
임베디드 이더넷 모듈의 설계 및 구현

김 영 진*

Design and Implementation of Embedded Ethernet Module

Yung-Jin Kim*

요 약

초고속 인터넷의 보급, 디지털 정보장비의 확산 및 생활 패턴의 변화에 따라 홈 네트워크의 필요성이 점차 확대되고 있다. 이에 따라 본 논문에서는 홈 네트워크화를 위해서 8051 MCU, 하드와이어드 TCP/IP(W3150A)를 이용하여 임베디드 이더넷 모듈을 설계, 구현하였다. 타겟 머신에 간단한 웹 서버를 구축하고 이를 이용한 원격제어 프로그램을 포팅하여 원격지에서도 인터넷으로 가전기기를 제어할 수 있도록 하였다. 개발한 모듈을 이용하여 유비쿼터스 시대에 저가격으로 홈 오토메이션을 구현할 수 있으리라 생각한다.

ABSTRACT

Upon the changes of human life style due to the distribution of ultra-high speed internet services and the development of high-tech information equipment, the idea of necessity for home networking has been widely accepted. Thus, "embedded ethernet module" was designed in this paper for the stabilization of home networking by using 8051 MCU, Hardwired TCP/IP (W3150A). By executing remote software on established simple web-server on the target machine, controlling electric equipment became available elsewhere. With this module, establishing a low-cost home automations system would become possible in a ubiquitous society.

키워드

유비쿼터스, 임베디드 이더넷 모듈, 하드와이어드 TCP/IP, 가전제어

I. 서 론

우리나라는 초고속 인터넷 보급이 1000만명을 돌파했고, 전 국민 63% 이상이 인터넷을 이용하고 있는 세계적인 인터넷 강국으로 부상하였으며, 이를 바탕으로 황금알로 떠오르고 있는 홈 네트워킹(Home Networking) 시장에서도 경쟁우위를 가질 수 있으리라 전망한다.

홈 네트워킹 관련 기술의 발전에 따라 그 범위가 점진적으로 확대되는 추세이다. 초기의 홈 네트워킹은 가정내 보안, 조명, 온도 등을 자동 통제하는 수준인 홈 오

토메이션, 홈 컨트롤 시스템 정도의 범위를 의미하는 것이었다.

그 후 한 가정 내에 여러 대의 PC 및 주변기기를 보유한 가구 수가 증가하면서 이들을 효율적으로 연결시키기 위한 네트워킹 기술들이 등장하기 시작했고, 이를 지원하기 위한 홈 네트워킹 키트들이 시장에 등장하기 시작하면서 홈 네트워킹이란 개념이 본격적으로 사용되기 시작했다. DSL(Digital Subscriber Line), 케이블 모뎀 등 가입자망의 광대역화로 초고속인터넷이 급속히 확대되면서 PC뿐만 아니라 각종 디지털 정보단말기들이

가정 내에 등장하면서 홈 네트워킹의 범위는 급격히 확대되었다[1].

따라서 오늘날 홈 네트워킹은 일반적으로 가정 내 디지털 정보가전들 간에 데이터를 주고받을 수 있는 통로를 제공함과 동시에 외부 인터넷 망과의 접속을 제공함으로써 지능화된 커뮤니케이션이 가능하도록 하는 네트워킹 방식을 의미하며, 궁극적으로 이러한 네트워킹화를 통해 가정 내 정보자원의 공유와 개별 제품들의 효율 극대화 추구를 그 목적으로 하고 있다[2].

홈 네트워크 표준화 작업, 유무선 통합 홈 게이트웨이 등장, 유무선 통합 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. 또한 홈 네트워크 장비가 너무 고가인 점도 시장 활성화에 부담이 되고, 인터페이스의 호환성, 개인의 정보보안 등도 시장 활성화를 위해서는 시급히 해결해야 할 과제이다 [3][4].

가정용 기기를 인터넷에 연결하여 다양한 서비스를 제공하고자 하는 요구가 급증하고 있으나 이미 실생활에서 사용되어 오고 있는 대부분의 기기는 RS-232나 적외선 리모컨 같이 매우 단순한 통신 프로토콜을 내장하거나 통신 기능이 전혀 없다. 따라서 이러한 기기를 인터넷에 연결하기 위해서는 일종의 게이트웨이 기능을 할 수 있는 특별한 연결 장치가 요구된다.

일반적으로 웹 서버를 이용하여 가전기기를 감시 및 제어 할 경우 두 가지 방법으로 접근할 수 있다[5].

