ECE Course Optimization Guide

Carnegie Mellon University

Updated as of Fall 2023

Electrical & Computer ENGINEERING

Table of Contents

About

If you are reading this document, this is hopefully because you:

- Want to know what courses are or aren't commonly taken for a concentration
- Are unsure of what courses to prioritize given your interests, and which courses have conflicts with other options
- Want more personal information about a course than is provided in its description or syllabus
- Want to learn about a course, but may not know anyone who has taken it

B.S. in ECE Requirements

ECE Academic Guide: Outline of all ECE requirements to graduate

CIT General Education Requirements:

Computing @ Carnegie Mellon

- Not an important course, but try to finish in any summer

CIT General Education

- Requirements
- Aim to spread these courses out over all your years at CMU because they take your mind off difficult classes
- Take classes on topics you enjoy, or else these classes will become chores

21-120 & 21-122: Calculus I & Calculus II

- Aim to take these freshman fall and spring, respectively
- Prerequisite for all relevant math courses

Two introductory engineering courses

- Take 18-100 (Introduction to ECE) and any other introductory course

33-141 & 33-142: Physics I & Physics II

- Physics II is relevant if you are interested in device sciences and circuits
- **Note**: Neither of these courses are a prerequisite for any ECE course
  - Thus, some people push these courses to later semesters due to their high workload, especially for freshmen, and general irrelevance
  - Physics II is known to be quite difficult
  - Some students decide to take these courses at the University of Pittsburgh for credit

15-112: Fundamentals of Programming and Computer Science

- An important class to take because the chain of prerequisites is: 15-112 → 15-122 → 15-213/18-213 → ECE Capstone
  - These are all required courses, so plan accordingly!
- If you are interested in the software track, this class is a **MUST**, assuming you don't have credit

🏠

General Technical Requirements

Mathematical Foundations of Electrical Engineering

- Unless you must save courses, it is recommended to **NOT** take 18-202 (Mathematical Foundations of Electrical Engineering) to satisfy this requirement
    - The general consensus is this course is taught poorly and provides a poor overview of the involved mathematical concepts
- Instead, most ECEs take 21-241 (Matrices and Linear Transformations) and 21-259 (Calculus in Three Dimensions)
    - If you are interested in the signals and systems track, 21-260 (Differential Equations) will be important and you should replace 21-259 with it

Two Math/Science Electives

- Take whatever you are interested in! Many statistics, math, biology, chemistry, and physics courses fit into this category

Probability

- If you enjoy math and want to obtain a deep understanding of probability, take 21-325 (Probability)
    - WARNING: This is a much more difficult course than 36-219 or 36-225 that requires a strong calculus and discrete mathematics background
- Otherwise, take either 36-225 (Introduction to Probability Theory) or 36-219 (Probability Theory and Random Processes)
    - 36-225 is known to be taught better

15-122: Principles of Imperative Computation

- A **priority** course for anyone interested in the software track due to it being a prerequisite for 15/18-213
- Very time-consuming (On average, students report spending 15.7 hours per week)
- If you enjoyed this course, consider taking 18-213, or doing the Software Systems concentration

21-127/21-128: Concepts of Mathematics

- Though this is a corequisite for 15-122, it is not entirely necessary for any track
    - Most ECEs do not delay this course, but this is simply preference
- Very time-consuming, especially for those who have never touched mathematical proofs before

🏠

Core Requirements

18-100: Introduction to Electrical and Computer Engineering

- Take in freshman year, ideally in the fall. Taking this in the fall will open the door for taking a core class in freshman spring if you are set on a track
    - Since freshmen are not officially ECE students yet, it may be difficult to register for a core, especially for more popular courses like 18-240
- If you enjoyed making analog circuits and learning about op-amps, consider taking 18-220, or doing the Circuits or Device Sciences concentration
- If you enjoyed adder lab, consider taking 18-240, or doing the Hardware Systems concentration
- If you enjoyed doing Radio Lab and learning about DSP, consider taking 18-290, or doing the Signals and Systems concentration

18-200: ECE Sophomore Seminar

- Take in sophomore fall
    - Useful for understanding what your plan and goals are as an ECE student

18-213, 18-220, 18-240, 18-290 *→ explained in the* ***Core Courses*** *section*

- Prioritize taking the cores most related to your desired concentration first
    - i.e. 18-240 and 18-213 for Hardware Systems
- If you are unsure of the track you want to follow, aim to finish all four cores by the end of sophomore year
    - The recommended progression is 18-213 + 18-220 in sophomore fall and 18-240 + 18-290 in sophomore spring because this provides similar workload and lab time across both semesters

Area Requirements

- Take **two** courses in the track if your concentration
- Take **one** interesting course in a track outside your concentration
    - Not recommended to put off until later because you should explore your options

Coverage Requirement

- Take a course you are interested in!
- Any ECE course with course code 18-300 or higher counts
- Many CS courses, along with some machine learning, software engineering, and robotics courses also count

Capstone Design Requirement

- Most common course taken for this requirement is **18-500: ECE Design Experience**
    - Course is focused around designing a product all the way from the brainstorming process through it's finalized design documents and full implementation
    - Course is focused around your "Capstone Project", which is done in teams of 2-3. If you cannot find anyone, there is allotted time for choosing partners during the beginning of the class.
    - Choose a project which you can have polished within a semester. This class has an emphasis on having a solid final product.

Free Electives

- All courses (besides Physical Education and StuCo courses once you have taken 9 units of them) count for this, so take whatever you want!

Core Courses

All of these cores are prerequisites for any future classes in their respective tracks, so choose which cores to take first accordingly.

Additionally, anyone who graduates with an ECE degree from any university should be expected to know the content in the core classes. Take these classes seriously!

Software Systems Track

18-213: Introduction to Computer Systems

- **Sem. Any Units: 12 FCE: 16.4 hrs/week**
- **Prerequisites: 15-122**
- **Notes:**
  - If you liked 15-112 or 15-122, you will likely enjoy this course
  - This course is a prerequisite for essentially every other Computer Systems course. Take it early if you plan on doing that concentration
  - Commonly overlaps with 18-240
  - Can be taken over the summer to unlock more classes earlier on

Device Sciences and Nanofabrication & Circuits Tracks

18-220: Electronic Devices and Analog Circuits

- **Sem. Any Units: 12 FCE: 10.7 hrs/week**
- **Prerequisites: 18-100**
- **Notes:**
  - If you enjoyed analog circuits and learning about op-amps in 18-100, you will likely enjoy this course
  - Many ECE students take this as one of their first cores because of its low difficulty
  - 18-220 lets you pick lab partners, so take it with a friend you work well with
  - This course primarily unlocks classes in the “Device Sciences and Nanofabrication” and “Circuits” tracks.

