6-13-2018

RealSense Relay Board

Shivam Gandhi
Santa Clara University, sgandhi@scu.edu

Sean Giblin
Santa Clara University, sgiblin@scu.edu

SANTA CLARA UNIVERSITY
DEPARTMENT OF ELECTRICAL ENGINEERING

I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION
BY

Sean Giblin
Shivam Gandhi

ENTITLED

RealSense Relay Board

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

BACHELOR OF SCIENCE
IN
ELECTRICAL ENGINEERING

Ramesh Abhari
6/13/2018

Thesis Advisor(s) (use seperate line for each advisor) Date

Shoba Krishnan
6/13/18

Department Chair(s) (use seperate line for each chair) Date

Santa Clara University
School of Engineering

Senior Thesis

Students
Shivam Gandhi and Sean Giblin
Electrical Engineering

June 13, 2018

Advisers
Ramesh Abhari Ph.D
Electrical Engineering
rabhari@scu.edu

1 Abstract

RealSense is an emerging technology owned by Intel. RealSense cameras are unique in the sense that they have depth sensing abilities allowing for object tracking and identification. Due to the wide range of applications RealSense cameras are found in a multitude of different environments. As with any new technology these cameras experience technical issues that can affect performance. It was the goal of this project to solve the particular problem of RealSense cameras freezing. RealSense cameras are data collection devices, often times the continuity of that data is extremely important and any downtime is to be avoided. Physical resets require someone on scene to unplug and plug back in each failing camera. This can be time consuming and ultimately costly. Often the data collected by the camera is operated on in a location thousands of miles from the actual physical location of the camera. In order to solve this problem we designed a interface Printed Circuit Board (PCB) that can be controlled via secure shell (SSH) from anywhere in the world and switch the power to the camera using a metal-oxide-semiconductor field-effect transistor (MOSFET). This particular board has a capacity of five RealSense cameras operating independently. After considerable design and manufacturing the boards did not pass the functionality test. Power control performs as intended, however, the SuperSpeed signal lines that pass the data from the RealSense through the PCB did not function as intended causing the cameras themselves to not function.

Contents

2 Senior Design Project Description

2.1 Project Description

This project is centered around fixing a real problem that exists within warehouses. Within a warehouse there would be multiple Intel RealSense cameras that detect various situations that occur in that environment. One example being object tracking and recognition within these warehouses; open source coding algorithms enable the cameras to detect and track various objects within a warehouse, both of an animate or inanimate variety. The cameras can detect objects and their characteristics within an environment like that. These cameras from Intel have the ability to not only detect objects and track them, but they also have the ability to detect depth giving them a wide range of capabilities.

Intel RealSense cameras have three lenses. A conventional lens, an infrared lens, and an infrared laser projector lens. They work together with onboard image processing hardware to help capture a 3D image.

These cameras themselves unfortunately have problems. There are many of these cameras being used within these warehouses and occasionally some of them stop functioning requiring a hardware reset. Currently when these devices need a hardware reset an individual will need to walk over to each individual non functioning camera and manually reset it. In other situations the cameras may be in locations where it is not easily accessible, making physical hardware resets difficult, or realistically impossible. Rather than rely on a person conducting a physical hardware reset we would like to offer a remote hardware solution. We would also like to give this board enough functionality that lets it be used not only in this niche application, but also lets it be used within the hobby community.

2.2 Objective

Our objective is to create a budget orientated PCB Board that can effectively hardware reset each individual camera without effecting the integrity of other connected cameras. There should be no errors within the signal coming into and out of the board, as well as making sure the board layout is designed with signal integrity in mind. The board needs to comply with USB 3.1 protocol, and pass their compliance testing.The board needs to control the power state of up to five RealSense cameras.

- Research USB 3.1 compliance objectives reducing the risk of signal problems related to reflections, transmission loss, crosstalk, etc.
- Simulate all high speed lines with an RF CAD tool such as ADS (Advanced Design System) by Keysight.
- Produce various graphs and diagrams from the simulations confirming that our simulations adhere to the protocol for USB 3.1 SuperSpeed.
- Obtain gerber files from the CAD tool and get a PCB manufactured.
- Test the physical board and compare it to the simulations.
- Display the cameras working using the board with a system showcasing its abilities.

2.3 Needs

Due to the camera's ability to track and detect various objects, as well as the ability to do facial and gesture recognition there are a lot of situations where people may benefit from our board. Although the initial conception of this project was focused on one problem that Intel was having, the functionality of the board can definitely be expanded.

- Facial recognition to help detect drooping eyes during driving, helping to alert sleepy drivers.
- Baby monitor cameras which can detect whether a baby's chest is moving up or down, alerting parents about the possibility of SIDS.
- The use of these cameras within a warehouse, detecting various personnel as well as objects situated around the location.
- The use of these cameras to do software modeling of objects since they can detect depth

2.4 Design Considerations

2.4.1 Engineering Requirements

These cameras must use USB 3.0 minimum. Due to this the design considerations of the PCB layout must be considered. Signal integrity is something that must be accounted for when researching, designing, simulating, and building this board.

- Follow USB 3.1 SuperSpeed protocol when designing the high speed data lines within ADS
- Produce the required simulation results and diagrams such as reflection coefficients and eye diagrams.
- When choosing components for the board choose components that are meant for high speed situations and follow guidelines strictly.
- Test the boards using compliance testing standards for USB 3.1 SuperSpeed.
- Understand the influences high speed circuits have on each other with issues such as crosstalk and both inductive and capacitive coupling.
- Being aware that at high frequency components such as resistors, capacitors, inductors, and traces must be modeled accurately, observing their effects and behaviors as frequency is increased.
- The PCB substrate material matters just as much as the layout. Choosing the right material can help increase the integrity of our signal, or help compensate when a layout may not be 100% perfect.
- Which type of trace to use, and balancing the pros and cons carefully to get the best performance without over complicating the design is important in both ease of reaching our goal and keeping costs as low as possible.
- Match our input and output loads to the characteristic impedance of our differential signal traces to reduce the reflections and achieve maximum power transfer.

2.4.2 Marketing Requirements

- Keeping the board as compact as possible (being able to fit within an average man's pocket) is an important requirement in today's small device world. Not only does this help us on the engineering aspect of our project, but it helps keep the device sleek and attractive to potential consumers.
- Design and development cost should be as low as we can make it without compromising on the integrity of our final product. For this we have set a maximum development cost (per board) to be less than $150. We expect manufacturing costs to be much lower, and much more attainable and attractive for the average hobbyist/corporation.
- A sleek enclosure must be made for the board. Having just a PCB bare-bones is not ideal, nor attractive for the average consumer. This will also help protect the board. The physical look of the final design is important as well. It must be intuitive, eye catching, and feel solid in the hands.
- In today's internet age the device must be easily accessible and controlled through the web. This puts usability at the forefront and allows users to manipulate the device with their phones or computers without having to step up to the device.
- Reliability is an important marketing requirement, as well as engineering requirement. If reliability is poor then usability is poor, which consumers will not like.

3 Proposed Solution

Our solution to the problem we were given is to design a board that has the capability of using a network connection that can be accessed from anywhere to hardware reset the devices attached to the interface board.

3.1 Block Diagram

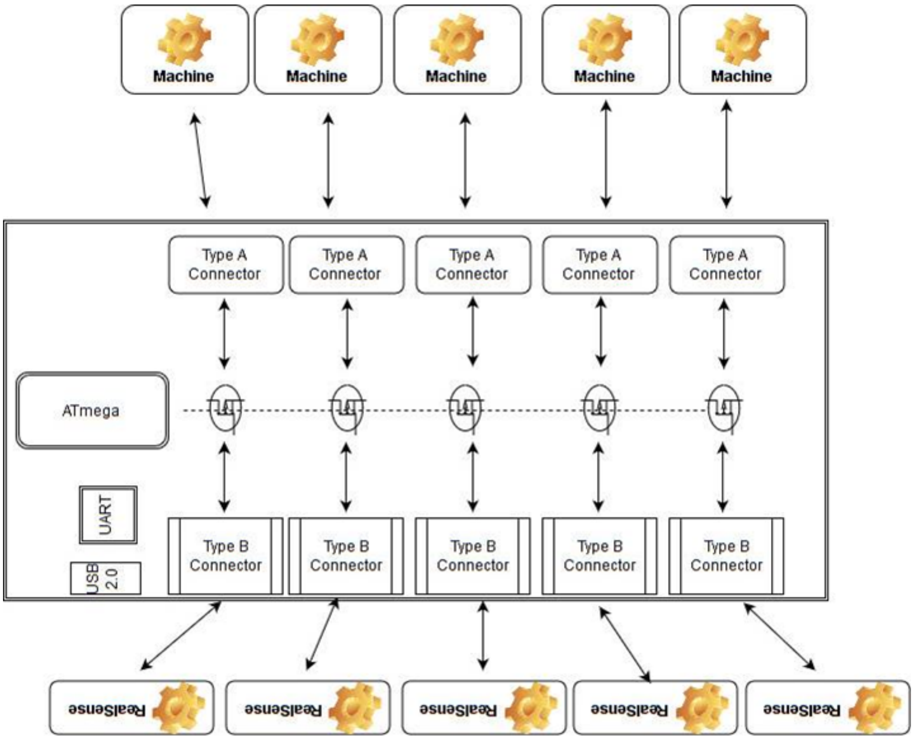

Figure 1: Interface Board Block Diagram.

