3DFENSE

D-20C Personal Eagle II — User Guide

Fully 3D-printable personal VTOL · Build, setup & flight manual

3DFENSE · D-20C Personal Eagle II · Revision 0.9 DRAFT · Wingspan 878 mm / 34.6 in



Introduction

The **D-20C Personal Eagle II** is a fully 3D-printable fixed-wing hybrid **VTOL** aircraft — the civilian variant of our D-20 platform. Make no mistake: this is **a working tool, not a toy** — a professional instrument that carries an **850 g payload over long distances** (up to 75 km one way, 70 minutes in the air). Two tilting front rotors and a third rotor inside the ring tail lift the aircraft straight up from a backyard, forest clearing or boat deck; then the front gondolas tilt forward and the D-20C flies like an efficient airplane.

It is designed around the **3DFENSE NavPack D04** — a single board carrying the autopilot, GNSS, magnetometer, pitot-static airspeed sensing and all power electronics. Together they make VTOL flying genuinely simple:

- **Autolaunch** — place the aircraft on the ground, plug in the battery, press the START button on the airframe. The aircraft takes off, flies its mission and lands back — no laptop, no RC transmitter required.

- **No pilot skills needed** — automated vertical takeoff and landing removes the highest-risk flight phases entirely.
- **100% 3D-printed airframe** — no carbon spars, no metal fasteners. The whole aircraft is assembled with CA glue in **8 bonding operations, under 30 minutes** (when printed on a Prusa XL; smaller printers split the parts into more segments, adding a few joints).
- **Tool-less field setup** — the wings detach for transport; from backpack to air in under 2 minutes.
- **Companion-computer ready** — a Raspberry Pi (or any SBC with the Pi 40-pin connector) plugs straight onto the NavPack: its own high-current power rail, switched on/off by the autopilot, MAVLink already wired. No cables, no extra BEC.

This is the easiest path to a VTOL aircraft there is — and a surprisingly affordable one. Price the parts separately — autopilot, GNSS, professional magnetometer, digital airspeed sensor, power monitor, three regulators, distribution board, plus all the cables — and the NavPack costs less than the shopping list. And instead of a box of modules you get one tested, calibrated, high-quality board, pre-flashed with the complete D-20C setup. No soldering, no wiring diagrams, no weeks of parameter tuning: you save money, time and a lot of headaches — and go flying.

> **Coming soon — printable transport backpack.** The D-20 printed backpack is being adapted to the civilian D-20C and will be added to your downloads as a **free update**: the whole aircraft, wings off, carried on your back — and literally from backpack to air in under 2 minutes.



Printable transport backpack — being adapted for the D-20C, coming to your downloads as a free update

NavPack vs. a classic DIY stack

What it takes to match the NavPack with off-the-shelf modules of comparable sensor quality (street prices, USD):

Function	DIY stack (e.g. Matek)	~Price	NavPack D04
Autopilot (STM32H743)	Matek H743-WING V3	$130	on board
GNSS + RM3100 professional compass	Matek M10-L4-3100	$89	on board
Digital airspeed + barometer (Amphenol AUAV combo)	Matek ASPD-AUAV (airspeed only)	$110	on board — AUAV: airspeed + baro, plus a second independent baro (dual-baro redundancy)
Precision power monitor	Holybro PM02D	$30	on board
Extra BEC (servo rail)	assorted	$10	3× 8 A regulators on board
Raspberry Pi / companion hookup — 5 V BEC + MAVLink wiring	assorted	$25	2×20 Pi header on board: plug in, autopilot-switched power (RELAY 1), MAVLink wired
Power distribution — 3× ESC, up to 12S	PDB + wiring	$15	on board (2× XT60, 5× XT30)
Connectors, cables, heat-shrink	—	$20	not needed
Total		**~$429**	**$395 — incl. firmware & calibration**

And the part no price tag shows — the work:

	NavPack D04	Classic DIY stack
Soldering	none	hours — ESC power leads, XT60, BECs, PDB
Wiring	plug in motors & servos	5–6 modules over UART/I2C/CAN, crimping
Firmware	golden image pre-flashed	flash & configure from scratch
Aircraft parameters	complete D-20C setup pre-loaded	universal param file + board-specific part yourself
Sensor calibration	factory-calibrated, test protocol included	all calibrations yourself
Autolaunch button	yes	not available
Battery input	3–12S, 2× XT60 pass-through	typically 2–8S
Companion computer	Pi plugs onto the board — mechanically secured, powered, relay-switched, MAVLink wired	extra BEC, UART soldering, cable harness + you must design and build a mount for the Pi itself
Support	one vendor, one board	several vendors
Time to first flight	~1–2 hours	a weekend or more

Typical civilian missions: aerial photography and video, mapping and surveying, search & support, FPV training, long-range cruising — the quick-swap payload bay accepts a camera gimbal, extender battery or cargo.

Safety first

- This is **not a toy**. A 3.3 kg aircraft with spinning propellers can cause serious injury.
- **Props clear!** Never approach the aircraft while armed. The front propellers tilt and the rear propeller runs inside the ring tail — treat all three as live at all times.
- Always remove propellers (or disconnect the battery) during bench setup, calibration and firmware work.
- Fly only in permitted airspace and follow your national UAS regulations (EU: EASA Open category rules, operator registration and Remote ID where applicable).
- Check the wind: maximum 8 m/s for VTOL phases, 15 m/s in fixed-wing flight.
- Li-ion packs: charge with a balance charger, never below 0 °C, store at ~3.7 V/cell.

Specifications

Parameter	Value
Type	Fixed-wing hybrid VTOL (tiltrotor 2+1)
Wingspan	878 mm
Length	896 mm
Height	260 mm
Wing area	18.2 dm²
Max takeoff weight (MTOW)	3350 g
Payload capacity	850 g
Stall speed	16 m/s (58 km/h)
Cruise speed	22 m/s (79 km/h)
Maximum speed	40 m/s (144 km/h)
Endurance — fixed wing	70 min (90 min with extender battery in payload bay)
Endurance — VTOL hover	25 min
Range	up to 75 km one-way ISR-profile mission
Wind resistance	15 m/s fixed-wing · 8 m/s VTOL
Operating temperature	–20 °C to +40 °C
Battery	6S2P Li-ion 21700, ~10 000 mAh
Airframe materials	PolyLight 1.1 · FlexiLight 1.0 · PET-G · PolyAir 1.0/2.0 / PLA+
Bonded joints / assembly time	8 joints, CA glue, < 30 min (XL parts; more segments on smaller printers)
Field setup (tool-less)	< 2 min



Top view — tiltrotor 2+1 layout, rear motor inside the ring tail



Side view



Front view — forward-flight configuration, camera gimbal under the fuselage

What you need

Electronics — equipment pack

Item	Qty	Spec
Front motor (tilt gondola)	2	2812, 900 KV
Front propeller	2	Aeronaut CAMcarbon Light 10×5", 1× CW + 1× CCW — or same-size alternative (e.g. APC 10×5 E + EP)
Tilt servo	2	in front gondolas
Rear motor (ring tail)	1	3110, 1200 KV (also proven: 2812, 1115 KV)
Rear propeller	1	Aeronaut CAMcarbon Light 8×4.5" — or same-size alternative (e.g. APC 8×4.5)
ESC	3	50 A, 6S, BDShot capable
Control-surface servo	3	2× aileron + 1× elevator
Camera tilt servo	1	optional, front camera tilt (NavPack output 1, RC13)
Gimbal holder servo	1	optional, swings the gimbal holder in/out of the fuselage (NavPack output 2)
Flight controller	1	**3DFENSE NavPack D04** (autopilot + GNSS + mag + airspeed + power)
Battery	1	6S2P Li-ion 21700, ~10 000 mAh, XT60
RC system	1	mLRS / ELRS / CRSF — recommended: Matek MR900-30 on the NavPack + MR900-30-TX in the transmitter

Motor brand does not matter — any quality motor of the given size and KV works. For the rear motor we have also successfully flown a **2812, 1115 KV**; avoid going to a lower KV, or the rear rotor loses thrust margin.

Tools & consumables

- CA glue (medium) + activator — the entire airframe is bonded with CA
- Hobby knife, side cutters, hex drivers
- Balance charger for 6S Li-ion
- USB-C data cable (NavPack flashing & setup)

3D-printed parts from the NavPack set

The NavPack D04 download package includes its own printed parts — **pitot-static tube** (non-foaming harder TPU, tested Shore 55D), **NavPack mount** (PET-G) and **Autolaunch START button** (PLA). Print them along with the airframe.

3D printing

Materials

Material	Used for	Notes
PolyLight 1.1 (active-foaming LW-PLA)	wings, fuselage shells, tail	expands up to 2.8× — 50%+ weight saving; print with 3MF profiles only
FlexiLight 1.0 (foaming flex)	nose cone, covers, bumpers, battery holder	energy absorption for landings
PET-G	motor mounts, prop spacers	heat & vibration resistance
PolyAir 1.0/2.0 / PLA+	motor gondolas, servo arms (hatches), holders, payload bay, setting tools	stiff, precise fits; sliced with the PLA print profiles

> **Important — a big change from classic 3DLabPrint airplanes:** the D-20C is **not printed in vase/spiral mode**. Where most 3DLabPrint designs print LW parts with a single perimeter and no infill, the D-20C parts print **conventionally — mostly with two perimeters and infill**. The airframe needs the extra strength, and with two perimeters the structure also handles sun exposure well. Do not carry over print settings from the older airplanes. One side effect compared to vase mode: expect **heavy stringing** on the LW parts — travel moves with near-zero retraction leave foamed strings behind. This is normal and harmless; the strings are brittle and clean off in seconds with a hobby knife.

> Use the supplied **3MF print projects** — all critical settings (foaming flow, walls, seams, orientation) are pre-configured. Ready-made G-code is provided for the **Prusa MK4S**; the `XL` folder contains parts for the **Prusa XL** — larger one-piece sections with fewer joints (the renders in this guide show the XL versions). On a smaller printer the parts are simply split into more segments, so you get a few extra CA-glued joints. Do not slice LW parts with standard PLA profiles.

Slicer settings inside the 3MF projects

You don't need to set any of this — it is already in the 3MF files. The table shows the key values so you understand *why* the profiles differ, and so you can port them to another printer or filament brand:

Parameter	Fuselage 2 & booms (PolyLight)	Wings, stabs, ailerons (PolyLight)	Nose, covers, bumpers (FlexiLight)	Gondolas, arms, holders, tools (PolyAir/PLA+)	Motor mounts (PET-G)
Base profile	0.20 SPEED	0.20 SPEED	0.20 SPEED	0.20/0.15 STRUCTURAL	0.20 STRUCTURAL
Layer height	0.3 mm (first 0.2)	0.3 mm (first 0.2)	0.3 mm (first 0.2)	0.2 mm (holder 0.15)	0.2 mm
Perimeters	2	**1**	2	2–3 (payload holder 6)	**10**
Top / bottom layers	2 / 2	0 / 1–2	2 / 2	3–5	10 / 10
Infill	3 % cubic, 200 % wide lines	**0 %** — structure is modeled in	10 % rectilinear (bumpers 20 % gyroid)	5–15 % grid	30 % gyroid
Flow (extr. multiplier)	**0.65**	**0.60**	**0.50**	1.0	1.0
Nozzle temp	255 °C (full foaming)	255 °C	250–255 °C	225–230 °C	240 °C
Bed temp	63 °C	60–62 °C	**25 °C**	60–64 °C	85 °C
Speed	90 mm/s constant	perimeter 40 mm/s	40 mm/s	45–80 mm/s	45–80 mm/s
Fan	20–50 %	20–50 %	30–90 %	70–100 %, always on	20–40 %, always on
Retraction	~none (0.01 mm)	~none (0.01 mm)	~none (0.01 mm)	0.7 mm	0.7 mm
Supports	off	off (brim 2–4 mm on stabs/ailerons)	off	**on**	**on**

What the numbers mean:

- **Foaming flow** — at 255 °C PolyLight expands to roughly 2.5× its volume, so the profiles push only 0.60–0.65 of the normal material flow (FlexiLight 0.50). Printing LW parts at flow 1.0 produces heavy, over-extruded parts.
- **Constant speed on foamed parts** — foaming is speed- and temperature-dependent, so the fuselage profile runs everything at a flat 90 mm/s and the wing perimeter at a flat 40 mm/s. Don't "optimize" speeds on LW parts — varying speed means varying wall thickness and surface.

- **Retraction ≈ 0** on all foaming materials — foamed filament cannot be pulled back without jamming; seam positions and travel moves are tuned in the projects instead.
- **Wings and control surfaces** print with a single perimeter and zero infill — the spars and ribs are modeled directly into the part. The two-perimeter fuselage carries the infill instead (3 % cubic with double-width lines).
- **FlexiLight parts** print slow, with a cold bed (25 °C) and gap fill disabled — flex needs no bed adhesion help and over-filled gaps would stiffen the crash zones.
- **PolyAir/PLA+ parts** (gondolas, servo arms, holders, setting tools) are ordinary non-foamed prints: standard Prusa STRUCTURAL profile, full flow, supports on — here dimensional accuracy of the fits matters, not weight.
- **Motor mounts** in PET-G are effectively solid (10 perimeters, 30 % gyroid) — they take motor heat and vibration.

If you substitute another active-foaming LW-PLA, check its maximum-foaming temperature and tune the flow so a single wall comes out solid and gap-free — then keep the rest of the profile as supplied.

