Design and Implementation of Tunnel Environment Monitoring System Based on LoRa

Gui Zhu, Honggang Zhao(✉), Zuobin Liu, and Chen Shi

College of Information and Communication,
National University of Defense Technology, Xi'an, China
`zhugui53@163.com`, `hgz_nwpu@163.com`, `2551105254@qq.com`,
`shishen26@163.com`

Abstract. Tunnel is the basic construction project to construct underground, underwater, and mountain buildings, such as mines, subways, and so on. Effective monitoring of tunnel environment plays on important part in the tunnel management, it can provide early warning before the accidents and minimize the hazard of the accident. Compared with the traditional IoT technologies such as ZigBee, Wi-Fi, and so on, the LoRa (Long Range) technology emerging in recent years performs better in terms of coverage range, connection number, and energy consumption, etc. It is foreseeable that the LoRa technology will certainly make a tremendous promotion in the field of tunnel environment monitoring. In this paper, A Lora-based tunnel environment monitoring system is designed and implemented. Firstly, the system architecture is described. Then, the detailed design and implementation of LoRa terminal hardware, LoRa terminal embedded software, LoRa server and monitoring App is given respectively. Finally, the experimental results show that the system has good network coverage capacity and communication reliability, can accurately monitor the tunnel environment and have low power consumption. This system has good promotion and application value.

Keywords: Tunnel · Environment monitoring · LoRa · Internet of Things

1 Introduction

Tunnel is the basic construction project to construct underground, underwater, and mountain buildings, such as mines, subways, and so on. With the wide application of IoT technologies in different areas, it has become an inevitable trend that the IoT technologies are used in the tunnel construction and maintenance, in order to improve the reliability of transmission and the intelligent level of data processing.

The main factors affecting tunnel include: toxic gas concentration, temperature and humidity, carbon dioxide concentration, and water leakage which may lead to the collapse. In literature [1], temperature, humidity, PM2.5 concentration and combustible gas concentration are monitored, then transmitted to the mobile terminal simultaneously by Bluetooth. Besides, the SMS alarm message is generated when gas concentration exceeds the threshold. In literature [2], the ZigBee technology is used, then the early

B. Li et al. (Eds.): IoTaaS 2019, LNICST 316, pp. 621–638, 2020.
https://doi.org/10.1007/978-3-030-44751-9_53

warning and judgment information from distributed gas sensor nodes, can be transmitted to the ground monitoring stations through the relay nodes. In literature [3], a ZigBee network is deployed, not only the mine's temperature, gas concentration but also the personnel information such as pulse, are monitored and transmitted to the server in real time.

Obviously, the traditional IoT technology has greatly promoted the tunnel environment monitoring system. However, due to the complexity of tunnel environment, it is difficult to solve the contradiction between low power consumption and wide coverage. By contrast, the LoRa technology emerging in recent years performs better in terms of coverage range, connection number and energy consumption, etc. It is foreseeable that the LoRa technology will certainly make a tremendous promotion in the field of tunnel environment monitoring (Table 1).

Table 1. The comparison between LoRa and traditional IoT technologies such as ZigBee [4]

Performance	Technologies			
	LoRa	ZigBee [5]	Wi-Fi [6]	GSM/GPRS [7]
Distance	5–15 km	10–75 m	100 m	10 km
Rate	300 Kbps	250 Kbps	11 Mbps	2 Mbps
Low-power	Ultra-low	Support	No support	No support

In this paper, the architecture of the tunnel environment monitoring system based on LoRa is proposed. It consists of LoRa terminal, LoRa gateway, LoRa server and monitoring App. The detail of LoRa terminal hardware, embedded software and monitoring App is given. The LoRa server is deployed on Aliyun. Finally, the communication performance, monitoring performance and terminal power consumption are tested. The experimental results show that the system has good network coverage capacity and communication reliability, can accurately monitor the tunnel environment and have low power consumption. This system has good promotion and application value.

2 LoRa-Based Tunnel Environment Monitoring System Architecture

In this paper, the problem that may be encountered during tunnel construction and maintenance, and the advantages of LoRa [8, 9, 10] technology, are fully taken into consideration, then the architecture of tunnel environment monitoring system based on LoRa is proposed, which is shown in Fig. 1.

The system consists of four parts: LoRa terminal, LoRa gateway, LoRa server and monitoring App. The tunnel environment is monitored by LoRa terminal, then the collected information is transmitted to the monitoring App via LoRa gateway and LoRa server.

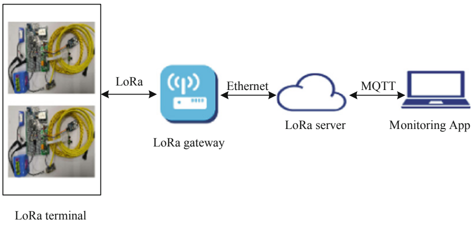


Fig. 1. LoRa-based tunnel environment monitoring system architecture.

The LoRa terminal is deployed in the tunnel to collect carbon dioxide concentration, toxic gas concentration, temperature, humidity, and water leakage. The number and location of LoRa terminals are determined by the actual tunnel environment. The LoRa terminal consists of 7 hardware parts: MCU module, leakage monitoring module, Temperature and humidity monitoring module, gas monitoring module, alarm function module, LoRa transmission module and power management module. The software contains of 8 parts: serial communication module, stop mode, external interruption, temperature and humidity acquisition, RTC clock wake-up, carbon dioxide concentration acquisition, alarm control, LoRa transmission. In order to reduce energy consumption as much as possible, the LoRa terminal periodically wakes up, then it collects the tunnel environment information and send it to the monitoring App by LoRa technology (Figs. 2 and 3).

