

# Elevator Safety Monitoring System Based on Internet of Things

<https://doi.org/10.3991/ijoe.v14i08.9179>

Zihan Ming<sup>(✉)</sup>, Shaoyi Han, Zhanbin Zhang, Shuang Xia  
Sichuan Special Equipment Inspection Research Institute, Sichuan, China  
zihanming2384@163.com

**Abstract**—In view of the frequent occurrence of elevator accidents, an elevator safety monitoring system based on the Internet of things (IOT) was designed. First, the requirements of elevator safety monitoring system were analyzed in terms of function and performance, and the feasibility of the system was evaluated from perspectives of demand, technology, and practical operation. The design scheme of the system was then presented, which combined the Browser/Server (B/S) and Client/Server (C/S) architectures. As the client of the command and control center, the front-end monitoring system communicated and interacted with it and used the standard real-time transport protocol (RTP) for transmission. Finally, the elevator safety monitoring system was implemented. The test showed that the function and performance of the proposed elevator safety monitoring system achieved the designed target of the system and had practical application value.

**Keywords**—Elevator, IOT, monitoring system, safety monitoring

## 1 Introduction

The continuous development of China's economic development and urbanization has led to the rapid increase in the number of building constructions and the dramatic upsurge in the number of elevators. Elevator has become an indispensable tool in people's life. As the special equipment, the elevator is divided into two parts, namely, the operator, and the gathered in all types of public fields, such as shopping malls, office buildings, sightseeing elevators and other public places. Most of operators are usually not trained on the safety feature and operation of elevators and do not have the necessary skills to address failures and accidents. Meanwhile, the common users are lacking in terms of knowledge and skills of elevator safety operation. Thus, in places where elevators are frequently used, the probability of elevator failure is high and the social effect is considerable. Among the main factors that cause the hidden dangers of elevator safety, the quality of the elevator itself is 16%, the improper installation factor is 24%, and the daily maintenance and use is up to 60%. Therefore, strengthening the daily supervision of elevators is particularly important.

The technology of IOT originated from the advanced Western countries and has been rapidly developed ever since. It has been widely applied to environmental moni-

toring, medical system, intelligent control, and other fields. Studies have considered the application system of the broadband wide area network (WAN) by investigating the elevator safety monitoring system through IOT and video security application. Through the sensor and the Internet, the operation and management server of the elevator equipment are connected to realize the ideological theory of linked objects. The elevator monitoring and application system, which is composed of monitoring, reporting, feedback, and solving, has combined the advantages of digital TV, audio, video, broadband WAN transmission, and IOT technologies in realizing the supervisory functions of elevator state early warning, fault alarm, and elevator emergency safety handling to reduce and avoid elevator accidents. Saving manpower and material and financial resources will render the elevator safety supervision system efficient, practical, and reliable.

## **2 Literature review**

The continuous improvement of the global economic level and building technology has led to the increasing number and height of buildings and the number of people taking elevators. Therefore, the requirements for elevators have also increased to provide safe, convenient, reliable, and comfortable services and promote the continuous intelligence of the electric ladder. Numerous solutions for monitoring and management have been proposed locally and abroad for the safe operation of elevators, and corresponding systems have been developed.

Tu et al. (2013) studied an elevator monitoring system based on embedded system and IOT. The multi-sensor information collection provides information, such as vibration, acceleration, speed, driving noise, direction, floor station, elevator car door switch, power supply voltage, current, and temperature, as well as noise and temperature information when an elevator is occupied. The system provides technical support for elevator safety management, and the actual operation result is satisfactory [1]. Liu et al. (2012) studied an elevator remote monitoring scheme based on multithreading technology. This method can realize the communication with embedded terminals because all threads in multithreading technology can access the same global objects and shared resources. The main thread interface function was implemented by the monitoring center software, and the other management operations were also performed in the system. The test results showed that the application of multithreading technology in monitoring system improved the efficiency and security of the system [2]. Pan (2016) studied an elevator safety monitoring system based on IOT. In this system, cc2530, DHT11, and ADXL345 were used as the processor and communication module, temperature and humidity sensor module, and acceleration sensor module, respectively. The experimental results showed that the operation system could monitor the elevator's environmental parameters well and realize the real-time monitoring of the elevator running state [3]. Chen et al. studied an elevator monitoring system based on ZigBee and used the micro programmed control unit to control the operation of the elevator model. The information transmission module of ZigBee was used to transmit the information on the operation of the elevator model to the upper computer monitoring system, and the information on the elevator's running state and

