

Software Manual

Underwater Robotics Research Centre (CIRS)
Computer Vision and Robotics Institute
Universitat de Girona, 17003, Girona, Spain.
web: <http://cirs.udg.edu>

Contact person: Marc Carreras
email: marc.carreras@udg.edu

January 2, 2014

Contents

1	Installation	5
1.1	Install Ubuntu	5
1.2	Install ROS Groovy	5
1.3	Install UWSim	5
1.4	Install COLA2 architecture	6
2	Getting started	6
2.1	Learn ROS	6
2.2	Learn UWSim	6
2.3	Learn Sparus II COLA2 architecture	6
3	Nodes	8
3.1	Node /captain	8
3.2	Node /dynamics	9
3.3	Node /ekf_slam	10
3.4	Node /keyboard	12
3.5	Node /map_ack	12
3.6	Node /merge_body_force_req	13
3.7	Node /merge_body_velocity_req	13
3.8	Node /merge_world_waypoint_req	14
3.9	Node /navigator_s2	15
3.10	Node /pilot	17
3.11	Node /pose_controller_s2	18
3.12	Node /safe_depth_altitude	19
3.13	Node /safety_s2	19
3.14	Node /set_zero_pose	20
3.15	Node /set_zero_velocity	20
3.16	Node /sim_actuators_s2	21
3.17	Node /sim_nav_sensors_s2	22
3.18	Node /teleoperation	23
3.19	Node /thruster_allocator	24
3.20	Node /velocity_controller_s2	25
4	Messages	26
4.1	Message file actionlib_msgs/GoalID	26
4.2	Message file actionlib_msgs/GoalStatusArray	26
4.3	Message file auv_msgs/BodyForceReq	27
4.4	Message file auv_msgs/BodyVelocityReq	27
4.5	Message file auv_msgs/NavSts	28
4.6	Message file auv_msgs/WorldWaypointReq	29
4.7	Message file cola2_control/Setpoints	30

4.8	Message file cola2_control/WorldWaypointReqActionFeedback	30
4.9	Message file cola2_control/WorldWaypointReqActionGoal	31
4.10	Message file cola2_control/WorldWaypointReqActionResult	31
4.11	Message file cola2_navigation/FastraxIt500Gps	32
4.12	Message file cola2_navigation/LinkquestDvl	32
4.13	Message file cola2_navigation/PressureSensor	33
4.14	Message file cola2_safety/MissionStatus	33
4.15	Message file cola2_safety/NavSensorsStatus	33
4.16	Message file cola2_safety/TotalTime	34
4.17	Message file diagnostic_msgs/DiagnosticArray	34
4.18	Message file geometry_msgs/PoseWithCovarianceStamped	35
4.19	Message file geometry_msgs/TwistWithCovarianceStamped	35
4.20	Message file nav_msgs/Odometry	36
4.21	Message file nav_msgs/Path	36
4.22	Message file pose_ekf_slam/Map	37
4.23	Message file sensor_msgs/Imu	37
4.24	Message file sensor_msgs/Joy	38
4.25	Message file sensor_msgs/Range	38
4.26	Message file std_msgs/String	39
4.27	Message file tf/tfMessage	39
4.28	Message file visualization_msgs/MarkerArray	40
4.29	Message file visualization_msgs/Marker	41
5	Services	42
5.1	Service file cola2_control/Goto	42
5.2	Service file cola2_control/String	42
5.3	Service file cola2_navigation/SetGPS	42
5.4	Service file cola2_navigation/SetNedOrigin	43
5.5	Service file cola2_safety/Submerge	43
5.6	Service file pose_ekf_slam/SetLandmark	43
5.7	Service file pose_ekf_slam/SetPosition	44
5.8	Service file std_srvs/Empty	44
5.9	Service file tf/FrameGraph	45
6	Config files	46
6.1	Config file /cola2_control/config/map_ack_s2.yaml	46
6.2	Config file /cola2_control/config/merge_s2.yaml	48
6.3	Config file /cola2_control/config/pilot.yaml	48
6.4	Config file /cola2_control/config/pose_controller_s2.yaml	49
6.5	Config file /cola2_control/config/teleoperation_s2.yaml	49
6.6	Config file /cola2_control/config/thruster_allocator_s2.yaml	50
6.7	Config file /cola2_control/config/velocity_controller_s2.yaml	50
6.8	Config file /cola2_navigation/config/navigator_s2.yaml	50

6.9	Config file /cola2_safety/config/safety_s2.yaml	51
6.10	Config file /cola2_sim/config/dynamics_s2.yaml	52
6.11	Config file /cola2_sim/config/sim_nav_sensors_s2.yaml	52

1 Installation

Sparus II AUV computer works under Ubuntu Server 12.04.3 LTS and ROS Groovy. In order to work with Sparus II robots it is highly recommended to set up another machine with Ubuntu Desktop, ROS, COLA2¹ and UWSim² for teleoperation, programming and simulation purposes. This section reports the required steps to set up a machine with Ubuntu, ROS, UWSim and Sparus II architecture COLA2.

1.1 Install Ubuntu

The first step to set up a machine is to instal Ubuntu Desktop 12.04.3 LTS. It is important the you leave some free space (e.g. at least 10 Gb) on the root to install the other required software later. You can download the image at <http://www.ubuntu.com/download/desktop>.

1.2 Install ROS Groovy

Once Ubuntu is running, the next step is to install ROS Groovy. Use the guide in the ROS webpage <http://wiki.ros.org/groovy/Installation/Ubuntu>. Install the Desktop-Full version to make sure that most of the required packages are installed.

Once the installation is done, open a terminal and install the following packages:

```
sudo apt-get install python-rosinstall python-rosdep
sudo apt-get install ros-groovy-joystick-drivers
```

Install Synaptic Package Manager. Search for libdc1394 and install libdc1394-22 and libdc1394-22-dev packages. Install ros-groovy-python-orocos-kdl too.

Once all packages have been installed, set up the ROS environment following: <http://wiki.ros.org/ROS/Tutorials/InstallingandConfiguringROSEnvironment>

Create also a ROS Workspace.

1.3 Install UWSim

Search for ros-groovy-uwsim in Synaptic Package Manager and install all listed packages. Further information and other installation methods can be found in <http://www.irs.uji.es/uwsim/>

¹COLA2: Component Oriented Layered-based Architecture for Autonomy. Is the control architecture used in Sparus II.

²UWSim: UnderWater Simulator. Is an open-source ROS-based simulator for underwater vehicles.

1.4 Install COLA2 architecture

Download COLA2 architecture source code and copy the cola2 parent folder to your workspace/src. Recently updated (i.e. not always tested) source code can also be found in https://bitbucket.org/udg_cirs/cola2 for experimental purposes.

Install GIT source code manager. Download auv_msgs ROS package in your workspace/src using GIT doing:

```
git clone https://bitbucket.org/udg_cirs/auv_msgs.git
```

Switch to groovy branch doing:

```
cd auv_msgs/  
git checkout --track origin/groovy
```

2 Getting started

If you are new to Linux, the following tutorial may be a good starting point to learn the basics <http://www.ee.surrey.ac.uk/Teaching/Unix/>. The main goal of this section is to learn the basics of ROS, UWSim and COLA2 architecture.

2.1 Learn ROS

First of all, you must learn ROS, e.g. by doing the tutorials on <http://wiki.ros.org/ROS/Tutorials>

By doing the tutorials two goals are accomplished: to learn ROS and to test the installation.

