

## Web Browser를 이용한 PLC 제어

°김지은\*, 박준명\*\*, 이강현\*

\* 조선대학교 전자정보통신공학부

멀티미디어 ASIC 설계/Bio-Information Lab.

\*\* 충북대학교 컴퓨터공학과

### The PLC Controlling on Web Browser

° Ji Eun Kim, Kang Hyeon Rhee

School of Electronics & Info-Comm. Eng.,

<http://multimedia.chosun.ac.kr>

{angel,khrhee}@vlsi.chosun.ac.kr

#### Abstract

In this paper, "The PLC Controlling on Web Browser" is designed and implemented to remote control and to monitor FA system.

Protocol bidirectional communication with PLC is written and setted. Then Administration system sets Web server and JAVA Sevelet Program.

The client system located at the monitoring Web browser are a JAVA Sevelet base system, and sends some commands to the server system which guards and controls the PLC system. The commands are delivered to the server system and the driver software exists in the server controls the PLC system directly.

#### I. 서론

생산 자동화기술은 우리나라 모든 생산 분야에 근간을 이루는 기술로서 특히 오늘날과 같이 국경이 없는 무한 경쟁시대 체제 하에서 가격경쟁력이 있는 제품을 만들어 내는 데 필수적인 요소이다. 생산 자동화기술은 제품을 생산하는 공정 모두를 포함하는 넓은 분야이므로 이러한 기술은 당연히 전기, 전자분야는 물론 컴퓨터, 기계, 화공, 철강, 재료, 산공 등 타 공학분야

본 논문은 KOSEF RRC(조선대학교 수송기계부품 공장자동화 연구센터)의 2001년 지원에 의한 일부 연구입니다.

어디에서나 그 응용 예를 찾아볼 수 있다. 그러나 기존의 자동화기술은 전화나 모뎀을 통하여 접속하여 제어신호를 보내는 방식과 직렬 통신을 이용하여 서로 데이터를 주고받는 방식이 있는데, 직렬 통신(RS-232, 422, 485)을 이용한 양방향 제어방식의 경우 거리상의 제한이 있고, 전화를 이용한 단방향 제어방식은 제어 대상의 정확한 확인이 불가능하다는 문제점을 지니고 있었다.

본 논문에서는 산업 및 가정용 기기를 관리하는 양방향 제어 방식에 거리 상에 제한과 정확한 제어가 되 있는가를 감시(monitering)할 수 없는 문제점을 해결하고자 인터넷이나 정보 통신망을 이용하여 지역적으로 멀리 떨어져 있는 기기에 대한 실시간 제어 및 감시를 위하여 PLC(Programmable Logic Controller)와 양방향 통신할 수 있도록 관리 시스템을 웹 서버(Web server)로 구축하고 JAVA 환경의 프로그램을 설치하여 웹 브라우저(Web Browser)를 통한 제어와 감시가 가능한 원격 PLC 제어 시스템을 구현하고자 한다.

#### II. 이론적 배경

##### 2.1 PLC의 기본 구조

PLC란, 종래에 사용하던 제어반 내의 릴레이 타이머, 카운터 등의 기능을 IC, 트랜지스터 등의 반도체 소자로 대체시켜, 기본적인 시퀀스 제어 기능에 수직연산 기능을 추가하여 프로그램 제어가 가능하도록 한 자율성이 높은 제어 장치이다. 미국 전기 공업회 규격(NEMA: National Electrical Manufacture Associati-

on)에서는 "디지털 또는 아날로그 입출력 모듈을 통하여 로직, 시퀀싱, 타이밍, 카운팅, 연산과 같은 특수한 기능을 수행하기 위하여 프로그램 가능한 메모리를 사용하고 여러 종류의 기계나 프로세서를 제어하는 디지털 동작의 전자 장치"로 정의하고 있다.

PLC는 마이크로프로세서(microprocessor) 및 메모리를 중심으로 구성되어 인간의 두뇌 역할을 하는 중앙처리장치(CPU), 외부 기기와의 신호를 연결시켜 주는 입·출력부, 각 부에 전원을 공급하는 전원부, PLC내의 메모리에 프로그램을 기록하는 주변 장치로 구성되어 있다. 그림 1은 PLC의 전체 구성도를 나타낸 것이다.

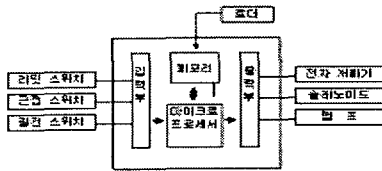


그림 1. PLC 전체 구성도

PLC의 CPU 연산부는 PLC의 두뇌에 해당하는 부분으로서, 메모리에 저장되어 있는 프로그램을 하나씩 꺼내서 해독하여 처리 내용을 실행한다. 이 절차는 빠른 속도로 반복되며, 모든 정보는 2진수로 처리된다.

