**RC1:**

Q1. The Ethernet transmission rate mentioned in the text is limited by the clock speed of the controller itself. However, theoretically, a higher transmission rate can be achieved through optical fiber transmission. This point should be clarified in the text.

Q2. This article detailedly addresses the issue of data exchange rate in marine electromagnetic construction and establishes a relatively comprehensive system. However, it is worth noting that there has strong electromagnetic interference in marine electromagnetic transmitters. The accuracy mentioned in the text needs to be verified under such conditions of strong electromagnetic interference. If the verification has been conducted, it should be mentioned in the text. If not, it is recommended that the author clearly state this fact for the reference of peers.

Q3. The SD card storage solution proposed in the article, can it support long-term underwater data collection? How long can it specifically achieve? It is hoped that the author should indicate this in the article.

A data processing software implemented using Python, and the real-time uploading and processing of data mentioned in this article is a novel direction. If real-time data can provide reference and assistance for offshore construction operations, it would be excellent. I am deeply interested in this and will pay attention to your future progress.

**AC:**

Thank you very much for raising your questions. I will answer them in order.

1.The transmission rate of fiber optic is indeed not limited to this. The STM32 we use has a clock frequency of 168MHz, and it can only support up to 100Mbps Ethernet. Furthermore, due to the numerous tasks it needs to handle internally, it does not run the Ethernet transmission thread separately, thus unable to achieve the ideal transmission rate. In our offshore operations, the sampling rate is 150Hz, and the current transmission speed is already able to meet our data transmission needs. This point will be mentioned in the subsequent revisions of this article.

2.Our equipment has undergone corresponding EMC verification, including but not limited to fast pulse group, electromagnetic radiation and interference resistance, electrostatic and insulation testing. However, this article primarily focuses on method elucidation, so excessive elaboration on hardware design is avoided. Based on your feedback, we will include an analysis of this aspect.

3.As mentioned in the first question, our data sampling rate is 150Hz, so the approximate data volume for one hour is around 1.3GB. The instrument is usually retrieved after a period of operation, and the data is also uploaded in real time. The purpose of the SD card is to ensure data safety in the event of unexpected network disconnection, rather than to store all the data locally.

**RC3:**

Questions：

Q1. Could you please provide a summary of the data flow between different modules in the entire underwater system?

Q2. What is the power supply method used for the underwater terminal system? Considering its extended underwater operation, is energy conservation a significant consideration?

Q3. How does the software manage the collected data? Is it stored directly or processed in any specific way?

Suggestions：

1. The authors have made significant contributions to the development of both the hardware and software components of the MCSEM system. They have also implemented a network-free monitoring and recording storage function, which facilitates timely identification and elimination of potential hazards. Consequently, the system's robustness and reliability have been enhanced. This work presents an alternative solution for the design and implementation of MCSEM systems. However, the reviewer points out that the main issue lies in the authors' failure to adhere to the requirements of general scientific papers when preparing the manuscript. For instance, the introduction section lacks clarity in presenting the research background, current research status, problems to be addressed, and the challenges associated with these problems. Additionally, it is unclear whether any other research groups have proposed solutions to these issues.

2. Inconsistencies in English grammar have been identified in the text, indicating the need for a thorough English proofreading review.

3. Real-time processing of marine electromagnetic data represents a relatively new research direction for the future. However, it is currently limited by the computing capabilities available on board. It is suggested that the author effectively integrate the existing software with the data transmission system in future research.

Overall, I appreciate the content of the authors' work; however, some modifications are necessary to enhance the novelty of the paper, establish its differentiation from existing research, and address grammar-related concerns.

**AC:**

First of all, thank you very much for pointing out the issues. In the subsequent revisions, I will carefully proofread and optimize the text, as well as add comparisons with peers to analyze the innovation of the article. Secondly, I will answer your questions one by one below.

1. The overall data flow is shown in Figure 3. It may cause misunderstandings because I did not label the data interaction between each module in the figure. GPS synchronizes via serial port, PPS transmits in pulse form, and other peripherals mainly communicate via CAN and SPI. After being aggregated to the main control unit, they are uniformly uploaded by the main control unit.

2. Our optoelectronic composite cable has power transmission function, which can continuously supply power underwater. Underwater power conversion adopts the combination of transformer + charger + battery, so there is no need to overly concern about power. However, our design for the underwater end focuses on low power consumption.

3. The upper computer software is mainly responsible for data reception, segmentation, and packaging. Specific data will be processed by dedicated software.

**RC4:**

This article presents the development process of an online status monitoring system within the context of MCSEM .I appreciate the work done in the article， it is innovative and can effectively improve the efficiency of MCSEM offshore work, providing guidance for marine exploration. Therefore, I agree with the publication of this article, but I have some questions and concerns regarding certain key technologies. I will present my suggestions and questions below.