첫 번째 방법은 웹 서비스를 제공하기 위하여 별도의 PC를 이용하는 것이다. 이 방법은 별도의 PC는 물론 기기를 제어하기 위한 제어보드를 내장해야 하기 때문에 가격이 높아지고 전력소모가 상당하기 때문에 유지비가 많이 소요된다. 또한 범용 PC의 운영체제는 신뢰도가 낮기 때문에 시스템 다운과 같은 여러 가지 문제가 발생할 수 있다. 두 번째 방법은 웹 서비스를 제공하기 위한 기능을 제어하려는 기기에 내장시키는 것이다. 이 방법은 웹 서버 기능의 구현과 시스템의 운영을 위해 특별한 운영체제를 사용해야 하고 원활한 기능의 구현을 위해서 강력한 마이크로프로세서와 고용량의 메모리

를 필요로 하여 시스템의 개발 비용이 증가하게 된다. 비용을 낮추기 위하여 임베디드 시장에 가장 많이 사용되고 있는 8비트 시스템을 채택하고, OS기반의 솔루션을 이용할 때는 네트워크의 안정성과 속도 등이 떨어지는 단점이 있다. 이러한 문제들을 해결하기 위하여 OS-less의 하드웨어인 TCP/IP 솔루션이 제안되고 있다.

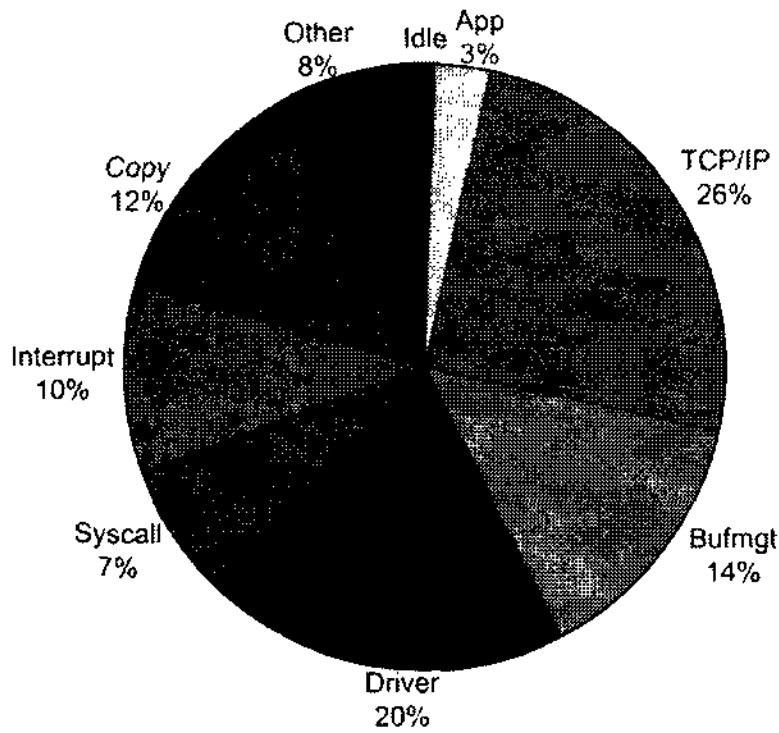
본 논문에서는 임베디드 시장에서 많이 사용하고 있는 8비트 시스템, 8051 MCU(Micro Controller Unit)와 하드웨어인 TCP/IP를 이용하여 가전기기를 인터넷으로 제어할 수 있는 임베디드 이더넷 모듈을 설계, 구현하여 저가격으로 홈 네트워킹화 할 수 있음을 보이고자 한다.

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[6]. 반면, 하드웨어인 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 [7],[8]. OS기반 솔루션의 경우 MCU가 주 작업과 실시간 OS처리 및 TCP/IP 프로토콜 프로세싱을 동시에 처리하기 때문에 실시간 멀티미디어 응용에서는 기대하는 만큼의 성능을 내지 못하고 있다. 이는 그림 1에서 나타난 것과 같이 CPU에서 수행되는 작업 중 TCP 프로토콜 프로세싱에 관련된 일에 CPU의 상당한 Computation Power(26%)를 소모하고 있기 때문이다.



Source by IEEE Computer, 2004

그림 1. 작업별 CPU 성능 점유율
Fig. 1 CPU Performance Ratio by Job

따라서, 별도의 하드웨어 Logic이 이러한 일들을 처리해준다면 MCU는 더 많은 Computing Power를 주 작업에 쓸 수 있게 되어 자연스레 전체 시스템의 성능이 올라가게 된다. 또한 하드웨어로 프로토콜 프로세싱이 이루어지기 때문에 OS기반 솔루션에서 발생하는 처리 지연이나 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 모듈은 크게 MCU 모듈, 하드웨어 TCP/IP 모듈 및 I/O 모듈로 구성되어 있다.