Hardware Systems Track

18-240: Structure and Design of Digital Systems

- **Sem. Any Units: 12 FCE: 17.1 hrs/week**

- **Prerequisites: 18-100**
- **Notes:**
    - If you enjoyed adder lab in 18-100, you will likely enjoy this course
    - This course unlocks courses in the "Hardware Systems" track
        - Many of the courses this unlocks are fall-only. Plan accordingly
    - Commonly overlaps with 18-213

Signals and Systems Track

18-290: Signals and Systems

- **Sem. Any Units: 12 FCE: 11.4 hrs/week**
- **Prerequisites: 18-100**
- **Notes:**
    - If you enjoyed learning about DSP and radio lab, you will likely enjoy this course
    - This unlocks the "Signals and Systems" tracks
    - WARNING: This class often shares a time conflict with 18-447 (Computer Architecture). 18-290 is required to take capstone, which may leave Hardware-concentration students in a tough situation

🏠

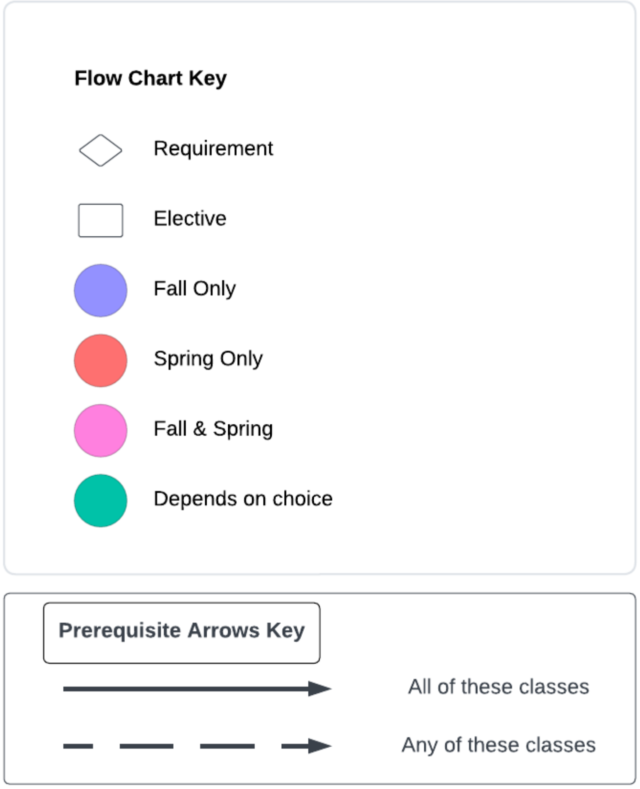
Tracks

About

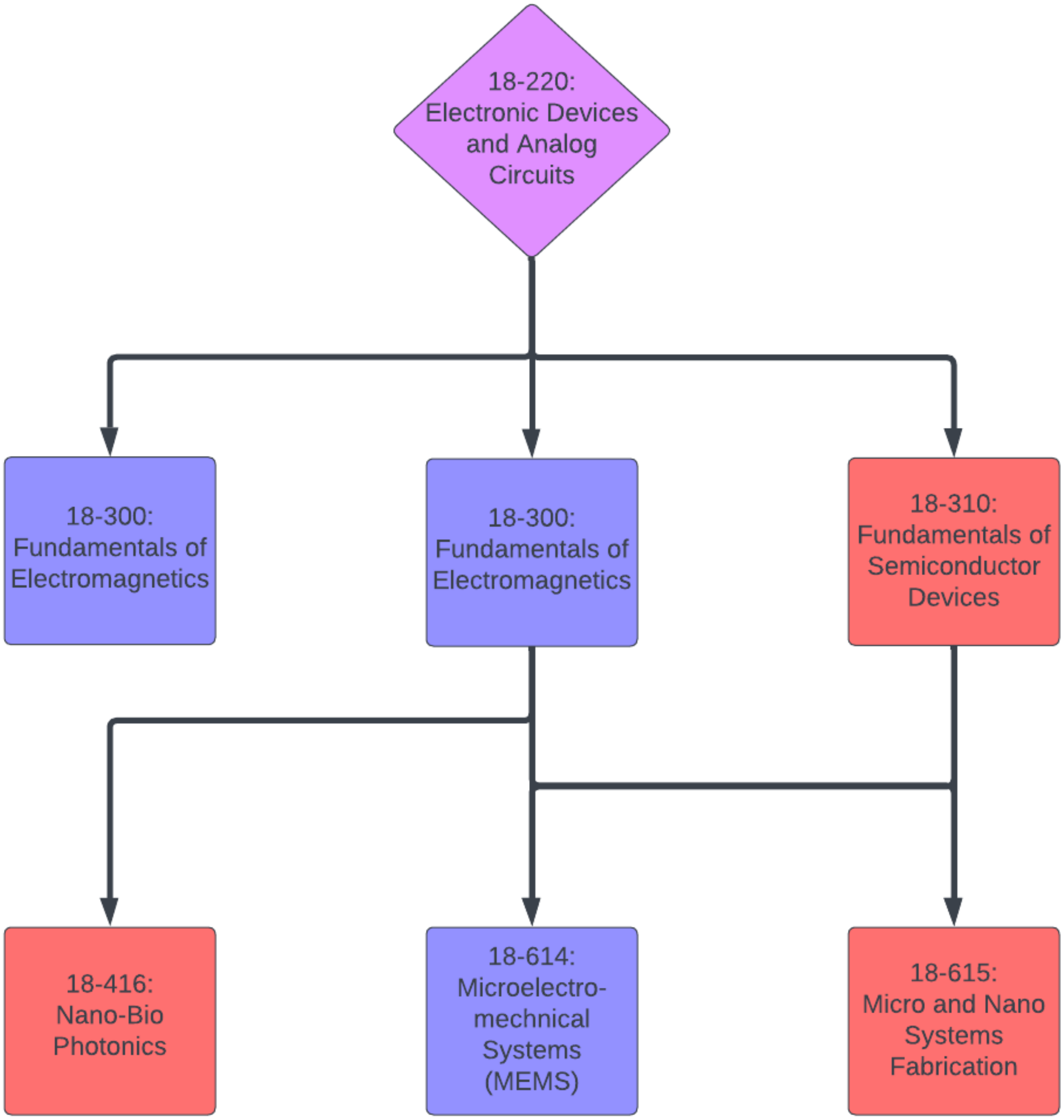
This section provides a course flow chart for every track as a convenient way for users to understand their general course progression. More information on many of these area courses can be found underneath each flow chart.