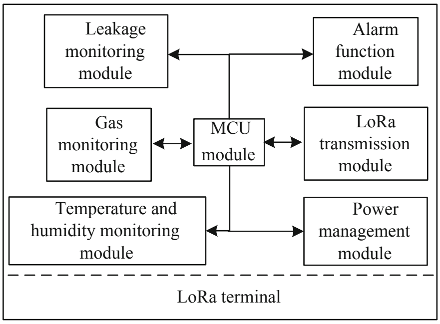


Fig. 2. The hardware structure diagram of LoRa terminal.

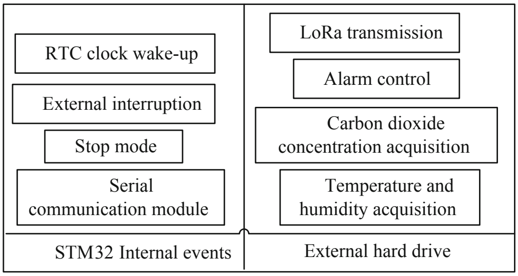


Fig. 3. The software structure diagram of LoRa terminal.

LoRa gateway collects the uplink data of LoRa terminal within the coverage, and establishes a connection with LoRa server through Ethernet or wireless communication. Then it sends the uplink data frame and the status frame to LoRa server through UDP. In this paper, the system uses the indoor gateway RGWC490LA-G of Changsha Rime Company.

LoRa server manages LoRa gateway and LoRa terminal. The LoRa server is set up based on the open source LoRa Server project. LoRa Server consists of a set of programs, including lora-gateway-bridge, loraserver, lora-geo-server and lora-app-server (Fig. 4).

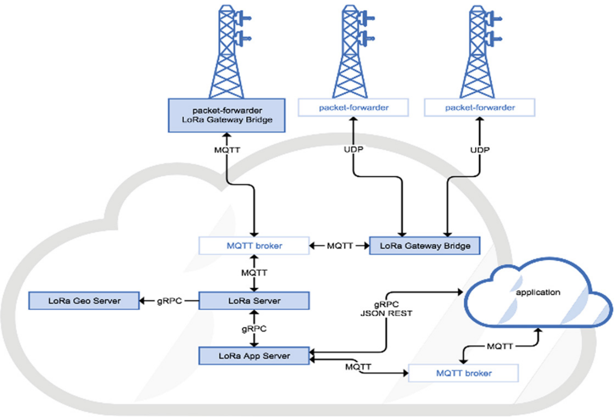


Fig. 4. The structure diagram of LoRa Server.

Monitoring App receives the uplink data from LoRa Server and stores it in SQL Server database. It consists of receiving data background, collecting data display module, management node module and alarm display module. Parallel processing of a large number of terminals is realized. It can receive and display data in real time.

3 Design and Implementation of LoRa Terminal

The LoRa terminal collects carbon dioxide concentration, temperature and humidity, toxic gas concentration and water leakage in the tunnel. It joins the LoRa network, and periodically sends the data to the monitoring App. In this section, the hardware and software of LoRa terminal are designed and implemented according to terminal requirements. At the same time, in order to achieve low power consumption performance, not only the power source classification control method, but also the periodical sleep and monitoring mechanism, are realized in the power management module hardware design and terminal embedded software design. Then the long-term stable operation of terminal power supply [11] can be realized through above method and mechanism.

3.1 LoRa Terminal Hardware Design and Implementation

LoRa terminal hardware consists of 7 modules, including MCU module, leakage monitoring module, gas monitoring module, alarm function module, temperature and humidity monitoring module, LoRa transmission module, and power management module. Detailed schematic diagram of LoRa terminal hardware is shown in Fig. 5.

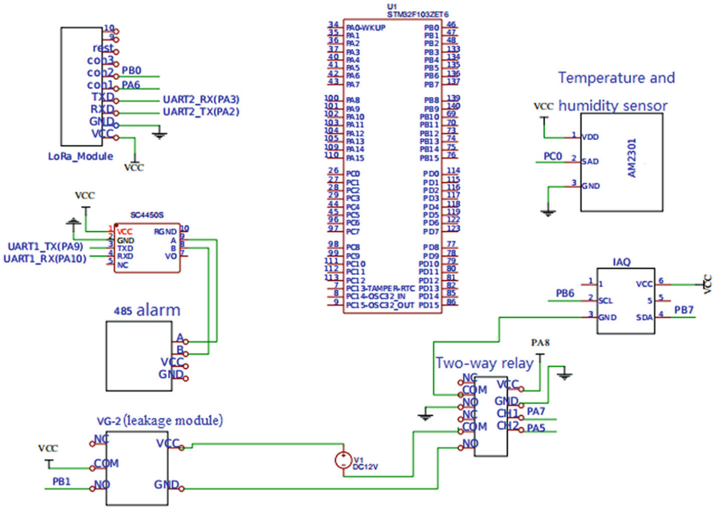


Fig. 5. LoRa terminal hardware schematic diagram.

