
LabView를 이용한 실시간 온실 모니터링 및 원격 제어

서정희* · 박홍복**

Real-Time Remote Greenhouse Monitoring and Control Using LabView

Jung-Hee Seo* · Hung-Bog Park**

이 논문은 2002년도 1학기 부경대학교 연구년교수지원에 의하여 연구되었음

요 약

최근에는 웹(Web)을 기반으로 하여 여러 분야에 적용되는 원격 감시, 자동화 제어 시스템 등의 개발에 관한 연구가 활발히 전개되고 있다. 본 논문에서는 기존의 자동화 시스템에 원격 제어 기술을 추가하였다. 오류의 수정과 적용이 쉬운 LabView 그래픽컬 언어를 사용하여 멀티미디어 데이터 처리, 일반적인 통신망(TCP/IP)상의 확장성을 고려한 실시간 모니터링 및 원격 제어를 위한 방법을 제안하고, 온실 환경 제어 시스템에 적용하였다. 적용 결과는 온실의 제어 상황을 원격으로 제어 및 감시하는데 매우 효율적인 것으로 평가되었다.

ABSTRACT

Recently, a study on the development of remote monitoring and automation control system based on Web is spread widely. In this paper, we have expended the remote control method on the existing automation. And we proposed method to do real-time monitoring and remote control considering multimedia data processing, the expansion of TCP/IP using LabView graphical language. Also we applied the method in greenhouse environment control system. The application result was evaluated by very efficient thing to remotely monitor and control situation of greenhouse.

키워드

automation control, real-time monitoring, web control, multimedia data processing

1. 서 론

산업 현장에서 생산하는 제품의 질을 향상시키고 생산 효율을 극대화시키기 위하여 인력, 자재 및 장생산 시스템을 구성하고 제조 환경을 최적화시킬 수 있는 시스템의 중요성은 날로 증가하고 있다. 이런 추세

에 따라 개인 농가에서도 컴퓨터의 보급이 널리 확대되고 있으므로, 컴퓨터를 이용한 작물 재배의 데이터 관리 및 원격 제어 모니터링을 위한 효율적인 생산 시스템이 개발되고 있다[1,2,4].

오늘날 대부분의 데이터 모니터링 및 제어를 위한 시스템들은 C, VC++ 언어 등 여러 프로그램 언어로 시스

*부경대학교 대학원 박사과정

**부경대학교 전자컴퓨터정보통신공학부 교수

템을 구축하고 있다[3,6,7]. 그러나 기존의 언어를 이용한 프로그램들은 실질적인 오류가 발생되어 데이터에 영향을 주고 있음에도 쉽게 발견하기 어려울 뿐만 아니라 프로그램 작성자 외에는 이러한 프로그램 오류를 수정하기가 어렵고 장치의 원격 제어를 수행하기가 어렵다. 이러한 단점을 보완하기 위해 오류 수정과 데이터 획득의 신뢰성과 사용상의 편의성을 제공하는 그래픽컬 언어인 내셔널인스트루먼트사의 LabView(Laboratory Virtual Instrument Engineering Workbench)가 널리 사용되고 있는 추세이다[8].

기존의 연구[10]에서 구현된 시스템은 웹 상에서 지역 환경 자원들의 모니터링을 자바로 구현하여 이식성이 좋고, 최적화된 작물 데이터베이스와 모듈별 설계로 재사용성의 장점이 있으나 프로세스의 자원 낭비, 데이터 수집 상태의 확인에 불편함이 있고, 지역 제어만을 위한 한정된 단점을 가지고 있다.

그러므로 본 논문에서는 기존의 시스템에서 처리가 불가능한 웹 상에서의 실시간 원격 제어 방법을 구현하여 온실의 환경 제어 시스템에 적용하였다. 단일 클라이언트 모니터링의 시간적, 공간적인 한계를 극복하기 위해 그래픽컬 언어인 LabView를 사용하여 실시간 원격 제어와 실시간 모니터링 정보의 시각적 표현에 중점을 두었으며, 웹을 통한 제어를 수행하므로 불법적인 접근을 방지하기 위해 이중 메시지 축약 알고리즘[10]을 이용하여 관리자 인증을 수행함으로써 사용자의 신뢰성을 향상시켰다. 그리고 환경 자원 데이터 분석을 위한 일별, 월별, 주별로 환경 자원들을 분석할 수 있도록 기존의 통합 시스템에 적용하였고, 적용 결과는 온실의 제어 상황을 감시하는데 매우 효율적인 것으로 평가되었다.

II. 시스템 구성

2.1 시스템 구성도

본 연구의 시스템 개인용 컴퓨터, 멀티미디어 재생 장치와 시스템 제어를 위한 제어 인터페이스 장치로 구성되며, 소프트웨어는 시스템 구축을 위한 LabView6.1, JBuilder6.0, ASP, SQL Server, 그리고 Windows 2000, windows 98 환경하에서 운용될 수 있도록 구성하였다.