3.1.1 MCU 하드웨어

개발 모듈 MCU의 Main Clock은 16MHz를 사용하고, X2 Mode로 동작한다. MCU는 T89C51RD2이며 이 MCU의 특징으로는 8051시리즈의 구조와 주변장치를 모두 가지고 있고, 인 시스템 프로그램 (ISP: In System Programming) 기능을 제공한다.

인 시스템 프로그램이란, T89C51RD2 칩이 보드에 납땜된 상태에서 T89C51RD2 내부의 플래쉬 메모리를 프로그램 하는 것으로 이를 사용하기 위해서는 PC와

T89C51RD2를 RS-232c 케이블로 연결하고, PC상에서 FLIP이라고 부르는 인 시스템 프로그램을 실행해야 한다[9].

3.1.2 이더넷 제어 설계

본 모듈에서는 이더넷 구현을 위해서 이더넷의 MAC layer와 TCP/IP layer의 일부를 구현해 놓은 하드웨어 TCP/IP칩 W3150A를 이용하였다. 하드웨어 TCP/IP칩은 그림 2에서 보는 바와 같이 기존 소프트웨어 TCP/IP 프로토콜과 이더넷 MAC을 하나의 칩 속에 구현하고 송수신 데이터를 위한 버퍼를 내장하고 있어 별도의 OS 없이 하드웨어로 TCP/IP 프로토콜을 처리할 수 있도록 하여 이더넷에 들어오는 데이터를 Line Speed가깝게 실시간으로 처리한다.

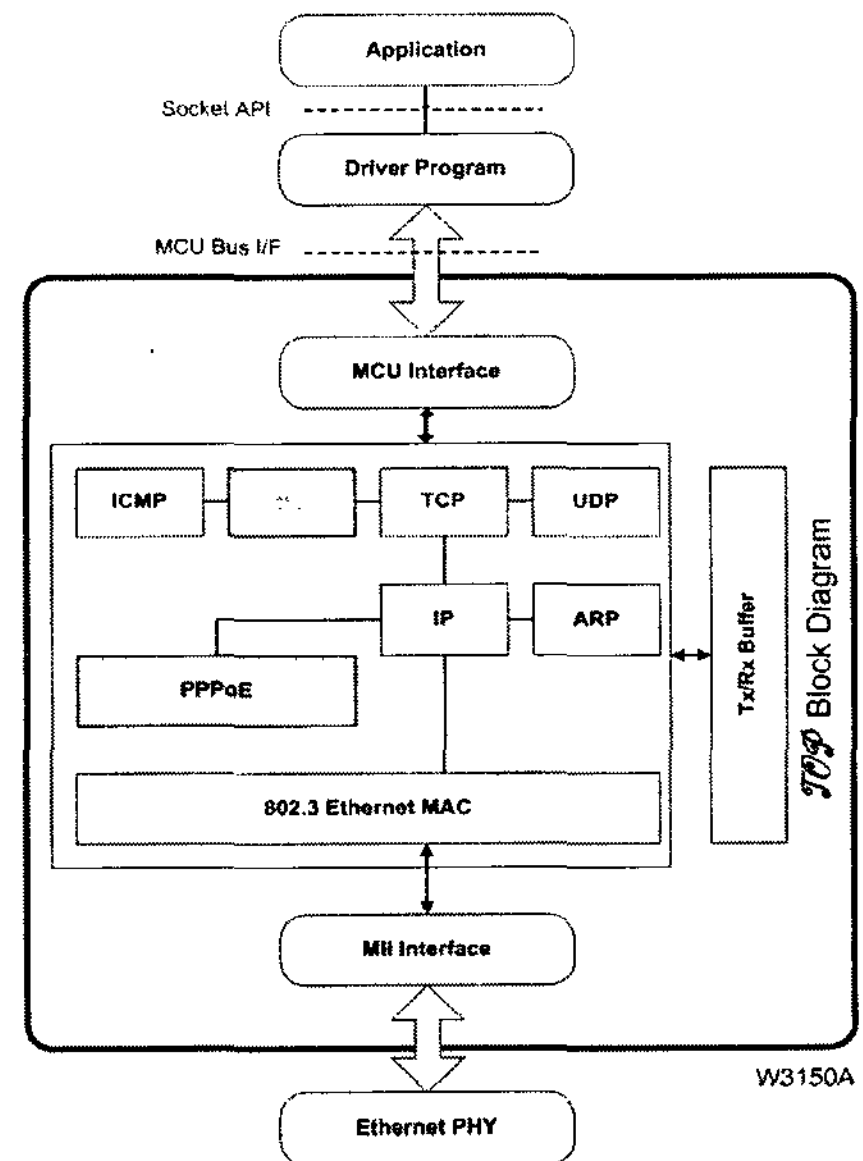


그림 2. W3150A 블록도
Fig. 2 W3150A Block Diagram

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

이더넷 제어관련 모듈은 하드웨어 TCP/IP칩 W3150A와 이더넷의 PHY layer를 구현해놓은 RTL8201를 사용하여 설계하였다.

