ECE 6745 Complex Digital ASIC Design, Spring 2025
Course Syllabus

School of Electrical and Computer Engineering
Cornell University

revision: 2025-01-22-20-59

1. Course Information

Credit	4 credits, letter grades only
Prereqs	ECE 4750 Computer Architecture
Instructor	Prof. Christopher Batten, `cb535` Office Hours: 323 Rhodes Hall, Wednesday, 4:30–5:30pm
Lectures	114 Gates Hall, Tuesday and Thursday, 1:00–2:15pm (until spring break)
Section	225 Upson Hall, Friday, 3:35–4:25pm (until spring break)
Required Materials	Neil H. E. Weste & David M. Harris "CMOS VLSI Design: A Circuits and Systems Perspective" 4th edition, Addison Wesley, 2010
Website	`http://www.csl.cornell.edu/courses/ece6745`

2. Description

This course aims to provide a strong foundation for students to understand the principle and practice of designing, implementing, testing, and evaluating complex standard-cell ASIC chips using automated state-of-the-art electronic design automation (EDA) tools. This course is at the intersection of computer architecture, digital circuits, and EDA and is suitable for students pursuing careers in both research and industry. For students pursuing research topics in computer architecture, the course will provide deeper insight into critical physical design issues for future computing systems, while for students pursuing research topics in digital circuits, the course will provide system-level insight into future large-scale chip designs. For students pursuing a career in the chip-design industry, the course will provide valuable design experience from architecture to digital circuits.

The course is divided into four parts: the first three parts are lecture-based, while the final part is an extensive design project. The first part provides an overview of ASIC design including: hardware description languages; CMOS devices; CMOS circuits; full-custom design methodology; automated design methodologies; testing and verification; packaging and I/O. The second part provides a deeper study of CMOS digital-circuit fundamentals including combinational logic, sequential state, and interconnect. The third part provides a deeper study of EDA algorithms including synthesis algorithms and physical design automation algorithms. The final part is an open-ended design project where small groups of students design, implement, test, and evaluate an interesting technique in computer architecture using functional-, microarchitectural-, register-transfer-, and layout-level modeling. This five-week long design experience will include weekly project meetings with the course instructors, milestone documents, a project demonstration, and a detailed final report.

3. Objectives

The field of computer systems can be visualized as a stack of abstraction and implementation layers with application requirements at the top and technology constraints at the bottom. The intermediate layers include devices, circuits, gate-level design, register-transfer-level (RTL) design, microarchitecture, instruction set architecture, compilers, operating systems, programming languages, and algorithms.

ECE 4750 Computer Architecture is in the middle of this stack and focuses on the fundamentals of designing processors, memories, and networks, and apply this knowledge through a series of lab assignments. Students gradually design, implement, test, and evaluate a simple multicore system capable of running parallel microbenchmarks at the register-transfer level. The lab assignments focus on cycle-level performance (i.e., the impact a technique has on the number of cycles it takes to execute a program). Although ECE 4750 teaches students some basic principles involved in evaluating the cycle time, energy, and area impact of various design decisions, they do not have an opportunity to put this into practice. In addition, the focus of ECE 4750 is firmly on general-purpose subsystems as opposed to application-specific subsystems. At the opposite end of the computer engineering spectrum is ECE 4740 Digital VLSI Design. This course teaches students the fundamentals of digital circuit design, but the scope of these courses is on small custom-designed subsystems involving hundreds of transistors.

This course bridges the gap between computer architecture and digital circuits. Students will learn how to take the RTL designs from ECE 4750 and use automated tools to generate realistic layout. The course will enable students to quantitatively evaluate the cycle time, energy, and area impact of the techniques they learned in ECE 4750. ECE 4750 and this course dovetail nicely together creating a year-long digital design experience for students. By the end of this course, students should be able to:

- **describe** concepts related to the overall ASIC design methodology, CMOS digital circuits, and EDA algorithms and explain how these concepts interact.
- **apply** this understanding to new ASIC design problems within the context of balancing application requirements against technology constraints; more specifically, quantitatively assess a design's execution time in cycles, cycle time, area, and energy.
- **evaluate** various design alternatives and make a compelling quantitative and/or qualitative argument for why one design is superior to the other approaches.
- **demonstrate** the ability to implement and verify designs of varying complexity at the register-transfer level and to push these designs through a commercial standard-cell ASIC flow.
- **create** new baseline and alternative designs at the register-transfer level, the associated effective testing strategies, and a thorough evaluation plan.
- **write** comprehensive technical reports that describe designs implemented at the register-transfer level and evaluate the designs to determine the superior approach.

