Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

Harald Brøyn Jordalen

Control and Position Monitoring of Buoyancy Vehicle

Master's thesis in Industrial Cybernetics
Supervisor: Jo Arve Alfredsen
June 2023

NTNU
Norwegian University of
Science and Technology

Harald Brøyn Jordalen

Control and Position Monitoring of Buoyancy Vehicle

Master’s thesis in Industrial Cybernetics
Supervisor: Jo Arve Alfredsen
June 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Engineering Cybernetics

NTNU
Norwegian University of
Science and Technology

Abstract

A small-scale prototype buoyancy vehicle (BV) for underwater actuation has been made. The BV is capable of changing its own volume using a piston-based displacement mechanism, which allows it to adjust its density and buoyancy force, and change its vertical movement in the water column. To stabilize the BV at specific depths, a continuous feedback PID controller has been implemented, utilizing the piston as an actuator and a pressure sensor for depth measurements. By letting the BV drift horizontally with the water flow, as a "package of water" and monitoring the position, the BV could be a valuable tool for measuring sub surface ocean current data as a Lagrangian drifter.

The goals of this project are the following:

- Complete the BV hardware and software design.
- Fine-tuning of BV control system.
- Analyse BV operational range wrt. theoretical sea water density.
- Prepare underwater acoustic tracking system.
- Test and verify BV and tracking system in sea trials, to characterize physical performance and limitations.

A mathematical model simulating the BV's dynamics has been developed in Matlab/Simulink for analyzing the BV performance and limitations in various scenarios. The simulation is also able to calculate the presetting value of the outer lid of the vehicle, for presetting the maximum/minimum density of the vehicle before starting a new dive. Presetting of the outer lid is dependent on the density profile at the target location. The vehicle has been analyzed in terms of theoretical seawater density (temperature, salinity, and pressure), where it was found that the vehicle is able to operate in most scenarios. However, with the presence of lower-density fresh water, the density profile of the target location could be wary too much, reducing the vehicle's capability for diving.

The tracking system utilizes the time difference of arrival (TDoA) method for real-time position monitoring of the BV position underwater. The tracking system is built upon hydrophone and acoustic transmitter from Thelma Biotel, and a SLIM module for time synchronization and positional data. Timestamps and other data are sent to a host computer through an ARF868 radio modem, where the data is processed and the positions (x,y,z) are plotted in real time.

Both BV and tracking system have been validated through physical experiments. The control system of the vehicle has been successfully tuned, enabling the vehicle to autonomously dive to a predefined depth, remain neutrally buoyant, and return to the water surface when the dive is finished. The implementation of IMU reveals stable behavior of the vehicle in water during a dive. The tracking system has been tested by placing an acoustic receiver at a known location and analyzing the calculated position. Both the BV and tracking system reveal promising results, enabling the BV to be used as a Lagrangian drifter, and a valuable tool for measuring subsurface ocean currents at various depths.

Sammendrag

En små skale prototype av en oppdriftsenhet, engelsk kalt buoyancy vehicle og heretter forkortet til BV. For undervannsaktivering har blitt produsert. BV er i stand til å endre sitt eget volum ved hjelp av en stempelbasert forskyvningsmekanisme, som lar den justere tetthet og oppdriftskraft, og endre sin vertikale bevegelse i vannsøylen. For å stabilisere enheten på en spesifikk dybde, er en PID-kontroller implementert, som bruker stempelet som en aktuator og en trykksensor for dybdemålinger. Ved å la BV drive horisontalt med vannstrømmen og overvåke posisjonen, kan BV være et nyttig verktøy som en Lagrange-drifter for å måle data for havstrømmer under overflaten.

Målene i denne rapporten er følgende:

- Fullføre BV maskinvare- og programvaredesign.
- Finjustere BV kontrollsystem.
- Analysere BV operasjonelle rekkevidde ift. teoretisk sjøvannstetthet.
- Forberede et undervanns akustisk sporingssystem.
- Teste og verifisere BV og sporingssystemet i sjøprøver for å karakterisere fysisk ytelse og begrensninger.

En matematisk modell er utviklet i Matlab/Simulink for å analysere BVs ytelse og begrensninger i forskjellige scenarier. Simuleringen kan også beregne forinnstillingen av enhetens ytre lokk, for å forinnstille maksimum/minimum tetthet før et nyt dykk starter. Forinnstillingen av det ytre lokket avhenger av tetthetsprofilen til vannet på stedet. Enheten er analysert i form av teoretisk sjøvannstetthet (temperatur, saltholdighet og trykk), hvor det ble funnet at enheten kan operere i de fleste scenarier. Imidlertid kan tilstedeværelsen av ferskvann med lavere tetthet variere tetthetsprofilen for mye, noe som reduserer hvor langt enheten klarer å dykke.

Sporingssystemet utnytter tidsforskjellen ved ankomst (TDoA) metoden for å overvåke BV's posisjon under vann i sanntid. Sporingssystemet er bygd på en hydrofon og akustisk sender fra Thelma Biotel, og en SLIM-modul for tidssynkronisering og posisjonsdata. Tidsstemplene og annen data sendes til en datamaskin gjennom en ARF868 radio modem, hvor dataen behandles og posisjonen (x, y, z) plottes i sanntid.

Både BV og sporingssystemet er verifisert gjennom fysiske eksperimenter. Kontrollsystemet til enheten har blitt vellykket justert, noe som gjør at BV autonomt kan dykke til en forhåndsdefinert dybde, holde seg nøytral i vannet, og returnere til overflaten når dykket er ferdig. Sporingssystemet er verifisert ved å plassere in akustisk sender på en kjent plass og analysere den beregnede posisjonen i forrhold til den faktiske posisjonen. Både BV og sporingssystemet viser lovende resultater og muliggjør systemets evne til å opperere som en Lagrangian drifter, ett verdifult verktøy for måling av undervanns hav strømninger.

Acknowledgement

I would first like to thank my project supervisor, Jo Arve Alfredsen of the Department of Engineering Cybernetics at NTNU, for providing me with guidance both practical and theoretical. And for facilitating the necessary equipment and experimental testing.

I would like to thank Glenn Angell and Daniel Bogen at the mechanical workshop ”kybernetisk verksted” for their work and insight into mechanical modifications. Thank’s to Tim Cato Netland at the Department of Electronic Systems for letting me use the acoustic water tank for testing the BV.

I would like to thank Emilie for endless support during this project, and my academic journey at NTNU.

Author

Harald Brøyn Jordalen

Table of Contents

List of Figures

List of Tables

Abbreviations

BV	=	Buoyancy Vehicle
SLIM	=	Synchronization and LoRa Interface Module
CTD	=	Conductivity Temperature Depth
PCB	=	Printed circuit board
SoC	=	System on Chip
SWD	=	Serial Wire Debug
RTT	=	Real Time Transfer
FSM	=	Finite State Machine
EMA	=	Exponential moving average
TDoA	=	Time Difference of Arrival
TBR	=	Thelma BioTel hydrophone receiver (TBR 700RT)
ADC	=	Analog to Digital Converter
GPIO	=	General purpose Input-Output
BLE	=	Bluetooth Low Energy
TWIM	=	Two Wire Interface Master
UART	=	Universal Asynchronous Receiver-Transmitter
I2C	=	Inter-Integrated Circuit
ASIC	=	Application Specific Integrated Circuit
SPI	=	Serial Peripheral Interface
PWM	=	Pulse Width Modulation

Chapter 1

Introduction

1.1 Lagrangian drifter

In oceanography, Lagrangian drifters are a valuable tool for studying ocean currents, water circulation patterns, and the transport of pollutants, among other phenomena. These drifting instruments follow the motion of ocean currents, allowing researchers to track their movement and collect valuable data. Lagrangian drifters consist of a buoyant sphere, a drogue or sea anchor, and a tracking device [1]. The buoyant sphere is designed to float on the surface of the ocean, while the drogue is attached to the sphere and is designed to remain submerged at a fixed depth, usually several meters below the surface. This combination of buoyancy and drag allows the drifter to follow the motion of the water in a Lagrangian frame of reference, i.e., moving with the water rather than with the surrounding atmosphere.

Lagrangian drifters are deployed at various locations in the ocean, either by ships or from shore-based stations. They can also be deployed in large numbers to cover a wide area, providing information on the spatial distribution of ocean currents and other water properties. Data from Lagrangian drifters can be used to study ocean circulation patterns, track the movement of oil spills, and study the effects of climate change on the ocean, among other applications.

The purpose of this report is to assess a new design for a Lagrangian drifter that utilizes buoyancy force to remain neutral at fixed depths. The device, referred to as a buoyancy vehicle (BV), is capable of adjusting its own buoyancy and vertical movement within the water column. The idea is that the BV is able to remain neutrally buoyant, and behaves as a ”package of water”, enabling the vehicle to drift with the ocean current. By integrating a suitable tracking system, the subsurface ocean current could be measured by monitoring the BV position (x,y,z) over time. The main application of this technology is for studying currents at different depths in fish farms. This report will provide an evaluation of the BV design, including its capabilities and limitations, implementation of real-time tracking system, as well as its potential for improving oceanographic research around fish farms.

1.2 Project Description

This project builds upon previous work by Stian Børseth [2] and Odd Inge Halsos [3] by developing a prototype buoyancy vehicle (BV) for underwater actuation. The BV is capable of changing its own volume using a piston-based displacement mechanism, which allows it to adjust its density and buoyancy force and change its vertical movement in the water column. To stabilize the BV at specific depths, a continuous feedback PID controller has been implemented, utilizing the piston as an actuator and a pressure sensor for depth measurements.

In this report, we will provide a detailed analysis of the operational range of the BV in relation to theoretical seawater density, as well as a characterization of the control system. To function effectively as a Lagrangian drifter, a suitable tracking system must be implemented. Therefore, we will also discuss the design and implementation of a tracking system that is capable of tracking the BV's position in all three dimensions (x, y, z) in real-time.

1.3 Previous work

In this section, we provide a brief overview of previous work carried out by Stian Børseth [2] in 2017 and Odd Inge Halsos [3] in 2020. While the specific details of their work will be discussed later in the report, it is important to note that a considerable amount of effort has been devoted to realizing the BV design.

1.3.1 Buoyancy Vehicle

A prototype buoyancy vehicle (BV) has been designed by Børseth, as shown in Figure 1.1. The BV is equipped with a stepper motor that is capable of changing the volume of the vehicle by translating rotational movement into linear movement. To use the stepper motor as a linear actuator, a wedge between the holster and housing cap prevents the holster from rotating, allowing rotating the lead screw to provide linear movement to the piston.

To power the sensors, stepper motor, and microcontroller, a custom-made battery package has been created to fit 16 1.5V alkaline batteries, providing 24V power. The battery package is positioned at the bottom of the vehicle, which shifts the center of mass towards the bottom and increases the stability of the BV in water.

The vehicle house is threaded, and the outer lid is screwed on with a seal to prevent water from entering the housing. The outer lid can be mounted at varying distances from the bottom of the vehicle house and serves as a presetting for the maximum volume/buoyancy of the BV before submerging in water. It is also used to handle additional weight gain or loss. We will provide further details on the design and performance of the BV in subsequent sections of this report.

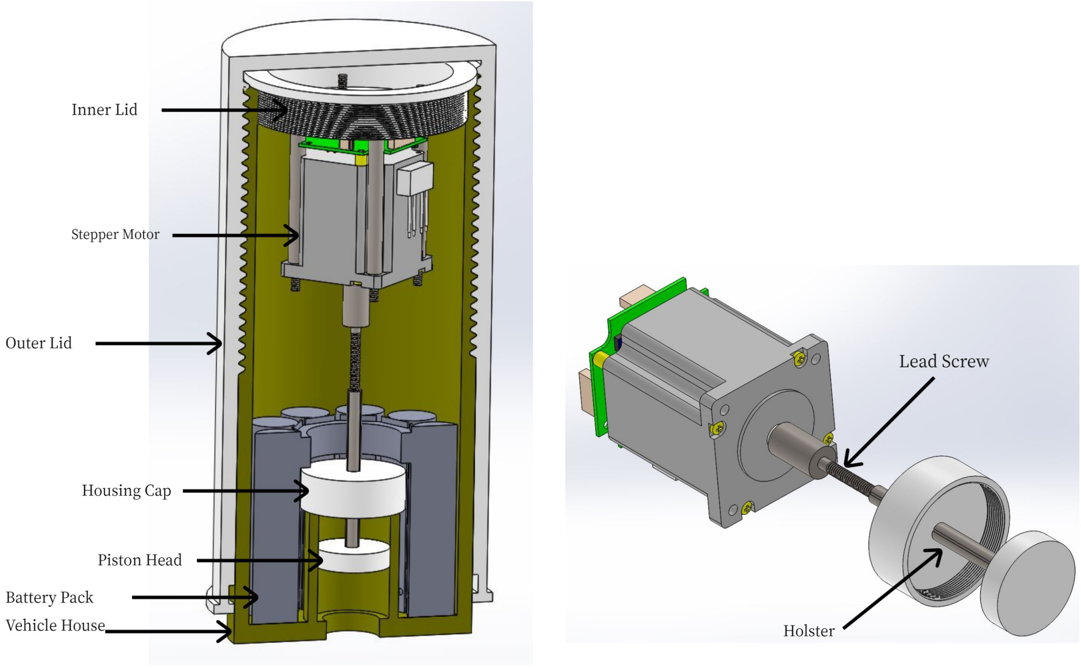


Figure 1.1: BV produced by Børseth [2]

1.3.2 Embedded system

Halsos developed an embedded control and sensor system, featuring a printed circuit board with sensor modules and a Bluetooth interface for operating the BV. The system is based on the Nordic Semiconductor nRF52832 system-on-chip as its main processing unit, as shown in Figure 1.2, which provides an overview of the major components and their connections to the main processing unit. Further details on each module can be found in Chapter 5, section 5.1.

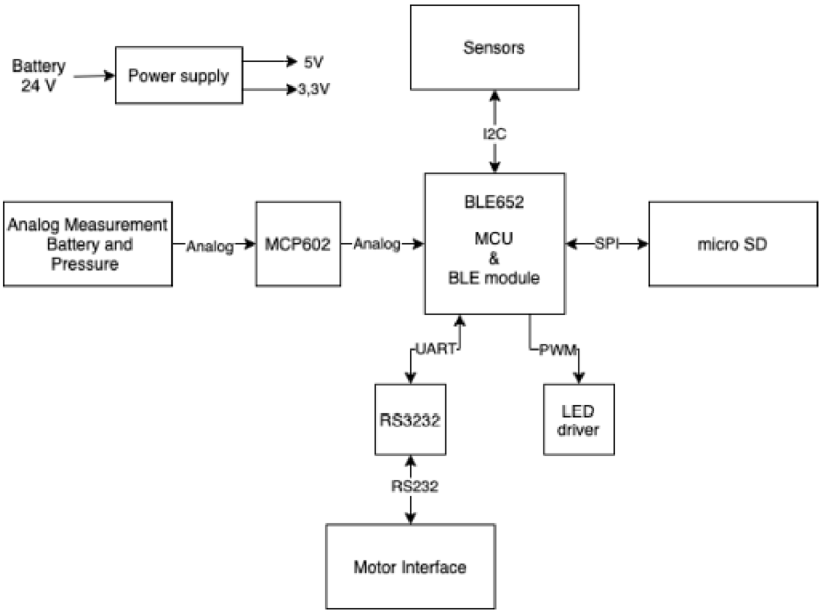


Figure 1.2: High-level system diagram to show modules and interface to the main processing unit, reproduced from [3]

The BV is equipped with sensors to measure pressure/depth, temperature, battery voltage, and inertial measurement unit (IMU), related data is stored on an SD-card for access after the mission is completed. While an IMU is implemented, no driver has been developed and no data is collected from the IMU at this time. Additionally, the BV features an RGB LED light, which is used to identify the vehicle in the water, and the color is used to indicate the current operating state. The stepper motor is equipped with limit switches to prevent damage to the piston and linear actuator.

The BV makes use of Bluetooth Low Energy (BLE) to communicate with a computer or phone, enabling the configuration of the vehicle, setting mission time and depth, and transferring or deleting mission log files. Halsos has developed a terminal application to interface with the vehicle from a computer, utilizing a menu structure as shown in 1.3 for configuring the vehicle and changing to different states. In addition, the BV can be interfaced with a phone by using the free application ”nRF Toolbox” from Nordic Semiconductor.

1.3.3 Specialization Project

In the autumn of 2022, a specialization project on the BV aimed to derive use cases and specifications for operating the vehicle and to use them as validation criteria. The BV was subjected to comprehensive hardware and software testing. A simulation of the BV dynamics was developed using MATLAB/Simulink to analyze the BV’s performance and limitations. The project also focused on understanding a relatively large code base in c, developed by Halsos.

The findings of the specialization project can be summarized as follows:

- Hardware testing revealed that the sensors on the I2C bus had to be disconnected due to

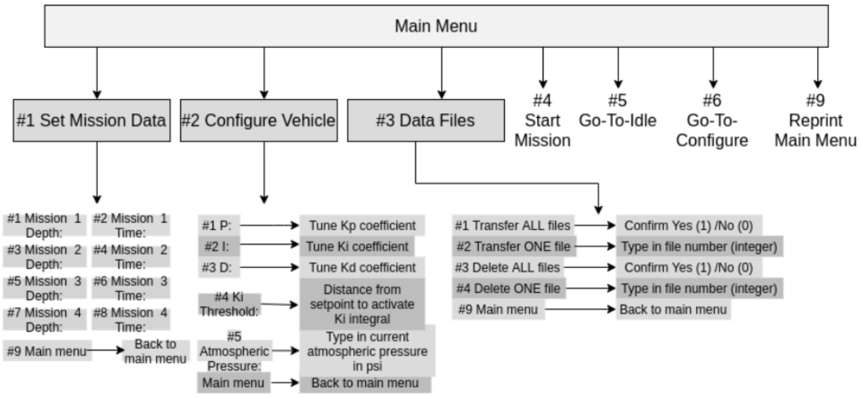


Figure 1.3: Diagram to illustrate the menu structure, which is interfaced through the Bluetooth application. Reproduced from [3]

noise on the bus line, which originated from a Hall-effect sensor. Chapter 5 describes the replacement of the Hall-effect sensor and the implementation of drivers for I2C sensors.

- During physical lab testing, the BV performance was evaluated in a freshwater tank with a depth of 2 meters. The PID controller was successfully tuned with acceptable performance. However, it was discovered that a lowpass filter with too small bandwidth was implemented, making the controller unreliable in different scenarios. This finding highlights the importance of careful design and selection of filters to ensure that they are appropriate for the system's intended use.
- A mathematical model was developed using MATLAB/Simulink to analyze the BV's performance and limitations. However, when comparing the model with physical lab tests, it was found that the model did not accurately represent the BV's physical performance as the simulation yielded different results than the lab tests. Subsequent investigation revealed that the MATLAB model did not accurately represent the PID controller. Chapter 4 provides details of the model and the new features added to improve its accuracy.

1.4 Scope and goal of project

The goal of this project is to bring the BV to a fully operational state, fulfilling the requirements of its intended use case as a Lagrangian drifter, and a proof of concept. To achieve this goal, the project has been divided into the following sub-tasks:

- Fix/Complete missing BV hardware and software components identified during the preceding Specialisation project.
- Detailed analysis, characterization, and fine-tuning of BV control system performance in lab tests.
- Mapping of BV's operational range wrt. theoretical sea water density (temperature, salinity, pressure) at given presets of the outer cap.
- Prepare underwater acoustic tracking system for tracking the BV in sea trials.
- Plan and carry out sea trials of the BV and tracking system to characterize its physical performance and limitations as a subsurface drifter

1.5 Structure of the thesis

Chapter 1 - Introduction

Present Lagrangian drifter concept, previous work, motivation, and project goal.

Chapter 2 - Theory

Present relevant background theory for the report.

Chapter 3 - Vehicle Density Control

Mapping of BV's operational range with right to theoretical seawater density.

Chapter 4 - Mathematical Simulation

Present Matlab/Simulink simulation of BV characteristics and performance.

Chapter 5 - Buoyancy vehicle

Present BV hardware and software implementation.

Chapter 6 - Tracking System

Present hardware and software implemented in the tracking system.

Chapter 7 - Results

Present results and setup of physical experiments and functional test plan.

Chapter 8 - Discussion and Conclusion

Discussion and conclusion of sub-goals and the functionality as a Lagrangian drifter.

Chapter 2

Theory

This chapter provides the theoretical background that this project is based upon and includes the following topics: An explanation and derivation of the buoyancy effect, an introduction to PID controller, an overview of digital filters, and the derivation of equations of motion for the BV. In addition, we discuss the density profile of seawater, including the application of the UNESCO formula for determining seawater density. Lastly, we will explore the Time Difference of Arrival method, for calculating the location of a transmitter.

2.1 Buoyancy Effect

Buoyancy is a phenomenon that describes the upward force experienced by an object when it is immersed in a fluid. This force is equal to the weight of the fluid displaced by the object, according to Archimedes' principle. This principle states that any object that is partially or completely submerged in a fluid experiences an upward force that is equal to the weight of the fluid displaced by the object. [4]

The buoyancy force is directly related to the density of the fluid in which the object is immersed. For example, an object will experience a greater buoyancy force when it is immersed in water (which has a density of approximately 1000 $\frac{kg}{m^3}$) than when it is immersed in air (which has a density of approximately 1.2 $\frac{kg}{m^3}$). The buoyancy force is also affected by the size, shape, and mass of the object. A larger object will displace more fluid and experience a greater buoyancy force than a smaller object, all other factors being equal.

The buoyancy force can be derived by analyzing the forces acting on an object of cylindrical shape that is immersed in a fluid, as seen in figure 2.1. The pressure on the object increases with depth, following the equation $p = \rho_f g h$, where p is the pressure, ρ_f is the density of the fluid, g is the gravitational force, and h is the depth of the object. The force on the object is proportional to the pressure and the surface area, expressed as $F = pA$. By analyzing the free body diagram, one can derive the buoyancy force (F_B) as

$$F_B = F_2 - F_1 = p_2 A - p_1 A = (h_2 - h_1) A g \rho_f \tag{2.1}$$

This formula can be generalized to objects of any shape as

$$F_B = V_o \rho_f g \tag{2.2}$$

Where V_o is the volume of the object, ρ_f is the density of the fluid, and g is the gravitational constant.

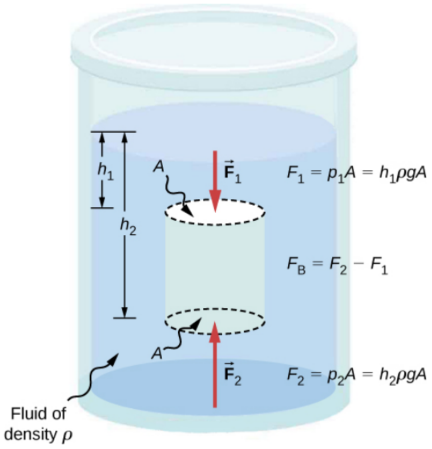


Figure 2.1: Free body diagram describing buoyancy force, reproduced from [4]

2.1.1 Equilibrium

When an object is submerged in a fluid, it experiences two vertically directed forces: buoyancy and gravity. The gravitational force acting on the object is proportional to its mass and can be expressed as $F_G = mg$. Newton's First Law can be used to derive the equilibrium of the object, by setting the sum of forces equal to zero.

$$\sum F = 0 \to F_G = F_B \to mg = V_o \rho_f g \to \frac{m}{V_0} = \rho_f \to \rho_o = \rho_f \tag{2.3}$$

When density of the object is equal to that of the fluid, the object is in a state of equilibrium, and its position remains constant. If the density of the object is less than that of the fluid, the resulting force is directed upwards, causing the object to float to the surface. Conversely, if the density of the object is greater than that of the fluid, the resulting force is downwards, causing the object to sink.

2.2 PID Controller

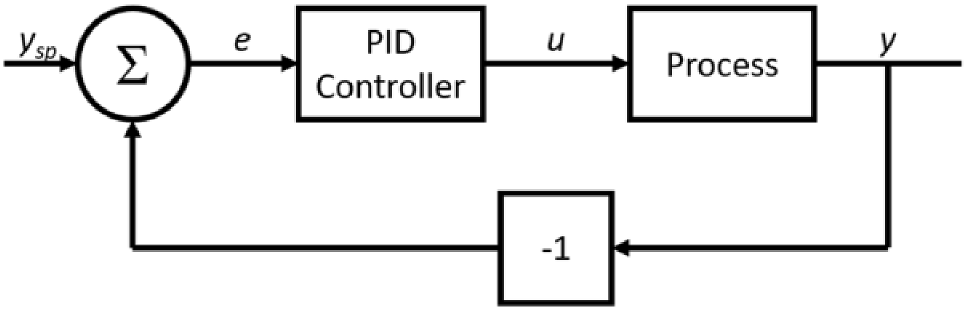


Figure 2.2: Block diagram of a feedback PID controller, reproduces from [5]

PID control is a widely-used method for controlling various processes [5]. The term PID stands for Proportional, Integral, and Derivative, which represent the three coefficients that are tuned to optimize the system response. In a feedback PID controller, the measured state is continuously fed back to the controller, and the error between a setpoint and the measured state is calculated and fed forward to the PID controller. The controller then feeds forward the output to a process. The block diagram of an ordinary feedback PID controller is shown in Figure 2.2.

In continuous form, a PID controller can be described by Equation 2.4. Here, $e(t)$ is the error between a reference point $r(t)$ and the measured state $y(t)$, $u(t)$ is the controller output, to the actuator or process, and K_p, K_i, and K_d are the weighted values for tuning the proportional, integral, and derivative terms.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \tag{2.4}$$
$$e(t) = r(t) - y(t)$$

When digital controllers are used for PID control, the incoming data is discrete and sampled at regular intervals, rather than continuously. The discrete form of the PID controller is given by Equation 2.5, where T is the sampling interval. The PID controller continuously updates the control input based on the current error, the previous errors and the rate of change in error, allowing it to quickly respond to changes in the system.

$$u[k] = K_p e[k] + K_i T \sum_{j=0}^{k} e[k] + \frac{K_d}{T}(e[k] - e[k-1]) \tag{2.5}$$
$$e[k] = r[k] - y[k]$$

The PID controller is based on three basic control actions: proportional, integral, and derivative. Each of these actions contributes to the overall control output in a unique way, resulting in a dynamic and responsive control system.

- **Proportional response:** Proportional term is based on the idea that the control output should be proportional to the error signal, which is the difference between the setpoint and the process variable. The proportional gain (K_p) determines the strength of the proportional action and adjusts the control output in proportion to the error signal. A P-controller often results in a steady state error, meaning that the process variable has stabilized with an error from the setpoint and is no longer changing.

- **Integral response:** The integral term accumulates the error over time and produces a control output that is proportional to the sum of the error signals. This is particularly useful in systems with steady-state errors that cannot be eliminated by proportional control alone. However, it is important to be cautious with the integral term. For large values of the integral gain (K_i) or slow systems, the integral summation could saturate the system, causing integral wind-up. Integral wind-up occurs when the integral summation becomes so large that it exceeds the physical limits of the system, leading to an unstable response. To prevent wind-up, the integral summation should be limited to reasonable values, preventing the integral term from contributing too much to the control output. This is commonly known as anti-wind-up and is an important technique used in PID controller design to ensure stability and accurate control.

- **Derivative response:** The derivative term is proportional to the rate of change to the error and has a damping effect by decreasing the output value in case of rapid changes in the system. The derivative term is particularly useful in systems where P and PI control alone result in oscillatory behavior. In discrete systems where the setpoint changes abruptly to a new state, the derivative term is exposed to derivative kick as the error in one time sample is much larger than the next time sample, remember that $e(t) = r(t) - y(t) \rightarrow \frac{de(t)}{dt} = \frac{dr(t)}{dt} - \frac{dy(t)}{dt}$ by assuming that the reference point is constant, the resulting derivative is zero. The derivative term can then be written as $-K_d * \frac{dy(t)}{dt}$. It's worth noting that the derivative term can be sensitive to noise in the measurement signal, so it's important to use a filter to reduce noise when applying the derivative term. Additionally, the derivative term can cause instability in the system if the gain is set too high, so it's important to tune the PID controller carefully to avoid this issue.

2.3 Mathematical Modelling

In this section, we will be discussing the process of modeling the BV using equations of motion. The first step in this process involves deriving differential equations that describe the system by analyzing the active forces acting upon the vehicle. Additionally, we will provide a model for density calculation of the vehicle.

2.3.1 Equation of motion

The BV is able to move in one dimension, up and down in the water column. Let's first analyze the affected forces on the vehicle:

Gravity:	$F_G = mg$
Buoyancy:	$F_B(t) = V_o(t)\rho_f g$
Drag:	$F_D(t) = \frac{1}{2}C_D\rho_f v(t)^2 A$
Disturbances:	$F_E(t)$

With respect to the equations above, we can derive the equation of motion by using Newton's 2nd law, where $a(t) = \ddot{x}(t)$, $v(t) = \dot{x}(t)$ and $x(t) = \textit{position}$. Note that the positive direction is downward, as using the water surface as a zero point is beneficial.

$$\begin{aligned} \sum F =& F_G - F_B(t) - F_D(t) + F_E(t) = ma(t) \\ \rightarrow a(t) =& \frac{1}{m}(F_G - F_B(t) - F_D(t) + F_E(t)) \\ \rightarrow a(t) =& \frac{1}{m}(mg - V_o(t)\rho_f g - \frac{1}{2}C_D\rho_f v(t)^2 A + F_E(t)) \end{aligned} \tag{2.6}$$

Hence, the equation of motion reads as follows:

$$\ddot{x}(t) + \frac{1}{m}\{\frac{1}{2}C_D\rho_f\dot{x}(t)^2 A - mg + V_o(t)\rho_f g - F_E(t)\} = 0 \tag{2.7}$$

Upon observation, it has been determined that the equation is nonlinear. This is due to the presence of the velocity squared term, and the constant term which contain the actuator $V_o(t)$. As a result, the equation of motion becomes quite challenging to analyze in either state space form or in the Laplace domain.

