Real-Time Subsea Turbulence and Sediment Monitoring System Software Requirements Specification

Prepared for: Dr. Boris Bam Nges

Abdulla Sadoun B00900541

Sohaib Zahid B00908166

Table of Contents

1. Introduction

1.1. Purpose

The purpose of this SRS document (Software Requirements Specification) is to detail the requirements for the Real-Time Subsea Turbulence and Sediment Monitoring System that we are trying to design, implement and test for our Real Time Systems course project. This embedded system collects real-time data from environmental sensors to monitor underwater sediment and turbulence levels, providing valuable insights for marine management and coastal protection. The intended audience includes project stakeholders, development engineers, and marine environmental analysts.

1.2. Scope

This document covers the design and implementation procedure, constraints and limitations as well as the functionality of the real-time monitoring system built on the STM32F411 microcontroller using FreeRTOS as we have been doing in lab the past few weeks, the system is capable of collecting and analyzing data from multiple sensors to assess underwater turbulence and sediment levels remotely. The system's objectives align with those outlined in the Call for Projects document, addressing oceanic environmental challenges by leveraging embedded systems and real-time data processing capabilities.

Data Acquisition, Real-Time Data Processing, Threshold-Based or Constrained Data Alerts and decision making as well as controller/sensor Communication are the key components in this real time system which would satisfy our main objective of Real-Time Subsea Turbulence and Sediment Monitoring.

Data Acquisition would be accomplished by compiling the signals from the sensors (pressure, flow rate and acoustic) to collect valuable and useful data. This data would be processed and used to make decisions or provide alerts based on predefined constraints for different values and data ranges which would then be communicated (transmitted through USART channel) to connect our system.

This project is inspired by and follows a similar design and implementation structure to the remote underwater sensor platform from lab 4. Similarly, our system will use the STM32 microcontroller provided, sensors or simulated sensor signals from another board if possible, and FreeRTOS, but we will focus on environmental sediment and turbulence monitoring to drive informative decision making by setting appropriate constraints and data processing algorithms to make these decisions or provide useful alerts for the user/decision maker.

1.3. Definitions, Acronyms, and Abbreviations

STM32F4: A family of ARM Cortex-M4 microcontrollers from STMicroelectronics.
STM32F411: The STM32 ARM-based Cortex-M4 microcontroller that we are using for our solution.

FreeRTOS: A free open source real time operating system for embedded systems, used for task scheduling, ISRs, timers, IOT, setting/controlling pins, controlling I/O devices and controlling or setting up various other tasks that the microcontroller is capable of.
USART: Universal Synchronous/Asynchronous Receiver-Transmitter, a serial communication protocol.
RTOS: Real-Time Operating System e.g. FreeRTOS.
SRS: Software Requirements Specification (This Document).
IEEE: Institute of Electrical and Electronics Engineers

1.4. References

- IEEE. IEEE Guide to Software Requirements Specifications. ANSI/IEEE Std 830-1984. IEEE, 1984.
- **MS5837-30BA Pressure Sensor Datasheet**: TE Connectivity. (n.d.). *MS5837-30BA Miniature Pressure Sensor*. Retrieved from TE Connectivity Datasheet.
- **SEN-HZ21WA Flow Rate Sensor Datasheet**: e-Gizmo. (n.d.). *SEN-HZ21WA Flow Meter Specifications*. Retrieved from e-Gizmo Datasheet.
- **HTI-96-MIN Hydrophone Datasheet**: High Tech, Inc. (n.d.). *HTI-96-MIN Hydrophone Specifications*. Retrieved from High Tech Inc. Datasheet.
- **DS18B20 Temperature Sensor Datasheet**: Maxim Integrated. (n.d.). *DS18B20 Programmable Resolution 1-Wire Digital Thermometer*. Retrieved from Digi-Key Datasheet.
- **STM32F411 Nucleo-64 User Manual**: STMicroelectronics. (n.d.). *UM1724: STM32 Nucleo-64 Boards (MB1136) User Manual*. Retrieved from STMicroelectronics Manual.

1.5. Overview

This document is organized into the following main sections following IEEE's SRS outline:

General Description: Provides an overview of the system's purpose, functions, constraints, and dependencies.

Specific Requirements: Defines detailed functional and non-functional requirements, covering data acquisition, processing, communication protocols, and system constraints.

