Engineering software you can prove, not just trust.

The number one requirement for engineering software is trust. Every calculator on Calcs.com is built to be inspected, checked, and held accountable - so you can defend every result that leaves your desk.

01 / Promise

Full transparency

Every formula in every calculator shows its specific reference in the relevant code or standard. No black boxes. Where a formula can't be shown, the underlying data and source code are open for inspection.

02 / Promise

Ruthless checking

Even the smallest change is run through millions of automated tests and peer reviewed. New calculators are independently checked by external domain experts and released as Beta for user review first.

03 / Promise

Safety first

If an issue could lead to an unsafe design, we contact affected users directly and tell you exactly which projects may be impacted, so you can act as fast as possible.

THE RELEASE PIPELINE

Every change passes four gates before it reaches your projects

1 Automated testing

Dozens of test cases per module run on every update, including historical versions. More than a million tests per minute change. Differences down to a billionth of a percent are flagged for engineer review.

2 Internal peer review

A second engineer compares output against the standard, runs hand calculations, enters typos and impossible scenarios, and investigates every unexpected result. Approval is required before release.

3 External independent review

For new calculators and major changes, an outside engineer with subject-matter expertise checks the calculator against their own work. Sign-off only happens when they would stamp a design produced by it.

4 Beta status

Major releases ship with a visible Beta tag so users can preview and provide feedback. The tag is removed only when our QA process is complete, and Beta designs carry the tag until upgraded.

WHEN SOMETHING SLIPS THROUGH

Three severity levels, all higher priority than new features

MINOR — **No effect on results.** Typos, overlapping text, unclear descriptions, incorrect references. Fixed within days. Safe to ignore.

BUG — **Obvious failure or always conservative.** Error bars, zero demands, overly aggressive checks. Top priority. Calculator may be rolled back while a proper fix is developed.

CRITICAL — **Possibility of an incorrect, unconservative, non-obvious result.** Could affect design safety. Triggers the response below.

IF A CRITICAL BUG IS EVER FOUND

Our commitment to you

- **Stop everything.** All current work pauses. Resources focus on the fix.
- **Revert immediately.** The calculator rolls back to a known-correct version, or is marked Beta.
- **Notify affected users directly.** We narrow the database to projects matching the failure conditions and contact every user.
- **Force the upgrade prompt.** Once fixed, every active copy of the calculator prompts users to update.
- **Review and harden.** The team addresses the root cause and adds safeguards so the same issue cannot recur.