Q1. It seems that the introduction is missing a comparison with the work of other individuals or teams. A rigorous scientific article should provide a detailed discussion to demonstrate the novelty of the work conducted.

Q2. In Figure 3, what is the necessity of the mentioned parallel transmission of serial port and Ethernet? Additionally, Figure 3 appears to be incomplete as it does not provide information about the data transmission methods between the different modules.

Q3. In section 2.1.4, the USMART technology is mentioned. What are the functions available for invocation in this module? What functionalities do they respectively implement?

Q4. The images and flowcharts in the article need to be carefully formatted to avoid inconsistent text sizes （Figure 5 and 6 and 7）.

**AC:**

First of all, thank you very much for your question. Regarding the comparison with other people's work that you mentioned, there is indeed a lack in the paper. I will consult more materials and include these comparisons in the subsequent revisions.

For the second question, the online transmission system we designed consists of two transmission channels. The serial port transmission channel is mainly responsible for transmitting control commands, while the Ethernet transmission channel is mainly responsible for transmitting a large amount of data. These two channels do not interfere with each other. This consideration is based on the requirements for data file integrity and accuracy, as well as the limitations of the lower-level hardware.

For the third question, with USMART, you can easily modify function parameters and view function execution results, enabling you to quickly solve problems. Modifying parameters is very convenient as it doesn't require compilation or downloading, making it easy to test and debug code.It mainly includes the following functional functions:

(void*)read_addr,"u32 read_addr(u32 addr)",
(void*)write_addr,"void write_addr(u32 addr,u32 val)",
(void*)delay_ms,"void delay_ms(u16 nms)",
(void*)delay_us,"void delay_us(u32 nus)",
(void*)ETH_ReadPHYRegister,"uint16_t ETH_ReadPHYRegister(uint16_t PHYAddress, uint16_t PHYReg)",
(void*)ETH_WritePHYRegister,"uint32_t ETH_WritePHYRegister(uint16_t PHYAddress, uint16_t PHYReg, uint16_t PHYValue)",
(void*)W25QXX_Erase_Chip,"void W25QXX_Erase_Chip(void)",
(void*)mf_mount,"u8 mf_mount(u8* path,u8 mt)",
(void*)mf_open,"u8 mf_open(u8*path,u8 mode)",

(void*)mf_close,"u8 mf_close(void)",
(void*)mf_read,"u8 mf_read(u16 len)",
(void*)mf_write,"u8 mf_write(u8*dat,u16 len)",
(void*)mf_opendir,"u8 mf_opendir(u8* path)",
(void*)mf_closedir,"u8 mf_closedir(void)",
(void*)mf_readdir,"u8 mf_readdir(void)",
(void*)mf_scan_files,"u8 mf_scan_files(u8 * path)",
(void*)mf_showfree,"u32 mf_showfree(u8 *drv)",
(void*)mf_lseek,"u8 mf_lseek(u32 offset)",
(void*)mf_tell,"u32 mf_tell(void)",
(void*)mf_size,"u32 mf_size(void)",
(void*)mf_mkdir,"u8 mf_mkdir(u8*pname)",
(void*)mf_fmkfs,"u8 mf_fmkfs(u8* path,u8 mode,u16 au)",
(void*)mf_unlink,"u8 mf_unlink(u8 *pname)",
(void*)mf_rename,"u8 mf_rename(u8 *oldname,u8* newname)",
(void*)mf_getlabel,"void mf_getlabel(u8 *path)",
(void*)mf_setlabel,"void mf_setlabel(u8 *path)",
(void*)mf_gets,"void mf_gets(u16 size)",
(void*)mf_putc,"u8 mf_putc(u8 c)",
(void*)mf_puts,"u8 mf_puts(u8*c)",
(void*)send_task,"void send_task(u8*file_path)",
(void*)RTC_Set_Time,"u8 RTC_Set_Time(u8 hour,u8 min,u8 sec,u8 ampm)",
(void*)RTC_Set_Date,"u8 RTC_Set_Date(u8 year,u8 month,u8 date,u8 week)",
(void*)RTC_Set_AlarmA,"void RTC_Set_AlarmA(u8 week,u8 hour,u8 min,u8 sec)",
(void*)RTC_Set_WakeUp,"void RTC_Set_WakeUp(u8 wksel,u16 cnt)",
(void*)Write_In,"void Write_In(void)",
(void*)Stop_Write,"void Stop_Write(void)",
(void*)Finish_Write,"void Finish_Write(void)",
(void*)TIM3_Int_Init,"void TIM3_Int_Init(u16 arr,u16 psc)",
(void*)stopsend_task,"void stopsend_task(void)",

For question 4, thank you very much for pointing out the issue. I will make careful revisions accordingly.

In conclusion, I would like to express my gratitude for your diligent review. Your suggestions and questions have been very helpful for my future work, and I appreciate your recognition of our efforts.