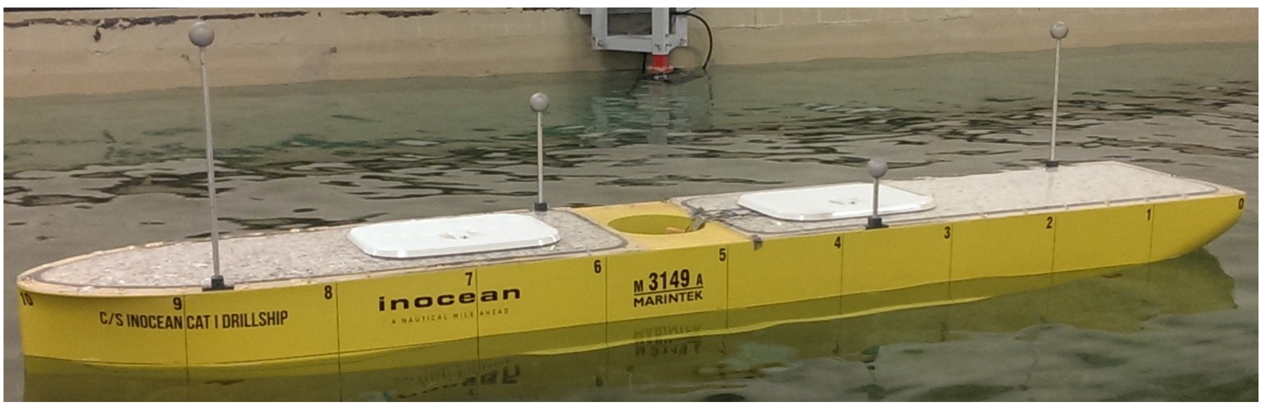
# CyberShip Arctic Drillship

# User Manual




NTNU – Trondheim
Norwegian University of
Science and Technology

Faculty of Engineering Science and Technology

Department of Marine Technology

# Preface

The purpose of this document is to provide a manual that ease the process of using CyberShip Arctic Drillship(CSAD), and concerns only software and hardware of CSAD specifically. For information about the Marine Cybernetics Laboratory(MCLab) and how to implement custom control systems on the vessel, the reader is referred to the CSAD ROS repo, which can be found on GitHub: https://github.com/NTNU-MCS/

# Structure of document

This User Manual is divided in three parts:

- Technical description(hardware, software, mathematical models etc.)
- Operation manual(launching, operation and demolition instructions)
- In the Appendix, a description of the extended IMU system is given(4 IMUs)

Table 1: CSAD main data

| **Parameter** | **Value** |
|---|---|
| Length over all | 2.578 [m] |
| Beam | 0.440 [m] |
| Depth | 0.211 [m] |
| Design draft | 0.133 [m] |
| Weight | 127.92 [kg] |
| Scale | 1:90 |
| RPi IP-address | 192.168.0.123 |
| Qualisys body | (960, -190, -575) [mm] |
| MATLAB Version | 2016b |
| LabVIEW Version | 2017 |
| VeriStand Version | 2017 |

# Known errors and further work

There are some known errors and weaknesses on CSAD:

- Due to a lack of any indexing feature on the belt-wheels on the servos , the wheels can slide around while the servos are stationary. This means the offset angle can change and the software can loose track of the actual position of the thruster. All bolts attaching the belt wheels to the servos have been tightened on the 1.july.2021. However servo nr.4 has loosened only 7 days later, causing it to loose the offset angle.
- The power is currently not connected to the power-switch, meaning that the boat will turn on as soon as a battery is connected.
- The fixture holding servo 3(front left) is not deep enough, making the screw holding the belt-wheel press against the bottom of the fixture causing the servo to stall. This problem is circumvented by installing two washers in-between the servo and the fixture.
- There are new constraints on maximum thrust. From the towing test carried out in Juny 2017, a bollard pull test was performed. It is suggested to implement this new maximum thrust values in the thrust allocation.
- The weight of the vessel is not correct, as it does not take the moon pool into account.

# Contents

# Part I

# Technical description

# Chapter 1

# Hardware

## 1.1 Introduction to CSAD

The CS Arctic Drillship was built and instrumented in 2016, with the intention of facilitating more research on Thruster-Assisted Position Mooring. For an in-depth description of the design and construction process, the reader is referred to Bjørnø (2016).

