Holybro

Kakute F4 Flight Controller

With Betaflight OSD

User Manual & Installation Guide

v1.1

Contents

Overview

The Holybro Kakute F4 flight controller supports important features of Betaflight/Cleanflight, such as OSD and Dshot. The faster F4 processor gives room for future feature development: run all features at the fast 8 kHz PID loop rate, at single-digit CPU utilization! Because of issues with serial inversion, other F4 boards struggle to support protocols such as FrSky SBUS and SmartPort at the same time. The Kakute F4 is designed to support all serial protocols at once.

Features

- Supports Betaflight and Cleanflight.
- Betaflight OSD. Change PIDs, adjust common configuration parameters, and change video transmitter channel and power level, all using your transmitter sticks and goggles.
- Soft-mounting built in. The IMU (“gyro”) chip on this board is mounted on vibration-isolating foam. This means that there is no need to soft-mount the board itself.
- New high-performance / low-noise / high sensitivity IMU. ICM20689 with 6-axis gyro and accelerometer. Can run at up to 32 kHz.
- Dedicated bootloader button for easy firmware flashing.
- Low-profile design fits into even very compact frames.
- Input voltage 7v to 42v. Power the board directly from the flight pack, up to 6S (on “BAT” pad only).
- Automatic voltage monitoring. No need to run a separate vBat wire for voltage monitoring; the Kakute F4 monitors voltage directly from the battery power lead.
- Filtered voltage output for clean, noise-free video. On-board regulators output 5v at up to 1.5 amps and 3.3v at up to 200 mA to power peripherals such as receiver, video transmitter, FPV camera, or LED strip.
- PPM input is shared with UART6 RX pad for simple receiver installation regardless of type.
- Supports BLHeli pass-through for easy ESC upgrade and configuration.

Specifications

- MCU: STM32F405RGT6 32-bit processor
- IMU: ICM20689 (SPI)
- USB VCP Driver (all UARTs usable simultaneously; USB does not take up a UART)
- 3 hardware UARTS (UART1,3,6)
- 128 Mbit Dataflash chip for Blackbox logging
- 5v Power Out: 1.5 A max
- 3.3v Power Out: 200 mA max
- Dimensions: 35x25x7mm (Includes USB height)
- Mounting Holes: Standard 30.5mm square to center of holes
- Weight: 5g

Warranty and Return Policy

If you believe that your Kakute F4 is defective, please contact us. If we determine that the board is defective, it will be repaired or replaced at no charge to you. We may ask you to send your Kakute to our service center for examination or repair. Shipping costs are your responsibility. Returned items should include the original packaging and all accessories.

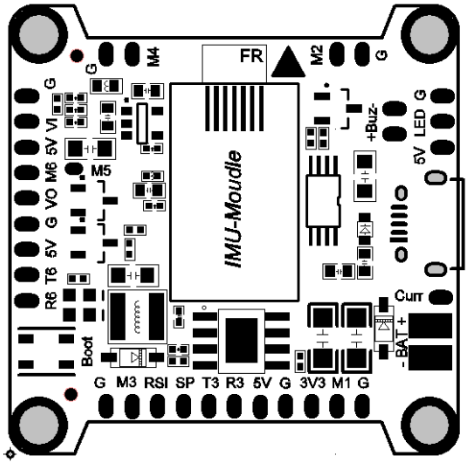
If product is damaged or defective, we will repair or replace it. Refunds are only given when product is lost by the shipping company. The refund amount is limited to the price of the product. Shipping costs are never refundable.

Contact us at:

- Email: productservice@holybro.com
- Facebook Page: Holybro
- Facebook Group: Holybro Shuriken Owner Group

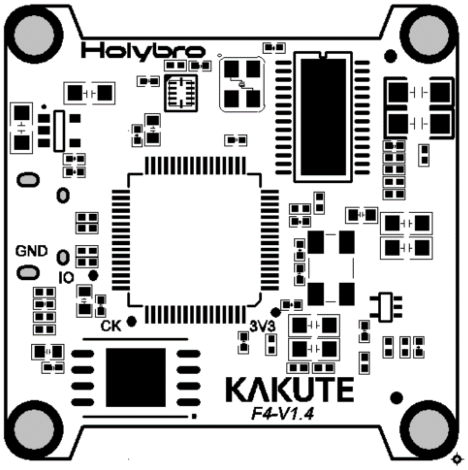
Pinout Diagram

Top of board



Pin	Function
BUZ-	Piezo buzzer negative leg
BUZ+	Piezo buzzer positive leg
LED	WS2182 addressable LED signal wire
SP	FrSky SmartPort Telemetry
R3, T3	UART3 RX and TX
R6, T6	UART6 RX and TX
RSI	Analog (0-3.3v) RSSI input
Curr	Analog (0-3.3v) Current sensor input
3V3	3.3v output (200 mA max)
5v	5v output (1.5 A max)
M1 to M6	Motor signal outputs
VO	Video output to video transmitter
VI	Video input from FPV camera
Boot	Bootloader button
G	Ground
B+	Battery positive voltage (2S-6S)

Bottom of board



Pin	Function
IO, 3V3, CK	Debugging interface for developers
GND	Ground

Installation Guide

Before you begin to install the Kakute F4, you should usually have your ESCS, and PDB fully installed in your frame and soldered up. The Kakute F4 will typically install on top of the PDB or a 4-in-1 ESC. If you install the Kakute F4 before soldering up the PDB and ESCs, you might have to remove it afterwards to get at the PDB.

Install the Kakute F4 in your quadcopter frame. Do not completely reassemble the frame. Leave the Kakute F4 accessible so that you can solder wires to it. For example, leave the top plate of your frame off, or remove the camera pod of an X-style frame so that you are working only with the base-plate.

Having the Kakute F4 mounted in the frame will allow you to ensure that wire lengths are correct as you solder accessories to the board. When in doubt, it's always better to leave a little extra wire.

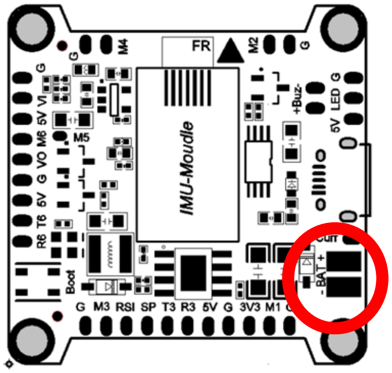
Be 100% sure that the front-facing arrow on the Kakute F4 faces the front of the quadcopter! If for some reason this is not possible, then you must use Cleanflight or Betaflight's "board align" feature to compensate.

"Board Align" Feature Documentation

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

Solder a wire from a battery-voltage source to the -BAT+ pads on the Kakute F4. The battery-voltage source could be the main battery pads on your PDB, but the PDB might also have a separate vBat accessory pad for this purpose. Because the Kakute F4 will only draw a few amps of current at most, this wire can be relatively thin: 22 to 26 gauge (the same as is used for typical servo wires) is fine.

