JBardwell F7

Flight Controller

Pilot's Manual

V1.3 – March 30, 2021

For Hardware Revision 1.0

RaceDayQuads / JBardwell F7

I'm Joshua Bardwell. This is MY flight controller. No compromises. I made a list of all the features I wanted, and then they made it for me. It's backed by RaceDayQuads' top-notch customer support. It's affordably priced. Oh, and let's not forget, technical support from none other than me: the FPV Know-It-All himself.

Input Voltage:	3S to 6S
Processor:	STM32F7
Gyro:	ICM20602
Power Outputs:	5v
	3.3v
	9v
	vBat
Form Factor:	30.5mm square
Betaflight Target:	JBF7ANALOG
	JBF7DJI
UARTs:	5
	SDA/SCL (i2c)
Interfaces	4-in-1 ESC plug
	DJI Air Unit plug

Manual written, and feature set selected by Joshua Bardwell.

Board layout focuses on flexibility and resilience. You can build what you want, how you want, without stress.

4-in-1 plug also has solder pads for those who want to direct-solder.

Plug-and-play support for DJI Air Unit. Voltage regulator allows up to 6S input.

Separate DJI and ANALOG targets with default settings customized for each use.

8 motor pads allow for easy resource remapping (or build an octocopter).

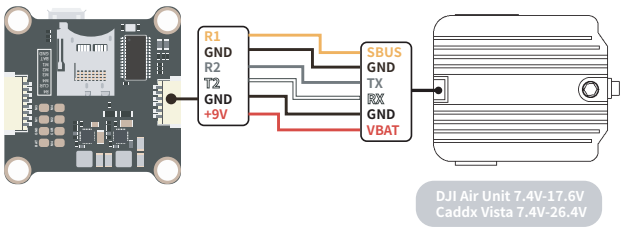
SD card for blackbox logging.

RaceDayQuads JBF7 Wiring Diagram

DJI Digital Transmitters

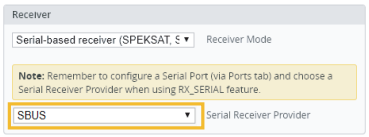
Firmware: JBF7 (Analog preset target)
JBF7_DJI (DJI preset target)

FC plug&play port and setup
compatible to DJI Air Unit and Caddx Vista



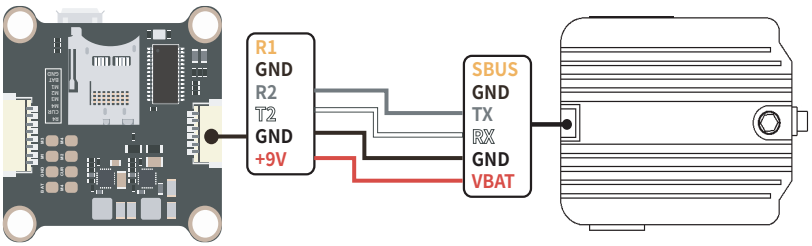
Identifier	Configuration/MSP	Serial Rx
USB VCP	115200	
UART1	115200	
UART2	115200	
UART3	115200	
UART4	115200	
UART5	115200	

- **Please check your protocols, otherwise your DJI Radio won' t input signals!**
- **DJI Goggle protocol and Betaflight protocol has to match!**
- **For lower signal latency use the SBUS_BAUD_FAST protocol option on both ends.**
- **For Betaflight Copy\Paste "set sbus_baud_fast=on" into your Betaflight Configurator CLI then hit enter.**
- **Use "save" and hit enter to save the changes.**
- **Default: sbus_baud_fast=off, Goggle protocol set to NORMAL**

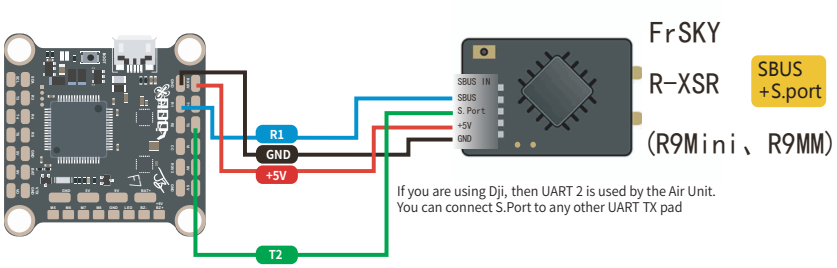


UART2 MSP: OSD passthrough

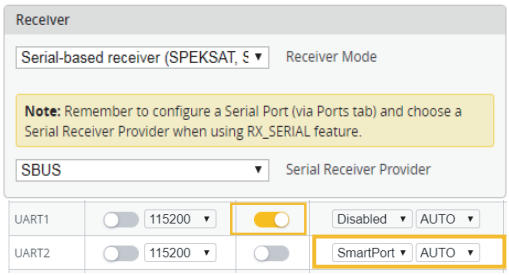
Any other Receiver



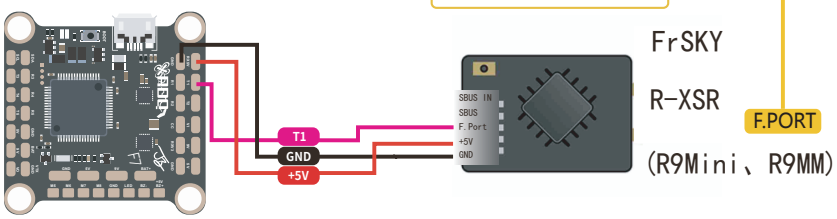
Identifier	Configuration/MSP	Serial Rx
USB VCP	115200	
UART1	115200	
UART2	115200	
UART3	115200	
UART4	115200	
UART5	115200	

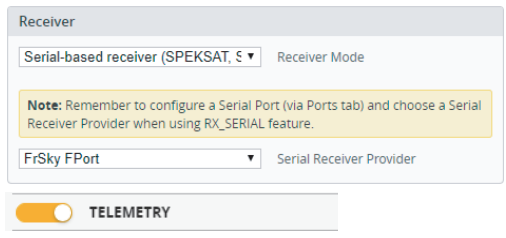


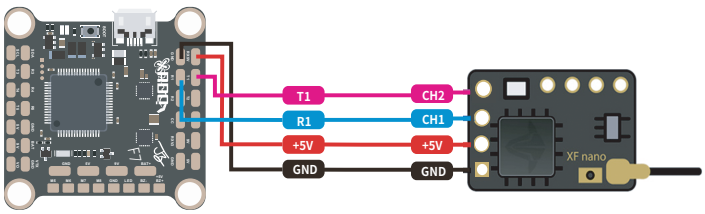
If you are using Dji, then UART 2 is used by the Air Unit.
You can connect S.Port to any other UART TX pad

