Hindawi
Scientific Programming
Volume 2021, Article ID 6070664, 15 pages
https://doi.org/10.1155/2021/6070664

Research Article

Design and Implementation of Wireless Environment Monitoring System Based on STM32

Qu Xilong,[1] Liu Shengzong,[1] Fu Sha,[1] He Hong,[2] Hu Ying,[2] and Xiao Leyi[1]

[1]*Hunan Provincial Key Laboratory of Finance & Economics Big Data Science and Technology, School of Information Technology and Management, Hunan University of Finance and Economics, Changsha 410205, China*
[2]*School of Computer and Communication, Hunan Institute of Engineering, Xiangtan 411104, China*

Correspondence should be addressed to Xiao Leyi; xiaolyttkx@163.com

Received 22 April 2021; Revised 14 July 2021; Accepted 27 September 2021; Published 28 October 2021

Academic Editor: Zhu Xiao

This paper first designed the overall scheme of a wireless-based intelligent environment monitoring system; this system is comprised of the following four parts: server part, client part, slave computer part, and master computer part. The master computer is equipped with a wireless transmission module and an LCD module. Slave computers are equipped with temperature-humidity sensing modules and wireless transmission modules. The server side has two major interfaces: data display and server establishment. An Android phone as the client can realize the communication between the mobile APP and server through network communication to obtain environmental data in real time. The hardware part of the system consists of wireless transmission module, temperature and humidity module, smoke module, lighting module, flame module, and LCD module. The software implementation part mainly includes the design and implementation of serial communication, the design and implementation of wireless communication, the design and implementation of collection of data acquisition in each sensor module, and the design and implementation of network communication as well as the design and implementation of display unit. With the cooperation of these four parts, we can automatically sense and control the state of the environment to realize human-computer interaction.

1. Introduction

With the science advancement and technology popularization, intelligent devices enjoy more and more widespread application, and problems such as the accuracy of environmental issues have been aroused therefrom [1, 2]. The previous environmental monitoring was carried out by means of manual monitoring, which has many shortcomings from the present point of view, including low efficiency and unscientific management system [3–6]. It was not conducive to the full utilization of resources and scientific management. Moreover, it was not difficult to find from the major accidents in these years that many were caused by human factors [7–10].

For example, greenhouses, industrial factories, and other places have higher requirements for their surrounding environment [11, 12]. Although many greenhouses and factories are also equipped with corresponding heating, humidifying, and ventilation equipment, these types of equipment are basically manually operated [13, 14]. Under the circumstances, once there is a large workload of equipment inspection and maintenance, it will cause problems such as high labor intensity. In addition, this method is not able to achieve accurate and real-time control of the environment, having become incapable of meeting the needs of the rapid development of today's society [15–17]. Therefore, the development of a wireless environmental monitoring system that integrates environmental data collection, transmission, and display will greatly reduce the labor intensity of operators and improve the accuracy of environmental monitoring [18, 19].

Now, traditional environmental monitoring technology has been widely used in the production of various industries in China, having developed into a relatively complete system

[20]. Intelligent environmental monitoring technology, which can achieve centralized monitoring of multiple environmental factors, is also applied in some developed countries [21, 22]. But its high cost makes it difficult to popularize. In recent years, many research institutes in China have studied intelligent environmental monitoring technology, and the country has also given great support [23–25]. Judging from the current domestic and foreign environments, the wired environment monitoring technology has been relatively mature, but its application range is very limited and has obvious limitations [26–28]. Therefore, it is necessary to design a system that can wirelessly send these data to the host for processing [29, 30].

This system is to design a wireless environment monitoring device based on environmental data collection, transmission, and display. Environmental monitoring based on wireless technology has sufficient development space and application prospects, and it can be widely used in the production and development of industry and agriculture. Therefore, the development of such an intelligent environmental monitoring system will be able to be widely used to meet the needs of market development.

The remainder of this paper is organized as follows: Section 2 introduces the related work. Section 3 introduces the overall design of this system. Then, the design and implementation of system hardware circuit are elaborated in Section 4. Section 5 gives the design and implementation of system software. And system function debugging is executed in Section 6. Finally, a conclusion is given in Section 7.

2. Related Work

Curran and Norrby used RFID to locate the items or people in the indoor environment [31]. But RFID is high cost and security is not strong enough, which limits its wide application. Yang et al. extended the process-oriented structured system analysis on the existing intelligent transportation architecture to establish an intelligent transportation system based on the new generation wireless communication technology, and further in-depth analysis will be performed to discuss the required additional capabilities of network, services, and application support in the view of user, function, information, connection, and communication [32]. Dey et al. considered a home-based wireless ECG monitoring system using ZigBee technology [33]. But such systems are often designed in isolation, with no consideration of existing home control networks and smart home solutions. Liu used multisensor data fusion technology, ARM technology, ZigBee technology, GPRS, and other technologies and studied and developed an intelligent environmental monitoring system [34]. But the cost of the GPRS communication module is high. A lot of research on wireless sensor networks optimized IIR broadband digital differentiators and integrators by using SALP group algorithm, automatically detected network interference and intrusion, and offloaded vehicular task by using the game-theoretic method [35–37]. The current wireless communication technology is more and more widely used. Therefore, using wireless technology to design a low-cost, stable, and reliable environmental detection system is of great significance.

3. Overall Design

After analyzing the system requirements, an integrated communication method is adopted, including slave computer, master computer, server, and client. STM32 is used as the master controller, QT is used to design the human-machine interactive interface, and TCP network communication is adopted to realize the communication with the client, so that the master computer, server, and client have the function of observing the environment. The system framework is as shown in Figure 1.

The slave computer of this system realizes data acquisition and transmits the data to the master computer through the wireless module. The master computer transmits the data to the server through the serial port, and the client obtains the server data through network communication. Various modules work in coordination to achieve real-time monitoring of the environment. This system mainly completes the design of the following four parts:

(1) Slave computer part

Temperature-humidity sensor module, photosensitive sensor module, smoke sensor module, and flame sensor module are used to collect data on surrounding environmental parameters.

(2) Master computer part

NRF24L01 wireless transmission module, digital tube module, and LCD screen module are adopted, which are connected to the slave computer through wireless technology to acquire data through wireless technology, so as to display these acquired data on the digital tube and LCD screen.

(3) Server

A computer is used as the server to realize communication with master computer through serial port. A QT interface is established to display environmental data and realize network communication with the client.

(4) Client

An Android phone is used as the client terminal to establish a client terminal display interface and implement network communication with the server.

This system will be applied in the wireless monitoring of the environment. It adopts the communication method integrating slave computer, master computer, server, and client so that users can monitor online through the client and PC to ensure the safe and high-quality production of the environment such as greenhouses and factories. This system is built on the basis of wireless, serial, and network communication, and various modules are interconnected through different communication methods to achieve human-computer interaction. Based on these functions, this system adds functional modules to the slave computer, such as temperature-humidity sensor, photosensitive sensor,

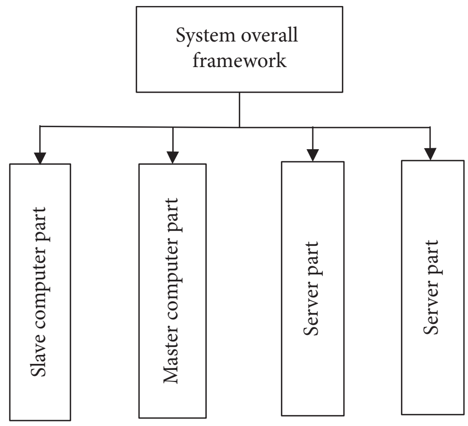

Figure 1: Overall framework of the system.

smoke sensor, and flame sensor as well as NRF24L01 module to achieve real-time monitoring of the environment.

