Digital Data Transmission through Free Space Optical LASER

Author: Sambit Bikas Pal, *Indian Institute Of Science Education & Research, Kolkata*

Abstract:
In this demonstration I have used a semiconductor laser to transmit binary data from one computer to another, without using any physical media.

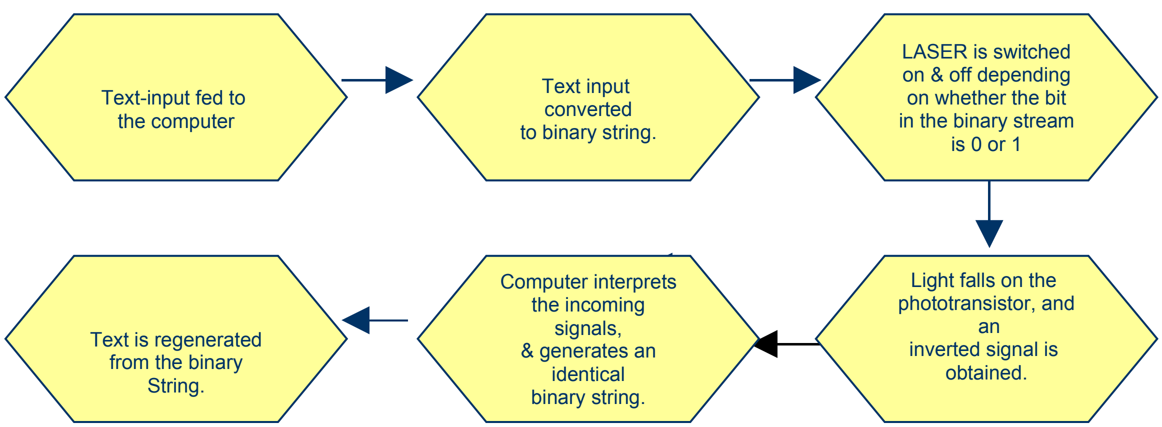
Keywords:
Laser, Telecommunication, Free space optics.

Introduction:

Free Space Optics (FSO) is a telecommunication technology that uses light propagating in free space to transmit data between two points. The optical links usually use infrared laser light. Distances up to the order of 10 km are possible, but the distance and data rate of connection is highly dependent on atmospheric conditions. In this demonstration I have used an ordinarily available semiconductor diode laser to construct the transmitter. For the receiver I have used a 2N5777 Phototransistor and a 2N2222 transistor to amplify the signals. For interfacing the circuits with the PC, "PHOENIX" interface was used.

Experiment:

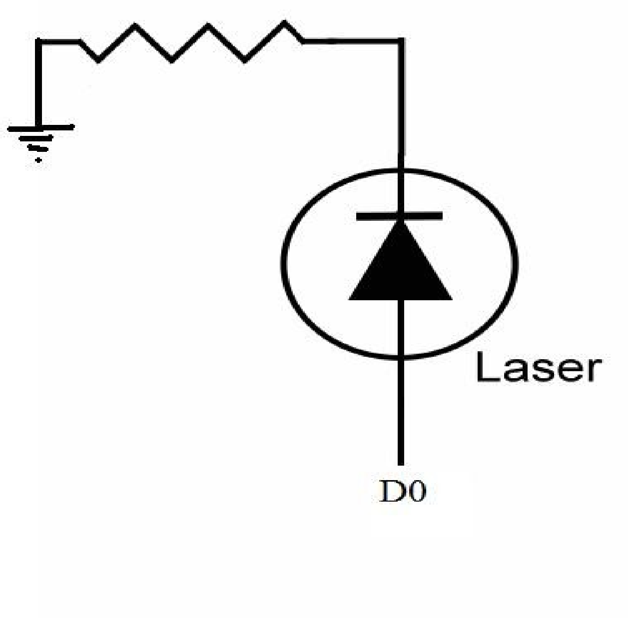
The basic procedure followed for the demonstration is:



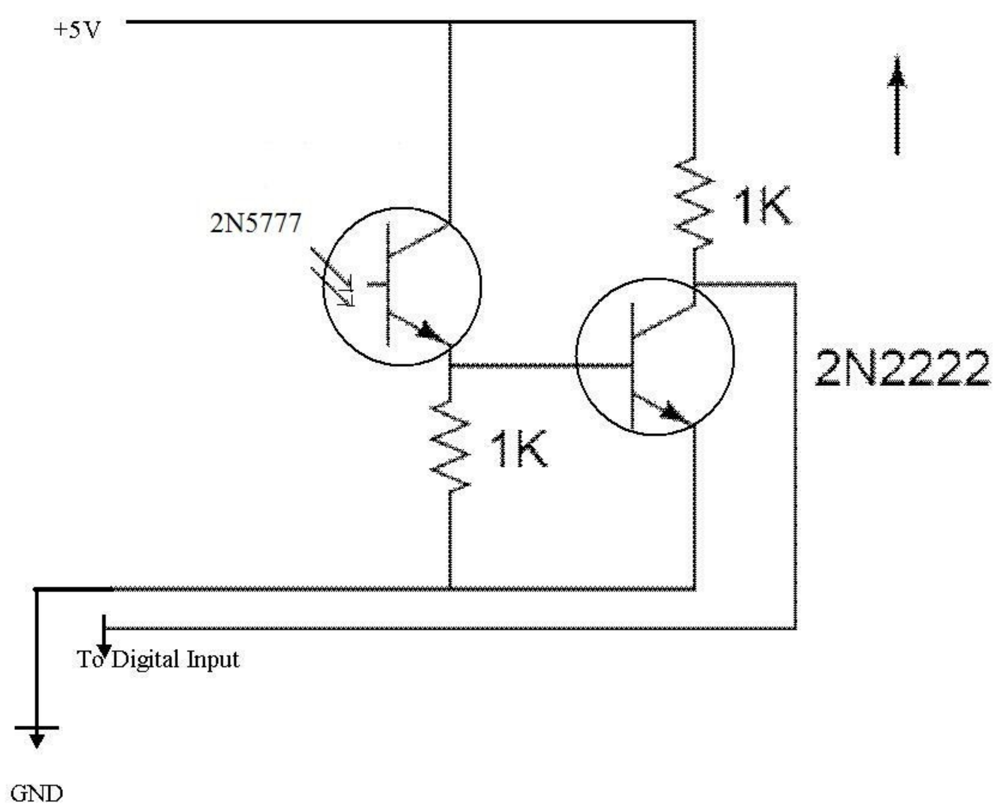
The transmitter consists of an semiconductor laser, powered by the digital output pin- D_0 of the "PHOENIX" box. The laser glows whenever the D_0 pin goes high.

The receiver has a 2N5777 phototransistor to receive the light signals. It is then amplified using a 2N2222 NPN general purpose amplifier, such that the output from the receiver is high when no light falls on the phototransistor and goes low when light falls.

The circuit diagram of the transmitter is given below:

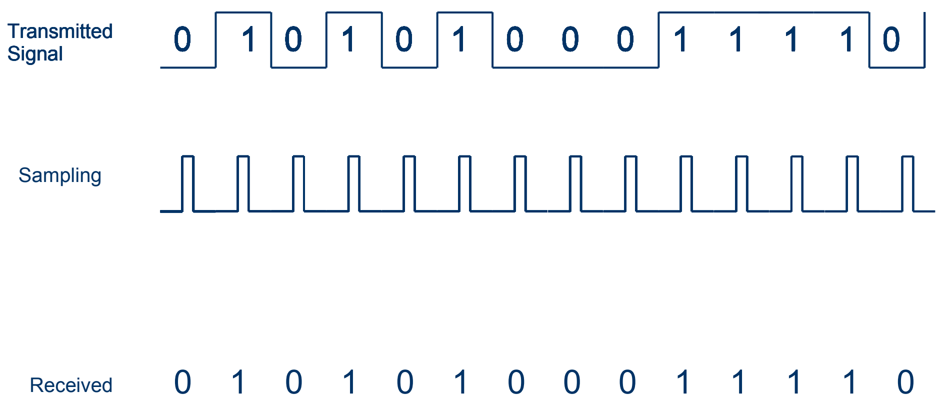


And this is the circuit diagram of the receiver:



Interfacing:
The interfacing with the PC was done using the "PHOENIX" Interface.The LASER was switched on and off using the digital output pin of the "PHOENIX".The incoming signal from the receiver was fed to the digital input pin of the "PHOENIX".

Theory:
The text string is first converted to its equivalent ASCII codes in binary form. Thus the text gets represented as a binary string. Now depending on whether the bit in the binary strung is 0 or 1, the laser is switched off & on.
Example:
Suppose the particular string 01010100011110 is to be transmitted. Then this will be sent to the transmitter accordingly. The receiver will look whether the value of each pulse is high or low and send the corresponding value to the digital input of the "PHOENIX" box.



The C codes used for the transmission and reception are:

TRANSMISSION:

```
/****Program to transmit a character string as binary data through a
LASER Channel********/

#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include "phlib.h"
void dectobin(char word[]);
int transmit(int byte[]);
void send(int *binword);
main(){
      printf("Enter a text string of max 10 letters\n");
      int i;
      char word[10];         //This array holds the text



      for (i=0;i<=9;i++){          //Initializes the word array
            word[i]=0;
      }

      gets(word);                    //Stores the entered word in the
char array "word"

      for (i=0;i<=9;i++){
            if (word[i]==NULL){
                  break;
            }
            printf("%d  %c  %d  \n", i,word[i],word[i]);

      }
      printf("..................\n");
      dectobin(word);
}


void dectobin(char word[]){
      int binword[80];        //This holds its binary equivalent.

      int i,byte,j,k=0;
      for(i=0;i<=79;i++){
            binword[i]=0;
      }
      for(i=0;i<=9;i++){
            for(j=k+7;j>=k;j=j-1){
                  binword[j]=word[i]%2;
                  word[i]=word[i]/2;
            }
            k=k+8;
      }
      for(i=0;i<=79;i++){
            byte=(i+1)%8;
```

```
            printf("%d\n",binword[i]);
            if(byte==0){
                  printf("......\n");
            }
      }
      send(binword);
}


void send(int binword[]){


      if(!open_phoenix())   //Checks whether PHOENIX drivers are loaded
        {
        printf("Phoenix driver open failed.\n");
        exit;
        }


      int byte[8];
      int i=0,j=0,k=1,l=0,t=100000;

      write_outputs(0);
      printf("Start receiver within five seconds\n");
      sleep(5);
      write_outputs(1);
      usleep (t);

      for(i=0;i<=79;i++){
            byte[l]=binword[i];
            if(l==7){
                  k=transmit(byte);
                  l=-1;
            }
            l=l+1;
      }
      printf("Transmission Complete\n");

}



int transmit(int byte[]){
      int j,t = 100000;       //t is the value of sleep time in usec

      for(j=0;j<=7;j++){
            write_outputs(byte[j]);
            usleep (t);
      }
      return 0;
      }
```

RECEPTION:

```
/*******Program to read binary stream from a laser channel through the
digital inputs of PHOENIX BOX***/

#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
#include "phlib.h"
#include<math.h>
int receive();
void bintodec(int byte[]);

main(){

      if(!open_phoenix())                              //Checks
whether PHOENIX drivers are loaded
        {
        printf("Phoenix driver open failed.\n");
        exit;
        }

        int a=0,b=0;
        printf("Enter 1 to start receiver\n");
        scanf("%d",&a);

        if(a==1){
            b=receive();
        }
        else{
            exit;
        }


        if(b==1){
            printf("Reception completed\n");
        }
        else{
            printf("Reception failed\n");
        }


}

int receive(){

      int binword[80];
      int byte[8];
      int i,j,t=100000;             //This t and transmit's t should be
same

      int laser=0;
      for(i=0;i>=-1;i++){           //This is to let the receiver
listen untill a high is encountered
            laser= !read_inputs();
```

```
		if(laser==1){
			break;
		}
		usleep(10000);
	}


	usleep(140000);			//keep the value slightly greater
than t(adjusted for midpoint of the pulse)
	i=0;
	for(j=0;j<=79;j++){

		byte[i]=!read_inputs();
		//printf("%d\n",read_inputs());
		if(i==7){
			i=-1;
			printf("..........\n");
			bintodec(byte);
		}
		i=i+1;
		usleep(t);

	}


	return 1;
}

void bintodec(int byte[]){
	int i,n=0;
	for(i=7;i>=0;i=i-1){
		n=n+(byte[i]*pow(2,(7-i)));
	}
	//printf("%d\n",n);
	printf("%c\n",n);
}
```

Discussions:

<u>Applications:</u>

- Though this is just a small scale demonstration, FSO is a very promising point to point communication technology.
- ***Typically scenarios for use are:***
- LAN-to-LAN connections in a city. *example, Metropolitan area network*
- To cross a road or other physical barriers.
- Temporary network installation.
- As an alternative or upgrade addon to existing wireless technologies.
- As a safety addon for important fiber connections (redundancy)
- For communications between spacecraft, including elements of a satellite constellation.
- For interstellar communication.
- The lightbeam can be very narrow, which makes FSO hard to intercept, improving security.

<u>Advantages:</u>

- It’s chief advantages over RF communication and Fiber Optics are:
- Quick link setup.
- No regulatory issues.
- High transmission security.
- High bit rates.
- Low bit error rate.
- No interference.
- Compared to a microwave link, the advantages are that it can support higher bit rates (under good conditions), that its dispersion is lower.
- No need of expensive Optical Fibres.

<u>Limitations:</u>

- For terrestrial applications, the principle limiting factors are:
- Beam dispersion
- Atmospheric absorption
- Rain / fog /snow (attenuation)
- Background light
- Shadowing
- Pointing/alignment stability in wind
- Pollution / smog .
- These factors cause an attenuated receiver signal and lead to higher bit error rates. Atmospheric and fog attenuation, which are exponential in nature, limit practical range of FSO devices to few kilometres.

References:

- Harvard Broadband Communications Laboratory. (http://www.seas.harvard.edu/hbbcl/fsoc.html)
- Wikipedia.
- PHOENIX C Library Manual by Dr. Ajit Kumar.

Acknowledgements:

I would like to thank my teachers Dr. Abhijit Poddar and Dr. Subhamoy Maitra for their invaluable guidance.

I would also like to thank Mr. Subhash Malo for his constant help.