Openpilot CC3D Revo 32F4 Setup

Introduction

Welcome to the Revolution (Revo) board setup page. Along with the tutorials you will also find links to the relevant Wiki pages to assist you in setting up the best Flight Controller board on the market.

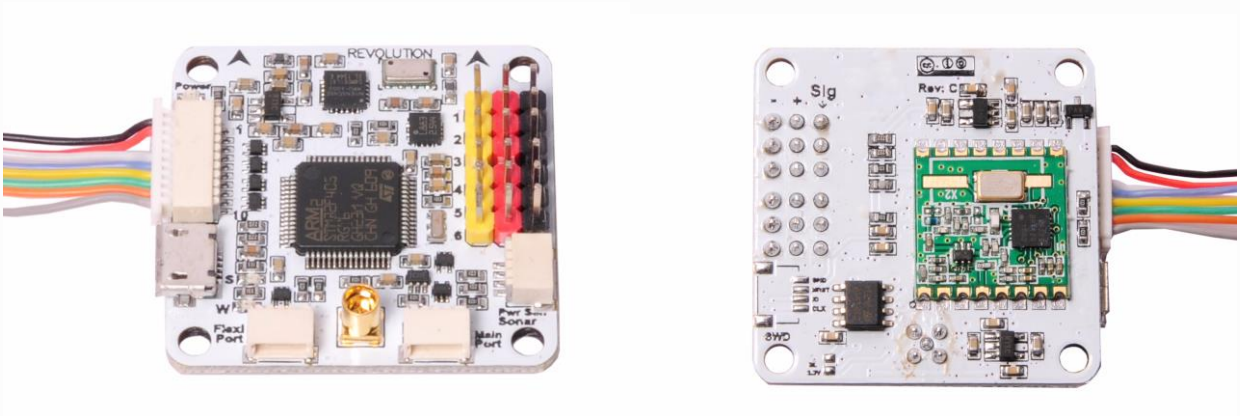
The OpenPilot Revolution board, also called 'Revo', is a new breed of Autopilot using the **STM32F4 series**, 210MIPS ARM Micro-controller. This is important, as it contains a hardware floating point unit (FPU), which is a huge advancement for the hobby-class autopilots. Of course, OpenPilot has been 32bit since day one, and the FPU is another step up the performance ladder. The FPU allows precise, low-latency processing of real-life measurements using advanced attitude estimation algorithms.

The Revolution is a flight control computer with autopilot, intended for multirotors, helicopters and fixed wings. It is a full 10DOF with gyroscope, accelerometer, magnetometer and pressure sensors.

Features:

- 3-axis Gyroscope array and 3-axis Accelerometer: MPU-6000
- Supports several common RC inputs: 6 PWM channels, combined PPM, Spektrum/JR DSM2, DSMJ, DSMX satellites, and Futaba S.Bus receivers
- Simultaneous support for multiple receivers
- Receiver Port functions (configurable): 6 PWM input channels or combined PPM stream, 4 PWM output channels
- MainPort functions (configurable): serial telemetry (default), GPS, S.Bus, Spektrum/JR satellites
- FlexiPort (configurable): serial telemetry, GPS, Spektrum/JR satellites, or I2C peripherals (under development)
- 10 PWM outputs to servos or ESC's, or for camera stabilization
- Camera stabilization: supports up to 3-axis camera mounts with stabilization and manual control from any of configured receivers
- Onboard USB connectivity for easy configuration
- USB and serial telemetry and configuration (including wireless with optional radio modules)
- Supported by powerful OpenPilot GCS
- 4 Mbit onboard memory
- 3C Quaternion based complementary filter running at 500Hz

Front and Rear views of CC3D boards



Flexiport
CopterControl also offers the innovative Flexi-port which provides either I2C connectivity or a second serial port. Flexi-port gives CopterControl the flexibility to use either I2C and a single serial port or dual serial ports, according to your connectivity needs.

Superior user experience
OpenPilot has tried to make your experience with the CC3D as easy and trouble-free as possible. Naturally for a high-end product, the hardware is complete; there is nothing to solder. The included RC harness allows you to connect any standard PWM receiver. All CopterControl boards are tested before shipping and come with bootloader already flashed. A standard mini-USB cable (not included) is used to load the latest firmware and configure everything for your unique airframe.

The OpenPilot project, whist still young, it trying to make Open Source much more user friendly. OpenPilot software is designed to be used by everybody! All major operating systems are supported and install packages are provided. The software is all Free software under the GPLv3 license and not only are you welcome to hack it, but we encourage it!

Setting up your Revolution Board for the first time

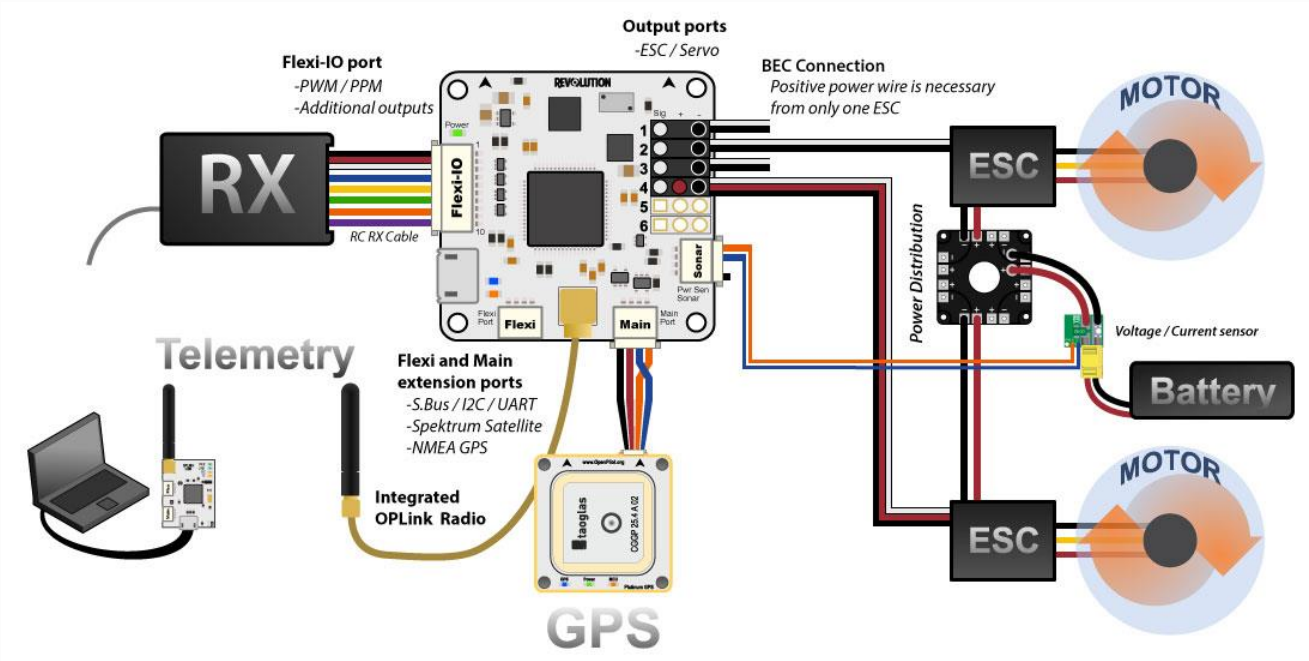
You have just received a brand new Revolution board and are itching to mount it in your air frame, follow the instructions below.

Vehicle Setup Wizard

Once you have mounted your Revo on your frame you need to configure it through the Ground Control Station (GCS) using the Vehicle Setup Wizard, follow the **Tutorial** below for setting up on a Multirotor.

Connection diagram

The diagram below summarizes how the overall Revolution system is connected.



Technical description

CPU (Central Processing Unit)

CPU is the STM32F405RGT6 chip, with ARM Cortex-M4 core at 210MIPS, FPU, and saturation arithmetic's DSP functions.

The chip features a range of built-in hardware modules that can be programmed once and function independently, requiring little to no CPU overhead. These include 14x multichannel timers, 3x synchronous-sampling ADC serving up to 24 channels, 2x DAC, matrix memory

controller with 16-stream DMA, and other. Communication modules include USB2.0, 3x I2C, 3x SPI, 4x USART, 2x CAN and SIDO. All these modules can be configured for accessing the chip pins using a flexible switch matrix, or disabled to save power.

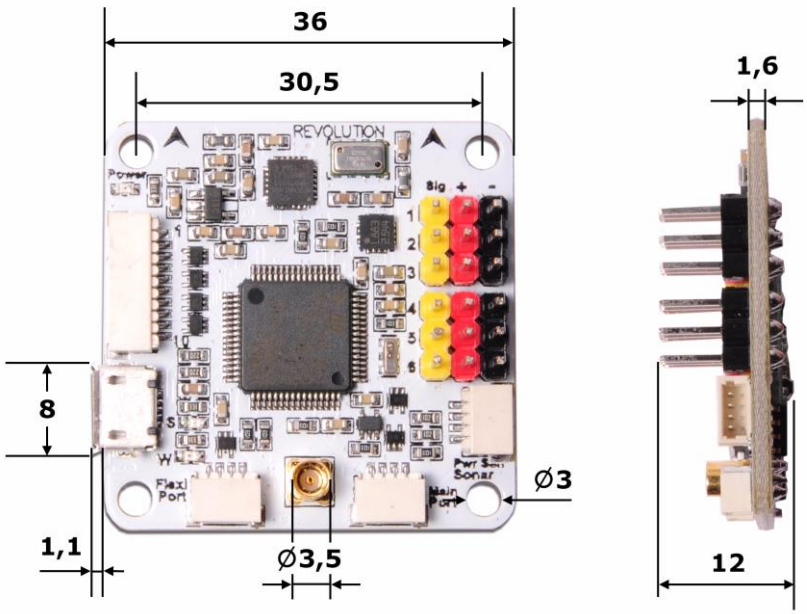
It even contains a real time hardware calendar if you want a wake up flight. The software and settings are loaded through USB connector and no-hassle update function in the GCS (Ground Control Station).

