

# Spring Quarter 2015

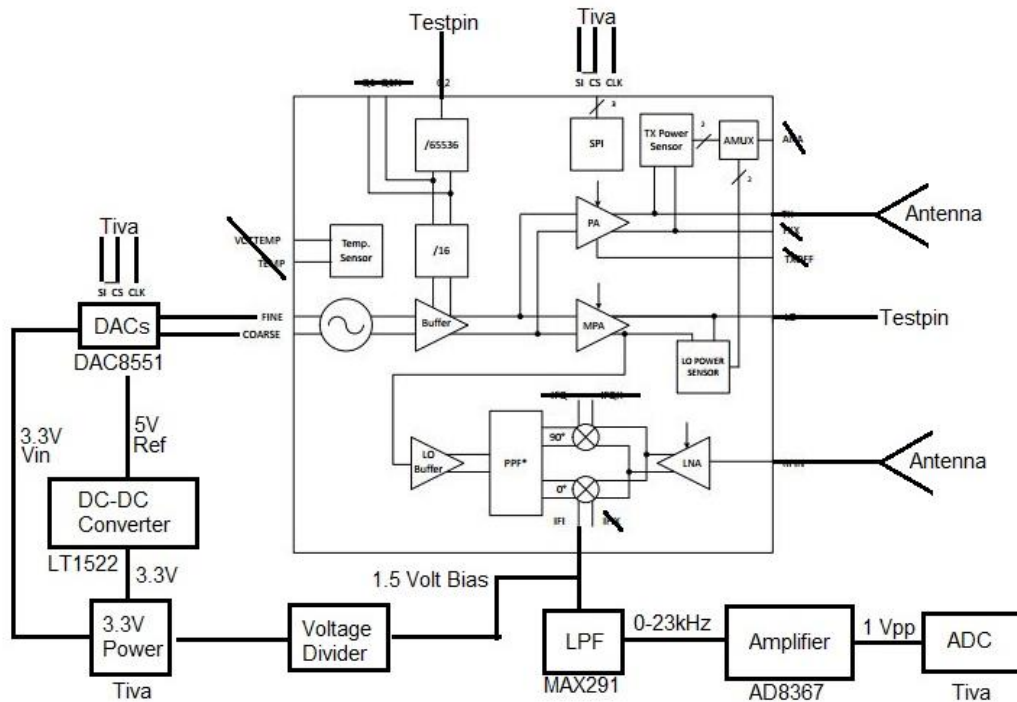
## EEC 134 Radar System Design

### Team Hero

### Vincent Li

## PCB Design on KiCad

## Application Note



### Design Goals

Our goal is to design a compact, low power and light weight radar system that is also powerful at detecting short distance object with great accuracy and resolution. At millimeter wave, the system can do imaging such as detecting the physical appearance of a rough surface or the speed

of micro fluid in a biomedical chip. With these ambitious goals in mind, we want to build the best PCB, and Code with minimal noise and maximum efficiency.