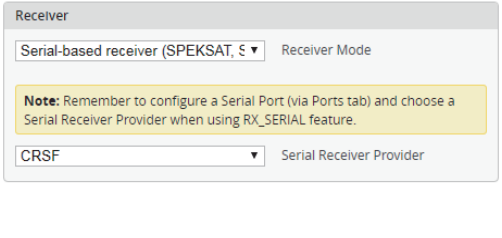


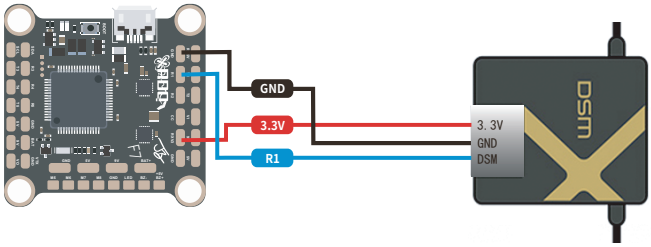
set serialrx_provider=FPORT
set serialrx_inverted=ON
set serialrx_halfduplex=ON

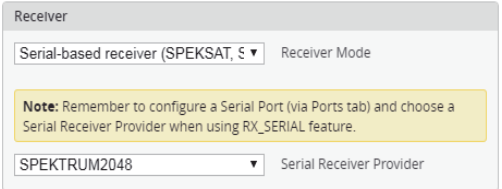


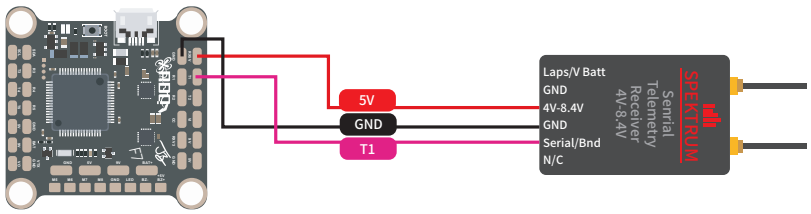










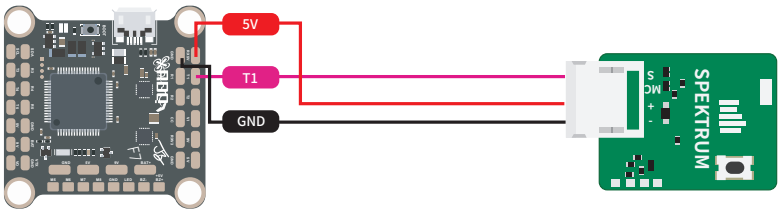


Receiver

Serial-based receiver (SPEKSAT, S ▾ Receiver Mode

Note: Remember to configure a Serial Port (via Ports tab) and choose a Serial Receiver Provider when using RX_SERIAL feature.

SPEKTRUM2048/SRXL ▾ Serial Receiver Provider



Receiver

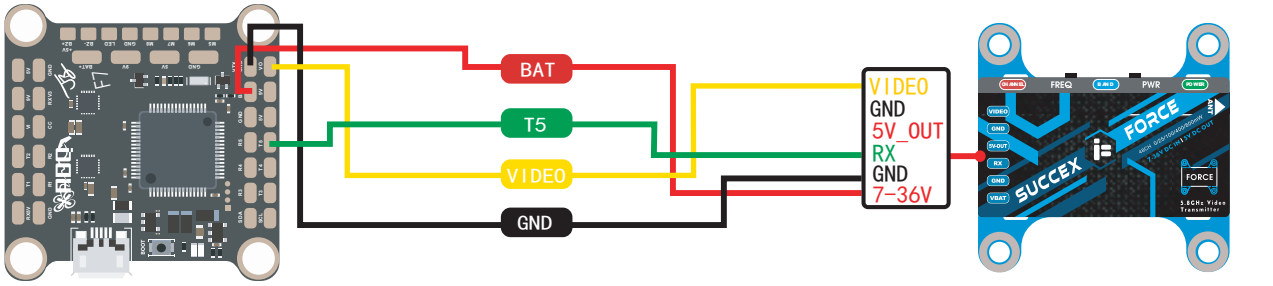
Serial-based receiver (SPEKSAT, S ▾ Receiver Mode

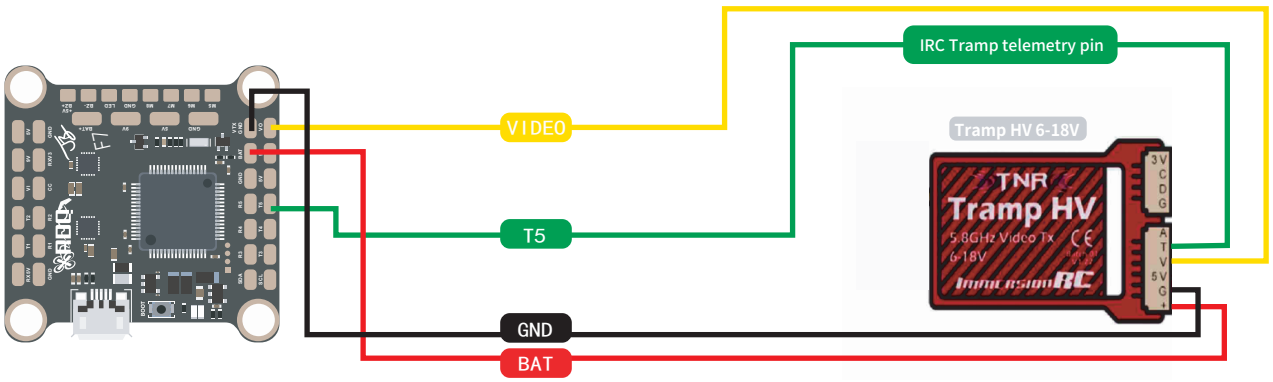
Note: Remember to configure a Serial Port (via Ports tab) and choose a Serial Receiver Provider when using RX_SERIAL feature.