The research process of this system is divided into the following steps:

(1) Hardware environment of the overall design
(2) Building an overall software platform
(3) Collection of environmental data by sensors
(4) Wireless transmission and display of data
(5) Data transmission realized by serial communication
(6) Data transmission between client and server realized by network programming
(7) Design and implementation of QT graphical interface
(8) Test for system operation stability

3.1. Overall Design Scheme for the System. After the analysis of the research topic, an integrated communication method of slave computer, master computer, server, and client is adopted. STM32 is used as the master controller, QT is employed to design the human-computer interaction interface, and the communication between the server and the client is realized by relying on TCP network communication, so that the master computer, the server, and the client have the function of real-time monitoring of the environment. The system is mainly designed to realize the following functions:

Data acquisition: Temperature-humidity sensor module, photosensitive sensor module, smoke sensor module, and flame sensor module are used to collect data from the surrounding environment.

Wireless transmission: Use NRF24L01 module to send and receive data, so as to realize communication between slave computer and master computer.

Fire protection function: Monitor the fire condition in the environment through flame and smoke sensors.

Alarm unit: The acoustooptic alarm device is adopted. By setting the threshold of environmental data, the alarm signal will be sent off once the relevant data exceed the threshold value.

Human-machine interaction interface: The server obtains and displays the environmental data from the master computer through the serial port, while the client obtains and displays the data from the server through the network communication.

3.2. Features of the System. This system uses a wireless transmission platform plus various sensor modules and integrates with serial communication and TCP network communication to form an intelligent wireless environment monitoring system [38–40]. It is used to substitute the original manual field monitoring work and ensures that users can obtain environmental data in real time through network transmission. Therefore, the staff can monitor and manage the environment in real time through the client APP, so that one person can monitor multiple environmental conditions at the same time, thereby greatly improving work efficiency.

This system is divided into four parts in terms of structure, namely, the slave computer part, the master computer part, the server part, and the client part.

(1) Slave computer part: Adopting photosensitive sensor, smoke sensor, flame sensor, and temperature-humidity sensor to realize the collection of environmental data and realize the wireless transmission of data through the NRF24L01 wireless module.
(2) Master computer part: NRF24L01 wireless module is used to realize data communication with slave computer, display data through digital tube and LCD screen, and send data to server through serial port.
(3) Server part: A computer is adopted as the server to realize communication with master computer through serial port, and QT interface is established to display environment data and realize network communication with client.
(4) Client part: An Android phone is adopted as the client terminal to achieve network communication with the server and obtain and display environmental data in real time.

This system uses a wireless transmission platform plus various sensor modules and integrates with serial communication and TCP network communication to form an intelligent wireless environment monitoring system, which has the following advantages compared with the traditional manual field monitoring mode:

(1) This system ensures continuous work for a long time and can work in various environments and even under special extreme conditions (including closed environment).
(2) The environment information can be controlled in real time, and comprehensive monitoring can be planned according to different parameter requirements so as to ensure that there is no omission or repetition in monitoring.
(3) This system can realize wireless transmission, and the client and server can communicate with each

other through the network. The master computer can wirelessly receive the data of the slave computer. As a result, working staff can remotely monitor one or more similar environments for convenient and efficient management.

This system uses multiple NRF24L01 wireless modules, which are installed on all master and slave computers, so as to realize the wireless communication between master and slave computers. This system adopts the network communication based on TCP protocol, which is characterized by being able to realize remote data acquisition in the local area network. Finally, this system employs STM32F103ZET6 as the core processing chip. The chip is characterized by high stability, high performance, and large memory, and thereby it is suitable for processing complex data.

Based on the system application requirements and characteristics, it is required to first make a general plan for the system, including the design of the system architecture, wireless transmission, serial communication, and selection and design of network communication. The system takes STM32-based wireless environment monitoring system as the research object, establishes and selects suitable object models, and finally designs a physical model of STM32-based wireless environment monitoring system through the design of software and hardware.

3.3. Key Technical Issues and Main Technical Indicators

3.3.1. Wireless Transmission Part. The wireless transmission part uses the NRF24L01 sensor and sends the data collected by the slave computer to the master computer for processing through the SPI protocol, thereby realizing the data communication between the master computer and the slave computer and then realizing real-time display of the environmental data and intelligent alarm function through the analysis of the received data. The SPI protocol works in a master-slave mode. This system uses one master one slave mode. The master device determines the slave device through the enable signal so that the master-slave devices can be connected and the data transmission can be synchronized by the clock signal.

3.3.2. Sensor Technology

(1) Photosensitive sensor module: This system uses the built-in photosensor of the demo board STM32 to determine the light value in combination with its internal analog-to-digital conversion.

(2) Temperature-humidity sensor module: This system uses a DHT11 digital temperature-humidity sensor to collect data of temperature and humidity to obtain the required data through Time Division Multiplexing (TDM) performed according to the transmission protocol.

(3) Flame sensor module: This system uses the flame sensor module YL-38 to detect the fire source. This module can detect whether there is an open flame around the system and then output high electrical level and low electrical level. The slave computer can determine if the warehouse is at risk of fire by detecting the high electrical and low electrical level.

(4) Smoke sensor module: This system uses the smoke sensor module MQ-2 to monitor the smoke concentration. It can detect whether the smoke concentration in the environment exceeds the standard. The module is equipped with an adjustable potentiometer, which can adjust the output sensitivity of TTL, so as to more precisely determine whether there are hidden dangers in the environment by combining with other module values in the environment.

(5) Wireless transmission module: This system uses the NRF24L01 wireless transmission module and realizes wireless communication based on SPI protocol. Users can easily realize wireless communication functions.

3.3.3. Network Communication Technology. Network communication technology is a very important component of this system. This system uses the network integrated by the QT development platform to build servers and clients by applying the TCP protocol so as to ensure that multiple clients can obtain the current environmental data. The TCP protocol can provide not only connection-oriented reliable services but also full-duplex reliable channels. TCP protocol requires such a connection process that the connection between the client and the server shall be established first before transmitting data, and the client will actively release after the data transmission is completed. Through the network programming interface provided by the socket, a special file descriptor (socket descriptor) is obtained by using the socket, and the subsequent steps of network programming based on the TCP protocol will be implemented through the socket descriptor.

4. Design and Implementation of System Hardware Circuit

This system is divided into server part, client part, slave computer part, and master computer part. The server and client terminals realize data transmission through TCP-based network communication; the master computer is composed of LCD display and acoustooptic alarm device; the slave computer part consists of temperature-humidity sensor, smoke sensor, photosensitive sensor, flame sensor, and NRF24L01 wireless module. The master computer communicates with the slave computer through the NRF24L01 wireless module, and the master computer communicates with the server through the serial port. The system integrates acquisition, display, and transmission and has a high degree of intelligence, which reduces the labor intensity of operators and improves the accuracy of environmental monitoring.

4.1. Slave Computer Part

4.1.1. Slave Computer MCU-STM32F103

(1) Introduction. STM32F1 series enhanced chips with Cortex-M3 as the kernel is used by this system as the

controller MCU of slave computer. Cortex-M3 is a 32-bit RISC processor with a built-in high-speed memory and a working frequency of 72 MHz. It has a rich I/O port and peripherals. It is capable of operating in the temperature range of −40°C to +105°C, and it has an unprecedented advantage in the 32-bit microcontroller market.

(2) Function. The chip is mainly responsible for the normal coordination of various sensor modules of the slave computer. The data collected by the sensor is simply processed, analyzed, and then sent to the master computer through the NRF24L01 wireless transmission module and assists the master computer to control the entire system operation. The lower part of this system uses the minimum boar of STM32F103ZET6.

