

University of Southern Queensland  
Faculty of Health, Engineering and Sciences

# Light Fidelity (Li-Fi) Prototype with Raspberry Pi

A Dissertation submitted by

Paul Fergusson

In fulfilment of the requirements of

ENG4111 & ENG4112 Research Project

Towards the degree of

Bachelor of Engineering (Honours) (Electrical and Electronics)

Submitted October 2016

# ABSTRACT

With globalisation and the thirst for connectivity across society, the demand placed on wireless infrastructure and the associated resource is growing exponentially. Very soon this resource will reach saturation point, due to the finite bandwidth available in the Radio Frequency (RF) spectrum. A method of countering the impending saturation needs to be found. That method can be Visible Light Communication (VLC).

Light Fidelity (Li-Fi) is a research field within VLC that utilises the visible light band within the electromagnetic wave spectrum. This band is 10,000 times larger than the RF band and cannot be ‘leased’ or saturated with users. Light waves can be modulated to carry an enormous amount of simultaneous data, at speeds faster than current consumer equipment can handle.

This Dissertation describes in detail the research, construction and testing of a Li-Fi prototype using Raspberry Pi. The prototype is compact, low cost, uses accessible components and provides a solid foundation for other students to follow on with further work in this field.

The prototype successfully demonstrates the principle of Visible Light Communication and shows the viability of using Python for coding, SPI for data transfer and lists suitable electronic components to process bit-wise data signals. The prototype shows that while it is possible to use addressable LED's as the transmitting element, the Dissertation concludes that they are not suitable outside of a heavily constrained environment.

University of Southern Queensland  
Faculty of Health, Engineering and Sciences

## ENG4111 & ENG4112 Research Project

### **Limitations of Use**

The Council of the University of Southern Queensland, its Faculty of Health, Engineering and Sciences, and the staff of the University of Southern Queensland, do not accept any responsibility for the truth, accuracy or completeness of the material contained within or associated with this project preliminary report.

Persons using all or any part of this material do so at their own risk, and not at the risk of the Council of the University of Southern Queensland, its Faculty of Health, Engineering and Sciences, or the staff of the University of Southern Queensland.

This project preliminary report reports an educational exercise and has no purpose or validity beyond this exercise. The sole purpose of the course pair entitles “Research Project” is to contribute to the overall education within the student’s chosen degree program. This document, the associated hardware, software, drawings, and any other material set out in the associated appendices should not be used for any other purpose: if they are so used, it is entirely at the risk of the user.

## CERTIFICATION

I certify that the ideas, designs and experimental work, results, analyses and conclusions set out in this project preliminary report are entirely my own effort, except where otherwise indicated and acknowledged.

I further certify that the work is original and has not been previously submitted for assessment in any other course or institution, except where specifically stated.

Paul Fergusson

Student Number: U1012000

## **ACKNOWLEDGEMENTS**

I would like to thank the people who have helped and supported me throughout my academic journey.

Firstly I would like to thank my family for being patient, thoughtful, tolerant and supportive during times of stress and pressure. Without such an understanding and considerate family this process would not have been possible, nor would I have enjoyed the success I have reached.

Secondly I would like to thank the teaching staff of the University of Southern Queensland, without their dedication and devotion, the external student programs of the University of Southern Queensland would not be the successful institution they are today. Their commitment and sacrifice is a cornerstone to building professional and competent engineers.

Lastly I would like to thank my supervisor, A/Prof Alexander Kist. Without his insight, discipline and technical support I would not have decided to take on the challenges presented by this project.

# TABLE OF CONTENTS

<b>Abstract.....</b>	<b>i</b>
<b>Certification.....</b>	<b>iii</b>
<b>Acknowledgements .....</b>	<b>iv</b>
<b>List of Figures.....</b>	<b>viii</b>
<b>List of Tables .....</b>	<b>ix</b>
<b>1    Introduction.....</b>	<b>1</b>
1.1    Current Situation.....	1
1.2    Project Aim .....	2
1.3    Project Scope .....	2
1.4    Objectives & Expected Outcomes .....	2
1.5    Critical Analysis.....	3
1.6    Result Interpretation.....	3
<b>2    Literature Review .....</b>	<b>4</b>
2.1    Visible Light Communication.....	4
2.2    Raspberry Pi.....	5
2.3    Transmitter.....	7
2.4    Receiver .....	10
2.5    Signal Management .....	11
2.6    Signal Modulation.....	12
2.7    Encoding/Decoding.....	13
2.8    Existing Prototypes .....	13
2.9    Knowledge Gap .....	14
<b>3    Project Methodology.....</b>	<b>15</b>
3.1    Overview.....	15
3.2    Initialisation .....	15
3.3    Start-Up.....	15
3.4    Production.....	16
3.5    Execution .....	16
3.6    Project Closure.....	16
<b>4    Prototype Development .....</b>	<b>17</b>
4.1    Project System Diagram.....	17
4.2    Transmission Module.....	18
4.2.1    Raspberry Pi.....	18
4.2.2    LED Circuit.....	24

4.2.3	Programming & Code .....	27
4.2.4	Cost .....	28
4.3	Receiver Module .....	29
4.3.1	Raspberry Pi.....	29
4.3.2	Photodiode Circuit .....	29
4.3.3	Programming and Code .....	33
4.3.4	Cost .....	34
4.4	Prototype System .....	35
<b>5</b>	<b>Results Analysis.....</b>	<b>36</b>
5.1	Analysis Approach.....	36
5.2	Tooling.....	37
5.3	Transmission Module.....	39
5.3.1	Transmission Circuit (PCB).....	39
5.3.2	Power Supply .....	40
5.3.3	GPIO Protection.....	40
5.3.4	Lighting the LED .....	40
5.3.5	Addressable LED Testing .....	43
5.3.6	File Processing .....	46
5.3.7	Transmitting File Data .....	47
5.3.8	Data Modulation .....	48
5.3.9	Raspberry Pi & MobaXterm .....	50
5.4	Receiver Module .....	51
5.4.1	Receiver Circuit (PCB) .....	51
5.4.2	Photodiode & Amplifier Testing.....	52
5.4.3	Python Coding .....	54
5.5	Prototype Module Integration .....	55
5.5.1	Distance.....	55
5.5.2	Transmitting Square Waves .....	56
5.5.3	Transmitting File Data .....	56
<b>6</b>	<b>Contributions, Conclusion &amp; Future Work.....</b>	<b>57</b>
6.1	Summary of Contributions.....	57
6.1.1	Addressable LED's .....	57
6.1.2	Raspberry Pi.....	57
6.1.3	Prototype Components.....	58
6.1.4	Construction.....	58
6.1.5	Signal Modulation.....	59

6.1.6	Python and SPI.....	59
6.2	Review of Project Objectives.....	60
6.2.1	Objectives 1, 2 and 3 – Research .....	60
6.2.2	Objective 4 – Build a Prototype.....	60
6.2.3	Objective 5 – Transmit Video .....	60
6.2.4	Objective 6 – Quantity the Prototype.....	61
6.2.5	Objective 7 – Future Work.....	61
7	<b>References.....</b>	<b>63</b>
<b>APPENDIX A – PROJECT SPECIFICATION.....</b>		<b>67</b>
<b>APPENDIX B – PYTHON CODE.....</b>		<b>68</b>
B1.	Transmitter Module Code .....	68
B2.	Receiver Module Code.....	69
<b>APPENDIX C – PROJECT MANAGEMENT .....</b>		<b>70</b>
C1.	Project Artefacts .....	70
C2.	Risk Management & Standard .....	71
<b>APPENDIX D – PRODUCT DATASHEETS .....</b>		<b>74</b>
D1.	Raspberry Pi 2B Layout Diagram.....	74
D2.	Shiji LED APA102C (DotStar) Datasheet.....	75
D3.	TI 74HCT125 Quad Level Shifter Datasheet.....	115
D4.	Hamamatsu S5973 Photodiode Datasheet .....	116
D5.	TI OPA380A Transimpedance Amplifier Datasheet .....	142

# LIST OF FIGURES

Figure 1 - Radio Frequency vs. Visible Light Frequency Bands. Courtesy of author .....	1
Figure 2 – Raspberry Pi 2B Microcontroller .....	5
Figure 3 – AdaFruit DotStar LED's. Courtesy of author.....	7
Figure 4 - Block Diagram of APA102C LED. Refer Datasheet.....	8
Figure 5 - APA102C 32 Bit Signal Components. Refer Datasheet. ....	9
Figure 6 - Table VII from Kalavally et al. LED Based Indoor Visible Light Communications; State of the Art, IEEE 2015.....	12
Figure 7 – Prototype System Diagram.....	17
Figure 8 - Raspberry Pi Zero. Courtesy Raspberry Pi Foundation. ....	19
Figure 9 - Raspberry Pi 2B Locations & GPIO. Courtesy RS Components.....	20
Figure 10 - MobaXterm Start Menu Ribbon.....	21
Figure 11 - MobaXterm Session Option Window .....	22
Figure 12 - Raspberry Pi Configuration Menu. ....	23
Figure 13 - Raspberry Pi Terminal Check for SPI.....	23
Figure 14 - Raspberry Pi Terminal Check for SpiDev.....	23
Figure 15 - LED Size Comparison. Courtesy Flexfire LED's.....	24
Figure 16 - APA102C LED Mounted to Breakout PCB.....	26
Figure 17 - Software Diagram for Transmission Module.....	27
Figure 18 - Hamamatsu S5973 Photodiode (TO-18). Refer Datasheet.....	30
Figure 19 - Connection Circuit of OPA380 Transimpedance Amplifier.....	31
Figure 20 – Texas Instruments OPA380 TIA Graph to Select R <sub>f</sub> Value .....	32
Figure 21 - GPIO Zener Protection Circuit.....	33
Figure 22 - Software Diagram for Receiver Module. ....	33
Figure 23 - Image of Li-Fi Prototype.....	35
Figure 24 - Developmental Approach for System Integration & Prototype Analysis .....	36
Figure 25 - Fluke ScopeMeter 192C.....	37
Figure 26 - Solder Paste applied to mount SOIC Components to DIP PCB's. ....	38
Figure 27 - TX Module Layout Diagram.....	39
Figure 28 - GPIO LED Data Signal with SPI .....	42
Figure 29 - SPI SCLK GPIO Output Waveform Comparison.....	43
Figure 30 – Collimator used to increase LED directivity .....	44
Figure 31 - Multiple LED connection example (Shiji, 2014) .....	45
Figure 32 - Video screen capture of 1MHz modulation .....	47
Figure 33 - Example of Long Data Sequences.....	48
Figure 34 - Differential Manchester Encoding .....	49
Figure 35 - Receiver Module Breadboard Layout .....	51
Figure 36 - Early Prototype Test Environment. ....	55

## LIST OF TABLES

Table 1 - Comparison of Present Market Micro-Controllers .....	6
Table 2 - Comparison of LED Alternatives for Transmission Module.....	9
Table 3 - Costs Associated with Construction of the Transmitter Module.....	28
Table 4 - Comparison Table of Photodiode Terminal Capacitance.....	30
Table 5 - Costs Associated with Construction of the Receiver Module .....	34
Table 6 - Results Table of TIA Feedback Resistor and Capacitor Testing.....	53

# 1 INTRODUCTION

## 1.1 Current Situation

We now live in a world that is infinitely connected through a multitude of invisible, networked pathways. They stretch and travel across houses, towns, countries and continents. We are a truly global society that now has an insatiable appetite for connection, information and convenience.

This intrinsic appetite has fuelled the proliferation of connection technology from crude military beginnings at DARPA (Dennis et al. 1988), through excruciatingly slow networks of public computers and onto networks increasing in speed and computational power, now without the use of cables.

Almost exponentially the world has developed and embraced technology that now allows us to carry super-advanced, micro-computers in our pockets. The majority of these devices are connected wirelessly to internet service providers, who in turn connect to the World Wide Web. With a swipe of our finger we can find out the weather at that exact moment in Nairobi, St Petersburg and Honolulu.

Aside from the developing detrimental effect this has on us socially, there is a price we are paying in the airwaves. This ‘wireless’ communication is carried out across the Radio Frequency (RF) band (ACMA, 2013) and limited by blocks of frequencies, each of a finite bandwidth. There will be a point in time when all of the available bands are all allocated, meaning the consumer blocks are clogged and causing ineffective and disrupted communication. Although a ‘First World Problem’ this will have a negative impact on society, with deeper exclusivity creeping into signal allocation and wireless communications.

A supplementary or superseding method of communication needs to be developed to combat the inevitable RF band saturation. There is such a method and it utilises a relatively untapped source of waves, with an extremely large bandwidth. This method uses the visible light spectrum band, which is shown (not to scale) in Figure 1 below.

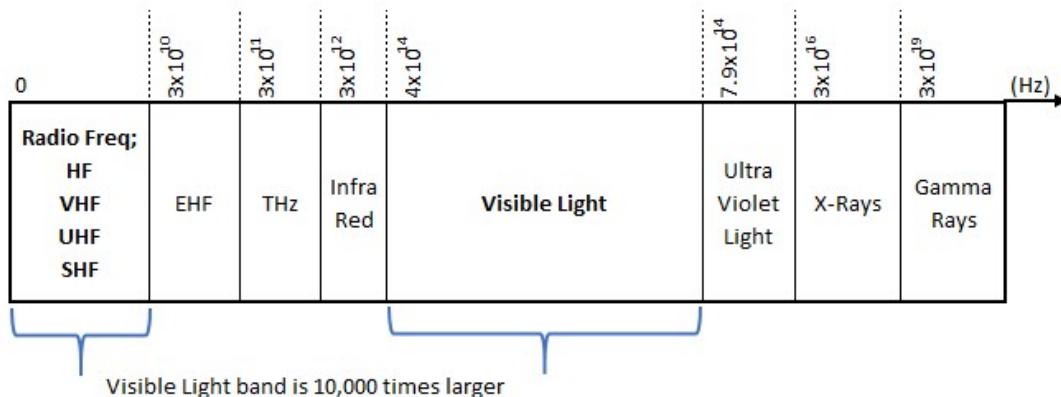


Figure 1 - Radio Frequency vs. Visible Light Frequency Bands. Courtesy of author.

It can be seen that the available electromagnetic spectrum of RF and Visible Light are disproportionately assigned in current communication methods. Not only is there larger space to grow into, it is ‘greener’, the band cannot be regulated, leased or saturated, it does not create electromagnetic interference with other devices and modulation of data can occur at frequencies that the human eye cannot detect (Stefan & Haas 2014). There is even research underway by Rajagpol et al. (2012) where data streaming is being conducted at light levels such that the light appears to be off according to the human eye. This could account for daytime or purposefully dark environments where people may use their devices.

With such a relatively new, green technology, the bounds are at the outer limits of current electrical equipment, much like Wi-Fi was when it was emerging. The applications appear endless, ranging from vehicle to vehicle communication, line of sight secure data networks, underwater communication (Rani et al. 2012) and all at communication speeds ( $>3\text{GB/s}$ ) currently not accessible to everyday society (Stefan & Haas 2014).

## 1.2 Project Aim

The aim of this project was to produce a functioning, ‘proof of concept’ prototype that utilises VLC technology to send information to a related device across free space. The prototype should be low cost, compact and simple to implement.

## 1.3 Project Scope

The scope of the project was to research, develop, build and test a prototype system that would modulate Light Emitting Diodes (LED) such that the integrity and quality of information was preserved across free space and in turn received, then displayed on a suitable device.

A Raspberry Pi microcontroller could be incorporated to allow a compact, yet powerful tool to handle the system demands. This could help to put the prototype components within low budget reach of other students, consumers and researchers.

One of the foundations of the project was to enhance research in the area of Light Fidelity (Li-Fi) such that a wider, functioning architecture could be developed and implemented domestically, commercially or where the benefits of Li-Fi could be realised.

**The scope of this project did not include;**

- Streaming audio and video signals,
- Bi-directional Li-Fi communication between the devices,
- Integration with Wi-Fi and internet services,
- Configuration of the video to interface with a mobile or portable device,
- Creation of an App to provide a User Interface, or
- Multi-channel communication.

## 1.4 Objectives & Expected Outcomes

In order to arrive at the final aim, there were a number of key objectives to satisfy during the project. These included:

1. Undertake a literature review on visible light communication, modulation and signal conditioning.
2. Undertake a basic requirements analysis
3. Critically evaluate alternative implementations and identify suitable components for the prototype implementation.
4. Build a basic prototype with physical components and achieve high speed square wave transmission.
5. Evaluate the performance and investigate improved modulation techniques for video transmission. Transmit basic video and improve to HD (720p).
6. Measure distance and video quality to define optimal prototype parameters.
7. Identify future research direction.

## 1.5 Critical Analysis

The prototype has a reasonably ‘black and white’ criteria for personal success. That is, if a file can be transmitted via modulation of LED’s across free space, then the prototype is largely a success.

To focus the level of success into an academic sense, once a file is confirmed as sent and received, increasing the size of the file will be tested to determine the maximum level achievable with this set-up. This will be critical in determining the industrial usability of the prototype and also give other academics a base to work up from. Also changing to other file types should be considered.

Coupled with the size of the data signals that can be transmitted, the distance that the signal remains useable is also critical in any application consideration. Once the signal is working, I propose to increase the distance between the transmitter and receiver modules and testing signal size at each increment. This should result in a table of distance vs quality for the prototype, that can be objectively used assess performance.

## 1.6 Result Interpretation

To reach the conclusions for the project dissertation, a large step will be to get the prototype working. Should it not work, I will make recommendations throughout on how to progress from the work that I successfully complete.

If the prototype is working, the resultant distance vs quality table will determine suitable applications. For a public transport environment, lighting is typically situated 0.5m (from head) to 1m (from lap) above a seated person. This height aligns with the reach of a standard person to access fresh air vents, luggage racks, light switches and ‘call’ buttons. For centre aisles a similar distance can be assumed due to the raised ceiling at these points. For a domestic application, ceilings are typically 2.5m from ground level, therefore a seated person would have a ‘device height’ of up to 2m if sitting. The worst case distance would then be 2.5m if the device was on the floor.

Therefore applicability of the prototype for its intended purpose shall be based on those two typical environmental distances. Dependent on test results, consideration could be given to specifying a restricted use environment or a larger envelope.

## 2 LITERATURE REVIEW

### 2.1 Visible Light Communication

This literature review covers the published research relating to video or data transfer via VLC (Light Fidelity), in particular utilising Raspberry Pi or similar low-cost, consumer accessible microprocessor modules. It covers key project concepts such as any potential transmission and receiver circuit configurations, signal management, encoding and decoding of the data into a bit-stream and vice versa, and modulation techniques.

VLC is a field of research that has recently taken on greater importance within our lives. Take for example, the industrial pursuit for releasing to the public, society's first truly operational 'driverless vehicle'. The foundation of this technological product is built on the foundation of VLC (coupled with radar and machine vision). The concept of transmitting data via light waves is evolving into a subject that can yield solutions across our lives, including RF band congestion and quenching the global thirst for faster and more complex data transmission.

While Dr Harald Hass demonstrated video data transmission during his TEDGlobal talk *Wireless Data from Every Light Bulb* (2011), the technological complexities involved in that demonstration were out of reach of anyone other than PhD level academics.

Dr Haas is by far the leading worldwide exponent in the field and outside of his academic research at the University of Edinburgh, he has started his own company called pureLiFi (pureLiFi 2014). This company takes the scientific advances they test in the laboratory and turn them into consumer orientated modules. At present they have a 10Mbps half-duplex system (Li-Flame) that works at a distance of up to 3m (pureLiFi 2014). Although this system focuses on bi-directional internet access, any development of this platform is limited to commercial partnerships with the parent company or through PhD pathways at the university. The Li-Flame system also requires additional roof modules to be mounted next to the LED lights and preclude it from being integrated into many publicly accessible applications. This is largely due to the vandalism exponent and the 'irregular' intrusion into a 'headspace' envelope. For these reasons a recessed or concealed system is mandatory for commercial acceptance. Domestic applications may also gravitate to a more streamlined aesthetic such that there isn't a multitude of 'modules' hanging down from the ceiling throughout the house.

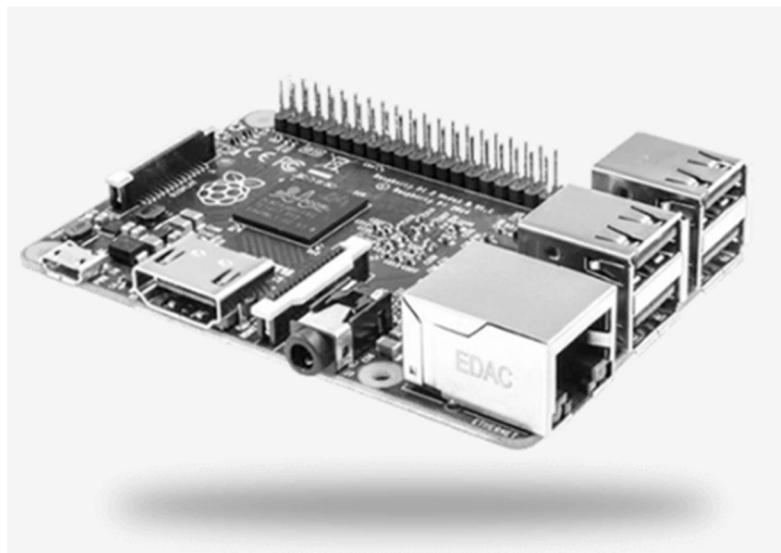
Since the publication of Dr Haas' research, there has been an exponential uptake in development of this technology and market research shows that within the next 7 years, the US market for VLC will grow to over USD\$100b (Grand View Research, 2015). On the back of the increasing interest, and potentially fuelling this growth, there has been an influx of papers, journal articles and conference proceedings that simply regurgitate the buzz words and add little value to the knowledge base. Likewise, the majority of the undergraduate projects in this area are largely around simple transmitter/receiver set-ups connected to an oscilloscope as a 'proof of concept'. If a simple search is run for "Li-Fi" on YouTube, it is plain to see under any of the Light Fidelity projects how many students are fishing for the project circuit diagrams.

There has been a large number of meaningful projects in the field with significant contributions from Grobe et al. (2013) and the OMEGA FP7 Project. This was a European joint task force that came together to create 'a real-life demonstration' 'where the user was able to download several HD video streams in parallel'.

## 2.2 Raspberry Pi

Whilst OMEGA achieved excellent data rates and parallel streaming, the package was not accessible to the consumer from a cost point or in a succinct package such as Raspberry Pi (RPi). Opening up the level of accessibility was attempted by Pathak et al. (2015) with a BeagleBone Black (BBB) module, but this research was limited to low data rate applications and not HD video transmission. Hunter, Conrad and Willis (2014) utilised a Raspberry Pi module coupled to a webcam in place of a photodiode receiver. Their execution of modulating a flashlight and using a webcam proved rudimentary but it did highlight via ‘proof of concept’ that Raspberry Pi modules have a place in these systems.

In 2012, Uhan & Akbas discussed ‘HD video transfer to a projection device’ using Beaglebone boards and PandaBoard modules. Their method of transmission was wireless rather than using VLC, but they concluded that Raspberry Pi could be a future research area. The 2015 conference paper submitted by Nikhade on sensor networks again utilised Raspberry Pi but without VLC technology. Probably the closest work that has come to light is the 2015 BSc Thesis by Ambady, Bredes & Nquyen, which attempted to build on previous student research by transmitting audio using VLC and low-cost processor modules. They assessed a number of processors and decided to use an STM32 module over the Raspberry Pi. They encountered problems with the Analogue to Digital Converter’s (ADC) and although they managed audio signal transfer, it wasn’t HD video. Since that paper was submitted, Raspberry Pi have released two more powerful and updated modules (Raspberry Pi 2B and Raspberry Pi 3), which would have greatly assisted their project.



**Figure 2 – Raspberry Pi 2B Microcontroller**

Small, micro-computers are finding their way into more and more consumer applications, from web controlled advertising signs to Internet of Things (IoT) applications. Cheuque et al. go on to describe controlling LED lighting in a smart home using Raspberry Pi. However, they fall short in making a key connection between modulating those LED lights to further control receptive devices. Embedding the less expensive, but equally capable Raspberry Pi Zero into these household devices can open up

possibilities of secure communication through the home lighting network. This is an area of future work for this prototype.

Klaver and Zuniga (2015) utilised Raspberry Pi and VLC in an effort to demonstrate multi-hop network viability using the technology. Their system was not wedded to a particular micro-computer but they did utilise the Serial Peripheral Interface (SPI) bus for receiving the modulated data as digital voltage levels. They also made use of Universal Asynchronous Receiver/Transmitter (UART) communications to permit bi-directional operation. UART is a possibility with Raspberry Pi and it has dedicated pins (BCM14 – TXD and BCM15 – RXD), but the grounds must be made common. A small tool called Minicom is used to assist with determining correct communication. This method would be useful for serial communication, and does need further consideration for Li-Fi applications.

Raspberry Pi does come with limitations and this has prevented other groups from utilising its emerging processing power. To start with, RPi is a 3.3V device and has current sourcing limits of 16mA per General Purpose Input Output (GPIO) pin with a maximum across all GPIO's of 50mA. This is fine if the application has low current requirements, but it can expose the RPi to damage if more current is required. Many hobbyists prefer the Arduino product due to being a 5V and able to source higher currents. There are a greater number of add-ons for Arduino as well, which many find more adaptable to RPi. The Raspberry Pi does win on processing power and is constantly evolving as a consumer product. It can capably handle media and can operate as a stand-alone computer. For this reason it is a suitable candidate for this application.

There are other comparable devices on the market, and a sample can be seen in Table 1 below. There is a wide and diverse range of options with varying positives and negatives for each. The two main producers are Arduino and Raspberry Pi. Arduino certainly have a significant following and array of prototyping options available, with their speciality in the robotics and interactive, small electronic projects area. They are widely stocked and supported with sensors, break-out and prototyping boards, lights, motors, actuators and anything that requires a small program to control.

The area where Raspberry Pi outperforms Arduino is the area of computing and micro-processing. The Raspberry Pi boards behave exactly as a miniature computer. They can connect to the internet, word process, operate with keyboards, TV's, media processing and generally capable of any computing that doesn't require significant processing or graphics. For this application they are ideal.

<b>Brand</b>	<b>CPU</b>	<b>CPU Speed</b>	<b>Memory</b>	<b>GPIO</b>	<b>Year</b>	<b>RRP (AUD)</b>
Arduino Uno R3	ATmega 16U2	16MHz	32KB flash	14 pin 7-12V	2016	\$43
Arduino M0 Pro	ARM Cortex M0	48MHz	256KB flash	48 pin 3.3V/6-15V	2015	\$55
Raspberry Pi 3B	ARM Quad Cortex A53	1.2GHz	1GB RAM	40 pin 3.3V/5V	2016	\$60
Raspberry Pi 2B	ARM Quad Cortex A7	900MHz	1GB RAM	40 pin 3.3V/5V	2015	\$50
Raspberry Pi Zero	ARM11 Core	1GHz	512MB SDRAM	40 pin 3.3V/5V	2015	\$35
BeagleBone Black C	Sitara Cortex A8	1GHz	512MB DRAM	46+46 3.3V	2014	\$85

**Table 1 - Comparison of Present Market Micro-Controllers**

## 2.3 Transmitter

Aside from media handling and modulation, a fundamentally important aspect of Light Fidelity is the performance of the Light Emitting Diode (LED) transmitter and photodiode receiver themselves. Regardless of whether the system is being used to transmit data packets for bi-directional internet access or packets of modulated video signal, the LED's need to be capable of switching on and off at the required rates. Hetian et al. (2015) prescribe an LED modulation rate of at least 2-3MHz to ensure that it is above the perceptible rate of the human eye, but this is not at a suitable level to transmit large volumes of high quality data. Grobe et al. (2013) touch on the subject of using laser diodes to achieve speeds greater than 3Gb/s. Although fantastic for a laboratory, these are approximately \$80 each (0.8mW blue) let alone spending more money on a suitable receiver. These would not be practical for a domestic or commercial lighting application.

Research from 2011 by Langer et al. utilised white (RGB) phosphorescent LED's (at up to 54MHz) with 'a low cost Positive Intrinsic Negative (PIN) photodiode', and confirmed transmission speed of 230Mb/s. Chang et al. (2015) produced a paper specifically on LED suitability for VLC with blue and green specific LED's giving the highest 'bandwidth of 225 and 463MHz respectively'. However, for a domestic or commercial lighting application white needs to be the visible colour. Probably the most comprehensive research published in 2015 for the subject of LED's and VLC was by Kalavally et al. They run through each marketable LED type and describe the properties in reasonable detail. A key fact to come out of their paper is that 'a study on the type of LED suitable for VLC (is) yet to be done'. This presents that while there are many options, one defined or perfect solution is not yet published.

One aspect of LED development that has emerged in recent years is the micro, addressable LED. These are typically 3-4 separate elements, each of a similar material that has the colour output determined by a varied digital signal sent to it. Many of these are coming in LED strips in 10's of meters long. They can be provided with a single clock signal and the data is passed from one LED to the next. Each LED can be separately addressed and provided with a colour signal, with typically low voltage and current requirements. Longer chains can experience degradation of signal logic, but they can be split with multiple, synched controllers. What makes these attractive to consumers is the ease of use and the multitude of lightweight and impressive applications.

Many LED's in this format are denoted by the size of the luminous element. These include the WS2812 which has a 5050 element and integrated controller chip and also the 3528 LED. The numbers represent the dimension of the LED, such as 3.5mm x 2.8mm for the 3528. The 3528 is commonly surface mounted to a strip at a rate of 60 per meter and have a low power demand ~5W at 12V. The 5050 LED strips are again typically at a rate of 60 per meter but have a higher demand (~15W at 24V) and a much higher lumens output.

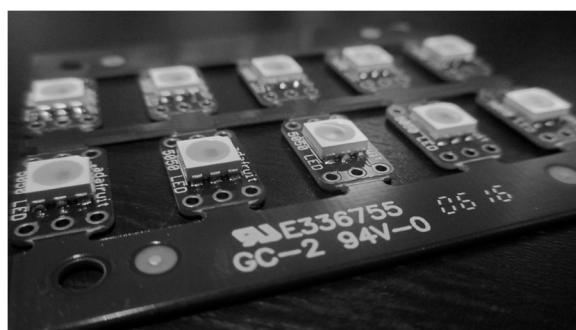


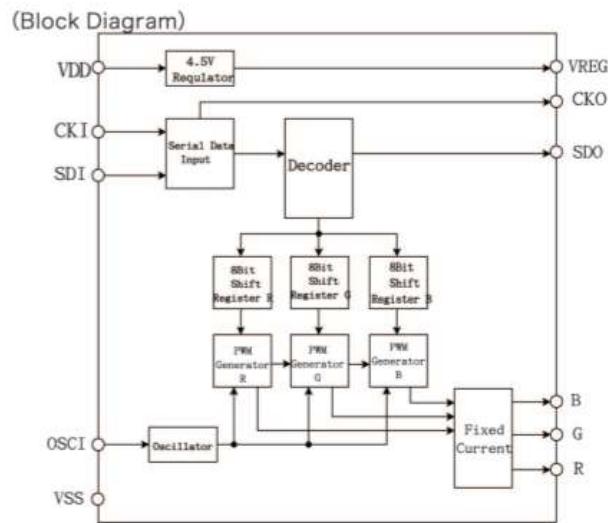
Figure 3 – AdaFruit DotStar LED's. Courtesy of author.

While these strip lights had an increasingly popular application in domestic and commercial environments such as hotel lobbies, aircraft lighting, home feature walls, exterior signage etc, they have not been widely considered as a communication device. A knowledge gap existed between the integration of an addressable and scalable LED strip as a feature or functional light source and as well as a VLC medium. Whilst a singular prototype could be used to prove the concept, the power supply could be adapted to power many strips and could possibly emerge to the market as a modular system.

The majority of the LED strip lights are powered by 5V, but have been tested to work (although not reliably) at 3.3V, they lean towards Arduino applications. In fact, the majority of publicised examples are of this combination rather than Raspberry Pi. AdaFruit Learning Systems have provided a NeoPixel Uber guide (Burgess, 2016) on their website to cover all sorts of applications and this includes tailoring the 5V LED's to a 3.3V microcontroller (RPi). The 3.3V signal is sent from a Raspberry Pi and passed through a 74AHCT125 Logic Level Shifter. This chip converts the lower voltage signal into the 5V signal necessary for the LED to recognise the logic level. This does complicate the circuit a little, but the benefits of media handling and PC-like operation make the effort worth it.

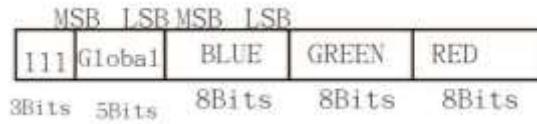
The Logic Shifter also is required to be used with 'buffer' resistors on the GPIO's to prevent inadvertent action of turning an output into an input, potentially causing damage to the Raspberry Pi. These are simple  $330\Omega$  resistors placed in line with each GPIO (SCLK and MOSI in this instance).

AdaFruit have released an improved version of the 5050 LED that makes them attractive when requiring a persistence of vision application (non-flicker) or for high speed display purposes. Each APA102C LED (DotStar) is integrated with a chip that includes a shift register, decoder, PWM generator and these facilitate very high clock speeds (up to 32MHz). There are 6 pins on the APA102C; DATA IN, DATA OUT, CLOCK IN, CLOCK OUT, GND and POWER (5V). They usually come in strips, but can be used individually. They really show their worth when connected in strips as they do not require a set PWM signal (unlike the WS2812 which is strictly 400Hz) and can operate up to 20MHz PWM.



**Figure 4 - Block Diagram of APA102C LED. Refer Datasheet.**

The data signal sent to the IN pin is in the form of 32 bits made up of the information below in Figure 5. Each 8 bits sent to determine the colour signal can be sent as 'FF' or '1111 1111' to give a white light from the LED. In the 32 bit frame the 'Global' bits allow control of the LED brightness in 32 steps.



**Figure 5 - APA102C 32 Bit Signal Components. Refer Datasheet.**

The LED's require a start frame of 32 '0' bits. One of the APA102C benefits comes when after each data signal is received by an LED, it will send a 32 bit string of '0', essentially telling the next LED to be ready for a change. This string is passed down the strip so that each LED is active and waiting. The LED's are addressable so you can instruct individual LED's to light any colour, turn on or off and at any brightness. This presents an advantage for moving pictures, feature lighting or large matrix displays.

The two wire Serial Protocol Interface (SPI) allow for separate clock and data in signals, which is another benefit above the single wire WS2812. This advantage removes the need for critical timing sensitive operations with the WS2812.

Because the LED can demand current right near the recommended values for the GPIO, a separate 5V power supply would typically be required. The RPi 5V output is actually a shared voltage with the demands of the processor and can be unreliable if you're running a number of applications at the same time. Because this is likely with playing or streaming video's plus encoding applications or code, a separate supply will take away that potential for issues.

Additional considerations are the optical wavelength output of the LED's, beam dispersion, the rate that they can be modulated at and the output power. You want to match the wavelength with that of the receiver photodiode to ensure that the signal is correctly received. Understanding that wavelength can also allow you to narrow your photodiode specification to reduce effects of ambient light.

If the output power of the LED's are low, the received signal may not be able to be distinguished from ambient light. If the output power is too high, you will need to factor in heat management and the application may be too harsh for interior domestic lighting! Dependent on the number of LED's, beam focussing via optical lens may be required. This would narrow down the use of the prototype so would be an option of the design needed it.

Model	Luminous Flux	Addressable	Type	Voltage/ Current	Frequency	Cost
Cree XML XLamp	260 Lm	No	W	3.1V	unknown	\$7.60 ea.
APA102 DotStar	18 Lm	Yes	RGB WWW	5V/120mA	20kHz PWM	\$4.5 (10LEDs)
WS2812 NeoPixel	18 Lm	Yes	RGB	5V/60mA	400Hz PWM	\$25 (60LEDs)
SK6812	7 Lm	Yes	RGB	5V/60mA	1.1kHz	\$25 (60LEDs)
Kingbright NSPW500	20 mcd	No	W	3.2V	unknown	\$1.50 ea.
Lumileds LXML	180 Lm	No	W	3V	unknown	\$4.25 ea.

**Table 2 - Comparison of LED Alternatives for Transmission Module**

## 2.4 Receiver

From an optical receiver perspective, there are typically two generic choices for the sensor; an Avalanche Photodiode or a PIN Photodiode. Kalavally et al. (2015) describe that the Avalanche has the better gain, but at a cost of shot noise in the photocurrent. PIN Photodiodes are better in high temperature, high saturation environments and are the general preference in VLC. Chung et al. (2015) proposed a filtered photodiode cluster arrangement to produce 3 separate channels (RGB) for a bi-directional system. While this may have a use for larger applications, unless bandwidth becomes a limiting factor, the complexity will preclude it from being explored further here.

The receiver is probably the most complex aspect of the system and the most critical to get right. With transmitted light, the received levels are very small and susceptible to noise, capacitance and amplifier circuit oscillation. Considerations need to be given to a low capacitance photodiode, one that can handle faster speeds, one that has a good bandwidth product (for signal gain) and one that has a level of simplified use. Cost was a minor consideration as well as accessibility to students or consumers. For this reason, consumer electronic retail outlets such as Element14 and RS Components were contemplated for provision of suitable devices. Many of the scientific papers were laboratory based and used expensive (>\$300) modules in their test beds and set-ups (Kamsula 2015). Here, the execution is required to be ubiquitous, accessible and compact. As such, the Hamamatsu Photonics S5973 series photodiode appeared as a suitable candidate.

Photodiodes can typically be connected in two different configurations; photovoltaic or photoconductive modes. Photovoltaic mode is where the diode is unbiased (without reference voltage) and the voltage produced is non-linear when compared to the amount of light on the sensor window. This method can be likened to the operation of solar cells when light produces a voltage but it also reduces the dynamic range of the diode. Photovoltaic mode is preferred when sensitive measurements are required, as there is a smaller signal offset and less noise induced in that area of the circuit and passed to the amplifier stage.

Photoconductive mode is where a reverse bias voltage is applied and results in lower capacitance, thus higher switching speeds at the detriment of producing additional heat. Other drawbacks of this mode are that the configuration is noisier and has a higher dark current over photovoltaic mode. The produced photocurrent is however linear over a certain range. Photoconductive circuits are more suited to high light ‘power’ and high speed applications.

Given that the Raspberry Pi power GPIO’s are either +3.3V or +5V (no negative swing), this will largely dictate the photodiode set-up and component selection. That is, a low power, low current diode that can operate suitably in photovoltaic mode. The amplifier stage will also need to perform within these bounds, in particular with a single supply.

In order to take the very small current produced by the photodiode, an amplifier of some configuration is required. There are a number of devices that are utilised, one of them being a Trans-Impedance Amplifier (TIA). These are typically set up as an integrated chip with various features and are primarily used as a single gain stage, turning photocurrent into a voltage for processing. Features for these devices often include offset correction, overvoltage protection, overload recovery single or multiple supply and single or differential outputs to name a few. Fujimoto and Mochizuki (2013) employed a photodiode and TIA configuration (OPA847) with another amplifier stage for very high speed bit streams. The same approach was utilised by Dimitrov and Haas for their high end bidirectional Li-Fi systems (2015).

