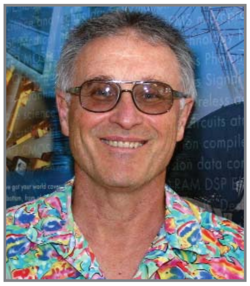
Interdisciplinary Engineering & Research

An Interview with Bruce Land



Bruce Land is a Senior Lecturer in Electronics and Computer Engineering at Cornell University (www.nbb.cornell.edu/neurobio/land/). This year he's teaching two courses: one covering microcontrollers as components in electronic designs, and one dealing with designing FPGA circuits for embedded applications. Bruce is also co-director of the Masters of Engineering Program in the Cornell School of ECE. Since 2002, seven of Bruce's articles have appeared in Circuit Cellar. *In July 2010, I interviewed Bruce about his background, work, and many interests, which range from electronics to neurobiology to programming to physics. — C. J. Abate, Editor-in-Chief*

CJ: Please describe your background and how you ended up at Cornell.

BRUCE: I went to Harvey Mudd College, graduated in 1968 with a degree in physics, and moved to Ithaca, New York, to start graduate school in neurobiology doing artificial neural net research with Frank Rosenblatt. Today, I am a senior lecturer in the School of Electrical and Computer Engineering (ECE) at Cornell University. I teach two design courses, one using microcontrollers and C, and one using FPGAs and Verilog plus C. I also collaborate with a couple of labs in the Department of Neurobiology and Behavior (NBB). In this role I mostly simulate neural systems, figure out ways of visualizing biological data, and build instrumentation.

CJ: When did you become interested in electronics and engineering? How long have you been reading *Circuit Cellar*?

BRUCE: I have been interested in electronics as long as I can remember. One of my first memories is of sticking two screwdrivers in a socket, then grabbing them. What I actually remember is flying across the room backwards. By middle school I was interested in RF and high voltage and built a Tesla coil with an 811A power triode that would light up a fluorescent light six feet away. I got interested in Boolean algebra and computing in high school, but did not have access to a computer until college. I have been reading *CC* since 1997, but read columns by Steve Ciarcia in *Byte* magazine starting in about 1979.

CJ: What about working with "embedded technologies" and your first MCU-based design?

BRUCE: Starting in 1973, I used an Intel 4004 processor set. By 1977, I was using a DEC PDP-11 to control biology experiments and record data. I bought an Apple II in 1979 and wrote 50 pages of assembler to make an animated orbital dynamics game. Beginning in 1986, I worked with computer graphics on supercomputers for 12 years. In 1997 I started teaching embedded design in ECE and building instrumentation in NBB.

I built a cricket call generator for Ron Hoy at Cornell using an Intel 4004 processor in 1973. It had a keypad, LED numeric display, and an analog sound synthesizer, all driven from 512 bytes of ROM. The device was hand-soldered and took nine months to build and debug. We entered test code by stepping through memory toggling front panel switches because we could not afford a development system.

CJ: You have an interesting background: physics, neurobiology, graphics/animation, computer science, and ECE. What is your main interest, and how does electrical engineering factor in?

BRUCE: Harvey Mudd College gave me an excellent math and physics background, which I have used ever since to learn and understand new fields. Between my junior and senior year at HMC, I spent the summer at Ames National Lab in Ames, Iowa, as an intern in a nuclear physics group. In Ames I had very little to do outside of work, so I spent a lot of time in the library, where I found a very interesting book on perceptrons written by Frank Rosenblatt. I decided to go to graduate school at Cornell because of that book. I worked on artificial intelligence for two years, but Frank was killed in a boating accident in 1971. I then switched advisors and started working on the membrane biophysics of developing muscle. The math/physics background made this possible and interesting. The mathematical modeling for this work was done analytically, but toward the end of my dissertation work, it became obvious that computer simulation methods were necessary for the next step. I stayed at Cornell as a postdoc doing electrophysiology and computer simulation of nonlinear reaction-diffusion equations to model the binding of acetylcholine (a neurotransmitter) to muscle. I became more interested in the simulation than the physiology, and when the Cornell Theory Center became one of five supercomputer sites funded by NSF in 1985, I switched departments to start supporting high-performance scientific computing. Over the next two years, I got interested in the challenges of converting vast numerical supercomputer output into understandable and informative graphics and ran the Cornell Theory Center graphics group for 10 years. During that time, I started teaching computer graphics in the Department of Computer Science, which was great fun. The Theory Center lost federal funding in 1997 and, by chance, I found out that Electrical and Computer Engineering needed someone to teach embedded design. So I applied. I really enjoy the interplay between hardware and software and watching students design interesting devices using all the tools they have learned. The physics and math from college, plus the one credit of FORTRAN that was required of all freshmen, form the basis for continually re-educating myself.

CJ: You were using computer modeling as early as the late 1970s for your neuromuscular studies. Was that the first time you used computers for your academic endeavors?

