

웹 기반 온실 환경 자원의 원격 제어를 위한 실시간 모니터링

◦김대업* · 박홍복*

*부경대학교 전자·컴퓨터·정보통신공학부

Real-Time Monitoring of Remote Control for Greenhouse Environmental Resources based on Web

◦Dea-Up Kim*, Hung-Bog Park*

*Div. of Electronic, Computer and Telecommunication Eng.
PuKyong National University

요 약

본 논문에서는 클라이언트/서버 환경에서의 온실 환경 자원을 원격지에서 제어하는 실시간 모니터링 방법을 제안하고 구현하였다. 자바의 JApplet과 자바의 경량 컴포넌트로 구현된 본 시스템은 웹을 사용하여 관리자의 인증을 거친 후 직접적으로 제어를 담당하는 서버에게 제어 명령을 하게 된다. 제어 서버는 자바 확장 패키지 JFC와 Commucation API를 이용하였고, 클라이언트는 JApplet과 JFC를 이용하였으며 서버는 IIS 5.0과 ASP를 이용하여 웹 서비스를 한다.

1. 서론

최근에 웹을 기반한 원격 감시 및 제어 시스템 개발에 관한 연구가 활발히 진행되고 있다. 환경 감시, 전력 설비, 무인 공장, 원자력 제어, 보안 시스템 등과 같이 사람이 현장에서 직접 제어하기 어려운 분야에 특성상 컴퓨터의 이용은 필수적이다[1]. 특히 인터넷이나 인트라넷을 기반으로 웹을 이용하는 시스템 환경 구축에 관한 연구가 활발히 진행되고 있다.

정보 통신의 발달로 인한 개인 농가에서도 컴퓨터의 보급이 널리 확대되고 있으므로, 컴퓨터를 이용한 원격지의 제어로 효율적인 생산 시스템의 필요성이 점점 증가하고 있다 [2]. 우리나라의 시설 원예는 재배 면적에서 본다면 일본 다음으로 세계에서 두 번째이고, 양적인 면에서는 우리나라 시설 원예의 규모는 결코 작지 않음을 알 수 있다. 그러나 질적인 면에서는 그 수준은 많이 미흡하다. 또한 시설 원예의 시장 개방에 맞춰 시설 원예 관련 종사자들에게 좁은 국토에 제한된 환경 속에서 작물 생산과 비용에 지대한 관심을 가지고 있다. 더욱이 농촌 지역의 생산 인력의 고령화로 인하여 공간 및 시간 제약적인 생산 환경에 적절한 대응을 하지 못하고 있다[3][4].

더구나 기존의 온실 환경 제어 시스템 및 농산물 팔러 관련 전자 상거래의 경우 사용자가 일일이 확인되어야만 하고 장소의 한계성을 극복하지 못하고 있다. 현재 시설 면적의 대부분을 차지하고 있는 비닐 온실의 경우 사용자가 항상 해야 하는 번거로움이 있으며 관리자가 원격지에 있는 경우 이상 기후로 인한 농작물의 피해가 많다. 또한, on/off 식의

제어만을 사용함으로써 발생하는 하드웨어적인 오류, 야간 또는 원격지에서의 온실 상황은 제어가 불가능하다. 이러한 문제를 해결하기 위해서 기존의 연구[5]에서는 시뮬레이션 환경에서 작물 생산 과정에 필요한 정보를 수집, 감시, 저장, 제어를 수행하는 효율적인 생산 시스템을 설계하고 시뮬레이션하였다. 기존의 논문[5]에서 구현된 온실 관리 시스템은 연동 온실의 확장성과 지역 온실과의 통신, 원격지 모니터링에 대한 모듈들이 하나의 Application으로 통합 설계되었지만, 원격 제어는 불가능하다.

따라서, 본 논문에서는 웹을 기반으로 온실 환경 자원에 대한 원격 제어를 구현하였다. 또한 기존 시스템의 단순한 GUI를 통한 정보 제공 방식과 하드웨어적인 결함으로 인한 농작물의 피해를 최소화하기 위한 단일 클라이언트 모니터링의 시간적, 공간적인 한계를 극복하기 위해 웹을 기반한 실시간 온실 환경 제어와 정보의 시각적 표현에 중점을 두었다.