### 2.2 Java Servlet의 인터페이스

자바 서블릿은 시스템 구조에 있어서 중요한 역할을 담당한다. 서블릿은 응용 처리(Application processing)을 중간계층에 할당해주고, 클라이언트에 대해 프록시로서의 임무를 수행하고, 새로운 프로토콜이나 다른 여러 특징들을 추가함으로써 중간계층의 특징을 증가시키기도 한다. 중간계층은 클라이언트/서버 시스템에서 응용 서버로서의 임무를 수행하면서, 웹 브라우저와 같은 클라이언트와 데이터 소스의 중간에 위치한다.

많은 시스템에서 중간계층은 클라이언트와 back-end 서비스간을 연결해주는 임무를 수행하게 되는데, 이러한 중간계층의 사용으로 클라이언트와 서버 모두의 부담을 동시에 줄일 수가 있다. 중간계층 처리의 장점은 단순한 연결 관리를 들 수 있다. 서블릿은 데이터베이스 서버와의 연결 같은 부담이 큰 연결에 대해 재사용을 함으로써 수백개의 클라이언트와의 연결을 관리할 수가 있다.

만약 애플릿에서 다른 시스템에 위치해있는 데이터베이스 서버와의 연결이 필요하다면 서블릿은 애플릿

의 한 부분으로서 이러한 연결을 가능하게 해준다. 또한 HTTP 프로토콜을 처리하고, 이 클래스로 서블릿들을 불러내기 위해 웹 서버를 사용하게 될 것이다. 서블릿은 웹 서버 시작 시에 자동으로 로드될 수 있기 때문에 자동적으로 쉽게 로딩(loading)하는 데에 이용되어질 수 있고, 이러한 프로그램은 전적으로 웹 서버에 독립적이고 특별한 기능을 제공할 수가 있다.

### 2.3 프레임 구성

프레임은 데이터 통신에서 송수신 자료를 일정한 형식으로 구성한 것이다. 이는 세그먼트[국번, 명령, 명령에 따른 파라미터], 동기화(同期化)를 위한 제어 문자 [ENQ] [ACK] [EOT] [ETX] 오류 검출을 위한 패리티, BCC 등의 추가 정보를 포함한다.

PLC Cnet의 시리얼 통신에 사용되는 프레임의 구조는 그림 2와 같다.

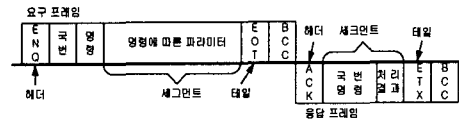


그림 2. PLC Cnet의 시리얼 통신

Cnet은 컴퓨터 링크 모듈(Computer Link Module)을 사용한 PLC 네트워크 시스템이다. 컴퓨터 또는 각종 컨트롤러, 타 기종의 PLC 등 다양한 프로토콜의 통신 장치와 통신을 하기 위한 기능과 모뎀을 통한 원거리의 PLC를 제어할 수 있는 기능을 갖추고 있다.

## III. 제안된 PLC 제어 프로그램 설계

### 3.1 프로토콜 신호

표준 RS-232C신호는 표 1의 (a)와 같다. 본 논문에서는 표준 RS-232C의 신호 배치선을 LG PLC의 Cnet 프로토콜에 적용하기 위하여 (b)와 같이 구성하였다.

표 1. 프로토콜신호  
(a) 표준 RS232C 신호

신호	약자	커넥터 상의 핀 번호
Request to send	RTS	4
Clear to send	CTS	5
Data set ready	DSR	6
Data terminal ready	DTR	20
Transmit data	TxD	2
Receive data	RxD	3
Ground	GRD	7

## Web Browser를 이용한 PLC 제어

### (b) LG PLC Cnet 프로토콜

신호	약자	커넥터 상의 핀 번호
Request to send	RTS	7
Clear to send	CTS	8
Data set ready	DSR	6
Data terminal ready	DTR	4
Transmit data	TxD	3
Receive data	RxD	2
Ground	GRD	5

윈도우 환경에서 동작하는 프레임 편집기를 사용하여 통신모드 및 프로토콜을 직접 작성하여 통신 장치와 접속이 용이하도록 하였다. 통신모드 및 프로토콜을 RS-232C 채널에 작성하여 채널로 동작이 가능하다. 전용 프로토콜을 이용하여 변수 읽기/쓰기 및 프로그램 읽기/쓰기로 최대 32대 접속이 가능한 멀티드롭을 구성하였다.

