

Good news: TP1 is **before** the ferrite bead (E8), so pulling E8/E9/E10/E11 already isolated the loads.

That means the problem is inside the buck stage around **U11** → **L1** → **C172/C175** → **R74/R75** → **FB1** (the 2.5 V rail).

From the schematic, this rail is set by the feedback divider **R74 = 15 k** (top) and **R75 = 10 k** (bottom). For an LT8652-class regulator (VFB ≈ **0.97 V**), the target is:

scss

Copy code

$$V_{out} \approx 0.97 \times (1 + 15k/10k) \approx 2.43 \text{ V } (\approx 2.5 \text{ V rail})$$

If you see **~2.3 V and falling**, the regulator isn't reaching regulation. Do the checks below in order (no loads attached):

#### 1. Measure key pins (DC)

- **PVIN1/2** of U11 → should be stable and comfortably above the UVLO (e.g., >5 V if fed from 5–12 V).
- **FB1** (the feedback pin tied to R74/R75) → when healthy it sits at **~0.97 V**.
  - If FB1 is well below 0.97 V, the loop is being limited (by soft-start/TRKSS, UVLO, or a fault).
- **TRKSS1** (soft-start / tracking) → must ramp **above 0.97 V**. If it stays near ~0.8–0.9 V the output will be clamped to ~2.2–2.3 V forever.

#### 2. Verify the divider values actually stuffed

- In-circuit DMM reading can mislead; quickly lift one end or read with the IC unpowered. If **R75 was accidentally fitted as 12 k** (common mix-up), Vout becomes ~2.2–2.3 V.
- Quick experiment: clip a **~200 k** resistor **in parallel with R75**; if Vout rises toward 2.5 V, the bottom leg is too large (wrong value or bad solder).

#### 3. Rule out a local short/leaky output cap (still on the regulator side even with beads removed)

- Measure resistance from **TP1 to GND** with power off. If it's just a few ohms, briefly lift **C172 or C175** or one side of **L1** to see if the rail pops up—replace the bad part.

#### 4. Check enable/UVLO network

- Ensure **EN/UVLO** pin on U11 is > its enable threshold during startup. If it's marginal, the chip can repeatedly enter UVLO and the output will droop.

#### 5. Look at SW1/inductor with a scope (if available)

- You should see clean switching. No switching or very narrow pulses → UVLO/SS limiting. Continuous current-limit behavior → shorted output part.

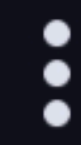
#### 6. Comp/Feed-forward parts

- **C171 (4.7 pF)** is a feed-forward cap across R74. If shorted or mis-stuffed (e.g., tens of nF), the loop will be wrong and the rail can sit low/unstable.

#### Most likely culprits (from experience on this exact topology)

- **R75 populated with the wrong value** (e.g., 12 k instead of 10 k) → ~2.3 V.
- **TRKSS1 never rises above ~0.97 V** (missing/shorted SS cap or pulled low) → output clamps near 2.3 V

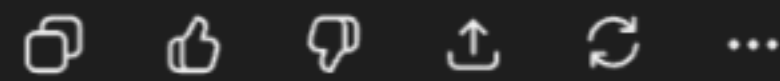




clamps near 2.3 V.

- **Leaky/shorted C172/C175** → regulator stuck in current limit.

Do the FB1/TRKSS1 checks first; they tell you immediately whether it's a divider/SS issue (FB  $\approx$  0.97 V when fixed) or a loading/UVLO problem. If you can share the readings for **PVIN, EN/UVLO, TRKSS1, FB1, TP1**, I'll pinpoint the next action.



After this when we run it shows calibration failed in tera term , we using altera chip

Thought for a few seconds

Thanks — useful info. If the board shows “**calibration failed**” in Tera Term after you fixed/checked TP1, that typically means the FPGA or its connected analog blocks (ADCs/PLLs/SerDes) didn't see correct supplies, clocks, configuration data, or a required reference during the calibration routine. Below is a prioritized, practical checklist and tests to isolate and fix it for an **Altera/Intel FPGA** system.

Read through and run the checks in order — each step is quick and rules out a major class of failure.

## 1) Re-check power rails & sequencing (highest chance)

- The FPGA needs several rails (VCCINT, VCCA, VCCIO, VREF/VREFH etc.). A *single missing/low* rail will make calibration fail.
  - Measure DC voltages (power off → measure resistor values if needed; power on → measure live): **VCCINT, VCCA1/VCCA2 (1.2/1.8/2.5 as used), VCCIO\_x** and **VREF**.
  - Confirm TP1 (you mentioned ~2.3 V) — what rail does TP1 feed? If it should be 2.5 V, any ~2.3 V means VREF or VCCA is low → calibration will fail.
- Check supply sequencing if your FPGA requires a sequence. Some Altera devices need VCCINT before VCCIO or specific order.

## 2) Check FPGA configuration signals & status

- Watch the FPGA config pins during power-up:
  - **nCONFIG / nSTATUS / CONF\_DONE**: nCONFIG should be released, nSTATUS should go high and stay high, and CONF\_DONE should go high once configured. If any of these are stuck low or toggling, configuration is failing.
- If configuration fails, the calibration routine (which runs from the loaded design/firmware) never starts or gets incomplete data.

## 3) Verify the clock(s)

- The calibration routine often needs reference clocks (oscillator/clock PLL).



- Probe the primary oscillator with a scope; confirm correct frequency and stable square wave at the FPGA clock input.
- If the clock is missing, PLLs or ADCs don't lock → calibration fails.

#### 4) Serial/TERATERM settings & boot messages

- Confirm serial parameters match the firmware: common is 115200, 8, N, 1, but your project might use something else.
- Reboot and capture the full boot text. Look for earlier errors (bootloader, CRC, missing files, filesystem errors).
- If messages show "FPGA not configured" or "device ID mismatch" that's a separate root cause.

#### 5) JTAG / reprogram FPGA directly

- Connect Quartus Programmer (or iMPACT) via JTAG and try to program the FPGA directly from PC.
  - If JTAG programming fails, the issue is hardware (power, JTAG wiring) or a dead device.
  - If it programs OK and calibration still fails, it's likely a config file, peripheral, or analog issue.

#### 6) Check peripheral power/refs used by calibration (ADC, PLL, mixed-signal)

- If calibration touches ADCs, PLLs, SERDES, or I/O banks that need VREF or 2.5 V, ensure those rails are present and within tolerance.
- Check any external ADC/clock ICs used in calibration — are they powered and responding (I2C/SPI)?

#### 7) Reset & watchdog / enable pins

- Ensure **FPGA enable** or any regulator EN pins are asserted. A disabled regulator could show partial voltage.
- Check reset lines to the MCU or calibration controller: stuck in reset = no calibration.

#### 8) Check EEPROM / flash / calibration data

- If calibration requires a calibration table stored in EEPROM/Flash, ensure that memory is present and readable:
  - Use I2C/SPI tools (or the firmware debug output) to confirm the board can read the device and the expected data is present.
  - A missing/corrupted calibration table will produce "calibration failed".

#### 9) Look for hardware damage or mis-populated parts

- Re-check R/C values in the feedback network (we discussed R74/R75 earlier). Wrong resistor value can make rails slightly off and confuse analog blocks.
- Replace or lift suspicious caps (output caps you already checked) or remove suspect shorted components.
- Check solder joints on the FPGA power pins and the surrounding nets.

#### 10) Enable verbose/debug logging in firmware

- If firmware has a verbose or debug mode, enable it and capture logs. Look for the point where calibration fails (e.g., "PLL lock failed", "ADC init failed", "VREF out of range").