MCU Module. In order to reduce the node's power consumption, this system uses STM32 [12] as the MCU module. STM32 is popular in the field of low-power IoT. MCU module communicates with temperature and humidity monitoring module and gas monitoring module through IIC protocol, with LoRa transmission module and alarm function module through serial port, and with water leakage monitoring module through interruption input.

Leakage Monitoring Module. The leakage monitoring module is composed of a water leakage monitoring controller VG-2 and a rope [13]. The leakage monitoring module is used to monitor the leakage within the coverage range of the rope. When the leakage is perceived, the leakage monitoring module pulls up the PB1 of STM32.

Gas Monitoring Module. Gas monitoring sensor [14, 15] uses iaq-core c. Its power supply voltage is 3.3 V. Iaq-core c is used to monitor carbon dioxide and toxic gas concentration. Its preheating time lasts 5 min. And the preheating process doesn't affect other module's operation, so the terminal's monitoring time can be minimize as much as possible.

Alarm Function Module. The alarm function module uses rs485 sound-light alarm equipment. It communicates with MCU module through TTL to 485 chip (SC4450S). It gives an alarm signal when toxic gas concentration exceeds the threshold.

Temperature and Humidity Monitoring Module. AM2301 is used as the temperature and humidity sensor [16]. Its precision for humidity is ±3% RH, and for temperature is ±0.3 °C. Its current consumption in sleep mode is 15 uA, and its current consumption in working mode is 500 uA.

LoRa Transmission Module. RNDU490L is used as the LoRa transmission module, and communication with the MCU module through serial port (TTL level). It is used to transmit the acquired carbon dioxide and toxic gas concentration, temperature, humidity and water leakage to the LoRa server.

Power Management Module. In the design of power management module [17], the power source classification control method is adopted to satisfy power supply requirements of different modules. The main power module consist of: leakage monitoring module for 9–30 V power supply without low power consumption mode, gas monitoring module for 3.3 V power supply without low power consumption mode, temperature and humidity monitoring module for 3.3 V power supply with low power consumption mode, LoRa transmission module for 3.3 V power supply with low power consumption mode.

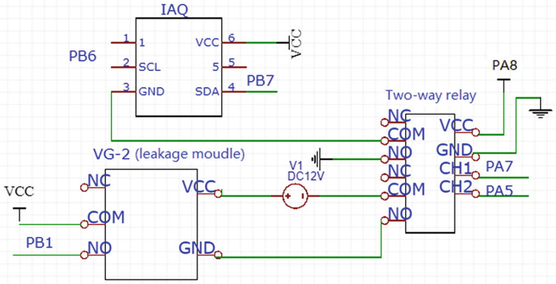


Fig. 6. Power management module.

As shown in the Fig. 6, the four modules can be divided into two types, according to the voltage levels of their power supplies. One is the leakage monitoring module, it is the only module that uses 12 V battery for power supply. The other three modules use the 3.3 V power supply, and they share the stabilized power source with STM32.

According to the working mode, the four modules can also be divided into two types. One is the LoRa transmission module, temperature and humidity module. They both have low power consumption mode. The other is the leakage monitoring module and gas monitoring module. They don't have low power consumption mode. So a relay is adopted to control their power supply, and the power supply is provided when needed. It is ensured that the unnecessary energy consumption is reduced when the leakage monitoring module and gas monitoring module don't work.

The RTC clock is used to wake up the STM32 periodically. After STM32 wakes up, the pin PA7 and pin PA5 are checked. If the pins are in high level, the MCU module reads the carbon dioxide and toxic gas concentration, temperature, humidity, and water leakage, then transmits it to the LoRa module. If the pins are in low level, the pin PA7 and pin PA5 are set to turn on the power supply of gas monitoring module and leakage monitoring module. Then the gas monitoring module preheats for 5 min, and the leakage monitoring module continuously monitors the water leakage.

3.2 Design and Implementation of LoRa Terminal Software

LoRa terminal software is the executive program of MCU module, which is used to realize the LoRa terminal workflow. It consists of the implementation of internal execution events and the preparation of external sensor drivers. According to the hardware design and software functional requirements, LoRa terminal software's workflow is divided into three working stages. The first stage is the 55 min sleep stage. The second stage is 5 min preheating stage. The third stage is acquisition and transmission stage of about 30 s [18].

Two issues should be fully taken into consideration in the LoRa terminal software design. One is to collect data correctly and transmit it through LoRa module. The other is to preheat the gas monitoring module for 5 min before it can be used.

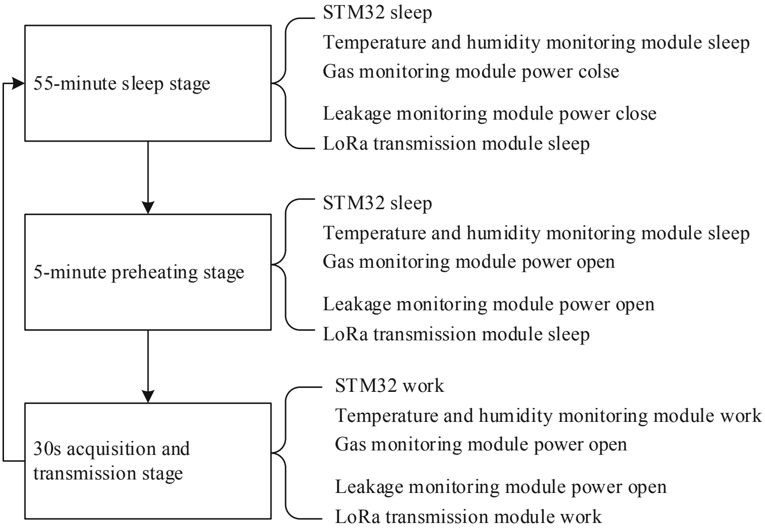


Fig. 7. LoRa terminal work phase.

