

Research Article

Wireless Machine to Machine Based Mobile Substation Monitoring for District Heating System

Cheng Bo,¹ Cheng Xin,² Zhang Chengwen,¹ Liu Ying,¹ and Chen Junliang¹

¹ State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China

² Sichuan Province Key Laboratory of Geological Nuclear Technology, Chengdu University of Technology, Chengdu, Sichuan 610059, China

Correspondence should be addressed to Cheng Bo; chengbo@bupt.edu.cn

Received 22 January 2014; Revised 23 April 2014; Accepted 10 May 2014; Published 26 June 2014

Academic Editor: Nirvana Meratnia

Copyright © 2014 Cheng Bo et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Central heating is a sophisticated process, and many factors influence the heating loading for boilers; without comprehensively monitoring and analysis, it is difficult to find out potential hazards of substation heating supplying electrical equipment. This paper presents a wireless machine to machine based district heating monitoring solution that uses smart mobile devices in a global network. In particular, focusing on the design and implementation for GPRS-based multiseria data transmission terminals to collect the PLC/OPC data in the substation, machine to machine communication network is used to handle the transmission monitoring data from boiler room to master station via GPRS devices and data center that uses different smart mobile devices in a global network, and we also illustrated the mobile substation monitoring scenarios for district heating system. Finally, we give the conclusions and future works.

1. Introduction

The classic way is to divide district heating system into three parts: heating source, heating network, and heat consumers. Heat from heat sources—for example, heat plants like boilers, industrial excess heat, and geothermal or solar heat—is distributed in the heating network and transferred to the heat consumers. The substations where heat meters are situated are distributed in the heating network as a link between hot-water network and internal heating installations in buildings. Adequate control of all relevant parameters of heat meter in the heating process is one of the most significant means of power consumption optimization. District heating substations are used to adapt high pressure hot water to fit the temperature and pressure conditions required by heat consumer buildings as well as by the systems for the preparation of hot service water in buildings. To control the energy transfer in the district heating substation, some kind of control system is needed. The overall efficiency of district heating could be improved by using new strategies for measurement and control.

District heating is a sophisticated process, and heating supplying should vary as outside temperature and time periods change. If heat output of boiler is not timely adjusted, some heating will be wasted. Central heating electrical equipment requires the continuous monitoring at frequent intervals. There are possibilities of errors at measuring and various stages involved with human workers, and high accident risks are increasingly serious. Without comprehensive monitoring, it is difficult to find out potential hazards of heating supplying electrical equipment [1–8]. Figure 1 shows the central heating electrical boilers and heating exchange station equipments.

With the rapid development of network technology and the promotion of industrial information technology, continuous improvement of industrial information technology and the expanding of its application field, the integration and networking of monitoring system have become an important direction of the central heating industrial areas. In heating substations, there is a need for measurement and supervision of different sizes and parameters, such as temperature of the water in the primary and secondary circle, the amount of heat submitted to heating substations, the amount of



FIGURE 1: Central heating boilers and heating exchange station equipments.

heat consumed by individual users, status of pumps, status of electric motor valves and electromagnetic valves, and distribution of the temperature of initial and recurrent lines of the heating grid, as well as monitoring other parameters and their transmission to the supervisory and management system at the dispatcher center.

There are many applications of different strategies for monitoring and control of district heating system. The Global Positioning System/General Packet Radio Service (GPS/GPRS) system for low power data acquisition using MSP430 Texas Instruments microcontroller for monitoring of the heating substation parameters is proposed [9]. The system is implemented in heating stations for a temperature and pressure monitoring. The distributed district heating substation monitoring system is proposed [10] for supervising the consumption of the customers and the parameters in every substation belonging to a district heating system as well as weather parameters that influence the whole thermal power demand. In every substation from district heating system, data acquisition centers are used for acquiring data from temperature, pressure, and flow sensors and also wireless sensor networks were developed for transferring the information from consumers, weather parameters sensors, and some particular technical measurement points. The upgrading of the district heating system through the wireless sensor networks is proposed [11], which will lead to more effective use of thermal energy, enabling also the provision of quality customer services, as the data concerning the status of the existing networks is available in a timely manner, and in the stated amounts. A pilot project conducted by Riga Heat (the main heating supplier in Riga, Latvia) has allowed gaining a real life experience as to the use of the wireless sensor network system in district in-house heating substations. Also, for the purpose, a control and communication architecture based on wireless sensor/actuator network is proposed [12], and the architecture is implemented through a wireless network of sensors and actuators to control the energy transfer from a district heating distribution network to a building while promoting the potential of Service Oriented Architecture (SOA) towards an easy integration and expansion of the pervasive technology. The intellectualized distributed data acquisition system based on the CAN bus with multichannel and multifunction is proposed [13]. The high accuracy

embedded controller with HMI (Human Machine Interface) which supported by the high resolution LCD and the long-distance communication ability. Fuzzy logic control is implemented and the good performance of the fuzzy control proves that this can be an alternative to the classic control, which can enhance the advantage of the central heating supply system and has a very high practicality value. Also, an alternative approach to the problem of district heating monitoring parameters selection is proposed [14–17]. The wireless technology compared with nonwireless technology has some important benefits, for example the reduced system cost and the easier installation and maintenance.

The communication network is the contact hub of the whole heating monitoring control system, through which heating transfer station, heat source, and monitoring nodes form a unified whole. To achieve centralized monitoring and control of the operating data, communication network must be established which can connect all monitoring nodes. With advances in mobile communication, new opportunities have opened up for the development of district heating systems that remotely monitor signals from the boiler and heating substations. The availability of a new generation of mobile phones has had an important impact on the development of such district heating systems, as they seamlessly integrate with a wide variety of networks (such as 3G, Bluetooth, wireless LAN, WCDMA, and GSM) and thus enable the transmission of recorded signals to staff from a central server located in a safety monitoring center.

The monitoring of heating network parameters may be accomplished with the use of equipment installed in heating substations or network chambers. If measuring points are scattered, the efficient communication between them is critical. The distance between the transmitter and the receiver may vary considerably and may reach even tens of kilometers. Currently, central heating system generally uses the following kinds of centralized communication mode: wired communication and wireless communication. Wired communication mainly contains dial-up communications, fiber optic communication, industrial fieldbus, and so forth. Wireless communication mainly contains satellite communication, wireless Bluetooth communication, microwave communication, and so forth. Because both modes have their advantages and disadvantages, it should be chosen by actual situation and requirements. The main communication modes are 485 bus, Public Switched Telephone Network (PSTN), wireless data radio, and GPRS network. Their advantages and disadvantages are shown in Table 1.

By comparing several means of communication modes, we chose the GPRS network to transmit between from heat transfer station and the master station considering the real situation. GPRS is a technology added to the GSM architecture to allow transmission of packet switched data (PSD). A packet-switched network makes it possible to transmit data as packets where each packet contains a destination address. This mean that all packets do not need to travel the same path and can be routed via different paths as traffic conditions change. The packets can even arrive out of order as each packet has a sequence number. By using the packet switched technology, it is possible to be online virtually all the time

TABLE 1: Comparison of communication modes.

Classify	Mode	Advantage	Disadvantage
Wired	485 bus	Reliable transmission, relatively low, expense	Limited transmission distance, limited network mode
Wired	PSTN	Easy operation, simple connection	Low transmission efficiency, long time spend on establishing channel, poor real time performance of data transmission
Wireless	Radio modem	Wide coverage, no operating cost	Large preequipment cost, poor stability
Wireless	GPRS network	Wide coverage, based on GSM network, no maintenance charge, high resource utilization, high flexibility	Low transmission speed, poor real-time performance

without having to pay large amount of money for merely being online.

