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 © 2021.

Team Size: 4

**Goals**

* To lay out and route a PCB board.
* To perform integration testing to verify combined interoperability of a system.
* To develop a test plan for the final product.

## Review

* Same as Lab 7.

**Starter files**

* Same as Lab 7.

## Background

Normally, one passes around the design cycle multiple times. However, in this class, Labs 7, 8 and 11 represent one pass around the design cycle. In this lab, you will build upon your prototype hardware and software implementations. The goal is to verify that your collection of chips, resistors, capacitors and connectors will operate as intended when you get to Lab 11. In Lab 7, you tested low-level hardware/software for all I/O interfaces. In Lab 8, you will layout the PCB and fit it with cables and connectors into an enclosure, as well as demonstrate that your system can work in tandem with each of its constituent components.

We note that this is a team effort, and proper allocation of work will get the most results in the shortest period of time with the least stress. As an example, if only one person is working on the PCB, get the others to write the drivers! Don’t all stare at their computer waiting for them to finish. Also, don’t be shy to ask team members, other teams, your TAs, or the professors for help (although you might want to be mindful, don’t ask them to do the work for you)!

## Preparation (do this before your lab period)

1. **Place all components in the PCB** area. **Do not start routing until completing the TA design review of your BRD.** The data sheets of many ICs offer suggested values and placements of capacitors, to improve performance. Make sure the **Snap to Grid** mode is active (experiment with different settings of the snap). Place all through-hole components on the top side. If possible, align all chips in the same direction. Configure the board so that most through-hole soldering occurs on the bottom side.

## Procedure (do during lab period)

1. **Consider how the PCB will fit** in the box. How will the PCB be mounted to the box? Consider the 3-D size of the Launchpad and LCD. How will you reach the test points for analog debugging?

You will place a male-male header on the PCB to simplify using the low-cost logic analyzer to digital pins. Connect strategic **digital** pins to the low-cost logic analyzer. Add test points at strategic points to assist in analog debugging. One simple way to make an analog test point is to place two 0.029 in. holes 0.1 in. apart. During Lab 11 you will solder a U wire into it. A second way is to make a 0.043 in. hole. During Lab 11 you will solder a test point into one hole (get from ECE lab checkout).

1. How does the operator reach the switches? How does the operator see the LEDs?
2. Can you place the LCD on the PCB such that no cable is required? How will the LCD be mounted?
3. **Verify that the parts** you collected in Lab 7 match the footprints on the PCB. Please procure or build the box for the project.
4. **Do not start routing until the TA has approved your parts placement. Finish PCB layout** using Eagle. The SCH-PCB file combination must pass all design rule checks and fit within 30in2.
   1. Add Top and Bottom Silk labels for your names, UTX2023xxx, TA’s initials, date, and purpose.
   2. Add Top and bottom silk labels that will assist you in construction and debugging,
   3. Avoid 90-degree turns, convert them to two 45 degree turns (enable the Miter option),
   4. Avoid loops (circles in the PCB traces) because they are a source of noise.
5. **Measure your board dimensions**. To view your X and Y dimensions, click on your board outline and execute the info command. You will get the following popup:

