ELEC3397

University Of New Haven

Junior Design:
LIFI Project

Written by Jamal Bouajjaj

Project by Jamal Bouajjaj and Salah LaTulippe

2021-05-08

Abstract

This project's goal is to use visible light in order to transmit data instead of radio waves. This can be useful in scenarios where visible light already exists, such as in vehicles or house lightbulbs, and where radio interference is not ideal.

Contents

1 Intro

The purpose of this project, as discussed in the Abstract, is to transmit data over a visible light medium instead of radio waves. This concept is already a thing, and it's called LIFI. LIFI is an attempt at replacing or substituting WIFI with light instead of radio waves.

Our job during the semester was to look into the feasibility of it as well as getting a working transmitter/receiver model, given our limited experience.

1.1 Terminology

Throughout the rest of this document, some terminology will be used that are explained bellow:

- TIA: Trans-Impedance Amplifier. An amplifier that takes in current and transforms it into voltage.
- OOK: On-Off-Keying. A modulation technique where a 1 is represented by a carrier wave, and a 0 being the lack of said carrier wave.
- LIFI: A technology that attempts at using light as a medium instead of radio waves for communication.
- Jellybean X: Where X is something like a comparator, the jellybean part of the word simply means common.
- UART: A asynchronous communication protocol used throughout this project due to it's simplicity and it's one signal per receive and transmit signal.

1.2 Our OSI Model

The OSI model is a model used to characterize modern communication, including the internet as a whole. As the goal of LIFI is to send internet data, it is appropriate to see our project as working on the physical layer of the OSI model. The physical layer is the layer that handles defining that is a bit/symbol, and how it's transmitted. If we had time, we also would be tasked with potentially defining the Data Link layer, the layer that handle defining data frames.

The graphic bellow (Figure 1) shows the OSI model, as well as a breakout of the physical layer (what we focused on for this project):

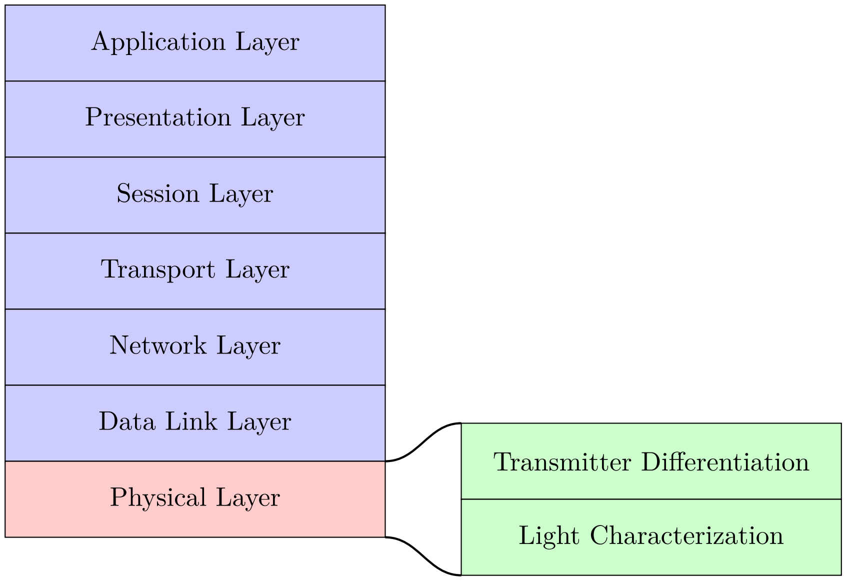


Figure 1: Our OSI Model representation

The Light characterization layer is defining how to convert photons into data and how to convert data into photons. The methodology used to accomplish that is described in each circuit's section.

The Transmitter Differentiation layer is how to differentiate one transmitter from another, assuming simultaneous communication. Out of the 3 circuits built, only the *OOK Transceiver* as this feature, the other 2 circuit do not.

1.3 Circuits

Initially only 1 circuit was built, but as the semester's ending was approaching, we built 2 other circuits throughout the semster to showcase a working model, as well as to characterize using light as a medium for one of the circuit. The 3 circuits built are:

- **OOK Transceiver Design:** This was the first circuit we set out to build, and what took the majority of our time
- **Jellybean IR Receiver Design:** A circuit using a common/jellybean IR receiver module to test out how well that will work for transmitting data across
- **Simple TIA+Comparator Design:** Built towards the end of the semester to get a working model of using light as a medium for communication. Also used to characterize transmission distance, transmission speed, and receiver sensitivity

2 Jellybean IR Receiver Design

2.1 Overview

This design was by far the simplest. It's purpose was to determine how well a generic IR receiver module, which are designed to receive data from a remote control, will work with sending data across, both with an IR LED and a white LED. UART will be used as the data that will be sent across of it. The model of the IR receiver is a TL1838IR, which in it's internal block diagram is what the *OOK Transceiver Design* is trying to accomplish, but at a much lower carrier frequency of $37kHz$, as shown in Figure 3.

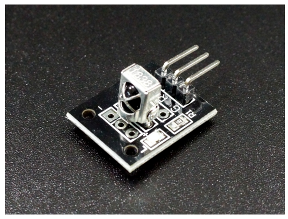


Figure 2: Generic IR Receiver

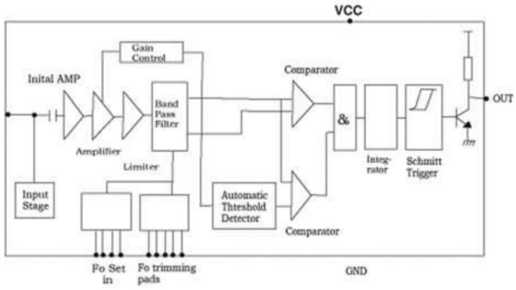


Figure 3: Internal Block Diagram of the TL1838IR Receiver Module

2.2 Schematic and Setup

The circuit for this design, shown in Figure 4, is a simple LED controlled by a UART TX pin on the transmitter side, and the infrared receiver module's output pin connected to the UART RX pin.

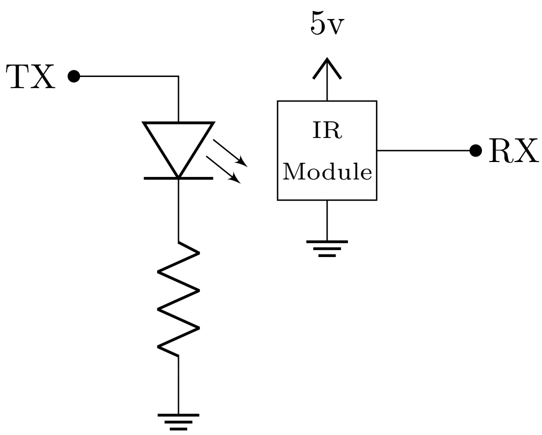


Figure 4: Schematic of the Jellybean IR receiver circuit

A picture of the setup is shown bellow:

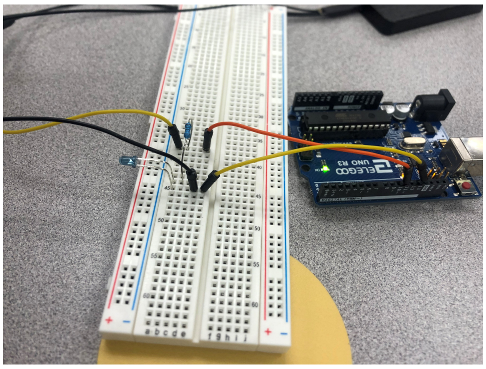


Figure 5: Transmitter Setup

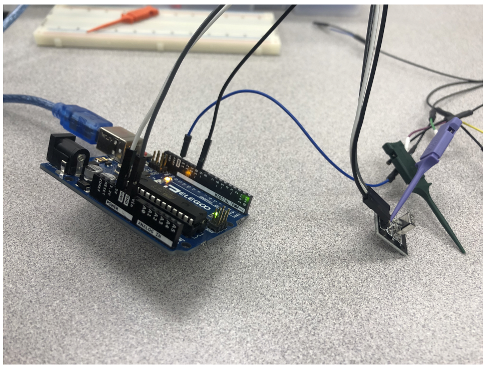
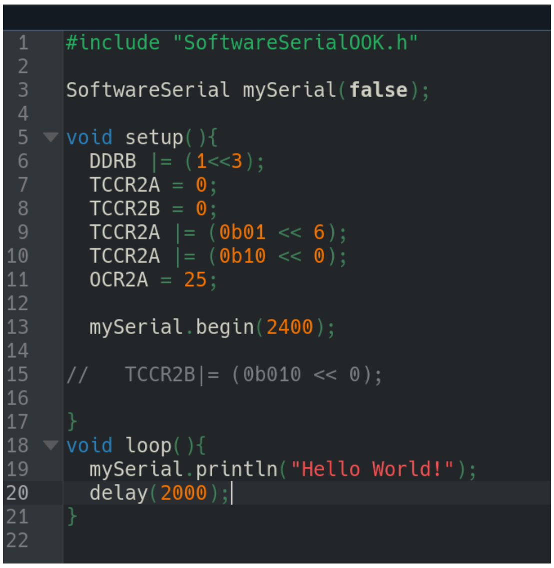


