https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

Research on On-line Data Transmission Technology in Marine Controlled Source Electromagnetic Transmitter

Chentao Wang[1], Ming Deng[1], Nini Duan[1], Xiaoxi Ma[1], Meng Wang[1]

[1] *China University of Geosciences (Beijing), School of Geophysics and Information Technology, Beijing CO 100083, China*

Corresponding author: Meng Wang (e-mail: wangmeng@cugb.edu.cn).

This work was supported by National Natural Science Foundation of China (41874142) ,the MOST Special Fund from the State Key Laboratory of Geological Processes and Mineral Resources, China University of Geosciences and the National Key R&D Program of China (2016YFC0303100).

Abstract This paper proposes a method for acquiring complete information and data from the Marine Controlled Source Electromagnetic (MCSEM) transmitter during offshore experiments. The lower position machine system is based on the STM32 platform and embedded with a real-time operating system. It utilizes the Internet of Things concept to interconnect various modules within the transmitter, enabling intelligent control and management. At the same time, data is uploaded to the control room on the deck through photoelectric composite cables and the upper computer's software designed by Python language will process and store all the data. This allows workers on the deck to control the lower computer and get high-precision complete data in real-time. The joint tests between the lower and upper computers have demonstrated the stability and reliability of the online transmitter system, which provides significant convenience for offshore exploration.

Keywords : MCSEM; Embedded system; Internet of Things; Online data transmission and processing

1 Introduction

The Marine Controlled Source Electromagnetic method plays a crucial role in exploring deep-sea oil and gas resources (Li et al., 2022; Constable, 2006, 2010). During marine exploration, the research vessel tows the electromagnetic transmitter close to the seafloor using the photoelectric composite deep-tow cable. The electromagnetic receiver then acquires the induction field source signal generated by the underground geological anomaly (Wang et al., 2013). After the instruments are recovered, the data collected from the receivers and transmitters are integrated, processed, and inverted to obtain the apparent resistivity of the seabed in the detected sea area (Connell et al., 2013). This effectively identifies hydrocarbons and provides the basis for the exploitation of oil and gas resource (Li et al., 2010; Wang et al., 2019; Constable et al., 2016).

The development of MCSEM research in China began in 2006, supported by the National High-tech Research and Development Program (863 program) and other national funds. A series of MCSEM instruments have been developed, including mixed field source electromagnetic receiver, towed receiver (Chen et al., 2013), towed transmitter system (Wang et al., 2017), and deployed transmitter (Wang et al., 2017).

Nonetheless, these methods depend on locally deployed data, with merely a small fraction being uploaded to the deck for monitoring purposes. The comprehensive dataset for analysis

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

typically becomes accessible only once the instrument is retrieved (Duan et al., 2018; Wang et al., 2022; Chen et al., 2020). This data acquisition method does not ensure the safety and promptness of the data. To minimize the risk of data loss and achieve real-time data transmission from the transmitter to the deck side, an online current data transmission system has been developed. Figure 1 depicts the relationship between the online transmission system and the entire system.

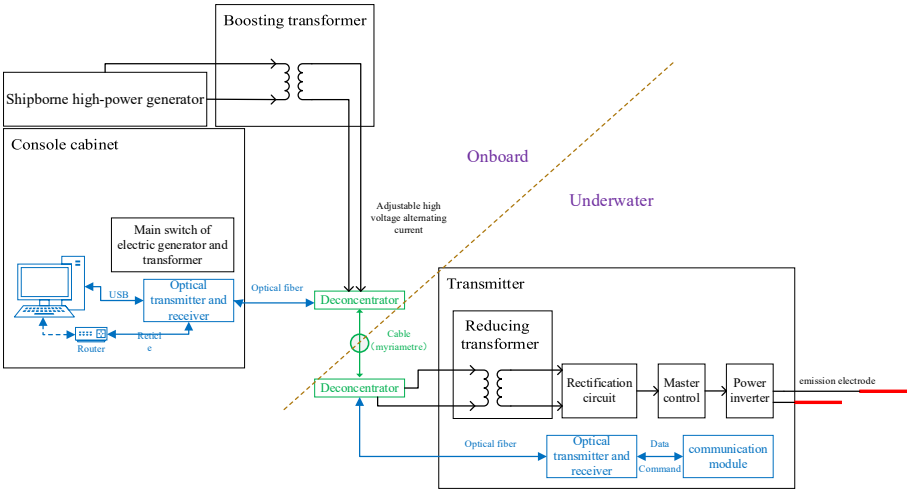

Fig. 1 Working mode of the on-line transmission system in the whole system

The main task of the lower computer part in the system is to store the data into the storage media as a file according to the certain format, and provide inter-module communication channel, once the data from different modules is consolidated, it is transmitted via Ethernet for uploading.

The upper computer is responsible for receiving data from the lower computer, storing it on the upper computer, and providing users with a user-friendly interface to control the lower computer.

2 The technical scheme design of lower computer

2.1 Overall framework

The main control unit in the lower computer is the STM32F407ZGT6 produced by STMicroelectronics. We deploy data acquisition module, data storage module, serial communication module, Ethernet communication module, and real-time operating system. In the system, data management is facilitated by the FATFS system, and we have reserved “USMART” serial port function identification, “NETCONN” Ethernet interface. The coordination of each module is illustrated in Figure 2.

