

양액재배를 위한 생육환경 모니터링 및 제어 플랫폼에 관한 연구

연인원*, 이원철^o

Study Growth Environmental Monitoring and Controlling Platform for Hydroponic

In-won Yeon*, Won-cheol Lee^o

요약

전 세계적으로 농업분야가 많이 줄어들고 있는 반면에, IT농업과 미래 먹거리에 대한 관심이 높아지고 있으며, ICT 기반 미래 농업 활성화를 위해 다양한 연구가 진행 중에 있다. 본 논문에서는 양액재배를 위해 온도, pH(hydrogen ion), EC(Electric Conductivity) 등 생육환경을 모니터링 하는 동시에 최적의 환경을 유지하도록 LED, 쿨링팬 제어가 가능한 시스템을 제안한다. 생육환경 모니터링 및 제어를 위해 아두이노(Arduino) 하드웨어와 자바(Java) 소프트웨어를 이용하였으며, Wi-Fi 공유기를 활용해 소켓통신을 하고 데이터베이스 및 웹서버 인터페이스를 통해 여러 환경에서 식물을 쉽게 관리할 수 있다. 더욱이, 장소 및 시간에 구애 받지 않고 LED를 활용한 재배로 병해충이 생기지 않는 환경에서 지속적인 관리가 가능하여 단순 양액재배보다 빠르게 재배할 수 있는 장점이 있다. 더 나아가, ICT 미래 농업 활성화에 기여할 수 있을 것으로 기대한다.

Key Words : Hydroponic, IoT, WiFi, Arduino, Socket Communication

ABSTRACT

According to global trend, despite the overall scale of agricultural industry has been downsized, agriculture accommodating cutting-edge ICT technologies has been proliferated, and various timely-issued relevant researches have been on progress to deploy the future food cultivation. In this paper, we propose an effective nutrient management system with web-based monitoring with functionality of controlling temperature, humidity, pH (hydrogen ion), EC (Electric Conductivity), LED and cooling fan to maintain the hydroponic nurturing environment being optimal. In this paper, in order the arduino hardware and java software are employed to control the nurturing environment automatically in optimal fashion. In proposed system, due to the usage of WiFi router with the socket communication and DB-assisted Web server with proper interfaces, it allows to facilitate the management to keep monitoring and controlling overall hydroponic nurturing environment. Since the proposed Web-based management system provides the superior reliability, the short nurturing period and the robustness to the pest by controlling LED emitting color rather than conventional system, so it can be applied and appropriate for in-house vegetable factory overcoming limitation of time and location.

* 본 연구는 2015년도 숭실대학교 교내연구비 지원과 농림축산식품부 농생명산업기술개발사업에 의해 이루어진 것임

• First Author : Soongsil University Department of Electronic Engineering, dntls12@ssu.ac.kr, 학생회원

o Corresponding Author : Soongsil University Department of Electronic and Information Engineering, wlee@ssu.ac.kr, 중신회원
논문번호 : KICS2016-05-105, Received May 31, 2016; Revised August 16, 2016; Accepted August 16, 2016

I. 서 론

식물공장의 재배방식은 1957년 덴마크 크리스텐센 농장에서 태양광을 이용한 크레스(cress; 새싹채소) 재배를 시작으로 하여 1960년대는 입체식 식물공장으로 발전하였고, 1970년대에는 완전 제어형 생산시스템 식물공장을 구축하였다. 그 이후 1980년대부터는 식물공장의 상업화가 시작되었다.^[1] 국외 동향을 살펴보면 네덜란드의 식물공장에서 재배된 식용 야채의 연간 생산량은 150만 톤으로 13.3억 유로(약 2조 원)로 추정되고, 독일은 정부와 함께하는 그린농업인 iGreen 프로젝트를 진행하고 있다. 일본의 경우 ICT 융합 기술로 기계화, 편리성 도모, 수익향상, 건강증대, 안정성 확보 등 여러 측면에서 광범위하게 적용되고 있다. 이에 비해, 우리나라는 좁은 경지 면적과 농기구, 관리 시스템 등 높은 생산 비용으로 인하여 선진농업국에 비해 경쟁력이 매우 취약한 상황이기 때문에 농업 생산성이 둔화되면서 농가의 삶은 매우 어려워지고 있으며, 특히 재배면적 및 노동력 감소, 노동인구의 고령화, 기후변화 및 환경오염 등 많은 제약 조건이 빠르게 증가하고 있다.^[2,9] 이와 같은 문제로 현대 스마트 농업은 세계적으로 작물의 생산성 향상과 국가 정책 및 기술개발을 중심으로 추진되고 있으며, 국내에서는 미래창조과학부에서 ICT 기술을 농업 분야에 접목하여 스마트 농업으로 발전할 수 있도록 농업과 과학-ICT를 융합하고 새로운 아이디어 및 고도화를 위한 창조 비타민 프로젝트인 비타민A(Agriculture) 정책을 제시하였다.^[3] 이에 현대 농업은 IoT(Internet Of Things) 기기 및 인터넷 망 기반의 여러 가지 센서를 활용하여 생산과 생활이 이루어질 수 있는 스마트 자동화 시대로 진화하여 ICT융합 기술을 활용한 다목적 생산 및 투명한 유통 관리 확대, 개인 단위의 농식품 실 구매 데이터 등 개방을 통해 지능형 재배 기술의 빅데이터 분석, 생산 모니터링 및 제어 시스템과 같은 사회적 가치 창출이 요구되고 있는 시점이다.^[4,5] ICT 기술을 농업분야에 접목한 하나의 기술로서 본 논문에서는 식물을 재배하는 농장과 수경재배 기기와 같은 단순히 pH, EC 등 제어만 하는 단점과 생육환경 관리가 어려운 점을 보완하기 위해 양액재배를 위한 온습도, pH, EC 등 모니터링이 가능하고 식물 광합성용 LED, 온도 조절용 쿨링팬^[6] 제어가 가능한 생육환경 관리 시스템을 제안한다. 본 논문의 구성은 다음과 같다. 2장에서는 소프트웨어를 이용하여 구축한 모니터링과 제어 플랫폼에 대해 소개하고, 3장에서는 가정용 양액재배기에서 재배한 식물의 씨앗밭아에서부터

