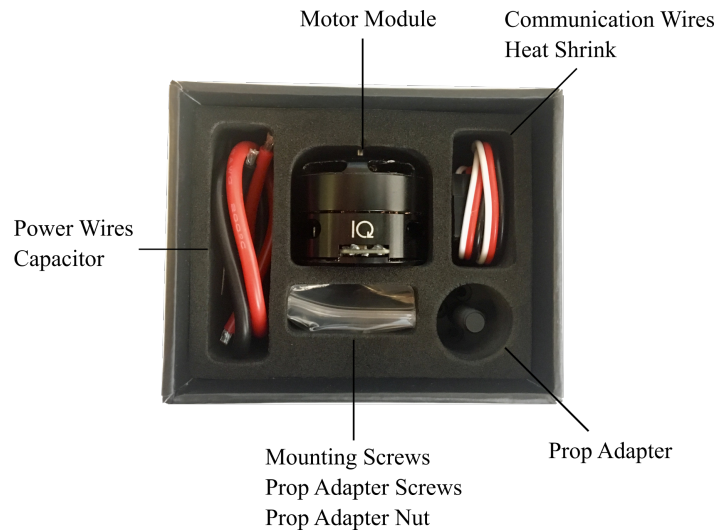


# Instruction Manual

Warning: you must initialize your motor module through the IQ Control Center before use

## 1 Box Contents

- Motor module
- Power wires
- Communication wires
- Capacitor
- Propeller adapter
- Propeller adapter nut
- Propeller adapter screws
- Mounting screws
- Heat shrink



## 2 Module Assembly

- Solder power wires to the top part of the PCB
  - Black wire to -
  - Red wire to +
- Solder communication wires to bottom part of the PCB
  - Black wire to -
  - Red wire to PCBs TX/Tel
  - White wire to PCBs RX/PWM
- Solder external capacitor on top of power wires\*
  - Cut capacitor leads down to desired size (rec: 8mm)
  - Side with white marking to - (over black power wire)
  - Side without marking to + (over red power wire)
- Put heat shrink over the wires and capacitor and apply heat



*\*optional, but suggested when applying more than 8V (2S battery)*

### 3 Module Settings

---

- Download the IQ Control Center (user interface)
  - Go to [www.iq-control.com/control-center](http://www.iq-control.com/control-center)
  - Click the link provided on the Control Center page. This will direct you to our GitHub page.
  - Download the Windows .exe file or Mac .dmg file
  - Proceed through installation
- Connect your module (see sections 7, 8, or 9 for specific details)
  - Make sure the module is disconnected from any application and does not have propellers attached
  - Plug FC (sec. 7), arduino (sec. 8), or USB to UART (sec. 9) chip into computer
  - Plug/solder communication wires to the above device
  - Plug/solder power wires to a power source (must be between 5V and 25V)
  - In the top right of the interface, select the correct com port. Note: you may need to install appropriate drivers.
  - Click the Connect button. If successful, the interface will say “Motor Connected Successfully” in the top message bar. If unsuccessful, it will give you an error message.
- Set module application values
  - At the top of the General tab, you can upload specific settings for your application
  - If your application is in the list, select it and click “Set”
    - \* There are many options for the speed modules, including pre-made settings for velocity mode, voltage mode, and PWM mode
    - \* The only pre-made setting for the position modules at the moment is the factory default
  - If your application is not in the list or you prefer to change settings yourself, you’ll have to change settings manually
- Navigate to the General, Tuning and Advanced tabs to change specific module settings
  - All users should visit the General tab before using their module
  - Change module settings to desired value or mode, if necessary
  - Click the Save button next to a changed setting to update it on the module
    - \* Any unsaved, changed setting will have a \* appear next to the parameter name
    - \* To revert to the value currently on the motor, click the “Check Saved Value” button
  - **WARNING:** exercise extreme caution when changing values in the Advanced tab. A bad setting may result in a motor being permanently damaged or cause erratic behavior

## 4 Drone Setup

---

- Prior to connecting the modules to your drone, open the IQ Control Center to initialize (see section 3)
  - Set appropriate module settings using the pre-made settings at the top of the IQ Control Center General tab (see section 11) or set your own
  - Set module direction (this includes both CW vs. CCW and 2D vs. 3D mode)
  - Set FC mode (2D vs. 3D). If FC is using dshot and in 3D mode, set this value to 3D. Otherwise set this to 2D.
  - If necessary, set motor PID gains in the Tuning tab. In most cases, the default values are satisfactory.
- Configure your flight controller to be compatible with IQ modules
  - Open your flight controller's user interface
  - Enable Air Mode
    - \* This prevents the motor from shutting off at minimum throttle
    - \* **WARNING:** you may see integrator wind up when the vehicle is on the ground depending on your FC and settings
  - Lower motor idle throttle (optional)
    - \* IQ motors do not require an idle throttle value
    - \* If you set this to 0, the motors will not spin when the FC is armed
  - Change other FC settings as you would normally
- Fasten the modules to your drone's frame with mounting screws provided. **WARNING:** using mounting screws that are too long will damage the motor. See the datasheet for details.
- Solder/plug in the power wires to the power distribution board
- Solder/plug in the communication wires to the flight controller
- When using an analog protocol (1-2ms standard, OneShot125, OneShot42, MultiShot) you will need to calibrate the ESCs. See your FC's instructions for the procedure.
- Arming Procedure
  - If this is your first time setting up the vehicle, do this without props to test motor settings
  - Arm your FC normally
  - The motors will only arm once throttle is in the appropriate place
    - \* 2D mode - motors arm at minimum throttle (throttle stick all the way down)
    - \* 3D mode - motors arm at mid throttle (throttle stick in the middle)
  - The motors will play the arming tune and spin slowly
  - Once you've tested motor settings, disarm your FC
- Add the propellers (see section 5)

- Repeat the arming process and fly!
- WARNING: if your motors are hot after a short hover test, decrease PD gains and lower the filter frequencies on your FC.