2.3.2 Density model

The vehicle manipulates its own volume by moving a smaller piston, letting water entering the cylinder, and effectively changing its own density. In general, the density of the vehicle is calculated as follows:

$$\rho = \frac{m}{V_{max} - x_{piston}A_{piston}}$$

The piston's initial position is always at $x_{piston} = 0$, as depicted in figure 2.3. Consequently, this results in a lower density, causing the vehicle to float toward the surface. Conversely, increasing x_{piston} will yield a higher density, causing the vehicle to sink.

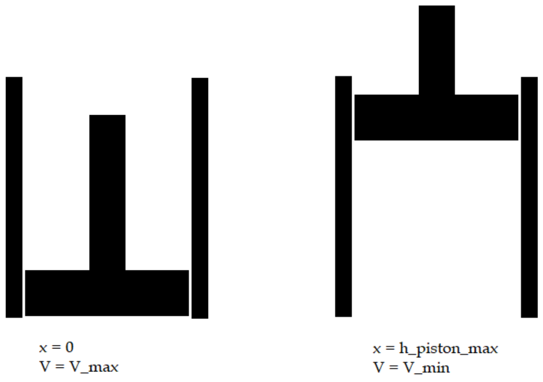


Figure 2.3: Piston position is defined as x_{piston}.

2.4 Digital Filter

Noise is a common presence in most sensors, and in the context of digital controllers, it can have a notable impact on the derivative term of a PID controller. This influence makes it challenging to fine-tune the controller and can compromise the system's robustness. To mitigate these issues and enhance signal quality, the implementation of digital filters is beneficial. These filters effectively reduce noise and contribute to the overall smoothing of signals, resulting in improved system performance.

2.4.1 EMA Filter

The Exponential Moving Average (EMA) filter [6] is a discrete low-pass filter that utilizes the previous smoothed value $y[n-1]$ and the current measurement $x[n]$ to calculate a new smoothed value. The equation for the EMA filter is represented as:

$$y[n] = \alpha x[n] + (1-\alpha)y[n-1] \tag{2.8}$$

In this equation, the weighted value α must be within the range of $\alpha \in [0,1]$. The value of α determines the extent of smoothing applied by the filter. A low value of α, such as $\alpha = 0.1$, results in high smoothing, but the filter responds slower to changes in measurements. Conversely, a high value of α provides lower smoothing but enables quicker response to measurement changes. It is worth noting that if $\alpha = 1$, the filter output will be identical to the sensor measurement without any filtering.

2.5 Density profile in seawater

Density refers the the amount of mass per unit volume, such as $\frac{Kg}{m^3}$. The density of seawater is primarily determined by its temperature, salinity, and pressure. [7]

- **Temperature:** As temperature increases, the density of seawater decreases, and as temperature decreases, the density increases. This is because the water molecules expand when they are heated and contract when they are cooled.
- **Salinity:** Salinity is the amount of dissolved salts and other minerals in seawater. As the salinity of seawater increases, the density also increases. This is because the dissolved salts add mass to the water without increasing its volume.

- **Pressure:** Pressure can also affect the density of seawater, especially at greater depths. At high pressures, seawater can become more compressed, which can increase its density.

These factors interact with each other in complex ways, making seawater density a function of multiple variables. The standard reference for seawater density is the International UNESCO formula, which provides a mathematical formula that takes into account temperature, salinity, and pressure to calculate the density of seawater. Seawater has an average density of $1027\frac{Kg}{m^3}$, but this varies over a range of about $1020\frac{Kg}{m^3}$ to $1029\frac{Kg}{m^3}$. Figure 2.4 illustrates how density varies with common values for temperature and salinity in seawater.

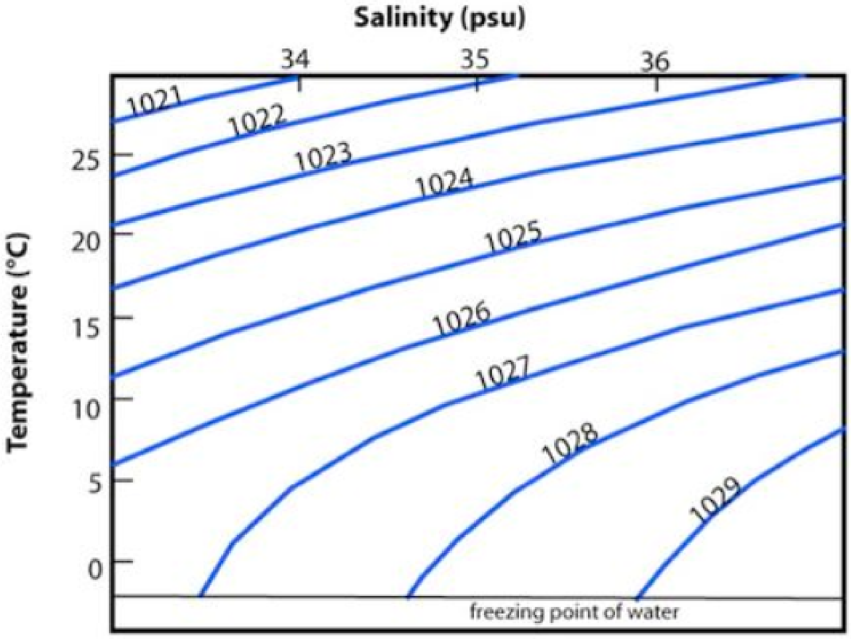


Figure 2.4: Lines of equal density $[\frac{Kg}{m^3}]$ given salinity and temperature, reproduced from [8]

2.5.1 UNESCO formula

Sea water density is calculated using the UNESCO formula [9], which combines theoretical and empirical models to estimate density over a wide range of conditions, from water surface to deep ocean. Density is calculated with temperature in the range [0 - 40]°C, salinity in the range [0 - 42] PSU (practical salinity unit), and pressure in BAR. [9]

Below is a brief description of how to use the UNESCO formula and the complete algorithm is provided in Appendix A.

- Calculate the SMOW density(Standard Mean Ocean Water)

$$\rho_{SMOW} = a_0 + a_1T + a2T^2 + a_3T^3\ a_4T^4\ a_5T^5$$

- Calculation of seawater density at the normal atmospheric pressure ($p = 0$)

$$\rho(S, T, 0) = \rho_{SMOW} + B_1 + C_1S^{1.5} + d_0S^2$$

  Where B_1 and C_1 are polynomial equations of temperature.

- Determination of the compressibility module at pressure $p = 0$

$$K(S, T, 0) = K_w + F_1S + G_1S^{1.5}$$

Where K_w, F_1, and G_1 are polynomial equations of temperature.

- Determination of final compressibility module of sea water.

$$K(S,T,p) = K(S,T,0) + A_1 p + B_2 p^2$$

Where A_1 and B_2 are polynomial equations of temperature and salinity.

- Final determination of seawater density (ρ) is calculated.

$$\rho(S,T,p) = \frac{\rho(S,T,0)}{1 - \frac{p}{K(S,T,p)}}$$

2.6 Time Difference Of Arrival

This section presents a theoretical overview of the Time Difference of Arrival (TDoA) positioning method, as described by Fang [10]. The TDoA method involves three fixed acoustic receivers positioned at stations A, B, and C. Station A serves as the reference point (0,0,0), while the second station (B) is located along the x-axis (b_x,0,0). The third station (C) is located in the xy-plane (c_x, c_y, 0), with both x and y coordinates being non-zero.

Let us consider an acoustic transmitter located at coordinates (x,y,z) with respect to point A, B, and C. The arrival times of the acoustic signal at stations A, B, and C are represented by T_a, T_b, and T_c, respectively. Furthermore, we denote the time difference of arrival between points A and B as T_{ba} and the time difference between A and C as T_{ca}. These relationships can be expressed by the following equations:

$$\begin{aligned} T_{ba} &= T_a - T_b \\ T_{ca} &= T_a - T_c \end{aligned} \tag{2.9}$$

Assuming that the signal travels at a velocity denoted by c, we can represent the distances between the transmitter and the fixed points A, B, and C as R_a, R_b, and R_c, respectively. To relate the time difference equations to the range difference equations, we can express them in terms of the distances between the fixed points A and B (R_{ab}) and A and C (R_{ac}):

$$\begin{aligned} R_{ba} &= T_{ba} * c \\ R_{ca} &= T_{ca} * c \end{aligned} \tag{2.10}$$

Using the geometry of the acoustic receivers setup, the distance of the acoustic transmitter placed at coordinates (x,y,z) from the three receivers is given by:

$$\begin{aligned} R_a &= \sqrt{x^2 + y^2 + z^2} \\ R_b &= \sqrt{(x - b_x)^2 + y^2 + z^2} \\ R_c &= \sqrt{(x - x_c)^2 (y - y_c)^2 + z^2} \end{aligned} \tag{2.11}$$

These distances can be written in terms of difference of range with respect to the receivers in location A as:

$$\begin{aligned} R_{ba} &= \sqrt{x^2 + y^2 + z^2} - \sqrt{(x - b_x)^2 + y^2 + z^2} \\ R_{ca} &= \sqrt{x^2 + y^2 + z^2} - \sqrt{(x - x_c)^2 (y - y_c)^2 + z^2} \end{aligned} \tag{2.12}$$

Squaring and simplifying Eq 2.12 yields:

$$\begin{aligned} R_{ba}^2 - b_x^2 + 2b_x * x &= 2R_{ba}\sqrt{x^2 + y^2 + z^2} \\ R_{ca}^2 - c^2 + 2c_x * x + 2c_y * y &= 2R_{ca}\sqrt{x^2 + y^2 + z^2} \end{aligned} \tag{2.13}$$

Where b_x and $c = \sqrt{c_x^2 + c_y^2}$ are the length between station A-B and A-C. When R_{ba} and R_{ca} are non-zero, Eq 2.13 can be combined and simplifying, one obtains:

$$y = g * x + h \tag{2.14}$$

Where:

$$g = \frac{R_{ca} * (\frac{b_x}{R_{ba}} - c_x)}{c_y} \tag{2.15}$$

$$h = \frac{c^2 - R_{ca} + R_{ca} * R_{ba}(1 - {\frac{b_x}{R_{ba}^2}}^2)}{2c_y} \tag{2.16}$$

Substituting 2.14 into 2.13, one obtains

$$z^2 = d * x^2 + e * x + f \tag{2.17}$$

Where:

$$\begin{aligned} d &= -\{1 - {\frac{b_x}{R_{ba}}}^2 + g^2\} \\ e &= b_x * \{1 - {\frac{b_x}{R_{ba}}}^2\} - 2g * h \\ f &= \frac{R_{ba}^2}{4} * \{1 - {\frac{b_x}{R_{ba}}}^2\}^2 - h^2 \end{aligned} \tag{2.18}$$

If we assume that the z-coordinate of the transmitter is known, we can proceed with solving the quadratic equation 2.17. This equation will provide us with two solutions for the x-coordinate. These solutions can then be inserted into equation 2.14 to obtain two solutions for the y-coordinate. Finally, these two sets of xy coordinates can be inserted into equation 2.12 to validate the correct solution for the x and y coordinates of the transmitter.

2.6.1 TDoA precision

Figure 2.5 illustrates the position uncertainty of the transmitter, using the TDoA method in two dimensions. Generally, higher precision is achieved within the triangular setup of receivers. As one moves outside of the triangle, higher levels of uncertainty are expected. The area behind the receivers typically exhibits the highest levels of uncertainty.

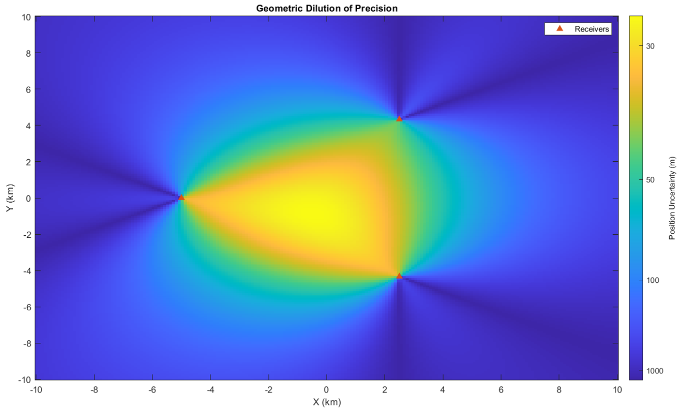


Figure 2.5: Precision of TDoA position in two dimensions, reproduced from [11]

Chapter 3

Vehicle Density Control

This chapter is about the controllable area of vehicle density, both in terms of presetting of the outer lid before diving, and the density range in which the vehicle is able to operate by manipulating the piston position and letting water entering the cylinder. We will discuss and analyze the operational range of the BV compared to the density of seawater.

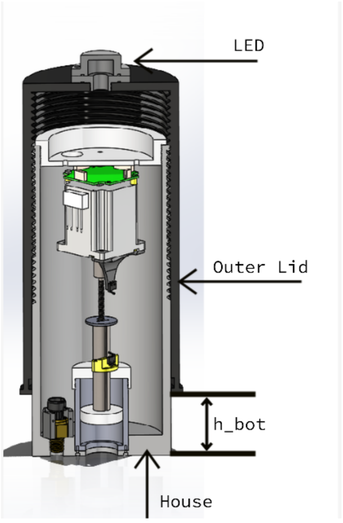


Figure 3.1: BV with units for calculating vehicle volume

3.1 Presetting of outer lid

The maximum volume of the vehicle is calculated in equation 3.1 by dividing the vehicle in units, where the sum of the units is the total volume of the vehicle. The units are illustrated in figure 3.1, where h_{bot} is the presetting variable for the maximum volume ranging from [10, 135]mm. V_{led} is the volume of led light, $V_{outerLid}$ is the volume of outer lid (including cap with bigger diameter), and V_{house} is the volume of the vehicle house from bottom to outer lid.

$$\begin{aligned} V_{\text{max}} &= V_{\text{led}} + V_{\text{outer lid}} + V_{\text{house}} \\ V_{\text{outer lid}} &= A_{\text{outer lid}} \cdot h_{\text{outer lid}} + A_{\text{cap}} \cdot h_{\text{cap}} \\ V_{\text{house}} &= A_{\text{house}} \cdot h_{\text{bot}} \end{aligned} \tag{3.1}$$

The vehicle's density is calculated using Equation 3.2, where mass is the total mass of the vehicle measured to 6.268 kg, and $V_{piston} = x_{h-piston} \cdot A_{piston}$ is the volume of the piston filled with water. $x_{h-piston}$ is the height of witch the piston are able to move, ranging from 0 to 51 mm.

$$\begin{aligned} \rho_{vehicle-min} &= \frac{mass}{V_{max}} \\ \rho_{vehicle-max} &= \frac{mass}{V_{max} - V_{piston}} \end{aligned} \tag{3.2}$$

As shown in Figure 3.2, presetting the outer lid enables the vehicle to operate in fluids with densities ranging from approximately [925 - 1220] $\frac{kg}{m^3}$, surpassing the typical density of seawater, which is approximately 1025 $[\frac{kg}{m^3}]$. This capability allows the vehicle to handle additional weight gain or loss. For seawater with density of $1020[\frac{kg}{m^3}]$, the vehicle must be preset to approximatly h_{bot} =0.095[mm], which result in a density range of about $\Delta\rho = \rho_{vehicle-max} - \rho_{vehicle-min} = 10[\frac{kg}{m^3}]$

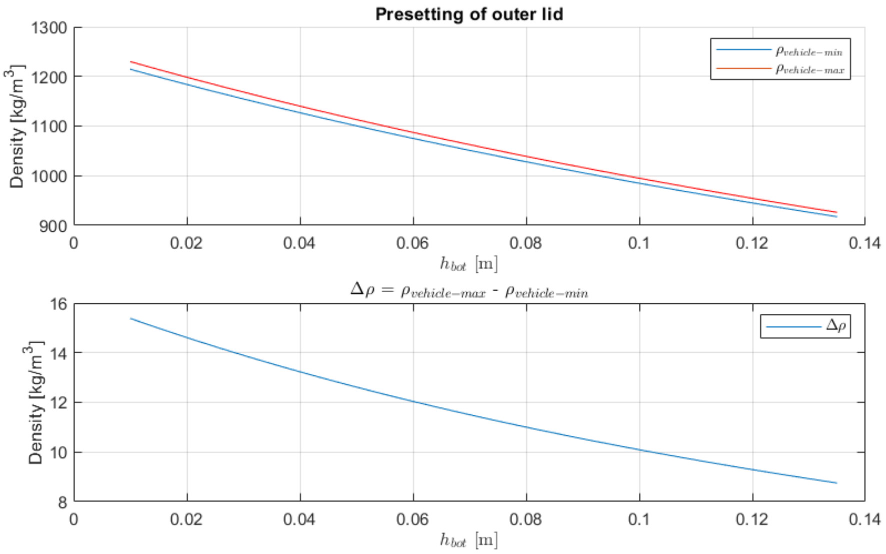


Figure 3.2: **Top:** Vehicle density presetting, by adjusting outer lid in range [10 - 135]mm from vehicle bottom. **Bottom:** Density range of vehicle, from piston top position to piston bottom position

3.2 Seawater Density

As described in Chapter 2 section 2.5.1, the density of seawater varies with temperature, salinity and pressure. In this section, we will discuss the effect of seawater density in terms of the UNESCO formula. To identify the operational range of the vehicle.

3.2.1 Pressure and seawater density

Let's begin by analyzing the pressure contribution in the UNESCO formula. As shown in Figure 3.3, the density of seawater changes with pressure ranging from 0 to 50 bar, equivalent to a depth of approximately 500 meters. Since the design criteria of the vehicle are to operate at a depth of 50 meters, which is roughly 5 bar, it can be observed from the figure that the density changes minimally within that range, given some extreme and typical values of salinity and temperature. Therefore, it is justifiable to use salinity and pressure as the primary factors for analyzing seawater density.

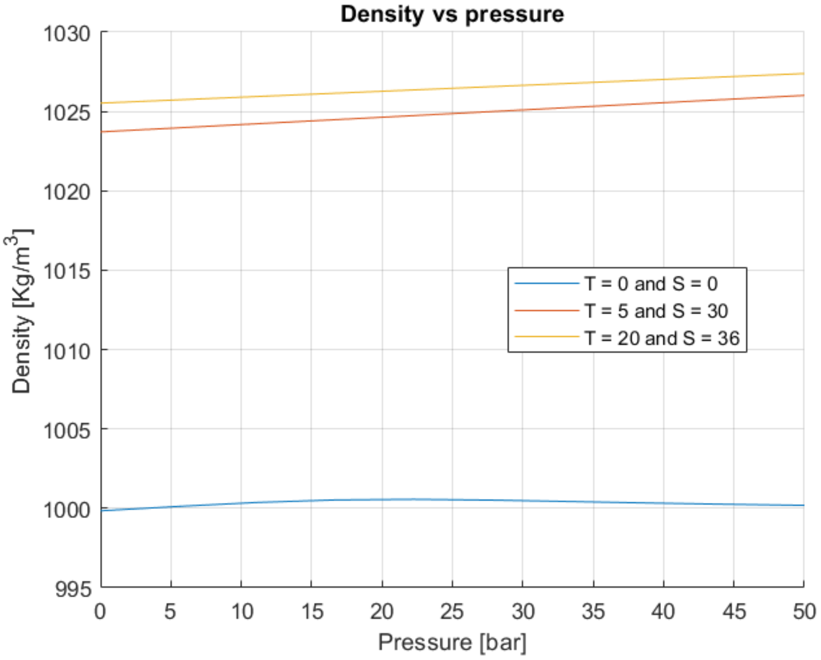


Figure 3.3: Density calculation by UNESCO formula vs pressure, given some extreme points of temperature and salinity

3.2.2 Density vs temperature and salinity

Figure 3.4 illustrates how seawater density varies with temperature and salinity. Lowest density of 997$\frac{kg}{m^3}$ is observed with zero salinity and a temperature of 25 degrees. With increasing salinity, density is rapidly increasing. With salinity of 36 PSU and temperature of 25 degree, result in a density of 1024$\frac{kg}{m^3}$. By further decreasing the temperature to 0 degrees, density is at the highest 1029$\frac{kg}{m^3}$. By changing piston position, the vehicel are able to change its own density approximately 10$\frac{kg}{m^3}$, its therefor clear that precisely presetting of the outer lid is important in terms of the density profile at the target location.

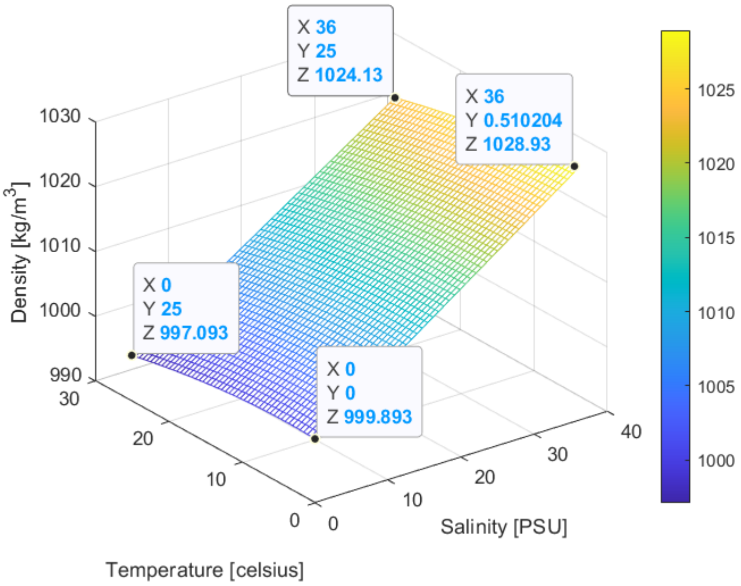


Figure 3.4: Density vs temperature and salinity, assuming constant pressure of 1 bar

3.2.3 Operational range of vehicle

Figure 3.5 illustrates the vehicle's operational range in relation to variations in temperature and salinity. The outer lid is set at a fixed height of $h_{bot} = 90mm$. At temperatures as low as 0 degrees Celsius, the vehicle is capable of functioning within a salinity range of 13.2 PSU to 26.4 PSU. In contrast, when the temperature is raised to 25 degrees Celsius, the vehicle can operate within a salinity range of 17.6 PSU to 31.6 PSU. This illustrates the importance of correctly adjusting the BV's outer lid in terms of salinity and temperature at the target location.

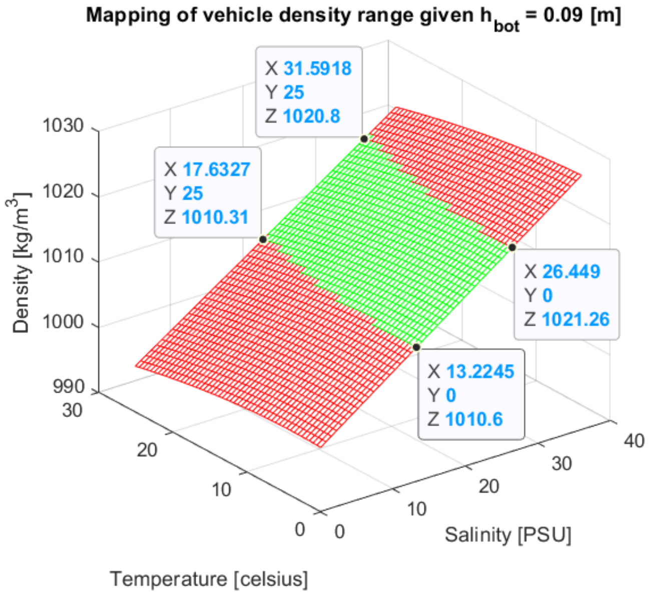


Figure 3.5: Green: density range which the vehicle is able to operate in by moving the piston Red: Outside density range of the vehicle

Chapter 4

Mathematical simulation

As previously stated, a mathematical model was developed during the specialization project to analyze the performance and limitations of the BV. The model was implemented using MATLAB/Simulink and aimed to accurately represent the vehicle's dynamics in various environments and situations. Since then, the model has been further developed to more accurately represent the physical BV, including the refactorization of PID controller to accurately mimic the controller of the BV firmware. Additionally, a search algorithm for presetting the outer lid has been integrated into the model.

4.1 Simulink model

The Simulink model contains three essential subsystems, namely the ”PID-controller”, ”Piston Dynamics”, and ”System Equations”, as depicted in Figure 4.1. The model is implemented as a feedback loop, where the simulated depth is subtracted from the reference depth to obtain the resulting error, which is then fed forward to the controller.

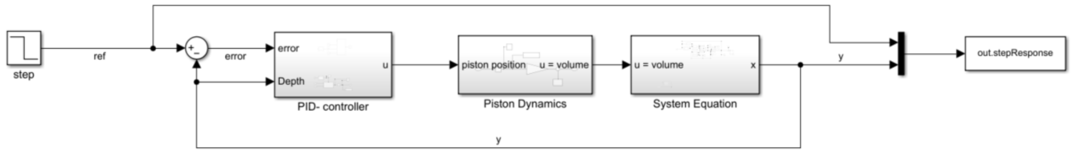


Figure 4.1: Overview of the system in Simulink, implemented as a feedback loop. Where the ”step” block contains the reference depth

4.1.1 PID-Controller

The PID controller implemented in this system is designed to effectively search for a piston displacement that results in an equilibrium between the vehicle and the water density. If the vehicle is above the target location, the controller will displace the piston to make the vehicle heavier and cause it to sink. Conversely, if the vehicle is below the target location, the controller will displace the piston towards the bottom, making the vehicle less dense and allowing it to float upwards. In essence, the PID controller's primary objective is to maintain the vehicle's depth and ensure that it remains in the desired position.

The PID controller is implemented in discrete form as equation 2.5, in Chapter 2 and has a time constant of 0.5 seconds. By utilizing a timer and trigger block, new calculations are done every time constant. As depicted in figure 4.2, each term of the PID controller is implemented using Matlab functions to mimic the control algorithm utilized in the BV firmware. The subsystem receives error and depth as inputs and outputs the actuator input. It should be noted that this model assumes a constant reference value, resulting in a derivative of zero. As discussed in section 2.2, meaning the derivative term only uses depth for calculating the derivative. To reduce noise in the derivative term, the depth measurements were filtered using an exponential moving average low-pass filter, as described in the EMA filter section 2.4.1. In order to ensure that the controller output does not exceed the physical limitation of the piston position in the BV, the output is limited between the range of [0, 51] mm. This ensures that the piston does not move beyond its physical range of motion.

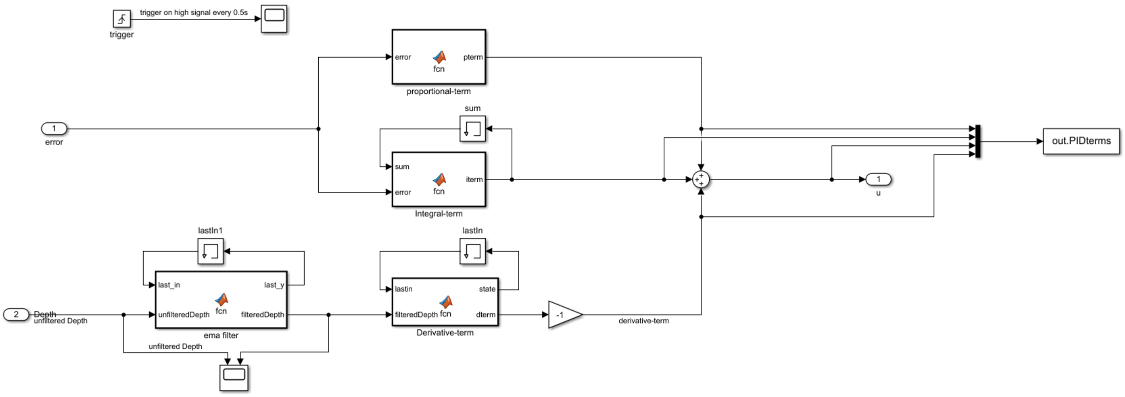


Figure 4.2: Parallel discrete PID controller and EMA lowpass filter implemented in simulink using matlab functions and memory blocks.

Below is s short description of each matlab function block and the corresponding code in that function:

- **P-term:** proportional term is calculated by simply multiplying the error with a constant.

```
function pterm = fcn(error, Kp)
    pterm = Kp*error;
end
```

- **I-term:** The integral term functions in two stages. Firstly, it checks if the error is below a certain threshold. This approach takes advantage of the fact that the integral term is activated when the vehicle is close to the target depth. Secondly, the integral term is computed, but with the additional check that the sum of the integral term does not surpass the physical limitations of the BV. To prevent the integral output from exceeding the physical limits, an integral anti-windup method is implemented.

```
function iterm = fcn(sum, error, Ki, sampleTime, integralThreshold)
    % sum is a memory block of previously calculated iterm.
    if (abs(error) < integralThreshold)
        iterm = sum + error*Ki*sampleTime;

        %integral anti wind up
        if iterm > 0.051
            iterm = 0.051;
        elseif iterm < 0
            iterm = 0;
        end
    else
        iterm = sum;
end
```

- **D-term:** The derivative term calculates the depth derivative and stores the previous depth measurement in a memory block for the next calculation.

```
function [state, dterm] = fcn(lastin, filteredDepth, sampleTime, Kd)
    %compute derivative
    dinput = ((filteredDepth - lastin)/sampleTime)*Kd;

    % store some variables for next calculation
    state = filteredDepth;
end
```

- **EMA-Filter:** Exponential moving average filter is applied to remove noise from the depth measurement before calculating the derivative term. The filter is implemented using equation 2.8, in Chapter 2.

```
function [last_y,filteredDepth ] = fcn(last_in, unfilteredDepth, alpha)

    % Calulate EMA filtered depth
    filteredDepth  =(alpha*unfilteredDepth) + ((1-alpha)*last_in);

    %store variable for next calculation
    last_y = filteredDepth;
end
```

4.1.2 Piston Dynamics

The piston dynamics subsystem shown in figure 4.3 takes piston position as input, and outputs the volume of the vehicle. The subsystem is responsible for modeling the dynamics of the piston, and the volume of the BV. The ”rate limit” block is utilized to limit the linear speed of the piston to match its physical characteristics, which is 2.33mm/s. The position of the piston is multiplied by the piston area and subtracted from the maximum volume of the vehicle to calculate the effective volume of the vehicle.

