 doi:10.20944/preprints201703.0068.v1

Article

Research and Application of SCADA System for the Microgrid

Shuangshuang Li [1], Baochen Jiang [1,*], Xiaoli Wang [1] and Lubei Dong [1]

[1] School of Mechanical, Electrical and Information Engineering, Shandong University, Weihai 264209, P. R. China; lishuangshuang_sdu@163.com

* Correspondence: jbc@sdu.edu.cn; Tel.: +86-186-6935-5366

Abstract: The effective Supervisory Control and Data Acquisition (SCADA) system can improve the reliability, safety and economic benefits of microgrid operation. In this research, the lower central controller and upper WEB monitoring system are connected by the SCADA system which is as the hub of microgrid intelligent monitoring platform. This system contains a set of specific functions programmed by Java as a middleware which can provide communication and control function between the central controller and the upper monitoring system. For the sake of the security and stability of the microgrid, the SCADA system realizes the business processing about real-time data acquisition and storage, the load balancing and resource recovery, concurrent security processing, parsing and transmitting the control instructions. All that functions have been tested and verified in the actual operation.

Keywords: Microgrid; SCADA; Java; Middleware

1. Introduction

With the energy problem and environment problem being increasingly prominent, the exploitation of clean renewable energy resources has become the world's economic and social important strategy of sustainable development. In order to discover the advantages of distributed generation in the economic, energy and environment, the microgrid technology using a large amount of renewable energy gains popularity and develops rapidly. A microgrid can be defined as a power supply system which contains Distributed Generation (DG), energy storage and control systems that can work together to supply the local loads [1, 2].

In microgrid, Supervisory Control and Data Acquisition (SCADA) [3] system is used for data acquisition, monitoring control and procedure control for the spot devices, which is a production procedure control and dispatching automation system based on computer [4, 5]. It can realize data acquisition, equipment control, measurement, parameters adjustment and all kinds of signal alarm, which plays an important role in improving the reliability, safety and economic benefits of power grid operation. Besides, it can reduce the burden of the dispatchers, realize the electric power dispatching automation and modernization, and improve the efficiency and the level of dispatch [6].

The SCADA system is usually divided into the hardware part and the software part, and it has been designed and applied in many fields. Paper [7] introduced the design and implementation of hardware components in microgrid SCADA system. Paper [8] presented the communication and PC interface of microgrid SCADA. And paper [9] presented a SCADA system using IEEE 802.22 standard to overcome all existing limitations. Besides, paper [10] used SCADA system to provide real-time control of power switching relays, and obtain information about their status and perform three-phase measurement. What's more, the primary aim of the paper [11] is to propose a suitable private hybrid cloud based SCADA architecture satisfying various necessities in the framework of

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 March 2017 doi:10.20944/preprints201703.0068.v1

interoperability of micro-grid platforms while maintaining security restriction conditions. Different from the above, in this research, we design a SCADA system programmed by Java to be as the middleware in microgrid intelligent monitoring platform innovatively. The lower central controller and upper WEB monitoring system are connected by the SCADA system which is as the hub of microgrid intelligent monitoring platform.

The remainder of the paper is organized in four sections. In Section 2, the architecture and main processing tasks of microgrid SCADA system are presented. Section 3 chooses three most important parts to describe the design of SCADA system in detail and Section 4 presents the implementation and operation test. The conclusions are drawn in Section 5.

2. The Architecture of SCADA System in the Microgrid

In this actual system, microgrid mainly contains a microgrid hardware system and a microgrid intelligent monitoring platform. This paper mainly introduces the SCADA system which is one of the middlewares in intelligent monitoring platform. The architecture of SCADA system in the microgrid is shown in Figure 1. The module of UART Ethernet receives the lower central controller's real-time data from the serial port, and then the data will be encapsulated and sent to SCADA system through the Ethernet at last. The SCADA system is mainly utilized for parsing the microgrid packaged data, and then the parsed data will be stored in the MySQL database. Besides, the SCADA system is used for communicating with the upper WEB monitoring. After receiving the upper client control instructions, the system will parse the instructions and send the parsed instructions to lower central controllers by the module of UART Ethernet. The main processing tasks of SCADA are system startup processing, clients connection processing, business processing, load balancing processing, resource recovery processing and concurrent security processing. Section 3 chooses three most important parts to describe the design of SCADA system in detail.

