

Probing questions



My first job after college was working for a major HVAC (heating/ventilation/air-conditioning) manufacturer. The company was developing a new home-heat-pump-controller design using a microcontroller, and a number of prototype units were already available for development when I came onboard. The company had hired a consultant, who taught microprocessor programming at a local college, to write just enough software to get the processor off the ground. I began by writing some preliminary software to exercise the digital I/O, which contained a number of output relays.

As I began testing the software on the prototype, I almost immediately smelled something burning; a resistor in the power-supply section of the prototype soon started to emit smoke and then began burning. I quickly shut off the power, but it was too late to save the resistor and the expensive microcontroller.

I took the charred board to the electronics lab to do some forensic work. I replaced the resistor, checked the power supply, and found everything in order. I then put in a preprogrammed microcontroller containing the consultant's code, and the prototype correctly executed the code upon power-up. After return-

ing to my office to continue testing, I downloaded my code; as soon as the prototype powered up, the same resistor started to smoke again. Following two more trips to the lab for repairs, I found that the prototype would consistently run with the consultant's code but overheat and fail while running mine.

The smoking resistor connected between the input and the output of the regulator, according to the application note, to boost output power. I connected an oscilloscope at a convenient spot—the output of the regulator and ground—and single-stepped through the code to see what event had trig-

gered the overheating. I patiently single-stepped through my code several times, but the system didn't fail. On the scope, I observed an occasional—and somewhat expected—spike in the 5V-dc rail when one of the relays cycled. I eventually decided to run the board at normal speed; it continuously executed my code without any problems.

The next time I powered up the board, the resistor immediately started to smoke again. At this point, I realized I had not reconnected the scope to the prototype after my inspection. Could the scope probe somehow have been making the difference? I reconnected the scope using an FET probe instead of the X1 probe I had previously been using and again began stepping through the code. This time, the scope showed a substantial ringing on the power-supply rail each time a relay cycled.

The ringing on the supply rail made me ponder the effects of a seemingly benign software change I had recently made. The consultant had written to the relay outputs using byte writes. I had changed to a BSET and BCLR instructions to independently control relays. As a result, my code would cycle the relays in rapid succession upon initialization or at the end of a cycle. The cycling was enough to cause the regulator to latch up, causing the resistor to overheat.

I discovered that the decoupling capacitor had been “combined” with another capacitor closer to the processor; the intent was to decouple the processor and regulator using one part to save costs. The capacitor's higher value, remote location, and vias in the path to it allowed the regulator to become unstable with sudden changes in load. The X1 probe had provided just enough “decoupling” to maintain the regulator's stability.

I soldered a 0.01- μ F capacitor between the ground and the output terminal of the regulator, and the stability problems immediately disappeared. I quickly learned about the subtleties of using application notes and paying close attention to decoupling techniques. **EDN**

Jan D Clarke is a professional engineer in York, PA.

DANIEL VASCONCELLOS