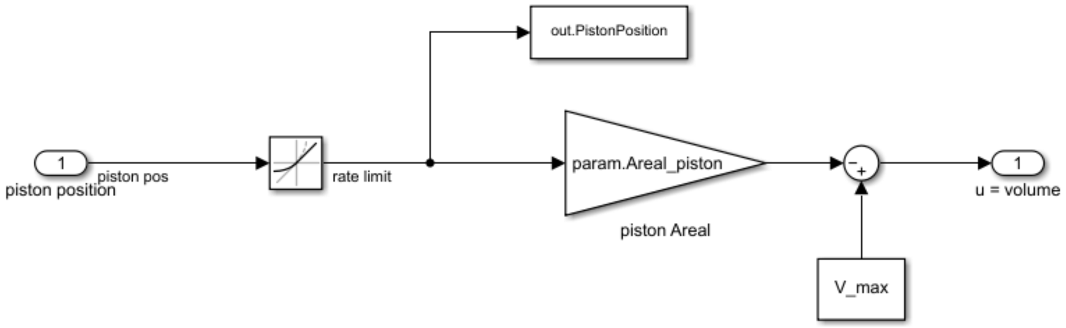


Figure 4.3: piston dynamics subsystem

4.1.3 System equation

The system equation subsystem implements the equation of motion in the vertical direction of the BV, as shown in equation 2.7, Chapter 2. The ”fluid density” block takes depth as input and outputs the water density. The density can be either constant, representing fresh water, or a function of depth, utilizing the density profile obtained by a CTD probe. As fluid density directly affects buoyancy and drag, it is crucial that fluid density accurately represents the target location. The ”surface limitation” block ensures that only the part of the vehicle that is submerged in water is affected by the buoyancy force. The acceleration is integrated twice to obtain the position, which is then affected by signal noise to represent noise in the measurements.

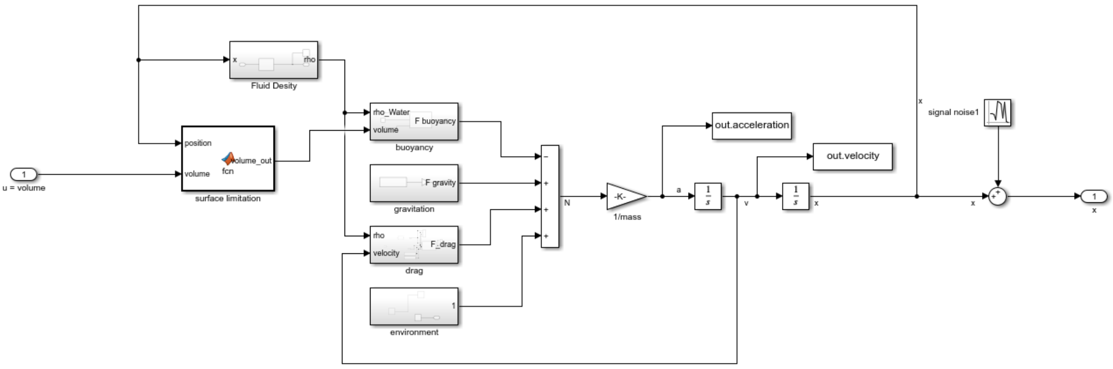


Figure 4.4: System equation subsystem

4.2 Presetting of outer lid

The model presented in this chapter has a unique feature that calculates the presetting of outer lid for a specific target location. This feature utilizes a density profile of the target location to search for an equilibrium at the minimum seawater density when the piston is displaced 10mm. The first 10mm of piston displacement is dedicated to allowing the vehicle to float to the surface. To illustrate, Figure 4.5 displays the maximum and minimum vehicle densities, as well as the density profile obtained in Børsa fall 2022. Using this profile, the model has found a presetting of $h_{bot} = 95.8mm$, which enables the vehicle to manipulate its density between the lines seen in Figure 4.5. It is noteworthy that the minimum vehicle density is less than that of water, which ensures that the vehicle is able to float to the surface. As the density of seawater can vary over time, safety margins are essential to ensure the vehicle’s successful return to the surface. Additionally, the maximum vehicle density crosses the water density at approximately 14.5 meters, meaning that with full piston displacement, the vehicle will be in equilibrium with the seawater and unable to dive any deeper. It should be mentioned that the density profile at Børsa is affected by low-density fresh water, from a nearby river. Causing higher density range, and limiting diving range of BV.

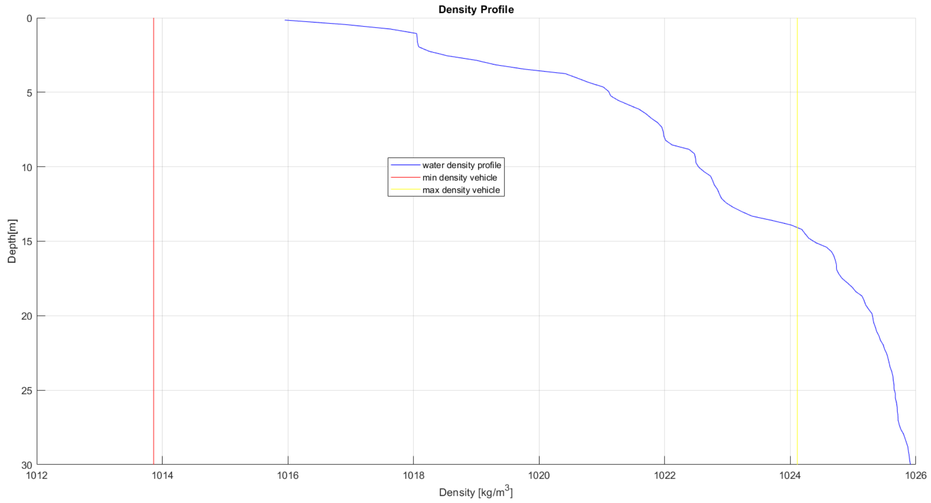


Figure 4.5: Plot of vehicle density at the maximum/minimum, corresponding to piston at bottom/upper location, and plot of density profile obtained from CTD- probe.

Chapter 5

Buoyancy vehicle

In This chapter, we will discuss modules and hardware implemented in the buoyancy vehicle (BV), and interface to the main processing unit. The design of the vehicle are based upon a use case derived during the specialization project, which is included in Appendix B. Further, we will discuss the firmware of the BV and the program flow. Finally, we will describe the mechanical modifications of the BV during the project.

5.1 Modules

Modules in embedded systems play a significant role in reducing complexity, improving reusability, and facilitating system scalability. Structuring a system using modules allows for greater ease in design, development, and debugging processes. Each module can be designed to perform a specific function, and then various modules can be integrated to build a complex system. In this section we will discuss the modules implemented in the BV. A high level block diagram of the BV hardware can be seen in figure 5.1

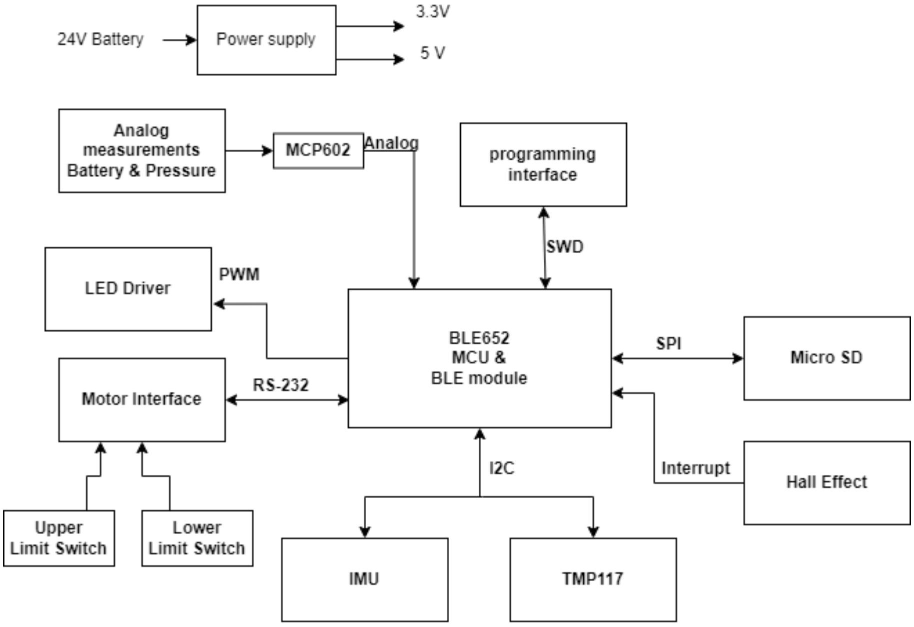


Figure 5.1: High level block diagram of the BV hardware

5.1.1 BLE652

The BLE652 module, depicted in Figure 5.2, incorporates Nordic Semiconductor's nRF52832 chip. This module is equipped with 31 General-Purpose Input/Output (GPIO) pins, which serve as interfaces for connecting with additional modules, external components, and devices. To enhance the Bluetooth signal, the module employs a FlexPIFA antenna.

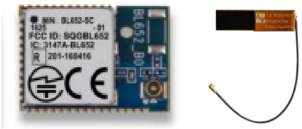


Figure 5.2: BL652 Radio module and FlexPIFA antenna

5.1.2 Si7210 Hall effect sensor

The Hall-effect sensor from Littelfuse [12], is a magnetic sensor designed to generate a low voltage signal in the presence of a nearby magnet. In this particular application, the original sensor had to be replaced due to an unknown error. The sensor is strategically positioned 13 cm above the pressure sensor. It functions by producing an interrupt signal that is relayed to the main processing unit. The sensor serves two primary purposes: waking the BV from sleep mode and initiating the configuration mode. Additionally, the sensor is utilized to abort a mission in the event that an unintended mission has been initiated.

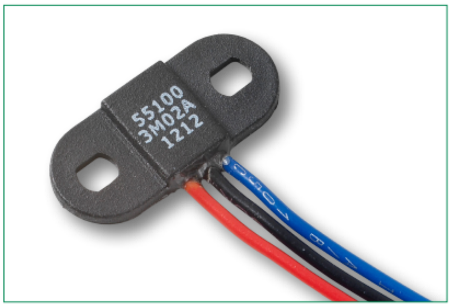
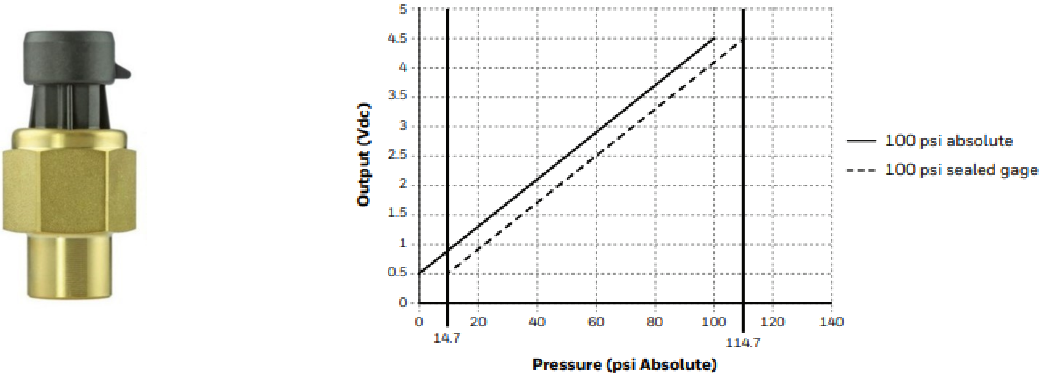


Figure 5.3: Hall-effect sensor. Reproduced from [12]

5.1.3 Pressure Sensor

The Honeywell PX3-series Heavy Duty Pressure Transducer [13], employs a piezoresistive element in conjunction with an Application Specific Integrated Circuit (ASIC). The specific model of the sensor is PX3AN1BH100PSAAX, and is illustrated in Figure 5.4. The sensor operates on a 5V input voltage and generates a linear output voltage within the range of 0.5V to 4.5V. It is designed to measure pressure ranging from 14.7 psi to 117.7 psi.

Pressure can be determined from the linear equation $y = a * x + b$, where y represents voltage and x represents pressure in PSI. The parameter a corresponds to the rate of change, which can be expressed as the derivative $a = \frac{dy}{dx} = \frac{4.5-0.5}{114.7-14.7}$, while b represents the constant term of 0.5. Solving this equation for x yields the expression $x = (y-b)\frac{dx}{dy}$, which provides a means of calculating pressure based on the measured voltage.



(a)

(b)

Figure 5.4: **a)** PX3 pressure sensor. **b)** Output voltage, given absolute pressure. Reproduced from data sheet [13]

5.1.4 Stepper Motor

The actuator responsible for piston movement and volume adjustment in the vehicle is the stepper motor. Figure 5.5 displays the Trinamic PD60-3-1161 stepper motor along with the TMCL-1160 control module [14]. The stepper motor operates within a voltage range of 10-30V, with a nominal supply voltage of 24V. The control module is an advanced microprocessor that features programmable TMCL firmware and flash storage. Two operational modes are available for the motor: standalone mode, where a program is uploaded directly to the control module, and direct mode, where individual commands are transmitted through serial communication. Direct mode is preferred in this scenario, as the nRF52832 serves as the primary processing unit. To establish communication with the control unit, the RS-232 serial communication protocol, and UART peripheral are utilized.

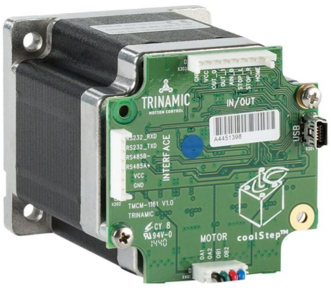


Figure 5.5: PD60-3-1160 Stepper Motor, and TMCL-1161 Control Module. Reproduced from data sheet [14]

Limit switches

During the installation of the BV, it is essential to account for the piston's variable initial starting position. To address this, limit switches are employed to establish a known position when restarting the system. Additionally, these limit switches serve to protect the linear actuator and piston from potential damage in the event of motor failure to stop. The TMCL-1161 control module offers three Input/Output pins specifically designed to accommodate end-limit switch inputs.

For this application, D2MQ-4L switches [15] mounted on 3D printed holders have been selected as the limit switches of choice. The Limit switch is connected to the TMCL-1161 control module, with the NC (Normally Closed) pin connected to ground. When the switch is activated, it interrupts the connection to ground, prompting the TMCL control module to trigger a signal that prevents the motor from moving in that particular direction.

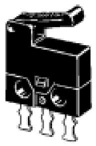


Figure 5.6: D2MQ-4l-1-L Limit switch. Reproduced from datasheet [15]

5.1.5 ICM20948 Motion Sensor

The ICM20948 motion sensor is a comprehensive module that integrates a 3-axis gyroscope, accelerometer, and compass. Additionally, it incorporates a 16-bit analog-to-digital converter (ADC) and a programmable filter [16]. The chip itself supports both I2C and SPI interfaces, but the module is restricted to utilizing the I2C interface. Regarding the accelerometer, it provides digital output measurements in the X, Y, and Z directions. The module allows for programmable full-scale range options, including ±2g, ±4g, ±8g, and ±16g. Furthermore, it incorporates a 16-bit ADC to ensure high-resolution measurements. The motion sensor continuously measures accelerometer and gyroscope during a dive and stores the data on the SD card for further processing and evaluation of movement during the mission,

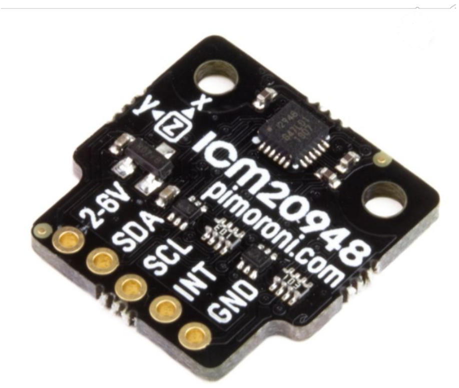


Figure 5.7: ICM20948 motion sensor mounted on the PIM448 module, reproduced from [17]

5.1.6 TMP117 Temperature Sensor

The TMP117 sensor, developed by Texas Instruments and depicted in Figure 5.8, is a low-power digital temperature sensor with a 16-bit resolution and I2C interface [18]. The sensor has an accuracy of ±0.1°C, and provides reliable temperature measurements within the range of -20°C to 50°C. During a dive, the TMP117 sensor continuously monitors and records the temperature. The temperature data is logged onto the SD card for further analysis. To facilitate its integration, the sensor is mounted on a 2.54x2.54 cm PCB module provided by Sparkfun. It is strategically positioned at the bottom of the BV, around the hex nut of the pressure sensor. It should be mentioned that the sensor does not measure water temperature directly, but the casting of the pressure sensor, which is assumed to have the same temperature as the surrounding water.

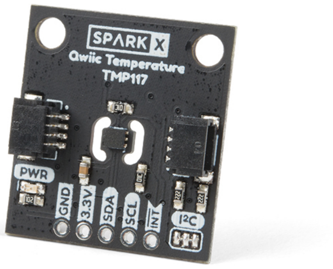
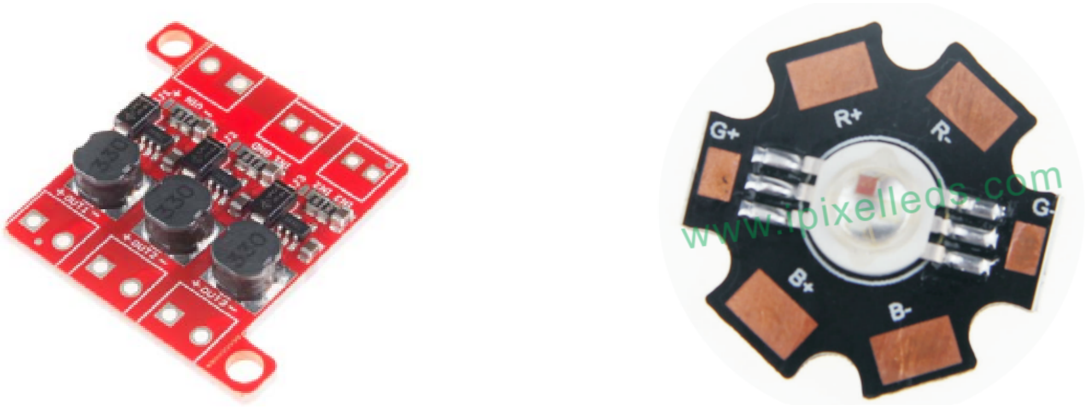


Figure 5.8: TMP117 temperature sensor module. Reproduced from [18]

5.1.7 RGB LED and Driver

The LED is positioned on the upper part of the BV, serving as an indicator of the BV's current state. It also plays a crucial role in facilitating the BV's visibility in water. To drive the LED, the PicoBuck 12v LED driver is utilized. This module is a high-power and constant current LED driver with three channels [19]. Each channel corresponds to the brightness control of red, green, and blue components of the LED. The brightness levels can be adjusted using either analog or Pulse Width Modulation (PWM) signals.



(a)

(b)

Figure 5.9: **a)** PicoBuck 3 channel LED driver. reproduced from datasheet [19] **b)** RGB powered LED. reproduced from [20]

5.2 nRF51832 and Seggger Embedded Studio

The BV relies on the nRF52832 microcontroller, developed by Nordic Semiconductor, as the main processing unit. This 32-bit System-on-Chip (SoC) provides the necessary GPIO pins and peripherals, including hardware support for Bluetooth Low Energy (BLE) functionality. For software development environment, Segger Embedded Studio v4.18 is used and can be downloaded from the official website at: https://www.segger.com/downloads/embedded-studio/. The firmware is flashed onto the SoC using the Segger debugger and the Serial Wire Debug (SWD) interface.

The firmware itself is built upon the nRF5 Software Development Kit (SDK) v16.0, which is available for download from https://www.nordicsemi.com/Products/Development-software/nrf5-sdk/download. This version of the SDK offers a comprehensive development environment specifically tailored for nRF52 series SoC. It includes a wide range of drivers, libraries, examples, softDevices, and radio protocols, providing a solid foundation for BV firmware development.

During the development process, the ability to access print statements is highly beneficial for debugging purposes. The SWD interface not only facilitates programming and debugging but also allows for the transmission of print statements to a terminal on the host computer. Real Time Transfer (RTT) is the mechanism used for this purpose. To view the printed messages, the RTT viewer terminal application can be downloaded from https://www.segger.com/products/debug-probes/j-link/tools/rtt-viewer/.

5.3 Firmware

The firmware is mainly written by Halsos, who created most of the drivers, Bluetooth communication and Finite State Machine (FSM) for the core program flow, with different states. Bluetooth configurations are build upon Nordic Semiconductor's softDevice, which is a pre-compiled and linked binary file. Detailed information about drivers and Bluetooth configuration won't be included in this report, but can be found in Halsos report [3]. During the project, a number of bugs were found and additional features have been made. Additionally, the I2C driver made by Halsos has been replaced, as the original driver failed when interacting with the ICM20948 chip.

5.3.1 Finite State Machine

The core program flow of the system is controlled by a Finite State Machine (FSM). The FSM consists of several states, each with its own set of transitions and conditions. State flow diagrams depicting these transitions and conditions can be found in Figure 5.10. The FSM is structured with five main states: INIT, IDLE, CONFIGURE, MISSION, and PICK-UP. These states represent different operational modes and functionalities of the system. Additionally, there are three additional states: SLEEP, FAILURE, and LOW-POWER, which play critical roles in handling exceptional situations. To provide visual feedback about the operational state of the BV, the color of the LED is used. Each state is associated with a specific LED color, making it easier to identify the current state of the system.

Below is a brief description of each state and its corresponding LED color:

- **RESET:** When the reset button is activated, it triggers a hard reset, causing the peripherals and core modules to be initialized before entering the FSM.
- **INIT:** The first state that is entered after a hard reset with the reset button or a soft reset triggered by the hall-effect button from sleep mode is the INIT state. In the INIT state, two timers are initiated to measure the battery voltage every 10 seconds and to update the FSM every 0.5 seconds. Additionally, in the INIT state, the LED is turned off.
- **IDLE:** After the INIT state, the IDLE state is entered, acting as an intermediary state between the system's off-mode and configuration mode. In the IDLE state, a time-to-sleep timer is activated. If this timer reaches 2 minutes, the BV transitions to the sleep state. To enter the CONFIGURE state, the hall-effect button must be activated, which resets and turns off the time-to-sleep timer. The IDLE state is indicated by a white LED light.
- **CONFIGURE:** In the CONFIGURE state, the Bluetooth functionality is activated, allowing the BV to start broadcasting and be connected via a dedicated desktop application or a serial terminal application on a phone or tablet. The detailed instructions for connecting to the BV can be found in section 5.4. In configuration mode, the vehicle and mission settings can be modified using a menu-based structure, which is explained in detail in section 5.4.1. Within the CONFIGURE state, BLE commands are utilized for transition to either the MISSION state or the IDLE state, additionally activating the hall-effect button triggers MISSION state with default mission and vehicle settings. The CONFIGURE state is indicated by a blue LED light.
- **MISSION:** In the MISSION state, Bluetooth stops advertising, and the mission algorithm is executed continuously until the mission time expires. Details of the mission algorithm are

described in section 5.3.3. Once the mission is completed or aborted using the hall-effect button, the piston is moved to the position $h_{piston} = 0$ in order to maximize buoyancy, allowing the BV to float to the water surface. The MISSION state is indicated by a green LED light. The PICK-UP state is triggered when the final mission is finished or when the hall-effect button is activated.

- **PICK-UP:** In the PICK-UP state, the BV utilizes BLE advertising to facilitate the transfer of mission files, vehicle reconfiguration, changing mission settings, or transitioning to another state. Through BLE commands, it is possible to enter the IDLE, CONFIGURE, or MISSION state. The PICK-UP state is indicated by a yellow LED light.
- **SLEEP:** The SLEEP state is entered when the 2-minute timer in the IDLE state expires. In this state, the system enters a low-power mode to minimize power consumption. To exit the SLEEP state, the system can be either reset by activating the RESET button or triggered by the hall-effect button, which initiates a soft reset. Upon exiting the SLEEP state, all settings are initialized to their default values. During the SLEEP mode, the LED is turned off to conserve power.
- **FAILURE:** In the event of a detected software fault, the BV enters the FAILURE state directly to ensure that it can safely float to the surface. In the FAILURE state, the piston is moved to $h_{piston} = 0$ to maximize buoyancy. To recover from the FAILURE state, a hard reset is required. The FAILURE state is indicated by a pink LED light.
- **LOW-POWER:** The battery voltage is continuously monitored every 0.5 seconds during a mission and every 10 seconds otherwise. If the battery voltage drops below 12.8 volts, the BV enters the LOW-POWER state directly. In the LOW-POWER state, the piston is moved to $h_{piston} = 0$ to ensure that the BV floats to the surface. The LOW-POWER state is indicated by a red LED light.

5.3.2 TWIM/I2C

The Two-wire Interface Master (TWIM) is a two-wire half-duplex master that enables communication with multiple slave devices connected to the same bus. It is compatible with the Inter-Integrated Circuit (I2C) protocol. To utilize the TWIM functionality, a driver instance is created using the SDK driver. This can be achieved by calling the following line of code: `nrf_drv_twi_t m_twi = NRF_DRV_TWI_INSTANCE(0);`. Detailed documentation for the TWIM driver can be found on the official website [21]. To handle events generated by the driver, an event handler is defined. This event handler is responsible for processing events specified in the driver description, such as the NRF_TWIM_EVENT_DONE event that is triggered when a TWIM operation, either send or receive, is completed.

Using the TWIM instance, several functions have been developed to facilitate reading from and writing to registers in devices such as the ICM and TMP117. For example, the `uint16_t TMP117_read(uint8_t reg)` function reads from an 8-bit register and returns the corresponding 16-bit register value. These functions abstract the low-level TWIM operations and provide a convenient interface for accessing specific registers in the target devices.

By leveraging the capabilities of the TWIM driver and implementing specific functions, the application can interact with the connected ICM20948 and TMP117 modules, enabling read and write operations to their respective registers.

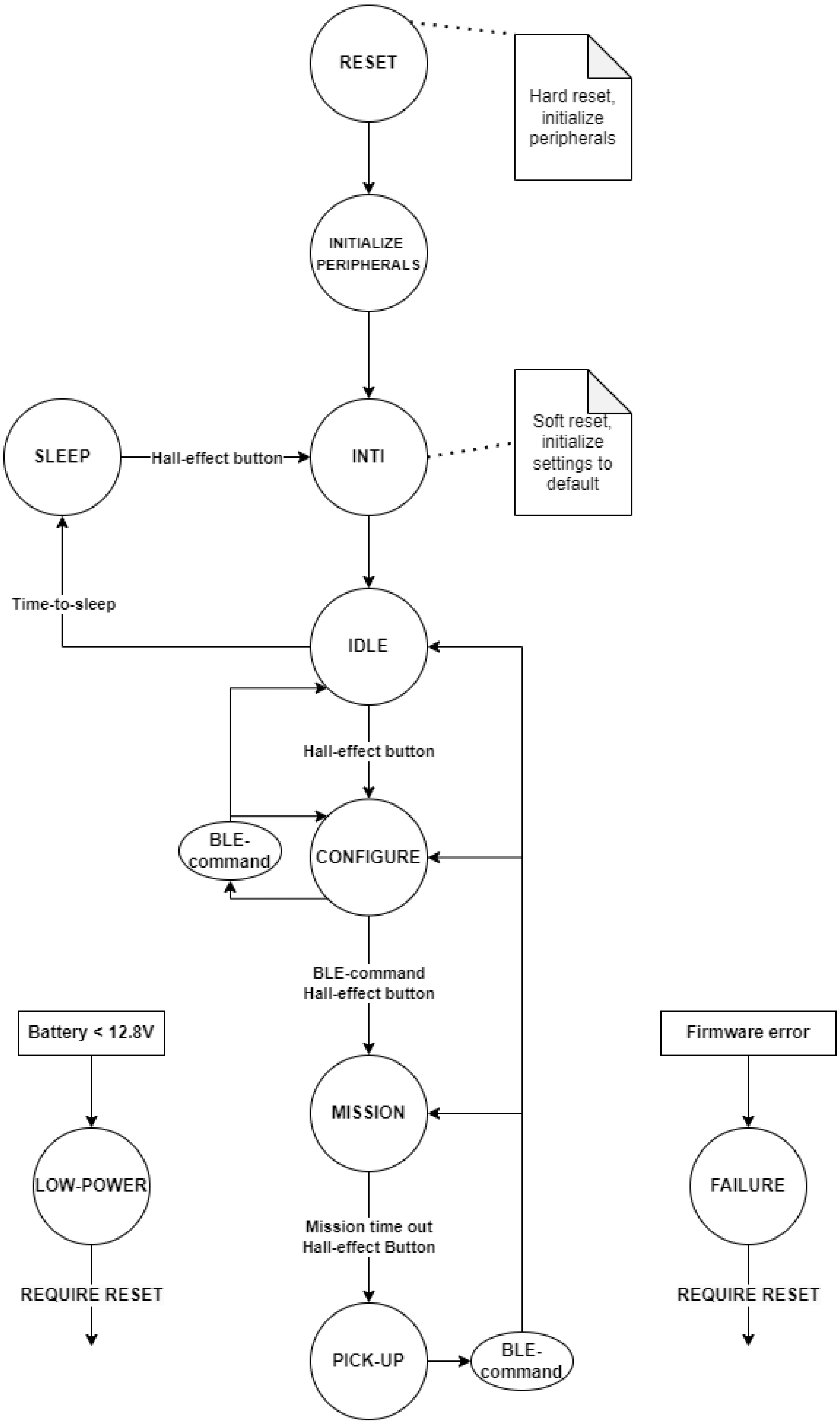


Figure 5.10: Flow diagram describing the core program flow managed by Finite State Machine

5.3.3 Mission

Upon system boot, the "mission" and "PID" structures are initialized with default values. This includes default PID coefficients and default mission time and depth for mission 1 and 2, while mission 3 and 4 are set to zero as placeholders. It's worth mentioning that each mission is defined by a specific depth and time duration.

