

Lab 8f Design and Layout of an Embedded System

This laboratory assignment accompanies the book, *Embedded Microcomputer Systems: Real Time Interfacing*, Second edition, by Jonathan W. Valvano, published by Thomson, copyright © 2006.

- Goals**
- To interface a memory to a microcomputer,
 - To study issues of power, clock, reset, and programming for an embedded system,
 - To layout a PCB board.

- Review**
- Valvano Chapter 9 on interfacing to your microcontroller,
 - Data sheets for your microcontroller,
 - Data sheets for your memory.

Starter files Lab8fprep.sch Lab8fprep.pcb

Background

You will use the CAD program **ExpressPCB** to layout a complete embedded system including reset, clock, power, memory and I/O. The software to activate the interface, hardware construction, and device testing will be performed in Labs 11 and 12. There will be an open house (like a science fair) to demonstrate your Lab 12 systems to the academic community. The design of the system must satisfy certain requirements (things it must do) within a set of constraints (limitations dictated by the realities of the project). The embedded system must do something useful. There are some options listed below, but you have flexibility to define exactly what it is to do.

Option 1. You can re-engineer one of the previous labs (e.g., stepper motor, alarm clock, thermometer, or music player) on this platform.

Option 2. You can make a hand-held game of pong. You will interface 8 LEDs to digital outputs of the microcontroller. The LEDs will be positioned in a straight line. At any given time exactly one of the LEDs is on, signifying the position of the ball. The ball is always moving to the left or to the right. There are two switches on each end of the LED line for the paddles. When the ball reaches the end, the player must hit the switch to volley the ball back in the other direction. Score can be displayed using additional LEDs.

Option 3. You can make a voice recorder. You will interface two switches and a DAC to the microcontroller. When you press the record switch, the analog voltage is sampled by the ADC and stored in external RAM memory. When you press the other switch, the previously recorded waveform is output using the DAC. The status of the system can be displayed using LEDs. If you choose this option, let your TA know so he can get you a RAM chip instead of an EEPROM. If you use an external RAM for data storage, your program will have to fit into the 512 bytes of EEPROM on the 6811, which means you'll probably be writing in assembly language.

Option 4. You can propose to your TA to design, implement, and test any 6811-based system demonstrating the educational objectives of this class. You will need specific approval from your TA for this option.

Requirements

- A microcontroller must run in expanded mode (with an external address and data bus),
- Memory must be interfaced to the address/data bus of the microcontroller,
- PCB layout of the system must include microcontroller, reset, clock, memory and I/O devices,
- There must be at least one input and one output,
- The final system (Lab 12) will be an actual device with chips soldered onto the PCB,
- The system should perform something useful similar to the above options.

Constraints

- The layout will be performed using **ExpressPCB**,
- Each **ExpressPCB** must be done using the \$51 Miniboard service (\$60.85 with shipping),
- Although 3 groups will combine to produce one shared PCB layout (one PCB file), there will be separate Lab 8 preparations (initial PCB and SCH files), there will be one Lab 8 mockup, one final SCH and one final PCB file, there will be one Lab 8 report, there will be separate Lab 11 and 12 software and reports,
- There will be a specific list of parts that we will be willing to give you for building this system,
- You must purchase any additional parts that you require.

The TAs will decide the team members. I prefer teams of 6, and I prefer each team of 6 has one TA. However, if the TA is willing to have a group size less than 6, or if two TAs are willing to share a group, then it is ok with me. No

one keeps this team of 6. You will do Lab 8 prep, and all of Labs 9,11,12 in your usual groups of 2. The group of 6 is a one time event just for the Lab 8 demo and report (the mockup and the PCB file).

Preparation (do this before your lab period)

Begin by drawing a circuit diagram of the entire embedded system using any available CAD software. One possibility is to use the **ExpressSCH** CAD drawing program (see starter file **Lab8fprep.sch**). Label all pin numbers on every connector and chip. Search for details on how to generate an appropriate reset signal in the data sheets of the microcontroller. You can also search for existing circuit designs based on the same microcontroller, which are commercially available (e.g., www.technologicalarts.com or <http://www.axman.com/>). Next, layout the circuit using the **ExpressPCB** program (see starter file **Lab8fprep.pcb**). Follow the layout advice for the crystal in the data sheets of the microcontroller. Please read the instructions included in **ExpressPCB** help menu. View the “Tips for Making PC Boards” page at <http://www.expresspcb.com>.