시스템의 구성은 그림 1과 같이 인터넷을 통한 모니터링과 환경 제어를 할 수 있는 클라이언트, 웹 서

비스와 이중 암호화 모듈을 제공하는 서버, 그리고 지역별의 환경 자원을 모니터링 및 제어 가능한 지역 제어 모듈로 구성된다. 클라이언트는 웹 서버로부터 전송된 자바 Applet과 ASP 웹 페이지를 이용하여 관리자 컴퓨터와 Socket 통신한다. 웹 서버는 Windows2000 운영 체제에 ASP를 사용하여 구축하였고, 자바 Applet, 기본 보안 패키지인 security 패키지 [6,9]을 사용하여 암호화 모듈을 구현하였다.

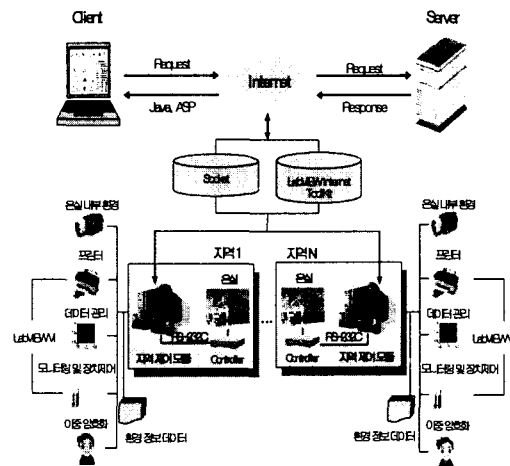


그림 1. 시스템 구성도
Fig. 1. A block diagram of system

지역 제어 모듈은 지역의 장치 제어와 지역간의 정보 교환이 가능하도록 LabView와 자바를 사용해 구축하였고, 세부 모듈로는 Active X 컨트롤을 사용하여 웹 브라우저를 통해 웹 카메라와 연동하여 지역의 환경 자원을 영상으로 모니터링 할 수 있는 온실 내부 환경 모듈, Report Generation vi를 이용하여 지역 프린터 초기화 및 환경 자원 데이터를 출력할 수 있는 프린터 모듈, 사용자의 데이터의 관리 및 분석을 위해 환경 정보 데이터를 파일 레지스트리에 저장하고 시간별, 일별, 주별, 월별 데이터를 XY 차트 vi를 이용하여 화면에 표시하는 데이터 관리 모듈, 사용자의 환경 설정에 따라 자동화 제어 엔진을 수행하는 모니터링 및 장치 제어 모듈, 또한, 암호화 모듈은 메시지 축약을 획득하기 위해 Java.security 패키지 MessageDigest 클래스의 getInstance 메소드를 통해 128bit 압축율을 가지는 SHA 알고리즘을 사용하였다.

그리고 온실의 각 장치를 제어하기 위한 컨트롤러는 기존의 개발된 제품[9]을 이용하였고, RS-232C 통신에 의해 각 장치들을 액정 패널 및 컴퓨터로 제어하게 되며, 시스템에 오류가 발생될

컨버터를 통한 센서의 전압 범위는 1V~5V이며, A/D 변환은 8비트 샘플링을 통해 수행되므로, 0에서 28-1(255)사이의 샘플링 값의 범위를 가진다. 따라서 아날로그 전압값을 디지털 값(t)으로 변환하는 식은 다음과 같다.

$$t = (((X1 - X2) / (256 - \beta)) \times (Y - \beta)) + (X1 + a),$$

- X1 : 센서의 측정 최저값
- X2 : 센서의 측정 최대값
- Y : 샘플링된 전압값
- a : 허용 오류값
- β : 최저 측정 전압값

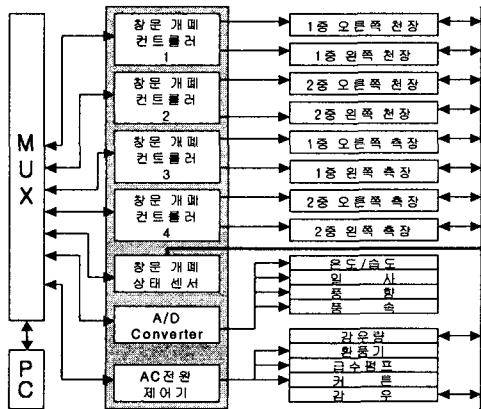


그림 2. 제안된 하드웨어 컨트롤러
Fig. 2. The proposed H/W controller

경우에는 상태 정보를 모니터에 표시하고 온실의 스피커 및 관리자의 핸드폰으로 메시지를 전송하게 된다.