4. Prerequisites

This course is targeted towards advanced senior undergraduates, M.Eng. students, and first-year Ph.D. students. ECE 4750 is a prerequisite for all undergraduates and M.Eng. students. Students are more likely to be successful in this class if they did well in ECE 4750. Since ECE 4750 is a prerequisite, students are expected to be very proficient with Linux and the command line, implementing designs using well-structured synthesizable register-transfer-level models, writing test harnesses, and evaluating designs using simulators. Students are expected to be familiar with all of the ECE 4750 lab assignments.

We will be using a combination of both Verilog and Python in the lab assignments and design project. Those students with less experience working with the Verilog hardware description language are strongly encouraged to read Chapter 4 in *"Digital Design and Computer Architecture, 2nd edition"* by D. M. Harris and S. L. Harris (Morgan Kaufmann, 2012), and/or to review *"Verilog HDL: A Guide to Digital Design and Synthesis, 2nd edition"* by S. Palnitkar (Prentice Hall, 2003). Students which have never used Python before may want to spend additional time reviewing the textbook titled *"Think Python: How to Think Like a Computer Scientist"* by A. B. Downey (Green Tea Press, 2014).

Note that we will cover enough circuits and EDA in this course such that advanced circuit-level or EDA courses are not prerequisites. However, those students that have taken such courses will be able to see how digital circuits are composed into much larger multi-million transistor designs, and how EDA algorithms can be used in practice.

5. Required Materials

The required textbook for the course is Neil H. E. Weste and David M. Harris, "CMOS VLSI Design: A Circuits and Systems Perspective, 4th edition," Addison Wesley, 2010. Please use the 4th edition, since there have been significant changes compared to earlier editions. This book is available through the Cornell Academic materials Program and the Cornell library. There will be occasional assigned readings from the book, but more importantly this is an excellent book that all serious digital ASIC designers should have on their bookshelf.

6. Format and Procedures

This course includes a combination of lectures, discussion sections, laboratory assignments, a midterm, and a five-week design project. The design project includes a preproposal, proposal, weekly meetings, milestone documents, demonstration, and final report. Students are expected to work with a partner on the lab assignments and the design project. Assessment rubrics for the lab assignments and design project will be distributed early in the semester.

- **Lectures –** Lectures will be from 1:00pm to 2:15pm every Tuesday and Thursday in 114 Gates Hall until spring break. There will be no lectures after spring break. We will start promptly at 1:00pm so please arrive on time. Students are expected to attend all lectures, be attentive during lecture, and participate in class discussion. Please turn off all cellular phones during class. Use of cellular phones and laptops during lecture is not allowed (see Section 9.B).
- **Discussion Section –** There will be a discussion section most Fridays before spring break from 3:35pm to 4:25pm in 225 Upson Hall. These discussion sections will be relatively informal, with the primary focus being on facilitating student's ability to complete the lab assignments

and prepare for the design project. There may be some problem-based learning activities during these discussion sections.