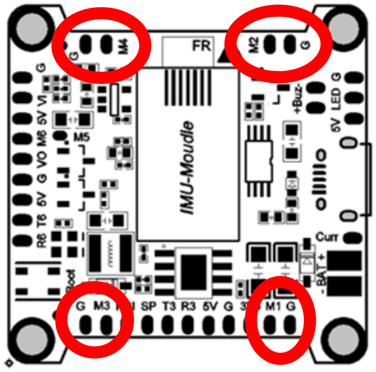
The wire should be fine-strand, silicone-insulation wire. This type of wire is commonly referred to as "silicone wire". **Do not use regular stranded copper wire.** It will become brittle over time and break.



When you route the wires to the Kakute F4, it's a good idea to route them in such a way that the board can later be lifted or folded out of the way. This will allow for easy maintenance later when it's required. Quadcopters crash. A lot. Make the repairs as easy on yourself as possible.

"Hinge Your Stack" Method of Wire Routing

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

Solder the ESC signal wires to the M1, M2, M3, and M4 pads. The signal wire is a thin wire, typically yellow or white. It's usually connected to the same side of the ESC as the power wires. Each ESC's signal wire should be soldered to the motor pad that is closest to the corner where the ESC is installed. If your ESC came "naked" (without wires attached) then you should use a piece of silicone wire between 22 and 26 gauge for the signal wire(s). Some ESCs have a signal ground wire and some ESCs don't. If your ESC has a thin black wire that is twisted with the signal wire, that is the signal ground. You can solder the signal ground to the Ground pad that is next to each M1, M2, M3, and M4 pad.	 G M4 FR M2 G G 5V VI VO M6 5V G R6 T6 M5 IMU-Moudle +Buz- 5V LED G Curr +BAT- Boot G M3 SP T3 R3 5V G M1 G

Solder the receiver signal wire to pad R3. If you are using a servo cable, the signal wire will be white or yellow.

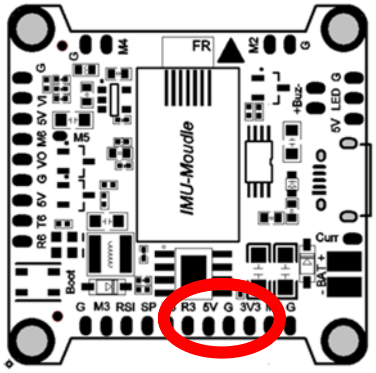
You must use a receiver that supports a serial protocol such as SBUS, iBus, and Spektrum Satellite. You cannot use a PPM or PWM receiver with the Kakute F4 AIO.

Solder the receiver ground wire to the G pad to the right of R3. In a standard servo cable, the ground wire will be black or brown.

If your receiver requires 5v power (most receivers except for Spektrum Satellite), solder its power wire to the 5v pad to the right of R3. In a standard servo cable, the power wire will be red or orange. The power wire is also always the middle wire of the servo cable.

If your receiver requires 3.3v power (most Spektrum Satellite receivers), solder its power wire to the 3v3 pad to the right of R3. **Do not connect a receiver that takes 3.3v power to a 5v pad or you will destroy it.**

Be sure to reference the pinout diagram for your receiver, to ensure that you are connecting the correct pads together.



Before wiring up your camera and video transmitter (vTX), you must determine whether you will power them by 5v or battery voltage (vBat). Many cameras and video transmitters today can run off up to 4S (16.8 volts) without issue. Some cameras and video transmitters can run off even higher voltage. You must refer to the product specifications for your camera and video transmitter to determine what maximum voltage they allow. These can typically be found on the product listing page of the vendor that sells the equipment.

If the voltage that you will use to power the copter (such as 4S or 5S) is too high for your camera or vTX, you must power them from the 5v regulator. If the voltage that you will use to power the copter is less than the maximum rated voltage of your camera and vTX, you may power them from vBat. However, we recommend powering the camera and vTX from 5v if possible, because the 5v regulator provides additional filtering, which helps ensure clean and noise-free video.

Remember that the rated load of the 5v regulator on the Kakute F4 is 1.5 amps. This means that the sum of the accessories you run from the regulator cannot exceed 1.5 amps. This should be enough current to run a camera, receiver, and video transmitter (even a high-powered vTX like the Unify Pro). However, if you have other 5v accessories, such as 5v LEDs, you might exceed the rated capacity of the 5v regulator. In that case, you would have to run some of the accessories from vBat, to remove load from the regulator. When accessories are run directly from vBat, they do not load the regulator.

To recap: the sum of the current drawn by all accessories on the 5v pads must be less than 1.5 amps.

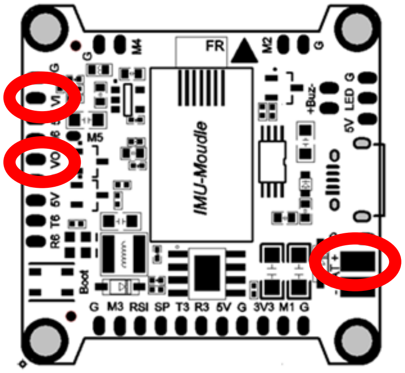
Solder the video wire from the video transmitter (vTX) to the VO pad on the Kakute F4.

Solder the video wire from the camera to the VI pad on the Kakute F4.

If you intend to run your vTX and camera from 5v, solder their power wires to the 5v pad that is right next to the VI pad. Twist the two wires together and tin them with solder before soldering them both at the same time to the 5v pad.

If you intend to run your vTX and camera from vBat, solder their power wires to any vBat source. On the Kakute F4, you can use the Bat+ pad. Be sure to route the wire neatly and remember the "hinge your stack" approach. Alternatively, you can run the wire to any vBat source on your PDB.

Solder the ground wire from the camera and the vTX to the G pad that is right next to the VI pad. Twist the two ground wires together and tin them with solder before soldering both at the same time to the G pad. If you are using vBat to power the camera and vTX, you may prefer to solder the ground wire to any ground source that is nearby to the vBat source. This will keep the wiring neat and logical.



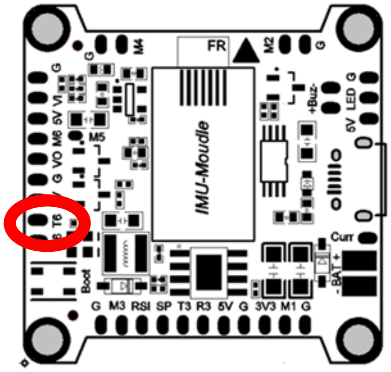
Some video transmitters allow the channel, transmit power, and other such parameters to be configured remotely, through the Betaflight OSD. This means you can change channel and transmit power using your goggles and transmitter sticks, instead of pushing a button or flipping DIP switches on the vTX itself. This is a huge convenience!

Two main video transmitters that support this feature are the ImmersionRC Tramp and the TBS Unify. Holybro's new vTX, the Atlatl, also uses the IRC Tramp telemetry protocol to allow remote configuration.

If you have a vTX that supports SmartAudio or ImpulseRC Telemetry, then:

For SmartAudio vTX including TBS Unify, solder the audio wire from the vTX to the T6 pad on the Kakute F4.

