E85: Digital Electronics & Computer Engineering

Spring 2025

Syllabus

Teaching Team

Professor:	Josh Brake	Parsons 2364	jbrake@g.hmc.edu
	David Harris	Parsons 2374	David_Harris@hmc.edu
Office Hours	Brake	Tu 4-5, Th 10-11	Digital Lab
	Harris	M 3-4, W 10-11, Th 3:30-4:30	Parsons 2374

Tutoring Hours (Digital Lab)	Mon 2:45-3:45 pm	Marina Bellido (TBΠ)
	Mon 9-10 pm	Vikram Krishna
	Tue 3-4 pm	Corey Hickson (TBΠ)
	Tue 7-8 pm	Jessica Liu
	Wed 7-8 pm	Roman De Santos
	Wed 8-9 pm	Jordan Carlin (TBΠ)
	Sun 2-3 pm	Troy Kaufman
	Sun 7-8 pm	Madeleine Kan

Schedule

Lecture: TR 1:15 – 2:30 Shan 3481

Feel free to stop by even if we do not have official office hours. One of the main reasons that we teach at Harvey Mudd is that we value working with students 1-on-1 and in small groups.

Text

You will get the most out of class if you do the reading before lecture. See the schedule for recommended reading. Copies of the textbook are available in the lab and in the Engineering lounge.

Harris & Harris, *Digital Design and Computer Architecture, RISC-V Ed.*, Morgan Kaufmann 2022.

Lab Kits

Please check out a breadboard from the stockroom for the semester, or use a personal breadboard if you prefer. One column about 6" tall is sufficient. You will use components from the E85 supply cabinet in the Digital Lab. Please return them to the cabinet if they are in good condition afterward. We have single lab set of RED-V boards and they are no longer in production, so they cannot leave the lab. If you have a friend who took E85 in the past and still has a lab kit, they might sell you their board. The accelerometer boards and LED matrix displays also cost enough that we'd prefer to keep them in the lab and reuse them. If you think you may have damaged a component, please put it in the "Possibly Damaged" box in the supply cabinet.

Electronic Communication

Class web page: http://pages.hmc.edu/harris/class/e85
Class email list: eng-85-1-2025-sp@g.hmc.edu, eng-85a-1-2025-sp@g.hmc.edu

You also will need a Harvey Mudd College computer accout to complete your labs. If you are not an HMC student, email me your full name and school affiliation and I will request an account for you.

Course Objectives

Digital systems have revolutionized our world. From television to cell phones to GPS to warfare to medicine to automobiles, computers and digital processing have reshaped the way we live and work. Computers are also a vital part of daily practice in every field of science and engineering. Some of you will practice as computer engineers and will use this material daily in your career. Many of you will focus primarily on other problems, but this class is intended to give you a set of tools, similar to math, science, programming, and writing, that will help you incorporate sensors, actuators, and microcontrollers as a component of a larger system, or understand the vocabulary, operations, capabilities, and limitations of digital systems when you collaborate with specialists.

Previous generations of engineers learned the "nuts and bolts" of the profession by doing things like disassembling and rebuilding engines. As technology has advanced, cars have become too complicated for the layperson to work on. Ironically, the same advances have made computers much easier to build. While most fields of engineering require extensive mathematics and complicated analysis of even rather simple components, digital systems merely require counting from 0 to 1. Their challenge, instead, is in combining many simple building blocks into a complex whole. Field programmable gate arrays (FPGAs), containing the equivalent of thousands or millions of logic gates, make it possible to build these complex systems in the lab without the tedium of manually connecting components. In this class, you will build your own microprocessor and test it on a FPGA. In the process, you will master the art and science of digital design. You will learn to speak to and control processors in their native tongue, assembly language. And you will put all the pieces together to demystify how a computer works.

As you probably know, very few complex systems work the first time you put them together. Engineers must become good at systematically and efficiently debugging their creations. One of the course objectives that can be frustrating but vitally important is to learn to teach yourself professional-strength computer-aided design tools and to use these tools to debug systems.

By the end of this course, you should be able to:

- Build digital systems at all levels of abstraction from transistors through circuits, logic, microarchitecture, architecture, and C culminating with implementing and programming a microprocessor soft core on a field programmable gate array.
- Manage complexity using the digital abstraction, data types, static and dynamic disciplines, and hierarchical design.
- Design and implement combinational and sequential digital circuits using schematics and hardware description languages.
- Program a commercial microcontroller in C and assembly language and use it in a physical system with sensors and actuators
- Begin the practice of implementing and debugging digital systems with appropriate lab techniques including breadboarding, interpreting datasheets, and using field-programmable gate arrays and microcontroller boards, simulators, debuggers, and test-and-measurement equipment.

Grading E85

In-class Activities	5%
Labs:	30%
Problem Sets:	20%
Midterm:	20%
Final:	25%

E85A

In-class Activities	5%
Labs:	30%
Problem Sets:	20%
Midterm:	45%

Lab 11 is the capstone of the labs, in which you design and simulate a microprocessor, drawing on most of the skills you have acquired over the semester. **You must turn in a working Lab 11 to pass E85**.

Solutions to the labs and problem sets from previous semesters are undoubtedly floating around campus and on the web. You may **not** refer to solutions while doing the assignments; they must be your own work. Many of the labs build on previous labs. If you do not turn in a lab, you may refer to the solutions handed out to work through the lab you missed to learn the skills needed for a subsequent lab. However, you may not simply copy another student's files.