Add labels for your initials, your TA’s initials, the date, and the purpose of the board,

Link the PCB file to your SCH file (blue dots will appear helping you see what to connect to what)

Place all components on the top side,

If possible align all chips in the same direction,

Configure the board so that all soldering occurs on the bottom side,

Add labeling to the top side to assist in construction and debugging,

Add test points at strategic points to assist in debugging (e.g., power, **E**, **/CE**, **/OE**, **/WE**, and ground),

 Either by placing two holes 0.1 in apart then soldering a U wire into it,

 or by making a 0.090 in pad with 0.043 in hole then solder a test point into the one hole

Each IC should have a bypass capacitor, placed as close to the chip as possible,

All components need labels (e.g., U1 R1 C1 J1), shown both on the board and the circuit diagram,

Avoid 90-degree turns, convert them to two 45 degree turns,

I suggest that you do NOT cover the unused area with a ground plane or a power plane.

One way to make it all fit is to go left-right on one side and up-down on the other side

There is a limit of 350 holes

Periodically execute **Layout->ComputeBoardCost** to verify the board still costs \$60.85 to make

Every lab group of two students will produce an independent initial circuit diagram and layout as part of the preparation. You must select the MiniBoard service, which creates three identical 3.8 by 2.5 inch boards for \$51 plus \$9.85 shipping. Add features such as test points and labels that will make it easy to test the hardware. Please print out this initial layout and turn it in along with the regular circuit diagram.

Write a one-page proposal to the TA describing exactly what your final system in Lab 12 will do. In other words, start with one of the above four *options* and add specific details on how it will work.

Procedure (do this during your lab period)

The lab groups will be combined so that three groups form a layout team. Discuss with your TA and the other layout team members about the proposed operation of the final Lab 12 system. The layout team will meet and integrate the three individual layouts into a single final layout. It is possible for one layout to solve similar, but not identical problems. Again using the simple layout program provided by **ExpressPCB**, create the final layout for the layout team. Please print out this final layout and turn it in to the TA for approval and manufacture.

You should print both sides of the PCB layout (without the mask). See Figure 8.1. Create a mirror image of the bottom layer and glue/attach the two pieces of paper to Styrofoam, cardboard, or wood. You will have to punch or drill holes in order to place components on this simulated “PCB”. Create a complete 3-D mockup of the system placing the actual components on this paper/cardboard “PCB”. See Figure 8.2. 3-D spacing will be critical if the system will be placed in a box. **ExpressPCB** will make three identical PC boards, one for each group of 2. See Figure 8.3.

In Lab 11, you will build and test the microcontroller/memory interface, and in Lab 12 you will design, implement and test the complete mechanical/electrical/software system. See Figure 8.4. At this point, you should collect all the devices that will be required to finish the Labs 11 and 12. When the PCB boards come in, we will be providing you with the MC68HC11E1CFN3, MC34164-005, 74HC573, 74HC138, and 28C64BL-12. You should decide if the system will be powered via the wall wart or by a 9V battery. You should decide whether or not the system will be placed into an enclosure.

Deliverables (exact components of the lab report)

A) Objectives (1/2 page maximum)

B) Hardware Design

Regular circuit diagram (must be prepared on the computer using any CAD program)
 PCB layout and three printouts (top, bottom and combined)

C) Software Design

Write a product description defining what your embedded system does (different for each group).

