Smart Greenhouse Management System based on NB-IoT and Smartphone

Fan Zhang
College of Information Science and Technology, Donghua University
Shanghai, China
2181287@mail.dhu.edu.cn

Xuefen Wan*
College of Computer/Hebei IoT Monitoring Engineering Technology Research Center, North China Institute of Science and Technology
Langfang, China
calmerd@ncist.edu.cn

Tao Zheng*
School of Economics and Management, YanShan University
Qinghuangdao, China
ztao@ysu.edu.cn

Jian Cui
School of Cyber Science and Technology, Beijing University of Aeronautics and Astronautics
Beijing, China
cuijianw@buaa.edu.cn

Xuedong Li
Qinghuangdao National Aquatic Germplasm Resources Protection Zone Management
Qinghuangdao, China
calmered@163.com

Yi Yang
College of Information Science and Technology, Donghua University
Shanghai, China
yiyang@dhu.edu.cn

Abstract—**In order to realize intelligent agricultural greenhouse management and improve the density and refinement of the sensing and control terminal, a smart greenhouse management system based on NB-IoT (Narrow Band Internet of Things) and smartphone is designed. The terminal node supports NB-IoT network and Bluetooth communication. Nodes periodically collect environmental data such as temperature, humidity, light intensity, wind speed and wind direction, and upload data to the cloud platform in real time through the NB-IoT network. The cloud platform that suit for NB-IoT is designed. The platform includes data storage, server-side and web applications for centralized management. The web application implements many functions such as user management, node information management, real-time monitoring, alarm recording and historical information querying. The Android APP is designed for distributed on-sited management of greenhouses. The APP implements real-time monitoring and temporary data storage of a single node. The test results show that the system has stable data transmission, the high packet transmission success rate in the greenhouse environment which meets the real-time management requirements for the greenhouse.**

Keywords—NB-IoT; smartphone; smart agriculture; cloud platform; Android APP; hybrid management

I. Introduction

In recent years, intelligent management systems such as artificial intelligence and big data have higher requirements for the density and refinement of the underlying sensing. Therefore, the reliability requirements of the sensing terminal and the execution terminal deployed in the greenhouse are higher [1-2]. At present, there are some problems in the system of greenhouse planting: (1) the cost is limited and low cost terminals is required, (2) the system require low power consumption, and (3) the agricultural environment is complex and diverse, it is difficult to deploy terminals by wired means.

In order to solve the above-mentioned problems, the researchers carried out related research work and completed the design of the greenhouse management system based on the traditional GPRS and ZigBee technologies [3-6]. GPRS technology has the advantages of low cost and long-distance communication, but the higher power consumption reduces the life of the terminal node. ZigBee technology effectively reduces the energy consumption and cost of terminal nodes, but ZigBee's single-hop communication distance is short and the deployment of its wireless sensor network is more complicated. It is necessary to build a multi-hop wireless network and set up a gateway. Compared with GPRS and ZigBee, NB-IoT technology has obvious advantages in low-power and long-distance communication [7]. NB-IoT still has good communication ability under the conditions of dense soil and crops. With the promotion of NB-IoT technology, its use cost will be further reduced. This article introduces a smart greenhouse IoT management system based on NB-IoT and smartphone. Centralized management of agricultural greenhouses is realized through NB-IoT, and distributed on-site management services are provided through smartphones.

II. Transmission Characteristics of NB-IoT in Agricultural Environment

NB-IoT is developed on the basis of LTE (Long Term Evolution) and widely uses LTE technology such as OFDMA (downlink orthogonal frequency division multiple access), SC-FDMA (downlink single carrier frequency division multiple access technology), \rate matching, and channel coding, etc. [8]. The transmission scheme of NB-IoT is shown in Table 1.

This work is supported by Langfang Research and Development Program about Science & Technology (2019011010), Qinhuangdao Science and Technology Research and Development Program (201805A016), Key Accident Prevention Technology Project for State Administration of Work Safety (zhishu-0016-2017AQ), China National Coal Association Science and Technology Research Guiding Program (MTKJ2017-311) and Hebei IoT Monitoring Engineering Technology Research Center (3142018055).
*Corresponding Authors: Calmerd@ncist.edu.cn, Ztao@ysu.edu.cn

978-1-7281-6167-9/20/$31.00 ©2020 IEEE

TABLE I. NB-IOT TRANSMISSION SCHEME

Link	Transmission scheme	
Uplink	BPSK/QPSK modulation	
	SC-FDMA	Single subcarrier, Subcarrier spacing is 3.75kHz and 15kHz
		Multiple subcarriers with subcarrier spacing of 15kHz
Downlink	QPSK modulation	
	OFDMA, Subcarrier spacing is 15kHz	

