



NTNU – Trondheim
Norwegian University of
Science and Technology

Development of DP System for Merlin WR200 ROV

Lasse Muri Knausgård

Master of Science in Engineering Cybernetics

Submission date: June 2013

Supervisor: Jo Arve Alfredsen, ITK

Co-supervisor: Ments Tore Møller, IKM Subsea
Sverre Wendt Slettebø, IKM Subsea

Norwegian University of Science and Technology
Department of Engineering Cybernetics



MASTEROPPGAVE

Kandidatens navn: Lasse Muri Knausgård
Fag: Teknisk kybernetikk
Oppgavens tittel: Utvikling av dynamisk posisjoneringssystem for Merlin WR200 ROV
Engelsk tittel: Development of Dynamic Positioning System for Merlin WR200 ROV

Oppgavens tekst

Merlin WR200 er en Klasse 3 arbeidsklasse-ROV med fullelektrisk fremdriftssystem og åpen rammeløsning. Den har pr. idag kun autofunksjoner for dybde og retning. For å øke presisjonen og lette operatørens arbeidsoppgaver under operasjoner nær havbunnen (dyp < 100 meter), er det nå ønskelig å utstyre farkosten med kapasitet for lokal dynamisk posisjonering (station keeping). (lokal i den forstand at det ikke eksisterer et tilgjengelig globalt referansesystem og det vil dermed være relativt til bunnen i operasjonsområdet) I prosjektoppgaven ble det utviklet en matematisk modell for farkosten, og etablert en strategi og et rammeverk for implementering av DP i farkostens styresystem. Masterprosjektet går ut på å videreutvikle rammeverket, utvikle og implementere en regulator for lokal dynamisk posisjonering av farkosten, samt teste og dokumentere ytelsen til denne. Oppgaven omfatter følgende oppgaver:

- Litteraturstudium: reguleringsstrategier for ROV DP-systemer.
- Utvikle og implementere en observer for navigasjonspakken til Merlin til dynamisk posisjonering.
- Utvikle og implementere en modellbasert regulator for DP til Merlin WR200.
- Utvikle og implementere HMI for testing/forsøk til Merlin WR200, bl.a. forskjellige modus for regulatoren. Auto retning, auto dybde, auto orientering og station keeping. Gjøre det enkelt å stille på parametere i regulatoren og modellen (Online tuning).
- Utvikle og implementere referansmodell mellom HMI og regulator. Bl.a. sikre oppnåelige ønskede settpunkter, hastigheter og akselerasjoner for regulatoren.
- Gjennomføre stegvis uttesting av arbeid som er/blir gjort. Bl.a. verifisering av tidligere utviklet modell av Merlin. Redusere antall feilkilder med tanke på vanskeligheten av å teste implementert system. Det vil være begrenset med muligheter til å teste systemet i praksis.
- Lage en test- og verifiseringplan for uttesting av valgt(e) regulator(er), bl.a. logging av tilgjengelige målinger

Oppgaven gitt: 17. januar 2013
Besvarelsen leveres innen: 20. juni 2013
Utført ved: Institutt for teknisk kybernetikk, NTNU
Veileder: Jo Arve Alfredsen, ITK, NTNU
Ments Tore Møller (IKM Subsea)
Sverre Wendt Slettebø (IKM Subsea)

Trondheim, 17. januar 2013

Jo Arve Alfredsen
Faglærer

Abstract

This thesis presents and documents the efforts to model, develop and implement a full six-degree-of-freedom dynamic positioning system for IKM Subsea's Merlin WR200 ROV. It also documents the performance of the system during full-scale sea trials. The aim of the system is to ease the burden on human pilots and increase the efficiency, reliability and endurance of both pilot and ROV. A dynamic positioning system allows a pilot to concentrate on more important tasks than keeping the ROV at a fixed position and orientation. It also introduces the possibility of low speed maneuverability as part of the system as well as an intuitive user interface that will simplify the challenge of conducting complicated maneuvers, thus possibly reducing the necessary training time for the user of the system.

Several large modules were created through the development of a complete control system for Merlin, including a mathematical six-degree-of-freedom model of ROV based on data from unique hydrodynamic experiments and CAD modeling. This is the basis for the implemented controller design that uses the ROV's inverted dynamics. Furthermore, development, implementation and validation of a comprehensive state estimator that uses raw data available from Merlin's existing sensors were done to enable local dynamic positioning. Development and implementation of the reference model and a reference frame for control input to the system have also been performed. Results from part-tests and sea trials where the total system was tested are included. The results of this thesis show a system that works and performs very well with few shortcomings considering the scope and complexity of the undertaking. However, more work around completion and testing of the system is necessary before it could be installed as a control system for operational use by Merlin.

Sammendrag

Denne avhandlingen presenterer og dokumenterer arbeidet med å modellere, utvikle og implementere et komplett seks frihetsgraders dynamisk posisjoneringssystem for IKM Subseas Merlin WR200 ROV samt viser og dokumenterer ytelsen til systemet under gjennomføring av fullskala sjøprøver. Målet med systemet er å lette på byrden, øke effektivitet, pålitelighet og utholdenhet for både pilot og ROV. Et dynamisk posisjoneringssystem vil la piloten konsentrere seg om viktigere oppgaver enn det å holde ROVen i en fast posisjon og orientering i tillegg til at mulighet for lav hastighet manøvrering som en del av systemet samt intuitivt brukergrensesnitt vil forenkle kompliserte manøvringer å redusere opplæringstiden for bruk av systemet.

Gjennom utviklingen av et komplett styringssystem for Merlin er flere store moduler blitt utarbeidet. Dette inkluderer en matematisk seks frihetsgraders modell av ROVen basert på data fra gjennomførte unike hydrodynamiske forsøk og produksjonstegninger. Denne ligger som basis for det implementerte regulatordesign som benytter seg av å invertere dynamikken til ROVen. Videre er utvikling, implementering og validering av en omfattende tilstandsestimator basert på tilgjengelige rådata fra Merlins eksisterende sensorer blitt gjort å muliggjør lokal dynamisk posisjonering. Utvikling og implementering av referansemødel samt en egen referanseramme for styring av systemet er også blitt utført. Resultater fra delforsøk samt godt dokumenterte resultater av sjøprøver der hele systemet blir satt på prøve er inkludert. Resultatene av denne avhandlingen viser et system som fungerer og presterer svært godt med få mangler tatt omfanget i betraktning. Det må likevel påregnes både mer arbeid rundt å ferdigstille systemet samt til testing for å installere det som et styringssystem for operativ bruk av Merlin.

Acknowledgement

I want to thank my supervisor, Jo Arve Alfredsen at NTNU, for agreeing to be my thesis supervisor; our constructive discussions during the process were a great help to me. I would also like to thank IKM Subsea and, specifically, Ments Tore Møller and Sverre Wendt Slettebø for accepting me into IKM's summer internship program and allowing me to work with the complex system that served as the seed of this thesis. It was very motivating to work with such talented people who were so generous with their knowledge and their time.

“Anyone can hold the helm when the sea is calm”

-Publilius Syrus

Contents

1	Introduction	1
1.1	Problem Definition	1
1.2	Main Objective	2
1.3	Outline	2
2	Background Information	4
2.1	Remotely Operated Vehicle	4
2.2	Dynamic Positioning	4
2.3	DP of ROVs	5
2.4	DP Control System Architecture	6
2.4.1	Creating a Successful Control System	8
2.5	Merlin WR 200	10
2.5.1	Previous Work on the Control System for Merlin WR200 from IKM Subsea	11
2.5.2	Previous Work on Merlin WR200 from Author	11
3	Modeling	13
3.1	Kinematics	14
3.1.1	Ocean Current	18
3.2	Equations of Motion	19
3.2.1	System Inertia	20
3.2.2	Hydrodynamics	22
3.2.3	Hydrostatics	24
3.2.4	Cable Drag	26
3.3	Thrusters	26
3.3.1	Thruster Model	26
3.3.2	Thrust Allocation	28
3.4	Modeling Results	32
3.5	DP Capability	33
4	Sensor Setup and Availability	34
5	Control System	38
5.1	Sensor Input and Transformation	38
5.1.1	Directe	38
5.1.2	Transformed	38
5.2	Discretization	40

5.2.1	Numerical Integration Method	40
5.2.2	Numerical Differentiation Method	40
5.3	State Estimators	41
5.3.1	Observer Design for Dynamic Positioning	41
5.3.2	Extended Kalman Filter	43
5.3.3	Testing and Tuning	46
5.3.4	Variables to be Estimated	47
5.4	State Machines	47
5.4.1	DVL Bottom Lock and Suitability	47
5.4.2	Control System	48
5.5	Human Machine Interface	49
5.5.1	Control Interface	49
5.5.2	Development Interface	51
5.6	Reference Models	52
5.6.1	Velocity and Acceleration	53
5.6.2	Attitude	54
5.6.3	Position	54
5.6.4	Active	54
5.7	Controller	55
5.7.1	Available Data for Control	55
5.7.2	State Feedback Linearization	55
5.7.3	Modes	60
5.7.4	Transitions	61
5.7.5	State Feedback	61
5.7.6	Velocity Control	61
5.7.7	Attitude and Position Control	62
5.7.8	Force Control	64
5.7.9	Integral action	64
5.7.10	Control Law	65
5.8	Thruster Allocation	65
6	Implementation	66
6.1	Estimators	66
6.1.1	North, East Position and Sea Current. Surge and Sway Velocities	66
6.1.2	Weight Matrices Values	68
6.1.3	Depth, Altitude and Down Velocity	70
6.1.4	Weight Matrices Values	70
6.2	Controller Parameters	71

6.2.1	Velocity Gains	71
6.2.2	Attitude and Position Gains	71
6.2.3	Anti-windup Gains	71
6.3	Software Presentation	72
6.4	Data Trace for Omron PLC	72
7	Practical Experiments	72
7.1	Doppler Velocity Log	75
7.1.1	Result	75
7.2	Test Bench	79
7.3	Interface with Existing System	79
8	Sea Trails	81
8.1	Test Setup	83
8.1.1	Data Logging and Video Recording	83
8.1.2	Online Tuning Interface	85
8.1.3	Experiments to be Performed	86
9	Results	89
9.1	Position and Velocity Estimator Test	89
9.1.1	Drift Test with Fixed Position	90
9.1.2	Drift Test with Movement and Fixed Start/Stop Position	91
9.2	Depth and Down Velocity Estimator	94
9.3	Heading Controller	96
9.3.1	In Wave Zone	97
9.4	Station Keeping	98
9.4.1	North and East with Varying Depth	98
9.4.2	North,East and Down Close to Seabed	100
9.4.3	North,East and Down Close to Surface and in Wave Zone	103
9.4.4	North,East and Down with Large Forced Disturbance	106
9.4.5	Roll and Pitch	108
9.5	Force Control	108
10	Discussion	109
10.1	Current Measurement	109
10.2	Estimator	110
10.3	Depth and Heading	110
10.4	Station Keeping	111
10.5	Force Control	111

11 Conclusion	113
12 Further Work	115
12.1 User Interface	115
12.2 Estimator and Controller	115
12.3 Tests	115
12.4 New Features	116
A Drawings	119
A.1 Thruster Placement	119
A.2 Global Coordinates	121
B Merlin Data	123
C Tests	127
D Software	134
D.1 Commanded Acceleration Control Law for Attitude and Position	134
E Mix	138
F Sensors	140
F.1 TOGSNAV	140
F.2 WORKHORSE DVL	152
F.3 DVL-Water Reference Layer Settings	155
G Merlin Overview	158
H File Attachment	160

List of Figures

2.1 DP Control System Architecture	7
2.2 Merlin WR200	10
3.1 Merlin's body-fixed reference frame	13
3.2 North-East-Down frame	15
3.3 Body-Seabed frame	16
3.4 Fitted curve relationship between frequency and thrust	28
3.5 Thruster numbering	30

3.6	Block diagram of thrust allocation	31
3.7	DP capability plot	33
4.1	Screenshot of the custom IKM TOGSNAV string in a terminal program	36
4.2	Water reference layer setup	37
5.1	Block diagram of the Kalman filter as a state estimator	44
5.2	The Kalman filter loop. (Brown & Hwang 1997)	46
5.3	State machine for the DVL	49
5.4	State machine for the DP control system	50
5.5	Control input for the body-seabed frame	51
5.6	Control input in the body-seabed frame	52
5.7	Example of reference model for Merlin	53
6.1	Screenshot of implemented sections and function blocks in Om- rons CX-programmer	73
7.1	Screenshot of the carriage control program	76
7.2	TOGSNAV mounted in a welded stand	77
7.3	The stand with the DVL mounted on the carriage	77
7.4	Trolling motor mounted on the carriage	78
7.5	Extended Kalman filter result of of tank test	78
7.6	Test bench	79
8.1	Sirevaag	81
8.2	Infrastructure in Sirevaag	82
8.3	Underwatercamera with house	83
8.4	Seabed camera mounted on a trestle	84
8.5	Picture of the HMI for online parameter edit	86
8.6	Merlin w\o forced disturbance from the TMS	88
9.1	Drift test with fixed position	90
9.2	Drift test with movment and fixed start/stop position	91
9.3	3D plot of drift test tour	92
9.4	Drift test with movment and fixed start/stop position for velocity	93
9.5	Depth and Down velocity estimator test	94
9.6	Down velocity controller and estimator test	95
9.7	Old vs. new heading controller test	96
9.8	Heading controller test in wave zone	97
9.9	North and East with Varying Depth	98
9.10	3D plot of station keeping with varying depth	99
9.11	North,East and Down Close to Seabed	100
9.12	North-East plot of station keeping	101
9.13	Picture of Merlin during station keeping seen from seabed camera	102

9.14	North,East and Down Close to Surface and in Wave Zone	103
9.15	North-East plot of station keeping in the wave zone	104
9.16	Merlin during station keeping in the wave zone	105
9.17	North,East and Down with Large Forced Disturbance	106
9.18	North-East plot for station keeping test with forced disturbance .	107
9.19	Roll and pitch during normal station keeping	108
C.1	Fitted curve for the drag in positive surge	131
C.2	Fitted curve for the drag in negative surge	132
C.3	Fitted curve for the drag in positive sway	133

List of Tables

1	Merlin's main data	11
2	The notation of SNAME for marine vessels	14
3	Pull test measurements made with Merlin decomposed to a single thruster	27
4	Direct and transformed estimated measurements form TOGSNAV	39
5	Discrete-time EKF	45
6	Control inputs from operator	51
7	Available measurements(from both TOGSNAV direct and estimator)	56
8	Available control inputs from reference model	57
9	Test bench components	80
10	Drag test results for Merlin	130

Nomenclature

Abbreviations

AUV	Autonomous Underwater Vehicles
CAD	Computer-Aided Design
CO	Center of Origin
DOF	Degrees Of Freedom
DP	Dynamic Positioning
DVL	Doppler Velocity Log
GNSS	Global Navigation Satellite System
PLC	Programmable Logic Controller
ROV	Remotely Operated Vehicle
RS	Recommended Standard
SI	Système International d'Unités
SNAME	Society of Naval Architects and Marine Engineers
TMS	Tether Management System
UML	Unified Modeling Language
WGS84	World Geodetic System 1984

1 Introduction

1.1 Problem Definition

Merlin WR200 is a Class 3 work class ROV with full electric propulsion system and open frame solution. It currently has only auto functions for depth and heading. To increase the precision and to ease operator workload during operations near the seabed (depth<100m), it is now desirable to equip the vehicle with the capacity for local(no global reference system available) dynamic positioning. A mathematical model of vehicle was developed in the project thesis Knausgård (2012) and it established a strategy and a framework for implementing dynamic positioning in the vessel's control system. This thesis aims to develop and complete the framework, develop and implement a controller for local dynamic positioning of the vehicle, and test and document the performance of this. The task involves the following tasks:

- Literature study: control strategies for ROV DP systems.
- Develop and implement an observer for the navigation package (TOGSNAV) for Merlin for dynamic positioning.
- Develop and implement a model-based controller for a dynamic positioning system for Merlin.
- Develop and implement a human machine interface for both testing and use for Merlin, including the possibility to switch between different modes in the controller, e.g., auto heading, auto depth/altitude, auto attitude and station keeping. Make it easy to set parameters for the controller and model(online tuning).
- Develop and implement a reference model between human machine interface (e.g. joystick) and the controller that ensures feasible desired set points, velocities and accelerations for the controller.
- Conduct gradual testing of work in progress as well as completed work, including verification of a previously developed model of Merlin. Reduce the number of sources of error in terms of the difficulty of testing the implemented system, as there will be limited opportunities to test the system in practice.

- Creating a test and verification plan for testing of the selected controller and the total system.

It is assumed that the reader of this report has knowledge of modeling and control theory. It is also advantageous to be familiar with the Handbook of Marine Craft Hydrodynamics and Motion Control by T.I. Fossen (Fossen [2011]), since it is used as a basis of this thesis.

1.2 Main Objective

- Develop a functional dynamic positioning system that can cope with varying conditions such as different equipment, missions and environments.
- Create a system that is intuitive to use, maintain and further develop.
- Provide an overall picture of the development from the start of the project thesis to the final result of this master thesis.

1.3 Outline

Chapter 2 provides a quick introduction to the topic of this thesis.

Chapter 3 presents a theory on modelling of a ROV in 6-DOF, mostly based on Fossen [2011]. Much of the work in Knausgård [2012] is reproduced but with some important corrections and extensions. The result is a process plant model of Merlin.

Chapter 4 provides an introduction to the measurements that are available from the sensor device with which Merlin is equipped. Very important corrections as compared to Knausgård [2012] work are made here, particularly changes to the sensor model. Various settings and setup of the device are also discussed.

Chapter 5 is the main chapter of the thesis related to theories and methods around the new control system for Merlin. Different choices of strategies for the system are examined.

Chapter 6 presents important decisions made during the implementation of the system (mainly the estimator). Focuses mostly on design of the implemented system rather than pure reproduction of programming as easily presented using appropriate software.

Chapter 7 shows experiments and practical tests of parts of the system done during development.

Chapter 8 provides an overview of where, how and which parts were to be tested during the main sea trials.

Chapter 9 shows the results of the system developed in this thesis during the sea trials. Shows and documents individual functionalities of the system and the complete system in full operation.

2 Background Information

2.1 Remotely Operated Vehicle

A remotely operated vehicle, commonly referred to as an ROV, is a tethered underwater vehicle. ROVs were introduced in military operations in the 1950s and then, in the following decades, used primarily in rescue and recovery missions. Currently, ROVs are used most extensively in the oil and gas industry, where more and more sophisticated processes take place in increasingly deeper water. ROVs allow oil companies to perform complex subsea tasks without risking human lives. An ROV is designed to be simple to maneuver and is usually controlled by a pilot aboard a mother ship. Communication between the ship and the ROV goes through a tether that also delivers the energy required to power the ROV. Since ROVs are used most often in deep water, it has become more common to use a tether management system(TMS) between vessel and ROV. The TMS reduces cable drag on the ROV as it typically hangs straight down from the vessel; often only 10 percent of the total tether length is on the TMS during operations. The depth of the TMS is controlled from a winch on the vessel. The TMS also serves as a garage for an ROV as it is lowered and lifted. There are several classes of ROVs; those mainly used at the depths the subsea oil industry operates are work class ROVs. These feature over 200 hp (propulsion) and are perfectly capable of doing the required work. Manipulators and capabilities to carry tools are also common. All ROVs are equipped with cameras and lights, and, often, several sophisticated sensors that make it easier to control.

2.2 Dynamic Positioning

A Dynamic Positioning(DP) system can be summed up in the following words: a system which automatically controls a vessel's position and heading exclusively by means of active thrust. The main aim when DP was developed was to keep a vessel stationary at a specific position with a specific heading. This special case of DP is often called station keeping. The terms DP and station keeping are often used somewhat interchangeably, but with station keeping it means that case where a fixed position (and often attitude) will be held. DP may also contain operational modes: semi-automatic DP mode in which some degrees of freedom are controlled manually while others are controlled automatically, e.g. the heading can be controlled manually, while the position of the vessel is kept in a fixed position. DP can also be used for tracking if it continuously receives

new preferred position commands which it will then hold/follow. For example, a ship always remains straight over an ROV under operation; the ship follows the movements of the ROV. All these are special cases of DP despite various designs and applications that have been developed around the same principles.

2.3 DP of ROVs

Dynamic positioning (DP) system for a ROV is a computer-controlled system that automatically maintains the position and heading of the ROV with thrusters using data from various sensors. It is also common to control the attitude of an ROV as it usually is possible to affect all degrees of freedom of an ROV. The reference for the DP system may vary for different tasks; it depends both on the sensors and the equipment available as well as the purpose of the DP system. Holding positions relative to the seabed is typical; with proper equipment, it can also keep its position relative to a vessel, etc. Although DP normally is used to hold a position fixed, it also usually includes low speed maneuverability as part of the DP. When ROVs work at the great depths normal today, it is important that they are both as reliable and effective as possible. The cost of having an ROV in operation for both the customer and the company operating the ROV is so high it is absolutely essential the ROV does what it is designed to do. Despite relatively rapid development in the ROV industry, it is still an unsustainable strain on an ROV pilot to ensure maximum efficiency. According to Stanley [2004], control systems often are neglected when new advances are made. Stanley says: "Frequently, the key focus when conceiving a new generation vehicle has been the vehicle's hardware and electrical interface capabilities. While these are, of course, important areas, the hardware's ease of use is in many ways an equally important factor, and the control system is a key area in designing a system that is straightforward to use". One area that is natural to look at is the development within autonomous underwater vehicles(AUV). Here, a pilot is not part of the development. The development of control systems has been forced to keep up with the development of mechanical and electrical components. AUVs can do relatively simple tasks in a fraction of the time of an ROV. Nevertheless, an ROV today is considered to be reliable and a good starting point for smart automation. Since there is no great hope that ROV operations can be performed in the near future without human intervention, development really must focus on easing the pilot's tasks.

Automatically maintaining desired depth and direction has long been the standard for ROVs. DP has not been very common but is about to be. The development of sensors for use in DP has been promising and the price is dropping

to an affordable level. The most commonly used navigation tool is a Doppler velocity log (DVL), which measures the velocity of an ROV over the seabed; the most advanced can also record the velocity of water below the ROV. If these data are combined with smart use of sensors, gyros, compasses and accelerometers can be fed into the control system. This will then enable various automatic functions associated with DP. Classic DP works by keeping position fixed relative to the bottom. Ability to hold the position while the manipulator is used (reducing the number of pilots) and accuracy control near installations are just some of the possibilities of creating a DP system.

2.4 DP Control System Architecture

As mentioned, most DP systems, almost regardless of purpose, are built around a relatively similar design. To give some insight into the complexity of such a control system, a graphical representation of such a system has been set up (as can be seen in Figure 2.1) where the main components of such a system are inscribed.

Here are the various components of the system from Figure 2.1 with a little explanation:

- **Signal processing:** Relevant measurements from various sensors are checked, verified, converted and normalized in the signal processing part of the system. For example, check for wild points (readings outside set limits), remove signals that are frozen, detect unstable signals (e.g., measurements that fall out constantly), verify logic around voting and redundancy in the system (accessed on several measurements that serve the same purpose), and convert/scale units (e.g., to SI units).
- **Observer:** An estimator or observer is a system that provides an estimate of the internal state of a given real system. It uses measurements of input and output of the real system to estimate desired state. Reasons to use an observable may be that it is impossible to measure the state, impractical, expensive or it may be that a measure is strongly influenced by noise and must be estimated in order to be used in a controller. For example, the velocity of the vessel can be estimated using the available position measurements, or a noise-affected compass measurement can be estimated and filtered. An observer must also ensure that the temporary loss measurements do not make the system come to a halt. The situation that occurs when an observable puts out an estimate of a state for a period in which it does not have any measurements to rely on is often called

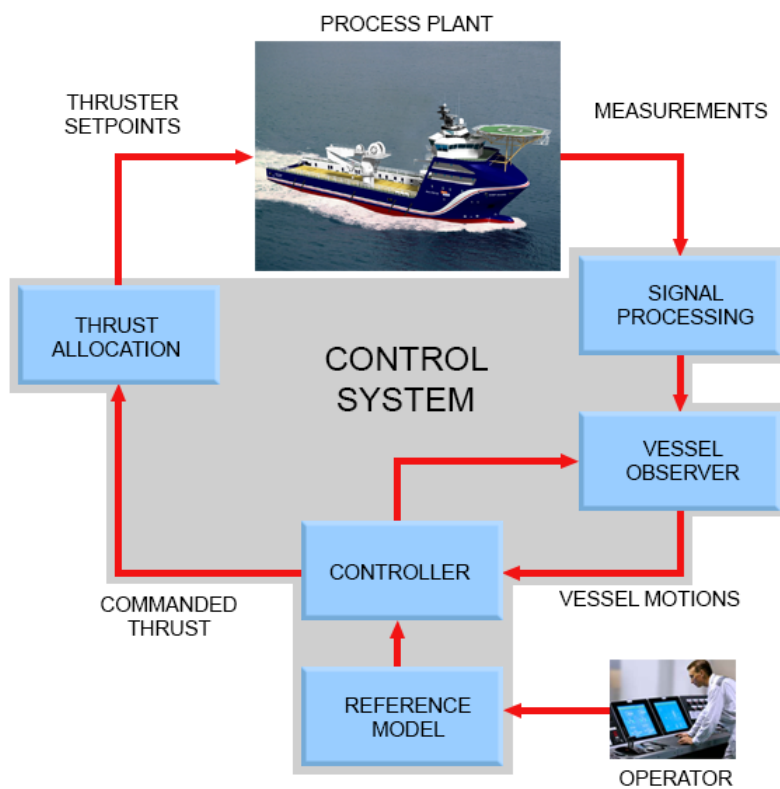


Figure 2.1: DP Control System Architecture

“dead reckoning”. This is important for all marine vehicles because use of sensors that depend strongly on the environment (e.g., Global Navigation Satellite System(GNSS), Doppler velocity log(DVL)) tend to drop out for small periods.

- Reference model: A reference model between raw inputs from the operator (such as joystick, turn-knob and set-points) is desirable to ensure that these inputs will be smooth and feasible for the controller.
- Controller: The controller is the brain of the control system and is where the control law(s) will be placed. The controller uses available measurements, compares them with the desired states and puts out a desired response calculated for the system to fulfill the control law . Logic associated with operating in different modes, transitions and switching between controllers, alarm detection, etc. will also be located here. Note that this is the place in the architecture of the DP system that varies the most according to its purpose. It may also be noted that a controller used for DP can be called a high-level control since it does not work directly on the actuator affecting the vessel but rather on the generalized forces being converted into voltages, speed, pitch, etc., in the lower levels, often outside of the control system.
- Thrust allocation: The desired response from the controller, usually given as force and moments in the different degrees of freedom, is normalized and transformed to the desired response from each actuator. In the allocation there may be limits on the actuator, differences in how effectively the various actuators work, logic to prevent unnecessary wear and tear, fuel optimization (often in conjunction with a "green-controller") and error handling. Depending on the vessel, the desired response from the actuator is transformed to fit the interface that the vessel actuators have (e.g., voltages, speed, and pitch).

2.4.1 Creating a Successful Control System

The key to a successful control system design is finding a delicate balance between conflicting objectives. Sørensen [2012] mentions three crucial ingredients for a successful design:

1. Knowledge (structure and coefficients) of the process to be controlled; structure (kinematics, kinetics: DOF/States, inertia, stiffness, dissipa-

tion, time constants, resonances, load characteristics, etc.); coefficients (parameters, settings, etc.)

2. Understanding of how performance will be assessed.
3. Knowledge of fundamental limitations that may prevent any design from achieving the desired performance: Feasibility and constraints.

“Once we know these factors, we can select from the different design methods of control theory the one that best suits our problem.”



Figure 2.2: Merlin WR200

2.5 Merlin WR 200

Figure 2.2 shows Merlin WR200 and Table 1 shows the main dimensions. Merlin is an electric work class ROV. Throughout the design process, the idea of creating an efficient and reliable ROV has been central. Being electrically driven ensures that any leak of hydraulic fluid is minimized. It also makes it one of the greener choices when it comes to operation for a ROV. The resistance in the water is reduced by using an open frame solution with rounded edges. This allows water to flow more freely around and through the ROV. Merlin has four horizontal and four vertical thrusters; altogether, it has an output of approximately 200 hp. It is equipped with two manipulators and several cameras and lights and has the ability to be fitted with various skids. Another concept that is relatively unique to Merlin is the placement of as many components as possible topside; this increases reliability and eases maintenance. Like many larger ROVs, Merlin is equipped with a TMS that provides greater maneuverability in deep waters. Some of the best available sensors and a modern control container from which everything is managed make Merlin and its infrastructure suitable as a platform for expansion and implementation of new features.

Table 1: Merlin's main data

Length	2.8 m
Width	1.8 m
Height	1.7 m
Weight	~ 3000 kg
Max. Depth	3000 m

2.5.1 Previous Work on the Control System for Merlin WR200 from IKM Subsea

Merlin has a large and comprehensive control system when all the functionalities of an ROV that do not involve maneuvering and control are taken into account. This will obviously not be discussed here but it is important to include in order to understand the interaction that should(will) be there even after the introduction of a new control system for dynamic positioning capability. The existing control system for Merlin has some auto-functions, such as direct thrust control, direct connection between desired gain (e.g., position of the joystick) and thrust. Automatic depth and heading controller and something IKM calls semi-auto-heading make it possible to change the heading and then lock it automatically to the new reference. To do this they use the symmetry for the placement of thrusters and assume a total decoupled model of the ROV

2.5.2 Previous Work on Merlin WR200 from Author

The project thesis Knausgård [2012] and this thesis are closely linked. To make the development of DP system possible, the work was divided up. Examinations and work were done in the project related to background information on DP systems, available infrastructure for Merlin and desired features of a DP system. Also, extensive tests were conducted on the hydrodynamics of Merlin and mathematical modeling of Merlin was performed. For more information, please refer to the project thesis.

Important Corrections from the Project Thesis(Knausgård [2012]):

Some assumptions and planning of this thesis had to be changed because the promised sensor device that Merlin would be equipped with (TOGSNAV2) was unable to be obtained due to delays with the supplier(CDL). This means the thesis scope was increased and there was a little change in focus since it was

no longer possible to fully concentrate on the controller as planned. Development, implementation and testing of an estimator using the existing sensor unit (TOGSNAV) will require a lot of effort. On the other hand, one can say that now we will have better control over the estimated data to be used. It will also be possible to enter model data from Merlin into the estimator, which is thought to be a major plus. Changes were made in the modeling section of Knausgård [2012], too, but were more or less rendered as a modeling section from Knausgård [2012] This is to provide overall context of this thesis work but also to correct major and minor errors and add information. Among the important changes in the modeling is that the thrust configuration matrix is set up again wherein the contribution that Merlin experiences in pitch and roll by, surge and sway motions are taken into account. This was thought to have little significance in Knausgård [2012] but is, after trials and discussions with IKM Subsea, included. Updated CAD drawings with other data has resulted in changes in the modeling of hydrostatics as well.

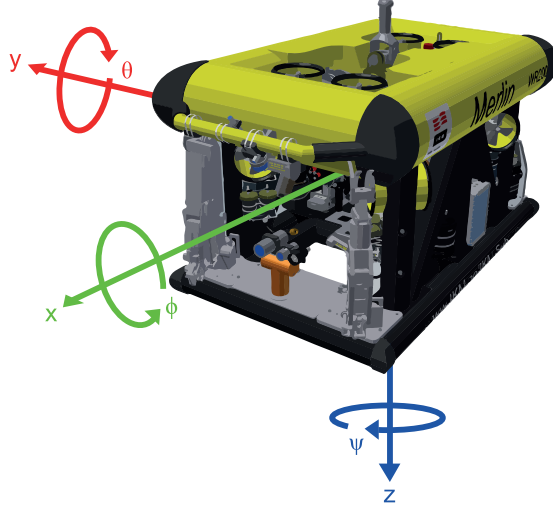


Figure 3.1: Merlin's body-fixed reference frame

3 Modeling

A 6-DOF control plant model will be developed for Merlin. It will be a mathematical representation that models the main characteristics of Merlin and will be of sufficient detail to satisfy what is needed to use it in a feedback-based controller. It will also be dynamic in the sense that it can be used with various types of equipment and tools (different load). Before designing a model, it is important to note the assumptions that are made for the modeling. Assumptions about the environment and dynamics are made for Merlin (Prestero [2001]). Some of these will not be relevant until the regulator is developed but they are included since there is some correlation between them. The assumptions are:

- The ROV will operate deeply submerged.
- The ocean current is irrotational and constant (slowly varying).
- The ocean current is assumed to be parallel to the seabed.
- Density of seawater is assumed constant for all depth.
- The ROV is modeled as a rigid body and the mass is constant.

Table 2: The notation of SNAME for marine vessels

DOF	Forces and moments	Linear and angular velocities	Positions and Euler angels
1 motion in the x direction (surge)	X	u	x
2 motion in the y direction (sway)	Y	v	y
3 motion in the z direction (heave)	Z	w	z
4 rotation about the x axis (roll)	K	p	ϕ
5 rotation about the y axis (pitch)	M	q	θ
6 rotation about the z axis (yaw)	N	r	ψ

- The NED-frame is inertial.
- The angular rates when performing station keeping are quite small.
- The regulator-loop will be faster than the fastest dynamics essential for the ROV.
- The delay in actuation is small compared to the time response of the different DOF.