Not a Prusa printer? No problem. The 3MF projects are prepared for the Prusa MK4S / XL, but they are not locked to them — with the table above you can carry the setup to any machine:

1. **Open the 3MF in PrusaSlicer or OrcaSlicer** (both free, both drive almost any printer) — you get the parts already oriented, grouped and with seams placed, plus all the print settings loaded.
2. **Switch the printer profile to yours** and keep the print and filament settings from the project. Check that the parts still fit your bed — the MK4S projects need 250 × 210 mm; on a larger machine use the `XL` folder parts instead.
3. **Walk through the table above as a checklist** — flow multiplier, temperatures, perimeters/infill, near-zero retraction and constant speed on the foamed parts are what make the airframe come out right; everything else (accelerations, travel, cooling details) you can leave to your printer's profile.
4. **Using a different slicer** (Cura, Bambu Studio…)? Open the 3MF just for the geometry and orientation, and enter the table values into your material/print profile manually.
5. **Print one small LW part first** (e.g. the ailerons) and check it: walls solid and gap-free, surface smooth, part feather-light. If the walls have gaps, raise the flow slightly or the temperature; if the part is heavy and glossy, lower the flow. Then print the rest.

A direct-drive extruder is strongly recommended — for the FlexiLight parts it is practically a requirement; bowden printers struggle with flex and with the near-zero-retraction foaming profiles.

Parts list & print times

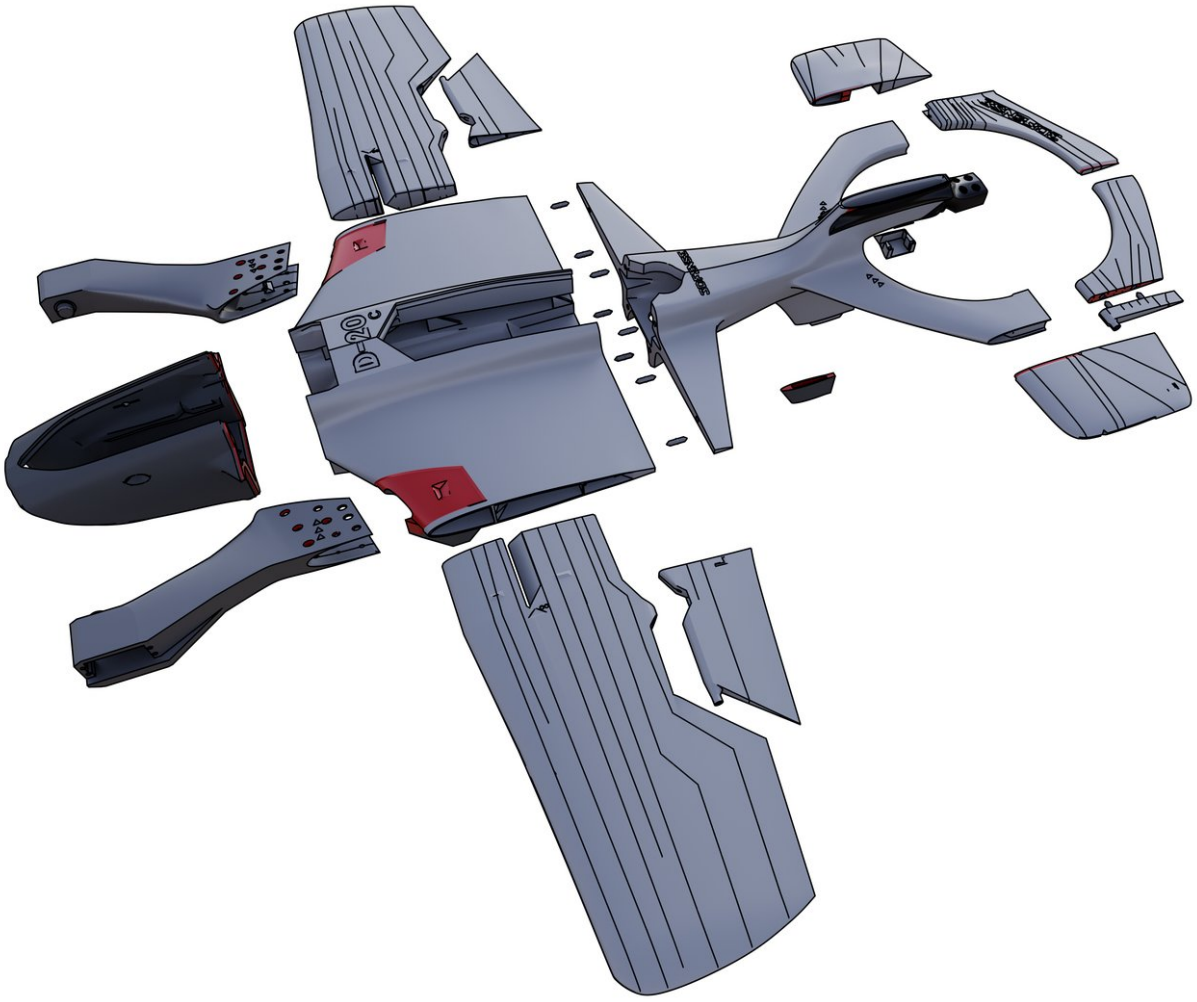
Part	File	Material	Print time (MK4S)
Nose cone (fuselage 1)	`D-20-C_fuselage_1_FLEX`	FlexiLight	7 h 06 m
Fuselage 2A	`D-20_C_Fuselage_2A_LW`	PolyLight	3 h 14 m
Fuselage 2B	`D-20_C_Fuselage_2B_LW`	PolyLight	3 h 00 m
Fuselage 2C	`D-20_C_Fuselage_2C_LW`	PolyLight	3 h 48 m
Fuselage 2D	`D-20_C_Fuselage_2D_LW`	PolyLight	3 h 30 m
Tail boom 3AB	`D-20_C_Fuselage_3AB_LW`	PolyLight	2 h 15 m
Tail section 3C	`D-20_C_Fuselage_3C_LW`	PolyLight	4 h 42 m
Wing left	`D-20-C_wing_L_LW`	PolyLight	5 h 23 m
Wing right	`D-20-C_wing_R_LW`	PolyLight	5 h 23 m
Ailerons	`D-20-C_ailerons_LW`	PolyLight	1 h 08 m
Horizontal stab	`D-20_C_Hstab_LW`	PolyLight	1 h 53 m
Vertical stab	`D-20_C_Vstab_LW`	PolyLight	1 h 07 m
Elevator	`D-20_C_Elevator_LW`	PolyLight	1 h 22 m
Covers	`D-20-C_covers_FLEX`	FlexiLight	5 h 44 m
Bumpers	`D-20-C_bumpers_FLEX`	FlexiLight	2 h 17 m
Motor mounts	`D-20-C_mot_mounts_PETG`	PET-G	5 h 25 m
Motor gondolas + servo arms (hatches)	`D-20-C_gondola_hatch_PLA`	PolyAir 1.0/2.0 / PLA+	8 h 30 m
Payload holder	`D-20-C_payload_holder_PLA`	PolyAir 1.0/2.0 / PLA+	38 m
Prop spacer	`D-20-C_PropSpacer02`	PET-G	3 m

Total printing time: roughly **66 hours** on a single Prusa MK4S.

Optional parts — print them only when you need them, they don't hold the basic airframe back:

Part	File	Material	Print time (MK4S)
Battery + GNSS holder	`D-20-C_batt_holder_gnss_FLEX`	FlexiLight	3 h 38 m
Gimbal holder	`D-20-C_gimb_holder_PLA`	PolyAir 1.0/2.0 / PLA+	2 h 44 m
Setting tools (CG & throw jigs)	`D-20-C_servo setting_PLA`	PolyAir 1.0/2.0 / PLA+	8 h 36 m

The setting tools are not part of the airframe, but **print them before the first setup** — you will need them to set the tilt end-points and control throws (see *Center of gravity & control throws*).



Complete printed parts set — exploded view

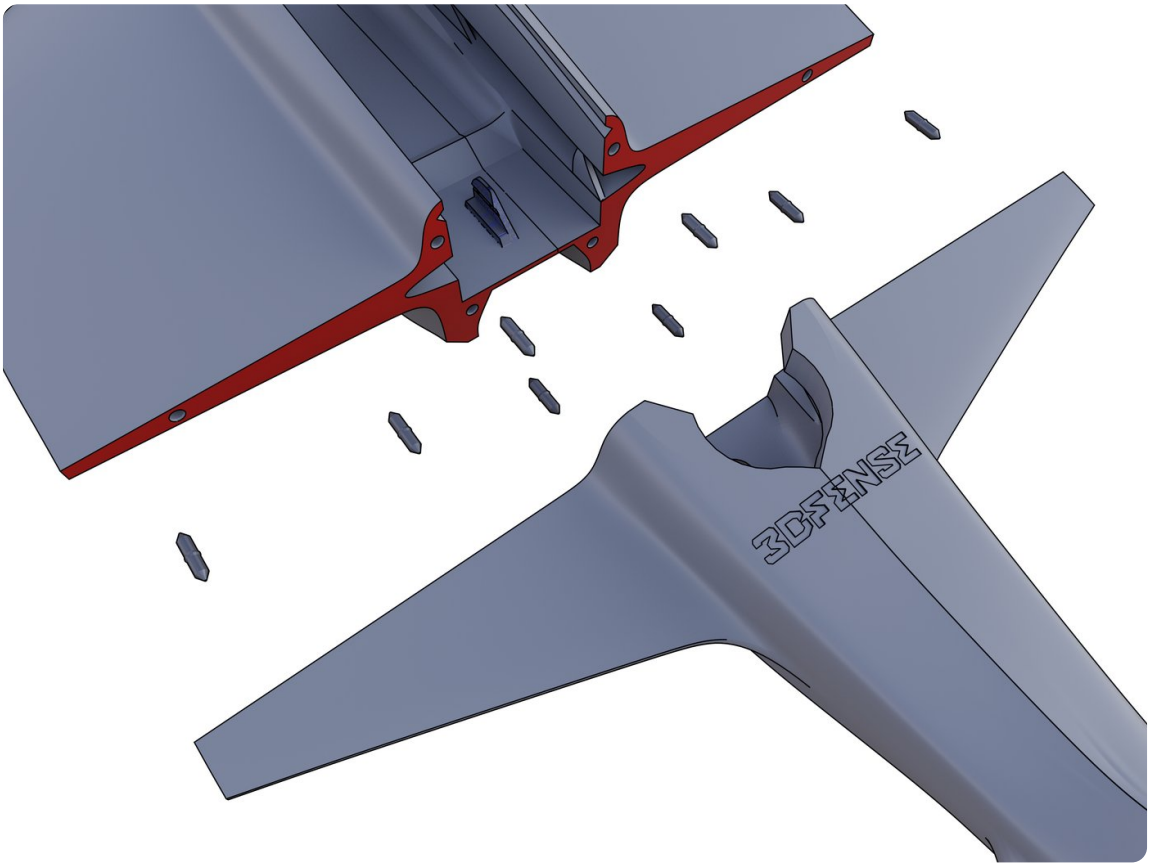
Assembly

The airframe is bonded with **CA glue** in 8 operations — this count applies to the **Prusa XL** parts shown in the figures; on a smaller printer (MK4S) the big sections come split into more segments, so expect a few extra glue joints. **Red surfaces** in the figures mark the glue joints; printed alignment pins keep every joint perfectly straight. Dry-fit every joint first, then apply CA and press together — the foamed PolyLight surface bonds instantly with activator.

> **TODO (Štěpán):** ověřit pořadí kroků a texty u jednotlivých obrázků — popisky níže jsou můj první odhad podle renderů.

1 — Center fuselage

Bond fuselage sections 2A → 2B → 2C → 2D together using the alignment pins.



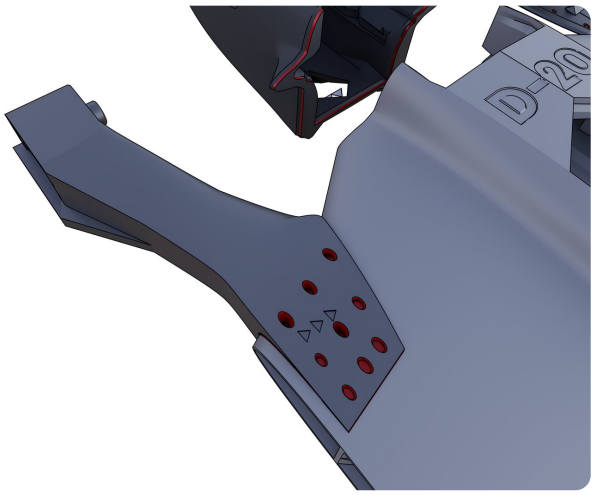
Alignment pins + red glue surfaces on the center-section joint

2 — Front gondola arms

Glue the gondola arms to the reinforced fuselage shoulder (red pad). The pin pattern carries the tilt-motor loads into the wing root.



Gondola arm — glue pad on fuselage



Arm seated on the pin pattern

3 — Nose cone

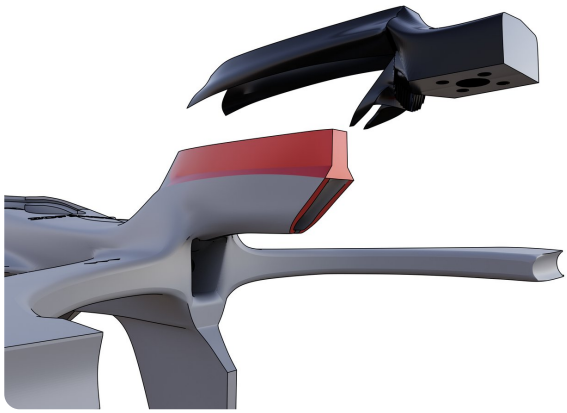
The FlexiLight nose cone glues onto the front bulkhead (red rim). It protects the airframe and payload during VTOL landings.



