BY VISHWAS VAIDYA • TATA MOTORS

Getting real with the real-time clock

DANIEL VASCONCELLOS

In the 1990s, I was the lead servo-system engineer for the world's largest radio telescope (**Reference 1**). Our team was responsible for real-time motion control of 30 45m-diameter giant antennas that composed the radio telescope. Astronomers sitting at a central console focused the antenna position on the target—the radio star under observation. Because the rotation of the earth causes apparent drift in the position of the target, the antenna had to track the target by moving accordingly. The servo system was supposed to "servo track" this motion under the supervisory control of the central computer. By 1993, a few of the prototype antenna systems were undergoing proving trials, which uncovered many painful surprises. The chief astronomer reported that the antenna was drifting off the target by minutes—and that was just one of our problems.

Surprisingly, the antenna time would remain accurate when antennas were not moving. To catch the culprit, we installed a spy counter in an embedded timekeeping routine. The counter would drift only when antennas were in tracking motion. The system would then periodically send the counter value to the central computer, which would record the signature of all the events occurring when the counter reported a miss. Heavy traffic comprising instantaneous trajectory information always coincided with the time slippage.

Analysis proved our worst doubts true. The onboard software RTC (real-time clock) was lagging behind whenever the CPU was busy servicing the dense traffic arriving from the central computer. Because this traffic was absent when the antenna was still, the software RTC in that case would be accurate. This board unfortunately lacked a hardware-RTC chip; hence, we had to implement a software RTC, which proved accurate enough during bench trials. Apparently, we could not accurately simulate the dense traffic from the central computer.

The project received its finances from the public, and redesign would have been suicidal. We had two options: Provide the single-board computer with a hardware RTC without touching the PCB (printed-circuit board) or wind up the project.

Fortunately, PCs were then widely using the cheapest RTC with nonvolatile RAM that we could afford. Unfortunately, the chip included the notorious multiplexed address/data bus. A little thought revealed that a bit of smart coding can indeed infuse enough intelligence inside those parallel I/Os to mimic a multiplexed bus. We successfully tested the idea and deployed it after necessary software changes (**Reference 2**).

Once we equipped the antennas with the RTC daughterboards, the antennas started obediently counting the time with accuracy within milliseconds. Modern embedded-system engineers may not make the mistakes we made during the design stage. However, this story amply illustrates how a clever piece of engineering can always pull you out of a crisis! EDN

REFERENCES

1 "Giant Metrewave Radio Telescope," Tata Institute of Fundamental Research, www.gmrt.ncra.tifr.res.in.
2 Vaidya, Vishwas, "I/O bus emulates multiplexed address/data bus," *EDN*, Feb 15, 1996, pg 120, www.edn.com/archives/1996/021596/04di7.htm.

Vishwas Vaidya is an assistant general manager at Tata Motors (Pune, India). You can contact him at vmv74342@tatamotors.com.

www.edn.com/tales