3.1.3 I/O 컨트롤러

본 모듈의 웹 서버를 활용하여 모듈과 연결된 장치들의 전원 제어 및 상태를 체크할 수 있다.

I/O 인터페이스 보드는 3개의 장치들의 전원 제어와 2개 장치의 상태 정보를 확인 할 수 있게 하기 위해 장치의 전원 제어는 MCU측에서 출력을 받아 래치에 의해 상태를 유지 시킨다. 장치의 동작상태 확인을 위하여 포토커플러 등을 이용하여 만들어진 장치의 동작 상태 정보를 비퍼로 보낸다.

P1는 DC 5V와 MCU I/O Port로 데이터를 입력과 출력을 받는 역할을 하며, P2는 비퍼, Transceiver, 래치를 제어 한다.

3.2 보드 제작

설계된 회로도 보드 제작을 하기 위한 중간 단계로 OrCAD를 이용해 레이아웃 작업을 수행하였다. 레이아웃 작업은 우선 선택되어진 보드에 칩들을 알맞은 위치에 배치하고 난 후 모든 라우팅은 수동 라우팅으로 수행하였다. 수동 작업 시 보드 전체에 전원이 안정적으로 공급이 될 수 있도록 하였고, 각 칩의 전원부에 커패시터를 연결하여 노이즈 제거와 전원공급의 안정화를 한후 최종적으로 보드를 제작하였다.

3.3 소프트웨어 설계

W3150A 소켓 API는 W3150A의 초기화, 소켓 생성, TCP 연결 설정 및 대기, 데이터 통신, 소켓 감시, 소켓 해제를 지원한다[10].

W3150A의 초기화 과정은 그림 3과 같다.

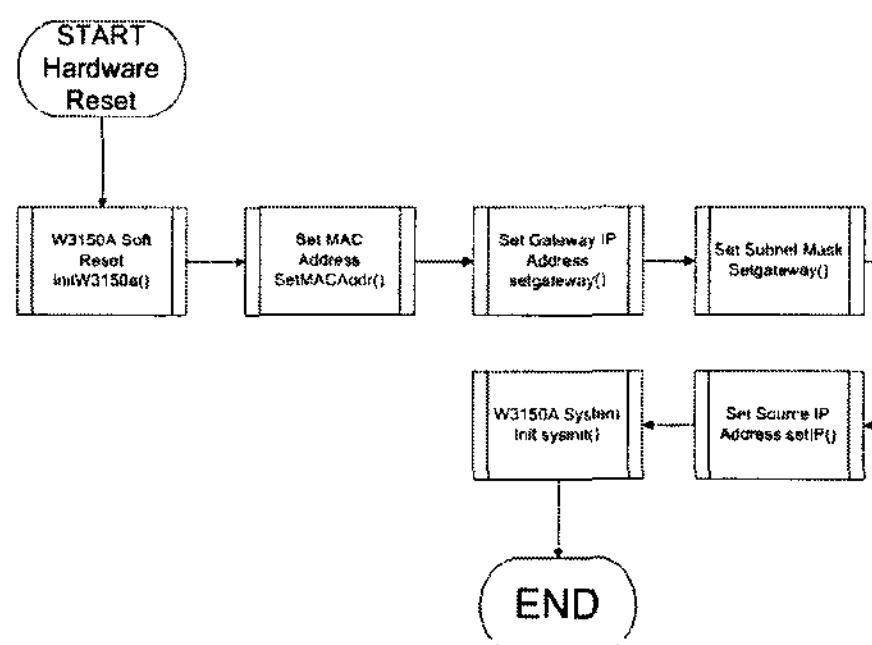


그림 3. W3150A의 초기화 과정
Fig. 3 Step of W3150A Initialization

W3150A 소켓 API로 서버 프로그램을 구현 할 경우는, 동일 Listen Port를 사용하는 서버 소켓을 최대 4개까지 사용할 수 있으며, 만약 동일 Listen Port의 서버 소켓이 하나 이상 존재할 경우, 클라이언트의 접속요구 시 Server 소켓들 중 채널 번호가 가장 낮은 소켓부터 접속을 받아들이고, 클라이언트 통신 소켓으로 전환된다. Server 모드와 클라이언트 모드의 흐름도를 그림 4, 5에 각각 나타내었다.

