CSV Flight log format

Altimeter Cloud

Every flight is saved to the Nano's drive as a flight log named `flightlog…-N.aclz` (the number is the flight number), with a matching `flightinfo…-N.html` page beside it summarising the flight. By default the log is an **ACLZ** file, the Nano's own compressed format; if you set `log_format` to 0 in the settings it is a **CSV** file instead. This page describes both, and the columns you get from each.

> **TIP**
>
> The easiest way to look at a flight is to upload the file straight to Altimeter Cloud. You do not need to make sense of the raw CSV yourself: the website lays your flight out in clear charts and figures that are simple to read. This page is really for anyone who wants to work with the raw data directly. There is a full walkthrough on the Altimeter Cloud page.

Why ACLZ is the default

ACLZ is the Nano's own compressed binary format, and we recommend leaving it as the default. It records exactly the same flight as a CSV, to the same precision, at roughly a tenth of the size, and that difference in size is what makes everything else possible:

- **Longer flights.** Up to 38,000 samples in one log, against 22,000 for a CSV (20,000 with `include_pressure`). At 400 Hz that is the difference between capturing a long descent in full and running out of log.
- **Far more flights on the drive.** A typical flight is a few tens of kilobytes as ACLZ, so the Nano keeps up to 50 flights before it has to delete the oldest. As CSV, around ten typical flights fill the drive, and a full-length CSV log takes almost all of it on its own.
- **More data, not less.** Every ACLZ flight also keeps the **raw altitude**, exactly as the barometer measured it before any processing, and the **pressure**. So you can always see what TrueFuse and TruePath did, and the Altimeter Cloud can chart the raw trace beside the processed one. A CSV written by the Nano has no room for the raw altitude.
- **Nothing is lost in the compression.** The CSV you download from the Altimeter Cloud is exactly the CSV the Nano would have written, digit for digit.
- **Signed twice.** The Nano signs the file itself and the CSV it expands to, so a CSV exported from the Altimeter Cloud can still be verified as genuine on its own, without the original file.
- **Quicker to handle.** Small files copy off the drive and upload faster, which you notice most when uploading straight from the Nano over USB.

The one trade-off is that you cannot open an ACLZ file in a spreadsheet. Upload it to the Altimeter Cloud (free, and no account needed) to see the charts and download a CSV, or open the flight info page saved beside it for a quick look straight off the drive with no internet. The format is fully documented, with a reference converter, if you want to read it with your own software. Only choose CSV if you specifically need to open logs straight off the drive in a spreadsheet, and your flights are short enough to fit.

	ACLZ (default)	CSV
Samples per flight	Up to 38,000	Up to 22,000 (20,000 with `include_pressure`)
Size of a typical flight	A few tens of KB	A few hundred KB
Flights kept on the drive	Up to 50	Around ten typical, one full-length
Raw altitude column	Yes	No
Pressure column	Yes	Rev1 to 3 always; Rev4+ only with `include_pressure`
Opens in a		

spreadsheet	No: upload it, then download a CSV	Yes
Signature	Ed25519, over the file and over the CSV it expands to	Ed25519, over the file

How ACLZ works inside, and how to read one with your own software, is covered in the software manual.

How a CSV log is laid out

A CSV, whether written by the Nano or downloaded from the Altimeter Cloud for an ACLZ log, has three parts, in this order: a block of header rows holding a summary of the flight and the exact conditions and settings it was recorded under, then a single blank row, then one row of column names starting with `Time(ms)`, followed by the flight data, one row per recorded sample. If you are writing a parser, find the row that starts with `Time(ms)` rather than counting rows, and treat everything from there onward as the data table.

The header is arranged as label and value pairs sitting side by side in columns, so it reads tidily when opened in a spreadsheet: a pair on the left, a pair in the middle, and on Rev4 and later a further pair on the right carrying the motion related figures.

The header

The header carries everything you need to understand the flight without reading a single data row. The fields fall into a few groups.

The **flight summary** gives the apogee in metres and the time it happened, the landing height and time, the maximum upward velocity, the landing velocity, and the average samples per second. The **identification** fields are your device tag, your competitor tag, and the altimeter's serial number, which are the details you use to add a flight to your account at altimetercloud.com.

The **conditions and settings** cover the sea level pressure and static temperature used to work out height, whether sensor temperature was used, the board temperature at the start and end of the flight, the battery voltage at the start and end, the firmware version, the board revision, and a number of individual settings such as orientation, launch protection, hybrid mode, sample speed and recording stop. On current firmware the header also records which versions of TrueFuse and TruePath processed the flight and whether TrueFuse was applied, the launch rail angle and whether the altimeter moved in its mount, the accelerometer calibration quality and any sensor saturation. There is also a single *Settings string* field that packs the complete configuration used for the flight into one line, and an *Adjusted log* field that reads "no" for an original flight and tells you if the log has since been re-derived with different pressure or temperature.

Because the Settings string is written in shorthand, here is what each key holds. The configurable ones match the options described on the settings page.