the basic information of the query were displayed[4]. Li et al. studied and implemented an elevator safety remote monitoring system based on uninterrupted power supply (UPS) with global system for mobile communications (GSM). The system's terminal utilized GSM wireless communication module as the main control unit. It had reliable wireless five-party call subsystems and UPS function. The experiment demonstrated that the system was stable and dependable and had good compatibility. The system truly realizes the real-time performance of fault information feedback and reduced the blindness of troubleshooting. It also has a broad market prospect [5].

On the basis of the aforementioned studies, a series of elevator failure detection and alarm theories and systems have been developed locally and internationally due to high awareness of elevator accidents. However, at present, the elevator monitoring system here and abroad has its own limitations, such as the relatively few signals monitored by the system. Moreover, it has not been able to obtain audio recording and video footage in elevators. Thus, in the present study, an elevator safety monitoring system is designed based on IOT technology and completed through the analysis of the requirements of the physical connection command system of elevator operation safety.

### **3 Methodology**

#### **3.1 Da Vinci software architecture**

The Da Vinci software technology framework highlights the characteristics of component-based and loose coupling in the process of software development. It stipulates the application and algorithm library of the system and stipulates that the application program interface (API) between them is a unified and standard interface, such that the application development of the ARM ends, the development of the data signal processor (DSP) ends the algorithm library, and the integration of ARM and DSP can be made. Likewise, independent synchronization is developed. The Da Vinci technology framework is an efficient digital multimedia solution containing four elements, namely, DSP-based processors, applications, development tools, and development technology support, which simplifies and reduces the time to develop digital video products and provides the ideal core power for video applications at the lowest cost [6]. In this technology architecture, the processor is only the basic link of the entire solution, and the complex processing links have been divided into several other parts. In the process of developing digital multimedia products, the developers can deliver the details of complex multimedia processing technology directly to the DSP provider. The mature API they provide directly bypasses the specific implementation process and completes the products, which can simplify the underlying details of the multimedia codec processing. The developers do not need to understand the mechanism of the audio and video codec algorithm. They only need to know the corresponding format that can complete the processing of the image and sound, and the entire system can be completed. Figure 1 shows the framework of Da Vinci software technology.

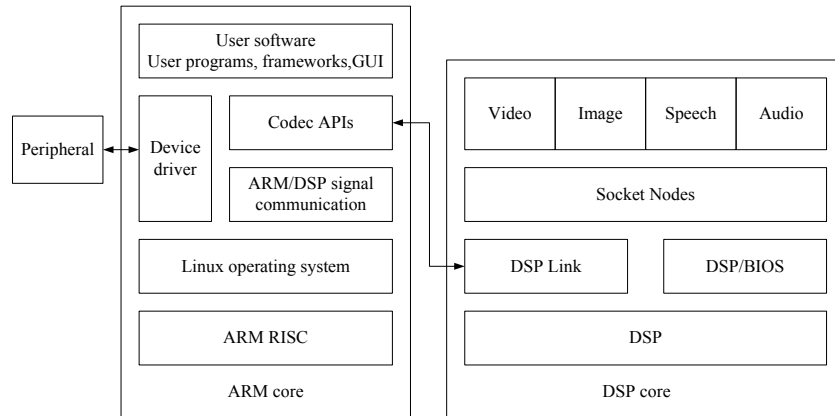


Fig. 1. Framework diagram of Da Vinci software technology

A complete digital multimedia application development based on Da Vinci software technology requires four steps, which are completed by the DSP algorithm engineer, codec server integration engineer, codec engine integration engineer, and ARM application engineer. In the actual digital multimedia applications, the part of ARM kernel based on the Da Vinci technology architecture is usually responsible for the application of the Linux operating system, whereas the DSP kernel is particularly responsible for audio and video codec and algorithm processing. A clear division of labor can accelerate the development process of the application, simplify the process, and improve the efficiency of the system.