RS-232C/RS-422(RS-485) 통신포트를 독립채널 또는 연동채널로 설정하여 사용이 가능하다. 300bps에서 76,800bps 까지 통신속도를 설정할 수 있다. 1:n 및 n:m 방식의 통신 방식을 지원한다.

PLC가 동작 중일 때는 RUN모드이며, 정지 상태일 때는 STOP모드, 프로그램이나 기계적인 오류가 있을 때는 ERROR 모드이다. 웹 페이지 상에 RUN, STOP, ERROR모드를 표현할 수 있고, RUN, STOP모드를 컨트롤로 작동시킬 수 있도록 웹 서버로 데이터 통신에서 송수신 자료를 일정한 형식으로 프레임을 구성하였다.

### 3.2 PLC 상태 모니터링



그림 3. 일반 PLC 래더 프로그램 결과

그림 3은 입력 자체를 출력으로 보내는 것이다. 예를 들어 입력(1010)이 들어오면 출력이 입력과 같게 출력(1010)을 한다.

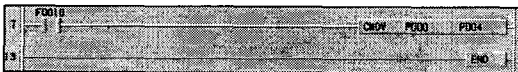


그림 4. 입력 반전 PLC 래더 프로그램 결과

그림 4는 입력을 반전하여 출력으로 보내는 것이다. 예를 들어 입력(1010)이 들어오면 출력은 입력이 반전되어 출력(0101)을 한다.

## IV. Web Server의 구성 및 PLC 제어

### 4.1 자바 서블릿 위한 서버 셋팅

현재, 자바 서블릿을 지원해 주는 웹 서버와 서블릿을 개발하기 위해 사용할 수 있는 도구들은 상당히 많다. 이 중에서 윈도우 2000 서버의 IIS(Internet Information Server) 5.0(기본 설치)과 JSDK2.1(Java Servlet Development Kit)을 기준으로 하여 서블릿을 개발하고 하였다.

서블릿은 c:\JavaWebServer1.1\servlets 디렉토리로 서블릿을 이동시키는 것으로 설치되어진다. 즉, 자바 웹 서버는 서블릿이 이 디렉토리에 추가되면 자동으로 인식하게 된다. 서블릿은 웹서버와 같이 같은 프로세스의 한 부분으로서 웹서버 플랫폼에서 돌아간다. 일반적으로 웹 서버는 80 포트, 자바 서버는 8080 포트에서 서비스를 수행한다.

애플릿이 웹 브라우저를 위한 것이라면 서블릿은 웹 서버를 위한 것이다. 애플릿은 웹 브라우저에서 실행이 되고, 특수한 요청을 통해서 수행되는데, 서블릿의 경우에는 모든 것이 동일하고 단지, 웹서버에서 실행된다는 것이 애플릿과 다를 뿐인 것이다.

### 4.2 PLC 제어 자바 서블릿 설계

Inline HTML 생성 자바 웹 서버같이 몇몇 웹 서버에서는 HTML파일에 직접 서블릿 태그를 입력하는 것이 가능하다. 서버가 이러한 태그를 발견하게 되면 클라이언트에 이 파일을 보내는 동안 서블릿을 호출하게 된다. 서블릿 엔진은 서블릿의 실행 환경에 대한 정보를 제공해 주는 ServletContext 객체를 리턴 해줌으로써 서블릿과 통신한다. 이때, 서블릿은 ServletConfig.getServletContext() 메소드를 이용하여 ServletContext 객체를 얻는다. ServletContext 객체는 서블릿이 초기화 될 때 웹 서버가 제공하는 ServletConfig 객체에 포함되어 있다. 따라서, Servlet.getServletConfig 메소드를 이용하여 이 ServletConfig 객체를 참조한다.

HttpServlet 클래스는 HTTP 서블릿을 생성할 수 있도록 기능을 제공해 주는 추상클래스이다. HTTP 서블릿은 일반적으로 웹 클라이언트로부터 온 요청을 받아서 처리하고, 그 결과를 다시 웹 클라이언트에게 되돌려 주는 작업을 수행한다. 또한 HttpServletRequest 인터페이스를 확장하는 HttpServletRequest 인터페이스는 HTTP 방식을 사용하는 HTTP 서블릿에 정보를 전달하기 위한 추가적인 기능을 제공한다.

서블릿 엔진은 클라이언트로부터 HTTP프로토콜을 사용하여 전달된 정보를 HttpServletRequest의 service 메소드에 전달하기 위해 HttpServletRequest 객체를 생성한

다. HttpServletRequest 인터페이스는 이렇게 HTTP 서버로부터 클라이언트로부터 온 정보를 전달하기 위해 필요한 기능들을 제공한다. 그림 5는 자바 서버릿 소스의 컴파일링(compiling) 과정이다.