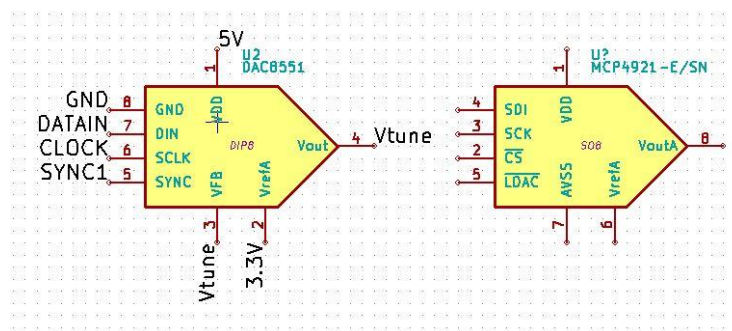
## Using KiCad

### Schematic

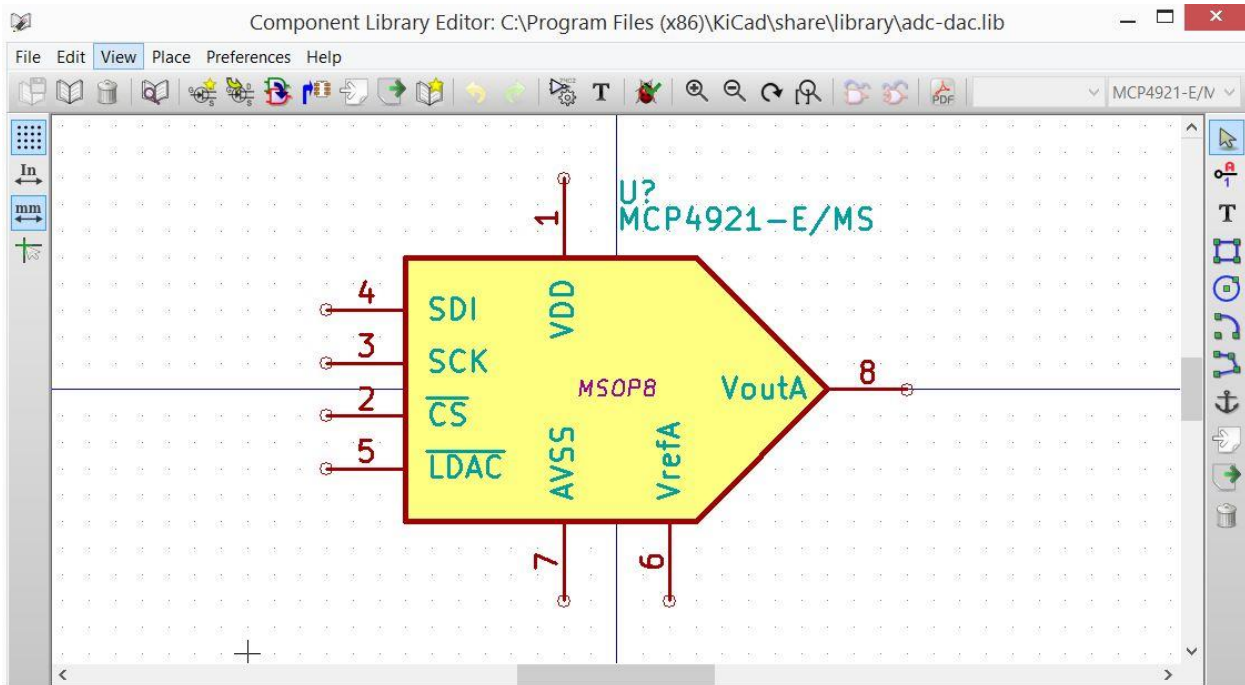
I focused my effort on using KiCAD to build a functional PCB. The software allows users to go from schematic to layout in a very straight forward fashion. The output will be a gerber files and a plot for drilling. Before going into the actual design, there is a learning curve to KiCAD, which will take the rest of the fall quarter to figure out and explore its capability.

First is finding the right figures in the schematics. After drawing everything down on the paper and deciding on the components, designers need to read the datasheet of the components. Realizing the application of the component can be challenging, since most datasheet does not provide very good examples on how to use the product. Or, there is a certain level of assumed knowledge that requires some expertise in the field to understanding. For beginner, all the acronyms, function blocks, settings, voltage and etc. are difficult to understand. These may or may not be text book knowledge. Using google to find the definition will take up majority of the research time.

After figuring out how to wire up everything, it is time to realize the schematics in KiCAD. When I first start using the program, I would always try to find the extract components from the existing library. Soon, I realize the KiCAD library is very incomplete and outdated. Then, I start creating my own components. I will choose an existing component with similar number of pins. Having the same number of pins is all I need to create a new component that will work for my own design. I would edit the component to include the correct pin number, pin name, and pin position. For example, I would use a MCP4291 to create my DAC8521.

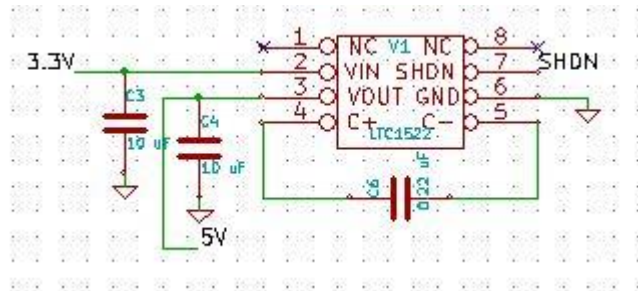


Notice that they are almost identical, besides the pin number and names. To modify the pins, I will open the library edition for the component.