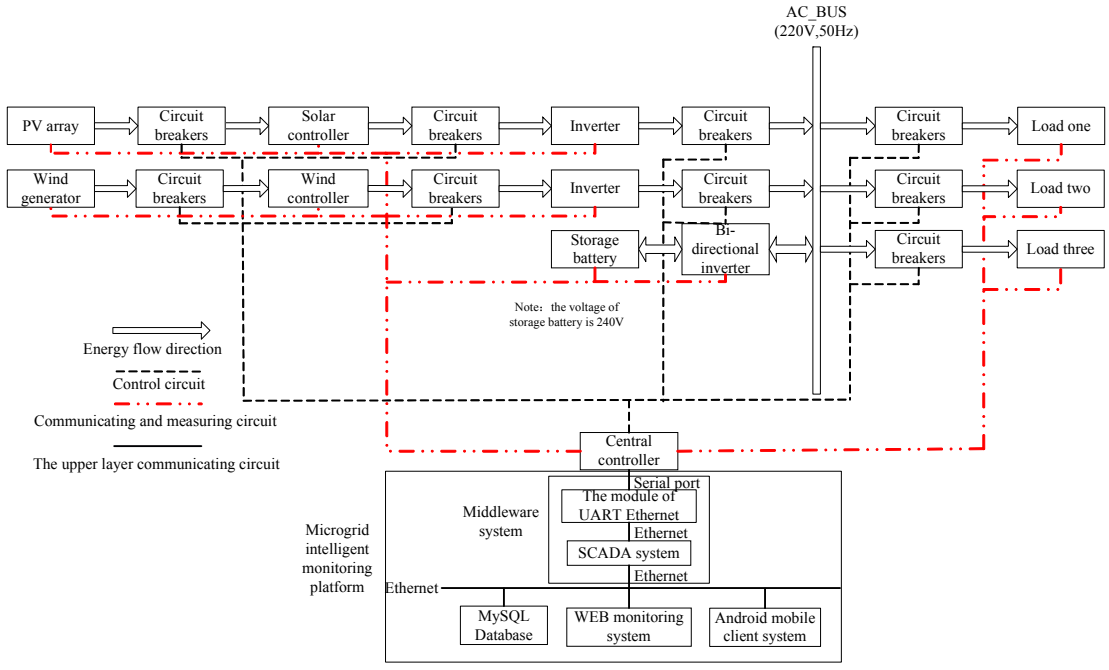

Figure 1. The Architecture of SCADA System in the Microgrid

3. Design of Microgrid SCADA System

3.1. Business processing in SCADA system

The SCADA system invokes the "start" method of Thread class to start the thread. The "start" method invocation will invoke asynchronously the "run" method of this Thread class, and the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 March 2017 doi:10.20944/preprints201703.0068.v1

business processing is in the "run" method. The system receives the Modbus/TCP protocol data frames by the IO stream in the "run" method. After the data frames have been parsed and verified, it will be classified according to the control code field in the protocol data frames. The data will be processed in five different ways according the results of the classification: electrical message business processing, alarm message business processing, message business processing of clients requesting the system to check the time, message business processing of upper monitoring system sending control instructions, message business processing of upper monitoring system giving power generation plan instructions. Finally the processed data is inserted into the corresponding MySQL database tables, which will be displayed in the web pages in WEB monitoring system. Business processing flow of SCADA system is shown in Figure 2.

