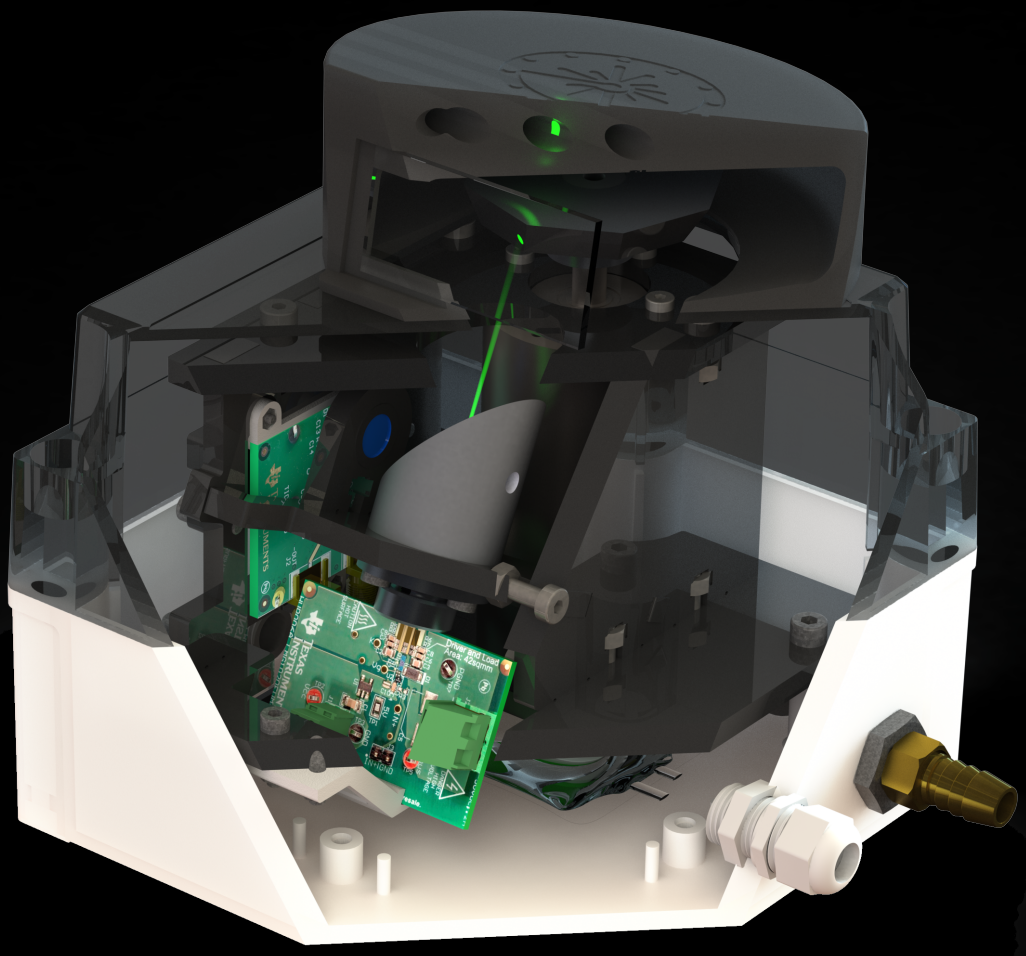
SLiDAR

Beam steering design report



TU Delft

Design of a Deep Sea LiDAR system

Beam steering design report

BSc Thesis

Authors:	J. Jonk	4577221
	B. Minderman	4577868
Project duration:	February 11 - July 5, 2019	
Supervisor:	Prof. Dr. Ir. G.J.T. Leus	
Thesis committee:	Prof. Dr. Ir. G.J.T. Leus	
	Prof. Dr. Ir. K.L.M. Bertels	
	Dr. D. Elkouss Coronas	

With contributions from the entire electrical engineering work group of the SLiDAR team in chapter 2.
Bsc. Electrical Engineering 2019 BAP group A :

A.E. Admiraal	4545532
J. Jonk	4577221
B. Minderman	4577868
B.M. Verdoes	4563263
L.R. Wix	4534654
E.P.M. Zwetsloot	4583817

This work builds on ideas from the mechanical engineering work group of the SLiDAR team.
Bsc. Mechanical Engineering 2019 BEP group A2 :

D. Looman
S.H.R. Rutten
B.R. van Vliet
D.O Wijnberg

In partial fulfilment of the requirements for the degree of

Bachelor of Science
in Electrical Engineering

at the Delft University of Technology.

An electronic copy of this thesis is available at https://repository.tudelft.nl/.

TU Delft

Abstract

Though several sensors are available for underwater scanning and ranging, they all have their limits. SONAR sensors are limited in resolution, and scanning mechanisms using a form of light for carrying the data suffer from high attenuation in turbid waters.

The goal of this project was to design a LiDAR system that is capable of overcoming these limitations. To tackle this problem, three groups were formed that focused on different parts of the LiDAR. However, first a complete system overview has been created by the entire project team. First a literature research has been performed. After that, system wide design decisions were discussed and made and also requirements were set. From there on, every subgroup would focus on their own part of the system. In this thesis a more detailed description of the beam steering module shall be given. Again some literature research has been performed. After that, the design decisions were made, based on the requirements. Finally a design was implemented and tested.

For the beam steering module to be successful, it should provide accurate control over the angle at which the laser beam is sent out. Though improvements can be made, the system does comply to the minimum accuracy requirement.

Preface

This thesis has been written as part of the bachelor graduation project of Electrical Engineering at the Delft University of Technology. For this project, groups of six students are formed. These six students are then further divided into three thesis groups of two students each. For this specific project, also four mechanical engineering students collaborated on the project. Hereby, we would like to thank Danny Looman, Stephan Rutten, Bas van Vliet and Onno Wijnberg for their contributions to the project.

Though our project was divided among three subgroups, also some work has been performed to which everybody has contributed. This is the research to the complete Light Distance And Ranging (LiDAR) system, which shall be described in the first few chapters. Though all have contributed to this part, some sections are more important for one subsystem than for another. Of particular importance for our thesis, is the discussion on the different LiDAR scanning techniques, as this determines what the electrical part of the beam steering module should be capable of.

As the system would be incomplete without any of the modules, a thank you to rest of the project group is in place. Furthermore, we would like to thank our sponsors. On top of this, we would like to thank Dr. Ir. Lager for his helpful feedback during the project. Most of all, however, we want to thank Prof. Dr. Ir. G.J.T. Leus, who was our supervisor during the entire project, for his time and effort and also for the useful feedback.

J. Jonk & B. Minderman

Delft, June 2019

Contents

List of Abbreviations

ADC Analog to Digital Converter

APD Avalanche photodiode

ASIC Application-Specific Integrated Circuit

AUV Autonomous Underwater Vehicle

BLDC Brushless Direct Current

CFAR Constant False Alarm Rate

CRC Cyclic Redundancy Check

DOT Design Option Tree

DTC Direct Torque Control

FBS Functional Breakdown Structure

FOC Field Oriented Control

FOV Field Of View

FPGA Field-Programmable Gate Array

IC Integrated Circuit

LiDAR Light Distance And Ranging

LLS Laser Line Scanning

LOBSTER Life OBserving Sea Traversing Explorer Robot

LSB Least Significant Bit

MEMS MicroElectroMechanical Systems

MSB Most Significant Bit

OPA Optical Phased Array

RADAR Radio Distance And Ranging

RDT Requirements Discovery Tree

SfM Structure from Motion

SLAM Simultaneous Localisation And Mapping

SLiDAR Subsea LiDAR

SNR Signal to Noise Ratio

SONAR Sound Navigation And Ranging

SPI Serial Peripheral Interface

SSI Synchronous Serial Interface

TDC Time to Digital Converter

ToF Time of Flight

UART Universal Asynchronous Receiver-Transmitter

VCSEL Vertical Cavity Surface Emitting Laser

VESC Vedder Electronic Speed Control

1 Introduction

Although deep water ecosystems are crucially important for human interests, they remain far from being fully understood [72]. It is simply very hard to get there. One of the challenges of subsea exploration is navigation. Satellite navigation signals are quickly attenuated in seawater and acoustic beacons are costly to set up. Conventional robotic systems employ Simultaneous Localisation And Mapping (SLAM) techniques to find their position from ranging information in this absence of positional references [59]. Limitations of subsea perception sensors make this difficult in the subsea environment [22].

LiDAR systems are often used in ranging applications above sea, but are not traditionally considered for use in subsea SLAM implementations [22]. Although subsea LiDAR systems exist, it is only marketed at asset inspection and metrology. Although it has high resolution and accuracy, the maximum scan speed of $2\,\mathrm{min}^{-1}$[17] and high price of €500k probably contribute to its lack of application in subsea SLAM.

The goal of the Subsea LiDAR (SLiDAR) project is to explore the feasibility of an inexpensive subsea LiDAR system optimised for SLAM. This challenge is jointly tackled by a team of four mechanical engineering students investigating mechanical problems in their bachelor thesis and a team of six electrical engineering investigating electrical aspects in their thesis.

This thesis focuses on the electrical design, which is divided over:

- the transmission stage, generating high power laser pulses;
- the receiver stage, converting the Time of Flight (ToF) of the laser pulse to a digital reading;
- a scanning stage, distributing the distance measurements in space;
- a digital system, to time the measurements and communicate them to the outside world.

The present work details the scanning stage.

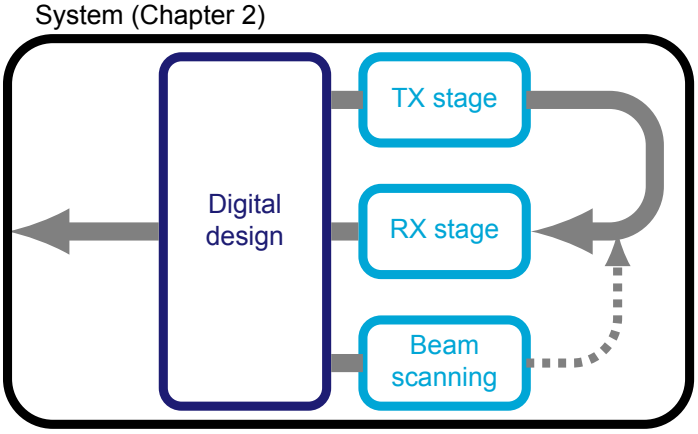


Figure 1.1: System overview.

1.1. Beam Steering

As part of the SLiDAR project, a control system for directing the laser beam has to be designed. For the LiDAR designed in this project, the beam steering is accomplished via a polygon mirror on a motor. Why this particular scanning technique has been chosen shall be further explained in chapter 2, but the consequence is that a proper motor has to be selected and a motor driver has to be designed. Besides this, also some form of motor position feedback has to be implemented as the motor position is an important parameter for the laser transmission and reception stage of the LiDAR.

In this thesis it shall be investigated what motor and driver are best suited for the specific application of SLiDAR. Also, a position feedback system shall be designed and implemented. A first step in this is conducting a literature research on motors, motor driving techniques and rotary position feedback mechanisms. After the literature has been studied, a suitable motor can be chosen and a driver can be designed based on the driving techniques found in the literature. Furthermore, a position feedback mechanism shall be designed, for providing the rest of the system with the required information. Once everything has been implemented, the performance of the beam steering module shall be tested based on the requirements stated in chapter 2.

1.2. Synopsis

The rest of the report shall be structured as follows. First of all a common part, on which all subgroups contributed, shall be discussed. This common part will provides a discussion on the system design in chapter 2. Here the importance of a subsea LiDAR shall be stated. Furthermore, this chapter contains all system wide design decisions as well as the requirement specifications. In chapter 3, the thesis specific part will start with a discussion on design decisions made for the beam steering module. This is followed by a detailed description of the implemented design in chapter 4. After the implementation, the performance deduced from the test results shall be discussed in chapter 5. Chapter 6 again belongs to the common part and discusses the integration of all three modules that have been created. Finally, in chapter 7, the conclusions are stated and a discussion is held. This chapter also contains a recommendation for future work.

2

SLiDAR System Design

In this chapter, we explore system architectures suitable for SLAM capable deep sea LiDAR and establish a subsystem division for our implementation along with requirements on each subsystem. To this end, section 2.1 first motivates the choice to investigate deep sea LiDAR as a solution to the underwater SLAM problem by exploring its relation to current underwater ranging methods in a review of current underwater ranging methods. From there, section 2.2 will examine the system architecture of a simple LiDAR system for context, before we take the discussion to a more abstract point in section 2.3, identifying the fundamental functions of LiDAR systems. Section 2.3.3 will provide a summary of them in the form of a Functional Breakdown Structure (FBS) for a more visual overview. It will also serve as the basis of a Requirements Discovery Tree (RDT) in section 2.4.1, ensuring exhaustive coverage by requirements. Section 2.4.2 will detail them in a programme of requirements. Having established the functions of and requirements on the deep sea LiDAR, we will seek design options to fulfil these by analysing the state of the art of LiDAR techniques in section 2.5. These will be summarised in a Design Option Tree (DOT) in section 2.5.5, again providing a visual overview for further reflection. This reflection comes in the form of the selection of techniques best suited to our application to form a concept design by weighing their advantages and disadvantages in section 2.6.From here, functionalities are grouped in subsystems in section 2.7. This split generates the need for more detailed requirements on system-level key parameters. To find these, some system-level modelling is performed in section 2.8. Finally, section 2.9 gives a high-level overview of the proposed LiDAR design, and specifies the data flow on the interfaces between the made subsystems.

2.1. Current underwater ranging methods

This section explores the ways in which an underwater SLAM capable LiDAR could contribute to underwater SLAM by evaluating the strengths and weaknesses of current underwater ranging methods and investigating the unique role a LiDAR system could play.

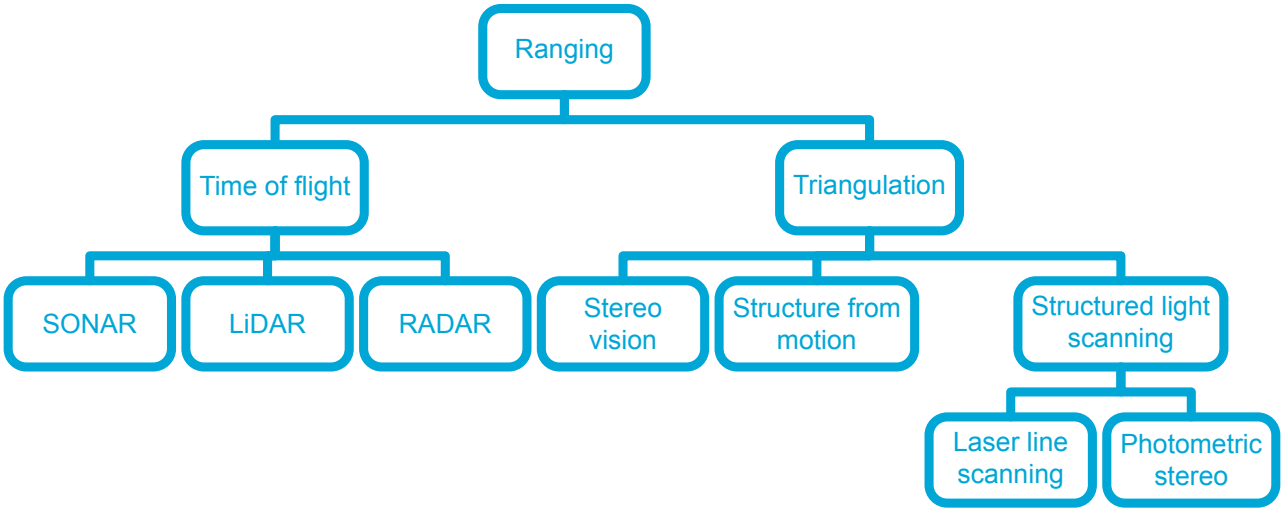


Figure 2.1: Classes of ranging methods. Adapted from [37, 29].

Current ranging methods can be classified as shown in figure 2.1. We distinguish two major classes based on the measured quantity:

- Triangulation methods estimate the distance to an object from the angles of at least two rays intersecting a certain point on an object.
- ToF methods estimate the distance to an object from the return time of an echo in a certain domain.

Triangulation methods can be further divided based on the types of angular measurements. A distinction can be made between stereo vision, structure from motion and structured light scanning. In industry, all of these systems are implemented optically. This introduces a trade-off between sensor size, sensitivity and resolution. Since sunlight does not reach the deep sea, there is also a trade-off between the sensor to illumination separation distance and interference due to volumetric backscatter.

Stereo vision uses concurrent angular measurements to a point from multiple vantage points. Any underwater can be used, from inexpensive consumer level action cameras, to machine vision cameras in $1500 pressure housings [61] or even dedicated deepwater equipment, such as the $60k Sulis subsea camera [68], any of the DeepSea Power & Light equipment [16] or Sidus cameras and lighting solutions [64]. Since stereo algorithms depend on detecting features of the environment, the barren seafloor can pose a challenge [44]. Nevertheless, stereo vision has been demonstrated in a natural environment at 1750 m [9]. Underwater stereo based SLAM has also been demonstrated [57], but low visibility remains an issue. Though few authors report maximum range, 10 m seems to be a good estimate based on qualitative inspection of the visual results in [57].

Structure from Motion (SfM) approaches find the 3D structure of an object through angular measurements from a great number of overlapping but unknown vantage points. They are closely related to stereo methods, also requiring significant processing. Again, any underwater camera can be used. Conventional SfM methods have been used to map coral reefs [11] and in general aquatic applications of the technology are regarded to be rapidly developing [13]. Difficulties in detecting features on the barren seafloor can again pose a challenge, but can ultimately be overcome such that underwater monocular SLAM has been demonstrated[44]. SfM approaches suffer the same limitations as stereo methods.

Structured light scanning uses a light source with a known angular distribution and measures the angles of the reflected light. Laser line scanners are common underwater structured light scanners. Examples include the Newton M310UW [45], the Kraken SeaVision [55] and the 2G Robotics ULS-500 pro [1]. Thanks to the use of laser light, excellent resolutions can be obtained down to 0.0057° (SeaVision). In clear water, the range is limited to 20 m (ULS-500 pro), though usually lower at 5 m (M310UW) or 8 m (SeaVision). To obtain the long range, the trade-off between low volumetric backscatter and sensor to light distance makes for a bulky 1.2 m system (ULS-500 pro). Fighting the trade-off between sensor size, sensitivity and resolution likely contribute to high system cost, $200k for the ULS-500 pro for example. Some systems, such as the ULS-500 pro are dynamic scanners, such that they rely on the movement of the vehicle for one of their scan axes. This makes them unsuited as a sole sensor for SLAM, though still usable in the more common fusion approach.

ToF systems can be further divided based on the domain of the echo. Sound Navigation And Ranging (SONAR) systems use acoustic echos, LiDAR systems use optical echos and Radio Distance And Ranging (RADAR) systems use radio echos.

RADAR is not suited for underwater use as the signal simply attenuates too fast in water to obtain usable results. For frequencies high enough for practical antenna sizes, the attenuation of radio waves in seawater is too high to perform medium distance ranging (>1 m), due to the combination of the conduction losses at low frequencies [6] and the attenuation of pure water at high frequencies [60]. This makes underwater RADAR unfeasible.