In Orozco's 2014 Technical Article about optimizing precision photodiode circuit design, he outlines that when selecting an amplifier, the order of important considerations are input offset voltage (to be as low as practical), input leakage current (again as low as practical), circuit layout (to minimise external leakage) and noise management. He goes into reasonable detail on how to compromise and reduce many of the common problems with this type of receiver.

Some designs have required a second stage amplifier which are AC coupled to block any unwanted DC voltages sent from the first amplifier stage (Kitchin, 2007).

A key limitation for this application is the low voltage supply level (3.3-5V) from the Raspberry Pi GPIO. One of the design aims is to make the prototype as 'modular' and self-contained as practical. Introducing additional separate power supplies move away from that concept.

A final consideration is to provide a protection circuit on the output of the receiver circuit that feeds to the Raspberry Pi GPIO. This will prevent damage to the microcomputer and is a small investment for the design. These can range from opto-isolators, transistor switches and Zener diodes (eLinux.org). The Zener circuit is simple to implement for RPi, with a  $330\Omega$  resistor and 3.3V Zener diode between the GPIO input pin and the output of the receiver circuit.

## 2.5 Signal Management

The ultimate goal of the transmitter and receiver system is to send a digital signal that represents a more complicated signal (video) and preserve the original content. For this to happen, the content must first have the binary data extracted, then transmitted with accuracy, received clearly and amplified to represent the same format as the transmitter was given.

Probably the main contributor for digital signal degradation is noise. Noise is the unwanted signals that interfere with the useful information you're transmitting/receiving/measuring. It can come in many forms such as thermal noise, shot noise, jitter and electromagnetic interference. Prototyping and small signal measurement circuits are susceptible to noise as often the environment (breadboards, multiple wires etc.) is highly capacitive and can easily swamp any small signals you're trying to utilise.

The Transimpedance Amplifier Application Report written by Xavier Ramus (2009) talks about compromises between gain and Johnson noise caused by the feedback resistor. If you choose a large value resistor, you will have plenty of gain, but introduce more noise to be amplified. Many of the other reports created in this field talk about using capacitors to short between power supplies and ground (Texas Instruments, 2007 & Orozco, 2014). Photodiode leads should be a short as possible and as close to the TIA as practical and breadboards are fine for circuit function, but should be transferred to PCB's for final execution.

At speeds above 10kbps Klaver and Zuniga (2015) experienced deformation of the generated square waves and this made the demodulation process complicated. They had to settle for a 1kbps limitation in transmission rate to guarantee that the quality of signal was preserved. The cause of the deformation was not discussed nor mitigated in their paper.

Ambient light can be a problem for the receiver circuit in that it can saturate the photodiode and the modulated signal is lost. Dimitrov and Haas (p97, 2015) describe that ambient light can be modelled as white Gaussian noise which includes the thermal and shot noise of the receiver. This type of noise can be filtered out using capacitors or adding band-pass or band-stop filters to the circuit.

## 2.6 Signal Modulation

The modulation of the light is a broadly debated topic. Kalavally et al. (2015) cover in detail over ten different methods, as well as a multitude of dimming techniques. Again, no defined modulation preference is presented and with reference to Table VII of their 2015 paper (pictured over), the closest preferred method is On-Off Keying (OOK) or a derivative (e.g. NRZ-OOK). Langer et al (2011) said for ‘data rates up to 100Mb/s, standard OOK is sufficient. To get higher, you need to use Dual Multi Tone (DMT) modulation techniques and you can get up to 800Mb/s’. OOK modulation will be considered for the first tests but the data rates, complexity and circuit performance may necessitate utilising alternative schemes.

If an addressable LED is used in the transmitter, consideration will be given to the size of data packet that needs to be sent to the LED. Additionally, the LED’s are typically capable of variable brightness (through PWM) so this will add another dimension of data encoding to the digital signal. This however, may complicate the receiver circuit with regards to rising/falling edge detection and level detection across 0-3.3V.

Year	Data rate	Distance (m)	Transmitter	Receiver	Modulation	Comment
2014	3 Gb/s	0.05	GaN $\mu$ -LED	PIN	OFDM	Pre and post equalization
2014	340 Mb/s	0.43	pc-LED	PIN	OOK-NRZ	Post equalization
2013	3.22 Gb/s	0.25	RGB	PIN	CAP (WDM)	
2013	1.5 Gb/s	N/A	$\mu$ -LED	PIN	OOK	$\mu$ -LED pixel with CMOS electronics
2013	1.1 Gb/s	1	pc-LED	N/A	OFDM (WDM)	4 X 9 MIMO system
2013	575 Mb/s	0.66	RGB	PIN	QAM (WDM)	Bidirectional link
2013	500 Mb/s	5	pc-LED	PIN	DMT	Data rate adaptation with FEC
2013	477 Mb/s	0.40	RGB	PIN	OOK-NRZ	
2013	150 Mb/s	0.5	pc-LED	PIN	OOK-NRZ	Neural network based receiver
2013	300 Mb/s	11	rc-LED	PIN	OOK-NRZ	Current shaping circuit
2013	10 Mb/s	1	pc-LED	PIN	PAM	NRZI code to mitigate interference
2013	3.4 Mb/s	0.80	pc-LED	APD	SSB-OFDM	CA-VLC system
2013	2 Mb/s	0.4	pc-LED	PIN	OOK-NRZ	Duplex communication between android mobiles
2013	2.7 Mb/s	0.10	OLED	PIN	PPM	
2013	1.4 Mb/s	N/A	OLED	PIN	OFDM	
2012	3.4 Gb/s	0.3	RGB	APD	OFDM (WDM)	
2012	2.1 Gb/s	0.1	RGB	APD	OFDM (WDM)	Optimum bit power loading allocation
2012	1.25 Gb/s	0.1	RGB	APD	OFDM (WDM)	
2012	1.1 Gb/s	0.23	pc-LED	PIN	CAP	
2012	1 Gb/s	0.1	pc-LED	APD	OFDM	Highest data rate for a single pc-LED
2012	780 Mb/s	2.5	RGB	APD	OFDM	Long distance high speed link
2012	806 Mb/s	0.08	RGB	PIN	OFDM	
2012	614 Mb/s	0.40	RGB	PIN	OOK	Duobinary technique was used
2012	80 Mb/s	0.1	pc-LED	PIN	OOK	Pre-equalization
2011	803 Mb/s	0.12	RGB	APD	OFDM (WDM)	First demonstration of WDM capability
2011	410 Mb/s	0.9	RGB	APD	OFDM (WDM)	Localization aided OWC
2010	1 Gb/s	N/A	$\mu$ -LED	PIN	OOK-NRZ	LED fabricated from epitaxial wafers
2010	513 Mb/s	0.3	pc-LED	APD	OFDM (QAM)	
2010	230 Mb/s	0.27	pc-LED	APD	OOK	
2010	220 Mb/s	1	pc-LED	PIN	OFDM (MIMO)	2 X 9 MIMO system
2010	84 Mb/s	1.4	pc-LED	PIN	OFDM	Real time link
2009	231 Mb/s	0.7	pc-LED	PIN	OFDM	
2009	125 Mb/s	5	pc-LED	PIN	OOK	
2008	40 Mb/s	2	pc-LED	PIN	OOK	Multiple resonant equalization

Figure 6 - Table VII from Kalavally et al. LED Based Indoor Visible Light Communications; State of the Art, IEEE 2015.

## 2.7 Encoding/Decoding

A common video format in use today is MPEG-4 (h.264). This format is used in Blu-ray discs, mobile devices, High Definition (HD) video storage and digital HD television (Marpe, 2006). It is favoured because of its ability to be greatly compressed without significant loss of quality. This fact would make it suitable for Li-Fi applications to reduce the amount of information processed, transmitted and received. That said, the management of codecs may prove to be outside the level of the project.

Caihong et al. (2013) introduce the benefits of a powerful open-source multimedia framework called FFmpeg as one that can handle various transmission protocols, media container formats and regular video codecs. They state that it has suitable algorithms for real-time media processing. This did not use Raspberry Pi for their solution, but the paper content does give hope it can play a part in managing codecs and video.

A number of papers exist where the authors have utilised Field Programmable Gate Array (FPGA) boards to manage the encoding and data transfer from parallel to serial bitstream (Zhu et al. 2015, Wang & Xu, 2015). An FPGA can also be used as a buffer to store frame data from the video file for processing. This can be applied to the transmitter and receiver modules. A buffer could be used on both sides of the system, with the use of pointers and read commands to create and collect the appropriate data in sequence.

Wang and Xu (2015) specify Manchester coding for long distance communication to compensate for a lack of synchronisation information and to provide a level of immunity to interference. They go further to use 4B/5B encoding that uses 5 bits to send each 4 bits of data from a predetermined dictionary. The downside is that more bits are required for the same information but the upside is that 4 bits without transitions (e.g. 0000, 1111) is sent with a corresponding 5 bits from the dictionary, with transitions introduced (e.g. 11110, 11101 respectively). This approach is also included in Jia et al. VLC/FPGA prototype. Additionally, they favour Manchester coding to lift the LED average power and thus eliminate any perceived light flicker.

## 2.8 Existing Prototypes

Utilisation of a Raspberry Pi microcontroller to stream video already exists and a low latency example using Wi-Fi was discussed by Petr Kout (2014). He employed an existing application called MJPG Streamer that used the 8080 port of the Raspberry Pi to transmit the video data via wireless channels. The application handled all of the video and audio mux conversion as well as the encode/decode process. While this is simplistic and market ready, it does not address the issue of Radio Frequency saturation, nor is it suitable for modulation of LED's. Perhaps the front end interface of MJPG Streamer can be tailored to modulate an LED to send the data packets.

There are other academics whom have used Raspberry Pi such as Nikhade (2015) but the majority have engaged wireless communication, as the RPi has the processing power and features to act as a device familiar to the population. Thus the knowledge level in application is already mature.

## **2.9 Knowledge Gap**

VLC is still young in its technological cycle, but with the groundswell of interest in this topic, it won't be long before it starts to make its way into mainstream environments and homes.

Whilst a handful of undergraduates are showing interest in this technology, they appear to be starting from scratch each time and would greatly benefit from a solid baseline to start from. Designing such a prototype and publishing the configuration and performance figures would greatly enhance the uptake for future development.

From the above literature review conducted on the topic, it is clear that there is a knowledge gap in the area of a low-cost, consumer accessible platform using Light Fidelity (and VLC) technology and addressable LED's to transmit high definition video signals. Indeed, the lack of collated information around Raspberry Pi video transmission, bit-stream creation and file management warrants the contribution of this research project.

With the proliferation of addressable LED's in domestic and commercial use, these present as an option for the transmission element. Imagine if your feature lighting also transmitted video, audio or data. This could equally work for hotel or public transport lighting. From the research I have conducted, there has not been any discussion or utilisation of addressable LED strips in a Visible Light Communication system.

## 3 PROJECT METHODOLOGY

### 3.1 Overview

This project was purely one of research and experimental work. The project was executed as a progressive development of concept following a body of research and expansion of ideas to achieve the aim. The steps generally followed a typical Project lifecycle model from the Project Management Book of Knowledge (Stackpole, 2010).

- **Project Initialisation** – research proposal, literature review, design information
- **Project Start Up** – procurement, production readiness
- **Project Production Phase** – build the prototype
- **Project Execution** – get the prototype working, test and develop
- **Project Closure** – write up dissertation, demobilisation

### 3.2 Initialisation

The Project Initialisation Phase was a combination of producing a preliminary report, deeper research into prototype specifics and formalisation of the project scope and outcomes.

This phase was purely research driven and the investigation carried out defined the general shape of the project. Significant information was gathered during this phase and a thorough search helped to prevent errors or miscalculations during construction.

**The outputs of the Initialisation Phase were;**

- A mental roadmap of the Project direction, with alternative routes in reserve,
- A list of required resources, and
- Set-up of a Project Diary to track daily progress and the reasons behind the various decisions made throughout the Project.

### 3.3 Start-Up

The Project Start-Up phase took the outputs from the Initialisation Phase and built on them to begin the procurement of Project resources, allocation of Project space, Project management systems and to mark the start of the physical work phase of the Project.

This phase moved the Project from an idea, theory and research based Project to one that was tangible.

**Outputs of the Start-Up Phase were;**

- An allocated workspace to construct the prototype,
- Focussed research to a preliminary modulation technique,
- Understood how to drive the LED's from a 3.3V Raspberry Pi output signal to a LED i/p signal
- All necessary software downloaded and installed, and
- Purchased and procured all test equipment, power supplies, components, devices, cables and frame work.

### **3.4 Production**

The Production Phase marked the start of the actual building of the prototype. This was undoubtedly one of the most exciting aspects of the project and brought with it the start of the major challenge to get it working.

**The outputs of the Production Phase were;**

- Build and develop the LED driver circuit to deliver the required voltage to modulate the LED's, and
- To completely construct all power supplies, equipment and components.

### **3.5 Execution**

This phase contained the largest amount of work, frustration, technical demand, support and investigation. The primary focus was to get it working and then to look for optimisation and improvement, time permitting.

**The planned sequence of work was;**

1. Connected power supply and signal management circuits to drive LED's on,
2. Confirmed that photodiodes could receive the LED's being manually switched on and off,
3. Used the Raspberry Pi (RPi) transmission module (TX) to create a simple square wave to send to the RPi receiver module (RX) and achieve basic communication, displayed on a screen or other output device,
4. Increased data speed (via small steps) and maintained expected outputs,
5. Attempted to send a file across free space, maintaining the quality of the original.

**The outputs of the Execution Phase were;**

- A Project Diary that recorded the path followed throughout the Project to provide a resource to build the Dissertation from,
- A working Prototype that could send files across free space using Visible Light Communication technology, and
- A final configuration (including software + code used) of the system.

### **3.6 Project Closure**

The Closure Phase intends to wrap up the Project and finalise all aspects ready for academic submission. Closure includes decommissioning the prototype system, returning borrowed or loaned equipment and writing up the results. Should the Project provide suitable material, a publishable paper could be written to give information to the wider engineering community.

**The outputs of the Project Closure phase are;**

- An academic Dissertation submitted to USQ for ENG4112,
- Decommissioning of the prototype, and
- A final parts list and configuration document to share with industry.

## 4 PROTOTYPE DEVELOPMENT

This chapter describes in detail the steps that were taken to arrive at a working prototype. It covers all of the technical information, principles and operation pertaining to the prototype.

### 4.1 Project System Diagram

The system can be broken down into two sub-systems; the Transmitter (TX) and the Receiver (RX). Each one is made up of smaller functions such as the LED electrical control circuit and board, the photodiode electrical control and board, the software in the TX and RX modules, the peripherals (keyboard, monitor, mouse etc.) and support software (e.g. MobaXterm).

The Figure below illustrates how the prototype is made up from a system point of view and shows the interaction between the varying components.

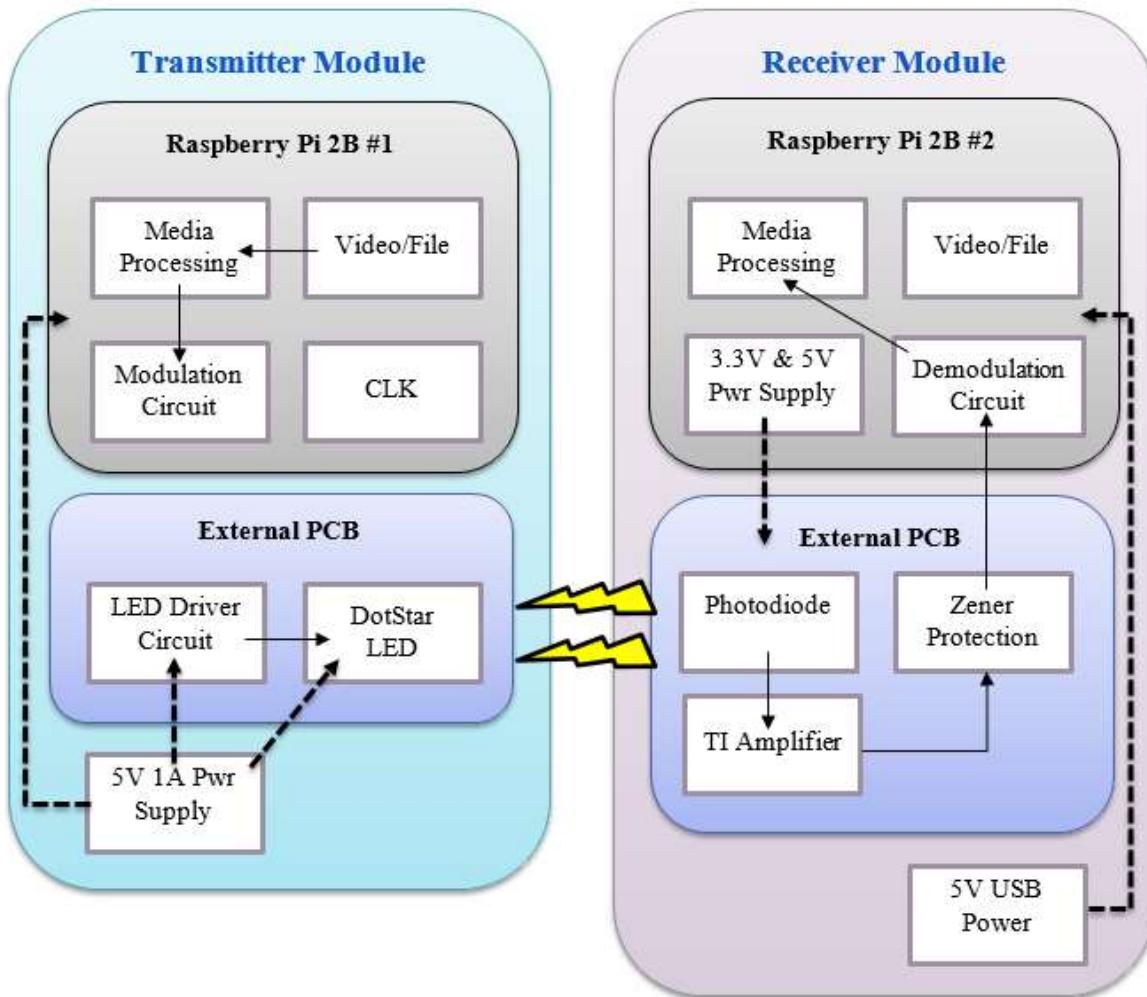


Figure 7 – Prototype System Diagram

The operation of the two modules will be outlined in the following chapters.

## 4.2 Transmission Module

The transmission module has the function of selecting the file to transmit, extracting the binary data from it, coding that to serial data on a GPIO, processing that data through a level shifter and turning ON/OFF an LED accordingly.

### 4.2.1 Raspberry Pi

#### 4.2.1.1 *Why Raspberry Pi?*

The Raspberry Pi was selected as the micro-controller over the Arduino family of devices due to its ability to handle faster processes, its versatility across all aspects of computing, low cost and simple to use package. The Arduino family does tick off the low cost and actually has a wider range of supporting devices such as sensors, displays, lights, data processors, but it was assessed for this project as not suitable for the potential media handling and general computing requirements of the project. Arduino's are very popular with robotics and hobby projects, more targeted to smaller engineering-type projects.

Raspberry Pi offered the diversity to prevent the project arriving at a roadblock that was unable to be overcome, such as processing speed or data management. The RPi 2B CPU has a clock speed of 900MHz and can be overclocked to 1000MHz whereas the Arduino's (in the same price bracket) have CPU speed <100MHz. Additionally, the RPI came as a complete package, with Arduino requiring 'bolt-on' components which would have increased the cost and complexity.

Other micro-computers were considered, but the support documentation, availability, cost and function removed them from final selection and purchase. A table of these has been presented earlier in this document.

#### 4.2.1.2 *The Raspberry Pi Family*

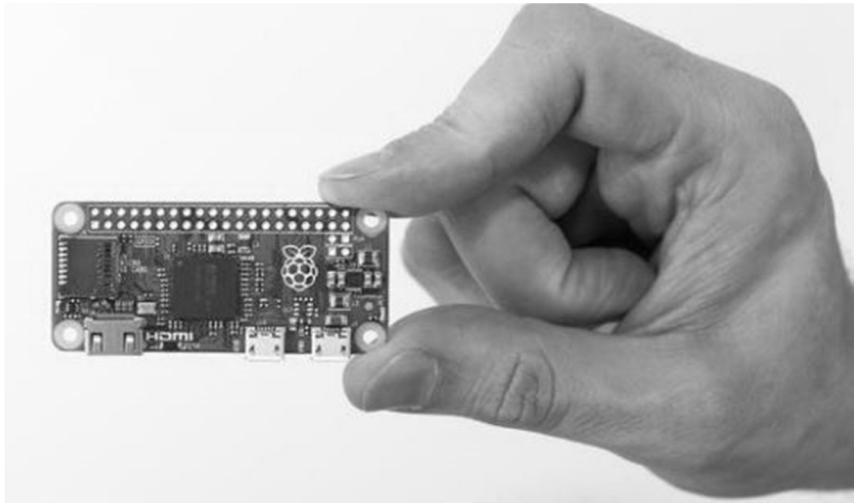
Raspberry Pi is compact, single board and relatively powerful microcontroller that was developed by the Raspberry Pi Foundation in 2012. The first board was named Raspberry Pi A version 1. It consisted of '256MB of RAM, one USB port and no Ethernet port' (Raspberry Pi Foundation 2015). The CPU was a 700MHz single core ARMv6 (32 bit) processor and is still regularly used for embedded projects or those that have simple requirements such as media management behind TV's.

From these simple and successful beginnings, a B model was released followed by A+ and B+ upgrades a year later. These retained the same processor and CPU, but had an increase in RAM to 512MB. While the size of the boards stayed the same, more ports, GPIO pins and peripheral connectors were added to increase project flexibility. These included HDMI 1.3, Camera slot, 3.5mm audio jack, Ethernet and a microUSB for power application.

It wasn't until the release of Raspberry Pi 2B that board performance really took a step forward from the early models. This was released in February 2015 and came with a 900MHz 32 bit Quad Core ARM Cortex A7 CPU and 1GB of RAM (Raspberry Pi Foundation 2015). USB ports were increase from 2 to 4 and the inclusion of a MicroSDHC slot to save space on the board. The release of RPi 2B coincided with a growth of hobbyists, students and teachers utilising RPi for learning and projects alike. Their versatility and relative low cost compared to performance meant more and more people could access the technology. At the time of commencing this project, the RPi 2B was the latest model.

Possibly as a show of ability rather than market share challenger, a Pi Zero microcontroller was released late in 2015. It actually shipped free with the December 2015 copy of their MagPi magazine. The retail of the Pi Zero was set at USD\$5 and weighed just 9g. Despite the miniature size (refer Figure below),

it still came with a 1GHz ARM11 single core CPU and 512MB of shared RAM. There was room for a USB and HDMI port with mini camera interfaces and a 40 pin GPIO.



**Figure 8 - Raspberry Pi Zero. Courtesy Raspberry Pi Foundation.**

With the Pi Zero shipping after the purchase of the RPi 2B's and the limited supply run, this hindered any potential use in the project. Regarding future work, it would be perfect to be the processor for any Internet of Things (IoT) projects due to the small size and reasonable performance. A wireless dongle can be connected to the USB port and easily configured to allow internet access for this mini-computer. Once on a network, a remote access application such as MobaXterm can be employed to control every aspect of the device if it was embedded or attached to another appliance.

In February 2016 the Foundation released another significant member of the family with the RPi 3B. The upgrades included a 64 bit ARMv8 (Cortex A53) Quad Core processor running at 1.2GHz. Wireless Ethernet (802.11n) and Bluetooth 4.1 were noteworthy additions to the board (Raspberry Pi Foundation 2015) above the specification of the RPi 2B. This does come at a slight cost in that you will need 2.5A to power these new performance features, which is up from 2A with the RPi 2B (RS Components 2015).

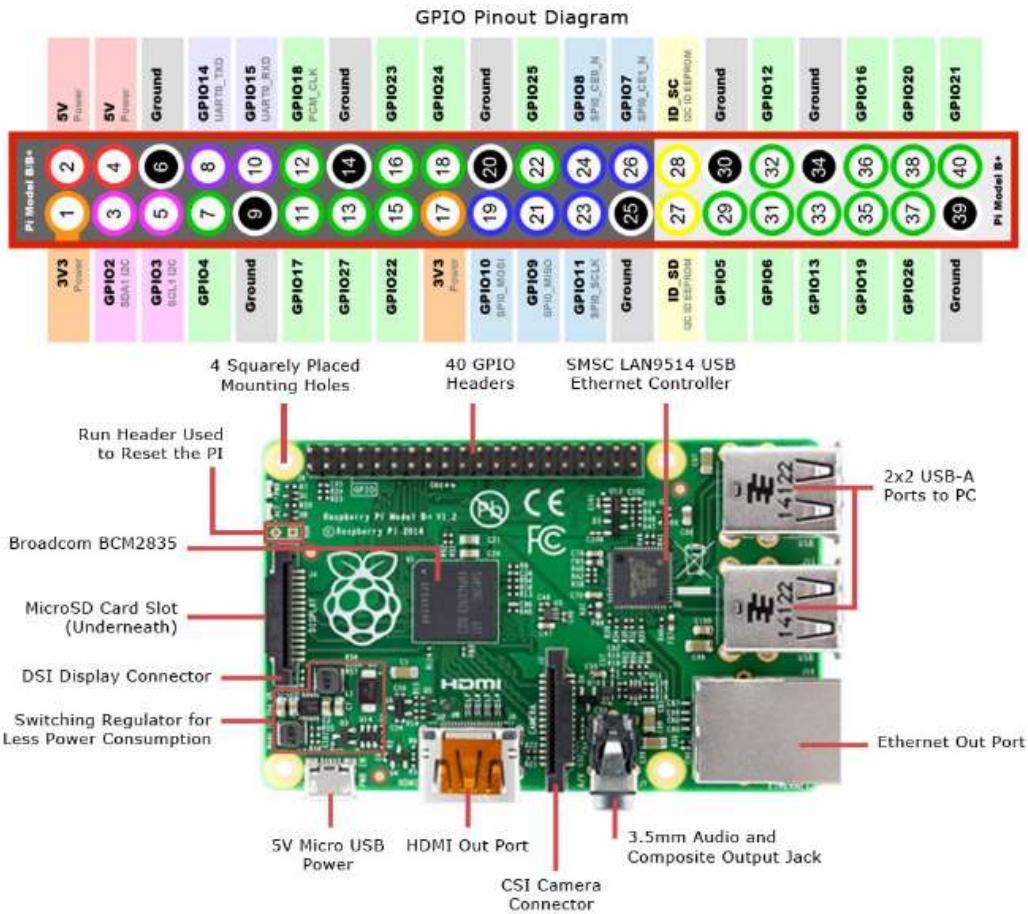
#### *4.2.1.3 Software*

Raspberry Pi functions with a Linux based distribution as its operating system (OS). The particular OS they provide for download (Raspberry Pi Foundation 2015) is Raspbian. This is a Debian based Linux version that uses the Linux Kernel and other applications, which are generated as Open Source software by a large group of contributing programmers (SearchEnterpriseLinux 2016).

Raspbian has a version of software available for complete beginners called ‘NOOBS’. This stands for ‘New Out Of the Box Software’ and has all of the standard settings and options ticked to make sure it is simple to install and operate. Typically you download the NOOBS software from the Raspberry Pi website and install it onto microSD card dependent on your version of RPi. Alternatively, a pre-downloaded SD card is often available when you purchase your RPi. There are a large number of other OS for Raspberry Pi and it is really up to the individual preference which one they choose. The Raspberry Pi Foundation recommend to start with NOOBS.

#### 4.2.1.4 Configuration

The 40 pin GPIO is what makes the RPi 2B so versatile in what it can be used for. Among the pin configuration there is +3.3V and +5V power supplies, but they actually share the supply current with the rest of the CPU. In fact, the RPi 2B is limited to a 2A input which is protected by a polyfuse (F1), and the 4x USB's are allowed 1.2A max. This then leaves 800mA (if everything is being driven hard) so if the CPU, USB's or other GPIO's are sinking a lot of current, the output voltage pins are not able to supply much at all. Conversely, if the USB devices use 1A, then there is 1A for the CPU, GPIO's and outputs.



**Figure 9 - Raspberry Pi 2B Locations & GPIO. Courtesy RS Components.**

The GPIO's themselves have current sink/source limits and care must be taken when planning to connect external devices. Individual pins must not pull more than 16mA and the entire GPIO must not source more than 50mA (Clifford, 2016). Therefore the RPi is only suitable for connecting low power demand components to rather than current hungry devices.

The RPi 2B does have a level of protection built in, with the polyfuse and the USB power control limiters. The polyfuse is a self-resetting polymer fuse that takes a few hours to close the circuit again. However, the GPIO's do not have any protection against incorrect feeds or shorting out, so particular attention needs to be given when planning any interface circuits or devices. Such protection can come in the form of a +3.3V Zener diode/resistor circuit, 330Ω resistors in series, physically covering the GPIO's, opto-isolators and level converter chips.

#### 4.2.1.5 *Peripherals*

To use the RPi as a micro-computer you will need some way to interface with the CPU. This typically comes in the form of a keyboard and mouse, with the output of your commands being displayed on a monitor. On the RPi 2B there are 4 USB ports to plug in such devices and an HDMI port for your monitor. There is a DSI Display Connector to plug in a TFT or LED display. A webcam can also be plugged into the CSI Camera Connector as well. The location of these on the board can be seen in the Figure above.

A key item to include when setting up is a wireless dongle, although if you wanted to you could connect a standard Ethernet cable to the available port. The wireless dongle is simply connected to one of the USB ports and this will allow you to connect to the internet as well as keep your Raspberry Pi updated with the latest software releases. Additionally, if you know the IP address you can connect to the RPi remotely using a multitude of applications. The Raspberry Pi Foundation website has a list of these methods under >>help>>documentation>>remote access. These include Weaved, VNC, SSH and Web Server.

#### 4.2.1.6 *MobaXterm*

For this project a freeware program called MobaXterm was used. This allowed access to the RPi using an existing computer, keyboard, monitor and mouse, so that additional peripherals weren't required to operate the RPi. All that was required was a wireless dongle, the IP address of the RPi and a download of the MobaXterm application (Mobatek, 2015).

Connecting the RPi is simple. Power up the device then open the MobaXterm application. Select >>Session from the menu shown in the Figure below;



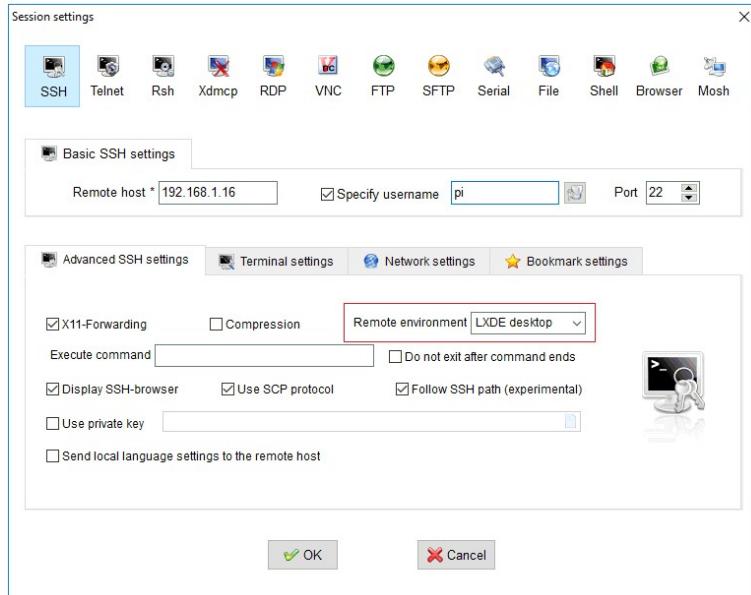
Figure 10 - MobaXterm Start Menu Ribbon

From there another window will pop up and select >>SSH, then enter the >>Remote Host IP address (your RPi), select >>Specify Username (your RPi name) and finally select >>Remote Environment as >>LXDE Desktop. Select OK.

A separate window will open that will show the RPi desktop on your PC's monitor. You will be required to enter your RPi password on the session window. Once this is all complete, you will be able to control your RPi from your existing peripherals.

A limitation of this set-up is that you cannot copy and paste or drag from the RPi window to your PC. Any information transfer needs to be done via the File Transfer Protocol (FTP) option on the left hand side of the session window. Here you can upload files from your PC to folders on the RPi and vice versa.

MobaXterm does come with many additional features but they were not utilised for this project.



**Figure 11 - MobaXterm Session Option Window**

#### 4.2.1.7 *Setting up RPi*

Once you have your RPi and the operating system of choice, it is as simple as following the installation guide on the Raspberry Pi Foundation's website. After you have the peripherals and/or remote access established, you should open the Terminal application (also known as Command Line) and start an update and upgrade of applications. This keeps your programs running with the latest features and bug fixes.

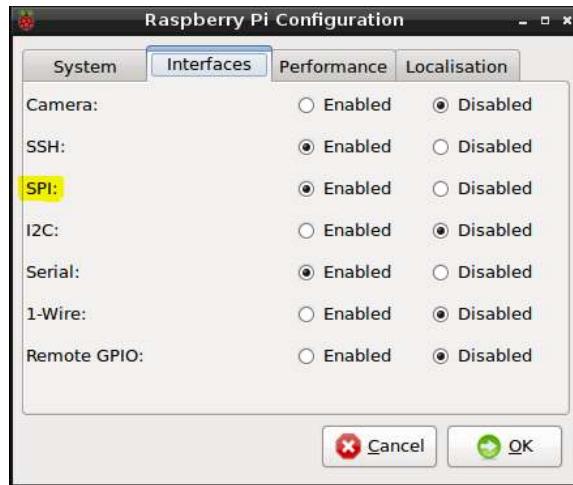
In the command line enter >>sudo apt-get update and follow the prompts. Next enter >>sudo apt-get dist-upgrade and follow the prompts.

If you are running out of space on your RPi, you can get some back by removing the unpackaged files that you download as part of these updates and upgrades. Again, go to the command line and enter>>sudo apt-get clean.

#### 4.2.1.8 *SPI*

The Raspberry Pi presents a reasonable amount of transmission methods and protocols for the movement of data. The two most common and well supported are Serial Protocol Interface (SPI) and Universal Asynchronous Receiver/Transmitter (UART). Both transmission functions were suitable for the method to send the data through the GPIO to the LED circuit but SPI was better supported with literature and internet information. SPI uses GPIO pins 19+38 (MOSI), 21+35 (MISO), 23+40 (SCLK), 24 (CE0) and 26 (CE1), so you it can be utilised for two separate sources.

SPI is easy to set up on the RPi, with the new >>raspi-config option under >>Main Menu>>Preferences>>Raspberry Pi Configuration. The menu in the Figure below should pop-up and it is as simple as enabling SPI and clicking OK.



**Figure 12 - Raspberry Pi Configuration Menu.**

A reboot is required for the change to take effect. Once it has been enabled, you can check that it is working by first typing `>>lsmod` in the Terminal application and you should see “`spi_bcm2835`” in the list.

```
pi@raspberrypi:~ $ lsmod
Module           Size  Used by
cfg80211        407580  0
rfkill            16036   2 cfg80211
8192cu          519604   0
snd_bcm2835      19802   3
bcm2835_rng       1763   0
bcm2835_gpiomem   2860   0
snd_pcm           73474   1 snd_bcm2835
spi_bcm2835       7074   0
snd_timer         18848   1 snd_pcm
snd               50779   9 snd_bcm2835,snd_timer,snd_pcm
uio_pdrv_genirq   2944   0
uio                7753   1 uio_pdrv_genirq
i2c_dev            5671   0
fuse               80694   3
ipv6              338660  69
pi@raspberrypi:~ $
```

**Figure 13 - Raspberry Pi Terminal Check for SPI**

To actually use SPI, you need to have a Python library installed to act as the ‘SPI wrapper’. This library is now included in the latest Raspbian image, but it pays to check that it is present. Type `>>apt-mark showauto | grep spi` in Terminal and you should see the two spidev libraries as per below;

```
pi@raspberrypi:~ $ apt-mark showauto | grep spi
gtk2-engines-clearlookspix
libatSpi2.0-0
libwinpr-spi0.1
python-spidev
python3-spidev
pi@raspberrypi:~ $
```

**Figure 14 - Raspberry Pi Terminal Check for SpiDev.**

The SPI clock (SCLK) can have the speed changed in powers of 2, from 500Hz up to 32MHz. Worth noting is that if you don't set it to one of the powers, it will default down to the nearest power of 2 value. For example, setting the SCLK to 21MHz will produce a clock speed of 16MHz.

#### 4.2.1.9 FFMPEG

FFMpeg is freeware that works with Linux to encode and decode audio and video files. It's used inside various popular video platforms to convert, play or stream video and audio. This package was reviewed as a candidate to handle the media conversion before the transmission circuit. After a detailed review of the complexity involved with video and audio streaming, the use of FFmpeg was discounted.

### 4.2.2 LED Circuit

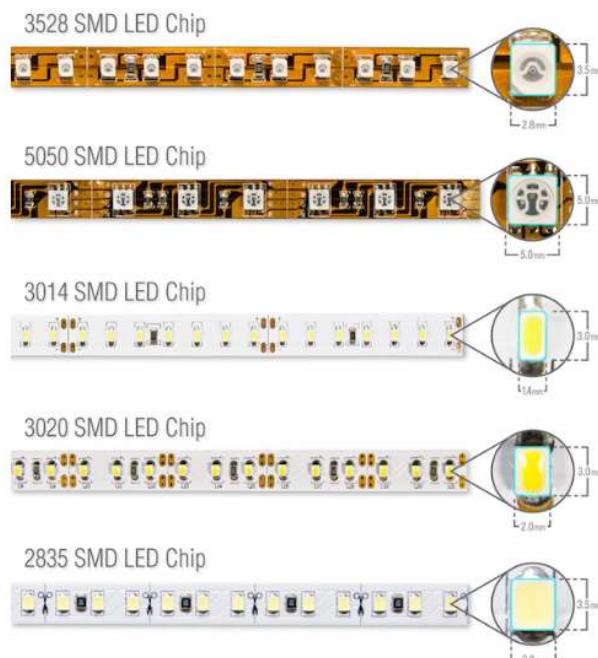
The LED circuit is the centrepiece of the project and directed the function and performance required from the other aspects of the prototype. The desire to utilise addressable LED's in an effort to explore data transmission from modern lighting set-up's (public transport, feature lighting etc.) eliminated standard singular LED's.

Reflecting on this decision, the use of a singular, bright LED would have been a simpler aspect to integrate into the prototype and possibly would have allowed deeper progression towards the aim and objectives.

This circuit also consisted of a method to shift the Raspberry Pi's +3.3V GPIO to the LED's required +5V, protection and mounting of a beam focusing device for the LED.

#### 4.2.2.1 Addressable LED's

Putting aside the singular LED style, the addressable LED market is limited to a small number of standard packages that are then tailored by manufacturers. Typically, these packages are dictated by the actual size of the lighting element. Common annotations are 5050, 2812 and 3528, with the chart below showing examples of Surface Mount Device (SMD) LED's on strips.



**Figure 15 - LED Size Comparison. Courtesy Flexfire LED's.**

Addressable LED's are able to be individually controlled by sending a series of bits to the on-board shift registers that contains information such as brightness and what segment to light up (RGB) to get the desired colour or hue. The information is clocked through and can either be changed each clock or left the same and selected to change later on. The advantage of this is that you can get a large amount of control and displays to happen with simple coding and libraries.