D) Measurement Data (none)

E) Analysis and Discussion (none)

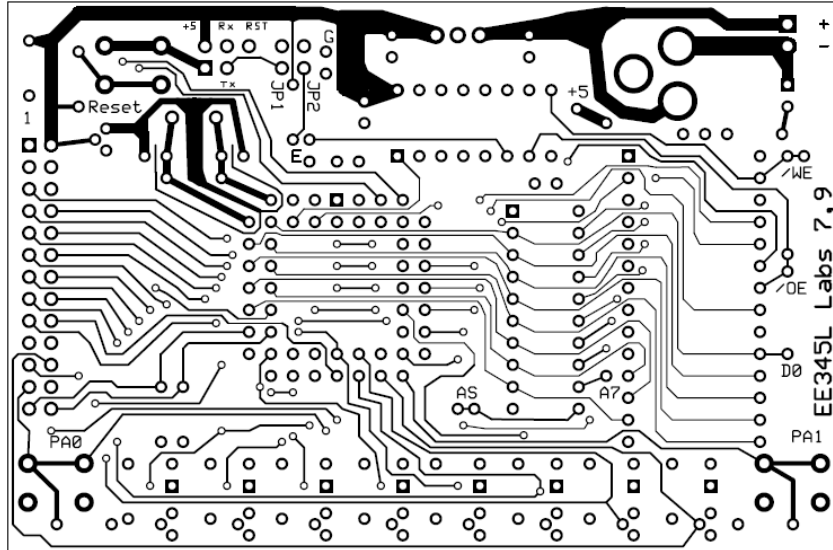


Figure 8.1. Example ExpressPCB printout of top copper layer (notice the test points for ground, power, E, AS, /WE, /OE, D0 and A7).

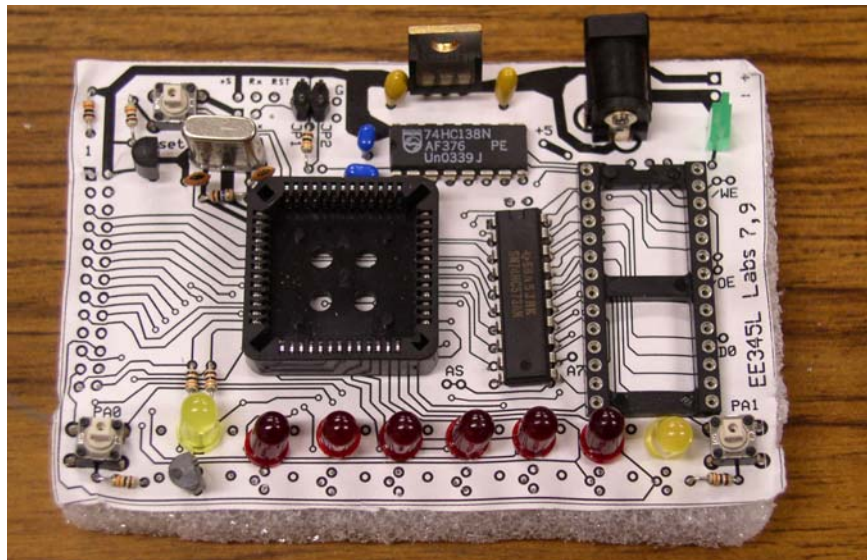


Figure 8.2. Example mockup of layout using Styrofoam.

Checkout (show this to the TA)

Show the 3-D mockup of the system to your TA. Explain how the system will be powered. Can you estimate the current required by the system? Discuss the testing features of your design. If you plan to incorporate an enclosure, show how the connectors, I/O devices, and PCB fit into the box.

Triple-check everything. It is much easier to fix a mistake before PCB manufacturing than after.

Hints:

1) RAM memory will function properly with any connection between the RAM and computer address pins, as long as each of the computer address pins is connected to a separate RAM address pin. This is not true for ROM

interfaces, which will require exact A0 to A0, etc. However, you are free to map any 6811 A6-A12 into any EEPROM A6 to A12, but you must map A5-A0 exactly.

- 2) Your EEPROM memory will function properly with any connection between the EEPROM and computer data pins, as long as each of the computer data pins is connected to a separate EEPROM data pin. This is not true for externally programmed ROM interfaces, which will require exact D0 to D0, etc.
- 3) The Miniboard service does not produce the silk layer, but you should use it to help document the design.
- 4) \overline{CE} / \overline{CE} and \overline{CE}^* are three equivalent ways to specify the signal is negative logic
- 5) The 5-pin connector, shown as J4, will be used to download programs into the system in Labs 11 and 12.