According to the above consideration and the hardware design, the workflow of LoRa terminal is divided into three stages as shown in the Fig. 7. During the first stage, all modules running at the lowest power consumption. During the second stage, except for the gas monitoring module and leakage monitoring module, the other three modules are still running at low power consumption. During the third stage, all modules are working [19] (Fig. 8).

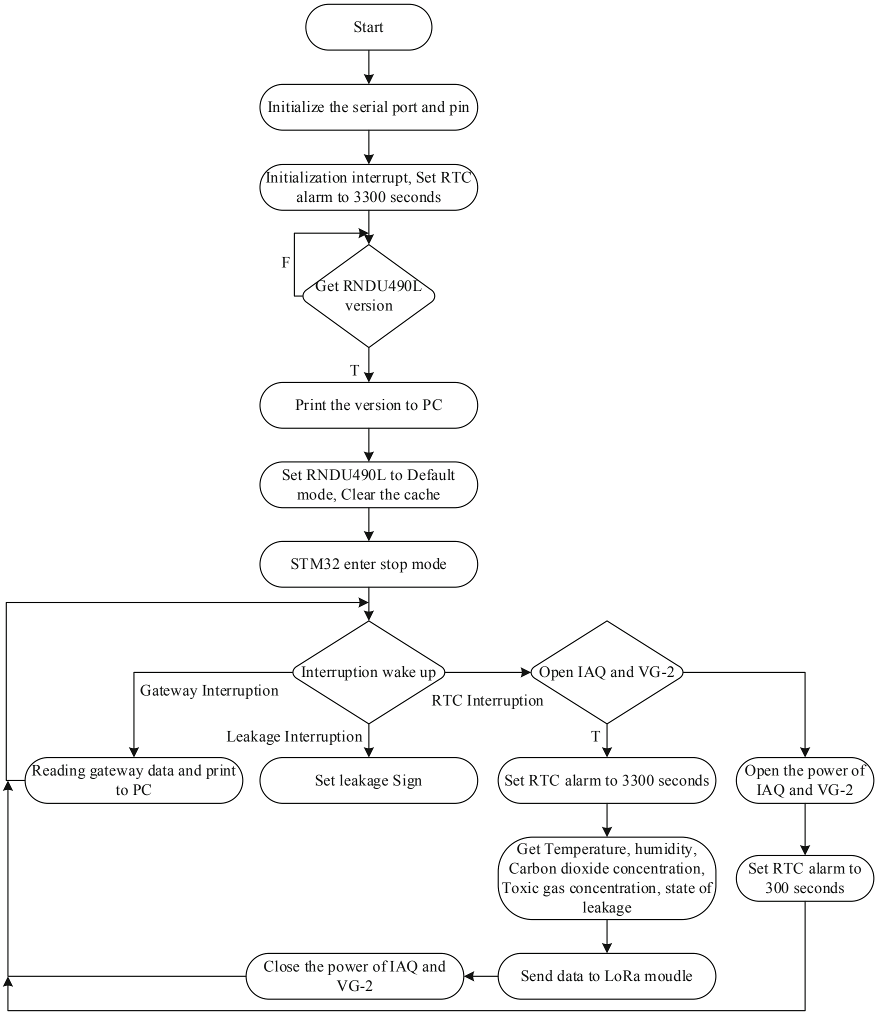


Fig. 8. Software flow chart.

The operation of software starts with a series of initialization, which consist of the initialization of three serial ports, input/output pin, delay function, interrupt function, iaq driver, and RTC clock.

The second serial port sends a command to RNDU490L to get the version, which verifies that RNDU490L is working properly. If the version is correctly obtained, execute the following procedure. Otherwise, the program delay 0.5 s, then send the command again.

Set RNDU490L as the default working mode. The default working mode means starting ADR (rate adaptive), setting the data rate to SF12, and sending power to 17 dBm.

STM32 enter stop mode.

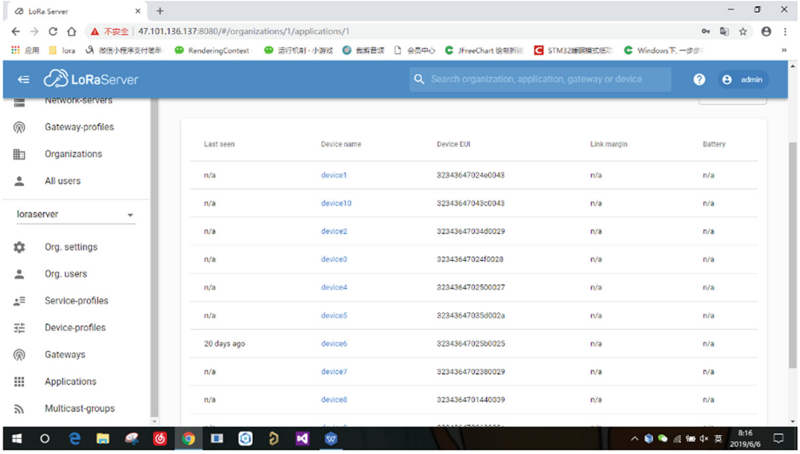
Exit low power mode by three interrupts: ① Timer interruption: determine whether the power of iaq and VG-2 are on. If not, turn on and set the RTC clock for 300 s next time. If it is turned on, set the RTC clock to 3300 s next time, then read the temperature, humidity, and the concentration of toxic gas and carbon dioxide. Send it to the LoRa server. If the toxic gas content exceeds the limit, send a signal to alarm; ② Interruption of LoRa module: when the gateway has downlink data, LoRa module will send a rising edge to trigger the interrupt. In this interrupt processing event, STM32 prints the downlink data received from the LoRa module to the PC. ③ VG-2 leakage interruption: set the water leakage flag bit.