Key

Follow this flow chart key to understand how we organized each chart. For clarification, any solid arrow leading from a course means it is a necessary prerequisite for the course it leads to. Any dotted arrow leading from a course means it is one of a few possible prerequisites; you can take a different class to satisfy the prerequisite.



Device Sciences and Nanofabrication


18-220: Electronic Devices and Analog Circuits
18-300: Fundamentals of Electromagnetics
18-300: Fundamentals of Electromagnetics
18-310: Fundamentals of Semiconductor Devices
18-416: Nano-Bio Photonics
18-614: Microelectro-mechnical Systems (MEMS)
18-615: Micro and Nano Systems Fabrication

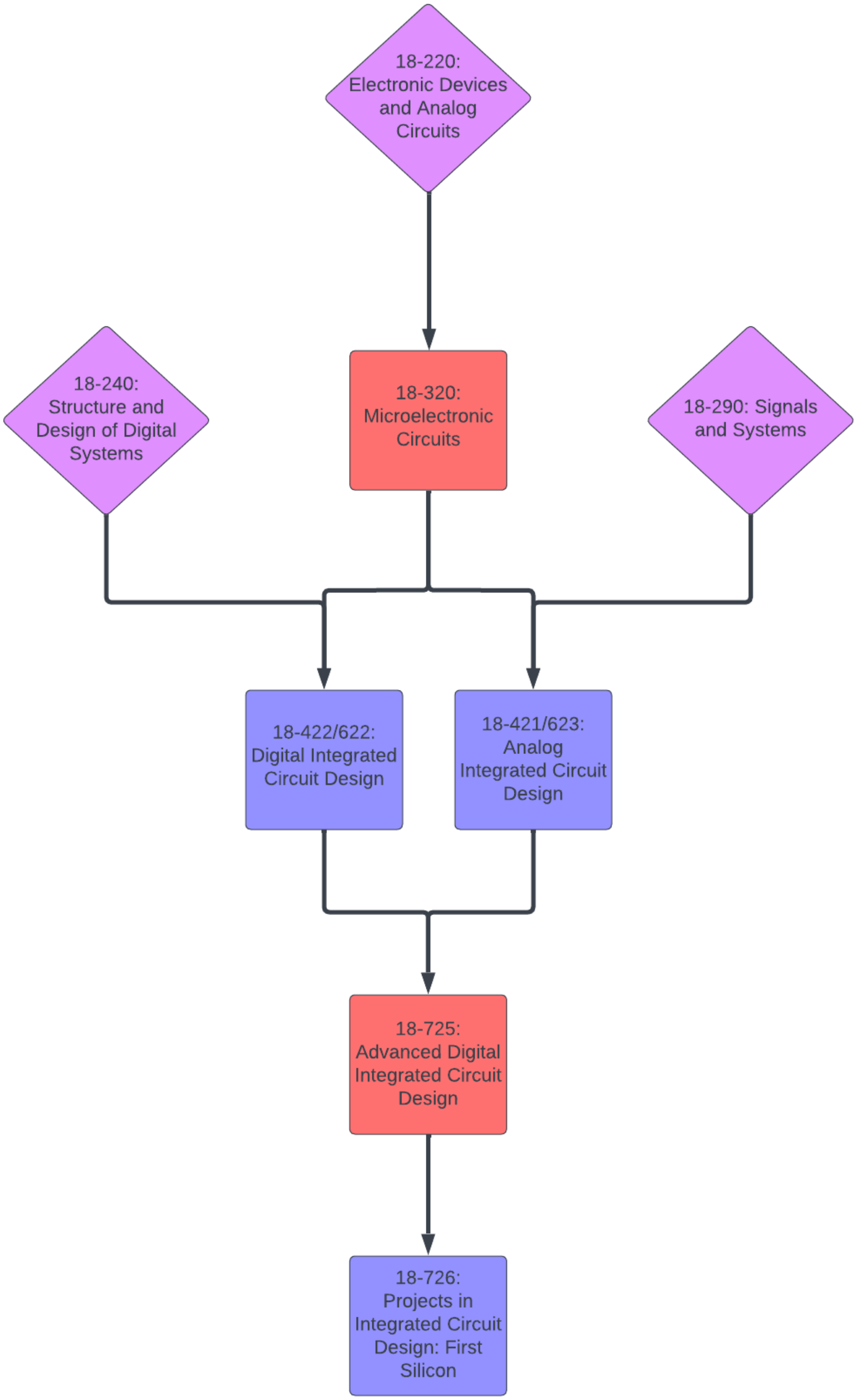
18-300: Fundamentals of Electromagnetics

- **Sem. Fall Units: 12 FCE: 9.0 hrs/week**
- **Prerequisites: 18-220**
- **Notes:**
    - This course mainly covers electromagnetics from the physics perspective. You will likely enjoy this class if you liked Physics 2.
    - Physics 2 is not a prerequisite, but it will help with this class.
    - Light weekly time requirement, less than or equal to 18-220.
    - Lectures are arguably more important than for other classes
    - Tests are of reasonable difficulty

18-310: Fundamentals of Semiconductor Devices

- **Sem. Spring Units: 12 FCE: 8.6 hrs/week**
- **Prerequisites: 18-220**
- **Notes**
    - Not much work if you do the bare minimum, but much to be learned if you do not
    - This class dives deep into the math, physics, and derivations involved with creating electrical components and circuits.
    - Class difficulty is similar to 18-220. There are homeworks and exams, but no labs.
    - Try to double check your homework with a partner, there are often many error-prone calculations involved

Circuits


18-220: Electronic Devices and Analog Circuits
18-240: Structure and Design of Digital Systems
18-320: Microelectronic Circuits
18-290: Signals and Systems
18-422/622: Digital Integrated Circuit Design
18-421/623: Analog Integrated Circuit Design
18-725: Advanced Digital Integrated Circuit Design
18-726: Projects in Integrated Circuit Design: First Silicon

18-320: Microelectronic Circuits

- **Sem. Spring   Units: 12   FCE: 11.6 hrs/week**
- **Prerequisites: 18-220**
- **Notes**
    - Conceptually more difficult than 18-220, but with a lower time commitment
    - Homeworks are easy, and labs occupy a large portion of the class's workload
    - If you take the class, make sure you understand the ins and outs of transistor operation and design! This is very important for exams, and for future classes.

18-422/622: Digital Integrated Circuit Design

- **Sem. Fall   Units: 12   FCE: 27 hrs/week**
- **Prerequisites: 18-240, 18-320**
- **Notes**
    - In 18-220, you create circuits with abstract diagrams. In this class, you learn how to arrange your circuit in a way where it can be manufactured as a chip.
    - 18-320 is sometimes waived for students, but is very valuable, as it teaches you how to use Cadence.
    - Heavy use of Cadence, applicable to the industry
    - There is a very steep ramp-up in difficulty partway through the course. Start the SRAM project early!