For Tramp Telemetry vTX including the ImmersionRC Tramp and the Holybro Atlatl, solder the T wire from the vTX to the T6 pad on the Kakute F4.

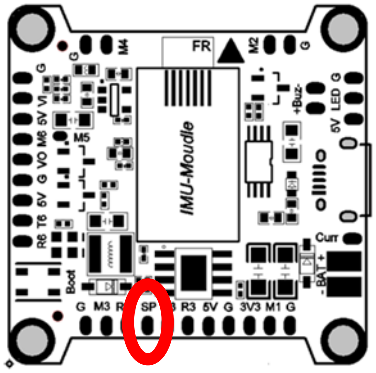


x`

Telemetry allows the Kakute F4 to report values, such as battery voltage, back to your transmitter. The transmitter can be configured to give audible alerts on low battery, and other such functions.

If you intend to use telemetry, solder the telemetry wire from your receiver to the SP pad on the Kakute F4. On a FrSky receiver, the telemetry wire will be labeled as S.Port or SmartPort.

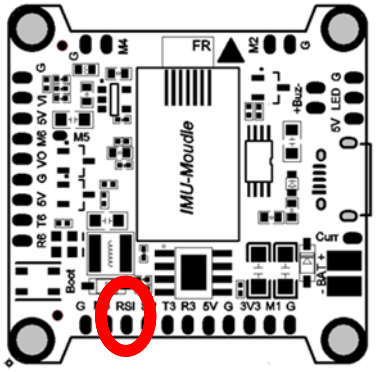
Since the Kakute F4 has a built-in OSD for battery voltage monitoring, many people choose to run without telemetry. However, some people prefer both the OSD and telemetry.



RSSI monitoring allows you to view the signal strength of the control link between your transmitter and your receiver in the OSD. This can give a warning when you are getting close to the edge of your range, as well as showing you if you have damaged equipment, such as an antenna that has been cut by a prop.

If you intend to use the RSSI input, solder the analog RSSI output of your receiver to the RSI pad on the Kakute F4.

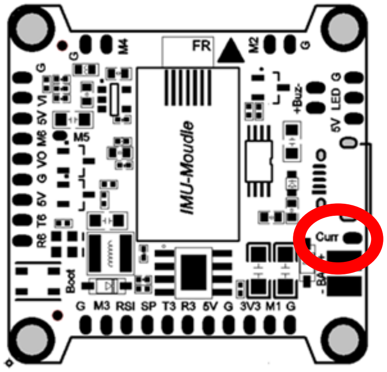
Many receivers do not support analog RSSI output, so you may not be able to use this feature.



Current monitoring allows you to view how many amps the quadcopter is drawing in the OSD. You can also view how much of your battery's capacity (mAh) you have used, kind of like a "fuel gauge" for your battery. To use current monitoring with the Kakute F4, you must have an external current sensor on the copter. Most commonly, this will be built into your PDB or your 4-in-1 ESC—but many PDB's and ESC's don't have a current sensor at all.

If you don't have a current sensor on your copter, you can't view amps and mAh in the OSD. Don't worry too much about this. Pilots have been flying without current sensors for a long time. The Kakute F4 can always show battery voltage in the OSD, and this is enough, although mAh sensing is better.

If you have a current sensor on your copter, you can solder its output wire to the Curr pad on the Kakute F4. This will enable the Kakute F4 to report amps and mAh in the OSD.



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 Kakute F4 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 CP210x 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”.

If you are on Windows, you must install the drivers manually. 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.

All About Betaflight Drivers, Including How to Install Them

https://www.youtube.com/watch?v=m4ygG6Y5zXI

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 3.
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 Kakute F4, plug it into your PC via USB. The ImpulseRC Driver Fixer should complete successfully.

The video linked above shows a process of using Zadig to replace the VCP driver. The ImpluseRC Driver Fixer is an easier way of doing the same thing. So, use the ImpulseRC Driver Fixer and don't mess around with Zadig like the video shows.

Is It Over Yet?

THAT WAS SUPER ANNOYING WASN'T IT. Yes… we know.

The good news is, you do not need to repeat this process again. Sort of. You never need to install the drivers again on this computer, unless you reinstall the operating system for some reason. If you use another computer for the first time, you will need to install those drivers on it. Also, every time you get a new flight controller, you must run the ImpulseRC Driver Fixer one time for that board, before you can flash the board. Unless the board has a CP210x instead of using VCP like the Kakute.

You know what? Let's just move on...

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".) The Betaflight GUI is a Chrome app. To run it, first you need to install Google Chrome on your computer.

https://www.google.com/chrome/browser/desktop/index.html

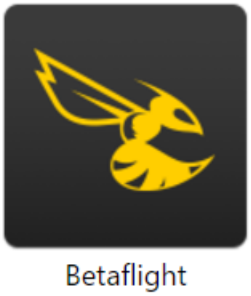
Some people dislike Google and object to installing Google Chrome on their computer. There are a few options for configuring Betaflight without Chrome, but they're mostly intended for quick changes in the field, not full-time use. If you intend to use Betaflight on your multirotor, you're probably going to have to install Chrome and use a Chrome app. Before you think of jumping ship, almost all other flight control firmwares (KISS, Cleanflight) use Chrome apps too.

Next, download the Betaflight Configurator GUI from the Chrome App Store:

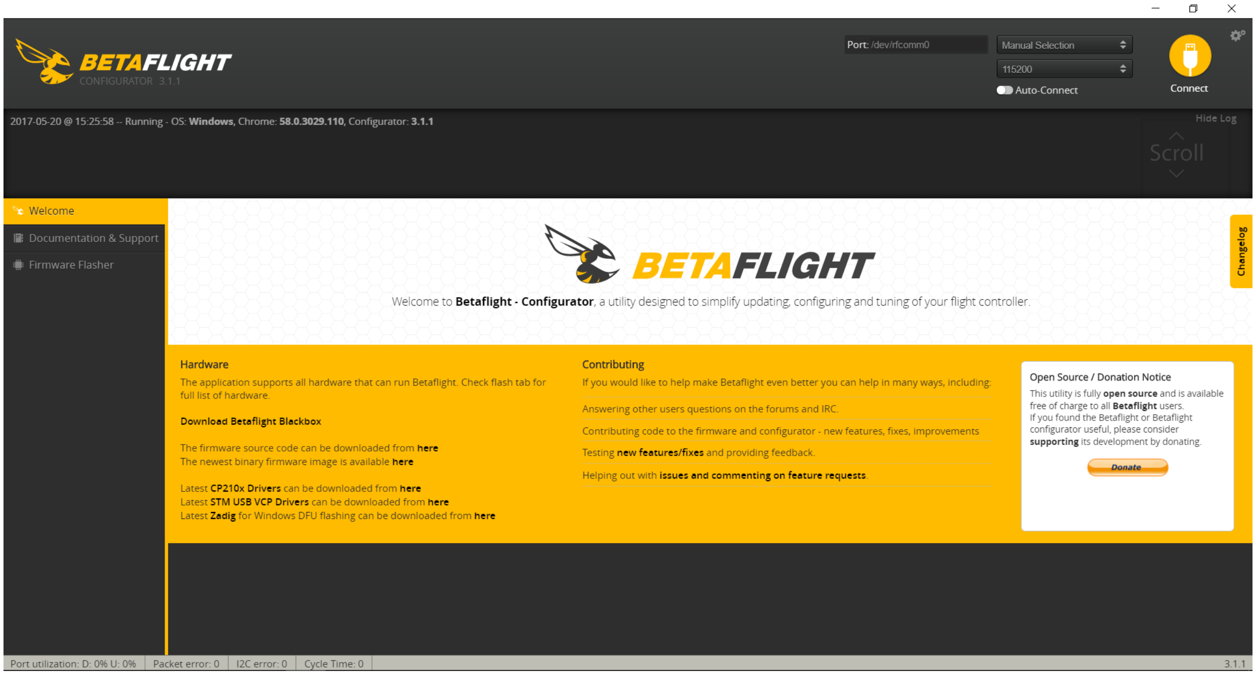
https://chrome.google.com/webstore/detail/betaflight-configurator/kdaghagfopacdngbohiknlhcocjccjao?hl=en

