ECE 445L Lab 7A

Prototype and Low-Level Software for an Embedded System

This laboratory assignment accompanies the book, [*Embedded Systems: Real-Time Interfacing to ARM Cortex M Microcontrollers, ISBN-13: 978-1463590154*](https://www.amazon.com/Embedded-Systems-Real-Time-Interfacing-Microcontrollers/dp/1463590156), by Jonathan W. Valvano, copyright © 2024.

# Table of Contents

[Table of Contents 1](#_Toc202438752)

[Team Size 3](#_Toc202438753)

[Goals 3](#_Toc202438754)

[Review 3](#_Toc202438755)

[Starter Files 3](#_Toc202438756)

[Required Hardware 3](#_Toc202438757)

[Teamwork evaluations 3](#_Toc202438758)

[Project planning 3](#_Toc202438759)

[Project Steps 3](#_Toc202438760)

[Project Constraints 5](#_Toc202438761)

[Competition Constraints 6](#_Toc202438762)

[Optional Components 6](#_Toc202438763)

[Pre-preparation 7](#_Toc202438764)

[Requirements Document 7](#_Toc202438765)

[Partner Signup 7](#_Toc202438766)

[Pre-Preparation Deliverables 7](#_Toc202438767)

[Preparation 7](#_Toc202438768)

[KiCad Project 7](#_Toc202438769)

[Preliminary BOM 7](#_Toc202438770)

[Preparation Deliverables 8](#_Toc202438771)

[Procedure 8](#_Toc202438772)

[Requirements Document 8](#_Toc202438773)

[Datasheets 8](#_Toc202438774)

[Procurement of parts 8](#_Toc202438775)

[Updated BOM 8](#_Toc202438776)

[Hardware schematic 8](#_Toc202438777)

[System data flow graph and call graph 9](#_Toc202438778)

[Build and test hardware and low-level software 9](#_Toc202438779)

[Lab Checkout 9](#_Toc202438780)

[TA and Professor Review 9](#_Toc202438781)

[Lab Report 10](#_Toc202438782)

[Deliverable 1 10](#_Toc202438783)

[Deliverable 2 10](#_Toc202438784)

[Deliverable 3 10](#_Toc202438785)

[Deliverable 4 10](#_Toc202438786)

[Deliverable 5 10](#_Toc202438787)

[Deliverable 6 10](#_Toc202438788)

[Deliverable 7 10](#_Toc202438789)

[Deliverable 8 (5pts Extra Credit) 10](#_Toc202438790)

[Deliverable 9 (10pts Extra Credit) 10](#_Toc202438791)

[Hints 11](#_Toc202438792)

[Other Stuff 11](#_Toc202438793)

# Team Size

The team size for Labs 7A, 7B and 10 is **4**.

# Goals

* Create a proposal and requirements document for your embedded system,
* Design a preliminary embedded system at the schematic level,
* Implement low-level software for each component of the system,
* Collect components and implement a prototype of your embedded system on a breadboard,
* Develop unit testing skills to evaluate your subsystems,
* And demonstrate that the project is feasible.

# Review

* Data sheets for your microcontroller and for your hardware components.
* [ebook Chapter 6](https://users.ece.utexas.edu/~valvano/EE445L/ebook/Chapter6_SystemComponents.htm)
* [[ebook Chapter 7](https://users.ece.utexas.edu/~valvano/EE445L/ebook/Chapter6_SystemComponents.htm)](https://users.ece.utexas.edu/~valvano/EE445L/ebook/Chapter7_SystemDesign.htm)
* [ebook Section 3.7.5 Team skills contract](https://users.ece.utexas.edu/~valvano/EE445L/ebook/Chapter3_HumanInterfaces.htm#3_7_5)

# Starter Files

* Starter project:
  + Final project sw and hw template provided in the GH Classroom repo.

# Required Hardware

We require you to collect and evaluate all hardware (chips, sensors, and actuators) needed for the final project.

