

This sprint used ChatGPT as a helper to design a 48 V solar battery pack for a remote IoT node. The work covered figuring out the load profile, planning for three days without sun, and checking cable losses. It showed how AI can be really useful for getting started with designs and calculations, but also reminded us that proper engineering checks are still essential.

Prompts that saved Time

Some prompts turned out to be especially helpful. For example, the “Plan” prompt gave me a clear ten-step roadmap with inputs, outputs, and tests, which I was able to drop straight into my README.md. That gave the project a solid starting structure.

Traps Hit:

- Assumed perfect battery efficiency ie ignored how DoD affects lifespan and usable capacity.
- Designs looked fine in theory but would underperform in real use.
- Not being clear about units which caused small conversion errors that were hard to catch

Mistakes ChatGPT made

The battery size was underestimated because the calculation didn't factor in the depth of discharge (DoD) needed to keep the battery healthy. As a result, the capacity looked smaller than what would actually work in practice.

The cable loss estimates were also too simple. They didn't take into account how temperature affects cable resistance, which matters a lot in outdoor or remote setups where conditions can swing quite a bit. That could mean more voltage drop and power loss than expected.

Guardrails

I will make sure to spell out all the key design parameters before asking for code or calculations. Parameters like DoD, battery type and efficiency, expected temperatures, and cable properties. I will also insist on having units and sources for constants. Finally, I'd make environmental edge-case testing a must, since real-world conditions can push systems beyond the average assumptions.