Modem

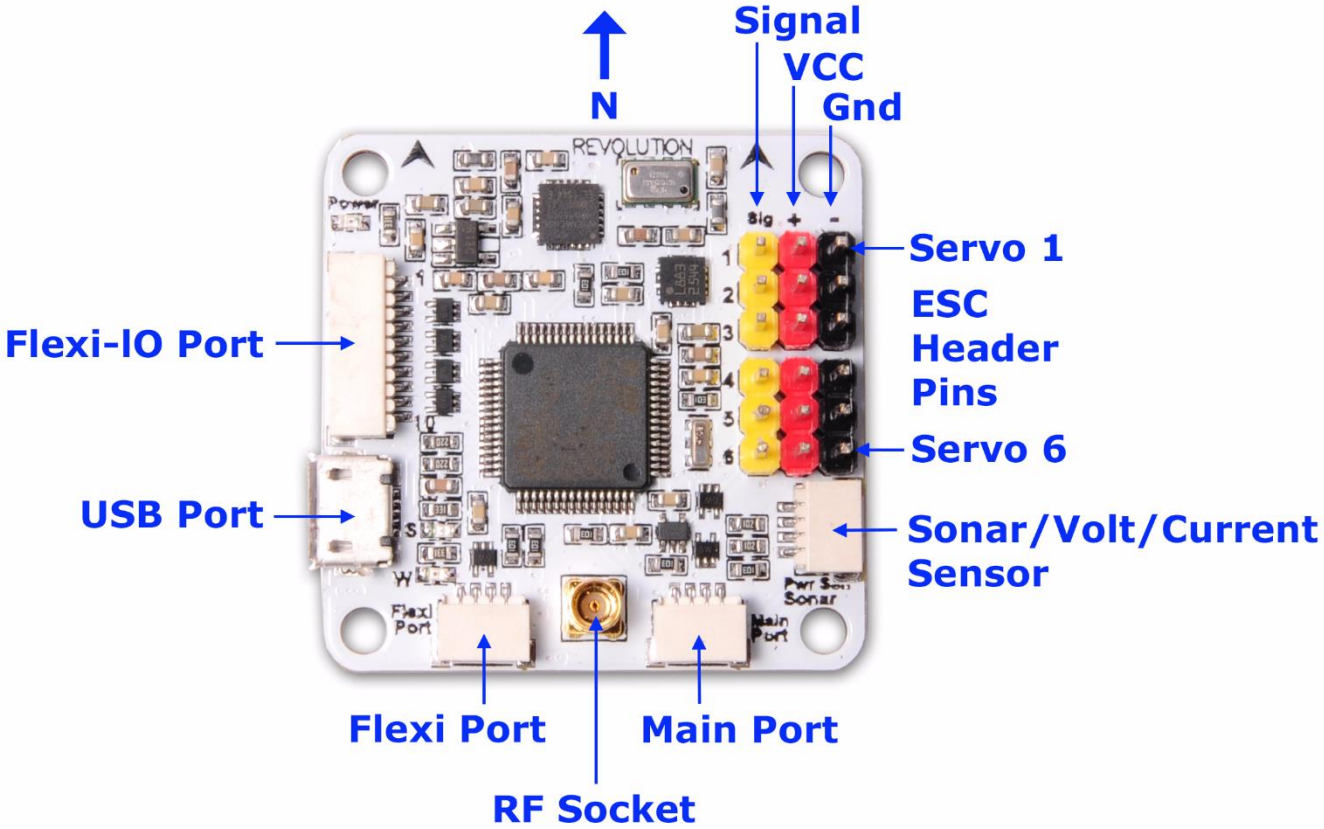
The board features built-in 433MHz OPLink Modem.

Dimensions

OpenPilot products use the standard OpenPilot footprint, and hence has the same dimensions and mounting holes as the OpenPilot Revo, GPS, OSD and PipX boards. (All dimensions are in millimeters.)



Port Layout of CC3d board



- **Servo 1-6**: These are the PWM outputs that go to servos or ESCs. Power is typically applied through these headers from only one of the ESCs. The positive (Vcc) and negative (Gnd) pins are indicated on this diagram and the board.

  Servo output pin layout is:

  - Outside –> ground
  - Middle –> 5V - 8.4V
  - Inside –> signal
- **Flexi-IO Port**: JST-SH 10-pin. The receiver port can act as an input or output port depending on the configuration which is set in the Hardware Settings. Configuring the receiver port as an output port allows the user to assign more output channels then the 6 standard servo outputs.
- **PWM -vs- PPM Receivers**

  Please be aware that not all receivers can be configured to use a PPM output. It is the user's responsibility to research this feature in regards to the desired receiver they wish to use for PPM and ensure it can be used as such. Many hours of frustration can occur while trying to troubleshoot why you can't get your radio to connect to the board with PPM if

using a receiver that isn't designed with that feature! Simply make sure the receiver can do it before trying to set it up that way.

- **MainPort**: JST-SH 4-pin. This is a serial USART whose baud rate can be adjusted through the GCS. Optionally, Futaba S. Bus receiver, Spektrum/JR satellite receiver or GPS can be mapped to the MainPort. Default configuration is Telemetry for connecting an RF modem.
- **Flexi Port**: JST-SH 4-pin. The function of this port also depends on the configuration and can be configured for I2C or Serial. The default configuration doesn't use this port, but it can be used for Telemetry, GPS, Spektrum satellite receivers (all working), and other I2C peripherals (under development).
- **RF Socket**: Antenna connection socket for on-board *OPLink Modem*.
- **Pwr Sen/Sonar Port**: JST-SH 4-pin. This port can be configured to accommodate an Autopilot current sensor and a low cost Sonar sensor such as the HC-SR04. It can also be used as a general purpose input/output port or as a one or two channel analog input port.

Note

Please note that the output rate on the output channels from the Receiver Port cannot be set individually. If servos are connected to this outputs, you must ensure that they can work with the defined output rate for choose a high output rate to support an octocopter configuration, the update rate from the output channels from the Receiver Port are bound to the update rate from channels 5 & 6. In this case, you cannot connect analog servos to these outputs since an analog servo only supports an output rate of 50Hz.

Sensor suite

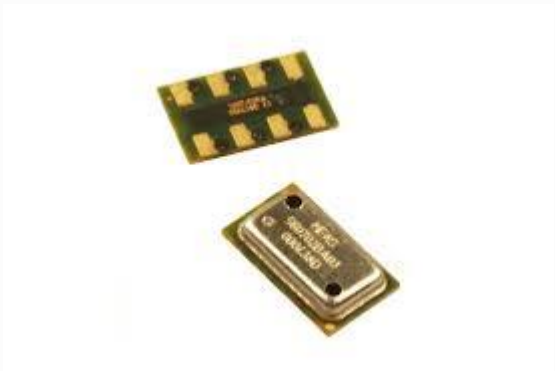
The CC3D is equipped with following sensors

- 3 Axis Gyro
- 3 Axis Accelerometer
- 3 Axis Magnetometer
- Barometric pressure sensor

MPU (Motion Processing Unit)

The MPU-6000 combines a 3-axis gyroscope and a 3-axis accelerometer on the same silicon die. This sensor can also be found on the CC3D and already has a proven track record of great flight performance.

Pressure Sensor/Altimeter



When it came time to select a barometric pressure sensor, there were many to choose from and many were not up to the task. For the Revolution, OpenPilot selected the Measurement Specialties MS5611.

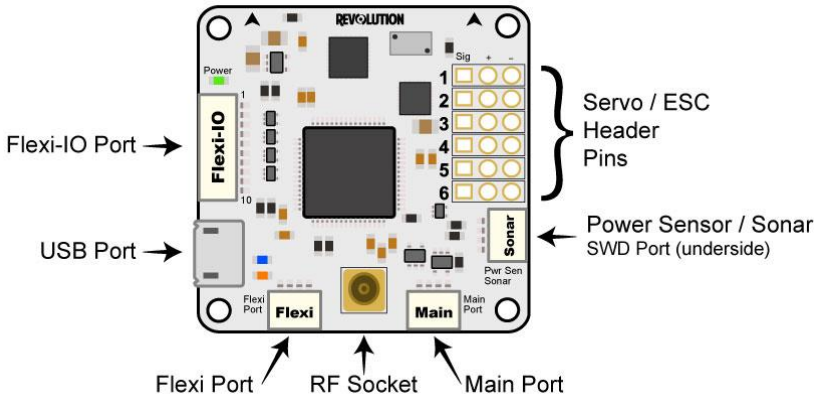
The MS5611 is not just any barometric pressure sensor, it is purpose-built, and has a very high resolution. As a result, it is ideal for use as a UAV altitude sensor. The sensor is so sensitive that it can sense a vertical shift of only 10 cm. The Revolution uses the newer MS5611-01BA03 version of this sensor that is far less susceptible to light interference than the older, plastic case versions.

Magnetometer

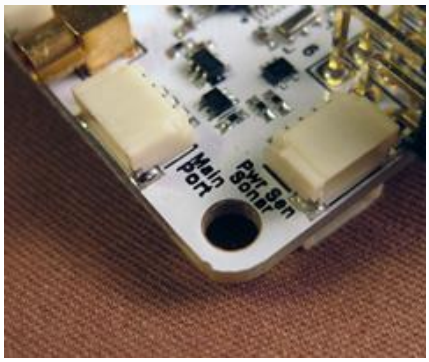
Whether you are in 'Position Hold' while taking aerial photographs, or you are flying a fixed-wing UAV on a pre-planned flight path, it's vitally important to have accurate heading information. The Honeywell HMC5883L is a three-axis digital compass module which provides rapid updates to changes in orientation which are accurate to a tolerance of 1° to 2°.

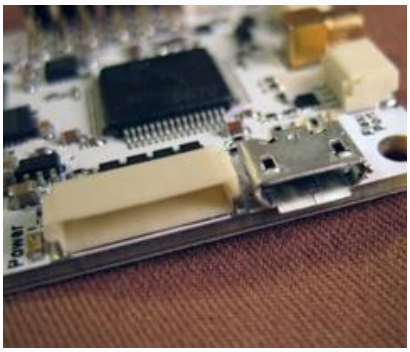
Connectivity

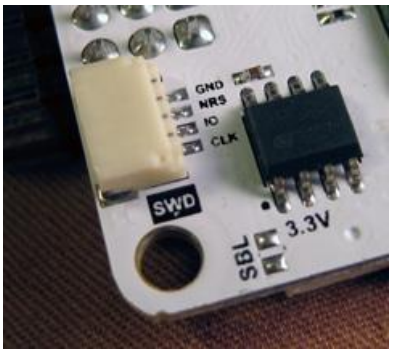
Just like CC & CC3D, the Revo has many ports, but a key new addition is the Flexi-IO Port. A lot of thought went into creating a small device that's flexible for use with multirotor platforms, helicopters and fixed wing aircraft, as well as making connectivity as fool proof as possible.



See pictures below for easy identifications of ports.







Flexi-IO

The Flexi-IO port uses a 10 pin JST-connector and is designed to perform several different functions. The port is used for control input and output and can be configured to serve 6 PWM inputs or outputs. A mixture of other modes can also be configured.

- 6 channel PWM input for receiver + 2 GPIO
- 1 channel PPM input for receiver + 5 PWM I/O channels for motor/servo or sensors
- 4 pin SPI + 1 UART + 2 PWM I/O (PPM mode)
- CAN with external transceiver, UART, 4 PWM I/O / PPM

The full unshared SPI I/O port can be used as master or slave, allowing you to interface the Revolution board to any other embedded system like, for example, the Overo Linux boards. This also allows use of the Revo for advanced robotic applications.

An additional board can provide raw CPU power and high level functions, ranging from networking and data storage, up to video analysis, autonomous behavior and artificial intelligence. The Revolution board will act as a reliable real time hardware controller and sensor platform, keeping your autonomous vehicle safely under control at all times.