# Teamwork evaluations

Each student will get a significant evaluation of their teamwork performance (5% of the total ECE445L grade). There should be four or more major components to the project, so we expect each member of the team to be responsible for at least one major component. The TA will show you your teamwork score after Lab 7B, and you will have a second teamwork performance grade for Labs 8, 9, and 10. The grading rubric considers:

* Completion and understanding of your assigned responsibilities(s) for the project.
* The number of meaningful commits you made to your teams GitHub repo.
* The effort you contributed to the project.
* Your ability to resolve conflicts.
* Your participation in your teams’ communications. You are encouraged to show your TA the communication during Labs 7A and 7B.

# Project planning

## Project Overview

Ideally, you and your team will go through multiple cycles of the design process to identify bugs, refine the design, and reduce project risk. However, given the limited time of this class you will only have one pass. One way to reduce your design risk is by performing unit tests here in Lab 7A that verify that your components will work as expected for the final system.

Lab 7A begins with a preliminary requirements document. Next, you will collect all hardware chips, sensors, and actuators needed for the project. Using LaunchPads and solderless breadboards, you will interface all input and output devices to the LaunchPad. Next, you will write low-level software to perform input and output for all your I/O devices. See Figure 7A.1.

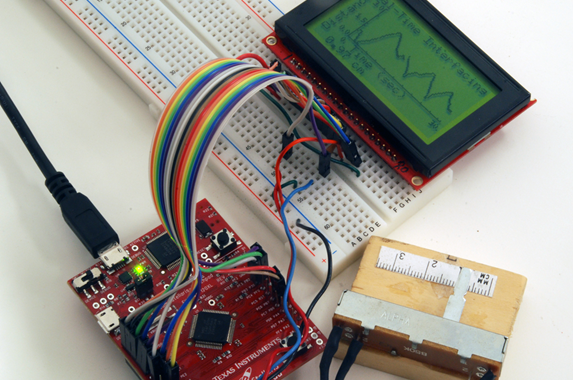


Figure 7A.1: Result of Lab 7A.

As part of Lab 7B, you will layout the PCB and consider how the system will be powered enclosed and tested. See Figure 7A.2.

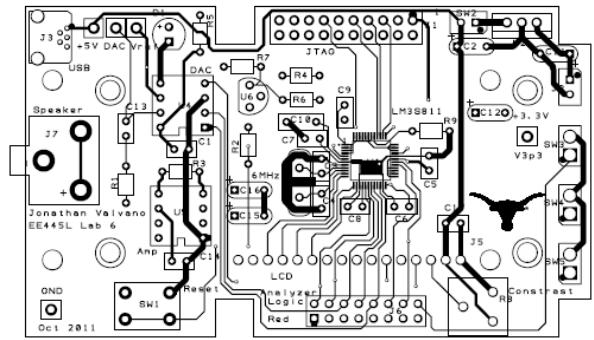


Figure 7A.2: Result of Lab 7B.

As part of Lab 10, you will build and test the system, writing the high-level software. See Figure 7A.3.

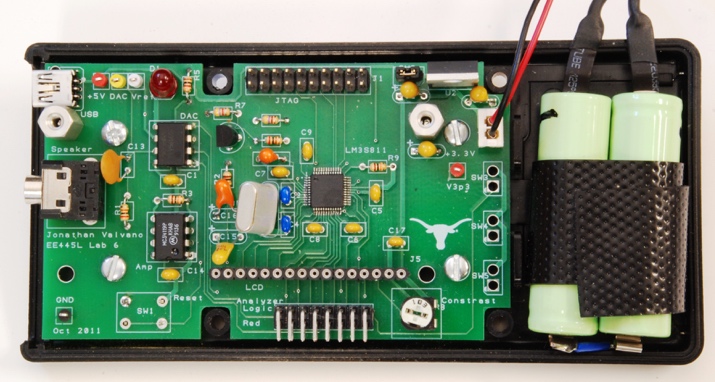


Figure 7A.3: Result of Lab 10.

## Project Constraints

The design of the system must satisfy certain requirements. Rather than simply redesigning one of the previous labs, this embedded system must do something new and useful. There are some ideas posted on Valvano’s [page](https://users.ece.utexas.edu/~valvano/EE445L/projectideas.htm), but you have flexibility to define exactly what it is to do. If you look at [Sparkfun.com](https://www.sparkfun.com/) and [Adafruit.com](https://www.adafruit.com/) you will see lots of ideas of I/O devices that you could attach to the system. The scope of the project is a microcontroller-based embedded system demonstrating the educational objectives of this class.

