



Microprocessor-based Systems (MicroPs) Syllabus

Contact Information

Instructor Prof. Josh Brake
Lab Assistant(s) Evan Hassman
Web page <http://pages.hmc.edu/brake/class/e155/fa21>
Email List eng-155@g.hmc.edu
Slack Workspace <http://micropsfa21.slack.com>

Be sure to join the class Slack workspace and check it regularly as it will be the main source of course-related communication for this semester.

Course Details

Name	Times	Location
Lecture	TR 1:20p - 2:35p	Shan 2460
Lab Checkoff	TR afternoons by signup (link)	Parsons B183
Office Hours	TBD or by appointment	Parsons 2364
Lab Hours	TBD	Parsons B183

You will be working on labs on your own time and it is not required that you attend the entire scheduled lab period. Instead, sign up for a time to get your lab checked off. Please sign up for a time during your lab section. If you are unable to find a spot that works for you, see if you can swap with one of your classmates. If you are still having trouble finding a time that works for you, reach out and let me know.

“Office hours” is code for “come hang out.” You are encouraged to attend office hours to ask questions, get help with your labs, talk about careers and graduate school, or chat about something on your mind (whether it is related to this class, academics, or anything else). In addition to things related to Engineering, embedded systems, digital electronics, and microcontrollers, a short (but not comprehensive!) list of things I enjoy talking about are sports (MLB: NY Mets; NFL: Tennessee Titans), running/biking/hiking, life design, time management, or books/essays I’m reading. I am available more often than not, so try dropping me a line via Slack or email if you are having a problem with your lab or want to set up a time. You may also contact the lab assistant(s) for questions when I am not available.

Course Learning Objectives

The overarching goal of this course is to take you from a basic familiarity and knowledge of digital design with field programmable gate arrays (FPGAs) and microcontroller programming and to expand your capabilities to design, build, and test embedded systems. In particular, by the end of this course you should be able to:

- Design and implement combinational and sequential circuits on an FPGA.
- Use an ARM-based microcontroller to interface with the real world via sensors and actuators.
- Build an embedded system project of your own design from the ground up.
- Select appropriate embedded hardware for a given task.
- Effectively and efficiently debug electrical systems with measurement tools such as an oscilloscope and logic analyzer.
- Define a problem, specify an embedded system which solves it, and deliver a system which meets them.
- Read and understand complicated datasheets at a level that enables you to incorporate them into your designs.

- Clearly communicate technical results in a professional manner through oral presentations and written reports.

In broad strokes, MicroPs can be divided into three parts. The two parts of the class focus on giving you fundamental embedded systems concepts in lecture which you learn by experience through seven, hands-on labs. One part is focused on the FPGA and the other on the MCU. These labs are designed to be loosely structured design projects – you will be given information about the required specifications and some pointers on how to get started, but much of the development process is left open to you. The final part of the class is focused on the project. The project gives you the opportunity to demonstrate independent and creative mastery of embedded system design in teams of two. The specific project task is very open-ended; the only requirement is that the project do something fun or useful and that it meaningfully uses both the FPGA and MCU. In addition to having a series of design review checkpoints, you will give a mid-project presentation to the class and a final demonstration of your project at the end of the course. The content of the lectures in the second half of the class focuses on exposing you to more advanced embedded systems concepts and exploring a range of various types of external hardware that may be useful in your project.

My Teaching Philosophy

I love teaching at Harvey Mudd and having the opportunity to learn alongside all of you in the classroom and lab is one of the best parts of my job. My ultimate goal for this course is to help you to master the material and become skilled embedded systems developers who understand how to build a system from a set of requirements and specifications and to verify that the system meets those specifications. I am continually reading and learning from the literature on effective pedagogy and working to implement best practices in my teaching.

Here are a few of the main pedagogical concepts that you can expect to see in this course.

