|  |  |  |  |
| --- | --- | --- | --- |
| Artifact ID:  GD-002 | Artifact Title:  QGroundControl Guide | |  |
| Revision:  1 | Revision Date:  2025-03-05 | |
| Prepared by:  Jadyn Christensen | | Checked by: | |
| Purpose:  Instructions for using QGroundControl with a Pixracer Pro on both the High Altitude and Low Altitude Drone | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Revision History | | | |
| Revision | Revised by | Checked by | Date |
| 01 | Jadyn Christensen |  | 2025-03-05 |

# **QGroundControl**

Initial Setup and Usage Guide

**Author**: Team 56 “Snowflake”  
**Date**: 2025-03-05

Table of Contents

[**QGroundControl** 2](#_Toc193785728)

[Flash Pixracer Pro 4](#_Toc193785729)

[Choose the vehicle type 6](#_Toc193785730)

[Binding radio controller to onboard receiver 7](#_Toc193785731)

[Output Channels 7](#_Toc193785732)

[Sensor Calibrations 7](#_Toc193785733)

[Flight Modes 7](#_Toc193785734)

[Safety Parameters 7](#_Toc193785735)

[Other Parameters that may become useful 7](#_Toc193785736)

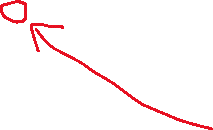
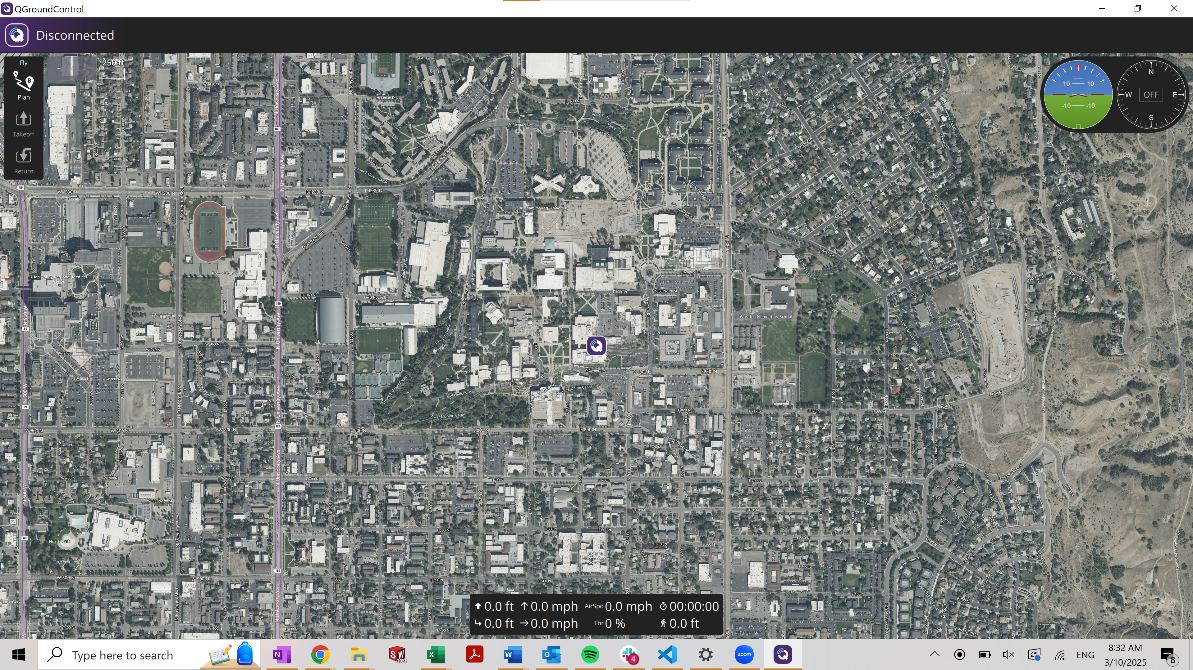
[Configure firmware to use ROS 2 7](#_Toc193785737)