4 Design and Implementation of LoRa Server and Monitoring App

The LoRa server manages gateways, LoRa nodes, and data transmission within LoRa network. The monitoring App receives data from LoRa server and conducts analysis, processing and usage. According to the system requirements, it is divided into four parts: receiving platform, data display module, manage node module and alarm display module. Meanwhile, in order to realize multi-node monitoring, two mechanisms are designed and implemented in the receiving platform. One is “thread programming” and the other is “data receiving and processing separation”.

4.1 LoRa Server Software

LoRa server software is based on the open source LoRa Server project, and the usage of OS is 64-bit ubuntu system (Fig. 9).



Last seen	Device name	Device EUI	Link margin	Battery
n/a	device1	32343647024e0043	n/a	n/a
n/a	device10	32343647043c0043	n/a	n/a
n/a	device2	32343647034d0029	n/a	n/a
n/a	device3	32343647024f0028	n/a	n/a
n/a	device4	3234364702500027	n/a	n/a
n/a	device5	32343647035d002a	n/a	n/a
20 days ago	device6	32343647025b0025	n/a	n/a
n/a	device7	3234364702380029	n/a	n/a
n/a	device8	3234364701440039	n/a	n/a

Fig. 9. The LoRa server Deployed on aliyun ubuntu platform.

After the deployment of LoRa server on aliyun ubuntu, it is necessary to open some ports of the cloud server, including the 1700 port which is used to communicate with the gateway through UDP, and the 8080 port which is used by Lora-app-server to provide web access services.

4.2 Monitoring App Design and Implementation

The function of monitoring App can be divided into four parts. Firstly, collect and store the uplink data. Secondly, display trends of various types of information. Thirdly, alarm when carbon dioxide concentration exceeds the threshold, or toxic gas concentration exceeds the threshold, or water leakage happens. Finally, manage node information [20] (Figs. 10 and 11).

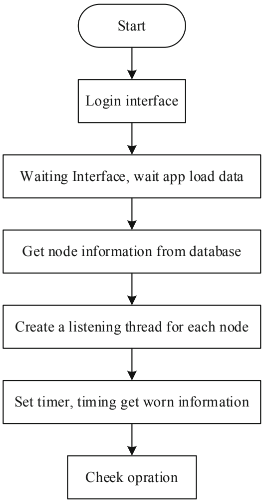


Fig. 10. Software flow chart.

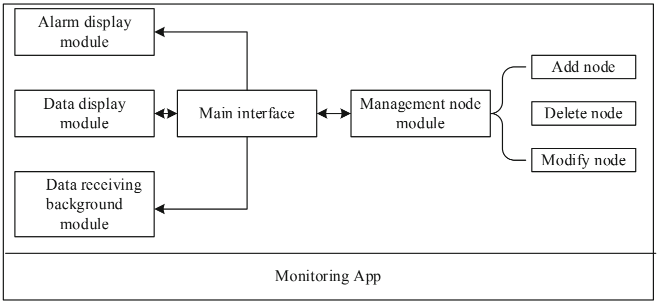
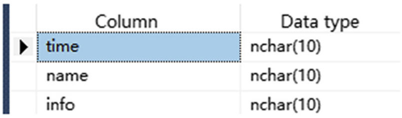


Fig. 11. Monitoring App functional structure.

Database. The database uses SQL server. We create a database named ‘server’ and a user. In ‘server’, there are three types of tables: node table, warning_log table, and each node’s table (Fig. 12).

column	Data type
name	nchar(10)
eui	varchar(50)

Fig. 12. Node table.

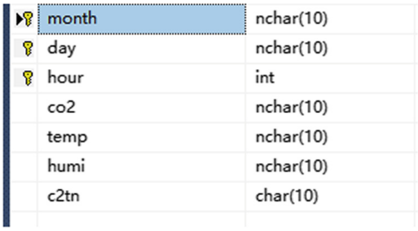


Column	Data type
time	nchar(10)
name	nchar(10)
info	nchar(10)

Fig. 13. Worning_log table.

The node table is used to store the name of the node and its corresponding eui. The node name is set by the user. Eui is a fixed value of each node, which is provided by the manufacturer (Fig. 13).

The Warning_log table is used to store details of each alarm, with columns ‘time’, ‘name’, and ‘info’. The column ‘info’ is used to indicate the pre-agreed event, and the alarm information can be shown by ‘check log’ button of the monitoring App.



month	nchar(10)
day	nchar(10)
hour	int
co2	nchar(10)
temp	nchar(10)
humi	nchar(10)
c2tn	char(10)

Fig. 14. Each node’s table.