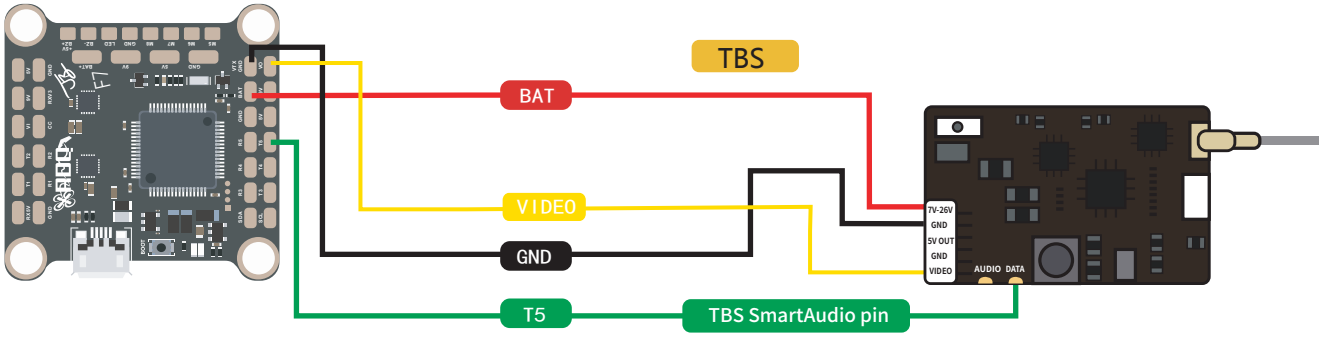
SPEKTRUM SRXL2 ▾ Serial Receiver Provider

VTX

The video transmitters shown here are all powered from battery voltage. Some VTX will have cleaner video if powered from the 9v pad. If you have video noise, then moving the vTX power from BAT to 9v may help. In addition, a few vTX (most notably the TBS Unify Pro) can only be powered from 5v. Always check your vTX input voltage rating!



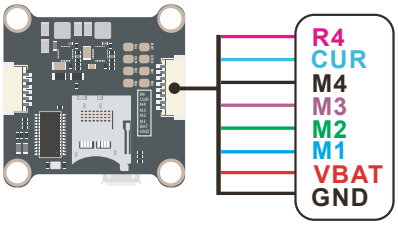




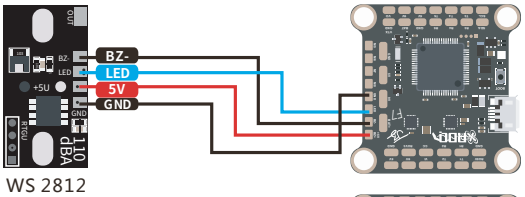
Identifier	Configuration/MSP	Serial Rx	Telemetry Output	Sensor Input	Peripherals
USB VCP	115200		Disabled AUTO	Disabled AUTO	Disabled AUTO
UART1	115200		Disabled AUTO	Disabled AUTO	Disabled AUTO
UART2	115200		Disabled AUTO	Disabled AUTO	Disabled AUTO
UART3	115200		Disabled AUTO	Disabled AUTO	Disabled AUTO
UART4	115200		Disabled AUTO	GPS 9600	Disabled AUTO
UART5	115200		Disabled AUTO	Disabled AUTO	VTX (TBS Sm AUTO

Disabled
Blackbox logging
VTX (TBS SmartAudio)
VTX (IRC Tramp)
Camera (RunCam Protocol)
Benewake LIDAR

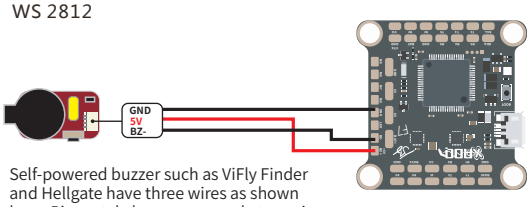
ESC



LED/BUZZER



WS 2812



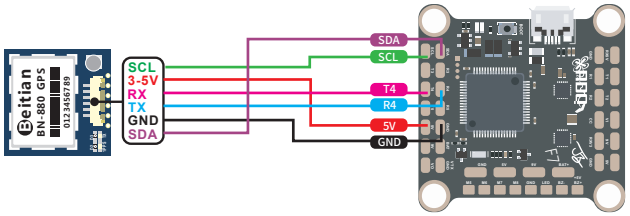
Self-powered buzzer such as ViFly Finder and Hellgate have three wires as shown here. Piezo style buzzers use only two wires, 5v and BZ-.

GPS

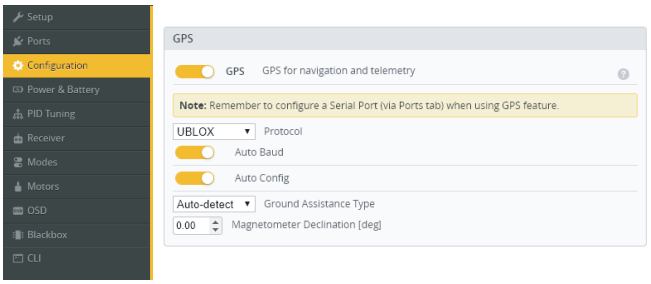
Identifier	Configuration/MSP	Serial Rx	Telemetry Output	Sensor Input	Peripherals
USB VCP	115200		Disabled AUTO	Disabled AUTO	Disabled AUTO
UART1	115200		Disabled AUTO	Disabled AUTO	Disabled AUTO
UART2	115200		Disabled AUTO	Disabled AUTO	Disabled AUTO
UART3	115200		Disabled AUTO	Disabled AUTO	Disabled AUTO
UART4	115200		Disabled AUTO	GPS 9600	Disabled AUTO
UART5	115200		Disabled AUTO	Disabled AUTO	Disabled AUTO

1

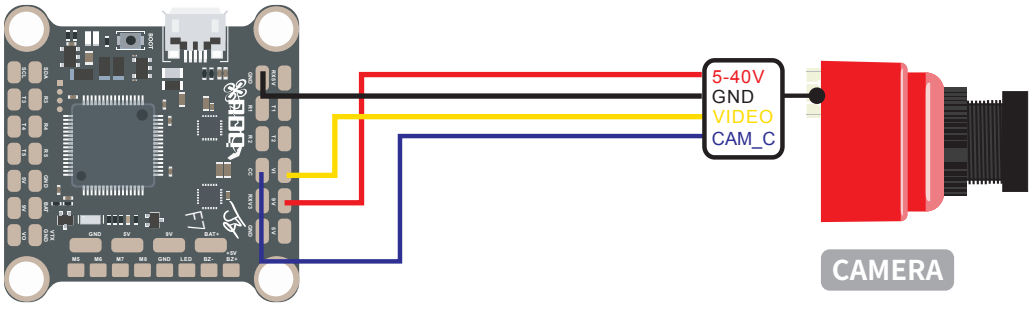
SDA/SCL pads are using the UART3 interface! Please make sure it's not occupied by R3/T3.
Some GPS do not have SDA/SCL wires, which is usually the additional Magnetometer.
If UART4 is occupied, please use any other UART to connect your GPS module.