18-421/623: Analog Integrated Circuit Design

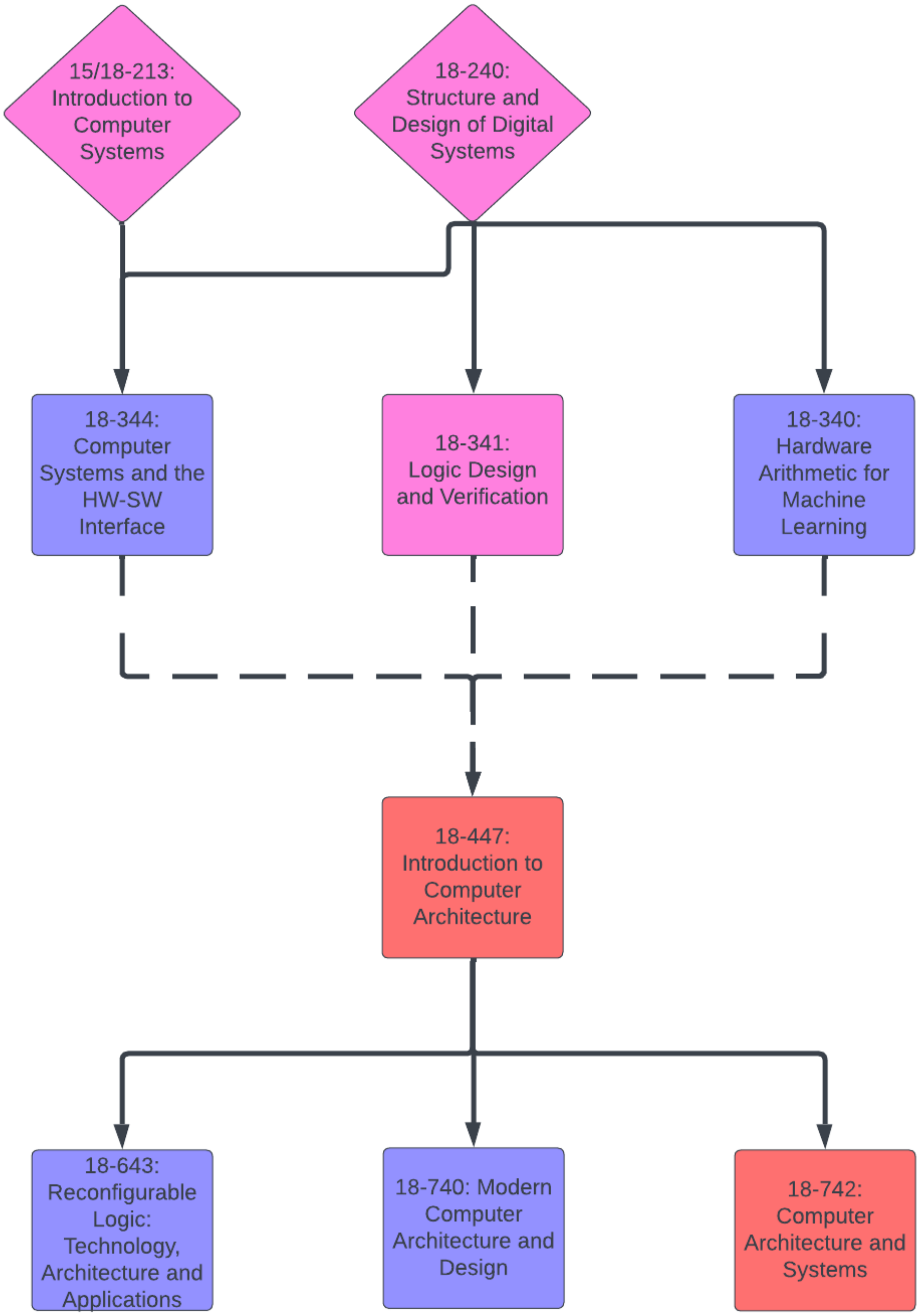
- **Sem. Fall   Units: 12   FCE: 15 hrs/week**
- **Prerequisites: 18-290, 18-320**
- **Notes:**
    - This class's focus is similar to that of 18-422, but it has a focus on analog circuits, as opposed to digital ones.
    - A much "friendlier" class than 422. Lighter workload, lots of extra credit opportunities, professor emphasizes "What to Expect" on midterm.
    - 2 design projects, 5-6 light homeworks
    - Workload is similar to 18-341
    - 18-320 is sometimes waived for students, but is very valuable, as it teaches you how to use Cadence – very applicable to the industry
    - Students can easily "upgrade" this class to the graduate version to count towards IMB

18-725/6: Advanced Digital Integrated Circuit Design and Projects in Integrated Circuit Design: First Silicon

- **Sem. Spring   Units: 12   FCE: 21 hrs/week**
- **Prerequisites: 18-622 or 18-422 or 18-623**
- **Notes**
    - Overview: Student teams create a chip of their choosing from scratch and test it
        - Rare to find anything else like it, dies are extremely expensive, will likely not be able to design a chip top-down at any other school
    - Can count for capstone credit

🏠

Hardware Systems


15/18-213: Introduction to Computer Systems
18-240: Structure and Design of Digital Systems
18-344: Computer Systems and the HW-SW Interface
18-341: Logic Design and Verification
18-340: Hardware Arithmetic for Machine Learning
18-447: Introduction to Computer Architecture
18-643: Reconfigurable Logic: Technology, Architecture and Applications
18-740: Modern Computer Architecture and Design
18-742: Computer Architecture and Systems

18-341: Logic Design and Verification

- **Sem. Any   Units: 12   FCE: 15.4 hrs/week**
- **Prerequisites: 18-240**
- **Notes**
  - If you are looking for a career in hardware design or verification, take this early. The content may not seem glamorous but it is great for interviews.
  - The workload is similar to 18-240, but Bill Nace (who nearly exclusively teaches the course) is more lenient
  - No exams.
  - Be careful, homework is graded harshly, and labs can be a lot of work - Both comprise your entire grade, so make sure you do them well.