4.1.2. Smoke Sensor Module

(1) Introduction. This system uses an MQ-2 smoke sensor to detect the smoke concentration in the environment. This sensor has a high sensitivity to gas and so on, with a long service life and reliable stability. The principle is that when flammable gas is present in the environment, the electrical conductivity of the sensor increases with the increase of the concentration of the flammable gas in the air, thereby determining the concentration of the flammable gas in the current environment. Use a simple circuit to output high electrical level and low electrical level, and set a threshold for exceeding the standard. When the output is of a high level, it means that the concentration exceeds the standard.

(2) Function. MQ-2 in this system is responsible for detecting whether the smoke concentration in the environment exceeds the standard.

(3) Working Principle. The signals output by most sensor modules are relatively weak, and there is no exception for smoke sensors. Therefore, the output signals need to be processed through a set circuit to meet the processor's requirements for input signal detection. This system uses a noninverting proportional amplifier circuit to reduce expansion on hardware and achieve the highest cost performance.

4.1.3. Flame Sensor Module

(1) Introduction. This system uses a flame sensor module YL-38 to detect the fire source in the environment. When a fire source exists in the environment where the sensor is located, the sensor module will send a high-level signal to the main control chip. It can monitor the infrared light with the wavelength range of 700 nm–1000 nm and detection angle of 60 degrees, and the value is inversely proportional to the intensity of external infrared light.

(2) Function. In this system, the flame sensor is responsible for detecting the flame in warehouse.

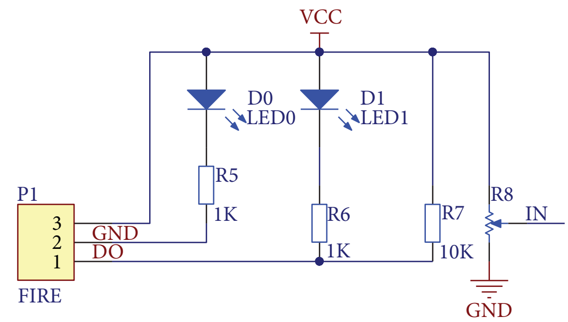

Figure 2: Schematic diagram of the flame sensor.

(3) Working Principle. The flame sensor module is equipped with a far-infrared flame probe, which can convert the change in the intensity of external infrared light into the change in current. The high level and low level of the digital interface DO can reflect whether the ambient flame spectrum exceeds the set threshold so that the processor can judge whether there is fire condition in the current environment only by checking the high level and low level of the digital interface DO. If it exceeds the set threshold, it will output high level, and the switch indicator will be lighted. Its schematic diagram is as shown in Figure 2.

4.1.4. Temperature-Humidity Sensor Module

(1) Introduction. This system is equipped with a temperature-humidity sensor module DHT11 on the slave computer to collect temperature and humidity information in the environment. It is a temperature-humidity composite sensor with a calibrated digital signal output. It is designed with a single bus data format, which can complete data communication every four milliseconds. This string of data is divided into a decimal part and an integer part. The specific data format is as follows: the first eight digits represent the integer part of the humidity, and the next eight digits represent the decimal part of humidity. The further next eight digits represent the integer part of the temperature, the following next eight digits represent the decimal part of the temperature, and the last eight digits represent the checksum data. This system is only accurate to the integer part of temperature and humidity. The physical object is as shown in Figure 3.

(2) Function. The temperature-humidity sensor module is responsible for collecting ambient temperature and humidity in this system.

(3) Working Principle. The module includes a resistive humidity sensing element and an NTC temperature measuring element and is connected to a high-performance 8-bit microcontroller. The sensing layer of the resistance humidity sensor will be interfered with by chemical vapor, and the diffusion of chemical substances in the sensing layer may cause the measurement value to drift and decrease the sensitivity. The hardware connection diagram is shown in Figure 4.

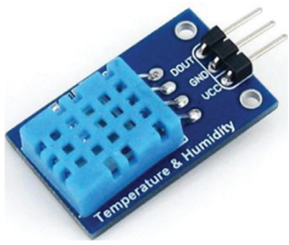

Figure 3: Physical picture of temperature-humidity sensor.

4.1.5. Photosensitive Sensor Module

(1) Brief Introduction. This system uses the photosensitive sensor that comes with the demo board STM32 to obtain the light illumination value by combining with its internal ADC.

(2) Function. The photosensitive sensor module in this system is responsible for the function of collecting ambient light illumination.

(3) Working Principle. This module is mounted on the master computer, reads the voltage value of the photosensitive sensor (LS1) through channel 6 (PF8) of ADC3, and converts it into a light intensity value of 0–100. The brighter the light on the module is, the larger the value could be; the darker the light on the module is, the smaller the value could be. Therefore, the change in light intensity can be viewed by blocking LS1 and illuminating LS1 with light during the experiment. The hardware connection diagram is as shown in Figure 5.

(1) Brief Introduction. This system uses multiple wireless modules, which are, respectively, mounted on the master computer and the slave computer to realize the wireless communication between the master computer and the slave computer. The wireless module NRF24L01 works in the 2.4 GHz–2.5 GHz ISM frequency band, uses FSK modulation, and integrates the Enhanced Short Burst protocol. Hardware Schematic Diagram of NRF24L01 Wireless Module is as shown in Figure 6. This module can not only realize point-to-point communication but also implement single-point-to-multipoint (up to one-to-six) communication. Its communication speed can reach 2 Mbps. This module is simple and easy to learn. Users can easily implement the function of wireless communication only by reserving 5 GPIOs and 1 interrupt input pin for the microcontroller system. Most demo boards have assigned pins to this module, and some of them are designed with related drivers, so it boasts of very good cost performance.

(2) Function. The wireless module in this system realizes the wireless communication between the master computer and the slave computer and is responsible for the data transmission between the two.

(3) Working Principle. The master computer and the slave computer are, respectively, equipped with wireless modules

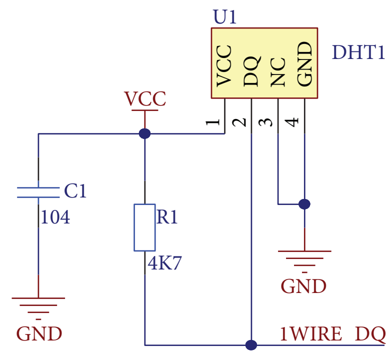

Figure 4: Hardware schematic diagram of temperature-humidity sensor.

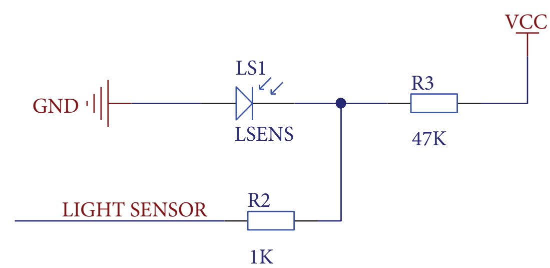

Figure 5: Hardware schematic diagram of photoelectric sensor.

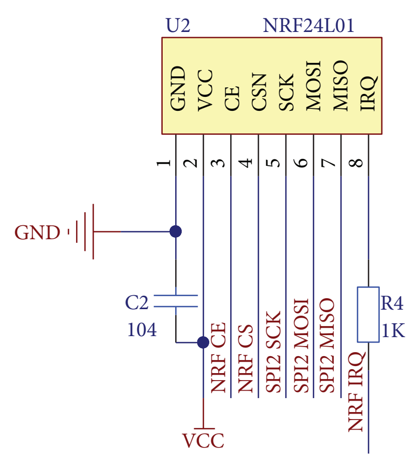

Figure 6: Hardware schematic diagram of NRF24L01 wireless module.

and configured with different modes. In this system, the slave computer is configured with data transmission mode, while the master computer is configured with data reception mode. This module uses the SPI protocol for communication, which just requires connection and data read/write in accordance with the standard SPI protocol.