The vessel is a 1:90 scale model of the Statoil Cat I Arctic Drillship. It is equipped with 6 azimuth thruster (3 fore and 3 aft), in addition to a moon-pool for turret and mooring lines. The main dimensions of the vessel are:

Table 1.1: Main dimensions of CSE1

| | |
|---|---|
| LOA | 2.578[m] |
| B | 0.440 [m] |
| D | 0.211[m] |
| T | 0.133[m] |
| $\Delta$ | 127.92 [kg][1] |

### 1.1.1 Literature

The development of CSAD and its systems is a product of research from several theses, which contain complementary information on the theory applied to the system. The vessel has also been used in experiments for papers.

**Journals and conferences**

- Modeling, parameter identification and thruster-assisted position mooring of C/S Inoceacn CAT I Drillship (Bjørnø et al., 2017)
- Distributed motion sensing on ships (Heyn et al., 2017)

**Specialization projects and master theses**

- Thruster-Assisted Position Mooring of C/S Inocean Cat I Drillship (Bjørnø, 2016)
- Constrained Optimal Thrust Allocation for C/S Inocean Cat I Drillship (Frederich, 2016)
- Force Field Identification and Positioning Control of an Autonomous Vessel using Inertial Measurement Units (Udjus, 2017)
- Autonomous Heading Control in Position Mooring with Thrust Assist (Johannessen, 2017)

## 1.2 Actuators

The installed azimuth thrusters are of the type Aero-naut Precision Schottel, with 30 millimeter diameter propellers. They are positioned according to the design of the full-scale ship, as given in Figure 1.1 and Table 1.2.

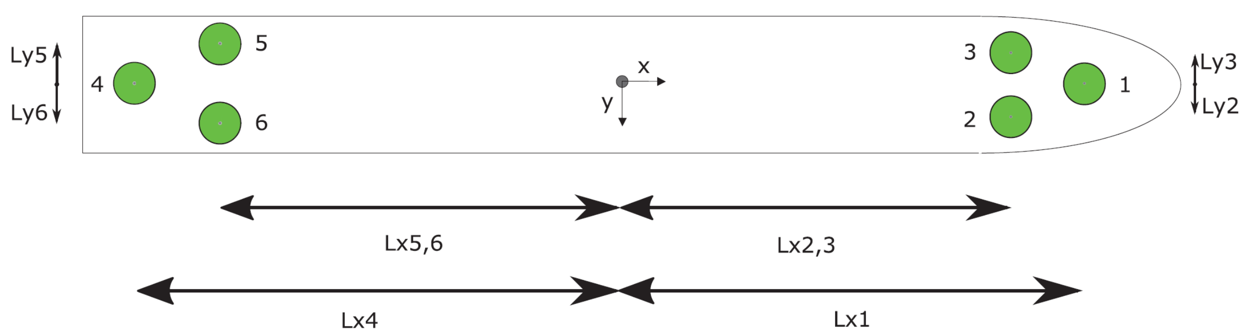



Figure 1.1: Illustration of thruster positions. Adapted from Frederich (2016)

In Frederich (2016), the thrust coefficients were estimated based on bollard pull tests. The values given in Table 1.3 are mean values from the tests, due to discrepancies in the bollard pull test at low thrust commands.

Table 1.2: Thruster positions

| **Thruster** | **Position X**[m] | **Position Y**[m] |
|---|---|---|
| Thruster 1 | 1.0678 | 0.0 |
| Thruster 2 | 0.9344 | 0.11 |
| Thruster 3 | 0.9344 | -0.11 |
| Thruster 4 | -1.1644 | 0.0 |
| Thruster 5 | -0.9911 | -0.1644 |
| Thruster 6 | -0.9911 | 0.1644 |