To model the dynamics of the ROV, this thesis uses the vectorial notation from Fossen [2011], which is based on the standard formulation given in SNAME [1950]. Table 2 shows a description of forces and moments, velocities and positions according to SNAME notation. The body-fixed coordinate system for Merlin as seen in Figure 3.1 has the origin (o_b) in CO(center of origin). CO is placed at the center of the cube the ROV constitute. This can be seen in appendix A.2. All forces and measurements will be seen from this point.

3.1 Kinematics

Kinematics describes geometrical aspects of motion without considering mass and forces (reference frames, variables and transformations). There are three reference frames that will be used in this thesis; following is an explanation of them:

- **NED** - The North-East-Down (*NED*) coordinat system $\{n\} = (x_n, y_n, z_n)$ with origin o_n is defined relative to Earth's reference ellipsoide(WGS84). For a local area operation (which is the case with an ROV), the assumption is constant longitude and latitude. Thus it can be seen as a tangent plane in the area (flat Earth navigation). See Figure 3.2.

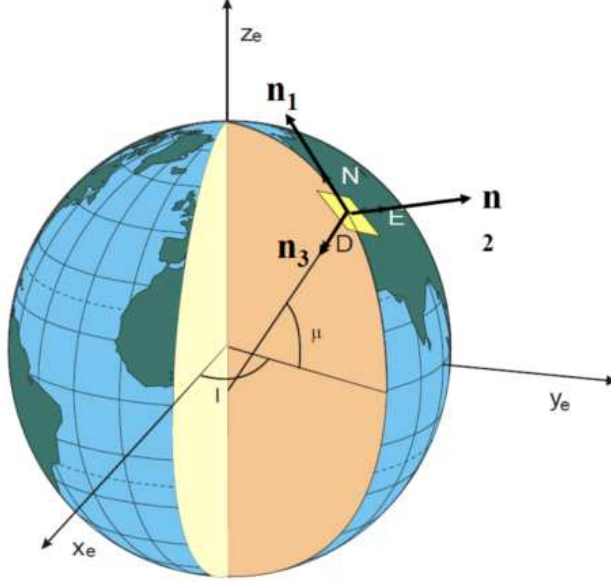


Figure 3.2: North-East-Down frame

- **BODY** - The body-fixed reference frame $\{b\} = (x_b, y_b, z_b)$ with origin o_b is a moving coordinate frame that is fixed to the craft. For Merlin this can be seen in Figure 3.1. Position and orientation is relative to a reference frame that is inertial (NED in this thesis). Both linear and angular velocities are given in the body-fixed frame.
- **BODY-SEABED** - The body-seabed-fixed reference frame $\{s\} = (x_s, y_s, D)$ with origin o_s is a semi-moving coordinate frame that has the Down-axis from the NED in combination with the x_s and y_s axis that are fixed to the crafts yaw angle (as the BODY-frame), and parallel to the seabed. (same the tangent plane for North-East in NED) Hence the subscript s for seabed; see figure 3.3. This reference system is used in the TOGSNAV and will be used for human-machine control input.



Figure 3.3: Body-Seabed frame

Linear and angular velocity of the ROV in both body-fixed and in NED-frame for control tasks is for 6 DOF - surge, sway, heave, roll, pitch and yaw:

$$\dot{\eta} = \begin{bmatrix} \dot{\eta}_1 \\ \dot{\eta}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_b^n(\boldsymbol{\Theta}_{nb}) & 0_{3 \times 3} \\ 0_{3 \times 3} & \mathbf{T}_\Theta(\boldsymbol{\Theta}_{nb}) \end{bmatrix} \begin{bmatrix} \nu_1 \\ \nu_2 \end{bmatrix} \quad (3.1)$$

where:

$\mathbf{R}_b^n(\boldsymbol{\Theta}_{nb})$ is the Euler angle rotation matrix between body- and NED-frame.

The argument is the vector $\boldsymbol{\Theta}_{nb} = [\phi \ \theta \ \psi]^T$

$\mathbf{T}_\Theta(\boldsymbol{\Theta}_{nb})$ is a transformation matrix for the angular velocity.

$$\eta_1 = [N \ E \ D]^T \quad (3.2)$$

$$\eta_2 = [\phi \ \theta \ \psi]^T \quad (3.3)$$

$$\nu_1 = [u \ v \ w]^T \quad (3.4)$$

$$\nu_2 = [p \ q \ r]^T \quad (3.5)$$

$$\nu = [\nu_1 \ \nu_2]^T \quad (3.6)$$

To relate the body-seabed-frame to body-frame and consequently also NED-frame can be done as follows:

$$\nu_1 = \mathbf{R}_s^b(\boldsymbol{\Theta}_{nb}) \nu_s \quad (3.7)$$

where: $\nu_s = [u_s \ v_s \ \dot{D}]^T$ and $\mathbf{R}_s^b(\boldsymbol{\Theta}_{nb})$ is the rotation matrix between body-seabed- and body-frame defined as:

$$\mathbf{R}_s^b(\boldsymbol{\Theta}_{nb}) = \mathbf{R}_{y,\theta} \mathbf{R}_{x,\phi} \quad (3.8)$$

with the properties of rotation matrices it is also easy to see that the rotation matrix relating body-seabed and NED-frame are:

$$\mathbf{R}_s^n(\boldsymbol{\Theta}_{nb}) = \mathbf{R}_b^n(\boldsymbol{\Theta}_{nb}) \mathbf{R}_{y,\theta} \mathbf{R}_{x,\phi} \quad (3.9)$$

Since TOGSNAV uses Euler-angle representation, singularities will occur. These will be ± 90 degrees pitch. This will be ignored as this is well outside the workspace to Merlin (± 20 degree).

3.1.1 Ocean Current

As described in the section with sensor analysis, Merlin is equipped with a DVL (Doppler Velocity Log) that measures velocity relative to the seabed as well as velocity relative to the water. This information is optimal as it can greatly improve behavior in difficult environments with ocean currents. These velocities are given in the body-fixed frame for surge and sway direction. This vector describing the water velocity relative to Merlin:

$$\nu_c = [u_c \quad v_c \quad 0 \quad 0 \quad 0 \quad 0]^T \quad (3.10)$$

It is assumed that the ocean current has no influence in heave at Merlin (based on the fact that Merlin will have a small workspace in pitch and roll). Furthermore, it is also assumed that the the ocean current is irrotational and constant:

$$\dot{\nu}_c = [\dot{u}_c \quad \dot{v}_c \quad 0 \quad 0 \quad 0 \quad 0]^T = 0 \quad (3.11)$$

By introducing a relative velocity (that the velocity relative to the bottom combined with the velocity of the water) it is possible to write this as:

$$\nu_r = \nu - \nu_c = \begin{bmatrix} u \\ v \\ w \\ p \\ q \\ r \end{bmatrix} - \begin{bmatrix} u_c \\ v_c \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} u - u_c \\ v - v_c \\ w \\ p \\ q \\ r \end{bmatrix} \quad (3.12)$$

The derivative is the same as before:

$$\dot{\nu}_r = \dot{\nu} - 0 \quad (3.13)$$

This will be used in the model of Merlin.

3.2 Equations of Motion

Equations of motion that will be used as a control plant for the DP-system expressed in the body-fixed frame:

$$\mathbf{M}\dot{\nu} + \underbrace{\begin{Bmatrix} \mathbf{M}_{skid_1} \\ \mathbf{M}_{skid_2} \\ \mathbf{M}_{skid_3} \\ \dots \end{Bmatrix}}_{\text{switch}} \dot{\nu} + \mathbf{D}(\nu_r)\nu_r + \mathbf{g}(\eta) + \underbrace{\begin{Bmatrix} \mathbf{g}(\eta)_{skid_1} \\ \mathbf{g}(\eta)_{skid_2} \\ \mathbf{g}(\eta)_{skid_3} \\ \dots \end{Bmatrix}}_{\text{switch}} = \tau_{thr} \quad (3.14)$$

where:

$\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A$ system inertia matrix (including added mass)

\mathbf{D} damping matrix

\mathbf{g} of gravitational/buoyancy forces and moments

τ_{thr} vector of control inputs

\mathbf{M}_{skid_i} system inertia matrix for additional equipment

$\mathbf{g}(\eta)_{skid_i}$ vector of gravitational/buoyancy forces and moments for additional equipment

Most of this is found in Fossen [2011]. The ability to switch between different outfits as part of the model is adapted in this thesis to the use of Merlin. The idea is that by keeping the model as reliable as possible, the controller's work can be eased (switching between different models will be done by the ROV pilot). Relatively low velocities are used for DP, typically under 2 meters per second, allowing the Coriolis-centripetal matrix with added mass to be removed. Note that the damping matrix is not linear; it is mainly because unique practical experiments provide access to very good numbers for damping (linear and quadratic). See also that the relative velocity is used in the model. In the following subsections, the various matrixes and vectors for Merlin are mentioned. Note again that everything is given in CO.

3.2.1 System Inertia

The system inertia matrix for Merlin is the mass of the ROV itself and added mass due to the inertia of the surrounding fluid.

$$\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A \quad (3.15)$$

$$\Downarrow$$

$$\mathbf{M} = \begin{bmatrix} m & 0 & 0 & 0 & mz_g & -my_g \\ 0 & m & 0 & -mz_g & 0 & mx_g \\ 0 & 0 & m & my_g & -mx_g & 0 \\ 0 & -mz_g & my_g & I_{xx} & -I_{xy} & -I_{xz} \\ mz_g & 0 & -mx_g & -I_{yx} & I_{yy} & -I_{yz} \\ -my_g & mx_g & 0 & -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix} + \begin{bmatrix} -X_{\dot{u}} & -X_{\dot{v}} & -X_{\dot{w}} & -X_{\dot{p}} & -X_{\dot{q}} & -X_{\dot{r}} \\ -Y_{\dot{u}} & -Y_{\dot{v}} & -Y_{\dot{w}} & -Y_{\dot{p}} & -Y_{\dot{q}} & -Y_{\dot{r}} \\ -Z_{\dot{u}} & -Z_{\dot{v}} & -Z_{\dot{w}} & -Z_{\dot{p}} & -Z_{\dot{q}} & -Z_{\dot{r}} \\ -K_{\dot{u}} & -K_{\dot{v}} & -K_{\dot{w}} & -K_{\dot{p}} & -K_{\dot{q}} & -K_{\dot{r}} \\ -M_{\dot{u}} & -M_{\dot{v}} & -M_{\dot{w}} & -M_{\dot{p}} & -M_{\dot{q}} & -M_{\dot{r}} \\ -N_{\dot{u}} & -N_{\dot{v}} & -N_{\dot{w}} & -N_{\dot{p}} & -N_{\dot{q}} & -N_{\dot{r}} \end{bmatrix} \quad (3.16)$$

where m is the mass. x_g, y_g and z_g are the distance from CO to COG. For the added mass the notation for hydrodynamic derivatives from SNAME(1950) is used.

The inertia matrix is found using the CAD model of Merlin. This data can be found in appendix B. Because CO coincides with CG is $x_g = y_g = z_g = 0$. The added mass matrix that is a part of the inertial matrix in this thesis is based on guesses and the fact that it will be small compared to the total mass of Merlin. Using this reasoning, and making an assumption that, 1) Merlin has symmetry in aft-fore and port-starboard, 2) Merlin does not use roll-motion, and, 3) Merlin has very limited use of pitch motion, the resulting matrix becomes:

$$\mathbf{M} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 & 0 & 0 & 0 \\ 0 & m - Y_{\dot{v}_r} & 0 & 0 & 0 & 0 \\ 0 & 0 & m - Z_{\dot{w}} & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{xx} - K_{\dot{p}} & -I_{xy} & -I_{xz} \\ 0 & 0 & 0 & -I_{yx} & I_{yy} - M_{\dot{q}} & -I_{yz} \\ 0 & 0 & 0 & -I_{zx} & -I_{zy} & I_{zz} - N_{\dot{r}} \end{bmatrix} \quad (3.17)$$

Inserted assumptions:

$$\mathbf{M} = \begin{bmatrix} 1.1m & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.1m & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.1m & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{xx} + 0.05m & -I_{xy} & -I_{xz} \\ 0 & 0 & 0 & -I_{yx} & I_{yy} + 0.05m & -I_{yz} \\ 0 & 0 & 0 & -I_{zx} & -I_{zy} & I_{zz} + 0.05m \end{bmatrix} \quad (3.18)$$

Where there is an assumption that the added mass in surge, sway and heave is 10 percent of the weight of Merlin, the roll, pitch and yaw is reduced to 5 percent. This is based on Shi-bo et al. [2012], but the assumptions are so insecure that they must be part of some tuning parameters for the final model.

3.2.2 Hydrodynamics

The damping represents several damping phenomena from hydrodynamics that is very difficult to model. For this particular thesis, major experimental tests of the damping of Merlin have been performed. This is discussed in detail in the chapter with the experimental results in Knausgård [2012]. The most important results from the testing are presented in Appendix C.

$$\mathbf{D} = \mathbf{D}_l + \mathbf{D}_q |\nu_r|^T \quad (3.19)$$

$$\Downarrow$$

$$\mathbf{D} = \begin{bmatrix} -X_u & -X_v & -X_w & -X_p & -X_q & -X_r \\ -Y_u & -Y_v & -Y_w & -Y_p & -Y_q & -Y_r \\ -Z_u & -Z_v & -Z_w & -Z_p & -Z_q & -Z_r \\ -K_u & -K_v & -K_w & -K_p & -K_q & -K_r \\ -M_u & -M_v & -M_w & -M_p & -M_q & -M_r \\ -N_u & -N_v & -N_w & -N_p & -N_q & -N_r \end{bmatrix} + \begin{bmatrix} -X_{|u|u} & -X_{|v|v} & -X_{|w|w} & -X_{|p|p} & -X_{|q|q} & -X_{|r|r} \\ -Y_{|u|u} & -Y_{|v|v} & -Y_{|w|w} & -Y_{|p|p} & -Y_{|q|q} & -Y_{|r|r} \\ -Z_{|u|u} & -Z_{|v|v} & -Z_{|w|w} & -Z_{|p|p} & -Z_{|q|q} & -Z_{|r|r} \\ -K_{|u|u} & -K_{|v|v} & -K_{|w|w} & -K_{|p|p} & -K_{|q|q} & -K_{|r|r} \\ -M_{|u|u} & -M_{|v|v} & -M_{|w|w} & -M_{|p|p} & -M_{|q|q} & -M_{|r|r} \\ -N_{|u|u} & -N_{|v|v} & -N_{|w|w} & -N_{|p|p} & -N_{|q|q} & -N_{|r|r} \end{bmatrix} |\nu_r|^T \quad (3.20)$$

Note that in the surge and sway, the relative velocity has been used. Fossen [2011] suggests a diagonal form of damping matrixes based on the fact that the diagonal elements will be totally dominant in the velocity in which Merlin operates. This assumes that the damping is decoupled. Thus:

$$\mathbf{D} = \begin{bmatrix} -X_u^* & 0 & 0 & 0 & 0 & 0 \\ 0 & -Y_v^* & 0 & 0 & 0 & 0 \\ 0 & 0 & -Z_w & 0 & 0 & 0 \\ 0 & 0 & 0 & -K_p & 0 & 0 \\ 0 & 0 & 0 & 0 & -M_q & 0 \\ 0 & 0 & 0 & 0 & 0 & -N_r \end{bmatrix} +$$

$$\begin{bmatrix} -X_{|u|u}^* & 0 & 0 & 0 & 0 & 0 \\ 0 & -Y_{|v|v}^* & 0 & 0 & 0 & 0 \\ 0 & 0 & -Z_{|w|w} & 0 & 0 & 0 \\ 0 & 0 & 0 & -K_{|p|p} & 0 & 0 \\ 0 & 0 & 0 & 0 & -M_{|q|q} & 0 \\ 0 & 0 & 0 & 0 & 0 & -N_{|r|r} \end{bmatrix} |\nu_r|^T \quad (3.21)$$

* indicates entry derived from experiment.

By remembering that Merlin will experience most sea current influence in the surge and sway, and that it is in these directions Merlin will operate with the highest velocity, a rough estimate of the damping in heave can be set to the same as that measured in sway. (xz-plane more like the xy-plane than the yz-plane). Furthermore, the damping in roll, pitch, and yaw is assumed to be very small; it should not allow large angular velocities in these directions either. In Chin and Lau [2012] and in Shi-bo et al. [2012], complete hydrodynamic experiments have been made with an ROV that has some similarities to Merlin. The results make clear that the damping in yaw was minimal in relation to surge and sway. Experiments carried out by Svendby [2007] suggest that the damping in yaw is also minimal. An approach to the damping of these degrees of freedom could be 10 percent of the average of the damping in the surge and sway. These will be tuning variables. Thus:

$$\mathbf{D} = \begin{bmatrix} -X_u^* & 0 & 0 & 0 & 0 & 0 \\ 0 & -Y_v^* & 0 & 0 & 0 & 0 \\ 0 & 0 & -Y_v^* & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} +$$

$$\begin{bmatrix}
-X_{|u|u}^* & 0 & 0 & 0 & 0 & 0 \\
0 & -Y_{|v|v}^* & 0 & 0 & 0 & 0 \\
0 & 0 & -Y_{|v|v}^* & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{-X_{|u|u}^* - Y_{|v|v}^*}{20} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{-X_{|u|u}^* - Y_{|v|v}^*}{20} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{-X_{|u|u}^* - Y_{|v|v}^*}{20}
\end{bmatrix} |\nu_r|^T \quad (3.22)$$

Since towing tank experiments showed that the damping of Merlin has a very clear quadratic form and all the linear damping are therefore omitted, this is what remains:

$$\mathbf{D} = \begin{bmatrix}
-X_{|u|u}^* & 0 & 0 & 0 & 0 & 0 \\
0 & -Y_{|v|v}^* & 0 & 0 & 0 & 0 \\
0 & 0 & -Y_{|v|v}^* & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{-X_{|u|u}^* - Y_{|v|v}^*}{20} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{-X_{|u|u}^* - Y_{|v|v}^*}{20} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{-X_{|u|u}^* - Y_{|v|v}^*}{20}
\end{bmatrix} |\nu_r|^T \quad (3.23)$$

3.2.3 Hydrostatics

In collaboration with IKM Subsea it is assumed that Merlin will be neutral in the water during the experiments to be performed. But it will not be the case later in normal operation, so even if great simplifications could be made in this sub-chapter, it is desirable that further work include more information. The forces and moments due to gravity and buoyancy are the restoring forces for Merlin:

$$\mathbf{g}(\eta) = \begin{bmatrix} (W - B) \sin(\theta) \\ -(W - B) \cos(\theta) \sin(\phi) \\ -(W - B) \cos(\theta) \cos(\phi) \\ -(y_g W - y_b B) \cos(\theta) \cos(\phi) + (z_g W - z_b B) \cos(\theta) \sin(\phi) \\ (z_g W - z_b B) \sin(\theta) + (x_g W - x_b B) \cos(\theta) \cos(\phi) \\ -(x_g W - x_b B) \cos(\theta) \sin(\phi) - (y_g W - y_b B) \sin(\theta) \end{bmatrix} \quad (3.24)$$

where W is force of gravity and B is the force of bouyancy acting on Merlin. Using elementary fluid dynamics and data about Merlin make it easy to set up expressions for these:

$$W = m \cdot g \quad (3.25)$$

where m is the mass of Merlin and g is gravitational acceleration ($g = 9.81 \text{ m/s}^2$). From the CAD model of Merlin: $m = 3300 \text{ kg}$.

$$B = \rho g V \quad (3.26)$$

where V is the volume of Merlin and ρ is the density of the seawater. From the CAD model of Merlin: $V = 3.2 \text{ m}^3$. Density of seawater: $\rho = 1024 \text{ kg/m}^3$

$x_g, x_b, y_g, y_b, z_g, z_b$ are the respective distances between the CG and CB to CO on the three axes. It is easy to verify that if $x_g = x_b$ and $y_g = y_b$ then CG and CB are located vertically on the z -axis. This is the case for Merlin and it can be written:

$$\mathbf{g}(\eta) = \begin{bmatrix} (W - B) \sin(\theta) \\ -(W - B) \cos(\theta) \sin(\phi) \\ -(W - B) \cos(\theta) \cos(\phi) \\ (z_g W - z_b B) \cos(\theta) \sin(\phi) \\ (z_g W - z_b B) \sin(\theta) \cos(\theta) \cos(\phi) \\ 0 \end{bmatrix} \quad (3.27)$$

z_g and z_b are the distance from CO to the centre of gravity and the centre of bouyancy. From the CAD model of Merlin: $z_g = 0 \text{ m}$ and $z_b = 0.16 \text{ m}$.

If the ROV had been absolutely neutral in the water, would the first 3 rows of 3.27 be 0. That is not the case for Merlin normally, but as mentioned, it is the case in this thesis. Eq. 3.27 can then be simplified to:

$$g(\eta) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ (z_g W - z_b B) \cos(\theta) \sin(\phi) \\ (z_g W - z_b B) \sin(\theta) \cos(\theta) \cos(\phi) \\ 0 \end{bmatrix} \quad (3.28)$$

Note that natural buoyancy is desirable if emergencies should occur. Usage of thrusters near the sea bottom may also disturb the fine sediment (mud); as a result, clouds of dust can dramatically decrease visibility. By having $B > W$, the use of thrusters towards the bottom is reduced. Since a dynamic model of Merlin is used, it should develop a unique $\mathbf{g}(\eta)$ for each outfitting. This means that by using a CAD model of the equipment, one can make these vectors relatively easily. It is worth noting that the simplifications that have been made about the Merlin are not necessarily valid here, so $x_g = x_b$ and $y_g = y_b$ is not valid. Thus it is necessary to use equation 3.24 to solve for them.

3.2.4 Cable Drag

The ROV Merlin is attached to an umbilical cable that provides it with power and control signals, etc. This cable imposes forces on the ROV due to drag, but since Merlin operates with TMS (unlike some other ROVs) the forces from the cable will be small. Therefore, cable drag is ignored in this thesis.

3.3 Thrusters

3.3.1 Thruster Model

Merlin is equipped with frequency controlled thrusters. Because it is desirable to have some sort of force-controlled thrusters, is it important to find a mapping between the applied frequency and the force the thruster provides. Note that there is no feedback from the thrusters. Since it is in the lower level of the control loop, it is a crucial factor in an underwater vehicle system (Kim and Chung [2006]). This is, in practice, very difficult to model because it includes complex

Table 3: Pull test measurements made with Merlin decomposed to a single thruster

Frequency[Hz]	Thrust[N]
0	0
8	10
15	35
23	121
30	242
38	415
46	587
53	829
69	1071
75	1382
80	1727
81	1865
83	1900

physical phenomena; the hydrodynamic effects are almost impossible to model in a satisfactory manner. Various thruster loss will also be present (Sørensen [2012]). This means that it falls off the scope of this thesis. The solution is proposed in Sørensen [2012] using empirical results in combination with analytical methods. IKM Subsea has conducted various drag tests (including maximum bollard pull experiments) wherein different control input (frequency) was applied to Merlin and the current force was measured (appendix C). Data from a pull test used to create a model can be seen in Table 3.

This data is not without some uncertainty. It is desirable to have a continuous function that describes the relationship between frequency and thrust. Because of this, only the end points of the test are used, and a simplification was made that fits quite well with test data. It turns out that by having a sine function defined with amplitude equal to a maximum frequency and phase giving it a period covering thrust area, one then has a good mapping used in the allocation. This connection is shown below and the text in appendix E describes how this is done with Matlab.

$$f_{Hz} = 83 \sin(0.0007909 F) \quad (3.29)$$

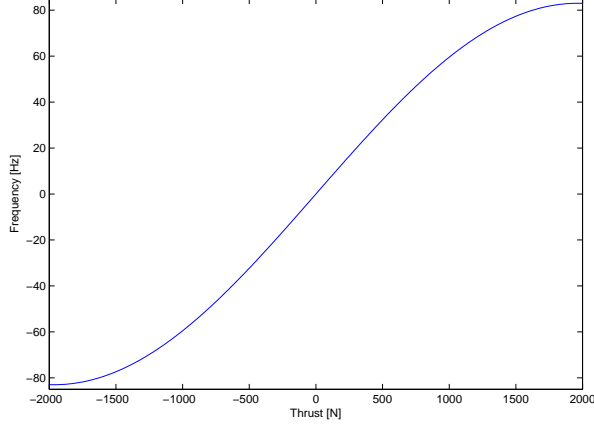


Figure 3.4: Fitted curve relationship between frequency and thrust

Where F is the force required by the thruster and f_{Hz} is the frequency that gives this thrust. Equation 3.29 is also valid for negative force. (See Figure 3.4) This relationship will be used later in thrust allocation.

3.3.2 Thrust Allocation

The thrusters' forces and moments relate to the control forces and moments can be written as:

$$\tau_{thr} = \mathbf{T} \mathbf{K} \mathbf{u} \quad (3.30)$$

The thruster configuration matrix \mathbf{T} describes the location of the thrusters on Merlin, and, consequently, how the force of each thruster works in each degree of freedom seen from the CO. u is control input to each thruster and \mathbf{K} is the force coefficient matrix. With the help of a drawing with dimensions and placement of the thruster (see appendix A) it is straightforward to set up the thruster configuration matrix for Merlin. The direction of the thrusters is defined as follows: The 4 horizontal thrusters are defined as positive where they made a positive contribution in the x-direction. The 4 vertical thrusters

correspond with positive z-direction contribution. The project thesis assumed that distance from the CO to the four horizontal thrusters in the Z- direction was so small that it could be ignored. Upon closer examination of the CAD drawings and discussions with IKM Subsea, this has now been changed. As one can see, a distance in the Z- direction of 0.1 meters has been added. This will make that the coupling in the degrees of freedom roll, pitch and surge, sway is reflected by the matrix.

$$\mathbf{T} = \begin{bmatrix} 0 & 0 & 0 & 0 & \sin(\alpha) & \sin(\alpha) & \sin(\alpha) & \sin(\alpha) \\ 0 & 0 & 0 & 0 & -\cos(\alpha) & \cos(\alpha) & -\cos(\alpha) & \cos(\alpha) \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ l_2 & -l_2 & -l_2 & l_2 & -l_5 & l_5 & -l_5 & l_5 \\ -l_1 & -l_1 & l_3 & l_3 & -l_5 & -l_5 & -l_5 & -l_5 \\ 0 & 0 & 0 & 0 & -l_4 & l_4 & l_6 & -l_6 \end{bmatrix} \quad (3.31)$$

where l_i is the length of the arm that creates momentum in roll, pitch and yaw:

$$\begin{aligned} l_1 &= 0.73 \text{ m} \\ l_2 &= 0.24 \text{ m} \\ l_3 &= 0.73 \text{ m} \\ l_4 &= 0.84 \text{ m} \\ l_5 &= 0.10 \text{ m} \\ l_6 &= 0.84 \text{ m} \end{aligned}$$

α is the angel thruster are located(in the xy-plane):

$$\alpha = 45^\circ$$

Since there are 8 identical thrusters on Merlin and it is advantageous to have a simple saturation control in the allocation, some modifications are made to what is found in Fossen [2011]. The relationship between the required force in each DOF and the force of each thruster is:

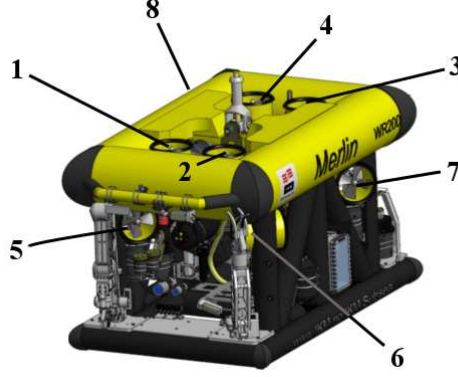


Figure 3.5: Thruster numbering

$$\tau_{thr} = \mathbf{T} \tau_u \quad (3.32)$$

$$\Downarrow$$

$$\begin{bmatrix} X \\ Y \\ Z \\ K \\ M \\ N \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & \sin(\alpha) & \sin(\alpha) & \sin(\alpha) & \sin(\alpha) \\ 0 & 0 & 0 & 0 & -\cos(\alpha) & \cos(\alpha) & -\cos(\alpha) & \cos(\alpha) \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ l_2 & -l_2 & -l_2 & l_2 & -l_5 & l_5 & -l_5 & l_5 \\ -l_1 & -l_1 & l_3 & l_3 & -l_5 & -l_5 & -l_5 & -l_5 \\ 0 & 0 & 0 & 0 & -l_4 & l_4 & l_6 & -l_6 \end{bmatrix} \cdot \begin{bmatrix} \tau_{u1} \\ \tau_{u2} \\ \tau_{u3} \\ \tau_{u4} \\ \tau_{u5} \\ \tau_{u6} \\ \tau_{u7} \\ \tau_{u8} \end{bmatrix}. \quad (3.33)$$

where τ is the desired force in the different DOF, \mathbf{T} is the thruster configuration matrix and τ_u is the desired force of each thruster. Each column of \mathbf{T} represents the contribution of a thruster in each DOF. The columns are defined from thruster numbering as in Figure 3.5. Now an expression can be set up for the desired force from the thrusters given the desired force in each DOF:

$$\tau_u = \mathbf{T}^\dagger \tau_{thr} \quad (3.34)$$



Figure 3.6: Block diagram of thrust allocation

where \mathbf{T}^\dagger is the *Moore-Penrose pseudo-inverse* defined as:

$$\mathbf{T}^\dagger = \mathbf{T}^T (\mathbf{T} \mathbf{T}^T)^{-1}$$

Note that weighting is not used in the different thrusters, since this was not necessary, but it can easily be added if needed. This is also where there could be an availability matrix for the thrusters that are available (for a fault-tolerant system), but it is not part of this task and is, therefore, omitted. Now that the force that is required from each thruster has been found, the associated thruster frequency can be mapped. However, to make the system a bit more fail safe, it is prudent to include the aforementioned saturation control. This should also be part of the logic to the reference model, but it is good rule to also add it to the lowest level. An easy way to make sure that ratio of the desired force is maintained even if the desired force is beyond the physical limitations of thrusters is scaling. Below is a possible solution where the infinity norm is used for this normalization:

$$\|\tau_u\|_\infty := \max(|\tau_{u_1}|, \dots, |\tau_{u_8}|) \quad (3.35)$$

$$\tau_{u_i} = \begin{cases} \tau_{u_i} & \text{if } \|\tau_u\|_\infty < \tau_{saturation} \\ \frac{\tau_{u_i}}{\|\tau_u\|_\infty} \tau_{saturation} & \text{otherwise} \end{cases} \quad \text{for } i = 1, \dots, 8 \quad (3.36)$$

As one can see, it is easy to implement the thruster allocation. In Figure 3.6, a block diagram of the entire thruster allocation is shown. The last step in allocation is to use the model from 3.29 of the thrusters to give the correct frequency to the frequency converters:

$$u_i = 83 \sin(0.0007909 \tau_{u_i}) \quad \text{for } i = 1, \dots, 8 \quad (3.37)$$

Note again that the model for the thrusters is identical for all 8 thrusters.

3.4 Modeling Results

The different results of this chapter's values from Merlin are inserted below.

System Inertia Matrix (including added mass):

$$\mathbf{M} = \begin{bmatrix} 3320 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3320 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3320 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1970 & 0 & -120 \\ 0 & 0 & 0 & 0 & 3215 & 7 \\ 0 & 0 & 0 & -120 & 7 & 3037 \end{bmatrix}$$

Damping:

$$\mathbf{D}_q = \begin{bmatrix} 1321 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2525 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2525 & 0 & 0 & 0 \\ 0 & 0 & 0 & 192 & 0 & 0 \\ 0 & 0 & 0 & 0 & 192 & 0 \\ 0 & 0 & 0 & 0 & 0 & 192 \end{bmatrix} |\nu_r|^T$$

Gravitational/Buoyancy Forces and Moments:

$$\mathbf{g}(\eta) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -4820 \cos(\theta) \sin(\phi) \\ -4820 \sin(\theta) \cos(\theta) \cos(\phi) \\ 0 \end{bmatrix}$$

The Thruster Configuration Matrix:

$$\mathbf{T} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0.71 & 0.71 & 0.71 & 0.71 \\ 0 & 0 & 0 & 0 & -0.71 & 0.71 & -0.71 & 0.71 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0.24 & -0.24 & -0.24 & 0.24 & -0.10 & 0.10 & -0.10 & 0.10 \\ -0.73 & -0.73 & 0.73 & 0.73 & -0.10 & -0.10 & -0.10 & -0.10 \\ 0 & 0 & 0 & 0 & -0.84 & 0.84 & 0.84 & -0.84 \end{bmatrix}$$

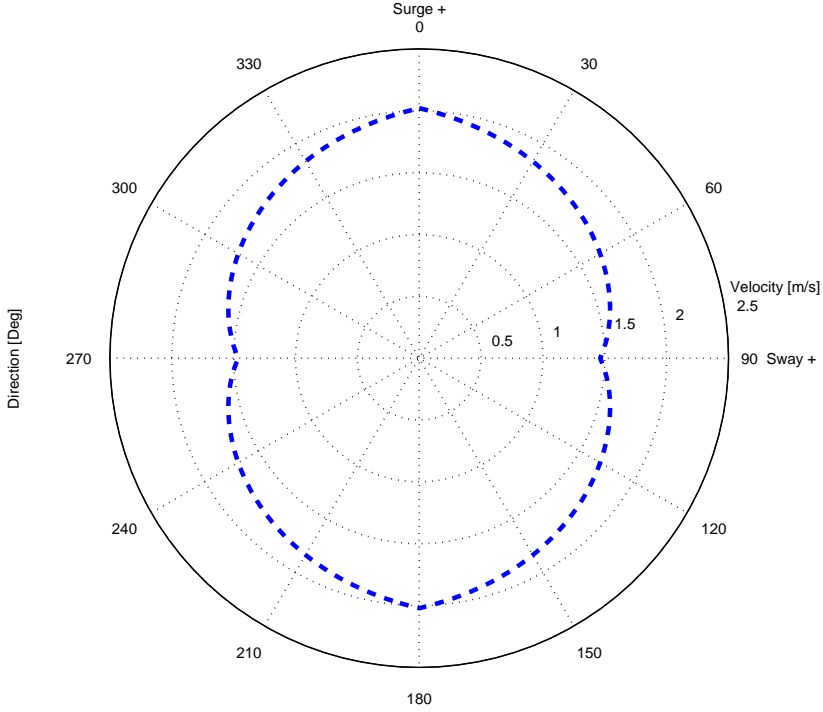


Figure 3.7: DP capability plot

3.5 DP Capability

Based on the results of a towing tank test and the known characteristics of thrusters and their location, a plot of the capabilities expected of Merlin has been created. Figure 3.7 shows this. How this plot is created can be seen in appendix E. Note that these are maximum values (margins should be included).

