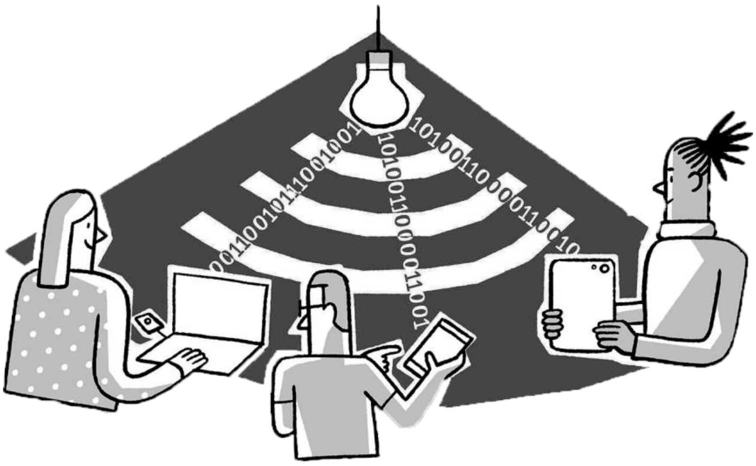
Science and Technology Council, IIT Kanpur

Visible Light Communication
LiFi v2.0



Final Report

Date: 18th Aug 20

Summer Camp 2020

Electronics Club

Contents

1 Acknowledgement **2**

2 Overview **3**

3 Why LiFi? **3**

4 Timeline **5**

5 Theory **6**

5.1 BPSK Modulation 7

5.2 BPSK Demodulation(Initial approach) 7

5.3 BPSK Demodulation(Current approach) 8

6 Hardware **11**

7 Result **13**

8 Improvements over last year **15**

8.1 Frequency channels 15

9 Future Developments **16**

10 Future Prospects **16**

11 TEAM **17**

1 Acknowledgement

We would like to thank Science and Technology Council, IIT Kanpur for providing us with golden opportunity of pursuing this project under Summer Camp 2020. We also take this opportunity to thank Electronics Club Coordinators Afzal Rao, Anshul Rai, Netravat Pendsey and Utkarsh Gupta for helping us throughout the course of this project by encouraging guidance and supervision.

2 Overview

. LiFi is based on Visual Light Communication (VLC) that using light emitting diodes (LEDs) tofully networked wireless system . LiFi enables the electronic device to connect to the internet withno wire. In order to make a communication line between node, a LiFi will need a transceiver to transmitand receive the data. This transceiver will have a modulation technique to make the LED enable to carrythe data using the light. The emergence of LiFi is to overcome the shortage of the current technology.We all know that right now WiFi is the most used technology to connect many devices to the internet.As time comes by, the use of internet based devices is increased. This increasing made the capacity of WiFi is reduced due the limitation of radio frequency resources.

3 Why LiFi?

Wireless data has become an essential commodity in our daily life. Wi-Fi is everywhere but sadly, we can't use it forever. Radio frequency, the technology that powers Wi-Fi, is running out of spectrum to support this digital revolution. More and more users use wireless connectivity with each passing year which brings us closer to a phenomenon called the spectrum crunch.

LiFi holds the potential to be more energy efficient and cheaper due to the nature of LED bulbs which are already efficient on their own. LiFi technology gives them another purpose, connectivity. This will save costs in homes and workplaces because it could do without electronic devices such as routers, modems, signal repeaters, wave amplifiers, and antennas. These devices have to be connected to power 24/7 in order for them to function. The fact that a lot of infrastructures probably has LED lights already in place, using LiFi would not be an extra cost.

With LiFi, every source of light can connect you to the internet. In the near future when the technology is already available to the general public, street lights, building lights, and transportation lighting can all communicate wirelessly and you can access the internet wherever you may be.

Why Modulating over a sine wave?

In the earlier versions of Li-Fi implementation, digital signal were being directly produced from Arduino's PWM pins; but digital signals theoretically use whole spectrum of frequency generating a lot of load on Arduino and also enclosing any development of multiple channels of frequencies for quicker data transfer.

Why Phase Shift Keying(PSK)?