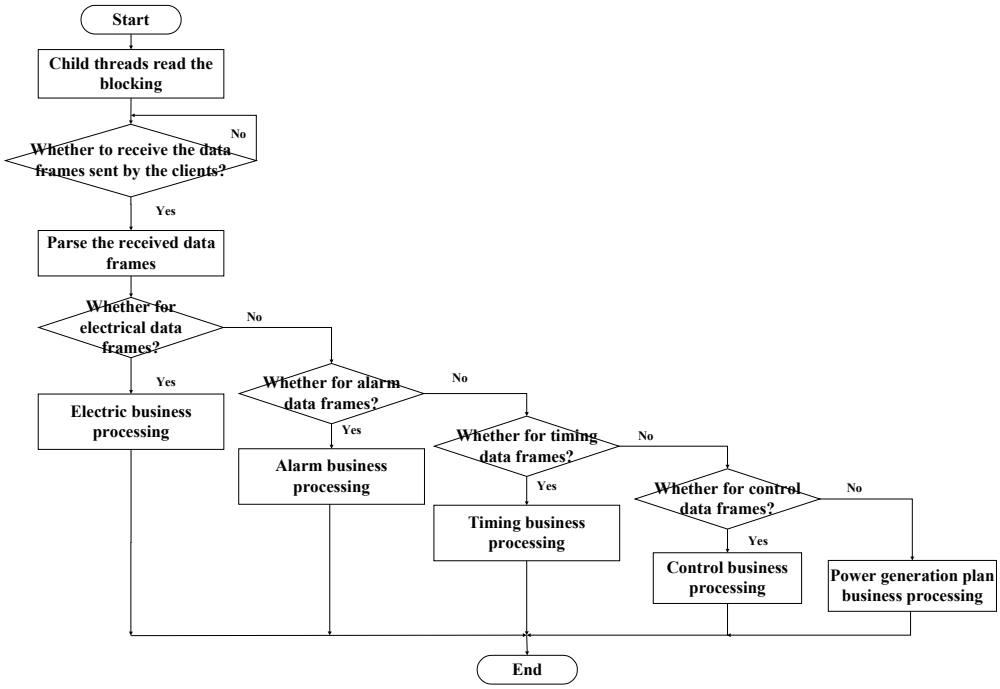

Figure 2. The Business Processing Flow in Microgrid SCADA System

For the information of temperature, wind speed, light intensity required by the microgrid, the system will start the web crawler service program to crawl the weather information from the observatory in this school which is very close to the microgrid system. Then the system will parse the weather information to generate the required weather information field and store it into the database.

The system inserts the aggregated electrical data into the microgrid electrical table. Because the SCADA system gathers the microgrid electrical data every five seconds, so there will be 17280 items one day. With the accumulation of time, the data volume will become huge. The query access to the microgrid electrical data is still very slow even creating index used for the microgrid electrical data table. This paper uses the sub-table mechanism to solve the problem above, which dynamically generates a new table every day. When we insert data into the microgrid electrical database table, the system will judge whether there is a current date microgrid electrical table using "Create Table If Not Exists" statement first. If the table exists, the system inserts the data into it, while if the table does not exist, the system will create this table and insert the data into it. Every time we get access to tables, even simple database connection may affect efficiency. This paper uses cache mechanism to solve the problem. The cache mechanism identifies the current date microgrid electrical table is that existent or nonexistent through the map cache of the current microgrid electrical table. And it combines with the "Create Table If Not Exists" statement to solve the cache empty problem when

 doi:10.20944/preprints201703.0068.v1

the system restarts, which ensures the microgrid electrical data acquisition is stable in the SCADA system.

3.2. Load balancing processing in SCADA system

Load balancing processing is mainly aimed to solve the problem of unbalanced loads in SCADA system. The SCADA system needs to use the multithreading technology to deal with the business requests which are sent from multiple central controllers. From section 3.1 after business processing, the processed data will be inserted into the MySQL database at the end. If the SCADA system supports more than one hundred central controllers to send the data to it, a "hot spots" problem will occur. That problem is that the MySQL database connections will be exhausted when all central controllers request MySQL connections to the SCADA system. So the unsuccessful abnormal links will appear. And with the increase of the number of the central controllers, the probability of this event occurrence will be higher. Although we can modify the value of "max_connections" in the mysql.ini file to increase the maximum number of MySQL connections to mitigate this problem, the value of the maximum number is different under the different operating systems. What's more, the number has certain limitation, so it can't solve the problem fundamentally.