Figure 6: Receiver Setup

2.3 Software

As the IR receiver expects a $37 - 38kHz$ carrier wave before it toggle the output on, a modified *SoftwareSerial* library was used, as a hardware serial would not be possible due to the required modulation. The *SoftwareSerial* library was modified so that instead of toggling a pin *HIGH* or *LOW* when sending out data, instead it would turn on *Timer2* inside the ATmega328 IC, which is setup to toggle the LED pin at $37kHz$. This results in a UART link where the transmitter actually toggles the LED at the carrier frequency in order to send a one. Pictures of the main program and the modified *SoftwareSerial* library is shown in Figures 7 and 8



```
#include "SoftwareSerialOOK.h"

SoftwareSerial mySerial(false);

void setup(){
  DDRB |= (1<<3);
  TCCR2A = 0;
  TCCR2B = 0;
  TCCR2A |= (0b01 << 6);
  TCCR2A |= (0b10 << 0);
  OCR2A = 25;

  mySerial.begin(2400);

//   TCCR2B|= (0b010 << 0);

}
void loop(){
  mySerial.println("Hello World!");
  delay(2000);
}
```

Figure 7: Main Program

```
void SoftwareSerial::tx_pin_write(uint8_t pin_state)
{
  if (pin_state == HIGH){
    TCCR2B &= ~(0b111 << 0);
    if(PINB & (1 << 3))
      TCCR2B |= (1 << FOC2A);
  }
  else {
    if((TCCR2B & 0b111) == 0){
      TCCR2B |= (0b010 << 0);
      TCNT2 = 0;
      }
  }
}

uint8_t SoftwareSerial::rx_pin_read()
{
//  return *_receivePortRegister & _receiveBitMask;/
    return 0;
}
```

Figure 8: SoftwareSerial Modification

2.4 Results

This design did work at very low baud rates. Looking at Figure 9, which is a logic analyzer capture of the TX and RX pin, it can be seen that there is a $300\mu S$ delay between when the LED start toggling and when the IR receiver toggles. This is, we believe, to be the primary reason behind the low transmission rate. Another issue was that with a white LED, the range was only a couple of centimeters while an IR LED had orders of magnitude the range. This is because the IR module is optimized for usage in the infrared spectrum, not visible light as can be seen by the IR filter on the module.

Figure 9: Transmitter and Receiver Delay

3 OOK Transceiver Design

3.1 Overview

This design was the first circuits made, as well as the one taking up the most amount of time. It's purpose is to be a fully functional transceiver capable of differentiating different transmitters thru the use of the OOK modulation scheme.

3.2 Schematic

The schematic for this design (Figure 10) is more complicated than the other 2 designs, but it isn't too difficult to understand.

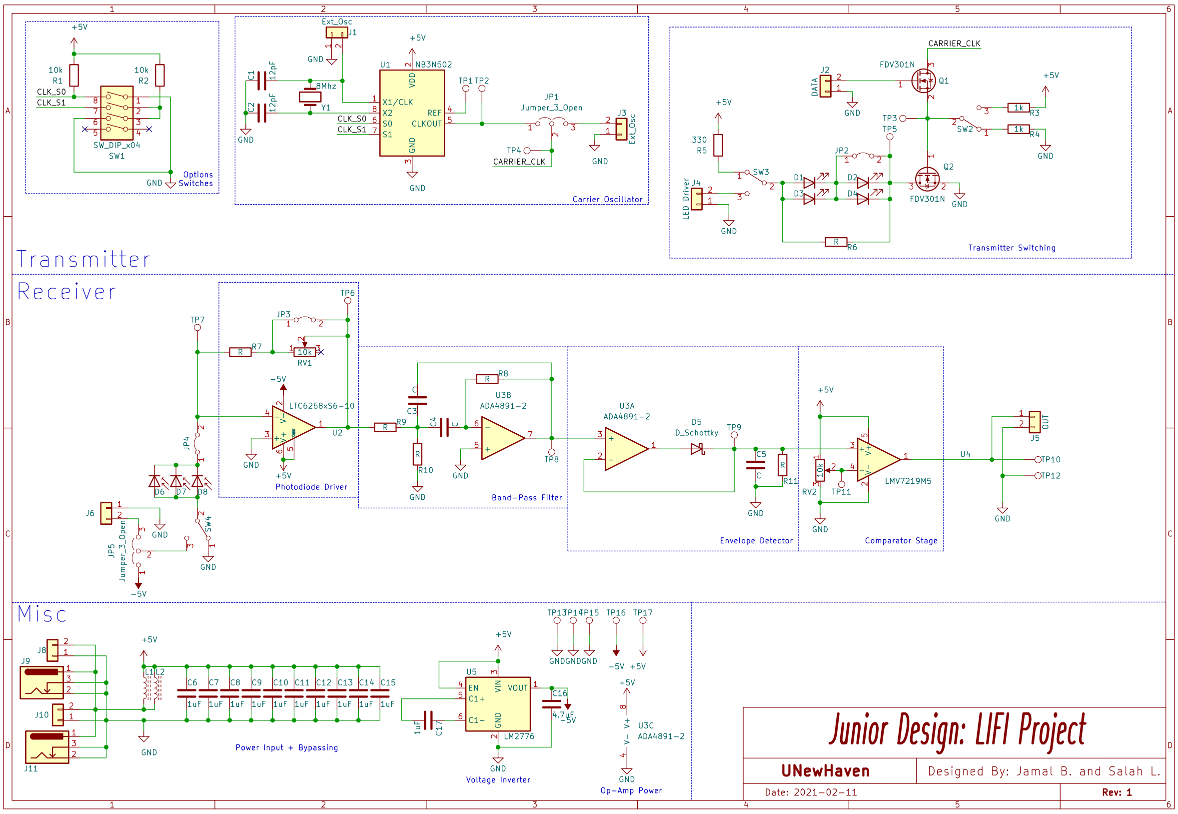


Figure 10: Schematic of the OOK Transceiver Design

3.3 Theory of Operations

For the transmitter side, it's a 2 transistor configuration so that a carrier wave can be fed in and a data signal, and it will modulate it to an OOK scheme. This OOK signal then toggles the LED. There is a clock multiplier on the transmitter side, but that never got used as it's minimum frequency is $14Mhz$, and we were struggling even with $1Mhz$.

For the receiver side, the photodiode and TIA convert the light level into votlage signal. The op-amp chosen for the TIA (LTC6268) was picked as it offered low input bias current. The lower the input bias current, the more that can go thru the feedback resistor thus a higher sensitivity. This signal then gets fed into a 2^{nd} order band-pass filter in order to differentiate the different potential transmitters. Then the signal goes thru an envelope detector, as we want to detect if there is a carrier wave or not. The envelope detector follows the contour of the signal up to a certain amount of time.

Finally, this gets compared to convert the analog voltage from the envelope detector into a digital signal.

There is also miscellaneous bypassing and filtering, as well as an inverting charge-pump converter on the circuit.

3.4 Simulations

Before ordering the PCBs, some simulations with the help of *LTSpice* was done in order to verify the component's design.

Band Pass Filter

In Figure 11, a simulated band-pass filter (the same configuration as the one in the schematic) with a center frequency of $1Mhz$ and a gain of 1 (what was soldered on the board later) is shown. As it can be seen, at the center frequency there is a gain of $0db$, and that gain steeply drops off as the frequency moves away from the $1Mhz$ center frequency.