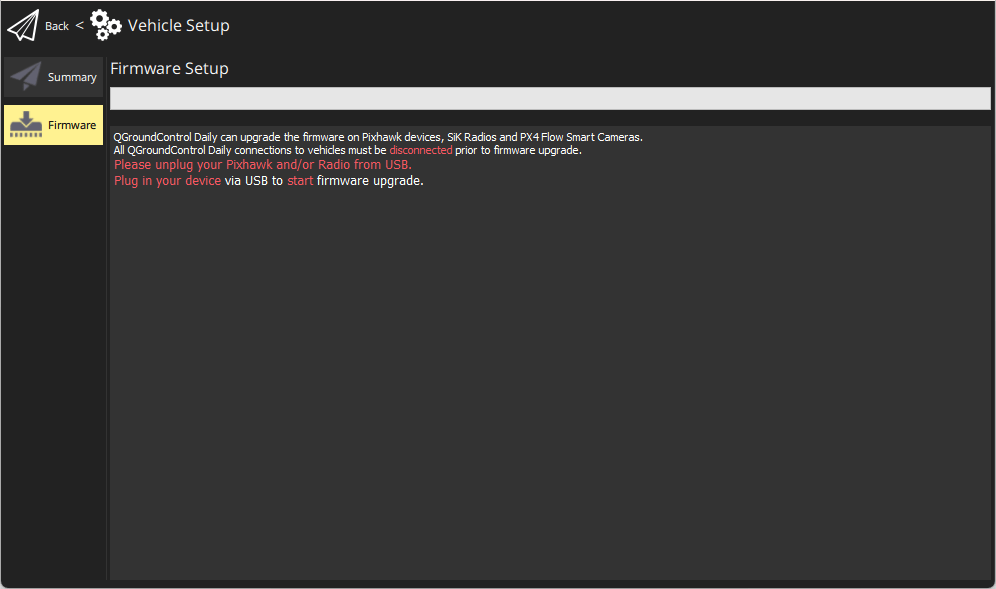
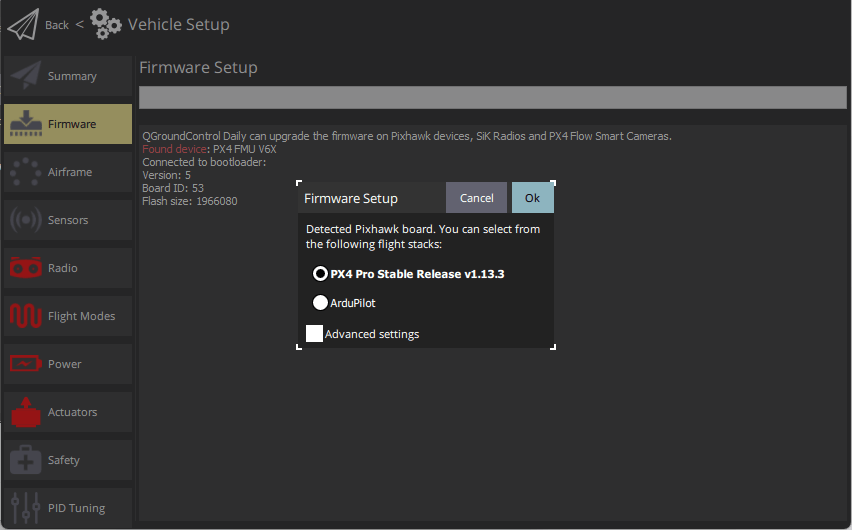
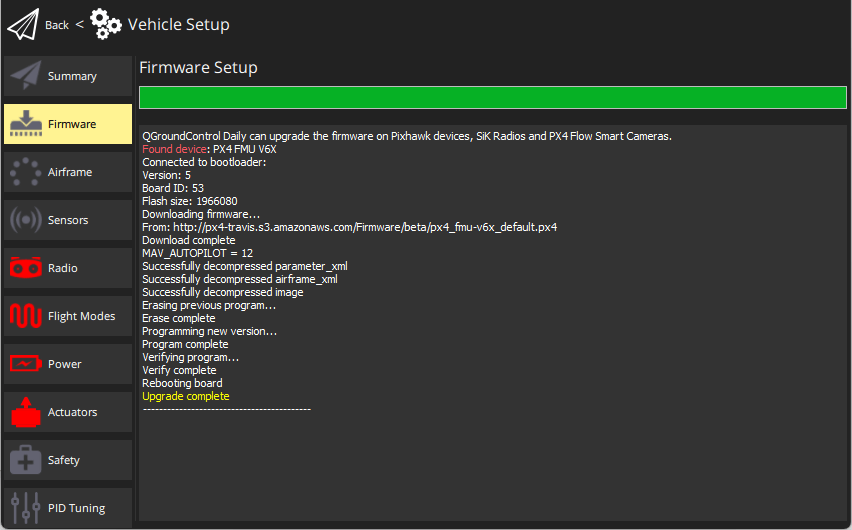
[Appendix 8](#_Toc193785738)

