

Tina Shahbaz-Khan
998134198
EEPower
EEC 134

Application Note

There are many stages involved in an engineering project. The first step is to understand the requirements and the specifications. Then, the members of the team need to think of a design. This design can be simulated on a computer to understand its feasibility and flaws. Then, if it is possible, a prototype (either breadboard or protoboard) of the design should be made for the first round of testing. After modifying the prototype, the design can be transformed into a form of PCB for further testing and improvements. Also, in engineering projects, team members should always have backup plans (just in case their original design doesn't work). Each of these steps is very important in engineering projects, but in this app note, the focus is on making the prototype board and PCBs (including making the footprints).

Prototype Board (Gain Stages and Active LPF)

When doing the design on a breadboard, it is easy for the connections to get loose and this affects the performance of the circuit. For our project, we faced this problem when we were wiring the baseband. To avoid these loose connections and also to make our design lighter smaller, I made a prototype board using the following schematic:

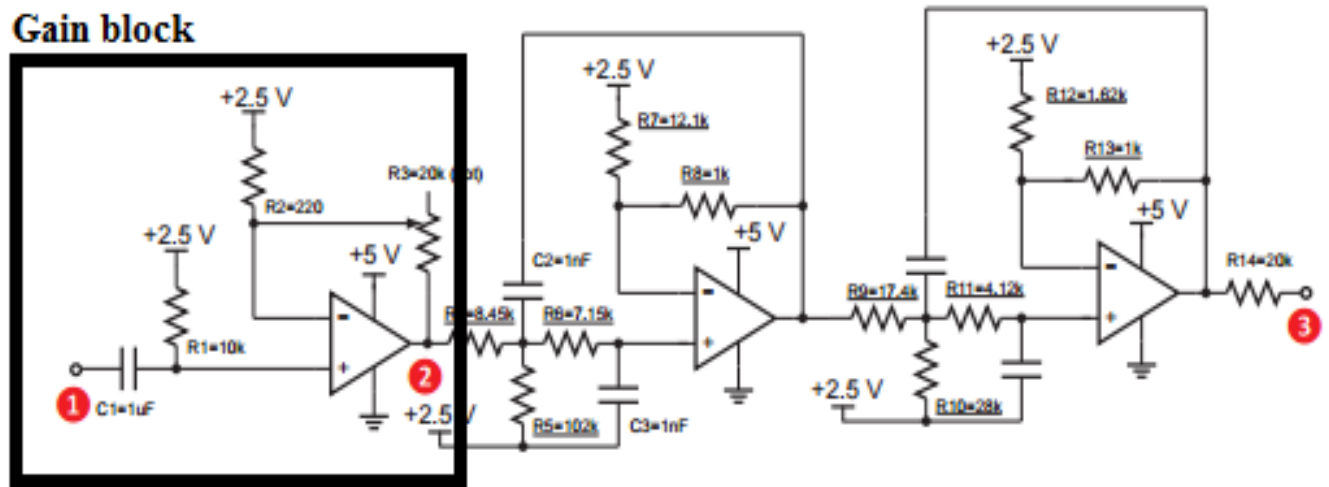


Figure 1: Schematic of Gain stage and LPF

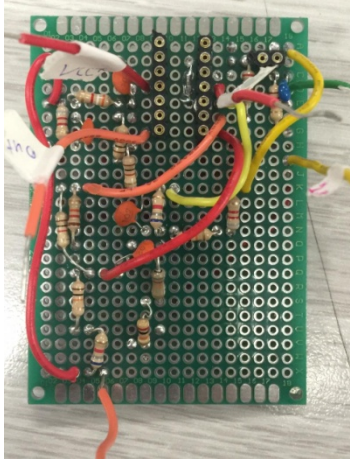


Figure 2: The picture above is the protoboard used in our radar system (5cm x 7cm, gain = 35, works up to ~50m).

Making a protoboard has a lot of benefits before making the PCB. With making this prototype, the weight of our project reduced significantly. Also, testing this board was easier than the breadboard version. The protoboard is stronger and steadier than the breadboard version. At the end, making the protoboard takes less time. This might seem trivial, but it really helped us during the testing process (of our radar system).