Machine to machine (M2M) was originally proposed by Sony Ericsson Mobile Communication several years ago. It represents a new concept necessary for proposing effective solutions to the problem of the communication between people and mobile equipment or machine to machine to achieve the purpose of remote monitoring. For example, the data transmission between machines could be solved by distributed data collection terminals. The collected data from the terminals will be transmitted by wireless transmission and then received by the specialized unified collection modules at the data control center which is responsible for processing the received data and sending control commands to the data terminals. The advantages of M2M technology are as follows: automatically data uploading without manual operation which could improve the efficiency of information processing; centralized data processing and preservation which could realize centralized information management; long retention time of data and high storage security; real-time monitoring and control; wireless transmission avoiding wiring; terminal operating status monitoring which insure the stable operation, and so forth.

This paper describes a wireless M2M district heating monitoring solution that uses mobile devices in a global network. We give below a summary of scientific and innovation aspects.

- (1) The GPS/GPRS is used to develop a wireless data acquisition system. The multiseriial data terminal is used to collect the PLC/OPC data from the central heating substations, which includes the different condition parameters measurement for central heating substation via GPRS wireless communications.
- (2) The GPRS based M2M network is implemented in central heating monitoring applications, which provides reliable communication to transmit the data from the substations to the master station server. The user terminals can directly communicate with the M2M nodes.
- (3) The mobile central heating substation monitoring system is developed for a novel M2M solution based on Android mobile devices. A smartphone is used as monitoring terminal, which has the capacity to

provide central substation monitoring services anywhere.

The rest of the paper is organized as follows. In Section 2, a wireless M2M district heating system architecture is proposed. Section 3 describes the experiments and performance measurement. The conclusions and future work are given in Section 4.

2. Architecture

The overall architecture of a wireless M2M district heating system for the monitoring of a substation's environment and devices state according to the flexible and scalable data center and mobile communication is shown in Figure 2.

This system has three components: GPRS-based multiseriial data transmission terminals, communication network, and data center solution that uses smart mobile devices in a global network. The specific functions are described as follows.

GPRS-based multiseriial data transmission terminals and data acquisition: series information collection devices and on-site sensors realize the parameters collection (such as temperature, pressure, flow, circulation pump status, tank water level status, etc.) and actuator control at heat transfer station with PLC system as the core.

M2M communication network selects different communication networks for different acquisition targets. For example, since the wide spread of the urban thermal stations, the data collected by terminals needs to be sent to the heating network monitoring center through GSM/GPRS wireless network. So the communication between terminals and monitoring center uses DTU communication model, which is a wireless terminal device designed to exchange IP data and serial data through GPRS wireless communication network.

Data center and mobile applications: data center realizes data transmission, data display, storage, query, and statistical printing and uses Android mobile devices to access the data center for the substation monitoring in a global network.

2.1. M2M Devices and Local Sensor Data Acquisition. Concerning GPRS-based multiseriial data transmission terminals and data acquisition layer, there are a lot of sensors connected to programmable logic controller (PLC) to obtain the terminal real time data. So in this part the data need to be read from

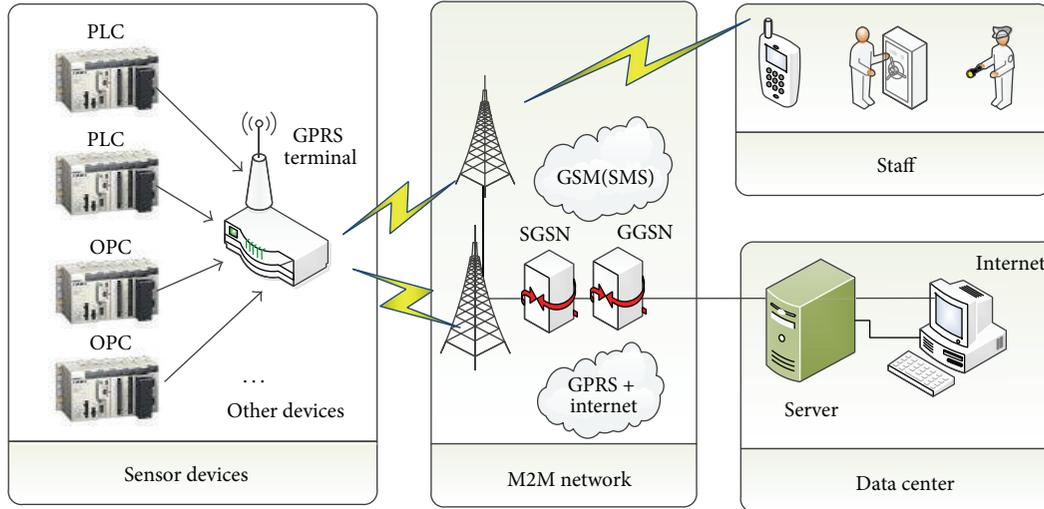


FIGURE 2: System structure.

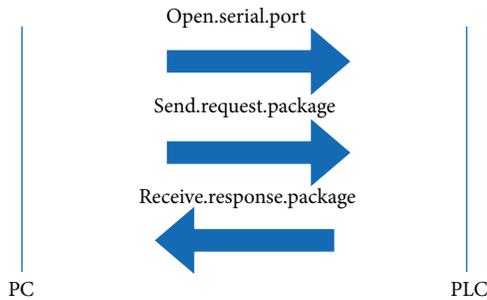


FIGURE 3: Data transmission model.

PLC to PC via COM1 port and stored in the database of PC. Local data collection protocol is used for providing real and useful data such as DELTA DVP Series PLC Communication Protocol.

The task of this part is to read the data from PLC to PC via COM1 port and store the data in the database of PC. The data transmission model is shown in Figure 3.

First, the PC opens the serial port so that PC and PLC could transmit data with each other. Then the PC will send a request package to tell the PLC the address and length of the needed data according to the protocol. Next, the PLC will send a response package back carrying the needed data. The flow chart is shown in Figure 4.

In this part, first there is a need to obtain the serial port parameters: baud rate, flow control in, flow control out, data bits, stop bits, parity, from the database and set the parameter to serial port. Then, to open the serial port to send the request package to PLC. When receiving the request package, PLC will make a response. PLC will send the response package to PC through the serial port. PC will get the response package via reading the serial port. And then PC will interpret the useful data and process the data and then store data in the real time database. The program will perform the tasks in loops to send and receive package again and again. Here, MODBUS

is an application layer messaging protocol, which provides client/server communication between devices connected on different types of buses or networks. The MODBUS protocol allows an easy communication within all types of network architectures. Every type of devices (PLC, HMI, Control Panel, Driver, Motion control, I/O Device, etc.) can use MODBUS protocol to initiate a remote operation. The same communication can be done as well on serial line as on an Ethernet TCP/IP networks. Gateways allow a communication between several types of buses or network using the MODBUS protocol. The MODBUS protocol is shown in Figure 5, which defines a simple protocol data unit (PDU) independent of the underlying communication layers. The mapping of MODBUS protocol on specific buses or network can introduce some additional fields on the application data unit (ADU).

The MODBUS application data unit is built by the client that initiates a MODBUS transaction. The function indicates to the server what kind of action to perform. The MODBUS application protocol establishes the format of a request initiated by a client. Two different serial transmission modes are defined: the RTU mode and the ASCII mode. It defines the bit contents of message fields transmitted serially on the line. It determines how information is packed into the message fields and decoded. When devices are setup to communicate on a MODBUS serial line using American Standard Code for Information Interchange (ASCII) mode, each 8-bit byte in a message is sent as two ASCII characters. This mode is used when the physical communication link or the capabilities of the device does not allow the conformance with RTU mode requirements regarding timers management. The request package format under ASCII mode is shown in Table 2.