2.2 Mux 기반의 하드웨어 컨트롤러

기존의 온실 컨트롤을 위해 사용된 on/off 기반의 회로는 어떤 문제가 발생했을 때 컨트롤 보드 전체를 교체해야 하는 결점을 가지고 있다. 이것은 크기와 컨트롤 보드의 모형이 온실의 특징에 따라 제작되었기 때문이다. 본 논문에서는 이 결점을 극복하기 위해 그림 2와 같은 하드웨어 컨트롤러를 개발하였다[10]. 이 컨트롤러는 창문 개폐 컨트롤러, 창문 개폐를 위한 센서, A/D 컨버터, AC 전원 공급기로 구성되어 있다.

Mux는 PC에 붙은 하드웨어 컨트롤러로부터 데이터를 전송한다. 그림 2의 PC는 그림 1의 지역 제어 모듈에 포함된 것이다. 또한, 오류 데이터를 방지하기 위한 프로토콜이 내장되어 있고, 하드웨어 장치 컨트롤러는 RS-232C 직렬 포트를 통해서 PC와 메시지 전송을 한다. 데이터 바이트는 직렬로 한번에 한 라인으로 보낸다. 또한, 컨트롤러는 I/O 인터페이스 모듈에서 제공되는 명령어에 따라 동작을 수행한다. A/D

III. 실시간 모니터링 및 원격 제어

3.1 모니터링 및 원격 제어

웹 기반의 실시간 모니터링 및 원격 제어는 클라이언트/서버 환경하에서 수행되며, 그림 3과 같이 웹 서버를 통해 Applet 로딩시 클라이언트에서 전송된 사용자명, 비밀번호, 난수, TimeStamp 입력으로 웹 서버의 암호화 모듈(Java.Security Package의 MessageDigest)을 수행하고, 소켓을 통해 지역 제어 모듈(관리자 컴퓨터)로 전송한다.

지역 또는 원격지에서 전송받은 데이터를 관리자 컴퓨터의 암호화 모듈과 비교하여 동일하면 LabView Internet Toolkit G-WebServer를 가동하고 Web Publish 서비스를 통하여 LabView Application을 모니터링 및 제어 가능한 HTML 웹 페이지로 변환한다 [8]. 변환된 HTML 페이지는 Http 프로토콜을 사용하여 Web Publish 서비스를 하며 클라이언트 웹 페이지에 로딩된다. 관리자 컴퓨터로부터 인증이 실패할 경우 Connection Error 메시지를 Socket을 통해 클라이언트로 전송하게 되며 환경 자원을 모니터링 할 수 있는 웹 페이지가 생성된다. 인증 성공시 사용자는 각 환경 자원들을 원격 제어 및 모니터링 가능한 페이지가 생성되고 사용자의 이벤트, 즉 제어 버튼을 이용하거나 메뉴의 선택시 해당 명령어를 G-WebServer의 Web Service를 통해 관리자 컴퓨터로 전송하고, 관리자 컴퓨터는 전송받은 명령어를 장치 제어 컨트롤러와 RS-232C 통신으로 제어한다. 제어 대상으로는 온

실의 환경 제어 자원인 천창

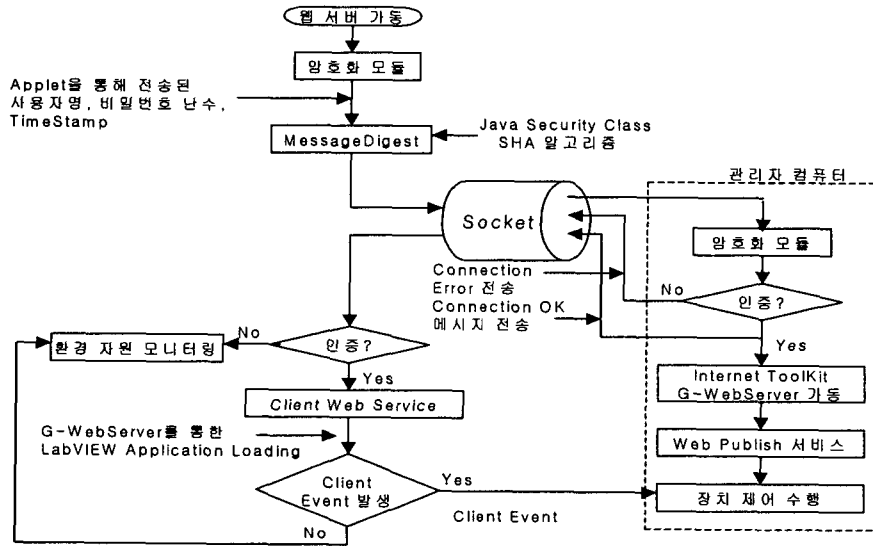


그림 3. 모니터링 및 원격 제어의 흐름도
Fig. 3. The flowchart of the monitoring and remote control

좌1, 천창 우1, 천창 좌2, 천창 우2, 측창 좌1, 측창 우1, 측창 좌2, 측창 우2, 커튼, 환풍기, 그리고 관수장치 등이며, 이들은 온·습도, 풍향, 풍속, 일사량, CO₂ 발생량을 센서로부터 입력받은 데이터에 의해 제어된다.