In the application of the agricultural Internet of Things such as crop planting and aquaculture, a large number of sensor nodes are often deployed in the agricultural production environment. The sensor nodes collect environmental information and transmit it through wireless network technology to achieve remote monitoring and control. In the actual agricultural production environment, environmental variables usually change little within a short time. The collection terminal often uses hours and days as the upload cycle. The control terminal intermittently receives control command. The frequency of data transmission is low and the amount of data each time is small. Agricultural Internet of Things has high tolerance for data transmission delay. The characteristics of data transmission in the agricultural environment are consistent with the characteristics of NB-IoT data transmission with low frequency, high delay, and a small amount of data [9]. At the same time, NB-IoT technology has low power consumption, wide coverage, and supports a large number of terminal connections to meet a wide range of agricultural environmental monitoring and management needs.

III. OVERALL SYSTEM DESIGN

The system architecture is shown in Figure 1. The sensor node periodically collect surrounding environmental data, and upload it to the IoT (Internet of Things) cloud platform in real time through the NB-IoT network. The cloud server obtains the collected environmental data through the IoT cloud platform interface for data processing, and stores the data in a MySQL database deployed on the cloud server to provide data support for the web application. The web application provides administrators with functions such as user management, node management, node control, real-time monitoring, alarm recording, and historical information query. The web application completes remote real-time monitoring and control of the greenhouse environment, and realizes centralized management of agricultural greenhouses.

In order to make it easier for managers to perform distributed on-site management of the greenhouse, an Android APP is designed for on-site monitoring. Managers can use the smartphone APP to communicate directly with the nodes deployed in the greenhouse via mobile phone Bluetooth. The APP provides real-time plotting of greenhouse environmental data, node device control, and data storage functions. On-site managers can use the APP to complete environmental information query and control management in greenhouses. It also avoids the problem that the webpage cannot be normally accessed for management when a problem occurs on the network.

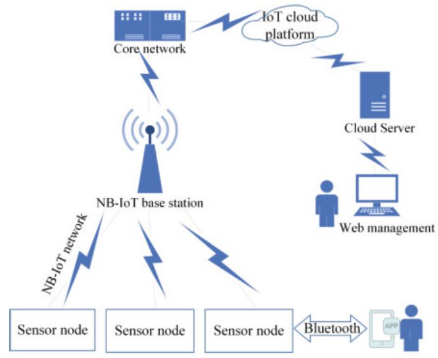

Fig. 1. System architecture

IV. SENSOR NODES DESIGN

In order to improve the density and refinement of terminal nodes, and to achieve reliable communication between sensor nodes and remote control terminals in complex agricultural environments, sensor nodes based on NB-IoT are designed. The block diagram of the node is shown in Figure 2. The node is composed of the microcontroller part of the main control unit, the sensor module, the relay module and the wireless communication module. The microcontroller is responsible for controlling each part of the node, the sensor module is responsible for collecting the environmental data in the greenhouse, the relay module is responsible for executing remote control commands to drive high-power devices, and the wireless communication module is responsible for sending data collected by the sensor module and receiving control commands. The physical picture is shown in Figure 3.

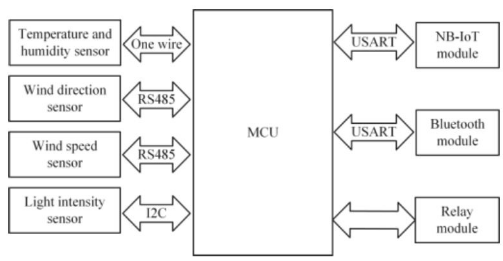

Fig. 2. Block diagram of node structure

A. Microcontroller Module

The microcontroller completes the control and data processing of each part of the terminal node. The microcontroller controls each sensor to collect and process data periodically. After the environmental data is collected, it is processed into a data frame encapsulated into an 8-bit byte array and transmitted in real time through the wireless communication module. When receiving control commands, the microcontroller control relay module drives the high-power device works. In order to perform tasks accurately and efficiently, the microcontroller uses the STM32F103ZET6 chip based on the ARM 32-bit Cortex™-M3 core. The chip has a maximum operating frequency of 72MHz, an operating voltage of 2.0 ~ 3.6V, and an operating temperature between -40 to +105℃. The chip integrates rich peripheral resources, sufficient IO ports and communication interfaces such as USART, I2C, SPI, CAN, USB2.0, SDIO, etc., to provide support for expanding other functions. The chip supports low-power modes such as sleep, shutdown, and standby with high performance, meeting long-term work requirements in complex agricultural environments.

