Go on green

DANIEL VASCONCELLOS

Troubleshooting a design—and adding things to that design to help in troubleshooting—is among the things engineers learn along the way. I learned this lesson during my first gig as a consultant. A crusty, old senior engineer had directed me to create a fixture, so I drew a schematic and gave it to the technician to build. When she was finished, she handed it to me. I tested and then disconnected it and put it on the senior engineer's desk.

About an hour later, he came stomping over to my desk and growled at me: "It doesn't work; fix it." I didn't argue but went into the lab to see what the problem was. The power came in through a connector that was supposed to be impossible to connect backward. Nevertheless, the senior engineer had managed to put it in backward. Putting the connector in backward was easier than I had imagined and could become a trap for a new user. I correctly plugged in the connector and tried the circuit to see whether any damage from voltage reversal had occurred.

When I was satisfied that the board was undamaged, I started to think about how I could prevent this scenario or at least show that it had happened. I got two LEDs—one red and one green—and two resistors out of the stockroom. I added these parts to the board so that, if a user correctly connected the power, the green LED would light. If the user incorrectly connected the power—that is, connected it backward—the red LED would light.

I reconnected the board to the test rig and informed the senior engineer that I had fixed it. He barked an acknowledgment, so I went off to do other things.

Until recently, every board I have designed has had at least one green LED to be able to start debugging when things go wrong. This arrangement lets me know that at least the power is connected and on. In other places, I add more LEDs to signal that things are working—or not. I add probe points so that I can easily stick a meter or a scope probe onto the board to more quickly get to the bottom of the problem. I also add a place I can solder in a bare wire; I can clip a ground lead to this wire.

Recently, though, when designing a board for a client, I decided to omit the LED. It was a simple enough board, so what could go wrong?

Several days after my client received the board, I received a panicked e-mail. This message set off a frantic series of messages back and forth about the board's not working to specification and signals' feeding through to other signals. What was the problem? After a day of worry, the client realized, and sheepishly admitted, that he had correctly connected the power but had neglected to turn on the bench supply. Everything appeared to be working because the microcontroller got its power from the USB (Universal Serial Bus) line and the microcontroller LED was on. He didn't make the connection that the power supply was off for more than a day. When he finally flipped on the power supply, everything was fine.

In the next revision, this board will get a green LED, too. Now, I take this step not just for me but also for my clients.

Here's a recent update: A later version of this board added fuses and TVS (transient-voltage-suppressor) devices for ESD (electrostatic-discharge) protection. In addition, the client told me that the board was to be compatible with automotive power systems. From research, I found that automotive voltages can range from 7.2 to 28.8V. So I tested the board over that range and noted that, just below 10V, the LEDs flickered and went out. Debugging showed that the fuses had blown, but not why. I replaced the fuses and tried again, with the same result. It turned out that the chosen TVS had a clamp voltage of 9.6V. When the input got to that voltage, the TVS acted as a short circuit and blew the fuse, saving the board but screwing up my testing. EDN

Brian Conley is an engineer with Circuitsville Engineering LLC (Beaverton, OR).

+ www.edn.com/tales