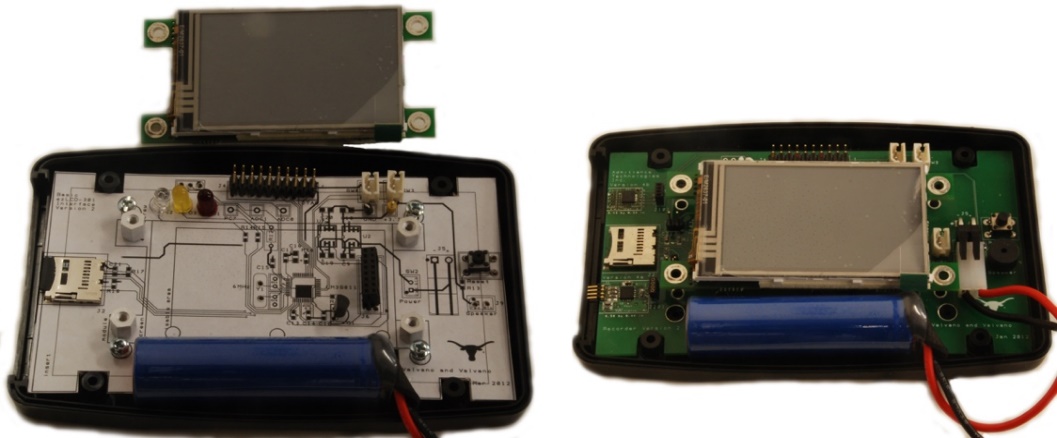
Table

Description automatically generated

*Figure 8.1. Process for determining board size.*

5000 is the X-dimension and 5000 is the Y-dimension. Note: the values are in “mils”, so 5000 is 5.000”. The total area must be less than or equal to 30 sq. inches. Make sure you have Your names, Project name, **Board Part Number** on your board in silk screen. The TAs will have to compile all this information and all your files so we can send it off for manufacturing. If you have questions or aren't sure how to make sure you deliver the files in the correct way, ask before the deadline. Any fixes we must do to your files/naming/anything after the deadline will result in 10 points being deducted. This includes missing board outlines and deleting internal routes (internal routing is NOT ALLOWED, use pads/vias to make holes.) **If you don't follow these directions, you will lose 10 points on Lab 8.**

1. Use bottom silk layer to help in soldering or testing. Go through the PCB ordering process detailed in Lab 6 and verify that the board can be ordered for around $10 plus shipping. Do not order the board yourself.
2. **Print out the PCB on paper** and glue the two pieces together. Punch and drill holes to place components on the simulated “PCB”. Verify that all components fit in the spaces allocated for them. See Figure 8.2.

**

*Figure 8.2. Example mockup of layout using paper printout glued on cardboard (LM3S811 into PacTec box).*

1. **Take the simulated PCB and place it inside the enclosure** with all components on the simulated PCB. Ensure that you can close the lid. This will ensure that the PCB is able to fit inside the enclosure in 3D.
2. Given the BoM you developed in Lab 7 (and have probably updated by now), **estimate the cost of the entire system** including parts given or lent to you, parts that you already own, PCB cost, Launchpad Cost, and any parts you purchased.
3. Upload your “Design Block” (i.e., PCB and SCH) **file to Canvas by the deadline**. This is a hard deadline for PCB submission. Specifically:
   1. Get an xxx part number from your TA (**UTX-2023Sxxx**).
   2. Place the part number on the Top Silk of the PCB

Save your SCH and PCB files to a “Design Block” using the **UTX-2023Sxxx** name (from part a.)

Graphical user interface, text, application

Description automatically generated

Figure 8.3

* 1. Your schematic must be ERC clean, and the PCB must be DRC clean. We do not have time to fix your schematics or layout.
  2. If you do not finish by this time, you must pay the manufacturing + shipping costs.

1. **Write and verify integration tests** to confirm that your subcomponents work with each other without unintended side effects. For example, you may link up switches with Wi-Fi, and pressing a switch sends a packet to Blynk. An example of a bug being discovered during integration testing is when two subcomponents might run on the same interrupt priority, and one subcomponent thread starves the other, causing large jitter. You may determine the number and size of your integration tests, as long as you have good code coverage of the system.
2. **Develop a test plan** for your PCB. This needs to be a list of 4 or 5 steps, detailing the bottom-up construction and test procedures. Each step will include hardware to be built, software to be written, debugging tools to be used, and expected results to be collected. As an example:
   1. Assemble the power regulation circuit. Plug in a supply (USB, power supply, battery, etc.), and verify that the regulators output the correct voltage.
   2. Assemble the LEDs and microcontroller. Flash a heartbeat program on the microcontroller and observe whether the LEDs Blynk appropriately.
   3. Assemble subcomponent A. Run the associated unit test to verify it works as expected (you should know what the outputs are from Lab 7).

An example with a more complex PCB can be found here: <https://github.com/lhr-solar/Power-Generation-Sunscatter-PCB/blob/v3.4.0/Documentation/ASSEMBLY_AND_TESTING.md>. You don’t need to get this detailed, but long, thought out procedures like these keep the all-nighters away. Especially if you design your PCB to make testing easy (Design for test)

1. Finally, you may want to get ahead of the curve and **begin writing a main program** based on your data flow graph and software call graph (feel free to modify these as you see fit). This isn’t required until Lab 11, but getting a head start may mean you don’t have to deal with some obscure bug the night before competition, or that you get leeway to add more features!