### 3.2 Audio and video real-time transmission technology

RTP, which was proposed by IETF’s multimedia transmission working group in 1996, defines the standard format for multimedia data packets to be transmitted on the Internet. The RTP is usually combined with RTSP for streaming media server systems and video conferencing systems. It provides point-to-point real-time transmission rules that can realize real-time transmission of multimedia data in unicast or multicast network mode [7].

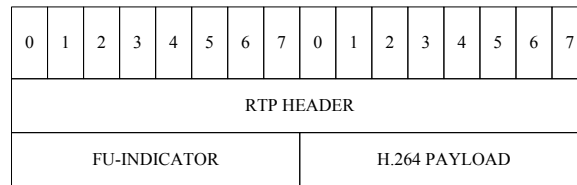
H.264 is the tenth part of the MPEC-4 standard, also known as advanced video coding. It is block-oriented, motion-compensated, high-precision video recording, compression, and release standard [8]. Usually H.264 video data are network abstraction layer unit (NALU) type, and the NALU head is one byte in length. Its structure is shown in Figure 2.

0	1	2	3	4	5	6	7
F	NRI		TYPE				

Fig. 2. NALU header data format

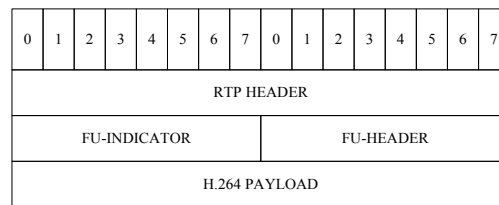
In Figure 2, F denotes the forbidden\_zero\_bit with default value of 0. If the receiver receives the data 1, this data can be discarded directly. The Nomura Research Institute (NRI) bit denotes nal\_ref\_idc and occupies 2 bits, the value of which represents the importance of the current packet. It is also convenient for discarding the packet in a different network environment. TYPE bits occupy 5 bits, which represent the status of H.264 data. If an H.264 frame is less than a maximum transmission unit (MTU), then it is only to be packaged into an RTP packet. TYPE can select any value from 1 to 23, and if the H.264 frame is larger than the MTU, then it must be divided into multiple RTP packets; thus, TYPE can be 28 or 29.

Figure 3 shows the structure of the RTP packet for an H.264 frame less than 1 MTU.



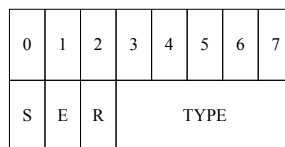
**Fig. 3.** RTP structure diagram without subcontracting

Meanwhile, Figure 4 shows the structure of the RTP packet for an H.264 frame larger than 1 MTU.



**Fig. 4.** Subcontracting RTP structure diagram

The structure of FU-INDICATOR in Figure 4 is the same as the NALU header data format. Figure 5 shows the FU-HEADER structure.



**Fig. 5.** Structure diagram of FU-HEADER data

In Figure 5, S denotes start; the current data set to 1 is the first segment, and the other cases are set to 0. E represents the end; the current data set to 1 is the last one, and the other situations are set to 0. R refers to reserved; the reservation bit must be

0.TYPE refers to the type, which can be 28 or 29. The FU-INDICATOR and FU-HEADER are implemented by the H.264 load content after segmentation, and their calculation algorithms are shown in Formulas (1) and (2), respectively.

$$FU - INDICATOR = (NAL \& 0 \times E0) | 0 \times 1C \quad (1)$$

$$FU - HEADER = (NAL \& 0 \times 1F) | 0 \times 80 \quad (2)$$

Advanced audio coding (AAC) is an audio coding technology based on moving picture experts group-2 development. The RTP transmission of AAC is defined in radio frequency chart 3640. After encoding, the AAC frame has 7 or 9 bytes of automated data and telecommunications service (ADTS) header, which contains the basic information of the frame. The transmission of AAC through RTP is relatively simple. The ADTS of the AAC frame is removed and 2 bytes of AU-HEADER\_LENGTH and 2 bytes of AU-HEADER are added. To simplify the operation, the RTP frame can be set to transmit only one AAC frame; thus, if the value of the FU-INDICATOR-LENGTH can be set to  $0 \times 10$ , then the frame length in ADTS is converted into the AU-HEADER. The calculation method is shown in Formula (3).