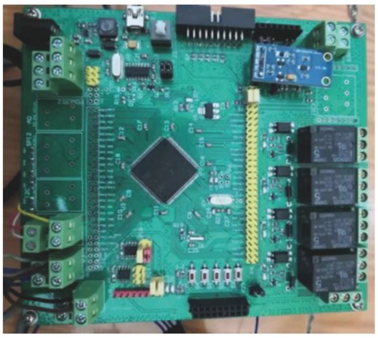

Fig. 3. Physical picture of node

B. Sensor Module

The sensor module completes the data collection of the temperature, humidity, light intensity, wind speed and wind direction of the greenhouse environment. The temperature and humidity collection uses the AM2306 digital temperature and humidity sensor. The AM2306 uses a simple single-wire communication method, and output the internally calibrated digital signal. The temperature measurement range is -40 to +125℃ with an error between 0.3℃. The humidity measurement range is from 0 to 99.9% with an error between 2%. With high reliability and excellent long-term stability, it is often used in outdoor environments to meet the needs of node long-term work. The light intensity sensor uses BH1750FVI and adopts the standard I2C communication protocol, which can detect the light intensity in the range of 1 lx-65535 lx, and has low power consumption. The wind speed and wind direction sensors are SM5385B and SM5384B sensors, both of which are RS485 bus communication and use ModBus-RTU protocol, with high measurement accuracy and strong stability. The SM5385B and SM5384B are also suitable for long-term outdoor work. SP3485 low-power half-duplex RS-485 transceiver is used in the design to implement TTL to RS485 circuit, and realize the communication between the microcontroller and the sensors.

C. Relay Module

In order to realize the control of the water pump and blower in the greenhouse, a 4-way relay module is designed. When the terminal node receives the control command from the remote control terminal, the microcontroller outputs a TTL level and controls the corresponding high-power device to run through the relay. Considering the influence of electromagnetic interference, an optocoupler module is added between the microcontroller's IO port. And the driving capability of the IO port was improved.

D. Wireless Communication Module

The wireless communication module completes the communication between the sensor node and the remote control terminal, including NB-IoT and Bluetooth.

The NB-IoT module is responsible for transmitting the data collected by sensors to the cloud server through the NB-IoT network and receiving remote control commands sent by the remote control terminal. In order to ensure the stable and reliable communication of the system, the data frame format transmitted is designed. The data frame format for NB-IoT and Bluetooth communication in the system is consistent. The format of the environmental information data frame used for uploading is shown in Table 2. The uploaded data frame is processed by the cloud server. The data frame header and end are set to 0x1a and 0x1d to check whether a complete frame of data has been received. The ID of each node has been set before deployment. IDs of different nodes are different and remain unchanged during operation. The data collected by the sensor are put into the data frame according to high 8 bits and low 8 bits. The data frame is verified by CRC-8. The result is used as the check digit. The control command is sent by the control terminal, and the terminal node receives and processes it. The format of the control command data frame is shown in Table 3. The frame header and end are set to 0x2a and 0x2d to distinguish it from the environmental information data frame. The node ID, control device, and control command are set by the control terminal. The data frame also uses the CRC-8 check method.

TABLE II. FRAME FORMAT OF ENVIRONMENT INFORMATION DATA PACKET

Head	Node ID	Length	Temperature		Humidity		Light intensity		Wind direction		Wind speed		Check	End
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O

TABLE III. FRAME FORMAT OF CONTROL COMMAND

Head	ID	Len	Device	Cmd	Check	End
A	B	C	D	E	F	G

The NB-IoT module uses WH-NB75-B5 based on Huawei Boudica chip. The module communicates with the microcontroller through a serial port, works in the 850MHZ frequency band. The module uses a low power design and supports UDP/CoAP network protocols. The system uses the CoAP protocol to transmit data. The CoAP protocol is a communication protocol based on the request/response model. UDP is used as the transport layer protocol of CoAP to reduce communication overhead and make it suitable for Internet of Things applications.