4.2. Master Computer Part

4.2.1. Master Computer MCU-STM32F103. The MCU is mainly responsible for receiving the data collected by the sensor modules transmitted by the slave computer, configuring the acoustooptic alarm device, combining with the data to implement the alarm function, and sending the data to the server through the serial port. This section uses the STM32F103ZET6 battleship board.

4.2.2. NRF24L01 Wireless Module. The master computer part of this system uses the same model of the NRF34L01 wireless module as the slave computer, of which the wireless module is used to send data, and the master computer is used to receive data.

4.2.3. LCD Module

(1) Introduction. This system uses the built-in TFTLCD liquid crystal module of the demo board. TFT is a thin film transistor. Each pixel of the liquid crystal display is equipped with one TFT. The LCD module is designed with the size of 2.8 inches, the resolution of 320 ∗ 240, and a 16-bit true color display (RGB565: 5 red, 6 green, and 5 blue). Different colors can be displayed through RGB descriptions, and its interface is designed with a 16-bit 80-parallel port. This LCD screen can greatly improve the image quality.

(2) Function. This module displays the data of the slave computer in this system. The hardware connection diagram is as shown in Figure 7.

4.3. Server Part and Client Part. This system uses a laptop as the server, and this system uses an Android phone as the client.

5. Design and Implementation of System Software

Software design is the process of developing an appropriate implementation plan based on the requirements. As the macroidea of the whole system, it reflects the entire software to a certain extent. The software development of the environmental monitoring system based on wireless transmission is divided into four parts: software design and implementation of data acquisition terminal on the slave computer, software design and implementation of data processing on the master computer, software design and implementation of the server, and software design and implementation of the client. Each part needs to be designed with a corresponding driver. The software structure of the system is as shown in Figure 8.

This system uses multiple types of sensor modules to collect environmental information (temperature-humidity, smoke concentration, etc.), so it is necessary to set a corresponding analysis program for each sensor. The following is the introduction to the configuration of each sensor.

5.1. Temperature-Humidity Sensor Module. This system is connected to DHT11 temperature-humidity sensor module by pin No. 11 of GPIOG and realizes the control. DHT11 is a digital sensor that collects the data of temperature and humidity. Designed with a single bus data format, DHT11

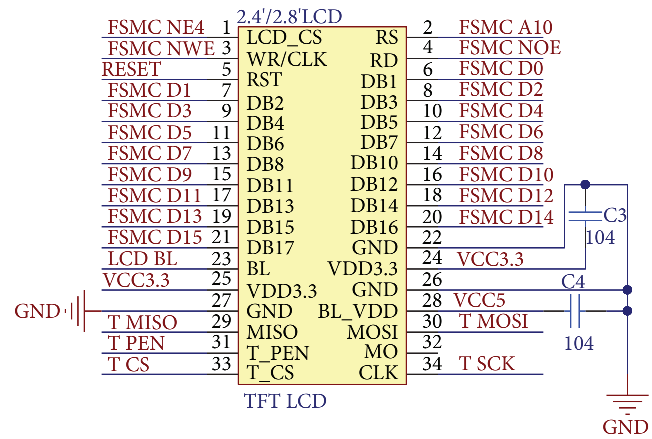

Figure 7: Hardware schematic diagram of LCD.

can complete data communication every four milliseconds. The specific data format is as follows: the first eight digits represent the integer part of the humidity, the next eight digits represent the decimal part of the humidity, the further next eight digits represent the integer part of the temperature, the following next eight digits represent the decimal part of the temperature, and the last eight digits represent the checksum data. The checksum data is the sum of the first four bytes. Using the checksum data can effectively ensure the accuracy of data transmission.

The specific process of temperature-humidity data collection is as follows: after the processor sends the start signal, the temperature-humidity module starts to work and sends out a response signal while sending a data packet representing the temperature and humidity. After receiving this part of the data, the processor can choose to use the specific part of the data according to users' requirements and conducted conversion on related data, such as temperature-humidity data with decimals and temperature-humidity without decimals.

The flame sensor module is controlled by No. 12 pin of GPIOA, and the IO port is set as the input mode. The smoke sensor module is controlled by No. 11 pin and No. 15 pin of GPIOA and converts the analog value into digital value through the AD chip. Therefore, the IO port is also set as input mode. The photosensitive sensor module is controlled by No. 8 pin of GPIOF and converts the analog quantity into digital quantity through the AD chip, so the IO port is set as the input mode at the same time.

5.2. Design for Wireless Transmission Driver. This system adopts wireless transmission technology to realize the data communication function between the master computer and the slave computer. It is required to first collect the data of each module, then perform quantitative processing on the data, and finally transmit the data.

5.2.1. Data Collection and Integration. This system needs to collect and integrate the following data, including temperature, humidity, flame status, smoke concentration, and light intensity.

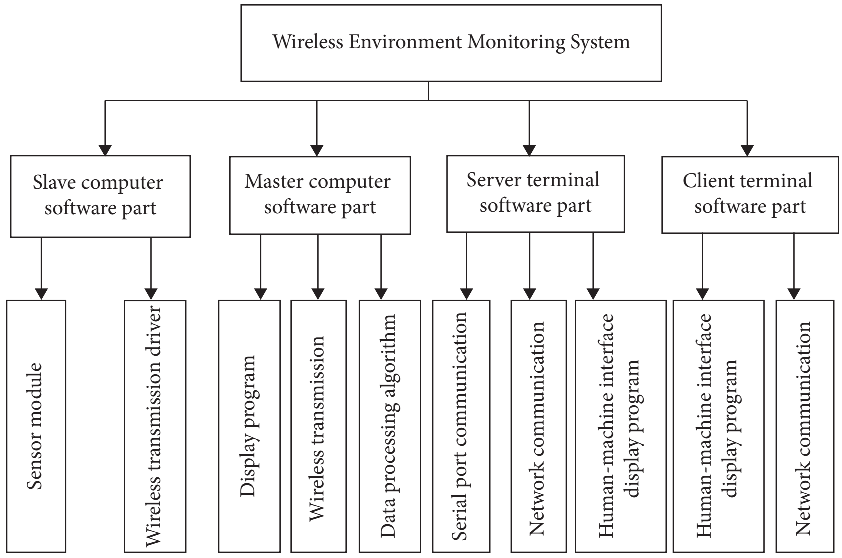

FIGURE 8: Structure diagram of system software.

It is realized by calling the data collection functions of each sensor module, including Get_SensorData (), DHT11_Read_-Data (& temperature, & humidity), and Lsens_Get_Val (). Then, the data are sent to the server through printf ().

5.2.2. Configuration of SPI Protocol. This system uses the NRF34L01 wireless transmission module to send data and adopts SPI communication protocol to realize communication. This system will, respectively, configure the STM32 SPI interface as the master-slave mode to achieve the initialization and read and write operations of the upper and slave computers. The initialization and read and write operations of the SPI interface can be realized through programming so that data communication can be implemented through NRF24L01.

5.2.3. Configuration of NRF24L01 Wireless Module. This system uses a wireless module in both the master computer and the slave computer. Data is sent from the slave computer to the master computer. Therefore, the slave computer is configured as the sending mode and the master computer is configured as the receiving mode.

5.3. Design of Serial Communication Driver. This system is designed to connect the master computer and the server through a serial port. Therefore, serial port communication is used to achieve communication between the master computer and the server. The serial port can be used to realize long-distance communication. As asynchronous communication, its most important parameters include the baud rate, data bit, stop bit, and parity bit. Therefore, to achieve serial communication between the master computer and the server, the user needs to configure the matches related parameters to the communication port. These parameters are described as follows:

Baud rate: the number of bits transmitted per second (9600, 115200, etc.)