Each node table is named according to the node name, for example, the table name of the Fig. 14 is “node 1”. Since the terminal program is designed to send uplink data every one hour, the columns of table are ‘month’, ‘day’, ‘hour’, ‘co2’, ‘temp’, ‘humi’ and ‘c2tn’ (toxic gas). The primary key of node table is (month, day, hour).

Receiving Platform. The Receiving Platform is used to acquire and store the terminal data. Due to the large number of terminals, two concurrency problems need to be taken into consideration in the design. Firstly, it is needed to solve how to promote the platform’s efficiency. Secondly, it is needed to solve how to receive data accurately.

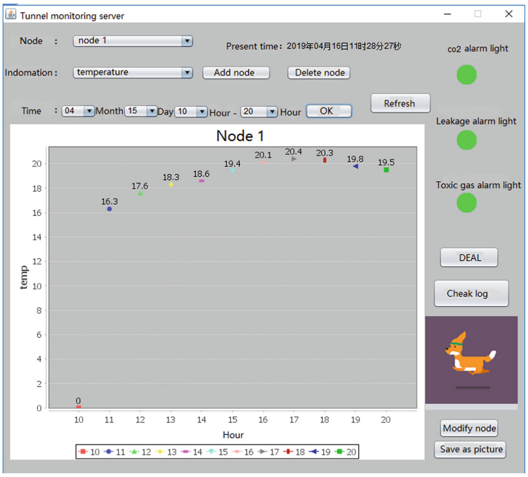
To solve the above problems, two mechanisms are proposed. One is “thread programming”. The other is “data receiving and processing separation” [21].

The “thread programming” mechanism means creating new thread listening to each node. When a terminal does not report information, the listening thread is blocked. It realizes that the receiving platform does not occupy the CPU resources of PC during the long period of terminal sleep.

The mechanism of “data receiving and processing separation” means storing all data into a buff array without doing any processing. The buff array has two variables, rcount and scount. Rcount is used to count data that has been processed, and scount is used to count data that has been stored. Create a timer to handle the buff array, judging the values of rcount and scount every 10 s. When rcount is greater than or equal to the value of scount, it is proved that no new data needs to be processed. Otherwise, start processing the buff array data from rcount until the scount, then update the values of rcount and scount.

Data Display Module

After selecting node name, information and time interval in the main interface, click “ok” button to display broken line graph of information change. Click the “save as picture” button to output the graph as an image and save it to the file system (Fig. 15).



Tunnel monitoring server

Node : node 1

Present time：2019年04月16日11时28分27秒

Indomation : temperature

Time : 04 Month 15 Day 10 Hour - 20 Hour

Fig. 15. The main interface displayed environment information.

Manage Node Module. The manage node module consists of the addition, modification and deletion of the node name and eui mapping.

5 System Test and Result Analysis

In this paper, three types of system tests are conducted, which consist of communication performance test, monitoring performance test and terminal power consumption test. The first test mainly tests the packet loss rate between LoRa terminal and gateway. The second test mainly tests whether the data uplinked by the LoRa terminal are consistent with the actual situation. The third test mainly measures the average power consumption of LoRa terminal.

5.1 Communication Performance Test

Communication distance has a great impact on the management of LoRa terminal, deployment costs and the performance of the system. In this test, the communication performance was tested in two experimental environments. Firstly, the gateway is deployed at the tunnel gate, and the LoRa terminal is placed deep in the tunnel. Secondly, the gateway and the LoRa terminal are deployed in an open environment [22].

In the first case, the LoRa terminal transmitted 100 uplink packets to the gateway. And the average communication success rate is 99%.

In the second case, the distance between gateways and the terminal are shown in Fig. 16.

Fig. 16. Terminal communication performance test topology.

The distance between the gateway 1 and terminal is 743 m. The terminal transmitted 50 uplink packets to the gateway 1. And average packet loss rate is 14%. Besides, the RSSI obtained from the gateway 1 fluctuates very little, its average value is −107 dbm (Fig. 17).

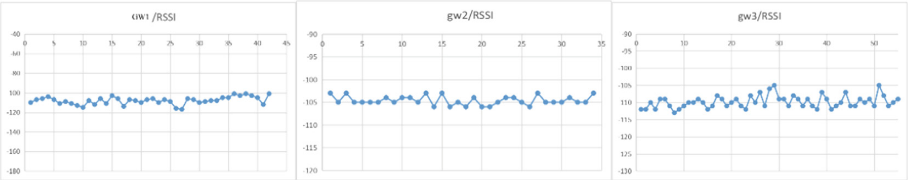


Fig. 17. Gateway 1, 2, 3's received signal strength fluctuates.

The distance between the gateway 2 and terminal is 821 m. The terminal transmitted 50 uplink packets to the gateway 2. And average packet loss rate is 20%. Besides, the RSSI obtained from the gateway 2 fluctuates very little, its average value is −104 dbm.

The distance between the gateway 3 and terminal is 606 m. The terminal transmitted 50 uplink packets to the gateway 3. And average packet loss rate is 2%. Besides, the RSSI obtained from the third gateway 3 fluctuates very little, its average value is −109 dbm.

The packet loss rate varies with the communication distance as shown in Fig. 18. And the packet loss rate apparently varies linearly with the change in communication distance.

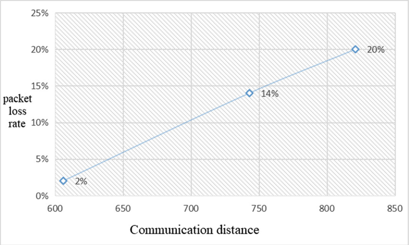


Fig. 18. Communication distance vs packet loss rate.

