

임베디드 웹 서버를 이용한 다축제어 시스템

김운업* · 주형민* · 곽균평*

*창원대학교

Multi Axis control system using Embedded Web Server

Yun-up Kim* · Hyung-min Chu* · Gun-pyung Kwak*

*Changwon National University

E-mail : firewolf@orgio.net

요 약

본 논문에서는 클라이언트 PC상에서 다축 모터를 제어하고 그 상태 데이터들을 모니터링 가능한 임베디드 웹 서버 기반의 다축 제어 시스템을 구성하였다. 하드웨어는 ARM 7312 CPU를 사용한 Nanowit사의 NanoWit2 보드를 사용하였다. 웹 서버는 이 보드에 리눅스 OS를 포팅하고 리눅스 프로그램인 GCC를 사용하여 구현하였다. 모션 컨트롤러는 이모션텍(주)의 MCS-40을 사용하여 웹 서버 보드에서 전송되는 제어 신호를 받아 다축 모터를 구동하고 상태 데이터들을 웹 서버 쪽으로 전송한다.

ABSTRACT

In this paper, we've studied multi axis control system using embedded web server which is controlling and monitoring the state data of motors. NanoWit2 Board has been ported Linux Operating System and fabricated Web Server by GCC. Motion controller is used MCS-40 of e-MOTION Tek Co., Ltd. which receiving control signal, controlling motors and sending the state data to Web Server.

키워드

웹 서버, 임베디드, 다축제어, TCP/IP

1. 서 론

산업이 발달되어 가면서 우리는 원격 제어 시스템들을 이용하여 다양한 자동화 설비들을 사용하고 있다. 그러나 대부분의 원격 제어 시스템들의 GUI (Graphical User Interface)는 특정 환경에 맞추어져 있거나 또는 특정 회사의 독점적 기술에 의해 구성되어 있다. 이러한 개별적 원격 제어 시스템들은 통일된 프로토콜(Protocol)을 사용하지 않기 때문에 각 자동화 설비마다 별도의 통신 선로를 설치하여야하며 이러한 선로는 거리의 제한을 받는다. 그러나 최근 빠르게 발전하는 인터넷 기술은 안전성, 편리성, 어디서나 접속이 가능한 용이성을 동시에 가지고 있기 때문에 사용자가 공간적 제약, 특정 GUI 환경 등을 극복하여 인터넷이 가능한 공간에서 웹 브라우저만으로 자동화 설

비들을 제어 할 수 있다. 하지만 일반 데스크 탑 컴퓨터로 인터넷을 이용하여 제어 시스템을 구현 할 경우 사용 장소에 따라 많은 제약을 받는다. 이러한 어려움을 극복하기 위하여 임베디드 시스템을 적용 시켰다. 그리고 임베디드에 채택되어지는 프로세스의 성능의 향상, 메모리 칩의 용량 증가 등의 영향으로 그 중요성은 더욱 증대되고 있다.

본 논문에서는 인터넷과 임베디드의 장점을 살려 임베디드 웹 서버에 다축 제어 장치를 구현하였다. 임베디드 웹 서버 기반 아래 TCP/IP Protocol를 구현하였고, 고유 IP를 부여하였다. 따라서 관리자가 인터넷 상에서 어떤 OS의 종류에 관계없이 직접 접근이 가능하게 한다.

이 임베디드 웹 서버는 기존의 인터넷망과 개인 컴퓨터의 웹 브라우저를 이용하여 임베디드 웹 서버가

장착된 기기를 원거리에서 제어할 수 있고 동시에 기기의 상태 확인이 가능하다. 또한 이 웹 서버는 장비의 설치, 유지가 간단할 뿐만 아니라 과도한 초기 투자비 절감 효과가 기대된다.

그리고 Java를 사용하여 웹 브라우저에 디스플레이 되는 웹 페이지를 구성하였다. 자바는 다른 응용 프로그램들과는 달리 실행파일을 만들더라도 사용할 PC에 프로그램을 셋업해야 하는 번거로움이 없는 장점을 가진다. 다시 말해 Java는 JVM(Java Virtual Machine)이 포함되어있는 Web Browser만 있으면 어디서든 실행이 가능하다. 그리고 OS의 종류에 구애를 받지 않기 때문에 더욱 적용할 만한 가치가 있다. 그리고 프로그램의 규모도 타 프로그램보다 작기 때문에 소형화 추세인 핸드폰, PDA 등에 적용하고자 많은 기업들이 연구하고 있다.