Parity bit: error check method in serial communication, which can indicate the accuracy of the transmitted data

Stop bit: the last bit of a single packet, which indicates the end of the packet

Data bit: a parameter that measures the actual data bits in a communication

The master computer of this system realizes the serial transmission of data by calling the printf () function. The following code initializes the serial port and configures the printf () function.

5.4. Network Communication Driver Design. This system uses TCP protocol to realize network communication so that communication can be realized between mobile phone client and computer server. The TCP/IP protocol has been implemented in the protocol stacks owned by mobile phones and computers and can be used quickly and easily. In addition, TCP is a connection-based transmission protocol, by which data can be transmitted more stably and reliably. To use the TCP protocol, the users first need to establish a socket to support the TCP/IP protocol and indicate specific applications.

This system configures TCP programming for the server and client respectively. The programming idea is as shown in Figure 9.

Socket is widely used in network communication. Socket can identify a specific application because it contains the information necessary for network communication: IP address (local; remote), protocol port number (local; remote), and communication protocol (TCP, UDP, etc.). The connection process can be divided into the following three steps:

(1) Server's monitoring: Monitor the network status in real time and wait for connection requests of the client.

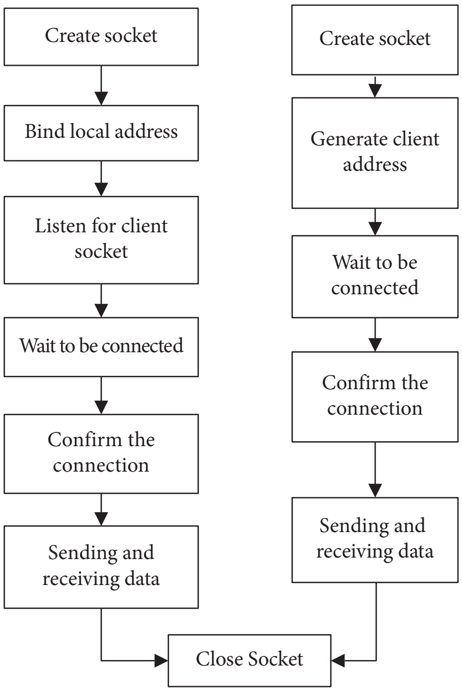

Figure 9: The programming idea.

(2) Client's request: The client's socket makes a connection request.

(3) Connection confirmation: After the server socket monitors the connection request of the client socket, it responds to the request of the client socket and establishes a new process.

The configuration procedure for establishing a connection by using socket is in Appendix A).

5.5. Design for Display Interface Program. The interface design of this system is divided into three parts according to the software design requirements: master computer interface design, server interface design, and client interface design. The master computer interface design includes digital tube display and LCD display; the server interface design includes environmental data display interface and server interface; the client part includes one data read interface.

The design for display interface program is in Appendix B, the server interface design is in Appendix C, and the client interface design is in Appendix D.

5.6. Data Processing Algorithm. According to the software design requirements, this system will adopt a user-defined data transmission format and implement the alarm function through data analysis.

The implementation of the alarm device is in Appendix E, and the data transmission protocol is in Appendix F.

6. System Function Debugging

This system mainly judges and debugs from the data display of each sensor module. Start the system and obtain multiple data in real time on the mobile APP, and judge whether the collected data meet the corresponding conditions according to different environmental conditions, such as the light intensity and the temperature.

6.1. Temperature-Humidity Module. The slave computer is equipped with a temperature-humidity sensor module to collect temperature and humidity information in the environment. DHT11 has obvious advantages, such as strong anti-interference ability. It is designed with a single bus data format, which can complete data communication every four milliseconds. This series of data is divided into decimal and integer parts. The specific data format is as follows: the first eight digits represent the integer part of the humidity, the next eight digits represent the decimal part of the humidity, the further next eight digits represent the integer part of the temperature, the following next eight digits represent the decimal part of the temperature, and the last eight digits represent the checksum data. This system is only accurate to the integer part of the temperature and humidity. Start the system, then open the serial port, and debug the mobile APP to receive the temperature and humidity data collected by the temperature-humidity sensor.

6.2. Flame Sensor Module. The slave computer is equipped with a flame sensor module to collect the flame status information in the environment. It can monitor the infrared light with the wavelength in the range of 700 nm–1000 nm, with the detection angle of 60 degrees, and the value is inversely proportional to the intensity of external infrared light. Start the system, then open the serial port, debug the mobile APP to receive the flame status data information collected by the flame sensor, and lean the lighter to the flame sensor module. The collected data is displayed in real time through the serial port.

6.3. Smoke Sensor Module. The slave computer is equipped with a smoke sensor module, which can collect the smoke concentration information in the environment. The MQ-2 gas sensor can detect flammable gases in the environment and judge the environmental safety status by combining with the flame sensor module. Start the system, then open the serial port, and debug the mobile APP to receive the smoke concentration data information collected by the smoke sensor. During the debugging process, it is required to use the unburned gas released by the lighter as the detected gas, and let the gas slowly approach the sensor.

6.4. Photosensitive Sensor Module. The slave computer is equipped with a photosensitive sensor to collect light intensity information in the environment. This system reads the voltage value of the photosensitive sensor through channel 6 of ADC3 and converts it into the light intensity value of 0~100. The brighter the light on the module is, the larger the value could be; the darker the light is, the smaller the value would be. Start the system, then open the serial port, and debug the mobile APP to receive the smoke concentration data information collected by the smoke sensor. It is required to check the light intensity change by adjusting the light source of mobile phone.

7. Conclusion

With the science advancement and technology popularization, intelligent devices enjoy more and more widespread application, and problems such as the accuracy of environmental issues have been aroused therefrom. The previous environmental monitoring was carried out by means of manual monitoring, which has many shortcomings from the present point of view, including low efficiency and unscientific management system. It was not conducive to the full utilization of resources and scientific management. Moreover, it was not difficult to find from the major accidents in these years that many were caused by human factors. The development of an intelligent monitoring system can solve the related problems. The system can implement the following functions:

(1) Real-time collection of environmental information is implemented, and data can be processed and analyzed through processors.

(2) The wireless transmission of data is realized so that the communication between the master computer and the slave computer can be implemented.

(3) The serial communication of data is realized so that the communication between the master computer and the server can be implemented.

(4) The network communication of data is implemented, and the environmental status can be monitored through a mobile APP from a long distance.

(5) The intelligent alarm function can be implemented.

Appendix

A. The Configuration Procedure for Establishing a Connection by Using Socket

```
// Create a socket class for declaring the required functions
class TcpClientSocket: public QTcpSocket
{
 Q_OBJECT
public:
   TcpClientSocket(QObject *parent = 0);
signals:
   void updateClients (QString, int); //
Update the client
   void disconnected (int); // Unlink
 protected slots:
   void dataReceived();
   void slotDisconnected();
 };
TcpClientSocket::TcpClientSocket(QObject *parent)
{
connect (this, SIGNAL (readyRead ()), this, SLOT (dataReceived ())); // This signal is emitted when there is incoming data that can be read. This signal is transmitted once every time there is new incoming data. It is worth noting that this data can only be read once, so this data must be read once; otherwise, this data will be discarded.
connect(this, SIGNAL(disconnected()), this, SLOT(slotDisconnected()));
}
```

B. Design for Master Computer Interface

The master computer interface of this system is used to display on the large screen around the monitored environment. The data in the environment is obtained through the wireless module and displayed on the interface, following the sequence of light intensity, temperature, humidity, smoke concentration, and flame status.