# Flash Pixracer Pro

Download QGroundControl onto a computer you will use to connect to your PixRacer Pro to download the firmware: <https://docs.qgroundcontrol.com/master/en/qgc-user-guide/getting_started/download_and_install.html>

1. Start *QGroundControl*
2. The splash screen appears as below. To access the options menu, click the Q icon in the top left corner of the screen.



1. Select **"Q" icon > Vehicle Setup > Firmware** (sidebar) to open *Firmware Setup*
2. Connect the flight controller directly to your computer via USB
3. Select the **PX4 Pro Stable Release vX.x.x** option to install the latest stable version of PX4 *for your flight controller* (autodetected)
4. Click the **OK** button to start the update. The firmware will then proceed through a number of upgrade steps (downloading new firmware, erasing old firmware etc.). Each step is printed to the screen and overall progress is displayed on a progress bar.
5. Once the firmware has completed loading, the device/vehicle will reboot and reconnect.

# Choose the vehicle type

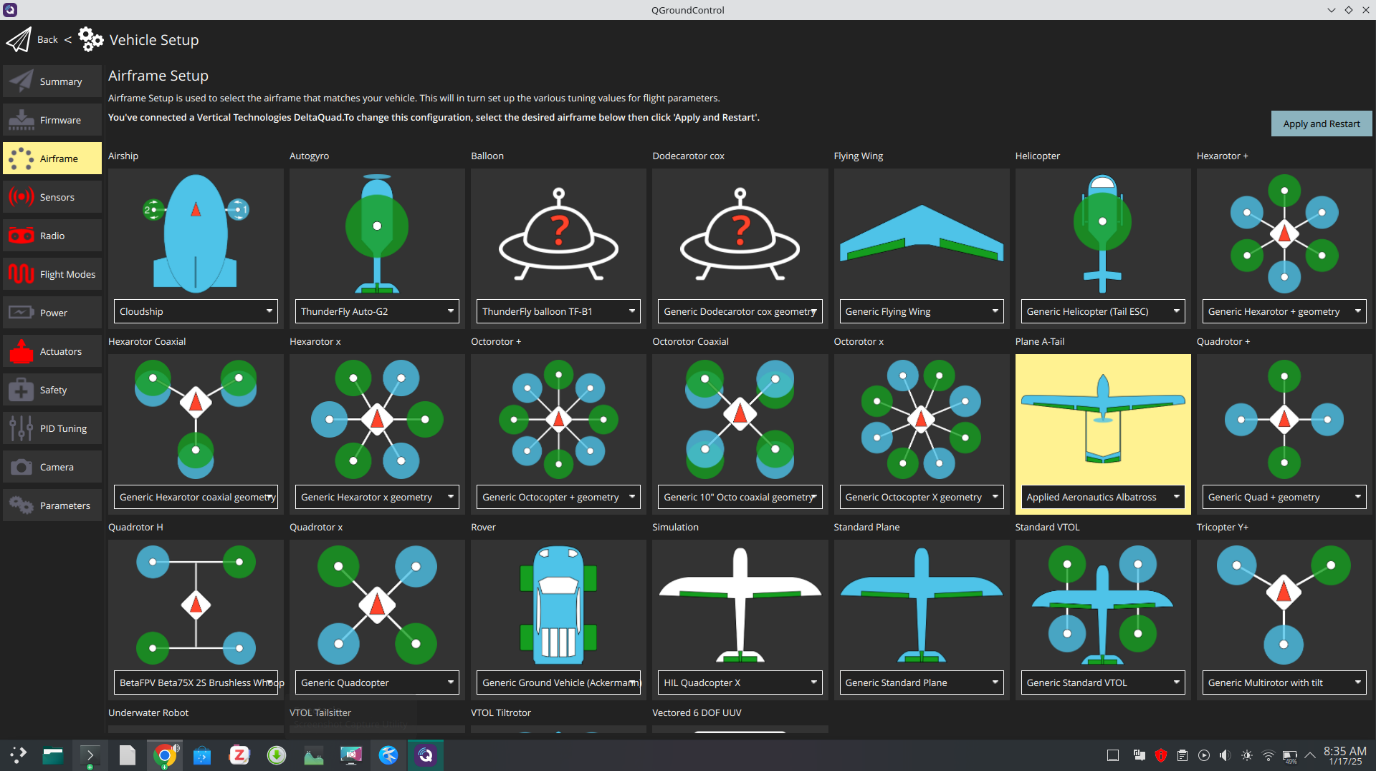
Now that the flight controller is flashed, you can connect to the drone either through USB-C connection or wirelessly through a ground radio as long as the radio is connected. See ART-001 and ART-009 for wiring diagrams.