3.2. 지역 제어 모듈

각 지역의 온실 환경을 제어하기 위한 지역 제어 모듈은 자동·수동으로 수행할 수 있으며, 자동모드로 수행할 경우 자동화 엔진에 의해 설정된 환경에 따라 자동적으로 수행되며, 수동 모드인 경우는 온실의 각 장치를 버튼으로 직접 제어하는 액정 패널을 이용하여 관리자가 제어한다.

하드웨어의 효율적 운용과 사용자의 신뢰성을 향상시키기 위해 지역제어 모듈은 온실의 각 단위별 계층적 분류를 통한 클래스로 구성하였다. 각 클래스는 실제 환경에 동적으로 적용시키기 위해 시스템 레지스트리(Program.ini, House.ini)를 이용하여 실행시 바운딩된다. House 레지스트리는 온실 자원에 대한 설정 상태와 관리 대상 항목을 나타낸다. 자원의형태에 따라 설정 항목이 달라지며, 자원 추가시House 레지스트리의 신규 자원에 대한 설정 항목을

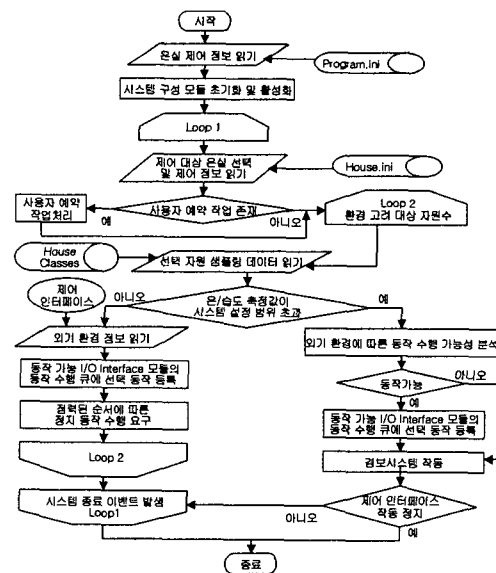


그림 4. 지역 제어 모듈의 흐름도
Fig. 4. The flowchart of local control module

제어 자원의 재원과 동일하게 설정해야 한다. 관리하고자 하는 신규 자원의 재원과 레지스트리 내에 자원 설정이 부정확하면 온실 환경 제어 시스템의 I/O Interface Module에서 입/출력 제어가 정확하지 않으므로 시스템이 불안정할 수도 있다. 지역 제어 모듈의 흐름도는 그림 4와 같다. 여기에는 두 개의 루프가 있는데, 첫 번째 루프는 같은 위치에 있는 온실의 장치 수와 관련이 있고, 각 온실의 장치는 구성이 다른 타입을 가지고 있다. 두 번째 루프는 특정 온실 장치의 지역과 관련된 것이다. 그러므로 순서도의 전체 루핑 수는 $m \times n$ 이고, m 은 온실 장치의 수, n 은 각 장치에 있는 하드웨어의 수를 나타낸다. 온실 환경 제어를 위해 적용된 자동화 엔진은 동작 대상 장치를 표현하고 제어 동작 수행을 위한 동작 명령어, 장치 상태값의 정보를 포함하고 있다. 또한, 자동화 엔진의 환경 제어 동작은 명령어의 리스트로 미리 정의되어 있으며, 각 자원의 우선 순위에 따라 처리되며, 그 동작 리스트와 우선 순위는 표 1과 같다.

표 1. 엔진 제어 동작 리스트와 우선 순위
Table 1. The engine control action list and priority

동작명	동작 순서	동작 정의
가운 순서	1. 커튼 폐쇄 2. Fan 정지 3. 측창 폐쇄 4. 천창 폐쇄 5. 보일러 작동	온도 변화에 비교적 적은 영향을 받는 자원 순서로 제어
감운 순서	1. 보일러 작동정지 2. 천창 개방 3. 측창 개방 4. Fan 작동 5. 커튼 개방	급작스런 온도 변화를 우려한 감시 제어 필요
가습 순서	1. 가습기(Mist) 작동 2. 커튼 폐쇄 3. Fan 정지 4. 측창 폐쇄 5. 천창 폐쇄	습도 변화에 비교적 적은 영향을 받는 자원 순서로 제어
감습 순서	1. 가습기 작동 정지 2. 천창 개방 3. 측창 개방 4. Fan 작동 5. 커튼 개방	급작스런 습도 변화를 우려한 감시 제어 필요

3.3. 경보 메시지 전송 방법

각 대상 온실의 관리자 부재시 대상 온실을 감시하는 클라이언트는 시스템 오류나 하드웨어적인 오류에 따른 오류 이벤트를 발생한다. 인터넷 기반의 온실 환경 제어 정보 시스템은 관리자에게 메시지를 통해 오류 발생 상황을 전달한다[9,10].