The biggest problem that we will have with our board is running the data lines using USB 3.1 protocol, which requires an in depth look at signal integrity when designing the PCB and understanding what happens to the signal as it travels through the traces.

On the top end of the board we will have a type A USB connector and at the bottom we will have a type B USB connector, both satisfying the requirements of USB 3.1. USB Types denote the type of cable shape. Type A is the connector that is usually plugged into a computer, while type B is what we usually plug into our phones and mobile devices.

Both power and signal traces will be passed through from one end to another, but the power traces will have a switching circuit in between them controlled by an ATmega processor. Connected to the ATmega processor we will have a modules allowing for internet and bluetooth connectivity. This allows for our device to be a 100% modular and placeable anywhere a consumer would want.

3.2 Expected Results

We expect to have a fully functioning, self-contained system that solves the current problem regarding the control of RealSense Cameras. The finished product will consist of a PCB enclosed in a 3D printed case. This case will allow a total of eleven jacks; five for input, five for output and an optional jack if tethered control via USB is desired. There will also be an option for remote control of the board via bluetooth or WiFi/Ethernet.

- We expect this board to be fully functional for an indefinite amount of time. This means that all 5 cameras that are connected to the board should be fully functioning and transmitting a readable signal to its respective system for an indefinite period of time.
- Once the cameras are fully functioning as user will be able to access the PCB via serial command or through WiFi in order to individually power cycle each camera without affecting the operation of the other cameras.
- There will be commands programmed into the operation aspect of the board to do three major tasks.
    1. Cycle power to one of the cameras
    2. Cycle power to the microcontroller on PCB
    3. Cycle power to all of the camera units simultaneously

The applications of this PCB are vast and will require functionality in various different environments. The final PCB design will be able to function in normal room temperatures. The cameras themselves will have to be located in a suitable temperature range for their functionality but the PCB controller may be placed in an attic or such a place where the temperature is not regulated.

3.3 Budget

The initial proposed budget below took into consideration of every part of the relay board we are designing. We had an additional section for parts replacement in case they arrive damaged or parts stop working while in use. The two biggest cost areas occur because of the Intel RealSense cameras and the general cost of manufacturing PCBs. We hope to keep the cost of our board alone below $150

Table 1: Budget List

Item	Description	Quantity	Price
Double Sided Single Layer PCB	Board Manufacturing	1	$70
USB 3.1 Type B	Connector	5	$1.43
USB 3.1 Type A	Connector	5	$2.14
ATMega328P	Microcontroller Processor	1	$4.30
Intel RealSense ZR300	Long Range Object Tracker	1	$289.00
Intel RealSense SR300	Facial Recognition Camera	1	$109.00
Arduino Ethernet Shield 2	Board Ethernet Connectivity	1	$45
HC-06 Bluetooth Module	Bluetooth Module for Wireless Control	1	$9
Total Cost	-	-	$543.95

The total cost of our board in our predicted budget list is $146.19, which keeps us within our proposed budget per board. We will see if that changes as the project develops further.

After designing our board fully and sending it out to get manufactured, with assembly our board ended up being roughly $1500. Due to the design constraints we had in order to adhere to signal integrity we ended up having a four layer board, which drove up the costs considerably [1]. For a single one off prototype board the cost of PCB manufacturing goes up dramatically compared to a single layer dual sided board.

3.4 Implementation Alternatives

We took a step back and looked at other implementation methods that we could use for our project and weighed their pros and cons against our current model.

- Cable couplers each with its own built in module for Internet connectivity. They will essentially look like standard cables with all of the electrical components housed on a small chip near one of the ends.
- Transmit the data of each camera over the Internet and have the computer receive the video without using any sort of cable at all. It will
- Instead of relying on a microcontroller processor use a much smaller more rudimentary system similar to what we find on IoT devices.

We thought long and hard about each of these implementation possibilities. They were each great in their own way, but they also had a lot of short comings that made us keep going back to our original design.
Cable couplers would be very neat, but connecting multiple cameras together and having them connected to a centralized network would make it difficult to package in a clean compact device.

Streaming the data from the camera to the computer would have fixed this problem multiple times over, however it is incredibly difficult to do this cheaply, and the processing requirement would be too much for a small microcontroller processor to handle. And that is just for one camera. Add in four more and it's clear to see why we need a physical connection.

A smaller more rudimentary system would be a way to go as a second redesign in order to cut cost, but as a proof of concept (and with these microcontroller processors being so cheap) it is much easier to stick with an ATmega processor.

Due to all these reasons we decided to stick with our original plan of designing a PCB that could handle all of the problems we were trying to solve in a small compact device.

3.5 Process and Implementation Method

The first and most fundamental process is the design process. This includes both a block diagram schematic, PCB schematic and a PCB layout in the appropriate CAD tool. We will be using ADS (Advanced Design Systems) from Keysight to do this.

ADS is a CAD tool commonly used within RF, microwave, and high speed digital applications. Within ADS not only can we create schematics and layout, but we can do a multitude of simulations for signal/power integrity as well as see how our circuit behaves over time.

Once we are satisfied with the dimensions and components used along with the apparent functionality of the board we can begin our computer simulations. These simulations will be concerned with the following:

- Crosstalk
- Capacitive and Inductive Coupling
- Reflection due to improper matching
- Capacitance, inductance and resistance of traces
- Appropriate length of traces for optimal signal delivery
- Characteristic impedance of our signal lines
- Showing how much of a difference the right PCB Substrate material can make

Once we have simulated and proven the desired results with 100% accuracy and consistency the PCB layout will be adjusted as needed and the respective Gerber files will be generated. The Gerber files will then be sent off a PCB manufacture house [1].

We will start off with ordering one board, proving its functionality, and then continuing to improve upon its design. Our first board will be purely about the actual signal integrity of the traces, making sure everything works as planned. After that we will begin to add more ports, and Internet functionality.

Once the board arrives we will test it with RealSense Cameras for different lengths of time and with different open source applications of the code that they operate on. That includes various situations such as object tracking and facial/gesture recognition

Another thing we must do once we have received the working boards form the fabrication house will be to make sure that we match the compliance standards set by the USB 3.0/3.1 spec [2]. Even though we simulated the board within ADS and made sure that the simulations were correct, under real life testing the results could be incredibly different than what we hoped to achieve.

In order to do this, we will need access to a high-level piece of equipment that can produce an eye diagram on a signal of GHz. If Intel cannot accommodate us at their Santa Clara campus (compliance testing is done in Hillsboro, OR), then we will explore our relationship with KeySight.

We can also test the reflections within the signal traces using a VNA, or "Vector Network Analyzer". This would give us the S-parameters of our board which we can then use to find out if our load is receiving the maximum power or not. Although we calculate the differential characteristic impedance of our traces to be as close to 90Ω as possible to match it with both our input and our load, manufacturing tolerances may affect that [2].

Once we are satisfied with both compliance and functionality we will design a 3D printed enclosure for our design and confirm its robust performance in various environments.

4 Preliminary Results

We started with a lot of research, which took up most of the quarter. It's very important that we take this step by step and look at the various factors that would impact our device in terms of signal integrity, and how we can achieve the best performance within our device.

4.1 Board Material

Different substrate material have different "Dielectric Constants" (abbreviated ϵ_r or Dk) and "Loss Tangents" (abbreviated $tan\delta$). Loss Tangent is also known as the Dissipation Factor (Df). Both the dielectric constant and the loss tangent have been a guiding parameters when looking at PCB material for high speed circuits. [7]

Dk is closely related to the impedance of the circuits that will be fabricated onto the material. We will see this more when we talk about the different trace types. Changes in the dielectric constant are seen as we see a change in frequency, temperature, or other factors. They can negatively affect the performance of these circuits and can change the impedances of the signal transmission lines in unpredictable ways. [7]

The higher the Dielectric Constant, the slower a signal travels on a trace, the lower the impedance of a given trace geometry and the larger the stray capacitance along a transmission line. Given a choice, lower dielectric constant is nearly always better. Loss tangent is a measure of how much of the signal pulse (electromagnetic wave) propagating down the PCB transmission line will be lost in the dielectric region (insulating material between copper layers) [7].