The response package format under ASCII mode is shown in Table 3.

Under the ASCII mode, except for STX and END, other data must be in the format of ASCII. Each character should be separated into two characters. For example, "5A" must be

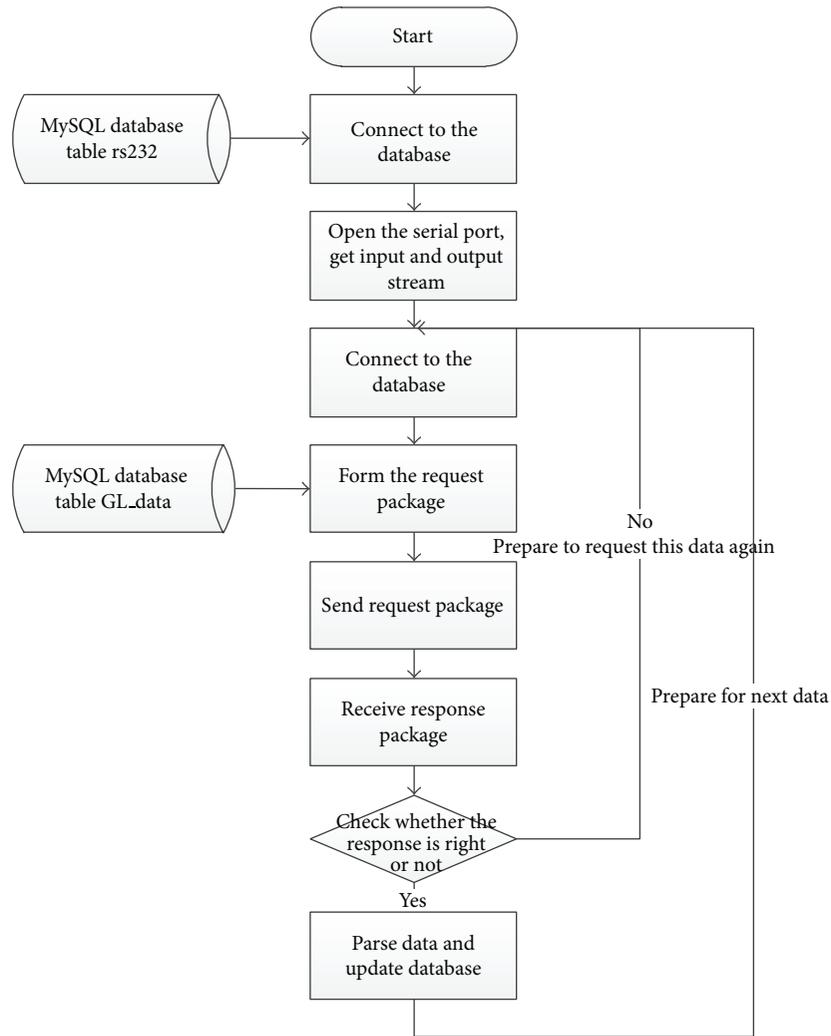


FIGURE 4: General flow chart of local data collection.

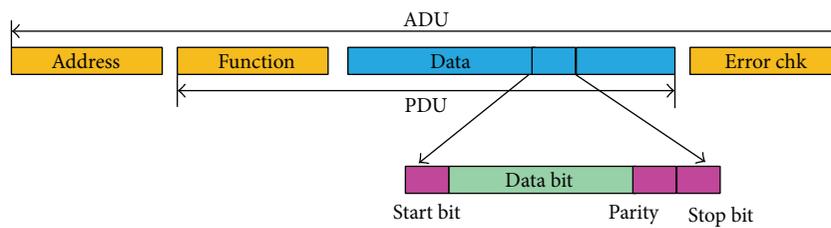


FIGURE 5: Modbus communication format.

TABLE 2

STX	Address	Function	Start address	Address counts	LRC	End
3A	1 byte	1 byte	2 bytes	2 bytes	1 byte	0D 0A

TABLE 3

STX	Address	Function	Data counts	Data	LRC	End
3A	1 byte	1 byte	1 byte	Counts * 2	1 byte	0D 0A

separated into “5” and “A,” and turn them to ASCII code. The ASCII code for “5” is 53 and the ASCII code for “A” is 65. So, “5A” will be 53 and 65.

When devices communicate on a MODBUS serial line using the Remote Terminal Unit (RTU) mode, each 8-bit byte in a message contains two 4-bit hexadecimal characters. The main advantage of this mode is that its greater character

TABLE 4

Address	Function	Start address	Address counts	CRC-16
1 byte	1 byte	2 bytes	2 bytes	2 bytes

TABLE 5

Address	Function	Data counts	Data	CRC-16
1 byte	1 byte	1 byte	Address counts * 2 bytes	2 bytes

TABLE 6

Device_ID	PLC_ID	Sensor_ID	Word_Count
9	1	D0	2

density allows better data throughput than ASCII mode for the same baud rate. Each message must be transmitted in a continuous stream of characters. The request package format under RTU mode is shown in Table 4.

The response package format under RTU mode is shown in Table 5.

There is no start and end characters in RTU mode. Instead, it uses the time interval to determine the end of a package. If the time interval is bigger than 3.5 character time length, it means that the package has been transmitted as shown in Figure 6.

In the RTU mode, it transmits data in character, and no need to transform them into ASCII code. The PLC reads data continuously in accordance with the register address. For example, if I want to collect the data in 0000, 0001, 0010, and 0011, it will collect all the data from 0000 to 0011. So in the response package, I need to pick out the data needed. Additionally, one register address (e.g., one hex form address 08AE) will get two bytes of data response. The user data length in the response package is the double of the address counts.

Here we used RTU mode, for example. After introducing the MODBUS protocol, I will explain how I form the request packet and process the response packet.

To create the request packet, it is needed to determine the table type. The request packet of analog table is different from digital table, and here we use the analog table shown in Table 6.

Here, sensor_id field means the address of register where the data is stored. Word_count field means the number of register unit that the data use. For example, sensor_id = D0 and word_count = 2 mean that the data whose Device_id = 9 and PLC_id = 1 is stored in D0 and D1.

In order to form the analog request packet, first there is a need to query the gl_analog_measure in database to find the sensor_ids and word_counts of the same device_id and sensor_id and then store them in an array. And change the address into DVP communication address in hexadecimal. Because the data is stored in PLC dispersedly, the data is split by comparing the high byte of the address. Start address will be the first address and data count will be the last address minus the first address. Now we can packet the package

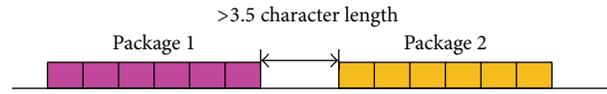


FIGURE 6: RTU mode.

the request format. And the request packet will be sent in circulation.

When receives the response packet, the first thing to do is to check the packet to determine whether it is right or not. If it is not right, it will request this packet again. If it is right, it will parse the data and then update the database. When it parses the data, the first thing to do is to copy the data in start address from the package which contains the data in useless register address. And then query the table address division in database, to check whether it is the record we need. If it is the right one, update the database, add the address according to word count, and check it in database until the package end. If it is not the right one, check the next address. And the response packet will also be received in circulation.

First establish a connection to the H2 database and obtain serial port parameters from database. Get the comport identifier and open the serial port. Open the connection to the serial port so that you can read or write to it. The class diagram is shown in Figure 7.