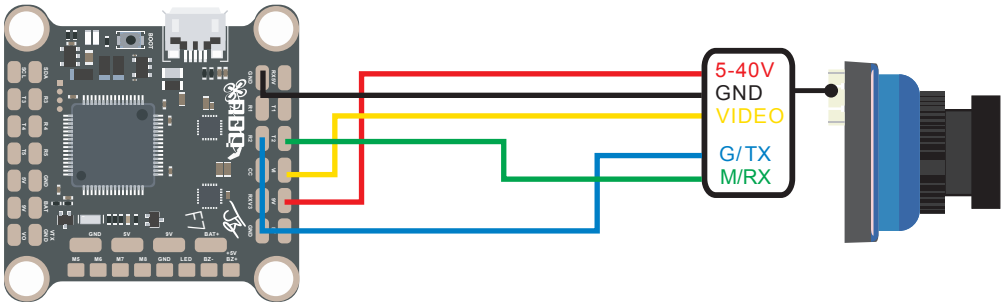


2



CAM





Functional Quick-Reference

The table below shows the UART and solder pad assignments for major features. Because this is an F7 flight controller, **you can use any UART for any function that you want**. There are no issues with SBUS and SmartPort inversion, like with F4 processors.

However, keep mind these restrictions:

A single UART cannot be used for more than one function. For example, if you are using an iBus receiver on RX6, then TX6 cannot also be used for ESC telemetry.

You cannot use both TX and RX on the same UART for two different functions. For example, you cannot use ESC Telemetry on RX4 and SmartAudio on TX4 at the same time.

Function	UART	Location
SBUS signal from DJI Air Unit (yellow wire)	1	DJI Plug
MSP signal from DJI Air Unit (gray and white wires)	2	DJI Plug
ESC Telemetry	4	4-in-1 Plug

The resources above are part of the 4-in-1 ESC plug or the DJI Air Unit plug. If you use these plugs, then you cannot use these pads elsewhere on the board. For example, if you plug in the DJI Air Unit, it will use UART 1 for SBUS and UART 2 for MSP, and you cannot use the T1/R1 and T2/R2 pads elsewhere on the FC.

The suggested UARTs shown below are based on our assumptions about what wires will be going where on the quad. The goal is simply to keep wiring neat and logical. For example, GPS is suggested for UART 4, because those pads are next to the SDA/SCL pads, which the GPS might also need. But you could put GPS on any other UART if you wanted to.

Function	UART	Pad
SBUS Receiver, IBUS receiver, Spektrum receiver using SPEKTRUM protocol (not SRXL)	1	R1
FPORT Receiver. Spektrum receiver using SRXL.	1	T1
Crossfire Receiver	1	T1/R1
SmartPort Telemetry	2	T2
vTX Remote Control (SmartAudio, Tramp Telemetry)	5	TX5
Programmable LEDs	N/A	LED
GPS Compass (I2C pads cannot be used at the same time as T3/R3)	4 I2C	T4/R4 SDA/SCL
Runcam digital camera control	2	T2/R2

Board Layout Quick-Reference

The table below shows the intended purpose of each of the pads on the board. Detailed instructions for wiring up the board are later in the manual.

TOP OF BOARD

RX 5v RX 3V3	5v or 3.3v output for receiver power. This output is active when plugged in to USB. Allows configuring receiver functions without plugging in battery.
GND	Ground
T1 / R1 T2 / R2	UART1 and UART2 connection. Intended for use with receiver and/or Runcam digital camera control.
VI	Video Input. Video wire from camera.
CC	FPV Camera OSD Menu Remote Control function. Connect to camera OSD input pin.
9v, 5v, BAT, BAT+	9v, 5v, or battery voltage output.
SDA/SCL	I2C interface pads most commonly for use with compass/magnetometer in GPS unit. You cannot use the TX3/RX3 pads if you use I2C.
R3/T3	UART3 connection. Suggested for use with GPS unless compass is used on SDA/SCL.
R4/T4	UART4 connection. Suggested for use with GPS if compass is used on SDA/SCL.
R5/T5	UART5 connection. Suggested for use with video transmitter remote control (SmartAudio) due to proximity to vTX pads.
VO	Video output. Video wire to video transmitter.
LED	Programmable LED output pin. Usable with Betaflight LEDStrip feature.
BZ-	Negative leg of piezo buzzer.
BZ+	Positive leg of piezo buzzer. This pad can also be used as a generic 5v output if need be.
VCC	Battery voltage. Primarily intended to be used for powering a video transmitter, since some vTX do better on vBat than on 9v output. Can also be used to power LEDs if they are rated for battery voltage.
VIDEO	Video out to FPV video transmitter.
M5, M6, M7, M8	Motor outputs 5-8. Can be used as spare outputs if a pad lifts, or can be reassigned to different resources, such as servo outputs, if needed. Or you can build a hexacopter or an octocopter if you want.

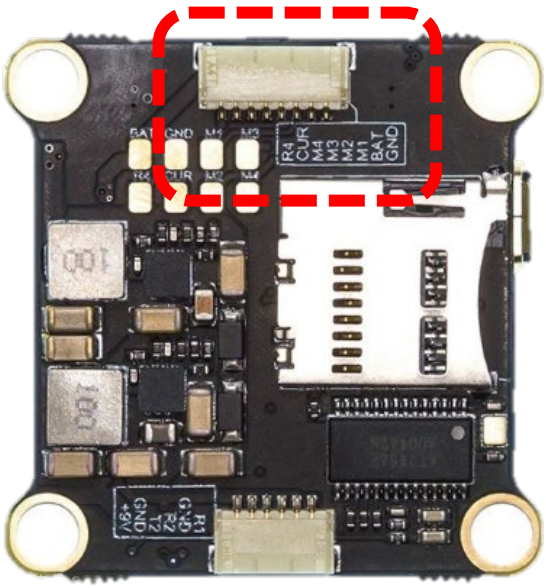
BOTTOM OF BOARD

BAT	Battery voltage input from ESC.
GND	Ground
R4	UART4 RX, used for ESC Telemetry from ESC
CUR	Analog current sense input from ESC
M1, M2, M3, M4	Motor outputs 1-4.