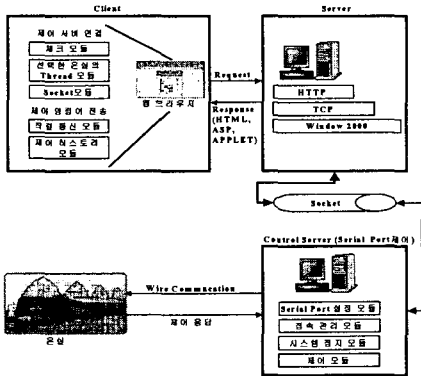
2. 구현 방법

본 논문의 구현 환경은 운영 체제로서 Windows 2000, 웹 서비스를 하기 위한 IIS 5.0, 소프트웨어는 OS에 독립적인 jdk 1.3, JBuilder 4.0, Windows 2000 서버와 호환이 잘 되는 ASP 3.0, 그리고 웹 에디터로는 DreamWeaver 3.0을 사용하였다.

시스템의 구성은 (그림 1)과 같이 서버 모듈, 컨트롤 서버 모듈과 클라이언트 모듈로 구성된다. 서버 모듈은 window2000 OS상에서 HTTP와 TCP를 사용하여 웹 서비스를 하며 사용자와의 인터페이스를 제공한다. 클라이언트 모듈은 세부적으로 GUI 모듈, 서버의 상태를 체크하는 체크 모듈, 사용자가 선택한 온실의 Thread 생성 모듈, 직렬 통신을 하기 위한 모듈, 서버에 접속하기 위한 Socket 모듈,

* 이 논문은 2001학년도 두뇌한국21사업에 의하여 지원되었음

사용자 선택한 온실 환경에 대한 제어 모듈, 그리고 사용자가 선택한 제어 히스토리 모듈로 구성된다. 컨트롤 서버 모듈은 세부적으로 GUI 모듈, 제어 명령어를 처리하기 위한 직렬 포트 설정 모듈, 웹으로 접속되어 오는 클라이언트/서버의 접속 관리 모듈, 클라이언트와 접속하기 위한 ServerSocket 생성 모듈, 서버의 기능을 정지하기 위한 시스템 정지 모듈, 그리고 제어 모듈로 구성된다.

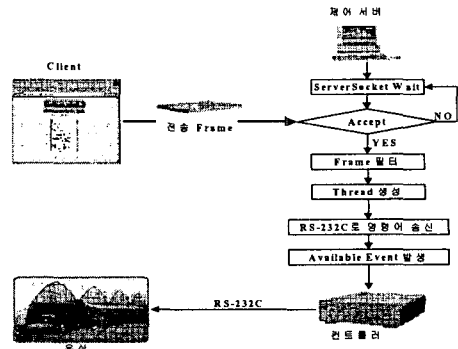


(그림 1) 원격 제어 시스템의 구성도

웹 기반의 원격 제어 시스템은 클라이언트/서버 환경하에서 제어를 수행한다. 클라이언트인 온실 사용자들은 자신의 온실을 제어하기 위해서 웹 브라우저를 통해 Applet으로 정보를 전송하며, 서버는 전송된 정보를 Socket을 통해 제어 명령어를 제어 서버로 송신한다. 제어 서버는 현재 접속되어 있는 IP와 시간을 GUI 화면에 표시한다. 또한 수신된 명령어를 직렬 포트 제어 모듈로 전송하고 온실 환경 자원을 제어한다. 제어가 완료된 후 컨트롤러의 정상적인 응답이 오면 서버는 ServerSocket을 통해 다시 클라이언트로 정상 동작 메시지를 보낸다. 이 메시지를 수신함으로써 원격지의 클라이언트 관리자는 정상적인 처리임을 알 수 있다.