$$AU - HEADER = (FrameLen \& 0 \times FFE0) | (FrameLen \& 0 \times 1F) \quad (3)$$

### 3.3 Demand analysis of elevator safety monitoring system

The elevator operation safety system is composed of three parts, namely, front-end monitoring system, command and control center system, and network transmission system. The front-end monitoring system is one of the core systems of the entire system. It communicates directly with the registration server, streaming media server, and elevator state server in the upper command and control center. It is responsible for collecting the state of the elevator operation and completing the monitoring and alarm work, as shown in Figure 6.

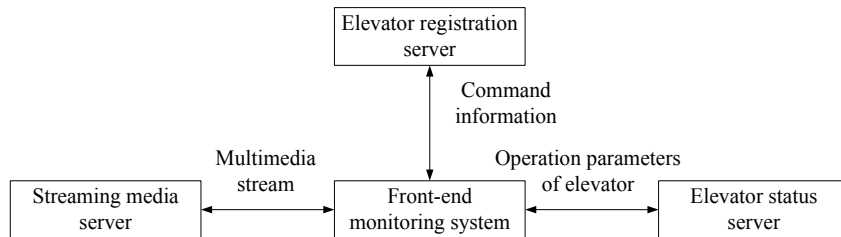


Fig. 6. Business relation diagram of the front-end monitoring system

The elevator operation safety control system is located in the monitoring and command of all the elevator equipment in a city or area. It is suitable for residential buildings, business offices, hospitals, and schools. However, the network environment of each elevator is inconsistent. Thus, various data transmission methods in the front-end monitoring system must be utilized to fit a variety of network environment. When

the system is transmitted in the 3G mode, it must support three different mobile networks, which can be Mobile, Unicom, and Telecommunication. After switching the transmission mode, the system can continue the active switching and complete the corresponding network setting work.

The front-end monitoring system must accomplish the 24-hour monitoring of the elevator and generate the corresponding running state records of each elevator in the command and control center, which is convenient for checking the operation state of the elevator in a certain stage. The system must collect real-time operation parameters of the elevator through sensors, including opening doors, closing doors, people, elevator running direction, and speed. After collecting such information, the system must report the elevator to the remote command and control center immediately while analyzing and summarizing the current state and determining whether the current elevator is under normal running state.

Although the front-end monitoring system is not a special security system, it needs to perform monitoring functions. The system provides local monitoring to provide security functions; meanwhile, after the convenience of elevator failure, the cause of the accident is tracked and the related personnel are held accountable. The system must record a video of the elevator operation for a period of time (for example, 24 hours). This function requires the system to record the audio and video in the elevator and synthesize the video files to be saved locally. The format of the video files synthesized by the system must be the mainstream data format and can be played in the mainstream multimedia. On the video screen, the current recording time must be displayed to facilitate the analysis of the situation at a certain time.

### **3.4 Design of elevator fault monitoring system**

From the system function, the front-end monitoring system adopts the combination of C/S and B/S architectures. The C/S architecture is used as the client of the upper command and control center and the command and control center platform for data exchange. It does not need to interact with the user. Meanwhile, the B/S architecture is utilized to provide the staff to install the equipment. The front-end monitoring system is set by the browser based on the B/S architecture and then it runs independently to build a streamlined Web server. In view of the non-functional requirements, a relative elevator independent data sensor is used to meet the high availability of the system, such that it will not destroy and affect the sensor system of the original elevator system and be compatible with all brands of elevators. The front-end monitoring system can be designed by multi-process, and Figure 7 shows the software skeleton diagram.