## 5 Propeller Setup

---

- Place the prop adapter over the steel extrusion and align the screw holes
- Screw in the adapter tightly with the 4 M2 screws (we recommend using threadlocker)
- Screw on the propeller
- Screw on the nut tightly

## 6 Position Application Setup

---

- Before connecting the module to your application, download the firmware that is appropriate for your application
  - The servo position firmware accepts standard PWM, OneShot, MultiShot, and DShot inputs as well as IQ's UART protocol and can operate in position, velocity, voltage, or PWM mode
  - The step/direction firmware mimics a stepper motor with one line as step (TX/Telem), and the other line as the direction (RX/PWM)
- Open the IQ Control Center to initialize (see section 3)
  - If your application involves a rotation to linear conversion, change the conversion factor to meters per radian in the General tab
  - In the Tuning tab, Set initial position, PID gains, etc.
  - In the Advanced tab, Set safety features, such as timeout and max angular speed
- Connect your module's power wires to a power source (between 5V and 25V)
- Connect your communication wires to your arduino, computer, etc.
- Mount your module to the application
- If communicating over UART, use the IQ libraries (available at [www.iq-control.com/libraries](http://www.iq-control.com/libraries)) to send commands to your module\*

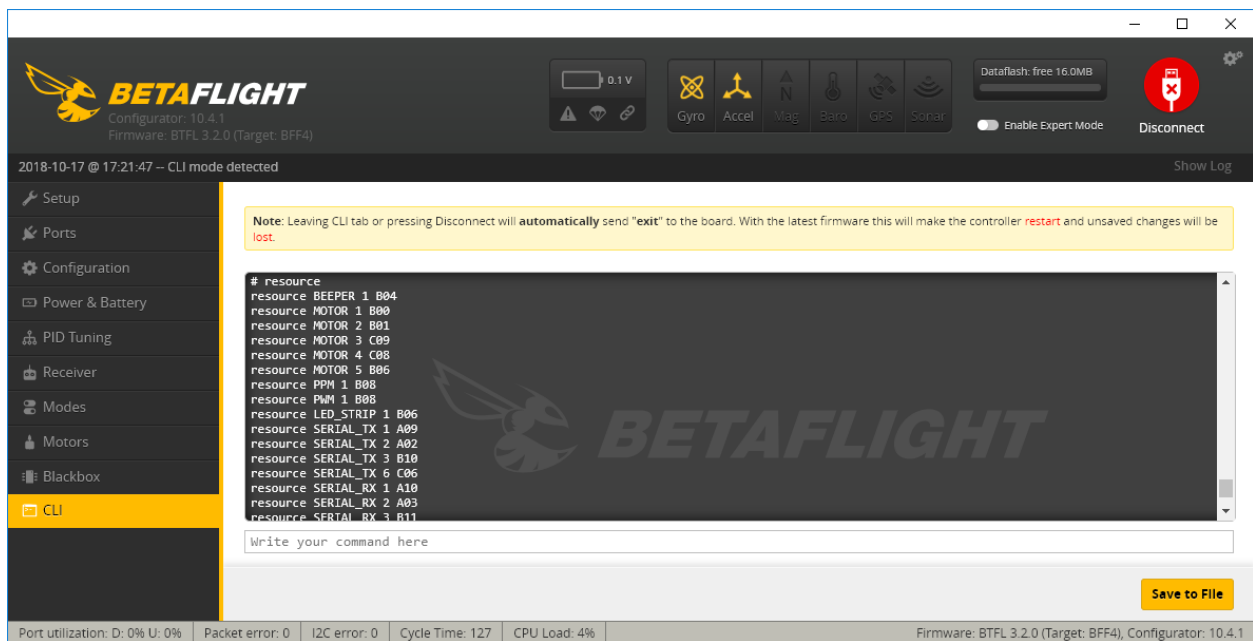
*\*Refer to the IQ Communication Manual for more detailed instructions on the commands*

## 7 Connect to IQ GUI: BetaFlight

### Set up FC as Passthrough Device

- Download and open the Betaflight Configurator
- Connect your Flight Controller (FC)
  - Plug your FC into your computer through USB
  - Before you Connect, select your FC's communication port (check device manager)
  - Click Connect
- Making a Backup
  - Click on the CLI tab in the menu on the left
  - In the textbox, type:
 

```
diff
```
  - Click Save to File in the bottom right corner
  - Name it something simple like *"bf\_backup.txt"* and press save



- Setting up Software Serial (use above image for example references)
  - To find which FC microcontroller pins are connected to the ESC lines, type this in the console:
 

```
resource
```
  - Take note of the motor's PWM/DShot line (*Example: MOTOR 1's resource is B00*)
  - Take note of the ESC telemetry line (*Example: Serial RX 1's resource is A10*)

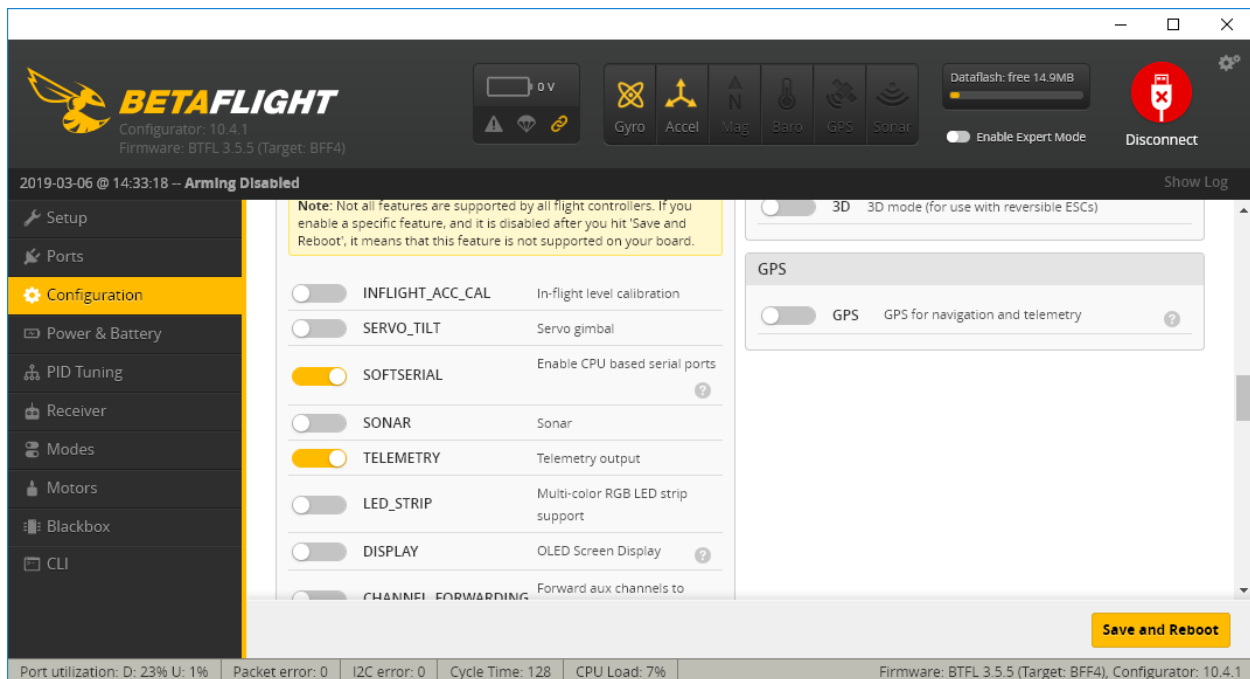
*Tip: look in the FC's documentation or on the FC's silkscreen to figure out motor location and which pin is connected to ESC telemetry*