The producer-consumer model is used to solve this problem. When SCADA system starts, it will start a custom "handler" thread to handle the business alone, and the "handler" thread inherits the Thread class and implements the "run" method. The "handler" thread can be started by invoking its "start" method. There is a double linked list in the handler thread, and elements in the double linked list are custom "DisposeBean" class objects, which have two parameters: the "sql" statement and "paras" arrays. The "sql" and "paras" arrays are respectively on behalf of "sql" statement and parameters when we create the tables or insert data into tables. Lock the double linked list to guarantee the safety of the thread. Besides, in order to prevent repeated switching of system context, the optimistic locking mechanism is used to achieve the locking operation of double linked list. The optimistic locking mechanism is to complete the locking operation by CAS mechanism of CPU that can lock instructions stream. Besides, the double linked list is used to realize a blocking queue. The "await"and "signal" operation can be used to set up the observer model on the queue. That is to say, if the producer finds that the queue capacity reaches the maximum value at one end of the queue, it will call await method to block until the consumer calls signal method to awaken the producer to stop blocking at the other end of the queue.

Through the above operation, the SCADA system can put the operation of obtaining connection from the MySQL database in the "handler" thread to manage. There is only generating a custom "DisposeBean" class object which will insert into the blocking queue without the requirement of obtaining connection from the MySQL database. So, this handling can resolve the "hot spots" problem, and make the system's loads balanced. Besides it may support business requests from the thousands of central controllers.

3.3. Concurrent Security Processing in SCADA System

Concurrent technology is mainly used for solving the problem of concurrent communication, and whether communication is based on what kind of communication technology, and it is essentially point-to-point communication technologies when the client communicates with the server. In order to support the communication between server and the client by the way of many-to-many, concurrent technology has to be used.

Concurrent security processing in SCADA system is mainly embodied in the thread resources competition. Threads are stored in the threadPools, and they are marked in unused state when they are initialized. Because SCADA system uses BIO model that is "one per one thread". SCADA system is as a server. There will be multiple clients to apply simultaneously available threads resources from threadPools, and there will be multiple clients to select simultaneously the same thread and to bound its own connection "socket" separately when the concurrent occurs, which may result in the loss of the connection "socket" to update. To solve these problems, this paper uses thread

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 March 2017 doi:10.20944/preprints201703.0068.v1

synchronization mechanism to resolve, which makes lock processing operation when the clients apply simultaneously available threads resources from threadPools to ensure that only one client can hold the lock. And the system will release the lock after the application process is completed and allow other clients to continue to get the lock using a synchronized synchronous block.

4. System Test

This SCADA system has been tested and implemented well at the real microgrid system. Tests mainly include the system startup test, the client connection test, business processing and load balancing test, resources recovery test and concurrent security module test.

4.1. System Startup Test

First, enter "net start mysql" to start the MySQL database at the command line console. Then enter "java -jar server.jar " to start the SCADA system. When the SCADA system starts, it will create "ServerSocket" connection socket, which provides two parameters to bind IP address and port number of this computer. Next, the system starts to read the configuration file, and then it will complete to read in the configuration information and load the startup module in configuration file by the static block. As we all know, the static block is executed during the initialization stage of the JVM loading and is executed only once. The system will create a threadPools first when it loads the startup module. The threadPools's maximum capacity is obtained from the configuration file. It will create a certain number of threads into on the threadPools in advance according to 10% of the maximum capacity in the threadPools. After that, the system will create two link tables: central controller link table (mcu link table) that can record the lower central controllers connection information and upper WEB monitoring system link table (web link table) that can record the upper WEB monitoring system connection information. At last, the system will invoke the "accept" method of "ServerSocket" connection socket to block and wait for the connection requests from clients. The results shows these threads's states including their serial number and flag that marks it used or not. Test results are shown in Figure 3 and Figure 4.