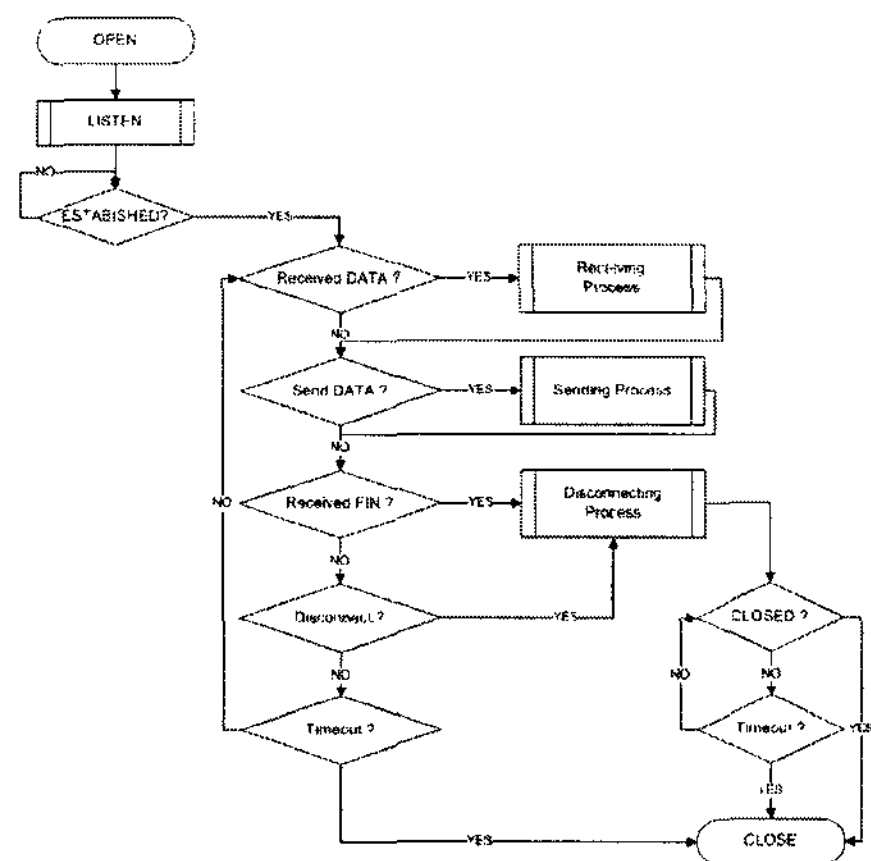


그림 4. TCP 서버 모드
Fig. 4 TCP Server Mode

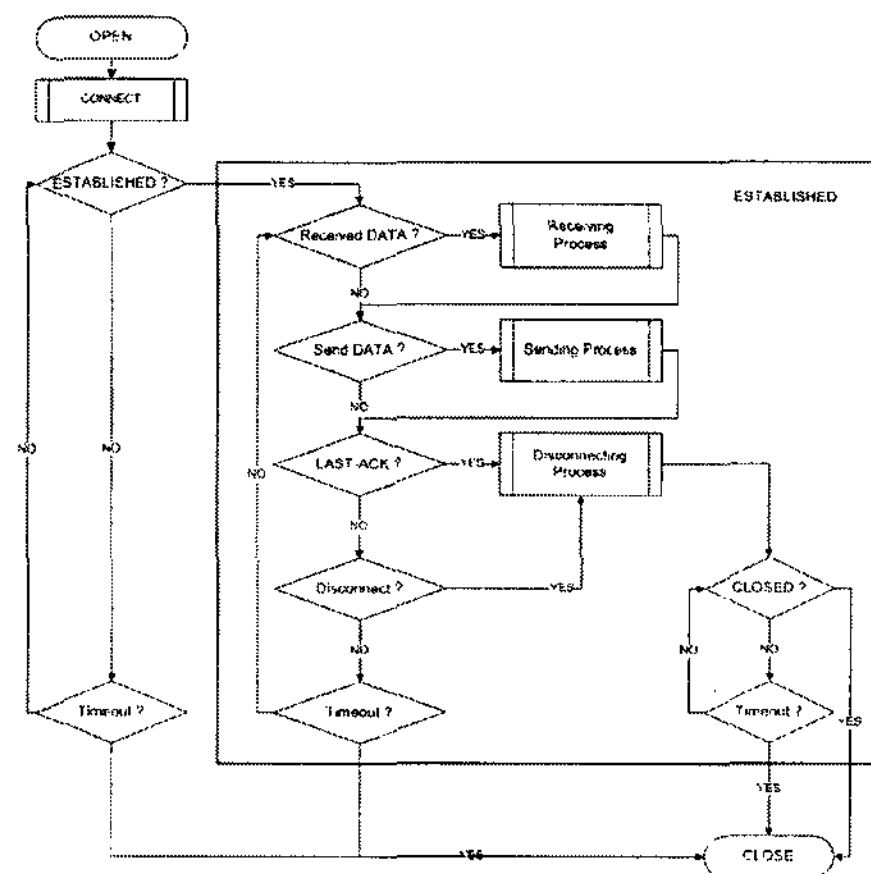


그림 5. TCP 클라이언트 모드
Fig. 5 TCP Client Mode

웹 서버는 웹 브라우저로부터 들어온 HTTP 요청 방법과 URI(Uniform Resource Identifier)를 분석하고 URI가 웹 페이지 요구시 해당 페이지를 찾아 전송하면

CGI(Common Gateway Interface)와 같은 Action을 요구하는 경우에는 이에 해당하는 Action을 취한 뒤 그 결과를 웹 페이지 형태로 알려준다[11].

그림 6은 웹서버 MAIN()함수의 흐름도이다.