The approach that was undertaken with this project was to control a single LED at first and to determine whether it was a viable option to continue onto the strip lighting. This has the effect of keeping cost to a lower level (individual LED's as opposed to a whole strip purchase) and also the result at first would not have to consider parallel signal paths through multiple LED's. This is something that would need to be considered for future work.

#### 4.2.2.2 *DotStar vs NeoPixel*

The two main addressable LED's on the market are produced by AdaFruit Industries, which is a company that was set up in 2005 to capture an increasing niche market in hobby electronics (AdaFruit, 2016). They manufacture and supply microcontrollers (Raspberry Pi, Arduino, Beaglebone) and supporting components such as cables, chips, break-out boards, electroluminescent panels, development boards and LED's.

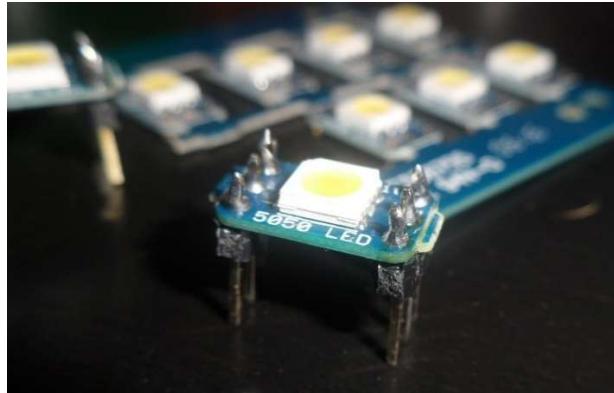
Their two most popular LED's are branded NeoPixel and DotStar. The NeoPixel's are effectively 5050 RGB LED's with an integrated driver chip inside a WS2811 LED. They have a strict timing requirement and are better suited to a 'real-time microcontroller or interpreted microcontroller' (AdaFruit, 2016), such as the Arduino, but not the RPi. This fact, coupled with the preference to use the Raspberry Pi, eliminated the slightly older and slower NeoPixel LED from being used on the project.

This left the newer DotStar branded APA102C smart LED as the preferred addressable option. They are sold as a strip and in single form (10 pack) from the AdaFruit website (AdaFruit Industries, 2016). They come in two colour temperatures (cool or warm) and in RGBW or WWWW element configuration. The WWWW option was selected to provide a more consistent and pure output compared to blending the 3 RGB elements to get white light out. The APA102C LED's have integrated shift registers and PWM encoders (iPixel LED, 2014) embedded into the LED that enable the user to set the brightness once and it is provided to the strip or singular LED until you change it.

The shift registers receive DATA IN and then pass it to DATA OUT pins, onto the next LED at the chosen clock speed (up to 32MHz). The PWM rate is 20 kHz and this was a key aspect that supported the decision to use the APA102C. If you are modulating the LED, you don't want the human eye to perceive the flicker and draw negative attention to that fact, so smooth blending is required for the concept to be effective. Herein lies one limitation of addressable LED's over conventional LED's. The DotStar requires a 32 bit signal to turn ON/OFF. So to transmit one bit of information (1 or 0) the LED must be turned ON/OFF accordingly. The coding must send 32 bits to the GPIO via the SPI to produce that ON or OFF state. Therefore if you operate at 32MHz SPI then your maximum data rate is 1MHz.

The DotStar's are supplied as a surface mount device (SMD) and to use with bread boards and prototyping boards, they were required to be soldered to a breakout PCB. The PCB's were specifically made for the purpose and required the LED terminals (6 of) to be attached using solder paste and a hot air gun. The mounting legs were soldered using traditional methods before the LED was attached to prevent overheating of the lower melting point solder paste. Three LED's were mounted to enable trials of sequentially connected LED's and the effect on data reception.

AdaFruit have an extensive support page for NeoPixel's and DotStar's with Phillip Burgess' (2016) popular Uber Guide providing an industry bible of sorts for hobbyists starting out on addressable LED projects. The Uber Guide covers mainly NeoPixel's but the principles can be applied to the DotStar as well. It includes detailed topics of power supplies, Arduino and Raspberry Pi Libraries, form factors, connections, coding and best practices.



**Figure 16 - APA102C LED Mounted to Breakout PCB.**

#### *4.2.2.3 DotStar Connection*

As the APA102C is a +5V device, and the Raspberry Pi has +3.3V GPIO signals, a level shifting chip is required to ensure the correct operation. The AdaFruit nominated level shifter is the Texas Instruments 74AHCT125 chip. This is a quad level shifter (4 signals) and connection is made by simply applying +5V to the VCC pin, grounding the GND and OE pins, then your input goes to pin A, with the shifted output at the corresponding pin Y. The chip was purchased in the Dual In Line (DIL) 14 pin package for easy breadboard mounting and installation.

As SPI was being used, both the SCLK and MOSI signals were sent to the level shifter before the LED. As with the Uber Guide recommendation,  $470\Omega$  resistors were applied between the GPIO pins and the inputs to the level shifter chip as protection against higher voltages being applied to the +3.3V GPIO's.

#### *4.2.2.4 FPGA*

With a view to reducing the code overhead and processing of 32 bit signals all the time I did look at Field Programmable Gate Arrays (FPGA) to hold the 1's and 0's to send to the chip rather than coding. After research into the operation of the addressable LED, it was decided that the bits were being processed by the shift register anyway and an FPGA held no improvement by sending from memory location as opposed to SPI or coding. This would have come at extra cost as well ( $>\$20$ , plus voltage considerations, plus size and surface mount etc.).

#### *4.2.2.5 LED Light Output*

In the event that LED did not produce sufficient and coherent light, the LED could have the light focussed with the use of lens, reflectors or collimators.

A lens with take light from a source and focus to a focal point determined by the characteristics of the lens. Lens work by refracting the light and the pattern and effectiveness depends on the materials used

as much as the size and shape of the lens. Lens do suffer from losses as the light enters a denser medium. This is the same as collimators.

A collimator uses an aperture to direct the light source into a parallel and coherent beam, thus giving greater focus, with no focal point. This has the effect of putting down a spot. The construction of the collimator needs to be precise in nature as there are many aspects of beam direction that need to be combined to be effective.

Reflectors use internally reflective surfaces to bounce the light in a specified pattern. They are typically pre-set in the way that the light interacts with the reflective surface and are sometimes complex in shape. A big plus though is that because the light doesn't enter the surface, a very large portion of the light energy is preserved. This however can be set off by the disadvantage that the light travelling vertically from the LED is not directed and remains on a spreading pattern.

#### 4.2.3 Programming & Code

Python was chosen as the programming software due to the broad technical support, availability with Raspberry Pi, being a simplistic language and of relative ease to pick up.

The code that accompanies addressable LED's are typically meant for Arduino microcomputers due to the fact that these addressable strips are just simply controlled rather than modulated with data. There are examples of code for Raspberry Pi, but they tend to relate to more complex display and lighting projects. These typically revolve around the persistence of vision applications and those with moving signs or LED banners.

##### 4.2.3.1 Software Diagram

Below is a software diagram that illustrates the intended structure of the Python code within the transmission module. The dashed boxes indicate functionality that wasn't able to be implemented during the project timeframe, but does show the intent of the code. The dotted arrow shows that a bitstream of the file data was able to be transmitted successfully.

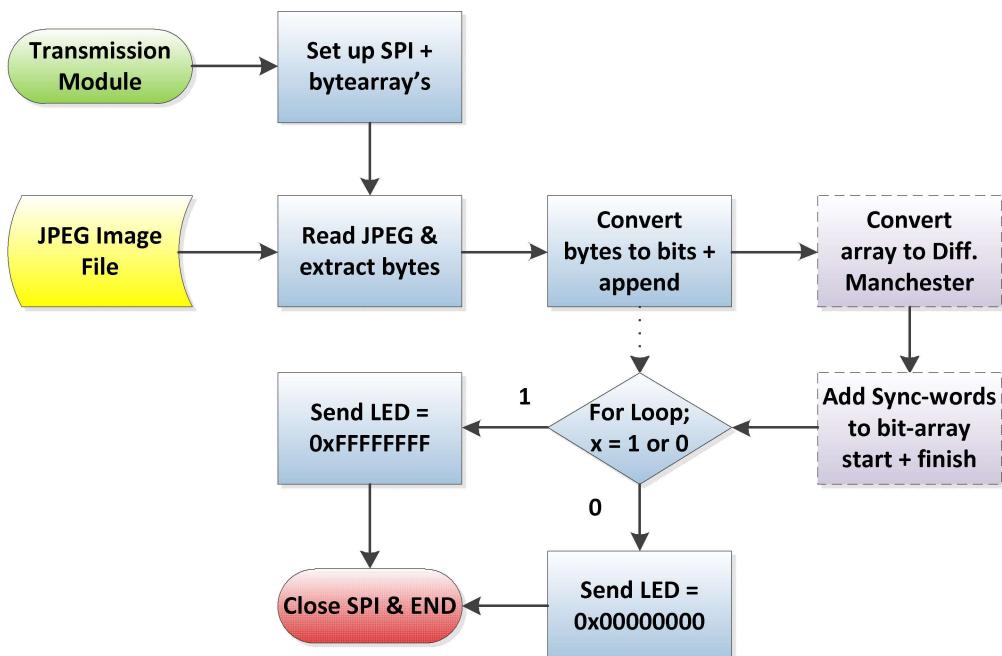


Figure 17 - Software Diagram for Transmission Module

#### 4.2.3.2 Encoding the File Data

The simplest way to encode data such that long strings of similar bits can be understood is Manchester Encoding. This method is data heavy in the fact that it transmits two bits for every one bit of data to send. For example to send a ‘1’ bit, the code must send ‘01’ and conversely to send a ‘0’, you need to send ‘10’. The benefit of this method is that the most number of the same bit type that you have together is two. If you want to send two bits such as ‘10’ the data you send is ‘0110’, so therefore each bit that you send has one state transition (from 0→1 or 1→0).

#### 4.2.3.3 Sync-words

The sync-words help to identify the start and finish of the data file. These are applied to the bytearray once the bitstream from the file is created. They are to be transmitted as well. The sync-words should be discernible from the Differential Manchester sequence so a normal data sequence is not taken to be the finish of file sync-word. It must also be a mixture of 1’s and 0’s to prevent the receiver module starting to store data in the receiver bytearray because a false signal has been seen.

### 4.2.4 Cost

With a view to keeping the project accessible to students, all components were purchased from mainstream electronic outlets. A Table of the costs associated with the construction of the transmitter module is shown below.

Transmitter module			
Item	Qty	Cost	Purchased from
Raspberry Pi 2B + case	1	\$49.65	RS Components
APA102 LED (DotStar)	10	\$7.34	Core Electronics
5050 LED Breakout boards	10	\$7.40	Core Electronics
74HCT125 Quad Level Shifter	1	\$2.22	Core Electronics
470Ω Resistor	8	\$1.08	Jaycar
Prototyping PCB	1	\$5.50	Jaycar
GPIO Jumper wire kit	1	\$5.95	Jaycar
5V, 1A Power Supply + socket	1	\$19.90	Jaycar
Wi-Fi Dongle	1	\$5.00	eBay
<b>TOTAL</b>		<b>\$104.04</b>	

Table 3 - Costs Associated with Construction of the Transmitter Module

## 4.3 Receiver Module

The receiver module required more considered selection of components than the transmission module. This was primarily due to the photodiode configuration, the signal amplification and the RPi GPIO requiring protection against over-voltage. The selection of the photodiode drove all of these requirements, much like the LED drove the selection of its coding and circuitry.

Another influence on the choice of the receiver circuit components was to make sure that each stage would not present a limiting factor to the ability of the prototype to receive high speed signals. In order for the LED to continue to be used for domestic use, it would need to be perceived as completely ON the whole time.

### 4.3.1 Raspberry Pi

The Raspberry Pi was of the same configuration as the transmission module, right down to the cases used to protect the RPi board against damage and contamination. Perhaps the only difference was the wireless dongle that was used and the desktop background (TX = white, RX = black) to tell the difference when viewing them on the same monitor.

#### 4.3.1.1 *GPIO configuration*

The MISO (Master In Slave Out) pin (BCM 9) was used on the GPIO to receive the signal from the photodiode circuit.

Due to the low power requirements of the amplifier circuit, the on-board +5V GPIO pin 2 was used to power the receiver circuit. The Transimpedance amplifier was selected partly due to its low drain to reduce the need for an additional power source.

The RPi GND pin 6 was used for the circuit to ensure common ground connections.

### 4.3.2 Photodiode Circuit

#### 4.3.2.1 *Photodiode Selection*

Research into a suitable photodiode for this application started early and the selection criteria included, low cost, high speed, +3.3V to +5V, used for optical circuits, low capacitance, low noise, possible DC bias compensation, small form, low dark current and accessible via a recognised commercial source such as RS Components, Element14 and Jaycar.

The photodiode had to have low current requirements due to the Raspberry Pi GPIO limited to 16mA each and 50mA total. If this couldn't be adhered to then potentially another power supply would need to be included, which is not desirable from a portability point of view.

Another key characteristic is the Terminal Capacitance  $C_t$ . Terminal Capacitance is created at the PN junction and it is one of the factors that determine response speed of the component. This contributes to the Time Constant of the device, along with the load resistance ( $R_L$ ). The formula to determine this is;

$$T_C = 2.2 \times C_t \times R_L$$

The value of this capacitance varies with the size of the active area, such that it increases proportionally if the area increases. This is particularly important when considering the overall capacitance of the

amplification circuit and the choice of the feedback resistor. If the capacitance is too high, you will need to add a feedback capacitor to compensate and this also may limit the frequency response and gain of the circuit. Other photodiode's considered had Terminal Capacitance as follows;

Name	Manufacturer	$C_t$	Notes
SFH 213	OSRAM	11pF	1.3V device
C30872	RCA Inc.	12pF	APD
APD110A2	Thorlabs	-	>\$300
VS1838B	Thorlabs	6pF	10-20V device
FCI HR005	Osi Optoelectronics	0.8pF	No local suppliers
BPW34	Vishay	25pF	Flat package
S5343	Hamamatsu	15pF	APD. 250MHz
<b>S5973</b>	<b>Hamamatsu</b>	<b>1.6pF</b>	<b>Si PIN. [Selected]</b>

Table 4 - Comparison Table of Photodiode Terminal Capacitance.

The Terminal Capacitance ( $C_t$ ) was just one factor in the selection process. There exists a large number of technical considerations for photodiodes such as construction, configuration, application, dark current, photo sensitivity, noise equivalent power and spectral response.

A decision was taken to look specifically PIN photodiodes as they were found to be better suited to VLC products and operated better in saturated environments, such as those with interior lighting. After considerable searching the Hamamatsu S5973 PIN photodiode was chosen. The S5973 is a high speed (1GHz) device that is listed as low cost, high sensitivity and is used in optical fibre and high speed photometry applications (Hamamatsu, 2003). It comes in a TO-18 package and operates at +3.3V. The S5973 series operate across the required spectral range (320-1000nm,  $\lambda$ ) and has peak sensitivity around 760nm, which is ideal for optical usage.



Figure 18 - Hamamatsu S5973 Photodiode (TO-18). Refer Datasheet.

#### 4.3.2.2 Photodiode Bias

When connecting the photodiode within the circuit, there are two choices regarding the bias applied to the diode. These are reverse and normally biased, which can be referred to as photoconductive and photovoltaic biasing respectively.

Photoconductive biasing is where the diode has an external voltage (or bias) applied to anode (-ve) and cathode (+ve) of the photodiode. This causes the depletion zone of the PN junction to increase and there is a corresponding decrease in capacitance. Reducing this capacitance also reduces the noise gain at high frequencies, which is where this circuit is expected to operate.

Advantages of this method include increase responsivity and linearity of the sensor, meaning they are better suited to faster switching applications. This does come at a cost and leads to an increase in dark current as well as noise currents.

Dark current is the current that flows when no light is applied to the sensor region. The bigger the PN region, the greater the movement of random electrons, which also increases with temperature. Given that the photo currents are so small, the ideal is to minimise dark current to as little as practical. This is can be by design (small sensor area), materials, compensation and simply applying a zero bias across the photodiode.

This is known as photovoltaic biasing. This method works on the principle that the voltage across the diode is maintained at zero volts and the PN junction is able to set up electron and hole flow that is proportional to the amount of light that falls on it. Using a zero bias configuration does reduce the amount of noise and dark current, such that they don't have a significant effect when the received signal is amplified. The preference for this project was to utilise the zero bias method for this reason. Consideration was given to the high switching suitability of a photoconductive biased set-up and this is something that can be trialled for future work.

#### 4.3.2.3 Transimpedance Amplifier

The very small photo current that is produced by the photodiode is not suitable to pass directly to the Raspberry Pi GPIO. It needs to be converted to a voltage and amplified. For this project the output. An efficient and simple way to achieve both requirements was with a Transimpedance Amplifier (TIA). They usually consist of an operational amplifier (op-amp) with a feedback resistor ( $R_f$ ) back to the inverting input (-ve). The value of  $R_f$  sets the gain of the TIA, and comes with other considerations when deciding on the value. Depending on the terminal capacitance of the connecting photodiode, a feedback capacitor  $C_f$  may also be required to prevent gain peaking.

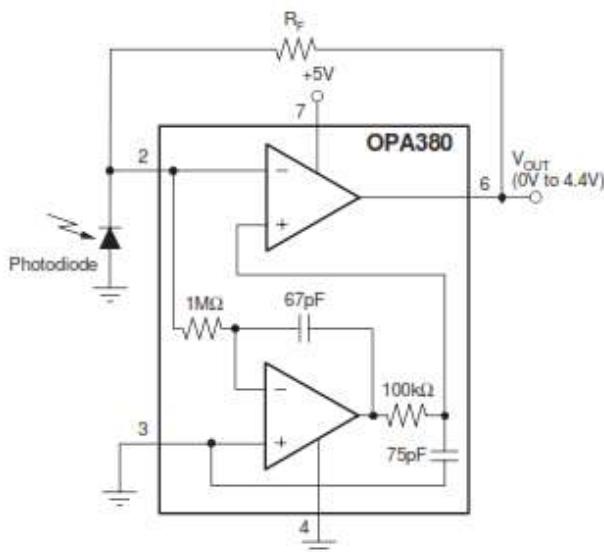


Figure 19 - Connection Circuit of OPA380 Transimpedance Amplifier

With  $R_f$  connected to the inverting input, the inverting input (+ve) is connected to ground. This configuration offers the greatest chance of noise, but also the highest gain. With a view to having the prototype working at distances of 1-2m, I opted for the high gain configuration and looked at other ways to reduce noise.

When locating the Transimpedance amplifier, particular attention was required to minimise stray capacitance. This additional capacitance adds to your circuit and can reduce the bandwidth of the amplifier. Guard traces should be used where possible, but in this case I used a small prototyping Printed Circuit Board (PCB) that did not have one.

The TIA was connected to +5V as a single supply to get the maximum available gain bandwidth. Again, this to ensure the data could still be recognised as logic levels at a reasonable distance.

The TIA can be set up to operate back down to 0V to get a true swing, but I chose not to do that as it meant extra components and it wasn't a requirement to detect logic high and low transitions. As long as the voltage dropped down under +1V, the Raspberry Pi would detect a logic '0'. Anything above +1.7-2V would be seen as a logic '1'. That said, adding a return to zero voltage swing would need to be considered for some encoding methods and those that would be subject to a high level of ambient light causing a higher DC bias seen on the signal. This lift in DC bias could cause issues with the RPi interpreting some signals correctly. The fact that the Raspberry Pi does not have a -5V output also drove that decision.

Setting the value of the feedback resistor is a balance between the gain required and the noise sensitivities of the circuit. For this prototype, a single gain stage was preferred to improve noise performance, as adding more amplifier stages degrades this noise performance. Having just a single amplifier stage also improves the signal to noise ratio, primarily around the signal across the resistor increasing linearly and the noise characteristic increases exponentially ( $\sqrt{R_f}$ ).

The OPA380 TIA datasheet (Texas Instruments, 2007) includes graphs to relate the Transimpedance gain to the operating frequency and the feedback resistor plus diode capacitance. The applicable graph for the S5973 photodiode ( $C_{diode} = 1.6\text{pF}$ ) is shown as the Figure below. Utilising a  $1\text{M}\Omega$  feedback resistor with no feedback capacitor, we can achieve a gain of up to 120dB across most operating frequencies. Above 1MHz, this does roll-off but the gain is still >90dB up to 10MHz.

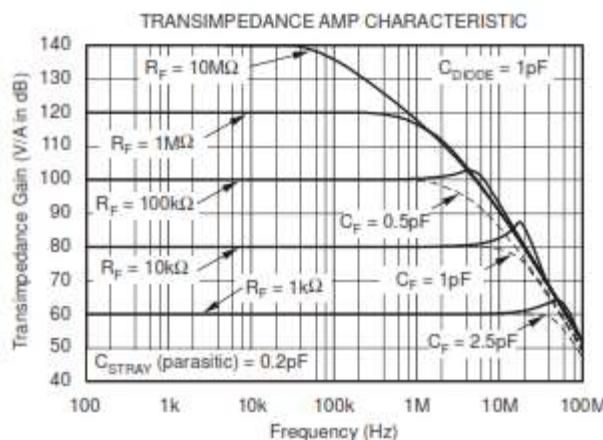


Figure 20 – Texas Instruments OPA380 TIA Graph to Select  $R_f$  Value

#### 4.3.2.4 Zener Protection Circuit

In order to stop the maximum output (+4.4V) from the photodiode and the amplifier reaching the Raspberry Pi's GPIO's and doing damage, a method of protection was required. This needed to cap or restrict the output of the amplifier to the GPIO's limit of +3.3V. The fact that it would 'clip' a signal was not a concern given that the receiver was dealing in digital signals rather than audio or sinusoidal signals.

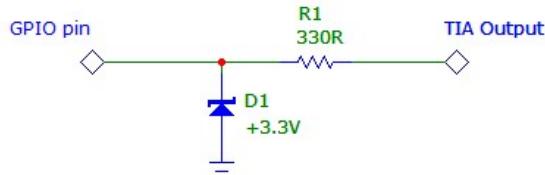


Figure 21 - GPIO Zener Protection Circuit.

A decision was taken to utilise a +3.3V Zener Diode and a  $330\Omega$  resistor in the arrangement below. The way that this circuit provides protection is that when a voltage greater than +3.3V gets to the Zener, the diode breaks down and it shorts it to earth. Conversely, if a negative voltage is applied, then the diode will only conduct a maximum of -0.7V to the GPIO, which is hardly damaging. The  $330\Omega$  resistor just limits the current flow in and out (10mA) as additional protection.

#### 4.3.3 Programming and Code

The code is required to 'listen' to the GPIO once the script is run and when the pre-defined sync-words are seen, the data is captured in an array for file reconstruction. This file is written to and can be opened or displayed once the processing has finished.

##### 4.3.3.1 Software Diagram

Below is a software diagram that illustrates the intended structure of the Python code within the receiver module. The dashed boxes indicate functionality that wasn't able to be implemented during the project timeframe, but does show the intent of the code.

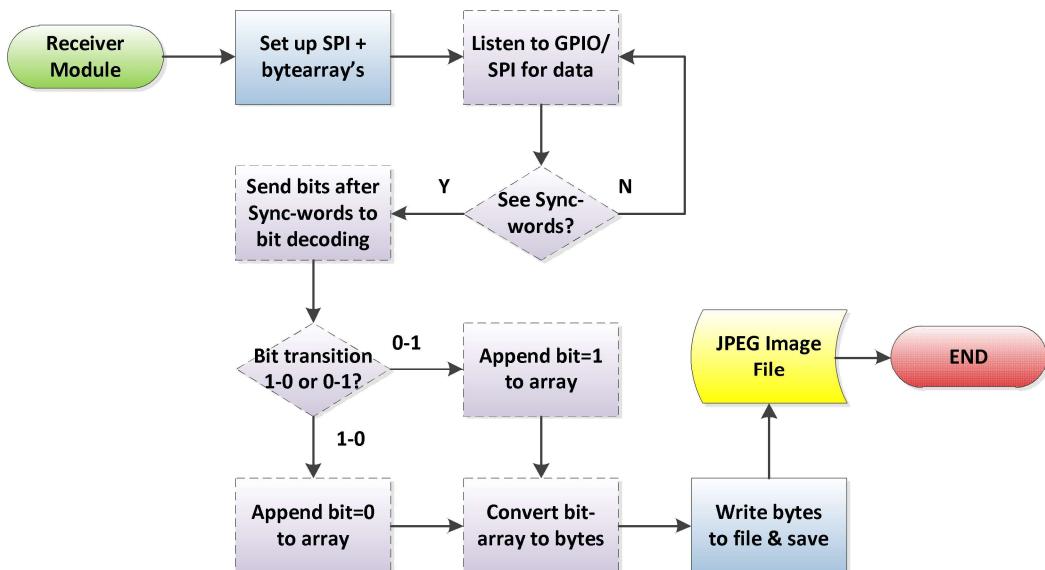


Figure 22 - Software Diagram for Receiver Module.

#### 4.3.3.2 Sync-words

The script listens to the GPIO for a predetermined sequence of bits. They are pre-programmed into the script and a LOOP is applied to the GPIO once it receives a logic 1 signal. Once it detects the first signal, it checks the next one and if it receives the correct sequence, will store the next bit into a bytearray to be passed to the decoding loop.

The sync-word must be chosen to be discernible from the Manchester sequence and also individual enough to not be mistaken as someone shining a torch on the photodiode. For example it shouldn't be a long string of 1's and should be broken up to include a 0 or two to prevent a false positive.

#### 4.3.3.3 Decoding

With the transmitter encoding the data as Differential Manchester, the receiver must also operate with this coding method. The Python script takes the data that is stored in the received array (post sync-word detection) and runs it through a LOOP to check for transitions between the logic 1's and 0's. When the appropriate transition is detected, the loop outputs the correct value to another bytearray to be written to a file.

#### 4.3.3.4 File Reconstruction

Once the bitstream has been separated and sent to an array, it is restored to bytes and then written to a file. This file is then able to be opened or displayed. The code was created at the same time as the transmission module code and tested as one script. This ensured that if the data was at the GPIO, that the script could return it to the original structure.

### 4.3.4 Cost

A table showing the cost to build the receiver module is shown below. The most expensive item is the photodiode, and for this one item, cost was not a leading factor in the decision to utilise it. The reason behind this was that it should not be your bottleneck to receiving and processing modulated light. Having a highly specified photodiode meant that there was no additional power supply, there was no additional feedback resistor for the TIA and that it could handle speeds up to 1GHz.

Receiver Module			
Item	Qty	Cost	Purchased from
Raspberry Pi 2B + case	1	\$49.65	RS Components
Hamamatsu S5973 photodiode	1	\$54.50	Element 14
OPA380 Transimpedance amplifier (TIA)	1	\$12.54	Element 14
Aries SOIC to DIP breakout PCB (for TIA)	1	\$5.85	Element 14
1N4728 3.3V Zener diode	1	\$0.65	Jaycar
330Ω Resistor	8	\$0.54	Jaycar
1MΩ Resistor	8	\$0.55	Jaycar
Prototyping PCB	1	\$5.50	Jaycar
1uF Capacitor	1	\$0.55	Jaycar
Wi-Fi Dongle	1	\$5.00	eBay
TOTAL		\$135.33	

Table 5 - Costs Associated with Construction of the Receiver Module

## 4.4 Prototype System

Until now we have talked about the transmission and receiver modules separately, but they are part of a system. Without each other, they are reasonably useless. Pictured below are the two modules, with the transmitter on the right and the receiver on the left.

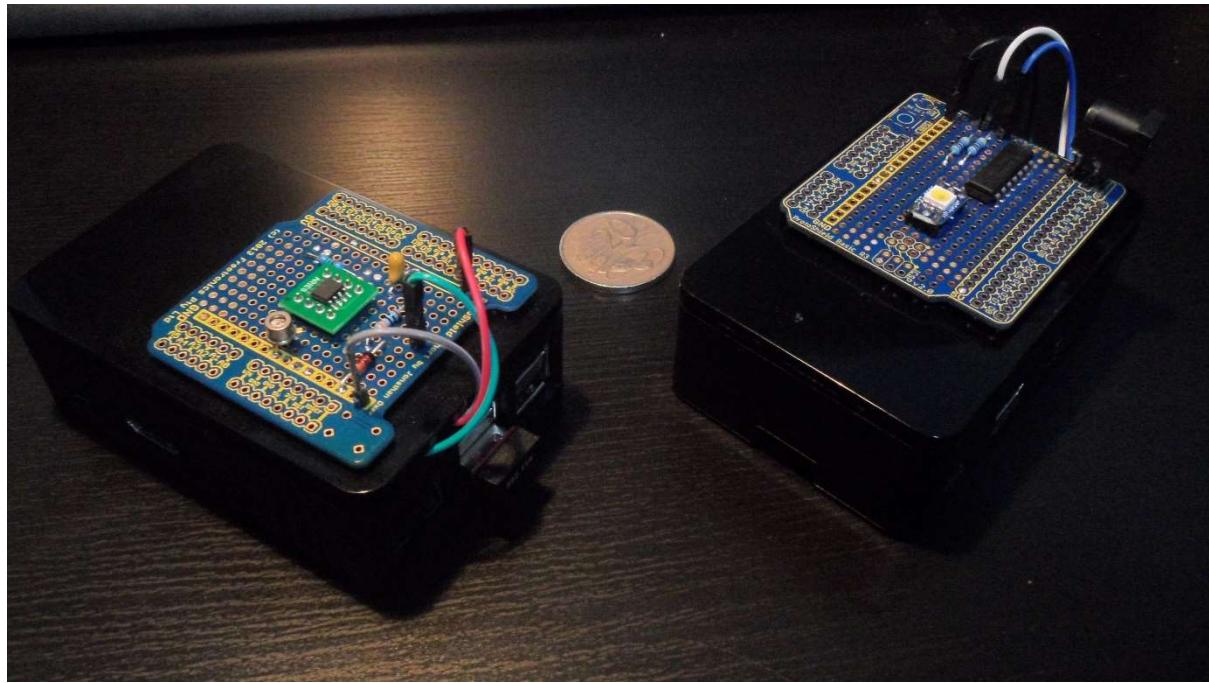


Figure 23 - Image of Li-Fi Prototype

The small size of the two modules means that they could sit in a ceiling (behind a light), or on a desk without being too cumbersome. Ideally, these modules should be as small as practical. If the transmitter was to be connected to an LED strip light, then the transmission module PCB would not need a separate power supply. The PCB could also be made significantly smaller and easily be contained inside the RPi case, with wires attaching to the strip light for DATA and SCLK.

The intent was to put the LED and photodiode elements outside of the Raspberry Pi case, and have the circuit boards inside. The GPIO wiring points would obviously have to change, but this is not difficult to introduce.

For the final testing in this state, the modules were placed on their sides, and positioned on a flat surface. This way the transmit and receive elements were relatively level with each other. Due to the beam focusing requirement (to maximise light power) this was critical to getting good signals generated in the receiver module.

The following section covers the testing and analysis of the prototype results to arrive at the final configuration.

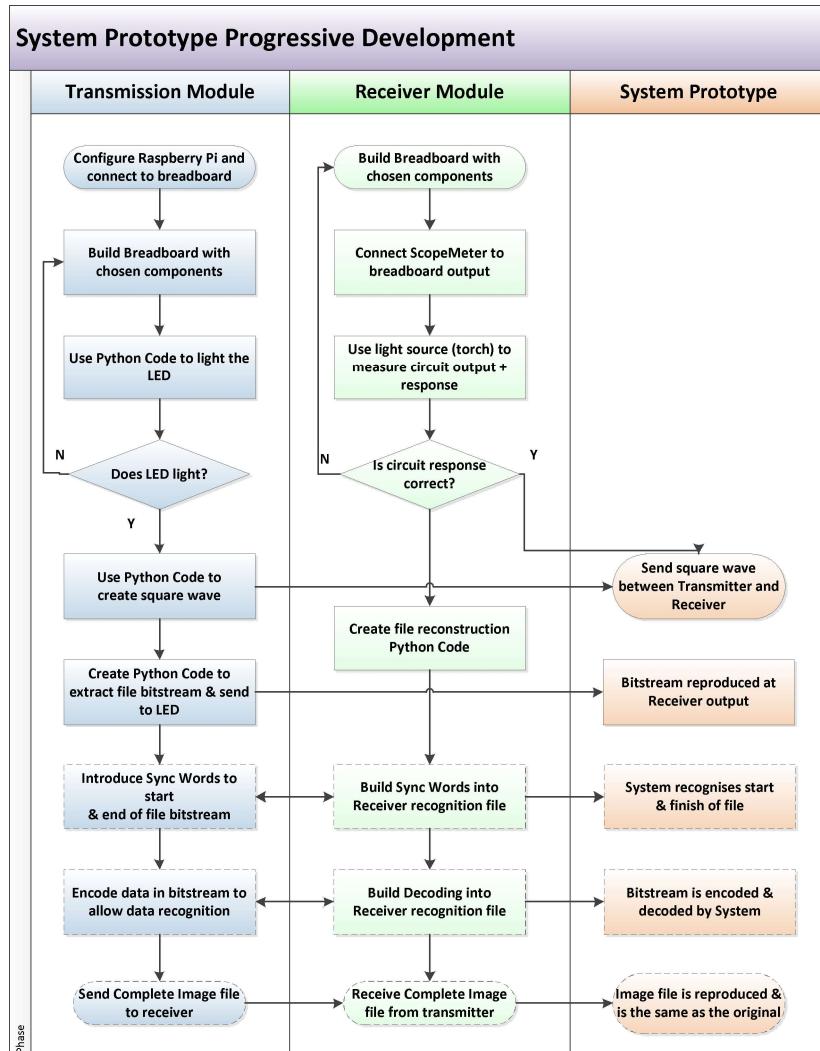
## 5 RESULTS ANALYSIS

This section will cover the processes and testing outcomes encountered during the project. Note that many of the desired test results and activities were unable to be completed as both Raspberry Pi modules had boot problems and they were inoperable for the final stages of the project. This manifested as solid ACT lights on the Raspberry Pi boards.

Due to the significant effort required to recover the information stored on the modules, and/or re-program the Raspberry Pi's, a decision was taken to finalise the achieved results and include any incomplete testing and analysis as discussion points throughout this section.

### 5.1 Analysis Approach

The development for the prototype had to be structured to build up the capability of the design and ensure robust investigation as each stage came together. This can be illustrated by the diagram below which shows the method of progressive development for the system.



**Figure 24 - Developmental Approach for System Integration & Prototype Analysis**

Note that the activities in the dashed boxes were not able to be achieved in the timeframe of the project.

While this proved to be successful, it did mean that the larger components were developed linearly rather than in parallel. This resulted in more time spent on those aspects than the prototype as a whole. A very large part of the development was spent on the transmission module and getting the LED to light. There was a significant lack of literature that brought the various facets together for the addressable LED, SPI, Python coding (and library) and general techniques to execute this. The majority of the literature was around just turning the addressable LED strip on and to display colour sequences, rather than modulating it for data transmission.

To verbalise the diagram above, the process to arrive at a working prototype was as follows;

- Place transmission module components on a breadboard and get the LED to light with a Python script,
- Place receiver module components on a breadboard and use a flashing torch to ensure correct signals are created at the output,
- Create a modulated (square wave) signal with the transmission module,
- Send the square wave transmission to the receiver module to check signal levels and amplified output from the receiver module,
- Build Python script to deconstruct a file into a bitstream such that it can be sent to the LED loop for transmission,
- Transmit the file bitstream to the receiver and confirm correct sequence received,
- Decode the received bitstream, and using a Python script, write to a file for processing and display.

The majority of these activities were successful, and the results will be discussed in the following sections.

## 5.2 Tooling

The primary tool used on the project was the Fluke ScopeMeter 192C. This allowed me to view the frequency and level of signals at each part of the circuit. This included the SCLK frequency, the data being sent to the LED, the signal received and amplified by the receiver circuit and the output of that circuit to the Raspberry Pi GPIO. This was critical when trialling new components and diagnosing circuit problems or restrictions.



Figure 25 - Fluke ScopeMeter 192C.

It did have some listed limitations as it had a 60MHz bandwidth and a 500MS/s sample rate, but these did not impact the signals that were involved in the prototype. It was noticed that the leads were susceptible to 50Hz interference and had to be connected to a good ground source, particularly when measuring small signals.

With any electrical project that involves PCB and components, a good soldering iron is essential. With both of the PCB, there was a reasonable amount of soldering to be done and to get a professional finish, you would also need some flux, a tip cleaner, some isopropyl alcohol, Scotchbrite (to clean the solder) and a lint free cleaning cloth.

The Transimpedance amplifier and LED were Small Outline Integrated Circuit (SOIC) devices which meant they needed to be surface mounted to a Dual-In-Line (DIP) breakout PCB to be used. Because of the small legs to be soldered, solder paste was the only viable option to attach these components. Solder paste comes in a small tube and can either be lead-free or lead solder. Small quantities are applied to the legs of the components and a hot air gun is applied until the paste melts and joins the two surfaces. It really is an easy process and produces good results. The key is to not overheat the component and to prevent from reheating already soldered parts. Tinfoil or some heat deflective material can be useful for this purpose. Examples of the solder paste method are shown below.

As well as soldering equipment, a good pair of side-cutter and long nose pliers can help with bending device legs and mounting small components.

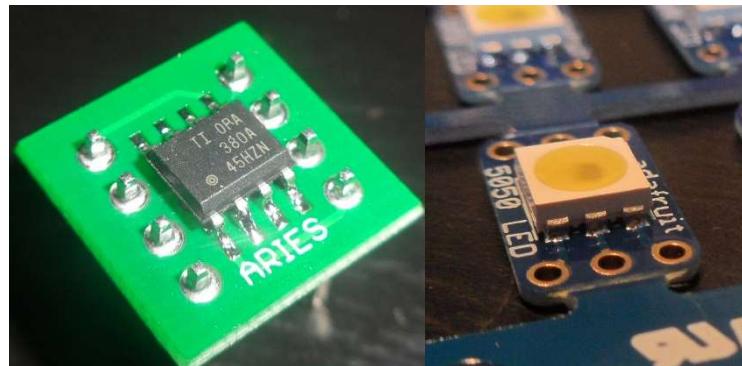


Figure 26 - Solder Paste applied to mount SOIC Components to DIP PCB's.

An unexpectedly useful (and later, necessary) tool that is required for diagnostic purposes was an LED torch or light source. A small tactical LED torch was used that had an in-built flash function. This was very useful to test the receiver circuit to ensure it received a signal with the correct frequency and light intensity. The light output was very bright (in fact brighter than the DotStar LED) and was able to be focused to a point on the photodiode.

For the early circuit construction, a breadboard complete with wires was necessary to swap out and test various components. However, these are prone to capacitance and while useful, circuits should be transposed to PCB's as soon as practical. The breadboard used at the start was limiting the bandwidth of the circuits and affecting the frequency of the transmitted and received signals.

A base was constructed to securely fasten the transmitter and receiver to allow objective testing.

## 5.3 Transmission Module

This section looks at the testing and analytical decisions taken to build the final configuration of the transmission module.

### 5.3.1 Transmission Circuit (PCB)

After deciding to utilise the DotStar LED (APA102c) and the recommended quad level shifting chip (74HCT125), the circuit was constructed on a breadboard. A representative layout of the breadboard is shown in the figure below. Note that this was the final layout and configuration after power supply testing, which is covered in the next section.