결과물 사례로 과정을 설명한 후, 마지막으로 4장에서 결론을 통해 마무리한다.

II. 생육환경 모니터링 및 제어 플랫폼

양액재배 생육환경 모니터링 및 제어를 위한 전체 시나리오 및 소프트웨어 시나리오를 소개하고 아두이노와 Wi-Fi 및 센서 종류들과 측정 범위를 확인한다. 또한 아두이노와 자바 간 소켓통신이 이루어지는 방법과 데이터베이스의 데이터를 JSP(Java Server Pages)에서 활용하는 방법을 설명한다.

2.1 양액재배를 위한 전체 시나리오 및 하드웨어 구성요소

본 절에서는 전체 시나리오와 기본적인 하드웨어 구성요소에 대해 설명한다. 그림 1은 본 논문의 식물 생육환경 모니터링 및 제어에 대한 전체 시나리오를 표현한 처리과정도를 나타내고 있다.

첫 번째 단계에서는 생육환경에 필요한 다양한 데이터를 측정하는 것으로 시작한다. Wi-Fi 공유기를 통해 통신 연결여부를 결정하고 통신이 성공하면 다양한 정보를 축적하는 데이터베이스로 전달하여 측정된 데이터를 업데이트하고 측정된 데이터를 통해 24시간 내내 문자열 테이블 또는 그래프 형태로 모니터링 가능하도록 데이터들을 Monitoring Page 및 Graph Page로 전달한다. 또한 Control Page의 명령 요청을 통해 데이터베이스에 On/Off 상태값을 변경하는 동시에 Control unit에 결합된 하드웨어 제어가 가능하다. 본 연구에서는 아두이노와 Wi-Fi Shield 및 온습도 센서, pH 센서, EC 센서, LED, 쿨링팬을 이용하였다. 표 1은 본 연구에서 활용한 생육환경을 측정하고 제어하기 위한 하드웨어를 나타내고 있다.

아두이노는 전자기술 경험이 없는 초급자들도 쉽게 사용할 수 있도록 설계된 하드웨어로 사물에 대해 전

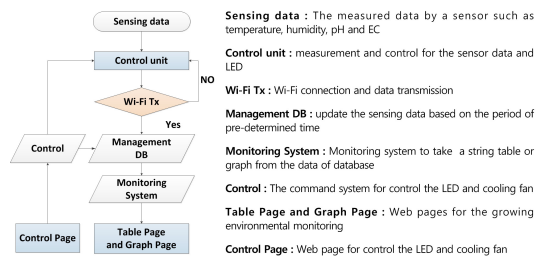





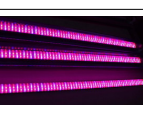



그림 1. 양액재배 환경 모니터링을 위한 전체 처리과정도
 Fig. 1. Flow chart for Hydroponic environmental monitoring

표 1. 하드웨어 규격
Table 1. Hardware Specification

Hardware	Specification	Product /Manufacturer
 Arduino	Micro Controller ATmega328P	Arduino Uno R3/Arduino
 Wi-Fi Shield	802.11b/g Specifications and SPI Communication	Arduino Wireless Proto Shield /Arduino
 Temperature and Humidity Sensor	Temperature measurement : -40℃ to 123.8℃ Temperature error factor : ±0.5℃ Humidity measurement : 0% to 100% Humidity error factor : ±3.5%	SHT-71 /Sensirion
 pH Sensor	pH measurement : 0pH to 14pH pH error factor : ±0.1pH	Analog pH Meter Pro /DFROBOT
 EC Sensor	EC measurement : 1mS/cm to 20mS/cm EC error factor : ±10%	Analog Electrical Conductivity Meter /DFROBOT
 LED	LED Light size : 550mm Color scale : Red, Blue, White = 2:1:1	LED/Vegshop
 Cooling Fan	Rated Speed : 3,500 R.P.M	DA8025H12/ D&J WITH