2.2 Learn UWSim

UWSim will be used mainly as a visualization tool. There is no need to fully understand how UWSim works to do basic simulations. Just for reference, there is a UWSim wiki at http://www.irs.uji.es/uwsim/wiki/index.php?title=Main_Page

Launch a roscore and try to launch UWSim by doing:

```
roslaunch uwsim uwsim
```

The first time you launch uwsim you will be asked to download some files from internet.

2.3 Learn Sparus II COLA2 architecture

To learn how to use Sparus II COLA2 architecture you can start by reading the main launch file, located in the package cola2_launch/launch/sparus2.launch. Configuring

its parameters you can choose which nodes you decide to launch³. You can learn what each node does by reading the Nodes section. In the following chapters you will learn how each package in COLA2 interacts with each other. Furthermore, you can use our interactive wiki to navigate through COLA2 architecture.

You can also learn a lot by using rqt Graph plugin from ROS and by simply launching the architecture and listening to all messages using standard ROS commands.

Once the architecture is running, you can drive the AUV by using the following keys:

- Use W and S to control surge velocity
- Use F and H to turn left and right using velocity controller
- Use T and G to go up and down using velocity controller
- Use A and D to turn left and right using position controller
- Use arrow Up and arrow Down to go up and down using position controller
- Use . to enable heave and yaw position controller
- Use , to enable heave and yaw velocity controller
- Use Space key to stop the AUV

In further versions of this documentation new chapters will be added, with further information about how to use COLA2 architecture.

³Avoid launching too many nodes on virtual machines, specially those which use intensive graphics.

3 Nodes

In this section there is a reference about the relationship between nodes and other architecture elements.

3.1 Node /captain

This node is used to load and execute missions or tasks. Using services, user is able to tell the captain to do different things. This node mainly interacts with the pilot.

Config file:

Publishers:

- /absolute_movement/goal
Nodes subscribed to this topic:
 /pilot
- /cola2_control/trajectory_path
- /cola2_control/mission_status
- /absolute_movement/cancel
Nodes subscribed to this topic:
 /pilot

Subscribers:

- /absolute_movement/result
Nodes publishing this topic:
 /pilot
- /absolute_movement/feedback
Nodes publishing this topic:
 /pilot
- /cola2_navigation/nav_sts
Nodes publishing this topic:
 /navigator_s2

- /absolute_movement/status
Nodes publishing this topic:

/pilot

Services:

- cola2_control/Goto
- std_srvs/Empty
- std_srvs/Empty
- cola2_control/String
- std_srvs/Empty
- std_srvs/Empty
- cola2_control/Goto
- std_srvs/Empty
- cola2_control/Goto
- cola2_safety/Submerge
- cola2_control/Goto
- cola2_control/Goto

3.2 Node /dynamics

This node uses simulated data of the actuators to compute the AUV dynamic behavior. This node can be used to simulate real AUV behavior and its interaction with the environment. User can add currents and a preliminary version of collision detection has been implemented.

Config file:

/cola2_sim/config/dynamics_s2.yaml

Publishers:

- /tf
Nodes subscribed to this topic:
 - /ekf_slam
 - /navigator_s2
- /sparus/ros_odom_to_pat
Nodes subscribed to this topic:
 - /sim_nav_sensors_s2

Subscribers:

- /cola2_control/sim_fins_data
Nodes publishing this topic:
 - /sim_actuators_s2
- /cola2_control/sim_thrusters_data
Nodes publishing this topic:
 - /sim_actuators_s2

3.3 Node /ekf_slam

EKF slam node is used to compute the AUV position and velocity. Navigator updates the filter using position or velocity updates and the filter returns the AUV position and velocity back to the navigator.

Config file:

Publishers:

- /pose_ekf_slam/odometry
Nodes subscribed to this topic:
 - /navigator_s2
- /pose_ekf_slam/covariance_marker
- /tf
Nodes subscribed to this topic:

/ekf_slam

/navigator_s2

- /pose_ekf_slam/map
- /pose_ekf_slam/landmarks

Subscribers:

- /tf

Nodes publishing this topic:

/dynamics

/ekf_slam

/navigator_s2

/sim_nav_sensors_s2

- /pose_ekf_slam/pose_update

Nodes publishing this topic:

/navigator_s2

- /pose_ekf_slam/imu_input

Nodes publishing this topic:

/navigator_s2

- /pose_ekf_slam/velocity_update

Nodes publishing this topic:

/navigator_s2

Services:

- pose_ekf_slam/SetPosition
- tf/FrameGraph
- pose_ekf_slam/SetLandmark

3.4 Node `/keyboard`

This node is used to drive the AUV in real time, just from the linux terminal.

Config file:

Publishers:

- `/cola2_control/keyboard_data`
Nodes subscribed to this topic:
`/map_ack`

3.5 Node `/map_ack`

This node is used to join all input devices, such as keyboards and joystics. Once joined, the `map_ack` node publishes a message. This node mainly interacts with the teleoperation node. When teleoperating the without cable, `map_ack` should be run outside the robot computer.

Config file:

`/cola2_control/config/map_ack_s2.yaml`

Publishers:

- `/cola2_control/map_ack_data`
Nodes subscribed to this topic:
`/teleoperation`
- `/cola2_control/map_ack_ack`
Nodes subscribed to this topic:
`/set_zero_pose`
`/teleoperation`

Subscribers:

- `/cola2_control/map_ack_ok`
Nodes publishing this topic:
`/teleoperation`

- `/cola2_control/keyboard_data`
Nodes publishing this topic:

`/keyboard`

- `/joy`

3.6 Node `/merge_body_force_req`

This node is used to merge messages, taking into account message priorities, from `/cola2_control/body_force_req` topic to `/cola2_control/merged_body_force_req` topic.

Config file:

`/cola2_control/config/merge_s2.yaml`

Publishers:

- `/cola2_control/merged_body_force_req`
Nodes subscribed to this topic:

`/thruster_allocator`

Subscribers:

- `/cola2_control/body_force_req`
Nodes publishing this topic:

`/velocity_controller_s2`

3.7 Node `/merge_body_velocity_req`

This node is used to merge messages, taking into account message priorities, from `/cola2_control/body_velocity_req` topic to `/cola2_control/merged_velocity_force_req` topic.

Config file:

`/cola2_control/config/merge_s2.yaml`

Publishers:

- `/cola2_control/merged_body_velocity_req`
Nodes subscribed to this topic:

`/velocity_controller_s2`

Subscribers:

- `/cola2_control/body_velocity_req`
Nodes publishing this topic:

`/pilot`

`/pose_controller_s2`

`/safe_depth_altitude`

`/set_zero_velocity`

`/teleoperation`

3.8 Node `/merge_world_waypoint_req`

This node is used to merge messages, taking into account message priorities, from `/cola2_control/world_waypoint_req` topic to `/cola2_control/merged_world_waypoint_req` topic.

Config file:

`/cola2_control/config/merge_s2.yaml`

Publishers:

- `/cola2_control/merged_world_waypoint_req`
Nodes subscribed to this topic:

`/pose_controller_s2`

Subscribers:

- `/cola2_navigation/nav_sts`
Nodes publishing this topic:

`/navigator_s2`

- `/cola2_control/world_waypoint_req`

Nodes publishing this topic:

`/pilot`

`/set_zero_pose`

`/teleoperation`

3.9 Node `/navigator_s2`

The navigator subscribes to sensor drivers, interacts with `ekf_slam` node, and publishes navigation status: position, velocities and more.

