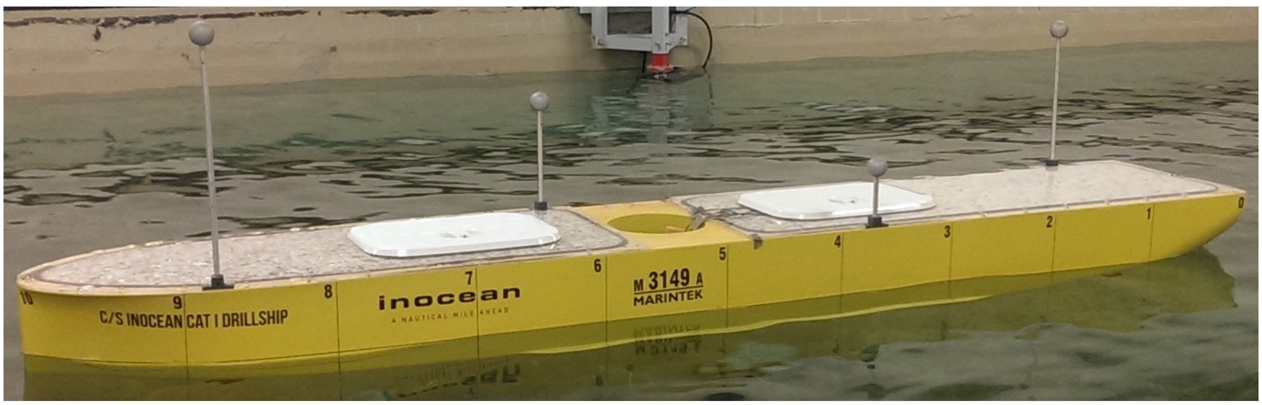
CyberShip Arctic Drillship

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 MCLab Handbook, which can be found on GitHub: https://github.com/NTNU-MCS/MC_Lab_Handbook

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
IP-address(port 1)	192.168.0.55
IP-address(port 2)	192.168.1.21
RPi IP-address	192.168.1.33
RPi Port Number	51717
Qualisys body	(960, -190, -575) [mm]
MATLAB Version	2016b
LabVIEW Version	2017
VeriStand Version	2017

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]
Δ	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

- 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)

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.

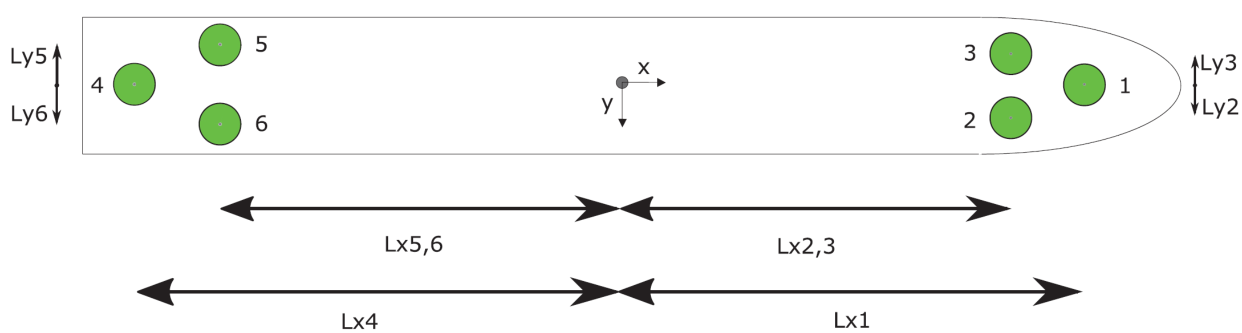


Figure 1.1: Illustration of thruster positions. Adapted from Frederich (2016)