경보 메시지 전달을 위해 시스템은 각 통신사의 SMS 게이트웨이와 연동하여 처리한다. Java의 Native method인 HttpURLConnection을 사용하여 SMS 게이트웨이에 접근하여 해당 CGI 서버로 접속한다.

경보 메시지는 오류 발생시 컴퓨터의 화면, 스피커, 핸드폰으로 오류 발생 장치를 음성 및 Text로 동시에 전송되며, 핸드폰으로 오류 메시지의 전송 방법은 그림 5와 같다.

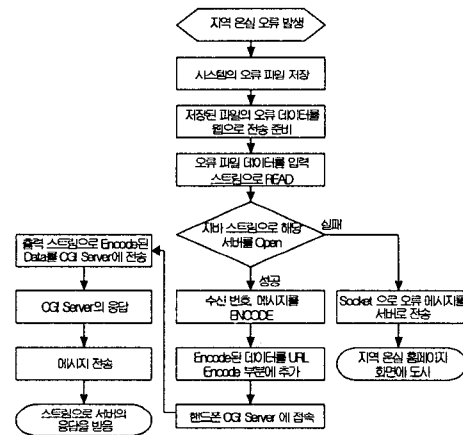


그림 5. 경보 메시지 흐름도
Fig. 5. The flowchart of alarm message

IV. 구현 및 분석

4.1 구현

본 논문의 구현을 위하여 출력은 컨트롤 대상인 창문 8개(천창 4개, 측창 4개), 관수 모터, 그리고 입력으로는 내부의 온·습도 센서 2개, 외부 온·습도 1개, 일사량, 감우, 풍향, 풍속 센서를 이용하였다. 또한 네트워크의 불법적인 접근을 방지하기 위해 이중 메시지 축약으로 관리자 인증을 구현하였고, 온실 내부 영

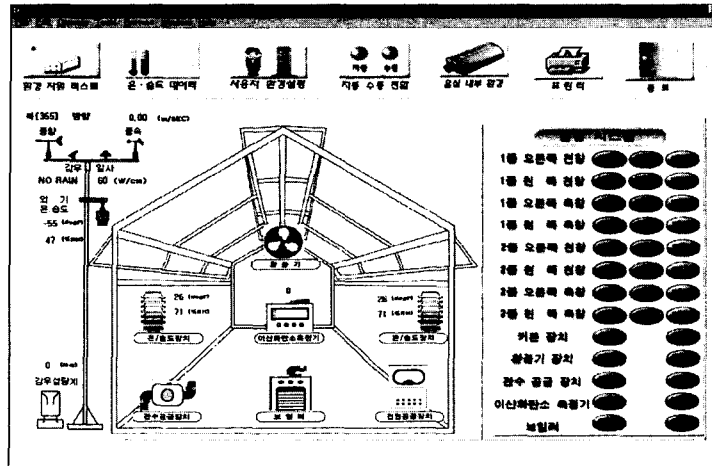


그림 6. 메인 윈도우
Fig. 6. Main window

상을 제공함으로써 하드웨어적인 결함에 대한 사용자의 신뢰성을 향상시켰다. 또한, 환경 정보 데이터를 도표로 제공함으로써 데이터 분석이 용이하게 하였다.

• 메인 윈도우

본 논문에서 제공되는 관리자 컴퓨터의 메인 윈도우는 그림 6과 같다. 이 메인 윈도우는 사용자 인터페이스라는 LabView의 프론트 패널에 해당하는 부분으로서 그래픽, 버튼, 입력, 출력에 해당하는 아이콘 요소가 위치하게 된다. 즉, 주메뉴, 이미지 리스트, 장치의 동작 수행을 위한 버튼의 세 부분으로 구성되어 있다.

주메뉴는 현재의 환경 자원을 표시하는 환경 자원 리스트, 자동화 엔진 가동시 사용자가 환경을 설정할 수 있는 사용자 환경 설정, 데이터의 수집 및 검색을 위한 온·습도 데이터, 환경 정보 데이터의 분석을 위한 데이터 관리, 자동화 엔진의 가동 여부를 설정하는 자동·수동, 온실의 내부 환경을 영상으로 제공하는 온실 내부 환경, 환경 데이터를 출력할 수 있는 프린터, 전체 시스템의 종료를 위한 종료 메뉴의 7개의 메뉴로 구성되어 있다.

