EEE 301: Baseband Digital Signal Transceiver

Name: Kai-Yu Lu
ID: 1614649

1. Introduction

Baseband signals commonly refer to the original signals which have been transmitted without modulation. In the field of telecommunication, most baseband signals are converted to high frequency signals so as to be transmitted over long distances. However, the baseband signals could be transmitted in a short range in real life. For instance, the signals transmitted inside computers are all baseband signals because the attenuation is small and the information in baseband signals will not vary. Similarly, Ethernet is a typical application using baseband signals to transmit data because it does not require signal modulation.

This assignment requires students to use MATLAB software to simulate the process of a baseband transceiver for binary digital signals. During this assignment, there are four types pf pulse shaping, which are rectangular in unipolar and bipolar and root raised cosine (RRC) in unipolar and bipolar. According to these types of pulse shaping, corresponding optimum receivers should be designed, such as their matched filters. Consequently, verifications of bit error rate (BER) performance based on relevant types of pulse shaping will be explored. Additionally, the baseband signal in time domain and the power spectrum density for each case will be provided. Finally, the conclusion of the assignment will be presented in the end.

2. Schematic block Diagram of the Baseband Digital Transceiver

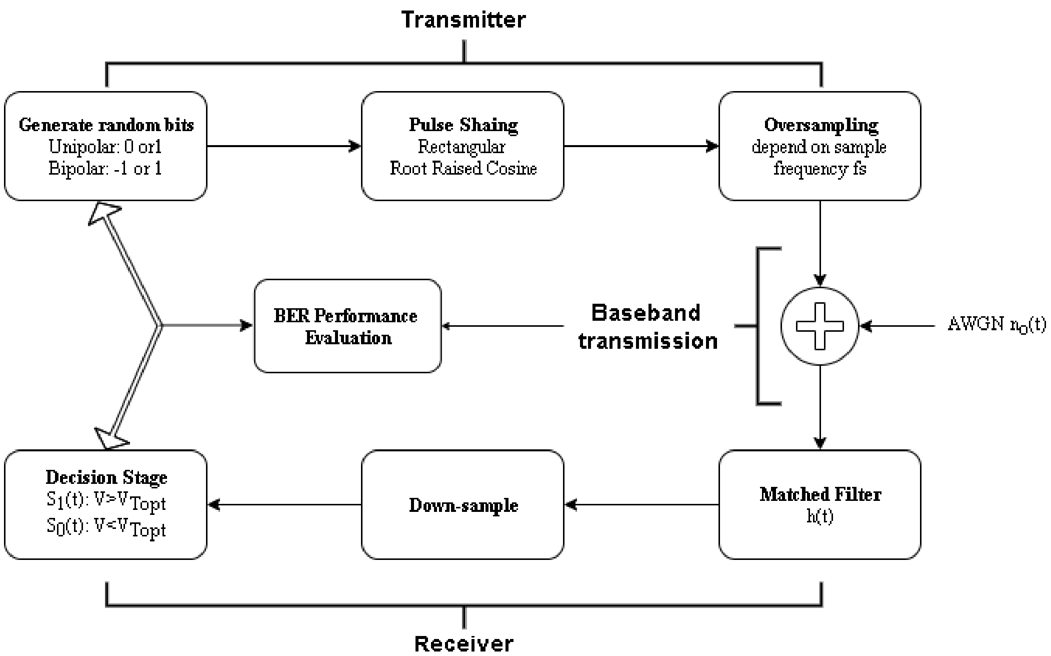

Figure 1: Schematic block Diagram of the Baseband Digital Transceiver

3. Math modeling

- **Energy of bit 1 and 0:** For bit 1: $E_1 = \int_0^T S_1{}^2(t)dt$. For bit 0: $E_0 = \int_0^T S_0{}^2(t)dt$.

- **Average energy per bit:** $E_b = E_0P(0) + E_1P(1)$, where $E_1 = \int_0^T S_1{}^2(t)dt$, $E_0 = \int_0^T S_0{}^2(t)dt$.

