

Electronics HW Design – Debugging and Root Cause Analysis

I have been designing electronic hardware for most of my 25 years of professional life including FPGAs, controllers, and processors. Debugging electronic circuits and performing root cause analysis can be a challenge depending on the symptoms of the circuit, the device, or the system. It can feel like trying to find a needle in a haystack. Where start?

I personally do not believe Universities spend much (if any) time on this subject. And since younger and less experienced Engineers are no longer buddied up with experienced Engineers, they often need to figure this out for themselves. I hope this article may help by providing some possible strategies on how to zero in on root causes. Debugging takes experience, instinct, and practice. Some issues are easier to spot than others.

Document the analysis. It makes it easier to follow a plan for getting to the root cause. Sometimes it is necessary to follow possible root causes and they do not pan out to be **the** root cause. Having this written down can help with making better choices on next steps. I define possible root causes as an interim step to explain the symptom(s) I am observing. They may or may not be the actual root cause.

It is often necessary to gain access to signals and power that may be buried in the PCB. Especially today it becomes more and more difficult to access pins as less and less copper is exposed. Allow for some debugging features in your design even if it means you connect some pads to unused pins that signals can be routed to via FPGA or processor fabric. If necessary, you may need to sacrifice one or more circuit boards to scratch solder resist away, cut traces, or solder wires to nets so you can observe them. Some creativity may be required.

I recently had a situation where some boards were reset by a micro during power-up (brown out). I was quickly able to identify it was a power dip issue caused by a combination of low impedance, high capacitance, and a DCDC converter current limit going into a semi shutdown state during over-current events. The time of the reset, the power dip, and enabling the supply all coincided. The root cause in the end was a power switch used to power up a portion of the supply for the purpose of power sequencing. The fix was finding a different switch with the same footprint, but with a slower soft start (10x). Using some creative soldering, I was able to proof root cause by putting a series power resistor into the switch path and measure the current and voltage dip across that resistor.

Study the Symptom

To find the cause we need to first get a better understanding of the symptom.

- What happens? Describe the event.
- When does it happen? Are there specific circumstances or conditions?
- What possible root causes exist, explaining the symptom? Write down which power rails and signals are involved.
- Set the parameters that most clearly show the symptoms repeatedly. This is important especially when dealing with intermittent issues. Some issues take minutes, hours, or even days to appear. This can easily lead to a false positive determination of the root cause.

Follow the Symptom

Sometimes it is necessary to follow the symptom to a next level, meaning that especially system level issues symptoms may be caused by a series of events purposefully designed for the system, but triggered in the wrong way. Understanding these mechanisms and finding the trigger is necessary to determine and zero in on possible root causes.

Chasing a Possible Root Cause

There are a number of generic steps that can be taken to verify possible root causes. If at any of these steps you find a correlation to the symptom, follow the trail.

- If possible, find a way to trigger the event. This makes it much easier to observe the power rails and signals that could help to identify the root cause.
- With or without the trigger, observe the power rails (best done with a scope, do not use a DVM as it is way too slow) of the devices that create and/or drive the signals causing the symptom. Observe these rails in physical proximity to the signal drivers. Power glitches can cause all kinds of nasty symptoms.
- Check related signals with a scope. Look for signal levels and any glitches between drivers and loads. Verify signals behave the way they should.
- In case of communication between devices it may be required to hook up a logic analyzer or scope to check the data across the bus. Again, the trigger can help to determine the point in time when the symptom appears.

Verifying the Root Cause

The importance of this step should not be underestimated. It is vital proof the root cause has been found.

- Apply a fix for the possible root cause and test the circuit, device, or system. If the symptoms have disappeared, the likely root cause has been determined. Make sure you can proof your theory of failure.
- Bring the circuit, device, or system back into its original state. Retest and verify the symptoms are back. Not doing this step can create ambiguity and falsify your findings.
- Reapply the fix and retest again. If the symptoms are gone, you can be confident you have found the root cause.
- Now verify the root cause on other circuits, devices, or systems and test them as well. Only when that test is positive should you announce your victory and provide the rework instructions to the broader design community in your company.

I wish you great success in debugging and root cause analysis. Make sure you use the right tools for the job. Too many times I have seen people use inadequate tools and missed the obvious root cause. This way problems propagated into different designs and were left undetected for many months or even years.