3DFENSE

NavPack D04 — User Guide

Firmware flashing via DFU (USB-C) · Required 3D-printed parts

3DFENSE · NavPack D04 · Revision 1.06

> **Already got a pre-flashed NavPack?** If you ordered your board with a firmware build selected at purchase, it ships flashed, tested and ready — **you don't need to flash anything**. Skip straight to Before the first flight. This guide is only for clean boards, switching to a different airframe build, or updating firmware.

What you are flashing

3DFENSE firmware images for the NavPack are **complete 2 MB flash images** — bootloader, ArduPilot firmware and (for aircraft-specific builds) the full airframe parameter set in a single `.bin` file. One flash puts the board into exactly the same known-good state as a unit leaving our production line.

Image naming example:

```
3DFENSE04_H743-Rev106-D20C_<githash>_full2MB_....bin
```

Part	Meaning
`3DFENSE04_H743`	board family (NavPack D04, STM32H743)
`Rev106`	board revision — must match your board
`D20C`	airframe build (D-20C Personal Eagle II); universal builds have no airframe suffix
`<githash>`	ArduPilot source revision of the build
`full2MB`	complete image incl. bootloader and parameters

> **Important:** a full image **overwrites everything**, including any parameters and calibrations currently on the board. Back up your parameters first (Mission Planner → Config → Full Parameter List → Save to file) if you want to keep them.

Download images from your order: **3dlabprint.com** → **My Account** → **Downloads** (included with every NavPack D04 purchase).

What you need

- NavPack D04 board
- USB-C data cable (no battery needed — the board powers from USB)
- The downloaded `.bin` image for your board revision
- **STM32CubeProgrammer** (free, Windows / macOS / Linux) — st.com/en/development-tools/stm32cubeprog.html — or `dfu-util` for the command line (see Appendix)

No ground-station software is involved in flashing. Mission Planner / QGroundControl come into play *afterwards*, for calibration and configuration.

Step 1 — Enter DFU mode

1. Disconnect the board from USB and battery
2. **Hold the BOOT button** on the board
3. Connect USB-C to your computer, then release BOOT

A factory-blank board (no firmware yet) enters DFU automatically — just plug it in, no button needed.

In DFU mode the status LEDs stay off and the board enumerates as **"STM32 BOOTLOADER"** / DFU device (VID 0483, PID DF11).

Step 2 — Flash with STM32CubeProgrammer

1. Start STM32CubeProgrammer
2. In the top-right connection panel select **USB** (not ST-LINK) — the DFU device appears as `USB1`
3. Click **Connect**
4. Open the **Erasing & Programming** tab (second icon in the left bar)
5. **File path**: browse to the downloaded `.bin` image
6. **Start address**: `0x08000000`
7. Leave *Verify programming* checked
8. Click **Start Programming** and wait for "*Download verified successfully*" (≈ 30–60 s)
9. Click **Disconnect**

Step 3 — First boot & verification

1. Unplug and replug the USB-C cable (without BOOT)
2. The board boots ArduPilot — status LED comes alive and the board enumerates as a serial device
3. Open Mission Planner, select the new COM port, **baud 115200**, click Connect
4. Verify in *Setup → Advanced → ArduPilot FW Version* (or the HUD messages tab) that the firmware version and build match the image you flashed

The board is now in the ship state of that build. For aircraft-specific images all airframe parameters, mixes and Autolaunch are already configured.

Step 4 — Before the first flight

A full image carries the **airframe configuration**, but sensor calibrations are **per-board** — always redo them on your unit:

1. **Accelerometer + level calibration** (Mission Planner → Setup → Mandatory → Accel Calibration)
2. **Compass calibration** (Setup → Mandatory → Compass) — do it away from metal benches and tools
3. **Airspeed calibration** — cover the pitot and use *preflight calibration*; verify response by blowing gently into the pitot
4. **RC setup** — bind and calibrate your receiver
5. If you operate multiple aircraft, set a unique `SYSID_THISMAV`

3D-printed parts — print them yourself

The box contains only the board and the silicone tubing. The **supporting parts are not included** — you download and 3D-print them. All files (STL, 3MF print projects and ready-made G-code for the Prusa MK4S) are in your order downloads: **3dlabprint.com → My Account → Downloads**.

Part	Material	Notes
Pitot-static tube	**Non-foaming TPU, harder grades** — factory-tested with **Shore 55D**	Softer TPU (85A-class) deforms and shifts the static reading; foaming TPU is not airtight. Print solid.
NavPack mount / holder	**PETG** (recommended) or PLA / PolyAir	Holds the board in the airframe
Autolaunch START button	**PLA**	Mounts on the airframe, wires to the START input

Printing notes:

- Use the supplied 3MF projects — print settings (walls, infill, orientation) are already configured
- Pitot: print seam position matters for airtightness — keep the 3MF defaults; after printing, verify both pressure channels are clear and leak-free before connecting the silicone tubing
- Mount: for hot environments (car in summer, dark airframes in sun) prefer PETG over PLA

Install the board in the mount, connect the pitot tubing (pitot = dynamic port, static = static port) and the START button **before** the calibrations in *Before the first flight*.

Troubleshooting

Symptom	Fix
DFU device not detected	Use a **data** USB-C cable (not charge-only); try another USB port; re-do the BOOT sequence
DFU not detected (Windows)	Install the DFU driver via Zadig (WinUSB for *STM32 BOOTLOADER*) if CubeProgrammer doesn't see the device
"Error: Target not held in reset" / connect fails	Click Connect again; replug in DFU mode
Board does not boot after flash	Verify start address was `0x08000000` and the file is the full-2MB image for **your board revision**; reflash
Mission Planner doesn't connect	Pick the correct COM port (the board exposes two — use the first), baud 115200; wait ~10 s after plugging in

Appendix — command-line flashing with dfu-util

For automated or headless flashing, `dfu-util` (dfu-util.sourceforge.net, on macOS: `brew install dfu-util`):

```
# check the board is visible in DFU mode
dfu-util -l

# flash the full image and run it
dfu-util -a 0 -s 0x08000000:leave -D 3DFENSE04_H743-Rev106-D20C_<githash>_full2MB.bin
```

A trailing `dfu-util: Error during download get_status` message after "*File downloaded successfully*" is **normal** — the board resets into the new firmware before the final status read.

3DFENSE — Navigation. Power. Launch. One board.