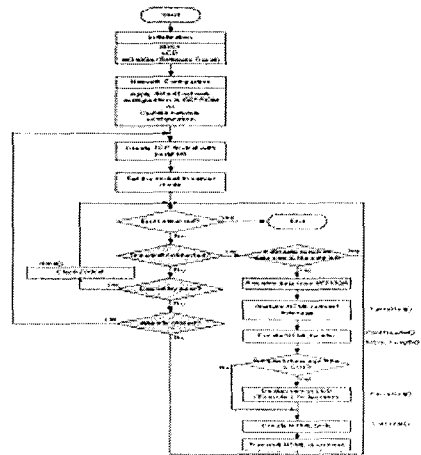


그림 8. 웹서버 MAIN() 함수의 flowchart

#### IV. 임베디드 이더넷 모듈 구현

##### 4.1 개발환경

본 논문에서 제안한 모듈의 개발환경으로는 윈도우즈 2000, 팬티엄 IV 1.4GHz이며 사용된 소프트웨어는 C와 HTML이며, keil 8051 c compiler v5.20로 컴파일한 후 HEXA 파일을 시리얼 통신으로 개발한 모듈에 다운로드 하였다. 그리고 모듈의 IP는 210.110.180.108로 설정하였다.

##### 4.2 웹서버 프로그램

본 모듈의 동작하기 위한 웹서버 Application 프로그램은 W3100A Socket API를 이용하였다.

그림 9는 웹서버 main()함수코드를 나타낸다.

```

void main()
{
    SOCKET s;
    u_char statetype;
    Init8051();
    PutStringLn("HTTP Server Program");
    InitLed();
    InitW3100();
    InitNetConfig();
    for (i = 0; i < MAX_SOCKET_NUM; i++) init_socket(i);
    while(1)
    {
        state = select(0, SEL_CONTROL);
        switch(state)
        {
            case SOCK_ESTABLISHED:
                if ((len = select(f, SEL_RECV)) > 0)
                {
                    if (len > MAX_BUF_SIZE) len = MAX_BUF_SIZE;
                    len = recv(i, rx_buf, len);
                    type = ParseReq(rx_buf);
                    len = PrintHeader(tx_buf, type);
                    switch (type)
                    {
                        case c.: len += DoHTML(tx_buf + len);
                    }
                    len = send(i, tx_buf, len); break;
                }
            }
    }
}
    
```

그림 9. 웹 서버 main()함수의 소스코드

그림 9에서 (1)은 8051을 초기화하고 (2)는 W3100A의 Network 부분을 설정한다.

초기화 과정이 끝나면 (3)을 통해 웹서버가 사용하게 될 Socket을 HTTP Sever Port(80)로 대기시킨다. 그다음에 (4)를 통해 각 소켓의 상태를 관찰하고 Web Browser로부터 HTTP Request가 수신되었는지 관찰한다. 만일 HTTP Request가 수신되었을 경우 (5)를 통해 해당 HTTP Request를 수신하고, 수신된 HTTP Request Message는 (6)을 통해 분석되며, (7), (8), (9)를 통해 HTTP Reponse Message를 전송하게 된다.

##### 4.3 웹서버를 이용한 원격 제어프로그램

오픈소스를 참고하여 웹서버에 I/O 제어를 위해

ParseReq()함수를 작성하였다.

그림 10은 원격제어를 위한 Parsereq()함수의 소스코드이다. (1)과 (2)에서는 'led' Parameter의 위치를 찾아내어 그에 해당하는 Parameter 값을 검사한 후에 해당 MCU I/O Port값을 Set하게 되어 I/O 제어가 가능하다.

```

while((Pointer = FindFirstStr(Pointer, led,)) != 0)
    // Parse switch selective value (1)
{
    Pointer++;
    switch(Pointer)
    {
        case '1': // Switch #1 is checked?
            LEDStatus |= 0x01; // LEDStatus Update
            T0 = 0; break; // LED ON
        case '2': // Switch #2 is checked?
            LEDStatus |= 0x02;
            T1 = 0; break;
        case '3': // Switch #3 is checked?
            LEDStatus |= 0x04;
            T0 = 0;
            T1 = 0; break;
    }
}
PI = LEDStatus;
// Apply LED status controlled by PORT1 to RELAY BOARD (2)
if ((LEDStatus & 0x04))
{
    if ((LEDStatus & 0x01)) T0 = 1; // SW1 OFF
    if ((LEDStatus & 0x02)) T1 = 1; // SW2 OFF
}
else if ((LEDStatus == 0)) // SW ALL OFF
    
```

그림 10. 원격 제어를 위한 소스코드

##### 4.4 구현결과

웹 서버에 접속한 화면은 그림 11과 같다.