그리고 화면 하단 왼쪽의 이미지 리스트는 PictureRing.vi를 사용하여 로딩 속도에 따른 시스템 오버헤드를 고려하였으며 각 장치의 동작 수행 여부를 표시한다. 또한, 오른쪽의 버튼들은 각 장치의 동작, 정지, 폐쇄 기능을 담당하며 예기치 않은 시스템의 오류가 발생하였을 때 전체의 시스템

을 통제하는 총괄 시스템으로 나누어져 있다.

그리고 시스템의 동작을 실시간으로 모니터링하고 제어를 수행하는 LabView G 언어의 제어 코드를 그림 7에서 나타내고 있다. 여기서서는 프론트 패널에 대응하는 창으로서 코드를 작성하는 창이다. 즉, 프론트 패널에서 값을 입력하는 컨트롤(control)이나 출력값을 나타내는 인디케이터(indicator)를 위치시키면 이에 대응되는 터미널이 표시되는데, 이 터미널과 해당 함수를 와이어(wire)로 연결하여 코드를 작성하게 된다.

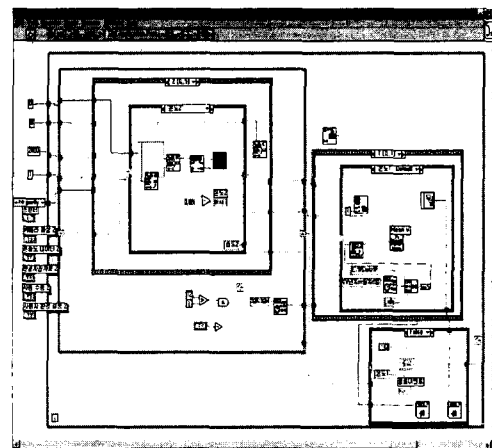


그림 7. 제어 소프트웨어의 LabView "G" 언어 코드
Fig. 7. The control software of Labview "G" language code

• 암호화 모듈

네트워크의 불법적인 접근을 방지하기 위해 자바의 확장 팩키지(JFC)로 구현된 관리자 컴퓨터의 암호화 모듈을 수행한 윈도우는 그림 8과 같다.

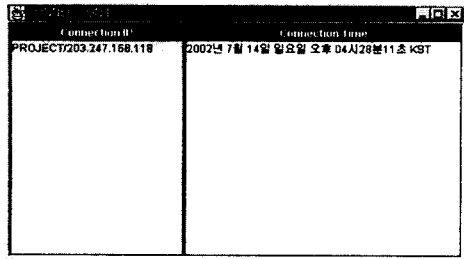


그림 8. 관리자 컴퓨터의 암호화 모듈윈도우
fig. 8. The cryptography module window of manager computer

관리자 컴퓨터의 암호화 모듈은 Applet을 통해 전송된 사용자 정보를 이중 메시지 축약(SHA 알고리즘)하고 클라이언트에서 축약한 메시지와 비교하여 동일하면 관리자 인증이 된다. 관리자 인증이 된 후 관리자 컴퓨터에서는 접속 시간, 접속 정보가 그림 8과 같이 표시되므로 현재 접속된 사용자를 알 수 있다.

•경보 메시지 전송

LabView에서 처리하기 어려운 휴대폰을 이용한 경보 메시지 전송은 무선 문자 메시지(Short Message Service) 서비스를 이용하여 블랜드 C++로 LabView 상에 연동하는 방법으로 구현하였다. SMS 서비스는 디지털 이동 전화의 액정 화면(LCD)에 한글 40자 내외의 간단한 문자 메시지(영문 80자)를 인터넷이나 휴대 전화로 전송할 수 있는 서비스이다. 경보 메시지 전송은 신세기 통신의 SMS 게이트웨이와 연동하여 경보 메시지를 처리한다. 그림 9는 핸드폰 전송 후 수신된 응답 화면이다. 즉, 정상적인 송신이 되었을 때 SMS 게이트웨이의 송신 응답 화면이다. 그리고 송신 응답을 받은 후 실제로 사용자에게 전송되는 핸드폰 오류 화면은 그림 10과 같다. 내용은 오류 메시지 내용과 발신지가 수신된다.

즉, 신세기 통신의 CGI 서버에 접속한 후, 출력 스트림을 생성하여(OutputStreamWriter) 해당 정보를 SMS 게이트웨이에 Encode 형식으로 송신한다. 정상적인 송신이 되었을 때 입력 스트림을 생성하여

(InputStreamReader) SMS 게이트웨이의 응답 메시지를 수신하게 된다.