This system uses the battleship board with a built-in 2.8-inch LCD screen. The display of strings and numbers is, respectively, implemented here mainly by calling two functions, that is, LCD_ShowString () and LCD_ShowxNum (). The following is the call to the LCD display function.

```
LCD_Init (); // Initialize LCD
POINT_COLOR = RED; // Set the font as red color
LCD_ShowString(30,50,200,16,16, “WarShip STM32”);
LCD_ShowString(30,70,200,16,16, “Wireless Environment”);
LCD_ShowString(30,90,200,16,16, “ATOM@ ALIENTEK”);
LCD_ShowString(30,90,200,16,16, “ATOM@ ALIENTEK”);
LCD_ShowString(30,110,200,16,16, “2018/6/6”);
POINT_COLOR = BLUE; // Set the font as blue color
LCD_ShowString(30,130,200,16,16, “Light:”); // Display the light intensity
LCD_ShowxNum(130,130,adcx,3,16,0);
LCD_ShowString(30,150,200,16,16, “Temperature:”); //Display the temperature
LCD_ShowxNum(130,150,tmp_value [2],3,16,0);
LCD_ShowString(30,170,200,16,16, “Humidity:”); // Display the humidity
LCD_ShowxNum(130,170,tmp_value [3],3,16,0);
LCD_ShowString(30,190,200,16,16, “Smoke:”); // Display the smoke density
LCD_ShowxNum(130,190,tmp_value[4],3,16,0);
LCD_ShowString (30,210,200,16,16, “Fire:”); // Display the flame status
LCD_ShowxNum(130,210,tmp_value[5],3,16,0);
```

This system uses an external four-bit common anode digital tube to display temperature and humidity, respectively. The main configuration of the digital tube is realized by two functions of singleDisplay () and smDisplay (). They are used for cycle digital display of single-digit digital tube and four-digit digital tube. Here, position selection signals D1, D2, D3, and D4 of the digital tube are sequentially connected to PD0, PD1, PD2, and PD3 of the microcontroller, while the segment selection signals A-H are sequentially connected to PA0-PA7 of the microcontroller.

```
void singleDisplay (char number) // Single digital tube display
{
switch(number)
  {
  case 0: Number_0(); break;
  case 1: Number_1(); break;
  case 2: Number_2(); break;
  case 3: Number_3(); break;
  case 4: Number_4(); break;
  case 5: Number_5(); break;
  case 6: Number_6(); break;
  case 7: Number_7(); break;
  case 8: Number_8(); break;
  case 9: Number_9(); break;
  }
}
void smDisplay (char * number) // cycle display of four-digit digital tube
{
CH_1 = 0;
singleDisplay(number[0]);
CH_2 = 1; CH_3 = 1; CH_4 = 1;
delay_us (2000);
CH_2 = 0;
singleDisplay(number [1]);
CH_1 = 1; CH_3 = 1; CH_4 = 1;
delay_us (2000);
CH_3 = 0;
singleDisplay (number [2]);
CH_2 = 1; CH_1 = 1; CH_4 = 1;
delay_us(2000); CH_4 = 0;
singleDisplay(number [3]);
CH_2 = 1; CH_3 = 1; CH_1 = 1;
delay_us(20);
}
```

C. Server Interface Design

The server interface design of this system is generally divided into two parts, one for displaying data and the other for creating server. The specific codes are as follows:

```
int main(int argc, char *argv[])
{ QApplication a(argc, argv);
MainWindow w1; // This window is used to display data
w1.setWindowTitle("Wireless Environment Monitoring System"); // Setting the interface title
w1.show();
TcpServer w2;
w2.show (); // This window is used to create server
return a.exec(); }
```

The main function defines two interfaces and sets the interface title. The following codes serve as the configuration for the data display interface.

```
MainWindow::MainWindow(QWidget *parent):
QMainWindow(parent), ui(new Ui::MainWindow)
{
ui ⟶ setupUi(this);
this ⟶ setWindowIcon (QIcon (" / back/image/xiaohui.jpg")); // Generate a window icon
QTextCodec*BianMa = QTextCode-c::codecForName ("UTF-8"); // This format must be used here, otherwise the problem of garbled characters will appear
myCom = NULL;
ui ⟶ portNameComboBox ⟶ addItem ("COM0"); // Add serial port
ui ⟶ statusBar ⟶ showMessage (BianMa ⟶ toUnicode ("Welcome")); }
Void MainWindow: on_actionExit_triggered () // Close the software
{
this ⟶ close();
}
Void MainWindow::on_actionAbout_triggered () // Software information
{
QTextCodec*BianMa = QTextCodec:codecForName ("UTF-8");
ui ⟶ statusBar ⟶ showMessage (BianMa ⟶ toUnicode ("This software is used to analyze the serial transmission of eight-bit data at 115200 bit rate");
}
```

The following codes serve as the configuration and UI layout of the server interface:

```
TcpServer::TcpServer(QWidget∗parent,Qt::WindowFlags f): QDialog(parent,f)
{
setWindowTitle (tr ("TCP Server")); // Set the title
ContentListWidget = new QListWidget;
PortLabel = new QLabel (tr ("Port:"));
PortLineEdit = new QLineEdit;
CreateBtn = new QPushButton (tr ("Create Server"))
mainLayout = new QGridLayout(this); mainLayout ⟶ addWidget(ContentListWidget,0,0,1,2);
mainLayout ⟶ addWidget(PortLabel,1,0);
mainLayout ⟶ addWidget(PortLineEdit,1,1);
mainLayout ⟶ addWidget(CreateBtn,2,0,1,2);
port = 8010;
PortLineEdit ⟶ setText(QString:number(port));
connect(CreateBtn, SIGNAL(clicked()), this, SLOT(slotCreateServer())); }
```

D. Client Interface Design

The client interface design of this system is based on an Android phone. In the local area network, a socket is established on the basis of TCP protocol to achieve interconnection with the server. Its interface layout and related functions are shown as follows:

```
int main (int argc, char ∗ argv []) // Create an interface and display
{
QApplication a(argc, argv);
QSize size = a.desktop () ⟶ screenGeometry () .size (); // For full screen display on mobile phone
TcpClient w;
w.resize(size);
w.show();
return a.exec();
}
```

The following function is used to layout the client interface. It is implemented by calling QT buttons, labels, and text boxes.

```
TcpClient::TcpClient(QWidget ∗parent,Qt::WindowFlags f): QDialog(parent,f)
{
setWindowTitle(tr("TCP Client"));
Light_1 = new QLabel (tr ("Lighting:"));
Light_2 = new QListWidget;
Temp_1 = new QLabel (tr ("Temperature:"));
Temp_2 = new QListWidget;
Humi_1 = new QLabel (tr ("Humidity:"));
Humi_2 = new QListWidget;
Smoke_1 = new QLabel (tr ("Flame:"));
Smoke_2 = new QListWidget;
Fire_1 = new QLabel (tr ("Smoke:"));
Fire_2 = new QListWidget;
sendLineEdit = new QLineEdit;
sendBtn = new QPushButton (tr ("Send"));
userNameLabel = new QLabel (tr ("user name:"));
userNameLineEdit = new QLineEdit;
serverIPLabel = new QLabel (tr ("Server address:"));
serverIPLineEdit = new QLineEdit;
portLabel = new QLabel (tr ("port:"));
portLineEdit = new QLineEdit;
enterBtn = new QPushButton (tr ("Enter the monitoring room"));
mainLayout = new QGridLayout(this);
mainLayout ⟶ addWidget(Light_1,0,0);
mainLayout ⟶ addWidget(Light_2,0,1);
mainLayout ⟶ addWidget(Temp_1,1,0);
mainLayout ⟶ addWidget(Temp_2,1,1);
mainLayout ⟶ addWidget(Humi_1,2,0);
mainLayout ⟶ addWidget(Humi_2,2,1);
mainLayout ⟶ addWidget(Smoke_1,3,0);
mainLayout ⟶ addWidget(Smoke_2,3,1);
mainLayout ⟶ addWidget(Fire_1,4,0);
mainLayout ⟶ addWidget(Fire_2,4,1);
mainLayout ⟶ addWidget(sendLineEdit,5,0);
mainLayout ⟶ addWidget(sendBtn,5,1);
mainLayout ⟶ addWidget(userNameLabel,6,0);
mainLayout ⟶ addWidget(userNameLineEdit,6,1);
mainLayout ⟶ addWidget(serverIPLabel,7,0);
mainLayout ⟶ addWidget(serverIPLineEdit,7,1);
mainLayout ⟶ addWidget(portLabel,8,0);
mainLayout ⟶ addWidget(portLineEdit,8,1);
mainLayout ⟶ addWidget (enterBtn, 9,0,1,2); // For obtaining data
status = false;
port = 8010;
portLineEdit ⟶ setText(QString:number(port));
serverIP = new QHostAddress();
connect(enterBtn, SIGNAL(clicked()),this, SLOT(slotEnter()));
connect(sendBtn, SIGNAL(clicked()),this, SLOT(slotSend()));
sendBtn ⟶ setEnabled(false);
}
```

