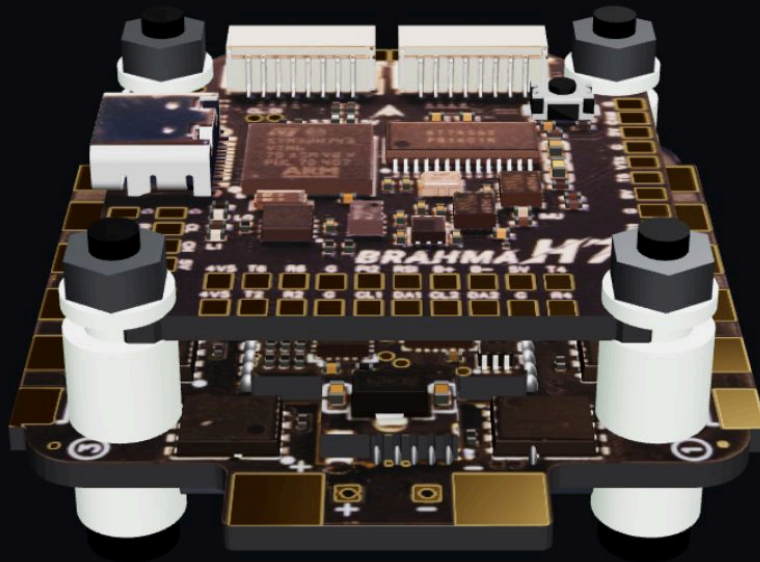


FLIGHT CONTROLLER + 4-IN-1 ESC · USER GUIDE

Stack H760



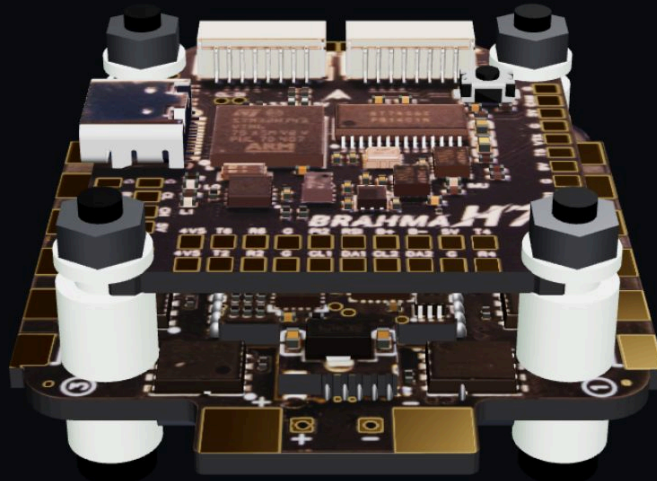
Contents

- 01 Stack H760 (Brahma H7 + Vishnu 60)
- 02 Brahma H7 flight controller
- 03 How to flash Betaflight on the Brahma H7
- 04 How to flash INAV on the Brahma H7
- 05 How to flash ArduPilot on the Brahma H7
- 06 How to flash PX4 on the Brahma H7
- 07 Connect a receiver to the Brahma H7
- 08 Connect a telemetry radio to the Brahma H7
- 09 Connect servos to the Brahma H7
- 10 Connect Herelink to the Brahma H7
- 11 Connect a Raspberry Pi to the Brahma H7
- 12 Vishnu 60 4-in-1 ESC

Need help?

WhatsApp or call **+91 80171 93277** (Mon-Sat, 12 PM-8 PM) · info@thedarkmatter.in · join the DM Community from thedarkmatter.in. The docs at thedarkmatter.in have this guide with every 3D view live: turn the board, click a step, and the pads light up.

01 Stack H760 (Brahma H7 + Vishnu 60)



Brahma H7 flight controller on a Vishnu 60 4-in-1 ESC, matched and shipped as one stack on the standard 30.5 × 30.5 mm mount. Drag the stack to turn it.

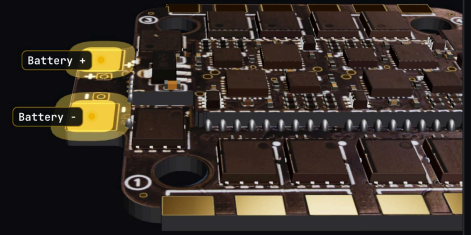
Assembly

1 Grommets and screws

Four M3 × 25 mm screws come up through the frame. A soft grommet goes in every mounting hole of both boards, fat body down, so frame vibration stays out of the gyro.

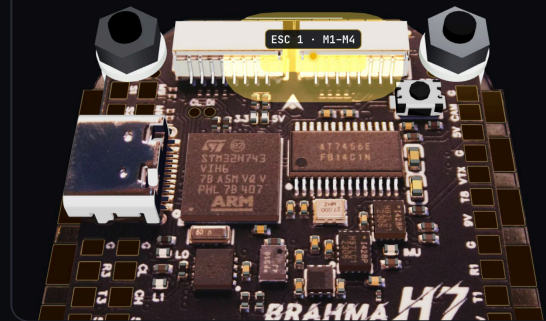
2 Vishnu 60 first

The Vishnu 60 goes on first, turned so its battery pads face away from the Brahma H7's ESC plug. Solder the capacitor and battery leads before you mount it.



3 Brahma H7 on top

The Brahma H7 goes on top, arrow pointing forward. Join the two boards with the 8-pin cable from the box, then fit the four nuts: snug, not tight, so the grommets keep their give.



The two boards

Brahma H7

Specifications, every pad, UARTs, wiring, firmware.

[Open the Brahma H7 docs →](#)

Vishnu 60

Specifications, battery, motor numbering.

[Open the Vishnu 60 docs →](#)

Firmware

The stack ships with the firmware you choose when you order. To reflash:

Betaflight

Freestyle and racing: the fastest tune and the biggest community.

[How to flash Betaflight on the Brahma H7 →](#)

INAV

GPS features for fixed-wing and long range: return to home, waypoints.

[How to flash INAV on the Brahma H7 →](#)

ArduPilot

Autonomous missions, mapping, delivery and industrial UAVs.

[How to flash ArduPilot on the Brahma H7 →](#)

PX4

Research and development, ROS 2, custom autonomy.

[How to flash PX4 on the Brahma H7 →](#)

In the box

- 1 × Brahma H7
- 1 × Vishnu 60
- 2 × 8-pin ESC-FC cable
- 8 × grommets
- 1 × 1000 µF low-ESR capacitor
- 1 × XT60 battery connector
- 12 AWG battery wire
- 4 × M3 25 mm screws
- 4 × M3 nuts
- 1 × tin box

Downloads

Stack H760 manual (PDF)

The original slide manual

Warranty

The Stack H760 has a 6-month warranty against manufacturing defects from the date of purchase. Keep your invoice: we need a copy for any claim.

Not covered:

- Physical damage from use, such as water damage or tampered components.
- A burnt circuit from wrong wiring or overloading.
- A burnt ESC from spinning motors without propellers above 15% throttle while checking motor direction.
- A burnt ESC from arming without propellers in Angle mode: the flight controller keeps trying to level the craft and the motors will not slow down when you lower the throttle.

How a claim works: we inspect the board for physical damage and the points above. If it is eligible we replace it at no cost, and help you with your build. WhatsApp or call [+91 80171 93277](tel:+918017193277).

02 Brahma H7 flight controller

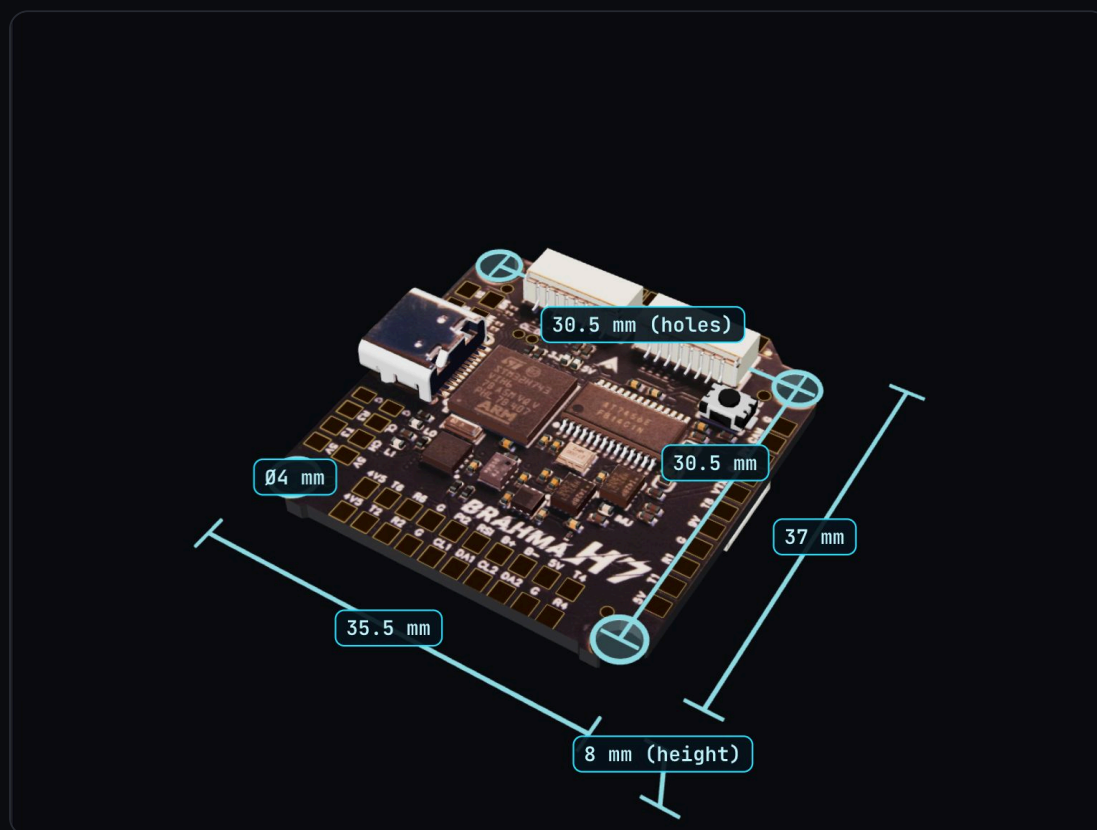
A 480MHz STM32H743 flight controller for Betaflight, INAV, ArduPilot and PX4: ten motor outputs, two servo outputs, seven UARTs, CAN, dual gyroscopes and a plug-and-play digital VTX port on a standard 30.5 × 30.5 mm mount.