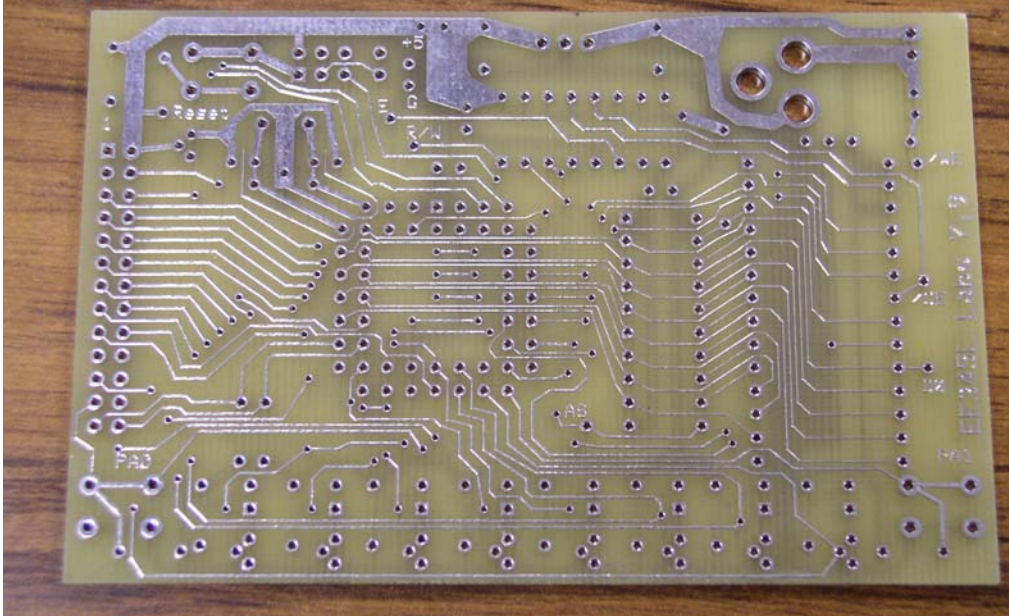


Figure 8.3. Example PCB made by ExpressPCB (used in Lab 11).

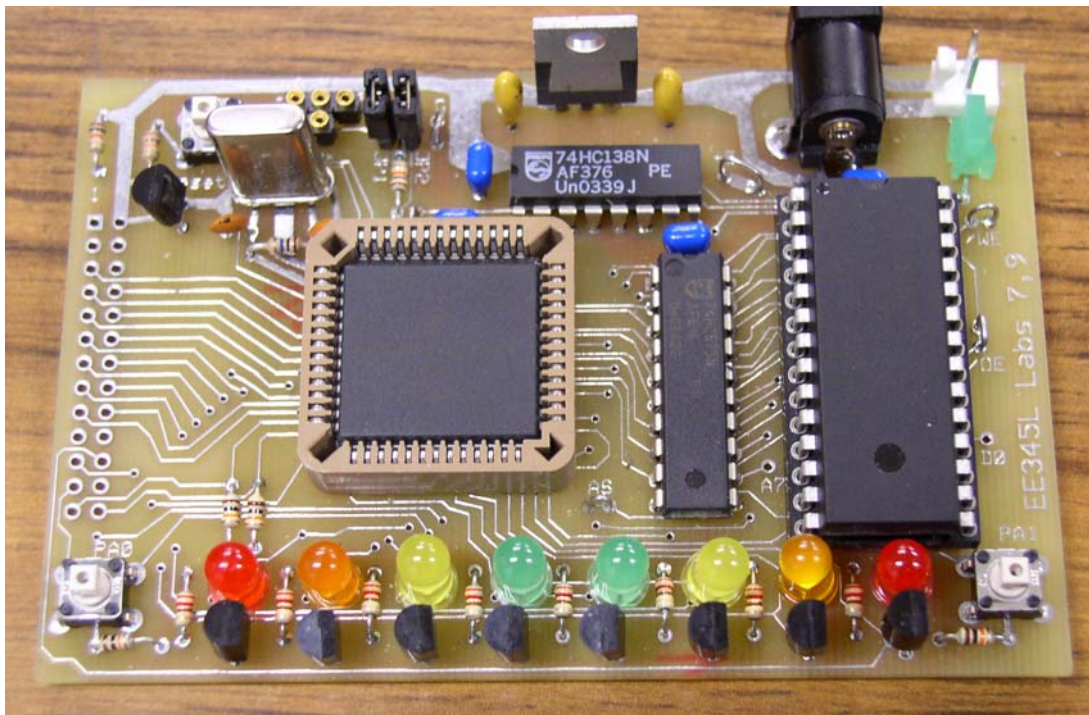


Figure 8.4. Example embedded system (created in Lab 12).

