PREDICTABLE DESIGNS

9 ESP32 DESIGN MISTAKES

THAT SABOTAGE YOUR PRODUCT

MISTAKE 1: CHOOSING THE WRONG ESP32 VARIANT

- ✓ Map out product needs before selecting a variant (Wi-Fi, Bluetooth, GPIOs, RAM, Flash, peripherals).
- ✓ Avoid defaulting to the most powerful or cheapest option.
- ✓ Check lifecycle status and availability before committing.
- ✓ Choose a variant with enough headroom, but without unnecessary overhead.

MISTAKE 2: NOT OPTIMIZING POWER CONSUMPTION

- ✓ Use deep sleep modes whenever possible.
- ✓ Disable Wi-Fi and Bluetooth when not needed.
- ✓ Define duty cycle and select low-power components early in design.
- ✓ Measure current draw in both active and sleep modes.
- ✓ Consider single-core variants for lower power needs.

MISTAKE 3: USING BARE CHIPS INSTEAD OF CERTIFIED MODULES

- ✓ Prefer certified modules unless producing hundreds of thousands of units.
- ✓ Avoid full RF layout and certification burden without in-house expertise.
- ✓ Certified modules reduce risk, speed time-to-market, and simplify compliance.

MISTAKE 4: BAD ANTENNA PLACEMENT OR DESIGN

- ✓ Follow Espressif's antenna placement guidelines.
- ✓ Keep antenna near board edge with clear keep-out zone (no copper/ground nearby).
- ✓ Avoid placing antennas near batteries, shields, or enclosure walls.
- ✓ Test final board inside enclosure to confirm wireless performance.

MISTAKE 5: IGNORING HEAT DISSIPATION

- ✓ Add copper pour and thermal vias under module to spread heat.
- ✓ Increase board thickness or use a heatsink if needed.
- ✓ Add venting or airflow in enclosure for high-power use.
- ✓ Perform thermal testing in real conditions early in development.

MISTAKE 6: POOR POWER SUPPLY DESIGN

- ✓ Use a regulator with fast transient response to handle Wi-Fi/Bluetooth spikes.
- ✓ Add bulk capacitors close to ESP32 for current smoothing.
- ✓ Test under real-world peak loads and wake-up transitions.

MISTAKE 7: DESIGNING WITHOUT OTA UPDATE CAPABILITY

- Set up flash partitions for OTA updates.
- Test OTA process thoroughly, including recovery from failed updates.
- Implement OTA early to ensure long-term product flexibility.

MISTAKE 8: NOT TESTING IN THE REAL WORLD EARLY ENOUGH

- Test inside the final enclosure with full firmware.
- Run extended stress tests (heat, cold, drained batteries, interference).
- Test for days, not just minutes, to catch rare issues like memory leaks or resets.

MISTAKE 9: WRONG GPIO USAGE

- Identify fixed-function pins first (ADC, touch, flash, strapping).
- Assign critical interfaces (SPI, I2C, UART) before general tasks.
- Avoid using restricted pins for LEDs or other low-priority signals.
- Consider routing distance and package when assigning GPIOs.

For Entrepreneurs and Startups:

Develop and launch your electronic product FASTER

without costly mistakes!

Doing it alone is slow and risky. Small missteps now can become big problems later. Skip the mistakes and launch faster.

Frustrated things aren't moving as fast as you want?

Feeling unsure what to do next?

Meet your guide: John Teel

Hey there, I'm a former microchip design engineer at Texas Instruments and founder of a hardware startup that sold products in hundreds of retail stores. My chip designs are in devices from Apple, Intel, and more.

Now, my full-time focus is helping people like you bring new electronic products to life, without wasting time, money, or risking everything.

CLICK HERE TO LEARN MORE!