Figure 3. The Server of MySQL Starting

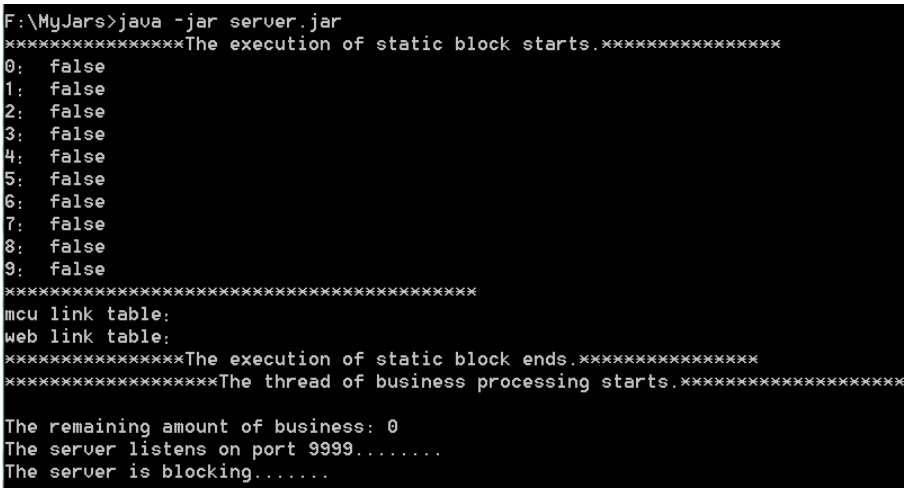

Figure 4. The SCADA System Starting

4.2. Client Connection Test

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 March 2017 doi:10.20944/preprints201703.0068.v1

After the SCADA system startup is completed, the client can send a connection request to the SCADA server. If the client's IP address is in the legal permission scope, the system will receive the protocol data frame by IO stream and parse it. If the protocol data frame parsed is legal according to the Modbus protocol specification and there is a thread available in threadPools, a connection will be established, otherwise the connection would be disconnected. Test results are shown in Figure 5 and Figure 6.

```
××××××××××××××××The execution of static block ends.××××××××××××××××
××××××××××××××××××××The thread of business processing starts.××××××××××××××××××××

The remaining amount of business: 0
The server listens on port 9999........
The server is blocking.......
New Link is coming: /127.0.0.1:36463
Registration failed, there is a illegal link!
The server is blocking.......
```

Figure 5. The SCADA System Receiving the Illegal Connection Request

```
New Link is coming: /127.0.0.1:19705
mcu link table:
1: Socket[addr=/127.0.0.1,port=19705,localport=9999]
web link table:
Send a frame.
0:  true
1:  false
2:  false
3:  false
4:  false
run method is executing: /127.0.0.1:19705
5:  false
Read the blocking......
6:  false
7:  false
8:  false
9:  false
The server is blocking.......
```

Figure 6. The SCADA System Receiving the Legal Connection Request

4.3. Business Processing and Load Balancing Test

The lower central controller sends electrical data frames to the SCADA system by means of UART Ethernet module. The SCADA system parses and processes received protocol data frames, and then it will insert them into the MySQL microgrid electrical table finally. The test results are shown in Figure 7 and Figure 8.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 March 2017 doi:10.20944/preprints201703.0068.v1