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

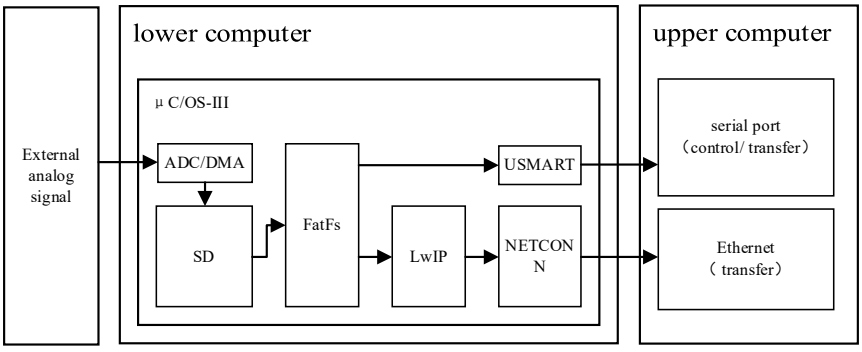

Fig. 2 The overall framework of the lower computer program

2.1.1 Task management based on μC/OS-III

We utilize μC/OS -III as the system to manage complex tasks the lower computer. It is a preemptive multi-task real-time operating system based on priority which can significantly optimizes the program structure, increases readability, and improves portability. Each functional module is regard as a subtask for the operating system to schedule. These modules are independent of each other and provide corresponding APIs, making it convenient to transplant and greatly optimizes the Ethernet data transmission and command interaction.

2.1.2 File management based on FatFs

FatFs is a general file management module that is independent of the I/O layer of the disk, making it hardware architecture agnostic. This allows it to implement the related functions of a Fat file system in small embedded systems. In the system, file management of the lower computer is based on FatFs, which stores collected data to the SD card or other media in the form of a file and provides corresponding read-write functions for file transmission, this allows the main control chip to focus more on data transmission once the data is integrated into a file.

2.1.3 Internet of Things based on LwIP

There are a large number of data exchange behaviors within the lower computer, and how they work together depends on smooth data transmission channels. Leveraging the concept of the Internet of Things (IoT), we connect various modules through WiFi and Ethernet channels to ensure seamless communication, as the Figure 3. Meanwhile, using the serial port for file transmission is often limited due to the baud rate. To improve the speed of transmission and enhance the user experience of the software, Ethernet file transmission is used as the primary transmission mode in this design. Reliable Ethernet data transmission cannot be achieved without the TCP/IP protocol. However, traditional TCP/IP protocols are rarely used in small embedded systems. LwIP has the basic functions of TCP/IP and takes up less RAM, making it suitable for embedded systems with limited program space. LwIP also supports the DHCP protocol, can dynamically assign IP addresses, and provides a special internal callback interface that effectively improves program performance.

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

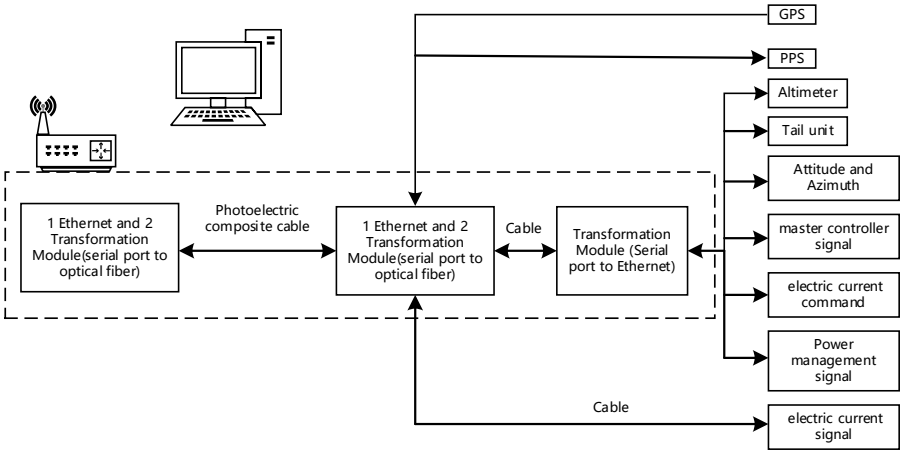

Fig. 3 Leveraging the concept of the Internet of Things (IoT)

2.1.4 User operation based on USMART

The USMART module is used as a bridge between the user and the program in this design. Users can use each function of the program through serial port commands, achieving the effect of an external interrupt, which is of great significance for systems requiring remote real-time control. Therefore, the USMART module is transplanted into the lower computer module to connect with the upper computer software and provide users with various control commands and parameters. This allows users to control the lower computer through the upper computer.

2.2 Design of data acquisition module

2.2.1 Configuration of ADC and DMA

The main control unit is equipped with a 12-bit successive approach analog-to-digital converter (ADC) capable of measuring the signal of 16 external sources, 2 internal sources, and Direct Memory Access (DMA) capabilities. DMA allows for quick transfer of data between peripheral devices and memory, saving MCU resources for other operations. In this design, file storage and data acquisition are carried out simultaneously, and the double buffer mode of DMA is used to solve this problem. This mode significantly reduces the MCU load, as shown in Figure 4, which lists the workflow of data acquisition and storage in three cases: without DMA, using DMA, and using DMA double buffer mode. The execution efficiency of DMA double buffer mode is higher than the other two modes.

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

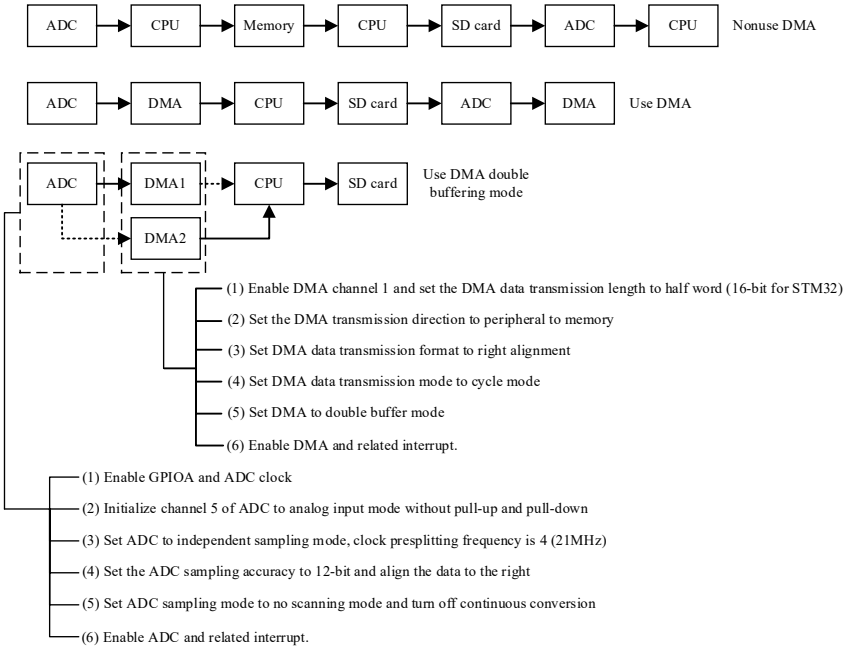

Fig. 4 Workflow of ADC / DMA in different modes and the configuration of DMA and ADC