The class DVP_MODBUS_RTU is about the protocol. The request package and response package are all formed in this class. The C232 class is about the serial port: open the serial port and read/write the serial port. So in this class, it calls the functions SendProcess() and ReceiveProcess() in DVP_MODBUS_RTU class to send request package or receive response package. SetupConnection class is the main class. It calls the functions Send(), Write(), Receive(), and SendBreak() in C232 class.

2.2. M2M Communication Network and Data Transmission.

Data transmission protocol is used for establishing data transmission channel. For example, the serial port protocol is used for providing serial port channel and HongDian H7000 protocol is used for providing socket channel. In the local data collection part, the terminal data has been stored in local database. So in the wireless transmitting part which handles processing and transparent transmission of data, I need to write the data to the GPRS device and make a socket connection between substation and master station. Then send data to master station via the connect connection. The flow chart of this part is shown in Figure 8.

First, there is a need to obtain data from H2 database and set the serial port parameters. Connect the H2 database to obtain the real-time data and then write the data to COM2, so the data could be transmitted using GPRS device wirelessly. The GPRS device needs a SIM card to transmit data. The principle of wireless data transmission is similar to surfing the internet with cellphone. The GPRS device will first register to the base station and then establish a socket connection to the master station using the IP address set in the device. The substation is the socket client. It sends data to the server. The master station is the socket server; it will listen to a port to

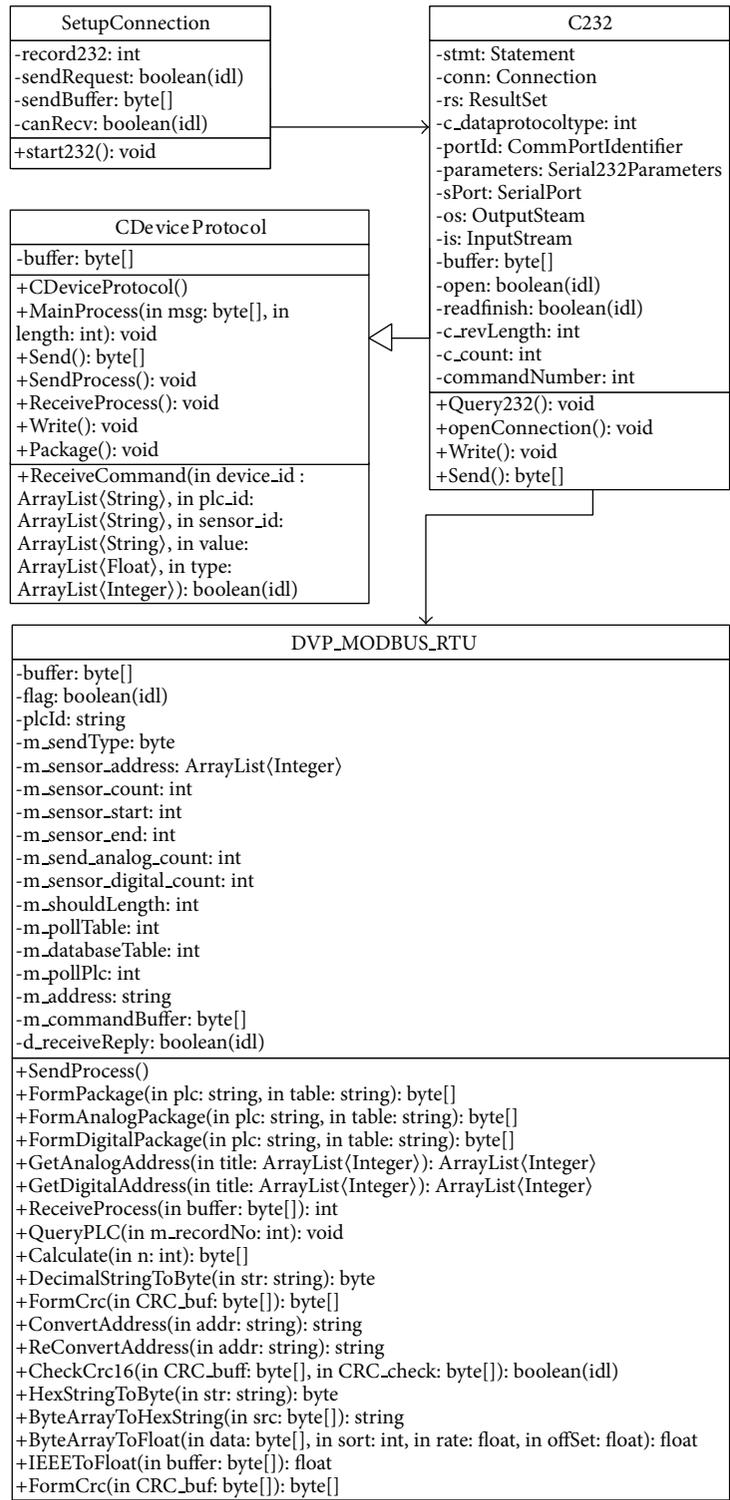


FIGURE 7: General class diagram.

wait the client's data. Multiple threads are used here, so the master station can receive the data from multiple clients. Then the master station will update its database according to the data it received from the substation. The class diagram of this part is shown in Figure 9.

The detailed explanation of each class in the class diagram for *substation* is introduced as follows. Here, the *H2Connection Class* is mainly responsible for open and close connection with the H2 database. When using java to connect the H2 database, you need to add the JAR

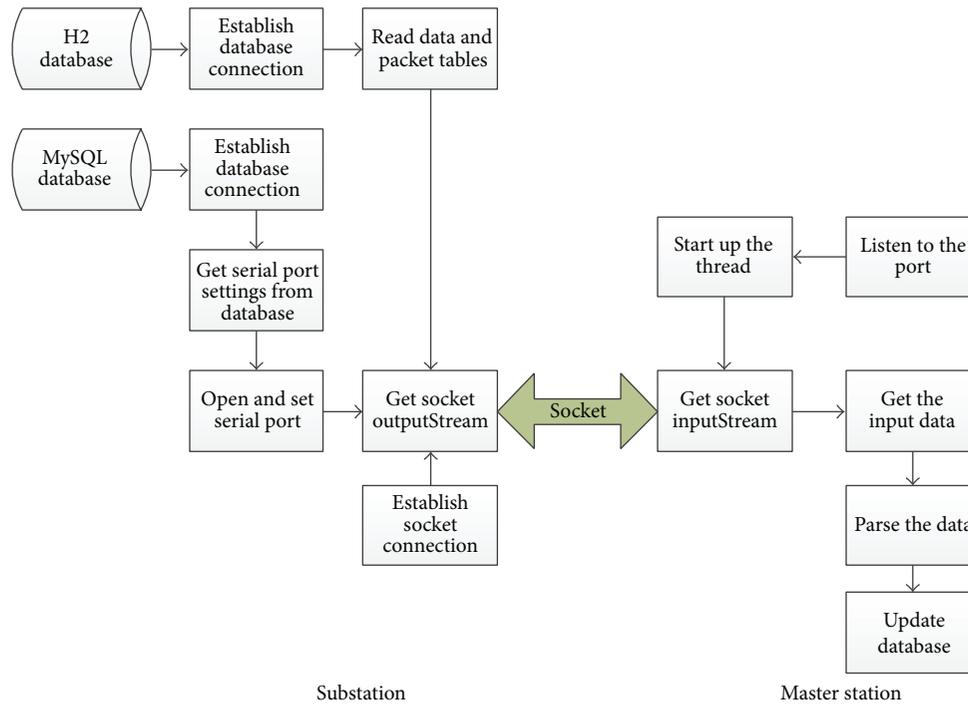


FIGURE 8: General flow chart of wireless transmission.

file-h2.jar-into eclipse referenced library. Because of the H2 console, SQL language can be used in Java code to query the H2 database. To access an in-memory database from another process or from another computer, a TCP server needs to be started in the same process as the in-memory database was created. The other processes then need to access the database over TCP/IP or SSL/TLS, using a database URL such as jdbc:h2:tcp://localhost/mem: test. The C232 Class is about the serial port. It is similar to the C232 class in Object Sensing Layer. The *PackageSender Class* is the response for sending data package wirelessly in the substation. Due to the GPRS device, the maximum bytes in a package are 1024 Byte including 16 Byte headers; the user data can have 252 float (1008 Byte).

