# CSCE 236: Embedded Systems

# Spring 2020

### Instructor

Jeffrey Falkinburg
Computer Science and Engineering
368 Avery Hall
jfalkinburg2@unl.edu

Office Hours:
See Course Website: Falkinburg in 368 Avery Hall
and by appointment.

### Teaching Assistants

Undergraduate TAs:
Cody Berglund, cberglund@cse.unl.edu
Garrett Beard, gbeard@cse.unl.edu
Daniel Shchur, dshchur@cse.unl.edu
Dan Thibodeau, dthibodeau@cse.unl.edu
Ryan Wallace, rwallace@cse.unl.edu

Office Hours:
See Course Website

### Course Information

Tuesday Lecture: 9:30-10:45am in Louise Pound Hall 124
Thursday Lecture: 9:30-10:45am in Louise Pound Hall 124
Thursday Lab: 11:00-12:15pm in Louise Pound Hall 21

### Course Website

The website for the course is: https://cse.unl.edu/~jfalkinburg/cse_courses/2020/236/index.html

We will also be using Canvas, Piazza, and Bitbucket.

This syllabus is subject to change, you will find the most up-to-date version of the syllabus on the course website, as well as information on readings, assignments, and projects. Please make sure to check it regularly. In addition to posting information and assignments on the course website, I will send information over email to the address you gave me at the start of the course.

I expect that you will check your email on a daily basis. Please make proper arrangements if you will not be able to check your email or if your email address changes.

## Email

It is CSE Department policy that all students in CSE courses are expected to are regularly check their email so they do not miss important announcements.

In this course, important announcements will be made through Canvas and you are expected to regularly check e-mail connected to Canvas.

## Course Description

Embedded Systems are everywhere. Every time you look at your watch, answer the phone, take a picture, or turn on the TV you are interacting with an embedded system. Embedded systems are also found in cars, airplanes, and robots. They far outnumber traditional computers (which also contain embedded processors) and learning to design and program embedded systems is a critical skill that is necessary for many industry and scientific jobs.

In this course, you will learn the fundamentals of designing, interfacing, and programming embedded systems. We will make use of the Arduino platform, which is an inexpensive, popular embedded system used by hobbyists, researchers, and in industry, to implement the techniques learned in class. By the end of the course you will have mastered the basics of embedded system design and programming, which will prepare you for careers in industry and scientific research.

## Datasheets

Arduino Uno R3 schematic: The schematic of the Arduino we are using in this course.
Atmega328 datasheet: Datasheet for the processor on the Arduino Uno we are using in this course

## Prerequisite

CSE 230, or equivalent; or instructor permission

## Course Books

Required:
Marilyn Wolf, Computers as Components, Third Edition: Principles of Embedded Computing System Design, 4th ed. Morgan Kaufmann, 2016
Available in the book store or purchase online.  Readings from this book are specified in the format Wx.x, where x.x indicates the chapter and subsection.

David Russell, Introduction to Embedded Systems, 2010.
Available for free download when on the UNL campus. Good reference for embedded C programming. Do not print out this book, it is less expensive to order a printed copy than to print it yourself. Readings from this book are specified in the format Rx.x, where x.x indicates the chapter and subsection. **Note:** this book is specifically designed for the Arduino platform, but

there are some differences between the version of the Arduino used in the book and the version we are using in class. So be careful!

Recommended / Optional:
Edward Lee and Sanjit Seshia, Introduction to Embedded Systems, A Cyber-Physical Systems Approach, 2011.
Available for free download. Do not print out this book, it is less expensive to order a printed copy than to print it yourself. There are no specific readings from this book, but it is a good reference for those interested in exploring some subjects further.

Brian W. Kernighan and Dennis M. Ritchie, C Programming Language (2nd Edition), 1988.
ISBN: 0131103628

A wiki-style C "text book" alternative
Another online C text book
C Frequently Asked Questions from the days of Usenet
Suggested references for those with little prior C programming experience. There are no specific readings from these, but they are good references.

## Topics Covered

**Embedded System Organization:** Major components in a typical embedded system, operating requirement, modes of operation, hardware/software codesigns, hardware-software trade-offs.

**Microcontroller Programming:** Basic structures of microcontrollers, basic features, memory interfacing, digital I/O, timers, analog interfaces, interrupt services, programming in high-level languages and assembly languages, basic data types, operators, constructs, data structures, compiler directives, power management.

**I/O Interfacing Concepts:** Input devices, output devices, memory mapping, bus structures, peripheral and external communication interfaces. Operating System: Design and organization of embedded and real-time operating systems, scheduling, power management, communication, debugging.

## Assignments and Grading

Assignments and due dates will be announced in class. Typically, all assignments are due before the beginning of class on the day they are due. Note that many of the assignments require check-off by the instructor or TA *before* the start of class.

Your final grade will be composed of a number of components. These are:

| Percentage | Assignment |
|---|---|
| 20% | Lab Assignments, Pop Quizzes, and Participation |
| 20% | Homeworks |
| 15% | Exam 1 |
| 15% | Exam 2 |

| | |
|---|---|
| 15% | Project 1 |
| 15% | Project 2 |
| 100% | Total |

Final grades will be assigned based on the following cutoff percentages

| Grade | A+ | A | A- | B+ | B | B- | C+ | C | C- | D | F |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Score | 97 | 93 | 90 | 87 | 83 | 80 | 77 | 73 | 70 | 60 | <60 |

## Homeworks

There will be a number of homeworks assigned throughout the semester. These will typically have both questions and programming components. Most often these will require demonstrating the final operation to the instructor or TA before the assignment is due during one of their regularly scheduled office hours, so please plan accordingly.