제어 대상으로는 온실의 환경 제어 자원인 천창 좌1, 천창 우1, 천창 좌2, 천창 우2, 측창 좌1, 측창 우1, 측창 좌2, 측창 우2, 커튼, 환풍기, 그리고 관수 장치 등이며, 이들은 온·습도, 풍향, 풍속, 일사량, Co2 발생량을 센서로부터 입력받은 데이터에 의해 제어된다. 이러한 제어 대상들은 웹 서버와 제어 서버를 통해 제어가 수행된다. 웹 서버는 ASP를 통해 웹 페이지 서비스를 하며, 플랫폼에 독립적인 자바 Applet과 자바의 확장 패키지인 경량 컴포넌트 JFC를 사용하여 서버와 클라이언트의 부하를 최대한 줄였다. 사용자가 웹 브라우저를 통해 접속시 제어 서버의 활성화 여부를 체크하고 관리자 인증을 거친 후 정상적이면 선택한 온실 환경 자원 제어 명령을 프레임 형식으로 Socket을 통해 제어 서버로 전송된다. 제어 서버는 접속한 클라이언트의 정보를 Vector Class에 저장하고 접속 시간을 HashTable에 저장한다. 접속 해제시 Vector Class에 있는 정보를 이용하여 HashTable의 접속 시간을 삭제한다. 제어 명령어를 수신 받은 후 명령어를 직렬 포트 제어 모듈로 전송하고 RS-232C를 통해 온실 환경 자원 제어 컨트롤러로 명령어를 전송한다. 직렬 포트 Available Event가 발생하면 하드웨어로 제어 명령어를 송신한 후 클라이언트로 정상 동작의 메시지를 ServerSocket을 통하여 전송한다. (그림 2)는 제어 서버의 처리를 나타내며, (그림 3)은 전송 프레임의 구조를 나타낸다.

제어 서버에서는 지역명과 동에 대한 송신된 Frame에 지



(그림 2) 제어 서버의 처리

지역명	동	제어 명령어
-----	---	--------

(그림 3) Frame 구조

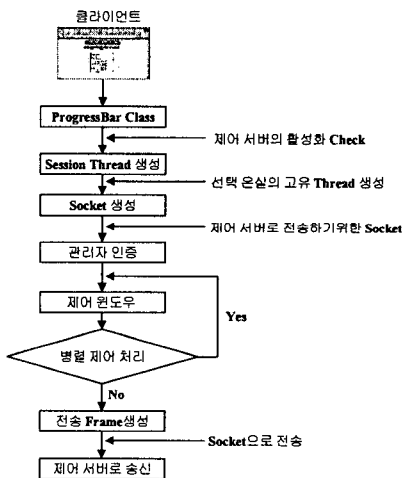
역명과 동을 필터한 후, 해당 스래드로 제어 명령어를 송신한다. (그림 3)의 Frame 구조에 대한 제어 명령어는 <표 1>과 같다.

<표 1> 제어 명령어 테이블

장치명	제어명령	동작	역동작	정지
1중의쪽천창	1WCA	0	C	S
1중으로쪽천창	1WCB	0	C	S
2중의쪽천창	2WCA	0	C	S
2중으로쪽천창	2WCB	0	C	S
1중으로쪽측창	3WCA	0	C	S
1중으로쪽측창	3WCB	0	C	S
2중의쪽측창	4WCA	0	C	S
1중으로쪽측창	4WCB	0	C	S
온도계1	1RAA	-	-	-
온도계2	1RAB	-	-	-
습도계1	1RAD	-	-	-
습도계2	1RAE	-	-	-
풍향계	1RAG	-	-	-
풍속계	1RAH	-	-	-
일사계	2RAA	-	-	-
CO2 발생기	2RAB	-	-	-
커튼(올림)	5WCA	-	-	-
환풍기	5WCB	-	-	-
관수장치	5WCC	-	-	-
커튼(닫힘)	5WCD	-	-	-