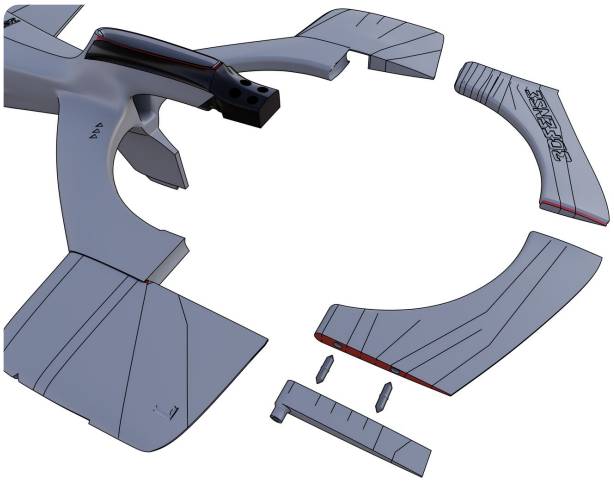
FlexiLight nose cone — glue rim marked red

4 — Tail & ring empennage

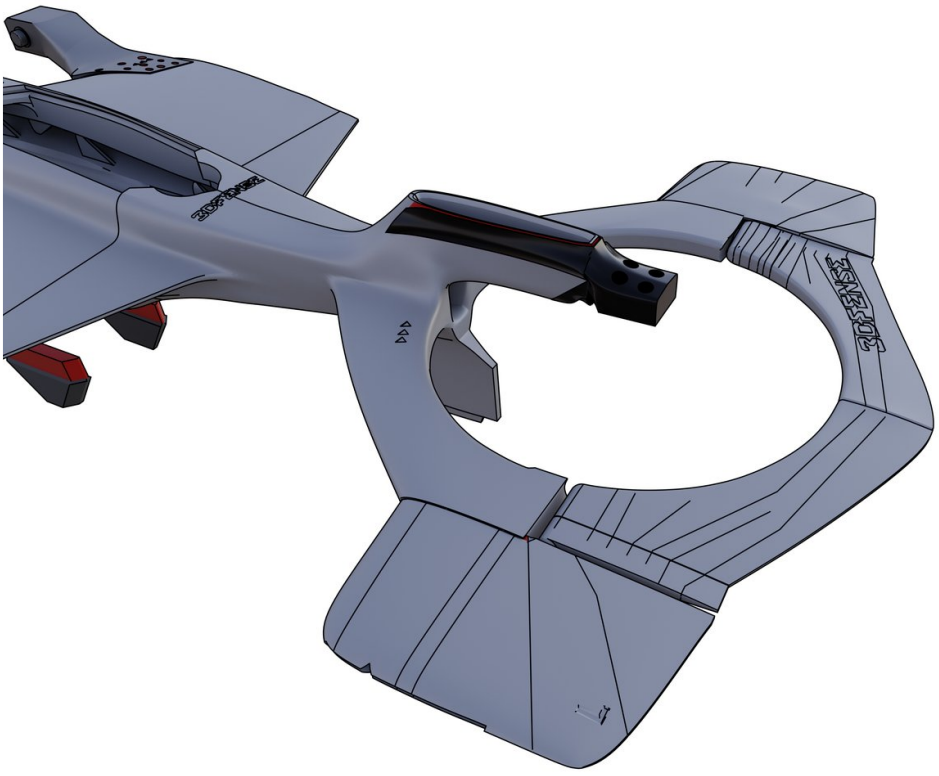
Assemble the tail boom (3AB + 3C), then close the ring tail around the rear rotor position. The PET-G rear motor mount bolts to the tail section before closing the ring.



Rear motor mount & pod on the tail section



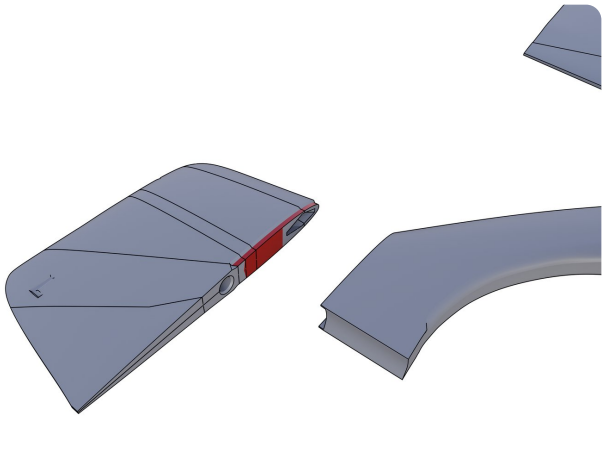
Ring tail segments with alignment pins + elevator



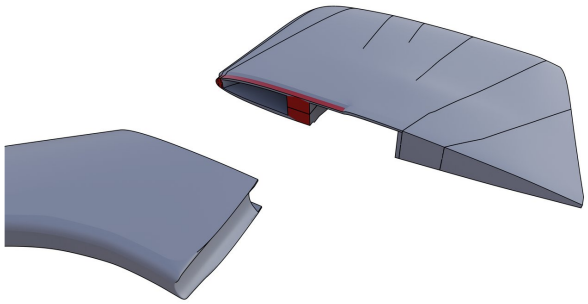
Completed ring tail on the booms

5 — Wings

Bond the outer wing panels to the inner sections — pins align the spar boxes (red faces).



Outer panel joint

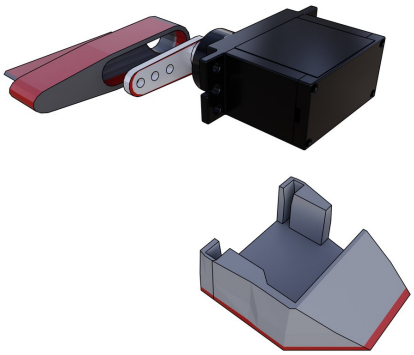


Spar box faces marked red

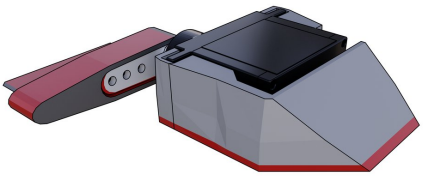
6 — Servos & control surfaces

The D-20C uses **no pushrods** — a printed servo arm is glued directly onto the servo horn and drives the control surface itself. Before gluing, **center the servo** (1500 µs / neutral on a servo tester) so the printed arm sits in the neutral position. Exact throws are set later with the red printed setting tools — see *Center of gravity & control throws*.

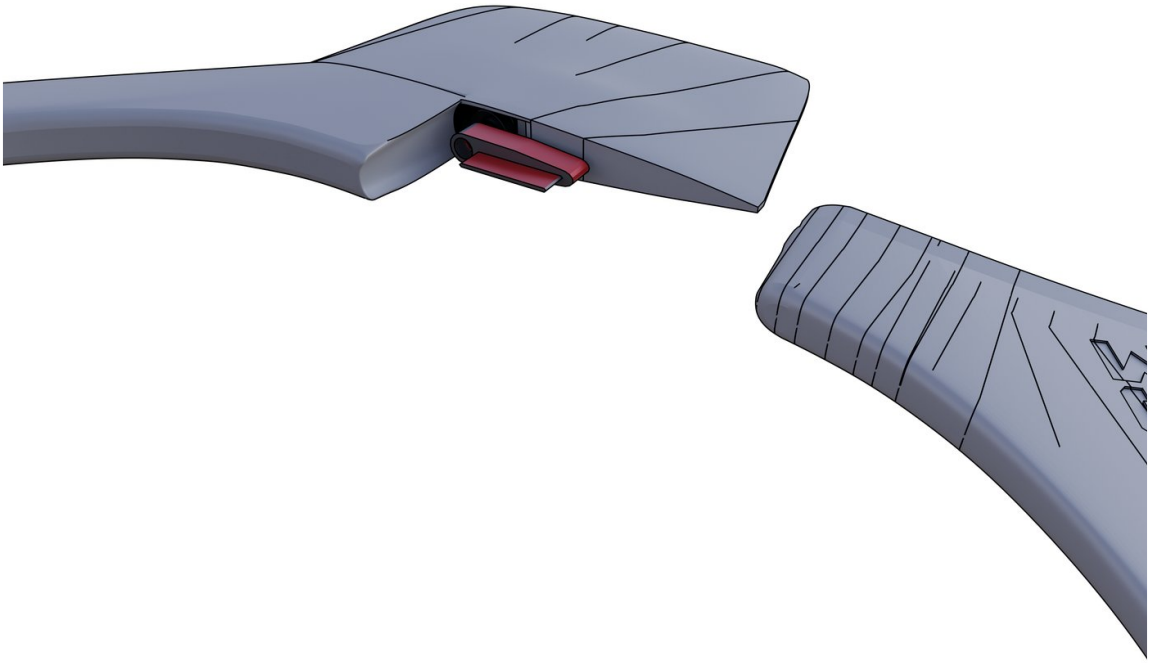
Elevator — glue the printed arm onto the servo horn and seat the servo (PDI-1109MG) in its printed mount; all glue faces are marked red. The elevator surface then simply slides onto the printed arm — it stays removable for transport.



Elevator servo, printed arm and mount — glue faces red

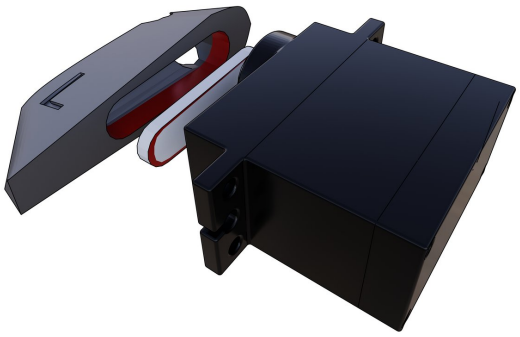


Servo seated in the printed mount, arm glued to the horn

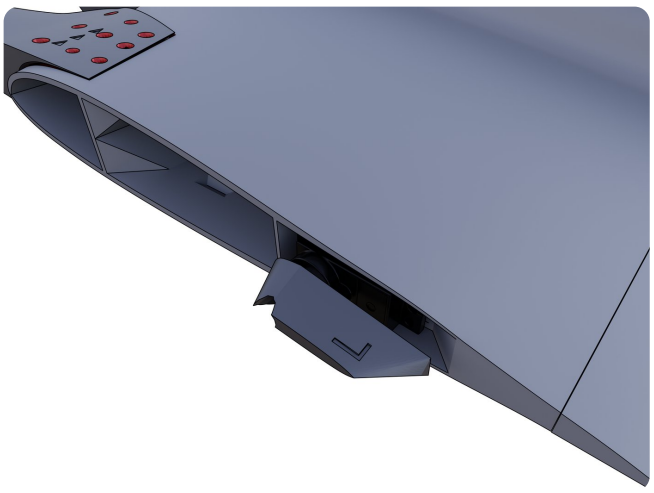


The elevator slides onto the printed arm — tool-less, removable field joint

Ailerons — same principle; the printed arms are marked **L / R**. Drop the servo into the wing pocket and the arm engages the aileron directly.



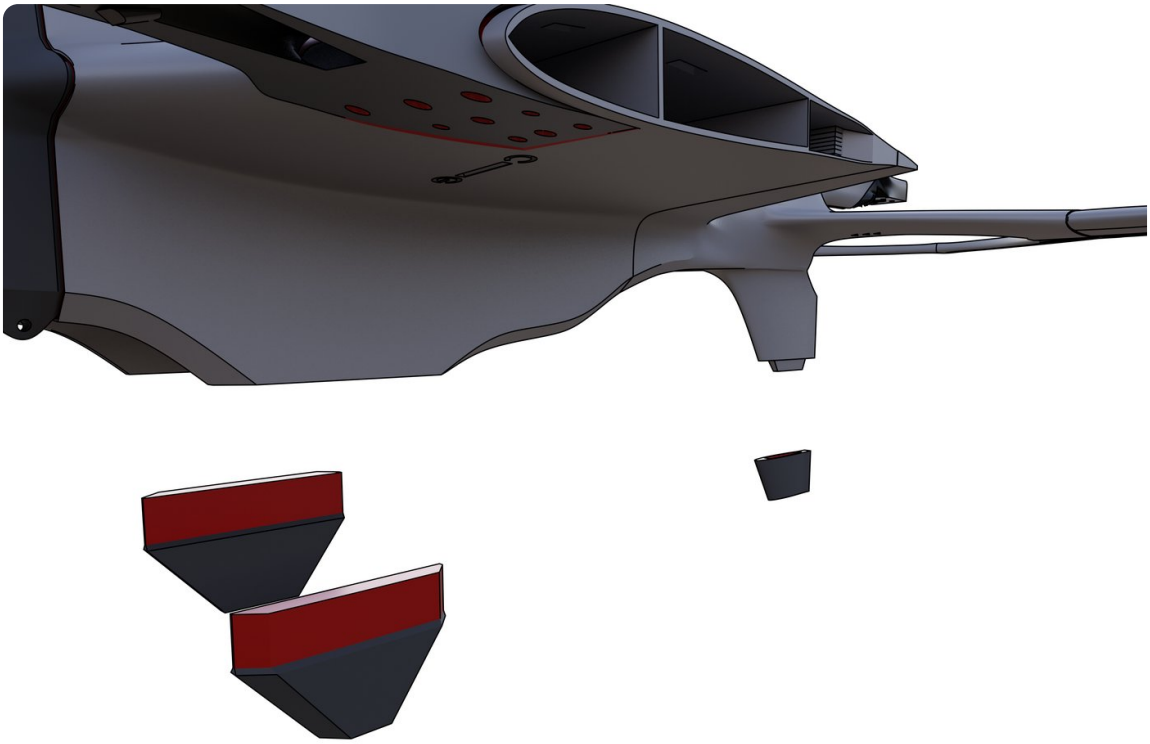
Aileron servo and printed arm (marked L) — glue faces red