4 Sensor Setup and Availability

Some preliminary work was done in Knausgård [2012] in connection with which data were available directly from the sensor package. As previously mentioned, it is not TOGSNAV2 but TOGSNAV that will be used. This gives some advantages, as it is the device that is currently standard on Merlin. This will make a working system developed in this thesis that can be easily transferred to IKM's existing systems. The cost will also be reduced significantly because one will not have to purchase much more expensive equipment. Not having the device we had originally planned for does present some major challenges, as there is only the built-in estimator for the measurements of heading, roll and pitch (and the angular velocities). The rest is raw data or data that is processed to a limited extent. While TOGSNAV2 had a complete INS (inertial navigation system) and the ability to easily extract the required data for station keeping and maneuvering, this must now be done in the PLC to make it available for the control system. The challenge is first to extract relevant and available data from TOGSNAV that will enable estimating the required data to the overall objective of the thesis.

As one can see in the user manual (part of this can be found in Appendix F.1) to the TOGSNAV it is set up in standard mode and output a specific string of data in a specific frequency through a RS-232 port. This string does not as standard include all data needed to use it for a DP system. Both some angular velocities and velocity measurements over the seabed and the water flow (current) under the device are missing. (Note: Refer to Appendix F.1 for a complete overview of the data that are possible to extract from the sensor unit TOGSNAV.) To set up a custom string containing the data needed for the DP system, the recipe in the user manual is followed. A terminal program called Datamate that the supplier of TOGSNAV CDL has available for download on their website is used to set the settings. This proved to be less user friendly than anticipated and so it took a lot of time to set up the desired string to the thesis purpose. Presented below is what is written as the desired format of the string coming out of the device:

```
$IKM1,%hea:f06.2%,%pit:f+07.2%,%rol:f+07.2%,%erf%,%mod%,,%dep:f08.3%,  
%bvx:/1000:f+05.2%,%bvy:/1000:f+05.2%,%br1:/100:f06.2%,%br2:/100:f06.2%,  
%br3:/100:f06.2%,%br4:/100:f06.2%,%rvx:/1000:f+05.2%,%rvy:/1000:f+05.2%,  
%acx:*50:f+09.5%,%acy:*50:f+09.5%,%acz:*50:f+09.5%,%gwx:f+09.5%,  
%gwy:f+09.5%,%gwz:f+09.5%
```

4 SENSOR SETUP AND AVAILABILITY

Where percent signs and numbers are to scale and structure the string maintains a constant length, making it easier to read it into the PLC and the different elements which are read out are described below:

hea: heading (true north) in degree.

pit: pitch angle in degree.

rol: roll angle in degree.

erf: not in use

mod: not in use

dep: depth in meters.

bvx: velocity x-direction over seabed(body-seabed reference frame) in metre per second.

bvy: velocity y-direction over seabed(body-seabed reference frame) in metre per second.

br1...4: distance to seabed from the DVLs 1...4 beam in meter.

rvx: velocity in x-direction water reference layer(body-seabed reference frame) in metre per second.

rvy: velocity in y-direction water reference layer(body-seabed reference frame) in metre per second.

acx: acceleration in x-direction in metre per second squared.

acy: acceleration in y-direction in metre per second squared.

acz: acceleration in z-direction in metre per second squared.

gwx: angular velocity around x-axis in radian per second.

gwy: angular velocity around y-axis in radian per second.

g wz: angular velocity around z-axis in radian per second.

```

$IKM1,338.65,+004.65,+001.51,00,2,,0009.229,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79736,-00.25930,-09.77914,+00.00000,+00.00000,+00.00000
$IKM1,338.65,+004.65,+001.51,00,2,,0008.740,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79350,-00.25703,-09.79606,+00.00001,+00.00001,+00.00000
$IKM1,338.65,+004.65,+001.51,00,2,,0008.749,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79710,-00.25883,-09.78144,+00.00000,+00.00000,+00.00000
$IKM1,338.65,+004.65,+001.51,00,2,,0008.749,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79777,-00.25795,-09.79238,+00.00001,+00.00000,+00.00000
$IKM1,338.65,+004.65,+001.51,00,2,,0009.000,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79504,-00.25568,-09.78069,+00.00000,+00.00000,+00.00000
$IKM1,338.65,+004.65,+001.51,00,2,,0008.742,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79821,-00.25769,-09.79233,+00.00000,+00.00000,+00.00000
$IKM1,338.65,+004.65,+001.51,00,2,,0009.304,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79594,-00.25807,-09.79079,+00.00000,+00.00000,+00.00000
$IKM1,338.65,+004.65,+001.51,00,2,,0009.100,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79618,-00.25794,-09.80496,+00.00000,+00.00000,+00.00000
$IKM1,338.65,+004.65,+001.51,00,2,,0009.100,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79748,-00.25530,-09.80878,+00.00001,+00.00000,+00.00000
$IKM1,338.65,+004.65,+001.50,00,2,,0008.963,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79847,-00.25358,-09.78953,+00.00000,+00.00000,+00.00000
$IKM1,338.65,+004.65,+001.51,00,2,,0009.031,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79697,-00.25843,-09.78941,+00.00000,+00.00000,+00.00000
$IKM1,338.65,+004.65,+001.51,00,2,,0009.208,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79519,-00.25903,-09.78374,+00.00000,+00.00000,+00.00000
$IKM1,338.65,+004.65,+001.51,00,2,,0009.210,-9.99,-9.99,000.00,000.00,000.00,000.00,-9.99,-9.99,+00.79571,-00.25762,-09.79358,+00.00000,+00.00000,+00.00000

```

Figure 4.1: Screenshot of the custom IKM TOGSNAV string in a terminal program

In conversations with IKM Subsea (who create the logic that reads the string in the PLC) and by looking in the user manual, the frequency and baud rate of the string is set to 10Hz and a baudrate of 115200Bd.

Figure 7.1 shows a screenshot of a terminal program that reads the RS-232 string from the device when it is placed on land. One sees that the Doppler is unable to make any measurements on land, so the values will then automatically be set to 9.99 by the unit. Note that the same will occur if it loses contact with the bottom (bottom lock). This will be later be utilized in the estimator and the logic around the DVL.

As mentioned in Knausgård [2012], TOGSNAV (described was TOGSNAV2, but it is the same for TOGSNAV) is a sensor package consisting of several sensors. One of these is a DVL from RDI which is mounted inside the TOGSNAV. The communication between the processor unit in the TOGSNAV and DVL occurs via RS-232 and is set up by CDL. To communicate directly with the DVL in order to change settings on this, one must follow procedures described in Appendix F.1. Entering a specific command to TOGSNAV enables it to communicate directly with the DVL and thus bypass the TOGSNAV. Appendix F.1 rendered some important pages from the user manual to the DVL. Settings affecting the measurements of velocity over seabed and the Doppler frequency are supposed to be set correctly by CDL. Some thoughts on the different settings here are done in Knausgård [2012]. Possible settings to be changed are those related to the measurement of current in the water below the DVL. The reason for this is that CDL's standard setting assumed TOGSNAV is to be mounted on a ship. Therefore, the minimum distance to the seafloor that the Doppler can measure is too large for Merlin's use.

The settings are adjusted to the measurement of current in a given reference layer in the water. There are three parameters to be set for the DVL (Figure 4.2): how close will it be able to measure the velocity of the water and the thickness of the layer, and how far away should it measure. The yellow line in the illustration shows the minimum and maximum distance while, in the

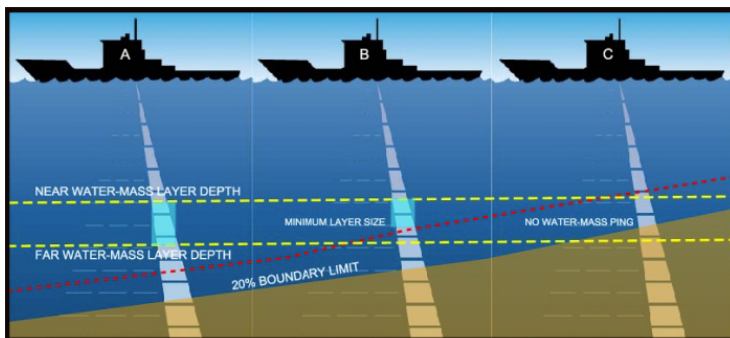


Figure 4.2: Water reference layer setup

middle, (illustrated by ship B) the minimum thickness of the layer is marked with a bright blue box. For Merlin it is desirable to have this as close as possible without creating disturbances in the measurement due to currents around the ROV itself. Note also that it is subject to a margin of 20 percent of the depth as an addition to the values set. By reading the manual one will see that the minimum distance from the DVL is 0.3 meters and minimum layer must be 0.1 meters. Accordingly, the maximum is minimum 0.4 meters. If the 20 percent is added, one sees that it can be difficult to get a current measurement when the distance to the bottom is short. This is something that needs to be tested. As a starting point this is the setting:

Min distance : 0.3m
 Min layer size : 0.2m
 Max distance : 0.6m

5 Control System

In this chapter, the goal is to provide a little insight into all the components of which Merlin's new control system consists. Figure 2.1 shows mainly how the overall design will look. Note that the size of the control system means that for the reader to get a really good insight, it must be seen together with the implemented system programmed modules.

5.1 Sensor Input and Transformation

The data from Merlin's sensors that were thought to be useful and necessary in developing a control system for Merlin were chosen in the chapter on sensor setup. Here, both available measurements and those transformed will be shown. Note that only measurements that are thought to be good enough to be used directly in the controller are included. For a complete listing of measurements, estimates as well as control inputs from the operator can be seen in table 7 and 8. How the string input of the RS-232 is done, this is completed by IKM Subsea and the thesis author has no knowledge about it except that the data become available in the PLC. For details, please refer to the I/O descriptions in Omron's user manual.

5.1.1 Directe

The direct available sensor inputs are shown in the section on sensor setup. With direct measurements, available means measurements are derived from an estimator. Those directly derived from an estimator can be seen in Table 4 (Note that they are shown together with the transformed measurements mentioned in the following section and are marked with an * symbol.). This is from the built-in estimator inside the TOGSNAV; see the chapter on the developed estimator for Merlin for which measurements must be estimated. Conversion including only conversion to SI-units is not discussed, but the measurements it applies to can be seen by comparing Table 4 and the unity of the relevant element in the string given in the sensor setup chapter.

5.1.2 Transformed

The Euler rate vector can be found using elementary use of rotation matrix and the assumptions made in the modeling chapter. Since all the parameters included are derived from the direct available as mentioned above, one can quickly see this is given by:

Table 4: Direct and transformed estimated measurements form TOGSNAV

Measurement	Unit	Reference frame
$\hat{\phi}$	rad	<i>NED</i>
$\hat{\theta}$	rad	<i>NED</i>
$\hat{\psi}$	rad	<i>NED</i>
$\hat{\phi}^*$	rad/s	<i>NED</i>
$\hat{\theta}^*$	rad/s	<i>NED</i>
$\hat{\psi}^*$	rad/s	<i>NED</i>
\hat{p}	rad/s	<i>Body</i>
\hat{q}	rad/s	<i>Body</i>
\hat{r}	rad/s	<i>Body</i>

$$\hat{\Theta}_{nb} = \begin{bmatrix} \hat{\phi} & \hat{\theta} & \hat{\psi} \end{bmatrix}^T = \mathbf{T}_{\Theta}(\hat{\Theta}_{nb}) \begin{bmatrix} \hat{p} & \hat{q} & \hat{r} \end{bmatrix}^T \quad (5.1)$$

where eq. 3.1 has been used.

Summarized in Table 4 are all direct\indirect measurements available that can be used in the controller. Those marked with * are transformed. The notation with a “hat” over the variable highlights that it is derived from an estimated measurement (i.e., it does not necessarily mean that it cannot be measured; estimators are also used to fix noisy measurements).

5.2 Discretization

All physical dynamics are continuous, but computer calculations, and consequently for a control system, are inherently discrete. In mathematics, discretization concerns the process of transferring continuous models and equations into discrete counterparts. This transferring is normally carried out as a first step toward making them suitable for numerical implementation on digital computers. Something that will not be mentioned here is the actual sampling as any digital system with sensor data input readings must do. This is often referred to as quantization and is very important to know, but not of great importance in this thesis as sensor readings and problems associated with this will be done in the sensor devices themselves and the implementation of the control system will be at higher levels. Below are the methods used for operations in the implementation of the control system. Note that discretization for the extended Kalman filter is mentioned in its own section. See section on state observable for more.

5.2.1 Numerical Integration Method

A frequently used method that will be used for integration (the few places it is necessary) in this thesis is Euler's method or, more precisely, forward Euler. Given a nonlinear time varying system of the form:

$$\dot{x} = f(x, u, t)$$

Using forward Euler, the integrated can be written as:

$$x(k+1) = x(k) + h f(x(k), u(k), t_k) \quad (5.2)$$

where h is the sampling interval and k index of sample function.

5.2.2 Numerical Differentiation Method

Numerical differentiation is usually sensitive to noisy measurements. Mentioned by Fossen [2011], a reasonable estimate can be obtained using filtered differentiation. Obviously, there must be consideration of what one should differentiate.

If the dynamics are very fast and it is considerably with the noise, then it cannot be used. The simplest filter is obtained by the first-order low-pass structure:

$$\dot{\eta}_f(s) = \frac{s}{Ts + 1} \eta(s) \quad (5.3)$$

where $\dot{\eta}_f$ is the estimate of the time derivative $\dot{\eta}$ of a signal η .

Note that the reference model described later uses a form of this in a mass-damper-spring system.

5.3 State Estimators

A state estimator or a state observer is a system that provides an estimate of the internal state of a given real system. It uses measurements of the input and output of the real system. It may be impossible to measure the state, or impractical or expensive or it may be that a measure is strongly influenced by noise and must be estimated in order to be used in a controller. If a system is observable (see e.g., Fossen [2011] for definition), it is possible to fully reconstruct the system state from its output measurements using the state observer. There are a number of estimators used that will not be discussed here. Please refer to numerous amounts of literature on the field. Only estimators that have been used for similar systems and have achieved good results will be discussed. It is also essential that the estimator can handle the nonlinearities that Merlin clearly showed it had in the experiments in Knausgård [2012].

5.3.1 Observer Design for Dynamic Positioning

Filtering and estimation are important features for most control systems, but there are some characteristics that are more important than others in the DP context. As mentioned in Sørensen [2012], there are a few points that are important:

- Reconstruction of non-measured data: For the use of Merlin as it is now, it does not have access to any global reference system, so it is desirable to estimate a position locally in relation to the bottom.
- Filter: Filtering the measured states that are associated with noise, e.g., Doppler velocity log.

- **Dead reckoning:** Ability to give out useful estimates of the states despite temporary loss of measurements, such as if the velocity measurement of the bottom disappears for a few seconds. It is known that most sensors are strongly linked and influencers of the environment have an error rate that make it very important that the system remains available during this time. For Merlin, the Doppler velocity log is such a device, while ships often have satellite systems, wind sensors, etc.

Of the methods used and used for these purposes, there are two that stand out:

- **Kalman filter design:** The Kalman filter is a widely used algorithm that has been around for more than 50 years since R.E. Kalman's research work was presented in 1960 in a paper entitled "A New Approach to Linear Filtering and Prediction Problems". Cadet [2003] summarizes the historical development as follows: "R.E. Kalman had the idea of applying the notion of state variables to the Wiener filtering problem. The first application of the Kalman filter was in aerospace when R.H. Battin made the Kalman filter part of the Apollo onboard guidance. In 1976, J.G. Balchen, N.A. Jenssen and S. Saelid wrote a paper on Dynamic Positioning Using Kalman Filtering and Optimal Control Theory. This new approach based on the concept of modern control theory was aimed at addressing the disadvantages of PID - controller, such as slow integral action and phase lag in the control loops. Since then Kalman filtering has been widely used in dynamic positioning applications." The Kalman filter is an efficient recursive filter that estimates the state of a linear (or nonlinear as seen below) dynamic system from a series of noisy measurements. It is a proven optimal recursive data processing algorithm optimal with respect to minimum variance (for linear systems). The Kalman filter incorporates all information that can be provided to it, processes all available measurements, and combines the data to generate the overall best estimate of the states. Temporary loss of measurements will lead the filter to work as a pure predictor.
- **(Nonlinear) Passive observer design:** The nonlinear observer is motivated from passivity arguments(see Fossen and Strand [1999]) and is proven to be globally asymptotic stable (GES), ensured through passive design. In contrast to all the parameters that must be set in the Kalman filter, which are not strongly related to the physics of the system, there are far fewer of the passive observed with these. Tuning is also much more closely connected to the physics of the system from which a more intuitive approach

can be provided. It will therefore reduce the tuning time significantly. The nonlinear passive observer has been simulated on computer models of supply vessels and implemented on full-scale ships with excellent results (Fossen [2011]).

The Kalman filter has been selected as the observer design and will be implemented in the control system. There are several reasons for this: Drag-tests and CAD-drawing of Merlin mean that the mathematical model is assumed to be very good. It also limits the states to be estimated since the TOGSNAV device has already had an observer for some of the states that are included in the model that will be used in the estimator (heading, roll, pitch and angular velocities). Ability to implement and test the observed data logged from the sensor in, for example, Simulink, means that the tuning required is assumed to be affordable. The author's knowledge of and familiarity with the Kalman filter also helps in choosing this design. Maintainability of code and a control system is assumed to be simpler for IKM Subsea if the Kalman filter is used due to greater knowledge of this in most companies and automation communities.

Since the Kalman filter is derived from stochastic theory, it is also the type of system for which it is best suited. Unlike a deterministic system in which all initial conditions and excitations are known in advance, a stochastic system fits nicely for description of the behavior of waves and wind and the measurement error that, for example, a Doppler velocity log has. An important consideration here is that dynamic systems excited by stochastic processes such as Merlin will, as a result, be stochastic systems. In other words, it is well suited for a Kalman filter. Figure 5.1 shows a block diagram of the Kalman filter as a state estimator. As mentioned in both Knausgård [2012] and earlier in the thesis, the mathematical model of Merlin is nonlinear, meaning the simplest form of the Kalman filter cannot be used without some modifications. These modifications will remove some properties of the filter, but is necessary in order to use it.

5.3.2 Extended Kalman Filter

The extended Kalman filter (EKF) is the nonlinear version of the Kalman filter that linearizes an estimate of the current mean and covariance. In the case of well-defined transition models, the EKF has been considered the de facto standard in the theory of nonlinear state estimation, navigation systems and GNSS (Wan [2006]). EKF is just extending the KF to nonlinear systems by linearizing the nonlinearities around the current estimate. So the EKF is not a purely nonlinear approach. The linearizing of the system is done to fit it into

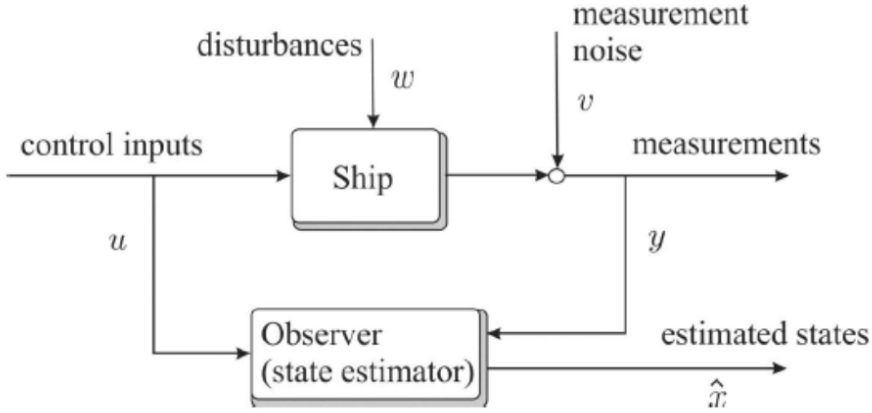


Figure 5.1: Block diagram of the Kalman filter as a state estimator

the frame of the linear KF. This will naturally give a poorer performance, as the original dynamics of the system are not being preserved in a complete way. Note that this is not the case for a nonlinear observer; in a nonlinear observer the original dynamics are preserved (with a few assumptions), giving better stability properties. However, using the EKF is closer to preserving the original dynamics than linearizing the system beforehand and using the regular Kalman filter, which is an alternative. This is because the linearization is updated as the estimate is updated. The EKF is not an optimal estimator because the linearization is only valid around the point of linearization and the filter may diverge if the initial state estimate is wrong or if it has an incorrect model. The starting point for both the regular and the extended Kalman filter is a mathematical model of the stochastic system encountered. The better the model, the better the obtained result. But it is clear that trade-offs in complexity and performance must be made. Various models are listed in Sørensen [2012], such as a simulation purpose requiring a high degree of complexity in the model or smaller control plant models that include only the most important characteristics of the system for control use (e.g., DP). The mathematical model can be written in state space form as two parts, one for dynamic and one for measurements. Such a nonlinear state space model is presented here:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} \\ \mathbf{y} &= \mathbf{H}\mathbf{x} + \mathbf{v}\end{aligned}\tag{5.4}$$

where $\mathbf{f}(\mathbf{x})$ is a nonlinear vector field. A function which is somewhat equivalent to the system matrix with the state vector $\mathbf{A}\mathbf{x}$ in a linear system, \mathbf{B} is the control input vector, \mathbf{H} is the measurement matrix, \mathbf{E} is the process noise matrix and \mathbf{v} the measurement noise vector.

It is only the discrete case, which will be further discussed and studied. Table 5 shows the discrete-time extended Kalman filters equations. $\mathbf{Q} = E(\mathbf{w}^T \mathbf{w})$ and $\mathbf{R} = E(\mathbf{v}^T \mathbf{v})$ are the covariance weight matrices for, respectively, process and measurement noise (used for tuning the filter; see section on tuning). The Kalman filter works in a loop as illustrated in Figure 5.2. Note that a slightly different notation on the measurements is used.

Table 5: Discrete-time EKF

Design matrices	$\mathbf{Q}(k) = \mathbf{Q}(k)^T > 0, \mathbf{R}(k) = \mathbf{R}(k)^T > 0$
Initial conditions	$\bar{\mathbf{x}}(0) = \mathbf{x}_0, \bar{\mathbf{P}}(0) = E[(\mathbf{x}(0) - \hat{\mathbf{x}}(0))(\mathbf{x}(0) - \hat{\mathbf{x}}(0))^T] = \mathbf{P}_0$
Kalman gain matrix	$\mathbf{K}(k) = \mathbf{P}(k)\mathbf{H}^T(k)[\mathbf{H}(k)\bar{\mathbf{P}}(k)\mathbf{H}^T(k) + \mathbf{R}(k)]^{-1}$
State estimate update	$\hat{\mathbf{x}}(k) = \bar{\mathbf{x}}(k) + \mathbf{K}(k)[\mathbf{y}(k) - \mathbf{H}(k)\bar{\mathbf{x}}(k)]$
Error covatiance update	$\hat{\mathbf{P}}(k) = [\mathbf{I} - \mathbf{K}(k)\mathbf{H}(k)]\bar{\mathbf{P}}[\mathbf{I} - \mathbf{K}(k)\mathbf{H}(k)]^T + \mathbf{K}(k)\bar{\mathbf{R}}(k)\mathbf{K}^T(k)$
State estimate propagation	$\bar{\mathbf{x}}(k+1) = \mathcal{F}(\hat{\mathbf{x}}(k), \mathbf{u}(k))$
Error covatiance propagation	$\bar{\mathbf{P}}(k+1) = \Phi(k)\hat{\mathbf{P}}(k)\Phi^T(k) + \Gamma(k)\mathbf{Q}(k)\Gamma^T(k)$

The discrete-time quantities can be found by using forward Euler integration:

$$\mathcal{F}(\hat{\mathbf{x}}(k), \mathbf{u}(k)) = \hat{\mathbf{x}}(k) + h[\mathbf{f}(\hat{\mathbf{x}}(k)) + \mathbf{B}\mathbf{u}(k)]\tag{5.5}$$

$$\Phi(k) = \mathbf{I} + h \left. \frac{\partial \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k))}{\partial \mathbf{x}(k)} \right|_{\mathbf{x}(k) = \hat{\mathbf{x}}(k)}\tag{5.6}$$

$$\Gamma(k) = h\mathbf{E}\tag{5.7}$$

Note again that there is no proof of global asymptotic stability when the system is linearized.

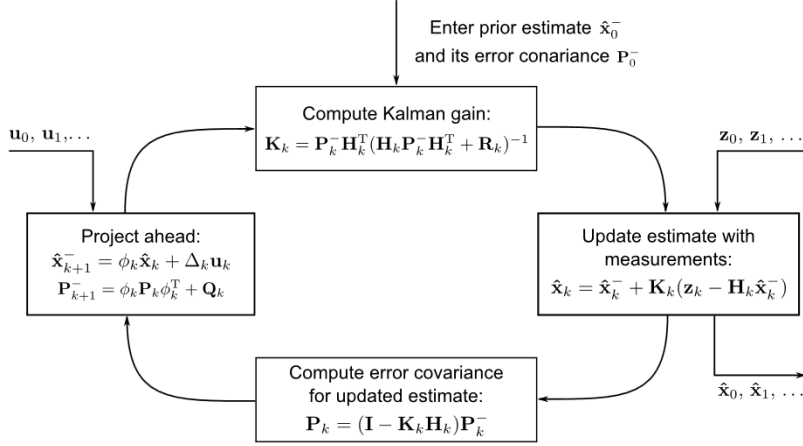


Figure 5.2: The Kalman filter loop. (Brown & Hwang 1997)

5.3.3 Testing and Tuning

Kalman filter performance can be tuned and, therefore, improved by adjusting the process noise \mathbf{Q} and the measurement noise covariance \mathbf{R} . To simplify tuning and to make it a little more intuitive to tune \mathbf{Q} and \mathbf{R} are chosen to be diagonal. This is normal and Sørensen [2012] talks about using Bryson's inverse square method to find these. As seen in Table 5, adjusting these two values will have consequences on the Kalman gain. If \mathbf{Q} is too large, then the Kalman gain will be too high and, as a result, the estimates will have a tendency to follow the measurements “too much” and will bounce around a lot. By choose \mathbf{Q} too small, exactly the opposite will occur. If \mathbf{R} is too large, the filter will not take into account the new measurements as much as it should. And vice versa. Note that if \mathbf{Q} and \mathbf{R} are constant (as is the case for this thesis) the estimation error covariance and the Kalman gain will stabilize quickly and then remain constant. However, one sees that it is possible to update these online. For example, different errors (noise) for the Doppler velocity log that depend on the distance to the bottom could be added to the model.

For the tuning it is assumed that the mathematical model of Merlin is satisfactory and that it is primarily the variance of the DVL that must be found. Small model errors will be compensated by a bias model in which one will see the implementation of the filter and, consequently, the corresponding parameter

in the process noises matrix will have great value.

5.3.4 Variables to be Estimated

Regarding which measurement is to be estimated, this will be shown in the chapter on implementation. Table 7 also shows which values have been estimated from the estimator as well as the few values that were directly available from the TOGSNAV.

5.4 State Machines

State machines or finite state machines is a method to describe systems with logical and dynamic/temporal behavior. The method results in a model called a state machine. This model is a tool to be used when one structures programming of more complex systems. The model state machine comprises a variety of conditions and events that the system changes from one state to another as well as action that is a result of events. The method used in application development in areas such as logical/digital control systems and real time systems. Key concepts in a state machine are states, events and actions. States is a term used to describe the system status/condition. States is a set of values/attributes describing the system properties. Events is a term used for inputs/impacts on the system. Events can be described as a sudden short lasting impact on the system. Actions are what comes out of the system, the result. An action is a response to an event.

There are several forms of state machines for how the graphic is produced. Unified Modeling Language(UML)) is a general-purpose modeling language in the field of software engineering that give extended possibilities in the preparation of a system as a state machine. For a proper review of this language, please refer to the literature in the field. The graphic representation of state machines in this paper can be viewed as a quasi UML representation where a few elements from the UML is used. This is expected to be self-explanatory to the reader.

5.4.1 DVL Bottom Lock and Suitability

By studying the user manual for TOGSNAV as well as what was mentioned in the chapter on sensor setup, one can see there are a number of limitations for a DVL. First, there is both a minimum and maximum distance to the seabed as required for it to give out velocity measurements. In addition, stones, metal

installations, rock formations, etc., can cause the measurement to fall for short or long periods. (Note that the DVL works with the measurement/fail test.) It is thus important to be aware that the entire system can come to a halt because of it. However, it is also important that there is a limit to how unstable a measurement can be before it is considered unusable. Consider two examples that illustrate this: 1. A measurement that has been stable for a long time falls for one second and then comes back to a stable situation. 2. A measurement falls in and out for small intervals over a longer period. In the first example, it would be meaningless to stop operation because of a little downtime, while in Example 2, there is much more uncertainty about what is correct. How long did the measurement fall out within a time interval? What is the uptime? There is clearly a trade-off between accuracy requirements versus uptime.

To show this, a state machine has been made which illustrates how it is done for Merlin. Note that the main use for this logic is for the estimator developed, hence the references to this (see information on dead reckoning in the part with state estimator). Operation explained briefly: the DVL logic requires a certain stable uptime to start; when this time is reached, it is believed that the estimator (EKF) is initialized and running. It is made a logic to ensure that small drop out of measurements can be tolerated, while if they are too long or get to stuttering the logic will say that the measurements from the estimator are not good enough to be used.(e.g. the system will not be able to perform station keeping) . The DVL logic will than again require a certain stable uptime to approve the estimates.

5.4.2 Control System

Figure 5.4 shows a very extensive state machine displaying an overview of the complete control system to be implemented in Merlin. It is best read in conjunction with the controller section.

◊ is a pseudo-state where a choice is taken.

Note that the models found for the control system to Merlin are models that not will be implemented as a state machine in the system. However, the behavior will be identical. There are methods to program that could have provided a tighter connection to the state machine as the use of "case", etc. But it will not be used beyond the fact that it is seen as a great way to develop, structure and give a good overview of the system.