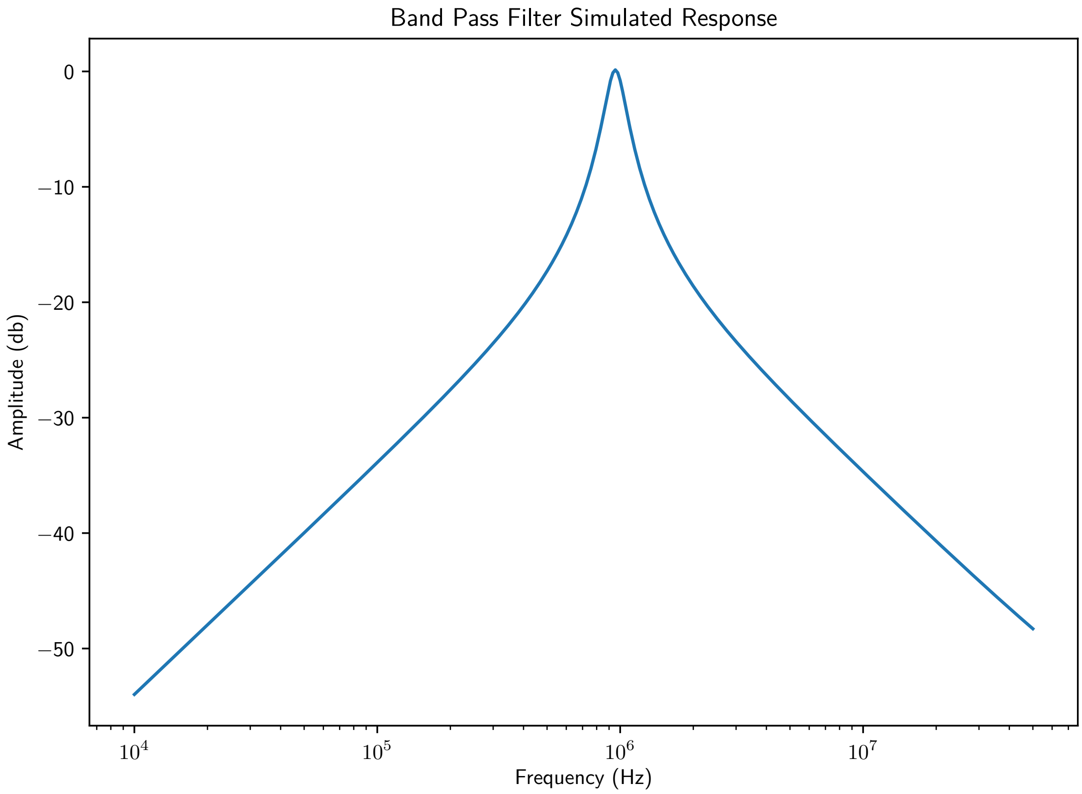


Figure 11: Simulation of the OOK circuit's band pass filter

Envelope Detector and Comparator

The envelope detector and comparator stage discussed in Section 3.3 has been simulated with an input carrier wave enabled for only 10 cycles, then turned off. The results are shown in Figure 12. There are 2 signals of interest: The output of the peak detector, and the output of the comparator. As it can be seen, as the capacitance on the envelope detector increases, the less of a dropoff between cycles exist but the longer it will take for the signal to drop off after the end of the carrier wave. The comparator results also demonstrates this, as with an increase the envelope detector's capacitance causes an increase in on time after the end of the carrier wave. This leads to a trade-off of speed versus reliability in ignore the mid-cycle dropoffs.

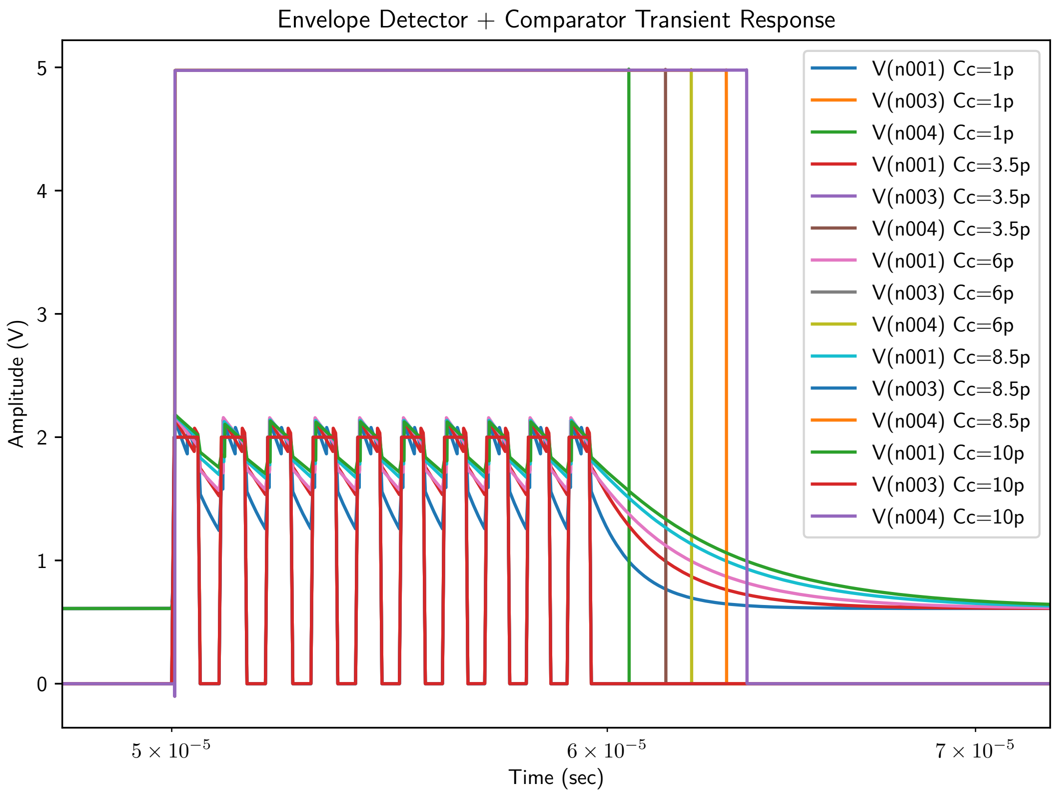


Figure 12: Simulation of the OOK circuit's envelope detector and comparator

3.5 TIA

The frequency response of the TIA has been simulated with different sensitivity (different feedback resistor), and that is shown in Figure 13. As it can be seen, with an increase in sensitivity (so an increase in feedback resistance), the low the bandwidth out of the TIA is, which is another compromise between speed and sensitivity that needs to be made.

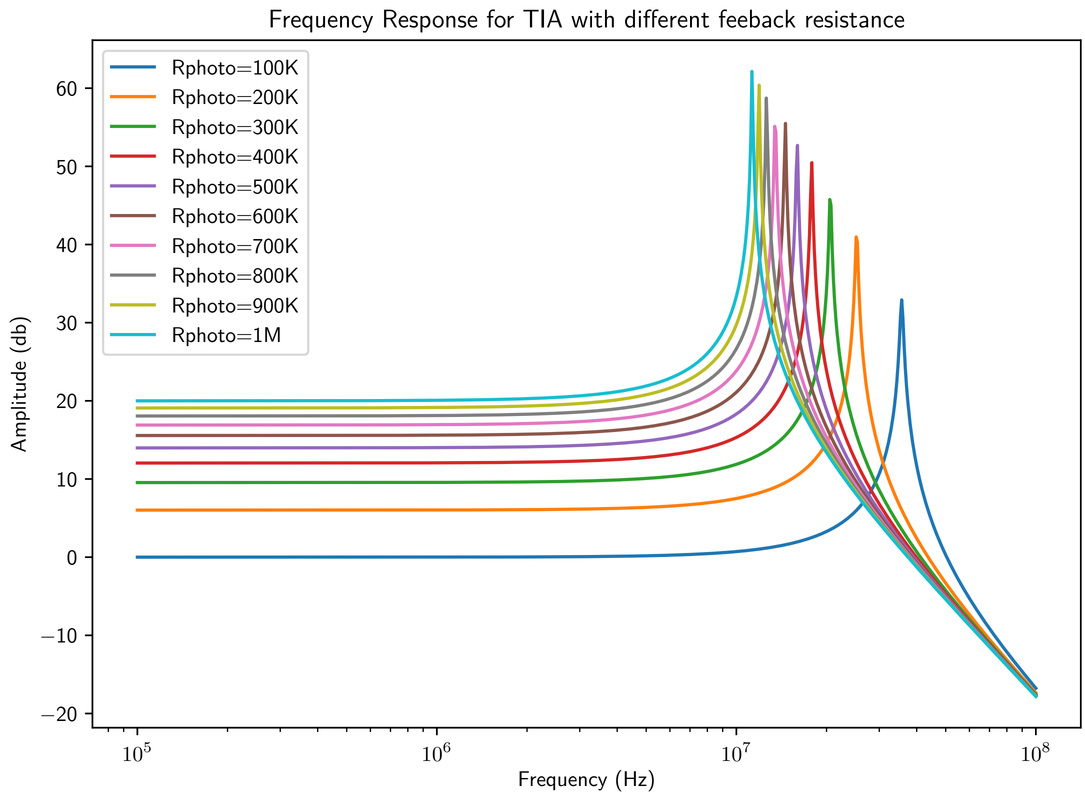


Figure 13: Simulation of the OOK circuit's TIA

3.6 PCBs

As this circuit was designed to be operated at a decently high carrier frequency, a low cost in getting a PCB made, as well as a lack of trust in breadboards in terms of their signal integrity (too much inductances and capacitances), it was decided to make a PCB for this circuit. This PCB is 2 circuits (the transmitter and receiver) made unto 1 board, allowing both to be split. A picture of a soldered transmitter and receiver is shown bellow in Figures 14 and 15:

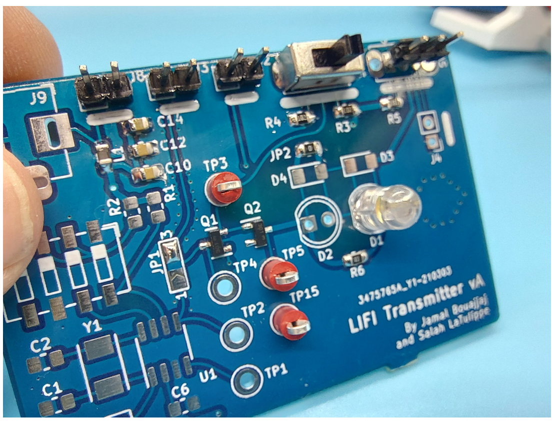


Figure 14: Transmitter side of the PCB

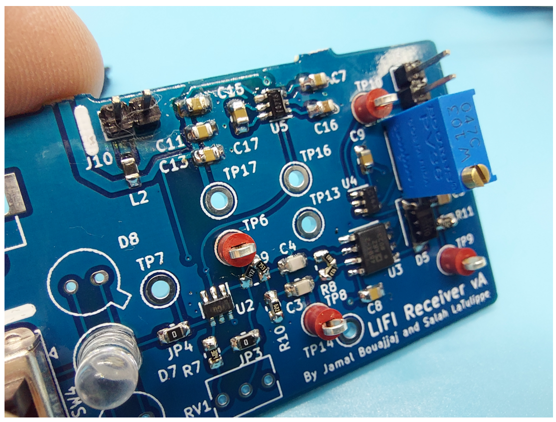


Figure 15: Receiver side of the PCB

3.7 Results

Unfortunately we were not able to get this circuit working before the end of the semester, due to the time spend on this design as well as multitudes of faults that exists in this design.

For the transmitter side, I ended up configuring the transistors incorrectly, as one of the MOSFETs (Q1) ended up being driving in reverse (a lower potential on the drain than source when the carrier clock signal is low), and MOSFETs have an internal reverse diode due to their construction. For this issue I ended up simply giving the OOK modulated signal directly to Q2 which drove the onboard LED.

On the receiver side, there were plenty of issues. To start out with, when I read the datasheet for the LTC6268 op-amp, I read the $5v$ supply spec as $\pm 5v$, whereas in reality the thing was only $5v$ supply rated (so a $\pm 2.5v$ for a dual supply). This caused multitude of issues, as the charge-pump convert I was using to create the negative supply had a minimum rating of $2.7v$, so an external $-2.5v$ needed to be fed in.

Initially I was trying to get away by only using a single supply for everything (by changing the LTC6268 op-amp's $-V_{cc}$ to GND), but that did not work. I did not further investigate this fault and decided to make the op-amp a dual supply operation.

Then the ADA4891 op-amp (for the band pass filter and envelop detector) did not work until I change it's supply for a single supply to a dual supply.

Then after all that, the TIA wasn't working. Before I spent too much time on that, I've decided to manually feed a signal into the rest of the circuit for testing. As shown in Figure 16, the band pass filter (purple channel) worked given a carrier wave (yellow signal) of $1Mhz$ (shifted by about $500kHz$, probably due to component tolerances). Sweeping the signal (not shown in this report) shows that the band pass filter's output decrease as the input frequency drifts away from the filter's center frequency, which show that the band-pass filter is actually filtering.

The envelope detector (light-blue channel) is also working, with the exception of a slight dropoff after the signal ends. This I have theorised to be cause by the diode's recovery time. Essentially if a diode's polarity is flipped, it doesn't immediately block current in the reverse direction but instead takes some amount of time before it starts doing so.

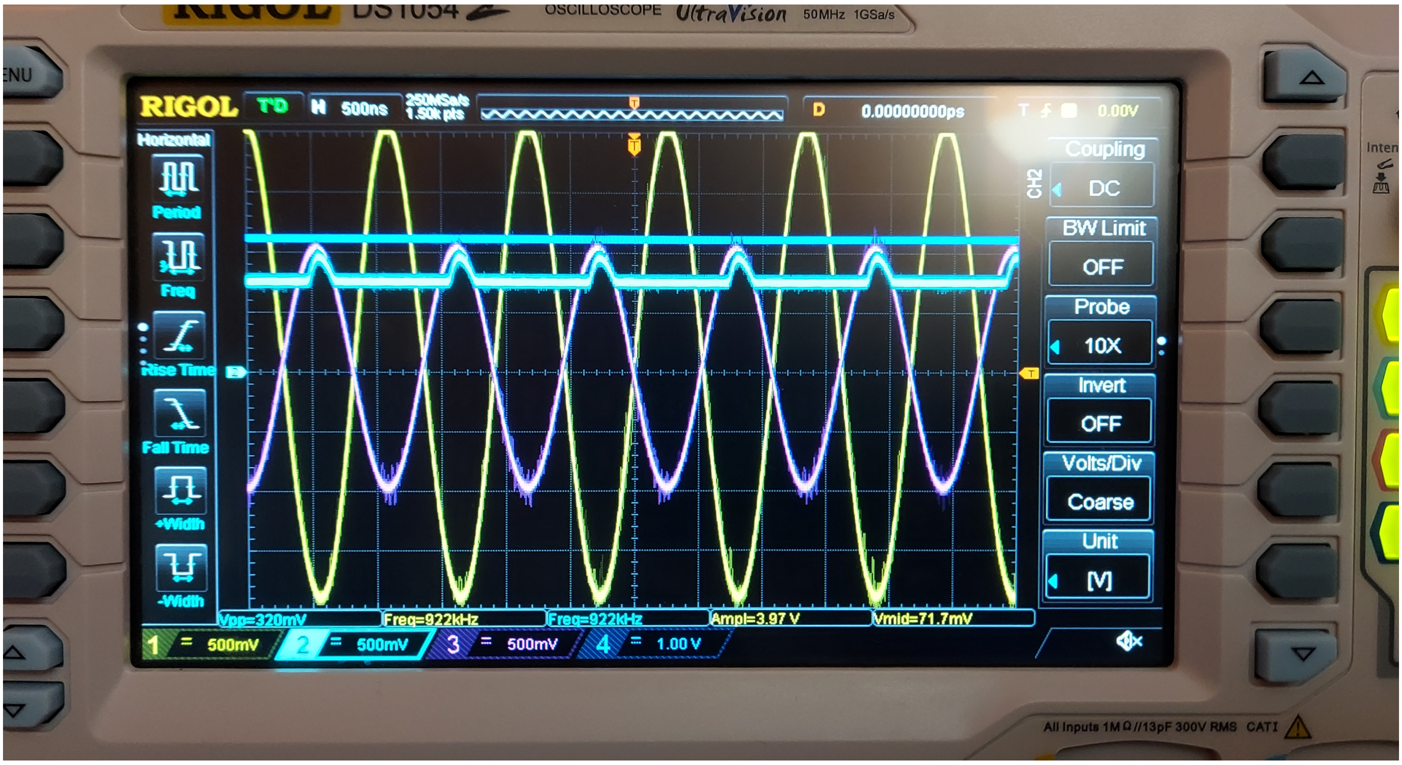


Figure 16: Scope picture of this design's band-pass filter and envelope detector

Then there is the comparator stage, which gave me the most trouble. While it seems simple, everytime the comparator switches it's output (for example after the carrier wave is turned off, thus the envelop detector's voltage decreases), it messes up the comparator by the addition of noise. A scope capture of that is shown in Figure 17, where the yellow trace is the output of the comparator and the pink signal is from the envelop detector.