Loss tangent is a function of the material's resin type and molecular structure (molecular orientation). We can also see that attenuation is a function of both the dielectric constant and the loss tangent [7].

$$\alpha = 2.3f \cdot tan(\delta) \cdot \sqrt{\epsilon_{eff}} \tag{1}$$

Where ϵ_{eff} equals:

$$\text{when } \left(\frac{W}{H}\right) < 1$$
$$\varepsilon_e = \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2}\left[\left(1 + 12\left(\frac{H}{W}\right)\right)^{-1/2} + 0.04\left(1 - \left(\frac{W}{H}\right)\right)^2\right]$$
$$\text{when } \left(\frac{W}{H}\right) \geq 1$$
$$\varepsilon_e = \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2}\left(1 + 12\left(\frac{H}{W}\right)\right)^{-1/2} \tag{2}$$

At higher frequencies both of these will play a large role. Within our design frequency is not something we can manipulate, since the USB standard is 10Gbps, so the only factors that are within our abilities to change are loss tangent and dielectric constant [2].

In our application we decided to use GETEK over FR4 [3]. FR4 is the most commonly used substrate made with composite material composed of woven fiberglass cloth with an epoxy resin binder that is flame resistant [3]. It exhibits fantastic results for its price, but its relatively high Dielectric Constant an Dispersion Factor make it difficult to choose within high speed circuits [7].

GETEK in comparison is made with polyphenylene Oxide epoxy resin (according to Prototron's datasheet), and is incredibly close to FR4, but has a much lower dielectric constant and dispersion factor as shown below. You can see a direct comparison between the data sheets of GETEK and FR4 in Figure 2 and 3.

Dk, Permittivity (Laminate & prepreg as laminated) Tested at 50% resin	A. @ 100 MHz (HP4285A) B. @ 1 GHz (HP4291A) C. @ 2 GHz (Bereskin Stripline) D. @ 5 GHz (Bereskin Stripline) E. @ 10 GHz (Bereskin Stripline)	3.81 3.78 3.60 3.50 3.50
Df, Loss Tangent (Laminate & prepreg as laminated) Tested at 50% resin	A. @ 100 MHz (HP4285A) B. @ 1 GHz (HP4291A) C. @ 2 GHz (Bereskin Stripline) D. @ 5 GHz (Bereskin Stripline) E. @ 10 GHz (Bereskin Stripline)	0.0110 0.0100 0.0090 0.0090 0.0100

Figure 2: GETEK data sheet outlining the dielectric constant and dispersion factor

Dielectric Constant @ 1MHz	4.7
Dissipation Factor @ 1MHz	0.014

Figure 3: FR4 data sheet outlining the dielectric constant and dispersion factor

Immediately you can see the difference between the two materials. GETEK across the board has a much lower Dk and Df than FR4, and meets the specs we are looking for. Since USB 3.0/3.1 operate between 5 Gb/s - 10 Gb/s we want out Df to be around 0.0100.

4.2 Traces and Routing Methods

The traces we use can have a big impact on our overall design. Figure 4 shows an image of the various types of routing we can do with our traces.

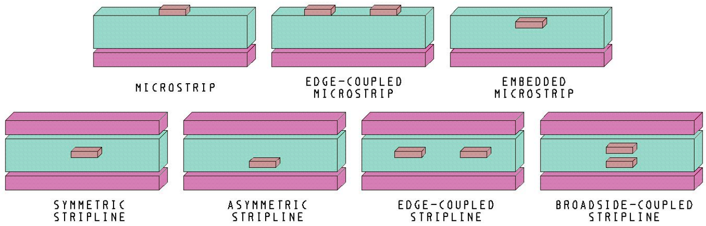

Figure 4: Various trace methodologies we can use within layout [14]

As you can see there are two big types of trace routing we can use; those being microstrip and stripline. Stripline is a transmission line trace surrounded by dielectric material suspended between two reference planes on the internal layers of the PCB. On the other hand, microstrip routing is a transmission line trace routed on an external layer of the board with a single reference plane on the bottom [15].

Microstrip transmission lines are quasi-TEM while striplines are TEM [15]. This means that although our signal characteristic is not as good as stripline we benefit from cheaper, lower cost manufacturing while retaining much of the benefits.

Striplines are more complex to manufacture because it requires multiple layers to support the embedded trace between two ground planes [3]. The good thing about stripline is that the width of a trace can be much smaller within a controlled impedance due to the second ground plane. This allows much greater circuit densities, although this is not practical in our application.

For our project microstrip routing is the way to go, and it helps reduce cost as well as maintain good signal characteristic, which is our main goal throughout all of this.

Because we are designing to USB, we must take into account the fact that our signal traces are differential [2].

The differential traces must be parallel to each other, so we will be using the technique of Edge-Coupled Microstrip routing. A representation is shown in Figure 5.

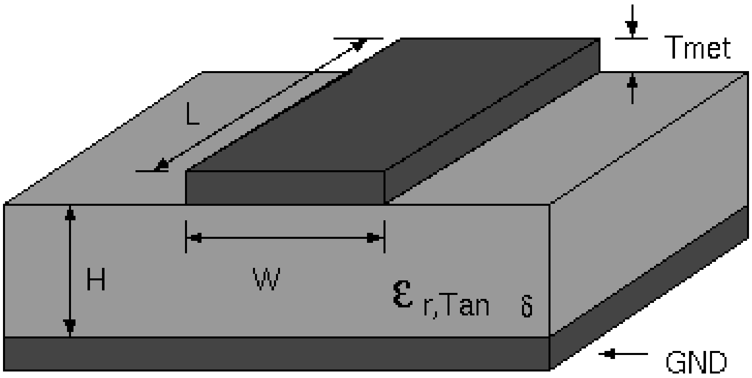

Figure 5: Cross section of a microstrip trace [10]

We also must take into account the different real life scenarios so we can model the line better.

$$\begin{aligned}
&\text{when } \left(\frac{W}{H}\right) < 1 \\
&\varepsilon_e = \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2}\left[\left(1 + 12\left(\frac{H}{W}\right)\right)^{-1/2} + 0.04\left(1 - \left(\frac{W}{H}\right)\right)^2\right] \\
&\text{when } \left(\frac{W}{H}\right) \geq 1 \\
&\varepsilon_e = \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2}\left(1 + 12\left(\frac{H}{W}\right)\right)^{-1/2}
\end{aligned} \tag{3}$$

Since part of the microstrip conductor exists in air the effective dielectric constant, ϵ_e is slightly lower than the dielectric constant. This is good for us since we want it to be as low as possible. It's important to note that these are just approximations, but they set us on the right path for when we simulate everything.

Another important equation for us to look at in terms of microstrip traces is its characteristic [15] impedance.

$$\begin{aligned}
&\text{when } \left(\frac{W}{H}\right) < 1 \\
&Z_0 = \frac{60}{\sqrt{\varepsilon_{eff}}} \ln\left(8\frac{H}{W} + 0.25\frac{W}{H}\right) \text{ (ohms)} \\
&\text{when } \left(\frac{W}{H}\right) \geq 1 \\
&Z_0 = \frac{120\pi}{\sqrt{\varepsilon_{eff}} \times \left[\frac{W}{H} + 1.393 + \frac{2}{3}\ln\left(\frac{W}{H} + 1.444\right)\right]} \text{ (ohms)}
\end{aligned} \tag{4}$$

However, when it comes to differential microstrip routing the equations become much more advanced:

$$Z_{0_{odd}} = Z_{0_{surf}} \cdot \left[\frac{\sqrt{\frac{er_{eff}}{er_{effo}}}}{1 - \left(\frac{z0_{surf}}{\eta_o} \cdot q_{10} \sqrt{er_{eff}} \right)} \right]$$

$$Z_{0_{even}} = Z_{0_{surf}} \cdot \frac{\sqrt{\frac{er_{eff}}{er_{eff,e}}}}{1 - \frac{z0_{surf}}{\eta_o} \cdot q_4 \cdot \sqrt{er_{eff}}}$$

Where:

$$\begin{aligned}
Z_{0_{surf}} &= \frac{\eta_o}{2\pi\sqrt{2}\sqrt{er_{eff}+1}} \cdot \ln\left(1 + \left(4 \cdot \frac{h}{w_{eff}}\right) \cdot \left(\left(4 \cdot \frac{h}{w_{eff}}\right) \cdot \left(\frac{14 \cdot er_{eff} + 8}{11 \cdot er_{eff}}\right) + temp\right)\right) \\
er_{eff1} &= \frac{er+1}{2} + \left(\frac{er-1}{2}\right) \cdot \left(\sqrt{\frac{w}{w+12h}} + .04\left(1 - \frac{w}{h}\right)^2\right) \\
er_{eff2} &= \frac{er+1}{2} + \left(\frac{er-1}{2}\right) \cdot \left(\sqrt{\frac{w}{w+12h}}\right) \\
a_0 &= .7287\left(er_{eff} - \frac{er+1}{2}\right) \cdot \left(\sqrt{1 - e^{-.179u}}\right) \\
b_0 &= \frac{.747 \cdot er}{.15 + er} \\
c_0 &= b_0 - (b_0 - .207) \cdot e^{-.414u} \\
d_0 &= .593 + .694e^{-.562u} \\
g &= \frac{s}{h} \\
w_{eff} &= w + \frac{t}{\pi} \cdot \ln\left(\frac{4e}{\sqrt{\left(\frac{t}{h}\right)^2 + \left(\frac{t}{w\pi + 1.1t\pi}\right)^2}}\right) \cdot \frac{er_{eff}+1}{2 \cdot er_{eff}} \\
er_{effo} &= \left((.5 \cdot (er+1) + a_0 - er_{eff}) \cdot e^{-c_0 \cdot g^{d_0}}\right) + er_{eff} \\
temp &= \sqrt{16\left(\frac{h}{w_{eff}}\right)^2 \cdot \left(\frac{14 \cdot er_{eff} + 8}{11 \cdot er_{eff}}\right)^2 + \left(\frac{er_{eff}+1}{2er_{eff}}\right) \cdot \pi^2} \\
q_1 &= .8695 \cdot u^{.194} \\
q_2 &= 1 + .7519 \cdot g + 1.89g^{2.31} \\
q_3 &= .1975 + \left(16.6 + \left(\frac{8.4}{g}\right)^6\right)^{-.387} + \frac{1}{241} \cdot \ln\left(\frac{g^{10}}{1 + \left(\frac{g}{3.4}\right)^{10}}\right) \\
q_4 &= \frac{2 \cdot q_1}{q_2\left(e^{-g} \cdot u^{q_3} + (2 - e^{-g}) \cdot u^{-q_3}\right)} \\
q_5 &= 1.794 + 1.14 \cdot \ln\left(1 + \left(\frac{.638}{g + .517 \cdot g^{2.43}}\right)\right) \\
q_6 &= .2305 + \frac{1}{281.3} \cdot \ln\left(\frac{g^{10}}{1 + \left(\frac{g}{5.8}\right)^{10}}\right) + \frac{1}{5.1} \cdot \ln\left(1 + .598 \cdot g^{1.154}\right) \\
q_7 &= \frac{10 + 190 \cdot g^2}{1 + 82.3 \cdot g^3} \\
q_8 &= e^{\left(-6.5 - .95 \cdot \ln(g) - \left(\frac{g}{.15}\right)^5\right)} \\
q_9 &= \ln(q_7) \cdot \left(q_8 + \frac{1}{16.5}\right) \\
q_{10} &= \left(\frac{1}{q_2}\right) \cdot \left(q_2 \cdot q_4 - q_5 \cdot e^{(\ln(u) \cdot q_6 \cdot u^{-q_9})}\right) \\
v &= \frac{u \cdot (20 + g^2)}{10 + g^2} + ge^{-g} \\
ae(v) &= 1 + \frac{\ln\left(\frac{v^4 + \left(\frac{v}{52}\right)^2}{v^4 + .432}\right)}{49} + \frac{\ln\left(1 + \left(\frac{v}{18.1}\right)^3\right)}{18.7} \\
b_e(e_r) &= .564\left(\frac{er - .9}{er + 3}\right)^{.053} \\
er_{eff,e} &= \frac{er+1}{2} + \frac{er-1}{2} \cdot \left(1 + \frac{10}{v}\right)^{-a} \cdot e^v \cdot b_e(e_r)
\end{aligned} \tag{5}$$

Because of this complexity we will rely on the LineCalc feature within ADS to help us solve these problems.

Most connectors these days are standardized to a 50Ω impedance. If we had single ended traces to be matched to the connector we would need a characteristic impedance of 50Ω as well. This is because of the reflection coefficient [15]. Since we have differential traces we want our differential characteristic impedance to be around 90Ω. This is due to the differential characteristic impedance being twice Z_{odd} [5].

$$\Gamma = \frac{Z - Z_o}{Z + Z_o} \tag{6}$$

Where Z is either Z_{in} indicating input impedance or Z_L which indicates load impedance. You can see that if the input impedance or the load impedance is 50Ω, and we design our differential signal trace to have a characteristic impedance of 90Ω then we get a reflection coefficient close to 0.

4.3 Layout Considerations

The layout of our board is incredibly important, and even with all the right calculations for our characteristic impedance, and choosing the right material with a low Dk and Df we can still break our design and have a non-functioning board at the end of all this work. Because of that we must take a careful look at how we route our traces when there is more than one trace involved.

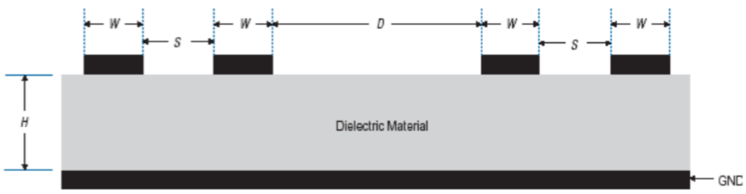

Figure 6: Cross section of an edge coupled microstrip [5]

Because we have differential signal lines we want to do our best to minimize crosstalk and inductive/capacitive coupling. Crosstalk is unwanted coupling of signals between parallel traces. Proper routing and layer stack-up can help minimize crosstalk.

$$D = 5W \tag{7}$$

Where W is the width of the traces. This helps stop crosstalk from happening between the two differential traces [5].

$$S = 3H \tag{8}$$

Where H is the height of the substrate. This is a lot more applicable to the single ended traces on our board. We would like to keep the single ended traces a certain distance apart from one another [5].

You can also see that in the diagram it has the ground plane right under our dielectric. This is important to help minimize crosstalk too. By keeping the distance between the signal layer and reference layer to a minimum we can reduce the amount of coupling that's going to occur by a dramatic amount [4] [5]. It helps decouple the transmission line from adjacent signals.

We also want to widen the spacing between signal lines as much as routing restrictions will allow. A distance of 5W apart is adequate, but more is better if possible [4] [5].

We want to use differential routing techniques and keep differential traces the same exact length. We also want to keep our traces as short as we possibly can.

Microstrip Layout Propagation Delay

$$T_{PD} = 85 \cdot \sqrt{0.475\varepsilon_r + 0.67} \cdot s \tag{9}$$

As you can see the longer the trace is the longer it takes for the signal to get across [5]. We would like to minimize this delay. For signal traces we will also avoid using vias as much as possible. Vias end up causing impedance changes and adds inductance to our traces, which will change how the signal behaves within the circuit. It also makes it harder to match the lines, causing reflections within our system.

4.4 Preliminary Simulation

Once all the research was done we started to put together what we learned and start simulating. We started with modeling the traces as seen in Figure 7.

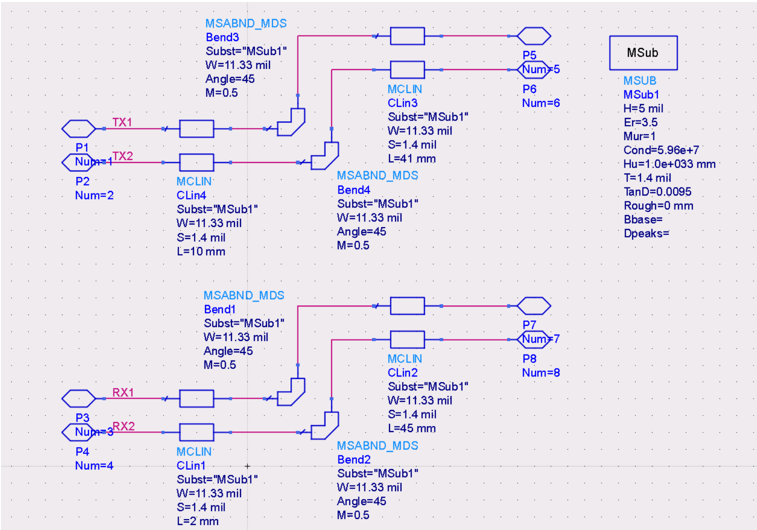

Figure 7: Modeling Microstrip Traces on PCB

After doing some calculations we came up with a rough estimate of what our characteristic impedance was, and we found the rest of the numbers from various data sheets.

We created a model for these 4 traces (2 Rx lines and 2 Tx lines) and then inputted it into a testbench for USB 3.1 compliance within ADS.

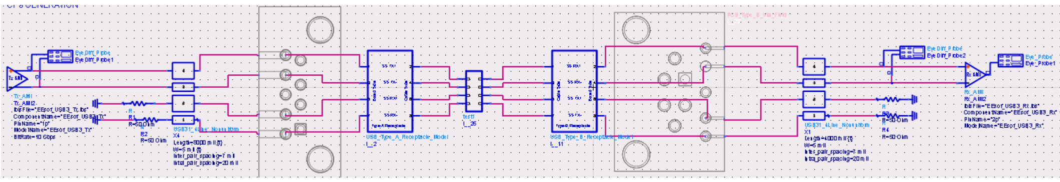

Figure 8: USB 3.1 Compliance testing testbench [17]

From there we were able to obtain the eye diagram shown in Figure 9.