SONAR widely employed for underwater scanning, since its range is not limited by the turbidity of the water. Edgetech [18], Klein Marine [63], Coda Octpus [14], Tritech [74] and Teledyne [69] all produce a multitude of marine SONAR systems. Two SONAR techniques that are often employed are multibeam and side scan SONAR. Multibeam sonar uses phased arrays to steer the laser beam. There is also side scan SONAR. A SONAR beam orthogonal to the direction of motion of the vessel, is combined with a simple altimeter to form a map. SONAR does have its limitations. [22] points out that side scan sonar does not really provide 3D information, since the reflection intensity depends on the orientation of the surface, rather than the distance. The researcher mentions that for practical transducer sizes, the angular resolution remains limited due to the relatively large wavelength of the sound waves. For example, the $25k Teledyne blueview has a resolution of 1° at a range of 10 m with up to 1 Hz update rates. The $200k Coda Octopus Echoscope 4G boasts beam spacing down to 0.19° and at a range of 20 m with up to 20 Hz update rates. The Tritech Gemini 720ik has resolution of 0.25° at a range of 1 m and an absurd 97 Hz.

To the best of our knowledge, only one company offers underwater LiDAR. 3DAtDepth has designed a $500k underwater LiDAR system that has a range precision of 6 mm and an angular resolution of less than 0.025°. The maximum

range is limited to 45 m [17]. Some authors also distinguish the INSCAN system by Teledyne CDL [19], but the 3DAt-Depth optical stage seems to have been used [34]. Thanks to the short optical wavelengths, the resolution of LiDAR is higher than SONAR. However, the scan rate is too low for SLAM applications at $2\,\mathrm{min}^{-1}$. Furthermore, the range is limited by turbidity. Nevertheless, LiDAR systems can assist in increasing efficiency of bathymetry surveys, though cost remains a limiting factor [19].

Further variations are found by combining these methods. For example, Pulse Gated Laser Line Scanning (LLS) employs range gated imaging to reduce interference from backscattering in a structured light scanning method [12].

In conclusion, each underwater ranging method comes with its unique detractors. All solutions in industrial use are either expensive or very expensive. Many are inherently depth-limited because of their reliance on pressure vessels. LiDAR systems are certainly not a solution to all underwater SLAM problems, but given their prominence in regular robotics, they deserve more exploration. Low cost is an especially attractive property. A significant part of the cost of current systems can be attributed to the pressure vessel [19], which can be omitted by using pressure tolerant design techniques [46]. For these reasons, the remainder of this thesis will focus on determining the feasibility of a deep sea SLAM capable pressure tolerant LiDAR system by facing the design challenge.

2.2. A simple LiDAR system

Having motivated the choice to investigate deep sea capable LiDAR as a potential solution to the deep sea SLAM problem, we will now examine a simple LiDAR architecture to provide sufficient context to perform more abstract analysis in later sections.

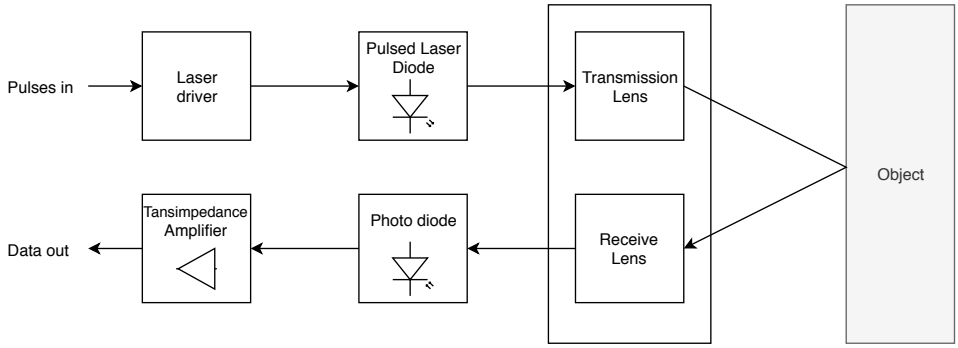


Figure 2.2: Schematic view of a simple LiDAR system, adapted from [71]

A general overview of a simplified LiDAR system can be seen in figure 2.2. The basic concept of a LiDAR system is to send out a pulse of light and measure the time it takes for the pulse to return. Assuming the propagation speed of light is known, a distance can be calculated from this time of flight. Since light spreads and attenuates over distance, the reflection is very weak. Detection is usually aided by sending out a very bright flash of light. The light is also commonly focussed to a tight, collimated beam. This ensures that all the light is reflected from a single target of interest. Laser diodes are commonly used as a means of providing fast light pulses. In most cases, a wavelength is chosen that is not dominantly present in sunlight. This minimises interference.

Generally, the creation of a light pulse is preceded by a timing system generating a pulse with the correct width and at the correct time. This signal generally does not have the signal strength to power the laser, so a laser driver is used. It amplifies the incoming pulses so that the laser diode receives sufficient power to shine brightly. When the light leaves the system, the light gets focused by a lens to obtain a collimated beam. The entire LiDAR system is usually rotated to scan the beam the environment using a gimbal. This ensures that a depth map can be built of the environment - not just a single point.

Once the laser beam hits an object, a small portion of light will reflect back towards the LiDAR system. The light that returns to the LiDAR system will then usually be focused on a photo-diode by a second lens. The photo-diode will produce a small current, which depends on the intensity of the light that hits the photo-diode. This signal is passed through a transimpedance amplifier for further processing. Often, a comparator is used to check whether it passes a certain threshold. If so, a target detection is considered to have taken place and the time is stored. This is then compared with the time the pulse was sent to find the time of flight.

2.3. Functions of LiDAR systems

We will now generalise the functions encountered in the simple LiDAR system described in the previous section to extract the more abstract functionalities of LiDAR systems, which will enable us to start defining requirements in the next section. We will only consider standard ranging LiDAR, not doppler LiDAR [8] or imaging LiDAR [21, 50].

2.3.1. Channel estimation and target detection

From an abstract view for the simple LiDAR system described in section 2.2, sending out a pulse and checking whether the return signal passes a certain threshold is not one operation, but two. The first is the estimation of the backscatter impulse response of the optical channel, which is the received response after sending out a light pulse. The second is the detection of an obstacle from this impulse response. This is accomplished by determining whether the estimated backscatter impulse response is larger or higher at a certain location using the comparator.

Though this additional distinction may seem arbitrary, it enables the classification of many LiDAR technologies, as we will show in section 2.5. It also allows convenient mathematical description, as shown in section 2.8.

2.3.2. The abstract channel & sample distribution

Until now, we have assumed that an optical channel is given, but in the simple LiDAR system described previously, it is determined by gimballing the LiDAR device such that the optical channel spans different parts of the environment. In a more abstract sense, this is a method to distribute the various distance samples over space. This is required in any 3D LiDAR system, since electrical signals are modelled as 1-dimensional quantities, whilst the environment is a 3-dimensional space. The distribution of measurement samples is thus the problem of splitting the 3-dimensional environment in 1-dimensional channels. The impulse response of the optical channel then becomes a function of the volumetric optical properties of the environment and a steering coefficient vector. It needs to be stressed that this channel need not necessarily be a simple linear trans-section of the environment, but could just as easily be a superposition of these.

2.3.3. Towards an FBS

In summary, measuring the distance to the environment by estimating the optical ToF has three facets:

- representing the 3-dimensional environment in 1-dimensional channels, since only these can be transduced to the inherently scalar electrical signals;
- optical channel estimation, where the backscatter of the channel over distance is estimated;
- target detection, where the location of this target is detected on the basis of the estimated channels.

Channel estimation is an experiment on the channel. This requires the emission of test signal into the channel and the reception of the return signal. Furthermore, like all systems the system needs to interface to a user and needs to preserve itself to be useable. This motivates the FBS in Figure 2.3.

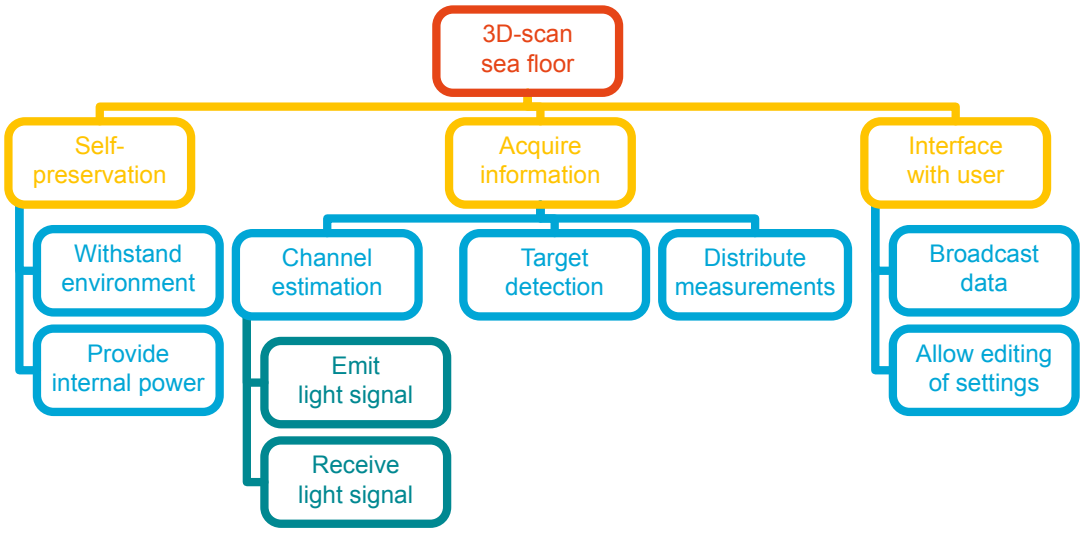


Figure 2.3: LiDAR FBS.

2.4. Requirements

2.4.1. Towards a RDT

In order to obtain an exhaustive list of requirements for the system, we constructed a RDT, which will serve as the basis of the upcoming programme of requirements. This entailed finding system-wide performance aspects for each abstract function of the system defined in the FBS discussed in the previous section. The resulting RDT can be found in figure 2.4.

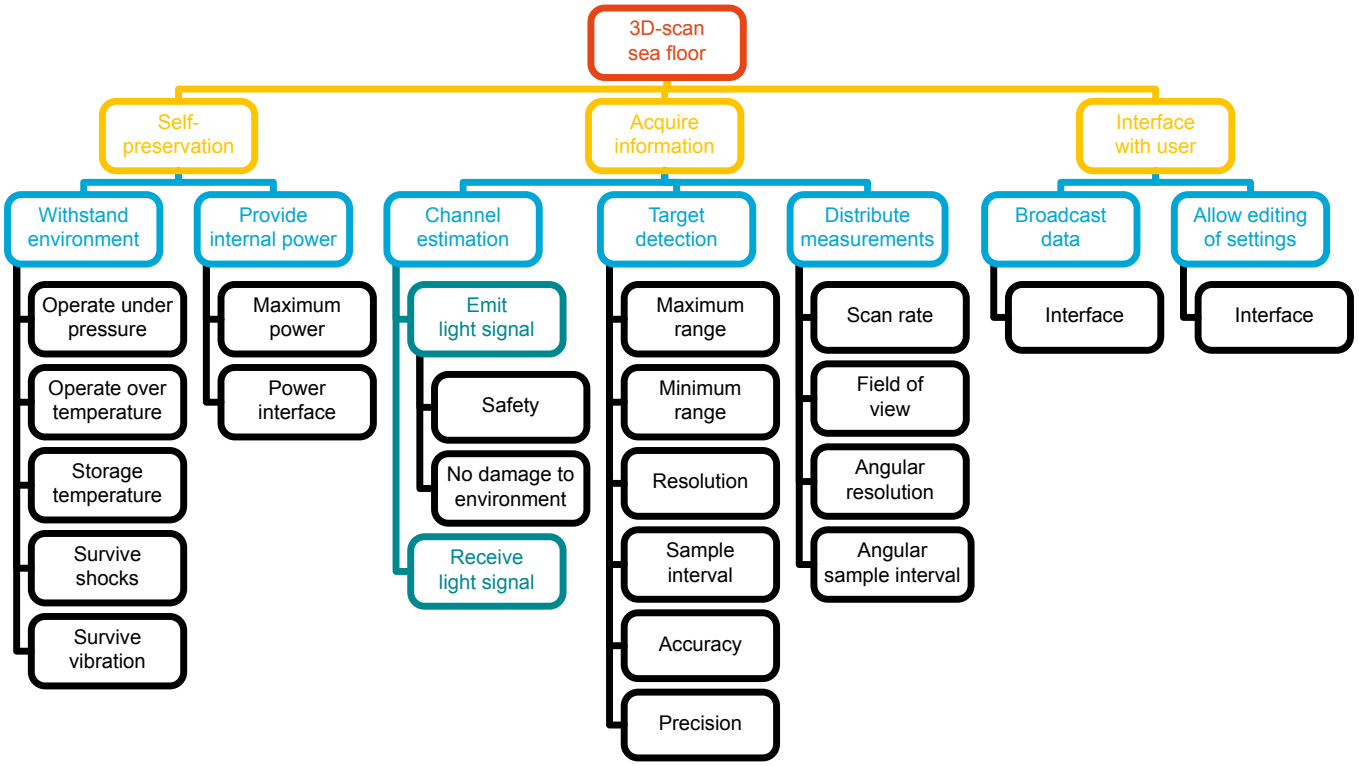


Figure 2.4: SLiDAR RDT.

2.4.2. Programme of Requirements

The functions found in the FBS, the requirements found in the RDT and the constraints on the project, form the basis of the program of requirements, which will enable judging of the suitability of design options to form a concept design in later sections. We quantified the requirements based on our market research (part of which is represented in section 2.5) and estimations of the performance required for SLAM. Most of the obligatory requirements were roughly quantified based on an inexpensive triangulating laser scanning system found on entry-level SLAM capable systems, the RPLidar-A1 [66]. The more ambitious goals are based on the unique and high end SL3 underwater LiDAR from 3DAtDepth [17] and the Velodyne Puck [76], which has been a standard for research into the application of SLAM to self-driving vehicles. Further detailing yielded the programme of requirements on the system shown in table 2.1. We have given each requirement a unique identifier for reference throughout this thesis. We have also given our justification for including the requirement or the value of the requirement for future reference. Finally, we have indicated how the requirement is to be verified as a first step in the creation of verification plans.

Table 2.1: System-wide requirements and justification

ID	Requirement	Verification				Justification
		Test	Demonstration	Analysis	Inspection	
FUN1	The system shall estimate the distance to the environment by measuring the optical time of flight at certain sample points within a certain angular region.	•				Basic functionality of a LiDAR.

Continued on next page

Table 2.1 – *Continued from previous page*

ID	Requirement	Verification				Justification
		Test	Demonstration	Analysis	Inspection	
FUN2	The system shall be able to stream this measurement data via a serial data protocol.		•			Enables use of the system.
FUN3	All variable parameters should be user-configurable over the serial data protocol.		•			Increases flexibility of the system.
ENV1-a	The system and its internals shall operate in compliance with the requirements up to a pressure of 100 bar.	•				Test limit of MacArtney pressure chamber.
ENV1-b	The system and its internals should operate in compliance with the requirements to a pressure of 600 bar.	•				Test limit of AWI pressure chamber.
ENV2-a	The system shall operate in compliance with the requirements in a temperature range between −10 °C and 50 °C.			•		Minimum and maximum ocean water temperatures, with margin.
ENV2-b	The system shall survive temperatures in a range between −60 °C and 70 °C.			•		Minimum and maximum atmospheric temperatures, with margin.
ENV3	The system should survive shocks.	•				In case the robot bumps into obstacles.
ENV4	The system should survive vibrations.	•				To avoid damage from thruster vibrations.
SAF1	The system shall comply with the intent of the IEC60825 safety standard.			•		Ensure team safety.
INF1	The system will receive a 48 V connection.				•	LOBSTER Explorer bus connection.
INF2	The system shall draw no more than 100 W.		•			Sufficiently low power for the LOBSTER Explorer.
INF3-a	The system shall be able to communicate over an Serial Peripheral Interface (SPI) bus.	•				Simple yet high speed and versatile interface.
INF3-b	The system should communicate over a 100BASE-TX Ethernet link or faster.	•				High-speed interface of the LOBSTER Explorer.
PER1-a	The system shall have a maximum range of at least 10 m in 95% transmissive ocean water in a typical ocean environment.	•		•		Braking distance with margin of LOBSTER Explorer.
PER1-b	The system should have a maximum range of at least 40 m in 95% transmissive ocean water in a typical ocean environment.	•		•		Range of commercial systems.
PER2-a	The system shall have a minimum range less than 1 m.	•		•		Braking distance with margin of LOBSTER Explorer.
PER2-b	The system should have a minimum range less than 0.05 m.	•		•		Range of commercial systems.
PER3-a	The system shall have a distance resolution of less than 5 cm.				•	Estimated minimum required for SLAM.
PER3-b	The system should have a distance resolution of 1 mm.				•	Resolution of commercial systems.
PER4-a	The system shall have a distance accuracy of a standard deviation of 5 cm.	•				Estimated minimum required for SLAM.
PER4-b	The system should have a distance accuracy of a standard deviation of 1 mm.	•				Accuracy of commercial systems.
PER5-a	The system shall have a distance precision of 1%.			•		Precision of commercial systems.
PER5-b	The system should have a distance precision of 100 ppm.			•		Estimated minimum required for SLAM.
PER6-a	The system shall have a minimum Field Of View (FOV) of 30° · 30°.				•	Estimated minimum required for SLAM.
PER6-b	The system should have a FOV of 180° · 90°.				•	FOV of commercial systems.
PER7-a	The system shall have a maximum azimuthal sample interval of 1°.				•	Estimated minimum required for SLAM.
PER7-b	The system should have a maximum azimuthal sample interval of 0.1°.				•	Azimuth sample interval of commercial systems.
PER8-a	The system shall have a maximum zenithal sample interval of 15°.				•	Estimated minimum required for SLAM.
PER8-b	The system should have a maximum zenithal sample interval of 0.1°.				•	Zenith sample interval of commercial systems.
PER9-a	The system shall have an angular accuracy better than 1°.	•				Angular accuracy comparable to minimum angular resolution.
PER9-b	The system should have an angular accuracy better than 0.1°.	•				Angular accuracy comparable to maximum angular resolution.
PER10-a	The system shall have an angular precision better than 1°.	•				Angular precision comparable to minimum angular resolution.
PER10-b	The system should have an angular precision better than 0.1°.	•				Angular precision comparable to maximum angular resolution.
PER11-a	The system shall have an maximum update rate of at least 10 Hz.				•	Estimated minimum required for SLAM.
PER11-b	The system should have an maximum update rate of at least 30 Hz.				•	Update rate of commercial systems.
PER13-A	The size of the system shall fit in the mechanical design of the LiDAR			•		As presented by the team of mechanical engineers [58]
PRO1	The project shall be completed by June 21th of 2019.				•	Requirement from BAP.
PRO2	The project shall be documented according to the Electrical Engineering BAP guidelines.				•	Requirement from BAP.
PRO3	The project shall be completed by a team of 6 3rd year bachelor students.				•	Requirement from BAP.

Continued on next page

Table 2.1 – *Continued from previous page*

ID	Requirement	Verification: Test	Verification: Demonstration	Verification: Analysis	Verification: Inspection	Justification
PRO4	The development and fabrication of the electrical system shall cost no more than €3000.				•	Budget.
PRO5	The mechanical scanning stage, optics and embodiment will be developed by a BEP team of mechanical engineers.				•	Agreement with the BEP team of mechanical engineers.

2.5. Design options

Having defined the functions of and requirements on our deep sea LiDAR system, this section aims to explore aspects of LiDAR system architectures which fulfil these so that they can be considered as design options for the concept design. To this end, we examine the current state of the art LiDAR techniques along with their merits and weaknesses.

We have organised the techniques by the function they implement to emphasise their commonalities and differences. After considering various solutions for distribution, channel estimation, target detection and digitisation, we summarise these in a DOT for further reflection.

