Altitude & apogee calculation

Altimeter Cloud

The Jupiter flight computer manual · v1.0

https://www.altimetercloud.com/support/jupiter/altitude-apogee-calculation/

Altitude & apogee calculation

Jupiter's altitude comes from its barometric pressure sensor, sampled continuously through the flight and converted to height using real atmospheric conditions rather than textbook assumptions. This page explains the conversion, the two ways the atmospheric reference can be set and why the automatic weather system is the better one, how apogee is detected, and why the answer is barometric rather than GNSS.

From pressure to altitude

Air pressure falls predictably with height, so measuring pressure precisely means knowing altitude precisely, provided you know two things about today's atmosphere: the sea-level pressure and the air temperature. Jupiter uses the standard hypsometric conversion:

$$\text{altitude} = ((T + 273.15) / 0.0065) \times (1 - (P / P0)^{0.190266669})$$

where P is the measured pressure, P0 is sea-level pressure and T is the ambient temperature in Celsius. Both reference values matter more than people expect. Near sea level, one hPa of pressure is about eight metres of altitude, and real sea-level pressure routinely sits tens of hPa away from the textbook 1013.25 as weather systems pass, so an unreferenced altimeter can be hundreds of metres wrong about absolute altitude before it leaves the ground. Temperature scales the entire height column: every degree away from the standard 15 °C stretches or shrinks all computed heights by about 0.35 percent, which on a cold winter day means apogee readings around five percent short of the truth even when the launch pad reads perfectly. The raw conversion is then smoothed by a Kalman filter seeded from the first real sample, so the reported altitude is stable against sensor noise without lagging real motion.

The altitude reference: automatic weather or fixed

The reference values can come from two places, chosen by the weather_mode setting. The fixed system uses the values you set: a sea-level pressure (default 1013.25 hPa) and a temperature (default 15 °C). It needs no connectivity and behaves identically everywhere, which makes it the right choice if you want strictly repeatable numbers between flights, but it inherits every error described above whenever the real atmosphere disagrees with your settings.

The automatic weather system is why Jupiter carries a modem to the pad. Once it has a network connection and a GNSS fix, the device requests a forecast for its actual location from altimetercloud, receiving several hours of sea-level pressure and temperature, authenticated to your specific device. The forecast is cached on the device for three hours and refetched automatically if you move more than five kilometres, so it survives patchy signal at the launch site. The reference stays live right up to launch, then freezes for the flight, so conditions drifting through a long pad wait are absorbed but nothing changes mid-flight. One protection is worth knowing about: a weather update shifts the altitude baseline, so Jupiter only ever applies one while the airframe has been completely still for a sustained period, guaranteeing a launch can never coincide with the ground moving underneath its own recording.

> **Fixed reference**
>
> Uses your configured sea-level pressure and temperature. No connectivity needed, perfectly repeatable, but wrong by whatever today's weather disagrees by: tens of hPa of pressure is hundreds of metres of absolute error, and temperature skews all heights by ~0.35% per degree.

> **Automatic weather**
>
> Fetches real sea-level pressure and temperature for your actual location, hour by hour, authenticated to your device. Cached for three hours, refreshed if you travel, live until the moment of launch and frozen through the flight. Accurate absolute altitude and correctly scaled heights.

Detecting apogee

Apogee detection runs on the filtered altitude and asks a simple, robust question: over a sliding look-back window, is virtually every sample-to-sample change non-positive? A rocket still climbing cannot satisfy that for long, and a rocket past its peak cannot fail it, so sustained descent is the trigger. When it fires, Jupiter reports where the peak actually was, not where the detection happened: the peak's altitude and timestamp were tracked all the way up, so the recorded apogee is the true top of the flight rather than a point some fraction of a second down the far side.

Just as important is when detection is not allowed to run, because most false apogees are prevented by the gates rather than the test. Detection cannot fire before motor burnout: while the motor is burning, apogee is a physical impossibility, so the detector simply is not consulted. For flights where no clean burnout was identified, a fallback gate holds detection off until at least 3.5 seconds after launch and until the measured acceleration has dropped below 3 g, so a still-thrusting rocket can never be judged. Detection is also suspended during pressure-spike lockouts: an ejection charge firing inside the airframe produces a pressure transient that momentarily corrupts the indicated altitude, and Jupiter recognises these excursions and holds all live altitude decisions until the reading is trustworthy again, so a deployment event can never be misread as the rocket descending.

The detection window itself adapts to the flight. It opens at motor burnout at its widest, 1.5 seconds, because a motor cutting off disturbs the pressure field around the airframe and can briefly dent the indicated altitude, and demanding sustained descent across a wide window rides that transient out. Over the following 1.1 seconds the window narrows linearly to its normal 400 milliseconds, trading that caution back for a fast, confident call once the transient has passed. After landing, the recorded log is refined once more: the saved flight carries both the live apogee the rules acted on and a post-flight recomputation from the cleaned altitude data, and the two are kept separately because each is a fact about the flight in its own right.

High altitude: the acceleration backup

Barometric detection has a physical ceiling. Above roughly nine kilometres the air is thin enough that each metre of height changes the pressure by very little, and the sensor is approaching the edge of its measurement range, so the descent signature that apogee detection relies on becomes progressively weaker. For flights in that regime, Jupiter uses its acceleration sensing as a backup witness for apogee, tracking the flight inertially so that the top of the trajectory can be called even where the barometer alone would struggle. Below that ceiling the barometer remains the primary and the most precise instrument for the job.

Why not GNSS?

Jupiter tracks GNSS altitude continuously and records it, but deliberately does not use it for apogee detection, for two reasons. First, precision: GNSS vertical accuracy is typically several metres at a low update rate, against centimetre-scale resolution at high rate from the barometer. Second, and more important, reliability at exactly the wrong moment: GNSS receivers are designed around dynamic limits, and the acceleration of a rocket under boost, beyond 5 g, is outside what they track, so the receiver can lose lock during the very phase of flight that precedes apogee. Reacquiring a fix takes time, and there is no guarantee it completes before the peak. A detection system that can be blind at the moment it matters is not a detection system, so GNSS altitude is treated as valuable recorded data and a cross-check, never the trigger. Deployment decisions ride on the barometer, with inertial backup where physics demands it.