First, open the serial port and set its parameter. Then establish the connection with H2 database. Packet the table in H2 database and write them into serial port. The rules of pack table: GL_ANALOG_CONTROL table are used for example in order to explain it clearly. Some parts are shown in Table 7.

Here, define some data structure as `float[] data = new float[252]`, `data[0]`: table name, and `data[1]` means the number of device_id in this table (e.g., in the GL_ANALOG_CONTROL table, there are four device_id, and the value of it is 9). The `data[2]` means the value of the first device_id (e.g., in the GL_ANALOG_CONTROL table, the value of them is 1,4,9,1000). The `data[3]` means the count of this device_id (e.g., there are 13 records whose device_id is 1). The `data[4]`—`data [(int)data[3]+3]` means the value of these records (e.g., `data[4]`—`data [21]` store the value). The `data[(int)data[3]+4]` means the value of the second device_id. The `data[(int)data[3]+5]` means the count of this device_id, and so on. The flow chart is shown in Figure 10.

The detailed explanation of each class in the class diagram for *master station* is introduced as follows. Here, the *ConnectKeeper Class* declares a constructor named *ConnectKeeper* used to keep socket connection and the operation of socket connection. It includes set and get method about device_id, dtu_id, and socket. The *ConnectList Class* declares a constructor about socket connect keeper and thread list. The *WirelessHandler Class* is used to get the input stream, parse data according to the protocol of GPRS DTU, and the rule of pack table and then update the database. Data packet formats of GPRS DTU are described in Table 8.

According to the protocol, first the GPRS device sends a register package from substation to master station whose type flag is 1. Then the master station sends an ACK package back. The master station will check whether the socket is included in the threads pool of *ConnectList*. If not, the socket will be added to the *ConnectList*. Next, the GPRS device sends a data packet to master station whose type flag is 9. Then it will pick out the data from data packet. Then it will parse the data according to the rule that packed it. Because those four tables are similar, and explain only one table: GL_ANALOG_CONTROL. The steps are shown in Figure 11.

The `data[0]` could tell which table that it is belongs. For `data[0] = 2`, it is the GL_ANALOG_CONTROL. `data[1]` could tell me the number of device_id; in this case, there are four device_id. Store the values of all device_id 1, 4, 9, and 1000 in an array `device_id2[]`. Store the value for one device id in an array. Then change it to float array in order to insert them into database. At last, update the database in publish-and-subscribe model.

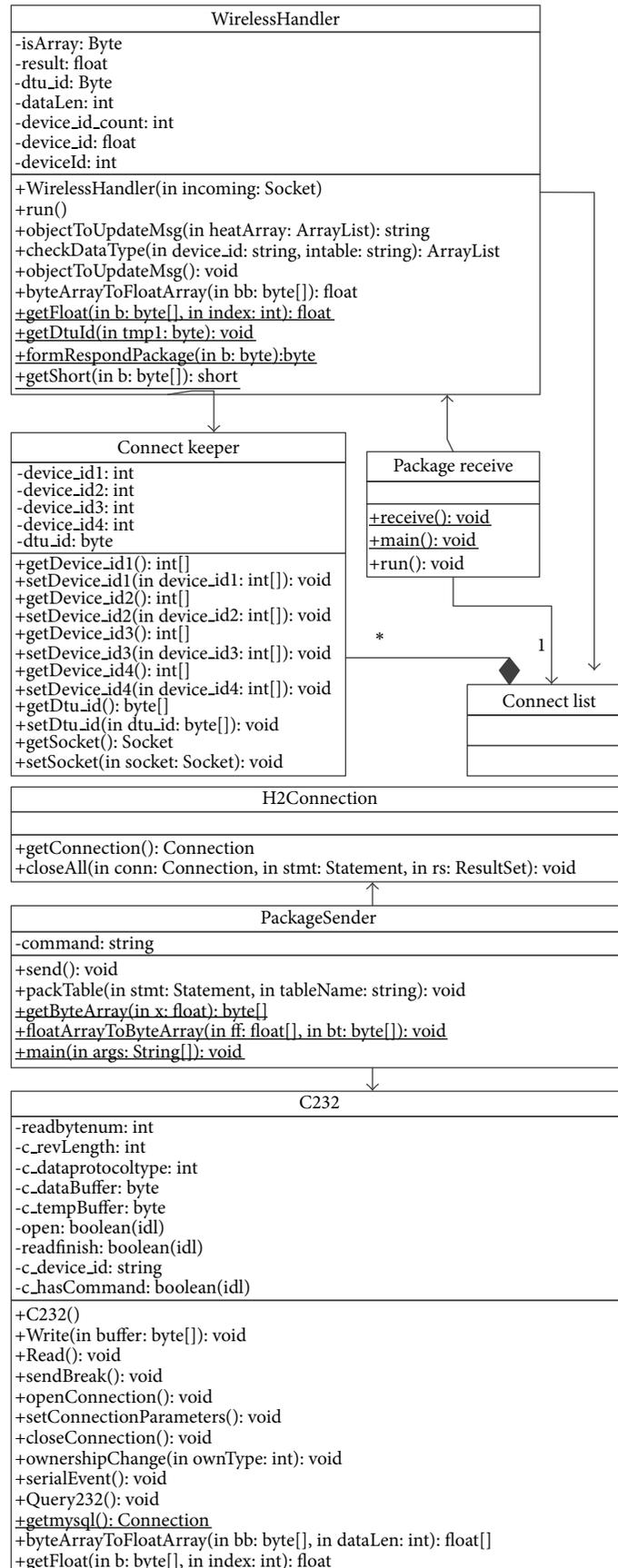


FIGURE 9: General class diagram.

TABLE 7: GL_ANALOG_CONTROL table.