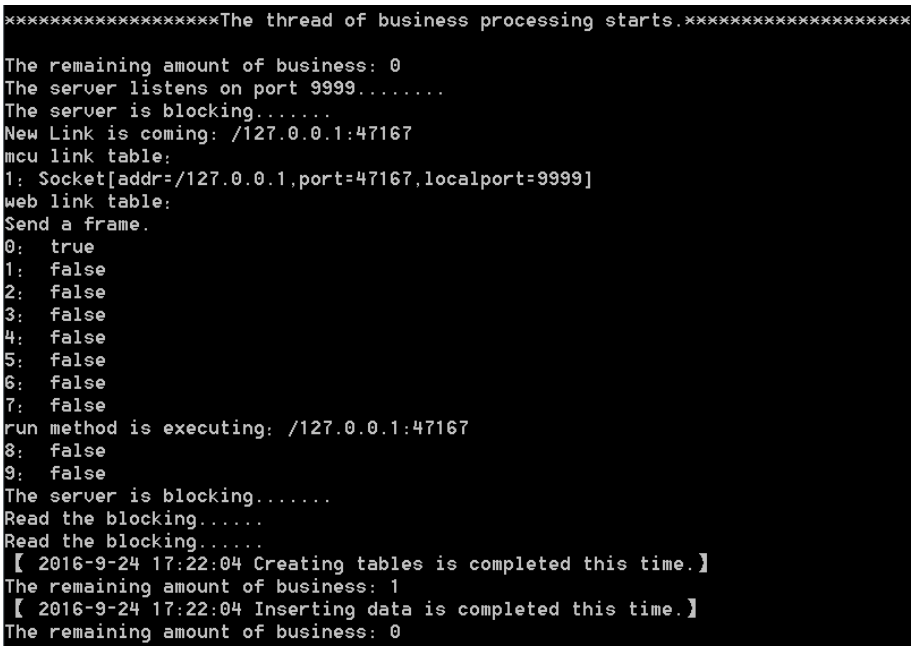

Figure 7. The Business Processing Test and Load Balancing Test

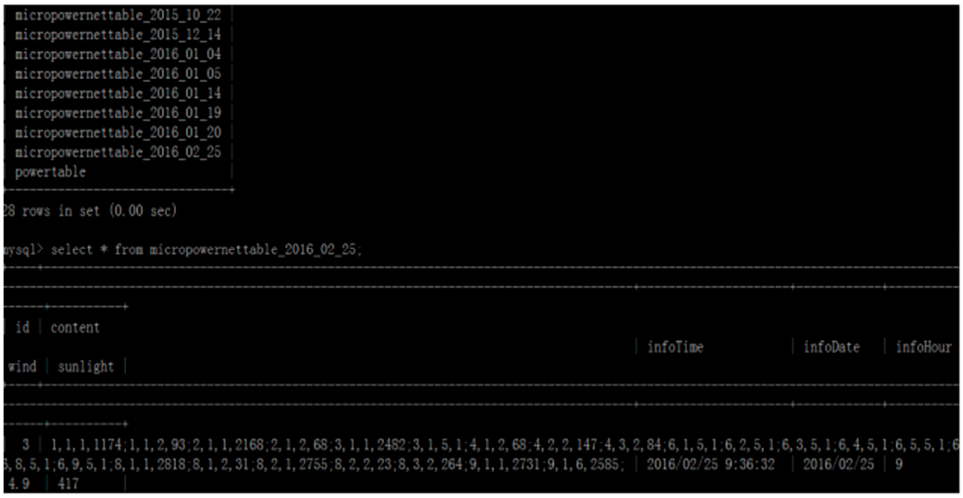

Figure 8. The Processed data Being Inserted into the Microgrid Electrical Table

4.4. Resources Recycling Test

SCADA system's resources recovery is mainly to solve the robustness problems of the system. The SCADA system is as a hub, so other communications need to be completed by using SCADA system. The SCADA system can detect abnormal link and then close the link. After that, SCADA system must recycle resources to prevent the OOM memory leak phenomenon, which may result in SCADA system downtime. Test result is shown in Figure 9.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 March 2017 doi:10.20944/preprints201703.0068.v1


```
java.net.SocketException: Connection reset
        at java.net.SocketInputStream.read(SocketInputStream.java:196)
        at java.net.SocketInputStream.read(SocketInputStream.java:122)
        at java.io.DataInputStream.read(DataInputStream.java:100)
        at com.dlb.Thread.MyThreadRun.run(MyThreadRun.java:67)
        at java.lang.Thread.run(Thread.java:745)
0:  false
1:  false
2:  false
3:  false
4:  false
5:  false
6:  false
7:  false
8:  false
9:  false
mcu link table:
web link table:
Close the MyThreadRun system resources!
The threads pool is recycling!
```

Figure 9. The SCADA System Closing the Abnormal Link and Recycling System Resources

4.5. Concurrent Security Module Test

When multiple clients connect to the SCADA system, the system can work normally. Besides, any abnormal client does not affect the other normal communication between the client and the SCADA server. Test results are shown in Figure 10 and Figure 11.

```
The server is blocking.......
run method is executing: /127.0.0.1:4539
Read the blocking......
New Link is coming: /127.0.0.1:4574
mcu link table:
1: Socket[addr=/127.0.0.1,port=4574,localport=9999]
web link table:
Send a frame.
0:  true
run method is executing: /127.0.0.1:4574
1:  true
Read the blocking......
2:  false
3:  false
4:  false
5:  false
6:  false
7:  false
8:  false
9:  false
The server is blocking.......
```