Pin	**Color**	**PWM**	**PPM**	**PPM Function**	**PPM + Telemetry Function**	**PPM + Outputs**	**Outputs Function**	**Telemetry Function**
1	**Black**	**GND**	**GND**	**GND**	**GND**	**GND**	**GND**	**GND**
2	**Red**	**VCC**	**VCC**	**VCC**	**VCC**	**VCC**	**VCC**	**VCC**
3								
4								
5	**White**	**PWM 1**	PPM Input 1-8	PPM Input 1-8	PPM Input	PWM Output 12		
6	**Blue**	**PWM 2**			PWM Output 7	PWM Output 7		
7	**Yellow**	**PWM 3**			Telemetry TX	PWM Output 8	PWM Output 8	Telemetry TX
8	**Green**	**PWM 4**			Telemetry RX	PWM Output 9	PWM Output 9	Telemetry RX
9	**Orange**	**PWM 5**				PWM Output 10	PWM Output 10	
10	**Violet**	**PWM 6**				PWM Output 11	PWM Output 11	

Flexi Port

The Revo uses the same FlexiPort as the CC3D. The port can be used as either a UART or for I2C bus connectivity. It can be connected to serial devices like the OP GPS or any I2C device like the the Eagle Tree Airspeed expander module, ADCs, I2C ESCs and a lot more. It can also be used to connect Spektrum DSM2/DSMX Satellite to be used as receiver, or any other custom component interfacing with I2C or a serial connection including custom extension boards. Of course, it's also possible to run a serial Telemetry link to the GCS over the Flexi Port.

Pinout

Color	JST-SH Pin	Voltage	Serial Function (GPS, Telemetry)	I2C Function	DSM
Black	1	GND	GND	GND	GND
Red	2	4.8V - 15V	PWR Out (VCC Unregulated)	PWR Out (VCC Unregulated)	PWR Out (VCC Unregulated)
Blue	3	3.3V	TX	SCL	
Orange	4	3.3V (5V Tolerant)	RX	SDA	TX (Signal)

Warning: The Spektrum adapter should only be powered by 3.3V, a step down adapter must be used.

Warning: The PWR Out voltage is dependent on the CC supplied voltage. Verify that you use the correct voltage for your S.BUS receiver.

Main Port

Standard serial port/S.Bus port (same as CC3D). This can be used to connect serial devices like Telemetry, OP GPS, Futaba S.Bus receivers or Spektrum DSM2/DSMX satellites (to be used as a receiver), freeing in these cases the Flexi-IO port for other uses. These systems use a single wire to help cut down cable clutter.

Pinout

Color	JST-SH Pin	Voltage	Serial Function (GPS, Telemetry)	DSM	S.BUS
Black	1	GND	GND	GND	GND
Red	2	4.8V - 15V	PWR Out (VCC Unregulated)	PWR Out (VCC Unregulated)	PWR Out (VCC Unregulated)
Blue	3	3.3V	TX		
Orange	4	3.3V (5V Tolerant)	RX	TX (Signal)	TX (Signal)

Current / Sonar

This port can be configured to accommodate an Autopilot current sensor and a low cost Sonar sensor such as the HC-SR04. It can also be used as a general purpose input/output port or as a one or two channel analog input port.

Pinout

Color	JST-SH Pin	Voltage	Power Sensor
Black	1	GND	GND
Red	2	4.8V - 15V	PWR Out (VCC Unregulated)
Blue	3	3.3V	Current Input
Orange	4	3.3V (5V Tolerant)	Voltage Input

Revo - Current/Voltage Sensor Setup

This page describes how to setup a voltage/current sensor using a Revolution board.

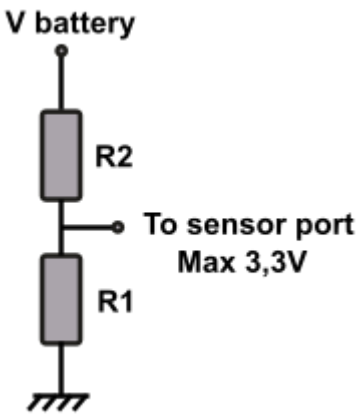
Color	Connector pin (board)	Description	AttoPilot pin (sensor)
Black	1	GND	GND
Red	2	Vcc	Not used
Blue	3	Current input	I
Orange	4	Voltage input	V

Basic voltage sensor

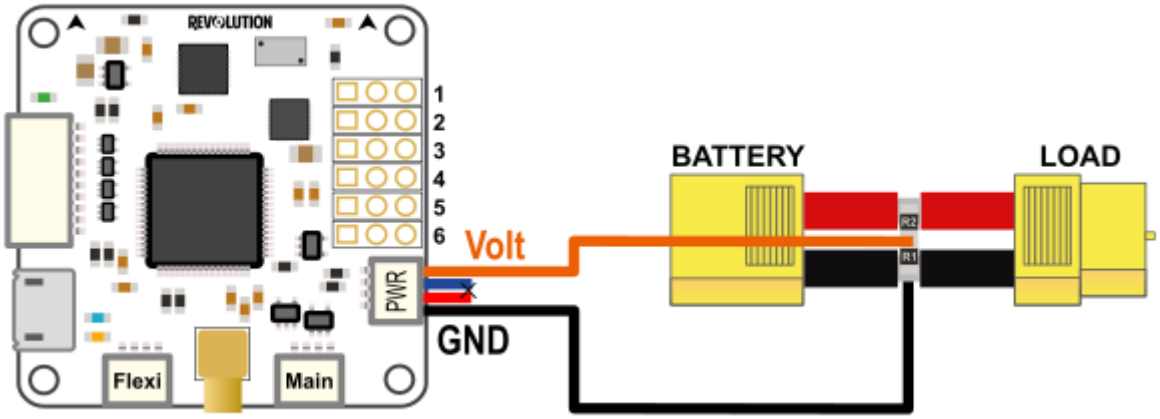
A basic voltage divider can be used, using two resistors connected between ground and **plus** from battery.

For a 4S battery (16.8Volts) the following values can be used:

R1: 2.2 KΩ
R2: 10 KΩ

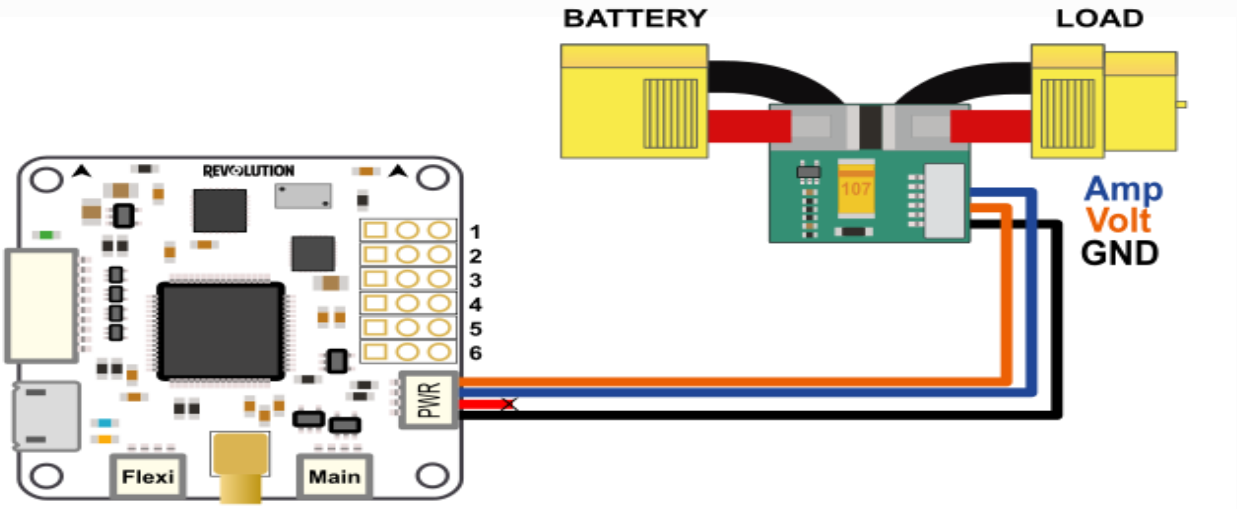


www.Hyperion-world.com
With V battery = 16.8V, V out = (16.8 * 2.2) / (10 + 2.2) = 3,03V



See also http://en.wikipedia.org/wiki/Voltage_divider

Current / Voltage sensor



Here is a list of common Attopilot sensors that can be used:

Sensor Type	Sensitivity	Voltage Factor	Current Factor	Voltage Zero	Current Zero
Attopilot 13.6V/45A	242.3mV/V, 73.20mV/A	4.127	13.661	0	0
Attopilot 50V/90A	63.69mV/V, 36.60mV/A	15.701	27.322	0	0
Attopilot 50V/180A	63.69mV/V, 18.30mV/A	15.701	54.645	0	0

Note: Attopilot 90A is same as RCTIMER 90A current sensor

Calculate Sensor Calibration values

To configure the battery monitor, you need to calculate the **Sensor Calibration** values.

For example, the documentation for Attopilot current sensor specifies the following information for each sensor type:

Sensor type	Voltage sensitivity	Current sensitivity
50V/180A	63.69mV/Volt	18.30mV / Amp
50V/90A	63.69mV/Volt	36.60mV / Amp
13.6V/45A	242.3mV/Volt	73.20mV / Amp

In order to use 50V/90A you'll have to do the following:

- Convert everything to Volt, so Volt/Volt and Volt/Amp:
  - 63.69mV/V / 1000 = 0.06369 V/V
  - 36.60mV/A / 1000 = 0.0366 V/A
- Then calculate the values to insert in **Voltage** and **Current Factors**:
  - **Voltage Factor** = 1 / 0.06369 = 15.701
  - **Current Factor** = 1 / 0.0366 = 27.322

Those factors are start values that can be adjusted later from readings, generally sensor's sensitivity are not so accurate.

Hardware settings

The first thing to do is to configure the hardware settings:

1. Connect the board to your computer and got to the **System tab** in your GCS
2. Browse the first part (Settings) and find the **HwSettings** UAVObject
3. Enable the Battery module: **OptionalModules** > **Battery** > **Enable**
4. Set the analog input pins for voltage and current:
   - **ADCRouting** > **adc0** > **BatteryCurrent**
   - **ADCRouting** > **adc1** > **BatteryVoltage**
5. Save changes, click **Upload** button
6. Reboot your board: disconnect **all power sources** and reconnect to your computer.