When making the protoboard, one should always try to debug the circuit before soldering. Also, one should always test the circuit part by part before soldering the whole project. For instance, in our baseband board, I assembled the first gain stage first. I checked the circuit, and then I soldered just the gain stage. After soldering the gain stage, I tested the circuit to see if it is working using the oscilloscope to see if the gain is reasonable. Also, one should always check to see if the solder is making a contact with the copper pads or not by making sure there is enough solder on the pads (avoid air connections). When soldering, one should make sure that the legs of the components are not touching each other (shorting the components).

Printed Circuit Board Design

After making and testing the project on a breadboard and a protoboard, the next step is to make the PCBs. PCBs can have one or more layers (copper on dielectric substrate). The benefits of making the PCB are to reduce the size and the weight, and to make the project more immune to unwanted issues (such as noise or loose connections). PCBs also look cleaner because all the wiring is engraved on the board, and it is easier to debug and test them. To make the PCBs, one needs to first design the board on a computer using a PCB designer software.

There are a lot of software tools available to do the PCB design. For our project, we didn't need to buy the expensive software tools. There were two tools introduced in our class. These tools were EAGLE CAD and KiCAD. I had experience working with EAGLE in my previous classes.

I learned how to use this software using a very useful tutorial online (by Jeremy Blum). EAGLE is really easy to use and has a user friendly interface. Because EAGLE is such a popular program, some PCB manufacturers, including OSH Park, accept EAGLE files instead of making you convert to Gerber files. This made the process of ordering the boards easier and this was useful when we were under a time pressure. For our project, I decided to just use the software that I was the most comfortable with (EAGLE).

One thing that helps a lot before working on the PCB is to have a clear schematic of the design. This way, one can follow that schematic to do the PCB.

For our senior design, we always had backup plans just in case our main PCBs didn't work. For instance, I was responsible for making small PCB modules for the gain stage and LPF circuits as our backup plan (just in case our main PCB didn't work or just in case we needed to add one more gain stage). To do this I used EAGLE to design each module.

Gain block

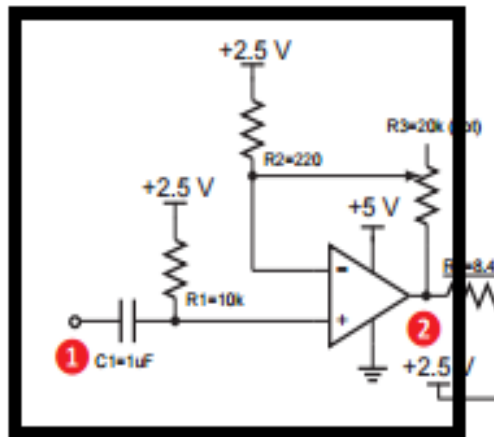


Figure 3: The schematic of the gain stage module

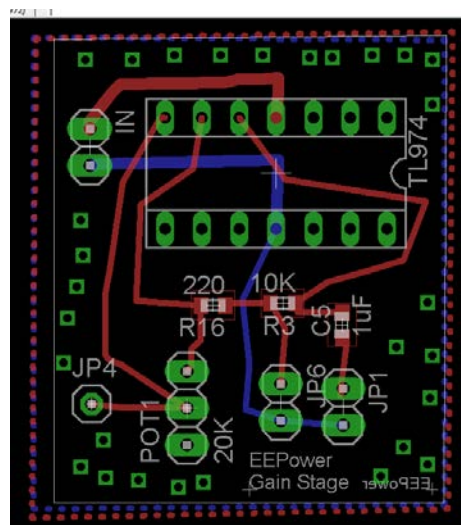


Figure 4: PCB layout Module for the gain stage (26.90x32.00 mm)

The schematic design is when components are put together to make the circuit. In the schematic window on EAGLE, one can search for the components and place them on the blank spaces in the window. After placing the components, the components can be connected by wiring them. In this stage one should also name the components and give specific values to each of them. A good suggestion is to name the components the same as your provided schematic and in an organized way (R1, R2, C1, C2, etc.). A good practice is to keep the wiring process clean in the schematic window. This is helpful when one goes back to his or her schematic to check his/her design.

One thing to note on EAGLE is that the software can't recognize a junction on the schematic window. The user needs to use the junction tool to place a junction on the connection.

Sometimes the user needs to download a new library to have access to more components on EAGLE. One library that is very useful and one can download for free is the adafruit.lbr that has the footprint for our op-amp (TL974) and JP or jumper that is used as test points.

Also, one good habit in PCB design is to have test points on the circuit. If the whole circuit is divided into smaller sections, a test point can be placed in between each of the smaller sections. This way, the board can be tested easily using these test points.