BRUCE: I started using computers (digital and analog) academically for data reduction and modeling for physics labs in college. The machine was an IBM 1620 with punch card input. In graduate school I started using the FOCAL language running on a PDP-8 interactive system with teletype input for statistics. But the first real time experimental use was hooking a PDP-11 to my electrophysiology rig, which at the time meant writing a device driver in assembler to run the A/D converters and building a hardware clock for the converters. This allowed me to collect hundreds of waveforms on a floppy disk in an afternoon of recording and reduce them in a few minutes to a useful summary. The output data from the experiments was then compared to simulations of the neurotransmitter reaction-diffusion equations underlying the waveforms. The simulations had to be run on an IBM mainframe and it was all we could manage to simulate a one-dimensional diffusion system, even though we knew the real system was probably at least 2-D. By comparing the simulated and real waveforms, we could deduce the chemistry of the binding of the neurotransmitter to muscle.

CJ: In the 1990s, you taught graphics, computer programming, and electronics courses at Cornell. How did this come about?

BRUCE: As part of my Theory Center job, I had an academic appointment in the Department of Computer Science from 1992 to 1997. This allowed me to teach computer graphics. When the Theory Center lost federal funding in 1997, I started teaching in Electrical and Computer Engineering, but I was actually shared between ECE and NBB. In neurobiology, I taught electronics and programming for biologists. I had a good feel for how biologists think (having worked in neurobiology for 18 years previously), so I converted the hardcore math approach, which tends to be used in engineering, into a form that was rigorous but much simpler to follow, with lots for examples based on the needs of the research biologist. For example, it is fairly easy to compute the length constant of a leaky cable (Lord Kelvin did it) using calculus, but it is even easier to use high school algebra and take the calculus-style limit at the very last step in a way that people without very much calculus can easily follow.

CJ: Since 2002, *Circuit Cellar* has published seven of your articles. With so many interests in different disciplines, how do you choose a subject or technology on which to focus?

BRUCE: I just do whatever seems interesting at the time. I read a lot, so some topics are based on a technique that I have read about. Some topics arose because they seemed necessary for the way I wanted to teach the microcontroller or FPGA course. The video generator came about this way and so did the DSP articles. Some topics are of mathematical interest to me, and sometimes I just want to see how much performance I can squeeze out of a small processor.

CJ: Your TV oscilloscope project (*Circuit Cellar* 161, 2003) was designed specifically for use in Cornell's neurobiology labs. Is it still in use? Any upgrades recent upgrades or redesigns?

BRUCE: There was one further design change of that system so that it is an almost complete electrophysiology system. I never wrote it up, but it is linked at www.nbb.cornell.edu/neurobio/land/PROJECTS/TVnuS/index.html. I also designed some model neurons for teaching, which I have not done much with, at www.nbb.cornell.edu/neurobio/land/PROJECTS/Model-Cell442/
index.html. All of the circuits have been used in teaching labs, but are not currently in use.

CJ: *Circuit Cellar* readers all over the world are now familiar with your work, as well as many of the projects your students completed in your ECE courses. During the last few years, we published several articles about projects that began in your lab. Examples include: "Self-Powered Solar Data Logger" (A. Krich, *Circuit Cellar* 198, 2007); "RFID Security System" (C. Ross and R. Goto," *Circuit Cellar* 199, 2007); and "Keystroke Communication" (N. Paya and V. Ganesh, *Circuit Cellar* 227, 2009). Few other instructors we know of have so many talented students producing such innovative MCU-based designs. Tell us about your design courses. Is the ultimate goal to build something? Describe the balance between classroom lectures and actual design time at a workbench.

BRUCE: The course ECE 4760 is a design course in which we ask the students to use all of their engineering background to produce interesting microcontroller-based devices. We spend time talking about specific programming techniques, intellectual property issues, debugging, and how to decode a cryptic datasheet. The students have an awesome theoretical background, and this course is one of several which gives them a chance to apply their theory to practical design. The course is primarily driven by lab exercises. There are no tests or quizzes; rather, students are expected to show up for lab ready to perform and are graded on their performance and the resulting lab reports. For the last five weeks of the semester, the students have only one assignment: to design, build, demonstrate, and document a device of their choice. The class webpage has over 350 projects. These project reports serve to set the expectation level for grading and act as a source of code for new projects (properly acknowledged). I am continually amazed at the creativity the students show and how hard they can work. At the end of the semester, I encourage and help the students to publish their work and, occasionally, to disclose a potential invention to Cornell in order to pursue a patent.

CJ: What do you ask of your students at the beginning of each semester? How do students choose designs and supplies?

BRUCE: I ask the students to be proactive and behave more like employees and less like students. They are expected to do the assignments without being forced by testing to memorize details. Project designs are specified in several steps. Informal discussion with each student group results in a formal project proposal, which is reviewed by the teaching assistants and by me. We comment on the hardware and software feasibility, lab facilities (e.g., Does the project require a milling machine, which we don't have?), and overall novelty and interest of the project. Typically, a class of 100 students will have about 40 to 45 different projects carried out in groups of two or three students. Parts for the projects are scavenged, purchased, or donated. Each group is limited to a budget of $75, but scavenged or donated parts do not count against the budget. I have a couple of cabinets of random parts, which the students can use, but some purchase their own parts, up to the limit of $75. Since $75 is only about one-half the cost of a textbook, I don't believe that it represents a significant burden to students.

