

ABSTRACT

SIMPSON, JIM ANTO. A 1 Mbps Underwater Communications System using LEDs and Photodiodes with Signal Processing Capability. (Under the direction of John F. Muth.)

The inability of radio frequency electromagnetic waves to propagate without attenuation in seawater has traditionally limited underwater communications to acoustics or tethered systems. High bandwidth optical communication systems have been demonstrated for terrestrial and space applications. There is growing interest to see if short range high bandwidth optical wireless systems can be made for the underwater environment. In this thesis we demonstrate a 1 Mbps optical wireless system using LEDs and PIN photodiodes that also incorporates capabilities for signal processing of the received data to be performed.

Lasers and Photomultiplier tubes offer high performance, and are generally used in most underwater optical communication systems. However, these components are relatively expensive and can have large form factors. As an alternative solution the much cheaper and more compact LEDs and photodiodes are used as transmitters and receiver components. However, compared to a laser and PMT based system, such a system would be strongly disadvantaged in photon limited environments. If one assumes that photons actually reach the receiver, using signal processing techniques, optimized modulation formats, and error-correction coding, one expects that the range of the system can be extended. The development of a prototype system for the experimentation and verification of this proposition is the main motivation of this thesis.

Small, compact transmitters using High Power LEDs and receivers using Si Photodiodes where the data can be digitally sampled such that signal processing techniques can be applied were constructed and demonstrated using a 12 foot, 1200 gallon tank that was also constructed for the project. It was shown that the LED and photodiode based system works well for short ranges, and that advantages can be obtained using digital signal processing. The applicability of this strategy to use digital signal processing techniques can be easily extended to higher performance Laser/PMT based systems.

A 1 Mbps Underwater Communications System using LEDs and Photodiodes with Signal
Processing Capability

by

Jim Simpson

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the degree of
Master of Science

Electrical Engineering

Raleigh, North Carolina

2007

APPROVED BY:

Dr. Leda Lunardi

Dr. Brian Hughes

Dr. John F. Muth
(Chair of Advisory Committee)

DEDICATION

For:

My father, Simson Yohannan,
who is my biggest inspiration.

BIOGRAPHY

After attending his freshman year in Computer Engineering at UNC-Charlotte, Jim Simpson came to NC State University for the Electrical Engineering program in 2003.

While at NC State, from August 2004 till August 2005, he worked on a DARPA funded project to build an underwater mine deactivating submarine along with other founding members of the NC State Underwater Robotics Club (URC). Since then he has been a part of URC, working on building autonomous underwater competition submarines for the club.

From August 2004 till December 2006 he worked as an Undergraduate Research Assistant with the Bionics Group at NC State. There, he assisted in the development of a wideband wireless high density neural stimulation platform and then developed a 4-channel stimulator capable of performing different customizable stimulation strategies. Jim also developed a closed loop gastrointestinal pacemaker system in collaboration with University of Texas Medical Branch for the treatment of morbid obesity and a variety of motility disorders.

Although he was able to gain valuable experience in designing high-efficiency analog and digital circuits and system level development while in the Bionics Group, his true interests in robotics and passion to develop a viable underwater communication system for the Underwater Robotics Club submarine drew him to working on a Master's thesis with Dr. John Muth.

Jim graduated with a Bachelor's in Electrical Engineering in Spring 2006 from NC State University. Upon completing his Master's in Electrical Engineering, he looks forward to pursuing his Doctoral Degree in the same field.

ACKNOWLEDGEMENTS

I would like to thank:

First and foremost, God, my Heavenly Father, for his infinite blessings and love

Dr. Muth for his continuing support and guidance

Dr. Leda Lunardi and Dr. Brian Hughes for their guidance and wisdom

other students in the lab, especially

William Cox and Carlo Domizioli

my parents, Simson Yohannan and Rose Simson, for their love and support

my friends, especially

Mike and Sarah Faircloth

and Rhea Motashaw

TABLE OF CONTENTS

LIST OF FIGURES	vii
LIST OF TABLES	x
CHAPTER 1. INTRODUCTION.....	1
1.1. Applications.....	2
1.2. Component Considerations.....	3
1.3. Hardware vs. Software Signal Processing.....	5
1.4. Proposed System Architecture.....	7
1.5. References.....	8
CHAPTER 2. BACKGROUND INFORMATION.....	9
2.1. Challenges for Free Space Underwater Optical Communications	9
2.1.1. <i>Properties of Water</i>	9
2.1.2. <i>Absorption</i>	10
2.1.3. <i>Scattering</i>	14
2.1.4. <i>Refractive Index Dependencies</i>	16
2.1.5. <i>Underwater Link Budget</i>	17
2.2. References.....	19
CHAPTER 3. TRANSMITTER AND RECEIVER FRONT-END SETUP.....	20
3.1. Transmitter Introduction	20
3.1.1. <i>LEDs</i>	21
3.1.2. <i>LED Drivers</i>	23
3.1.3. <i>LED Secondary Optics</i>	24
3.2. Receiver Introduction.....	26
3.2.1. <i>Si PIN Photodiodes</i>	27
3.2.2. <i>Photodiode Pre-Amps</i>	29
3.2.3. <i>Photodiode Optics</i>	30
3.3. Channel Emulation Setup	32
3.4. References.....	33
CHAPTER 4. DATA ACQUISITION AND PC INTERFACE SETUP.....	34
4.1. Data Acquisition Device.....	34
4.1.1. <i>FPGA</i>	35
4.1.2. <i>Analog to Digital Converter with Variable Gain Amplifier</i>	37
4.2. PC Interface	40
4.2.1. <i>USB Interface</i>	41
4.2.2. <i>Streaming Implementation of USB</i>	42
4.3. PC Software	43
4.4. References.....	44
CHAPTER 5. EXPERIMENTS AND RESULTS.....	45
5.1. Underwater optical link.....	45
5.1.1. <i>Receiver Testing</i>	46
5.1.2. <i>Transmitter Testing</i>	47
5.1.3. <i>Receiver and Transmitter Testing</i>	48

5.1.4.	<i>Estimation of the channel attenuation using the Tx/Rx</i>	49
5.2.	LED Properties in water	50
5.2.1.	<i>Crude Beam Profiling</i>	50
5.3.	Attenuation Comparison	54
5.4.	System Improvements	57
5.5.	References	58
CHAPTER 6.	CONCLUSION	59
6.1.	References	62
APPENDIX.		63
	Schematics and Layout	64
	FPGA Verilog Code	73
	PC C++ Code	84

LIST OF FIGURES

Figure 1-1 Traditional implementation of Underwater FSO Link using Lasers and PMTs	1
Figure 1-2 Proposed implementation of low cost Underwater FSO Link	1
Figure 1-3 SNR vs. Optical Power for 2 kHz bandwidth [10].....	4
Figure 1-4 Proposed System Architecture	7
Figure 2-1 Absorption of pure seawater as a function of wavelength [4]	11
Figure 2-2 Absorption coefficient of phytoplankton [6].....	12
Figure 2-3 Spectral Absorption coefficient of CDOM [6].....	13
Figure 2-4 Absorption (β) and scattering (σ) of natural water samples [8].....	14
Figure 2-5 Scattering coefficients of large and small particles [7].....	15
Figure 2-6 Variations in Refractive Index for various parameters	16
Figure 2-7 Block Diagram showing parameters to consider for medium loss calculation [6]	17
Figure 3-1 Transmitter and Receiver Front-End Setup.....	20
Figure 3-2 Transmitter optics, source, driver, and interface set up at the transmit window...	21
Figure 3-3 Cree XR7090 Blue LED on PCB with.....	21
Figure 3-4 Luminous Flux for the different CREE LEDs	22
Figure 3-5 Radiant Flux for the different CREE LEDs	22
Figure 3-6 Absorption Coefficient Spectrum	22
Figure 3-7 Photosensitivity of the Hamamatsu photodiode.....	22
Figure 3-8 NMOS Switch LED Driver	23
Figure 3-9 LED Driver mounted on FPGA board	23
Figure 3-10 Cree XR Series LED Far Field Pattern	25
Figure 3-11 Mechanical drawing for Dialight OPTX1006.....	25
Figure 3-12 Dialight OPTX100 collimating lens for CREE LEDs	25
Figure 3-13 LED collimation assembly	26
Figure 3-14 Front View	26
Figure 3-15 Back View - With LED on PCB	26
Figure 3-16 Back View without LED on PCB	26
Figure 3-17 Receiver optics, pre-amp, batteries, digitizer and interface setup.....	27

Figure 3-18 Reverse Voltage [7].....	28
Figure 3-19 Spectral Response of the Hamamatsu S5973-02 Si PIN Photodiode	29
Figure 3-20 Basic Transimpedance Amplifier Circuit.....	29
Figure 3-21 Lens Assembly – front view	31
Figure 3-22 Photodiode Assembly – front view	31
Figure 3-23 Lens Assembly – back view.....	31
Figure 3-24 Photodiode Assembly – back view	31
Figure 3-25 Pre-Amp Board front view.....	31
Figure 3-26 Pre-Amp board back view.....	31
Figure 3-27 Water Tank with 2.5 ft. of water	32
Figure 4-1 Interfacing Front-End with a PC	34
Figure 4-2 Transmitter DAQ Block Diagram.....	36
Figure 4-3 Receiver DAQ Block Diagram	36
Figure 4-4 KNJN Development Board Xylo-EM.....	37
Figure 4-5 ADC with VGA.....	37
Figure 4-6 Gain vs. Gain Code for Varying Input Voltages.....	39
Figure 4-7 Typical performance specification for AD9283 (From Datasheet [4]).....	39
Figure 4-8 Measured performance characteristics of VGA + ADC + Interface.....	39
Figure 4-9 Power Spectrum of input source from the Arbitrary Waveform Generator.....	40
Figure 4-10 Double Buffering for the transmitter.....	42
Figure 4-11 Double Buffering for the receiver	42
Figure 4-12 Receiver Sampling Rate vs. Packet Loss	43
Figure 5-1 Link Testing Setup	45
Figure 5-2 Experiment setup for crude beam profiling	50
Figure 5-3 Measured luminance as a function of exposure for different Maalox amounts....	52
Figure 5-4 Drop in intensities at constant exposure for varying Maalox concentrations	52
Figure 5-5 CMOS Image Intensities of LED at 12 ft through water	52
Figure 5-6 CMOS Image Intensities of LED at 12 ft through water	53
Figure 5-7 Experiment Setup for Attenuation Comparison.....	54
Figure 5-8 Results for Attenuation Comparison.....	56
Figure 5-9 Receiver Revision 2	57

Figure 5-10 Transmitter Revision 2.....	57
Figure 5-11 Square pulse received by pre-amp	57
Figure 6-1 Example of advantages of DSP based detection over a level detector	60

LIST OF TABLES

Table 1-1 Lasers and PMTs vs. LEDs and Photodiodes.....	3
Table 1-2 Comparison between PMTs and Photodiodes.....	3
Table 1-3 Transmission Modulation Scheme vs. Receiver Signal Processing.....	5
Table 2-1 Summary of Attenuation underwater [3].....	10
Table 3-1 2SK0601 NMOS Specifications.....	23
Table 3-2 74LS125 Quad 3-State Buffer Specs.....	23
Table 3-3 TLC5922 IC Specifications.....	24
Table 3-4 Specifications for the S5973-02.....	29
Table 3-5 Specifications of the AD8015 Transimpedance Amplifier.....	30
Table 4-1 Altera Cyclone II EP2C5 FPGA Specifications.....	36
Table 4-2 AD9283 ADC Specifications.....	38
Table 4-3 AD8369 VGA Specifications.....	38
Table 4-4 PC Interface Bus Comparisons.....	40
Table 5-1 Test results for the receiver built.....	46
Table 5-2 Test results for the transmitter built.....	47
Table 5-3 Test results for the receiver and transmitter built.....	48
Table 5-4 Attenuation by water calculated from measured parameters.....	49
Table 5-5 Estimated Maalox concentrations and seawater turbidity.....	50
Table 5-6 Increasing exposure relationship to Luminance.....	51
Table 5-7 Results from Attenuation Comparison.....	55

CHAPTER 1. INTRODUCTION

In this thesis, the infrastructure with which to study underwater optical links is developed. The emphasis of the thesis is primarily on constructing small, compact transmitters and receivers where the data can be digitally sampled such that signal processing techniques can be applied. This potentially allows coding techniques that were developed for RF wireless to be applied to underwater optical systems. The use of signal processing techniques may also permit the use of lower cost components such as LEDs and Photodiodes rather than Lasers and PMTs for some applications. The ultimate goal of constructing this infrastructure is to be able to measure the performance of underwater communications links under a wide range of laboratory conditions that are similar to the water types found in the natural environment.

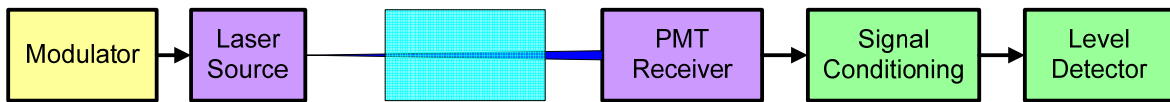


Figure 1-1 Traditional implementation of Underwater FSO Link using Lasers and PMTs

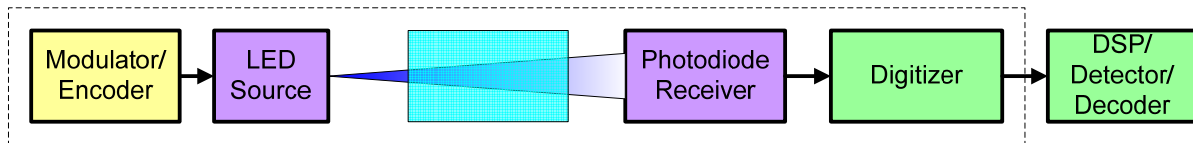


Figure 1-2 Proposed implementation of low cost Underwater FSO Link using LEDs and Photodiodes with Signal Processing capabilities (Dashed box indicates infrastructure built)

The motivation for this work is that high bandwidth free space optical (FSO) links have already been developed for terrestrial and space applications [1]. Comparatively, most of the advancements in communications underwater have been through acoustics [2]. However, as the amount of scientific research, industrial and defense work has increased underwater, alternatives to acoustic are starting to be investigated [3-5]. The principle advantage to underwater optical communication is the promise of much higher data rates than acoustics, at least for short range applications. It is an open question at this point as to what ranges one can expect underwater optical links to be viable.

1.1. Applications

Underwater communications are useful in a variety of fields. Underwater sensor networks, for example, are a network of sensors or underwater observatories that collect oceanographic data over an interval of time and send the data to the outside world periodically for processing [6]. Such sensors usually transmit significant amounts of data which require a high bandwidth link to the outside world [7]. In some cases, optical fibers along the sea floor or tethered connections to a floating buoy are appropriate, but these types of links are vulnerable to the environment and fix the sensor to one location. The bandwidth limitations of acoustics are problematic for distributed sensor networks, or underwater observatories. High bandwidth optical wireless links are one possible solution especially if coupled to autonomous underwater vehicles (AUVs).

Presently there is a proliferation of underwater and surface autonomous vehicles for a diverse number of applications including collecting environmental and oceanographic data, finding underwater mines, surveying underwater pipelines, and assisting in harbor security. [8]. Most AUVs are limited in size, and since they are typically battery powered, have very constrained power budgets. Optical communications are potentially very power efficient, especially in asymmetric systems where the power constrained system can be interrogated by a laser and the data returned to the interrogating source via a modulating retroreflector. However, the use of high bandwidth optical sources also has system level advantages when considering power constraints by reducing the amount of time that an AUV has to loiter near a sensor to collect the information. Station keeping, loitering or circling around a specific location is very expensive from a power budget perspective.

The same considerations are applicable for Remotely Operated Vehicles (ROVs) as well. ROVs, currently used more often than AUVs, send back in real-time, telemetry information, usually from bandwidth intensive sources like high-definition video cameras. Real-time navigational telecommands also need to be sent to the ROV. A tethered link can be dangerous in ship-wreck diving conditions and can also restrict an ROV to four or less degrees of freedom while most missions demand all six degrees of freedom that most ROVs can offer. An underwater FSO link can provide the efficiency (for power requirements), data-rate (for bandwidth requirements), and wireless (for untethered operation), that these sensor nodes, observatories, submarines, AUVs, and ROVs require.

1.2. Component Considerations

There are a variety of light sources and detectors to consider for free space optical transmitters and receivers. Lasers and Photomultiplier tubes offer high performance, and are expected to be used in many underwater optical communication systems. However, these components are relatively expensive and can have large form factors. An alternative solution would be to use the much cheaper LEDs as transmitters and photodiodes as receivers. Table 1-1 below [9] outlines the advantages and disadvantages of each of these alternatives.

Table 1-1 Lasers and PMTs vs. LEDs and Photodiodes

Parameter	Laser Diodes	LEDs
Maximum modulation	1 GHz and faster	100-300MHz
Optical Power (Radiant Flux)	Up to 100's of mW	Depends on LED materials and construction. In the 100's of mW for High Power LEDs.
Optical Bandwidth	<1 nm	40 to 100nm
Receiver Filtering	Narrow – lower noise floor	Wide – higher noise floor
Light Source	Coherent, self-interference	Incoherent, no self-interference
Minimum output beam divergence	Can be milliradians, ~0.01 degrees is reasonably obtainable.	~0.5 degrees
Lifetime	Medium lifetime with power level degradation with aging	Long lifetime
Temperature dependence	Varying temperature dependence	Little temperature dependence
Drive electronics	Temperature compensating output power control circuitry	Simple modulated current source
System Cost	High	Cheap

Table 1-2 Comparison between PMTs and Photodiodes

Parameter	Photomultiplier Tubes	Photodiodes
Area of detector	Large Up to 20 inches, but typically smaller, as small as 1 cm ²	Small
Volume	Large (cm scale)	Small mm scale
Power	High Voltage typically 1000 V	
Quantum efficiency	Depends on Wavelength, ~ 40%	High ~85%
SNR	High	Low when coupled with amplifier
Gain	High, photon counting capable	None
Bandwidth	Typically up to 5 MHz	Up to GHz for small diode.

By examining the table, one sees that photodiodes are strongly disadvantaged in photon limited, or high noise environments. If one assumes that photons actually reach the receiver, these problems could potentially be minimized using signal processing techniques like optimized modulation formats, error-correction coding, and other techniques instead of expensive hardware changes. Such processing techniques have been demonstrated to be effective for wireless and optical communications in air. Furthermore, these techniques can be applied to laser diode or other light sources. This can be supported by the illustration in Figure 1-3. The limiting parameter in performance differences between a PMT and a Photodiode is not sensitivity, but noise [10]. The PMT has a fairly good Signal-to-noise ratio (SNR) even at very low light powers, only limited by its shot noise. However the photodiode SNR completely drops out at 4.25pW. This is primarily due to amplifier noise from the Transimpedance amplifier which converts the photocurrent to a voltage and is primarily a function of the photodiode shunt resistance and terminal capacitance. However, it can be hypothesized that Bit Error Rates for a photodiode based receiver can be increased by the signal processing techniques mentioned above. The development of a prototype system for the experimentation and verification of this proposition is the main motivation of this thesis.

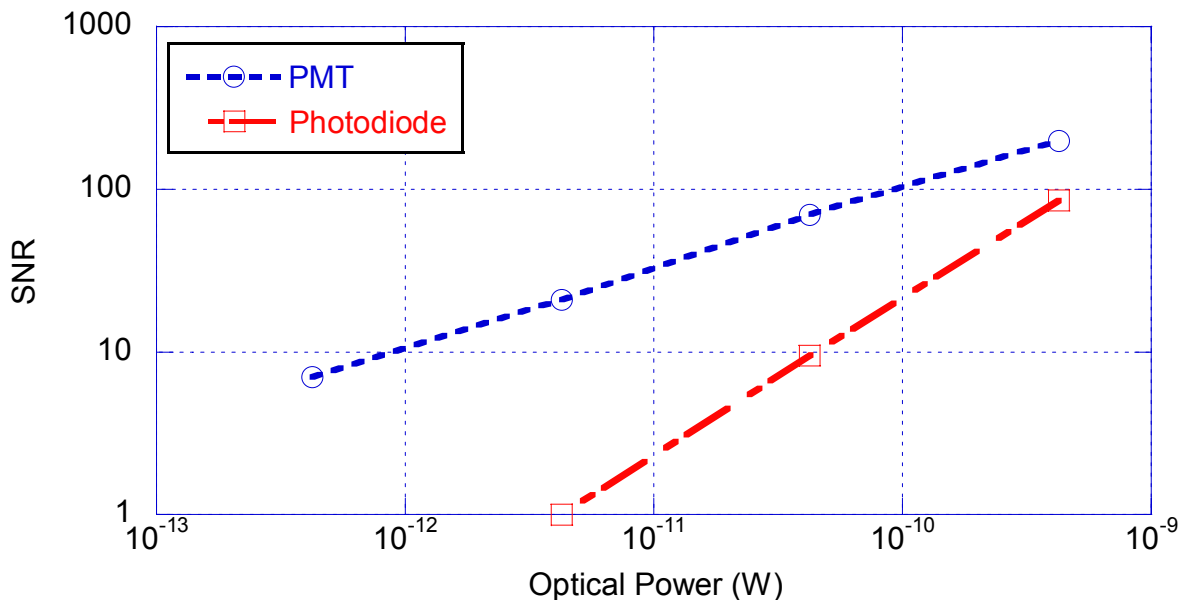


Figure 1-3 SNR vs. Optical Power for 2 kHz bandwidth [10]

1.3. Hardware vs. Software Signal Processing

Incorporating signal processing capabilities and the ability to add error-correction coding to the system requires a different approach to transmitting, as well as collecting and processing received signals. Table 1-3 on the next page indicates two differences in transmission modulations schemes in combination with two different ways of approaching detection of received signals.

Table 1-3 Transmission Modulation Scheme vs. Receiver Signal Processing

		Transmission Modulation Scheme	
		Analog (AM, FM, etc)	Digital (OOK)
Receiver Signal Processing	Hardware	<p><i>Pros:</i></p> <ul style="list-style-type: none"> Higher efficiency Can utilize existing interfaces to a PC – Plug and play <p><i>Cons:</i></p> <ul style="list-style-type: none"> Stricter requirements on receiver hardware 	<p>Traditional Implementation</p> <p><i>Pros:</i></p> <ul style="list-style-type: none"> Easy to Implement Can utilize existing interfaces to a PC – Plug and play. Eg. RONJA <p><i>Cons:</i></p> <ul style="list-style-type: none"> Limited control over performance – Harder to implement FEC encoder/decoder
	Software	<p><i>Pros:</i></p> <ul style="list-style-type: none"> Highest Efficiency Highly Customizable <p><i>Cons:</i></p> <ul style="list-style-type: none"> Hardest to implement Stricter requirements on receiver hardware Requires digitization of received signal Requires custom high bandwidth protocol over existing interface to PC 	<p>Proposed Implementation</p> <p><i>Pros:</i></p> <ul style="list-style-type: none"> Highly customizable Ideal environment for implementing, testing and debugging FEC Can provide buffering for FEC decoder easily <p><i>Cons:</i></p> <ul style="list-style-type: none"> Requires digitization of received signal Requires custom high bandwidth protocol over existing interface to PC

Considering the high effects of environment parameters on signal quality underwater, On-Off Keying (OOK) was the natural choice for modulating the transmitted signal. Although this simplifies the transmitter and receiver hardware, adding capabilities to digitize and process the received signals in software adds higher bandwidth requirements on the interface to the processor. A basic implementation of an 8-bit digitizer at 10 times oversampling requires that data-rate to be 80 times that of the transmitted data-rate. For a transmitted data-rate of 1Mbps, that equates to 80Mbps at the receiver.

One possible solution to this problem would be to implement the signal processing on an embedded processor or an FPGA with a parallel interface to the digitizer. However the ease of implementing coding schemes using pre-baked functions in PC tools like MATLAB or LabVIEW greatly facilitates development of the system compared to the non-intuitive embedded environments during the prototyping stages. Interfacing, tweaking, and debugging these algorithms during prototyping stages can also be done much more easily by implementing them on a PC. Thus the decision was made to keep the system as modular as possible, to exploit the advantages of PCs to provide good graphical user interfaces and to keep in mind that the ultimate goal is to transition the system to an embedded solution.

In summary, the advantages of digitization and signal processing in software incorporated onto the proposed system can be listed as:

- Capabilities to add Forward Error Correction (FEC) coding
- Easier to implement Equalization
- Can buffer and/or store (in memory) received data for processing in blocks
- Easier to test and debug different processing algorithms
- Can calculate BERs by storing transmitted bits in memory and cross-correlating it with received data
- Capabilities to normalize and apply Automatic Gain Control techniques
- Can discriminate bits for detection significantly better than a level detector
- Possibilities to run the same captured data through different processing techniques to compare differences without changing channel variables

1.4. Proposed System Architecture

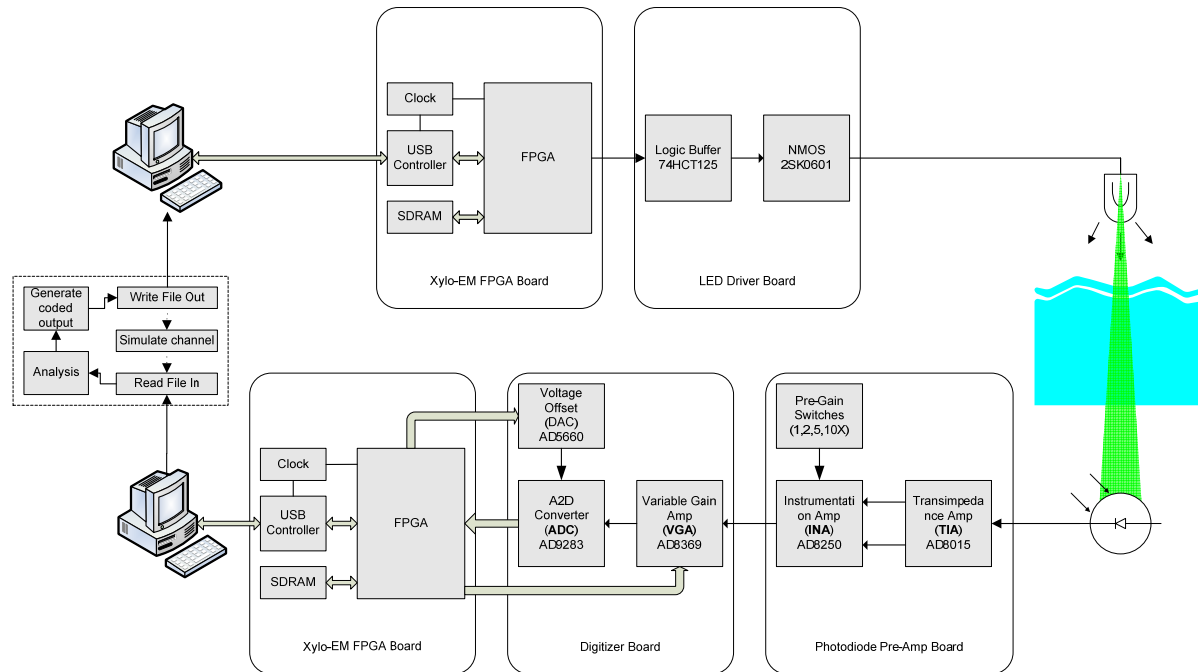


Figure 1-4 Proposed System Architecture

A block diagram of the system built up on these specs discussed is shown in Figure 1-4. As discussed earlier, LEDs and Photodiodes are chosen for light sources and detectors. Details about this are discussed in Chapter 3. Also described in Chapter 3 is a method of emulating an underwater channel through an indoor above ground water tank. Details about the data acquisition at the receiver and the interface to a PC using a Field Programmable Gate Array (FPGA) and Universal Serial Bus (USB) are given in Chapter 4. Also discussed is the interfacing and processing software on the PC. Experimental data obtained using the system is discussed in chapter 5. Schematics of the system and other documentation are included in the Appendices.