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

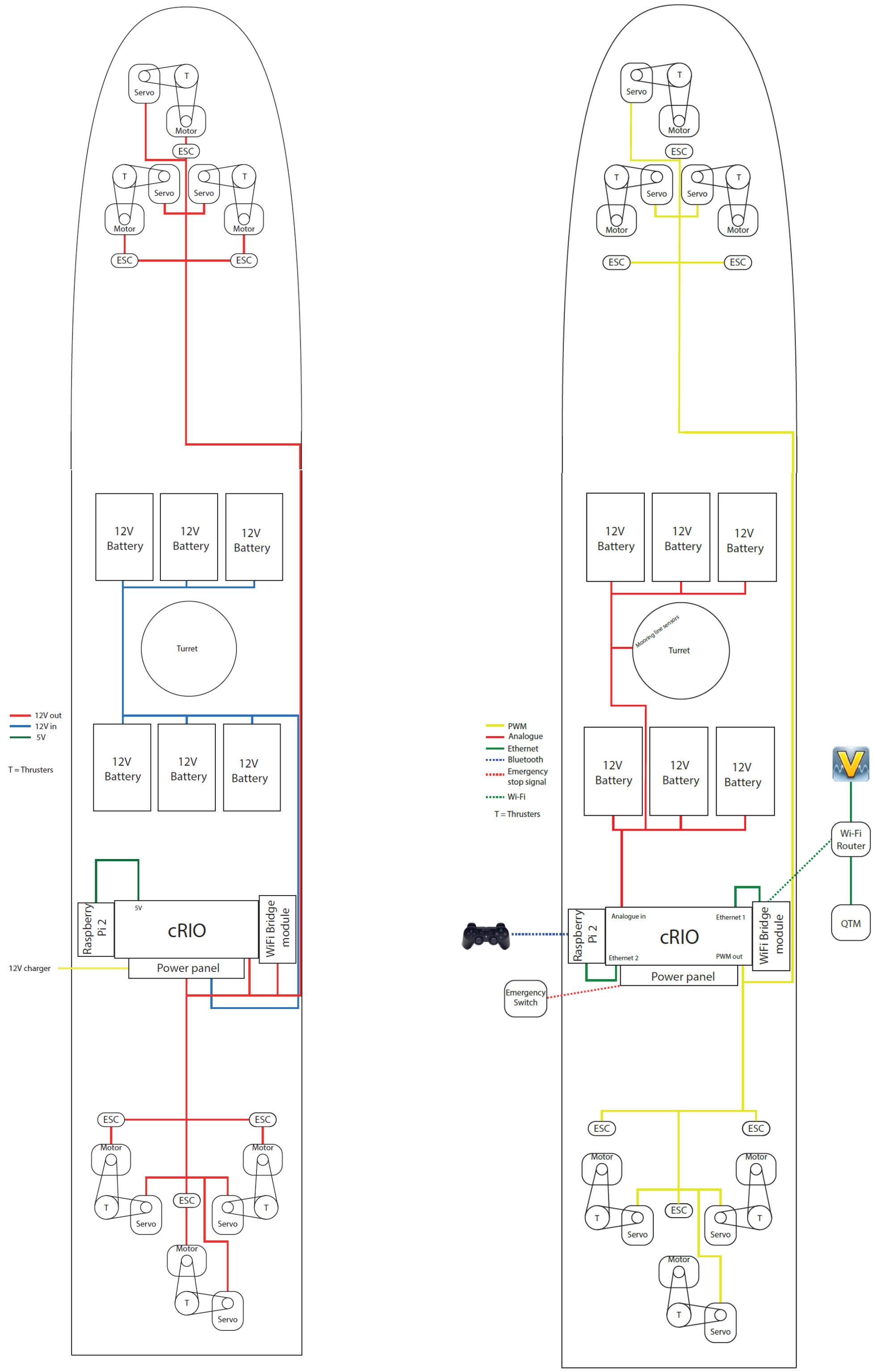
	Thruster 1	**Thruster 2**	**Thruster 3**	**Thruster 4**	**Thruster 5**	**Thruster 6**
K_T	0.3763	0.3901	0.3776	0.5641	0.4799	0.5588
K_Q	0.0113	0.0117	0.0113	0.0169	0.0144	0.0168

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	± 350	^{o}/sec
	Sensitivity	0.0125	^{o}/sec/LSB
	Bias stability, σ	0.007	^{o}/sec
	Angular random walk	2.0	$^o/\sqrt{hr}$
	Output noise	0.8	^{o}/sec rms
Accelerometers	Dynamic range	± 5.25	g
	Sensitivity	1.00	mg/LSB
	Bias stability, σ	0.1	mg
	Velocity random walk	0.12	m/sec/$\sqrt{hr}$
	Output noise	5	mg rms
Power supply	Operating voltage	5.0 ± 0.25	V