- **Transparent Teaching** – you should not have to guess what you are supposed to get out of a given activity or assignment in this class. I strive to be as transparent about why we are doing what we are doing. This is most clearly articulated through explicit learning goals that accompany each lecture, lab, or project. If the purpose of anything we do in this class is ever unclear, please ask me and I will be happy to clarify it.
- **Psychological Safety** – A psychologically safe environment is one where each person feels able to share their questions, concerns, or mistakes without feeling embarrassed or looked down upon by others. Building a psychologically safe classroom is a joint venture which I expect each of you to join with me in pursuing.
- **A Growth Mindset** – growth can only happen when you reach the end of what you already know. Each of you coming into this class will have different levels of experience with the types of skills that are useful in this class like working with embedded systems, programming, debugging, design, etc. My goal as an instructor is to take you from wherever you are and bring you as far along in your journey as possible. To do that, I encourage each of you to ask questions and push yourself to the edge of your knowledge. It can be frustrating or embarrassing to have questions that you feel you should already have the answer to. You should take these moments as opportunities to ask questions and fill the gaps in your knowledge. In the end, having a growth mindset is strongly linked with deep curiosity about what you are learning and a realization that while being honest about the limits of your understanding might be challenging, embracing that discomfort allows you to grow and improve quickly.
- **Frequent, Low-stakes Testing** – Research has shown that having frequent opportunities to assess your knowledge is a powerful way to learn and correct any misconceptions. One way that this will take place is through regular in-class quizzes and activities.
- **Interleaving** – The scheduling of the material is arranged in such a way that you will return to many similar concepts throughout the semester with some time in between. For example, the labs are arranged so that you alternate between labs that focus on the FPGA and the MCU as opposed to having all the FPGA-focused labs followed by all the MCU-focused labs in blocked fashion. The goal of this is to have you return to familiar concepts after some time away so that you reinforce the material after working on something else. This has been shown to improve long-term retention of the material (see *Small Teaching* by James Lang if you are curious in learning more).



Recommended Texts

I assume you have a mastery of digital design at the level of E85; review *Digital Design and Computer Architecture, ARM Edition* if you feel rusty on a topic. There are copies available in the engineering lounge and digital lab as well as online access via the library ([link](#)). If you are considering a career in this field, this book is well worth the investment if you haven't purchased it already.

We will not be closely following a textbook, but you may find *The Definitive Guide to ARM® Cortex®-M3 and Cortex®-M4 Processors* by Joseph Yiu to be a helpful, comprehensive reference for the microcontroller. An electronic version of the book is available through the library ([link](#)). If you are looking to deepen your knowledge of C Programming I highly recommend taking a look at *Expert C Programming* by Peter van der Linden ([link](#)). In particular it has some excellent discussions about the intricacies of the C related to declarations (chapter 3) and pointers (chapters 4, 9, and 10). Both of these texts are available to you for free through the library as part of the O'Reilly online learning platform.

Schedule

Week	Tuesday's Class	Thursday's Class	Due
8/30	Intro & Analog Behavior of Digital Systems	Combinational and Sequential Logic	(Lab Demos)
9/6	Verilog Coding	Synchronous Design	Lab 1 - FPGA Board
9/13	Toolchain & Assembly Language	Microcontroller Programming	Lab 2 - Multiplexed 7-segment Display
9/20	FPGA & MCU Datasheets	TBD	Lab 3 - ARM Assembly Sort
9/27	C Programming	Peripherals	Lab 4 - Keypad
10/4	Internet of Things	Interrupts	Lab 5 - Digital Audio
10/11	Advanced Encryption Standard (AES)	Project Kickoff	Lab 6 - The Internet of Things
10/18	Happy Fall Break! No Class	PCB Design - Pt. 1	
10/25	PCB Design - Pt. 2	Motors and Speakers	Lab 7 - AES
11/1	Graphics and Displays	Interrupts	Project Proposal
11/8	Direct Memory Access (DMA)	Intro to Real Time Operating Systems	Proposal Debriefs
11/15	TBD	Advanced MCU Topics	
11/22	Presentations	Presentations	Project Status Report & Demo
11/29	Presentations	Happy Thanksgiving! No class	
12/6	Interview Questions		Project Checkoffs & Report & Demo Day Party