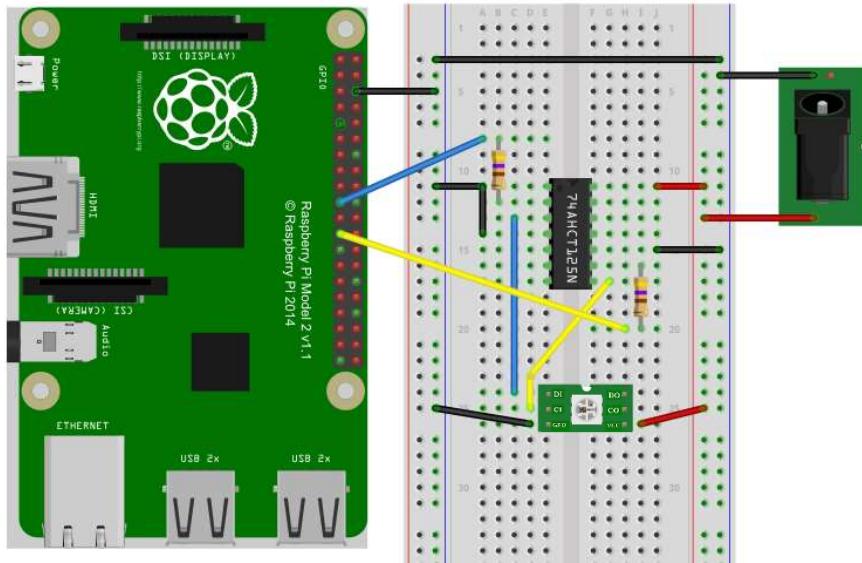


Figure 27 - TX Module Layout Diagram.

- **GND (Black wiring);**
  - All ground connections were made common between the +5V power supply and the Raspberry Pi.
  - All OE pins used on the 74HCT125 are grounded.
  - APA102C pin 3 is grounded.
- **+5V Power Supply (Red wiring);**
  - +5V power supply connected to breadboard rail, VCC pin 14 of 74HCT125 and VCC pin 4 of APA102C.
- **DATA Signal (Blue wiring);**
  - +3.3V RPi GPIO MOSI pin 19 (BCM10) is connected to pin 1A of the 74HCT125 chip via a  $470\Omega$  resistor.
  - +5V level shifted DATA from 74HCT125 pin 1Y goes to APA102C pin 1 (DI).
- **SCLK Signal (Yellow wiring);**
  - +3.3V RPi GPIO SCLK pin 23 (BCM11) is connected to pin 3A of the 74HCT125 chip via a  $470\Omega$  resistor.
  - +5V level shifted SCLK from 74HCT125 pin 3Y goes to APA102C pin 2 (CI)

### 5.3.2 Power Supply

DotStar LED's are a +5V device and the initial plan was to utilise the Raspberry Pi's +5V power supply to light the LED and power the quad level shifting chip. On AdaFruit's DotStar Overview webpage (Burgess, 2016), they discuss that it is possible to power the LED's off +3.3V but that it is not recommended. I attempted to do this initially, but was unable to get a single LED to respond.

For the first trials, the Raspberry Pi +5V supply from the GPIO was used, with light output significantly lower than what would be expected from a bright LED. Without modulation, the LED was observed to flicker. Through research this was found to be a result of the Raspberry Pi's current sharing scheme with other internal processes. The RPi was powering a Wi-Fi USB dongle, running Python code on the desktop, using SPI, and also providing power to the quad level shifter. This unreliable nature of current sharing forced the purchase of a separate +5V, 1A power supply.

Although this was openly discussed on online forums as necessary when powering an entire LED strip, it did improve the brightness and robustness of illumination from the circuit. Implementing a separate supply was at the expense of my compact prototype objective, but could be removed on future revisions by a higher rated USB supply or a new model RPi with USB3.0.

### 5.3.3 GPIO Protection

In order to prevent against damage due to inadvertent supply of current back through the output GPIO's (MOSI, SCLK),  $470\Omega$  resistors were placed between the GPIO and the input to the level shifting chip. Because the LED was powered by the +5V power supply, the value of resistor had no bearing on the brightness, which was set by the 32 bit data signal.

To see if there was an effect on the data and SCLK signals from the GPIO,  $330\Omega$ ,  $470\Omega$  and  $1k\Omega$  resistors were tested as protection. The value of resistor had no measureable or detrimental effect on the signal strength or quality coming out of the GPIO. The quad level shifter also just boosted the value to +5V and passed to the LED.

### 5.3.4 Lighting the LED

With the circuit connected to support illumination of the LED, various methods were researched to provide the 32 bit signal to the circuit to light it. The two methods that had the most supporting literature were using the AdaFruit Python library and using SPI.

The first method attempted was using the AdaFruit Python library and the basic script 'strandtest.py'.

#### 5.3.4.1 Strandtest.py

This Python module is available for download on their site (AdaFruit Industries, 2016) and offers an alternative to the better supported Arduino library. This download includes the required library, a simple 'strandtest.py' script to test your connections and a Python Image module example for light painting which your strip light. All of these scripts cater to LED strips rather than single LED's.

The 'strandtest.py' code was used to get the LED to light for the first time. This verified that the connections and configuration of the transmission module were correct. The download comes with a README.txt file that helped to get started. Of course, the library must be installed on the RPi before trying to run the code.

The first issue encountered was with the strandtest.py code was that it was designed for an LED strip and not a singular LED. This was overcome by setting the ‘`numpixels`’ variable to 4. Next the ‘`datapin`’ and ‘`clockpin`’ variables needed to be set to the GPIO pins that you intended to use.

From there, it was decided to set the SPI at 32MHz to get a fast transmission speed. This option was enabled by ‘uncommenting’ the code per the instructions. Next was setting the brightness to 255 to get the maximum available light. The code started with the recommended setting of 64, but testing was carried out to explore the effects of increasing it to the maximum. The light output made one appreciable increase in brightness from 64-128, but above that it was very similar for each increase. All further testing was conducted at 255 (max) brightness, with no considerations to current draw due to the separate power supply.

The variable ‘`color`’ was set to 0xFFFFFFFF to ensure that the LED’s were displaying white light rather than a blend or a lighter shade. Later, this was determined as not critical as this variable was more pertinent to output colour from an RGB LED rather than a WWW one which was being used for the project. The ‘`time.sleep`’ value was increased to (1.0 / 100000) to give a faster frame rate.

The code was then run from the RPi Terminal command line and the LED was able to light. Changing the ‘`time.sleep`’ value allowed a square wave to be sent to the circuit and proved that the module was capable of data transmission, as a first validation activity.

The difficulty with the strandtest.py script was that it had limitations with getting a bitstream of data fed into the code to turn the LED on or off dependent on each bit. What the code was doing was running through a strip of ‘one’ LED and the sleep timing value was being altered to light that one LED again and again. This was confirmed by looking at the GPIO output and seeing that the ‘off’ time was longer than the ‘on’ time. Given that there was no apparent timing synchronisation or control over using the strandtest.py script, it was abandoned in favour of a more timing orientated method embedded within the Raspberry Pi.

#### 5.3.4.2 SPI

Of the other methods reviewed to send data to the LED, SPI was the best supported and most widely used. In order to utilise SPI to handle the data transmission, specific Python code was located on GitHub (2016) created by user ‘doceme’ and called ‘py-spidev’. This code formed the basis of a simple yet effective script to use the SPI to turn the LED ON/OFF. The code from user doceme was modified and is copied below.

```
import spidev  
  
spi = spidev.SpiDev()  
  
spi.open(0,1)  
  
resp = spi.xfer([0x00, 0x00, 0x00, 0x00])  
  
resp = spi.xfer([0xFF, 0xFF, 0xFF, 0xFF])
```

The code lit the LED, but it did not provide a square wave signal. This was solved by adding a loop that was dependent on the bits that would come from the file extraction.

An issue was encountered with the data set produced. The image below shows the output of the GPIO MOSI pin with the previously listed SPI code of `resp = spi.xfer([0xFF, 0xFF, 0xFF, 0xFF])`. This can

be seen as 4 distinct blocks of data separated by at least clock pulse. Note the +3.3V peak to ground voltage. The probe setting was on 10:1 rather than 1:1 and was a legacy from the last user.

As a result of this image the code was improved to have the intermediate bits removed (e.g. `resp = spi.xfer([0xFFFFFFFF])`) and this alleviated the gaps in the data to the LED.

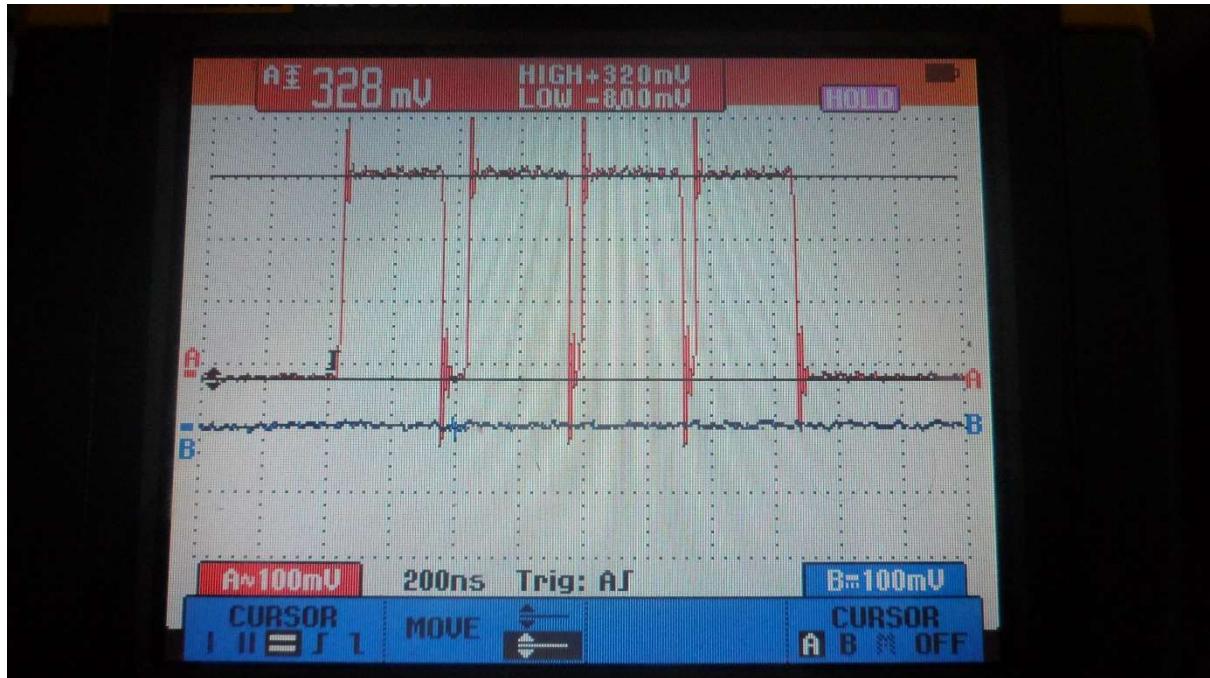


Figure 28 - GPIO LED Data Signal with SPI

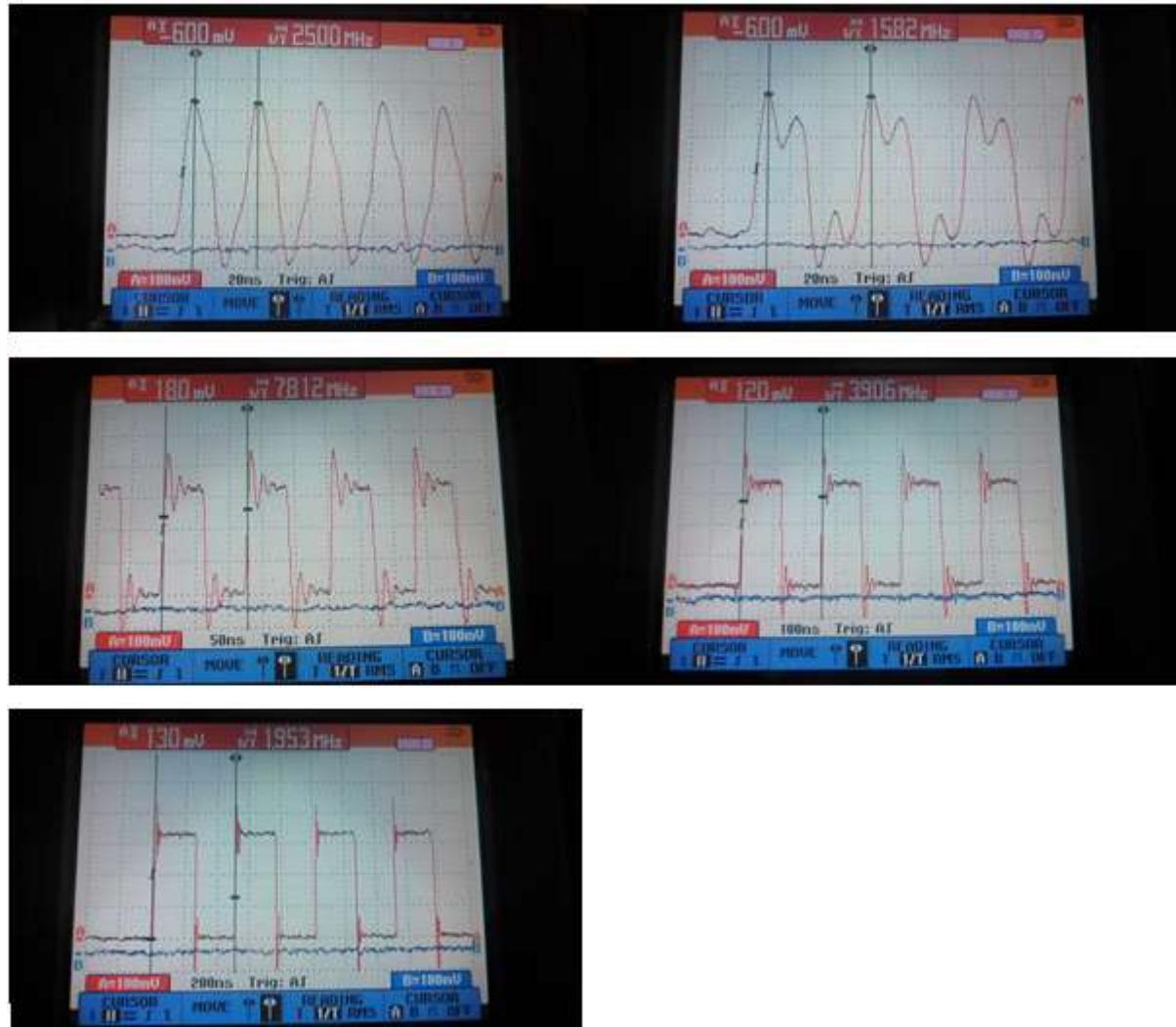
The SPI code from above formed the bits transmission portion of the code. This was far simpler than selecting and GPIO and sending it to that GPIO. SPI is also capable of sending bytes, but it was decided to get the bitstream working correctly first and then reduce coding overhead by taking out that conversion (bytes to bits) at a later point in time. There is no evidence to say if this would actually work as the bits would need to be modulated and not sure that sending bytes to SPI would necessarily translate to a valid 32 bit ON signal. Regardless, SPI was used at a bit level and it performed adequately.

#### 5.3.4.3 SPI SCLK

With the limitation of sending a 32 bit signal to transmit (via LED ON) one bit of data, the clock rate (SCLK) plays a key role in transmission. The Raspberry Pi has SCLK speeds from 125MHz down to 7629Hz (Raspberry Pi Foundation, 2015). This can be set using the CDIV parameter or simply via a command in your Python script. This command is '`spi.max_speed_hz = 31200000`' and will set the SCLK to 31.2MHz.

A number of trials were conducted with setting the SCLK value and measuring the output of pin 23 (BCM 11, SCLK), plus recording the waveform. The results showed that the signal deformed the higher it went. Regardless of the signal quality the prototype was able to use the 31.2MHz speed to send square waves near 1MHz speed. This illustrated that perhaps the measurement of the waveform was causing the deformation rather than the Raspberry Pi outputting such an irregular clock signal. See over for

recorded waveforms. Note that the displayed voltage levels are not relevant, as they are taken off the trigger.



**Figure 29 - SPI SCLK GPIO Output Waveform Comparison**

To get anywhere near nil flicker the 31.2MHz SCLK setting was utilised. This would give a theoretical transmission speed (of bits) of 1MHz. If Manchester Coding or another bit hungry coding method was to be used, then this would be halved or at least reduced further.

### 5.3.5 Addressable LED Testing

With this aspect being one of the keystones of the project, there was considerable effort into the testing of the LED. As well as trying to get the best out of the prototype, there was genuine curiosity with regards to addressable LED operation.

#### 5.3.5.1 Brightness & Luminous Intensity

Once the LED was able to be lit with the Python code and components, the LED was observed to be bright to look at on the maximum setting. All tests were carried out with the 32 bit brightness code set to 255 or maximum brightness. This ensured the best performance as far as possible distance of transmission and light output was concerned.

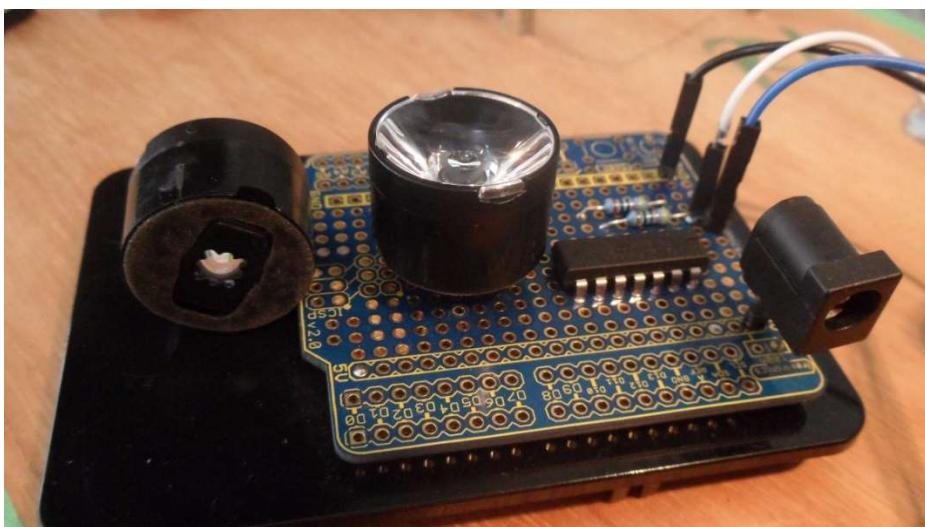
The DotStar LED is listed at having an output of 18 lumens (Shiji, 2014) and on reflection this is a low value for a singular LED. Other addressable LED's are slightly less (SK6812 = 6-7 lumens, WS2812b = 12-13 lumens) for singular units, but when combined these low values are inconsequential. When comparing these to normal LED's and to other chip-based LED's, there are few performance benefits of using addressable LED's. The Cree XLamp discussed in Section 2.3 (Table 2) is 260 lumens at a supply voltage of +3V and could have been a more suitable candidate in hindsight. The drawback of this type of LED is the high current requirements, meaning that it could not be powered directly off the Raspberry Pi GPIO (16mA max). Given that a separate power supply was required for the DotStar LED anyway, the current constraint for the Cree LED could have been overcome. There are also regular 'through-hole' (T-1) LED's that have a high luminous intensity and match the supply and configuration requirements of the prototype (e.g. Cree LC503FWH1).

The project did not have access to test equipment to measure the actual luminous intensity from the DotStar, but using the datasheet values (angle = 125° and luminous flux = 18Lm) the luminous intensity is 5.32 cd. The Cree LC503FWH1 from above has a luminous intensity of 24cd, so is significantly brighter and more direct.

When testing a single DotStar in a dark room, it was not sufficiently bright enough to illuminate the entire room. Testing was conducted on the breadboard using 3 LED's and this improved the illumination, but at least 10 LED's would be required to provide enough light to work under. This is where the DotStar is expected to operate and not just as a single element.

#### *5.3.5.2 LED Light Directivity*

Given that the DotStar had such a wide angle of light output (125°) the intensity of the light falling on the photodiode, was very low. At first, it was barely perceptible when measuring the amplified output at the RPi GPIO. Rather than change the transmission element and the coding to get it working at that stage of the project, it was decided to utilise optics to focus the light specifically onto the photodiode. This was a departure from one of the guiding principles behind the prototype, whereby the addressable LED needed adaptation to operate successfully as the transmitter. Additionally, the lens was positioned in front of the LED such that it focused the light to a point at a fixed distance. This initial distance was 200mm, which was considerably under the objective of greater than 2m of free space between modules. Regardless, this was chosen to get the prototype working and performance enhancements would occur if time permitted.



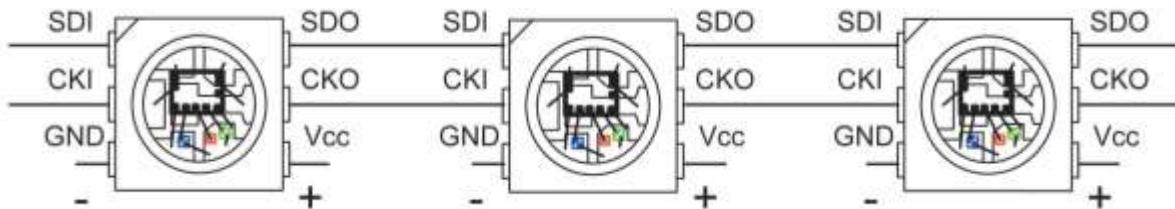
**Figure 30 – Collimator used to increase LED directivity**

A lens of unknown physical characteristic was first trialled to good effect with the LED. A beam was able to be directed to the photodiode with a size of approximately 10mm at a distance of 200mm. Through deeper research and a desire to achieve greater separation between the modules, a number of collimators were purchased to trial. Narrow beamwidths of 5°, 10° and 15° were selected to give a sharper point, thus allowing the LED to be further away from the receiver. It was considered to have the collimator sit on the photodiode instead of the LED, but it was never trialled.

The collimator and holder sit over the LED as shown in the figure above. If the LED was surface mounted, the holder would sit flush with the PCB, making for a cleaner aesthetic. Due to time constraints around the completion of the project, more formal test results of collimator beamwidth, distance and focal point were not produced. Informally, the collimator out-performed the simple lens and allowed for increased distance between the two modules. This was informally 100-200mm over the lens arrangement. The increase in receiver module output voltage was not measured for a comparison at a fixed distance.

#### 5.3.5.3 *Multiple LED's*

As part of the investigation into the suitability of addressable strip LED's for light modulation, it was decided to test 3 LED's in a similar set-up as would be encountered on the strip. This was conducted at the breadboard stage. The primary LED was left wired to the circuit as normal, but the additional LED's Data In (DI) and Clock In (CKI) were connected to the Data Out (DO) and Clock Out (CKO) of the preceding LED. The connections are as per the figure below.



**Figure 31 - Multiple LED connection example (Shiji, 2014)**

The strip LED's are designed to have the same information sent to all LED elements simultaneously and pulsed through with the clock signal. The data signal is sent to each one near instantaneously so there should be nil perceivable delay in the transmission of the modulated light from each LED. What you should see is coherent transmission of the information from the strip.

What I saw (but did not record on the ScopeMeter) was 3 distinct and separate square wave signals at the receiver circuit GPIO. These presented as overlapping square waves, but could be identified by small overshoot on the rising edge of each pulse. This meant that although each signal was being emitted at the same time with the same information, it was being received by the photodiode out of phase and incoherent. The effect of this was to cause corruption of the data being sent in the fact that a square wave of a specific period was now masked by the over-lapping other square waves.

On reflection, this is simple wave theory. There was one receiver (photodiode). There were 3 LED's mounted on a flat surface (breadboard) at a distance away from the receiver. Each LED would have been at a different distance away from the receiver to the other LED's. This is what caused the signal overlap at the receiver.

This project discovery is not a new phenomenon with Visible Light Communication and there are many papers that cover the significant mathematics around the topic. This however has not been applied to addressable LED strips. What this means is that if a signal was sent out via the strip, and the proximity of each LED to the next, there would be data corruption as a result. Normally, with competing domestic LED light sources, a method of locking onto a particular channel or source is implemented. The transmitter and receiver check for signal strength and packet loss, and determine if a hand-off to another channel is required. Again, there are significant numbers of papers on this topic. For this to be applied to an LED strip would create unnecessary processing overhead unless there was a clear communication degradation that resulted in a hand-off to another source. That is, it would have to be significantly low to avoid the constant hopping between LED's as your preferred data to receive. For this reason, addressable LED strips are not currently suitable as transmitting elements in a Light Fidelity system.

### 5.3.6 File Processing

To send anything via the prototype it needs to be in a format that can be sent with an LED turning on or off to represent the data.

#### 5.3.6.1 File Conversion to Bits

In order to transmit the contents of a file, the binary information that makes up that file needs to be extracted and provided to the transmission circuit. The first file to be trialled was a small jpeg image of the raspberry pi logo (called [logo.jpg](#)). The image is 238 by 212 and is 6.8kbytes. This file type was chosen as it was the easiest file type to extract binary data from. The file was saved on the Raspberry Pi transmission module and the name, including location had to be manually entered into the python script before being run. The plan was to make this automatic and possibly select the file or even run an executable from the command line to send the file from start to finish.

Firstly, SPI parameters were set up and two arrays were created to allow storage of the bits during extraction and after being received. The file was opened using the '`rb`' (read binary) annotation and read into a bytearray called '`bytes`'. From there two for loops took the bytes information a byte at a time and converted that byte into bits. Inside the bit loop the bits were appended to one of the previously created arrays. An IF/ELSE decision then processed each bit according to the binary status (1,0). If the bit was 1 then SPI would send '`resp = spi.xfer([0xFFFFFFF])`'. For a 0, '`resp = spi.xfer([0x0000000])`'. This series of bits was transferred to the LED circuit and the LED was thus modulated according to the bits within the file.

#### 5.3.6.2 File Rebuild Code

To make sure that the method used to extract and rebuild the file was correct, the code was developed as one script to start with. The receiver portion would then be cut out and dropped into the script on the receiver module, with the knowledge that if the data was received, it could be reconverted back into a file correctly. This would reduce the amount of debugging on the receiver side, as it had previously been working without encoding or transmitting bits across free space.

The buffer created to append the extracted bits to, was then split into bytes and the bytes were then rearranged into the same structure as previous and stored in another array called '`picture`'. A new file called '`relogo.jpg`' was opened using the '`wb`' command (write binary) and the file was rewritten with the picture array. The file was placed in the same directory and was opened to show nil errors or loss of data when compared to the original file.

This code is listed in Appendix B1.

### 5.3.7 Transmitting File Data

#### 5.3.7.1 Code

Now that the LED and the code to extract the bits level data from the file were working, the methods had to be brought together. This was done through a FOR loop that the ‘buffer’ bytearray data was passed through. This code is shown below.

```
for byte in bytes:  
    for i in range(8):  
        bit = (byte>>i) & 1  
        buffer.append(bit)      #send bits to the LED using SPI  
        if bit == 1:  
            resp = spi.xfer([0xFFFFFFFF])  
        else:  
            resp = spi.xfer([0x00000000])
```

For the initial debugging, the data was printed to the screen as the bits came through, which made it easier to see whether the transmission was correct or not. As per previous sections, the data was sent via SPI out the MOSI pin on the GPIO.

#### 5.3.7.2 LED Performance

With the clock set to 32MHz, the data was sent through at around 850 kHz as measured on the ScopeMeter. This was first checked with a square wave. At that speed, there was a very minor perceptible flicker with the LED. Due to time constraints around the project, there was no time to diagnose the source of the flicker. It could have been caused by script timing, physical circuit characteristics (current) or LED buffering (shift registers/PWM). Research online was unable to find any literature that highlighted any frequency limits of operation for the DotStar.

During the initial trials, a 1MHz signal was being transmitted to the LED and it didn’t appear to be flashing. Using a digital camera, the modulation of the square wave was able to be detected. This picked up the pulsing of the light and can be seen as pulsing artefacts around the LED in the image below.

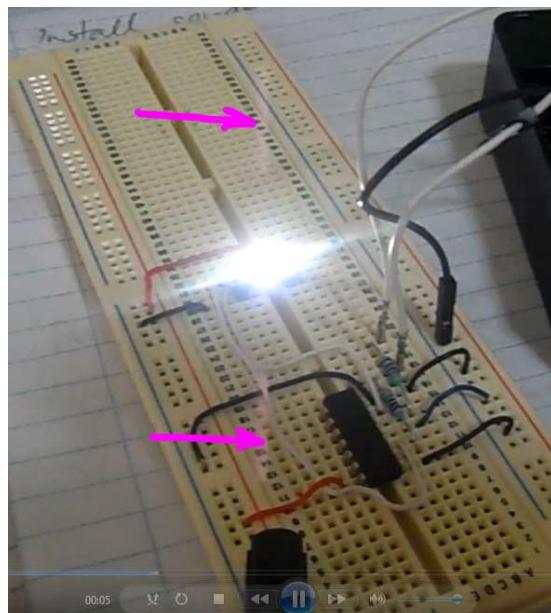


Figure 32 - Video screen capture of 1MHz modulation

### 5.3.8 Data Modulation

With the transmission of real data came the most significant hurdle to the success of the prototype. Modulation and data encoding had previously been researched and discussed on the project. It was not completely understood how to implement this at a coding level for the application.

An example of the bitstream is shown below. This was captured from the ScopeMeter at the input to the receiver module GPIO. Note the long sequences of 1's and 0's. The code had a lack of a method to separate those sequences and extract the data to its original state. Due to time constraints to deliver the project and the modules having boot problems, this was not able to be rectified.



Figure 33 - Example of Long Data Sequences

By way of a simple implementation of Manchester encoding, the Python script was amended to transmit a '01' if the data bit was a 1 and a '10' if the data bit was a 0. This was unsuccessful and the root cause was unable to be determined. The attempted code is shown below.

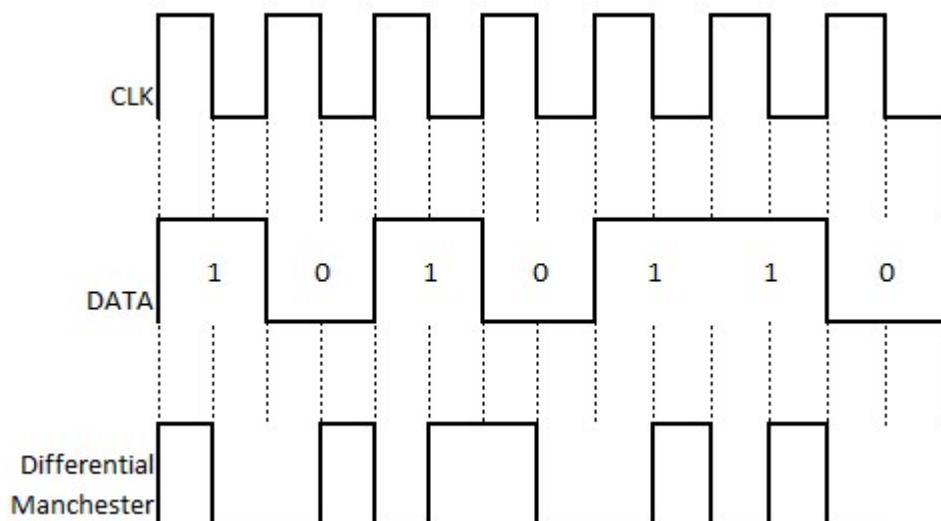
```
for byte in bytes:  
    for i in range(8):  
        bit = (byte>>i) & 1  
        buffer.append(bit)      #send bits to the LED using SPI  
        if bit == 1:  
            resp = spi.xfer([0x00000000])  
            resp = spi.xfer([0xFFFFFFF])  
        else:  
            resp = spi.xfer([0xFFFFFFF])  
            resp = spi.xfer([0x00000000])
```

Manchester coding is bit hungry in that to transmit one bit of data, you need to send two bits. This reduces your transmission speed by half, but with a 32MHz clock speed, would still give you a speed in the region of 500 kHz. This is still quick and should be above any perceptible flicker frequency.

Should there have been more time, potential implementation of differential Manchester coding scheme or simple Manchester code that transmitted the transmission SPI clock speed as information to set the receiver module clock speed by.

#### *5.3.8.1 Differential Manchester*

Differential Manchester coding is where the original clock signal is XOR'ed with the data to allow clock recovery. Combining them means that there is always a ‘transition’ for each bit. This transition can either be a rising or falling signal, which can be detected by the receiver. Python and the GPIO can be used to detect those rising or falling signals and convert them back into the original data structure.



**Figure 34 - Differential Manchester Encoding**

This transition almost removes the reliance on a signal strength value and also helps to combat ambient light conditions causing a lift in the ‘DC bias’ applied to the signal.

The script would be structured inside a LOOP that takes the buffer bytearray bits and applies conditions dependent on whether the bit is a 1 or 0. Another bytearray would be created to store the encoded data. The first output bit would be set to 1 and then the buffer bytearray is checked. If it is a 1, then the output is set to 0. The next step is to wait half a clock cycle, then set the output bit to 1. Another half cycle is waited and there is a check to see if the buffer bytearray is finished or not. If not, then check the next bit and so on. Once the buffer bytearray is processed, it has the sync-words applied and sent to the LED LOOP to be sent.

On the receiver side, the script looks for the sync-words, and then checks for data transitions. For each transition checked, an output bit goes to another bytearray, which gets written to a file.

### *5.3.8.2 Manchester Encoding*

This requires further coding work to include synchronisation with clock speed to decode the signals. Again, two bits are required to be sent for every bit of data. To overcome clock synchronisation, the SPI SCLK value in the transmitter can be given a binary value and be added to the front of the file. This value is seen after the sync-word and it is used to set the receiver SPI SCLK speed before decoding starts. The issue that I see with this is that it is possible that the transmission speed is not the same as the speed related to the SCLK. It could be off a small degree due to circuit components or process inefficiencies. To improve the accuracy of this method, a timing ‘reset’ byte could be included at regular intervals of the data stream. This helps to get it back on track and reduce the error rate. The difficulty would be to make the timing bit of a sequence not likely to be replicated by data, e.g. ‘1111’.

Other encoding methods would need to be binary based as you can’t send carrier wave. Methods such as varying the ‘amplitude’ of the light might work with fixed applications, but not suitable for those involving movement or changes in the environment. For future work, it is recommended to implement Differential Manchester encoding.

### **5.3.9 Raspberry Pi & MobaXterm**

One aspect that was not able to be assessed, was the performance of the Raspberry Pi CPU time allocation and its effect on transmission performance. There are many papers and forums discussing this deficiency of the RPi and the preference to use Arduino for real time processing.

It was noticed that when using MobaXterm to view the remote desktop of the modules, that if both were being used, the commands on those desktops became quite laggy. Perhaps this was wireless performance rather than RPi CPU performance issues.

Apart from the boot problem at the end of the project, the Raspberry Pi’s performed as expected. This may be enough to trial an Arduino for its simplicity in action.

## 5.4 Receiver Module

This section looks at the testing and analytical decisions taken to build the final configuration of the receiver module.

### 5.4.1 Receiver Circuit (PCB)

#### 5.4.1.1 *Receiver Module Breadboard Layout*

After component selection was completed, the circuit was constructed on a breadboard, but was not connected to the RPi GPIO until the output levels were known.

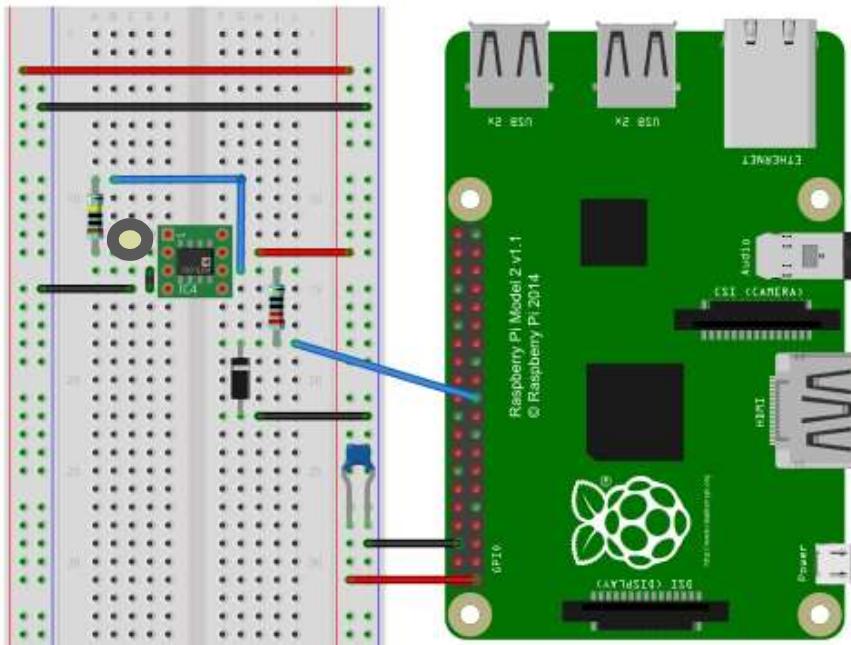


Figure 35 - Receiver Module Breadboard Layout

- **GND (Black wiring);**
  - All ground connections were made common.
  - Anode of S5973 Photodiode grounded.
  - Pins 3 and 4 of OPA380 TIA are grounded.
  - Anode of 1N4728 Zener grounded.
- **+5V RPi (Red wiring);**
  - +5V supply from RPi GPIO pin 2 connected to breadboard +V rail.
  - Pin 7 of OPA380 TIA connected to +5V.
  - 1uF ceramic capacitor connected across +5V rail and ground.
- **DATA Signal (Blue wiring);**
  - 1MΩ resistor connected across Pins 2 and 6 of OPA380 TIA.
  - Cathode of S5973 Photodiode connected to Pin 2 of OPA380 TIA.
  - Cathode of 1N4728 Zener connected to Pin 6 of OPA380 TIA.
  - 330Ω resistor connected between OPA380 TIA Pin 6 and RPi GPIO MISO pin 21 (BCM9).

## 5.4.2 Photodiode & Amplifier Testing

The first aspect of the receiver circuit that was checked was the connection of the photodiode and the ability to receive signals. The photodiode by itself produces a very small photo-current that wasn't able to be detected on the ScopeMeter. This meant that to achieve a larger signal, the Transimpedance amplifier was required to be connected up early.

### 5.4.2.1 Photodiode

The photodiode was connected in photovoltaic mode and photoconductive mode was considered but not tested. That said, it would have been a worthwhile exercise to test in both modes and note the level of operation, speed, noise and general suitability to the application. This will be recommended in future work.

The photodiode was connected with the legs as short as possible. This was to help cut down on noise added to the tiny photo-current and being amplified with the data signal. It was also connected straight to ground, with the shortest distance to the ground connection point, again to minimise noise.

### 5.4.2.2 Transimpedance Amplifier

To connect up the Transimpedance amplifier, there were only a couple of parameters to change and adapt for the circuit. These included the feedback resistor, feedback capacitor and any resistor biasing if required. As the prototype didn't have the facility to apply a reverse voltage, the biasing option was not needed. Aside from that fact (it could have been created), with the small levels of DC bias and the logic levels for the Raspberry Pi, there was no real need to have the signal totally return to 0V. Perhaps if there was a requirement to reduce the transmission power by having a signal go negative, this could have been an option to implement.