기적 연결로 사용자의 요청에 응답 혹은 제어할 수 있는 음향시설, 로봇, 홈 오토메이션 등과 같이 IoT에 적합한 기기로 각광받고 있으며, Atmel사의 ATmega328P, 디지털, 아날로그, PWM 등 마이크로 컨트롤러 및 다양한 핀으로 이루어져 있다^[11]. Wi-Fi Shield는 아두이노를 무선통신 할 수 있도록 Wi-Fi 공

유기에 접근할 수 있는 터미널이다. SHT-71 온습도 센서의 측정범위 및 오차율은 각각 -40℃에서 123.8℃, 오차율 ±0.5℃, 0%에서 100%, 오차율 ±3.5%로 높은 정확도로 측정 가능하며 pH센서의 측정범위는 0pH에서 14pH, 25℃환경에서 오차율이 ±0.1pH이며, EC센서는 1mS/cm에서 20mS/cm, 오차율 ±10%이다. EC센서는 오차율이 조금 높은 편이지만 물의 농도에 따라 저항을 조정하면 오차율을 대폭 줄일 수 있다. 이와 같은 하드웨어들을 이용해 C언어 기반의 코딩과 외부 전력을 이용해 최소 5V에서 최대 12V까지 출력 가능한 어댑터로 구동시켜 센서 데이터를 발생시킬 수 있으며, 다양한 플랫폼 제어가 가능하다.

2.2 소프트웨어 자바 시나리오

본 절에서는 생육환경 모니터링 및 제어를 위한 소프트웨어 자바의 시나리오와 그에 따른 세부사항에 대해 설명한다. 그림 2는 자바서버의 처리과정도이다. 처리과정도에 대한 설명은 다음과 같다. 아두이노와 자바 간 소켓통신이 이루어지면 센서로 측정된 데이터는 데이터베이스로 송신해 DAO(Data Access Object)로 전달하고 다시 DTO(Data Transfer Object)로 데이터를 리턴하여 웹서버로 보낸 후 웹서버를 기준으로 JSON(JavaScript Object Notation) 객체형식을 통해 모니터링 가능한 JSP로 전달한다. 또한 명령이 필요할 때 웹서버에서 DAO를 호출해 다시 DTO로 명령 데이터를 리턴하여 웹서버가 전달 받은 후에 MVC(Model-View-Controller) 패턴 중 하나인 Controller가 UI의 사용자 액션에 응답하고 데이터 흐름을 제어하여 On/Off 명령을 송신하고 소켓을 통해

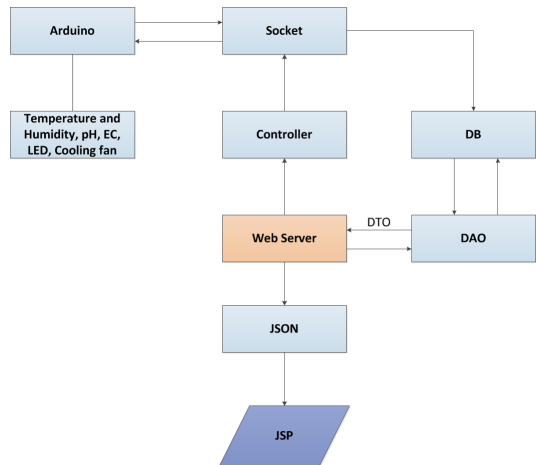


그림 2. 소프트웨어 자바 전체 처리과정도
Fig. 2. Flow chart for the software Java

명령한 동작을 아두이노에서 수행하게 한다. 데이터 모델인 DTO와 DAO 클래스를 활용하였으며, 생육환경 모니터링 및 제어를 위해 servlet을 이용하여 JSP에서 요청 받은 데이터베이스 Select, Insert, Update, 소켓서버연동, 제어 등을 수행하는 역할을 수행할 수 있도록 하였다. 데이터 모델 DAO는 데이터 접근을 목적으로 하기 위한 클래스이며, DTO는 데이터가 포함된 객체를 다른 시스템으로 전달하는 클래스로 DAO에서 접근된 데이터를 기반으로 DTO에 모든 데이터를 순서대로 전달하고, 서버의 자원을 이용하는 JSP는 HTML내에 자바코드를 삽입하여 웹서버에서 동적으로 웹 브라우저를 관리하는 언어이며 servlet은 클라이언트 요청을 처리하고 그 결과를 다시 클라이언트에게 전송하는 프로그램이다. 사용자가 시스템에서 업무 요청을 하면 클라이언트의 자원을 이용하는 자바스크립트(Java Script)에서 데이터 확인 및 가공하여 servlet으로 요청하며 servlet은 데이터를 받아 요청에 맞는 로직을 수행 후 결과값을 JSP로 전달한다. 또한 모니터링 시스템에 데이터를 표현하기 위해서는 객체형식으로 자료를 표현해야하기 때문에 JSON을 이용하였으며, JSON은 단순히 데이터를 받아 객체나 변수로 할당하여 사용하기 위한 도구이다. 그림 3은 JSON의 개념도이다.