Finally, run the Betaflight Configurator GUI:

1. Enter the URL chrome://apps/ in Chrome's address bar.
2. Click the icon for the Betaflight Configurator



3. The configurator will start.



2017-05-20 @ 15:25:58 -- Running - OS: **Windows**, Chrome: **58.0.3029.110**, Configurator: **3.1.1**

Welcome to **Betaflight - Configurator**, a utility designed to simplify updating, configuring and tuning of your flight controller.

Hardware

The application supports all hardware that can run Betaflight. Check flash tab for full list of hardware.

Download Betaflight Blackbox

The firmware source code can be downloaded from **here**
The newest binary firmware image is available **here**

Latest **CP210x Drivers** can be downloaded from **here**
Latest **STM USB VCP Drivers** can be downloaded from **here**
Latest **Zadig** for Windows DFU flashing can be downloaded from **here**

Contributing

If you would like to help make Betaflight even better you can help in many ways, including:

Answering other users questions on the forums and IRC.

Contributing code to the firmware and configurator - new features, fixes, improvements

Testing **new features/fixes** and providing feedback.

Helping out with **issues and commenting on feature requests**.

Open Source / Donation Notice

This utility is fully **open source** and is available free of charge to all **Betaflight** users.
If you found the Betaflight or Betaflight configurator useful, please consider **supporting** its development by donating.

Port utilization: D: 0% U: 0% | Packet error: 0 | I2C error: 0 | Cycle Time: 0 | 3.1.1

To save time in the future, you can right-click the app icon in Chrome and choose "Create Shortcuts". This will create shortcuts in your start menu and on your desktop, to launch the app directly.

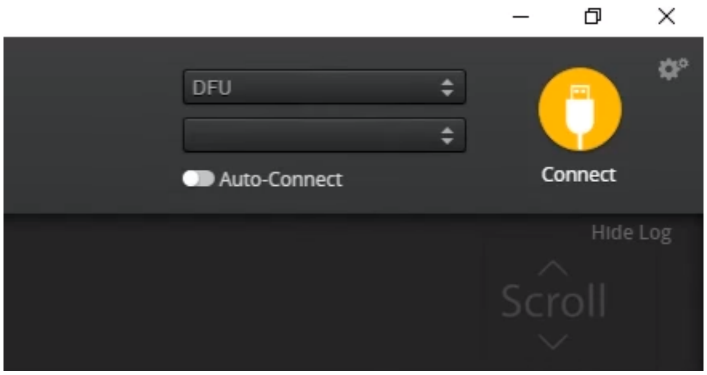
Flashing New Firmware

At this point, if you want to update your firmware, here is how to do it. But if you are annoyed and frustrated at this point and just want to go fly, please, go for it! You don't have to be running the absolute latest firmware to have a good time. Just forget about this nonsense!

Still here?

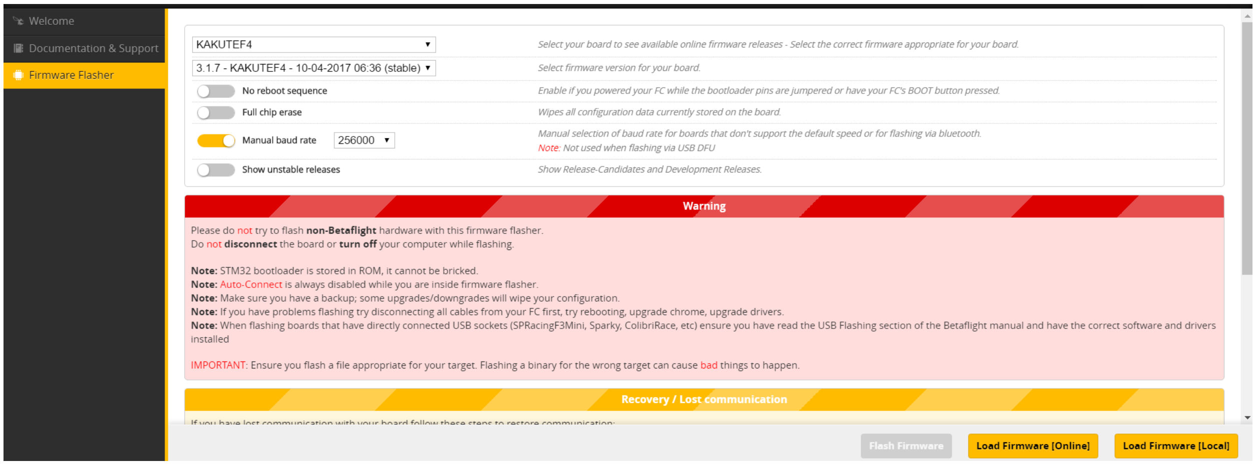
To flash your firmware, you must connect the board in "bootloader mode". Bootloader mode means that the board is ready to accept new programming. To put the board into bootloader mode, hold down the bootloader button while plugging in the USB cable. Leave the button pressed for a moment after plugging in the USB cable to be sure it "takes".

If your Kakute F4 is in bootloader mode, then you will see "DFU" in the pulldown menu in the upper-right of the configurator, as shown here:



If you don't see DFU in the pulldown menu, then either the board didn't detect that you had the bootloader button pressed or your drivers are not installed correctly. If you don't see DFU in the pulldown menu, you cannot flash new firmware to the board. It won't work.

For the advanced users, a simpler way of getting into bootloader mode is to go to the CLI and type "dfu". This will reboot the board into bootloader mode. This is especially useful if the board is installed in a copter where it isn't convenient to press the bootloader button.