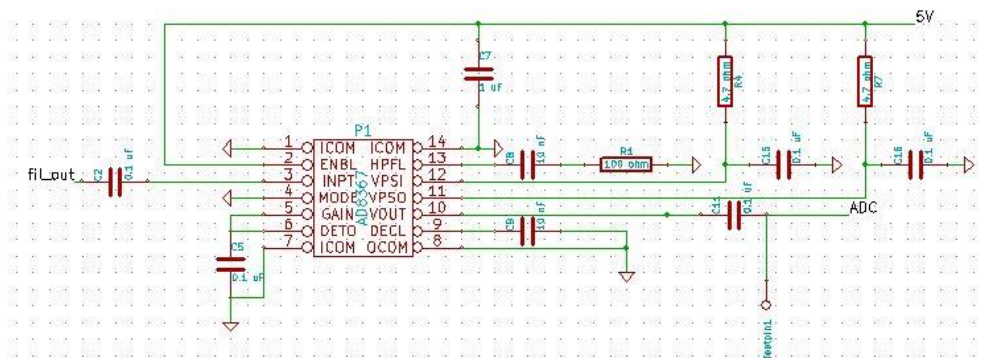


After changing the component, I will save it as my new component to be used for the schematic.

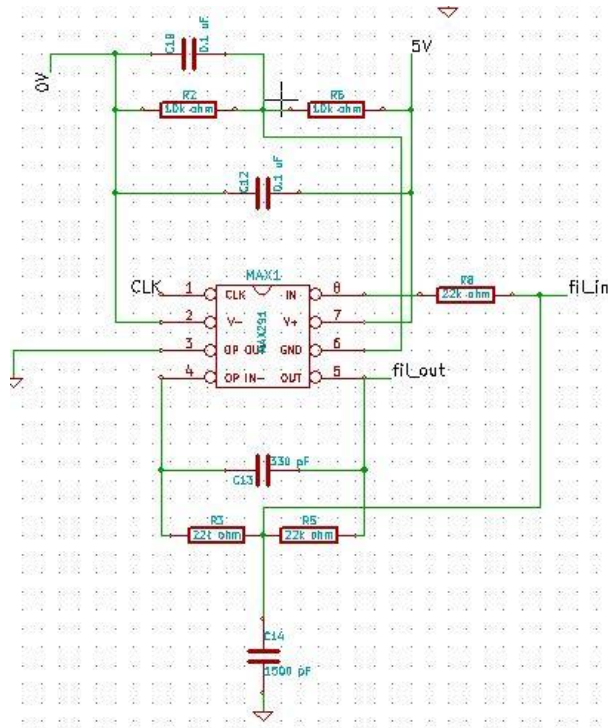
Similarly, I created the LTC1522



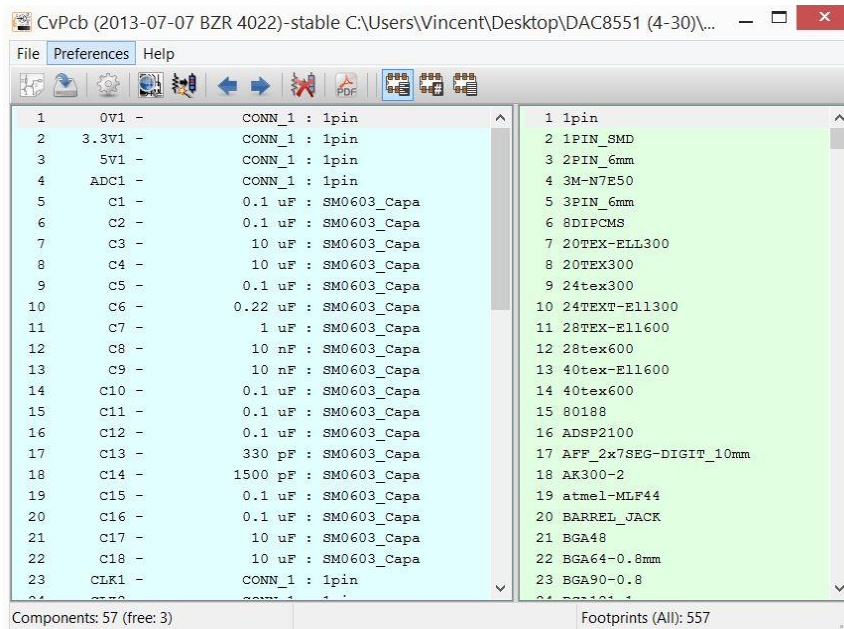
**AD8367**



**Max291**



I label my pins so that they can be connected to the appropriate headers without using wire. After all the labeling, headers and components are set, most people usually do an auto annotation to make sure all the pins are label independently. I also used the error checking function built in KiCAD to check my schematic before I produce a netlist, that will be my shopping list to convert my components to modules. I open CvPcb, which gives me the following chart.