JSONObject는 비순서화된 구조를 가지고 있으며, 좌 중괄호로 시작하고 우 중괄호로 끝내어 표현한다. 각 이름 뒤에 콜론을 붙이고 콤마로 이름 또는 값을 구분한다. 그리고 JSONArray의 경우 순서화된 구조를 가지고 있으며, 좌 대괄호로 시작해서 우 대괄호로 끝내어 표현하며, 콤마로 Array의 값들을 구분한다.

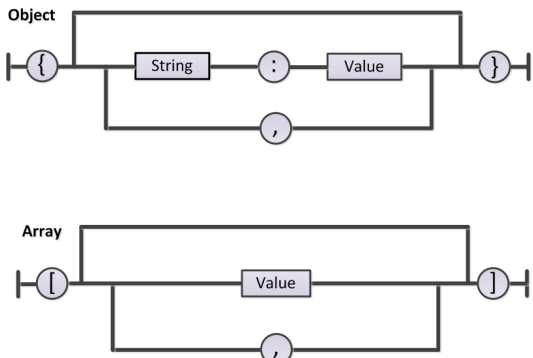


그림 3. 데이터 교환언어 JSON 개념도
Fig. 3. Data exchange language JSON concept

2.3 소켓통신 및 데이터베이스

본 절에서는 아두이노와 자바 간 동일한 네트워크

를 통해 서로 다른 PC에서 수행되는 프로세스 간의 통신 채널인 소켓통신 하는 방법과 발생한 데이터를 데이터베이스로 업데이트하는 방법에 대해 설명한다. 클라이언트인 아두이노와 서버인 자바와의 소켓통신을 위해서는 동일한 네트워크 내에서 이루어져야 하며 PC의 IP주소, Wi-Fi 공유기 아이디와 패스워드 그리고 자바서버와의 PORT를 동일하게 설정하고 아두이노와 자바서버를 구동시키면 Wi-Fi 공유기를 통해 IP주소를 할당받아 설정된 PORT로 접속해 소켓통신이 가능하게 된다. 그림 4는 아두이노 클라이언트가 소켓서버 및 데이터베이스 접속하고 현재 접속한 클라이언트 수의 모습을 나타내고 있다.

```
(Start)
ServerSocket server;
try {
    server = new ServerSocket(serverPort, 100);
    System.out.println("start running");
}

(Connection)
private void conn(){
    String url="jdbc:mysql://localhost:3306/temhum";
    String user="root";
    String password="";
    conn=connectionPool.getConnection(url, user, password);
    System.out.println("MySQL connect success");
}
(Client number)
if ("%" equals(message) && "!" equals(message)) {
    System.out.println("The number of currently
    connected client: " + (clients.size()));
    System.out.println("msg: " + message);
}
```

그림 4. 소켓서버 및 데이터베이스 접속과 현재 접속한 클라이언트 수

Fig. 4. Socket server and database access and a number of clients currently connected

	TEM_DT	temp	hum	tem_dy
<input type="checkbox"/>	2016-05-09 09:48:10	24.96	33	20160509
<input type="checkbox"/>	2016-05-09 09:48:14	24.98	32.83	20160509
<input type="checkbox"/>	2016-05-09 09:48:16	24.99	32.76	20160509
<input type="checkbox"/>	2016-05-09 09:48:20	25.02	35.06	20160509
<input type="checkbox"/>	2016-05-09 09:48:24	25.06	35.65	20160509
<input type="checkbox"/>	2016-05-09 09:48:26	25.06	35.09	20160509

(a)

	tem_dt	ph	tem_dy
<input type="checkbox"/>	2016-05-09 21:22:09	6.540	20160509
<input type="checkbox"/>	2016-05-09 21:22:13	6.540	20160509
<input type="checkbox"/>	2016-05-09 21:22:16	6.540	20160509
<input type="checkbox"/>	2016-05-09 21:22:20	6.540	20160509
<input type="checkbox"/>	2016-05-09 21:22:24	6.540	20160509
<input type="checkbox"/>	2016-05-09 21:22:26	6.540	20160509

(b)

	tem_dt	ec	tem_dy
<input type="checkbox"/>	2016-05-10 08:49:18	1.310	20160510
<input type="checkbox"/>	2016-05-10 08:49:20	1.310	20160510
<input type="checkbox"/>	2016-05-10 08:49:24	1.340	20160510
<input type="checkbox"/>	2016-05-10 08:49:26	1.350	20160510
<input type="checkbox"/>	2016-05-10 08:49:28	1.310	20160510
<input type="checkbox"/>	2016-05-10 08:49:30	1.310	20160510