6) Parts distribution will occur in two phases. The first phase includes:

- 1- 78M05 0.5 amp +5V regulator [Jameco 192233] (78m05.pdf) \$0.19
- 1- JACK DC Power MALE 2.1mm [Jameco 101179] (2.1mmJack.pdf) \$0.55
- 3- resistor 1/8W 5% 2.7K OHM [Jameco 108062] \$0.0069
- 6- resistor 1/8W 5% 10K OHM [Jameco 108126] \$0.0069
- 1- resistor 1/8W 5% 10M OHM [Jameco 108274] \$0.0069
- 2- CAP CERM DISC 22pF 100V 5% [Jameco 15405, Digikey P4841-ND] \$0.11
- 1- 52 pin PLCC socket [Jameco 72442] (Plcc.pdf) \$0.99
- 1- 28 pin machine-pin IC socket \$0.64
- 1- 8 MHz crystal [Jameco P/N 14728]
- 2- 0.47 uF Tantalum 35V 10% caps [Jameco 332137] (0.1in footprint) (tantalumCap.pdf) \$0.23
- 4- 0.1uF ceramic capacitors, bypass cap for each chip, (0.1in footprint) \$0.05
- 3- B3F tactile push button switch (B3f-1059.pdf) \$0.17
- 3- 1.6V 1mA HLMP-D150 LED (LEDHLMP-D150.pdf Digikey 516-1323-ND) \$0.173
(I have yellow and green low current LEDs as well)
- 1- 5 pin, 3" jumpers with matching pcb mount, All Electronics CON-55, \$1.35
- 1- 2-pin header with 2-pin jumper

You can have additional switches, PN2222A, 10k, 220 ohm, connectors, and LEDs after the boards come in, and you are ready to solder. The statically sensitive parts 74HC138, 74HC573, MC34164, MC68HC11E1CFN2, and 8k EEPROM will be given in Phase 2.

- 1- MC68HC11E1CFN3 [Jameco#:248604] (M68hc11e.pdf, 6811techref.pdf) \$6.55
- 1- 28C64BL-12 120ns 8K by 8 EEPROM (28C64B.pdf) \$2.00
- 1- MC34164-005 reset circuit (Mc34164.pdf, MC34164b.pdf) \$0.54
- 1- 74HC573 [Jameco 46076] (74hc573.pdf) \$0.31
- 1- 74HC138 [Jameco 45330] (74hc138.pdf) \$0.28

7) There are two types of LEDs you can have. Low current red/yellow/green HLMP-D150 LEDs can be connected directly to a 6811 output using just a 1 k Ω to 2.7 k Ω resistor. The other colors and sizes that I have require 10 mA and will need an interface (like a 2N2222 and a 220 Ω resistor.) You should test the LED/resistor circuit on a breadboard to make sure the brightness is acceptable.

8) My advice is to do a little bit of Lab 8, then have someone check it. **DO NOT DO THE COMPLETE DESIGN SCH/PCB THEN GET IT CHECKED.** To have Lab 8 checked, you can contact your TA, or email files to Valvano. We will evaluate your

- Address decoder equation
- Design equations for CE/ OE/ WE/
- ExpressSCH files for 6811/memory design errors
- ExpressSCH files for gross design errors in the I/O interface
- ExpressPCB files for style (line width, corners)

You should have a TA or Valvano certify the 6811/memory interface is valid before you begin PCB layout. There are at least four possible valid solutions for the 6811/memory interface (two ways to turn it off, and two ways to synchronize). Neither the TAs nor Valvano have the time to verify accuracy of your PCB. We can not also certify all the I/O circuits will work, but we can check to make sure you are using 6811 input/output pins appropriately (e.g., some 6811 pins are input only and others are output only).