Key	Holds
fp	Sea level pressure used to work out height (hPa)
ft	Static temperature used (°C)
ut	Whether sensor temperature was used (0 or 1)
ss	Sample speed (Hz)
sr	Sample ratio
ms	Maximum samples
ld	Launch detect threshold (m)
lp	Launch protection (mG)
sl	Startup lock

rs	Recording stop
mp	Maintain power
hm	Hybrid mode
or	Orientation
se	Sync enable (Rev4 and later)
os, ii, kf	Internal pressure handling settings (oversampling and filter parameters)
rcp	The ground reference captured at launch
rmi	Launch time, in milliseconds
cgx, cgy, cgz	Stored gyro calibration offsets (Rev4 and later)
cax, cay, caz	Stored accelerometer calibration offsets (Rev4 and later)
zbb	Internal test flag, always 0 on a normal flight
sda, sdm, sdn, sds	Start-up ground reference: the trimmed average and median altitude of the pad readings, how many were taken, and their spread (cm)
sbf, sbt	The pressure sensor's first reading after power-on and its reading at the end of its warm-up (hPa)

Finally the **Verification hash** field holds the log's Ed25519 digital signature, which lets anyone confirm the file has not been altered (logs from firmware before 1.52 carry an HMAC-SHA256 hash here instead). On Rev4 and later the right hand side of the header adds a set of **motion statistics**: burnout and ejection times, burnout and descent velocities, the maximum and average accelerations during ascent, descent and the burn, the launch pitch, roll, yaw and tilt, the dominant axis, and timing details.

> **IMPORTANT**
>
> Do not edit a flight log if you want to keep it valid. The verification signature is worked out from the exact contents of the file, so changing anything at all, even a single number or an added space, means it no longer matches. Altimeter Cloud only accepts original, unedited logs, so an edited file will fail to verify and cannot be uploaded. If you want to work with the data, copy the file first and edit the copy, and keep the original safe.

The data columns (Revision 4 and later)

After the blank row and the row of column names, each row is one recorded sample. The columns appear in this order:

Column	Units	Format	What it is
Time(ms)	ms	2 decimals	Time from launch, with launch as zero
Altitude(m)	m	3 decimals	Height above the launch point, after TrueFuse and TruePath
Velocity(m/s)	m/s	2 decimals	Vertical speed, worked out from the processed altitude
acceleration_x(mG)	mG	whole numbers	Acceleration along the board's X axis
acceleration_y(mG)	mG	whole numbers	Acceleration along the Y axis
acceleration_z(mG)	mG	whole numbers	Acceleration along the Z axis

acceleration_total(mG)	mG	1 decimal	Overall acceleration magnitude with the 1 G of gravity taken out, so it sits near zero at rest
gyro_x	°/s	whole numbers	Rotation rate about the X axis
gyro_y	°/s	whole numbers	Rotation rate about the Y axis
gyro_z	°/s	whole numbers	Rotation rate about the Z axis
Board temp(C)	°C	2 decimals	Sensor temperature, written on every 50th row and left blank on the rows in between
pitch	degrees	2 decimals	Orientation angle from the fusion filter
roll	degrees	2 decimals	Orientation angle from the fusion filter
yaw	degrees	2 decimals	Orientation angle from the fusion filter
tilt	degrees	2 decimals	Angle away from vertical
Pressure(hPa)	hPa	6 decimals	Air pressure for the processed altitude, worked out with the standard atmosphere. Always in an ACLZ log; in a Nano-written CSV only with `include_pressure` set to 1
Raw altitude(m)	m	3 decimals	Height above the launch point exactly as the barometer measured it, before any processing. ACLZ logs only

A few things are worth knowing. Time is measured from launch, so launch is zero; because the Nano keeps a buffer of samples from before lift off, the first rows show negative times. From firmware 2.00 the time is written to two decimal places of a millisecond, because at 400 Hz the samples are 2.5 ms apart; earlier firmware wrote whole milliseconds. The acceleration and gyro axes are the board's own axes, which line up with the rocket according to your orientation setting. Every data row ends with a trailing comma, so a strict parser will see one empty field at the end of each row.

Altitude and Raw altitude share the same zero, so you can plot one against the other and see exactly what the processing changed: the two lie on top of each other for most of the flight and part only where the launch, an ejection charge or a pressure spike upset the barometer. The Altimeter Cloud can show the raw altitude as its own line on the flight chart.

The row of column names for each case, exactly as it appears:

```
CSV from an ACLZ log (Rev4 and later)
Time(ms),Altitude(m),Velocity(m/s),acceleration_x(mG),acceleration_y(mG),acceleration_z(mG),acceleration_total(mG),gyro_x,gyro_y,gyro_z,Board t

CSV written by the Nano (Rev4 and later; Pressure(hPa) only with include_pressure 1)
Time(ms),Altitude(m),Velocity(m/s),acceleration_x(mG),acceleration_y(mG),acceleration_z(mG),acceleration_total(mG),gyro_x,gyro_y,gyro_z,Board t

CSV from an ACLZ log (Rev1 to 3)
Time(ms),Altitude(m),Pressure(hPa),Velocity(m/s),Board temp(C),Raw altitude(m)

CSV written by the Nano (Rev1 to 3)
Time(ms),Altitude(m),Pressure(hPa),Velocity(m/s),Board temp(C)
```

ACLZ logs from older firmware, made before the raw altitude column was added, expand without it.

Revision 3 and earlier

Boards without the motion sensor use the same file in the same shape, just simpler. The motion columns, that is the three acceleration columns, the total acceleration, the three gyro columns and the pitch, roll, yaw and tilt, are absent, and the header does not carry the motion statistics on its right hand side. The Pressure(hPa) column sits third, straight after Altitude, in both

formats, and an ACLZ log adds Raw altitude(m) at the end. Everything else on this page, the header groups, the time being measured from launch and the temperature on every 50th row, applies in the same way.