2.5.1. Distribution

There are two main approaches to implement the distribution of samples over space:

- scanning systems, where a single measurement beam is rotated over time so that consecutive samples cover different directions;
- concurrent sampling or scannerless systems, where multiple directions are sampled concurrently.

Gimballing systems

The advantage of scanning systems is that they make efficient use of the relatively expensive ToF estimation ('laser profiling') hardware. Hence, when LiDAR technology was first developed, only scanning systems were economically feasible. Early scanning systems were mechanical in nature, gimballing the entire laser profiling assembly. The advantage of gimballing is that light beams always enter the receiver from the same angle, simplifying the optical path and ensuring that the effective reception area is constant over any beam deflection ϕ. For these reasons, this method still enjoys widespread use today, primarily in 'spinning LiDARs', which rotate over a single axis. Examples include [49, 76, 65]. There is the disadvantage of requiring connections over one or more rotating axles. This is solved either using slip rings, which are widely considered unreliable and which are unsuited for high-speed data transfer, or using coupled coils for power and optical transmission for data [35], which add complexity.

Polygon scanners

Polygon scanners are another mechanical scanning method, where a rotating mirror is used to direct the beam over the FOV [36]. Irregular polygon scanners even enable scanning in the vertical direction by angling each face of the polygon over an axis orthogonal to the rotation direction [36]. These decouple the scanning and laser profiling, but they introduce a Trade-off between the maximum FOV and the amount of vertical samples. They can be used with a separate, fixed reception stage, but this would decrease the effective reception area with $\cos(\phi)^2$ for samples towards the edges of the FOV. In contrast, it is possible to use the mirror as the return path for the light [32]. In this case, the mirror is at an angle of $\phi/2$ to the returning light, such that the effective reception area decreases with $\cos(\phi/2)^2$ towards the edges of the FOV.

MEMS micromirrors

MicroElectroMechanical Systems (MEMS) micromirrors are tiny oscillating mirrors used to direct the the beam over the FOV [81]. Their small size is an advantage, though the aperture size and resonant frequency are at odds with

each other [42]. They are often driven at resonance to extend the FOV, but when used in liquids, their frequency response is damped such that this is no longer possible and their scan frequency reduces by an order of magnitude from typicall 300 Hz to 30 Hz [80, 77].

Optical Phased Arrays (OPAs)

With the advent of autonomous driving, manufacturers want to increase the reliability of scanning methods, generating interest in solid-state scanning methods. One of these is the optical phased array, where the beam is composed of a superposition of many smaller beams with tunable phases, resulting in a steerable beam [52]. Advantages include high scan speed, small size and high reliability, but the technology is still in its infancy and thus very expensive, in academia [42] and in industry [48].

Scannerless versus scanning systems: merits and weaknesses

Thanks to modern advances in miniaturisation, scannerless systems have become viable. The advantage of scannerless systems is that they have a greater scan time per sample, so that the same pulse energy can be achieved using lower peak power illumination. In addition, when limited by eye-safety, longer pulses are permitted higher pulse energies. However, macroscopic mechanical scanning methods and especially spinning LiDARs have the advantage that different pulses are emitted in different positions, such that only a fraction of the total emitted flux flows through an eye-sized area. This means that the pulse energy can be increased [73]. All scanning systems have the advantage of being able to incorporate more refined laser profiling hardware, since cost and effort do not have to be shared over multiple channels but can be focused in a single channel [42].

Scannerless arrays

Perhaps the simplest is creating an array of differently directed laser profilers operating simultaneously, as demonstrated by Ouster in [51]. They show that laser emitters can be integrated on a single Integrated Circuit (IC) using Vertical Cavity Surface Emitting Laser (VCSEL) technology. Advantages include high reliability and low system complexity. Scannerless arrays are often referred to as 'flash multibeam LiDAR'. In this, 'flash' denotes the simultaneous capture of multiple samples, like in a camera sensor illuminated by a flash, and 'multibeam' denotes the multiple targeted beams of light, only illuminating the environment where sampled [51].

Diffuse source

Instead of replicating directed light sources with each laser profiler in an array, a single diffuse light source can be used [40]. This technique is often referred to as 'flash LiDAR', since multiple samples are captured simultaneously using the same flash of light. Shorter-range systems meant are also referred to as ToF cameras [20]. The advantage is lower system complexity, however much will fall on areas of the environment which are not sampled. Since the total luminous flux flows through a single eye-sized area, the energy of the flash is limited by eye-safety. Then this limits the effective pulse energy at each sample location compared to systems incorporating transmitter arrays.

Coded aperture

One intriguing possibility is to use Compressed Sensing techniques to reconstruct a scan from samples of a small amount of select beam patterns. A few investigations have been carried out on coded aperture LiDAR [27] and [62]. Both systems send out a diffused source and capture the returned signal with multiple sensors. In comparison with flash LiDAR, less sensors could be used if a static coded aperture is used, this would reduce the cost significantly. Although most literature on compressed sensing does not discuss the effect of noise introduced to the signal before detection, it has been proven that in some cases the signal to noise ratio decreases due to noise folding [5].

2.5.2. Channel estimation

The complexity of optical channel estimation techniques is limited due to the required high operating speed. Because of this, only matched filter approaches are commonly implemented and the possible application of modulation is the main difference between these techniques. We will now consider the most common modulations.

Single pulse

Single pulse LiDAR, or 'pulsed' LiDAR, is perhaps the simplest modulation, and certainly very common in industry [30]. With it, a single pulse high-intensity pulse is emitted each sample [71]. Since the pulse can be modelled as a bandwidth-limited delta pulse, the matched filter is simply a low-pass filter, which is inherent to any input stage. Pulsed

illumination can also be easily achieved using a push-pull driver, such that the output stage can be very simple. In short, simplicity is the main advantage of single pulse LiDAR, and it is certainly prevalent in scannerless systems. This simplicity does come at the cost of a relatively large susceptibility to external disturbances. Since all energy is concentrated in a single peak, the pulse energy will be limited by the peak power of the electro-optical transducer, which may be low. Furthermore, at short pulse lengths only low pulse energies are considered to be eye-safe.

Harmonic

Some time of flight cameras use a harmonic signal[1] for channel estimation. In this case, the time of flight is calculated from a relatively simple phase shift measurement [31]. The advantage is that the usual complexities of high speed ToF estimation can be avoided, since only low-speed amplitude measurements are required. However, usually a harmonic signal would not even be considered for channel estimation. This because due to the tiny bandwidth, all objects in the channel will alias on top of each other. Thankfully, objects farther away in the channel generate weaker returns due to R^{-2} spreading. Combined with the small spot enabled by the optical extremely high frequencies exploited by LiDAR this makes the nearest return dominant, which is usually the only return of interest. However, it is impossible to distinguish a dominant return from a dominant return at a length $c \cdot f/2$ further away, since both alias to the same point. Hence, the domain of harmonic channel estimators is usually limited to $[0, \cdot f/2)$ in order to prevent these range ambiguities. Though this limit could be overcome by using a select few frequencies, any amount of near-field volumetric backscatter due to the environment could become the dominant return at longer ranges due to higher R^{-2} spreading. Hence, harmonic channel estimation is usually reserved for short range indoor use in time of flight cameras.

FMCW

Frequency Modulated Continuous Wave (FMCW) systems continuously transmit a frequency modulated light wave. This modulation enables a Continuous Wave system to be used for distance measurements. The distance measurement for FMCW is done by comparing the received signal with a reference signal [33]. The distance between the source and the reflecting object can be calculated as $R = c_0|\Delta t|/2$. Here Δt is the measured delay time between the reference and the received signal. An inherently high SNR makes this technique advantageous. A disadvantage is the need of high bandwidth to increase the maximum ranging distance, and the need for a heterodyne receiver.

Pulse trains

In pulse train implementations, multiple pulses are sent for every distance sample. The received signals are then averaged for higher Signal to Noise Ratios (SNRs). Correlation processing is then used to find the ToF [29]. The advantage of a pulse train is that the total pulse energy is spread over many pulses with an equivalent bandwidth, such that the peak power can be reduced. The downside is that in low SNR regimes, the nonlinearity of detection methods cause noise standard deviation after detection to grow superlinearly as a function of noise standard deviation before detection. Because of this, the pulses must be averaged before detection. This either requires expensive full digitisation of the amplitude of the pulses or complex analog filters.

2.5.3. Target detection

As with channel estimation, the complexity of optical channel estimation techniques is limited due to the required high operating speed. We will now consider the most common techniques.

Static thresholding

The simplest technique is static thresholding, where the return signal is compared to a constant threshold [71]. A target detection is recorded when the signal passes this threshold. The advantage is the simplicity, but the disadvantage is that the method is unreliable in the presence of volumetric backscatter. In that case, backscatter from scattering media close-by will be brighter than returns from faraway legitimate targets, due to the attenuation of light over distance. In air, this usually is not a problem, since it is a near-perfect medium. However, bad weather such as mist can interfere with this detection method [78].

[1] A square is often used for simplicity, but the input signal is typically thresholded, such that the square wave can be regarded as the binary equivalent of a sine wave.

Adaptive thresholding

The susceptibility of static thresholding implementations to volumetric backscatter can be overcome by adapting the threshold over time or based on the returned signal. Examples include Constant False Alarm Rate (CFAR) detectors, which adapt the threshold according to a function of the background surrounding the sample and also more complex Bayesian methods [47]. The main advantage is lower susceptibility to volumetric backscatter, but the approaches are noncausal and may require sophisticated processing (Bayesian methods). This either requires expensive full digitisation or complex analog filtering.

2.5.4. Digitisation

The incoming light is converted to an analog electrical signal, but most LiDAR systems provide their ToF estimates digitally. Hence the signal must be digitised in order for the digital part of the system to properly process the information. Different types of digitisation techniques were considered for the design of the LiDAR and will be discussed in the forthcoming sections.

Analog to Digital Converter (ADC)

An ADC (as the name entails) fully converts the analog signal to a proper digital signal. They are often used in systems which fully digitise the light intensity signal. An advantage of ADC based implementations is the lower analog complexity, as complex signal processing can be implemented in software. Another advantage of an ADC implementation is the potential of acquiring volumetric density information about the 3D space, which could be of interest to the user. However, due to the high bandwidth required for LiDAR applications, sufficiently capable ADCs tend to be expensive. Also, the dynamic range of these high-speed ADCs is often insufficient for LiDAR applications, so the input dynamic range must be compressed using either nonlinear circuits [7] or time-variant amplifiers [3].

Time to Digital Converter (TDC)

Time to Digital Converter (TDC) is a device which reads out a digitised time interval between two rising edges [10]. Advantage of a TDC are the conceptual simplicity and low cost. The disadvantage is that a TDC based implementation forces signal processing to be implemented in analog electronics, increasing complexity. Another disadvantage is that the output data is only the distance to a target and thus no other information about the channel can be acquired such as target strength or multiple returns.

Phase measurement

In phase measurement digitisation, the time of flight is found by comparing the accumulated electrical of the normalised received pulse while the system was transmitting Q_1 and the accumulated normalised received electrical charge while the system is not transmitting Q_2. From these, the distance is calculated using equation 2.1.

$$d = 0.5 \cdot c \cdot \Delta t \cdot \left(\frac{Q_1}{Q_1 + Q_2} \right) \tag{2.1}$$

The advantage of this method is the simplicity, and it is often used in ToF cameras [31]. The major disadvantage is the susceptibility to volumetric backscatter. The returned pulse is required to be either fully on or fully off for processing, which entails thresholding. However this threshold is difficult to define in the presence of volumetric backscatter, as explained previously.

2.5.5. Design Option Tree (DOT)

We will now present a short visual summary of the high-level design options for LiDAR systems discussed so far in the DOT shown in figure 2.5. This will serve as an accessible reference for the upcoming concept selection.

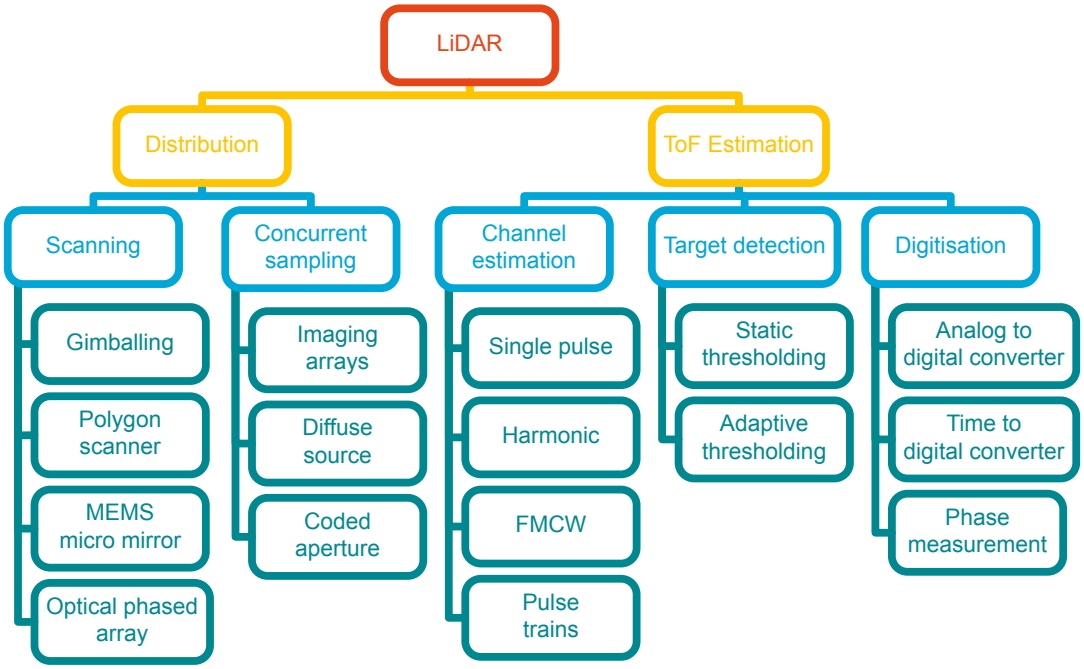


Figure 2.5: High-level LiDAR DOT.

2.6. Concept selection

In this section, all of the discussed design concepts for each part of the LiDAR system are weighed in upon. Given the criteria for each part, a decision is then made as to which design concept is most suitable for each part of the LiDAR. In the forthcoming sections, the decision for each part of the LiDAR is discussed.

2.6.1. Distribution

Table 2.2 illustrates the advantages of the different distribution methods. Based on the criteria given in the table, it was decided to implement a polygon scanner for the distribution system, since we only considered its implementation to be time feasible.

Table 2.2: Trade-off table for the distribution

	Distribution							Explanation
	Gimball	Polygon	MEMS	OPA	Array	Diffuse	Coded	
Size	±	-	+	+	+	+	±	Size should comply to requirement PER13-A
Fluid Function	±	±	-	±	+	+	-	The system will be fluid-filled as per ENV1-a and ENV1-b
Implementation time	±	+	±	-	±	±	-	Development time is limited, as per PRO1

2.6.2. Channel estimation

Table illustrates 2.3 channel estimation trade-offs. We decided to use the single pulse technique based on since it had the best feasibility per PRO1 and PRO3 at an acceptable range to satisfy PER1-a and PER1-b.

Table 2.3: Trade-off table for the channel estimation method

	Channel estimation				Explanation
	Single	Harmonic	FMCW	Pulse train	
Modulation	+	+	-	±	Simplicity of the modulation technique
Detection	+	±	-	±	Complexity of the detection method
Range	±	-	+	+	Maximum range that can be achieved

2.6.3. Digitisation method

For the digitisation method, it is of utmost importance to minimise the noise in the received signal and implementation complexity. For the options that were considered for the LiDAR, it is important to discuss the quality of information versus the complexity of the implementation. Table 2.4 illustrates the advantages for each digitisation method. In order to maintain feasibility as per PRO1 and PRO3 at the decision was made to implement a TDC for the digitisation since it can be validated the quickest.

Table 2.4: Trade-off table for the digitisation technique

	Digitisation			Explanation
	ADC	TDC	Phase	
Complexity	±	-	-	Complexity of the implementation
Flexibility	+	-	-	Flexibility of the implementation with the other devices
Time to prototype	-	+	-	Time to first test of the simplest iteration

2.6.4. Target detection

When choosing a target detection technique it is of importance to keep in mind that the operating environment is sea water. The trade-offs between static and adaptive target detection can be found in table 2.5. Though the adative target detection is very complex, its resilience against backscatter is of the utmost importance. Since static detection would give too many spurious detections to meet PER9-a and PER9-b, it was decided to implement adaptive detection, despite the additional complexity, possibly interfering with PRO1 and PRO3.

Table 2.5: Trade-off table for the target detection techniques

	Target detection		Explanation
	Static	Adaptive	
Complexity	+	-	Complexity of the implementation
Flexibility	±	±	Flexibility of the implementation with the other devices
Reliability	-	+	Reliability in the presence of close-by scattering media

2.7. Subsystem selection

As described in the overview of a simple LiDAR in section 2, a transmit stage consisting of a laser diode together with a laser driver are necessary to convert an arbitrary timing signal to a laser pulse that leaves the system. Also, a receiving stage consisting of a photo diode, together with the necessary data processing was needed to in fact retrieve the light and process the data. The focusing of the beam is not considered in the scope of the electrical part of the SLiDAR because this is mainly a mechanical issue. The system described in section 2 does not have any form of beam steering and can thus only measure one point. However, the SLiDAR has to measure multiple points in space, thus some sort of beam steering has to be implemented. Furthermore, The data has to be processed and all actions, such as choosing a point in space to measure the distance or sending out a laser pulse, have to be coordinated. This is why we came to the following subsystem division.

- A transmitting stage
- A receiving stage
- A beam Steering stage
- A Data processing stage

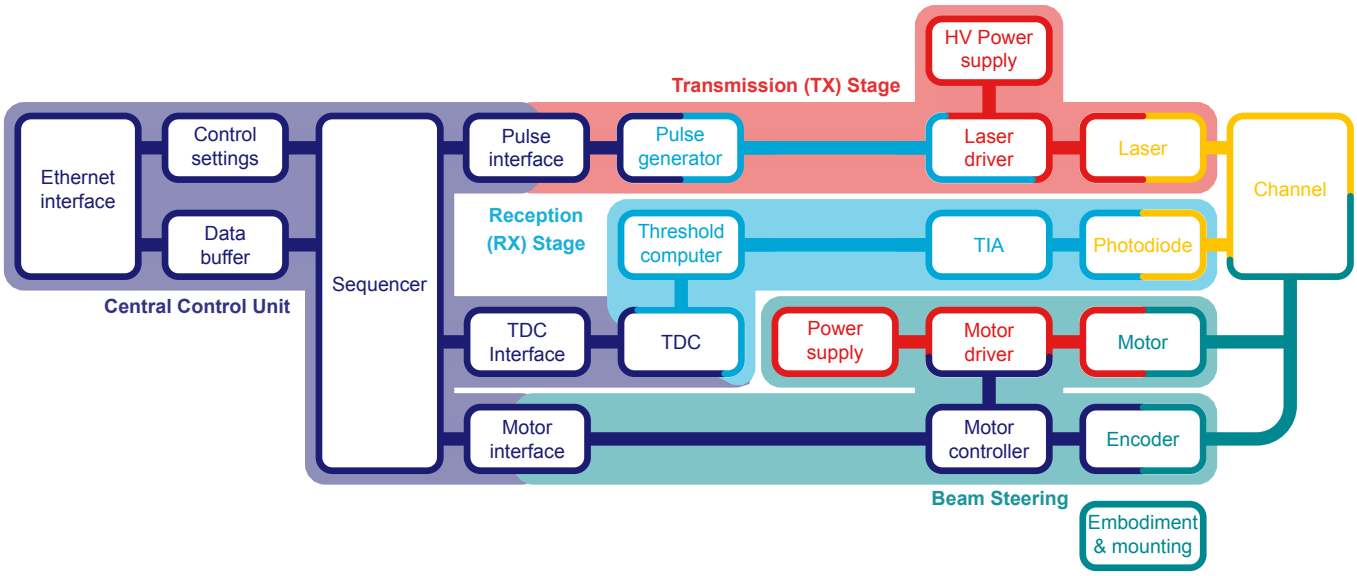


Figure 2.6: Low-level system architecture. Digital is purple, analog signal processing is cyan, high power is red, optical is yellow and mechanical is green. Modules converting domains are a mix of colours.