1.5. References

- [1] D. Killinger, "Free Space Optics for Laser Communication Through the Air," *Opt. Photon. News*, vol. 13, pp. 36-42, 10/01. 2002.
- [2] J. Partan, J. Kurose and B. N. Levine, "A survey of practical issues in underwater networks," in *WUWNet '06: Proceedings of the 1st ACM International Workshop on Underwater Networks*, 2006, pp. 17-24.
- [3] M. A. Chancey, "Short range underwater optical communication links," Masters Thesis, North Carolina State University, 2005.
- [4] (2002, June 15, 2002). Free-space optics (FSO) submerged. *Optics Report* Available: http://www.opticsreport.com/content/article.php?article_id=1015
- [5] N. Farr, A. D. Chave, L. Freitag, J. Preisig, S. N. White, D. Yoerger and F. Sonnichsen, "Optical modem technology for seafloor observatories," in *OCEANS 2006*, 2006, pp. 1-6.
- [6] J. Delaney, G. R. Heath, A. Chave, H. Kirkham, B. Howe, W. Wilcock, P. Beauchamp and A. Maffei, "NEPTUNE: Real-time, long-term ocean and earth studies at the scale of a tectonic plate," in *OCEANS, 2001. MTS/IEEE Conference and Exhibition*, 2001, pp. 1366-1373 vol.3.
- [7] S. M. Glenn, R. Chant, J. Kohut, J. Reinfelder, O. Schofield and J. McDonnell, "Opening a window to the sea: The potential of the ocean observatories for education," in *OCEANS 2006*, 2006, pp. 1-5.
- [8] M. D. Iwanowski, "Surveillance unmanned underwater vehicle," in *OCEANS '94. 'Oceans Engineering for Today's Technology and Tomorrow's Preservation.'* *Proceedings*, 1994, pp. I/116-I/119 vol.1.
- [9] C. Moursund. 2006, LEDs vs. laser diodes for wireless optical communication. Available: http://clearmesh.com/downloads/wp_led_0407.pdf
- [10] K. Kaufmann, "Choosing Your Detector," *OE Magazine*

CHAPTER 2. BACKGROUND INFORMATION

2.1. Challenges for Free Space Underwater Optical Communications

The biggest challenges to free space underwater optical communications originate from the fundamental characteristics of ocean water. Light sources and detector technologies are relatively well developed on the component level, but there is little information available concerning the use of ocean water as an optical communication medium. A successful system for making underwater free space optical communication will require a thorough understanding of, and be based on, the physical optics of ocean water. Ocean water is a complex physiochemical biological system. It contains dissolved substances, suspensions, and a multitude of different living organisms. Its optical properties depend upon its composition and on its physical state (temperature, pressure, etc.) [1]. These properties have been studied for a wide range of other applications by oceanographers and marine biologists. A few of the most relevant optical properties like absorption and scattering are discussed. A brief overview of a possible free space underwater optical link budget is also presented.

2.1.1. Properties of Water

The optical properties of water can be classified broadly as inherent optical properties (IOPs) and apparent optical properties (AOPs). Inherent optical properties are independent of the properties of the light source and are a function only of the material properties of the medium. Apparent optical properties are those derived properties which depend on both the geometrical structure (diffuse or collimated) of the light field and the inherent optical properties of the medium [2].

Inherent optical properties include the attenuation of light underwater through absorption and scattering, while the apparent optical properties include radiometric quantities such as irradiance and reflectance ratios [2]. For purposes of underwater communication, the former is more important and more relevant than the latter.

The main source of attenuation underwater is the spectral absorption coefficient and the spectral scattering coefficient. The contributions by these on water and its different constituents are outlined in the table below.

Table 2-1 Summary of Attenuation underwater [3]

	Absorption		Scattering	
	Characteristic	λ dependence	Characteristic	λ dependence
Water	Invariant at constant temperature and pressure	Strong	Invariant, Small compared to absorption	λ^{-4}
Sea Salts (Inorganic)	Negligible in the visible, Weak in the ultraviolet	Some increase towards short λ	Appreciable	None
Colored Dissolved Organic Matter	Variable	Increase towards short λ	None	None
Particulate Matter	Variable	Increase towards short λ	Variable	Variable

2.1.2. Absorption

The absorption coefficient of a given medium of water contributes towards the attenuation of a free space optical signal sent through the medium. The origin of the attenuation can be generally divided into inorganic and organic causes, and into categories where the light is truly absorbed and those where the photon is redirected or scattered. For seawater the intrinsic absorption from inorganic materials are the:

- Absorption of the water molecule,
- Absorption of dissolved salts.

Absorption that arises from organic substances include:

- Absorption by chlorophyll in phytoplankton
- Absorption by Colored Dissolved Organic Matter (CDOM).

Both inorganic and organics can scatter light and will be discussed later.

Absorption by pure seawater

Absorption of pure seawater can be considered as the sum of absorption of optically pure water (freshwater devoid of dissolved and suspended particulate matter) and the absorption by the salts present in pure saltwater. The latter can be assumed to be negligible in the visible spectrum [4].

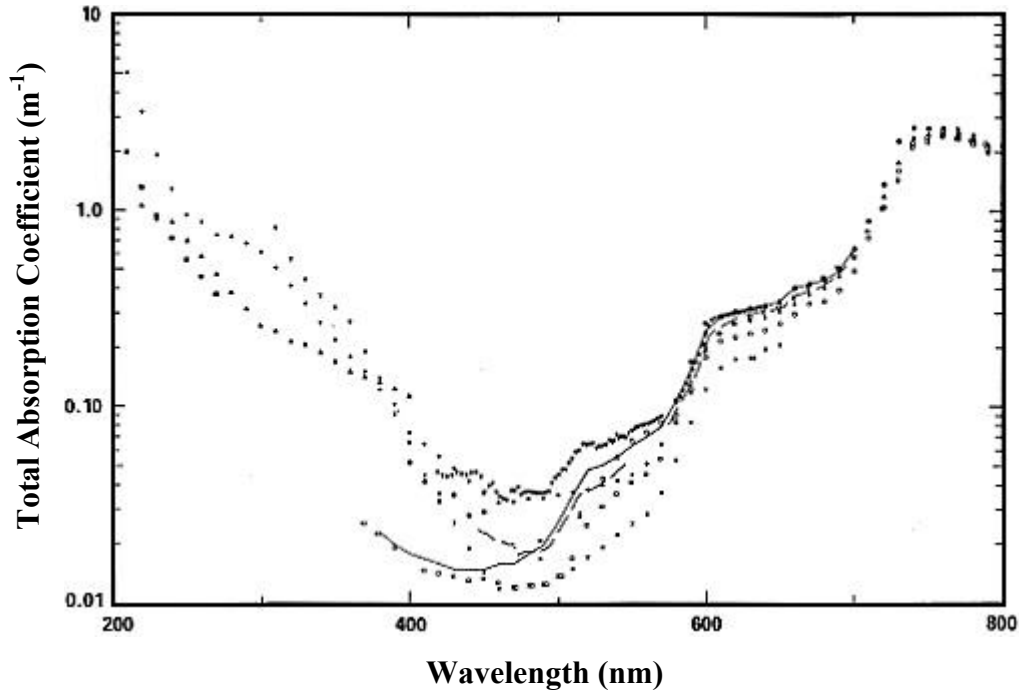


Figure 2-1 Absorption of pure seawater as a function of wavelength as given by various authors [4]

The previous figure from [4] combines the experimental results obtained by numerous authors. The overlap of the data shows a strong window in the blue-green region of the spectrum where pure seawater is least absorptive. An increase in absorption with increasing wavelength is due to the properties of pure water and an increase in absorption with decreasing wavelength is due to dissolved substances and particles in it [1]. Hence, as turbidity increases, the minimum of the blue-green window is shifted from blue to green or even green-yellow.

Absorption by Phytoplankton

The most important cause of absorption of light underwater is small microscopic plants called phytoplankton. Phytoplankton particles contain a large number of different colored substances. These are Chlorophyll a, b, and c, carotenoids, phaeophytins, chlorophyllides and phaeophorbides, etc [1].

However, studies have pointed out chlorophyll-a as the prominent pigment behind phytoplankton absorption [5]. This is due to the fact that the absorption spectra of different pigments are close to each other and the total concentration of pigments is closely correlated with the content of chlorophyll a. As indicated in Figure 2, chlorophyll-a absorbs more of the blue and red light leaving green as the least absorbed in the visible spectrum.

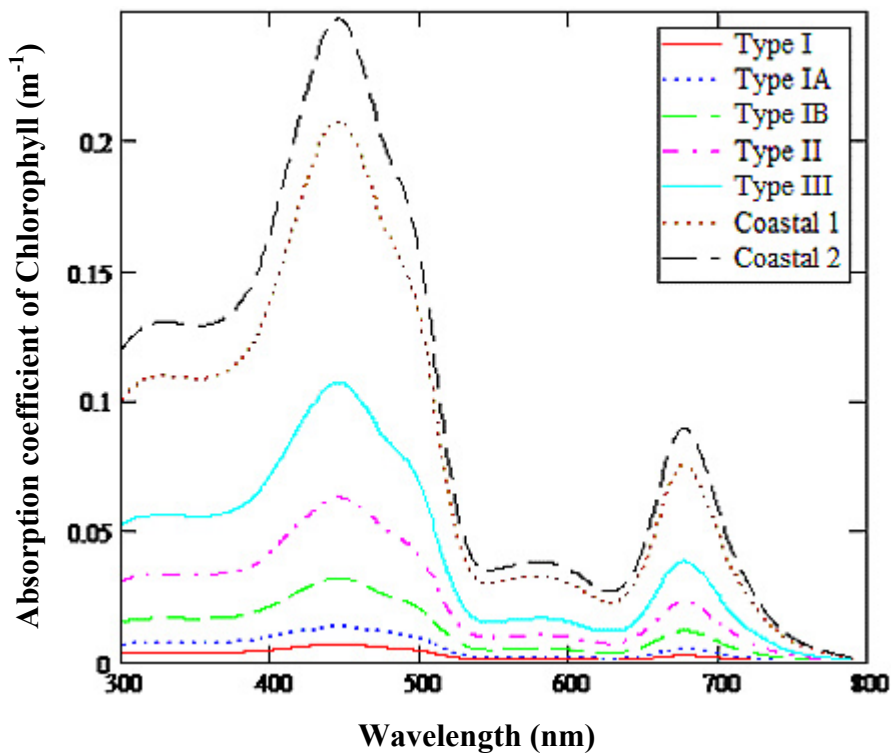


Figure 2-2 Absorption coefficient of phytoplankton shown for different Jerlov water types [6]

Absorption by Color Dissolved Organic Matter (CDOM)

CDOM, consisting of two components, humic and fulvic acid, are formed from the decomposition of chlorophyll-a. Both components absorb blue light predominantly, leaving it looking yellow [5]. Figure 3 below illustrates the exponential relationship with wavelength as well as a function of water type [6].

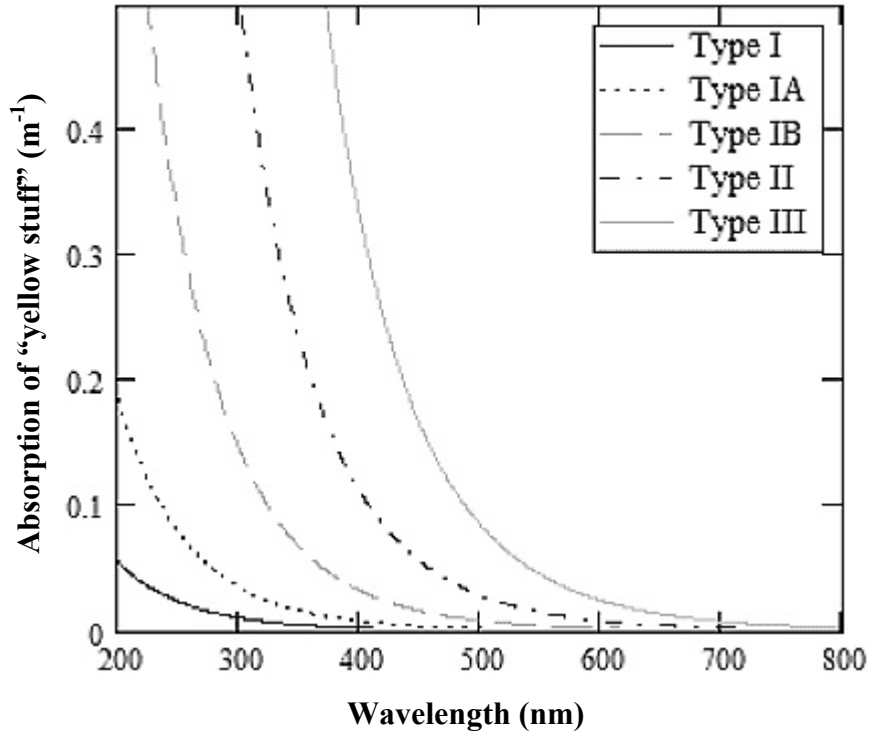


Figure 2-3 Spectral Absorption coefficient of CDOM [6]

It can be noticed in the above figure that the absorption spectrum is highly variable and can be attributed to the strong fluctuation in the amount of yellow substance in different water types.

The second noticeable feature is the sharp increase in absorption at shorter wavelengths. This region of the spectrum is related to the electron absorption spectrum. The absorption of yellow substance which we observe in the ultraviolet region is particularly related to the long-wave part of the electronic absorption band. It can be shown that the absorption coefficient, hence, increases with frequency in complex molecules [1].

2.1.3. Scattering

Scattering of light underwater is the deviation of photons from its true path upon interaction with particulate matter in water. Similar to free space air, photons experience Rayleigh scattering from interactions with particles smaller in size than the wavelength of light and Mie scattering from interactions with particles that are larger than the wavelength of light.

Similar to absorbance, scattering can be considered as the result of the inherent scattering in pure seawater along with the scattering due to particulate matter suspended in it.

Scattering by pure seawater

Scattering by pure water can be attributed to the salt ions present in pure seawater, which is more effective at shorter wavelengths [7]. Pure distilled water, absent of salts, has also been studied and shown to have scattering effects as illustrated in Figure 4 by [8].

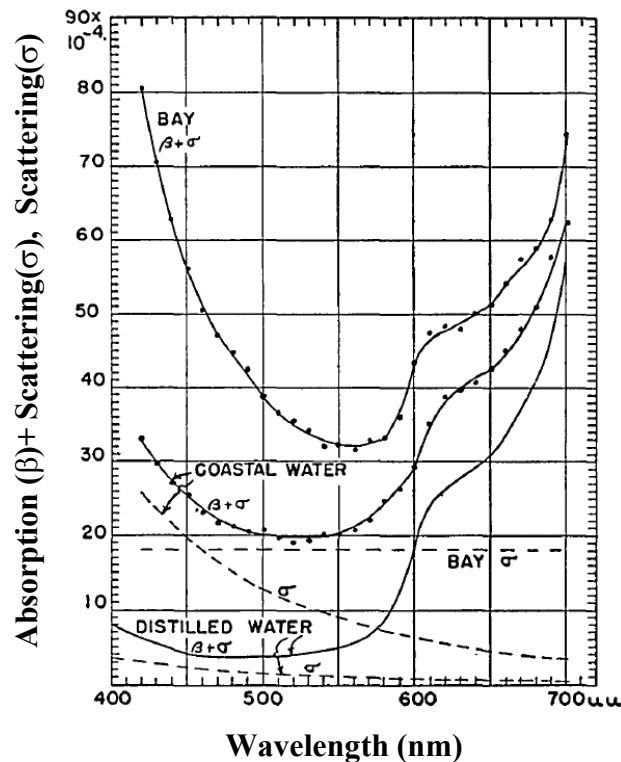


Figure 2-4 Absorption (β) and scattering (σ) of natural water samples [8]

Scattering by Particulate Matter

Suspended particulate matter comprises of a variety of both organic and inorganic particles. Organic particles include phytoplankton and other Color Dissolved Organic Matter (CDOM) while inorganic particles include mostly terrigenous particles and are found closer to coastal waters and continental shelves [6].

Unlike absorption, scattering by particles occurs without a change in energy, but results in a change in direction of propagation [5]. The angle of change is determined by the size of the particles and can be explained using the Rayleigh and Mie mechanisms. A variety of models have been developed to emulate both phenomenon and the results of the one-parameter chlorophyll model [5] are shown in figure 5.

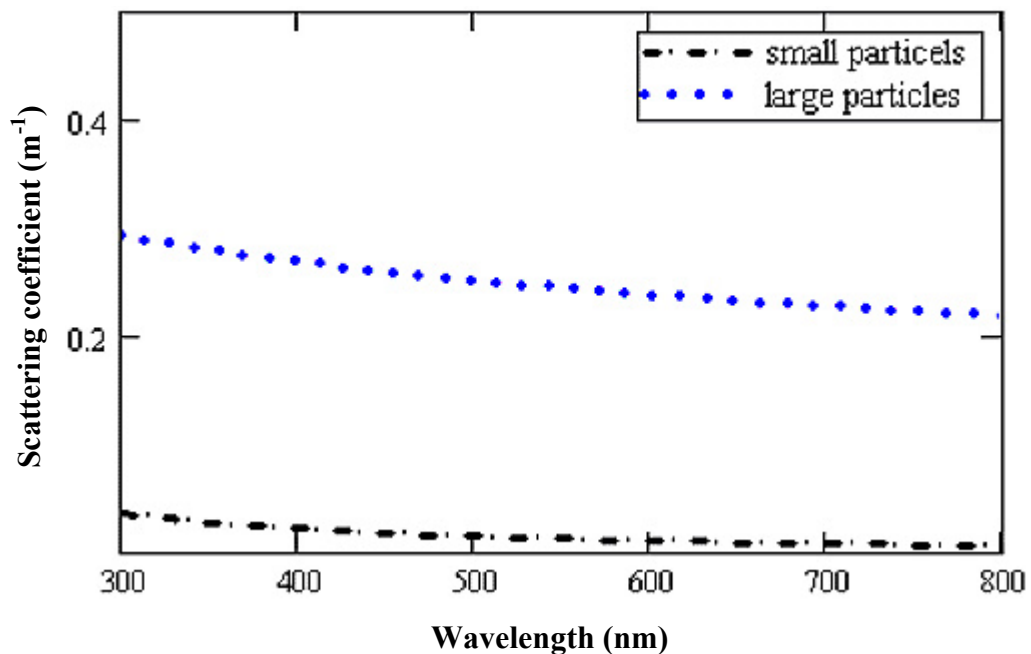


Figure 2-5 Scattering coefficients of large and small particles as a function of wavelength [7]

The data in the figure above shows that the spectral scattering coefficient weakly depends on the wavelength, slowly decreasing as wavelength increases. Lowest values for scattering coefficient were measured at depths more than 1000m to be about 0.022m^{-1} and values as high as 2.7 to 3.3m^{-1} were measured in coastal waters [1].

2.1.4. Refractive Index Dependencies

The refractive index of sea water is a function of several parameters, particularly salinity, temperature, and pressure. The refractive index is also dependent on the wavelength of the light [9].

Empirical equations have been developed [10] and used to calculate the variations in refractive index as a function of wavelength, temperature, salinity and pressure. The results are shown below in Figure 6. The non-variable parameters were set at 35‰ for salinity, 20° for temperature, and 500nm for wavelength.

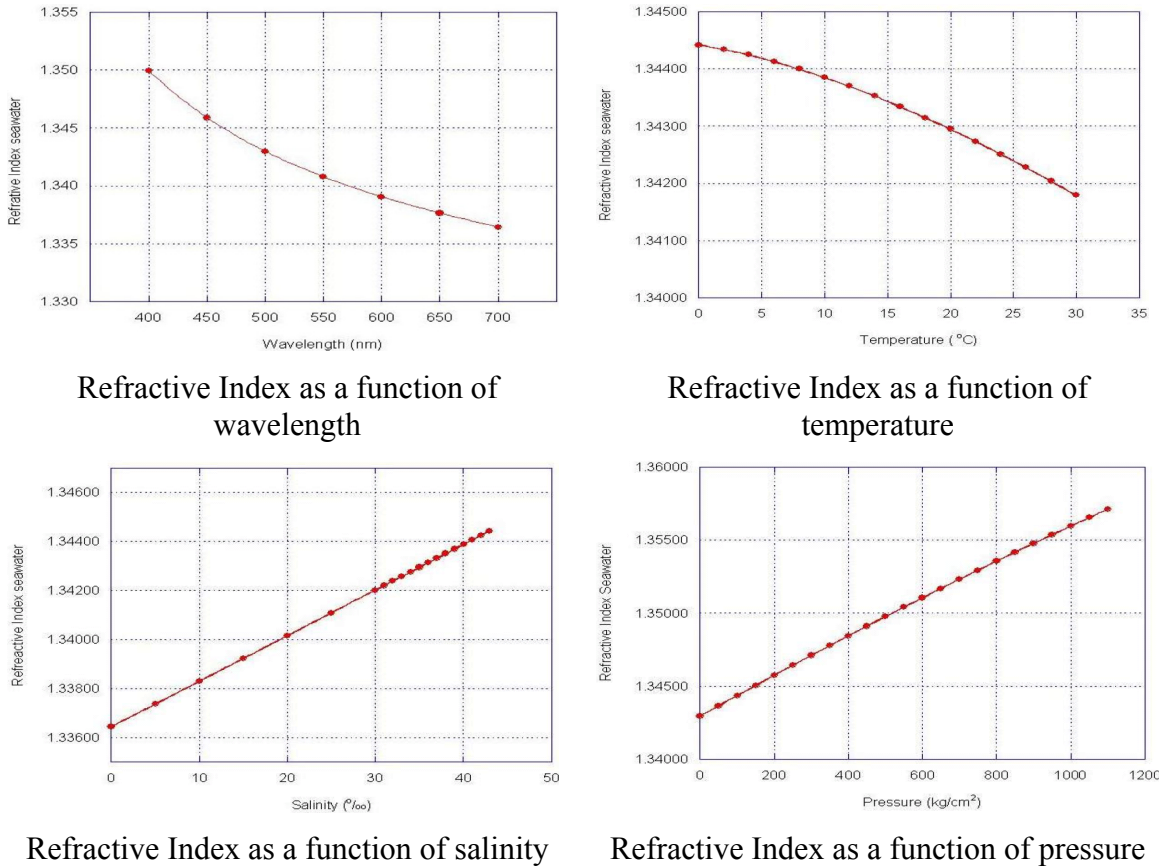


Figure 2-6 Variations in Refractive Index for various parameters

While these effects are marginal compared to the adverse attenuation effects of absorption and scattering, they could be prominent for longer distances, pointing and tracking, as well as orientation of optical path [6].

2.1.5. Underwater Link Budget

A link budget for power gives the total received power (in dB) as a difference between the transmitted power (in dB) and sum of all the losses (in dB). The typical free space optical link equation can be written as follows [6]:

$$P_{received} = P_{transmitted} \cdot Gain_{transmitter} \cdot LOSS_{transmitter} \cdot LOSS_{Free_Space} \cdot Gain_{receiver} \cdot LOSS_{receiver} \quad (1)$$

This equation can be extended to underwater medium by calculating the $LOSS_{Free_Space}$ for seawater as the medium. The block diagram below summarizes the parameters to be considered in calculating this.

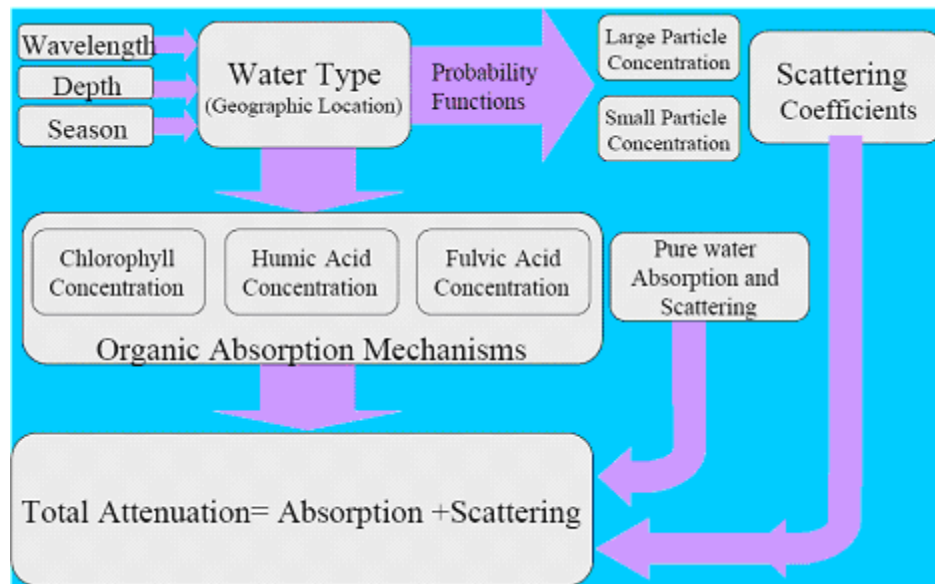


Figure 2-7 Block Diagram showing parameters to consider for medium loss calculation [6]

For the rest of the link equation, the following block diagram summarizes the variables required as inputs in calculating the link budget.