  $P(1) = P(0) = 0.5$, if the signals of 1 and 0 are equally-likely.

- **Optical threshold voltage:** $V_{Topt} = \frac{E_1 - E_0}{2}$. If $V > V_{Topt}$, "1" is sent. If $V < V_{Topt}$, "0" is sent.
- **Optimum (Matched Filter) Receiver:** $\boldsymbol{h_{opt}(t) = s_1(T-t) - s_0(T-t)}$
- **Average Power of AWGN:** $P_{AWGN} = \frac{N_0}{2}$.
- **Minimum BER in theory:** $P_{e,min} = Q[\sqrt{\frac{E_g}{2N_0}}]$,

  where $E_g = \int_0^T (S_1(t) - S_0(t))^2 = E_0 + E_1 - 2\rho\sqrt{E_0 E_1}$. In unipolar, ρ is 0. In bipolar ρ is -1.

  Therefore, unipolar: $P_{e,min} = Q\left[\sqrt{\frac{E_g}{2N_0}}\right] = Q\left[\sqrt{\frac{E_b}{N_0}}\right]$; bipolar: $P_{e,min} = Q\left[\sqrt{\frac{E_g}{2N_0}}\right] = Q\left[\sqrt{\frac{2E_b}{N_0}}\right]$
- **Root Raised Cosine Filter:**

$$H(f) = \begin{cases} \sqrt{T} & (0 \le |f| \le \frac{1-\beta}{2T}) \\ \sqrt{\frac{T}{2}\left\{1 + \cos\left[\frac{\pi T}{\beta}\left(|f| - \frac{1-\beta}{2T}\right)\right]\right\}} & (\frac{1-\beta}{2T} \le |f| \le \frac{1+\beta}{2T}) \\ 0 & (|f| \ge \frac{1+\beta}{2T}) \end{cases}$$

Figure 2 is the spectrum of RRC and its frequency response with varying rolloff β.

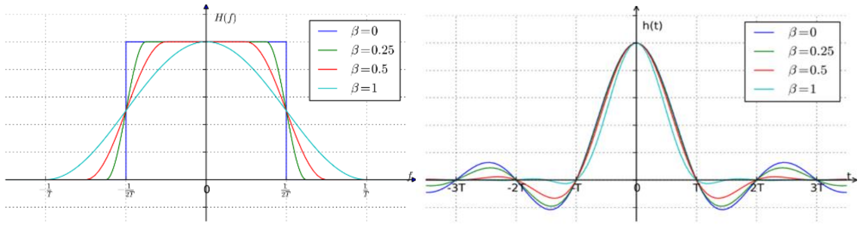

Figure 2: Spectrum of RRC and its frequency response with varying rolloff β.

- **Discussion between RRC pulse and rectangular pulse:**

  In baseband communication, the symbols are converted into the waveforms which are compatible with the characteristics of the channel during transmission. During this assignment, the digital baseband signals are required to transmit in pulse shaped of rectangular and RRC. The rectangular pulse shape can be implemented easily with minimal memory requirements. However, the sharp edges in the time domain will present large bandwidth in the frequency domain. The Fourier transform of the rectangular pulse is a sinc function and it is shown in Figure 3.

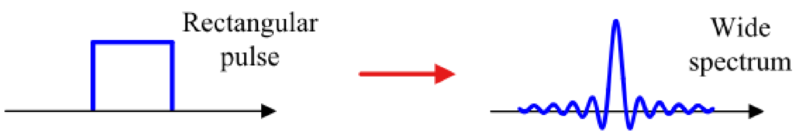

Figure 3: The Fourier transform of the rectangular pulse.

From Figure 3, it could be observed that the bandwidth of a rectangular pulse is infinite. As a result, the rectangular pulse becomes unpractical in real applications because it will cause Inter symbol Interference (ISI), which means the signals shaped in rectangular pulses cannot pass through the device which has finite bandwidth. Otherwise, the previous sampling points will sample the next symbols, which is the phenomenon of ISI. Although Ideal Low Pass Filter could limit the bandwidth of the rectangular pulse, it requires extreme precise synchronization which is not practical. On the contrary, root raised cosine (RRC) is quite practical to implement the pulse shaping. From Figure 2, it indicated that the bandwidth of RRC becomes finite both in time domain and in frequency domain. Additionally, the bandwidth of the RRC could be adjusted by varying the roll-off β. Therefore, RRC filter is commonly used in pulse shaping in real application.