Specifications

Product	Darkmatter® Brahma H7 flight controller
Processor (MCU)	STM32H743VIT6 ARM, 480MHz
Gyroscope (IMU)	Dual ICM42688P
Barometer	DPS310
On-screen display	AT7456E
Blackbox	microSD card slot
USB	Type-C
Firmware	Betaflight, INAV, ArduPilot, PX4. We flash the one you choose when you order; you can reflash any of them later.
Power input	2–6S LiPo
Mounting	30.5 × 30.5 mm, 4 mm holes
Dimensions	37 × 35.5 × 8 mm
Weight	9 g
5V output	9 groups, 3A total
9V output	1 pad, 3A. Switch it on and off from your radio (PINIO1), for example to keep the VTX off until the GPS has locked
4.5V output	2 pads: 500mA when powered from USB (for the receiver on the bench), 3A on battery
3.3V output	Up to 1A
LED	2 NeoPixel LED pad groups
Buzzer	BZ+ and BZ– pads for a 5V active buzzer
Motors	10: two plug-and-play ESC ports (M1–M4, M5–M8) and solder pads for M9, M10
PWM outputs	13 in total: 10 motors, 2 servos (S1, S2), 1 LED
Digital VTX	Plug-and-play port
CAN	Yes: CH and CL pads
SPI	2 breakouts

BOOT button	Hold it while plugging in USB to enter DFU mode for flashing
RSSI input	RSI pad
SmartPort	Any UART TX pad
UART	7: UART1, 2, 3, 4, 6, 7, 8
UART flow control	UART7 has RTS and CTS
ADC	6
Pin IO	2. PINIO1 switches the 9V pad; PINIO2 is the PI2 pad (3.3V logic) for a relay or similar
I ² C	2 buses: CL1/DA1 and CL2/DA2, for compasses, sonar and more
Operating temperature	-20 °C to 70 °C
Made in	India

Dimensions and mounting



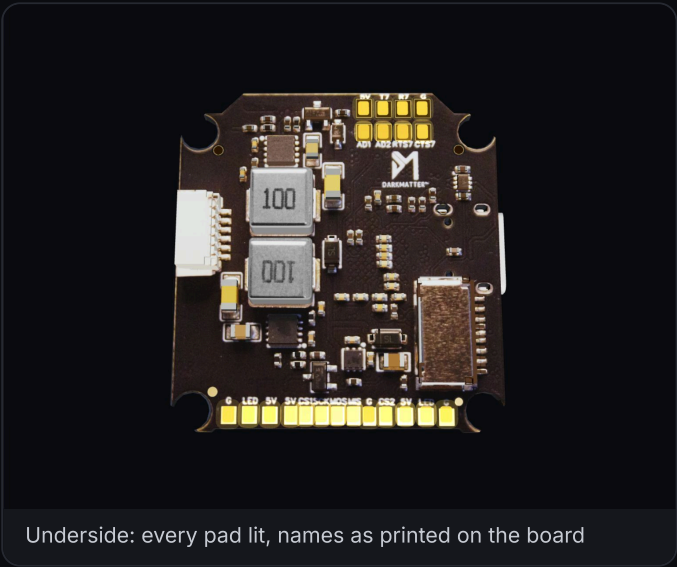
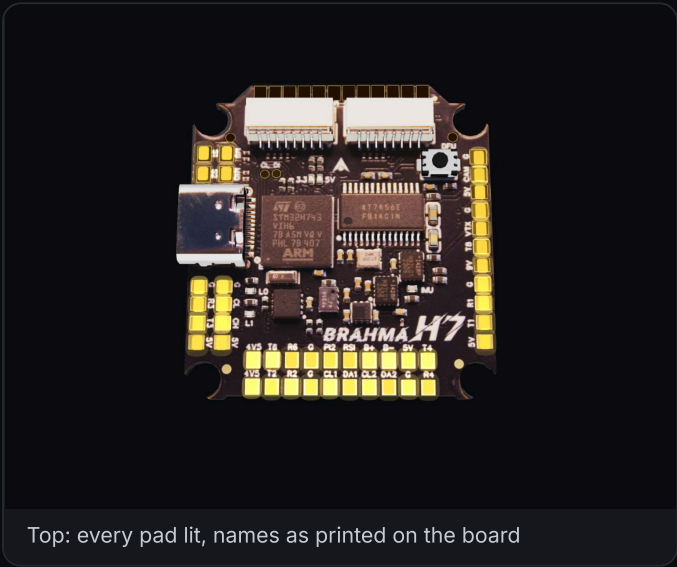
37 × 35.5 mm, 8 mm tall with the connectors, 1.6 mm PCB. Holes 4 mm on a 30.5 × 30.5 mm pattern. Use the soft grommets from the box so frame vibration stays out of the gyro.

UARTs and serial ports

UART	Pads	ArduPilot	Notes
UART1	T1 R1	SERIAL2 · Telemetry 2	
UART2	T2 R2	SERIAL3 · GPS 1	GPS
UART3	T3 R3	SERIAL4 · GPS 2	
UART4	T4 R4	SERIAL6	Also on the digital VTX plug
UART6	T6 R6	SERIAL7 · RC input	Receiver
UART7	T7 R7 RTS7 CTS7	SERIAL1 · Telemetry 1 (with flow control)	Underside, with RTS/CTS
UART8	T8	SERIAL5	TX on the VTX column; RX8 is ESC telemetry on ESC plug 1

ArduPilot order from the Brahma H7 ArduPilot build (manual, default UART order).

Every pad



Every pad, by name. The same names are printed next to the pads on the board.

- R1R2R3R4R6R7 undersideT1T2T3T4T6T7 undersideT84V5 × 2
- 5V × 55V underside × 49VAD1 undersideAD2 undersideBZ+BZ-CAMCHCLCL1
- CL2CS1 undersideCS2 undersideCTS7 undersideDA1DA2G × 8G underside × 4LED underside × 2
- M10M9MISO undersideMOSI undersidePI2RSIRTS7 undersideS1S2SCK underside
- VTX

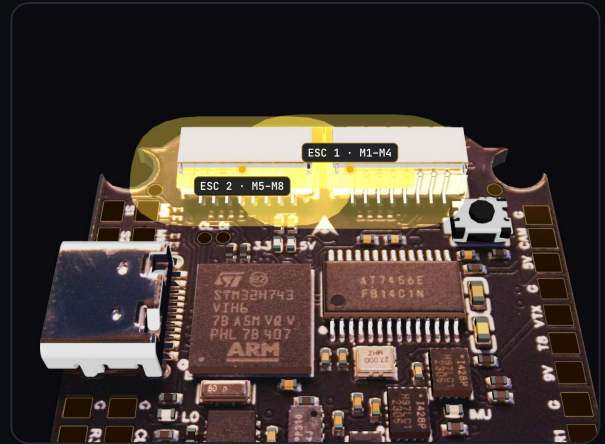
Wiring

1 ESC

Plug a Vishnu 4-in-1 ESC into **ESC 1** with the 8-pin cable: M1–M4. A second 4-in-1 ESC goes into **ESC 2** for M5–M8 (hexa- and octocopters). Pin order, ESC 1: M4, M3, M2, M1, RX8 (ESC telemetry), C1 (current), GND, VB (battery voltage). ESC 2: M8, M7, M6, M5, RX7, C2, GND, DNC.

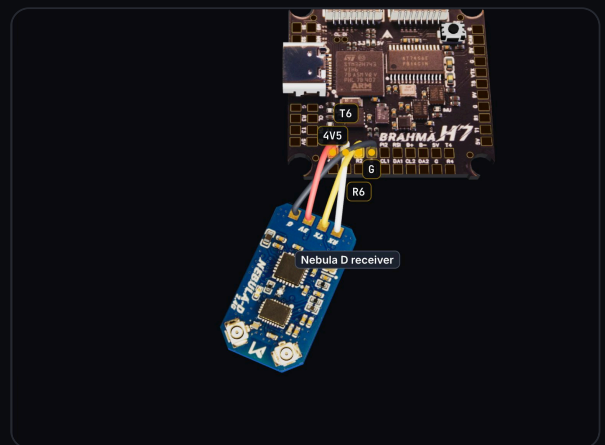
IMPORTANT

Do not connect anything to the DNC (do not connect) pin of ESC 2.



2 Receiver (ExpressLRS / CRSF)

Nebula D or any CRSF receiver on UART6: receiver TX → **R6**, receiver RX → **T6**, 5V → **4V5**, G → **G**. The 4.5V pad is live on USB too, so the receiver works while you configure on the bench.



3 GPS and compass

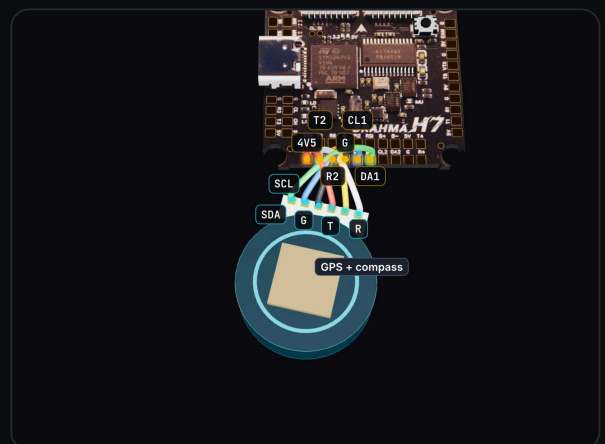
GPS 1 on UART2: GPS TX → **R2**, GPS RX → **T2**. Compass on I²C 1: SCL → **CL1**, SDA → **DA1**. Power from the **4V5** pad next to T2 and the **G** next to R2: 4V5 is live on USB too, so the GPS gets its fix on the bench without a battery. A second GPS goes on UART4 (T4/R4) with I²C 2 (CL2/DA2); in ArduPilot UART4 is SERIAL6, so set `SERIAL6_PR0T0C0L = 5` (GPS). UART3 (T3/R3) is ArduPilot's GPS 2 port by default.

TIP

Mount the GPS on a mast, away from the VTX, the ESC and the battery leads, with its arrow pointing forward. The compass reads the current in those wires as magnetic noise.

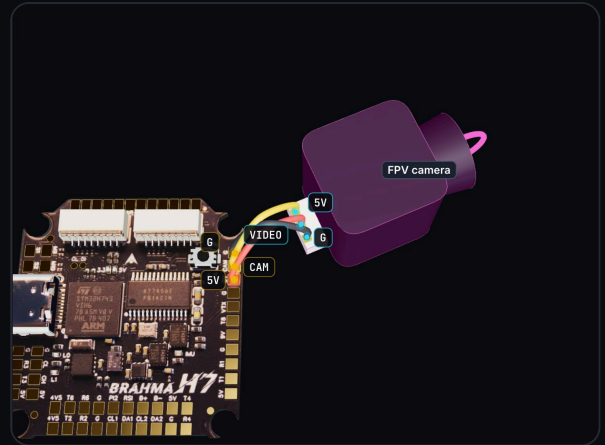
GOOD TO KNOW

If you use a DJI Air Unit on the 6-pin plug, put the second GPS on another UART such as UART1 or UART3: the plug also uses UART4.



4 Analog camera

Camera video → **CAM**, power from **5V** and **G** on the same column.



5 Analog VTX

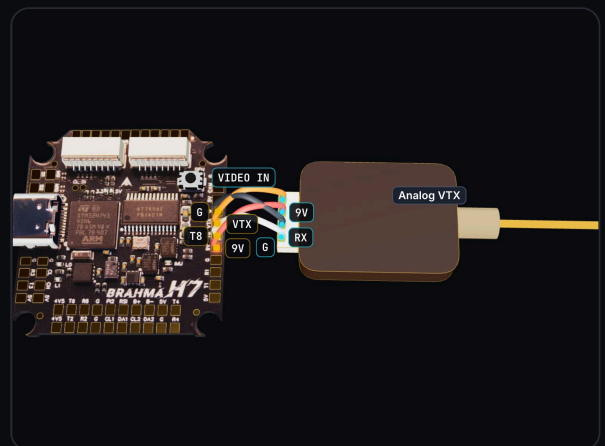
The video goes through the flight controller so the OSD can draw on it: camera video → **CAM**, and **VTX** → the video transmitter's video input.

VTX video in ← **VTX**, power from **9V** and **G**, and the VTX control wire (SmartAudio or Tramp) on **T8**; in Betaflight set it as the VTX peripheral on UART8 in the Ports tab. The 9V pad can be switched off from your radio (PINIO1), so the VTX stays off until the GPS has locked.

Pick **TBS SmartAudio** or **IRC Tramp** for that UART in the Ports tab (check which one your VTX uses) and load a VTX table in the Video Transmitter tab, or the VTX menu will not work. Camera and VTX grounds go to the same ground pad, or pads next to each other, to keep lines out of the picture.

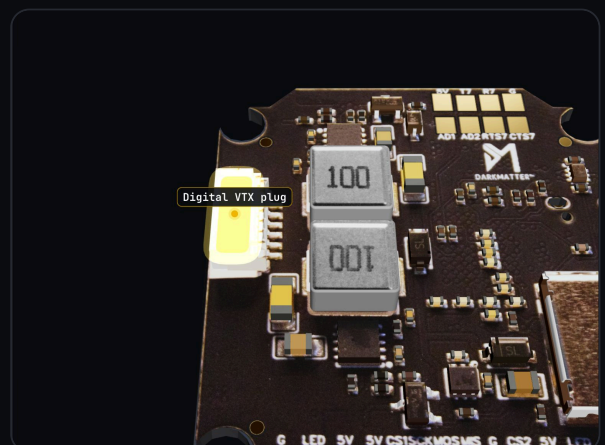
IMPORTANT

Screw the antenna on before the VTX gets power. A VTX that transmits without an antenna can burn out in seconds.