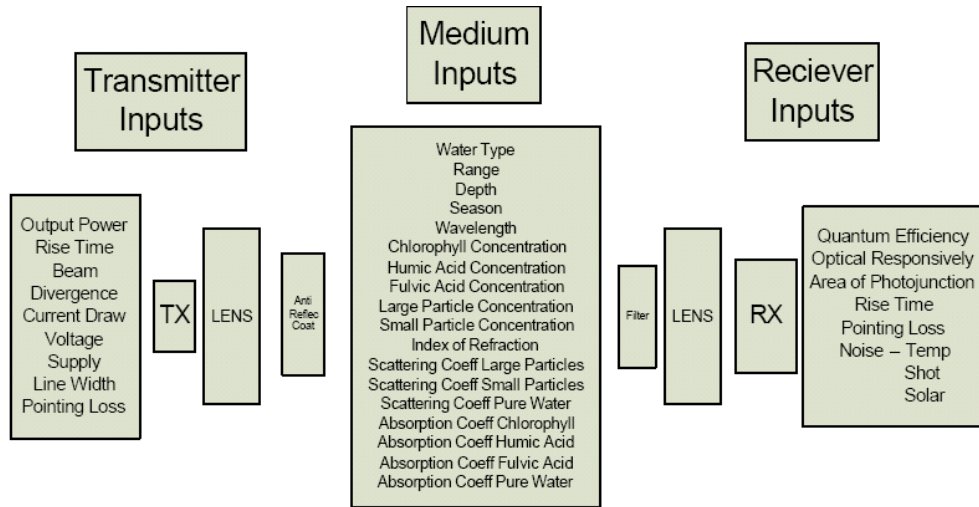


Figure 8 Block diagram showing parameters to consider for link budget calculation [6]

2.2. References

- [1] Shifrin, *Physical Optics of Ocean Water*. 1988,
- [2] M. A. Blizard, "Ocean optics: Introduction and overview," in *Ocean Optics VIII*, 1986, pp. 2.
- [3] N. G. Jerlov, *Marine Optics*. , 2nd ed. ed. Amsterdam: Elsevier Scientific Publ. Comp., 1976.
- [4] Smith. 1981, Optical properties of the clearest natural waters (200-800 nm). *Applied optics* 20(2), pp. 177.
- [5] Haltrin. 1999, Chlorophyll-based model of seawater optical properties. *Applied optics* 38(33), pp. 6826.
- [6] M. A. Chancey, "Short range underwater optical communication links," Masters Thesis, North Carolina State University, 2005.
- [7] Y. Gawdi, "Underwater free space optics," Masters Thesis, North Carolina State University, 2005.
- [8] Hulburt. 1945, Optics of distilled and natural water. *Journal of the Optical Society of America. B, Optical physics* 35(11), pp. 698.
- [9] P. Halley et al., "Optics of the sea (interface and in-water transmission and imaging)," National Technical Information Service, Springfield, Va., 1973.
- [10] R. W. Austin and G. Halikas, "The index of refraction of seawater," Scripps Institution of Oceanography, Tech. Rep. Tech. Rep. No. SIO Ref. 76-1, 1976.

CHAPTER 3. TRANSMITTER AND RECEIVER FRONT-END SETUP

The transmitter and receiver front-end setup for experiments consists of a High Power LED source and its driver on the transmitter side and a Si photodiode and its pre-amp on the receiver side. Each of the blocks in this setup, as illustrated by Figure 3-1 is explained in this chapter.

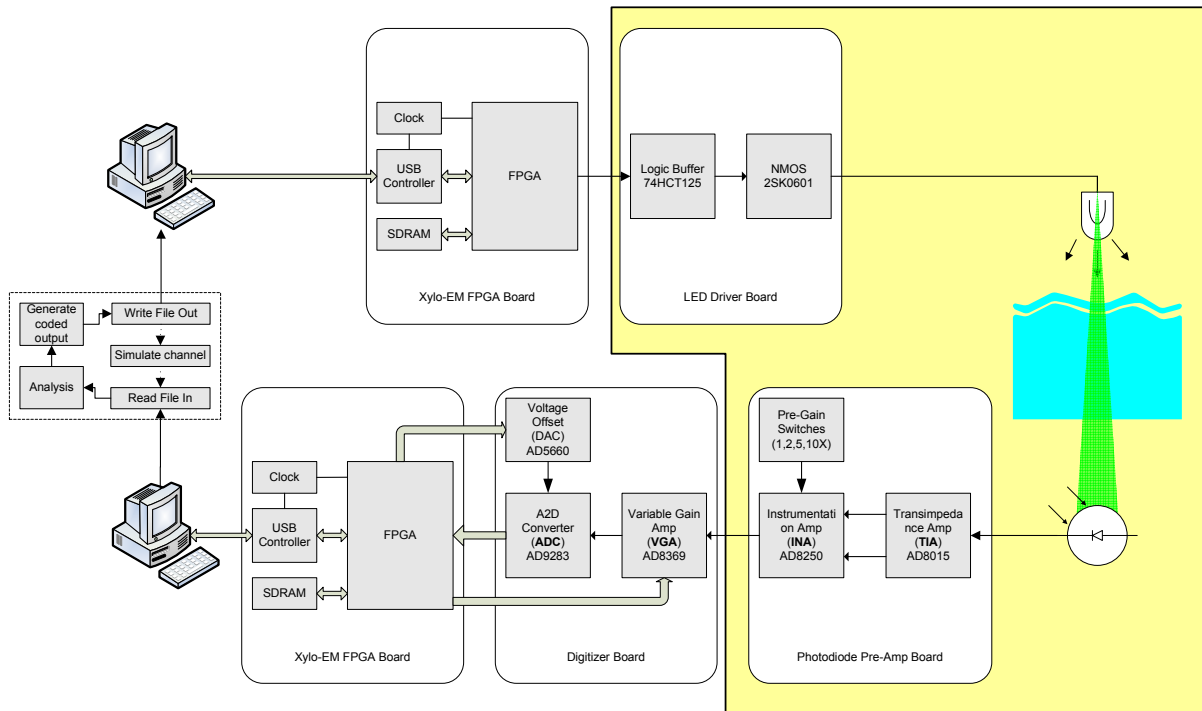


Figure 3-1 Transmitter and Receiver Front-End Setup

3.1. Transmitter Introduction

As mentioned previously, in switching from a traditional laser-source and PMT-receiver setup to a more cost effective LED-source and photodiode-receiver setup, careful attention has to be given to the selection of LEDs and their use to compensate for the loss in performance as compared to lasers diodes.

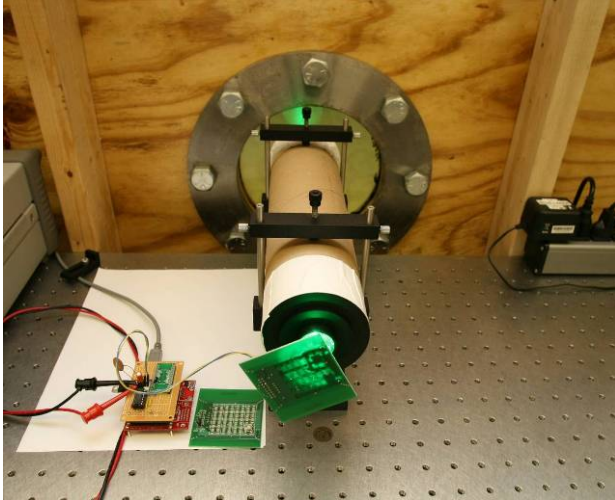


Figure 3-2 Transmitter optics, source, driver, and interface set up at the transmit window

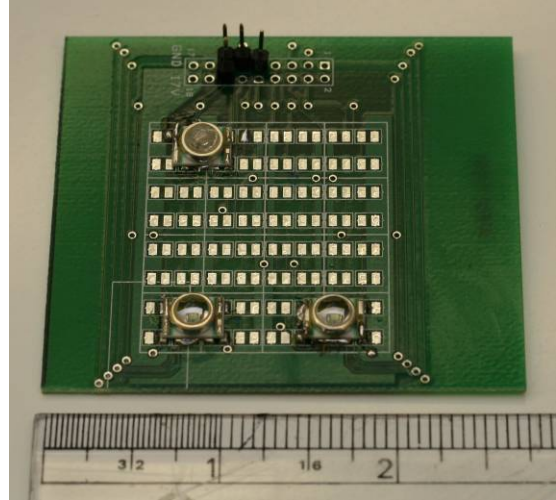


Figure 3-3 Cree XR7090 Blue LED on PCB with Green and Royal Blue sharing the same heat sink plane

3.1.1. LEDs

The LED (Light Emitting Diode) is a semiconductor device that produces a relatively narrow spectrum light, dependent on the material used with a particular brightness dependent on the forward bias current applied. The speed at which an LED can be modulated is usually limited by the die size for high brightness LEDs. This implies a trade off between power and speed, since larger die size provides higher brightness. For example, High Power LEDs (>1 Watt) were introduced in 2002 by Lumileds and were evaluated as a potential light source for modulation [1]. However, it was concluded that the large capacitance, due to the large die sizes in High Power LEDs, resulted in very low bandwidth for modulation purposes. This suggested that arrays of smaller LEDs would need to be used. More recently, the Cree XR Series High Power LEDs have been examined and found to work at data rates of at least as high as 2Mbps. Information from the datasheet [2] was used to calculate the radiant flux for all seven colors available in the series. Here, we also see the tradeoff between color and intensity in the blue/green region that is due to material science issues with the Indium Gallium Nitride that is used as the light emitting material. The Amber and Red LEDs are made from GaP and AlGaAs compounds, respectively.

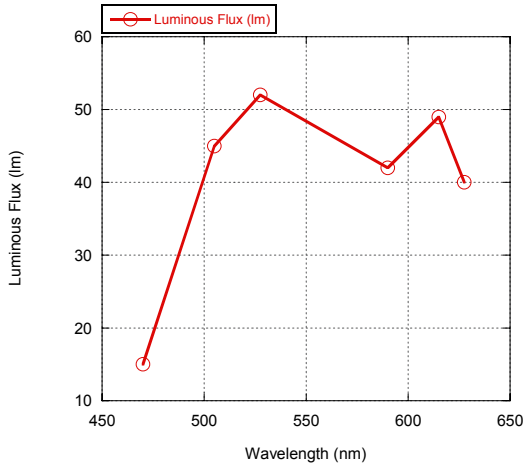


Figure 3-4 Luminous Flux for the different CREE LEDs

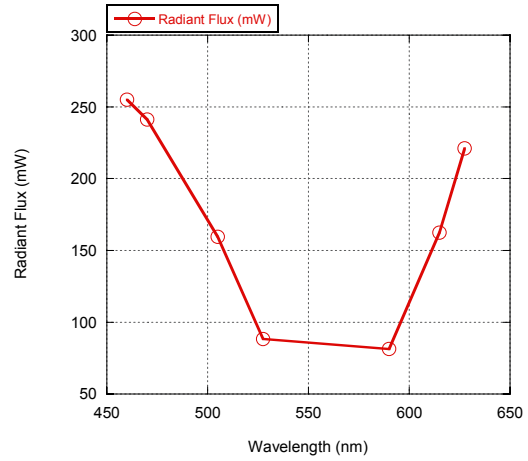


Figure 3-5 Radiant Flux for the different CREE LEDs

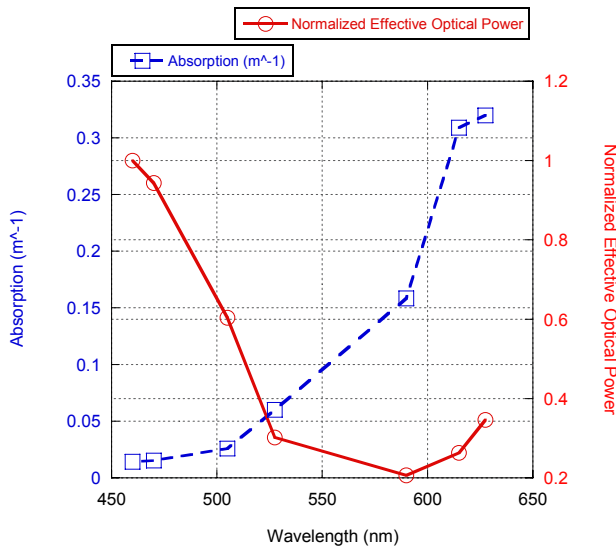


Figure 3-6 Absorption Coefficient Spectrum (blue trace) of clear ocean water [1] and spectrum multiplied with the corresponding radiant flux for each LED after normalization (red trace)

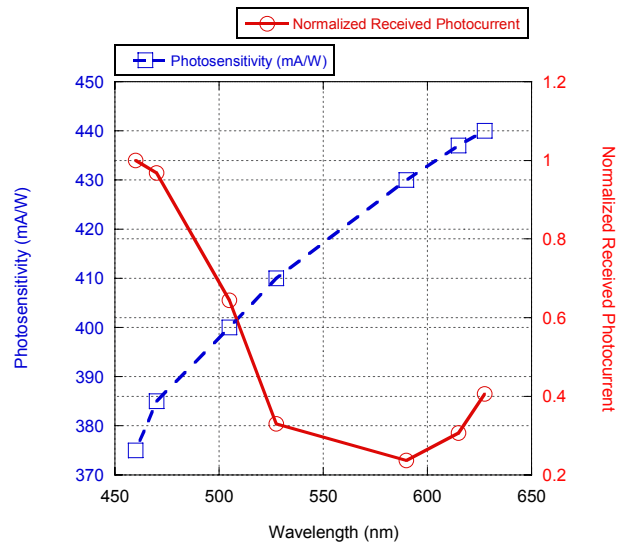


Figure 3-7 Photosensitivity of the Hamamatsu photodiode (blue trace) and photosensitivities multiplied with the normalized effective optical power from Figure 3-6 (red trace)

The Royal Blue and Blue LEDs from the XR series fall in the blue-green wavelength range permissible for transmission in water. They also have relatively high radiant flux values. Both these LEDs are made from InGaN material on a SiC crystal.

From Figure 3-4 through Figure 3-7, it is shown by combining the absorption spectrum and the LED power spectrum that Blue (470nm) or even Cyan (505nm) is most efficient as a light source underwater.

3.1.2. LED Drivers

Two different methods are investigated for driving the Cree High Power LEDs mentioned in the previous section. First, a simple FET based system is constructed to drive the LEDs 1Mbps. A commercial LED driver IC from Texas Instruments is then evaluated and its benefits over a FET driver listed.

FET Based LED Driver

A simple FET is capable enough to drive the XR LEDs for rates as high as 10Mbps. A 2SK0601 NMOS [3] driven by a 74LS125 Quad 3-State Buffer [4] was used to achieve this. The specifications for each of those ICs are given in Table 3-1 and Table 3-2.

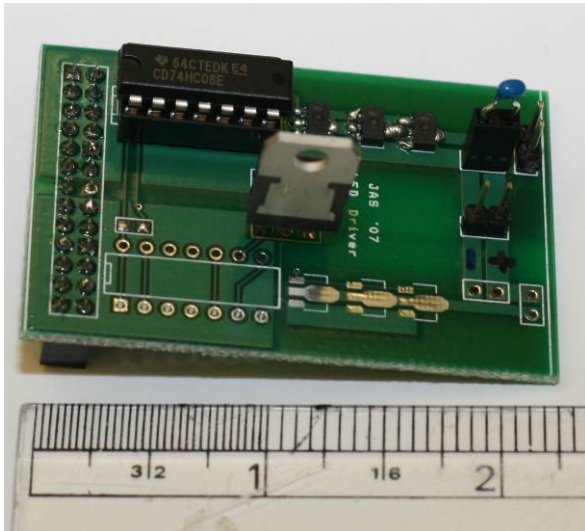


Figure 3-8 NMOS Switch LED Driver

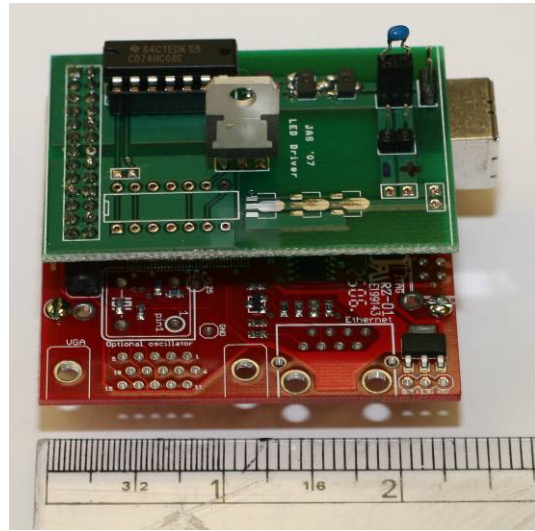


Figure 3-9 LED Driver mounted on FPGA board

Table 3-1 2SK0601 NMOS Specifications

Turn on time	15ns
Turn off time	20ns
Gate Threshold Voltage	>1.5V
Drain-source Voltage, V_{DS}	80V Max
Gate-source Voltage, V_{GS}	20V Max
Drain Current, I_D	0.5A Max
Peak Drain Current, I_D	1A Max

Table 3-2 74LS125 Quad 3-State Buffer Specs

Rise time	15ns
Fall time	22ns
High level voltage	3.4V
Low level voltage	0.34V

LED Driver IC (TI TLC5922)

An alternative to using FETs to drive LEDs is to use a dedicated LED Driver IC like the TLC5922 from Texas Instruments. The TLC5922 [5] has 16 output channels, all of which can be connected in parallel to generate constant continuous output currents as high as 1.28A. All outputs can be turned on or off simultaneously over a 1-bit interface, which can be modulated as high as 30 MHz. Furthermore, an additional serial interface allows each output to be individually turned on or off. This allows the transmitter to control the output current in real-time in 16 discrete steps. Specifications for the chip are listed in Table 3-3.

Table 3-3 TLC5922 IC Specifications

Rise Time	10ns
Fall Time	10ns
Number of output channels	16
Constant Current per channel	80mA
Total parallel current output	1.28A
LED Supply Voltage	0-17V
Data Interface	30MHz Max

The FET based driver should be enough for most applications. The commercial LED Driver IC would be helpful when a feedback loop is implemented and used to adjust the transmitter intensity real-time during operation.

3.1.3. LED Secondary Optics

The CREE XR Series High Power LEDs include an integrated lens on the packaging that puts the Full Width Half Maximum (FWHM) viewing angle at 100 degrees. A secondary lens designed specifically for this model of LEDs, the Dialight OPTX1006 [6], shown in Figure 3-11 is placed directly above the LED to cut the viewing angle to a half divergence of 6 degrees.

300mm away from the secondary lens is a telescopic lens assembly which collimates the light further to a spot size of ~2 inches at the other end of the tank (at a distance of 12 ft.). An image of this whole assembly with the LED mounted on a PCB is shown in Figure 3-13.

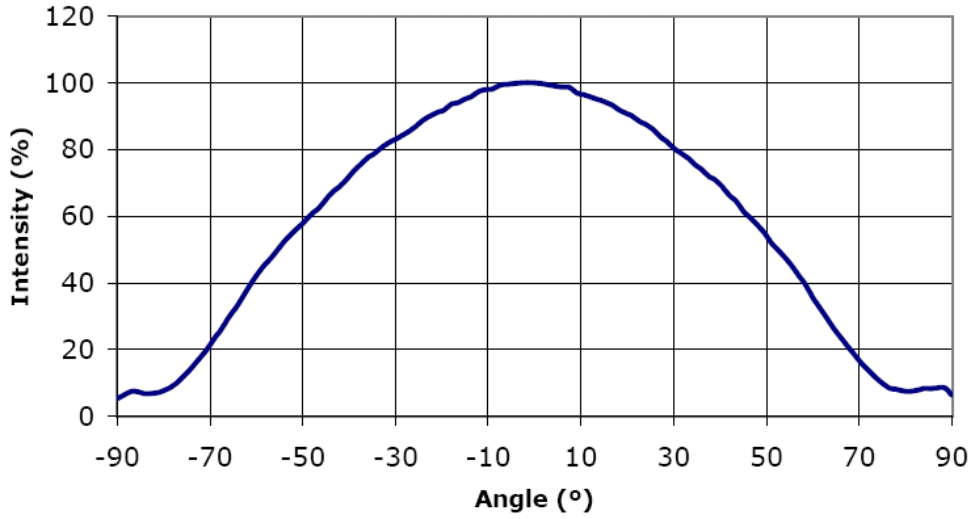


Figure 3-10 Cree XR Series LED Far Field Pattern

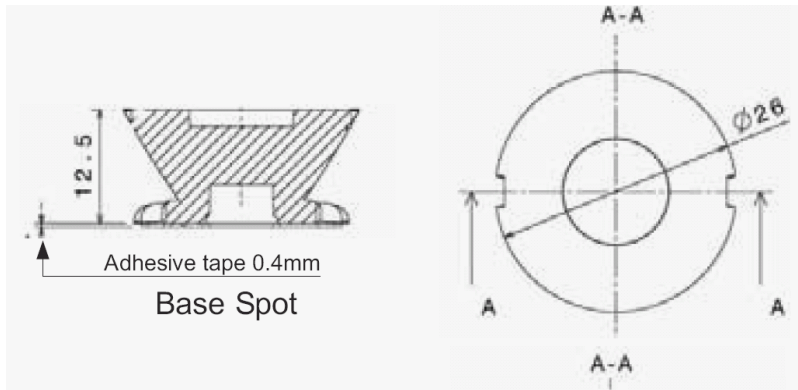


Figure 3-11 Mechanical drawing for Dialight OPTX1006



Figure 3-12 Dialight OPTX100 collimating lens for CREE LEDs

The completed assembly was measured to have a beam divergence of ~ 3.2 degrees. This produces a 4" beam spot at a distance of 12 feet.

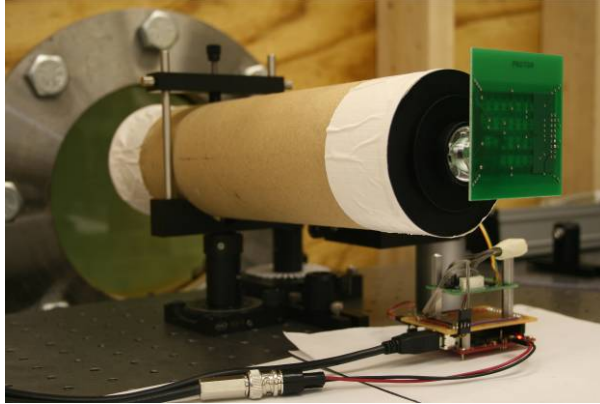


Figure 3-13 LED collimation assembly



Figure 3-14 Front View

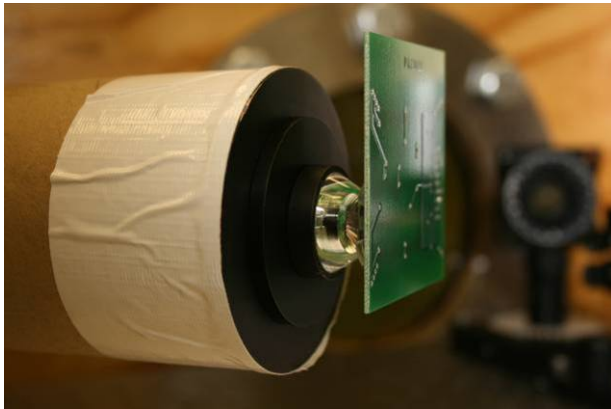


Figure 3-15 Back View - With LED on PCB



Figure 3-16 Back View without LED on PCB

3.2. Receiver Introduction

A photodiode is a semiconductor device that generates a current when light illuminates its PN junction. They combine a wide spectral response with good light-current linearity in a cheap, rugged, and compact package. Photodiodes also often provide good frequency response, but at the expense of additional noise, or vice versa. The main types of Si photodiodes are PN, PIN, and Avalanche photodiodes. PN photodiodes are general purpose photodiodes for light level detection while avalanche photodiodes are generally used for low light level detection.

3.2.1. Si PIN Photodiodes

The responsivity of photodiodes is dependent on the material chosen. Si photodiodes are the most common and easily available, but alternative choices include the possibility of wider band gap materials such as GaP, GaAs or even GaN/InGaN to obtain better response in the blue portion of the spectrum. When looking at these materials it was not obvious that GaP or GaAs would give significantly better system level response, and for simplicity Si photodiodes were chosen.

PIN photodiodes are relatively simple to work with, can have high responsivity at their peak responsivity wavelength, and have a linear response over many decades. Furthermore, they have low capacitance (which, like the LED, is dependent on the die size) and can provide a frequency response into the GHz range. Frequency response and linearity can be further improved by providing an external reverse voltage. However, this does have the disadvantage of increasing dark current and noise levels. These changes for the Hamamatsu S5973 Series Si PIN Photodiodes are shown in Figure 3-18.