The Bluetooth module is responsible for data transmission between the terminal node and the Bluetooth device such as the smart phone at the greenhouse scene. The Bluetooth module uses the HC-08 wireless Bluetooth communication module based on the CC2540 chip. The microcontroller communicates with the Bluetooth module through the serial port. HC-08 supports the Bluetooth 4.0 low-power communication protocol, the working voltage is 3.2 ~ 6V, and the sleep current is as low as 0.4uA in the low-power mode to meet the needs of long-term work. The module has a communication distance of 80m in an open environment.

E. Node Software Design

Node software workflow chart is shown in Figure 4. The software program of the node is written through Keil uVision5 software, and the programming language is C.

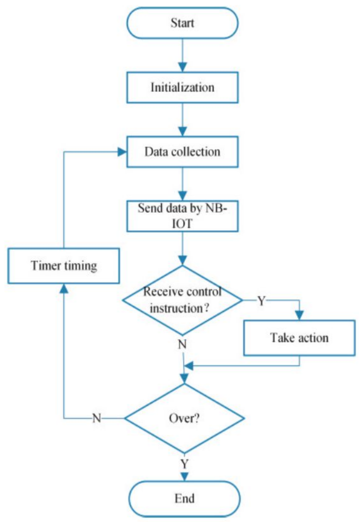

Fig. 4. Sensor node workflow

The node first performs initialization operations, including initialization of variables, STM32's GPIO port initialization, serial port initialization, interrupt initialization, NB-IoT module initialization, Bluetooth module initialization, and each sensor initialization. After the initialization of each part is completed, the main flow of the cycle is entered. Each sensor periodically collects the surrounding environment information. The microcontroller processes and encapsulates the data into a data frame after the collection is completed. The processed data frames is uploaded to the cloud server through the NB-IoT module. And the sensor node receives control commands from the remote cloud server or a mobile phone APP. By default, no data is sent to the Bluetooth module. When the node receives the corresponding control commands, it will send monitoring data to the Bluetooth device. After each data transmission, the timer is timed and then the node repeats the collection and transmission process. After receiving the control command, the NB-IoT module or Bluetooth module will trigger the interrupt of the microcontroller. According to the control command data frame defined in Table 3, the microcontroller processes the received data frame. The microcontroller determines the integrity of the data frame according to the header and tail of the received data frame, and then performs a CRC-8 check calculation on each bit of the data frame to obtain a check code for confirmation. Finally, the microcontroller performs the corresponding control operations according to the control device and control commands obtained from the data frame.

V. CLOUD PLATFORM FOR NB-IOT

In order to avoid compatibility problems caused by inconsistent user platforms, the cloud platform adopts the B/S architecture. The cloud platform uses the web page as the administrator's front-end operation page. Using a browser to access the web page can effectively avoid platform compatibility issues and the hassle of downloading client software. The cloud platform includes 3 parts: data storage, web application and server back-end.

A. Database Storage

The cloud platform database of the management system uses the MySQL relational data management system for storage management. MySQL has the advantages of small size, fast speed, and low cost, which can meet the management requirements of the system database. The data to be stored in the system is mainly divided into 4 parts: administrator user information, terminal node information, node collection environment information, and alarm logs. In fact, the administrator user information, terminal node information, and alarm log data are relatively less, so the system use a single table to store them. The environmental information collected by nodes is the most primary part of the system. In order to avoid the degradation of database query performance when deploying a large number of nodes, the environmental data collected by the nodes are stored in different tables according to the different node ID.

B. Web Application

The development of the cloud platform adopts a front-end and back-end separation method. The web application provides remote web services for the agricultural greenhouse manager. The user interacts with the web page to send an HTTP request to the server. The web page displays the response data after receiving the response returned by the server. The front-end uses JavaScript and HTML language, and adopts on the Vue.js framework and the Element UI component library to develop the front-end pages. It is deployed on the cloud server using Nginx, which is lightweight and highly concurrent.

The web interface is shown in Figure 5. After the user successfully logs in at the login page, the web page is redirected to the management page. There are 7 tab pages in the management page. Among them, the user management tab provides users with services to modify passwords and user names; the node management tab is used to view and edit the deployed terminal node information and mark the location of each node on the map; the node control tab is used to the single node or all nodes send control commands; the real-time monitoring tab is used to view the real-time environmental information collected by each node and draw a line chart in real time; the historical information query tab is used to query the historical data and supports downloading in excel file format; the alarm records tab is used to query the historical alarm records of each node; the help tab introduces the use of the management page.