## Demo

For the lab demo, we ask for the following:

1. Open your BRD file within Eagle and show that it passes all design rule checks, fits within 30in2.
2. Explain how the system will be powered. Even if you are using the TM4C Launchpad, you still are required to have a regulator. You should be able to explain how the regulator works.
3. Show any connectors, I/O devices, or other devices that will connect to the PCB. Explain how they will be powered and communicate with your system.
4. Show a printout of the PCB layout (top and bottom) taped to cardboard. Make sure it’s not mirrored… put it in your box with the Launchpad and any other parts to show that it fits.
5. Demonstrate that subcomponents of your system can work together without issue.
6. Explain your testing procedure for when the PCB arrives.
7. Show that the simulated PCB has correct sizing of the components and that it fits within the enclosure.
8. Discuss your top three worries about finishing Lab 11 on time.

## Deliverables (exact components of the lab report)

1. A summary of the lab and what the student expects to learn from it.
2. An updated requirement document.
3. A summary of the software and hardware that was created to complete the lab (keep the full software on GH, paste a screenshot/printout of HW in the report).
4. Any measurements taken by the student to verify the design. We like performance metrics 😊.
5. Discuss any interesting aspects of your design process (your methodology to placement/routing; neat design for test tips that you used in your board, bugs that came up that you had to fix).
6. Any extra credit components.

## Lab 8 grading

You will be graded on the quality of your prototype design. More specifically, you should make this Lab 8 prototype system as close to the final system as possible. You should be able to state where all your prototype design differs with the actual system and how you will account for this difference while doing your measurements.

1. **(5) Attendance**, graded by (1) presence (in person or otherwise) at lab prep AND demo days, as well as (2) equal contribution to lab completion.
2. **(15) Lab prep**, graded by the parts placement on the PCB.
3. **(35) Lab demo and Q/A**. Prototype demonstration, focusing on the quality of design; points will be deducted if you do not have 80% or more of the parts and connectors needed for the final project. Also, identifying top 3 worries - Not identifying a worry is a worry. The earlier you identify it, the better it is. So, think hard on it.
4. **(30) Software and/or hardware quality.** Software (or hardware quality) of the lab will be evaluated by focusing on the quality of the design and formatting of the code or hardware schematic/layout. This includes but is not limited to: consistent syntax and proper use of version control, a clear flow of logic or proper handling of edge cases and user inputs, schematic/layout that passes ERC/DRC, organized component placement and proper documentation/readability, and understanding of placement of components or selection of components relative to each other (e.g., why XX capacitor and YY resistor for this low pass RC filter?).
5. **(15)** **Lab report**, only one member of the team must submit. Make sure the complete .SCH, .BRD files are uploaded to canvas on time by the deadline outlined in the syllabus (separate assignment).

## Bonus (Can be obtained either during Lab 7 or 8)

1. **(5 points)** Have at least 1 member of your team complete 3-D printing or laser cutting training in the Texas InventionWorks. There are sign-up sheets for training on the 0 floor of the Texas InventionWorks.
2. **(10 points)** Use a TM4C123 chip in your design in addition to the launchpad. Follow the instructions in processor implementation option 2.
3. **(10 points)** Validate at least two performance metrics of your system from the following list:
   * CPU utilization (thread profile) measured separately for each module.
   * Ping latency with Wi-Fi
   * Maximum execution time for ISRs
   * DAC or ADC sampling jitter
   * Signal to noise ratio (SNR)

You must also provide a short write up (~1 paragraph, 1-2 sentences ea.) for each of the metrics validated describing:

* + How you performed this measurement,
  + Does the measurements meet your system’s requirements,
  + And what sources of errors may affect your measurement.

## Precautions

1. Please download and use Eagle Design Rule Check Deck 2-layer JLCPCB because the built in one is too generic. Fix all errors in the DRC. You should see a window like this when you open DRC:

Graphical user interface, text, application

Description automatically generated

1. The components at the back of the board need to be mirrored. In Eagle, in "board view" (where you design the PCB), select the "i" tool (the tool that brings up the "properties" dialog) and click on the component you want on the other side of the board, check "mirrored.” This changes from the top layer to the bottom layer. You will see that the silk screen related items which are on tPlace, tNames, tValues will change to be on bPlace, bNames and bValues. You can also try turning off the visibility of the top layers (tPlace, tNames, etc.) to see whether Eagle thinks your connector is still on the top.
2. Please check if the distance between the wire and the wire, the wire and the component pad, the wire and the through-hole, the component pad and the through-hole, the through-hole and the through-hole are reasonable, and whether they meet the production requirements. Generally, the distance between the wires and the wires must be at least 4 mils. The minimum line spacing is also the line-to-line, line-to-pad distance. From a production point of view, the larger the better, the more common is 10 mils.
3. Please make sure your 3D dimensions are also reasonable. Some students have difficulties enclosing their embedded system because the TM4C bulges out of the enclosure.
4. JLCPCB only places top silk on layers 21-tPlace and 25-tNames (not 27-tValues). Similarly, JLCPCB only places bottom silk on layers 22-bPlace and 26-bNames (not 28-bValues). I suggest you toggle 27-tValues on and off in View->Layer settings to see if there is any text in 27-tValues you wish to move to 25-tNames. Similarly, I suggest you toggle 28-bValues on and off to see if there is any text in 28-bValues you wish to move to 26-bNames.