Supporting Information: Includes appendices for definitions, symbols, and an index for easy navigation.

2. General Description

2.1. Product Perspective

The Real-Time Subsea Turbulence and Sediment Monitoring System is an independent embedded system designed to measure underwater turbulence and sediment levels in real-time, leveraging the STM32F411 microcontroller and FreeRTOS. The system uses a combination of pressure, flow rate, and acoustic sensors to collect and process data used to make decisions based on predefined constraints or flag alerts that can prove helpful when detecting sediment buildup and turbulence events.

Our system utilizes the STM32 platform with FreeRTOS to perform real-time data acquisition, manage tasks, inter-task communication and processing as we have seen in previous labs to come up with a containerized embedded solution targeting environmental assessment related to sediment and water turbulence.

USART will be used as our main communication protocol and interface and our pressure, flow rate, and acoustic sensors used for data acquisition might need to be simulated as sensor signals outputted from another microcontroller if they are not available.

2.2. Product Functions

Our product is engineered to execute the following essential functions:

1. Sensor Data Acquisition: It continuously collects data from pressure, flow rate, and acoustic sensors, facilitating real-time observation of environmental conditions.

2. Real-Time Data Processing: It employs real-time algorithms to evaluate sensor data and assess levels of sediment and turbulence.

3. Threshold-Based Alerts: It activates alerts when predefined thresholds for sediment and turbulence are exceeded, thereby informing operators of possible hazards.

4. Data Communication: It transmits the processed information to a host device or interface via USART or an acoustic modem, allowing for remote monitoring.

2.3. User Characteristics

This system is designed for utilization by experts in marine environmental monitoring, which includes the following roles:

Marine Environmental Analysts: They utilize the data to examine sediment and turbulence trends.

Coastal Management Authorities: They track sediment concentrations and water turbulence to support environmental impact evaluations and management strategies.

Development Engineers: They are responsible for the maintenance, updates, and configuration modifications of the system in accordance with environmental needs.

2.4. General Constraints

The system is governed by the following constraints:

Power Limitations: Being an underwater device, it is essential to prioritize power efficiency, which necessitates low-power operation to extend battery life and operation time.

Environmental Conditions: The device must function effectively in underwater environments, where it may encounter varying pressures, temperatures, and salinity levels.

Real-Time Processing Requirements: Data processing must occur with minimal latency to facilitate prompt alert generation and communication, thereby requiring effective task scheduling within FreeRTOS.

2.5. Assumptions and Dependencies

The effective functioning of the system is contingent upon several key assumptions and dependencies:

1. Sensor Calibration: It is essential that all sensors are properly calibrated and protected for underwater environments, ensuring that accurate readings are consistently obtained over time.

2. Network and Communication Reliability: The system's operation is reliant on stable communication via USART or acoustic modems for the transmission of data some the sensors would use .

3. FreeRTOS Compatibility: The real-time performance of the STM32 microcontroller must be adequate to fulfill the processing requirements of FreeRTOS, thereby facilitating reliable data management and alert generation.

3. Specific Requirements

3.1. Functional Requirements

3.1.1. Data Acquisition

The system is required to consistently gather data from pressure, flow rate, and acoustic sensors to facilitate real-time monitoring of sediment levels and water turbulence.

3.1.1.1 Sensor Sampling: Each sensor will gather data at predetermined intervals (for instance, every second).

3.1.1.2 Data Precision and Filtering: The raw data obtained from the sensors must undergo filtering to minimize noise, thereby ensuring the accuracy of turbulence and sediment measurements.

3.1.1.3 Data Calibration: All sensor data should be calibrated in relation to environmental baselines to account for the influences of underwater pressure and temperature.

3.1.2. Real-Time Processing

Data obtained from the sensors needs to be processed instantaneously to enable prompt evaluation of sediment and turbulence levels.

3.1.2.1 Sediment Level Evaluation: The data from pressure sensors should be examined to ascertain the depth of sediment.

3.1.2.2 Turbulence Identification: Information from acoustic and flow rate sensors must be analyzed to detect variations in underwater turbulence.

3.1.2.3 Anomaly Identification: The system is required to recognize and categorize anomalies in the data, such as sudden spikes in sediment or turbulence, which may signal potentially hazardous conditions.