5.2 Monitoring Performance Test

Monitoring performance is tested to verify that the system can accurately monitor the tunnel environment.

Firstly, the system powered by battery was tested in the laboratory for 2 days, and the carbon dioxide concentration, temperature, humidity, toxic gas concentration and water leakage are monitored continuously. The temperature acquired is shown in Fig. 19.

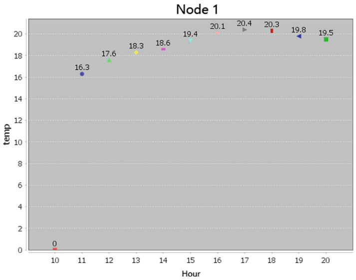


Fig. 19. Temperature curve under laboratory test.

In the tunnel test, the terminal is deployed to the actual tunnel, and collect tunnel environment every 10 s. Secondly, the system powered by battery was tested in the actual tunnel, and the tunnel information was collected every 10 s. Parts of the data are shown in Table 2. The average humidity was 72.18%, the average temperature was 17.94 °C, the average carbon dioxide concentration was 317 ppm (3.17% of the air), and the average toxic gas concentration was 125 ppb ($1.25 \times 10^{-3\%}$ of the air).

In Table 2, the first column ‘node eui’ indicates the hardware number of LoRa transmission module in LoRa terminal. The format of ‘time’ is ‘minutes-seconds’. ‘Humidity’ refers to the humidity in the air. Column ‘temperature’ indicates the temperature in

Table 2. The first ten groups of test data under pit test.

node eui	Time	Humidity	Temperature	CO_2	Tvoc	FCnt	Isleak
32343647034d0029	26–30	71.3	18.8	318	125	0	N
32343647034d0029	26–40	71.2	18.2	318	125	1	N
32343647034d0029	26–53	69.5	18.1	318	125	2	N
32343647034d0029	27–04	71	18.1	318	125	3	N
32343647034d0029	27–15	70	18.1	318	125	4	N
32343647034d0029	27–27	70.2	18.1	318	125	5	N
32343647034d0029	27–39	71.4	18.1	317	126	6	N
32343647034d0029	27–51	71.5	18.1	318	125	7	N
32343647034d0029	27–51	71.6	18.1	318	125	8	N
32343647034d0029	28–14	70.4	18.1	318	125	9	N

degrees Celsius. 'CO_2' refers to the concentration of carbon dioxide in ppm (parts per million). 'Tvoc' refers to the concentration of toxic gases in ppb (concentration per billion parts). 'FCnt' is the frame count between the gateway and LoRa terminal. 'Isleak', 'Y' is leakage and 'N' is no leakage.

5.3 Terminal Power Consumption Test

Only LoRa terminal power consumption needs to be considered. Therefore, the power consumption of LoRa terminal is the key factor to determine the working life of the system, and it is also the advantage of the tunnel environment monitoring system designed based on LoRa. It is of great significance to the promotion and application of the system. This test uses oscilloscope method. As shown in Fig. 20, the high precision resistor is connected in series between the LoRa terminal and the power supply. The voltage difference across the resistor in the three working stages is measured by an oscilloscope. According to ohm's law, the working current of the LoRa terminal can be calculated. The resistance value of the test resistor is 10 Ω.

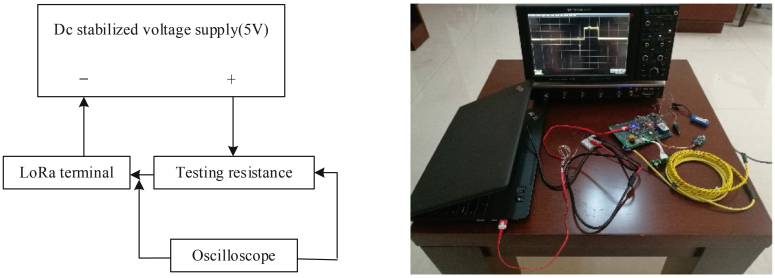


Fig. 20. Schematic diagram of oscilloscope method.

During the 55 min sleep stage, the voltage difference measured by the oscilloscope is 23 mV. According to Fig. 21, the voltage difference across the LED is 15 mV, so the voltage difference across the necessary circuit is 8 mV. That is to say that the system current consumption is 0.8 mA.

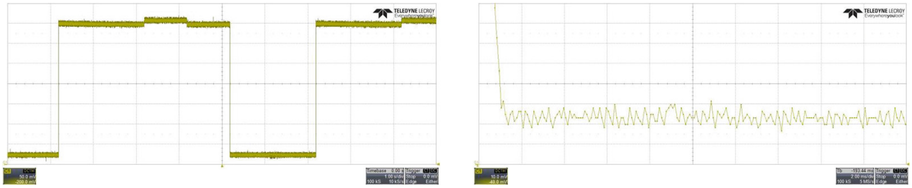


Fig. 21. The first stage is the voltage change of oscilloscope.

During the 5 min preheating stage, the voltage difference measured by the oscilloscope is about 400 mV. That is to say that the system current consumption is 40 mA (Fig. 22).