Figure 10. Multiple Clients Connecting to the SCADA System

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 March 2017 doi:10.20944/preprints201703.0068.v1


```
The server is blocking.......
java.net.SocketException: Connection reset
        at java.net.SocketInputStream.read(SocketInputStream.java:196)
        at java.net.SocketInputStream.read(SocketInputStream.java:122)
        at java.io.DataInputStream.read(DataInputStream.java:100)
        at com.dlb.Thread.MyThreadRun.run(MyThreadRun.java:67)
        at java.lang.Thread.run(Thread.java:745)
0:  true
1:  false
2:  false
3:  false
4:  false
5:  false
6:  false
7:  false
8:  false
9:  false
mcu link table:
web link table:
Close the MyThreadRun system resources!
The threads pool is recycling!
```

Figure 11. One Client Being Abnormal but not Affecting Others

4.6. Control function Test

In this WEB monitoring system, the administrator can control switches of six parts of microgrid hardware platform: Load 1, Load 2, Load 3, PV array, Wind generator and Storage battery. When the power generation is insufficient to supply all the loads, the administrator can give an instruction to choose one load to be disconnected to the microgrid system. As shown in Figure 12, if the administrator chooses Load 1 to be disconnected to the microgrid system and click the confirm button in the WEB monitoring system, the background program will form a frame of control according to the instructions of the foreground interface. And then the control frame will be sent to the SCADA system through the IO stream of socket. After that, the SCADA system sends the instruction to the central controller by the module of UART Ethernet and inserts this operation into the microEventTable of MySQL database as shown in Figure 13 and Figure 14. At last, the microgrid hardware platform will carry out the instruction to cut off the Load 1 to make the microgrid be in motion safely and stably which is shown in Figure 15.

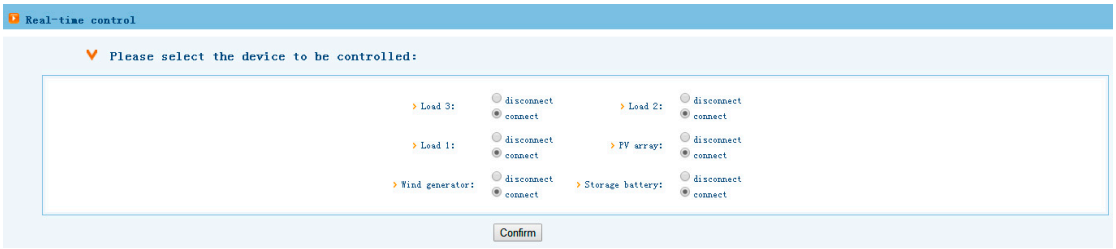

Figure 12. Real-time control user interface in the WEB monitoring system

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 13 March 2017 doi:10.20944/preprints201703.0068.v1


```
The server is blocking.......
New Link is coming: /127.0.0.1:43710
mcu link table:
1: Socket[addr=/127.0.0.1,port=43679,localport=9999]
web link table:
51: Socket[addr=/127.0.0.1,port=43710,localport=9999]
Send a frame.
Print threads' s states including their serial number and flag:
0:  true
1:  true
2:  false
3:  false
4:  false
5:  false
6:  false
7:  false
8:  false
run method is executing: /127.0.0.1:43710
9:  false
The server is blocking.......
Read the blocking......
Send a frame.
Scada sends control instruction to mcu!
Read the blocking......
【 2017-3-7 19:00:25 Inserting data is completed this time.】
The remaining amount of business: 0
```

Figure 13. The SCADA system transmiting the control instruction to the module of UART Ethernet

```
Stable Library
=========================================
Native lib Version = RXTX-2.1-7
Java lib Version   = RXTX-2.1-7
Connect to the COM3 successfully!
Search port COM3 completely!
The mcu is receiving blocking!
The control frame is:
0 1 6 0 0 1 0 0 0
The module of UART Ethernet sends the control instruction to mcu!
The mcu is receiving blocking!
```