그림 9. 수신된 응답 화면
Fig. 9. The received response image

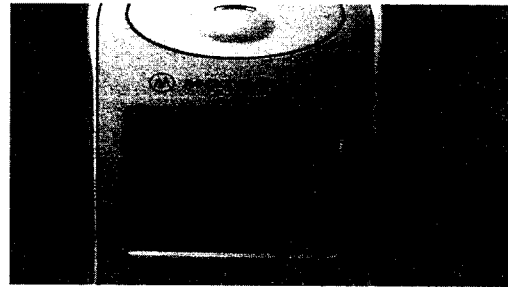


그림 10. 전송된 오류 화면
Fig. 10. The sent error image

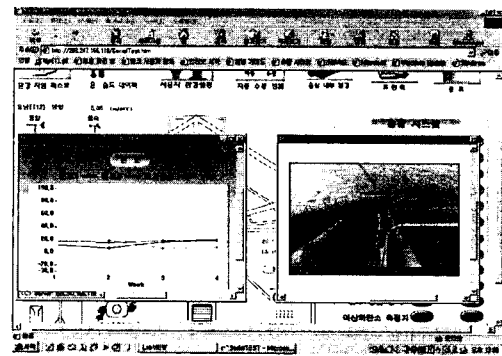


그림 11. 웹을 통한 데이터 분석 및 영상 모니터링

Fig. 11. Data analysis and image monitoring based on Web

•원격 제어 및 모니터링

Applet을 통한 클라이언트 사용자 인증 후 제어 서버는 LabView Internet ToolKit G-WebServer를 통해 LabView Application을 Html로 변환하고 클라이

언트 사용자에게 웹을 통한 모니터링 및 원격 제어 상태를 제공하기 위해 웹 페이지를 생성한다. 웹을 통한 모니터링 및 원격 제어 윈도우는 그림 4의 메인 윈도우와 동일하며, 지역의 관리자 컴퓨터에 저장된 온·습도 데이터의 주별 차트와 웹 카메라를 이용한 온실의 상황 정보를 원격으로 모니터링 및 제어할 수 있다. 지역의 관리자 컴퓨터의 데이터, 온실 내부의 환경에 대한 상황 정보 등을 모니터링하고 제어한 결과는 그림 11과 같다.

4.2 분석

제안된 시스템은 G 언어인 LabView를 사용하여 실시간 원격 제어 및 모니터링을 구현하였다. LabView를 이용한 실시간 원격 제어 및 모니터링 시스템은 클라이언트/서버의 중앙 집중식 모니터링 방식을 적용하여 지역 제한적 제어 시스템의 한계를 극복할 수 있다.

따라서, 본 논문에서 구현된 LabView를 이용한 실시간 모니터링 및 원격 제어 시스템은 온도, 습도, 일사량 등의 입력 데이터로 인해 설정된 환경 조건에 따라 지역 및 원격으로 제어되며, 설정한 현재 온실의 상황을 네트워크를 통해 효과적으로 모니터링 가능하여 사용자의 온실 관리 및 정보 획득에 대한 편의성을 제공할 수 있었다. 그리고 텍스트 기반의 비효율적 순차 제공 메뉴 방식을 계층적 GUI로 보완하여 컴퓨터 운용 능력이 미숙한 사용자에게 보다 쉬운 접근을 제공할 뿐만 아니고, 시스템의 암호화를 위해 자바의 확장 팩키지(JFC)로 구현된 관리자 컴퓨터의 암호화 모듈을 수행하여 기존의 시스템에서 제공되지 않은 네트워크의 불법적 침입을 방지하였으며, 웹 카메라를 통해 온실 내부 환경을 영상으로 제공함으로써 하드웨어의 오작동에 대한 모니터링을 할 수 있기 때문에 사용자의 편의성과 신뢰성을 향상시켰다.

그리고 환경 데이터를 차트와 디지털 값으로 표시함으로써 사용자의 데이터 분석을 용이하게 하였다. 그러나 제안된 시스템의 단점으로는 데이터 통신 방식이 직렬 통신이기 때문에 한 장치에 명령을 내린 후, 또 다른 장치에 명령을 내렸을 때 수행 동작의 속도가 느리다. 즉, 한 버튼의 동작 명령을 수행한 후 실제 동작이 되는 경우는 약 3초 정도의 수행 대기 시간이 걸린다. 여러 장치의 동시 처리 동작을 위해서도 일정 시간의 간격이 생기므로 시스템의 보완이 필

요하다. 그리고 기존 시스템과 제안된 시스템과의 비교를 표 2에 나타내었다.