Config file:

`/cola2_navigation/config/navigator_s2.yaml`

Publishers:

- `/tf`

Nodes subscribed to this topic:

`/ekf_slam`

`/navigator_s2`

- `/pose_ekf_slam/pose_update`

Nodes subscribed to this topic:

`/ekf_slam`

- `/pose_ekf_slam/imu_input`

Nodes subscribed to this topic:

`/ekf_slam`

- `/pose_ekf_slam/velocity_update`

Nodes subscribed to this topic:

`/ekf_slam`

- `/cola2_navigation/nav_sts`

Nodes subscribed to this topic:

`/captain`

`/merge_world_waypoint_req`

- /pilot
- /pose_controller_s2
- /safe_depth_altitude
- /set_zero_velocity
- /teleoperation
- /thruster_allocator
- /velocity_controller_s2
- /cola2_safety/nav_sensors_status

Subscribers:

- /pose_ekf_slam/odometry
Nodes publishing this topic:
 - /ekf_slam
- /tf
Nodes publishing this topic:
 - /dynamics
 - /ekf_slam
 - /navigator_s2
 - /sim_nav_sensors_s2
- /cola2_navigation/imu
Nodes publishing this topic:
 - /sim_nav_sensors_s2
- /cola2_navigation/linkquest_navquest600_dvl
Nodes publishing this topic:
 - /sim_nav_sensors_s2
- /cola2_navigation/fastrax_it_500_gps
Nodes publishing this topic:
 - /sim_nav_sensors_s2
- /cola2_navigation/pressure_sensor
Nodes publishing this topic:

/sim_nav_sensors_s2

Services:

- tf/FrameGraph
- cola2_navigation/SetNedOrigin
- cola2_navigation/SetGPS
- std_srvs/Empty

3.10 Node /pilot

The pilot node, directed by the captain, publishes position and velocity setpoints to the position and velocity controllers. Pilot uses `move_mode.py` to compute requests.

Config file:

/cola2_control/config/pilot.yaml

Publishers:

- /absolute_movement/result
Nodes subscribed to this topic:
 /captain
- /cola2_control/waypoint_marker
- /absolute_movement/feedback
Nodes subscribed to this topic:
 /captain
- /absolute_movement/status
Nodes subscribed to this topic:
 /captain
- /cola2_control/world_waypoint_req
Nodes subscribed to this topic:
 /merge_world_waypoint_req

- `/cola2_control/body_velocity_req`
Nodes subscribed to this topic:
`/merge_body_velocity_req`

Subscribers:

- `/absolute_movement/goal`
Nodes publishing this topic:
`/captain`
- `/cola2_navigation/nav_sts`
Nodes publishing this topic:
`/navigator_s2`
- `/absolute_movement/cancel`
Nodes publishing this topic:
`/captain`

Actionlibs:

- `cola2_control/WorldWaypointReqActionResult`

3.11 Node `/pose_controller_s2`

Position controller of Sparus II AUV.

Config file:

`/cola2_control/config/pose_controller_s2.yaml`

Publishers:

- `/cola2_control/body_velocity_req`
Nodes subscribed to this topic:
`/merge_body_velocity_req`

Subscribers:

- `/cola2_navigation/nav_sts`
Nodes publishing this topic:
`/navigator_s2`
- `/cola2_control/merged_world_waypoint_req`
Nodes publishing this topic:
`/merge_world_waypoint_req`

3.12 Node `/safe_depth_altitude`

This node check AUV depth and altitude, and is mainly used to avoid collisions.

Config file:

`/cola2_safety/config/safety_s2.yaml`

Publishers:

- `/cola2_control/body_velocity_req`
Nodes subscribed to this topic:
`/merge_body_velocity_req`

Subscribers:

- `/cola2_navigation/nav_sts`
Nodes publishing this topic:
`/navigator_s2`

3.13 Node `/safety_s2`

Safety node is used to check an absolute timeout. More functionalities will be added in the near future.

Config file:

/cola2_safety/config/safety_s2.yaml

Publishers:

- /cola2_control/thrusters_data
Nodes subscribed to this topic:
 /sim_actuators_s2
- /cola2_safety/total_time

3.14 Node /set_zero_pose

If teleoperation is lost for more than 5 seconds, this node tells the robot to surface.

Config file:

Publishers:

- /cola2_control/world_waypoint_req
Nodes subscribed to this topic:
 /merge_world_waypoint_req

Subscribers:

- /cola2_control/map_ack_ack
Nodes publishing this topic:
 /map_ack

3.15 Node /set_zero_velocity

If the robot is deep enough and teleoperation is giving only disabled setpoints, this node tells the robot to keep velocities to zero.

Config file:

/cola2_safety/config/safety_s2.yaml

Publishers:

- /cola2_control/body_velocity_req
Nodes subscribed to this topic:
 /merge_body_velocity_req

Subscribers:

- /cola2_navigation/nav_sts
Nodes publishing this topic:
 /navigator_s2

3.16 Node /sim_actuators_s2

This node is used to simulate Sparus II actuators. It is only used in simulation.

Config file:

Publishers:

- /cola2_control/sim_fins_data
Nodes subscribed to this topic:
 /dynamics
- /cola2_control/sim_thrusters_data
Nodes subscribed to this topic:
 /dynamics

Subscribers:

- /cola2_control/thrusters_data
Nodes publishing this topic:
 /safety_s2

/thruster_allocator

- /cola2_control/fins_data

3.17 Node /sim_nav_sensors_s2

This node is used to simulate Sparus II navigation sensors. It is only used in simulation.

Config file:

/cola2_sim/config/sim_nav_sensors_s2.yaml

Publishers:

- /tf
Nodes subscribed to this topic:
 - /ekf_slam
 - /navigator_s2
- /cola2_navigation/imu
Nodes subscribed to this topic:
 - /navigator_s2
- /cola2_navigation/linkquest_navquest600_dvl
Nodes subscribed to this topic:
 - /navigator_s2
- /cola2_navigation/fastrax_it_500_gps
Nodes subscribed to this topic:
 - /navigator_s2
- /cola2_navigation/pressure_sensor
Nodes subscribed to this topic:
 - /navigator_s2

Subscribers:

- /sparus/range

- /sparus/ros_odom_to_pat
Nodes publishing this topic:

/dynamics

3.18 Node /teleoperation

This node is subscribed to the map_ack output message. It is used to compute position and velocity setpoints out of the input joy message.

Config file:

/cola2_control/config/teleoperation_s2.yaml

Publishers:

- /cola2_control/map_ack_ok
Nodes subscribed to this topic:

/map_ack

- /cola2_control/world_waypoint_req
Nodes subscribed to this topic:

/merge_world_waypoint_req

- /cola2_control/body_velocity_req
Nodes subscribed to this topic:

/merge_body_velocity_req

Subscribers:

- /cola2_control/map_ack_data
Nodes publishing this topic:

/map_ack

- /cola2_navigation/nav_sts
Nodes publishing this topic:

/navigator_s2

- `/cola2_control/map_ack_ack`
Nodes publishing this topic:

`/map_ack`

3.19 Node `/thruster_allocator`

This node is used to convert from forces to thruster setpoints.

Config file:

`/cola2_control/config/thruster_allocator_s2.yaml`

Publishers:

- `/cola2_control/thrusters_data`
Nodes subscribed to this topic:

`/sim_actuators_s2`

- `/diagnostics`

Subscribers:

- `/cola2_control/merged_body_force_req`
Nodes publishing this topic:

`/merge_body_force_req`

- `/cola2_navigation/nav_sts`
Nodes publishing this topic:

`/navigator_s2`

Services:

- `std_srvs/Empty`
- `std_srvs/Empty`

3.20 Node `/velocity_controller_s2`

Velocity controller of Sparus II AUV.