There are separate constraints for the project and the final design competition. You can still get a 100 on the final project without conforming to the design competition constraints, however you cannot compete officially in the competition if you do not satisfy all constraints.

1. The system should perform something useful.
2. The system shall include at least two inputs, two outputs, and two interrupt service routines.
3. The system must contain four or more identifiable major subcomponents, where each team member must be responsible for at least one major subcomponent.
4. This system shall mount the LaunchPad or TM4C123 processor, and other chips to the PCB.
5. The final system must fit in an appropriate enclosure.
6. TAs will judge if the project is sufficiently complicated.
7. Each team can order a two-layer PCB up to 30 square inches (e.g., 5 by 6 in) from JLCPCB.
8. If you use an ESP8266 from the checkout desk, do NOT solder it to the PCB.
9. Each PCB must conform to [JLCPCB's design capabilities,](https://jlcpcb.com/capabilities/Capabilities)
10. Parts that are not provided must be purchased by the group.
11. You can use two motors and their corresponding wheels from the Lab 8 supplies, but they must be returned.
12. You should NOT use a ground pour (this will introduce difficulties with debugging and fixing your board if there are any errors).

## Competition Constraints

There will be a "science fair"-like public demonstration for Lab 10. Students with the best design will be presented with special awards. The judging will be performed by the other ECE445L students by viewing demos in person.

Competition restrictions:

1. The design shall use only the TM4C123 as the primary microcontroller.
2. I/O components such as LCD displays, switches, sensors, LEDs, speakers, keypads, and microphones can be off the PCB.
3. All other electronics (resistors, capacitors, diodes, ICs, etc.) shall be on the PCB.
4. The team shall spend no more than $60 on extra components not provided for other labs.

Details on the $60 budget:

1. There are two system costs you will calculate:
   1. Cost of all components (regardless of where they were obtained)
   2. Cost of all components that factor into the $60 limitation.

The 1b) metric will be used when determining whether the project is eligible for winning the competition. For example, you happen to already own a component that is unlikely for every group to also already own; the 1b) cost will be the price for another group to purchase the same component. For example, the 1b) cost will be 0.

1. Parts that do NOT count toward the 1b) $60 limitation:
   1. Parts obtained from the ECE lab checkout counter or the [Lab7BOM.xlsx](https://github.com/ECE445L/ECE445L-Final-Lab/blob/cbc8974bb119772c4ba4bbe3bc5a70f1535af41d/resources/bom/Lab7BOM.xlsx)
      1. Passive components do not have to be returned
      2. Solid state components may have to be returned (please ask the checkout counter)
   2. Free samples
   3. Components that is likely for every group to also already own (e.g., speaker, laptop, cellphone, wire, video monitor, or mouse)
   4. The PCB (If ordered by Professor Valvano or Professor McDermott)

## Optional Components

* 1. ESP8266 if you desire Wi-Fi capabilities. Keep in mind that the ESP8266 requires on average 80mA (can go much higher). ESPs from Lab must be returned.



Figure 7A.4: ESP8266 version 1

1. Audio amp:
   1. For single channel you can use an audio amp like the LM4890
   2. For stereo sound, use two separate audio channels and two separate speakers (four speaker wires, not three)
2. Accelerometers and other sensors:
   1. There are lots of devices available as free samples from chip vendors, but be aware that they may be very difficult to solder
   2. The easiest way add sensors is to purchase a module (chip and breakout board) from a hobby store like [SparkFun](https://www.sparkfun.com/), [Adafruit](https://www.adafruit.com/), or [Pololu](https://www.pololu.com/)
3. Enclosure:
   1. You may purchase an enclosure and count it toward your $60 budget.
   2. You may get a big and ugly enclosure from Valvano/McDermott (come check it out)
   3. You may build your enclosure separately (the enclosure will be judged on functionality and not beauty). You can use resources from [Texas InventionWorks (TIW)](https://linktr.ee/texasinventionworks) to create your box (resources used from TIW will not count toward your $60 budget). Here are some resources TIW offers:
      1. 3D printers (requires training)
      2. laser cutters (requires training)

# Pre-preparation

## Requirements Document

Update the template Requirements.docx file in the deliverables folder to reflect your design. The goal is to create a preliminary, one-page **requirements document** for the system. We expect the document to change throughout the project, so keep it up to date as you progress through the design, implementation, and testing phases.