2.8. System modelling

The subsystem division generates requirements on the interfaces of subsystems. Two important interfaces is the interface of the transmission stage to the optical channel and the channel to the reception stage. These take the form of the optical output power of the transmission stage, the input sensitivity and input referred noise of the reception stage, the wavelength and the bandwidth of both subsystems. Together, these determine the minimum range, maximum range and distance resolution. In order to dimension these subsystem requirements, their relation to the system requirements needs to be found. In this section, we construct a simplified model of the optical channel to find this relation.

2.8.1. Towards a simple model

We have defined a LiDAR device as an apparatus which measures the distance to objects by estimating the optical time of flight to those objects. Then a LiDAR device detects features of its environment. We model the environment as two scalar fields:

- A transmissivity field $T\left(\vec{R}\right)$, which models the amount of light which is not absorbed or reflected when a beam travels through a medium, such that the forward intensity $\lim_{\vec{\Delta\ell}\to\vec{0}} I\left(\vec{R}+\vec{\Delta\ell}\right) = I\left(\vec{R}\right)\cdot T^{\left|\vec{\Delta l}\right|}$. It must always have a value between 0 and 1, since we assume that there are no energy sources to amplify the light in the environment.
- A reflectivity field $\beta\left(\vec{R}\right)$, which models the amount of light reflected back at the source at each point, such that the magnitude of the backscattered light is $I_b = I\cdot(1-T)\cdot\beta$. It must also always have a value between 0 and 1.

LiDAR devices can detect only these fields. This is accomplished by characterising one or a variety of optical paths through the medium and applying channel estimation techniques. Other models model the scattering of the beam in space using distribution function [41]. Assuming that multiple backscatter effects are insignificant and using the fact that our system has a coaxial receiver and transmitter, only the amount of light propagating to the next distance segment needs to be considered, and the simplified model suffices.

Sometimes knowledge of these scalar fields is enough. Cloud observations would be complete with complete knowledge of these scalar fields. In other cases, distances to interesting objects must be inferred from these fields. In

SLAM applications, one is interested in solid obstacles. Most are characterised by $T \approx 0$[2].

Then LiDAR devices must do two things:

- Channel estimation
- Obstacle detection

To constrain the key system parameters, We will now model the channel and the optical signal therein in the context of obstacle detection.

2.8.2. Channel impulse response

For a perfectly collimated beam travelling between points $\vec{a}$ and $\vec{b}$ the intensity I:

$$I\left(\vec{b}\right) = I\left(\vec{a}\right) \cdot \exp\left(\int_{\vec{a}}^{\vec{b}} \ln\left(T(\vec{R})\right) \cdot \left|\mathrm{d}\vec{R}\right|\right)^2 \tag{2.2}$$

We define the magnitude of light reflected back at each position as:

$$I_b\left(\vec{R}\right) = I\left(\vec{R}\right) \cdot \left(1 - T\left(\vec{R}\right)\right) \cdot \beta\left(\vec{R}\right) \tag{2.3}$$

Incorporating x^2 spreading back from the target to the receiver for large x, modelling optical inefficiencies using a factor η, assuming lambertian reflection and assuming that the full beam is reflected we find for the impulse response in terms of power:

$$p(x) = \eta \cdot \exp\left(\int_0^{x/2} \ln\left(T(\tau)\right)\, \mathrm{d}\tau\right)^2 \cdot \left(1 - T\left(x/2\right)\right) \cdot \beta\left(x/2\right) \cdot \frac{A_0}{\left(x/2\right)^2} \tag{2.4}$$

In which $x = u_p \cdot t$, A_0 is the effective receiver area.

2.8.3. Towards wavelength requirements

At longer distances, the exponential term in equation 2.4 modelling bulk absorption will start to play a dominant role. The absorption is highly dependent on the wavelength. Hence, a wavelength should be selected such that absorption is minimised. The absorption in different types of sea water is very well classified by Jerlov's water mass classification [43]. This shows that absorption is maximised for a wavelength around $520\,\mathrm{nm}$ in type I, II and III oceanic waters. Traditionally, this wavelength would be avoided due to the significant solar irradiance causing interference, but this light does not penetrate to the deep sea. We have detailed this consideration as requirement TX-PER1 in table 2.6.

2.8.4. Towards power & input sensitivity requirements

In order to find the required power and input sensitivity, we modelled a maximally absorptive homogeneous channel using $T = 95\%$ as per PER1-a and PER1-b. We assume that volumetric backscatter is caused by floating particulates of the same material as sediments covering the sea floor, such that $\beta(x) = \beta$ is everywhere constant at a minimum value of 0.1 based the darkest sediments in underwater reflectance measurements in [79].

A main factor in the attenuation of the transmitted signal, are the optical elements. Firstly lenses always have a small inefficiency, this has to be taken into account. Secondly, focusing the outgoing laser light on the photo detector also greatly influences the received optical power. The mechanical design therefore already introduces some losses in the optical path.

We implemented a Matlab script with our model to find the range associated with input threshold and output power values. We used values given in the reference design [71] as a basis for the the parameters and adjusted to find a reasonable division of requirements, given in table 2.6. The Matlab script can be found in B.

2.8.5. Towards bandwidth requirements

The bandwidth is directly related to the outgoing pulse width, so determining the pulse width fully constrains the bandwidth. We will now investigate the effects of changing the pulse width on the channel estimate to find the required pulse width.

[2]Glass is an exception, but it is rare in underwater LiDAR applications

In general range finding, the pulse width determines the distance resolution of the system. If objects are closer to each other than the pulse width, they become indistinguishable. Because of the small beam diameters of LiDAR systems, the assumption is often made that all light is reflected by a singular target, such that there is only a single return. However, the backscatter also generates weak returns. Since these returns are distributed over space, the signals ranging between those on the front of the pulse reflected from farther away and the signals on the back of the pulse reflected from shorter away will reach the receiver simultaneously. This generates interference, which scales with pulse width and must be limited. We will now use our simplified model to express this interference in terms of the pulse width of the subsystems to constrain the pulse width.

To find estimate the maximum interference versus distance, we assume clear water before that distance and the most absorptive water thereafter. To simplify calculations, we also ignore the absorption within the length of the channel occupied by the pulse, which will lead to slight overestimation. We can then remove the exponential term from equation 2.4 and correlate with a square pulse to find the maximum interference versus distance in equation 2.5.

$$p_i(x) = 4\eta\beta A_0 \cdot (1 - T_{min}) \cdot \left(\frac{1}{x - L} - \frac{1}{x} \right) \tag{2.5}$$

Any detector should not register a detection at this maximum interference level. This puts a lower limit on the threshold. We can express this by modelling the maximum factor α that the threshold can be lower to maximum return. This is obtained when $T = 0$ after clearwater, such that bulk absorption can be ignored. This yields equation 2.6 for $x > L$.

$$p_{max}(x) = 4\eta\beta A_0 \cdot \frac{1}{x^2} \tag{2.6}$$

Since $p_i(x)$ falls superquadratically for $x \to \infty$, there is a range where $p_i(x_{min}) = p_{max}(x_{min})/\alpha$. Below this range, it can not be guaranteed whether an obstacle is detected or volumetric backscatter. Solving the equation yields equation 2.7.

$$L = \frac{x_{min}}{(\alpha \cdot (1 - T_{min}) \cdot x_{min} + 1)} \tag{2.7}$$

We assume that $\alpha = 6$ is a reasonable value, $T_{min} = 0.8$, and $x_{min} = 1\,\text{m}$ per PER2-a or $x_{min} = 0.05\,\text{m}$ per PER2-b. This gives a pulse width $L = 0.45\,\text{m}$ per PER2-a, which corresponds to $2\,\text{ns}$ and a pulse width of $L = 0.05\,\text{m}$ per $0.2\,\text{ns}$ PER2-b. This corresponds with bandwidths of $0.5\,\text{GHz}$ and $5\,\text{GHz}$ respectively. These requirements are detailed in table 2.6.

Table 2.6: Subsystem requirements

ID	Requirement	Verification				Justification
		Test	Demonstration	Analysis	Inspection	
TX-PER1	Light emitted by the transmission stage shall have a wavelength between 505 and 535 nm .		•			Minimises absorbance while allowing slight deviations due to changing operating conditions.
TX-PER2	The optical power emitted by the transmission stage shall have a value higher than 6 W	•				In order to meet the range specifications.
TX-PER3a	The pulse width of the light emitted by the transmission should remain smaller than 2 ns		•			In order to limit the interference.
TX-PER3b	The pulse width of the light emitted by the transmission should remain smaller than 0.2 ns		•			In order to limit the interference.
RX-PER1	The threshold of the RX stage shall be less than 0.05 W.			•		In order to detect at the required range.
RX-PER2a	The bandwidth of the RX stage shall be at least 500 MHz			•		In order to limit the interference.
RX-PER2b	The bandwidth of the RX stage should be at least 5 GHz			•		In order to limit the interference.

2.9. Functional Overview

This section serves as a brief overview of the functional operation flow of the proposed LiDAR design from section 2.7. From this overview, the interfaces between subsystems are defined. The reader can use this section to put the design of the subsystems in following chapters into the perspective of the LiDAR system as a whole.

2.9.1. Functional flow

A high-level functional overview of a single distance measurement of the proposed LiDAR design is given in figure 2.7. The motor in the diagram rotates the polygon mirror. The direction in which the laser pulse is sent depends on the angular position of this mirror. The angular position is most accurately known when the mirror rotates with a steady-state angular velocity. Hence, the CCU waits for steady-state motor rotation before measurements start.

A distance measurement consists of a number of sequential events. The CCU instructs the Transmission (TX) Stage to send a laser pulse to the measurement channel. The timestamp of this event is recorded by the TDC. A laser reflection comes back from the channel, and is detected by the APD. The Reception (RX) stage converts this detection to a pulse, of which the TDC records the timestamp. The CCU computes a distance estimate from the two TDC timestamps, and streams it to the user via a serial data protocol.

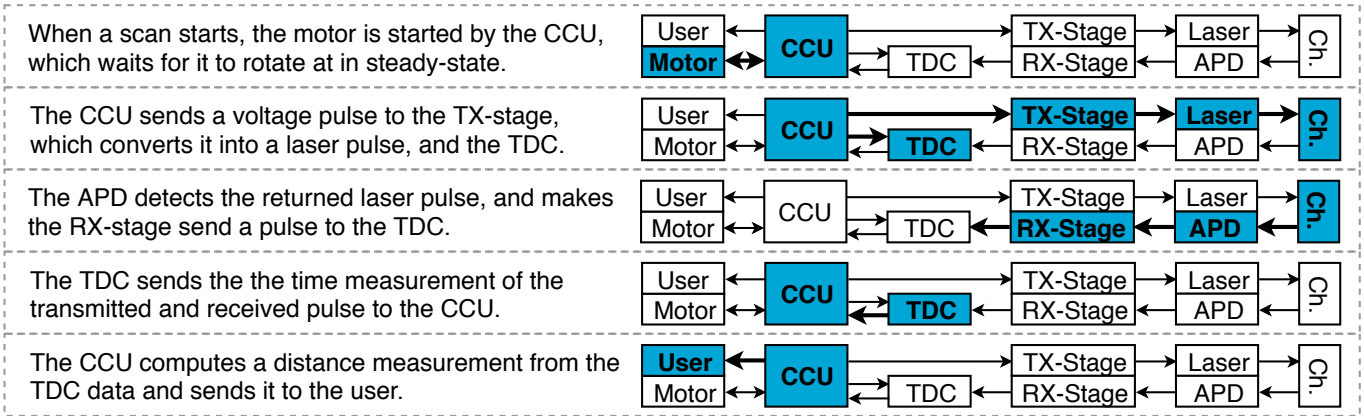


Figure 2.7: High-level functional overview of a single distance measurement. Active parts of the system during a step are highlighted.

Laser pulses are transmitted in packets called frames. The relation between the timing of frame transmission and the rotation of the mirror is illustrated in figure 2.8. When the line of sight of the laser diode crosses a corner of the mirror, the Beam Steering system will detect this as an FOV-corner event. When this detected corner marks the start of the FOV, it is called an FOV-start event. The CCU uses these events to time the laser pulse transmission of the Transmission Stage. The red areas in the figure indicate dead zones. These are margins within which the laser might hit a mirror corner, which would make the measurement invalid. The CCU will not send.

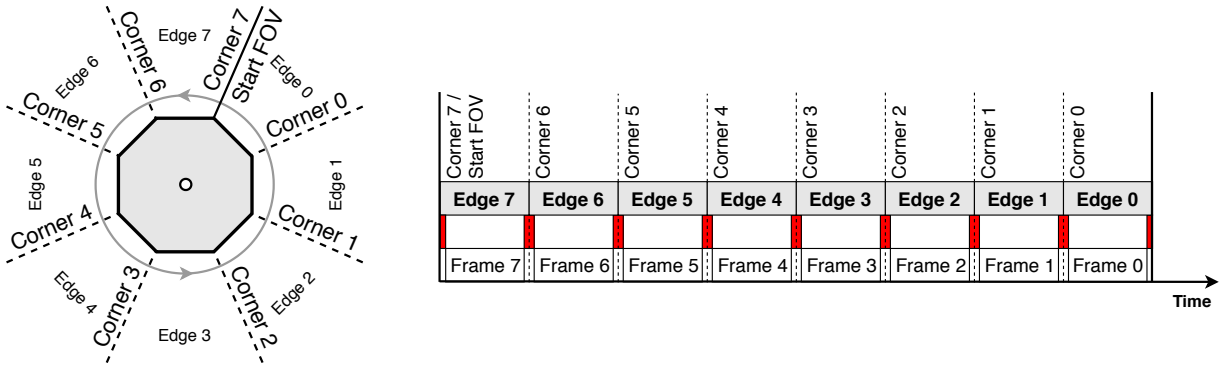


Figure 2.8: Illustration of the correspondence between polygon mirror rotation and pulse timing.

2.9.2. Interface specification

The high-level functional overview of each subsystem (see figure 2.6) can be derived from the discussion in section 2.9. Table 2.7 shows the data flow between subsystems. This table assumes the use of the AS6501 TDC IC.

Table 2.7: Data flow between subsystems from figure 2.6.

From:	To: CCU	Transm. Stage	Reception Stage	Beam Steering
CCU	None	• Pulse frames • Reference signals for the laser driver	• Control signals for the threshold computer • Oscillator to aid in generation of bias signal for the Avalanche photodiode (APD) • Configuration and reference signals for the TDC • Pulse frame triggers	• Motor start and stop signals
Transm. Stage	• Serial LVDS TDC timestamp data (2 channels)	None	None	None
Beam Steering	• Indication if motor rotates in steady-state • FOV-start and FOV-corner event detections	None	None	None

3

Design Decisions

In chapter 2, it became apparent that a module has to be designed that is capable of directing a laser beam. This chapter, will discuss the design decisions that must be taken in order to create such a module.
The beam steering will require a motor to create some motion. Therefore, a discussion on different motor types and how to drive them will be given. As for the transmission and reception stage, it is of importance to know the orientation of the motor, a discussion will be held on different ways to determine this. Lastly, the motor driver has to be linked to the rest of the system. How this interface will be created shall also be discussed.

3.1. Motor Types

There are three types of motors to choose from when working with DC power(requirement INF1). These are:

- Brushed DC motors [25]
- Brushless Direct Current (BLDC) [23, 28]
- Stepper motors [24]

To decide which motor can best be used, the application has to be considered. The rotational frequency of the polygon determines the refresh rate of the system. A steady refresh rate is desired, as this makes it easier to integrate the whole SLiDAR in a final system(i.e. LOBSTER) by means of software. Also, a steady speed within a single rotation is desired. This makes for a constant frequency of the laser pulses. To achieve this, an as low as possible torque ripple is needed. A ripple in the torque means a ripple in the force applied to rotate the polygon. This in turn leads to a variability in speed.
An extra challenge is that this has to be performed under high pressure. The high pressure fluid in which the system is submerged will amplify the torque ripple effect due to the higher drag. Therefore, it is of the utmost importance to minimise the torque ripple.

Brushed DC motors can simply operate from a DC voltage source, without any electrical commutation required as this is done mechanically through the brushes. Though brushed DC motors have been extensively used in the past for their good and simple position and speed control, nowadays they are mostly replaced by brushless motors. The reason for brushed motors being replaced is the fact that they require a lot of maintenance due to wear in the brushes. This wear in the brushes causes reliability and lifetime to decrease. These effects increase for tough conditions like high pressure.

A BLDC on the other hand, does not have any brushed contacts and, therefore, has a longer lifetime and lower susceptibility to external conditions such as the high pressure. This does, however, mean that commutation has to be done electrically, resulting in more complex driving circuits. Besides the robustness of this type of motor, another advantage is that it is very well suited for constant speed applications due to its similarities with permanent magnet synchronous motors and it is also very power efficient.

The last motor type to consider is the stepper motor. Like a BLDC, a stepper motor is also brushless, making it a robust solution as well. In fact one might even consider the stepper motor a BLDC motor with a large number of poles. The strength of a stepper motor is the possibility of position control without a feedback loop. This is done by stepping the motor from pole to pole by switching the electric current from pole to pole. Though this is an easy way of controlling the position, it sometimes occurs that steps are missed, so for more accurate position control, a missed step detector has to be created. Though position control is easily implemented for a stepper, running the motor at constant speeds is actually much harder, since this would require smooth transitions between all steps.

The most important characteristics of the three motor types above, are summarised in table 3.1. One of the key features of SLiDAR is its pressure tolerance. Therefore, it was decided not to use a brushed DC motor, since it is very much doubted that it will work at all when exposed to the high pressures of the deepsea. Also, a stepper motor is very well at the specific task of precise position control, however, it is not so much designed for constant speed applications. It was therefore decided that for the SLiDAR, the BLDC motor is the best option. Though a driver is harder to design, it offers good performance at constant speeds with little ripple and high efficiency.

Table 3.1: Comparison of relevant parameters of the different parameters.

Motor	Commutation	Lifetime	Drive complexity	Strengths
Brushed DC	Mechanically	Shorter	Very low	Simplicity, Speed/Position control
Brushless Direct Current (BLDC)	Electrically	Longer	High	Constant speed, High efficiency, Robustness
Stepper Motor	Electrically	Longer	Medium/High	Open loop position control, Robustness

3.2. Motor Driving Techniques

For different motor types, different control techniques are available. For the BLDC type of motor, the three most commonly used techniques are [26]:

- Direct commutation
- Direct Torque Control (DTC)
- Field Oriented Control (FOC)

Before discussing the three driving techniques, it is good to first have a quick look at the motor itself. A BLDC motor consists of a rotor and a stator. At the rotor, usually permanent magnets are present whereas the stator carries field windings for three different phases. Usually the motor is driven by properly energising the motor phases, where current is fed into one phase and leaving through another with the third phase being idle. The magnets on the motors rotor will try to align with the magnetic field created in the energised coils. Properly switching the current between the phases will cause the motor to rotate. In order to time the switching of the current such that the motor will rotate, it is common to use hall effect sensors that will give some feedback on the position of the rotor. However, still different techniques are available for commutation.