표 2. 기존 시스템과의 비교
Table. 2. Comparison for existing system

구분	상용 시스템	기존 시스템[9]	제안 시스템
동작 수행 방식	On/Off식 제어	액정, 컴퓨터에 의한 제어	액정, 컴퓨터에 의한 제어
단동·연동 온실 제어	지역 제어	지역 제어	지역, 원격 제어
환경 데이터의 보관 및 분석	가능	가능	가능
경보 메시지 전송	불가능	가능	가능
실시간 상황 감시	불가능	불가능	가능
웹상의 데이터 획득 및 모니터링	불가능	불가능	가능
지역 및 온실별 상호 통신	불가능	가능	가능
불법적 침입 방지	불가능	불가능	가능
주문·판매 기능	불가능	가능	가능
가격	저가	중고가	중고가

V. 결론

기존의 온실 환경 제어 시스템들은 on/off 제어만을 이용한 지역 제어이거나, 사용자에게 편의성을 제공하는 모니터링, 오류 메시지 전송 등의 기능은 수행되지 않으며, 소프트웨어의 유지·보수가 어렵다. 따라서 본 논문에서는 사용자에게 편의성을 제공하고, 웹 기반의 원격 제어 기술을 쉽게 구현하고 유지·보수가 간편한 프로그램의 작성을 위해 NI사의 LabView 그래픽 언어를 이용하여 구현하였다.

현재 국내의 여러 농가들은 과학적 영농을 필요로 하고 있지만 컴퓨터의 운용 능력이 미숙하다. 그러므로 본 논문의 실시간 원격 제어 및 모니터링 시스템은 LabView를 이용하여 웹의 개념을 도입하였고, 오류의 수정과 적용이 쉽고 데이터 획득이 편리하므로 농·수산업 등의 산업 분야에 응용이 용이하다. 또한, 기존 시스템에 국한되어 있는 시간적, 공간적 기능의

한계를 탈피하고, 웹 서버로서의 기능과 인터넷의 목적을 최대화하는데 성과를 거둘 수 있다. 그러나 농촌 지역의 초고속 인터넷 지원 시설의 미비로 전 농가에 서 정보화 및 제안된 원격 제어 기술을 활용하는데는 정부의 강력한 의지가 있어야 된다고 본다.

참고문헌

[1] Theodore R. Haining, Darell D.E. Long, Patric E. Mantey, CraigM. Wittenbrink, "The Real-Time Environmental Information Network and Analysis System", Proceeding of COMPCON, March 1995.

[2] Keith A. Butler, "Introduction and Overview of Human-Computer Interaction", CHIVAS, April 1994.

[3] Woltring H. J, "A Fortran package for generalized, cross-validatorspline smoothing and differentiation", Advancss in Engineering Software, 1986.

[4] C. Serodio, J. BoaventuraChunha, C. Couto, "Bring the World Wide Web to Agricultural Systems based on the Java Environment", Proc. Int. Conf. and British Israeli Workshop on Greenhouse Tech. Towards 3rd Mill, pp.293-300, 2000.

[5] P. R. Fsher and R. D. Heins, "The Greenhouse Care System: A Decision-Support System for Height Control and Scheduling of Potted Flowering Plants", Proc. Int. Conf. and British Israeli Workshop on Greenhouse Crop Models, Ed.R.U. Larsen, pp.41-45, 1996.

[6] Merlin Hughes, Michael Shoffner, Derek hmner, mesh Bellur, "java NetworkProgramming", Manning, 2000. 5.

[7] Lan Burns, Andy Wellings, "Real-Time Systems and Programming Language", ADDISON WESLEY, pp.95-125.

[8] Robert H.Bishop, "LabView student Edition 6i, Prentice Hall, 2001.

[9] Jonathan Knudsen, "Java Cryptography", O'REILLY, 2000.

[10] 김대업, 박홍복, "인터넷에 기반한 온실 환경 제어 시스템에 관한 연구", 정보처리학회 논문지 제5 권 3호, pp.506-516, 2001. 6

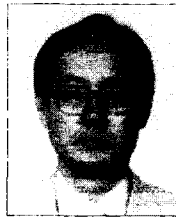
저자 소개



서정희(Jung-Hee Seo)

1994년 신라대학교 전자계산학과 (전산학)
1997년 경성대학교 전산통계학과 (이학석사)

※관심분야 : 원격교육, 멀티미디어, 영상처리, 정보보안



박홍복(Hung-Bog Park)

1982년 경북대학교 컴퓨터공학과 (공학사)
1984년 경북대학교 대학원 컴퓨터 공학과(공학석사)

1995년 인하대학교 대학원 전자계산학전공 (이학박사)
1984년~1995년 동명대학 전자계산학 부교수
2001. 2~2002. 2 The University of Arizona 객원교수
1996년~현재 부경대학교 전자컴퓨터정보통신 공학부 교수
※관심분야 : 실시간 시스템, 멀티미디어 응용, 산업 자동화, 원격교육, 프로그래밍 언어 및 컴파일러