You will be judged on the clarity of your proposal. You should be able to explain what you intend to do and how at an abstract level. This will be like a marketing requirements document (MRD) presentation to your TA. An MRD explains what a product does from a user perspective (how can we sell it).

## Partner Signup

Use the signup sheet during pre-prep. Missing the deadline will incur grade penalties.

## Pre-Preparation Deliverables

1. One-page Requirements.docx file
2. Partner signup

# Preparation

## KiCad Project

Create or repurpose a KiCad project in the hardware folder. As you add components to your schematic, you may need to create or find footprints. Most of the components in the professor’s cabinet are in the ECE445L library. Ask your TA about this semester’s naming convention for PCB submissions.

## Preliminary BOM

You will use the BOM feature of KiCad to list components on the PCB, and you will use the template BOM.xlsx file in the deliverables folder to list the remaining components. At this point, passive components like resistors and capacitors need not be specified. Include integrated circuits, transducers, output devices (displays, speakers), input devices (microphones, switches, keypads), and actuators. List the locations from which you will acquire the parts:

* ECE lab checkout desk
* Professor's cabinet
* Places like [SparkFun](https://www.sparkfun.com/), [Adafruit](https://www.adafruit.com/), or [Pololu](https://www.pololu.com/)

## Preparation Deliverables

1. KiCad schematic file with all major parts added (resistor capacitor values not needed yet)
2. Preliminary Bill of materials
3. System data flow graph

# Procedure

## Requirements Document

By the completion of Lab 7A, you should have a very clear idea about your project. You should be able to describe the lower-level interface of the system. This will be like a product requirements document (PRD) presentation to your TA. A PRD explains how a product will be developed from an engineering perspective (how do we build it).

1. Process: How will the project be developed?
2. Roles and Responsibilities: Who will do what? Who are the clients?
3. Scope: List the phases and what will be delivered in each phase.
4. Prototypes: How will intermediate progress be demonstrated?
5. Performance: Define the measures and describe how they will be determined.

## Datasheets

While compiling the BOM, gather the datasheets of the relevant parts. Note their communication protocols and interfaces, as well as projected current consumption. Collect these datasheets into the deliverables folder **OR** record the links to the datasheets in Datasheets.xlsx.

## Procurement of parts

Collect as many of the components as you can. You will be graded on the completeness. Components for power and enclosure will be performed in Lab 7B and not needed for Lab 7A.

## Updated BOM

The bill of materials should now include capacitors, cables, connectors, and resistors. Determine the expected cost for the system and prepare to justify the cost to the TA. Components for power and enclosure will be performed in Lab 7B and not needed for Lab 7A.

## Hardware schematic

**Complete the schematic** (.kicad\_sch) for the system using [KiCad](https://www.kicad.org/). You must follow these rules:

* 1. All components must have labels (U1, R1, C1, J1, etc) on the schematic and board.
  2. Each IC should have a bypass cap placed as close to the component as possible (look at your component's datasheet to find the recommended size of each bypass capacitor).
  3. For resistors, specify impedance (10k ohms), wattage (1/4 watt), and tolerance (5%).
  4. For capacitors specify capacitance (100uF, tolerance (20%), and material (ceramic, tantalum, electrolytic, etc).
  5. It must pass ERC (ask your TA which errors/warnings you can ignore)

## System data flow graph and call graph

Update the data flow graph if needed. Draw a call graph. The names should match the names of the header files. Both graphs will illustrate how your hardware will interact with low-level software drivers. Add these graphics to the requirement document.