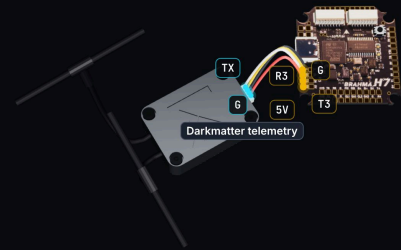
6 Digital VTX

DJI, Walksnail and other digital systems plug into the 6-pin port on the underside: R3, G, R4, T4, G, 9V (UART4 for the video link, R3 for SBUS from the air unit).



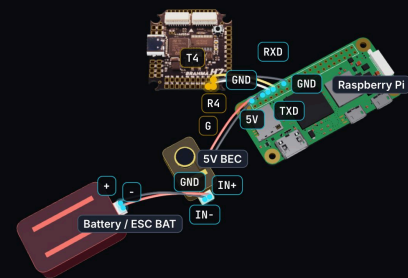
7 Telemetry radio

The Darkmatter telemetry air unit (or any MAVLink telemetry radio) on UART3: its TX → **R3**, its RX → **T3**, **5V**, **G**, all in one column on the left edge. [Full guide](#) →



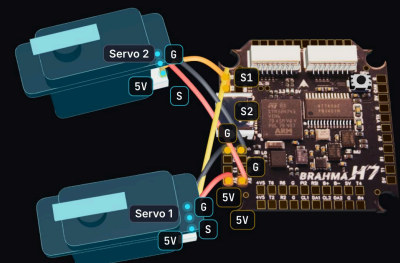
8 Companion computer

A Raspberry Pi on UART4: **T4** → **Pi pin 10** (RXD), **R4** → **Pi pin 8** (TXD), **G** → **Pi pin 14** (GND). The Pi is powered by its own 5V BEC on pins 2 and 6, fed from the ESC battery pads or any suitable battery supply. The Pi gets its own 5V BEC, never the flight controller's 5V. [Full guide](#) →



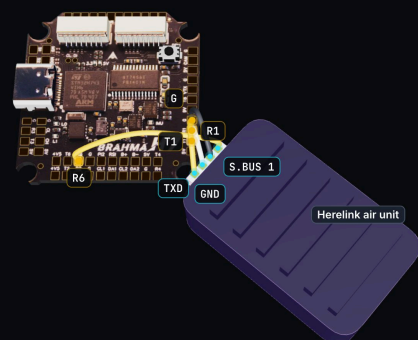
9 Servos

Two servo pads, **S1** and **S2** (outputs 11 and 12), with 5V and G beside them. [Full guide](#) →



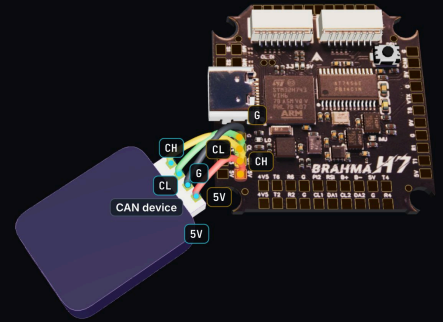
10 Herelink

S.Bus out 1 → **R6**, telemetry on UART1 (**T1/R1**). [Full guide](#) →



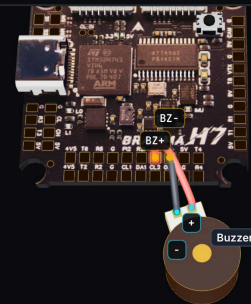
11 CAN

DroneCAN GPS, airspeed and other CAN devices: **CH** (CAN high), **CL** (CAN low), **5V**, **G**.



12 Buzzer

A 5V active buzzer: + → **BZ+**, - → **BZ-**.



Firmware

Every Brahma H7 ships with the firmware you pick when you order. You can reflash it any time with one of these guides.

Betaflight

Freestyle and racing: the fastest tune and the biggest community.

[How to flash Betaflight on the Brahma H7 →](#)

INAV

GPS features for fixed-wing and long range: return to home, waypoints.

[How to flash INAV on the Brahma H7 →](#)

ArduPilot

Autonomous missions, mapping, delivery and industrial UAVs.

[How to flash ArduPilot on the Brahma H7 →](#)

PX4

Research and development, ROS 2, custom autonomy.

[How to flash PX4 on the Brahma H7 →](#)

Connect accessories

- [Connect a receiver \(ExpressLRS, IBUS\)](#)
- [Connect servos](#)
- [Connect a telemetry radio](#)
- [Connect Herelink](#)
- [Connect a companion computer \(Raspberry Pi\)](#)

In the box

- 1 × Brahma H7 flight controller
- 2 × SH 1.0 mm 8-pin cable, 30 mm (flight controller to ESC)

Downloads

Brahma H7 manual (PDF)

The original slide manual

Firmware files: see each firmware guide above, or **all downloads**.

Warranty

The Brahma H7 flight controller has a 6-month warranty against manufacturing defects from the date of purchase. Keep your invoice: we need a copy for any claim.

Not covered:

- Physical damage from use, such as water damage or tampered components.
- A burnt circuit from wrong wiring or overloading.

How a claim works: we inspect the board for physical damage and the points above. If it is eligible we replace it at no cost, and help you with your build. WhatsApp or call [+91 80171 93277](tel:+918017193277).

03 How to flash Betaflight on the Brahma H7

Betaflight 4.5.2 for the Brahma H7 is a Darkmatter build (the official Betaflight target is on its way). About 5 minutes.

You need: [Betaflight Configurator](#), a USB-C data cable

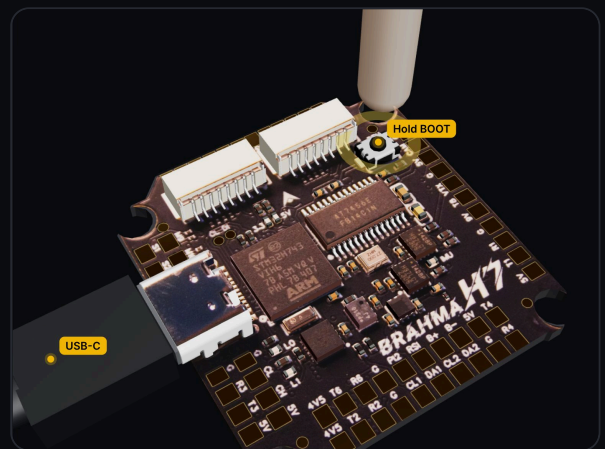
1 Download the firmware

betaflight_4.5.2_BRAHMA_H7_CRSF.hex

Betaflight 4.5.2, CRSF build

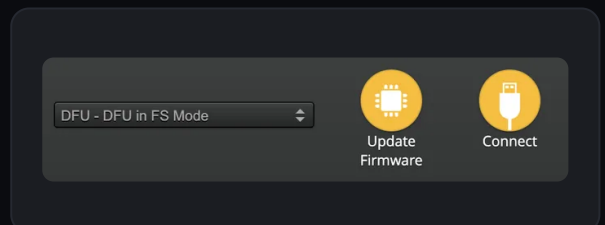
2 Hold BOOT and plug in USB-C

Unplug the battery. Press and hold the **BOOT** button on the Brahma H7, then plug the USB-C cable into your computer. Release the button once it is plugged in: the board starts in DFU (flashing) mode.



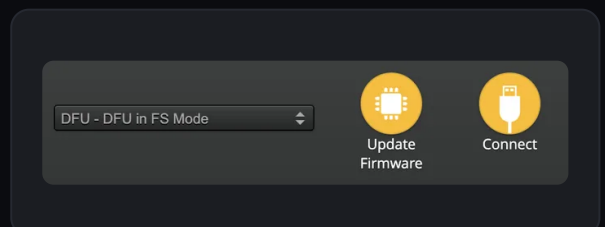
3 Open Betaflight Configurator

Open Betaflight Configurator. The board shows up as **DFU** instead of a COM port. If it does not, try another cable (some USB-C cables carry power only) and hold BOOT while plugging in again. On Windows, a board that shows up as "STM32 BOOTLOADER" but not as DFU needs its driver fixed: run ImpulseRC Driver Fixer, or switch it to WinUSB with Zadig.



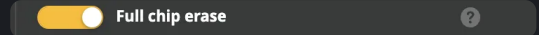
4 Update Firmware

With **DFU** selected in the port list, click **Update Firmware**.



5 Full chip erase on

Turn on **Full chip erase**.



6 Load the file

Click **Load Firmware [Local]** and choose `betaflyght_4.5.2_BRAHMA_H7_CRSF.hex`.

Load Firmware [Local]

 `betaflyght_4.5.2_BRAHMA_H7_CRSF.hex`

7 Flash

Click **Flash Firmware** and wait for erasing and flashing to finish. Do not unplug the board. If it stops with "read protection not active", untick **Full chip erase** and flash again, then type `defaults` in the CLI before pasting the settings below.

Flash Firmware

Erasing ...

8 Close the warning, open the CLI

Connect. Betaflight warns that the accelerometer is not calibrated: click **Close** (calibrate it later on a level surface in the Setup tab). Then open **CLI** from the menu on the left.

Warning

The following problems with your configuration were detected:

- the accelerometer is enabled but it is not calibrated.

If you plan to use the accelerometer, please follow the instructions for 'Calibrate Accelerometer' on the 'Setup' tab. If any function that requires the accelerometer (auto level modes, GPS rescue, ...) is enabled, arming of the craft will be disabled until the accelerometer has been calibrated.

If you are not planning on using the accelerometer it is recommended that you disable it in 'System configuration' on the 'Configuration' tab.

You need to fix these problems before attempting to fly your craft.

Close

Setup

Ports

Configuration

Power & Battery

Failsafe

Presets

PID Tuning

Receiver

Modes

Adjustments

Motors

OSD

Video Transmitter

LED Strip

Sensors

Tethered Logging

Blackbox

CLI

9 Paste the Brahma H7 settings

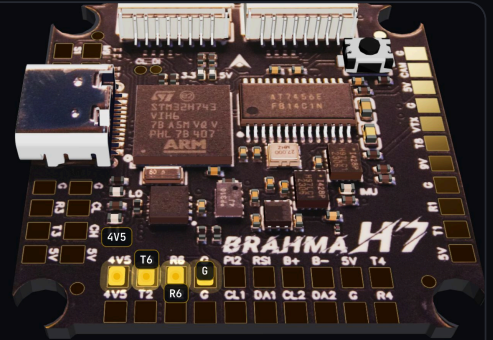
Paste these lines into the CLI command box and press Enter. `save` stores them and reboots the board.

```
set gyro_1_sensor_align = CW180
set gyro_2_sensor_align = CW180
set vcd_video_system = AUTO
set osd_displayport_device = MAX7456
set pinio_config = 129,1,1,1
save
```

Write your command here. Press Tab for AutoComplete.

10 Receiver

This build is for CRSF receivers (ExpressLRS, Crossfire). In the **Ports** tab, turn on **Serial Rx** for **UART6**, where the receiver is wired.



Before the first flight

1 Accelerometer

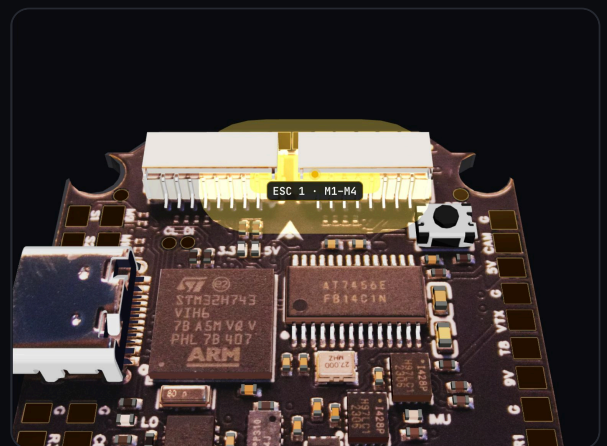
Setup tab: put the quad on a level surface and click **Calibrate Accelerometer**. Tilt it by hand: the 3D model on screen must follow.

2 Receiver

Ports: Serial Rx on UART6. **Receiver** tab: Serial (via UART), provider CRSF, telemetry on. Move each stick and check the bars follow the right channel (AETR for ExpressLRS). Set a switch for ARM in **Modes**.

3 Motors, props off

Battery through a smoke stopper, props off. In **Motors**, spin each motor at low throttle and check it matches the position and direction in Betaflight's diagram. Reorder or reverse there instead of re-soldering.