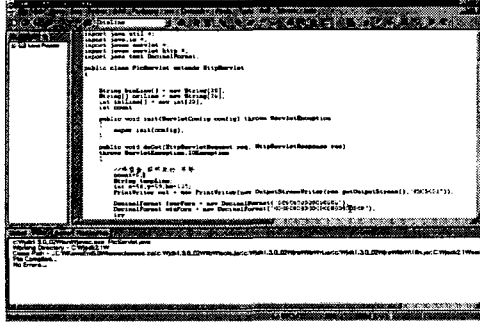


그림 5. 자바 서버릿 소스 컴파일러

자바 서버릿 소스 컴파일링한 서버 시스템은 클라이언트 시스템에서도 PLC의 Input, Output 상태를 수시로 점검할 수 있을 뿐 아니라 강제로 임의의 Data를 입력하여 PLC에 입력시킬 수 있다. 클라이언트 시스템은 네트워크 연결관리자를 통해 PLC가 있는 서버 프로세스와 연결하고, 주어진 이벤트는 이벤트 관리자에서 인식하여 자바 서버릿으로 처리된 결과를 네트워크 연결관리자를 통한 웹 브라우저로 모니터링이 가능하다. 그림 6은 자바 서버릿을 이용한 웹 브라우저상의 PLC 제어 상황으로 네트워크가 있는 환경에서는 원격 제어 모니터링이 가능하다.

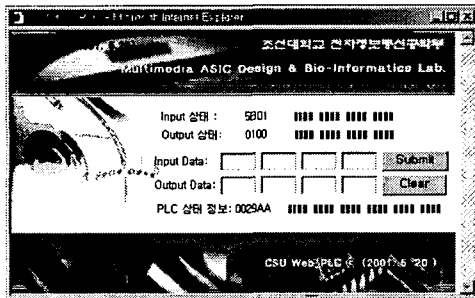


그림 6. 자바 서버릿을 이용한 Web Browser상의 PLC 제어 상황

서버 시스템은 PLC 제어 현장에 위치하며 네트워크 관리자, 이벤트 관리자, 결과 관리자의 4개의 프로세스로 구성되어 있다. PLC를 실시간 체크하여 기록하고 있다가 클라이언트가 요구할 때 기록 상황을 전송하여 준다.

### V. 결론

본 논문에서는 산업 및 가정용 기기를 관리하는 양

방향 제어 방식에 거리 상에 제한과 정확한 제어가 되 있는가를 감시 할 수 없는 문제점을 정보 통신망을 이용하여 지역적으로 멀리 떨어져 있는 기기에 대한 실시간 제어 및 감시를 위하여 PLC와 양방향 통신할 수 관리 시스템을 웹 서버로 구축하고 JAVA 환경의 프로그램을 설치하여 웹 브라우저를 통한 제어와 감시가 가능한 원격 PLC 제어 시스템을 구현하였다.

이러한 양방향으로 감시, 제어가 가능하게 됨으로써 초고속 정보통신망에서 원격제어 및 모니터링뿐만 아니라 물류 및 통합공정 시스템 등 다양한 분야에 활용될 수 있는 기술의 기반이 될 수 있다. 원격지 클라이언트 시스템과의 보안을 위해서는 암호화가 가능해야 하는데, 이 시스템의 개발을 기반으로 안정적인 인터페이스에 관한 개발의 가속화를 기대할 수 있게 된다.

또한 실시간 원격 시스템으로 전송 프로토콜과 보안을 강화하여 공장자동화에 적용될 수 있는 시스템으로 개선되어 갈 것으로 기대하며 생산공정에 관한 기술적 인 협력을 도모하여 실용방안의 모색이 절실히 필요하다.

### Reference

- [1] Laplante P.A., Real-Time Systems Design and Analysis. An Engineer's Handbook., IEEE Press, New York, 1993
- [2] Dr. J. Mills. D. Goldstein, C. Lindahl, "A Flexible, Modular Computer Architecture for CIM," ICCIM, 1991.
- [3] G. J. Park, S. H. DO, and J. L. IM, "Design of Cell Controller from the software development viewpoint," presented at the 12th IFAC World Congress, July, 1993.
- [4] The Java Language Environment: A white Paper. Sun Microsystems, Inc. 1995.
- [5] Wilner, D., Personal conversation at ACM Programming Languages Design and Implementation Conference, July, 1995.
- [6] Nilsen, K., Real-Time No Longer a Small Specialized Niche, Fifth Workshop on Hot Topics in Operating Systems (HotOS-V), 1995. Orcas Island, Washington: IEEE Computer Society Press.