1. Software instructions



Table of Contents

[1 Global 3](#_Toc105146557)

[• Necessary software 3](#_Toc105146558)

[1.1 Establish a connection between the GCS and the autopilot through telemetry 3](#_Toc105146559)

[1.2 Upload of the ardupilot parameters 3](#_Toc105146560)

[1.3 Test the communication between the RC and the autopilot and calibrate it 4](#_Toc105146561)

[1.4 Test the reception of a GPS signal 4](#_Toc105146562)

[1.5 Configure the RTK GNSS 4](#_Toc105146563)

[1.6 Configure the RC 7](#_Toc105146564)

[1.7 Calibrate the autopilot 8](#_Toc105146565)

[2 Acoustic tracking : 11](#_Toc105146566)

[2.1 Setting up the Raspberry OS 11](#_Toc105146567)

[2.1.1 Setting up the companion software 11](#_Toc105146568)

[2.1.2 Setting up the waterlinked receiver board 11](#_Toc105146569)

[3 Bathymetric survey 12](#_Toc105146570)

[4 Photogrammetric survey : 13](#_Toc105146571)

# Global

## Necessary software

|  |  |  |
| --- | --- | --- |
| Name of the software | Compatible platform | Link |
| Mission planner | Windows | https://firmware.ardupilot.org/Tools/MissionPlanner/MissionPlanner-latest.msi |
| Modem Tools | Windows | http://files.rfdesign.com.au/tools/ |
| Firefox | Windows/linux | https://www.mozilla.org/fr/firefox/new/ |
| Rufus | Windows | https://rufus.ie/fr/ |

## Establish a connection between the GCS and the autopilot through telemetry

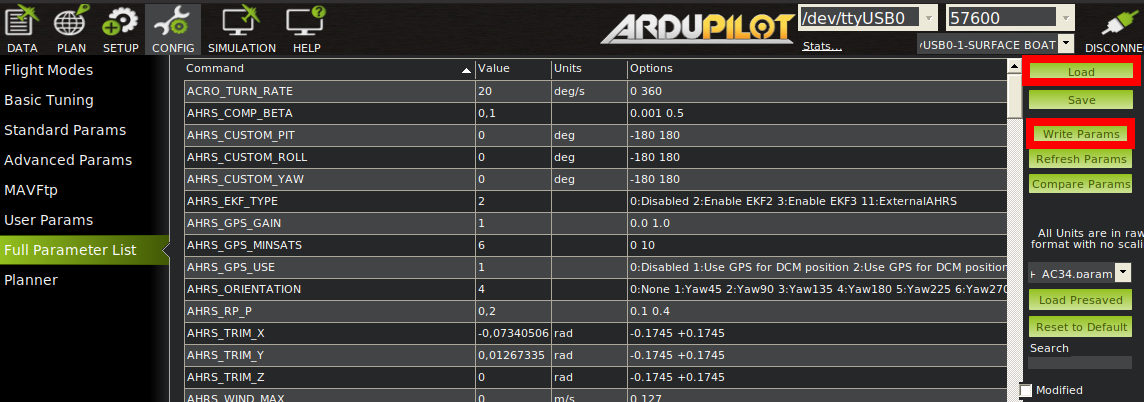
* + Plug a SD card into the Pixhawk
  + Power the board by plugging in the battery
  + Load the rover firmware onto the board following the instructions at https://ardupilot.org/planner/docs/common-loading-firmware-onto-pixhawk.html
  + In order to check if the autopilot works fine, the telemetry link is essential. We thus need to set up the RFD900 module as a first step. The following tutorial lays out the firmware upload, the definition of the parameters and the connection:

<https://mikeisted.wordpress.com/2018/08/25/rfd868x/>

* Power your PC, download Mission planner from ”https://ardupilot.org/rover/docs/common-install-gcs.html”,  plug in another  RFD900  with  a  FTDI  connector  and  try  and  connect  to  the  pixhawk  by selecting the right port and 57600 baud.