18-344: Computer Systems and the Hardware-Software Interface

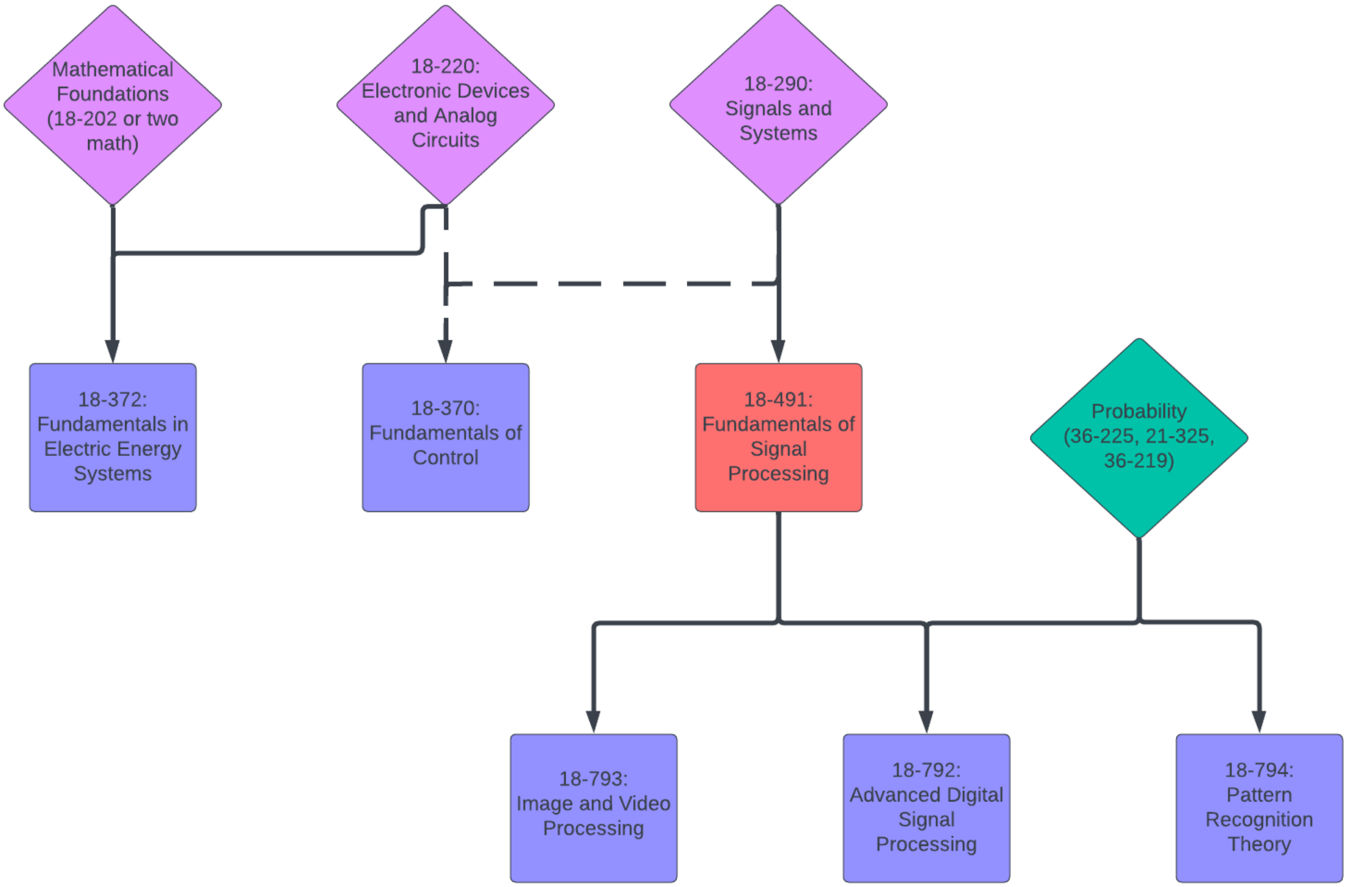
- **Sem. Fall   Units: 12   FCE: 12.2 hrs/week**
- **Prerequisites: 18-213, 18-240**
- **Notes**
  - Content includes computer architecture from a theoretical perspective and measuring how changes in hardware affects the speed of software - such as cache size or VM configurations
  - Teaches many concepts that are useful for 18-447
  - Labs take up much of the work
    - Open-ended analysis of results (up to ~10 pages)
  - Exams are consistently poorly written
  - Note: For students in the Hardware Systems concentration, this counts towards the Software Systems area requirement

18-447: Introduction to Computer Architecture

- **Sem. Spring   Units: 12   FCE: 17 hrs/week**
- **Prerequisites: 18-240, (18-213 or 15-213), (18-349 or 18-320 or 18-340 or 18-330 or 18-341 or 18-344)**
- **Notes**
  - This course consists of a thorough introduction to many aspects and tradeoffs involved with designing a processor, including pipelining, superscalar execution, and cache coherence.
  - Labs primarily focus on an implementation of a RISC-V-compliant processor, as well as several critical optimizations that are seen in nearly all modern CPUs.
  - Students complete labs in groups of 3. Try to find reliable partners before the class begins, as flaky partners will significantly increase your workload.
  - Exams are known to be tight on time, make sure you have a solid grasp on the material before taking midterms.

🏠

Signals and Systems


Mathematical Foundations (18-202 or two math)
18-220: Electronic Devices and Analog Circuits
18-290: Signals and Systems
18-372: Fundamentals in Electric Energy Systems
18-370: Fundamentals of Control
18-491: Fundamentals of Signal Processing
Probability (36-225, 21-325, 36-219)
18-793: Image and Video Processing
18-792: Advanced Digital Signal Processing
18-794: Pattern Recognition Theory

18-370: Fundamentals of Control

- **Sem. Fall   Units: 12   FCE: 7.8 hrs/week**
- **Prerequisites: 18-220 or 18-290 or 24-352**
- **Notes**
  - Most of the class covers the theory behind controls systems, which are heavily used in robotics
  - Homeworks every 2-3 weeks, midterm and final exams
  - Can be used as a prerequisite for some more interesting classes, such as Embedded Controls Systems (18-474), or other robotics courses.
  - Class does not take much work, but make sure to attend lectures to know what is going on