E. Alarm Device

This system adopts an acoustooptic alarm device, with an alarm threshold set on the master computer. The automatic alarm function will be realized when the relevant value reaches the threshold. This system uses the built-in buzzer of the battleship board to implement the alarm function. The macrodefinition and initialization function in its header file are shown as follows:

```
#define ALARM PBout (8) // Buzzer interface
void ALARM_Init (void); // Initialize the buzzer
void ALARM_Init(void)
{
GPIO_InitTypeDef GPIO_InitStructure;
RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOB, ENABLE); // Enable GPIOB port clock
GPIO_InitStructure.GPIO_Pin = GPIO_Pin_8; // Port configuration
GPIO_InitStructure.GPIO_Mode = GPIO_Mode_Out_PP; // Push-pull output
GPIO_InitStructure.GPIO_Speed = GPIO_Speed_50 MHz;
GPIO_Init(GPIOB,&GPIO_InitStructure); // Initiate GPIOB according to parameters
GPIO_ResetBits(GPIOB, GPIO_Pin_8); // Output 0, the buzzer is off
}
```

For the realization of the automatic alarm function, the codes are as follows:

```
if (adcx> = 50 || tmp_value [2]> = 30 || tmp_value [3]> = 30 || tmp_value [4]> 200 || tmp_value [5] == 1) // Set the threshold
{
ALARM = 1; // Set the buzzer status
}else
{
ALARM = 0;
}
```

F. Data Transmission Protocol

This system collects environmental data in real time, so the values of these data are uncontrollable. In order to ensure the stable and secure transmission of data, self-defined data format will be adopted here. The data format is specified in the program, in which temperature, humidity, and light intensity each occupy two digits, smoke concentration occupies three digits, and flame status occupies one digit. The master computer part implements data conversion by calling the Protocol () function. The codes are as follows:

```
void Protocol(unsigned char L, unsigned char T, unsigned char H, unsigned char S, unsigned char F)
{
unsigned char pro [11]; int i; pro [0] = L/10; // Take the value of each amount
pro [1] = L % 10;
pro [2] = T/10;
pro [3] = T % 10;
pro [4] = H/10;
pro [5] = H % 10;
pro [6] = S/100;
pro [7] = (S-pro [6] * 100)/10;
pro [8] = (S-pro [6] * 100) % 10;
pro [9] = F % 10;
for(i = 0; i < 10; i++)
printf ("%d", pro[i]);
}
```

The server will separate and display the data read through the serial port according to the data format. The codes are as follows:

```
QByteArray temp = myCom ⟶ readAll (); // Read the data transmitted from the serial port
QTextCodec * BianMa = QTextCodec: codecForName ("UTF-8"); // This is used to ensure accurate display of Chinese characters and prevent garbled codes
QString datas = (QString) temp; // Forced data
QString Light = datas.mid (0,2); // Extract the light intensity value
QString Temp = datas.mid (2,2); // Extract temperature value
QString Humi = datas.mid (4,2); // Extract humidity value
QString Smoke = datas.mid (6,3); // Extract the smoke concentration value
QString Fire = datas.mid (9,1); // Extract the flame status value
DATA = Light + Temp + Humi + Smoke + Fire;
ui ⟶ lcd_Light ⟶ display(Light.toFloat(&ok));
ui ⟶ lcd_Light ⟶ display (Light.toFloat (&ok));
ui ⟶ lcd_Temp ⟶ display (Temp.toFloat (&ok));
ui ⟶ lcd_Humi ⟶ display(Humi.toFloat(&ok));
ui ⟶ lcd_Smoke ⟶ display(Smoke.toFloat(&ok));
ui ⟶ lcd_Fire ⟶ display(Fire.toFloat(&ok);
ui ⟶ statusBar ⟶ showMessage (BianMa ⟶ toUnicode ("Successfully read% 1 byte data"). arg (temp.size ()));
```

Data Availability

The data, models, and code used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Research Project of the Education Department of Hunan Province (18A441 and 18C0958), Educational Planning Key Project of Hunan Province (XJK18DJA1), Provincial Natural Science Foundation of Hunan (2019JJ60041 and 2020JJ4587), Social Science Achievement Evaluation Committee Project of Hunan Provincial under Grant XSP20YBZ031, and Provincial Natural Science Foundation of Hunan ("Research on Mobile Robot Tracking Based on Parallel Nonlinear Filtering").

References

[1] C. Canali, N. Rahman, F. Chen, M. D'imperio, D. Caldwell, and F. Cannella, "Theoretical and kinematic solution of high reconfigurable grasping for industrial manufacturing," *Procedia Manufacturing*, vol. 11, pp. 265–274, 2017.

[2] K. Lee, H. Kim, S. Lee, S.-W. Choo, S.-M. Lee, and K.-T. Nam, "High precision hand-eye self-calibration for industrial robots," in *Proceedings of the 2018 International Conference on Electronics, Information, and Communication (ICEIC)*, Honolulu, HI, USA, April 2018.

[3] W. Mustafa, N. Pugeault, A. G. Buch, and N. Krüger, "Multi-view object instance recognition in an industrial context," *Robotica*, vol. 35, no. 2, pp. 271–292, 2017.

[4] N. Kim, S. T. Hong, K. Sung, G. E. Yu, and J. H. Seo, "A case study on risk assessment for personal care robot (mobile servant robot)," in *Proceedings of the 2018 18th International Conference on Control*, Automation and Systems (ICCAS), PyeongChang, South Korea, October 2018.

[5] J. Wang, Y. Yao, and X. Kong, "A reconfigurable tri-prism mobile robot with eight modes," *Robotica*, vol. 36, no. 10, 2018.

[6] Y. Lin, H. Min, H. Zhou, and F. Pei, "A human-robot-environment interactive reasoning mechanism for object sorting robot," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 10, no. 3, pp. 611–623, 2018.

[7] Y. J. Heo, D. Kim, W. Lee, H. Kim, J. Park, and W. K. Chung, "Collision detection for industrial collaborative robots: a deep learning approach," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 740–746, 2019.

[8] H. M. La, T. H. Dihn, N. H. Pham, Q. P. Ha, and A. Q. Pham, "Automated robotic monitoring and inspection of steel structures and bridges," *Robotica*, vol. 37, 2018.