(c)

그림 5. 데이터베이스에 실시간으로 업데이트된 센서값
Fig. 5. The sensor values updated in real time to the database

그림 4의 왼쪽 그림은 시작 및 접속 파트별 코드를 나타내며, 오른쪽 그림은 코드에 해당하는 실행화면을 나타낸다. Start 파트는 소켓서버를 개방하기 위한 포트설정과 Startrunning인 서버 시작 메시지를 알리는 작업이고 Connection 파트는 데이터베이스의 url 경로, 아이디 및 패스워드 설정과 MySQL 접속 성공 메시지를 알리는 작업이며, Client number 파트는 현재 소켓에 접속한 클라이언트 수를 알려주는 동시에 클라이언트에서 보내는 접속 성공 메시지인 Hello를 자바에 전달한다. 또한 자바는 다수의 기기와 수집 정보를 받아들이어야 하기 때문에 멀티쓰레드 개념을 이용하여 동시에 작업을 할 수 있도록 하였다. 멀티쓰레드는 하나의 프로세스가 다수의 작업을 각각 쓰레드를 이용하여 동시에 작동 시킬 수 있다. 즉, 1:N 통신체계를 말한다. 그림 5는 다수의 클라이언트에 각각 결합된 (a) 온습도 센서, (b) pH센서, (c) EC센서가 소켓 및 데이터베이스에 접속하여 측정된 값들이 실시간으로 각각의 테이블로 업데이트 된 모습이다.

2.4 모니터링 및 제어 시스템

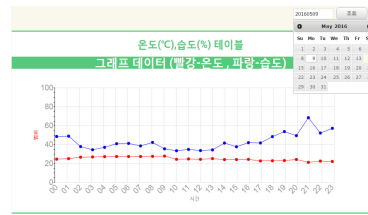
본 절에서는 문자열 형태의 테이블 모니터링 시스템과 그래프로 표현한 그래프형 모니터링 시스템 및 LED, 쿨링팬 제어 시스템에 대해 설명한다. 모니터링 및 제어 시스템은 2.3절 설명과 같이 소켓통신 및 데이터베이스 기반으로 설계하였다. 그림 6은 센서 측정값을 테이블로 모니터링이 가능한 시스템이다.

테이블 모니터링 시스템은 jQuery UI인 jqGrid 라이브러리를 이용하였으며, 제작한 테이블 안은 데이터베이스에 업데이트한 정보를 조회하여 생육환경

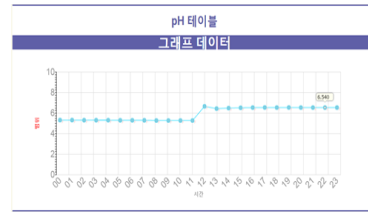
데이터를 문자열 테이블로 표현하였고 최근 18시간 기준으로 모니터링 가능하도록 설계하였다. 그래프형 모니터링 시스템도 동일하게 데이터베이스에 업데이트한 정보를 조회하여 모니터링이 가능하고 추가

시간	온도	습도	pH	EC
2016년 05월 13일 09시				1.360
2016년 05월 13일 10시				1.300
2016년 05월 13일 11시				1.400
2016년 05월 13일 12시				1.330
2016년 05월 13일 13시				1.330
2016년 05월 13일 14시				1.300
2016년 05월 13일 15시				1.360
2016년 05월 13일 16시				1.320
2016년 05월 13일 17시				1.280
2016년 05월 13일 18시				1.380
2016년 05월 13일 19시				1.380
2016년 05월 13일 20시				1.310
2016년 05월 13일 21시				1.360
2016년 05월 13일 22시				1.320
2016년 05월 13일 23시				1.280
2016년 05월 14일 00시				1.380
2016년 05월 14일 01시				1.380
2016년 05월 14일 02시				1.310
2016년 05월 14일 03시				1.360
2016년 05월 14일 04시				1.320
2016년 05월 14일 05시				1.280
2016년 05월 14일 06시				1.380
2016년 05월 14일 07시				1.380
2016년 05월 14일 08시				1.310
2016년 05월 14일 09시				1.360
2016년 05월 14일 10시				1.320
2016년 05월 14일 11시				1.280
2016년 05월 14일 12시	26.3	51.44	5.910	1.380
2016년 05월 14일 13시	26.37	44.65	5.808	1.380
2016년 05월 14일 14시	25.58	41.48	5.800	1.310
2016년 05월 14일 15시	25.16	41.65	5.800	1.360
2016년 05월 14일 16시	24.94	37.09	5.790	1.320
2016년 05월 14일 17시	24.53	41.71	5.790	1.280

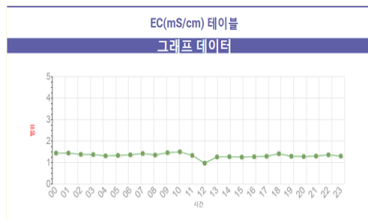
그림 6. 테이블 모니터링 시스템
Fig. 6. Table monitoring system



(a)



(b)



(c)

그림 7. 그래프형 모니터링 시스템
Fig. 7. Graph-type monitoring system

로 다른 날짜의 정보 조회가 가능하도록 달력을 이용하여 설계하였다. 그림 7은 (a) 온습도, (b) pH, (c) EC의 그래프형 모니터링 시스템 모습이다.