Table 1.3: Thruster coefficients

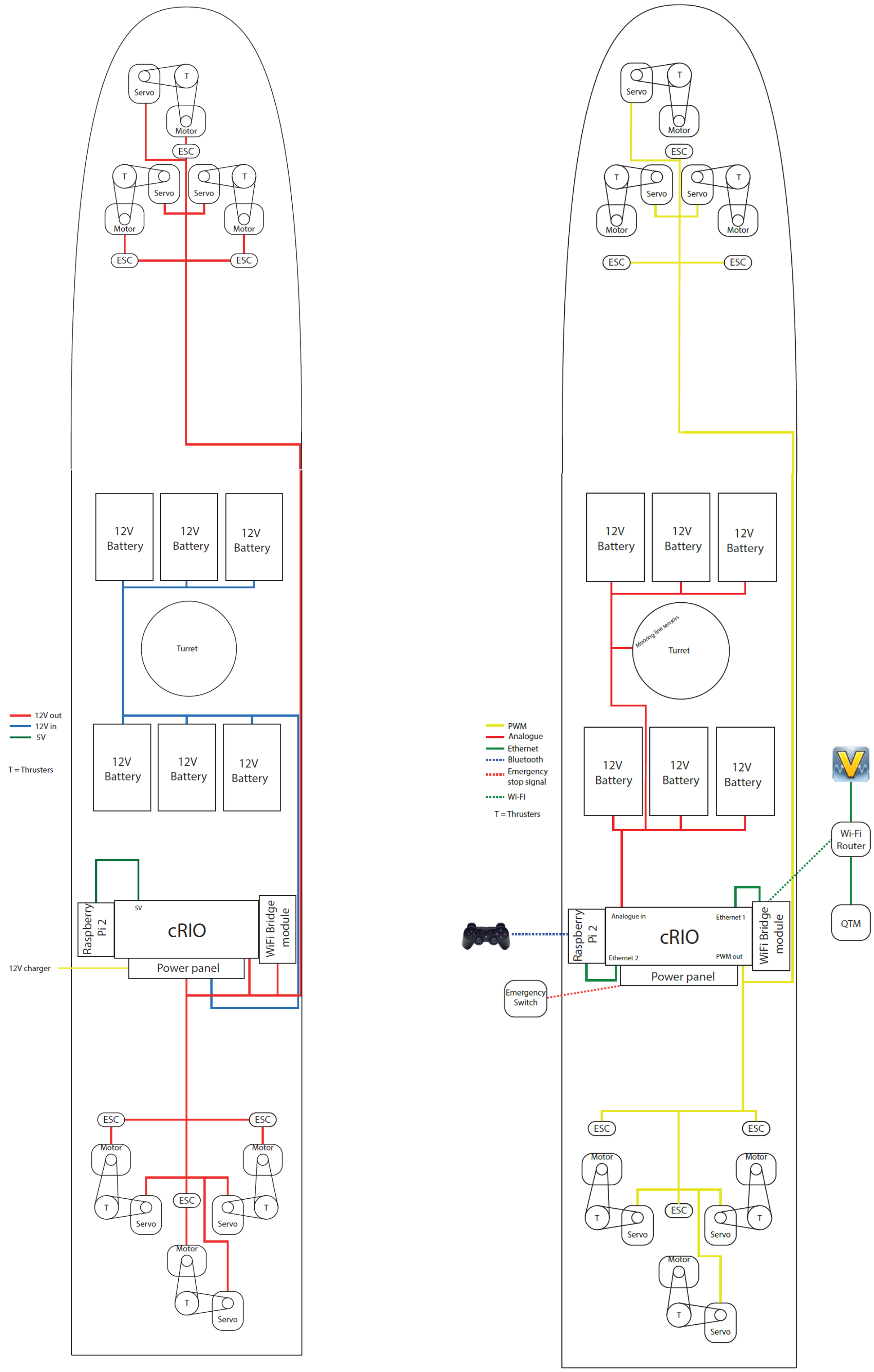
| | **Thruster 1** | **Thruster 2** | **Thruster 3** | **Thruster 4** | **Thruster 5** | **Thruster 6** |
|---|---|---|---|---|---|---|
| $K_T$ | 0.3763 | 0.3901 | 0.3776 | 0.5641 | 0.4799 | 0.5588 |
| $K_Q$ | 0.0113 | 0.0117 | 0.0113 | 0.0169 | 0.0144 | 0.0168 |

## 1.3 Power system

The vessel is powered through six 12V 12Ah batteries, connected in parallel. Figure 1.2a show a schematic drawing of the power system.

## 1.4 IMU

CSAD is equipped with one Inertial Measurement Unit (IMU) from Analog Devices. The sensor mounted on-board is the ADIS16364 and includes a triaxis gyroscope and triaxis accelerometer. The sensor has built-in compensation for bias, alignment and sensitivity, and provides accurate measurements over a temperature range of -10 to +70 degrees Celsius. The most relevant data is presented in Table 1.4, and for supplementary information the reader is referred to the data sheet Analog Devices (2017). The reference frame of the sensor is illustrated in Figure 1.3, with positive directions illustrated by arrows. As seen, the standard reference frame for linear accelerations uses left-hand orientation, while the angular rates uses right-hand orientation. It is advised to change the reference frame of accelerations to right-hand, which is achieved by multiplying the accelerations with -1. Note that this also changes the positive direction, defined as the direction of acceleration that produces a positive output.