4-IN-1 SOCKET

BAT	Battery voltage input from ESC.
GND	Ground
R4	UART4 RX, used for ESC Telemetry from ESC
CUR	Analog current sense input from ESC
M1, M2, M3, M4	Motor outputs 1-4.

The 4-in-1 plug makes connecting the JBF7 to a 4-in-1 ESC easy. The plug contains connections for power, ground, motor signal 1-4, and analog current sensing or ESC telemetry.



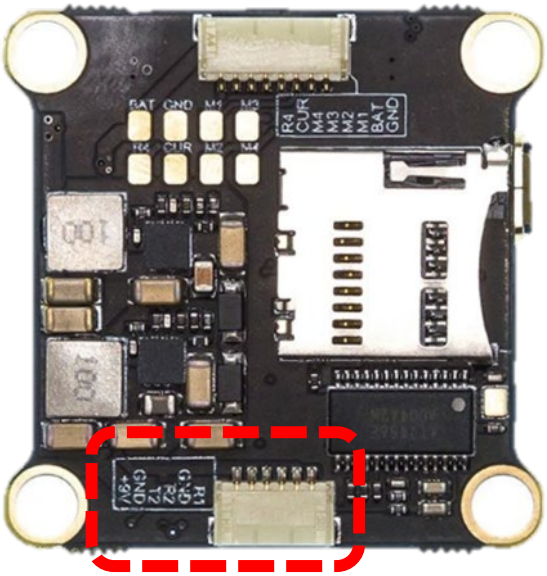
WSince there is no universal standard for 4-in-1 ESC plugs, you cannot simply plug your ESC into the JBF7 and assume that everything will work. In fact, you might fry your ESC and/or the JBF7 if you did this! The plug on the JBF7 has been designed to be directly compatible with recent iFlight ESCs, however even if you are using one of these ESCs, **you should manually verify that the pinout is compatible before you plug in the ESC**. Damaging the JBF7 or your ESC by plugging in incompatible ones will not be covered under warranty.

If you need to make your own custom wire harness to connect your ESC to the JBF7, RaceDayQauds sells a pre-crimped wire header set that is perfect. It can be purchased here: https://www.racedayquads.com/products/sh-1-0-silicone-cable-set

The 4-in-1 plug has solder pads next to it in case you damage the socket or if you prefer to direct-solder the wires to the FC.

DJI SOCKET

+9V	9v output from regulator for DJI Air Unit (allows use of up to 6S battery on the FC)
GND	Ground
T2/R2	UART2 connection for MSP interface (allows battery voltage and OSD to appear in DJI goggles)
R1	UART1 connection for SBUS control from DJI Air Unit



The DJI plug allows direct connection of a DJI Air Unit to the JBF7 without any soldering. The JBF7 includes a wire harness that goes from DJI plug on the JBF7 to the DJI Air Unit.

All you need to do is plug the Air Unit into the DJI plug on the JBF7, then flash the "JBF7DJI" target to the FC (covered later in this manual) and you have full functionality of the DJI Air Unit.

Soldering Tips

There's an uncomfortable truth in this hobby: a whole lot of us are much better with a transmitter than we are with a soldering iron. We have designed the JBF7 to be resilient, but any board can be damaged if it is soldered on incorrectly. Before you take a soldering iron to your board, here are some tips to help make sure you're not making a call to support, trying to convince them you, "have no idea how that pad lifted off I barely touched it with the iron and it just came off!"

1. We suggest using a temperature controlled iron if at all possible. This type of iron has a temperature sensor in its tip and adjusts its output power to keep the tip at a constant temperature. If your iron doesn't have adjustable temperature, you will need much more skill and experience to get good results. You don't have to spend $100 on a high-end Hakko or Weller iron to get temperature control. There are budget irons available in the $30 price range that will do the job.
2. We suggest a temperature of about 400°C / 750°F. Some beginners believe that a colder temperature is safer because it's less likely to damage the board, but this isn't true. A cold iron requires more time in contact with the board, and makes it harder to get a good joint. So, you are just as likely to damage the board if the iron is too cold, as if it is too hot.
3. We suggest a chisel tip, approximately 2mm wide for the signal and ESC/motor wires. A thin pencil tip will not transfer heat effectively. A heavy bevel tip will be hard to maneuver precisely.
4. Be sure the tip is clean. If the tip has black or grey oxidation on it; if the tip has been damaged by abrasive or chemical cleaning; if the tip has been damaged by overheating; then you are more likely to damage the board. The tip should be shiny silver and the solder should stick to the tip easily and not want to "ball up" on the tip.
5. Use leaded solder. Yes, lead is bad for you, but you don't get significant lead exposure when soldering. Unless you eat the solder. You're not going to eat the solder, right? Okay… then you're fine. Use leaded solder. Lead free solder is much harder to solder with. It's largely intended for commercial use, where keeping lead out of landfills is important.
6. Before soldering the joint, prep the joint and wire. Tin the pad by putting a small amount of solder on it. Then strip the wire, twist the strands together, and tin the wire. Snip the wire to length. Then hold the wire against the pad and touch both the pad and the wire with the hot iron to flow the solder and complete the joint.

Updating Betaflight Firmware

Like all software, the software that runs your flight controller has versions. Just like Windows XP was followed by 2k, then 7, 8, and 10. The software that runs your flight controller is called Betaflight. Putting a new version of Betaflight on your JBF7 is called “flashing” your board.

Even if you decide you don't want to update your firmware right now, you still need to install the VCP driver to configure the board. So, you must at least go through step 1 below to use your board.

Installing Drivers

Before you can flash your board, you must get the drivers installed on your computer. If you are on MacOS or Linux, Your O/S comes with the drivers you need pre-installed. You do not need to perform this step. You may skip directly to “Installing Betaflight Configurator”.