As the user is working on the schematic, EAGLE, automatically creates a layout window and places the components on the window. This way, the user can go to the layout window and start working on the PCB layout. (The components are not going to places in an organized fashion when the user is working on the schematic).

On the layout window, there is a white square. This white square/box is the board area and all the components must be placed in it. The size of the white box can be changed (doesn't need to be a square necessarily.) For our purposes, I only needed to design a simple low frequency two layer PCB. In placing the components, one should pay attention to the area that each of the components occupy.

The user needs to drag and rotate all the components before wiring them. EAGLE offers an auto wiring feature to connect the components for the user. The auto wiring feature is not recommended in our case because there is higher chance the wires fall very close to each other, and this causes errors in the final step when EAGLE does a design check.

In wiring the components, one should pay attention to the thickness of the wires (traces) to avoid overheating the circuit (current overflow). In our design, the traces' thickness was chosen to be 16 and for the traces that would power our op-amp, we chose the trace thickness to be 30. When wiring the components, one shouldn't connect traces with angles sharper than 90 degrees (so the traces can be fabricated correctly). In order to shorten trace length one should place the power supply close to all active components. Also, when the components that should be connected to each other are close, the traces will be shorter.

The reason the board is two layered is because we want to make sure there are no cross paths between the traces (noise and losses). Therefore, the two layered board is used to connect the bottom layer to the top layer (with using vias). For instance the blue traces in figures 4 and 6 are the traces that are on the bottom layer (ideally the blue traces are used for ground). Finally for the design of the two layer baseband board one should flood both the top and bottom with a ground plane, and connect the two planes with a large number of vias.

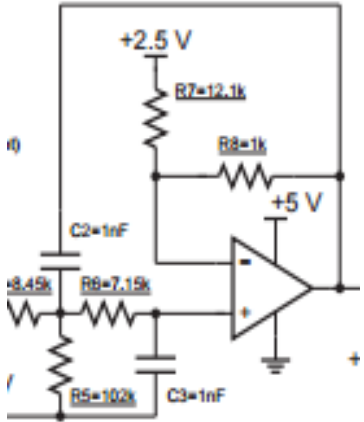


Figure 5: This is the schematic of LPF (two of these schematics with different resistor values)

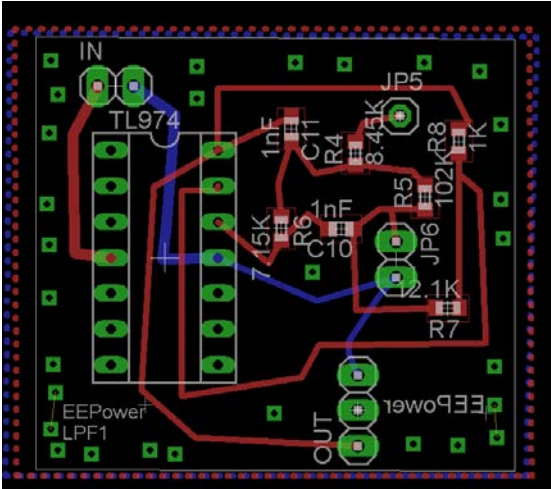


Figure 6: PCB layout for the LPF (34.09 x 30.94 mm)

After finishing up the PCB design (both the schematic and the layout). One should make sure the all the parameters, such as the trace widths and the component separations all match with the components' datasheet. The datasheets give us a lot of information regarding the components that can be used when one does the design check. After running the design check, EAGLE lists all the errors involved with the design. By clicking on the errors, the software takes you to where the problem is. After, all the errors are fixed, the PCB design is ready to be sent for fabrication.

If the manufacturer does not accept EAGLE files, such as Bay Area Circuits, then it must be converted to a Gerber file. When one receives his/her PCB it is important to check and ensure that everything was manufactured correctly. Osh Park (another PCB manufacturer) in contrast accepts EAGLE files, and the prices vary for each board.

Making the Footprints for RF Components on Cadence

For our radar system, I also helped with making the footprints of the RF components. Since our teammate was using Cadence, I wanted to help him with making the footprints so he wouldn't spend a lot of time on making the footprints. This way, he would focus more on the RF design.