- Map motor resource to software serial resource
    - \* First, disconnect the the current resources from the pins by typing the following in the console:
 

```
resource motor 1 none
resource Serial_RX 1 none
```
    - \* Now to connect the software serial resources, type:
 

```
resource Serial_TX 11 B00
resource Serial_RX 11 A10
```
- Tip: the red-colored text above indicates that particular value will change for each motor. So for motor 2, you would type: **resource motor 2 none** and **resource Serial\_TX 11 B01**. The other two commands remain the same.*
- Save your settings by typing:
 

```
save
```
  - Your flight controller will restart. If the GUI doesn't disconnect automatically, press "Disconnect"



- Enable Software Serial (use above image for reference)
  - Click the "Connect" button again
  - Click Configuration tab in the menu on the left
  - Scroll down to Other Features and enable SOFTSERIAL
  - Click Save and Reboot in the bottom right corner
  - Your flight controller will restart. If the GUI doesn't disconnect automatically, press "Disconnect"
- Make the FC a Passthrough Device
  - Click the "Connect" button again

- Select the CLI tab in the menu on the left

*Tip: for future use you can save this configuration to easily talk to your motors. Use the **diff** command now by mimicking the steps in the Making a backup section so you can make this process easier the next time. Use the steps in Restore your FC to setup the softserial. Make sure you use a different filename for your backup file!*

- In the textbox, type:

```
serialpassthrough 30 115200
```

- The console should say something like this:

```
serialpassthrough 30 115200
```

```
Port 30 opened, baud = 115200.
```

```
Forwarding, power cycle to exit.
```

- Do not power cycle!
- Click the Disconnect button and move to the next section

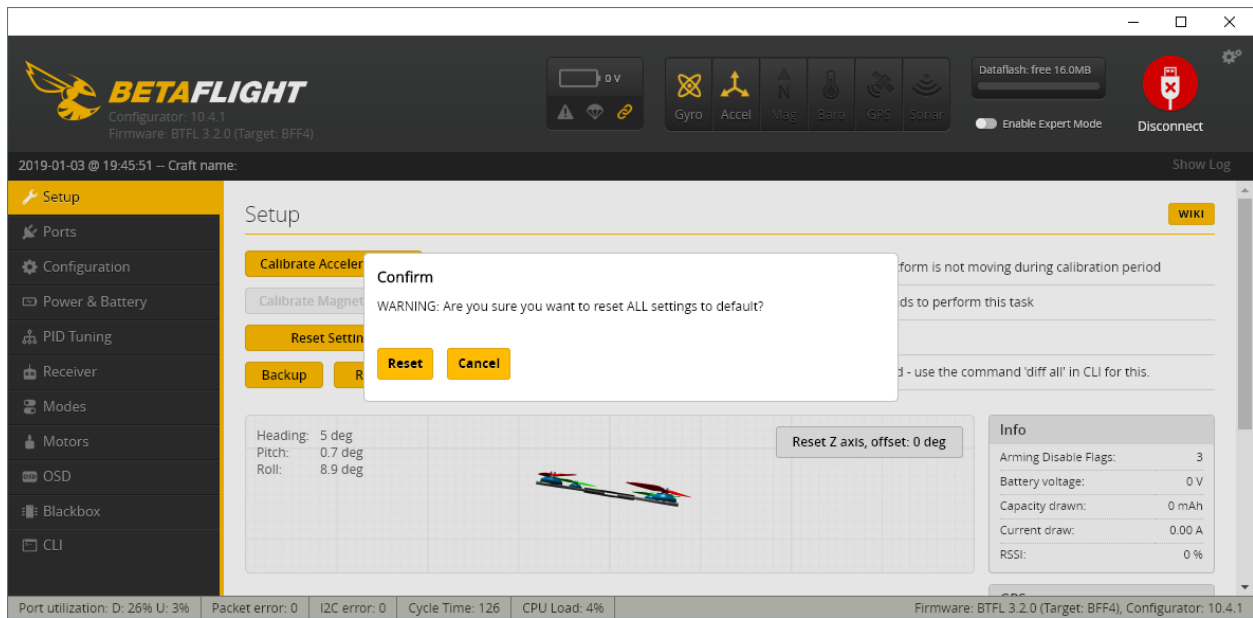
### Change module settings in IQ GUI

- Download and open the IQ GUI (see section 3)
- Ensure the communication wires are connected to the FC in the proper location
- Power the motor (anything between 5V and 25V)
- Select the same com port used to connect to your FC in the top right
- Click the "Connect" button
- Now you can change settings (see sections 3 and 4) and flash new firmware (see section 12)