Device_id	Sensor_id	PLC_id	Word_count	Devicetype	Boilerroom	Boiler	Field	Description	Timestamp	Measure_type	Measure_unit	Value
1	D2210	1	2	MODBUS	1	1	QHBCQ	1 climate compensator	1/8/2013	QLI		1
1	D2222	2	2	MODBUS	1	2	QHBCQ	2 climate compensators	1/9/2013	KZMS		1
4	D128	1	1	MODBUS	7	1	RQGL	4 gas boilers	1/10/2013	PFHPLZ	HZ	23.56
4	D3003	4	1	MODBUS	7	4	RQGL	1 gas boiler	1/11/2013	WFZY		27.05
9	D2000	4	2	MODBUS	9	1	QHBCQ	1 climate compensator	1/12/2013	SWWKXZ	C	21.63
9	D2258	41	2	MODBUS	9	1	QHBCQ	1 climate compensator	1/13/2013	ECYCSD	Mpa	21.07
1	D2000	1	2	MODBUS	13	1	QHBCQ	2 climate compensators	1/14/2013	SWWKXZ	C	0

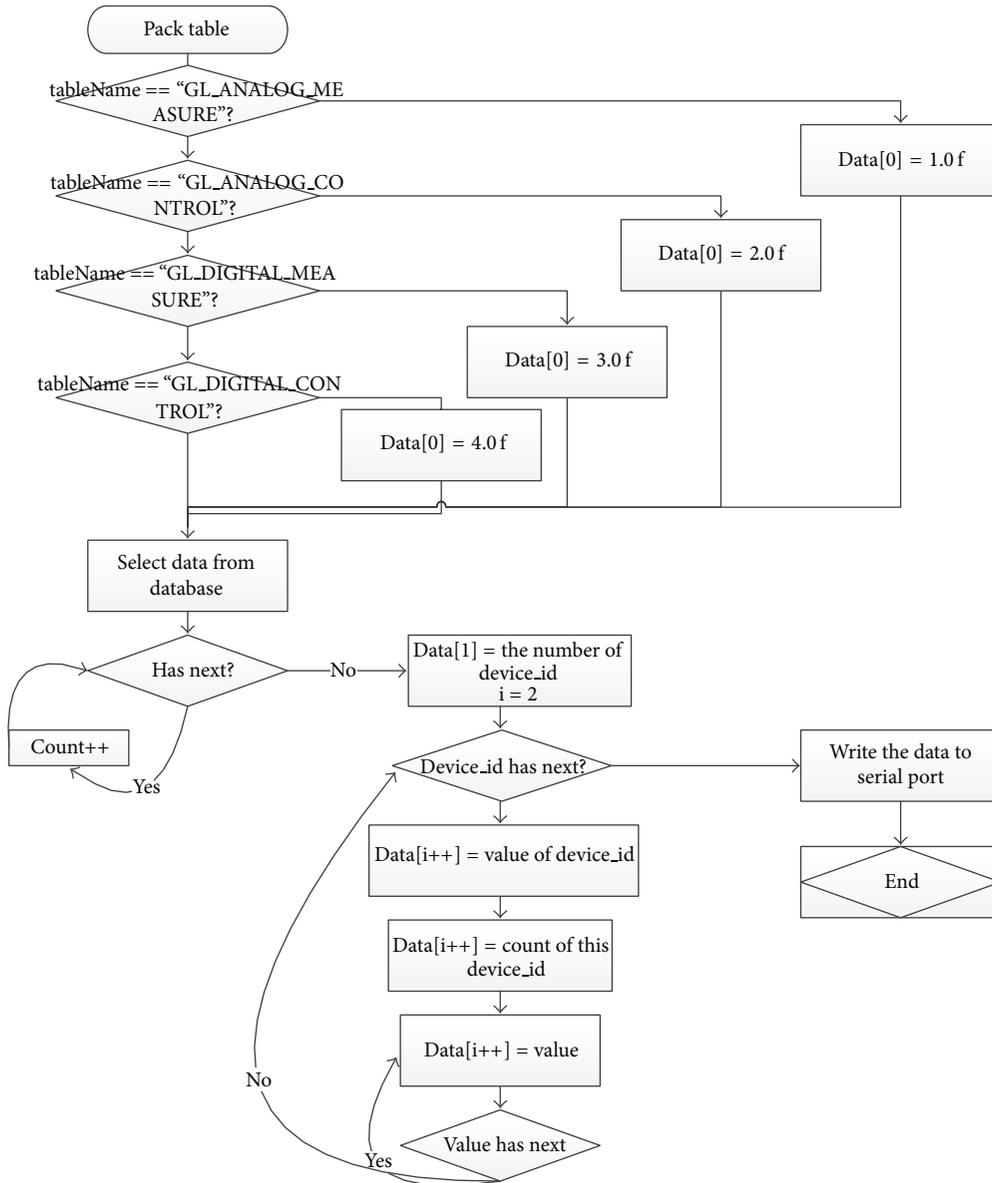


FIGURE 10: Flow chart of pack table.

Finally, the *PackageReceiver Class* is used to receive data at the middleware platform. It is a socket server using multiple threads. The server will listen to the port. So that it can receive data package from different substations. It keeps on running.

2.3. *Data Center and Mobile Application Services.* In the data exchange part, the data has been already transmitted into master station and inserted into database. So in the Application Service part, this data needs to be displayed in a mobile application. Nowadays, there are many kinds of operation systems, such as Android, IOS, and Windows Phone. So it is eager to build the cross-platform application. A good solution to this problem is PhoneGap—a platform could build cross-platform application using standardized web APIs-JavaScript, CSS, and html. An Android application

will be used as an example. First data need to be extracted from the database, converted into an XML files, and then displayed in a table of Android App. Because XML data is expressed in plain textual format, XML is a platform independent language to store data which provides an easier way that data can be shared by different applications. Dom4j is used to build the XML file. Since the real-time data in database is changing all the time, the XML file will be generated for every 10 seconds. The function of creatXmlFile is described in Algorithm 1.

The PhoneGap framework is used to create Android apps using standardized web APIs-JavaScript, CSS, and html. First, there is a need to parse data from the XML file. Here, Dom4j also used to parse the XML file and used one node for example. First, it is important to get the child node of the root node. Next get its child root. And then get the node value.