There was a file (provided in the Appendix section) that I used to make the footprints. This file explained in detail how to make the footprint for a component. The main steps in making the footprint are: 1) creating the pads, 2. adding the pads to the package, 3. Adding bound_top layer, 4) adding silkscreen_top layer, 5) adding assembly_top_layer and 6) adding labels. In creating the footprints one should pay a lot of attention to the datasheet's first few pages where it explains in detail the dimension and spacing of the pins and connections of the components. After making the pads, they should be placed in an organized way to make the footprint of the components. This process can take a long time since the measurements provided by each of the datasheets vary, so one should pay attention while labeling the pins to avoid any kind of confusion for the PCB designer.

When designing the footprint, there are some critical things to pay attention to. First, one should make sure that the units used are consistent with the units used on Cadence. Land patterns have their dimensions in either mil or mm. A mil is 1/1000 inch, (or 1 mil is .0254 mm). When reading the dimensions one should read the dimensions that correspond to the grid settings in Cadence. Another important parameter to look for is the orientation of the component in the land pattern. Sometimes they will be given as if one is looking down from the top of the component, while the others give the view looking from the bottom. It is important to make sure that the orientation is consistent with the pin labeling.

The following figures show some of the footprints I made on Cadence:

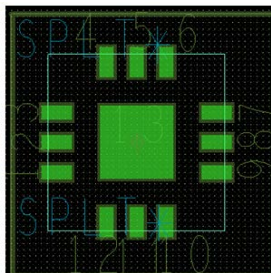


Figure 7: GP2X+ Footprint (Power Splitter)

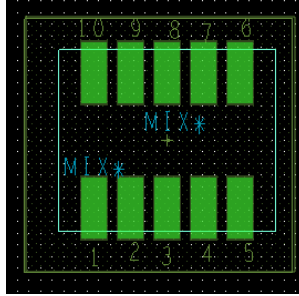


Figure 8: MAC60 Footprint (Mixer)

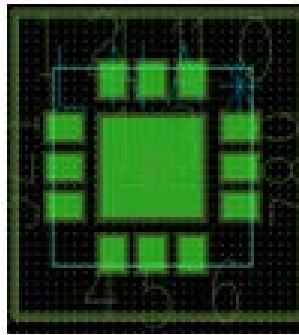


Figure 8: MAMF-010614 (LNA)

Appendix

- The following links helped me a lot when I was working on my PCBs:

<https://www.youtube.com/watch?v=1AXwjZoyNno>

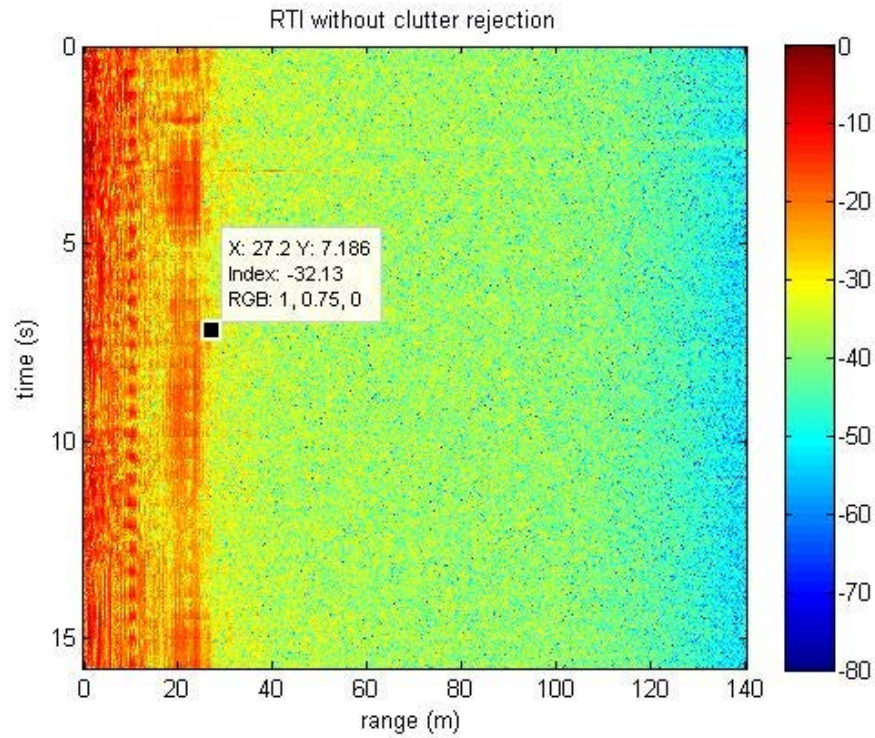
http://www.cadsoftusa.com/fileadmin/journalist/Documents/V7.3_tutorial_en.pdf

