Decal: Going Down the EECS Stack
Syllabus Fall 2017

Facilitators: Grace Park, Chris Powers, Dalton Omens, Titlis Wan
Contact: decal@hkn.eecs.berkeley.edu

Introduction

There are a lot of EE and CS classes at Berkeley, but can you take all of them? As classes become harder to get into, it is important that students know which classes are the most useful, according to their personal interests. This Decal is offered with the goal of providing students with a broad survey of EECS and a general sense of courses and subfields within the major. Additionally, this Decal requires no EE or CS experience since we aim to cover each topic from the bottom up.

Professor Sahai notes in his unofficial advice page, “Set yourself a goal of understanding some system from top to bottom before you graduate. For example, you might want to know how the entire process of downloading and listening to an MP3 or OGG file works. This can be a good tool for integrating knowledge across different courses.” Since taking all of the courses may be infeasible and inefficient under the current system, we would like to provide a cocktail-party level of understanding.

Course Overview

This course will take students on a journey through YouTube starting from the browser request to the underlying circuitry and physics. In addition, staff will plan trips to various research labs or host talks by current researchers in the fields to provide deeper insight into various areas of research. The syllabus itself covers 11 weeks of instruction, including guest lectures by either faculty or a current researcher, lab tours, and other activities!

Each week will move lower on the Software/Hardware stack, with a deconstructive approach to software and a constructive one towards hardware. Sessions will consist of 2 hours of instruction, with interleaved interactive demonstrations and lecture.

Assignments and Grading

To pass the course a total of 17 points must be obtained by each student. Attendance and weekly submissions are worth 1 point each (there are 11 classes and 9 weekly assignments), and the final project is mandatory.

Homework
Homework should be submitted every week before class begins. These homework assignments are intended to reinforce the ideas presented in class, while being open-ended and rewarding. These assignments should not take many hours to complete. Students have the ability to choose from the following options every week for their homework assignment. Each option can be completed a maximum of four (4) times during the semester, to encourage diversification.

1. Students can submit a written assignment about a product or large-scale project outside of the course which makes use of the technologies discussed during the most recent class. Students will find websites, news articles, or other media to learn more about the product or project, and incorporate ideas from this course in the assignment (300-400 words).

2. Students can find out if there are any past or current research projects at UC Berkeley pertaining to what was taught in the previous class. Students will write details about a research project and explain how it relates to what was discussed in class. In addition, they will find other relevant research done at an institution outside of Berkeley, and compare it to Cal’s research. (300 - 400 words).

3. Students can design a piece of technology using the ideas from the most recent class. This can be in a variety of forms, whether it be a piece of code, a breadboard design, or something different. The student’s creation need not be complex, because of the introductory nature of this course, but it should be more than an idea. A short write-up must be included with the submission to explain the project (200 - 300 words).

4. Students can share their newly acquired knowledge with their peers (and show them how cool EECS is) by giving a friend an overview, demonstration, and/or lecture about the most recent class material. It is not sufficient to simply show class slides. A written analysis of the learning session must be included, explaining the experience, outcomes, and methods used for instruction (300-400 words).

**Final Project**

Students will need to submit an “overarching project” which will incorporate ideas spanning the entire semester. Tentative project details for this semester are as follows:

Students will work in a small group (no more than 3 students in a group) to envisage an imaginary company. This company will provide a product, service, or research that utilizes the technologies and tools you have learned throughout the entire semester. The group will

a. Develop prototype designs for the product or service. The product or service MUST include technologies or ideas from AT LEAST four different classes from the semester.

b. Create a “sales pitch” to convince investors to invest in the company. The sales pitch must reference the technologies used, specifically relating to course material.

c. Find a real-life company that is similar in purpose, product, or design, and analyze how that company uses technologies found in class material to further their goals. The group must also explain why their company would be a better investment. (note that the other company does not need to be a startup)

Additional details about this project will be released as the semester continues.
Schedule

Classes are held Wednesdays 6-8PM at 275 Soda, with the exceptions of the classes on 10/18 and 11/1, which will be held at 125 Cory.

Weekly topics and demos are subject to change.

Introduction - 9/6

This week will provide a quick preliminary overview of what the course will be about, distribution of course materials, as well as an introduction to the course staff.