Property	Value	Unit
HwSettings		
Meta Data		
CC_RcvrPort	PWM+NoOneShot	function
CC_MainPort	Telemetry	function
CC_FlexiPort	Disabled	function
RV_RcvrPort	PWM	function
RV_AuxPort	Disabled	function
RV_AuxSBusPort	Disabled	function
RV_FlexiPort	Disabled	function
RV_TelemetryPort	Telemetry	function
RV_GPSPort	GPS	function
RM_RcvrPort	PPM+Telemetry	function
RM_MainPort	GPS	function
RM_FlexiPort	I2C	function
TelemetrySpeed	57600	bps
GPSSpeed	57600	bps
ComUsbBridgeSpeed	57600	bps
USB_HIDPort	USBTelemetry	function
USB_VCPPort	Disabled	function
OptionalModules		
[CameraStab]	Disabled	
[GPS]	Enabled	
[Fault]	Disabled	
[Altitude]	Disabled	
[Airspeed]	Disabled	
[TxPID]	Enabled	
[Battery]	Enabled	
[Overo]	Disabled	
[MagBaro]	Disabled	
[OsdHk]	Disabled	
ADCRouting		
[adc0]	BatteryCurrent	
[adc1]	BatteryVoltage	
[adc2]	Disabled	
[adc3]	Disabled	
[adc4]	Disabled	

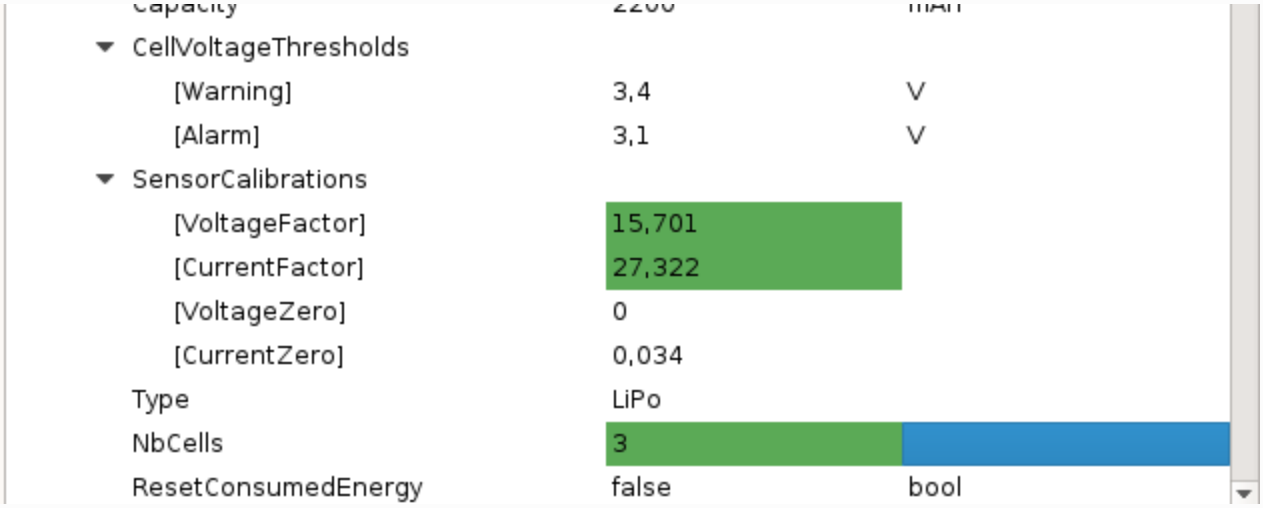
Calibrate sensor

Now connect the battery to the battery sensor, if changes are done and battery module enabled you can see the display on PFD:

But values displayed are not accurate, we need to calibrate sensor!

Setting Flight Battery Settings Properties

1. Connect the board to your computer and go to the **System tab** in your GCS
2. Browse the first part (Settings) and find the **Flight Battery Settings** UAV Object
3. Set the **Voltage Factor** and **Current Factor** as in the previous section.
4. Set the cell **Capacity** if you want a warning to be issued when there are less than 2 minutes of flight time available; otherwise, set to 0.
5. Set **Nb Cells** with the number of cells in series (2 for 2S, 3 for 3S, 3 for 3S2P)
6. Set the **Warning** and **Alarm** thresholds if you would like a warning or critical alarm to be issued when going under the values that you specify.
7. Save changes, click **Upload** button.



Note:

Sometimes sensor are not accurate with small readings, especially for current.

A Current Zero setting can be set without any load change this value to obtain a 0 amp. display.

PWM output headers

Just like the CC & CC3D, the Revo has a bank of 6 PWM output headers. If more PWM outputs are needed - the Flexi-IO port can be configured to support up to an additional 6 PWM channels if so required. PWM port 5 can also be configured to communicate with an external analog airspeed sensor or a governor for helicopters.

SWD Port

Serial wire debug port. This allows the use of cheap boards like the STM F4 Discovery as an in-circuit debugger to ease the firmware development.

Pinout

Color	JST-SH Pin	Pin Description
Black	1	GND
Red	2	NRS
Blue	3	IO
Orange	4	CLK

Micro USB

We have decided to move away from using the mini USB port found on the existing CC and CC3D and instead use a Micro USB port. The Micro USB port has several benefits, it's physically smaller, more robust, a much more widely adopted standard, and is the same type of port found on the majority of mobile phones. In all likelihood, you'll already have one of these cables at home.

The USB port provides a USB composite device with the following functions:

- OpenPilot HID device (default GCS interface, uses PC system drivers)
- CDC virtual serial port (telemetry, debugging, serial bridge mode relaying data from/to physical serial port to the virtual one for GPS/Bluetooth module setup, etc.)
- 8-channel HID joystick (passes data from all supported R/C inputs to PC flight simulators)

OPLink Modem

The Revolution has its own *OPLink Mini* built right onto the board! This is not only a 10DoF flight controller with an ST32F4 processor in the same small footprint as the CC3D, but also has its own LRS modem. The modem is directly powered from the Revolution itself, so you don't need to worry about any additional power supply.

Of course, the on-board modem will have the same functionality as the OPLink Mini. Both are fully configurable from the GCS.

Operating on the 433MHz band (a 900Mhz version will be released at a later date), the modem provides a direct telemetry link between the GCS and your flight controller. And just as with the OPLink Mini, you can adjust the output RF power for compliance with any governmental RF regulations, or it can be disabled entirely.

Note: For OPLink setup please see manual.

For DIY Board Users

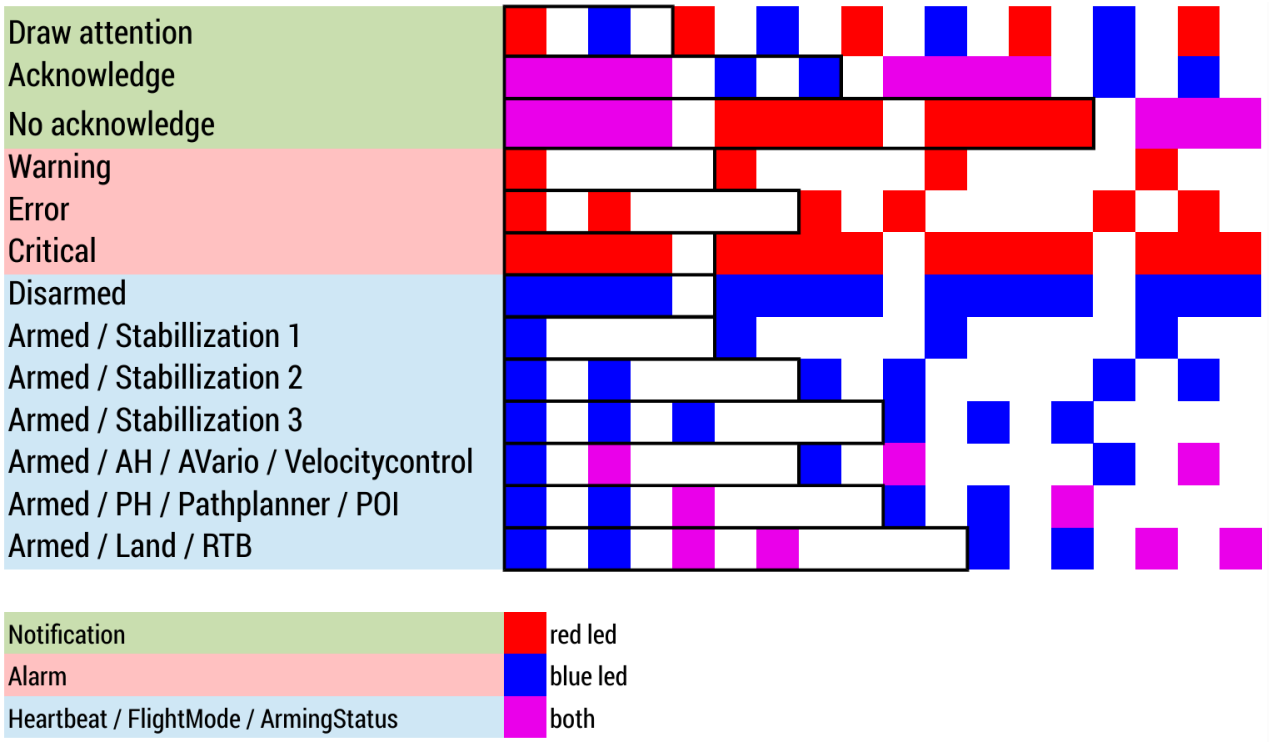
Please download below file for Schematics, PCB Layout.

http://opwiki.readthedocs.io/en/latest/_downloads/Revolution.zip

Revo - LED Indications

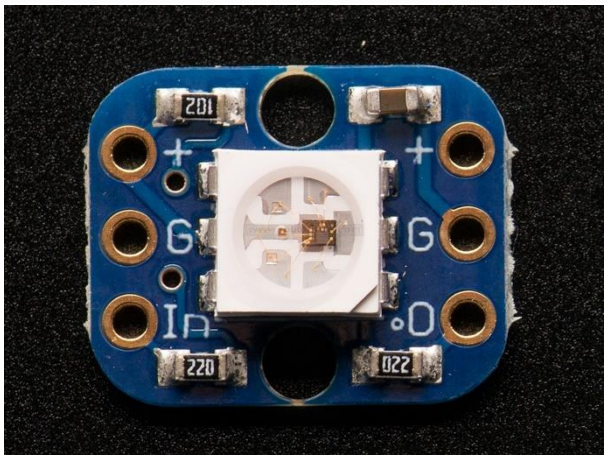
They will show in sequence:

- Notifications
- Alarms
- Heartbeat/Flight Status/Arming Mode.



Revo - External LED setup

Overview



This document will describe the necessary hardware and software setup to connect an external notification LED to Revolution. WS2811, WS2812 and WS2812B multicolor LEDs are supported in both single LED or multiple LED configurations. The LED is useful for debugging and it provides visual information about flight modes and warnings during flight. The battery low voltage warning is particularly useful.

Please refer video link below

https://www.youtube.com/watch?v=S7CISYWC7MA

Hardware Connections

The OpenPilot firmware supports LEDs wired in both parallel and series configurations. A single LED is capable of displaying all of the data. Both the LED strips and the modules have voltage input (5V), ground (GND) and signal input (DI) pins, and also often corresponding output pins. For a single LED configuration, the output pins are not used; and in a multiple LED configuration, you can wire the LEDs together serially in a chain.

On the left is an example of a very common single LED module which is readily available on eBay.

If you use a single LED, make sure that the breakout board has a capacitor on it; otherwise, inrush current spikes can damage the LED. The capacitor is usually a surface mounted component just like the brown one next to the LED unit in the photo. A capacitor is optional on configurations with two or more LEDs.