4 Failsafe

Failsafe tab (turn on Expert Mode if you do not see it): stage 2 is **Drop** by default, right for freestyle. For long range use GPS Rescue, and only after the GPS shows 8 or more satellites and you have tested it close and low. Switch the radio off with the props off and check the quad disarms.

5 RPM filter

With BlueJay on the Vishnu ESC, use **DShot300**, turn on bidirectional DShot and set the motor poles (usually 14). In the Motors tab every motor must show RPM and an error rate (E) of 0%; if not, drop to DShot150.

FlySky IBUS receivers

An IBUS receiver needs a Betaflight build with IBUS selected. The IBUS build for the Brahma H7 is coming soon. See [Connect a receiver](#).

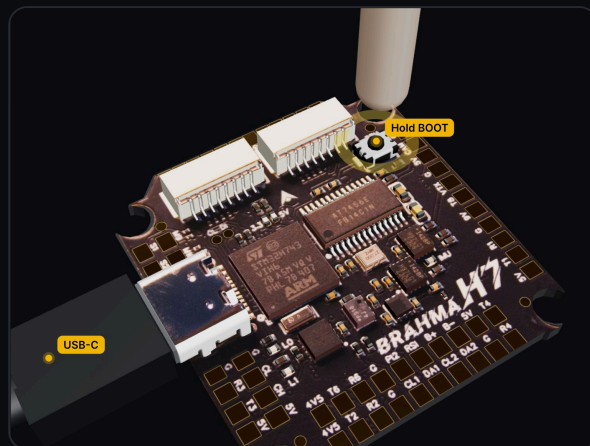
04 How to flash INAV on the Brahma H7

The Brahma H7 is an official INAV target, `BRAHMA_H7`. About 10 minutes, including the setup wizard.

You need: [INAV Configurator](#), a USB-C data cable

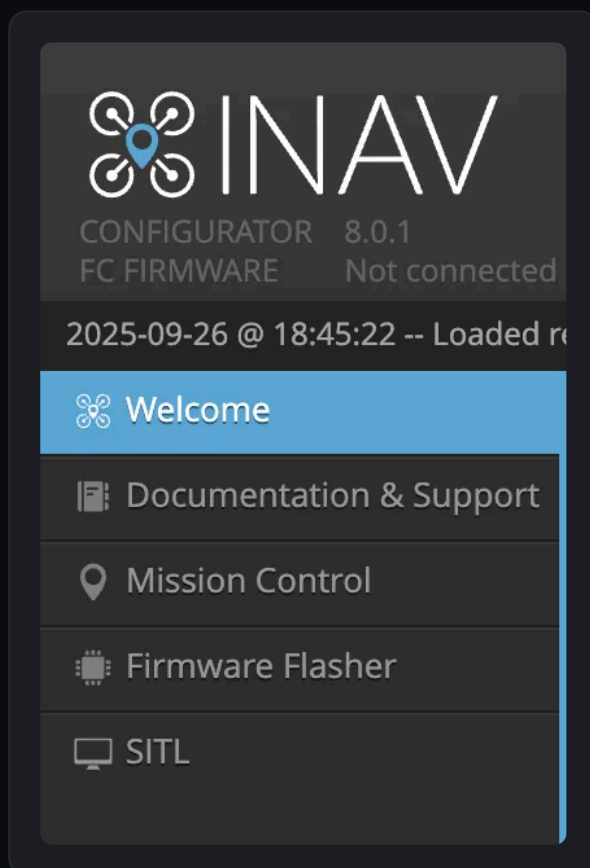
1 Hold BOOT and plug in USB-C

Unplug the battery. Press and hold the **BOOT** button on the Brahma H7, then plug the USB-C cable into your computer. Release the button once it is plugged in: the board starts in DFU (flashing) mode.



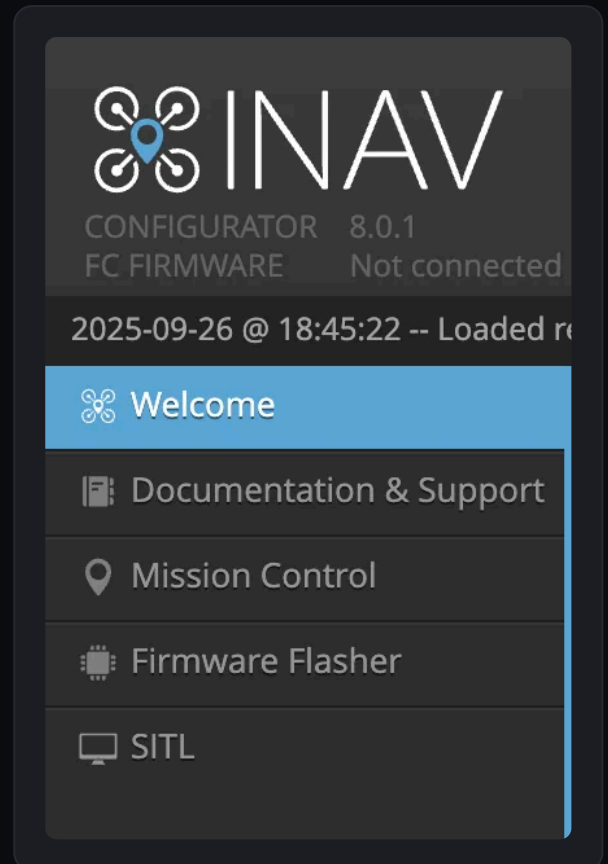
2 Open INAV Configurator

Open INAV Configurator. The board shows up as **DFU** instead of a COM port. If it does not, try another cable (some USB-C cables carry power only) and hold BOOT while plugging in again. On Windows, a board that shows up as "STM32 BOOTLOADER" but not as DFU needs its driver fixed: run ImpulseRC Driver Fixer, or switch it to WinUSB with Zadig.



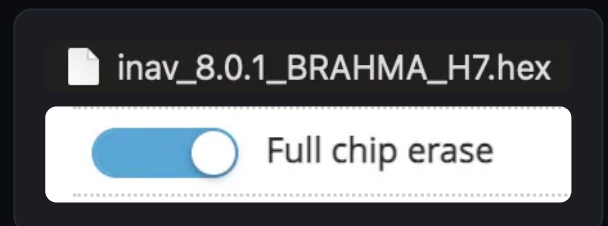
3 Firmware Flasher

Once the board shows as DFU, open **Firmware Flasher**.



4 Choose the target

Target: **BRAHMA_H7**, the latest stable release. Turn on **Full chip erase**.



5 Flash

Click **Load Firmware [Online]**, then **Flash Firmware**. Wait for erasing and flashing to finish.



6 Craft type

Connect. The setup wizard asks what you are building: pick your craft so INAV loads the right defaults.

Default values

INAV Configurator would like to know which kind of UAV you are configuring. Based on this information it will modify some default values to unlock the best flying performance.

Avoid blindly restoring all PID and filter configuration from the previous version of INAV. The best result can be obtained by retuning from defaults!

Mini Quad with 3" propellers

Mini Quad with 5" propellers

Mini Quad with 7" propellers

Airplane with a Tail

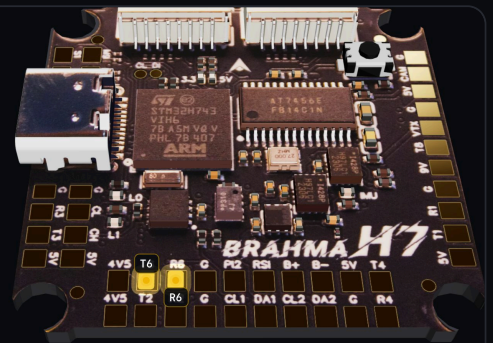
Airplane without a Tail (Wing, Delta, etc)

Rovers & Boats

Keep current settings (Not recommended)

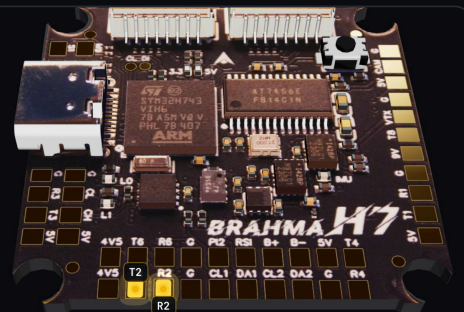
7 Receiver

Receiver serial port: **UART6**. Protocol: **CRSF** for ExpressLRS.



8 GPS

GPS serial port: **UART2** (T2/R2), 115200 baud, protocol UBLOX.



05 How to flash ArduPilot on the Brahma H7

ArduPilot for the Brahma H7 is a Darkmatter build with the bootloader included, flashed once with STM32CubeProgrammer. About 10 minutes.

You need: [STM32CubeProgrammer](#), [Mission Planner](#), a USB-C data cable

1 Download the firmware

ArduCopter (multirotors)

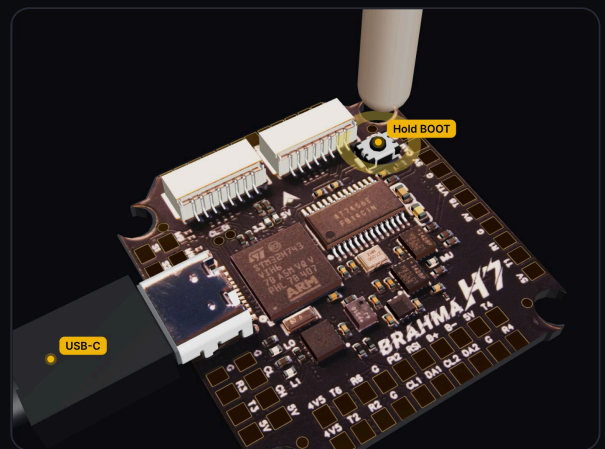
ArduCopter for Brahma H7, with bootloader

ArduPlane (fixed wing, VTOL)

ArduPlane for Brahma H7, with bootloader

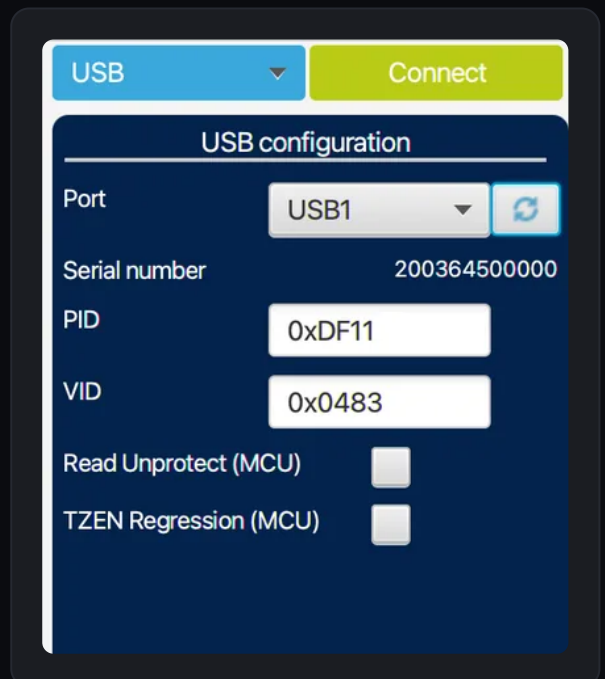
2 Hold BOOT and plug in USB-C

Unplug the battery. Press and hold the **BOOT** button on the Brahma H7, then plug the USB-C cable into your computer. Release the button once it is plugged in: the board starts in DFU (flashing) mode.



3 Open STM32CubeProgrammer

Open STM32CubeProgrammer. The board shows up as **DFU** instead of a COM port. If it does not, try another cable (some USB-C cables carry power only) and hold BOOT while plugging in again. On Windows, a board that shows up as "STM32 BOOTLOADER" but not as DFU needs its driver fixed: run ImpulseRC Driver Fixer, or switch it to WinUSB with Zadig.



4 Connect STM32CubeProgrammer

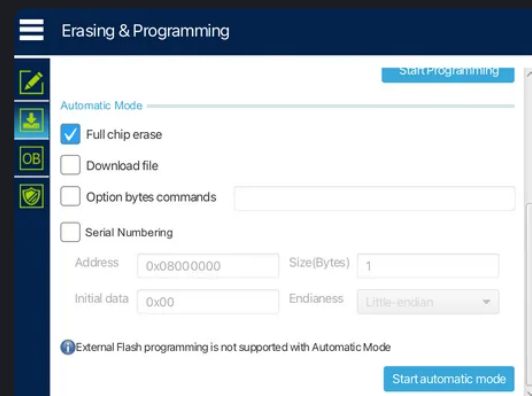
Choose **USB** in the drop-down and click **Connect**.



