

[Guides](#)[Current](#)[ESC Telemetry](#)

ESC Telemetry

There are two ways to get ESC data to the flight controller:

- DShot Telemetry, which means sending data back from the ESC over the same single DShot signal wire that is being used to control the motors
- Serial ESC Telemetry, where one or more ESCs are connected to the flight controller by a serial port.

NOTE

This document explains the process of getting RPM and related data from the ESC to the Flight Controller. It does not discuss how to get that data from the Flight Controller back to the radio receiver.

DShot Telemetry

DShot RPM Telemetry

JoeLucid, in March 2019, developed the bidirectional DShot code to send RPM data back to the Flight Controller via the single DShot communication wire. RPM telemetry over DShot was a significant improvement over serial ESC telemetry because it updated the data much faster, and required no additional wiring. It is now supported by nearly all DShot ESC firmware.

DShot RPM Telemetry provided the following user benefits:

- **RPM filtering**, where dynamic notch filters tracked each motor's center frequency - 12 dynamic RPM filters for a quadcopter - greatly improving RPM based noise rejection
- **Dynamic Idle**, where the drive to the motors can be rapidly changed, under PID control, so that a motor's RPM does not fall below a defined minimum RPM value. Compared to a simple minimum idle throttle value, this reduces the likelihood of a desync and allows the PIDs to use the full range of motor drive values.

- **RPM Limiting**, which allows the user to limit maximum average RPM, for spec class racing.
- RPM logging for testing and debugging. Since Debug values can be shown in the OSD, we can display 'live' RPM values in the OSD.

Extended DShot Telemetry

This sends additional parameters such as ESC Temperature, Voltage, Current and others to the FC. It is not yet supported on all ESC firmwares, with Daniel Mosquera's **Bluejay** being most up-to-date.

We don't have a wiki article about EDT, but it is described in these [Betaflight Github EDT PR's](#) and on this [bird-sanctuary's GitHub page](#).

Serial ESC Telemetry

The following information is quite old and may not be accurate.

Requirements:

- Supported flight controller with Betaflight version 3.1.0 RC1 or later.
- Betaflight App version 1.8.5 or later.
- ESC with Serial Telemetry enabled (usually requires AM32 or BLHeli32)
- A spare hardware UART on the flight controller.
- (Optional) Telemetry from flight controller to RC receiver.

Procedure:

Install the ESCs to the quad and connect the ESC's Tx pin, and ground, to the RX pin of the spare hardware UART on the flight controller. With individual ESCs, a loom to split the single wire from the RX pin of the UART into four wires (one for each ESC) will be required.

Open the Betaflight App (V 1.8.5 or later) and go to the Ports tab.

Find the UART that the ESC telemetry is connected to and in the column marked "Sensor Input", select ESC in the left box and leave AUTO in the right box. Don't forget to hit the "save and reboot" button.

Still in Betaflight App, switch to the Configuration tab.

Make sure you are using the DShot protocol for ESC communication!

Under "Other Features", make sure that "ESC_SENSOR" is enabled.

On the Battery Voltage section enable "VBAT" and select ESC Sensor from the drop down menu below. Configure the cell voltages as required. (Defaults should be fine)

On the Current Sensor section enable "CURRENT_METER" and select ESC Sensor from the drop down menu below. Configure the scale and offset as required. (Defaults should be fine)

To test, set it all up then plug in the flight controller as well as the battery, open the configurator to the configuration tab and arm via the tx within a few seconds of powering on the flight controller. PROPS OFF ON THE BENCH!

Confirm that telemetry is being received by the transmitter.

If using a Taranis, don't forget to arm the quad when you "Discover Sensors" or the ESC telemetry won't be discovered.

Taranis users may want to change the "FUEL" sensor from % to mAh for an accurate reading.

Youtube

 [Joshua Bardwell](#)

 [Ivan Efimov](#)

Community

 [Discord](#)

 [IntoFPV](#)

Links

 [Oscar Liang](#)

 [VitroidFPV](#)

Feeds

 [RSS](#)

 [Atom](#)

 [JSON](#)



Copyright © 2025 All rights reserved Team Betaflight

Built with Docusaurus

made with ❤️ by **VitroidFPV** and **un!t**