[9] O. SeungSub, H. Jehun, J. Hyunjung, L. Soyeon, and S. Jinho, "A study on the disaster response scenarios using robot technology," in *Proceedings of the 2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, Jeju, Korea, July 2017.

[10] H. Su, C. Yang, and G. Ferrigno, "Improved human-robot collaborative control of redundant robot for tmis," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1447–1453, 2019.

[11] G. Yang, S. Wang, and J. Yang, "Desire-driven reasoning for personal care robots," *IEEE Access*, vol. 7, pp. 75203–75212, 2019.

[12] G. Dwyer, F. Chadebecq, M. T. Amo et al., "A continuum robot and control interface for surgical assist in fetoscopic interventions," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1656–1663, 2017.

[13] S. Gao, R. Song, and Y. Li, "Cooperative control of multiple nonholonomic robots for escorting and patrolling mission based on vector field," *IEEE Access*, vol. 6, pp. 41883–41891, 2018.

[14] T. Shimizu, R. Saegusa, S. Ikemoto, H. Ishiguro, and G. Metta, "Robust sensorimotor representation to physical interaction changes in humanoid motion learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 5, pp. 1035–1047, 2015.

[15] J. Nassour, V. Hugel, F. B. Ouezdou, and G. Cheng, "Qualitative adaptive reward learning with success failure maps: applied to humanoid robot walking," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 1, pp. 81–93, 2013.

[16] H. Soh and Y. Demiris, "Spatio-temporal learning with the online finite and infinite echo-state Gaussian processes," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 3, pp. 522–536, 2015.

[17] Z. Chun-Yi Su and C.-Y. Su, "Neural-Adaptive control of single-master-multiple-slaves t for coordinated multiple mobile manipulators with time-varying communication delays and input uncertainties," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 9, pp. 1400–1413, 2013.

[18] Z. Zhijun Zhang, A. Beck, and N. Magnenat-Thalmann, "Human-like behavior generation based on head-arms model for robot tracking external targets and body parts," *IEEE Transactions on Cybernetics*, vol. 45, no. 8, pp. 1390–1400, 2015.

[19] I. Gravagne and I. D. Walker, "Properties of minimum infinity-norm optimization applied to kinematically redundant robots," in *Proceedings of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems. Innovations in Theory, Practice and Applications (Cat. No.98CH36190)*, Victria, Canada, October 1998.

[20] Y. Zhang, W. Li, and Z. Zhang, "Physical-limits-constrained minimum velocity norm coordinating scheme for wheeled mobile redundant manipulators," *Robotica*, vol. 33, no. 6, pp. 1325–1350, 2015.

[21] Y. Zhang, J. Yin, and L. Tian, "Bi-criteria acceleration minimization of redundant robot manipulator using LVI-based primal-dual neural network," in *Proceedings of the 2007 Chinese Control Conference*, Zhangjiajie, China, July 2007.

[22] H.-B. Choi, S. Lee, and J. Lee, "Minimum infinity-norm joint velocity solutions for singularity-robust inverse kinematics," *International Journal of Precision Engineering and Manufacturing*, vol. 12, no. 3, pp. 469–474, 2011.

[23] Y. Yunong Zhang, J. Jun Wang, and Y. Youshen Xia, "A dual neural network for redundancy resolution of kinematically redundant manipulators subject to joint limits and joint velocity limits," *IEEE Transactions on Neural Networks*, vol. 14, no. 3, pp. 658–667, 2003.

[24] S. W. S. Tang and J. Wang, "A primal-dual neural network for kinematic control of redundant manipulators subject to joint velocity constraints," in *Proceedings of the 6th International Conference Neural Information Process. (ICONIP)*, vol. 2, pp. 801–806, Perth, Australia, November 1999.

[25] Z. G. Zeng-Guang Hou, L. Long Cheng, and M. Min Tan, "Multicriteria optimization for coordination of redundant robots using a dual neural network," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 4, pp. 1075–1087, 2010.

[26] Y. Zhang, X. Lv, Z. Li, Z. Yang, and K. Chen, "Repetitive motion planning of PA10 robot arm subject to joint physical limits and using LVI-based primal-dual neural network," *Mechatronics*, vol. 18, no. 9, pp. 475–485, 2008.

[27] K. Madhava Krishna, R. Alami, and T. Simeon, "Safe proactive plans and their execution," *Robotics and Autonomous Systems*, vol. 54, no. 3, pp. 244–255, 2006.

[28] M. H. Farzanehkaloorazi, M. T. Masouleh, and S. Caro, "Collision-free workspace of parallel mechanisms 478 based on an interval analysis approach," *Robotica*, vol. 35, no. 8, pp. 1747–1760, 2018.

[29] R. Domínguez and M. D. Somonte, "Design in robotics based in the voice of the customer of household robots," *Robotics and Autonomous Systems*, vol. 79, pp. 99–107, 2016.

[30] A. K. Singh and K. M. Krishna, "Feasible acceleration count: a novel dynamic stability metric and its use in incremental motion planning on uneven terrain," *Robotics and Autonomous Systems*, vol. 79, pp. 156–171, 2016.

[31] K. Curran and S. Norrby, "RFID-enabled location determination within indoor environments," *Ubiquitous Developments in Ambient Computing and Intelligence: Human-Centered Applications*, vol. 1, pp. 301–304, 2011.

[32] W. Yang, X. Wang, X. Song, Y. Yang, and S. Patnaik, "Design of intelligent transportation system supported by new generation wireless communication technology," *International Journal of Ambient Computing and Intelligence*, vol. 9, no. 1, pp. 78–94, 2018.

[33] N. Dey, A. S. Ashour, F. Shi, S. J. Fong, and R. S. Sherratt, "Developing residential wireless sensor networks for ECG healthcare monitoring," *IEEE Transactions on Consumer Electronics*, vol. 63, no. 4, pp. 442–449, 2017.

[34] Q. Liu, "Intelligent environmental monitoring system based on multi-sensor data technology," *International Journal of Ambient Computing and Intelligence*, vol. 11, no. 4, pp. 57–71, 2020.

[35] S. Yahya, M. Moghavvemi, and H. A. F. Mohamed, "Manipulability constraint locus for a six degrees of freedom redundant planar manipulator," in *Proceedings of the International Symposium on Computer, Consumer and Control*, pp. 290–293, Taichung, Taiwan, June 2012.

[36] X. Ge and J. Jin, "Dynamics analyze of a dual-arm space robot system based on kane's method," in *Proceedings of the International Conference on Industrial Mechatronics and Automation*, pp. 646–649, Wuhan, China, May 2010.

[37] M. Chen and A. M. S. Zalzala, "A genetic approach to motion planning of redundant mobile manipulator systems considering safety and configuration," *Journal of Robotic Systems*, vol. 14, no. 7, pp. 529–544, 1997.

[38] Y. Wang, "Dual 7-degree-Of -freedom robotic arm remote teleoperation using haptic devices," *University of South Florida*, Dissertations & Theses-Gradworks, Tampa, FL, USA, 2015.

[39] C. Fan, C. Ding, J. Zheng, L. Xiao, and Z. Ai, "Empirical mode decomposition based multi-objective deep belief network for short-term power load forecasting," *Neurocomputing*, vol. 388, pp. 110–123, 2020.

[40] A. M. Zanchettin, L. Bascetta, and P. Rocco, "Achieving humanlike motion: resolving redundancy for anthropomorphic industrial manipulators," *IEEE Robotics and Automation Magazine*, vol. 20, no. 4, pp. 131–138, 2013.

[41] H. Yuan, P. W. Y. Chiu, and Z. Li, "Shape-reconstruction-based force sensing method for continuum surgical robots with large deformation," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1972–1979, 2017.