18-491: Digital Signal Processing

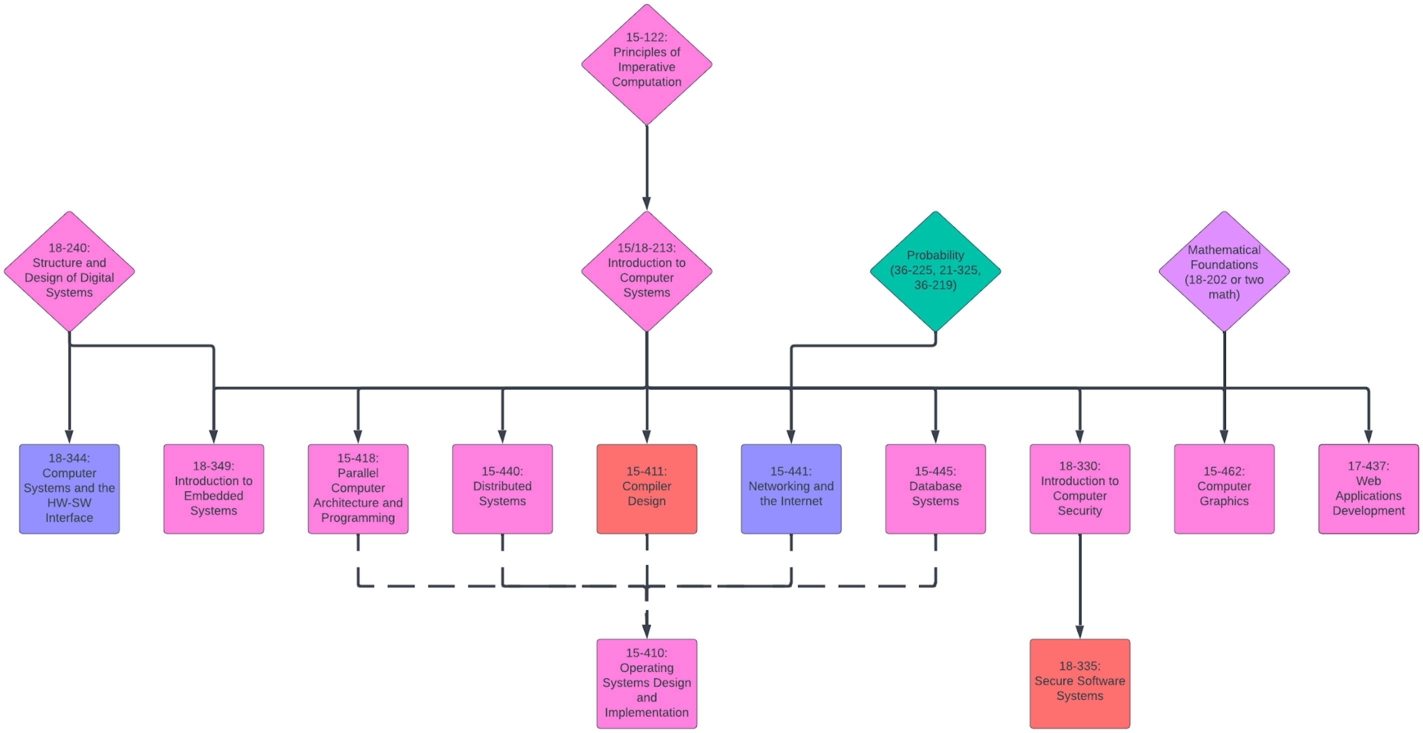
- **Sem. Spring   Units: 12   FCE: 12.8 hrs/week**
- **Prerequisites: 18-290**
- **Notes**
  - Entry course into the higher level signals courses 18-792 (Advanced Digital Signal Processing) and 18-793 (Image and Video Processing)
  - The first half of this class provides a mathematical-based introduction to 1D discrete signals and the second half explores applications of DSP and builds upon more mathematical concepts.
  - By the end of the course, students should know how to sample discrete signals and build different types of filters
  - The homeworks and exams are much more difficult than those given in 18-290 – be wary
  - Attending lectures is highly recommended because the professor works out the concepts and equations on the spot, which provide students a better understanding of the content

18-792: Advanced Digital Signal Processing

- **Sem. Fall   Units: 12   FCE: 15.2 hrs/week**
- **Prerequisites: 18-491, Probability (36-225 or 36-219 or 21-325)**
- **Notes**
  - Dives deep into many essential signals processing topics, such as sampling, polyphase filters, fourier analysis/synthesis, lattice filters, adaptive array processing, and more
  - This class has you implement many algorithms pertaining to speech and music data, allowing students to apply learned topics
  - Make sure to use all supplemental resources (textbook, recitation, office hours) because they will make your life much easier when completing homeworks

Software Systems

Note: Many other courses count towards the software concentration, including Computer Science courses. (check ECE Academic Guide for a complete list) However, others are less frequently taken.


15-122: Principles of Imperative Computation
18-240: Structure and Design of Digital Systems
15/18-213: Introduction to Computer Systems
Probability (36-225, 21-325, 36-219)
Mathematical Foundations (18-202 or two math)
18-344: Computer Systems and the HW-SW Interface
18-349: Introduction to Embedded Systems
15-418: Parallel Computer Architecture and Programming
15-440: Distributed Systems
15-411: Compiler Design
15-441: Networking and the Internet
15-445: Database Systems
18-330: Introduction to Computer Security
15-462: Computer Graphics
17-437: Web Applications Development
15-410: Operating Systems Design and Implementation
18-335: Secure Software Systems

18-330: Introduction to Computer Security

- **Sem. Any Units: 12 FCE: 11.5 hrs/week**
- **Prerequisites: 18-213**
- **Notes**
    - One of the easiest Software Systems classes, often one of the first taken by people doing the concentration.
        - Low hourly requirement, and TAs willing to provide help, generous bucket system, take advantage of them
    - Material covers “attack lab on steroids”, cryptography proofs, human factors, and web-based attacks
    - 21-127 knowledge may be helpful for understanding cryptography, but not required.

18-349: Introduction to Embedded Systems

- **Sem. Any Units: 12 FCE: 14.7 hrs/week**
- **Prerequisites: 18-240, 18-213**
- **Notes:**
    - This course covers a broad introduction to embedded systems, including both software and hardware-oriented topics such as scheduling, concurrency or PCB manufacturing
    - The labs make up a majority of the work in this course.
        - Labs are largely implementation-based. There is emphasis on learning how to read datasheets, and lots of bit-level hardware manipulation involved
    - WARNING: This class has had drastic changes in the past two semesters, and is currently in a very volatile state. Sources regarding more specific details may be inaccurate

15-410: Operating System Design and Implementation

- **Sem. Any Units: 15 FCE: 24.6 hrs/week**
- **Prerequisites: (15-411 or 15-418 or 15-441 or 15-445 or 15-440)**
- **Notes:**
    - This class has an extremely heavy workload! Unless you are extremely confident in your abilities, take this class with no more than 1 lighter technical class, and light Gen Ed classes.
    - In this class you write an operating system from scratch, which gives you
        - Experience designing in a larger codebase
        - Practice designing safe and robust concurrent code
        - Experience making good design decisions. (The course staff make it a point to not tell you if a design is good or bad)
    - This is a partner class, so it can be good to have a partner set aside before taking the class. This is especially important because it is extremely difficult to complete all the work for a lab on your own.