Apart from PSK, Amplitude Shift Keying(ASK) and Frequency Shift Keying(FSK) were considered. ASK, although being robust and easiest to implement, offers no further prospect of faster data transfer as in PSK. FSK, on the other hand, is complicated and recovery of signal makes it prone to error.

PSK circuit components are less prone to failures and also offer further prospect of increasing the speed at least fourfold without major changes or addition to circuit.

4 Timeline

- **Dual Wire Transfer**

  To understand the concept of clock and data synchronicity, our first task was to implement transfer of bits using two wires. One wire transferred a serial stream of bits and the other one maintained a clock to notify an incoming bit at the receiver end. At the receiver end the bits were received in packets of 8 and were converted into their decimal equivalent.

- **Single Wire Transfer**

  Here we tried to transfer data bits using a single wire. For using Li-Fi we have to achieve a method for asynchronous transfer of data. Single wire Asynchronous Data Transfer required We encoded 1 as 110 and 0 as 100 at the transmitter Arduino. At the receiver's end our final data was decoded from the input stream by measuring the delay between a HIGH and corresponding LOW signal. For instance , if our data was bit 1 the delay was twice as much as that in bit 0. Here too, our data stream was decoded into packets of 8 bits and was subsequently converted to corresponding decimal.

- **Emulating Li-Fi**

  In order to emulate actual transfer through light and corresponding receiving through a Photo sensor, we used a CNY70 sensor(a reflective optical sensor, generally used for guiding paths) available in the Proteus library . Though this step was far from the actual conditions (lack of real world noise and many other factors), but still helped us understand the basic circuit requirements of light transfer. This was similar to the single wire transfer, except we used an amplifier and comparator setup at the receiver's end to recover the original digital wave, from the photo sensor output. Theoretically speed up to 8-8.5 kbps was achieved using this method.

- **Modulation of Data to Sine wave**

  Owing to the practical limitations, we decided to explore ways to make our data transfer effective. In order to increase the speed and bandwidth of transfer and to enable frequency channels in our setup, we tried to modulate our data through a carrier wave. It was decided to proceed with the Phase Shift Keying modulation (PSK) technique.

5 Theory

Various Steps involved in transmission of data are:

1. Sine wave and 180°(inverted) signal generation using Op Amp as inverter.

2. Encoding of output signal according to digital bits involved using Multiplexer.

3. Amplifying output signal and being sent to LED for medium transmission.

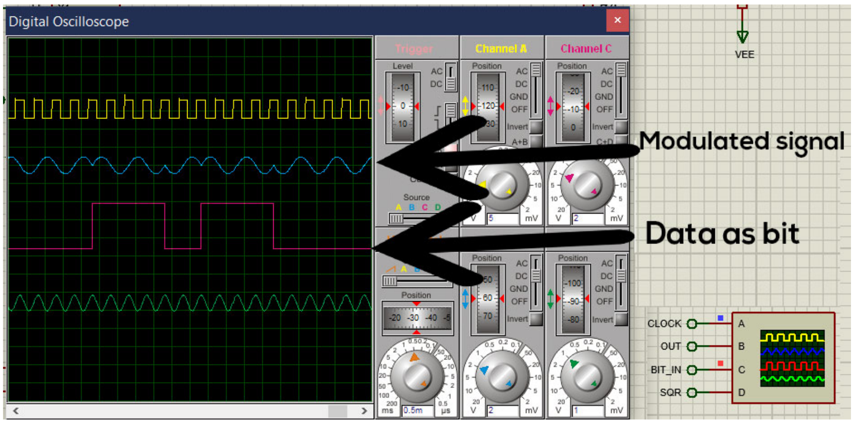


Figure 1: MODULATED SIGNAL

Demodulation of PSK signal is a tricky one. Various Steps involved in receiving data are:

1. Amplifying the input received and passing through filters for involved freq. Range.

2. Squaring the signal (producing the signal of twice the original frequency) and using it to demodulate the signal.

3. Checking the value of original signal w.r.t. clock generated by squared signal using interrupts of Arduino.

$$2sin^2x = 1 - cos2x$$

Clearing the DC offset and plotting w.r.t. original signal.