II. 본 론

1. 임베디드 시스템

임베디드 시스템이란 독립장비 혹은 내장화한 시스템이라고 말할 수 있다. 정확한 정의를 내린다면 미리 정해진 특정 기능을 수행하기 위해 컴퓨터의 하드웨어와 소프트웨어가 조합된 전자 제어 시스템이라고 할 수 있다. 즉, 일반 범용의 다양한 기능을 수행하는 컴퓨터로 사용되는 것이 아닌 가정, 빌딩, 공장의 제어 장비 또는 기계장치에 탑재되어 운영되는 시스템으로 기본적인 운영체제 커널이 포함되어 있으며 제어장비나 기계장치의 고유한 기능에 맞추어 정형화된 작업을 수행하는 특수 목적용 컴퓨터 시스템을 말한다.[1]

임베디드 시스템은 뭔가 굉장한 것 또는 어려운 것으로 비춰질 수 있지만 실제로 주변에 상당히 많이 있다. 자동차엔진제어 시스템, 엘리베이터제어시스템, 현금지급기시스템, 각종 자동판매기, 원격제어시스템 등의 산업적 용도 외에 PDA(Personal Digital Assistance), 가정용 셋톱박스, 휴대단말기, 게임기, mp3 플레이어, 일반 가전기기에 이르기까지 우리의 생활 곳곳에서 찾아볼 수 있다. 즉, 우리 생활에서 쓰이는 각종 전자기기, 가전제품, 제어장치는 단순한 전자회로로만 구성된 것이 아니라 마이크로프로세서가 내장되어 있고, 그 마이크로프로세서를 구도하여 특정한 제한된 기능을 수행하도록 프로그램이 내장되어 있으며 그런 시스템을 임베디드 시스템이라고 한다.[2]

임베디드 시스템은 보통 크기나 성능에 관계없이 마이크로프로세서가 삽입된(Embedded) 시스템을 총칭하지만, 일반적으로 32bit 이하의 마이크로프로세서를 사용한 시스템으로 그 범위를 한정한다. 보통

임베디드 시스템의 경우 전체 시스템 가격이나 소비전력을 낮추기 위해 시스템에 많은 제한을 가하는 특성이 있다. 그리고 범용 운영체제를 사용하기보다는 특화된 실시간 운영체제를 사용하거나 운영체제 없이 모니터 프로그램에 의해 로드되어 필요한 기능만을 수행하는 단일 프로그램을 사용한다.

2. 자바

자바는 C++를 모델로 하여 Sun Microsystems사에서 개발한 객체지향 프로그래밍 언어이며 개발환경이다.

자바는 운영체제와 관계없이 이식될 수 있도록 고안된 플랫폼 독립적인 언어이다. "플랫폼 독립적이다"는 것은 예를 들면 윈도우에서 돌아가는 워드프로세스를 자바로 만들었다면, 같은 프로그램을 매킨토시에서도 그대로 돌릴 수 있다는 말이다. 그러한 플랫폼에 자유로울 수 있는 이유는 각 플랫폼마다 각기 특성에 맞는 자바 가상 기계(JVM : Java Virtual Machine)를 배포하고 있기 때문이다. 하지만 Virtual Machine은 인터프리터 엔진이 아니며 컴파일 된 자바 class가 실행되도록 플랫폼에 적합한 환경을 조성해주는 셸(Shell)적인 부분이라고 보는 것이 더 타당하다.[3][4]

자바의 실행환경 즉 자바 플랫폼은 자바 API(Application Programming Interface)와 자바 가상 기계(Java Virtual Machine)로 이루어진다. 자바의 실행환경은 운영체제(Operating System)와 하드웨어(Hardware)를 감싸서 자바 프로그램이 동작할 수 있는 환경을 제공하는 역할을 하고, 이러한 환경에서는 모든 자바 프로그램이 동일하게 동작을 한다. 이러한 실행 환경으로는 자바 OS와 Virtual Machine이 이에 해당한다.[5][6]

3. TCP/IP

인터넷은 전세계에 흩어져 있는 네트워크들의 집합체이다. 이러한 네트워크들에 접속되어 있는 여러 다른 기존의 컴퓨터의 연결을 위해 이들의 통신을 가능하게 하는 프로토콜이 필요하게 되었는데 이를 위해 고안한 것이 TCP/IP(Transmission Control Protocol, Internet Protocol)다.[7]

메시지를 전송할 때 일단 메시지를 일정한 길이의 패킷(packet)으로 나누게 되는데 이 역할을 TCP가 하며, TCP는 패킷에 패킷 번호와 수신 측의 주소, 그리고 에러 검출용 코드를 추가한다.

패킷으로 쪼개진 메시지는 IP에 의해서 수신 컴퓨터로 보내지게 된다. 수신 측의 TCP는 에러 유무를 검사하고 에러가 발견되면 재전송을 요구하게 된다. 즉 TCP는 전송 데이터의 흐름을 관리하며 데이터의 에러 유무를 검사하고, IP는 데이터 패킷을 전송한다.