클라이언트 모듈은 java의 JFC를 사용하였다. 원격지에서 서버에 접속시 서버의 접속과 동시에 제어 서버의 활성화 상태를 ProgressBar Class에서 각 진행 정도에 따라 서버에 접근하고 활성화를 검사한다. 정상인 경우 즉 제어 서버가 활성화되어 있고 시스템이 ON 상태이면 서버는 다음 페이지를 로딩한다. 관리자 인증을 거친 후 온실 환경 자원 제어 윈도우가 생성되며, 관리자가 제어 대상을 병렬로 제어한다면 선택한 명령어를 Vector class에 저장한다. 각 명령어는 Enumeration Class를 사용하여 Vector class에 저장되어 있는 제어 명령어의 추출과 동시에 정의된 Frame의 구조로 생성하고 제어 서버로 송신한다. 만약 병렬적으로 제어를 할 경우 Main Thread에 지연 시간을 0.3초 생성시킨 후에 두 번째 명령어가 송신된다. 0.3초의 지연 시간은 연속된 명령어로 인한 하드웨어의 오작동을 사전에 방지하기 위한 수칙이다. 예를 들어, 3개 이상의 제어 명령어를 병렬 처리할 경우에 0.3초 이하의 지연시간에는 제어 명령이

소멸되어버리기 때문에 03초 이상의 지연시간이 주어진다. 클라이언트의 처리 구성도는 (그림 4)와 같고, 병렬 제어 코드는 (그림 5)와 같다. (그림 5)의 병렬 제어 코드에서는 메시지 다이얼로그 박스를 통해 OK Event가 발생되면 ParallerControl 클래스를 호출하고 인자로 명령어 저장하기 위한 Vector class의 인스턴스인 V_Resource와 제어명령어인 String형의 인스턴스인 RS232Comm을 가진다. Vector class인 V_Resource에 RS232Comm을 요소로 추가시킨다. 또한 Vector class에 저장되어 있는 제어 명령어들을 Enumeration Class를 통해 접근하고 그 제어 명령어들이 존재할 때까지 String형인 MultiCommand 변수에 제어명령어가 할당된다. 생성된 값은 출력 스트림인 PW를 사용하여 소켓을 통해 서버로 전송되어지며 전송된 제어 명령어 값이 정상적인 처리가 되었는지 알기 위해 서버의 응답을 입력 스트림 BR을 통해 확인한다. 즉 제어 서버에서의 응답은 ServerResult String형 변수에 할당된다.



(그림 4) 클라이언트 처리 구성도

```

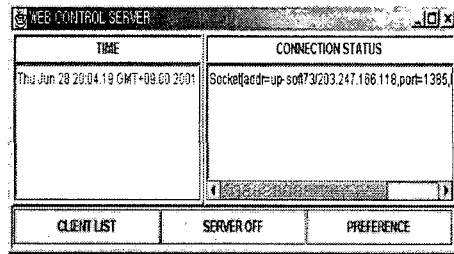
case JOptionPane.OK_OPTION:
    PC = new ParallerControl(V_Resource,
        RS232Comm);
    . . .
    V_Resource.addElement(RS232Comm);

try{ Enumeration enum =
    V_Resource.elements();
while(enum.hasMoreElements()) {
    MultiCommand =
    String.valueOf(enum.nextElement());
if(MultiCommand.equals
    ("J11 RIGHT WINDOWOPEN")) {
    MultiCommand = "JISSE";
    PW.println(MultiCommand);
try{((ServerResult = BR.readLine()) != null){
    TA_list.setText("시스템 공지: 다중 제어=>"
    + ServerResult);
    Thread.sleep(300); }
// 다음 명령 제어
else if(MultiCommand.equals
    ("J11 LEFT WINDOWOPEN"))
    { MultiCommand = "JISIE";
    PW.println(MultiCommand);
    . . .
    }
    }
    
```

(그림 5) 병렬 제어 코드

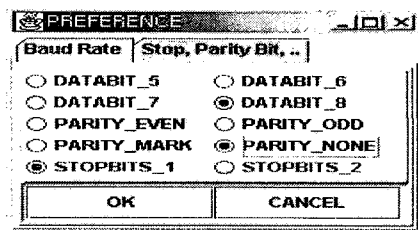
3. 구현 결과

본 논문의 실시간 원격 제어를 위한 모니터링은 원격지에서 대상 농가의 온실 환경 자원을 실시간 제어 할 수 있도록 하였고, 서버나 클라이언트의 부하를 줄이기 위해 자바의 경량 컴포넌트인 JFC를 사용하여 구현하였다. 또한 사용자의 편의를 위해 계층적인 메뉴를 구성하였고 자바의 확장 패키지인 Javax.comm을 이용하여 직렬 통신을 수행하였다. 시스템 상주 윈도우인 제어 서버의 메인 윈도우는 (그림 6)과 같다.