Direct commutation, as the name suggest directly controls the motor phases (usually three phases). A form of direct commutation is six-step commutation. The phase excitations for six-step commutation are based on the outputs of three digital hall sensors (for three phase motors). The exact scheme can be found in figure 3.1. From this scheme it can be deduced that quite some ripple in the torque and, therefore, also in the speed should be expected due to the sudden phase changes. What happens is that at every phase change the voltage suddenly changes from low to high or the other way around. This can be viewed as the motor getting a 'kick' at every phase transition, instead of the transition being smooth. Therefore, direct commutation can best be used for low-performance applications where simplicity is key. To improve the performance another possibility is to use sinusoidal commutation, where the six-step block waves are replaced by sinusoids. However, creating the proper sinusoids can be really challenging.

Besides direct commutation, there are also two other commonly used driving techniques. One of them is Direct Torque Control (DTC), which can directly control the torque. To do so, first the flux linkage is estimated in the α-β frame. In figure 3.2 it is shown that the current, but also flux linkage, of the three phases can also be represented by a two dimensional vector in this frame. Using the transformed voltage and current, and resistance and inductance of the motor phases, the flux and torque can be computed and compared to reference values. Based on this comparison, the best voltage pattern for exciting the three phases is selected using a look-up table. Due to the direct control over the torque, less ripple is produced compared to direct commutation. Furthermore, still high dynamics can be reached since only one simple transform and a few comparators are required.

The last technique considered in this thesis is Field Oriented Control (FOC). FOC is not only capable of controlling the torque, but also of controlling the force component orthogonal to the torque (i.e. the component that does not do any work). For FOC, first the Clarke transform [2] is applied, but also an extra step is required. Instead of just projecting the current vector in the stationary reference frame, a reference frame is chosen that rotates at the same speed of the motor, where one axis represents the current responsible for creating the torque (I_q) and the other current that only creates undesired compression forces (I_d) [2]. The Park transformation is also depicted in figure 3.2. Due to

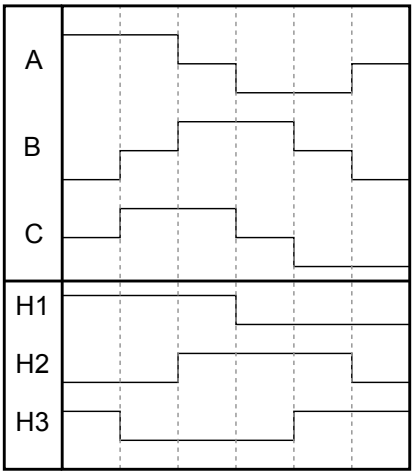


Figure 3.1: Timing diagram for six-step commutation of a BLDC. A, B and C represent the three motor phases and H1, H2 and H3 are the hall effect sensors used for feedback.

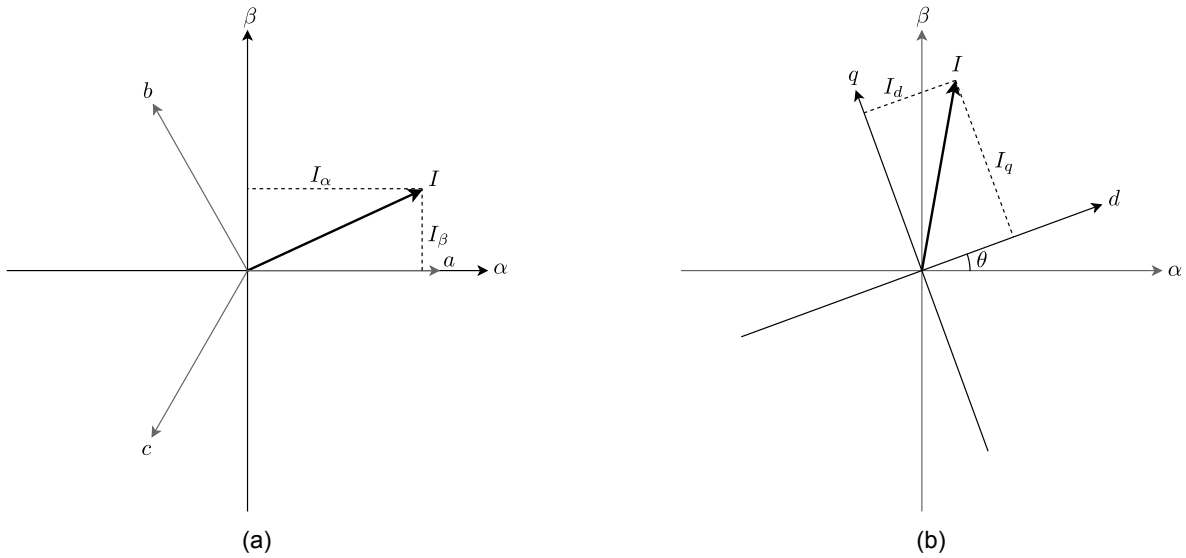


Figure 3.2: The Clarke and Park transform depicted in a figure. The Clarke transform, (a), transforms the phase currents to a two dimensional vector in the α-β frame. The Park transform, (b), transforms the current from the α-β frame to the d-q frame, which rotates along with the rotor of the motor.

the fact that the reference frame now rotates along with the rotor, the torque can be displayed on one axis and the other current component can be tuned to zero. In that case for constant torque, the controller can tune to a constant, an application for which PI controllers provide very good performance. Therefore, the performance of FOC in terms of low torque ripple and high efficiency is very good. On the other hand it might perform slightly worse, compared to DTC, in high dynamic applications due to the complex calculations that have to be performed to switch reference systems.

In choosing a driver, performance and simplicity have to be weighed against each other [67]. In table 3.2 the different characteristics of the three driving techniques can be compared. For an application where constant speed is important, such as SLiDAR, the torque ripple produced by six-step commutation is unacceptable. Though direct commutation can also be performed using sinusoidal signals creating the proper signals is very cumbersome. Therefore, DTC and FOC are the remaining options. Again taking into consideration the application, FOC turns out to be the best option. Though a bit more complex circuitry is required, compared to DTC, it does offer great performance when it comes to stability. Besides this, it also is the most power efficient solution, which is also an advantage given that the SLiDAR will most probably be applied in battery operated Autonomous Underwater Vehicles (AUVs).

3.3. Position Feedback

To properly drive a BLDC motor, always some form of position feedback is required. For SLiDAR position feedback is also required for driving the laser. Since the slightest deviation in angle of the motor will result in a large error at 10 m distance, a very precise feedback is required. This kind of precision is provided by rotary encoders. However,

Table 3.2: Comparison between the different driving techniques for a BLDC motor. The direct commutation refers to the six-step commutation specifically.

Drive technique	Complexity	Torque Ripple	Power efficiency	Dynamics
Direct Commutation	Low	High	Low	High
Direct Torque Control (DTC)	Medium	Low	Medium	High
Field Oriented Control (FOC)	High	Very low	Very High	Medium

different types are available of which one should be chosen. Available encoder types are:

- Resistive
- Magnetic
- Capacitive
- Optical

In the remainder of this section these different encoder types shall be compared to one another.

For a pressure tolerant system, there is actually only one suitable type of encoder. The resistive encoder has the same problem as a brushed DC motor, since it also relies on brushed contacts that will probably not be able to function properly under high pressure. An optical encoder is normally a good decision for applications requiring high accuracy feedback. However, the pressure fluid will very likely disturb the optical path inside the encoder, making it unsuitable for pressure tolerant systems such as SLiDAR. Then, there are only two options left: magnetic or capacitive encoder. The capacitive encoder, is a relatively new type of technology [4]. This means that there is little known about its performance under more extreme conditions. As there were some worries that the dielectric constant of the pressure fluid would influence the performance for the worse, it was decided to opt for the magnetic encoder type.

3.4. Digital Interface

To link the motor to the rest of the SLiDAR, a digital interface will be implemented. For a digital interface some sort of processing unit is needed to coordinate all actions and process the data. To choose the best option, the function of the system has to be considered. The most important function of the system is to rotate the motor at a constant speed. Due to the speed at which the motor is rotating, delays in the digital interface will result in an inaccuracy in the measurement of the angle. For this reason, a high order of parallelism is ideal. This means multiple tasks can be executed at the same time further minimising the delay.

The four main options for a processing unit are:

- A microcontroller
- A microprocessor
- A Field-Programmable Gate Array (FPGA)
- An Application-Specific Integrated Circuit (ASIC)

An ASIC is an integrated circuit that is designed for one specific application. The speed could be as high as modern technology would allow it to be and it could also be designed to be highly parallel. However, once the IC is made, the design cannot be changed. Thus making it unsuitable for rapid prototyping. ASICs are also not the cheapest option by far.

Microcontrollers and microprocessors are quite similar. The main difference is that the microcontroller has peripherals such as memory built-in. This leads to a trade-off between ease of integration and customizability. Their main advantage is that they are very easy to program, since most microcontrollers and microprocessors have extensive support and code libraries supplied by the manufacturer. However, The order of parallelism is fairly low. They can generally only execute one instruction per core at a time.

The FPGA, on the other hand, can be programmed to do a lot of actions at the same time. Just like the microcontroller and microprocessor it can also be programmed fairly easy and, on top of this, it is also very fast. The only drawback is that an FPGA does not contain peripherals and, thus, is harder to integrate.

Because of the speed and high order of parallelism, the FPGA was chosen as the processing unit and interface. The other subsystems also use the FPGA for processing.

4 Implementation

Now that the design decisions have been made, the implementation can be discussed. This chapter will elaborate on the specific parts of the design and how they are linked together. A general overview of the implementation can be seen in figure 4.1. This chapter will discuss this implementation divided into three sections. Section 4.1 that will discuss the motor, encoder and the motor driver that have been chosen. After this, the digital motor control on the FPGA will be discussed in section 4.2. Lastly, the power converter for powering the whole system will briefly be discussed in section 4.3.

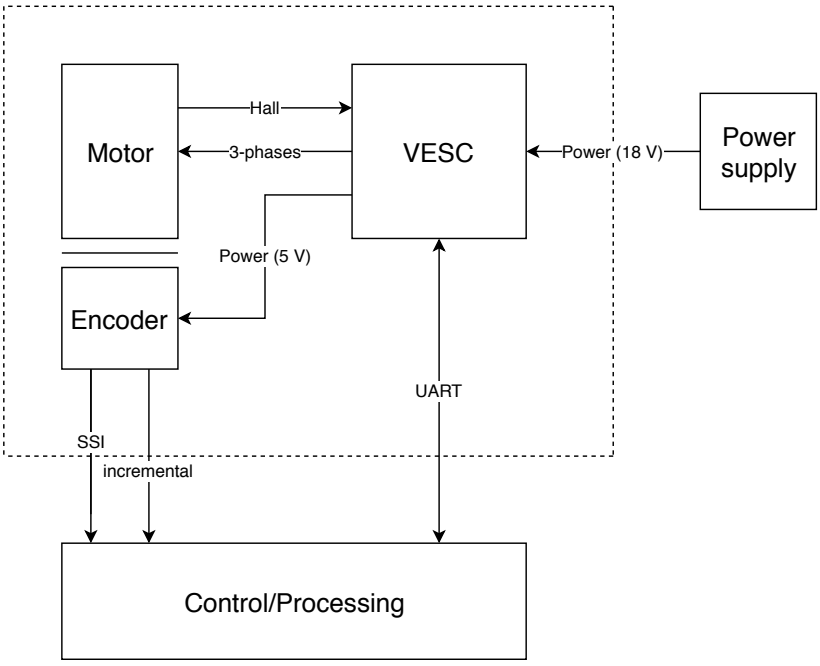


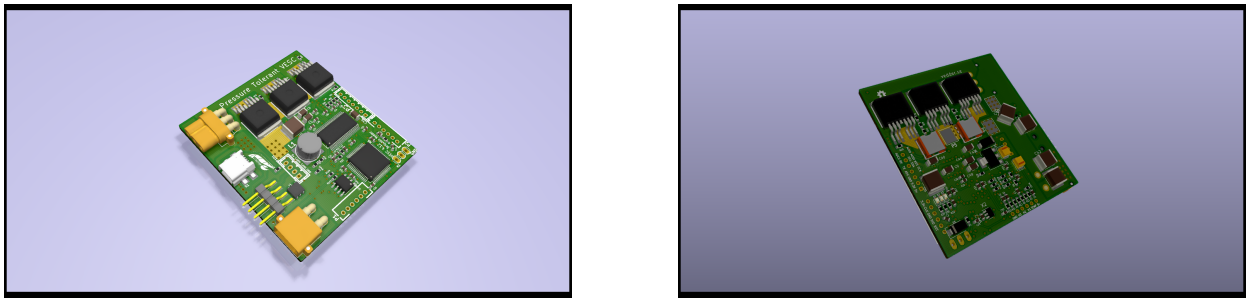
Figure 4.1: Overview of the motor driver subsystem of SLiDAR. The motor and the hardware around it to drive it is encapsulated in the dashed box. Further, also the power supply and the digital control unit have been depicted.

4.1. Motor, Driver and Encoder

As can be read in chapter 3.1, it was decided to use a BLDC motor. When searching for a suitable motor, mostly mechanical requirements had to be taken into consideration. For starters, the motor should be capable of delivering enough torque to spin the motor in a high pressure fluid. Furthermore, the motor should be able to rotate fast enough to meet Requirement PER11-b. Along with this, from an electrical point of view, the requirement was set that Hall sensors should be built-in in the motor. FOC enabled drives often require Hall sensors. So, to ensure proper operation of the driver, this requirement was set. Besides this, the motor should be of high quality in the sense that it is able to run smoothly. This in order to be able to at least meet requirement PER9-b. After some research, the *Maxon EC-i 40* was found to tick all of these boxes and chosen to be implemented in the system [39].

Alongside with the motor, a suitable driver is required. In chapter 3.2 it was already established that this driver should

use FOC as a control method. However, to design a complete FOC driver within ten weeks, along with the rest of the system seemed undoable. It was, therefore, decided to buy an off-the-shelf motor driver. The motor driver had to be pressure tolerant to comply with requirement ENV1-a and ENV1-b. However most systems used non-pressure tolerant components that are cheaper than its pressure tolerant counterparts. Luckily there was one FOC enabled driver that is completely open-source. This is the Vedder Electronic Speed Control (VESC) [75]. The VESC is a software based speed controller solution named after its creator *Benjamin Vedder*. The reference design that is created by *Vedder* uses components that do not have a high pressure tolerance. But, seeing as the documentation is open-source, a custom design was created. The footprint of the PCB was changed for easier integration and the pressure intolerant components were swapped out for pressure tolerant ones. These pressure tolerant components do not contain air or gas so that they do not implode under the high pressures. Once the VESC has been made pressure tolerant, the motor parameter detection can be run, after which the motor itself can be run in FOC mode. The VESC also has a configurable current limit that can be set, which was done to ensure the power usage of the motor to stay within the limits set by requirement INF2. To control the motor from the FPGA, the VESC comes with a Universal Asynchronous Receiver-Transmitter (UART) connection, of which the technical details will further be discussed in section 4.2.



(a) (b)

Figure 4.2: A render of the pressure tolerant VESC

Finally, a magnetic encoder has to be chosen. Generally speaking, there is a trade off between cost and performance, with performance being resolution and accuracy. To meet the $0.1°$ requirement PER4-a, really expensive encoders were needed. This is why a little less accurate but much more affordable encoder was opted for. However, the encoder should still meet requirement PER4-b.
A manufacturer, *RLS* [54], reached out to us willing to sponsor one of their products, a *RMF44* encoder. All encoders in the *RMF44* series are within the $1°$ accuracy requirement PER4-b. Because they are sponsored, money was left in the budget to spend on other parts. The products are listed in table 4.1.

Table 4.1: The magnetic encoders offered by RLS

Name	Method of data retrival	
RMF44AC	Analogue sinusoidal	
RMF44IE	Incremental	Open collector
RMF44IC	Incremental	
RMF44SC	Synchronous Serial Interface (SSI)	
RMF44SI	SSI + Incremental	
RMF44MD	Analogue sinusoidal + SSI + Incremental	Has slower readout speed than other options.
RMF44Vx	Linear voltage	Lower resolution than other options.

There are a lot of encoders to choose from. A first restriction is that the encoders are read by an FPGA, meaning that an analog output or open collectors are not ideal as it would only be harder to integrate. So there are four options left: *RMF44IC*, *RMF44SC*, *RMF44SI* and *RMF44MD*. The *RMF44SI* was chosen because the lower readout speed of the *RMF44MD* only adds a limitation. Furthermore, the *RMF44SI* has the functionality of the *RMF44IC* and *RMF44SC* combined while not introducing any consequences.

4.2. Digital Motor Control

To coordinate all actions with respect to the motor and process the data, a digital motor control was implemented. As stated in chapter 3.4, an FPGA was used. Because of this, the whole digital structure could be split up and designed in parallel. An overview of the system and its nodes can be seen in figure 4.3. The system was divided into six nodes.

- Motor interface

- UART
- Position
- Speed
- Trigger
- Control FSM

Together, these nodes are able to communicate with the VESC and the rest of the SLiDAR system. Briefly explained: The *motor interface* takes care of interfacing the digital system of the motor with the rest of the digital systems. The *UART* sends commands to the VESC. The *Position* and *Speed* node takes care of tracking the position and speed of the shaft respectively. The *Trigger* node sends out flags based on the rotation of the shaft for the rest of the system. And, lastly, The *Control FSM* node takes care of monitoring and controlling all the other nodes. In the remainder of this section a more detailed description of all entities shall be given.

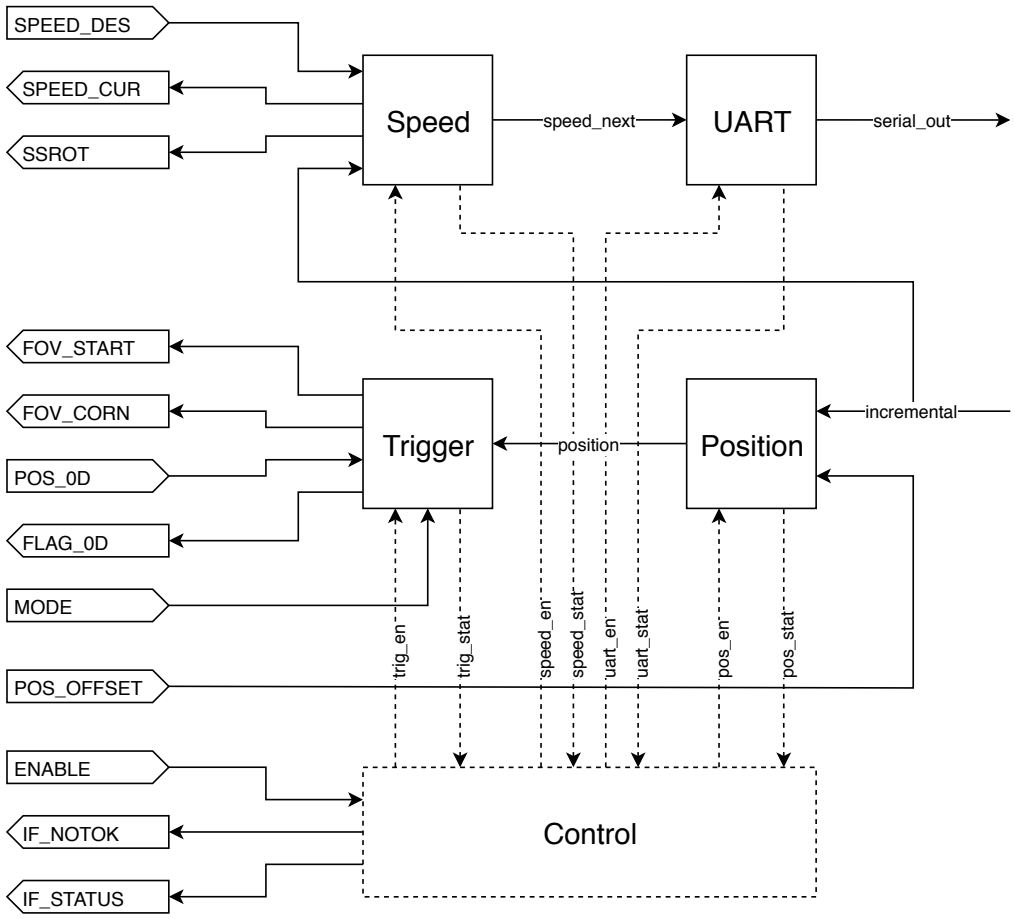


Figure 4.3: Overview of the digital design of the system. On the left all the interface signals are given. The squares are the different entities making up the digital motor control. All dashed lines are part of the control structure of the rest of the motor digital design.