Revolution can command the LED(s) from various outputs. For the output signal, servo output rail pins 1-6 are supported, and Flexi-IO port pins 3 and 4 (from the left) are supported. Power for the LED is available from Flexi-IO pins 1 (GND) and 2 (5V), or from the servo output rail. After making the necessary connections, it is recommended that you mount the LED so that it is visible from below while flying.

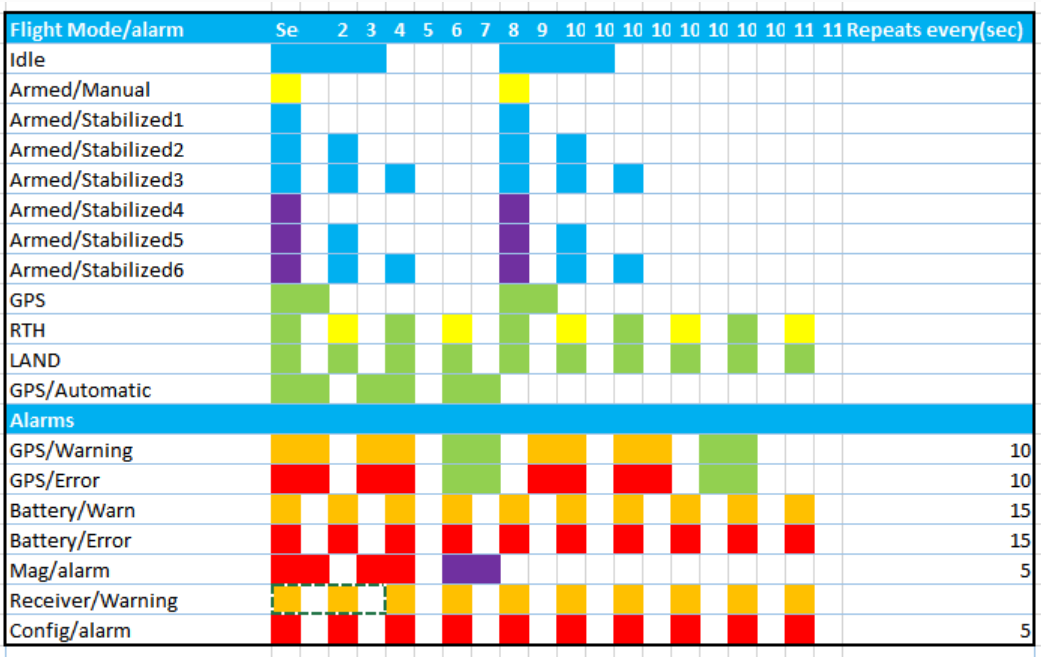
Software Setup

Following hardware setup, configure the controller in GCS as follows:

- Go to **System tab > HwSettings > WS2811LED_Out** and choose the pin where you connected the LED(s)
- Click the red **Save button** in the top part of the view
- Disconnect Revolution from PC
- Connect your flight battery. LED(s) should start showing system status according to the graphic in the Light Codes section below.

Light Codes

The following graphic shows different notifications Revolution will display with the LED(s).



Revo - Sensor Calibration

This panel will allow you to undertake a Manual Calibration of the Sensors on the Revolution board. There are six (6) different ways of Calibrations, that can be performed.

- **Setting Home Location**
- **Thermal Calibration**
- **Accelerometer Bias Calibration**
- **Magnetometer Calibration**
- **Board Level Calibration**
- **Gyro Bias Calibration**

https://www.youtube.com/watch?v=dn9IDw2D1qw

Setting Home Location

Warning

You must set your home location before performing the Sensor Calibrations

There is two ways of setting your home location:

1. Using GPS Module
2. Manually Select

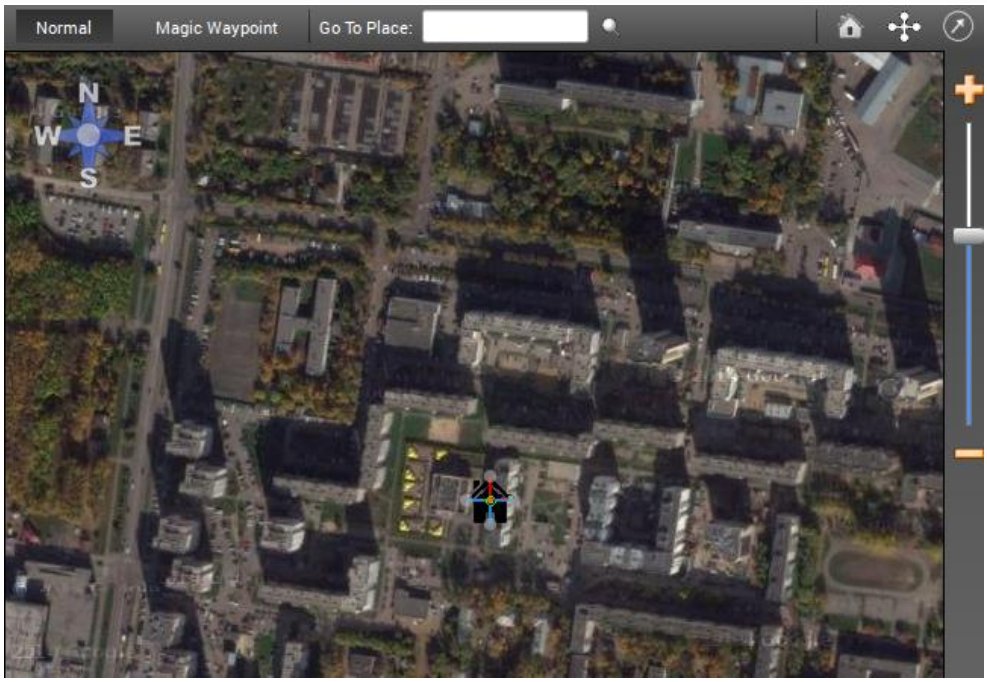
- Go to **Flight Data Tab**
- Either wait for GPS to go to your home location (using GPS Module) or manually zoom map up to your location (Manual Select)
- Once found, right click on the map and click on **Set the Home Location**

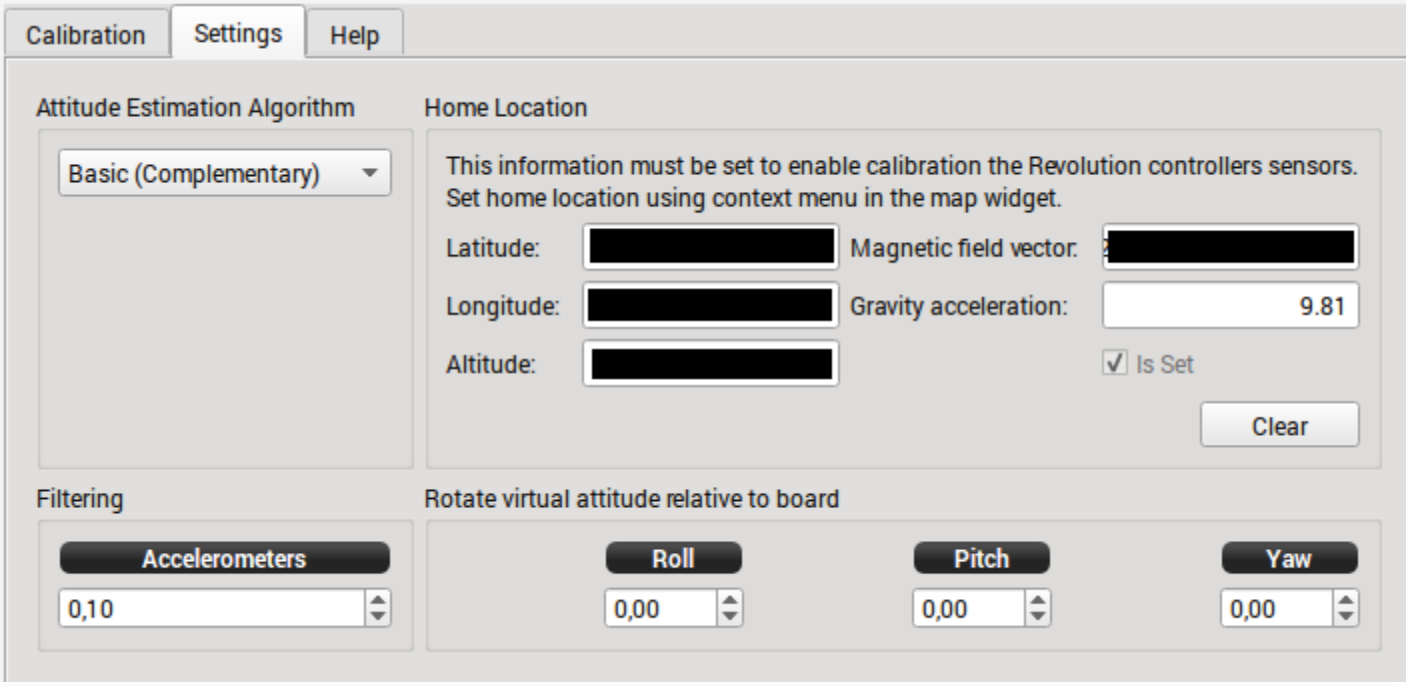
Refer image below

- Enter the Altitude of your home location in meters, click **OK**

Note

You can check the home location coordinates by going to the Configuration Tab - Attitude/Settings tab





Thermal Calibration

Note

The procedure below outlines the way to thermally calibrate the Revo to meet large temperature changes, it is not essential to put in the freezer as the min and max temp ranges of the calibration only need to be below and above the ambit temps of your location. (approx. ±10/15 deg C above and below)

However if the temperature rises or falls above or below the min/max temps you calibrated at, you will need to redo this calibration, so it is best to use some form of cooling to lower the min start temp and raise above max ambit temps of your location. Also note this calibration is not needed, but will improve the performance of your board

1. Plug your USB cable into your Revo. Start the OP GCS and ensure that you have disabled the OPLink modem (i.e. **Max Power** = 0). Then **Save** and **Disconnect**. Remove USB cable from computer but leave Revo plugged in.
2. Put your Revo in a plastic bag and seal it reasonably well, but with the USB cable exiting the baggie so it can be plugged in the computer.

Note

The bag is because we are going to freeze Revo and it will get condensation on it if we don't do this. For added temperature range (higher temperatures) you might build a very simple hot box out of a low wattage incandescent light and a shoe box. You also might consider just wrapping Revo up in a good insulator, like a warm lightweight modern jacket. You could also set it out in the hot sun in the summer if you can plug it into USB at the same time.

3. Put Revo/baggie/cable in the freezer for 20 minutes or so.

4. Take it out, put it in the hot box or wrapped up or in the sun, or not... Immediately start the thermal calibration.

- Don't move it at all while calibrating this test (gyros and baro are being calibrated).
- Don't allow your home heater or air conditioner fan to run while doing this test. (air pressure changes).
- Don't open or close any doors in the house. (air pressure changes).
- Don't do this on a windy day (chimney, etc.).
- Don't do this on a stormy day (rapid changes in barometer).
- Don't play 1812 Overture or let your sister dance in the next room.