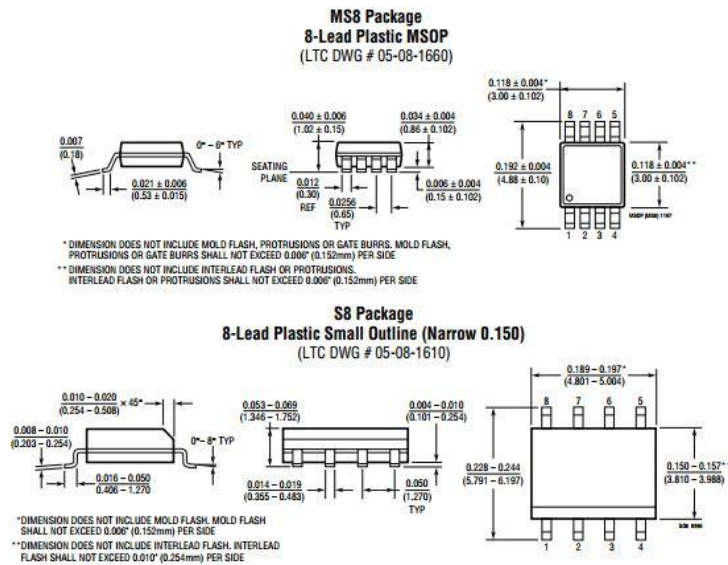


Usually all the capacitors, inductors and resistors are matched with SM0603. The more special components are rarely found in the module library. I match the components to the nearest module and edit the module at the layout.

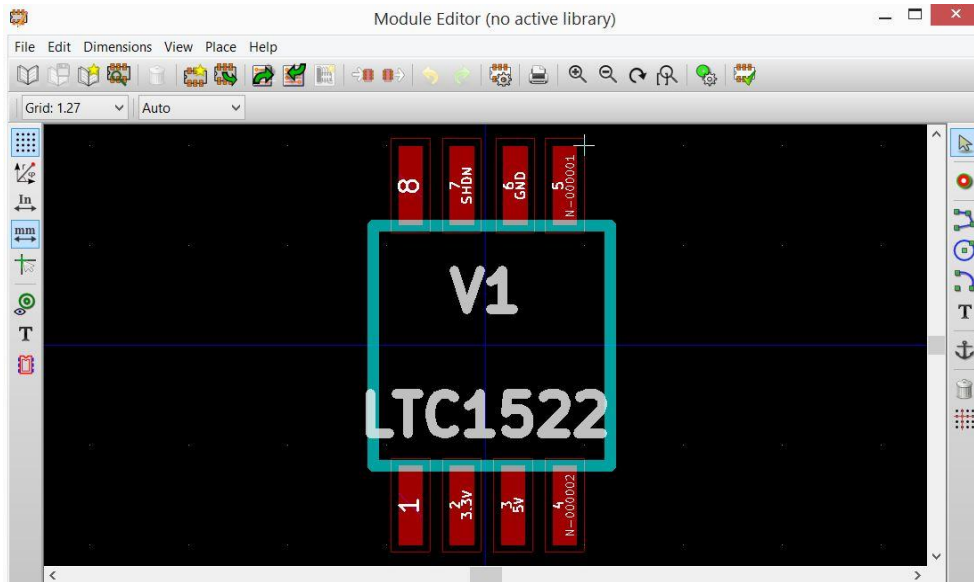
## Layout

In this part, knowing all the numbers about the exact features of the module is very important. It will determine the success of the PCB, whether all the components will fit onto it. The period of waiting for the PCB is long so extra attention is required before sending the PCB in. At this point, a new detail reading of the data sheet is needed to accomplish the physical design. For example, the package description in the datasheet for the LTC1522 is shown below.