Depending on the version of Windows that you have, you might need to install the drivers manually. The most recent versions of Windows 10 seem to have the VCP driver pre-installed, so you should try simply plugging in the FC and see if things work correctly.

If you plug in the FC and you don’t see a new COM port appear in the configurator, then you need to manually install the drivers. This is a place where beginners often struggle. We’re going to present the steps here, and if you can follow them, then great, but many people won’t be able to follow them. If you need a more personal approach, here is a link to a video that walks you through the process in more detail.

If your version of Windows does not have the drivers pre-installed, here are the steps you will perform:

1. Download the Virtual COM Port (VCP) driver installer from here. Unfortunately, you must either create a login or give them your email address, then wait for them to email you a link, to download the actual installer. We recommend that you create a user login because the "send you a link in an email" method sometimes takes a long time to go through.
    a. To create a login at the STM site, first go to this URL.
    b. Click "Create an Account"
    c. Enter your information in the next page and click "Register"
    d. You will receive a confirmation email. Complete the confirmation process.
    e. Log in to the STM site using the login you just created and download the VCP driver installer from the link in step 1.
2. Run the VCP installer and let it finish.
3. Download the ImpulseRC Driver Fixer from here.
4. Run the ImpulseRC Driver Fixer. It will instruct you to plug in your flight controller.
5. While holding down the bootloader button on the FC, plug the FC into your PC via USB. The ImpulseRC Driver Fixer should complete successfully.

Installing Betaflight Configurator

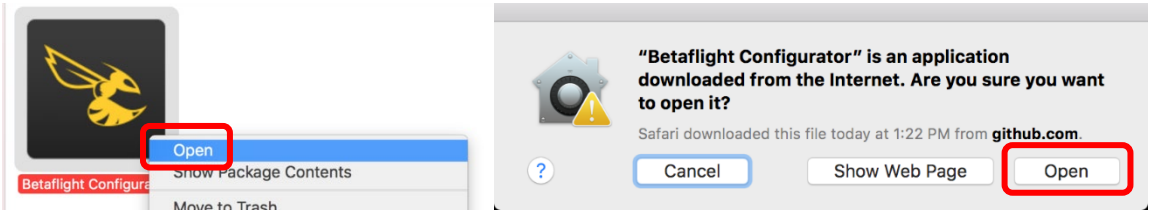
Betaflight is managed using the Betaflight Configurator application, also known for short as the Betaflight GUI, or just, "The GUI". (GUI is pronounced "gooey" in case you wondered. Only weirdos say, "Gee You Eye".)

You can download the Betaflight Configurator from here:

https://github.com/betaflight/betaflight-configurator/releases/

This is a standard application package for Windows, MacOS, or Linux. After you download it, install and run it the same as any other application.

On MacOS you must give the application permission to run and access the network. The simplest way to do this is to right-click the installer and choose Open. A security prompt will appear asking, "Are you sure you want to open it?" Click Open on this prompt and the relevant configuration changes will be made automatically.

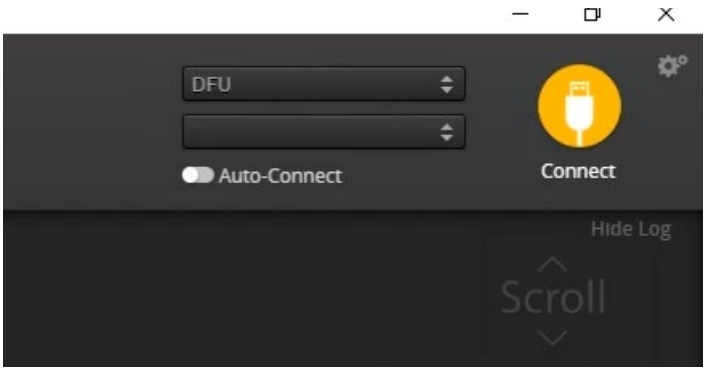


Flashing New Firmware

Why would you need to flash new firmware to your brand new FC? Shouldn't it have the latest and greatest? The first reason is that new versions of Betaflight are constantly being released, and it's possible that you haven't got the latest and greatest. The other reason is if you are using the JBF7 with a DJI Air Unit. In this case, you can flash the JBF7DJI target to the flight controller, which will pre-configure the default settings to work with the Air Unit. If you are using analog FPV gear, you can use the JBF7 target (which ships on the board).

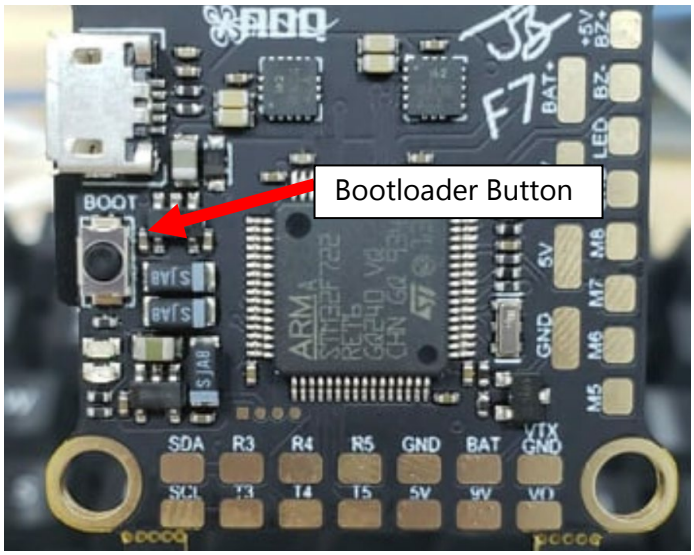
To flash your firmware, you must connect the board in “bootloader mode”. Bootloader mode means that the board is ready to accept new programming. When you try to flash firmware to the FC, Betaflight will try to automatically put the board into bootloader mode.