3.1.2.4 Optimization of Data Processing: The analysis of real-time data should be refined to eliminate system delays, ensuring that data processing remains within a limit of 500 milliseconds per cycle.

3.1.3. Communication Protocols

Data obtained from the sensors needs to be processed instantaneously to enable prompt evaluation of sediment and turbulence levels.

3.1.2.1 Sediment Level Evaluation: The data from pressure sensors should be examined to ascertain the depth of sediment.

3.1.2.2 Turbulence Identification: Information from acoustic and flow rate sensors must be analyzed to detect variations in underwater turbulence.

3.1.2.3 Anomaly Identification: The system is required to recognize and categorize anomalies in the data, such as sudden spikes in sediment or turbulence, which may signal potentially hazardous conditions.

3.1.2.4 Optimization of Data Processing: The analysis of real-time data should be refined to eliminate system delays, ensuring that data processing remains within a limit of 500 milliseconds per cycle.

3.1.4. Data Thresholds and Alerts

The system is required to establish and oversee specific thresholds for sediment and turbulence levels to facilitate real-time notifications.

3.1.4.1 Threshold Specifications: Define threshold values for sediment levels and turbulence intensity. For example:

- Low Sediment Level: Sediment depth < 10 cm.

- Moderate Sediment Level: 10 cm $\leq$ Sediment depth < 30 cm.

- High Sediment Level: Sediment depth ≥ 30 cm.

- Normal Turbulence: Acoustic and flow data remain within the anticipated range.

- High Turbulence: Acoustic and flow data surpass the established turbulence threshold.

These would be our default values, the user would be able to slightly alter them in our final product.

3.1.4.2 Alert Activation: The system must activate an alert when thresholds are exceeded. Alerts should include details such as sensor ID, event timestamp, and alert level (e.g., high sediment or high turbulence).

3.1.4.3 Alert Prioritization: Alerts should be ranked according to their severity, ensuring that high-risk incidents, such as abrupt increases in turbulence, are communicated without delay.

3.1.4.4 Alert Documentation: Each alert must be recorded with a timestamp and relevant sensor data for future analysis and system auditing.

3.2. Performance Requirements

3.2.1. Timing and Latency

The system is required to guarantee that data acquisition, processing, and communication are executed within defined time limits to uphold real-time functionality.

3.2.1.1 Data Acquisition Latency: The system must obtain data from each sensor within 200 milliseconds of the predetermined sampling interval.

3.2.1.2 Data Processing Latency: The system is obligated to process the sensor data and deliver output results within 500 milliseconds following data acquisition.

3.2.1.3 Communication Latency: Data transmission, whether through USART or an acoustic modem, must take place within 100 milliseconds after the completion of processing to facilitate prompt alerts and monitoring.

3.2.1.4 Alert Generation Timing: Alerts are required to be generated and communicated within 300 milliseconds after the detection of threshold crossing.

3.2.1.1 Data Collection Interval: The system must gather data from all sensors every second to ensure ongoing real-time monitoring.

3.2.1.2 Processing Latency: Data processing should be completed within 500 milliseconds per cycle, ensuring that new data is prepared for analysis prior to the arrival of the next data set.

3.2.1.3 Alert Response Time: The system must initiate and record alerts within 200 milliseconds of identifying a threshold breach, providing near-instantaneous feedback on critical occurrences.

3.2.1.4 Communication Delay: Data transmission from the underwater unit to a host or remote interface must not exceed 2 seconds via USART or 5 seconds through the acoustic modem, taking into account environmental communication limitations.

3.2.2. Reliability Metrics

The system is required to uphold a high level of reliability to guarantee uninterrupted operation and precise data acquisition.

3.2.2.1 Uptime Requirement: The system must attain a minimum operational uptime of 95% throughout each deployment phase, thereby ensuring dependable data collection and oversight.

3.2.2.2 Sensor Accuracy: Sensors are required to deliver measurements with an accuracy of ±5% of the reading across all anticipated ranges (such as pressure, flow, and acoustic).

3.2.2.3 Error Rate: The permissible error rate for data transmission must not surpass 2% of the messages sent, thereby ensuring the integrity of data in the monitoring process.

3.2.2.4 Fault Recovery: The system must incorporate fault recovery protocols to automatically restore functionality following a failure, with a recovery duration not exceeding 1 minute for non-critical errors.