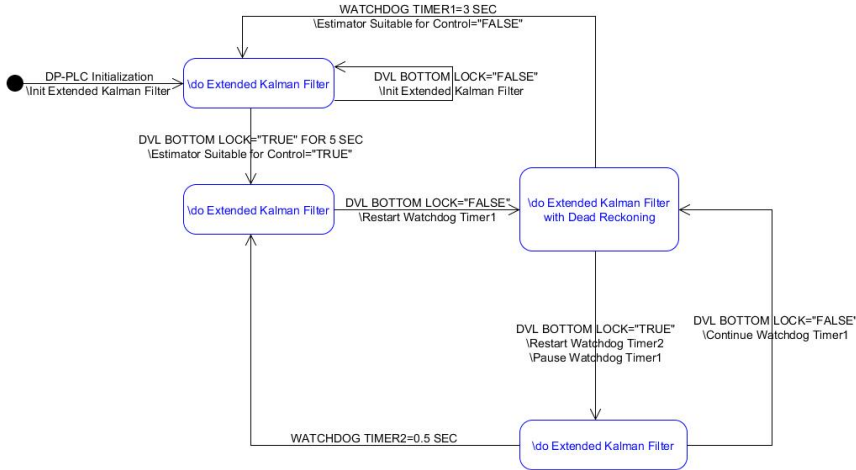


Figure 5.3: State machine for the DVL

5.5 Human Machine Interface

5.5.1 Control Interface

Since the very beginning of the development of DP system, the idea to create an intuitive system has played a key role. Perhaps the area most in need of this relates to the design of the interface with the operator. Questions such as what should be automated, how should the interaction take place, etc., are crucial issues. The solution to this was a combination of suggestions from the ROV pilots and, almost the opposite, someone who has never touched an ROV or has an idea about how it should be operated. ROV pilots tend to want to keep the system as it is today. They have learned to operate the ROV by trial and error. Involuntary or not, they have (from an engineering point of view) taken on tasks that could have been solved by a control system. For instance, they manual give an extra thrust in the start of a movement like feed forward of acceleration generated by a reference model could do instead. The author may be a candidate to represent a user without experience that want to control the ROV; the author want an intuitive control interface, i.e., pushing the joystick forward to move the ROV forward at a velocity equal to the distance it is pushed

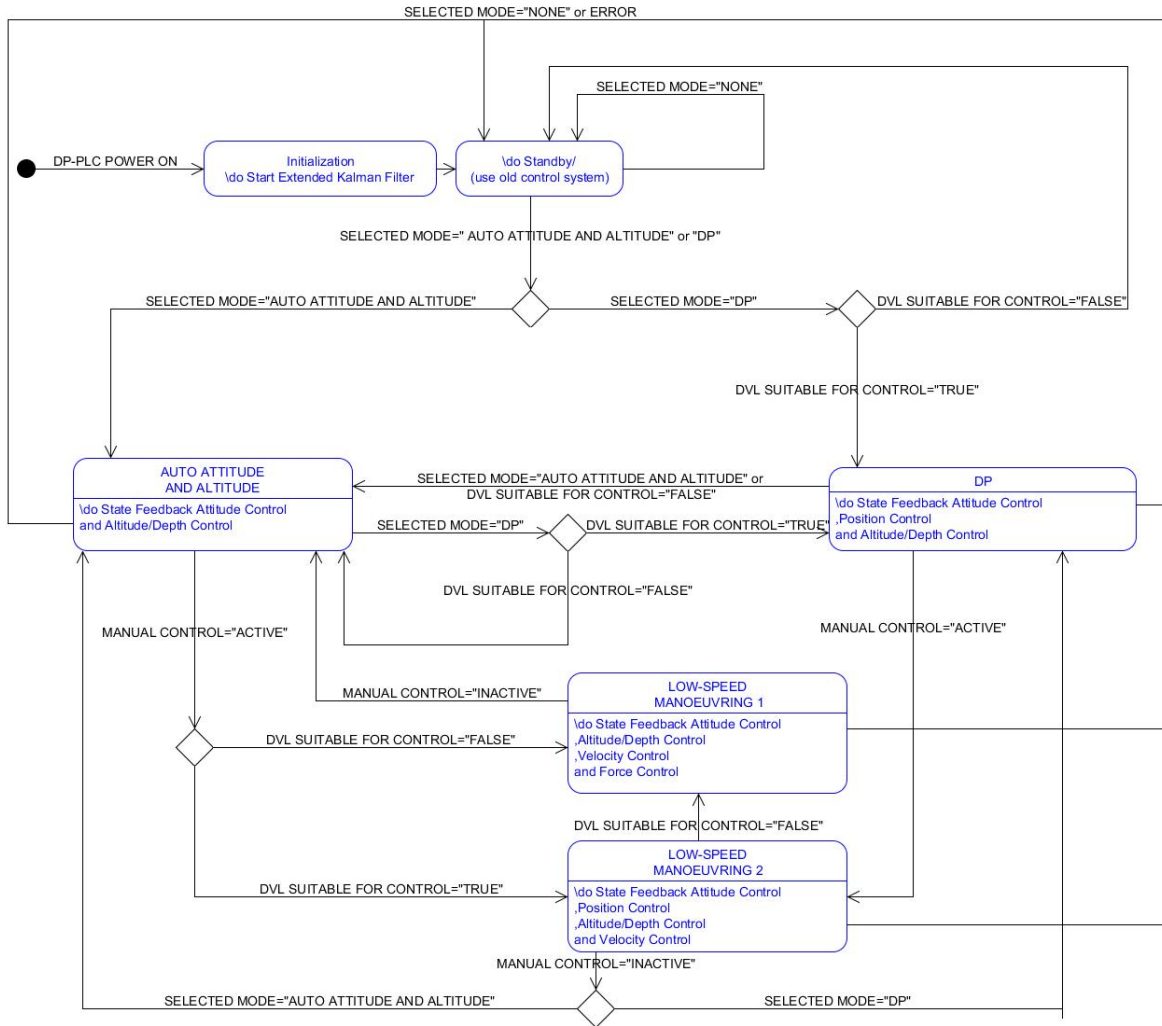


Figure 5.4: State machine for the DP control system

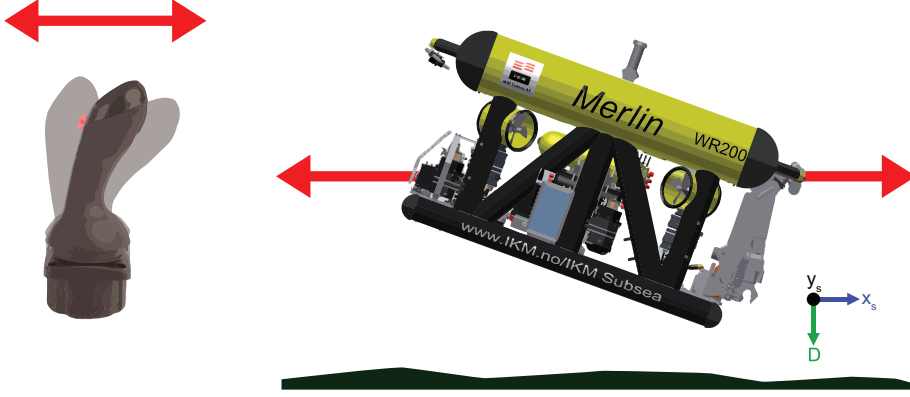


Figure 5.5: Control input for the body-seabed frame

Table 6: Control inputs from operator

Measurement	Unit	Reference frame
u_{sd}	m/s	<i>Body – Seabed</i>
v_{sd}	m/s	<i>Body – Seabed</i>
\dot{D}_d	m/s	<i>NED</i>
ϕ_d	rad	<i>NED</i>
θ_d	rad	<i>NED</i>
$\dot{\psi}_d$	rad/s	<i>NED</i>

forward. This is called rate control and will be used for the implementation for Merlin and could be used in all the relevant degrees of freedom.

But this is not so simple, since Merlin operates under water and can take many different attitudes. Thus arises a new question: Which reference system will be used to control input? It could have been useful to be able to choose between different systems, but a choice was made that the most intuitive system is to use the body-seabed frame as mentioned in the modeling section. This is illustrated in Figure 5.5 and 5.6.

5.5.2 Development Interface

A simple HMI interface using Omron touch screen for control system use is made for the sea trials. It was developed in close collaboration with IKM Subsea.

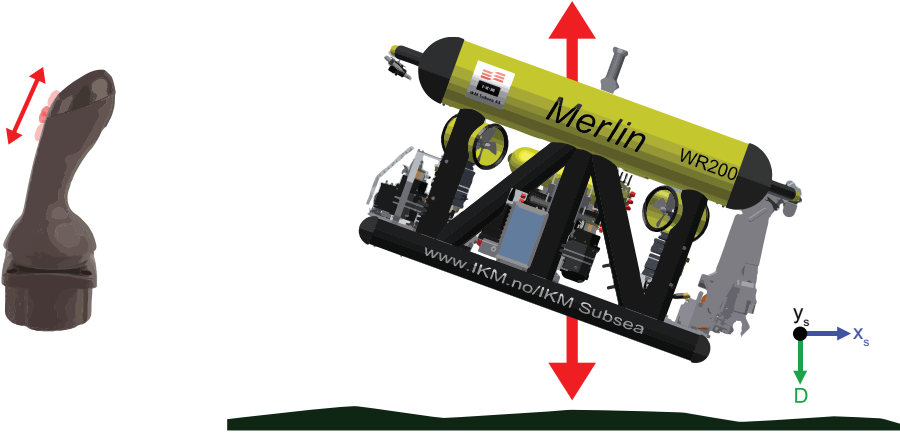


Figure 5.6: Control input in the body-seabed frame

This is shown in Figure 7.6. It is possible to change the mode of the controller (e.g., force control or DP). It is possible to tune and set the parameters of the controller online. There is access to read the various data that is useful in developing desired force in the different degrees of freedom and the status of the bit indicating whether DVL is in contact with the bottom (required for using the DP mode). Note that need for custom software and prioritization made IKM Subsea implement this and it is not attached or described further.

5.6 Reference Models

A reference model between raw inputs from the joystick and turn-knob to adjust the roll and pitch is desirable to ensure that these inputs will be smooth and feasible for the controller. For control of the very slowly varying input as the reference to the desired roll and pitch will be, it is sufficient to use a low-pass filter which only filters the desired angle. Velocities in surge, sway, heave and heading-rate will require a slightly more sophisticated solution that ensures a smooth and feasible reference for velocities and provides a smooth and feasible reference for the desired acceleration. In addition, saturation for acceleration will be added. To create a model-based controller able to switch between station keeping and maneuvering as well as providing an intuitive user interface, it is crucial to have a sort of forward coupling (feed forward) of acceleration. For Merlin's application, it is assumed that a second order mass-damper-spring sys-

tem as mentioned in Fossen [2011] can serve as a sufficient model. An important characteristic of a useful reference model is that the bandwidth of it must be lower than for the ROV's motion control system. Reference model uses inputs as shown in Table 6 and outputs the values that can be seen in Table 8. Note that there are some rotation matrices involved to get the desired velocities in the body- fixed frame.

5.6.1 Velocity and Acceleration

For velocity and acceleration the following reference model will be used:

$$\ddot{\nu}_d + 2\Delta\Omega\dot{\nu}_d + \Omega^2\nu_d = \Omega^2r^b \quad (5.8)$$

$$\lim_{x \rightarrow \infty} \nu_d(t) = r^b \quad (5.9)$$

where r^b denote the raw desired operator input, ν_d the smooth desired velocity and $\dot{\nu}_d$ the smooth desired acceleration ($\ddot{\nu}$ is interpreted as the desired “jerk”, and only used as a part of the mass-daper-spring system). Figure 5.7 shows the behavior of the reference model for a raw operator commanded velocity input.

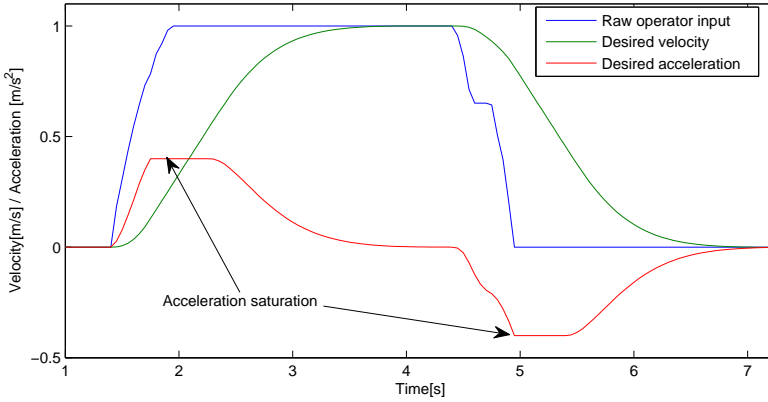


Figure 5.7: Example of reference model for Merlin

5.6.2 Attitude

As mentioned, one can set the desired roll and pitch angle using two turn-knobs. It is desirable to only allow very small velocities in these directions. It provides motivation to use the simplest form of a reference model; first order low-pass filter. The desired velocity and acceleration are set to zero. A low-pass filter structure will be used for implementation.

5.6.3 Position

There is no need for a reference model for position control since this should be kept to a fixed set point set when the system locks into a position that is desirable when the stick is released and the system is in the correct mode. But it could be necessary if it were possible to use set-points.

5.6.4 Active

A problem that is not as extensive but which is important for implementation of the controller is to have a clear definition of when the manual control is active. This can be illustrated by the following example: Merlin is running at full speed in the surge direction, then the stick is released quickly. Merlin will then lock at the current position. Should one see on the raw value from the stick or should one look on the smooth reference value to determine if it is active or not. (It will be to see if it is zero or not.) It is clear that the last one is preferable, but one has to wait until it has reached zero to do that. It could mean that Merlin locks into a desired position too late (remember that it will take a long time before the reference will be completely zero). The solution here is to define a threshold on the output of the reference model that defines whether it is active or not. In addition, there must be a requirement that the desired acceleration must be below a certain threshold.

5.7 Controller

Before it became clear that the promised sensor was not available, the plan was to develop multiple controllers using different designs and theories. But with limited time for this thesis, a decision was made to choose the design that the author assumed would both provide the best behavior with the conditions and could be easily implemented in the PLC. Since it is assumed that the model of Merlin is very good and the measurements is good, it is relevant to consider that a design with state feedback of the dynamics is suitable.

Fossen [2011] presents some state feedback linearization designs that are suitable for velocity, attitude and position control. Nonlinear design will make the process more orderly as nonlinear terms are canceled and, thus, result in a linear system. Note that by using other design methods one could use the nonlinear damping as good damping to the system (i.e., backstepping). To be able to keep track of the system being developed here, it will be useful to rely on the state machine shown in Figure 5.4. While Fossen [2011] only gives examples of one and one design for different objectives for state feedback linearization, the design for Merlin combines several. An effort has been made to keep the different controllers as separate as possible and work for development of an elegant design of the logic associated with the smooth transition between them. While many designs of control systems require an operator actively participating in setting the system in the different modes for application, the design presented here seeks to create a more robust controller.

5.7.1 Available Data for Control

Tables 7 and 8 display a complete overview of available sensor data and control input for controller. Note that Table 7 also demonstrates that any measurements are only available when DVL is suitable for controlling (as the estimator has dependencies). In addition, data from the logic associated with the DVL's state machine and the HMI interface (e.g., parameters to the controller and selected mode) are available.

5.7.2 State Feedback Linearization

The concept of state feedback linearization is to transform nonlinear systems dynamics into a linear system. This is achieved by canceling the nonlinearity and then control the system remaining with use of basic control techniques. This method stems from robotics where it is often called computed torque control. The method is well suited (Fossen [2011]) for ships and underwater vehicles since

Table 7: Available measurements(from both TOGSNAV direct and estimator)

Measurement	Unit	Reference frame	Availability
\hat{N}	m	<i>NED</i>	DVL must be suitable for control. See 7.1
\hat{E}	m	<i>NED</i>	DVL must be suitable for control. See 7.1
\hat{D}	m	<i>NED</i>	Constantly
$\hat{\phi}$	rad	<i>NED</i>	Constantly
$\hat{\theta}$	rad	<i>NED</i>	Constantly
$\hat{\psi}$	rad	<i>NED</i>	Constantly
$\hat{\dot{N}}$	m/s	<i>NED</i>	DVL must be suitable for control. See 7.1
$\hat{\dot{E}}$	m/s	<i>NED</i>	DVL must be suitable for control. See 7.1
$\hat{\dot{D}}$	m/s	<i>NED</i>	Constantly
$\hat{\dot{\phi}}$	rad/s	<i>NED</i>	Constantly
$\hat{\dot{\theta}}$	rad/s	<i>NED</i>	Constantly
$\hat{\dot{\psi}}$	rad/s	<i>NED</i>	Constantly
\hat{u}	m/s	<i>Body</i>	If DVL not suitable for control it will depend on $\hat{\dot{D}}$
\hat{v}	m/s	<i>Body</i>	If DVL not suitable for control it will depend on $\hat{\dot{D}}$
\hat{w}	m/s	<i>Body</i>	If DVL not suitable for control it will depend on $\hat{\dot{D}}$
\hat{p}	rad/s	<i>Body</i>	Constantly
\hat{q}	rad/s	<i>Body</i>	Constantly
\hat{r}	rad/s	<i>Body</i>	Constantly
\hat{u}_c	m/s	<i>Body</i>	Constantly
\hat{v}_c	m/s	<i>Body</i>	Constantly

Table 8: Available control inputs from reference model

Measurement	Unit	Reference frame
u_d	m/s	<i>Body</i>
v_d	m/s	<i>Body</i>
w_d	m/s	<i>Body</i>
p_d	rad/s	<i>Body</i>
q_d	rad/s	<i>Body</i>
r_d	rad/s	<i>Body</i>
ϕ_d	rad	<i>NED</i>
θ_d	rad	<i>NED</i>
$\dot{\phi}_d$	rad/s	<i>NED</i>
$\dot{\theta}_d$	rad/s	<i>NED</i>
$\dot{\psi}_d$	rad/s	<i>NED</i>
\dot{u}_d	m/s ²	<i>Body</i>
\dot{v}_d	m/s ²	<i>Body</i>
\dot{w}_d	m/s ²	<i>Body</i>
\dot{p}_d	rad/s ²	<i>Body</i>
\dot{q}_d	rad/s ²	<i>Body</i>
\dot{r}_d	rad/s ²	<i>Body</i>

these are basically represented as nonlinear mass-damper-spring systems. The fact that the method can be easily transformed to control velocity, attitude and position makes it ideally suited to this thesis. Below is a brief introduction to the key concepts:

Given a description of a vessel of this form:

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{n}(\boldsymbol{\nu}, \boldsymbol{\eta}) = \boldsymbol{\tau} \quad (5.10)$$

That fits nicely with the model that was found for Merlin. The function \mathbf{n} is a nonlinear function as in the case of Merlin is a function containing nonlinear damping and hydrostatics. (See the developed mathematical model of the Merlin.) These are assumed to be known.

For velocity control (body-fixed control), the following can be done by remembering that $\dot{\boldsymbol{\nu}}$ is acceleration:

$$\boldsymbol{\tau} = \mathbf{M}\mathbf{a} + \mathbf{n}(\boldsymbol{\nu}, \boldsymbol{\eta}) \quad (5.11)$$

By using \mathbf{a} as input (commanded acceleration vector) one can by putting 5.11 into 5.10 find the following error dynamics:

$$\mathbf{M}(\dot{\boldsymbol{\nu}} - \mathbf{a}) = 0 \quad (5.12)$$

By select the input \mathbf{a} as shown in Fossen [2011] as a PI controller with feed forward of the the desired acceleration \mathbf{a} can be written as:

$$\mathbf{a} = \dot{\boldsymbol{\nu}}_d - \mathbf{K}_p \tilde{\boldsymbol{\nu}} - \mathbf{K}_i \int_0^t \tilde{\boldsymbol{\nu}}(\tau) d\tau \quad (5.13)$$

where $\tilde{\boldsymbol{\nu}} = \boldsymbol{\nu} - \boldsymbol{\nu}_d$. Remember that the subscript d marks the desired values, and consequently inputs here. These are all assumed to be smooth and here derive from a reference model. By inserting 5.13 into 5.12 and choose a simple

pole placement method as described in Fossen [2011] choose the diagonal gain matrix \mathbf{K}_p and \mathbf{K}_i of the linear system like this:

$$\begin{aligned}\mathbf{K}_p &= 2\Lambda \\ \mathbf{K}_i &= \Lambda^2\end{aligned}$$

where $\Lambda > 0$ is a diagonal design matrix used for tuning the system.

When the state feedback linearization will be used to control the attitude and position, some considerations around kinematics must be done. It is clear that all actuation occurs in the body frame, while the control objective will be defined in the NED-frame. Acceleration in the NED-frame can be expressed using the body-fixed acceleration in a simple way by making some simplification that must be done with caution.

By setting \mathbf{a}^n as the as desired input into the system (defined in the NED-frame) as

$$\ddot{\eta} = \mathbf{a}^n \quad (5.14)$$

Remember from the modeling chapter that the relationship between the velocities in the body and the NED-frame can be written as:

$$\dot{\eta} = \mathbf{J}_\Theta(\eta)\nu \quad (5.15)$$

where $\mathbf{J}_\Theta(\eta)$ contains both the rotation and the transformation matrix. Making the assumption that the angular velocities are so small that the time derivative of $\mathbf{J}_\Theta(\eta)$ is zero, a relationship between acceleration in the body and the NED-frame can be set up:

$$\ddot{\eta} = \mathbf{J}_\Theta(\eta)\dot{\nu} \quad (5.16)$$

As can easily be seen, it is the same as:

$$\mathbf{a}^n = \mathbf{J}_\Theta(\eta)\mathbf{a} \quad (5.17)$$

\mathbf{a} can then easily be described using the desired input \mathbf{a}^n and then used as in 3.21:

$$\mathbf{a} = \mathbf{J}_\Theta^{-1}(\eta)\mathbf{a}^n \quad (5.18)$$

$$\tau = \mathbf{M}\mathbf{J}_\Theta^{-1}(\eta)\mathbf{a}^n + \mathbf{n}(\nu, \eta) \quad (5.19)$$

By select the input \mathbf{a}^n as shown in Fossen [2011] as a PID controller \mathbf{a}^n can be written as:

$$\mathbf{a}^n = -\mathbf{K}_d\dot{\tilde{\eta}} - \mathbf{K}_p\tilde{\eta} - \mathbf{K}_i \int \tilde{\eta}(\tau)d\tau \quad (5.20)$$

where $\tilde{\eta} = \eta - \eta_d$. By choosing a simple pole placement method as described in Fossen [2011] the diagonal gain matrix $\mathbf{K}_d, \mathbf{K}_p$ and \mathbf{K}_i of the linear system can be choosen like this:

$$\begin{aligned} \mathbf{K}_d &= 3\Lambda \\ \mathbf{K}_p &= 3\Lambda^2 \\ \mathbf{K}_i &= \Lambda^3 \end{aligned}$$

where $\Lambda > 0$ is a diagonal design matrix used for tuning the system.

5.7.3 Modes

Merlin will operate in several modes depending on the available measurements and desired mode for the operator. The main differences are if the DVL provides the data or not. DP mode here will be that Merlin keeps both attitude and position while it can be maneuvered with velocity control in all degrees of freedom. If Merlin then loses DVL data (i.e., EKF stops running in dead reckoning), it enters auto attitude and depth with force control in the direction it does not have measurements. Force control can be described briefly as a control

mode that gives out a force that should correspond to the desired velocity under perfect conditions.

5.7.4 Transitions

The transitions between using velocity control, attitude and position control or a mixture of these with state feedback linearization can be done in several ways. The method chosen here is to let the controllers run in parallel and manipulate the measurement data and control inputs. As previously mentioned, this is supposed to make it easier to familiarize with the system. It also means that in a trial phase controllers can be tested separately.

5.7.5 State Feedback

For Merlin, it is clear that both damping and hydrostatics are nonlinear and the described meaning of state feedback is to cancel them. This is done by feedback of this dynamics. Since the measurement attitude is constantly available and the hydrostatics are only a function of this, the contribution from this part can be returned continuously. The second contribution is a bit more complicated. First, it is not certain that the current measurements will be available all the time. Velocity measurements over the bottom definitely will not be, meaning the contribution will vary. Depending on the desired mode of the controller and measurements, the force control can be active. When it is active, the feedback term related will be set to zero, regardless of whether it is measuring or not. Read about this in the section on force control. The feedback to the system would then be

$$\underbrace{\mathbf{D}((\nu - \nu_c))(\nu - \nu_c)}_{\text{Feedback depending on mode and measurements}} + \underbrace{\mathbf{g}(\eta)}_{\text{Continuous feedback}} \quad (5.21)$$

Note that the feedback is the same for both velocity, attitude and position control. This can be seen by looking at eq. 5.11 and 5.19.

5.7.6 Velocity Control

The most challenging part of the controller is velocity control. There are two reasons for this. First, as the control objective is defined in its own reference

frame, some transformation is required. Second, unlike the attitude and position controller (for the moment, can later be expanded to handle set- points, etc.), there is dynamic manual desired control input. This means that there must be much more logic around this part of the controller. It is also very important to remember that there is a reference model between the operator and the values passed to the controller. This means there will be some form of memory in these values and if one is not careful, some unwanted steps in the values going into the controller could appear. The solution is to create the manipulation of the variables before the reference model, not after, although at first glance this could seem sensible. But one must be very careful here as to the mode and measurements available. For example, the measurement of the depth is available all the time, as is the velocity in down direction. This means that the control for depth rate will be available at all times. If so, the DVL will not bottom lock the system and assume that the velocity in the north and east is zero. But it does because Merlin can both pitch and roll not mean that the body-fixed velocities u and v is equal to zero. Depending on the orientation, the depth rate definitely can make a contribution in these directions.

It requires a lot of logic, but in short, this is what happens in the raw input to the reference model:

$$u_{sd} = \begin{cases} 0 & \text{if } MODE = "Auto Attitude and Depth/Force Control" \\ u_{sd} & \text{if } MODE = "DP" \end{cases} \quad (5.22)$$

$$v_{sd} = \begin{cases} 0 & \text{if } MODE = "Auto Attitude and Depth/Force Control" \\ v_{sd} & \text{if } MODE = "DP" \end{cases} \quad (5.23)$$

5.7.7 Attitude and Position Control

Attitude and position control is such that if the control input indicates that the operator wishes to move one or more directions, the corresponding directions will be released by the controller. For movements to the north, east, up and heading change, the following variable manipulation will happen:

Algoritme 1 Discontinuity-fix for heading error

```

ERROR := PSI - PSI_DESIRE
IF ERROR <= -PI THEN
  ERROR := 2PI - ERROR;
ELSIF ERROR > PI THEN
  ERROR := -2PI + ERROR;
END_IF;

```

$$\begin{aligned}
\hat{N} &= N_d = 0 \\
\hat{E} &= E_d = 0 \\
\dot{\hat{N}}_d &= \hat{\dot{N}} \\
\dot{\hat{E}}_d &= \hat{\dot{E}}
\end{aligned} \tag{5.24}$$

$$\begin{aligned}
D_d &= D \\
\dot{D}_d &= \dot{D}
\end{aligned} \tag{5.25}$$

$$\begin{aligned}
\psi_d &= \psi \\
\dot{\psi}_d &= \dot{\psi}
\end{aligned} \tag{5.26}$$

This allows the contribution of this part of the controller to disappear. Note also that it is a local positioning system developed for the positions in the north and east. This means that the desired position during station keeping will always be $\nu_{1_d} = [0, 0, D]^T$. Note also that the corresponding integral for the position and attitude must be reset. This can cause problems if not done correctly. More about this in the section on integral effect.

Heading Merlin is for practical purposes and for design reasons restricted to have a larger angle in roll and pitch than ± 20 degrees. This means that the measurements will be continuous in the whole of the workspace. The measurement of true north (ψ) will be different, as there is no limit to how many rotations can be made (apart from the practical difficulties of rotation of the tether). So the angle measurement ψ is certainly not continuous; it has a jump from 0 rad to 2π rad. Since the control law using the error e between measured angle and desired, it is clear that the control around this discontinuity can cause problems. One way to solve this is to convert the error to an error described as a value between $\pm\pi$ where the problem is “shifted π radian”. A pseudo code of this

solution is given in Algorithm 1. This will be used in the implementation of the attitude controller for Merlin.

Lock The desired position and heading locks to their new values by either one enters a mode where there is a functionality for this or because the manual control in this direction stops. How this is done is discussed in the reference model section.

5.7.8 Force Control

Force control is a mode for control of Merlin that only applies to the two control inputs u_{s_d} and v_{s_d} in the body-seabed frame where measurements are not available all the time. Instead of having a controller that compares the desired and estimated velocity and feeds back the dynamic as part of the state feedback concept, the corresponding contribution that should be required under ideal circumstances is feed forward. By having access to current measurements and a good model, one could achieve a very similar performance as with a controller. As mentioned in the section on the feedback, the contribution for the state feedback has been set at zero when the controller is in force control mode. Force control switches on and off using the values from a reference model exactly similar to that used for the velocity control. Note that this reference model is constantly active and that the switching happens after the reference model. This is done simply to reduce and prevent possible steps when switching modes since the reference model might be active at the same moment the mode is changed. Below are the different cases. Both desired velocity and acceleration are derived from the reference model and then transformed.

$$\mathbf{a}_F^b = \begin{cases} \dot{\nu}_d & \text{if } MODE = "Auto Attitude and Depth/Force Control" \\ 0 & \text{if } MODE = "DP" \end{cases} \quad (5.27)$$

$$\nu_F = \begin{cases} \nu_d & \text{if } MODE = "Auto Attitude and Depth/Force Control" \\ 0 & \text{if } MODE = "DP" \end{cases} \quad (5.28)$$

5.7.9 Integral action

To prevent adverse effects of integral action and to put a saturation on the controller output, a simple anti-windup routine is added to the velocity, attitude

and position controller. This can be seen in Appendix D. In addition, since one is the design developed in the thesis to switch between modes and thus controller, a feature was added that releases the integrals in a smooth manner. This is to prevent sudden movement if any large compensations made by the integral effect to remove offsets occurs if they are released momentarily. This can also be seen in Appendix D.

5.7.10 Control Law

The complete control law for implementation in Merlin's control system can be written as follows:

$$\tau = \mathbf{M}(\mathbf{a}^b + \mathbf{J}^{-1}\mathbf{a}^n + \mathbf{a}_F^b) + \mathbf{D}((\nu + \nu_F - \nu_c))(\nu + \nu_F - \nu_c) + \mathbf{g}(\eta) \quad (5.29)$$

Where, depending on the mode, the terms that are zero and active will vary.

5.8 Thruster Allocation

The thruster allocation is described thoroughly in the chapter taken from Knausgård [2012] and can be seen in the modeling chapter.

6 Implementation

6.1 Estimators

As explained in Knausgård [2012], there is no support for matrix operations in the PLC from Omron. Therefore, a library was developed to make it possible to use basic functions believed necessary to implement the DP system. When it was developed, it was not known that an estimator for the system also had to be made. Although there is some connection in the degrees of freedom of Merlin, a choice was made to have two separate estimators, respectively for positions (North, East), velocities (in body-seabed frame), current (in North, East) and depth, altitude and down velocity. To understand, notations must be read in conjunction with the modeling chapter.

6.1.1 North, East Position and Sea Current. Surge and Sway Velocities

The main equations and layout used in the implementation of the EKF for the position, velocities in the body-seabed frame:

$$\begin{bmatrix} \dot{N} \\ \dot{E} \end{bmatrix} = \dot{\eta}_1 = \mathbf{R}_b^n(\hat{\psi})\nu_1 \quad (6.1)$$

$$\begin{bmatrix} \dot{N}_c \\ \dot{E}_c \end{bmatrix} = \eta_2 = \mathbf{R}_b^n(\hat{\psi})\nu_2 \quad (6.2)$$

$$\mathbf{M}\dot{\nu} + \mathbf{D}(\nu_1)\nu_1 - \mathbf{D}(\mathbf{R}_n^b(\hat{\psi})\eta_2)\mathbf{R}_n^b(\hat{\psi})\eta_2 - \mathbf{b} = \tau \quad (6.3)$$

where $\nu_1 = [u_s \ v_s]^T$, $\nu_2 = [u_c \ v_c]^T$, $\eta_1 = [N \ E]^T$, $\eta_2 = [\dot{N}_c \ \dot{E}_c]^T$, $\mathbf{b} \in \mathbb{R}^2$ is the body-seabed-fixed bias vector, and $\tau = [\tau_{X_s} \ \tau_{Y_s}]^T$ is the control input vector in the body-seabed-fixed reference frame.

$$\mathbf{R}_b^n(\hat{\psi}) = \begin{bmatrix} \cos(\hat{\psi}) & -\sin(\hat{\psi}) \\ \sin(\hat{\psi}) & \cos(\hat{\psi}) \end{bmatrix} \quad (6.4)$$

$$\mathbf{M} = \begin{bmatrix} 1.1m & 0 \\ 0 & 1.1m \end{bmatrix} \quad (6.5)$$

$$\mathbf{D} = \begin{bmatrix} X_{|u|u} & 0 \\ 0 & Y_{|v|v} \end{bmatrix} \quad (6.6)$$

$$\mathbf{M}\dot{\nu} + \mathbf{D}(v_1)\nu_1 - \mathbf{D}(\mathbf{R}_n^b(\hat{\psi})\eta_2)\mathbf{R}_n^b(\hat{\psi})\eta_2 - \mathbf{b} = \tau \quad (6.7)$$

Bias model The bias model is modelled as random walk as mentioned in Sørensen [2012]

Measurements The measurements equation is written

$$\mathbf{y} = [\nu_1 \quad \mathbf{R}_b^n(\hat{\psi})\nu_2]^T + \mathbf{v}$$

where $\mathbf{v} \in \mathbb{R}^4$ is the zero-mean Gaussian measurement noise vector. Note the use of the rotation matrix for the measurement of the sea current.