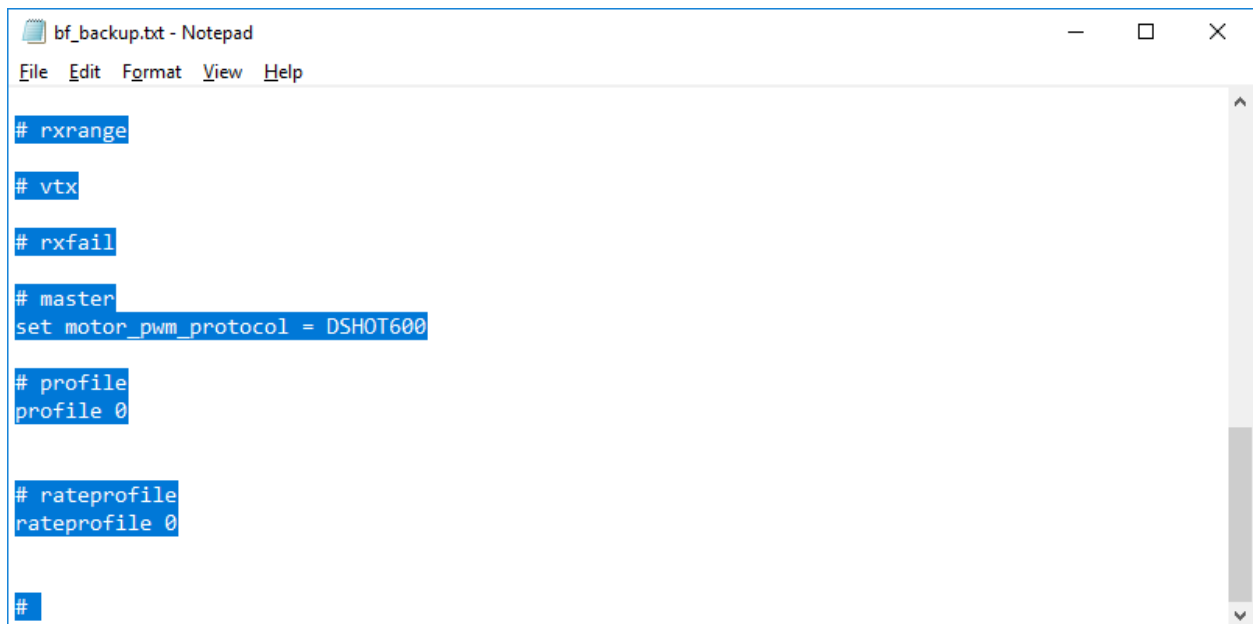
*Tip: the IQ GUI may disconnect or not read a setting from time to time on some older flight controllers because softserial requires a lot of CPU. If this happens, just reconnect the motor and check to make sure you saved all the settings you changed.*

### Restore your FC

- Remove your FC from serialpassthrough mode by unplugging all power, then reconnecting to USB (power cycle)
- Switch back to the BetaFlight GUI and connect the FC
- In the Setup tab, click Reset Settings (don't worry, we will put your custom settings back on the FC in the next step)
- In the popup confirmation window, click Reset (see image below for reference)



- Open your backup file in any text editor
- Select all (ctrl-a in Windows) and copy the text (ctrl-c in Windows)



- Switch back to the BetaFlight GUI
- Click on the CLI tab
- In the textbox, paste the copied settings (ctrl-v in Windows), then press enter
- When all of the commands have been sent, type and enter:  
 save



- Your flight controller will restart. If the GUI doesn't disconnect automatically, press Disconnect

You're done! Get flying!

## 8 Connect to IQ GUI: Arduino

---

There are two styles of Arduino (or similar) boards regarding their USB capabilities: the first are the simple boards that have a USB to UART converter (UNO, etc.) and the second are the boards that have a direct connection to USB (Leonardo, etc.) or multiple serial ports. You must first determine the type of USB connection of your Arduino, then use the appropriate instructions below.