(a) Power system (b) Control system

Figure 1.2: Illustration of power and control system. Figures adapted from Bjørnø (2016).

Table 1.4: IMU specifications

| | Parameter | Typical value | Unit |
|---|---|---|---|
| **Gyroscopes** | Dynamic range | $\pm 350$ | $^o$/sec |
| | Sensitivity | 0.0125 | $^o$/sec/LSB |
| | Bias stability, $\sigma$ | 0.007 | $^o$/sec |
| | Angular random walk | 2.0 | $^o/\sqrt{hr}$ |
| | Output noise | 0.8 | $^o$/sec rms |
| **Accelerometers** | Dynamic range | $\pm 5.25$ | g |
| | Sensitivity | 1.00 | mg/LSB |
| | Bias stability, $\sigma$ | 0.1 | mg |
| | Velocity random walk | 0.12 | m/sec/$\sqrt{hr}$ |
| | Output noise | 5 | mg rms |
| **Power supply** | Operating voltage | $5.0 \pm 0.25$ | V |

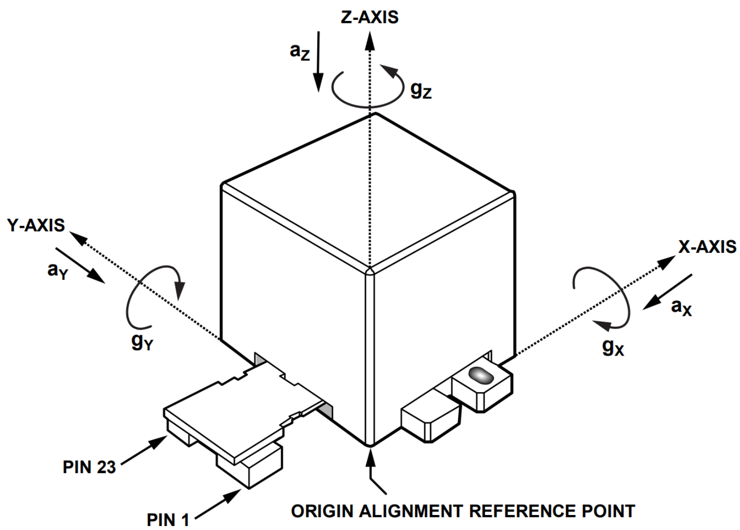



Figure 1.3: IMU reference frame from manufacturer

## 1.5 Control system

The on-board control system is illustrated in Figure 1.2b, and consists of the following parts:

- one Raspberry pi 4 model b
- six electronic speed controllers (ESC) connected to six motors controlling thruster speed
- six servos controlling thruster angle

For complementary information on the ESC, motors, servos and PWM signals, see Bjørnø (2016).

### 1.5.1 RPi

The Raspberry Pi is running ROS melodic master node at http://192.168.0.123:11311. Only the ds4_driver _node is launched at startup to communicate with the ds4 controller.

### 1.5.2 ESC and DC-motor

The ESC's are of the type O.S. OCA-150 50 A BL, connected to brushless UMA-2820-950 DC motors driving the thrusters. The ESC's are controlled with PWM signals. As the motors are much more powerful than desired for the model. The servos should not exceed 50% thrust in either direction.

### 1.5.3 Servo

The servos controlling the thruster angles are of the type Dynamixel MX-106R, and are geared 1:1 with the thrusters(1 degree turn on servo results in 1 degree on thruster). The servos are manually tuned to have a zero-angle offset in initial start. As of July 2021, the initial offsets are given in Table 1.5.

$$\boldsymbol{\alpha} \in [-10240°, 10240°] \tag{1.1}$$

Table 1.5: Servo angle offset(rad)

| Thruster | Offset |
|---|---|
| $\alpha_1$ | -0.52 |
| $\alpha_2$ | 1.475 |
| $\alpha_3$ | -1.568 |
| $\alpha_4$ | -1.319 |
| $\alpha_5$ | -0.02 |
| $\alpha_6$ | -1.156 |

The offsets can be changed in the csad_actuator_driver in src/scad_actuator_driver.cpp on line 36 in the servoOffsets variable.