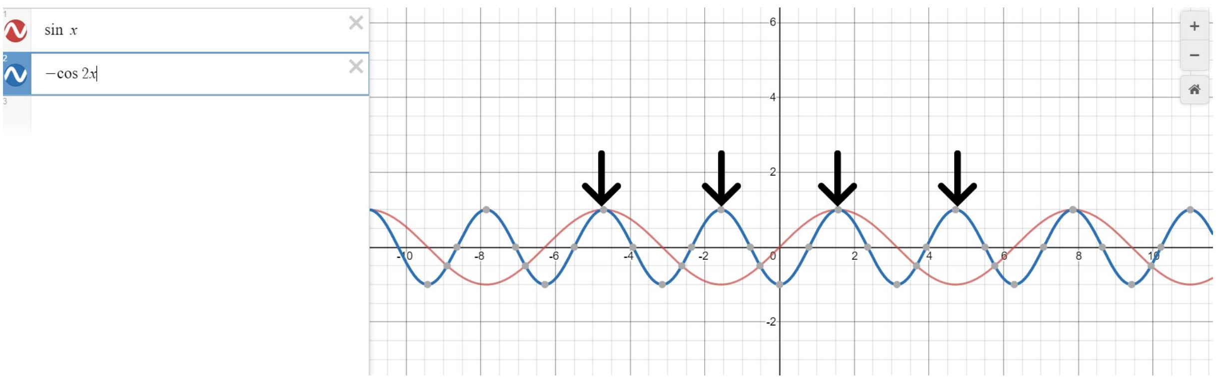


Figure 2: Squared and original signal are related

Crests of -cos2x align perfectly with either of peaks of sinx. We have used this to demodulate signal. Passing -cos2x signal through comparator to make signal digital and use it as a clock for checking actual voltage of original signal.

UNDERSTANDING PSK

PSK is the digital modulation technique in which the phase of the carrier signal is changed by varying the sine and cosine inputs at a particular time. PSK technique is widely used for wireless LAN s, biometric, contact less operations, along with RFID and Bluetooth communications.

In PSK, the constellation points chosen are usually positioned with uniform angular spacing around a circle. This gives maximum phase-separation between adjacent points and thus the best immunity to corruption.

5.1 BPSK Modulation

The modulation of BPSK is done using a balance modulator, which multiplies the two signals applied at the input. For a zero binary input, the phase will be 0°and for a high input, the phase reversal is of 180°.

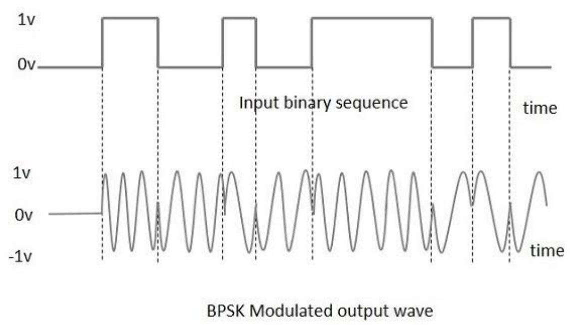


Figure 3: Modulated BPSK signal

5.2 BPSK Demodulation(Initial approach)

BPSK modulated signal is mathematically represented as $m(t)*sin(\omega t)$, where m(t) represents bit stream and $sin(\omega t)$ is the carrier signal. At the receiver end this part of incoming signal, after being amplified, was multiplied with the original carrier wave $sin(\omega t)$, which made the signal $m(t) * sin^2(\omega t)$.
The multiplication of signals was achieved using IC AD633J available in the Proteus library.

Recovery of Message Signal

Signal is reduced to

$$m(t) * sin^2(\omega t) = m(t) * (1 - cos(2\omega t))/2.$$

This signal is passed through a series of low pass filters to remove the oscillator frequencies $cos(2\omega t)$ and recover the original message signal. The output of the low pass filter is still in analog form. A digital stream was required for final processing. This was obtained by passing the analog signal through a comparator. Hence our data was recovered.