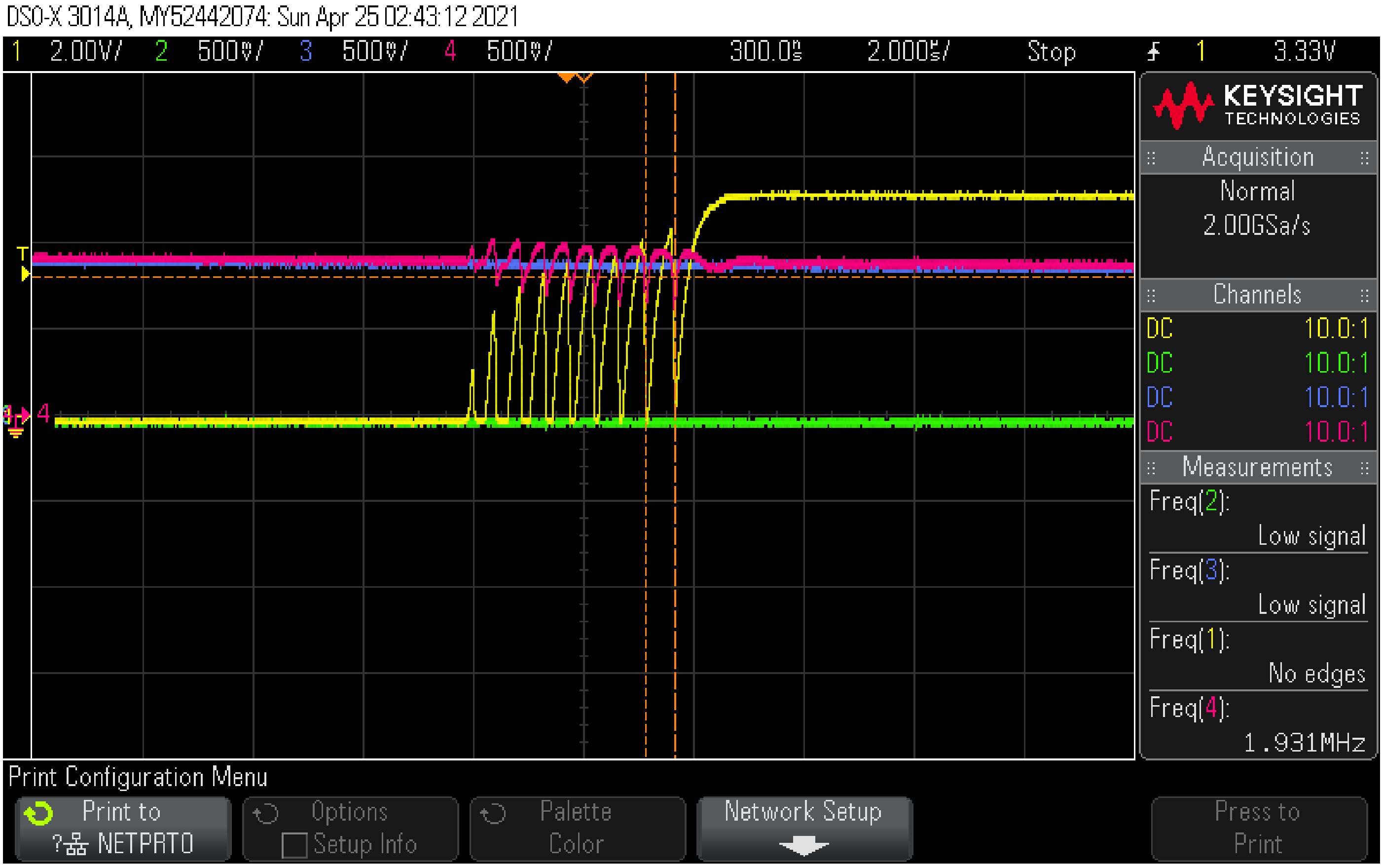


Figure 17: Scope picture of this design's envelope detector and comparator switching

Initially I tough it was some input independence problem, so I decided to not use the comparator on the PCB and opted for an external breadboarded comparator with a buffer (to buffer the envelop detector which is output current sensitive from the buffer). That also did not work, as there was still that noise on the output of the buffer. The envelop detector's signal was fine tough (as expected by usage of the buffer). This noise/interference on the comparator switching I couldn't explain, and I ran out of time before I can further investigate this issue.

As for the TIA, due to time I also wasn't able to get that working as of writing this report. So in conclusion the band-pass filter and envelop detector works, but not the TIA and comparator stage on the receiver side.

4 Simple TIA+Comparator Design

4.1 Overview

This circuit was made as we started to run out of time before the end of the semester, and as mentioned in Section 3.7 the *OOK Design* wasn't working, and we wanted a working model of transmitting data over visible light. This circuit was also used to quantify and characterize what TIA feedback resistances worked up to what distances. This circuit does not feature a Transmitter Differentiation layer. UART was used as the data layer, with a USB-UART to control the data sent out as well reading what was received.

4.2 Schematic

The schematic for this design, shown in Figure 18, is simply an LED driver at the transmitter side (an 2N2222 NPN to be exact). The LED chosen as a modified dollar store flashlight which provided great brightness.

On the receiver side, a jellybean op-amp (TL082) is used as the op-amp for the TIA to convert the photodiode's voltage to current, then a comparator is used to set a threshold for what is low or high.

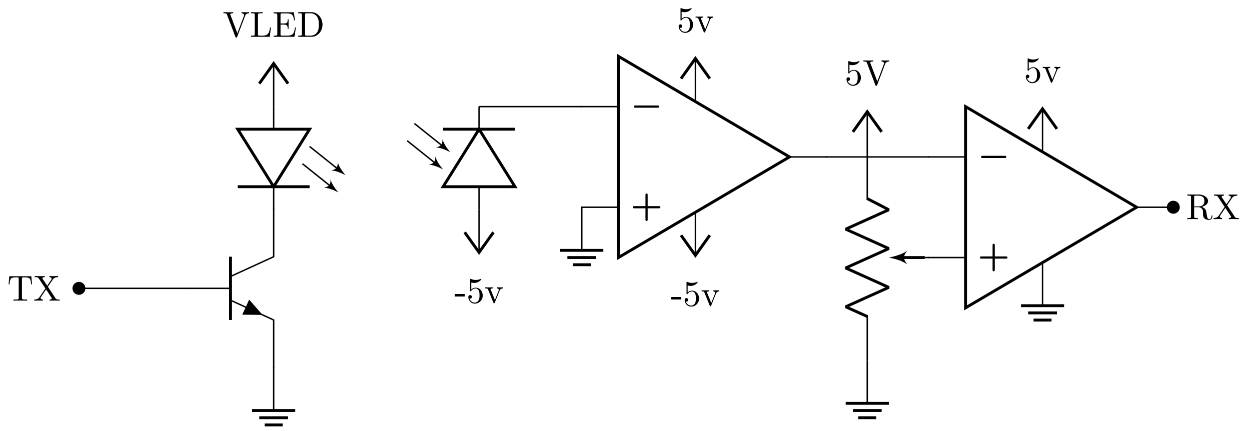


Figure 18: Schematic of the Simple TIA+Comparator Design

4.3 Automated Tests

As mentioned in Section 4.1, this circuit was primarily used to A) Prove that transmitting data thru visible light is feasible, and more importantly B) Gather data about the LED's range, TIA's feedback resistor selection, and transmission rate. So a series of tests were done to characterization those factors.

Setup

The TIA's feedback resistor and LED's range was changed per test manually, as well as the comparator's threshold. The comparator's threshold was changed by running an initial test, analyzing the TIA's output on the oscilloscope and adjusting the comparator's threshold so that it's right in the middle of the TIA's signal. As for the transmission rate (in this case a UART's baud rate) was changed with an automated Python program. This program will, per baud rate change, send out 8 random bytes and read the UART input 100 times. If the bytes are the same, increment some counter which will be used, with the total number of bytes sent, to determine the failure rate. A picture of the physical setup is shown bellow in Figure 19