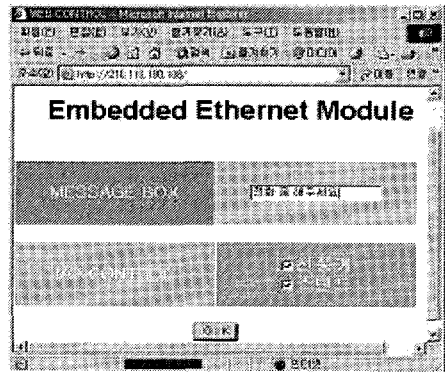


그림 11. 웹서버 접속 화면

웹 Browser에서 TEXTBOX에 "ON/OFF control" 문자열을 입력하고 선풍기와 스탠드를 선택하고 OK 버튼을 선택했을 경우 Request 메시지가 모듈로 전송되면 MCU I/O Port의 값을 set 한 후 가전기기를 제어하게 된다. 그림 12는 실제 구현된 결과이다.

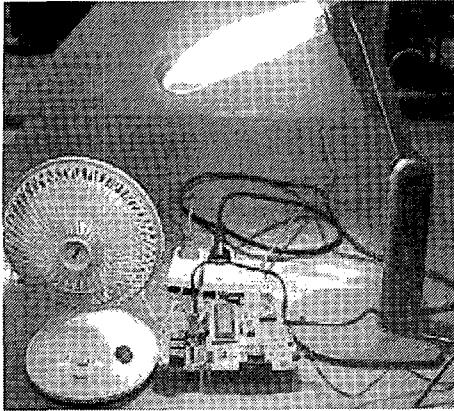


그림 12. 임베디드 이더넷 모듈의 구현 결과

## V. 결론

본 연구에서는 8비트 시스템에서 네트워크의 안정성과 속도를 어느 정도 보장하면서 가전기기를 제어할 수 있는 임베디드 이더넷 모듈을 개발하는데 중점을 맞추었다. 개발된 임베디드 모듈은 인터넷이 가능한 곳에서는 언제 어디서나 가전기기를 제어할 수 있고, 이의 동작 상태를 확인 할 수 있었다.

개발된 모듈의 기대 효과로는 언제 어디서나 컴퓨팅이 가능한 u-KOREA(Ubiquitous Korea)의 조기 실현이 가능하고, 홈 네트워크 시장의 틈새를 공략할 수 있을 것이다 그리고 홈 네트워크가 지원이 되지 않는 기존 가전에 모듈을 삽입하여 네트워크 가능하도록 하므로 많은 추가비용 없이 홈 네트워크의 구축이 가능하므로, 홈 네트워크 시장 활성화에 기여하리라 생각된다.

## 참고문헌

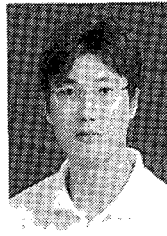
- [1] 김진경, “홈네트워크 업계동향” Digital Contents 2월호, pp71-81, 2004.2
- [2] 김동택, “홈 종합 정보화를 위한 홈 서버 개발”, 한국해양정보통신학회, 제 8권 제 1호, pp. 755-759, 2004
- [3] 오민정, “기기 제어를 위한 저가의 초소형 임베디드 웹서버”, 정보처리학회, 제 9권-A권 제 1호

pp.1-8, 2002. 3

- [4] 추광재, “임베디드 인터넷” 임베디드 월드 6월호, pp82-87, 2004, 6
- [5] <http://sample.co.kr/se8051icd/>
- [6] 김우열, “임베디드 인터넷 프로그래밍” 임베디드월드 7월호 pp.58-64 2004. 7.
- [7] 김우열, “임베디드 인터넷 응용 1”, 임베디드월드, 8월호, pp. 78-87, 2004. 8.

## 저자소개

### 김판규(Pan-Kyu Kim)



2004년 2월 경성대학교 컴퓨터공학과(공학사)  
2005년 현재 경성대학교 교육대학원 석사과정

※ 주관심분야 : e-러닝, 멀티미디어 콘텐츠, 영상처리

### 황태문(Tae-Moon Hoang)

1987년 2월 울산대학교 전자계산학과(공학사)  
2001년 8월 경성대학교 정보공학과(공학석사)  
2005년 현재 신라대학교 신기술응용공학부 박사과정  
1991년 3월~1998년 2월 동의공업대학 겸임강사  
1998년~현재 대전정보통신고등학교 컴퓨터정보과 교사  
※ 주관심분야 : 멀티미디어 콘텐츠, 보안, 영상처리

### 이종혁(Jong-Hyeok Lee)



1975년 부산대학교 전자공학과 학사  
1980년 부산대학교 전자공학과 석사  
1991년 부산대학교 전자공학과 박사  
1990년~현재 경성대학교 컴퓨터공학과 교수

1998년 7월~1999년 6월 미국 Beckman Institute, Univ. of Illinois, 객원연구원

※ 주관심분야 : 인공지능, 음성인식, 신호처리