

Weather or not: All signs point to “no”



Business had been going well at my former place of employment, a start-up company selling micro-processor-controlled weather stations to schools and broadcast entities. The start-up was taking off, orders were picking up, and our booth was popular at trade shows. Always on the lookout for things new and different, our president had become enamored of scrolling LED signs, and he insisted we needed one in the booth at the next show. “Just imagine: Up-to-the-second weather readings scrolling and flashing in bright colors will be right behind the presenter! Who wouldn’t stop to take a look at that?” he said. Although I had reservations about the aesthetic appeal of barometric-pressure readings blinking in rainbow hues, I had to admit that it would be a nice challenge that would show off some of our company’s technical expertise. So I agreed, but I regretted it when I found out that we had limited time to get this show on the road.

The boss had already selected a vendor for the scrolling signs, so I took

a look at the interface specification. I was pleasantly surprised to find that it supported a rich serial protocol for transmitting and updating messages for display. I needed only to convert our serial-data stream from the weather station to a format that the sign understood. The job was starting to seem a lot easier.

We wanted a stand-alone unit that we could later sell, so we went with an 8051-family microcontroller, which would receive weather data on one

UART and transmit display data with the other. To save time and effort, we ordered an OEM board from a vendor we had used before, and we recycled a number of embedded C routines to handle the serial-data streams. The only remaining task was to write some simple C code to transfer weather data from one stream to another.

The initial work went perfectly. I had hard-coded some test messages into the interface, and the sign displayed them perfectly. I set up the serial-input buffer and state machine on the receiving port, and I could see the extracted weather data in the debugging output. I had only to include the weather data in the output stream. I expanded the output buffer to hold the larger data, and I then reset the system.

Panic quickly set in. Instead of my orderly flow of temperatures and precipitation counts, I got only garbled letters and symbols. Something was seriously wrong. Even though we had tested the serial code in a number of other applications, I began to comb through it, looking for any mistake I could have made. I checked and double-checked every inch of the code. After banging my head against the wall for a few hours, I shut off my computer and went home.

The next morning I realized what the problem was. Prying up the main RAM chip on the OEM board from its socket, I found that one of the data pins was bent under the chip, causing all the ASCII characters to get scrambled before I sent them to the sign, which dutifully displayed the gibberish anyway.

After we straightened the pin, it was not long before we had a working sign-interface module, just in time for the next trade show. Even our president was impressed. “Looks great!” he said. “Now, I have this idea for a weather billboard.” **EDN**

JT Kloplic is a technical specialist with United Parcel Service Information Services (Timonium, MD). You can reach him at jtk_1997@yahoo.com.

www.edn.com/tales