(그림 6) 제어 서버 메인 윈도우

(그림 6)은 인터넷을 통해 제어 서버로 접속된 시간과 접속 정보를 알 수 있다. 접속 시간은 Vector class로 저장되어 있으며 이 Vector 값이 HashTable의 Value값으로 저장된다. 또한 접속 정보도 Vector Class로 저장되어 있으며 HashTable의 Key값으로 입력된다. 접속 정보 Vector의 요소 중 접속을 해제했을 경우, HashTable의 Key값으로 Time을 검색하고 검색한 Time과 접속 정보는 삭제된다. 제어 서버 메뉴 중에 CLIENTLIST는 현재 접속되어 있는 Client를 나타내며, SERVEROFF는 제어 서버와 현재 연결되어 있는 클라이언트의 연결을 종료한다. PREFERENCE는 RS-232C와 통신을 하기 위한 직렬포트 통신 옵션이고, 윈도우는 (그림 7)과 같다.

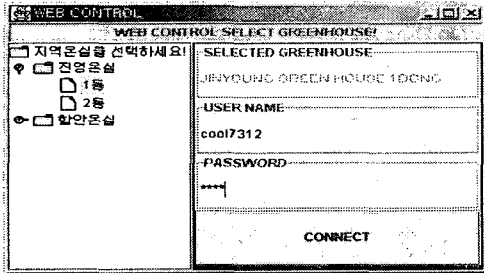


(그림 7) 제어 서버에서 PREFERENCE 윈도우

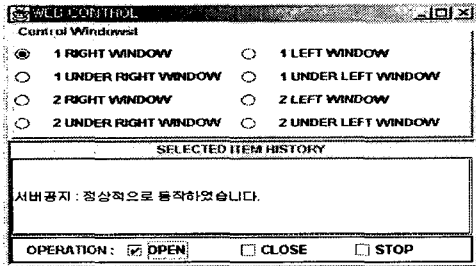
직렬 통신을 하기 위해서는 Serial Port 옵션을 설정해야 한다. 현재 시험중인 하드웨어의 Serial Port 옵션은 Baud Rate 2400, Data Bit 8, Stop Bit 1 그리고 Parity는 None이다. 또한 사용자의 서로 다른 하드웨어를 지원하기 위하여 (그림 7)과 같은 윈도우를 제공한다.

클라이언트 윈도우는 ProgressBar Class를 통해 제어 서버 활성화 여부를 검사한다. 또한 활성화 상태이면 제어 서버와 Socket 통신을 한다. 즉 클라이언트에서 생성된 프레임을 Socket을 통해 제어 서버와 데이터를 송·수신한다. 자바의 확장 패키지인 JFC로 UI가 구성되어 있으며 가상 소켓을 생성하여 제어 서버와 접속을 시작한다. 정상적인 소켓 생성이 되면 제어 서버가 활성화되어 있는 것으로 판단하고 제어 명령어를 송신하기 위한 준비를 하게 된다. 제어 서버의 활성화가 확인되면 제어를 하기 위한 윈도우가 생성

된다. JTree Class를 통하여 현재 등록되어 있는 온실을 Tree 형식으로 제공하고, 또한 해당 온실의 동을 선택하면 오른쪽에 관리자 인증 Frame이 활성화된다. (그림 8)은 제어 온실의 선택 후 해당 온실의 관리자 인증 윈도우이다. 관리자가 Connect 버튼을 선택했을 때 관리자 확인을 거친 후 선택한 동에 대한 제어를 위한 온실 환경 자원을 선택할 수 있는 윈도우가 (그림 9)와 같이 생성된다