4. Parameter Setting

Parameter Setting of Rectangular pulse:

- **Number of data bits:** $N = 8192$.
- **Bit duration:** $T_b = 1s$.
- **Sampling rate:** $fs = 8\ Hz$.
- **Amplitude:** $A = 1$
- **Noise in dB:** [-39, 26]; increasement: 1.
- **Number of FFT:** 2^{15}

Parameter Setting of Rectangular pulse:

- **Number of data bits:** $N = 8192$.
- **Bit duration:** $T_b = 1s$.
- **Sampling rate:** $fs = 8\ Hz$.
- **Amplitude:** $A = 1$
- **Noise in dB:** [-39, 26]; increasement: 1.
- **Number of FFT:** 2^{15}
- **Roll-off:** $\beta = 1$.
- **Filter span**: span = 8.
- **Samples per symbol**: sps = 8.

5. Result

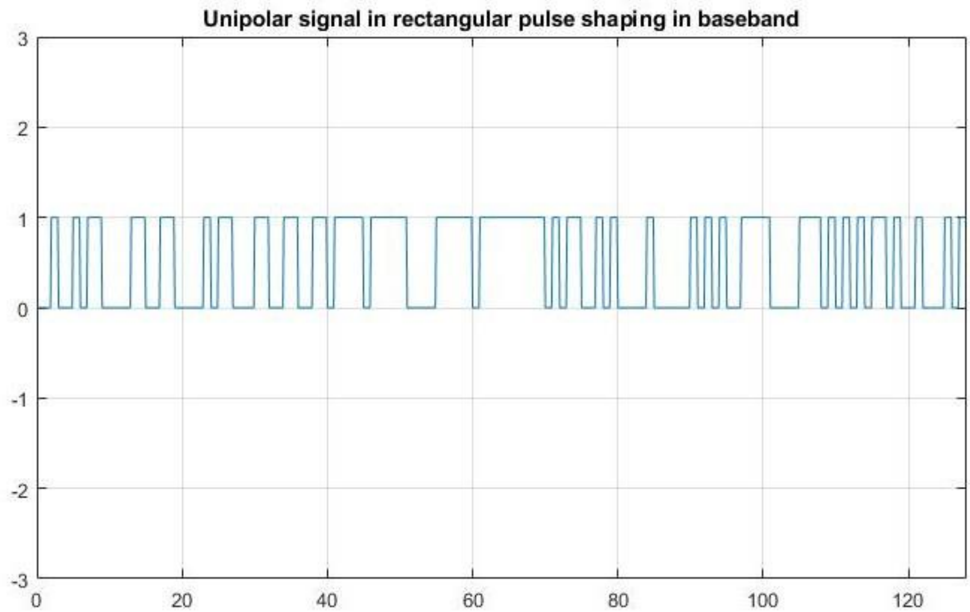

Figure 4: Baseband original unipolar signal in time domain (rectangular).