III. 시스템 환경 구축

1. 전체 시스템 구성

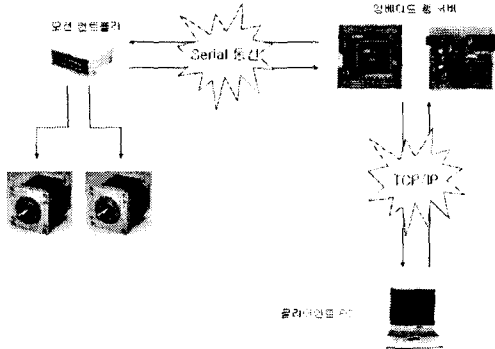


그림 1. 전체 시스템 구성도

그림 1에서처럼 임베디드 웹 서버 시스템은 클라이언트 PC와 TCP/IP 프로토콜을 이용하여 제어 신호를 주고받으며 모션 컨트롤러와는 시리얼 통신을 통하여 스텝 모터를 제어하게 된다. 본 논문에서는 임베디드 인터넷 기기인 Nanowit사의 NanoWit2 보드를 사용하였다. 인터넷에 연결된 PC에서 임베디드 인터넷 기기의 IP로 접속하면 모터들의 동작 상태를 확인할 수 있다.

2. 시스템 설정

2.1 IP 주소 설정

NanoWit2 보드에서 IP주소는 NanoWit2 유틸리티 소프트웨어로 설정할 수 있다. NanoWit2의 IP주소는 공장 출하시 0.0.0.0으로 설정되어 있으므로, 최초로 전원이 연결되면 BOOTP가 IP주소의 할당을 요구하므로, NanoWit Device 상의 TxLED가 주기적으로 깜박 거린다. NanoWit2의 IP주소 관리 및 설정은 IP주소 데이터베이스에 의하여 이루어지므로, 사용자는 적절히 할당된 NanoWit2의 MAC-IP 주소 레코드를 IP 주소 데이터베이스에 입력하여야 한다. NanoWit2의 IP 주소는 사용자의 네트워크 환경에 따라 사용가능한 IP 주소로 할당해야 한다. 여기서는 203.246.31.42로 설정하였다.

2.2 시리얼 통신 파라미터 설정(RS-232)

RS-232 통신을 위해서는 MCS-40에 연결된 시리얼 COM 포트와 NanoWit2 커넥터 사이에 공급된 RS-232 케이블을 연결한다. 그 뒤 파라미터를 설정한다. 이때, 파라미터 설정은 NanoWit2의 시리얼 파라미터 설정과 동일하게 한다.

3. System Flow Chart

그림 2는 전체 시스템의 flow chart이다. Nano - Wit2가 동작을 하면, 우선 웹 서버 프로그램이 실행되어, 소켓과 쓰레드가 초기화되고 자바 프로그램을 클라이언트에게 전송할 준비 상태가 된다. Ethernet을 통해 클라이언트가 접속을 했을 경우 서버는 클라이언트에게 자바 프로그램을 전송하고 클라이언트는 그 프로그램을 통해 서버로 제어 신호를 전송하게 된다. 서버는 시리얼 통신으로 그 제어 신호를 컨트롤러에게 보내게 되고 컨트롤러는 스텝 모터를 구동하게 된다. 그리고 그 상태들을 서버에게 다시 전송하게 되고 서버는 클라이언트에게 보내준다. 그래서 클라이언트는 그 상태들을 확인할 수 있게 된다.[8]