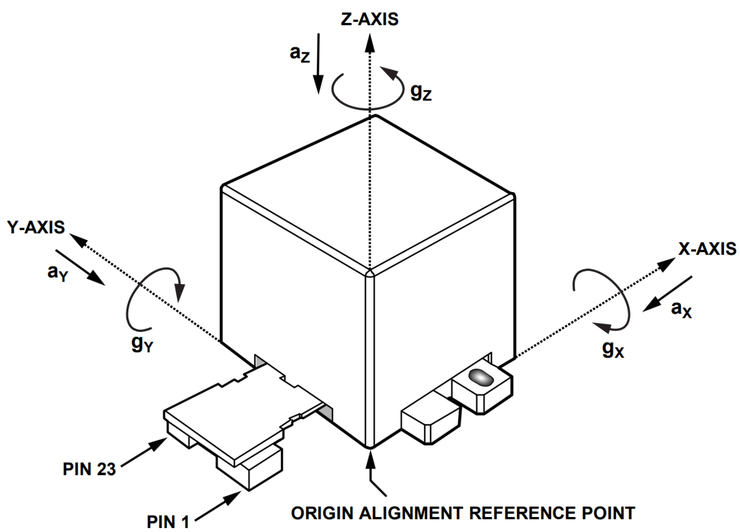


Figure 1.3: IMU reference frame from manufacturer

1.5 Control system

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

- one National Instruments compact reconfigurable input/output (cRIO) embedded controller
- one Raspberry Pi (RPi) single-board computer
- 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 cRIO

The model on-board is the cRIO-9024, and it is connected to 4 FPGA modules for analog and digital I/O:

- One NI-9215, Analog input
- Two NI-0237, Analog bridge
- Two NI-9401, Bidirectional digital input
- One NI-9411, Digital input
- One NI-9474, Digital output
- One NI-9871, Serial interface

1.5.2 RPi

The Raspberry Pi provides communication with the Siaxaxis controller, as described in Section 1.6. It works as an embedded system, and once powered it will start connecting to the wireless controller. When connection is established, it continuously sends the Sixaxis controller output to the cRIO over Ethernet.

If there are problems establishing connection between Sixaxis and RPi, try restarting the RPi. If this doesn't work, contact Torgeir Wahl or see the MCLab Handbook on Github for description on how to reconfigure the RPi.

1.5.3 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 PWM signal is constrained. Table 1.5 gives the specification on the PWM signal for the ESC connected to the propellers, and Table 1.6 gives the constraints on the PWM signal.

Verify the PWM con-

Table 1.5: PWM specification for ESC

Initial value	**Scaling**	**Offset**	**PWM period** [Ticks]
0	100	0	800.000

Table 1.6: PWM constraints

Direction	**Signal range PWM**	Tick
Full reverse	6.25 [%]	50.000
Neutral	7.5 [%]	60.000
Full forward	8.75 [%]	70.000

1.5.4 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 2017, the initial offsets are given in Table 1.7. Maximum and minimum angles for all servos are given in (1.1). If one thruster reaches maximum/minimum angle, it is reset to initial neutral position, as described in the software section. For complementary information.

update min/max values on PWM in init file

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

Table 1.7: Servo angle offset

Thruster	**Offset**
α_1	-149°
α_2	50°
α_3	5°
α_4	-17°
α_5	127°
α_6	-36°

1.6 High-level communication

Following Figure 1.4 from left to right:

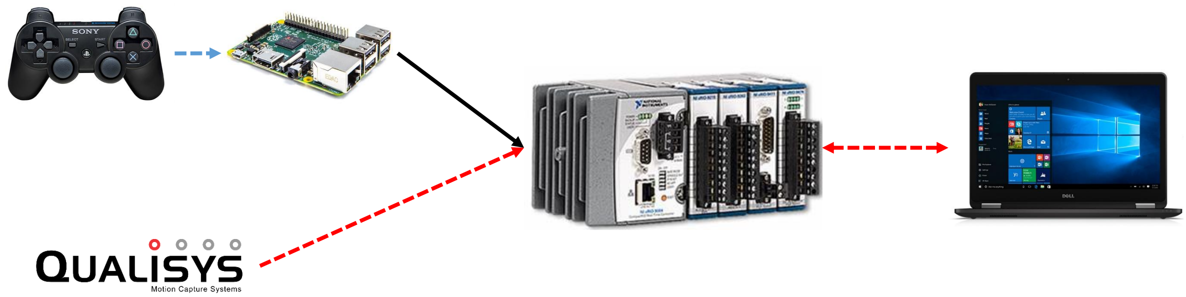


Figure 1.4: CSE1 communication diagram

Sixaxis transmits its Joystick information to the RPi over Bluetooth communication.

RPi receives Sixaxis data through the USB dongle and forwards it using Ethernet connection (TCP).

cRIO reads QTM broadcast positioning data through the Wi-Fi bridge on Ethernet port 1, Sixaxis data on Ethernet port 2. Online data and laptop input is transmitted and received on Ethernet port 1 by the VeriStand Engine.

Laptop reads simulation data and sends input to the cRIO over MC Lab Wi-Fi.

Hence, there are two possible methods for controlling the vessel:

- laptop connected to the MC Lab wireless network
- Sixaxis wireless gamepad(labeled *CSAD*)

Chapter 2

Software

2.1 Introduction

In order to control CSAD, there are several software parts that runs. This chapter gives a description of the software topology on CSAD. Note that the software is ready to use, and alterations in the software described here is not necessary(except modifying *ctrl_custom*, see Part II).

2.2 Control system

Figure 2.1 illustrate the software architecture, and gives an overview on how the different modules and I/O from the Simulink models are connected in VeriStand. In general, the software can be divided into 2 groups:

MATLAB generated parts: ctrl_custom, ctrl_TAPM, ctrl_sixaxis2thruster, STOP and u2pwm

LabVIEW generated parts: IMU, Oqus, WL_Joystick and FPGA

All of these modules are described in the latter.

2.2.1 ctrl_custom

This is the only reconfigurable software, and does not consist of any control system. In Part II a description on how to configure and upload the code to CSAD is given.

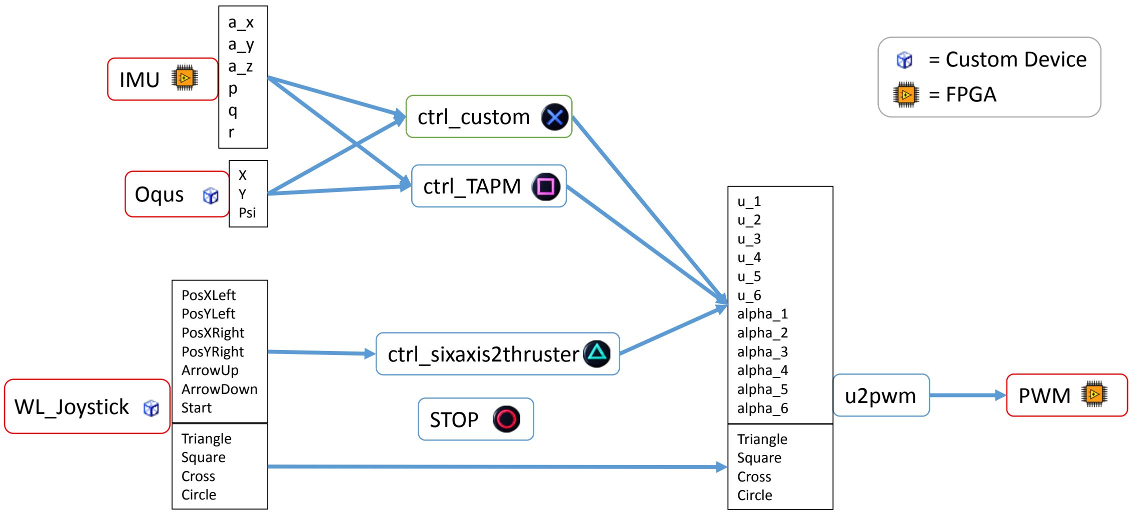


Figure 2.1: CSE1 control software

2.2.2 ctrl_TAPM

This code is a basic TAPM control system, intended for demonstrations of the vessel. The desired position of the vessel is modified in VeriStand.