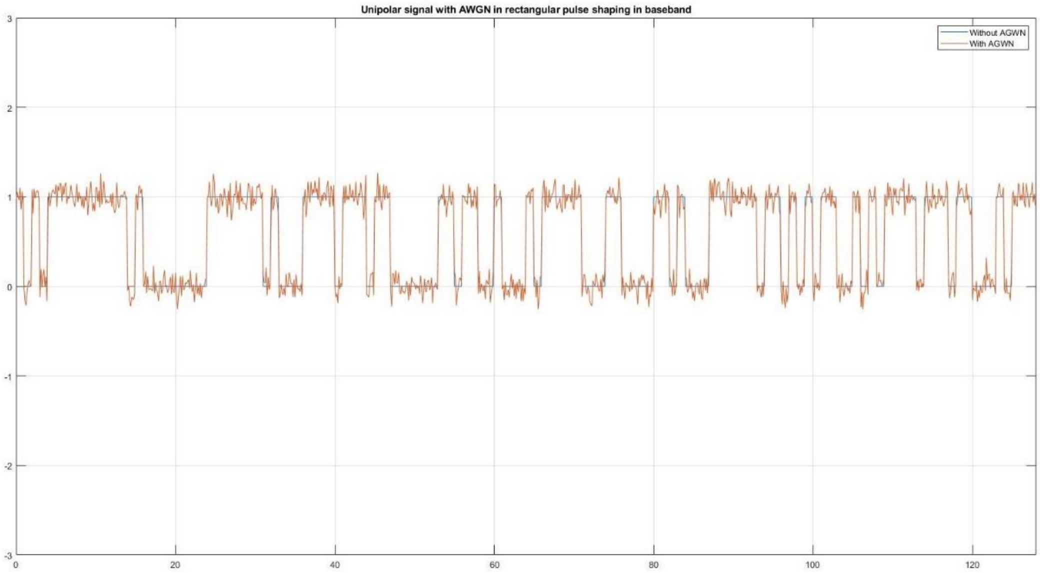

Figure 5: Baseband original unipolar signal with AGWN in time domain (rectangular).

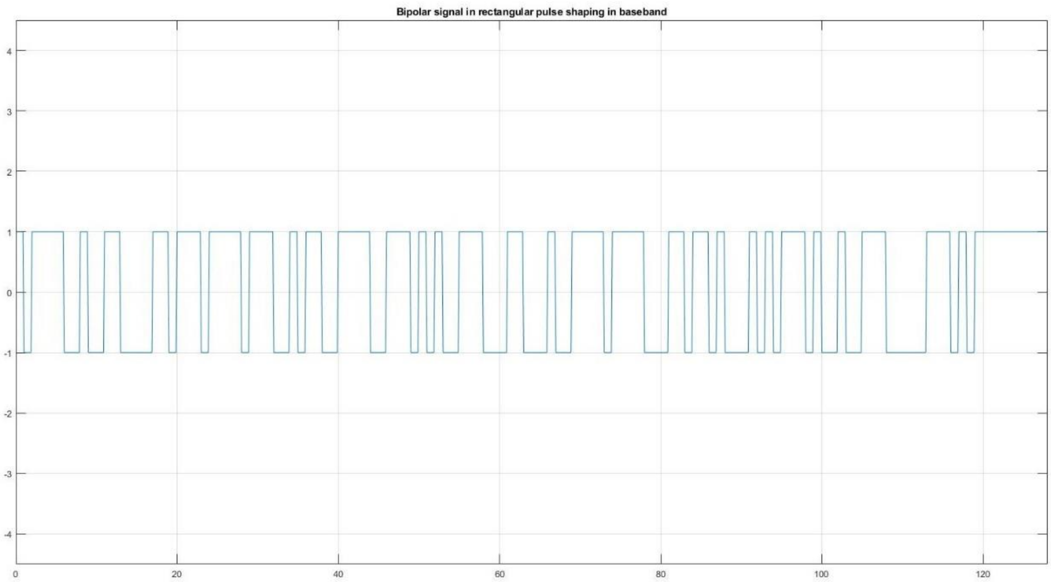

Figure 6: Baseband original bipolar signal in time domain (rectangular).