그래프형 모니터링 시스템은 rGraph의 Line charts 라이브러리를 이용하였으며 24시간 내내 데이터를 확인할 수 있도록 설계하였다. (a) 온습도 테이블 우측상단 달력을 통해 지속적으로 측정된 데이터를 사용자가 원하는 날짜로 조회하여 온습도, pH, EC의 생육환경 정보를 확인할 수 있다. LED 및 쿨링팬의 제어를 위한 시스템은 클라이언트와 서버 간 On/Off 상태값을 일치시켜야 하며, 사전에 동일한 특정 메시지가 존재해야 하기 때문에 클라이언트와 서버 간 메시지를 공통코드로 정의를 하였고, 공통코드의 양식은 특수문자 4개의 조합으로 자바 및 데이터베이스를 통해 관리하고 있다. 그림 8은 On/Off 제어 시스템이다.

제어 시스템은 웹서버에서 On/Off 명령을 요청하였을 때 해당하는 명령을 수행하고 데이터베이스화하여 실시간으로 제어값을 변경함으로써 현재의 제어상태를 데이터베이스 및 자바에서 파악할 수 있도록 설



그림 8. LED 및 쿨링팬 On/Off 제어 시스템
Fig. 8. LED and cooling fan On / off control system

제하였다. 그림 8의 1번은 On상태로 변경하여 제어할 하드웨어에게 명령을 수행시켜 작동하게 한 모습이며 2번은 Off상태로 변경하여 하드웨어의 작동을 멈추게 한 모습이다. 클라이언트와 서버 간 소켓통신이 이루어져야 On/Off 명령체계를 수행하며 문제가 발생하면 데이터베이스의 On/Off 상태값을 확인하여 대처할 수 있다.

III. 실험

본 실험에서는 생육환경에 필요한 파라미터와 가정이나 회사 내에서 모니터링 및 제어가 가능한 응용 플랫폼을 개발한 시스템을 기반으로 실제로 35일간의 식물 재배를 실험한 과정과 결과를 확인한다.

3.1 양액재배 필요조건

양액재배를 위한 여러 가지 생육환경 조건이 있는데 크게 고려해야할 4가지 조건은 온도, 습도, pH, EC 환경이다. 표 2는 식물 생육환경 최적 환경 파라미터 값을 나타내고 있다.

표 2의 4가지 파라미터의 최적값은 식물마다 다르기 때문에 본 논문에서 고려한 식물의 기준값을 나타내며, 그 중 식물에게 영양분을 공급할 수 있는 EC가 큰 비중을 차지한다. 본 실험에서는 EC는 야마자키 양액법^[7]으로 제조하여 식물에게 영양분을 충분히 공급하였다. 표 3은 야마자키 배양액 조성표를 나타낸다.

표 3의 조성표 양액은 A액과 B액 나뉘는데 단순하

표 2. 생육환경 최적 파라미터 값
Table 2. Growth Environment optimal parameter values

Temperature (°C)	18°C ~ 25°C
Humidity (%)	40% ~ 60%
pH (pH)	5.0pH ~ 7.0pH
EC (mS/cm)	1.2mS/cm ~ 1.7mS/cm

게 화학적 성분차이가 있으며, 염 생성 방지를 위해 A액과 B액을 따로 첨가해야 하고 비율은 50:50으로 섞어야 한다. 또한 광량이 높은 시기에는 작물은 상대적으로 많은 양의 수분을 흡수하게 되며 EC가 급격하게 증가할 수 있어 배액을 통하여 EC의 상승을 방지시켜야 한다.

3.2 청치마와 비타민 재배

본 논문의 실험은 2016년 4월 24일부터 시작하였으며, 고려된 식물은 청치마와 비타민 2가지 종류이다. 또한 최적의 생육환경을 유지하기 위해 지속적으로 모니터링 및 제어를 수행하였다. 온도의 최적 파라미터 중 18°C 이하로 낮아지는 경우는 거의 없었으며, 25°C 이상이 되면 쿨링팬 제어로 온도를 조절하고 습도는 가습기를 이용하여 각각 21°C에서 25°C, 35%에서 55%으로 조절하였다. pH농도는 5.0pH 이하로 내려가는 경우는 없었으며 7.3pH 이상이 되면 황산(H₂SO₄) 또는 질산(HNO₃)을 첨가하여 감소시켜 5.2pH에서 7.0pH 사이를 유지하였다.^[5,8] 만일 pH농도가 5.0pH이하가 되면 암모니아수(NH₄OH)를 첨가하여 올려야 한다. EC농도의 최적 환경을 위해 양액과 물은 10L기준으로 1:500의 비율로 A액과 B액을 각각 20cc만큼 따로첨가하여 1.4mS/cm로 시작하였으며, 1.2mS/cm 이하가 되면 양액을 충분히 첨가하였고 1.7mS/cm 이상이 되면 수도물을 섞어 1.2mS/cm에서 1.7mS/cm 사이를 유지하여 관리하였다. 양액재배를 위해서 페트리디쉬 안에 솜과 그 위에 거즈를 깔고 그 위에 씨앗을 뿌린 후 암실에서 씨앗을 발아시키는 것으로 실험을 진행하였다. 그림 9는 발아를 위한 페트리디쉬 내 청치마와 비타민 씨앗 모습이다.