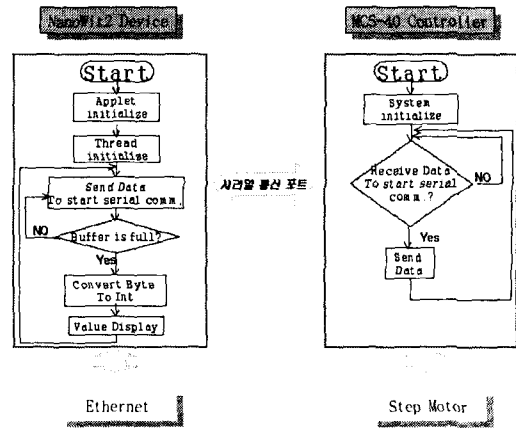


그림 2. 시스템 흐름도

IV. 실험 결과

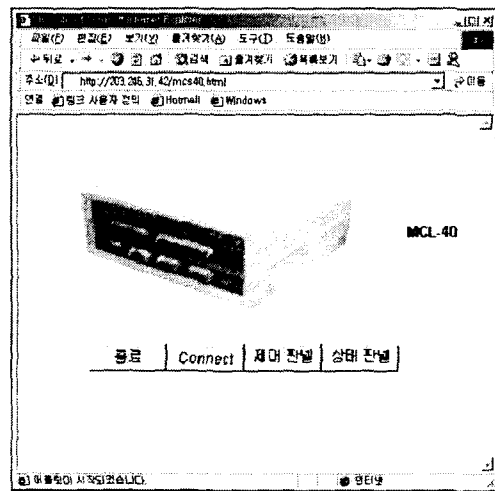


그림 3. 서버 프레임

클라이언트에서 웹 서버로 접속을 할 경우 그림 3과 같은 웹 브라우저를 볼 수 있다. 종료 버튼은 서버와 접속을 끝낼 때, Connect 버튼은 서버와 소켓 통신을 연결할 때 사용한다.

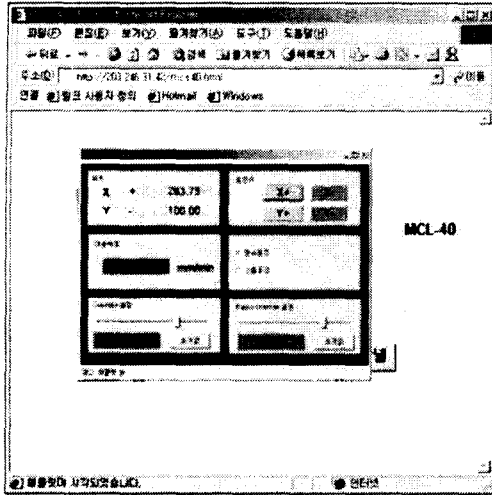


그림 4. 제어 판넬 프레임

그림 3에서 제어 판넬 버튼을 클릭하면 그림 4와 같은 프레임이 나타나게 된다. 이 판넬 프레임에서 모터를 제어하게 된다.

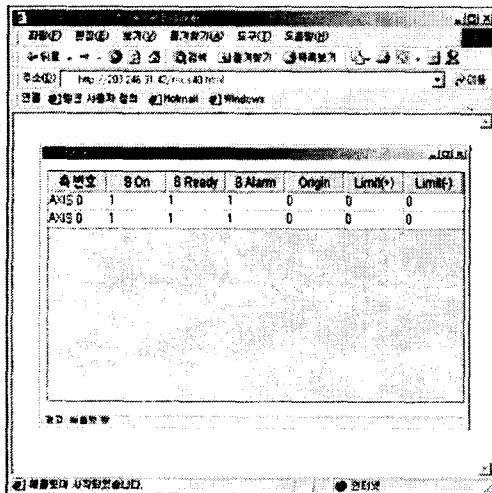


그림 5. 상태 판넬 프레임

그림 3에서 상태 판넬 버튼을 클릭하면 그림 5와 같은 판넬 프레임이 나타나며 컨트롤러의 상태를 나타내게 된다. 그리고 직접 상태를 제어할 수 있다.

V. 결 론

최근 통신 매체와 인터넷의 발달은 모든 기기의 네트워크화를 확산시키고 있다. 이런 현상은 산업 현장에 있는 기기들에도 적용되고 있는데 지금까지 이루어지고 있던 산업 기기의 네트워크화는 PC기반에서 이루어지고 있었는데 본 논문에서는 이를 임베디드화하여 원격지에서도 모션 컨트롤러들을 제어할 수 있는 임베디드 웹 서버를 연구, 실험하였다. 이 웹 서버를 산업 현장에 응용하게 된다면 경제적, 시간적, 공간적 부담을 줄일 수 있으리라 생각된다.

참고문헌

- [1] 이상협, "웹을 이용한 홈 오토메이션 시스템의 개발", 영남대학교 대학원 학위논문, 2000.
- [2] 이보영, "임베디드 시스템을 위한 GNU C 라이브러리", 전국대학교 대학원 학위논문, 2001.
- [3] 김충석, "알기 쉽게 해설한 자바", 이한출판사, 2001.
- [4] H.M.Deitel, P.J Deitel, 손진욱 편저, "자바2 프로그래밍", 정보문화사, 2000.
- [5] 이현우, 김형국, 홍성민, "자바 프로그래밍 바이블 ver.2", 영진출판사, 2000.
- [6] Lemay, Cadenhead, "초보자를 위한 Java2", 인포북, 1999.
- [7] 안광혁, "원격 감시를 위한 내장형 소형 웹서버", 아주대학교 대학원 학위논문 2001.
- [8] Kurt Wall, "Linux Programming 언리쉬드", 정보문화사, 2000.

본 연구는 과학기술부·한국과학재단 지정 창원대학교 공작기계기술연구센터의 지원에 의한 것입니다.