The major problem with this demodulation method was the generation of the carrier signal at the receiver's end. Under practical situation, generation of a signal with a same phase as that of the carrier shall render difficulties. Also we tried to recover our

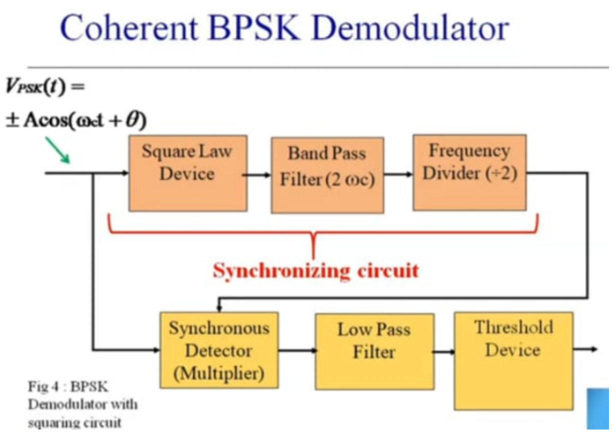


Figure 4: BPSK Demodulation circuit flow chart

carrier from the modulated signal by passing it through a square law device, removal of dc offset and a frequency divider using a flip-flop. However a combination of these circuits considerably altered the phase of the signal making it of no use.

5.3 BPSK Demodulation(Current approach)

We decided to recover the message using a clock. One part of the modulated signal was converted to digital form for the recovery of data using clock. Second part of the modulated signal was passed through a square law device which gave us the output $m2(t)*sin2(\omega t)$.This was passed through a high pass filter for the removal of DC offset and hence the signal recovered was $cos(2\omega t)$. This signal being twice in frequency of original signal, had crest aligned with crest or trough of original modulated signal. This signal after being converted to digital form was used as clock signal to measure original digitalized signal at specific intervals by triggering external interrupt of receiver Arduino. This method allowed us to recover our original bit stream successfully.

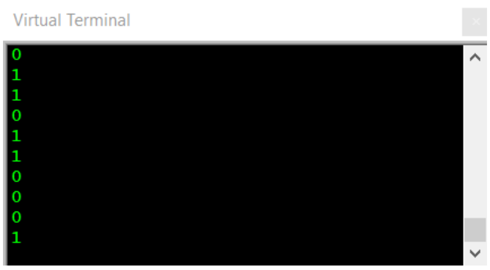


Figure 5: Received sequence of bits

Proposed QPSK Modulator

This works on the fact that QPSK signals can be generated by modulating sin and cos waves using BPSK and then adding the two.

There were two proposed modulators :

1. Uses an 4051 IC (8 channel analog multiplexer) and an Op-amp as an inverter.

2. This one uses 741 Op-amp, 74HC4016 single throw analog switch, and an inverter.

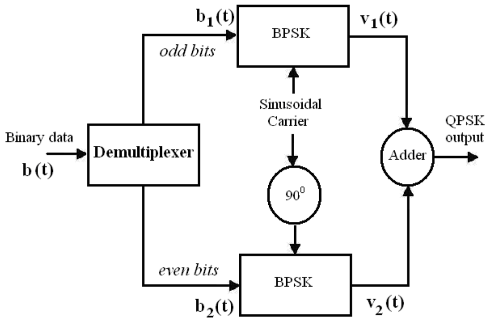


Figure 6: QPSK modulation using two BPSKs

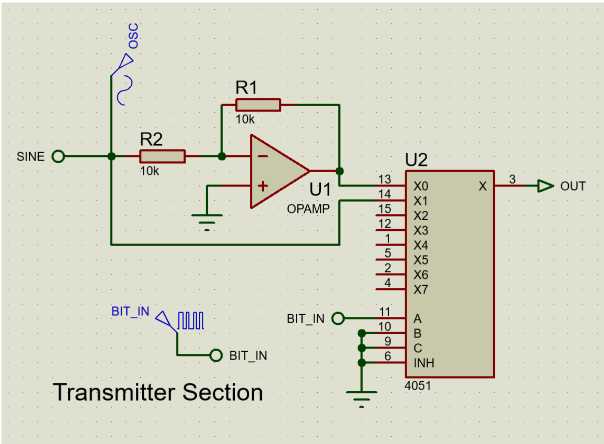


Figure 7: QPSK Modulation Circuit 1

Both work on the basic principle that the carrier is inverted when there is a change in voltage of data stream. Exploits the principle the inverted cos and sin wave are at a 180°phase.

