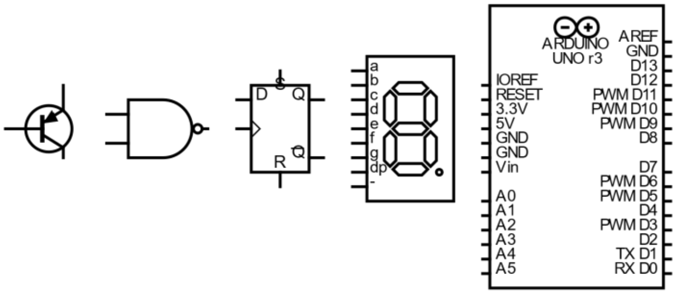
Princ of Digital Electronics (ENR 325 / ENR 325L)



Fall, 2025
Lectures (ENR 325): MWF/1:00 pm to 1:50 pm — CHS 148
Labs (ENR 325L): T/8:00 am to 10:50 am — CHS 148
Professor: Xiang Li
Contact Information: xili@coe.edu
Student Hours: walk-in (WF 9:00 am – 12:00 pm) or by individual appointments
Department Chair: Firdevs Duru, fduru@coe.edu

Course Description:
Welcome to the world of digital electronics! We are going to re-live the days when brilliant physicist and mathematicians built something almost magical; and brought us the last two industrial revolution. In fact, they were so successful, that they essentially reshaped their own discipline, and gave birth to new fields and industries (such as the Silicon Valley). By study & practice their way of thoughts and application, I hope that this course can be more about circuits and I/Os. but about having hand-on experience of the peak of the American innovation in the last century.

There are many topics to cover so we often don't have time to do in-depth studies, but luckily, we have labs and open-ended final projects so you can showcase your newly required knowledge and skill sets. I cannot put it better than Richard Hamming:

> *"Education is what, when, and why to do things, Training is how to do it."*
> —Art of doing science and engineering: Learning to learn. Chapter 1. CRC Press, 1997.

Measurable Learning Objectives:
For lecture outcomes, you will be able to:

- Understand the digital abstraction and asses if a system obeys the static discipline.
- Identify and calculate Shannon's entropy in different applications.
- Build a Huffman tree.
- Understand and apply Hamming coding system with hand calculation as well as coding.
- Perform binary arithmetic operations (e.g. conversion, addition and subtraction) by hand.
- Implementing advanced binary arithmetic operations (e.g. more efficient addition or multiply) with coding and bare metal hardware.

- Associate logic gates with truth table, Boolean operations, De Morgan's laws, K map, and sum of products expressions.
- Construct basic digital components latches, flip-flops, adder, and ALU in software, if not in bare metal hardware.
- Implement MUX and bus in circuits design for Hamming codes.
- Understand and derive CMOS based logic gates (inverter, NAND, XOR gates).
- Design photolithography masks for three terminal metal contacts and development fabrication recipe based on datasheets.
- Design basic logic gates realizing Hamming codes in IC with the help of digital design and design rules.
- Write introductory programs for FPGAs and microcontrollers (MCUs) for applications such as encoding/decoding and I/O control.

For lab outcomes, you will be able to:

- Safely and correctly use lab equipment: oscilloscopes, function generators, power supplies, probes, breadboards, and other proper tooling.
- Implement Hamming coding in breadboard.
- Work with data transmitting interfaces such as SPI.
- Have hands on experience with PCB design.
- Have hands on experience with pin soldering and surface mount soldering.
- Build an organic transistor and generate its characteristic curves.
- Build an IoT application with either breadboard, PCB design, or programmable IC.

Required Texts and/or Course Materials:

- Lecture slides and lab resources will be posted and maintained on Moodle and GitHub.
- There is no traditional textbook; curated online references and readings will be provided throughout the term.

Course Policies:

Official catalog description: ENR-325/325L Principles of Digital Electronics and Laboratory Introduces digital devices used in modern electronics. Topics covered include logic gates, flip-flops, timers, counters, multiplexing, analog-to-digital and digital-to-analog devices. Emphasis is on constructing, analyzing, verifying, and troubleshooting digital circuits using appropriate techniques and test equipment. Prerequisite: Principles of Analog Electronics and Laboratory (PHY-245/245L).