Resulting Control Plant Model

$$\begin{aligned} \dot{\eta} &= [\mathbf{R}_b^n(\hat{\psi})\nu_1 \quad 0]^T \\ \dot{\mathbf{b}} &= \mathbf{E}_b \mathbf{w}_b \\ \mathbf{M}\dot{\nu} &= -\mathbf{D}(v_1)\nu_1 + \mathbf{D}(\mathbf{R}_n^b(\hat{\psi})\eta_2)\mathbf{R}_n^b(\hat{\psi})\eta_2 + \mathbf{b} + \tau \\ \mathbf{y} &= [\nu_1 \quad \mathbf{R}_b^n(\hat{\psi})\nu_2]^T + \mathbf{v} \end{aligned} \quad (6.8)$$

Extended Kalman Filter Design

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{B}\mathbf{u} + \mathbf{E}\mathbf{w} \\ \mathbf{y} &= \mathbf{H}\mathbf{x} + \mathbf{v} \end{aligned}$$

$$\hat{\mathbf{x}}(k) = \begin{bmatrix} \hat{N} \\ \hat{u}_s \\ \hat{E} \\ \hat{v}_s \\ \hat{N}_c \\ \hat{E}_c \\ \hat{b}_{X_s} \\ \hat{b}_{Y_s} \end{bmatrix}$$

where subscript s means body-seabed-frame and c current. b is bias.

$$\mathbf{H} = \begin{bmatrix} 0 & [bottom_lock] & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & [bottom_lock] & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c(\hat{\psi}) & -s(\hat{\psi}) & 0 & 0 \\ 0 & 0 & 0 & 0 & s(\hat{\psi}) & c(\hat{\psi}) & 0 & 0 \end{bmatrix}$$

where the $[bottom_lock]$ is a value coming from the DVL logic. 1 if measurement 0 otherwise.

$$\mathbf{f}(\mathbf{x}(k)) + \mathbf{B}\mathbf{u}(k) = \begin{bmatrix} \frac{c(\hat{\psi})\hat{x}_2 - s(\hat{\psi})\hat{x}_4}{-X_{|u|u}|\hat{x}_2|\hat{x}_2 + \hat{x}_7} \\ \frac{1.1m}{s(\hat{\psi})\hat{x}_2 + c(\hat{\psi})\hat{x}_4} \\ \frac{-Y_{|v|v}|\hat{x}_4|\hat{x}_4 + \hat{x}_8}{1.1m} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} \frac{0}{\tau X_s} \\ \frac{m}{\tau Y_s} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \frac{c(\hat{\psi})\hat{x}_2 - s(\hat{\psi})\hat{x}_4}{-X_{|u||u}|\hat{x}_2|\hat{x}_2 + \hat{x}_7 + \tau X_s} \\ \frac{1.1m}{s(\hat{\psi})\hat{x}_2 + c(\hat{\psi})\hat{x}_4} \\ \frac{-Y_{|v|v}|\hat{x}_4|\hat{x}_4 + \hat{x}_8 + \tau Y_s}{1.1m} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\Phi(k) = \mathbf{I} + h \left. \frac{\partial \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k))}{\partial \mathbf{x}(k)} \right|_{\mathbf{x}(k) = \hat{\mathbf{x}}(k)}$$

\Downarrow

$$\begin{bmatrix} 1 & hc(\hat{\psi}) & 0 & -hs(\hat{\psi}) & 0 & 0 & 0 & 0 \\ 0 & 1 - \frac{2hX_{|u|u}|\hat{x}_2|}{1.1m} & 0 & 0 & \frac{2hc(\hat{\psi})X_{|u|u}|c(\hat{\psi})\hat{x}_5|}{1.1m} & \frac{2hs(\hat{\psi})X_{|u|u}|s(\hat{\psi})\hat{x}_6|}{1.1m} & \frac{h}{1.1m} & 0 \\ 0 & \frac{hs(\hat{\psi})}{1.1m} & 1 & hc(\hat{\psi}) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 - \frac{2hY_{|v|v}|\hat{x}_4|}{1.1m} & \frac{-2hs(\hat{\psi})X_{|v|v}|s(\hat{\psi})\hat{x}_5|}{1.1m} & \frac{2hc(\hat{\psi})X_{|v|v}|c(\hat{\psi})\hat{x}_6|}{1.1m} & 0 & \frac{h}{1.1m} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

6.1.2 Weight Matrices Values

The values used during sea tests:

$$\mathbf{R} = \begin{bmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Note that the measurement noise variance is set to zero for current measurements. This is because it fails to involve them during the sea trials (the current measurements were set to zero).

$$\mathbf{Q} = \begin{bmatrix} 0.001 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.000001 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.001 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.000001 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 100.0 \end{bmatrix}$$

6.1.3 Depth, Altitude and Down Velocity

The main equations and layout used in the implementation of the EKF for the depth, altitude and velocity:

$$\hat{\mathbf{x}}(k) = \begin{bmatrix} \hat{D} \\ \hat{\dot{D}} \\ \hat{A} \\ \hat{b}_{Z_s} \end{bmatrix}$$

where D is depth. A altitude and b bias.

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\mathbf{f}(\mathbf{x}(k)) + \mathbf{B}\mathbf{u}(k) = \begin{bmatrix} \hat{x}_2 \\ -Z_{|w|w}|\hat{x}_2|\hat{x}_2 + \hat{x}_4 \\ \frac{1.1m}{-\hat{x}_2} \\ 0 \end{bmatrix}$$

$$\Phi(k) = \mathbf{I} + h \left. \frac{\partial \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k))}{\partial \mathbf{x}(k)} \right|_{\mathbf{x}(k) = \hat{\mathbf{x}}(k)}$$

\Downarrow

$$\begin{bmatrix} 1 & h & 0 & 0 \\ 0 & 1 - \frac{2hZ_{|w|w}|\hat{x}_2|}{1.1m} & 0 & 0 \\ 0 & -h & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

6.1.4 Weight Matrices Values

The values used during sea tests:

$$\mathbf{R} = \begin{bmatrix} 0.01 & 0 \\ 0 & 1.5 \end{bmatrix}$$

$$\mathbf{Q} = \begin{bmatrix} 0.001 & 0 & 0 & 0 \\ 0 & 0.000001 & 0 & 0 \\ 0 & 0 & 0.001 & 0 \\ 0 & 0 & 0 & 1.5 \end{bmatrix}$$

6.2 Controller Parameters

The values used during sea trials are presented below.

6.2.1 Velocity Gains

$$\mathbf{\Lambda} = \begin{bmatrix} 0.3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.2 \end{bmatrix}$$

6.2.2 Attitude and Position Gains

$$\mathbf{\Lambda} = \begin{bmatrix} 0.43 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.43 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.45 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.85 \end{bmatrix}$$

6.2.3 Anti-windup Gains

Relatively conservative choices of values to prevent dangerous situations.

For velocity control:

$$\nu_{antiwindup} = [0.20 \quad 0.20 \quad 0.35 \quad 0 \quad 0 \quad 0.4]^T$$

For attitude and position:

$$\eta_{antiwindup} = [0.25 \quad 0.25 \quad 0.20 \quad 0 \quad 0 \quad 0.18]^T$$

As seen and described in the results are both roll and pitch decoupled. (Anti-windup set to 0)

6.3 Software Presentation

When a project is so comprehensive in terms of scale and programming code written, it is difficult to present in a simple way. Some code is placed in Appendix D, while the rest of the code is attached in the enclosed DVD/compressed folder. But since it required a license from Omron to read the program written, a recording has been made of the screen for about 10 minutes where this author browses through the key components, giving the reader an insight into the structure and structuring. This is especially vital to see how this is done in a PLC.

6.4 Data Trace for Omron PLC

Some special solution was needed to enable logging of the necessary amount of data for analysis and documentation from the PLC. In Omron's CX-programmer software, a logging program is built in that can log specific memory addresses in one connected PLC. However, there are several limitations; the PLC must be connected to a PC via USB for data logging and only a single PLC can be connected per PC. Furthermore, it can only log 16 memory addresses simultaneously. And since most of the variables of interest are floating point numbers, this number will be halved.

The solution to the problem is to make sure the values to be logged are small numbers. This means that by multiplying the values by 1000 and converting to integers (or directly convert if the number could be large), the number can be doubled. But since there are over 40 values that are logged, an additional PLC only for logging must be set up. This means, therefore, a total of 3 PLCs (i.e., old PLC control system, DP system PLC and log PLC) will be used. As stated earlier, these must be connected to their own computer running CX-Programmer. Since the variables of interest are all located on the DP-system's PLC, it sends over the values to the others (in addition to a synchronization value to synchronize the measurements later).

7 Pratical Experiments

As part of the development of the system, it was important to constantly try, in the best ways possible, to test and validate the modules of the system to eliminate unnecessary errors before execution of the sea trials. Facilities associated with the University (NTNU), specifically the Marine Technology Institute, made it possible to do some very good tests for the developed estimator. Also IKM

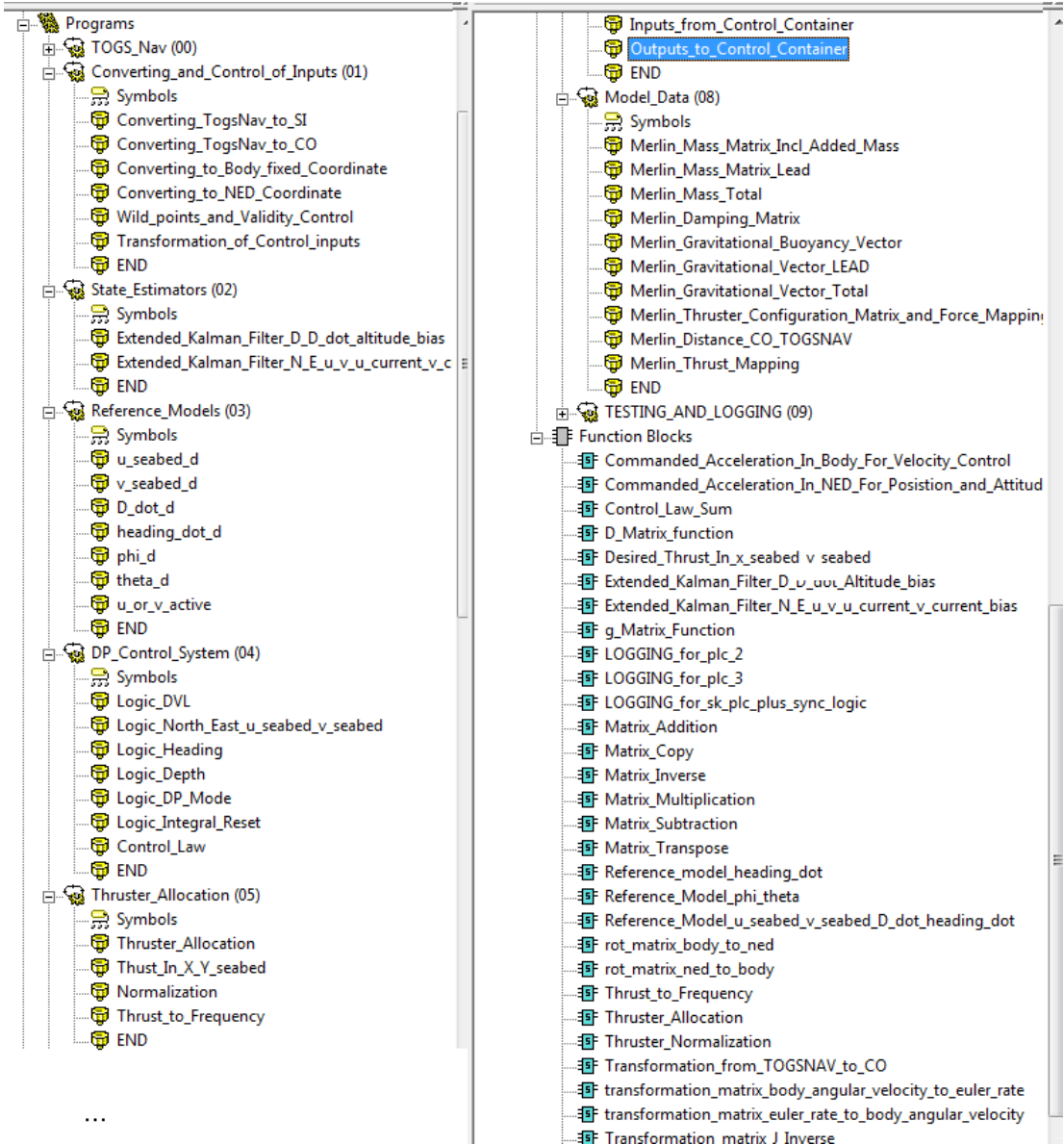


Figure 6.1: Screenshot of implemented sections and function blocks in Omrons CX-programmer

Subsea was very helpful in setting up a test bench where the main components of the system could be tested before the sea trials.

7.1 Doppler Velocity Log

Since an estimator was developed for position, current and velocity, it was important to test this, especially because the estimator requires some tuning to function satisfactorily. To do this, relevant measurement data was collected from the DVL in action and then worked on in a similar model of the estimator that will be implemented in the PLC in MATLAB (Simulink). To collect the data from the DVL, it was desirable to have control on the measurement. In other words, to have a device which can move in a fixed known velocity over the bottom. After some research to find a suitable location, NTNUs "Lilletanken" test tank at Marine Technical Institute at Tyholt was selected. Here they have a tank with a carriage that easily can be controlled in a given velocity. The carriage can be controlled via a simple control interface shown in Figure 7.1.

To attach the TOGSNAV to the carriage a stand was welded, in collaboration with a mechanic at NTNU Tyholt, that could easily assemble the sensor as well as attach it to the carriage. The TOGSNAV mounted in the welded stand can be seen in Figure 7.2 and both the stand and TOGSNAV can be seen mounted to the carriage in Figure 7.3. Furthermore, a small electric trolling motor was mounted to the carriage to imitate the sea current. This can be seen mounted in the Figure 7.4. The test setup instrumentation is very simple: A RS-232 USB-serial converter driver sends the data from the TOGSNAV into a terminal program on a PC. The actual logging of data is done by reading in the sensor string in the PLC and logging the data using a data trace program from Omron. Relevant data from the DVL logged: velocities of both the bottom and the reference layer in the water (current measurement). Note that DVL during the tank test will only move in the surge direction.

7.1.1 Result

Several beneficial results came out of the test performed. First, it was useful to have an experience with the behavior of the TOGSNAV. Moreover, it was very nice to clarify that it would not put out any measurements or any measurements that showed any useful data from the current measurement. It is believed that a greater distance to the bottom is necessary to obtain measurements that can be used. Distance during the tests was 0.7 meters. The problem is that this distance will not necessarily be much greater when Merlin is used in operation and needs DP. This will be something that needs to be examined when the sea trials are performed.

Logged raw velocity from the DVL can be seen in Figure 7.5. As one can see

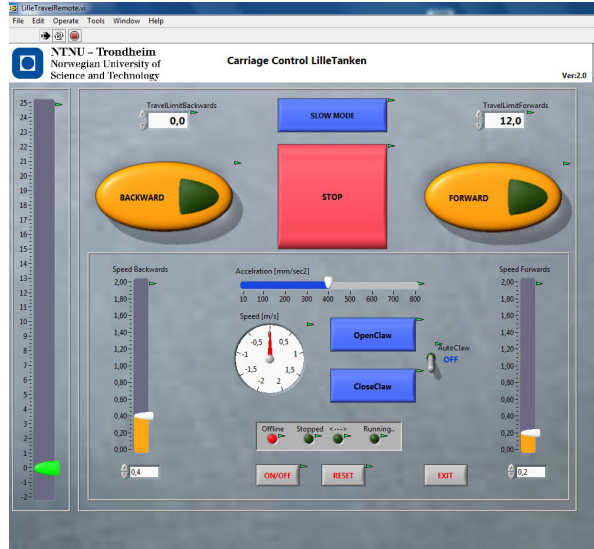


Figure 7.1: Screenshot of the carriage control program

there are several places with spikes or large steps in the velocity measurement. These mark the places where the bottom contact/lock is lost and the TOGSNAV set the values to 9.9. These data were imported to Matlab and then run through a Simulink model of the extended Kalman filter. This was tuned on the available data and the result of this is also seen in Figure 7.5. As one can see it is not tuned more on the variance of the bias to remove/eat up the error. This means that there is a lag in the estimated velocity. The reason for this is that in the completed system there will be access to the desired force from the controller. This will then be included as a feed forward in estimator and thus eliminate (most of) this lag. Otherwise, this result shows that the developed estimator seems promising. It may also be noted that the carriage that was run is much lighter and experienced a much higher acceleration and larger vibrations than the slow dynamics of Merlin will.



Figure 7.2: TOGSNAV mounted in a welded stand

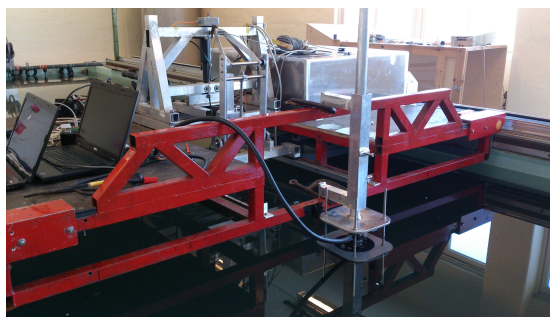


Figure 7.3: The stand with the DVL mounted on the carriage

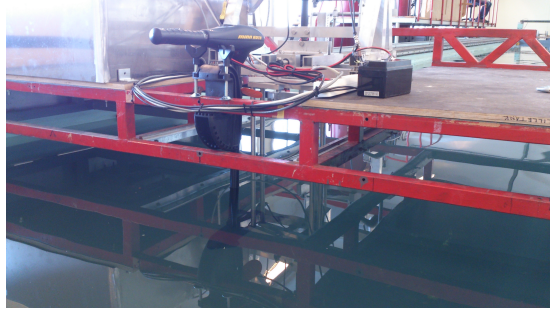


Figure 7.4: Trolling motor mounted on the carriage

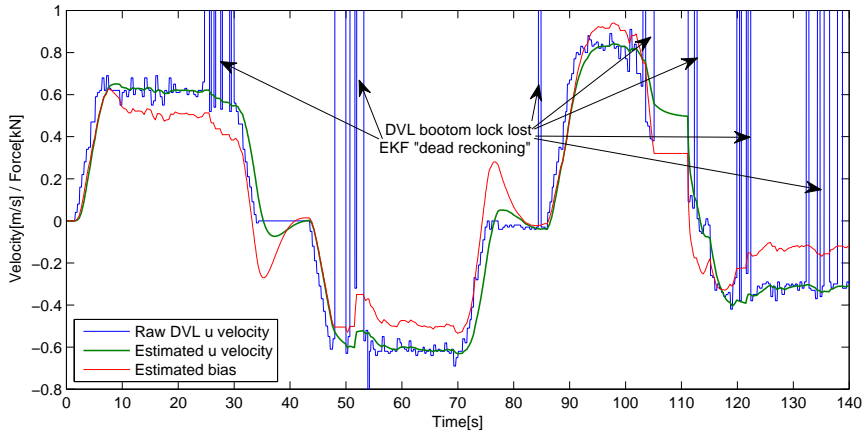


Figure 7.5: Extended Kalman filter result of of tank test

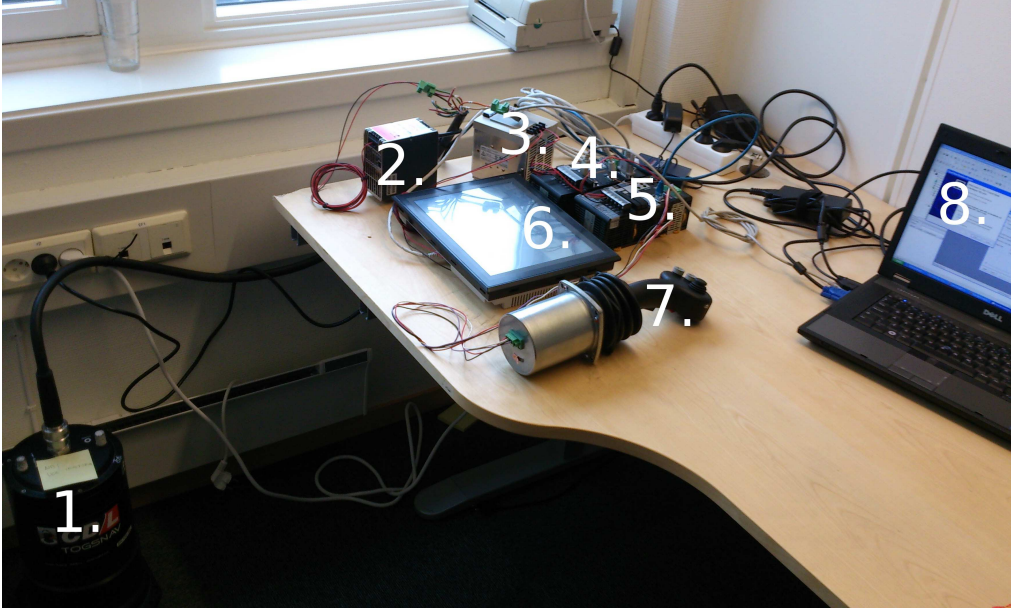


Figure 7.6: Test bench

7.2 Test Bench

Figure 7.6 shows the test bench that was set up at IKM Subsea before sea tests. This was useful for testing the reference models and the controller for attitude by rotating and tilting the TOGSNAV Table 9 shows the components.

7.3 Interface with Existing System

Although the system has been tested in a test bench that has the same components and user interface, it was important to conduct a test of the system as it will appear during the sea trials. The test bench gave no opportunity to test the interface between the control container (where the PLCs are placed) and the frequency converters (actuation interface). Neither could it test the estimator. This is something that is thought to be important to perform, as is finding a plan to see how the sea trials should be done. Determining opportunities for putting a camera on the bottom to retrieve results should also be examined. Given the limited time to carry out the sea trials, it was a useful experience for

Table 9: Test bench components

Number	Component
1.	TOGSNAV
2.	Transformer 220v ac -> 24v dc
3.	Transformer 220v ac -> 24v dc
4.	New Control System PLC
5.	Old Control System PLC
6.	HMI touchscreen
7.	Joystick
8.	PC with CX-programmer

this author and my supervisor at IKM Subsea to see which facilities there were in the test location.

Testing/validation was performed at the International Ship and Port Facility Security (ISPS) deepwater dock in Sirevag (Hå municipality, Rogaland county) 11 April. No logging or documentation of this testing was done as it was seen as a simple validation in terms of impending sea trials. Anyway, the following tests were done and all gave promising results:

- HMI: Sending parameters online to the controller, changing the mode of the control system, switching between the old and new control system, reading out the desired thrust to the different degrees of freedom.
- Control Inputs: Testing various input from the joystick through reference models.
- Thruster Allocation: Sending desired force in the different degrees of freedom to get (apparently) associated thrust on ROV with correct rotation direction of the thrusters.
- Estimators: Fly the ROV with visual reference point to look at the data from the estimators. See that the estimators give the correct (apparently) data in response to specific movements. Testing that the logic around DVL-bottom-lock works.

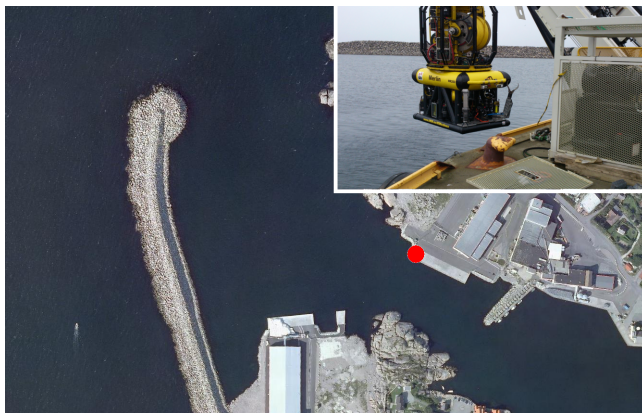


Figure 8.1: Sirevaag

8 Sea Trails

The sea trials of the complete system were performed at ISPS deepwater dock in Sirevag (Hå municipality, Rogaland county) as seen in the map section in Figure 8.1. Figure 8.2 also partly shows there is a large amount of infrastructure required to operate the system: A large diesel generator, winch (that raises/lowers the TMS), transformer container (where frequency converters are located), control container (control system and operator) are all needed. As a result, testing the system required time and effort. Small problems with the TMS meant the test was delayed a few weeks and were conducted from 27 April to 30 April. The test environment is a relatively sheltered marine environment without much depth. The tests will mainly take place at depths to 30 meters.



Figure 8.2: Infrastructure in Sirevaag



Figure 8.3: Underwatercamera with house

8.1 Test Setup

8.1.1 Data Logging and Video Recording

The methods to be used to obtain data of the test:

- Sensor and controller data logged as described under implementation.
- Use cameras on the ROV and visual reference point seen from the ROV.
- Use a camera placed on the seabed as an independent measurement.

The first two are relatively easy to perform and should not be a problem. But placing a camera on the bottom is quite a bit more challenging. The reason for it is desirable to have a camera on the bottom is that it would not exist if some independent measurements to document the performance of the system if not. The idea is to mount a simple standard underwater camera with the ability to make long recordings. In addition, it should be wide angle to catch as much as possible of the movements of Merlin. This is because it can be difficult to know whether the ROV is within view. A camera that met these requirements and that will be used is the AEE SD20 Action camera. This can be seen in Figure 8.3. In talks with IKM Subsea, it was decided that mounting the camera on a trestle mean it can easily be placed (according to them) on the seabed of ROV's manipulator. Figure 8.4 shows the camera mounted on the trestle right before it should be placed on the bottom. Leveling to ensure that the camera points as straight upwards as possible will be done visually from the ROV from multiple angles.



Figure 8.4: Seabed camera mounted on a trestle

8.1.2 Online Tuning Interface

As mentioned in the chapter with practical experiments, an HMI was developed to easily change the parameters of the controller on the fly. Since the pole placement method (see controller chapter) reduces the number of parameters, it is not as confusing as it would be if this method was not used. In fact, the number of parameters was reduced from 48 to 24. These can easily be set from the Omron touch interface in Figure 8.5.

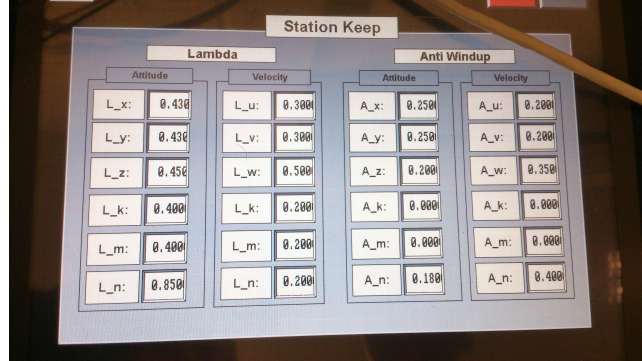


Figure 8.5: Picture of the HMI for online parameter edit

8.1.3 Experiments to be Performed

With limited time and the consideration that there are many parameters to be tuned, it is desirable to carry out sea trials in steps. This will also ensure some results in the case where much of the implemented system does not work. In other words, one will have data from the heading controls and depth controller although the station keeping system was proven not to work. It will also be difficult to identify possible sources of error if too large a portion of the system is being tested at once. Another prominent point is the ability to compare the existing controllers to Merlin with those developed in this thesis. This applies to heading and depth controllers. The tests should be performed in two distinct parts. One part deals with the consistency of the data coming out of the estimators, especially for the local position in the north and east. This provides the basis for the reference to the station keeping system, in that if the data gives good results it will be used for logging and hence say something about the performance of the station keeping system. The second part of the testing are the attempts to use the controller. Below is a list of the tests to be performed:

- Stationary estimator test for north/east: Using the manipulators to grip and hold the ROV to an installation on the seabed and thus maintain a constant position relative to the seabed. Also keep a visual reference point as an additional confirmation. Logging estimated position in north and east.
- Movement estimator test north/east and surge/sway velocities: Using the manipulators to grip and hold the ROV to keep an initial position. Fly

a distance of 20-50 meters with ROV and then return to the initial position. To confirm that the same position is reached by checking visually in combination with heading measurement. Logging the estimated position and velocities.

- Movement estimator test for depth: Conduct a descent with ROV with the old system to test the depth estimator. Logging raw data and the estimated depth.
- Old heading controller test: Keeping a constant heading with the old control system. Logging the desired heading and the real.
- New heading controller test: Keeping a constant heading with the new control system. Logging the desired heading and the real. Turn off the impact in all other degrees of freedom and allow ROV to float freely during this test period.
- New depth controller test: Keeping a constant depth with the new control system. Logging the desired depth and the estimated. Turn off the impact in all other degrees of freedom and allow ROV to float freely during this test period.
- Depth rate controller test: Keeping a constant depth rate with the new control system. Logging the estimated depth, desired and estimated rate. Turn off the impact in all other degrees of freedom and allow ROV to float freely during this test period.
- Roll/pitch controller test: Keeping a constant roll/pitch angle. Logging desired and measured roll/pitch angle. Turn off the impact in all other degrees of freedom and allow ROV to float freely during this test period.
- Force control mode test: Turn on the heading and depth controller and then maneuver the ROV with the developed force control system. Make simple maneuvers in the surge, sway, depth ratio and heading rate. Logging of relevant data and video.
- Station keeping test I: Perform station keeping with near seabed. Logging of relevant data
- Station keeping test II: Perform station keeping in the wave zone. Logging of relevant data.

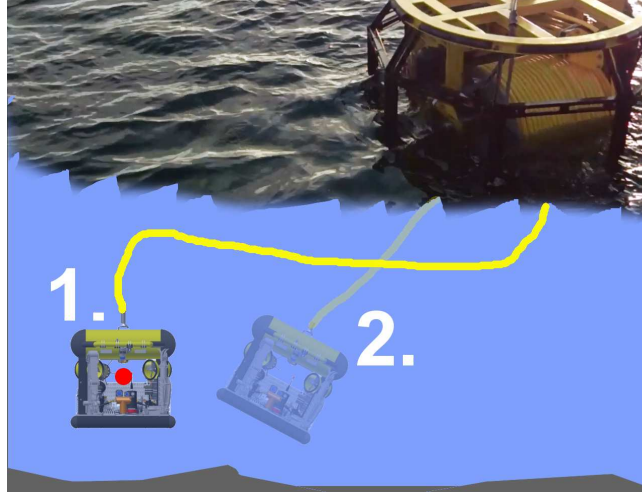


Figure 8.6: Merlin w/o forced disturbance from the TMS

- Station keeping test III: Perform station keeping in north east with varying depth. Carry out an ascent from the ocean floor (20-30m) to the surface (wave zone) and back down (moving along the Down axis of NED). Logging of relevant data.
- Station keeping test IV: Conducting station keeping in the free water masses and then applying a large external disturbance forcing the ROV from the operating point (reference point for station keeping system from scratch). This disturbance is performed using the TMS to reel the cable in more than the distance from the ROV to TMS. This can be seen in Figure 23.24. Mark 1. shows the ROV when it conducts stationkeeping with sufficient slack in the cable. The red dot marks the desired position of CO to the ROV. Mark 2. shows that the ROV has been dragged out from this point. The test is to see whether the ROV is able to get out of the working point and if it will get smoothly back when the cable once again is made slack enough.

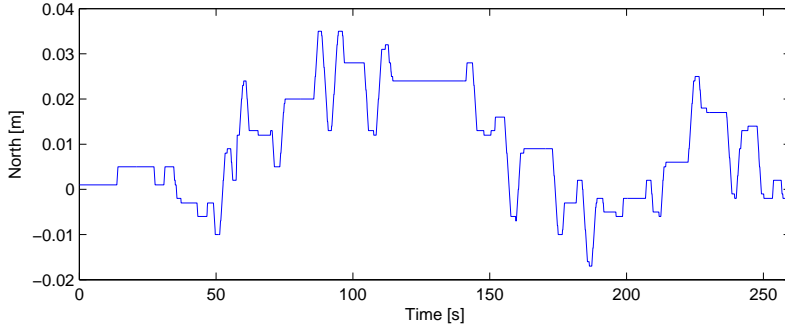
9 Results

Early in the experiments it became clear that controlling roll and pitch was difficult and despite some tuning, these remained unstable and where thus decoupled from the system. This means that all the results here will be based on a controller for 4-DOF. It was also difficult to use the velocity controller for maneuvering. So all handling done during the test was done with force control. The current measurements are as previously mentioned excluded since they do not give out any sensible data. Further tuning seems meaningless since the distance required to the bottom will be so great that it will not be usable for the intended use. In addition, it was decided to drop the controller output input to the estimator since it showed better behavior without. This is exactly the same as was done during the test in the tank at Tyholt. All of this was a necessary choice to finish quickly due to lack of time for the tests. Presented below are the results of the tests described in the test setup chapter. All results should be self-explanatory. Comments concerning the various results are given in the discussion section. Note, if no information is given in the plot of the desired position then this is always the origin.

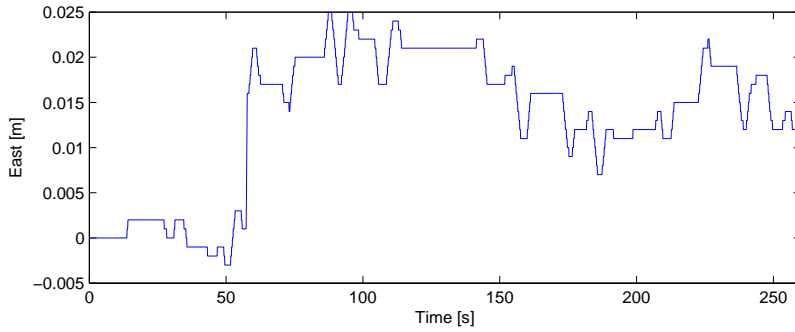
9.1 Position and Velocity Estimator Test

To determine if the estimator and thus the estimated position is precise enough that it can be used to gather results, the first test of this was conducted.