2.2.2 Sampling rate configuration and data acquisition process monitoring

To ensure suitable sampling, the sampling rate should be adjusted according to the actual situation. In this design, the sampling rate of the ADC is controlled by the TIM3 timer. The trigger mode of the ADC is set to the TIM3_TRGO event, and the working state of the ADC is controlled by the overflow interrupt of the TIM3 timer. The overflow time of TIM3 can be changed by external input parameters, such as the frequency division number and reset value. The ADC sampling rate calculation formula is as follows, where ARR represents the reset value and PSC represents the frequency division number:

$$ADCCLK=84000000/(ARR*PSC)$$

Additionally, an additional timer (TIM4) is used to capture the number of times that the ADC conversion is completed in a certain period. This allows for real-time estimation of the ADC sampling rate and monitoring of the ADC's running state.

2.3 Design of file storage module

In the file storage module, the data files are stored on an SD card, and the management of files relies on the FatFs file system. The porting process of the file system is not important, and the focus is on the method of realizing file storage. The storage process of the data file is illustrated in Figure 5. The write-flag and the stop-write flag are given by the serial port command, allowing users to start and terminate storage at any time. At the same time, the write-status flag is used to mark the current state of the file storage module. Setting the maximum number of writes can control the size of each data file to about 10MB. When the maximum number of writes is reached, the module will automatically close the current file and create a new file to continue writing. This ensures that when the system works for a long time, there are no problems caused by the size of a single data file.

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

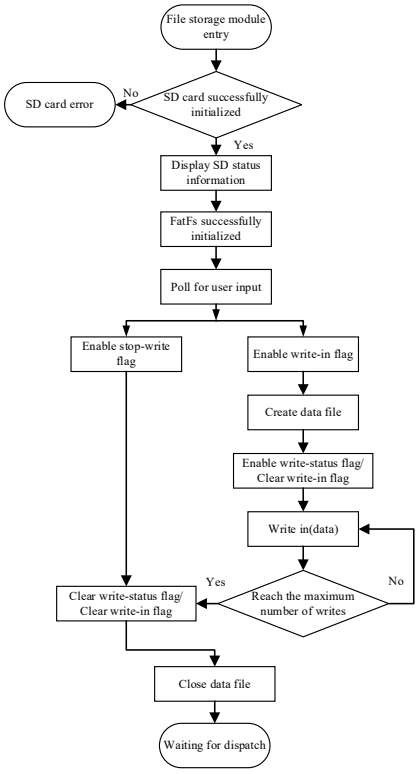

Fig. 5 Data file storage process

2.4 Design of file transmission module

The file transfer module consists of Ethernet data transmission and serial data transmission functions. Ethernet data transmission is based on LwIP, and serial transmission is used as a backup channel for Ethernet transmission. The default baud rate for serial port transmission is 115,200 bps, and the default port number for Ethernet transmission is 8,087. The lower computer acts as the client, while the upper computer is the server.

2.4.1 Data file transmission of Ethernet

The LwIP protocol stack cannot be directly transplanted to the μC/OS-III system and requires encapsulation. LwIP has three commonly used programming interfaces: RAW, NETCONN, and socket interfaces. The RAW programming interface is often used when LwIP is used alone, while the NETCONN and socket interfaces are suitable to use with an operating system. The NETCONN interface is a structure abstracted from LwIP and is more concise than the socket interface. It does not waste memory when copying files. Therefore, the NETCONN programming interface is used in this design.

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

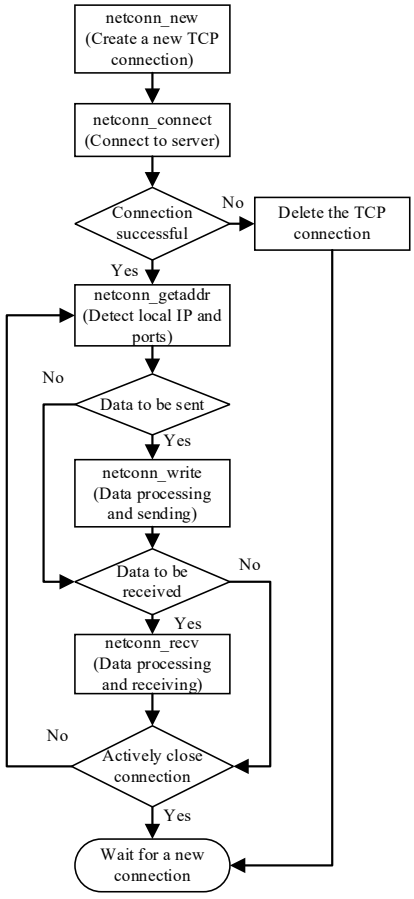

Fig. 6 Ethernet data file transmission process

The flow chart of Ethernet data file transmission with the NETCONN programming interface is shown in Figure 6. In the process of data file reading, the length of single data should not be too small to reduce the impact of instruction judgment on transmission speed. In limited program space, the dynamic memory allocation method is necessary. The program uses the malloc function to dynamically allocate a data buffer of 512*32 bytes in SRAMIN (extended internal SRAM), which is automatically released after use. The size of the cache is a multiple of 512 bytes because the read and write object of the FatFs file system is a sector (512 bytes). After setting the length of a single data, it is necessary to increase the Ethernet data sending window. If the default sending window is used, packet loss will occur when the window length is less than the data length, which will affect the accuracy of the data.