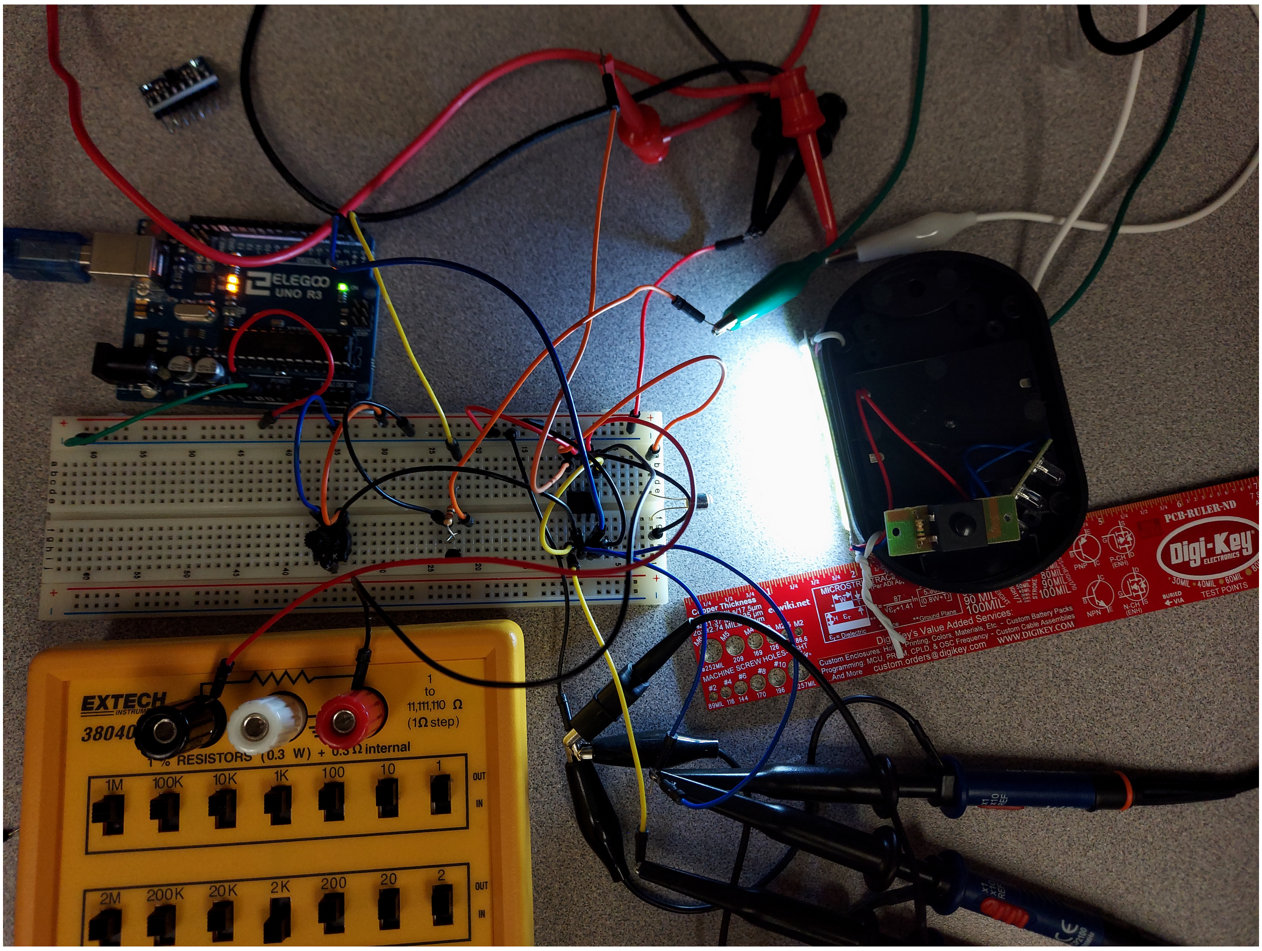


Figure 19: Automated Test Setup for Simple TIA+Comparator Design

Test Results

Putting the LED 2in away from the photodiode, we get the following failure rates:

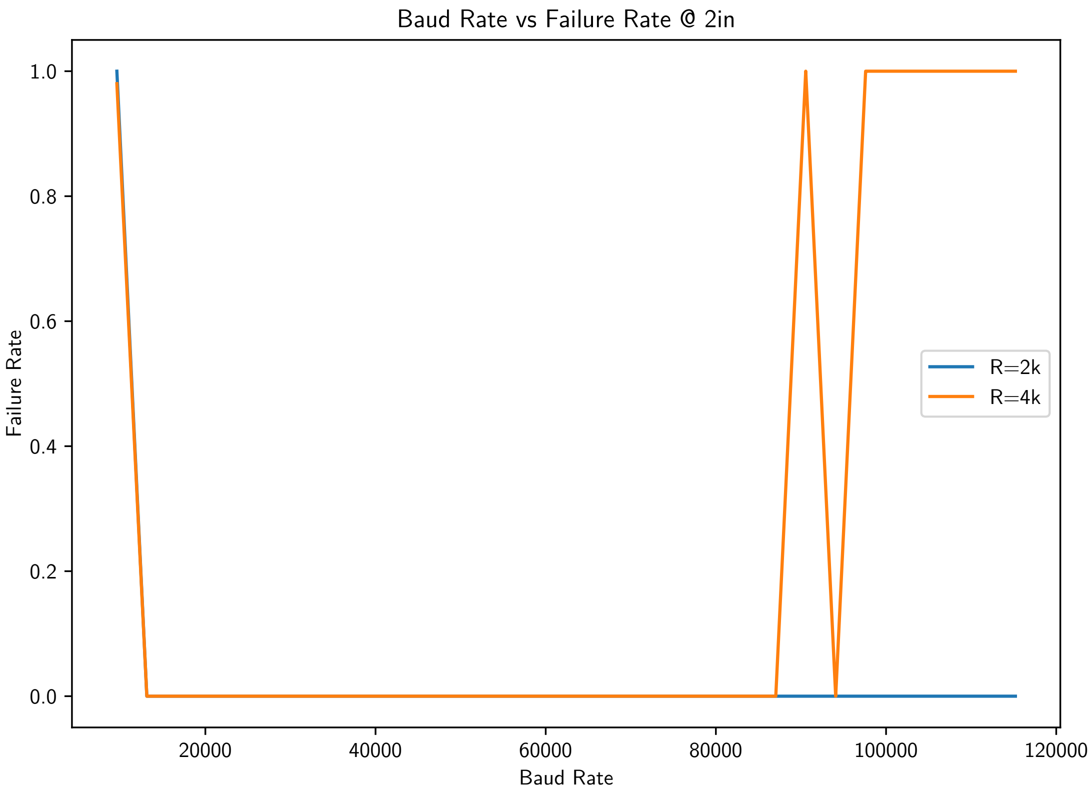


Figure 20: Test Result of LED 2in from Receiver Photodiode

For some reason, the failure rate is 100% at very low baud rates, which is unexpected. Other than that, the rest of the graph makes sense where the failure rate is 0%, but jumps up at 100% as the baud rate is increases. The $2k\Omega$ resistor has a 0% failure rate up to the highest baud rate tested (115200), but the $4k\Omega$ feedback resistor was only able to sustain up to $\simeq 90000$ baud.

Putting the LED 6in away leads to the following result:

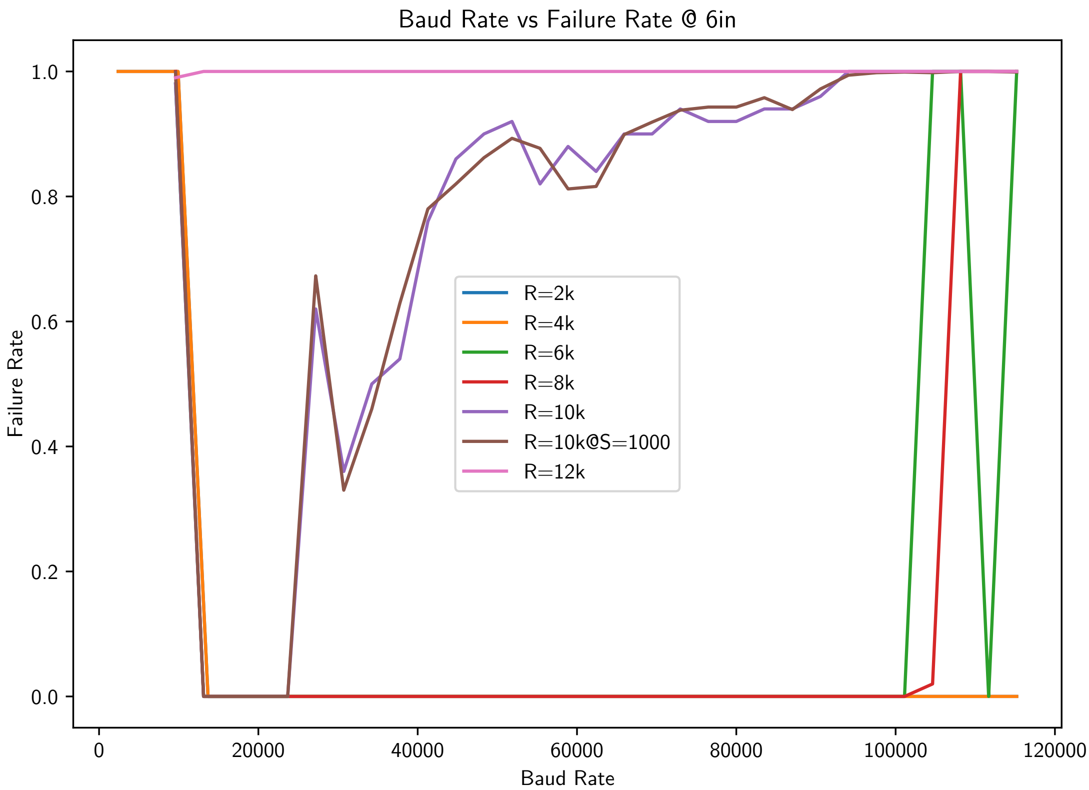


Figure 21: Test Result of LED 6in from Receiver Photodiode

This one also fails at very low baud rates, even more so than when the LED was 2in away from the receiver. Other than that, it has the same characteristics as the 2in test, where is is some ideal feedback resistor value that passes everything, and as that changes the failure rate increases, especially at higher baud rates. The increase in failure rate as baud rate increases is perfectly illustrated with the *R=10k* data point.