To prepare for a new mission, the function $prepareMission()$ is executed. This function updates the PID coefficients to the latest values and determines the number of missions to be executed. Additionally, it creates a new mission file on the SD card, naming it with a number higher than the latest existing file (e.g. 23.txt).

The mission is initiated by calling the function $runMission()$. The flow of the mission execution is outlined in Figure 5.11. To facilitate data logging and sensor sampling, a timer is started to update the mission log and collect sensor data every 0.5 seconds. Furthermore, an initial timestamp is recorded to mark the start of the mission execution at time 0.0 seconds. The mission execution is structured within a for loop, which iterates over each configured mission. Each iteration of the for loop corresponds to a specific mission with its defined time and depth parameters. Within this loop, a while loop is employed to continuously read sensor data, calculate the depth, write mission log data to the SD card, and perform PID calculations. The while loop continues until the mission time expires, at which point it is terminated.

By utilizing this mission execution framework, the BV can effectively execute multiple missions consecutively, adhering to the defined time and depth parameters. The continuous sampling and logging of sensor data, along with the real-time PID calculations, enable precise control and monitoring of the BV's behavior throughout the mission execution.

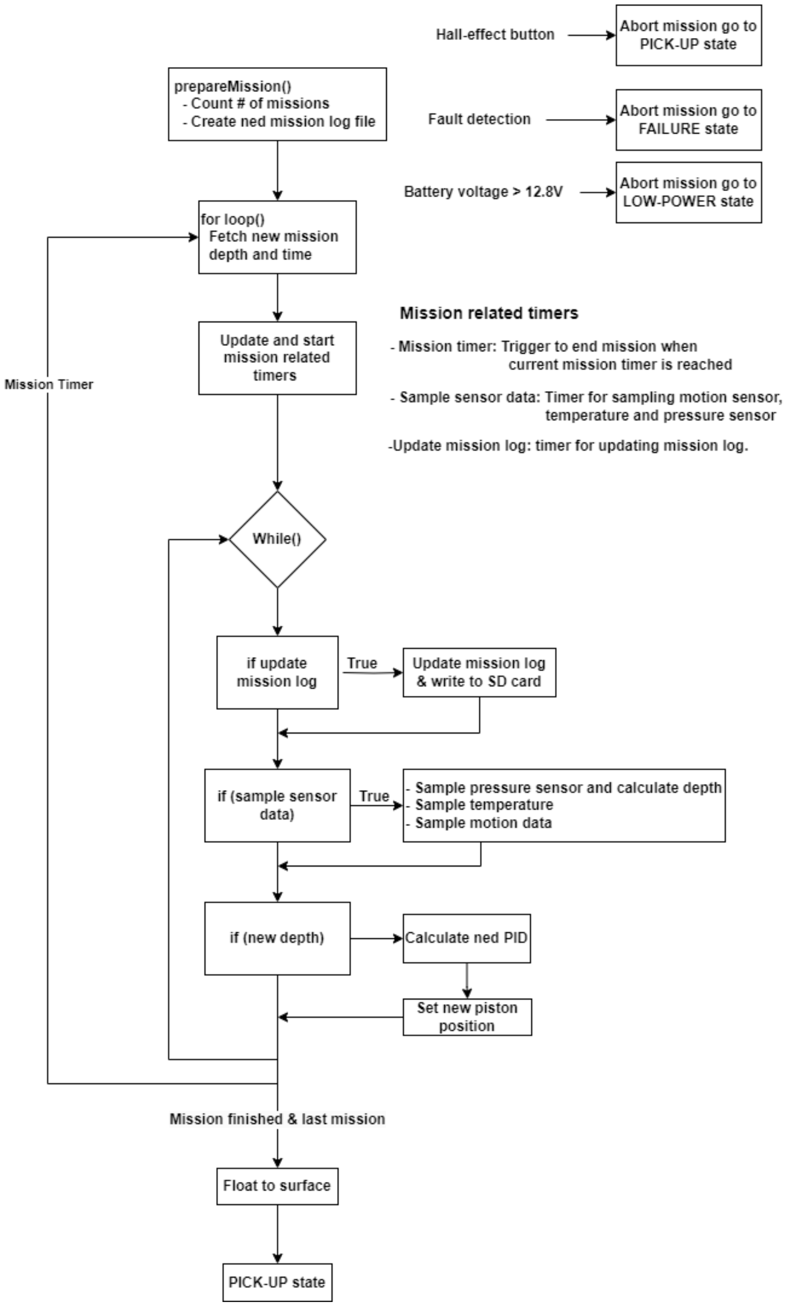

Hall-effect button
Abort mission go to PICK-UP state
prepareMission()
- Count # of missions
- Create ned mission log file
Fault detection
Abort mission go to FAILURE state
Battery voltage > 12.8V
Abort mission go to LOW-POWER state
for loop()
Fetch new mission depth and time
Update and start mission related timers
Mission Timer
Mission related timers
- Mission timer: Trigger to end mission when current mission timer is reached
- Sample sensor data: Timer for sampling motion sensor, temperature and pressure sensor
-Update mission log: timer for updating mission log.
While()
if update mission log
True
Update mission log & write to SD card
if (sample sensor data)
True
- Sample pressure sensor and calculate depth
- Sample temperature
- Sample motion data
if (new depth)
Calculate ned PID
Set new piston position
Mission finished & last mission
Float to surface
PICK-UP state

Figure 5.11: Flowchart explaining firmware mission flow

5.3.4 PID Controller

The PID controller is a crucial algorithm employed in BV to achieve and maintain a specific depth. The implementation of the PID controller is based on plain C code, utilizing the discrete PID equation 2.5, in Chapter 2. The PID controller is based on a library by Ruben Santa Anna [22], and operates in two different configurations, depending on the vehicle's proximity to the target depth. When the vehicle is within a certain threshold of the target depth, the controller functions as a PID controller, otherwise, it operates as a PD controller.

To mitigate the impact of noise on depth measurements, an EMA filter is employed. Further details regarding the EMA filter can be found in Section 2.4.1, in Chapter 2. The filter is specifically used to calculate the derivative term in the PID controller. To prevent the integral term from exceeding the physical limits of the underwater vehicle, an integral anti-windup mechanism is employed. This mechanism ensures that the integral term does not ”blow up” beyond the predefined boundaries of the vehicle.

Finally, the output of the PID controller corresponds to the piston position, which is constrained within the range of 0 to 51 mm, reflecting the physical limitations of the vehicle's operation. Below is a pseudo code for the PID calculation algorithm:

```
procedure pid_compute(pid)

    in <- unfiltered depth
    inFiltered <- filtered depth
    lastin <- filtered depth from previous calculation
    tau <- time since last computation

    // Compute error
    error = setpoint - in

    // Compute integral
    iterm = iterm + (Ki * tau * error)

    // Integral anti windup
    if iterm > omax then
        iterm = omax
    else if iterm < omin then
        iterm = omin

    // Compute differential on input
    dinput = (inFiltered - lastin)/tau

    // Compute PID output
    out = Kp * error + iterm - Kd * dinput

    // Apply limit to output value
    if out > omax then
        out = omax
    else if out < omin then
        out = omin

    pid.out = out

    // Keep track of variables for next execution
    lastin = inFiltered
end procedure
```

5.3.5 Calibrating Pressure Sensor

The accuracy of the pressure sensor is influenced by variations in atmospheric pressure. To account for this, the pressure sensor can be calibrated before initiating a new mission, this calibration process is activated using BLE commands. The calibration of the pressure sensor involves taking 40 measurements at intervals of 500ms. During this calibration procedure, the BV is positioned to float on the water's surface. By averaging these measurements, the obtained value is considered as the offset. This offset value is used to correct for any deviation from the expected value of 0 meters, ensuring accurate pressure readings. By calibrating the pressure sensor in this manner, the impact of atmospheric pressure differences on the sensor's measurements is effectively minimized, leading to more precise depth calculations during the mission.

5.4 BLE Commands

Bluetooth commands are utilized in order to modify vehicle and mission settings, transfer files, calibrate the pressure sensor, and alter the FSM state. The BV can establish a connection either through a desktop application developed by Halsos or via a mobile application. The desktop application is created using the JavaScript framework Node.js, instructions on downloading and getting started with the desktop application is described in detail by Halsos [3]

The functionality of the application has been thoroughly tested and validated on Mac OS by Halsos, as well as on Windows during this project. The Windows and Linux versions of the application utilize the noble API [23] to access Bluetooth drivers. It should be noted that compatible USB 4.0 USB adapters are required for Windows and Linux platforms. Throughout this project, the "CSR8510 A10" USB adapter has been used successfully.

The BV can also be connected through third-party applications that emulate UART over BLE such as "nRF UART v2.0" on Android and "nRF toolbox" on iPhone.

5.4.1 Menu

Figure 5.12 depicts a block diagram illustrates the user interface BLE commands. This menu structure is presented to the client upon establishing a connection and consists of three sub-menus: Set mission data, configure vehicle, and mission files. The "Set mission data" sub-menu enables the configuration of mission time and depth parameters. In the "configure vehicle" sub-menu, user can adjust PID coefficients, alpha value in the EMA filter, as well as perform pressure sensor calibration. The "Mission files" sub-menu facilitates the transfer and deletion of files stored on the SD card.

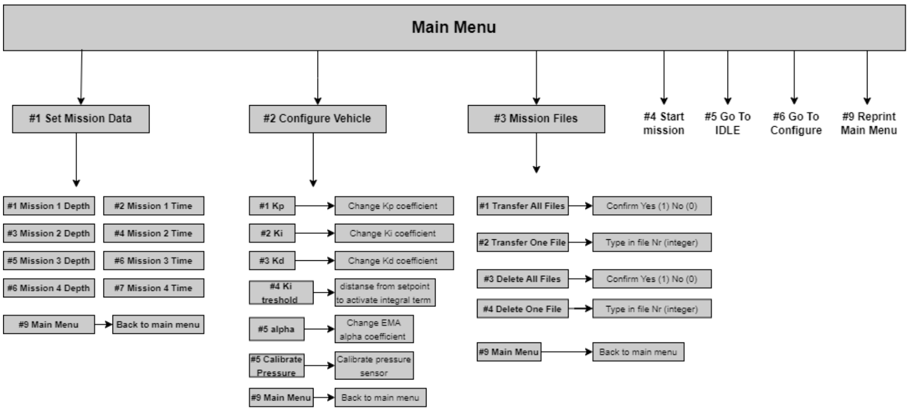


Figure 5.12: Diagram to illustrate user interface menu structure.

5.5 Mechanical modifications

In order to meet the evolving needs of the BV and tracing system, several mechanical modifications were introduced throughout the course of this project. This section focuses on presenting these modifications, which were implemented using the computer-aided design program ”Solidworks.” The program was used for creation of 3D models for prototyping and assembling the vehicle, ensuring the compatibility and proper fit of the modifications with the existing vehicle.

5.5.1 Hall-effect Holder

To ensure the hall-effect sensor remains securely positioned near the house wall and to prevent any movement or wire entanglement during dives, a dedicated holder has been made and is illustrated in Figure 5.13. The purpose of this holder is to securely fasten the hall-effect button in place and maintain its proximity to the wall, minimizing the distance required to activate the button using an external magnet.

The holder has been designed to fit between the battery package and the inner diameter of the vehicle house, providing a stable location for the hall-effect sensor. To minimize any negative impact on the vehicle’s buoyancy, the holder has been 3D printed with 50% infill, reducing its weight while maintaining structural integrity.

By utilizing this specially designed holder, the hall-effect sensor can be reliably positioned near the house wall, ensuring accurate and consistent functionality during dives without the risk of movement or wire entanglement.

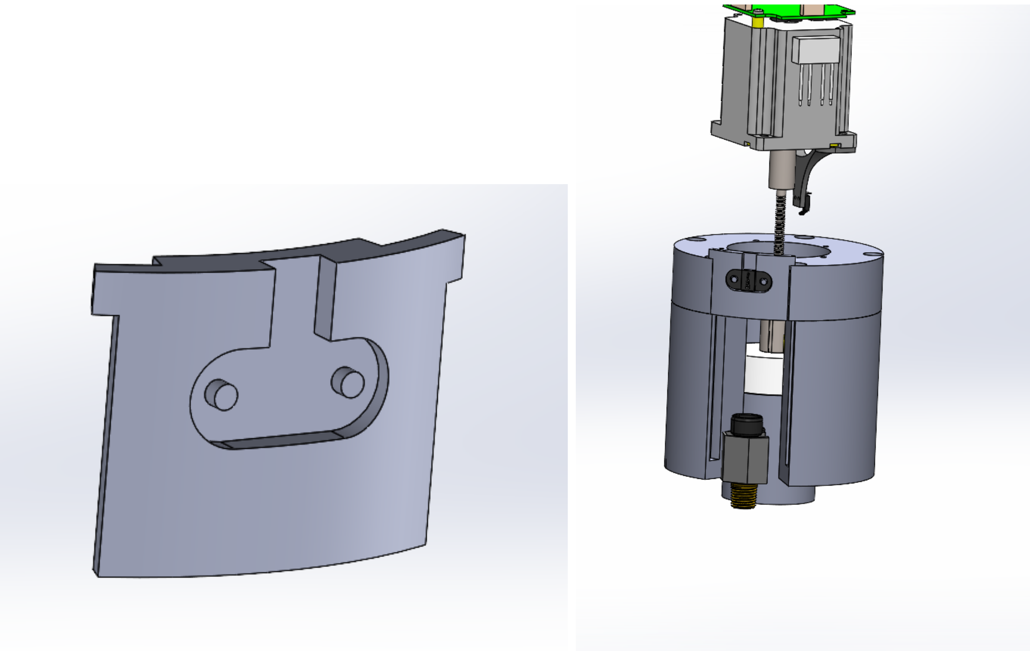


Figure 5.13: Hall effect holder, located 13 cm above the pressure sensor

5.5.2 Line Guider

To ensure safety and prevent the loss of the vehicle during seawater testing, a secure attachment has been developed for the BV. A line guide attachment, as depicted in Figure 5.14, has been specifically designed for this purpose.

The line guide is fabricated using polyoxymethylene (POM) material and machined at the ”Kybernetisk verksted.” It is strategically mounted on the outer lid of the BV to provide a reliable and secure connection. The attachment process involves passing a rope through a 50mm hole in the line guide and attaching a weight to the rope. The weight ensures that the rope remains taut, extending from the seabed to the surface, effectively preventing the BV from drifting horizontally.

This line guide attachment serves as a crucial safety measure, ensuring the BV remains securely in place during seawater testing. By allowing unrestricted vertical movement while effectively preventing horizontal drift, the line guide attachment minimizes the risk of losing the vehicle and provides a stable platform for conducting experiments in the open sea environment.

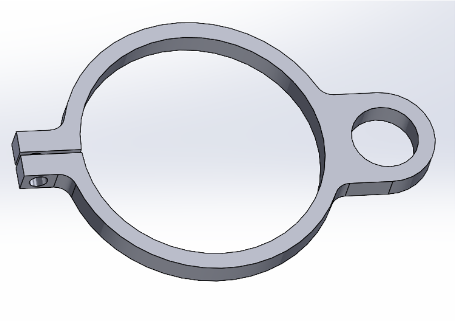


Figure 5.14: Machined line guider for securing the vehicle during testing in the sea

5.5.3 Fish Tag Holder

The 3D printed fish tag holder, as depicted in Figure 5.15, is specifically designed to securely hold a 13-mm fish tag. To ensure optimal signal transmission, the holder is strategically mounted on the top of the vehicle, as shown in Figure 5.15. The holder is fabricated using PLS material through 3D printing, with a high infillment rate of 80% to ensure robustness and stability of the fish tag during dives.

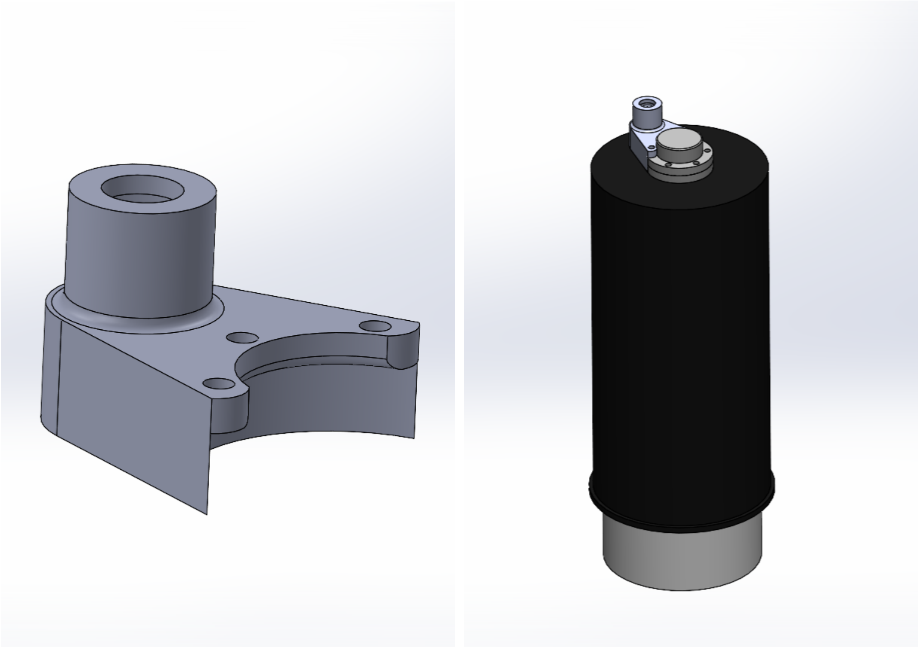


Figure 5.15: Fish tag holder, designed to be mounted on LED House on top of vehicle.

Chapter 6

Tracking System

Gathering data about subsurface ocean currents is achieved by letting the BV drift with the current, and tracking the movement. In this tracking system, the position of the BV is calculated using the time difference of arrival (TDoA). This method employs a transmitter that is mounted to the BV, which emits data packets at regular intervals. Surrounding the area are three strategically placed hydrophones that detect the transmitted data packets and record the precise time of their arrival. The collected time of arrival and data is subsequently transmitted to a SLIM module, the data is further transmitted to a host computer, where sophisticated calculations are performed to determine the position of the BV. Figure 6.1 illustrates the setup of the tracking system

In this chapter, we will discuss the utilization and implementation of the hydrophones and fish tags designed by Thelma Biotel. These devices play a crucial role in facilitating the transmission and reception of data packages between the BV and the water surface. Additionally, we will discuss the functionality of the Synchronization and LoRa Interface Module (SLIM), which serves as a receiver for data captured by the hydrophones and enables the seamless transfer of this data to a host computer. The SLIM module also assures the synchronizing of the clocks of the three hydrophones, a critical requirement for accurately timestamping the TDoA measurements. Finally, we will discuss the transmission of data from the SLIM module to a host computer and the processing of calculating the position.

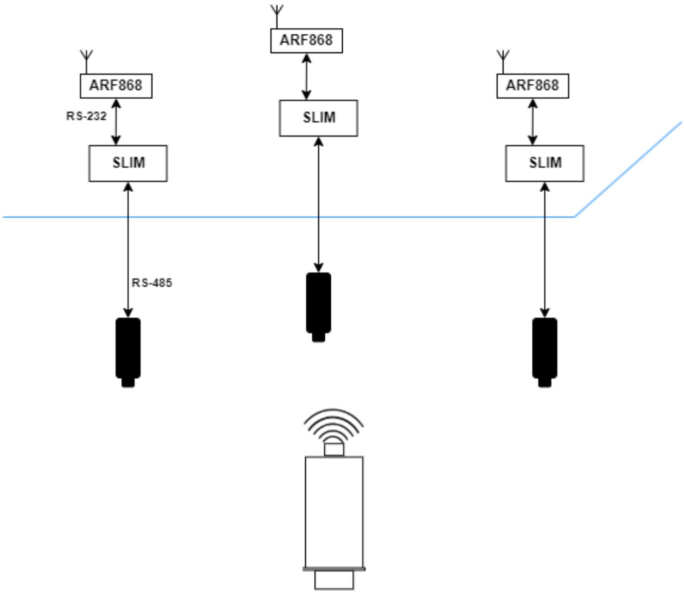


Figure 6.1: Illustration of tracking system setup. Each node is defined as hydrophone, SLIM and ARF868 radio modem combined

6.1 Fish tag and Hydrophone

Both the fish tag and hydrophone components, developed by Thelma Biotel, are illustrated in Figure 6.2. The fish tag act like an acoustic transmitter and is available in various models, featuring diverse sizes and configurations in terms of power output and transmission intervals. Its primary function is to emit a data package using sound waves, which is subsequently decoded by the TBR700RT hydrophone.

Upon detecting of a new data package, the TBR700RT hydrophone promptly timestamps the arrival of the signal. It then decodes the contents of the package, storing the data locally. Additionally, the data is transmitted as a text string through the RS-485 data bus. Connectivity between the TBR700RT hydrophone and the SLIM module is established via RS-485, enabling data transfer from the TBR700RT to the SLIM module, as well as enabling command transmission from the SLIM module to the hydrophone. Detailed technical specifications and information about the hydrophone can be obtained from the Thelma Biotel datasheets [24].

To prevent clock drift in the hydrophone, the SLIM module issues synchronization commands. Two types of synchronization commands, namely ”basic sync” and ”advance sync,” are utilized. The ”advance sync” command conveys a 9-digit UNIX timestamp, where the final digit represents tenths of a second and is assumed to be zero. On the other hand, the ”basic sync” command aligns the hydrophone's clock to the nearest 10-second interval. By employing these synchronization commands, the SLIM module ensures accurate and consistent timing between the hydrophone and the rest of the tracking system.

Additionally, it is important to note that the depth of the BV is included in the data package transmitted by the fish tag. By including the depth measurement in the data package, it enables the use of three hydrophones and SLIM modules for TDoA-based position calculations, instead of using four.

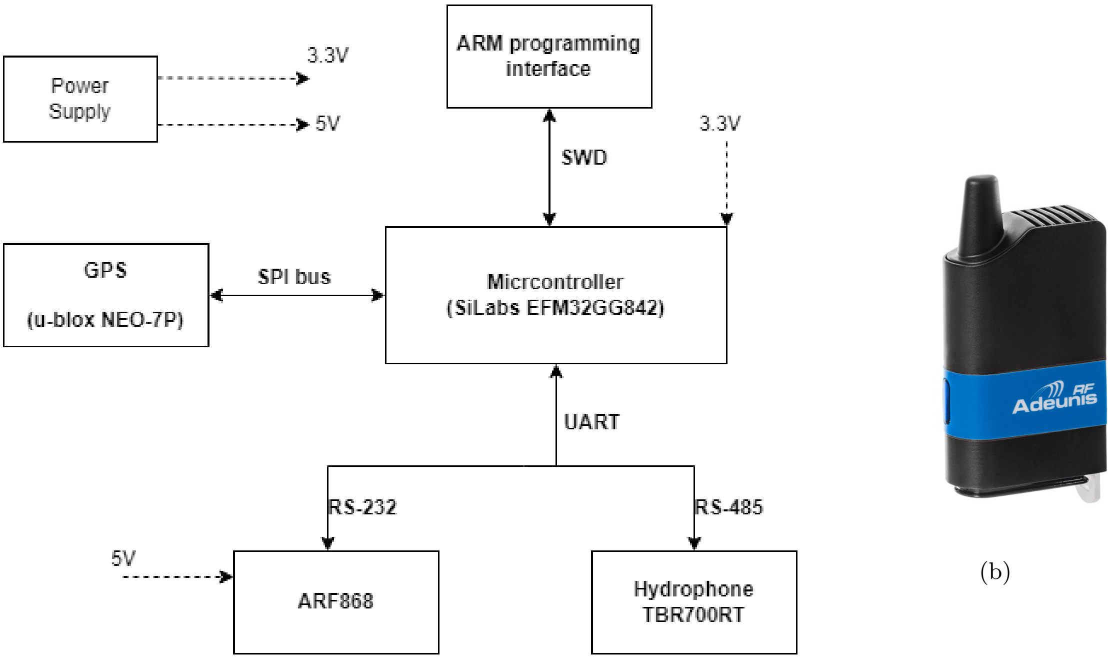
(a)

(b)

Figure 6.2: **a)** TBR700RT Hydrophone. **b)** 9mm acoustic transmitter. Reproduced from [25]

6.2 SLIM

Central to the tracking system is the Synchronisation and LoRa Interface Module (SLIM), which is a battery-operated surface unit that provide time synchronization for TBR700RT hydrophones and transmission of data from the hydrophone to a host computer. The slim module has been used in various Internet of Fish (IoF) concepts and originally operated with LoRa radio, but this has been replaced with an ARF868 standalone radio modem with UART interface. Figure 6.3 illustrates a high level block diagram of the modules used by the SLIM in this project and the ARF868 radio modem.



(a)

(b)

Figure 6.3: (a) High level block diagram of the SLIM module. (b) ARF868 radio modem

The core of the SLIM module is centered around an EFM32GG microcontroller, which serves as the main processing unit. This microcontroller, provided by Silicon Labs, are programmed using Simplicity Studio, a development environment that includes debugger with Serial Wire Debug (SWD) interface. The firmware for the SLIM module is written in the C programming language and is built upon the Gecko SDK Suite v3.3 and GNU ARM v7.1, both of which are available for free, and can be downloaded in the Simplicity Studio editor. To gain a better understanding of the firmware's operation, a flow diagram illustrating the firmware flow can be found in Figure 6.4.

The SLIM module includes a GPS module, specifically the u-blox NEO8 [26], which serves two main purposes. The GPS module provides an accurate time reference for synchronization, as well as the position of the SLIM module itself. The GPS module communicates with the main processor using the Serial Peripheral Interface (SPI). During the startup process, the GPS module undergoes a cold start, which typically takes approximately 26 seconds to acquire satellite signals, gather navigation data, and calculate a position solution, known as a fix. In contrast, a hot start takes only about 1 second. The GPS module not only provides the UNIX time but also emits a Pulse Per Second (PPS) signal, which is essential for clock synchronization. To achieve clock synchronization, the SLIM module leverages the 1PPS output from the GPS module. This signal is connected to an interrupt on the main processing unit, enabling precise timing control. By utilizing the UNIX time and the PPS signal, the SLIM module synchronizes the acoustic receivers through dedicated synchronization messages. This synchronization process eliminates any drift in the internal timer, ensuring accurate and consistent timing across the system.

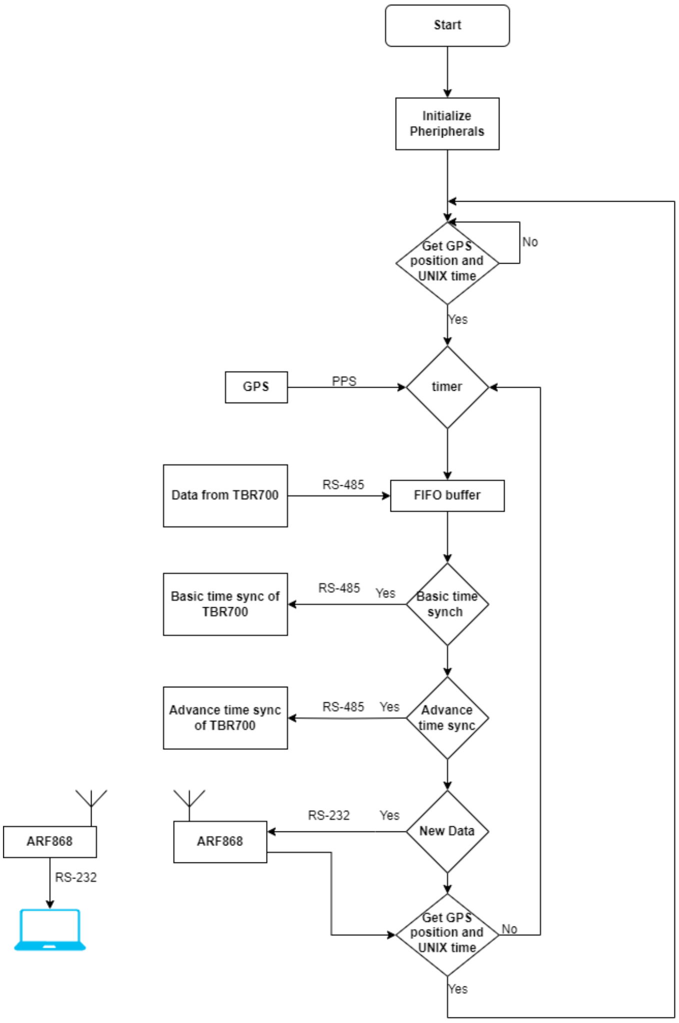


Figure 6.4: Flowchart explaining firmware operation of the SLIM.

The SLIM module is equipped with the ARF868 radio modem [27], which serve a crucial role in converting data from serial link into a radio frame for transmission. The modem is designed to facilitate both point-to-point and multi-point communication, this modem operates within the harmonized European 863-870MHz band, which can be used without a license. The radio modem receives dedicated power supply through a jack plug and is powered by the same battery as the SLIM module. Interface options for the modem include RS-232 and RS-485, with RS-232 being the chosen configuration for this project. This allows seamless communication between the the SLIM module and host computer. By utilizing the UART interface, specific commands can be sent to the modem for configuration purposes. These commands enable the setup of the radio modem, to suit the requirements of the tracking system. In the SLIM module, each node is equipped with a radio modem that facilitates the transmission of data from the acoustic receiver and GPS location. This data is sent to a host computer for storage and further processing, forming a crucial component of the overall tracking system's functionality.