With the photo-current proportional to the amount of light on the photodiode, the goal was to have as much gain as the circuit would permit, without overdriving any components or inducing unnecessary noise levels. The TIA was fed with +5V signal from the RPi GPIO and the Open Loop Gain of the TIA is up to 110dB with an output voltage of Vcc-0.6V. This meant that the maximum output would be +4.4V which is what was recorded during testing. Of course this was clamped at +3.3V with the Zener protection circuit, which is discussed later in this section. The idea was to have as much gain as practical in order to enable the maximum distance between the two modules and still produce a viable signal.

Using the OPA380 datasheet (Texas Instruments, 2007), specifically the graphs on page 7, values of the feedback resistor were selected for testing. These values were  $10\text{k}\Omega$ ,  $100\text{k}\Omega$ ,  $1\text{M}\Omega$  and  $10\text{M}\Omega$ . In addition, values for the feedback capacitor were also trialled. The ceramic capacitor values chosen were  $0.5\text{pF}$ ,  $1\text{pF}$  and  $2.2\text{pF}$ . A test was conducted with a  $100\text{nF}$  capacitor for the  $10\text{M}\Omega$  resistor as one was already on hand.

The testing was carried out at a distance of 250mm with an LED torch on full light or off. The torch was a standard 3x AAA battery powered LED torch with a super bright LED. The output was a typical 1-3W and of the kind purchased at any hardware store. The light was focussed to a point on the photodiode. The ambient light conditions consisted of a ceiling light and possible contribution from a PC monitor. The light was not modulated, so there was no frequency component with the signal measured. A table of results is shown below.

What the results show is that the larger the Rf resistor, the greater the gain. This is in line with common understanding for a Transimpedance amplifier circuit. The value of Cf did have a small effect, but ideally should have been tested under a frequency signal to see the effect on signal roll-off or

attenuation. In order to achieve maximum gain with tolerable ambient light levels, the  $1M\Omega$  resistor and nil  $C_f$  capacitor was selected for the circuit.

Test	Rf	Cf	Output OK	Ambient Light	Full Light	Suitable?
1	$1M\Omega$	1pF	Y	0.47V	2.2V	Y
2	$10M\Omega$	1pF	Y	1.25V	4.4V	N
3	$10M\Omega$	100nF	N	-	-	N
4	$10M\Omega$	0.5pF	Y	0.8V	4.3V	Y
5	$1M\Omega$	-	Y	0.2V	4.4V	Y
6	$100k\Omega$	0.5pF	Y	0.1V	3.0V	Y
7	$100k\Omega$	-	Y	0.1V	3.1V	Y
8	$10k\Omega$	1pF	Y	0.1V	1.8V	N
9	$10k\Omega$	0.5pF	Y	0.1V	1.7V	N

Table 6 - Results Table of TIA Feedback Resistor and Capacitor Testing

What the results also show is that the other specified values from the OPA380 datasheet, also provide good amplification and ambient light performance. However, the full light values appear to be limiting when referenced to a distance of 250mm at maximum light conditions. This could be overcome with a second amplifier stage but increases the risk of additional noise being added and amplified to the final signal.

#### 5.4.2.3 Circuit Testing

With the selection of the TIA components, the receiver circuit was ready for testing to see the response of a frequency based signal. For all receiver circuit testing, the output of the circuit was not connected to the Raspberry Pi to eliminate potential damage. In order to see circuit performance, the Zener Diode protection circuit was not introduced. It was important to determine the final configuration before implementing that in the event that it was either not required or needed to be changed depending on output levels. This follows the principal of changing one variable at a time to observe and act to changes attributable to that one variable.

Initial receiver circuit testing was conducted with an LED torch that had a built in flash function. This was a 2Hz flash rate, but the purpose was to observe replication of square waves rather than full light level appearing as just a ‘DC’ signal. The response of the circuit was excellent and it was able to produce a very clear signal with the correct parameters when measured by the ScopeMeter. The torch had a focussing lens on the front and this was again focussed on the photodiode to give maximum lighting levels. Distances from the torch to the photodiode were not formally recorded, but were (informally) in the region of 400-500mm, whilst still receiving a valid square wave signal. Increasing the distance required the torch focal length to also be changed to achieve the best possible signal output.

Integration with the transmission module to prototype testing is covered in the next section.

#### 5.4.2.4 Ambient Light

The prototype test environment was a small room with filtered external light. However, the majority of the testing took place at night, so artificial light was the main source of ambient light. With the torch testing, the signal was still strong and clean despite having contributing light from an 18W LED ceiling

light, a PC monitor and a halogen desk lamp shining on the receiver. This ambient light presented as a DC bias on the output, but was constant so could be filtered out if required. Many TIA come with DC Offset filtering, including the OPA380, but the signal levels didn't warrant pursuit of this parameter. It could become essential when signals are being received at larger distances between the transmitter and receiver. Note that if the selection of the feedback gain resistor is too high, it will also amplify the ambient light bias and have a greater effect on the final output signal. Consideration therefore needs to be given to the environment that the prototype is operated in.

Additionally, man-made light will have a frequency component to it due to the power source. I did see a 50Hz component on the ambient light signals when expanding the time base of the ScopeMeter. This component was not present under sunlight only. If this was an issue, then a high-pass filter could be applied to block any signals <100Hz. Testing would need to be conducted though to ensure that any useful signal components were not also blocked.

#### *5.4.2.5 Zener Diode Protection.*

As already discussed, a Zener Diode protection circuit was chosen as a simple method to prevent the signal coming from the TIA overdriving the Raspberry Pi GPIO and causing damage. A 3.3V Zener connected across a  $330\Omega$  resistor provided over-voltage protection. From an amplifier output of 4.24V the circuit clamped the signal down to 3.44V which is tolerable for the GPIO to handle.

### **5.4.3 Python Coding**

The premise behind the receiver circuit coding was to listen to the GPIO, particularly looking for a series of defined sync-words that signalled the start of a genuine data set. At present the two Python scripts were required to be opened from the command line, but it was planned to have them enabled or woken up by the transmission module. The more independent the prototype is, the better the user experience and take-up of the idea.

Unfortunately, due to time constraints with the project, the receiver module code was not able to be implemented nor conduct any meaningful tasks. Significant elements of the script were not put in place and the only working part was the file post processing.

#### *5.4.3.1 File Processing*

The portion of code that takes the decoded output, converts it back into the original structure and writes it to a file, was completed and working with the transmission module code. This code was then planned to be dropped into the end of the receiver script to either open or display the transmitted file.

```
picture = bytearray()          #this is where I store the final bits
buffer = bytearray()           #put the bits in an array to rebuild
#convert back to right structure// for RX module
for i in range((len(buffer)/8)):
    bitstobytes = buffer[i*8:i*8+8]
    byte = 0
    for x in xrange(8):
        byte = byte + pow(2*bitstobytes[x],x)*bitstobytes[x]
    picture.append(byte)
#write to file and display result
refile = open("relogo.jpg", "wb")
refile.write(picture)
```

## 5.5 Prototype Module Integration

With the finalisation of the individual modules, the true performance needed to be tested as a prototype system. Without the integration of these two aspects, the suitability for a Li-Fi system could not be assessed. This section covers the testing that took place with the transmitter and receiver sending data via light modulation.

Below is an early set-up for the prototype integration testing and this provided with some control over distance and lighting level variables. The focussing lens can be observed on the right hand side, which was set to a fixed position such that the focal point was on the photodiode consistently. Without this level of consistency, assessment of results and signal states could not be made objectively. While it was far from ideal, it was necessary to ensure correct function for each modules role in the prototype.

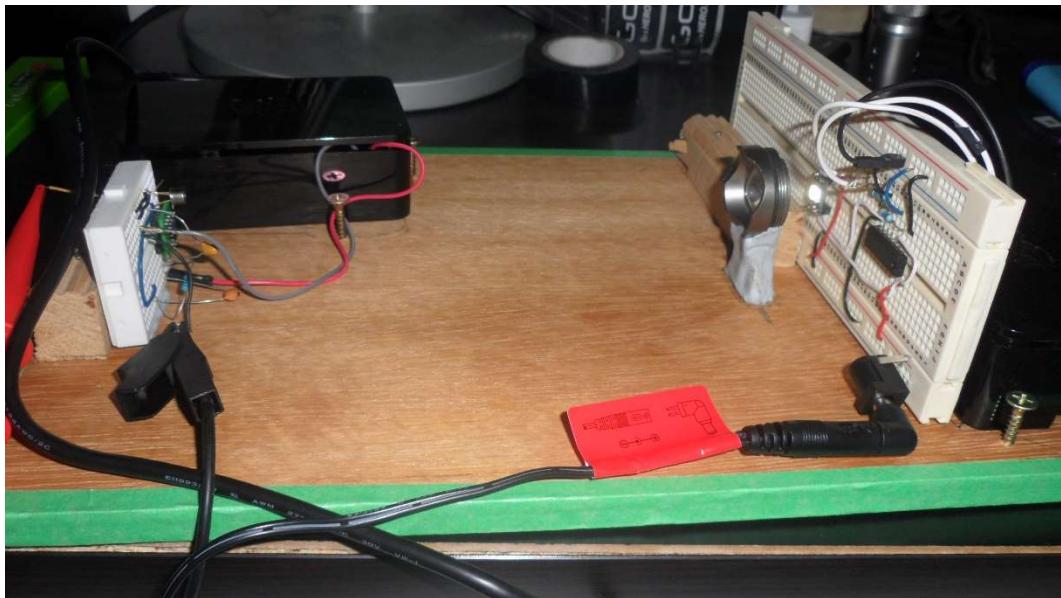


Figure 36 - Early Prototype Test Environment.

For all of the system level testing, the output signal at the receiver for the GPIO was monitored with the ScopeMeter.

### 5.5.1 Distance

Once the transmission module was set up with the receiver, the fears around the lack of light directivity and intensity were confirmed. The modulated light was barely registering on the photodiode and at the amplified output, although it could still be seen as a square wave. This was from an initial distance of 500mm which was considered reasonable given the target of 1-2m. The LED brightness was also set to the maximum level and it was determined that focusing the light and closing the distance was the only option to ensure meaningful communication. This decision was taken with a view to improve performance after the initial objectives were achieved. The distance for the preliminary system testing was set at 200mm and the lens was focussed accordingly.

### **5.5.2 Transmitting Square Waves**

Before attempting the processing of file data, the system was tested with square waves from the transmission module. These were set in code and ranged from 1MHz down to 1 kHz. All square waves were shown at the receiver output and had good definition of the signal. At the higher frequencies it was noted that some of the 0's did not return to the same level as adjacent 0's, which would have been expected. The root cause of this occurrence was not able to be determined. The issue was also seen when the file data was transmitted.

### **5.5.3 Transmitting File Data**

With the successful processing of square waves, the script was finished off to process the binary data of a file. This file was a picture of the Raspberry Pi logo in a jpeg format. The file was uncomplicated, but it didn't matter as the only meaningful aspect was that it contained binary data to extract and send.

When the file data was first transmitted, the concept of data encoding was not at the forefront of the project. However, once the output was seen, it was clear that this aspect was critical to the success of sending and processing any information. A screen capture of the data at the output of the receiver can be seen in Section 5.3.8. It shows the long sequences of similar bits and unless the original clock signal or timing is known, the bits cannot be separated into the original stream. Note that this is only a small snapshot of the data that gets transmitted and received.

Because of the boot-up failures of the Raspberry Pi modules, the final testing to determine transmission bit rate and actual frequency were unable to be performed. Timing of the transmission and confirmation of the bit-wise size of the file data array (i.e. how many elements) could have been used to determine this system performance characteristic.

Aside from measuring the system capability, it was able to successfully extract a file and transmit the data across free space using modulation of an LED.

# 6 CONTRIBUTIONS, CONCLUSION & FUTURE WORK

## 6.1 Summary of Contributions

In order to quantify the level of success to the body of knowledge for this topic, a summary of project contributions are included below.

### 6.1.1 Addressable LED's

#### 6.1.1.1 *1MHz Speed Limit*

Due to the nature of communication with addressable LED's, they are not suitable as a transmission element, unless for low speed applications. These speeds would have to be above those perceptible by the human eye and also those that are likely to cause medical events such as seizure triggers, headaches and nausea. Transmission speeds with a clock signal of 32MHz are therefore limited to 1MHz before any bandwidth or circuit limiting conditions.

#### 6.1.1.2 *Adjacent Element Interference*

In an addressable LED strip, the proximity of the illuminating elements to each other, may result in difficulties locking onto a particular element or interference caused by adjacent elements. This could be overcome with a rigorous hand-off regime or assigning transmission elements spaced a 1m or greater apart.

#### 6.1.1.3 *Lack of Effective Brightness*

With a singular element luminous flux of 18 lumens, the luminous intensity does not offer enough 'power' to transmit data over distance greater than 400mm. For a domestic application this does not allow an associated device to be far enough away from a ceiling mounted lighting installation. An opportunity does exist for smaller installations such as desk lamps.

#### 6.1.1.4 *DotStar LED's can be modulated*

The project has shown that DotStar (APA102c) LED's can be integrated with Raspberry Pi and modulated via Python to carry file data.

#### 6.1.1.5 *DotStar Needs a Separate Power Supply*

The Raspberry Pi +5V GPIO and current sharing scheme mean that the addressable LED requires a separate power supply to operate. The LED was unable to be powered with +3.3V from the GPIO. Requiring a separate power supply removes some of the compact and 'independent' nature that a similar project might seek. This would be evident in an Internet of Things situation requiring small device.

#### 6.1.1.6 *Conclusion*

Addressable LED's at present are not suitable for inclusion in a Light Fidelity system.

## 6.1.2 Raspberry Pi

### 6.1.2.1 *Raspberry Pi as a Suitable Platform*

The key feature that makes Raspberry Pi suitable for Li-Fi applications is its ability to handle everything in the system. It has Wi-Fi, Bluetooth, USB's, a PC based environment, a suitable coding platform, significant processing speed and memory. It does come with a familiarity due to its PC environment

and the plethora of support and information available on the internet. Another key feature is that it is \$50 per module.

#### *6.1.2.2 Raspberry Pi Deficiencies*

Where it falls down is the voltage supply constraints, the GPIO only accepts +3.3V, GPIO current constraints, CPU process sharing and vulnerability with booting up from microSD cards.

#### *6.1.2.3 Use MobaXterm to Save on Peripherals*

To work with Raspberry Pi, you effectively need peripherals to interface with it. Using Shell Scripting with a freeware program called MobaXterm, the need to connect new peripherals is removed. You can work with the Raspberry Pi using your own PC peripherals. It's simple to set up and has the capability for much wider technical uses.

#### *6.1.2.4 More Powerful Devices Coming*

Raspberry Pi continually release more and more powerful modules. They also have smaller devices in the Raspberry Pi Zero.

#### *6.1.2.5 Conclusion*

The Raspberry Pi platform is suitable to use in a Li-Fi system.

### **6.1.3 Prototype Components**

#### *6.1.3.1 Photodiode*

The Hamamatsu S5973 photodiode performed very well in the receiver module. The intended signals were received and processed as expected. Very low terminal capacitance. A little expensive.

#### *6.1.3.2 Transimpedance Amplifier*

The Texas Instruments OPA380 TIA was simple to set up and provided the required signal gain. It does come in a SOIC package, so needs a breakout board or special soldering.

#### *6.1.3.3 Level Shifting Chip*

The Texas Instruments 74HCT125 quad level shifting chip was very simple to set up and essential to use with addressable LED's and +3.3V GPIO microprocessors.

#### *6.1.3.4 Conclusion*

All components used in the system are recommended for future applications.

### **6.1.4 Construction**

#### *6.1.4.1 Inexpensive*

This project shows that a Li-Fi prototype can be made for <\$250. Cost could be further reduced if a cheaper photodiode is sourced and a single LED (non-addressable) is used as the transmission element. This eliminates the need for an additional power supply, beam focusing device and level shifting chip.

#### *6.1.4.2 Compact*

This project shows that a small and compact Li-Fi prototype can be made. With a suitable case and PCB to mount the circuit components, that PCB can be embedded into the Raspberry Pi case. This would provide protection and a more robust module.

#### *6.1.4.3 Accessibility*

All components were purchased from domestic electronic stores, such as Jaycar, Element14 and RS Components. This ensures that the prototype can be reproduced by any student following the instruction and research in this report.

#### *6.1.4.4 Conclusion*

A compact and inexpensive Li-Fi prototype can be made by students.

### **6.1.5 Signal Modulation**

Signal modulation and encoding is critical to successful transmission of data. Without an encoding scheme, the data stream cannot be determined from long sequences.

#### *6.1.5.1 Differential Manchester*

Although not implemented in this project, preliminary research showed that Differential Manchester has a place in the encoding and decoding of the data. The software diagrams included show where to implement it and the role it plays in the transmission and receiver modules.

#### *6.1.5.2 Conclusion*

Data encoding and decoding is essential to the success of the transfer of data.

### **6.1.6 Python and SPI**

#### *6.1.6.1 Python*

Python proved to be a suitable language for the coding and file processing. It was simple to learn and came with the Raspberry Pi. Python does fall short in many benchmark tests against other coding languages, but for a prototype implementation it is suitable and was shown to be able to modulate an addressable LED to carry binary data.

#### *6.1.6.2 SPI*

SPI was an effective method to send the 32 bit data signal to the GPIO and the LED circuit. It cut down on coding lines and allowed the speed to be determined by the user. SPI was also effective at modulating an addressable LED to carry binary data.

#### *6.1.6.3 Conclusion*

Python and SPI are effective methods to send and process information in a Li-Fi prototype.

## **6.2 Review of Project Objectives**

### **6.2.1 Objectives 1, 2 and 3 – Research**

These objectives were to conduct a literature review, conduct a requirements analysis and identify suitable components for prototype implementation.

The literature review was comprehensive and covered all aspects of Visible Light Communication, alternative technology as well as the State of the Art. Light Fidelity is a fast moving research field with it gaining in popularity through the media and topical journalism. Nevertheless, a knowledge gap was identified and targeted with this prototype.

Components were reviewed in light of the requirements analysis and selected to fulfil the various roles within the prototype. Aside from the addressable LED (which formed some of the knowledge gap) there were no other components selected that were deemed inappropriate.

These 3 objectives can be considered successfully completed.

### **6.2.2 Objective 4 – Build a Prototype**

This objective was to build a basic prototype with physical components and to achieve high speed square wave transmission. This was achieved with the transmission of a 1MHz square wave between the two modules. The square wave was at the correct level and was clamped to +3.3V to prevent damage to the Raspberry Pi GPIO.

This objective can be considered successfully completed.

### **6.2.3 Objective 5 – Transmit Video**

This objective was to evaluate the performance and investigate improved modulation techniques for video transmission. Another part of the objective was to transmit basic video.

The original project intent was to transmit and receive video files, but the complexity of this task as well as building a light fidelity prototype from scratch meant that this goal was unobtainable in the timeframe available.

The complexity of transmitting video signals at a bit level was found to be at a significantly difficult level for an undergraduate project. Just transmitting a video file is complex enough, but to stream the video and muxed audio signals is very advanced work. In hindsight (and only discovered through research) this should have been put into the ‘as time permits’ section rather than a project objective.

As such, the goal was amended to transmitting a file across free space and that was partially achieved. There was also difficulty encountered with implementation of an encoding scheme, with the realisation coming late in the project development. With other academic commitments, the ability to deliver a coherent report took precedence over implementing the chosen encoding scheme.

This objective can be considered partially successful, but not completed.

## 6.2.4 Objective 6 – Quantity the Prototype

This objective was to quantify the performance parameters of the prototype, in particular the distance of transmission and the quality of file reproduction.

With the challenges presented by the addressable LED, the distance was only able to be quantified at 200mm with a focusing lens and up to 400mm with a collimator to focus the light beam onto the photodiode. Note that the distance parameter was improved by altering an aspect of the prototype.

Regarding quality of the reproduction, the file was not able to be reproduced in the receiver module due to difficulties encountered with the encoding and modulation scheme. The code tested was able to accurately reproduce the original file, but this was within the transmission module rather than across free space.

So although the performance of the prototype in regards to distance and quality (at this point in time) is poor, this object can be considered partially successful.

## 6.2.5 Objective 7 – Future Work

There was a significant process of discovery and learning throughout the project and this has resulted in a clear understanding of where this research needs to go in the future.

### 6.2.5.1 *Internet of Things*

Light Fidelity has a place in the domestic environment to contribute to the Internet of Things. The release of small microprocessors mean that similar prototype fundamentals shown here can be implemented on a much smaller scale. The components used in this project are compact and can easily be adapted to fit into a small package such as the Raspberry Pi Zero. An LED transmitter would need to be in line of sight of the receiver and communication can take place between household devices.

### 6.2.5.2 *Improvement of LED Transmitter Element*

It is clear that addressable LED's are not presently suitable for this application, so research could continue to find a small, bright LED that could be used as the transmission element for a prototype. This would cut down on coding and support components. Consideration should be given to ceiling mounted LED's or even modulation of domestic ceiling lights.

### 6.2.5.3 *Implementation of Data Encoding Techniques*

This project was not able to successfully implement a data encoding technique to allow data reproduction in the receiver module. The project has suggested that Differential Manchester is suitable and also shown software diagrams of where to integrate it. Other encoding schemes could be trialled.

### 6.2.5.4 *Possible Introduction of Arduino*

With the success (with some limitations) there is an opportunity to tailor a light Fidelity prototype using Arduino products. These remove the challenges faced with a +3.3V device interfacing with +5V components and remove the current and CPU sharing schemes of the Raspberry Pi.

### 6.2.5.5 *Alternative Coding Language*

Python is regularly out-performed in coding benchmarks by other languages such as C and C++. Python could present a limitation in a real-time environment particularly if streaming data live. An option would be to explore an alternative coding language to handle the file processing and LED modulation.

#### *6.2.5.6 Observe Flicker Rates and Human Perception*

A prototype could be constructed for the purpose of measuring human perception to LED flicker, while still transmitting information.

#### *6.2.5.7 Vehicle to Vehicle Communication*

Being able to modulate an LED means that this can be applied to any LED. Many vehicles now come with LED lights in the front and rear and if these could be modulated to carry information, then vehicles could communicate. This application can lend itself to sending voice activated messages to drivers in front or behind, or warn a trailing vehicle of the speed it is doing, or that a vehicle is too close the one in front. A potential invention could be the sending of SMS-like messages to other drivers, converted from voice to text and sent via modulated head or taillights. Imagine being able to tell another driver that he has one headlight out or he's left his wallet on the roof or that there's a speed camera up the road.

#### *6.2.5.8 Photodiode in Photovoltaic vs Photoconductive Modes*

Test the photodiode in photoconductive mode and a control of photovoltaic mode to observe differences in speed, noise, signals etc.

#### *6.2.5.9 I2C, UART and SPI*

There are alternative methods to move data around a microcontroller. These could be explored for speed, quality and ease of implementation.

#### *6.2.5.10 Implementation of Data Packet Transfer*

This is already happening in the realm of Li-Fi, but to implement a low cost version would really boost the body of knowledge in this area. Refer to IEEE 802.15.7 for the standard.

## 7 REFERENCES

- AdaFruit Industries, 2016. About AdaFruit - Press, Limor Fried / Ladyada & more.... [ONLINE] Available at: <https://www.adafruit.com/about>. [Accessed 05 August 2016]
- AdaFruit Industries. 2016. APA102 5050 Cool White LED w/ Integrated Driver Chip - 10 Pack [~6000K] ID: 2351 - \$4.95: AdaFruit Industries, Unique & fun DIY electronics and kits. [ONLINE] Available at: <https://www.adafruit.com/products/2351>. [Accessed 29 August 2016].
- Ambady, S, Bredes, M & Nquyen, C 2015, 'Visible Light Communication', BSc Thesis, Worcester Polytechnic Institute, Massachusetts, USA.
- Australian Communications & Media Authority, 2013, Australian Radio Frequency Spectrum Chart 2013, ACMA, Sydney, Available at: [www.acma.gov.au](http://www.acma.gov.au). [Accessed 09 October 2015].
- Burgess, P 2016, AdaFruit NeoPixel Uberguide. AdaFruit Learning Systems, New York. Available at: <https://learn.adafruit.com/downloads/pdf/adafruit-neopixel-uberguide.pdf> [Accessed 12 December 2015].
- Caihong, W, Xiaohua, L & Xiuhua, J, 2013. 'Design and Implementation of a Real-time Video Stream Analysis System Based on FFMPEG'. Fourth World Congress on Software Engineering. IEEE.
- Chang, Y, Liao, C, Ho, C, Wu, M, Hsieh, Y, Houng & Yang, C 2015, 'Light-emitting diodes for visible light communication', Wireless Communications and Mobile Computing Conference (IWCMC), 2015 International, pp.665-667.
- Cheque, C, Baeza, F, Marquez, G & Calderon, J 2015, 'Towards a Responsive Web Service for Smart Home LED Control with Raspberry Pi', 34<sup>th</sup> International Conference of the Chilean Computer Science Society (SCCC), 2015.
- Chung, Y, Sewaiwar, A, Vikramaditya, V & Tiwari, S 2015, 'Novel user allocation scheme for full duplex multiuser bidirectional Li-Fi network', Optics Communications, vol. 339, pp. 153-156.
- Clifford, P 2016. GPIO Electrical Specifications, Raspberry Pi Input and Output Pin Voltage and Current Capability. [ONLINE] Available at: <http://www.mosaic-industries.com/embedded-systems/microcontroller-projects/raspberry-pi/gpio-pin-electrical-specifications>. [Accessed 01 August 2016].
- Dennis G. et al 1988,'The ARPANET and the DARPA Internet', Library Hi Tech, Vol. 6 Iss 2 pp. 51 – 62.
- Dimitrov, S, Haas, H 2015. 'Principles of LED Light Communications: Towards Networked Li-Fi', Cambridge, UK. Cambridge University Press, 228 p.
- Flexfire LED, 2016. What is the difference between 3528 LEDs and 5050 LEDs. [ONLINE] Available at: <https://www.flexfireleds.com/pages/Comparison-between-3528-LEDs-and-5050-LEDs.html>. [Accessed 05 August 2016]
- GitHub Incorporated. 2014. doceme/ py-spidev. [ONLINE] Available at: <https://github.com/doceme/py-spidev>. [Accessed 3 April 2016].

Grand View Research 2015, ‘*Visible Light Communication (VLC) Market Analysis, Market Size, Application Analysis, Regional Outlook, Competitive Strategies, and Forecasts, 2015 To 2022*’, Grand View Research, San Francisco USA, Online, <<http://www.grandviewresearch.com/industry-analysis/visible-light-communication-market>>, [Accessed 10 October 2015].

Grobe, L, Paraskevopoulos, A, Hilt, J, Schulz, D, Lassak, F, Hartlieb, F, Kottke, C, Jungnickel, V & Langer, K 2013, ‘High-speed visible light communication systems’, in Communications Magazine, IEEE , vol.51, no.12, pp.60-66, December 2013.

Hamamatsu Photonics, K.K., 2012. High-speed photodiodes (S5973 series: 1 GHz). Hamamatsu City, Japan.

Hass, H 2011, Wireless Data from Every Light Bulb, TEDGlobal, online video, viewed 17 October 2015, <[https://www.ted.com/talks/harald\\_haas\\_wireless\\_data\\_from\\_every\\_light\\_bulb?language=en](https://www.ted.com/talks/harald_haas_wireless_data_from_every_light_bulb?language=en)>

Hetian, Z, Minglun, Z, Chao, W, Xujin, G & Yangan, Z 2015, Design of a visible light Internet access system in Ubiquitous and Future Networks (ICUFN), 2015 Seventh International Conference on: proceedings of the Ubiquitous and Future Networks (ICUFN), 2015 Seventh International Conference on pp. 49-52.

Hunter, K, Conrad, J & Willis, A 2014, ‘Visible light communication using a digital camera and an LED flashlight’, Conference Proceedings - IEEE SOUTHEASTCON.

iPixel LED, 2014. APA102C Datasheet [ONLINE], Shiji Lighting.

Jia, Y, Zhang, M, Huang, Y & Zhang Y, 2014. ‘Design of an FPGA Based Visible Light Communication System’. 12<sup>th</sup> International Conference on Optical Internet Proceedings, 2014. IEEE.

Kalavally, V, Zafar, F, Kalavally, Karunatilaka, D V & Parthiban, R 2015, ‘LED Based Indoor Visible Light Communications: State of the Art’, Communications Surveys & Tutorials, IEEE, vol.17, no.3, pp.1649-1678.

Kamsula, P 2015, Design and Implementation of a Bi-directional Visible Light Communication Testbed. University of Oulu, Department of Electrical and Information Engineering. Master’s Thesis, 59 p.

Kitchin, C 2007, ‘Avoid Common Problems when Designing Amplifier Circuits’, Analog Devices. Analog Dialogue, vol 41-08. Available at: [http://www.analog.com/library/analogdialogue/archives/41-08/amplifier\\_circuits.pdf](http://www.analog.com/library/analogdialogue/archives/41-08/amplifier_circuits.pdf) [Accessed 12 January 2016].

Klaver, L & Zuniga, M 2015, ‘Shine: A Step Towards Distributed Multi-Hop Visible Light Communication’, IEEE 12<sup>th</sup> International Conference on Mobile Ad Hoc and Sensor Systems’, Delft University of Technology, Netherlands, 2015.

Kout Petr, 2014. Petr Kout - Engineering & Physics. [ONLINE] Available at: <http://petrkout.com/electronics/low-latency-0-4-s-video-streaming-from-raspberry-pi-mjpeg-streamer-opencv/>. [Accessed 8 March 2016]

Langer, K, Vucic, J, Kottke, C, Fernandez, L, Habe, K, Paraskevopoulos, A, Wendl, M & Markov, V 2011, ‘Exploring the potentials of optical-wireless communication using white LEDs’, in Transparent Optical Networks (ICTON), 2011 13th International Conference, pp.1-5.

Marpe, D, Sullivan, G & Weigand T, 2006. 'Standards Report: The H.264/MPEG Advanced Video Coding Standard and its Applications'. IEEE Communications Magazine (pp. 134-143). August 2006. IEEE.

Mobatek, 2015. Raspberry Pi without monitor or keyboard. [ONLINE] Available at: <http://blog.mobatek.net/post/control-raspberry-pi-without-monitor-or-keyboard-thanks-to-MobaXterm/>. [Accessed 12 July 2016]

Nikhade, S 2015, 'Wireless sensor network system using Raspberry Pi and Zigbee for environmental monitoring applications', in Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), 2015 International Conference, pp.376-381.

Orozco, L 2014. 'Technical Article: Optimizing Precision Photodiode Sensor Circuit Design'. Analog Devices. Norwood, MA, USA. Available at: <http://www.analog.com/media/en/technical-documentation/technical-articles/Optimizing-Precision-Photodiode-Sensor-Circuit-Design-MS-2624.pdf> [Accessed 01 December 2015].

Pathak, P, Feng, X, Hu, P & Mohapatra, P 2015, 'Visible Light Communication, Networking and Sensing: A Survey, Potential and Challenges', in Communications Surveys & Tutorials, IEEE , vol.PP, no.99, pp.1-1.

pureLiFi. 2014. Light Becomes Data. [ONLINE] Available at: <http://purelifi.com/>. [Accessed 14 June 2016]

Queensland Government 2015, Risk Management. [ONLINE] Available at: <http://education.qld.gov.au/health/safety/managing/risk.html>. [Accessed 08 October 2015].

Rajagopal, S, Roberts & RD Sang-Kyu, L 2012, 'IEEE 802.15.7 visible light communication: modulation schemes and dimming support', Communications Magazine, IEEE, vol. 50, no. 3, pp. 72-82.

Ramus, X 2009. 'Application Report: Transimpedance Considerations for High-Speed Amplifiers'. Texas Instruments, Dallas Texas. Available at: <http://www.ti.com.cn/lit/an/sboa122/sboa122.pdf> [Accessed 10 January 2016].

Rani, J, Chauhan, P & Tripathi, R 2012, 'Li-Fi (Light Fidelity)-the future technology in wireless communication', International Journal of Applied Engineering Research, vol. 7, no. 11 SUPPL, pp. 1517-1520.

Raspberry Pi 2 Model B, 2015, Raspberry Pi 2 Model B. [ONLINE] Available at: <https://www.raspberrypi.org/products/raspberry-pi-2-model-b/>. [Accessed 05 October 2015].

Raspberry Pi Foundation. 2015. Raspberry Pi. [ONLINE] Available at: <https://www.raspberrypi.org>. [Accessed 17 April 2016]

RPi Tutorial EGHS: GPIO Protection Circuits. eLinux.org. Online. Available at: [http://elinux.org/RPi\\_Tutorial\\_EGHS:GPIO\\_Protection\\_Circuits#Zener\\_Diode\\_Protection](http://elinux.org/RPi_Tutorial_EGHS:GPIO_Protection_Circuits#Zener_Diode_Protection). [Accessed 24 March 2016].

SearchEnterpriseLinux. 2016. What is Debian? - Definition from WhatIs.com. [ONLINE] Available at: <http://searchenterpriselinux.techtarget.com/definition/Debian>. [Accessed 01 August 2016].

Stackpole, C 2010, A user's manual to the PMBOK guide, Wiley, Hoboken NJ.

Standards Australia 2009, AS/NZS ISO 31000:2009 Risk Management, Standards Australia, Sydney.

Stefan I & Haas H 2014, 'Hybrid visible light and radio frequency communication systems', IEEE Vehicular Technology Conference 2014.

Texas Instruments, 2007. OPA380 Precision, High-Speed Transimpedance Amplifier, Texas Instruments, PO Box 655303, Dallas Texas.

Uhan, H & Akbas, A 2012, 'Designing a system allowing high-definition video transfer with minimum latency and multi-use access to projection device by wireless', Complex Systems (ICCS), 2012 International Conference, pp.1-6, 5-6.

Wang, F & Xu, C, 2015. 'LED Visible Light Communication System Based on FPGA'. IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC) 2015 (pp. 428-432). IEEE.

Zhu, H, Zhang, M, Wang, C, Guo, X and Zhang, Y, 2015. 'Design of a visible light Internet access system'. In Ubiquitous and Future Networks (ICUFN), 2015 Seventh International Conference on (pp. 49-52). IEEE.

End.

## APPENDIX A – PROJECT SPECIFICATION

**Student:** Paul Fergusson

**Title:** Light Fidelity (Li-Fi) Prototype with Raspberry Pi

**Major:** Electronic and Electrical Engineering

**Supervisor:** Alexander Kist

**Enrolment:** ERP\_2016 - Semester 1 & 2 2016

**Project Aim:** The aim of this Project is to produce a functioning, ‘proof of concept’ prototype that utilises Visible Light Communication (VLC) technology to stream video across free space using Raspberry Pi.

**Programme:** Rev B, dated March 2016.

8. Undertake a literature review on visible light communication, modulation and signal conditioning.
9. Undertake a basic requirements analysis
10. Critically evaluate alternative implementations and identify suitable components for the prototype implementation.
11. Build a basic prototype with physical components and achieve high speed square wave transmission.
12. Evaluate the performance and investigate improved modulation techniques for video transmission. Transmit basic video and improve to HD (720p).
13. Measure distance and video quality to define optimal prototype parameters.
14. Identify future research direction.

*If time permits;*

15. Research improvements of the prototype to increase performance.
16. Try alternative modulation techniques to understand effects on prototype performance.

## APPENDIX B – PYTHON CODE

### B1. Transmitter Module Code

```
#!/usr/bin/python
from PIL import Image
import time
import spidev
import RPi.GPIO as GPIO

#set up SPI for LED
spi = spidev.SpiDev()
spi.open(0, 0)
spi.max_speed_hz = 31200000

#define start and finish sync-words

#implement differential Manchester coding with SCLK
#store new encoded data in new bytearray

#append start and finish sync-words to data bytearray

#opening the image file// for the TX module
buffer = bytearray()          #put the bits in an array to rebuild
picture = bytearray()          #this is where I store the final bits
file = open("logo.jpg", "rb")
bytes = bytearray(file.read())
for byte in bytes:
    for i in range(8):
        bit = (byte>>i) & 1
        buffer.append(bit)

#send bits to the LED using SPI
if bit == 1:
    resp = spi.xfer([0xFFFFFFFF])
else:
    resp = spi.xfer([0x00000000])
spi.close()
```

## B2. Receiver Module Code

```
#!/usr/bin/python
from PIL import Image
import time
import spidev
import RPi.GPIO as GPIO

#set up SPI
spi = spidev.SpiDev()
spi.open(0,1)
spi.max_speed_hz = 31200000

#define sync-words for start and finish

picture = bytearray()          #this is where I store the final bits
buffer = bytearray()           #put the bits in an array to rebuild

#Insert GPIO listening for start sync-word
#once start sync-word seen store next data to bytearray for decoding
#listen for finish sync-word and stop sending to bytearray

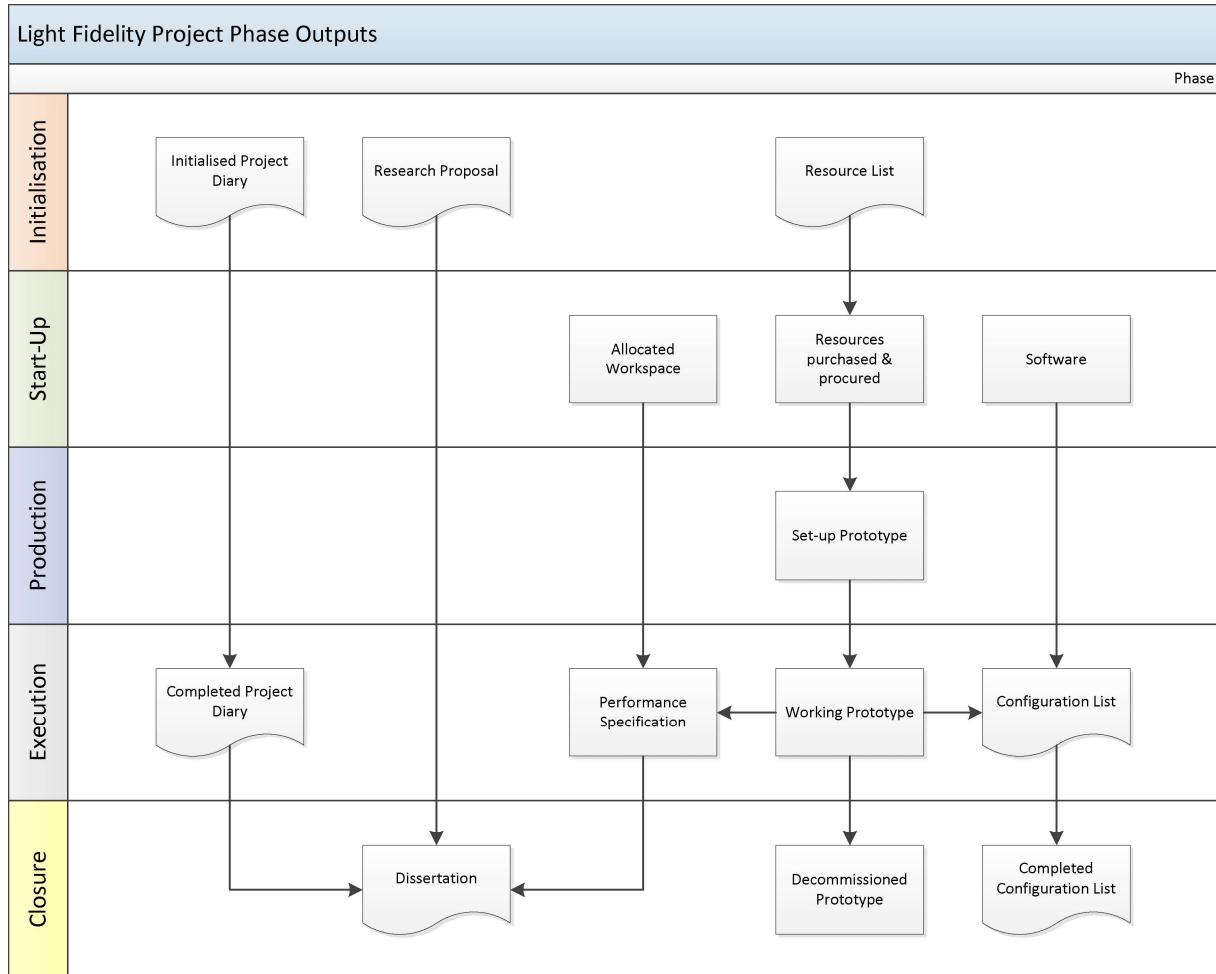
#decode bytearray looking for transitions in data.
#send 1's or 0's to new bytearray for conversion back to image

#convert back to right structure// for RX module
for i in range((len(buffer)/8)):
    bitstobytes = buffer[i*8:i*8+8]
    byte = 0
    for x in xrange(8):
        byte = byte + pow(2*bitstobytes[x],x)*bitstobytes[x]
    picture.append(byte)

#write to file and display result
refile = open("relogo.jpg", "wb")
refile.write(picture))
```

# APPENDIX C – PROJECT MANAGEMENT

## C1. Project Artefacts



## C2. Risk Management & Standard

To quantify and qualify the risks I have employed a simple 5x5 risk matrix to assess each risk against, based on the AS/NZS ISO 31000:2009 Risk Management Standard. This approach is also utilised by the Queensland Government.