## Upload of the ardupilot parameters

* + Go to “Config”/”Full parameters list”/”Load”, select the .param file from your pc and click “Write Params”



* Wait for the operation to complete and reboot the pixhawk

## Test the communication between the RC and the autopilot and calibrate it

* + On Mission planner, go to ”Initial setup”/”Mandatory Hardware”/”RC”
  + Check that green bars appear and move as you move joysticks and buttons around on the RC
  + Click on ”Calibrate RC” and follow the procedure displayed here:

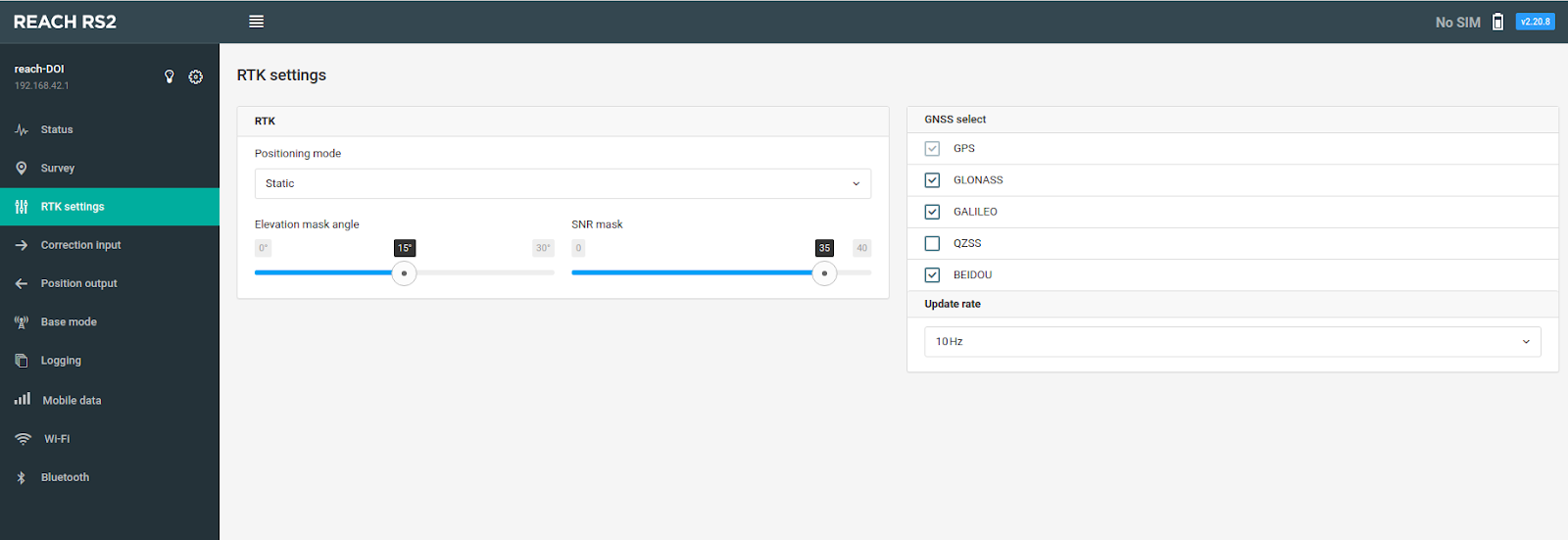
<https://ardupilot.org/rover/docs/common-radio-control-calibration.html>