3일이 경과하여 발아된 씨앗은 상하지 않도록 양액 재배기용 스펀지에 담아 양액재배기로 옮기기 전까지 충분히 뿌리가 자랄 수 있도록 암실에서 발아를 진행하였으며, 씨앗이 마르지 않도록 부족한 수분을 충분



그림 9. 페트리디쉬 안 청치마와 비타민 씨앗
Fig. 9. Petri dish with LeafLettuce, Cheong Chi Ma and Tatsoi seed

표 3. 야마자키 잎상추 배양액 조성표
Table 3. Yamazaki Leaf lettuce nutrient solution composition table

	Fertilizer types	100 drain /1000L	Unit	100 drain /1000L	Unit	200 drain /1000L	Unit	200 drain /1000L	Unit	200 drain /1000L	Unit
A fluid	Calcium nitrate $C_a(NO_3)_2$	23.6	kg	472	g	944	g	28.32	kg	2.832	kg
	Potassium nitrate KNO_3	20.2	kg	404	g	808	g	24.24	kg	2.424	kg
	Fe-EDTA	1.5	kg	30	g	60	g	1.8	kg	0.18	kg
B fluid	Potassium nitrate KNO_3	20.2	kg	404	g	808	g	24.24	kg	2.424	kg
	Ammonium dihydrophosphate	5.8	kg	116	g	232	g	6.96	kg	0.696	kg
	Magnesium sulfate $MgSO_4$	12.3	kg	246	g	492	g	14.76	kg	1.476	kg
	Copper sulfate $CuSO_4$	3.9	g	0.1	g	0.156	g	4.68	g	0.468	g
	Boric acid H_1BO_3	176.3	g	3.5	g	7.052	g	211.56	g	21.156	g
	Manganese sulfate $MnSO_4$	61.5	g	1.2	g	2.46	g	73.8	g	7.38	g
	Zinc sulfate $ZnSO_4$	8.8	g	0.2	g	0.352	g	10.56	g	1.056	g
	Ammonium molybdate $H_{24}Mo_7N_6O_{24}$	9.2	g	0.2	g	0.368	g	11.04	g	1.104	g

히 보충시켜 관리하였다. 그림 10은 스펀지 내에서 6일이 경과된 청치마와 비타민 씨앗의 모습이다. 약 7일에서 8일 경과 후에 양액재배기에 옮길 수 있을 만큼 뿌리가 성장하였을 때 식물이 담겨진 스펀지를 그대로 옮겨 양액재배 실험을 진행하였다. 초기 모니터링 실험 결과 양액 영양섭취 속도가 느렸으며, 식물이 LED와 양액 환경에 익숙해져야했던 것으로 생각

된다. 또한 양액재배기 물의 오염도를 고려하여 주기적으로 갈아주며 진행하였다. 그림 11은 청치마와 비타민 식물이 15일 경과된 모습이다.

15일 경과 후부터 35일까지의 모니터링 실험 결과 중기에 온도가 급격히 증가하여 쿨링팬 제어를 수행하여 조절하였으며, 식물이 LED와 양액 환경에 익숙해져 영양분을 많이 섭취하게 되어 눈에 띄게 빠른 속



그림 10. 6일 경과된 청치마와 비타민
Fig. 10. Six days elapsed LeafLettuce, Cheong Chi Ma and Tatsoi



그림 11. 15일 경과된 청치마와 비타민
Fig. 11. Fifteen days elapsed LeafLettuce, Cheong Chi Ma and Tatsoi



그림 12. 35일 경과한 청치마와 비타민
Fig. 12. Thirty-five days elapsed LeafLettuce, Cheong Chi Ma and Tatsoi

도로 성장하기 시작한 것으로 생각된다. 그림 12는 35일 경과한 청치마와 비타민의 모습이다.