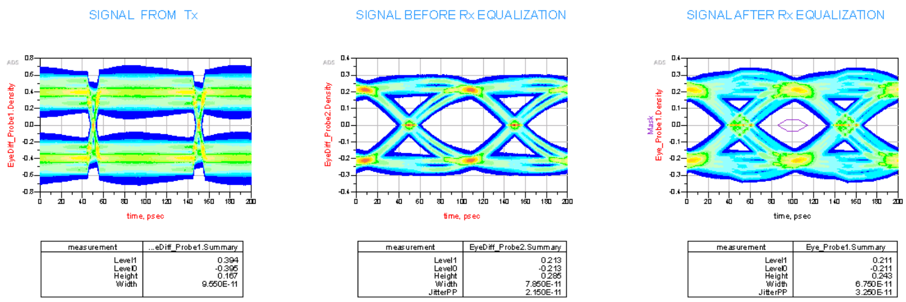

Figure 9: Eye Diagram Simulation Results

Observing at the eye diagram alone you can see that we meet compliance pretty easily, but more work can be done to make the eye bigger. Due to the equalization being a software component within the testbench we were using for USB compliance we get an equalized eye smaller than the unequalized eye [17]. For all intents and purposes we will ignore the equalized eye and pay attention only to the unequalized eye.

We set up our testbench for S-Parameter simulation.

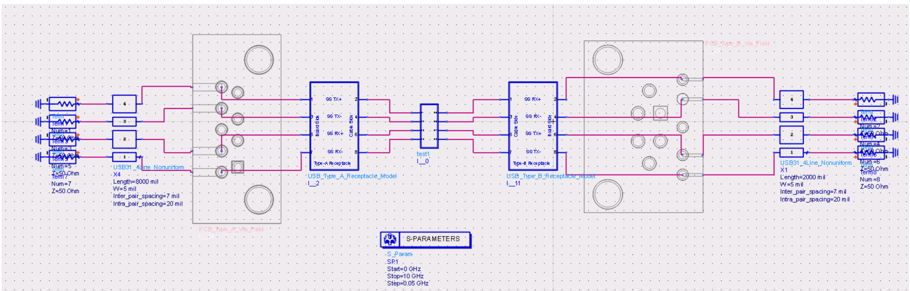

Figure 10: S-Parameter Testbench Schematic [17]

Looking at the S-Parameters of our differential signal we obtain the graphs shown in Figure 11.

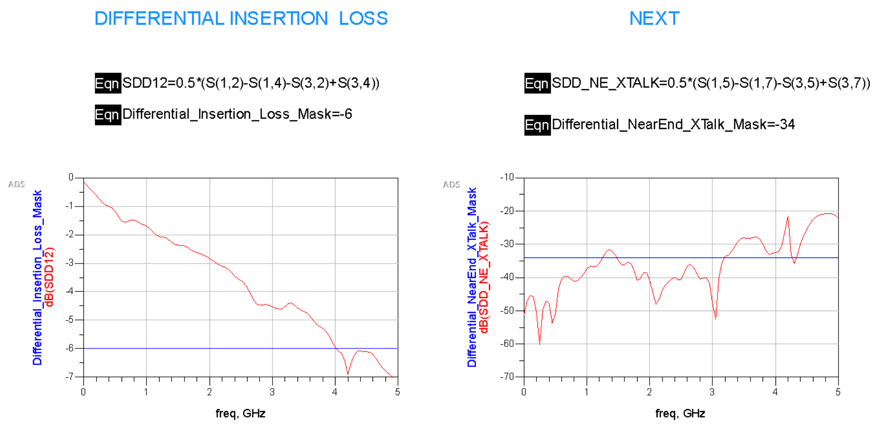

Figure 11: Differential S-Parameter Simulation Results

We pass both compliance tests for USB 3.1 here as well for both the insertion loss and near end cross talk. For differential insertion loss our lower limit should not pass -15dB. Our differential insertion loss falls above that limit. For near end crosstalk, between 100MHz and 2.5GHz, if we are lower than -32dB then we have passed [2].

We also ran simulations for Time Domain Reflectometry (TDR) although the results show that we must do more work to reduce impedance drops. Below is the testbench we designed for it:

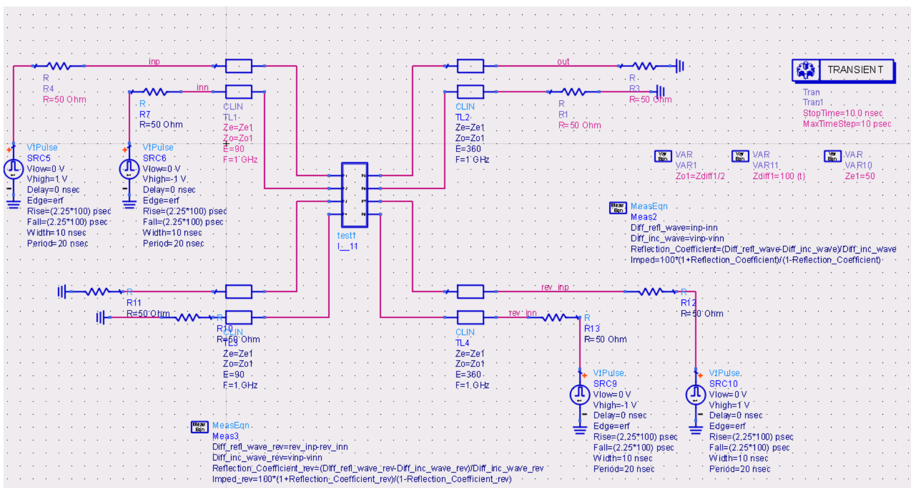

Figure 12: TDR testbench schematic [17]

Figure 13 shows the simulation results we obtained from the testbench:

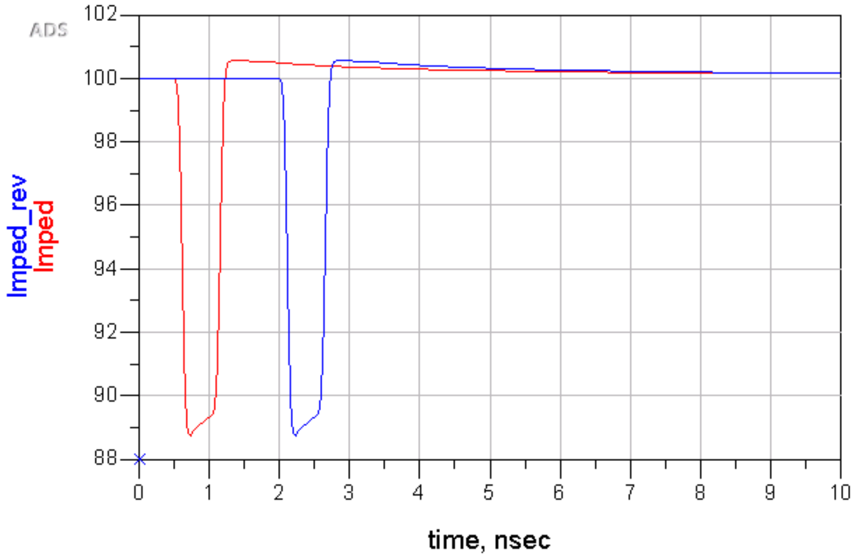

Figure 13: Time-Domain Reflectometer Simulation

The red line is the Tx line and the blue line is the Rx line. You can see an impedance dip which is most likely caused by bends within the traces as well as some impedance mismatch. The characteristic impedance was previously calculated by hand, so another revisit using ADS LineCalc is going to be a much better option.

5 Final Design

5.1 Finalized Design Choices

Starting with the actual layout, we stuck with using edge coupled microstrip lines to cut down on costs versus stripline. The PCB material we ended up going with was FR4. It was much cheaper and easier to source compared to GETEK, while being similar in terms of functionality. In our final design we were able to have all of our signal traces on the top layer, and PCB layout design progression will be shown. We ended up going with a four layer board, where the top layer is signal, the next two layers are ground, and then the last layer is for our control lines. Due to budget we had to cut out higher functionalities, such as the ability to access the Internet or bluetooth connectivity.

5.2 High Level Schematic

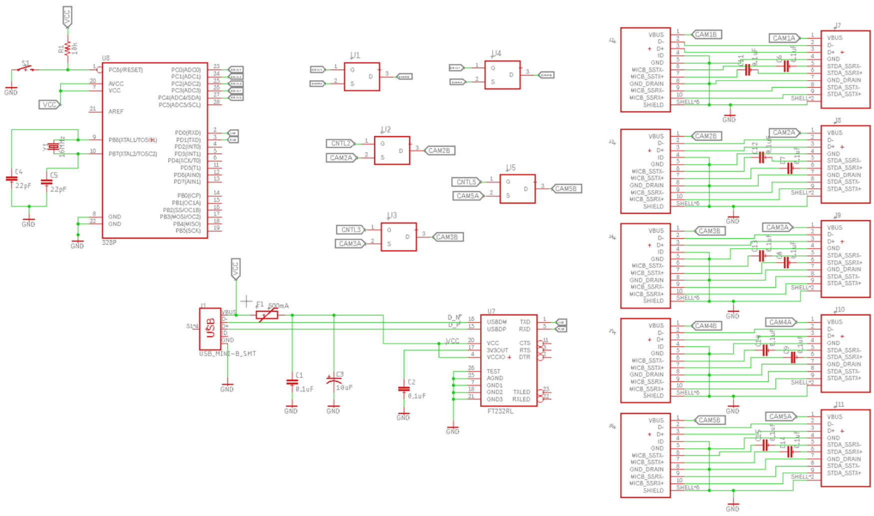

Figure 14: High Level Schematic

The main point of the schematic is to give us a central view of what our circuit would look like logically before we laid it down on the board. On the right hand side you see our traces go from our USB Type A connector to the USB Type B connector. In the middle you can see our MOSFETs which control the power going to the cameras. On the left hand side you can see the ATmega which control the MOSFETs like a switch. And on the bottom is how we talk to the ATmega using our computer.