3.2.3 Data Handling and Storage

The system is required to effectively manage and store the data it collects to enable subsequent analysis and monitoring.

3.2.3.1 Data Storage Capacity: The system should be capable of maintaining a minimum of 48 hours of uninterrupted data logging from all sensors without any data loss, utilizing local memory resources.

3.2.3.2 Data Logging Frequency: The system must record data from each sensor at a minimum frequency of once per second, ensuring that the temporal resolution is adequate for precise sediment and turbulence analysis.

3.2.3.3 Data Transfer Rate: The system must have the ability to transfer logged data to a host interface at a minimum rate of 9600 bits per second (bps) through USART.

3.2.4 User Interaction and Reporting

The system must offer user interfaces that allow for effective monitoring and configuration of system parameters.

3.2.4.1 User Response Time: The user interface must react to user inputs (such as commands to modify thresholds or request data) within one second.

3.2.4.2 Report Generation Time: The system must be able to generate and present summarized reports of sediment and turbulence levels within two seconds of a user request.

3.2.4.3 Alert Notification Mechanism: Alerts must be displayed on the user interface immediately upon detection, clearly indicating the type and urgency of the alert.

3.2.3 Environmental Resilience

3.2.3.1 Temperature Range: The system must operate accurately in underwater temperatures ranging from -2°C to +35°C, ensuring reliable data collection across various marine environments.

3.2.3.2 Pressure Resilience: The system must endure pressure variations corresponding to depths of up to 100 meters, ensuring consistent data collection and maintaining device integrity at these depths.

3.2.3.3 Waterproof and Corrosion Resistance: The enclosures must ensure waterproof integrity and resist corrosion from saltwater exposure, facilitating prolonged underwater operation without maintenance interruptions.

3.2.4 System Scalability

3.2.4.1 Data Handling Capacity: The system must accommodate up to three sensors simultaneously without compromising performance or processing speed.

3.3. Design Constraints

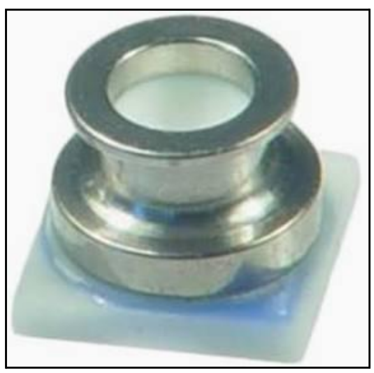
3.3.1. Hardware Compatibility

The system should be engineered to function effectively within the constraints of the STM32F411 microcontroller and other designated hardware components.

3.3.1.1 Microcontroller Constraints: All software and processing functions must be refined to operate within the STM32F411's 512 KB of Flash memory and 128 KB of SRAM according to their data sheet.

3.3.1.2 Sensor Integration: The design must facilitate the smooth integration of the specified pressure, flow rate, and acoustic sensors, ensuring accurate data collection without surpassing the microcontroller's I/O limitations. Sensor chosen for our product include:

MS5837-30BA Sensor for Pressure Monitoring:



The Sensor chosen above would be used to acquire the pressure around the deployed platform underwater. It is chosen as it can operate in depths of around 300 meter below sea level and up to 30 bar and it utilizes the I2C protocol for communication with our Nucleo board.

The sensor requires around 1.2mA of current while performing the measurements rated at around 3.3 to 5 Volts according to the datasheet.

SEN-HZ21WA Hall Effect Sensor:

The Sensor chosen above would be used to acquire the water flow measurement around the deployed platform under the water's surface. It is chosen as it can operate reliably in our temperature ranges at the required depths and can be easily implemented requiring no extra communication protocols. It utilizes a regular GPIO pin to provide pulses for communication and data transmission with our Nucleo board.

The sensor requires around 15mA of current while operating rated at around 3.3 to 5 Volts according to the datasheet.

HTI-96MIN Acoustic Sensor:

The Sensor chosen above would be used to acquire “sound” waves in the environment of the deployed platform underwater. This would be required to pick up the lower frequency waves which have been proven to bring a direct relation to currents and/or sediment movement. It uses a simple ADC pin for communication with our board.

The sensor requires around 2-3mA of current while performing the measurements rated at around 3.3 Volts according to the datasheet.