KAKUTEF4 — *Select your board to see available online firmware releases - Select the correct firmware appropriate for your board.*

3.1.7 - KAKUTEF4 - 10-04-2017 06:36 (stable) — *Select firmware version for your board.*

No reboot sequence — *Enable if you powered your FC while the bootloader pins are jumpered or have your FC's BOOT button pressed.*

Full chip erase — *Wipes all configuration data currently stored on the board.*

Manual baud rate 256000 — *Manual selection of baud rate for boards that don't support the default speed or for flashing via bluetooth.*
Note: Not used when flashing via USB DFU

Show unstable releases — *Show Release-Candidates and Development Releases.*

Warning

Please do not try to flash **non-Betaflight** hardware with this firmware flasher.
Do not **disconnect** the board or **turn off** your computer while flashing.

Note: STM32 bootloader is stored in ROM, it cannot be bricked.
Note: Auto-Connect is always disabled while you are inside firmware flasher.
Note: Make sure you have a backup; some upgrades/downgrades will wipe your configuration.
Note: If you have problems flashing try disconnecting all cables from your FC first, try rebooting, upgrade chrome, upgrade drivers.
Note: When flashing boards that have directly connected USB sockets (SPRacingF3Mini, Sparky, ColibriRace, etc) ensure you have read the USB Flashing section of the Betaflight manual and have the correct software and drivers installed

IMPORTANT: Ensure you flash a file appropriate for your target. Flashing a binary for the wrong target can cause bad things to happen.

Recovery / Lost communication

If you have lost communication with your board follow these steps to restore communication:

Here are the remaining steps:

1. Go to the "Firmware Flasher" tab.
2. Select "KAKUTEF4" in the "Choose a board" pulldown menu. If you flash any other board type, the Kakute F4 will not function. It won't be damaged, it just won't work until you flash KAKUTEF4 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 orange.

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”. Verification sometimes fails, but this isn’t a problem. If the flash completes, everything is usually fine.
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 KakuteF4 target when you flashed the board. Or it may indicate that your CP210x drivers were not installed correctly.

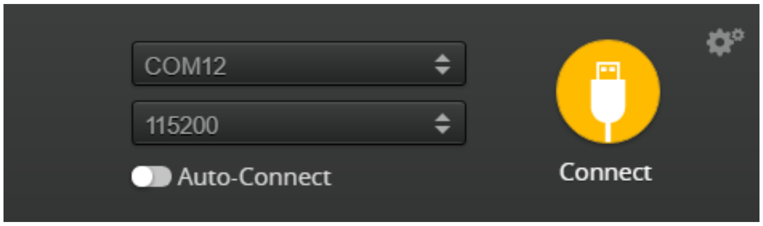
You are ready to configure your board.

Initial Configuration

The full configuration of Betaflight could take hours to document. In this section, we'll describe a few things that are specific to this board. This won't be enough to get you into the air, so we'll also point you to some videos you can watch if you're not perfectly sure what else you need to do. Even people have a few builds under their belt may be skipping some important steps without realizing it!

Connect to The Board

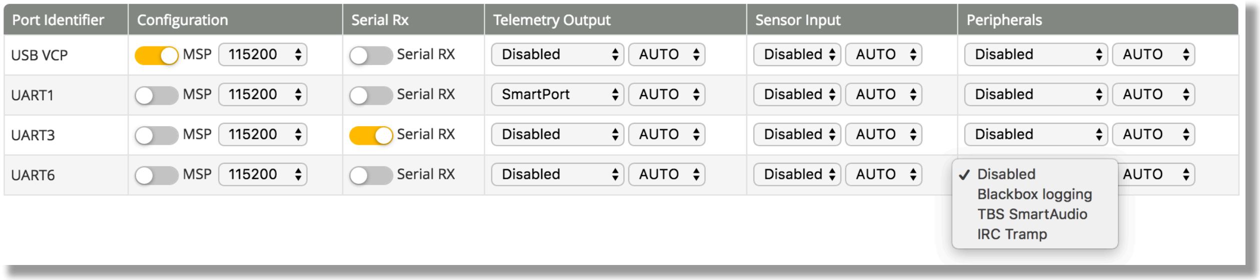
Plug the board in to USB. Start Betaflight Configurator. You should see "COM3" (or some other number) in the upper-right menu. Click "Connect".


COM12
115200
○ Auto-Connect

The GUI will load.

Ports

Click on the "Ports" tab on the right side of the window.



Port Identifier	Configuration	Serial Rx	Telemetry Output	Sensor Input	Peripherals
USB VCP	● MSP 115200	○ Serial RX	Disabled AUTO	Disabled AUTO	Disabled AUTO
UART1	○ MSP 115200	○ Serial RX	SmartPort AUTO	Disabled AUTO	Disabled AUTO
UART3	○ MSP 115200	● Serial RX	Disabled AUTO	Disabled AUTO	Disabled AUTO
UART6	○ MSP 115200	○ Serial RX	Disabled AUTO	Disabled AUTO	✓ Disabled Blackbox logging TBS SmartAudio IRC Tramp AUTO

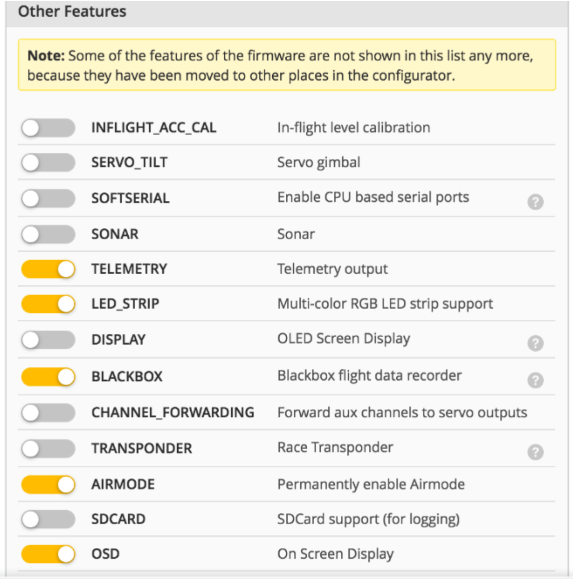
- USB VCP is the port that is used to talk between the Kakute F4 and the Configurator GUI. MSP is the protocol that they use to talk to each other. If you ever turn off MSP on USB VCP, you will not be able to contact the board any more via the Configurator. You will have to re-flash the board and erase your configuration to get back into it. Suffice it to say, this is bad, and you shouldn't do it. **Don't ever turn off MSP on the USB VCP line in the Ports tab.**
- If you are using FrSky SmartPort telemetry: on the UART1 line, in the Telemetry Output column, choose SmartPort from the pulldown menu.
- On the UART3 line, enable "Serial RX".

- If you are using SmartAudio or ImmersionRC Tramp Telemetry to control your video transmitter: on the UART6 line, in the Peripherals column, choose either TBS SmartAudio or IRC Tramp, depending on which type of transmitter you are using.

Configuration

Click on the “Configuration” tab on the right side of the window. Scroll down to the “Other Features” section.

- If you are using SmartPort telemetry, enable TELEMETRY.
- If you are using a programmable LED strip, enable LED_STRIP.
- The Kakute F4 has a built-in Dataflash chip for blackbox logging. If you intend to use this feature, enable BLACKBOX.
- Air Mode increases authority when the throttle is all the way down. We recommend leaving this option on all the time.
- The Kakute F4 has built-in Betaflight OSD. The OSD option should always be enabled, unless you want to turn off the OSD for some reason.