9.1.1 Drift Test with Fixed Position



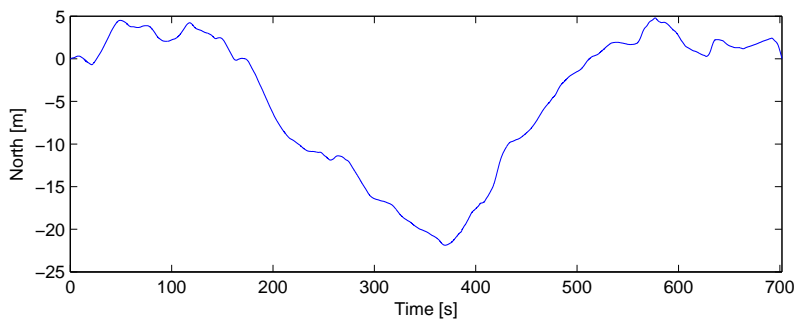
(a) North position for estimator drift test with fixed position



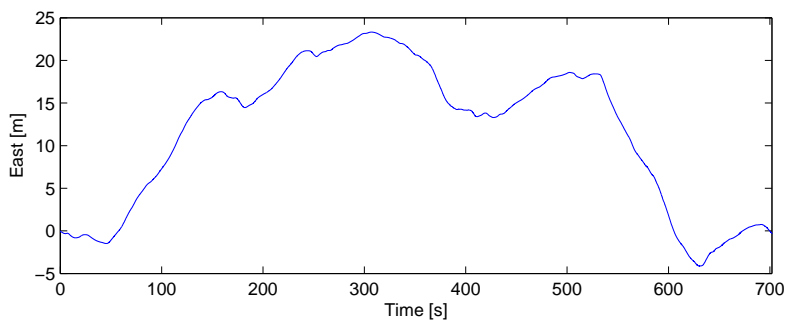
(b) East position for estimator drift test with fixed position

Figure 9.1: Drift test with fixed position

9.1.2 Drift Test with Movement and Fixed Start/Stop Position



(a) North position for estimator drift test with fixed start/stop position



(b) East position for estimator drift test with fixed start/stop position

Figure 9.2: Drift test with movement and fixed start/stop position

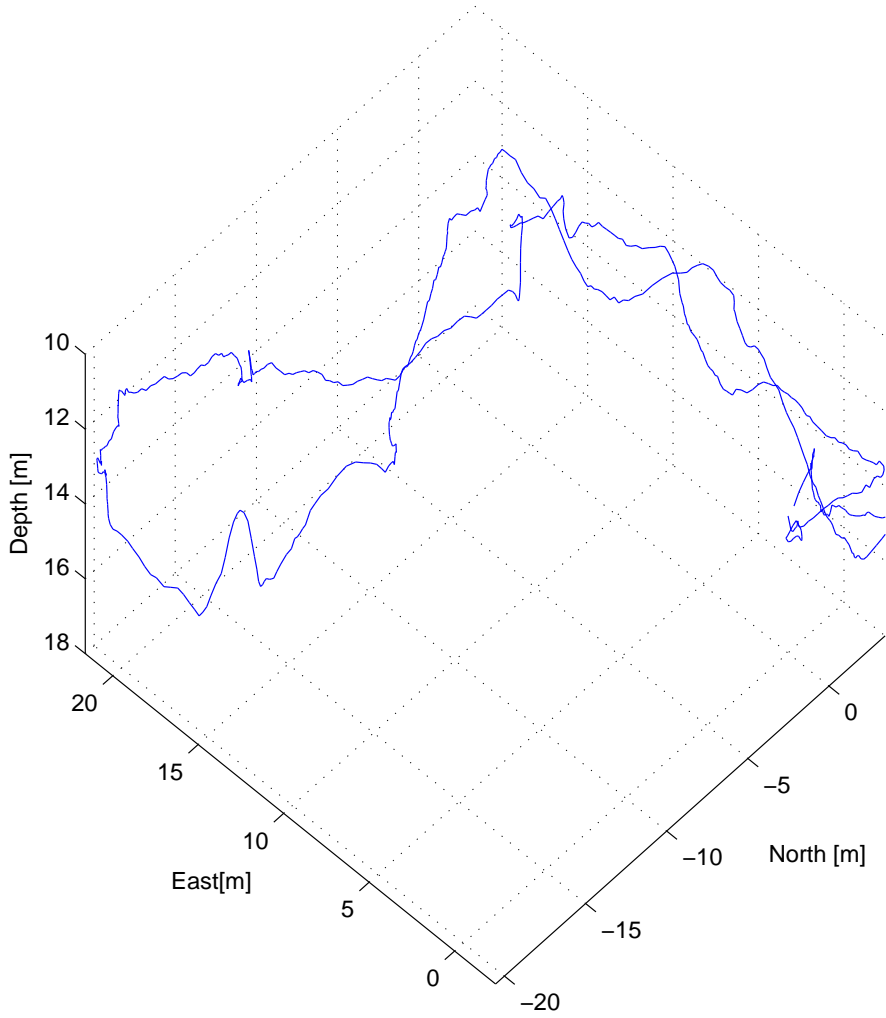
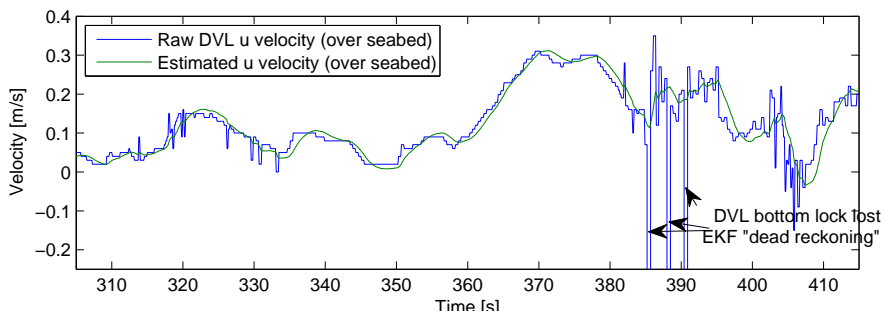
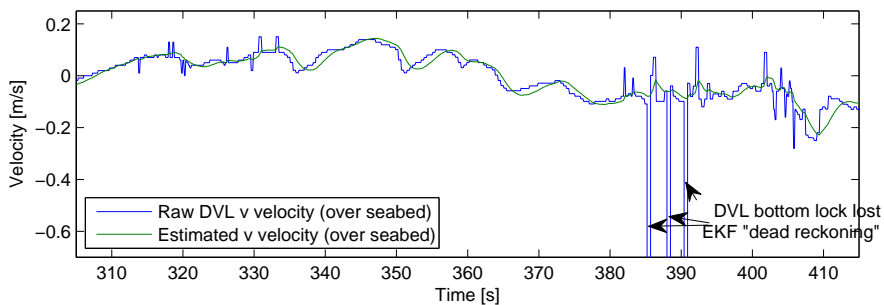


Figure 9.3: 3D plot of drift test tour



(a) Raw and estimated velocity in u body-seabed direction for drift test tour



(b) Raw and estimated velocity in v body-seabed direction for drift test tour

Figure 9.4: Drift test with movement and fixed start/stop position for velocity

9.2 Depth and Down Velocity Estimator

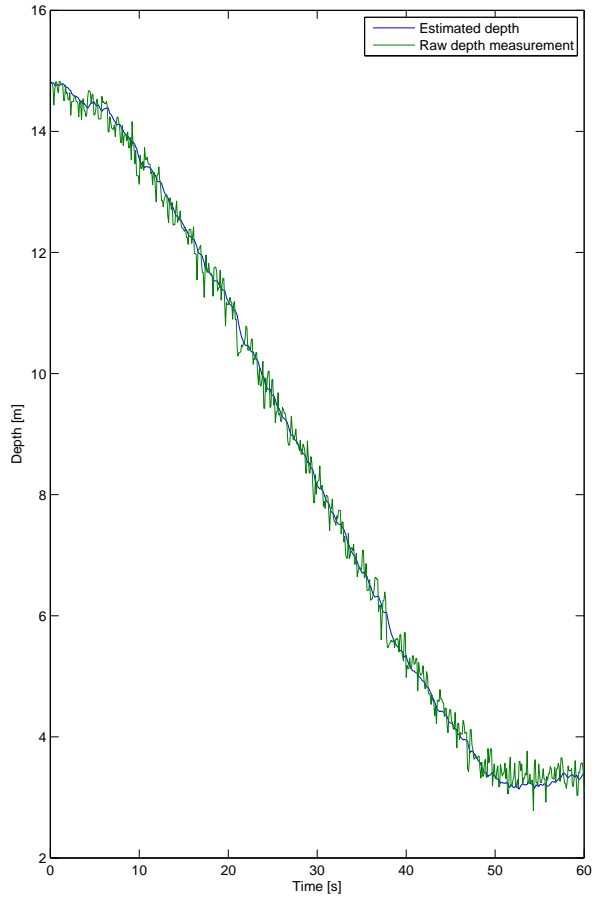


Figure 9.5: Depth and Down velocity estimator test

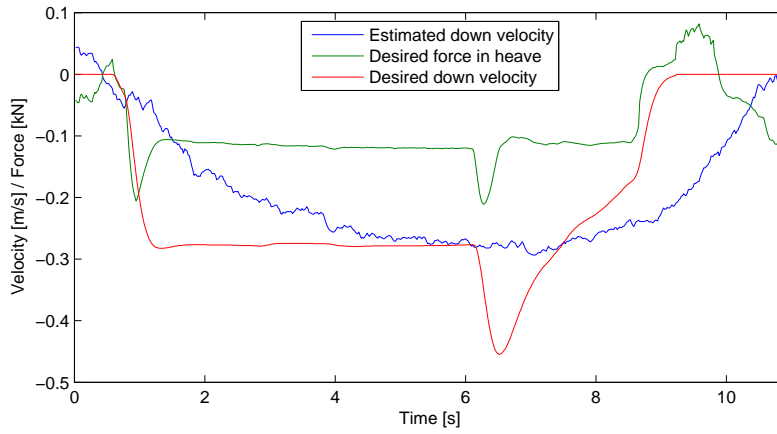


Figure 9.6: Down velocity controller and estimator test

9.3 Heading Controller

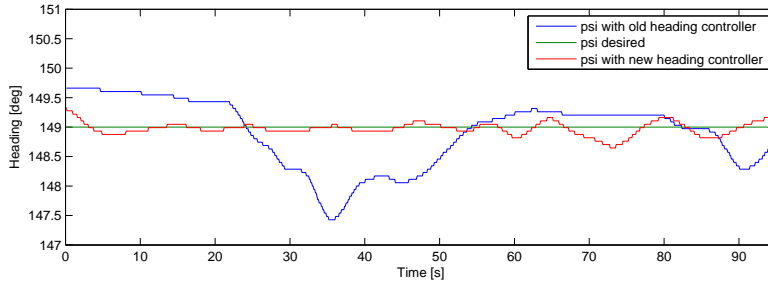


Figure 9.7: Old vs. new heading controller test

9.3.1 In Wave Zone

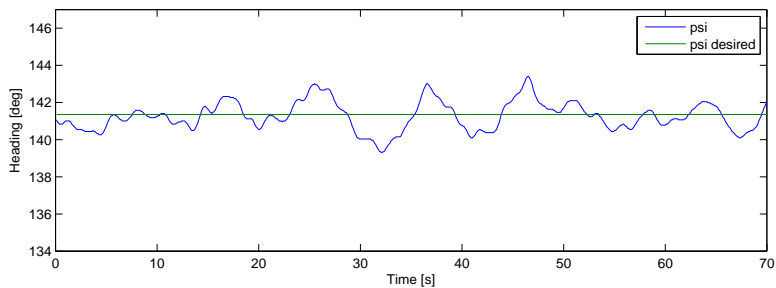
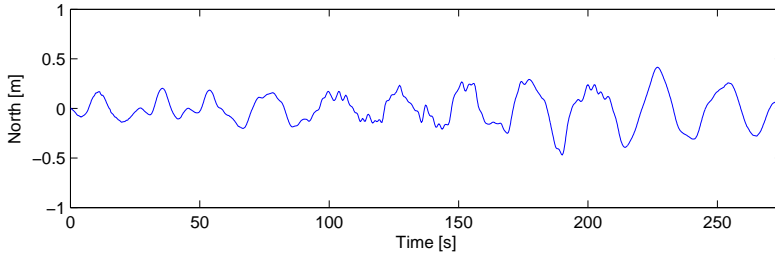


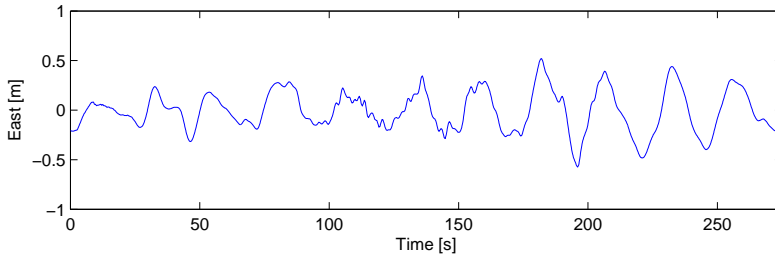
Figure 9.8: Heading controller test in wave zone

9.4 Station Keeping

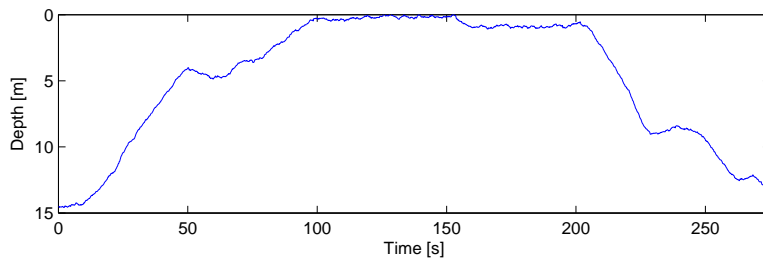
9.4.1 North and East with Varying Depth



(a) North position for station keeping test with varying depth



(b) East position for station keeping test with varying depth



(c) Depth for station keeping test with varying depth

Figure 9.9: North and East with Varying Depth

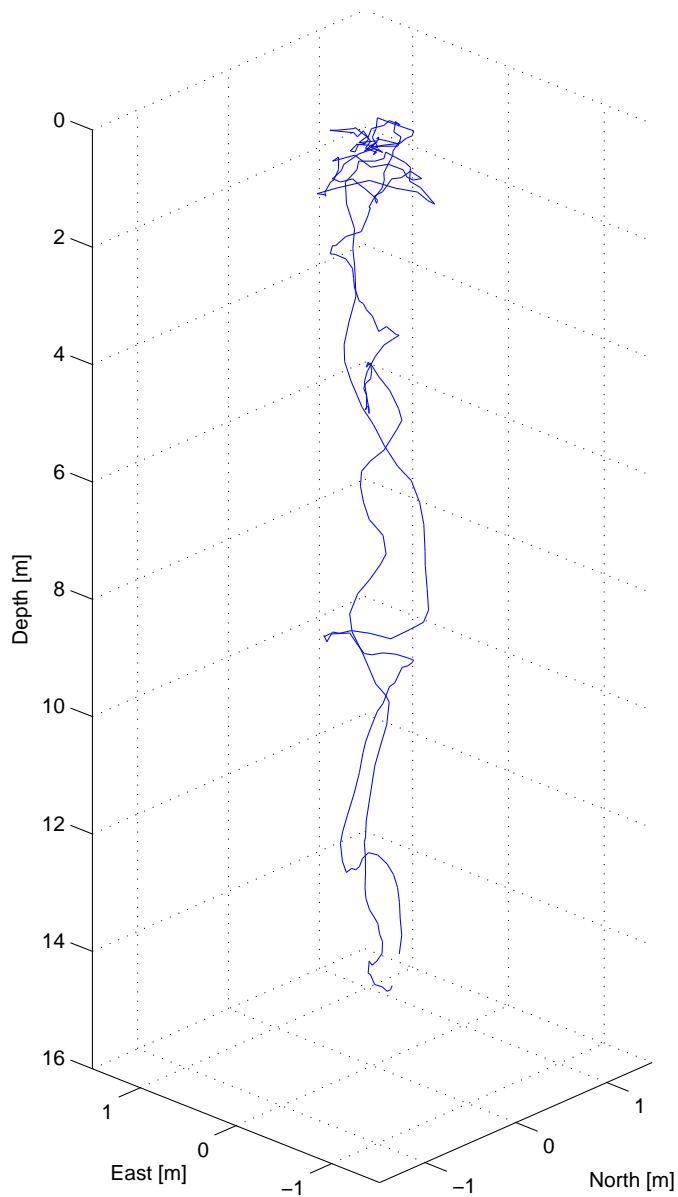
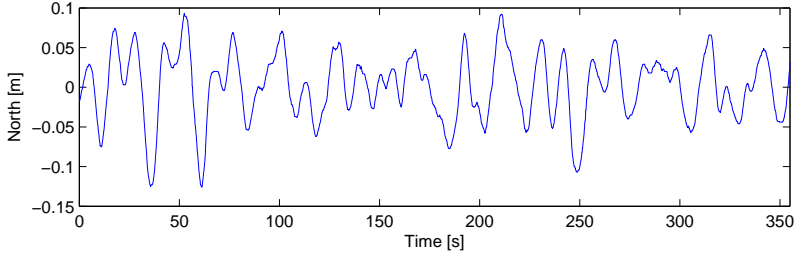
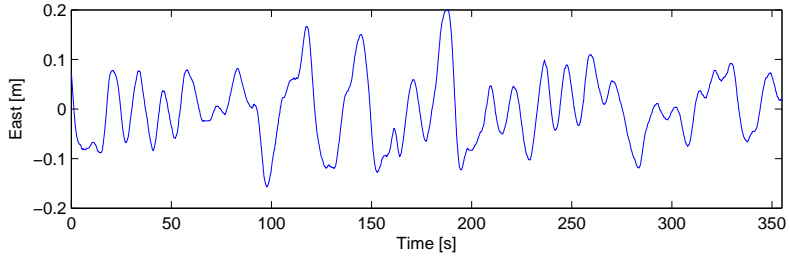


Figure 9.10: 3D plot of station keeping with varying depth

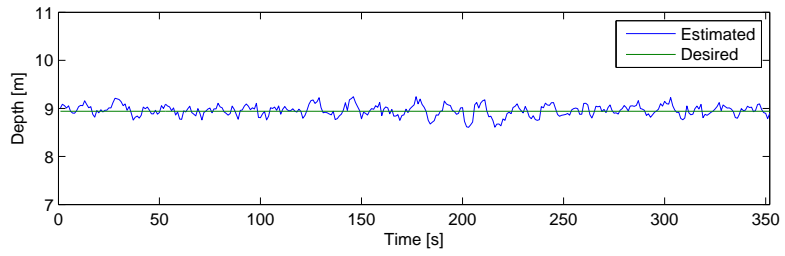
9.4.2 North,East and Down Close to Seabed



(a) North position for station keeping test



(b) East position for station keeping test



(c) Depth for station keeping test

Figure 9.11: North,East and Down Close to Seabed

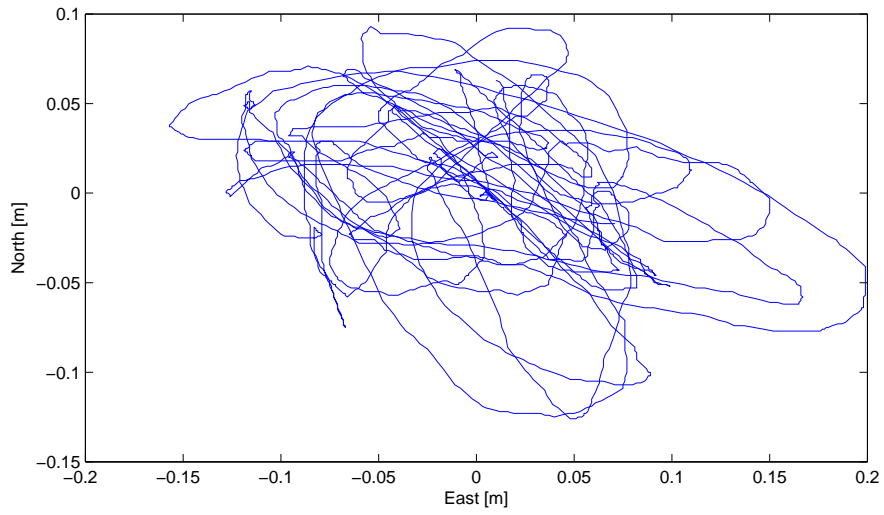


Figure 9.12: North-East plot of station keeping

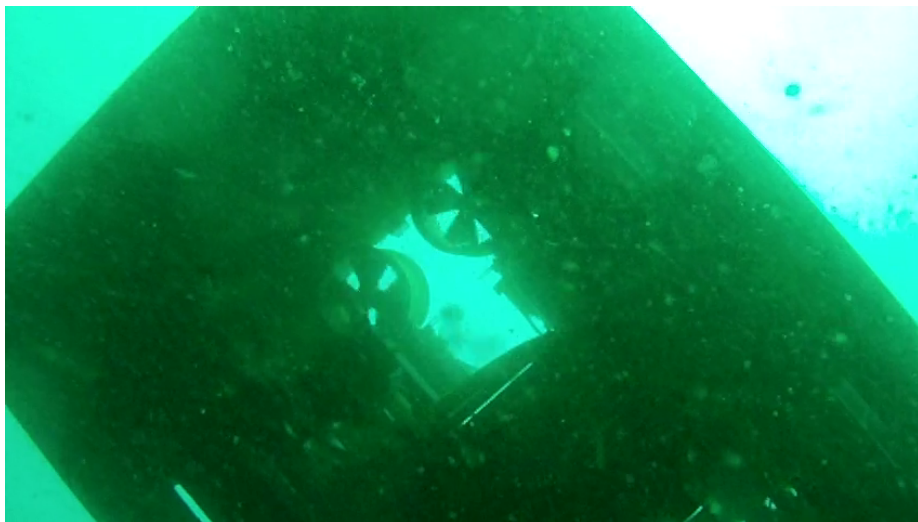
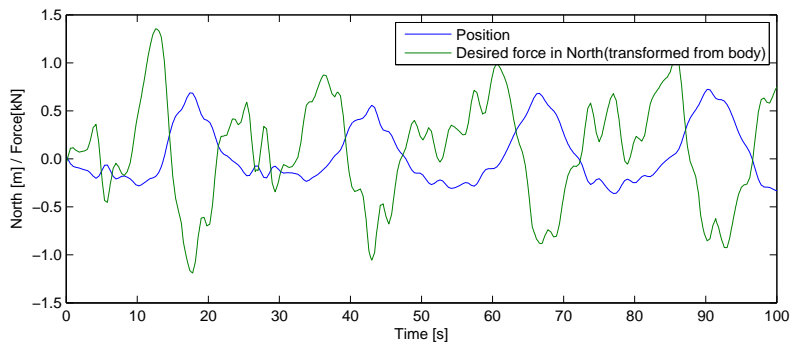
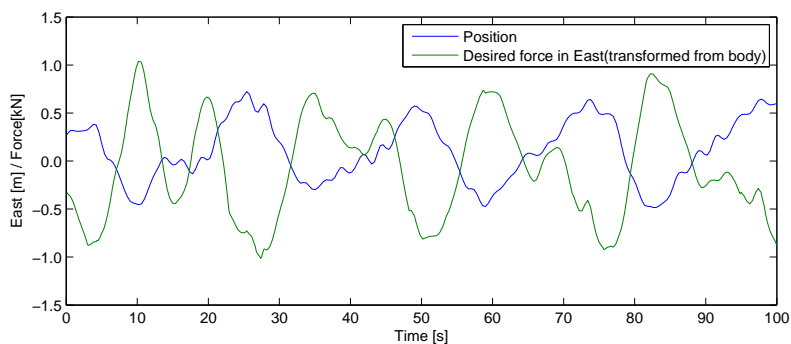


Figure 9.13: Picture of Merlin during station keeping seen from seabed camera

9.4.3 North, East and Down Close to Surface and in Wave Zone



(a) North position and desired thrust for station keeping test in the wave zone



(b) East position and desired thrust for station keeping test in the wave zone

Figure 9.14: North, East and Down Close to Surface and in Wave Zone

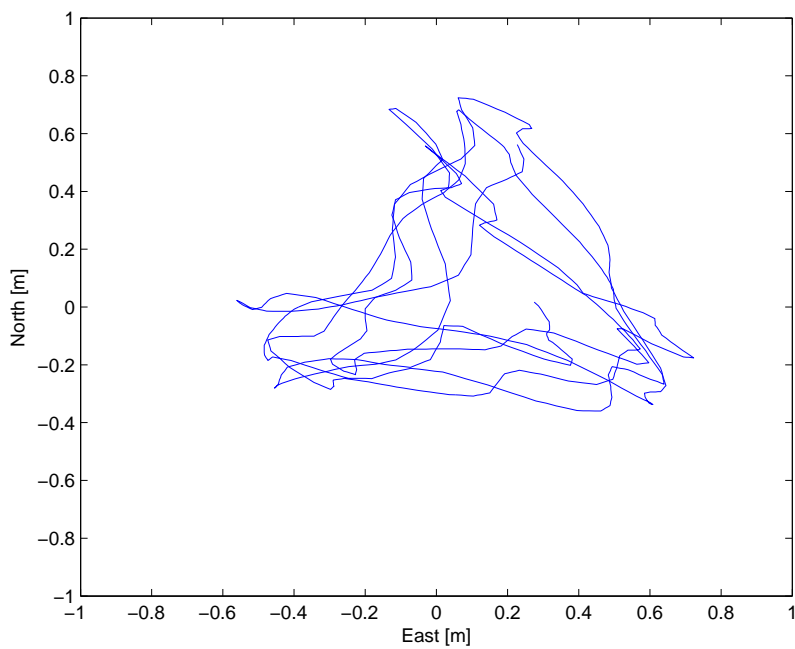
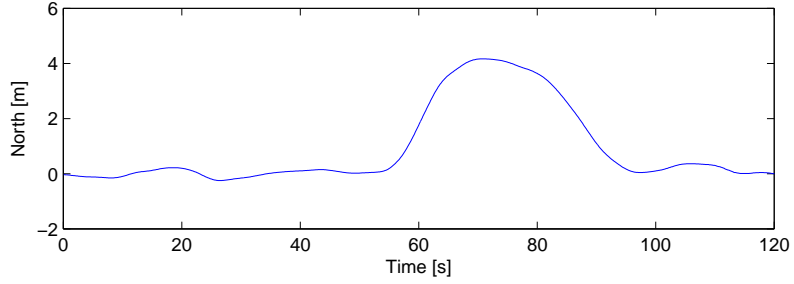


Figure 9.15: North-East plot of station keeping in the wave zone

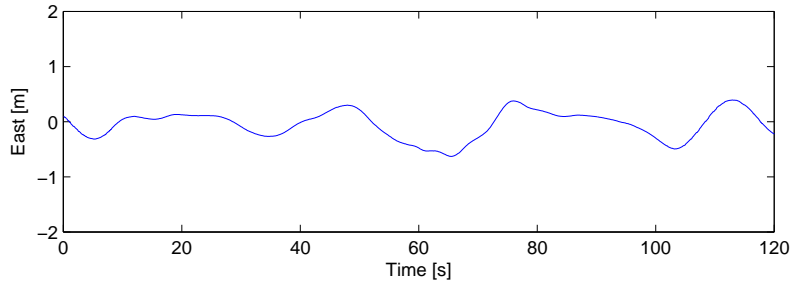


Figure 9.16: Merlin during station keeping in the wave zone

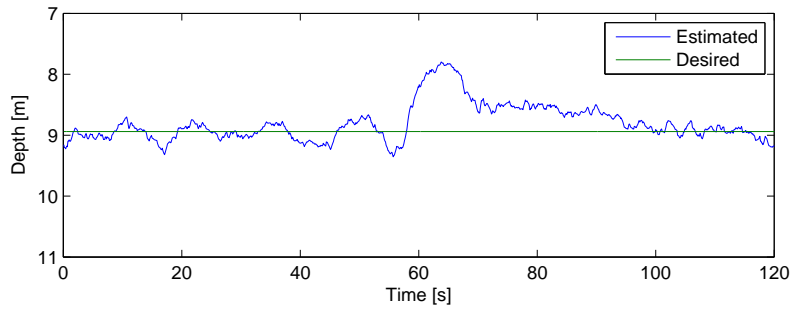
9.4.4 North,East and Down with Large Forced Disturbance



(a) North position for station keeping test with forced disturbance



(b) East position for station keeping test with forced disturbance



(c) Depth for station keeping test with forced disturbance

Figure 9.17: North,East and Down with Large Forced Disturbance

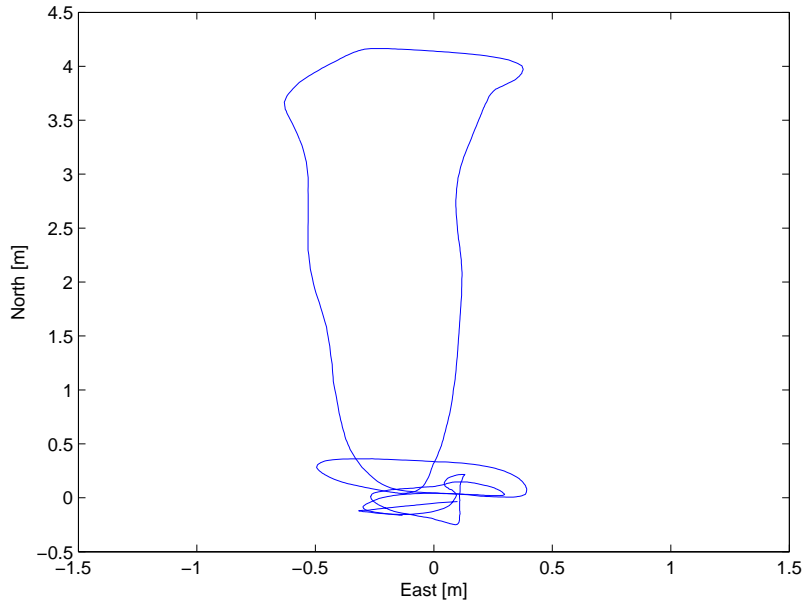


Figure 9.18: North-East plot for station keeping test with forced disturbance

9.4.5 Roll and Pitch

Roll and pitch measurements made during station keeping test near the bottom.

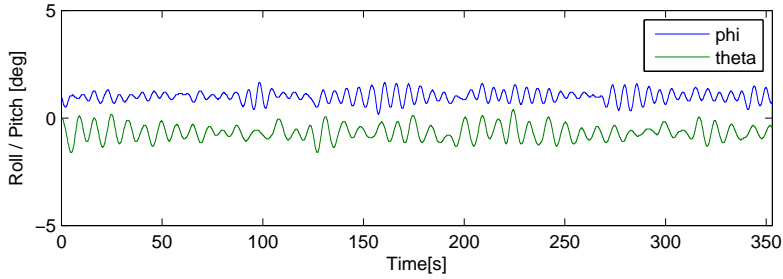


Figure 9.19: Roll and pitch during normal station keeping

9.5 Force Control

No data was logged from testing where the force control was used. However, as evidence of the behavior, here are some comments from Project Manager Operation, Svein Magne Kleven at IKM Subsea, who tested the system using force control during the sea trials:

“The behavior of the ROV with the new developing control system makes ROV twice as easy to maneuver”. The use of what the author calls a reference model and the use of the thrusters configuration to distribute the force to the different thrusters: “This makes the system more intuitive to use, and how the system switches between the station keeping and manoeuvring is also very good. When releasing the stick, Merlin holds its current position. Problems related to unwanted roll and pitch motions by acceleration in surge and sway are also improved. All in all a brilliant system that is highly desired to be included as a permanent part of Merlin’s control system”.

An attached video clip from Merlin’s camera showing the behavior when Merlin operates in force control mode near the bottom.

10 Discussion

The focus of this thesis has been to create a working DP system that could be used in sea trials. It is important to keep this in mind when considering that this thesis is so large, each component cannot be sufficiently described here. The reader should aim to see the big picture of the thesis and the results it gave. By keeping this focus in both the project paper and, especially, in this thesis, the necessary momentum has been maintained. Challenges such as not receiving promised equipment has, in many ways, meant that the author had more control over the work and reduced the number of dependencies. The plan to test parts of the system as the system was developed proved very successful. This applies particularly to the estimator that was tested and tuned before the sea trials.

Conducting the tests with the roll and pitch decoupled did not pose significant problems. Merlin shows a naturally stable behavior as seen in the logged data from the roll and pitch during station keeping and will not be taken into further reflections.

10.1 Current Measurement

The trouble getting any useful measurements of the current from the DVL resulted in this measurement being disregarded during the sea trials. This means that the measurements were manipulated to zero for the input to the estimator. In the test of the DVL at Tyholt, there was the same problem as appeared under the sea trials. Not many attempts were made to change the parameters for DVL, as time was short. The reason why it supplies rare measurements is assumed to be related to the thinness of the reference layer. This can cause the error to be very large while looking random. A larger reference layer was not selected simply because the distance Merlin must go over the bottom is so large that it is meaningless. According IKM Subsea, the distance to the bottom must be under 1 meter when station keeping is used. In other words, the other solutions considered if current measurement should be part of the system. One possible solution that the author suggested is to mount a DVL that targets upward. This will solve the problems of too little reference layer, but is a significant investment.

10.2 Estimator

The development and implementation of the estimator, and the experiments that were performed on it means the behavior was secured and the number of errors was reduced before the sea trials. The results of this single experiment and other results that have been presented show the estimator has excellent behavior. This is largely due to the quality of the measurements from the sensor, but the test also shows that if it falls out for some time and has some spike, it is important not to pass on to the controller. The major focus on improving the estimator as much as possible is motivated by the idea that no matter how much effort is placed in the development of a controller, the result cannot be better than the available measurement data. Small problems with the current measurements were that they proved useless and that using control input into the estimator seemed to have negative effects. Thus, some quick decisions had to be taken on the fly during the sea trials. The current measurements and control input into the estimator was set to zero and the variance of the bias was set up to compensate for this. The fact that the bias model is implemented body-fixed to originally compensate for small model errors now used to compensate for a mixture of body and NED-fixed forces can clearly cause problems. It should be changed in a further development of the system, depending on whether one finds a way to measure water current or not.

The results of the drift test of the estimator is so good that it is further assumed that the actual position is equal to the estimated. This is due to the position of Merlin being completely locked in the relevant logging period. The test of running Merlin through a tour with fixed start and stop also showed good results. This test is not relevant to the thesis, but can clearly be useful in developing new features in a future system.