DS18B20 Temperature Sensor:

The Sensor chosen above would be used to acquire temperature readings around the deployed platform underwater. It uses one wire protocol, and needs around 0.5mA of power.

And for the board we will use the same Nucleo board which has an MAX current input of 100mA to operate and give power to its sensors and I/O properly with inputs around 300mA damaging both the port and the Host PC. Deducting the sensors out of the equation we would need only 20-40mA to operate basic functions that don't require I/O (pin) devices so we will take 30mA as the average for the final calculation.

This means we need a battery that drives the operation of all of these devices currents which is a little bit less than 50mA in total for around 5 days (to require weekly maintenance between deployments). Meaning we would need a battery that is around 50mA * 24hours *5 days=6000mA rated at around 5V. Which should be an extremely easy find and a wire that is mini-usb to connect to the board.

3.3.1.3 Power Efficiency: The system should aim to reduce power consumption to prolong battery life, incorporating sleep modes or low-power states whenever feasible to sustain efficiency during ongoing operations after calculation of the projected power consumption from the data sheets of all the sensors as well as the microcontroller, it looks like we would require a simple power bank that has around 6000mAH.

3.3.2. Environmental Conditions (e.g., underwater deployment constraints)

3.3.4.2 Acoustic Modem Protocol: In applications that necessitate acoustic communication underwater, it is essential that data packets adhere to the specified protocol format for the acoustic modem, with data being compressed to minimize transmission durations.

3.3.4.3 Interference Management: The system must be designed to consider potential interference in underwater signals, incorporating error-checking and retransmission capabilities as necessary to maintain data integrity.

3.3.3. Maintenance and Modularity

The system should be constructed to enable straightforward maintenance and upgrades, permitting future modifications without extensive system renovations.

3.3.5.1 Modularity: Our software architecture is modular, comprising distinct code segments for data acquisition, processing, and communication, which allows for seamless updates or alterations to individual modules as required so far all it needs is a simple battery recharge **EVERY 5 DAYS** if only one battery is deployed with it.

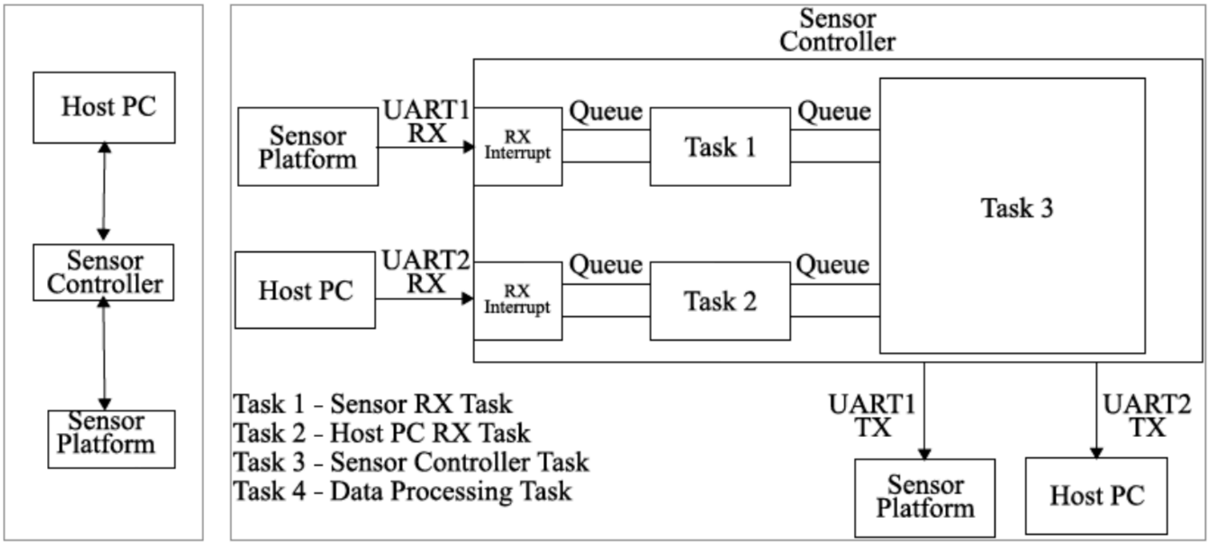
3.3.5.2 Firmware Update Capability: Our system facilitates firmware updates through a serial interface, enabling modifications or enhancements to be implemented without the need for hardware alterations.