5. Watch the thermal calibration and unplug the light bulb when it gets within a few degrees of your desired high temperature (i.e. 10-15 deg above what hottest day in your location).

6. If it finishes thermal calibration before reaching your desired temperature, you can start over or use what you have.

7. Save your thermal calibration with the **Save** button in the lower right.

Note

You can check the temperature min-max (temp_calibrated_extent). Go to the System Tab then Settings/Accel Gyro Settings/temp_calibrated_extent. You can then see the min & max calibrated temps (below is screen shot of uncalibrated board showing 0 for both min & max, once calibrated you will see your boards min & max temps)

Accelerometer Calibration

Warning

During this calibration the board MUST remain perfectly still during each position of calibration. Even vibrations on a table or someone walking past could upset the calibration

On the **Calibration Tab**, click start "**Accelerometer calibration**".

1. Place the board as shown in the picture on screen and click **Save Position**.

2. If the calibration failed, you will need to restart the accelerometer calibration over.

Multirotor Vehicle Setup Wizard

The Vehicle Setup Wizard allows users to step through and setup all the initial settings including the TX.

When the GCS is started, the Vehicle Setup Wizard can be accessed by clicking on the "green" Vehicle Setup Wizard button.

Ensure board is connected through USB port before starting Wizard.

Depending on the flight controller board, some screens may or may not be displayed as you progress through the wizard.

First Flight

Pre-flight checks.

There are a few things that you'll want to check before flying. Especially in the case of a multi-rotor. If you have swapped your motor outputs or your servo output is reversed, then the multi-rotor may flip or spin directly upon lift off.

Warning

REMOVE PROPELLERS!

You should always remove your propellers prior to starting tests or connecting your battery for the first time.

Arming (Turn your board on)

At start-up the system is disarmed. When disarmed, the motor(s) will not function, but the servos should work (*If installed, e.g. on Tri-Copter*). Before flying, the system needs to be armed with a specific stick input. The input that is needed for arming (and disarming) is configurable. For safety reasons, the default configuration is such that the system will not arm under any condition. Hence, it's mandatory that you configure the arming in the GCS input configuration.

Check your system alarms

The flight controller will not arm itself if any system status alarms are present. They can be checked by connecting to the flight controller board with OpenPilot Ground Control Station, and looking at system status indication under the primary flight display. All status objects should be green. PATH can remain yellow. For more information on alarms, read the next page.

Check your stick inputs

With props removed, after you have armed your board, the blue LED should start flashing rapidly indicating that the board is armed. If you have specified that the "motors should spin at neutral output" in the output configuration, the motors should start spinning right now. Otherwise you can now apply some throttle and get the motors spinning.

Verify that the correct motors follow your stick input. E.g. rolling left should increase the throttle on your right motor(s) and vice versa. If the wrong motors are spinning faster, you need to recheck your configuration.

Rate mode

Note that some motors could start spinning faster if you have already selected a stability mode on your transmitter with the Flight Mode switch. You should do your first tests in rate mode only.

Check your stabilization

While the motors are spinning, you can move the aircraft in the pitch or roll direction. The flight controller should react promptly and accelerate the correct motors to counteract the movement. Please make sure that the correct motors are spinning up. If the opposite motors spin up, then you would flip your multi-rotor immediately upon lift off. In this case, you need to recheck your configuration.

Important: Check fail-safe!

Arm the aircraft and run your motors, now turn off the transmitter and confirm that all motors stop.

Your first flight (multicopters and helicopters)

You've done your pre-flight checks, your transmitter and flight batteries are all fully charged, and you are confident that the vehicle has been configured correctly - it's time to fly!

But, it's also time to take a deep breath and honestly evaluate your flying skills - if you are a novice then you should consider buying a flight simulator and developing some basic skills - crashes in a simulator cost nothing, so you will rapidly justify the investment for this software.

If you are feeling confident, take the model to a large, clear and deserted open space - you don't want any trees or overhead wires nearby, you definitely don't want any people or dogs

in the area, and you shouldn't be too close to a road. A grass field is better for first flights, rather than concrete or tarmac. Ideally there will be no wind, but a very slight breeze shouldn't be a problem.

Important

You may be disconnecting and connecting power to the model several times during these first steps - you must wait at least 30 seconds before reconnecting power, otherwise the model may behave unpredictably.

Remember to keep the model still for about 10 seconds after each connection, while the gyros are calibrated.

The following steps have been written on the assumption that 'Zero gyros while arming aircraft' has been selected on the Attitude Configuration page of GCS (the recommended setting). This can help with the stabilization of the model.

Note:
Remember to keep the model still for several seconds after arming, while the gyros are zeroed.

1. If you have brought anybody with you to witness the first flight, make sure that they know where to stand, how to behave, and what to do if the vehicle goes out of control. If there are any children, make sure that they are under close supervision.
2. Put the throttle stick to minimum and switch on the transmitter. If your transmitter has a throttle lock facility, set this to 'locked'. Put the transmitter on the ground next to where you plan to stand while flying the vehicle.
3. Place the model on the ground about 10 paces away with the tail towards the transmitter. If there is a breeze, put the model either upwind or downwind of the transmitter - you don't want a cross-wind.
4. Connect the power to the vehicle and **do not move it for about 10 seconds while the gyros are calibrated**. Return to your transmitter.
5. Take a good look around for safety's sake, then unlock the throttle and arm your board - the blue LED on the board will now flash rapidly and the rotors will spin if you've selected 'Motors spin at neutral output when armed'. **IMPORTANT: you must wait a few seconds after arming while the gyros are zeroed** (see note above). The aircraft may be unstable if this isn't done. Use this time to have another good look around you.
6. Steadily increase the throttle until the vehicle is about to lift off the ground - any tendency to flip or spin will be apparent at this time. Close the throttle immediately if the vehicle does anything unexpected, and then take a look at the problem-finding guide.

7. If everything looks OK. then close the throttle, disarm the vehicle, take a deep breath and have yet another look round the field - it's time for lift-off!
8. Arm the vehicle, wait for a few seconds, then open the throttle confidently until the aircraft lifts off the ground. Try to hover about 1 or 2 meters off the ground, while keeping in one position.

You are now flying! Obviously this bit is not as easy as it sounds and much practice is required.

The important thing to remember is to close the throttle if the aircraft goes out of control - you will crash at some time, and cutting the power will minimize the damage.

Get into the habit of disarming the board when landing for more than a few seconds or when approaching the vehicle to handle it, and don't forget the short wait after re-arming - again, use this period to look around the field before flying.

Disarm the board and set the throttle lock when you have finished flying, then put down the transmitter a little way from the model. Disconnect the power from the aircraft, then turn off your transmitter.

Now that you have proven that the aircraft will fly, you might like to try the training exercise videos for helicopters found on this website page. Many of these are also suitable for multi-rotors.

Optimizing values

Apart from tuning the stabilization settings, there are some values which advanced users may want to change pretty soon. The default values will fly your aircraft perfectly fine, but would limit some users in their flying style.

Please find below a few settings which can easily be cranked up for more experienced users. Note: these are available in each of the three Settings Banks.

Full stick angle in Attitude mode

Location	Stabilization panel, Responsiveness, Attitude mode response (deg)
Standard value	55°
Tuned+ value	65° or more

Specifies how many degrees the vehicle will bank on a full stick deflection when in attitude mode.

If you fly your multi-rotor in heavy wind, you may find that low values are not sufficient to counteract the wind fast enough. The default value is fairly good for a beginner.

Full stick response in Rate mode

Location	Stabilization panel, Responsiveness, Rate mode response (deg/s)
Standard value	220°/s
Tuned+ value	360°/s or more

Specifies how many degrees per second a full stick deflection commands in all modes **except Attitude mode**.

If you want to do flips with your multi-rotor you should increase this setting. Flips that take too long to complete can result in too much loss of altitude for the beginner. To get some idea of how fast you want to flip, imagine the flip taking one second to complete - that would equate to 360°/s.

Note

This is the value to control rotation rate when using attitude mode

Full stick response limit in any mode

Location	Stabilization panel, Responsiveness, Max rate limit (deg/s)
Standard value	300°/s
Tuned+ value	360°/s or more

Specifies the maximum rotation rate in degrees per second commanded by a full stick deflection on the associated axis **in any mode**.

Make sure this is the same or higher than Rate mode response above.

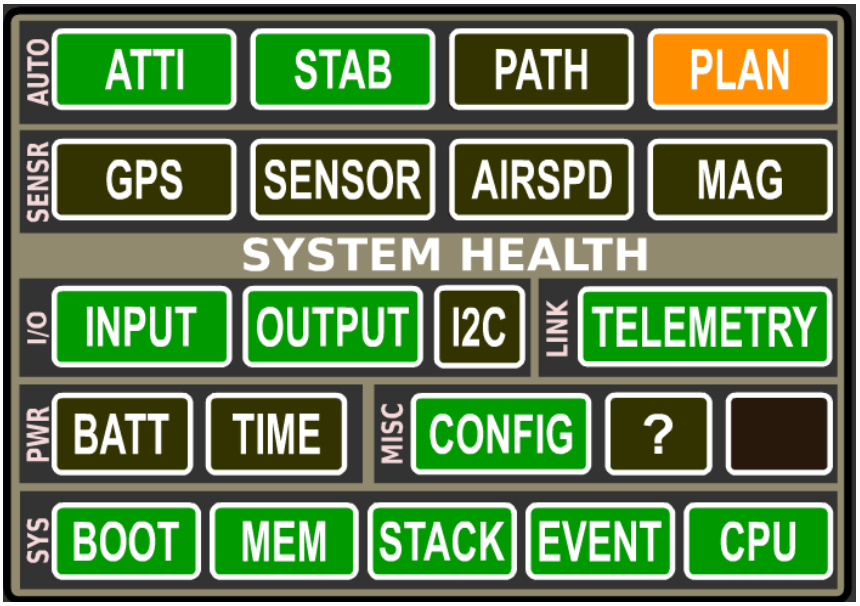
MaxAxisLock

Location	Stabilization panel, Expert Tab, Axis Lock
Standard value	5°
Tuned+ value	15°

The maximum number of degrees that the control accumulates error. The default setting is changed to 15° in newer firmware & should be a good value for multi-rotors.

Alarms - What they mean and how to fix them

Status widget overview



The OpenPilot/LibrePilot flight control firmware has a built-in status system that gives you an overview about what's happening with the board. The System Health Status widget in the Ground Control Station can be used to diagnose various problems, and to check which modules of the firmware are running.

Arming

The flight control board cannot be armed until crossed-out alarms and red alarms have been solved. All objects, except for the special "Plan", should be green for optimal flight performance. The information corresponds to the UAVObject **DataObjects > SystemAlarms > Alarm**.

Status object explanations

Auto (Autonomous)

Atti

Shows the status of the board's attitude data. If all is well with gyroscope and accelerometer, it turns green after gyroscope calibration upon power up, or if you are using "**GPS Navigation (INS13GPSOutdoor)**" stabilization mode, when the Inertial Navigation System's **Extended Kalman Filter (EKF)** has fired up. EKF is a sophisticated sensor fusion algorithm that takes data from relevant sensors and creates a best possible estimation of the board's angle, velocity and position.

