SENSOR FUSION

TrueFuse

The barometer tells the truth about altitude, except for the few moments each flight when it doesn't. TrueFuse finds those moments as they happen, bridges them with inertial data, and hands control straight back. It can decline to trust the barometer. It can never invent an altitude the barometer did not support.

RUNS	INPUTS	LOOK-AHEAD	FAILURE MODE
Live and at save	**Baro, accel, gyro**	**None**	**Decline to raw**

What goes wrong with a barometer

A barometric altimeter measures static pressure and converts it to height. For most of a flight this is the most trustworthy altitude source on the board: it does not drift, it needs no initial condition, and it is immune to the errors that accumulate in an integrated accelerometer.

For a few seconds of that same flight, it is badly wrong. Three events dominate:

- **Ejection.** The charge pressurises the airframe and the port sees a pressure step that has nothing to do with altitude. On the chart it appears as a sharp dip or spike, tens of metres deep, lasting a few hundred milliseconds.
- **Launch.** The motor plume and the sudden airflow over the vent holes disturb the port before the rocket has left the rail, and the recorded altitude can climb, hesitate or hump while the real altitude is still near zero.
- **Poorly sized or blocked vents.** The internal volume lags or steps behind the external pressure, so the trace acquires a staircase shape under a smooth flight.

An accelerometer is unaffected by all three, but integrating it twice turns a small bias into a large altitude error within seconds. Neither sensor is good enough alone. TrueFuse uses the barometer as the reference and the accelerometer only to carry the altitude across a window where the barometer has been shown to be untrustworthy.

FIG. 1 A real ejection event. The pressure port reports a 40 m descent and recovery in a third of a second while the airframe is still coasting. TrueFuse opens a window at the onset, carries the altitude across it inertially, and rejoins the barometer once it agrees again. Apogee, which occurs after the window closes, is measured from the barometer as usual.

Staying live through a pressure event

This is the practical reason TrueFuse exists, rather than an accuracy improvement measured after the fact.

An altimeter that trusts pressure alone has exactly one safe response to a pressure event: stop believing itself. The charge fires, the trace dives forty metres in a third of a second, and every decision keyed to altitude is suddenly reading a number the airframe never flew. Deployment at a set height, airbrake control, staging, altitude triggered outputs, telemetry state, all of them are looking at the same corrupted value.

The conventional answer is a lockout: suspend altitude driven decisions for a fixed blanking period and wait for the sensor to settle. It works, and it costs something real. The blanking lands at the busiest moment of the flight, its length has to be chosen in advance for the worst charge the design might ever see, and so every flight pays for the worst case. During that window the flight computer is flying blind by policy, not because the information was unavailable.

TrueFuse removes the reason for the blanking. The altitude is carried across the window rather than abandoned, so the vehicle still has a believable height and vertical speed while the port is recovering, and the window ends when the barometer genuinely agrees again rather than when a timer expires. Decisions can keep running through the disturbance instead of being parked for it.

> **Marked, not hidden.** Every window is recorded with its start and end, so anything that should still behave cautiously during a disturbance can be told exactly when one was open. The difference TrueFuse makes is that caution becomes a choice rather than the only option.

Signal chain

Every stage below runs on the flight computer. Nothing is reconstructed on the ground, and nothing in the live channel uses a sample that had not yet been taken.

FIG. 2 Conditioning happens before any fusion decision: the channels are time aligned, the accelerometer scale is measured on the pad, the gyroscope establishes which way is up, and saturated samples are marked. The grammar then decides, sample by sample, whether the barometer is believable. The red path is the decline route, taken when the flight itself fails a validity test.

Window grammar

TrueFuse holds one of four states. The state names appear in the diagnostics and describe exactly what the filter was doing at that moment in the flight.

FIG. 3 The four states in flight order. A flight may open several windows, launch and ejection being the common pair, and the count is reported in the log header as TrueFuse windows.

Inside a window

While a window is open the altitude is carried forward from the accelerometer, projected onto the tracked world vertical, with a closed form ballistic continuation used when the motor has stopped and the vehicle is coasting. The carry is anchored to the altitude and velocity that were valid at the moment the window opened, so the error inside a window grows only with the duration of that window, typically a few hundred milliseconds.

Windows have a bounded life. If the barometer has not become believable again within the time limit, the filter stops carrying and returns to the barometer regardless, because a carry that runs long has no way to prove itself right.

DESIGN CONTRACT

The barometer is the reference. The accelerometer never sets the altitude, it only carries it between two barometric anchors.

No look-ahead in the live channel. Every decision uses samples already taken. A window is opened on evidence available at the time, not on knowing how the trace ends.

Declining is always available. Every guard in this document resolves to the same safe action: stop fusing and output the barometer.

Sensor synchronisation

Fusing two sensors requires knowing which instant each sample describes. A barometer sample is old by the time the processor sees it: the ADC averages pressure over a conversion, the result waits to be read, and the sensor's own filter delays the ramp further. An accelerometer sample is comparatively fresh. Comparing them without correcting for this makes a perfectly good barometer look like it disagrees with a perfectly good accelerometer.

The delay is not measured from flight data. It is a property of the configuration and is calculated from it:

$$\textbf{sync} = t_{conv}/2 \;+\; T_{odr}/2 \;+\; c_{iir} \times T_{odr} \;+\; t_{kalman} \;-\; t_{imu}$$

- `t_conv/2` the ADC integrates over the conversion, so the value describes the middle of that interval.
- `T_odr/2` mean age of the most recent completed conversion when the processor reads it.
- `c_iir × T_odr` the sensor's internal IIR filter delays a ramp by exactly its coefficient in sensor samples, not in loop samples.
- `t_kalman` the altitude smoothing filter's own ramp lag, the only term that varies with speed, and the smallest.
- `t_imu` the accelerometer's own small conversion and filter delay, subtracted because it applies to the other side.