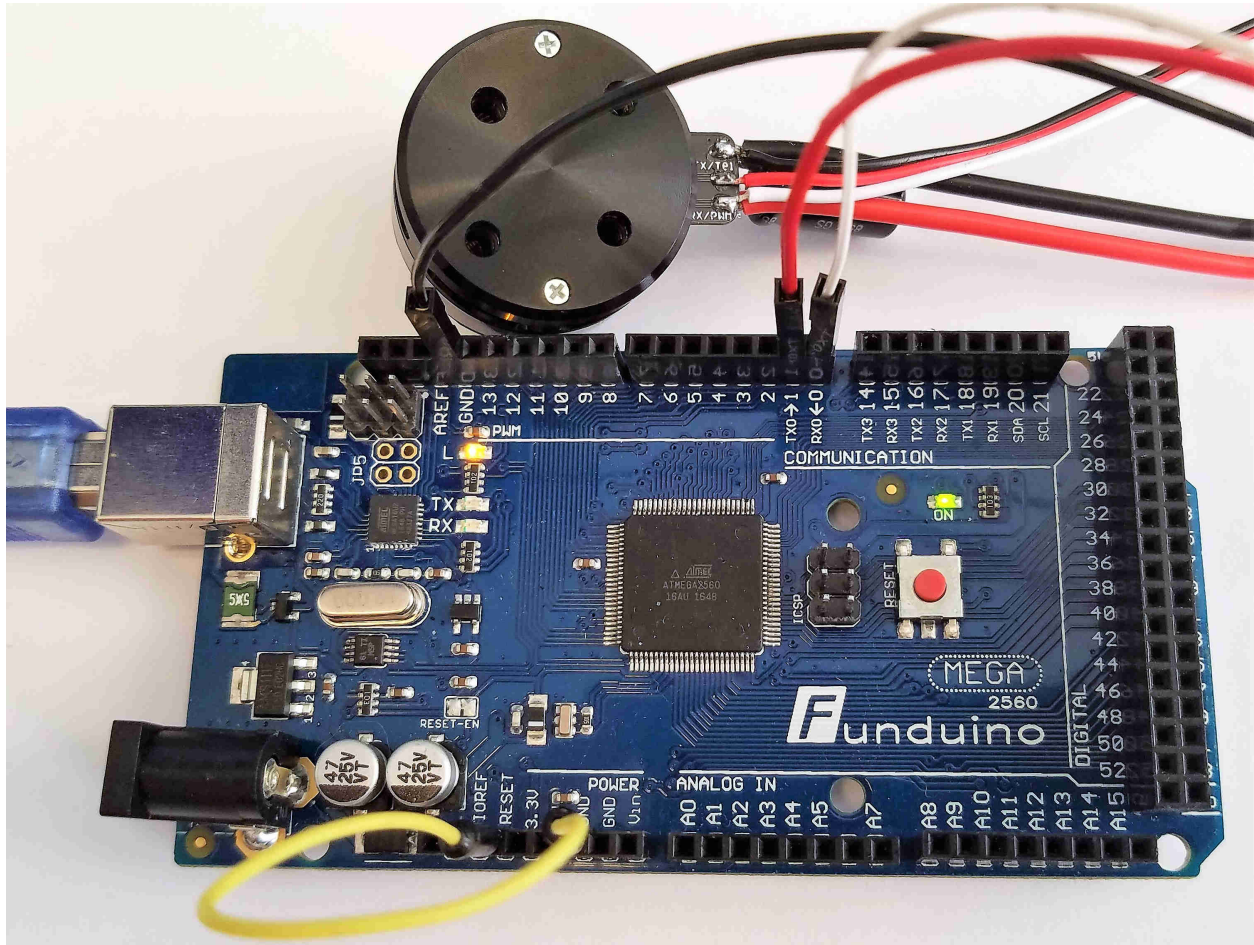
### Devices with USB to UART device

Use these instructions for any device that has a USB to UART chip on the board, such as the Arduino Uno, Nano, Mini, and Mega.

- Disconnect the Arduino's microcontroller pins from the Serial lines
  - Method 1: Connect the Reset pin to GND on the Arduino
  - Method 2: Physically remove the microcontroller from the Arduino board
  - Method 3: Load the following sketch onto the Arduino

```
void setup(){
    pinMode(0,INPUT); // Assumes TX/RX is on pin 0
    pinMode(1,INPUT); // Assumes RX/TX is on pin 1
}
void loop(){
}
```

- Connect the Arduino to the USB port of your computer
- Connect the GND of your USB to UART device to the GND (black) wire on the motor
- Connect the RX of your Arduino to the RX (white) wire on the motor
- Connect the TX of your Arduino to the TX (red) wire on the motor



Motor connected to Arduino Mega using Method 1

**Devices with direct USB connection or multiple serials**

Use these instructions for any device uses the USB CDC which is built into the microcontroller, such as the Arduino Leonardo, Micro, Yn, MKR, Zero, Due, 101. You can also use this method for devices with multiple Serial ports like the Mega.

- Connect the Arduino to the USB port of your computer
- Load the below sketch on your Arduino (also available in the IQ Arduino examples)

```
// Use this #define to indicate which UART/TTL port to use
#define UART_SERIAL Serial1

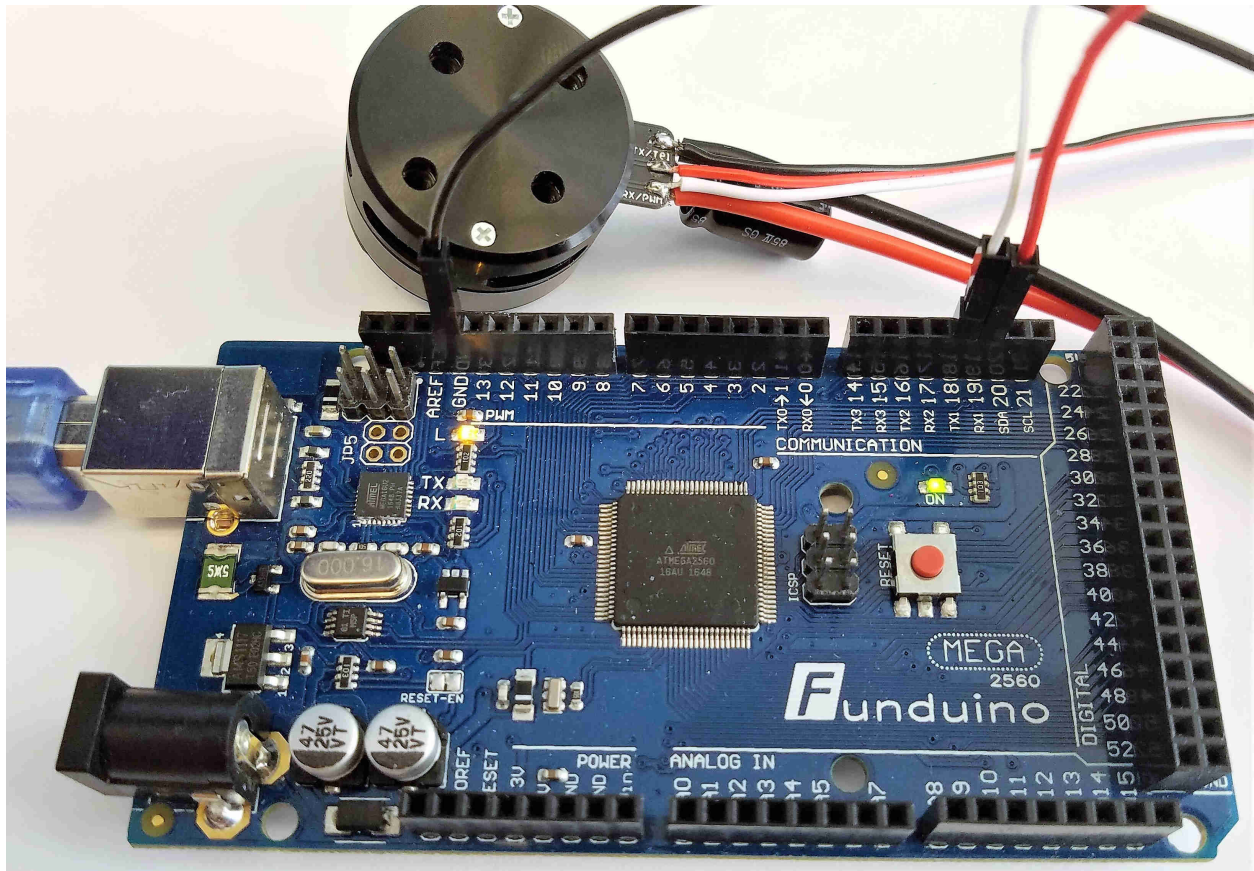
void setup() {
  // Setup the USB Serial connection
  Serial.begin(115200);

  // Setup the UART/TTL Serial
  UART_SERIAL.begin(115200);
}

void loop() {
  // Get byte from USB
  if(Serial.available())
    UART_SERIAL.write(Serial.read());

  // Get byte from UART/TTL
  if(UART_SERIAL.available())
    Serial.write(UART_SERIAL.read());
}
```