The screenshot shows the 'USB configuration' window in STM32CubeProgrammer. At the top, there is a blue tab labeled 'USB' and a green 'Connect' button. Below this, the 'USB configuration' section has a 'Port' dropdown set to 'USB1' with a refresh icon, a 'Serial number' field showing '200364500000', a 'PID' field with '0xDF11', and a 'VID' field with '0x0483'. There are also checkboxes for 'Read Unprotect (MCU)' and 'TZEN Regression (MCU)', both of which are currently unchecked. At the bottom of the window is a large green 'Connect' button.

5 Full chip erase

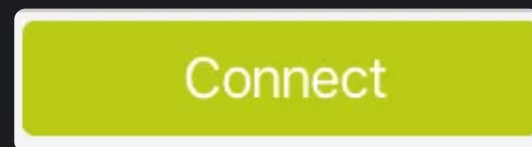
Open the second menu item, **Erasing & Programming**, and click **Full chip erase**. Confirm, and wait for the log to say it is done and to ask you to disconnect. Disconnect.



The screenshot shows the 'Erasing & Programming' window. On the left is a sidebar with icons for 'Automatic Mode', 'Full chip erase', 'Download file', 'Option bytes commands', and 'Serial Numbering'. The 'Automatic Mode' tab is selected. In the main area, the 'Full chip erase' checkbox is checked. Below it are fields for 'Address' (0x08000000), 'Size (Bytes)' (1), 'Initial data' (0x00), and 'Endianness' (Little-endian). A message at the bottom states 'External Flash programming is not supported with Automatic Mode'. There are 'Start Programming' and 'Start automatic mode' buttons.

6 Connect again

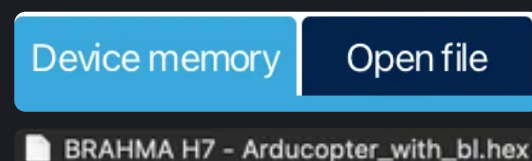
Click **Connect** again.



The screenshot shows a large green 'Connect' button.

7 Open the file

Go to the first tab, click **Open file** and choose the .hex you downloaded.



The screenshot shows the 'Device memory' tab in STM32CubeProgrammer. It has a blue 'Device memory' tab and a green 'Open file' button. Below these is a file list showing 'BRAHMA H7 - Arducopter_with_bl.hex'.

8

Download

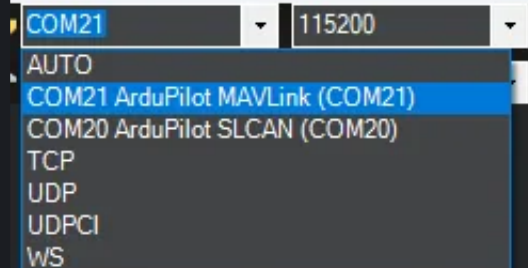
Click **Download** and wait until flashing is complete.

Download

9

Mission Planner

Unplug, plug USB in again (without BOOT), open [Mission Planner](#), choose the COM port that shows as **ArduPilot**, and click **Connect**.



Serial ports in ArduPilot

ArduPilot	Port	Default
SERIAL0	USB	MAVLink
SERIAL1	UART7 (T7/R7, with RTS/CTS)	Telemetry 1
SERIAL2	UART1 (T1/R1)	Telemetry 2
SERIAL3	UART2 (T2/R2)	GPS 1
SERIAL4	UART3 (T3/R3)	GPS 2
SERIAL5	UART8	—
SERIAL6	UART4 (T4/R4)	—
SERIAL7	UART6 (T6/R6)	RC input

06 How to flash PX4 on the Brahma H7

Three parts: **A** the bootloader, once per board, with STM32CubeProgrammer; **B** PX4 1.16.1 from QGroundControl, for every update; **C** a quick check. Keep the battery disconnected while flashing.

You need: STM32CubeProgrammer, QGroundControl 5, a USB-C data cable (charge-only cables will not work)

1 Download both files

darkmatter_brahma_h7_bootloader.bin

PX4 bootloader for Brahma H7

darkmatter_brahma_h7_default.px4

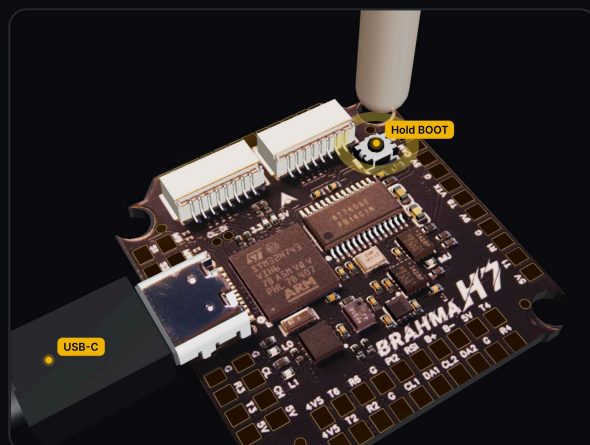
PX4 1.16.1 firmware for Brahma H7

2 A · Unplug everything

Disconnect the battery and the USB-C cable.

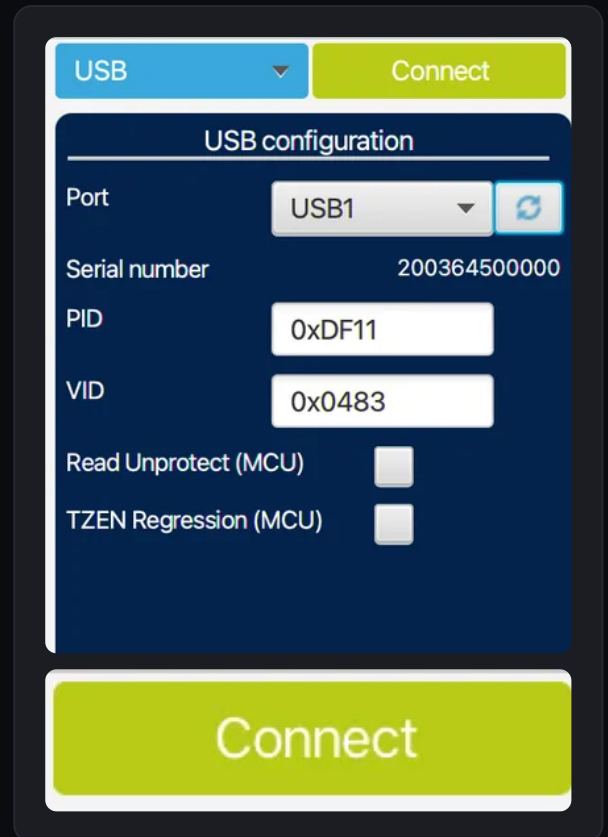
3 A · Hold BOOT, then plug in USB-C

Keep **BOOT** pressed while you plug the cable into your computer, then let go. The board is in DFU mode and shows up as **DFU in FS Mode**.



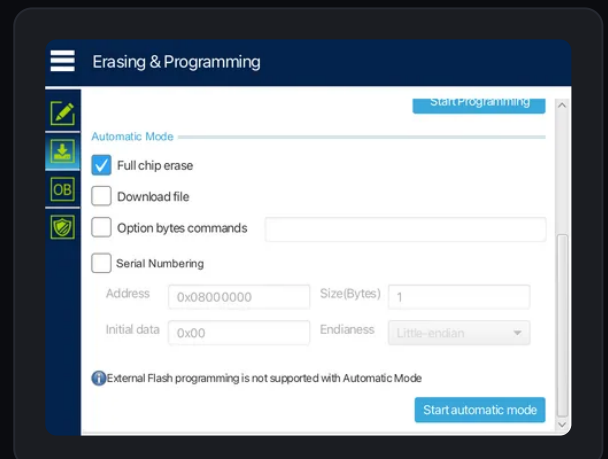
4 A · Connect in STM32CubeProgrammer

In the panel on the right choose **USB**, press refresh until a port appears, then press **Connect**.



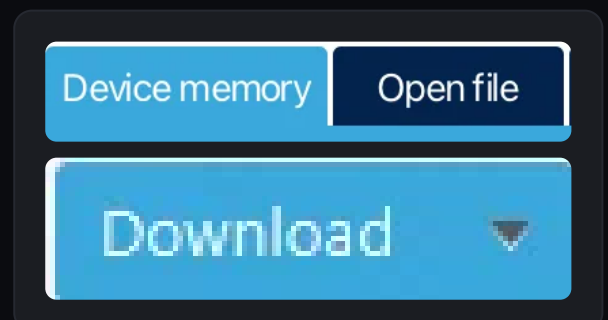
5 A · Erase the whole chip

Open **Erasing & Programming** and press **Full chip erase**. This removes the old firmware and all saved settings.



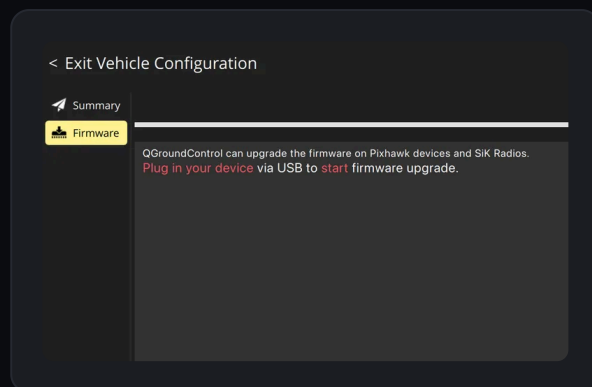
6 A · Write the bootloader

Browse to `darkmatter_brahma_h7_bootloader.bin`, set the start address to `0x08000000` and press **Start Programming**. Wait for "File download complete", then **Disconnect** and unplug the board.



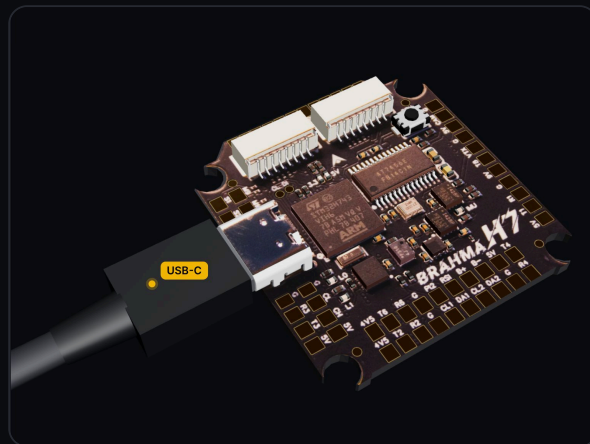
7 B · Open the firmware page first

Start QGroundControl, click the **Q** icon at the top left, choose **Vehicle Configuration**, then **Firmware**. Keep the board unplugged until this page is open.



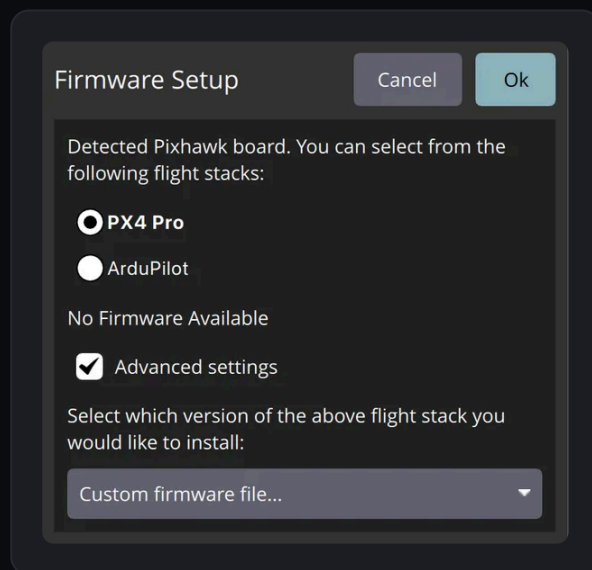
8 B · Plug in USB-C

Plug the board in normally, without BOOT. QGroundControl detects it and opens the firmware options.