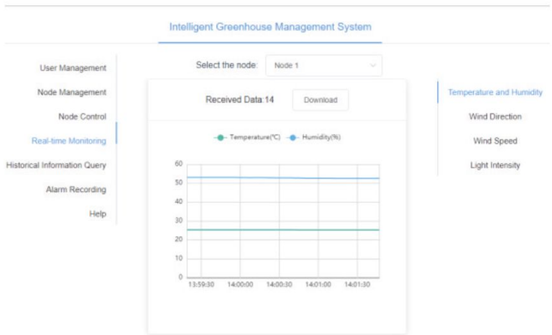

Fig. 5. Web interface

C. Server Back-end

The server back-end is responsible for data transmission, data processing and data storage of the entire system. The server needs to receive and process the data packets sent by the sensor nodes and HTTP requests from the web page. The server needs to be designed with long-term stability in mind. The server program uses a mature and stable Java language and integrates Spring Boot, Spring MVC and Spring Data JPA frameworks to simplify development. After the server program is deployed, the server will begin to receive data packets sent by the sensor node and HTTP requests sent by the web front end. The server will begin to process the data packet and request through the corresponding method. After confirming that the data packet is correct, obtain the node ID and the data collected by the sensor. At the same time, the reception time of the data packet is recorded and the data packet is stored. If outliers are found, an alert record is generated and stored in the log.

VI. ANDROID APP DESIGN

The Android APP is used to perform distributed on-site management of greenhouses. It consists of three parts: data management, Bluetooth management and real-time monitoring.

A. Data Management

The data management part completes the query and management of the data stored in the device, and it is implemented using a lightweight database SQLite attached to Android. Considering that all data is already stored in the database of the cloud platform, all historical data can be queried by the user when the user's device is connected to the network. Therefore, Android devices only store data for three days, while the environmental data collected by the sensor node is stored in a single table in SQLite. The APP uses the Android SDK to complete database creation and SQL statement encapsulation, and implement related methods for creating tables, querying, inserting, updating, and deleting tables. The structure of the database table is the same as that of the cloud platform database table.

B. Bluetooth Management

The Bluetooth management part completes the search and connection of nearby nodes, it is implemented by the Bluetooth function in the Android SDK. After entering the Bluetooth management page, the APP gets the Bluetooth object of this machine and check whether the Bluetooth function is turned on. If it is not turned on, the APP will start the Bluetooth function of the device. In the callback function of scanning the Bluetooth device, the device is added to the list of scanned Bluetooth devices to make the page update in real time when the Bluetooth device is scanned. The click event of the Bluetooth device list component is set to call the Bluetooth connection method to connect to the specified Bluetooth device. After the device is successfully connected to the sensor node, the APP completes the communication with the sensor node through Bluetooth read and write methods.

C. Real-time Monitoring

The real-time monitoring part completes the real-time monitoring of the surrounding environment of the connected terminal node. According to the communication protocol specified in the system, the APP sends control commands to the node and receives data packets sent by the node, and then draws a line chart in real time. After the data is received by Bluetooth, the application stores the data using a byte array and confirms the integrity of the received packet by judging the frame head and frame end. After receiving the complete data packet, the APP will start a new processing thread to process the data, and then obtain the node ID, temperature, humidity, light, wind direction, wind speed, and light intensity. The APP stores the environment information after recording the current time. Because the page in the Android APP cannot be updated in the child thread, the data will be passed to the main thread through Handler, and the line chart will be redrawn.

VII. TESTING

The system was tested in a greenhouse in Songjiang District, Shanghai. Each sensor node collects the environmental data of the greenhouse in real time and uploads it to the cloud server through the NB-IoT network. Users can remotely access the web application to manage the greenhouse through the NB-IoT network. Users can also use the Android APP on site to manage it via Bluetooth. The test environment includes 3 sensor nodes, a PC and an Android phone. The data collection interval of the sensor node is set to 10s Web interface.

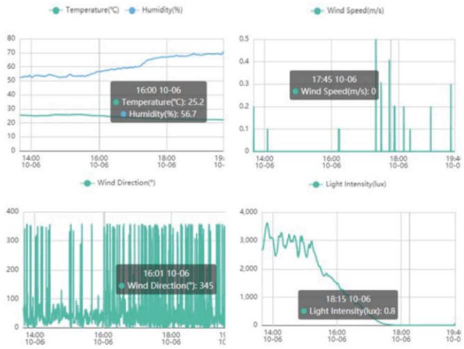

Fig. 6. History environmental information of the greenhouse

The system takes the packet transmission success rate ρ as the evaluation criterion of the communication stability, ρ can be obtained from formula (1):

$$\rho = \frac{p_s}{p_r} \times 100\% \tag{1}$$

Among them, P_r indicates the number of data packets successfully received during the sensor node sending process, and P_s indicates the number of data packets sent by the sensor node.