10.3 Depth and Heading

The depth and heading controller perform well in all the tests without any problems. Remember that the system is connected but for analysis it is assumed it is possible to see the different degrees of freedom and the corresponding controller separately so long as the objective is to keep the ROV relatively calm. For pure test of heading controller, the other parameters for attitude and position were ignored so the ROV could flow freely.

IKM Subsea's heading controller as compared to the developed in this thesis clearly shows that the new one has a better behavior. In fact, it remains within plus minus 0.5 degrees over a period of 90 seconds, where the old shows results

for the same test of plus minus 1.5 degree. There is clearly an improvement. We also see that the header is kept well in the other tests, including in the swell.

The results of the depth controller can be seen in the results of station keeping. This also showed a good behavior, which is expected as this is the easiest to control. Plus/minus 0.15m for station keeping is considered good without very active tuning.

10.4 Station Keeping

The results of the most important tests related to the behavior of the DP system in the station keeping mode in different environments show very encouraging response. Despite the fact that the current in the area were performed cannot be quantified one can see that there is a significant current on a video clip taken from the seabed camera during station keeping that can be seen in the attached DVD / folder. The most relevant example is the close to seabed station keeping test where Merlin's center point (here:CO) is being held within the volume of a sphere of radius 0.2 meters for several minutes. With the fact that the current is not measured and that not much tuning of the system is done the result is considered good. In the experiment with changing depth and fixed position in north and east there is some larger deviation, but this can be caused by the significant swell in this zone. In the station keeping test, near the surface (wave zone), Merlin is driven slightly out of position but it continues to function well, taking swells into account. The attached video clip of Merlin close to the surface shows some of the challenges with the swell. Note also that the desired effect of Merlin as shown in Figure 9.14 is incredibly smooth even though any wave filter is not developed. The last test with a large forced disturbance shows Merlin in incredibly good behavior when going calmly back to its position. This despite a difference of several meters from the desired set point. This shows a robust behavior that can be attributed to the use of anti-wind up primarily.

10.5 Force Control

The force control part of the controller was developed as a supplement to the velocity control and was intended to be a way to ensure functionality to the system when necessary readings and bottom velocity was lost. It showed a good behavior as mentioned and described by the experienced pilot at IKM. In addition, how it worked was tested in interaction with station keeping and the result was so positive that consideration should be taken as to whether it can replace the velocity control completely. Especially if the current measurement

is available and the impact of this can be compensated for. How forces are distributed using the thruster allocation and how the reference model helps to facilitate the feed forward of acceleration also contribute to the response. That Merlin behaves with such stability with respect to the described problems with unwanted roll and pitch motions is very positive. An important point is that if IKM wish to use this improvement without making major changes it can easily be done by picking out the modules for the reference model, transformations and thruster allocation. This will be simple to add and may be a good way to gradually introduce the new control system.

11 Conclusion

The main objective of this thesis has been to develop and implement a complete dynamic positioning system for Merlin WR200 ROV and test and validate the system. The developed mathematical 6-DOF model of Merlin from Knausgård [2012] have been developed further and used to the development of the estimator and controller that enables dynamic positioning. A comprehensive system built in modules and implemented in a PLC with extensive use of the developed software framework in Knausgård [2012], which includes, among other things, the possibilities of using matrix operations, has been tested and validated in segments to create a complete and well-functioning system. Sea trials of the system confirm that the system works very well and it shows a robustness during tests that put the system on real trials. Both the forced disturbances and the operation in swells the system handled satisfactorily.

The current measurement using the Doppler velocity log is something that does not work the place this is now mounted. This because it requires a larger distance to the bottom than expected to work. It is difficult to say how much this meant for the system when the trials were conducted, but it is concluded that for operation in environments with heavy sea current, it will be negative and reduce the behavior of the system. During testing, it was soon clear that using the model-based controller for controlling roll and pitch was difficult. Due to limited time for sea tests, a huge effort was not made to tune the model and controller. The author would still claim that another controller design should be considered since it requires very accurate models and that the ROV easily can start a pedal movement. Because of this, all the results from sea tests are based on the controller for 4-DOF. Merlin is naturally stable so it was not of great importance here but the behavior of the ROV in different attitudes involving these degrees of freedom is, therefore, not tested.

The mode called force control that is used when there are no velocity measurements over the bottom gave very encouraging results. The reference model and the use of thruster configuration and a designed control input reference frame that are not being used in Merlin's current system gave a much simpler user control interaction and made maneuvering considerably easier. Undesirable motions in roll and pitch due to actuation in surge and sway are also removed. This is due to thruster configuration being added. How the controller and the logic surrounding the transition from modes worked is also promising.

As indicated in Knausgård [2012] and concluded in this thesis, the model for thruster characteristics are not good enough. More and better tests must be performed. This error means that the concept of state feedback linearization

quickly gets sizable errors due to the model dependence that is the underlying concept for this controller. It is believed that this is why the velocity controller worked poorly. This means that the force the system assumed was needed to keep the desired velocity is incorrect, and thus all the terms in the controller will struggle. Furthermore, it is certain that when this incorrect force is fed into the estimator it will then propagate the error. This connection was removed in all the trials. One can see that the estimates of the velocities have a lag in the tests performed, so it should certainly be included when the model has become good enough.

The estimators that are designed for velocity, position and bias has showed an incredibly good behavior in all tests. It can be concluded that this is the key foundation and essential to achieve the results that have been made.

12 Further Work

Although the system as it is presented here works well, several aspects in the further work need to be addressed in order to put it into operation. Some of the aspects and challenges that should be seen more closely are outlined below.

12.1 User Interface

The developed user interface used in the tests proved to work fine for the purpose it was exposed to here but must clearly be further developed in terms of features and layout. IKM Subsea can best describe what they want. Conversations with ROV pilots who use the systems daily will be useful here.

12.2 Estimator and Controller

Since it proved difficult to measure the water current with today's location of the DVL and the distance to the bottom must be significant to make usable measurements, a consideration can be made that this measurement must be excluded in further development. Mounting a DVL to point upwards is an alternative but this is believed to be too large an additional cost to be particularly viable. This author's idea is that the focus should be on a good mathematical model of Merlin and excellent velocity measurement and position estimate and that by a clever use of a bias estimate and the commanded thrust a good foundation can be laid for a feed forward of this bias into the controller. It will mainly contain slowly varying errors (ocean current) and could thus cause the system to behave in almost the same manner as it would with the intended developed estimator. The DP system as it stands now is only a local positioning system but it is possible that in the future it will be desirable to include other measurements for the system that can do that a global reference system will be available. It should be possible to extend the estimator by taking the existing estimator as a starting point.

12.3 Tests

Before starting the task of designing the system, this author had conversations with IKM Subsea about the importance of getting access to better data of thrust characteristic of the thrusters. This is something that the results of the sea tests confirm and is essential if further development of the system should be based

on the controller in this thesis where the accuracy of the model parameters is essential.

Conducting additional sea trials of the system as it currently stands is something that should be considered. Limited time during the trials means that debugging and tuning could have been done better.

This thesis has been faithful to performing gradual testing of new modules and this is something supremely recommended for further development of the system. Verify all new components as much as possible.

12.4 New Features

Based on conversations with the ROV pilots who participated in the testing of the system, it was clear that the system as it stands now is of interest and contributes to an easier and more efficient performance. Studying the quality of the position estimates determined that there is hope that maps, waypoints, and even more advanced features such as an opportunity to repeat the pattern flown by the ROV, navigate to waypoints, following/tracking a pipeline or a ship may be included in the future

References



- Olivier Cadet. Introduction to kalman filter and its use in dynamic positioning systems. Houston, September 2003. Dynamic Positioning Conference.
- Cheng Chin and Michael Lau. Modeling and testing of hydrodynamic damping model for a complex-shaped remotely-operated vehicle for control. Technical report, School of Marine Science and Technology, University of Newcastle, 2012.
- Thor I. Fossen. *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley and Sons Ltd, London, 2011.
- Thor I. Fossen and Jann P. Strand. Passive nonlinear observer design for ships using lyapunov methods: Experimental results with a supply vessel. *Automatica*, 35, 1999.
- Jinhyun Kim and Wan Kyun Chung. Accurate and practical thruster modeling for underwater vehicles. Technical report, Robotics and Bio-Mechatronics Lab. Pohang University of Science and Technology, 2006.
- Lasse M. Knausgård. Development of dp system for merlin wr200 roV. Technical report, Norwegian University of Science and Technology, 2012.
- Timothy Prestero. Verification of a six-degree of freedom simulation model for the remus autonomous underwater vehicle. Master’s thesis, Massachusetts Institute of Technology, 2001.
- Fan Shi-bo, Lian Lian, and Ren Ping. Research on hydrodynamics model test for deepsea open-framed remotely operated vehicle. Technical report, Shanghai Jiao Tong University, Shanghai, 2012.
- Asgeir J. Sørensen. *Marine Control Systems: Propulsion and Motion Control Systems of Ships and Ocean Structures*. Department of Marine Technology, the Norwegian University of Science and Technology, Trondheim, Norway, second edition, 2012.
- Jason R. Stanley, editor. *The StationKeep Function: Dynamic Positioning for Remotely Operated Vehicles*, 201 Cousteau Place, Davis, California 95616-5412 U.S.A., 2004. Schilling Robotics, Underwater Magazine.
- Eirik Svendby. Robust control of roV/auvs. Master’s thesis, Norwegian University of Science and Technology, 2007.

Eric Wan. Sigma-point filters: An overview with applications to integrated navigation and vision assisted control. Cambridge, UK, September 2006. Nonlinear Statistical Signal Processing Workshop, IEEE.

A Drawings

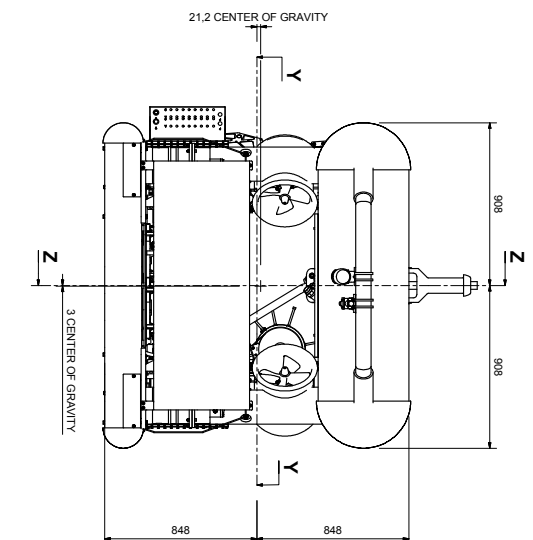
A.1 Thruster Placement



REVISION HISTORY - APPROVED BY					
REV.	DATE	REVISION DESCRIPTION	DWG.	ENG.	HQ.
21	06.10.17	ISSUED FOR MASTER THESIS	P40		
DOK. NO.					
REFERENCE DRAWING					
DESCRIPTION					
<p>This drawing is the property of ILM Subsea and may not be reproduced or transmitted in any form or by any means electronic or mechanical, including photocopying, recording, or by information storage and retrieval systems without written permission from ILM Subsea AS.</p> <div style="text-align: center;">  IKM Subsea AS </div>					
STANDARD PARTS					
NO. OF THIS MATERIAL					
					

[illegible]

A.2 Global Coordinates



VIEW1 (1 : 25)

VIEW3(1:25)

X - Positiv framover (bauretning)

Y - Positiv lateralt utover fra styrbord

Z - Positiv nedover

(etter høyrehåndsregelen)

Koordinatsys


X - Positiv frak

Y - Positiv lar

Z - Positiv ne

(etter høyreh

REV / DATE	RE
Z1 05.11.12	SSS
DOK. NO.	





IKM

 Sub

STANDARD

VIEW2 (1 : 25)

REV.		DATE		REVISION HISTORY / APPROVED BY	
21	05.11.12	ISSUED FOR MASTER THE SIS		DRW. ENGR.	PROJ.
				PHOTO	SCALE
<div> <div>  <p>IKM Subsea AS</p> </div> <div> <p>THIS DRAWING IS THE PROPERTY OF IKM. IT IS LOANED, AND NOT SOLD, TO YOU. IT IS TO BE USED FOR THE DESIGNATION OF OTHERS, UNLESS OTHERWISE STATED. NO PARTS MAY BE REPRODUCED OR COPIED WITHOUT WRITING PERMISSION.</p> </div> </div>					
<div> <div> <p>STANDARD PARTS</p> <p>FOR THE SUBSEA SYSTEMS</p> </div> <div>  </div> </div>					
<div> <div> <p>P315-6000 - MERLIN WR200 ASSEMBLY</p> </div> <div> <p>STD Merlin WR200 - Complete (A) ver21.1 1 OF 1</p> </div> </div>					
A3	SCALE	DATE	REV.	HEET	TOTAL

B Merlin Data

Merlin WR200 - complete (new)

Physical Properties for HKM.10.132.51_Ver2
General Properties:

**Center of Gravity:		{}	
		Material:	1,039 g/cm³
		Density:	3019,249 kg (Relative Error = 0,000245%)
		*Mass:	1865422,78, 399 mm²2 (Relative Error = 0,005466%)
		Area:	2,9082508672E+009 mm³3 (Relative Error = 0,000245%)
		Volume:	
		X:	-2,341 mm (Relative Error = 0,000245%)
		Y:	3,014 mm (Relative Error = 0,000245%)
		Z:	-21,193 mm (Relative Error = 0,000245%)
**Mass Moments of Inertia with respect to Center of Gravity(Calculated using negative integral)		Ixx	1819,601,608,273 kg mm²2 (Relative Error = 0,000245%)
		Iyy Iyy	-2,6968864,126 kg mm²2 (Relative Error = 0,000245%)
		Izz Izz	1203,697,18, 084 kg mm²2 (Relative Error = 0,000245%)
**Mass Moments of Inertia with respect to Global(Calculated using negative integral)		Ixx	1820,985,055, 487 kg mm²2 (Relative Error = 0,000245%)
		Iyy Iyy	-2,679,584, 550 kg mm²2 (Relative Error = 0,000245%)
		Izz Izz	1202,499,57, 803 kg mm²2 (Relative Error = 0,000245%)
**Principal Moments of Inertia with respect to Center of Gravity		I1:	1806,195,213,213 kg mm²2 (Relative Error = 0,000245%)
		I2:	3,0653,30883145E+009 kg mm²2 (Relative Error = 0,000245%)
		I3:	2,90023377515E+009 kg mm²2 (Relative Error = 0,000245%)
**Rotation from Global to Principal		Rc:	2,61 Ang (Relative Error = 0,000245%)
		Ry:	-6,34 Ang (Relative Error = 0,000245%)
		Rz:	-0,37 Ang (Relative Error = 0,000245%)

*Calculations are based on user overwriten values
**Values do not reflect user-overriden mass or volume

3,0649590136613E+009 kg mm²2 (Relative Error = 0,000245%)
-7207982,764 kg mm²2 (Relative Error = 0,000245%)
2,887168166629E+009 kg mm²2 (Relative Error = 0,000245%)
3,06562768108E+009 kg mm²2 (Relative Error = 0,000245%)
-7015121,476 kg mm²2 (Relative Error = 0,000245%)
2,887212140375E+009 kg mm²2 (Relative Error = 0,000245%)

Skid for Merlin WR200

Physical Properties for IKM-1011436
General Properties:

Material:		{	
Density:		1.173 g/cm³	
*Mass:		433,669 kg (Relative Error = 0,000014%)	
Area:		41881338,990 mm² (Relative Error = 0,000818%)	
Volume:		369757350,897 mm³ (Relative Error = 0,000014%)	
** Center of Gravity:			
X:		210,107 mm (Relative Error = 0,000014%)	
Y:		-19,542 mm (Relative Error = 0,000014%)	
Z:		997,832 mm (Relative Error = 0,000014%)	
** Mass Moments of Inertia with respect to Center of Gravity(Calculated using negative integral)			
Ixx		182842049,706 kg mm² (Relative Error = 0,000014%)	
Iyy Iyz		4569077,270 kg mm² (Relative Error = 0,000014%)	
Izz		34693521,784 kg mm² (Relative Error = 0,000014%)	
** Mass Moments of Inertia with respect to Global(Calculated using negative integral)			
Ixx		614799060,933 kg mm² (Relative Error = 0,000014%)	
Iyy Iyz		6349716,485 kg mm² (Relative Error = 0,000014%)	
Izz		-5625843,607 kg mm² (Relative Error = 0,000014%)	
** Principal Moments of Inertia with respect to Center of Gravity			
I1:		179627185,164 kg mm² (Relative Error = 0,000014%)	
I2:		424587719,658 kg mm² (Relative Error = 0,000014%)	
I3:		570801136,913 kg mm² (Relative Error = 0,000014%)	
Rx:		-2,65 deg (Relative Error = 0,000014%)	
Ry:		-5,07 deg (Relative Error = 0,000014%)	
Rz:		1,45 deg (Relative Error = 0,000014%)	
** Rotation from Global to Principal			
*Calculations are based on user overridden values			
**Values do not reflect user-overridden mass or volume			

424787743,076 kg mm² (Relative Error = 0,000014%)
-7161148,100 kg mm² (Relative Error = 0,000014%)
5,67386248,953 kg mm² (Relative Error = 0,000014%)

875723404,263 kg mm² (Relative Error = 0,000014%)
1295407,669 kg mm² (Relative Error = 0,000014%)
5,866596139,122 kg mm² (Relative Error = 0,000014%)

Merlin WR200 - Center of buoyancy

Physical Properties for WK-1011496

General Properties:

** Center of Gravity:	Material:	{ }	
	Density:	10.348 g/cm ³	
	Mass:	3.98515 kg (Relative Error = 0.000185%)	
	Mass:	3.98834200 kg mm ² (Relative Error = 0.000185%)	
	Volume:	2.907196739402E+009 mm ³ (Relative Error = 0.000185%)	
	X:	-29.332 mm (Relative Error = 0.000185%)	
	Y:	-5.887 mm (Relative Error = 0.000185%)	
	Z:	-1.56161 mm (Relative Error = 0.000185%)	
	Ixx	2.0776394492946E+010 kg mm ² (Relative Error = 0.000185%)	
	Iyy	2.0776394492946E+010 kg mm ² (Relative Error = 0.000185%)	
** Mass Moments of Inertia with respect to Center of Gravity(Calculated using negative integral)	Izz	79209719.109 kg mm ² (Relative Error = 0.000185%)	
	Ixy	3.38081579982727E+010 kg mm ² (Relative Error = 0.000185%)	
	Iyz	79209719.109 kg mm ² (Relative Error = 0.000185%)	
	Ixx	3.3385620764715E+010 kg mm ² (Relative Error = 0.000185%)	
	Iyy	51216943.258 kg mm ² (Relative Error = 0.000185%)	
	Izz	3.1805910235790E+010 kg mm ² (Relative Error = 0.000185%)	
	Ixy	3.1778984653550E+010 kg mm ² (Relative Error = 0.000185%)	
	Iyz	3.1778984653550E+010 kg mm ² (Relative Error = 0.000185%)	
	Ixx	2.1590017066850E+010 kg mm ² (Relative Error = 0.000185%)	
	Iyy	2.1590017066850E+010 kg mm ² (Relative Error = 0.000185%)	
** Principal Moments of Inertia with respect to Center of Gravity	Ixx	1.78659247780 kg mm ² (Relative Error = 0.000185%)	
	Iyy	3.18276365.484 kg mm ² (Relative Error = 0.000185%)	
	Izz	2.1590017066850E+010 kg mm ² (Relative Error = 0.000185%)	
	Ixx	2.0776394492946E+010 kg mm ² (Relative Error = 0.000185%)	
	Iyy	2.0776394492946E+010 kg mm ² (Relative Error = 0.000185%)	
	Izz	79209719.109 kg mm ² (Relative Error = 0.000185%)	
	Ixy	3.1778984653550E+010 kg mm ² (Relative Error = 0.000185%)	
	Iyz	3.1778984653550E+010 kg mm ² (Relative Error = 0.000185%)	
	Ixx	2.1590017066850E+010 kg mm ² (Relative Error = 0.000185%)	
	Iyy	2.1590017066850E+010 kg mm ² (Relative Error = 0.000185%)	
** Rotation from Global to Principal	Rx:	-4.05 deg (Relative Error = 0.000185%)	
	Ry:	-2.44 deg (Relative Error = 0.000185%)	
	Rz:	-0.67 deg (Relative Error = 0.000185%)	
	Rx:	-4.05 deg (Relative Error = 0.000185%)	
	Ry:	-2.44 deg (Relative Error = 0.000185%)	
	Rz:	-0.67 deg (Relative Error = 0.000185%)	
	Rx:	-4.05 deg (Relative Error = 0.000185%)	
	Ry:	-2.44 deg (Relative Error = 0.000185%)	
	Rz:	-0.67 deg (Relative Error = 0.000185%)	
	Rx:	-4.05 deg (Relative Error = 0.000185%)	
	Ry:	-2.44 deg (Relative Error = 0.000185%)	

*Calculations are based on user overwriten values

**Values do not reflect user-overriden mass or volume

C Tests

Pull test with original thruster funnels

The pull test was performed with a load cell hooked to the front of the ROV.

All tests were performed pulling in aft direction.

Pull test using all four horizontal thrusters

Thruster gain	Output frequency	Ampere	Pull force
10%	7,7Hz	6,3	0
20%	15,2Hz	6,8	10,0 kg
30%	22,7Hz	6,8	35,0 kg
40%	30,3Hz	6,8	70, 0 kg
50%	38,0Hz	7,9	120 kg
60%	45,7Hz	9,2	170 kg
70%	53,3Hz	10,9	240 kg
80%	60,9Hz	12,7	310 kg
90%	68,5Hz	15,7	400 kg
100%	75,0Hz	18,0	500 kg
	80,0Hz	22,7	540 kg
	83,0Hz	26,7	550 kg

By using curve fitting function in MATLAB(cf tool) for a quadratic function for each of the directions it is possible to come up with the following functions for the drag:

$$F_{surge+} = 1321 |u| u \quad (C.1)$$

$$F_{surge-} = 1221 |u| u \quad (C.2)$$

$$F_{sway+} = 2525 |v| v \quad (C.3)$$

where F is drag in Newton and u and v are respectively velocity through the water in the surge and sway given in m/s. This approach can be seen in figure C.1,C.2 and C.3. It show that there is a remarkably good approximation. There is no linear contributions of significance.

Table 10: Drag test results for Merlin

V_m [m/s]	Drag surge + [N]	Drag surge - [N]	Drag sway + [N]
0.10	14	16	17
0.20	50	54	82
0.30	116	120	204
0.40	206	200	369
0.50	340	317	604
0.60	478	447	887
0.80	862	804	1540
1.00	1324	1270	2506
1.15	-	-	3408
1.25	2054	1930	-
1.50	-	2702	-

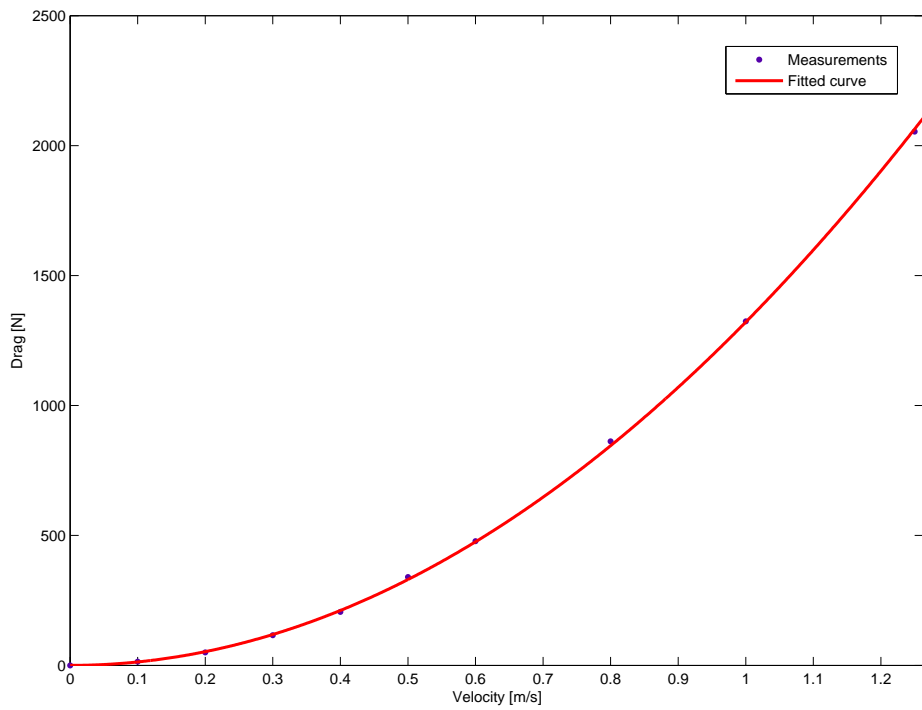


Figure C.1: Fitted curve for the drag in positive surge

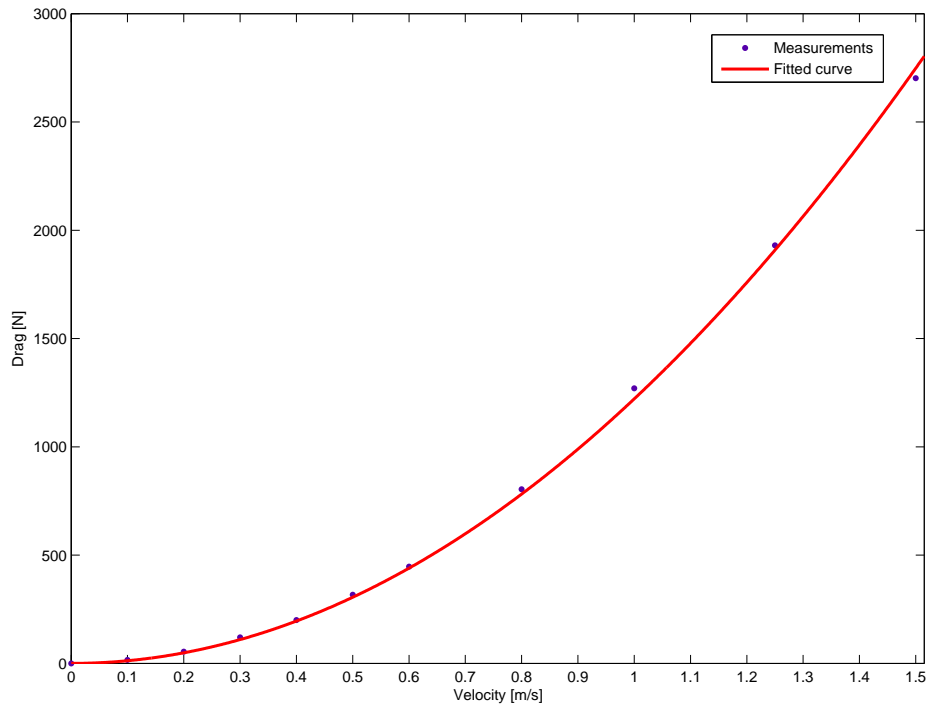


Figure C.2: Fitted curve for the drag in negative surge

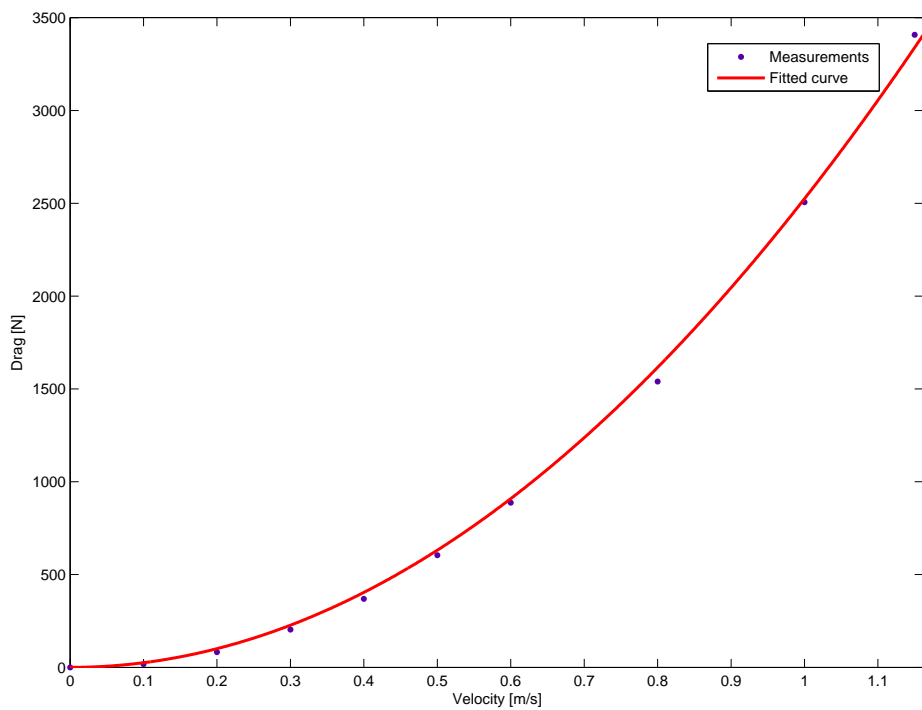


Figure C.3: Fitted curve for the drag in positive sway

D Software

D.1 Commanded Acceleration Control Law for Attitude and Position

```

1  (*scantime*)
2  dt:=UDINT_TO_REAL(P_Cycle_Time_Value)*0,0001;
3
4
5  (*zero-crossing fix for heading error*)
6  IF (psi_hat-psi_d)<= -DEG_TO_RAD(180,0) THEN
7      psi_error:= DEG_TO_RAD(360,0)-(psi_hat-psi_d);
8  ELSIF (psi_hat-psi_d)> DEG_TO_RAD(180,0) THEN
9      psi_error:= -DEG_TO_RAD(360,0)+(psi_hat-psi_d);
10 ELSE
11     psi_error:=(psi_hat-psi_d);
12 END_IF;
13
14 (*clear*)
15 commanded_acc_NED_position_and_attitude_control[0]:= 0,0;
16 commanded_acc_NED_position_and_attitude_control[1]:= 0,0;
17 commanded_acc_NED_position_and_attitude_control[2]:= 0,0;
18 commanded_acc_NED_position_and_attitude_control[3]:= 0,0;
19 commanded_acc_NED_position_and_attitude_control[4]:= 0,0;
20 commanded_acc_NED_position_and_attitude_control[5]:= 0,0;
21
22 (*lambda to PID-gains mapping*)
23 Kp_1:= 3,0*(lambda_1*lambda_1);
24 Kp_2:= 3,0*(lambda_2*lambda_2);
25 Kp_3:= 3,0*(lambda_3*lambda_3);
26 Kp_4:= 1,0*(lambda_4*lambda_4);
27 Kp_5:= 1,0*(lambda_5*lambda_5);
28 Kp_6:= 3,0*(lambda_6*lambda_6);
29 Ki_1:= (lambda_1*lambda_1*lambda_1);
30 Ki_2:= (lambda_2*lambda_2*lambda_2);
31 Ki_3:= (lambda_3*lambda_3*lambda_3);
32 Ki_4:= 0,0*(lambda_4*lambda_4*lambda_4);
33 Ki_5:= 0,0*(lambda_5*lambda_5*lambda_5);
34 Ki_6:= (lambda_6*lambda_6*lambda_6);
35 Kd_1:= 3,0*lambda_1;
36 Kd_2:= 3,0*lambda_2;
37 Kd_3:= 3,0*lambda_3;
38 Kd_4:= 8,0*lambda_4;
39 Kd_5:= 7,0*lambda_5;
40 Kd_6:= 5,0*lambda_6;
41
42 (*P*)
43 commanded_acc_NED_position_and_attitude_control[0]:=Kp_1*(N_hat-N_d);
44 commanded_acc_NED_position_and_attitude_control[1]:=Kp_2*(E_hat-E_d);
45 commanded_acc_NED_position_and_attitude_control[2]:=Kp_3*(D_hat-D_d);
46 commanded_acc_NED_position_and_attitude_control[3]:=Kp_4*(phi_hat-phi_d);
47 commanded_acc_NED_position_and_attitude_control[4]:=Kp_5*(theta_hat-theta_d);
48 commanded_acc_NED_position_and_attitude_control[5]:=Kp_6*(psi_error);

```

```

49
50 (**)
51 Integral_1:= Integral_1+(Ki_1*(N_hat-N_d)*dt);
52 Integral_2:= Integral_2+(Ki_2*(E_hat-E_d)*dt);
53 Integral_3:= Integral_3+(Ki_3*(D_hat-D_d)*dt);
54 Integral_4:= Integral_4+(Ki_4*(phi_hat-phi_d)*dt);
55 Integral_5:= Integral_5+(Ki_5*(theta_hat-theta_d)*dt);
56 Integral_6:= Integral_6+(Ki_6*(psi_error)*dt);
57
58 (*Soft integral reset*)
59 IF Reset_Integral_1=TRUE THEN
60     Integral_1:= Integral_1-Integral_1*0,5*dt;
61     IF ABS(Integral_1)<0,01 THEN
62         Integral_1:= 0,0;
63     END_IF;
64 END_IF;
65 IF Reset_Integral_2=TRUE THEN
66     Integral_2:= Integral_2-Integral_2*0,5*dt;
67     IF ABS(Integral_2)<0,01 THEN
68         Integral_2:= 0,0;
69     END_IF;
70 END_IF;
71 IF Reset_Integral_3=TRUE THEN
72     Integral_3:= Integral_3-Integral_3*0,5*dt;
73     IF ABS(Integral_3)<0,01 THEN
74         Integral_3:= 0,0;
75     END_IF;
76 END_IF;
77 IF Reset_Integral_4=TRUE THEN
78     Integral_4:= 0,0;
79     IF ABS(Integral_4)<0,01 THEN
80         Integral_4:= 0,0;
81     END_IF;
82 END_IF;
83 IF Reset_Integral_5=TRUE THEN
84     Integral_5:= 0,0;
85     IF ABS(Integral_5)<0,01 THEN
86         Integral_5:= 0,0;
87     END_IF;
88 END_IF;
89 IF Reset_Integral_6=TRUE THEN
90     Integral_6:= Integral_6-Integral_6*0,5*dt;
91     IF ABS(Integral_6)<0,01 THEN
92         Integral_6:= 0,0;
93     END_IF;
94 END_IF;
95

