

Foreword

By Jeff Duntemann KG7JF

The first computer program I wrote for a machine of my own didn't add, or display text, or crunch data. It turned on an LED. This was what we might call an omen. Or maybe imprinting.

The year was 1976, and the machine was a 6"×8" piece of perfboard with a couple of chips and some toggle switches on it and a rat's nest of wire-wrapping underneath. It was called a COSMAC Elf, and it had been published as a do-it-yourself project in *Popular Electronics* magazine in April of that year. It had 256 bytes of memory. Bytes—not kilobytes or megabytes or (lord knows) gigabytes. It wasn't much of a number cruncher. But boy, could it control LEDs.

Over the next several years I lashed together numerous COSMAC machines and branched out beyond LEDs into controlling motors, relays, and then (haltingly) graphics patterns on a TV screen. In 1981 I submitted an article to *Popular Electronics* describing the COSMAC IMP (Inexpensive Matrix Printer) which was a little cash-register thermal printhead/motor module with a simple parallel interface to the Elf. Alas, said *Popular Electronics*, the age of Elves is past us. But if I had anything on this brand new IBM PC thing, well, they'd fall at my feet.

And thus the path forked: Down *this* road went the PC, to revolutionize the whole data processing the industry, and down *that* road went the embedded machines, with exotic processors and built-in PROM, spinning steppers and winking LEDs inside everything electronic that *wasn't* a PC. Like most of us, I had to choose, and looking back over my shoulder said bye-bye to a process I actually enjoyed very much. After all, I'd been doing electronics when computers cost a gazillion dollars and were never seen by mere mortals. *Beware of programmers with screwdrivers*, so all the management handbooks went. With that in mind I entered the PC age, and left my screwdriver at the door.

I've been basically living the PC since 1982, without my screwdriver, and without the time to remember too hard what I'd been doing in those golden Seventies. And since 1982 something very significant has happened: The PC has eaten its way both up and down the food chain, devouring typewriters at the bottom and minicomputers at the top. It's now rapidly encroaching on Big Iron turf up on the high side, and—strangest of all strange things!—swallowing the embedded systems market whole.

The Embedded PC's ISA Bus

The almost unimaginable scale on which PCs and their components are manufactured has sent prices through the floor, to the point where midrange and high-end embedded work can be done economically on PC compatible devices. The PC architecture is cheap, fast, and almost infinitely versatile. It's well-documented, and the available software works well and can be had for a song.

So the two paths have converged again. If you understand the PC, you can pick up your screwdriver again and not have to learn a new architecture or instruction set. All you really need to learn is the way to hook those LEDs (and LCDs, and what have you) to the PC's well-understood bus. Ed can teach you that. You may be a little surprised (especially if you've spent the last fifteen years coding in Basic) at what the PC actually does at the chips-and-memory level—but that's all here as well.

There may come a day when Windows CE (or whatever they call it this week) greets you from an LCD on the side of your toaster. But until then, it's you, the bits, and the bus. If there's ever been a better book than this for developing those new skills (or scraping the rust off some very old skills) I've yet to see it.

Good luck. Don't forget your screwdriver.

Jeff Duntemann KG7JF
Scottsdale, Arizona
March, 1997