3.3.5.3 Diagnostic Support: The system should incorporate diagnostic routines that can be activated remotely to assess sensor functionality, memory utilization, and communication capabilities, thereby aiding in troubleshooting and maintenance efforts.

3.4. Attributes

3.4.1. Modularity

The architecture of the system should facilitate modularity to enhance maintenance, troubleshooting, and future upgrades.



a) Network of Devices b) Detailed Structure of the System Controller

3.4.1.1 Code Modularity: The software of the system should be structured into separate modules dedicated to data acquisition, processing, communication, and alert management, ensuring well-defined boundaries and limited interdependencies.

3.4.1.2 Component Isolation: Hardware elements, including sensors and communication modules, should be integrated with minimal interaction between components, enabling the independent updating or replacement of individual parts without impacting the overall system.

3.4.2. Scalability

The system must possess scalability to facilitate potential future enhancements or the incorporation of new functionalities without necessitating significant redesign efforts.

3.4.2.1 Sensor Expandability: The system architecture is designed to permit the integration of additional sensors or various types of sensors, as long as they are compatible with the STM32F411 microcontroller.

3.4.2.2 Communication Protocol Flexibility: The communication framework should be adaptable to support alternative protocols (such as I2C and USART and SPI) should future expansions or modifications to the system be required.

3.4.2.3 Task Adaptability: FreeRTOS tasks should be structured to allow for the addition of new functionalities, ensuring that extra processing tasks can be integrated with minimal alterations to the core scheduling.

3.4.3. Reliability

The system is required to exhibit exceptional reliability to guarantee consistent and precise monitoring during prolonged underwater operations.

3.4.3.1 Fault Tolerance: It is essential to incorporate mechanisms for error detection and management to recognize and address prevalent issues such as sensor malfunctions, data corruption, and disruptions in communication.

3.4.3.2 Self-Check Routines: The system should conduct routine self-assessments to confirm the operational status of sensors, communication components, and memory, thereby ensuring dependable functionality throughout the deployment period.

3.4.3.3 Data Integrity: All data transmitted must incorporate checksums or alternative verification techniques to avert data corruption during the transmission process.

3.4.4. Maintainability

The system should be designed for ease of maintenance, incorporating features for diagnostic assessments and updates.

3.4.4.1 Remote Diagnostics: Enable remote diagnostic commands to oversee sensor conditions, communication pathways, and overall system integrity during operation.

3.4.4.2 Firmware Update Procedures: Implement a user-friendly firmware update process that facilitates in-field software enhancements to modify thresholds, adjust sensor settings, or rectify issues.

3.4.4.3 Error Logging: The system is required to keep a record of errors, system restarts, and critical occurrences to aid in troubleshooting and performance evaluation.

3.4.5. Portability

The system software must be developed with the capability to operate on various STM32 series microcontrollers, should future modifications necessitate alternative hardware configurations.

3.4.5.1 Code Portability: Whenever feasible, employ standard C libraries and STM32 HAL (Hardware Abstraction Layer) functions to facilitate the transition to other STM32 microcontrollers.

3.4.5.2 Platform-Independent Components: Structure software components to reduce reliance on hardware-specific features, thereby improving compatibility with additional STM32 devices.

3.4.6. Security

Security protocols are essential to safeguard the system from unauthorized access and data manipulation.

3.4.6.1 Data Encryption: Sensitive information transmitted through communication channels should be encrypted to prevent interception.

3.4.6.2 Access Control: Access to vital system functions, including firmware updates and threshold adjustments, should be limited to authorized individuals only.

3.4.6.3 Secure Firmware Updates: It is crucial to implement verification checks within firmware update processes to ensure the integrity of updates, thereby preventing unauthorized or compromised installations.

3.5. External Interface Requirements

3.5.1. Sensor Interfaces

The system is designed to connect with pressure, flow rate, and acoustic sensors, which are essential for effective data collection.

3.5.1.1 Pressure Sensor Interface: The system is required to establish a connection with the pressure sensor through either an analog or digital input channel, ensuring consistent readings of underwater pressure to evaluate sediment levels. Calibration parameters for the sensor should be integrated into the software to guarantee precise measurements across varying pressure conditions.

