Gordon Bell: Building Blocks of Computing

Charles Severance

Gordon Bell describes how the minicomputer served as a building block for computing-enabled innovation and the eventual evolution of computing applications.

Gordon Bell has had a long and varied career: early employee of Digital Equipment Corporation (DEC), designer of many of DEC's PDP series of minicomputers, leader in computer science at the National Science Foundation, long-time proponent of the Computer History Museum in Mountain View, California, and creator and funder of the Gordon Bell Prize, which the ACM awards for improvements to parallel and scalable computing. Today, he's a Researcher Emeritus at Microsoft.

I recently met with Bell at the Computer History Museum in front of a fully restored and operational PDP-1. To watch the full interview, visit www.computer.org/computingconversations.

PDP-1

In the late 1950s, Bell was pursuing his PhD in speech recognition at MIT and working on analysis by synthesis:

> I had to build a tape control unit for the TX-0 [one of the first transistor-based computers] and went to Digital [Equipment Corporation] to buy modules—flip-flops and gates, all the components that today are so low level to computer scientists. I bought 50 or so and put them together to make a tape control unit.

But once he moved on to building the software for speech recognition, he realized that all the fun was in making and building the hardware and that achieving effective speech recognition with 1950s technology was virtually impossible:

> What I learned was that I didn't want to be a researcher. Above all, I didn't want to be a speech researcher because I saw recognition not occurring for another 20 years. I enjoyed building the program to do analysis by synthesis, building the hardware and making it work, and I really wanted to do the engineering.

DEC soon recruited Bell, and his first task was to design the PDP-1's input-output subsystem. As part of that design, he invented the first universal asynchronous receiver/transmitter (UART), which was essential for moving data between systems or components that didn't share a common clock. The focus on I/O meant that the PDP-1 became more of an extendible building block than a computer built for a particular purpose:

> The first documentation I wrote for the PDP-1 was a brief manual about how to connect stuff to it because we saw researchers as our market. The difference between the PDP-1 and subsequent machines that we built was that the I/O got much more flexible and easier to use, so that when somebody read the manual, he or she would think, "Oh my god, I'm going to connect that for process control, or I'm going to connect that and make an oscilloscope or a pulse height analyzer or some other instrument."

This approach, which focused on expandable building blocks and flexibility, led to the minicomputer. Its expandable architecture was a core idea for the PDP and VAX series of computers and is what sets this category apart from supercomputers and mainframes:

> What the minicomputer did was enable the computer to be used as a component for everything. I just wrote an article about the rise and fall of the minicomputer. I had to say, "Look, the function we were creating

Published by the IEEE Computer Society

was nothing like the past." It wasn't a scaled-down mainframe that you could interact with: that wasn't the goal. It wasn't a record keeper. It wasn't a computer in the sense of a supercomputer. We weren't after numbers or bits or records. We were building a component that could be used in any number of ways, and it ended up being used for message switching—telegraph wires going in and out—which is a core part of networking today. Minicomputers were the pioneers of interactive and personal computing.

PDP-11

Much of the architecture in these minicomputers evolved from standardizing the connections and connectors, which led to the bus architectures that allowed devices from vendors and system builders to be plugged into the same computer. DEC pioneered the original equipment manufacturer (OEM) market with the PDP-8 that most of us consider to be the first minicomputer:

The idea was to make our minicomputers as flexible as possible so that they could be interconnected, take data, and interface with other systems and people. In the PDP-1, we had a core of processor memory and some I/O, with wires connecting to other things. When we made the PDP-11, we had a single wire, with memories, processors, and everything else connected to the Unibus. In 1981, when we announced Ethernet, I claimed it was the "Unibus of the '80s." Later, Ethernet became the "UART of the '90s."

With technologies like the Unibus making it so that any enterprising engineer could prototype and build an expansion card, the number of applications for a minicomputer like the PDP-11 was limitless. DEC produced a standards-based expandable component, and almost 100 vendors—including Beckman Instruments, Foxboro, Western Electric, and Olivetti—used it for everything from process control to rocket navigation to telephony and telegraph operations. The PDP series of computers was a hacker's delight; it was where new innovations like the ARPANET could gestate. The notion of building a technology that's usable as a component has continued over time:

In every generation, you get a machine or computer that's a component. For example, the microprocessor as it first came out was a component five years before it was a personal computer. Once the microprocessor got a little bigger, the PC began to be used as a component.

---

ccc
Ccc
ccc
Ccc

The modern cell phone is the most amazing component you could ever think of—well, for another few years, at least! Mine has 64 Gbytes of storage, enormous processing power, radios, an accelerometer, and a GPS unit. It can be used as a component for medical diagnosis, a quadrotor autopilot, and an interface to almost everything.

SPACEWAR

Video games are another example of a minicomputer-like architecture in which a central computing and storage capability is surrounded by rich sensors and other I/O forms. Examples of early computer video games abound, but the PDP-1's Spacewar is the first example of a widely available, interactive game because the early PDP systems were the first widely available minicomputers with a light pen and graphical display:

Harlan Anderson, one of the founders of DEC, discovered that at the end of the year, he could give a computer away and get a total deduction for it. I don't recall if that was what made us profitable that year or not. The PDP-1 we gifted to MIT was right next door to the TX-0, so all the students at that time took software ideas that had been developed for the TX-0 and put them on the PDP-1: the first text editor, which we called "the expensive typewriter," and an interactive debugging environment and all the light pen interactions and connections. The most famous one that is still in use in a sense was Spacewar.

While much of Bell's career has been dedicated to engineering and building state-of-the-art computing technology to be manufactured and sold, his most recent work has been to reflect on what we've seen and learned over the past 50 years of computing exploration. He describes his last 15 years as being an "accidental researcher," exploring what it means to have your whole life captured digitally. He gets curious about an idea and then digs into it, connecting the past, present, and future in wonderfully interesting ways. If you would like to see an example that describes the role of minicomputers in the history of computing, read "STARS: Rise and Fall of Minicomputers" (http://ieeeghn.org/wiki/index. php/STARS:Rise_and_Fall_of_ Minicomputers). **C**

Charles Severance, Computing Conversations column editor and Computer's multimedia editor, is a clinical associate professor and teaches in the School of Information at the University of Michigan. Follow him on Twitter @drchuck or contact him at csev@umich.edu.