The course content is split up into five major modules;

- Digital and information system
- Logic circuits and interface design
- CMOS, IC and digital design
- FPGA, MCU and IDE development
- IoT and other smart things

Schedule of major due dates

For lectures, assignments will be issued in lectures and collected a week later.

For labs, there will be four mini projects, each last three week long. Final lab reports will be collected at the end of the third week. The draft version of lab reports can be submitted anytime for feedbacks.

Here is the list of four mini projects:

- Bare-metal logic circuits
- Bare-metal coding system
- System application with I/O
- System application with biomatrix

Almost none of the assignments have point values, the ones do are given 0 (Retry) or 1 (exemplary/success) points. There is no partial credit and no averaging.
The work you turn in will be evaluated against quality standards that will be made clear on each assignment. If your work meets the standard, you will receive full credit for it. You will also get feedback, a chance to revise and update your grading.

The goal is to create a learning loop: try something, make mistakes, reflect, revise, and try again—without penalty for mistakes, as long as you demonstrate growth.
This process is mirror how evaluation of work happens post college. In your futural jobs, your manager (or PI in academia) will provide you feedback, and you will then demonstrate growth. A good project team will always move forward by such iteration.

The individual types of assignments are listed as follows:

Assignment	**Basis for grading**	**What will be recorded**
Foundation skills	Overall correctness	Success/Retry
Applications	Completeness and overall correctness	Exemplary/Success/Retry
Lab reports	Completeness, overall correctness, writing, and presentation	Exemplary/Success/Retry/or Incomplete
Advanced skills	Completeness and efforts	Exemplary/Pass/No pass
Presentation	Writing, documentation, and presentation	Pass/No pass

Method for determining final grade

Your grade is assigned with the table listed below. Each row indicates the minimum number of successful results needed to satisfy the requirement for that grade. For example, you will be required to earn 13 or more success mark on foundation skills to get grade of B.

Grade	Foundation skills (out of 16)	Application (out of 10)	Lab reports (out of 4)	Advanced skills (out of 5)	Presentation (out of 5)
A	15	6	4	3	3
B	13	5	3	2	3
C	11	4	2	1	2
D	5	2	1	1	1

A grade of “F” is assigned if not all of the requirement for a “D” are met.
Plus grades: A "plus" is given on a letter grade if you satisfy all the requirements plus one of the following:

- More than half of the recorded grades are exemplary, when applicable.

- Complete two categories in the grade table for the next grade level up, no including presentation.

Minus grades: no “minus” will be given on a letter grade.

Participation expectations, if applicable

Participate actively in each class will get the most of the course and avoid spending extra time self-teaching.

Attendance is tracked but not graded. I will follow up with student with excessive absences, but not direct penalty will be incurred. You do not need to seek permission to miss a class.

Late work policy, if applicable

You will start the semester with 5 tokens. You will spend 1 token to:

- Extend the deadline on any submissions by 36 hours
- Retry a second time (i.e., submission for a third time).

Deadline extension must be request before the original deadline.

Acceptable methods for submitting work

Microsoft PowerPoints/google slides will be the preferred submission format. Hand writing, hand sketch, or multimedia format will be acceptable only if it’s comprehensive to me, but won’t earn an “Exemplary” grade. “Slide deck skill” will be essential for your future career, now it’s a good time to practice!

Course management system /website, if applicable

Lecture slides and lab resources will be posted and maintained on Moodle and GitHub.

Information about Final Exam and course meetings through the end of the term

You will have chance doing presentation, and a large component of your final project will be the presentation, i.e. not just the method, data, result, but a good story telling.

Any additional information or class policies on ChatGPT/AI/generative technology

General rule-of-thumb: AI tools are roller skates, not crutches.

AI tools (like ChatGPT or Copilot) may be used as long as they support your learning and do not replace your independent thought or effort.

All AI-generated content must be clearly cited and labeled. You must include the prompt(s) used as an appendix to your assignment or code.

All AI-generated code must be clearly labeled, here’s an example below:

```
#################### Begin AI generated code ########################

print("Hello human")
print("Welcome our AI overlord")
#################### End AI generated code ##########################
```