2.4.2 Data file transmission and status control of serial port

As a key debugging tool and data communication interface, the importance of serial port is self-evident. In this design, CH340G is used as the serial port to USB chip because the MCU needs to communicate with the host computer through the USB interface. The CH430G chip can be compatible with RS485, RS232, and other interfaces through the external level converter. As a full-

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

speed USB device interface, it is compatible with USB2.0 and serial port applications under the Windows operating system.

With the hardware environment in place, the configuration of the serial port in the program follows these steps: (1) Enable GPIOA clock and enable USART1 clock. (2) Reuse GPIOA9 and GPIOA10 for USART1. (3) Enable GPIOA9 and GPIOA10. (4) Set the baud rate of the serial port to 115200bps, the data-bit is 9, the stop-bit is 1, and the check mode is odd check. (5) Enable serial port transceiver mode. (6) Write the interrupt function of USART1 and enable the interrupt.

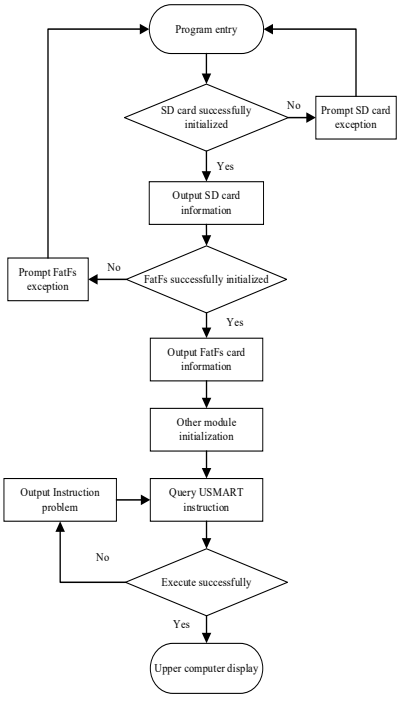

Fig. 7 Flow chart of serial port status control

In the process of serial port configuration, special attention should be paid to the 9 data bits, where the eighth bit is the data and the ninth bit is the parity check bit. The parity check mode corresponds to the single-chip microcomputer. After completing the above configuration process and main program preparation, the MCU can realize analog signal acquisition, serial port transmission, file management, and other functions.

Figure 7 shows the flow chart of using the serial port to control the lower computer. After the initialization of the ADC/DMA, file system, SD card, and other modules, the MCU can realize the corresponding functions through the commands and parameters provided by USMART. However, if the command or parameter is incorrect, the program will automatically return the error reason and output it to the serial port to prompt the user to verify.

3 The technical scheme design of upper computer

The upper computer receiving software for the marine controlled source online transmitter system is designed using the Python programming language. Python is a cross-platform high-level programming language with good portability and perfect functional modules, which can meet the needs of online data transmission. Additionally, Python has unique advantages in machine learning

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

and data analysis. In the future, the Python platform can be used to analyze and learn the data in the file, providing real-time prediction and early warning for marine experiments during software iteration.

3.1 Python programming environment construction

The upper computer program is developed in PyCharm integrated development environment, and its configuration process is shown in the Table 1.

Table 1 PyCharm environment construction process

PyCharm programming environment setup	Function
Configure Anaconda	Provide interpreter, PyQt5 module and other tools
Add Qt Designer tool	Provides graphical design tools
Add PyUIC	Convert the .ui file into the .py file
Import serial module	Provide methods related to serial port
Import os module	Provide methods for document management systems
Import PyQt5 module	Provides the definition and usage of related widgets
Import time module	Provides methods for program delay
Import datetime module	Provides methods for system time
Import thread module	Provides multithreading related methods
Import socket module	Provides the underlying Ethernet-related approach

3.1.1 Configure Anaconda interpreter

The most important thing to build PyCharm environment is the choice of interpreter. Anaconda is an open source package manager. It contains more than 100 packages and dependencies, such as Panda (a tool set for analyzing structured data), NumPy (a module that supports a large number of dimensional arrays and matrix operations, and also provides a large number of mathematical function libraries for Array Operations), PyQt (a toolkit for creating GUI applications), etc. This design uses Anaconda as the interpreter environment of PyCharm, and uses PyQt5 contained in Anaconda for UI design.

3.1.2 Import of function modules

This design needs to transmit data files and control commands through serial port, so serial communication is the most basic function. As a Python module, serial integrates most of the commonly used serial port configuration methods, so it is necessary to import the serial module into the project. Use “import serial” statement to import serial module.

This design needs to extract the data files in the lower computer to the PC, so the OS file management module is essential. Through the method of the OS module, a series of operations such as adding, deleting, and rewriting can be realized. Use “import OS” statement to import the OS module.

PyQt5 is a Python module under Qt framework, which contains hundreds of classes and thousands of functions and methods. This design mainly uses two modules in PyQt5, one is Qt Core (including core non-GUI functions, such as process time, files and directories, various data types, streams, URL, MIME types, threads, and processes). The second is Qt Widgets (the module contains the classic desktop style user interface and provides a set of UI element classes). Use “PyQt5 import Qt Widgets”, “from PyQt5.QtCore import Qt timer” statements to import module respectively.

This design needs to make the RTC of MCU synchronize with PC, so it needs to obtain the system time of upper computer for MCU. The main function of datetime module is to enable the

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

program to obtain the current time of the system and output it in the specified data format. Import the datetime module need to use “import datetime” statement.

Time module provides the function of program delay, which is mainly used to reserve sufficient time for serial transmission. Use “import time” to import the time module.