- Connect the GND of your Arduino to the GND (black) wire on the motor
- Connect the TXx of your Arduino to the RX (white) wire on the motor, where x is the serial number used in your sketch
- Connect the RXx of your Arduino to the TX (red) wire on the motor, where x is the serial number used in your sketch

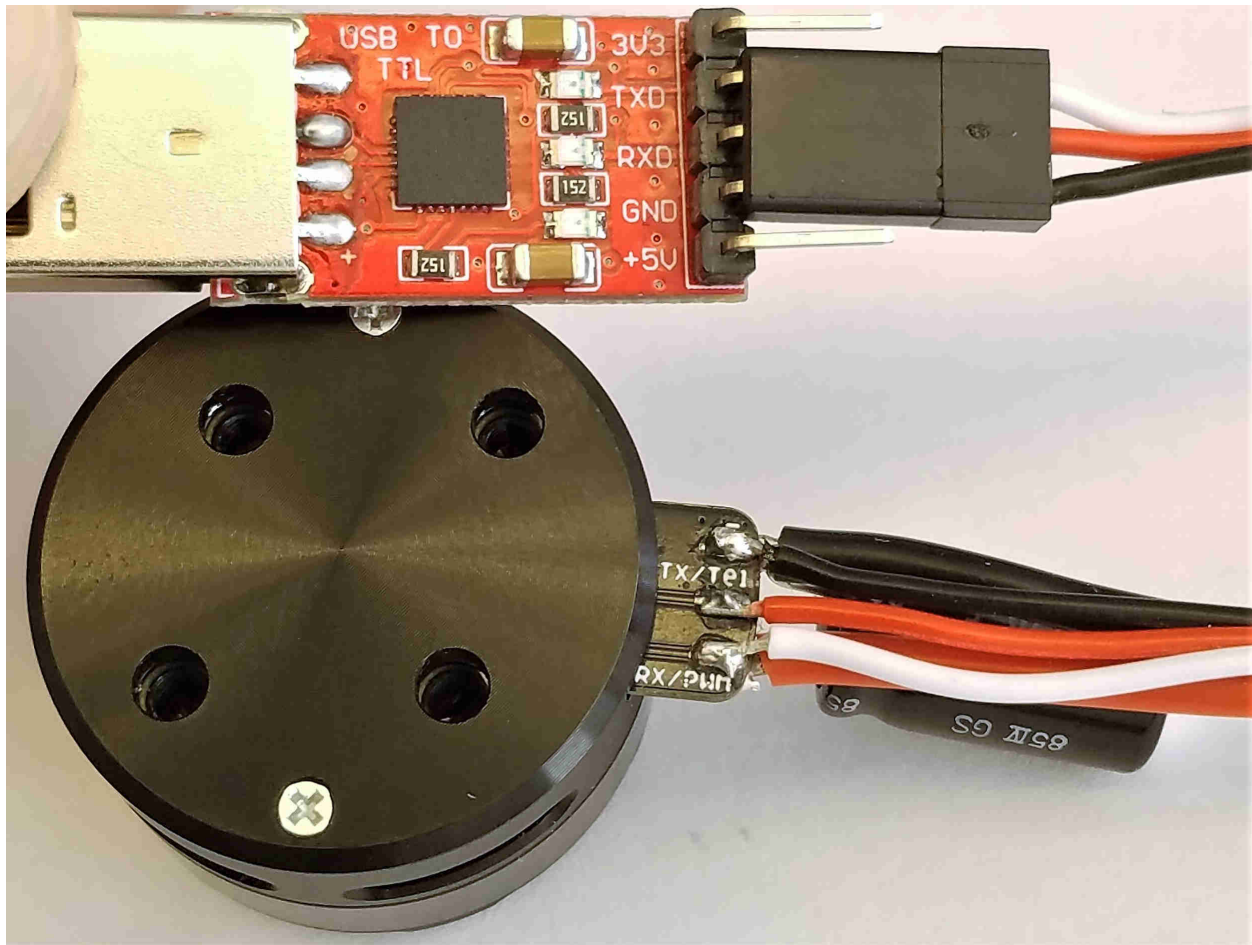


Motor connected to Arduino Mega using Serail1 for the motor and Serial for USB

## 9 Connect to IQ GUI: USB to UART

- Connect your FTDI, CP2102, CP2104 or similar device to USB
- Using jumper wires connect:
  - The GND of your USB to UART device to the GND (black) wire on the motor
  - The TX of your USB to UART device to the RX (white) wire on the motor
  - The RX of your USB to UART device to the TX (red) wire on the motor





Motor connected to CP2102 board

## 10 Change settings using DShot

- Open the BetaFlight Configurator and connect to your FC
- Select the CLI tab in the menu on the left
- The syntax for the dshotprog command is below:
 

```
dshotprog - program DShot ESC(s)
<index> <command>+
```
- NOTE: index is motor number minus 1, so...
  - Motor 1 is <index> 0
  - Motor 2 is <index> 1
  - Motor 3 is <index> 2
  - Motor 4 is <index> 3
- The following command values are supported by the IQ motors:
 

```
0 <Disarm>
```

```

1 <Beep 1>
2 <Beep 2>
3 <Beep 3>
4 <Beep 4>
5 <Beep 5>
7 <Spin direction CCW>
8 <Spin direction CW>
9 <3D mode off>
10 <3D mode on>
12 <Save direction and 3D mode>

```

- For example, if you want to change Motor 3 to CW 3D, type the following in the CLI:

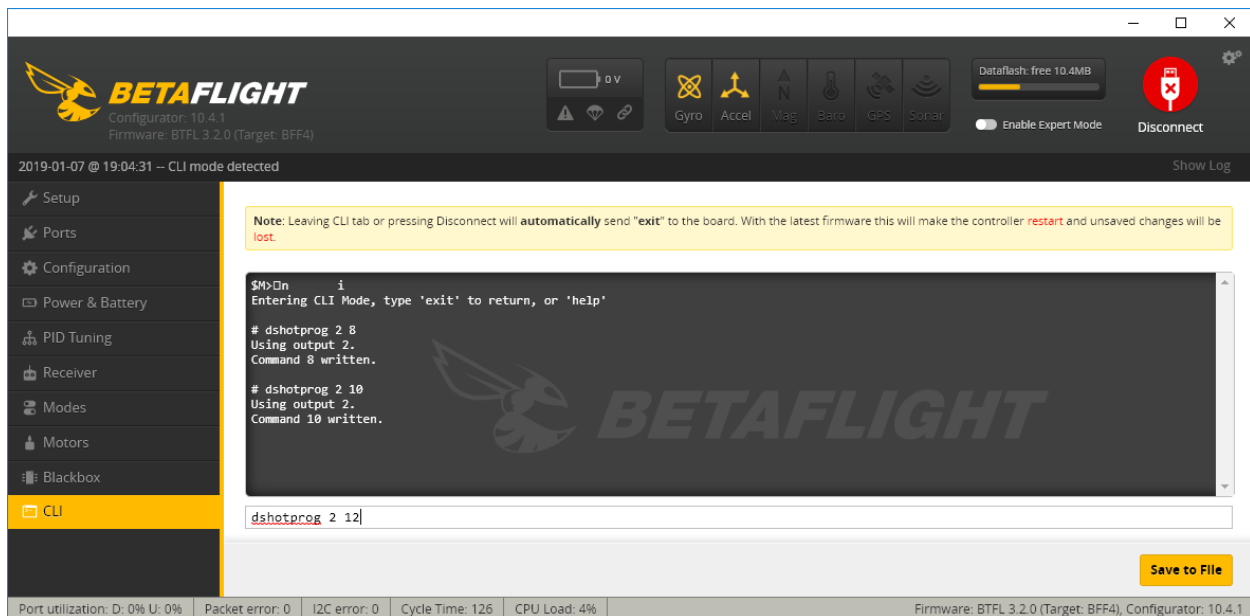
```

dshotprog 2 8
dshotprog 2 10

```

- After changing the spin direction or the 3D mode, you must save the changes
- Continuing the above example on Motor 3, type:

```
dshotprog 2 12
```



## 11 Pre-Made Application Settings

- These pre-made settings were designed to give users maximum performance in their specific application
- Settings Description: Speed Module
  - speed\_factory\_default - factory default, unconfigured

- APC6x4 - standard 2D drones using APC6x4 props, velocity (closed-loop) mode
- APC6x3R - reversible 3D drone using APC6x3R props, velocity mode
- APC5x4R - reversible 3D drone using APC5x4R props, velocity mode
- Dal5045 - standard 2D Dal5045 props, velocity mode
- pwm\_mode- use any prop as you would with a standard motor/ESC, PWM (open-loop) mode.
- 2s\_empty - voltage mode, motor will run as if it's running an empty 2s battery
- 2s\_full - voltage mode, motor will run as if it's running a full 2s battery
- 3s\_empty - voltage mode, motor will run as if it's running an empty 3s battery
- 3s\_full - voltage mode, motor will run as if it's running a full 3s battery
- 4s\_empty - voltage mode, motor will run as if it's running an empty 4s battery
- 4s\_full - voltage mode, motor will run as if it's running a full 4s battery
- 5s\_empty - voltage mode, motor will run as if it's running an empty 5s battery
- 5s\_full - voltage mode, motor will run as if it's running a full 5s battery
- 6s\_empty - voltage mode, motor will run as if it's running an empty 6s battery
- 6s\_full - voltage mode, motor will run as if it's running a full 6s battery

*Tip: The defaults for specific props allow you to run the motor in velocity (closed-loop) mode for maximum performance. If you're unable to do this, we suggest using PWM mode as you would with a standard motor/ESC or one of the voltage modes*

- Settings Description: Position Module
  - position\_factory\_default - factory default, unconfigured
- Make your own pre-settings (for advanced users)
  - Locate and open the IQ Control Center's "resources" folder
  - Open the "defaults" folder
  - Here you will see all of the .json files that correspond to the GUI's pre-made settings list
  - Create your own file by copying one of the factory default .json files and renaming it
  - In that new file, you can change the parameter values. Save this file once you're done editing.
  - Quit out of and reopen the IQ Control Center
  - Your new setting should now appear in the list of pre-settings in the General tab

## 12 Flashing New Firmware

### Download Firmware

- Go to: [www.iq-control.com/firmware](http://www.iq-control.com/firmware)
- Select and download your desired firmware (.bin file)