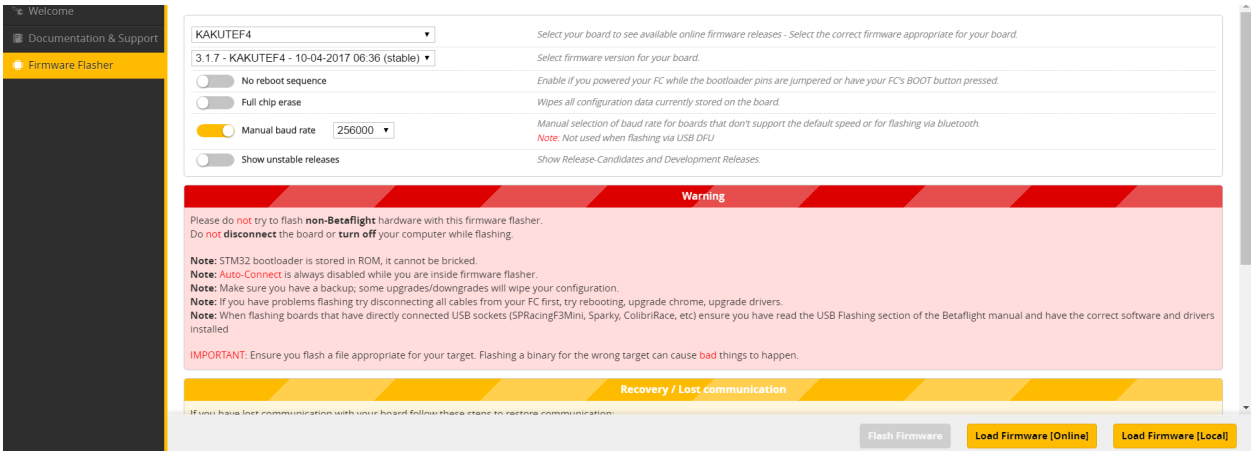
If your JBF7 is in bootloader mode, then you will see “DFU” in the pulldown menu in the upper-right of the configurator, as shown here:



Sometimes, the computer isn’t able to put the FC into bootloader mode automatically. You’ll know this has happened when you try to flash, and instead of seeing “DFU”, you see … anything else.

To manually put the board into bootloader mode, hold down the bootloader button while plugging in the USB cable. Leave the button pressed for a moment after plugging in the USB cable to be sure it “takes”.





Here are the remaining steps to flash your FC.

1. Go to the “Firmware Flasher” tab.
2. Select “JBF7” or “JBF7_DJI” in the “Choose a board” pulldown menu. If you flash any other board type, the JBF7 won’t function. It won’t be damaged, it just won’t work until you flash the correct target to the board.
3. Select the latest version of Betaflight in the “Choose a firmware version” pulldown menu.
4. Click the “Load Firmware (Online)” button. The button will change to read “Downloading”. The Flash Firmware button will change from gray to yellow.
5. Click the “Flash Firmware” button. The screen should automatically scroll down to show the status bar, which will fill with orange as the flashing process completes.
6. Flashing will be followed by a process called “Verifying”.
7. Un-plug your board and then plug it back in again, this time without holding down the bootloader button.
8. For Windows users, the pulldown menu in the upper right of the configurator will read COM3 (or some other number). For MacOS and Linux users, the pulldown will read something starting with /dev/tty. This is normal. If the pulldown menu reads “Manual Selection” then your board is not being detected. This may indicate that you did not use the JBF7 or JBF7DJI target when you flashed the board. Or it may indicate that your Virtual COM Port (VCP) drivers were not installed correctly.

You are ready to configure your board.

Saving Your Configuration

Once you have finished building, configuring, and tuning your multirotor, it's a good idea to back up your configuration so that you can restore it later. This is useful if you lose your quad, or if you damage your flight controller, or if you accidentally lock yourself out of your flight controller and must reset it to get back in.

Before we show you the right way to save and restore your configuration, let us warn you about the wrong way. Betaflight and Cleanflight have a "save configuration" and "restore configuration" button. Without going into too much detail, they have some significant drawbacks that mean we don't recommend that you use them.

The correct way to save your configuration is as follows.

- Connect your JBF7 to your PC by plugging in USB.
- Start the Betaflight GUI app.
- Go to the CLI tab.
- In the text box at the bottom of the CLI, type "diff all" and hit enter. This will cause the flight controller to display all configuration options that you have changed from the default values.
- In the lower-right corner of the configurator, click the "Save to File" button.
- Save the file somewhere you won't lose it.

To restore your configuration, do this:

- Open the text file in your text editor.
- Highlight the entire contents of the file.
- Right-click in the text editor window and choose "Copy".
- Connect your JBF7 to your PC by plugging in USB
- Start the Betaflight GUI app.
- Go to the CLI tab.
- Click the mouse once in the text box at the bottom of the CLI tab to place the cursor there.
- Instead of typing any commands, right-click in the text box and choose "Paste".
- Press the Enter key on your keyboard. The pasted-in text will rapidly scroll past.
- Type "save" in the text box at the bottom of the screen.
- Press Enter. The flight controller will reboot and the configuration will be restored.

Glossary of Terms

Flight Controller (FC) – The "brain" of the quadcopter. It takes commands from the pilot to know what the quadcopter should be doing. It reads data from sensors like the gyro and accelerometer to know what the quadcopter is doing. Then it sends commands to the ESCs and motors to make the quadcopter do what it should be doing. Modern flight controllers often have many non-flight-related functions built in, such as On-Screen Display (OSD).

Gyro – Short for gyroscope. A sensor that detects rotation.

Accelerometer – A sensor that detects acceleration. Often shortened to just "acc". Since gravity manifests as a downwards acceleration, the accelerometer is the main sensor that can tell whether the quad is right-side up or not. The accelerometer is required for autolevel flight modes.

Receiver – Receives RF signals from the transmitter. Translates them into a protocol that the FC can understand and forwards them to the FC.

SBUS, iBus, SpekSat – These are all protocols that different receivers use to communicate the pilot's commands to the FC. SBUS is used by FrSky devices primarily. iBus is used by FlySky devices. SpekSat is used by Spektrum satellite receivers. FrSky, FlySky, and Spektrum are three different brands or families of transmitter and receiver.

Transmitter – Takes control inputs from the pilot via sticks, switches, and so forth. Converts those control inputs to RF signals and transmits them to the receiver.

Telemetry – Allows the receiver to send data about the aircraft back to the transmitter, instead of only receiving commands from the transmitter. Example telemetry data might include battery voltage, battery capacity consumed, motor temperature and RPM, GPS coordinates, and so forth. When the receiver and transmitter support telemetry, it is possible to configure alarms on the transmitter, so that the pilot can know about the aircraft's status (e.g. "battery is low!") and respond to it (e.g. "land now!"). Most RC control systems support telemetry today, however not all receivers within a system support it.