- 

Attitude data not available, waiting for gyroscope calibration. Don't move the vehicle while gyros are being calibrated upon board power-up. There is no data coming in from the sensors, which usually indicates faulty onboard sensors on CC3D and unknown home location on Revo. For Revo this is normal when you have not had GPS fix yet on a new build. The home location will be automatically set when GPS gets enough satellites and good fix. The sensors can be damaged in a bad crash. Otherwise, contact board seller.
- 

Data is received from the sensors, but attitude information is not available. This usually happens when the EKF is not running yet. Make sure that GPS and MAG alarms are green, and that all calibrations have been done properly. It sometimes helps to move the vehicle around a bit to give EKF a better view of what's going on with sensors.
- 

EKF is running, but the state estimation is not optimal. Good calibration and moving the vehicle a bit helps this situation.
- 

The system has a clear view of the state of the vehicle.

"The Bug"

There is a very specific condition that can arise within the system that can sometimes occur after the EKF has been initialized, and the vehicle has been left stationary for an extended period of time. Fortunately this condition is easily detected and the flight firmware mitigates the event. However, the solution is extremely complicated and beyond the capability of most people, and until such stage as the developers have a chance to address the matter, it's occurrence is indicated as follows:

- A large red X appears over the PFD.
- A Yellow ATTI alarm with STAB green

This event has been nicknamed "The Bug", and it is still possible to arm and fly the vehicle in this condition.

Should you wish to clear the indication, one can reset the Revolution flight controller, or alternatively, change the Attitude Estimation Algorithm to **Basic(Complementary)**, and then change back to **GPS Navigation(INS13)**. This action will cause the EKF to reinitialize and the indications will be cleared.

Stab

Shows whether the board is capable of stabilizing flight. This status goes very much hand in hand with Atti.

- 

Waiting for gyroscope calibration. Don't move the vehicle while gyros are being calibrated upon board power-up.
- 

The stabilization module cannot stabilize flight. See red Atti status for explanation.
- 

Can be a brief alarm when one gyroscope update is missing.
- 

Stabilization of flight can be performed.

Path

Shows whether the Revolution board is capable of autonomous path following. Autonomous flight requires a GPS and a stabilization algorithm set up as GPS navigation.

- 

The flight controller has not been configured to do autonomous flying. Autonomous flight is only possible with stabilization mode "GPS Navigation".

- PATH The system has been configured to initialize Path Follower module, but it can't be used at the moment. It happens usually because EKF is not running, see Atti red explanation.
- PATH All is good with the Path Follower module. You can use GPS flight modes.

Plan

Shows the status of an autonomous flight plan that can be uploaded to Revolution using the Ground Control Station. The status of Path remains yellow until a proper plan has been uploaded, and turns green if all is good with the plan. A valid plan can be activated with a path follower flight mode.

- PLAN The flight controller has not been configured to do autonomous flying. Autonomous flight is only possible with stabilization mode "GPS Navigation".
- PLAN Path has been uploaded, but data is invalid and cannot be used for autonomous missions.
- PLAN No path plan has been uploaded, but the system is ready to receive a plan. This is okay if you don't intend to do autonomous missions right now.
- PLAN A valid and usable flight plan has been uploaded and stored on Revolution.

GPS

Shows the status of the GPS that can be connected to an OpenPilot flight controller. GPS is required for autonomous missions and more sophisticated flight modes.

- GPS A GPS has not been configured to be used.
- GPS The GPS has been configured, but no valid data is coming in. This is normal if flight battery is not connected, because GPS only gets power from external sources, not USB. Check the baud rate and the used protocol of your GPS. Double check serial connection, TX/RX need to be crossed between board and GPS.
- GPS Serial communication is fine but the GPS has no valid fix. Wait for GPS to gather satellites, and preferably have your vehicle in an open area.
- GPS The GPS has a fix and navigation can be used. However, the position quality is very low (the indication <7 satellites and/or PDOP > 3.5m). A blue LED will flash on the OP v8 and v9 GPS.
- GPS The GPS has a valid 3D fix.

Initial GPS setup information

When powering up the GPS for the first time, it might take over 30 minutes for the GPS to download almanac information from the satellites and acquire a good lock. Be patient, and have a clear view of the sky.

Sensor

Shows the status of the sensor handler module.

- SENSOR Sensor module is not being used in current configuration.
- SENSOR Sensor communications are up and ok.

Airspd

Shows the status of an optional air speed sensor that can be used with fixed wing aircraft.

- AIRSPD Airspeed sensor has not been configured to be used.
- AIRSPD Valid data is coming in from the airspeed sensor.

Mag

Shows the status of Revolution's magnetometer, or the status of an auxiliary magnetometer on board the OpenPilot GPSv9 if the GPS is configured to be used.

- MAG Magnetometer is not being used in current configuration, or auxiliary magnetometer is not feeding data. If using GPSv9, power up the board externally. Home location has to be set to enable magnetometer.
- MAG Data is coming from the magnetometer, but the readings are off by over 15%. This can be caused by various reasons:
  - Magnetometer has not been calibrated with current vehicle (or after recent modifications to vehicle)
  - There are high currents in wires, interfering with the magnetometer. Twist wires and route them away from magnetometer.
  - Calibration was properly done outside, away from metallic objects, but the vehicle is now inside in a different magnetic environment. This behavior is normal.
- MAG Magnetometer readings are off by over 5%.
- MAG Magnetometer is working properly and the quality of the measurements is good.

I/O (Input / Output)

Input

Input module handles the data that is coming from your receiver.

- INPUT R/C input has not been configured. Use **Input tab** or **Transmitter Setup Wizard** to configure your radio channel inputs.
- INPUT No R/C input data. Power up receiver with the flight battery.
- INPUT Valid R/C input data is coming in.

Output

Output module takes motor speed and servo position data from stabilization algorithms, and feeds it into output channels.

- OUTPUT Channel outputs have not been configured. Use **Vehicle Setup Wizard** to configure them automatically.
- OUTPUT Outputs are configured and can be updated.

I2C

I2C is a bus that connects onboard or auxiliary sensors and handles the data transmissions. I2C is designed for communications internal to a PCB, and does not work well via wire connections. It is okay to use for LED controls and similar functions, but is absolutely not recommended for flight-critical sensor connections.

- I2C I2C module is not being used.
- I2C I2C module is in error state.
- I2C I2C communications are up and working properly.

Telemetry

Shows the status of Telemetry communications module

- TELEMETRY Telemetry module has encountered an error. Set up only one telemetry output port.
- TELEMETRY Telemetry data communications are working properly.

Pwr (Power)

Batt

Battery status shows whether you have enough voltage in the battery to fly. Set limits for this in FlightBatt (CHECK this) settings. It requires a battery voltage sensor to work. Battery monitoring module can be enabled in system settings' optional modules.

- BATT Battery monitoring module is not enabled.
- BATT Not enough battery voltage to safely take off. Default limit is 3.1V/Cell.
- BATT Battery voltage is low, but flying is possible. Default limit is 3.4V/Cell.
- BATT Battery voltage is ok.

Time

Shows whether you have enough energy in the battery left for flying, and requires a battery voltage and current sensor to work. Currently has a bug when not using a current sensor; set the battery capacity to 0. This disables the estimated flight time counter and associated alarms.

- TIME Battery monitoring module is not enabled, see above Batt explanation.
- TIME Battery energy is low, flying cannot be performed safely.
- TIME Low amount of energy in the battery, flying is still possible.
- TIME Good amount of energy left in the battery to fly.

Config

Shows whether your flight controller board has been properly set up.

- CONFIG Board configuration problem. If you have set up GPS modes (GPS Assist, PosHold, RTB) to one flight mode, then make sure that "GPS Navigation (INS13)" fusion algorithm is selected.

  CC3D can't use GPS Navigation (INS13) and do not support GPS Assisted modes.

  Select **GPS Navigation (INS13)** in **Config** > **Attitude tab** > **Parameters** > **Attitude Estimation Algorithm**.
- CONFIG Board configuration ok.

Sys (System)

Boot

Shows that a board reboot is required, or fail-safe settings have been loaded upon boot.

- BOOT Boot alarm can be caused by various reasons:
  - No valid telemetry option selected, so board will boot with default USB telemetry
  - Board init failed due to driver, module or RAM issues, and the board has been booted up in fail-safe state
  - Board has been put to safe mode by the user
  - Board needs a reboot after hardware configuration changes
- BOOT Flight controller booted up properly.

Mem

Displays the status of remaining memory (RAM) that are used by processes internal to the flight controller.

- MEM Very low RAM left, flying cannot be done safely. Less than 40 bytes for CC/CC3D or 500 bytes for other boards.
- MEM Low amount of RAM left, flying can be done but don't enable more software modules. This is common with older flight controllers such as CopterControl. Less than 200 bytes for CC/CC3D or 1000 bytes for other boards.
- MEM Sufficient amount of RAM left for system to operate and expand.

Stack

Shows the status of the microcontroller's stack, which is a place where low-level functions store data.

- STACK Stack status ok.

Event

Shows the status of event system. A very heavy load can cause the event system to be overloaded.

- EVENT Event system error or overloaded. This can be caused by a bug or too high telemetry update rates when OPLink has low baud, for example.

- EVENT Event system at high stress. See above.
- EVENT Event system ok.

CPU

Indicates CPU load.

- CPU CPU load is very high, flight cannot be performed safely.
- CPU CPU load is high, but flight can be performed. Don't enable more software modules like TPS or board rotation. Should only occurs for CC/CC3D.
- CPU CPU load is at an acceptable level, and flying is safe.

Bootloader Update

Introduction

The bootloader is a small piece of software resident on the flight controller which is started as soon as the board is powered. It performs hardware checks and loads and executes the firmware. It also handles the USB port to communicate with GCS to support bootloader and firmware updates. You can check the version of the bootloader installed on any OP board to make sure it is the most up-to-date version and upgrade the bootloader software if required.

Warning

Upgrading the bootloader can be tricky. If something goes wrong, you may render the flight controller inoperable. Happily, almost all bad mistakes can be rectified via the methods described on the "How to" sub-pages given in the manual below.

How to Check the Bootloader Version

The bootloader version is shown when the flight controller is in boot mode. You can enter boot mode by:

- using either the **Halt** option or,
- by using **Rescue**.