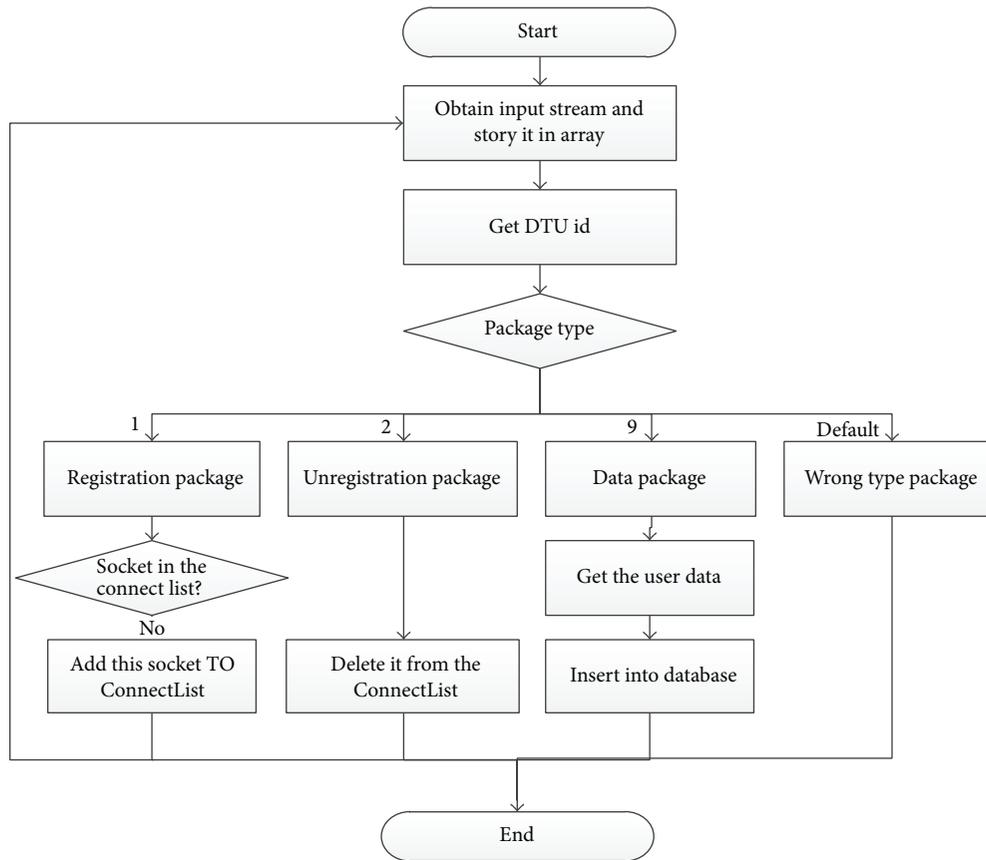


FIGURE 11: Flow chart of WirelessHandler.

```

public void createXmlFile(ResultSet rs, ResultSetMetaData rsmd, File file) {
    XMLWriter xmlWriter = null;
    String elementValue = "";
    Document doc = DocumentHelper.createDocument();
    Element rootElement = doc.addElement("ROWDATA");
    while (rs.next()) {
        Element rowElement = rootElement.addElement("ROW");
        for (int i = 1; i ≤ rsmd.getColumnCount(); i++) {
            Element elementName = rowElement.addElement(rsmd.getColumn(i));
            Object obj = rs.getObject(rsmd.getColumn(i));
            if (obj != null) {elementValue = rs.getObject(rsmd.getColumn(i)).toString().trim();
            } else {elementValue = ""; }
            elementName.setText(elementValue); }
        xmlWriter = new XMLWriter(new FileWriter(file)); xmlWriter.write(doc); }
  
```

ALGORITHM 1

Because the XML file is reloaded every 10 seconds, the HTML file will be refreshed every 10 seconds to show the real-time data as shown in Algorithm 2.

3. Experiments and Performance Measurement

Here, the HongDian H7000 GPRS DTU (Date Transfer Unit) is used for wireless transmission which is a terminal product based on GPRS data communication network. It embedded

many protocols PPP, TCP/IP, and DDP into the GPRS module. The GPRS module needs a SIM card to transmit the data wirelessly. The principle of wireless data transmission is similar to surfing the internet with cellphone. The GPRS device registers to the base station and get an IP address. So, the GPRS module could connect with the server and establish a socket connection. Write the data to COM2 from COM1 of PC and transport the data using GPRS. At the substation, the GPRS M2M device is connected to the PC as shown in Figure 12.

```

function GetXML(){
  var xmlhttp = new XMLHttpRequest();
  xmlhttp.open("POST", "table.xml", false);
  xmlhttp.send();
  document_xml = xmlhttp.responseXML;
  var Rootnode = document_xml.documentElement;
  var FirstNode = Rootnode.firstChild;
  var typeNode1 = FirstNode.firstChild;
  var valueNode1 = typeNode1.nextSibling;
  var unitNode1 = FirstNode.lastChild;
  var type1 = typeNode1.firstChild.nodeValue;
  var value1 = valueNode1.firstChild.nodeValue;
  var unit1 = unitNode1.firstChild.nodeValue;
  document.getElementById("a1").innerHTML = type1;
  document.getElementById("b1").innerHTML = value1;
}

```

ALGORITHM 2

```

***** init completed
***** 232 serial port is running...
***** Query RS232, set the serial port parameters
***** Open serial port
-----
***** No.1 send the request package
***** PLC.ID: 1, Function: 3, Table:GL_anlog_measure
***** Generate CRC-16 check code
***** Write data to serial port: 1 3 16 0 0 18 -63 7
---Disconnection, reconnect after 1 minute---
***** Read the serial port
***** No.1 receive the response package
***** C232 serial port is receiving
***** MODBUS processing
***** Generate CRC-16 check code
***** Response package length: 41, should length:41, through head/end/CRC check,
the content is 1 3 16 0 0 18 -63 7 UP

Device address: 9, PLC address: 1, sensor address: D0
9.1_GSWD
Real value: 0.0

Device address: 9, PLC address: 1, sensor address: D2
9.1_HSWD
Real value: 0.0

update database

```

ALGORITHM 3

The PLC has connected with PC via serial port. Transmit data from PLC to PC. The output information is shown in Algorithm 3.

Transmit data from substation to master station via GPRS device. The output at the master station is shown in Algorithm 4.

Finally, a cross-platform application is built using standardized web APIs-JavaScript, CSS, and html to display the

data in table. Figure 13 shows the value for mobile substation monitoring application scenario.

Figure 14 shows the real-time curve for water temperature. In particular, the x -axis stands for time and the value is updated in every 5 s.

In particular, the industrial fields exchange data with the server using GPRS connection protocol which has the packet loss situation. So, the test for statistics and analysis

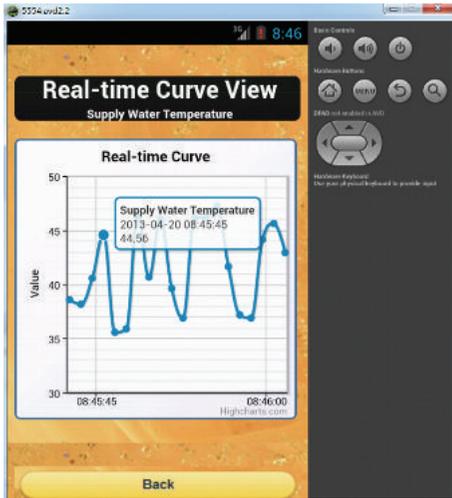


FIGURE 14: "Supply Water Temperature" in real-time curve scenario.

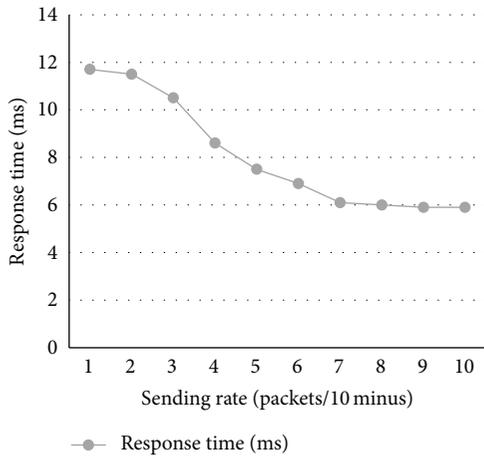


FIGURE 15: Response time experiment.

generated by the communication delay between the server and sensor. In this experiment, 10 GPRS module is deployed to transmit underlying sensor data with a circle of every 30 seconds. We made 10 experiments in total and recorded every delay in a file. The server calculated a delay when it received 2000 data packets. Figure 17 shows the statistical result of the experiment.

It is observed from Figure 17 that the value of the abscissa indicates the number of data in the 10 experiments receiving 2000 data each. Ordinate represents the number of delay with the unit of millisecond. We can find from the figure that the delays of 10 experiments are 9.324691, 8.304323, 7.43123, 8.021342, 8.211280, 8.33452, 7.78127, 9.58279, 8.32645, and 8.78372. The delays are mainly less than 10 millisecond which basically meets the requirements of real-time data transmission.

The fourth experiment was to test the CPU utilization for master station server. When multiple substations connect with master station, and the master station used multiple threads to communicate with those substations. This test

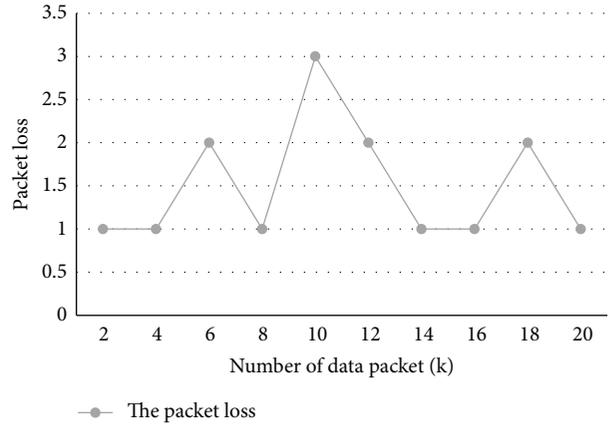


FIGURE 16: Packet loss statistics.