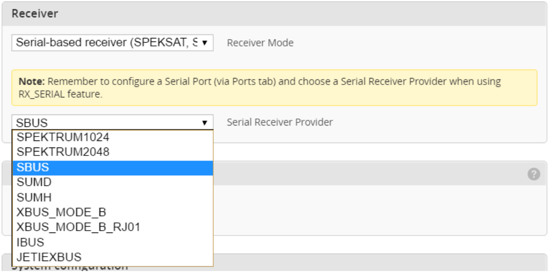
Other Features

Note: Some of the features of the firmware are not shown in this list any more, because they have been moved to other places in the configurator.

○	INFLIGHT_ACC_CAL	In-flight level calibration
○	SERVO_TILT	Servo gimbal
○	SOFTSERIAL	Enable CPU based serial ports
○	SONAR	Sonar
●	TELEMETRY	Telemetry output
●	LED_STRIP	Multi-color RGB LED strip support
○	DISPLAY	OLED Screen Display
●	BLACKBOX	Blackbox flight data recorder
○	CHANNEL_FORWARDING	Forward aux channels to servo outputs
○	TRANSPONDER	Race Transponder
●	AIRMODE	Permanently enable Airmode
○	SDCARD	SDCard support (for logging)
●	OSD	On Screen Display

Next, go to the Receive section of the “Configuration” tab. Since the Kakute F4 only support serial-type receivers, configuration of this section is simple.

- Set the Receiver Mode to “Serial-based receiver”.
- Se the Serial Receiver Provider to the type that matches what kind of receiver you have. FrSky and other SBUS receiver should use SBUS. Spektrum Satellite receivers should use SPEKTRUM2048 or SPEKTRUM1024 depending on whether they are DSM2 or DSMX. FlySky receivers use iBus. These are the most common serial receiver types that are likely to be encountered.



Receiver

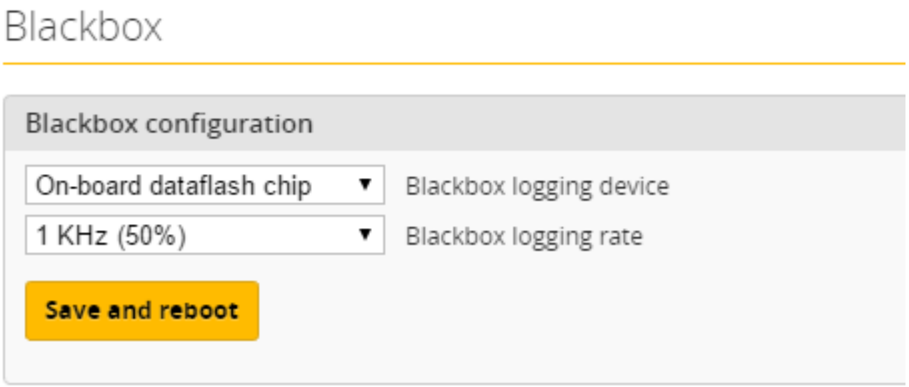
Receiver Mode: Serial-based receiver (SPEKSAT, S

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

Serial Receiver Provider: SBUS

Blackbox

If you have enabled the Blackbox feature, go to the Blackbox tab on the left-hand side of the window. In the Blackbox tab, at the top, set the logging device to “On-board dataflash chip” and the Blackbox logging rate to 1 kHz.



Blackbox

Blackbox configuration

On-board dataflash chip ▾ Blackbox logging device

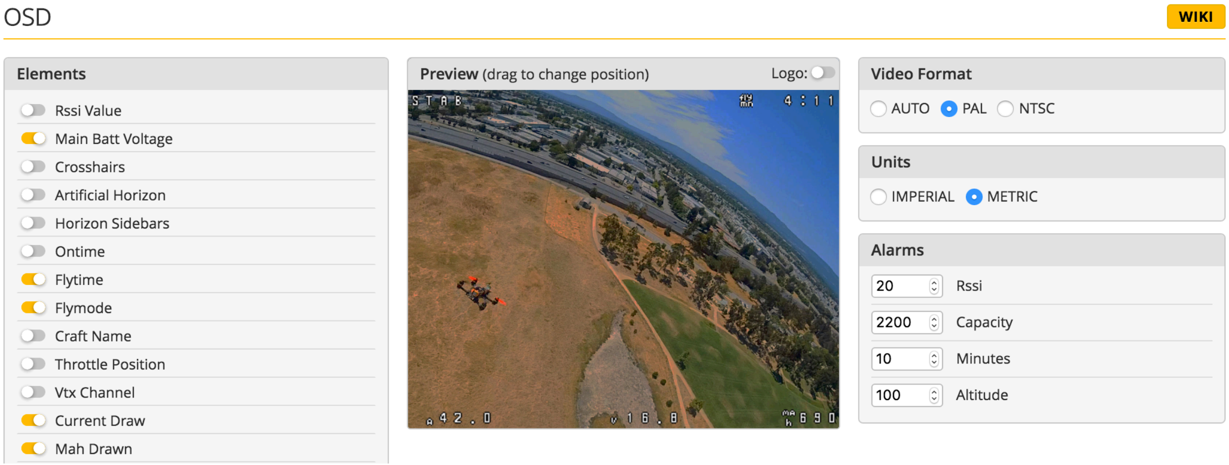
1 KHz (50%) ▾ Blackbox logging rate

OSD

In the OSD tab, you can choose which values you want to see on screen while you are flying. Enable and disable individual elements using the Elements toggles on the left.

The Video Format section lets you choose whether your camera is NTSC or PAL. Betaflight defaults this value to Auto, but Auto sometimes picks wrong. If this happens, you might not see any OSD text, or the bottom of the OSD text might be off the bottom of the screen. In some cases, you won't see any video at all—just the OSD. Because of this, we recommend manually setting the Video Format to NTSC or PAL depending on which type of FPV camera you have.

You can rearrange the individual OSD elements on screen by dragging them with your mouse.



Elements

- ○ Rssi Value
- ● Main Batt Voltage
- ○ Crosshairs
- ○ Artificial Horizon
- ○ Horizon Sidebars
- ○ Ontime
- ● Flytime
- ● Flymode
- ○ Craft Name
- ○ Throttle Position
- ○ Vtx Channel
- ● Current Draw
- ● Mah Drawn

Preview (drag to change position) Logo: ○

Video Format

○ AUTO ● PAL ○ NTSC

Units

○ IMPERIAL ● METRIC

Alarms

20	Rssi
2200	Capacity
10	Minutes
100	Altitude

Some individual OSD Elements of note are:

Main Batt Voltage: This is probably the single most important element to use. Your battery must never go below 3.3 volts per cell under any circumstances, or it will suffer damage. For a 3S battery, this would be 3 times 3.3 = 9.9 volts. For a 4S battery, it would be 13.2 volts. Any time the battery goes below this level, it is likely being damaged, at least a little. More importantly, you will have very little power and may crash if you are doing aggressive maneuvers.

