


# IoT를 사용한 센서 네트워크 기반의 실시간 토양 습도 모니터링

김경헌<sup>1,2</sup>, 김희동<sup>3</sup> 

<sup>1</sup> 경상국립대학교 융합전자공학부

<sup>2</sup> 유타대학교 전기컴퓨터공학과

<sup>3</sup> 세종대학교 전자정보통신공학과 및 지능형드론융합전공

## Real-Time Soil Humidity Monitoring Based on Sensor Network Using IoT

Kyeong Heon Kim<sup>1,2</sup> and Hee-Dong Kim<sup>3</sup>

<sup>1</sup> Department of Convergence Electronic Engineering, Gyeongsang National University, Jinju 52725, Korea

<sup>2</sup> Electrical and Computer Engineering, University of Utah, Salt Lake City 84112, United States

<sup>3</sup> Department of Electrical Engineering and Convergence Engineering for Intelligent Drone, Sejong University, Seoul 05006, Korea

(Received February 22, 2022; Revised April 22, 2022; Accepted May 26, 2022)

**Abstract:** This paper reports a method to use a wireless sensor network deployed in the field to real-time monitor soil moisture, warning when the moisture level reaches a specific value, and wirelessly controlling an additional device (LED or water supply system, etc.). In addition, we report all processes related to wireless irrigation system, including field deployment of sensors, real-time monitoring using a smartphone, data calibration, and control of additional devices deployed in the field by smartphone. A commercially available open-source Internet of Things (IoT) platform, NodeMCU, was used, which was combined with a 9V battery, LED and soil humidity sensor to be integrated into a portable prototype. The IoT-based soil humidity sensor prototype deployed in the field was installed next to a tree for on-site demonstration for the measurement of soil humidity in real-time for about 30 hours, and the measured data was successfully transmitted to a smartphone via Wifi. The measurement data were automatically transmitted via e-mail in the form of a text file, stored on the web, followed by analyses and calibrations. The user can check the humidity of the soil real-time through a personal smartphone. When the humidity of a soil reached a specific value, an additional device, an LED device, placed in the field was successfully controlled through the smartphone. This LED can be easily replaced by other electronic devices such as water supplies, which can also be controlled by smartphones. These results show that farmers can not only monitor the condition of the field real-time through a sensor monitoring system manufactured simply at a low cost but also control additional devices such as irrigation facilities from a distance, thereby reducing unnecessary energy consumption and helping improve agricultural productivity.

**Keywords:** Sensor network, Remote control, IoT, Soil humidity monitoring

---

✉ Hee-Dong Kim; [khd0708@sejong.ac.kr](mailto:khd0708@sejong.ac.kr)

Kyeong Heon Kim and Hee-Dong Kim equally contributed to this work.

Copyright ©2022 KIEEME. All rights reserved.

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

## 1. 서론

정밀 농업에서 효율적인 물 관리는 농업 생산성을 크게 향상시킬 수 있다. 현재 세계의 주요 문제 중 하나가 물 부족이고 그럼에도 불구하고 물은 농업에서 효율적으로 관

리되지 않고 있다. 따라서 저비용으로 효율적으로 물 관리를 할 수 있는 시스템의 개발이 필요하다 [1]. 최근 정보 기술의 발달로 인해 주요 선진국을 중심으로 물을 효율적으로 관리하기 위한 스마트 관개(smart irrigation)가 활발히 연구되고 있다 [2,3]. 특히 센서를 기반으로 하는 사물인터넷(internet of things, IoT)의 발달은 시스템과 사용자간의 상호관계를 강화시켰고, 사용자가 좀더 효과적으로 농업용수를 관리할 수 있도록 도와준다 [2,4]. 이런 스마트 관개는 정확한 정보와 의사소통의 부족으로 인한 물의 낭비를 줄일 수 있기 때문에 수자원을 효율적으로 관리할 수 있을 뿐 아니라 무선 자동화 시스템을 통해 인력과 전력 소비 또한 크게 줄일 수 있다 [1].

IoT 기반 습도 센서는 토양의 수분을 분석하여 사용자에게 원격으로 언제 어디서나 토양의 습도를 체계적으로 알려줄 수 있다 [5]. 즉, 토양의 수분 양을 측정 한 후, 수분이 특정 값 이하로 떨어지면 무선 통신(wireless communication)을 이용해서 시간과 장소에 제한 없이 사용자에게 경고할 수 있다. 이를 통해 물 공급이 필요하지 않음에도 불구하고 일정 시간 간격에 따라 물을 주기적 공급하거나, 반대로 표면상 물이 일시적으로 말라 보임에 따라 잘못된 판단으로 물 공급을 하는 등의 문제를 해결할 수 있다. 이를 통해 사용자는 객관화된 데이터를 근거로 효과적으로 물을 관리할 수 있으며, 물 손실을 최소화할 수 있다 [6,7].

최근 이러한 스마트 관개를 이용한 효과적인 물 관리에 대한 연구가 활발히 진행되고 있다. S. Darshna는 2015년에 ESP8266 Wi-Fi 모듈과 마이크로컨트롤러(micro-controller)를 이용하여 수치적 데이터를 근거로 물이 필요한 영역에 정확한 양의 물을 공급하는 자동화된 스마트 관개 시스템을 보고하였다. 이전의 수동 제어 관개의 경우 건조한 지역에서 필요한 물의 양을 정확히 확인하기 어렵기 때문에 물의 낭비가 심하였다 [1]. P. Singh는 2016년에 완전 자동화로 전력 유지 관리가 쉬운 Arduino 기반의 스마트 관개 시스템을 보고하였다. 하지만 nanoseconds 단위의 지속적인 통신은 사람에게 많은 정보를 제공해주는 반면 전력소비가 큰 단점을 갖는다 [2]. G. Parameswaran는 2016년에 주기적으로 데이터 서버의 업데이트 통해 데이터를 축적하는 Arduino 기반 스마트 관개 시스템을 보고하였다. 축적된 데이터는 추적 가능하고, 일반적이지 않은 상황에 효과적으로 대응할 수 있다 [8]. P. Padalalu는 2017년에 현장의 물을 정확히 측정하고 물 공급을 제어 함으로써 효과적인 물을 관리 할 수 있는 Android 기반 스마트 물 공급 시스템을 보고하였다 [9]. R. K. Kodali는 2017년에 ESP8266를 이용해서 저전력 및 낮은 유지 비용을 갖는 스마트 관개 시스템을 보고 하였다.

암호화 프로토콜을 통해 보안을 강화하였고 모바일 앱과 web 페이지를 통해 토양의 습도와 water pump의 상태를 모니터링 하였다 [10].

이 논문에서는 모바일 앱을 통해 토양의 습도를 실시간으로 모니터링하고 원거리에서 현장의 관개 시스템을 제어할 수 있는 NodeMCU 기반의 스마트 관개 시스템을 보고한다. NodeMCU는 Android에 비해 많은 추가적인 기능을 포함하지 않기 때문에 전력소비와 유지비용을 줄일 수 있다. 측정된 데이터는 수치화 되어 주기적으로 이메일을 통해 사용자에게 보내지기 때문에 데이터를 축적할 수 있을 뿐 아니라 추적이 가능하다. 또한 개인 이메일의 사용은 데이터의 보안을 향상시킬 수 있다. 사물인터넷 기반 스마트 관개에서 측정, 제어, 데이터 저장 및 보정을 모두 구현한 시스템은 보고된 적 없다. 토양 습도 센서는 부식을 막고 정확성을 높이기 위해 capacitive 타입의 습도 센서가 사용되었고 토양의 습도가 약 30시간 동안 모니터링 되었다. 추가적으로 이 습도 센서가 연결된 NodeMCU에 9 V 배터리와 발광다이오드(light-emitting diode, LED)가 결합되었다. 측정된 데이터는 실시간으로 Wi-Fi 기반 무선 통신을 통해 smartphone 앱으로 전송되었다. 또한 데이터는 분석을 위해 Text 파일 형식으로 사용자가 지정한 이메일로 전송되었고 분석 및 보정 되었다. 토양의 습도가 특정 값에 도달하였을 때 원거리의 사용자는 smartphone 앱을 통해 NodeMCU에 연결된 LED에 제어 신호를 보내 전원을 공급하였고, 성공적으로 LED에 불을 켤 수 있었다. 이 논문에서는 간단한 데모를 위해 LED가 사용되지만, LED에 전원 공급 여부만을 제어해주는 것이기 때문에 LED는 물 공급 장치의 전원 부분과 직접적으로 대체될 수 있다. 이러한 무선 모니터링 자동제어 기술은 다양한 센서 및 제어장치와 조합될 수 있으므로 농업 분야뿐만 아니라 환경 모니터링, 스마트 빌딩 등을 포함한 다양한 산업 분야에 응용 적용될 수 있다.

## 2. 모바일 앱을 통해 추적 및 저장 가능한 실시간 토양 습도 모니터링 시스템

### 2.1 전체 시스템 및 동작 원리

그림 1은 구축된 스마트 관개 시스템의 개념도를 보여준다. 시스템은 크게 현장 모니터링과 무선 통신 부분으로 나누어졌다. 첫번째로 현장 모니터링 부분은 실제 현장에서 토양의 습도를 측정하기 위한 하드웨어 부분으로 마이크로컨트롤러 및 무선 통신 모듈(ex. NodeMCU 등), 센서(ex.

토양 습도 센서 등), 제어 장치(ex. LED 또는 water pump의 릴레이 등)로 구성됐다(그림 1 왼쪽). 무선 통신 모듈을 포함한 마이크로 컨트롤러는 현장에서 다양한 센서(온도, 습도, pH 등)를 사용하여 여러가지 변화를 감지할 수 있다. 이 논문에서는 코디네이터 node인 NodeMCU에 연결된 토양 습도 센서를 통해서 토양의 습도의 변화를 측정하고 얻어진 데이터는 통신 모듈을 통해 원거리의 사용자에게 보내주는 시스템을 구현하였다. 한편 NodeMCU는 측정 데이터 전송뿐 아니라 사용자로부터 제어 신호를 받아 연결된 전자장치를 동작시킬 수 있다. 이 논문에서는 모바일 앱을 통해 LED에 전원을 공급하거나 끄는 방식으로 on/off 동작을 제어하였다. 이러한 전원공급 제어를 통한 전자 장치의 on/off 방식은 다른 전자 장치로의 확장성이 뛰어나다. 따라서 복잡한 수정 없이 전압공급 선의 교체만으로도 LED는 water pump를 위한 릴레이로 대체되어 on/off 될 수 있다.

두번째로 무선 통신 부분은 그림 1의 오른쪽에 표현되었다. NodeMCU에서 전송된 데이터를 수집하고, 디스플레이

이하고, 저장하고, 분석 및 보정하고, 전자장치에 제어 명령을 내리는 부분이다. NodeMCU은 Wi-Fi 근거리 무선망을 통해 모바일과 통신할 수 있다. Wi-Fi는 일상에서 가장 많이 사용되는 무선 통신 방식으로 일반 노동환경에서 쉽고 빠르게 접속이 가능한 장점을 갖는다. NodeMCU으로부터 수집된 습도 센서에 의해 생성된 매개변수값은 모바일 앱을 통해 디스플레이 되고, 또한 보안을 위해 Text 파일 형태로 개인 gmail을 통해 자동적으로 laptop으로 전달되었다. 사용자는 얻어진 데이터를 저장/분석/교정/추적할 수 있고, 센서가 감지한 파라미터가 원하는 특정 임계값에 도달했을 경우, 필요에 따라 LED를 on/off 할 수 있다.

### 2.2 하드웨어

그림 2는 실제 제작된 실시간 토양 습도 모니터링 시스템을 보여준다. 9 V 알카라인 건전지, 토양 습도 센서, NodeMCU Module, LED가 결합되었다. 토양 습도 센서는 아날로그 신호를 감지하기 위해 NodeMCU의 A0, GND,

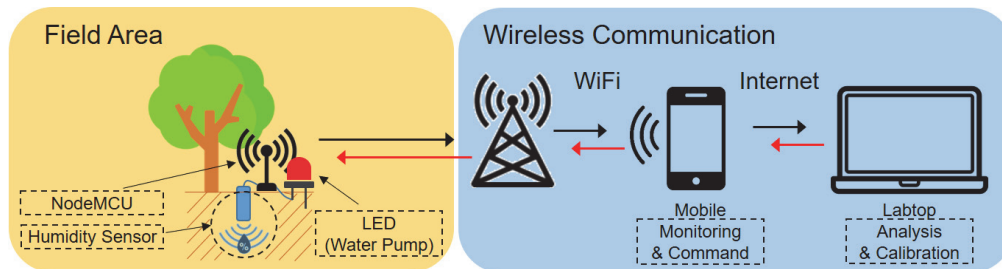


Fig. 1. Conceptual diagram of a built real-time smart sensing system.

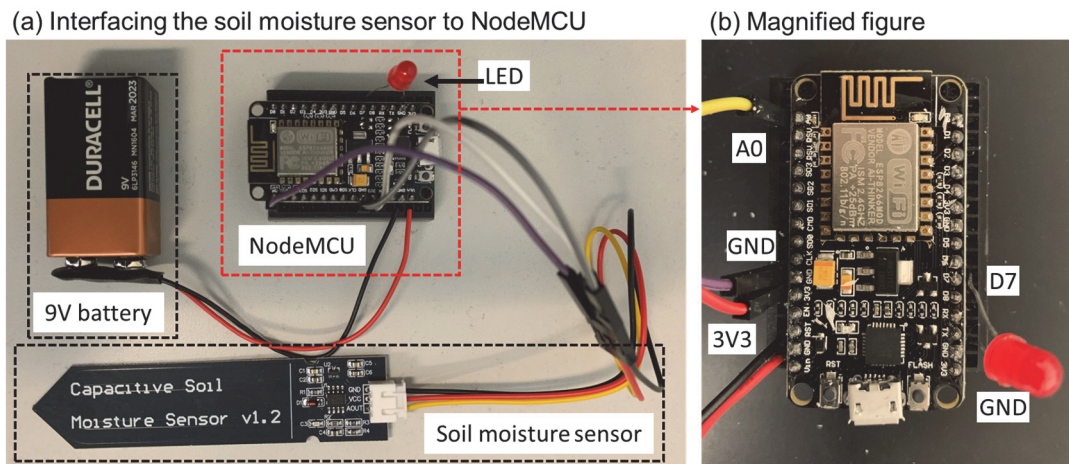


Fig. 2. Real-time soil humidity monitoring system. (a) Combined hardware photographs using 9 volt battery, soil humidity sensor, NodeMCU module, and LED and (b) enlarged NodeMCU connection.

3V3에 연결되었고, LED는 디지털 신호를 받기 위해 D7에 연결되었다. Smartphone에서 제어 신호가 전달되면 D7를 통해 LED에 전원이 공급되거나 차단될 수 있다. 다음은 각 하드웨어에 대한 설명이다.

- ① NodeMCU는 12 V까지 넓은 범위에서 작동가능한 오픈소스 IoT 플랫폼이다. 이 논문에서 9V의 일반 건전지를 사용하여 NodeMCU에 전원이 공급하였다. NodeMCU를 위한 프로그래밍 코드는 ESP8266 라이브러리 설치를 위해 Arduino IDE (integrated development environment) 프로그램이 이용되었고 업로드 되었다. 이 NodeMCU는 32 비트 마이크로컨트롤러, 표준 디지털 주변 장치 인터페이스, 안테나 스위치, RF balun, 전력 증폭기 및 저잡음 수신 증폭기, 필터 및 전력 관리 모듈을 작은 패키지로 통합한 SoC (system-on-chip)이다. 통신을 위해 ESP8266 Wi-Fi SoC에서 동작하는 firmware를 포함하고 있으며, ESP-12 module 기반의 하드웨어를 가지고 있다. Firmware는 Lua scripting 언어가 사용되었다 [11].
- ② 토양습도센서는 3.3 V 작업 환경을 지원하는 전압 레귤레이터(regulator) 칩과 높은 시간 정밀도를 갖는 TLC555CDR칩이 내장된 capacitive 타입의 센서가 사용되었다. 이 센서는 토양의 습도를 아날로그값으로 출력할 수 있다. 따라서 외부 아날로그-디지털 신호 변환 모듈(analog-to-digital conversion, ADC)을 연결하면 센서를 정상적으로 동작 시킬 수 있다. 특히 Capacitive 타입의 센서는 resistive 타입의 센서와 달리 메탈이 노출되지 않아 메탈의 부식을 막을 수 있고 높은 정확성을 갖는다.

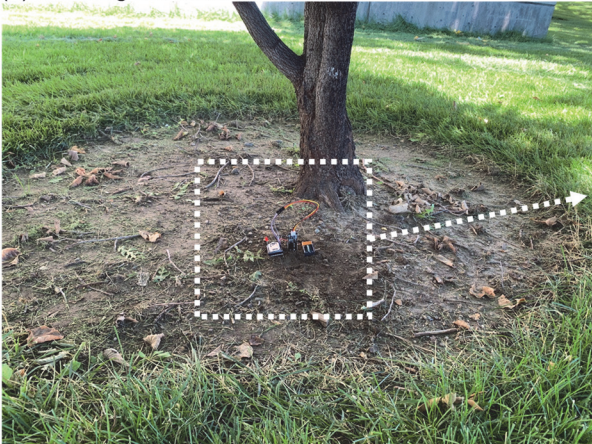
## 2.3 소프트웨어

- ① 2.2에서 언급한 것처럼 NodeMCU에 작성된 코드를 업로드 하기 위해 오픈 소스 소프트웨어인 Arduino IDE 프로그램이 사용되었다. 소스 코드는 Windows OS 기반에서 C 프로그램 언어를 사용해서 작성되었다. NodeMCU에 코드 업로드를 위한 설정값은 Flash Size는 4M (3M SPIFFS), CPU Frequency는 80 Mhz, Upload Speed: 921600로 설정되었다. 작성된 코드는 Compile 후 오류 없이 업로드 되었다.
- ② 센서에 의해 측정된 데이터 값을 smartphone을 통해 모니터링하고 원격으로 NodeMCU에 제어 신호를 보내기 위해 사물 인터넷 플랫폼인 Blynk 앱이 사용되었다. Blynk은 Android OS 기반에서 동작되었으며 측정된 값은 그래픽 사용자 인터페이스 (graphical user interface, GUI) 기술을 사용하여 사용자에게 보여진다 [12].

## 3. 실험과정 및 결과

제작된 센서 노드는 유타 대학교(University of Utah) 캠퍼스 안의 Wi-Fi를 사용할 수 있는 지역에서 설치되었다. 그림 3은 실제 설치된 센서 노드의 사진을 보여준다. (a)는 나무 옆에 설치된 센서 노드의 모습을 보여주고, (b)는 확대된 사진을 보여준다. 센서노드는 Wi-Fi 연결 및 관찰이 용이하도록 숲이 아닌 나무 한 그루가 독립적으로 있는 개방된 위치에 설치되었다. 토양의 습도는 2019년 8월

(a) Installing soil moisture sensors in the field



(b) Magnified figure

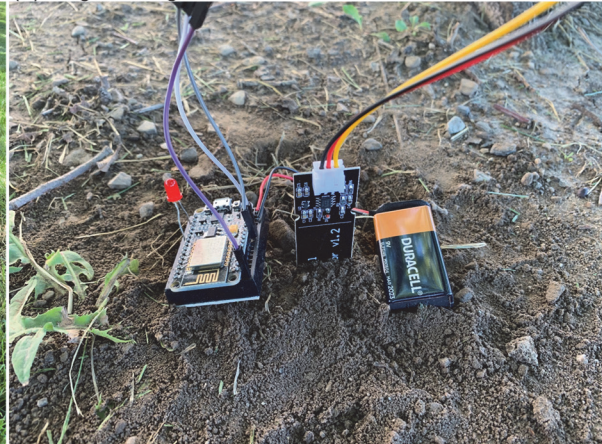


Fig. 3. (a) Sensor nodes installed on site and (b) enlarged images.

4일 오후 2시부터 8월 5일 오후 8시까지 약 30시간 동안 측정되었다. 습도 측정은 1시간에 25회 측정되었다.

그림 4는 Blynk mobile app을 이용해서 모니터링 된 토양의 수분 변화를 보여준다. (b)는 Blynk app 동작화면이고 (a)는 (b)의 가운데 그래프 부분을 확대된 사진이다. 아날로그 출력을 갖는 습도 센서는 2019년 8월 4일 오후 2시부터 8월 5일 오후 8시까지 약 30시간 동안 토양의 습도를 모니터링 하였다. 데이터 값은 초반 디바이스 안정화를 위한 급격한 기율기 변화 부분을 제외하고, 약 780%에서 630%의 범위를 가진다. 센서 노드의 정상적 동작을 확인하기 위해 약 20시간이 지난 8월 5일 오전 9시 50분에 Blynk app을 통해 센서노드로 LED on 제어 신호를 보냈다. 그림 4(b)의 왼쪽 위의 녹색 동그라미는 NodeMCU에 연결된 전자장치에 전원을 공급하기 위한 제어 스위치이다. 이 스위치를 통해 성공적으로 LED를 동작 시킬 수 있었다. 앞에서 언급했듯이 이 LED는 water pump를 위한 릴레이로 대체

될 수 있다. 또한 토양 습도센서의 정상 동작을 확인하기 위해 토양에 물을 공급하였고, 습도의 변화가 측정되었다. 그림 4(b)의 아래쪽의 이미지는 센서노드의 위성항법시스템(global positioning system, GPS) 정보를 보여준다. Blynk app는 GPS를 사용하여 구글 맵을 통해 NodeMCU의 위치정보를 보여준다. 따라서 특정 위치의 토양 습도를 정확히 확인할 수 있다. 그림 4(b)의 오른쪽 위의 Humidity value 값은 실시간으로 토양의 습도를 보여준다.

그림 4(a)에서 보는 것처럼, Blynk app은 토양 습도 센서의 아날로그 값을 그대로 디스플레이 하기 때문에 데이터 값의 보정이 필요하다. 그림 4(a)에서 물공급에 의해 토양의 습도가 상승할 경우, 반대로 값이 떨어진다. 또한 780%에서 630%의 습도값의 범위도 0%에서 100% 범위안으로 보정하는 것이 필요하다. 따라서 센서에 의해 얻어진 데이터를 엑셀(Microsoft Excel)을 사용하여 보정 되었다.

그림 5는 센서값 보정을 위해 실험을 보여준다. (a)는 실

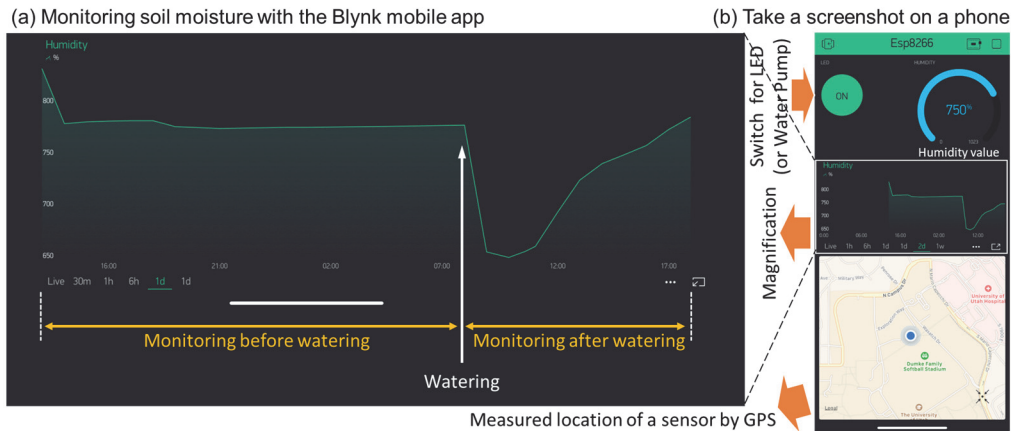


Fig. 4. Screen shot of Blynk mobile app. (a) Change trend in humidity and (b) blynk mobile app action.

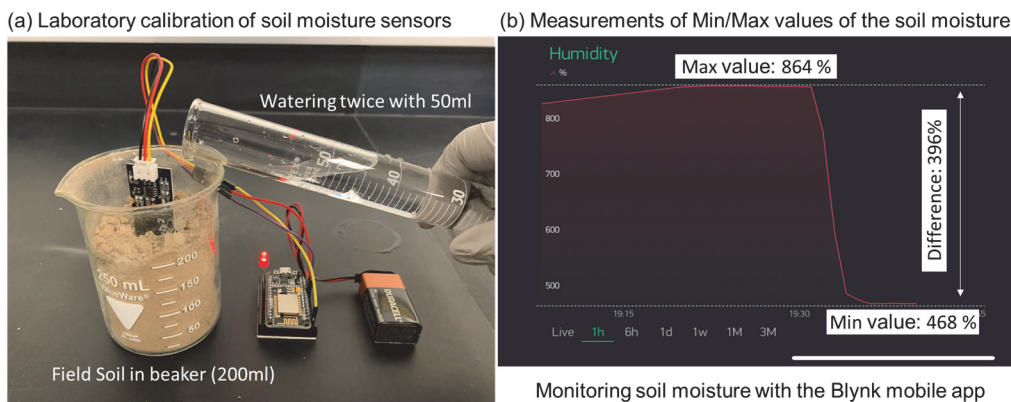


Fig. 5. Experiment with sensor calibration. (a) Experiment with calibration and (b) humidity graph displayed via Blynk app.