Grading

Item	Percentage
Labs	50%
Final Project	45%
Participation	5%

Your grade in the class is based on seven labs and a final project. You may have an extension of up to one week on one of your labs; otherwise, late work is not accepted except in extenuating circumstances such as illness. Your lowest lab score will have half the weight of the others. Labs are done individually but you are welcome to discuss them with other students or the instructor after you have made an effort by yourself. Please list the names of other students you have worked with on your lab report. Do make sure that if you work with others you turn in your own work, not work identical to that of another person. It is an honor code violation to simply copy someone else's work. Solutions to labs from past years labs have been handed out and may be floating around the web. Obviously, it is also an honor code violation to refer to these solutions while doing your lab.

Each lab is accompanied by a brief written report which should include the following:

- An overview which must not exceed one page, that provides a summary of the work, context for the other materials, an explanation of major design decisions, and a succinct explanation of whether the lab met requirements or how it deviates.
- Schematics of the circuits constructed for the lab. Use standard symbols for standard components, and be sure the schematic is sufficient for another engineer to reproduce your circuit.
- All code written for the lab.
- Figures that illustrate the output of the circuits or other results of the lab.

The purpose of these deliverables is to train you in professional, technical communication, so they will be held to high standards. All schematics must feature neat layouts and wiring, up-to-date values for every component, and ports labeled with both the pin and the signal associated with them. Figures must be readable and must include labeled axes and legends. Code should be commented; omit library code that is provided elsewhere. The report should be grammatically correct, succinct, tightly focused, and complete. Properly referencing figures will help keep the report brief.

We will use Sakai to collect lab submissions. Each lab should be submitted as a single PDF with any schematics, source code, or figures included.

Labs are graded on a 7-point scale. 3 points are given for the system meeting its specified requirements and 3 points are given for the report, schematics, and code. The rubric is as follows:

Points	Criteria
3	Meets all requirements & no errors
2	Minor errors
1	Serious errors
0	Not attempted

The last point for the lab grade is given for answering a fault tolerance question (FTQ). Many students often feel some stress about the FTQs, but these are designed to test the depth of your understanding about the concepts explored in the lab. Take these as an opportunity to practice for interview-type situations where you may not have much time to process and respond to a question. The goal of the FTQs is to see not only if you can answer the problem correctly, but to train you to think on your feet and help you practice answering questions about something that you may not immediately know from memory.

The final project will be done in groups of two. More information about the final project will be provided in the middle of the semester, but begin thinking now about who you might want to work with and begin brainstorming projects. You may see the course webpage for a listing of projects from previous years for inspiration. The final project is designed to give you a great deal of flexibility in choosing something you are interested in – the main requirement is that your project uses both the microcontroller and FPGA meaningfully to do something useful or interesting. Using the FPGA meaningfully is often the most difficult part so think about this carefully. You are more than welcome to discuss potential project ideas with me at any time.

The participation portion of your grade will be awarded for attending and engaging during class time and for completing various in-class activities. I put significant time and effort into making the most of the lecture periods we have together and hope that you also can bring a positive and enthusiastic attitude as we learn together this semester.

COVID-19 Policies

As we resume in-person instruction this fall there are still many open questions about the impact that COVID-19 may have on the semester. As an instructor, my ultimate goal is to offer the best educational experience possible while making an effort to keep everyone safe and healthy. Please make sure to follow all of the Mudd COVID-19 policies in the classroom and lab spaces, especially related to masking and social distancing in order to maximize the safety for everyone using those shared spaces. If you are feeling sick, please let me know and don't attend class. We will work out a plan for catching you up.



Lab Kit

While there is not a textbook to purchase, you will need to buy a lab kit. The fee is \$75 in Claremont Cash, and should be paid by filling out the Google Form ([link](#)) which authorizes Sydney Torrey in the Engineering office to charge your Claremont Cash account. Once you have paid for your kit via the form, see Sam Abdelmuati in the stockroom to pick up your kit. If you cook your board this semester, you can buy and rebuild a replacement, but ask the instructor for help troubleshooting first. You'll also check out a large breadboard from the stockroom, and will need to return it at the end of the semester.