- **Lab Assignments –** The course will include two lab assignments that allow students to begin quantitatively evaluating area, cycle time, cycle counts, and energy consumption for various design alternatives. Students must work with a partner (see Section 9.E for collaboration policy). Students will be using the ECE Computing Resources to complete the lab assignments, the lab code must be submitted via GitHub, and the lab report must be submitted in PDF format via the online Canvas assignment submission system (see Section 10). No other means of submission or electronic format will be accepted.
- **Midterm –** The course includes an 180-minute midterm that covers all of the lecture material except for the final lecture which is on the same day as the midterm. If students have a scheduling conflict with the exam, they must let the instructor know as soon as possible, but no later than one week before midterm. Usually the only acceptable scheduling conflict is due to another exam at the same time. Exceptions can only be made for serious illness or family emergency. Taking the midterm is a requirement for passing the course. Graded exams and the exam solutions are only available for review under the supervision of a course instructor. You may not remove your graded exam, nor may you remove the exam solutions.
- **Design Project Preproposal/Proposal –** Students are required to submit both a project preproposal and proposal as PDFs via the online Canvas assignment submission system (see Section 10). Students are strongly encouraged to discuss their project ideas with the instructors before the preproposal is due. The preproposal will be reviewed by the course instructors and feedback delivered to the students to factor into their proposal.
- **Design Project Meetings –** After spring break, each project group will meet with the course instructors once a week during the regularly scheduled course meeting times. Project meetings will be held in the instructor's office, 323 Rhodes Hall. Students are expected to show up on time and be prepared for the meeting. Students will be asked to demonstrate progress by remotely logging into the course computing resources and running unit tests or small experiments on their design.
- **Design Project Milestones –** There will be three project milestone documents due after spring break. Each milestone document must be submitted as a PDF via the online Canvas assignment system (see Section 10). Each project milestone document focuses on a portion of the design project final report; so an initial draft of the final report can consist of simply assembling the project milestone documents into a coherent narrative.
- **Design Project Demonstration –** During the final week of classes, students will schedule time to meet with the instructors and demonstrate their design project. This demonstration is a key part of the project assessment. Students should prepare a step-by-step demonstration that illustrates the project's code quality, functionality, and illustrates some of the results discussed as part of the project report.
- **Design Project Report –** The project report is due during the final exam period according to the final project schedule published by the University. Students are not allowed to make significant changes to their code after the project demonstration. Instead, students should focus on writing a well-structured report which describes the motivation, related work, baseline design, design alternatives, testing strategy, and evaluation for their design project.

This a hybrid course with three credits of instruction within a formal classroom setting and one credit of project work outside the formal classroom. The course includes 2260 minutes of in-classroom instruction across the entire semester. The course will require approximately 6+ hours per week of supplemental work outside the classroom before spring break and 8+ hours per week of project work outside the classroom after spring break.

Lectures	19	×	75 min	=	1425 min	(2/week before spring break)
Discussion Sections	10	×	50 min	=	500 min	(1/week before spring break)
Midterm Exam	1	×	180 min	=	180 min	
Project Meetings	5	×	25 min	=	125 min	(1/week after spring break)
Project Demo	1	×	30 min	=	30 min	
In-Classrom Instruction					2260 min	
Project Work	5	×	480 min	=	2400 min	(8 hours/week after spring break)

7. Assignment and Exam Schedule

The current schedule is on the course website. All assignments are due at 11:59pm, except for project milestone documents which are due on Fridays at 11:59pm (see Section 9.C for late assignment policy). All assignments should be submitted electronically via the online Canvas assignment system, except for lab and project code which is submitted via GitHub. Project demonstrations will occur during the final week of classes. Changes to this schedule will be posted as announcements via Ed.

Thu	Feb	20	Lab 1 – ASIC Integer Multiplier
Thu	Mar	6	Preproposal Due
Thu	Mar	20	Lab 2 – ASIC Sorting Accelerator
Tue	Mar	25	Proposal Due
Thu	Mar	27	Midterm from 7:30–10:30pm in 219 Phillips Hall
Fri	Apr	18	Milestone 1 – Baseline Design and Testing Strategy Document
Fri	Apr	25	Milestone 2 – Alternative Design Document
Fri	May	2	Milestone 3 – Evaluation Document
Fri	May	9	Design Project Demonstrations
TBD			Design Project Report Due

8. Grading Scheme

This course will be adopting a philosophy of "grading for equity" where grading is exclusively used to assess mastery of the material covered in the course as opposed to rewarding effort and/or incentivizing specific behaviors. To this end, each part or criteria of every assignment is graded on a five-point scale without any curve according to the following rubric.

- **5 (Mastery):** Submitted work demonstrates no misunderstanding (there may be small mistakes which do not indicate a misunderstanding) or there may be a very small misunderstanding that is vastly outweighed by the demonstrated understanding. Student has mastered learning objectives; can independently apply course material in later courses and/or career.