# Chapter 2

# ROS

## 2.1 Introduction

The CSAD is setup to be controlled either directly through a ROS package or through messages sent to a ROS-node running on the ship.

## 2.2 CSAD_actuator_driver

csad_actuator_driver is a package that includes libraries to control CSAD with a couple of simple methods. csad_actuator_driver is dependent on the dynamixel_sdk ROS library to work. This means the dynamixel_sdk package must be in the same catkin_workspace in order to work.

**mapping**

All variables reading and setting the position of the servos are given in rads with 0rads meaning the thruster is pointing directly forward. All variables setting and reading the power of the motors are mapped from -1.0 to 1.0, with 1 being 100% forward and -1.0 being 100% backwards. As the motors are much more powerful than desired for the model the thrust is halved. This means the 1 is actually just 50% of the capability of the motors. Note that there is nothing in the code stopping anyone from sending 2 and setting the motor to 100% power.

### 2.2.1 using CSAD_Actuator_driver library

To use this library copy folders: csad_actuator_driver and dynamixel_sdk into the source folder of a catkin workspace. run catkin_make in the workspace to build the packages.

```
catkin_make
```

Include csad_actuator_driver in CMakeList.txt the package that is going to be controlling the ship.

```
find_package(catkin REQUIRED COMPONENTS
csad_actuator_driver
)
```

Include catkin_libraries in CMAKEList.txt normally found on line 148 of a standard ros package CMakeList.txt file.

```
Specify libraries to link a library or executable target against
target_link_libraries(${PROJECT_NAME}
${catkin_LIBRARIES}
)
```

Add csad_actuator_driver in the package.xml file

```
<depend>csad_actuator_driver</depend>
```

To use the library simply declare a ship object.

```
CSAD_Actuator ship;
```

And call the methods included in the library.

**Methods**

All methods can be called from any CSAD_Actuator object

- float getServoPresentPosition(uint8_t servoNumber);
  - This return the current servo positions of the servo with the number given in the sevoNumber variable.
- void getAllServoPresentPositions(double positions[]);
  - This returns all servo positions in the variable positions where the first element corresponds to the servo with the ids specified in the first element of the servoIds variable in csad_actuator.cpp
- void setServoPosition(double position, uint8_t servoNumber);
  - This sets the goal-position of the servo specified in the servoNumber variable, to the position specified in the position variable.
- void setAllServoPositions(double positions[NUMBER_OF_SERVOS]);
  - This sets the goal-positions of all servos to the positions specified in the positions variable, where the first element of the positions array corresponds to the goal-positions of the servo with the id matching the first element of the servosIds array in scad_actuator.cpp
  - The number of elements in position needs to be 6.
- void setMotorPower(uint8_t motor, double power);
  - This function sets the motor power of the motor specified in the motor variable, tp the value set in the power variable. 1.0 being 100
- void setAllMotorPower(double power[NUMBER_OF_SERVOS]);
  - Sets the power of all motors the the values specified in the Power array variable, 1.0 being 100% forward, and -1.0 being 100% in reverse.

- void closeI2CPort();
  - This closes the I2C port in the raspberry pi, and stops communication with the PCA9685 PWM-module.
- void resetPCAModule();
  - Resets the PCA9685 PWM-module.

### 2.2.2 controlling CSAD through nodes

In order to control CSAD through nodes we have to run CSAD_node

```
rosrun csad_node CSAD_node
```

Then we can controll the servos and motors by publishing a std_msgs::Float64MultiArray to topic:"CSAD/u". The first six elements corresponds to Thruster power (1 being 100% and -1 being reverse 100%) and the last 6 elements corresponds to the angle of the thrusters( -pi being -180deg and pi being 180deg)