9 B · Choose a custom firmware file

Select **PX4 Pro**, tick **Advanced settings**, pick **Custom firmware file...** and press **OK**.



10 B · Select the Darkmatter firmware

Choose `darkmatter_brahma_h7_default.px4`. QGroundControl erases, programs and verifies the board, then restarts it. Do not unplug until it says the upgrade is complete.

GOOD TO KNOW

Only Darkmatter files fit: the Brahma H7 bootloader accepts firmware built for this board. If QGroundControl says the firmware is not suitable for this board, you picked the wrong file.

 `darkmatter_brahma_h7.px4`

11 C · Confirm the version

In QGroundControl open **Analyze Tools** → **MAVLink Console** and type `ver all`. You should see `HW arch: DARKMATTER_BRAHMA_H7` and `PX4 version: Release 1.16.1`.

Summary

Firmware

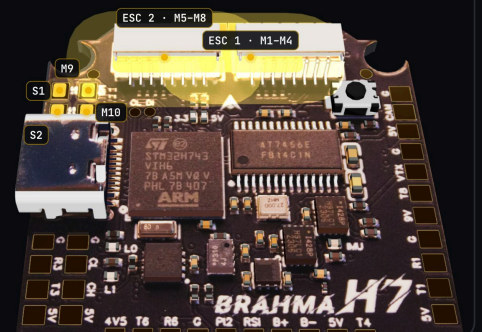
```
QGroundControl can upgrade the firmware on Pixhawk devices and SIK Radios.
Plug in your device via USB to start firmware upgrade.
Firmware: PX4 Durendav1
Connected to bootloader:
Version: 5
Board ID: 6666
Flash size: 1966080
Downloading firmware.
From: /Users/jankhattacharya/Documents/darkmatter_brahma_h7.px4
Download complete
MAV_AUTOPILOT = 12
Successfully decompressed parameter_xml
Successfully decompressed airframe_xml
Successfully decompressed image
Erasing previous program.
```

12 C · Calibrate

The chip erase cleared all settings. In **Vehicle Configuration** set the airframe, then calibrate the **Sensors** (gyro, accelerometer, level horizon, compass) and the **Radio**.

13 Outputs

The board starts as a generic quad X with DShot600 on motors 1–4. In the **Actuators** tab, outputs 1–10 are the motors (ESC 1, ESC 2, M9, M10) and outputs 11–12 are the servo pads S1 and S2.



Serial ports in PX4

PX4 port	UART	Pads
GPS 1	UART2	T2 / R2
TELEM 1	UART3	T3 / R3
TELEM 2	UART4	T4 / R4
RC input	UART6	T6 / R6
TELEM 3	UART7	T7 / R7 (RTS/CTS)
TELEM 4	UART8	T8

A CRSF receiver on UART6 is detected on its own. The PX4 port names differ from ArduPilot's: in ArduPilot, Telemetry 1 is UART7.

If something does not work

No DFU device in STM32CubeProgrammer	Hold BOOT a little longer while plugging in. Try another USB-C data cable or USB port.
QGroundControl does not detect the board	Open the Firmware page before plugging in. Unplug and plug the board back in.
"Firmware not suitable for this board"	Use darkmatter_brahma_h7_default.px4. Firmware for other boards is rejected.
"Missing FMU SD Card" warning	Normal without a microSD card. Insert a FAT32 card to record flight logs.
"Accel / Gyro / Compass uncalibrated"	Expected after flashing. Do the calibration in part C.
Board stuck after a failed update	Repeat part A. The BOOT button always gets you back to DFU mode.

Custom builds

Building PX4 yourself? Extract the board pack, put the `darkmatter` folder inside the PX4 `boards/` directory, then build `darkmatter_brahma_h7_default`. See the [PX4 developer guide](#).

darkmatter.zip

PX4 board pack (for custom builds)

07 Connect a receiver to the Brahma H7

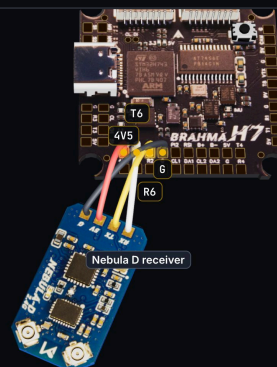
Your receiver goes on UART6. Here it is wire by wire.

1 ExpressLRS (Nebula D)

Four wires to UART6: receiver **TX** → **R6**, receiver **RX** → **T6**, **5V** → **4V5**, **G** → **G**. The 4V5 pad is live on USB, so you can bind and set up on the bench without a battery.

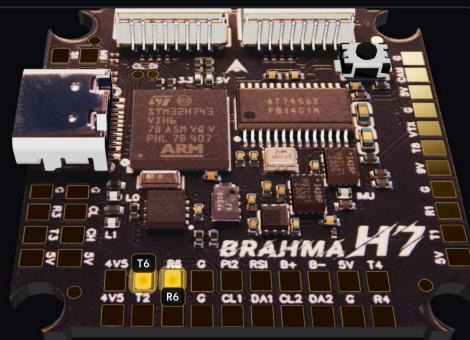
TIP

Receiver and transmitter must run the same ExpressLRS major version (3.x with 3.x). Set the same binding phrase on both, and the receiver binds on its own.



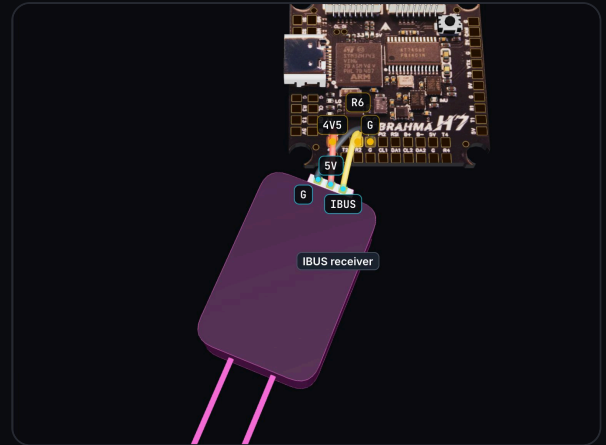
2 Betaflight / INAV

Betaflight: **Ports** tab → **Serial Rx** on UART6, receiver protocol CRSF. INAV: the setup wizard, receiver port UART6, CRSF.



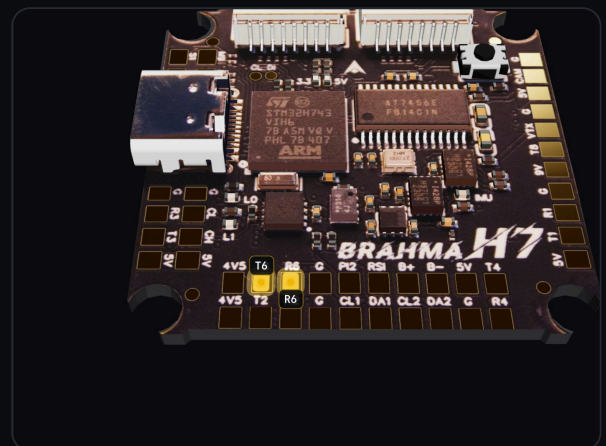
3 FlySky IBUS

Three wires: IBUS signal → **R6**, 5V → **4V5**, G → **G**. In Betaflight the firmware must be built with IBUS selected: reflash with IBUS, then turn on Serial Rx on UART6.



4 ArduPilot

SERIAL7_PROTOCOL = 23 (RC input). This is already the default.



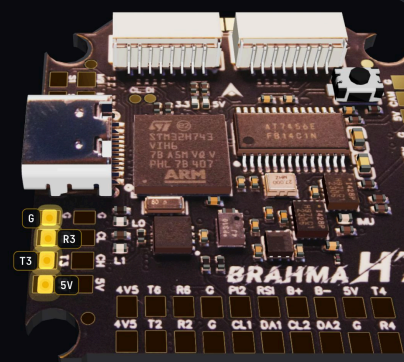
08 Connect a telemetry radio to the Brahma H7

A telemetry radio links the drone to your ground station: live map, battery, parameters and missions in the air. On the Brahma H7 it goes on **UART3**, with MAVLink 2. Shown here with the Darkmatter telemetry air unit (5V, G, TX, RX lead); any MAVLink radio wires the same way.

You need: a MAVLink telemetry radio pair, [Mission Planner](#) or [QGroundControl](#)

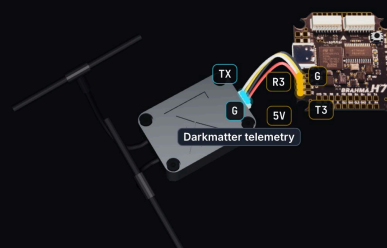
1 Find UART3

Left edge, one column: **G, R3, T3, 5V**.



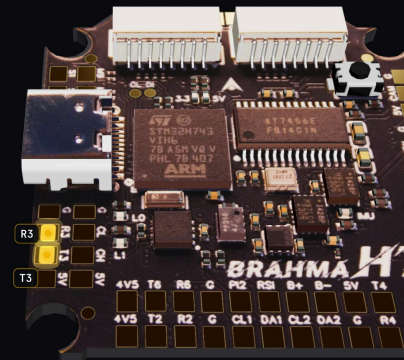
2 Wiring

Radio **TX** → **R3**, radio **RX** → **T3**, **5V** → **5V**, **GND** → **G**. TX always goes to RX.



3 Baud rate

Use the baud rate the radio's maker gives for its serial port. **57600** is the safe start: it is the default of most SiK radios (RFD900, Holybro, mRo). The air radio and the ground radio must also be set to the same air speed and Net ID by the radio's own tool.



4 ArduPilot

UART3 is **SERIAL4** on the Brahma H7 (GPS 2 by default), so set it to MAVLink 2:

```
SERIAL4_PROTOCOL = 2      (MAVLink 2)
SERIAL4_BAUD = 57         (57600)
```

5 PX4

UART3 is **TELEM 1** in PX4 and already runs MAVLink at 57600 (`MAV_0_CONFIG = TELEM 1` , `SER_TEL1_BAUD = 57600`). Change `SER_TEL1_BAUD` if your radio uses another rate.

6 INAV

Ports tab: on UART3 turn on **MAVLink** telemetry at 57600.

7 Ground station

Plug the ground radio into your computer (the Darkmatter ground unit connects by USB-C), open Mission Planner or QGroundControl, pick its COM port at **57600** and connect. The parameters download over the air.

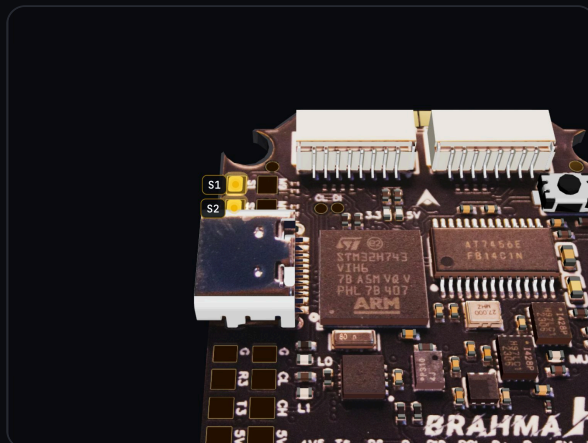
09 Connect servos to the Brahma H7

The Brahma H7 has two servo pads, **S1** and **S2**: outputs 11 and 12, on their own timer, so they can run at servo rate (50 Hz) while the motors run DShot (in PX4 and INAV, see step 5).

You need: [Mission Planner](#) or another ground station

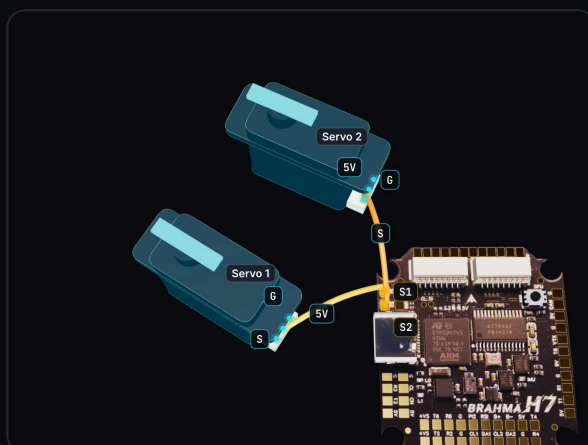
1 Find S1 and S2

Top left of the board, next to M9 and M10.