- **4 (Accomplished):** Submitted work demonstrates more understanding than misunderstanding. Student has accomplished learning objectives; would probably need some additional learning/help to apply course material in later courses and/or career.
- **3 (Progressing):** Submitted work demonstrates more misunderstanding than understanding. Student is still progressing towards learning objectives; would need additional study and practice to apply course material in later courses and/or career.
- **2 (Beginning):** Submitted work is significantly lacking in some way. Student is just beginning towards learning objectives; would need significant additional study and practice before being able to apply course material in later courses and/or career.
- **1 (Minimal Understanding)**

A score of 5 corresponds to an A, 4 corresponds to a B, 3 corresponds to a C, and so on. A score of 5.25 is reserved for when the submitted work is perfect with absolutely no mistakes or is exceptional in some other way.

Total scores are a weighted average of the scores for each part or criteria. Parts or criteria are usually structured to assess a student's understanding according to four kinds of knowledge: basic recall of previously seen concepts, applying concepts in new situations, qualitatively and quantitatively evaluating design alternatives, and creatively implementing new designs; these are ordered in increasing sophistication and thus increasing weight. In almost all cases, scores are awarded for demonstrating understanding and not for effort. Detailed rubrics for all assignments are provided once the assignment has been graded to enable students to easily see how the score was awarded. For lab assignments, a detailed Lab Assignment Assessment Rubric will be available on Canvas.

The final grade is calculated using a weighted average of all assignments with the following distribution. Note that the design project as a whole is worth roughly half of the final grade. **If you are not willing to put a significant amount of work into this course after spring break, please do not take this course.**

Lab Assignment 1	10%	
Lab Assignment 2	15%	
Midterm	30%	
Design Project Milestones	10%	(evenly weighted)
Design Project Demonstration	10%	
Design Project Report	25%	

9. Policies

This section outlines various policies concerning usage of cellular phones and laptops in lecture, turning in assignments late, regrading assignments, collaboration, copyright, and accommodations for students with disabilities.

9.A Auditor and Listener Policy

Casual listeners that attend lecture but do not enroll as auditors are not allowed; you must enroll officially as an auditor. Auditors are allowed to enroll in the course as long as there is sufficient capacity in the lecture room. Auditors must attend most of the lectures. If you do not plan on attending the lectures, then please do not audit the course. Please note that students are not allowed

to audit the course and then take it for credit in a later year unless there is some kind of truly exceptional circumstance.

9.B Cellular Phones and Laptops in Lecture Policy

Students are prohibited from using cellular phones and laptops in lecture unless they receive explicit permission from the instructor. It is not practical to take notes with a laptop for this course. Students will need to write on the handouts and quickly sketch diagrams during lecture. The distraction caused by a few students using (or misusing) laptops during lecture far outweighs any benefit. Tablets are allowed as long as they are kept flat and used exclusively for note taking. If you feel that you have a strong case for using a laptop during lecture then please speak with the instructor.

9.C Late Assignment Policy

Lab assignment code must be submitted electronically via GitHub. Lab reports must be submitted electronically in PDF format via Cavnas. **No other formats will be accepted!** All assignments must be submitted by 11:59pm on the due date unless otherwise specified. No extensions will be granted except for serious illness or family emergency. The instructors must be notified of this serious illness or family emergency in advance if at all possible. You can continue to resubmit your files as many times as you would like up until the deadline, so please feel free to push your code to GitHub and upload to Canvas early and often. **If you attempt to submit an assignment even one minute past the deadline, then the assignment will be closed and you will not be able to submit.**

9.D Regrade Policy

Addition errors in the total score are always applicable for regrades. Regrades concerning the actual solution should be rare and are only permitted when there is a significant error. Please only make regrade requests when the case is strong and a significant number of points are at stake. Regrade requests should be submitted online via a private post on Ed within one week of when an assignment is returned to the student. You must provide a justification for the regrade request.

9.E Collaboration and Artificial Intelligence Policy

The work you submit for the lab assignments is expected to accurately demonstrate your understanding of the material. The use of a computer in no way modifies the standards of academic integrity expected under the University Code. You are encouraged to discuss information and concepts covered in lecture and relevant to the lab assignments with other students. You can give "consulting" help to or receive "consulting" help from other students about the lab assignments. Students can also freely discuss basic computing skills or the course infrastructure. **However, this permissible cooperation should never involve one student (or group) having possession of or observing in detail a copy of all or part of work done by someone else, in the form of an email, an email attachment file, a flash drive, or on a computer screen.** Students are not allowed to seek consulting help from online forums outside of Cornell University. **If a student receives consulting help from anyone outside of the course staff, then the student must acknowledge this help on the submitted assignment.**