```

D SOFTWARE Commanded Acceleration Control Law for Attitude and Position

```

101  (*Anti-windup routine*)
102
103  IF (commanded_acc_NED_position_and_attitude_control[0]+Integral_1)<-antiwindup_1 THEN
104      Integral_1:= -antiwindup_1-commanded_acc_NED_position_and_attitude_control[0];
105  ELSIF (commanded_acc_NED_position_and_attitude_control[0]+Integral_1)>antiwindup_1 THEN
106      Integral_1:= antiwindup_1-commanded_acc_NED_position_and_attitude_control[0];
107  END_IF;
108
109  IF (commanded_acc_NED_position_and_attitude_control[1]+Integral_2)<-antiwindup_2 THEN
110      Integral_2:= -antiwindup_2-commanded_acc_NED_position_and_attitude_control[1];
111  ELSIF (commanded_acc_NED_position_and_attitude_control[1]+Integral_2)>antiwindup_2 THEN
112      Integral_2:= antiwindup_2-commanded_acc_NED_position_and_attitude_control[1];
113  END_IF;
114
115  IF (commanded_acc_NED_position_and_attitude_control[2]+Integral_3)<-antiwindup_3 THEN
116      Integral_3:= -antiwindup_3-commanded_acc_NED_position_and_attitude_control[2];
117  ELSIF (commanded_acc_NED_position_and_attitude_control[2]+Integral_3)>antiwindup_3 THEN
118      Integral_3:= antiwindup_3-commanded_acc_NED_position_and_attitude_control[2];
119  END_IF;
120
121  IF (commanded_acc_NED_position_and_attitude_control[3]+Integral_4)<-antiwindup_4 THEN
122      Integral_4:= -antiwindup_4-commanded_acc_NED_position_and_attitude_control[3];
123  ELSIF (commanded_acc_NED_position_and_attitude_control[3]+Integral_4)>antiwindup_4 THEN
124      Integral_4:= antiwindup_4-commanded_acc_NED_position_and_attitude_control[3];
125  END_IF;
126
127  IF (commanded_acc_NED_position_and_attitude_control[4]+Integral_5)<-antiwindup_5 THEN
128      Integral_5:= -antiwindup_5-commanded_acc_NED_position_and_attitude_control[4];
129  ELSIF (commanded_acc_NED_position_and_attitude_control[4]+Integral_5)>antiwindup_5 THEN
130      Integral_5:= antiwindup_5-commanded_acc_NED_position_and_attitude_control[4];
131  END_IF;
132
133  IF (commanded_acc_NED_position_and_attitude_control[5]+Integral_6)<-antiwindup_6 THEN
134      Integral_6:= -antiwindup_6-commanded_acc_NED_position_and_attitude_control[5];
135  ELSIF (commanded_acc_NED_position_and_attitude_control[5]+Integral_6)>antiwindup_6 THEN
136      Integral_6:= antiwindup_6-commanded_acc_NED_position_and_attitude_control[5];
137  END_IF;
138
139  (*PI+D sum*)
140  commanded_acc_NED_position_and_attitude_control[0]:=-(commanded_acc_NED_position_and_attitude_control[0]+Integral_1)-
141  Kd_1*(N_dot_hat-N_dot_d);
142  commanded_acc_NED_position_and_attitude_control[1]:=-(commanded_acc_NED_position_and_attitude_control[1]+Integral_2)-
143  Kd_2*(E_dot_hat-E_dot_d);
144  commanded_acc_NED_position_and_attitude_control[2]:=-(commanded_acc_NED_position_and_attitude_control[2]+Integral_3)-
145  Kd_3*(D_dot_hat-D_dot_d);
146  commanded_acc_NED_position_and_attitude_control[3]:=-(commanded_acc_NED_position_and_attitude_control[3]+Integral_4)-
147  Kd_4*(phi_dot_hat);
148  commanded_acc_NED_position_and_attitude_control[4]:=-(commanded_acc_NED_position_and_attitude_control[4]+Integral_5)-
149  Kd_5*(theta_dot_hat);
150  commanded_acc_NED_position_and_attitude_control[5]:=-(commanded_acc_NED_position_and_attitude_control[5]+Integral_6)-
151  Kd_6*(psi_dot_hat-psi_dot_d);

```


E Mix

```
% pulltest result

test=[-0 -0;-15.2 -20;-22.7 -50;-30.3 -80;
-38 -140;-45.7 -170;-53.3 -250;-60.9 -330;-68.5
-410;-83 -575;0 0;15.2 20;22.7 50;30.3 80;38
140;45.7 170;53.3 250;60.9 330;68.5 410;83 575];

%pulltest 1st col: freq[hz] 2nd col: force [kg]

%decompose and convert to N.
one_thruster_normal_test=test*[1 0;0 (1/0.71)*(1/4)*9.81]
%4 thrusters . 45 degree. cos(45)=0.71 see drawing of the Merlin

%ONLY USING 3 points. freq(-83 0 +83)

%USE CFTOOL

%General model Sin1:
%      f(x) = a1*sin(b1*x)
%Coefficients (with 95% confidence bounds):
%      a1 =      83 (36.25, 129.8)
%      b1 = 0.0007909 (-4.631e+12, 4.631e+12)

%Goodness of fit:
%  SSE: 4.039e-28
%  R-square: 1
%  Adjusted R-square: 1
%  RMSE: 4.874e-15
```

```
function [max_current_in_rad_dir] = DP_capability(rad)
rad=rad+pi./2;
T=[0.71 0.71 0.71 0.71;-0.71 0.71 -0.71 0.71];
%Thrusterconfig.matrix
temp=[0:0;0;0];
x=0;
while norm(temp,inf)<1900
% 1900: max thruster force
temp=(T')*((T*(T'))^-1)*[x*sin(rad);x*cos(rad)];
%find max force in a rad direction. rad=0=surge+
x=x+1;
end
surge_current=x*abs(sin(rad));
sway_current=x*abs(cos(rad));
current_max_surge=sqrt(surge_current/1321);
%equal force from sea current in surge (in m/s)
current_max_sway=sqrt(sway_current/2525);
max_current_in_rad_dir=sqrt((current_max_sway.^2)+(current_max_surge.^2));
%equal force from sea current in rad dir(in m/s)
end

%USE DP_capability(rad) to plot DP_capability plots! :
%i=1;
%for R=0:0.01:2*pi
%vektor(i)=DP_capability(R);
%i=i+1;
%end
%polar(R,vektor,'--b');
%view(+90,-90);
```

F Sensors

F.1 TOGSNAV

1 Introduction

1.1 Warnings and Notes

Throughout the manual the following symbols are used:



Indicates a warning.

Failure to follow these instructions will result in serious injury, damage to equipment or incorrect operation of equipment.



Indicates a note.

This indicates important information that should be followed to ensure correct operation of the unit.

1.2 General Description

1.2.1 System Overview

The CDL TOGS-NAV is a solid state gyrocompass based around a Fibre Optic Gyro (FOG).

The TOGS-NAV contains an Inertial Measurement Unit (IMU), a processor, a communications interface, a power supply system, an RDI Workhorse DVL and an optional Valeport IPS pressure / depth sensor.

This assembly, together with the navigation processor, provides the TOGS-NAV a full self-contained Attitude Heading Reference system (AHRS).

The TOGSNAV IMU is required to be initialized with accurate Latitude for the location that is will be used.

After the initial static alignment initialisation has completed, the TOGSNAV will attain full AHRS mode.

At this time the TOGSNAV can be moved and can periodically be updated with a new position Latitude to maintain accuracy.

2.5 Electrical Installation

The TOGS-NAV can be configured with two different connectors: Standard subsea wet mate connectors or SEANET connectors.

2.5.1 TOGSNAV Connector

The TOGS-NAV normally has one Umbilical connector that is wired simply with the RS232 Port-1 user control and data output interface.

An optional second DVL output data connector can be supplied with the TOGSNAV unit as required.

An alternative option that is available is the expansion of the one Umbilical connector that is wired to provide the RS232 Port-1 user control and data output interface, plus the RS232 Port-2 data output and the RS232 DVL data output.

Table 1: TOGSNAV Connector Options

Connector	Function
1508	Primary Communication Port (Port 1) plus DC Power
Optional DVL Port	DVL Data Output Port (115K/8/Odd/1)
Optional 1508	Primary Communication Port (Port 1) plus DC Power Secondary Communication Data Output Port (Port 2) DVL Data Output Port (115K/8/Odd/1)



If any of these connectors are not in use, they **MUST** be fitted with blanking plugs.

Interface with the TOGSNAV unit can be by RS232 serial communications only at standard baud rates in the range 9600 – 115200 bps.

3.2 Embedded software Main Menu

Customisation of the TOGS-NAV's operation is allowed through embedded firmware. Various configuration options allowed in the embedded firmware have been compiled into a logical hierarchical menu structure and are accessible on Port 1.

The menu can be accessed through any PC terminal program by entering “**menu**” [ENTER]. This will stop data output and display the Main Menu.

The Main Menu provides options for control of the TOGS-NAV and allows configuration of the communication ports, output strings and various navigation settings. The Main Menu also provides options to restart the TOGS-NAV and to set latitude and longitude.

Navigating the menu is done by entering the menu number ID and pressing [ENTER].

Pressing “0” [ENTER] will always go back to the previous menu.



To retain configuration during power off, the settings must be saved to FLASH using the Save settings to FLASH in the menu or the SAVE Quick Command. This process will take about a couple of seconds, during which power must be maintained.

3.2.1 Main menu

The main menu has the following entries

```
TOGSNAV 1.1.5 - Main - Tue 27 Nov 2102 18:16:52
Mode = 9   Status = 00   IPS = dBar
1. Set initial latitude [57.1910000 deg]
2. Ports & strings
3. Advanced
4. Configuration dump
0. Back
```

- [1] Allows the user to set the initial latitude. The value is entered as decimal degrees. Negative latitude is on the southern hemisphere – [see Quick Command in section 3.4.](#)
- [2] Allows the user to change port and string settings.
- [3] Allows the user to change advanced options.
- [4] Dumps all the settings as text – [see section 3.2.14](#)
- [0] Will quit the menu, asking if the user want to save any changes which might have made any.



The menu header provides information on the Firmware, the menu level, the system Data and Time, the IMU Mode and any active Status errors, plus the IPS depth sensor units of measure.



Entering incorrect latitude will decrease the accuracy of the TOGS-NAV. The entered latitude should be within 1 degree of the actual latitude where the system is used to ensure proper operation of the system.



3.2.2 Ports & strings menu

```
TOGSNAV 1.1.5 - Ports & strings - Tue 27 Nov 2102 18:19:04
Mode = 9   Status = 00   IPS = dBar
1. Port 1
2. Port 2
0. Back
```

[1] Enters the port 1 menu

[2] Enters the port 2 menu

If the user has made any changes to the port settings without saving, an asterisk [*] will be displayed next to the port name.

3.2.3 Port menu

```
TOGSNAV 1.1.5 - Port 1 - Tue 27 Nov 2102 18:19:12
Mode = 9   Status = 00   IPS = dBar
1. Output strings [TOGSNAV (2.50 Hz)]
2. Baud rate      [115200]
3. Mode           [RS232]
4. Parity         [None]
5. Stopbits       [1]
0. Back
```

[1] Enters the output strings menu. The currently selected strings are displayed in brackets.

[2] Allows the user to change the baud rate of the port

[3] Allows the user to change between RS232, RS485 and RS422 mode

[4] Allows the user to select No, Even or Odd parity

[5] Allows the user to select 1 or 2 stop bits

If you make changes to a port, an “Apply settings” option appear, which, if selected, will apply the current settings to the port.



Decreasing the baud rate of a port may cause it to have insufficient bandwidth to send all the selected data strings. The menu system will issue a warning if the bandwidth is too low.



Selecting RS422 on port 1 will make it impossible to get into the TOGS-NAV menu and change the setting back.

Always test a setting by choosing “Apply settings” before saving settings to flash. Incorrect settings may make it impossible to communicate with the TOGS-NAV.



3.2.4 Output strings menu

```
TOGSNAV 1.1.5 - Output strings - Tue 27 Nov 2102 18:19:21
Mode = 9   Status = 00   IPS = dBar
1. TOGS                                9. TSS2 (no status)
2. TOGSNAV [2.50 Hz]                  A. Watson/Tritech Gyro
3. MiniRLG1                            B. Tokimec 2
4. MiniRLG2                            C. Custom 1
5. MiniPos3 (binary)                  D. Custom 2
6. Tokimec 1                          E. Custom 3
7. TCM2 HPR                           F. Custom 4
8. HMR3000
0. Back
```

The output menu allows you to define which output strings are selected and at what interval. Selecting a menu point will enter a menu where you can change the settings for that string.



Selecting too many strings for simultaneous output on one port and/or too high an update rate can exceed the bandwidth of the output ports baud rate. The menu system will issue a warning if the limit has been reached.

3.2.5 String menu

```
TOGSNAV 1.1.5 - TOGS - Tue 27 Nov 2102 18:19:31
Mode = 9   Status = 00   IPS = dBar
1. Enabled [Yes]
2. Frequency [2.50 Hz (400 ms)]
3. Modes [Any]
0. Back
```

Selecting [1] toggles if the string is selected.
Selecting [2] changes the frequency at which the string is output.
Selecting [3] changes which modes the string is sent out in.

Settings that are changed in this menu are applied immediately.

Direct Communication with the DVL

In order to communicate directly with the DVL in order to check or possibly change DVL settings, a special command can be sent to make the TOGS-NAV pass through any message to the DVL. It will also pass every message from the DVL to the user.

The special command that puts the TOGS-NAV in this special mode is:

\$cdl.directcomm.dvl.on<cr><lf>

This command will set the TOGS-NAV into the special mode and send a [BREAK] command to the DVL which takes the DVL into command mode. This will ensure the DVL is ready to accept commands send from the TOGS-NAV.

The special command that takes the TOGS-NAV back to normal mode is:

\$cdl.directcomm.dvl.off<cr><lf>

After sending this command the TOGS-NAV will start to output data again.

Example 1: In order to check the DVL output string format the following commands should be sent:

\$cdl.directcomm.dvl.on<cr><lf>

PD?<cr><lf>

The DVL will respond with:

PD = 00 ----- Data Stream Selected

Example 2: In order to set the DVL output string to PD0 format and save to the DVL, the following commands should be sent:

\$cdl.directcomm.dvl.on<cr><lf>

PD0<cr><lf>

CK<cr><lf>

\$cdl.directcomm.dvl.off<cr><lf>

When the DVL is in direct mode and the native DVL "CS" command is sent in order to read out raw DVL data. It is possible to break into the DVL menu again by sending a break command to the DVL. This can be achieved by sending the following command:

\$cdl.dvl.break<cr><lf>

This will sent the break command to the DVL and it will be possible to send additional 'native DVL' commands to the DVL again.



When setting the TOGS-NAV in this special direct communication mode, the TOGS-NAV might output a "Packet loss" message. This can be ignored.

4 Data Output

4.1 Data Formats

The TOGS-NAV is able to output a range of industry standard ASCII strings to enable it to be interfaced to other systems.

The predefined string outputs are listed below and are changed via the menu system.

It is also possible to define custom strings from the menu using the output string generator.

Figure 6 shows the 'normal' CDL product sign convention.



Figure 5: Sign convention for TOGS and MiniRLG strings



CDL 'standard' Pitch and Roll Convention:

+VE Pitch Bow up / +VE Roll Port up



4.3.2 TOGSNAV

(+ve Pitch Bow Up, +ve Roll Port Side Up)

\$TGNV,hhh.hh,+ppp.pp,+rrr.rr,cc,m,,dddd.ddd,,+x.xx,+y.yy,qqq.qq,www.ww,eee.aa,aaa.aa
<cr><lf>

Where:

hhh.hh	is heading in degrees hhh(deg).hh(decimal)
ppp.pp	is pitch in degrees ppp(deg).pp(decimal) see figure 6
rrr.rr	is roll in degrees rrr(deg).rr(decimal) see figure 6
cc	is TOGSNAV error status flags - see section 4.2
m	is the TOGSNAV IMU mode flag - see section 4.2
dddd.ddd	is depth in meters dddd(m).ddd(decimal)
x.xx	is velocity in x-direction with respect to the seabed in m/s x(m/s).xx(decimal)
y.yy	is velocity in y-direction with respect to the seabed in m/s y(m/s).yy(decimal)
qqq.qq	is distance to the seabed from DVL beam 1 in meters qqq(m).qq(decimal)
www.ww	is distance to the seabed from DVL beam 2 in meters www(m).ww(decimal)
eee.aa	is distance to the seabed from DVL beam 3 in meters eee(m).aa(decimal)
aaa.aa	is distance to the seabed from DVL beam 4 in meters aaa(m).aa(decimal)



5.4.7 Identifiers

Table 10: Identifiers for custom string generator

Identifier	Name	Format
acx	X-axis Accelerometer reading (unscaled)	Floating point
acy	Y-axis Accelerometer reading (unscaled)	Floating point
acz	Z-axis Accelerometer reading (unscaled)	Floating point
br1	DVL beam 1 bottom range in cm	UInt16
br2	DVL beam 2 bottom range in cm	UInt16
br3	DVL beam 3 bottom range in cm	UInt16
br4	DVL beam 4 bottom range in cm	UInt16
bvx	DVL X bottom velocity in mm/s	Int16
bvy	DVL Y bottom velocity in mm/s	Int16
bvz	DVL Z bottom velocity in mm/s	Int16
crc	CRC	Hexadecimal
dep	Depth in dBar pressure units (see Note 3)	Floating point
dvs	DVL reference layer status	HEX8
dvt	DVL temperature in degrees Celsius	Int16
epo	Epoch	Integer
erf	Error flags	HEX8
frm	IMU frame counter	Int32
gwx	X-axis angular velocity in rad/sample (see Note 2)	Floating point
gwy	Y-axis angular velocity in rad/sample (see Note 2)	Floating point
g wz	Z-axis angular velocity in rad/sample (see Note 2)	Floating point
hea	Heading in degrees	Floating point
ihr	Change in heading in degrees/minute (see Note 1)	Floating point
ipr	Change in pitch in degrees/minute (see Note 1)	Floating point
lrr	Change in roll in degrees/minute (see Note 1)	Floating point
mod	Mode	Hexadecimal
pit	Pitch in degrees	Floating point
rol	Roll in degrees	Floating point
rve	DVL reference layer error velocity in mm/s	Int16
rvx	DVL reference layer X velocity in mm/s	Int16
rvy	DVL reference layer Y velocity in mm/s	Int16
rvz	DVL reference layer Z velocity in mm/s	Int16
tmp	IMU temperature in degrees Celsius *100	Int32



Note 1: Delta or “change in” values display the difference between the current value and the value at the time of the previous output string.

Note 2: The TOGS processor sample rate is fixed at 50Hz

Note 3: TOGSNAV firmware version 1.1.5 configuration of the IPS sensor is set to output pressure units in decibar, and not meters of depth of seawater using the UNESCO pressure to depth calculation formula.

5.4.8 Operators

Table 11: Operator Functions for Custom String Generator Fields

Field	Operation
+	Add to value
-	Subtract from value
/	Divide by
*	Multiply by
M	Limit value to maximum
m	Limit value to minimum
a	Absolute value (modulus)
s	Sign of value

CDL acknowledges that users without previous programming experience may have difficulty with the custom string generator.

If you require a specific string format, which is not included in the TOGS-NAV menu options, please contact us and we shall consider including it.

CDL contact details can be found at the end of this document.



6 Specifications

AHRS Specification:

Heading Accuracy:	0.5 deg sec(lat)*
Pitch and Roll	0.1 deg*

Power Requirements:

Voltage	18-30Vdc
Power	<20W

Telemetry Interface:

Serial outputs	Port 1 isolated 2-way communications Port 2 data output DVL PD0 data output (fixed @ 115K/8/Odd/1) RS232 RS422, RS485 with Burton connectors
Serial protocol options	

Weight and Dimensions:

Subsea unit (3000m)	241mm dia x 388mm
Weight in Air	16kg
Weight in Water	6.7kg

Mounting Holes:

M8 on 157mm square pitch

DVL 1200KHz:

Depth	3000m
Accuracy	+/- 0.2%
Drift	+/- 2mm/s
Range	0.7m (min) 30m (max)

Optional Pressure / Depth Sensor:

Range	300Bar
Accuracy	+/- 0.01% Range
Resolution	0.0001 Bar



TOGSNAV firmware version 1.1.5 configuration of the IPS sensor is set to output pressure units in decibar, and not meters of depth of seawater using the UNESCO pressure to depth calculation formula.

*1 Sigma RMS

F.2 WORKHORSE DVL



Workhorse Navigator

DOPPLER VELOCITY LOG (DVL)

Precision Navigation for the Marine Environment

The Workhorse Navigator is the industry's first choice for precision navigation applications. Teledyne RDI's highly acclaimed Doppler Velocity Log (DVL) provides precise velocity and altitude updates for a wide variety of underwater tasks.

The highly flexible design allows the unit to be used in a standalone configuration or integrated with other navigation systems.

The compact and powerful Workhorse Navigator provides:

- Patented BroadBand processing technology, providing users with both short and long-term high-precision velocity data
- Reliable and accurate high-rate navigation and positioning data
- Proven bottom detection algorithms, and single ping bottom location, for robust and reliable bottom tracking over indeterminate terrain
- Superior low-altitude bottom tracking capability
- Real-time current profiling data



Navigator Applications:

- Subsea vehicle and surface vessel navigation
- Hydrographic, geophysical, and oceanographic survey positioning data
- LBL and USBL position aiding
- Spool piece metrology
- Inertial navigation correction and integration
- Cable burial operations
- Deep water positioning
- Station keeping and autopilot control
- Pipeline touchdown monitoring
- Dredge spoils, plume, and sediment tracking

Navigator full suite of capabilities:

- Bottom track velocity
- Water track velocity
- Altitude: 4 individual measurements
- Error velocity (data quality indicator)
- Temperature
- Heading/Tilt
- Acoustic echo intensity
- Pressure and depth (optional)
- Current profiling (optional)



TELEDYNE
RD INSTRUMENTS
Everywhere you look™

Workhorse Navigator

DOPPLER VELOCITY LOG (DVL)



Technical Specifications

Model	WHN 300	WHN 600	WHN 1200
Bottom Velocity			
Single-ping precision			
Std dev at 1m/s ¹	±0.3cm/s	±0.3cm/s	±0.3cm/s
Std dev at 3m/s ¹	±0.6cm/s	±0.5cm/s	±0.4cm/s
Std dev at 5m/s ¹	±0.8cm/s	±0.6cm/s	±0.5cm/s
Long-term accuracy	±0.4%±0.2cm/s	±0.2%±0.1cm/s	±0.2%±0.1cm/s
Minimum altitude ²	1.0m	0.7m	0.5m (0.25 optional)
Maximum altitude ²	200m	90m	30m
Parameters			
Velocity range ³	±10m/s	±10m/s	±10m/s
Velocity resolution	0.1cm/s	0.1cm/s	0.1cm/s
Ping rate	7Hz max	7Hz max	7Hz max
Water Reference Velocity			
Accuracy	±0.4% ±0.2cm/s	±0.3% ±0.2cm/s	±0.2% ±0.1cm/s
Layer size	selectable	selectable	selectable
Minimum range	1m	0.7m	0.25m
Maximum range	110m	50m	18m
Environmental			
Operating temperature	-5 to 45°C	-5 to 45°C	-5 to 45°C
Storage temperature	-30 to 75°C	-30 to 75°C	-30 to 75°C
Depth rating	3000m or 6000m		
Weight in air:			
3000m	15.8kg	15.8kg	12.4kg
6000m	20.1kg	20.1kg	18.0kg
Weight in water:			
3000m	8.8kg	8.8kg	6.1kg
6000m	13.6kg	13.6kg	12.1kg
Power			
DC input	20–50VDC, external supply (48VDC typical)		
Current	0.4A minimum power supply capability		
Transmit ⁴			
Peak power @ 24VDC	66w	21w	8w
Average power (typical)	8w	3w	3w

¹Standard deviation refers to single-ping horizontal velocity, specified at half the maximum altitude.

²@5°C and 35 ppt, 42VDC.

³Maximum bottom-tracking range may be reduced due to flow noise at high speed and/or cavitation.

⁴@ 15% duty cycle at peak power (standby 1mW).

Standard Sensors

Compass: ±2° @ 60° dip, 0.5g
Tilt: ±0.5° up to ±15°
Temperature: -5° to 45°C

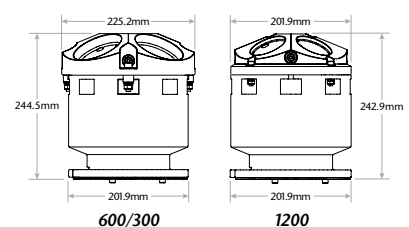
Hardware

Configuration: 4-beam Janus array convex transducer, 30° beam angle
Communications: NMEA0183, ASCII or binary outputs at 1200–115,200 baud user-selectable; serial port is switch-selectable for RS232 or RS422
Trigger inputs: 1) ASCII; 2) RDS3; 3) low latency

Options

- Current profiling firmware upgrade
- Integrated pressure sensor (±0.25% full scale)
- 25m serial/DC/computer cable
- 5m serial/DC/computer cable
- Internal memory cards (2GB max)
- Enhanced low altitude bottom tracking for model 1200

Dimensions



Better the DVL you know...

TELEDYNE
RD INSTRUMENTS
Everywhere you look™
www.rdinstruments.com
www.dvlnav.com

Free 24/7 emergency support

Teledyne RD Instruments

14020 Stowe Drive, Poway, CA 92064 USA
 Tel. +1-858-842-2600 • Fax +1-858-842-2822 • E-mail: rdisales@teledyne.com
 Les Nertieres 5 Avenue Hector Pintus 06610 La Gaude France
 Tel. +33-49-211-0930 • Fax +33-49-211-0931 • E-mail: rdie@teledyne.com

Specifications subject to change without notice. ISO 9001:2008 certification applicable to Poway, CA facility only.
 © 2006 Teledyne RD Instruments, Inc. All rights reserved. Nav-1004, Rev. 12/11



F.3 DVL-Water Referance Layer Settings

BK – Water-Mass Layer Mode

Purpose	Selects the ping frequency of the water-mass layer ping
Format	BK <i>n</i>
Range	<i>n</i> = 0 to 3
Default	BK0



Recommended Setting. The default setting for this command is recommended for most applications.

Description BK selects how often the ADCP performs a water-mass layer ping while bottom tracking. The number of water-mass layer pings per ensemble is dependent on the BP command (bottom pings per ensemble) and this command setting. Use the BL command to set the location of the water-mass layer.

Table 4: Water-Mass Reference-Layer Modes

Command	Description
BK0	Disables the water-mass layer ping.
BK1	Sends a water-mass layer ping after every bottom-track ping
BK2	Sends a water-mass layer ping after every bottom-track ping that is unable to find the bottom.
BK3	Disables the bottom-track ping and enables the water-mass ping.

BL – Water-Mass Layer Parameters

Purpose	Sets bottom-track water-mass layer boundaries and minimum layer size.
Format	BL <i>mmm,nnnn,fff</i>
Range	<i>mmm</i> = Minimum Layer Size (0 to 999 decimeters) [meters x 10] <i>nnnn</i> = Near Layer Boundary (0 to 9999 decimeters) [meters x 10] <i>fff</i> = Far Layer Boundary (0 to 9999 decimeters) [meters x 10]
Default	BL320,640,960 (150 kHz), BL160,320,480 (300 kHz), BL80,160,240 (600 kHz), BL40,60,100 (1200kHz), BL20,20,40 (2400kHz)



Recommended Setting. The default setting for this command is recommended for most applications.

Description The BL command sets a water-mass layer. You can use this layer as a reference point when the bottom is out of range or is incorrect. Water-mass layer output data are available when both BK - Water-Mass Layer Mode and BP - Bottom-Track Pings Per Ensemble are non-zero values, and the bottom must be at least the Minimum Layer Size + Near Layer Boundary + 20% of the reported depth away from the transducer. The Far Layer Boundary (fff) must be less than the maximum profiling distance or the ADCP sends Error Code 011.

The user-defined water-mass layer is used unless the minimum layer comes within 20% of the water boundary (sea floor for down-looking systems; surface for up-looking systems). As the user-defined water-mass layer comes within 20% of the boundary (Figure 4, B), the layer compresses in size until the minimum water-mass layer size is reached. When the boundary moves closer to the transducer (Figure 4, C), no water mass ping will be sent.



The water-mass layer is operational only if BP > zero and BK > zero.

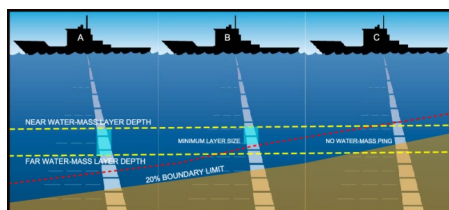


Figure 4. Water-Mass Layer Processing

BM – Bottom Track Mode

Purpose Sets the Bottom Track mode.

Format BM*n*

Range $n = 4, 5$, (see description), 7 (available as a feature upgrade for 1200 kHz WorkHorse ADCP ADCPs with firmware version 16.19 or higher)

Default BM5 (150, 300, 600, and 1200 kHz), BM6 (2400 kHz)



Recommended Setting. The default setting for this command is recommended for most applications.



The BM command is not available for systems with standard Bottom Track (BT-RA - see [OL command](#)).

Description See below

Bottom Track Mode 4

Bottom Track Mode 4 uses the correlation side-peak position to resolve velocity ambiguities. It lengthens the lag at a predetermined depth to improve variance.

Bottom Track Mode 5

Bottom Track Mode 5 is similar to Bottom Track Mode 4, but has a lower variance in shallow water by a factor of up to four. In very shallow water at slow speeds, the variance is lower by a factor of up to 100. Bottom Track Mode 5 also has a slightly slower ping rate than Bottom Track Mode 4.



Bottom Mode 5 (default setting) will shift to Bottom Mode 4 if the conditions warrant.

The ADCP limits searching for the bottom to the value set by the BX command (max bottom tracking altitude) + 0.5 transmit length. This allows a faster ping rate when the bottom altitude is close to the BX command setting.

Table 5: BM4/BM5 Minimum Tracking Depths

Frequency (kHz)	BM4/BM5 Minimum Tracking Depths (m)
150	2.0
300	1.5
600	1.0
1200	0.8

G Merlin Overview

ROV Specifications:

Depth rating	3000 msw
Length	2.8 m
Width	1.8 m
Height	1.7 m
Weight	2800 kg
Manipulator	Schilling Titan 4 (or client spec.)
Manipulator	Schilling Rig master (or client spec.)
Thrusters	8 of Electrical 12" Dual Counter Rotating Propellers
Configuration	4 of Horizontal (vectored), 4 off Vertical
Pulling force	8 kN Forward / Aft. / Lateral
	11 kN Vertical
Auxiliary Tool HPU	1 of 18-30 kW Electrical Hydraulic Power Pack
	49-80 l/min adjustable up to 315 bar
Auxiliary ROV HPU	1 of 8-18 kW Electrical Hydraulic Power Pack
	20-49 l/min adjustable up to 250 bar
Valve pack 1	8 of proportional NG 3 valves
Valve pack Tool	8 of proportional NG 3 valves & 1 off Ng 10
Subsea Electrical interface	Communication: RS 232, RS 422, RS 485, Ethernet, fiber (HD)
	Power: 24V, 110V, 3000V
Cameras	1 of Low Light Camera (pan & tilt)
	1 of Color & Zoom Camera (pan & tilt)
	2 fixed color cameras (on front bar).
	2 of color camera (one rear, one center for TMS docking)
	Total number of camera slots: 8 (prepared for add. pan & tilt)
Lights	4 of Q-LED, 3 of MV-LED
Sensors:	
Depth	Digiquartz & altimeter
Heading	Gyro - as specified by client
Pitch & Roll	+/- 20 degrees
Sonar	MS-1000
Auto functions	Auto Heading / Auto Depth / Auto Altitude
Tooling	Wire cutter, ROV hook/shackle, rope cutter, grinder - optional tools
	according to client request
Power Requierments:	
ROV	250 kVA
Control - Container	30 kW, 440V/50-60Hz

This is standard equipment for the Merlin WR200. Different options for lighting, cameras, manipulator arms, tools etc. may be selected.

H File Attachment

- ControlContainerDP.cpx
- cx_programmer_dp_system_screen_capture.mp4
- force_control_free_tour.mp4
- station_keeping_merlin_wr_200_seen_from_seabed_camera.mov
- station_keeping_merlin_wr_200_wave_zone.mp4