Although 3.3 volts is the recommended absolute minimum, you will get the best life out of your batteries if you keep the voltage above about 3.5 volts per cell—10.5 volts for 3S and 14.0 volts for 4S. If you are flying and you notice the battery going below this level, it might be time to land, or at least go easier on the throttle. If your batteries consistently drop to this level when you try to fly, it might be time to get new batteries.

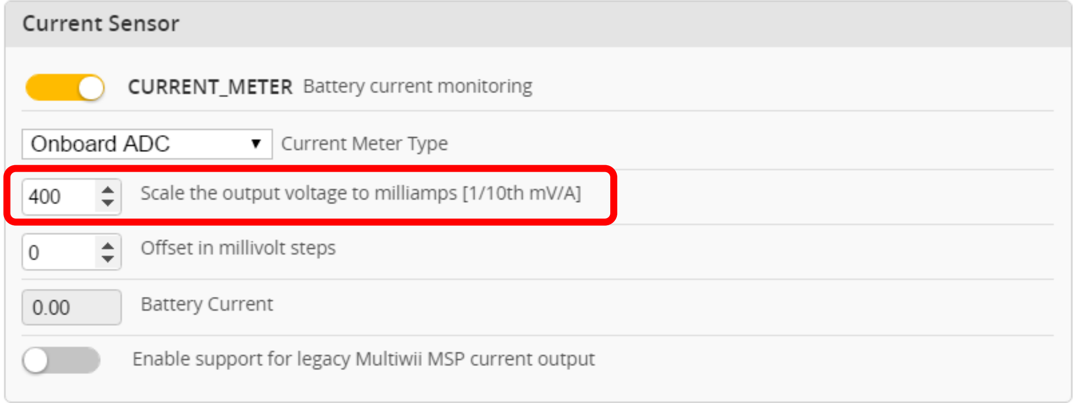
When you are using the battery, its voltage will drop. This is referred to as "sag". After you land, the battery will recover a small amount and its voltage will rise. Our recommendation is that, the battery should be at no less than 3.75 volts per cell—11.25 volts for a 3S and 15.0 volts for a 4S—after it has been allowed to rest at the end of a flight. If your batteries are consistently resting at below this level at the end of a day of flying, then you might be shortening their lifespan at least a little.

mAh Drawn: Although voltage is what ultimately determines whether a battery is being damaged, mAh may be a better way of deciding when to land. Because voltage sags when you raise the throttle and recovers when you lower the throttle, it can be hard to tell exactly how used-up the battery is. mAh is like a "gas gauge" for your battery. It shows how much capacity you have consumed, independent of other factors.

A good practice is to draw from a battery no more than about 80% of its rated mAh. So, a 1300 mAh battery would be able to deliver about 1000 to 1100 mAh. With a healthy battery and an honest battery rating, this will result in the battery resting at about 15.0 volts after flying. If you draw 80% of a battery's rated capacity and it is resting at higher than 15.0 volts, the manufacturer might have under-rated its capacity. If the battery is resting at lower than 15.0 volts, the manufacturer might have over-rated its capacity. Also, remember that batteries lose capacity as they age. A battery that consistently fails to deliver 80% of its rated capacity without sagging below 14.0 volts might need retirement.

To measure the mAh drawn, the quadcopter must have a current sensor. Many quadcopters won't have a current sensor, but some PDB's and 4-in-1 ESC's have a current sensor built in. If you have a current sensor on your quad, and if you wired it up to the Curr pad on the Kakute F4, then you will be able to see mAh drawn in the OSD.

The Kakute F4 must be calibrated to match the readings put out by your current sensor. We recommend that you record the mAh Drawn at the end of your first few flights, then compare it to the mAh that your battery charger puts back into the batteries. Then adjust the Current Scale in the Configuration tab to compensate for any difference.



Current Sensor

CURRENT_METER Battery current monitoring

Onboard ADC — Current Meter Type

400 — Scale the output voltage to milliamps [1/10th mV/A]

0 — Offset in millivolt steps

0.00 — Battery Current

Enable support for legacy Multiwii MSP current output

Here is an example:

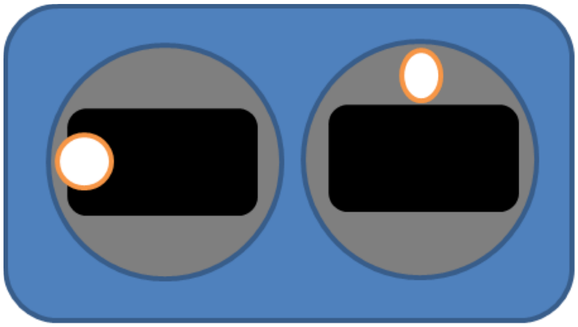
- OSD shows 1100 mAh Drawn at the end of the flight.
- Charger shows 1000 mAh put back in to the battery.
- 1100 / 1000 = 1.10. The OSD is reading 10% high.
- Current Scale works backwards from how you might think. To make the OSD read lower, you make Current Scale larger. Since the OSD is reading 10% high, we need to INCREASE Current Scale by 10%. This will cause the OSD to read 10% lower.
- If the Current Scale was at 400, we will multiply it by 1.10 to add 10%, meaning the new value should be 440.

The mAh put back in by the battery will seldom perfectly match the mAh reported by the OSD, but by taking several measurements and averaging the results, you can usually get it reasonably close.

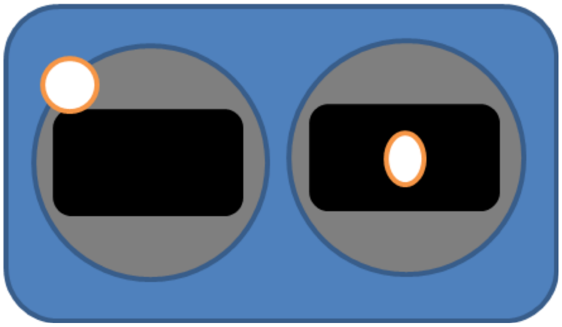
Using The OSD

The Kakute F4 includes Betaflight OSD, which displays information like battery voltage and mAh consumed while you fly. In addition, the Betaflight OSD can be used to configure the quadcopter, making in-field adjustments and tuning more convenient.

