Investigation of Performance of Visible Light Communications for Use in Power Plants

by

Ethan Linderman

B. S. in Mechanical Engineering, University of Pittsburgh, 2019

Submitted to the Graduate Faculty of the

Swanson School of Engineering in partial fulfillment

of the requirements for the degree of

Master of Science in Mechanical Engineering

University of Pittsburgh

2022

UNIVERSITY OF PITTSBURGH

SWANSON SCHOOL OF ENGINEERING

This thesis was presented

by

Ethan Linderman

It was defended on

April 8, 2022

and approved by

William Clark, PhD, Professor, Mechanical Engineering and Material Science

Nikhil Bajaj, PhD, Assistant Professor, Mechanical Engineering and Material Science

Jeffrey Vipperman, PhD, Professor, Mechanical Engineering and Material Science

Thesis Advisor: Jeffrey Vipperman, PhD, Professor, Mechanical Engineering and Material Science

Investigation of Performance of Visible Light Communications for Use in Power Plants

Ethan Linderman, M. S.

University of Pittsburgh, 2022

Wireless communication is an integral part in modern society, however sending data via certain parts of the electromagnetic spectrum can raise a security risk because of their transmission through barriers and typical omnidirectional characteristics. Additionally, the use of many wireless devices in a local space can cause spectrum pollution, a bad side-effect especially when dealing with machines and instrumentation that are sensitive to wireless signals. This paper documents the investigation of another form of wireless technology that uses visible light as its communication medium and does not suffer from the previously mentioned problems. The criteria considered were distance of data transmission, speed of transmission, and the effects of the environment on both communication distance and speed. Theories of light physics and electronic data communications were used to design a system that could meet the design requirements. Prototypes of the system were made and tested in a variety of conditions to validate the theory. These tests gave a great understanding of visible light communication (VLC) and its benefits and shortcomings when compared to other forms of communication like Wi-Fi.

Table of Contents

List of Tables

List of Figures

Preface

I would like to thank Dr. Vipperman, Dr. Chorpening and the team at NETL for helping make this research possible. I would like to thank ORISE and NETL for providing the funding to make this research possible. I would also like to thank the Canonsburg police for allowing me to do outdoor testing in a church parking lot at night.

A majority of this testing was performed during the lockdown period of the COVID-19 pandemic where access to standard lab equipment was unavailable.

1.0 Introduction

Visible light communication (VLC) is a relatively new form of wireless communication that uses a portion of the electromagnetic (EM) spectrum to transmit data. The visible light portion of the EM spectrum (400-700nm) is unlicensed and the hardware to implement it is readily available. Visible light communication in theory is a semaphore system, or a system that uses visual cues to transmit information over distance. Simple semaphore systems can trace their roots back to ancient times where the Ancient Greeks built fires on certain mountain tops in a chain. The fires would be arranged in groups which, when deciphered, would convey individual letters and subsequently words to the next mountain in the chain, some 20 miles away. It is said that the Greek acropolis of Mycenae was notified about the fall of Troy via this method. In more modern times, heliographs use mirrors to reflect the sun to an observer some distance away in a similar manner to the Greeks. These heliographs were in integral part of military communications in pre-radio times. In the late 1800's, the US military had a chain of heliographs that could span up to 140 miles over rough land that would make standard horseback communication difficult and slow.

Visible light communication today is often talked about in situations where overhead lighting is already implemented, and a supplement to the downlink of data would be beneficial. For instance, when walking around in a mall and connected to free Wi-Fi, it may be beneficial to have the overhead lights transmit data to a mobile device over VLC to supplement the free Wi-Fi. In this case, the distance from the light source to receiver (mobile device) is relatively short and the receiver can move around the mall so long as there are overhead lights. The VLC form discussed in this paper is slightly different. The concept of sending data by the imperceptible modulation of light remains, but the distances in this paper are much longer and the receiver is stationary.

Currently there are many forms of wireless communication that transmit information effortlessly at high speeds. However, it may be problematic to use devices that emit parts of the EM spectrum that can potentially harm sensitive equipment. Moreover, the radio band of the EM spectrum, which includes things like Bluetooth, Wi-Fi, cellular connectivity and of course radio, is finite and allocated to different services by the Federal Communication Commission (FCC). Finally, using a wireless communication such as radio waves or Wi-Fi can pose security risks since devices that emit those signals emit them in all directions, and it would be difficult to control this outdoors at a powerplant. Cyber-attacks are not uncommon in the modern world, and attack on a powerplant would have numerous consequences. Hackers would potentially be able to access key files, gain control of the plant or even shut it down completely for a period leaving thousands of residents without power. It is widely known that in times of power outages, injuries and even deaths go up dramatically.

Visible light communication tries to mitigate these problems in three ways: by not emitting radiation in the parts of the EM spectrum that can potentially harm equipment, by using the part of the EM spectrum that is not licensed or allocated by the FCC, and by controlling the direction and divergence of the light beam. This means that VLC can be a viable option in any sector that uses equipment that is susceptible to EM spectrum radiation, any sector that wishes to minimize EM spectrum pollution, or any sector that wishes to have highly secure wireless data transfers.

There are challenges when designing a battery powered VLC system which can transmit data over 100 meters outdoors at a powerplant. The goal is to make the system power-efficient enough to run from a battery and incorporate solar power as a supplement. The system must also be able to operate in all lighting and weather conditions. This paper will discuss the challenges of making a VLC system such as: modulating a high-power LED, determining the best optics to make the beam travel 100 meters, keeping power draw to a minimum, keeping overall size to a minimum, determining the best software platform to control the system, and testing all the components together. This paper will also discuss digital communication theory and how it relates to designing the best system possible to meet the requirements.

2.0 Optics

To send a beam of light over any distance is quite simple. A photon of light will travel forever until it interacts with something. Light travels very well in a vacuum because of this. Light can also travel through other common mediums like air and water, but by interacting with the mediums' particles along the path, light will lose energy and travel only a finite distance. For the distances discussed in this paper (up to 100m), the amount of energy light loses through the atmosphere is not negligible, but it is low, and accounted for in the mathematical model.

Imagine a point light source that has no dimensions; an infinitely small point that radiates light in all directions equally. This is the most basic building block of the theory discussed in this paper. Take readings of its light intensity at regular intervals away from the point source, and notice it diminishes in a pattern. It quicky loses intensity up close and loses it more slowly the farther away a reading is taken. It "levels off" so to speak. This is called the law of inverse squares; the light's intensity at any distance is proportional to the inverse of the distance from the point source squared.

$$Light\ intensity \sim \frac{1}{distance\ from\ source^2} \tag{2-1}$$

To further understand this law, we need to know how light gets measured. Light sources emit photons. The total amount of photons emitted is proportional to the number of lumens it emits. However, human eyes don't directly see lumens, we can more easily detect flux, which is a measure of something per unit area. In this case, luminous flux (in units of lux) or luminosity is a measure of lumens per square meter. This flux measurement is akin to perceived brightness.

Consider figure 1. Two lights project onto two walls. Light A has double the amount of luminous output (represented by the blue dots) verses light B. However, light B is closer to a wall than light A is and thus the area that the light is projected is different. In fact, even though light B has less output, the projection it creates is brighter because there are more lumens per unit area.

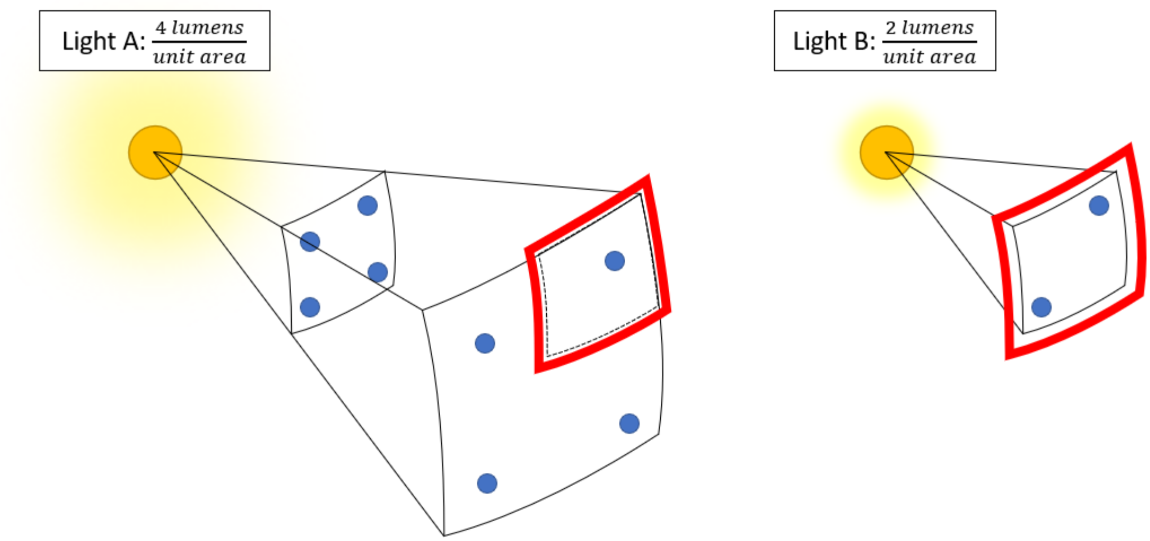


Figure 1: Lux Comparison Between Two Lights

We can tell this by taking the same unit area (red square) and counting the blue dots. Light B has 2 dots per area while light A only has 1 dot.

Light energy can be characterized in multiple ways. The most accurate way is to measure its irradiance. Irradiance is defined by the amount of electromagnetic spectrum (EM) energy falling perpendicularly onto a unit surface and has units of W/m^2. This measurement is unweighted, giving a normalized reading. Irradiance meters can be purchased and are typically used to measure sunlight.

Another way of measuring light is by its luminosity, like previously mentioned. Conceptually it is similar to irradiance; however, lux readings weight certain parts of the EM spectrum differently so that brightness values collected with a luminosity meter (lux meter) closely match the response of the human eye. To weight the EM spectrum, lux uses something called the Luminous efficiency function, shown below.

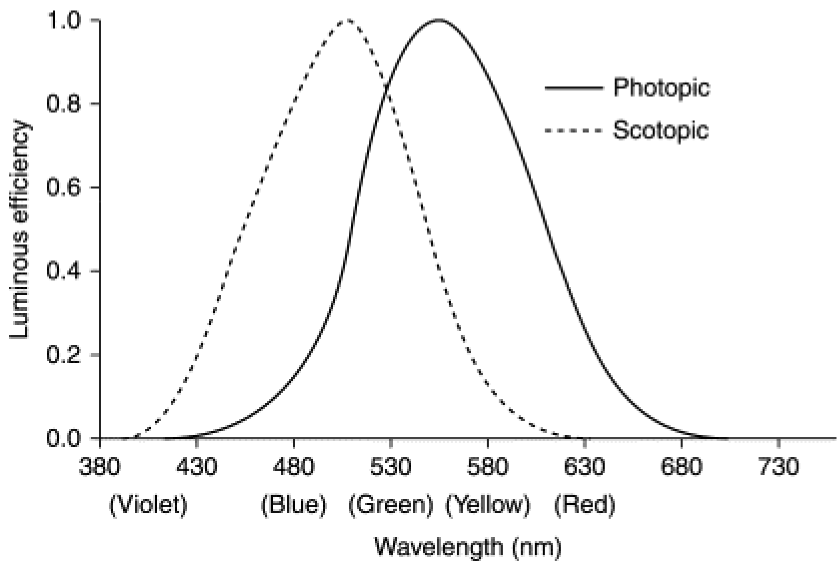


Figure 2: Luminous Efficiency Function

Notice the solid black line for Photopic vision, or vision under well-lit conditions (as opposed to Scotopic, or dark conditions). As can be seen, the Luminous efficiency function adds weight to wavelengths only in the visible part of the spectrum, and within the spectrum, it most heavily weights wavelengths that are most sensitive to the cells in the human eye, about 555nm.

Take 2 equal power monochromatic lights that emit only 1 wavelength of light, one blue light at 480nm and one greenish yellow light at 560nm. The blue light will appear dimmer to the human eye than an equally powered greenish-yellow light. Similarly, the lux meter will give a higher reading for the greenish yellow light because it weighs the green part of the EM spectrum higher to match the human eye response. On the other hand, the same two lights being measured with an irradiance meter will output the same value, since their powers are equal, and the irradiance meter weights all wavelengths equally.

As can be seen, luminosity meters can be problematic when looking at specific wavelengths because of this Luminous efficiency function. However, when comparing two white lights with a luminosity meter, the issue is mitigated because white light is a mixture of all wavelengths in the visible part of the EM spectrum. In the real world, perceived white lights typically do not have a flat spectrum and so using a lux meter to compare 2 white lights may give different results. If the wavelengths that make up two white lights are known and equal, then the lux meter is an accurate tool to use to compare the two.

Due to availability, a lux meter was used for the entirety of this study. The light sources used are LEDs and each LED's wavelength makeup is known.

Consider our infinitely small point source again; it emits light in all directions equally. By tracing two rays of light emitted from the source, we can see that they will always diverge unless interacting with something. This diverging of the two rays means that the light will appear dimmer the farther away an observer is from the source. Likewise, the lux measured on a surface will be lower the farther away a surface is from the light. The formula for lux is:

$$lux = \frac{lumens}{area} \tag{2-2}$$

2.1 Collimation

Because of this law of inverse squares, it is not useful to illuminate something that is relatively far away (say 100m) with just a point source alone. To do this, the light would have to be extremely bright, and thus consume a relatively high amount of power. If the light is sufficiently bright, then illumination at 100m is possible, but it would be inefficient. An example of this is the sun, a source of light that for the most part does not interact with anything before it reaches the earth.

To better illuminate something 100m away, it is useful to use a device that changes the path of the light rays so that they diverge less over distance. Rays that are parallel to each other are considered collimated. If light rays are perfectly collimated, then their projection onto a surface does not change in area with distance from the source, and thus the brightness or lux on said surface will not change. With collimated rays, a light can be less powerful to achieve the same brightness at 100m than a light with diverging rays.

To perfectly collimate something is nearly impossible because the theory of collimation assumes that the source of light is an infinitely small point. In the real world, light sources have dimensions. This fact alone means that perfect collimation is not possible, and any rays that appear to be collimated diverge slightly. More on this later.

A device is needed that can shape the light rays to approximately parallel to collimate light. Two common devices for this task are lenses and reflectors.

A computer program, Ray Optics Simulation [7], is an open-source web app to simulate the reflection and refraction of light suitable for education and demonstration will be used throughout this section. In this program, parabolic lenses and reflectors are depicted as straight lines.

Consider our infinitely small point that emits light. Pair this light with a lens. By placing the source of light at the lens' focal point, the light rays that pass through the lens will come out collimated, as shown in Figure 3.

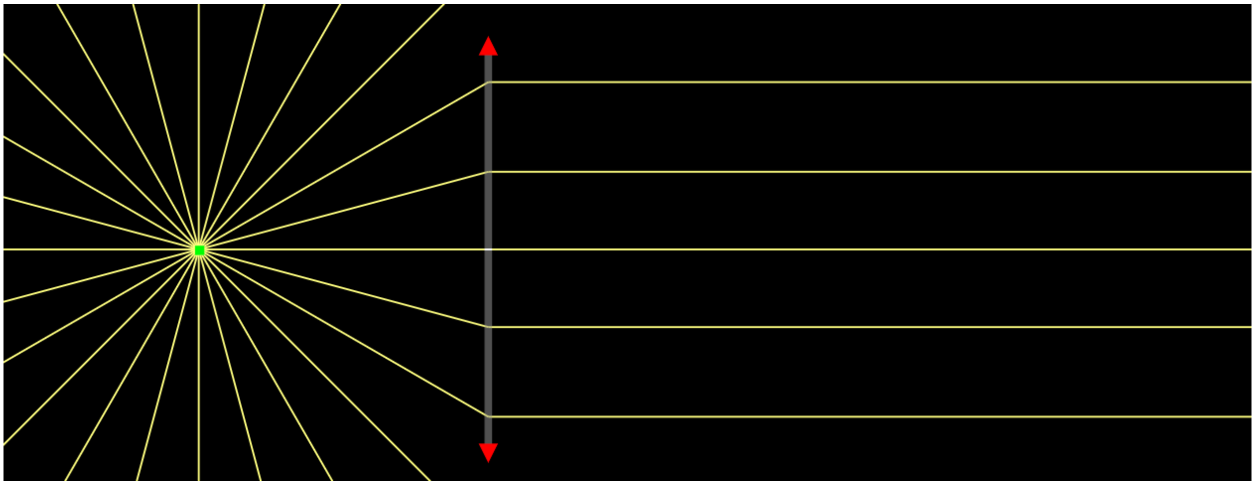


Figure 3: A point source emitting light into a lens

The path of the rays that pass through the lens are altered due to the interaction of the light into a different medium. From there, if the rays fall onto a surface, their projection on that surface will be the same size no matter the distace from the source.

Another way to get these rays to be parallel is to have them bounce off a special surface. The surface must have parabolic curvature and be reflective, as shown in figure 4. Again, note that the program depics the prabolic refelctor as a stright line.

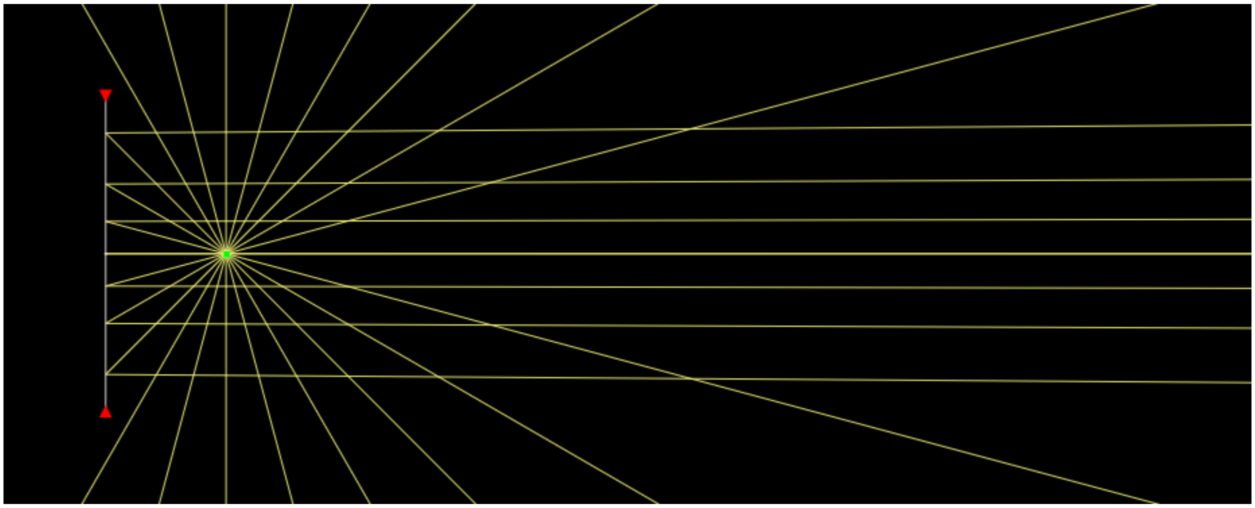


Figure 4: A point source emitting light onto a parabolic reflector

The source of light (green point) emits rays in all directions; however, the rays that interact with the special surface (shown as a silver line) bounce off and become parallel. Each of these two methods have their pros and cons, and have both been explored and tested.

2.2 Real World Collimation

If an observer is far enough away from a light source, then the rays can appear collimated. For instance, the sun is so far away that when its rays reach earth, they are practically parallel. This creates an interesting phenonenon with shadows. Shadows created by sunlight do not appreciably change size the closer or further away they are from a surface. This is unlike a close light source such as a lamp where a shadow created by a hand will change in size depending on how close or far the hand is from a surface.

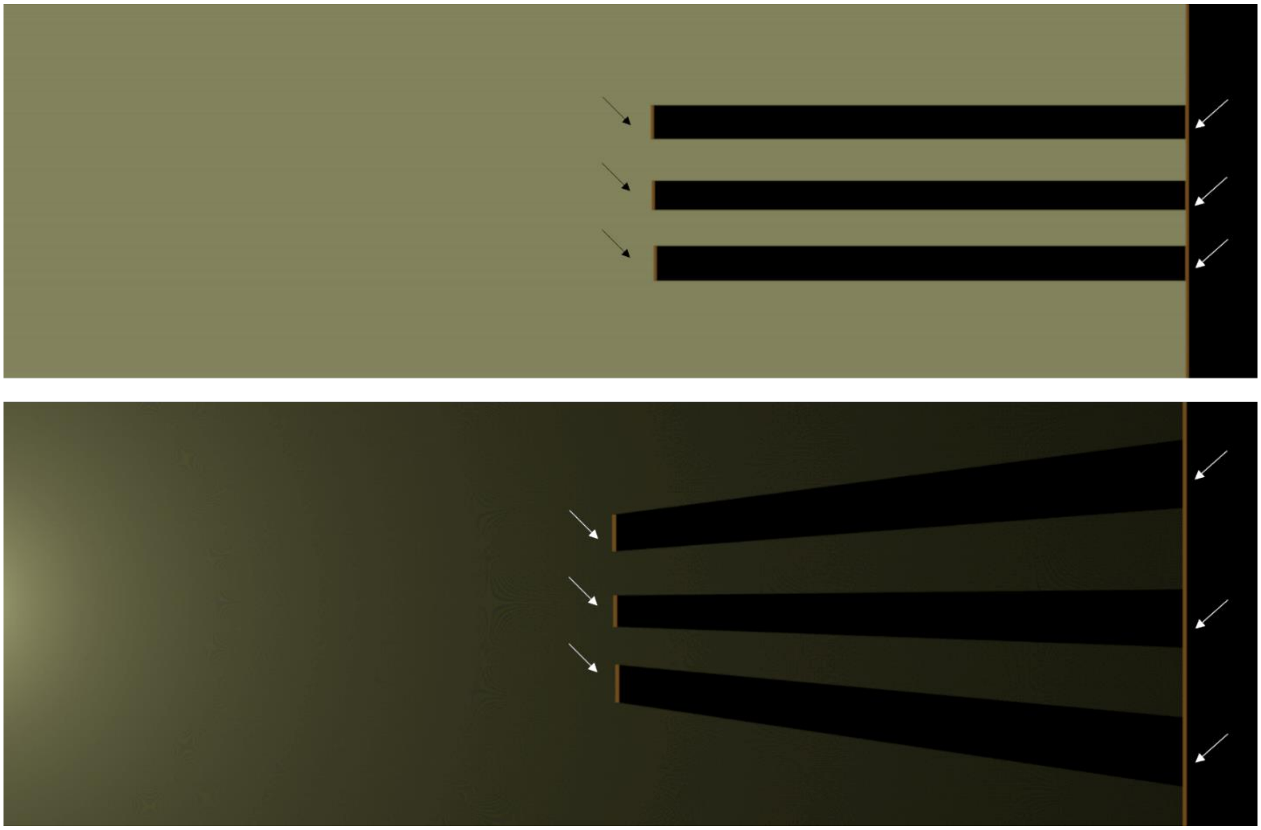


Figure 5: Difference in shadows between collimated and non collimated light

As we can see from figure 5, three objects create a shadow in the light. In the top half of figure 5, the light source is far enough away and the light is nearly collimated. Thus, the shadows from the objects remain the same size until they get projected onto a surface. Conversley, the bottom half of the figure shows the light source much closer. In this photo, the arrows point to the widths of the shadows, which grow in size until they get projected onto a surface.

Because all of the sources of light discussed in this paper have dimensions to them, the collimation achieved is not perfect. The beam divergence, ϕ, can be related to the source dimension, D_s and lens focal length, f_{col}, as:[1]

$$\emptyset = \frac{D_S}{f_{col}}. \tag{2-3}$$

As can be seen, to decrease the beam divergence, we can either decrease the diameter of the light source or increase the focal length of the lens. Each method acheives the goal of decreasing the beam's divergence, but introduces an important trade-off. By decreasing the size of the emitter, D_s, the light source typically becomes less powerfull. On the other hand, by increasing the focal length, f_{col}, the lens will inherently be farther away from the source, and thus capture a smaller portion of the total light emitted. Figures 6 and 7 demonstrate this.

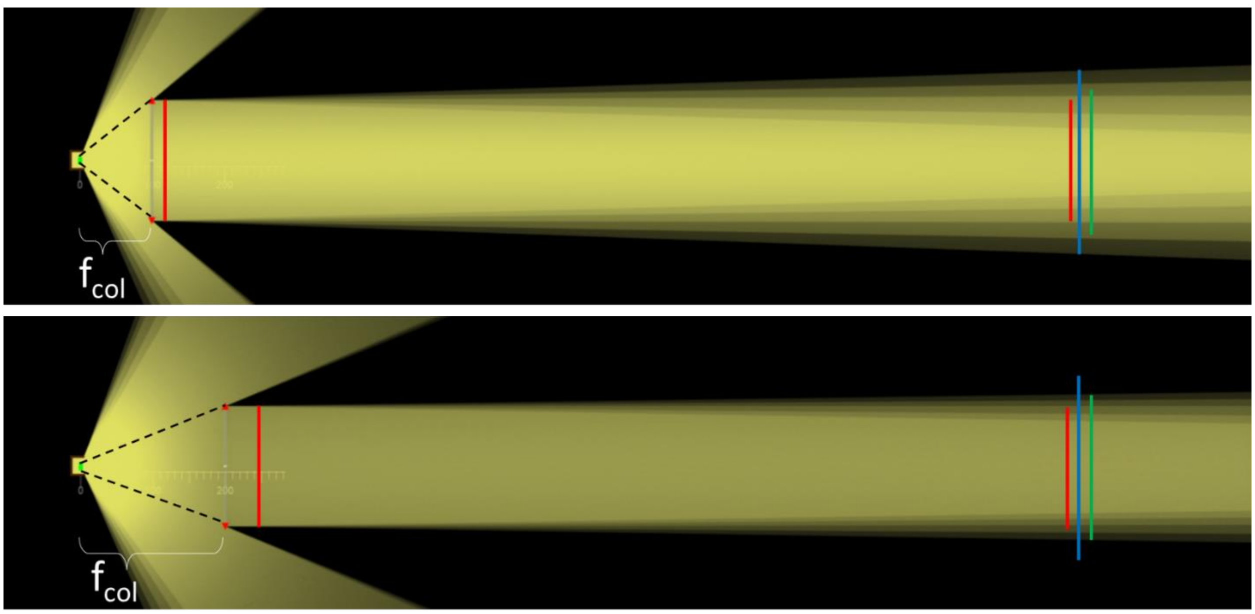


Figure 6: Effects of changing focal length

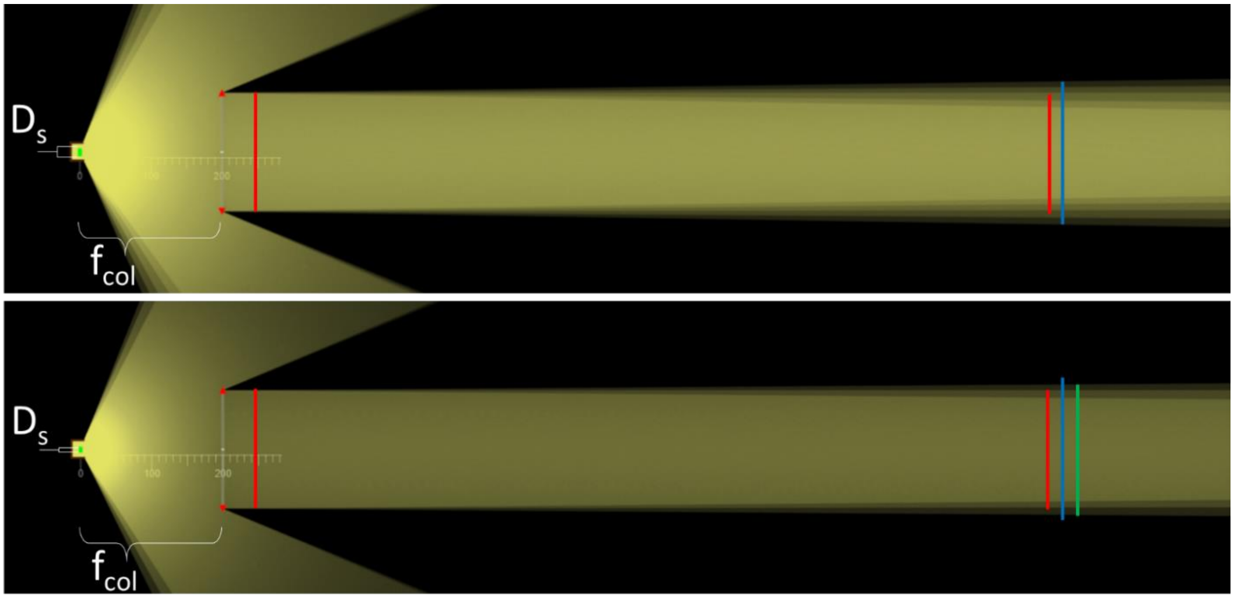


Figure 7: Effects of changing light source size

In figure 6, the red, blue, and green lines, respectively, are equal length. We can see that in the top half of figure 6 , f_{col} is small and creates a beam at a distance that has a height equal to the blue line. So, the beam height goes from the red line to the blue line. In the bottom half of figure 6, f_{col} is larger. Consequently, the height of the beam at a distance is equal to the green line. The blue line has been copied down from the top half to be easily compared to the green line. From this comparison, we can see that just by increasing the focal length of the lens, the beam diverges less and becomes more collimated. The trade-off is that by moving it farther away, it captures a smaller portion of the total light from the source, meaning that a larger diameter lens (if permissible for the application) would need to be used to take full advantage of the longer focal length.

In figure *7*, all same colored lines are again equal length. This figure explores the effects of only changing the size of the light source. The bottom half has a smaller light source and consequently the diameter of the beam at a distance is smaller.

The testing done in this paper explored both large and small light sources as well as large and small focal lengths. In addition, a mathmatical model based on equations 1, 2, and 3 as well as taking into account different LEDs and lenses has been developed that predicts the brightness of a particular light and lens setup at any distance. This model can be used to determine the approximate optical performance of a light transmitting system before it gets purchased and can potentially be used as a design tool.

3.0 Digital Communication

Digital communication is a large part of a VLC system. Relaying info to a computer usually happens in one of two ways, either analog or digital. Analog signals are usually continuous and are ‘analogous’ to the real-world thing they are signaling. On the other hand, digital signals are discrete and typically only have two states the signal can be in, represented in binary. Putting the two discrete states into a certain order can relay info quite efficiently. Consider a computer system that wants to read the temperature outside. The analog way would be to send a voltage proportional to the temperature being read; the higher the temperature, the higher the signal voltage. This is an easy concept to understand because humans work in this analogous way. The alternative is to send the signal digitally. For instance, if the temperature was 70°F, then the system might send a combination of two discrete states that would follow a standardized code. When it gets decoded, it would read 7 and 0.

Consider a system that reads analog signal and digital signal. To read the analog signal clearly, it must be a sensor that has a good enough precision to read small differences in analog signal. For instance, if the signal was 5 volts, would the system be able to discern between 5V and 5.01V? On the other hand, if the same system only had to discern between two distinct voltages, say 0V and 5V, the distinction would be much clearer.

With this knowledge, the VLC system should use digital communication because reading an analog signal over light can become muddled with other light in the real world. With a VLC only having to discern between two digital states, say no light and maximum light, the signal can be much more robust.

3.1.1 Type of Digital Communication

It has been established that the VLC system should use digital communication, and that the two states that it reads should be no light and maximum light. This method of turning off and on the light, called On Off Keying (OOK), can be put into a standardized system of binary code.

Take for example the letter ‘E’. This character has an agreed upon binary representation via the American Standard Code for Information Interchange, or ASCII.[2] This binary representation is 1000101. If we wanted to just send this character in its most simple form, then using our method of turning on and off the light, we could turn the light on (representing a 1). Then, shut the light off for triple the amount of time as it was on (representing 000). Then once again turn the light on for a single unit of time, then off, then on (representing 101).

Already, we can see some issues with this. Firstly, how would the system know when to start reading the signal? If the light is on with no signal, then starting the signal with a 1 does not change anything and causes issues. Conversely, if the light is off with no signal, then starting the signal with a 0 can cause issues. The same can be said about when to stop reading the signal. Furthermore, how does the receiving system know how long the unit of time is? Is the first 1 of the signal only a 1, or could it be multiple 1’s? These issues all get resolved through clever communication standards.

There are multiple agreed upon communication standards that solve this issue. Different communication standards require different numbers of communication channels. Some communication methods use two channels, one dedicated to sending data, and the other dedicated to a pulsing clock signal. The benefit is that with these systems, the transmitter and receiver can be operating at different speeds but become synchronized by the shared clock channel. The downside is that this method uses multiple channels, which are usually wires (in our case light beams) and can be undesired and costly. The VLC system discussed in this paper only has one communication channel and so it uses a communication system called UART or Universal Asynchronous Receiver Transmitter.

3.1.2 Universal Asynchronous Receiver Transmitter

Universal Asynchronous Receiver Transmitter communication uses 1 channel to relay information. Unlike other methods that use two channels, it is more challenging to know when to start receiving data, since one channel must act as the clock, start and stop bits, and data. UART has two main states, idle and sending data.

In the idle state, the logic level is held high. This stems from the past with telephone land lines where the idle state was held as a high voltage to show that there was not a break in the connection between two locations.

In the sending data state, the data is put into a package with a start bit, stop bit, data, and potentially a parity bit. The start bit is a logic low and the stop a logic high. This guarantees that there will always be at least two logic level changes per packet. The full process is shown in figure 8:

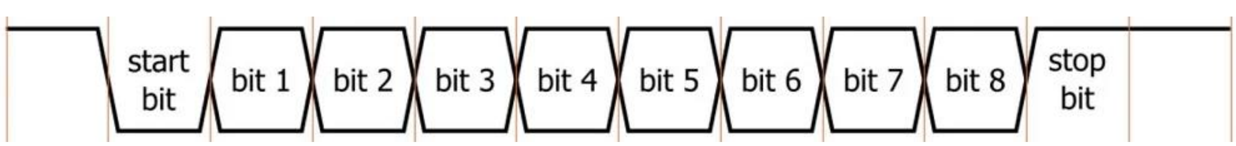


Figure 8: How UART packet works

By taking the logic levels low after being high at idle, the system knows to start reading data. By having a set length packet with a stop bit, the receiver knows when to stop reading data.

The only issue that UART now has, as opposed to a system with a dedicated clock channel, is timing. To be sure the system works properly, both Transmitter and Receiver need to be operating at the same clock speed and baud rate. Baud rate is simply the inverse of the period of 1 bit.

The microcontrollers on this VLC system use the same chips and clock speeds, and to make sure they send and receive data correctly, the same baud rate (number of transmitted bits per second) is chosen in both the transmitter and receiver's code.

3.1.3 Encoding

Another process that is commonly applied to digital signals is encoding, or using different patterns to represent the 1's and 0's. For a transmission like 10000011, we can see that there is a string of 0's in a row. For some electronic systems, this may be hard to read, since the data stream would remain unchanged for a long period of time while sending the five 0's for instance. Some forms of encoding help with this issue by always having a bit change for every bit transmitted.

Manchester encoding uses these bit changes by sending a 1 as 01 and a 0 as 10. Every bit now gets two bits to represent it after the encoding process so for data like 10000011, it would be sent as 0110101010100101. Note the length of the data transmitted has now doubled, which may be viewed as a negative. Also note that there is no chain of the same binary number longer than 2 (as opposed to a chain of five in the original byte). Finally, note that this is purely the data, and does not include any start or stop bit, or any other special bits used in the communication protocol.

3.1.4 Signal inversion

Like previously mentioned, the idle state is a logic level high. In this VLC system, the signal is inverted so that the idle state is logic level low. This helps preserve battery life by keeping the LED off and only turning it on for short pulses when it transmits data.

4.0 VLC System Design

A VLC system consists of two main parts, a transmitter and receiver. The transmitter must modulate a light source and send that light over a distance to reach the receiver. The receiver then takes the light signal and decodes it into something usable. This is the most basic form of a VLC system and one that only uses unidirectional communication. This means that only the transmitter can communicate with the receiver, and not the other way around. Many digital communication systems use bidirectional communication, where each side is a receiver and transmitter or transceiver and can both send and receive data. This is useful for handshaking and error checking between the two sides.

In the VLC system discussed in this paper, there is only unidirectional communication, which means there is no way for the receiver to communicate back with the transmitter. Having unidirectional communication keeps the system simpler for testing. To incorporate bidirectional communication into this system, the hardware would essentially be doubled.

4.1 Hardware

A high-level schematic of the developed system is shown in figure 9.

Figure 9: High Level VLC system Overview

Each box in figure 9 can be broken down into its components. There have been many iterations and the following subsections will go over the final prototype version and only reference older versions as needed.

4.1.1 Transmitter Electronics

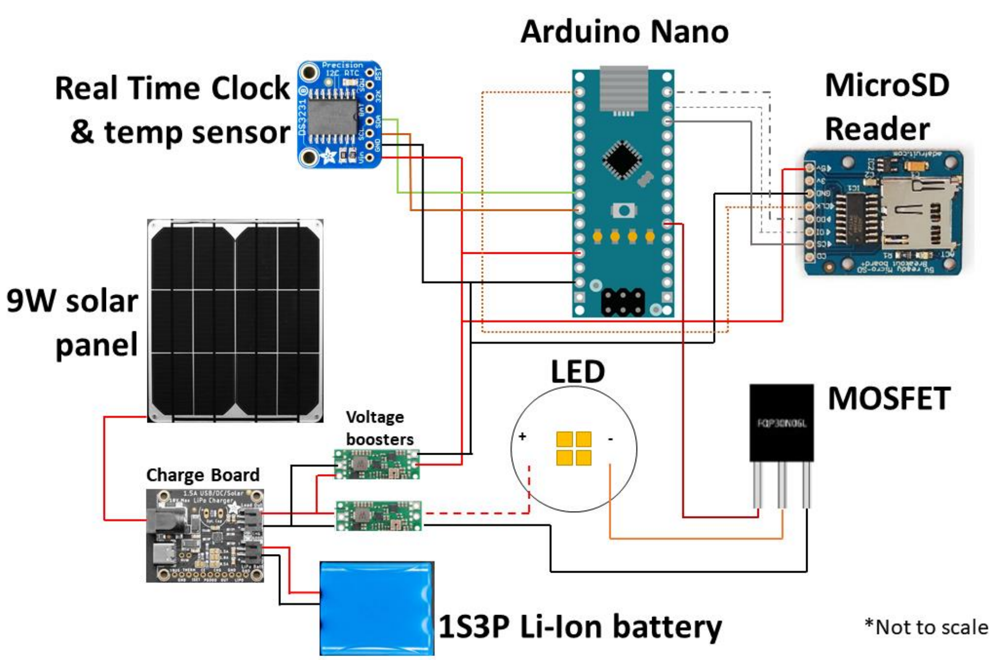


Figure 10: Transmitter Circuitry

This project uses an Arduino Nano microcontroller to control the transmitter. The code on the Arduino modulates one of the pins which is connected to a MOSFET transistor. The transistor then modulates the transmitting LED. There is an SD-card reader on the transmitter, which takes a copy of the transmitted data packet and stores it onto an SD-card. This SD card can then be removed and viewed later. There is also an SD-card reader on the receiver side so that the transmitted and received data on the two SD-cards can be compared for errors in the VLC transmission.

The transmitter also has a real-time clock (RTC) module, which is essentially a digital quartz clock. This module will keep the real time and date and send it to the main micro-controller. In the event of a total power loss, the onboard watch battery in the RTC will continue to keep time. When the power comes back on, the micro-controller will continue to run its code and also know the exact time.

For power management, the system was initially designed to last 1 week on a full charge. Solar was incorporated to increase the battery life greatly, with the potential of never needing to remove the battery to charge, depending on the presence of the sun. A 9-watt solar panel is connected to a solar charge board. This board does near max point power tracking meaning that as the sunlight hitting the panel changes throughout the day, the board will adjust its internals to always get the most power from the panel. The solar charge board is connected to a 6.6Ah lithium-ion battery. This battery pack is 1S3P 18650 cell pack meaning it has 3 standard 18mm x 65mm cylindrical batteries in parallel giving a total voltage of 4.2V and a capacity of 6600mAh when fully charged. This pack is not a high enough voltage to run the Arduino, so a voltage booster, MT3608 DC-DC converter, is provided to increase the voltage level to 5V. The LED runs at around 12V, 1A at its full power, so another voltage booster is provided to produce this voltage.

The draw of roughly 12W means the voltage booster must receive a current of about 3.33A at 4V due to conservation of power and efficiency losses. This would heat up the voltage booster and LED if the LED was on for anything longer than 10-15 seconds, but like previously mentioned, the signal is inverted, so the LED is off for most of the time.

With an Arduino Nano as the micro-controller, the transmitter uses about 4.6Wh of energy per day. This value would change with the length of the data packet, more data would have the light on for longer bursts and thus use more energy. The 9W solar panel theoretically outputs enough power to charge system in a little over 1 hour; however, from testing this winter, the solar panel was shown to rarely output 9W, even in direct, cloudless sunlight. In those conditions, the panel put out roughly 7W and less in weaker daylight.

During initial testing, the panel was oriented vertically in a window, normal to the earth's surface, and only exposed to direct sunlight for about 3 hours per day before shadows from nearby buildings cast over the panel. With the Arduino Nano as the micro controller, the system has been shown to transmit a data packet roughly 250 bits long every second for 2 weeks without shutting off.

Further, more realistic testing was performed on the roof of the engineering building of the University of Pittsburgh with the electronics inside of a weatherproof enclosure. The system lasted 11 days before it was pulled from the roof and data showed that the system charged the battery to full capacity each day meaning it could last in those conditions indefinitely.

Table 1 List of Parts Used for Transmitter

PART	PART NUMBER
LED	Cree XHP-35.2
Microcontroller	Arduino Nano
MOSFET	FQPF30N06L
Op-Amp	LTC6269
Solar Panel	Adafruit 2747
Battery	Adafruit 5035
Charge board	Adafruit 4755
Voltage boosters	MT3608
Real time clock	Adafruit 3296
SD card reader	Adafruit 254

4.1.2 Transmitter optics

The optical system on the transmitter is rather simple. The LED sits on a heat sink designed for 3D printers. The heatsink also has a fan, but in the system's current state, active cooling is not needed, so the fan remains off. This whole LED and heatsink assembly sits atop a custom 3D-printed tower that raises the LED to the center line of a large Fresnel lens.

There are currently two slightly different versions of the custom tower. One is designed for the most airflow to the heatsink and is not as user friendly to adjust height. It is more robust for having more fasteners. It is designed to go in a NEMA rated enclosure for an outdoor field test at the National Energy Technology Lab in Morgantown, WV. The other one has slightly less airflow accessible to the heatsink but is more user friendly to adjust and is designed as a demo to showcase the VLC technology at the University of Pittsburgh (Pitt). Both custom towers have vertical adjustability, and the demo system for Pitt also has forward/aft movement, whereas the NETL system has the forward/aft movement built into the lens brackets. This allows the LED to be perfectly centered at the focal length of the lens. The Pitt demo system is shown in figure 11.

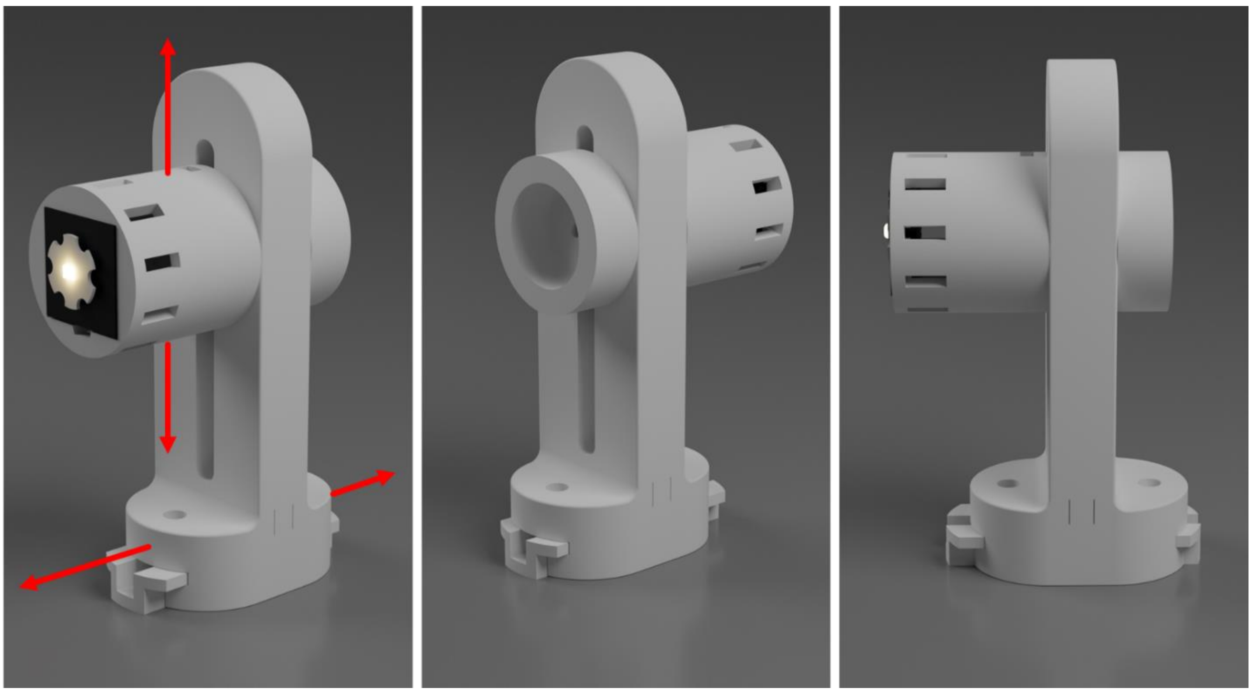


Figure 11: Transmitter tower for University of Pittsburgh

The tower slides upon the U-shaped track that is fixed to the base and can be tightened in place. The horizontal cylinder can move vertically on the upright slot and can also be tightened in place. The square shaped cutout fits the black heatsink and there are vents to allow air to move freely through the heatsink fins. Active cooling is not necessary for both systems, although like previously mentioned, the NETL system has a bit more freedom for the air to reach the heatsink since it will be outdoors and temperature will vary more than the indoor demo unit. It is show in figure 12.

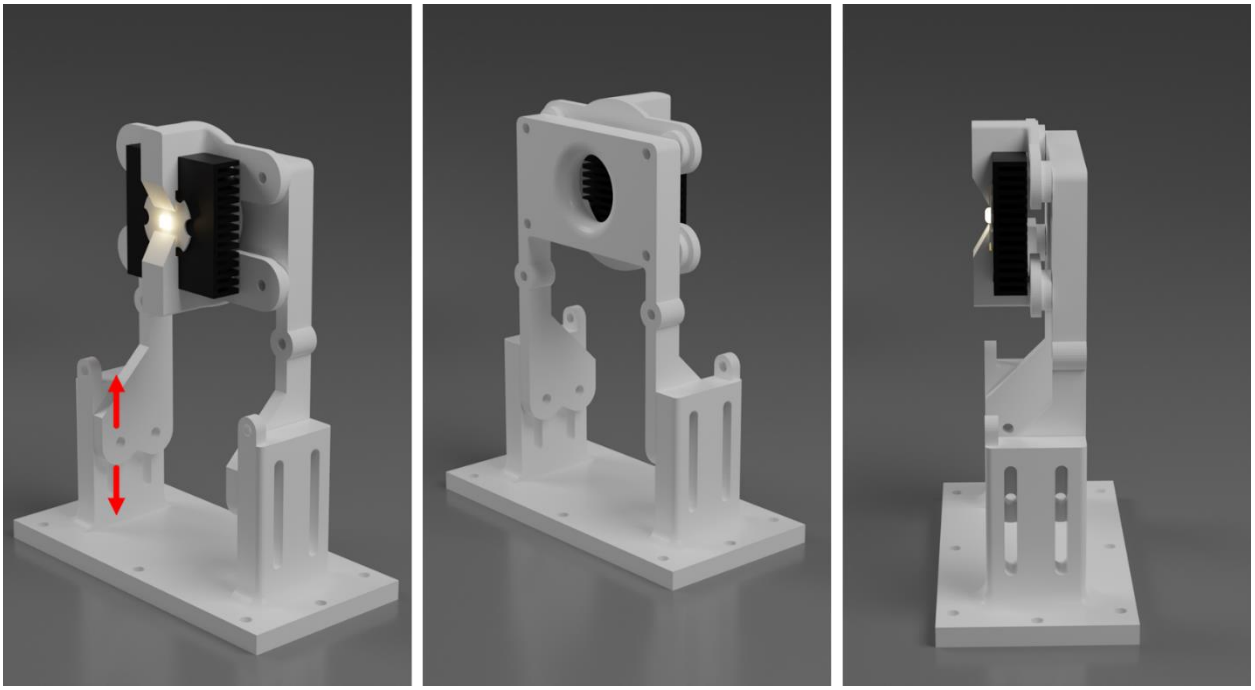


Figure 12: NETL Transmitter tower

The Fresnel lens is 300mm in diameter and 360mm in focal length. A simple convex lens would work, but to reach the goal of transmitting over 100m, the lens would have to be very large and heavy. By using a Fresnel lens, the lens can still have a large diameter and focal length and be relatively light weight at 190 grams. It is shown in figure 13.

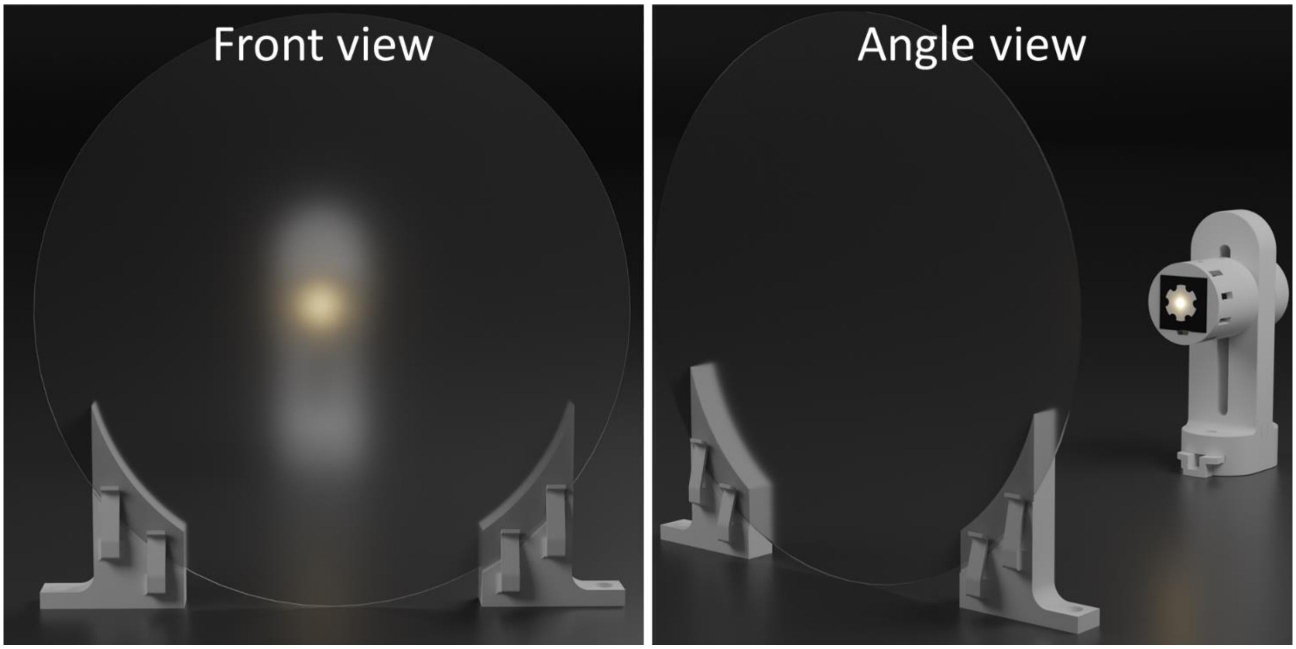


Figure 13: Pitt Transmitter Lens

Again, there is a slightly different version of this for NETL. To fit inside the stock NEMA enclosure's dimensions, top and bottom portions of the lens had to be cut off to save size. Furthermore, the actual aperture that the light exits from the enclosure is a circle with diameter smaller than the lens. This means that the total light passing through the lens is greater than the total light exiting the enclosure, since some of the light passes through the sides of the lens and doesn't exit through the aperture. These rays bounce around inside the enclosure until they dissipate. The internal surface of the enclosure is painted black so that the rays dissipate quickly. Additionally, the distance between the LED and the lens is adjusted at the lens by sliding the brackets along the aluminum extrusion and tightening in place. This is shown in figure 14.

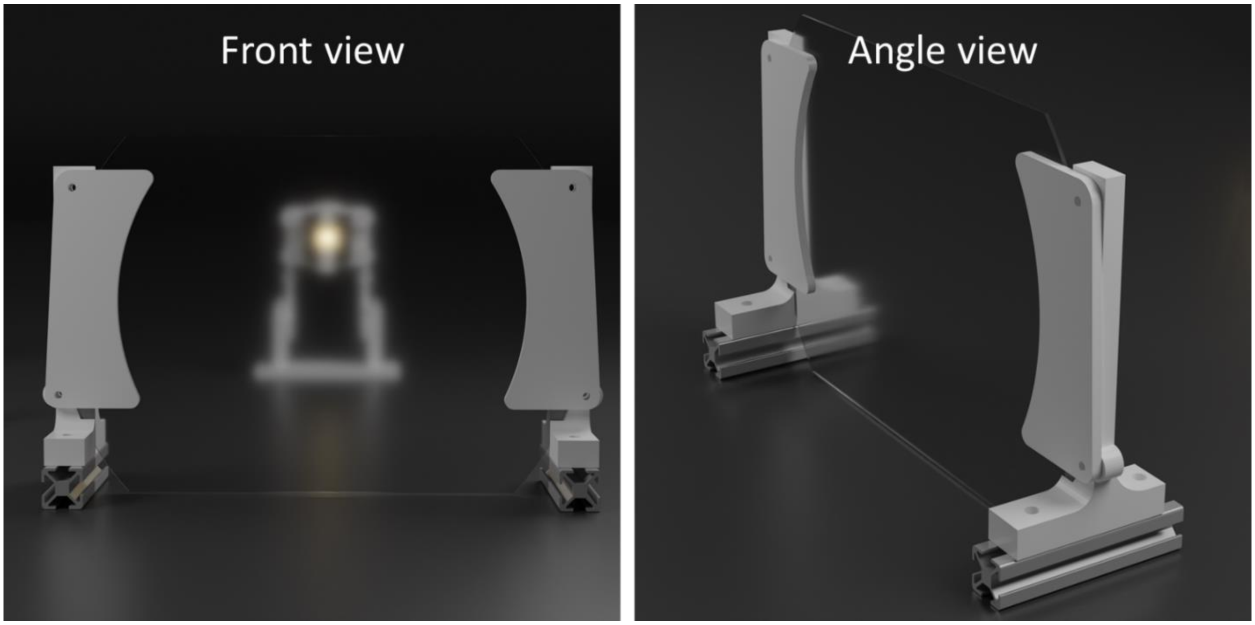


Figure 14: NETL Transmitter lens

A few issues are introduced by using this lens, however. The first is that the lens may have some warp in it which can affect the beam. This can be solved by putting the lens in brackets to keep it planar. The second issue is that the lenses might not exactly have the focal length that they should due to poor manufacturing. For instance, if the lens should have a focal length of 360mm, it may actually range anywhere from 355mm to 365mm. This can be solved by making the system adjustable to add focusing capability. The third issue is the Fresnel lens profile. Because this Fresnel lens uses a basic spherical convex lens profile, the light that passes through it will suffer from chromatic aberrations, giving a prismatic effect. This is due to different wavelengths of light having different focal lengths. By using a single lens with a fixed focal length, different wavelengths of light will diverge at different rates. This shows up practically as the beam becoming different colors throughout its length. Since the only part of the beam that is important is the final part that hits the receiver, it is not important that the beam changes colors throughout, but only the beam's color when it reaches the receiver. Because the center part of the beam remains a near white while the edges change dramatically, this problem is mostly mitigated when the receiver is placed at the center of the beam.

A mathematical model was developed which uses optics equations 1, 2, and 3, geometry of the system as well as test data to predict the brightness of a light beam at a particular distance given an LED's parameters and a lens' parameters. This model was then used to find the best focal length lens to use for the system.

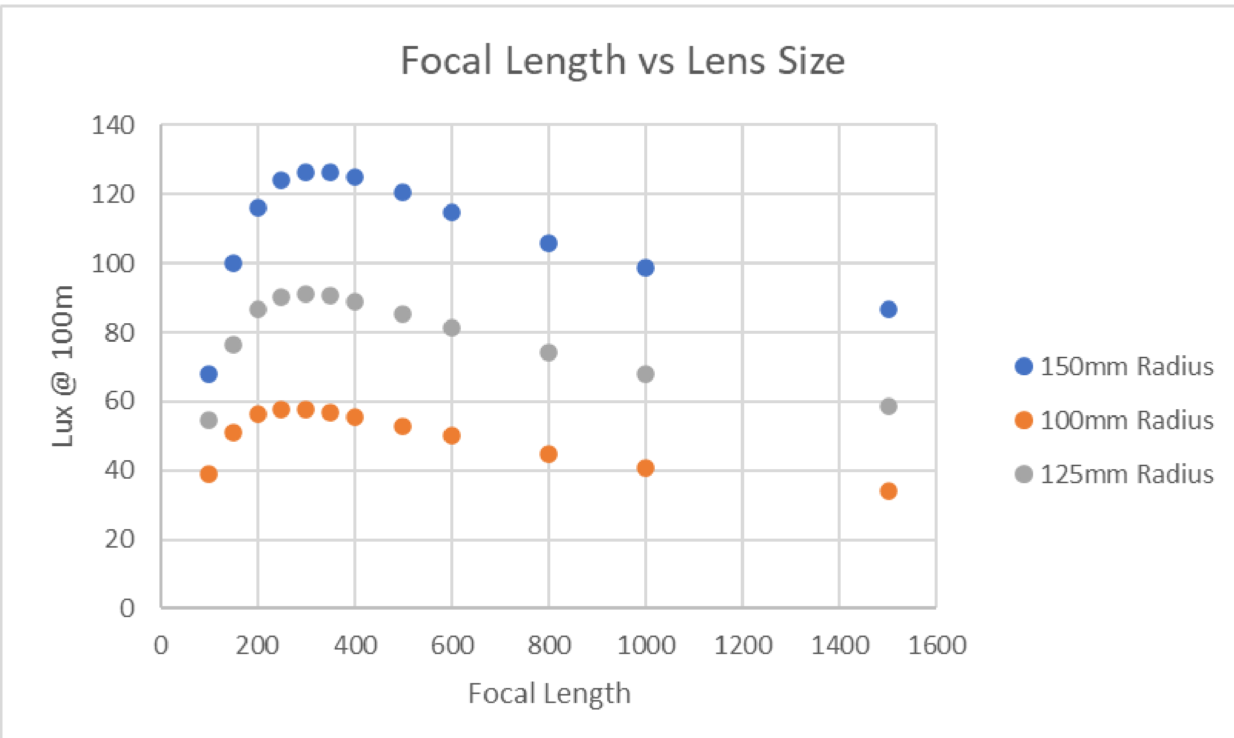


Figure 15: Model Predictions about Focal length

Figure 15 shows what the model predicts will be the brightness of a beam at 100m (the design distance) for different lens diameters and focal lengths. Of course, the bigger the lens in terms of diameter, the more light is captured and thus the brighter the beam. Going from orange to grey to blue in Figure 15 shows exactly this. Within each set of colored points, there exists a pattern. From left to right the focal length of the lens is increasing and the lens is always placed exactly 1 focal length distance away from the light source. Clearly, when the lens is very close to the light source, it will capture much of the light. However, due to equation 3, it will have a widely diverging light beam that will not reach far. On the other side, if the lens has a large focal length, it will not capture much of the light from the source, but due to equation 3, what light is captured will diverge very little and will reach far. There exists an optimum focal length between these two extremes that changes depending on lens diameter. Furthermore, the supplier of Fresnel lenses only sells in discrete focal lengths. So, it is important to use a lens that is available for purchase and fits closely to the peak of roughly 350mm focal length. This is how the 360mm focal length lens was settled upon.

4.1.3 Transmitter enclosure

The enclosure that houses all of this must meet certain functional requirements. It must be large enough to fit everything inside, strong enough to withstand vibrations, wind, external forces, etc. and weatherproof to be kept outside for long periods of time. The enclosure should also not obtain unnecessary heat from the sun. This can be achieved by adding a shade layer above the box and painting the box white. Additionally, this enclosure of course needs to have a transparent panel to allow the VLC light beam to exit, and it should minimize infrared light from the sun entering the box. These characteristics have all been considered when creating this enclosure.

4.1.4 Receiver Optics and enclosure

When the system is operating outdoors, ambient light can vary wildly in magnitude. It can go from the darkness of night (~3lux) to a clear sunny day (~100,000 lux). Given the fact that the VLC system achieves a brightness of about 80 lux at 100m, the ambient light can be magnitudes higher than the VLC beam. This is difficult because the system must be sensitive enough to read the VLC signal at 80 lux, but also have a large enough dynamic range to not become saturated by the ambient light. The first step in solving this issue is to pass all light through a sheet of infrared blocking glass before it goes into the enclosure. The next step is to magnify only the VLC beam and not the ambient light. This is done by placing a coaxially aligned lens in front of the receiver's photo diode. If a ray of light is coaxially aligned with the lens, it will pass through and reach the receiver's photodiode. If light comes into the receiver system at some angle off axis (i.e. starting above or to the side of the transmitter) then when it passes through the lens, it will be magnified and projected somewhere other than the light sensor. Figure 16 demonstrates this.

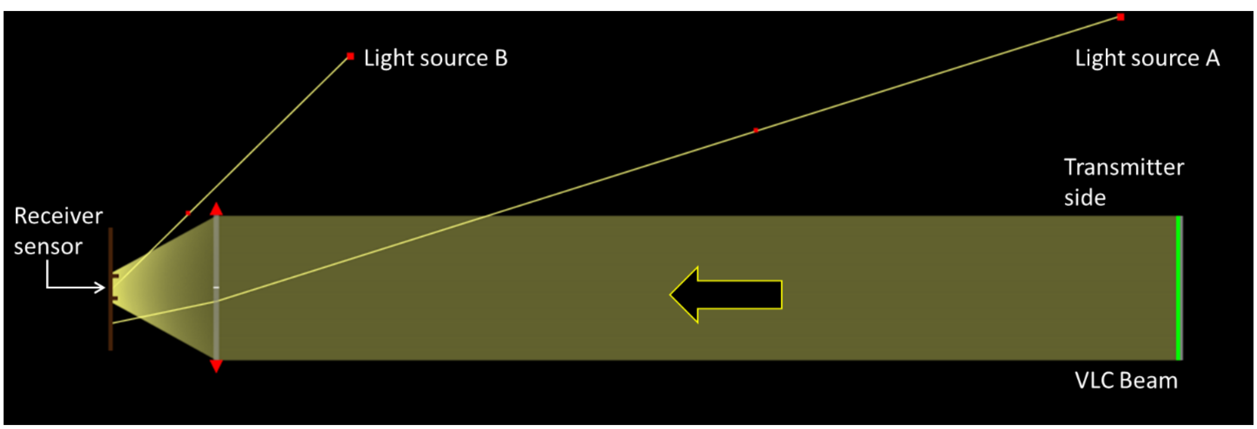


Figure 16: Path of Ambient light onto Rreceiver

As can be seen from Figure 16, the VLC beam starts on the right and enters the receiver lens. After approximately parallel transmission, it gets further collimated down to the receiver sensor. Now look at what spurious *light source A* does (e.g. the sun). It also enters the receiver lens, but it does not hit the receiver sensor because it is not coaxially aligned with the VLC beam. Now look at *light source B.* It also starts above the entire VLC system and is not coaxially aligned, but it does in fact hit the light sensor. This can be solved by placing the receiver components in a box, creating a "hood," which is routinely used in photography.

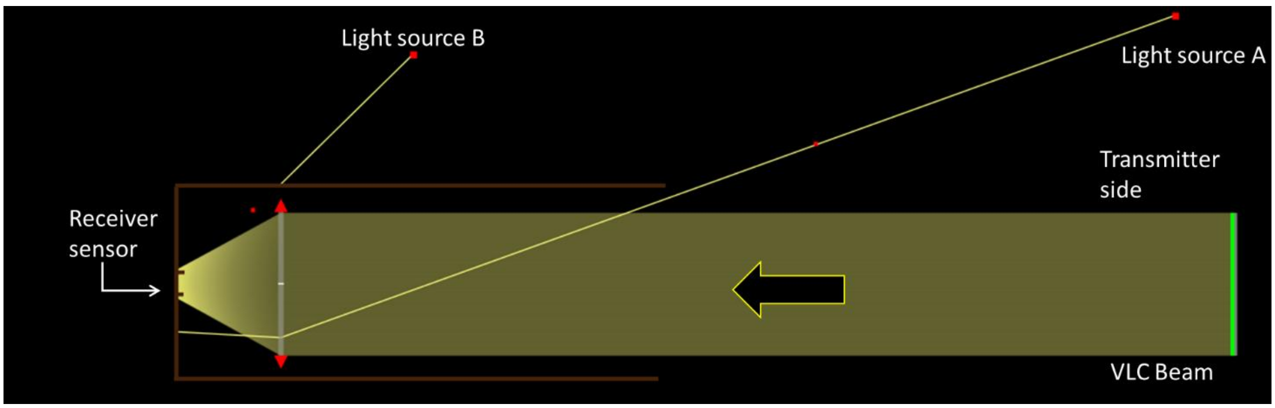


Figure 17 Path of Ambient Light onto Receiver with Enclosure

Looking at this system, we can see that light source B does not hit the sensor anymore. In fact, most of the ambient light now will not hit the sensor unless it comes from extremely near the VLC transmitter source. However, there is an added issue with an enclosure: light can bounce around inside the enclosure and eventually find its way to the sensor. Figure 18 depicts and exaggerated version of this.

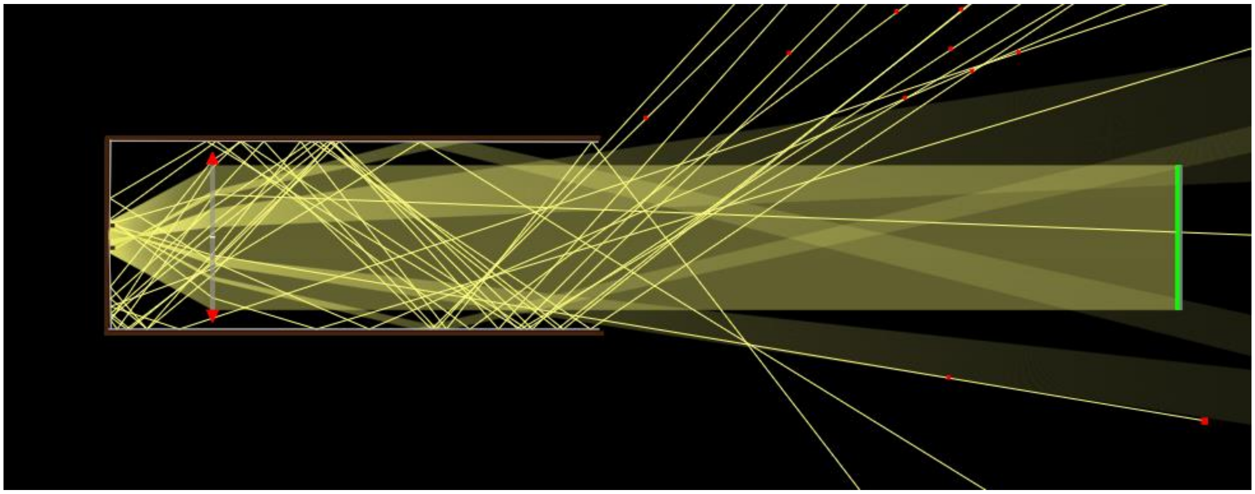


Figure 18 Receiver Enclosure with non-absorptive internal walls

Figure 18 shows a box whose internal walls have not been treated with an absorptive coating. The rays of ambient light will enter the box and have the potential to bounce around and hit the light sensor. To prevent this, a few layers of a special paint called *Black 2.0* are used. In testing, this paint was shown to absorb 96% of incident light. With this long enclosure and special paint applied, light levels inside the box are up to 99% lower than ambient light. For example, on a sunny day, light levels reached about 92,000 lux. During that test, the internal light level of the receiver enclosure was about 100 lux, a reduction of greater than 99%, shown in figure 19.

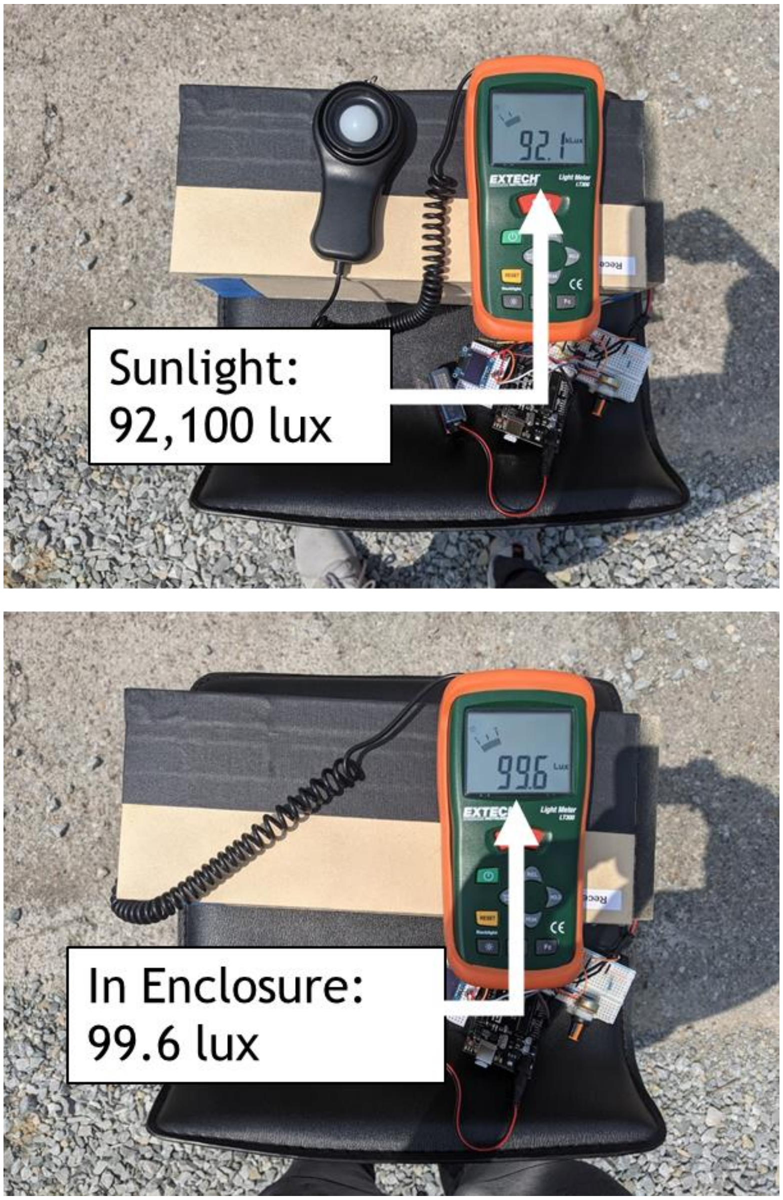


Figure 19 Light levels inside and outside the enclosure

4.1.5 Receiver Electronics

The electronics system used for much of the prototyping process was simply based on a photo transistor voltage divider circuit. This circuit output a voltage that is proportional to the amount of light on the sensor. That voltage was fed through a comparator which compared the rising and falling voltage signal from the VLC beam to a constant intermediate voltage. When the receiver voltage was above a certain threshold, the comparator output a logic high and vice versa for when the receiver voltage was below that same threshold. This system worked with most ambient light levels, as long as they remained relatively constant. For a test on a cloudy or sunny day when the ambient light did not change much, this system would operate fine because the user could set the intermediate reference voltage to which the signal was compared. The circuit was simple and had very few components to get working properly.

The downsides to this system are that it responded relatively slowly, so the max communication speed was 38,400 kb/s. The other downside to the system is that it did not work in a wide range of ambient light conditions. The enclosure blocked out 99% of light, however the remaining light that it did not block could be on the same order of magnitude of brightness as the VLC signal brightness (100-300 lux). Thus, a large downward swing in ambient light from 90,000 to 10,000 lux would result in a corresponding swing inside the enclosure of about 100-200 lux. This would drastically change the signal that gets received.

Any ambient light that gets added or removed from the sensor vertically shifts the signal either up or down. As long as the sensor is not saturated with light, the AC part of the signal (which contains the information) is conserved.

To solve those two issues of speed and ambient light swings, the current system was developed. This system uses a trans-impedance amplifier, a common circuit used with photodiodes detecting light levels. It also incorporates filters to smooth out high frequency noise as well as a DC blocking capacitor that centers the signal about 0 volts, regardless of the ambient light. A schematic is provided in figure 20.

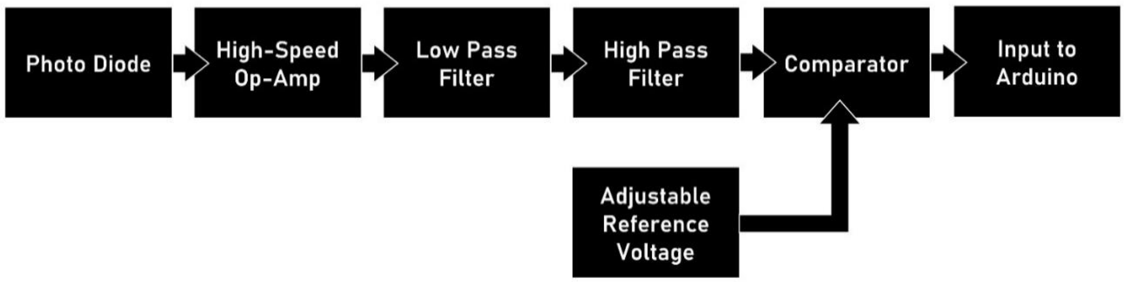


Figure 20 Receiver Circuit Block Diagram

The magnified VLC light beam falls onto a high-speed photodiode (PD) with a spectral sensitivity similar to that of the human eye. The photo diode is connected to an operational amplifier circuit which converts the PD's current output to a voltage that is proportional to the light falling on the sensor. The signal coming from this circuit is noisy and heavily affected by ambient light levels. The next step is to remove the high frequency noise with a passive low pass filter. The, the signal passes through a high pass filter which centers the signal about 0V. This filter removes any DC bias from ambient light. From there, the signal is sent into a comparator which takes another user-adjustable reference voltage with which to compare the signal voltage. The comparator has a very fast response, so the output from this is a very clean, square shaped signal. This can then be put into the Arduino to be decoded and read. The receiver has an SD card and reader to store all the data it receives.

Some versions of the receiver enclosure have had temperature sensors to monitor the temp of the enclosure. It is useful to monitor this factor especially for use outdoors.

4.2 Software

The following software section is similar to the Digital communication section but with specific details about the software on the Arduinos.

4.2.1 Transmitter Software

The software on the transmitter consists of reading sensors and outputting serial communication. The Arduino micro controllers have built in chips and I/O pins on the board to handle basic serial comms. The platform also has libraries for serial communication; however, these libraries are for standard UART, which mentioned previously holds its idle state high. To invert the signal and have the idle state low, another library is used called Software Serial.[3] This allows the Arduino to use other digital I/O pins instead of the built in TX and RX pins and also allows for signal inversion at a software level.

The Arduino takes in data from the sensors such as real time and date as well as any actual data needing to be sent such as temperature or noise levels. Once all the data is gathered it is packaged in a specific way with each piece of data separated by semicolons. Then it adds two more pieces of data separated by semicolons that is not gathered from a sensor.

Each character has a binary and decimal representation agreed upon in the American Standard Code for Information Interchange. The code runs through each character of the packet it creates and adds up each character's decimal representation. It tacks this new calculated number at the end of the packet. It is called the calculated length and will be used by the receiver for error checking. Finally, it adds a known character at the end of the packet that is also used by the receiver for error checking.

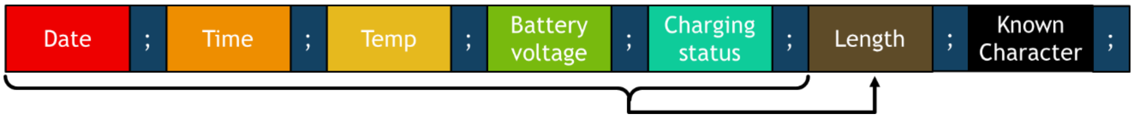


Figure 21: Transmitted Packet

The Software Serial library takes this data and converts it to binary pulses one of the Arduino's digital I/O pins which is then connected to the circuitry.

The Arduino also has an SD card module on the transmitter side to write the data to an SD card for further data analysis. The transmitter will most likely always be able to write to the SD card, but it may not always be able to transmit over VLC. The transmitter side SD card acts as a backup to make sure the data is always gathered. For our purposes, it also permits validation of the received data to determine the data transmission and receiver performance of the system.

4.2.2 Receiver Software

On the receiver side, the code is a bit trickier. The same set-up is required on the receiver so that the transmitter and receiver communicate well with each other. The Software Serial library and baud rate are all specified. Then, the Arduino runs a command which returns if there is signal is idle or not. Since idle state remains low, as soon as the LED goes high, the receiver sees it and a start bit is signified. The system starts receiving.

The system receives binary data that it converts to the decimal system. It looks up each decimal number in the American Standard Code for Information Interchange character chart and gets its corresponding character. Once it has received a complete packet, it checks first if there is the known character at the end of the packet. If there is, it will continue to the second error check which is to calculate and add up the decimal value of every character in the packet, called the calculated length. Once it has this, it separates the packet by the semicolons and compares the calculated length to the received calculated length. If the calculated length and the received calculated length are equal, then the system most likely has not created an error, and it can open the SD-card file and store that packet of data. After this, it closes the file and starts the loop over.

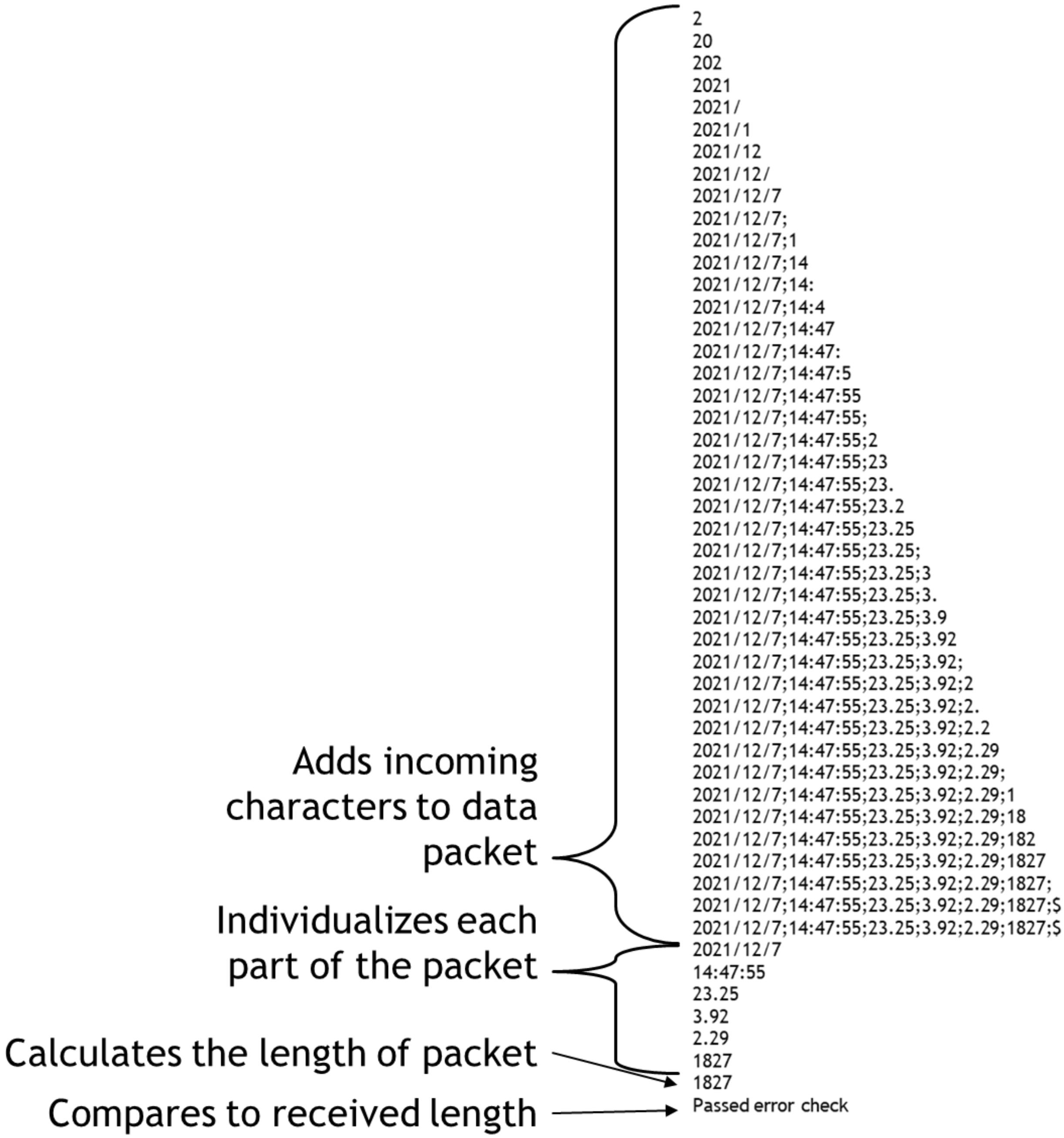


Figure 22 Receiver Receiving and Checking for Error

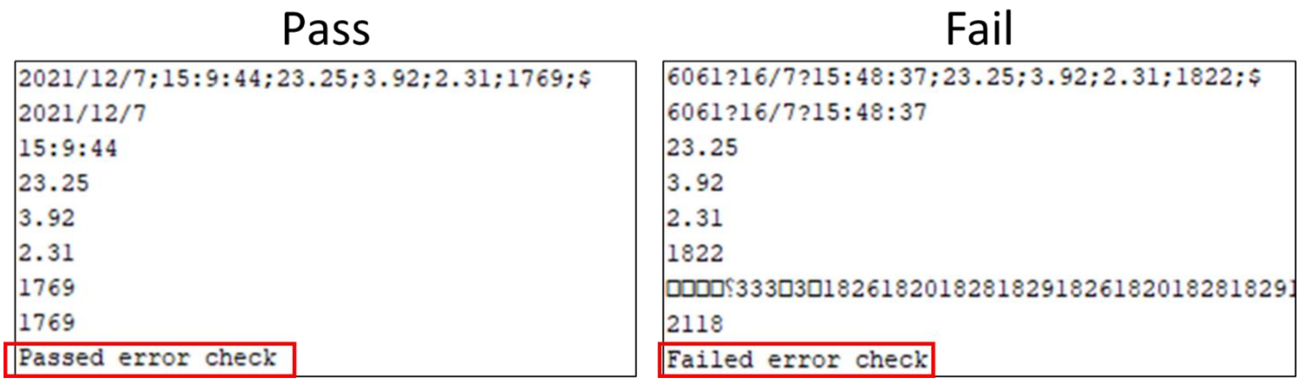


Figure 23: Example of a Pass and Fail Error Check

Figure 22 shows a good example a passed and failed error check. The failure occurred because the date was received incorrectly. The 2's were received as 6's, and /'s received as ?'s. If only the known character at the end was used as an error check, then the error checking would not be sufficient because the data on the right received the final "$" correctly. This means it would pass the 'known character' error check but fail the calculated length error check.

5.0 Model

The theoretical model that was developed to help predict beam characteristics has been mentioned throughout this paper. This section will go over how the model works.

The model was developed in excel as a spreadsheet and it accepts 6 inputs which it uses to determine a beam's brightness at long distances (~100m).

Single Lens MODEL					
Mathematical model					
inputs			outputs		
EFL	360	mm	**Angle of Capture**	15.52	degrees
Lens Rad	100	mm	**Lumen Capture**	73.67382	lumens
Emitter Dia	3.45	mm	**Half angle of Div.**	0.2745	degrees
LED Lumens	1270	lumens	**Inner Beam dia.**	0.967924	meters
Distance	101	m	**Outer Beam dia.**	1.167924	meters
Lens Trans %	98	%	**Inner Lux at dist.**	100.1246	lux
			Outer Lux at dist.	68.76922	lux
			Inner beam weight	37	%
			Outer beam weight	63	%
			Atmospheric losses	19.19	lux
			Total Lux	61.1807	lux

Figure 24: Mathematical Model for Single Lens Ssystems

As can be seen, the model has the inputs on the left, and the final output on the right, highlighted. Some things to note: this particular screenshot of the model is for a system with a 360mm focal length lens with a radius of 100mm and a transmittance of 98%, an LED with an emitter diameter of 3.45mm and a luminous output of 1270 lumens and is calculating the beam's brightness at 101m. It says that this system will produce a brightness of about 61 lux at 101m.

The first step the model does is determine how many lumens from the total output of the LED the lens is capturing. The LED does not output light through space evenly, and it emits less light the further away the angle from center. Each LED has slightly different spatial distribution, and it can be found in the manufacture's documentation. The Cree XHP35.2 is shown in figure 25.

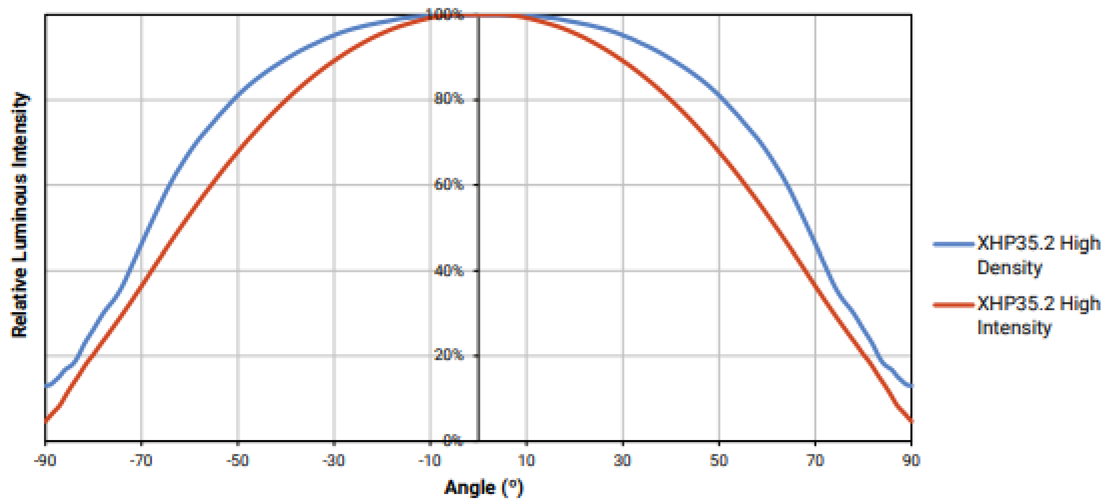


Figure 25 Spatial Distribution of Cree XHP35.2

The system uses the high intensity LED, the red line. From there, a 3D-revolution was created using the red line shown in figure 25.

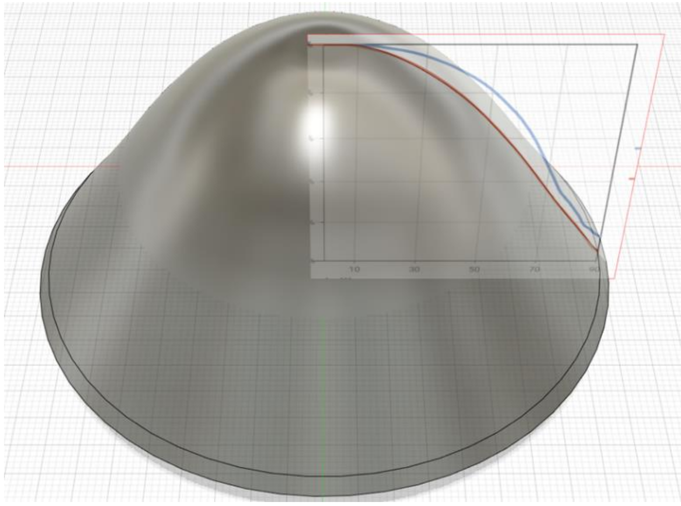


Figure 26 Spatial Distribution of Cree XHP35.2 in 3D

The volume of this shape represents the total amount of luminous output. From there, simple geometry is used to determine an angle from the LED to the edges of the lens. This angle determines how much of luminous output from the LED gets captured and transmitted by the lens. An example of this is shown in figure 27.

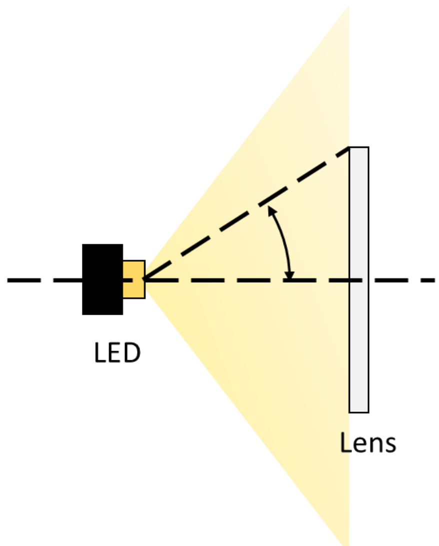


Figure 27 Angle Captured by TX Lens

The next step is determining the actual number of lumens that get passed through the lens by taking the angle of capture from the previous step and creating a new 3D geometry from the spatial distribution chart. This new volume is compared to the total volume which creates a ratio that tells us what percent of the total luminous output is captured by the lens (shown in figure 27).

After multiplying this lumen value by the lens transmittance (usually >95%), the total number of lumens passing through the VLC chain is determined.

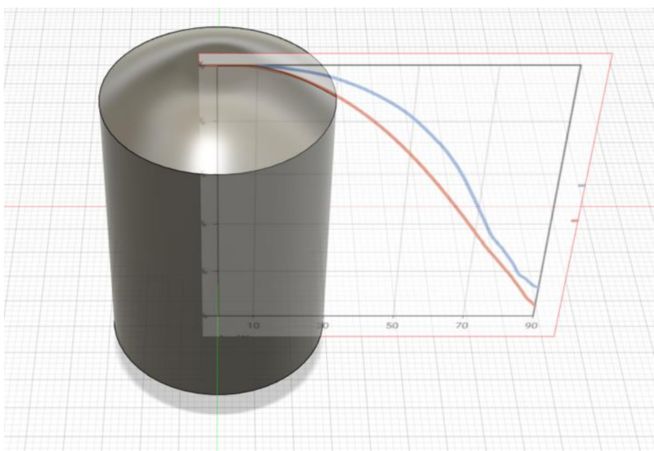


Figure 28 New Volume Created with Angle of Capture

The next step is using equation (3) to determine the angle of divergence of the beam. This then gives a cross sectional area of the beam at any distance. Once the area of the cross section and the total lumens is determined, equation (1) can be used to calculate the lux at any distance.

There are other heuristically determined factors that get considered when getting the final lux value. For instance, over 100m, there are some losses due to the atmosphere. They are small and get subtracted linearly from the final value based on distance. The amount that gets subtracted due to the atmosphere was determined by examining the intensity at different distances and noticing a small linear difference compared to the model with no atmospheric losses.

Another interesting note is that the beam has some odd characteristics, such as an inner and outer beams with weights for each. This came from a heuristic method of testing that found the beam to have a strange property when using these relatively inexpensive Fresnel lenses. There seemed to be two concentric beams, one starting with a radius of 0 at the transmitter lens, and the other starting with a radius equal to the transmitter lens. These two beams would then have the same divergence for the entirety of the beam. Figure 29 demonstrates this. Figure 30 shows real pictures of the beam over 5 meter intervals. The prismatic effect is also visible.

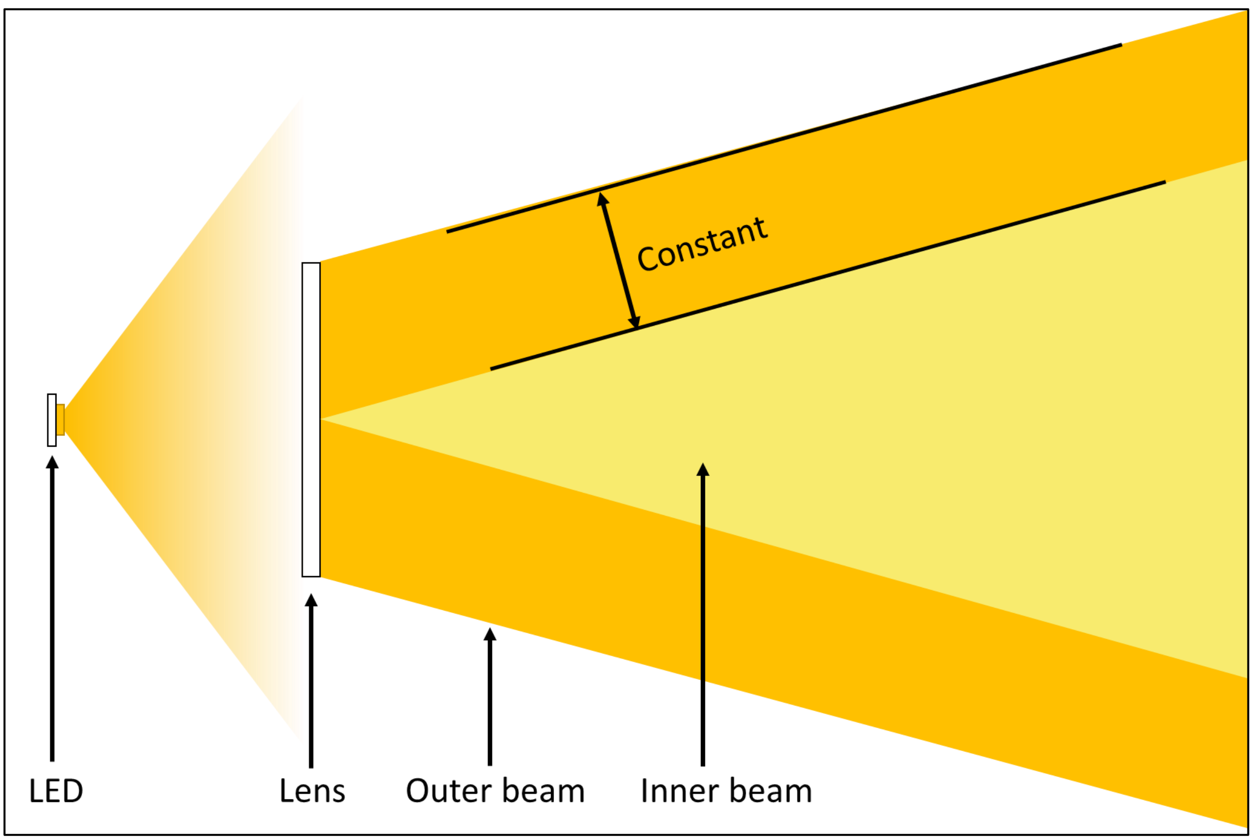


Figure 29: Inner and Outer Beam Formation

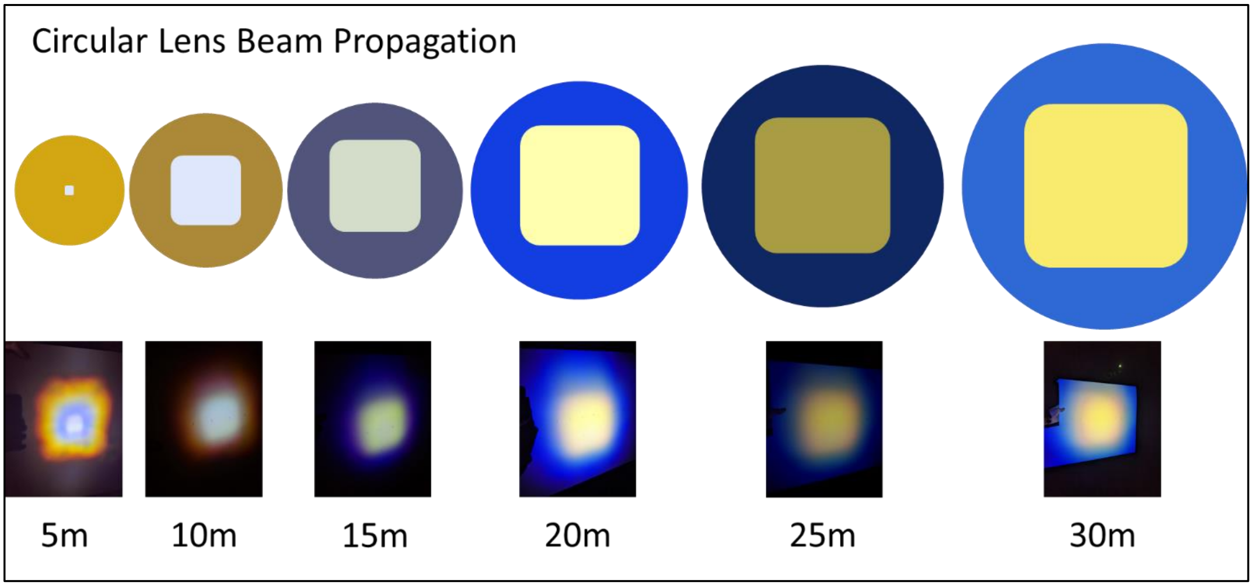


Figure 30: Circular Lens Beam Propagation at 5m Intervals

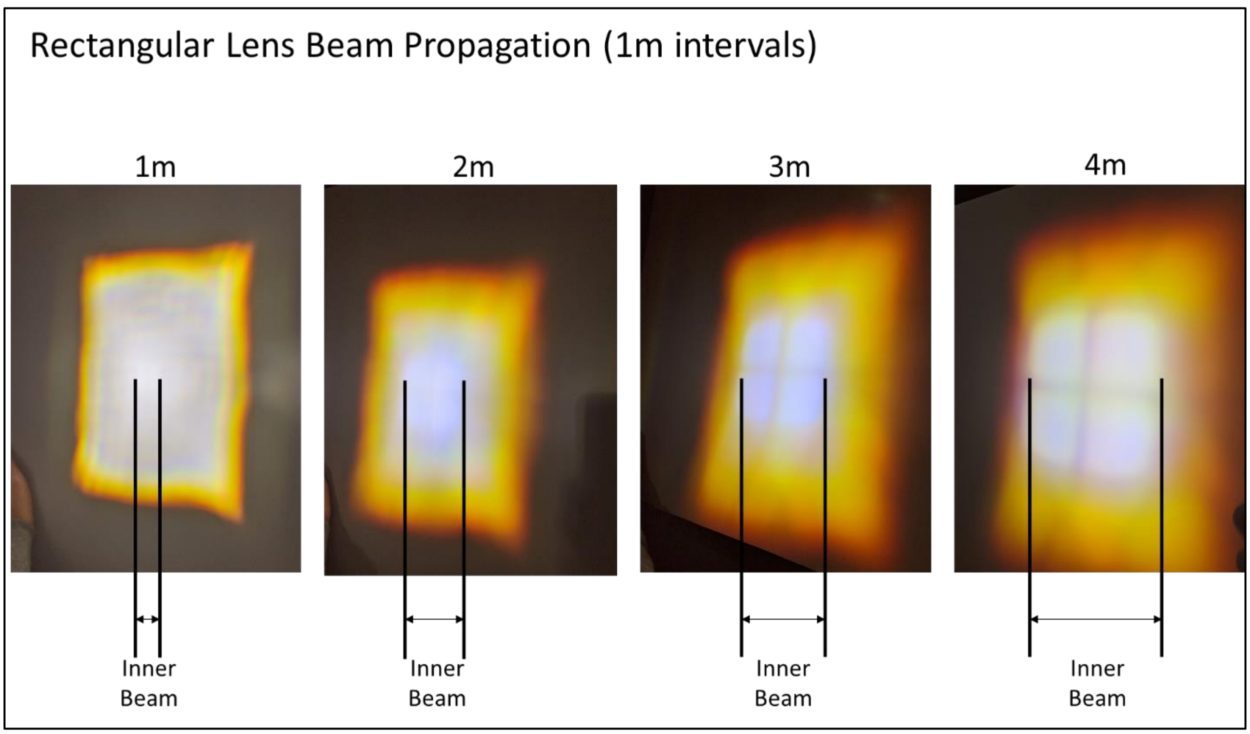


Figure 31: Rectangular Lens Beam Propagation at 1m Intervals. This light had 4 LEDs in a grid pattern, which are clearly visible in the last figure.

Figure 31 shows the same phenomenon at shorter intervals. The inner beam is noticeably brighter than the outer beam, and the final lux value measured at 100 meters usually falls in between the inner and outer beams' brightness. It is not exactly 50/50, and thus comes the weighting system. For instance, if the inner beam is 100 lux at 100m and the outer beam is 50 lux, the lux value measured at 100m would fall somewhere between 50 and 100 lux, not the average at 75, but perhaps 65 lux (disregarding atmospheric losses).

Now, we can examine the model's accuracy. After inputting the system's parameters, the model calculates the beam's brightness at any distance. From there, we can overlay the results of the experimental results with the model results, shown in figure 32.

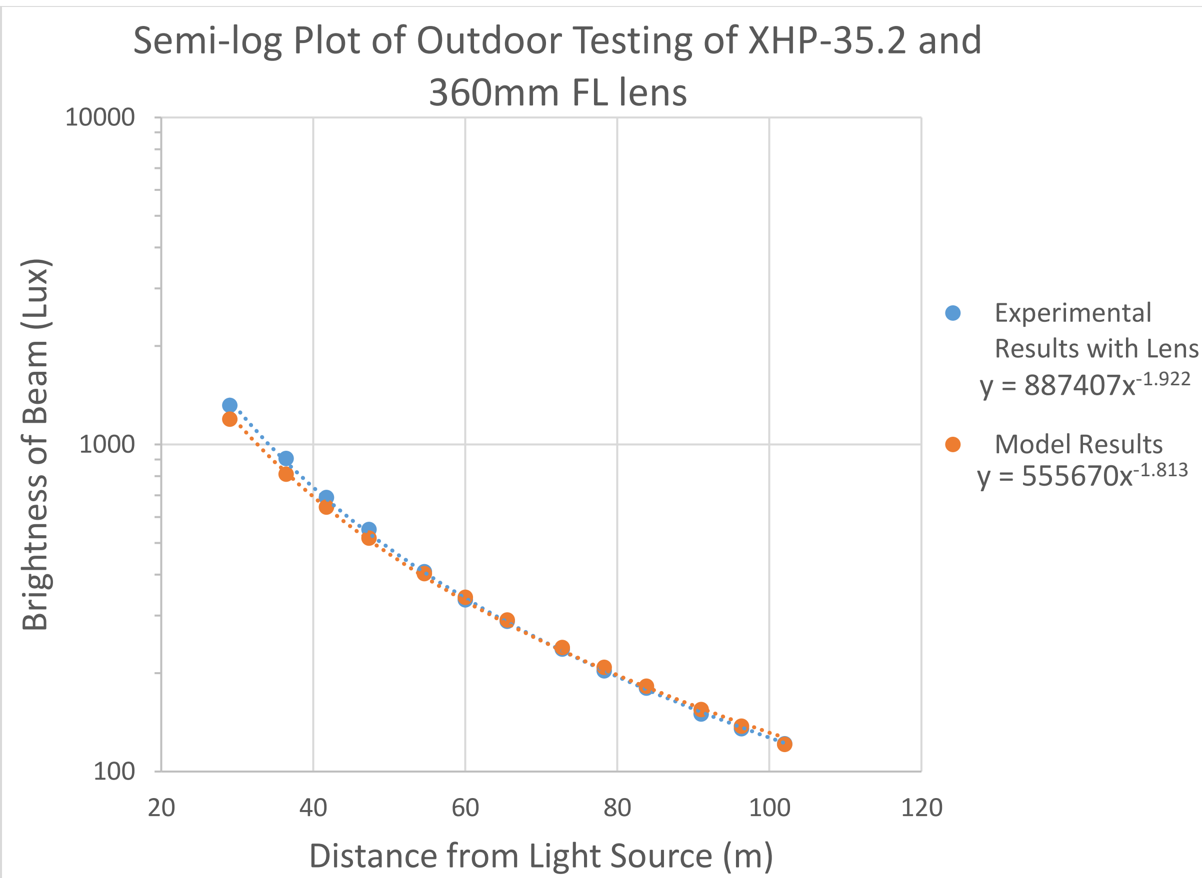


Figure 32: Experimental vs Theoretical Results

As can be seen, the two are matched quite closely, which makes the model useful in quickly finding an approximate brightness value (in lux) of a beam with a given set of LED and lens parameters.

6.0 Testing Results and Refinements

The system was tested in a variety of conditions during this process. The standout tests will be documented in this section.

6.1 Light testing and Modeling

Before communication testing could commence, it was important to characterize the light beam output with different optical set-ups. Many lenses and LEDs were tested. Special interest was paid to keeping the brightness of the beam bright at 100m while minimizing the overall power draw and size of the system. Testing involved indoor short-range tests with a lux meter as well as outdoor tests with a lux meter at night. All the data was then used to develop a mathematical model which was also based on equations 1, 2 and 3. The model was used to determine the best LED and lens set up that minimized over system dimensions and power draw while maximizing the beam intensity. Figure 33 shows some of the LEDs that were tested. Figure 34 shows some of the different lenses and reflectors that were tested.

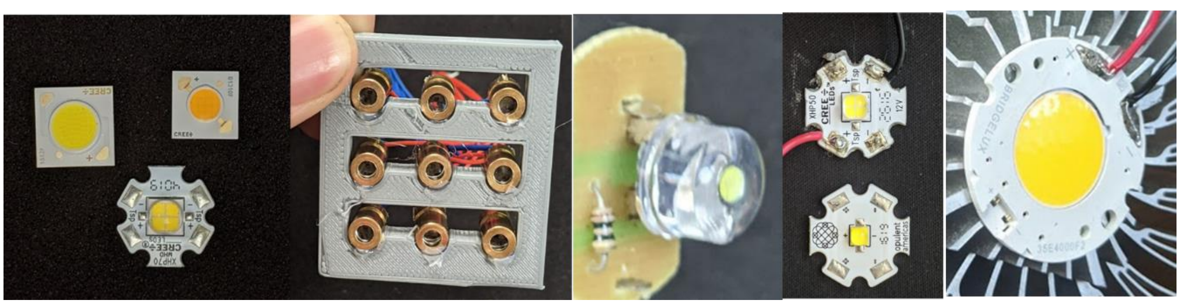


Figure 33: Some of the various LEDs that were tested

Figure 34: Some of the various tested lenses and reflectors

Figure 35 shows some of the outdoor testing that was performed. It is interesting to note the beam is visible at night as it interacts with the atmosphere. Some photos show rain and snow visible in the beam.

Figure 35:Various outdoor testing pictures at night

The model determined the best practical combination of LED and lens, and testing began on a new optical set-up. A Cree XHP35.2 was paired up with a Fresnel lens with a focal length of 360mm and a diameter of 300mm. Figure 36 shows the testing results of the best LED-lens set-up. Take note of the blue dot set which is the LED and lens pair, as compared to the grey dot set which is just the LED.

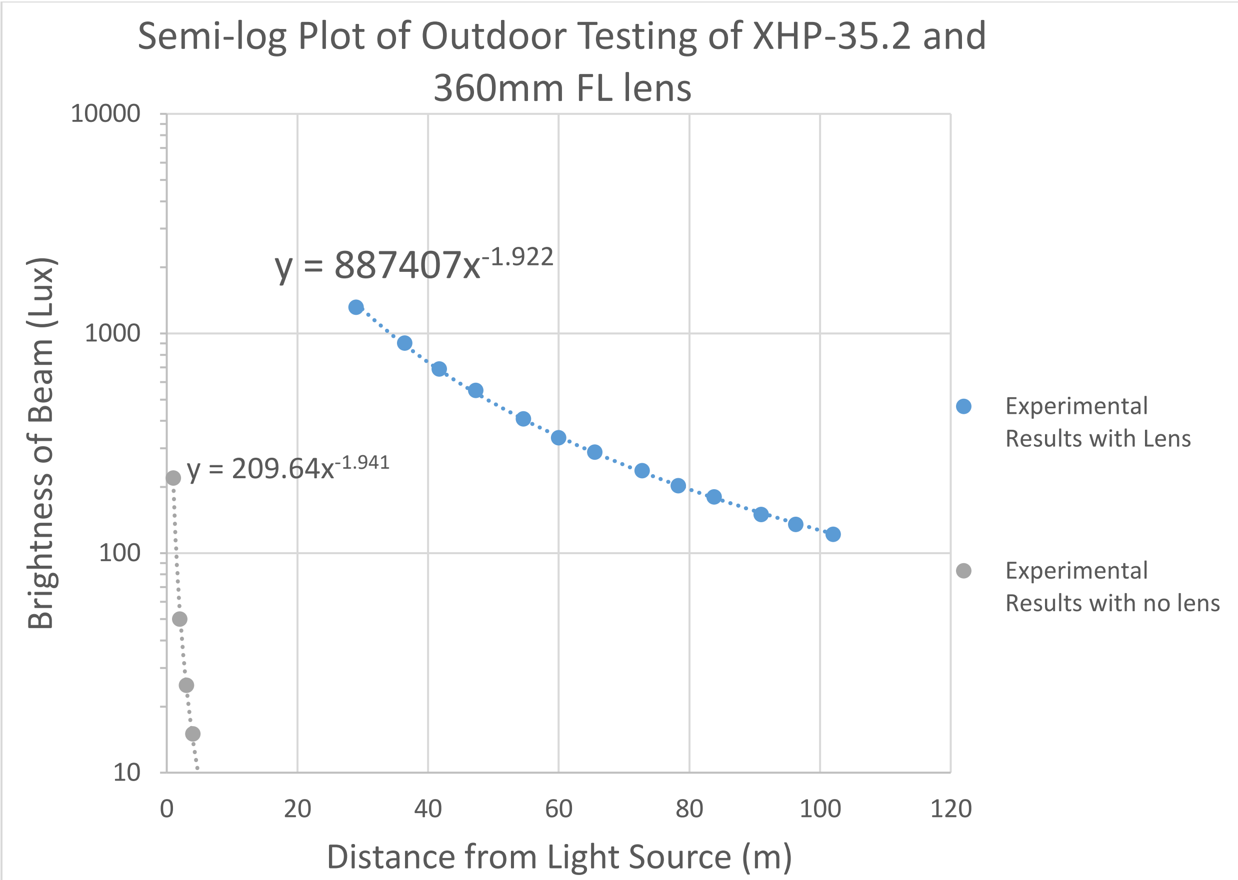


Figure 36:Outdoor Results of Best single LED set-up

Note that the y axis is a log scale. Also, note that the line of best fit is a power equation with an exponent of -1.922, which falls closely in line with the law of inverse squares. This lets us know that the beam is not collimated. However, this beam is much better than an LED with no lens, corresponded by the grey set of dots. The line of best fit for the grey set of dots has a similar exponent of -1.941, but the coefficient is much lower, and the fall off is much steeper. In fact, the light of the LED falls off so much by 10 meters it is similar in brightness to the darkness of night.

6.2 Prototype Evolution Through Long Distance Communication Testing

A large milestone early in the project came in the form of getting communication working with no error correction at night over 80m. This system used the older, slower photo transistor circuitry in the receiver and had no filters. The transmitter was also an early version of the single LED system before the mathematical model was developed. The transmitter repeatedly sent out the phrase "Can u read this?" at 1 second intervals. The receiver incorporated a display that showed what the receiver received to make troubleshooting easier. Figure 37 shows a schematic of this set up. Figures 38 and 39 show the results.

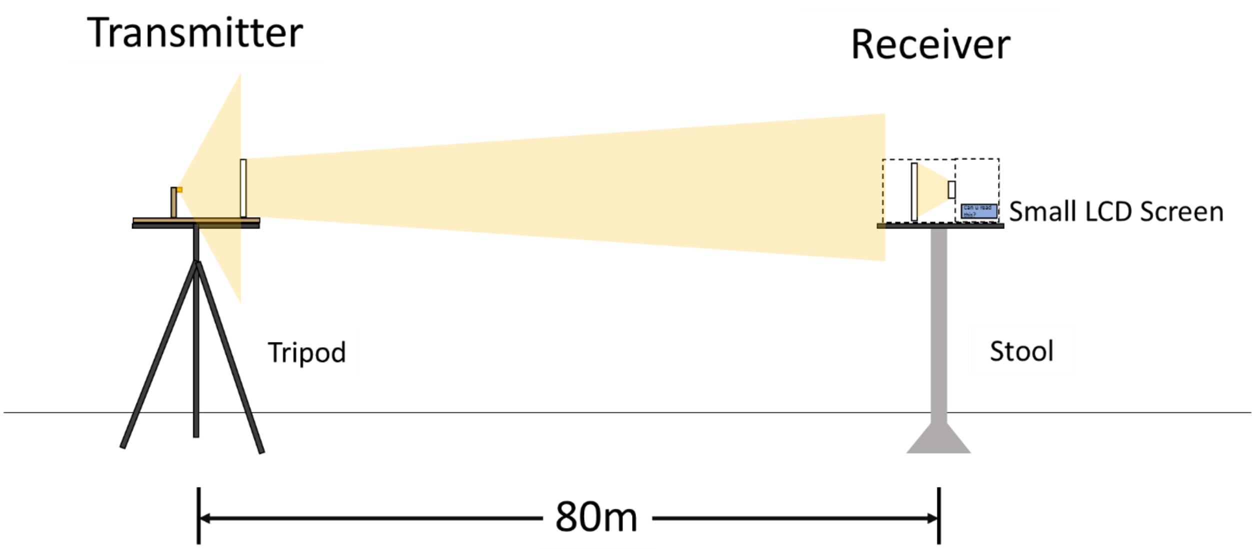


Figure 37: Outdoor Night 80m Communication Set-up

Figure 38: Outdoor Night 80m Communication Set-up Photos

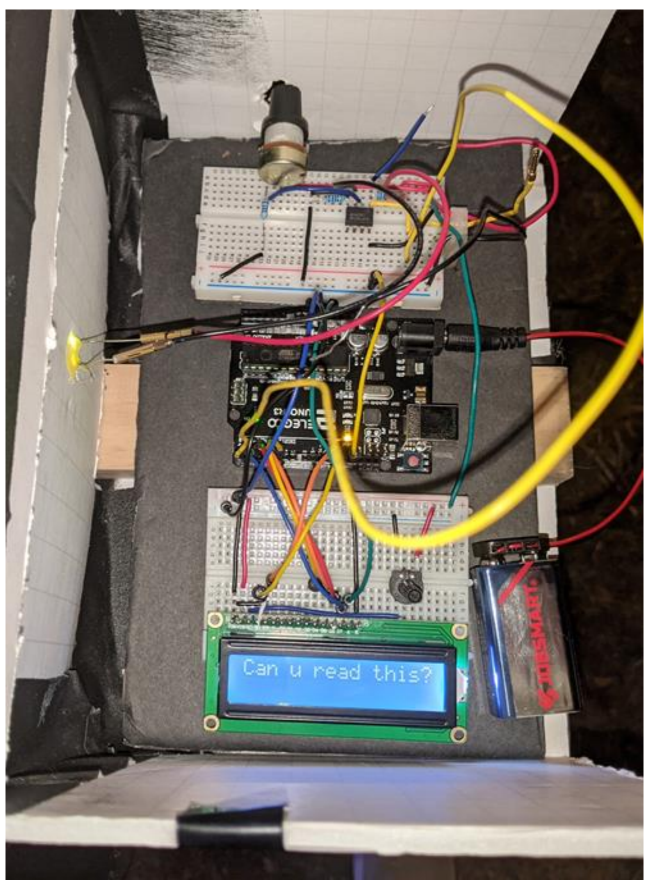


Figure 39: Outdoor Night 80m Communication Results

In figure 39 the small LCD screen can be seen showing the correctly received transmission, “Can u read this?” It can also be seen that this system included an Arduino Uno, no electronic filters and a photo transistor circuit. All of these items have since been changed, as described previously, for performance and efficiency.

Another important milestone was the full 100m test in morning daylight. This system used a different receiver box which was larger to capture more light. Figure 40 shows the test.

Figure 40: 100m Outdoor Communication Test

Notes about this particular test include the following: The ambient lighting is 20 kLux, which is far off from the full 100 kLux of bright summer daylight. The text on the small OLED screen is difficult to capture on camera, but the screen reads “Can u read this?”. Furthermore, although the receiver box is larger to capture more VLC light, it also captures more ambient light. As previously mentioned, however, this is not a problem as long as the ambient light rays do not enter the box colinearly with the VLC source light. For instance, streetlights may enter the box and pass through the receiver lens, but because they are not colinear with the system, they get projected to the side of the photodiode, as shown in figure 41.

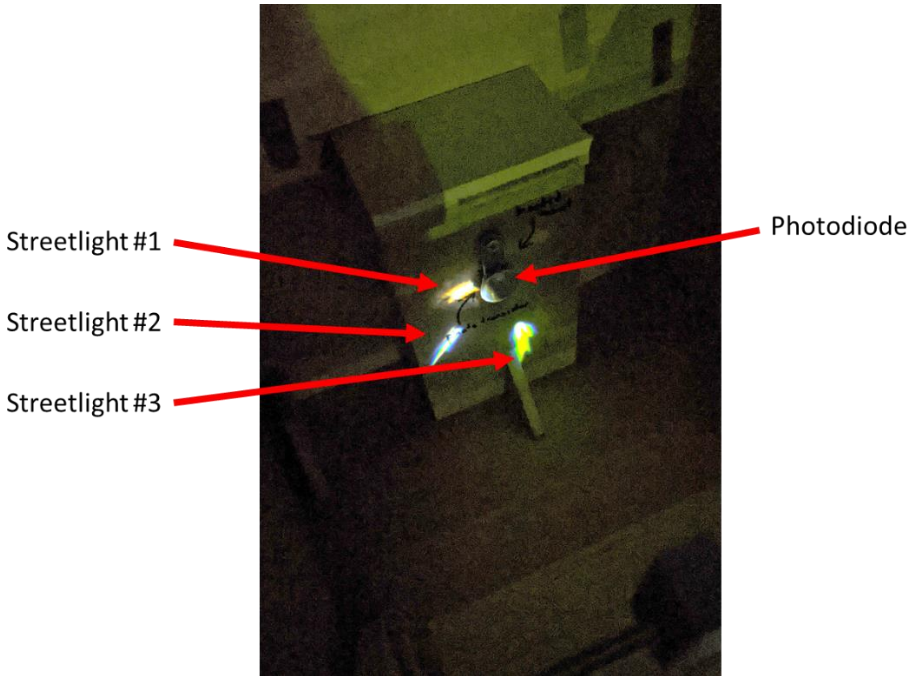


Figure 41: Outside Lights Entering the Reciver Box

Another change from this test to now is the receiver. To get the 100m goal, dual LEDs were tested along with dual lenses. This of course doubled the power draw when the LEDs were on. It also made the system larger to include double the hardware. This transmitter is shown in figure 42. Note that the picture was taken at night, a different time than the testing performed in figure 40.

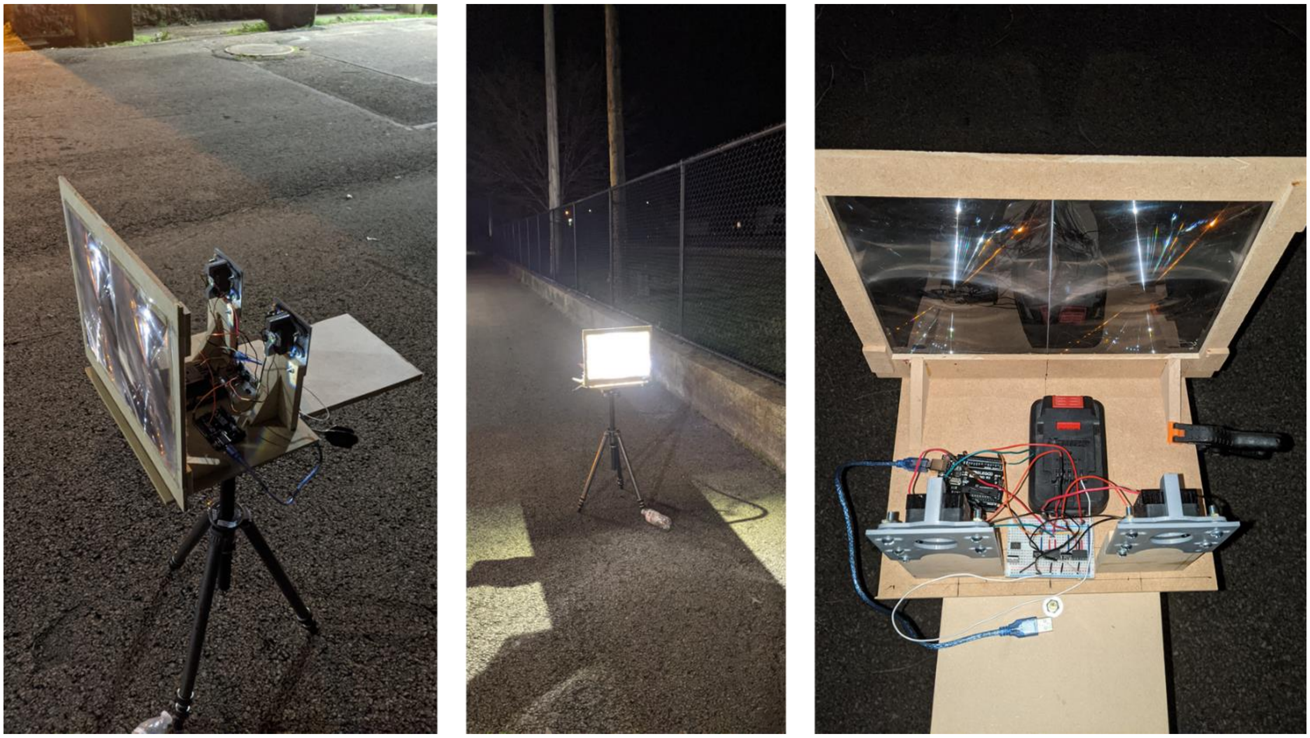


Figure 42: Dual LED Transmitter

6.3 100 Meter Communication in Full Daylight with Single LED

The final version of this system included a few changes, mainly size of transmitter and receiver. The transmitter moved back to a single Fresnel lens and LED. The electronics were revised to include a faster photodiode circuit and electronic filters to block any constant ambient light. The test shown in figure 43 shows this set-up in full 100 kLux summer daylight at 100 m, meeting the initial criteria.

Figure 43: Outdoor 100m Test in Fully Bright Daylight

Note the red circles offer zoomed in views of the transmitter/receiver 100 m away. Testing was performed in an alley with no shadows on either the transmitter or receiver. Interestingly, the receiver electronic filtering worked well enough that the box lid on the receiver could be open, allowing sunlight to enter the enclosure and still work properly. This is shown in figure 43.

Figure 44: 100m Outdoor Test in Bright Daylight with Open Lid on Receiver

Observing figure 44, we can see the lux meter is showing over 100kLux of brightness, the highest recorded during testing. Once again, the small OLED screen does not capture well on camera, but in this picture, it was showing the test phrase “Can u read this?” showing that the test executed successfully. Also note that the power supply is now two batteries in order to get a split voltage power supply.

6.4 Long Term Energy Consumption Testing

An important test in this study was to determine if the calculated power consumption was accurate and if the system could run indefinitely. The data showed that with only a few hours of sunlight, the solar system could generate more energy than was used by the electronics per day. A waterproof testing enclosure was made to store all of the transmitter electronics and solar panel inside, as shown in figure 45. The electronics transmitted VLC data with the Arduino and LED each second simulating the regular transmitting conditions.

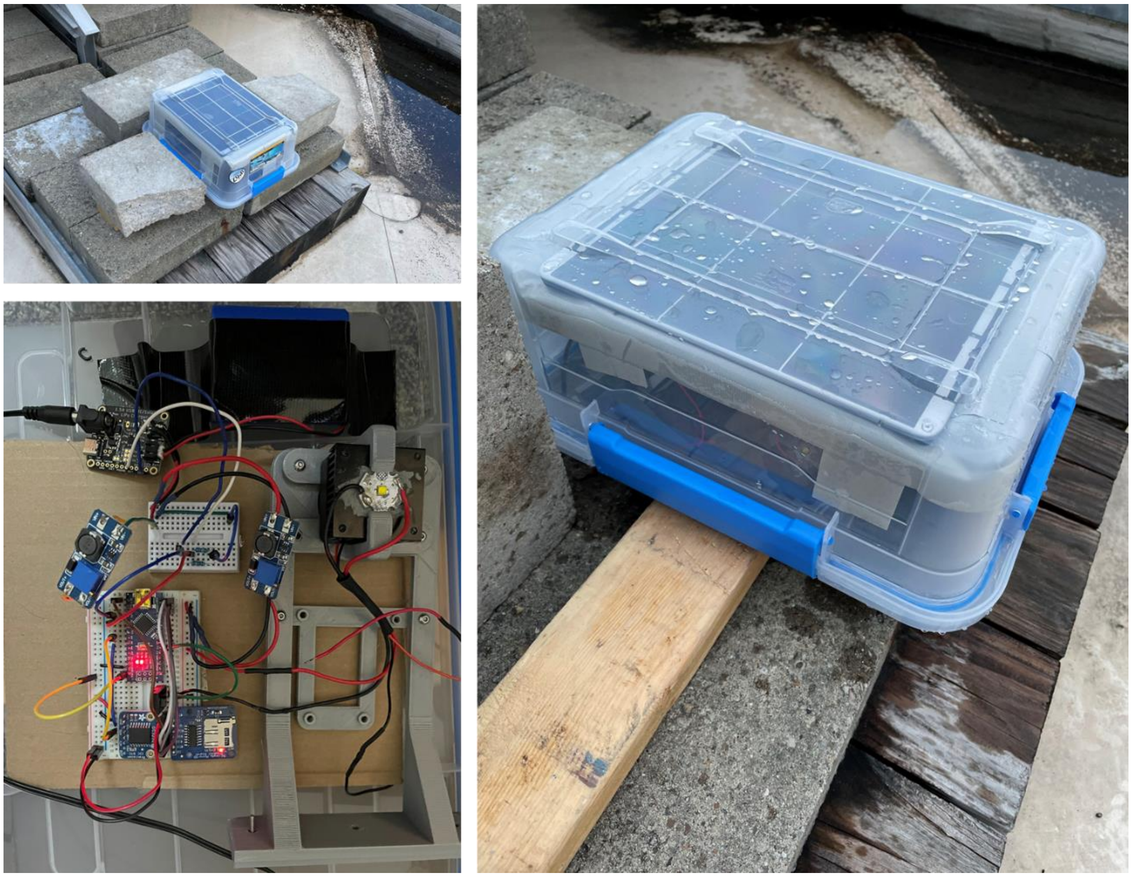


Figure 45: Waterproof Enclosure for Power Comsumption Testing of TX

Also shown in the bottom left of figure 45 is the internal components. The Solar panel is attached to the inside roof of the enclosure. The enclosure was placed on the roof of the engineering building at the University of Pittsburgh for multiple sets of 4 days or longer, with the longest being an 11-day period. The system gained temperature data, charging data, and battery voltage data. The battery when fully charged reaches a voltage of 4.5V. With this information, we can get a rough estimate of that battery's charge state by measuring the voltage of the battery and plotting it with time. Likewise, the solar charge board changes voltage depending on if the sun is powerful enough to charge the system. Figure 46 shows the temperature data results.

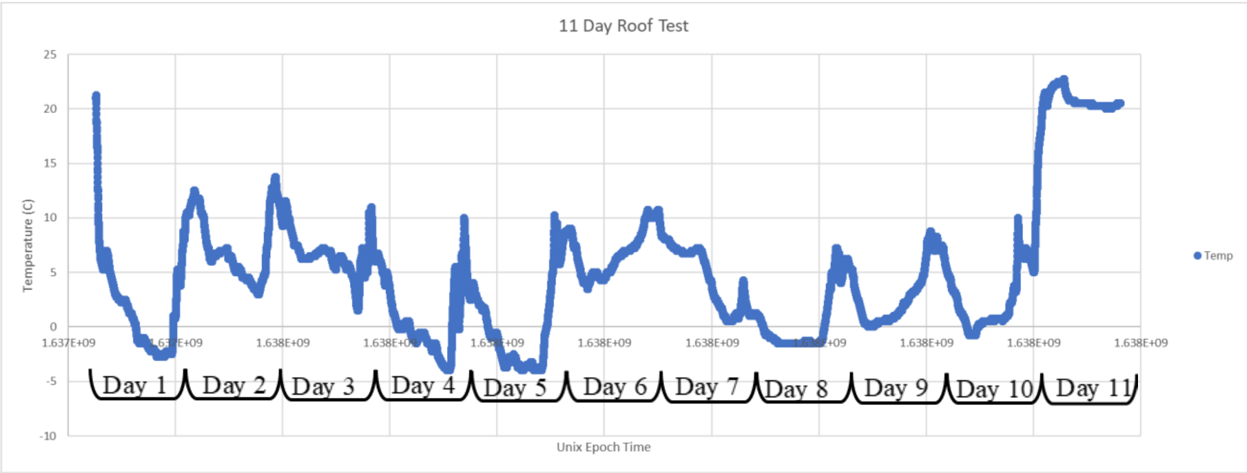


Figure 46: 11 Day Outdoor Temp Data

The x-axis is in Unix Epoch time, which is the amount of seconds from Jan 1, 1970 to now, and it can be directly converted to MM/DD/YYYY format. The 11 days are broken into segments for easier reading. There are 11 peaks for the hottest part of 11 days, but overall it was a relatively cold test, with temperatures dipping below freezing 5 times. The real time clock is able to keep time even in a power loss or shut down, but luckily there are no gaps in the blue line, meaning that the system was able to work in freezing conditions. Figure 47 shows the charging and battery voltage data.

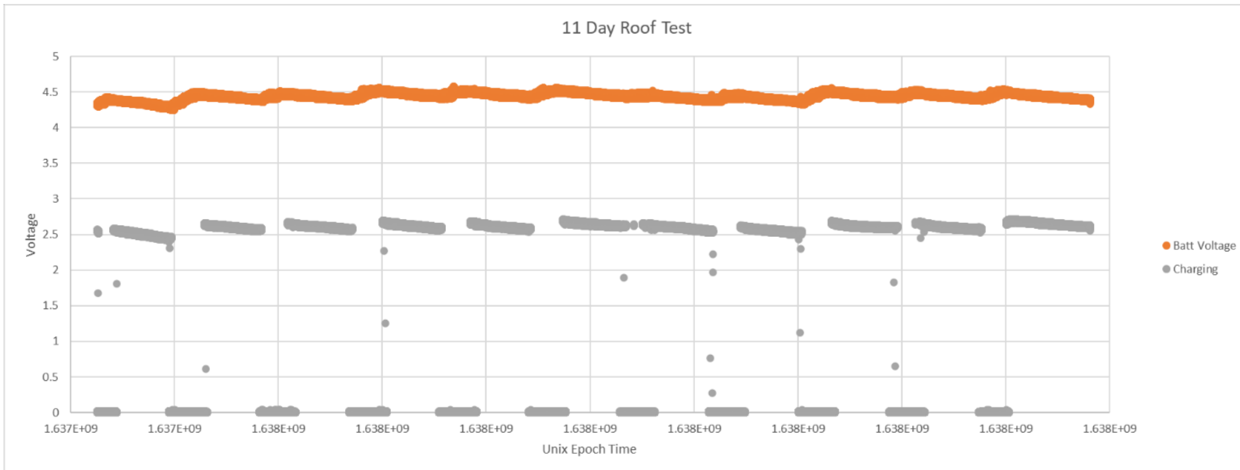


Figure 47: 11 Day Outdoor Charging and Battery Data

In figure 47, note the grey points which tell us if the sun is powerful enough to charge the system. The grey points go to 0V when it charges and 2.55V when it does not. There are 11 segments at 0V corresponding with the 11 daytimes that the system charged. Also note the orange dots which correspond with the battery voltage. Each time the grey dots go to 0V, the orange dots charge upwards, then when the grey line goes back up to 2.55V (system not charging) the orange dots trend downwards showing the system discharging. Now note the general trend of the orange dots from the start of the 11-day period to the end. They started slightly below 4.5V and had a general trend upwards over 11 days to about 4.5V. This shows that the battery was not fully charged when going into the test and leaving the test with more charge than it started with. Also note that some days had less charging than others, particularly days 7 and 10, meaning that the sky was cloudy.

This test confirmed that with only 5-6 hours of charging per day in the winter, the transmitter can gain enough energy to work without having to pull the system from a test to charge manually. The receiver has a lower power draw since it has the same micro controller, but no LED.

7.0 Field Test Design

The NETL version of this has been mentioned a few times throughout this thesis. The NETL system is a ruggedized and weatherized final prototype that can be used outdoors for a long period of time.

The enclosure is a NEMA rated enclosure measuring, with a transparent window for the light to pass through. The cutout is 250mm in diameter, smaller than the 300mm diameter of the lens. The transmitter and receiver enclosures are equal in design. Each enclosure has a removable bottom panel that the electronics and 3D-printed bracketry is mounted to. The enclosures have solar panels that are mounted above the lid, which also serves to shade the box from direct sunlight.

The system required a smaller aperture in the enclosure than the prototypes, so the performance is slightly decreased from the prototype performance. Both the transmitter and receiver have been mounted in place at the outdoor test site at NETL, and testing is underway.

8.0 Conclusion

The contributions of this research include:

1. A prototype, self-powered, VLC system capable of communicating at 100 m in various weather and lighting conditions.
2. A methodology for performing the optical design of the VLC system
3. Methods of reducing the effects of spurious light on VLC communication
4. Low-power transmitter circuitry with solar charging
5. Methods of simple error checking for received packets

As stated in the introduction, the purpose of this research was to explore the viability of visible light communications in an outdoor powerplant as opposed to other wired and wireless communication methods. This was achieved by testing the performance of various optical set-ups in conjunction with various control methods. The performance criteria considered were distance of communication, speed of communication, power consumption, robustness in various weather conditions and overall size.

Multiple prototypes were developed; however, the best results for long distance and low power consumption came in the form of a small LED with a large lens. This follows the theory in equation 3, giving a low angle of divergence. This combination also made the physical size rather large, so a proper balance was found between overall size, and distance of transmission by using a mathematical model developed through theory and testing. In the end, a Cree XHP35.2 LED was used for its high power output and small emitter diameter in conjunction with a 300mm diameter, 360mm focal length Fresnel lens. A lux value of 110 lux was achieved at 100m with this set up.

In terms of control, multiple microcontrollers were explored such as Beaglebone Black, Arduino Uno, Arduino Due, but the Arduino Nano was settled upon for its easy-to-use interface as well as low power draw. Custom circuitry was developed for both the receiver and transmitter involving an active transimpedance amplifier which is both faster and more controllable than the photo-transistor circuit used early on. The circuitry also includes a passive bandpass filter to rid high frequency noise as well as any DC gains. The receiver circuitry can work with as little as 40 lux of VLC signal light, giving headroom since the prototype produces 110 lux at 100m.

Power draw is kept to a minimum by using the Arduino nano as well as by inverting the VLC signal so that the LED is mostly off with short on pulses. This was achieved by using the Software Serial library with the Arduino which lets the developer invert the output signal from the GPIO pin on the Arduino. A battery was sized appropriately to allow the system to last 1 week with no recharge, and solar panels were incorporated to increase that battery life indefinitely. Battery longevity tests included putting the transmitter electronics with solar panel in a weather-proof enclosure and leaving on the roof of a building for 11 days. Results showed that the system had a net gain of charge state after the 11-day test.

Results of the full communication at 100m showed that the system worked well even in broad daylight as shown in figure 43 and 44. The system failed only when a vehicle or person drove in between the transmitter and receiver in the test alley way.

For error checking, two simple methods were developed, a known character check, and a calculated length check. A known character is placed at the end of every data transmission, as shown in figure 21. If this character is not received correctly, it is likely that the entire transmission was received incorrectly. However, it is possible that the transmission had an error part way through the data packet but was able to send the known character at the end correctly. In this case, the calculated length error check becomes beneficial. The receiver adds up the ascii decimal value of each character received and compares it to the received length as part of the packet. If the two numbers match, then the transmission is assumed to be correct and recorded to an SD card.

Finally, two systems were developed: one prototype for the University of Pittsburgh, as well as a prototype for NETL. The NETL prototype is much more robust and will be used outdoors for a period of testing. It differs from the main prototype as it is within a full metal enclosure with a glass aperture. The glass aperture is smaller than the lens, so the overall performance is decreased. The mathematical model predicts that the system should still achieve around 60 lux of brightness at 100m. Lastly, there is a metal tube on the outside of each enclosure. This tube has a fixed radius, and thus the natural divergence of the beam will cause some of the light to be absorbed by the interior of the tube. This is not accounted for in the model, and thus testing will be performed to see how it affects performance.

This research succeeded in developing a great understanding of optical physics in conjunction with digital signal communications. Visible light communication exhibits multiple advantages over existing wireless communications like Wi-Fi, but it also has downsides, speed and distance being the two main ones. If a use case of VLC has criteria designed around VLC's shortcomings, then visible light communication can be a viable option for wireless data transfers.

Appendix A Codes

The following appendix shows some of the code that was used for testing.

Transmitter Code:

```
#include <SoftwareSerial.h>
#include "RTClib.h"
#include <SPI.h>
#include <Wire.h>
#include <SD.h>
File myFile;
RTC_DS3231 rtc;
char daysOfTheWeek[7][12] = {"Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"};
SoftwareSerial mySerial(2, 3, true); // RX, TX, inverse_logic = true
long tim = 1;
int sensorPin = 0;
double Battval;
double Solval;
String Message;
int calculatedlength;

void setup() {
  while (!Serial) {
    ; // wait for serial port to connect. Needed for Native USB only
  }
  pinMode(3, OUTPUT);
  digitalWrite(3, HIGH);
  delay(6000);
  digitalWrite(3, LOW);
  mySerial.begin(57600);
  Serial.begin(57600);

  #ifndef ESP8266
  while (!Serial); // wait for serial port to connect. Needed for native USB
  #endif

  //troubleshooting with SD card on TX
  /*/can block commment this out if no SD cad on TX
  Serial.print("Initializing SD card...");
  if (!SD.begin(10)) {
    Serial.println("initialization failed!");
    while (1);
  }
  Serial.println("initialization done.");

  //*/
```

```
  if (! rtc.begin()) { // setting up Real time clock
    Serial.println("Couldn't find RTC");
    Serial.flush();
    abort();
  }

  if (rtc.lostPower()) {
    Serial.println("RTC lost power, let's set the time!");
    // When time needs to be set on a new device, or after a power loss, the
    // following line sets the RTC to the date & time this sketch was compiled
    //rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
    rtc.adjust(DateTime(2021, 10, 9, 23, 5, 0));
    // This line sets the RTC with an explicit date & time, for example to set
    // January 21, 2014 at 3am you would call:
    // rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0));
  }

  // When time needs to be re-set on a previously configured device, the
  // following line sets the RTC to the date & time this sketch was compiled
  // rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  // This line sets the RTC with an explicit date & time, for example to set
  // January 21, 2014 at 3am you would call:
  // rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0));
}



void loop() {
    calculatedlength = 0;

    DateTime now = rtc.now();
    Battval = (analogRead(A0)/1024.0)*5.0;
    Solval = (analogRead(A1)/1024.0)*5.0;
    Message = (String)now.year() + "/" + (String)now.month()+ "/" + (String)now.day() + ";" +
    (String)now.hour() + ":" + (String)now.minute() + ":" + (String)now.second() + ";" +
    (String)rtc.getTemperature() + ";" + (String)Battval + ";" + (String)Solval + ";";

    for(int i = 0; i <= Message.length(); i++){
      calculatedlength = calculatedlength + (int)Message[i];
    }
    Message = Message + (String)calculatedlength + ";$";
```

```
//This chunk writes everything to the serial terminal over USB for trouble shooting
/*
Serial.print(now.year(), DEC);
Serial.print('/');
Serial.print(now.month(), DEC);
Serial.print('/');
Serial.print(now.day(), DEC);
Serial.print(';');
Serial.print(now.hour(), DEC);
Serial.print(':');
Serial.print(now.minute(), DEC);
Serial.print(':');
Serial.print(now.second(), DEC);
Serial.print(';');
Serial.print(rtc.getTemperature());
Serial.print(';');
Serial.print(Battval);
Serial.print(';');
Serial.print(Solval);
Serial.println();
//*/
Serial.println(Message);

//this chunk writes everything to the VLC LED
/*
mySerial.print(now.year(), DEC);
mySerial.print('/');
mySerial.print(now.month(), DEC);
mySerial.print('/');
mySerial.print(now.day(), DEC);
mySerial.print(';');
mySerial.print(now.hour(), DEC);
mySerial.print(':');
mySerial.print(now.minute(), DEC);
mySerial.print(':');
mySerial.print(now.second(), DEC);
mySerial.print(';');
mySerial.print(rtc.getTemperature());
mySerial.print(';');
mySerial.print(Battval);
```

```
  mySerial.print(';');
  mySerial.print(Solval);
  mySerial.println();
  */
  mySerial.print(Message);


  //If have an SD card on TX side
  ///////////////////////////////////////////////////////*
  ///*
  myFile = SD.open("TestTwo.txt", FILE_WRITE);
  if (myFile) {
    myFile.print(now.year(), DEC);
    myFile.print('/');
    myFile.print(now.month(), DEC);
    myFile.print('/');
    myFile.print(now.day(), DEC);
    myFile.print(';');
    myFile.print(now.hour(), DEC);
    myFile.print(':');
    myFile.print(now.minute(), DEC);
    myFile.print(':');
    myFile.print(now.second(), DEC);
    myFile.print(';');
    myFile.print(rtc.getTemperature());
    myFile.print(';');
    myFile.print(Battval);
    myFile.print(';');
    myFile.print(Solval);
    myFile.println(' ');
    myFile.close();
  }
  else {
    // if the file didn't open, print an error:
    Serial.println("error opening");
  }\
  //*/
  /////////////////////////////////////////////////////////*/

  delay(1000);
}
```

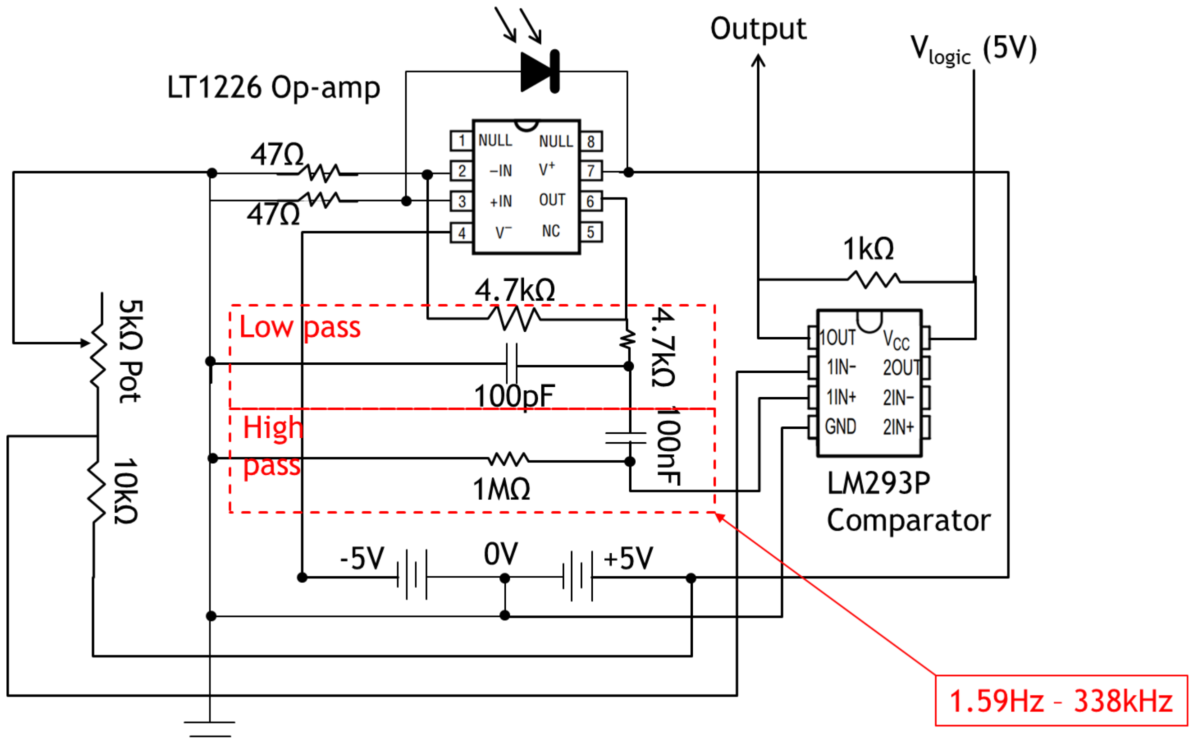
Receiver Code:

```
#include <SoftwareSerial.h>
#include <string.h>
#include <SD.h>
File myFile;
SoftwareSerial mySerial(2, 3, true); // RX, TX

String Msg, datedata, timedata, tempdata, batterydata, solardata, lengthdata;
//unsigned int  Msg.length();
void setup()
{
  Serial.begin(57600);// set the data rate for the USB serial
  while (!Serial) {
    ; // wait for serial port to connect. Needed for Native USB only
  }
  mySerial.begin(57600);// set the data rate for the SoftwareSerial port
  ////////////////////////////////////////////////////////////////////////////////////////////// SD Card
  /*
  Serial.print("Initializing SD card...");
  if (!SD.begin(10)) {
    Serial.println("initialization failed!");
    //while (1);
  }
  Serial.println("initialization done.");
  */
  //////////////////////////////////////////////////////////////////////////////////////////////
}



void loop() // run over and over
{
  int callength = 0;
  int i = 0;
  /////////////////////////////////////////////////////////////////Receive incoming characters and put them in a string
  if (mySerial.available()){
    char inchar = mySerial.read();
    Msg+=inchar;
    //Serial.println(Msg);
  }

  else{
    if (Msg.charAt(Msg.length()-1)=='$'){//check for final char and end string
      Serial.println(Msg);//print to console
```

```
//////////////////////////////////////////////////////////////Sparsing of Data
      while(Msg[i] != ';'){
        datedata = datedata + Msg[i];
        callength = callength + (int)Msg[i];
        i++;
      }
      i++;
      while(Msg[i] != ';'){
        timedata = timedata + Msg[i];
        callength = callength + (int)Msg[i];
        i++;
      }
      i++;
      while(Msg[i] != ';'){
        tempdata = tempdata + Msg[i];
        callength = callength + (int)Msg[i];
        i++;
      }
      i++;
      while(Msg[i] != ';'){
        batterydata = batterydata + Msg[i];
        callength = callength + (int)Msg[i];
        i++;
      }
      i++;
      while(Msg[i] != ';'){
        solardata = solardata + Msg[i];
        callength = callength + (int)Msg[i];
        i++;
      }
      i++;
      while(Msg[i] != ';'){
        lengthdata = lengthdata + Msg[i];
        i++;
      }
      callength +=295;
      Serial.println(datedata);
      Serial.println(timedata);
      Serial.println(tempdata);
      Serial.println(batterydata);
      Serial.println(solardata);
      Serial.println(lengthdata);
      Serial.println(callength);
      //int lengthdata=1;
```

```
      if (lengthdata.toInt()==callength){
        Serial.println("Passed error check");
        //can print to SD card now
        ///*
        myFile = SD.open("VLC_Test.txt", FILE_WRITE);
        if (myFile) {
          myFile.println(Msg);
          myFile.close();
        }
        //*/
      }
      else{
        Serial.println("Failed error check");
      }
      Msg="";
      datedata = "";
      timedata = "";
      tempdata = "";
      batterydata = "";
      solardata = "";
      lengthdata = "";
    }
  }
}
```

Receiver Circuitry:



Bibliography

1) Diaz, Edwin, and Matthias Knobl. "Prototyping Illumination Systems with Stock Optical Components." *Edmund Optics*, 2012, https://www.edmundoptics.com/globalassets/knowledge-center/articles/prototyping-illumination-systems-with-stock-optical-components-en.pdf.

2) *What is ASCII (American standard code for information interexchange)?* What is ASCII (American Standard Code for Information Interexchange)? (2021, May 2). Retrieved March 14, 2022, from https://www.computerhope.com/jargon/a/ascii.htm

3) SM. (2019). *SoftwareSerial Library*. Arduino. Retrieved 2021, from https://www.arduino.cc/en/Reference/softwareSerial

4) Chi, N. (2018). *LED-Based Visible Light Communications*. Retrieved 2020.

5) Boucouvalas, A. C., Chatzimisios, P., Ghassemlooy, Z., Uysal, M., & Yiannopoulos, K. (2015). Standards for Indoor Optical Wireless Communications. *IEEE Communications Magazine*.

6) Galisteo, A., Juara, D., & Giustiniano, D. (2019, May). *Research in Visible Light Communication Systems with OpenVLC1.3*.

7) Tu, Y. (2018). *Ray Optics Simulation*. Ray Optics Simulation. Retrieved 2021, from https://ricktu288.github.io/ray-optics/

8) Jovicic, A., Li, J., & Richardson, T. (2013). *Visible Light Communication: Opportunities, Challenges and the Path to Market*.

9) *SoftwareSerial Library*. (2021). Arduino. Retrieved 2021, from https://www.arduino.cc/en/Reference/softwareSerial