Aileron servo in the wing pocket, printed arm engaging the aileron

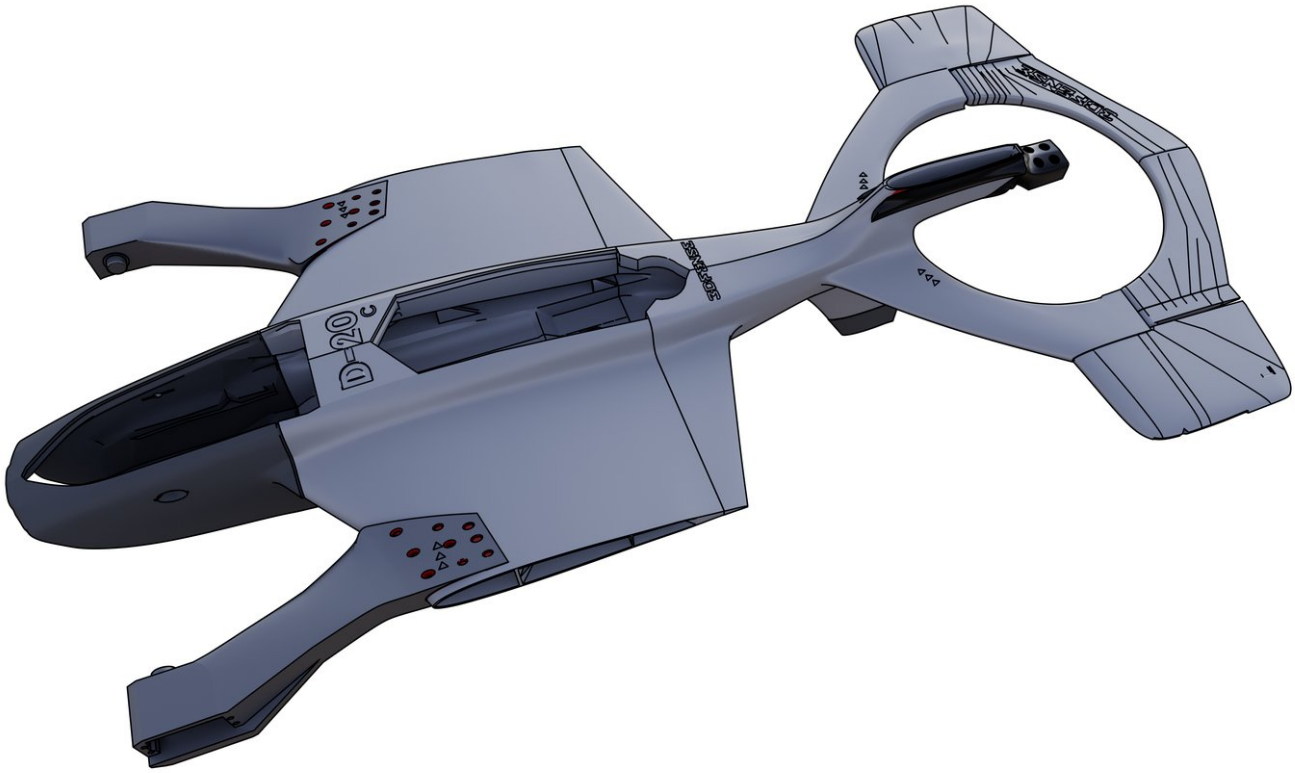
7 — Bumpers

Glue the FlexiLight landing bumpers to the fuselage underside — they are the landing gear.



FlexiLight bumpers — glue faces red

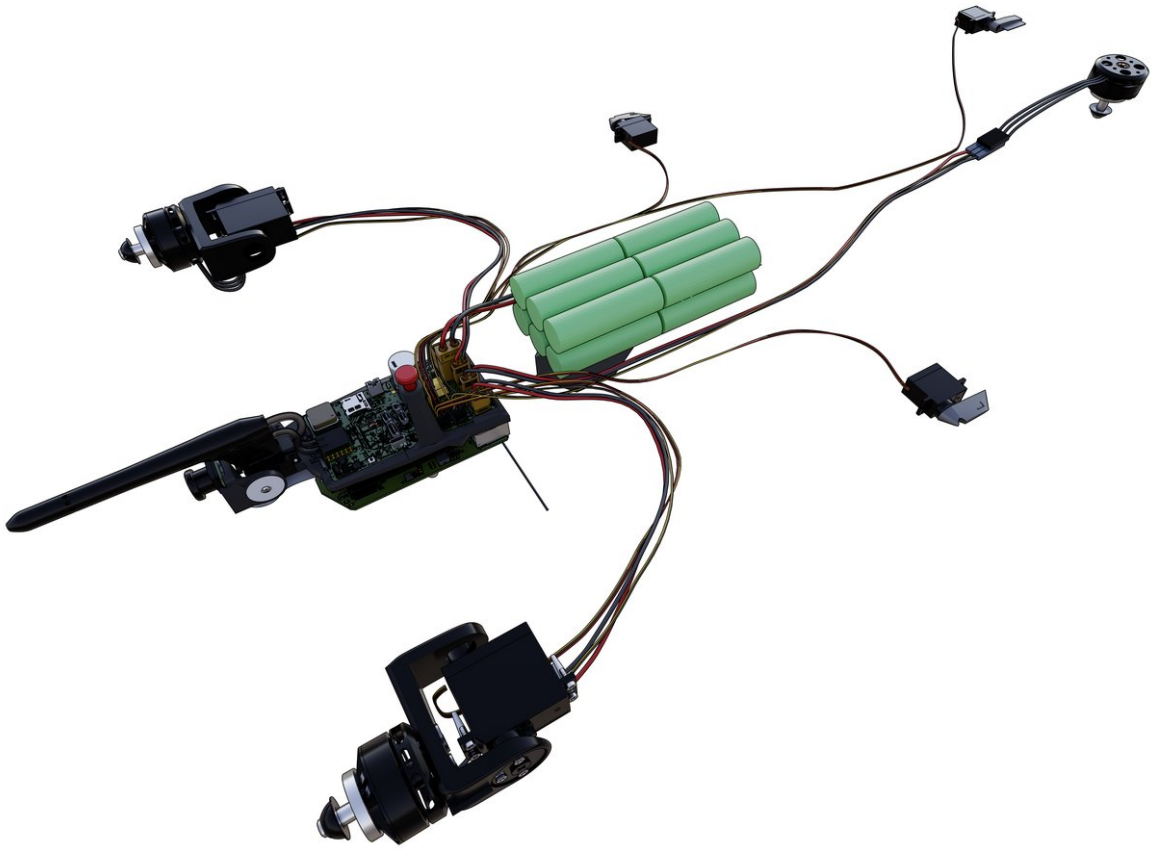
8 — Final check



Completed D-20C airframe

Electronics installation

Everything that flies in the D-20C in one view:



Complete electronics set: NavPack D04 with pitot & START button, 6S2P 21700 Li-ion battery, two front tilt gondolas with ESCs, rear motor with ESC, control servos

1. **NavPack D04** — mount the board in its printed holder in the center fuselage bay, arrow forward. See the *NavPack D04 User Guide* for connector pinout. Installation video: youtu.be/cLdsuQkZW94
2. **Pitot-static tube** — install the printed pitot in the nose area, connect the silicone tubing: pitot → dynamic port, static → static port. Verify both channels are clear and leak-free.
3. **ESCs & motors** — assemble the front tilt gondolas first (see *Tilt motor mount installation* below), then route the cables through the arms; rear ESC lives in the tail boom. All three ESC signal leads go to the NavPack ESC outputs (BDShot — no calibration needed). We use the **Flycolor Raptor5 G071 50 A**; any equivalent ESC works, but it **must support bidirectional DShot (BDShot)** — the NavPack reads motor RPM from the ESCs for the dynamic notch filter and health monitoring. Motor channels follow the same left-to-right rule: front L (3) → rear (4) → front R (5). Cable lengths: front ESCs 35 cm power & signal, rear ESC 54 cm — see the wiring figure below. Motor **rotation direction is set purely in software** — with DShot there is no wire swapping. The parameter `SERVO_BLH_RVMASK` (Mission Planner → CONFIG → Full Parameter List) is a bitmask of reversed outputs: output 3 = 4, output 4 = 8, output 5 = 16 — add the values to reverse several motors. Factory setting is `SERVO_BLH_RVMASK = 4` (front-left motor reversed). Change the value, write params, reboot, then re-check all directions **with propellers removed**.

4. **Propellers** — front tilt motors: **Aeronaut CAMcarbon Light 10×5"**, one **CW** and one **CCW** (the front motors counter-rotate). Rear: **CAMcarbon Light 8×4.5"** — it is made in one rotation direction only, so **check that the rear motor spins the right way before the first flight**. The renders show 5-blade props — you can try those as well. With the recommended props you may need to print a small spacer washer so the blades don't collide with the motor bell.

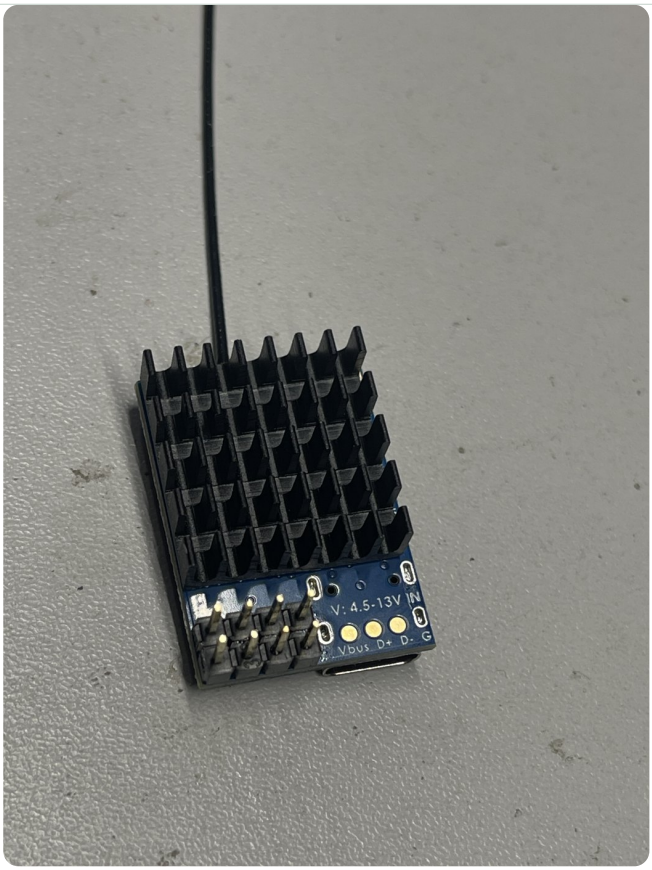
   Alternative propellers of the same size work too — e.g. **APC**: a 10×5 E + 10×5 EP (pusher) pair at the front and an 8×4.5 at the rear. Stay close to the stock diameter and pitch, check that the blades clear the airframe and motor bell, and expect slightly different hover current and endurance — of everything we have tested so far, the stock Aeronaut CAMcarbon props are still the most efficient. We have also flown the front motors on **9" props** — it works as a compromise: more clearance, but higher hover current and less thrust margin, so expect shorter endurance. The harmonic notch filter tracks ESC telemetry, so it adapts to different propellers automatically — still, balance the props and watch the vibration levels on the first flights.

5. **Servos** — plug into the NavPack servo rail. Channel numbers follow the servo position in the airframe, **left to right, starting at CH 9**: tilt L (9) → aileron L (10) → elevator (11) → aileron R (12) → tilt R (13). See the wiring figure below for servo types and lead lengths.

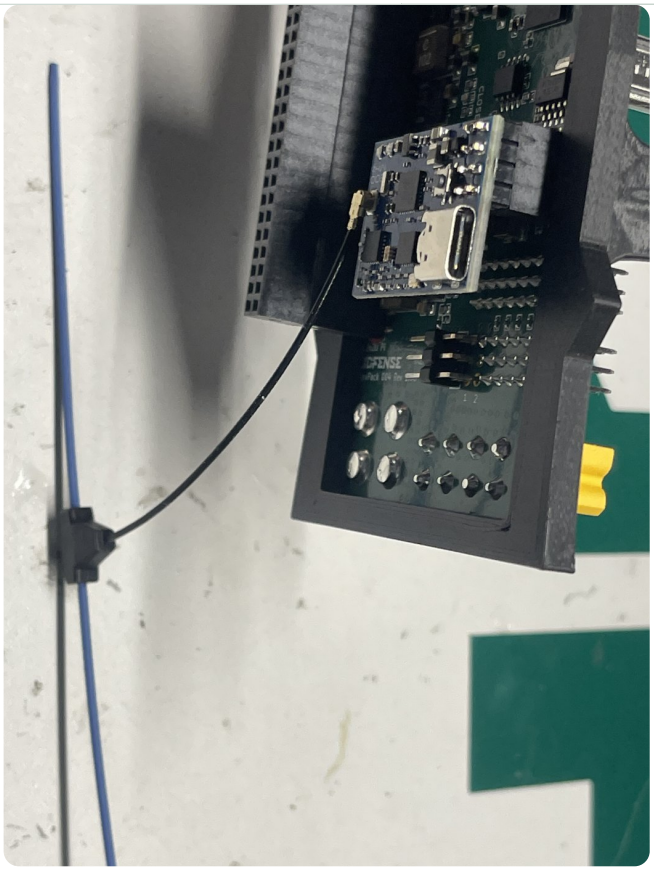
6. **Battery** — we use a **6S2P 21700 Li-ion** pack (~10 000 mAh), but any pack of matching voltage and similar capacity works (Li-Po etc.). It slides into the FlexiLight holder on the CG; XT60 to the NavPack pass-through. Our pack is held perfectly by the printed holder and covers; **if you use a different battery, always secure it** so it cannot travel inside the fuselage — a shifting pack changes the CG in flight.

7. **START button** — mount the printed Autolaunch button on the fuselage and wire it to the NavPack START input.

8. **RC receiver** — mLRS / ELRS / CRSF module connects to the mLRS combo connector (RC + full MAVLink telemetry in one plug). Our favourite is the **Matek mLRS MR900-30** (mateksys.com) — its 2.54 mm pin header plugs straight onto the NavPack mLRS combo connector and it works with our firmware flash **out of the box, no modifications**. In your transmitter use the matching **MR900-30-TX** module (mateksys.com) — besides the RC link it streams **MAVLink over Bluetooth** to your notebook ground station (Mission Planner or QGroundControl).