4.2.1. Motor interface

The motor interface takes care of the communication with the rest of the SLiDAR subsystems. It does this by buffering eight signals that follow a specific protocol. These signals can be seen on the left side of figure 4.3. A detailed description about these signals can be seen in table 4.2.

4.2.2. UART

As stated in chapter 4.1, the VESC uses a UART bus to communicate with the rest of the system. Here, the implementation of the UART interface on the FPGA will be discussed, but first a more detailed description of the protocol used by the VESC shall be given.

Table 4.2: A description of the motor interface signals

Signal	I/O	Description
SPEED_DES	IN	The desired speed for the motor to rotate at
SPEED_CUR	IN	The speed at which the motor is currently rotating
SSROT	OUT	A flag determining if the motor rotates at the desired speed
POS_OFFSET	IN	A configurable offset to compensate for the angular offset of the mirror
FOV_START	OUT	A flag that goes high if the polygon is facing the laser with one specific edge of the polygon, indicating one full rotation of the polygon
FOV_CORN	OUT	A flag that goes high if the polygon is facing the laser with one of its edges
POS_0D	IN	The desired rotational angle of the polygon to pulse in 0D mode
FLAG_0D	OUT	A flag that goes high if the position of the polygon matches POS_0D
IF_STATUS	OUT	A vector containing the status of all nodes
IF_NOTOK	OUT	A flag that goes high if the motor has to stop spinning immediately(e.g. a fatal error). This flag does not influence the speed in any way but is here to let the rest of the system know that the motor has been stopped
ENABLE	IN	An enable flag for the whole motordriver subsystem
MODE	IN	A 2 bit vector that determines the operating modes of the system. The Most Significant Bit (MSB) is for the 0D mode and the Least Significant Bit (LSB) is for continuous mode. Setting one of these two bits high will enable the respective mode

The UART protocol

UART is an asynchronous communication protocol, meaning no external clock is needed. It also is a protocol where data can be sent and received at the same time. For data to be successfully sent and received, the UART clocks of both devices, the so called baudrate, have to be the same. In the case of the VESC, the baudrate is fixed at 115 200 Hz, a common baudrate in UART applications. A data line is initially held high. To start a transfer, the data line is pulled low for one clock cycle. This is to let the receiver know data is coming. The format of data being transmitted differs from system to system. In the case of the VESC, one byte is transferred at a time, starting with the LSB. If the bit is high, the data line will be held high for one clock cycle, and when the bit is low it will be pulled low. After the MSB the data line is pulled high for one clock cycle to indicate the end of the byte. Not a lot of data can be transferred with one byte, because of these bytes being sent sequentially in a specific format to generate a message. The general overview of a message is as follows [15]:

1. One Start byte
2. One or two bytes specifying the payload length
3. The payload of the packet
4. Two bytes with a Cyclic Redundancy Check (CRC) checksum on the payload
5. One stop byte

Start byte
The first byte is a start byte, telling the VESC a message is coming. In most general cases this is a 2 in binary. However, for messages with a payload longer than 256 bytes a 3 is used.

Payload length
If the start bit is a 2, only 1 byte is expected. If a 3 is used for the start bit, 2 bytes are expected. This is where the described limit of 256 comes from, the maximum unsigned number for 1 byte is 256.

Payload
The payload contains all the data that has to be send to the VESC to change a setting or request data. The payload also has a fixed format that has to be used. First a 1 byte identifier is used to define the action. Then, if metadata is needed, it is sent next. Metadata differs from action to action. For example, if a current limit is set, an integer is needed. However, if data is requested, an identifier is generally enough. After an identifier and its metadata are sent, a new action may be defined.

Cyclic Redundancy Check
To check the integrity of the data, a Cyclic Redundancy Check (CRC)[56] is implemented. In this case the CRC is always 16 bits or 2 bytes.

Stop byte
The stop byte defines the end of a complete message, this is always 1 byte with the decimal value of 3.

Example
As an example, a new value for the speed will be set and the current draw of the motor is requested. Firstly, the payload is determined. The identifier for setting a speed is 8. The speed has an integer as metadata, the desired speed. To request general data such as the current draw, the identifier 4 is used. This operation uses no metadata. The total number of bytes in the payload is 6, this means only 1 byte is needed to determine the payload length, so a startbyte of 2 is used. After the payload follows the CRC and the stop byte, these are always 2 and 1 byte respectively.

Table 4.3: The final message, values are defined in decimal, send from left to right

Start Byte	Payload length	Speed ID	Speed as integer				Current draw ID	CRC		Stop byte
2	6	8	MSB	...	...	LSB	4	MSB	LSB	3

UART implementation

With the protocol cleared up, the FPGA implementation can be discussed. An overview of the UART structure can be seen in figure 4.4. As seen in figure 4.4, the system is further divided into four entities.

- Baud generator
- CRC
- Flip and merge
- UART shift register

Together, these make the UART node in the toplevel motor control design. These four nodes together send one type of message, setting the speed. This is a message with a fixed length and format. This makes the design a lot easier. The layout of the message can be seen in table 4.4. The whole UART entity is based on this design. More details on the different entities within the UART design are given in the rest of this section.

Table 4.4: The layout of the speed message, the numbers are displayed in decimal, MSB and LSB stand for most significant byte and least significant byte respectively

Start Byte	Payload length	Speed ID	Speed as integer				CRC		Stop byte
2	5	8	MSB	...	...	LSB	MSB	LSB	3

Baud generator
As discussed in section 4.2.2, the baudrate of the VESC is fixed at $115\,200\,\text{Hz}$ and thus the baudrate of the UART node should also be $115\,200\,\text{Hz}$. So this nodes generates a 1 clock pulse after every period of the baudrate.

CRC
To verify the integrity of the message, CRC is performed over the payload. A lot of types of the cyclic redundancy check exist, however the VESC uses the *CRC-CCITT*[56] standard. The result will always be a 16 bit vector. The design is inspired by the *maxim integrated 1-Wire CRC* design [38], however, the system was modified to use the $0x1021$ characteristic polynomial as defined by the *CRC-CCITT* standard. The 1-Wire CRC uses a serial interface. It shifts the bits one by one through the system. After all bits are shifted through, the result can be buffered and passed through the next entity. The incoming data is parallel, thus a parallel to serial entity has to written. Together these form the CRC entity. There are only two intermediate signals between these two nodes, The current bit that is shifted and a flag that specifies if it is the last bit of the input. The resulting inputs and outputs can be seen in table 4.5.

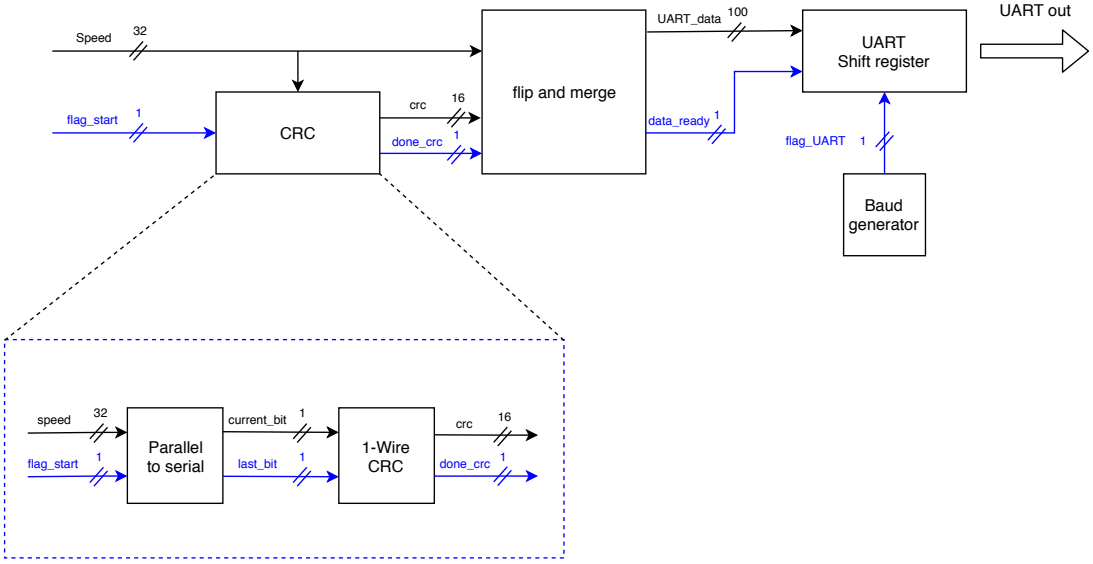


Figure 4.4: Overview of the UART node of the system, data signals are displayed in black, control signals in blue

Table 4.5: A description of the signals going in and out of the CRC entity

Signal	I/O	Description
SPEED	IN	The desired speed for the motor to rotate at
FLAG_START	IN	A flag that triggers a recalculation of the CRC checksum
CRC	OUT	The result of the CRC
DONE_CRC	OUT	A flag that incdicates when the CRC is done

Flip and merge

The function of the *flip and merge* is to put all the separate pieces of data in the right order so the UART shift register only has to shift the data stream out. As part of the UART protocol as described in section 4.2.2, The least significant bit of a byte is sent first. However, the order of the bytes does not change. Meaning the order of the bits have to be reversed while leaving the order of the bytes intact. Hence the name flip and merge, it flips the bits and merges the data.
A description of the signals going in and out of the flip and merge entity can be seen in table 4.6.

Table 4.6: The signals going in and out of the flip and merge entity

Signal	I/O	Description
SPEED	IN	The desired speed for the motor to rotate at
CRC	IN	The result of the CRC
DONE_CRC	IN	A flag that incdicates when the CRC is done
UART_DATA	OUT	The 100 bits in the right order ready to be transmitted
DATA_READY	OUT	A flag telling the UART shift register that the data is stable and valid

UART shift register

Because the length of a message is fixed, there will always be 80 bits of data. When including the start and stop bits to every by native to the UART protocol to every byte the length will be 100 bits. These 100 bits will be passed form the *flip and merge* entity in the right order. They will be shifted one by one onto the data line when given a flag from the baud generator. For safety reasons the motor will stop spinning if it stops receiving speed commands. So after the message is sent, it will start over again. To prevent sending the wrong data, there is a control signal coming from the *flip and merge* entity. This DATA_READY is there to indicate that the data coming from the *flip and merge* is valid and can be used to transmit.

4.2.3. Position

As described in section 3.3, a magnetic encoder is used for position feedback. The magnetic encoder can differentiate between 8192 different positions in one rotation, effectively giving it a maximum quantization error of $0.04°$.
The encoder has three outputs, these are *A*, *B* and *Z*. *A* and *B* are used to differentiate steps, they rise and fall in

predetermined steps during a rotation thus the position can be tracked. *Z* is a reference pulse and is only high at one position. This way, the position can be tracked based on a refrence position of the polygon.

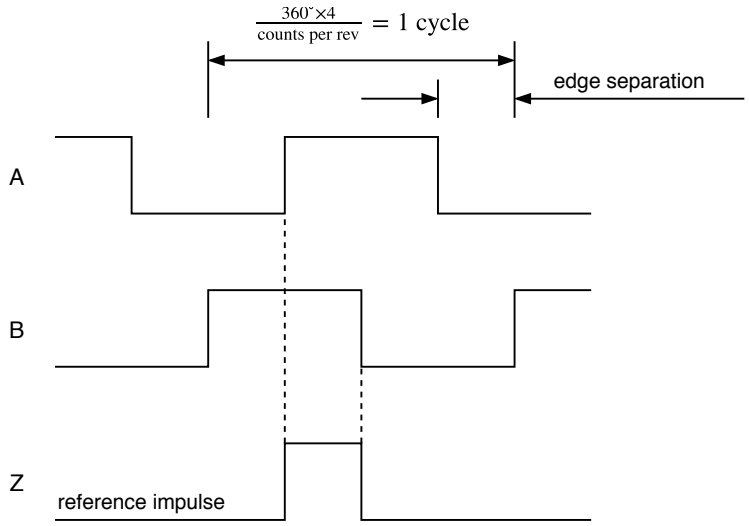


Figure 4.5: Timing diagram for the incremental encoder [53]

When the entity is reset, it waits for a reference pulse. Then it starts counting. The counting does not start at 0, but at a configurable offset. This is because the mirror is not fixed to the shaft and can, therefore, be mounted under another angle. If the offset is configurable, this can easily be adjusted.

As shown in figure 4.3, the position entity has very few inputs besides *A*, *B* and *Z*. These are listed and explained in table 4.7

Table 4.7: A description of the signals going in and out of the position entity

Signal	I/O	Description
POS_OFFSET	IN	A configurable offset to compensate for the angle of mirror
POSITION	OUT	The tracked angle of the rotor shaft in a 13 bit scale

4.2.4. Speed

In the speed entity, two main tasks are executed. The first one is to compute the current speed. The second is to compare speeds to each other to determine the state of the motor and to set a next state. The signals used and created in this entity are described in table 4.8. For computing the speed, one of the incremental encoder signals is taken and its period is converted to a speed. After that the computed speed has to be compared to the desired speed to determine whether or not the motor rotates at its steady state speed. Furthermore, a next speed has to be send over the UART to the VESC. Due to the current limit placed on the VESC, the motor is not able to spin up immediately to every speed. Instead, a ramp-up has to be performed. The speed will slowly increase untill it reaches the final speed. Only when the system rotates in steady state, the measurements can start.

Table 4.8: A description of the signals going in and out of the speed entity

Signal	I/O	Description
SPEED_DES	IN	The desired speed, at which the motor should rotate in steady state
INCREMENTAL_A	IN	Signal A of the incremental encoder
SPEED_CUR	OUT	Measured speed at which the motor is currently running
SPEED_NEXT	OUT	Speed that will be send over the UART to the VESC
SSROT	OUT	A flag that goes high when the motor rotates at the desired speed

4.2.5. Trigger

The trigger entity is used to determine when the laser pulses. It has two operating modes, triggering on a corner for continuous operation and trigger at a single position for 0D mode. The two modes can work along side each other.

A more detailed description about the signals can be seen in table 4.9.

0D mode
For the 0D mode the entity gets an position from the motor interface. If the position of the shaft is the same as this position and the mode is enabled, an output will go high.

Continuous mode
In Continuous operating mode, the system sends out an pulse when the polygon mirror is facing the laser with an edge. Together with the current speed, a linear interpolation can be done to calculate when pulses have to be sent out. Proof showing that the position can be linear interpolated can be seen in appendix A. Because all sides of the polygon have the same length, the distance between every pulse is the same length, every 1024 steps of the position. As mentioned earlier, the position of the mirror with respect to the motor shaft is not fixed and thus a offset has to be given. This was already done in the position entity. Because of this the first corner is always at 0. The advantage of compensating in the position entity is that no calculations have to be done to determine the positions of the corners. Meaning simple comparators can be used, reducing the delay of the system.

Table 4.9: A description of the trigger entity's signal

Signal	I/O	Description
POSITION	IN	Current position of the motor shaft
MODE	IN	A 2 bit vector that determines the operating modes of the system. The MSB is for the 0D mode and the LSB is for continuous mode. Setting one of these two bits high will enable the respective mode
FOV_START	OUT	A flag that goes high if the polygon is facing the laser with one specific edge of the polygon, indicating one full rotation of the polygon. Only active if continuous mode is enabled
FOV_CORN	OUT	A flag that goes high if the polygon is facing the laser with one of its edges, only active if continuous mode is enable
POS_0D	OUT	The desired position on the polygon to pulse in 0D mode
FLAG_0D	OUT	A flag that goes high if the position of the polygon matches POS_0D, only goes high if 0D mode is enabled

4.2.6. Control FSM

The control FSM takes care of the overarching control of the whole motor system. It checks all statuses of all the separate nodes and communicates them to the rest of the system. It also handles all the enable flag of all the individual nodes, meaning it can turn off and on separate functions. However, due to time constraints, the control FSM was not yet implemented. There are no status flags coming out of the individual entities yet. Also the enable is looped through to the enable flag of all nodes. A more detailed description about the signals can be seen in table 4.10.

4.3. Power Supply

Of course the system needs power to operate, the motor that was chosen has a peak efficiency of 87% at $18\,\mathrm{V}$ volt according to the datasheet [39]. A $48\,\mathrm{V}$ high power supply is available. A DC-DC $48\,\mathrm{V}$-$18\,\mathrm{V}$ regulator had to be designed. To keep the footprint of the system small, a chip with an internal switching MOSFET was chosen. A chip from *Texas Instruments* met these criteria and could deliver a maximum of $5\,\mathrm{A}$. That is below the $100\,\mathrm{W}$ requirement listed in INF2. This is also where the $5\,\mathrm{A}$ limit comes from described earlier. The chip used is the *TPS4560* [70] chip. Other options were either really big by means of footprint or deemed too expensive.

Table 4.10: A description of the trigger entity's signal

Signal	I/O	Description
ENABLE	IN	Enable signal of the whole motor system
IF_STATUS	OUT	A vector containing the status of all nodes
IF_NOTOK	OUT	A flag that goes high if the motor has to stop spinning immediately(e.g. a fatal error). This flag does not influence the speed in any way but is here to let the rest of the system know that the motor has been stopped
TRIG_STAT	IN	A flag that determines the state of the trigger node. If the flag goes high there is something wrong with the entity
TRIG_EN	OUT	Enable flag for the trigger node
SPEED_STAT	IN	A flag that determines the state of the speed node. If the flag goes high there is something wrong with the entity
SPEED_EN	OUT	Enable flag for the speed node
POS_STAT	IN	A flag that determines the state of the position node. If the flag goes high there is something wrong with the entity
POS_EN	OUT	Enable flag for the position node
UART_STAT	IN	A flag that determines the state of the UART node. If the flag goes high there is something wrong with the entity
UART_EN	OUT	Enable flag for the UART node

5

Testing and Validation

To confirm the working of the whole system and check if it stays within the established requirements, different tests have to be conducted. All parts have to be tested separate from each other to confirm that they are working individually, but should also be tested together. The following tests were conducted:

- Motor and driver:
  - Testing the VESC through a UART connection with an *Arduino* as host
  - Tracking the power consumption of the motor
  - Testing the consistency in the rotation of the motor
  - Testing the functionality of motor, driver and encoder in a pressurised environment, up to 100 bar
- Digital motor control;
  - Simulation of the digital motor control in *ModelSim*, both the separate entities and the complete system
  - Testing of the digital motor control on the FPGA, both the separate critical entities and the complete system
- Power supply:
  - Confirm the working of the power supply

5.1. Motor and Driver

The requirements important for motor and driver are INF2, PER7-a and PER7-b and PER9-a upto PER11-b. These requirements are all related to power consumption and rotational stability and speed of the motor.