Current QPSK Modulator

$$f(t) = A * cos(\omega t + (2m - 1) * \pi/4$$

$$wherem = 1for10$$

$$m = 2for00$$

$$m = 3for01$$

$$m = 4for11$$

Proposed QPSK Demodulator

The generated QPSK signal is transmitted via a medium i.e in practical uses through light.

The Demodulator has basically 4 parts

- **Carrier Recovery Circuit** - The QPSK signal is raised to it's 4th power, then is passed through a filter of 4fc then eventually divided by 4. BY doing this we get a cos wave of the same phase as the QPSK signal. It is then phase shifted through 90 to get the sine wave.

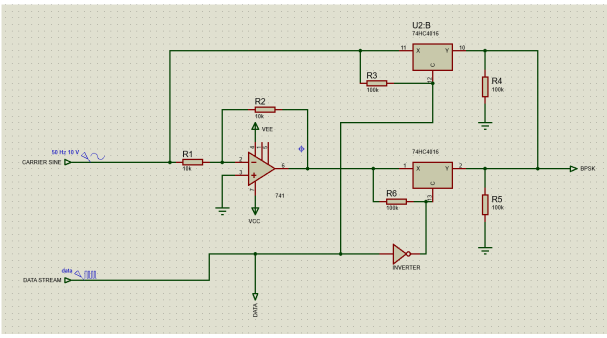


Figure 8: QPSK Modulation Circuit 2

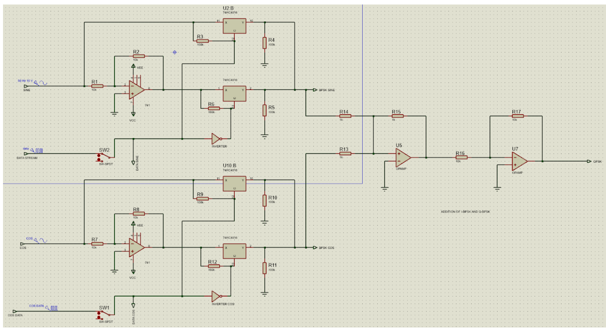


Figure 9: Current QPSK Modulation Circuit

- **A Multiplier** - The carrier is multiplied to the QPSK signal.
- **An Integrator** - The output from the multiplier is then integrated through 1 bit period.
- **A threshold device** - This is an Op-amp based comparator which makes a decision on each integrated bit based on a threshold. With this we get the odd and even bits from sine and cosine arms.

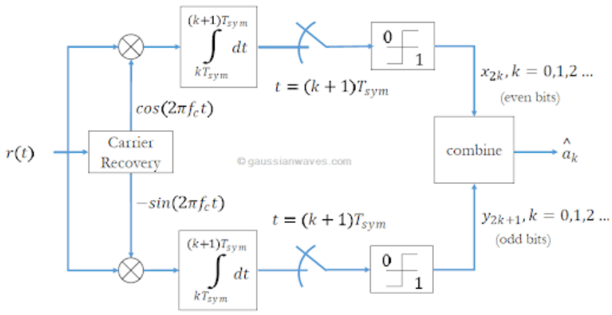


Figure 10: Proposed QPSK Demodulation Circuit

6 Hardware

- **Ardiuno**
  The Arduino is the brain of the setup. Encoding of data at the transmitter side , and subsequent decoding at the receiver end is done by Arduino. It generates a digital stream of encoded bits which is fed into the multiplexer. To decrease the load on Arduino and improve its performance tasks other than encoding and decoding are performed by IC's.

- **Multiplexer**
  74HC4051 multiplexer is available in the proteus library. It consists of 8 input channels numbered from 0 to 7. Additional three pins are available to feed a three bit binary number which decides the output of the multiplexer. For details refer BPSK modulation.

