

V. Hunter Adams
Cornell University
Department of Mechanical and Aerospace Engineering

Teaching philosophy statement

Curiosity and engineering

The path to the frontier of every field, but particularly to the frontier of engineering, is longer than it has ever been. That path used to be shorter, and the walls that separate it from the paths associated with other fields used to be lower. You could peek over the walls and see the nearby paths associated with neurobiology, ecology, oceanography, and all the other sciences. That view is not so clear anymore. A side-effect of advancement seems to be hyper-specialization and wall-building between fields.

One of my great joys as a graduate student has been to bushwhack my own trails from engineering to some of these other fields. At various times, I've collaborated with researchers in plant sciences (Justine Vanden Heuvel and Tim Martinson), veterinary science (Francisco Yeal-Lepes), ornithology (Julian Kapoor), and electrical engineering (Kirstin Petersen). I started doing this simply out of curiosity, but I soon discovered that it was making me a better engineer. I started having ideas that would have never come to me otherwise. The ways in which learning about dairy cattle will help with spacecraft engineering are not always obvious, but I have never found it unproductive to indulge a curiosity.

This has had a profound effect on my teaching. I cannot make the path to the frontier any shorter, but maybe I can lower the walls between paths. When I teach about a particular topic, I try to pull examples from as many fields as I can. I recently gave a lecture about swarms of spacecraft in MAE 4060 which drew heavily from ideas in ecology and animal science, where we also see large collections of individuals cooperating toward a common goal. I supervise many

undergraduate microcontroller-based projects, one of which involved building an underwater communication system. I encouraged that particular student to spend some time studying the animals that communicate underwater to see what he could learn from them. I make many similar suggestions in the mechatronics course that I TA, in which students build autonomous robots using microcontrollers and off-the-shelf sensors.

And I'll often say it explicitly. When I'm addressing a class that I haven't met, I will start the talk by saying the thing that I wish my instructors had said to me: *Be curious*. There is a growing sentiment that a diversity of interests indicates a lack of commitment to one's particular field of study. It doesn't. It makes for better engineers.

Teaching to all levels of abstraction in engineering

My favorite part of teaching project-based lab courses like mechatronics, or about supervising undergraduate microcontroller projects, is the moment when a student discovers one of the deepest truths of engineering: *nothing ever just works*. Many of the students at this university have grown very accustomed to success, and it can be a jarring to discover that building things is fraught with so much failure. Coaching students through this is a delicate process that must be done thoughtfully.

At this first moment of frustration, I like to tell the students that they've discovered something very deep. *There are levels of abstraction to engineering*. There is theory, which they have learned in the classroom, and there is the building of things, which they are learning now. Each informs the other. Being good at theory helps you build things, but it does not get you all the way there. Problem solving in the lab is different from problem solving in the classroom.

I attempt to get them into the appropriate psychology for debugging hardware. Many will begin with the same mindset that they use to approach a problem set, which is generally one of frenetic stress. The goal is to get them into the mindset of *play*. Rather than being entirely goal-focused, I

help them get into the mindset where debugging is an enjoyable process on its own. This is critical when working with hardware, where almost all of one's time is spent fixing things that don't work. Debugging must be a joy, or the student will not be happy in the class or in her career.

There are circumstances which can prevent this mindset, the most significant of which is stress. Removing as much stress as possible is one of my principle responsibilities as an instructor or project advisor.

Making learning and creativity possible

Engineering students lead incredibly stressful lives, which is not conducive to learning or to creative thought. Many of these sources of stress are outside the control of an instructor, but there are some things that he can (and must) do to help.

Perhaps most significant is making certain that the student has the resources that they require in order to be productive when they sit down to work. This means the obvious things, like having impeccably organized notes and figures, but what it really means is *access to the instructor*. It is absolutely imperative that the instructor for a project-based course, or the advisor to a hardware-based project, spend a significant amount of time each day in the laboratory with the students. This is true for any sort of class, but it is particularly true for hardware.

There are some bugs that are instructive for students to work through on their own. Getting a microcontroller to communicate with a sensor, for example, is an incredibly instructive problem to tackle. Hardware debugging is also filled with the sorts of bugs that add no value to the learning experience, but which can hold up a project for hours or days. For all of the student projects I've advised, I've been certain to be in the lab with the student for at least an hour each day to move them past these sorts of errors. Being in the lab also helps me celebrate each of their

small victories with them. That's an important part of developing the play-mindset toward hardware work.

Absolute respect for the student

I treat students identically to the way that I treat colleagues in another field. The operating assumption is that they are of equivalent intelligence to me, but that they lack information in the particular topics covered in the class. That respect has, in my experience, made students feel much more comfortable contributing their ideas to a discussion, or proposing a new project.

Above all, I care. I care about the topics that I'm teaching and how well I am teaching them. I care how well the students are learning and about their mental health. Teaching, indulging curiosities, and building projects are the things that make me happiest.