Matek mLRS MR900-30 — 2.54 mm pin header on the bottom, 4.5–13 V input



Plugged straight onto the NavPack mLRS combo header — ready to fly, no wiring

Tilt motor mount installation

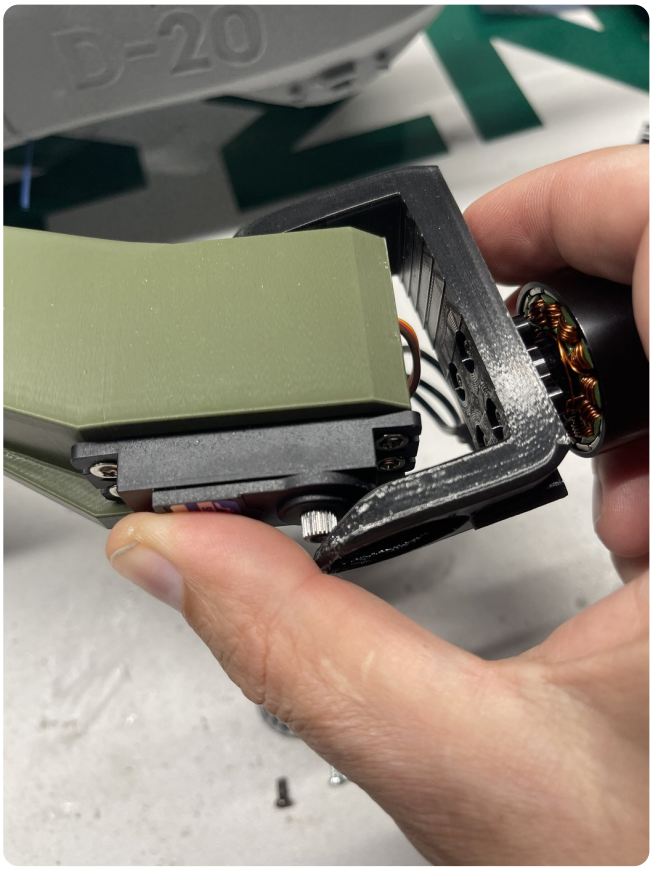
The printed tilt motor mount transfers the servo motion to the whole motor gondola, so the servo horn must be attached in exactly the right position. The easiest way is with a simple **servo tester**: set **1000 µs = motor horizontal** (forward flight) and **2000 µs = motor vertical** (hover), then lock the horn.

Step 1 — Screw the tilt servo (PS-5515MG) into the gondola arm — four screws. Prepare the printed tilt mount, the round servo horn and the screws.



Tilt servo screwed into the gondola arm, printed mount and horn ready

Step 2 — Fit the printed mount over the servo — with the motor already bolted on, the mount wraps around the servo body; its round pocket sits over the servo output shaft.



Fit the printed tilt mount over the servo



Mount seated — the round pocket accepts the servo horn

Step 3 — Set the servo with a servo tester. Plug the tilt servo into a servo tester and find the position of the horn on the output splines so that:

- at **1000 µs** the motor points **horizontal** (forward flight),
- at **2000 µs** the motor points **vertical** (hover).

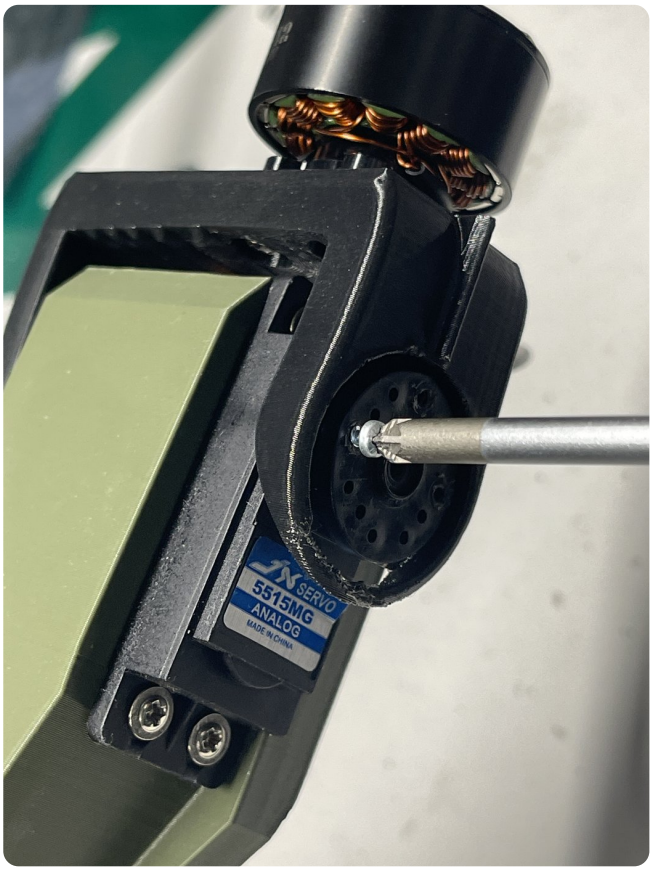


1000 µs on the tester — motor horizontal



2000 µs on the tester — motor vertical

Step 4 — Lock the horn. With the position found, press the round horn into the mount pocket, fix it with the central servo screw, then lock it with the three **2.5 × 6 mm** self-tapping screws through the mount.



Central servo screw through the horn



Three 2.5 × 6 mm self-tapping screws lock the horn to the mount



Completed tilt gondola

The 1000/2000 µs endpoints are only a mechanical starting point — the exact flight angles are fine-tuned later with the printed **–17 / –2 setting jig** (see *Center of gravity & control throws*).

NavPack outputs (golden firmware mapping)

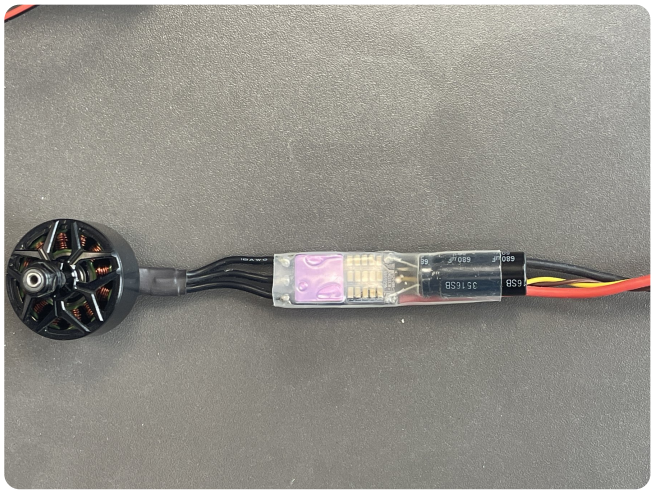
Output	Function	Component / cable
1	Camera mount tilt	—
2	Gimbal holder in/out	—
3	Front motor — left (BDShot)	2812 · 900 KV — leads 11 cm, ESC 50 A, power & signal 35 cm
4	Rear motor (BDShot)	3110 · 1200 KV — leads 12 cm, ESC 50 A, power & signal 54 cm
5	Front motor — right (BDShot)	2812 · 900 KV — leads 11 cm, ESC 50 A, power & signal 35 cm
9	Tilt servo — left	PS-5515MG · 43 cm
10	Aileron — left	PDI-1109MG · 41 cm
11	Elevator	PDI-1109MG · 75 cm
12	Aileron — right	PDI-1109MG · 41 cm
13	Tilt servo — right	PS-5515MG · 43 cm

All cable lengths in the table and the wiring diagrams below are **total lengths, measured to the end of the connector**. The easiest way to get them: buy ready-made **servo extension leads**, cut each one to the required length and **solder it onto the cable coming from the servo** (heat-shrink every joint). It also works completely solder-free — use the extensions and servo leads as they are, connector to connector — but the extra connectors in the middle make the harness bulkier and harder to pull through the fuselage and wings.

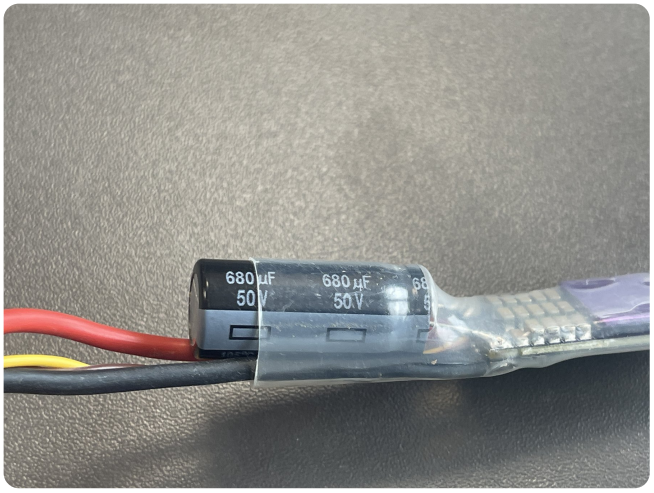
The ESC **power leads are 1.5 mm² (≈ 15–16 AWG) silicone wire** — do not go thinner over these lengths — and they connect through **XT30 connector pairs**. When soldering the power harness (motor leads, XT30s, capacitors) use a **quality soldering flux**: joints carrying 50 A must be shiny and fully wetted, a dry joint here means a motor cutting out in flight.

Do not skip the ESC capacitors. Every ESC gets a **low-ESR electrolytic capacitor, 680 µF / 50 V**, soldered directly across its power input (+ / –, mind the polarity — the stripe marks the negative lead) and heat-shrunk together with the ESC. The long power leads (35 cm front, 54 cm rear) have enough inductance to produce voltage spikes at every throttle change; without the capacitor these spikes stress and can destroy the ESC. Keep the capacitor's own legs as short as practical.

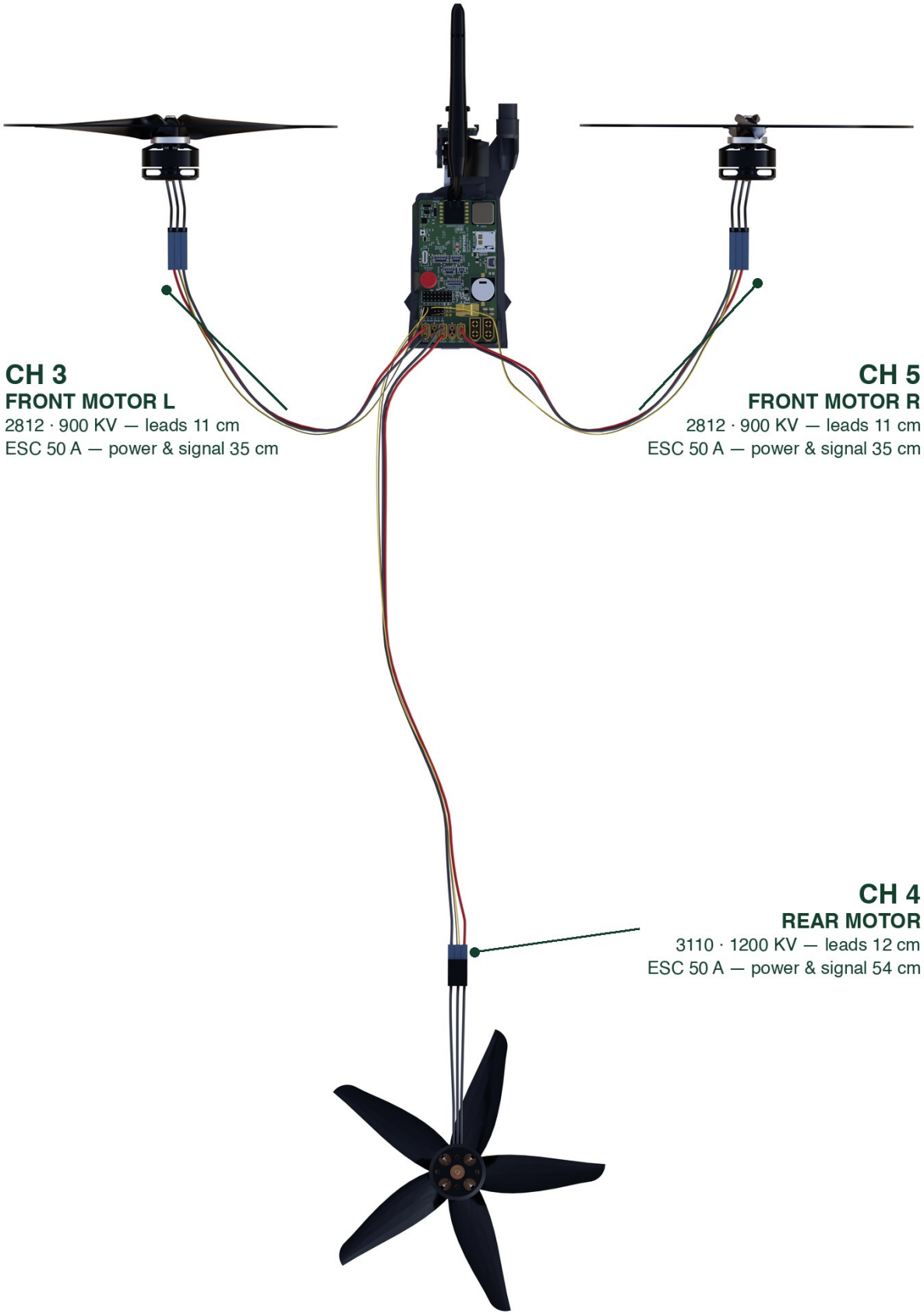
> **Coming soon:** for now the motor–ESC harness is the one part you still solder yourself. We are preparing complete **motor + ESC packs** for the e-shop — motors, ESCs, capacitors, leads and XT30s pre-soldered, heat-shrunk and tested, ready to bolt in. Watch 3dlabprint.com.