2.2.3 ctrl_sixaxis2thruster

All 6 thruster can be controlled manually using the Sixaxis controller. The right joystick controls the direction of the three thrusters aft, while the left joystick controls the front thrusters. ArrowUp and ArrowDown sets the thrust limits, and Start resets the limitation to 0.

2.2.4 STOP

This system simply stops the vessel, by setting the thrust of all thrusters to 0.

2.2.5 u2pwm

This code transforms the control input to PWM signals, which are sent to the FPGA module. There are 2 groups of inputs, namely the control signal and a switch signal. The switch signal is used to switch between the 4 control systems described previously. Switching is simply achieved by pressing either one of the four symbols, and the mapping between the buttons and models is shown in Figure 2.1. The code is not supposed to be altered, and should work as it is. Figure 2.2 illustrate how the code works, and the different parts are described in the latter.

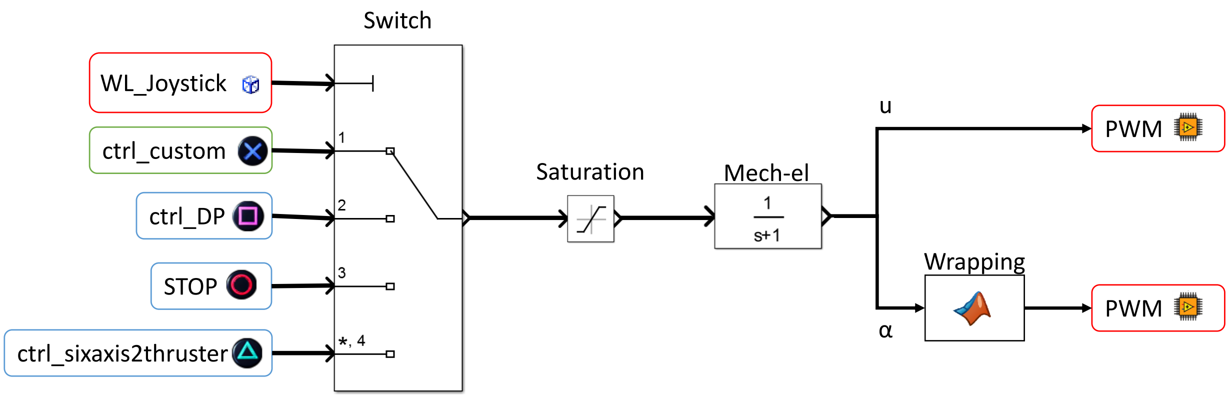


Figure 2.2: Schematic of u2pwm

Switch

The switch simply forward the input from the desired model, as given by the switch signal:

- ctrl_sixaxis2thruster when △ is pushed
- ctrl_sixaxis2direction when □ is pushed
- ctrl_DP when ○ is pushed
- ctrl_student when × is pushed

Saturation

The inputs are saturated to avoid overload on the thrusters. All inputs are saturated as given in Table 2.1.

Table 2.1: Saturation of input

min	Variable	max
$-0.5 \leq$	u	≤ 0.5

Mecl-el

This block simulates an mechanical-electrical system, by use of a transfer function as. The transfer function is equal for all variables, and the time constant is set to 1. By default, this block is not active, and must be activated in the Workspace. There is a NI VeriStand Input, named "mech el enable", which activates the block. Using this block will make the vessel behave more realistic.

Wrapping

This block simply calculates the shortest distance for the thrusters, when they have a new desired angle. As the thruster can rotate both clock-wise and counter-clock wise, the wrapping block makes sure the thrusters rotate in the shortest direction.

2.2.6 Oqus

The Qualisys Track Manager software broadcast the position data of the vessel over MC Lab Wifi. Reading these data is done on the cRIO through a Custom Device module named Oqus. The Oqus software simply listen to the network for data from QTM, and once it receives data it forward the position and orientation to the models as given in Figure 2.1. The Custom Device is programmed by Torgeir Wahl, and can be found on GitHub.