Config file:

`/cola2_control/config/velocity_controller_s2.yaml`

Publishers:

- `/cola2_control/body_force_req`
Nodes subscribed to this topic:
`/merge_body_force_req`

Subscribers:

- `/cola2_control/merged_body_velocity_req`
Nodes publishing this topic:
`/merge_body_velocity_req`
- `/cola2_navigation/nav_sts`
Nodes publishing this topic:
`/navigator_s2`

4 Messages

In this section there is a reference about the relationship between messages and other architecture elements.

4.1 Message file `actionlib_msgs/GoalID`

Nodes using this message file:

- `/captain`
- `/pilot`

Message file description:

```
1 time stamp
2 string id
```

4.2 Message file `actionlib_msgs/GoalStatusArray`

Nodes using this message file:

- `/captain`
- `/pilot`

Message file description:

```
1 std_msgs/Header header
2   uint32 seq
3   time stamp
4   string frame_id
5 actionlib_msgs/GoalStatus[] status_list
6   uint8 PENDING=0
7   uint8 ACTIVE=1
8   uint8 PREEMPTED=2
9   uint8 SUCCEEDED=3
10  uint8 ABORTED=4
11  uint8 REJECTED=5
12  uint8 PREEMPTING=6
13  uint8 RECALLING=7
14  uint8 RECALLED=8
15  uint8 LOST=9
16  actionlib_msgs/GoalID goal_id
17    time stamp
18    string id
19    uint8 status
20    string text
```

4.3 Message file `auv_msgs/BodyForceReq`

Nodes using this message file:

- `/merge_body_force_req`
- `/thruster_allocator`
- `/velocity_controller_s2`

Message file description:

```
1  std_msgs/Header header
2    uint32 seq
3    time stamp
4    string frame_id
5  auv_msgs/GoalDescriptor goal
6    uint32 PRIORITY_LOW=0
7    uint32 PRIORITY_NORMAL=10
8    uint32 PRIORITY_AVOID_OBSTACLE=20
9    uint32 PRIORITY_EMERGENCY=30
10   uint32 PRIORITY_MANUAL_OVERRIDE=40
11   string requester
12   uint32 id
13   uint32 priority
14  geometry_msgs/Wrench wrench
15    geometry_msgs/Vector3 force
16      float64 x
17      float64 y
18      float64 z
19    geometry_msgs/Vector3 torque
20      float64 x
21      float64 y
22      float64 z
23  auv_msgs/Bool6Axis disable_axis
24    bool x
25    bool y
26    bool z
27    bool roll
28    bool pitch
29    bool yaw
```

4.4 Message file `auv_msgs/BodyVelocityReq`

Nodes using this message file:

- `/merge_body_velocity_req`
- `/pilot`
- `/pose_controller_s2`
- `/safe_depth.altitude`
- `/set_zero_velocity`
- `/teleoperation`

- /velocity_controller_s2

Message file description:

```
1 std_msgs/Header header
2   uint32 seq
3   time stamp
4   string frame_id
5 auv_msgs/GoalDescriptor goal
6   uint32 PRIORITY_LOW=0
7   uint32 PRIORITY_NORMAL=10
8   uint32 PRIORITY_AVOID_OBSTACLE=20
9   uint32 PRIORITY_EMERGENCY=30
10  uint32 PRIORITY_MANUAL_OVERRIDE=40
11  string requester
12  uint32 id
13  uint32 priority
14 geometry_msgs/Twist twist
15   geometry_msgs/Vector3 linear
16     float64 x
17     float64 y
18     float64 z
19   geometry_msgs/Vector3 angular
20     float64 x
21     float64 y
22     float64 z
23 auv_msgs/Bool6Axis disable_axis
24   bool x
25   bool y
26   bool z
27   bool roll
28   bool pitch
29   bool yaw
```

4.5 Message file auv_msgs/NavSts

Nodes using this message file:

- /captain
- /merge_world_waypoint_req
- /navigator_s2
- /pilot
- /pose_controller_s2
- /safe_depth_altitude
- /set_zero_velocity
- /teleoperation
- /thruster_allocator
- /velocity_controller_s2

Message file description:

```

1  uint8 STATUS_FAULT=0
2  uint8 STATUS_LOCAL_FRAME_OK=1
3  uint8 STATUS_GLOBAL_FRAME_OK=2
4  uint8 STATUS_POSITION_OK=3
5  uint8 STATUS_VELOCITY_OK=4
6  uint8 STATUS_ESTIMATION_ERROR_OK=8
7  uint8 STATUS_ALL_OK=15
8  std_msgs/Header header
9    uint32 seq
10   time stamp
11   string frame_id
12  auv_msgs/DecimalLatLon global_position
13    float64 latitude
14    float64 longitude
15  auv_msgs/DecimalLatLon origin
16    float64 latitude
17    float64 longitude
18  auv_msgs/NED position
19    float64 north
20    float64 east
21    float64 depth
22  float32 altitude
23  geometry_msgs/Point body_velocity
24    float64 x
25    float64 y
26    float64 z
27  auv_msgs/RPY orientation
28    float32 roll
29    float32 pitch
30    float32 yaw
31  auv_msgs/RPY orientation_rate
32    float32 roll
33    float32 pitch
34    float32 yaw
35  auv_msgs/NED position_variance
36    float64 north
37    float64 east
38    float64 depth
39  auv_msgs/RPY orientation_variance
40    float32 roll
41    float32 pitch
42    float32 yaw
43  uint8 status

```

4.6 Message file auv_msgs/WorldWaypointReq

Nodes using this message file:

- /merge_world_waypoint_req
- /pilot
- /pose_controller_s2
- /set_zero_pose
- /teleoperation

Message file description:

```

1  std_msgs/Header header
2    uint32 seq
3    time stamp
4    string frame_id

```

```

5  auv_msgs/GoalDescriptor goal
6  uint32 PRIORITY_LOW=0
7  uint32 PRIORITY_NORMAL=10
8  uint32 PRIORITY_AVOID_OBSTACLE=20
9  uint32 PRIORITY_EMERGENCY=30
10 uint32 PRIORITY_MANUAL_OVERRIDE=40
11 string requester
12 uint32 id
13 uint32 priority
14 bool altitude_mode
15 auv_msgs/NED position
16 float64 north
17 float64 east
18 float64 depth
19 float32 altitude
20 auv_msgs/RPY orientation
21 float32 roll
22 float32 pitch
23 float32 yaw
24 auv_msgs/Bool6Axis disable_axis
25 bool x
26 bool y
27 bool z
28 bool roll
29 bool pitch
30 bool yaw
31 geometry_msgs/Vector3 position_tolerance
32 float64 x
33 float64 y
34 float64 z
35 auv_msgs/RPY orientation_tolerance
36 float32 roll
37 float32 pitch
38 float32 yaw

```

4.7 Message file cola2_control/Setpoints

Nodes using this message file:

- /dynamics
- /safety_s2
- /sim_actuators_s2
- /thruster_allocator

Message file description:

```

1  std_msgs/Header header
2  uint32 seq
3  time stamp
4  string frame_id
5  float64[] setpoints

```

4.8 Message file cola2_control/WorldWaypointReqActionFeedback

Nodes using this message file:

- /captain
- /pilot

Message file description:

1 `Unknown type message`

4.9 Message file cola2_control/WorldWaypointReqActionGoal

Nodes using this message file:

- /captain
- /pilot

Message file description:

1 `Unknown type message`

4.10 Message file cola2_control/WorldWaypointReqActionResult

Nodes using this message file:

- /captain
- /pilot

Message file description:

1 `Unknown type message`

4.11 Message file cola2_navigation/FastraxIt500Gps

Nodes using this message file:

- /navigator_s2
- /sim_nav_sensors_s2

Message file description:

```

1  std_msgs/Header header
2    uint32 seq
3    time stamp
4    string frame_id
5  float64 latitude
6  int32 latitude_hemisphere
7  float64 longitude
8  int32 longitude_hemisphere
9  string time_utc
10 int32 data_quality
11 float64 north
12 float64 east
13 string utm_zone
14 float32 h_dop
15 float32 v_dop
16 string[] raw_data

```

4.12 Message file cola2_navigation/LinkquestDvl

Nodes using this message file:

- /navigator_s2
- /sim_nav_sensors_s2

Message file description:

```

1  std_msgs/Header header
2    uint32 seq
3    time stamp
4    string frame_id
5  string errCode
6  int32[4] dataGood
7  float64[4] altitudeBeam
8  float64[4] bottomVelocityBeam
9  float64[4] waterVelocityBeam
10 float64[4] waterVelocityCredit
11 float64[3] velocityInst
12 int32 velocityInstFlag
13 float64[3] velocityEarth
14 int32 velocityEarthFlag
15 float64[3] waterVelocityInst
16 int32 waterVelocityInstFlag
17 float64[3] waterVelocityEarth
18 int32 waterVelocityEarthFlag
19 float64 roll
20 float64 pitch
21 float64 heading
22 float64 altitude
23 float64 temperature
24 float64 pressure
25 float64 salinity
26 float64 soundSpeed
27 string rawData

```


4.13 Message file cola2_navigation/PressureSensor

Nodes using this message file:

- /navigator_s2
- /sim_nav_sensors_s2

Message file description:

```
1 std_msgs/Header header
2   uint32 seq
3   time stamp
4   string frame_id
5 float64 pressure
6 float64 temperature
```

4.14 Message file cola2_safety/MissionStatus

Nodes using this message file:

- /captain

Message file description:

```
1 std_msgs/Header header
2   uint32 seq
3   time stamp
4   string frame_id
5 int32 current_wp
6 int32 total_wp
7 bool altitude_mode
8 float64 current_north
9 float64 current_east
10 float32 current_depth
11 float32 current_altitude
12 float64 wp_north
13 float64 wp_east
14 float32 wp_depth_altitude
15 int32 wp_remaining_time
```

4.15 Message file cola2_safety/NavSensorsStatus

Nodes using this message file:

- /navigator_s2

Message file description:

```

1 std_msgs/Header header
2   uint32 seq
3   time stamp
4   string frame_id
5 bool dvl_status
6 bool imu_status
7 bool svx_status

```

4.16 Message file cola2_safety/TotalTime

Nodes using this message file:

- /safety_s2

Message file description:

```

1 uint32 total_time

```

4.17 Message file diagnostic_msgs/DiagnosticArray

Nodes using this message file:

- /thruster_allocator

Message file description:

```

1 std_msgs/Header header
2   uint32 seq
3   time stamp
4   string frame_id
5 diagnostic_msgs/DiagnosticStatus[] status
6   byte OK=0
7   byte WARN=1
8   byte ERROR=2
9   byte level
10  string name
11  string message
12  string hardware_id
13  diagnostic_msgs/KeyValue[] values
14    string key
15    string value

```

4.18 Message file geometry_msgs/PoseWithCovarianceStamped

Nodes using this message file:

- /ekf_slam
- /navigator_s2

Message file description:

```
1 std_msgs/Header header
2   uint32 seq
3   time stamp
4   string frame_id
5 geometry_msgs/PoseWithCovariance pose
6   geometry_msgs/Pose pose
7     geometry_msgs/Point position
8     float64 x
9     float64 y
10    float64 z
11    geometry_msgs/Quaternion orientation
12    float64 x
13    float64 y
14    float64 z
15    float64 w
16 float64[36] covariance
```

4.19 Message file geometry_msgs/TwistWithCovarianceStamped

Nodes using this message file:

- /ekf_slam
- /navigator_s2

Message file description:

```
1 std_msgs/Header header
2   uint32 seq
3   time stamp
4   string frame_id
5 geometry_msgs/TwistWithCovariance twist
6   geometry_msgs/Twist twist
7     geometry_msgs/Vector3 linear
8     float64 x
9     float64 y
10    float64 z
11    geometry_msgs/Vector3 angular
12    float64 x
13    float64 y
14    float64 z
15 float64[36] covariance
```

4.20 Message file nav_msgs/Odometry

Nodes using this message file:

- /dynamics
- /ekf_slam
- /navigator_s2
- /sim_nav_sensors_s2

Message file description:

```

1  std_msgs/Header header
2  uint32 seq
3  time stamp
4  string frame_id
5  string child_frame_id
6  geometry_msgs/PoseWithCovariance pose
7  geometry_msgs/Pose pose
8  geometry_msgs/Point position
9  float64 x
10 float64 y
11 float64 z
12 geometry_msgs/Quaternion orientation
13 float64 x
14 float64 y
15 float64 z
16 float64 w
17 float64[36] covariance
18 geometry_msgs/TwistWithCovariance twist
19 geometry_msgs/Twist twist
20 geometry_msgs/Vector3 linear
21 float64 x
22 float64 y
23 float64 z
24 geometry_msgs/Vector3 angular
25 float64 x
26 float64 y
27 float64 z
28 float64[36] covariance

```

4.21 Message file nav_msgs/Path

Nodes using this message file:

- /captain

Message file description:

```

1  std_msgs/Header header
2  uint32 seq
3  time stamp
4  string frame_id
5  geometry_msgs/PoseStamped[] poses
6  std_msgs/Header header
7  uint32 seq
8  time stamp

```

```
9   string frame_id
10  geometry_msgs/Pose pose
11  geometry_msgs/Point position
12  float64 x
13  float64 y
14  float64 z
15  geometry_msgs/Quaternion orientation
16  float64 x
17  float64 y
18  float64 z
19  float64 w
```

4.22 Message file pose_ekf_slam/Map

Nodes using this message file:

- /ekf_slam

Message file description:

```
1  std_msgs/Header header
2  uint32 seq
3  time stamp
4  string frame_id
5  pose_ekf_slam/Landmark[] landmark
6  time last_update
7  string landmark_id
8  geometry_msgs/PoseWithCovariance pose
9  geometry_msgs/Pose pose
10 geometry_msgs/Point position
11 float64 x
12 float64 y
13 float64 z
14 geometry_msgs/Quaternion orientation
15 float64 x
16 float64 y
17 float64 z
18 float64 w
19 float64[36] covariance
```

4.23 Message file sensor_msgs/Imu

Nodes using this message file:

- /ekf_slam
- /navigator_s2
- /sim_nav_sensors_s2

Message file description:

```
1 std_msgs/Header header
2   uint32 seq
3   time stamp
4   string frame_id
5 geometry_msgs/Quaternion orientation
6   float64 x
7   float64 y
8   float64 z
9   float64 w
10 float64[9] orientation_covariance
11 geometry_msgs/Vector3 angular_velocity
12   float64 x
13   float64 y
14   float64 z
15 float64[9] angular_velocity_covariance
16 geometry_msgs/Vector3 linear_acceleration
17   float64 x
18   float64 y
19   float64 z
20 float64[9] linear_acceleration_covariance
```

4.24 Message file sensor_msgs/Joy

Nodes using this message file:

- /keyboard
- /map_ack
- /teleoperation

Message file description:

```
1 std_msgs/Header header
2   uint32 seq
3   time stamp
4   string frame_id
5 float32[] axes
6 int32[] buttons
```