CJ: Tell us about your interest in FPGAs. We're familiar with your hybrid computing article, and we know they are a central topic covered in your ESC 5760 course.

BRUCE: FPGAs are great when you need computing bandwidth or hardware flexibility, or want to play with parallel computing. Students can design and prototype a 32-bit microcontroller in a couple of hours using modern tools. ECE 5760 assumes that the students can program in C and design hardware in Verilog. We ask the students to combine custom processors with special-purpose hardware (which they design) to solve real-time problems like video or audio synthesis. Often, the solution will involve parallel hardware algorithms to get speed. I think that training in parallel computing is essential at this time.

CJ: I'm sure many of the young engineers you work with have a decision to make. They have a design idea, and then they ask: Do I use microcontrollers, FPGAs, or CPLDs? Any general advice?

BRUCE: There are many trade-offs: ease of programming, familiarity, cost, I/O bandwidth, fabrication, and so on. We chat about all of these when starting a project.

CJ: Speaking of FPGAs, you recently informed me that you wrote some code for computing chemical kinetics solutions on an FPGA. What exactly is chemical kinetics, and how are you using the FPGA?

BRUCE: Chemical reactions are processes by which chemicals are converted to other chemicals. A huge number of such reactions are going on in your body as you read this. For instance, sugar is being converted to less energetic compounds, plus high-energy ATP, a chemical which is the energy currency of the cell. Chemical kinetics describes the dynamics of the conversions as a rate of conversion which depends on the concentrations of the reacting chemicals, the temperature, the pH, and numerous other factors. These factors are often summarized as a "rate constant." Traditionally, the rate equations are written as a set of differential equations and solved using numerical methods, such as Runga-Kutta integration or "stiff equation" solvers. But real reactions always occur between individual molecules, which are not "differentially smooth," except in the limit of large numbers of molecules. An alternative approach is to solve the system by treating the rate constant as a probability of reaction over a short period and asking if the number of molecules changed in the short time step, then repeating. This Monte Carlo approach is more time consuming than the differential equation solution but is more accurate when only a few molecules are reacting and gives more information. It is possible to parallelize the computation on the FPGA by generating many random numbers simultaneously. Speedups of between 10 and 500 are possible over the PC solution of the system.

CJ: FPGA projects, programming, neuroscience, graphics, physics. Where do you find the time for all of this tech work? Do you have any free time?

BRUCE: I do some of the work, but a large number of very good students do a lot of it. I supervise about a dozen Master's of Engineering projects and undergraduate projects each year, outside of the two classes I teach. In my free time I do some forestry and gardening and try to live sustainably. We have 5.6 kW of photovoltaic solar and 3 kV of solar hot water collectors. We have chickens and honey bees and grow most of our own food.

CJ: Planning any other projects for the near future?

BRUCE: A collaboration with the Cornell Lab of Ornithology may result in some interesting bird-based ad hoc networks. I am working on the simulation of toadfish swim bladder sound production to try to understand some, apparently, chaotic sounds. I am designing a video analysis system for fish activity.

CJ: You interact with up-and-coming electrical engineers, embedded designers, computer scientists, and programmers on a daily basis. What would you say are the "hot" topics exciting this new generation of engineers?

BRUCE: Energy (control, production, and storage); biomedical instrumentation, game design; robotics; parallel or multicore, multithread computing; photonics and optical techniques in computing; and human-computer interaction.

CJ: What is the biggest "growth area" for the embedded industry?

BRUCE: Instrumentation: energy control, biomedical measurements. Geriatric applications for us baby boomers: smart walkers, web-attached pill dispensers, etc. And, of course, mobile devices.

NEED-TO-KNOW INFO

Knowledge is power. In the computer applications industry, informed engineers and programmers don't just survive, they *thrive* and *excel*.

For more information about Bruce Land's work, the *Circuit Cellar* editorial staff recommends the following content:

—

Floating Point for DSP

by Bruce Land

***Circuit Cellar* 235, 2010**

For DSP and other fine-grained parallel operations, you need to pick a floating-point representation and implement five basic operations. The 18-bit floating point described here allows up to 70 floating-point multipliers and around 150 floating-point adders to be placed on an FPGA. Topics: DSP, Floating-Point Math, FPGA, Conversion, Matlab

Go to: www.circuitcellar.com/magazine/235.html

—

Hybrid Computing on an FPGA

by Bruce Land

***Circuit Cellar* 208, 2007**

Bruce explains how to simulate the parallel functions of an analog computer on an FPGA. Now you can harness the advantages of parallel execution and a general-purpose CPU on the same chip. Topics: FPGA, CPU, Parallel Execution, Analog Computer, VHDL

Go to: www.circuitcellar.com/magazine/208.html