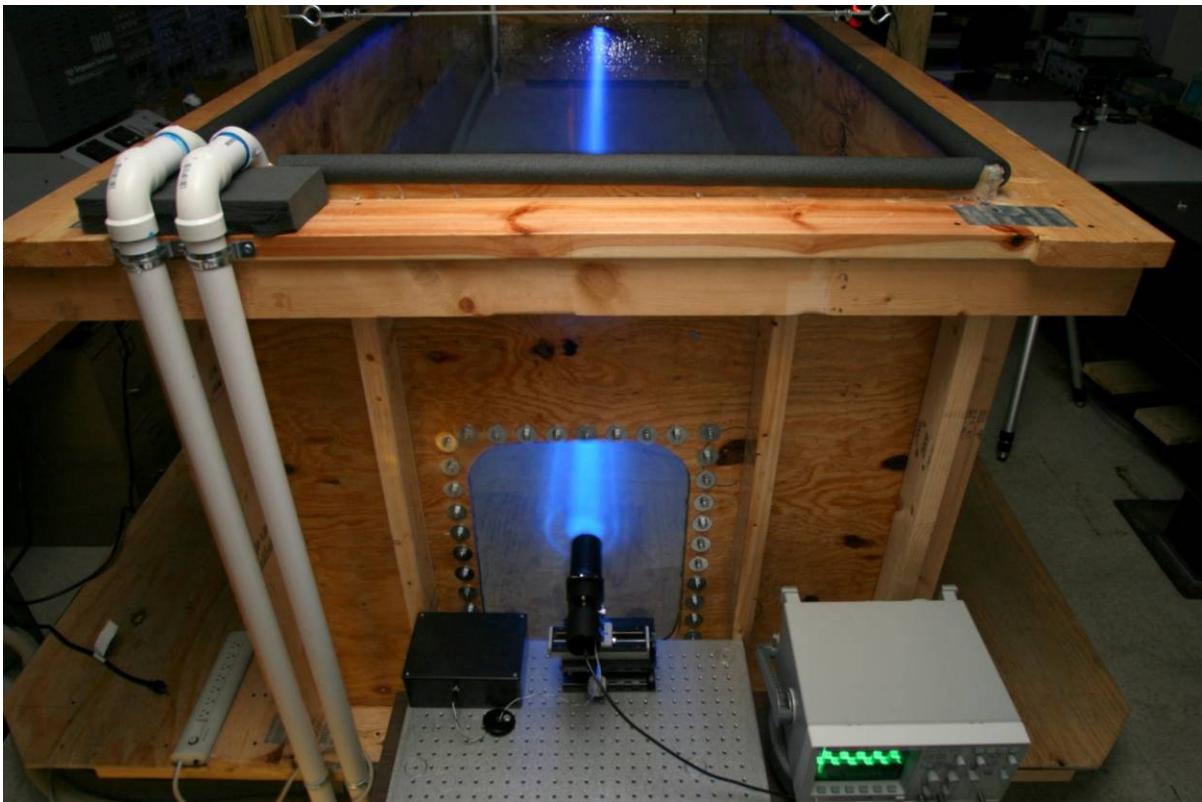


Figure 3-17 Receiver optics, pre-amp, batteries, digitizer and interface setup at receiving end of water tank

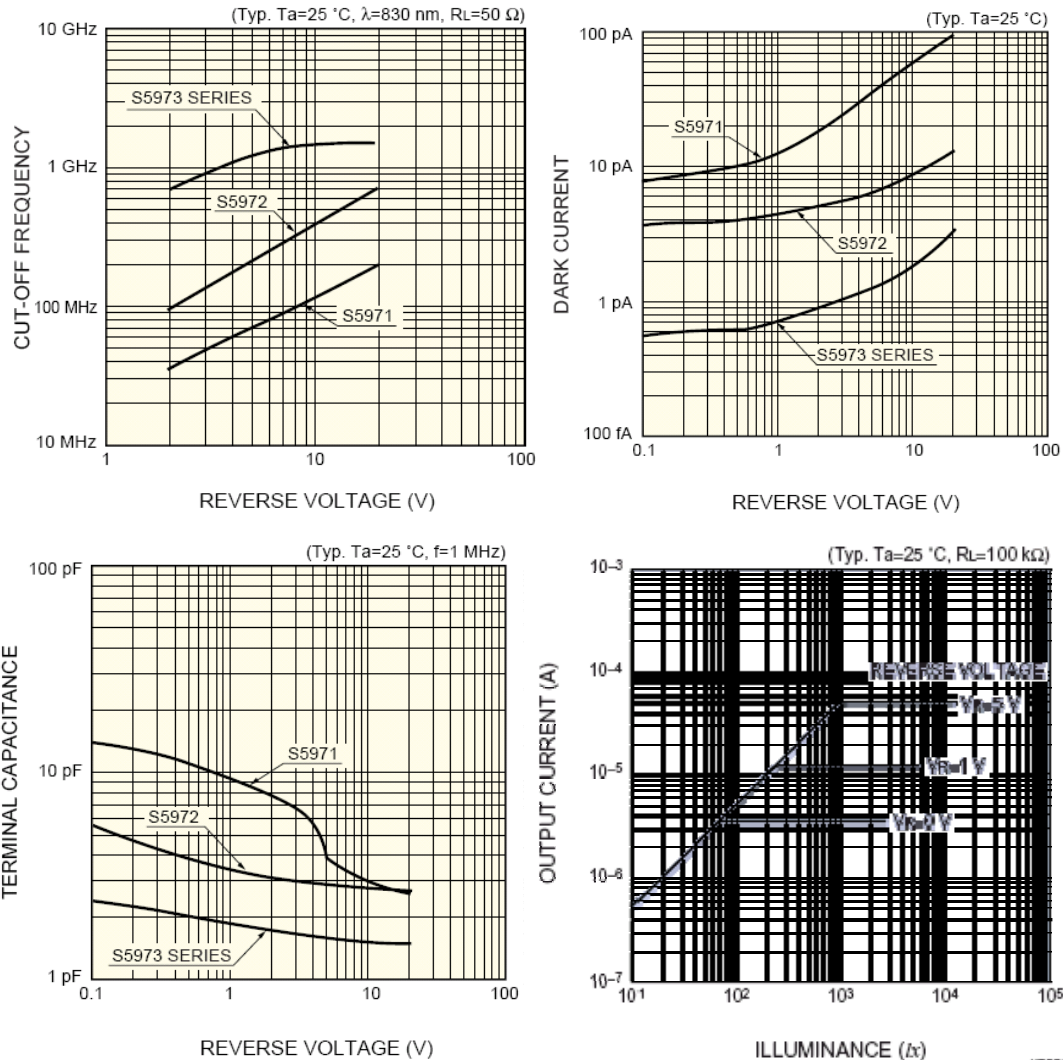


Figure 3-18 Cut-Off Frequency (Desirable - Increases with increase in reverse voltage), Dark Current (Undesirable - Increases with increase in reverse voltage) , Terminal Capacitance (Desirable – Decreases with increase in reverse voltage), and Output Current (Desirable - Increases with increase in reverse voltage) as a function of Reverse Voltage [7]

Hamamatsu S5973-02 Series Si PIN Photodiode

The S5973-02 [7] is a good choice for the detector since it has a good spectral response of ~0.4A/W in the blue-green wavelengths. Additional specifications of the S5973-02 are given in Table 3-4.

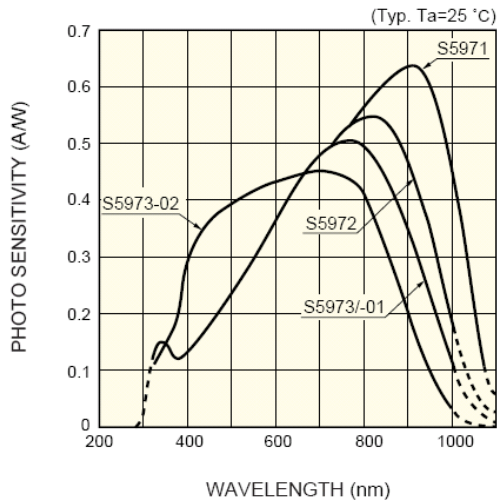


Figure 3-19 Spectral Response of the Hamamatsu S5973-02 Si PIN Photodiode

Cut-off Frequency	1.5GHz
Sensitivity at 400nm	0.3A/W
Sensitivity at 500nm	0.4A/W
Short Circuit Current	0.09 μ A
Dark Current	0.001nA
Terminal Capacitance at 1MHz	1.6pF
Package	TO-18
Active Area Size (radius)	0.4mm
Effective Active Area	0.12mm ²
Reverse Voltage	20V Max

3.2.2. Photodiode Pre-Amps

The small output current from a photodiode needs to be converted to a voltage and conditioned before digitization. This is achieved using a transimpedance amplifier followed by an instrumentation amplifier.

Transimpedance Amplifier

A transimpedance amplifier is simply an operational amplifier configured to generate an output voltage that follows the current at its non-inverting input as shown in the figure below.

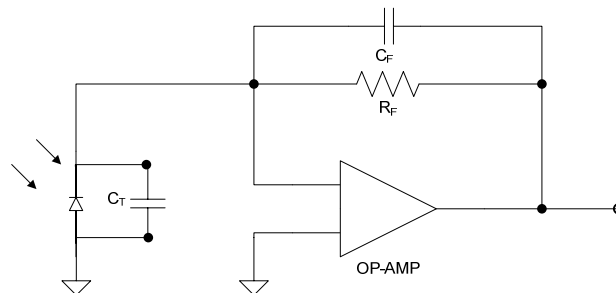


Figure 3-20 Basic Transimpedance Amplifier Circuit

C_T is the internal terminal capacitance of the photodiode and causes two problems. It reduces the bandwidth of the amplifier since it acts as a short at high frequencies. It also creates a low pass filter on the feedback path in combination with the feedback resistor R_F . This creates a negative phase to the feedback loop making the

circuit unstable. This effect can be countered by introducing a feedback capacitor C_F that acts as a high pass filter which adds a positive pole to the feedback loop. However, in practice, the value of this capacitor needs to be in the order of 1 or 2 pFs. Since dealing with this small value externally and the effect of other component tolerances on the performance of the circuit is significant, an integrated solution is most ideal as discussed in the next section.

Analog Devices AD8015 155 Mbps Transimpedance Amplifier

This integrated transimpedance amplifier is optimized for use as a fiber optic photodiode receiver pre-amp and can function with only one external component. In addition to the normal function of a transimpedance amplifier, it also implements current integration techniques on the input stage to improve performance.

The AD8015 also provides a linear output in the region of operation. The Hamamatsu S5973-02 photodiode being used outputs a current in the range of $0.5\mu\text{A}$ to $50\mu\text{A}$. This corresponds to a voltage of 0mV to 750mV from the AD8015. Other specifications of the amplifier are given in Table 3-5 [8].

Table 3-5 Specifications of the AD8015 Transimpedance Amplifier

Bandwidth	240MHz
Rise Time/Fall Time	1.5ns
Input current noise	3.0pA/Hz @ 100MHz
Optical Sensitivity	-36dBm at 155.52Mbps
Output Impedance	50 Ω

3.2.3. Photodiode Optics

The LED collimation optics discussed previously has a ~ 2 inch beam size at the other end of the test water tank (12 ft.) This requires the use of a light collecting and focusing assembly in front of the receiver photodiode. A simple plano-convex lens with a short focal length of 60mm is used for this purpose. Additionally an XY translating mount is used for the photodiode which connects the lens assembly section with the photodiode pre-amp section. The adjustment screw provides a $250\mu\text{m}/\text{rev}$ XY translation that allows the photodiode to be translated in order to illuminate the focal point on the

active area. Since the active area of the photodiode is only 0.12mm^2 this is crucial and provides excellent performance for alignment and light collection.



Figure 3-21 Lens Assembly – front view



Figure 3-22 Photodiode Assembly – front view



Figure 3-23 Lens Assembly – back view



Figure 3-24 Photodiode Assembly – back view

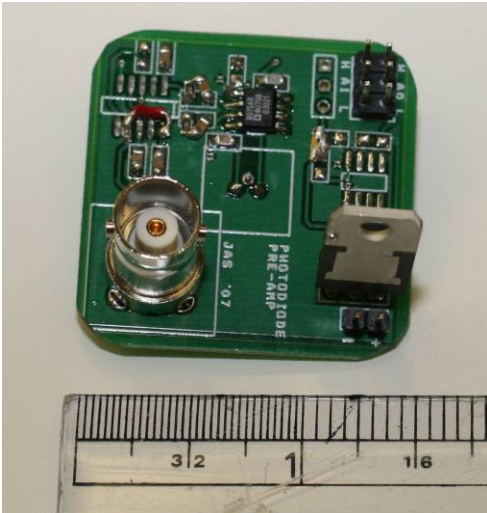


Figure 3-25 Pre-Amp Board front view

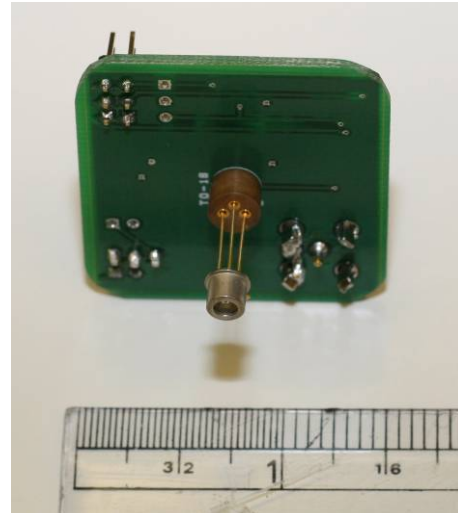


Figure 3-26 Pre-Amp board back view

3.3. Channel Emulation Setup

A 12 feet long, 4 feet wide, and 4 feet tall water tank was constructed to emulate different water parameters. An 8 inch diameter circular window allows the light source to enter the water tank while a 2 feet wide rectangular window on the other end allows the light to be collected by a receiver.

An 80 GPM pump and filter allows the water to be recirculated, drained, or filtered down to 3 microns.



Figure 3-27 Water Tank with 2.5 ft. of water

3.4. References

- [1] M. A. Chancey, "Short range underwater optical communication links," Masters Thesis, North Carolina State University, 2005.
- [2] Cree XLamp LEDs documentation. Available:
http://www.cree.com/products/xlamp_docs.asp
- [3] Panasonic 2SK0601 silicon NMOS. Available:
<http://www.datasheetarchive.com/2SK060100L-datasheet.html>
- [4] "Fairchild Semiconductor 74LS125 Quad 3-State Buffer Datasheet"
- [5] Texas instruments TLC5922 16 channel LED driver IC. Available:
<http://focus.ti.com/docs/prod/folders/print/tlc5922.html>
- [6] Dialight OPTX1006 cree LED lens product page. Available: <http://ledsupply.com/12-optx-1-006.php>
- [7] Hamamatsu S5973 Si PIN photodiode datasheet. Available:
http://sales.hamamatsu.com/assets/pdf/parts_S/S5971_etc.pdf
- [8] Analog devices AD8015 155Mbps transimpedance amplifier product page. Available:
<http://www.analog.com/en/prod/0%2C2877%2CAD8015%2C00.html>

CHAPTER 4. DATA ACQUISITION AND PC INTERFACE SETUP

This chapter explains how the front-end optics and electronics are linked with the PC for data transmission and processing. While this task is done by an FPGA with a USB2.0 controller on the transmitter side, the receiver side includes a Digitizer (Analog to Digital Converter (ADC) with a Digitally Programmable Variable Gain Amplifier (VGA) in the front) in addition to the FPGA and USB controller.

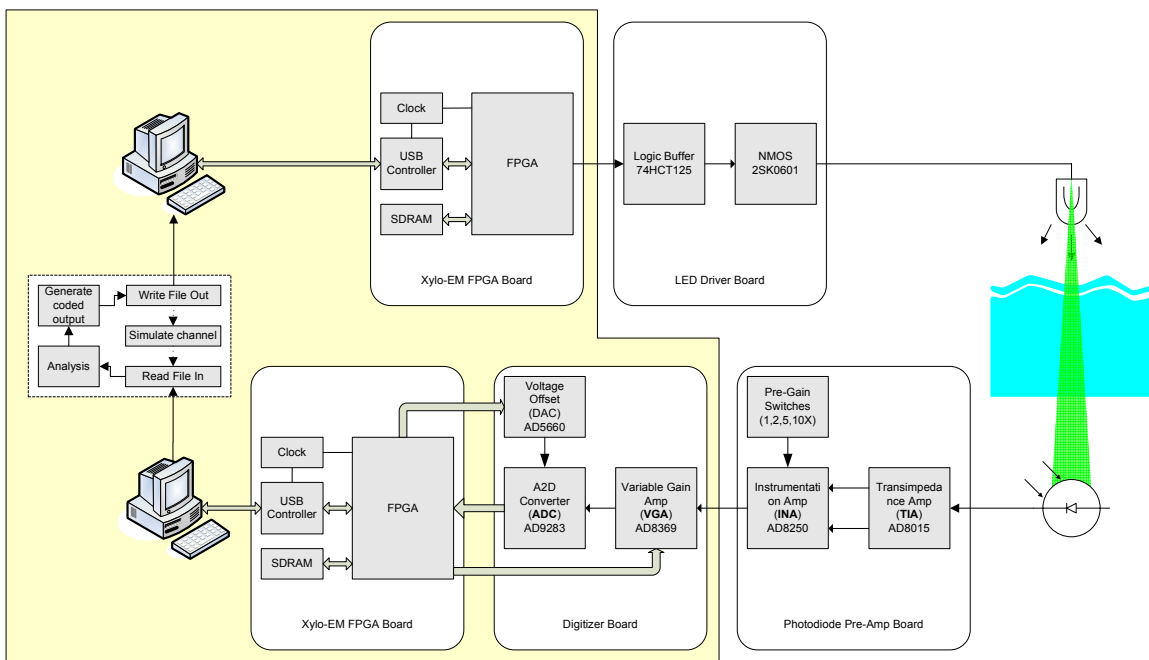


Figure 4-1 Interfacing Front-End with a PC

4.1. Data Acquisition Device

A variety of Data Acquisition (DAQ) products are available in the market that could be adapted to be used as a digitizer for interfacing the analog front end with the digital signal processor. However, they all have the disadvantages listed below that prevent them from being used in this particular application.

- Most of the commercially available DAQs are developed for test and measurements purposes only and not for real-time communications. Hence, they do not provide any sort of streaming capabilities to a real-time processor. Some provide streaming to a Hard Disk Array for future processing.

- At the data-rates required for the developed system most of these data acquisition devices interface directly to buses like PCI or PCI Express that require a dedicated desktop computer.
- Almost all commercially available data acquisition devices are only data acquisition devices in that they are not programmable to process the incoming data in addition to digitizing the received signal.

Since the project goals call for a modular system that can be implemented on a small Autonomous Underwater Vehicle (AUV), it was decided that a custom Data Acquisition device was to be built. Also, this would provide the capability to port the final digital signal processing algorithms to the point of acquisition (digitization). The advantages of this have already been discussed in Chapter 2.

For the transmitter the DAQ consists of an FPGA with a USB Controller. The receiver, in addition to the FPGA and the USB Controller, has an Analog to Digital Converter with a Variable Gain Amplifier. The following sections discuss these components of this Data Acquisition Device.

4.1.1. FPGA

A Field Programmable Gate Array (FPGA) is a re-programmable digital integrated array of thousands of basic digital logic elements capable of being configured to form the simplest of digital circuits to the most complex, space and resource permitted. It can be considered as a prototyping alternative to Application Specific Integrated Circuits (ASICs). While the circuit is designed in a high level Hardware Description Language (HDL) like Verilog, low level functions like place-and-route, verification and timing analysis are automatically handled by the manufacturer's Electronics Design Automation (EDA) tool.

FPGAs are ideal as an interface between analog circuits and Digital Signal Processors (DSPs) due to its inherent true parallel processing nature. This enables interfacing with multiple data converters simultaneously possible as well as for performing repetitive primitive operations easily on large amounts of data. FPGAs also have a large number of IOs and can be programmed to support any standardized serial data interface as well. A large number of pre-cooked functions can also be purchased as

“IP cores” and used along with user defined combinational logic saving development time.

At the heart of the Data Acquisition Device (DAQ) is an FPGA. The FPGA being used is an Altera Cyclone II EP2C5 device [1]. The same FPGA is used on the transmitter side and the receiver side. As shown in Figure 4-2 and Figure 4-3 the transmitter only includes the USB controller and the system clocks whereas the receiver includes the Analog to Digital Converter (ADC) and the Variable Gain Amplifier (VGA).

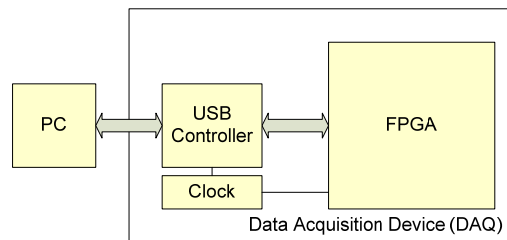


Figure 4-2 Transmitter DAQ Block Diagram

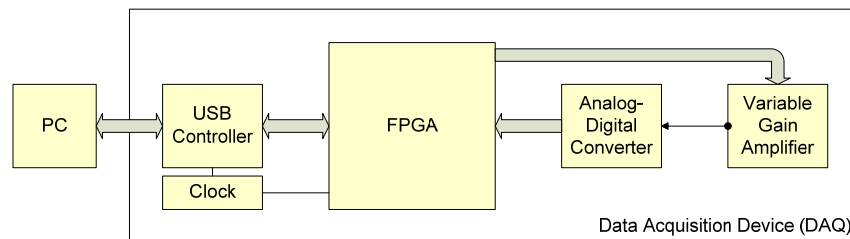


Figure 4-3 Receiver DAQ Block Diagram

Table 4-1 Altera Cyclone II EP2C5 FPGA Specifications

Logic Elements	4,608
Total Usable RAM (excluding parity bits)	104 kbits
Embedded 18x18 Multipliers	13
PLLs	2
I/O Pins	158

The specifications for the Altera Cyclone II EP2C5 FPGA are listed in Table 4-1, This particular FPGA is popular among prototype board developers like KNJN Electronic Development Boards [2]. One of their models, the Xylo-EM, was purchased as a development platform for the FPGA. An image of the Xylo-EM is shown in Figure 4-4. The board dimensions are 57mm x 60mm.

The Xylo-EM also includes, in addition to the Cyclone II FPGA, a Cypress USB 2.0 Controller [3] and a Micron 16Mbit SDRAM. The board also supports VGA and

10BASE-T Ethernet. In addition, 19 IO pins from the FPGA are brought out onto a 13x2 dual row 0.1” header. This header is used to interface the FPGA with the transmitter and receiver front-ends.

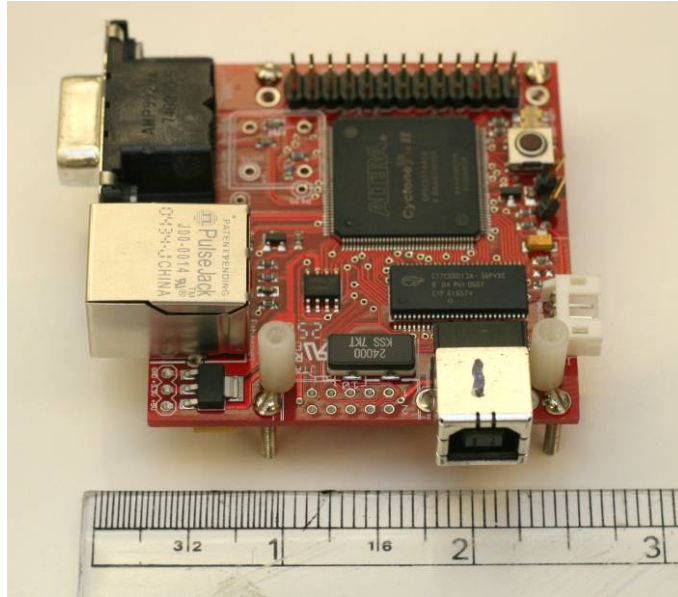


Figure 4-4 KNJN Development Board Xylo-EM

As previously mentioned, the receiver DAQ includes a Variable Gain Amplifier and an Analog to Digital Converter. Details about each of these follow.

4.1.2. Analog to Digital Converter with Variable Gain Amplifier

An Analog to Digital Converter (ADC) is used along with a Variable Gain Amplifier (VGA) to digitize the signal received from the photodiode pre-amp. An image of the board is shown in Figure 4-5.

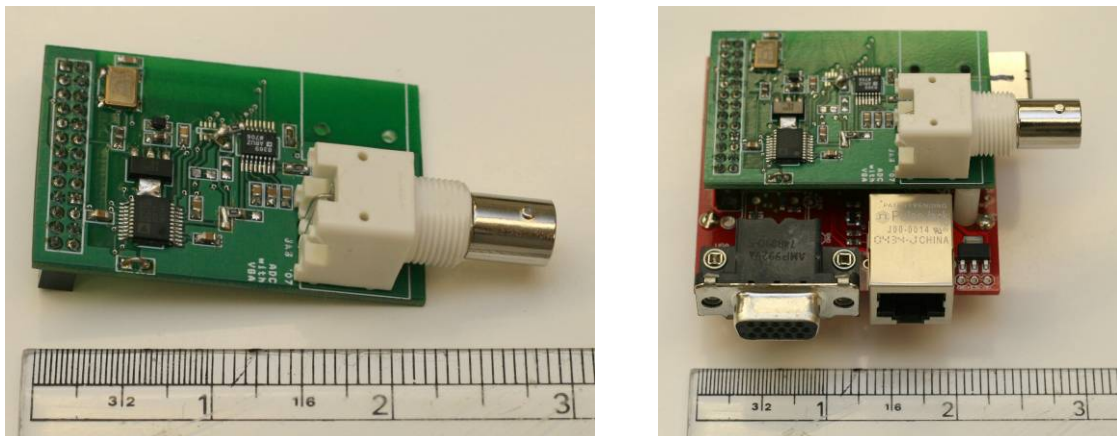


Figure 4-5 ADC with VGA

The ADC used is an Analog Devices AD9283 [4] and has a single channel with 8-bit resolution at 100MSps. The ADC has an on-chip reference thereby cutting the list of external components required to a clock and passive devices for filtering and decoupling. Detailed specifications of the ADC are listed in Table 4-2.

Table 4-2 AD9283 ADC Specifications

Resolution	8 bits
Max Sampling Rate	100MSps
Analog Bandwidth	475MHz
Interface	Parallel
Analog Input Range	1 Vpp
Input Resistance	10k Ω
Gain Error	$\pm 2.5\%$ FS
SNR	47dB

The VGA being used is an Analog Devices AD8369 [5] 45dB Digitally Controlled Variable Gain Amplifier. Using a four bit interface, the gain on the ADC can be set from -10dB to 35dB. Important specifications of the VGA are given in Table 4-3.

Table 4-3 AD8369 VGA Specifications

Bandwidth	600MHz
Input Resistance	200 Ω differential
Output Resistance	200 Ω differential
Noise Figure at maximum gain	7dB
Noise Figure at lowest gain	50dB
SNR at 70MHz and 2Vp-p output	51dB
SFDR at 70MHz and 2Vp-p output	-90dBFS

The DAQ was first characterized for gain linearity. A series of four tests were performed with varying input voltages and gain codes. The actual gain is then calculated from the ADC quantization codes and converted to dB. The result is plotted in Figure 4-6. It can be noted that although the gain switching is linear, the actual gain values seem to be higher than expected for low input voltages (20mVpp).

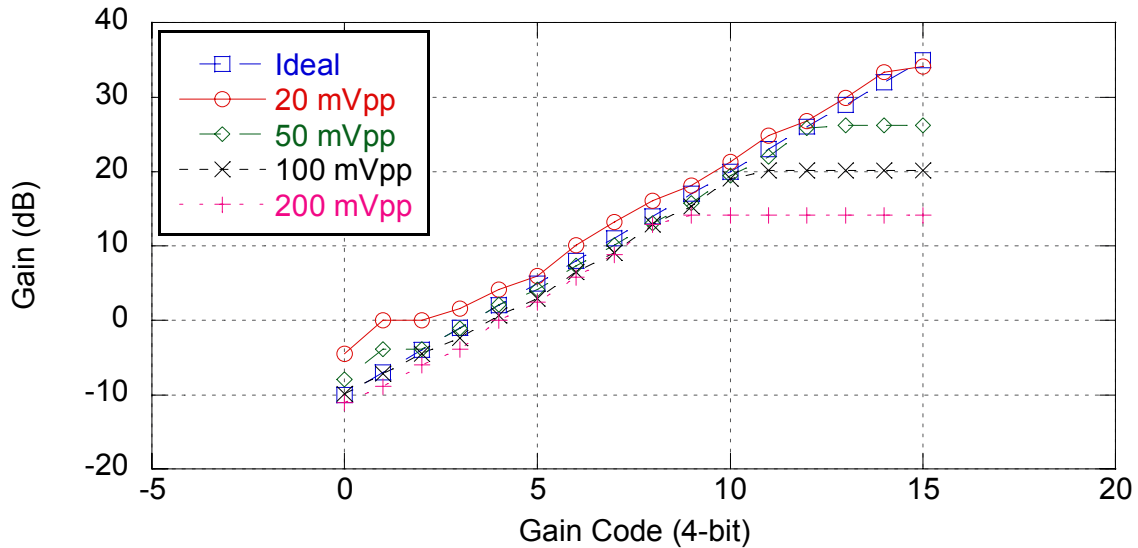


Figure 4-6 Gain vs. Gain Code for Varying Input Voltages

The DAQ was then characterized for noise performance using a 1MHz 900mVpp sinusoid input from an Arbitrary Waveform Generator. An FFT was done on the sampled data using LabVIEW. Even with the Variable Gain Amplifier (set at a gain code of 3) the noise floor is only ~5dB higher than typical performance for just the ADC alone.

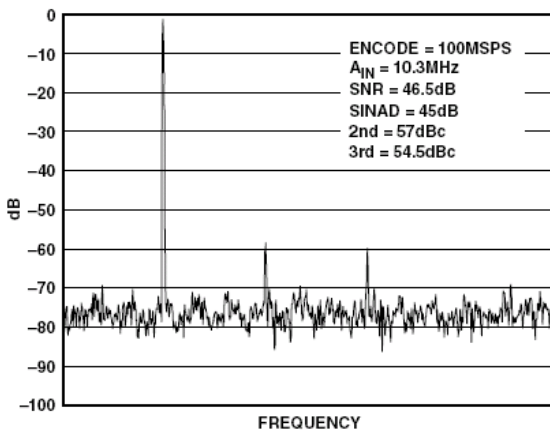


Figure 4-7 Typical performance specification for AD9283 Analog to Digital Converter (From Datasheet [4])

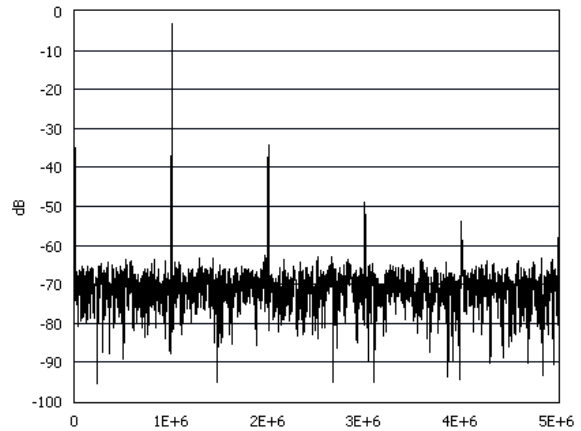


Figure 4-8 Measured performance characteristics of VGA + ADC + Interface. FFT generated from sampled data using LabVIEW

The presence of harmonics was speculated to be from the source itself. A spectrum analyzer was used to measure the power spectrum of the source and is shown in Figure 4-9. This, coupled with the performance of the ADC at second and third

harmonics, can be attributed to the appearance of harmonics in the power spectrum of the sampled signal.

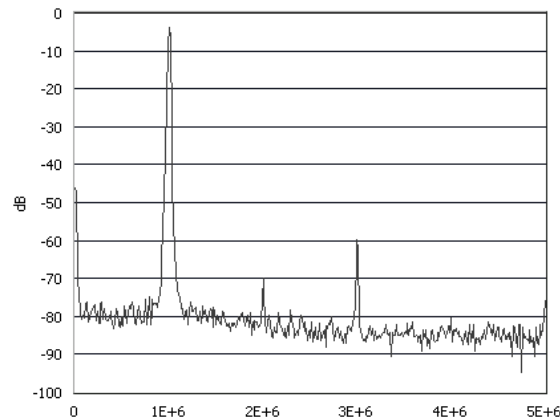


Figure 4-9 Power Spectrum of input source from the Arbitrary Waveform Generator

4.2. PC Interface

Previously in Chapter 2, it was shown that implementing the signal processing on a PC instead of an embedded processor, an FPGA, or in hardware would require a high bandwidth to the PC. For non-streaming applications, one possible alternative would be to store the data on an external storage medium and then slowly send the data to the PC. For a communications link, this is not really an option.

It can be calculated that for a transmitted data-rate of 1Mbps, an 8-bit ADC at 10 times oversampling would require a link to a PC with a bandwidth of at least 80Mbps. A variety of standard interfaces operate at or above that rate. Table 4-4 below outlines some of the popular ones [6].

Table 4-4 PC Interface Bus Comparisons

Bus	Data Rate
RS-232	921.6Kbps
10Base-T Ethernet	10Mbps
100Base-T Ethernet	100Mbps
USB 2.0	480Mbps
Firewire 1394b	800Mbps
Gigabit Ethernet	1Gbps

Three of these standards, RS-232, 10Base-T Ethernet and USB 2.0 are available on the Xylo-EM development kit. Since neither RS-232 nor 10Base-T Ethernet supports data rates at or above 80Mbps, USB was selected for the interface.

The USB Controller IC on the Xylo-EM is a Cypress EZ-USB FX2 CY7C68013A IC. Details about the USB architecture are explained in the next section.

4.2.1. USB Interface

Universal Serial Bus (USB) is probably the most popular interface standard for the PC. The architecture consists of at least 1 “host” (the PC) connected serially through a tiered-star topology to up to 127 “devices”. Each device can have up to 8 pairs of “pipes” for transferring data to and from the PC.

A USB Host consists of a PC with a USB Host Controller IC and the correct device drivers for the USB device about to be connected. The USB device consists of a Device Controller IC with firmware to communicate with the host. Upon connecting the device and the host, the host “enumerates” the device by first resetting, and then reading from the device its “Vendor ID” and “Product ID”. The host then uses this information to load the device’s corresponding driver. All communication is initiated by the host and is done in a round-robin fashion between all the connected devices.

USB 2.0 has a theoretical bandwidth of 480Mbps shared between all the devices connected to a single host. Actual data rates are usually much lower than 480Mbps. Simple tests revealed data rates as high as 400Mbps are achievable. Although this rate is much higher than the target rate of 80Mbps, there are some issues that prevent USB from being used as a reliable real-time communications link. Some of these are outlined below:

- The USB bandwidth is shared between all the devices connected to the host. So actual data rates vary and belong much lower than the realistic rate of 400Mbps.
- The USB host implements a round-robin scheme of polling the devices. Adding more devices increases latency between polls.
- USB protocol works on a host-master interface. The device can only communicate with the host when the host specifically requests for data.
- Additionally, a USB device communicates with the host in bursts every 250 μ s (polling period for USB 2.0) followed by a delay until the next polling cycle.

All of these problems, especially the latter, mean that USB is not ideal for streaming continuous data. There are some solutions, the simplest of which was implemented and is detailed in the next section.

It is conceivable that USB 3.0 standard will provide even higher data rate, but this is not yet available and will have its own set of unique problems to consider.

4.2.2. Streaming Implementation of USB

A simple solution to converting a “bursty” USB interface to the PC to a streaming interface is to implement a cyclical buffer on the FPGA. This “double buffering” scheme ensures that the ADC is continuously sampled and the data is placed in an array which is sent off to the PC in bursts of USB packets. The same technique is implemented on the transmitter side to ensure that the LEDs are driven continuously without interruption. An illustration of the mechanism is shown in Figure 4-10 and Figure 4-11. A USB packet can range anywhere from 1 to 65,535 bytes. It was also experimentally determined that the ideal size for a reliable USB interface was using packet sizes of 4096 bytes.

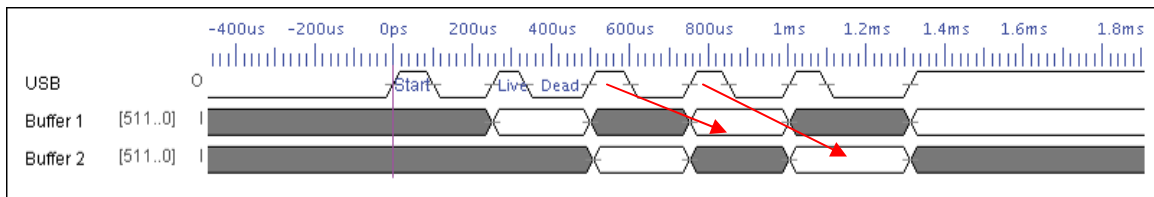


Figure 4-10 Double Buffering for the transmitter

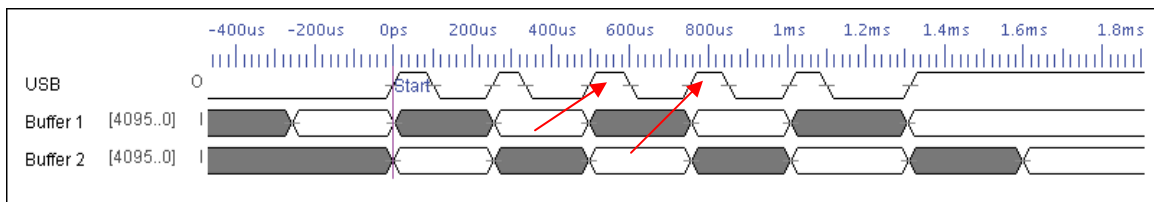


Figure 4-11 Double Buffering for the receiver

The transmitter buffers were emptied onto an IO line driving the LEDs at 1Mbps. The rate at which the receiver buffers are filled determines the sampling rate at the ADC. A sampling rate too high would result in the buffers being filled before the USB was ready to send data to the PC. This would result in an entire buffer (packet of 4096 Bytes) being dropped. A sampling rate too low would not be utilizing the USB bus as much. Statistics for USB “packet loss” due to varying sampling rate was conducted and the results are shown in Figure 4-12.

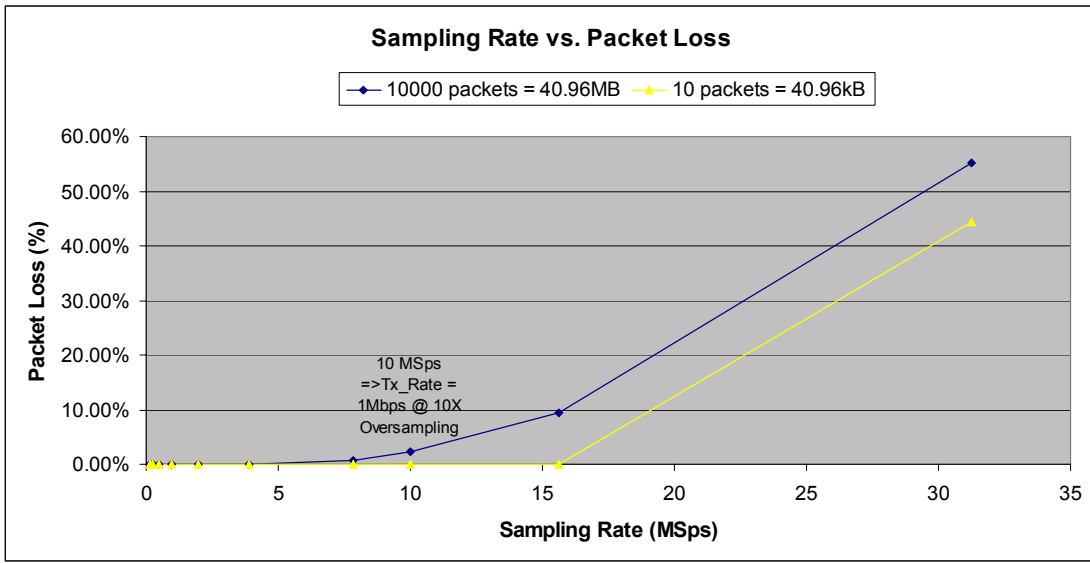


Figure 4-12 Receiver Sampling Rate vs. Packet Loss

This is consistent with the 250 μ s polling period of the USB host PC which equates to a polling frequency of 4 kHz. Since 4096 bytes (samples) are sent every polling period, the theoretical breakdown point for the “double buffering” scheme would be 16.384MSps, which can be verified by the experimental results in Figure 4-12. Since this includes the desired sampling rate of 10MSps, the simple solution of double buffering provides a streaming interface over a “bursty” USB interface.

4.3. PC Software

The PC software for prototyping purposes consists of a C++ based windows console application and DSP algorithms implemented in LabVIEW.

The C++ application can send data files of binary format to the transmitter and receive data files of sampled analog data from the digitizer. The LabVIEW application is capable of receiving and demodulating the sampled data. It is also capable of running BER analysis by comparing the detected data to the transmitted data. A built in simulator for the channel allows the application to test out scenarios and expected results before even conducting the actual experiment.

4.4. References

- [1] Altera Cyclone II FPGA Product Page. Available:
<http://www.altera.com/products/devices/cyclone2/cy2-index.jsp>
- [2] KNJN Electronic Development Products. Available: <http://www.knjin.com>
- [3] Cypress EZ-USB FX2 USB controller product page. Available:
<http://www.cypress.com/portal/server.pt?space=CommunityPage&control=SetCommunity&CommunityID=209&PageID=215&gid=9&fid=14&category=All&showall=false>
- [4] Analog devices AD9283 8-bit 100MSPS ADC product page. Available:
<http://www.analog.com/en/prod/0%2C2877%2CAD9283%2C00.html>
- [5] Analog devices AD8369 programmable variable gain amplifier product page.
Available: <http://www.analog.com/en/prod/0%2C2877%2CAD8369%2C00.html>
- [6] PC interface buses. Available: http://www.interfacebus.com/Interface_PC_Buses.html

CHAPTER 5. EXPERIMENTS AND RESULTS

5.1. Underwater optical link

The transmitter and receiver were initially tested separately. The photoreceiver circuit was tested with modulated laser diode sources first. The LED light source was then detected and measured using an off the shelf Thorlabs 400-1100nm Large Area Si Amplified Photodetector. Then the performance of the LED transmitter and photoreceiver was tested together as an optical underwater link. The laser diodes were red (630 nm) and blue (405 nm). The LED light sources were Blue (470 nm) and Green (525 nm). The photoreceiver circuit used a blue enhanced Si photodiode.

For each test a data file containing 30,000 random bits were transmitted at 1Mbps and over sampled at 10MSPs. The digital data was stored for future analysis to compute Bit Error Rates. A screenshot of this recorded data is shown below. An image of the test setup at the receiver end for each test is included in the corresponding section. The oscilloscope in the images show a stream of alternating 1's and 0's sent at 1Mbps. This is to provide a visual record and to better understand the received waveform and identify issues such as ringing and rise and fall times, which will be useful in understanding the bandwidth limitations of the system.

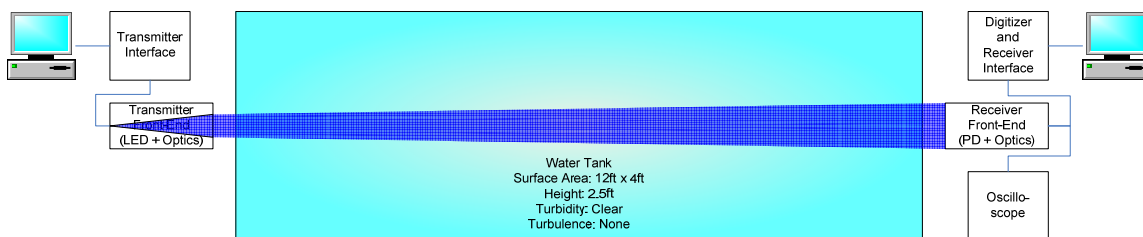


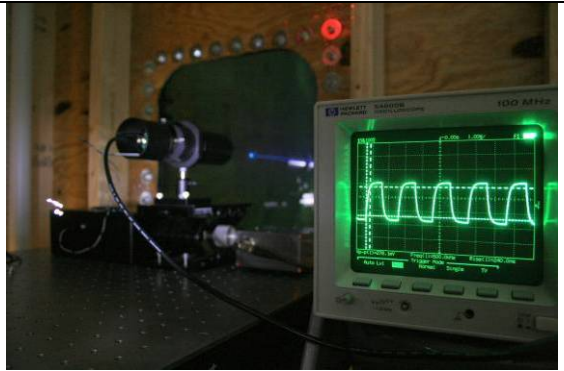
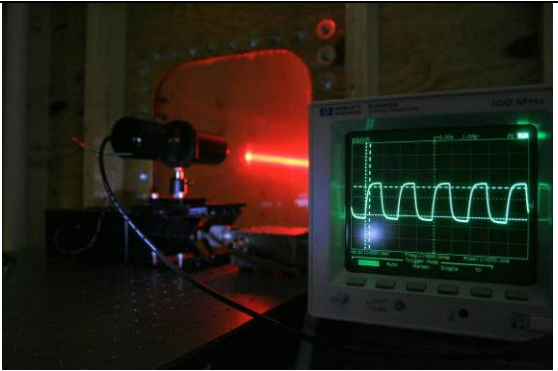
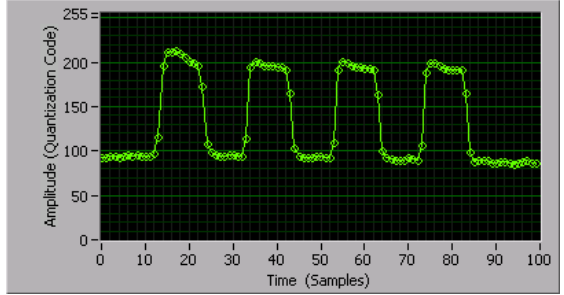
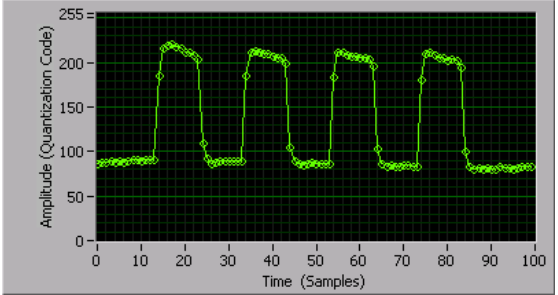
Figure 5-1 Link Testing Setup

5.1.1. Receiver Testing

The Hamamatsu S5973-02 photodiode based receiver was first characterized by receiving light from two separate pulse train modulated laser sources, first with a blue (405nm) laser and then with red (630nm). The photodiode photosensitivity is about 440mA/W at 630nm and about 300mA/W at 405nm [1]. Additionally, the red laser diode output was set at about ~15mW while the blue laser diode was set at ~12mW. As expected, this resulted in the output voltage of the receiver from the red light being higher than that from the blue for clear water, where the absorption is not significantly different.

Note that in the pictures the responsivity of the camera is also much higher in the red portion of the spectrum than the blue. Thus, while the blue light is actually scattered more, the red scatter is more easily visible. The oscilloscope displays the direct output from the photodiode pre-amplifier, while the sampled waveform includes an additional gain from the variable gain amplifier between the pre-amp and the ADC, the gain of which is shown by VGA in the table. Since the input impedance of the scope is very high while the pre-amp expects a 50Ω load, the rise times appear longer (higher) on the scope.

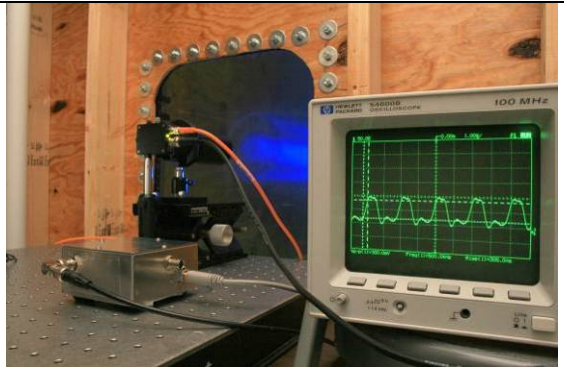
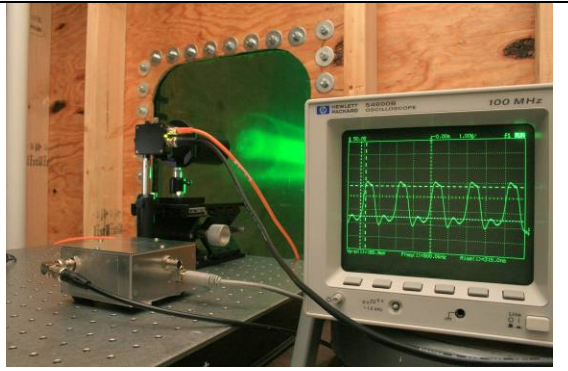
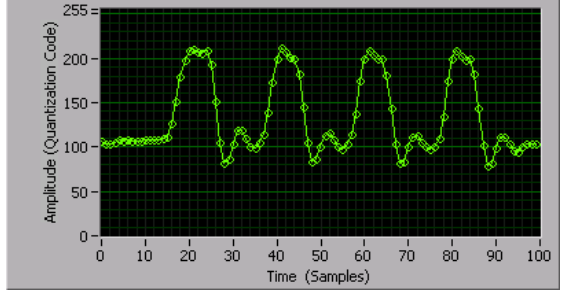
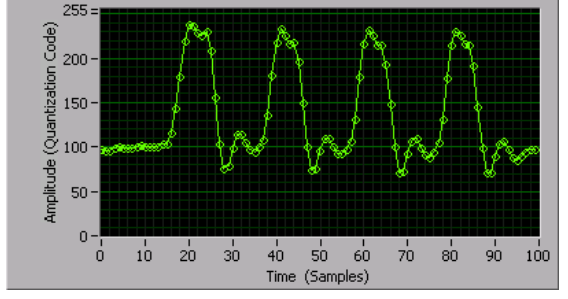
Table 5-1 Test results for the receiver built

Transmitter: Blue Laser Diode Receiver: Hamamatsu S5973-02 Based	Transmitter: Red Laser Diode Receiver: Hamamatsu S5973-02 Based
	
$V_{pk-pk}:278mV$; Rise Time:240ns; Sampled@VGA=6	$V_{pk-pk}:535mV$; Rise Time:270ns; Sampled@VGA=5
	

5.1.2. Transmitter Testing

The Cree XR7090 LED based transmitter was then tested using a Thorlabs PDA100A 400-1100nm large area Si amplified Photodetector [2]. This detector is relatively slow, with the gain of the detector set at 0dB the bandwidth of the detector is 1MHz. This corresponds to a rise time of $\sim 300\text{ns}$. Only 240mA of current were sent through the LED. According to datasheets, these LEDs are rated for as high as 700mA continuous current. The NMOS driving the LED is capable of pushing up to 750mA of current at 5V between the drain and source and 5V at the gate. This was tested separately and found to be accurate based on current readings on the power supply. A square pulse train was used to modulate the LEDs. The ringing on the waveforms was found to be mostly controlled by power bypassing at the LED light source. A high value capacitor ($>10\mu\text{F}$) would smooth out the ringing, but replacing this component with a suitable ceramic or tantalum capacitor was postponed so system level testing could be performed. For the data presented here, a much smaller capacitor ($0.01\mu\text{F}$) was used.

Table 5-2 Test results for the transmitter built

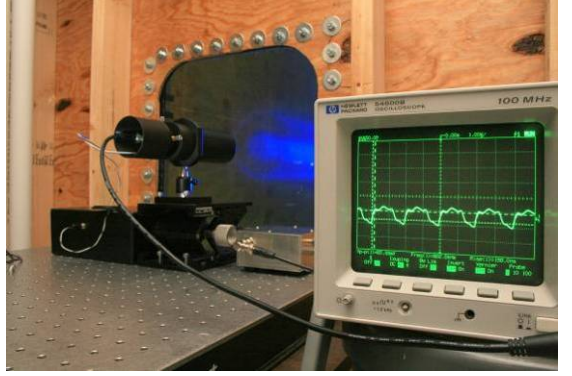
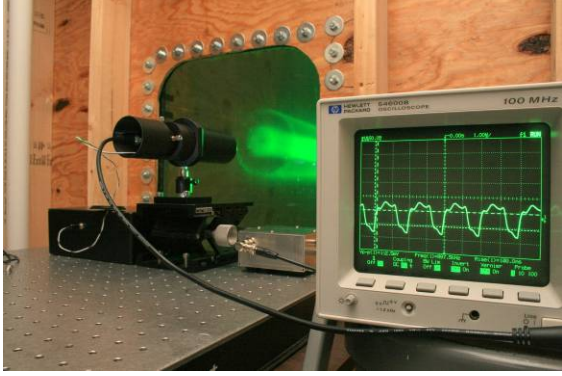
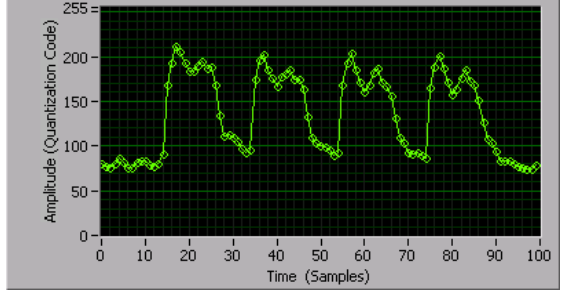
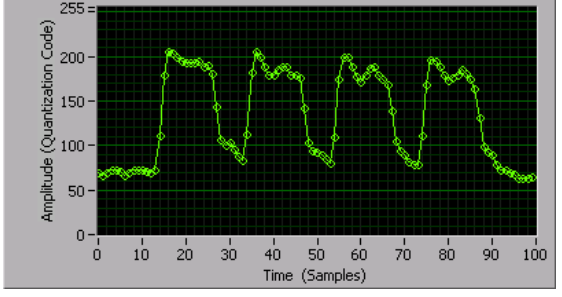
Transmitter: Blue Cree XR7090 LED Receiver: Thorlabs PDA100A	Transmitter: Green Cree XR7090 LED Receiver: Thorlabs PDA 100A
	
$V_{pk-pk}: 100\text{mV}; \text{Rise Time}: 305\text{ns} \text{ Sampled@VGA}=9$	$V_{pk-pk}: 161\text{mV}; \text{Rise Time}: 315\text{ns} \text{ Sampled@VGA}=8$
	

5.1.3. Receiver and Transmitter Testing

Finally, the receiver and transmitter built were tested together. A square pulse train was used to modulate the transmitter. Voltage levels similar to what was observed from the commercial Thorlabs detector was obtained. Since the photosensitivities and the active area of the two detectors are different, the results are dependent on the alignment of the optical beam. This makes it impractical to make a quantitative comparison, but it is very useful to qualitatively compare the commercial detector to our photoreceiver approach in order to understand our performance.