4.25 Message file sensor_msgs/Range

Nodes using this message file:

- /sim_nav_sensors_s2

Message file description:

```
1 uint8 ULTRASOUND=0
2 uint8 INFRARED=1
3 std_msgs/Header header
4   uint32 seq
5   time stamp
6   string frame_id
7 uint8 radiation_type
8 float32 field_of_view
9 float32 min_range
10 float32 max_range
11 float32 range
```

4.26 Message file std_msgs/String

Nodes using this message file:

- /map_ack
- /set_zero_pose
- /teleoperation

Message file description:

```
1 string data
```

4.27 Message file tf/tfMessage

Nodes using this message file:

- /dynamics
- /ekf_slam
- /navigator_s2
- /sim_nav_sensors_s2

Message file description:

```
1 geometry_msgs/TransformStamped[] transforms
2 std_msgs/Header header
3   uint32 seq
4   time stamp
5   string frame_id
6   string child_frame_id
7   geometry_msgs/Transform transform
8   geometry_msgs/Vector3 translation
```

```

9      float64 x
10     float64 y
11     float64 z
12     geometry_msgs/Quaternion rotation
13     float64 x
14     float64 y
15     float64 z
16     float64 w

```

4.28 Message file visualization_msgs/MarkerArray

Nodes using this message file:

- /ekf_slam

Message file description:

```

1 visualization_msgs/Marker[] markers
2   uint8 ARROW=0
3   uint8 CUBE=1
4   uint8 SPHERE=2
5   uint8 CYLINDER=3
6   uint8 LINE_STRIP=4
7   uint8 LINE_LIST=5
8   uint8 CUBE_LIST=6
9   uint8 SPHERE_LIST=7
10  uint8 POINTS=8
11  uint8 TEXT_VIEW_FACING=9
12  uint8 MESH_RESOURCE=10
13  uint8 TRIANGLE_LIST=11
14  uint8 ADD=0
15  uint8 MODIFY=0
16  uint8 DELETE=2
17  std_msgs/Header header
18    uint32 seq
19    time stamp
20    string frame_id
21  string ns
22  int32 id
23  int32 type
24  int32 action
25  geometry_msgs/Pose pose
26    geometry_msgs/Point position
27      float64 x
28      float64 y
29      float64 z
30    geometry_msgs/Quaternion orientation
31      float64 x
32      float64 y
33      float64 z
34      float64 w
35  geometry_msgs/Vector3 scale
36    float64 x
37    float64 y
38    float64 z
39  std_msgs/ColorRGBA color
40    float32 r
41    float32 g
42    float32 b
43    float32 a
44  duration lifetime
45  bool frame_locked
46  geometry_msgs/Point[] points
47    float64 x
48    float64 y
49    float64 z
50  std_msgs/ColorRGBA[] colors
51    float32 r
52    float32 g
53    float32 b
54    float32 a
55  string text

```



```

56 string mesh_resource
57 bool mesh_use_embedded_materials

```

4.29 Message file visualization_msgs/Marker

Nodes using this message file:

- /ekf_slam
- /pilot

Message file description:

```

1  uint8 ARROW=0
2  uint8 CUBE=1
3  uint8 SPHERE=2
4  uint8 CYLINDER=3
5  uint8 LINE_STRIP=4
6  uint8 LINE_LIST=5
7  uint8 CUBE_LIST=6
8  uint8 SPHERE_LIST=7
9  uint8 POINTS=8
10 uint8 TEXT_VIEW_FACING=9
11 uint8 MESH_RESOURCE=10
12 uint8 TRIANGLE_LIST=11
13 uint8 ADD=0
14 uint8 MODIFY=0
15 uint8 DELETE=2
16 std_msgs/Header header
17   uint32 seq
18   time stamp
19   string frame_id
20 string ns
21 int32 id
22 int32 type
23 int32 action
24 geometry_msgs/Pose pose
25   geometry_msgs/Point position
26     float64 x
27     float64 y
28     float64 z
29   geometry_msgs/Quaternion orientation
30     float64 x
31     float64 y
32     float64 z
33     float64 w
34 geometry_msgs/Vector3 scale
35   float64 x
36   float64 y
37   float64 z
38 std_msgs/ColorRGBA color
39   float32 r
40   float32 g
41   float32 b
42   float32 a
43 duration lifetime
44 bool frame_locked
45 geometry_msgs/Point [] points
46   float64 x
47   float64 y
48   float64 z
49 std_msgs/ColorRGBA [] colors
50   float32 r
51   float32 g
52   float32 b
53   float32 a
54 string text
55 string mesh_resource
56 bool mesh_use_embedded_materials

```

5 Services

In this section there is a reference about the relationship between services and other architecture elements.

5.1 Service file cola2_control/Goto

Nodes using this service file:

- /captain

Service input params:

float64 north_lat float64 east_lon float32 z bool altitude_mode float32 tolerance

Service output params:

bool attempted

5.2 Service file cola2_control/String

Nodes using this service file:

- /captain

Service input params:

string mystring

Service output params:

5.3 Service file cola2_navigation/SetGPS

Nodes using this service file:

- /navigator_s2

Service input params:

Service output params:

string success

5.4 Service file cola2_navigation/SetNedOrigin

Nodes using this service file:

- /navigator_s2

Service input params:

float64 latitude float64 longitude

Service output params:

bool success

5.5 Service file cola2_safety/Submerge

Nodes using this service file:

- /captain

Service input params:

float32 z bool altitude_mode

Service output params:

bool attempted

5.6 Service file pose_ekf_slam/SetLandmark

Nodes using this service file:

- /ekf_slam

Service input params:

geometry_msgs/PoseWithCovariance landmark geometry_msgs/Pose pose geometry_msgs/Point
position float64 x float64 y float64 z geometry_msgs/Quaternion orientation float64 x
float64 y float64 z float64 w float64[36] covariance string topic_name

Service output params:

5.7 Service file pose_ekf_slam/SetPosition

Nodes using this service file:

- /ekf_slam

Service input params:

geometry_msgs/Point position float64 x float64 y float64 z

Service output params:

5.8 Service file std_srvs/Empty

Nodes using this service file:

- /captain
- /navigator_s2
- /thruster_allocator

Service input params:

Service output params:

5.9 Service file tf/FrameGraph

Nodes using this service file:

- /ekf_slam
- /navigator_s2

Service input params:

Service output params:

string dot_graph


```

62 0, 0, 0, 0, 0, 0, 0, 0, 0,
63 0, 0, 0, 0, 0, 0, 0, 0, 0,
64 0, 0, 0, 0, 0, 0, 0, 0, 0,
65 0, 0, 0, 0, 0, 0, 0, 0, 0,
66 0, 0, 0, 0, 0, 0, 0, 0, 0,
67 0, 0, 0, 0, 0, 0, 0, 0, 0,
68 0, 0, 0, 0, 0, 0, 0, 0, 0,
69 0, 0, 0, 0, 0, 0, 0, 0, 0,
70 0, 0, 0, 0, 0, 0, 0, 0, 0,
71 0, 0, 0, 0, 0, 0, 0, 0, 0,
72 0, 0, 0, 0, 0, 0, 0, 0, 0,
73 0, 0, 0, 0, 0, 0, 0, 0, 0,
74 0, 0, 0, 0, 0, 0, 0, 0, 0,
75 0, 0, 0, 0, 0, 0, 0, 0, 0]
76
77 map_ack/joystick/buttons_to_shadow_buttons:
78 [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
79 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
80 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
81 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
82 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
83 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
84 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
85 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
86 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
87 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
88 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
89 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
90 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
91 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
92 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
93 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
94 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
95 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
96 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
97 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
98 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,
99 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
100 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
101 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,
102 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
103 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
104
105 map_ack/joystick/shadow_buttons_to_output_axes:
106 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
107 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
108 0, 0, 0, 0, 0, 0, 0.05, -0.05, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
109 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
110 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
111 0, 0, 0, 0.125, -0.125, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
112 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
113 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
114 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
115 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
116 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
117 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
118
119
120 # Keyboard device
121 map_ack/keyboard/n_axes: 0
122 map_ack/keyboard/n_buttons: 17
123 map_ack/keyboard/axes_to_output_axes: []
124 map_ack/keyboard/axes_to_shadow_buttons: []
125 map_ack/keyboard/buttons_to_shadow_buttons:
126 [1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
127 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
128 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
129 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
130 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
131 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
132 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
133 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,
134 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0,
135 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0,
136 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1,
137 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
138 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
139 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
140 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,
141 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
142 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
143 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,
144 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
145 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
146 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,
147 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
148 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
149 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,

```

```

150 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
151 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
152
153 map_ack/keyboard/shadow_buttons_to_output_axes:
154 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
155 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
156 0, 0, 0, 0, 0, -0.05, 0.05, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
157 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
158 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
159 0, 0, 0, -0.125, 0.125, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
160 0, 0, 0.1, -0.1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
161 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,
162 0, 0, 0, 0, 0, 0, 0, -0.1, 0.1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
163 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,
164 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,
165 0, 0, 0, 0, 0, 0, 0, 0, 0, -0.1, 0.1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

```

6.2 Config file /cola2_control/config/merge_s2.yaml

Nodes using this configuration file:

- /merge_body_force_req
- /merge_body_velocity_req
- /merge_world_waypoint_req

Config file description:

```

1 # SPARUS2 CONFIG FILE FOR ALL MERGE NODES
2 merge/rate: 10
3 merge/frame_id: "sparus2"

```

6.3 Config file /cola2_control/config/pilot.yaml

Nodes using this configuration file:

- /pilot

Config file description:

```

1 #Pose PID per DoF
2 pilot/pid_x_z_yaw_feed_forward_force: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
3 pilot/pid_x_z_yaw_kp: [0.2, 0.0, 0.3, 0.0, 0.0, 0.8]
4 pilot/pid_x_z_yaw_ti: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
5 pilot/pid_x_z_yaw_td: [0.2, 0.0, 0.0, 0.0, 0.0, 0.3]
6
7 pilot/pid_x_y_z_yaw_feed_forward_force: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
8 pilot/pid_x_y_z_yaw_kp: [1.0, 1.0, 0.5, 0.0, 0.0, 0.8]
9 pilot/pid_x_y_z_yaw_ti: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
10 pilot/pid_x_y_z_yaw_td: [0.2, 0.2, 0.1, 0.0, 0.0, 0.3]
11
12 pilot/relative_pid_x_y_z_yaw_feed_forward_force: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]

```



```

13 pilot/relative_pid_x_y_z_yaw_kp: [0.2, 0.2, 0.2, 0.0, 0.0, 0.4]
14 pilot/relative_pid_x_y_z_yaw_ti: [8.0, 8.0, 8.0, 0.0, 0.0, 8.0]
15 pilot/relative_pid_x_y_z_yaw_td: [0.1, 0.1, 0.1, 0.0, 0.0, 0.1]
16
17 # pilot/max_velocity: [0.5, 0.2, 0.5, 0.0, 0.0, 0.3] # --> check basic_mission_parameters.yaml
18 pilot/min_velocity_los: [0.2, 0.0, 0.0, 0.0, 0.0, 0.0]
19 pilot/max_angle_error: 0.3

```

6.4 Config file /cola2_control/config/pose_controller_s2.yaml

Nodes using this configuration file:

- /pose_controller_s2

Config file description:

```

1 # SPARUS 2 POSE controller config file
2 # Per DoF [X, Y, Z, Roll, Pitch, Yaw]
3 pose_controller/velocity_max: [0.0, 0.0, 0.5, 0.0, 0.0, 0.3]
4 pose_controller/start_depth_with_fins_velocity: 2.0
5 pose_controller/max_pitch: 0.5
6 pose_controller/surface_zone: 0.5
7
8 # Velocity PID per DoF
9 pose_controller/pid_pose_feed_forward_velocity: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
10 pose_controller/pid_pose_kp: [0.0, 0.0, 1.0, 0.0, 1.0, 1.5]
11 pose_controller/pid_pose_ti: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
12 pose_controller/pid_pose_td: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
13 pose_controller/pid_pose_i_limit: [1.0, 1.0, 0.75, 1.0, 1.0, 0.1]
14
15 pose_controller/pid_depth_with_fins_kp: -0.5
16 pose_controller/pid_depth_with_fins_ti: -0.07
17 pose_controller/pid_depth_with_fins_td: -0.2

```

6.5 Config file /cola2_control/config/teleoperation_s2.yaml

Nodes using this configuration file:

- /teleoperation

Config file description:

```

1 # teleoperation.py config file
2 teleoperation/max_pos: [0.0, 0.0, 5.0, 3.14159265359, 1.0, 3.14159265359]
3 teleoperation/min_pos: [0.0, 0.0, 0.0, -3.14159265359, -1.0, -3.14159265359]
4 teleoperation/max_vel: [1.0, 0.0, 0.4, 0.0, 1.0, 0.4]
5 teleoperation/min_vel: [-1.0, 0.0, -0.4, 0.0, -1.0, -0.4]
6 teleoperation/pose_controlled_axis: [False, False, False, False, False, False]
7 teleoperation/base_pose: [0, 0, 0, 0, 0, 0]
8 teleoperation/actualize_base_pose: False

```

6.6 Config file /cola2_control/config/thruster_allocator_s2.yaml

Nodes using this configuration file:

- /thruster_allocator

Config file description:

```

1 # SPARUS 2 thruster allocator config file
2 thruster_allocator/frame_id: "sparus2"
3 thruster_allocator/n_thrusters: 3
4 thruster_allocator/max_force_surge_plus_yaw: 60.0
5 thruster_allocator/force_to_thrusters_ratio: 30.0
6 thruster_allocator/max_setpoint: 1.0
7
8 # To linearize the actuators output a polynomial is applied 30*0.18257418598=5.4772255794 with 0.5 (sqrt)
9 thruster_allocator/thrusters_polynomial_linearization: [0.0, 1.0]
10 thruster_allocator/thrusters_polynomial_linearization_exp: [0.0, 1.0]
11
12 # Dof x actuators
13 thruster_allocator/thruster_control_matrix: [[0.0, 1.0, 1.0],[0.0, 0.0, 0.0],[1.0, 0.0, 0.0],[0.0, 0.0, 0.0],[0.0,
0.0, 0.0],[0.0, -0.16, 0.16]]

```

6.7 Config file /cola2_control/config/velocity_controller_s2.yaml

Nodes using this configuration file:

- /velocity_controller_s2

Config file description:

```

1 # SPARUS 2 velocity controller config file
2 # Per DoF [X, Y, Z, Roll, Pitch, Yaw]
3 velocity_controller/force_max: [30.0, 0.0, 30.0, 0.0, 30.0, 60.0]
4
5 # Velocity PID per DoF
6 velocity_controller/pid_velocity_feed_forward_force: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
7 velocity_controller/pid_velocity_kp: [0.5, 0.0, 2.0, 0.0, 1.0, 0.2]
8 velocity_controller/pid_velocity_ti: [4.0, 0.0, 7.0, 0.0, 0.0, 10.0]
9 velocity_controller/pid_velocity_td: [0.0, 0.0, 0.5, 0.0, 0.0, 0.0]
10 velocity_controller/pid_velocity_i_limit: [1.0, 1.0, 1.0, 1.0, 1.0, 1.0]
11
12 # Open loop controller. It adjust the desired velocity for each Dof using a polynomi. Ex. [0.1, -1.2, 2.0] => y =
0.1*x^2 -1.2*x + 2.0
13 velocity_controller/open_loop_adjust_poly_x: [0.0, 0.0]
14 velocity_controller/open_loop_adjust_poly_y: [0.0, 0.0]
15 velocity_controller/open_loop_adjust_poly_z: [0.0, 0.0]
16 velocity_controller/open_loop_adjust_poly_roll: [0.0, 0.0]
17 velocity_controller/open_loop_adjust_poly_pitch: [0.0, 0.0]
18 velocity_controller/open_loop_adjust_poly_yaw: [0.0, 0.0]

```

6.8 Config file /cola2_navigation/config/navigator_s2.yaml

Nodes using this configuration file:

- /navigator_s2

Config file description:

```

1 # SPARUS2 NAVIGATOR CONFIG
2 # Frames
3 navigator/robot_frame_name: "sparus2"
4 navigator/world_frame_name: "world"
5
6 # DVL parameter used for filtering purposes
7 navigator/dvl_max_v: 3.0
8
9 # Depth sensor correction. This value is added to the depth
10 navigator/depth_correction: 0.0
11 navigator/water_density: 1030
12
13 # GPS parameters
14 navigator/gps_update: false
15 navigator/gps_init_samples: 8
16
17 # Check sensors parameters
18 navigator/check_sensors_period: 2.0
19 navigator/dvl_max_period_error: 1.0
20 navigator/imu_max_period_error: 1.0
21 navigator/max_init_time: 25.0
22
23 # NED parameters
24 navigator/fixned: true
25 navigator/ned_latitude: 43.105805
26 navigator/ned_longitude: 5.884812

```

6.9 Config file /cola2_safety/config/safety_s2.yaml

Nodes using this configuration file:

- /safe_depth_altitude
- /safety_s2
- /set_zero_velocity

Config file description:

```

1 safety_s2/absolute_timeout: 3600
2 safety_s2/emerge: False
3 safety_safe_depth_altitude/max_depth: 10.0
4 safety_safe_depth_altitude/min_altitude: 2.5
5 safety_set_zero_velocity/set_zero_velocity_depth: 1.5
6 safety_set_zero_velocity/set_zero_velocity_axis: [[False, True, True, True, True, True, True],[True, False, True, True,
  True, True], [True, True, False, True, True, True], [True, True, True, False, True, True], [True, True, True,
  True, False, True], [True, True, True, True, True, False]]

```

6.10 Config file /cola2_sim/config/dynamics_s2.yaml

Nodes using this configuration file:

- /dynamics

Config file description:

```

1 # sparus2 dynamics model
2 # WARNING!!!
3 # The thrusters allocation matrix "b" is defined in the code. It can not be changed using parameters.
4
5 vehicle_name: "sparus2"
6
7 dynamics/sparus2/number_of_thrusters: 3
8 dynamics/sparus2/thrusters_topic: "/cola2_control/sim_thrusters_data"
9 dynamics/sparus2/number_of_fins: 1
10 dynamics/sparus2/fins_topic: "/cola2_control/sim_fins_data"
11 dynamics/sparus2/a_fins: 0.035
12 dynamics/sparus2/k_fins: 1.0
13 dynamics/sparus2/max_fins_angle: 1.0
14
15 dynamics/sparus2/period: 0.05
16 dynamics/sparus2/mass: 34.5
17 dynamics/sparus2/gravity_center: [0.0, 0.0, 0.01]
18 dynamics/sparus2/g: 9.81
19 dynamics/sparus2/radius: 0.2009312118
20 dynamics/sparus2/surface_radius: 0.4
21
22 # Forward and backward thrusters coef 0.000020833333
23 dynamics/sparus2/ctf: 0.000020833333
24 dynamics/sparus2/ctb: 0.000020833333
25
26 dynamics/sparus2/max_thrusters_rpm: 1200
27 dynamics/sparus2/dzv: 0.05
28 dynamics/sparus2/dv: 0.05
29 dynamics/sparus2/dh: 0.40
30 dynamics/sparus2/density: 1030.0
31
32 dynamics/sparus2/thrusters_matrix: [[0.0, 1.0, 1.0],[0.0, 0.0, 0.0],[1.0, 0.0, 0.0],[0.0, 0.0, 0.0],[0.0, 0.0,
33 0.0],[0.0, -0.16, 0.16]]
34
35 # [0.1337, 0.0, 0.0, 0.0, 1.6643, 0.0, 0.0, 0.0, 1.6643]
36 dynamics/sparus2/tensor: [0.7, 0.0, 0.0, 0.0, 8.0, 0.0, 0.0, 0.0, 8.0]
37 dynamics/sparus2/damping: [-20.0, -60.0, -60.0, -1.0, -8.0, -8.0]
38 dynamics/sparus2/quadratic_damping: [-30.0, -60.0, -60.0, 0.0, -10.0, -10.0]
39
40 #Meters and Rads [X, Y, Z, Roll, Pitch, Yaw]
41 dynamics/sparus2/initial_pose: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
42 dynamics/sparus2/initial_velocity: [0, 0, 0, 0, 0, 0]
43
44 dynamics/sparus2/odom_topic_name: "/sparus/ros_odom_to_pat"
45 dynamics/sparus2/frame_id: "sparus2_dynamics"
46 dynamics/sparus2/world_frame_id: "world"
47
48 # Collisions from UWSim. Let it empty if not available
49 dynamics/sparus2/uwsim_contact_sensor: ""
50
51 #WATER CURRENTs SIMULATION
52 dynamics/current_mean: [0.00, 0.1, 0]
53 dynamics/current_sigma: [0.01, 0.01, 0.01]
54 dynamics/current_min: [-0.0, -0.0, -0.0]
55 dynamics/current_max: [0.0, 0.0, 0.0]

```

6.11 Config file /cola2_sim/config/sim_nav_sensors_s2.yaml

Nodes using this configuration file:

- /sim_nav_sensors_s2

Config file description:

```
1 # Sensors information
2 dynamics/sparus2/ned_origin_latitude: 41.967283
3 dynamics/sparus2/ned_origin_longitude: 2.836075
4
5 # Sant Feliu (41.78077, 3.04748)
6 # Piscina (41.967283, 2.836075)
7
8 # Transformation wrt vehicle frame: [x, y, z, roll, pitch, yaw] (in meters and degrees!!!)
9 adis_imu/tf: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
10 linkquest_dvl/tf: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
11 gps/tf: [0.0, 0.0, 0.0, 0.0, 0.0, 0.0]
12
13 # Sensors publication periods
14 adis_imu/period: 0.05
15 linkquest_dvl/period: 0.2
16 gps/period: 1.0
17
18 # Sensors covariances
19 adis_imu/orientation_covariance: [0.0001, 0.0001, 0.0001]
20 linkquest_dvl/velocity_covariance: [0.0015, 0.0015, 0.0015] #Not computed!!!
21 gps/position_covariance: [0.5, 0.5] #Not computed!!!
22
23 dynamics/sparus2/altitude_range_topic_name: "/sparus/range"
```