Mode 2

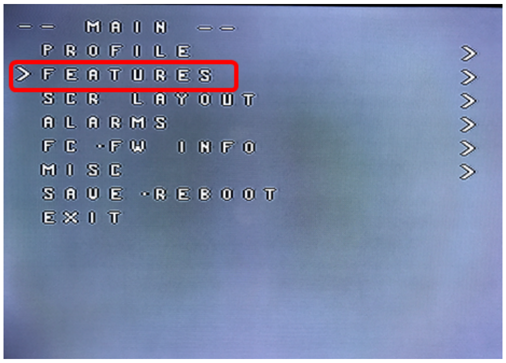


Mode 1

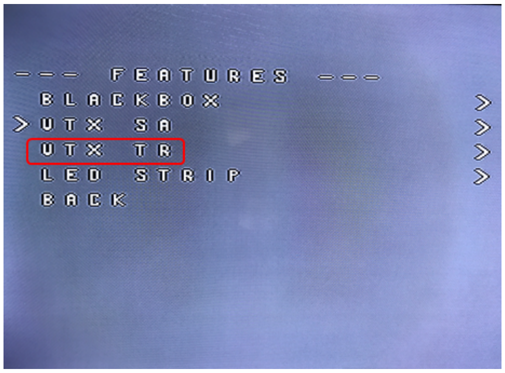


The graphics above show the stick command to bring up the OSD menu. The stick command is: throttle centered, yaw left, pitch forward. The exact stick command therefore depends on which mode your transmitter sticks are in.

In the OSD menu, use pitch up/down to move the cursor between menu items. When a menu option has a > symbol to the right of it, this indicates that it contains a sub-menu. Roll-right will enter the sub-menu. For example, in the screen to the right, moving the cursor to "Features" and then moving the roll stick to the right will enter the "Features" sub-menu.



If you are using a video transmitter that supports remote configuration, enter the "Features" menu to configure the vTX. From there, enter either "VTX SA" if you are using SmartAudio (TBS Unify) or "VTX TR" if you are using IRC Tramp Telemetry.



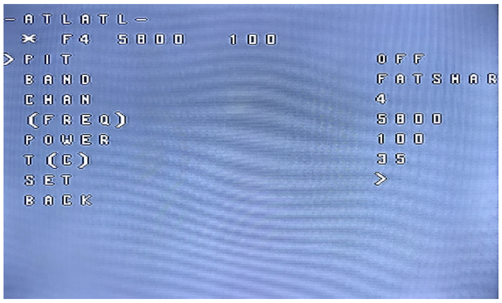
To adjust PIDs, rates, and other tuning-related parameters, enter the "Profile" sub-menu.

In the "Scr Layout" sub-menu, you can move the OSD elements (like battery voltage, mAh, and so forth) around on the screen.

The "Alarms" sub-menu lets you control when the OSD will try to alert you that battery voltage is too low or mAh consumed is too high.

When a parameter can be modified, the parameter's current value will be shown on the right-hand side of the screen. In this case, roll left/right will adjust the parameter up and down.

The screen to the right shows the current vTX settings. From here, you can change the frequency band, channel, and power level of the video transmitter. After making the changes, move the cursor to "Set" and press roll-right to confirm the settings.



```
-ATLATL-
 ✕  F4  5800    100
>PIT                    OFF
 BAND                   FATSHAR
 CHAN                   4
 (FREQ)                 5800
 POWER                  100
 T(C)                   35
 SET                    >
 BACK
```

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.

1. Connect your Kakute to your PC by plugging in USB.
2. Start the Betaflight GUI app.
3. Go to the CLI tab.
4. In the text box at the bottom of the CLI, type “diff” and hit enter. This will cause the flight controller to display all configuration options that you have changed from the default values.
5. Scroll back to the top of the window to find the line where you typed “diff”.
6. Click your mouse and drag/scroll down to select all of the text below where you typed “diff”. This is your configuration data.
7. Right-click and choose “Copy” to copy the highlighted text to the clipboard.
8. Open any text editor, such as Windows Notepad or MacOS TextEdit.
9. Right-click in the text editor window and choose “Paste” to paste the highlighted text into the editor.
10. Save the document somewhere you won’t lose it.

To restore your configuration, do this:

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

How To Save and Restore Your Configuration

https://www.youtube.com/watch?v=HsxTqp76Brs

Additional Reference

Here are some links to additional videos to help you build your quadcopter successfully.

Holybro Shuriken X1 Initial Configuration

https://www.youtube.com/watch?v=lNmEeCMryu4

This video is not based on the latest release of Betaflight, but the basic steps of setting up the quadcopter, such as calibrating the ESCs and configuring failsafe, are the same.

How to Calibrate Your ESCs

https://www.youtube.com/watch?v=o3Mg-9M0l24

If you are using an analog protocol like Oneshot or Multishot, calibrating your ESCs is mandatory. If you are fortunate enough to be using Dshot, then you can skip this step.

Failsafe

https://www.youtube.com/watch?v=dikr9oDzQqc

Failsafe is what happens when you fly too far away or your receiver gets disconnected from your flight controller. If you don’t configure failsafe, the quad could fly away, or it could crash into something… or someone.

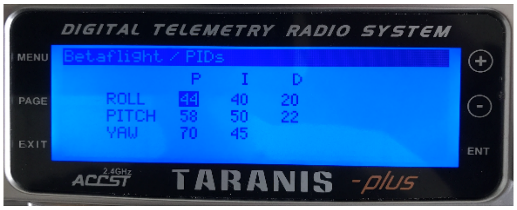
DON’T FLY WITHOUT VERIFYING THAT FAILSAFE WORKS CORRECTLY.

To verify failsafe, remove your props, then plug in your battery and arm your copter. Then turn off your transmitter. Your motors should stop within 1 to 3 seconds at most.

If your motors don’t stop when you turn off your transmitter, DO NOT FLY YOUR MULTIROTOR UNDER ANY CIRCUMSTANCES.

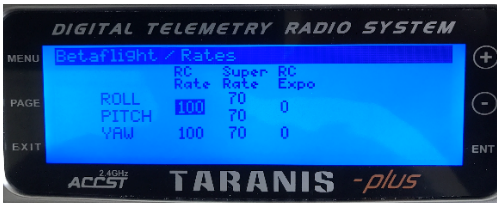
Adjust PIDs / Rates / vTX from Taranis

If you have a FrSky Taranis radio and if you are using SmartPort telemetry, you can use your Taranis to change your PIDs and rates.



Betaflight / PIDs

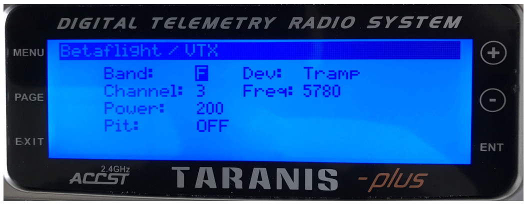
	P	I	D
ROLL	44	40	20
PITCH	58	50	22
YAW	70	45	



Betaflight / Rates

	RC Rate	Super Rate	RC Expo
ROLL	100	70	0
PITCH		70	
YAW	100	70	0

This is done by installing a piece of programming code called a Lua script on your Taranis. If you are also using SmartAudio, you can use a Lua script to change your vTX settings. This is the same as if you were using the Betaflight OSD, but it works without you having to put your goggles on.



Betaflight / VTX

Band: F
Channel: 3
Power: 200
Pit: OFF
Dev: Tramp
Freq: 5780

Detailed instructions for how to do this are beyond the scope of this document. Oscar Liang has an article with instructions, here:

How to Adjust PIDs, Rates, and vTX Settings via Lua Script

https://oscarliang.com/betaflight-change-pid-vtx-settings-taranis/