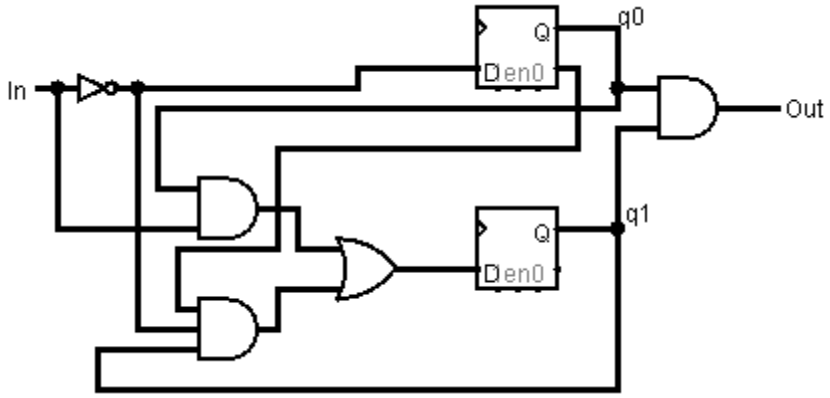
PHYS-234: Digital Elec/Microprocessors - Fall 2026

Jed Brody



Digital electronics is the invisible infrastructure of modern civilization, and it's all based on just two numbers: 0 and 1.

Course Objectives

You will learn how to design and analyze digital circuits, including a simple computer, starting from the smallest building blocks: transistors and logic gates. You will learn the Verilog hardware description language (HDL) to implement your circuits on field programmable gate arrays (FPGAs).

Course Competencies

After completing this course, you will be able to:

- Design logic gates made of CMOS transistors
- Use K maps to minimize logic equations
- Design and analyze multiplexers, arithmetic logic units, flip-flops, and other common digital circuit elements
- Design Moore and Mealy machines to detect sequences of 0's and 1's
- Write simple programs in MIPS assembly language and machine language
- Design circuits to transmit and receive serial data
- Design and synthesize circuits on FPGAs using the Verilog hardware description language

Instructor Contact Info

- email: jbrody@emory.edu. I reply within 24 hours on weekdays. Please contact me as often as you like! If you have lots of questions about the course, you're welcome to email me 10+ times a day!
- Office hours: Unlimited, by appointment or drop in, in N308. Please feel free to make appointments whenever you want.

Useful Links

- Textbook: Harris and Harris, Digital Design and Computer Architecture, 1st edition, 2007. **The Emory library has unlimited copies!**
- Solutions to odd-numbered exercises (link will be on Canvas)
- MIPS instruction reference (for the second half of the semester)
- Asynchronous lectures from 2020 (link will be on Canvas)

Grading

Lab reports: 44% (4% for each of 11 lab reports)

Projects: 16% (8% for each of 2 projects)

Tests: 40% (20% for each of 2 tests)

Lab Reports

You are encouraged to work with a lab partner on the labs and projects, but you'll write your lab reports (and project reports) by yourself. Lab reports may be brief and do not need to adhere any specific format. Just be sure to:

- Include your Verilog code (except in some of the later labs that use code from the textbook or elsewhere).
- Include your observations of outputs as function of inputs, as a truth table if appropriate.
- Include anything else specifically requested in the lab manual.

Late Policy

- If you submit a reasonable first draft by the deadline, you may submit as many additional drafts as you like until Dec. 15. I'll provide feedback on all drafts.
- If you miss the deadline, you get only one draft, which must be submitted by Dec. 15. Otherwise, there's no late penalty.

	Topic (approximate)	Textbook sections (in HDL examples, ignore VHDL)	Due	Optional ungraded exercises
8/27	Binary arithmetic; logic gates	1.4-1.5, 4.1.1, 4.2.1, 4.2.7		
9/1	Lab 1			
9/3	Transistors; Boolean algebra	1.6-1.8, 2.1-2.6		Ex. 1.19, 1.33, 1.35, 1.37, 1.51
9/8	Lab 2		Lab 1	
9/10	K maps; always blocks	2.7-2.8, 4.2.5, 4.5.1, 4.5.2		Ex. 2.1, 2.5, 2.15, 2.17; Ques. 2.1
9/15	Lab 3		Lab 2	
9/17	Sequential logic	2.9, 3.1-3.3, 4.4		Ex. 2.19, 2.29, 4.1, 4.5, 4.21
9/22	Lab 4		Lab 3	
9/24	Finite state machines	3.4, 4.6		3.1, 3.3, 3.7, 4.25, 4.45
9/29	Lab 5		Lab 4	
10/1	Circuits making sounds			
10/6	Project 1		Lab 5	
10/8	Test 1			
10/15	Arithmetic operations	4.3, 5.2.1-5.4.1		
10/20	Lab 6		Project 1	
10/22	Memory; MIPS assembly language	4.8, 5.5, 5.6.1, 6.1-6.2		Ex. 5.9, 5.11, 5.25, 5.27, 5.29
10/27	Lab 7		Lab 6	
10/29	Machine language	6.3, 6.4.1-6.4.2		Ex. 5.45, 6.3, 6.9, 6.11, Ques. 6.1
11/3	Lab 8		Lab 7	
11/5	High-level code constructs	6.4.3-6.4.6, 6.5		Ex. 6.7, 6.19, 6.23, 6.25
11/10	MIPS microarchitecture	7.1-7.3	Lab 8	Ex. 6.31
11/12	Lab 9			
11/17	Lab 10		Lab 9	
11/19	Test 2			
11/24	Serial communication (UART)			
12/1	Lab 11		Lab 10	
12/3	Timing Diagrams	3.5.1-3.5.3, 3.6		Ex. 3.19, 3.23, 3.27, 3.29, 4.23
12/8	Project 2		Lab 11	
12/15			Project 2 (in lieu of final exam)	