3.2 Use of multithreading

The smooth operation of the upper computer software depends on the repainting of UI interface. However, both serial file transfer and Ethernet file transfer will interrupt the redrawing process, leading to software death. In order to solve this problem, we need to use multithreading in the program.

Multithreading is not that multiple tasks run at the same time, CPU switches between different tasks quickly according to the needs, and achieve the effect which likes running at the same time. Under Windows platform, the independence of each task in multi process is better than that in multithreading, but its system occupancy is much higher than that in multithreading. Therefore, this program uses multithreading. As shown in Figure 8, the program is mainly divided into the following three threads: 1. The main thread, mainly responsible for command processing and UI interface redrawing. 2. Serial port transmission thread, mainly responsible for the construction of serial port channel, the receiving of serial command and data file. 3. Ethernet thread, mainly responsible for the construction of Ethernet channel and Ethernet file transmission.

Through “thread(target=thread name)” method to create the process. Through “threading.Event” method to set flag events for each thread to control the start or stop.

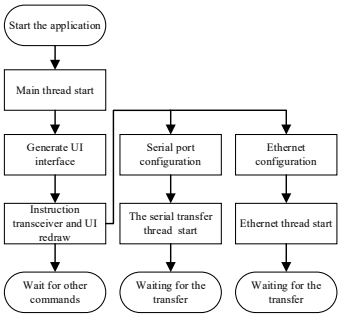

Fig. 8 Thread workflow

3.2.1 Construction of serial transmission thread

The serial module is needed to build the serial port channel for the upper computer software. The construction process is divided into the following steps: (1) Use “serial.Serial” method to create a serial port object. (2) Use “self.ser.port” method to set the serial port to be opened. Use “self.ser.baudrate” method to set the baud rate. Use “self.ser.bytesize” method to set the number of bits of data transmission and “self.ser.stopbits” method to set the stop-bit and “self.ser.parity” method to set the parity bit. (3) Use “try” statement to call the “self.ser.open” method. Open the serial port according to the above settings. If the serial port can be opened, it will display that the serial port has been opened, and enable to close the serial port button, and set the open serial port button to the unselected state. If it cannot be opened, port error is displayed.

3.2.2 Ethernet thread construction

The upper computer software requires the socket module to build the server. The construction process is divided into the following steps: (1) Use the "socket.gethostname" method to obtain the

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

local IP address. (2) Use the "socket.socket(AF_INET, SOCK_STREAM)" method to create the Ethernet interface. (3) Use the "sk.bind" method to bind the local address. (4) Use the "sk.setsockopt(SOL_SOCKET, SO_RCVTIMEO, 1000)" method to set the timeout to 1000ms. (5) Use the "sk.accept" method to wait for the client to connect.

3.3 UI design based on Qt Designer

To design a UI interface in PyQt5, it can be achieved by writing code directly or by using the graphical design interface of Qt designer. Compared to directly achieving the UI through the code, the graphical interface can be more intuitive to see the relationship between the various controls, and the designer will have a better grasp of the overall interface. Qt designer is used to design the UI and can be added to PyCharm's external tools from the Anaconda package. The UI interface design results are shown in Figure 9.

The .ui file generated by Qt designer cannot be used directly. PyUIC is needed to make Qt designer and Python work together. PyUIC is an automated scripting tool that can convert .ui files into executable .py files for Python.

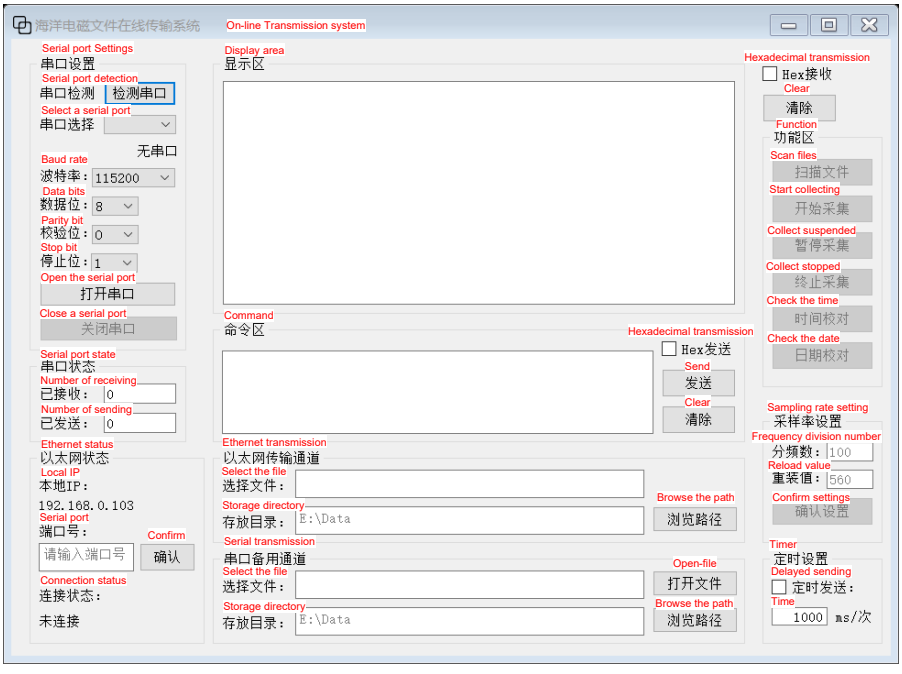

Fig. 9 UI interface

3.4 Compilation of function

3.4.1 Create class

After using Qt designer to complete the design of UI, the generated. UI file is saved in PyCharm project directory, and the .ui file is converted to .py file by using PyUIC. This file will contain a file called “UI_Form” class, which contains the basic information of the designed UI. In their own projects, “UI_Form” is imported as a module, and then a subclass “My_Client” is created and make it inherit “UI_Form” properties, and an instance of UI interface is built.

