REPORT ON DIGITAL COMMUCATION SYSTEM USING MATLAB

BY TANTRESHWAR KUMAR (171220050), ECE 3rd Year

Submitted to DR MANISHA BHARTI

Brief introduction

>> Digital communication is the process in which data is transferred in a digital format which is very immune to noise and the circuitry for transmitter and receiver can be easily implemented. That’s why it is preferred over analog communication

The digital communication can be divided in some basic steps:

Transmitter:

- **Sampling of analog signal**: It is done because analog signal is continuous signal. So, the value does not change much if there is less time gap. Therefore, instead of transmitting for each instant it is better to transmit samples.
- **Quantisation**: Signal amplitude is broken in levels. The amplitude of each sample is assigned one level. This is known as quantisation.
- **Encoder**: The encoder converts each quantised level in binary number. It is these bits that are being transferred over the channel.

Receiver:

- **Decoder**: The bits received is decoded and the quantised level is detected.
- **Digital to Analog conversion**: From quantised level the analog value belonging to that quantised level is assigned.

Working

The analog data is first sampled. It is then quantised and then each sample is assigned corresponding quantised level on basis of its amplitude. The level is represented in form of binary numbers which can be of n-bits. Then each bit is transferred over internet to the Wi-Fi module (IoT concept). It is done through http request.

The receiver consists of Wi-Fi module. It stores all the received bits and then it groups them in n-bits (no. of bits used by encoder). Then it calculates the corresponding amplitude for that quantised level. That's how each sample is plotted.

Code:

//Transmitter Code:

```
clc;
clear all;
import matlab.net.*
import matlab.net.http.*
r = RequestMessage;
uri=URI("https://cloud.boltiot.com/remote/e90ab9b0-1def-4da5-90cf-
2c2c9165fdf9/digitalWrite?pin=2&state=HIGH&deviceName=BOLT1115146")
resp = send(r,uri)
str=1
t=0:0.5:2*pi
y=sin(t)
j=length(t) %No. of samples
stem(t,y)
%Below loop checks for pin 3.Note that pin 2 and 3 are shorted.If pin 3 is
low means reciever is on or handshake is successful.
while str==1
    r = RequestMessage;
    uri=URI("https://cloud.boltiot.com/remote/e90ab9b0-1def-4da5-90cf-
2c2c9165fdf9/digitalRead?pin=3&deviceName=BOLT1115146")
    resp = send(r,uri);
    str=str2num(resp.Body.Data.value)
    disp("Waiting for reciever!!")
    pause(3)
end
%In below loop samples are now quantised and send
for p=1:j
m=quantiser(y(p),2) %Quantises the samples
digitalTrans(m)     %Transmits the quantised sample bitwise
end
r = RequestMessage;

%Finally since transmission has finished pin 0 is reset.
uri=URI("https://cloud.boltiot.com/remote/e90ab9b0-1def-4da5-90cf-
2c2c9165fdf9/digitalWrite?pin=0&state=LOW&deviceName=BOLT1115146")
resp = send(r,uri);
```

Parts of Transmitter Code:

Following two functions are the essential functions of transmitter part. The Quantiser part quantises the signal (converts it to binary number). While the digital transmitter transmits the bits.

//Quantiser:

```
function [m]=quantiser(x,pk)
%5 bit therefore number of level is 2^5 ie loop for 32 times
    for n=0:31
        %if a sample value x is in b/w n and n+1 such that it can be of form
below.Then conversion for n is done
        if x>(-pk/2+n*pk/32)&& x<(-pk/2+(n+1)*pk/32)
            m=decimalToBinaryVector(n,5)
        elseif x==-pk/2
            m=[0 0 0 0 0]
        elseif x==pk/2
            m=[1 1 1 1 1]
        %if a sample value x is having n such that it can be of form
below.Then direct conversion is done
        elseif x==(-pk/2+n*pk/32)
            m=decimalToBinaryVector(n,5)
        end
     end
```