As expected, rise times shorter than the commercial detector were observed by using our design. Depending on the gain setting and component choices, rise times as low as 70ns were observed for different values for the above parameters. This was not optimized since the target data rate of oversampling the 1Mbps data rate by 10 times was being met. We anticipate that for higher data rates it will be necessary to optimize the receiver bandwidth.

Table 5-3 Test results for the receiver and transmitter built

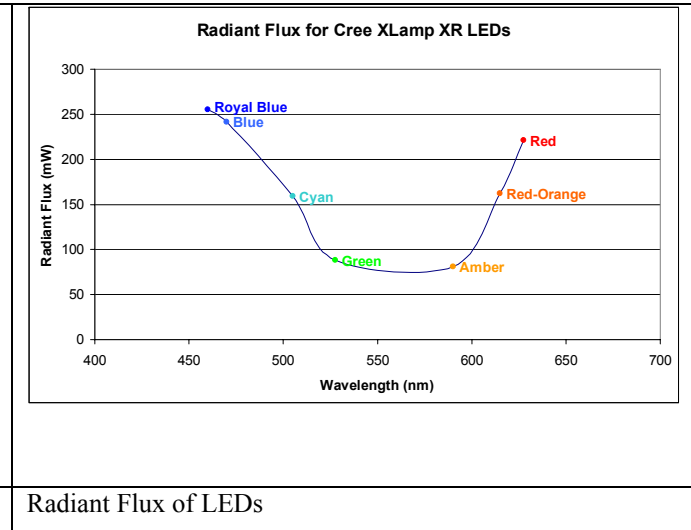
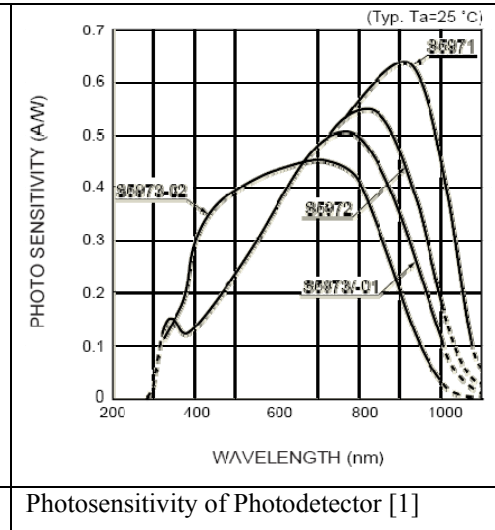
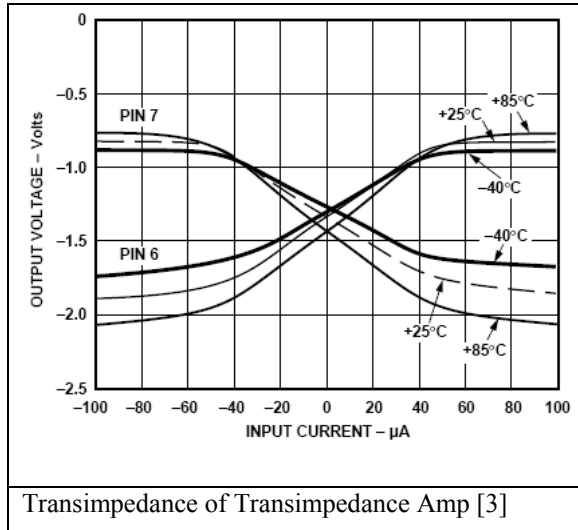
Transmitter: Blue Cree XR7090 LED Receiver: Hamamatsu S5973-02 Based	Transmitter: Green Cree XR7090 LED Receiver: Hamamatsu S5973-02 Based
	
$V_{pk-pk}: 67mV$; Rise Time: 150ns Sampled@VGA=10	$V_{pk-pk}: 112mV$; Rise Time: 140ns Sampled@VGA=9
	

5.1.4. Estimation of the channel attenuation using the Tx/Rx

Since the optical power at the transmitter and receiver responsivity and transimpedance are known, we can estimate the attenuation of the channel. Table 5-4 uses the measured values from section 5.1.3 and specifications from the device datasheets to calculate attenuation at the two LED wavelengths.

Table 5-4 Attenuation by water calculated from measured transmitter and receiver parameters

Specified	Measured	Specified	Specified	Calculated	Calculated	Calculated	Calculated	Specified	Measured
Wavelength	Received Voltage	Pre-Amp Transimpedance	Photodiode Photosensitivity	Receiver Volts per Watt	Received Optical Power	Water Attenuation	Transmitted Optical power	LED Radiant Flux for 350mA	Transmitted Current
(nm)	(V)	(V/A)	(A/W)	(V/W)	(W)	(dB)	(W)	(mW)	(A)
470nm	67mV	10V/mA	375mA/W	3.75V/mW	17.86uW	39.66dB	165.25mW	241mW	240mA
525nm	112mV	10V/mA	420mA/W	4.2V/mW	26.67uW	33dB	60mW	88mW	240mA



5.2. LED Properties in water

5.2.1. Crude Beam Profiling

In order to conduct a crude beam profiling to understand the attenuation due to scattering of light from the LED source, an experiment as outlined in Figure 5-1 was conducted. A Digital SLR camera with a 10.1 megapixel CMOS sensor (22.2 x 14.8mm) [4] was used along with a 50mm lens to capture the images for analysis in MATLAB. Although the peaks are saturated, the images still provide some information about the drop in intensities as turbidity is increased. Increase in turbidity was achieved by adding Maalox to the water [5]. A rudimentary comparison between the amounts of Maalox as related to seawater is given in Table 5-5 [6].

Two sets of captures were taken for each increment in turbidity. The first set was taken at a constant exposure (Shutter Speed of 1 sec and F-Stop of F10.0) and the second set was taken at increasing exposures in order to collect the same amount of light in all increments. All tests were done with the LED operating at 240mA.

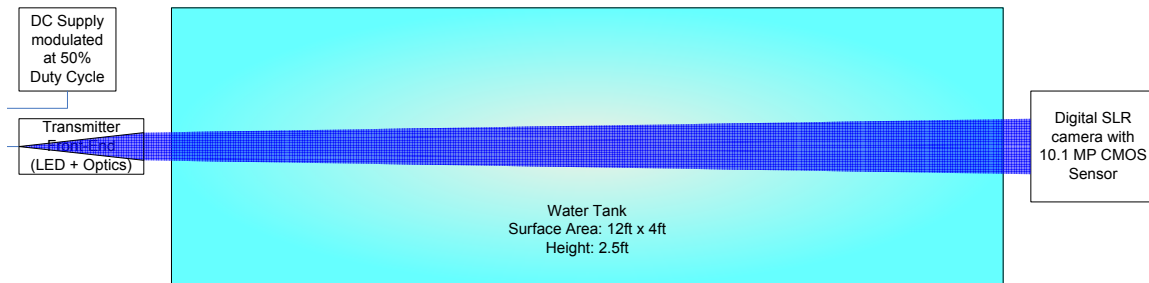


Figure 5-2 Experiment setup for crude beam profiling

Table 5-5 Estimated Maalox concentrations and seawater turbidity

Water Type	Maalox Amount	Water Amount	Maalox Conc.
Clear water	0	3398 L	0 ppm
Coastal water	100 mL	3398 L	~30 ppm
Harbor water	200 mL	3398 L	~60 ppm
Turbid water	300 mL	3398 L	~90 ppm

The camera can be viewed as an array of individual photodetectors with similar responsivity and good linearity over a wide range of light intensities. In this portion of the thesis, this linearity is assumed in that, for a given photon flux, the recorded signal is doubled

if the exposure time is doubled. Issues such as blooming, which is defined as the transfer of excess electrons to adjacent pixels, are ignored. In future work, the response of the detector array can be calibrated and more quantitative results can be obtained. However, these results provide qualitative information that is very useful in understanding the propagation of light through the tank and how it impacts the information channel.

The camera exposure can be controlled by adjusting either or both the shutter speed and the lens aperture. With increased turbidity, the amount of light at the camera decreases. This limits the ability to capture and plot a good beam profile. A simple technique to account for this drawback and to obtain a better beam profile for the particular amount of light available is to increase the exposure as the turbidity increases. If the aperture is fixed, then linearity can be assumed. At high and low intensity levels the aperture can be adjusted to widen the effective dynamic range. The downside to this approach is as the aperture size is changed, the numerical aperture is effectively changed, and thus a different range of scatter photon angles can be received. This can be addressed empirically by overlapping data sets taken at different exposure and aperture combinations, or by modeling the optical system. However, this was beyond the current scope of work for this Masters Thesis. In general, these complications do not prevent obtaining suitable data to qualitatively understand the beam spread as the beam traverses the tank. Table 5-6 summarizes the exposure conditions.

Table 5-6 Increasing exposure relationship to Luminance

	Lens Aperture	Shutter Speed	Exposure	Luminance	Illuminance
	(f-stop)	(sec)	(lux sec)	(cd/m ²)	(lx)
Constant setup	10.0	1	6.64	12.5	250
Increasing setup					
0 mL	2.5	1/80	8.97	62.50	1250
25 mL	2	1/60	7.91	30	600
50 mL	1.8	1/25	6.34	10.13	202.5
75 mL	1.8	1/10	5.02	4.05	81
100 mL	1.8	1/4	3.7	1.62	32.4
200 mL	1.8	1.6	1.02	0.25	5.06
300 mL	1.8	5	-0.63	0.08	1.62

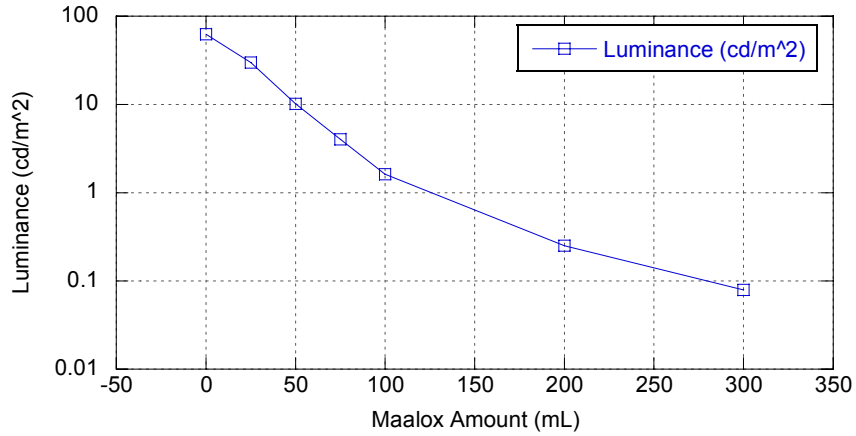


Figure 5-3 Measured luminance as a function of exposure for different Maalox amounts

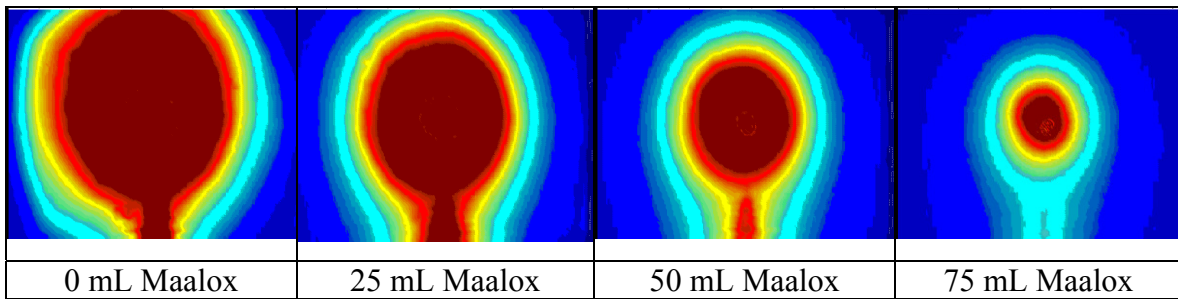


Figure 5-4 Drop in intensities at constant exposure for varying Maalox concentrations

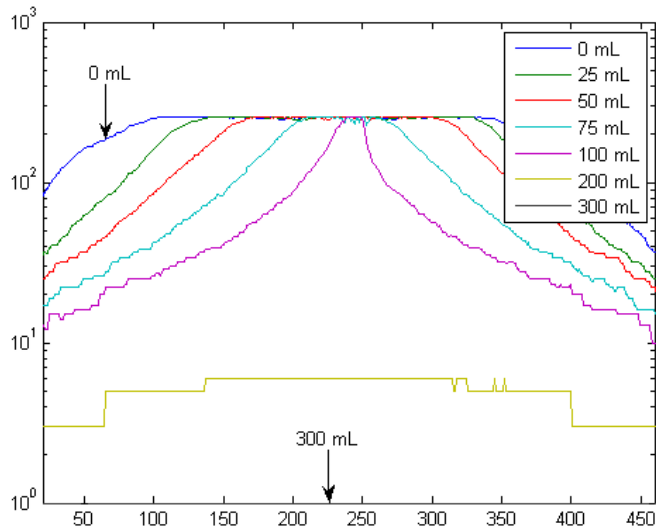


Figure 5-5 CMOS Image Intensities of LED at 12 ft through water for varying concentrations of Maalox (Constant exposure on CMOS sensor)

Figure 5-4 and Figure 5-5 were taken for constant exposure while varying Maalox concentrations. It can be noted that amount of light reaching the receiver decreases in both the forward and sideways directions. Beyond 200 mL of Maalox (harbor water), very little light is reaching the receiver as compared to ambient light.

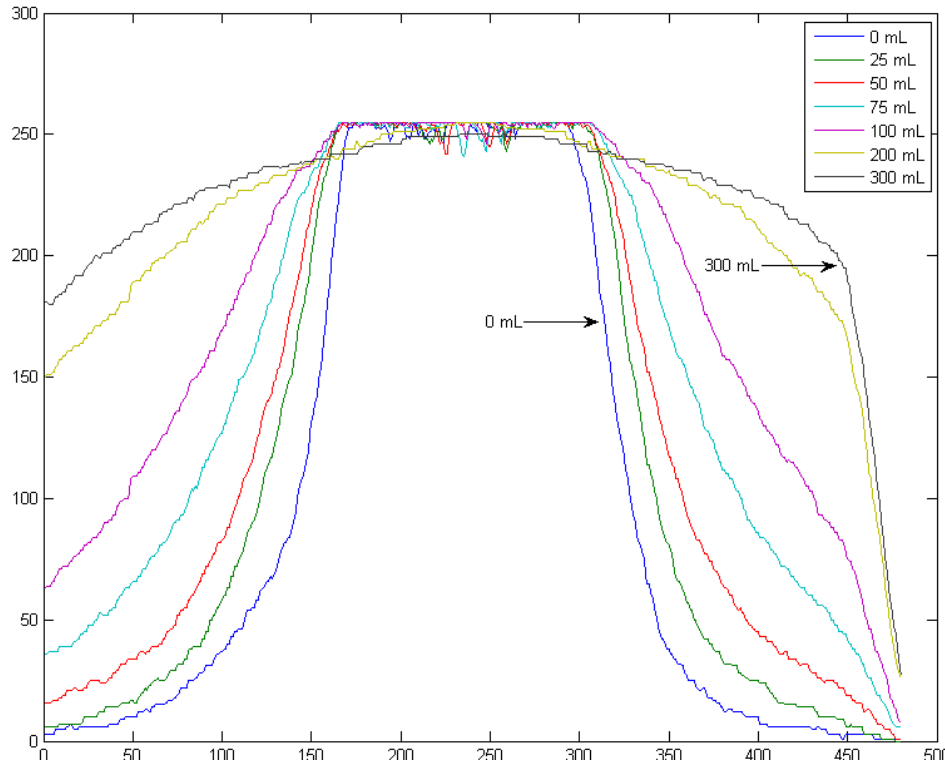


Figure 5-6 CMOS Image Intensities of LED at 12 ft through water for varying concentrations of Maalox (Increasing exposure on CMOS sensor)

Figure 5-6 shows data taken for increasing exposure with varying Maalox concentrations. The beam width appears to double in diameter between clear water and 200mL of Maalox. The drop in intensity due to forward scattering cannot be estimated because of the CMOS sensor saturation.

5.3. Attenuation Comparison

An experiment for the analysis of the properties of light from an LED source through water was conducted by comparing two light sources, a blue LED (470nm) and a green LED (525nm), through three different media, as shown in Figure 5-7.

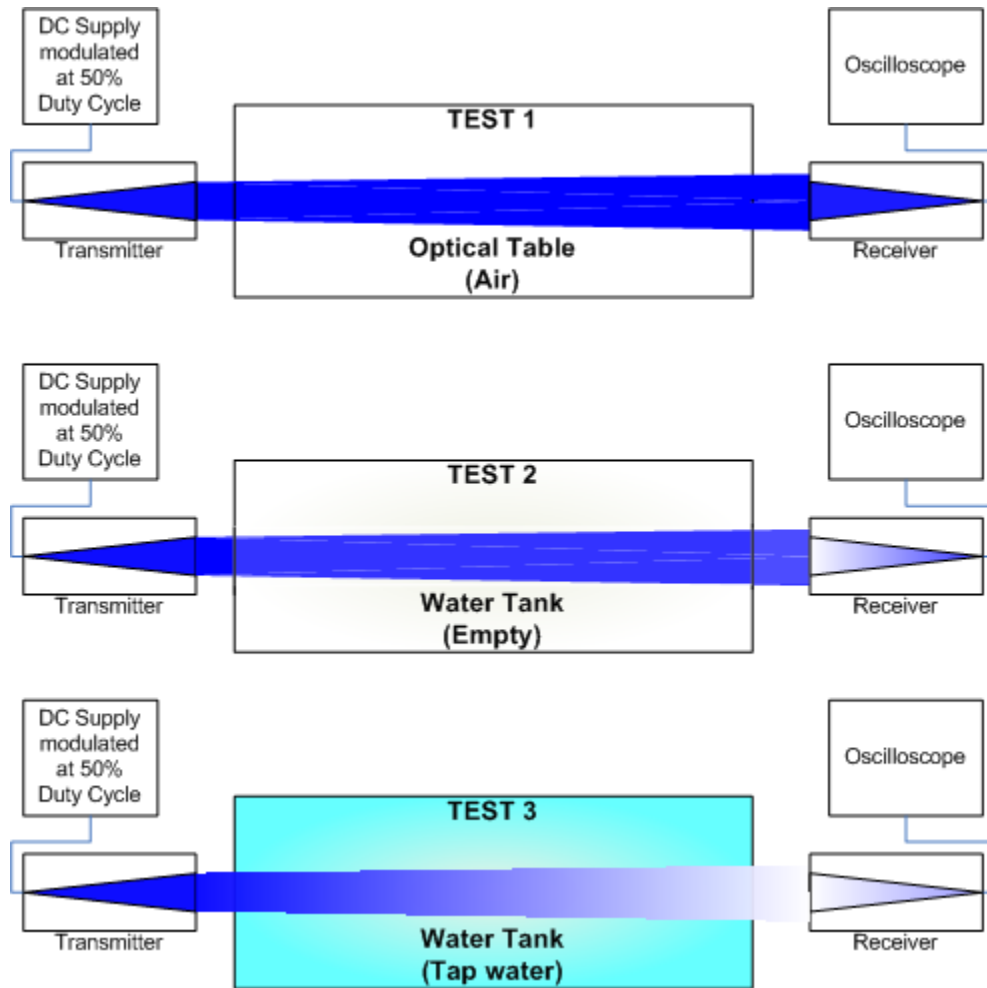


Figure 5-7 Experiment Setup for Attenuation Comparison

The first test provides a relative value for attenuation through air for both blue and green wavelengths. The second test provides a comparative value for the attenuation resulting from transmit and receive acrylic windows on the water tank. Finally, the third test provides a comparative value for attenuation of the same wavelengths through water.

Table 5-7 Results from Attenuation Comparison

Specified	Measured	Specified	Specified	Calculated	Calculated	Calculated	Calculated	Specified	Measured
Wavelength	Received Voltage	Pre-Amp Trans-impedance	Photodiode Photosensitivity	Received Optical Power	Attenuation	Attenuation	Transmitted Optical power	LED Radiant Flux for 350mA	LED Current
Optical Table - air									
(nm)	(V)	(V/mA)	(mA/W)	(W)	(X)	(dB)	(W)	(mW)	(mA)
470	150.00E-03	10	375	40.00E-06	6.03E+03	37.8056932	241.34E-03	241.34	350mA
525	123.00E-03	10	420	29.29E-06	3.02E+03	34.7940324	88.32E-03	88.32	350mA
Water Tank - empty									
(nm)	(V)	(V/mA)	(mA/W)	(W)	(X)	(dB)	(W)	(mW)	(mA)
470	63.75E-03	10	375	17.00E-06	14.20E+03	41.5218039	241.34E-03	241.34	350mA
525	49.37E-03	10	420	11.75E-06	7.51E+03	38.7584522	88.32E-03	88.32	350mA
Water Tank - tap water									
(nm)	(V)	(V/mA)	(mA/W)	(W)	(X)	(dB)	(W)	(mW)	(mA)
470	5.00E-03	10	375	1.33E-06	181.01E+03	52.5769057	241.34E-03	241.34	350mA
525	8.00E-03	10	420	1.90E-06	46.37E+03	46.6621836	88.32E-03	88.32	350mA

Transmitted current and received voltages were recorded and uses along with specifications of the LED, photodiode, and transimpedance amplifier to calculate the transmitted and received optical power. This was used to calculate the attenuation through the various media for the two wavelengths. The results were also plotted as is shown in

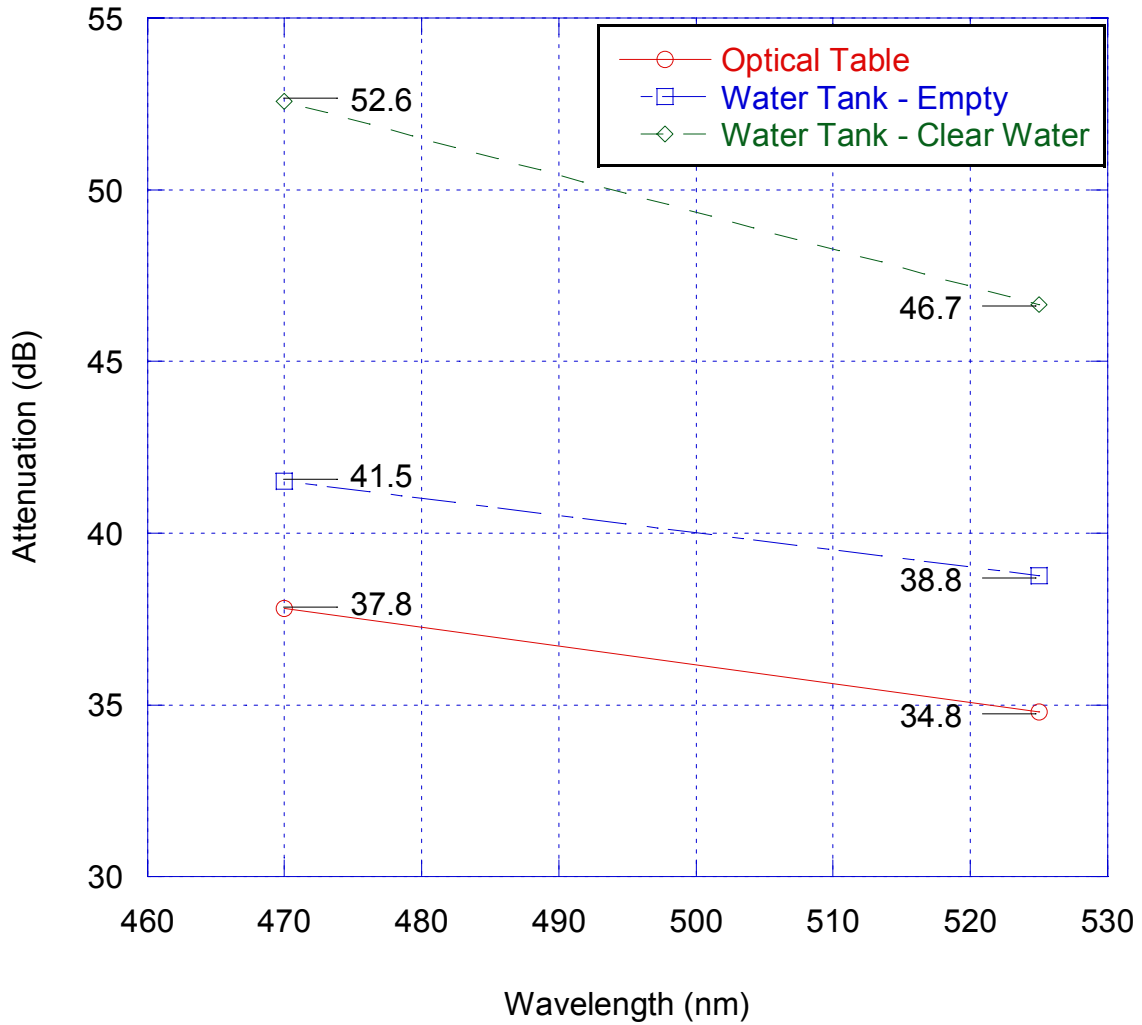


Figure 5-8 Results for Attenuation Comparison

The results from air show that blue is attenuated about 3dB more than green in air. The windows on the water tank attenuate the signals about 4dB. The tap water, by itself, attenuates blue by 11.1dB and green by 7.9dB compared to attenuation through the empty tank.

5.4. System Improvements

A second revision of the pre-amp and digitizer were made to improve the performance of the system. The main change made was to keep the signal path from the pre-amp to the ADC fully differential. A 3MHz passive RC low-pass filter was also added at the output of the pre-amp for filtering high frequency signals without rounding off the square pulses too much. Simulation results showed the filter rise times to be $\sim 65\text{ns}$. An improved LED board was also built for better heat sinking for the Cree LEDs.

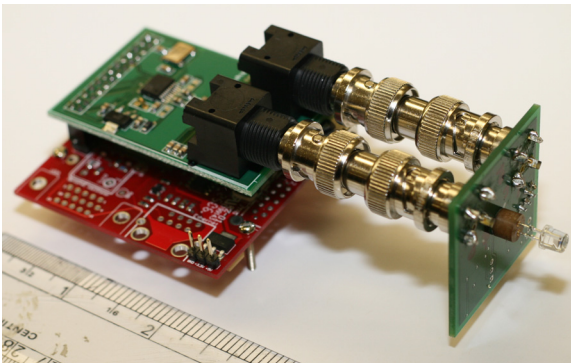


Figure 5-9 Receiver Revision 2

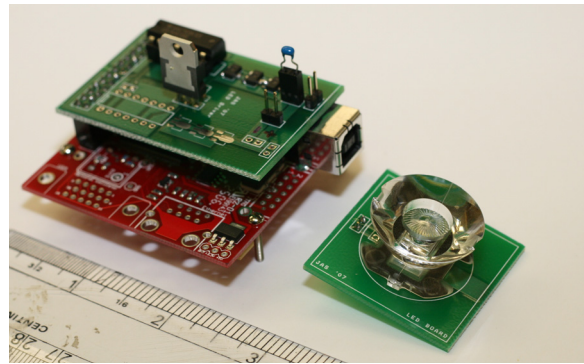


Figure 5-10 Transmitter Revision 2

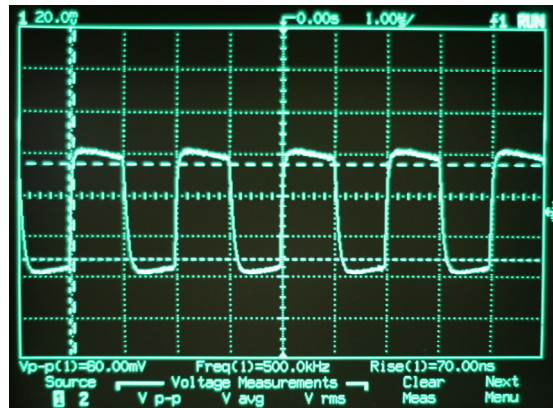


Figure 5-11 Square pulse received by pre-amp

A square pulse test signal was transmitted and received by the pre-amp. The screenshot above from an oscilloscope shows the rise times to be $\sim 70\text{ns}$, potentially limited by the rise time of the RC filter. The tilt in DC levels during high and low states are caused by the series AC coupling capacitors.

5.5. References

- [1] Hamamatsu S5973 Si PIN photodiode datasheet. Available:
http://sales.hamamatsu.com/assets/pdf/parts_S/S5971_etc.pdf
- [2] Thorlabs PDA100A amplified photodetector product page. Available:
<http://www.thorlabs.com/thorProduct.cfm?partNumber=PDA100A>
- [3] Analog devices AD8015 155Mbps transimpedance amplifier product page. Available:
<http://www.analog.com/en/prod/0%2C2877%2CAD8015%2C00.html>
- [4] Canon XTi digital SLR product page. Available:
<http://www.usa.canon.com/consumer/controller?act=ModelInfoAct&fcateoryid=139&modelid=14256>
- [5] M. A. Chancey, "Short range underwater optical communication links," Masters Thesis, North Carolina State University, 2005.
- [6] Y. Gawdi, "Underwater free space optics," Masters Thesis, North Carolina State University, 2005.

CHAPTER 6. CONCLUSION

Underwater communications are useful in a variety of different fields. Some of the most applicable ones are:

- Underwater sensor network and observatories
- Security and harbor inspections
- Maintenance and surveying of underwater pipelines and oil rigs
- Mine warfare and anti submarine warfare
- Autonomous underwater vehicle communications
- Remotely operated vehicle telemetry and telecommand.

Most of these applications benefit from following features of optical communications:

- High data rates (1Mbps to 1Gbps)
- Wireless operation
- Conservative power budget
- Low cost and reproducibility
- High Fidelity of data transmission.

A short study of the current state of the art implementations led to the following conclusions. High data rate (1 – 100 Mbps) communications at relatively short distances (<100m) are feasible [1-3]. The demand to replace existing acoustics communication links with free space optical communication is valid and significant [4]. However, most systems are implemented using expensive lasers for light sources and/or Photomultipliers tubes (PMTs) for detectors which impede creating a low cost platform and ease of reproducibility. A low cost alternative using LEDs as light sources and Photodiodes as receivers is a viable option for very short range operations. However, this would be achieved at the expense of being strongly disadvantaged in photon limited and high noise environments. Digital Signal Processing (DSP) techniques like optimized modulation formats, error-correction coding, and others demonstrated successful in RF applications could be applied to an LED-Photodiode

system to compensate for some of the performance losses. Almost all current implementations use On-Off Keying (OOK) and simple detection methods in hardware to discriminate and detect the received bits. Applying DSP requires the capabilities to digitize the received signals and interface with a processor. Since this is a unique approach in the realm on underwater free space optical communications, it was decided to use a PC itself as the processor for maximum flexibility, availability of pre-cooked algorithms, and simulation. An example of having a higher degree of control over discriminating and detecting the received bits is illustrated by the same implemented on data received using the developed system. It can be seen that the detected bits correlate with the transmitted bits without error even when the actual digitized signal (and its power spectrum) looks noisy for a level detector.

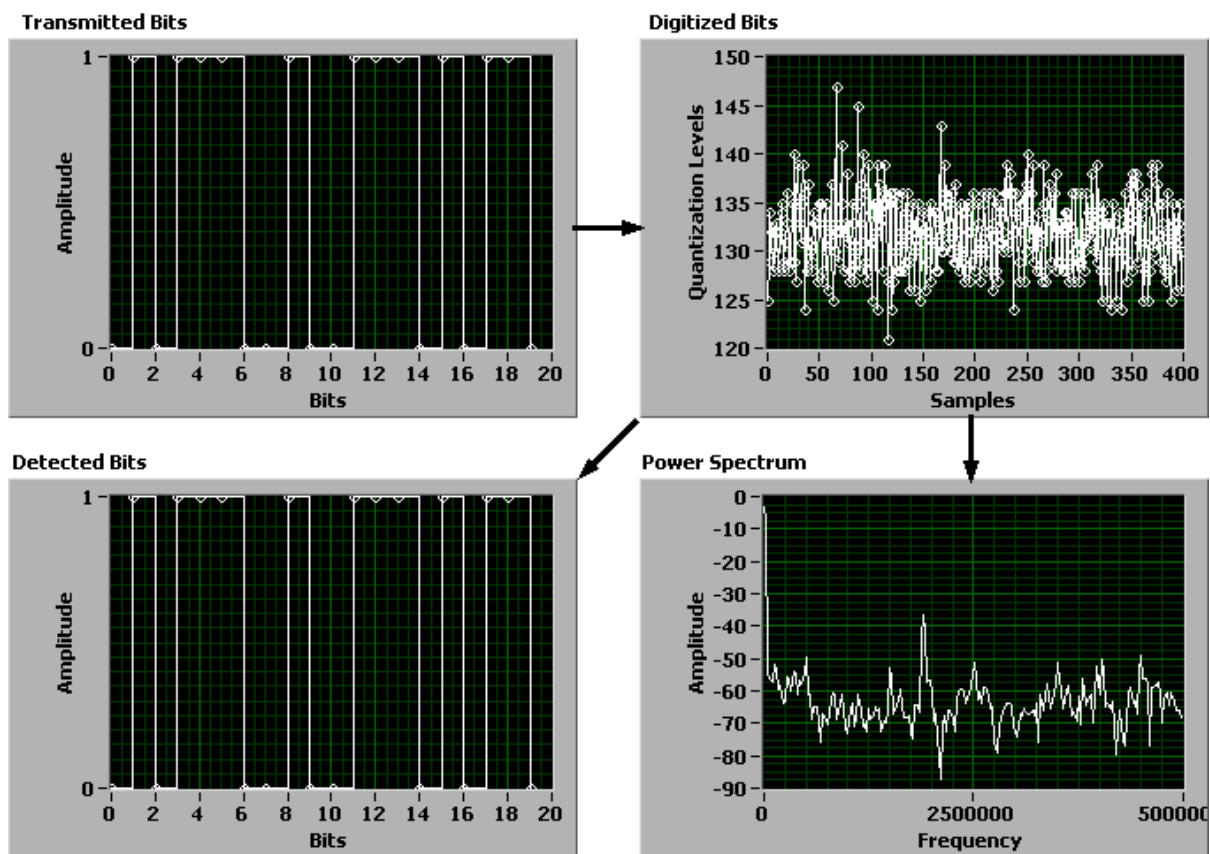


Figure 6-1 Example of advantages of DSP based detection over a level detector. 20 transmitted bits (top left) sent using RZ coding and sampled at 10X correspond to 400 samples at the processor (top right)

The following infrastructures were set up to prototype and validate the above claims:

- A Cree XR7090 LED based transmitter including collimating optics and drive electronics capable of operating at or above 1Mbps.
- A Hamamatsu S5973 Si PIN photodiode based receiver including light collecting optics and pre-amplifier electronics capable of operating at or above 1Mbps.
- USB 2.0 based interface to and from the front-ends to a PC including a digitizer for sampling received signals for Digital Signal Processing at 8-bits with 10 times oversampling.
- PC Software to transmit and receive sampled data over High Speed USB 2.0 interface.
- PC Software to receive, analyze and implement DSP techniques.
- A 12 ft x 4 ft x 4 ft water tank to emulate different ocean water optical properties in lab.
- Transmitters and Receivers small in form factor and efficient in power resources to be carried on a small Autonomous Underwater Vehicle (AUV).

In summary, the infrastructure has been set up to implement and characterize Digital Signal Processing techniques for Underwater free space optical communications. With efficient algorithms, a low cost, high fidelity optical communication system capable of providing an alternative to acoustic communication underwater is possible in the near future.

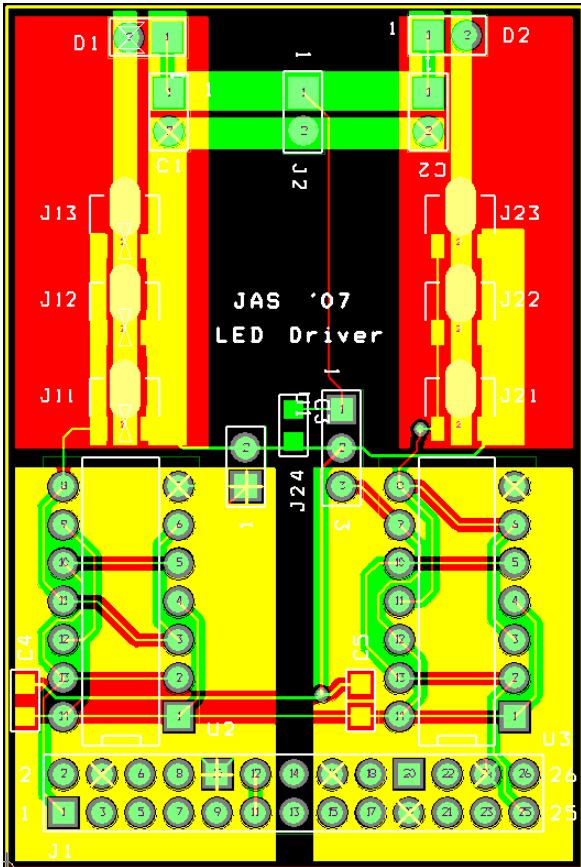
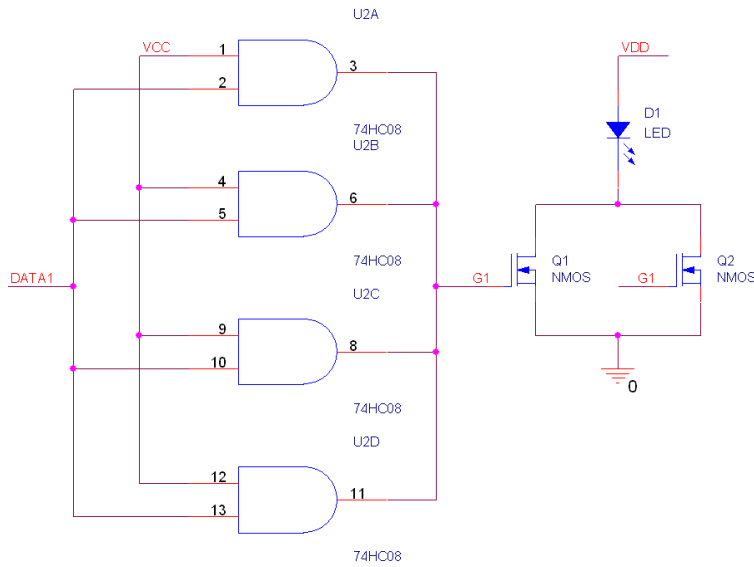
6.1. References

- [1] M. A. Chancey, "Short range underwater optical communication links," Masters Thesis, North Carolina State University, 2005.
- [2] (2002, June 15, 2002). Free-space optics (FSO) submerged. *Optics Report* Available: http://www.opticsreport.com/content/article.php?article_id=1015
- [3] N. Farr, A. D. Chave, L. Freitag, J. Preisig, S. N. White, D. Yoerger and F. Sonnichsen, "Optical modem technology for seafloor observatories," in *OCEANS 2006*, 2006, pp. 1-6.
- [4] J. Partan, J. Kurose and B. N. Levine, "A survey of practical issues in underwater networks," in *WUWNet '06: Proceedings of the 1st ACM International Workshop on Underwater Networks*, 2006, pp. 17-24.

APPENDIX

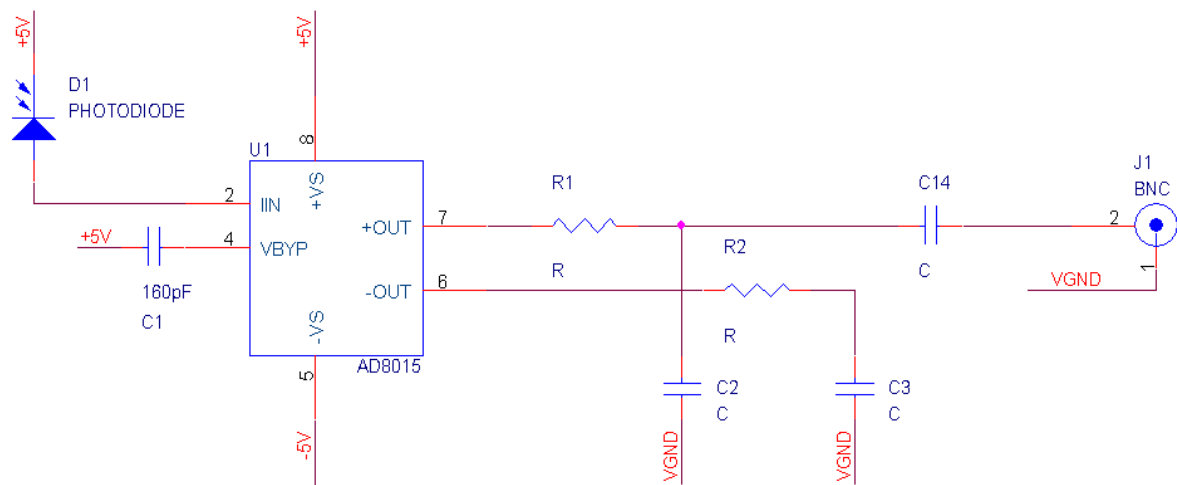
Schematics and Layout

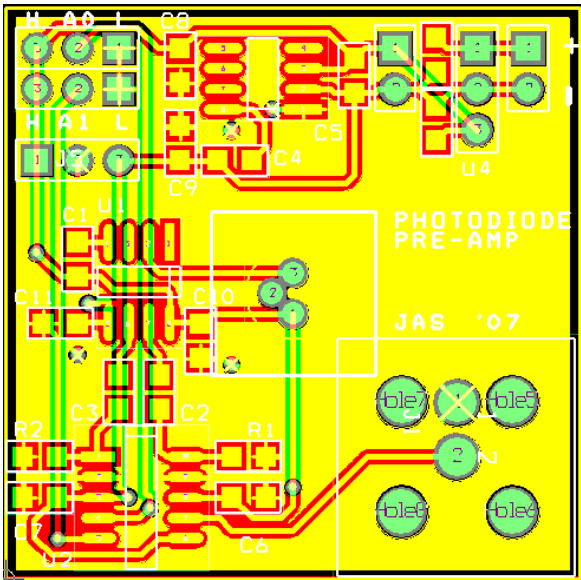
Transmitter – LED Driver



Item	Quantity	Value	Description	Part Reference
1	1	CAP NP		C1
2	1	CAP NP		C2
3	1	CAP NP		C3
4	1	CAP NP		C4
5	1	CAP NP		C5
6	1	LED		D1
7	1	LED		D2
8	1	HEADER 13X2		J1
9	1	HEADER 2		J2
10	1	HEADER 3		J11
11	1	HEADER 3		J12
12	1	HEADER 3		J13
13	1	HEADER 3		J21
14	1	HEADER 3		J22
15	1	HEADER 3		J23
16	1	HEADER 2		J24
17	1	78XX/TO		U1
18	1	74HC08		U2
19	1	74HC08		U3

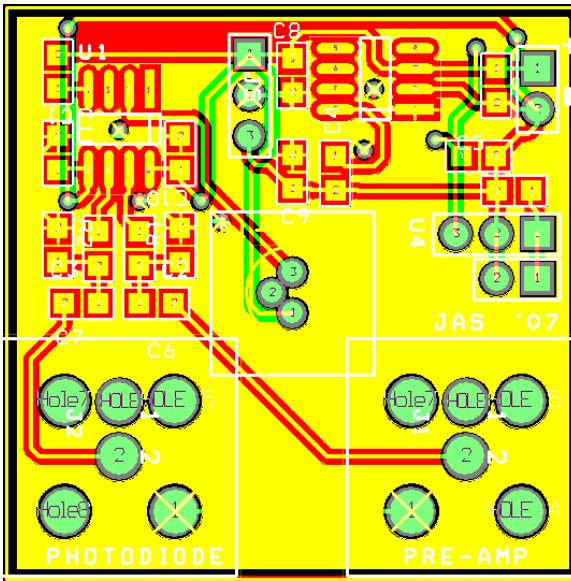
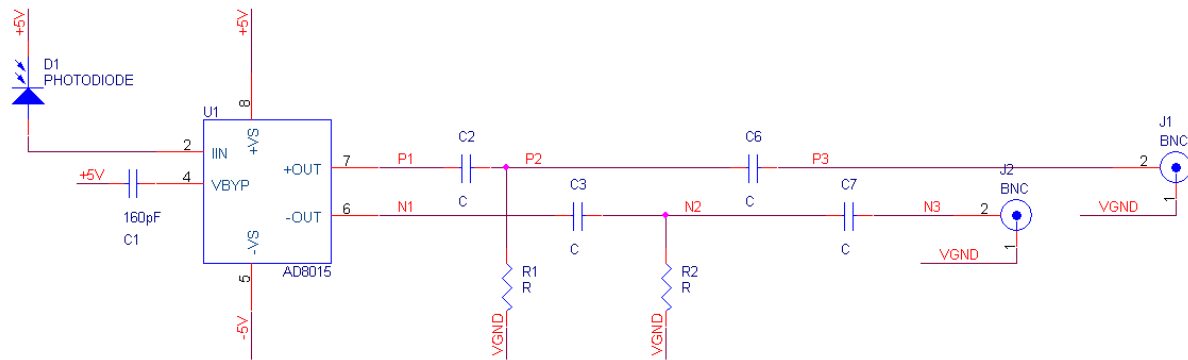
Receiver – Pre-Amp Rev 1 Schematic





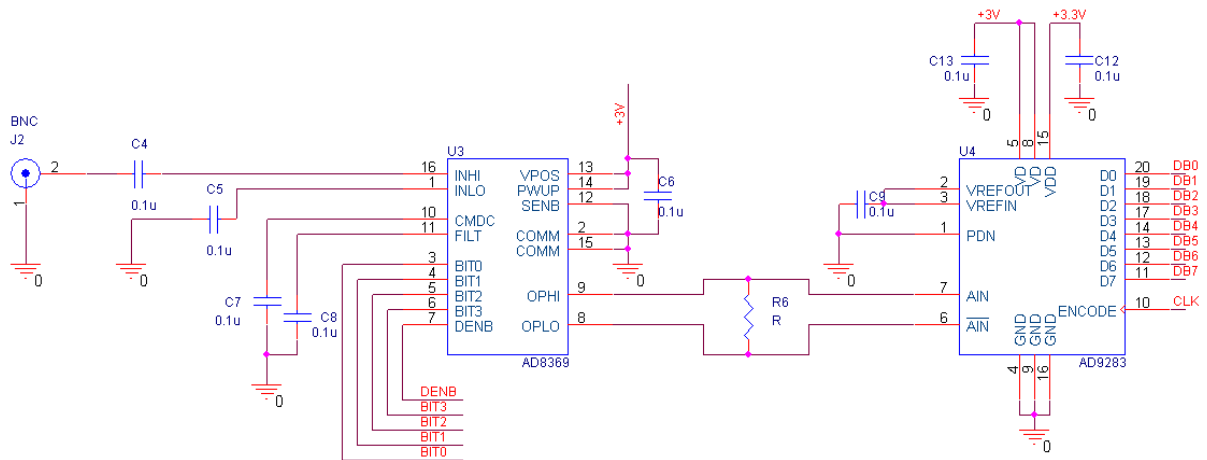
Item	Qty	Value	Description	Part Ref
	1	160pF		C1
	2	C		C2
	3	C		C3
	4	1u		C4
	5	CAP		C5
	6	0.1u		C6
	7	0.1u		C7
	8	10u TANT		C8
	9	10u TANT		C9
	10	0.1u		C10
	11	0.1u		C11
	12	0.1u		C12
	13	0.1u		C13
	14	BNC		J1
	15	JMP1		J2
	16	JMP0		J3
	17	SINGLE PWR		J4
	18	DUAL PWR		J5
	19	TO-18		J6
	20	PWR		J7
	21	R		R1
	22	R		R2
	23	AD8015	IC,TRANSIMPEDANCE AMP,DIFFERENTIAL OUTPUT, SOIC8	U1
	24	AD8250		U2
	25	TLE2426	GROUND GENERATOR	U3
	26	78XX/TO		U4

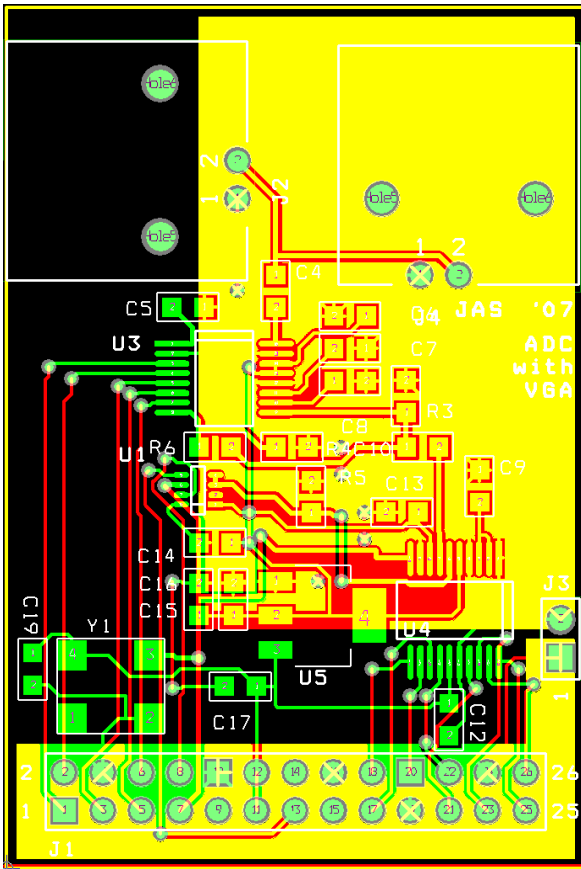
Receiver – Pre-Amp Rev 2 Schematic



Item	Quantity	Value	Description	Part Reference
			IC, TRANSIMPEDANCE AMP, DIFFERENTIAL OUTPUT, SOIC8	
1	1	160pF		C1
2	1	C		C2
3	1	C		C3
4	1	1u		C4
5	1	CAP		C5
6	1	C		C6
7	1	C		C7
8	1	10u TANT		C8
9	1	10u TANT		C9
10	1	0.1u		C10
11	1	0.1u		C11
12	1	0.1u		C12
13	1	0.1u		C13
14	1	BNC		J1
15	1	BNC		J2
16	1	SINGLE PWR		J4
17	1	DUAL PWR		J5
18	1	TO-18		J6
19	1	PWR		J7
20	1	R		R1
21	1	R		R2
22	1	AD8015		U1
23	1	TLE2426	GROUND GENERATOR	U3
24	1	78XX/TO		U4