3.4.2 Define methods and connect to widgets

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

In order to implement functions in a UI interface, each widget needs to be given corresponding methods. The initialize function is defined, in which the connect method is used to connect the function and the widget by using the syntax "self.widget_name.signal.connect (self.function_name)".

For button widgets, such as "self.open_button.clicked.connect(self.port_open)", this means that when the "open_button" is pressed, the program will execute the "port_open" function.

For text widgets, such as "self.s1__box2.currentTextChanged.connect(self.port_imf)", this means that when the value of "s1_box2" is changed, the program will execute the "port_imf" function.

For check box widgets, such as "self.timer_send_cb. stateChanged.connect(self. data_send_timer)", this means that when the checkbox is selected, the program executes the "data_send_timer" function.

According to the above method, each widget is given a corresponding function.

3.4.3 Write the function corresponding to each widget

Table 2 shows the names and functions of each widget in the program. The following is a brief introduction to the realization process of several main functions.

Table 2 function names and functions

Control items	Function names	Function performance
port_check_button	port_check	Find out the serial port
port_open_button	port_open	Open the serial port
port_close_button	port_close	Close the serial port
data_send_button	data_send	Send data
data_receive_browser	data_receive	Receive data
send_clear_button	send_data_clear	Clear the data in the area
file_open_button	file_open	Open the file
copy_file_button	copy_file	Transfer files
sampling_rates_button	sampling_rates	Set sampling rate
time_button	set_time	Set time
write_button	write_in	Data recording
finish_button	finish_write	Stop recording
send_timer_button	data_send_timer	Timed transmission

The "port_close" function is used to close the current serial port. This function first stops the timer and stops transferring the contents of the buffer. It then uses the "try" statement to call the "self.ser.close" method to close the serial port. Finally, it clears all the buffer areas and display area contents, resets all widgets to the default state, and changes the serial port status to "closed".

The "data_send" function sends the content of the command area through the serial port. This function first uses the "toPlainText" method to format the text of the command area to the local variable "input_s" when the serial port has been opened. It then checks whether the hexadecimal sending status bar is selected, and if so, the "input_s" will be converted into hexadecimal. After processing the data, the function uses the "self.ser.write" method to write it to the serial port.

The "data_receive" function displays the data sent by the lower computer to the host computer in the receiving area through the serial port. This function first uses the "self.ser.inWaiting" method to return the length of the data in the buffer and stores it in "num". It then uses the "self.ser.read" method to read out the data. In this process, the value of "num" is recorded and accumulated to get the total amount of data received. The "textcursor" method is used to obtain the position of the cursor in the receiving area, and at the same time, the "movePosition.end" method is used to move the cursor target to the bottom (to ensure that the cursor remains at the end of the data write).

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023


Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

The "send_data_clear" function uses the "setText" method to set the contents of the command area to an empty string.

The "receive_data_clear" function uses the "setText" method to set the contents of the display area to an empty string.

The "file_open" function uses the "toPlainText" method to obtain the file name specified in the address bar, and the content of the command area is set to the function that the lower computer can recognize to open the file. The "ser.write" method is used to send the command.

The "copy_file" function uses the "os.chdir" method to change the program running directory to the specified location, and the files specified in the lower computer are extracted to the upper computer. The "ser.inWaiting" method is used in the process of transmission to determine whether the transmission is completed, and a prompt is given in the command area after the transmission is completed.

The "sampling_rates" function uses the "self.text" method to obtain the frequency division number and the reload value input by the user. The current sampling rate is calculated by the formula 8400000/(ARR×PSC) and displayed in the command area. Finally, the command of changing the sampling rate is sent to the lower computer.

The properties in the 'init' function are automatically generated when the class is instantiated. The "super(My_Client, self).init" method is used to call the function from its parent class to complete initialization. The "setupUi" method from the parent class is called to make its UI interface layout according to the designed UI interface. The name of the form is set to "marine electromagnetic file online transmitter system" using the "setWindowTitle" method. The "serial.Serial" method is used to configure the API functions related to the serial port.

After the above process, we have created a class that meets the expected function. For Python, it is equivalent to completing a drawing. Next, we should instantiate it. Firstly, the "QtWidgets.QApplication(sys.argv)" method is used to create a form application with parameters ("sys.argv" is the bridge between the program and the external parameters and points the external action to the program). Then, the designed class "My_Client" is used to create an instance named "my_show", and the "show" method is used to visualize the program window. Finally, the "sys.exit(app.exec_())" method is used to leave an exit for the program.

4 Joint tests

Figure 10 shows the actual situation of the joint test of the lower computer and the upper computer.

