Research on Fault Transmission Method of Switchgear Based on Internet of Things

Jianxin Chen*, Shujiang YU, Yijun Jiang, Genming Wang

Hangzhou Power Supply Company of State Grid Zhejiang Electric Power Co. LTD, Hangzhou, 310016, China
*Corresponding author: yhdlcjx@126.com

Abstract—In order to meet the intelligent function of the switchgear, the fault transmission method of the switchgear based on the Internet of Things is researched and designed. Firstly, use the data collection terminal and monitor to monitor the parameters in the switchgear and transmit the detected faults; Secondly, use the Internet of Things method to design a monitoring switch platform to provide an operating environment for fault monitoring; Finally, verify the accuracy of the IOT(internet of things) technology in the fault transmission of switchgear.

1. INTRODUCTION
With the increasing popularity of smart grids in urban construction, smart grids require switchgear systems to achieve more intelligent functions. The transmission problem is the core problem that the switchgear must face, and the humidity problem is a more important problem in the transmission process. Especially in my country's high humidity areas, such as the Meiyu belt in the middle and lower reaches of the Yangtze River, most insulation accidents are caused by condensation problems[1-2]. Because the environment of the switchgear is prone to condensation, it will corrode the line for a long time and cause a great hidden danger to the operation of electrical equipment. The tightness of the switchgear equipment causes poor heat dissipation in the cabinet and gives the switchgear Dehumidification brings great inconvenience. Therefore, it is necessary to monitor the temperature and humidity in the switchgear through power transmission and distribution equipment and Internet of Things technology to prevent failures.[3-6]. When the transmission value reaches the set value of condensation, remotely dehumidify the switchgear. Using the combination of power transmission and transformation equipment Internet of Things technology and switchgear, it is of practical significance to design and develop an intelligent switchgear that can dynamically realize status detection and remote dehumidification[7-10].

2. SWITCHGEAR OVERALL ARCHITECTURE DESIGN
The overall structure diagram of the intelligent dehumidifier of the switchgear is shown in Figure 1. The switchgear intelligent dehumidification device is composed of STM32 as the controller as the lower computer, using Modbus for data transmission, the C# window design display interface as the upper computer, SAE as the server and APP as the monitoring client. The external temperature and humidity sensor of the STM32 controller transmits the data to the PC serial port through RS-485. After receiving the information, the PC serial port receives the data through the C# control and displays it on the interface. At the same time, the temperature and humidity information collected by
the lower computer transmits data to the server through the HTTP transmission protocol. The server receives the information and responds, and then stores the received data in the database. Eventually, the remote client accesses the database of the server and displays it on the interactive interface. If the detected data exceeds the threshold, the remote client will send control commands to the lower computer to achieve the function of remote dehumidification. The system can be monitored and remotely controlled on the PC side and the mobile communication equipment terminal. The PC side displays the real-time temperature and humidity information of the Switchgear in the host computer software, and draws a graph to realize the status information of the switchgear temperature and humidity. real-time monitoring. When it is found that the temperature and humidity information exceeds the set threshold, you can manually send control commands to the lower computer controller, or you can automatically achieve protective shutdown after the temperature and humidity of the switchgear reach the threshold. Through the Ethernet communication, the server is used as the communication relay to realize the remote monitoring of the temperature and humidity of the switchgear and the remote intelligent dehumidification function.

Figure 1 Overall design
3. DEVICE DESIGN
The lower computer of this system uses STM32 as the controller, including display OLED, 6 buttons, buzzer, temperature and humidity sensor, RS-485 module, ESP8266 wifi module, etc. The STM32 single-chip microcomputer master simulates the temperature and humidity sensor through a single bus, and then reads the temperature and humidity values of the sensor through the read sequence. The GPIO software of STM32 sets the high and low levels to control the relay, so as to adjust and control the environmental temperature and humidity of the switchgear. STM32 receives the command frame data from the host computer through RS-485 communication, and parses the command frame formulated according to Modbus, by judging whether it corresponds to the function code, and then makes the operation corresponding to the function code, and simultaneously transfers the collected data to the PC. The fixed area of the upper computer is displayed. Finally, STM32 will upload the collected data wirelessly to the data platform through ESP8266 wifi, and the corresponding real-time data can also be seen on the server side.