A sharp Memory LCD display is included to provide on-site personnel with information on the operational status of the SLIM in the field. The display shows information about TBR700RT connection and time synchronization, as well as GPS status. The main advantage is that information can be read without opening the enclosure and the RS-232 interface can be dedicated to data transfer mainly.

6.3 SLIM-GUI

In this section we will discuss the SLIM GUI, implemented in Matlab. The GUI receives incoming data from the ARF868 radio modem through a serial port, and store the data locally on the host computer. When a tag is recognized, the GUI promptly calculates the TDoA position of the tag, and in real time plots the results. The GUI is activated by executing the "SLIM_GUI.m" script in Matlab. The illustration in figure 6.5 provides an example of a plot showcasing the BV's position during a dive. The GUI can be terminated by clicking on the "Stop Loop" button.

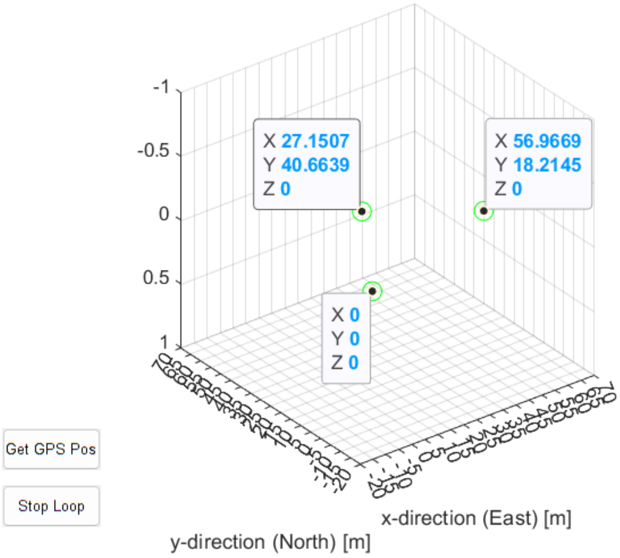


Figure 6.5: Real time position plot generated by SLIM-GUI. The green points illustrates the position of Node1, Node2 and Node3. The push button "Get GPS Pos" send command to the nodes to transmit GPS location, the other push button "Stop Loop" terminates the GUI and store the received data locally

Upon initialization, the GUI waits for the SLIM module to transmit its GPS coordinates. The received position, expressed in geographic coordinates (latitude and longitude), is then converted to xy Cartesian coordinates utilizing Matlab's "Mapping Toolbox". By default, the SLIM's updated position is transmitted every 2 minutes. However, the GUI can actively request the updated position by clicking on the "Get GPS Pos" button, causing the SLIM to transmit position to the host computer.

As outlined in section 2.6 of Chapter 2, TDoA calculations assume the second hydrophone to be aligned along the x-axis. To ensure the visual orientation of nodes remains consistent in the north/east direction, the nodes positions are rotated prior to TDoA position calculation, and the resultant position is then rotated back to align with the plot.

Chapter 7

Results

In this chapter, we will present the setup and results from physical experiments conducted on the BV and the tracking system. This chapter include the following experiments: first, the BV was tested in a controlled environment of a freshwater tank, allowing for successful tuning of the PID regulator. Next, the BV was tested in seawater conditions to evaluate its performance in a more realistic and challenging environment. Finally, the tracking system was tested by placing a fish tag at known locations in water, and analysing the system's functionality and the visualization of the obtained results. These experiments provide valuable insights into the performance and effectiveness of both the BV and the tracking system.

7.1 Functional Test Plan of BV

A functional test plan has been made to conduct functional verification of the vehicle before it undergoes testing in water. The purpose of this test plan is to ensure that all operational aspects of the vehicle, including the hall-effect button, LED, pressure sensor, log file, etc, are thoroughly tested. The functional test plan, which includes procedures and criteria for each test, is included in Appendix B.

7.2 Functional Test Plan of Tracking System

Maintaining clock synchronization between the three nodes is a critical aspect of the tracking system. To evaluate the effectiveness of clock synchronization, a test setup was conducted in dry conditions. The setup consisted of two hydrophones positioned with a fish tag placed in the middle. Ideally, the arrival times at the hydrophones would be equal, allowing for a comparison of the time of arrival. The test was conducted over a duration of three hours, with 183 recorded detections. The time differences between the two receivers, along with the corresponding percentages, are presented in Table 7.1.

Δt [s]	-0.002	-0.001	0	0.001
Percentage	5%	48%	42%	4%

Table 7.1: Time difference of the two receivers, over a duration of 3 hours and 183 detections.

In the majority of cases, the hydrophones exhibited identical timestamps or a deviation of 1 millisecond. Considering that sound travels at approximately 1500 m/s in water, a deviation of 1 millisecond corresponds to an error of 1.5 meters. This deviation should be taken into account when analyzing the accuracy of the system.

The results of the test demonstrate that the clock synchronization mechanism successfully maintains consistent timestamps between the hydrophones in most instances. The minimal time differences observed indicate that the synchronization method effectively mitigates clock drift and enables accurate TDoA calculations for position estimation.

7.3 Physical experiment #1

This section desribes the experimental testing of the vehicle in freshwater tank located at EL-bygget in Gløshaugen. The tank is approximately 2 meters deep, making it suitable for early stage testing of controller, filter and general performance. A density profile of the tank was not conducted, as a CTD probe was unavailable at the time. The tank has also an unknown amount of chlorine, which has unknown effects on the density. Its however assumed that the density is constant 997 $\frac{kg}{m^3}$, as no salt is dissolved in the water.

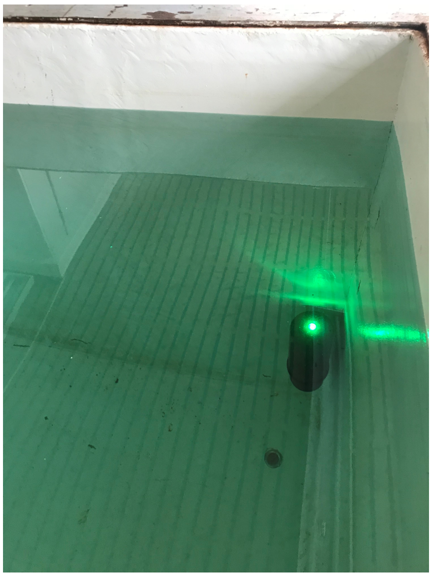


Figure 7.1: BV during a dive in fresh water tank located at Department of Electronic system

7.3.1 Tuning PID regulator

The tuning of the PID controller was carried out through a series of experimental steps. Initially, the P-controller was tuned, followed by the PI-controller, and finally, the derivative term was introduced to create a PID-controller. Throughout the tuning process, the raw depth measurement was used. However, to reduce noise in the derivative term, and improve performance, an EMA filter was introduced to smoothing out the depth measurement used to calculate the derivative term.

The buoyancy vehicle's outer lid was initially set to a predetermined value of $h_{bot} = 95mm$, which corresponds to the vehicle being in equilibrium with the water when the piston is positioned at $h_{piston} = 20mm$. The resulting density of the vehicle is depicted in Figure 7.2. To tune the regulator, a target depth of 1 meter was set for a duration of 3 minutes. The vehicle's step response and piston position were carefully analyzed during this period. These observations were instrumental in determining the appropriate adjustments required to optimize the controller's performance.

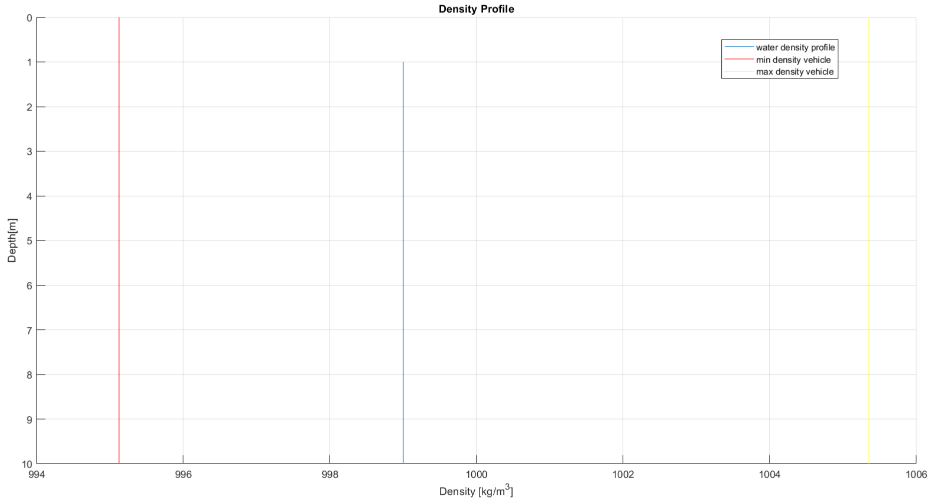


Figure 7.2: BV maximum and minimum density, compared to water density. The vehicle is preset to $h_{bot} = 95mm$, which result in an equilibrium when the piston position is $h_{piston} = 20mm$

P-tuning

The proportional term of the PID controller was set to $K_p = 0.023$, and the resulting step response is illustrated in figure 7.3. Upon analyzing the step response, it was observed that the vehicle continuously oscillates between 0.1 and 0.35 meters. This behavior corresponded to the piston oscillating between 17.5 and 24 mm, causing the vehicle to transition from a floating state to a sinking state during the mission. These observations indicate that the proportional term alone is insufficient to maintain stability in the system. The oscillatory behavior and steady state error in the step response is a common characteristics of a P-controller when it lacks the necessary control authority to stabilize the system.

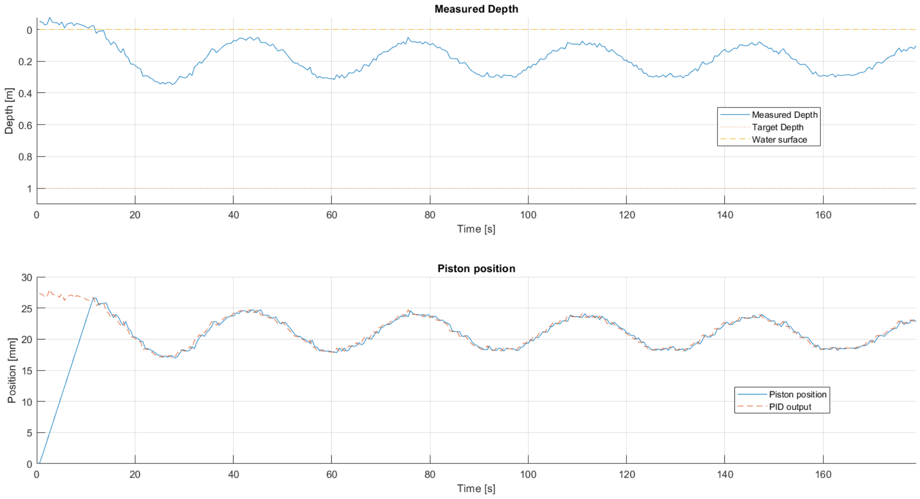


Figure 7.3: Step response and piston position, with P-controller only.

PI-tuning

The integral term of the PID controller was introduced with a coefficient of $K_I = 0.001$. The resulting step response and piston position are illustrated in Figure 7.4. The choice of a small coefficient for the integral term is deliberate, as its primary purpose is to search for a piston position that establishes an equilibrium between the BV and the water. Upon analysis of the step response, it was observed that the vehicle continuously oscillated between 1.2 and 0.75 meters, with the oscillations almost symmetrical around the desired setpoint of 1 meter. This behavior highlights the need for the introduction of a derivative term to eliminate oscillations and achieve a more stable response.

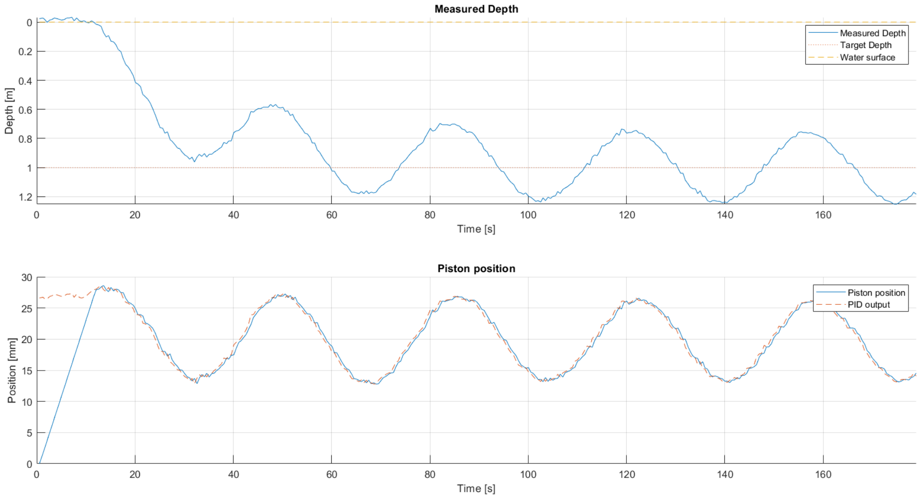


Figure 7.4: Step response and piston position, with PI-controller.

PID-tuning

The final stage of tuning involved introducing and fine-tuning the derivative term of the PID controller. The resulting step response and piston position can be observed in Figure 7.5. Through the tuning process, the optimum coefficient for the derivative term was determined to be $K_D = 0.1$. The introduction of the derivative term had a significant impact on the system's performance. It effectively reduced the oscillations, leading to a much more stable behavior of the BV around the setpoint. This improvement signifies the derivative term's ability to dampen the system's response to changes and enhance stability.

However, an interesting observation emerged during the tuning process. The high derivative coefficient introduced noise in the PID output, which can be seen in Figure 7.5. This noise in the output signal is a consequence of noise in depth measurements, which has a significant impact on the derivative calculations. Further adjustments or filtering techniques may be necessary to address this issue and achieve a smoother PID output.

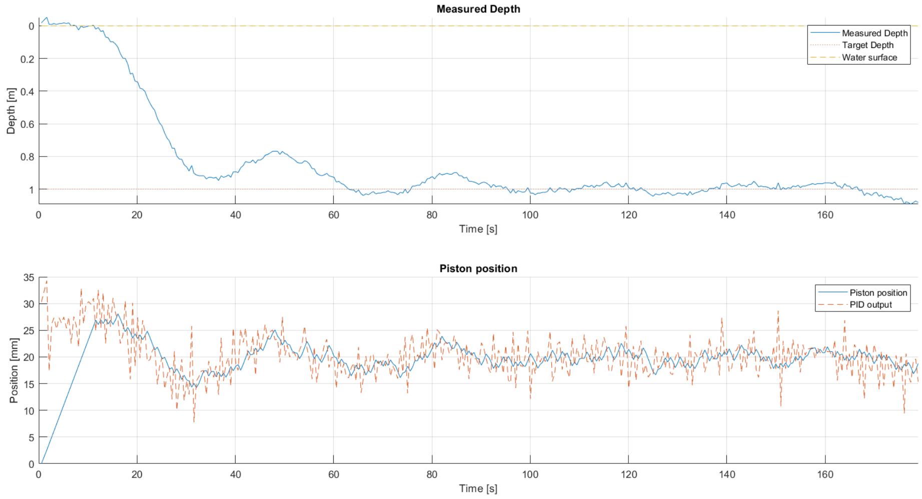


Figure 7.5: Step response and piston position, with PID-controller

EMA-filter tuning

An exponential moving average filter was implemented to reduce the noise sensitivity of the derivative term in the PID controller, as described in section 2.4.1. Tuning the filter is done by changing the alpha value, where $\alpha \in [0, 1]$. The optimum alpha was found to be $alpha = 0.4$. Figure 7.6 displays the raw depth measurements alongside their corresponding filtered measurements. The filtered derivative term exhibit a notable reduction in noise while still possessing sufficient bandwidth to accommodate rapid changes in depth, such as 0.8 meter in approximately 15 seconds. However, further reduction in noise can be achieved by decreasing the value of α, but at the expense of bandwidth. It is essential to tune the filter with a balance between noise reduction and system stability, as excessively reducing the bandwidth may result in an unstable system that is challenging to tune for the PID regulator. The optimum alpha value was found to be $\alpha = 0.4$.

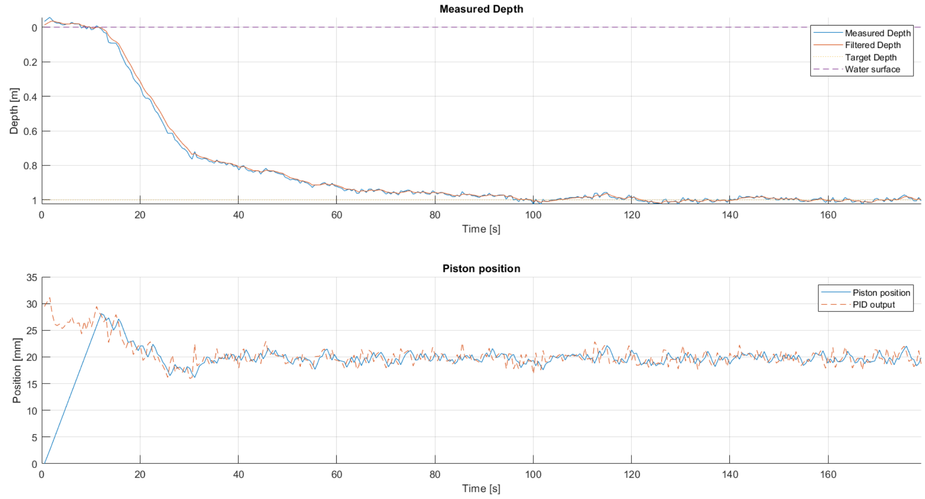


Figure 7.6: Step response and piston position of PID-controller and EMA-filtering of depth measurements.

7.3.2 Optimum PID control

K_P	K_I	K_D	Integral Threshold	EMA alpha	h_{bot}
0.026	0.001	0.1	1 [m]	0.4	95 [mm]

Table 7.2: Vehicle configuration parameters, during physical experiments #1

Table 7.2 displays the optimum vehicle configuration, previously found to be the optimum PID controller, EMA-filter, and presetting of the outer lid.

Presented in Figure 7.7 is the optimal regulator previously determined with two setpoints. The initial setpoint was set to achieve a depth of 1.25 meters for a duration of 180 seconds, followed by a second setpoint of 0.5 meters for the same duration. At the start of the experiment, the piston position is 0 mm. The PID output was approximately 30 mm due to an initial error of 1.25 meters. The piston then moved linearly at a speed of 2.27 mm/s (which is the programmed speed of the motor). The vehicle experienced a decreasing sinking rate, and after 70 seconds, it reached a depth of 1.17 meters. It took an additional 30 seconds for the vehicle to reach the target depth of 1.25 meters, after which it remained stable without any oscillations. At the 180-second mark, the second setpoint was applied, causing the vehicle to overshoot the target depth. This can be explained by the drag coefficient being lower when floating to the surface than when sinking, making the vehicle faster and more challenging to stop.

The simulated step response, utilizing identical K_p, K_i, K_d, and filter coefficients, has been included in order to replicate the diving scenario and its respective environmental conditions. Upon observation, it can be noted that the simulated response accurately matches the measured response of the vehicle for the first setpoint. Furthermore, the simulated piston position converges to the same equilibrium point as the logged piston position. An interesting phenomenon arises, where the first setpoint has a piston position of 20 mm at equilibrium, whereas the second setpoint has a piston position of approximately 17 mm. This can be reasonably explained by the presence of lower-density water located closer to the water surface, resulting in different equilibrium points. It should be noted that the simulation assumes a constant water density of 997 $\frac{kg}{m^3}$, resulting in a return to the same equilibrium for both setpoints. This also explains the minor differences between the simulation and measured step response for the second setpoint, where the simulation is slightly faster and has a slightly greater overshoot.

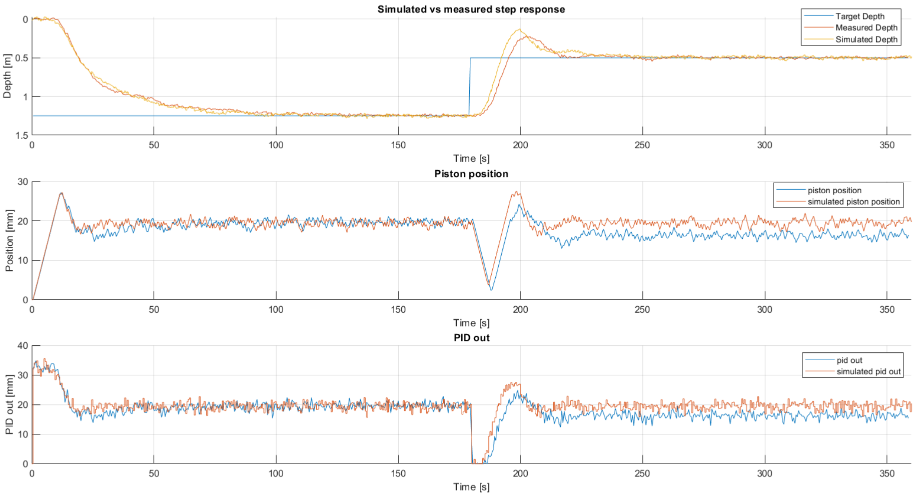


Figure 7.7: Simulated and logged step response, piston position and PID output

Analyzing PID contribution

To gain some understanding of the contribution of different terms in PID regulator, the terms were logged. Figure 7.8 displays the effect of P, I and D term in the controller.

- **P-term:** The proportional term is directly proportional to the error and is the dominant term at the beginning of the dive. However, as the BV moves towards the setpoint, the proportional term gradually decreases in influence. At 180 seconds, when the new setpoint is introduced, the proportional term undergoes a rapid change to -20mm, which is below the physical limitations of the piston. As the BV approaches the new setpoint, the proportional term decreases and eventually changes sign as the BV overshoots the setpoint before stabilizing, leading to a proportional term of zero.

- **I-term:** The integral term is activated at 15 seconds, as the error falls below the threshold of 1 meter. The integral term then increases rapidly before stabilizing at 20 mm, which is the equilibrium point of the BV. When the new setpoint is introduced, the integral term stabilizes at a new equilibrium point, corresponding to the different equilibrium point observed in 7.7. A relatively small K_i constant is chosen, as the primary role of the integral term is to accumulate the error and search for a piston position that results in an equilibrium between the BV vehicle and the water.

- **D-term:** The derivative term remains at zero until the BV begins to sink, at which point it becomes negative and reduces the sinking speed of the vehicle. As the sinking speed decreases, the derivative term gradually approaches zero. For the second setpoint, the derivative term is positive and reduces the speed of the BV as it floats to the new setpoint. The derivative term is the only term that utilizes filtered measurements. However, it is affected by noise, as can be observed by the 2 mm jumps in the derivative term when the vehicle is in equilibrium due to measurement noise.

- **PID-out:** PID output is the summation of P, I and D-term. The output is limited to the physical limits of the piston [0, 51]mm. In the beginning of the mission, the proportional term is the dominant, and as the vehicle are stabilizing the integral term donimates the PID output. Noise from the derivative term is always observed in the PID output.

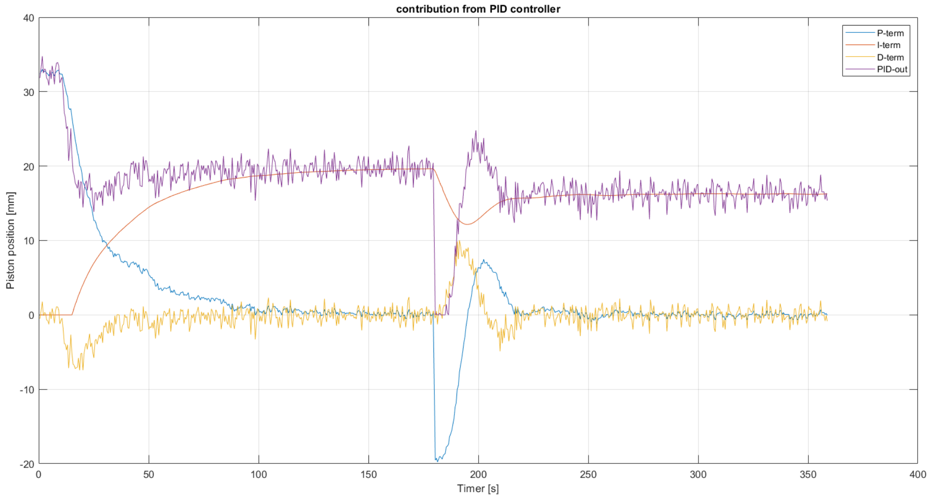


Figure 7.8: Contribution from proportional, integral and derivative terms in PID controller, with optimum PID controller in physical experiment #1.

7.3.3 IMU sensor

Figure 7.9 displays the measured tilt from the accelerometer during the experiment. It is evident that within the first 20 seconds after initiating the dive, the buoyancy vehicle exhibits a tilt ranging between 0 and -1 degrees. Subsequently, the vehicle stabilizes around -0.5 degrees and remains relatively stationary throughout the mission. These results align with the visual observations from the experiment, which indicate that the vehicle is not significantly affected by tilting and demonstrates stable behavior in the water. The consistent and minimal tilt suggests that the vehicle maintains its orientation and stability.

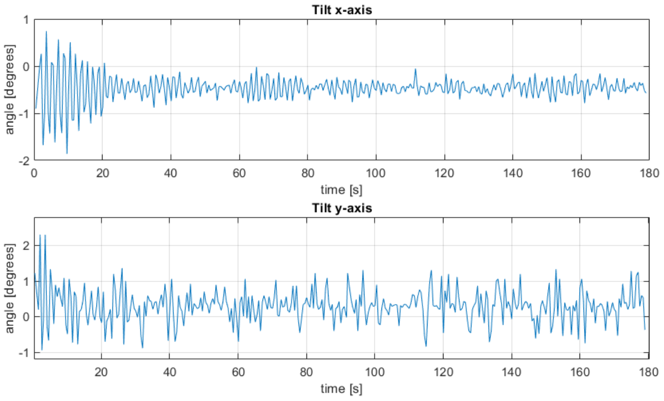


Figure 7.9: Tilt movement of the BV during dive in fresh water tank.

7.3.4 Temperature sensor

Figure 7.10 shows the measured temperature over time. Prior to the experiment, an external temperature sensor indicated that the tank was at 16 degrees Celsius. However, it is important to note that the temperature inside the vehicle may have also affected the measurement. The initial measurement taken using the TMP117 sensor was approximately 18.1 degrees Celsius, and this gradually decreased until reaching a steady state at 220 seconds, with a temperature of 17.8 degrees Celsius. It is worth mentioning that the sensor does not directly measure the temperature of the water, but rather the temperature of the pressure sensor casting. This explains the gradual decrease in measurement during the first 220 seconds.

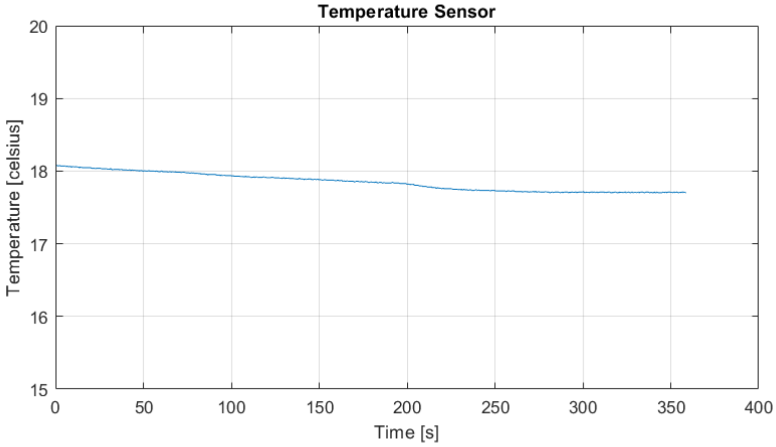


Figure 7.10: Logged temperature from TMP117 sensor, during a dive in fresh water tank.

7.4 Physical experiment #2

The second physical experiment was conducted in the harbor at Børsa, located outside Trondheim, to evaluate the performance of the vehicle in seawater. To minimize the risk of losing the vehicle, a line guide was attached. The physical environment of the experimental site is illustrated in figure 7.11a, while the line guide setup is shown in figure 7.11b. The line was attached to a weight and lowered to the seafloor, which was measured to be at a depth of 10 meters. The line guide was utilized to prevent any horizontal drift of the BV. Moreover, the line served as a method of retrieving the BV in case of any inability to return to the surface, by pulling it back to the dock.

(a) Location at Børsa, where the experiment was conducted

(b) The safety line in set up at a distance from the dock, to prevent the BV from colliding with the dock wall during the mission.

7.4.1 Equipment's

Below is a list of the essential equipment used during the experiment, the equipment is also shown in figure 7.12.

- **Buoyancy vehicle:** The BV is mounted with a line guider to ensure that the vehicle does not drift horizontally during the mission.
- **Computer:** A computer with Bluetooth connection to the BV is used to set mission data, configure the BV and deployment. The computer is also used to transfer mission files and plotting the result immediately after a dive is conducted.
- **CTD-probe:** A CTD-probe is used to obtain the density profile of the seawater.
- **Mounting tool:** As outer lid on the BV is hard to turn around, a special tool is needed for mounting the vehicle and adjusting the outer lid.
- **Rope and weight:** A rope of 20 meters and a weight of 10kg is used as a line guide to secure the vehicle during the mission.
- **Caliper:** A caliper is used to accurately mount the outer lid at the desired height. The caliper has an accuracy of 0.1 mm. Which is sufficient to ensure repeatability and accuracy when presetting the outer lid.