IV. 결 론

본 논문에서는 양액재배용 생육환경 모니터링 및 제어 플랫폼 연구와 플랫폼을 이용하여 생육환경을 관리가 가능한 연구를 진행하였다. 동일한 네트워크 내에서 하드웨어인 아두이노와 소프트웨어인 자바 간 소켓 통신을 하고 센서 측정값을 데이터베이스화하여 온습도, pH, EC에 대한 생육환경을 24시간 내내 모니터링이 가능함을 확인하였다. 또한, 필요시 하드웨어 플랫폼 제어를 수행하여 생육환경 온도를 21℃에서 25℃ 사이로 유지했으며, 습도는 35%에서 55% 사이를 유지하였고, pH농도는 5.2pH에서 7.0pH, EC농도는 1.2mS/cm에서 1.7mS/cm를 유지하였다. 시스템적인 비용이 든다는 단점이 있지만 실험 결과와 같이 제안된 시스템을 기반으로 생육환경 모니터링 및 제어를 수행하여 회사 또는 가정 및 식물공장 내에서 양액재배를 위한 생육환경을 손쉽게 관리할 수 있어 큰 이점이 있다. 더욱이 병해충이 없고 무공해이며, 일반 광원 생육보다 LED를 이용한 생육이 빠르기 때문에 생산의 효율성 좋아 유통, 소비과정^[10]에 큰 장점이 있다. 향후 본 논문에서 제안된 양액재배 모니터링 및 제어 시스템은 관리 측면에서의 편리함과 식물을 재배하여 다양한 먹거리를 제공할 수 있으며 ICT기반 농업 활성화를 위한 참고자료로써 활용될 것으로 예상된다.

References

[1] J. H. Kim, "Plant factory technology

development and commercialization," *The direction Workshops of future agricultural development of plant factory and chungbuk*, pp. 15-31, Jun. 2011.

- [2] T.-H. Yeom, S.-M. Park, H.-I. Kwon, D.-K. Hwang, and J. Kim, "A smart farming system based on visible light communications," *J. KICS*, vol. 38, no. 5, pp. 479-485, May 2013.
- [2] TTA, *Smart Farm*, ICT Standardization Strategy Map, pp. 447-521, 2014.
- [4] J. Moon, I. Jang, Y. C. Choe, J. G. Kim, and G. Bock, "Case study of big data-based agri-food recommendation system according to types of customers," *J. KICS*, vol. 40, no. 05, pp. 903-913, May 2015.
- [5] S.-I. Hwang, J.-M. Joo, and S.-Y. Joo, "ICT-based smart farm factory systems through the case of hydroponic ginseng plant factory," *J. KICS*, vol. 40, no. 4, pp. 780-790, Apr. 2015.
- [6] C.-Y. Lee, Y.-K. Huang, T.-S. Lin, J.-C. Shieh, J.-J. Chou, C.-Y. Lee, and J.-A. Jiang, "A smart fan system for temperature control in plant factory," *EFITA-WCCA-CIGR Conf.*, pp. 24-27, Torino, Italy, Jun. 2013.
- [7] Y. Kwack, D. S. Kim, and C. Chun, "Growth and quality of baby leaf vegetables hydroponically grown in plant factory as affected by composition of nutrient solution," *Protected Horticulture and Plant Factory*, vol. 24, no. 4, pp. 271-274, Dec. 2015.
- [8] S. G. Lee, C. S. Choi, J. G. Lee, Y. A. Jang, C. W. Nam, K.-H. Yeo, H. J. Lee, and Y. C. Um, "Effects of different EC in nutrient solution on growth and quality of red mustard and pak-choi in plant factory," *J. Bio-Environ. Control*, vol. 21, no. 4, pp. 322-326, Dec. 2012.
- [9] G. J. Kim and J. D. Huh, "Trends and prospects of smart farm technology," *2015 Electron. Telecommun. Trends*, vol. 30, no. 5, Oct. 2015.
- [10] J. Hwang, J. H. Seok, and Y. Hyun, "Agri IT convergence technology trends and development prospects," *Inf. and Commun.*

Mag., vol. 30, no. 10, pp. 53-60, Oct. 2013.

- [11] S. Jindarat and P. Wuttidittachotti, "Smart farm monitoring using raspberry pi and arduino," *IEEE I4CT 2015*, pp. 284-288, Kuching, Malaysia, Apr. 2015.

연 인 원 (In-won Yeon)



2014년 2월 : 유한대학교 전자
정보과 전문학사 졸업
2015년 2월 : 국가평생교육원
정보통신공학과 학사 졸업
2015년 3월~현재 : 송실대학교
전자공학과 석사과정
<관심분야> IoT, 식물공장, 전
자공학, 통신공학

이 원 철 (Won-cheol Lee)



1986년 2월 : 서강대학교 전자
공학과 (공학사)
1988년 2월 : 연세대학교 전자
공학과 (공학석사)
1994년 5월 : Polytechnic Institute
of New York University (공
학박사)

2009년 3월~2013년 2월 : 송실대학교 산학협력단
단장

1995년 9월~현재 : 송실대학교 전자정보공학부 교수
<관심분야> Cognitive Radio, Software Define
Radio, TVWS, 이동통신시스템, 적응 빔형성 기
법, 디지털 필터