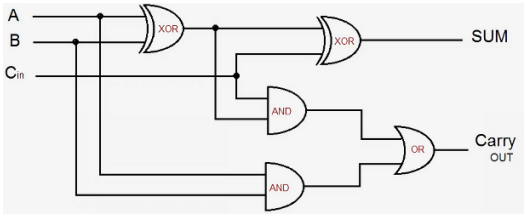
CS 355-1: Advanced Computer Architecture

```
Define Full_Adder C_In A B | CarryOut Sum;
  Xor aa A B x;
  Xor ab x C_In Sum;
  And bb A B y;
  And cb C_In x z;
  Or bc-cc y z CarryOut;
Endef;
```



Course Description and Objectives

This course is an elective course in the Computer Science curriculum. The pre-requisite for CS355 is CS255 (and CS253) where you have learned the basics of the operation of the **computer system**. The **goal** of CS355 is to show you how the computer system operates at the **digital circuit level**.

In CS355, you will first learn how the elementary digital circuit operates: and-circuit, or-circuit, not-circuit, etc. We will then construct switching circuits, arithmetic circuits and memory circuits using the elementary digital circuits. Next we will design a simple ALU circuit and a simple memory circuit using the switching and arithmetic circuits and complete the first half of the course by studying (and building) a simple (micro-programmed) CPU, the system bus that connects the CPU to I/O devices and how the CPU communicates with I/O devices. In the second half of the course, we will study advanced designs in CPU architecture by studying a pipelined CPU design, cache memories and parallel processors. We will also learn how to program an SIMD (Single Instruction Multiple Data) parallel computer (an NVidia GPU) using the CUDA programming language.

Syllabus Contents

Your Professor

Dr. Shun Yan Cheung, PhD
Office: MSC Room W428, Phone: 404-727-3828
Email: cheung@emory.edu (preferred)
Office hours: MW 4-5, TuTh 1-2 and by appointment
Zoom office URL: see Canvas

Course resources:

Class website:	http://www.mathcs.emory.edu/~cheung/Courses/355/index.html
Text book: (online)	http://www.mathcs.emory.edu/~cheung/Courses/355/Syllabus/syl.html
Online Zoom lecture:	See Canvas for the URL
Online Zoom office:	See Canvas for the URL

Class Policies

- Homeworks/projects and their deadlines are posted on the class website and they must be turned in prior to the dead line unless you have requested an extension.
- You can request an extension for homework/project for **upto 3 times**. Extensions will always be granted and you will get at least 3 extra days when you receive an extension. You cannot request two extensions for the **same** homework/project.
- Requests for homework/project extension **must** be made ***before*** the assignment is due.
- "Free" extensions are allowed for illnesses, accidents or personal matters that require you to leave for home. “Free” extensions are not counted towards the maximum of 3 allowed extensions. In order to receive a “free” extension, I will need documentation, like a doctor's note. A "free" extension must requested **ASAP**.
- OAS students can make unlimited extension and receive extra time for exams when OAS status approved
- Homeworks/projects must be turned in according to the instructions given in each homework/project assignment write-up. I.e.: after completing the homework/project, you must follow the instructions in the homework/project write-up and turn the work in.
- I have online office hours (at the Zoom link given on Canvas). Office hours are not used as lecture time where I reteach a topic that I have covered in some lecture. It is time for you to **ask specific question(s)** about lecture materials that you have question(s) on or explore the material in more depth. Outside office hours, you can ask me questions through email – I will usually respond the same day.
- Abide to the Emory Honor Code: http://catalog.college.emory.edu/academic/policies-regulations/honor-code.html

Taking this course

Unlike all other CS course, CS355 contains very little programming. You will be using a circuit simulation system to connect digital circuits together and build a (working) CPU ! Ever wondered how a computer can add, multiply, store data, execute instructions, etc.? After taking CS355, you will understand exactly how a computer operates internally. You will learn why computers do not *actually* compute, but only perform logical operations (such as AND, OR, NOT, etc.).

Grading:

Weight of assignments:

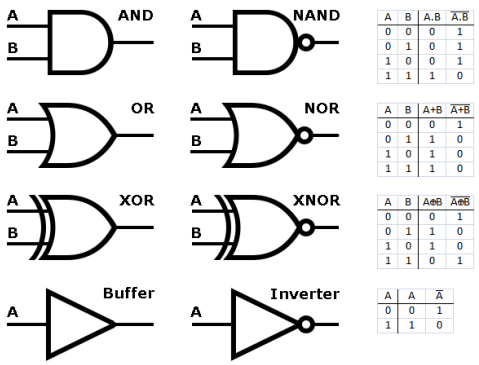
- Homeworks and projects: 40%
- Midterm exam: 25%
- Final exam: 35%

Missing exams:

If you miss an (midterm or final) exam, you will **only** be allowed to do a "make up exam" with the permission the **Office of the Undergraduate Education** (i.e., the **dean**)

Grade scale:

Letter	Percentage
A	93.3 - 100
A-	90.0 – 93.2
B+	86.6 – 89.9
B	83.3 – 86.5
B-	80.0 – 83.2
C+	76.6 – 79.9
C	73.3 – 76.5
C-	70.0 – 73.2
D+	65.0 – 69.9
D	60.0 – 64.9
F	< 60.0



A	B	A.B	$\overline{A.B}$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

A	B	A+B	$\overline{A+B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

A	B	A⊕B	$\overline{A \oplus B}$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1

A	A	$\overline{A}$
0	0	1
1	1	0

Lecture format:

How:	Synchronous online lectures using Zoom. I will record each lecture and post the recording on Canvas – you can view the recording asynchronously later when available.
Zoom URL:	See Canvas for the URL
Netiquette of online Zoom lectures:	**Mute** your microphone during the synchronous lecture, **except** when asking question. When you want to ask a question, unmute your microphone and then say: "Question" to interrupt the lecture. I will stop and let you ask your question. You must mute your microphone when you finish asking your question to minimize background noise.

Class Schedule (tentative)

Week	Lecture dates	Topics	What to read	To do assignment
1-2	1/12, 1/19	Logic elements and Boolean Algebra	Chapter 1	Project 1 (designing a combinatorial circuit)
3	1/24, 1/26	Switching circuits and Arithmetic circuits	Chapters 2-3	
4	1/31, 2/2	Sequential circuits and Finite State Machines	Chapters 4-5	Project 2 (multiply circuit) and Project 3 (ALU and shifter)
5	2/7, 2/9	Bi-directional transfer and Memory (RAM)	Chapters 6-7	Project 4 (Clock and MPC)
6	2/14, 2/16	CPU architecture (data path)	Chapter 8	Project 5 (Micro-instructions)
7	2/21, 3/23	The system bus	Chapter 9	Project 6 (CPU design)
8	2/28, 3/2	I/O communication	Chapter 10	
9	3/14, 3/16	Pipeline design - intro	Chapter 11	
10	3/21, 3/23	Pipeline design – basic pipelined CPU	Chapter 11	

11	3/28, 3/30	Pipelined design – data forwarding hardware and control hazard	Chapter 11	Homework 7 (Pipelined CPU)
12	4/4, 4/6	Cache memories	Chapters 12	
13	4/11, 4/13	Parallel processors and intro to CUDA (SIMS) programming	Chapters 13	Project 8 (CUDA programming)
14-15	4/18, 4/20, 4/25	Advanced CUDA programming	Chapter 14