### PACKAGE DESCRIPTION Dimensions in inches (millimeters) unless otherwise noted.

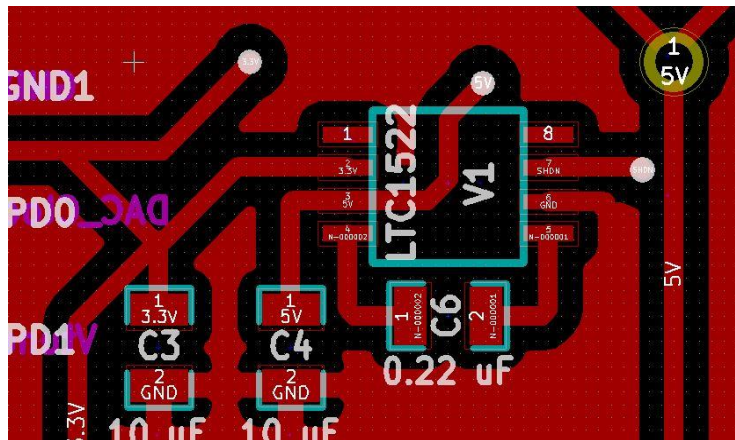


From the Package Description, the pin size, location, distance, and the overall chip width and length are presented. After learning about the detail of the module, it is time to open the editor and start with a raw module. It is all about paying attention to the detail, and have all the right calculation in mind. The actually components are very small so the space for error is very small. Another tricky part about reading from the datasheet is getting the right information. Some components require extra resistor and capacitor for protection or noise elimination. Circuit tricks can help to preserve some of the very expensive components for unnecessary damage. Reflection can do tremendous damage to the chip. Sometimes, these damages are irreversible. It would be devastating to find out that a component is fried after a few test trials. Removing the components from the PCB is as risky as putting them on. The reason is due to the temperature intolerance of Silicon integrated circuit chip. At 200 degree celsius for 10 minutes could lead to shortage inside the chip.

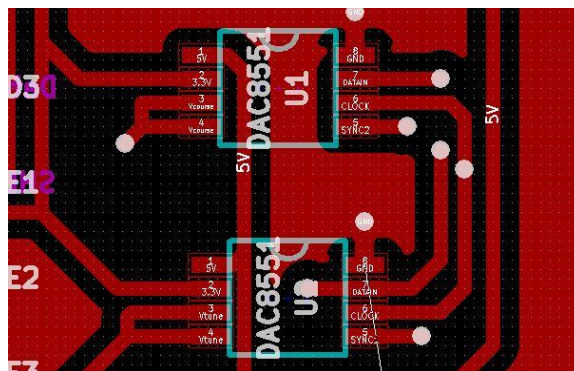


The connections are very important. Each pin must connect exactly to its destined location.

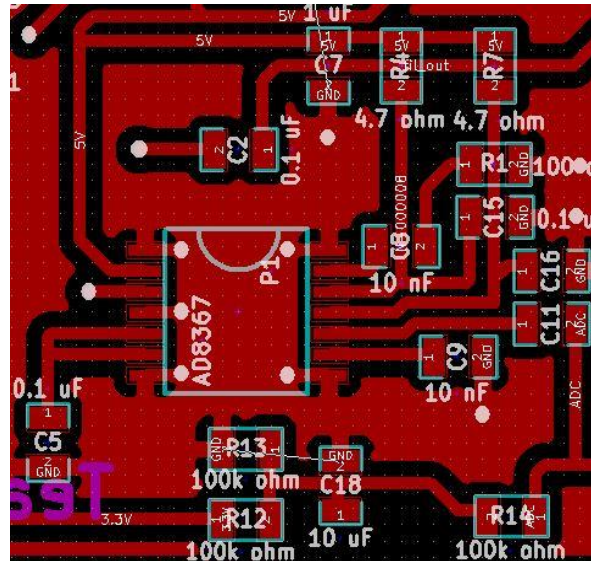
I created the LTC1522



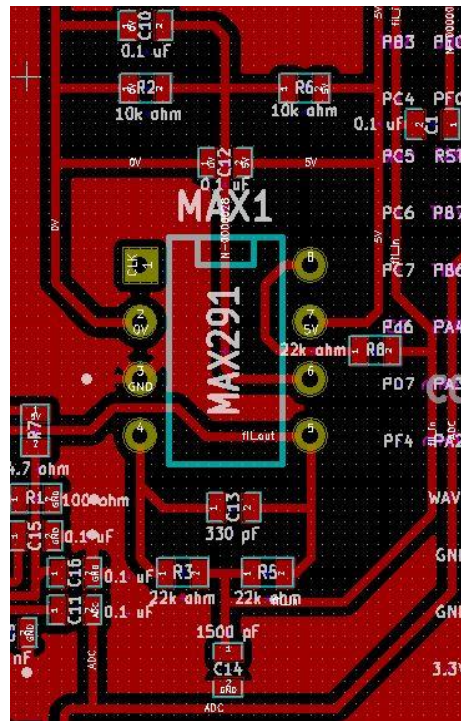
DAC8551



AD8367



MAX291



I recommend creating a 8 pin through socket instead of the package. It is also reusable in case the component is broken. This option may not be available for all the modules since some components are only offered in surface mount.