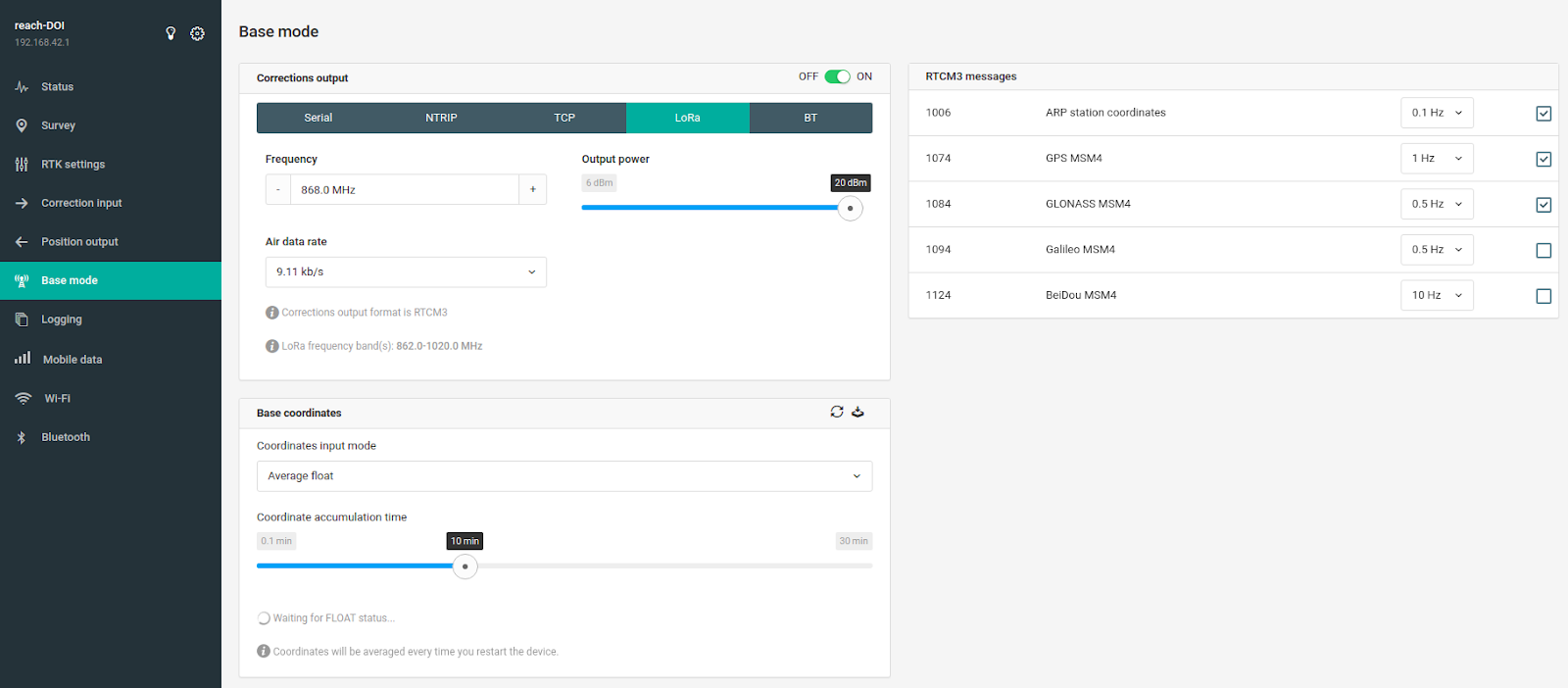
## Test the reception of a GPS signal

* + Plug your setup on battery and go to the open air.  Make sure no obstacle is to be seen in a 30° angle above the GNSS antenna
  + Open Mission planner and check out the GPS status on the top left hand corner of the Data page.
  + It should read ‘3D fix’.  If no GPS is found by ardupilot, double check the connections between the GNSS module and the autopilot.