3.5.1.2 Flow Rate Sensor Interface: The flow rate sensor must be linked via an analog input or pulse input, delivering information regarding underwater flow velocity to analyze turbulence levels. Data from the flow rate sensor should be sampled and processed in real time, with the ability to configure specific sampling intervals.

3.5.1.3 Acoustic Sensor Interface: The acoustic sensor is to be connected to the system using a digital communication protocol (such as I2C, SPI, or UART) to gather sound data. This information is crucial for identifying turbulence levels, with sensitivity settings adjustable within the software to accommodate different underwater environments.

3.5.2 Communication Interfaces

The system is required to enable data communication with external devices through USART or an acoustic modem, thereby allowing for remote monitoring and control capabilities.

3.5.2.1 USART Interface: The USART interface should be set to a baud rate of 9600 bps to maintain compatibility with established communication protocols. Data packets sent via USART must consist of a header, a data payload, and a checksum to facilitate error detection.

3.5.2.2 Acoustic Modem Interface: To accommodate deeper or remote underwater operations, the system must provide support for data transmission through an acoustic modem. This interface should adhere to a protocol designed for underwater conditions, incorporating data compression techniques to enhance transmission reliability in acoustically challenging environments.

3.5.2.3 Data Packet Structure: Each data packet transmitted, regardless of whether it is through USART or an acoustic modem, must adhere to a defined structure that includes sensor ID, timestamp, data type, measurement values, and checksum. This standardized format guarantees compatibility with monitoring interfaces and aids in error detection upon receipt.

3.5.2. User Interfaces

The system is required to offer a user interface (UI) that facilitates monitoring, configuration, and diagnostics, which can be accessed via a connected host device or computer.

3.5.2.1 Monitoring Dashboard: The UI should have an option to present real-time information from each sensor, showing current levels of sediment and turbulence in a structured and user-friendly manner. It can incorporate data visualization tools, such as graphs illustrating recent measurements, to assist users in understanding data trends this depends on how much time we have for the project else it would just save a .json or .csv/.txt file and then the user would have to plug in another tool to perform this analysis.

3.5.2.2 Configuration Interface: The UI must enable users to modify settings, allowing for adjustments to sensor sampling intervals, threshold and constraints/parameter values for sediment and turbulence notifications. Access control measures should be implemented to safeguard these settings against unauthorized modifications.

3.5.2.3 Diagnostics and Error Reporting (not important): The UI should feature a diagnostics section that enables users to monitor sensor statuses, system uptime, and memory utilization. Real-time error messages and alerts must be provided to notify users of any operational problems, along with clear instructions for corrective measures.

3.5.3. Power Interface

The system is designed to utilize a power source that is appropriate for underwater use, guaranteeing continuous functionality for prolonged durations, the interface used can also be just a power bank with at least 6000mAH in energy capacity and must supply at least 5V of power at around 100mA for optimum sensor and microcontroller operation as stated in section **3.3.1 about Hardware Compatibility.**

3.5.3.1 Battery Oversight: The system is required to track battery status and provide notifications for low battery levels to enable prompt replacement or recharging which is usually included in the battery module and can be just forked from there.

Supporting Information

Appendices

A. Glossary

B. List of Symbols

Index

- **IEEE Guide to Software Requirements Specifications** – Referenced in Table of Contents, Section 1.4 (References)
- **MS5837-30BA Pressure Sensor Datasheet**: TE Connectivity. (n.d.). *MS5837-30BA Miniature Pressure Sensor*. Retrieved from TE Connectivity Datasheet.
- **SEN-HZ21WA Flow Rate Sensor Datasheet**: e-Gizmo. (n.d.). *SEN-HZ21WA Flow Meter Specifications*. Retrieved from e-Gizmo Datasheet.
- **HTI-96-MIN Hydrophone Datasheet**: High Tech, Inc. (n.d.). *HTI-96-MIN Hydrophone Specifications*. Retrieved from High Tech Inc. Datasheet.
- **DS18B20 Temperature Sensor Datasheet**: Maxim Integrated. (n.d.). *DS18B20 Programmable Resolution 1-Wire Digital Thermometer*. Retrieved from Digi-Key Datasheet.
- **STM32F411 Nucleo-64 User Manual**: STMicroelectronics. (n.d.). *UM1724: STM32 Nucleo-64 Boards (MB1136) User Manual*. Retrieved from STMicroelectronics Manual.