### **Put Motor in “Boot Mode” in the IQ User Interface**

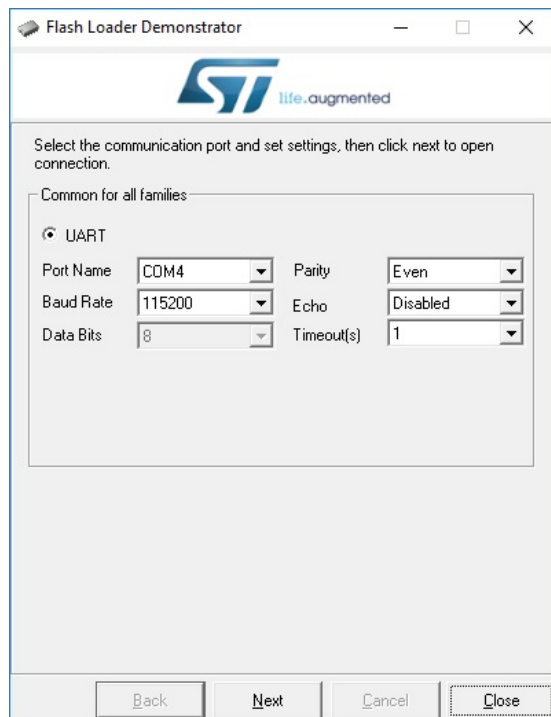
- Disconnect motor from your application and remove props
- Open the IQ GUI (see section 3 for download instructions)
- Connect your module (see sections 7, 8, or 9 for specific details)
  - Plug FC, Arduino, or USB to UART chip into computer
  - Plug/solder IQ module’s communication wires to above device
  - Plug/solder wires to power source (anything between 5V and 25V)
  - In the GUI, locate the correct com port in the upper right corner
  - Click the Connect button
- Navigate to the Firmware tab
- Click the Boot Mode button (this will disconnect the module)
- Keep the motor plugged in and go to the ST Demonstrator
- **WARNING:** You will have 3 seconds to click “Next” in the first window of the ST Demonstrator (see following subsection) after clicking “Boot Mode” in the IQ GUI. This is a safety feature. If 3 seconds elapses before you click “Next” in the first window of the ST Demonstrator, your module will reboot into “normal mode” and play the startup tune. Reconnect your module to the IQ GUI, put it in “Boot Mode,” and try again.

### **ST Demonstrator**

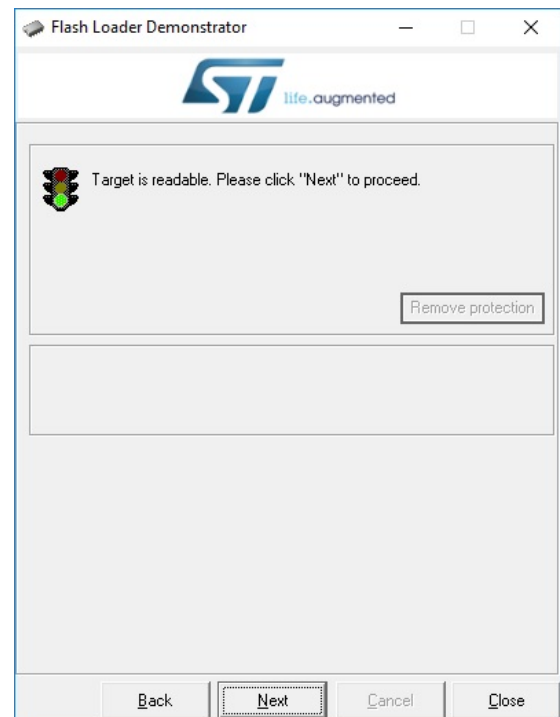
- Install the ST Demonstrator: <http://www.st.com/en/development-tools/flasher-stm32.html>
- Run the Demonstrator. It will open to window 1 (images below)
- Window 1
  - Locate the correct com port (should be the same as in the IQ GUI)
  - Set the Timeout to 1 second (optional)
  - Click “Next.” This will open window 2
- Click “Next.” This will open window 3
- Click “Next.” This will open window 4
- Window 4
  - Click “Download to device”
  - Click “...” and navigate to the folder in which you saved the firmware file
  - Change the file type from the default to .bin and select the binary file
  - Click the “Jump to the user program” and “Verify after download” boxes
  - Ensure “Erase necessary pages” is selected. DO NOT select “Global Erase”
  - Click “Next.” This will open window 5
- Window 5
  - Wait for program to load and verify



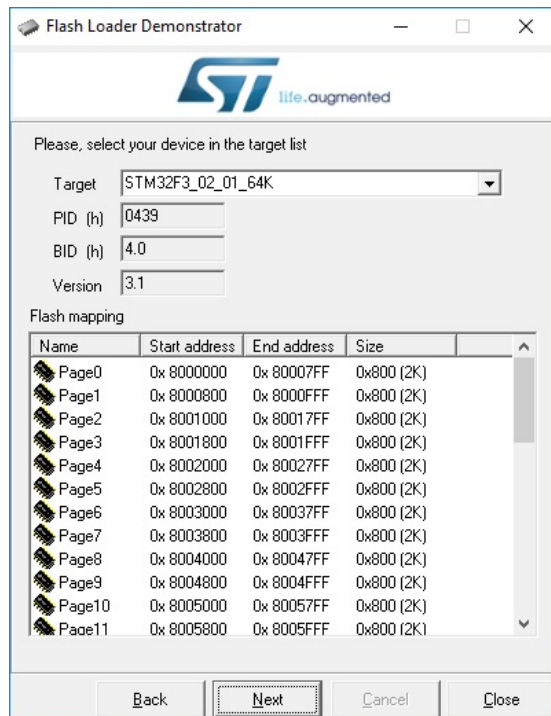
- If it verifies correctly, it will display green. You have successfully put new firmware on your motor!
- If it fails it will display red. Restart the process
- Contact IQ if you are unable to get the module back into programming mode



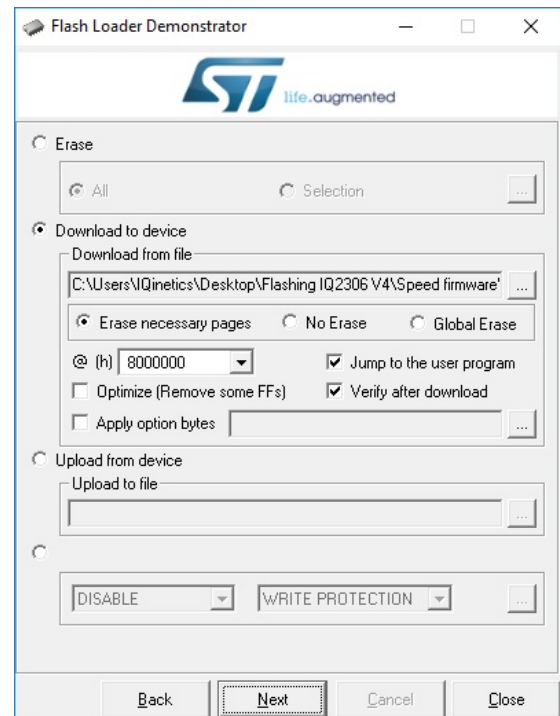
Window 1



Window 2



Window 3



Window 4



Window 5