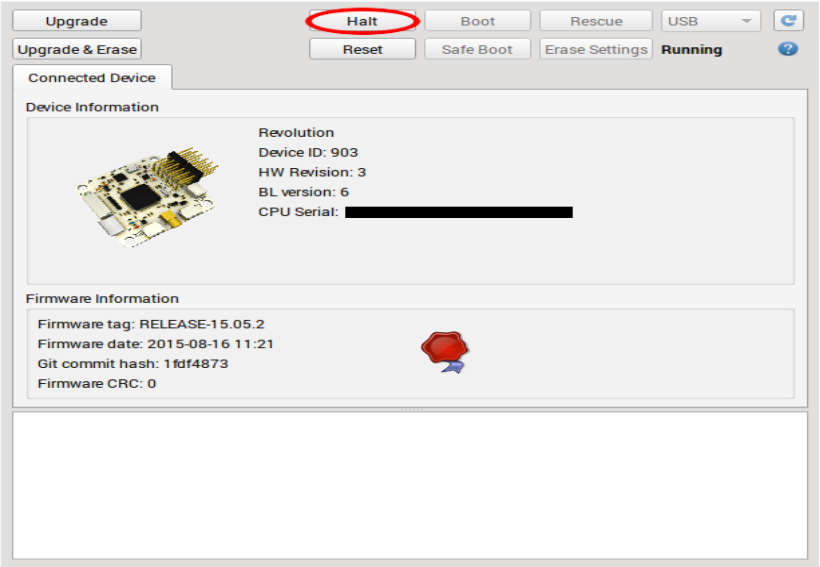
Either method will reveal the bootloader version on your board and both methods are explained below.

Halt Option

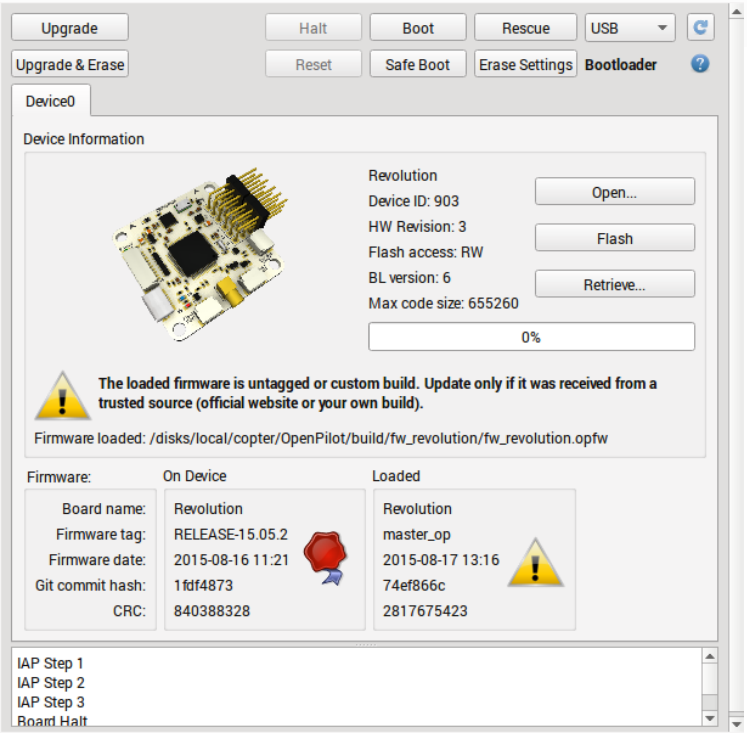
Connect the flight controller to GCS via USB and open the Firmware page. On the Firmware page,

click **Halt**.

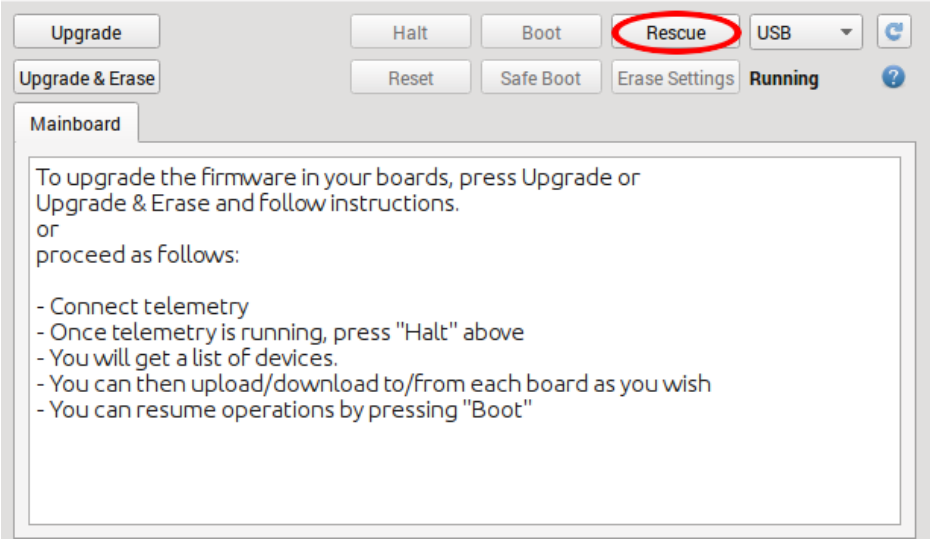
Once the board is halted, the bootloader (BL) version of the board is shown.



Click **Boot** to restart the flight controller once you have noted the bootloader version.



When the board is connected, its bootloader (BL) version is shown. Click **Boot** to restart the flight controller once you have noted the bootloader version.



Rescue Method

Ensure that the flight controller is **not** connected to GCS via USB. Click **Rescue** and connect the flight controller via USB when prompted.

Downloads

Bootloader Versions

- **Version 1** - First bootloader version which was loaded onto all CopterControl boards.
- **Version 2** - Safe boot capabilities were added to prevent the user from being locked out of the board due to a bad hardware configuration.
- **Version 3** - Added different USB serial numbers for bootloader and firmware. This helps Windows separate the two functions of the board by making it think there are two different boards connected. Twitching servo movement during the board start has been eliminated.
- **Version 4** - Added support for internal settings erasure so that no more special firmware is required; simply enter boot mode and click Erase Settings (CC/CC3D/Atom and OPLM)
- **Version 5** - Added better F4 processor support (Revo and OSD)

The bootloaders for CopterControl (CC), CopterControl3D (CC3D), Atom, Revolution (Revo), OPLink mini (OPLM) and On Screen Display (OSD) are available for download below.

Caution

All OP boards are different and require the appropriate bootloader to be installed, please ensure you download and use the correct version listed below.

Also note that you should have the appropriate GCS installed (in most cases, the latest version) on your PC when you flash the new bootloader so that the following Auto Update you perform will automatically install the correct version of firmware that is embedded in the version of the GCS on your PC.

The PCB for CC or CC3D boards should have "CopterControl" or "CC3D" printed clearly on the board. This indicates which bootloader needs to be flashed to the board.

Version 3 bootloaders can only be used with GCS versions 12.10.2 or older. Version 4 bootloaders can only be used with GCS versions 13.06.01 and newer.

Bootloader Updater Files

Board	BL Ver.	Updater Bootloader	Note
CC	3	bu_coptercontrol-20120630_5a1efef3.opfw	For use with GCS 12.10.2 and lower
CC3D	3	bu_CC3D-20120620_f44b9d3.opfw	For use with GCS 12.10.2 and lower
CC	4	CC V4 bu_cc.opfw	For use with GCS 13.06.01 and higher
CC3D/Atom	4	CC3D V4 bu_cc3d.opfw	For use with GCS 13.06.01 and higher
Revolution	5	Revo V5 bu_revolution.opfw	For use with GCS 13.06.01 and higher
Revolution	6	Revo V6 bu_revolution_6.opfw	For use with LibrePilot
Revolution Nano	6	Revolution Nano V6 bu_revonano.opfw	For use with LibrePilot
OPLMini	4	OPLink Mini bu_oplinkmini.opfw	For use with GCS 13.06.01 and higher
OSD	5	OSD bu_osd.opfw	For use with GCS 13.06.01 and higher

How to Upgrade the Bootloader and Erase Settings

If it is necessary to upload the bootloader, **strictly** follow these instructions:

- Download the appropriate bootloader (i.e. CC3D - BL4 or Revo - BL5) and save it to your hard drive where you can find it again
- Using the **Firmware** workspace in GCS, and with the board disconnected from USB, click **Rescue** and follow the onscreen instruction to connect the board
- After the board is detected, click **Open** and select the BootloaderUpdater (BU) file from where you saved it on your hard drive
- Click **Flash** to flash it to your board
- After the flashing is complete, press **Boot** and wait until the blue LED is on, then flashes, and finally goes off (normally ±15 seconds).
- Wait 10 seconds more.
- Disconnect the board from USB.
- Click **Rescue**, connect the board, click **Erase Settings** and wait for the erasure to complete.
- Disconnect the board from USB.
- Click **Upgrade** and follow the onscreen instruction to connect the board to automatically install firmware.

LED Behavior

- A slowly blinking blue LED indicates that the board is booted and running the firmware; this is the normal operating mode.
- Bootloader mode. A slow fading in and out of the blue LED with the green LED on indicates that the board is in bootloader mode.
- A rapidly blinking blue LED during a bootloader update indicates an error state. An invalid bootloader image was likely detected and the update hasn't been performed.

FAQs

What's the difference between firmware, bootloader (BL) and bootloader updater (BU)?

The **firmware (FW)** is the application to be loaded by the bootloader after the board has been powered up and initialized. The firmware is regularly updated and newer firmware typically include new features and bug fixes. The firmware and GCS version must match in order to be able to configure the board.

The **bootloader (BL)** is a small piece of software which is started as soon as the board is powered up. Every board ships with a bootloader preloaded and is not normally required to be upgraded by the user.

The **bootloader updater (BU)** is a special firmware which is loaded by the current bootloader and replaces the old bootloader with the new bootloader which it contains. This approach is required because the bootloader can't erase and overwrite itself.

The bootloader version isn't updated after the update.

There is a built in check that prevents the user from updating the bootloader with an incompatible version. If the blue LED blinks rapidly and continuously during the upgrade process, the updater is in an error state. Reboot the board and repeat the process using the correct bootloader updater.

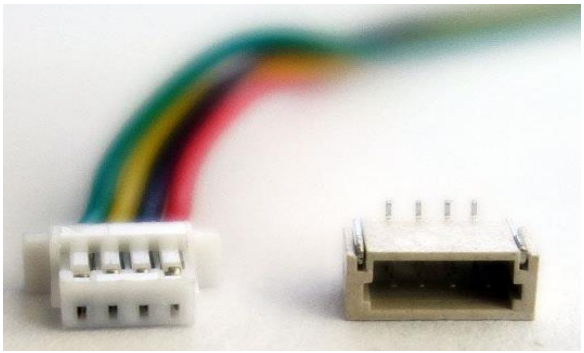
JST-SH Connectors

Description

JST-SH is a 1.0 mm pitch/ disconnectable crimp style connector used on several of the OpenPilot boards. It is used for its small size and reliable connection. Do not confuse it with JST-ZH or JST-XH that are different connectors and not used in the OpenPilot project.

Uses

The Revolution is using the JST-SH 4-pin for telemetry, GPS, UART and other IO ports.



CopterControl is using the JST-SH 4-pin for telemetry and the FlexiPort. It also uses a JST-SH 8-pin on the ReceiverPort for PWM or PPM input, and optionally for PPM input and servo output.

For more information, please refer below sites.

https://www.openpilot.org/products/openpilot-coptercontrol-platform/

http://opwiki.readthedocs.io/en/latest/user_manual/cc3d/cc3d.html