Likelihood	Consequence				
	Insignificant	Minor	Moderate	Major	Critical
Almost Certain	Medium	Medium	High	Extreme	Extreme
Likely	Low	Medium	High	High	Extreme
Possible	Low	Medium	High	High	High
Unlikely	Low	Low	Medium	Medium	High
Rare	Low	Low	Low	Low	Medium

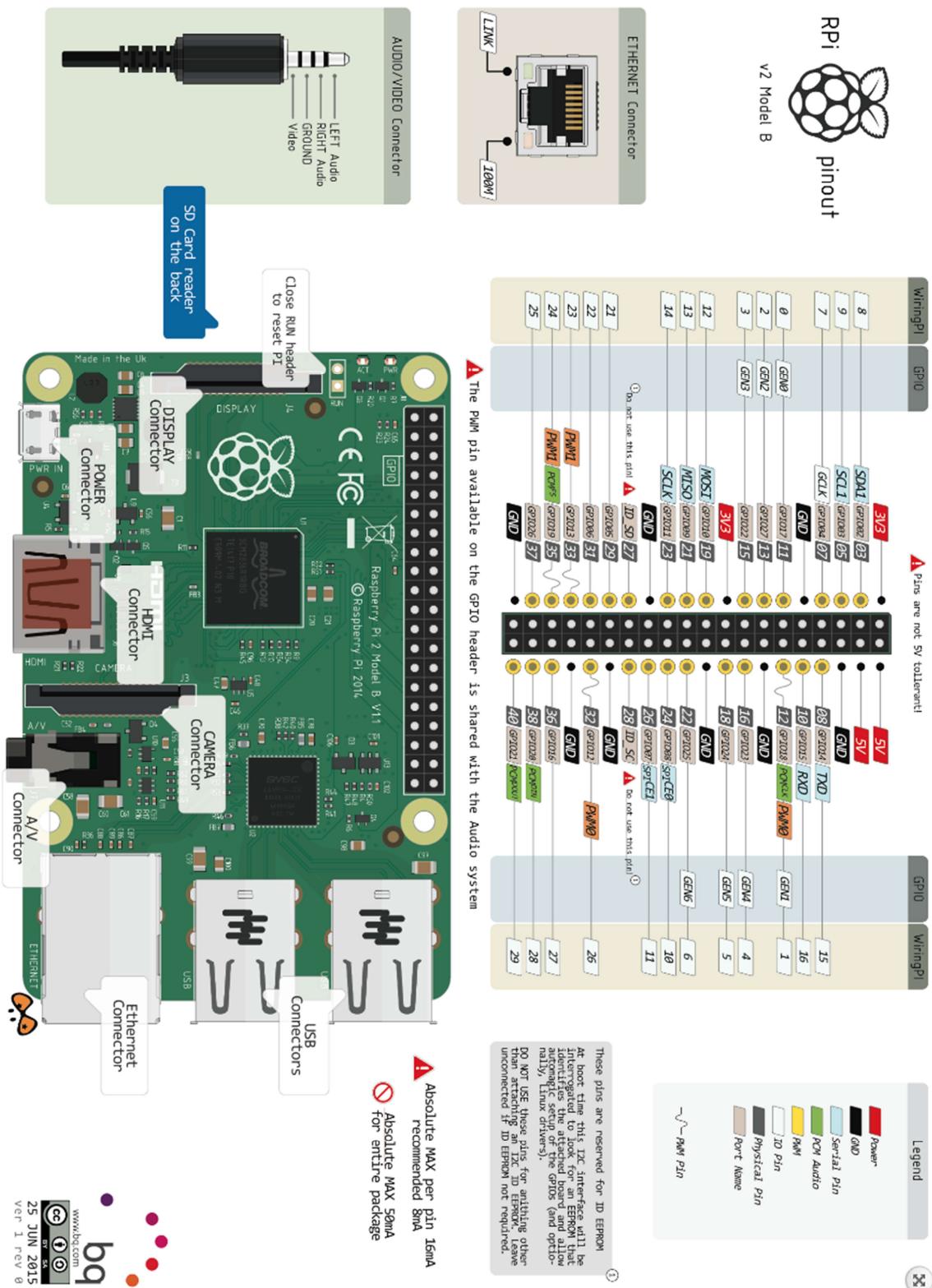
Consequence	Description of Consequence	Likelihood	Description of Likelihood
1. Insignificant	No treatment required	1. Rare	Will only occur in exceptional circumstances
2. Minor	Minor injury requiring First Aid treatment (e.g. minor cuts, bruises, bumps)	2. Unlikely	Not likely to occur within the foreseeable future, or within the project lifecycle
3. Moderate	Injury requiring medical treatment or lost time	3. Possible	May occur within the foreseeable future, or within the project lifecycle
4. Major	Serious injury (injuries) requiring specialist medical treatment or hospitalisation	4. Likely	Likely to occur within the foreseeable future, or within the project lifecycle
5. Critical	Loss of life, permanent disability or multiple serious injuries	5. Almost Certain	Almost certain to occur within the foreseeable future or within the project lifecycle

No.	Risk (there is a risk that.....)	Likelihood	Affect	Consequence	Initial Rating	Mitigation	Residual
1	the LED's will not be the right spec to modulate at the required frequency	Possible	Wasted time, unable to get working prototype, late project, stress.	Moderate	High	Review literature and determine a tested LED solution, determine the modulation specification early.	Medium
2	Workload to get a working prototype is under-estimated.	Unlikely	Program slip, stress, delayed academic deliverables, run into S2 subject workload, late project.	Moderate	Medium	Start early, stick to program, speak to supervisor early, if bogged down; move on, continuous working. Understand this is a research project and slight deviation from original method is satisfactory in order to meet the expected outcomes.	Low
3	Project is too complex for level of student.	Possible	Wasted time, unable to get working prototype, late project, stress.	Moderate	High	Deep literature review, discuss with supervisor, be honest with student ability. Understand this is a research project and slight deviation from original method is satisfactory in order to meet the expected outcomes.	Medium
4	Equipment failure delays progress.	Unlikely	Project cost increase, obsolescence, and delay to project.	Minor	Low	Spare LED's, use common items (Raspberry Pi), use spare power supplies, follow safety management items, and understand lead times of key items.	Low
5	Other academic or life priorities override Project.	Possible	Program slip, stress, delayed academic deliverables, subject deferral, late project.	Moderate	High	Have contingency in program, review key life events for 2016 and incorporate into program, start early, talk to supervisor, understand ability for extensions.	Medium

No.	Risk (there is a risk that.....)	Likelihood	Affect	Consequence	Initial Rating	Mitigation	Residual
6	Another student/academic releases similar project during 2016 (taking my knowledge gap).	Rare	Academic questions about authenticity of project, project loses impact and importance.	Insignificant	Low	Continuous review of literature throughout 2016, submission of Research Proposal and idea registration with USQ.	Low
7	Prototype idea is not possible to get to function (ie cannot transmit video data).	Possible	Wasted time, unable to get working prototype, late project, stress.	Moderate	High	Deep literature review, discuss with supervisor, consider back-up ideas or methodologies. Get transmission of high speed square wave first. Understand this is a research project and slight deviation from original method is satisfactory in order to meet the expected outcomes.	Medium
8	Student loses focus in Project.	Unlikely	Program slip, late deliverables, poor execution, poor quality project.	Moderate	Medium	Keep in contact with supervisor (weekly), post program at workstation, constant reminder that this is last year and to strive for excellent mark.	Low
9	Prototype is damaged due to outside event (fire, water etc.).	Rare	All project work is lost, all equipment is damaged, cannot deliver project, add another year to degree!!	Major	Low	Follow project safety management lists, keep project diary, take pictures at each step, review prototype workshop for hazards such as water ingress, flammable storage and remove.	Low
11	Electric shock to student while working on prototype.	Possible	Injury, health issues, medically treatment, delay to project.	Major	High	Follow project safety management lists, reduce the need to use higher voltages, turn off equipment before opening covers, test twice before touching, remove faulty equipment.	Medium

## APPENDIX D – PRODUCT DATASHEETS

### D1. Raspberry Pi 2B Layout Diagram



## **D2. Shiji LED APA102C (DotStar) Datasheet**



## RGB Full Color LED control IC

### ● ( General Description )

APA102C for the three-color RGB LED dimming control string Then IC , using the CMOS process , providing three-color RGB LED output driver to adjust the output with 256 gray-scale and 32 brightness adjustment APA102 with two-output WAY , the CLK signal by synchronization , so that the crystal cascadePiece of output movements synchronized.

### ● ( application)

LED lamps  
Large LED screen  
LED billboards

### ● ( Features )

CMOS process , low voltage, low power consumption

Synchronous of two-lane

Choose positive output or negative output RGB tri-color LED output, 8 Bit ( 256 level ) color Set, 5Bit ( 32 level ) brightness adjustment Built-20mA constant current output

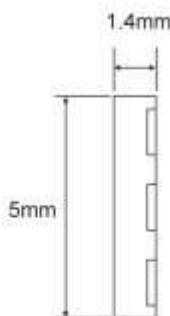
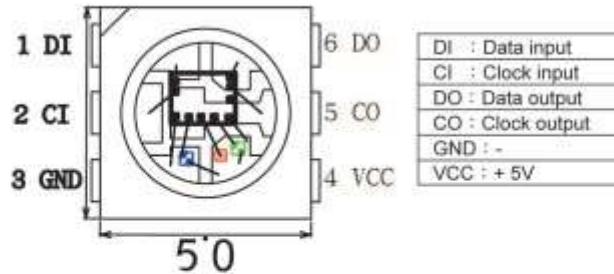
With self-detection signal

Built-in support for continuous oscillation PWM output can be maintained Static Screen

## PRODUCT SPECIFICATIONS

Model number	Color	Millicandela	refresh rate	Applied voltage	Power consumption	View angle	weight (g)	Dimensions(mm)	Operating temperature
SUPER LED	Full Color 16777216	R 500-650 mcd G 370-530 mcd B 120-165 mcd	400 cycle	5VDC	0.2W (MAX:1W)	H:160	0.1	5x5x1.4	-40°C-70°C

## PHYSICAL DIMENSIONS

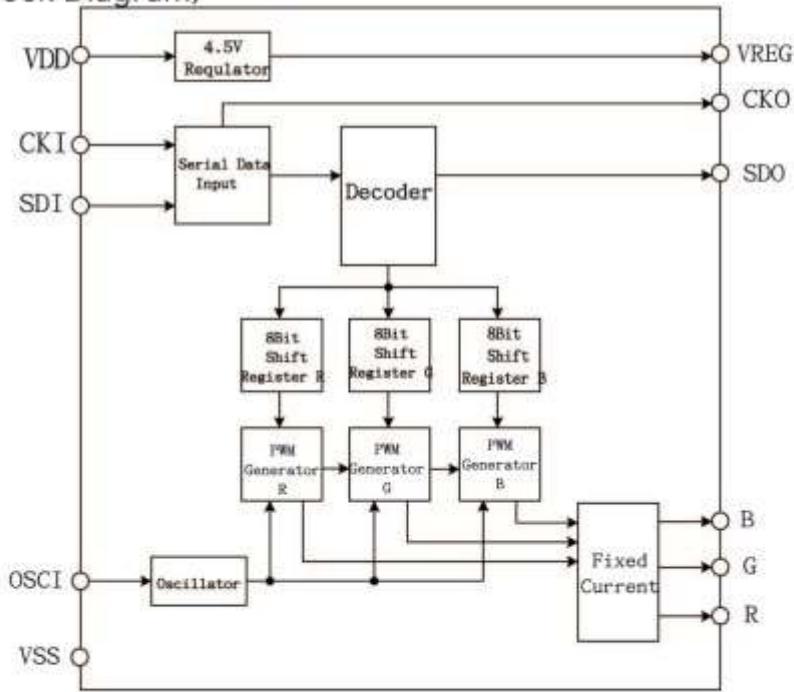




● 腳位說明 (Pin Description)

NO.	PIN NAME	I/O	FUNCTION
1	VDD	P	Power is terminal
2	VREG	O	4.5V regulator output
3	CKO	O	Series with the output clock signal
4	SDO	O	Series with the output data
5	VEN	I	Self-test function selection
6	CSEL	I	Invert the clock signal cascade
7	POLAR	I	Positive and negative output options
8	OSCI	I	Oscillator input
9	SDI	I	Series with the input data
10	CKI	I	Series with the input clock signal
11	REXT	I	Constant current source to adjust side
12	VSS	P	Power supply negative terminal
13	G	O	Green LED output
14	R	O	Red LED output
15	B	O	Blue LED output

● 功能方塊圖 (Block Diagram)





● 最大額定範圍 (Absolute Maximum Ratings)

Supply Voltage———0.3V to 6.0V

Input Voltage———VSS-0.3 to VDD+0.3

Operating Temperature———40°C to 70 °C

Storage Temperature———50°C to 125 °C

Note: Stress above those listed may cause permanent damage to the devices

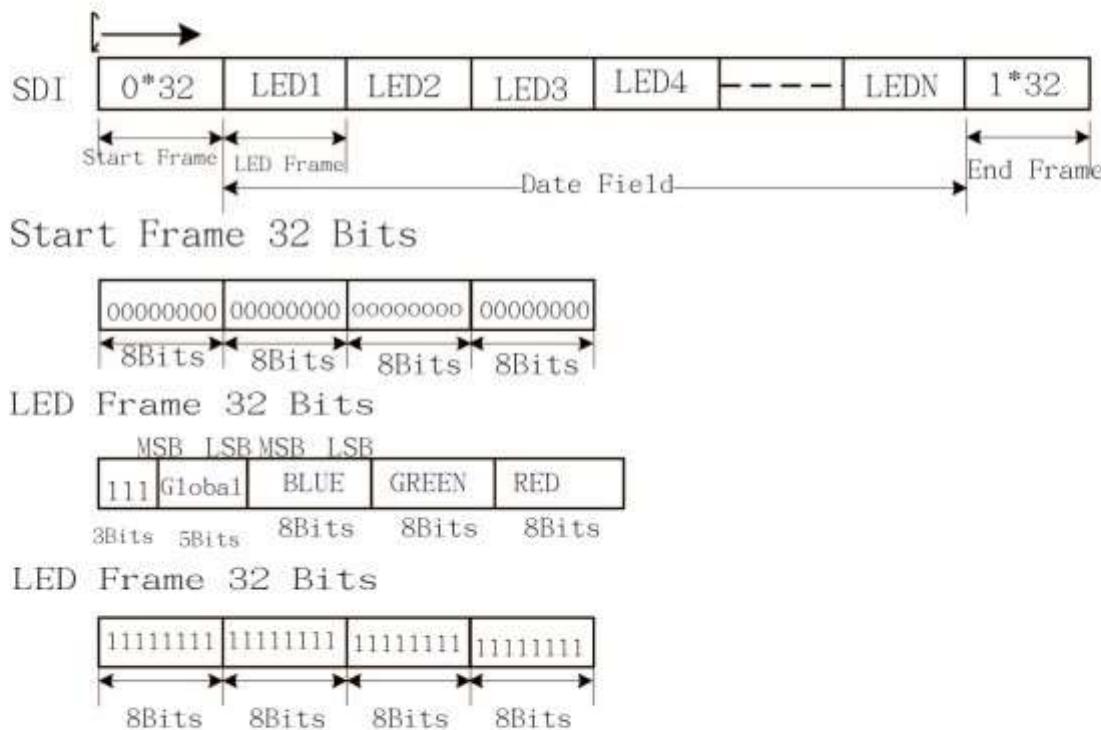
● 電氣特性 (Electrical Characteristics)

Symbol	Parameter	Condition	Min.	Typ.	Max	Units
VDD	Supply Voltage			5.0	5.5	V
VIH	Input High Voltage		0.7VDD		VDD+0.3	V
VH	Input Low Voltage		Vss-0.3		0.3VDD	V
LOL	Sink Current Voltage ( RGB )	@VDD=5V, VOL>1V	22.5	24.5	26.5	mA
RIN	Pull High	@VDD=5V		570		KΩ
VREG	Regulator Voltage ( VREG )	@VDD>5V	4.4	4.5	4.7	V
FOSC	Oscillator Frequency		800		1200	KHz

● 功能說明 (function description)

(1) .cascading data structure

Tabdem N-LED

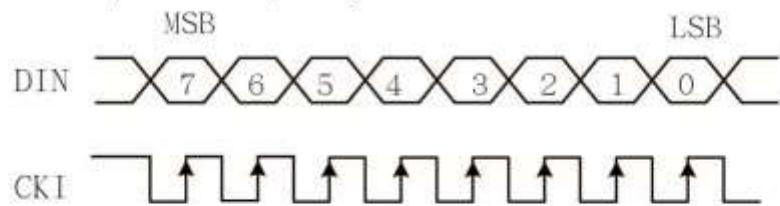




Global bit : 5-bit ( 32 level ) brightness setting , while controlling R, G, B three-color constant current output value , if set the Global bit for the 10000 ( 16/31 ) is the output current is half again the original PWM settings.

DATA MSB↔LSB	Driving Current
00000	0/31
00001	1/31
00010	2/31
...	
11110	30/31
11111	31/31(max)

#### PWM input and output signals Relations



Data MSB—	Duty Cycle
00000000	0/256(min)
00000001	1/256
00000010	2/256
...	
11111101	253/256
11111110	254/256
11111111	255/256(max)

2).The number of pixels per second sent to CKI frequency ( FCKI ) minus the Start Frame bit divided by the number 40 the number of LED Frame bit 32 , if CKI frequency ( FCKI ) to 512KHz , the pixel number ( 512000–40 ) /32=15998 , if the 50 second update Views can be connected in series LED number 15998/50=319.To increase the number of cascaded IC CKI frequency to be increased.

(3).POLAR to empty , R, G, B for the negative output; POLAR access VSS , R, G, B is positive output.

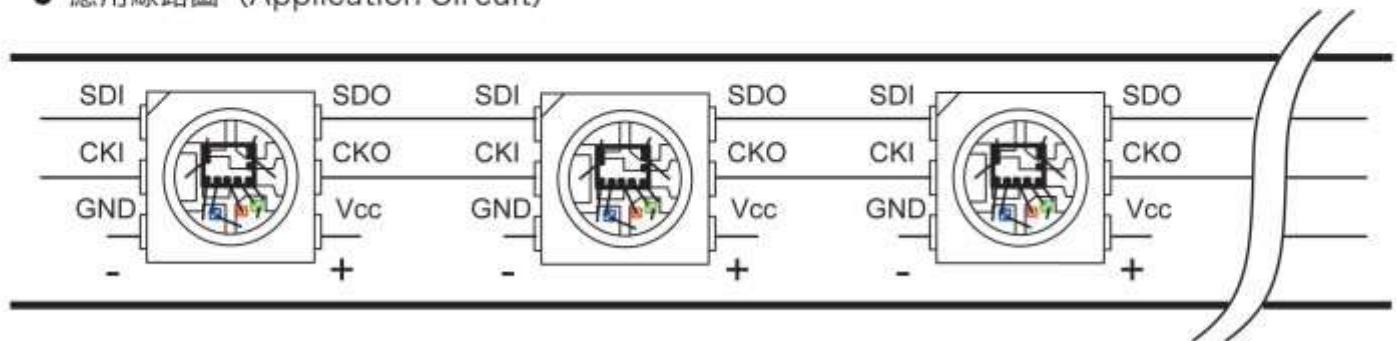
(4).VEN: Self-detection

Data Field to the middle of 3bit were B, G, R in the MSB of the opposite phase , otherwise regarded as invalid data. VEN close to empty when the self-detection: when VEN VSS then activated self-detection.

(5).CSEL to empty when the CKO and CKI RP :CSEL connected with VSS when the CKO compared with CKI.



● 應用線路圖 (Application Circuit)



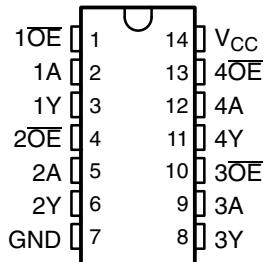
### **D3. TI 74HCT125 Quad Level Shifter Datasheet**

# SN54AHCT125, SN74AHCT125 QUADRUPLE BUS BUFFER GATES WITH 3-STATE OUTPUTS

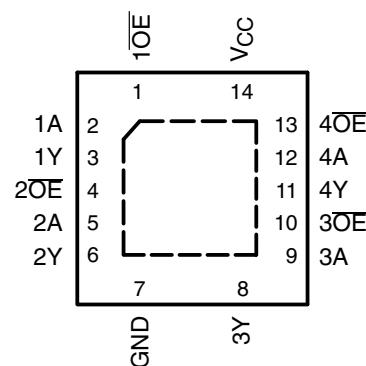
SCLS2640 – DECEMBER 1995 – REVISED JULY 2003

- Inputs Are TTL-Voltage Compatible
- Latch-Up Performance Exceeds 250 mA Per JESD 17
- ESD Protection Exceeds JESD 22
  - 2000-V Human-Body Model (A114-A)
  - 200-V Machine Model (A115-A)
  - 1000-V Charged-Device Model (C101)

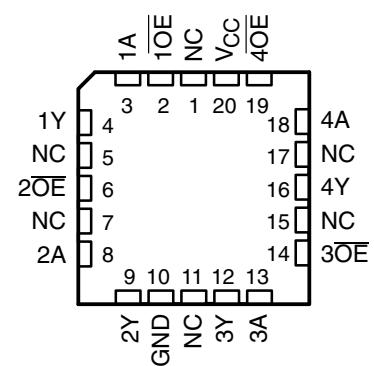
SN54AHCT125 . . . J OR W PACKAGE  
SN74AHCT125 . . . D, DB, DGV, N, NS,  
OR PW PACKAGE  
(TOP VIEW)



SN74AHCT125 . . . RGY PACKAGE  
(TOP VIEW)



SN54AHCT125 . . . FK PACKAGE  
(TOP VIEW)



NC – No internal connection

## description/ordering information

The 'AHCT125 devices are quadruple bus buffer gates featuring independent line drivers with 3-state outputs. Each output is disabled when the associated output-enable ( $\overline{OE}$ ) input is high. When  $\overline{OE}$  is low, the respective gate passes the data from the A input to its Y output.

To ensure the high-impedance state during power up or power down,  $\overline{OE}$  should be tied to V<sub>CC</sub> through a pullup resistor; the minimum value of the resistor is determined by the current-sinking capability of the driver.

## ORDERING INFORMATION

T <sub>A</sub>	PACKAGE <sup>†</sup>		ORDERABLE PART NUMBER	TOP-SIDE MARKING
–40°C to 85°C	QFN – RGY	Tape and reel	SN74AHCT125RGYR	HB125
	PDIP – N	Tube	SN74AHCT125N	SN74AHCT125
	SOIC – D	Tube	SN74AHCT125D	AHCT125
		Tape and reel	SN74AHCT125DR	
	SOP – NS	Tape and reel	SN74AHCT125NSR	AHCT125
	SSOP – DB	Tape and reel	SN74AHCT125DBR	HB125
	TSSOP – PW	Tube	SN74AHCT125PW	HB125
		Tape and reel	SN74AHCT125PWR	
–55°C to 125°C	TVSOP – DGV	Tape and reel	SN74AHCT125DGVR	HB125
	CDIP – J	Tube	SNJ54AHCT125J	SNJ54AHCT125
	CFP – W	Tube	SNJ54AHCT125W	SNJ54AHCT125W
	LCCC – FK	Tube	SNJ54AHCT125FK	SNJ54AHCT125FK

<sup>†</sup> Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at [www.ti.com/sc/package](http://www.ti.com/sc/package).



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.



POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

Copyright © 2003, Texas Instruments Incorporated  
On products compliant to MIL-PRF-38535, all parameters are tested unless otherwise noted. On all other products, production processing does not necessarily include testing of all parameters.

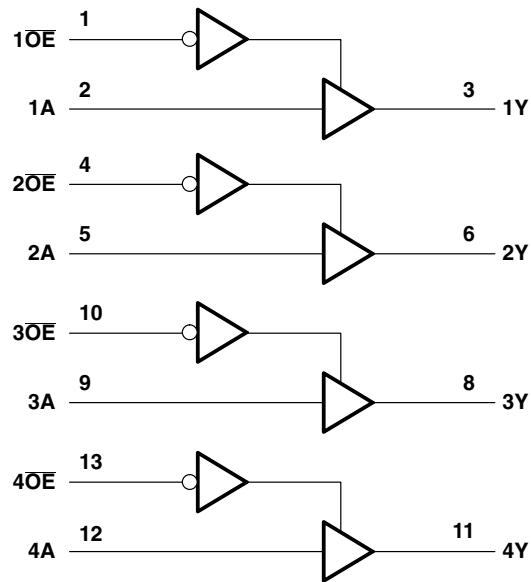
**SN54AHCT125, SN74AHCT125  
QUADRUPLE BUS BUFFER GATES  
WITH 3-STATE OUTPUTS**

SCLS264O – DECEMBER 1995 – REVISED JULY 2003

**FUNCTION TABLE  
(each buffer)**

INPUTS		OUTPUT Y
OE	A	
L	H	H
L	L	L
H	X	Z

**logic diagram (positive logic)**



Pin numbers shown are for the D, DB, DGV, J, N, NS, PW, RGY, and W packages.

# **SN54AHCT125, SN74AHCT125 QUADRUPLE BUS BUFFER GATES WITH 3-STATE OUTPUTS**

SCLS264O – DECEMBER 1995 – REVISED JULY 2003

**absolute maximum ratings over operating free-air temperature range (unless otherwise noted)**

<sup>†</sup> Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

**NOTES:**

1. The input and output voltage ratings may be exceeded if the input and output current ratings are observed.
2. The package thermal impedance is calculated in accordance with JESD 51-7.
3. The package thermal impedance is calculated in accordance with JESD 51-5.

**recommended operating conditions (see Note 4)**

		SN54AHCT125		SN74AHCT125		UNIT
		MIN	MAX	MIN	MAX	
V <sub>CC</sub>	Supply voltage	4.5	5.5	4.5	5.5	V
V <sub>IH</sub>	High-level input voltage	2		2		V
V <sub>IL</sub>	Low-level input voltage		0.8		0.8	V
V <sub>I</sub>	Input voltage	0	5.5	0	5.5	V
V <sub>O</sub>	Output voltage	0	V <sub>CC</sub>	0	V <sub>CC</sub>	V
I <sub>OH</sub>	High-level output current		-8		-8	mA
I <sub>OL</sub>	Low-level output current		8		8	mA
Δt/Δv	Input transition rise or fall rate		20		20	ns/V
T <sub>A</sub>	Operating free-air temperature	-55	125	-40	85	°C

**NOTE 4:** All unused inputs of the device must be held at V<sub>CC</sub> or GND to ensure proper device operation. Refer to the TI application report, *Implications of Slow or Floating CMOS Inputs*, literature number SCBA004.

**SN54AHCT125, SN74AHCT125  
QUADRUPLE BUS BUFFER GATES  
WITH 3-STATE OUTPUTS**

SCLS264O – DECEMBER 1995 – REVISED JULY 2003

**electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)**

PARAMETER	TEST CONDITIONS	$V_{CC}$	$T_A = 25^\circ C$			SN54AHCT125	SN74AHCT125	UNIT
			MIN	TYP	MAX	MIN	MAX	
$V_{OH}$	$I_{OH} = -50 \mu A$	4.5 V	4.4	4.5		4.4	4.4	V
	$I_{OH} = -8 mA$		3.94			3.8	3.8	
$V_{OL}$	$I_{OL} = 50 \mu A$	4.5 V		0.1		0.1	0.1	V
	$I_{OL} = 8 mA$			0.36		0.44	0.44	
$I_I$	$V_I = 5.5 V$ or GND	0 V to 5.5 V		$\pm 0.1$		$\pm 1^*$	$\pm 1$	$\mu A$
$I_{OZ}$	$V_O = V_{CC}$ or GND	5.5 V		$\pm 0.25$		$\pm 2.5$	$\pm 2.5$	$\mu A$
$I_{CC}$	$V_I = V_{CC}$ or GND, $I_O = 0$	5.5 V		2		20	20	$\mu A$
$\Delta I_{CC}^\dagger$	One input at 3.4 V, Other inputs at $V_{CC}$ or GND	5.5 V		1.35		1.5	1.5	mA
$C_i$	$V_I = V_{CC}$ or GND	5 V		4	10		10	pF
$C_o$	$V_O = V_{CC}$ or GND	5 V		15				pF

\* On products compliant to MIL-PRF-38535, this parameter is not production tested at  $V_{CC} = 0 V$ .

† This is the increase in supply current for each input at one of the specified TTL voltage levels, rather than 0 V or  $V_{CC}$ .

**switching characteristics over recommended operating free-air temperature range,  
 $V_{CC} = 5 V \pm 0.5 V$  (unless otherwise noted) (see Figure 1)**

PARAMETER	FROM (INPUT)	TO (OUTPUT)	LOAD CAPACITANCE	$T_A = 25^\circ C$			SN54AHCT125	SN74AHCT125	UNIT	
				MIN	TYP	MAX	MIN	MAX		
$t_{PLH}$	A	Y	$C_L = 15 pF$	3.8**	5.5**		1**	6.5**	1	6.5
				3.8**	5.5**		1**	6.5**	1	6.5
$t_{PZH}$	$\overline{OE}$	Y	$C_L = 15 pF$	3.6**	5.1**		1**	6**	1	6
				3.6**	5.1**		1**	6**	1	6
$t_{PHZ}$	$\overline{OE}$	Y	$C_L = 15 pF$	4.6**	6.8**		1**	8**	1	8
				4.6**	6.8**		1**	8**	1	8
$t_{PLH}$	A	Y	$C_L = 50 pF$	5.3	7.5		1	8.5	1	8.5
				5.3	7.5		1	8.5	1	8.5
$t_{PZH}$	$\overline{OE}$	Y	$C_L = 50 pF$	5.1	7.1		1	8	1	8
				5.1	7.1		1	8	1	8
$t_{PHZ}$	$\overline{OE}$	Y	$C_L = 50 pF$	6.1	8.8		1	10	1	10
				6.1	8.8		1	10	1	10
$t_{sk(o)}$			$C_L = 50 pF$		1***				1	ns

\*\* On products compliant to MIL-PRF-38535, this parameter is not production tested.

\*\*\* On products compliant to MIL-PRF-38535, this parameter does not apply.

**noise characteristics,  $V_{CC} = 5 V$ ,  $C_L = 50 pF$ ,  $T_A = 25^\circ C$  (see Note 5)**

PARAMETER	$SN74AHCT125$		UNIT
	MIN	MAX	
$V_{OL(P)}$ Quiet output, maximum dynamic $V_{OL}$		0.8	V
$V_{OL(V)}$ Quiet output, minimum dynamic $V_{OL}$		-0.8	V
$V_{OH(V)}$ Quiet output, minimum dynamic $V_{OH}$	4.4		V
$V_{IH(D)}$ High-level dynamic input voltage	2		V
$V_{IL(D)}$ Low-level dynamic input voltage		0.8	V

NOTE 5: Characteristics are for surface-mount packages only.

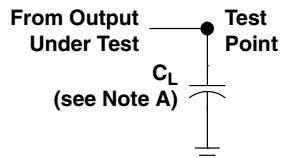


POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

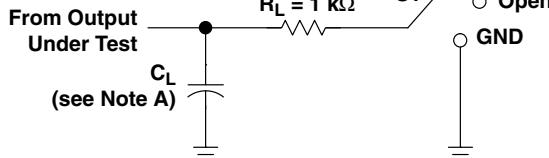
**operating characteristics,  $V_{CC} = 5 \text{ V}$ ,  $T_A = 25^\circ\text{C}$**

PARAMETER	TEST CONDITIONS	TYP	UNIT
$C_{pd}$ Power dissipation capacitance	No load, $f = 1 \text{ MHz}$	14	pF

**PARAMETER MEASUREMENT INFORMATION**

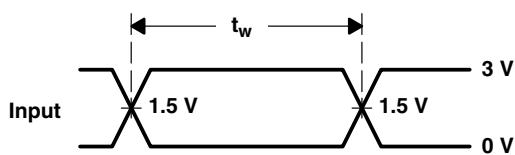


LOAD CIRCUIT FOR  
TOTEM-POLE OUTPUTS

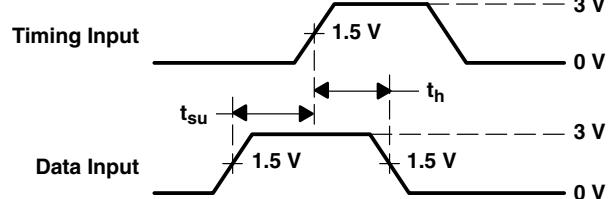


LOAD CIRCUIT FOR  
3-STATE AND OPEN-DRAIN OUTPUTS

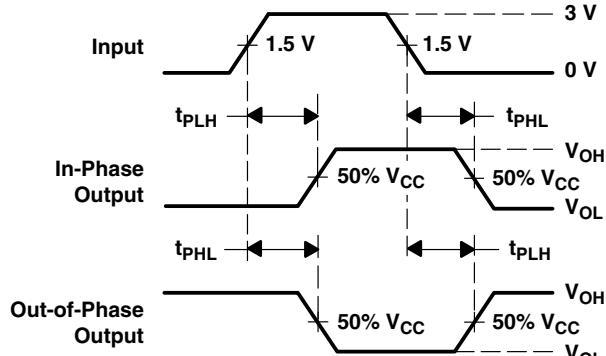
TEST	S1
$t_{PLH}/t_{PHL}$	Open
$t_{PLZ}/t_{PZL}$	$V_{CC}$
$t_{PHZ}/t_{PZH}$	GND
Open Drain	$V_{CC}$



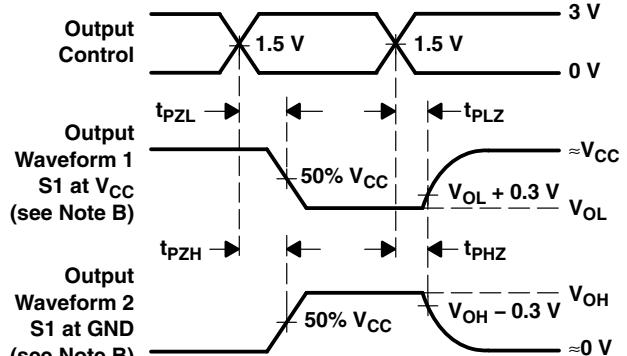
VOLTAGE WAVEFORMS  
PULSE DURATION



VOLTAGE WAVEFORMS  
SETUP AND HOLD TIMES



VOLTAGE WAVEFORMS  
PROPAGATION DELAY TIMES  
INVERTING AND NONINVERTING OUTPUTS



VOLTAGE WAVEFORMS  
ENABLE AND DISABLE TIMES  
LOW- AND HIGH-LEVEL ENABLING

- NOTES:
- A.  $C_L$  includes probe and jig capacitance.
  - B. Waveform 1 is for an output with internal conditions such that the output is low except when disabled by the output control. Waveform 2 is for an output with internal conditions such that the output is high except when disabled by the output control.
  - C. All input pulses are supplied by generators having the following characteristics: PRR  $\leq 1 \text{ MHz}$ ,  $Z_O = 50 \Omega$ ,  $t_r \leq 3 \text{ ns}$ ,  $t_f \leq 3 \text{ ns}$ .
  - D. The outputs are measured one at a time with one input transition per measurement.
  - E. All parameters and waveforms are not applicable to all devices.

Figure 1. Load Circuit and Voltage Waveforms

## **D4. Hamamatsu S5973 Photodiode Datasheet**

## PHOTODIODE

# Si PIN photodiode S5971, S5972, S5973 series

High-speed photodiodes (S5973 series: 1 GHz)



S5971, S5972 and S5973 series are high-speed Si PIN photodiodes designed for visible to near infrared light detection. These photodiodes provide wideband characteristics at a low bias, making them suitable for optical communications and other high-speed photometry. S5973 series includes a mini-lens type (S5973-01) that can be efficiently coupled to an optical fiber and a violet sensitivity enhanced type (S5973-02) ideal for violet laser detection.