The system can realize the communication function of three servers in the command and control center system through three core processes, namely, elevator status reporting process, streaming media service process, and command and receive process. These three processes are independent of one another. The inter-process communication is realized through the Unix domain socket. In addition, the front-end monitoring system also needs to provide two additional processes; that is, the embedded Web server basic object adapter (BOA) process, which facilitates the installation

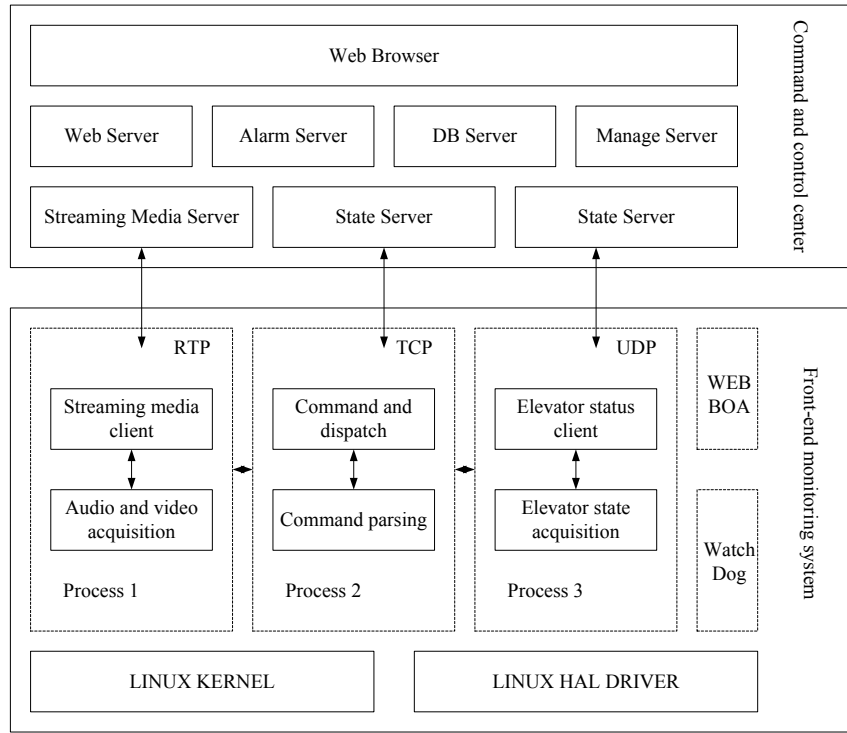


Fig. 7. Overall framework design of the front-end monitoring system software

for the basic system configuration, and the watchdog process, which ensures that the system can run independently and prevents system error for the reliable operation of the entire system.

As the terminal part of the whole system, the front-end monitoring system needs to provide online functions, such as social software. It can respond to the commands sent by the registered server in time while reporting the elevator running data collected by the terminal sensor to the streaming media server. The front-end monitoring system terminal is installed in the elevator car and is in an unattended mode. Therefore, the system design requirements can have high reliability and stability. The system has also developed two modes of communication between the cable network and 3G to ensure high throughput. The remote and local configuration methods are developed to facilitate system configuration. These two methods are based on the Web mode, which ensures that users can configure the system only through the browser. At the same time, the watchdog module is added and runs independently to guarantee that the system can run reliably for a long time under the condition of unattended operation.



### 3.5 Realization of the elevator fault monitoring system

The system is divided into nine modules based on the functional module division of the front-end monitoring system. The nine modules are the equipment registration and live module, command and receive module, elevator status acquisition and reporting module, network transmission module, audio and video transmission module, audio and video synthesis module, remote configuration upgrade module, BOA server module, and watchdog module. The modules that have complex functions in the embedded system are usually implemented by the mature open-source module. The embedded system also employs this solution to implement each module. However, this study will only explain the implementation process of the elevator state collection and reporting module due to limitation of article space.

The collector part of the front-end monitoring system is placed on the upper part of the elevator car and communicates with the host through Zigbee communication. Figure 8 shows the flowchart of an elevator status report.

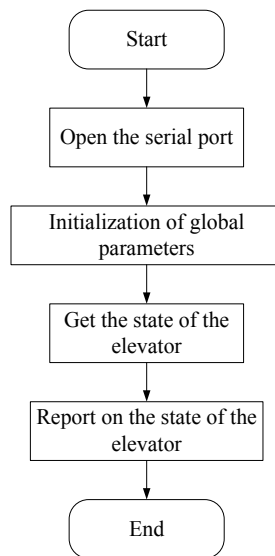


Fig. 8. Flowchart of elevator status report

The network transmission module of the front-end monitoring system and the transmission of wired Ethernet mode do not require excessive design. The ARM9Linux system itself is supported by the system, which focuses on the realization of 3G mode transmission. The system supports the network mode of the three main operators to be compatible with the 3G network mode. The front-end device must develop the drivers of the three different modes of 3G modules to be compatible with China Mobile, China Unicom, and China Telecom because all the 3G module hardware in the current market can only support one mode.