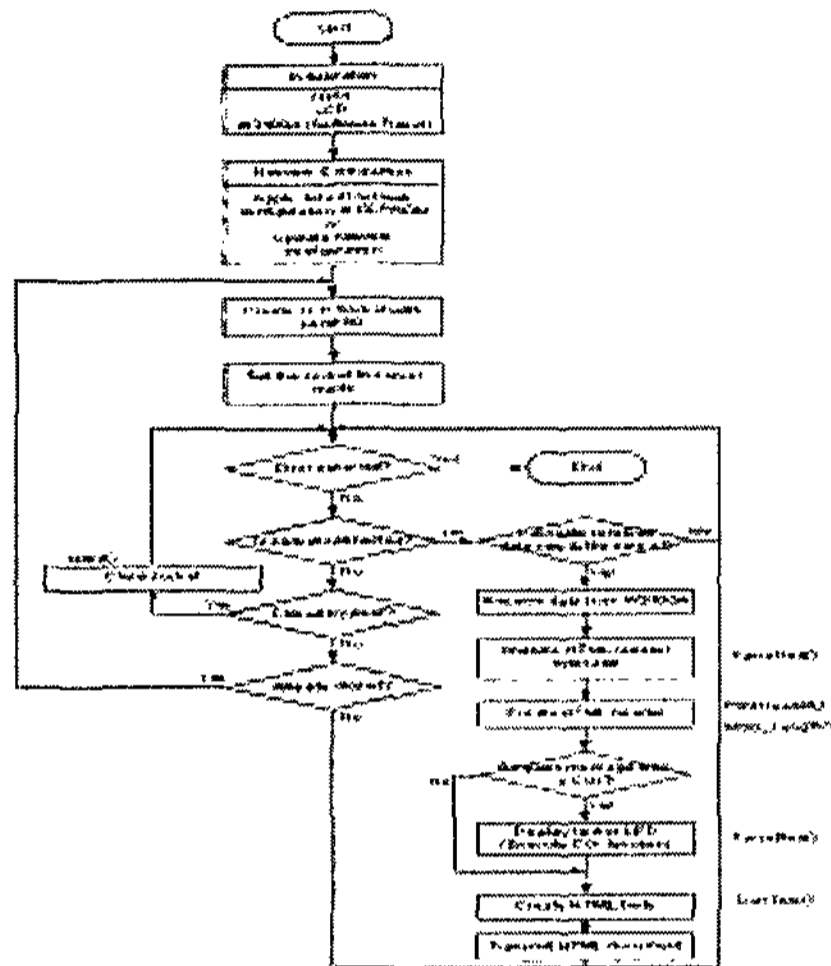


그림 6. 웹서버 MAIN() 함수의 흐름도
Fig. 6 Flowchart of Web Server Main Function

IV. 임베디드 이더넷 모듈 구현

4.1 개발환경

본 논문에서 제안한 모듈의 개발환경으로는 윈도우즈 XP Professional 2002, 펜티엄 IV 3.0GHz이다. 사용된 소프트웨어는 C와 HTML이며, keil 8051 컴파일러 v5.20로 컴파일 한 후 HEXA 파일을 시리얼 통신으로 개발한 모듈에 다운로드 하였다. 그리고 모듈의 IP는 210.110.180.108로 설정하였다.

4.2 웹서버 프로그램

본 모듈을 동작하기 위해 웹서버 응용 프로그램은 W3150A 소켓 API를 이용하였다.