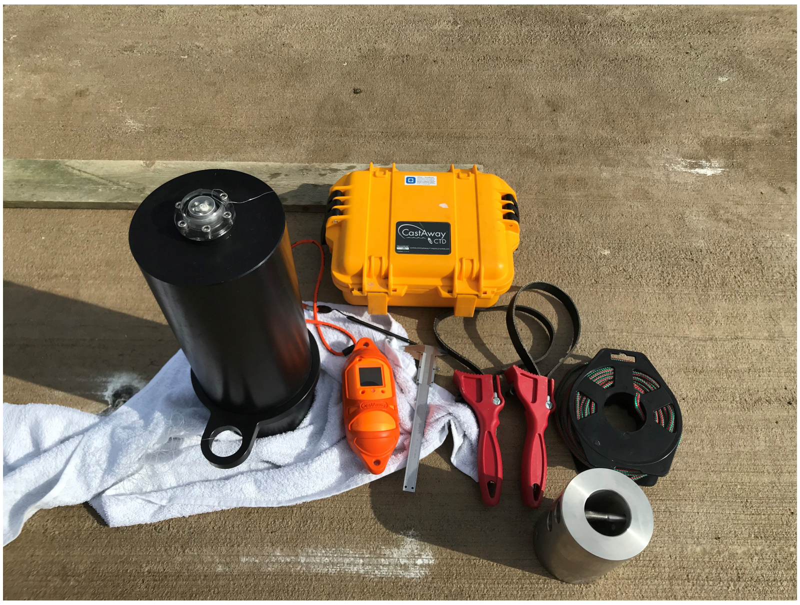


Figure 7.12: Equipment used during the experiment

7.4.2 Presetting of outer lid

Figure 7.13 demonstrates the density profile at the location, indicating the maximum and minimum seawater densities within which the vehicle can operate after the outer lid has been preset. The lowest seawater density is found at the water surface and remains nearly constant for the first 0.6 meters of depth. A noteworthy observation is the sudden increase in water density from approximately 0.8 to 1.6 meters, where the density rises from about 1020 $\frac{kg}{m^3}$ to nearly 1026 $\frac{kg}{m^3}$. Subsequently, the seawater density remains relatively constant from 1.8 meters until the seafloor at 10 meters. The most crucial consideration when evaluating the water density is ensuring the vehicle's ability to return to the surface after completing a mission, where the piston is at the bottom position, and the vehicle's density is at its lowest. Using the MATLAB simulation and the obtained density profile, a search algorithm was utilized to determine an appropriate presetting of the outer lid, resulting in a presetting of $h_{bot} = 91.3mm$. This allows the BV to operate within the maximum and minimum density range shown in fig 7.13, the BV's density ranges from 1018$\frac{kg}{m^3}$ to 1028$\frac{kg}{m^3}$ when the piston is at the bottom position and upper position, respectively.

As depicted in the figure, the BV has a minimum density of 1018$\frac{kg}{m^3}$, whereas the lowest seawater density is 1020$\frac{kg}{m^3}$. This indicates that the vehicle has a buoyancy force of 2$\frac{kg}{m^3}$ dedicated to floating to the surface. Additionally, the BV has a maximum density of 1028$\frac{kg}{m^3}$, while the density of seawater is 1026$\frac{kg}{m^3}$, indicating that the vehicle can sink to the bottom with no limitations. This is due to the fact that the BV's density is higher than the density of seawater.

If the outer lid is mounted incorrectly to a height of $h_{bot} = 90mm$, the density of the vehicle when the piston is at the bottom position would be 1021 $\frac{kg}{m^3}$. In this case, the vehicle would be in equilibrium at a depth of around 1 meter, as it would be unable to further increase its buoyancy force. Consequently, the vehicle would get stuck at 1 meter and be unable to return to the surface. If the BV is deployed without a safety line and the outer lid is mounted incorrectly, it is considered a worst-case scenario as there is a risk of losing the vehicle in the ocean. To mitigate such risks, it is crucial to ensure that the outer lid is mounted correctly and that the vehicle's buoyancy is adjusted appropriately to enable it to return to the surface after a mission.

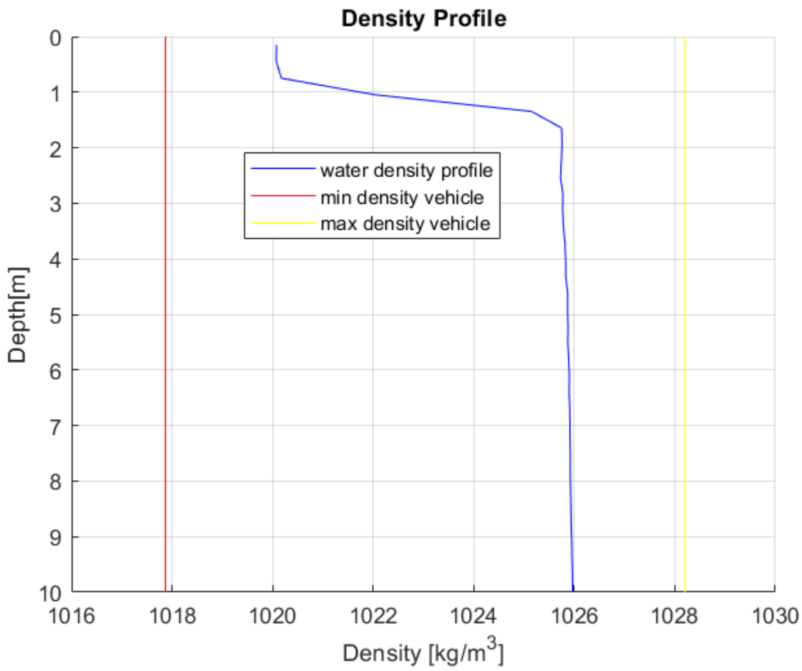


Figure 7.13: Density profile of the seawater, and maximum/minimum density of the vehicle by presetting the outer lid at $h_{bot} = 91.3mm$.

7.4.3 Vehicle configuration

Prior to deployment, a simulation was conducted to test the step response of the BV using the optimum PID values previously determined. The only modification made was setting the threshold for activating the integral term to 10 meters, which allowed for the integral term to be utilized throughout the entire mission. As well as mounting the outer lit at $h_{bot} = 91.3$mm, to match the seawater density. Table 7.3 provides an overview of the BV's configuration settings. Additionally, the pressure sensor was calibrated before the first dive.

K_P	K_I	K_D	Integral Threshold	EMA alpha	h_{bot}
0.026	0.001	0.1	10 [m]	0.4	91.3 [mm]

Table 7.3: Vehicle configuration parameters, during physical experiment #2

7.4.4 Mission 1

The first mission was set up to dive to a depth of 5 meters for 3 minutes. The resulting step response and piston position are shown in Figure 7.14. The corresponding PID calculations for the proportional, integral, and derivative terms are displayed in Figure 7.15

The first 15 seconds of the mission were allocated to submerging the vehicle and flipping it upside down to release any air trapped in the piston. The vehicle reached the desired depth of 5 meters after 87 seconds, but initially overshot before eventually stabilizing at the target depth after 140 seconds, where it remains stable until the end of the mission. The piston position began at 0mm (bottom position) and reached saturation after about 20 seconds, whereas the PID controller reached saturation from the beginning of the mission. In figure 7.15, one can observe that the P-term exceeded the saturation limits at the start of the mission, the integral term sum up to saturation limit after about 15 seconds, where the integral anti-wind up is limiting the integral sum to 55mm. After approximately 90 seconds, when the vehicle overshot the target depth, the PID controller began moving the piston downwards. This can be seen in the slow decay of the I-term, which eventually stabilized the piston at around 40mm where the vehicle's density was in equilibrium with that of the seawater.

One interesting observation is the variation in the vehicle's speed when diving to the first 1.5 meters, followed by a slower and constant speed from approximately 2 meters. A reasonable hypothesis is that the abrupt change in seawater density in this area increases the vehicle's buoyancy and pushes it upwards, leading to slower sinking speeds. Due to the slow dynamics in the system, the derivative term has a minimal impact on the PID output. Despite being negative during the sinking period (15-80 seconds), the proportional and integral terms saturate the output, effectively overriding the derivative term. The high gain of the proportional term also introduces noise in the PID output. As observed in the freshwater tank, the derivative term continually moves the piston 1-2mm for every new calculation, even when the vehicle is stable at the reference depth. This leads to significant battery drain, and it should be considered to implement logic that turns off the proportional term when the BV is stable at the reference point to reduce unnecessary battery consumption.

One can observe in figure 7.14 that the PID output saturates at 55mm, whereas the piston position saturates at 51mm, in later time it was found that a bug in the software saturates the PID output between 0 and 55 mm, while limit switches limits the piston between 0 and 51 mm.

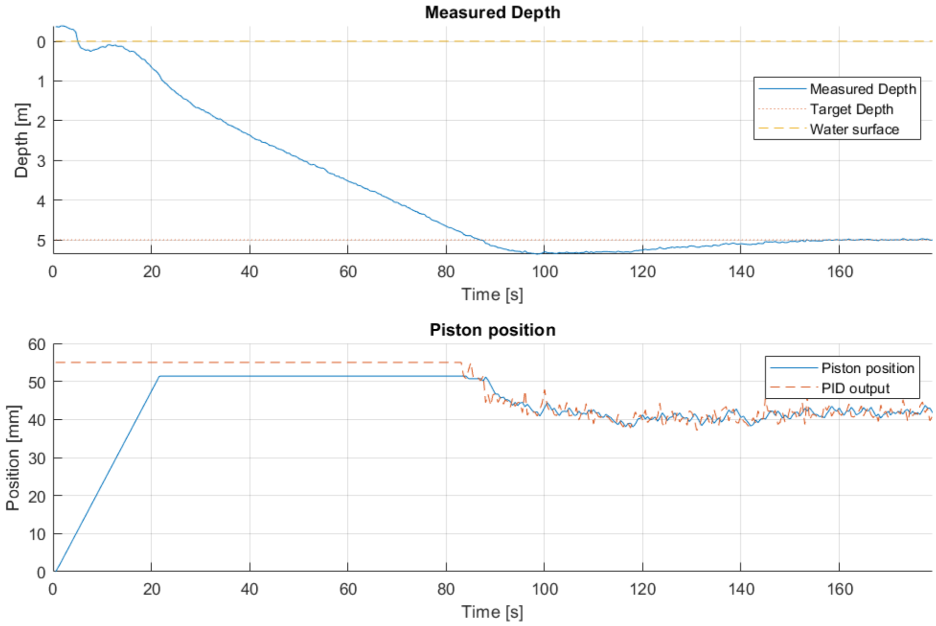


Figure 7.14: **TOP:** BV step response, with target depth of 5 meters. **BOTTOM:** BV piston position during the mission.

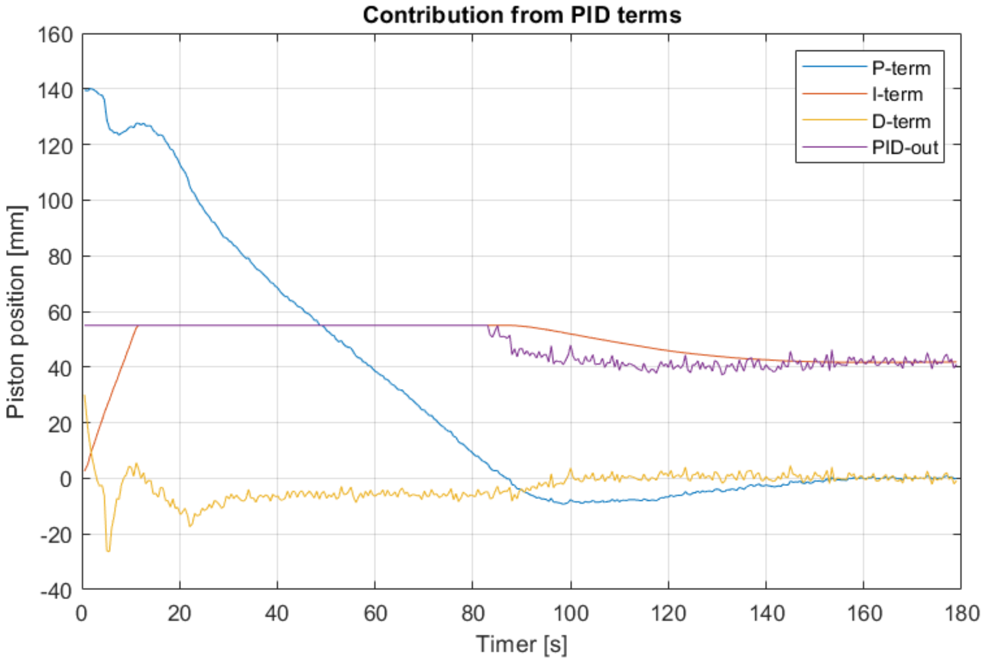


Figure 7.15: Contribution from proportional, integral and derivative terms in PID controller, during mission 1 in physical experiment #2.

7.4.5 Mission 2

The second mission involved the BV diving to a depth of 8 meters for 4 minutes, followed by floating up to a depth of 5 meters for 3 minutes. The step response and piston position are shown in Figure 7.16. The results were similar to the first mission, with the BV taking around 120 seconds to reach the first reference depth of 8 meters, slightly overshooting it, and then stabilizing with the same piston position of 40mm. The second reference depth of 5 meters was initiated after 240 seconds, with the piston being driven to the bottom position (0mm) to increase buoyancy. However, this caused the vehicle to overshoot the target depth by approximately 2 meters, and the piston was driven to saturation (51mm). The BV then sank to 5.4 meters before slowly stabilizing at 5 meters after 340 seconds, with the same piston position as before.

Figure 7.16 illustrates the measured and simulated step response and piston position for the second mission. It was observed that the simulated step response reached the reference depth of 8 meters after about 70 seconds, which was 50 seconds earlier than in the physical experiment. A possible explanation is that the friction between the rope and the line guide generated a force in the opposite direction of movement, which was not accounted for in the simulation. Otherwise, the simulation appeared to be reasonably accurate, with both the BV and the simulation stabilizing at a piston position of 40mm, where the BV was in equilibrium with seawater. Both the simulation and the physical BV overshot the second reference depth of 5 meters when floating upwards. This is attributed to the fact that the vehicle floats more easily than it sinks in the given presetting of outer lid and seawater density.

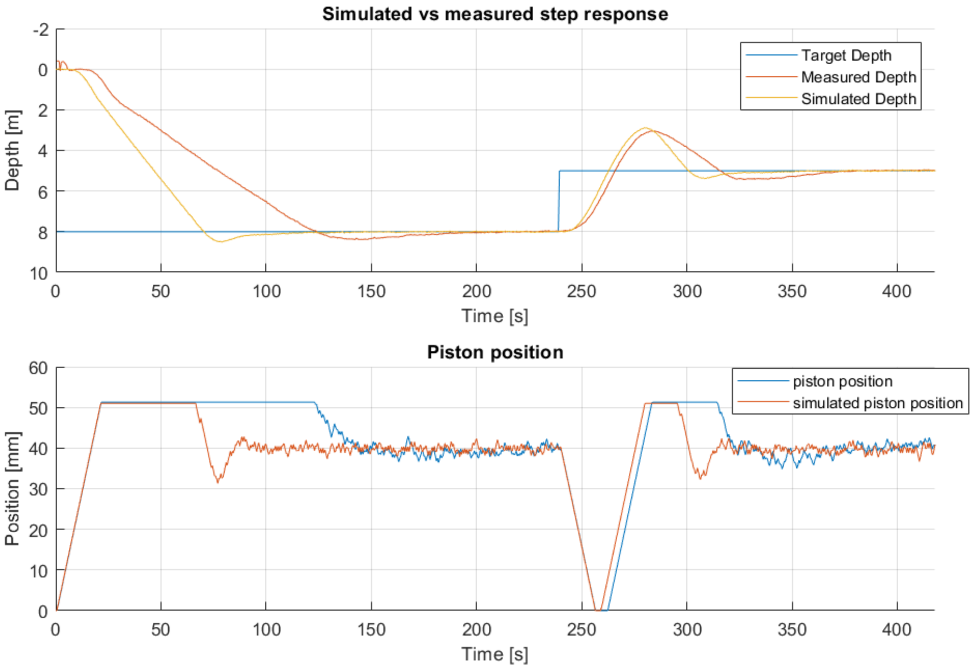


Figure 7.16: **TOP:** logged and simulated step response. **BOTTOM:** logged and simulated piston position

7.4.6 Temperature sensor

Figure 7.17 depicts a comparison between the measured temperature from the TMP117 sensor and the temperature obtained using a CTD probe. The CTD probe is known for its accuracy of 0.05°C and serves as a reference for the comparison. At the water surface, the CTD probe measures a temperature of 6.85°C, which gradually increases to 7.1°C at a depth of 2 meters. As the depth increases further, the temperature decreases to approximately 7°C at a depth of 4.5 meters, where it stabilizes until reaching the seabed at 12 meters.

In contrast, the TMP117 sensor initially measures a temperature of 8.75°C at the water surface. This temperature gradually decreases to 8.45°C at a depth of 8 meters. Eventually, the temperature reaches 8.15°C by the end of the mission. The observed temperature decrease over time aligns with the trend observed in the freshwater tank experiments. This pattern supports the hypothesis that the temperature of the pressure sensor's brass material cools down gradually as heat disperses through the entire brass structure over time.

Additionally, it is worth noting that the TMP117 sensor shows a deviation of approximately 1.3°C compared to the CTD probe measurements. A plausible explanation for this difference is that warmer air inside the buoyancy vehicle affects the TMP117 sensor, causing it to measure a higher value than the actual brass nut and water temperature.

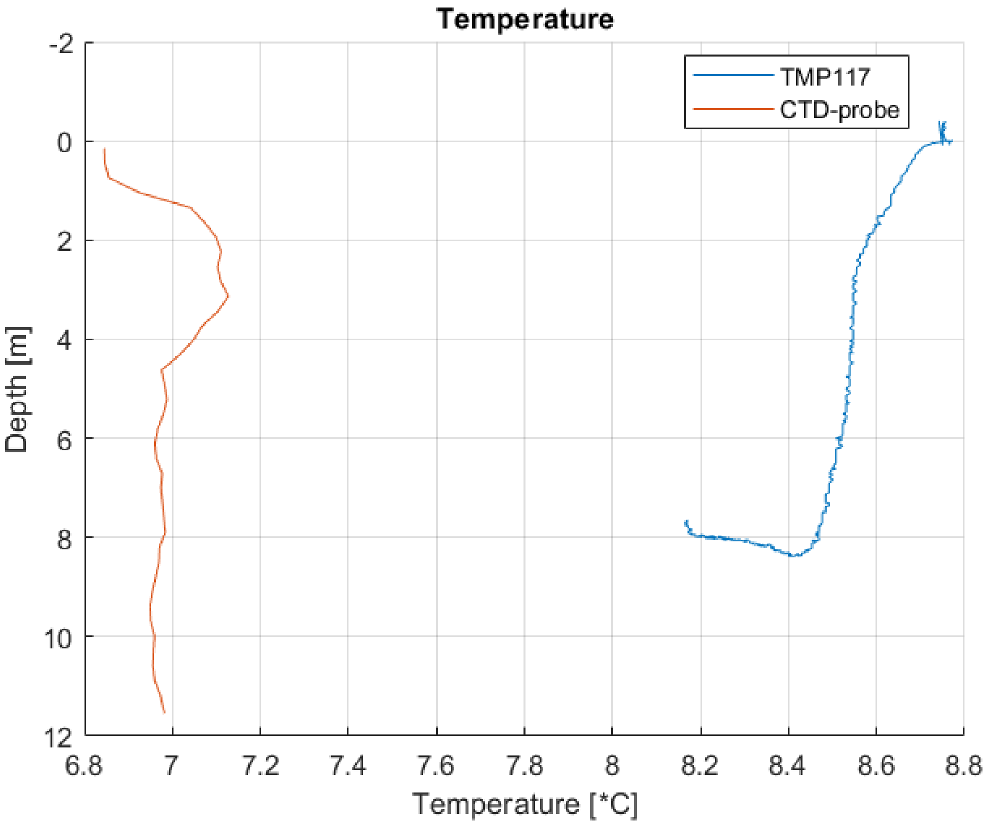


Figure 7.17: Logged temperature from TMP117 sensor and CTD- probe.

7.4.7 IMU

Figure 7.18 presents the logged tilt angle of the BV in the xy plane while submerged in seawater. The initial 25 seconds of the dive have been excluded as the vehicle was not fully submerged during this period. Overall, the tilt angle of the BV varies between -2 and 1 degrees in the y-axis and between 0 and 1 degree in the x-axis throughout the mission. However, there are instances where the tilt angle experiences peaks reaching -3 to 2 degrees in the y-axis and -2 to 3 degrees in the x-axis. It remains uncertain whether these peaks are a result of measurement noise or if the vehicle actually exhibits such movements.

It is noteworthy that the BV demonstrates more pronounced tilting in the outdoor seawater environment compared to the freshwater tank. This observation aligns with expectations, considering the influence of strong winds and increased water movement, which contribute to greater tilting and instability in the buoyancy vehicle.

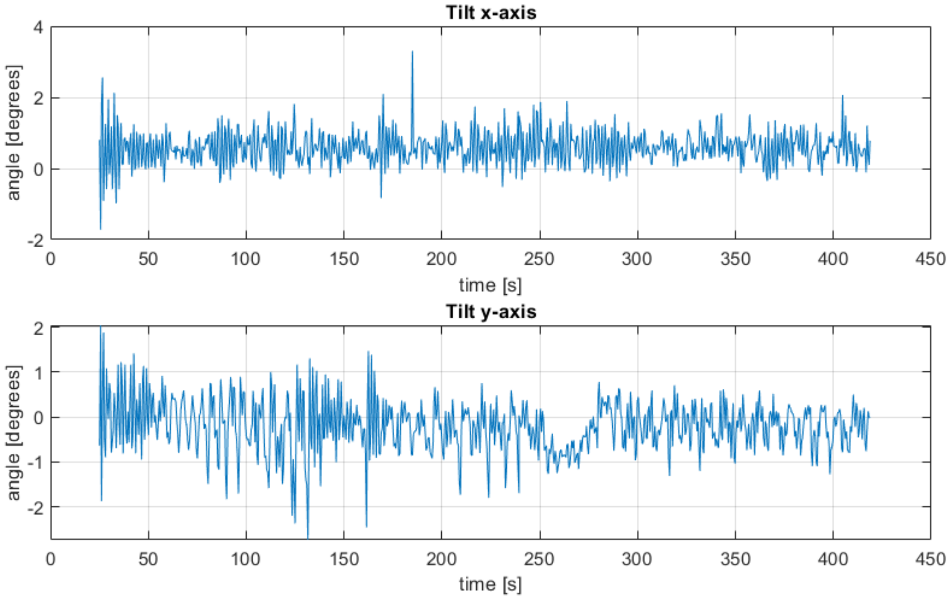


Figure 7.18: Tilting of the vehicle during a dive at Børsa

7.5 Physical experiment # 3

In this section, we will present the physical experiment conducted to evaluate the performance of the tracking system. The experiment took place at Brattora in Trondheim, Figure 7.19 shows a screenshot from Google Maps illustrating the experimental setup. Three nodes were positioned in a triangular configuration, with each node consisting of a hydrophone, SLIM module, and ARF868 radio modem. Due to the unavailability of a boat during the experiment, the locations of the nodes and fish tag were limited to the land and dock areas. The nodes were positioned to ensure free line of sight between them.

The experiment included two main tests. The first test involved dropping the fish tag at a known location and monitoring its position for a duration of 15 minutes. This allowed for an analysis of the distribution of the TDoA position calculations. The second test involved walking along a designated road while comparing the TDoA position estimations obtained from the tracking system.

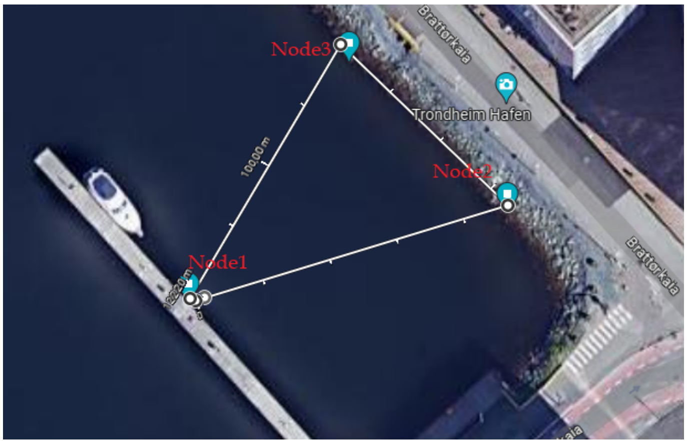


Figure 7.19: Experimental setup of nodes, during physical experiment #3

7.5.1 Test1 Fixed Point

The initial test involved dropping a tag into the water at a known position. To obtain the actual position of the tag, a GPS position from an iPhone 7 was used. The fish tag was placed at an arbitrary location between node2 and node3, ensuring free line of sight to all three hydrophones.

	Node1	Node2	Node3
Tag Detection	64	57	62

Table 7.4: Number of tag detection during fixed point experiment

Table 7.4 include the different nodes detection of the fish tag, out of these detections, 54 occurred within the same second, resulting in 54 timestamps available for TDoA calculations. Figure 7.20 illustrates the positions calculated using the TDoA method, alongside the actual position of the tag. It can be observed that the calculated positions are distributed around the actual tag position. It is worth noting that out of the 54 timestamps available for TDoA calculation, 43 resulted in numerical values, and 9 timestamps failed during the TDoA calculation process. Further analysis of the positional distribution revealed that the maximum error between the actual position and the TDoA position was 4.7 meters, with an average error of 1.6 meters.

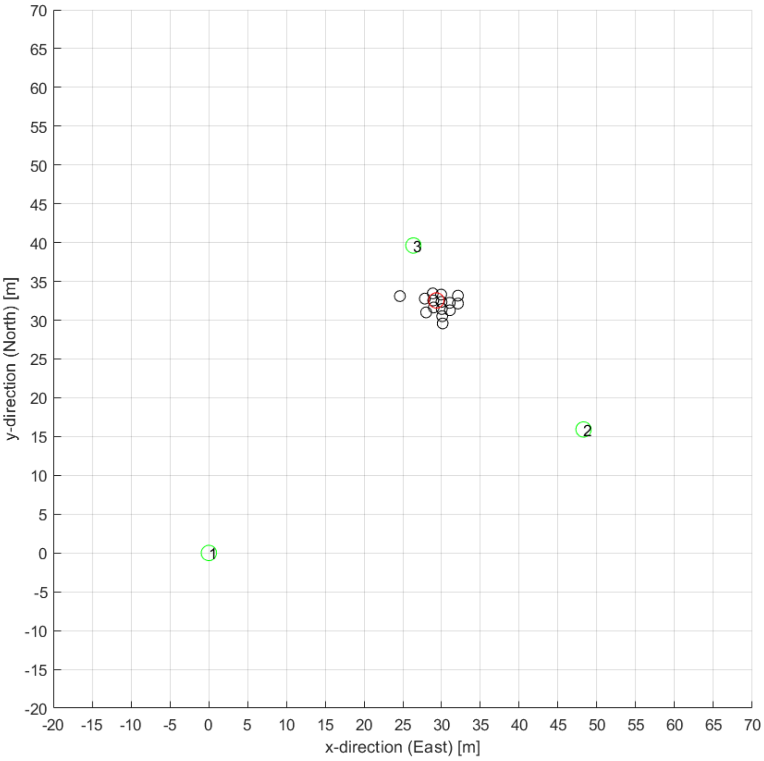


Figure 7.20: TDoA position of stationary tag. Green point illustrates Node 1, 2 and 3. Black point illustrates tag position, and red point illustrates the known position of the tag.

7.5.2 Test2 Walking Around the Area

The second test involved walking around the area between node3 and node2, followed by walking along the dock where node1 was located. Additionally, the fish tag was dropped into the water along the road between node2 and node1. It is important to note that TDoA positions are expected to have lower accuracy when the fish tag is located nearby one of the nodes, or when its located outside of the triangular shape formed by the three nodes.

Figure 7.21 provides a visual representation of the TDoA-calculated positions obtained while walking around the area. Notably, between node3 and node2, the calculated positions closely align with a straight line, indicating reasonable results and meeting the expected higher precision of the TDoA calculation in this region. However, when walking along the dock at node1, the presence of nearby parked boats limited the available path. Consequently, this area near node1 has lower precision in the TDoA calculation, which is reflected in the obtained positions. Although five positions were successfully calculated, it is generally more challenging to obtain accurate positional data due to low precision in this region. Lastly, when the tag was dropped along the road outside of the triangle, resulting in four detections. The distribution of the calculated positions displayed variability. This pattern aligns with the anticipated precision of the TDoA calculation in this location, where accuracy may be compromised due to the tag's position outside the primary coverage area.

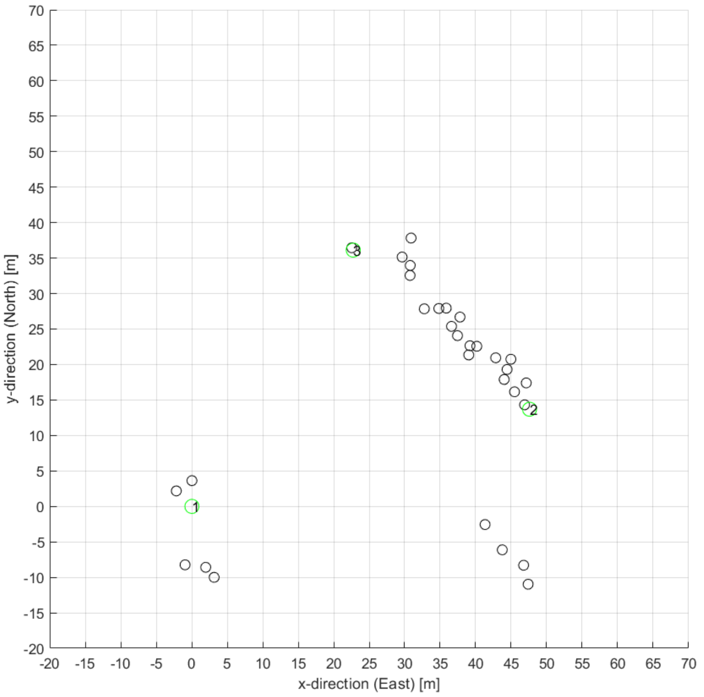


Figure 7.21: TDoA position when walking around the area.