To note for Figure 21, the data point *R=10k@S=1000* is the same as *R=10k* but with an increased number of data samples, in this case 1000 vs 100.

Running the test with the LED 12in away leads to the following result

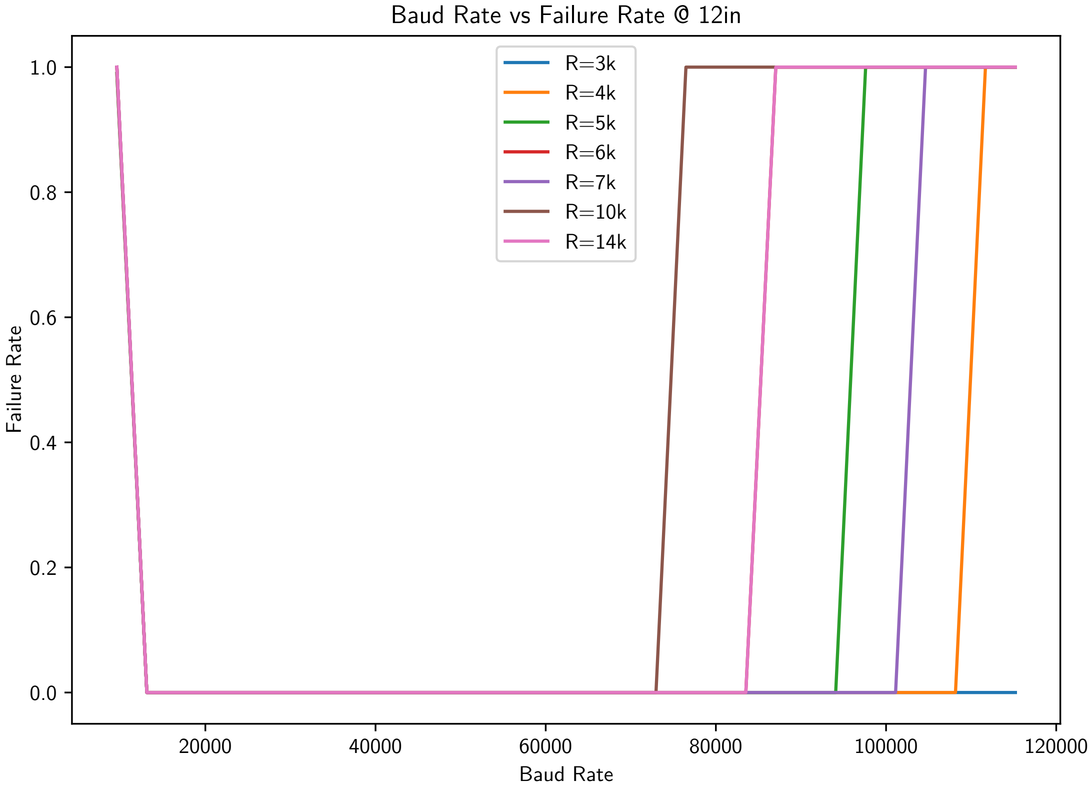


Figure 22: Test Result of LED 12in from Receiver Photodiode

This result also has the same characteristics as the 2in and 6in test, where it fails at low baud rates, but also fails at higher baud rates depending on the feedback resistor value.

We also tried angling the LED away from the receiver. A picture of that setup is shown in Figure 23, and the data for that is shown in Figure 24.

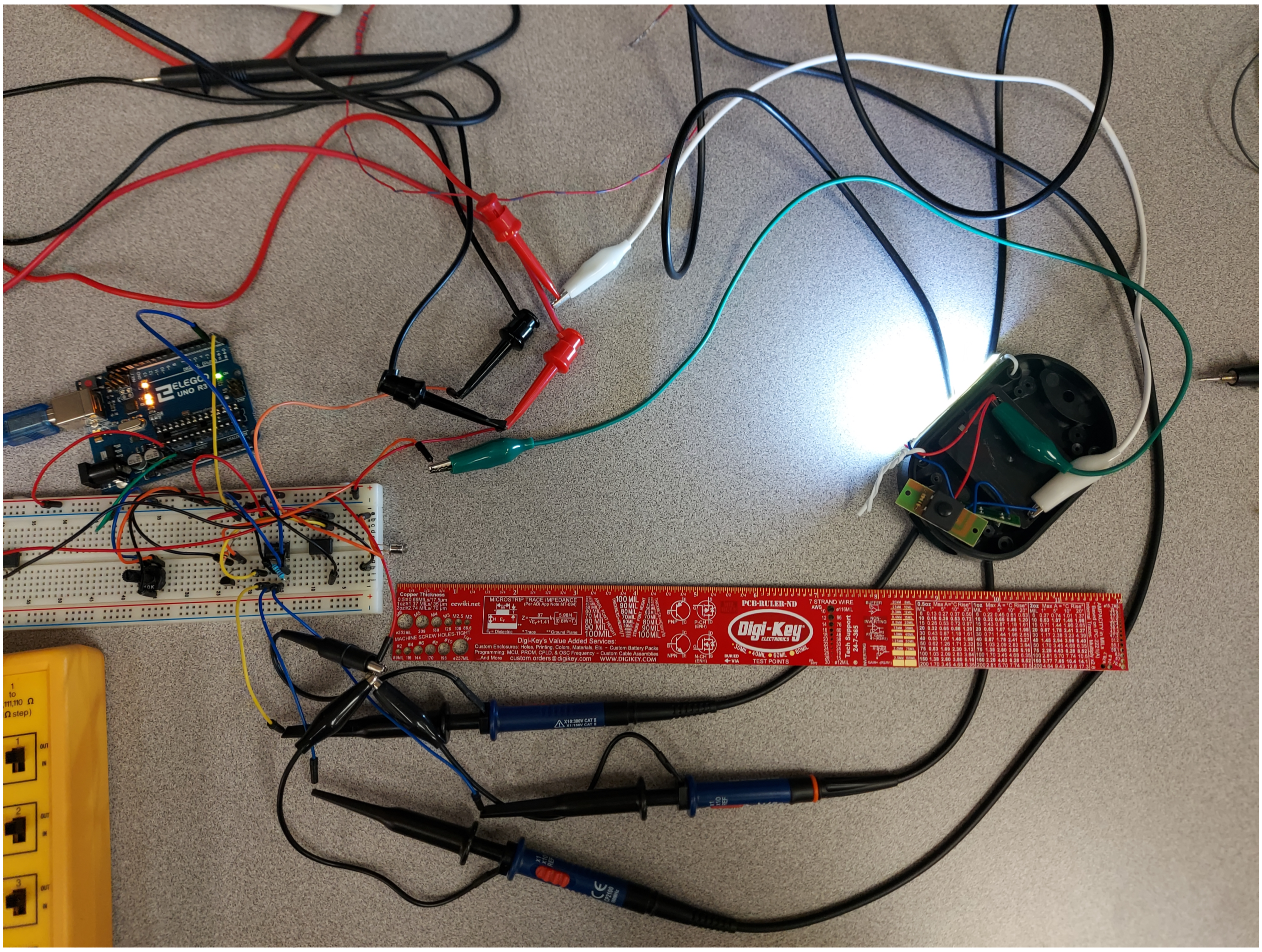


Figure 23: Automated Test Setup for Simple TIA+Comparator Design with a 45° angle