5.3 First Iteration of Layout

The first layout iteration is shown in Figure 15. There were several different problems

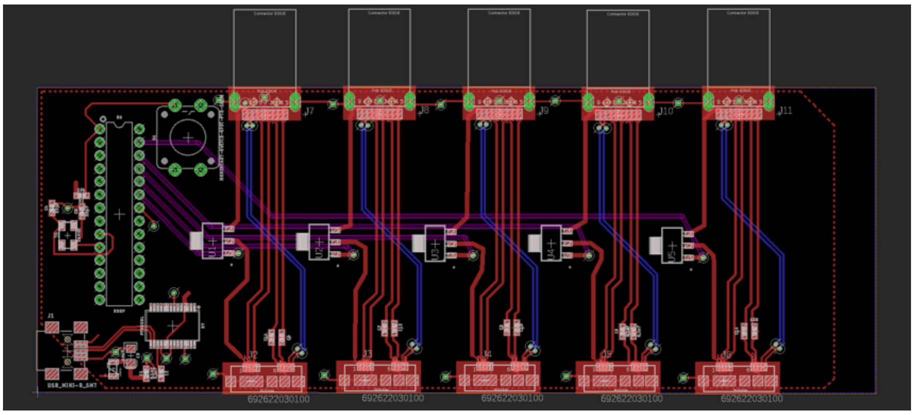

Figure 15: First Layout Iteration

with this iteration. The power plane was way too big. The power plane covered the entire board, but only a small section on the left hand side of the board needs power. There were also issues with the Rx signal having to cross over the Tx signal trace on a different layer due to the interconnects used. This signal crossing would cause a lot of interference because the Tx lines would use the Rx lines as a return path instead of the ground. It turns out that the wrong connector was used for the USB Type A. It was a male connector instead of a female connector which caused the crossing of the trace issue.

The traces also ended up having matching issues. Once the signal was sent down the via we did not match the traces to the new reference distance on the second layer. The MOSFETs are also over sized by a considerable amount. They are rated for 600V, while we only need them for around 5V. Blind vias were used instead of through hole vias, and there were several other routing issues that would have caused problems in terms of manufacturing.

Stack up was: (1) Signal, (2) Signal, (3) Ground, (4) Power

5.4 Final Iteration of Layout

All issues that were pertinent in the first layout were fixed. Proper sized components were used, and we shrunk the power plane down dramatically. The connector was fixed, which gave us the ability to keep all of our signals on the top layer. The board

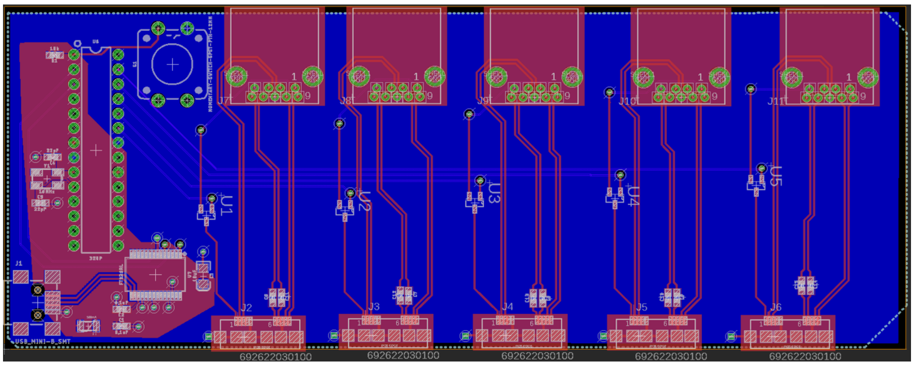

Figure 16: Final Layout Iteration

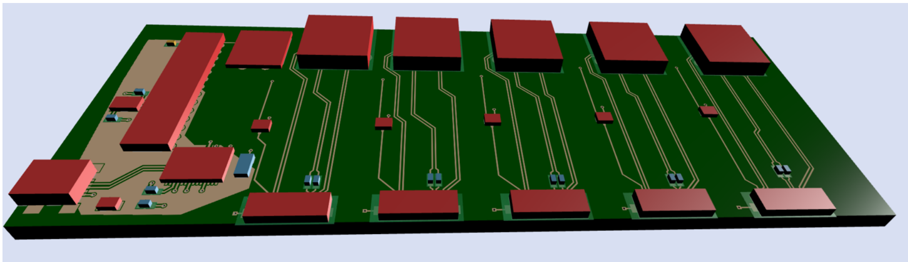

Figure 17: 3D Final Layout Iteration

was improved dramatically compared to the first iteration, and the improvement was easily seen within the simulations.

Stack up is: (1) Signal, (2) Ground, (3) Ground, (4) Signal

5.5 Layout Simulations

The two main simulations done for the layout design are S-Parameters and Eye Diagram. S-Parameters help us characterize the signal by showing us what the reflections and loss are within a system. In Figure 18 we see a simulation done of the Rx SuperSpeed trace on our layout.

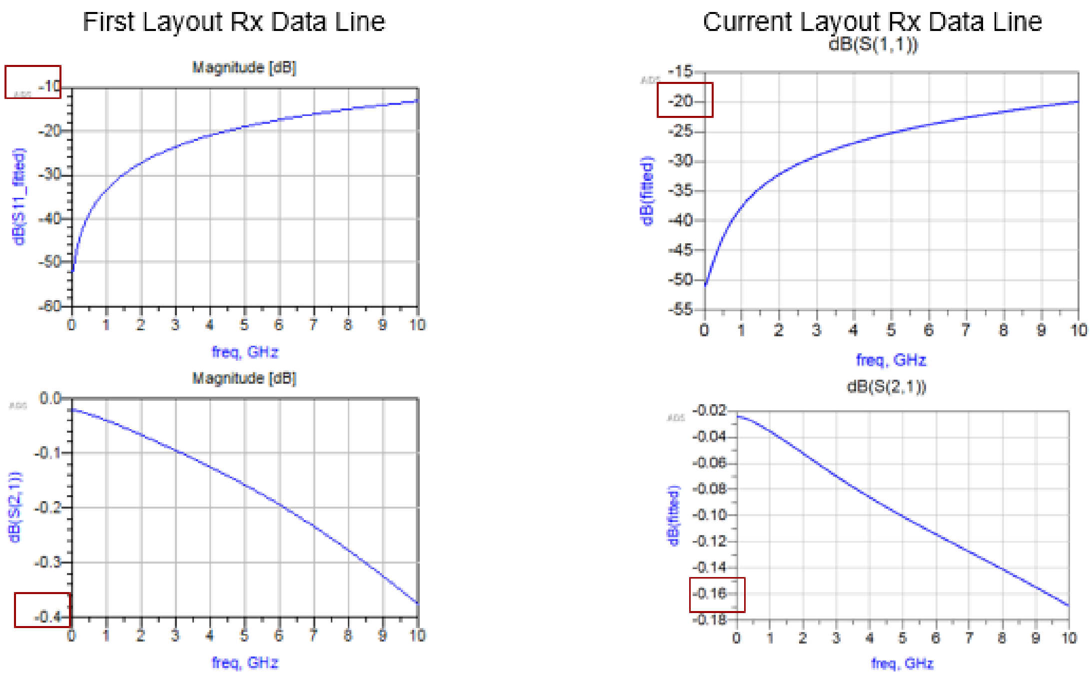

Figure 18: S Parameter Simulations

The top half of the S-Parameter chart shows us reflections while the lower half shows us transmission loss. As shown, in the first layout at 10GHz, our traces had about -10dB of signal reflection, while in our current layout we have -20dB. That is an improvement of 10dB, which on a log scale is a tremendous increase. For our transmission loss, our first iteration had about -.4dB of power loss, while our current iteration has only -.16db of power loss.