The kit fee can be waived in cases of financial hardship. To request a waiver fill out the form here ([link](#)). Course instructors will not know about waiver requests.

Lab Access

The Digital Lab (Parson B183) is available for you to use when working on your labs. The current door code is 31-4-5. There are Windows PCs available with VS Code and Quartus installed along with the drivers required to program your board. There is electronics assembly equipment needed to solder and a number of oscilloscopes at the lab stations. There is also a lab cabinet with various resistors and some of the parts like wires, seven-segment LEDs, and transistors you will need for some of your labs. You are welcome to use these while working on your lab, but please make sure to return the components to the lab cabinet when you are done.

In addition, the software we will be using for programming the MCU (PlatformIO in Visual Studio Code) and FPGA (Intel Quartus Lite) are free and supported on a variety of platforms. PlatformIO is supported on Windows, MacOS, and Linux and Quartus is supported on Windows and Linux. If you are running MacOS, you can download and virtualize Windows using VMWare Fusion Player under a Personal Use License for free. More details and download links can be found here ([link](#)).

Honor Code Violations

It goes without saying that students in this class are expected to follow the honor code. An honor code policy appears below and prescribes behavior that is considered honorable, so read those maxims and follow them closely. Any honor code violations will be handled through JB.

Honor Code Policy

1. All students enrolled in this course are bound by the HMC Honor Code. More information on the HMC Honor Code can be found in the HMC Student Handbook.
2. It is your responsibility to determine whether your actions adhere to the HMC Honor Code. If this document does not clarify the legitimacy of a particular action, you should contact the course instructor and request clarification.
3. Work you submit for individual assignments should be your own, and you should complete all assignments based on your own understanding of the underlying material. If you work with, or receive help from, another individual on an assignment, provide a written acknowledgement in complete sentences that includes the person's name and the nature of the help. This document is not meant to be an exhaustive list of every possible Honor Code violation. Infractions not explicitly mentioned here may still violate the Honor Code.
4. **Boundaries of Collaboration** Verbal collaboration with other students on individual assignments is encouraged AFTER you have given serious thought to each component yourself. However, all submitted written work should be written by yourself individually, and not a collaborative effort or copied from a common source (e.g., a chalkboard). It is NOT acceptable to work on labs in lockstep with another classmate.
5. **Use of Computer Software** The use of graphing calculators and computer software to aid in course work is acceptable, as long as it does not substitute for an understanding of the course material.



6. **Use of Web Resources** The use of Internet resources to aid in course work is acceptable, as long it does not substitute for an understanding of the course material. Plagiarism and direct copying from online (or any other) sources is strictly prohibited.
7. **Use of Your Own Work from Previous Semesters** If you have previously attempted this course, you may resubmit your work from previous semesters as this semester's coursework, as long as you understand the underlying material.
8. **Use of Other Course Resources from Previous Semesters** You may not reference assignments (labs, problem sets, activities) of this course from previous semesters.
9. **Retention of Course Resources** Assignments and exams from this course may not be committed to dorm repositories or otherwise used to help future students.

Inclusiveness and Harassment

We do difficult work in this class and everyone should feel comfortable engaging with the material. We explicitly want students to feel safe doing this work, so it is worth stating that the instructors are committed to making the class a safe space for everyone regardless of race, gender, ethnicity, sexual orientation, religion, and academic history. If you feel that you are experiencing a hostile environment, speak to an instructor immediately.

Educational Accessibility

HMC is committed to providing an inclusive learning environment and support for all students. As we return back to in-person instruction, we recognize that the challenges facing students may be different and student accommodation needs may change. Students with a disability (including mental health, chronic or temporary medical conditions) who may need accommodations in order to fully participate in this class are encouraged to contact the Office of Accessible Education at access@g.hmc.edu to request accommodations. Students from the other Claremont Colleges should contact their home college's Accessible Education officer.