For the encoded audio and video data, uploading the commands after receiving them from the command and control center is necessary. For the real-time require-

ments, the monitoring system uses RTP for transmission. The front-end monitoring system is implemented by transplanting the jrtp-lib-3.7.1 protocol library and the jthread-1.2.1 thread library to support the RTP, thereby saving the work of manually encapsulating the RTP message.

After the introduction of reply into the thread library, it only needs to add the macro definition of RTP\_SUPPORT\_THREAD when the program is written, and the jrtp-lib will automatically call the jthread library for thread creation. Moreover, jrtp-lib provides a simple way to invoke and it can send and receive RTP data in few steps.

## 4 Results

The system is tested in two parts. One is to test the data on the front-end monitoring system correctly through the command and control center system, which can be performed directly through the browser access. The other part is the direct test of the front-end monitoring system, in which the software is tested. The test results are obtained directly in the front-end monitoring system.

When clicking on an elevator, the video and audio dialogue of the elevator can be checked. If multiple elevators are opened, multiple elevator monitoring videos can be also viewed, monitoring the screen, providing a multi-screen preview, and watching the real-time video pictures of multiple elevators simultaneously. Figure 9 shows the diagram of the actual test run effect.



Fig. 9. Sub-screen monitoring interface of the command and control center

As shown in Figure 9, the simultaneous video surveillance of four elevators don't affect one another. The front-end monitoring system can select different data channels based on different devices, such that the data will not be confused. At present, four elevators are tested in different ways of communication, that is, mobile TD-SCDMA, WCDMA, EVDO, and cable. From the diagram, the definition of the four pictures is basically the same, which proves that the data bandwidth occupied by the multimedia is less (Figure 9). Even at a relatively slow speed compared with the wired network, the 3G network can still transmit smoothly and has high bandwidth utilization.

In the audio reception, the audio sound is fresh and synchronizes with the real-time video. The overall video network delay is only 1s, thereby fully satisfying the high real-time demand of the system. In front-end monitoring system and command and control center for voice intercom, the speech is clear and no carton exists, which can meet the established requirements.

The command and control center provides the function of the elevator operation and simulates the real time operation of the elevator, the floor, the door switch, and the running direction to display the running state of the elevator intuitively. This function can reflect the elevator running state of the front-end monitoring system intuitively, as shown in Figure 10.



**Fig. 10.** Simulated elevator interface diagram

As shown in Figure 10, the real-time operation status of the elevator is reported to be normal and the floor is accurately assessed. When the elevator fails, the front-end monitoring system can report the failure voluntarily and the function remains normal in the actual testing process.

After setting up the upgrade parameters remotely, the front-end monitoring system is upgraded. At this point, the FTP server in the network where the front-end monitoring system is located must be guaranteed to be normally open. During testing, if the FTP server is not opened, then the result of the feedback upgrade fails. Moreover, when the upgrade is completed, the front-end monitoring system is automatically restarted and run according to the default system parameters. The front-end monitoring system can open the FTP service normally and download the corresponding upgrade package, parse it, and complete the upgrade.

Table 1 presents the software performance test and results of the front-end monitoring system.

The test results completed on the front-end monitoring system at the end of the command and control center and the front-end of the monitoring system shows one failure in achieving the expected goal, that is, a long-time remote video function (Table 1). The video time is changed to 12 hours by modifying the test parameters and reducing the test requirements. After several tests, the front-end monitoring system can operate normally. In an actual situation, a long-time video operation will not exceed 3 hours should there be an elevator failure. Usually, 12 hours of normal talk time can fully meet the daily needs; if speaking for more than 12 hours is necessary, then a

re-talk must be opened to solve the problem. Therefore, on the basis of the test results and the analysis of the failure of the test, the IOT-based elevator fault monitoring system can meet the actual needs and complete the required functions. This system has the value and significance for development.