## Configure the RTK GNSS

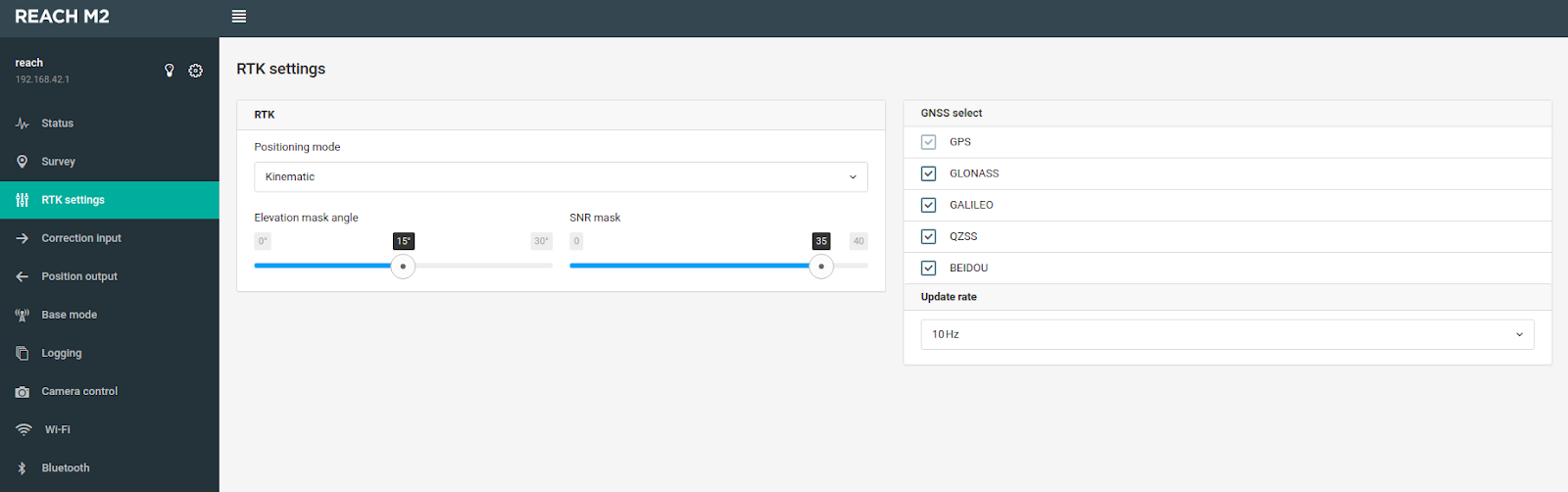
* + Sit your RS2 a few tens of meters away, screw the antenna on and turn it on
  + Once again, make sure you are in the open air and no obstacle covers the sky view above the M2 and RS2 module
  + Turn on the wifi on your computer or phone and connect to the wifi network generated by the RS2
  + Connect to the RS2 GUI by typing its IP address in the research bar of a browser and set the parameters as displayed below:

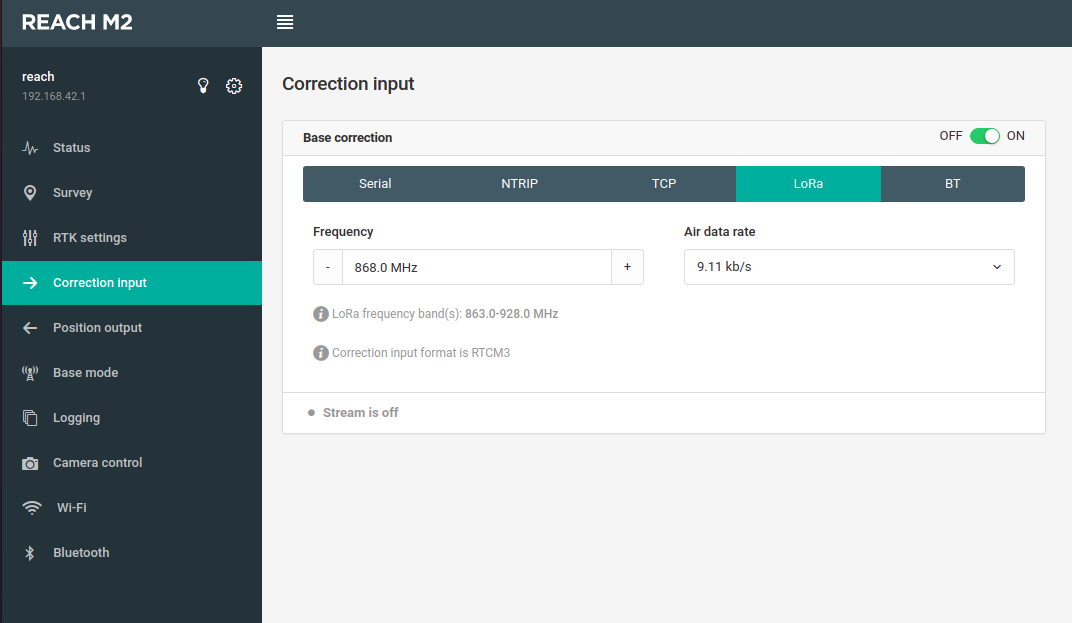


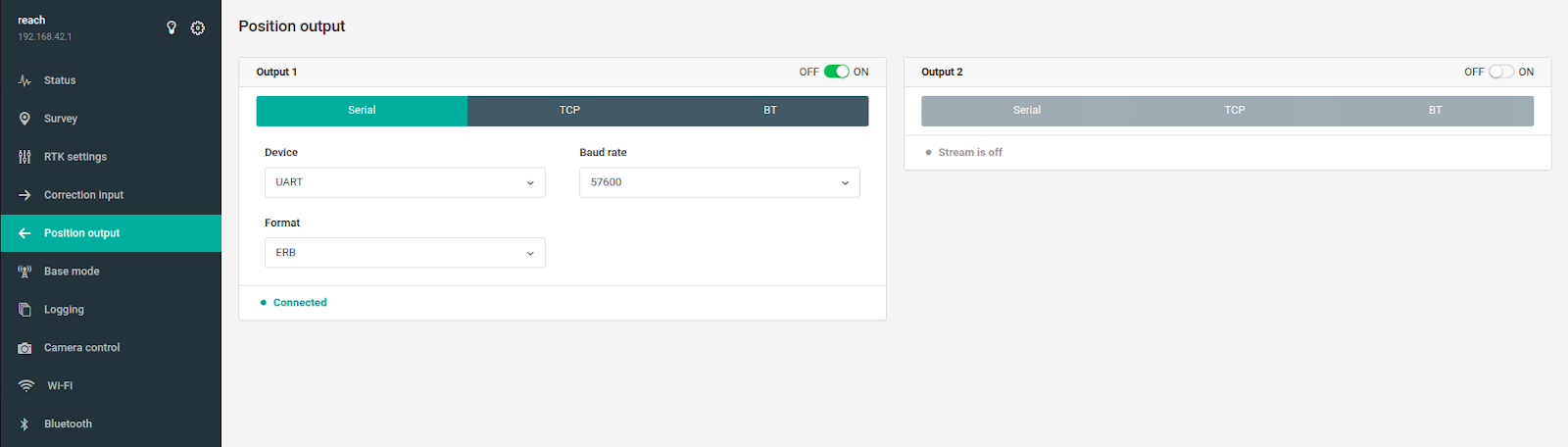


Connect to the M2 GUI in the same way and set the following parameters

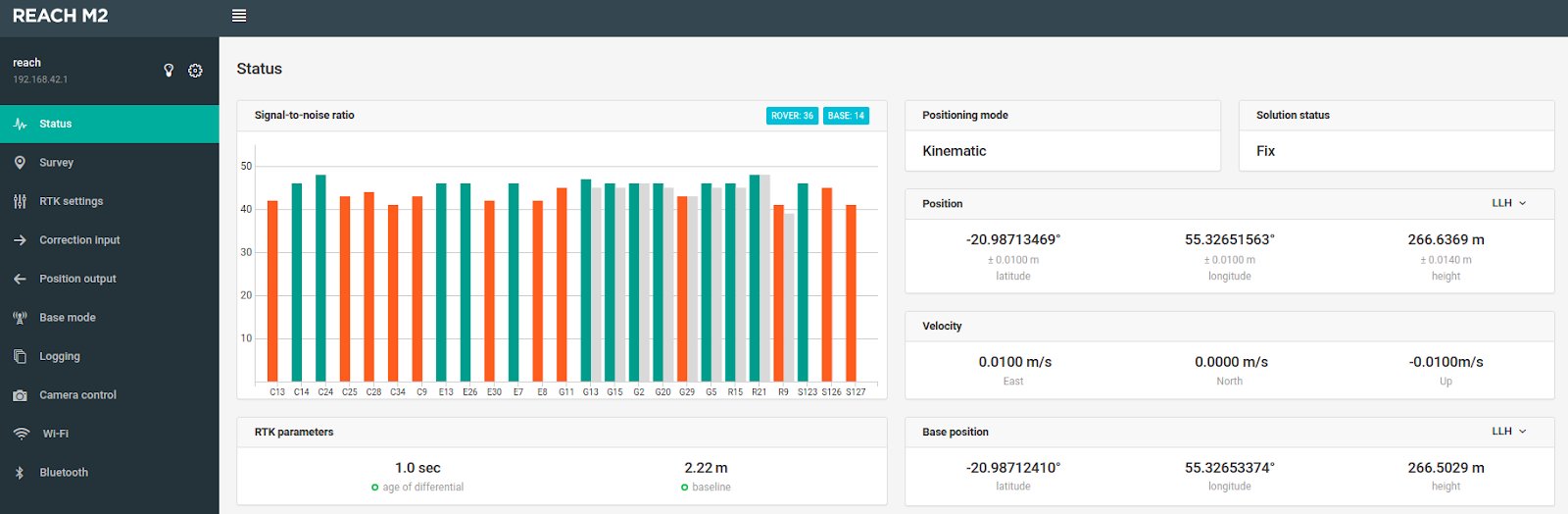








* Get back to the main tab on the M2 GUI and have a look at the signal bars diagram.  If everything is set correctly, you should see grey bars next to the color bars.