7.6 Physical experiment # 4

In this section, we present the final physical experiment. It showcases the complete system, including the BV with fish tag, and real-time position tracking system. The experiment was carried out at the Børsa dock, located outside Trondheim. The harbor at Børsa is displayed in Figure 7.22a, a screenshot from Norgeskart.

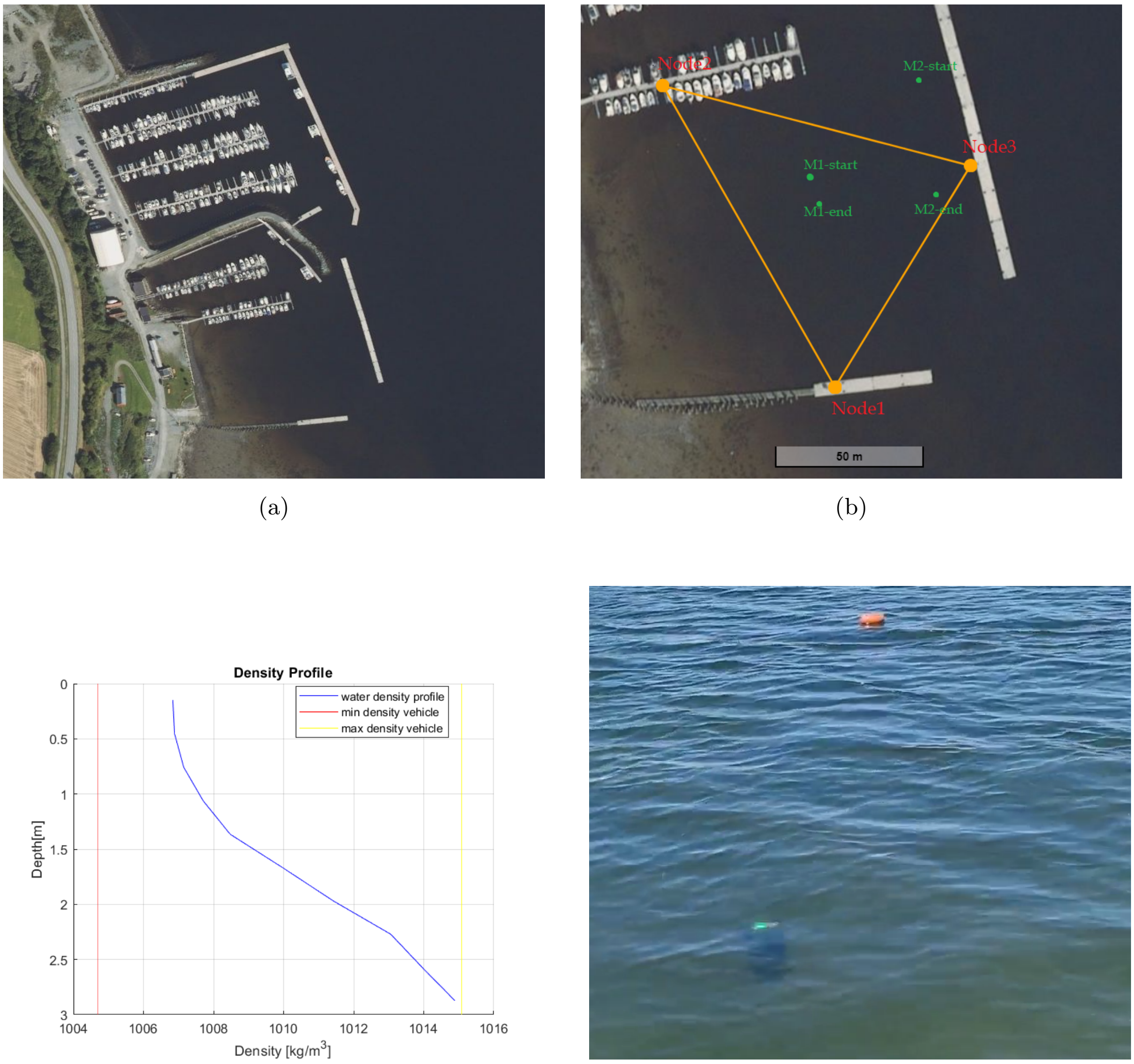
Figure 7.22b illustrates the placement of the nodes, as well as the locations for the BV's deployment and recovery during the two tests described in this section. The nodes were arranged in a triangular formation, according to the TDoA position method, for optimal line of sight the hydrophones where positioned 2 meters beneath the water's surface. The Børsa area offers stationary breakwaters around the dock, where two of the nodes were positioned, and a traffic-minimal open zone, making it an ideal location for deploying the vehicle. Figure 7.22d displays the BV during a dive. It was connected to a floater using a 6-meter weightless line, to both mark the BV's location and abort the dive in case the BV are drifting close to moorings.

Before the experiment, a sonar was used to measure the depth of the planned BV deployment area, which ranged from 2.5 to 3.5 meters. The density profile of the location is represented graphically in Figure 7.22c. This also demonstrates the maximum and minimum density of the vehicle after setting the outer lid to $h_{bot} = 92$mm. The density ranged from $1006.8\frac{kg}{m^3}$ at the surface to $1014.9\frac{kg}{m^3}$ at the 3-meter seabed. Notably, the low density, relative to general seawater, may be attributed to high surface temperature and low salinity, resulting from high melting and the entrance of freshwater from a nearby river. The first meter of depth exhibits minimal density variation, which then dramatically increases from 1 meter down to the seabed at 3 meters. The BV's outer lid was adjusted to maintain a minimum density of $1004.7\frac{kg}{m^3}$ and a maximum density of $1015\frac{kg}{m^3}$, ensuring the vehicle's ability to return to the surface.

The BV was fitted with a fish tag that transmits a data packet every 10 seconds, thereby achieving a positional resolution of 10-second intervals. These data packets also include depth measurements. BV configuration parameters are summarized in table 7.5

K_P	K_I	K_D	Integral Threshold	EMA alpha	h_{bot}
0.026	0.001	0.1	10 [m]	0.4	92 [mm]

Table 7.5: Vehicle configuration parameters, during physical experiment #4



(a)

(b)

(c)

(d)

Figure 7.22:
a) Overview at Børsa harbour
b) Experimental setup of nodes and approximate location of the starting and ending points for mission 1 and 2
c) Density profile at the location, compared to maximum/minimum density of the vehicle after setting the outer lid to a height of $h_{\text{bot}} = 92\text{mm}$.
d) BV during a dive at 0.5 meters, attached to a floater.

7.6.1 Test1

The initial test involved deploying the vehicle within the triangular arrangement of nodes, as illustrated in Figure 7.22b. This is an area where the TDoA positioning algorithm exhibits high precision. The vehicle was deployed with a target depth of 0.5 meters for a duration of 10 minutes. The raw position log is displayed in Figure 7.23a, and was plotted in real time during the dive. From the raw position log, it is apparent that the vehicle followed a southeast trajectory. This trajectory aligns with visual observations during the dive, supporting the results' credibility.

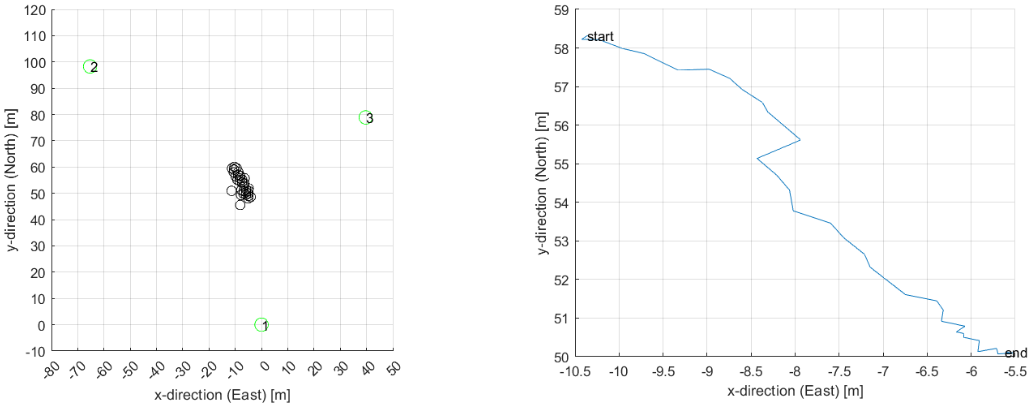
Figure 7.23b presents the BV's path during the dive. The raw position log was refined using a moving average filter, specifically the $movmean()$ function [28] in Matlab. Within the local Cartesian frame, the BV was deployed at P_1 = (-10,59) and concluded the dive at position P_2 = (-5.4, 48.5). By assuming the vehicle traveled a straight line, this corresponds to a horizontal distance of:

$$|P_1 - P_2| = \sqrt{(-10 - -5.4)^2 + (59 - 48.5)^2} = 11.5[m]$$

For a duration of 10 minutes, this results to an average speed of $1.9[\frac{cm}{s}]$.

Figure 7.24a portrays the BV's filtered path, with depth measurements provided by the fish tag. Furthermore, Figure 7.24b highlights the magnitude and direction of the velocity during the dive. It should also be mentioned that during the dive, a boat was used to observe the BV, and maneuvering the boat around the vehicle may have affected the BV's path.

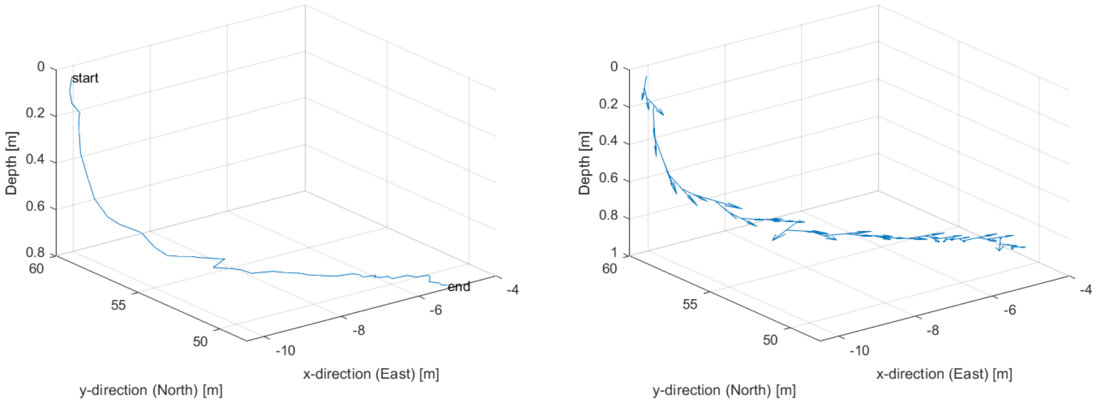
The logged depth and piston position of the vehicle during the dive are illustrated in Figure 7.25. The initial 10 seconds of the mission were allocated for lowering the vehicle into the water and inverting it to release air from the piston. The vehicle then ascended to the surface before stabilizing at a depth of 0.5 meters after about 150 seconds. At this point, minor deviations of up to 7cm from the target depth were observed, but for most of the dive, the vehicle maintained a neutral buoyancy at a depth of 0.5 meters. The BV attained equilibrium with the water at a piston position of h_{piston} = 22mm.



(a)

(b)

Figure 7.23: **a)** Raw position log, where nodes are indicated by green circles **b)** Filtered path of BV.



(a)

(b)

Figure 7.24: **a)** Filtered 3D position plot. **b)** Vector plot of velocity with magnitude and direction

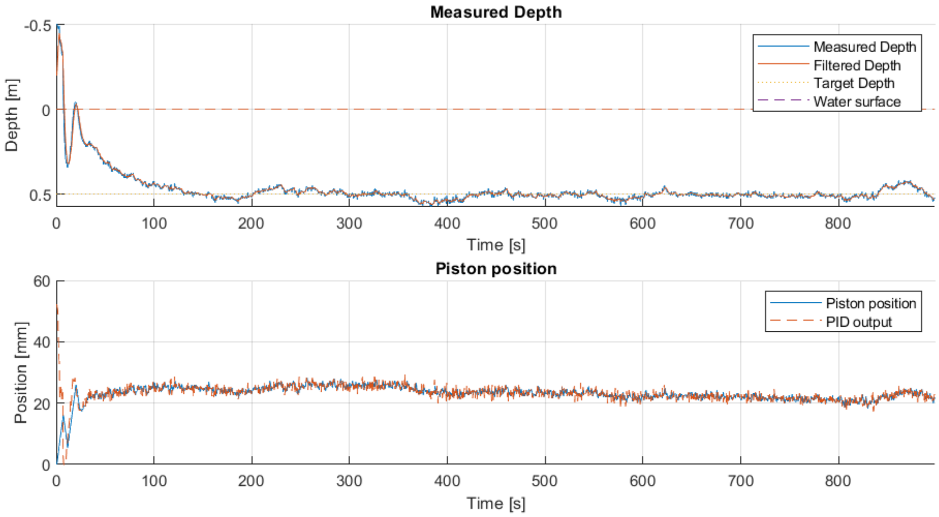


Figure 7.25: Logged depth and piston position of BV, during test#1

7.6.2 Test2

The second test involved deploying the vehicle to the north of node3, as shown in Figure 7.22b. This specific location was selected due to the higher surface current observed in this region. In areas outside of the node triangle, the TDoA positioning method is is known to have less precision.

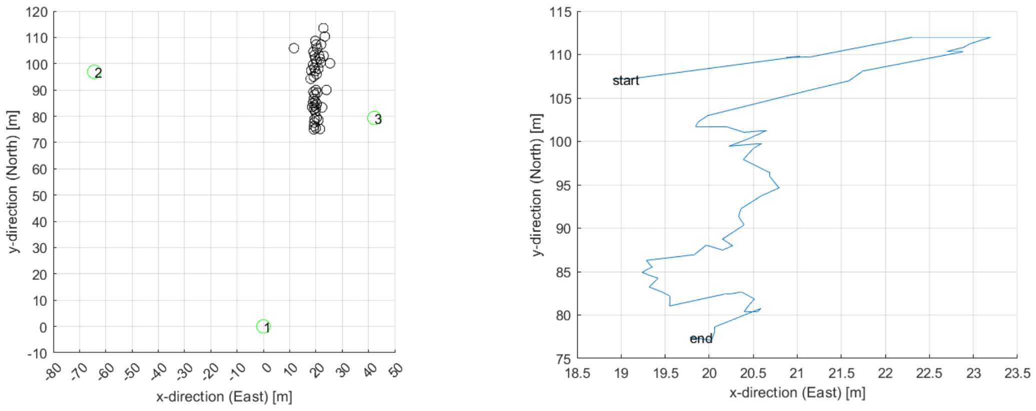
For this test, the vehicle was programmed to dive to a depth of 0.5 meters for 15 minutes, followed by a depth of 0.8 meters for 3 minutes. The raw position log is depicted in Figure 7.26a, where it is evident that the positions outside the triangle display greater variance. This observation aligns with the expected lower precision of the TDoA positioning method in this region. Despite this, most positions align along a southward line, which also aligns with the visual observations during the dive. The filtered position is displayed in Figure 7.26b, showing that the data indeed reflects the reduced positional accuracy.

The BV was deployed at position P_1=(19, 107) and completed the dive at position P_2=(20,77). Assuming the vehicle traveled along a straight path, the horizontal distance covered can be computed as:

$$|P_2 - P_1| = \sqrt{(19-20)^2 + (107-77)^2} = 30[m]$$

Given the duration of the dive as 18 minutes, the average velocity is calculated to be $2.7\frac{cm}{s}$, a significant increase compared to the first dive.

Figure 7.27 exhibits the logged depth of the BV. It shows that the vehicle remain neutrally near the target depths of 0.5 meters for 15 minutes and 0.8 meters for 3 minutes. At a depth of 0.5 meters, the vehicle reached equilibrium with the piston at $h_{piston} = 23.5$mm. For the second target depth of 0.8 meters, equilibrium was reached with $h_{piston} = 24.5$mm. This corresponds to the increased water density encountered with greater depth.



(a)

(b)

Figure 7.26: **a)** Raw position log, where nodes are indicated by green circles **b)** Filtered path of BV.

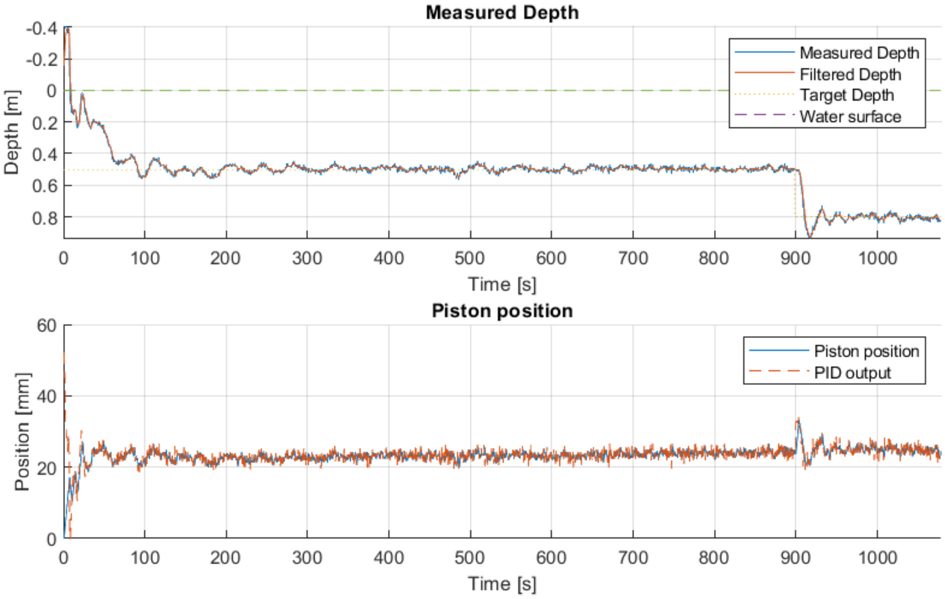


Figure 7.27: Logged depth and piston position of BV, during test#2

Chapter 8

Discussion and Conclusion

In this chapter we will discuss and conclude sub goals of the project and the functionality as a subsurface drifter. Finally, we will present future work.

8.1 Buoyancy vehicle

The BV has undergone extensive testing, including on-land hardware configuration checks and numerous dives in both freshwater tanks and seawater. Its performance matched the expectations outlined in the use case and functional test plan, as referenced in Appendix B. The Hall-effect button was successfully replaced, thereby dedicating the I2C bus line to the TMP117 and ICM20948 sensors. The drivers for these temperature and motion sensors were integrated, proving to be a valuable factor for post-dive analysis of the vehicle's behavior. Successful implementation of on-site calibration for the pressure sensor was achieved, enabling the vehicle to adapt to variations in atmospheric pressure.

Figure 8.1 presents a screenshot from Segger Embedded Studio, showcasing the utilization of SoC flash and RAM. The used Flash amounts to 330.5 Kb out of a total of 512Kb (64.5%), while used RAM are 33.7Kb out of 64Kb (52.6%). The available capacity indicates potential for further development, without necessitating a shift to a new module with superior specifications.

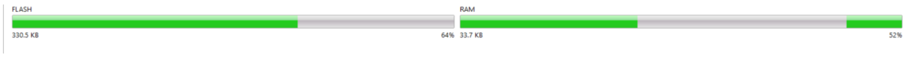


Figure 8.1: SoC flash and RAM usage. Flash: 64% RAM: 52%

8.1.1 Battery consumption

Experiment #	U_{start} [V]	U_{end} [V]	Duration [h]
1	22.6	20.9	5

Table 8.1: Battery voltage measured in sleep mode

Table 8.1 presents the measured battery voltage using an external voltmeter when the vehicles are in sleep mode. Assuming that the voltage level decreases linearly over time, an estimate of the battery life can be determined. By calculating the rate of voltage decrease, we find:

$$\frac{22.6 - 20.9}{5} = 0.34 \frac{V}{\text{hour}}$$

Based on this rate, the sleep time can be estimated using the formula:

$$t_{\text{sleep}} = \frac{24 - 12.8}{0.34} \approx 33 \text{ hours}$$

These findings indicate that the vehicles are not entering a fully sleep mode, as the motor driver and other sensors continue to drain the battery unnecessarily. To prevent this unnecessary drain, it is recommended to disconnect the battery when the vehicles are not in use.

$U_{start}[V]$	$U_{end}[V]$	Duration [h]
22.1	20.9	0.5

Table 8.2: Battery voltage measured before and after a dive of 30 minutes

Table 8.2 present the measured battery voltage using an external voltmeter, before and after a dive of 30 minutes. Again assuming linear decrease of battery voltage, the rate of voltage decrease are found:

$$\frac{22.1 - 20.9}{0.5} = 2.4\frac{V}{hour}$$

The estimated lifetime of the battery package during diving is then found:

$$\frac{24 - 12.8}{2.4} \approx 5 \text{ hours}$$

Regarding battery consumption during a dive, it should be mentioned that the stepper motor is programmed to cut off the current 500ms after the establishment of a new position. Due to the noise interference in pressure sensor readings, the stepper motor receives a fresh positional value even when the buoyancy vehicle maintains stability at the reference depth. Consequently, the piston undergoes slight oscillations, moving up and down within a range of 1-2 mm after reaching the reference depth. This causes unwarranted battery drain, accounting for a considerable proportion of the overall battery usage.

8.1.2 Vehicle density control

An in-depth analysis of the vehicle's operational capacity in relation to seawater density had been done. It has been established that the vehicle is capable of modifying its own density roughly by $10\frac{kg}{m^3}$. The vehicle's diving capabilities depend on the location-specific salinity and temperature. In most scenarios, the vehicle faces no constraints in terms of altering its density beyond the maximum seawater density, thereby enabling it to sink to the intended depth and return to the surface. Nonetheless, when encountering low-density freshwater at the surface, the water's density increases with depth and the vehicle reaches an equilibrium with the maximum density it can achieve. Consequently, the vehicle can't dive any deeper. As depicted in Figure 4.5 in Chapter 4, the vehicle's diving limit is restricted to 14.5 meters under these circumstances. Increasing the BV capacity to manipulate its density could be achieved by extending the piston displacement or increasing piston diameter. These modifications require a redesign of the linear actuator of the BV.

To fine-tune the vehicle's configuration for a specific target location, it is essential to have a comprehensive understanding of the location's density profile. Even minor adjustments to the outer lid can significantly impact the vehicle's performance. During this project, a CTD-probe, specifically the SonTek CastAway-CTD, was used to determine the density, yielding satisfactory results.

8.2 Simulation

The mathematical simulation of the vehicle's behavior evolved from the specialization project and is designed to represent the physical BV along with the implemented PID regulator and the environmental factors at a specific location. The simulation has proven to be an effective tool for evaluating the BV at a new location. It interprets the density profile at the location, facilitating rapid on-site evaluation of the vehicle's limitations as well as computation of the preset value of h_{bot} for the outer lid.

The simulated performance of the vehicle aligns closely with the physical experiments. As described by the freshwater tank experiment in Chapter 7.3, Figure 7.7 illustrates the measured and simulated responses. It's reasonable to conclude that the mathematical simulation accurately represents the physical BV. Minor deviations in the simulated response are expected and considered acceptable, as mathematical models inevitably have limitations in fully encapsulating physical environments and system responses.

It's worth mentioning that the simulation and the physical BV use different presetting of the outer lid to accurately align with the water's equilibrium point at the same piston position. For instance, while the simulation uses a presetting of $h_{bot} = 95$, the physical BV has a presetting of $h_{bot} = 93.5$. This difference of 1.5mm arises from inaccuracies in measuring the vehicle's mass and dimensions for volume calculation.

8.3 Tracking system

The tracking system underwent testing using a fish tag placed at known locations, allowing for a comparison with the calculated positions. In the fixed point test described in section 7.5.1, it was determined that the average error between the actual position and the calculated position was approximately 1.6 meters. This closely aligns with the fact that the time synchronization of nodes exhibited a deviation of 1 millisecond, which corresponds to a distance of 1.5 meters considering the speed of signal travel. Additionally, it was observed that the maximum error reached almost 5 meters.

The results of the testing are considered promising, indicating that the tracking system is capable of providing reasonably accurate position estimations. However, it is important to note that when tracking the BV over a period of time, a deviation of up to 5 meters should be taken into account.

8.4 Physical experiments

The vehicle has been submitted to extensive testing both in fresh water tank and seawater. The PID controller has been successfully tuned, enabling the vehicle to dive to a desired depth and remain neutral during the dive. The contribution of PID terms has been extensively analysed, and the behavior of the vehicle yield promising results. It has been discovered that the derivative term are affected by noise in the depth measurements, this noise does not affect the performance itself, as the vehicle behaves stable at the desired depth. However, the noise makes new PID calculations, and give new commands to the motor control module to move the piston 1-2 mm every 500ms, which cause unnecessary battery drain, for longer dives.

During a dive, the IMU sensor has reveled that the vehicle behave stable in the surrounding water, enabling the vehicle to behave as a "package of water". The TMP117 temperature sensor has been compared to the CTD probe, where it was found that the sensors deviated approximately 1.5 °C.

8.5 Conclusion

The testing of the BV and tracking system has yielded promising results, particularly in the context of collecting subsea ocean current data. The ability of the BV to maintain neutral buoyancy at a predetermined depth proves to be a valuable asset, allowing for effective data collection and analysis at various depths within fish farmes. This capability provides the ability to gather information about subsea currents and their characteristics.

The real-time tracking system integrated with the BV proves to be an useful tool for monitoring the vehicle's position during dives in the open sea and for post dive analysis. This tracking system enables for tracking the BV movements in real time, ensuring its safety and allowing for monitoring positioning throughout the dive.

The final test of BV and tracking system combined demonstrates the functionality of the system to estimate subsea current flow. Where direction and magnitude of the velocity are estimated. It should be mentioned that the moving average filter of the positional data is not optimally tuned, more accurate representation of the data could be achieved by tuning the filter, or by using more advanced techniques. The fish tag are tailor made with defined ID#, and minor re-factorisation of the Matlab GUI, to seperate data based on the tag ID would enable for tracking multiple BV at the same time, with the same tracking system.

To summarize, a prototype BV has been made and the proof of concept are verified through physical experiments. The BV are able to handle an additional mass of 0.5kg by presetting the outer lid, and micro-controller's capacity of Flash and RAM, enables for further development of the BV, by implementing more sensors. Extending the functionality of the BV.

8.6 Future work

The experimental trials of the buoyancy vehicle and its tracing system have yielded encouraging results. Yet, there is opportunity for improvement, with particular focus on power usage and strategies for an efficient collection of subsurface current data.

- A comprehensive investigation centered around power usage should be conducted to characterize power consumption in various modes and optimize the nRF52832 SoC's power demands.
- The LED's power usage can be reduced by integrating a duty cycle, which would essentially activate the LED for brief intervals and otherwise keep it switched off.
- Power demands of the stepper motor can be minimized by enabling it to cut off current, as discussed in section 8.1.1. This could be achieved by deploying a more advanced filter of the pressure sensor, to eliminate redundant set points of the piston when the vehicle maintains stability at the reference depth. Alternatively, logic could be introduced to switch the PID regulator off when the vehicle is stable at the reference depth and activate it when required.
- Power consumption in sleep mode should be optimized, disconnect motor interface and put external sensors to sleep.
- Implementation of filtering the accelerometer data, to more accurately evaluate the tilted angle of the BV during a mission.
- The ICM20948 module should be configured to accumulate compass data, a feature that would be useful in post dive evaluation of the BV rotation.
- Accurate measurements and computations of the BV mass and volume should be performed, and the Matlab model should be updated for more precise presetting of the outer lid and simulation.
- Fine tuning of moving average filter of positional data.
- Develop and implement a strategic plan related to the collection of subsurface current data, as well as visualization of this data.

Bibliography

[1] Scripps Institution of Oceanography. *SURFACE VELOCITY PROGRAM BAROMETER (SVPB) DRIFTER.* URL: https://gdp.ucsd.edu/ldl/svpb/ (visited on 30/05/2023).

[2] Stian Børseth. 'Low Energy Buoyancy Actuator for Vertical Underwater Motion'. In: (2017).

[3] Odd Inge Halsos. 'Embedded System for Underwater Data Acquisition'. In: (2020).

[4] UNIVERSITY PHYSICS VOLUME 1. *Archimedes' Principle and Buoyancy.* URL: https://pressbooks.bccampus.ca/universityphysicssandbox/chapter/archimedes-principle-and-buoyancy/ (visited on 09/03/2023).

[5] Scott Zhuge. *PID control theory.* URL: https://www.crystalinstruments.com/blog/2020/8/23/pid-control-theory (visited on 13/02/2023).

[6] Geoffrey Hubter. *Exponential moving average (EMA) filters.* URL: https://blog.mbedded.ninja/programming/signal-processing/digital-filters/exponential-moving-average-ema-filter/ (visited on 09/03/2023).

[7] Paul Webb. *INTRODUCTION TO OCEANOGRAPHY.* URL: https://rwu.pressbooks.pub/webboceanography/chapter/6-3-density/ (visited on 10/03/2023).

[8] David Bice. *Modeling Thermohaline Circulation.* URL: https://serc.carleton.edu/integrate/teaching_materials/earth_modeling/student_materials/unit8_article1.html (visited on 10/03/2023).

[9] S.R. Massel. *Sea Water Density According to UNESCO Formula.* URL: https://link.springer.com/content/pdf/bbm:978-3-319-18908-6/1.pdf (visited on 10/03/2023).