Of course, before testing the motor itself, first the proper working of driver should be verified. To validate the UART protocol and the modified VESC, the UART was emulated with an *Arduino* compatible microcontroller. The microcontroller was programmed to start spinning the motor as soon as it was powered. Various speeds were tested and the motor seemed stable during all of them. This was also performed at 30 Hz, meaning that requirement PER11-b has been met. An important discovery was made during this test. The VESC does not compensate for extra polepairs. The motor that was used has seven pole pairs per phase meaning that it needed seven full commutation cycles before it makes one rotation. Because of this, the value of the speed sent to the VESC should be seven times higher than the desired speed.

With the working of the VESC confirmed, the performance can be tested. First the power usage has been measured. To test the power usage of the motor, the steady state power consumption was measured in air and in the pressure fluid it will be placed in once the system is fully integrated. Steady state power consumption means the motor has already reached the desired speed and has no current spikes that result from the motor speeding up. The result of the test can be seen in figure 5.1. As seen in this figure, the power usage of the motor in air stays relatively constant. For the test in the liquid however, the power draw goes up significantly. However, the power usage stays below the 100 W limit described in requirement INF2 even when passing the maximum frequency described in requirement PER11-b. Also the consistency of the rotational speed of the motor has been measured. Although there are no direct requirements for a constant rotation speed, it was desired as discussed in section 3.1. Unfortunately, during an earlier test the UART bus of the VESC broke down, so the stability could only be tested in air, since now the VESC had to be controlled via USB. Although this test does give a good indication of the stability, the instability will only be amplified

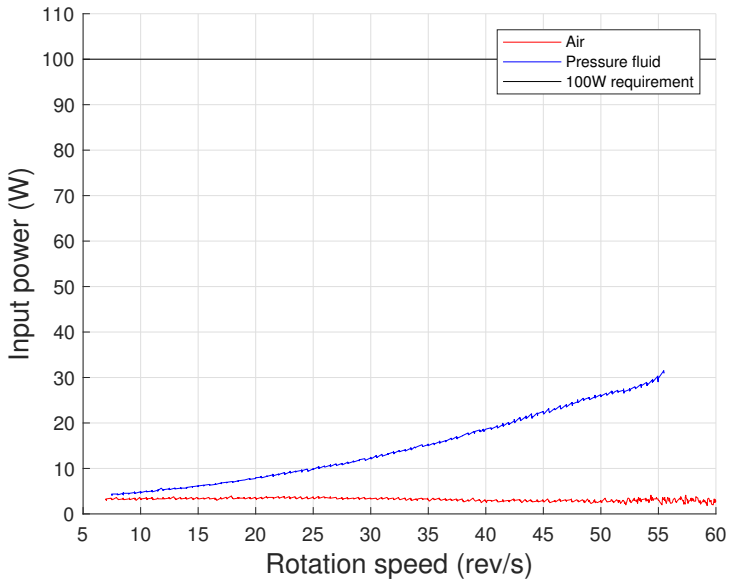


Figure 5.1: Power usage of the system in both air and liquid

because of the higher drag of the pressure fluid. The tests were conducted with the built-in VESC software. Because the motor has 7 pole pairs per phase, the Engine Rotations Per Minute (ERPM) is seven times higher than expected. The results can be seen in the figure 5.2. It can clearly be seen that the mean of these plots is very close to the desired frequencies. However there are inaccuracies in the rotational speed. The greatest deviation is around 20 ERPM which results in $0.05\,\mathrm{Hz}$. For $10\,\mathrm{Hz}$ this would lead to a maximum deviation of 0.5%. Though a speed ripple is present, no large error contribution in the timing of the laser pulses is expected, since the frequency of the ripple present is much lower than the sample rate of the encoder. Therefore, the ripple in the speed can be well accounted for.

Since ripples in the speed will not contribute to positional inaccuracies, errors in position will only occur from encoder read out. Taking into account all uncertainties in the encoder readout, an inaccuracy of $0.54°$ was computed. Though the should accuracy requirement was hereby not met, the design does meet the shall requirement (PER9-a).

To test if all the components could withstand the high pressures set by requirements ENV1-a and ENV1-b, they will be exposed to when in the deapsea, all components were tested up to $100\,\mathrm{bar}$. $100\,\mathrm{bar}$ is the maximum pressure the pressure tank that was available could reach. This may not nearly be the maximum pressure the system will be operating at, but it is a good indication that the system will survive in high-pressure environments. Unfortunately, at the time of the test, the system was unable to rotate because the UART bus stopped working. As the UART turned out to be unreliable, further testing had to be done with the VESC software operating the motor through a USB connection. This also meant the system could not be tested in fluid anymore since access to physical USB connection is needed to operate the motor. After the whole system was pressurised to $100\,\mathrm{bar}$ and depressurised again, the system was found to be fully functional, except for the still unreliable UART, proving that the system can withstand pressures up to $100\,\mathrm{bar}$ and at least meets requirement ENV1-a.

5.2. Digital Motor Control

To test the complete behaviour of each VHDL node, they have been simulated in software. This was done for all nodes and, after some iterations, they all worked as expected in the simulation. However, when the nodes work in the simulation, it does not mean they still work when implemented on an FPGA. To verify this last point, first the most critical nodes and then the whole digital design have been implemented on the FPGA. The system was tested and the outputs were verified. However, it appeared that the clock frequency of the FPGA was half of what was initially specified, so the code had to be adjusted to work with the new clock frequency. This, however, could be done without too much effort or trouble.

5.3. Power Supply

Of course the system is not capable of running without power. As mentioned in section 4.3, a power supply was designed. Due to time constraints, however, the power supply could not be assembled and, thus, also not be tested.

Important to note, all tests mentioned above are done with an external 18 V power supply capable of delivering plenty of current.

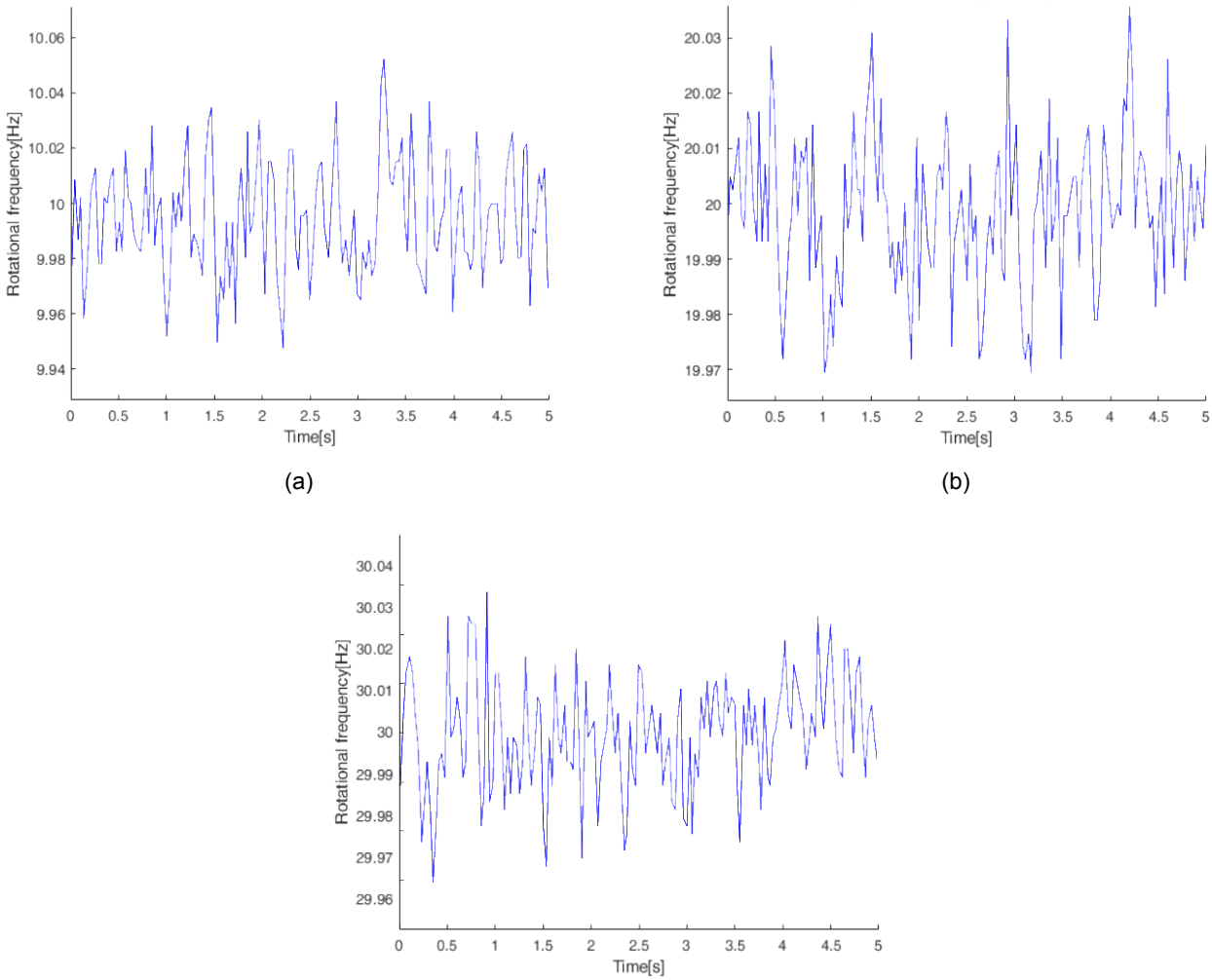


Figure 5.2: The stability of the motor speed measured at different speeds, at 10 Hz, 10 Hz and 30 Hz for (a), (b) and (c) respectively. Because the motor has 7 polepairs per phase, the ERPM is seven times as high as expected.

6

System Integration

This chapter is about the integration of the different modules. However, due to time limitations no proper testings of the complete system could be performed. The modules were, however designed to easily integrate. Therefore, when all modules are truly complete, integration should not take up too much time anymore.

7

Conclusion

In this thesis the design of the electrical part of a LiDAR laser beam steering module has been described. The goal was to create a beam steering system based on a polygon mirror attached to a motor shaft, that provides high accuracy beam steering at a maximum refreshrate of the FOV of 30 Hz. This can be accomplished through high accuracy feedback on the rotary position of the motor.

From the work done, it can be concluded that in air most *should* requirements, but at least all *shall* requirements have been met. The power draw of the motor in steady state is a around 4 W, which is far below the requirement of less than 100 W for the complete system. Furthermore, for the accuracy it was concluded that the motor stability has little influence on the position fed back to the rest of the system. Therefore, the inaccuracy in position is almost completely determined by the encoder. This would result in an inaccuracy of 0.54°, consisting of 0.5° true inaccuracy and 0.04° quantization noise. This is below the *shall* requirement of 1°. To draw the same conclusions about accuracy in fluid or under pressure cannot be done. Due to the VESC's UART breaking, not all tests in fluid could be performed. It is, however, expected that to great extent the requirements will still be met. At least all components are able to survive a pressure of up to 100 bar and the maximum steady sate power is with 13 W well below the 100 W requirement. Which gives hope for further fluid and pressure tests.

Of course the beam steering module is only a part of the complete SLiDAR. Therefore, to link this module to the rest of the system, an interface was created on the FPGA. However, not only an interface, but also some control functionality was implemented in this way. From simulation results and tests on the FPGA, it was concluded that the digital motor control functions properly.

A part of the system that was not finished, due to time constraints, was the buck converter supplying the subsystem of power. However, the design is ready to be implemented. For the time being a lab voltage source can be used to power the system. The buck converter is only needed in the fully integrated system.

Next to the beam steering module, also two other submodules of the SLiDAR have been designed. However, due to the problems with the UART connection, the motor could not be driven in the completely integrated system. Also, due to time limitations no integration tests of the other two modules of the system could be performed.

Future work

Besides finalising the last parts of the design and fixing the UART, an important improvement to make is increasing the angular accuracy to meet the *should* azimuth angular accuracy requirement. One way to do this is get a better encoder, however, this will greatly impact the price of the system. Another option would be to perform some low pass filtering on the Fourier transform of the measured time between two encoder pulses. This in order to filter out the noise. Based on the improved speed readout, that can be obtained from this, also a better position estimation can be derived.

Of course, also a complete design test should be performed as soon as possible.

Bibliography

[1] 2GRobotics. *ULS-500 PRO Dynamic Underwater Laser Scanner*. URL: https://www.2grobotics.com/products/underwater-laser-scanner-uls-500/.

[2] Haitham Abu-Rub, Atif Iqbal, and Jaroslaw Guzinski. "High Performance Control of AC Drives with Matlab / Simulink Models". In: *High Performance Control of AC Drives with Matlab / Simulink Models*. John Wiley & Sons, Incorporated, 2012. Chap. 2: Mathematical and Simulation Models of AC Machines, pp. 26–30. ISBN: 9781119969235. URL: https://ebookcentral-proquest-com.tudelft.idm.oclc.org/lib/delft/detail.action?docID=881653.

[3] A. Almslmany et al. "Time varying gain amplifier linearity enhancement for wide dynamic range in radar receiver". In: *Proceedings of 2015 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization, NEMO 2015*. IEEE, Aug. 2016, pp. 1–4. ISBN: 9781479968114. DOI: 10.1109/NEMO.2015.7415022. URL: http://ieeexplore.ieee.org/document/7415022/.

[4] *AMT Modular Encoders | CUI Inc*. URL: https://www.cui.com/amt-modular-encoders (visited on 06/20/2019).

[5] Ery Arias-Castro and Yonina C. Eldar. "Noise folding in compressed sensing". In: *IEEE Signal Processing Letters* 18.8 (2011), pp. 478–481. ISSN: 10709908. DOI: 10.1109/LSP.2011.2159837.

[6] Shlomi Arnon. "Underwater optical wireless communication network". In: *Optical Engineering* 49.1 (2010), p. 015001. ISSN: 0091-3286. DOI: 10.1117/1.3280288. URL: http://opticalengineering.spiedigitallibrary.org/article.aspx?doi=10.1117/1.3280288.

[7] W.L. Barber and E.R. Brown. "A true logarithmic amplifier for radar IF applications". In: *IEEE Journal of Solid-State Circuits* 15.3 (June 1980), pp. 291–295. ISSN: 0018-9200. DOI: 10.1109/JSSC.1980.1051386. URL: http://ieeexplore.ieee.org/document/1051386/.

[8] *Blackmore LiDAR*. URL: https://blackmoreinc.com/afdl (visited on 06/16/2019).

[9] V. Brandou et al. "3D Reconstruction of Natural Underwater Scenes Using the Stereovision System IRIS". In: *OCEANS 2007 - Europe*. IEEE, June 2007, pp. 1–6. ISBN: 978-1-4244-0634-0. DOI: 10.1109/oceanse.2007.4302315. URL: http://ieeexplore.ieee.org/document/4302315/.

[10] J.D.A. van den Broek. "Design and implementation of an Analog-to-Time-to-Digital converter". PhD thesis. UT Twente, 2012. URL: http://essay.utwente.nl/69501/1/MSc.%20report%20Broek%20van%20den,%20J.D.A.pdf.

[11] J. H.R. Burns and D. Delparte. "Comparison of commercial structure-from-motion photogrammety software used for underwater three-dimensional modeling of coral reef environments". In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*. Vol. 42. 2W3. Feb. 2017, pp. 127–131. DOI: 10.5194/isprs-archives-XLII-2-W3-127-2017. URL: https://www.int-arch-photogramm-remote-sens-spatial-inf-sci.net/XLII-2-W3/127/2017/.

[12] F. M. Caimi and F. R. Dalgleish. "Performance considerations for continuous-wave and pulsed laser line scan (LLS) imaging systems". In: *Journal of the European Optical Society* 5 (Apr. 2010), 10020s. ISSN: 19902573. DOI: 10.2971/jeos.2010.10020s. URL: https://www.jeos.org/index.php/jeos%7B%5C_%7Drp/article/view/10020s.

[13] Jonathan L. Carrivick and Mark W. Smith. "Fluvial and aquatic applications of Structure from Motion photogrammetry and unmanned aerial vehicle/drone technology". In: *Wiley Interdisciplinary Reviews: Water* 6.1 (Jan. 2019), e1328. ISSN: 2049-1948. DOI: 10.1002/wat2.1328. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/wat2.1328.

[14] *Coda Octopus, sound underwater intellegence*. URL: https://www.codaoctopus.com/.

[15] *Communicating with the VESC using UART | Benjamin's robotics*. URL: http://vedder.se/2015/10/communicating-with-the-vesc-using-uart/ (visited on 06/21/2019).

[16] DeepSea Power & Light. *Subsea Cameras*. URL: http://www.deepsea.com/products/cameras/ (visited on 07/12/2019).

[17] 3D at Depth. *SL3 Subsea LiDAR Laser*. URL: https://www.3datdepth.com/product/subsea-lidar-sl3.

[18] Edgetech. *EdgeTech Sonar*. URL: https://www.edgetech.com/products/side-scan-sonar/.

[19] Andrew Filisetti et al. “Developments and applications of underwater LiDAR systems in support of marine science”. In: *OCEANS 2018 MTS/IEEE Charleston, OCEAN 2018* January (2019), pp. 1–10. DOI: 10.1109/OCEANS.2018.8604547.

[20] Miles Hansard et al. *Time of Flight Cameras : Principles , Methods , and Applications*. 2012. DOI: 10.1007/978-1-4471-4658-2. URL: https://hal.inria.fr/hal-00725654/PDF/TOF.pdf.

[21] Duo-Min He and Gerald G.L. Seet. “Divergent-beam Lidar imaging in turbid water”. In: *Optics and Lasers in Engineering* 41.1 (Jan. 2004), pp. 217–231. ISSN: 0143-8166. DOI: 10.1016/S0143-8166(02)00138-0. URL: https://www-sciencedirect-com.tudelft.idm.oclc.org/science/article/pii/S0143816602001380.

[22] Franco Hidalgo and Thomas Braunl. “Review of underwater SLAM techniques”. In: *ICARA 2015 - Proceedings of the 2015 6th International Conference on Automation, Robotics and Applications*. IEEE, Feb. 2015, pp. 306–311. ISBN: 9781479964666. DOI: 10.1109/ICARA.2015.7081165. URL: http://ieeexplore.ieee.org/document/7081165/.

[23] Austin Hughes and Bill Drury. “Chapter Nine - Synchronous and Brushless Permanent Magnet Machines and Drives”. In: *Electric Motors and Drives (Fourth Edition)*. Ed. by Austin Hughes and Bill Drury. Fourth Edition. Boston: Newnes, 2013, pp. 281–313. ISBN: 978-0-08-098332-5. DOI: https://doi.org/10.1016/B978-0-08-098332-5.00009-7. URL: http://www.sciencedirect.com/science/article/pii/B9780080983325000097.

[24] Austin Hughes and Bill Drury. “Chapter Ten - Stepping and Switched-reluctance Motors”. In: *Electric Motors and Drives (Fourth Edition)*. Ed. by Austin Hughes and Bill Drury. Fourth Edition. Boston: Newnes, 2013, pp. 315–347. ISBN: 978-0-08-098332-5. DOI: https://doi.org/10.1016/B978-0-08-098332-5.00010-3. URL: http://www.sciencedirect.com/science/article/pii/B9780080983325000103.

