



BY HOWARD JOHNSON, PhD

## Diagnostic testing (and tasting?)

In 1971, I enjoyed one blissful summer away from school working in a TV-repair shop. The first day on the job, my boss, Paul, explained how to work with customers at the repair counter: “Mainly, use your sense of smell. If it’s just dusty, that’s probably fine. If it smells like a burned resistor or burned capacitor oil, that’s going to cost. If it smells like baked milk, tell me, and I’ll sell ’em a new set. They already know their kid spilled milk into it, so they feel guilty. I’ll have the new set in place before hubby comes home.”

Wow! I had a lot to learn, and fast. Paul used all of his senses diagnosing a broken set. He could feel the 60-Hz hum from the power transformer, see the blue ionizing glow of a bad tube, hear the squeal of the 15-kHz horizontal oscillator, and smell the difference between a blown capacitor and a burned resistor. As for taste, Paul licked the terminals of a 9V battery to tell whether it was any good.

Today, human senses play a smaller role in diagnosing digital systems, but the philosophy remains the same: Use every tool at your disposal. Stick your hand into the chassis to feel the airflow. Touch the processor to see

how hot it’s getting. Listen to the disk chatter. Gather as much information as possible.

Compliance testing rests at the opposite end of the test spectrum. A compliance test begins with a specific list of features or performance metrics that you must verify. The test determines whether a system meets the acceptable criteria. If the system fails, the operator either discards the unit or sets it aside for rework. He makes no attempt at diagnosis.

The most interesting part of compliance testing happens before you hook up the first device under test. You must prove that your test-equipment configuration works, and works well enough to make the required measurements.

The diagnostic process, in contrast, is a more broadly based activity. It requires a keen awareness of all aspects of the system at hand. The operator must remain ever-vigilant during testing, aware of even the tiniest clue about system behavior (**Figure 1**).

David Agans, in his terrific book, *Debugging*, articulates a coherent nine-point strat-

egy for debugging that involves, first, deciding when to invoke the strategy (**Reference 1**).

As an everyday matter, you aren’t going to walk around trailing a cart full of data scopes, recording analyzers, and EMI detectors, because that would be too much of a hassle. You need not record everything, or take careful notes, or think before you act, as long as everything goes smoothly in the lab. You instinctively know how to tweak your schematic to casually fix the obvious problems that crop up.

To help keep things easy, break the system down into small pieces—so small that there is likely only one error (or no errors) in each section. Taken in small sections, problems tend to stand out in an obvious way. That makes debugging a snap.

The time to apply your high-level-debugging expertise is when things start to look ugly. Systems that harbor multiple errors often display inexplicable, impossible, or unreliable behavior. When that situation occurs, start thinking like Paul.

Take careful notes, augment your senses with recording devices that capture the history and current state of your equipment, and spend as much time thinking and planning for the next test as you do in the lab gathering data. **EDN**

### REFERENCE

1 Agans, David, *Debugging*, Amacom, September 2006, ISBN 978-0814474570.

**MORE AT EDN.COM**

Check out Howard Johnson’s videos at [www.edn.com/techclips](http://www.edn.com/techclips).



**Figure 1** A test professional pays close attention to every aspect of testing.

Howard Johnson, PhD, of Signal Consulting, frequently conducts technical workshops for digital engineers at Oxford University and other sites worldwide. Visit his Web site at [www.sigcon.com](http://www.sigcon.com) or e-mail him at [howie03@sigcon.com](mailto:howie03@sigcon.com).