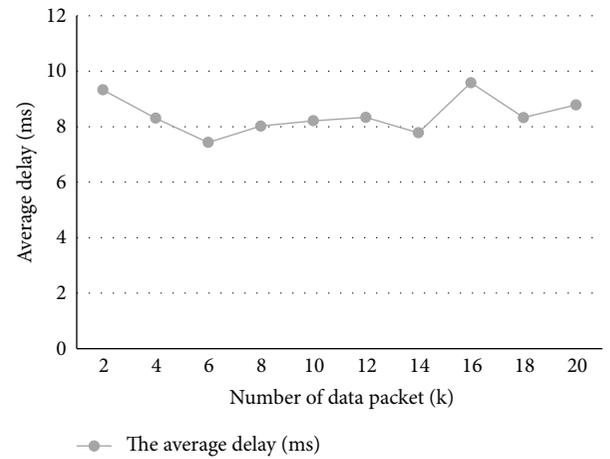


FIGURE 17: Statistical result of delay.

shows the relationship between the number of substation and the CPU utilization. Figure 18 shows the result of CPU utilization.

It is observed from Figure 18 that, when the number of substation increases, the CPU utilization in master station is increasing because of the increasing number of the number of threads. So the number of substation cannot be too large.

4. Conclusion and Future Work

This paper constructs the GPRS based wireless machine to machine remote substation monitoring system for district heating. Focusing on the design and implementation for GPRS-based multiseria data transmission terminals to collect the PLC/OPC data in the substation, machine to machine communication network is used to handle the transmission monitoring data from boiler room to master station via GPRS devices and data center that uses different smart mobile devices in a global network, and we also illustrated the mobile substation monitoring scenarios for district heating system. In the future work, to realize the alarming function of the system. If there is data exceeding the threshold value,

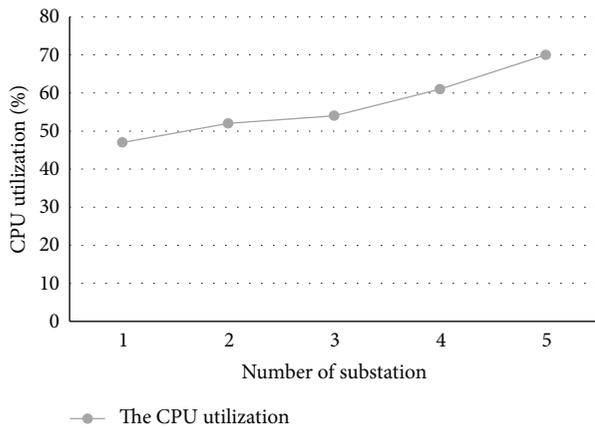


FIGURE 18: The CPU utilization experiment.

the system will warn the principal to ensure the safety of the boiler room. Since there are lots of types of protocols for electrical equipment, the system will be extended to suit more protocols, and the performance should be tested in larger scale deployment environment.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

- [1] G. R. Biswal, R. P. Maheshwari, and M. L. Dewal, "Modeling, control, and monitoring of S³RS-based hydrogen cooling system in thermal power plant," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 1, pp. 562–570, 2012.
- [2] M. D. Cubells-Beltrán, C. Reig, J. Martos, J. Torres, and J. Soret, "Limitations of magnetoresistive current sensors in industrial electronics applications," *International Review of Electrical Engineering*, vol. 6, no. 1, pp. 423–429, 2011.
- [3] L. H. Fang, Z. Z. Wu, R. X. Kang, and L. J. Wei, "Integrated monitoring and management system for heat-supplying process," in *Proceedings of the IEEE International Conference on Information and Automation (ICIA '10)*, pp. 2065–2069, Harbin, China, June 2010.
- [4] M. Kezunovic and T. Popovic, "Substation data integration for automated data analysis systems," in *Proceedings of the IEEE Power Engineering Society General Meeting (PES '07)*, pp. 1–6, Tampa, Fla, USA, June 2007.
- [5] Mastacan, I. Olah, and C. C. Dosoftei, "District heating substations water temperature control based on soft computing technology," in *Proceedings of the 6th International Conference on Electromechanical and Power Systems*, pp. 172–175, Chişinău, Moldova, October 2007.
- [6] S. Haiwen, D. Lin, L. Xiangli, and Z. Yingxin, "Energy-saving potential of seawater-source heat pump district heating system over boiler-house district heating system," in *Proceedings of the International Conference on Energy and Environment Technology (ICEET '09)*, pp. 121–124, Guilin, China, October 2009.
- [7] J. van Deventer, J. Gustafsson, J. Eliasson, J. Delsing, and H. Mäkitavola, "Independence and interdependence of systems in district heating," in *Proceedings of the 4th Annual IEEE Systems Conference (SysCon '10)*, pp. 267–271, San Diego, Calif, USA, April 2010.
- [8] J. Gustafsson, *Distributed wireless control strategies for district heating substations [Licentiate thesis]*, Luleå University of Technology, Luleå, Sweden, 2009.
- [9] A. Peulic, S. Dragicevic, Z. Jovanovic, and R. Krneta, "Flexible GPS/GPRS based system for parameters monitoring in the district heating system," *International Journal of Computers Communications & Control*, vol. 8, no. 1, pp. 105–110, 2013.
- [10] U. Florina, L. R. Gabriel, and P. Daniela, "Distributed monitoring architecture for district heating systems," in *Proceedings of the Annals of DAAAM*, pp. 793–794, 2007.
- [11] A. Zabasta, V. Selmanovs-Pless, N. Kunicina, and L. Ribickis, "Wireless sensor networks for optimisation of district heating," in *Proceedings of the 15th International Power Electronics and Motion Control Conference and Exposition (EPE/PEMC '12)*, pp. 11–15, Novi Sad, Serbia, September 2012.
- [12] N. Kaur, C. S. McLeod, A. Jain et al., "Design and simulation of a SOA-based system of systems for automation in the residential sector," in *Proceedings of the IEEE International Conference on Industrial Technology (ICIT '13)*, pp. 1976–1981, Cape town, South Africa, February 2013.
- [13] H. Sun, C. Niu, T. Liang, Z. Lei, and L. Yun, "An intelligent distributed monitoring and control system for heat supply station," in *Proceedings of the 6th World Congress on Intelligent Control and Automation (WCICA '06)*, pp. 6649–6652, Dalian, China, June 2006.
- [14] J. van Deventer, J. Gustafsson, J. Delsing, and J. Eliasson, "Wireless infrastructure in a district heating substation," in *Proceedings of the 3rd International Systems Conference*, pp. 139–143, Vancouver, Canada, March 2009.
- [15] O. Friman, P. Follo, J. Ahlberg, and S. Sjokvist, "Methods for large-scale monitoring of district heating systems using airborne thermography," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 23, no. 4, pp. 1–8, 2013.
- [16] S. A. Meybodi, M. Dohler, A. N. Askarpour, J. Bendtsen, and J. D. Nielsen, "The feasibility of communication among pumps in a district heating system," *IEEE Antennas and Propagation Magazine*, vol. 55, no. 3, pp. 118–134, 2013.
- [17] P. Malinowski and P. Ziembicki, "Analysis of district heating network monitoring by neural networks classification," *Journal of Civil Engineering and Management*, vol. 12, no. 1, pp. 21–28, 2006.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