## Build and test hardware and low-level software

1. **Build the hardware** on a solder-less breadboard and connect the hardware to a Launchpad. Because you are doing low-level testing, it is appropriate to use a separate solder-less breadboard and a separate Launchpad to test each module. We recommend using the I/O pins you’re planning to use in the schematic, although you should prepare to have alternative I/O pin initializations if the final layout works out better if you use a different peripheral pinout.
2. **Implement the low-level software** working with the hardware. The low-level modules need not be integrated together. The system integration will be a Lab 10 task. The goal is to perform enough low-level testing to be confident the wiring of the design is proper.
3. **Write unit tests** to confirm your subcomponents work and to profile its performance. For example, if there are 5 I/O modules, we expect there to be 5 different projects, one for each module. We suggest creating projects like those found in Valvanoware to test the low-level software drivers, and that the unit test tests all the functionality of your driver to make sure the driver behavior is as expected. The following is an example testing plan for a DAC used to create audio
   1. Write the SSI/DAC software driver.
   2. Build the DAC circuit.
   3. Write interrupting software that outputs a sinewave.
   4. Verify the DAC output is a sine wave using the scope and spectrum analyzer.
   5. Build the speaker amplifier circuit.
   6. Verify the speaker outputs are sine waves using the scope and spectrum analyzer.
   7. Connect the speaker and verify the waveforms are still correct.

Note that because this is unit testing, we ask that you minimize the integration of other drivers and keep the tests as simple as possible. For example, the above testing plan would be a lot harder to debug if you’re also playing music from a music driver and not just a sine wave because there are more things to test.

1. **Verify that the unit tests meet** any performance or functionality tests as defined in the requirements document.

# Lab Checkout

For lab demo, we ask for the following:

1. Show the hardware on a solderless breadboard and attached to the Launchpad.

2. Demo low-level software working with each subcomponent of the hardware.

# TA and Professor Review

*Students shall submit their preliminary PCB to* *GitHub. See syllabus for due dates*

# Lab Report

## Deliverable 1

Updated Requirements Document including a Data flow graph and Call graph. Describe how responsibilities of team members are distributed.

## Deliverable 2

KiCad Schematic (power and enclosure will be performed in Lab 7B).

## Deliverable 3

Updated BOM. Record in your report the total cost estimation in the BOM. Additionally, record the total cost of only the components which count towards competition eligibility.

## Deliverable 4

Estimate the current consumption of each subcomponent in active and idle states, including microcontroller and I/O devices. **Note:** a system may draw more power than the sum of its components due to the regulators used to create the given voltage rails.

## Deliverable 5

Header and code files for all low-level software modules. Unit test code (main programs) for each module.

## Deliverable 6

Include results of unit testing.

## Deliverable 7 (5pts Extra Credit)

Have at least 1 member of your team complete 3-D printing or laser cutting training in the Texas Invention Works. There are sign-up sheets for training on the 0th floor of the Texas Invention Works. Provide some proof that you got the training in your report. This extra credit can be completed until lab 7A and will be applied to lab 10.

## Deliverable 8 (10pts Extra Credit)

Use a TM4C123 chip in your design in addition to the launchpad. You will need to add the component to the PCB like in Lab 6. You will need to ensure that it is Launchpad redundant (e.g. if the chip fails, the LaunchPad can be plugged in and used). One way to make it redundant is to add headers to the PCB where jumper can be run from the launchpad to the PCB. This extra credit can be completed until lab 7B and will be applied to lab 10.

# Hints

1. You must collect ALL parts (except battery, regulators, connector, and cables) while you are doing Lab 7A so you can be sure they function properly.
2. There are two types of LEDs you can get from the professor’s office. Low current red/yellow/green HLMP-D47xx LEDs can be connected directly to a microcontroller output using just a 680 ohm or 1 kohm resistor. The other colors and sizes that we have require 10 mA and will need an interface (like a 2N2222 and a 100ohm resistor.) You should test the LED/resistor circuit on a breadboard to make sure the brightness is acceptable.