Youtube Video - 9/13

A Tale of Momentum and Inertia

Youtube is one of the most iconic pieces of the modern internet. In this course, we will dissect the process of requesting a video, and playing it in the browser. During this class, we will watch the behavior of the browser, as well as analyzing the methods by which servers send data to the client. This session will also focus heavily on the mechanics of audio and graphics, as well as their representations.

In addition, we will perform a simple overview of audio signals and compression using various methods, such as SVD, Fourier Transforms, and Zip.

LAB: Audio, video, and networking demonstration

Mechanics Behind Youtube ML - 9/20

Machine Learning is the science of making software “learn”. We will introduce the big ideals behind machine learning and discuss how YouTube uses them to figure out which videos its users are more likely to watch. Along the way we'll gain some hands-on experience at designing and running your own learning algorithms.

LAB: YouTube Recommender System and Machine Learning Techniques

Security and Applications - 9/27

Applications serve as the interface between computers and people. These include web applications such as Youtube, or mobile applications such as Flappy Bird. Security sits at the heart of many applications, such as games or web browsers, and is achieved by utilizing information that only “authorized” users know. Vulnerabilities then appear when attackers are able to impersonate authorized users or exploit loopholes in the program.

LAB: Network and Web Security
Applications/OS Interface - 10/4

Applications run by writing code, but that code doesn’t automagically run on the physical computer. This session will cover Operating Systems, Compilers, and Binary -- the translation and execution of code from human-readable format to computer-readable format. This session will also cover hardware device interfaces, such as network sockets and drivers.

LAB: Simple Compiler for Circuits

How to Build a Computer - 10/11

Applications run on top of hardware, which is typically a set of wires and gates. In this session, we will examine digital logic, memory units, and the hardware-software interface. We will also examine digital circuit design, clocking, and hardware-peripheral interfaces.

LAB: Logisim CPU

Physics - 10/18

Transistors are the building blocks of computers, but what exactly are transistors? This session will describe the physics behind how transistors work, in a qualitative, simple way. We will build up from the ideas of bonding in silicon to different charge carriers in semiconductors and finally MOSFETs.

LAB: Visual simulations of MOSFETs

Fabrication - 10/25

Now that you know how a transistor works, how do you actually build one? What are the fundamental challenges involved in building transistors? In this week, we’ll go over a sample process, and talk about the basic chemical processes involved in creating a transistor, and the challenges involved in fabrication. In addition, we hope to be able to give a small lab tour of the EE143 lab so that students can see where transistors are fabricated.

LAB: TBD

Analog and Digital Circuits - 11/1

We’ve taken this dive through how modern computing works, but there’s an essential missing component, measurement! The real world is not discrete or digital, but luckily analog circuits are here to save the day. We will learn how analog circuits turn signals into digital information for computers, build an inverter out of transistors, and explore digital logic through an analog view.

LAB: Inverters and logic gates
Embedded Systems and Controls - 11/8

So far, this course has focused primarily on the top-to-bottom stack of a typical computing device. This session will cover physical interaction with the external world, as well as feedback from the environment. We will examine the system used by the EE16B robot, and also discuss other systems such as camera image stabilization.

LAB: Advanced Robotics, Camera Stabilizer

Conclusion/Ethics - 11/15

The final session will provide an overarching unification of the topics covered in the course, with review of how an application travels from the user interface to the wires. We will also cover non-technical aspects of EECS, such as ethics, morality, and policy.

Academic Accommodations

If you require disability-related accommodations in this class, have emergency medical information you wish to share with us, or need special arrangements in the case of evacuation from the building, please inform us immediately. Please see us privately after class or in the office.

Academic Integrity

Academic dishonesty and plagiarism will not be tolerated in this course. Plagiarism is the act of presenting other people's work or ideas as your own. It includes both directly copying someone else's work and summarizing the writing or ideas of another person without giving proper citation. In all of your assignments, you may use words or ideas written by other individuals in publications, websites, or other sources, but only with proper attribution. “Proper attribution” means that you have fully identified the original source and extent of your use of the words or ideas of others that you reproduce in your work for this course, usually in the form of a footnote or parenthesis.