BME 194: Applied Circuits closing survey

Kevin Karplus

March 18, 2013

1 Design Goal

Improve the applied circuits course.

Some design goals for the course will remain unchanged, but how those goals are met can be changed and improved. The goals I see as important are

- By the end of the course, students should be able to design, build, and debug simple electronic circuits for interfacing sensors to computers (instrumentation amplifiers, and low-pass and high-pass filters, for example). For many students, this will be the sole electronics course they take, so they must be able to do something useful without further courses.
- Building should include soldering on PC boards, and not just bread boarding, so that students can build circuits that last more than a day or two, should they need to.
- Students should learn how to use standard electronics bench equipment (power supplies, multimeters, oscilloscopes, and function generators).
- Students should be able to read a data sheet for a component that they are familiar with (such as a thermistor or op amp) and be able to find crucial information about the part.
- The course should have a strong design emphasis (rather than theoretical analysis), to compensate for the shortage of design work in early courses in the bioengineering major. To the extent possible, every lab should include design questions that are essential to the lab.
- The design problems and labs should be relevant to bioengineers.
- Students should write extensively and get detailed feedback on their writing, as the bioengineering major (as most majors) has insufficient feedback on contentful writing for students to develop essential writing skills.
- Students interested in bio-electronics should be able to use the course as a very strong base for subsequent electronics courses.

Which of these design goals have been met in the current course? Which not? What can be done to meet them better? What other goals should the course have? Should any of these goals be removed or modified?

2 Background

This was the first offering of the Applied Circuits for Bioengineers course and was intended as a prototype, testing out what worked and what didn't work. Some parts were carefully designed months in advance, some were added at the last moment—corrections for unanticipated problems. For example, some labs came late in the quarter because they took longer to design rather than because the material belonged later in the quarter.

The end of the quarter is an excellent time to reflect on what worked and what didn't and to begin planning for the next offering of the course. I will have to file revised course approval paperwork soon in order to get the course into the catalog, so any major changes in the course need to be thought about immediately.

Think about the prerequisites for the course. I tried to pare them down to essentials (which I took to be single-variable calculus and physics electricity and magnetism). Were any of the prereqs unneeded? Should other prereqs have been added? I deliberately left computer programming off the list or prereqs, though some prior programming courses would have helped with understanding gnuplot scripting and with what the capacitance touch-sensor program does. Should a programing course be added as a prereq? If so, which one(s)?

Is the course accessible to a sophomore in the bioengineering program? Would it encourage students to enter bio-electronics as a concentration?

I've relied heavily on lab handouts, Wikipedia, data sheets, and on-line documents such as *All About Circuits* `http://www.allaboutcircuits.com` for the background "textbook" material for the course. Partly this was to avoid over-priced textbooks, but largely it was because I could not find a textbook with appropriate coverage of material for the course. (The *All About Circuits* on-line textbook is as good a fit as any of the paper ones I've seen.) Should I continue looking for a paper textbook? Should I assign more readings from *All About Circuits*?

3 Pre-lab assignment

Go over the list of labs and think a little about each one.

1. Thermistor
2. Microphone
3. Electrodes
4. Hysteresis oscillator
5. Audio amp (op amp)
6. FET & phototransistor (tinkering lab)
7. Sampling and aliasing
8. Pressure sensor (instrumentation amp)
9. Class-D power amp
10. EKG (instrumentation amp, high gain)

What did that lab teach? How could the lab have been improved? How much time should be allocated for the lab? Was there too much scaffolding provided (so that you were doing cookbook procedures, rather than your own design work)? Was there too little scaffolding provided (so that you were floundering with no hope of doing your own design)?

Think also about the labs as a whole: what order should the labs be in? which labs should be eliminated? what additional labs are needed? How much time per week should be allocated to labs and how should that time be split up?

I wanted students to do design work before coming to the labs, so that time in the lab was not spent doing preparation for the lab. This did not seem to have been done uniformly. Is there some way to encourage or enforce pre-lab preparation?

Should I assign more routine homework exercises in addition to the design exercises for the labs? Are any of the worksheets for *All about Circuits* worth the time they would take?

4 Parts, tools, and equipment needed

Go over the tools and parts lists at `http://users.soe.ucsc.edu/~karplus/bme194/w13/tools-and-parts.html` For each tool or part, consider how much you used it and how much it cost.