Labs and homework are due by the end of class. You may have a one-week extension on one assignment of your choice (except Lab 11) and are responsible for tracking this yourself under the honor code (no need to notify the instructor; just turn it in with the following week's assignment). Your lowest problem set and lab grade will each be dropped. Please ration your extension and drops carefully lest you find yourself ill at the end of the semester and out of options. The class moves quickly and it is difficult catching up if you fall behind. Contact the Academic Deans (academicdeans@g.hmc.edu) if you have or anticipate a protracted situation; they can coordinate additional flexibility for this class and/or others.

You are welcome to discuss labs and problem sets with other students or with the instructor or lab assistants or tutors **after** you have made an effort by yourself. However, you must turn in your own work, not work identical to that of another student. For labs, asking classmates or tutors for help when you are stuck on a specific issue is encouraged (especially on difficulties with the tools and equipment), but sitting at adjacent computers and working through the lab together in lock-step is specifically prohibited. Pair/group programming is also prohibited, as is getting your answer from an AI, search engine, or other source that bypasses your own critical thought and opportunity to practice the material. Be sure to credit at the top of your assignment anyone with whom you discussed ideas. **It is an honor code violation to simply copy someone else's work.**

An exception to the AI policy is that each week's problem set will have one optional problem in which AI is required rather than prohibited. You can use ChatGPT or any other AI tool. The question will be worth one point of extra credit, not to bring the problem set total above 100%. You may photograph the question into the AI app, or cut and paste it into the prompt. Learning objectives include:

- Begin to understand what AI is and is not capable of answering in this technical discipline
- Critically assess the quality of the answer, including whether the AI is hallucinating, and whether the solution is similar, better, or worse than what you would have done yourself in a reasonable amount of time.
- Explore modifying your prompt to get a better answer if the initial answer is unsatisfactory.

Readings for each lecture are listed on the schedule below. Many students say they have found the readings valuable and enjoyable. You'll get the most out of the class if you read the sections in advance of the lecture and come with questions, and then reread as necessary when you work your problem sets and labs.

Health and Wellness

College students often experience issues that may interfere with academic success such as academic stress, sleep problems, juggling responsibilities, life events, relationship concerns, or feelings of anxiety, hopelessness, or depression. If you or a friend is struggling, we strongly encourage you to seek support. Helpful, effective resources are available on campus, at no charge.

- If you are struggling with this class, please visit during office hours or contact me by email at David_Harris@hmc.edu.
- Check-in with an academic dean if you are struggling in courses or unsure what academic resources are available at HMC by emailing academicdeans@g.hmc.edu.
- Reach out to The Office of Health and Wellness to discuss options available by emailing wellness@hmc.edu
- **MCAPS provides crisis support services 24/7/365. Students can call us at 909-621-8202 and press "1" at the prompt to speak with a crisis counselor.** Other prompt options are available for those not in crisis.

Tentative Schedule

Lecture	Date	Topics	Readings	Assignment
0	1/21	Introduction: digital abstraction, numbers	1.1-1.5	
1	1/23	Logic gates, Static discipline, transistors	1.6-1.9, A1-A7	
10	1/28	Combinational logic design	2.1-2.8	PS 1 due
11	1/30	Timing, sequential circuits	2.9-2.10, 3.1-3.2	Lab 1 due Digital Circuits
100	2/4	Finite state machines	3.3-3.4	PS 2 due
101	2/6	Dynamic discipline, metastability	3.5-3.7	Lab 2 due Comb Logic
110	2/11	Hardware description languages: Verilog	4.1-4.3	PS 3 due
111	2/13	Verilog, Part II	4.4-4.10	Lab 3 due Structural FSM
1000	2/18	Arithmetic circuits	5.1-5.2	PS 4 due
1001	2/20	Fixed and floating-point number systems	5.3	Lab 4 due Behavioral FSM
1010	2/25	Sequential building blocks, arrays	5.4-5.7	PS 5 due
1011	2/27	Midterm Review		Lab 5 due Music keyboard
	3/4	Midterm		Midterm Due 2:30 pm
1100	3/6	C Programming	C.1-C.7	
1101	3/11	C Programming	C.8-C.11	
1110	3/13	Microcontrollers: Memory-mapped I/O	9.1-9.3.3	
	3/18	Spring Break! NO CLASS		
	3/20	Spring Break! NO CLASS		
1111	3/25	Parallel & serial interfacing, ADCs	9.3-9.4	
10000	3/27	I/O libraries and examples		Lab 6 due C Programming
10001	4/1	Assembly language	6.1-6.3.6	PS 6 due
10010	4/3	Function calls	6.3.7-6.3.8	Lab 7 due C I/O
10011	4/8	Machine Language	6.4-6.9	PS 7 due
10100	4/10	Single-cycle processor datapath	7.1-7.3.1	Lab 8 due C Peripherals
10101	4/15	Single-cycle processor control, Verilog	7.3, 7.6	PS 8 due
10110	4/17	Multicycle processor	7.4	Lab 9 due Assembly
10111	4/22	Pipelining	7.5.1-2	PS 9 due
11000	4/24	Advanced architecture: a sampler	7.7	Lab 10 due Multicycle Control
11001	4/29	Case study: Processors	6.7, 8.7, 8.5	PS 10 due
11010	5/1	Class summary and review		Lab 11 due Multicycle CPU