- **Multiplier**
  IC AD633J (analog multiplier) was used for the multiplication of analog signals during demodulation, in the below configuration :

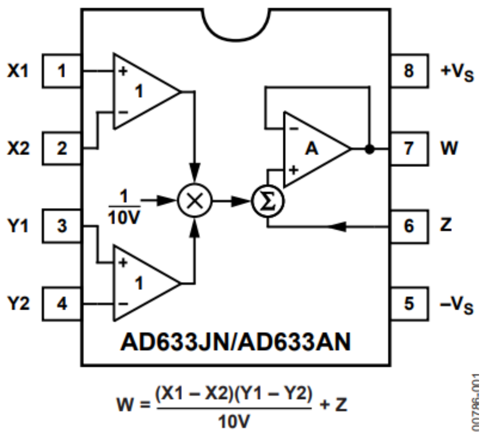
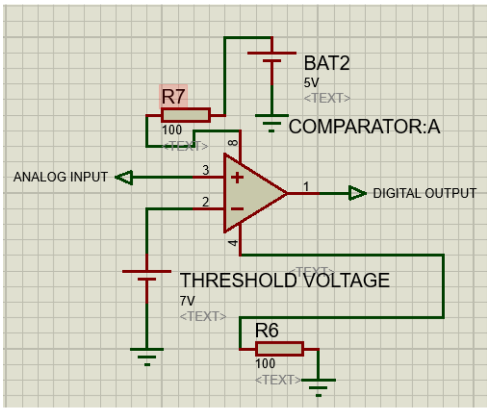


Figure 11: IC AD633J

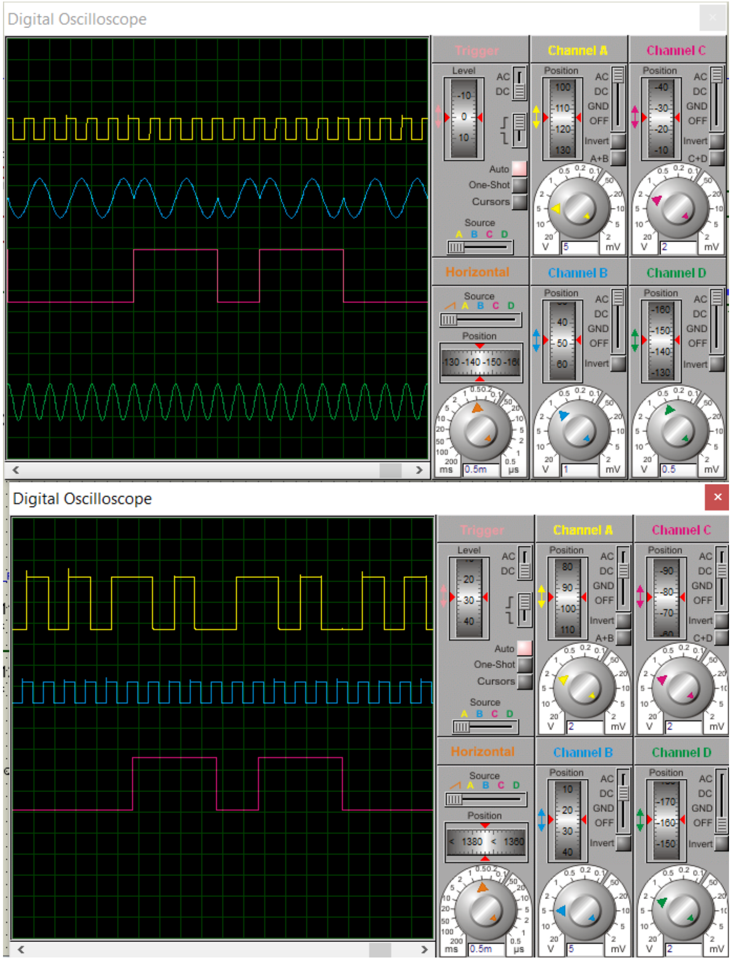
- **Comparator**
  The data received from the photo-sensor is in the form of an analog signal. This needs to be converted to a digital signal before feeding it into the arduino. However the inbuilt ADC of the arduino is too slow for our needs. Hence we need to use a

dedicated ADC. Since the data is encoded purely in terms of the frequency of the signal, we do not care about any intermediate values of the amplitude. We just need to distinguish between the HIGH and LOW cases. This is achieved by using a Op-Amp in a Comparator configuration.'



- **Amplifier**
  The strength of the signal received from the photo-sensor is considerably low. For further processing this incoming analog signal needs to be amplified using an OP-AMP in a non inverting configuration mode. Op Amp was used as non-inverting amplifier.

- **Inverter**
  For implementation of the PSK , we need to have an inverted signal of the original carrier wave. This is obtained using OP-AMP as an inverter.



7 Result

Hardware

Due to practical limitations, all of circuits were designed and simulated in the Proteus software environment. As this is a still developing field, none of them guided us into the circuitry, but only theories and test results using various modes. Most of them were centered around transmission of signals, rather than the circuits behind the scenes. Thus we have successfully designed and implemented our circuits from scratch.

Software

The codes of the transmitter and receiver side Arduino were original and developed by our team. We encoded each byte of our data into packets of eight at the transmitter side Arduino. This was successfully decoded and recovered in packets of 8 at the receiver side Arduino.

We had also intended to design a GUI software, which would enable a user to transmit data between two arduinos without knowing its technicalities. However due to lack of physical equipments , we realized it wouldn't be a great option.

Frequency Channels

We also designed a frequency channel in proteus which would enable the receiver , to switch to a particular frequency of the carrier waves. For this we added all the incoming modulated signals, using an OP-AMP and passed the resulting signal through a band pass filter whose threshold could be adjusted using a rheostat. This successfully allowed us to listen to a particular stream of data.

With the aforementioned developments we were able to design a working prototype which could transfer data bits from one arduino to other using light. The theoretical speed we had achieved was close to 8-8.5 kbps.

8 Improvements over last year

Modulation

For increasing the rate of data transfer and the bandwidth of our signal, we implemented modulation techniques in our circuit.Rather than square wave being sent and using larger frequency band,we used sine wave for transmission,which opens further possibility of multiple channels and parallel bit transfer. This brought the following advantages to our lifi setup :

- No signal mixing occurs.Frequency predefined for data transfer.
- Less attenuation due to high carrier frequencies. Carrier Frequency can be changed as per need.
- Adjustments in the bandwidth is allowed.
- Reception quality improves. Noises can now be filtered more easily listening only to specific frequency.

8.1 Frequency channels

Incorporating Keying bit to analog signal to out setup brings an added advantage of creating frequency channels. With different transmitters sending data at different frequencies, we can filter a frequency of desired choice using band pass filters. The threshold value of these filters can be adjusted using a knob which shall control the values of resistors and capacitors. This enables the users to tune to a particular data channel, much similar to a radio.

9 Future Developments

With the use of Raspberry Pi or microprocessors with multiple threads operating simultaneously speed of around 400kbps can be obtained by using algorithm used here.

QPSK could be used instead of BPSK to increase speed and various band of frequencies could be used for parallel bit transfer.

- Currently we are using separate generators for sine and cos waves; which when practically implemented could use only one generator to eliminate the possibility of phase difference in the two carriers.
- The data is currently fed separately into the stream with 4 bits each. A multiplexer can be used to split the bits and send them to the respective streams.
- There is no carrier recovery circuit owing to some problems, a robust carrier recovery circuit can be made to reduce ambiguity in phase of the carrier and QPSK signal during demodulation.
- In theory QPSK has a lower bandwidth than BPSK so a concept of multiple channels can be introduced.
- A duplex system can be made to make the data transmission even more reliable and robust.

10 Future Prospects

Safer and Faster

LiFi is a promising future avenue for data transfer as it transfers data at an previously unforeseen speed. Apart from potentially faster speed Li-Fi has multiple other benefits over Wi-Fi. Li-Fi offers a more secure network as light cannot pass through walls; this also minimizes the interference between devices. Li-Fi is also useful in electromagnetic sensitive areas such as in aircraft cabins, hospitals and nuclear power plants without producing electromagnetic interference.

Large Unused Radio Spectrum

Li-Fi technology uses visible light spectrum and can thus communicate data and unravel capacity which is 10,000 times greater than that accessible within the radio spectrum. The present visible light spectrum is abundant, free and unlicensed; this will help in mitigating the radio frequency spectrum crunch effect.

11 TEAM

Team Members

Munish Gupta
Saksham Mehra
Vatsal Chaudhary

Team Mentors

Afzal Rao
Anshul Rai
Netravat Pendsey
Utkarsh Gupta