2.2.7 WL_Joystick

This is the second Custom Device that runs on the cRIO, which listen to the Ethernet 2 port for input from the RPi. It forwards this data to the respective models as given in Figure 2.1. The software is designed by Torgeir Wahl, and can be found on GitHub.

2.2.8 FPGA

For CSAD, 8 FPGA modules are in use. The FPGA software is described in the MC Lab Software Handbook, and provides a guide on how to create an FPGA module. The modules can be found on GitHub, but are as standard. The PWM signal is mapped as described in Table 2.2.

Table 2.2: PWM mapping

u2pwm	FPGA PWM
pwm_u_1	PWM Out 0
pwm_u_2	PWM Out 1
pwm_u_3	PWM Out 2
pwm_u_4	PWM Out 3
pwm_u_5	PWM Out 4
pwm_u_6	PWM Out 5

u2pwm	FPGA Thruster
alpha_1	Angle0
alpha_2	Angle1
alpha_3	Angle2
alpha_4	Angle3
alpha_5	Angle4
alpha_6	Angle5

2.3 Connecting software

All the different software parts described in the previous Section are connected together in VeriStand. On CSAD, VeriStand 2017 is used. In the system definition file *CSAD.nivssddf*, all necessary mappings of variables are done. In addition, here the different Custom Devices, FPGA code and Simulink Models can be included. However, the standard setup should not be altered, as all necessary code and mappings is already taken care of. For description on how to implement the modified *ctrl_custom* Simulink model, the reader is referred to Part II.

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
$X_{\vert u\vert u}$	0
X_{uuu}	-8.557

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

Yaw	
Parameter	Value
N_r	-0.01675
$N_{\vert r\vert r}$	-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.

Loss of laptop control: Wireless network instability may result in loss of connection between the laptop user interface and the cRIO. In this event, fall back to manual thruster control, by pushing △ on the Sixaxis, or alternatively press ○ to stop the vessel.

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 Update customized simulink code

In the MC-Lab Handbook, a description on how to compile and upload the customized Simulink code is given. Here is only the specific application for CSAD given. All necessary files to control CSAD is found on GitHub: https://github.com/NTNU-MCS/CS_Drillship_cRIO. Download the folder with all files, and follow the instructions in MC-Lab Handbook for implementing your control system in the VeriStand project.

After successfully implementing the customized simulink code in the VeriStand project, continue to the next Section.

4.2 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. Turn on the power switch. This is located on the backside of the plastic box containing the cRIO. It might be a bit hard to find at first, use your left hand and search with your fingers around the centerline.

5. Verify that the WiFi-bridge connects, the blue light is continuous.

6. Once the bluetooth dongle connected to the RPi starts blinking(frequency of 1 Hz), press the PS-button on the Sixaxis. On the controller, indicator 1 lights continuously red when successfully connected.

7. 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.

8. 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 $(x, y, z) = (960, -190, -575)[mm]$.

9. 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.

4.3 Upload VeriStand project to the vessel

With the hardware prepared, follow these instructions to upload the VeriStand project to the cRIO onboard CSAD.

1. Make sure the computer is connected to the MC-Lab network(either by Ethernet cable or the WiFi).

2. Check the communication between the laptop and CSAD. Open Command Prompt(*cmd.exe*), write the following: `ping 192.168.0.55`. The response time varies, but it is crucial that the command returns 0% loss.

3. Open the VeriStand project (*CSAD.nivsproj*). If the project has been updated as described in the MC-Lab Handbook, press the deploy button (see Figure 4.1). The code is then uploading to the cRIO onboard the vessel. If the deployment is not successful, verify that the sixaxis controller is connected to the RPi and that the body is tracked in Qualisys. Try deploying again. If it still does not work, try restarting the vessel (either resetting the power switch in the vessel, or restart the cRIO in NI MAX). Also check that the Qualisys computer is connected to the same network(`ping 192.168.0.10`).

4. When successfully deployed project, open the Workspace (*CSAD.nivsscreen*) as shown in Figure 4.1. The code is now running on the cRIO, continue to the next Chapter for instruction on operating the vessel.