15-411: Compiler Design

- **Sem. Spring Units: 15 FCE: 22.5 hrs/week**
- **Prerequisites: 15-213 or 18-213**

🏠

- **Notes:**
    - This class has a heavy workload, take a lighter schedule alongside it
    - Functional languages pose a significant advantage when writing a compiler. Make sure you know OCaml or Rust. If you have learned SML from 15-150, OCaml is easy to pick up
    - The labs all build on each other. This is one of the few classes where you can actively see tech debt affect you in the long run, which is itself a valuable experience
    - This is a partner class. While can be better to know your partner beforehand, 15-411 does a good job matching partners with the same ambitions and skill levels using a questionnaire

15-418: Parallel Computer Architecture and Programming

- **Sem. Any Units: 12 FCE: 14.3 hrs/week**
- **Prerequisites: 15-213 or 18-213**
- **Notes**
    - This class primarily focuses on parallelizing algorithms using hardware and software constructs
    - The majority of the work from this class comes from labs
    - Exams are doable, but cover a wide array of concepts. Make sure to set aside enough time to cover all of them.
    - Having a good partner in this class is useful. Code is written in C++, which can be very difficult to debug when using low-level software constructs.

15-440: Distributed Systems

- **Sem. Any Units: 12 FCE: 15.7 hrs/week**
- **Prerequisites: 15-213 or 18-213**
- **Notes**
    - Class covers how to effectively distribute a workload among many computers.
    - Known to be one of the easier Systems classes, but ramps up towards the end
    - Lectures are somewhat dry, and homeworks cover different material than the labs
    - Projects vary greatly in difficulty
    - WARNING: This class varies greatly depending on whether you take it in the fall or spring
    - Fall:
        - Language: Go
        - Go handles much of the tricky concurrency for you, allowing students to do more conceptually challenging labs
        - Partner-based
    - Spring:
        - Language: C/C++ or Java, depends
        - More focus on how to write robust concurrent code in a lower-level language
        - Not partner-based

Common Minors for ECE Majors

Make sure you talk with the advisor associated with your desired minor. This is necessary for:

- Applying for the minor
- Clearing up the many ambiguities with double-counting classes
- Planning your schedule to ensure you finish the minor at a reasonable pace

Minor in Computer Science

This is a popular minor because many of the prerequisites/requirements, such as 15-112, 21-127, 15-122, and 15/18-213, are already required for ECE. Many ECEs are also interested in other CS theory / software engineering concepts that are not taught in ECE

Notes:

- Requirements
- The requirements state that either 15-251 or 15-213 are required. However, ECE students must already take 18-213, which covers the exact same material as 15-213. As such, they must take either 15-251, or one additional CS elective (3 in total)
- ECE students can double-count a maximum of 2 courses towards a CS minor, unless they fulfill the ECE free elective requirement.

15-150: Principles of Functional Programming

- **Sem. Any Units: 12 FCE: 12.8 hrs/week**
- **Prerequisites: (15-151 or 21-127 or 21-128), 15-112**
- **Notes**
    - Introduction to functional programming using the language SML
        - This class acts as many students first introduction to functional programming thus they may find it more challenging than other CS classes
    - Similar workload and difficulty as 15-122
    - Homeworks consist of coding and written assignments, homeworks are a large portion of the final grade with many students struggling with the written assignments

15-210: Parallel and Sequential Data Structures and Algorithms

- **Sem. Any Units: 12 FCE: 14.3 hrs/week**
- **Prerequisites: 15-122, 15-150**

🏠

- **Notes:**
    - This class focuses on teaching students to devise algorithms to solve a wide variety of problems, and the tools necessary to accurately reason about their speed.
    - Most of the time is spent doing homeworks (1 per week), which typically feature difficult problems you must solve and implement a solution for.
    - It is very easy to get stuck while trying to find a solution.
        - Office hours can be very good for getting you unstuck. However, queues are very long
        - If you have friends taking the class, it is better to discuss the problem than it is to be stuck on it for hours
    - The ability to do accurate complexity-bound analysis is extremely valuable for interviews. Take this class early to take advantage of this.

15-251: Great Ideas in Theoretical Computer Science

- **Sem. Any Units: 12 FCE: 14.5 hrs/week**
- **Prerequisites: (15-122 or 15-150), (21-127 or 21-128 or 15-151)**
- **Notes:**
    - Only required to take either this class or 15/18-213
    - Not recommended for students pursuing the minor who did not enjoy 21-127/21-128 as this course is significantly harder and has a heavier workload
    - Second discrete math class that computer science students will take, gives broad introduction to concepts such as graph theory and the P vs NP problem
    - Students form groups of 4 at the beginning of the semester which they will do the homework problems with
    - Instead of handing in homework in a traditional manner, this class employs weekly “writing sessions” which are essentially 1.5 hour tests where students will rewrite solutions to the week’s homework
        - **Note:** While this seems like it would encourage pure memorization, students should still aim to understand every homework problem as memorizing every question is infeasible and punishes students on exams, which are the majority of the final grade

Minor in Machine Learning

Machine learning is currently a very hot topic in both academia and the industry. As such, many ECE students choose to take ML classes to be competitive for these positions

Notes

- Most relevant information can be found on the ML minor requirements page

🏠

- If you are interested in machine learning courses, namely 10-301/10-315, prioritize taking its prerequisites: 15-122, 21-127/21-128, and one of the probability courses

10-301/10-315: Introduction to Machine Learning

- **Sem. Any Units: 12 FCE: 13.6 hrs/week**
- **Prerequisites: 15-122, (21-127 or 21-128 or 15-151), (36-219 or 15-259 or 36-225 or 21-325 or 36-217 or 36-235 or 36-235 or 15-359 or 36-218 or 36-220)**
- **Notes:**
    - This is an introductory course in Machine Learning. By the end, you can expect to have a reasonable understanding of what researchers are talking about, but not necessarily enough to do advanced deep learning research.
    - The concepts in this class are difficult to understand, but the grading scheme is consequently generous. As such, the Homeworks and Tests tend to be the more difficult parts of this course
    - Weekly time requirement is similar to 21-127

10-417: Intermediate Deep Learning

- **Sem. Fall Units: 12 FCE: 11.2 hrs/week**
- **Prerequisites: (10-301 or 10-315 or 10-715 or 10-601 or 10-701)**
- **Notes:**
    - Covers a solid amount of theory and probability involved with modern Deep Learning
    - Unlocks several more interesting ML electives
    - This is a common class to take after 10-301
    - Only has 3 homeworks and 1 project (no quizzes, no exams). However, beware; each of these counts for a lot
    - Project requires partners. It is worth 30% of your final grade, so find a good partner.
    - Not too difficult to get a good grade if you do all of the work

10-414: Deep Learning Systems

- **Sem. Fall Units: 12 FCE: 11.7 hrs/week**
- **Prerequisites: (15-213 or 18-213 or 15-513), (21-240 or 21-241), (21-128 or 21-127 or 15-151), (10-301 or 10-315 or 10-601 or 10-701 or 10-715)**
- **Notes:**
    - Learn about algorithms and hardware optimizations involved in making deep learning fast. While 10-301 glosses over how certain algorithms work, DLS describes them in detail, and how they can be optimized.
    - Machine Learning is a hot topic, and this can be an entrypoint for Systems programmers to take part in it
    - Lab-only class, so if you can finish all of the work, a good grade is not too difficult to get.
    - Can be time-consuming, but is rewarding

Minor in Robotics

This minor is popular amongst ECE students as many electives can count for both majors. Additionally, this minor allows students to more deeply explore the hardware software interfaces that are vital to robot operations.