# Chapter 3

# Modeling

## 3.1 3 DOF Maneuvering model

Here, a 3DOF model of the vessel is presented. For complementary information on the procedure and accuracy, the reader is referred to Bjørnø (2016). The 3DOF model is based on system identification from towing tests in MC-Lab. The model is valid for low speed, and it is noteworthy that the presented model does not include cross-coupled damping terms. The model is based on the standard 3DOF maneuvering model from Fossen (2011):

$$\dot{\boldsymbol{\eta}} = \mathbf{R}(\psi)\boldsymbol{\nu} \tag{3.1}$$

$$\mathbf{M}\dot{\boldsymbol{\nu}} + \mathbf{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \mathbf{D}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau}_{env} + \boldsymbol{\tau}_{thruster} \tag{3.2}$$

where $\boldsymbol{\eta} = [x, y, \psi]^T \in \mathbb{R}^3$, $\boldsymbol{\nu} = [u, v, r]^T \in \mathbb{R}^3$ and $\boldsymbol{\tau} = [X, Y, N] \in \mathbb{R}^3$. The matrices are

$$\mathbf{R}(\psi) = \begin{bmatrix} cos(\psi) & -sin(\psi) & 0 \\ sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.3}$$

$$\mathbf{M} = \mathbf{M}_{RB} + \mathbf{M}_A = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 \\ 0 & m - Y_{\dot{v}} & mx_g - Y_{\dot{r}} \\ 0 & mx_g - Y_{\dot{r}} & I_z - N_{\dot{r}} \end{bmatrix} = \mathbf{M}^T > 0 \tag{3.4}$$

$$\mathbf{C}(\boldsymbol{\nu}) = \mathbf{C}_{RB}(\boldsymbol{\nu}) + \mathbf{C}_A(\boldsymbol{\nu}) = \begin{bmatrix} 0 & 0 & (-mx_g + Y_{\dot{r}})r + (-m + Y_{\dot{v}})v \\ 0 & 0 & (m - X_{\dot{u}})u \\ (mx_g - Y_{\dot{r}})r + (m - Y_{\dot{v}})v & (-m + X_{\dot{u}})u & 0 \end{bmatrix} \tag{3.5}$$

$$\mathbf{D}(\boldsymbol{\nu}) = \mathbf{D} + \mathbf{D}(\boldsymbol{\nu}) = -\begin{bmatrix} d_{11}(u) & 0 & 0 \\ 0 & d_{22}(v) & d_{23}(r) \\ 0 & d_{32}(v) & d_{33}(r) \end{bmatrix} \tag{3.6}$$

The damping coefficients are

$$d_{11}(u) = X_u + X_{|u|u}|u| + X_{uuu}u^2 \tag{3.7}$$

$$d_{22}(v,r) = Y_v + Y_{|v|v}|v| + Y_{vvv}v^2 \tag{3.8}$$

$$d_{23}(v,r) = Y_r + Y_{|r|r}|r| + Y_{rrr}r^2 \tag{3.9}$$

$$d_{32}(v,r) = N_v + N_{|v|v}|v| + N_{vvv}v^2 \tag{3.10}$$

$$d_{33}(v,r) = N_r + N_{|r|r}|r| + N_{rrr}r^2 \tag{3.11}$$

The rigid body and added mass parameters are given in Table 3.1, and the drag coefficients are given in Table 3.2.

Table 3.1: Rigid body and added mass parameters

| Rigid body | |
|---|---|
| Parameter | Value |
| $m$ | 127.92 |
| $I_z$ | 61.967 |
| $x_g$ | 0 |

| Added mass | |
|---|---|
| Parameter | Value |
| $X_{\dot{u}}$ | 3.262 |
| $Y_{\dot{v}}$ | 28.89 |
| $Y_{\dot{r}}$ | 0.525 |
| $N_{\dot{v}}$ | 0.157 |
| $N_{\dot{r}}$ | 13.98 |

Table 3.2: Drag coefficients in surge, sway and yaw

| Surge | |
|---|---|
| Parameter | Value |
| $X_u$ | -2.332 |
| $X_{\|u\|u}$ | 0 |
| $X_{uuu}$ | -8.557 |

| Sway | |
|---|---|
| Parameter | Value |
| $Y_v$ | -4.673 |
| $Y_{\|v\|v}$ | 0.3976 |
| $Y_{vvv}$ | 313.3 |

| Yaw | |
|---|---|
| Parameter | Value |
| $N_r$ | -0.01675 |
| $N_{\|r\|r}$ | -0.01148 |
| $N_{rrr}$ | 0.0003578 |

# Part II

# User Manual

It is assumed that the reader has studied the MCLab Handbook before using CSAD, and has knowledge about Lab equipment, procedures and Safety precautions. In addition, the following is important to keep in mind when using CSAD:

**Water damage:** CSAD is watertight when the hatches on the top are closed properly.

**Propeller dry running:** The thruster gears are lubricated with water, and thus the propellers always has to be in the water when running. Hence, always keep the vessel in the basin when the power is connected.