Then check the signal status window below the diagram. This one should read RTK FIX.

* As a final check, open Mission planner and verify that the GNSS status is RTK FIX too.  At this point, RTK is operational.



## Configure the RC

* + Power up the electronics by plugging in the battery
  + Turn on the RC
  + Use a pencil to push the receiver side button. It starts flashing blue and purple.
  + When it turns still, the RC is paired.
  + Now that the communication is established between ardupilot and the RC, we need to ensure that the right command triggers the right reaction from the thrusters. Power off the board, unplug the thrusters (you can simply pull off the fuses), open mission planner and follow the following tutorial:

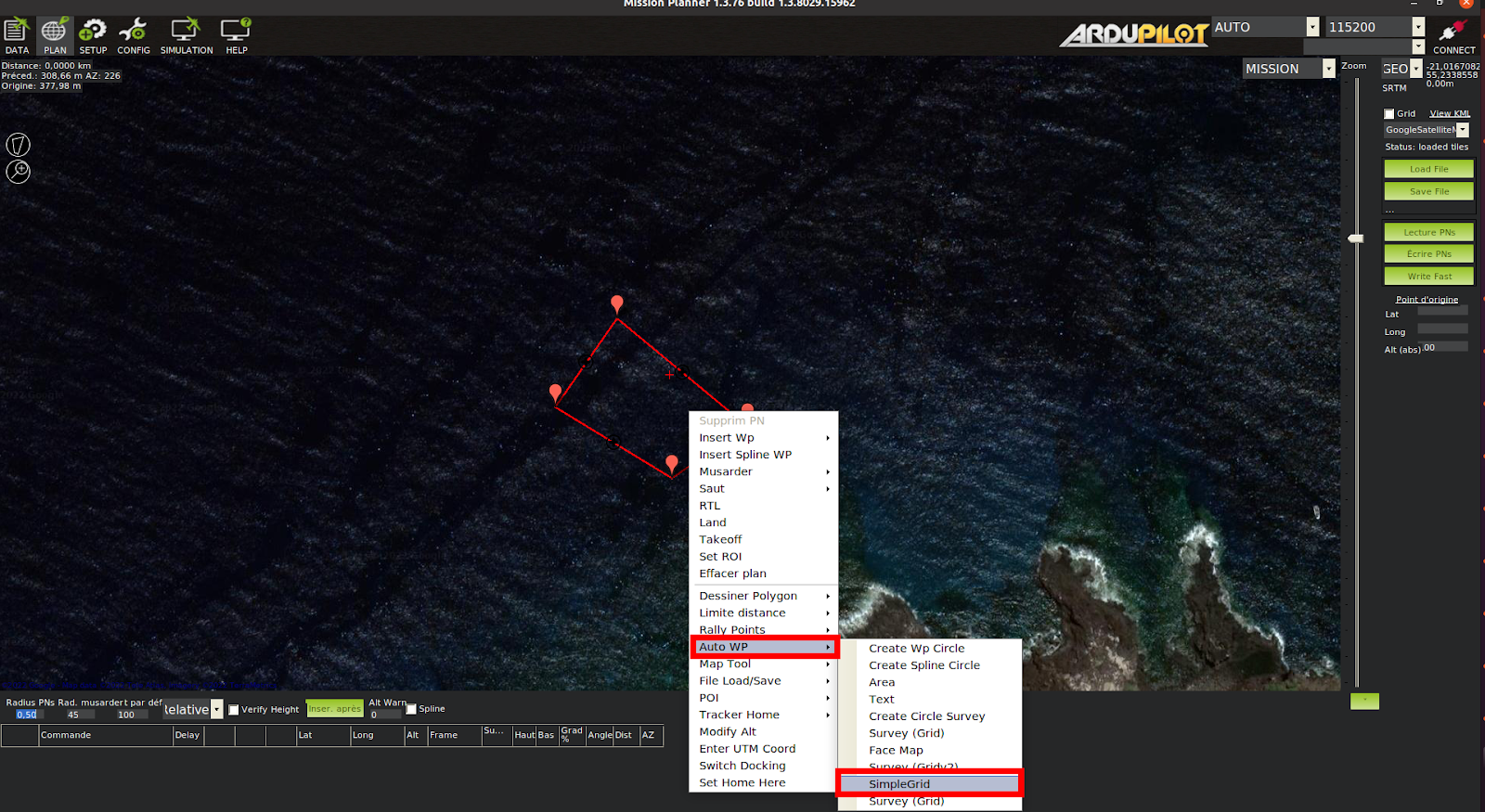
<https://ardupilot.org/rover/docs/rover-motor-and-servo-configuration.html>