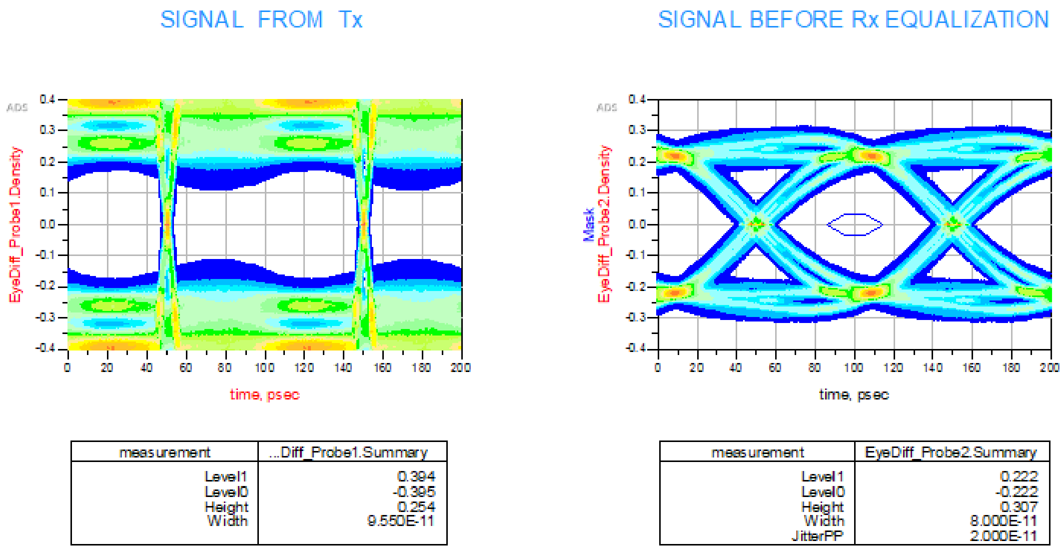

Figure 19: Layout Eye diagram

We were able to take the data generated by the S-Parameter simulation and generate an eye diagram from it. As you can see our eye is wide open and the signal that we inputted is the signal that we outputted.

6 Testing

There are two important components to the testing phase of the RealSense Relay board. Due to budget constraints and cost of manufacturing the first round of testing was done via simulation tools in ADS software, this testing was performed before the CAD files were sent of to manufacturing. As previously mentioned all simulation results in ADS displayed USB 3.1 compliance at around 5Ghz which is the operating point of RealSense cameras. With these results reinforcing our resolve we sent the CAD files to manufacturing.

The second and most critical phase of testing was a real time functionality test. Once the boards were completed and received form the PCB fabrication house it was time to perform a functionality test. This functionality test had several layers to it. The three major blocks that had to be functioning were:

6.1 Logic controller (ATMEGA)

A controlling machine, once connected to the board via USB 2.0 had to be able to communicate with the ATMEGA and by doing so, control the on/off state of each camera. This was achieved by writing some code using the C programming language and downloading it onto the ATMEGA. Once a device has connected via serial (using just a terminal) commands could be sent and received. If a “1” was typed into the terminal camera 1 would be turned on and a corresponding message was sent to the screen displaying the text “Cam 1: ON”. If the value “1” was entered into the terminal camera 1 would be shut off and the terminal would display “Cam 1: Off”.

The first order of business was to make sure that the ATMEGA was functioning as required. This was done by opening a terminal connection to the board, sending commands, and testing the mosfet voltage with a Digital Multimeter (DMM). When a “1” was entered into the terminal the camera should be on, and the DMM should display a voltage of approximately 3.7. This is due to the 1.3 V drop across the Mosfet. The RealSense cameras require an input voltage of 3.3V. Physical testing revealed that this block of operation was functioning as desired. Values of “1”- “5” entered into the terminal did indeed turn on/off cameras 1-5 as expected.

6.2 Signal integrity

The next phase of testing, and a rather critical one, requires some very expensive equipment that we are not able to acquire. Ideally we want to perform an eye diagram test of our physical board and compare the result with both the compliance requirements of USB 3.1 and our simulation results. Unfortunately these types of scopes are very expensive and we do not have them at Santa Clara University. Our attempts at working with local companies in hopes of using their equipment failed. In absence of this testing option we could only perform a physical functionality test.

6.3 Physical Functionality Test

After plugging the RealSense opening a serial connection the the RealSense relay board and connecting the RealSense camera as required it became clear the the signal was not passing through the PCB as desired. The power control was working, and with the aid of a DMM it was confirmed that the required voltage and current were making it to the camera. The problem is in the signal itself and without the use of a high powered scope it is difficult to say exactly what the issue is. We later learned it was the capacitors added in series on our Tx lines.

6.4 Testing Conclusions

The simulations performed in the ADS software tool passed compliance and the eye diagram appeared sufficient. However, there is a big gap between simulation results and actual physical results. The issue has been isolated to the transmission of the signal from port A to port B through the PCB. To isolate the issue further we tested a USB 2.0 device which performed as expected. This narrows the issue down ever further to just the SuperSpeed data lines (Tx/Rx). What we learned was that the capacitors added to the Tx lines of our board were causing the issues. Our theory is that the cameras already have capacitance within them for the Tx lines, and by adding more in series we effectively reduced the capacitance to below the operations value. Once we removed the capacitors on our board and used a solder blob to bridge the pads the cameras worked as expected, switching on and off when we wanted them to.

7 University Core Requirements

In the below sections an explanation is made for how our project hits the requirements of environmental, social, economic sustainability as well as ethics and artwork.

7.1 Ethics

In today's world, there is a surge of entrepreneurial spirit that is undeniable. As technology improves the average person is able to get their hands on the type of computing power that was previously restricted to niche corporations with the financial fortitude to afford such power. The motivation behind the RealSense relay is one rooted in the world of the hobbyist. The brightest and most powerful technologies that are dominant today all had humble beginnings, often starting in the garage of a hobbyist. Three dimensional cameras and virtual reality in general are positioning themselves to be incredibly important, including tasks ranging from gesture/facial recognition to medical applications such as stroke detection. It is the goal of the RealSense relay to allow a typical garage hobbyist to interface and control multiple RealSense cameras for any application they see fit.

The current issue is that these new generation three dimensional cameras are so bandwidth heavy that they do not operate under the typical USB 2.0 standards, they require high speed signals transmission with USB 3.0. These types of signals are very susceptible to noise and losses in the system. The average garage hobbyist does not have access to the testing equipment and design tools required to interface with these high-speed cameras. It is the goal of RealSense relay to provide a board that will allow for the transmission of high speed signals through a printed circuit board. This would allow the hobbyist to control the on/off state of each camera separately opening the door to countless applications. A fully functioning RealSense Relay will give the hobbyist the freedom to unleash their imagination and not be restricted by lack of resources.

7.2 Character of an Engineer

The biggest take away from our project is that the character of an engineer is not something you can label and put in a box. In fact, it is those types of assumptions that contribute to the lack of diversity in engineering as a discipline. While there are many different types of personalities and characteristics that are attributable to engineers there is only one common theme. All engineers must have curiosity, curiosity to ask questions, challenge the status quo and not be afraid to dream up big ideas. One good idea can create a whole ecosystem of potential jobs and new avenues to explore. RealSense cameras for example have opened the door to applications and jobs in both the gaming and medical fields. Applications are important when considering a new project, and can even be unknown at the beginning. The RealSense relay has an application in mind but the important thing as an engineer is to keep it open source and allow for hackability. In other words, we want people to be able to take our board and with a little bit of work adapt it to perform a function that we did not or could not think of. This is the key to making ethical products that not only satisfy a need but satisfy future unknown needs.

7.3 Ethical Pitfalls

In designing a piece of technology that interfaces with cameras all of the same ethical pitfalls that exist with cameras are absorbed into our domain. There is no way to guarantee that these cameras in conjunction with our RealSense relay will be used to do good. With the ubiquitous nature of these cameras and the ability to fit them into small packages, the issue of privacy becomes a concern. By allowing the control and interface of multiple cameras we are in fact giving anyone who wishes to use these cameras in a negative way, such as violating the privacy of someone else, more capability. Without monitoring the application and use of the RealSense board there is no way to ensure that the applications are ethical. Monitoring the applications is unethical in and of itself. This is a situation where we cannot avoid any ethical malpractices without ourselves being unethical, not to mention the impossible feat of actually monitoring ever application. For the aforementioned reasons, we are hoping and imploring that our RealSense relays are used in an ethical fashion.

There is another side to the ethical dilemma, and that is the actual functionality of the board. Whenever power is involved there is a risk of certain components overheating and causing damage to the surrounding environment. A RealSense has no restrictions as to where it can be placed and if any component gets too much current or exceeds its maximum power rating there is a concern of fire. A potential application for the RealSense board is at gas stations above each gas pump. The board will control multiple cameras at multiple pumps and monitor the activity at each pump. If one of the components in the board overheats and creates spark there is a potential for catastrophe. As engineers it is our responsibility to ensure that we have selected components that are made by a reputable manufacturer and that they are operating well within the safety thresholds of current and voltage.

7.4 Civic Engagement

Technology undoubtedly touched the lives of millions across the globe. Living and going to school in Santa Clara, the heart of silicon valley has put us at the center of the technical revolution we are currents experiencing. New ideas are celebrated and often funded by venture capitalists taking their own risks. Technical devices of all types are ubiquotes and our interaction with them as humans is growing exponentially. Our fundamental believe is in freedom, freedom to test and experiment with new technology. The ability to tinker and experiment with new technology provides the possibility for new unique discoveries that not only an individual or group of individuals can benefit from, but the whole world.