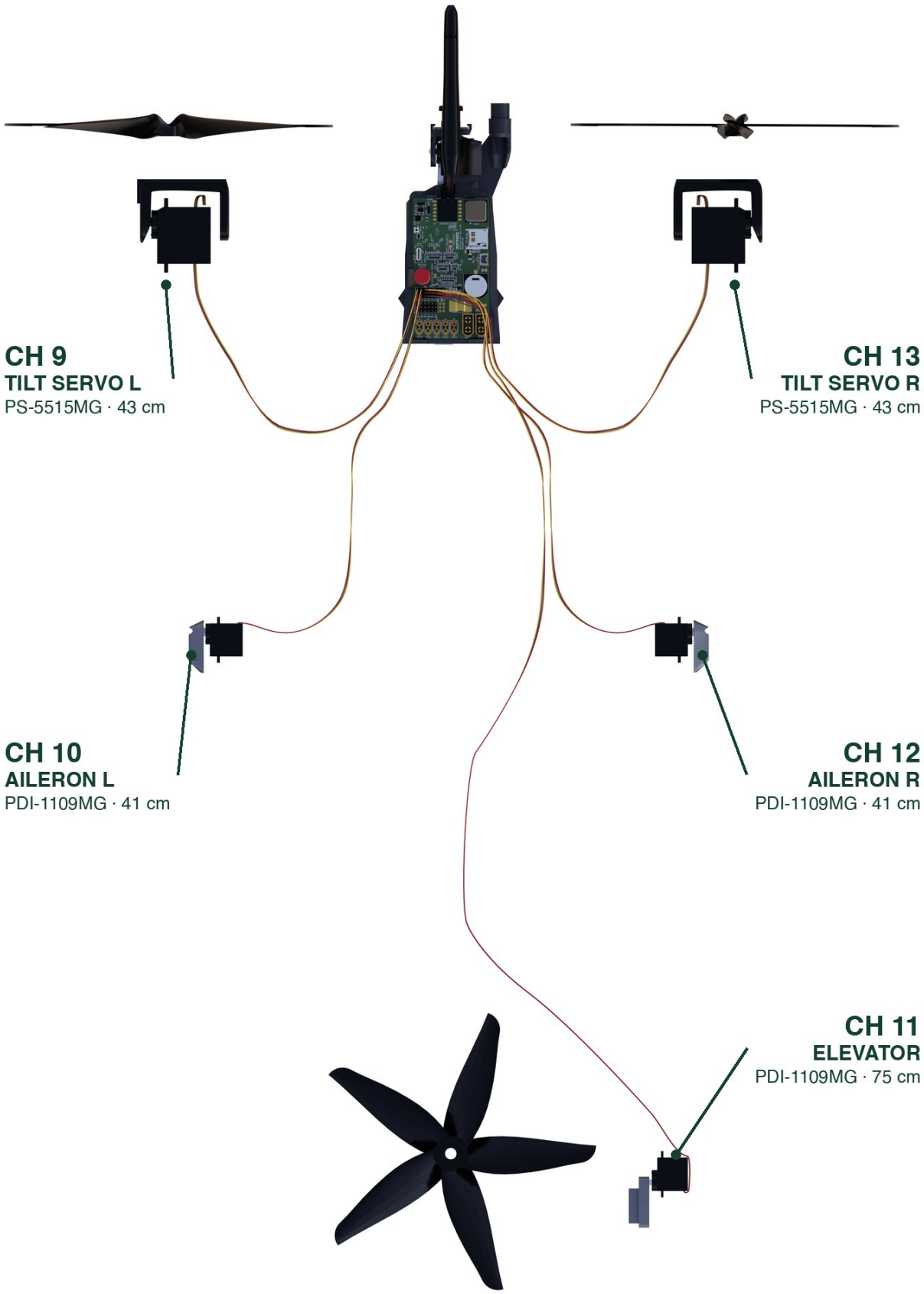
ESC with the capacitor heat-shrunk at its power input, motor side on the right



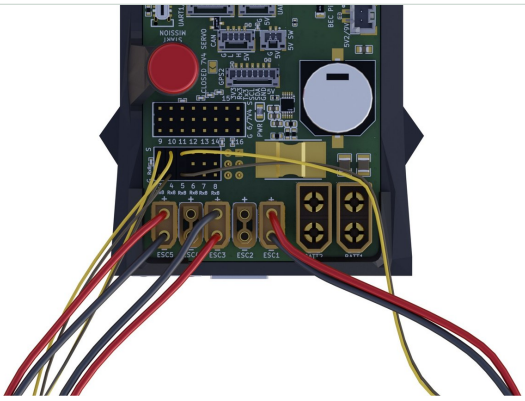
Detail — low-ESR 680 µF / 50 V electrolytic across + / –



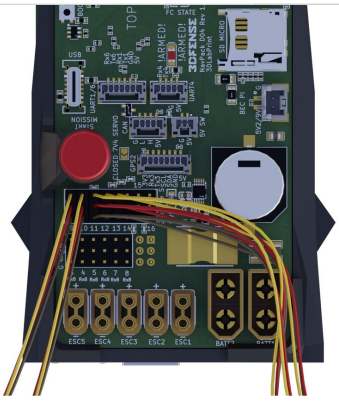
Motor & ESC wiring — types, total cable lengths (to the end of the connector) and channel numbers (CH 3–5, left to right across the airframe)



Servo wiring — types, total lead lengths (to the end of the connector) and channel numbers (CH 9–13, left to right across the airframe)



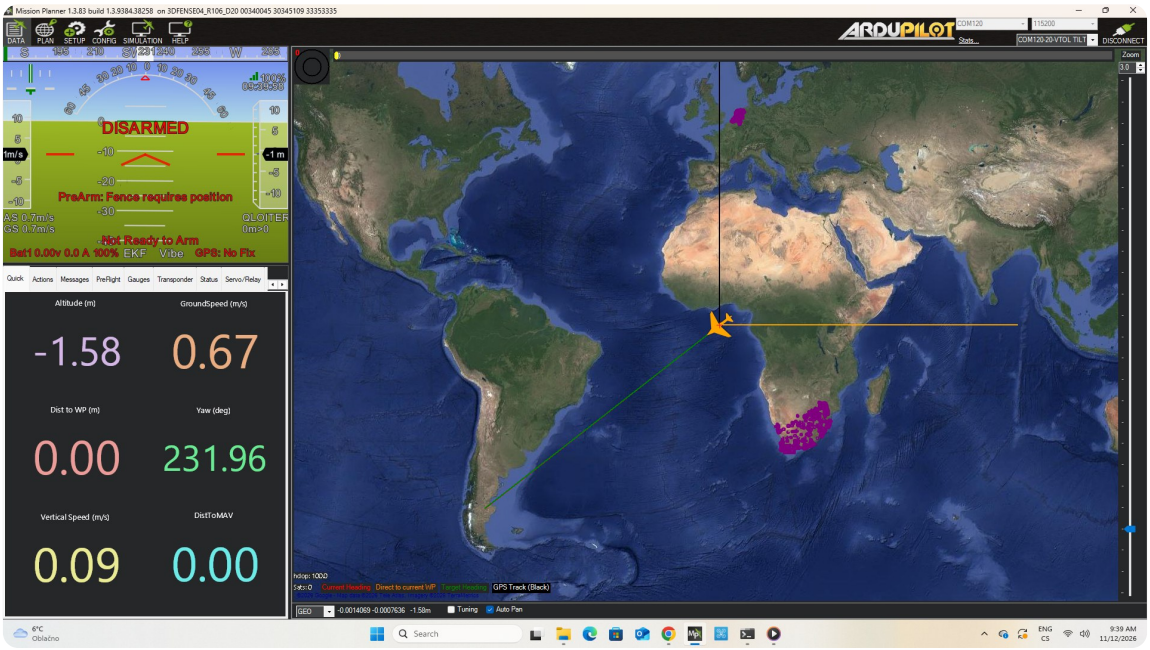
ESC connection detail — power leads into the + / – bullet connectors, signal leads into outputs 3–5



Servo connection detail — plugs on outputs 9–13, mind the S / + / G orientation marked on the board

Firmware & configuration

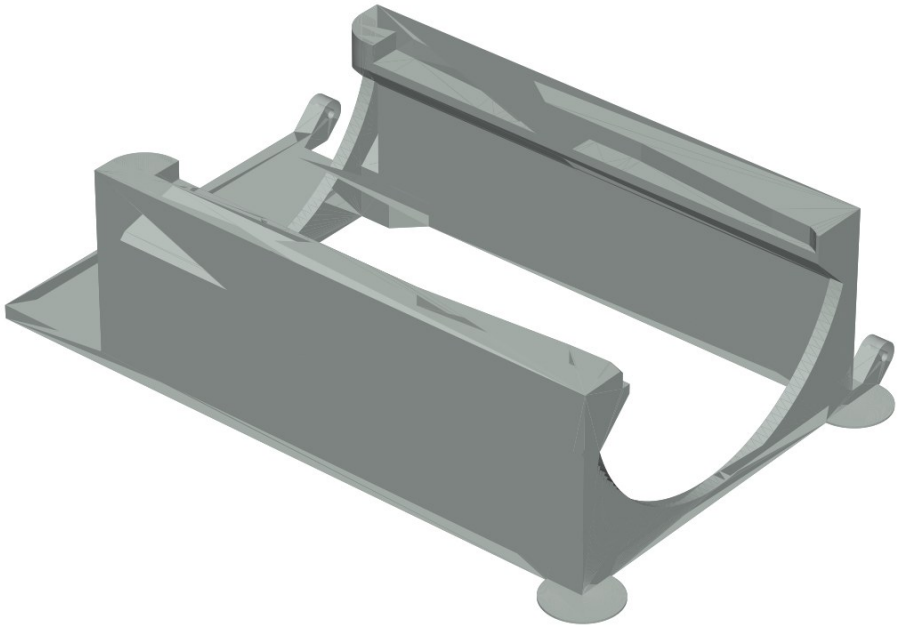
The NavPack D04 is a finished, ready-to-fly unit — there is nothing to build or program. It flies the **3DFENSE golden firmware image**: bootloader, ArduPilot and the complete D-20C parameter set (mixes, transition logic, Autolaunch) in a single file. Ordered with the **D-20C VTOL firmware/setup** option, it arrives flashed and tested. Connect the board to **Mission Planner** over USB-C (select the COM port, *CONNECT*) and the DATA screen should greet you just like this — board identified as `3DFENSE04`, mode **QLOITER**, ready for the calibrations:



First connection — Mission Planner DATA screen, NavPack D04 with the D-20C golden image (DISARMED, QLOITER)

Everything NavPack-specific — flashing & updating the firmware, connector pinout and the **per-aircraft calibrations** (accelerometer, compass, airspeed, RC, battery monitor) — is covered step by step in the **NavPack D04 User Guide** included in your downloads, and in the NavPack video: youtu.be/cLdsuQkZW94. Run the calibrations once on the finished airframe before the first flight; this guide only deals with what is specific to the D-20C airframe.

Print the calibration cradle. Your downloads include `3DF_NavPack_cradle.stl` — a simple printed cradle (PLA/PLA+, ~15 min job) the NavPack drops into for bench work. It holds the round nose section perfectly steady and level, which makes the **accelerometer calibration** (holding the unit still in all six orientations) and the final **level-horizon calibration** far easier and more accurate than balancing the bare cylinder on a table.



Printable calibration cradle (3DF_NavPack_cradle.stl) — the NavPack sits in it steady and level for the accelerometer and level calibrations

Using a different flight controller

The D-20C flies best on the NavPack, but it is a standard **ArduPlane QuadPlane (tiltrotor)** airframe — any ArduPlane-capable flight controller can fly it, for example a **Matek H743 Wing**. To save you weeks of tuning we publish the complete airframe tuning as a universal parameter file:

D-20C_universal_anyFC_v5.param — all D-20C-specific tuning (VTOL & fixed-wing PIDs, TECS, transition logic, servo functions and end points, Siyi gimbal and gimbal-holder setup), with everything board-specific stripped out.

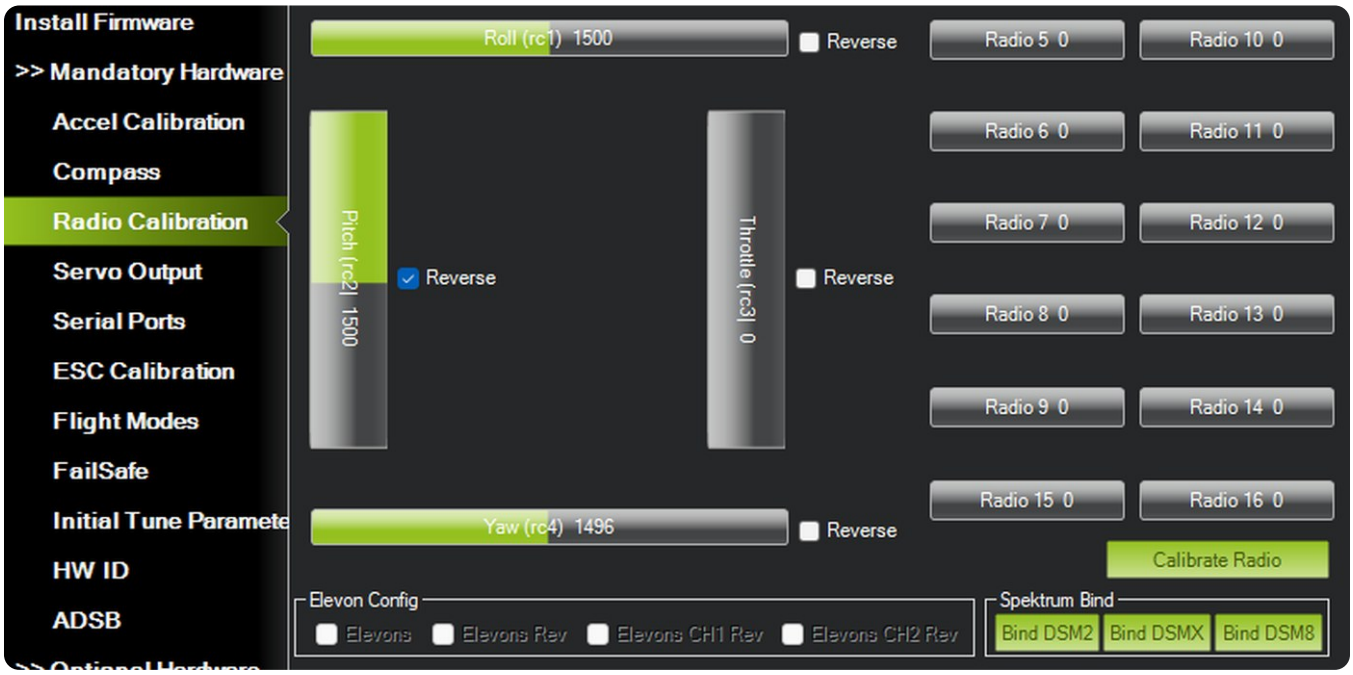
Requirements and limitations:

- **ArduPlane 4.6 or newer** with QuadPlane support (a 2 MB-flash board recommended).
- At least **10 PWM outputs**: 3× motor on a DShot300-capable group, 2× tilt servo, 2× aileron, 1× elevator, plus optional gimbal holder and front camera tilt. Functions may be remapped to any output — keep each function's MIN/MAX/TRIM/REVERSED values together with it (they are the end points of the printed airframe).
- **You must still configure yourself:** accelerometer/compass/RC calibration and board orientation, battery monitor, airspeed sensor type + calibration, GPS type (the NavPack uses DroneCAN), SERIAL port mapping — including one free UART for the Siyi gimbal (`SERIALx_PROTOCOL = 8` , `SERIALx_BAUD = 115`) — and LEDs/relays.
- Load the file **twice with a reboot in between** (gimbal and landing-gear sub-parameters only appear after the first write).