그림 7에 웹서버 main()함수코드를 나타내었다. 그림 7에서 (1)은 8051을 초기화하고 (2)는 W3150A의 네트워크 부분을 설정한다.

초기화 과정이 끝나면 (3)을 통해 웹서버가 사용하게 될 소켓을 HTTP 서버 Port(80)로 대기시킨다. 그 다음에 (4)를 통해 각 소켓의 상태를 관찰하고 웹 브라우저로부

터 HTTP 요청이 수신되었는지 관찰한다. 만일 HTTP 요청이 수신되었을 경우 (5)를 통해 해당 HTTP 요청을 수신하고, 수신된 HTTP 요청 메시지는 (6)을 통해 분석되며, (7), (8), (9)를 통해 HTTP 응답 메시지를 전송하게 된다.

```

void main(){
    SOCKET i;
    int len;
    u_char state, type;
    Init8051();
    PutStringLn(" HTTP Server Program: ");
    InitLcd();
    InitW3150();
    InitNetConfig();
    for (i = 0; i < MAX_SOCKET_NUM; i++) init_sock(i);
    while(1) {
        i = 0;
        for(i=0; i<MAX_SOCKET_NUM; i++)
        {
            state = select(i, SEL_CONTROL);
            switch(state) {
                case SOCK_ESTABLISHED:
                    if ((len = select(i, 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;
                            default: break;
                        }
                    }
                    break;
                case SOCK_CLOSE_WAIT: close(); break;
                case SOCK_CLOSED: INT_SOCKET(); break;
            } //switch
        } //for
    } //while
}
    
```

그림 7. 웹서버 main함수의 소스 코드
Fig. 7 Source Code of Web Server main Function

4.3 웹서버를 이용한 원격 제어프로그램

오픈소스를 참고하여 웹서버에 I/O 제어를 위해 ParseReq()함수를 작성하였다.

그림 8은 원격제어를 위한 Parsereq()함수의 소스코드 일부분이다.

그림 8의 (1)과 (2)에서는 'led' 파라미터의 위치를 찾아내어 그에 해당하는 파라미터 값을 검사한 후에 해당 MCU I/O Port 값을 설정하게 되어 I/O 제어가 가능하다.

4.4 구현결과

웹 브라우저의 TEXTBOX에 “ON/OFF control” 문자열을 입력하고 제어하고자 하는 출력장치를 선택하고 OK버튼을 선택했을 경우

```

.....
while((Pointer = FindFirstStr(Pointer, ..led..)) != 0)
    // Parse switchselective value      (1)
{
    Pointer+=4;
    switch(*Pointer)
    {
    case '1':                // Device #1 is checked?
        LEDStatus |= 0x01;    // LEDStatus Update
        T0 = 0;    break;    // LED0 ON
    case '2':                // Device #2 is checked?
        LEDStatus |= 0x02;
        T1 = 0;    break;
    case '3':                // Device#3 is checked?
        LEDStatus |= 0x04;
        T0 = 0;
        T1 = 0;    break;
    }
    }
    P1 = LEDStatus;
    // Apply LED status to RELAY BOARD  (2)
    if (!(LEDStatus & 0x04))
    {
        if (!(LEDStatus & 0x01)) T0 = 1; // Device1 OFF
        if (!(LEDStatus & 0x02)) T1 = 1; // Device2 OFF
    }
    if (LEDStatus == 0) // Device ALL OFF
    .....

```

그림 8. 원격 제어를 위한 소스코드
Fig. 8 Source Code for Remote Control

요청 메시지가 모듈로 전송되면 MCU I/O Port의 값을 set 한 후 출력장치를 제어하게 된다. 그림 9는 실제 구현된 결과이다.