[25] Austin Hughes and Bill Drury. “Chapter Three - Conventional D.C. Motors”. In: *Electric Motors and Drives (Fourth Edition)*. Ed. by Austin Hughes and Bill Drury. Fourth Edition. Boston: Newnes, 2013, pp. 73–111. ISBN: 978-0-08-098332-5. DOI: https://doi.org/10.1016/B978-0-08-098332-5.00003-6. URL: http://www.sciencedirect.com/science/article/pii/B9780080983325000036.

[26] Muhammad Ikhlas. “Rotor Position Identification for Brushless DC motor”. In: *Electronic Thesis and Dissertation Repository* (Feb. 2015). URL: https://ir.lib.uwo.ca/etd/2676.

[27] Achuta Kadambi and Petros T. Boufounos. “Coded aperture compressive 3-D LIDAR”. In: *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings* 2015-Augus (2015), pp. 1166–1170. ISSN: 15206149. DOI: 10.1109/ICASSP.2015.7178153.

[28] Sang-Hoon Kim. “Chapter 10 - Brushless direct current motors”. In: *Electric Motor Control*. Ed. by Sang-Hoon Kim. Elsevier, 2017, pp. 389–416. ISBN: 978-0-12-812138-2. DOI: https://doi.org/10.1016/B978-0-12-812138-2.00010-6. URL: http://www.sciencedirect.com/science/article/pii/B9780128121382000106.

[29] LeddarTech. *Leddar Optical Time-Of-Flight Sensing Technology*. 2016.

[30] B. Timothy Lee. “How 10 leading companies are trying to make powerful, low-cost lida”. In: *Ars Technica* (2019). URL: https://arstechnica.com/cars/2019/02/the-ars-technica-guide-to-the-lidar-industry/.

[31] Larry Li. *Time-of-Flight Camera–An Introduction*. 2014.

[32] Jerry Liao. *Bar code reader with polygon mirror having curved reflection surfaces*. Sept. 2005. URL: https://patents.google.com/patent/US20070069025.

[33] Daniel J. Lum, Samuel H. Knarr, and John C. Howell. “Frequency-modulated continuous-wave LiDAR compressive depth-mapping”. In: *Optics Express* 26.12 (June 2018), p. 15420. ISSN: 1094-4087. DOI: 10.1364/OE.26.015420. URL: https://www.osapublishing.org/abstract.cfm?URI=oe-26-12-15420.

[34] Neil Manning. *Teledyne CDL*. 2014.

[35] Marco Reps. *All about the Xiaomi Lidar Scanner and the Sunfounder RasPad*. URL: https://www.youtube.com/watch?v=4sQCz75BfrM (visited on 06/18/2019).

[36] Gerald F. Marshall and Glenn E. Stutz. *Polygonal Scanners: Components, Performance, and Design*. 2011, p. 788. ISBN: 1439808791. URL: http://books.google.com/books?id=MLWUatLv0s0C%7B%5C&%7Dpgis=1.

[37] Miquel Massot-Campos and Gabriel Oliver-Codina. “Optical Sensors and Methods for Underwater 3D Reconstruction.” In: *Sensors (Basel, Switzerland)* 15.12 (Dec. 2015), pp. 31525–57. ISSN: 1424-8220. DOI: 10.3390/s151229864. URL: http://www.ncbi.nlm.nih.gov/pubmed/26694389%20http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=PMC4721784.

[38] Maxim Integrated. *Understanding and Using Cyclic Redundancy Checks with Maxim 1-Wire*. URL: https://www.maximintegrated.com/en/app-notes/index.mvp/id/27 (visited on 05/02/2019).

[39] Maxon Motor. *Maxon EC-i 40*. 2018. URL: https://www.maxonmotor.nl/medias/sys%7B%5C_%7Dmaster/root/8830903517214/2018EN-250.pdf.

[40] Paul F. McManamon et al. "Comparison of flash lidar detector options". In: *Optical Engineering* 56.3 (2017), p. 031223. ISSN: 0091-3286. DOI: 10.1117/1.oe.56.3.031223.

[41] L. E. Mertens and F. S. Replogle. "Use of point spread and beam spread functions for analysis of imaging systems in water". In: *Journal of the Optical Society of America* 67.8 (Aug. 1977), p. 1105. ISSN: 0030-3941. DOI: 10.1364/josa.67.001105. URL: https://www.osapublishing.org/abstract.cfm?URI=josa-67-8-1105.

[42] Michael Watts. *A review of Optical Phased Array LiDAR*. 2018. URL: https://www.youtube.com/watch?v=H-ZYe2IONOs (visited on 06/18/2019).

[43] P M Moser. *Spectral transmission of light through sea water*. Tech. rep. September. Pacific-Sierra Research Corporation, 1992. URL: http://www.dtic.mil/docs/citations/AD1012965.

[44] Pep Lluis Negre, Francisco Bonin-Font, and Gabriel Oliver. "Cluster-based loop closing detection for underwater slam in feature-poor regions". In: *Proceedings - IEEE International Conference on Robotics and Automation*. Vol. 2016-June. IEEE, May 2016, pp. 2589–2595. ISBN: 9781467380263. DOI: 10.1109/ICRA.2016.7487416. URL: http://ieeexplore.ieee.org/document/7487416/.

[45] Newton. *M310UW Dual Usage Laser Scanner*. URL: https://www.newtonlabs.com/scan%7B%5C_%7Dm300uw%7B%5C_%7Dsys%7B%5C_%7Dspecs.html.

[46] Nic Bingham. *Designing pressure-tolerant electronic systems*. 2013. URL: https://www.uutech.com/ptepaper/ (visited on 07/12/2019).

[47] Takashi; Ogawa and Gerd; Wanielik. "FUSION 2016 : 19th International Conference on Information Fusion : proceedings : Heidelberg, 5-8 July 2016." In: (2016), p. 2337. URL: https://ieeexplore-ieee-org.tudelft.idm.oclc.org/document/7528061.

[48] "Optical phased array lidar system and method of using same". In: (May 2014). URL: https://patents.google.com/patent/US20160161600A1/en.

[49] Ouster. *Ouster Lidar product listing*. URL: https://www.ouster.io/ (visited on 06/18/2019).

[50] Bing Ouyang, Fraser R Dalgleish, and Anni K Vuorenkoski. *Feasibility Study of Compressive Sensing Underwater Imaging Lidar*. 2014. URL: https://apps.dtic.mil/docs/citations/ADA622707.

[51] Angus Pacala. *How Multi-Beam Flash LIDAR Works (Ouster blog)*. 2018. URL: https://www.ouster.io/blog-posts/2018/11/8/how-multi-beam-flash-lidar-works (visited on 06/18/2019).

[52] Christopher V. Poulton et al. "Coherent solid-state LIDAR with silicon photonic optical phased arrays". In: *Optics Letters* 42.20 (2017), p. 4091. ISSN: 0146-9592. DOI: 10.1364/ol.42.004091.

[53] RLS. "RMB28 angular magnetic encoder module". In: 6 (2010), pp. 1–10.

[54] *RLS: Rotary and linear motion sensors*. URL: https://www.rls.si/ (visited on 06/21/2019).

[55] Kraken Robotics. *The Kraken SeaVision: 3D RGB Underwater Laser Scanner*. URL: https://krakenrobotics.com/products/seavision/.

[56] Ross N. Williams. "A PAINLESS GUIDE TO CRC ERROR DETECTION ALGORITHMS". In: (1993). URL: http://ceng2.ktu.edu.tr/%7B~%7Dcevhers/ders%7B%5C_%7Dmateryal/bil311%7B%5C_%7Dbilgisayar%7B%5C_%7Dmimarisi/supplementary%7B%5C_%7Ddocs/crc%7B%5C_%7Dalgorithms.pdf.

[57] Matija Rossi et al. "Real-Time Underwater StereoFusion". In: *Sensors (Basel, Switzerland)* 18.11 (Nov. 2018), p. 3936. ISSN: 14248220. DOI: 10.3390/s18113936. URL: http://www.mdpi.com/1424-8220/18/11/3936.

[58] S Rutten et al. "On the way to a pressure tolerant LiDAR for deep sea robot navigation". In: 4472659 ().

[59] Davide Scaramuzza et al. "Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age". In: *IEEE Transactions on Robotics* 32.6 (2016), pp. 1309–1332. ISSN: 1552-3098. DOI: 10.1109/tro.2016.2624754.

[60] David J. Segelstein. "The complex refractive index of water". In: (1981). URL: https://mospace.umsystem.edu/xmlui/handle/10355/11599.

[61] Sexton Corporation. *Sexton camera enclosures*. URL: http://www.thesextonco.com/shop/camera-enclosures/ (visited on 07/12/2019).

[62] Yoni Sher et al. “Low Intensity LiDAR using Compressed Sensing and a Photon Number Resolving Detector”. In: (Feb. 2018). arXiv: 1802.09354. URL: http://arxiv.org/abs/1802.09354.

[63] *Side scan sonar, Klein Marine inc.* URL: http://kleinmarinesystems.com/products/side-scan-sonar/.

[64] Sidus Solutions LLC. *Cameras*. URL: http://www.sidus-solutions.com/product-category/cameras/ (visited on 07/12/2019).

[65] Slamtec. *RPLIDAR S1 product listing*. URL: https://www.slamtec.com/en/Lidar/S1 (visited on 06/18/2019).

[66] Slamtec. *RPLIDAR-A1 product listing*. URL: https://www.slamtec.com/en/Lidar/A1 (visited on 06/18/2019).

[67] Rafa Souad and Houcine Zeroug. “Comparison between direct torque control and vector control of a permanent magnet synchronous motor drive”. In: *2008 13th International Power Electronics and Motion Control Conference*. IEEE, Sept. 2008, pp. 1209–1214. ISBN: 978-1-4244-1741-4. DOI: 10.1109/EPEPEMC.2008.4635433. URL: http://ieeexplore.ieee.org/document/4635433/.

[68] Sulis Subsea. *SULIS Z70*. 2018. URL: http://www.sulissubsea.com/wp-content/uploads/2018/10/SULIS-Z70%7B%5C_%7Ddatasheet-20181029.pdf.

[69] *Teledyne technologies*. URL: http://www.teledyne.com/.

[70] Texas Instruments. *TPS54560 4 . 5 V to 60 V Input , 5 A , Step Down DC-DC Converter with Eco-mode ™*. 2017. URL: http://www.ti.com/lit/ds/symlink/tps54560.pdf.

[71] Texas Instruments Incorporated. *LIDAR Pulsed Time of Flight Reference Design*. 2018. URL: http://www.ti.com/tool/TIDA-00663%7B%5C#%7Dtechnicaldocuments.

[72] A. R. Thurber et al. “Ecosystem function and services provided by the deep sea”. In: *Biogeosciences* 11.14 (July 2014), pp. 3941–3963. ISSN: 17264189. DOI: 10.5194/bg-11-3941-2014. URL: https://www.biogeosciences.net/11/3941/2014/.

[73] Timothy B. Lee. *Why spinning lidar sensors might be around for another decade*. 2018. URL: https://arstechnica.com/cars/2018/05/why-bulky-spinning-lidar-sensors-might-be-around-for-another-decade/ (visited on 06/18/2019).

[74] Tritech. *Gemini 720im Multibeam Sonar*. URL: https://www.tritech.co.uk/product/gemini-720im.

[75] Benjamin Vedder. *VESC – Open Source ESC*. 2015. URL: http://vedder.se/2015/01/vesc-open-source-esc/ (visited on 06/20/2019).

[76] Velodyne. *Velodyne Puck product listing*. URL: https://velodynelidar.com/vlp-16.html (visited on 06/18/2019).

[77] Huikai Xie et al. “Wide-angle structured light with a scanning MEMS mirror in liquid”. In: *Optics Express* 24.4 (2016), p. 3479. DOI: 10.1364/oe.24.003479.

[78] Fangpei Zhang. “Broad band direct modulation for chirp AM lidar”. In: *Optik* 130 (Feb. 2017), pp. 383–392. ISSN: 00304026. DOI: 10.1016/j.ijleo.2016.10.097. URL: https://www-sciencedirect-com.tudelft.idm.oclc.org/science/article/pii/S0030402616312475.

[79] Hao Zhang et al. “Bidirectional reflectance measurements of sediments in the vicinity of Lee Stocking Island, Bahamas”. In: *Limnology and Oceanography* 48.1part2 (Jan. 2003), pp. 380–389. ISSN: 00243590. DOI: 10.4319/lo.2003.48.1_part_2.0380. URL: http://doi.wiley.com/10.4319/lo.2003.48.1%7B%5C_%7Dpart%7B%5C_%7D2.0380.

[80] Xiaoyang Zhang et al. “MEMS mirrors submerged in liquid for wide-angle scanning”. In: *2015 Transducers - 2015 18th International Conference on Solid-State Sensors, Actuators and Microsystems, TRANSDUCERS 2015* (2015), pp. 847–850. DOI: 10.1109/TRANSDUCERS.2015.7181056.

[81] □□□□□□□□□□□□□□□. *Laser radar based on MEMS micro mirror*. Nov. 2015. URL: https://patents.google.com/patent/CN205120965U/en?oq=lidar+mems.

A

The polygon

In the greenlight assessment, the question was raised if the angle of the shaft can be linearly interpolated.

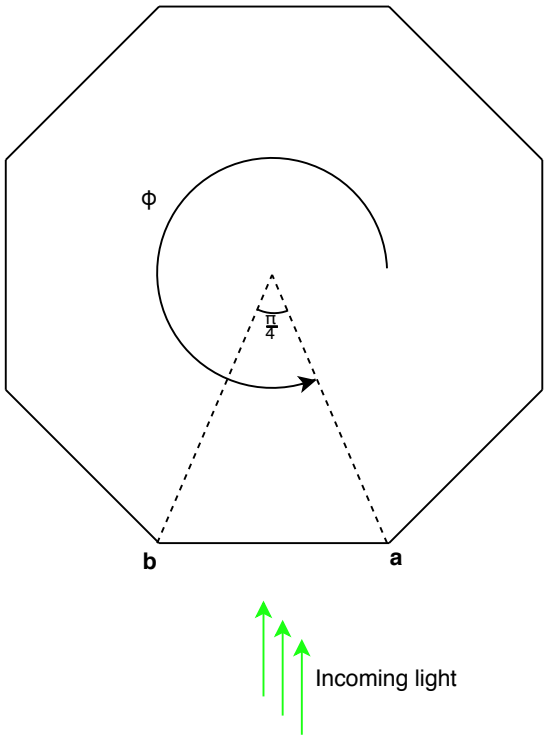


Figure A.1: Simplification of situation to prove angle can be linearly interpolated.

The situation is displayed in figure A.1. The following parameters can be defined.

$$\mathbf{a} = \begin{bmatrix} r\sin(\phi) \\ -r\cos(\phi) \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} r\sin(\phi - \frac{\pi}{4}) \\ -r\cos(\phi - \frac{\pi}{4}) \end{bmatrix}$$

$$\|\mathbf{a}\| = \|\mathbf{b}\| = r$$

$$ab = \mathbf{b} - \mathbf{a} = \begin{bmatrix} b_x - a_x \\ b_y - a_y \end{bmatrix}$$

Due to symmetry

$$0 \leq \phi \leq \frac{\pi}{4}$$

the normal vector $\hat{n}$ is ab shifted by 90 degrees and normalised

$$\hat{n} = \frac{1}{\|\mathbf{b}-\mathbf{a}\|}\begin{bmatrix} a_y - b_y \\ b_x - a_x \end{bmatrix}$$

Because the laser and the centre of the polygon are both in the y-z plane, the incoming light can be modeled as

$$r_i = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

r_o is the reflection vector of r_i over $\hat{n}$

$$r_0 = (r_i - \text{proj}_n(r_i)) - \text{proj}_n(r_i) = r_i - 2\hat{n}(\hat{n} \cdot r_i)$$

$$r_o = \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \frac{2a_x - 2b_x}{\|\mathbf{b}-\mathbf{a}\|^2}\begin{bmatrix} a_y - b_y \\ b_x - a_x \end{bmatrix}$$

$$\|\mathbf{b}-\mathbf{a}\|^2 = (b_x - a_x)^2 + (b_y - a_y)^2$$

so

$$r_o = \begin{bmatrix} 0 \\ 1 \end{bmatrix} + \frac{2a_x - 2b_x}{(b_x - a_x)^2 + (b_y - a_y)^2}\begin{bmatrix} a_y - b_y \\ b_x - a_x \end{bmatrix}$$

Substituting and simplifying gives

$$r_0 = \begin{bmatrix} -\cos(2\phi + \frac{\pi}{4}) \\ -\sin(2\phi + \frac{\pi}{4}) \end{bmatrix} = \begin{bmatrix} \cos(2\phi + \frac{5\pi}{4}) \\ \sin(2\phi + \frac{5\pi}{4}) \end{bmatrix}$$

Clearly, the angle of the reflection is $2\phi + \frac{5\pi}{4}$ with respect to the x-axis. This means that the angle of the reflected light is linear with ϕ and thus can be linearly interpolated. From this the Field Of View (FOV) can be derived. Because the domain was specified as $0 \leq \phi \leq \frac{\pi}{4}$, the horizontal FOV is $90°$. It was also known that the vertical angle from one plane to the next differs $4.5°$. This means that the vertical FOV is $36°$. Unfortunately, this does not meet *shall* requirement PER6-a, but it complies with the minimal PER6-b requirement.k

B

Range Estimation

.

```
%% define some symbols
syms distance
assume(distance, ''positive)

%% system specs
% transmission specs
slope_efficiency.min = 0.6; % W/A
slope_efficiency.max = slope_efficiency.min; % W/A

transmit_current.min = 1;  % A
transmit_current.max = 1; % A

% detector specs
effective_area.min = 8e-3 * 17e-3; % m^2
effective_area.max = 8e-3 * 17e-3; % m^2

detector_threshold_current.min = 0.4e-6; % A
detector_threshold_current.max = 0.4e-6; % A

detector_efficiency.min  = 8; % A/W
detector_efficiency.max  = 8; %0.25; % A/W

% efficiencies
angular_effiency.min   = cos(deg2rad(75 / 4)) * 0.4;
angular_effiency.max   = 1;

hole_efficiency.min    = 0.95;
hole_efficiency.max    = 0.95;

optical_efficiency.min = 0.9;
optical_efficiency.max = 0.9;

% target specs
albedo.min = 0.1;
albedo.max = 0.2;

% medium specs
water_transmission.min = 0.95; % Np/m
water_transmission.max = 0.95; % Np/m

%% find rx current
efficiency.min   = angular_effiency.min * hole_efficiency.min *       ...
```

```
                  optical_efficiency.min;
efficiency.max    = angular_effiency.max * hole_efficiency.max *       ...
                  optical_efficiency.max;

medium_transmission.min = water_transmission.min^(2*distance);
medium_transmission.max = water_transmission.max^(2*distance);

% assumes lambertian full reflectance
spread.min        = effective_area.min / (2 * pi * distance^2);
spread.max        = effective_area.max / (2 * pi * distance^2);

transmission.min = efficiency.min * medium_transmission.min * spread.min * albedo.min
    ;
transmission.max = efficiency.max * medium_transmission.max * spread.max * albedo.max
    ;

tx_power.min = transmit_current.min * slope_efficiency.min;
tx_power.max = transmit_current.max * slope_efficiency.max;

rx_power.min = transmission.min * tx_power.min;
rx_power.max = transmission.max * tx_power.max;

rx_current.min = rx_power.min * detector_efficiency.min;
rx_current.max = rx_power.max * detector_efficiency.max;

%% find distance
expected_distance.min = vpasolve(rx_current.min == detector_threshold_current.max,
    distance, 0.1);
expected_distance.max = vpasolve(rx_current.max == detector_threshold_current.min,
    distance, 0.1);

expected_distance.min
expected_distance.max
```