Adapting Visible Light Communication Technology for Inter-Vehicular Use

by

Russell C. Andres

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science
(Computer Engineering)
in the University of Michigan-Dearborn
2021

Master's Thesis Committee:

      Associate Professor Sridhar Lakshmanan, Chair
      Associate Professor Paul Watta
      LEO Lecturer IV Michael Putty

Russell C. Andres

rcandres@umich.edu

ORCID iD: 0000-0002-8791-5080

Acknowledgements

I would like to thank Dr. Sridhar Lakshmanan for all the opportunities he has offered me, from selecting me to be his graduate research assistant, including me on this project, to helping me find employment as I completed this thesis. I'd also like to thank Dr. Michael Putty for lending me his time and electronic engineering expertise with the analog components, especially in the gain adjustment system, when I ran into problems with the prototype. Dr. Paul Watta has my admiration for his teaching abilities and my gratitude for his unstinting support. Finally, I would like to thank Cristian Adam for his patience, assistance in testing iterations of the prototype, and his guidance through the university's processes especially during the pandemic.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ADC - Analog-to-Digital Converter
ASIC - Application-Specific Integrated Circuit
CV2X - Cellular Vehicle-to-Everything
DSRC - Dedicated Short-Range Communications
FPGA - Field-Programmable Gate Array
IAVS - Institute for Advanced Vehicle Systems
LED – Light-Emitting Diode
PLL - Phase-Locked Loop
UART - Universal Asynchronous Receiver-Transmitter
VGA - Variable-Gain Amplifier
VLC - Visible Light Communication

Abstract

With the advent of connected and autonomous vehicles rapidly approaching, automakers are finding that existing technologies for radio-frequency wireless inter-vehicle communication are not keeping up with demand. A potential solution lies with Visible Light Communication (VLC), sometimes also called LiFi, a technology that can send data wirelessly through visible or near-visible light rather than through radio waves. This thesis proposes a prototype that adapts VLC technology for inter-vehicular use. It will determine the feasibility of obtaining usable bandwidth while maintaining a reasonable range via the use of ambient light filtering and adaptive gain controls in the analog side of the receiver. Research was also done into methods of encoding the data for improved signal-to-noise ratios and reduced error rates, both of which were physically measured on a real system. Promising results were obtained, and a short list of future adaptations are discussed that should further increase the performance of the design. Advantages of the technology will include higher bandwidth and a line-of-sight limit to signal propagation, meaning that multiple network cells can be emplaced in a small area without interference and improving performance in high-traffic situations.

Chapter 1 Introduction

1.1 Background

Visible Light Communication (VLC) uses visible light from ordinary light sources, usually LEDs, to transmit data in place of radio waves [1], [2].
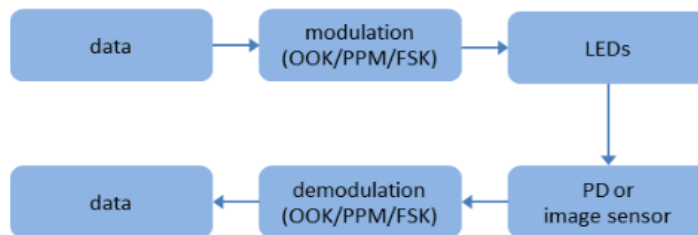
Figure 1.1 How data is transmitted using VLC [3]

It has the advantages of being very high bandwidth, short range, fairly secure, and not being regulated [4]. These factors are related. Light is naturally higher frequency, which means that transmission systems can be switched faster to send more data. Because the frequency is high, it is also blocked more easily by objects, which prevents the signal from travelling too far (which could cause signal interference and speed decreases) or leaving sealed rooms (which can allow eavesdropping) [1]. The lack of regulation is likely based on both the relatively new status of the technology and on the fact that light is both sufficiently common and has such a large bandwidth that it would be difficult to start regulating it now.

1.2 Problem Statement

All of these aspects of VLC make it apparent that it is a valuable future technology for communication. There is one specific use of this technology that shows particular value, and that is in inter-vehicular communications.

Up until quite recently, cars primarily were controlled by their drivers and signaling between the cars was in the form of simple, two-state visual or audible signals, such as brake lights, turn signals, or the horn. These were sufficiently comprehensible to another driver, but in the current age, automation and driver assistance systems have proliferated, and these systems are rather poor at reading the nuances of such uncomplicated signals, if they detect them at all [5]. (Even the most modern algorithms have trouble with image processing, and it requires far more resources than any other task the car's computers might be doing [6], [7].) As a result, cars now need to have signals that transmit more information in a form better understood by the vehicle's computers. Currently, this is provided via radio-frequency wireless systems such as Dedicated Short-Range Communications (DSRC) or Cellular Vehicle-to-Everything (CV2X, where the X could be infrastructure or other vehicles).

Unfortunately, radio systems indiscriminately send a radio signal in all directions, which interferes with nearby radios that may also be trying to send at the same time [8]. To safely transmit, the radio must either wait until no other radios are sending (which cannot be easily known, since the very medium it communicates over is also the resource it needs to use) or attempt to re-send its message if it gets corrupted by another radio transmitting at the same time. This results in a latency issue, and while that is permissible when browsing the Internet or when using a mobile phone, a delay of a safety-critical message between connected vehicles can contribute to accidents [9].

As mentioned, however, light is directional, has a limit on how far it can travel, and is significantly impeded when traveling past an intervening vehicle. This means that messages can be sent using VLC that will by nature stay within the immediate neighborhood of nearby vehicles, which is also the main area where these messages are relevant, and it cannot travel further to interfere with more than a few other vehicles' communications. The higher bandwidth of optical systems also means that more data can be sent, or that the same amount of data can be sent with less latency; both of which are potentially helpful when designing future intelligent vehicles.

1.3 Design Overview

It is this thesis' goal to produce a prototype VLC system that might be integrated into LED vehicle headlight or tail-light systems. The light would act as a transmitter to send encoded data to receivers located in or near the lights of other cars. Since the light is blocked if it travels too far or encounters another car, it is thus much less likely for interference to occur like it can in the case of a radio network. This is ideal, since only the vehicles immediately near the transmitting vehicle really need to know its headway, acceleration, or other information. Any other information can be provided just as well by the intervening cars, such as time to slow or stop when approaching a traffic light. Alternatively, both DSRC and VLC can be used together. DSRC can be used for less-frequent or broadcast-style announcements (or in non-line-of-sight conditions), while VLC simultaneously allows for local, low-latency communications. Alternatively, they can be used for complementary sides of the same link [10], [11].

1.4 Review of Existing Strategies and Related Works

VLC is currently a very popular technology. A large number of people are conducting research in this field with varying results. Some companies, notably PureLiFi, have had great success with the technology in an indoor setting. Fewer focus on the outdoor or vehicular environment because it has proven to be more challenging.

As a demonstration of the difficulty in building a high-performing VLC system, several groups have designed VLC data transmitters that are limited to ranges of at most three meters or a data rate measured in tens of bytes per second even in an indoor environment. S. Sivaguru, B. Sundari, and N. Rupavathi have designed an RS-232-based system that achieves 115200 baud, but that is limited to an operational range of 2.3 meters at night (1.76 meters during the day) and a 3% error rate due to the use of an unamplified, direct-to-digital comparator without ambient light compensation fed by a photodetector and voltage divider. Despite the unusual design choice, the receiver appears as if it could be configured to work in any light level, although it would require manual adjustment the moment that the ambient light level changed [12].

Tahmid H. Talukdar and Shahadat H. Parves developed a system that transmits data using an atypical variant of PWM where a short logic low followed by a long logic high represented a 1, and a long logic low followed by a short logic high represented a 0. The transmitter was an entire office light panel rated for 18 watts at 20 volts. For the sensor, it used a BPW34 photodetector and a voltage divider apparatus that was manually tuned to produce a voltage that would trigger a digital pin if the transmitting light was on but to provide zero volts with the light off. This would be highly dependent on the ambient light in the room and has a limited overall range of operation. The eventual data rate was revealed to be only 78 bytes per second, or 624 bits per second, and the range of the system was not given [13].

A final example comes in the form of a system produced by researchers at the University of Engineering and Management in Kolkata in West Bengal, India. Their encoding was a unique pulse-count design that used three light pulses to represent a 1, and two pulses to represent a 0. This protocol deserves further research for its self-synchronizing properties and because it can be designed to ignore other frequencies. Their receiver uses the same BPW34 detector and voltage divider design as Talukdar and Parves, but rather than a digital conversion, they used the analog-to-digital converter of an Arduino microprocessor to detect the change in light level. The data rate was 50 bits per second over a distance of one foot after manual adjustment to the detector to calibrate it to the room's ambient light [14].

In contrast, Harald Haas is a pioneer in the field of optical data transmission. He coined the term LiFi in 2011 and went on to found the company PureLiFi, focusing on the idea of a fully integrated visible-light networking stack [15]. While his research is predominantly on indoor VLC, it has been so influential that it requires a mention. By 2015, he and his team had developed transmitter and receiver Application-Specific Integrated Circuits (ASICs) for high-speed optical data transfer. The benefit of designing the components using an ASIC is the ability to integrate all of the required components like detectors or digital-to-analog converters right alongside custom-designed hardware that can rapidly process the data going to or from the other components.

Haas's transmitter is capable of 25 million samples per second transmission of analog light signals across a total of eight channels for color-shift keying, multi-user support, or dimming. It is capable of 1GB/s transmit rates when using all channels at once. His receiver, meanwhile, contains 49 acutely sensitive avalanche photodiodes in a grid formation with some of them featuring color or directional tuning for use with more complex protocols. He also

championed the idea to use both RF and optical data transmission together for improved overall performance [1]. Since then, a 3 Gbps data rate was demonstrated with a micro-LED, a type of LED designed to have a faster switching time than the normal variety [15].

In the field of vehicular VLC communication, researchers from Yonsei University implemented a vehicular VLC system that transmitted a signal taken directly from the vehicle's CAN bus over a distance of 20 meters at 500 thousand baud with a receiver design similar to the one discussed in this thesis. Unfortunately, the data sent over their network was not actually decoded and used, merely read by an oscilloscope. It would have been interesting to see their solution. Their receiver, meanwhile, had problems with the presence of sunlight and required an optical filter and a "light shield" to function properly outdoors. From their article, it appears that this may be due to the use of a second "limiting" amplifier that was mainly used to reduce the gain of the detector amplifier, likely resulting in the first saturating quite easily due to its high gain setting [16]. While this thesis' VLC prototype uses a similar receiver circuit, it was created with no prior knowledge of Yonsei University's design. Unfortunately, it did not obtain as high of a data rate as the system used at Yonsei University, mainly due to the decision to encode and decode the data rather than to carry an external signal without processing it. However, the differences in the VLC prototype's circuit resulted in a system with much higher resistance to ambient light compared to the design from Yonsei University, mainly due to this thesis's choice to use a lower gain in the first amplifier and the use of the second amplifier as the primary gain source.

Similarly, researchers at Telkom University designed a car-to-car VLC network that used UART to send data at 38400 baud between vehicles at up to 11 meters at night. In the day, the range was limited to 2 meters. The data being sent was extracted from a vehicle's CAN bus and

included brake, gas, and speedometer readings. Solar cells were used to detect the light signal [17].

Hossien B. Eldeeb, Evsen Yanmaz, and Murat Uysal (in Turkey) have simulated a VLC network that uses a digital modulation scheme based on IEEE 802.15.7 that achieves a 30-meter range (21 meters in fog) and uses a carrier sense multiple access/collision avoidance scheme to prevent collisions of transmitted messages. It additionally uses multihop retransmission of packets to further increase the range of the data being sent. The data rate is unlisted [18].

Cen Liu, Bahareh Sadeghi, and Edward W. Knightly have designed a system with a 100 kilobit-per-second data rate and apparent range of 101 meters that is highly resistant to passive interference of different frequencies or saturation from ambient light sources including sunlight but is highly sensitive to any signals due to its use of Manchester encoding. An important part of this ability is from a bandpass filter that is used to reduce the sensitivity of the circuit to any frequencies other than that of the designated transmitter [19]. Their highly interesting design outperforms that of this thesis, but the use of a Code Division algorithm makes this VLC prototype more resilient against noise in general while its hardware leaves it nearly as immune to ambient light as Liu, Sadeghi, and Knightly's system.

Chapter 2 Challenges of an Optical System

Even though radio frequencies and light are both on the electromagnetic spectrum, they have very different behaviors. Information can be encoded directly into a radio signal in multiple ways (changing the power, changing the frequency, or changing the phase) but even with LEDs or lasers, light can only be adjusted in frequency or amplitude, and creating or detecting frequency changes requires different hardware for each frequency. Transmitting data with light thus requires vastly different hardware and new encoding techniques compared to radio. A common approach is known as "intensity modulation," the transmission of data by turning the light source on and off, and many advanced protocols are still rooted in this technique [1], [20].

Additionally, most radio bands do not have natural sources and diffract around solid objects, while light is commonly produced by numerous competing, non-data sources and is easily blocked. As a result, some facets of transmitting with VLC have a number of complexities that do not appear in radio-frequency data transmission: interference, clock recovery, ambient light, and signal strength changes. These need to be resolved before VLC can be used as a viable means of communication.

2.1 Interference

Interference is merely the first of the problems that complicate VLC data transmission outdoors. Both natural and artificial light sources can interfere with data transmissions, as can some weather conditions. Natural interference sources can include sunlight, reflected sunlight (including from puddles), and lightning. Artificial sources include lights on other vehicles (both

traditional and VLC), light sources on pedestrians or bicyclists, street lighting, and home or business lighting. Because this thesis focuses on the feasibility of VLC being used to communicate between vehicles, light changes from the car's motion are also a factor to be considered. In addition to actual light sources interfering, the signal itself can be scattered or attenuated by airborne substances like dust, snow, or fog, although rain has only a minimal effect [18], [21].

Attenuation from fog or dust is a large issue, since the signal from the transmitter is already difficult to detect. This can be mitigated by increasing the sensitivity of the receiver, but that makes it more susceptible to interference from other light sources and may reduce the maximum speed of the channel. This is, however, a problem that radio signals are mostly unbothered by and which can be mitigated by the use of a combined VLC/DSRC communication suite [1], [10], [20].

Artificial light sources will likely be the largest problem faced by a workable VLC solution. Many modern electronics inadvertently emit high frequency modulated light because of brightness control or from a previously harmless side effect of their power supply's operation, but to VLC this can be a source of strong interference. This was even seen during early testing of the VLC prototype's brightness response when an LED flashlight did not properly filter its output and caused the prototype to pick up a signal that was not supposed to be there.

Competing VLC sources are likely to cause at least some degree of interference as well. If they use the same protocol, the limited number of cars that can receive (or interfere with) any given signal will be handled by the protocol itself. If multiple standards are in use and the receivers or transmitters are not capable of detecting the signals of other standards, it is possible

for adjacent vehicles to talk over each other and reduce the signal quality of the other. This is made worse if one of the protocols is sensitive to noise [19].

Finally, most natural light sources are unlikely to cause interference problems because they are relatively constant, and any changes are extremely infrequent. Sunlight with intermittent cloud cover or reflections caused by sunlight, however, can temporarily overwhelm the receiver, causing the loss of reception for a fraction of a second. Lightning can also overwhelm the receiver, especially in otherwise dark conditions. In most cases, reception will resume in short order, but the danger lies in the chance that extremely fast and urgent transmissions may potentially be missed during these events. This is a safety issue that can be solved by ensuring that important messages are sent with enough time for the receiving vehicle to react, requiring a form of acknowledgement to the message or it will be re-sent, transmitting them multiple times to begin with, or by filtering out certain frequencies of light from the receiver. The last is infeasible since that would greatly reduce the available bandwidth and would also prevent the dual use of vehicle headlamps for both data transmission and illumination. The others cause some degree of overheard but allow the use of normal channels.

2.2 Ambient Lighting

Ambient lighting is actually a problem for VLC systems, at least if there is enough of it. This is for three reasons:

1. It reduces the sensitivity of detectors slightly. A detector in a dark environment can trust that the photons it detects are solely from the transmitter, but as the ambient light increases, more and more of the photons are from the ambient light instead. This reduces its ability to discern the actual signal source [22]. This is especially true of the incredibly sensitive

10

avalanche photodetectors since they are capable of detecting single photons by liberating their entire charge on a detection event. Such detectors need to recharge faster than new photons arrive, or else the sensitivity is reduced proportionally, and the upstream components have to be able to detect smaller signals from the unprimed detector.

2. The detector can simply saturate. If it is exposed to far too much light, something that is easy for sensitive or high-gain detectors to do, the output of the detector or its amplifier will reach the highest voltage that it can provide. Further increases in light level (as might be provided by the transmitter switching on) will be ignored as it is physically incapable of reporting them [16] [20].

3. The receiver blindly reports the brightness of the light in front of the sensor. It does not know what the source of that light is. A reading of 2.45 volts might be due to the transmitter turning on, or the transmitter might be off and there is a lot of ambient light. Because the encoded signal also relies on the light's intensity, the decoder or receiver has to autonomously discard the ambient light levels, or it will try to "decode" it in place of the much smaller signal from the transmitter.

2.3 Signal Strength Changes

The last challenge is from changes in signal strength. Even with ambient-light filtering, the amplitude of the received signal coming from the detector is very important. If the amplitude decreases too much, the decoder will fail to receive any data, or it will trigger on noise instead.

At the same time, the gain of the detector cannot be too high or ambient light saturation could occur instead. Additionally, increased gain also increases the amount of noise in the system.

Chapter 3 System Implementation

The VLC prototype discussed in this thesis was designed from the ground up in all three parts: hardware, transmission protocol, and software. It is intended to solve a number of problems seen with other outdoor VLC designs and features increased range. The prototype VLC receiver uses hardware to clean up the received signal before it passes into the signal decoder, resulting in increased performance. The encoding and software are also designed to reduce the influence of noise on the signal for further improvements.

3.1 Hardware Design
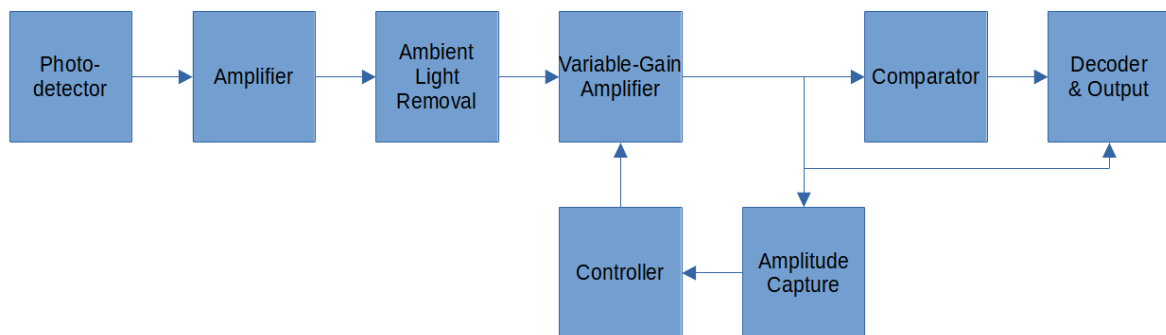
Figure 3.1 The high-level hardware block diagram for the VLC prototype's receiver

A well-known difficulty with operating outdoor VLC systems at range stems from the fact that the very medium they use to communicate is also shared with natural and artificial sources that are not attempting to send carefully encoded data. Additionally, interference may also stem from other VLC sources or improperly filtered non-VLC LED light sources. Lastly, for long-range VLC systems, the amplitude of the signal will decrease as the range increases, making it difficult to detect the signal. Yet, if the detector's gain were to be increased to better

detect distant sources, it would saturate on sunlight in the daytime and miss the transmitter's signal anyway.
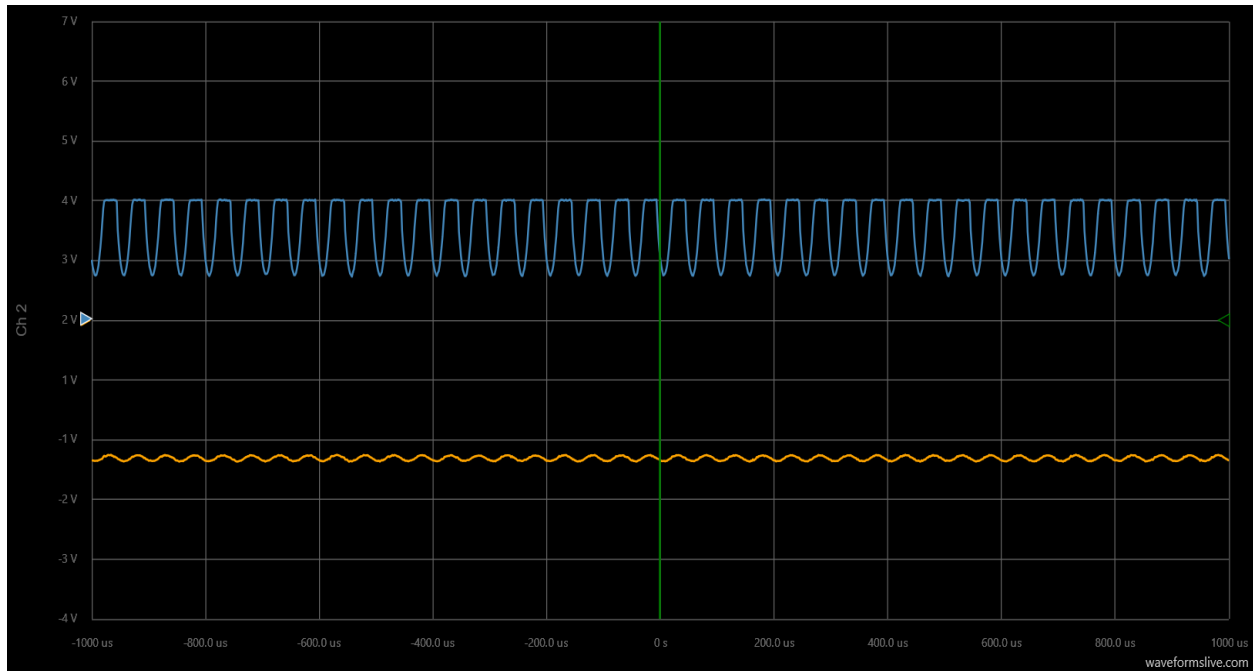
Figure 3.2 This image shows the saturation of the amplifier (blue) when a high gain is applied to a signal from a photodetector (orange). The detector has both a DC offset and an AC amplitude, but the amplifier applies the gain to both, cutting off the desired signal when it can no longer increase its output.

The proposed design of the VLC hardware in this thesis will convert the combined ambient light-and-data signal into a bipolar waveform that can be easily processed in both the analog and digital domains depending on the requirements of the user. Digital processing is faster and cheaper, but it may be more error prone. Analog detection is slower [23] but allows more advanced techniques to better remove noise. The design also features an improved multipart amplifier configuration that will allow for the detection of distant signals without saturating the detector. The addition of an improved clock recovery system is also possible but is beyond the scope of this thesis.

14
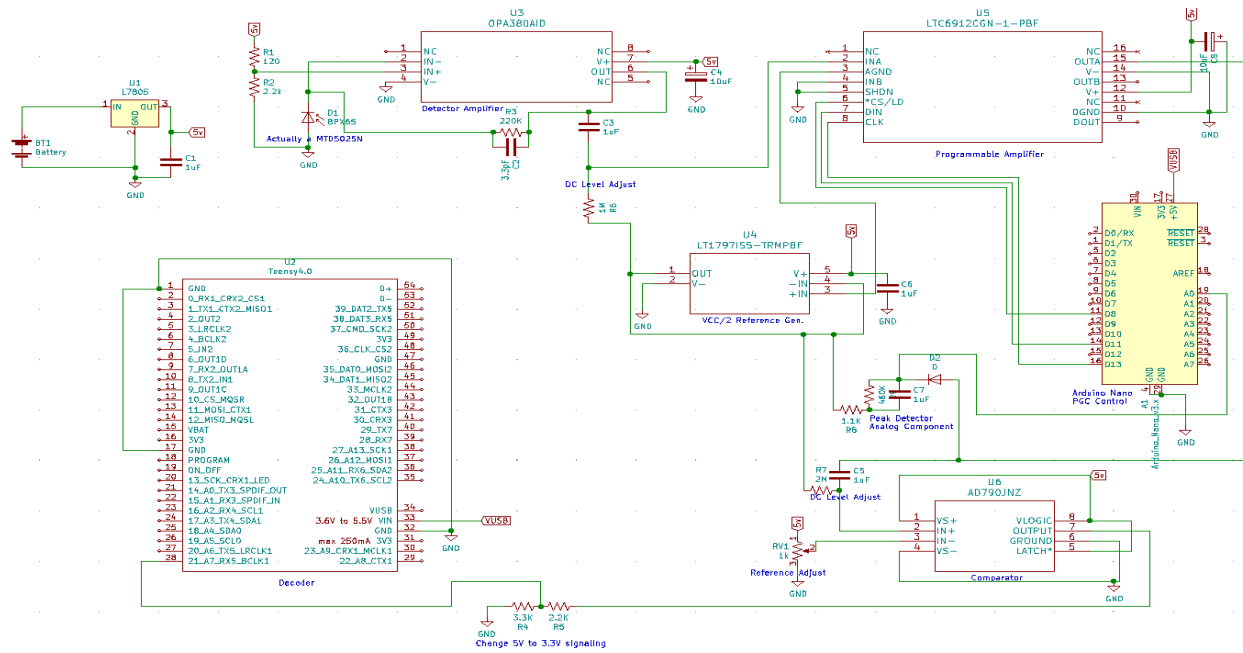
Figure 3.3 The schematic for the VLC prototype's receiver

3.1.1 Transmitter

The transmitter used with the proposed prototype was a 12V, 20-watt LED vehicle headlamp with an 8.17° vertical field of view and a 62.53° horizontal field of view. Modulation was established through an RFP30N06LE power MOSFET controlled by a Teensy 4.0 microprocessor (just like the one used in the receiver).
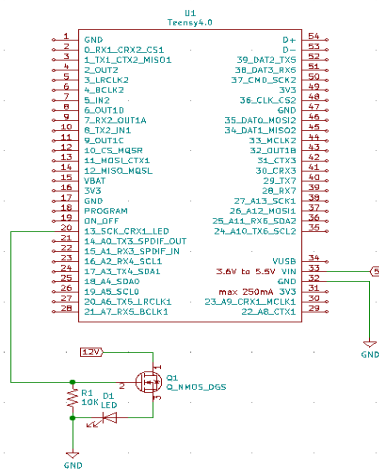
Figure 3.4 The transmitter circuit used for testing the VLC prototype

The Teensy microcontroller is responsible for digitally switching the LED at high speed to produce a fixed-amplitude digital signal.

3.1.2 Light Sensor

The VLC prototype's receiver is composed of multiple stages, as seen in Figure 3.1. The first stage of the receiver, the light sensor, detects and amplifies visible light, primarily in the blue light range. This is most directly associated with the original color of a white LED, and will have less time-domain blurring caused by the white phosphors [24], [25]. Optionally, an optical filter could also be added to further focus the receiver's sensitivity to only the light frequency of the transmitted signal [16], [22].
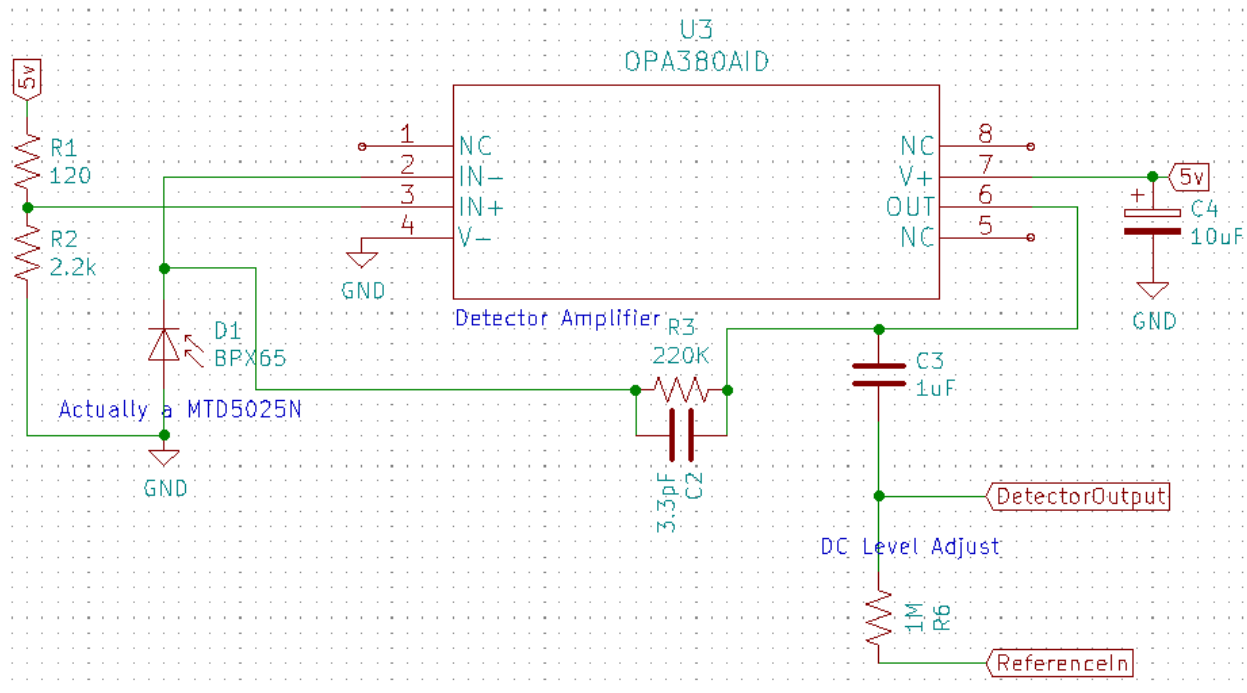
Figure 3.5 The photodetector stage's schematic

This stage is composed of a highly sensitive MTD5025N photodiode and an OPA380 op-amp designed for optical transimpedance amplification purposes. The prototype avoids ambient light saturation by limiting gain and by using a directional photodetector to ensure that the signal

being detected is from the transmitter, but this has also resulted in the prototype itself being highly directional. If this were to be put into production, multiple detectors would likely be needed to cover all angles the vehicle might receive from.

Electrical & Optical Characteristics (Ta = 25°C)

ITEMS	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNIT
Light Current	IL	*1	2.5	4.5	--	μA
Dark Current	ID	VR=5V	--	5	30	pA
Spectral Sensitivity	λ	--	--	410--580	--	nm
Peak Sensitivity Wavelength	λp	--	--	525	--	nm
Responsivity	S	*2	--	0.3	--	A/W
Angular Response	Θ	--	--	±12	--	deg
Shunt Resistance		VR=10mV	--	350G	--	ohm
Capacitance		VR=0V	--	100	--	pf

*1: VR=5V, λ=525nm, Ee=1mW/cm², *2: VR=0V, λ=525nm.

Table 3.1 Details of the photodetector used from the manufacturer's datasheet [26]

3.1.3 Ambient Light Filtering

Ambient light levels usually change slowly, so the output of the detector is capacitively coupled through a high-pass filter (C3 and R6 in the schematic above - see Figure 3.5), causing the mostly constant component from ambient light to be bled off while the high-frequency signal is allowed to pass.
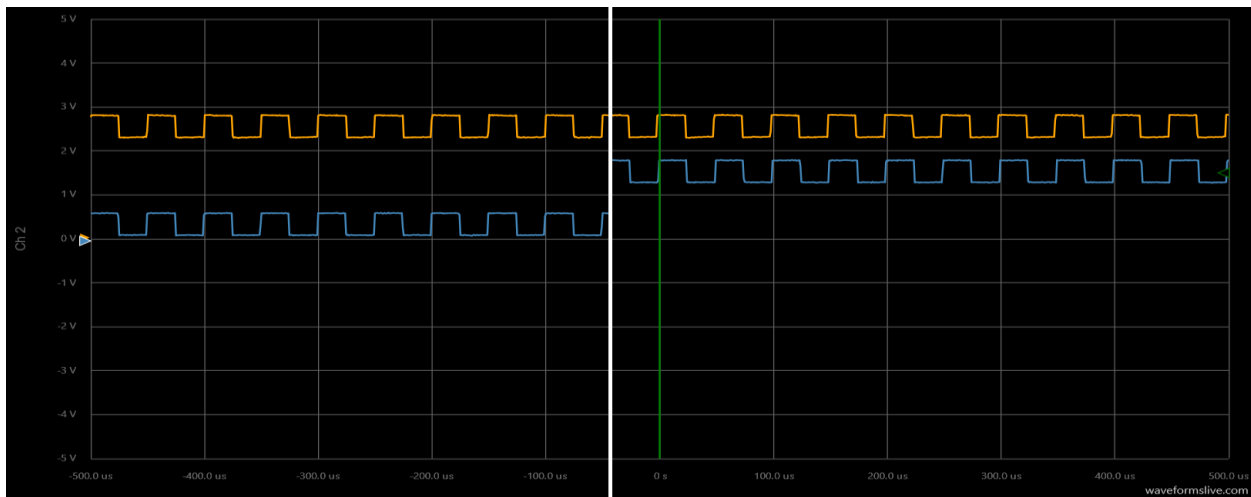
Figure 3.6 This oscilloscope trace shows how the output signal (orange) is unaffected by the DC voltage offset caused by ambient light on the sensor.

Another benefit of the filter is that its resistor is connected to a 2.5-volt reference. This converts the incoming signal into a bipolar one centered on the same 2.5 volts, making it suitable for both digital conversion and for more complex encoding protocols.
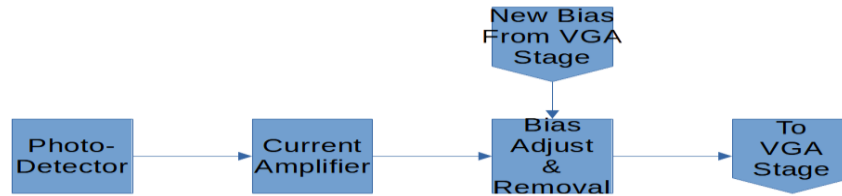
Figure 3.7 The photodetector and noise removal stages' hardware block diagram

3.1.4 Adaptive Signal Gain Subsystem

The adaptive signal gain stage of the receiver consists of several discrete components, a second, dedicated microcontroller, a buffer op-amp, and a LTC6912 variable-gain amplifier (VGA). The buffer op-amp provides the VGA's reference voltage to other parts of the system (such as the noise filter), but the voltage is also used internally to ensure the most stable and clean operation of the VGA stage itself.
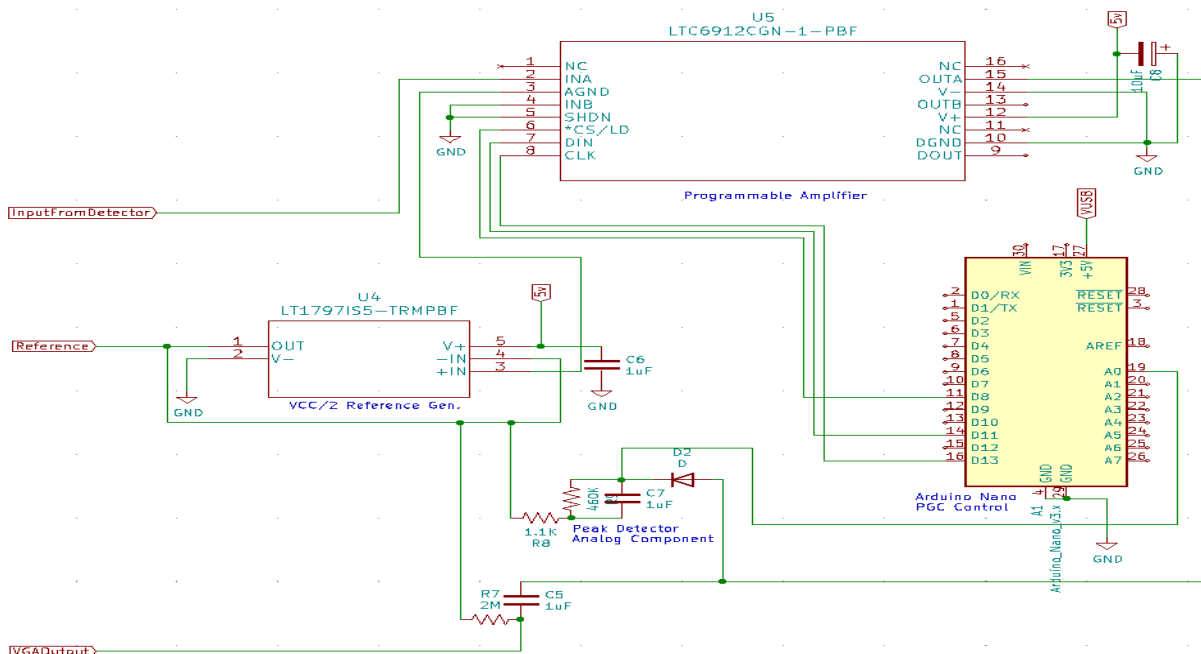
Figure 3.8 The VGA amplifier subsystem's schematic

The detector's signal, once it has been filtered, now consists of the VLC signal centered around a known reference voltage, but the signal could be weak and hard to detect if the transmitter is far away. Noise could be mistaken for the signal and cause an error. Conversely, if the receiver's gain is increased to detect distant signals better, ambient light could overwhelm the receiver and block out the actual data (as seen in Figure 3.2). The high-pass filter would then block the ambient light's signal, but there would still be no data present for downstream components to detect. However, using a second amplifier with adjustable gain allows the circuit to normalize small (or large) signals and scale the signal to a relatively fixed amplitude. This keeps the signal at a level that can be easily measured while minimizing added noise and keeping a safe gain in the detector.
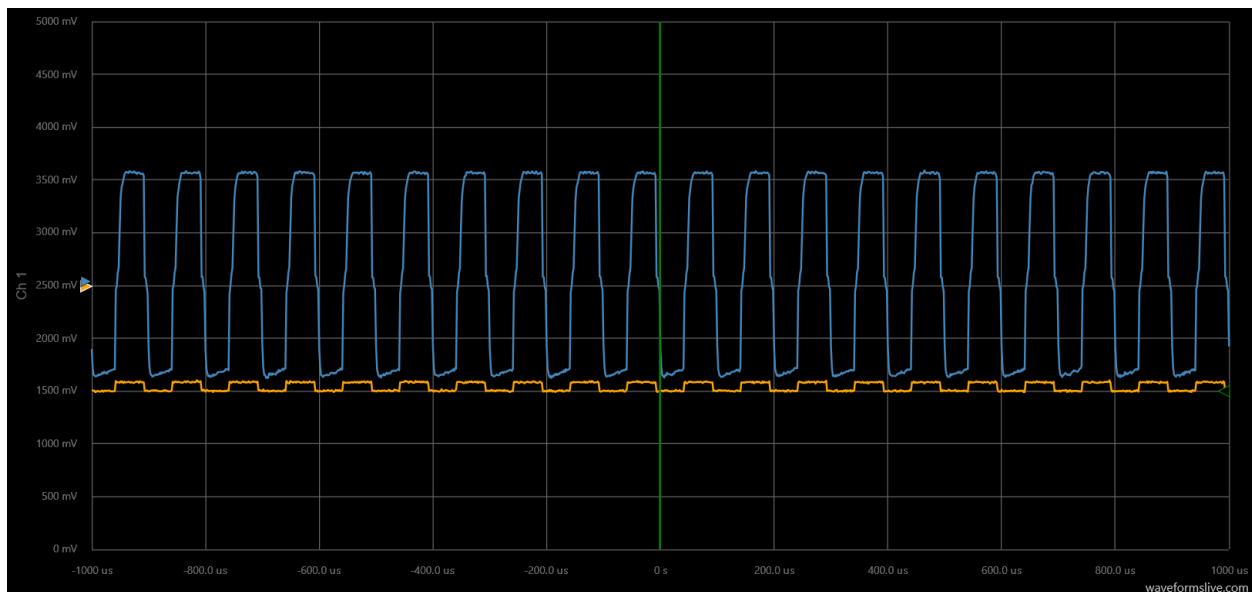
Figure 3.9 As can be seen in the image, the orange input signal from the detector has a DC offset of 1.5 volts from the first amplifier's gain acting on ambient light. Despite the gain, the signal amplitude is still small. The orange signal can be passed through the filter stage previously discussed and then to the second amplifier, which is now able to selectively increase only the signal and not the ambient light portion. The much larger, blue signal is the result.

The adjustment process begins with reading the amplitude of the signal via a diode and storing the value in a capacitor to detect the peak voltage, at which point a microcontroller (with an isolated power supply to reduce interference) determines if increasing or decreasing the signal

gain is necessary to maintain a minimum amplitude. The amplifier used in the VLC prototype has fixed gain levels, so the microcontroller must consider if it is safe to adjust the gain level if the new gain produces a signal outside the configured bounds in the opposite direction. If this is the case, it will choose the option that keeps the higher of the two potential gain settings to avoid the signal potentially becoming too small to detect.
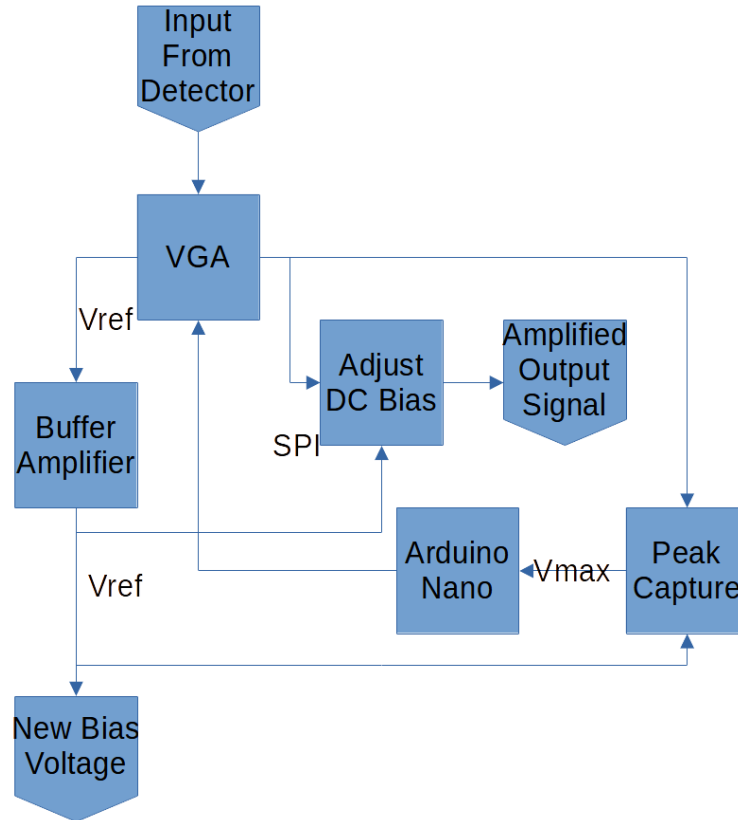
Figure 3.10 The VGA amplifier subsystem's block diagram

The controller also has a small waiting period to ensure that the low or high amplitude reading that would cause a change is not simply due to a spurious signal but that it is actually needed. The output of this stage is, ideally, a signal with a larger amplitude that is easy to interpret for the later stages of the receiver (see Figure 3.9).

UPPER/LOWER NIBBLE				NOMINAL VOLTAGE GAIN		MAXIMUM LINEAR INPUT RANGE ($V_{P\text{-}P}$)			NOMINAL INPUT IMPEDANCE (kΩ)	NOMINAL OUTPUT IMPEDANCE (Ω)
Q7 Q3	Q6 Q2	Q5 Q1	Q4 Q0	Volts/Volt	dB	Dual 5V Supply	Single 5V Supply	Single 3V Supply		
0	0	0	0	0	−120	10	5	3	(Open)	0.4
0	0	0	1	−1	0	10	5	3	10	0.7
0	0	1	0	−2	6	5	2.5	1.5	5	3.4
0	0	1	1	−5	14	2	1	0.6	2	3.4
0	1	0	0	−10	20	1	0.5	0.3	1	3.4
0	1	0	1	−20	26	0.5	0.25	0.15	1	6.4
0	1	1	0	−50	34	0.2	0.1	0.06	1	15
0	1	1	1	−100	40	0.1	0.05	0.03	1	30
1	0	X	X	0	−120	10	5	3	(Open)	(Open)
1	1	X	X	Not Used (Note 11)					Not Used	

Table 3.2 The gain levels of the VGA as listed in the datasheet [27]

3.1.5 Digitization

The next stage of the receiver is built upon an AD790 precision comparator. The incoming signal from the second stage varies around a center voltage, so by comparing this signal with a selected reference voltage at or near the center, the AD790 will rapidly slew its output to produce a digital result representing whether the voltage of the incoming signal is greater than the reference signal.
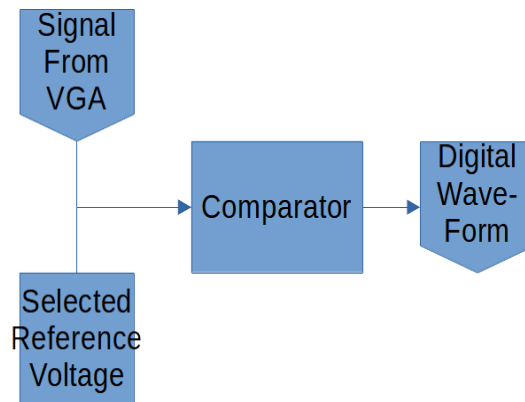
Figure 3.11 The comparator section's block diagram
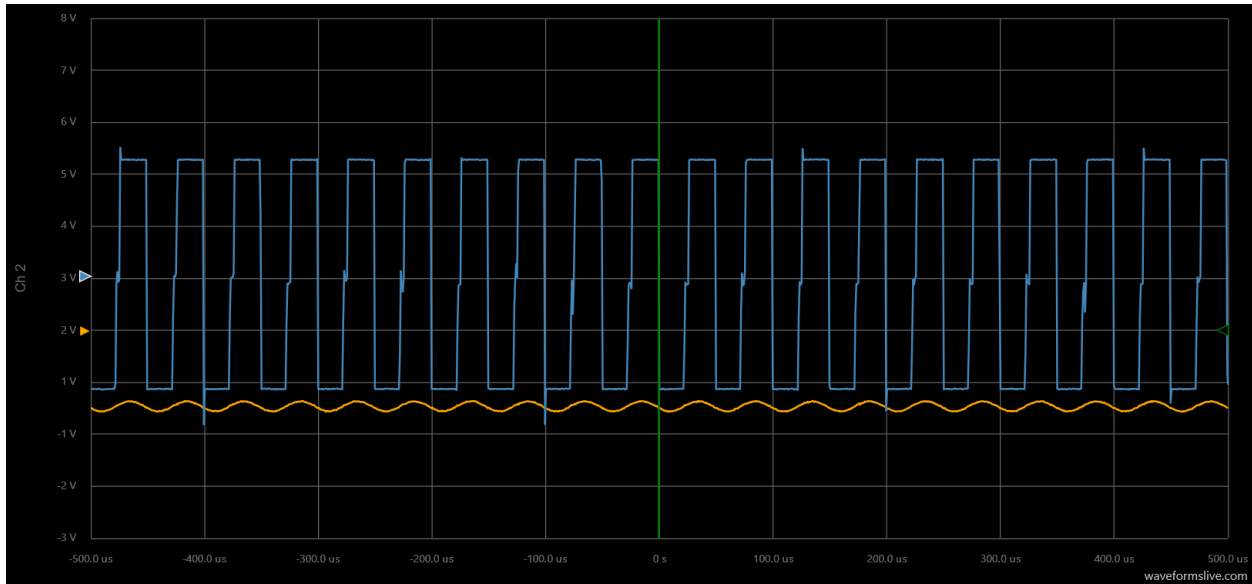
Figure 3.12 This graph shows the input and output voltage waveforms of the comparator. When the input (orange) crosses the reference voltage (not shown due to limitations of the test equipment), the output (blue) switches almost instantly to 0 volts (if the input is below the reference) or 5V (if the input is above the reference).

The reference adjustment is set manually, once, to choose a point in the signal near the VGA's bias voltage where it rarely if ever deflects due to noise. In ideal operation, the VGA keeps the incoming signal large enough that received data passes the chosen voltage level and causes the comparator to output a digital "1." This result may then be interpreted directly by the receiving microprocessor.
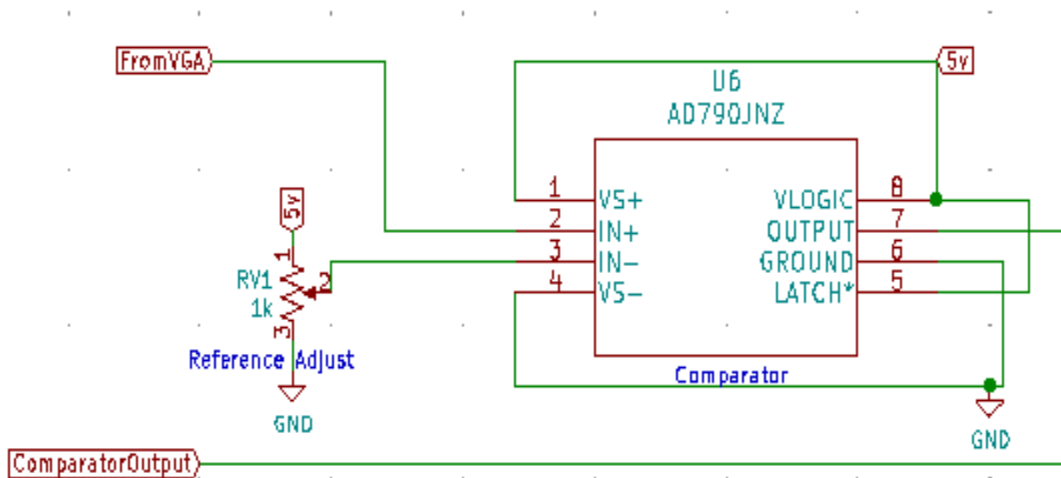
Figure 3.13 The comparator section's schematic

3.1.6 Decoding
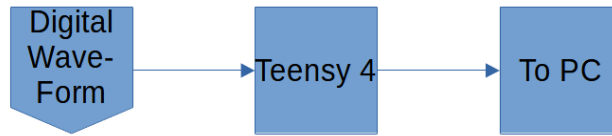
Figure 3.14 The block diagram of the decoder

The decoding system consists of the receiving microprocessor, a Teensy 4.0. This device was chosen because of its high-speed CPU and abundant memory [28]. The Teensy reads the digital signal from the AD790 via pin 21, detects a start-of-frame sequence, and then starts clocking the data into a shift register to be recovered and transmitted to an observing computer.
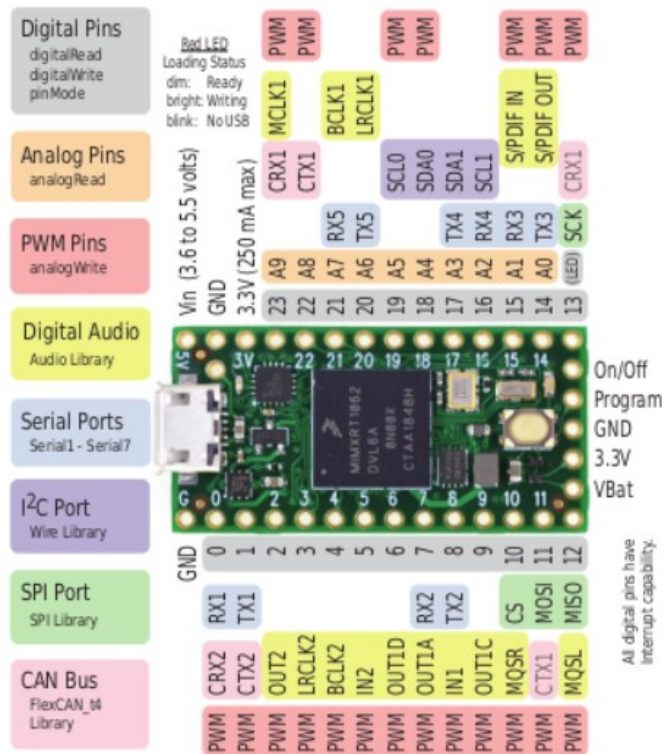
Figure 3.15 The pinout of the Teensy 4.0 used as the decoder [28]

The 5V signal from the rest of the logic passes through a voltage divider since the Teensy is 3.3V logic and could otherwise be damaged. Like the other microprocessor used in the gain control system, this one also has an isolated supply to avoid interfering with the VLC signal.

U2
Teensy4.0

Pin	Left	Right	Pin
1	GND	D+	54
2	0_RX1_CRX2_CS1	D−	53
3	1_TX1_CTX2_MISO1	39_DAT2_TX5	52
4	2_OUT2	38_DAT3_RX5	51
5	3_LRCLK2	37_CMD_SCK2	50
6	4_BCLK2	3V3	49
7	5_IN2	36_CLK_CS2	48
8	6_OUT1D	GND	47
9	7_RX2_OUT1A	35_DAT0_MOSI2	46
10	8_TX2_IN1	34_DAT1_MISO2	45
11	9_OUT1C	33_MCLK2	44
12	10_CS_MQSR	32_OUT1B	43
13	11_MOSI_CTX1	31_CTX3	42
14	12_MISO_MQSL	30_CRX3	41
15	VBAT	29_TX7	40
16	3V3	28_RX7	39
17	GND	27_A13_SCK1	38
18	PROGRAM	26_A12_MOSI1	37
19	ON_OFF	25_A11_RX6_SDA2	36
20	13_SCK_CRX1_LED	24_A10_TX6_SCL2	35
21	14_A0_TX3_SPDIF_OUT		
22	15_A1_RX3_SPDIF_IN		
23	16_A2_RX4_SCL1	VUSB	34
24	17_A3_TX4_SDA1	3.6V to 5.5V VIN	33
25	18_A4_SDA0	GND	32
26	19_A5_SCL0	max 250mA 3V3	31
27	20_A6_TX5_LRCLK1	23_A9_CRX1_MCLK1	30
28	21_A7_RX5_BCLK1	22_A8_CTX1	29

Decoder

3.3k R4   2.2K R5   FromComparator

GND

Change 5V to 3.3V signaling

VUSB

GND

Figure 3.16 The schematic of the decoder section

Alternatively, the output of the automatic gain control subsystem may be imported directly into the receiving microprocessor's Analog-to-Digital Converter (ADC), specifically to pin 14, for slower but more precise measurement. This feature is used later for the Code Division adaptations.

3.2 Communication Protocol Design

A very common initial approach to VLC, and one used in this prototype as well, is for the designer to use a variant of a simple digital protocol like UART or Manchester encoding to modulate and demodulate the transmitted data [12], [13], [17], [19]. Manchester encoding is even mentioned as an option in the IEEE 802.15.7 standard [29], [30]. This approach has the benefit of masking noise at higher signal amplitudes and works well with a variety of hardware and software decoders. Unfortunately, it was discovered to not be very robust because at low

amplitudes the digitizer detects noise in place of the signal. An alternate method also explored in this project is the transformation of the received signal into a bipolar format that makes possible the use of modern techniques from the field of radio frequency transmission (see the output of the ambient light filter in Figure 3.6).

Regardless of the exact form of the signal, wireless communication frequently only has a single link between the sending and receiving sides of the wireless connection. (VLC is an exception to this in that it can feature multiple channels through using different frequencies of light.) In contrast, a wired connection frequently has more than one wire, allowing the transmitting side to send a clock signal or synchronization pattern through a second channel alongside the data. However, using an additional link for non-data purposes is inefficient, slower, and costs more, so protocols have evolved to allow data to be sent without a clock signal or in a way that one can be recreated on the receiving side from context cues. These are known as asynchronous protocols. They feature independent clocks on each end of the channel so that the receiver reads the contents of the channel at the same time as the sender writes data to it. The clocks can drift out of synchronization in short order, however, so some degree of overhead (though usually less than that needed for a synchronous protocol) is still required [31]. The clocks are typically synchronized by a framing signal, which is named such because it usually "frames" some amount of data. The result is that asynchronous protocols can use fewer wires or channels than a synchronous one, or that they can use those channels for increased throughput.

As an example, one of the simplest such protocols is the Universal Asynchronous Receiver-Transmitter (UART), which requires both sides to have a shared clock frequency to read the bits being sent at the correct rate and uses rudimentary synchronization in the form of a start bit that prepares the receiver for reception when the line changes from the idle state. The

start bit controls the receiver's clock phase and aligns the received bits. An unsynchronized receiver will eventually drift toward proper synchronization simply because the data being sent will not quite line up with the proper start bit's location. UART is suited mainly to wired serial applications where the signal is strong, however, because it is very sensitive to noise and timing drift. Any noise would, at best, immediately corrupt a bit and has a roughly ten percent chance of disrupting synchronization. As a result, more complex protocols should be used to support VLC.

3.2.1 Manchester Encoding Scheme

The asynchronous Manchester encoding was the first protocol considered for the VLC prototype's transmission method, as it is more robust than UART and was even incorporated into some older VLC standards, such as IEEE 802.15.7 [29], [30]. The digital protocol is purposely designed to avoid ambiguity issues during long, repeated bit strings, and does so by encoding each transmitted bit as two alternating logic levels in sequence. A "1" bit would be encoded as logic levels "10" and a "0" bit as "01." This means that there is a level transition that takes place in the middle of each received bit for easier clock recovery and that a long sequence of identical bits still has level transitions, allowing the decoder to accurately count the bits [32].
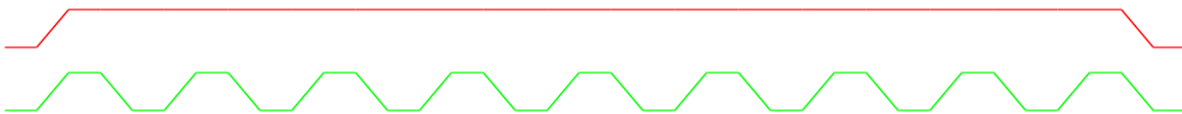
Figure 3.17 An example of a direct transmission (red) of the signal "1111…" vs a Manchester-encoded version (green) of the same signal. Note how the first transmission's bits blur together while the second is still distinct!

There are more complicated or more efficient encodings, but Manchester requires low amounts of processing and overhead, so it initially appeared to suit the microprocessor-based hardware used for the prototype.

3.2.1.1 Synchronization and Framing

A complication arises, however, because the transmitted encoding must alternate during each sent bit. If two already-alternating bits are sent (the message is originally "10" or "01"), there is no transition between the two bits once encoded. The decoder must therefore detect not only the level transitions, but the amount of time between them. If the next bit is the same, there is an extra transition before the bit, but if they differ, the next transition takes twice as long but is the next bit. As a result, the protocol requires the decoder to accurately know during startup whether it is seeing an intra-bit or an inter-bit transition, since all following data must rely on that knowledge to detect further bits properly.
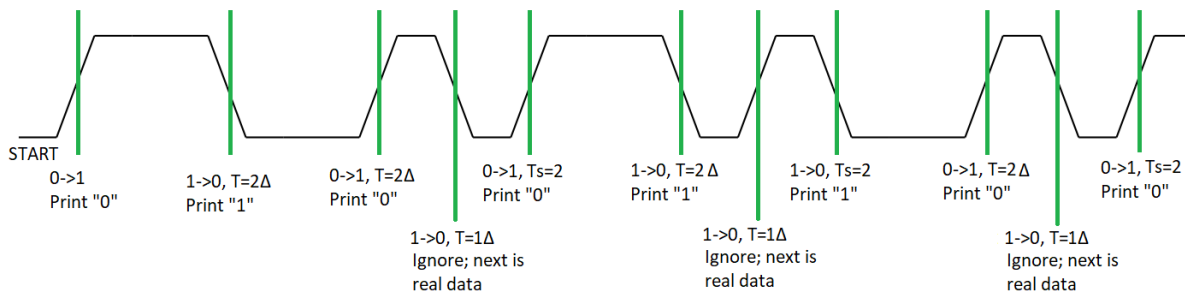
Figure 3.18 A pictorial showing the level interrupts (green lines) and the time calculations that allow the decoder to interpret their meaning.

In addition, noise on the channel can induce level transitions or transients that are hard to compensate for or detect. A transient signal might soon return to the actual value, but it will have already changed the state of the decoder. If one of these events occurs, or the light signal is occluded, data will be lost, or the decoder may even acquire an additional and incorrect bit in its message. If error detection was used, the message can be marked as invalid, but the decoder still needs to figure out how the incomplete data it has can be resynchronized to the transmitter's information. A partial solution to this problem was to make the decoder only sensitive to light level transitions around the time that the transmitter was expected to send one. In this way, only

transitions at the end or middle of a bit were accepted. It still allowed erroneous data through, but

only if it occurred near when an actual transition was supposed to happen.
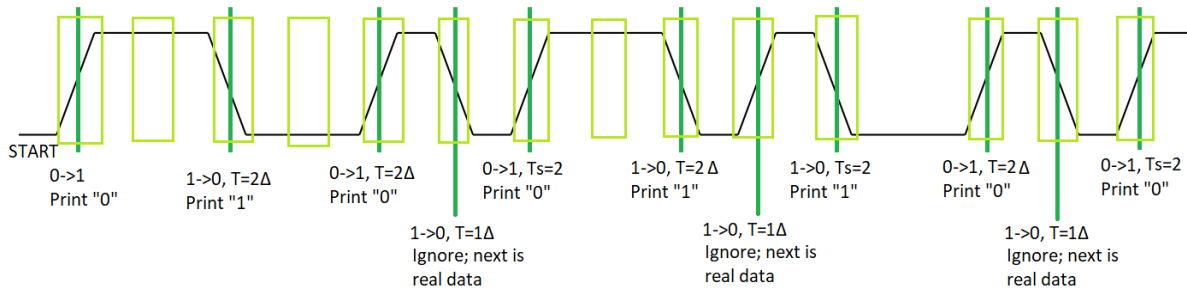
Figure 3.19 Light green boxes represent the times that the decoder can accept level transitions (not to scale). Transitions outside the borders are ignored, reducing noise.

An alternate solution to the timing window approach might have been to have the

decoder activate periodically rather than on each level transition, but the use of a clock instead of

using the level triggering negates many of the benefits of using Manchester encoding. A major

risk would be timing drift causing the system to miss a level change, and the decoder still may

encounter errors if a transient occurs during sampling. In addition, level-triggered techniques are

more efficient, since the incoming data can be processed in an interrupt, and the time-triggered

approach still does not solve the issue of lost data due to interruptions in the signal.
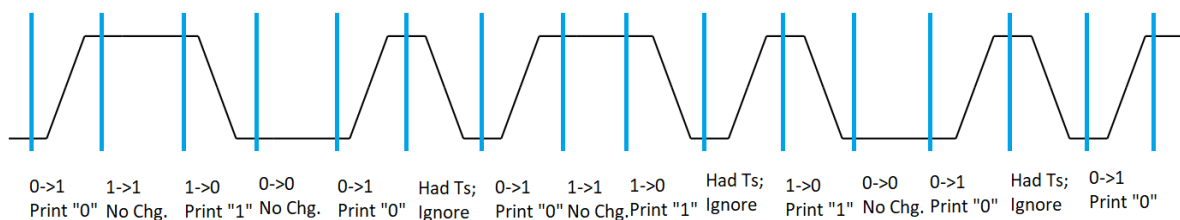
Figure 3.20 This diagram shows how a time-triggered mechanism might read the same bit sequence. Instead of checking the transitions themselves, it periodically reads the signal level (blue lines) and computes whether it has changed from the last level seen, which would indicate a bit (but only if two bit-times have passed). However, this method requires more processing power and has a risk of drifting across bit boundaries, resulting in errors.

A better solution to both problems was the addition of a packet framing sequence (see

Figure 3.22 for what a framing sequence looks like). Without framing, if the decoder lost a bit

due to occlusion, for example, it would still read eight bits into a message byte, but in doing so

would now consume a bit from the next message before marking the byte as "complete" (this is partially because the input to the decoder is activated by the level transitions of the Manchester encoding and has to process bits that already take different times to arrive). At this point, the next message would also be missing a bit, causing the problem to repeat indefinitely. This problem can occur to any type of transceiver, not just to one particular encoding technique.
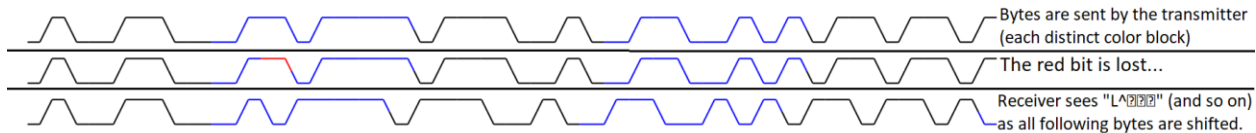
Figure 3.21 A diagram showing how one lost bit can cause a chain reaction that corrupts all following bits in the sent data. The output in quotes is the output of the corrupted data, which consists of invalid characters.

The addition of a framing sequence forces the decoder to resynchronize its message bytes to the ones being transmitted to avoid this sort of ongoing data corruption problem. The receiver will periodically stop and wait before resuming reception when it detects the start-of-frame signal. This ensures that if it gets out of synchronization, the data corruption is blocked from spreading further when the receiver forcibly resynchronizes with the transmitter. Conveniently, this approach may be used alongside one of the other noise-reduction methods mentioned above. Since the code for that had already been written, and worked better with the framing sequence detection methods, the interrupt-driven version was used.

3.2.1.2 Packet Format

The framing method used in the VLC prototype is to send a short preamble to set the channel to a known state followed by a sequence of eight transitions with timing that would not be easily reproduced by noise. After the preamble and frame, the transmitter sends four bytes' worth of data. This creates a packet where at most four bytes of data can get corrupted by any single noise or transient event before the decoder will halt and wait for a new start signal. A

29

blocked signal (caused by something moving between the sending light source and the receiver) can take longer to resume, as the decoder needs to complete its current message and then start looking for a new start sequence. In rare cases, it could take up to eight bytes of data to resynchronize if the new start sequence arrived during the period in which the light was blocked. However, while fewer bytes would be lost in a corruption event if only one byte were sent per frame, the framing sequence is approximately one and a half times the length of a byte, so the data-to-overhead ratio is greatly improved when sending multiple bytes.

Figure 3.22 The data sent in one frame of the Manchester encoding in the VLC prototype. It features four bytes of data, noted in alternating blue and green. The red portion is the start-of-frame signal.

3.2.2 Code Division Encoding Scheme

The Manchester coding scheme previously discussed was quite feasible, but synchronization was difficult to achieve. At long ranges, or with more than a small amount of noise present, the framing sequences themselves could get corrupted, causing the receiver to drop the entire frame, and the prototype began having trouble resynchronizing in general since every time a frame was corrupted via an off-by-one error it would also fail to find the next frame. Additionally, the signal was nearly imperceptible at fifty feet, and the tuning of the hardware to detect this started to break down at this distance. As a result of these discoveries, the VLC prototype was modified to use a code-division algorithm in hopes of improved range or error-resistance, but at the cost of the data rate. A small amount of error detection was also added to the prototype to reduce corrupted data after some was observed in the output of the decoder during the Manchester attempts.

Code Division was predominantly developed for mobile phones to allow multiple handsets to talk to a base station. The protocol is largely immune to noise since it selectively amplifies only the actual signal intended for that receiver. As a side effect, it is also highly resistant to interference from other transmitters, which is what allows the mobile phones to ignore communications from the other phones in the network [32].

The original protocol encodes data either on top of mutually orthogonal vectors (called "chips") or as a stream of pseudorandom bits. The decoder then correlates the historical patterns of the received data with its vector or with the pseudo-random seed shared with the transmitter. If the chip is not yet fully received or if it is intended for a different receiver, there is no correlation. However, once the chip finishes arriving, it aligns with the receiver's copy and produces a strong signal indicating that a bit has arrived and what its value is. This solves the bit synchronization problem since the decoder can now identify with certainty if and when it is seeing a bit, and it also reduces the influence of other transmissions on the channel. Noise is reduced as well, since it will only corrupt a small portion of the signal even as the rest remains highly correlated [32].

1	1	-1	1	1	1	-1	-1
1	-1	1	-1	-1	-1	1	1
1	-1	-1	-1	-1	-1	-1	-1

Dissimilar Chip Sequence Demo   Correlation: -6

1	1	-1	1	1	1	-1	-1
1	-1	1	1	1	-1	-1	1
1	-1	-1	1	1	-1	1	-1

Shifted Chip Sequence Demo   Correlation: 0

1	1	-1	1	1	1	-1	-1
1	1	-1	1	1	1	-1	-1
1	1	1	1	1	1	1	1

Matching Chip Sequence Demo   Correlation: 8

Table 3.3 This table shows how different chip sequences (the top two rows of each block) do not correlate (the third row of each block plus result), or how even the same chip sequence will not correlate if it does not perfectly align.

The Code Division protocol ordinarily requires a bipolar signal, where the bits of the chips are represented as 1 and -1 with an average signal level of 0, but the transmitter for this

prototype emits only 0 and 1 logic levels. However, the receiving hardware converts the original digital signal into a bipolar signal in the process of filtering ambient light (see Figure 3.6). The resulting bipolar signal may now be used for Code Division since the transmitter's 0s become -1 levels in the process of the conversion. The decoder can detect the presence of both the 1 and -1 signals over the channel average of 0, allowing the use of Code Division and signaling whether the channel is in use (all signals would be 0 if it were idle). The bipolar signal differs in another important way from a regular binary signal as well. When another device is transmitting, the -1 and 1 bits add a level of random noise, but it does not actually interfere with the reception of the chosen transmitter. If this were a binary encoding, 0 and 1 bits sent at the same time would add together to form a 1 bit, resulting in an incorrect message.
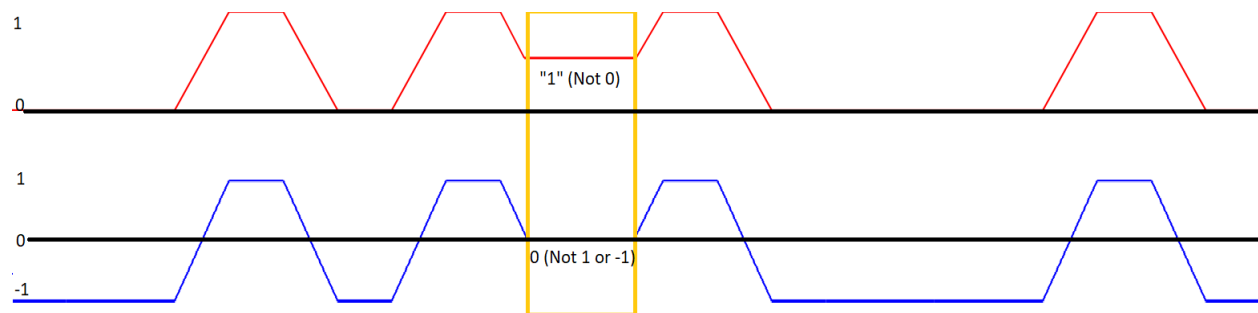
Figure 3.23 In this diagram, the difference between a digital and a bipolar signal can be seen. Both signals are affected by a positive interference spike (yellow bounding lines, representing another transmitter sending a 1), but the digital signal (red) is subtly corrupted, while the bipolar signal (blue) is detectably incorrect and can be handled to mitigate the error. A bipolar detector can detect the ground state (black lines) as well as the excited states of the line, whereas a digital version only detects that the signal is non-zero.

The prototype samples each bit of the chip sequence twice, but one of the samples is effectively discarded as the double-sampling is only used to passively keep the decoder in phase with the transmitter's signal. As a result, the ability to detect the chip sequences is limited to what is essentially a majority voting operation, which limits the operating range from what is possible with this technique. Even so, the code-division VLC decoder has improved performance

(a gain in range of up to twenty feet was observed) and synchronization over the Manchester design.
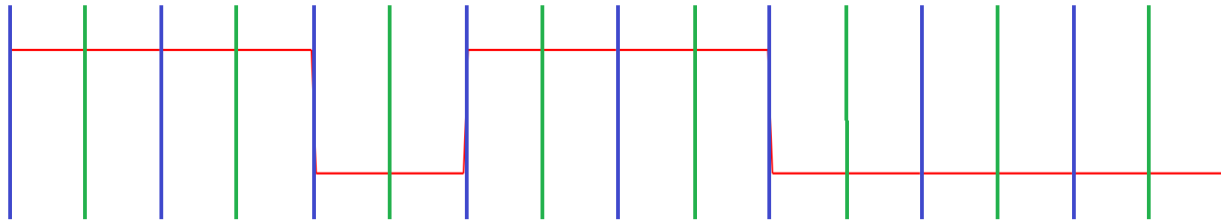
Figure 3.24 The double sampling of the decoder can be seen in this diagram. The green lines (here, the even samples) were in-phase with the signal and will be kept for use in decoding the signal. The blue lines (odd samples) overlapped with the bit transitions and thus may be inaccurate, so they will be discarded. The decoder does not initially know the alignment of the samples, so it must sample twice as frequently so that at least one of the sets will be in-phase.

3.2.2.1 Synchronization and Framing

Because the VLC unit designed in this thesis was a prototype, the multiple access functionality of Code Division was less immediately important than the ability to filter noise and improve synchronization. To that effect, the prototype used three different chips for data transmission for improved separation of the potential data being sent. The three chips were eight bits in length each, with four bits of each logic level for a constant average signal level. One chip signified a 1-bit, a second signified a 0-bit, and a third, the starting or S-chip, was used for framing the data being sent. Unlike in the Manchester design, the decoder does not need priming to detect bits properly, since the chips arrive at a known rate and are instantly discernable once they have arrived. Either or both factors can be used to verify the accuracy of the current chip. However, the reassembly of the bits into readable data still requires framing to convert the received bitstream into a sequence of bytes. If a chip were to be lost, the same off-by-one issue as seen in the Manchester decoder could corrupt all following data.

33

```
S: {1,1,0,1,1,0,0,0}
0: {1,0,1,0,0,0,1,1}
1: {0,0,1,0,1,1,1,0}
```
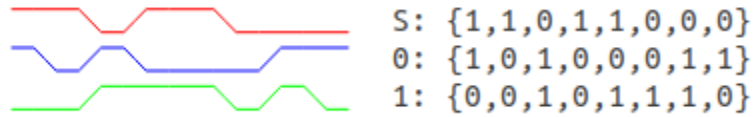
Figure 3.25 The three chip sequences used in the Code Division version of the VLC prototype. The red sequence is the start sequence, the blue version is a 0, and the green version is a 1.

The VLC prototype recognizes a chip if more than six bits of the chip match with either a 1- or 0-bit during the receive state or a start chip if it is synchronizing. To facilitate synchronization, both timing and correlation are used to detect the presence of a chip. Only both together will confirm that one has arrived. The decoder continually scans for the start chip during the acquisition stage. Once the start chip is detected, the decoder knows that the following data chips are going to finish arriving at fixed times (offset from when it sees the complete S-chip) and thus only performs the correlation operation once the chip has finished arriving, rather than every time a bit of the chip is received.

3.2.2.2 Packet Format

After the start sequence there are eight data chips, one for each bit of the byte being transmitted. There is also an even-parity code sent after the data chips, intended to reduce invalid data that makes it through the Code Division or that results from a dropped connection due to an obstructed light beam. This is sent as an additional 1- or 0-chip after the data chips, for a total of 10 bytes sent per real byte of data. Again, the intended advantage of this technology is for improved resilience and range, but it reduces the available speed for any single receiver device, which is a classic tradeoff in channel design.

Figure 3.26 An example character encoded via the Code Division protocol. The red chip is the start signal, the blue chips are 0-valued, and the green chips represent 1. The last item in purple is a parity chip also with the value of 1.

3.3 Software Design

The original design specification of the VLC prototype included a queue for preparing signals to be sent over the VLC link, but it was removed early on. It proved easier for testing to use a fixed message programmed into the transmitter since the fixed message was easy to check for data errors. An additional benefit was that the transmitter now operated in a standalone format and did not need an attached PC. The software was written in C++ in the Teensyduino environment and used timer (and pin, in the case of the Manchester design) interrupts to activate in response to events.

3.3.1 Encoding

The encoding system was built on a Teensy 4 processor that would control the state of a digital GPIO pin to turn the headlamp on and off. A state machine was implemented that would perform the pin state changes to create the desired signal. State transitions were triggered by a timer register exceeding a set bound, which triggered a function in the regular code outside of the interrupt. The timer continued to increment even while the regular code operated, ensuring that there would be no timer drift.

3.3.1.1 Manchester Transmitter

A selected ASCII character sequence was chosen as the data to be sent. For the Manchester system, this was four bytes to fit the frame size being used (see Figure 3.22). The state machine for the transmitter was segregated into two parts. Both parts operated similarly, in that they would trigger after a specific time, but the first part would invert the pin state after a set list of delay times to form the start-of-frame signal and the second operated on fixed times and set the pin state based on the value of the bit in the sequence to be sent. In the second mode, the

first value set was the actual state of the bit and then, half of the overall bit time later, the inverse

of the bit would be sent to form the Manchester code for that bit's value.
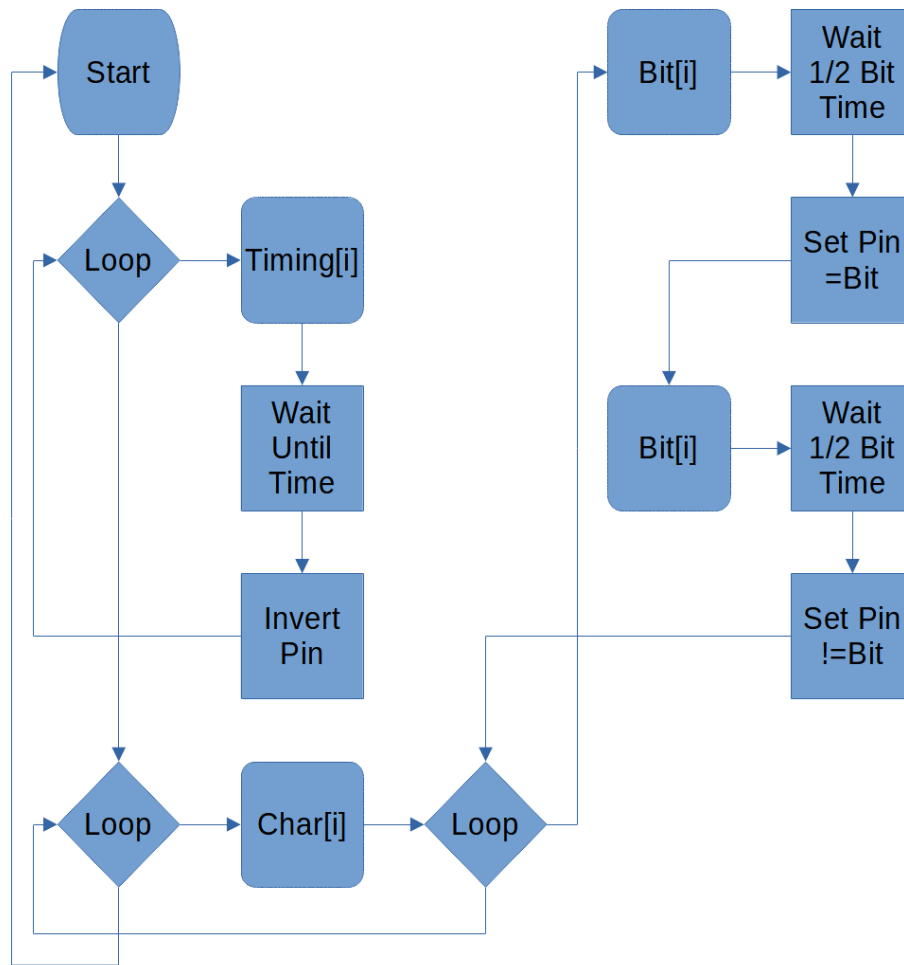
Figure 3.27 The flowchart of the Manchester encoding transmitter code

3.3.1.2 Code Division Transmitter

The Code Division transmitter was more complex than the Manchester version. The Code

Division system had three sections to its state machine, and rather than output the value of the bit

(or the inverse of the bit), it had to index into and then send voltage levels corresponding to the

bits inside the associated chip sequences. The three states, in order, sent the start sequence, sent

the eight sequences associated to the bits of the byte being sent, and sent the parity sequence

associated to the odd parity of the byte computed via XOR. The bit time for the Code Division

system was slightly larger than that of the Manchester system, both because of the larger chip compared to the simpler Manchester sequence but also because the decoder was more complex and could only process data so quickly.
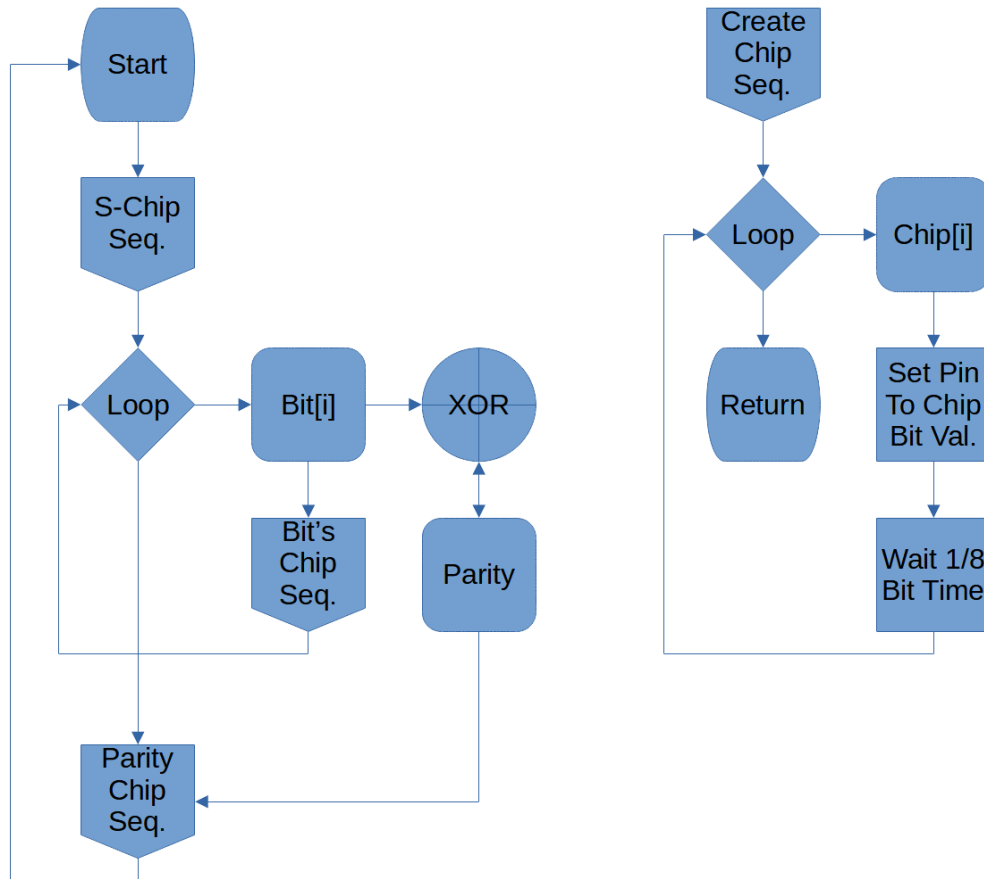
Figure 3.28 The flowchart for the Code Division transmitter's software

3.3.2 Decoding

Both decoders were far more complex than the transmitter code (compare Figures 3.27 and 3.30 or 3.28 and 3.33). Where the transmitter had to simply step through bits, the decoders had to continually sanity-check the incoming signal without the benefit of a real clock input or even a firm digital signal (in the case of the Code Division design). Signal lock-on was acquired through continually testing the incoming signal to see if it adhered to a known pattern, at which case the mode would switch to reading the actual data. This was how the decoder handled the

framing of the sent data. If the signal it was detecting did not match with the start signal, it would

wait idly until the start sequence was detected and it could resynchronize. This was especially

important in the case of the Manchester decoder, where unexpected noise in the signal could

cause a perceived bit to take an unpredictable amount of time, resulting in a loss of alignment

with the sent information.

3.3.2.1 Manchester Decoder

The Manchester Decoder software contained the twin of the transmitter's framing

sequence in its own memory. Every time the pin interrupt occurred, it would save the system

time that it happened. The background task would then compare the time interval between the

interrupt and the previous one to determine if the time delay matched that of the start sequence.

If it did, the state machine would advance one stage. If the time no longer matched, it would

reset to the beginning. Once the state machine had read the entire framing sequence (minus a

quick high-low transition to reset the channel, only present in the transmitter's sequence), the

decoder switched modes to read the following data bits.
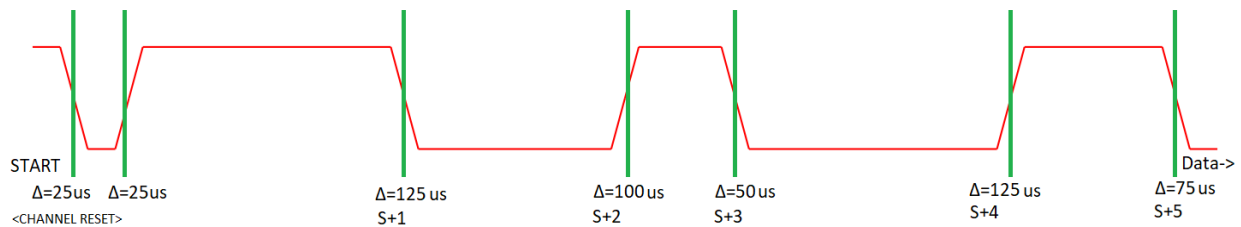
Figure 3.29 This figure shows how the decoder's state machine detects and interprets the timing of the start
sequence. The green lines represent the level interrupts, and the decoder measures both their direction and the time
since the previous one to determine their validity. If both match the stored values, the state advances until the start
sequence is complete and data can be sent.

It was here that the design got slightly complicated. In the event of a repeated bit, the

signal would transition after the end of the bit. This resulted in an extra transition which had to

be discarded. However, if the bits had alternating values, the next transition would be in the

middle of the next bit, which had to be kept. To handle this properly, the decoder continued to log the time at which the bit transition had occurred. If it took two half-bit-times, it was a new bit. But if it took one half-bit-time, it was the inter-bit transition, and the next transition could be expected to be the real bit (see Figure 3.18). Either way, once the bit was detected, the direction of the interrupt (rising or falling) was used to determine the value of the bit. A falling edge was a 1 (a 1→0 sequence; since it started with a 1, it represented a 1), and a rising edge was a zero (a 0→1 sequence). Every eight bits (a value additionally reset by the arrival of a new frame, to ensure alignment of the bits in each byte with those being sent), the collected bits were shifted into a byte and sent to the attached PC to be validated.

Figure 3.30 The Manchester decoder software's flowchart
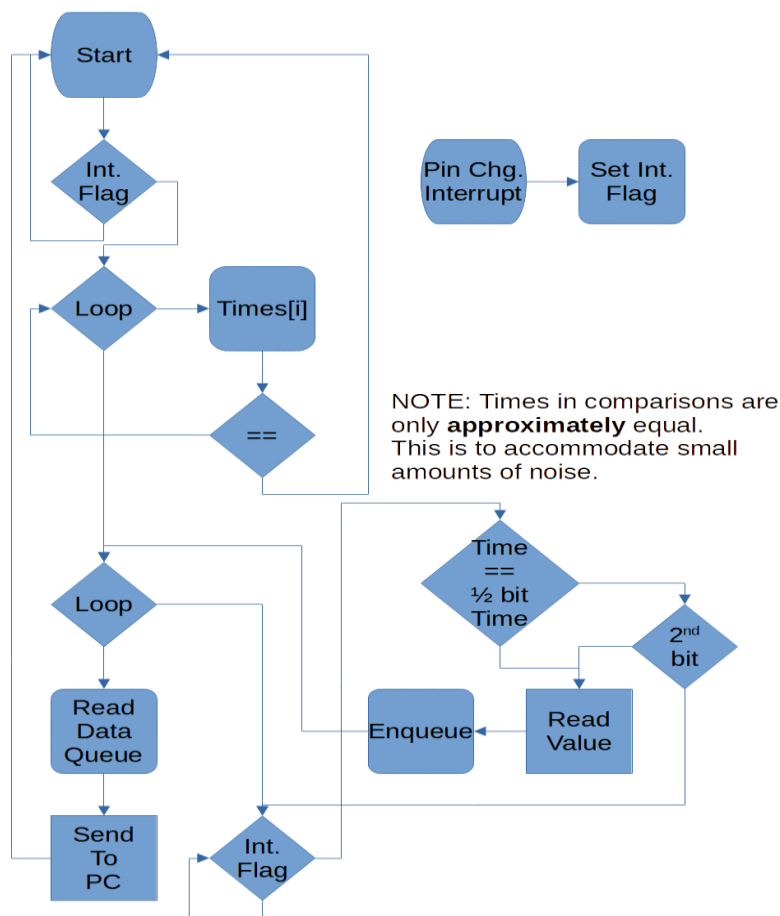
39

3.3.2.2 Code Division Decoder

The Code Division decoder went entirely into a time-triggered design after the experience with uncertain or inaccurate bit-times in the Manchester model. Since there was no external synchronization source even here, alignment with at least some phase of the transmitter's output was ensured by sampling the analog value of the signal at twice the fastest rate that bits could change in the incoming data stream. Each sample corresponding to the same phase in the incoming stream was inserted into a different queue. For convenience, these will be referred to as even and odd samples. Every two samples during the acquisition phase, the decoder will compare the values of the samples with the steady-state level of the channel (taken as a long-term exponential average for this prototype; the chip sequences were all designed to have an average level of 0 by containing an equal number of +1 and -1 levels in each) to create the +1 and -1 readings of this slice of the queue. The +1 and -1 levels are then correlated with the desired chip sequence to detect a hit.

-1	1	-1	1	-1	1	1	-1	Correlation:	2
-1	-1	1	-1	1	-1	1	1	Correlation:	-6
1	-1	-1	1	-1	1	-1	1	Correlation:	2
1	1	-1	-1	1	-1	1	-1	Correlation:	2
1	1	1	-1	-1	1	-1	1	Correlation:	0
-1	1	1	1	-1	-1	1	-1	Correlation:	-2
1	-1	1	1	1	-1	-1	1	Correlation:	0
1	1	-1	1	1	1	-1	-1	Correlation:	8

Figure 3.31 This graphic shows the arrival of the start chip and how the detector can "lock on" to its exact arrival time by checking the correlation of the arrival queue with the known chip sequence's value. The detector thus finds both the start sequence and the exact time it arrives.
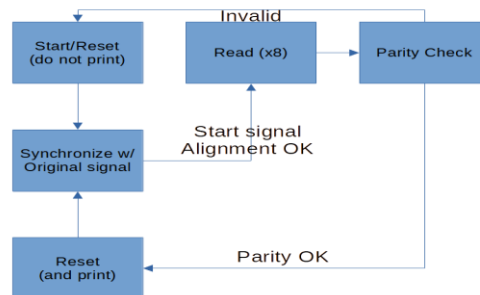
Figure 3.32 The decoder's state diagram

Once one of the queues aligns with the start chip, the decoder will use that phase of the clock (for this example, the even phase - see Figure 3.24) to read in the next set of chips that represent the byte being transmitted. For eight chips it will read in the signal, perform the correlation to find the most likely bit value, and then shift the bits into a holding register to be prepared for transmission to the validating PC (see Figure 3.32). After those eight chips, it performs one last receive-and-correlate to get one more bit. This bit is compared with the parity of the received byte to determine accuracy. If they match, the byte is sent to the PC. For a reminder of the format, see Figure 3.26.
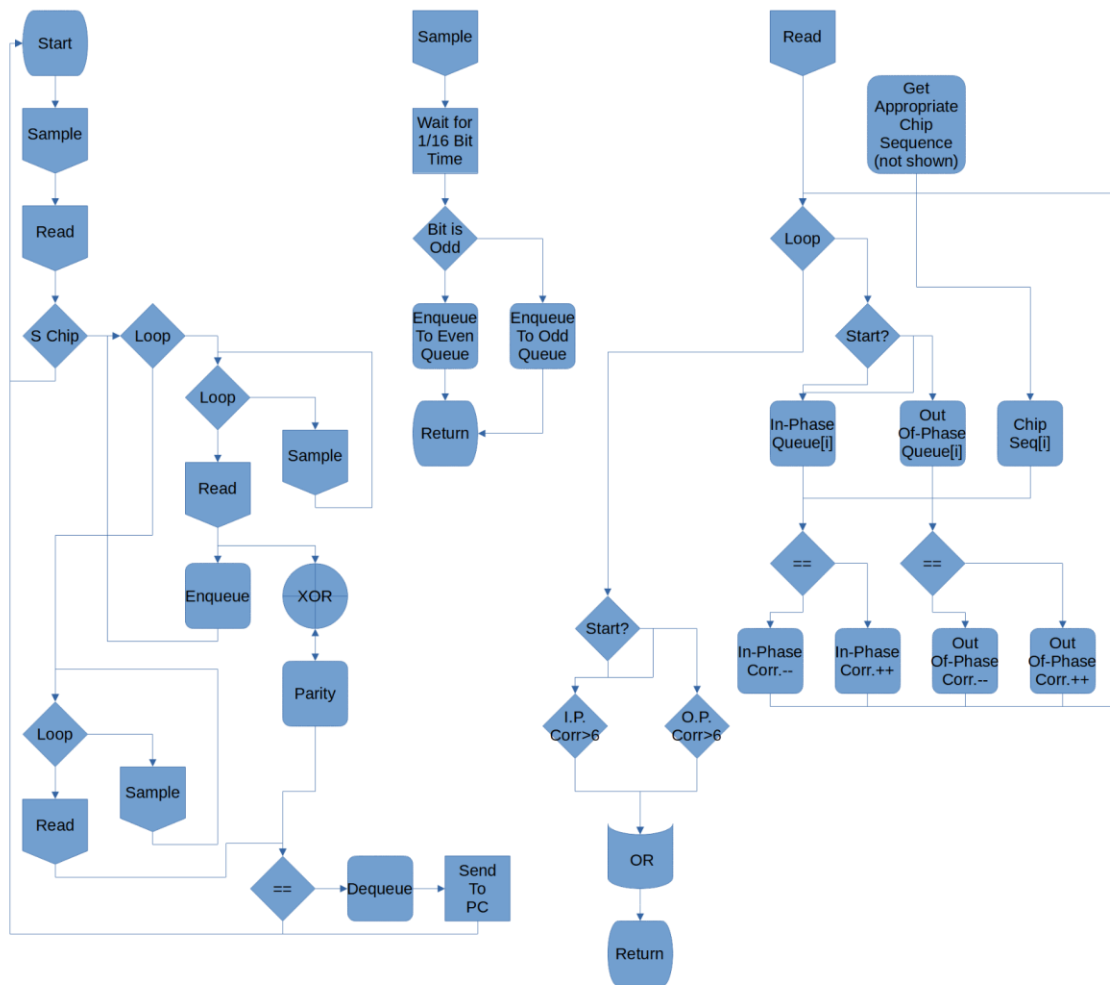
Figure 3.33 The flowchart for the Code Division decoder's software design

41

3.4 Testing

Some of the preliminary versions of this prototype were tested indoors in the Institute for Advanced Vehicle Systems (IAVS) at the University of Michigan-Dearborn, a very large building that is brightly lit by both natural and artificial sources. This provided a great environment for testing the system's vulnerability to interference.
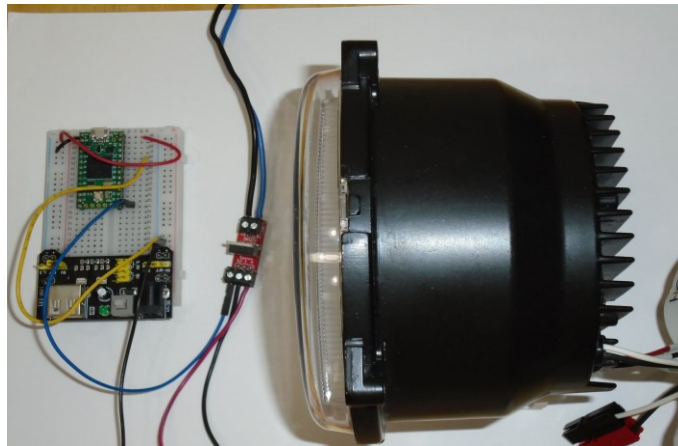
Figure 3.34 The VLC transmitter used throughout the testing of the prototype
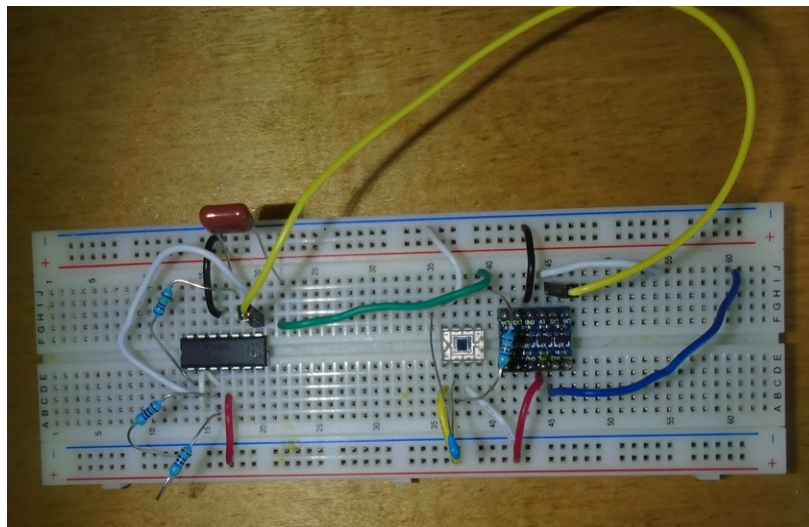
Figure 3.35 The earliest prototype built

The first prototype was based around an Arduino Nano at 16 MHz and used UART as its protocol. This was the prototype that showed the issues with using UART over a noisy channel, as it was very difficult to get it to sync up properly. In addition, the photodetector used was not

very sensitive and the components did not respond quickly. In dark conditions, it would easily

receive too little light and fail to respond. Too much light was also a problem, but it did achieve

a range of 11.5 feet during moderately lit conditions.
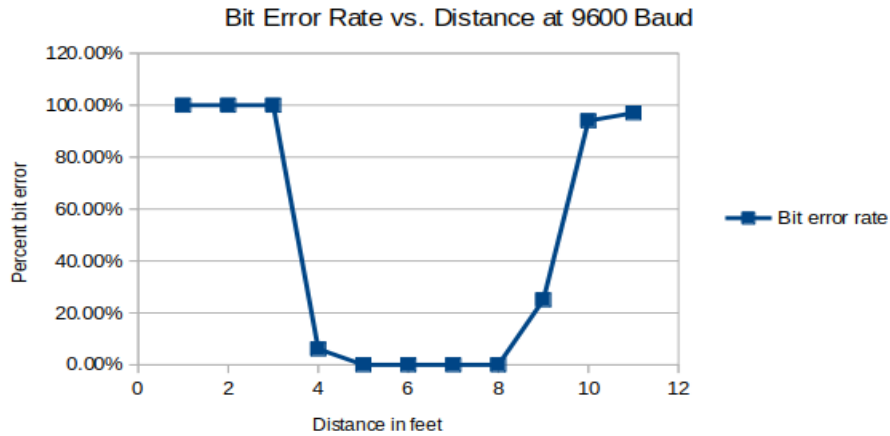
Figure 3.36 Bit error rate vs. distance (measurements taken at night with first prototype)
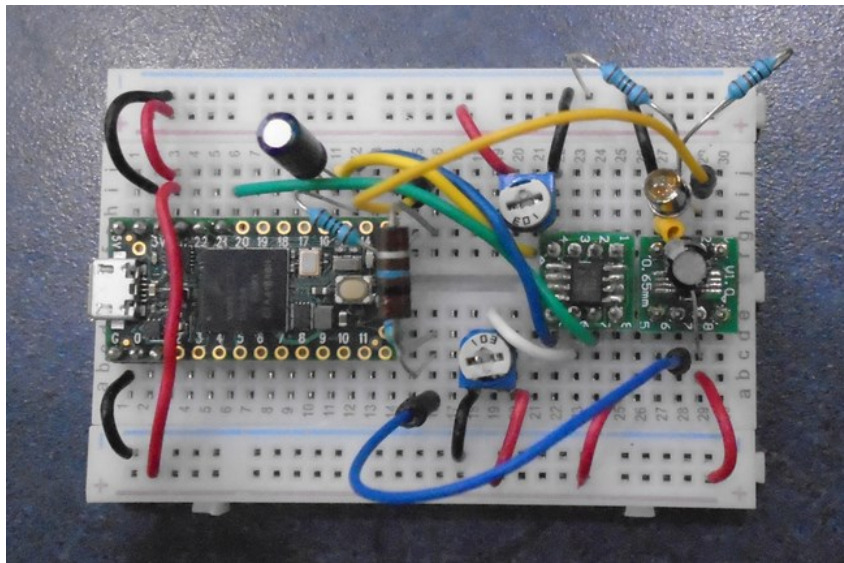
Figure 3.37 The second prototype built and tested

The next iteration of the design featured a much more sensitive photodetector from

Marktech (the same one used all the way to the current model) and a new transimpedance

amplifier rather than the all-in-one solution used in the first design. This proved to be much more

sensitive and was capable of higher data rates. It also was where the Teensy 4.0 appeared as the

43

microprocessor of choice for encoding and decoding the signal. However, an inadvertent step backward was made when it was decided to use a new digitizer that needed two potentiometers. It required manual tuning to adjust for changing light levels, and the range was a bit low, despite some improvement.
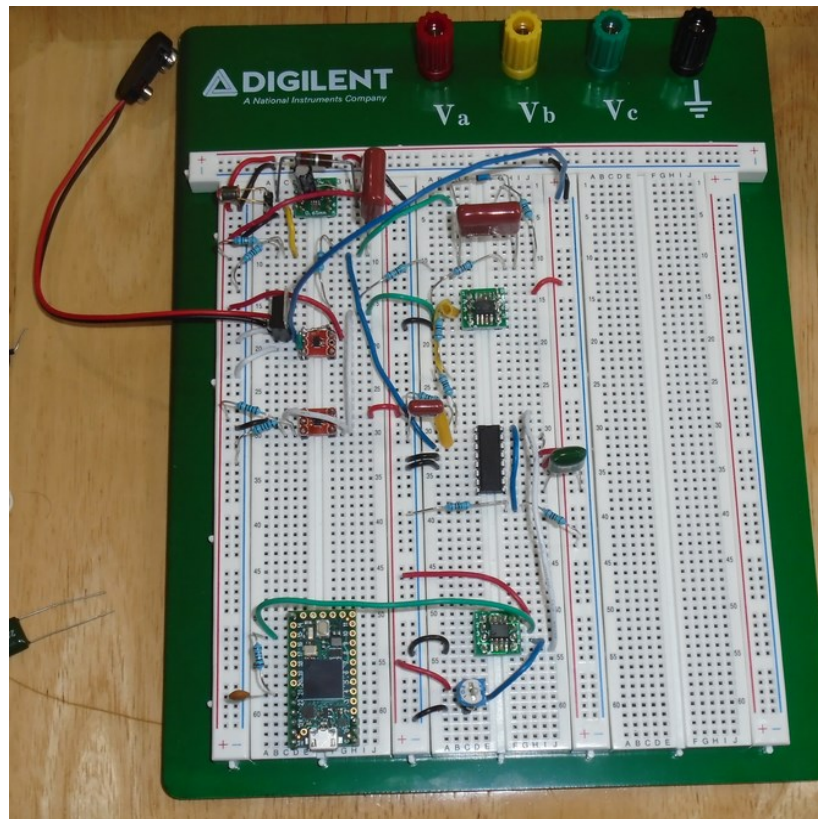
Figure 3.38 The third prototype of the VLC receiver, and the last to be built on a solderless prototyping board

The third version was redesigned to again feature automatic ambient light removal, and attempts were made to add a second amplifier that would increase gain as needed for detection of even very weak signals. Unfortunately, despite much effort and the purchase of many seemingly promising components, the Covid lockdown prevented effective technical consultation on the elimination of ongoing and unpredictable issues with signal oscillations. As a result, the automatic gain control system never worked as intended. Several setbacks regarding unsuitable

components also occurred during this time, causing further difficulties in getting the system working.
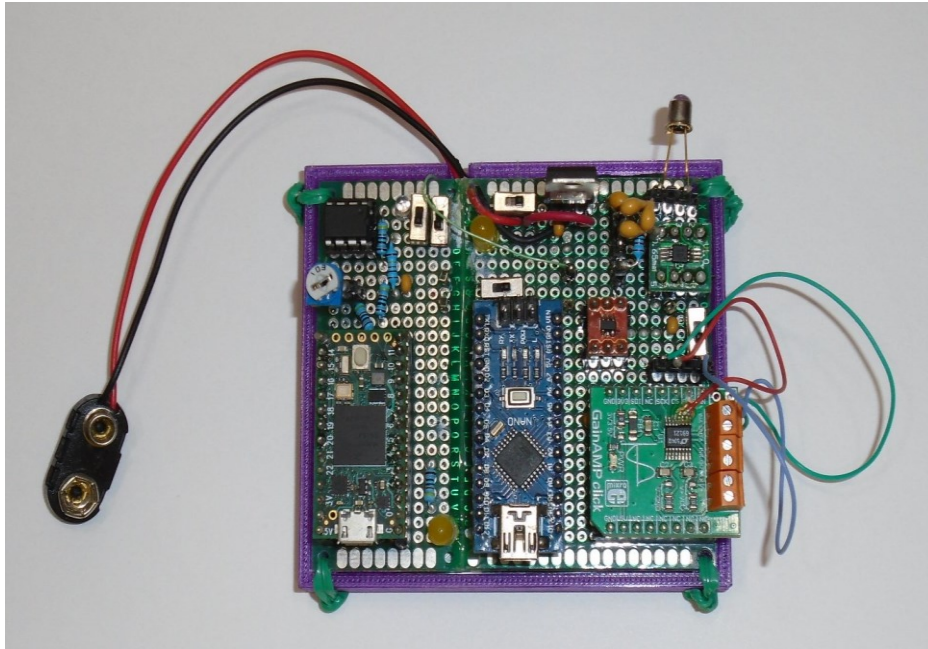
Figure 3.39 The final prototype receiver

Eventually, work was halted on the third-generation receiver and efforts were switched to a new version of the receiver with a digitally controlled second amplifier in the hopes that this would stop the oscillation issue in the amplifier's feedback circuit. This proved successful, although the sudden gain steps from the digital amplifier did cause short noise spikes in operation, compared to the smooth gain scaling of the analog-controlled one in the prior prototype (when that one worked). This final version drew upon all the knowledge gained in the previous generations and worked without adjustment in all light levels.

Testing for the last two prototypes was performed outdoors both in bright sunlight (noon) and at night to explore the impact of extreme light conditions. This was done because other designs have historically had difficulties with operating with ambient light or bright daylight [12]–[14], [16], [17].

Figure 3.40 Outdoor testing of the VLC prototype in daylight (transmitter, left; receiver, right)

The ambient light filtering worked as designed, and a minor increase in performance was observed during the day tests. The system's range functioned between zero and seventy feet before total signal loss was observed. The test entailed moving the prototype in a straight line away from the transmitting headlamp in distances of five feet before checking the data arriving from the receiver module. Once signal loss was observed, the receiver's position was fine-tuned in distances of one foot to find both the exact location of signal loss and the first location that errors started appearing.
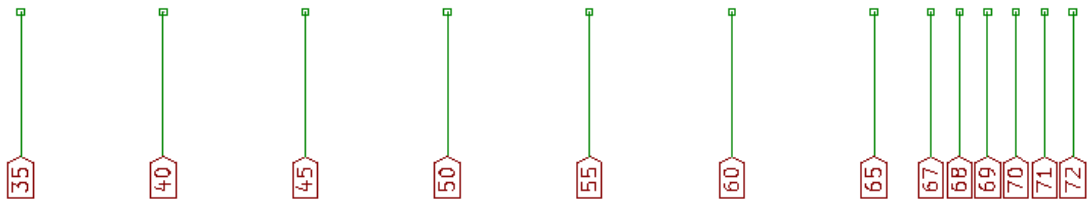
Figure 3.41 An illustration of how measurements were performed, in a case where the signal was lost at 70 feet

Each test was performed twice, on two different days with similar conditions (weather, time, and location), and the exact range was carefully checked multiple times per test to be sure that the limits were accurate. The distance at which near-total signal loss occurred with the code-division receiver was seventy feet during the day, but the signal contained few errors until it passed sixty-six feet (sixty-four and sixty-one feet at night, respectively). At this point errors

began occurring at a rate of one error every 20-40 bits on average where before there had been almost none. Manchester was observed to remain almost perfectly error-free up to fifty feet, but it abruptly failed to receive at all at fifty-two feet under daylight conditions. During the night the Manchester receiver failed at fifty feet while working perfectly at forty-eight feet.

	Manchester, Day	Manchester, Night	Code Division, Day	Code Division, Night
Max. distance with few to no errors (feet)	50	48	66	61
Distance at which effective signal loss occurred (feet)	52	50	70	64

Table 3.4 Average distances of testing results

Ambient light levels during the daylight testing averaged 135,000 lux, with 52,000 lux at the sensor. Night testing verified an ambient light level of 0 lux. During the night testing, it was dark enough to measure the light levels from the headlamp. At fifty feet from the transmitter, its power level was 33 lux. At sixty feet, that signal had decreased in intensity to 16 lux.
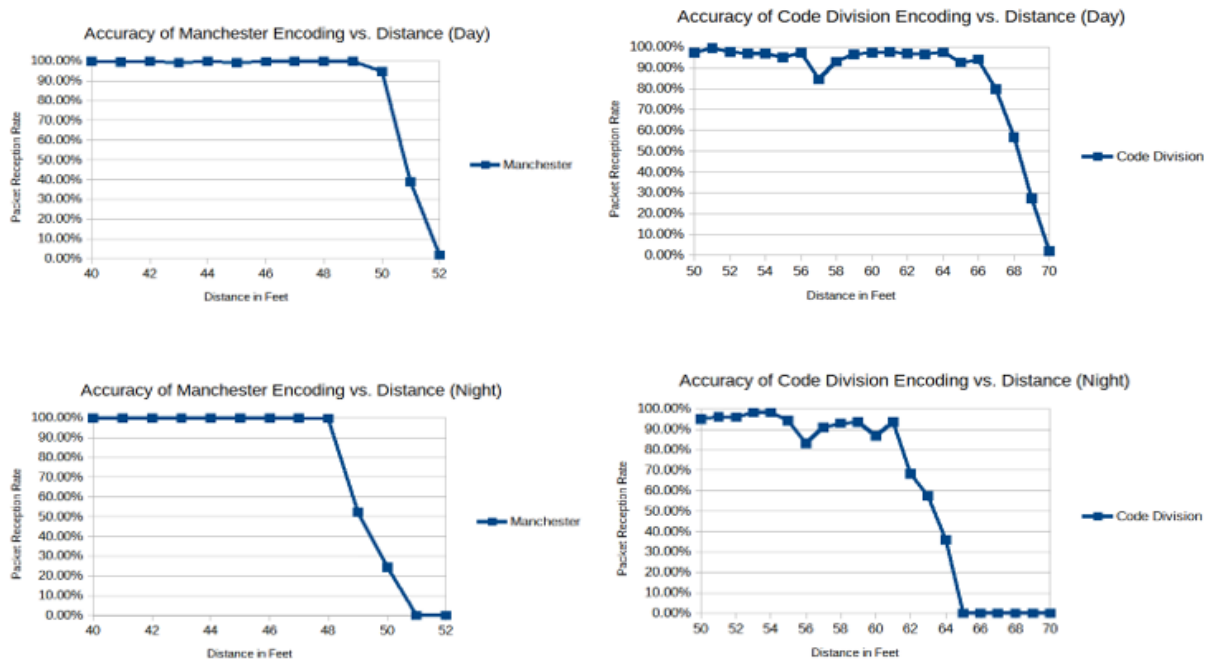
Figure 3.42 Comparison of Manchester and Code Division encoding error rates until loss of signal

Another important metric of testing was the data rate of the overall system. As mentioned earlier, an ongoing problem with the prototype was in getting sufficient speed out of the decoder. Manchester, with its low overhead and interrupt-triggered design, was able to achieve 920 bytes per second, or 7360 bits per second. This was from a channel rate of 20,000 transitions per second. The code division system had a much greater degree of overhead in the decoder and redundancy in the signal to obtain its range increase, and was therefore capable of 313 bytes per second, or 2504 bits per second, with a base channel transition rate of 12500 transitions per second.
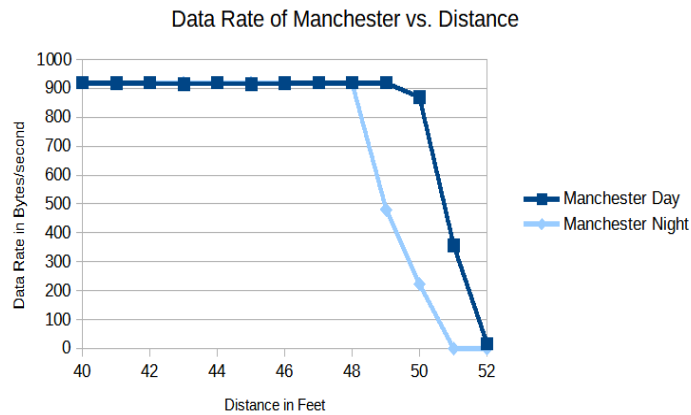
Figure 3.43 Effective data rates of the prototype using Manchester encoding
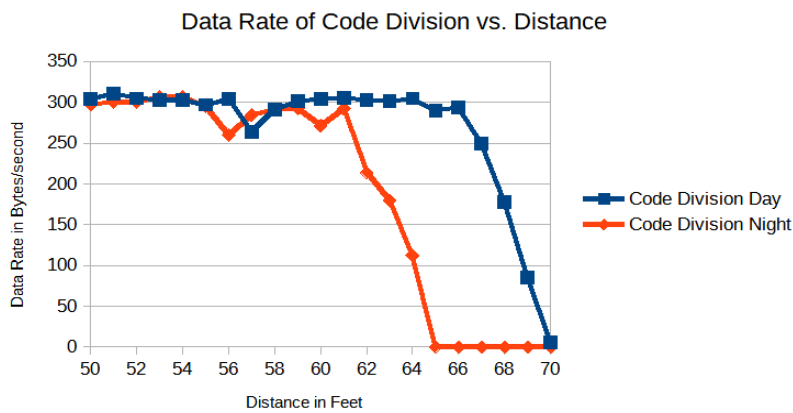
Figure 3.44 Effective data rates of the prototype using Code Division encoding

The prototype was originally conceived as a unidirectional point-to-point link and was never intended to deal with a multi-user or interference-containing environment. However,

testing for the range revealed that the prototype has resistance to interference after all, an unanticipated but advantageous side effect of the Code Division protocol. This is because each of the transmitted chips is resistant to interference unless a noise spike affects three or more bits of the chip while the transmitter light is off. The decoder corrects any lesser number of errors per chip and can detect one erroneous chip per frame, but it cannot detect or correct the case where two or more chips are erroneous. The error rate would be highly dependent on the relative brightness of the interfering light, the timing, and the pattern.

Other references elsewhere in the paper that mention VLC's ability to overcome interference is not based on actual testing but instead based on the known capabilities of the protocol in general and projections of what the system could do in the future with the suggested modifications.

Chapter 4 Future Work and Conclusion

4.1 Future Work

While this VLC prototype has been an exciting start, there is still room for further growth and development. The primary limitation on data rates in the VLC prototype was not in the analog frontend or even the digitization hardware, but instead the rate at which data could be processed in the decoder before it would start arriving faster than it could be decoded. The reason that the decoder had problems keeping up with the transmitter was because it needed to process the incoming data, recreate the clock signal or otherwise count and detect bits, and then output the data it had processed. The transmitter, meanwhile, was a state machine that strobed the headlamp in the appropriate pattern, but it did not require any large degree of processing to do so. Instead, it was mainly a sequence of bit operations and conditional statements.

Fortunately, this is not an insurmountable problem. The decoder software is highly parallel and would benefit greatly from implementation in programmable hardware like a Field-Programmable Gate Array (FPGA) or an ASIC, especially in the case of the Code Division system. An FPGA is a set of predefined logic elements that can be connected in arbitrary order to create custom digital logic circuits, albeit at a minor cost in speed due to the configurable connections' delay. FPGAs would be preferable to ASICs while changes to the prototype are still being made because the FPGA, while not quite as fast, would still significantly increase the speed and can be reprogrammed if there is a problem in the design. An ASIC, on the other hand,

must be designed and fabricated like any other microchip, which requires a great deal of lead time and expense even if the system were to work on the first try.

Another desired change would be the adjustment of the high-pass filter used for ambient light removal so that it removes low-frequency interference signals as well (like AC line noise). An example of this functionality is shown below in Figure 4.1, but it could not be fully implemented in the prototype because there was enough side-band attenuation on received signals that it reduced the range if installed. The higher speeds from an exchange of processors, and the resulting greater difference between the transmitter's frequency and the common noise sources, would be required to prevent the loss of range of the prototype when using this feature [20].
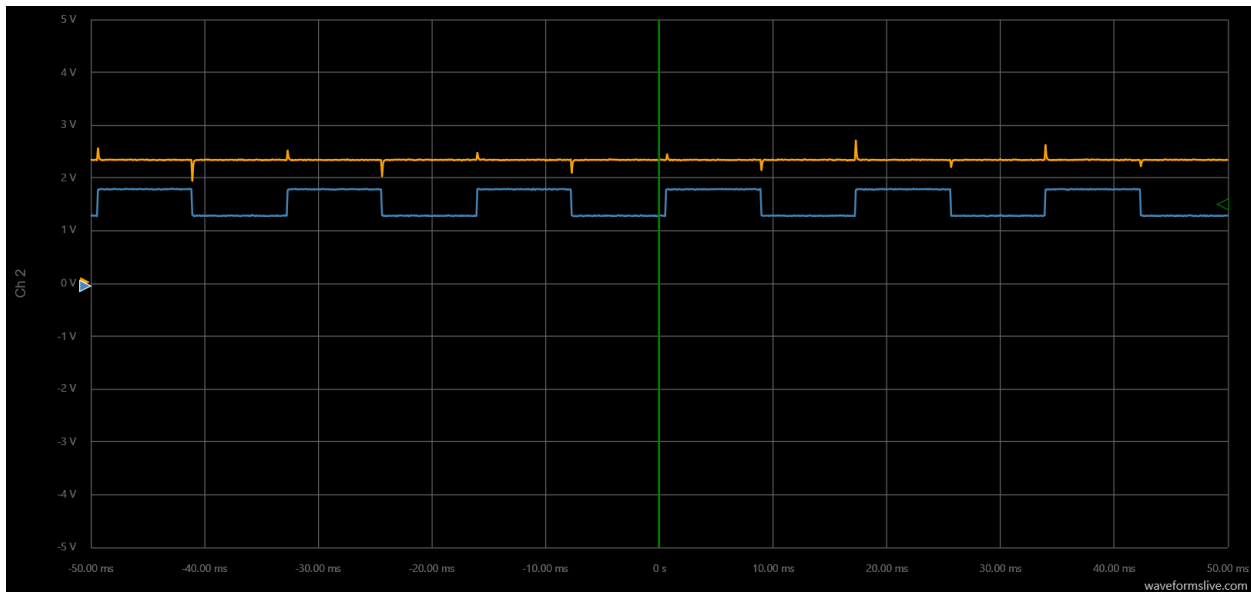
Figure 4.1 Low-frequency signals (blue line) also being absorbed by the filter (output orange)

The next step would be the incorporation of a phase-locked loop (PLL). The PLL can fine-tune its period and phase with that of an incoming signal, allowing the decoder to synchronize its signal sensing attempts with the emissions of the transmitter. This would allow the decoder to continually sample the signal over its entire period to get a better idea of the

51

transmitted value. The repeated sampling allows for greater accuracy, a very large range increase, and high noise immunity, which together would probably be the largest single improvement to the system's performance. While it would have the most effect for a Code Division system, it would still be useful for digital signals as well. For example, the PLL would have replaced the mechanism of ignoring erroneous level transitions that was used in the Manchester decoder. Unfortunately, due to hardware limitations of the receiver, as well as budget constraints stemming from the pandemic, it was decided not to add the feature, as it would have required a complete rebuild of the basic design to work with the additional analog circuitry and higher speeds used by most PLLs. Software workarounds were instead used in the decoder to increase its performance, but that is still not comparable to the potential results from a proper PLL-based design.

With higher noise resistance obtained from either a PLL or the filter change, it would also be possible to reclaim some bandwidth by increasing the frame size. The current implementation limits the length of the frames to reduce the number of bytes corrupted in any given error event before a new frame would force a resynchronization. With an improvement in accuracy, it should be possible to add more data to each frame while still observing a reduction in the number of erroneous data bytes. Code Division currently sends eight chips per ten sent, and Manchester sends four bytes per approximately six. Increasing their ratios, even a little, would allow for a proportionate increase in data rate.

Lastly, a white LED that uses phosphors is used in the transmitter. This currently is not an issue with the present speeds, but as the system transfers more data, it may become inadequate. At that point, it could be replaced by micro-LEDs. This new technology works by disrupting the alignment of the LEDs' native piezoelectric effect. With it disrupted, electrons and

holes formed by current flowing through the LED will recombine more quickly, resulting in even higher switching speeds as the LED can remain lit for less time after a current pulse. It also allows for a higher power output per LED [25], allowing for a potential increase in range for the same size of headlamp. These micro-LEDs are used for the highest-speed VLC designs in use today, and could also be used here, assuming that the receiver circuit is capable of operating on that sort of bandwidth. Of course, this would require that the vehicle implementing this technology use a special headlight instead of their current standard one, so the system would no longer be a drop-in module to add VLC capability, but it might be worth it if an automaker wants the greatest speed.

4.2 Conclusion

The VLC prototype proposed in this thesis was designed to be a compelling first step toward a competitive radio replacement for vehicular communication. It has a long range compared to most other VLC systems, especially outdoor units, at 70 feet, and its dual amplifier design and ambient light filtering are valuable subsystems that are needed to make any sort of long-range VLC system operational. The dual amplifier almost entirely protects the receiver circuitry from light saturation by moving much of the gain away from the vulnerable sensor circuit, and it contributes to the range by allowing for higher overall gain levels than otherwise possible. The ambient light filtering enables the receiver to ignore any signal inconsistencies caused by ambient light, which improves the performance of the system and removes the need for calibration to the environment. Neither of these features were fully implemented anywhere else at the time of this research, nor were they applied in the same manner.

Another innovative approach of this prototype is its use of Code Division, which has traditionally been used in radio applications but not in VLC. A possible reason for this is prior to the creation of this prototype, no known researchers were able to get a bipolar signal out of their receiver. The inclusion of this Code Division algorithm not only increases the range by reducing noise and improving the decoder's ability to detect the signal, but it also helps make the system more resistant to interference. This will position the prototype for great success if put into production, since full implementation will have many units transmitting simultaneously and it is important that they not interfere with each other.

In summary, the proposed VLC prototype provides a viable hardware design capable of excellent performance regardless of the ambient light level and, combined with proper protocol support and hardware decoding as described in the "Future Work" section, could be used to design a robust high-speed and long-range optical communications system for inter-vehicular use.

References

[1] H. Haas, L. Yin, Y. Wang, and C. Chen, "What is LiFi?," *J. Light. Technol.*, vol. 34, no. 6, pp. 1533–1544, Mar. 2016, doi: 10.1109/JLT.2015.2510021.

[2] S. Rehman, S. Ullah, P. Chong, S. Yongchareon, and D. Komosny, "Visible Light Communication: A System Perspective—Overview and Challenges," *Sensors*, vol. 19, no. 5, p. 1153, Mar. 2019, doi: 10.3390/s19051153.

[3] T.-H. Do and M. Yoo, "Potentialities and challenges of VLC based outdoor positioning," in *2015 International Conference on Information Networking (ICOIN)*, Cambodia, Jan. 2015, pp. 474–477. doi: 10.1109/ICOIN.2015.7057950.

[4] H. Haas, "LiFi is a paradigm-shifting 5G technology," *Rev. Phys.*, vol. 3, pp. 26–31, Nov. 2018, doi: 10.1016/j.revip.2017.10.001.

[5] S. Anthony, "Self-driving cars still can't mimic the most natural human behavior," *Quartz*. https://qz.com/1064004/self-driving-cars-still-cant-mimic-the-most-natural-human-behavior/ (accessed Jun. 09, 2021).

[6] R. Gilani, "Main Challenges in Image Classification," *Medium*, Jun. 14, 2020. https://towardsdatascience.com/main-challenges-in-image-classification-ba24dc78b558 (accessed Jun. 09, 2021).

[7] D. Heaven, "Why deep-learning AIs are so easy to fool," *Nature*, vol. 574, no. 7777, Art. no. 7777, Oct. 2019, doi: 10.1038/d41586-019-03013-5.

[8] H. Onishi, F. Mlinarsky, F. Watanabe, and C. Velasquez, "DSRC performance assessment for crash warning applications," in *2013 International Conference on Connected Vehicles and Expo (ICCVE)*, Las Vegas, NV, USA, Dec. 2013, pp. 769–774. doi: 10.1109/ICCVE.2013.6799893.

[9] "Performance analysis of DSRC priority mechanism for road safety applications in vehicular networks", doi: 10.1002/wcm.821.

[10] A.-M. Cailean, B. Cagneau, L. Chassagne, V. Popa, and M. Dimian, "A survey on the usage of DSRC and VLC in communication-based vehicle safety applications," in *2014 IEEE 21st Symposium on Communications and Vehicular Technology in the Benelux (SCVT)*, Delft, Netherlands, Nov. 2014, pp. 69–74. doi: 10.1109/SCVT.2014.7046710.

[11] H. Haas *et al.*, "Introduction to indoor networking concepts and challenges in LiFi," *J. Opt. Commun. Netw.*, vol. 12, no. 2, p. A190, Feb. 2020, doi: 10.1364/JOCN.12.00A190.

[12] R. N, S. B, and S. S, "A High Speed Open Access Visible Light Communication System based on Intensity Modulation," *Int. J. Sci. Technol. Eng.*, vol. 3, no. 08, p. 5, Feb. 2017.

[13] T. H. Talukdar and S. H. Parvez, "Small Scale Wireless Data Transmission via Light Using Light Source," p. 6, Jan. 2017.

[14] S. Chatterjee, V. Kothari, A. Kumar, M. Nair, and E. Lokesh, "POINT-TO-POINT DIGITAL COMMUNICATION," p. 3, Sep. 2019.

[15] "Shedding-Light-On-LiFi-webv2.pdf." PureLiFi, 2017. Accessed: Jun. 13, 2021. [Online]. Available: http://purelifi.com/wp-content/uploads/2017/06/Shedding-Light-On-LiFi-webv2.pdf

[16] D.-R. Kim, S.-H. Yang, H.-S. Kim, Y.-H. Son, and S.-K. Han, "Outdoor Visible Light Communication for inter- vehicle communication using Controller Area Network," in *2012 Fourth International Conference on Communications and Electronics (ICCE)*, Hue, Vietnam, Aug. 2012, pp. 31–34. doi: 10.1109/CCE.2012.6315865.

[17] A. P. Ardi, I. S. Aulia, R. A. Priramadhi, and D. Darlis, "VLC-Based Car-to-Car Communication," *J. Elektron. Dan Telekomun.*, vol. 20, no. 1, p. 16, Aug. 2020, doi: 10.14203/jet.v20.16-22.

[18] H. B. Eldeeb, E. Yanmaz, and M. Uysal, "MAC Layer Performance of Multi-Hop Vehicular VLC Networks with CSMA/CA," in *2020 12th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, Porto, Portugal, Jul. 2020, pp. 1–6. doi: 10.1109/CSNDSP49049.2020.9249593.

[19] C. B. Liu, B. Sadeghi, and E. W. Knightly, "Enabling vehicular visible light communication (V2LC) networks," in *Proceedings of the Eighth ACM international workshop on Vehicular inter-networking - VANET '11*, Las Vegas, Nevada, USA, 2011, p. 41. doi: 10.1145/2030698.2030705.

[20] B. Turan and S. Ucar, "Vehicular Visible Light Communications," in *Visible Light Communications*, J.-Y. Wang, Ed. InTech, 2017. doi: 10.5772/intechopen.69536.

[21] M. Elamassie, M. Karbalayghareh, F. Miramirkhani, R. C. Kizilirmak, and M. Uysal, "Effect of Fog and Rain on the Performance of Vehicular Visible Light Communications," in *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, Porto, Jun. 2018, pp. 1–6. doi: 10.1109/VTCSpring.2018.8417738.

[22] M. S. Islim *et al.*, "The Impact of Solar Irradiance on Visible Light Communications," *J. Light. Technol.*, vol. 36, no. 12, pp. 2376–2386, Jun. 2018, doi: 10.1109/JLT.2018.2813396.

[23] B. Razavi, *Principles of data conversion system design*. New York: IEEE Press, 1995. Accessed: Jun. 09, 2021. [Online]. Available: http://books.google.com/books?id=mKYoAQAAMAAJ

[24] S.-W. Wang *et al.*, "A high-performance blue filter for a white-led-based visible light communication system," *IEEE Wirel. Commun.*, vol. 22, no. 2, pp. 61–67, Apr. 2015, doi: 10.1109/MWC.2015.7096286.

[25] K. James Singh *et al.*, "Micro-LED as a Promising Candidate for High-Speed Visible Light Communication," *Appl. Sci.*, vol. 10, no. 20, p. 7384, Oct. 2020, doi: 10.3390/app10207384.

[26] "MTD5052N.pdf." Marktech Optoelectronics Inc., Mar. 29, 2021. Accessed: Jun. 11, 2021. [Online]. Available: https://marktechopto.com/pdf/products/DataSheet/MTD5052N.pdf

[27] "6912fa.pdf." Linear Technology. Accessed: Jun. 11, 2021. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/6912fa.pdf

[28] P. Stoffregen and R. Coon, "Teensy® 4.0." https://www.pjrc.com/store/teensy40.html (accessed Jun. 05, 2021).

[29] G. Povey, "An IEEE Standard for Visible Light Communications | Visible Light Communications," Aug. 29, 2013. https://web.archive.org/web/20130829162247/http://visiblelightcomm.com/an-ieee-standard-for-visible-light-communications/ (accessed Jun. 22, 2021).

[30] R. R. Sharma, A. Sanganal, and S. Pati, "Implementation of A Simple Li-Fi Based System," vol. 1, no. 9, p. 7, 2014.

[31] "Wired Communications – The basics – Programme Your Home," *Programme Your Home*. https://programozdazotthonod.hu/tech101/communication/the-basics/ (accessed Jun. 09, 2021).

[32] A. S. Tanenbaum, *Computer networks*, 4th ed. Upper Saddle River, NJ: Prentice Hall PTR, 2003.