- The tuning was flown on the factory motors and propellers. With different hardware bench-check motor order, tilt directions and control surfaces **without propellers**, verify the first hover in QHOVER carefully, and expect to re-run autotune.
- Features that rely on NavPack hardware (Autolaunch button, status LEDs, DroneCAN GPS/compass) are not included and need your own setup.

Flight modes & RC setup

First bind your receiver and **calibrate the radio**: Mission Planner → **SETUP** → **Mandatory Hardware** → **Radio Calibration** → *Calibrate Radio* — move both sticks and all switches through their full travel, then confirm. Check the bars respond to every channel; on our setup the **Pitch (RC2) channel is reversed** (Reverse checkbox), as in the screenshot:



Mission Planner → SETUP → Radio Calibration — full stick travel on all channels, Pitch (RC2) reversed

Channel mapping

The golden parameter set expects the following channel layout — set up your transmitter to match:

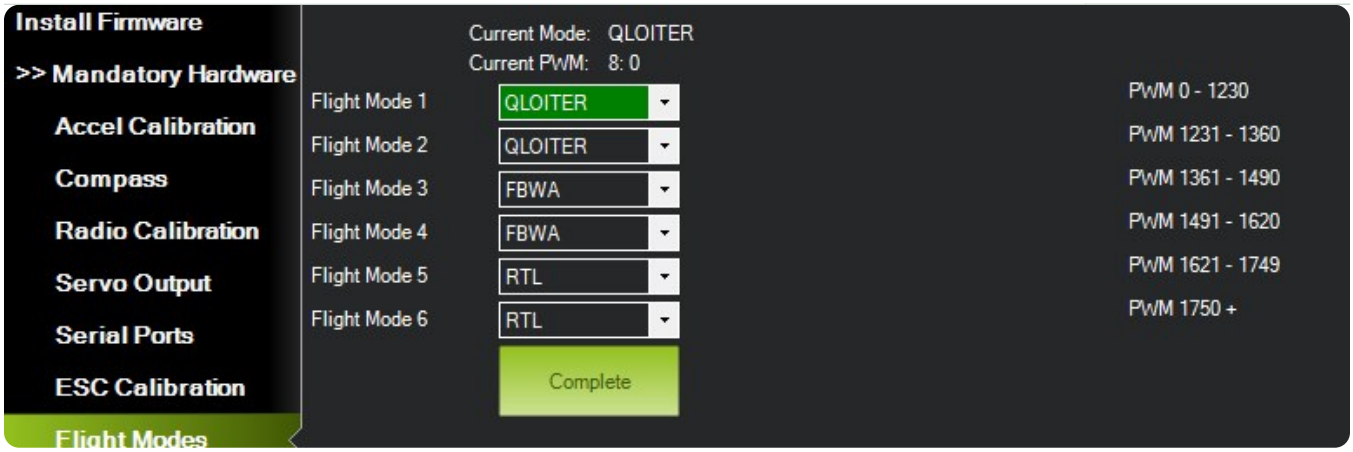
Channel	Function	Notes
RC1	Roll (ailerons)	
RC2	Pitch (elevator)	**reversed**
RC3	Throttle	
RC4	Yaw	
RC8	Flight mode switch	3-position switch, see table below
RC9	Gimbal holder in/out	2-position switch, optional — normally automatic by altitude
RC10	Arm / Disarm	2-position switch
RC11	Camera gimbal — tilt (pitch)	knob or slider
RC12	Camera gimbal — pan (yaw)	knob or slider, reversed
RC13	Front camera tilt	knob or slider, optional
RC14	Camera trigger	momentary switch / button — takes a photo
RC15	Relay 1 — companion (Raspberry Pi) power	2-position switch, optional
RC16	Relay 2 — auxiliary 5 V payload output	2-position switch, optional

The optional **front camera tilt servo** (NavPack output 1) is a direct RC13 passthrough — the servo simply follows the knob, independently of the payload-bay gimbal. Limit its travel with the end points on your transmitter.

To start, you only need **three modes on a single 3-position switch**. ArduPilot has six mode slots, so each mode is simply assigned twice:

Switch pos	Mode slots	Mode	What it does
1	1 + 2	**QLOITER**	VTOL — GPS position & altitude hold; takeoff and landing
2	3 + 4	**FBWA**	fixed wing — stabilized, angle-limited cruise
3	5 + 6	**RTL**	return to launch — comes home and lands by itself

Set it in Mission Planner: **SETUP → Mandatory Hardware → Flight Modes** — pick the mode for each slot and press *Complete*. Flip your mode switch and watch *Current Mode* to verify the positions:



Mission Planner → SETUP → Flight Modes — QLOITER / FBWA / RTL on one 3-position switch

Transitions are fully automatic: push into forward flight from QLOITER and the gondolas tilt forward as airspeed builds; slow down into a Q-mode and the aircraft converts back to hover. Once you are comfortable, add more modes (QHOVER, AUTO for missions) on a second switch.

> **The D-20C hovers nose-up — by design.** In all Q-modes the golden parameters set `Q_TRIM_PITCH = 11°` : in hover the aircraft does not sit level but pitched about 11° back, like a flared landing. **This is intentional — do not try to trim it out.** The attitude comes from the aerodynamics of the whole airframe and gives much better penetration into a headwind and better efficiency whenever the aircraft moves forward in the hover modes.

Failsafe: on RC/telemetry loss the aircraft automatically returns home (`FS_LONG_ACTN = 1` → RTL).

Center of gravity & control throws

Center of gravity

No measuring, no tables: the correct CG is **engraved directly on the wing undersides**, near both wing roots. With the battery and payload installed, support the aircraft on your fingertips exactly at the two CG marks — it should balance level or very slightly nose-down.



CG marks engraved on the wing undersides — balance the aircraft here

If the aircraft is tail-heavy, shift the battery forward in its FlexiLight holder (and vice versa). Always re-check the CG after changing payload.

Control throws — printed setting tools

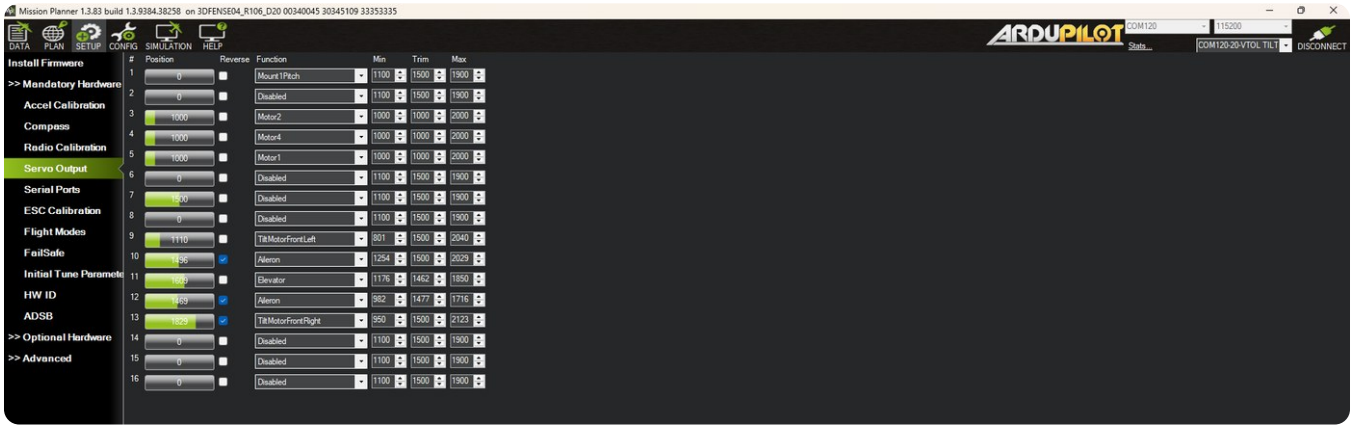
You never measure deflections on the D-20C either. The print data include a set of **red setting tools** (part `D-20-C_servo setting_PLA`) that clip onto the airframe and carry the correct angles engraved:

Control	Tool marks	Where it fits
Front tilt motors	**–17** = hover / **–2** = forward flight	over the motor gondola
Ailerons	**40** up / **0** neutral / **25** down	over the wing trailing edge
Elevator	**35 UP / 30 DOWN**	over the horizontal stab & elevator



The complete set of red setting tools mounted on the airframe

Procedure: in Mission Planner open **SETUP** → **Mandatory Hardware** → **Servo Output**. Each output has **Reverse / Min / Trim / Max** fields: adjust **Trim** until the surface sits on the neutral mark of the tool, then **Min / Max** until the end positions line up with the engraved marks (use **Reverse** if the surface moves the wrong way). The exact values differ slightly from airframe to airframe — the screenshot below is a real example.



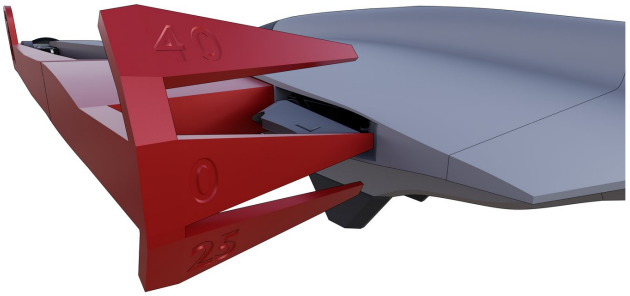
Mission Planner → SETUP → Servo Output — trim and end-points for every channel (example values from a real D-20C)



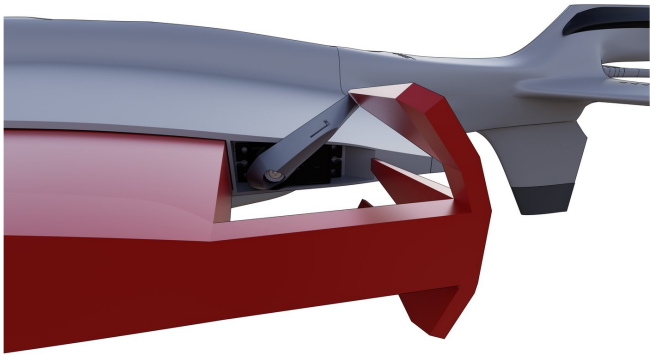
Forward-flight end-point — motor axis on the –2 mark



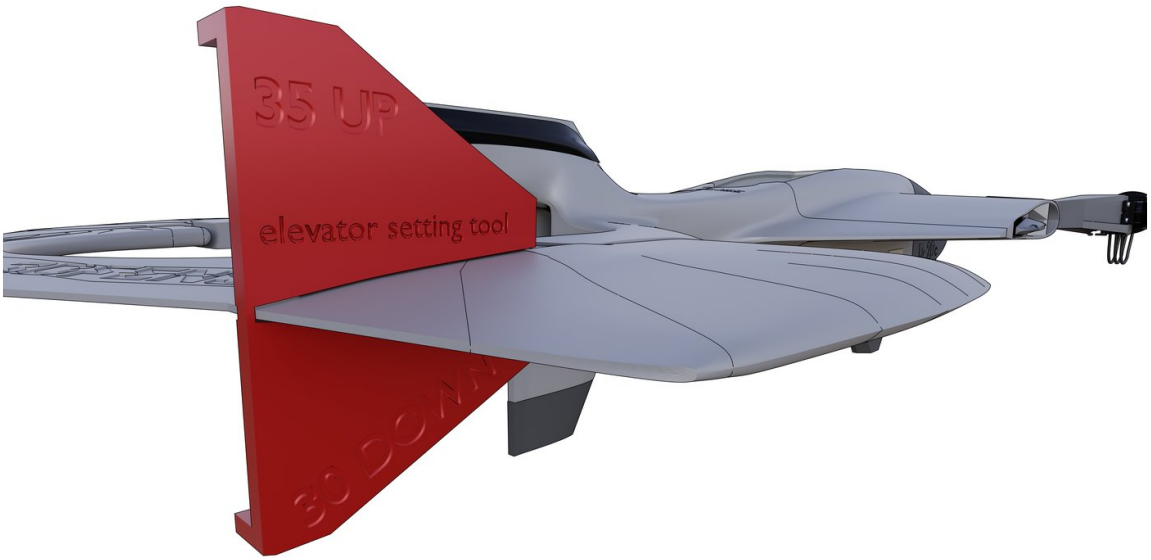
Hover end-point — motor tilted up to the –17 mark



Aileron tool — 40 up / 0 / 25 down marks on the trailing edge



Tool clipped over the wing at the aileron servo



Elevator setting tool — 35 UP / 30 DOWN

First flight

Do not send the aircraft on an auto mission right away. A few minutes of checks on the ground catch rigging and setup errors **before they destroy the aircraft**:

1 — Hand check without propellers. Props off, battery in, switch to **QLOITER** and arm. Take the aircraft in your hands and tilt it around: the motors must react against the motion (tilt it right — the right motor spools up, etc.). Check the **rotation direction of every motor** against the prop markings — in hover all three propellers must push **upwards**. If a motor spins the wrong way, reverse it in software with `SERVO_BLH_RVMASK` (see *ESCs & motors* in Electronics installation) — no wire swapping needed.