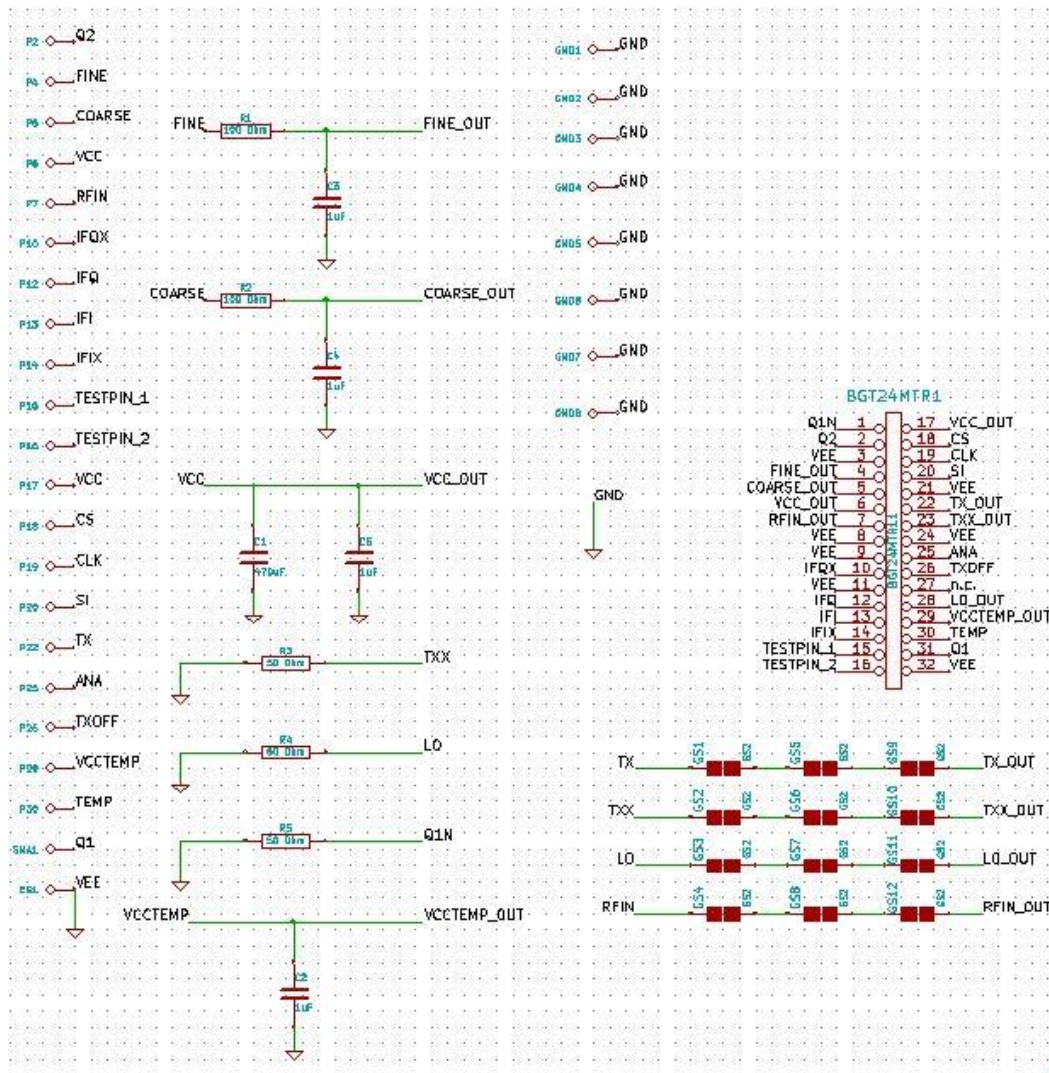
## Gerber

The final part is exporting the layout into Gerber file, zip and upload to OSHpark.

## Infineon

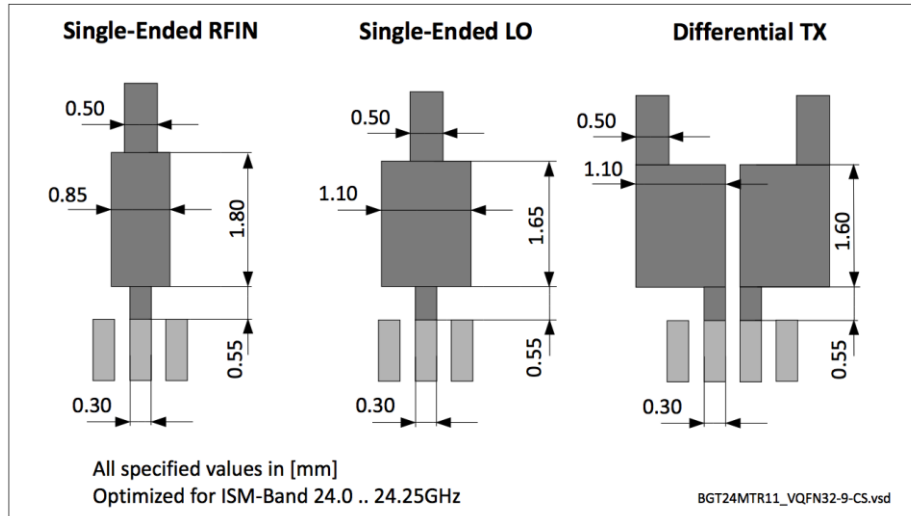
Our team took the challenge to build a 24 Ghz Radar system that utilizes the Infineon BGT24MTR11 Chip as the transceiver. This system involves touch screen on board DSP, Tiva Programming, Low Pass Filter & DAC & Amplifier PCB Design, and Antenna PCB Design. The 24 Ghz PCB uses RO4350B Laminates material in order for the patch antennas to radiate out the signal. Our vision is to create high frequency speed detection and range detection with a low weight and low power system.