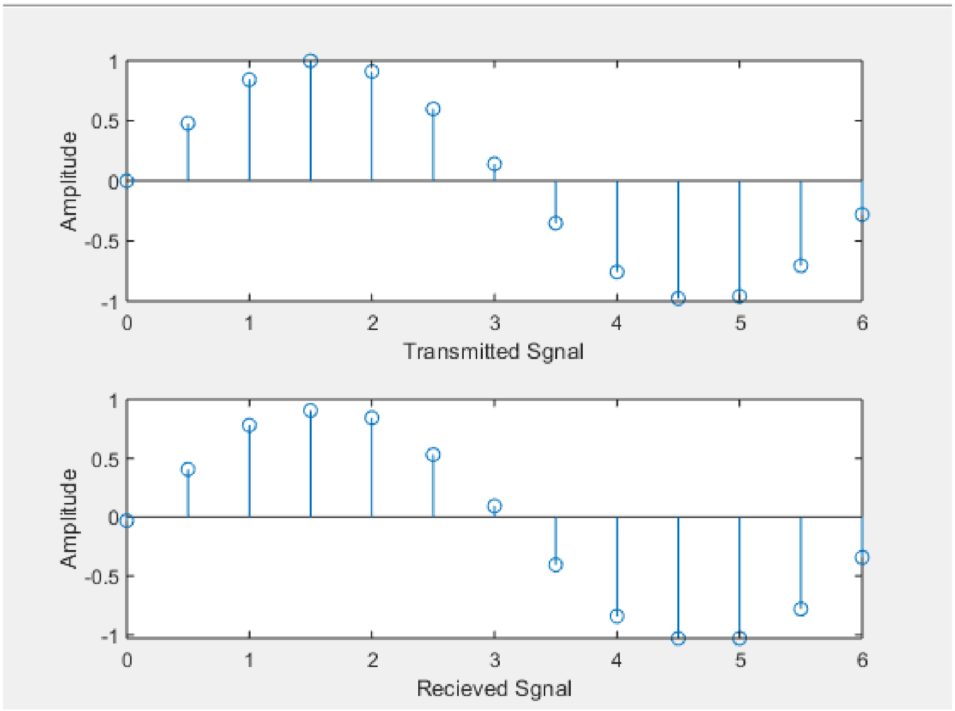
//Digital Transmitter (HTTP request sender):

```
function []=digitalTrans(m)
    import matlab.net.*
    import matlab.net.http.*
    %Below loop converts the quantised data bitwise to LOW or HIGH string
    for i=1:5
        if m(i)==0
            str="LOW";
        else
            str="HIGH";
        end
        disp(str)
        r = RequestMessage;
        uri =URI('https://cloud.boltiot.com/remote/e90ab9b0-1def-4da5-90cf-
2c2c9165fdf9/digitalWrite?pin=0&state='+str+'&deviceName=BOLT1115146');
        resp = send(r,uri);
        pause(10)
    end
```

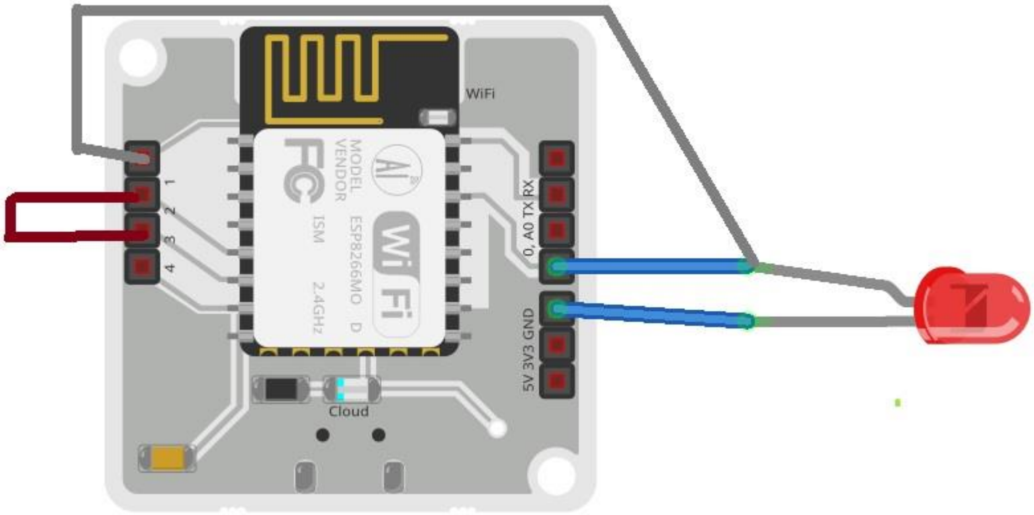
Receiver Code:

```
import matlab.net.*
import matlab.net.http.*
t=0:0.5:2*pi;
j=length(t); %No. of samples
r = RequestMessage;
%Pin 2 seted by transmitter is set to LOW value.
uri=URI("https://cloud.boltiot.com/remote/e90ab9b0-1def-4da5-90cf-
2c2c9165fdf9/digitalWrite?pin=2&state=LOW&deviceName=BOLT1115146");
resp = send(r,uri);
m=[];
pause(3)        %Waits for transmitter handshake check loop to finish
for i=1:(j*5)   %Since 5 bits are there total number of bits in transmission
is 5*no. of samples
    r = RequestMessage;
    uri=URI("https://cloud.boltiot.com/remote/e90ab9b0-1def-4da5-90cf-
2c2c9165fdf9/digitalRead?pin=1&deviceName=BOLT1115146");
    resp = send(r,uri);
    str=resp.Body.Data.value;
    m=[str2num(str) m]; %Stores all reciever bits in m matrix
    pause(10);
end
disp("Roger That!!");
ch=input('Copy That?Yes=1:No=0');
if ch==1
    an=[];
    %Below loop divides the whole bit matrix (m) in 5 bits units. ie.
quanised bit sequence is retrieved and then dequantised
    for p=1:j
        o=[]
        for t=(5*p-4):5*p
            o=[m(t) o];
        end
    n=binaryVectorToDecimal(o)
    an=[-2/2+n*2/32-2/(2*32) an]; %Dequantises the recieved signal ref
Quantiser function
    end
t=0:0.5:2*pi
stem(t,an);
end
```

Result:



Receiver Schematic:



Implementation: