

Passive part becomes aggressive



In 1981, I landed a job maintaining a computerized telephone-answering service. The system comprised a 16-bit minicomputer, disk drives, a paper-tape reader, a card reader, and a Teletype. The computer had switches and lights on the front panel and external I/O cards for all of the devices. The answering service assisted doctors, ambulances, and other critical-care clients, and the computer had to stay up and running as much as possible. There were two computers, aptly named A and B. One was supposed to be online, and the other was to be ready to go online if the online system failed.

The system ran on a real-time operating system and was very impressive in its time. The computer had 32 kbytes of core memory and four 1-Mbyte disk drives, yet it could route hundreds of phone calls to 16 operators.

The B computer was the online system, and the A computer was the offline system. My colleagues informed me that, although the A system had passed all of the diagnostic tests, it could not run the online system, and the B system could not pass the diagnos-

tic tests but ran the online system fine. I did my own testing and found this situation to be true. I wasn't content to leave it that way, though.

The A system would run the online software for 10 minutes, then become unresponsive. The previous repair people had replaced every card in the A system, and it still failed.

I had to somehow compare the systems to determine differences, and I first looked at the interrupts. I knew that the diagnostic software disabled

every interrupt except for the interrupt for the device under test. I also knew that the real-time operating system had good interrupt handlers for the devices it used. The only interrupt left to worry about was a spurious interrupt. The minicomputer had 128 possible interrupts, and the real-time operating system used only 10. I wrote software to count the number of spurious interrupts and found that the offline system had hundreds of spurious interrupts per minute. The online system had thousands of spurious interrupts per minute. The B system was far noisier electrically than the A system and still worked better.

Next, I modified the operating system, a backup copy, to "think" it was online. I forced disk, terminal, and Teletype activity to load the system. After 15 minutes, it went to sleep. I ran that test three more times just to prove that the A system would fail. The next step was to run only the Teletype to see whether the computer would continue to run. The test ran this way for several hours.

I then added one device at a time. The test failed with the addition of the second disk-controller card. This result was confusing because this card passed all of the offline diagnostics. The only part that I had not repeatedly swapped was the backplane card that held the I/O cards.

I found that I could run both disk-controller cards, with some rewiring, in the same backplane. When I did, the system tests ran perfectly. I swapped out the bad backplane.

I then put the A system online and watched it for several hours. It did not fail, and none of the users knew anything had changed. I was finally able to take the B system down for some much needed rest. **EDN**

Craig Hermann is underemployed in Fort Myers, FL. You can reach him at chermann@att.net. Like Craig, you can share your Tales from the Cube and receive \$200. Contact edn.editor@reedbusiness.com.

www.edn.com/tales