### BGT24MTR11



The feed-in system for TX, TXX, LO, RFin take some time to design. We need to adjust the width of the wire blocks connecting out from the BGT24MTR11 chip and into the chip. The purpose is to reduce reflection from the load by impedance matching.





## Sending out the PCB

We sent out the PCB to Bayarea Circuit. Due to the customization of the 24Ghz antenna PCB on RO4350B material, we waited three weeks before the fabrications finished. Before sending out the antenna, we setup the Chamber for testing the antenna, making sure that it has -15 dbm at 25 Ghz so we know it operates at the right frequency. We also designed multiple versions of the PCB for testing purposes. The regular PCB came back within a week. We ordered all the electrical components from Digikey. These came within a week as well. With everything done, we are at week 7. Everything came back to us at week 10, which put us into the soldering and testing phrase.

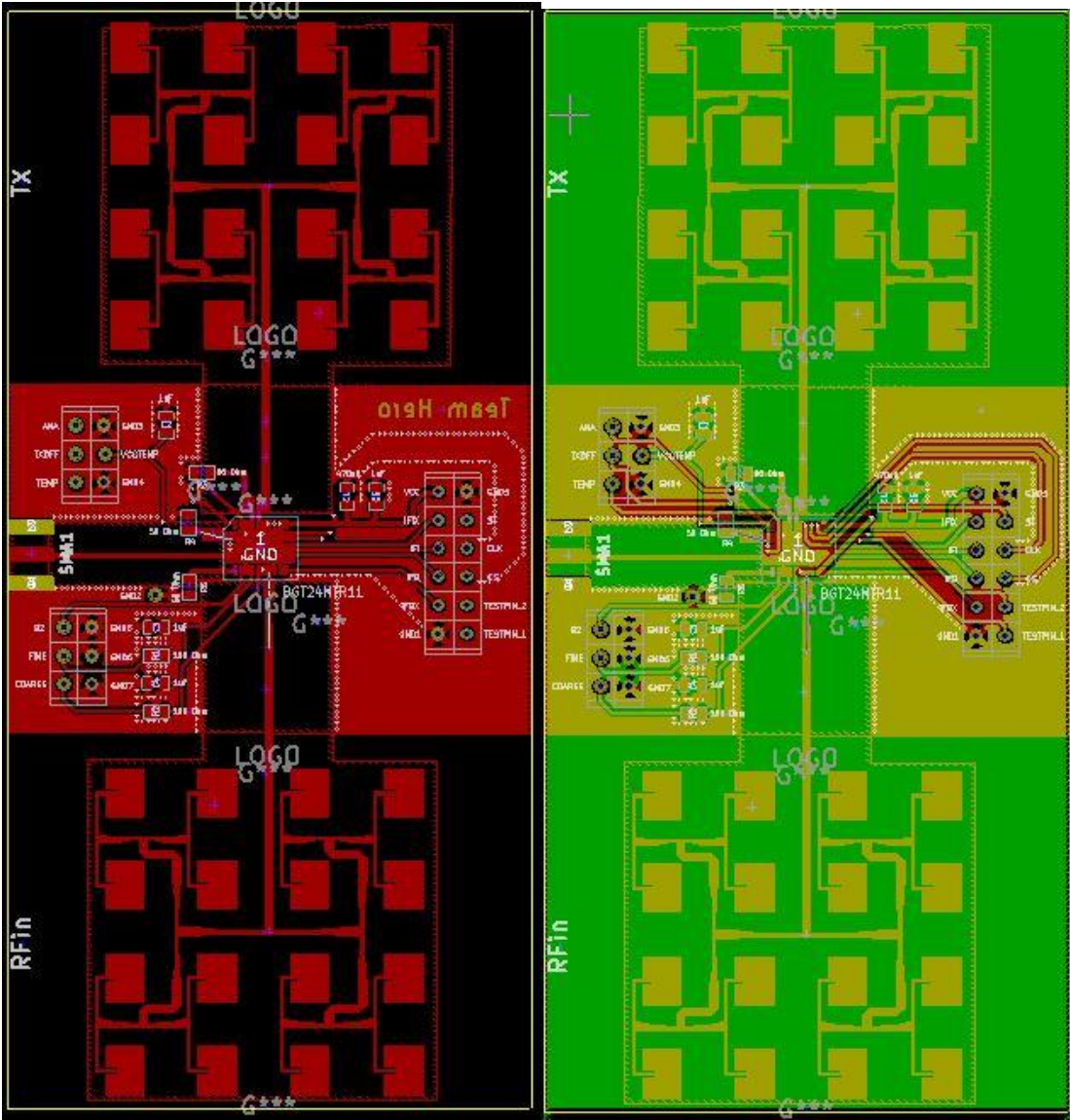
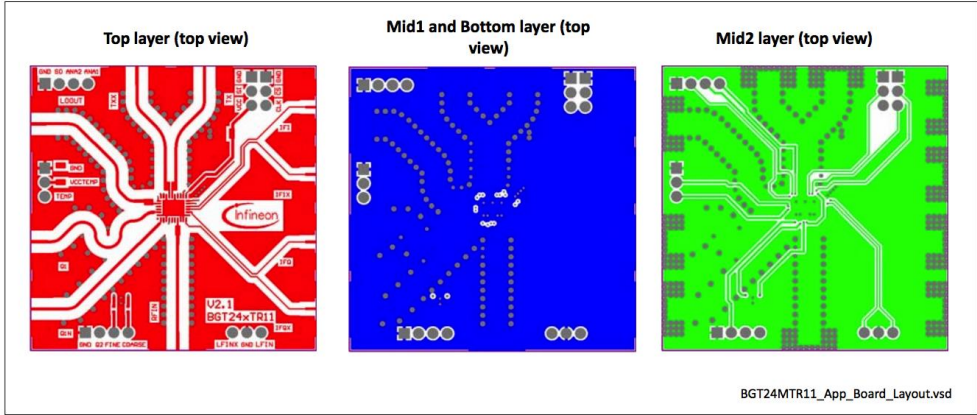
## PCB Soldering