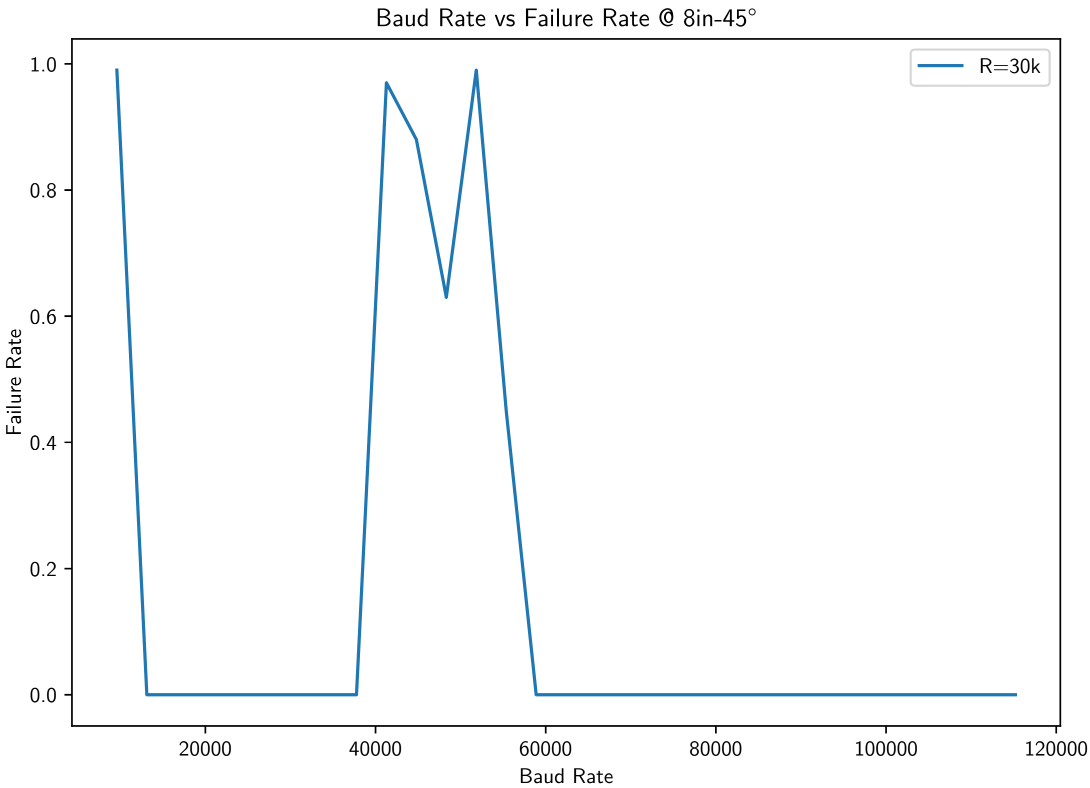


Figure 24: Test Result of LED 8in from Receiver Photodiode at a 45° angle

As it can be seen, it also most passes everything (except for low baud rates), with the exception of some middle baud rates. This I theorise to be because I did move mid-test, and I had to increase the TIA's feedback resistor up to $30k\Omega$. More about this is discussed in Section 4.3

Result Conclusion

From the data shown, we can conclude that at the right sensitivity this simple transmitter/receiver setup works up to the tested 115200 baud, which is more that I personally expected. I also successfully test the LED-receiver distance up to 18in away and it continued to operate fine (I did not showcase the data for that setup).

One thing I noticed is the need for adjustment in both feedback resistor and comparator threshold. This proved more evident in the angled test, where the TIA's feedback resistor needed to be $30k\Omega$ in order to receive the signal. The comparator threshold also depended on ambient light conditions. On Presentation Day, when I presented this setup, as my body moved (so blocking and unblocking some ambient light behind me), the TIA's output level changed due to the change in ambient light reaching the photodiode, thus an adjustment to the comparator's threshold was required and I needed so stand still after adjusting the comparator's threshold.

In a real product, this manual adjust will be unacceptable. This, as discussed in Section 6.1, can be potentially solved with a digitally controlled TIA where the feedback resistance is changed, as well as a digitally controlled comparator.

5 BOM

The following 3 tables are the BOM for each of the different designs. I personally would take it with a grain of salt, as this isn't a finished design and only an R&D project. Also some components, especially for the OOK BOM, has not been taken into account (miscellaneous capacitors and resistors, so a negligible cost) or some components have not been used (for example the transmitter's clock IC and switches). I have also attached the price of a decade box to the *Simple TIA+Comparator Design*'s BOM, which is where the majority of the cost is at.

Item	**Quantity**	**Price/Unit**	**Extended Price**
Generic White LED	1	\$0.10	\$0.10
Arduino Uno or Nano	2	\$5.33	\$10.66
IR Receiver Module	1	\$1.63	\$1.63
		Total Cost	\$12.39

Table 1: JellyBean IR Receiver Design BOM

Item	Quantity	Price/Unit	Extended Price
TL082 Op-Amp	1	$0.39	$0.39
USB-UART Bridge	1	$2.00	$2.00
White LED	1	$0.50	$0.50
Photodiode	1	$2.00	$2.00
LM393 Comparator	1	$0.37	$0.37
Decade Box	1	$80.00	$80.00
10k Potentiometer	1	$0.10	$0.10
		Total Cost	$85.36

Table 2: Simple TIA+Comprator Design BOM

Digi-Key Part Number	Reference Designator	Quantity	Unit Price	Extended Price
478-1305-1-ND	C1, C2,	2	$0.15	$0.30
1276-6471-1-ND	C6, C7, C8, C9, C10, C11, C12, C13, C14, C15, C17,	11	$0.06	$0.63
1276-1244-1-ND	C16,	1	$0.12	$0.12
1080-1007-ND	D1, D2,	2	$0.74	$1.48
NRVTSA3100ET3GOSCT-ND	D5,	1	$0.41	$0.41
1125-1007-ND	D8,	1	$5.32	$5.32
2092-KLDX-0202-B-ND	J9, J11,	2	$0.54	$1.08
RMCF0805ZT0R00CT-ND	JP2, JP3, JP4,	3	$0.10	$0.30
732-1613-1-ND	L1, L2,	2	$0.22	$0.44
FDV301NCT-ND	Q1, Q2,	2	$0.27	$0.54
RMCF0805JG10K0CT-ND	R1, R2,	2	$0.10	$0.20
RMCF0805JT1K00CT-ND	R3, R4,	2	$0.10	$0.20
RMCF0805JT330RCT-ND	R5,	1	$0.10	$0.10
490-2875-ND	RV1, RV2,	2	$1.50	$3.00
CT3117CT-ND	SW1,	1	$0.83	$0.83
CKN9565-ND	SW2, SW3, SW4,	3	$0.51	$1.53
36-5005-ND	TP1, TP2, TP3, TP4, TP5, TP6, TP7, TP8, TP9, TP10, TP11, TP12, TP13, TP14, TP15, TP16, TP17,	17	$0.42	$7.14
NB3N502DR2GOSCT-ND	U1,	1	$4.56	$4.56
LTC6268IS6-10#TRMPBFCT-ND	U2,	1	$8.40	$8.40
ADA4891-2ARZ-R7CT-ND	U3,	1	$2.19	$2.19
LMV7219M5CT-ND	U4,	1	$3.81	$3.81
296-43957-1-ND	U5,	1	$1.08	$1.08
535-9831-1-ND	Y1,	1	$0.87	$0.87
-	PCB	5	$1.00	$5.00
			Total Cost	$49.53

Table 3: OOK Transceiver Design BOM

6 Conclusion

Overall, we believe that the concept of LIFI is viable, but difficult to get working properly. Some issues we had during this design was due to time, some due to our lack of experience, and others due to the nature of photodiode. We do not believe LIFI will take over, but it will find it's niche purposes. Some of those purposes and potential applications includes:

- VR Applications, where line-of-sight is no problem, there is already lighting in the room, and high data rates are required.

- Vehicle to vehicle communication, where a vehicle already has headlights it can use to communicate to other cars, for example for a future autonomous vehicle system.

With that said, there are some issues we must overcome in order for the applications to be viable:

- As this technology is to be implemented in a general environment, the use of a larger-die photodiode is required. As the photodiode die size increases, so does it's capacitance and thus a lower bandwidth.

- The sensitivity and threshold of the receiver circuit must be adjustable by software, and it can vary throughout different environments

6.1 Future Works

Clearly this project needs more work put into it, especially if a product was to be made with this technology. The following list is some future works for this project

- Perform further tests with the *Simple TIA+Comparator Design* to further gather more data on what works to transmit data with light.
- Finish attempting to get the *OOK Transceiver Design* circuit functional.
- Design a receiver with a digitally controlled TIA sensitivity as well as a digitally controlled comparator threshold.
- Perhaps look into adding a mixer to the *OOK Transceiver Design* if the carrier frequency was to be increased.
- Look into other more complex but more rewarding modulation techniques.