Design and Implementation of IoT-Based Smart-farm Management System

Hee-Dong Park, Jung-Woong Park

Abstract*: *This paper proposes an IoT-based smart-farm management system. The proposed system consists of a control unit using arduino with sensors, an agent program for controlling the smart-farm system, and a web application for users. The controller unit transmits sensed data such as temperature, humidity, light and moisture to the agent program, and the agent program stores the data in the DB or transmits user's control value to the controller unit. Users can check the sensing information of the farm through the web or remotely control various actuators. In addition, the proposed smart-farm system includes autonomous control functions based on context awareness. As a result of the implementation, it was confirmed that the proposed smart-farm system model works very well.

Index Terms*: *Smart-farm, IoT, Sensors, Arduino, Context awareness

I. INTRODUCTION

There have been so many cases in which various information and communication technologies (ICT) are fused and utilized in many fields. Especially, researches are being actively carried out to utilize the Internet technology of objects in various industrial fields as well as home appliances and indoor environment control at home [1-3]. These ICT convergence studies are also actively being carried out in agriculture [4], and smart-farm is a typical example of ICT application in agriculture. There have been many researches on adapting wireless sensor network technology to agriculture [5, 6]. Recently, advanced IoT technology has been applied to smart-farm [7]. In addition, web/Internet, mobile, and computer vision are also important techniques in configuring smart-farms [8-11].

This paper proposes a smart-farm management system model that can not only monitor in real time various environmental information affecting crops such as temperature and humidity of greenhouse, but also automatically maintain the greenhouse environment all day long. The composition of this paper is as follows: the configuration of the proposed system is described in Section Ⅱ and then the design and operation algorithms of the proposed system is described in Section Ⅲ. Section Ⅳ shows the implementation results and then Section Ⅴ makes a conclusion.

Revised Manuscript Received on July 05, 2019

Hee-Dong Park, Associate Professor, School of IT Convergence, Korea Nazarene University/ Cheonan-city, Korea. Corresponding author.

Jung-Woong Park, School of IT Convergence, Korea Nazarene University/ Cheonan-city, Korea.

II. CONFIGURATION OF THE PROPOSED SYSTEM

The proposed system has an Arduino-based environment control device which is designed to be able to monitor and control smart-farm environment remotely via the web. As shown in Fig. 1, the overall hardware configuration consists of actuators for environment control and sensors for environment value measurement.

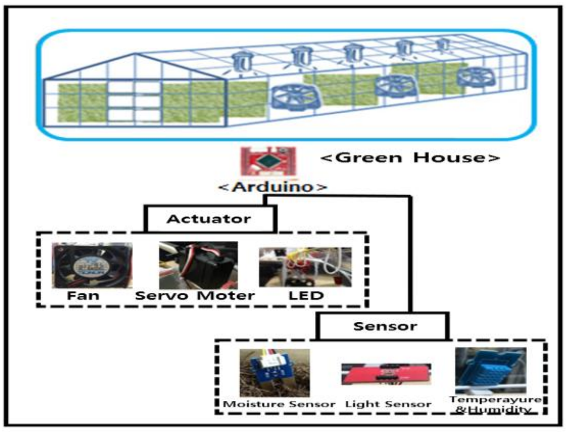

Fig. 1. H/W configuration of the proposed system.

Greenhouse ventilation facilities and light intensity control facilities are needed to control environmental factors such as temperature, humidity, Co2 concentration, light intensity, etc. Therefore, actuators for controlling each environmental factor include a small fan, a servomotor capable of 360 ° rotation, and four LEDs. A ventilation facility of the greenhouse is constructed with a small fan and a greenhouse ceiling opening/closing system using a servomotor. Four LEDs were arranged on the ceiling to control the light amount. In order to monitor the environment in the greenhouse in real time, temperature and humidity integrated sensors, moisture sensors, and light sensors were installed to measure environmental factors. A moisture sensor checks the moisture in the soil by measuring the change of the amount of current flowing between the conductors.

Fig. 2 shows data communication of the proposed system. The local PC in the greenhouse exchange data with the web server using TCP/IP communication. It monitors environmental values sensed inside the greenhouse and controls actuators through Arduino-based environment control device using an agent program. The agent program exchanges sensed data and control messages with Arduino using serial communication.

Users can monitor and control the greenhouse through the web interface provided by the web server. User's requests are delivered to the agent program of the local PC via web server. The agent program receives sensed data at regular intervals from the Arduino and stores them in the DB. The web server remotely controls or monitors the greenhouse by sending an environmental sensing or control request to the agent program when the user requests it.

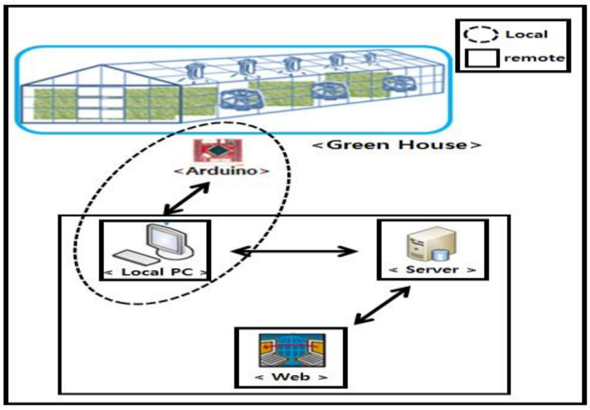

Fig. 2. Data communication.

III. SYSTEM DESIGN AND OPERATING ALGORITHM

The smart-farm management system is divided into three parts: control unit, agent, and web application.

A. Control Unit

The control unit consists of an Arduino and sensors. The operating algorithm of the Arduino is shown in Fig. 3. Arduino receives data from installed sensors at regular intervals using timer interrupt [12] and exchanges data with the agent program through serial communication. As shown in Fig. 4, the data is received as a character string, which can be parsed through the delimiter.

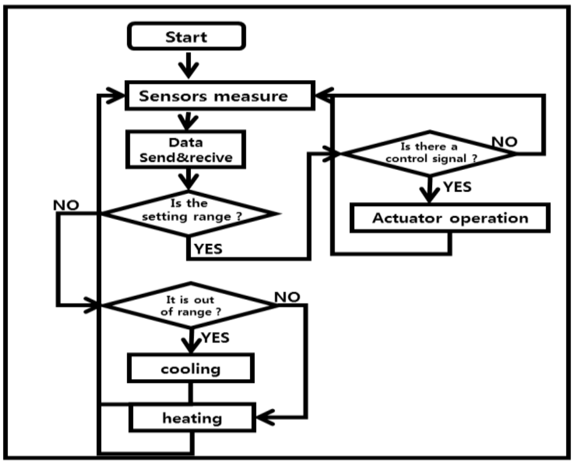

Fig. 3. Operating algorithm of the control unit.

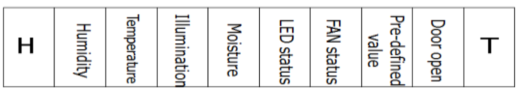

Fig. 4. Data format.

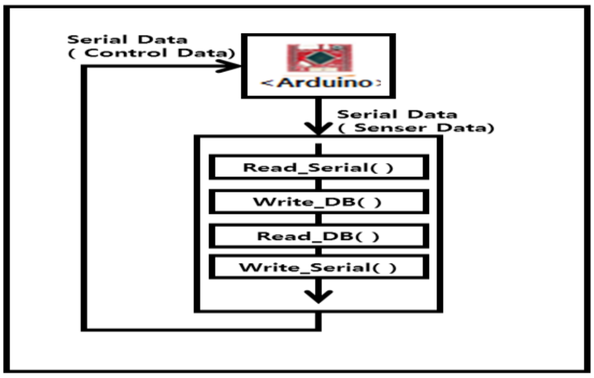

Fig. 5. Agent's class configuration and operating procedure.

B. Agent Program

The agent program processes data received from Arduino. It is a core program of local PC for storing data in DB, processing remote control signal, and greenhouse control. It is implemented in Java and its class configuration and operating procedure is as shown in Fig. 5.

C. Web Application

Web application provides users with user interfaces to monitor and control the greenhouse. It calls up sensed data stored in DB and displays them on a chart. Users can easily control the greenhouse through the web from a remote site. Web application is implemented using JSP and servlet. The newly defined and implemented classes are shown in Table 1.

Table 1. Web application classes.

Class names	Functions
DBConnection	DB connectioin
LoginCK	login check
AddMember	adds member
ReadSensorData	reads sensor data of DB
SendControlData	sends control data

IV. IMPLEMENTATION RESULTS

The implementation environment and hardware specifications are shown in Table 2 and Table 3, respectively.

Table 2. Implementation environment.

Language	Java, JSP, C
Web application server	Tomcat 8
DBMS	MySQL
Arduino type	mega 2560

Table 3. H/W specification.

H/W	Model no.
Illuminance sensor	TSP09541
Moisture sensor	TE92355PSEN
Temperature/Humidity sensor	TE11301PSEN
Servo-motor	TSR9065
Ventilator motor	F6025M05B

A. Control Unit

The library corresponding to each H/W should be added to the Arduino control code as shown in Table 4. Since the timer interrupt is declared before the setup function and the timer interrupt occurs every second, the count variable should be incremented by 1 every time a timer interrupt occurs. The value of each sensor is transmitted via serial communication whenever count variable is 5 by timer interrupt. The transmission data consist of sensing data and the state of actuator with the character ‘,’.

Table 4. Libraries.

H/W	Libraries
Sensors (Temperature, Humidity)	DHT.h
Servo-motor	Servo.h
Timer	MsTimer2.h

B. Agent

The agent uses the RXTX API to support serial communication with Arduino and uses Swing for GUI implementation. The agent reads the serial value and stores it in DB after checking the start character and end character. In addition, it reads control data and sends them to Arduino. Fig. 6 shows the user interface of the agent program: 1) Setup serial communication with Arduino, 2) Setup DB connection, 3) Refresh or delete the log window, 4) Log window for displaying connection information and sensor values, and 5) Greenhouse control or temperature setup.

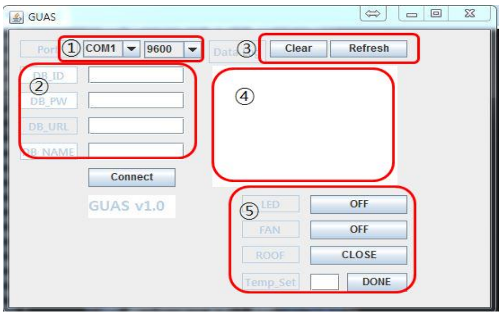

Fig. 6. User interface of agent program.

C. Web Application

With the web application, users can monitor sensed data stored in DB. And users can also monitor and control actuators such as LED, fan, and ceiling opening/closing system.

Fig. 7. User interface of web application.

D. Model of a Greenhouse

Fig. 8 shows a model of greenhouse which is made to verify the operation of the proposed system. Fig. 8. Four LEDs were placed on the ceiling of the greenhouse and ventilator, light, and temperature and humidity sensors were placed on the wall. The water sensor was installed in the soil and the servo motor was installed outside of the ceiling of the greenhouse, and the vinyl was rolled up to open and close the ceiling. Fig. 9 shows the actual operation of the proposed system model.

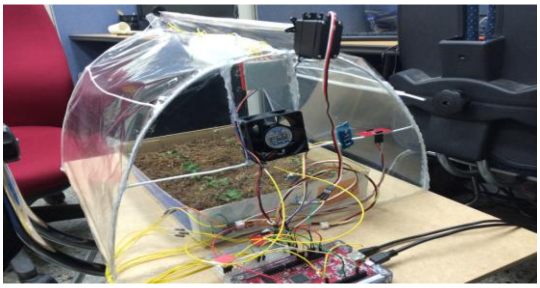

Fig. 8. A model of greenhouse.