The host computer design of this system adopts C# programming of Visual Studio for design. C# programming has the characteristics of strong visibility and simple operation. It is an object-oriented programming language. The design of the host computer is divided into three aspects: interface design, code design, and data communication. The interface design adopts the control layout, and the data display interface, data sending button and command sending button are designed according to the requirements. The received data can be displayed on the host computer interface in real time. The data send button can send data to the server, and the command send button can send the server's control commands to the host computer. The code design assigns the data received from the serial port to the specified variable after calling the database class and network class, and passes the acquired data to splice the data points into a graph. In the data communication part, the serial port control of Visual Studio receives the data uploaded from the serial port of the lower computer. When the received switchgear status data, such as temperature and humidity exceeds the set threshold, it will automatically send an alarm signal to the lower computer. If the dehumidifier switch is not manually touched after the device alarm is detected, the upper machine will automatically send the dehumidifier trigger command to the lower machine within the set time to achieve the protection of the switchgear.

4. TRANSMISSION SERVER DESIGN
The server system design consists of code editing and database design. First of all, a simple server for new SAE applications is deployed in the server system, and secondly, the code editing module uses PHP language to receive and store the data of the host computer and store it in MySQL and feedback the control commands of the mobile client. The design of the server code is divided into two: receiving the data feedback control command from the host computer and receiving the control command from the client. PHP files make up the sending and receiving of data and control commands. Database design designs different database tables according to data types, and designs different data types to prepare for data storage. The form of data storage uses the database language of UPDATE to update data in real time. When the client needs data and the lower computer needs control commands, the required data can be feed back to the lower computer and the monitoring interface through database query. The communication process is shown in Figure 2.
In the data communication transmission part of the remote client, App Inventor interacts with the system server through the Web application. First, it configures the web application's URL and communication text protocol. When the Web configuration is completed, the server will send back a response text after sending the request to the server through POST. And use the display controls to display on the client interactive interface. The remote client's remote dehumidification command is also sent through the HTTP command. When the control button is pressed, the client will send the control command to the server, and the server will send it to the server after receiving it.

5. CONCLUSION
Based on the Internet of Things technology, this paper studies and designs the intelligent transmission problem of the switchgear, studies the dehumidification failure problem, and proposes the Internet of things monitoring and monitoring based on the environmental status information collection of the switchgear, host computer monitoring, server reception, and mobile communication terminal monitoring. Control platform. Intelligent remote temperature and humidity data monitoring and transmission is a control center for intelligent switchgear for fault dehumidification, which realizes integrated monitoring management and control of the switchgear. STM32 is the controller temperature and humidity data acquisition terminal that realizes on-site switchgear environmental status data parameter collection It provides a foundation for the dehumidification design of switchgear. The interface operating system in the host computer and mobile communication equipment client provides a powerful guarantee for remote monitoring and automatic dehumidification.

REFERENCES
[1] Ningning Xiao, Wei Yu, Xu Han. Wearable heart rate monitoring intelligent sports bracelet based on Internet of things[J]. Measurement,2020,164.
[2] Manik Sharma, Samriti Sharma, Gurvinder Singh. Remote monitoring of physical and mental state of 2019-nCoV victims using social internet of things, fog and soft computing techniques[J]. Computer Methods and Programs in Biomedicine,2020,196.
[3] Oihane Gómez-Carmona, Diego Casado-Mansilla, Frank Alexander Kraemer, Diego López-de-Ipiña, Javier García-Zubia. Exploring the computational cost of machine learning at the edge
for human-centric Internet of Things[J]. Future Generation Computer Systems, 2020, 112.

[4] Changhao Zhang. Design and application of fog computing and Internet of Things service platform for smart city[J]. Future Generation Computer Systems, 2020, 112.

[5] Jozef Mocnej, Adrian Pekar, Winston K.G. Seah, Peter Papcun, Erik Kajati, Dominika Cupkova, Jiri Koziorek, Iveta Zolotova. Quality-enabled decentralized IoT architecture with efficient resources utilization[J]. Robotics and Computer-Integrated Manufacturing, 2021, 67.

[6] Frédéric Lemoine, Tatiana Aubonnet, Noëmie Simoni. Self-assemble-featured Internet of Things[J]. Future Generation Computer Systems, 2020, 112.

[7] Christian Esposito, Oscar Tamburis, Xin Su, Chang Choi. Robust Decentralised Trust Management for the Internet of Things by Using Game Theory[J]. Information Processing and Management, 2020, 57(6).

[8] Hongpeng Zhu. Image quality assessment model based on multi-feature fusion of energy Internet of Things[J]. Future Generation Computer Systems, 2020, 112.

[9] Frédéric Lemoine, Tatiana Aubonnet, Noëmie Simoni. Self-assemble-featured Internet of Things[J]. Future Generation Computer Systems, 2020, 112.

[10] Bander Alzahrani, Nikos Fotiou. Enhancing Internet of Things Security using Software-Defined Networking[J]. Journal of Systems Architecture, 2020, 110.