**Total loss of control:** Pull CSAD with a boat hook, and keep the vessel in water while disconnecting batteries.

**Launching** CSAD is a large model, and care must be taken when launching the vessel to the basin. Always be two persons, and make sure the vessel does not hit the basin wall when launching or removing it from the basin. When launching, remove all weights(batteries and ballast). Still, the vessel is heavy, and the lifting up and down to the basin might be harder than expected.

# Chapter 4

# Launching

## 4.1 Vessel and lab preparations

Follow these vessel-specific instructions when preparing for experiments with CSAD:

1. Make sure all batteries and weights are removed from the vessel when lifting it. Launch the vessel in the basin.

2. Place all 6 batteries(12V 12Ah, marked CSAD) in the vessel, at their dedicated places. 3 in front of the moonpool, 3 behind it. Connect the batteries, positive/7red first then negative/black.

3. Place the ballast weights. 20 kg in the aft, and 27.5 in the front. Manually adjust their position, such that the vessel does not have any heel (slagside). Check with the design draft indicated on the outside of the hull.

4. run /launch necessary rosnodes for controlling the ship.( The setMotorPower method should be called at least once with 0 as power for each motor.) All motors should beep one note followed by three rising notes, indicating that they are ready to control the motors.

5. Press the PS-button on the ps4 controller and wait until the indicator light on the controller stops blinking and turns a solid purple colour.

6. Place the vessel inside the region of sight for Qualisys (check on the Qualisys computer that all 4 reflectors are visible for all cameras). Align the vessel with 0°heading in the basin frame, i.e. with the bow pointing towards the command center.

7. On the Qualisys computer, aqcuire the body. This process is described in MC-Lab Handbook, with information on debugging. In the body frame, the highest marker has position $(x, y, z) = (960, -190, -575)[mm]$.

8. Go to 3D Visualization in QTM, and verify that the body is correct. The body x-axis should be parallel to the lines between the markers on starboard and port side.

CSAD and the lab is now set up for experiments.

# Chapter 5

# Demolition

When the experiments are finished, follow the procedure given here to shut down.

1. Navigate CSAD near the basin wall
2. Terminate all rosnodes
3. Turn of the power switch, and disconnect the batteries
4. Remove the ballast weights
5. Remove the batteries from the vessel
6. Lift CSAD from the basin, and put it in its rack in the storage
7. Leave the ds4 controller in the vessel
8. On the Qualisys computer, quit Qualisys Track Manager
9. If you recorded any videos with the Camera System, export these videos to a memory stick, quit the software and turn of the TV-monitor
10. Do a general clean up, bring all your personal belongings with you when you leave

NOTE on charging the batteries: When charging the batteries in CSAD, the WiFi bridge must be disconnected. Unplug the power wire from the WiFi bridge(connected on the side of the Ethernet cable). All batteries must be placed in the vessel when charing, and connected, but with the power switch turned off. Connect the charger to the charging wire located in the aft hatch. The charger is located in the shelf in the storage, marked CSAD. Set the charging mode to the motorcycle symbol.

# Appendix A

# Extended IMU system

# Bibliography


Analog Devices, 2017. ADIS16364 Data Sheet. Technical Report. URL: http://www.analog.com/media/en/technical-documentation/data-sheets/ADIS16364.pdf.

Bjørnø, J., 2016. Thruster-Assisted Position Mooring of C/S Inocean Cat I Drillship. Master's thesis. NTNU.

Bjørnø, J., Heyn, H.M., Skjetne, R., Dahl, A., Frederich, P., 2017. Modeling, parameter identification and thruster-assisted position mooring of c/s inoceacn cat i drillship. Offshore & Arctic Engineering .

Fossen, T.I., 2011. Handbook of marine craft hydrodynamics and motion control. John Wiley & Sons.

Frederich, P., 2016. Constrained Optimal Thrust Allocation for C/S Inocean Cat I Drillship. Master's thesis. NTNU.

Heyn, H.M., Udjus, G., Skjetne, R., 2017. Distributed motion sensing on ships. IEEE Oceans .

Johannessen, S.A., 2017. Autonomous heading control in position mooring with thrust assist. Master's thesis. NTNU.

Udjus, G., 2017. Force Field Identification and Positioning Control of an Autonomous Vessel using Inertial Measurement Units. Master's thesis. NTNU.