* Dip the thrusters into a pool of water. Move each stick separately and check the right reaction of the thrusters. For instance, as you move the left stick up, you should witness a backward waterjet. on both sides of the board.

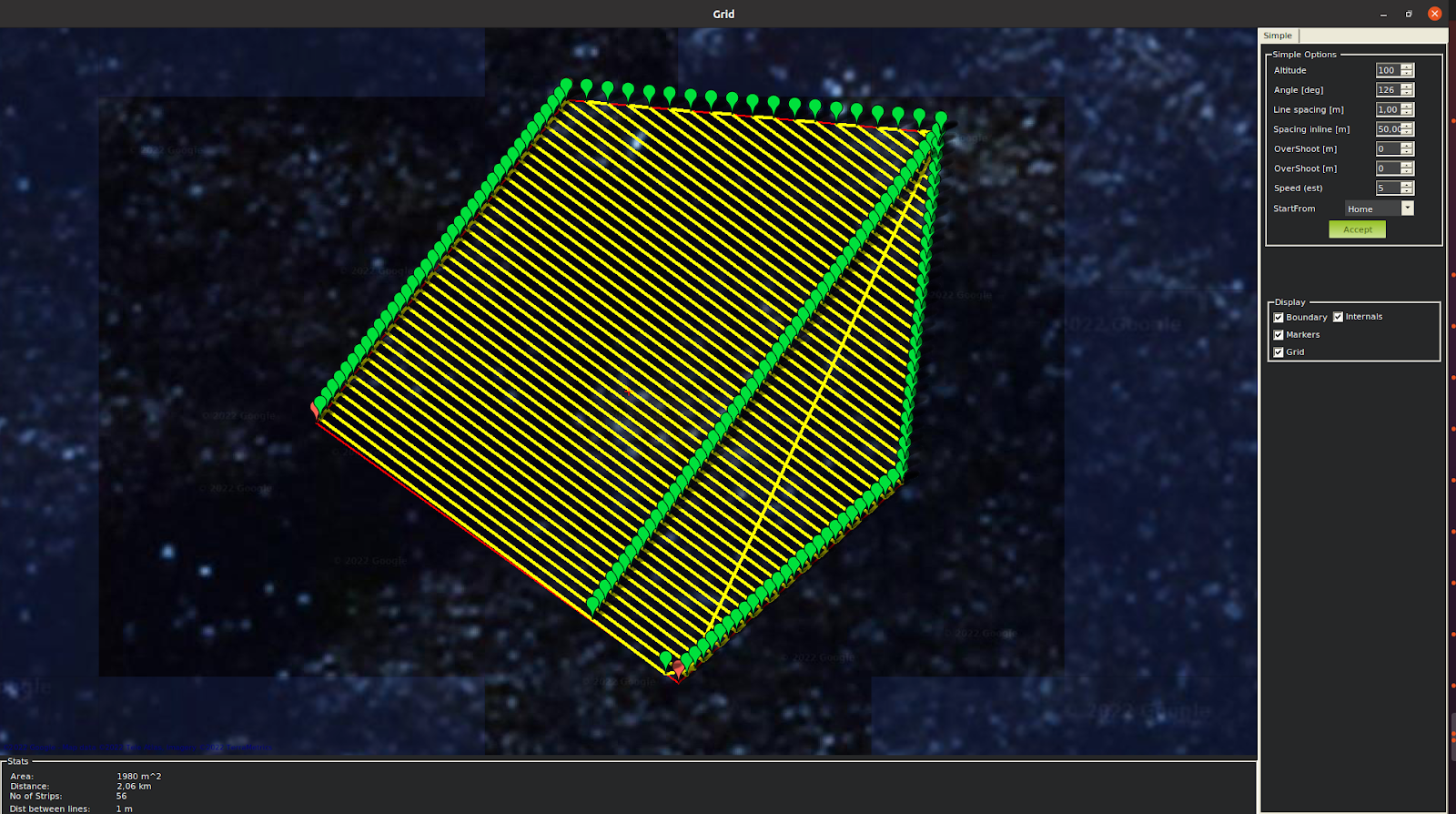
## Calibrate the autopilot

* + To perform this calibration, you will need to drive the board along transects on a large surface of water.
  + To create those transect, please visit the official ardupilot page: <https://ardupilot.org/planner/docs/mission-planner-flight-plan.html>

As for our missions, we start by drawing polygons:



Set the line spacing at 1m or 2m for the grid. This will force the board into tight turns that will evidence the possible overshoots at the end of the lines.





* Power up the board
* Establish a connection through telemetry with the board from your PC
* Test the right functioning of the engines and the RC
* Put the board to the water
* Go through the calibration guide on the ardupilot page <https://ardupilot.org/rover/docs/savetrim.html> . If your board architecture is identical to ours, your parameters should not differ much from our own

# Acoustic tracking :

The acoustic tracking is performed thanks to an exchange of informations between the Raspberry Pi (companion computer), the pixhawk autopilot and the waterlinked D1 receiver.

The companion computer’s job is to request the position from the autopilot and feed it to the waterlinked receiver. It requires to be loaded with the propre software to complete that operation.:

## Setting up the Raspberry OS

* Load a microSD card and of at least 8Go with a Raspian OS. This can be done simply by using the “restore image” feature on ubuntu and selecting the SD card reader and the proper image on your computer. Alternatively, on windows, download the latest Rufus software.
* Plug the micro SD card into the Raspberry and power it on.
* Wait for the boot to complete

### Setting up the companion software

* Download the companion software from this articles files
* Create a folder called “/pi” in the home directory of the raspberry and place the unzipped content of the companion software.
* Follow the instructions on the Readme.md file to install the dependencies
* Launch the command “$source activate idocean”, then “$./idocean\_script”

### Setting up the waterlinked receiver board

* Power on the waterlinked board
* A hotspot should be broadcast that you can connect to. Password: “waterlinked”.
* Open a web browser and connect to the GUI by typing the following IP on the research bar: “192.168.7.1”

# Bathymetric survey

* **Configure the echosounder as presented in h**[ttps://ardupilot.org/copter/docs/common-echologger-ect400.html](https://ardupilot.org/copter/docs/common-echologger-ect400.html) **using either a terminal or the dedicated software.**
* The echosounder data will automatically be logged into the ardupilot dataflash log and displayed in real time in the Mission planner interface under “status” tab in the “Data” page under the “sonarrange” denomination

# Photogrammetric survey :

* Turn on the Gopro camera by first plugging in the USB C and then pushing the power button on the side of the gopro
* The configuration of the gopro should be as follows:
  + RES | IPS : 1080 | 60
  + Objectif : Linéaire
  + HyperSmooth : Boost
  + Faible luminosité : Non
  + Zoom : x1,0
  + Clips : non
* Débit : Standard
* Obturateur : Auto
* Comp V.E : 0
* Bal.blancs : Auto
* Iso min : 100
* ISO max : 6400
* Netteté : Élevé
* Couleur : GoPro
* Audio RAW : NON
* Vent : Auto