## Other Stuff

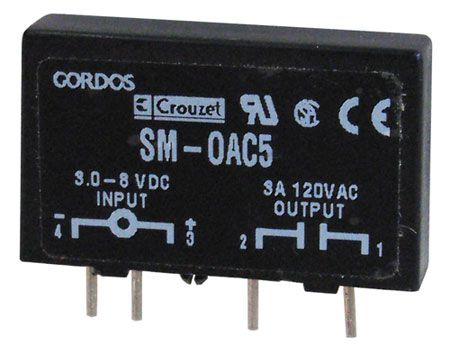
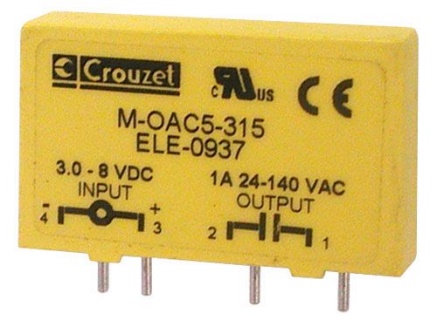
Web site references:

1. Enclosure manufacturers
   1. http://www.OKW.CO.UK/ OKW
   2. http://www.TEKOENCLOSURES.COM/ TECO
   3. http://www.PACTECENCLOSURES.COM/ PACTEC

Most box manufacturers will not ship samples to students.

1. General purpose suppliers
   1. http://www.SPARKFUN.COM Spark Fun - Transducers, Buttons, Displays, etc. Lots of cool stuff.
   2. http://www.allelectronics.com All Electronics - All sorts of random stuff
2. LCDs
   1. http://www.newhavendisplay.com/ New Haven Displays - LCD Manufacturer. Has a bunch of displays around $10
   2. http://www.varitronix.com/ Varitronix - LCD Manufacturer.
   3. http://www.crystalfontz.com/ Crystalfontz - good quality and price LCD's
   4. http://www.sparkfun.com/products/710 $20 Sparkfun LCD-00710 64 by 128 LCD
   5. http://www.circuit-ed.com/128x64-BLWH-TOUCHSCREEN-GLCD-P146.aspx $26 64by128 LCD touch screen
   6. http://www.sparkfun.com/commerce/product\_info.php?products\_id=8977 Touch Screen
3. Batteries
   1. http://www.powerstream.com/ PowerStream – Batteries
   2. http://www.batteryjunction.com/ Battery Junction - Li-Ion packs
4. Industrial suppliers
   1. http://www.digikey.com/ Digikey - Useful parametric search. Lots of standard components. This is the easiest way to find multiple manufacturers of something common like connectors
   2. http://www.mouser.com/ Mouser - Pretty much any standard chip you want is available here. Ships from Dallas, so usually faster than Digikey, but sometimes a bit pricier.
   3. http://www.avnet.com/ Avnet - Some higher-end stuff and hard to get chips can be found here.
   4. http://www.newark.com/ Newark - Similar to Avnet

http://www.ladyada.net/library/procure/samples.html Getting Samples - How to get free samples.

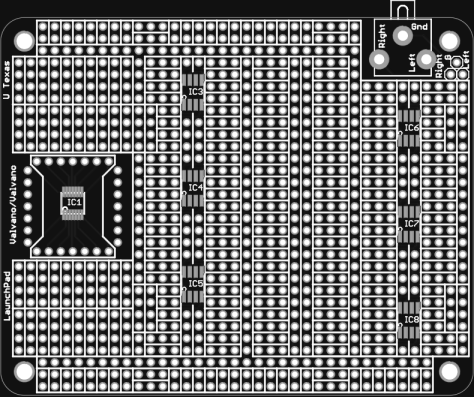


https://www.crouzet.com/products/solid-state-relays/

18-bit color, 128\*160, 1.8" TFT LCD display, Sitronix ST7735R, www.Adafruit.com part number 358, $19.96



If a project is so much more complicated than the typical ECE445L project that they need two boards, please see your professor. One option for creating more than one device is to make one with the PCB layout and make the second one by soldering parts onto an ECE319K project board.



ECE319K Project Board.

Lab 7 hint: If you want to learn much more about PCB design (details beyond what is needed for ECE445L, but awesome if you wish to create PCBs professionally), Matthew Yu created two playlists with many hours of educational material.

https://www.youtube.com/playlist?list=PLqUBXn7oPxmxXZYo1X\_91ucJfMEAJdrzt (PCB Design)

https://www.youtube.com/playlist?list=PLqUBXn7oPxmzVwcAnNQxI12CLg\_SvGdOF (PCB Backlog)