For our designs, traditional soldering will not work with just rosin wires and hot iron. Due to the compact design and small components, we used Rosin Flux (a type of soft rosin material) and hot plates. We carefully put Flux onto all the solder mask and smaller amount on the Surface mount components; such as the DAC's, and amplifier. We set the hot plate to 200 degree celsius for roughly 2-4 minutes. When the PCB heats up, the Flux turns into metal that glues all the components onto the PCB. We examined the binding carefully under a microscope to ensure that we are not shorting any of the pins or misplacing any components.

## PCB difficulty

The most difficult part was making sure that the multiple pin surface mount chip would not be shorting. When we placed the Flux onto the solder mask, there were always chances that we would put extra. Since the solder mask pins are very small, a small leak of the Flux will glue two pins together. We had to be patient about the soldering process to move the components in circular motion so that the flux would move around as well. In case there is noticeable electric shortage, we must remove the components and take out the extra flux. Another concern is that IC





This is the antenna PCB. The array antennas are on the top and bottom of the board for transmission and reception. The feed in system is designed for 50 ohm impedance match,

following the guideline from the Data sheet. Notice that we placed resistors to some of the pins. It is because the board will transmit the signal at two phases with 90 degree difference to cancel out the noise. To utilize the feature, it require more complicated setup. For our project, we do not need a very clean & strong signal to be detected. Therefore, we will only use one phase of the signal and tie the other end to a high frequency 50 ohm resistor.

The power consumption of the chip is 700 mW. But, we can turn off the transceiver with a duty cycle of operation for 10 ms and down for 490 ms. This is equivalent to a power reduction from 528 mW needed in the continuous mode, with maximum output power and current consumption of the BGT24MTR11, down to merely 12 mW when the radar is operated in the duty-cycling mode, while providing the maximum output power of the IC.

### Low Pass Filter, DAC, DC-DC Converter, Amplifier

We designed a very compact circuit to fit everything onto one chip. The process took a lot of planning, and trials & Error. We use both side of the PCB to allocate space for wiring.