## Common mistakes that will delay the boards for the entire class.

1) Having traces/holes/spacings too small (do not change the design rules)

2) Having drill holes closer to each other (do not change the design rules)

3) Board name does not match Google Sheet submission

4) Having a board area larger than 30 in2

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

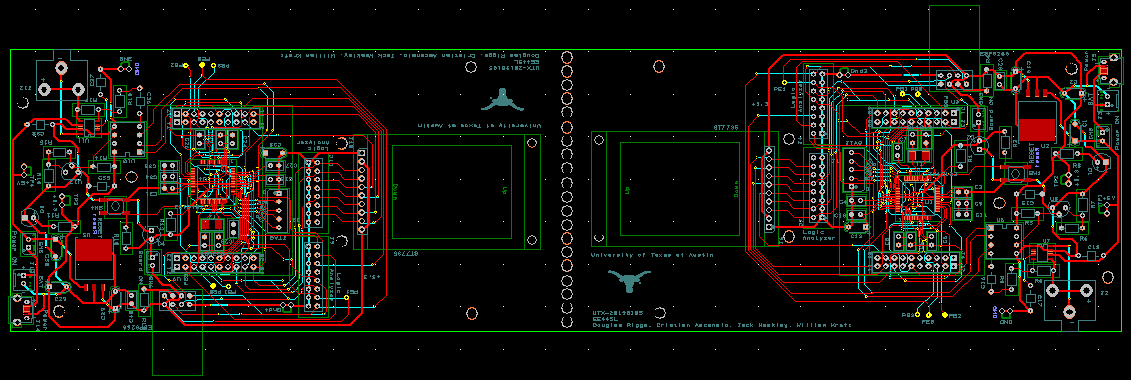
Description automatically generated

Graphical user interface, text, application, email

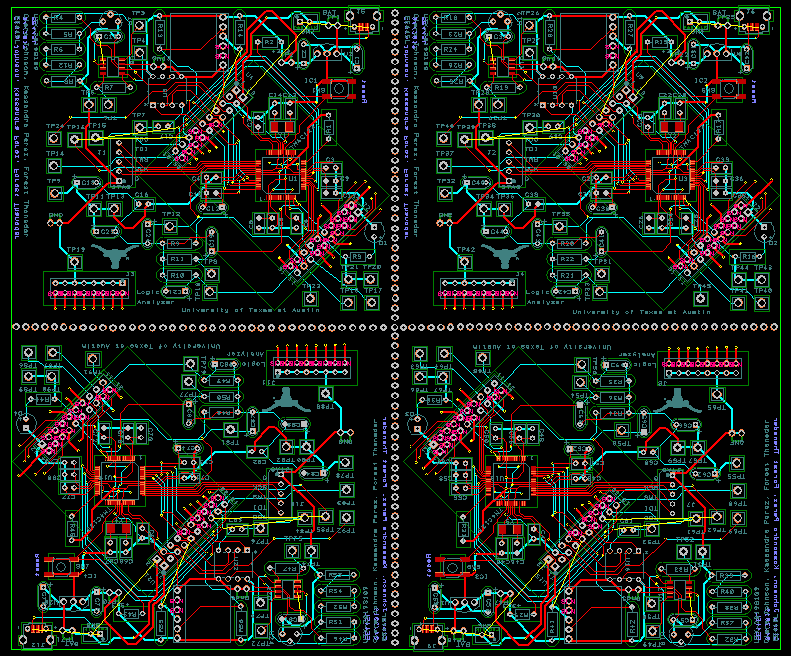
Description automatically generated

*Figure 8.4. Execute Edit->DesignRules and check the above settings.*

You may **not** create more than one board using one PCB file. This type of submission is not allowed. This semester we will get 5 copies of your design. The following two PCBs were not allowed and students who submitted these PCB files caused all boards for the entire class to be delayed by two days.



Creating multiple boards from one PCB file is NOT allowed.



Creating multiple boards from one PCB file is NOT allowed.