[10] Bertrand T. Fang. 'Simple solutions for hyperbolic and related position fixes'. In: *IEEE TRANSACTIONS ON AEROSPACE AND ELECTRONIC SYSTEMS* 26 (1990), pp. 748–753.

[11] MathWorks. *Object Tracking Using Time Difference of Arrival (TDOA).* URL: https://se.mathworks.com/help/fusion/ug/object-tracking-using-time-difference-of-arrival.html (visited on 10/05/2023).

[12] Littelfuse. *Hall Effect Sensor Datasheet.* URL: https://www.elfadistrelec.no/Web/Downloads/_t/ds/Littelfuse_55100_eng_tds.pdf (visited on 11/05/2023).

[13] Honneywell. *PX3-series.* URL: https://prod-edam.honeywell.com/content/dam/honeywell-edam/sps/siot/en-us/products/sensors/pressure-sensors/industrial-pressure-sensors/px3-series/documents/sps-siot-heavy-duty-pressure-px3-series-datasheet-32313757-ciid-159273.pdf?download=false (visited on 11/05/2023).

[14] Trinamic. URL: https://www.trinamic.com/products/drives/details/pd60-3-1161/ (visited on 11/05/2023).

[15] Omron. *Ultra Slim Snap-action Switch with 2.7 mm in depth.* URL: https://omronfs.omron.com/en_US/ecb/products/pdf/en-d2mq.pdf (visited on 11/05/2023).

[16] InvenSense. *World's Lowest Power 9-Axis MEMS MotionTracking™ Device.* URL: https://invensense.tdk.com/wp-content/uploads/2016/06/DS-000189-ICM-20948-v1.3.pdf (visited on 11/05/2023).

[17] core-electronic. *ICM20948 9DoF Motion Sensor Breakout.* URL: https://core-electronics.com.au/icm20948-9dof-motion-sensor-breakout.html (visited on 11/05/2023).

[18] Texas instrument. *TMP117 high-accuracy, low-power, digital temperature sensor with SMBus™- and I2C-compatible interface.* URL: https://cdn.sparkfun.com/assets/9/e/5/1/8/tmp117.pdf (visited on 08/12/2023).

[19] Sparkfun. *PicoBuck Hookup Guide V12.* URL: https://media.digikey.com/pdf/Data%5C%20Sheets/Sparkfun%5C%20PDFs/PicoBuck_HookupGuide_V12_Web.pdf (visited on 11/05/2023).

[20] Unknown. *iPixel LED light.* URL: https://cdn.sparkfun.com/assets/e/c/9/5/d/M018001MA3LZ.pdf (visited on 11/05/2023).

[21] Nordic Semiconductior. *TWI master.* URL: https://infocenter.nordicsemi.com/index.jsp?topic=%5C%2Fcom.nordic.infocenter.nrf52832.ps.v1.1%5C%2Ftwim.html (visited on 15/05/2023).

[22] Rubén Santa Anna. *PID controller algorithm C.* URL: PID%20controller%20algorithm%20C (visited on 15/05/2023).

[23] noble git repository. *noble.* URL: https://github.com/noble/node-bluetooth-hci-socket#windows (visited on 15/05/2023).

[24] thelmaBiotel. *downloads.* URL: https://www.thelmabiotel.com/downloads/ (visited on 10/05/2023).

[25] thelmabiotel. *TBR700.* URL: https://www.thelmabiotel.com/receivers/tbr-700/ (visited on 10/05/2023).

[26] Ublox. *u-blox M8 concurrent GNSS modules.* URL: https://content.u-blox.com/sites/default/files/NEO-M8-FW3_DataSheet_UBX-15031086.pdf (visited on 11/05/2023).

[27] Adeunis. *ARF868 Radio Modems.* URL: https://www.adeunis.com/wp-content/uploads/2019/09/ARF868_ARF794xx_RADIO_MODEMS_UG_V2.2.3_FR-GB.pdf (visited on 03/05/2023).

[28] MathWorks. *movmean.* URL: https://se.mathworks.com/help/matlab/ref/movmean.html (visited on 18/06/2023).

Appendix

Appendix A

UNESCO formula

This appendix include the full calculations of the UNESCO formula for calculating seawater density. The appendix is reproduced from [9]

Appendix A
Sea Water Density According to UNESCO Formula

Density of sea water for a given temperature T in the range $0 < T < 40\,^{\circ}\mathrm{C}$, salinity S in the range $0 < S < 42\,\mathrm{PSU}$ and pressure p is determined from the equation of state (UNESCO 1981):

$$\rho = \frac{\rho(S,T,0)}{1 - \dfrac{p}{K(S,T,p)}} \tag{A.1}$$

where $K(T, S, p)$ is module of sea water compressibility.

Sea water density is determined using the following algorithm:

- calculation of the SMOW density (*Standard Mean Ocean Water*):

$$\rho_{SMOW} = a_0 + a_1 T + a_2 T^2 + a_3 T^3 + a_4 T^4 + a_5 T^5 \tag{A.2}$$

where:

$$\left.\begin{aligned} a_0 &= 999.842\,594 \\ a_1 &= 6.793\,953 \times 10^{-2} \\ a_2 &= -9.095\,290 \times 10^{-3} \\ a_3 &= 1.001\,685 \times 10^{-4} \\ a_4 &= -1.120\,083 \times 10^{-6} \\ a_5 &= 6.536\,332 \times 10^{-9} \end{aligned}\right\} \tag{A.3}$$

- calculation of sea water density at the normal atmospheric pressure (*normal atmosphere*) ($p = 0$):

$$\rho(S,T,0) = \rho_{SMOW} + B_1 S + C_1 S^{1.5} + d_0\, S^2 \tag{A.4}$$

with:

$$B_1 = b_0 + b_1 T + b_2 T^2 + b_3 T^3 + b_4 T^4 \tag{A.5}$$

© Springer International Publishing Switzerland 2015

S.R. Massel, *Internal Gravity Waves in the Shallow Seas*,
GeoPlanet: Earth and Planetary Sciences, DOI 10.1007/978-3-319-18908-6

$$C_1 = c_0 + c_1 T + c_2 T^2 \tag{A.6}$$

where:

$$\left.\begin{aligned} b_0 &= 8.2449 \times 10^{-1} \\ b_1 &= -4.0899 \times 10^{-3} \\ b_2 &= 7.6438 \times 10^{-5} \\ b_3 &= -8.2467 \times 10^{-7} \\ b_4 &= 5.3875 \times 10^{-9} \end{aligned}\right\} \tag{A.7}$$

$$\left.\begin{aligned} c_0 &= -5.7246 \times 10^{-3} \\ c_1 &= 1.0227 \times 10^{-4} \\ c_2 &= -1.6546 \times 10^{-6} \\ d_0 &= 4.8314 \times 10^{-4} \end{aligned}\right\} \tag{A.8}$$

- determination of the compressibility module at pressure $p = 0$:

$$K(S, T, 0) = K_w + F_1 S + G_1 S^{1.5} \tag{A.9}$$

coefficients K_w, F_1 and G_1 are given by:

$$K_w = e_0 + e_1 T + e_2 T^2 + e_3 T^3 + e_4 T^4 \tag{A.10}$$

where:

$$\left.\begin{aligned} e_0 &= 19\,652.210\,000 \\ e_1 &= 148.420\,600 \\ e_2 &= -2.327\,105 \\ e_3 &= 1.360\,477 \times 10^{-2} \\ e_4 &= -5.155\,288 \times 10^{-5} \end{aligned}\right\} \tag{A.11}$$

$$F_1 = f_0 + f_1 T + f_2 T^2 + f_3 T^3 \tag{A.12}$$

where:

$$\left.\begin{aligned} f_0 &= 54.674\,600 \\ f_1 &= -0.603\,459 \\ f_2 &= 1.099\,870 \times 10^{-2} \\ f_3 &= -6.167\,000 \times 10^{-5} \end{aligned}\right\} \tag{A.13}$$

and

$$G_1 = g_0 + g_1 T + g_2 T^2 \tag{A.14}$$

where:

$$\left.\begin{aligned} g_0 &= 7.9440 \times 10^{-2} \\ g_1 &= 1.6483 \times 10^{-2} \\ g_2 &= -5.3009 \times 10^{-4} \end{aligned}\right\} \tag{A.15}$$

- determination of final compressibility module of the sea water:

$$K(S, T, p) = K(S, T, 0) + A_1 p + B_2 p^2 \tag{A.16}$$

with:

$$A_1 = A_w + (i_0 + i_1 T + i_2 T^2)\, S + j_0\, S^{1.5} \tag{A.17}$$

$$A_w = h_0 + h_1 T + h_2 T^2 + h_3 T^3 \tag{A.18}$$

where:

$$\left.\begin{aligned} h_0 &= 3.239\,90 \\ h_1 &= 1.437\,13 \times 10^{-3} \\ h_2 &= 1.160\,92 \times 10^{-4} \\ h_3 &= -5.779\,05 \times 10^{-7} \end{aligned}\right\} \tag{A.19}$$

$$\left.\begin{aligned} i_0 &= 2.283\,80 \times 10^{-3} \\ i_1 &= -1.098\,10 \times 10^{-5} \\ i_2 &= -1.607\,80 \times 10^{-6} \\ j_0 &= 1.910\,75 \times 10^{-4} \end{aligned}\right\} \tag{A.20}$$

and

$$B_2 = B_w + (m_0 + m_1 T + m_2 T^2)\, S \tag{A.21}$$

where:

$$B_w = k_0 + k_1 T + k_2 T^2 \tag{A.22}$$

with:

$$\left.\begin{aligned} k_0 &= 8.509\,35 \times 10^{-5} \\ k_1 &= -6.122\,93 \times 10^{-6} \\ k_2 &= 5.278\,70 \times 10^{-8} \end{aligned}\right\} \tag{A.23}$$

$$\left.\begin{aligned} m_0 &= -9.9348 \times 10^{-7} \\ m_1 &= 2.0816 \times 10^{-8} \\ m_2 &= 9.1697 \times 10^{-10} \end{aligned}\right\} \tag{A.24}$$

- final determination of the sea water density $\rho(S, T, p)$ from Eq. (A.1):

Example 1: salinity $S = 8\,\mathrm{PSU}$, temperature $T = 10\,°\mathrm{C}$, pressure $p = 0\,\mathrm{bar}$. As the pressure $p = 0$, the density $\rho(S, T, 0)$ is given by Eq. (A.4). Thus we have:

- $\rho_{SMOW} = 999.702\,09$ from Eq. (A.2),
- $\rho_w(8, 10, 0) = 1005.946\,59$ from Eq. (A.4).

Example 2: salinity $S = 8\,\mathrm{PSU}$, temperature $T = 10\,°\mathrm{C}$, pressure $p = 10\,\mathrm{bar}$. Similarly to example above we obtain:

- $\rho_{SMOW} = 999.702\,09$ from Eq. (A.2),
- $\rho(S, T, 0) = 1005.946\,59$ from Eq. (A.4),
- $K_w = 20\,916.794\,90$ from Eq. (A.10),
- $A_1 = 3.285\,74$ from Eq. (A.17),
- $B_2 = 0.000\,20$ from Eq. (A.21),
- $K = 21\,351.408\,20$ from Eq. (A.16),
- $\rho(8, 10, 10) = 1006.417\,97$ from Eq. (A.1).

Reference

UNESCO (1981) Tenth report of the joint panel on oceanographic tables and standards. UNESCO Technical Papers in Marine Science, Paris, 25 p

Appendix B

Buoyancy Vehicle

This appendix presents the use case, system specifications, and functional test plan for the BV. The content of this appendix draws inspiration from the work conducted by Halsos [3], as well as previous work carried out in the specialization project.

Buoyancy vehicle User Case

Overview.

The user case includes the following steps.

- Presetting of vehicle volume.
- Power-On
- Configuration mode
- Mission Deployment
- Pick up Mode
- Power-Off

Exceptions:

- Low power mode
- Error mode

Case: Presetting vehicle volume

1.Brief Description

Setting the maximum volume of the vehicle by adjusting the outer lid. A special tool is needed to rotate the outer lid and to hold the vehicle.

2.Actors

- User

3.Pre-Conditions

- Battery is connected.
- LED house is removed, for air to escape and avoid pressure inside the vehicle.
- Vehicle on land (dry environment).
- Known density profile of water, and height from vehicle bottom to outer lid is known.

4.Basic Flow

- Outer lid is rotated to calculated height from bottom
- Special tool is used to rotate outer lid
- Outer lid is self locked, by tight sealing ring.
- LED house is mounted with sealing.

5.Alternate/Exception Flows

- None

6.Post Conditions.

- Maximum volume of vehicle is set.
- Vehicle in IDLE (sleep state) or in standby mode.

Case: Power-On

1.Brief Description

Power on system from IDLE (sleep state), power on system by connection batteries to the system or resetting the MCU with the “reset” button.

Starts when: Hall effect button is activated, battery is connected or when system is reset with “reset button”.

Ends when: Vehicle is initiated and in stand by mode.

2.Actors

- User
- MCU/system

3.Pre-Conditions

- Battery has more than 12V voltage supply.
- Vehicle located on land.
- LED off

4.Basic Flow

1. System in sleep mode.
2. Hall effect “button” is activated with a magnet.
3. MCU/system is powered by batteries.
4. Initialization code is executed.
5. Piston is moved upwards for 2 seconds.
6. Piston is moved downwards.
7. Piston stops when lower limit switch is activated, and the position is stored.
8. Vehicle in standby mode.
9. Standby mode is indicated by white LED light.

5.Alternate/Exception Flows

Number of alternative or exception flows are indicated at which the variation/exception occurs. For example, a variation at step 2 is indicated as 2a and second variation is indicated as 2b.

1a system is disconnected from battery source. Then step 2 is ignored

1b system is reset with “reset button” then step 2 is ignored.

3a batteries are below 12V and low power mode activated.

6.Post Conditions.

- The system is powered on and in standby mode (indicated by white LED light)
- The necessary drivers are initiated
- Piston position at bottom position (maximum buoyancy).

Case: Configuration

1.Brief Description

The vehicle goes from standby mode to configuration.

Start when: Hall effect button is activated.

Ends when: Mode is changed to deploy mode or standby mode.

2.Actors

- User
- MCU/system
- Bluetooth protocol
- Memory
- Phone/computer with Bluetooth.

3.Pre-Conditions

- The system is initiated.
- The system is in standby mode.
- Sufficient batter power.
- Vehicle on land.

4.Basic Flow

1. Hall effect button is activated and configure mode is selected.
2. Config mode is indicated by blue LED light.
3. Vehicle broadcast Bluetooth.
4. Phone/computer establish Bluetooth connection with the vehicle.
5. Configuration data is sent from vehicle to phone/computer.
6. New configuration data is sent from phone/computer by user.
7. Configuration data is stored by the vehicle.

5.Alternate/Exception Flows

1a Hall effect button is not activated within 2 minutes and vehicle go back to sleep state.

4a vehicle not able to connect with computer.

5a mission data is requested and sent through Bluetooth.

6.Post Conditions.

- The system has received new configuration settings (PID tuning parameters).
- The system has received new mission data (time, depth and number of missions).
- The system remain in configuration mode, or has returned to standby mode.

Exception:

- Low power mode.

Case: Mission Deployment

1.Brief Description

Vehicle goes from configuration mode to start mission.

Starts when: Deploy mode is selected by user

2.Actors

- User
- MCU/system

3.Pre-Conditions

- System is in configuration mode.
- System is configured and ready for mission.
- Vehicle in water.

4.Basic Flow

1. Mission mode selected through Bluetooth.
2. Indicated with green LED light.
3. Mission data storage is executed.
4. Pressure sensor is calibrated at known position (floating at surface).
5. Mission algorithm is executed, and mission data is stored in SD- card.
6. When mission is completed, the piston is driven to bottom, and the vehicle float to surface.
7. Vehicle in pick-up mode, indicated by yellow LED light.

5.Alternate/Exception Flows

1a Error is detected, and mission is aborted.

5a Sensor reding error

6.Post Conditions.

Successful mission:

- Data is stored in SD- card.
- The vehicle floats to surface.
- Vehicle in pick-up mode.

Exceptions:

- Error (data storage, sensor reading, state etc) is stored in SD- card.
- Wrong presetting of vehicle volume:
  - Vehicle left floating at surface
  - Vehicle sinking at maximum volume, worst case scenario is losing the vehicle.
  - Bad performance of controller.
- Low power mode.

Case: Pick-up state

1.Brief Description

Vehicle goes from deployment mode to pick-up mode when the mission is finished.

Starts when: mission is finished

Ends when: configure state or stand-by is selected through Bluetooth.

2.Actors

- User
- MCU/system

3.Pre-Conditions

- System finished with a mission.
- Vehicle is floating on the water surface, or taken on land.
- System is powered on.

4.Basic Flow

1. Pick-up mode is selected with software.
2. Yellow LED light.
3. Bluetooth is broadcasting.
4. Bluetooth connection with phone/computer.
5. Set vehicle in configure state or stand by, with Bluetooth.

5.Alternate/Exception Flows

1a. Error is detected, and and error state is activated.

1b. Low powe mode.

6.Post Conditions.

- New mode is selected by Bluetooth.

Case: Low Power state

1.Brief Description

Battery voltage drop below 12v, and low power mode is activated.

2.Actors

- MCU/System

3.Pre-Conditions

- Battery connected
- Necessary drivers are initiated.

4.Basic Flow

- LED light is changed to red.
- Piston is driven to bottom (to maximize buoyancy).

5.Alternate/Exception Flows

6.Post Conditions.

- Vehicle remain in low power mode until the system is reset with “reset button”.

Case: Error mode.

1.Brief Description

When error is detected by software, vehicle is set in error mode

2.Actors

- MCU/System

3.Pre-Conditions

- Battery connected
- Necessary drivers are initiated.

4.Basic Flow

- LED light is changed to pink.
- Piston is driven to bottom (to maximize buoyancy).
- Error message is stored In SD- card.

5.Alternate/Exception Flows

-

6.Post Conditions.

- Vehicle remain in error mode until the system is reset with “reset button”.

System specifications Buoyancy vehicle

Use case – Power on:

- The system is powered on by activating a magnetic button.
- The vehicle should have LED light to localize the vehicle, and to indicate the system status and mode.
- When initialized the vehicle, default mode is standby/IDLE.

Use Case – Configuration:

- Configurations mode is entered with a magnetic button from standby mode.
- In configuration mode, Bluetooth communication will be established between the vehicle and a computer/phone
- The phone/computer will receive a menu to orient between vehicle settings, mission settings, transferring mission data and change mode.
- Status LED should indicate system mode.

Configuration variables:

Vehicle settings:

- PID parameters: Kp, Ki, Kd

Mission settings:

- Time [seconds]
- Depth [meters]
- Number of missions.

Use case – mission deployment:

- LED should be used to easily detect vehicle, and to indicate mission mode.
- The vehicle should dive to a certain depth for a certain length of time.
- The vehicle should be stable at the reference depth of plus/minus 0.2 meters over time.
- The following mission data should be measured and stored: Depth/pressure, time, piston position, temperature, and movement (accelerometer). The data should be stored on SD-card.

Use Case – Pick-Up:

- Status LED to indicate location and mode.
- Bluetooth communication should be establish on order to transfer/delete mission data files, and to change vehicle mode to standby, configuration or new mission.

Use Case – Power-Off:

- The system is power of by entering standby mode, where a timer turn off the system after 2 minutes of inactive period.

Exceptions/error handing:

- Status LED should indicate system status.
- In case of system error or sensor error, the system should float up to the surface, and indicated by status LED.
- In case of low battery power the vehicle should float to surface, and indicated by status LED.

Summary:

- The vehicle needs a magnetic activation button that powers the system on and change mode.
- LED with RGB color are used to locate the vehicle, and to indicate mode.
- SD-card is used to store mission data.
- Pressure sensor, timer, piston position, temperature and accelerometer data to be stored at SD-card, for every dive.
- Vehicle have several operating modes: Sleep state, Standby/IDLE, configuration mode, mission mode, pick-up mode, error mode and low power mode.
- A configuration interface is designed to communicate with the vehicle from computer/phone.
- Battery voltage is measured and logged.
- End limit switches is implemented to avoid damage on piston and actuator. End limit switches is also used to locate bottom piston position.
- The vehicle is designed for vertical movement between 0 and 50 meters below water surface.
- The vehicle should be able to stabilize at a reference depth with a variation of maximum plus/minus 0.2 meters.

Buoyancy vehicle functional test plan

- Reset system with “Reset” button.
  - Start program, initialized without error, enter IDLE state, LED white light, piston stop at bottom position (lower limit switch is activated).
- Waiting 2 minutes.
  - After 2 minutes in IDLE state system enters sleep state, LED turn off.
- Wake up system with magnetic button (Hall-effect interrupt).
  - System enters IDLE state, indicated with white LED light.
- Enter configuration state with magnetic button.
  - Vehicle enters configuration state, indicated by blue LED light, Bluetooth connected to phone/computer.
- Navigate through menu, change mission settings and vehicle settings.
  - Mission settings and vehicle settings are changed.
- Navigate to IDLE through menu, and back to configuration with magnet button
  - Settings hold the same value as above.
- Start new mission through menu.
  - Bluetooth disconnected, LED light changed to green, 1.txt file is created, PID controller saturates (due to constant pressure/depth value), mission data and time is stored in 1.txt file.
- Mission ended, and pick-up state entered.
  - Bluetooth connected, LED light changed to yellow.
- Start new mission through menu.
  - Same as above, new mission file is named 2.txt
- Mission ended, and pick-up state entered.
  - Same as above
- Configure state entered through menu.
  - Indicated by blue LED light.
- Transfer mission files (1.txt and 2.txt) to computer with Bluetooth.
  - Mission files are successfully transferred and stored on the computer.
- Navigate to IDLE state through menu.
  - IDLE state entered.
- Reduce power supply beneath 12v
  - Low-power mode entered, indicated by red LED light.
- Increase power supply and restart system
  - IDLE state entered
- Turn off system, remove SD-card and connect to a computer
  - Verify that the transferred mission file is identical to SD-card. (checking that file size matches should be sufficient).

Appendix C

Real time position monitoring

This appendix include Use Case, system specifications and functional test plan of the tracking system.

Use Case: Real time position monitoring

1. Brief Description

The tracking system utilize time difference of arrival (TDoA) to calculate the position of an acoustic receiver. TDoA uses the timestamp of signal arrival, of three fixed acoustic receivers. And the TDoA algorithm calculates the position of the acoustic transmitter.

2. Actors

- User
- MCU/system

3. Pre-Conditions

- Acoustic signal has free line of sight between all three hydrophones at the target location
- RF signal has free line of sight between SLIM modules and host computer.

4. Basic Flow

1. Prepare BV with acoustic transmitter (fish tag).
   - Make sure to use a fish tag that transmit depth data.
2. Place 3 nodes with ID 1,2 and 3 in a triangular shape around the target location.
   - Each node include acoustic receivers, SLIM module and ARF868 radio modem.
   - Node 1 is defined as the reference node and I placed towards west (to match plotting settings).
   - The acoustic receiver is placed just below water surface.
3. Power SLIM modules and establish GPS connection.
   - GPS connection is verified by on display onboard SLIM module.
4. Start host computer and wait for GPS location from SLIM module
5. Connection is verified with UNIX time being sent from each node every 5 seconds, and printed in terminal.
6. Place the BV in water and start a new dive.
7. Upon a new tag detection the timestamp, depth and SNR are sent to the host computer.
8. Host computer log the raw tag detection.
9. Host computer calculate TDoA position, save the data and plots the position.
10. BV are finished with the dive and return to surface.
11. Power down BV and SLIM module.

5. Alternative/Exeption Flows

7a. Only one or two acoustic receivers detect the tag fish.

- Count numbers of detections that where not completed.

6. Post Conditions

- Raw log file and calculated position are stored in two separate files when the dive is finished.

System specifications: Real time position monitoring.

- Display onboard SLIM module display connection between SLIM and Receiver, and GPS connection.
- Wireless transmission of data between host computer and SLIM module.
- Sufficient battery capacity for 10 hours.
- GPS location are sent from SLIM module to host computer.
- “Heart beat signal” is sent from the SLIM module to host computer every 5 seconds.

Functional test plan: Real time position monitoring.

- Set up two nodes (in dry conditions) and place a fish tag with equal distances between the two acoustic receivers.
  - Verify connection between SLIM module and host computer.
  - Verify GPS location.
  - Verify UNIX time from GPS.
  - Leave the two nodes for 10 hours and verify that both timestamps are equal within ±1ms.

Appendix D

User Guide

User Manual for Buoyancy Vehicle and TDoA Tracking System

Buoyancy Vehicle (BV):

Git repository

BV Firmware: https://github.com/haraldjord/Master-BV-Firmware

BV Plot Mission Log: https://github.com/haraldjord/Master-BV-plotTestData

BV Simulation: https://github.com/haraldjord/Master-BV-simulation

Getting started with Segger embedded Studio

Prerequisite:

- SDK v16.0 --> https://www.nordicsemi.com/Products/Development-software/nrf5-sdk/download
- Segger Embedded Studio v4.18 --> https://www.segger.com/downloads/embedded-studio/
- Debugger with SWD interface and pin connection.

Download SDK file, rename folder to <SDK_dir> and place the folder at the same level as BV Firmware files

Prerequisites

Before getting started with the Buoyancy Vehicle, make sure you have the following:

- Mac OS with Bluetooth or Windows/Linux OS with compatible Bluetooth USB
- Segger Embedded Studio
- Matlab
- Terminal app, checkout Halsos report for instructions on downloading the application and NodeJs
- Magnet
- Caliper
- Mounting tool

Step-by-step guide:

1. BV Initialization:

- When the BV is powered on, it enters the IDLE state.
- Use a magnet to activate the Hall effect button located 13 cm above the pressure sensor to enter configuration state.
- The BV will enter sleep mode after 2 minutes. Use the magnet again to re-enter the IDLE state.

2. Bluetooth Connection:

- Establish a Bluetooth connection through a terminal app or mobile device.

3. Mission Configuration:

- Configure the mission parameters such as time and depth, and the vehicle settings. The default settings should work fine in most cases.

4. Outer Lid Mounting:

- Mount the outer lid on the BV. See "Presetting of Outer Lid" section for detailed instructions.

5. Pressure Sensor Calibration:

- Calibrate the pressure sensor using BLE commands. The BV must be floating at the water surface during calibration (20 seconds).

6. Mission Start:

- Start the mission using BLE commands or the Hall effect button. The mission initiation is indicated by a green LED.
- To abort an unintended mission, use the Hall effect button.

7. Mission Completion:

- -When the mission is finished, the BV will float back to the surface.

8. Data Transfer:

- Re-establish the Bluetooth connection and transfer the mission log.

9. Data Analysis:

- Open the "plotTestData.m" file in Matlab and change the path to the last mission log.
- Various data, including step response and piston position, can be plotted and analyzed.

Important Notes:

- The BV will enter sleep mode if no Bluetooth connection is established, and default settings will be initiated, including the pressure sensor offset.

Presetting of Outer Lid:

Presetting of the outer lid is crucial for the BV's performance and its ability to return to the surface after diving. Follow these instructions:

- The "main_simulation.m" in the BV Simulation repository can interpret the density profile obtained from the CastAway CTD probe and calculate the recommended presetting of the outer lid. The simulation generates a text file named "BV-info" that provides the length from the lid to the bottom of the vehicle (h_bot), which is used for presetting the outer lid.
- Use a caliper to adjust the length according to the calculated value.
- Alternatively, when the water surface reaches the top of the 6 Allen bolts surrounding the LED, the vehicle should be preset for good performance.
- If a fish tag holder is mounted, the water surface should be approximately at the top of the holder.
- During the assembly of the outer lid, it is important to unmount the LED to release air inside the housing and avoid pressure buildup.
- During this project a presetting of h_bot = 95mm has been used in fresh water, and h_bot = 91 in sea water.

SLIM and Tracking System:

Git Repository

SLIM Firmware with Time Synchronization: https://github.com/haraldjord/SSM_v5-timeSyncing
TDoA Positioning Algorithm and Plotting: https://github.com/haraldjord/Master-positioning-algo

Getting started with SLIM project in Simplicity Studio 5:

1. Install Simplicity Studio 5:

- Download and install Simplicity Studio 5 (SS5) from the official website. Create a user account and log in if required.

2. Install SDK and Toolchains:

- When the Installation Manager pops up in SS5, select "Install by technology type" and choose "32-bit MCUs".
- Accept the agreements and wait for the installation to finish.
- Click on "Install" in SS5 and select "Manage installed packages".
- Choose the "SDK" option and click "Install All" in the Gecko SDK 3.2 box.
- Follow the same steps in the Installation Manager and install the GNU ARM v7.2 toolchain.

3. Clone and Import the Project:

- Find the clone link to the project on GitHub.
- Use Git to clone the project into the following directory: C:\Users\<user>\SimplicityStudio\v5_workspace\
- In SS5, click "File > Import..." and select the project folder in the specified path.

Setup of the Tracking System:

Prerequisites:

- TBR700RT hydrophones x3
- SLIM module x3
- ARF868 radio modem x4
- Fish tag with a depth sensor
- Host computer with Matlab and mapping toolbox

Node Setup:

- Set up the nodes in a triangular configuration, where each node consists of a SLIM module, a hydrophone, and a radio.
- Use the SLIM module with ID1 as the reference node, which will be placed at the origin in the local frame.

Data Acquisition and Processing:

- Start the Matlab script "SLIM-GUI.m" and change the COM port to match the radio modem.
- The script will plot the nodes as green circles when they transfer GPS location at intervals of 2 minutes.
- Incoming data will be saved as a raw log in a text file, separated by node ID, for backup purposes.
- Processed data will be saved as a struct "node" containing information from all three nodes.
- Note that the Matlab GUI is a working prototype, and bugs may occur.

It should be mentioned that TBR700RT store tag detections locally, if backups are required download ComPort from thelmabiotel.

NTNU
Norwegian University of
Science and Technology