## Features

- High-speed response  
S5971 : 100 MHz (VR=10 V)  
S5972 : 500 MHz (VR=10 V)  
S5973 series: 1 GHz (VR=3.3 V)
- Low price
- High sensitivity  
S5973-02: 0.3 A/W, QE=91 % ( $\lambda = 410$  nm)
- High reliability

## Applications

- Optical fiber communications
- High-speed photometry
- Violet laser detection (S5973-02)

## ■ General ratings / Absolute maximum ratings

Type No.	Dimensional outline/ Window material *1	Package (mm)	Active area size (mm)	Effective active area (mm <sup>2</sup> )	Absolute maximum ratings							
					Reverse voltage VR Max. (V)	Power dissipation P (mW)	Operating temperature Topr (°C)	Storage temperature Tstg (°C)				
S5971	①/K	TO-18	Ø 1.2	1.1	20	50	-40 to +100	-55 to +125				
S5972			Ø 0.8	0.5								
S5973			Ø 0.4	0.12								
S5973-01												
S5973-02												

## ■ Electrical and optical characteristics

Type No.	Spectral response range $\lambda$ (nm)	Peak sensitivity wavelength $\lambda_p$ (nm)	Photo sensitivity S (A/W)				Short circuit current Isc 100 lx ( $\mu$ A)	Dark current Id		Temp. coefficient of Id TCID (times/°C)	Cut-off frequency fc (GHz)	Terminal capacitance Ct $f=1$ MHz (pF)	NEP VR=10 V (W/Hz <sup>1/2</sup> )		
			λ p	660 nm	780 nm	830 nm		Typ. (nA)	Max. (nA)						
S5971	320 to 1060	900	0.64	0.55	0.6	1.0	0.07 *3	1 *3	1.15	0.1 *3	3 *3	$7.4 \times 10^{-15}$	$3.1 \times 10^{-15}$		
S5972		800	0.57		0.44	0.55	0.42	0.01 *3		0.5 *3					
S5973	320 to 1000	760	0.52	0.51	0.47	0.42	0.001 *4	0.1 *4		1 *4	1.6 *4	$1.5 \times 10^{-15}$			
S5973-01															
S5973-02			0.45	0.3 *2	0.42	0.37	0.09								

\*1: Window material K: borosilicate glass, L: lens type borosilicate glass

\*2:  $\lambda = 410$  nm

\*3: VR=10 V

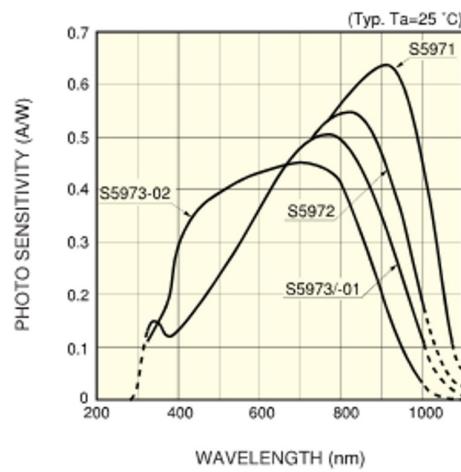
\*4: VR=3.3 V

**SOLID STATE DIVISION**

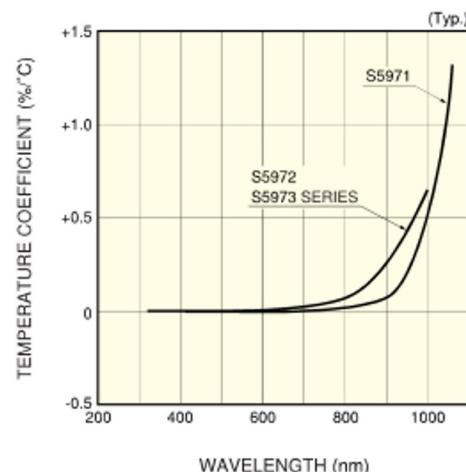
**HAMAMATSU**

## Si PIN photodiode S5971, S5972, S5973 series

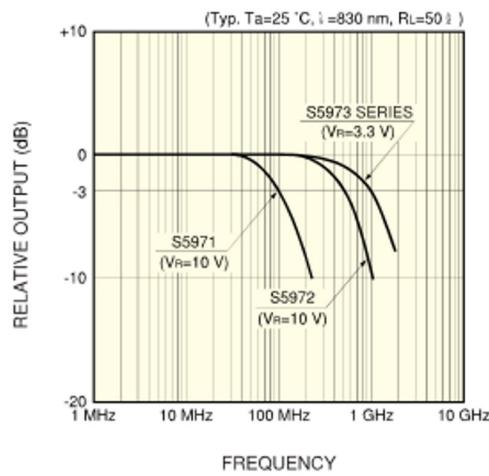
### ■ Spectral response



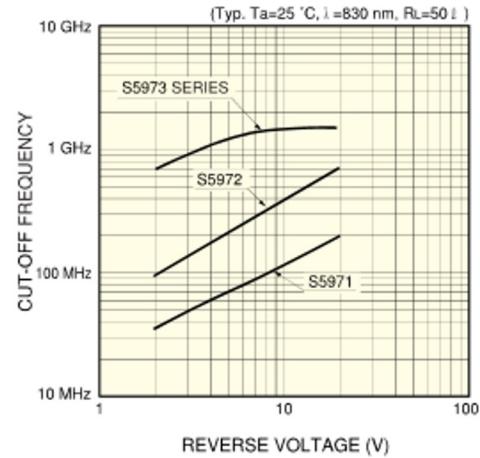
### ■ Photo sensitivity temperature characteristics



### ■ Frequency response

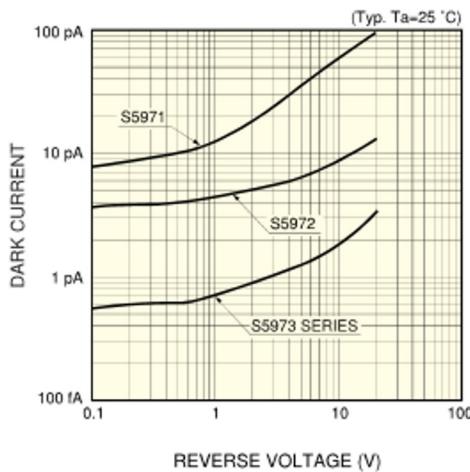


### ■ Cut-off frequency vs. reverse voltage

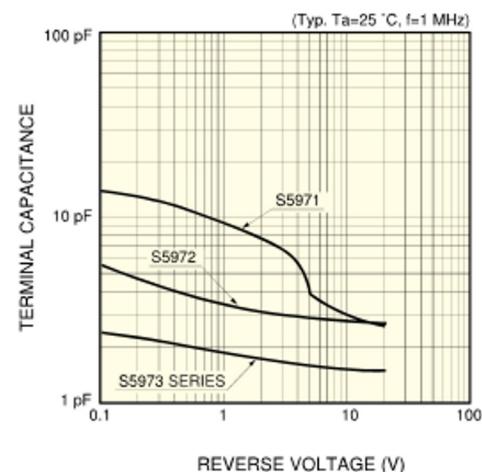


## Si PIN photodiode S5971, S5972, S5973 series

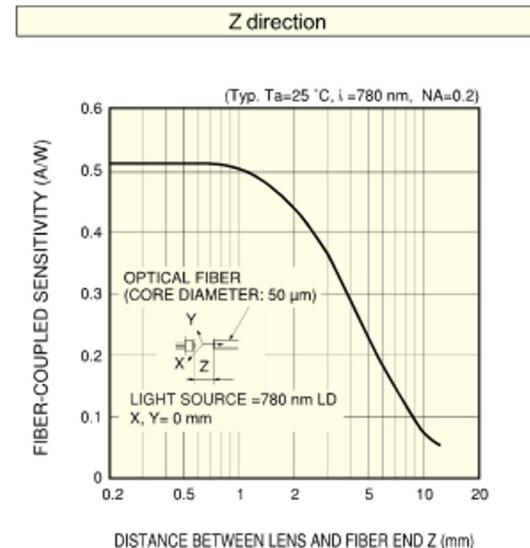
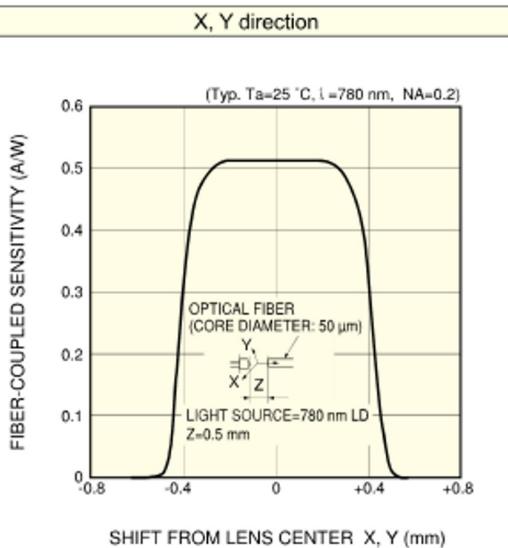
■ Dark current vs. reverse voltage



■ Terminal capacitance vs. reverse voltage



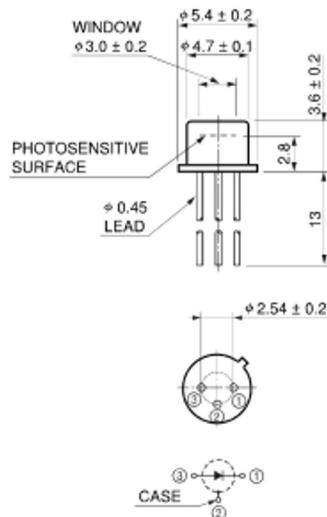
■ Fiber coupling characteristics (S5973-01)



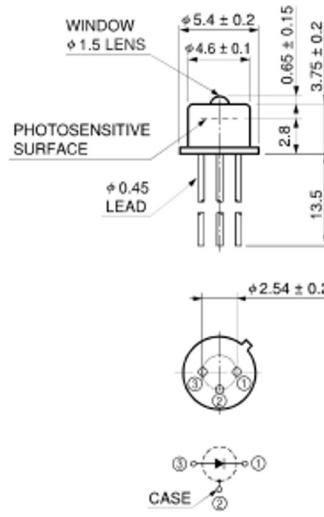
## Si PIN photodiode S5971, S5972, S5973 series

■ Dimensional outlines (unit: mm)

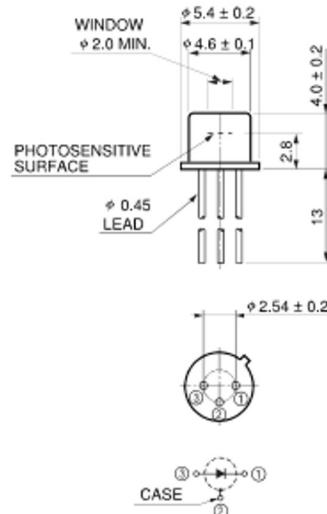
① S5971, S5972, S5973



② S5973-01



③ S5973-02



**HAMAMATSU**

HAMAMATSU PHOTONICS K.K., Solid State Division

Information furnished by HAMAMATSU is believed to be reliable. However, no responsibility is assumed for possible inaccuracies or omissions. Specifications are subject to change without notice. No patent rights are granted to any of the circuits described herein. ©2003 Hamamatsu Photonics K.K.

1126-1 Ichino-cho, Hamamatsu City, 435-8558 Japan, Telephone: (81) 053-434-3311, Fax: (81) 053-434-5184, <http://www.hamamatsu.com>

U.S.A.: Hamamatsu Corporation: 360 Foothill Road, P.O.Box 6910, Bridgewater, NJ 08807-0910, U.S.A., Telephone: (1) 908-231-0960, Fax: (1) 908-231-1218

Germany: Hamamatsu Photonics Deutschland GmbH, Arzbergerstr. 10, D-8221 Herrsching am Ammersee, Germany, Telephone: (49) 08152-3750, Fax: (49) 08152-2658

France: Hamamatsu Photonics France S.A.R.L.; 8, Rue du Saule Trapu, Parc du Moulin de Massy, 91882 Massy Cedex, France, Telephone: 33-(1) 69 53 71 00, Fax: 33-(1) 69 53 71 10

United Kingdom: Hamamatsu Photonics UK Limited; 2 Howard Court, 10 Teelin Road, Welwyn Garden City, Hertfordshire AL7 1BW, United Kingdom, Telephone: (44) 1707-294888, Fax: (44) 1707-325777

North Europe: Hamamatsu Photonics Norden AB; Smidesvägen 12, SE-171 41 Solna, Sweden, Telephone: (46) 8-509-091-00, Fax: (46) 8-509-091-01

Italy: Hamamatsu Photonics Italia S.R.L.; Strada della Mola, 1/E, 20020 Arese, (Milano), Italy, Telephone: (39) 02-935-81-733, Fax: (39) 02-935-81-741

Cat. No. KPIN1025E04  
Aug. 2003 DN

## **D5. TI OPA380A Transimpedance Amplifier Datasheet**



Burr-Brown Products  
from Texas Instruments



OPA380  
OPA2380

SBOS291G – NOVEMBER 2003 – REVISED SEPTEMBER 2007

# Precision, High-Speed Transimpedance Amplifier

## FEATURES

- > 1MHz TRANSIMPEDANCE BANDWIDTH
- EXCELLENT LONG-TERM  $V_{OS}$  STABILITY
- BIAS CURRENT: 50pA (max)
- OFFSET VOLTAGE: 25 $\mu$ V (max)
- DYNAMIC RANGE: 4 to 5 Decades
- DRIFT: 0.1 $\mu$ V/ $^{\circ}$ C (max)
- GAIN BANDWIDTH: 90MHz
- QUIESCENT CURRENT: 7.5mA
- SUPPLY RANGE: 2.7V to 5.5V
- SINGLE AND DUAL VERSIONS
- MicroSize PACKAGE: MSOP-8

## APPLICATIONS

- PHOTODIODE MONITORING
- PRECISION I/V CONVERSION
- OPTICAL AMPLIFIERS
- CAT-SCANNER FRONT-END

## DESCRIPTION

The OPA380 family of transimpedance amplifiers provides high-speed (90MHz Gain Bandwidth [GBW]) operation, with extremely high precision, excellent long-term stability, and very low 1/f noise. It is ideally suited for high-speed photodiode applications. The OPA380 features an offset voltage of 25 $\mu$ V, offset drift of 0.1 $\mu$ V/ $^{\circ}$ C, and bias current of 50pA. The OPA380 far exceeds the offset, drift, and noise performance that conventional JFET op amps provide.

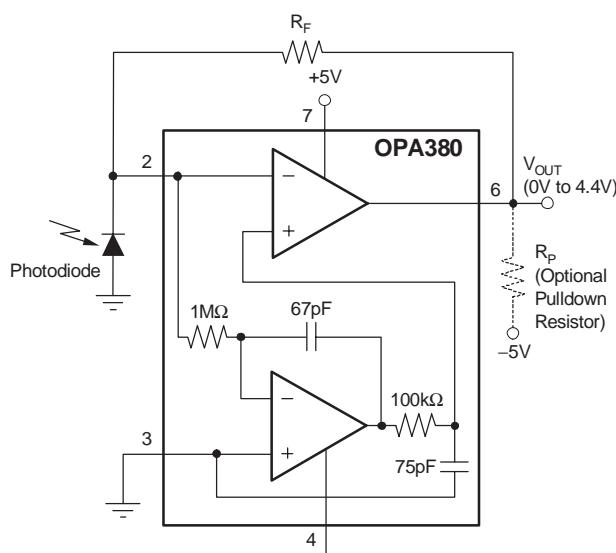
The signal bandwidth of a transimpedance amplifier depends largely on the GBW of the amplifier and the parasitic capacitance of the photodiode, as well as the feedback resistor. The 90MHz GBW of the OPA380 enables a transimpedance bandwidth of > 1MHz in most configurations. The OPA380 is ideally suited for fast control loops for power level on an optical fiber.

As a result of the high precision and low-noise characteristics of the OPA380, a dynamic range of 4 to 5 decades can be achieved. For example, this capability allows the measurement of signal currents on the order of 1nA, and up to 100 $\mu$ A in a single I/V conversion stage. In contrast to logarithmic amplifiers, the OPA380 provides very wide bandwidth throughout the full dynamic range. By using an external pull-down resistor to -5V, the output voltage range can be extended to include 0V.

The OPA380 (single) is available in MSOP-8 and SO-8 packages. The OPA2380 (dual) is available in the miniature MSOP-8 package. They are specified from -40 $^{\circ}$ C to +125 $^{\circ}$ C.

## OPA380 RELATED DEVICES

PRODUCT	FEATURES
OPA300	150MHz CMOS, 2.7V to 5.5V Supply
OPA350	500 $\mu$ V $V_{OS}$ , 38MHz, 2.5V to 5V Supply
OPA335	10 $\mu$ V $V_{OS}$ , Zero-Drift, 2.5V to 5V Supply
OPA132	16MHz GBW, Precision FET Op Amp, $\pm$ 15V
OPA656/7	230MHz, Precision FET, $\pm$ 5V
LOG112	LOG amp, 7.5 decades, $\pm$ 4.5V to $\pm$ 18V Supply
LOG114	LOG amp, 7.5 decades, $\pm$ 2.25V to $\pm$ 5.5V Supply
IVC102	Precision Switched Integrator
DDC112	Dual Current Input, 20-Bit ADC



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

All trademarks are the property of their respective owners.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

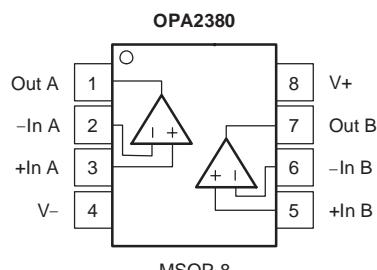
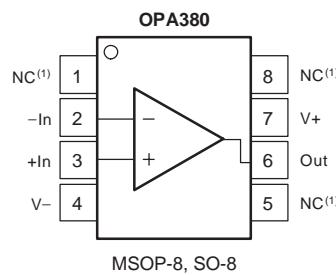
## ABSOLUTE MAXIMUM RATINGS(1)

Voltage Supply .....	+7V
Signal Input Terminals <sup>(2)</sup> , Voltage .....	-0.5V to (V+) + 0.5V
Current .....	±10mA
Short-Circuit Current <sup>(3)</sup> .....	Continuous
Operating Temperature Range .....	-40°C to +125°C
Storage Temperature Range .....	-65°C to +150°C
Junction Temperature .....	+150°C
Lead Temperature (soldering, 10s) .....	+300°C
ESD Rating (Human Body Model) .....	2000V

- (1) Stresses above these ratings may cause permanent damage. Exposure to absolute maximum conditions for extended periods may degrade device reliability. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those specified is not implied.
- (2) Input terminals are diode clamped to the power-supply rails. Input signals that can swing more than 0.5V beyond the supply rails should be current limited to 10mA or less.
- (3) Short-circuit to ground; one amplifier per package.

## PIN ASSIGNMENTS

### Top View



NOTES: (1) NC indicates no internal connection.

## ELECTROSTATIC DISCHARGE SENSITIVITY



This integrated circuit can be damaged by ESD. Texas Instruments recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

## PACKAGE/ORDERING INFORMATION(1)

PRODUCT	PACKAGE-LEAD	PACKAGE MARKING
OPA380	MSOP-8	AUN
OPA380	SO-8	OPA380A
OPA2380	MSOP-8	BBX

(1) For the most current package and ordering information, see the Package Option Addendum at the end of this document, or see the TI web site at [www.ti.com](http://www.ti.com).

## ELECTRICAL CHARACTERISTICS: OPA380 (SINGLE), $V_S = 2.7V$ to $5.5V$

**Boldface** limits apply over the temperature range,  $T_A = -40^\circ\text{C}$  to  $+125^\circ\text{C}$ .

All specifications at  $T_A = +25^\circ\text{C}$ ,  $R_L = 2\text{k}\Omega$  connected to  $V_S/2$ , and  $V_{\text{OUT}} = V_S/2$ , unless otherwise noted.

PARAMETER	CONDITION	OPA380			UNITS
		MIN	TYP	MAX	
<b>OFFSET VOLTAGE</b>					
Input Offset Voltage	$V_{\text{OS}}$				
Drift vs Power Supply	$dV_{\text{OS}}/dT$				
Over Temperature	PSRR				
Long-Term Stability <sup>(1)</sup>					
Channel Separation, dc					
<b>INPUT BIAS CURRENT</b>					
Input Bias Current	$I_B$				
Over Temperature					
Input Offset Current	$I_{\text{OS}}$				
Over Temperature					
<b>NOISE</b>					
Input Voltage Noise, $f = 0.1\text{Hz}$ to $10\text{Hz}$	$e_n$				
Input Voltage Noise Density, $f = 10\text{kHz}$	$e_n$				
Input Voltage Noise Density, $f > 1\text{MHz}$	$e_n$				
Input Current Noise Density, $f = 10\text{kHz}$	$i_n$				
<b>INPUT VOLTAGE RANGE</b>					
Common-Mode Voltage Range	$V_{\text{CM}}$				
Common-Mode Rejection Ratio	CMRR	$(V-) < V_{\text{CM}} < (V+) - 1.8\text{V}$	$V-$ <b>100</b>	$(V+) - 1.8\text{V}$ <b>110</b>	$\text{V}$ <b>dB</b>
<b>INPUT IMPEDANCE</b>					
Differential Capacitance				1.1	$\text{pF}$
Common-Mode Resistance and Inverting Input Capacitance				$10^{13} \parallel 3$	$\Omega \parallel \text{pF}$
<b>OPEN-LOOP GAIN</b>					
Open-Loop Voltage Gain	$A_{\text{OL}}$	$0.1\text{V} < V_O < (V+) - 0.7\text{V}, V_S = 5\text{V}, V_{\text{CM}} = V_S/2$ $0.1\text{V} < V_O < (V+) - 0.6\text{V}, V_S = 5\text{V}, V_{\text{CM}} = V_S/2,$ $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$ $0\text{V} < V_O < (V+) - 0.7\text{V}, V_S = 5\text{V}, V_{\text{CM}} = 0\text{V},$ $R_P = 2\text{k}\Omega$ to $-5\text{V}$ <sup>(2)</sup> $0\text{V} < V_O < (V+) - 0.6\text{V}, V_S = 5\text{V}, V_{\text{CM}} = 0\text{V},$ $R_P = 2\text{k}\Omega$ to $-5\text{V}$ <sup>(2)</sup> , $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$	<b>110</b>	<b>130</b>	<b>dB</b>
				110	$\text{dB}$
				<b>106</b>	$\text{dB}$
				106	$\text{dB}$
<b>FREQUENCY RESPONSE</b>					
Gain-Bandwidth Product	GBW	$C_L = 50\text{pF}$		90	$\text{MHz}$
Slew Rate	SR	$G = +1$		80	$\text{V}/\mu\text{s}$
Settling Time, 0.01% <sup>(3)</sup>	ts	$V_S = +5\text{V}, 4\text{V Step}, G = +1$		2	$\mu\text{s}$
Overload Recovery Time <sup>(4)(5)</sup>		$V_{\text{IN}} \times G > V_S$		100	$\text{ns}$
<b>OUTPUT</b>					
Voltage Output Swing from Positive Rail		$R_L = 2\text{k}\Omega$		400	$\text{mV}$
Voltage Output Swing from Negative Rail		$R_L = 2\text{k}\Omega$		60	$\text{mV}$
Voltage Output Swing from Positive Rail		$R_P = 2\text{k}\Omega$ to $-5\text{V}$ <sup>(2)</sup>		400	$\text{mV}$
Voltage Output Swing from Negative Rail		$R_P = 2\text{k}\Omega$ to $-5\text{V}$ <sup>(2)</sup>		-20	$\text{mV}$
Output Current	$I_{\text{OUT}}$			See Typical Characteristics	
Short-Circuit Current	$I_{\text{SC}}$			150	$\text{mA}$
Capacitive Load Drive	$C_{\text{LOAD}}$			See Typical Characteristics	
Open-Loop Output Impedance	$R_O$	$f = 1\text{MHz}, I_O = 0\text{A}$		40	$\Omega$
<b>POWER SUPPLY</b>					
Specified Voltage Range	$V_S$			2.7	$\text{V}$
Quiescent Current	$I_Q$	$I_O = 0\text{A}$		7.5	$\text{mA}$
Over Temperature				5.5	$\text{mA}$
				9.5	$\text{mA}$
				<b>10</b>	$\text{mA}$
<b>TEMPERATURE RANGE</b>					
Specified and Operating Range				-40	$^\circ\text{C}$
Storage Range				-65	$^\circ\text{C}$
Thermal Resistance	$\theta_{JA}$			150	$^\circ\text{C/W}$
MSOP-8, SO-8					

(1) 300-hour life test at  $150^\circ\text{C}$  demonstrated randomly distributed variation approximately equal to measurement repeatability of  $1\mu\text{V}$ .

(2) Tested with output connected only to  $R_P$ , a pulldown resistor connected between  $V_{\text{OUT}}$  and  $-5\text{V}$ , as shown in Figure 5. See also applications section, *Achieving Output Swing to Ground*.

(3) Transimpedance frequency of  $1\text{MHz}$ .

(4) Time required to return to linear operation.

(5) From positive rail.

**ELECTRICAL CHARACTERISTICS: OPA2380 (DUAL),  $V_S = 2.7\text{V}$  to  $5.5\text{V}$** 
**Boldface** limits apply over the temperature range,  $T_A = -40^\circ\text{C}$  to  $+125^\circ\text{C}$ .
All specifications at  $T_A = +25^\circ\text{C}$ ,  $R_L = 2\text{k}\Omega$  connected to  $V_S/2$ , and  $V_{\text{OUT}} = V_S/2$ , unless otherwise noted.

PARAMETER	CONDITION	OPA2380			UNITS
		MIN	TYP	MAX	
<b>OFFSET VOLTAGE</b>					
Input Offset Voltage	$V_S = +5\text{V}$ , $V_{\text{CM}} = 0\text{V}$		4	25	$\mu\text{V}$
<b>Drift</b>	$dV_{\text{OS}}/dT$		<b>0.03</b>	<b>0.1</b>	$\mu\text{V}/^\circ\text{C}$
vs Power Supply	$V_S = +2.7\text{V}$ to $+5.5\text{V}$ , $V_{\text{CM}} = 0\text{V}$	2.4		10	$\mu\text{V}/\text{V}$
<b>Over Temperature</b>	$V_S = +2.7\text{V}$ to $+5.5\text{V}$ , $V_{\text{CM}} = 0\text{V}$			<b>10</b>	$\mu\text{V}/\text{V}$
Long-Term Stability <sup>(1)</sup>			See Note (1)		
Channel Separation, dc		1			$\mu\text{V}/\text{V}$
<b>INPUT BIAS CURRENT</b>					
Input Bias Current, Inverting Input	$I_B$		3	$\pm 50$	pA
Noninverting Input	$I_B$		3	$\pm 200$	pA
<b>Over Temperature</b>					
<b>NOISE</b>					
Input Voltage Noise, $f = 0.1\text{Hz}$ to $10\text{Hz}$	$e_n$		3		$\mu\text{V}_{\text{PP}}$
Input Voltage Noise Density, $f = 10\text{kHz}$	$e_n$		67		$\text{nV}/\sqrt{\text{Hz}}$
Input Voltage Noise Density, $f > 1\text{MHz}$	$e_n$		5.8		$\text{nV}/\sqrt{\text{Hz}}$
Input Current Noise Density, $f = 10\text{kHz}$	$i_n$		10		$\text{fA}/\sqrt{\text{Hz}}$
<b>INPUT VOLTAGE RANGE</b>					
Common-Mode Voltage Range	$V_{\text{CM}}$				V
<b>Common-Mode Rejection Ratio</b>	$CMRR$	$(V-) < V_{\text{CM}} < (V+) - 1.8\text{V}$	$V-$ <b>95</b>	<b>105</b>	$(V+) - 1.8\text{V}$
<b>INPUT IMPEDANCE</b>					
Differential Capacitance			1.1		pF
Common-Mode Resistance and Inverting Input Capacitance			$10^{13} \parallel 3$		$\Omega \parallel \text{pF}$
<b>OPEN-LOOP GAIN</b>					
Open-Loop Voltage Gain	$A_{\text{OL}}$	$0.12\text{V} < V_O < (V+) - 0.7\text{V}$ , $V_S = 5\text{V}$ , $V_{\text{CM}} = V_S/2$	<b>110</b>	<b>130</b>	<b>dB</b>
		$0.12\text{V} < V_O < (V+) - 0.6\text{V}$ , $V_S = 5\text{V}$ , $V_{\text{CM}} = V_S/2$ , $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$	110	130	dB
		$0\text{V} < V_O < (V+) - 0.7\text{V}$ , $V_S = 5\text{V}$ , $V_{\text{CM}} = 0\text{V}$ , $R_P = 2\text{k}\Omega$ to $-5\text{V}$ <sup>(2)</sup>	<b>106</b>	<b>120</b>	<b>dB</b>
		$0\text{V} < V_O < (V+) - 0.6\text{V}$ , $V_S = 5\text{V}$ , $V_{\text{CM}} = 0\text{V}$ , $R_P = 2\text{k}\Omega$ to $-5\text{V}$ <sup>(2)</sup> , $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$	106	120	dB
<b>FREQUENCY RESPONSE</b>					
Gain-Bandwidth Product	$GBW$	$C_L = 50\text{pF}$		90	MHz
Slew Rate	$SR$	$G = +1$		80	$\text{V}/\mu\text{s}$
Settling Time, 0.01% <sup>(3)</sup>	$t_s$	$V_S = +5\text{V}$ , 4V Step, $G = +1$		2	$\mu\text{s}$
Overload Recovery Time <sup>(4)(5)</sup>		$V_{\text{IN}} \times G > V_S$		100	ns
<b>OUTPUT</b>					
Voltage Output Swing from Positive Rail		$R_L = 2\text{k}\Omega$		400	mV
Voltage Output Swing from Negative Rail		$R_L = 2\text{k}\Omega$		80	mV
Voltage Output Swing from Positive Rail		$R_P = 2\text{k}\Omega$ to $-5\text{V}$ <sup>(2)</sup>		400	mV
Voltage Output Swing from Negative Rail		$R_P = 2\text{k}\Omega$ to $-5\text{V}$ <sup>(2)</sup>		-20	mV
Output Current	$I_{\text{OUT}}$				
Short-Circuit Current	$I_{\text{SC}}$				
Capacitive Load Drive	$C_{\text{LOAD}}$				
Open-Loop Output Impedance	$R_O$	$f = 1\text{MHz}$ , $I_O = 0\text{A}$			
<b>POWER SUPPLY</b>					
Specified Voltage Range	$V_S$				V
Quiescent Current (per amplifier)	$I_Q$	$I_O = 0\text{A}$		2.7	mA
<b>Over Temperature</b>				7.5	mA
<b>TEMPERATURE RANGE</b>					
Specified and Operating Range				5.5	
Storage Range				9.5	
Thermal Resistance	$\theta_{JA}$			<b>10</b>	
MSOP-8					

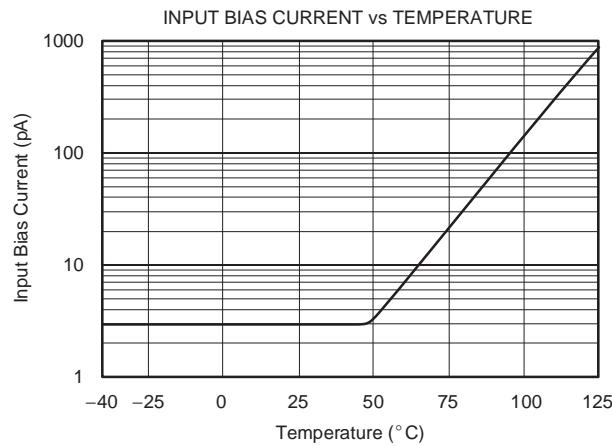
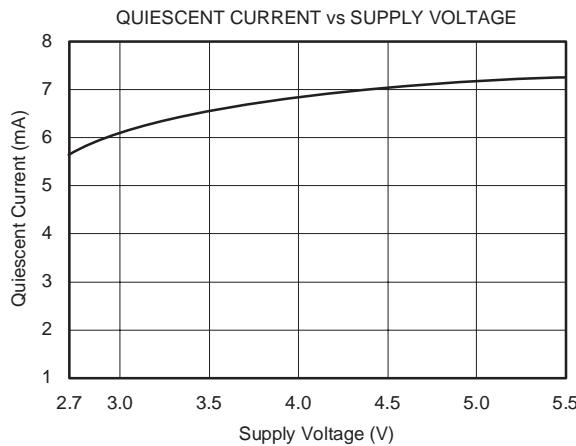
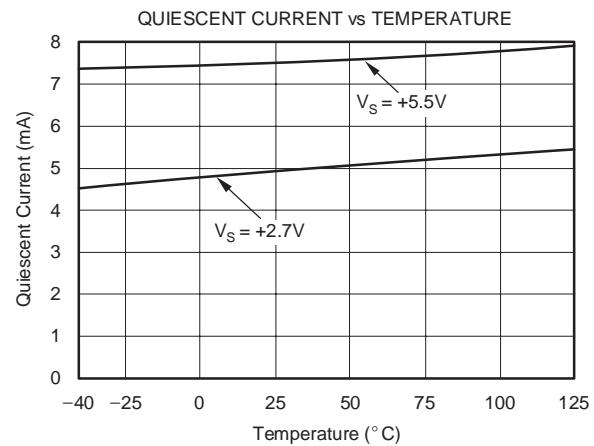
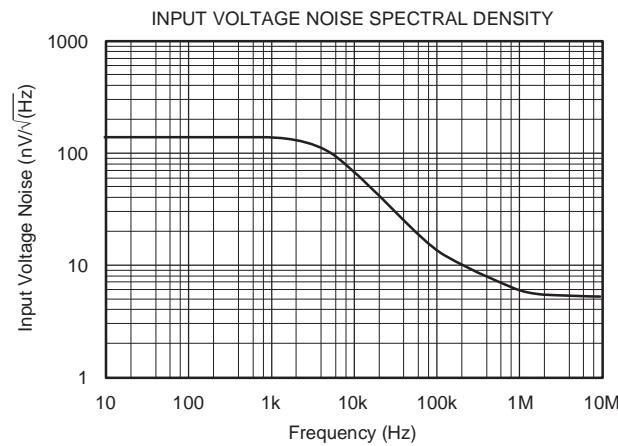
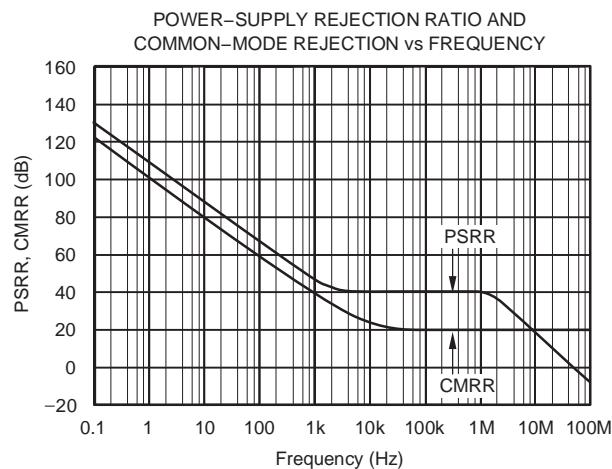
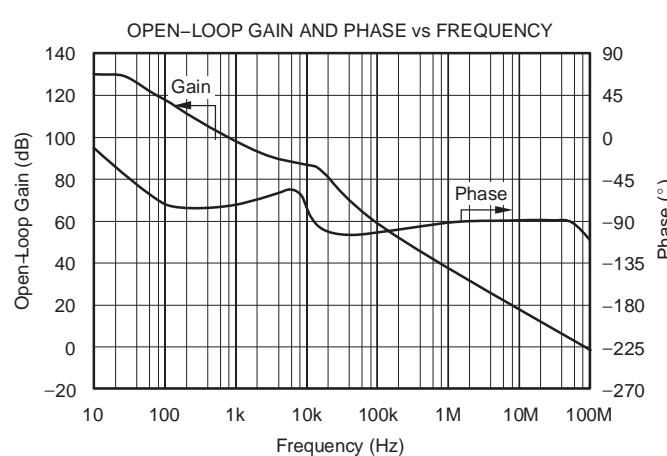
(1) 300-hour life test at  $150^\circ\text{C}$  demonstrated randomly distributed variation approximately equal to measurement repeatability of  $1\mu\text{V}$ .(2) Tested with output connected only to  $R_P$ , a pulldown resistor connected between  $V_{\text{OUT}}$  and  $-5\text{V}$ , as shown in Figure 5. See also applications section, *Achieving Output Swing to Ground*.(3) Transimpedance frequency of  $1\text{MHz}$ .

(4) Time required to return to linear operation.

(5) From positive rail.

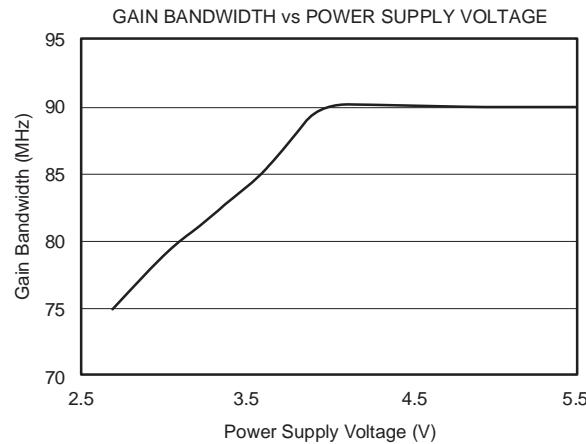
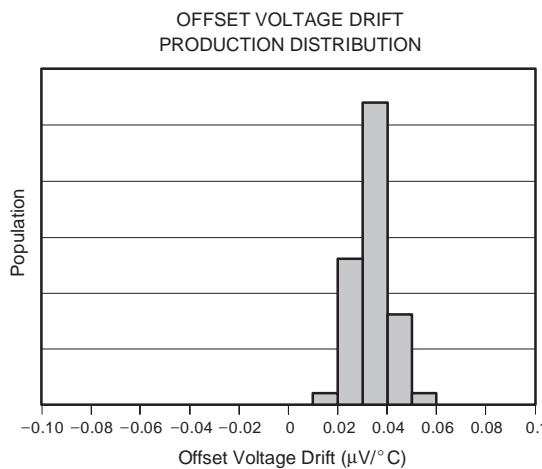
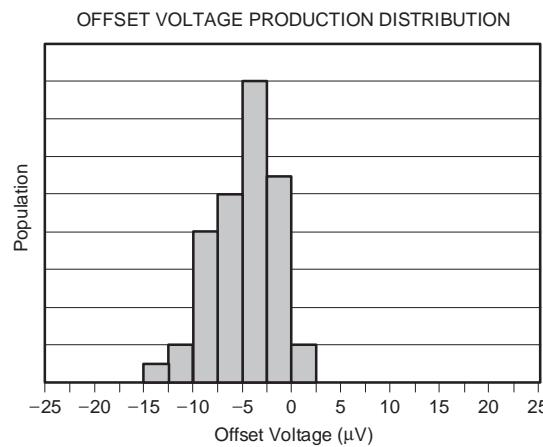
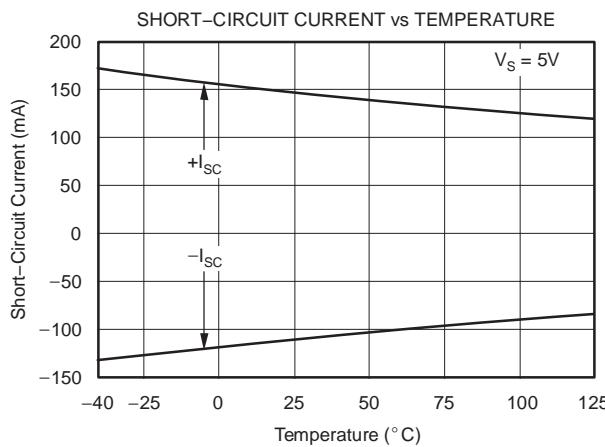
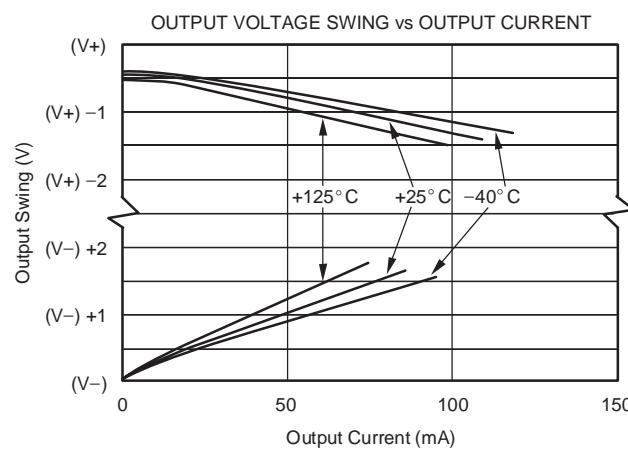
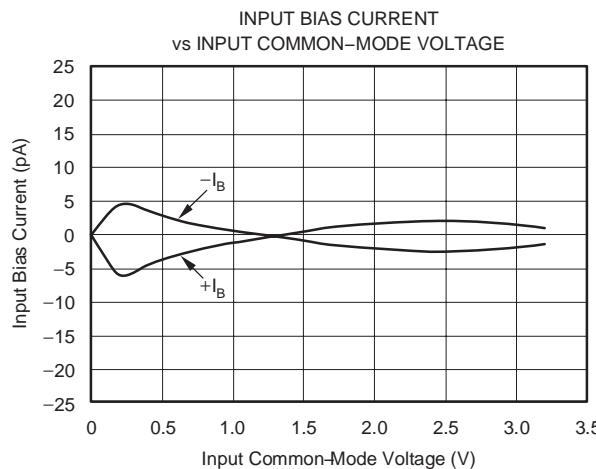
## TYPICAL CHARACTERISTICS: $V_S = +2.7\text{V}$ to $+5.5\text{V}$

All specifications at  $T_A = +25^\circ\text{C}$ ,  $R_L = 2\text{k}\Omega$  connected to  $V_S/2$ , and  $V_{\text{OUT}} = V_S/2$ , unless otherwise noted.