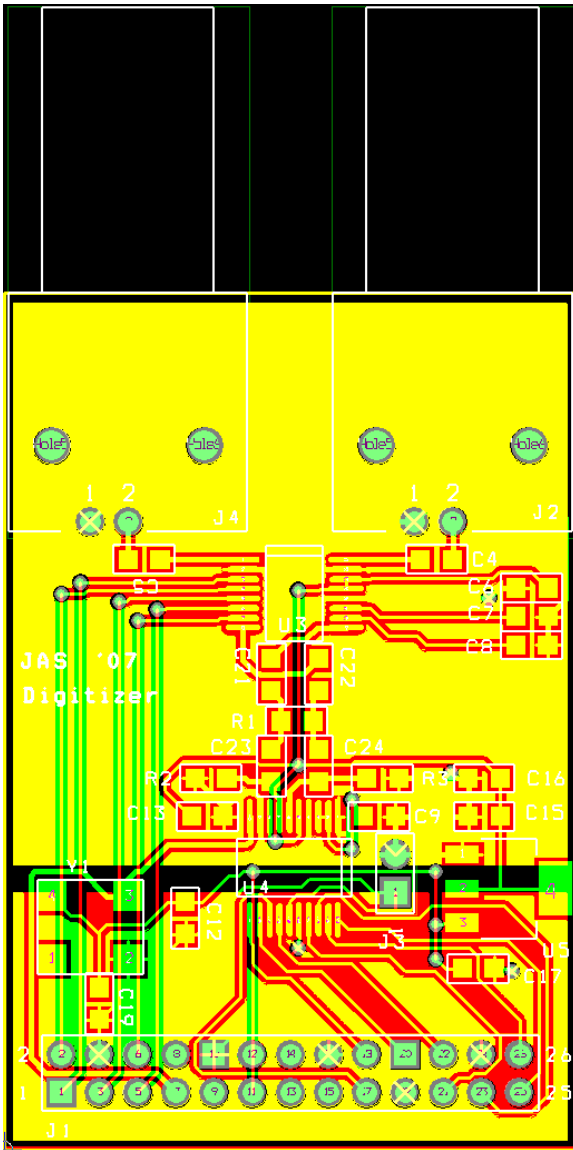
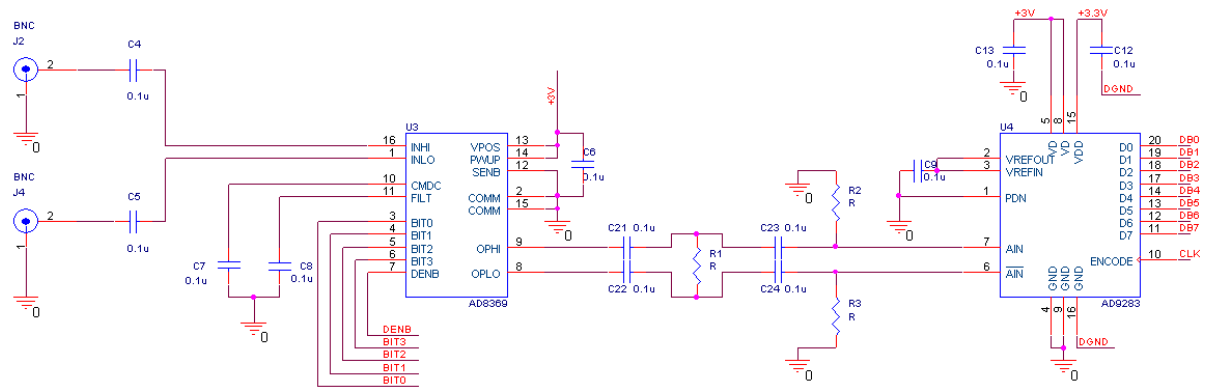
Receiver – Digitizer Rev 1 Schematic





Item	Qty	Value	Description	Part Ref
1	1	0.1u		C4
2	1	0.1u		C5
3	1	0.1u		C6
4	1	0.1u		C7
5	1	0.1u		C8
6	1	0.1u		C9
7	1	0.1u		C10
8	1	0.1u		C12
9	1	0.1u		C13
10	1	0.1u		C14
11	1	10u TANT		C15
12	1	1u		C16
13	1	1u		C17
14	1	0.01u		C19
15	1	HEADER 13X2		J1
16	1	BNC		J2
17	1	HEADER 2		J3
18	1	BNC		J4
19	1	R		R3
20	1	R		R4
21	1	R		R5
22	1	R		R6
23	1	AD5660	IC,D/A CONV,16-BIT,8 PIN, SOT-23	U1
24	1	AD8369	IC,VAR GAIN AMP,16 PIN, TSSOP	U3
25	1	AD9283	IC,A/D CONV,8-BIT,3V,100MSPS, SSOP20	U4
26	1	ADP3338	IC,VOLT REG,LDO,3.3V,1A, SOT-223-3	U5
27	1	CWX813-10.0M		Y1

Receiver – Digitizer Rev 2 Schematic



Item	Quantity	Value	Description	Part Reference
1	1	0.1u		C4
2	1	0.1u		C5
3	1	0.1u		C6
4	1	0.1u		C7
5	1	0.1u		C8
6	1	0.1u		C9
7	1	0.1u		C12
8	1	0.1u		C13
9	1	10u TANT		C15
10	1	1u		C16
11	1	1u		C17
12	1	0.01u		C19
13	1	0.1u		C21
14	1	0.1u		C22
15	1	0.1u		C23
16	1	0.1u		C24
17	1	HEADER 13X2		J1
18	1	BNC		J2
19	1	HEADER 2		J3
20	1	BNC		J4
21	1	R		R1
22	1	R		R2
23	1	R		R3
24	1	AD8369	IC,VAR GAIN AMP,16 PIN, TSSOP	U3
25	1	AD9283	IC,A/D CONV,8-BIT,3V,100MSPS, SSOP20	U4
26	1	ADP3338	IC,VOLT REG,LDO,3.3V,1A, SOT-223-3	U5
27	1	CWX813-10.0M		Y1

FPGA Verilog Code

Transmitter

```
module FX2_TX(  
    FX2_CLK, FX2_FD, FX2_SLRD, FX2_SLWR, FX2_flags,  
    FX2_PA_2, FX2_PA_3, FX2_PA_4, FX2_PA_5, FX2_PA_6, FX2_PA_7,  
    DB, ADC_CLK, P_64, P_57, data_1_out, sync_line, LED1, LED2,  
    out_byte1  
);  
  
input FX2_CLK;  
inout [7:0] FX2_FD;  
input [2:0] FX2_flags;  
output FX2_SLRD, FX2_SLWR;  
  
//output FX2_PA_0;  
//output FX2_PA_1;  
output FX2_PA_2;  
output FX2_PA_3;  
output FX2_PA_4;  
output FX2_PA_5;  
output FX2_PA_6;  
input FX2_PA_7;  
  
input [7:0] DB;  
input ADC_CLK;  
output P_64;  
output P_57;  
output data_1_out;  
output sync_line;  
output LED1;  
output LED2;  
  
output [7:0] out_byte1;
```

```

////////////////////////////////////
// Rename "FX2" ports into "FIFO" ports, to give them more meaningful //
names
// FX2 USB signals are active low, take care of them now
// Note: You probably don't need to change anything in this section

// FX2 outputs
wire FIFO_CLK = FX2_CLK;

wire FIFO2_empty = ~FX2_flags[0];
wire FIFO2_data_available = ~FIFO2_empty;
wire FIFO3_empty = ~FX2_flags[1];
wire FIFO3_data_available = ~FIFO3_empty;
wire FIFO4_full = ~FX2_flags[2];
wire FIFO4_ready_to_accept_data = ~FIFO4_full;
wire FIFO5_full = ~FX2_PA_7;
wire FIFO5_ready_to_accept_data = ~FIFO5_full;
//assign FX2_PA_0 = 1'b1;
//assign FX2_PA_1 = 1'b1;
assign FX2_PA_3 = 1'b1;

// FX2 inputs
wire FIFO_RD, FIFO_WR, FIFO_PKTEND, FIFO_DATAIN_OE, FIFO_DATAOUT_OE;
wire FX2_SLRD = ~FIFO_RD;
wire FX2_SLWR = ~FIFO_WR;
assign FX2_PA_2 = ~FIFO_DATAIN_OE;
assign FX2_PA_6 = ~FIFO_PKTEND;

wire [1:0] FIFO_FIFOADR;
assign {FX2_PA_5, FX2_PA_4} = FIFO_FIFOADR;

// FX2 bidirectional data bus
wire [7:0] FIFO_DATAIN = FX2_FD;
wire [7:0] FIFO_DATAOUT;
assign FX2_FD = FIFO_DATAOUT_OE ? FIFO_DATAOUT : 8'hZZ;

```

```

////////////////////////////////////
// So now everything is in positive logic
//   FIFO_RD, FIFO_WR, FIFO_DATAIN, FIFO_DATAOUT, FIFO_DATAIN_OE,
//   FIFO_DATAOUT_OE, FIFO_PKTEND, FIFO_FIFOADR
//   FIFO2_empty, FIFO2_data_available
//   FIFO3_empty, FIFO3_data_available
//   FIFO4_full, FIFO4_ready_to_accept_data
//   FIFO5_full, FIFO5_ready_to_accept_data

////////////////////////////////////
// Here we wait until we receive some data
// We count the number of bytes received and we send that count back
//parameter [1:0] CLK_31_cnt_bit = 9;

////////////////////////////////////
// Here we generate the required clocks
reg [6:0] clock_divisor = 0;
wire new_clk; //16 MHz data output clock, downsample it to get 1MHz
PLL20 PLLclk20(.inclk0(FIFO_CLK), .c0(new_clk)); //16MHz output
always @(posedge new_clk) clock_divisor <= clock_divisor + 1;
wire Data_18Clk = clock_divisor[6];
wire Data_1Clk = clock_divisor[3];

reg [7:0] buffer [0:1023];
reg [9:0] write_pos = 0;
reg [9:0] read_pos = 0;
reg [7:0] pkt_cnt = 0;
reg [7:0] out_bytel = 0;
reg [2:0] serial_cnt = 0;
reg data_1_out;

```

```

always @(posedge Data_1Clk)
// Data_1Clk runs 8 times faster than Data_18Clk
begin
    out_byte1 = buffer[read_pos];
    if (state_31 != 3'b000)
    // Don't start Txing bits till we get a start sig. from USB
    begin
        data_1_out <= out_byte1[serial_cnt];
        // Send each bit of a bit out IO pin
        serial_cnt = serial_cnt + 1;
        // Iterate over 8 bits
    end
end

reg [2:0] state_31 = 3'b000;
always @(posedge Data_18Clk)
begin
case(state_31)
    3'b000: if (start) state_31<=3'b100;
        // Wait for start from SM below
    3'b100: begin read_pos <= read_pos+1'b1;
            if (read_pos==511)
            begin pkt_cnt<=pkt_cnt+1'b1;
                state_31<=3'b101;
            end end
            // Read frombuffer 0 - 511, write to data_1 (all zeros)
    3'b101: begin read_pos <= read_pos+1'b1;
            if (read_pos==1023)
            begin pkt_cnt<=pkt_cnt+1'b1;
                state_31<=3'b110;
            end end
            // Read from buffer 512 - 1023 and write to data_1
    3'b110: begin read_pos <= read_pos+1'b1;

```

```

        if (read_pos==511) begin pkt_cnt<=pkt_cnt+1'b1;
        state_31<=3'b101;
        end end
        // Read from buffer 0 - 511 and write to data_1
    default: state_31 <= 3'b000;
endcase
end

reg start = 1'b0;
reg [2:0] state = 3'b000;
always @(posedge FIFO_CLK)
case(state)
    3'b000: if( FIFO2_data_available) state <= 3'b001;
            // wait for data packet in FIFO2
    3'b001: if(~FIFO2_data_available) state <= 3'b010;
            // wait until end of data packet
    3'b010: begin
                start <= 1'b1;
                if (state_chg2==3'b101) state<=3'b100;
            end
    3'b100: begin
            //Read from USB and write data into buffer 0 - 511
                if((write_pos==511)) state<=3'b101;
                // go to state 101 (below)
                //else if(1)
                if(FIFO2_data_available)
                begin
                    write_pos <= write_pos+1'b1;
                    buffer[write_pos] <= FIFO_DATAIN;
                end
            end
    3'b101: if(state_chg2==3'b110)
            // wait for buffer read state to change
            begin
                state<=3'b110;

```



```

        write_pos <= 512;
    end
3'b110: begin
    // Read from USB and write data in to buffer 512 - 1023
    if((write_pos==1023)) state<=3'b111;
    // go to wait state below
    if(FIFO2_data_available)
    begin
        write_pos <= write_pos+1'b1;
        buffer[write_pos] <= FIFO_DATAIN;
    end
    end
3'b111: if(state_chg2==3'b101)
    //wait for buffer read state to change
    begin
        state<=3'b100;
        write_pos <= 0;
    end
    default: state <= 3'b000;
endcase

reg [2:0] state_chg1;
always @(posedge FIFO_CLK) state_chg1 <= state_31;
reg [2:0] state_chg2;
always @(posedge FIFO_CLK) state_chg2 <= state_chg1;

assign FIFO_RD = (state==3'b001 | state==3'b100 | state==3'b110);
assign FIFO_WR = 1'b0;
assign FIFO_DATAIN_OE = 1'b1;
assign FIFO_DATAOUT_OE = 1'b0;
assign FIFO_FIFOADR = {1'b0, 1'b0};
assign FIFO_PKTEND = 1'b0;

assign sync_line = (state_31 == 3'b110);

```

```

//Chose at what point the start signal is asserted

assign LED1 = start;
assign LED2 = state[0];

endmodule

```

Receiver

```

module FX2_RX(
    FX2_CLK, FX2_FD, FX2_SLRD, FX2_SLWR, FX2_flags,
    FX2_PA_2, FX2_PA_3, FX2_PA_4, FX2_PA_5, FX2_PA_6, FX2_PA_7,
    DB, ADC_CLK, P_64, P_57, LED1, LED2,
    VGA_BIT0, VGA_BIT1, VGA_BIT2, VGA_BIT3, VGA_DENB
);

input FX2_CLK;
inout [7:0] FX2_FD;
input [2:0] FX2_flags;
output FX2_SLRD, FX2_SLWR;

//output FX2_PA_0;
//output FX2_PA_1;
output FX2_PA_2;
output FX2_PA_3;
output FX2_PA_4;
output FX2_PA_5;
output FX2_PA_6;
input FX2_PA_7;

input [7:0] DB;
input ADC_CLK;
input P_64;
output P_57;
output LED1;
output LED2;

```

```

output VGA_BIT0, VGA_BIT1, VGA_BIT2, VGA_BIT3, VGA_DENB;

////////////////////////////////////
// Rename "FX2" ports into "FIFO" ports, to give them more meaningful //
names
// FX2 USB signals are active low, take care of them now
// Note: You probably don't need to change anything in this section

// FX2 outputs
wire FIFO_CLK = FX2_CLK;

wire FIFO2_empty = ~FX2_flags[0];
wire FIFO2_data_available = ~FIFO2_empty;
wire FIFO3_empty = ~FX2_flags[1];
wire FIFO3_data_available = ~FIFO3_empty;
wire FIFO4_full = ~FX2_flags[2];
wire FIFO4_ready_to_accept_data = ~FIFO4_full;
wire FIFO5_full = ~FX2_PA_7;
wire FIFO5_ready_to_accept_data = ~FIFO5_full;
//assign FX2_PA_0 = 1'b1;
//assign FX2_PA_1 = 1'b1;
assign FX2_PA_3 = 1'b1;

// FX2 inputs
wire FIFO_RD, FIFO_WR, FIFO_PKTEND, FIFO_DATAIN_OE, FIFO_DATAOUT_OE;
wire FX2_SLRD = ~FIFO_RD;
wire FX2_SLWR = ~FIFO_WR;
assign FX2_PA_2 = ~FIFO_DATAIN_OE;
assign FX2_PA_6 = ~FIFO_PKTEND;

wire [1:0] FIFO_FIFOADR;
assign {FX2_PA_5, FX2_PA_4} = FIFO_FIFOADR;

```

```

// FX2 bidirectional data bus
wire [7:0] FIFO_DATAIN = FX2_FD;
wire [7:0] FIFO_DATAOUT;
assign FX2_FD = FIFO_DATAOUT_OE ? FIFO_DATAOUT : 8'hZZ;

////////////////////////////////////
So now everything is in positive logic
//     FIFO_RD, FIFO_WR, FIFO_DATAIN, FIFO_DATAOUT, FIFO_DATAIN_OE,
//     FIFO_DATAOUT_OE, FIFO_PKTEND, FIFO_FIFOADR
//     FIFO2_empty, FIFO2_data_available
//     FIFO3_empty, FIFO3_data_available
//     FIFO4_full, FIFO4_ready_to_accept_data
//     FIFO5_full, FIFO5_ready_to_accept_data

////////////////////////////////////
Here we wait until we receive some data
// We count the number of bytes received and we send that count back

reg [7:0] buffer [0:8191];
reg [12:0] write_pos = 0;
reg [12:0] read_pos = 0;
reg [7:0] pkt_cnt = 0;

assign FIFO_DATAOUT = buffer[read_pos];

reg [2:0] state_31 = 3'b000;
always @(posedge ADC_CLK)
begin
if((write_pos==0)|| (write_pos==4096)) buffer[write_pos] <= pkt_cnt;
else buffer[write_pos] <= DB; //DB
case(state_31)
    3'b000: if (start2) state_31<=3'b100;
    3'b100: begin write_pos <= write_pos+1'b1;
                if (write_pos==4095) begin pkt_cnt<=pkt_cnt+1'b1;

```

```

        state_31<=3'b101; end end //000
    3'b101: begin write_pos <= write_pos+1'b1;
        if (write_pos==8191) begin pkt_cnt<=pkt_cnt+1'b1;
            state_31<=3'b110; end end //010
    3'b110: begin write_pos <= write_pos+1'b1;
        if (write_pos==4095) begin pkt_cnt<=pkt_cnt+1'b1;
            state_31<=3'b101; end end //100
    default: state_31 <= 3'b000;
endcase
end

wire start2;
assign start2 = (start && P_64)? 1:0;
reg [3:0] data_in = 0;

reg start = 1'b0;
reg [2:0] state = 3'b000;
always @(posedge FIFO_CLK)
case(state)
    3'b000: if( FIFO2_data_available) state <= 3'b001;
        // wait for data packet in FIFO2
    3'b001: begin {data_in,VGA_Gain} <= FIFO_DATAIN;
        if(~FIFO2_data_available) state <= 3'b010; end
        // wait until end of data packet
    3'b010: begin
            start <= 1'b1;
            if (state_chg2==3'b101) state<=3'b100;
        end
    3'b100: begin
        //0 - 4095
            if((read_pos==4095)) state<=3'b101;
            else if(FIFO4_ready_to_accept_data)
                read_pos <= read_pos+1'b1;
        end
    3'b101: if((state_chg2==3'b110)&&FIFO4_ready_to_accept_data)

```

```

        // wait
        begin
            state<=3'b110;
            read_pos <= 4096;
        end
3'b110: begin
    // 4096 - 8191
        if((read_pos==8191)) state<=3'b111;
        else if(FIFO4_ready_to_accept_data)
            read_pos <= read_pos+1'b1;
        end
3'b111: if((state_chg2==3'b101)&&FIFO4_ready_to_accept_data)
    //wait
        begin
            state<=3'b100;
            read_pos <= 0;
        end
    default: state <= 3'b000;
endcase

reg [2:0] state_chg1;
always @(posedge FIFO_CLK) state_chg1 <= state_31;
reg [2:0] state_chg2;
always @(posedge FIFO_CLK) state_chg2 <= state_chg1;

assign FIFO_RD = (state==3'b001);
assign FIFO_WR = ((state==3'b100) || (state==3'b110));
assign FIFO_DATAIN_OE = ((state==3'b000) || (state==3'b001));
assign FIFO_DATAOUT_OE = ((state==3'b100) || (state==3'b110));
assign FIFO_FIFOADR = {state[2], 1'b0};
assign FIFO_PKTEND = 1'b0;

assign P_57 = pkt_cnt[0];
assign LED1 = ~start;

```

```

assign LED2 = ~start2;

////////////////////////////////////
Here we assign the VGA gain

reg [3:0] VGA_Gain = 0;
assign {VGA_BIT3,VGA_BIT2,VGA_BIT1,VGA_BIT0} = VGA_Gain;
assign VGA_DENB = 1'b1;

endmodule

```

PC C++ Code

Transmitter

```

#include <windows.h>
#include <assert.h>
#include <stdio.h>
#include <conio.h>
HANDLE XyloDeviceHandle;

////////////////////////////////////
void USB_Open()
{
    XyloDeviceHandle = CreateFile("\\\\.\\EzUSB-0", GENERIC_WRITE,
    FILE_SHARE_WRITE, NULL, OPEN_EXISTING, 0, NULL);
    assert(XyloDeviceHandle!=INVALID_HANDLE_VALUE);
}

void USB_Close()
{
    CloseHandle(XyloDeviceHandle);
}

////////////////////////////////////
void USB_BulkWrite(DWORD pipe, void* buffer, DWORD buffersize)
{
    DWORD nBytes;
    assert(buffersize<0x10000);
    DeviceIoControl(XyloDeviceHandle, 0x222051, &pipe, sizeof(pipe),
    buffer, buffersize, &nBytes, NULL);

    assert(nBytes==buffersize);    // make sure everything was sent
}

DWORD USB_BulkRead(DWORD pipe, void* buffer, DWORD buffersize)
{

```

```

    DWORD nBytes;
    assert(bufferSize<0x10000);
    DeviceIoControl(XyloDeviceHandle, 0x22204E, &pipe, sizeof(pipe),
    buffer, bufferSize, &nBytes, NULL);

    return nBytes;
}

////////////////////////////////////
void main()
{

    FILE * pFile;
    pFile = fopen ("../../../../../../../../Data/ToTransmitter.txt","rb");
    //Open the file in binary mode - IMPORTANT

    unsigned char buf[512];
    unsigned char last_buf[512];
    unsigned char clear_buf[512];

    long fileSize = 0;
    int lastPacketCnt = 0;
    int j = 0;
    int i = 0;

    fseek(pFile,0,SEEK_END);
    fileSize = ftell(pFile);
    rewind(pFile);

    printf("File Size is %d bytes",fileSize);
    lastPacketCnt = (fileSize / 512);

    for (i = 0; i<512; i++) {
        buf[i] = 0;
        last_buf[i] = 0;
        clear_buf[i] = 0x00;
    }

    Sleep(500);
    printf("Running...\n");
    USB_Open();

    j=0;

    fseek(pFile,4,SEEK_SET);
    USB_BulkWrite(2, buf, 5);    // Send a start command

    while(!feof(pFile))
    {
        if (j < lastPacketCnt) {
            fread(buf, 1, 512, pFile);
            USB_BulkWrite(2, buf, sizeof(buf));
        } else {

```



```

        fread(last_buf, 1, 512, pFile);
        USB_BulkWrite(2, last_buf, sizeof(last_buf));
    }

    printf("%d\n", j);
    j++;
}

USB_BulkWrite(2, clear_buf, 512);
//Send 1024 bytes of '0'
USB_BulkWrite(2, clear_buf, 512);
// This assures the TX stops
USB_Close();

Beep(1000,100);
Beep(750,100);
Beep(500,100);
}

```

Receiver

```

#include <windows.h>
#include <assert.h>
#include <stdio.h>
#include <conio.h>

HANDLE XyloDeviceHandle;

////////////////////////////////////
// Open and close the USB driver
void USB_Open()
{
    XyloDeviceHandle = CreateFile("\\\\.\\EzUSB-0", GENERIC_WRITE,
FILE_SHARE_WRITE, NULL, OPEN_EXISTING, 0, NULL);
    assert(XyloDeviceHandle!=INVALID_HANDLE_VALUE);
}

void USB_Close()
{
    CloseHandle(XyloDeviceHandle);
}

////////////////////////////////////
// USB functions to send and receive bulk packets
void USB_BulkWrite(DWORD pipe, void* buffer, DWORD buffersize)
{
    DWORD nBytes;
    assert(buffersize<0x10000);
    DeviceIoControl(XyloDeviceHandle, 0x222051, &pipe, sizeof(pipe),
buffer, buffersize, &nBytes, NULL);
}

```

```

        assert(nBytes==buffersize); // make sure everything was sent
    }

DWORD USB_BulkRead(DWORD pipe, void* buffer, DWORD buffersize)
{
    DWORD nBytes;
    assert(buffersize<0x10000);
    DeviceIoControl(XyloDeviceHandle, 0x22204E, &pipe, sizeof(pipe),
buffer, buffersize, &nBytes, NULL);
    //assert(nBytes==buffersize);
    return nBytes;
}

////////////////////////////////////

void main()
{
    FILE * pFile1;
    FILE * pFile2;
    FILE * pFile3;
    pFile1 = fopen ("FromReceiver.txt","w");
    pFile2 = fopen ("PacketsLost.txt","w");
    pFile3 = fopen ("VGA_Gain.txt", "rb");

    unsigned char buf[4096];
    unsigned char buf_spacer[4096];
    int nb_bytes_received;
    int j = 0;
    int i = 0;
    bool k = true;
    unsigned int avg = 0;
    unsigned char buf_avg[600];
    unsigned char PacketsLost[1];

    int pkt_cnt_prev = -1;
    int pkt_cnt_lost = 0;
    int pkt_cnt_diff = 0;

    char gain[10];

    //Initialize buf spacer
    for (i=0; i<sizeof(buf_spacer); i++)
    {
        buf_spacer[i] = 0;
    }

    Sleep(1000);
    USB_Open();
    Beep(500,100);
    Beep(750,100);
    Beep(1000,150);
    Sleep(1000);
}

```

```

buf[0] = 0x00;
fscanf(pFile3, "%d", &gain);
buf[0] = gain[0];
printf("Setting gain to: %d\n", buf[0]);
USB_BulkWrite(2, buf, 1);

printf("Waiting for start2...\n");
j=0;

// coded - use 322 for j
// uncoded - use 161 for j
while(j<322)
{
    nb_bytes_received = USB_BulkRead(4, buf, sizeof(buf));

    if(k)
    {
        printf("Receiving... %d\n", nb_bytes_received);
        k = false;
    }
    if(nb_bytes_received!=4096)
    {
        printf("ERROR... %d\n", nb_bytes_received);
    }
    pkt_cnt_diff = buf[0] - pkt_cnt_prev;
    if(pkt_cnt_diff>=0)
    {
        pkt_cnt_lost = pkt_cnt_lost + (pkt_cnt_diff-1);
    }
    else
    {
        pkt_cnt_lost = pkt_cnt_lost + (pkt_cnt_diff+255);
    }
    pkt_cnt_prev = buf[0];
    buf[0] = buf[1];
    fwrite (buf , sizeof(buf[0]) , sizeof(buf) , pFile1 );

    j++;
}

PacketsLost[0] = (unsigned char)pkt_cnt_lost;
fwrite
(PacketsLost, sizeof(PacketsLost[0]), sizeof(PacketsLost), pFile2);

USB_Close();
printf("Packets lost: %d\n", pkt_cnt_lost);
if(pkt_cnt_lost!=0) { printf("Press a key to exit"); getch(); }
}

```