These are individual assignments. It is ok to discuss concepts behind the problems in the homeworks with classmates, however, you cannot do them together. If you do discuss problems with classmates or other people, you must acknowledge this on the assignment (this will not lead to any grade reduction). As a metric for what level of discussion is allowed, it is ok to meet and talk over coffee about the assignment. It is not ok to show someone your solution or to work on the details of the problems together. In addition, you should not take notes while discussing the problems. If in doubt, ask the instructor or TA questions about assignments.

Late homeworks will not be accepted.

## Labs

There will be a number of small group assignments that will be done during class times. Some labs may have small components that must be performed individually before the start of lab. Groups will be randomly assigned at the start of class. Your group will be graded based on what you are able to successfully accomplish during class. Homework assignments will often build on these mini-labs. If you are not in class or are not properly prepared for the lab, you will receive a zero for that lab.

## Pop Quizzes

There will be several unannounced short quizzes at the very start of some classes. These will be open book and notes and will cover materials recently discussed in class. There are no makeups for these quizzes, so if you are not in class you will receive a zero on the quiz.

## Course Project

There will be two projects in this class that will build on and combine the components you develop in your homeworks. The projects will involve an in-class competition where you compete with your classmates to accomplish a task. In addition, there will be an associated project report. The first project/competition will be due in the middle of the semester. The second project/competition will be due the last week of classes. Additional details will be announced in class.

## Absences

Students are encouraged to attend all classes. Absences will likely lead to a reduction in the class participation grade. In addition, missing an in-class lab, exam, or project competition will result in a zero for that assignment and being absent is not an excuse for not turning in an assignment.

In very rare cases the instructor may make an exception to this policy in case of an illness or family emergency if acceptable written evidence is given and you notify me as soon as possible. Even if you are sick or leaving town on short notice you should be able to have a friend notify me that you will be missing class or an assignment.

## University Writing Center

The University of Nebraska-Lincoln Writing Center can provide you with meaningful support as you write for this class as well as for every course in which you enroll. Trained peer consultants are available to talk with you as you plan, draft, and revise your writing. Please check the Writing Center website (http://www.unl.edu/writing/) for locations, hours, and information about scheduling appointments.

## CSE and UNL Policies

The CSE Department has a Student Resource Center in Avery 13A (http://cse.unl.edu/src) where you can go for assistance with any of your courses.

All students enrolled in any computer science course is bound by the Computer Science and Engineering academic integrity policy:
http://cse.unl.edu/ugrad/resources/academic_integrity.php
You are expected to read, understand, and follow this policy.

For this course, do not plagiarize (writing or code) and make sure to properly cite any sources you use. Any cheating or plagiarism will be reported to the Chair of your department and your Dean, and will result in an F for the assignment and possibly for the course.

The CSE Department has an anonymous suggestion box:
http://cse.unl.edu/department/suggestion.php
that you may use to voice your concerns about any problems in the course or department if you do not wish to be identified.

## Disabilities

Students with disabilities are encouraged to contact the instructor for a confidential discussion of their individual needs for academic accommodation. It is the policy of the University of Nebraska-Lincoln to provide flexible and individualized accommodation to students with documented disabilities that may affect their ability to fully participate in course activities or to meet course requirements. To receive accommodation services, students must be registered with the Services for Students with Disabilities (SSD) office, 117 Lewis Pound Hall, 472-3787 voice or TTY.

## Updated Course Policies for Online Instruction

Like we discussed in class previously the course will be moving to an online format for the rest of the semester. The following sections discuss how we will be transitioning to online instruction.

## Communication Plan

I will continue to communicate course updates and changes via Canvas and Piazza. Additionally, I will host a Zoom meetings during the regular class/lab periods. The TAs and I will still be responsive to on Piazza and will connect via the Zoom link posted on Canvas and Piazza and during our posted office hours to help you stay on track with the class!

Communication is a two way street. Make sure you let me know if you have any issues getting things completed and turned in on time. Since we will no longer have in-person class sessions it will be imperative that you start publishing public videos to YouTube or gif files showing your lab/homework/project functionality. The last lab and homework assignments will be converted to a completely Canvas submission for functionality proof, questions, and code. You will be required to in insert images and/or video links of functionality into your Canvas submissions. All the code, proof of functionality documents, and final report for the Project Part 2 should still be pushed to BitBucket as normal. However, just like I did for Project 1; I will make Canvas submissions for functionality proof videos on Canvas so that we can quickly grade your submissions.

## Remote Teaching Plan

There will not be too many formal lectures for the rest of the semester since most of the work is an individual lab, homework, and final project. For traditional course lectures I will continue to post videos for the lectures and will host meetings via Zoom during the class period for you to ask questions about the material and get assistance on the labs and projects. It will be imperative that you review the lecture videos/notes and keep up on the labs and homeworks to be successful on Exam 2 and in the course as a whole. Please take the time to watch the videos and review the online notes, handouts, and slides.

## Assessment Plan

Assessments will be submitted online via Canvas and Bitbucket. There are four major assessments left on the syllabus and here are the updated plans for each:

- **Lab 5:** Completely Canvas submission for functionality proof, questions, and code.
- **Homework 5:** Completely Canvas submission for functionality proof, questions, and code.
- **Project 2:** Canvas submission for functionality proof and Final Report. BitBucket will be used for code, documentation, and the final report submission backup.

- **Exam 2:** Exam 2 will be hosted via Canvas and Responsdus Lockdown Browser with monitor. Potentially it may be a combination of Canvas Quiz with a take home portion.

### Backup Plan

If we are unable to connect via Zoom I will post links to Microsoft Teams or Google Hangouts.

### Course Schedule

See Course Website