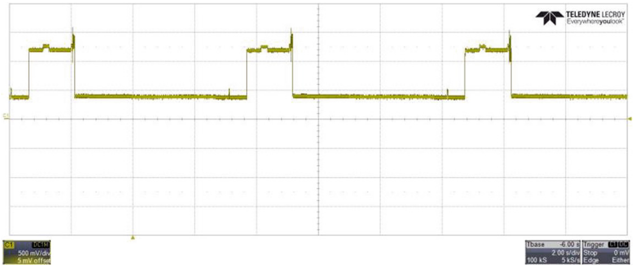


Fig. 22. The second and third stage voltage.

During the reading and transmission stage, the voltage difference measured by the oscilloscope is about 1.2 V. That is to say that the system current consumption is 120 mA.

Table 3. LoRa terminal power consumption.

Stage	Plate on the current
The first stage lasts 55 min	0.8 mA
The second phase lasts 5 min	40 mA
The third stage takes about 30 s	120 mA

Then the average current consumption in one period, can be represented by formula 1.

$$\mathrm{I_h} = \sum_{i}^{3} \frac{T_i}{T_h} \times I_i \tag{1}$$

T_i is the duration of working stage i, and T_h is 60 min which is the duration of one period. Substitute data from Table 3 into formula 1, the calculated I_h is 5.06 mA. The capacity of two AA batteries is about 3000 mAh, which means the LoRa terminal power by two AA batteries in this system can run for up to 592 h.

6 Conclusion

Compared with the traditional IoT technologies such as ZigBee, LoRa technology performs better in terms of coverage range, connection number, and energy consumption, etc. It is foreseeable that the LoRa technology will certainly make a tremendous promotion in the field of tunnel environment monitoring. In this paper, A Lora-based tunnel environment monitoring system is designed and implemented. The system architecture is described, and the detailed design and implementation of hardware and software are also described. The experimental results show that the system has good network coverage capacity and communication reliability, can accurately monitor the tunnel environment and have low power consumption. This system has good promotion and application value.

References

1. Tao, X., Yan, Z., Yong, M.: Underground tunnel environment wireless monitoring system. Inf. Technol. (8), 46–49 (2017)
2. Liu, W., Li, Q.: Research on mine explosion-proof monitoring system based on ZigBee. Autom. Instrum. (4), 26–29 (2009)
3. Yun, H.: Research on mine personnel sign monitoring system based on ZigBee. Chang'an University, Xian (2011)
4. Feng, W.: LoRa Internet of Things technology and application. Telecom World (2), 91–92 (2017)
5. Guo, C.: Discuss the advantages and applications of short distance wireless communication technology. China New Telecommun. **19**(5), 94 (2017)
6. Lee, K., Lee, J., Yi, Y.: Mobile data offloading: how much can WiFi deliver. In: International Conference, vol. 21. ACM (2010)
7. Lin, S.: Characteristics and application of GPRS technology. Telecommun. Technol. (3), 2–5 (2002)
8. Aref, M., Sikora, A.: Free space range measurements with Semtech Lora^{TM} technology. In: 2014 2nd International Symposium on Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS-SWS) (2014)
9. Won, Y.-Y., Yoon, S.M., Seo, D.: Optical access transmission with improved channel capacity using non-orthogonal frequency quadrature amplitude modulation. Opt. Quant. Electron. **49**(2), 1–13 (2017)
10. Akhlaghi, M.: High-speed optical FSK demodulator using plasmonicnano bi-dome. Optik Int. J. Light Electron Optics **127**(19), 8030–8035 (2016)
11. Chung, Y.W., Chung, M.Y., Sung, D.K.: Adaptive algorithm for mobile terminal power on/off state management. Comput. Commun. **24**(14), 1411–1424 (2001)
12. Zhang, H., Kang, W.: Design of the data acquisition system based on STM32. Procedia Comput. Sci. **17**, 222–228 (2013)

13. Mei, G., G-H., Cai, L.: Design of water leakage detection device in computer room based on Hetai single chip microcomputer. Electron. Instrum. Cust. **24**(3), 34–40 (2017)
14. Patel, N.G., Panchal, C.J., Makhija, K.K.: Use of cadmium selenide thin films as a carbon dioxide gas sensor. Cryst. Res. Technol. **29**, 1013–1020 (1994)
15. Xiao, B., Jian, F., Fan, L.: Design and development of remote indoor air monitoring system based on Yeelink. Comput. Program. Skills Maint. **10**, 26–27 (2015)
16. Arshak, K.I., Twomey, K.: Investigation into a novel humidity sensor operating at room temperature. Microelectron. J. **33**(3), 213–220 (2002)
17. Xing, F., Fei, T., Yan, L.: Design of low power consumption wireless Internet of Things terminal power management system. Semicond. Technol. **43**(12), 883–886 (2018)
18. Yu, Z., Rui, S., Kun, W.: Development of greenhouse environment monitoring system based on Proteus and Keil software. Trans. Chin. Soc. Agric. Eng. **28**(14), 177–183 (2012)
19. Ning, L., Qiong, C., Lei, Y.: Design of serial communication display system based on STM32 minimum system. Ind. Control Comput. **30**(8), 33–36 (2017)
20. Lin, Y.: Student achievement information management system based on Java Web. Jilin University (2015)
21. Hua, W., Yun, X.: Java multithreading mechanism and its application. Comput. Mod. **1**, 1–6 (2000)
22. Tao, S.: Computer application system performance test technology and application research. Electron. Test (8), 69–70+66 (2019)