You now need to select a vehicle type and frame configuration. The different options can be found here: <https://docs.px4.io/main/en/airframes/airframe_reference.html>

For this project, we used the “Applied Aeronautics Albatross” for the high-altitude drone because it matches the a-frame tail design of the Anaconda. We used “Generic Standard VTOL” for the low altitude drone.

The steps to set this up are below

1. Select "Q" icon > Vehicle Setup > Airframe (sidebar) to open Airframe Setup.
2. Select the broad vehicle group/type that matches your airframe and then use the dropdown within the group to choose the airframe that best matches your vehicle.

The example above shows *Applied Aeronautics Albatross* selected

1. Click Apply and Restart. Click Apply in the following prompt to save the settings and restart the vehicle.

# Sensor Calibrations

On the sensor tab in Qgroundcontrol prior to each test flight ensure that you precede through the accelerometer, compass, level horizon, and gyroscope. **We suggest that these be performed before every flight.**

1. Follow the instructions on each test. They are very clear and self explanatory.

It’s also important to do an ESC calibration. That is found in the power tab, not the sensor tab.

1. Unplug battery, connect to flight controller via USB.
2. Hit start test.
3. Plug in battery.

This test tends to be extremely sensitive to how quickly one connects the battery. We suggest that there be 2 people, one to hit “start test”, and a second to immediately connect a battery.

# Binding radio controller to onboard receiver

We followed this guide here to bind the receiver and the radio: <https://www.expresslrs.org/quick-start/binding/>

Once it is bound, ensure that the drone is in communication with qgroundcontrol by going to the radio tab. You should be able to see inputs making a difference in the levels.

# Output Channels

On the model tab, you’ll see that there are many options. Here, each output channel (i.e. left aileron, pusher motor, etc.) need to be associated with an input command. This takes a little bit of a guess and check.

1. Figure out which input channels are associated with which movement of the radio. Pushing the left joystick left and right is one channel, up down is another, and the same for the right joystick. Each switch on the radio is also associated with one.
2. Assign each input channel to an output. Our outputs are shown below.
3. This is also the section where you can set servo ranges. Find what PWM value is the center of each servo travel, and set that as the center of the servo range.

# Configure firmware to use ROS 2

The PX4 firmware now needs to be configured to use ROS 2 instead of MAVLINK. To learn how to find and update parameters, see this guide: <https://docs.px4.io/main/en/advanced_config/parameters.html>.

1. To configure the ROS2 connection, set the following configurations to their respective settings.

A screenshot of a phone

AI-generated content may be incorrect.