After deploying the cloud platform and sensor nodes, we tested the web application and Android APP through PC and mobile phone. The greenhouse historical environment information collected by a node is shown in Figure 6. During the testing, each node sent an average of 4032 data packets to the cloud server, and an average of 4 data packets were lost during the transmission. The packet transmission success rate ρ calculated by formula (1) is 99.9008%. We used an Android phone with the APP installed for testing, it can successfully communicate with sensor nodes within 50 meters in barrier-free places. During the testing, each node sent an average of 1015 data packets to the mobile phone Bluetooth, and lost an average of 3 data packets with a packet transmission success rate of 99.7044%, which meets the requirements for stable transmission.

VIII. Conclusion

This paper introduces an intelligent management system for greenhouses based on the NB-IoT network and smartphone. The specific content includes the design of the sensor node, cloud platform, and Android APP. The environment information are collected by the sensor node and uploaded in real time, and the remote centralized management and on-site distributed management are implemented through the web application and Android APP, respectively, to realize the fine perception and control of agricultural greenhouses. The sensor node integrates the NB-IoT module, which implements data upload and down through the NB-IoT network, and effectively reduces communication power consumption. System transmission is stable. The data packet successfully transmitted rate from the node to the cloud server and the smartphone APP is 99.9008% and 99.7044%, respectively. The use of system application is also convenient.

IX. Acknowledgment

This work is supported by Langfang Research and Development Program about Science & Technology (Grant No. 2019011010), Qinhuangdao Science and Technology Research and Development Program (Grant No. 201805A016), Key Accident Prevention Technology Project for State Administration of Work Safety (Grant No. zhishu-0016-2017AQ), China National Coal Association Science and Technology Research Guiding Program (Grant No. MTKJ2017-311) and Hebei IoT Monitoring Engineering Technology Research Center (Grant No. 3142018055). And special thanks to Professor Xueqin Jiang of Donghua University, Dr. Han Hai of Donghua University for their useful discussion and encouragement.

References

[1] X. Wan, T. Zheng, J. Cui, F. Zhang, Z. Ma, and Y. Yang, "Near Field Communication-based Agricultural Management Service Systems for Family Farms," Sensors, Vol. 19, No. 20, pp. 4406, 2019.

[2] X. Shi, X. An, Q. Zhao, H. Liu, L. Xia, X. Sun, and Y. Guo, "State-of-the-art internet of things in protected agriculture," Sensors, Vol. 19, No.8, pp. 1833, 2019.

[3] Z. F. Sun, H. T. Cao, H. L. Li, K. M. Du, Y. C. Wang, and X. F. Su, "GPRS and WEB based data acquisition system for greenhouse environment," Transactions of the CSAE, Vol. 22, No. 6, pp. 131-135, 2006.(In Chinese)

[4] X. Li, X. Cheng, K.Yan, and P. Gong, "A monitoring system for vegetable greenhouses based on a wireless sensor network," Sensors, Vol. 10, No.10, pp. 8963-8980, 2010.

[5] Y. Zhou, X. Yang, X. Guo, M. Zhou, and L. Wang, "A design of greenhouse monitoring & control system based on ZigBee wireless sensor network," 2007 International Conference on Wireless Communications, Networking and Mobile Computing. IEEE, pp. 2563-2567, 2007.

[6] D. H. Park, B. J. Kang, K. R. Cho, C. S. Shin, S. E. Cho, J. W. Park, and W. M. Yang, "A study on greenhouse automatic control system based on wireless sensor network," Wireless Personal Communications, Vol. 56, No. 1, pp. 117-130, 2011.

[7] M. Chen, Y. Miao , Y. Hao, and K. Hwang, "Narrow band internet of things," IEEE access, Vol. 5, pp. 20557-20577, 2017.

[8] Y. P. E. Wang, X. Lin, A. Adhikary, A. Grovlen, Y. Sui, Y. Blankenship, ... and H. S. Razaghi, "A primer on 3GPP narrowband Internet of Things," IEEE communications magazine, Vol. 55, No. 3, pp. 117-123, 2017.

[9] J. Chen, K. Hu, Q. Wang, Y. Sun, Z. Shi, and S. He, "Narrowband internet of things: Implementations and applications," IEEE Internet of Things Journal, Vol. 4, No. 6, pp. 2306-2314, 2017.