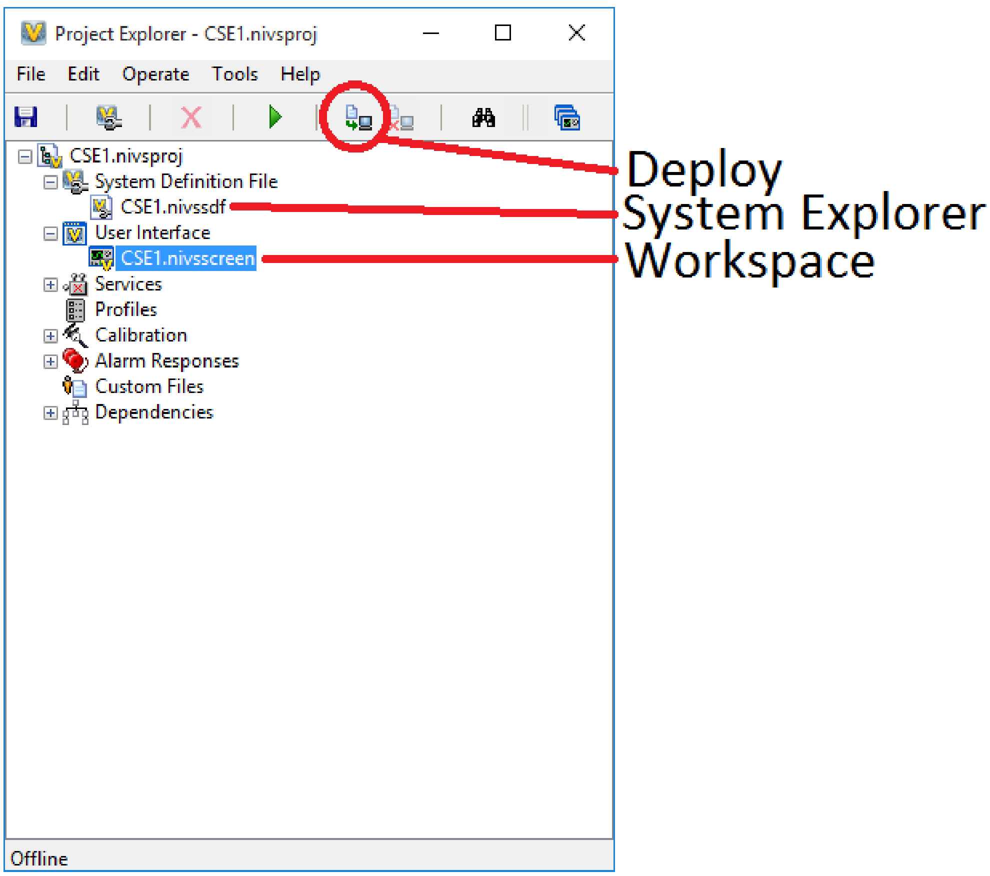


Figure 4.1: User interface in VeriStand Project Explorer

Chapter 5

Operating

After successful deployment of the VeriStand project, you control the vessel with the sixaxis-controller and/or the laptop. The sixaxis controller works as the main controller, and thus always have the sixaxis controller at hand in case of error in your control system. Use the sixaxis controller to switch between the different operation modes:

- △ - ctrl_sixaxis2thruster
- □ - ctrl_custom
- × - ctrl_DP
- ○ - STOP

5.1 Workspace

On the laptop, use the Workspace to monitor and control the different parameters in the Simulation models. There are 1 screen for each operation mode. You should not alter the screen for ctrl_sixaxis2thruster or ctrl_DP, while the ctrl_custom can be modified as desired. How to use and modify the Workspace is described in the MC-Lab Handbook.

Data logging can be done in two ways, as described in MC-Lab Handbook.

Chapter 6

Demolition

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

1. Switch to *ctrl_sixaxis2thruster*, and navigate CSAD near the basin wall
2. In the Project Explorer window, press to undeploy the code
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 sixaxis 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. Tech. rep. 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.

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.

Udjus, G., 2017. Force field identification and positioning control of an autonomous vessel using inertial measurement units. Master's thesis, NTNU.