**Table 1.** Software performance testing and results of the front-end monitoring system

Test items		Test steps and data	Expected results	Pass/Fail
Broken wire reconnection	Data collector and the host	Without power cut, the data collector and the host will be disconnected. After a period of time, they can be restored to the state of being connected before.	Connection can be automatically restored	Pass
	Host and platform	Without power cut, let the host and the platform disconnect. After a period of time, they can be restored to the state of being connected before.	Connection can be automatically restored	Pass
Battery use		Battery is powered, until it is depleted. Power it on, and the device is restarted.	Equipment can run normally	Pass
Long-time run		Equipment runs a week for a long time	Equipment can be operated correctly	Pass
Long-time intercom		Long time intercom for 48 hours	Equipment can operate normally	Pass
Long local video		Long-time local video for 48 hours	Equipment can be videotaped normally	Pass
Long-time remote video		Long-time remote video for 48 hours	Equipment can operate normally	Fail
Reboot of equipment		Reboot the equipment many times	Equipment can operate normally	Pass

## 5 Conclusion

This study introduces the Da Vinci software architecture and real-time audio transmission technology. The system can be compatible with other systems and can be played directly through the mainstream media player for the transmission of data by using the standard RTP. This study presents the implementation plan of the system based on the analysis of system requirements. The entire system adopts the combination of B/S and C/S architectures and the front-end monitoring system that acts as the client of the command and control center to communicate and interact with it. The results of the system test indicate that the functions and performance of the elevator safety monitoring system have reached the intended target of the system design, completed the required functions, and have the value and significance for development.

## 6 References

- [1] Tu, SL. Wu, ZY. Qian, B. Research of the Elevator Monitoring System Based on the Internet of Things. *Applied Mechanics & Materials*, 2013, vol. 423(426), pp. 2380-2385. <https://doi.org/10.4028/www.scientific.net/AMM.423-426.2380>
- [2] Liu, J. Fan, DS. Wu, Y. Fu, GJ. Liu, MJ. Study of Remote Elevator Monitoring Fault Detection System. *Advanced Materials Research*, 2012, vol. 499, pp. 459-463.
- [3] Pan, X. The Design and Reliability Analysis of Elevator Monitoring System Based on the Internet of Things. *International Journal of Smart Home*, 2016, vol. 10(12), pp. 183-192. <https://doi.org/10.14257/ijsh.2016.10.12.17>
- [4] Chen, JW. Yang, HJ. Fan, WL. Research of Elevator Remote Monitoring System Based on Zigbee Technology. *Advanced Materials Research*, 2014, vol. 1049(1050), pp. 1222-1226.
- [5] Li, WF. Zhang, SL. Yuan, HR. The Design of the Elevator Remote Monitoring System Based on GSM. *Applied Mechanics & Materials*, 2015, vol. 743(7), pp. 876-881. <https://doi.org/10.4028/www.scientific.net/AMM.743.876>
- [6] Zhang, XQ. Mechanics Performance of Elevator Monitoring System Based on GPRS and ZIGBEE Network. *Applied Mechanics & Materials*, 2012, vol. 214, pp. 543-547. <https://doi.org/10.4028/www.scientific.net/AMM.214.543>
- [7] Yu, X, Feng, SC. Luo, JW. Design of the Elevator Wireless Monitoring System Based on LabVIEW. *Applied Mechanics & Materials*, 2014, vol. 741, pp. 328-332.
- [8] Liu, XK. Chen, Y. Yu, HN. Research on Web-Based Elevator Failure Remote Monitoring System. *Applied Mechanics & Materials*, 2014, vol. 494(495), pp. 797-800. <https://doi.org/10.4028/www.scientific.net/AMM.494-495.797>

## 7 Author

**Zihan Ming, Shaoyi Han, Zhanbin Zhang and Shuang Xia** are with the Sichuan Special Equipment Inspection Research Institute, Sichuan, China.

Article submitted 17 June 2018. Resubmitted 23 July 2018. Final acceptance 01 August 2018. Final version published as submitted by the authors.