Notes:

- Requirements
- You are restricted to double-counting a maximum of 2 courses, not including the prerequisites, towards other degrees and minors
- The robotics minor has 3 required electives and 2 free electives. Several courses that fulfill these elective requirements also fulfill ECE area and coverage requirements such as 16-384 (Robot Kinematics and Dynamics), and 18-349 (Introduction to Embedded Systems).

16-280: General Robotic Systems

- This course does not exist yet. However, it will replace 16-311 to act as the introductory robotics course starting next year.

16-311: Introduction to Robotics

- **Sem. Spring Units: 12 FCE: 17.5 hrs/week**
- **Prerequisites: (24-311 or 21-260 or 18-202 or 21-241 or 21-240)**
- **Notes:**
    - Course has a very heavy workload for a introductory course
    - Covers a range of core concepts from computer vision to kinematics
    - Written homeworks tend to be light and occasionally have a coding component
    - Lab work will take up the majority of your time in this class
        - Students are advised to form groups of 3 with differing majors so that all of ECE, CS and Mechanical Engineering are covered
        - Labs are also a significant portion of your final grade so it is advised to start them early and to seek help from TAs often
    - General advice is to not take this class until later, many students opt to take 16-311 after all of the other requirements due to the excessive workload and rigor
        - While still not known, it is likely that this class' replacement, 16-280 will be a much friendlier introductory course so we advise taking that course instead when it comes out

Creating Your Own Schedule

Guidelines for planning your schedule

Use your resources

- We recommend taking a look at our **Supplemental Resources** section
    - cmucourses.com is great for learning about the curriculum and statistics of your courses, such as the average workload and quality of the professors
        - Most of the time, the statistics don't tell the full story, so we recommend speaking to people who have taken the courses you are interested in. You can likely find these people in the CS/ECE Study Group discord
    - Stellic is good for creating a plan over all of your years at CMU
        - It is slightly buggy, especially when adding minors/majors
    - Your advisors can answer other questions, such as graduation requirements and double-counting
        - Definitely check in with your advisor from time to time to ensure your plans put you on track to graduate
        - **Note:** Your advisors do not want you to take multiple cores/area classes simultaneously

Learn and follow your limits!

- We suggest taking at most three difficult classes (roughly ≥14 FCE) at once, but if you feel that is too much or too little, adapt accordingly!
    - Spend your earlier years at CMU learning your study habits, strengths and weaknesses, and generally how much work you can do without burning out
    - Don't be afraid to drop classes and use vouchers if necessary! Mental health is key
        - More information about P/F and course dropping here
- Consider your life outside of classwork
    - To stay sane, everybody should have fun activities they do outside of classwork. Burnout should be avoided at all costs
    - Factor these extracurricular activities into your weekly hours
        - If you have significant extracurricular work, don't be afraid to maybe take one less class

Plan out your cores and track courses at least one semester ahead

- As stated in the tracks section, many track courses only run in either the fall or spring semesters, so it important to know exactly when you want to take them
    - Nobody wants to wait a whole year for a course to come around

Try to make each semester evenly distributed in workload

- Don’t take all your general education requirements in one semester and don’t take all your cores in one semester – pace yourself!
- Striking a consistent balance between workload and other activities will help you perform consistently in classes and thus make school more enjoyable

Have backups for courses, especially core/track courses

- With how the registration system works and how popular many courses are, it is inevitable that you won’t receive all the classes you want
    - This can be especially annoying for core/track courses that carry much more weight in your current and future schedules
- To get around this, always have some backups that will keep you on track to graduate and prevent any major changes to future schedules
    - If the course is absolutely necessary and you cannot substitute it, try talking to your advisor about letting you into the course

🏠

Supplemental Resources

Of course, there are other resources on ECE courses and course and registration in general. Here are the main resources:

- **HKN CMU ECE/CS Guide**
    - Useful for seeing descriptions of courses from a student's perspective
    - **WARNING:** Outdated and possibly misleading guide for certain courses. As a comical example, see the 18-447 webpage
- **Official ECE Academic Guide**
- **CIT Freshman Course Registration Guide**
    - Course Information Browser (CMU Courses)
        - The go-to resource for finding the history of FCEs, professors, prerequisites, descriptions, and more for every course!
    - Course Planner (Stellic)
- **CS/ECE Study Group @ CMU**
    - An active Discord server filled with CS and ECE upperclassmen who can provide up-to-date suggestions on courses and more
    - Fun place to hang out!
- **Your advisors**
    - For any questions none of these other resources can answer

Updating This Document

An important part of this document is maintaining its usefulness over many years. As classes, professors, and requirements change, this guide can become outdated quickly. Thus, we have instructions for keeping this guide fresh.

Goal: This guide is meant to teach ECE students how to optimize their course selection by providing common progressions for all 5 concentrations and personal descriptions for courses in these tracks

GitHub: This document is available on GitHub, meaning it is viewable by anyone. Any time a change is made, we will commit, or save, our changes. This will create a history of changes that we can revert to in the case of misinformation or general contamination of the document. Additionally, whenever someone wants to change the document, they can submit a pull request (PR), which can be discussed by students and ultimately accepted or denied. This will allow for community discussions and constant changes to the document. At some point, we will transfer this document to new owners, who will stay updated on classes and make decisions on new pull requests.

Guidelines: Whenever you want to change the document, submit a PR! (link)

- **Note:** We have by no means taken every single one of these classes. If you think there is an important class to add, or some information that could be better phrased or emphasized, feel free to make a PR. However, try to stick with the spirit of listing common CMU classes and providing unbiased, useful class descriptions
- If class descriptions become outdated, update them
- If any links become outdated, which they will, update them
- If a class isn't taught anymore, replace it with the equivalent course
  - Equivalent means similar prerequisites, course content, and importance to the track