As an exception to the outside consulting policy described above, **students *are* allowed to use artificial intelligence (AI) systems (e.g., OpenAI ChatGPT, Anthropic Claude, Microsoft Copilot) in absolutely any way they want in the course as long as all submitted material still represents**

the students' understanding. Students are free to use AI to explain lecture concepts, create practice problems, explain problem solutions, write Verilog code, debug Verilog code, analyze Verilog compile-time errors, brainstorm test cases, and/or edit lab reports. **The only condition is that students *must* acknowledge how they used AI in any submitted work.** Students must include an AI acknowledgment as a comment at the top of any source code for which AI was used in any way. Students must include an AI acknowledgment at the end of lab report for which AI was used in any way. The AI acknowledgment should clearly specify which AI was used and how it was used. **Using AI without acknowledgment will be considered and academic integrity violation.** Students are responsible for all of their submitted work and that work must represent their understanding even with an AI acknowledgment. The instructor reserves the right to use a short oral inquiry with students to verify that they understand anything they submit as their own work.

During examinations, you must do your own work. Talking or discussion is not permitted during the examinations, nor may you compare papers, copy from others, or collaborate in any way. Students must not discuss an exam's contents with other students who have not taken the exam. If prior to taking it, you are inadvertently exposed to material in an exam (by whatever means) you must immediately inform an instructor.

Should a violation of the code of academic integrity occur, then a primary hearing will be held. See `https://theuniversityfaculty.cornell.edu/academic-integrity` for more information about academic integrity proceedings.

9.F Copyright Policy

All course materials produced by the course instructor (including all handouts, tutorials, homeworks, quizzes, exams, videos, scripts, and code) are copyright of the course instructor unless otherwise noted. Download and use of these materials are permitted for individual educational non-commercial purposes only. Redistribution either in part or in whole via both commercial (e.g., Course Hero) or non-commercial (e.g., public website) means requires written permission of the copyright holder.

9.G Accommodations for Students with Disabilities

In compliance with the Cornell University policy and equal access laws, the instructor is available to discuss appropriate academic accommodations that may be required for students with disabilities. Students must register with Student Disability Services (SDS) within the first three weeks of the semester to verify their eligibility for appropriate accommodations. If students register with SDS after the first three weeks of the semester, the instructor may or may not be able to make the appropriate accomodation.

For students with testing accommodations, this course is participating in the SDS Alternative Testing Program. If you have an approved testing accommodation, you must request it for this course and complete an Exam Request Form for the midterm exam via the SDS student portal by early February. Failure to do so may result in the inability to use your accommodation.

Coordination of make-up exams (i.e., for students who have been granted prior permission by the instructor to take the exam on a day other than the scheduled date of the main exam) will be handled by the instructor. The SDS Alternative Testing Program will not be involved in the logistics for any make-up exams. If you miss your scheduled accommodated exam, you should notify the instructor, not SDS.

10. Online and Computing Resources

We will be making use of a variety of online websites and computing resources.

- **Public Course Website –** `http://www.csl.cornell.edu/courses/ece6745` This is the main public course website which has the course details, updated schedule, lecture slide handouts, tutorials, and most handouts. We intend for all course content to always be available on Canvas. The public course website is just for public access to some of this content.
- **Canvas Course Site –** We will be using Canvas to manage course content, assignment submission, and grade distribution.
- **Ed –** We will be using Ed for all announcements and discussion on course content, lab assignments, and the projects. The course staff is notified whenever anyone posts on the forum and will respond quickly. Using the forum allows other students to contribute to the discussion and to see the answers. Use common sense when posting questions such that you do not reveal solutions. Please prefer posting to Ed Discussions as opposed to directly emailing the course staff unless you need to discuss a personal issue.
- **ECE Computing Resources –** The ECE department has a cluster of Linux-based workstations and servers which we will be using for the programming assignments. You can access the ECE computing resources remotely using various methods. More information will be made available to students shortly.
- **GitHub –** GitHub is an online Git repository hosting service. We will be using the commercial GitHub service to distribute code and as a mechanism for student collaboration on the lab assignments. Students will also use GitHub for submitting the code for their lab assignments. Students are expected to become familiar with the Git version control system. Note that we are not using the Cornell hosted version of GitHub as in some other courses; we are using `github.com`.