1. MAV\_1\_CONFIG=0 disables MAVLink on TELEM2 and UXRCE\_DDS\_CFG=102 enables the uXRCE-DDS client on TELEM2. SER\_TEL2\_BAUD rate sets the comms link data rate.
2. Check that the uxrce\_dds\_client module is now running. You can do this by running the following command in the QGroundControl MAVLink Console

uxrce\_dds\_client status

1. If the client module is not running you can start it manually in the MAVLink console:

uxrce\_dds\_client start -t serial -d /dev/ttyAMA0 -b 921600

# Appendix

**Useful links divided by section**

Flash SD Card with Ubuntu 24.04

* <https://ubuntu.com/tutorials/how-to-install-ubuntu-desktop-on-raspberry-pi-4#2-prepare-the-sd-card>

Boot Ubuntu Desktop and Set Up SSH

* <https://averagelinuxuser.com/how-to-install-and-use-ssh-on-linux/>

Enable UART0 on RPi

* <https://www.raspberrypi.com/documentation/computers/configuration.html#cm1-cm3-cm3-and-cm4>

ROS 2 “Jazzy” Installation & ROS 2 Workspace Setup

* ROS 2 installation and workspace setup: <https://docs.ros.org/en/jazzy/Installation/Ubuntu-Install-Debs.html#install-ros-2>
* ROS 2 examples: <https://github.com/ros2/examples>

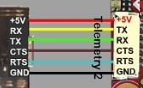
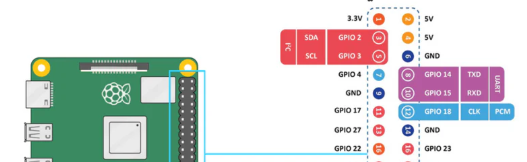
PixRacer Pro Setup

* Qgroundcontrol firmware installation: <https://docs.px4.io/main/en/config/firmware.html#install-stable-px4>
* Qgroundcontrol vehicle selection: <https://docs.px4.io/main/en/config/airframe.html>
* Airframe type reference: <https://docs.px4.io/main/en/airframes/airframe_reference.html>
* How to change parameters in Qgroundcontrol guide: <https://docs.px4.io/main/en/advanced_config/parameters.html>
* TELEM 2 parameter config: <https://docs.px4.io/main/en/companion_computer/pixhawk_rpi.html#ros-2-and-uxrce-dds>

uXRCE\_DDS Agent Installation & uXRCE\_DDS Agent Activation

* <https://docs.px4.io/main/en/companion_computer/pixhawk_rpi.html#ros-2-and-uxrce-dds>

RPi and PixRacer Pro Wiring

* RPi and PixRacer pro wiring: <https://docs.px4.io/main/en/companion_computer/pixhawk_rpi.html#wiring>
* PixRacer Pro hardware layout: <https://docs.3dr.com/autopilots/pixracer-pro/#downloads>
* 
* 

**Extra links**

ROS 2 “Jazzy” Integration with PX4

* ROS 2 Integration with PX4: <https://docs.px4.io/main/en/ros2/>
* uXRCE-DDS (PX4-ROS 2/DDS Bridge): <https://docs.px4.io/main/en/middleware/uxrce_dds.html>
* uORB Message Reference:
* <https://docs.px4.io/main/en/msg_docs/>
* ROS 2 User Guide (with PX4): <https://docs.px4.io/main/en/ros2/user_guide.html>
* PX4-Autopilot/IntegrationTests: <https://github.com/PX4/PX4-Autopilot/tree/main/integrationtests/python_src/px4_it/mavros>

PX4

* PX4 Autopilot User Guide: <https://docs.px4.io/main/en/>
* MAVLink Messaging: <https://docs.px4.io/main/en/middleware/mavlink.html#mavlink-overview>
* UAV Data Transmission and Protocols PowerPoint: <https://robolabor.ee/img/cms/projektid/UAV%20Data%20Transmission%20and%20Communication%20Protocols.pdf>
* General PixRacer Documentation: <https://bkueng.gitbooks.io/px4-user-guide/content/en/flight_controller/pixracer.html>

# Troubleshooting

Occasionally (haha maybe more than that) the Pixhawk enters strange states. In this case, we suggest trying to reflash the firmware.

The MAGICC lab was an incredibly helpful resource for setting up drones.