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

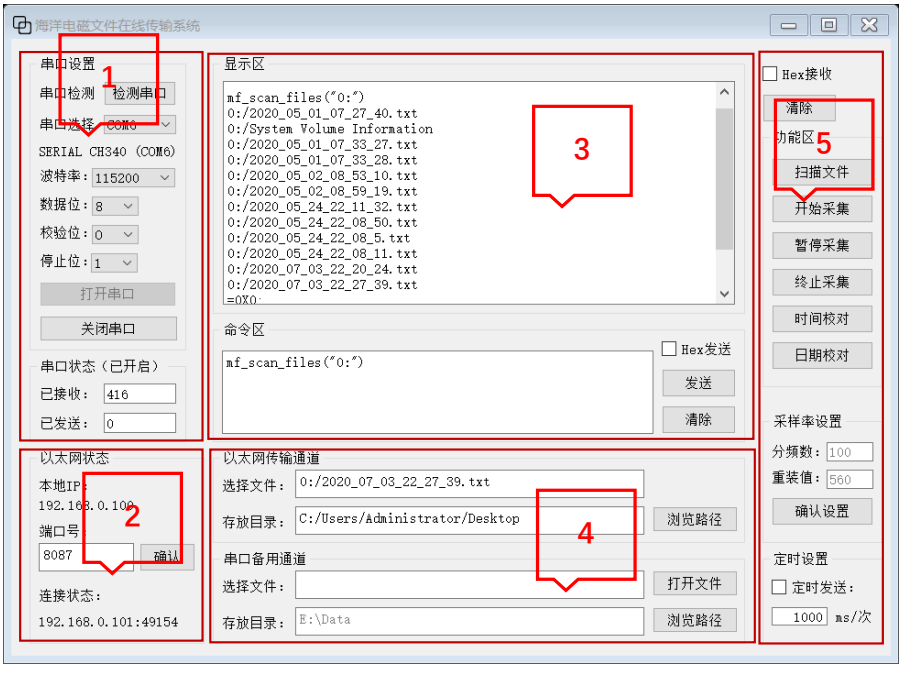

Fig. 10 Software running status

Area 1 is the serial port configuration area, where the baud rate is set to 115200bps, the data is set to 8-bit, the verification mode is set to ODD, and the stop-bit is set to 1. After the configuration is completed, the user can see that the serial port is successfully opened, and the port is set to COM6.

Area 2 is the Ethernet configuration area, where the local IP is automatically obtained by the software, and the port number is set to 8087. After clicking the confirm button, the user can see the remote IP and port in the connection status area.

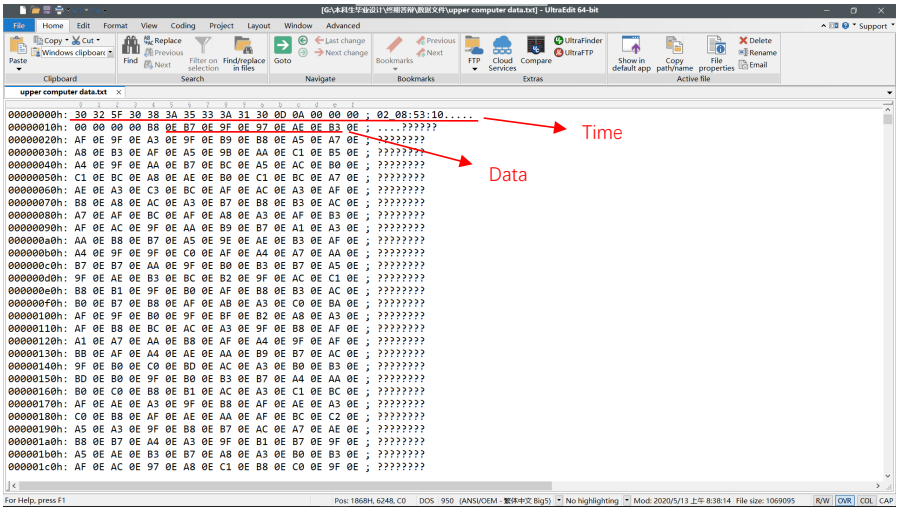

Fig. 11 Data file extracted to upper computer (Ultra Edit)

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

Area 3 is the command area and display area. In the figure, it shows the file query function, and the display area shows all files and folders under the root directory of the SD card.

In order to verify the accuracy of data transmission, it is necessary to compare the data files stored in the SD card with those extracted to the upper computer. As shown in Figure 11 and Figure 12, we can see that the byte difference between the two files is zero, that is, the two files are equal. After many tests, the accuracy of the file on-line transmitter system in this paper is proved.

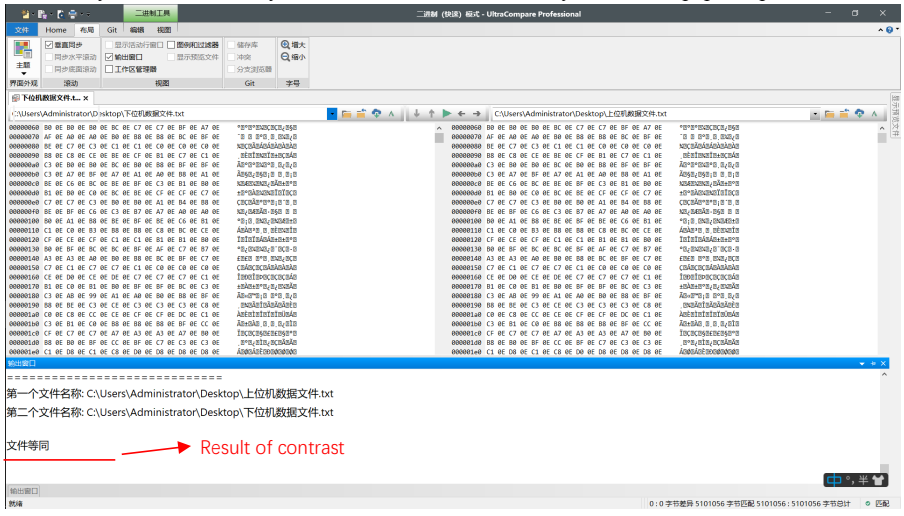

Fig. 12 Comparison of data files (Ultra Compare)

In the transmission process of a standard data file with a size of 10 MB, we tested its transmission rate, including serial transmission rate and Ethernet transmission rate. The Ethernet transmission rate is much higher than the serial transmission rate. The average Ethernet transmission speed is 600 KB/s compared to the 10 KB/s of serial communication rate. The data file transmission can be completed in about 15 seconds, as shown in Figure 13.

Compared with traditional data transmission methods, on-line transmission has significant advantages in real-time data transmission. The data can be transmitted in real-time, allowing for immediate decision-making and action to be taken. This is especially important in marine operations, where timely information can prevent accidents.