https://www.youtube.com/watch?v=oGfvVkD_3IU

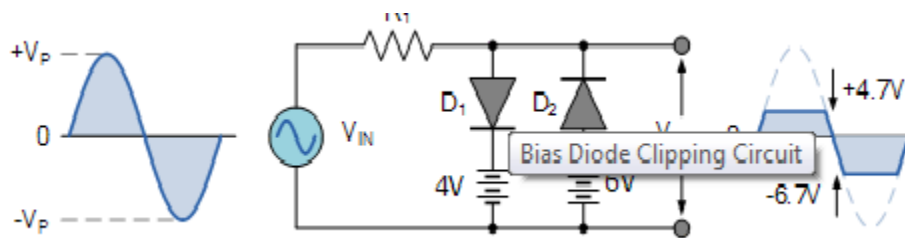
<https://drive.google.com/open?id=0B825Y0yL9dcIWINTQmJoUkV4ckU>

<http://ucdart.github.io/education/eec134>

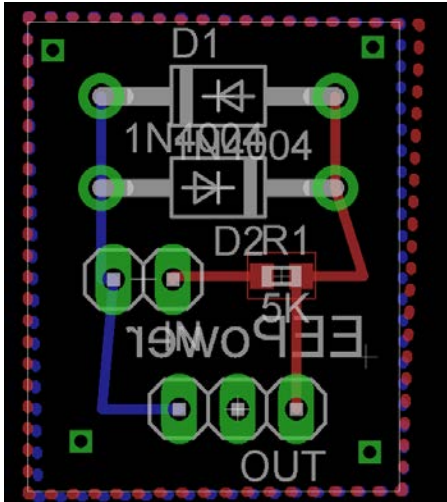
- The following pictures show the result of using the protoboard in our radar system:
Gain = 35, distance = ~30m, standing still with the metal sheet.



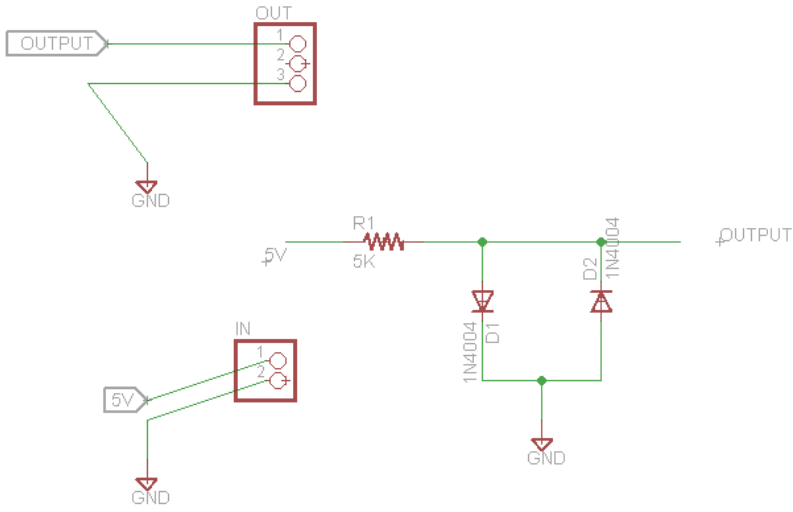
- Also, as the original plan (it wasn't used in the final design, but it was tested), we wanted to have a voltage limiter in our radar system that looked like this:



I also used EAGLE to design this voltage limiter circuit module to be connected to the whole project:



PCB Layout for the voltage limiting circuit (16.10x20.27mm)



This is the schematic

**Extra Note:

For our senior design I worked on:

- Module PCBs for the gain stage and LPF (a quick design to help us with the measurements since our main baseband wasn't working properly)
- Helped with PCB designs
- Made accurate footprints for the RF components
- Made the protoboard for the baseband that was used during the last stages of measurements since our main baseband PCB doesn't work.
- Helped with testing the PCBs
- Helped with do the measurements and testing the radar during each step of the project
- Helped with soldering the PCBs, Prototype boards
- Worked a little on the Matlab code for signal processing to make the program more accurate and faster (we didn't do a lot of modification. We just changed the tp time amount to shorten the time it takes for the program to run.