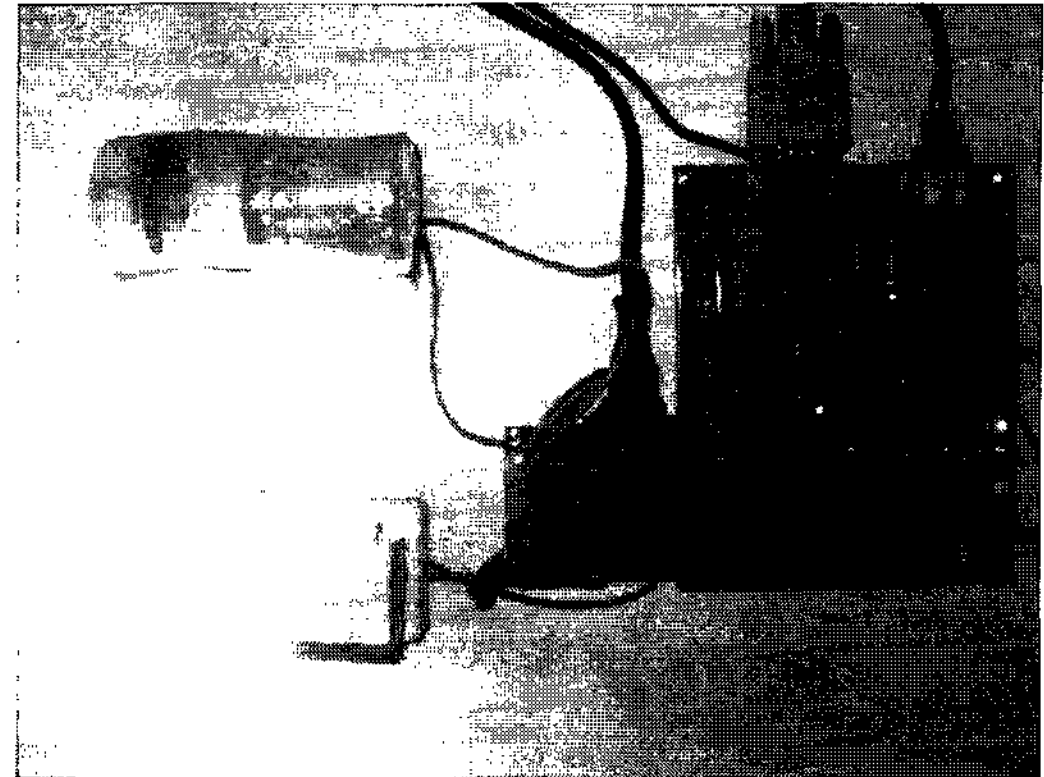


그림 9. 임베디드 이더넷 모듈의 구현 결과
Fig. 9 Result of Embedded Ethernet Module Implementation

V. 결론

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

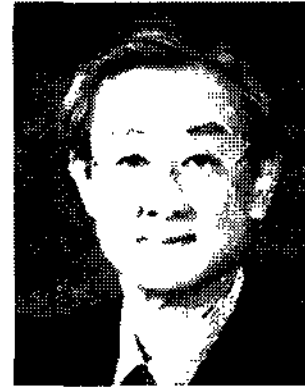
개발된 모듈의 기대 효과로는 언제 어디서나 컴퓨팅이 가능한 u-KOREA(Ubiquitous Korea)의 조기 실현이 가능하고, 홈 네트워크 시장의 틈새를 공략할 수 있을 것이다. 그리고 홈 네트워크가 지원이 되지 않는 기존 가전에 모듈을 삽입하여 네트워크가 가능하도록 하므로 많은 추가비용 없이 홈 네트워크의 구축이 가능하므로, 홈 네트워크 시장 활성화에 기여하리라 사료된다.

<This Work was supported by Dong-eui University Foundation Grant 2005>

참고문헌

- [1] 박용우, “홈네트워킹”, 정보통신기기, 293-318, 2001.10
- [2] 박용우, “국내 홈 네트워킹 시장구조 분석 및 기술 발전 전망”, 정보통신정책, 15권 20호, pp.26~49, 2003.11
- [3] 김진경, “홈네트워크 업계동향” Digital Contents 2 월호, pp71-81, 2004.2
- [4] 김동택, “홈 종합 정보화를 위한 홈 서버 개발”, 한국해양정보통신학회, 제 8권 제 1호, pp. 755-759, 2004
- [5] 오민정, “기기 제어를 위한 저가의 초소형 임베디드 웹서버”, 정보처리학회, 제 9권-A권 제 1호 pp.1-8, 2002. 3
- [6] 추광재, “임베디드 인터넷“ 임베디드 월드 6월호, pp82-87, 2004, 6
- [7] <http://www.internetnews.com/ent-news/article.php/3484511>
- [8] <http://www.eetimes.com/showArticle.jhtml?articleID=20000118>
- [9] <http://sample.co.kr/se8051icd/>
- [10] 김우열, “임베디드 인터넷 프로그래밍” 임베디드 월드 7월호pp.58-64 2004. 7.
- [11] 김우열, “임베디드 인터넷 응용 1”, 임베 디드 월드, 8월호, pp. 78-87, 2004. 8.

저자소개



김 영 진(yung-Jin Kim)

1968년 광운대학교 전자공학과 학사
1980년 건국대학교 전자공학 석사
1997년 경남대학교 전자공학과 박사

1977년~현재 동의대학교 전자공학과 교수

※ 주관심분야: 회로 및 시스템, 임베디드 시스템