2 Signal wires

Servo 1 signal (usually yellow, white or orange) → **S1**,
servo 2 signal → **S2**.

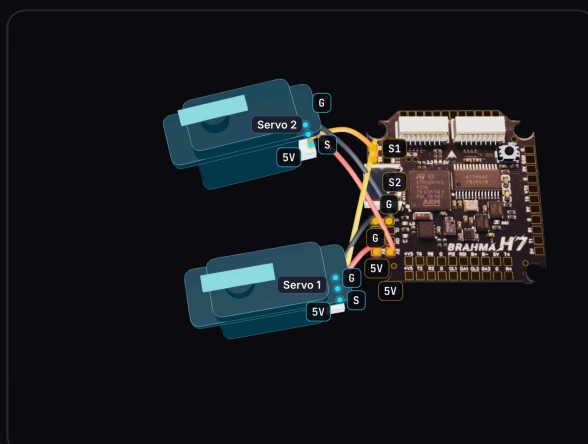


3 Power and ground

Small servos can take power from the board's **5V** and **G** pads next to them (3A shared by everything on 5V).
Bigger or several servos need their own 5–6V BEC: join its ground to the board's G.

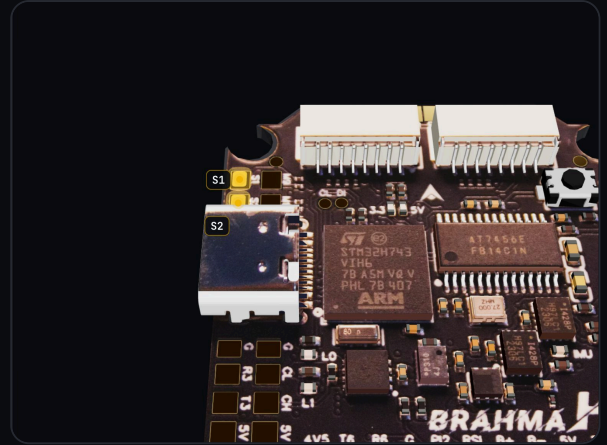
IMPORTANT

A stalled servo can pull more than an amp. If the 5V rail dips, the flight controller resets in the air.
When in doubt, use a separate BEC.



4 ArduPilot: move it from your radio

Mission Planner → **Setup** → **Servo Output**. On output **11** (S1) set **Function** to the RC channel of the switch or knob that should move it, for example **RCIN9**; do the same for output **12** (S2). Set **Min**, **Trim** and **Max** on the same row so the servo stops at its ends without straining, and tick **Reverse** if it moves the wrong way. Move the switch: the bar for that output follows.



5 INAV and PX4

PX4: in **Actuators**, set outputs 11 and 12 to the function you want, and set the timer group for outputs 11-12 (S1/S2) to **PWM 50 Hz** before connecting a servo.

INAV: **Mixer** tab, add your servo rules. Then in the **Outputs** tab set the timer of outputs 11-12 (S1/S2) to **Servos** and the motor timers to **Motors**; otherwise INAV puts servo 1 on M5.

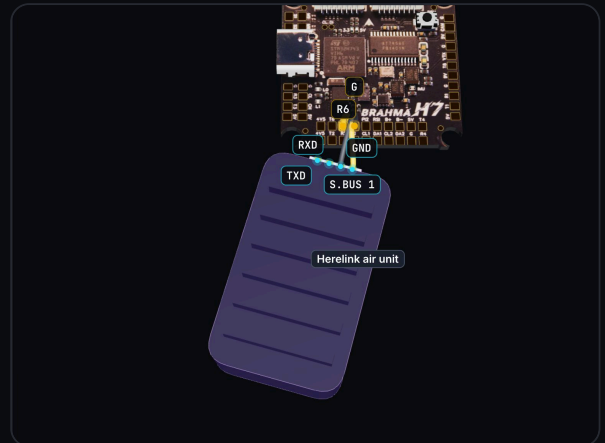
10 Connect Herelink to the Brahma H7

Herelink gives you control (SBUS) and telemetry (MAVLink) over one link. ArduPilot.

You need: [Mission Planner](#)

1 SBUS

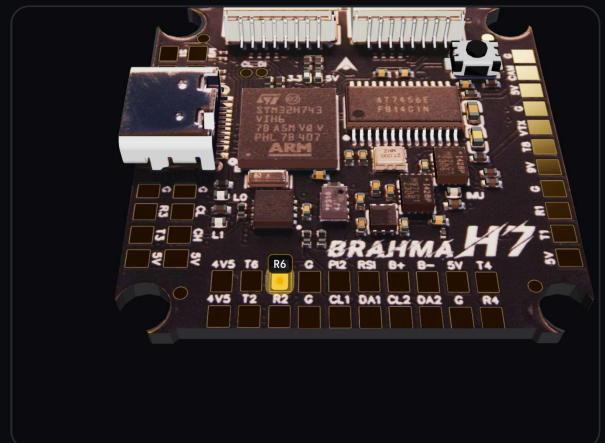
Herelink **S.Bus out 1** → **R6**, Herelink GND → **G**.



2 SBUS parameters

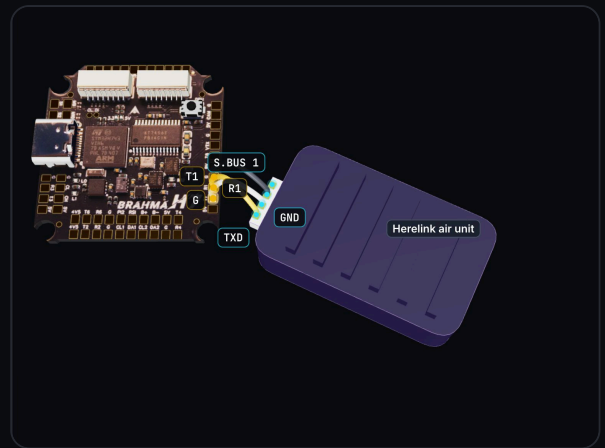
In Mission Planner, Full Parameter List:

```
SERIAL7_PROTOCOL = 23    (RC input)
SERIAL7_OPTIONS = 3      (invert RX and TX:
SBUS is an inverted signal)
```



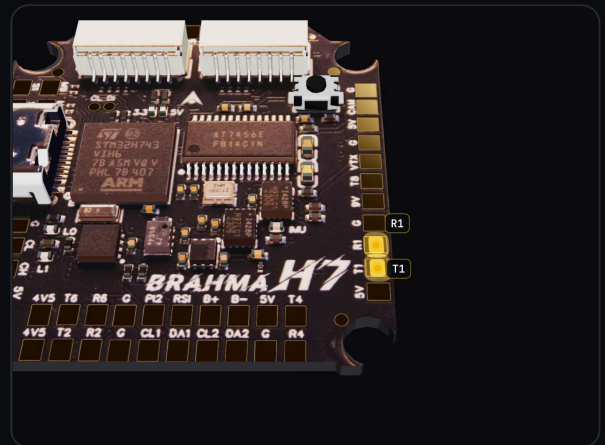
3 Telemetry

Herelink UART RXD → T1, TXD → R1.



4 Telemetry parameter

```
SERIAL2_PROTOCOL = 2    (MAVLink 2)
```



11 Connect a Raspberry Pi to the Brahma H7

A companion computer talks MAVLink to ArduPilot over UART4. Any Raspberry Pi works.

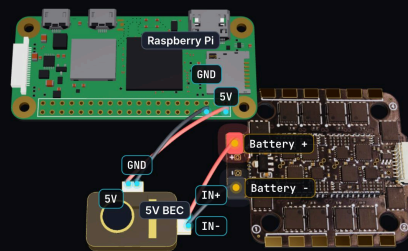
You need: [Mission Planner](#)

1 Power the Pi

Use a separate **5V BEC** rated for your battery (2–6S, so up to 25.2V in) and at least **3A** out (**5A** for a Raspberry Pi 5). **BEC IN+ → ESC BAT+**, **BEC IN- → ESC BAT-** (or any suitable battery supply), then **BEC 5V → Pi pin 2 (5V)** and **BEC GND → Pi pin 6 (GND)**.

TIP

Check the BEC output with a multimeter before you connect the Pi: it must read 5.0–5.2V. In the next step a second Pi ground (pin 14) goes to the flight controller's G.

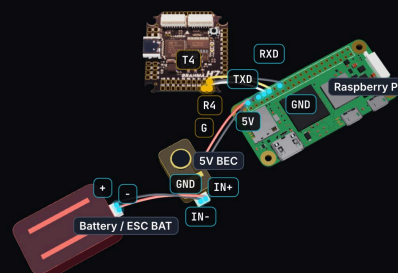


2 Wiring

Brahma **T4** → **Pi pin 10 (RXD, GPIO15)**, **R4** → **Pi pin 8 (TXD, GPIO14)**, **G** → **Pi pin 14 (GND)**. The BEC feeds **pin 2 (5V)** and **pin 6 (GND)**; its input (IN+ / IN-) goes to the ESC battery pads or any suitable battery supply.

IMPORTANT

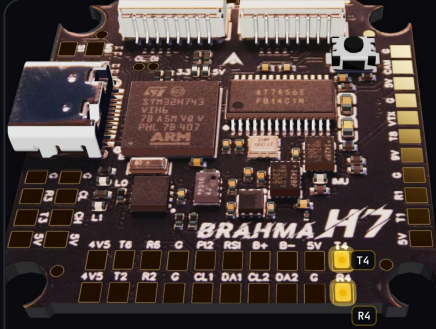
Do not power the Pi from the flight controller. A Pi draws current in bursts and the board can restart. Give the Pi its own BEC with a stable 5V.



3

UART4 is SERIAL6 on the Brahma H7. Mission Planner
→ Full Parameter List:

```
SERIAL6_PROTOCOL = 2      (MAVLink 2)
SERIAL6_BAUD = 921        (921600)
```



4

On the Raspberry Pi:

```
sudo raspi-config
# Interface Options → Serial Port
# "Login shell over serial?" → No
# "Enable serial hardware?" → Yes
```

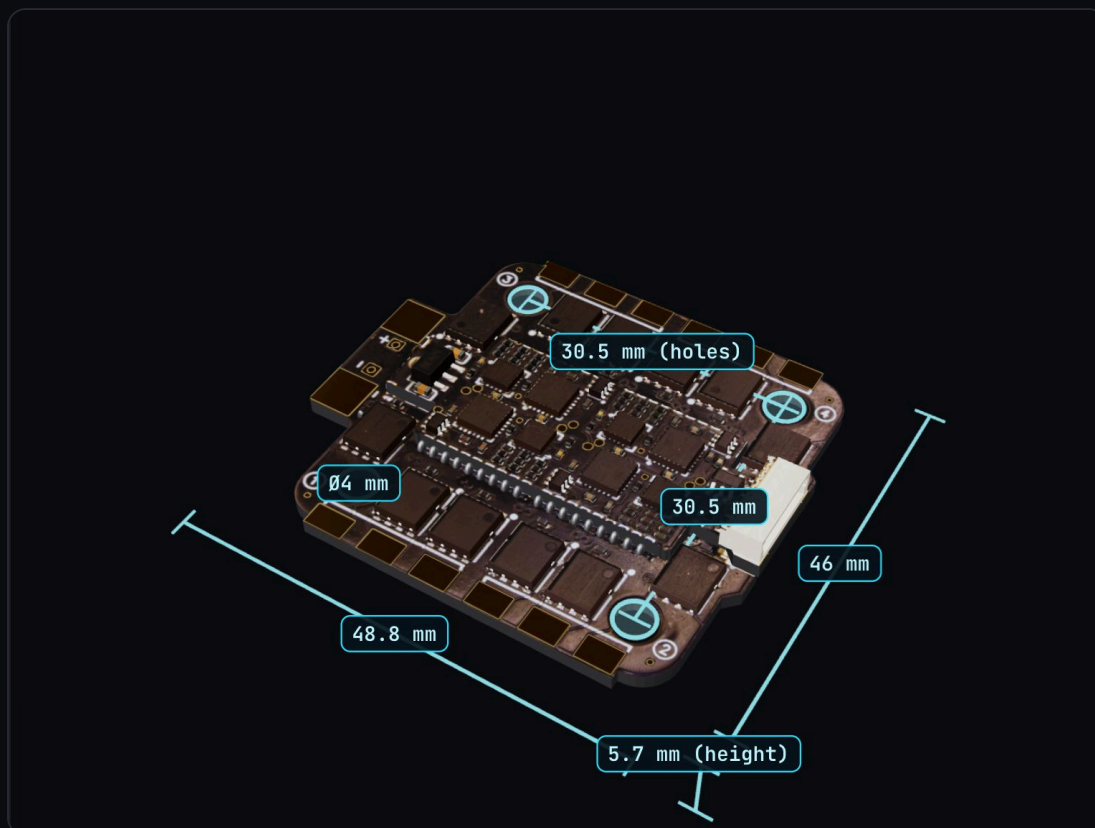
12 Vishnu 60 4-in-1 ESC

A 4-in-1 ESC for 2–6S builds: 60A continuous per motor, 90A burst, ships with BLHeli_S (BlueJay ready), DShot300/600 and a built-in current sensor, on a 30.5 × 30.5 mm mount.