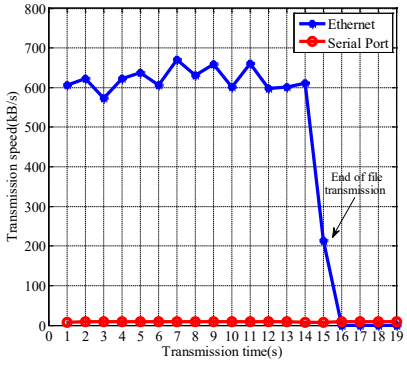

Fig. 13 Transmission rate test of 10M data file

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023


Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

4 Conclusion

The technical scheme of the marine controlled source electromagnetic current file on-line transmitter system uses STM32 as the lower computer to complete the functions of data acquisition, data storage, and file transmission. The upper computer software is designed using Python language, allowing users to control the single-chip microcomputer and acquire data files through the upper computer software.

In the lower computer module, the ADC is used to accurately collect the analog current signal, and the DMA double buffer mode is used in the acquisition process, which greatly improves the utilization of CPU. The FatFs file system is used to save the collected data to the SD card in the form of a file. After receiving the command from the upper computer, the data file can be transmitted to the upper computer through the serial port and Ethernet.

In the receiving software of the upper computer, Python language is used to realize the basic serial communication and Ethernet communication functions, and the UI design of the upper computer software is completed with the help of Qt designer. The software uses multithreading technology to solve the problem of program death in the process of file transmission.

In the joint test of the upper computer and the lower computer, the serial communication, file transmission, time correction, data acquisition, and other functions are tested. The final test result shows that the on-line transmitter system can run stably for a long time, and the upper computer can successfully control the lower computer. Data files can be obtained from the lower computer to the upper computer, and the Ethernet file transfer rate can reach 600 KB/s, which can meet the needs of marine operations. By using data comparison software, the difference between the data files in the SD card and the data files transferred to the upper computer is compared, and the results prove the accuracy of data transmission.

The on-line system for marine controlled source electromagnetic data not only provides great convenience for marine experiments but also provides security for the experimental data. In the future, software optimization can take advantage of Python language in machine learning and data analysis, so that it can provide a real-time visual interface, complete warning and prediction, and the on-line transmitter system will provide greater assistance for marine experiments.

5 Statement

This manuscript satisfies the following statements that: 1) all authors agree with the submission, 2) the work has not been published elsewhere, either completely, in part, or in another form, and 3) the manuscript has not been submitted to another journal.

6 Reference

[1] S. Y. Li, C. Y. Gu, J. Y. Yang, Y. Zhang, S. Diao, and Y. J. Ji, "A review of marine controlled-source electromagnetic data preprocessing technology," (in English), *Aip Adv,* vol. 12, no. 9, Sep 1 2022, doi: 10.1063/5.0090082.

[2] S. Constable, "Marine electromagnetic methods—A new tool for offshore exploration," *The Leading Edge,* vol. 25, no. 4, pp. 438-444, 2006, doi: 10.1190/1.2193225.

[3] S. Constable, "Ten years of marine CSEM for hydrocarbon exploration," *GEOPHYSICS,* vol. 75, no. 5, pp. 75A67-75A81, 2010, doi: 10.1190/1.3483451.

[4] M. WANG *et al.*, "Marine controlled source electromagnetic launch system for natural gas hydrate resource exploration," *Chinese Journal of Geophysics,* vol. 56, no. 11, pp. 3708-3717, 2013.

https://doi.org/10.5194/gi-2023-6
Preprint. Discussion started: 9 June 2023

Geoscientific Instrumentation Methods and Data Systems Discussions
Open Access
EGU

[5] D. Connell and K. Key, "A numerical comparison of time and frequency-domain marine electromagnetic methods for hydrocarbon exploration in shallow water," *Geophysical Prospecting,* vol. 61, no. 1, pp. 187-199, 2013.

[6] L. Yu-Guo and S. Constable, "Transient electromagnetic in shallow water: insights from 1D modeling," *Chinese Journal of Geophysics,* vol. 53, no. 3, pp. 737-742, 2010.

[7] S. Wang, S. Constable, V. Reyes-Ortega, and C. A. Rychert, "A newly distinguished marine magnetotelluric coast effect sensitive to the lithosphere–asthenosphere boundary," *Geophysical Journal International,* vol. 218, no. 2, pp. 978-987, 2019.

[8] S. Constable, P. K. Kannberg, and K. Weitemeyer, "Vulcan: A deep-towed CSEM receiver," *Geochemistry, Geophysics, Geosystems,* vol. 17, no. 3, pp. 1042-1064, 2016.

[9] Chen K, Jing J E, and Wei W B, "Numerical simulation and electrical field recorder development of the marine electromagnetic method using a horizontal towed-dipole source," *Chinese J. Geophys,* vol. 56, no. 11, pp. 3718-3727, 2013.

[10] M. Wang, M. Deng, Z. Wu, X. Luo, J. Jing, and K. Chen, "The deep-tow marine controlled-source electromagnetic transmitter system for gas hydrate exploration," *Journal of Applied Geophysics,* vol. 137, pp. 138-144, 2017.

[11] M. WANG, M. DENG, Z.-L. WU, X.-H. LUO, J.-E. JING, and K. CHEN, "New type deployed marine controlled source electromagnetic transmitter system and its experiment application," *Chinese Journal of Geophysics,* vol. 60, no. 11, pp. 4253-4261, 2017.

[12] N. Duan, M. Wang, G. Wang, P. Yu, M. Deng, and X. Li, "Research on the isolation and collection method of multi-channel temperature and power supply voltage under strong marine controlled source EMI," *IEEE Access,* vol. 7, pp. 6400-6411, 2018.

[13] M. WANG, M. DENG, P. YU, C. YIN, K. CHEN, and X. LUO, "High-power time-frequency transmission and multi-chain cable multi-component electromagnetic system for deep-water exploration," *Chinese Journal of Geophysics,* vol. 65, no. 9, pp. 3664-3673, 2022.

[14] K. Chen, M. Deng, P. Yu, Q. Yang, X. Luo, and X. Yi, "A near-seafloor-towed CSEM receiver for deeper target prospecting," *Terrestrial, Atmospheric & Oceanic Sciences,* vol. 31, no. 5, 2020.