힘 과정을 보여주고, (b)는 Blynk app을 통해 디스플레이 된 습도그래프를 보여준다. 그림 5(a)에서 200 ml의 토양은 150도 핫 플레이트 위에서 약 3시간 동안 완전 건조되었다. 건조된 토양은 그림 5(b)에서 볼 수 있듯이 약 864%의 습도값을 보여준다. 건조된 흙에 약 100 ml (50 ml 두 번)의 물이 공급되었다(그림 5(a)). 충분히 젖은 토양은 약 468%의 습도 값을 나타낸다. 최대값과 최소값 사이의 차이는 약 396%를 보여주었다(그림 5(b)). 센서값 보정을 위해, 토양이 완전히 젖었을 때를 864%로 가정할 수 있고, 토양이 완전히 말랐을 때를 468%로 가정할 수 있다. 이것을 근거로 그림 4의 습도값을 보정하였다.

그림 6은 보정된 습도 그래프를 보여준다. 그림 5(b)에서 864%는 습도 100%로 보정되었고, 468%는 0%로 보정되었다. 또한 이 보정값을 기준으로 그림 4에서 측정된 데이터는 보정되었다. 결과적으로 8월초 Salt Lake City, Utah, USA의 토양의 습도는 평균 23%를 유지하고

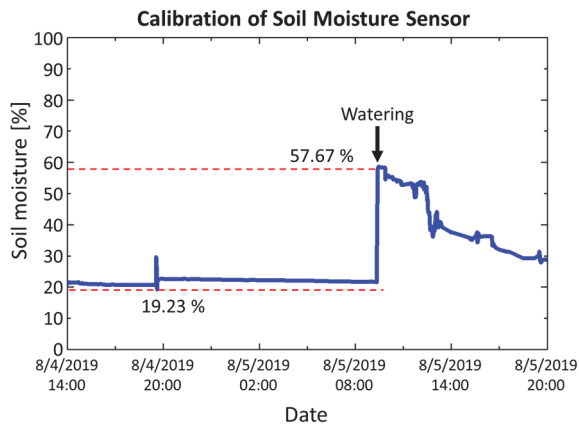


Fig. 6. Calibrated humidity graph.

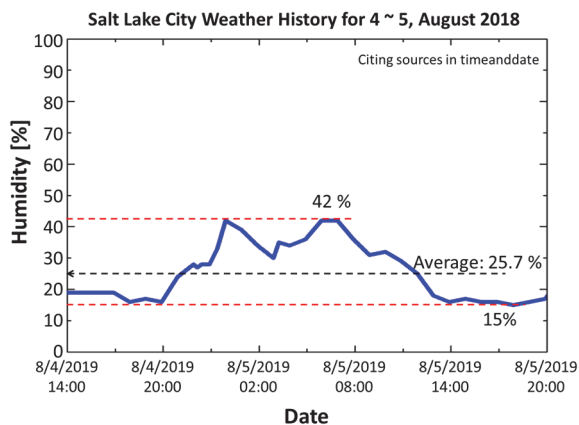


Fig. 7. Atmospheric humidity graphs in salt lake city.

있다. 또한 약 50 ml의 물이 나무에 공급되었을 때, 최대 57.67%까지 토양의 습도가 증가하였다. 증가한 습도는 시간이 지남에 따라 평균값으로 근접해 갔다. 토양의 습도가 약 30% 밑으로 떨어질 때까지 약 10시간이 걸렸다. 그림 7은 timeanddate.com라는 날씨 사이트에 의해 검색된 Salt Lake City의 대기중 습도를 보여준다. 대기의 습도와 토양의 습도는 정확히 일치하지 않았다. 토양의 습도는 평균 23%로 거의 일정하게 유지되는 반면(그림 6), 대기의 습도는 최소 15%에서 최대 42%로 약 27%의 편차를 가졌고 평균적으로는 약 25.7%의 습도를 보여주었다.