Specifications

Product	Darkmatter® Vishnu 60 4-in-1 ESC
Firmware	Ships with BLHeli_S. For RPM filtering (bidirectional DShot), flash BlueJay in ESC Configurator: target J-H-50
Continuous current	60A × 4 = 240A
Burst current	90A for 7 seconds
Protocol	DSHOT300 / DSHOT600
Power input	2–6S LiPo
Power output	VBAT
Current sensor	Built in
Mounting	30.5 × 30.5 mm, 4 mm holes
Dimensions	46 × 48.8 × 5.7 mm
Weight	14 g
Motors	Check the maximum current in your motor maker's data: it should stay below 70–80A. If you will fly at full throttle most of the time with the biggest propeller the motor supports, stay at 60A or below.
Operating temperature	–20 °C to 70 °C
Made in	India

Dimensions and mounting



46 × 48.8 mm, 5.7 mm tall, 1.6 mm PCB. Holes 4 mm on the standard 30.5 × 30.5 mm pattern, so it stacks under any Brahma.

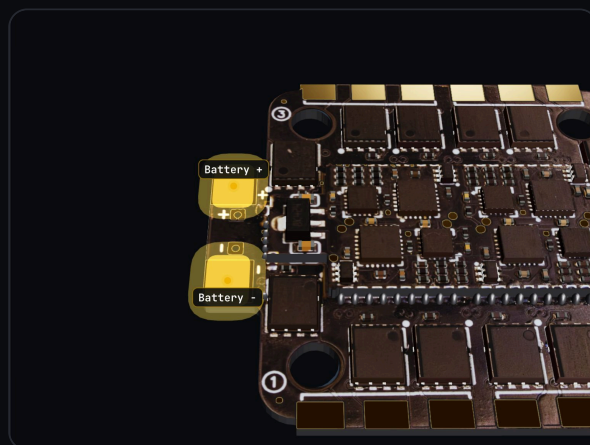
Battery and capacitor

1 Solder the capacitor first

Solder the 1000 μ F low-ESR capacitor from the box across the battery pads: + to +, - to -. Connect the battery - lead to the capacitor's - side only.

IMPORTANT

Without the capacitor, voltage spikes from the motors can destroy the ESC.



2 Battery leads

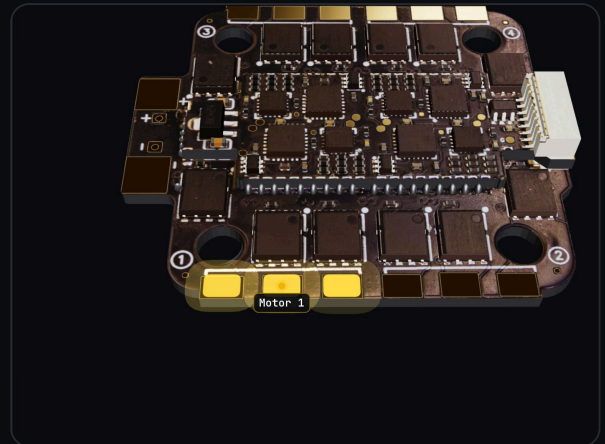
Solder the 12 AWG leads and the XT60 connector from the box to the same pads. Red to +, black to -. Check twice: reversed polarity is not covered by the warranty.

Motor numbering

With the board this side up, motor 1 is bottom left, 2 bottom right, 3 top left and 4 top right, matching the numbers printed on the board. Each motor has three pads; swap any two of a motor's wires to reverse it (or reverse it in BLHeli/BlueJay).

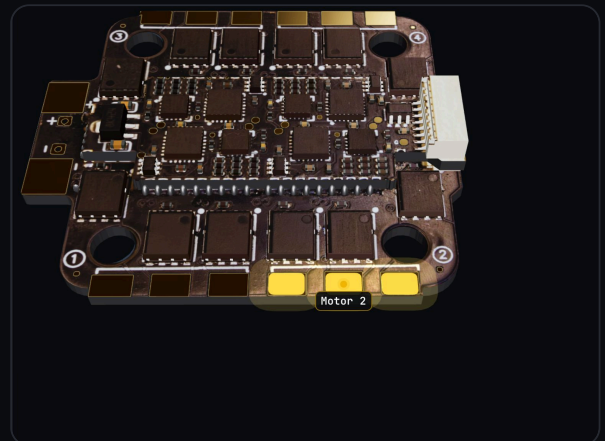
1 Motor 1

The three pads for motor 1.



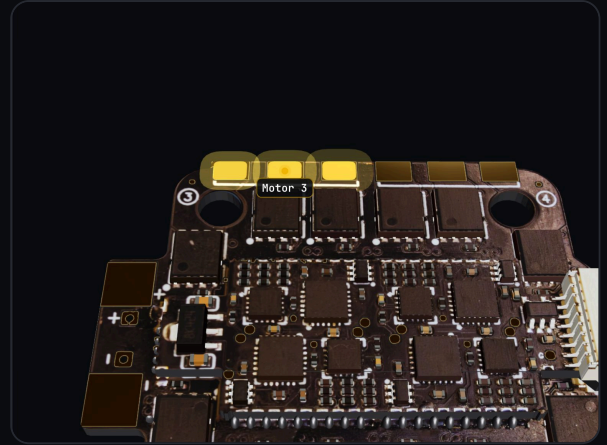
2 Motor 2

The three pads for motor 2.



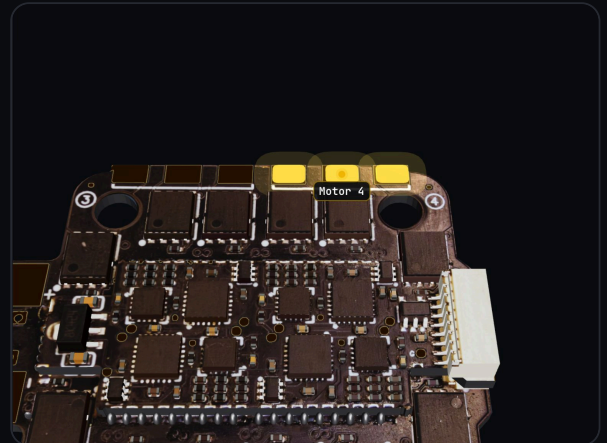
3 Motor 3

The three pads for motor 3.



4 Motor 4

The three pads for motor 4.

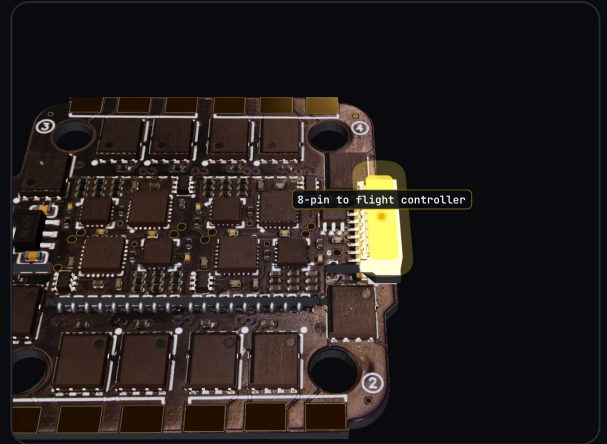


Connect to the flight controller

Use the 8-pin cable from the box. It carries the four motor signals, current, telemetry, ground and battery voltage in one plug, straight into any Brahma. Or solder to the 7 pads on the underside (4, 3, 2, 1, C, G, V).

1 8-pin plug

Plug one end here and the other into the flight controller's ESC port.



2 Direct-solder pads

On the underside: 4, 3, 2, 1 (motor signals), C (current), G, V (battery voltage).



Stacks: [Stack H760](#), [Stack 760](#), [Stack 460](#).

Before the first flight

- **First power-up through a smoke stopper** (or a current-limited bench supply). A solder bridge then costs you a fuse, not an ESC.
- **Bidirectional DShot:** with BlueJay on the ESC, use DShot300 and turn on bidirectional DShot in Betaflight (Configuration tab, or `set dshot_bidir = 0N`) and set your motor pole count (`set motor_poles = 14` for most 5-inch motors). That enables the RPM filter, the biggest single improvement to a tune. Stock BLHeli_S does not support it.
- **Motor order:** check it in the Motors tab with the props off. The flight controller expects its own motor order; if a motor spins in the wrong place, remap it rather than swapping wires around.

IMPORTANT

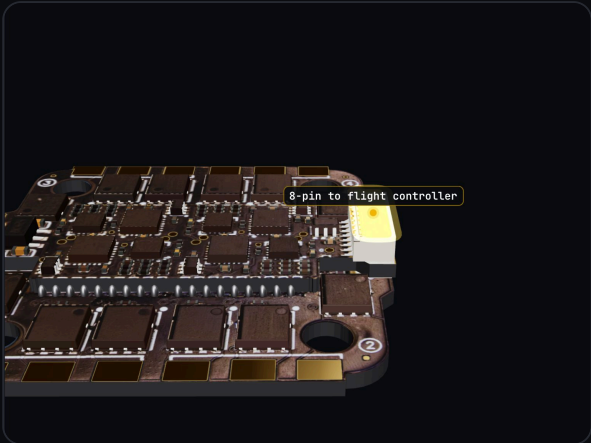
Test motor direction **without propellers** and keep it below 15% throttle. Never arm without propellers in Angle mode: the flight controller tries to level the craft and the motors will not slow down when you lower the throttle.

BlueJay and PWM frequency

The Vishnu 60 ships with BLHeli_S. For the RPM filter (bidirectional DShot) you need BlueJay.

1 Flash BlueJay

For RPM filtering (bidirectional DShot), flash BlueJay in the [ESC Configurator](#): target **J-H-50**.



2 Set PWM by prop size

After flashing BlueJay, set PWM by prop size:

Props	PWM frequency
Larger than 5"	24 kHz
5"	48 kHz
Smaller than 5"	96 kHz

A lower PWM frequency gives the motors firmer braking and keeps switching heat in the ESC down, which suits bigger props; it does not add thrust. A higher frequency runs the motors smoother and more efficiently at part throttle, which helps small motors most. If motors stutter, desync or will not start, flash again one step lower.

In the box

- 1 × Vishnu 60 4-in-1 ESC
- 1 × 1000 µF 35V low-ESR capacitor
- 12 AWG battery wire
- 1 × XT60 connector
- 1 × SH 1.0 mm 8-pin cable, 30 mm

Downloads

Vishnu 60 manual (PDF)

The original slide manual

Warranty

The Vishnu 60 ESC has a 6-month warranty against manufacturing defects from the date of purchase. Keep your invoice: we need a copy for any claim.

Not covered:

- Physical damage from use, such as water damage or tampered components.
- A burnt circuit from wrong wiring or overloading.
- A burnt ESC from spinning motors without propellers above 15% throttle while checking motor direction.
- A burnt ESC from arming without propellers in Angle mode: the flight controller keeps trying to level the craft and the motors will not slow down when you lower the throttle.

How a claim works: we inspect the board for physical damage and the points above. If it is eligible we replace it at no cost, and help you with your build. WhatsApp or call [+91 80171 93277](tel:+918017193277).