2 — Hand check with propellers. Mount the props (front CW/CCW on the correct sides). Hold the aircraft firmly above the ground at low throttle and try the sticks: pitch, roll and yaw must each respond **in the correct direction**. Keep hands and body well clear of the props and grip the fuselage firmly — the aircraft fights back.

3 — First hover. Take off in **QLOITER**, ideally in calm wind. The aircraft stands visibly **nose-up (~11°)** in hover — that is the built-in `Q_TRIM_PITCH`, not a trim error (see *Flight modes & RC setup*). Stay low, try gentle stick inputs in all axes and make sure the aircraft responds predictably. Land, take a breath, repeat.

4 — First transition. Climb to **at least 30 m** and flip the switch to **FBWA**. The aircraft accelerates into forward flight by itself — **during the transition it is controlled automatically and your stick inputs have no effect**; once the transition completes, you are flying a normal airplane.

5 — RTL. Fly some distance away and switch to **RTL** — the aircraft comes back on its own, converts to hover and lands at the takeoff point.

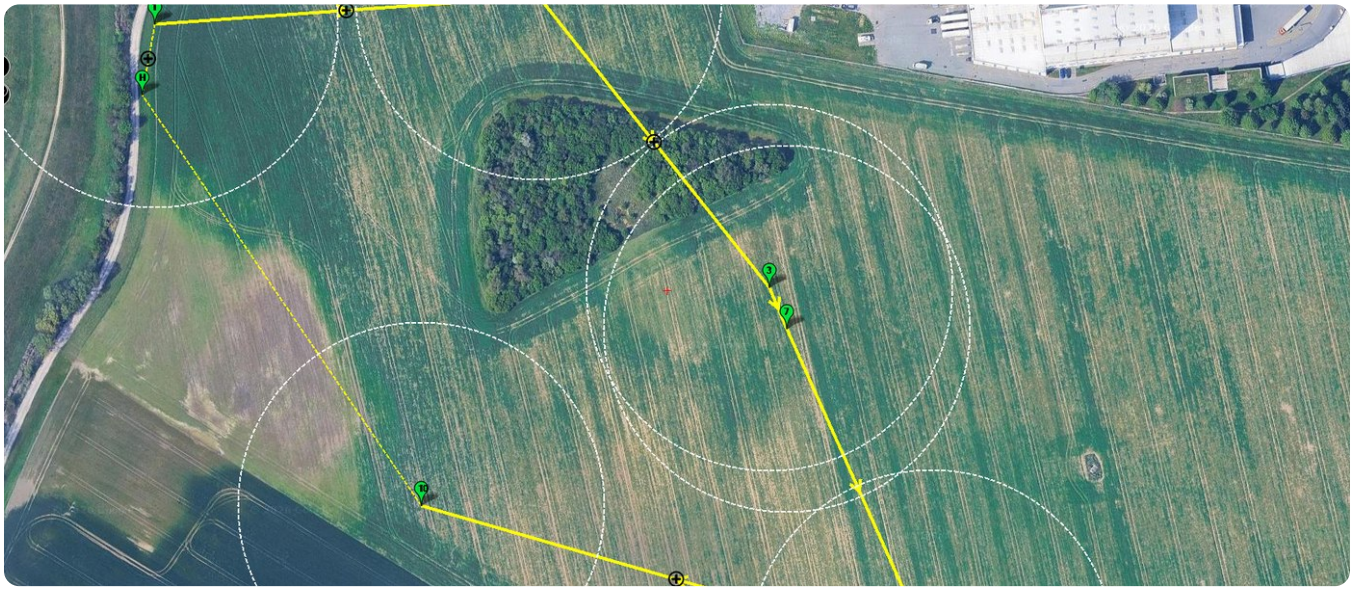
6 — Autonomous missions. Only now load an auto mission (read up on the basics of ArduPilot **VTOL auto missions** first) and use the Autolaunch button — next chapter.

Autolaunch — autonomous missions

The D-20C is designed to fly a complete sortie **without a laptop or RC transmitter**. The red START button on the airframe runs the Autolaunch script on the NavPack: it switches to AUTO, runs pre-arm checks, counts down, arms and flies the stored mission.

Mission

A mission must be loaded in the NavPack and it **must begin with a VTOL TAKEOFF command** — the script refuses to launch otherwise. Plan it in Mission Planner (**PLAN** screen → place waypoints → *Write*): VTOL takeoff, `DO_VTOL_TRANSITION` to forward flight, your waypoints, and `RETURN_TO_LAUNCH` at the end — the aircraft comes back, converts to hover and lands at the takeoff point.



Typical D-20C mission in Mission Planner — VTOL takeoff, transition, waypoint circuit, return to launch

Launch procedure

1. Complete all calibrations and a bench range check.
2. Find an open area; wind below 8 m/s.
3. Place the aircraft level on the ground, nose into the wind, props clear.
4. Plug in the battery, step back, wait for GNSS lock.
5. **Hold the START button for ~3 seconds** — a beep confirms the press is registered — then **release**. The aircraft switches to AUTO (rising two-tone).
6. **5-second countdown** — one beep per second. During the countdown the script checks that the mode is AUTO, the mission starts with a VTOL takeoff and the AHRS is healthy; any failure aborts with an error tone.
7. After the countdown (rising two-tone again) the aircraft **arms and launches**: vertical climb, transition, mission, return, hover landing.

Abort: press the START button again during the countdown — a falling tone confirms the cancel. After landing and disarm the script resets itself; just press the button again for the next launch.

Sound	Meaning
single beep	button registered / countdown tick (1 per second)
two rising tones	AUTO engaged / countdown complete, arming
falling tones	launch cancelled / disarmed
long low tone	pre-arm check failed — launch aborted

Payload



Quick-swap payload bay — the camera gimbal locks into the fuselage with the red levers, no tools needed

The bay under the center fuselage accepts up to **850 g** of payload and is designed for field swaps: the printed quick-lock levers release the whole module in seconds, and the bay sits at the aircraft CG, so typical payload swaps need no rebalancing — still, **verify the CG after installing any new payload** (see Center of gravity).

Current payload options:

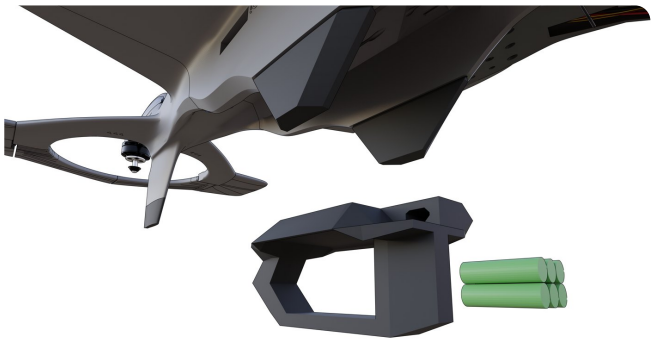
- **Camera gimbal** — the golden parameter set is pre-configured for the **Siyi A8 mini 4K zoom** gimbal camera (hobbydrone.cz): connect it to NavPack **SERIAL6** (UART4, 115200 baud) and it works with no extra setup — RC11 tilts, RC12 pans, RC14 takes a photo. The printed gimbal holder swings in and out

of the fuselage on its own servo (NavPack output 2), driven automatically by altitude (switches at 20/30 m) so you never land on the gimbal; RC9 overrides it manually if needed.

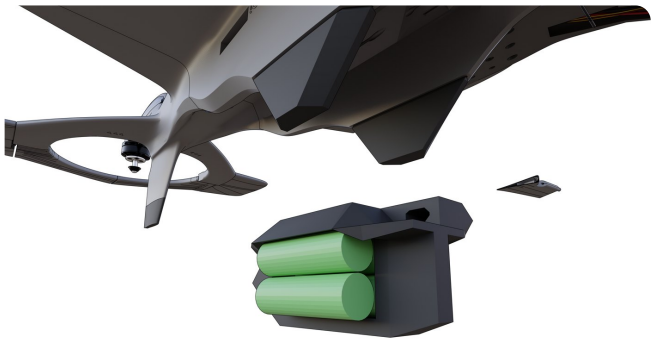
- **Battery range extender** — a second pack in the bay (6S1P 21700 Li-ion, ~5 000 mAh, ~450 g) stretches fixed-wing endurance from 70 to ~90 minutes.
- **Cargo module** — printable container for small cargo up to the payload limit.



Camera gimbal exploded — tilt servo in the printed quick-lock mount, stabilized camera on an anti-vibration plate



Battery range extender — printed holder and 6× 21700 Li-ion cells (6S1P, ~5 000 mAh)



Loaded extender pack ready to lock into the payload bay

Companion computer (Raspberry Pi)

Beyond the payload bay, the NavPack itself carries a **Raspberry Pi GPIO-compatible 2×20 header**: a Raspberry Pi — or any single-board computer with the Pi 40-pin connector, including AI modules — mounts directly onto the board, with no cables at all.

- **Power ready** — one of the NavPack's three on-board buck regulators is a dedicated high-current 5 V rail (8 A class) for the companion computer; no extra BEC needed even for a Pi 5 with peripherals.

- **Switched by the autopilot** — the companion rail is ArduPilot **RELAY 1** (the auxiliary 5 V payload output is **RELAY 2**): keep the computer powered down until the mission needs it, or power-cycle it in flight from the GCS, a mission item or an RC switch. In the golden parameter set both are already on your transmitter: **RC15 switches Relay 1** (Pi power), **RC16 switches Relay 2** (payload 5 V).
- **MAVLink wired** — the header UART pins connect straight to the FC telemetry port (TELEM1), so the Pi talks to the autopilot (mavlink-router, MAVSDK, DroneKit…) the moment it boots.

Typical uses: on-board AI (object detection & tracking), LTE / mesh / Starlink connectivity, photogrammetry capture, custom mission logic.

Maintenance & repairs

- Inspect propellers and tilt mechanisms after every hard landing; print spares in advance.
- The airframe is the spare part: any damaged section reprints in hours and glues back with CA.
- Check pitot tubing for condensation/dust; keep the static ports clear.
- Store the battery at ~3.7 V/cell; never charge below 0 °C.
- Firmware updates: reflash the latest golden image (your calibrations must be redone afterwards — a full image overwrites everything).

Troubleshooting

Symptom	Fix
Motors won't arm	check GNSS lock, safety/arming state, battery voltage > 19.2 V
Weak/no airspeed reading	pitot tubing swapped or leaking; redo preflight calibration
Drifts in QLOITER	redo compass calibration away from metal; check GNSS view of sky
Harsh transition / sink	fly transitions with more altitude margin; check tilt servo travel
Vibration warnings	check prop balance, motor mount bolts, NavPack mount seating

Shopping list

Item	Qty	Note
3DFENSE NavPack D04	1	3dlabprint.com e-shop (flashed or Clear variant)
Motor 2812 900 KV	2	front tilt gondolas
Motor 3110 1200 KV	1	rear, ring tail; 2812 1115 KV also proven — don't go lower in KV (rear thrust margin)
ESC Flycolor Raptor5 G071 50 A	3	6S; must support bidirectional DShot — RPM telemetry to the NavPack
Low-ESR capacitor 680 µF / 50 V	3	one per ESC power input — see *Electronics installation*
XT30 connector pair (M + F)	3	ESC power connections
Silicone wire 1.5 mm², red + black	~2 m	ESC power leads (≈ 15–16 AWG) — 35 cm front, 54 cm rear
Servo extension leads	~6	cut to length and solder onto the servo/signal cables
Solder, quality flux + heat-shrink	1	clean, fully wetted 50 A joints; shrink every joint
Self-tapping screws 2.5 × 6 mm	6	tilt servo horn locking — 3 per gondola
Propeller Aeronaut CAMcarbon Light 10×5"	2	front, 1× CW + 1× CCW — or same-size alternative (e.g. APC 10×5 E + EP)
Propeller Aeronaut CAMcarbon Light 8×4.5"	1	rear; one rotation direction only — check motor direction before first flight; alternative e.g. APC 8×4.5
Tilt servo PS-5515MG	2	43 cm lead
Control servo PDI-1109MG	3	2× aileron (41 cm lead), 1× elevator (75 cm lead)
Li-ion 6S2P 21700 pack	1	~10 000 mAh, XT60
Matek mLRS MR900-30 receiver	1	plugs straight onto the NavPack mLRS combo header
Matek MR900-30-TX module	1	for the transmitter; MAVLink over Bluetooth to MP / QGC
PolyLight 1.1 filament	~0.8 kg	wings, fuselage, tail
FlexiLight 1.0 filament	~0.5 kg	nose cone, covers, bumpers, battery holder
PolyAir 1.0/2.0 / PLA+ filament	~0.3 kg	setting tools & PLA parts
PET-G filament	~0.1 kg	motor mounts
CA glue + activator	1	

Filament quantities are approximate — keep a margin for failed prints and optional parts (gimbal mount, spare battery holders etc.).

Gallery



Red START button on the nose — Autolaunch with no laptop and no transmitter



Hover — front gondolas tilted up, camera gimbal in the payload bay



Hover configuration from below — rear rotor inside the ring tail



Inside the fuselage — 6S2P 21700 pack and the NavPack D04

3DFENSE — Navigation. Power. Launch. One aircraft that does it all.