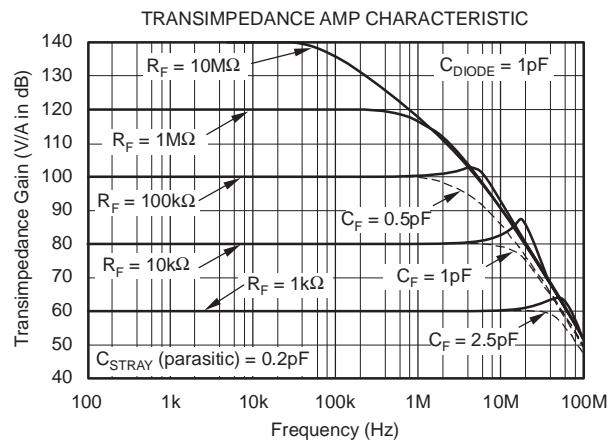
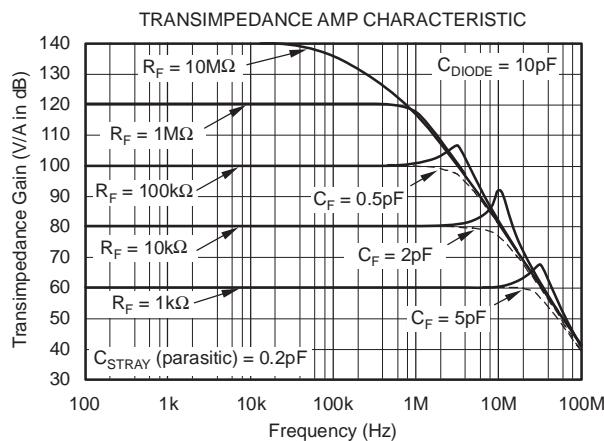
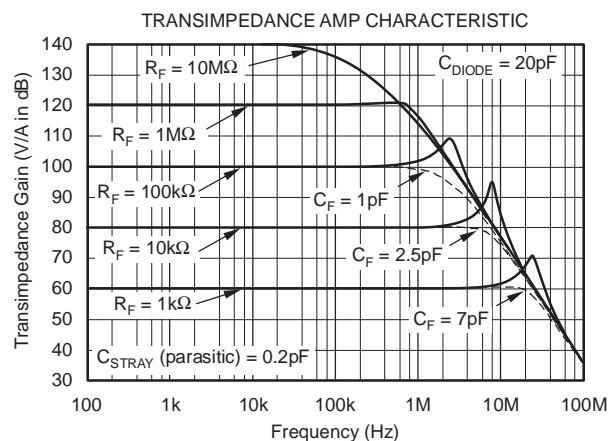
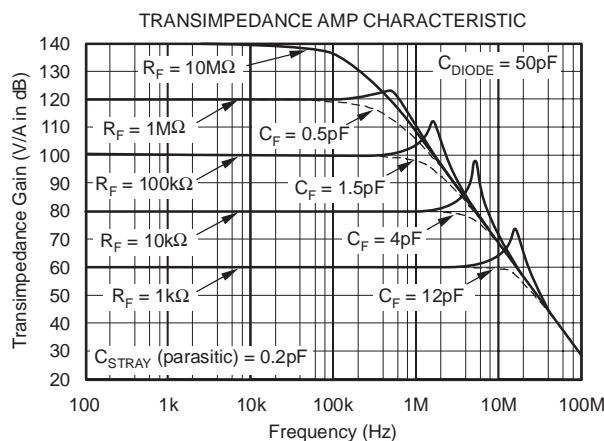
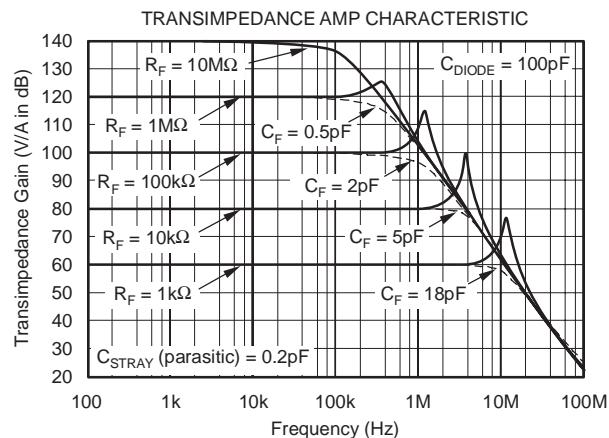
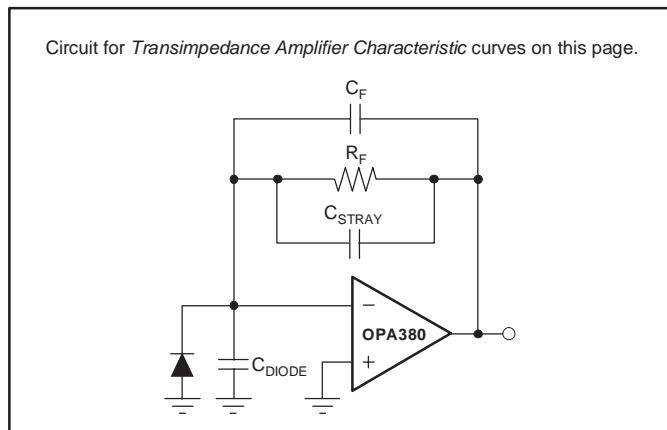
### TYPICAL CHARACTERISTICS: $V_S = +2.7\text{V}$ to $+5.5\text{V}$ (continued)

All specifications at  $T_A = +25^\circ\text{C}$ ,  $R_L = 2\text{k}\Omega$  connected to  $V_S/2$ , and  $V_{\text{OUT}} = V_S/2$ , unless otherwise noted.



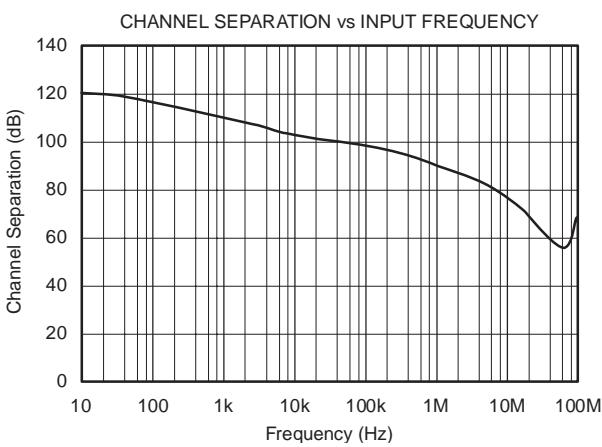
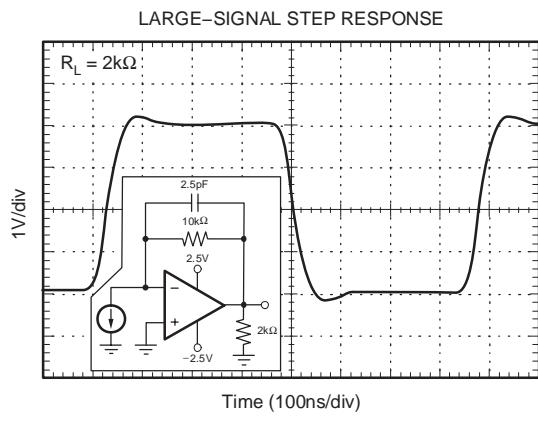
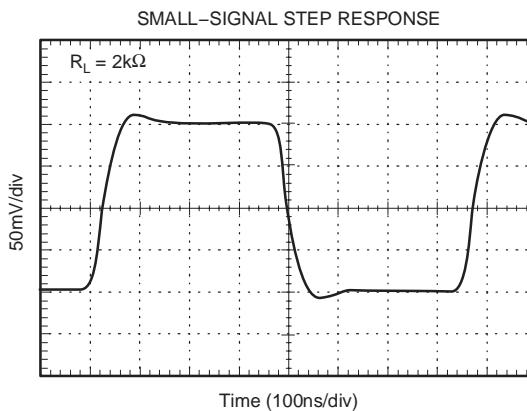
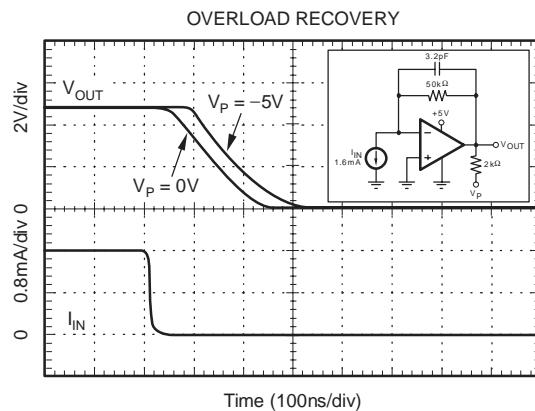
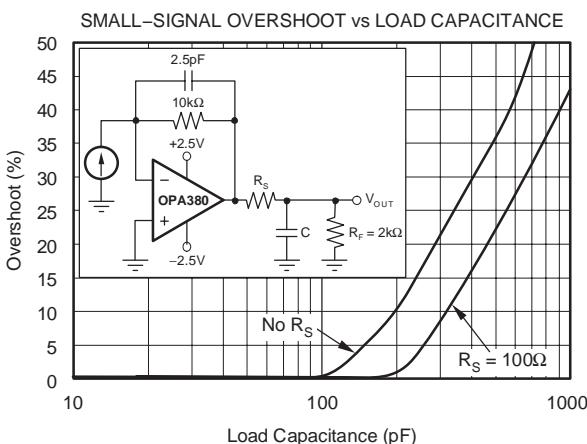
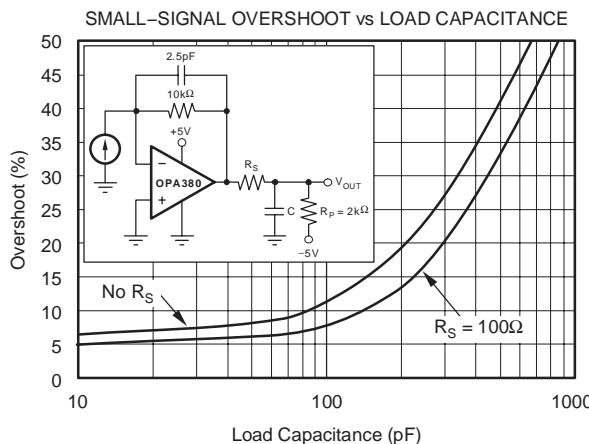
## TYPICAL CHARACTERISTICS: $V_S = +2.7V$ to $+5.5V$ (continued)

All specifications at  $T_A = +25^\circ\text{C}$ ,  $R_L = 2\text{k}\Omega$  connected to  $V_S/2$ , and  $V_{\text{OUT}} = V_S/2$ , unless otherwise noted.



### TYPICAL CHARACTERISTICS: $V_S = +2.7V$ to $+5.5V$ (continued)

All specifications at  $T_A = +25^\circ\text{C}$ ,  $R_L = 2\text{k}\Omega$  connected to  $V_S/2$ , and  $V_{\text{OUT}} = V_S/2$ , unless otherwise noted.



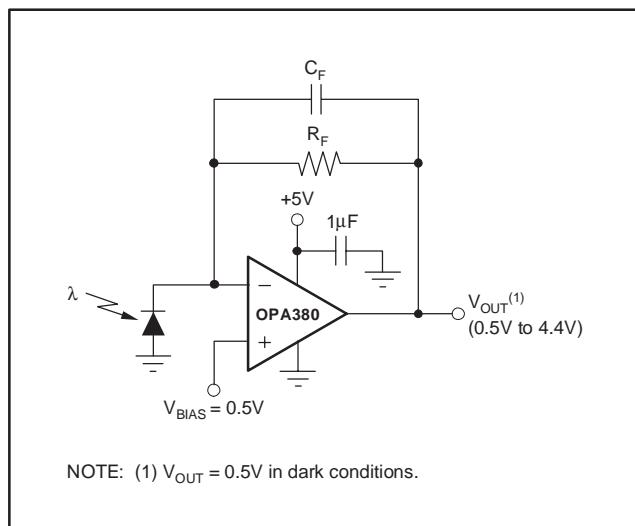
## APPLICATIONS INFORMATION

### BASIC OPERATION

The OPA380 is a high-performance transimpedance amplifier with very low 1/f noise. As a result of its unique architecture, the OPA380 has excellent long-term input voltage offset stability—a 300-hour life test at 150°C demonstrated randomly distributed variation approximately equal to measurement repeatability of 1µV.

The OPA380 performance results from an internal auto-zero amplifier combined with a high-speed amplifier. The OPA380 has been designed with circuitry to improve overload recovery and settling time over a traditional composite approach. It has been specifically designed and characterized to accommodate circuit options to allow 0V output operation (see Figure 3).

The OPA380 is used in inverting configurations, with the noninverting input used as a fixed biasing point. Figure 1 shows the OPA380 in a typical configuration. Power-supply pins should be bypassed with 1µF ceramic or tantalum capacitors. Electrolytic capacitors are not recommended.



**Figure 1. OPA380 Typical Configuration**

### OPERATING VOLTAGE

The OPA380 series op amps are fully specified from 2.7V to 5.5V over a temperature range of -40°C to +125°C. Parameters that vary significantly with operating voltages or temperature are shown in the Typical Characteristics.

### INTERNAL OFFSET CORRECTION

The OPA380 series op amps use an auto-zero topology with a time-continuous 90MHz op amp in the signal path. This amplifier is zero-corrected every 100µs using a proprietary technique. Upon power-up, the amplifier requires approximately 400µs to achieve specified  $V_{OS}$  accuracy, which includes one full auto-zero cycle of approximately 100µs and the start-up time for the bias circuitry. Prior to this time, the amplifier will function properly but with unspecified offset voltage.

This design has virtually no aliasing and very low noise. Zero correction occurs at a 10kHz rate, but there is very little fundamental noise energy present at that frequency due to internal filtering. For all practical purposes, any glitches have energy at 20MHz or higher and are easily filtered, if required. Most applications are not sensitive to such high-frequency noise, and no filtering is required.

### INPUT VOLTAGE

The input common-mode voltage range of the OPA380 series extends from  $V_-$  to  $(V_+)$  - 1.8V. With input signals above this common-mode range, the amplifier will no longer provide a valid output value, but it will not latch or invert.

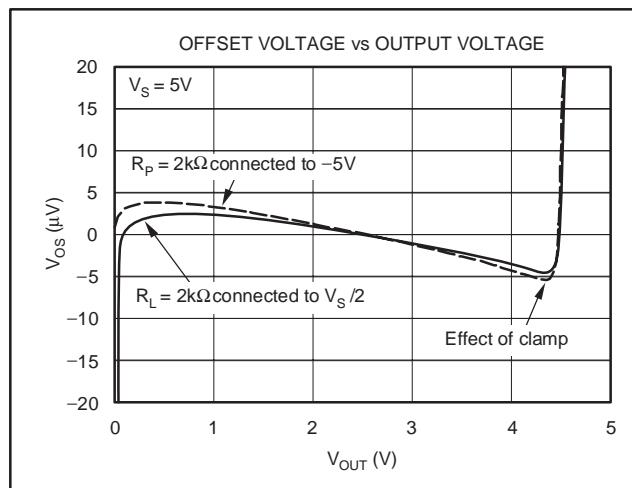
### INPUT OVERVOLTAGE PROTECTION

Device inputs are protected by ESD diodes that will conduct if the input voltages exceed the power supplies by more than approximately 500mV. Momentary voltages greater than 500mV beyond the power supply can be tolerated if the current is limited to 10mA. The OPA380 series feature no phase inversion when the inputs extend beyond supplies if the input is current limited.

## OUTPUT RANGE

The OPA380 is specified to swing within at least 600mV of the positive rail and 100mV of the negative rail with a  $2\text{k}\Omega$  load with excellent linearity. Swing to the negative rail while maintaining good linearity can be extended to 0V—see the section, *Achieving Output Swing to Ground*. See the Typical Characteristic curve, *Output Voltage Swing vs Output Current*.

The OPA380 can swing slightly closer than specified to the positive rail; however, linearity will decrease and a high-speed overload recovery clamp limits the amount of positive output voltage swing available, as shown in Figure 2.



**Figure 2. Effect of High-Speed Overload Recovery Clamp on Output Voltage**

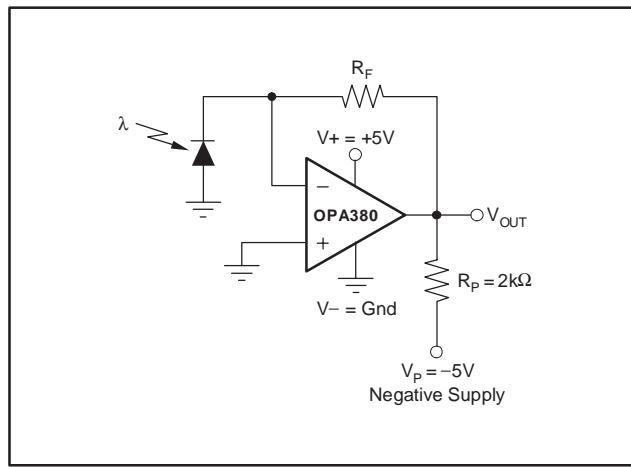
## OVERLOAD RECOVERY

The OPA380 has been designed to prevent output saturation. After being overdriven to the positive rail, it will typically require only 100ns to return to linear operation. The time required for negative overload recovery is greater, unless a pull-down resistor connected to a more negative supply is used to extend the output swing all the way to the negative rail—see the following section, *Achieving Output Swing to Ground*.

## ACHIEVING OUTPUT SWING TO GROUND

Some applications require output voltage swing from 0V to a positive full-scale voltage (such as +4.096V) with excellent accuracy. With most single-supply op amps, problems arise when the output signal approaches 0V, near the lower output swing limit of a single-supply op amp. A good single-supply op amp may swing close to single-supply ground, but will not reach 0V.

The output of the OPA380 can be made to swing to ground, or slightly below, on a single-supply power source. This extended output swing requires the use of another resistor and an additional negative power supply. A pull-down resistor may be connected between the output and the negative supply to pull the output down to 0V. See Figure 3.

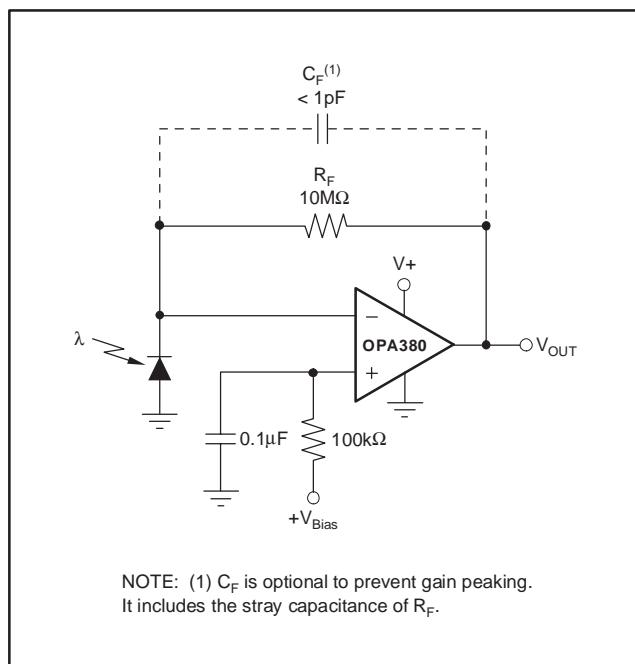


**Figure 3. Amplifier with Optional Pull-Down Resistor to Achieve  $V_{OUT} = 0\text{V}$**

The OPA380 has an output stage that allows the output voltage to be pulled to its negative supply rail using this technique. However, this technique only works with some types of output stages. The OPA380 has been designed to perform well with this method. Accuracy is excellent down to 0V. Reliable operation is assured over the specified temperature range.

## BIASING PHOTODIODES IN SINGLE-SUPPLY CIRCUITS

The +IN input can be biased with a positive DC voltage to offset the output voltage and allow the amplifier output to indicate a true zero photodiode measurement when the photodiode is not exposed to any light. It will also prevent the added delay that results from coming out of the negative rail. This bias voltage appears across the photodiode, providing a reverse bias for faster operation. An RC filter placed at this bias point will reduce noise, as shown in Figure 4. This bias voltage can also serve as an offset bias point for an ADC with range that does not include ground.



**Figure 4. Filtered Reverse Bias Voltage**

## TRANSIMPEDANCE AMPLIFIER

Wide bandwidth, low input bias current, and low input voltage and current noise make the OPA380 an ideal wideband photodiode transimpedance amplifier. Low-voltage noise is important because photodiode capacitance causes the effective noise gain of the circuit to increase at high frequency.

The key elements to a transimpedance design are shown in Figure 5:

the total input capacitance ( $C_{TOT}$ ), consisting of the photodiode capacitance ( $C_{DIODE}$ ) plus the parasitic common-mode and differential-mode input capacitance (3pF + 1.1pF for the OPA380);

the desired transimpedance gain ( $R_F$ );

the Gain Bandwidth Product (GBW) for the OPA380 (90MHz).

With these three variables set, the feedback capacitor value ( $C_F$ ) can be set to control the frequency response.  $C_{STRAY}$  is the stray capacitance of  $R_F$ , which is 0.2pF for a typical surface-mount resistor.

To achieve a maximally flat, 2nd-order, Butterworth frequency response, the feedback pole should be set to:

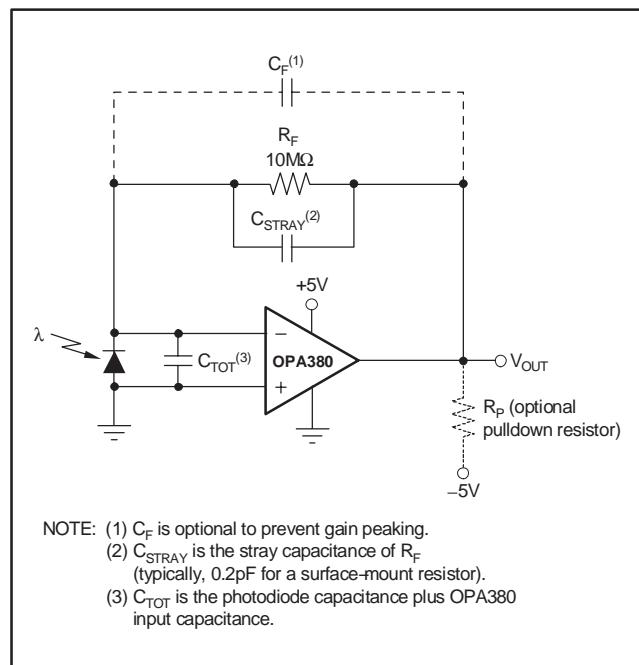
$$\frac{1}{2\pi R_F(C_F + C_{STRAY})} = \sqrt{\frac{GBW}{4\pi R_F C_{TOT}}} \quad (1)$$

Bandwidth is calculated by:

$$f_{-3dB} = \sqrt{\frac{GBW}{2\pi R_F C_{TOT}}} \text{ Hz} \quad (2)$$

These equations will result in maximum transimpedance bandwidth. For even higher transimpedance bandwidth, the high-speed CMOS OPA300 (SBOS271 (180MHz GBW)), or the OPA656 (SBOS196 (230MHz GBW)) may be used.

For additional information, refer to Application Bulletin AB-050 (SBOA055), *Compensate Transimpedance Amplifiers Intuitively*, available for download at [www.ti.com](http://www.ti.com).



**Figure 5. Transimpedance Amplifier**

## TRANSIMPEDANCE BANDWIDTH AND NOISE

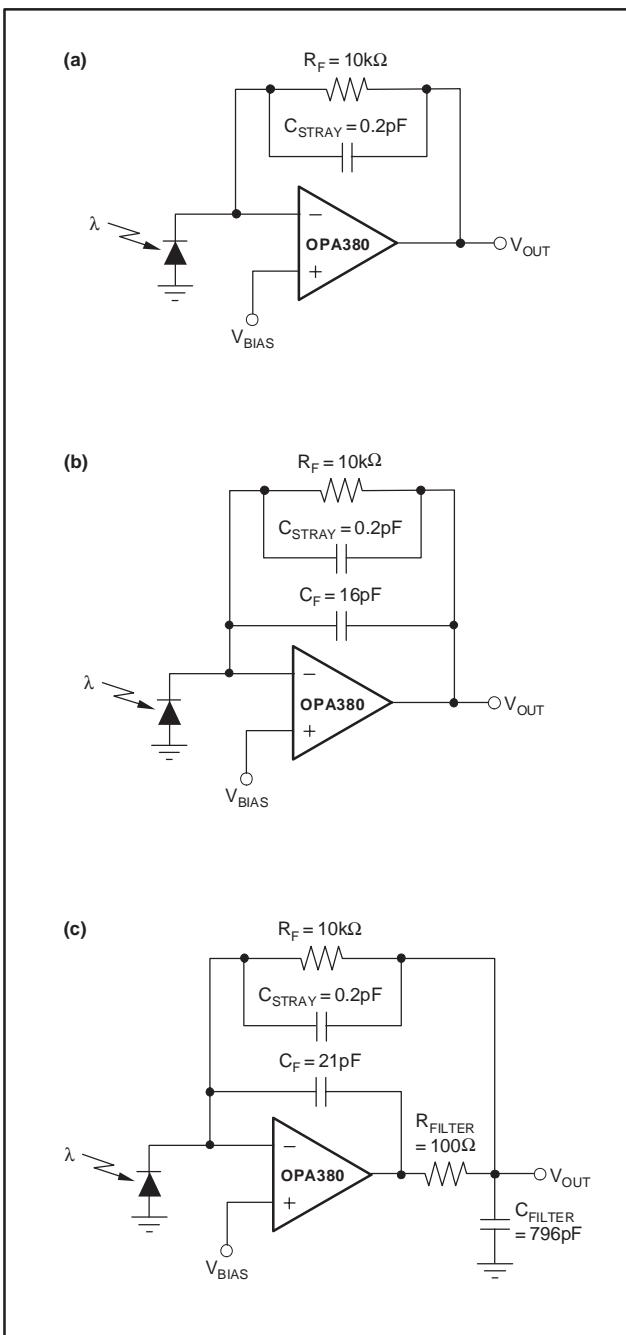
Limiting the gain set by  $R_F$  can decrease the noise occurring at the output of the transimpedance circuit. However, all required gain should occur in the transimpedance stage, since adding gain after the transimpedance amplifier generally produces poorer noise performance. The noise spectral density produced by  $R_F$  increases with the square-root of  $R_F$ , whereas the signal increases linearly. Therefore, signal-to-noise ratio is improved when all the required gain is placed in the transimpedance stage.

Total noise increases with increased bandwidth. Limit the circuit bandwidth to only that required. Use a capacitor,  $C_F$ , across the feedback resistor,  $R_F$ , to limit bandwidth, even if not required for stability if total output noise is a concern.

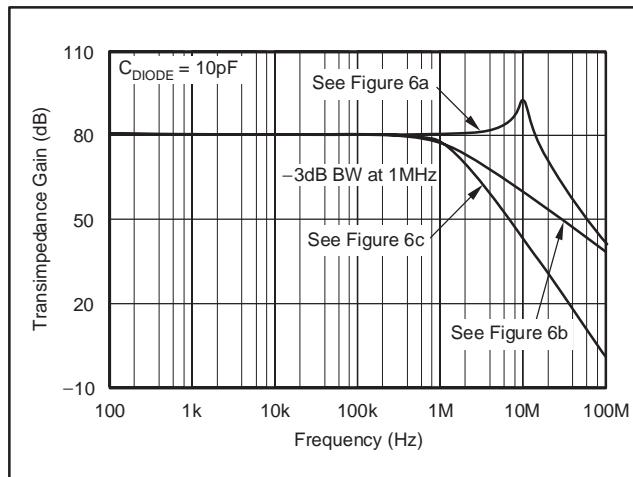
Figure 6a shows the transimpedance circuit without any feedback capacitor. The resulting transimpedance gain of this circuit is shown in Figure 7. The  $-3\text{dB}$  point is approximately 10MHz. Adding a 16pF feedback capacitor (Figure 6b) will limit the bandwidth and result in a  $-3\text{dB}$  point at approximately 1MHz (see Figure 7). Output noise will be further reduced by adding a filter ( $R_{\text{FILTER}}$  and  $C_{\text{FILTER}}$ ) to create a second pole (Figure 6c). This second pole is placed within the feedback loop to maintain the amplifier's low output impedance. (If the pole was placed outside the feedback loop, an additional buffer would be required and would inadvertently increase noise and dc error).

Using  $R_{\text{DIODE}}$  to represent the equivalent diode resistance, and  $C_{\text{TOT}}$  for equivalent diode capacitance plus OPA380 input capacitance, the noise zero,  $f_Z$ , is calculated by:

$$f_Z = \frac{(R_{\text{DIODE}} + R_F)}{2\pi R_{\text{DIODE}} R_F (C_{\text{TOT}} + C_F)} \quad (3)$$



**Figure 6. Transimpedance Circuit Configurations with Varying Total and Integrated Noise Gain**



**Figure 7. Transimpedance Gains for Circuits in Figure 6**

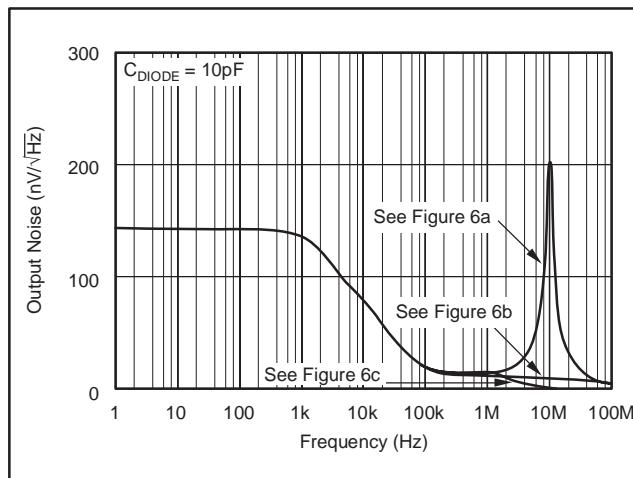
The effect of these circuit configurations on output noise is shown in Figure 8 and on integrated output noise in Figure 9. A 2-pole Butterworth filter (maximally flat in passband) is created by selecting the filter values using the equation:

$$C_F R_F = 2C_{\text{FILTER}} R_{\text{FILTER}} \quad (4)$$

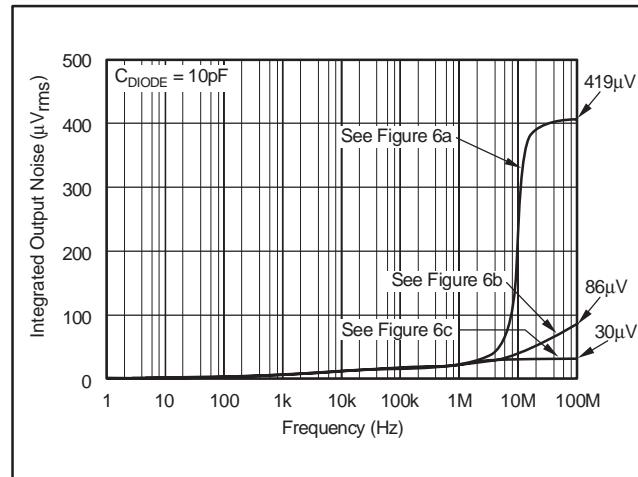
with:

$$f_{-3\text{dB}} = \frac{1}{2\pi \sqrt{R_F R_{\text{FILTER}} C_F C_{\text{FILTER}}}} \quad (5)$$

The circuit in Figure 6b rolls off at 20dB/decade. The circuit with the additional filter shown in Figure 6c rolls off at 40dB/decade, resulting in improved noise performance.

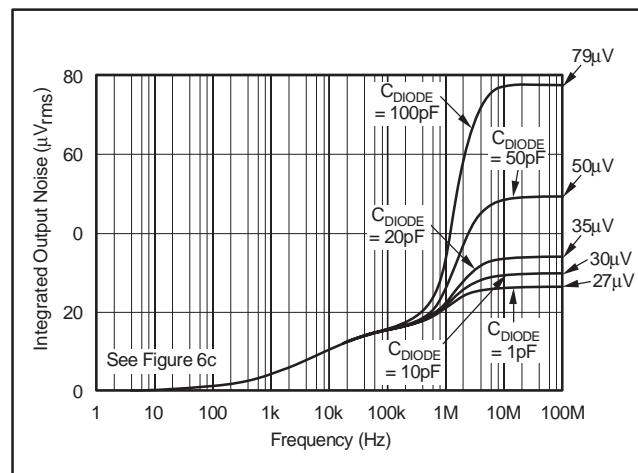


**Figure 8. Output Noise for Circuits in Figure 6**



**Figure 9. Integrated Output Noise for Circuits in Figure 6**

Figure 10 shows the effect of diode capacitance on integrated output noise, using the circuit in Figure 6c. For additional information, refer to *Noise Analysis of FET Transimpedance Amplifiers* (SBOA060), and *Noise Analysis for High-Speed Op Amps* (SBOA066), available for download from the TI web site.



**Figure 10. Integrated Output Noise for Various Values of  $C_{\text{DIODE}}$  for Circuit in Figure 6c**

## BOARD LAYOUT

Minimize photodiode capacitance and stray capacitance at the summing junction (inverting input). This capacitance causes the voltage noise of the op amp to be amplified (increasing amplification at high frequency). Using a low-noise voltage source to reverse-bias a photodiode can significantly reduce its capacitance. Smaller photodiodes have lower capacitance. Use optics to concentrate light on a small photodiode.

Circuit board leakage can degrade the performance of an otherwise well-designed amplifier. Clean the circuit board carefully. A circuit board guard trace that encircles the summing junction and is driven at the same voltage can help control leakage, as shown in Figure 11.

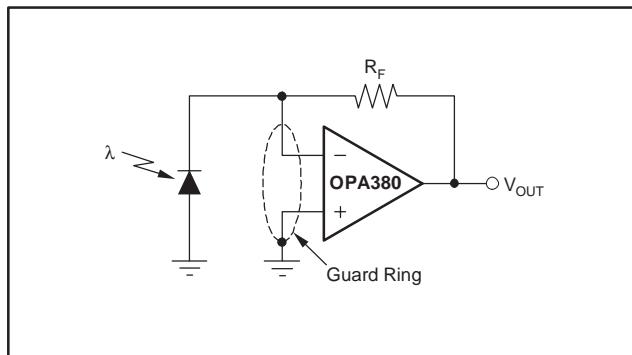


Figure 11. Connection of Input Guard

## OTHER WAYS TO MEASURE SMALL CURRENTS

Logarithmic amplifiers are used to compress extremely wide dynamic range input currents to a much narrower range. Wide input dynamic ranges of 8 decades, or 100pA to 10mA, can be accommodated for input to a 12-bit ADC. (Suggested products: LOG101, LOG102, LOG104, and LOG112.)

Extremely small currents can be accurately measured by integrating currents on a capacitor. (Suggested product: IVC102.)

Low-level currents can be converted to high-resolution data words. (Suggested product: DDC112.)

For further information on the range of products available, search [www.ti.com](http://www.ti.com) using the above specific model names or by using keywords *transimpedance* and *logarithmic*.

## CAPACITIVE LOAD AND STABILITY

The OPA380 series op amps can drive up to 500pF pure capacitive load. Increasing the gain enhances the amplifier's ability to drive greater capacitive loads (see the Typical Characteristic curve, *Small-Signal Overshoot vs Capacitive Load*).

One method of improving capacitive load drive in the unity-gain configuration is to insert a 10Ω to 20Ω resistor in series with the load. This reduces ringing with large capacitive loads while maintaining DC accuracy.

## DRIVING FAST 16-BIT ANALOG-TO-DIGITAL CONVERTERS (ADC)

The OPA380 series is optimized for driving a fast 16-bit ADC such as the ADS8411. The OPA380 op amp buffers the converter's input capacitance and resulting charge injection while providing signal gain. Figure 12 shows the OPA380 in a single-ended method of interfacing the ADS8411 16-bit, 2MSPS ADC. For additional information, refer to the ADS8411 data sheet.

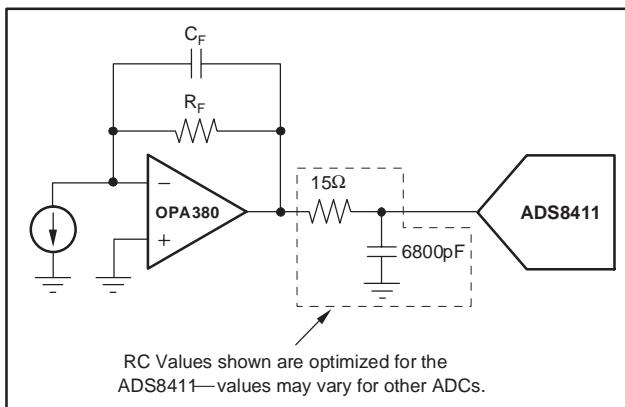


Figure 12. Driving 16-Bit ADCs

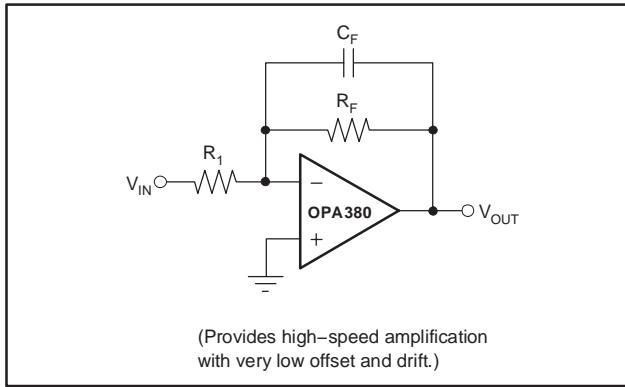


Figure 13. OPA380 Inverting Gain Configuration