#### 4. 결론

이 논문은 토양의 습도를 모바일 앱을 통해 실시간으로 모니터링하고, 얻어진 데이터를 전송/저장/추적/보정할 수 있으며, 모바일을 통해 제어신호를 보낼 수 있는 IoT 기반 센서 노드의 개발 및 실증을 보고하였다. 센서 노드는 현장에서 9 V 배터리를 사용해서 약 30시간 동안 동작하였고, 토양의 습도는 평균 23%로 측정되었다. 센서노드에 연결된 LED 전자장치는 모바일을 통해서 on 제어 신호를 보냈을 때 정상적으로 전원이 공급되어 발광 되는 것을 확인하였다. 이를 통해 LED가 water pump에 연결된 릴레이로 대체되었을 때, 물을 공급할 수 있는 가능성을 확인하였다. 이러한 시스템은 저비용으로 스마트 관개 시스템을 구축할 수 있어 물이 부족한 저소득 국가의 농민이 객관적인 습도 데이터를 바탕으로 효율적으로 물을 사용하는데 도움을 줄 것으로 예상된다.

#### ORCID

Hee-Dong Kim

<https://orcid.org/0000-0003-1555-7509>

#### 감사의 글

이 연구는 교육부 지원 한국연구재단(NRF)을 통한 기초 과학연구사업의 지원을 받았습니다(No. NRF-2022R1F1A1060655).

#### REFERENCES

- [1] S. Darshna, T. Sangavi, S. Mohan, A. Soundharya, and S. Desikan, *IOSR-JECE*, **10**, 32 (2015).

- [2] P. Singh and S. Saikia, *Proc. 2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)* (IEEE, Agra, India, 2016) pp. 1-4. [DOI: <https://doi.org/10.1109/R10-HTC.2016.7906792>]
- [3] P. Padalalu, S. Mahajan, K. Dabir, S. Mitkarand, and D. Javale, *Proc. 2017 2nd International Conference for Convergence in Technology (I2CT)* (IEEE, Mumbai, India, 2017) pp. 659-662. [DOI: <https://doi.org/10.1109/I2CT.2017.8226212>]
- [4] J. Shenoy and Y. Pingle, *Proc. 3rd International Conference on Computing for Sustainable Global Development (INDIACom)* (IEEE, New Delhi, India, 2016) pp. 1456-1458.
- [5] V. H. Andaluz, A. Y. Tovar, K. D. Bedón, J. S. Ortiz, and E. Pruna, *Proc. 2016 IEEE International Conference on Automatica (ICA-ACCA)* (IEEE, Curico, Chile, 2016) pp. 1-6. [DOI: <https://doi.org/10.1109/ICA-ACCA.2016.7778389>]
- [6] K. Lekjaroen, R. Ponganantayotin, A. Charoenrat, S. Funilkul, U. Supasitthimethee, and T. Triyason, *Proc. 2016 International Computer Science and Engineering Conference (ICSEC)* (IEEE, Chiang Mai, Thailand, 2016) pp. 1-6, [DOI: <https://doi.org/10.1109/ICSEC.2016.7859873>]
- [7] P. Srivastava, M. Bajaj and A. S. Rana, *Proc. 2018 Fourth International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)* (IEEE, Chennai, India, 2018) pp. 1-5, [DOI: <https://doi.org/10.1109/AEEICB.2018.8480949>]
- [8] G. Parameswaran and K. Sivaprasath, *IJESCI*, **6**, 5518 (2016).
- [9] C. Kamienski, J. P. Soininen, M. Taumberger, R. Dantas, A. Toscano, T. S. Cinotti, R. F. Maia, and A. T. Neto, *Sensors*, **19**, 276 (2019). [DOI: <https://doi.org/10.3390/s19020276>]
- [10] R. K. Kodali, B. S. Sarjerao, *Proc. 2017 IEEE Region 10 Symposium (TENSYP)* (IEEE, Cochin, India, 2017) pp. 1-5. [DOI: <https://doi.org/10.1109/TENCONSpring.2017.8070095>]
- [11] NodeMcu, <http://www.nodemcu.com/>
- [12] Blynk, <https://blynk.io/>