CALCULATED BAROMETER DELAY, MILLISECONDS, AT THE 50 M/S REFERENCE

SENSOR	CONFIG	FIXED PART	SMOOTHING OFF	LIGHT	MEDIUM	HEAVY
BMP581	80 Hz, os×16, iir 7	99.4	95.8	106.2	111.2	118.4
BMP390	50 Hz, os×8, iir 7	159.5	155.9	171.7	179.0	189.3

The correction is applied by delaying the inertial channel until it is as old as the barometric one. Both channels then describe the same instant, which is what fusion requires. A consequence worth knowing when reading a log: every timestamp in it trails the physical event by the barometer's chain delay, uniformly, so intervals and rates are correct even though absolute instants are late.

FIG. 4 Where the barometer's delay comes from. The sensor's internal IIR filter dominates: at a coefficient of 7 it alone contributes 87.5 ms on a BMP581 running at 80 Hz and 140 ms on a BMP390 at 50 Hz. Lowering the IIR setting shortens the delay and raises the noise, which is the trade the setting exists to make.

Scale calibration

An accelerometer at rest measures exactly one gravity. Its reported magnitude at rest is therefore a direct measurement of its own scale error, and TrueFuse uses that rather than assuming the nominal scale is correct.

On the pad, any sample with a plausible rest magnitude and a quiet barometer seeds a running estimate. Later samples join the estimate if they sit within 6 percent of it, or 60 mG, whichever is larger. The calibration factor is then the ratio of true gravity to the measured rest magnitude.

The alternative, a fixed acceptance band around 1000 mG, fails silently on a unit whose scale is off: it admits no samples at all, the factor stays at one, and the filter integrates an accelerometer that is reporting the wrong number. A flight in the reference set rested at 2050 mG for exactly this reason, and its altitude was manufactured, not measured, until the calibration was made a measurement.

What this means in practice. Nothing to configure. Leave the altimeter still for a few seconds before launch, which the pad detection requires anyway, and the calibration takes care of itself. The measured rest magnitude is reported in the diagnostics so an out of specification sensor is visible rather than hidden.

World vertical

An accelerometer measures force along its own axes, which are fixed to the airframe. To contribute to an altitude, that force has to be projected onto the direction that is actually up, and the airframe rotates during flight.

TrueFuse establishes the vertical from gravity while the vehicle is at rest, then rotates that vector with the gyroscope for the rest of the flight. Thrust is projected onto the tracked vertical, so a rocket flying at an angle contributes only its vertical component, which is the physically correct behaviour.

FIG. 5 The tracked vertical follows the airframe. If it passes 100 degrees from the pad vertical before burnout, the vehicle has tipped over under thrust, which no ascending rocket does, and the filter is disabled for the whole flight. A vertical that dies after burnout is ordinary recovery tumbling and is treated as a guard, not a verdict.

Guards

Four conditions can stop TrueFuse trusting the accelerometer. Each resolves to the barometer, which is why none of them can produce an inflated altitude.

GUARD	TEST	ACTION
Tip over	tracked vertical past 100 deg before burnout	Decline for the whole flight
Tumble	sustained rotation about a perpendicular axis	Decline from that point
Saturation	any axis at full scale, 5 samples running	Suspend the inertial carry
Window timeout	barometer not believable again in time	Return to the barometer

Saturation, and why a few samples are tolerated

A clipped accelerometer reports its rail, not the real force, so the reading is a floor. Using a floor briefly understates the acceleration and pulls the fused altitude toward the barometer, which is the harmless direction. Hard motors touch full scale for a few milliseconds at ignition, and treating that as a fault would suspend the carry at the exact moment it is most needed. TrueFuse therefore ignores saturation shorter than five consecutive samples, and treats anything longer as a genuine out of range burn where integration would be dishonest.

The clip threshold is derived from the configured measurement range rather than assumed, because the range is a setting.

Live and post channels

TrueFuse produces two altitude series from the same grammar.

The **live** channel is what the flight computer can know in flight. It is strictly causal: at every sample it uses only what had already been measured. This is the channel that matters for anything the altimeter has to act on, it is the only honest basis for an in flight decision, and it is what keeps a usable altitude available through a pressure event instead of forcing a blanking period.

The **post** channel runs once at save, over the same recorded window structure, with one extra piece of information the live channel cannot have: where the barometer ended up after each window closed. Knowing the far side of a window lets the displacement across it be anchored at both ends rather than one, which removes the small residual offset a one sided carry leaves behind. Velocity in the post channel is recomputed with a symmetric stencil, which is why the recorded velocity trace is smoother than the live one.

> **Which one is in my log?** Both. The recorded altitude column is the post channel, because the log is written after the flight and there is no reason to store the less accurate of two available answers. The live channel is what the device used while flying, and it is charted alongside on Altimeter Cloud so the two can be compared directly.

Reading the log header

Every flight records what the filter did, so a chart can be interpreted without guessing.

FIELD	MEANING
TrueFuse	Version, and whether it was applied or declined
TrueFuse windows	How many windows were opened during the flight
TrueFuse repaired (s)	Total flight time spent inside windows
TF declined (s)	Present only on declined flights: when the verdict was reached
Sync offset (ms)	The calculated alignment applied to the inertial channel
T0 source	Whether launch time came from the accelerometer or the barometer
T0 shift (ms)	How far the accelerometer moved the zero from the pressure estimate

A declined flight is not a failed flight. It means the filter judged the fusion unsafe and gave you the barometer, which is exactly what an altimeter without fusion would have recorded.