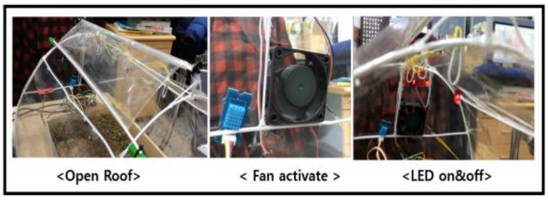

Fig. 9. Operating experiment.

V. CONCLUSION

Recently, ICT convergence has been attempted in various fields. In particular, the agricultural sector is emphasizing the creation of new growth engines and added value through ICT convergence, and related projects are under way. This paper proposed a smart-farm management system model that utilizes Arduino, which makes it easier and less expensive to construct the system than the existing greenhouse management system. Users can easily monitor the greenhouse environment and control various actuators remotely through the web interface provided by the proposed system. In addition, the proposed system provides smart management functions through context awareness and autonomous control functions. Implementation results confirmed that the proposed system works very well.

ACKNOWLEDGMENT

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2017R1D1A3B06035024). This research was also supported by Korea Nazarene University Research Grants.

REFERENCES

1. A. Luigi, L. Antonio, M. Giacomo, “The Internet of Things: a survey,” *Int. J. Comput, Telecommun. Netw*. Vol. 54, No. 15, pp. 2787–2805, 2010.
2. EC, “Internet of Things - An action plan for Europe,” Commission of the European Communities, 2009.
3. “ The Internet of Things: The Future of Consumer Adoption,” ACQUITY GROUP’S 2014 INTERNET OF THINGS STUDY.
4. Omid, M., “A computer-based monitoring system to maintain optimum air temperature and relative humidity in greenhouse,” *International journal of agriculture and biology*, Vol. 6, pp. 1084-1088, 2004. 5
5. Y. Zhou, X. Yang, X. Guo, M. Zhou, L. Wang, “A Design of Greenhouse Monitoring & Control System Based on ZigBee Wireless Sensor Network,” *International Conference on Wireless Communications, Networking and Mobile Computing*, 2007. 7
6. Striemer, G.M., D.L. Story, A. Akoglu, and M. Kacira, “A node and network level self-recovering distributed wireless sensor architecture for real-time crop monitoring in greenhouses,” *Transaction of the ASABE*, Vol. 54, No. 4, pp. 1521-1527, 2011. 14
7. Li, G., W. Zhang, and Y. Zhang, “A design of the IOT gateway for agricultural greenhouse,” *Sensors & Transducers*, Vol. 172, No. 6, pp.75-80, 2014. 23
8. Y. S. Choi, H. J. Lee, S. T. Joung, “A Design and Implementation of Web-based Green House Automation System,” *Journal of The Korea Institute of Electronic Communication Sciences*, Vol. 7, No. 6, pp. 1519-1527, 2012. 28
9. David, S. and M. Kacira, “Design and implementation of a computer vision-guided greenhouse crop diagnostics system,” *Machine vision and applications*, Vol. 26, Issue 4, pp. 495-506, 2015. 30
10. J. W. Park, Y. M. Choi, H. D. Park, “Design and Implementation of Smart Green House Management System,” *Journal of Digital Convergence*, Vol. 14, No. 2, pp. 259-264, 2016. 27
11. Zheng, L., M. Li, C. Wu, H. Ye, R. Ji, X. Deng, and W. Guo, “Development of a smart mobile farming service system,” *Mathematical and computer modelling*, Vol. 54, No. 3, pp. 1194-1203, 2011. 29
12. J. H. Kim, J. H. Jung, “Implementation of Wireless Communication Module with Pointto-multipoint Media Access Control,” *The Journal of IIBC*, Vol. 12, No. 5, pp. 267-273, 2012. 32