With the development of better technology bandwidth and data rates increase accordingly. This means what used to be a simple breadboard project now becomes something more advanced. The goal of our project was not only to meet the requirements of RealSense cameras but also to create an affordable board that they average hobbyist in our community and around the world could use to make their projects easier, more affordable and ultimately possible.

In our own process we discovered that USB 3.0 testing devices are not available at our school. It is our sincere hope that this is not the case for much longer. Indeed it cannot be, with higher data rates becoming the norm important testing will not be possible if Santa Clara University does not upgrade some of its testing equipment. Indeed, when our final product did not work we were not able to analyze the signal in a meaningful way. An enthusiastic high school student who wants to experiment with a new camera using USB 3.0 protocol will be out of luck. It is our hope that our experience with USB 3.0 will encourage the political powers of Santa Clara University to consider acquiring high speed testing equipment to benefit the students and the potential millions of people their projects and discoveries can help.

We believe that our project aims to fix this by making interfacing with high speed USB devices possible, affordable and accessible. As you read through the thesis you will find several artistic mock ups during the planning phases of this project.

8 Conclusion

Through this project we learned a lot about all the different components within our system. Not only did we have to navigate technical details, but there were a lot of social aspects that we did not consider when initially working on this. Overall we were able to solve the problem of the freezing cameras as well as improve the quality of the videos due to the of the videos due to the emphasis on signal integrity. For further continuity we can solder a small zero ohm resistor to bridge the gap of the surface mount capacitors on the Rx lines. The courses that helped out the most in this project were: ELEN 105 (Basics of Transmission Lines), ELEN 706 (more in depth of transmission lines), ELEN 624 (Signal Integrity), and ELEN 115 (How we ended up doing the switching).

9 References

[1] “Circuit Board Parts - PCB Print,” Bay Area Circuits, 10-Jan-2017. [Online]. Available: https://bayareacircuits.com/cost-determining-factors-to-consider-when-quoting-pcbs-part-i/. [Accessed: 07-Dec-2017].

[2] “Electrical Compliance Test Specification Enhanced SuperSpeed Universal Serial Bus,” Promotor group.

[3] “High Performance Materials for Advanced Design,” Advanced Circuits, May 2013.

[4] “High Speed Interface Layout Guidelines,” Texas Instruments, vol. App Note, Aug. 2014.

[5] “High Speed Layout Design Guidelines,” Feescale Semiconductor, vol. App Note.

[6] “How to Correlate USB Type-C Simulation and Measurement,” Keysight, Mar-2016. [Online]. Available: http://literature.cdn.keysight.com/litweb/pdf/5992-1391EN.pdf?id=2727680. [Accessed: 07-Dec-2017].

[7] J. Coonrod, “Selecting PCB Materials For High-Speed Digital Circuits,” Microwave Journal, 22-Jan-2015. [Online]. Available: http://www.microwavejournal.com/blogs/1-rog-blog/post/23743-selecting-pcb-materials-for-high-speed-digital-circuits.

[Accessed: 07-Dec-2017].

[8] Jeong Tae - Jong, . Jeong Tae-Jong, “Board Design & Simulation with Recent Enhancements EEsof ADS Signal and Power Integrity,” Board Design & Simulation with Recent Enhancements EEsof ADS Signal and Power Integrity.

[9] M. S. Sharawi, “Practical Issues in High Speed PCB design,” Portland State University, 2004. Object Tracking [F200,SR300]. [Online]. Available: https://software.intel.com/sites/landingpage/realsense/camera-sdk/v1.1/documentation/html/doc_ot_object_tracking.html. [Accessed: 07-Dec-2017].

[10] [Online]. Available: http://wcalc.sourceforge.net/microstrip.png. “PRINTED CIRCUIT BOARD (PCB) DESIGN ISSUES ,” Printer Circuit Board Issues.

[11] “S-parameters: Signal Integrity Analysis in the Blink of an Eye,” Signal Integrity Journal RSS. [Online]. Available: https://www.signalintegrityjournal.com/articles/432-s-parameters-signal-integrity-analysis-in-the-blink-of-an-eye. [Accessed: 07-Dec-2017].

[12] “Super Speed USB with Type - C Connector,” Keysight Technologies. T. U. Editor, “microwave RF information for engineers encyclopedia calculators tools,” Microwaves101. [Online]. Available: https://www.microwaves101.com/encyclopedias/

microstrip. [Accessed: 07-Dec-2017].

[13] “Transmission Lines and Characteristics,” Texas Instruments, vol. App Note, pp. 1–9, May 2004. “USB 3.1 Compliance Test Bench,” Keysight, vol. software, Jan. 2016.

[14] Juvtmall.com. (2018). 8. Stripline vs Microstrip: Understanding Their Differences and Their PCB Routing Guidelines,PCB Design. [online] Available at: http://juvtmall.com/wiki/8-stripline-vs-microstrip-understanding-their-differences-and-their-pcb-routing-guidelines_i0115.html [Accessed 27 Nov. 2017].

[15] Pozar, D. (2012). Microwave engineering. Hoboken, NJ: Wiley.

[16 ARTICLES, T., PRODUCTS, N., ELECTRONICS, G., PROJECTS, C., MICRO, E., Lectures, V., Webinars, I., Training, I., Search, P., DB, T., Tool, B. and Search, D. (2018). Edge Coupled Microstrip Impedance Calculator. [online] Allaboutcircuits.com. Available at: https://www.allaboutcircuits.com/tools/edge-coupled-microstrip-impedance-calculator/ [Accessed 27 May 2018].]

[17] Edadownload.software.keysight.com. (2018). USB 3.1 Compliance Test Bench. [online] Available at: http://edadownload.software.keysight.com/eedl/ads/ctb/USB3p1_Compliance_
[Accessed 27 Sep. 2017].

10 Appendix

10.1 Appendix A: Gantt Chart

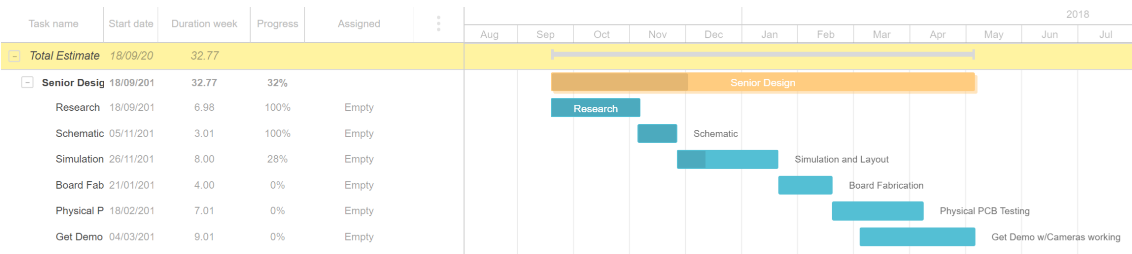

Figure 20: Gantt Chart for Senior Design

Our original plan was a lot more optimistic than we had hoped. There were a lot of delays that we had not considered when making the original gantt chart. Our board took much longer to manufacture, and learning how to use the CAD tools took much longer than anticipated. We ended up receiving our board post-presentation, so physical testing still has to be done.

10.2 Appendix B: ATMEGA Programming

```
void setup() {

Serial.begin(9600);

pinMode(A0, OUTPUT);
pinMode(A1, OUTPUT);
pinMode(A2, OUTPUT);
pinMode(A3, OUTPUT);
pinMode(A4, OUTPUT);
}

int count1=1;
int count2=1;
int count3=1;
int count4=1;
int count5=1;

void loop() {
  int cam = Serial.read();

  if (cam == 49){
    count1++;
    if((count1 % 2) == 0){
      digitalWrite(A0, HIGH);
      Serial.println("Cam 1: On");
    }
    else{
    digitalWrite(A0, LOW);
    Serial.println("Cam 1: Off");
    }
  }

  if (cam == 50){
    count2++;
    if((count2 % 2) == 0){
      digitalWrite(A1, HIGH);
      Serial.println("Cam 2: On");
    }
    else{
```

```
    digitalWrite(A1, LOW);
    Serial.println("Cam␣2:␣Off");
    }
  }

  if (cam == 51){
    count3++;
    if((count3 % 2) == 0){
      digitalWrite(A2, HIGH);
      Serial.println("Cam␣3:␣On");
    }
    else{
    digitalWrite(A2, LOW);
    Serial.println("Cam␣3:␣Off");
    }
  }

  if (cam == 52){
    count4++;
    if((count4 % 2) == 0){
      digitalWrite(A3, HIGH);
      Serial.println("Cam␣4:␣On");
    }
    else{
    digitalWrite(A3, LOW);
    Serial.println("Cam␣4:␣Off");
    }
  }
  if (cam == 53){
    count5++;
    if((count5 % 2) == 0){
      digitalWrite(A4, HIGH);
      Serial.println("Cam␣5:␣On");
    }
    else{
    digitalWrite(A4, LOW);
    Serial.println("Cam␣5:␣Off");
    }
  }
}
```