Figure 14. The module of UART Ethernet sending the control instruction to mcu

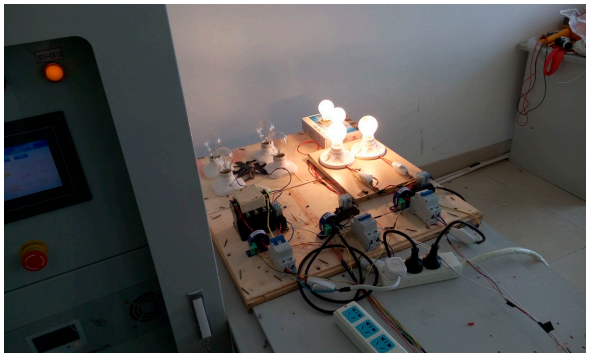

Figure 15. The microgrid hardware platform carrying out the instruction to cut off the Load 1

5. Conclusions

This paper introduces the importance and functions of SCADA system in the microgrid first. Then this paper presents the architecture of SCADA system as a middleware in microgrid and describes the design of SCADA system by Java in detail. It can realize the real-time data acquisition and storage, the control command parsing and transmitting, the system security and stability, load balancing and resource recovery of the microgrid. At last, the implementation and operation test of SCADA system is proved to be practicable and feasible. Its implementation can provide a common interface for other business system to access to the SCADA system data. And the SCADA system can be easily integrated into the microgrid monitoring systems. So it has good reusability, stability and easy expansibility. However, in order to improve the storage performance and access speed of the

system, the distributed file storage system HDFS and the parallel computing framework MapReduce can be used as the storage and computing platform for microgrid data in the future.

Author Contributions: Baochen Jiang, and Xiaoli Wang conceived and designed the experiments; Shuangshuang Li and Lubei Dong performed the experiments; Shuangshuang Li, Lubei Dong, and Xiaoli Wang analyzed the data; Shuangshuang Li and Lubei Dong wrote the paper. Baochen Jiang and Xiaoli Wang supervised the research project.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Z. Q. Shen, W. Deng, W. Pei, L. S. Mu and H. Ouyang, Design and implementation of microgrid SCADA platform, *Advanced Materials Research*, **2013**, *732-733*, 1358-1364.
2. R. Palma-Behnke, D. Ortiz, L. Reyes, G. Jiménez-Estévez and N. Garrido, A social SCADA approach for a renewable based microgrid—The Huatacondo project, Proc. of the IEEE Power and Energy Society General Meeting, Detroit, USA, **2011**, 7 pages.
3. B. R. Mehta and Y. J Reddy, Industrial Process Automation Systems: Design and Implementation, Elsevier Publishers, Amsterdam, The Netherlands, **2015**, 7, 237–300.
4. Pampashree and M. F. Ansari, Design and implementation of SCADA based induction motor control, *International Journal of Engineering Research and Applications*, **2014**, *4*, 5-18.
5. K. Cao, SCADA system research of the grid, *China New Technologies and Products*, **2013**, *5*, 23-23. (in Chinese)
6. K. T. Z. Win and H. M. Tun, Design and implementation of SCADA system based power distribution for primary substation (control system), *International Journal of Electronics and Computer Science Engineering*, **2014**, *3*, 254-261.
7. Y. N. Chen and W. Pei, Design and implementation of SCADA system for Micro-grid, *Information Technology Journal*, **2013**, *12*, 8049-8057.
8. E. Lázár, R. Etz, D. Petreuş, T. Pătărău and I. Ciocan, SCADA development for an islanded microgrid, Proc. of the 21st IEEE Int. Symp. for Design and Technology in Electronic Packaging, Brasov, Romania, **2015**, 147-150.
9. M. B. Mollah and S. S. Islam, Towards IEEE 802.22 based SCADA system for future distributed system, Proc. of the 1st Int. Conf. on Informatics, Electronics and Vision, Dhaka, Bangladesh, **2012**, 1075-1080.
10. M. Regula, A. Otcenasova, M. Roch, R. Bodnar, and M. Repak, SCADA system with power quality monitoring in Smart Grid model, Environment and Electrical Engineering (EEEIC), 2016 IEEE 16th International Conference, **2016**, June 7-10.
11. Van Hoa Nguyen, Quoc Tuan Tran, and Yvon Besanger. Scada as a service approach for interoperability of micro-grid platforms, Sustainable Energy, Grids and Networks, **2016**.