9) The datasheets for the components used in this lab can be found on the datasheets page <http://users.ece.utexas.edu/~valvano/Datasheets/>

10) There will be a "Science Fair"-like public demonstration for Lab 12. We will present special awards to the team of two with the best design. The preliminary round will be judged by your TA, and the final round will be judged by an independent panel (e.g., Daryl Goodnight, Paul Landers, and Perry Durkee etc.) Some students will plan to put extra electronics off the PCB, because it doesn't all fit on the PCB. If you do have off-board electronics, then you will need a connector or something to create the bridge. You can get good grades in Labs 8, 11, 12 with off board electronics, but you will not be eligible to win "best design". In particular, your grade depends on if the required tasks are completed on-time, if your 6811/memory interface works, and if your eventual project (I/O software computer) works. However to win "best design" you will need to meet the following restrictions:

- runs using the 6811 and external EEPROM/RAM memory on the PCB

all electronics (resistors, capacitors, ICs) are on the Miniboard PCB
your team of 2 spends less than \$30 on extra components (which are readily available to all students).

It is possible to have external I/O devices, like speakers, switches, thermistor, and/or an LCD off the PCB and still win "best design. In particular, you can use a LCD that you check out from the second floor, but I do not have LCDs to give away (i.e., you will have to give the LCD back to the second floor). If you want additional components that I do have (LEDs, switches, thermistors, 74HC00, 74HC595, connectors, resistors, capacitors, speakers, boxes) you need to come to my office and show me your SCH file, circling or listing the needed components. Components from Valvano's office do not count against your maximum of \$30 additional components.

- 11) The starter files SCH and PCB show J4 (straight 5-pin jack with multicolor wires) as
- pin 1 ground
 - pin 2 PT2=PD1
 - pin 3 PT3=PD0
 - pin 4 PT4=/reset
 - pin 5 power

The pin numbers do not matter. To program the 6811 you will need to connect

- 6811 power (+5) to 6812 power (+5V)
- 6811 ground to 6812 ground
- 6811 PD1 to 6812 PT2
- 6811 PD0 to 6812 PT3
- 6811 /reset to 6812 PT4

- 12) The footprint for the MC34164-005 is identical to the PN2222 (TO226AA), and is shown as Q1 on both starter files, SCH and PCB. Many students mistakenly switch the PN2222 NPN transistor and the MC34164-005 power-on reset circuit.

- 13) If you plan to put the system in a box, you should create a 3-D mock-up of the system including the box during Lab 8. Starting to think about squeezing all the components into the little box once you get to Lab 12 will be difficult. Placing components in the proper place on the PCB during Lab 8 will greatly simplify the box-building process.

- 14) ExpressPCB has a feature that allows for ground planes. I **STRONGLY SUGGEST** you do NOT use this feature. Ground planes are useful for high frequency and/or low noise systems. The ground plane makes it much harder to visually see what wire connects to what pin, it makes it much harder to cut/add traces in Lab 11 to fix mistakes, and it makes it harder to create good solder joints without using a high-temperature soldering iron.

- 15) One common mistake new PCB layout designers make is placing two wires too close to each other. Subsequently, during fabrication, these two wires may become shorted because of the tolerances of the manufacturing process. A general rule of thumb is that you should allow enough space between two wires to fit the smallest allowable trace between them. For this PCB manufacturer, separate all traces by at least 0.007 inch.

- 16) Here are a few tips when things get tight. Notice in Figure 8.1 and 8.3 that two traces go between two 6811 pins. This was performed using 0.007 in traces and temporarily disabling snap to feature. For example, assume the 6811 pins are at y=1.30 and 1.40 inches. You can double-click one end of the 0.007 trace and manually set the y location to 1.34. Double click the other end and set it also to 1.34. For the second trace, double-click its ends and manual set their y location to 1.36. A second tip when things get crowded is to shrink the size of the vias. A via is a connection from top to bottom that will not be holding any physical component like a resistor or chip. The default size of the via is large and you can double click the via and reduce it to 0.0031 in via with 0.0014 in hole. It warns the via may be filled, but that is OK (do not reduce the size of any hole that will be accepting a physical component.)