The parts kits were intended to provide all the parts needed for the labs, so that a student working without a partner would be able to complete everything, and not be overly constrained by the availability of components. Were there any parts you needed more of? What parts did you never use? Which parts were over-priced for the amount you used them?

The tool kit was intended to provide tools that could be used for future projects, either as hobbyists or in subsequent electronics courses. Which tools were used extensively? which ones not used at all? Which ones were too cheap to do the job? Which ones too expensive for the amount they were used? What tools should be added? (For example, a soldering iron of the quality used in the labs costs about $20, but a cheaper iron without the safety cage, tip-cleaning sponge, or adjustable output can be bought for about $5—should a soldering iron be added to the tool kit?)

The Arduino is a popular board for hobbyists and artists and is useful for projects other than the course (such as do-it-yourself thermal cyclers), but is moderately expensive at $30. Was it used enough in the course to be worth the price?

Abe would like feedback on the Data Logger software. What features would you like to see added? Some he is considering include

- Faster live charts, so that the plots don't stall at high sampling rates.
- Better labeling and scaling for analog inputs—for example, being able to specify the values of the low and high end of the range and scaling the numbers read appropriately, and having a name for each input.
- Providing plotting output (in eps, pdf, png, svg, or other formats) directly from the data logger, with labeled axes and titles, to eliminate the temptation to do low-quality screen shots.
- Provide gnuplot scripts as a possible output format for the plotter, to allow further customization, such as model fitting.
- Supporting other boards, such as the Arduino Due, which has more channels and a faster A/D converter.
- More sophisticated triggering modes, such as burst triggering, where each pin change or timer interrupt causes a burst of n reads at a specified sampling rate.
- Hand-entered data for things not measured by the Arduino, so that you can do a recording, change a parameter, and continue the recording, with the parameter values as another column in the data file.
- Better resolution on sampling period, so that periods that are not multiples of 1 msec can be specified (and sampling can be specified as a frequency, like 60Hz).

Which of these would be most valuable? What other features might you want? If you want to give Abe a "testimonial" that he can use for "advertising" Data Logger at the state science fair, feel free to do so either by e-mail or on the Data Logger issue tracker `https://bitbucket.org/abe_k/arduino-data-logger/issues/new` Of course, please report any problems or suggestions for improvements there also.

We did not use all the equipment in the lab (for example, there were frequency meters that we never used) and we didn't come close to using the full capabilities of the equipment. Would more instruction in the use of the equipment have been useful? We covered only the basic features that any instrument of a given type is likely to have, not the more complicated features specific to the expensive instruments in the lab. Would it have been worthwhile to spend more time on the details of the instruments?

5 Procedures

As much as possible, I wanted students to do their own designs and to test things in the lab, rather than relying on an instructor telling them whether they were "right" or "wrong". Did the "try it and see" approach to course work lead to students having better design skills and more confidence in their ability to do things?

Students sometimes took very poor notes while working in the lab, so that their reports did not contain accurate records of the circuits they tested or the measurements they made. What can be done to encourage better lab notebook habits?

Block diagrams turned out to be more difficult for students than I had expected, and less used as a design tool. Is there a way to teach block diagramming earlier in the course as a way to divide problems into sub-problems? Or do the circuits have to get complicated before the value of dividing into sub-problems becomes apparent?

6 Demo and writeup

We did not have a formal check off of demos in the lab, because the class was small enough not to need a formal system. As the course ramps up to 40 students or more a year, we might need a more formal demo check. Would a checklist of what needs to be demonstrated be useful for the students as well as the TA/instructor?

Gnuplot was used extensively in the course for preparing graphs of data and for fitting models to the data. Is this a tool you expect to use again? Should more time have been spent teaching how to use the tool effectively?

The writing and writing feedback are very time consuming both for students and for the instructor, but improvement in technical writing comes only from extensive practice with feedback. Most courses at the level of the applied circuits course have only fill-in-the-blank and similar formulaic writing, which reduces the workload but provides no writing practice. What is the right balance between the circuits content and the writing practice?

In place of some of the writing, we could use less labor-intensive assessment, such as exams and homework. What is the optimal balance between different assessment methods?