ESC – Short for Electronic Speed Controller. Its function is to make the motors spin. Each motor requires its own ESC. The ESC receives throttle position commands from the FC and makes the motors speed up or slow down to match the commanded throttle position. ESCs may or may not support "ESC telemetry", a feature where the ESC can send data about the motor to the FC.

UART – A serial interface on the FC that is used to communicate with peripheral devices such as the receiver.

vTX (Video Transmitter) – Receivers video signal from the FPV camera and transmits it wirelessly into the air.

vTX Remote Control (SmartAudio, **Tramp Telemetry)** – Allows the FC to change the channel, transmit power, and other settings on the vTX, without the pilot having to touch the vTX directly. This allows the pilot to configure the vTX indirectly, such as through the OSD or the transmitter. SmartAudio and Tramp Telemetry are two different protocols that the FC can use to talk to the vTX. Which one you use will depend on which vTX you have purchased.

OSD – Short for "On Screen Display". Displays important data about the aircraft directly on top of the FPV feed, for the pilot to view in the goggles. OSD is built into most modern flight controllers. The data communicated by the OSD overlaps with that communicated by telemetry, although OSD may provide additional visual information that telemetry does not (e.g. artificial horizon).

FPV Camera – A camera, typically very small, mounted to the front of the quad. Sends a picture to the vTX so that the pilot can see in their goggles what the quad is doing. The JBF7 supports the FPV Camera OSD Menu Remote Control feature, which allows the pilot to adjust the camera's menu settings via the transmitter sticks, without plugging a joystick into the back of the camera.

LED, Programmable – One or more LEDs that can be commanded to change brightness and color by communicating over a digital serial interface. The JBF7 can communicate with programmable LEDs to customize their appearance in response to quadcopter movement, battery status, and other conditions.

Faq / Troubleshooting

Camera Control doesn't work

The JBF7 comes from the factory with a camera control pad already set up, and a resistor already installed. However, you should know that this isn't a guarantee that the feature will simply work with every camera. The feature is known to require some manual tweaking on many cameras. If yours works right out of the gate, wonderful!

For full details, please visit the Setting Up section of the camera control wiki page. In short, here is what you should try if it doesn't work right off the bat:

- Increase the camera control key delay by going to the CLI and typing, "set camera_control_key_delay=250" then press enter, then type "save" and press enter. If this results in double-clicks of the joystick, reduce the value.
- Change the camera control internal resistance value by going to the CLI and typing, "set camera_control_internal_resistance=100". Unfortunately, there does not seem to be a methodical way of deriving the correct value for this. People have reported it working with values that seem like they must be incorrect. Valid values range from 100 to 500, so work your way up in increments of 50 and see if one works.
- Change the camera control reference voltage by going to the CLI and typing, "set camera_control_ref_voltage=310". Values from 300 to 340 might produce good results.
- Especially if you are using a micro camera, you might need to install a capacitor on the camera side of the OSD circuit. Instructions for doing this are in the camera control wiki page, linked above.

If you are using a Runcam camera with digital camera control interface (it has TX/RX pins in addition to MENU/GND pins), you might need to switch the camera between joystick and digital control. When you first power up this type of camera, it will say either "UART CONTROL" or "JOYSTICK CONTROL" in the upper-left corner of the screen.

If you are using the CC pad on the JBF7, that emulates a joystick, and you would want the camera to say "JOYSTICK CONTROL". If that's working for you, stick with it. But it often doesn't work, in which case you would want to use digital (UART) control.

To switch the camera between UART and JOYSTICK control, you need to short together the TX and RX pins on the back of the camera. The easiest way to do this is to twist together two wires that go to a joystick plug and plug it in. Here's a video demonstrating the method.

Setup Tips for Cleanest Video

The JBF7 is designed so that most builds should not need any additional capacitors or filtering to have clean video. But noise can't be eliminated for every possible setup. Here are tips on how to setup your board for best video quality and the least likelihood of needing capacitors for perfect video.

The JBF7 has a filtered 9v and 5v power output for vTX and FPV camera. On paper, this provides the cleanest power supply for cleanest video. However, we have found that some vTX perform better when supplied with direct battery voltage. Although this is counter-intuitive, it's consistent enough that we're passing the recommendation on to you. If you're using the RaceDayQuads Mach 2, ImmersionRC Tramp, or TBS Unfiy HV, we suggest powering from vBat instead of the 9v regulator. Go figure.

We recommend that everyone else start by wiring the JBF7 as shown in this manual. If you finish soldering up your JBF7 and have noise, here are our tips:

- Adding a 330 uF, low-ESR, electrolytic capacitor to the power pads of your ESCs will totally eliminate noise in almost all situations. We recommend 25v to 35v rated capacitors for 4S, and 35 to 50v rated capacitors for 6S setups. If you are using a 4-in-1 ESC, add a single 1000 uF capacitor to the main battery power pads. Some 4-in-1 ESCs have separate vBat pads specifically for adding a capacitor; use those if they exist.

If you prefer not to add capacitors, consider one or more of the following options:

- Some vTX have a built-in voltage regulator for powering the camera. It typically outputs 5v. If your vTX can power your FPV camera, then wire your FPV camera directly to the output wire on your vTX, instead of to the power pad on the JBF7.
- If you are one of those purist pilots who doesn't like to use an OSD, you can run the FPV camera's video wire directly to the vTX, instead of using the CAM and vTX pads on the FC.
- Assuming that the FPV camera and vTX can both take the voltage, run both of them off the same power pad on the board. Not just, like, both of them running off of 9v, but literally the same physical pad on the board. Do the same for their ground wires.

If none of the above works, a final step can be to isolate the vTX SMA connector from the frame of the quad. (Assuming you're using an antenna with an SMA connector that is.) In rare cases, the SMA connector touching the frame causes a ground loop that causes horrible video noise.

Video transmitter won’t change channel or very short range

Betaflight 4.2 fundamentally changed how the video transmitter (smartaudio) interface works. Before you can use SmartAudio or Tramp Telemetry to change power and channel on your vTX, you need to configure a “vtxtable” in the FC. If the vtxtable is not configured correctly, the vTX will lock itself to its lowest output power, which can result in very short range, even when you think you should be at maximum output power. If you encounter any of these issues, please watch this video, which explains how to troubleshoot them.