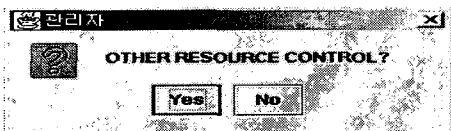
(그림 8) 제어 온실 선택 및 관리자 인증 윈도우



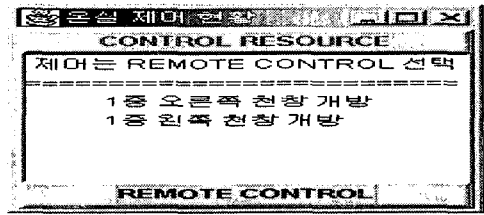
(그림 9) 환경 자원 선택 윈도우

각 자원들은 CheckBox로 구성되어 있으며 각 자원의 동작은 정지, 폐쇄, 개방 세가지로 제어할 수 있다. 각각의 CheckBox를 선택했을 때 SELECTED ITEM HISTORY Text Area에 표시하며, 제어 서버를 통해 정상적인 제어가 되었다면 제어 서버에서 Socket을 통해 정상 동작이 되었다는 메시지가 (그림 9)와 같이 클라이언트로 수신된다. 또한 사용자가 하나의 환경 자원이 아닌 병렬적으로 온실 환경 자원을 제어할 경우는 생성된 (그림 10)과 같은 팝업 윈도우를 사용한다.

Yes 버튼을 선택하면, Yes_Option Event가 발생하여 Vector class로 명령어가 저장되며 사용자가 No 버튼을 선택할 때까지 계속 Vector class에 저장이 된다. 그리고 No 버튼을 선택했을 때 No Option Event가 발생하여 (그림 11)과 같은 병렬 처리가 수행된 자원의 윈도우가 표시된다. 사용자가 REMOTE CONTROL 버튼을 선택했을 때 제어 명령을 내린 순서대로 Vector Class에 저장되어 있는 제어 명령어를 추출한다. 추출된 제어 명령어는 0.3초 간격으로 제어 서버로 송신한다.



(그림 10) 병렬 처리 팝업 윈도우



(그림 11) 병렬 처리 제어 윈도우

4. 결론

본 논문에서는 클라이언트/서버 형태로 Java 확장 패키지인 JFC, Communication API를 이용하여 지역 온실의 환경 자원을 원격 제어하는 모니터링 방법을 제안하고 구현하였다. 원격 제어 시스템은 클라이언트/서버의 형태로서 인터넷을 이용하여 장소의 한계를 극복할 수 있다. 또한 자바의 경량 컴포넌트를 사용하여 클라이언트와 서버의 부하를 최소화하였으며 사용자의 편의성을 도모하기 위하여 계층적인 메뉴를 사용하였으며 기능별로 서버를 구성하여 유지·보수를 쉽게 할 수 있다.

개발된 원격 제어 시스템을 이용하면 웹으로 원격지의 온실에 접근 가능하여 온실 관리자가 원격지에 있을 경우에도 온실의 관리 및 제어를 할 수 있는 장점이 있으며, 온실 환경 자원을 제어한 후 제어 서버의 정상 동작을 ServerSocket을 통해 송신되므로 사용자는 그 결과를 실시간으로 모니터링 할 수 있다. 또한 Applet이 로딩될 때 제어 서버의 활성화를 검사하여 제어 서버의 이상 유무를 검사할 수 있다. 그리고 텍스트 기반의 비효율적인 순차 제어 메뉴를 방식을 계층적 GUI로 보완하여 컴퓨터 운용 능력이 미숙한 사용자에게 보다 쉬운 접근을 제공할 수 있다.

참고 문헌

- [1] A.Krishnamurthy, T.D.C Little, "Connection-oriented service renegotiation for scalable video delivery," In Proceeding of the international Conference on the multimedia Computing and Systems, pp.502-507, May 1994.
- [2] Keith A. Butler, Robert Jacob, Bonnie E. Jone, Introduction and Overview of Human-Computer Interaction, CHIVAS, April 1994.
- [3] 권영삼, "국내 원예시설의 유형별 특성과 발전 방향", 국내외 원예 시설의 특성과 시스템에 관한 심포지엄, 한국시설원예 연구회, pp.111-125, 1995.
- [4] 박홍복, "시설 원예를 위한 비닐 온실 자동화 관리 시스템 개발", '99년도 부경대학교 산학연 컨소시엄, 1998. 10.
- [5] 김대업, 서상진, 박홍복, "웹 기반의 온실 관리 시스템의 설계 및 구현", 인터넷정보학회 제2회 추계학술발표대회 논문집 제1권 2호, pp.32-36, 2000.12.