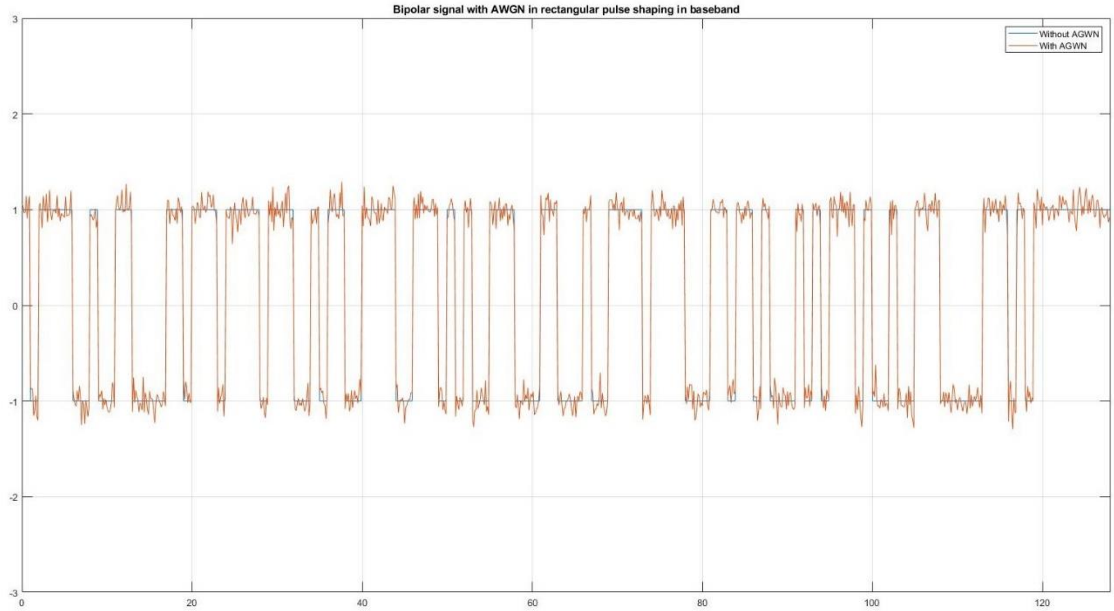

Figure 7: Baseband original bipolar signal with AGWN in time domain (rectangular).

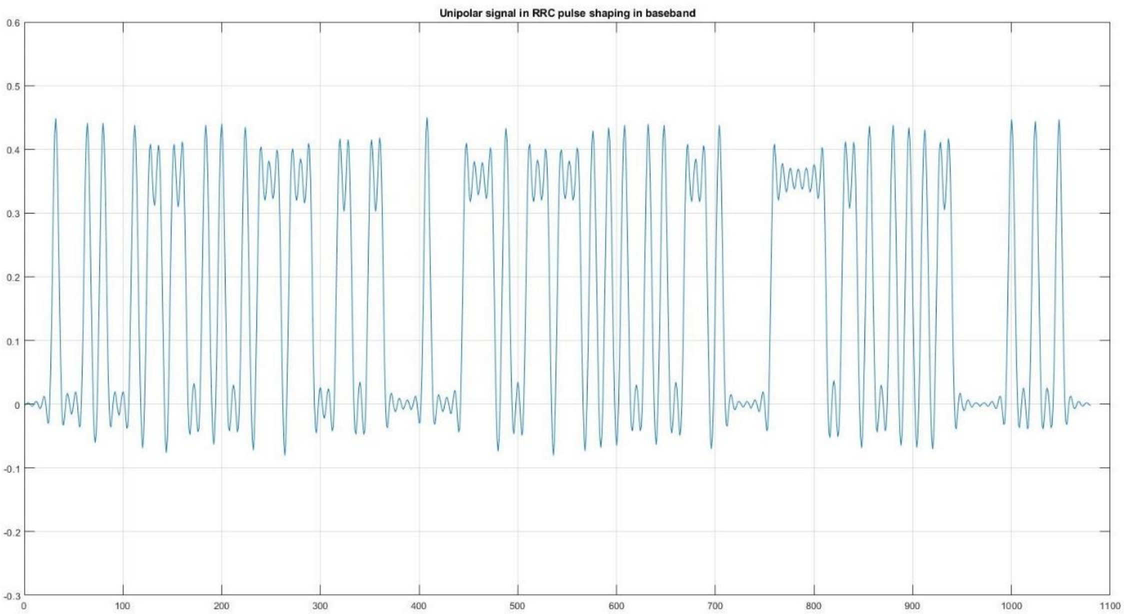

Figure 8: Baseband original unipolar signal in time domain (RRC).

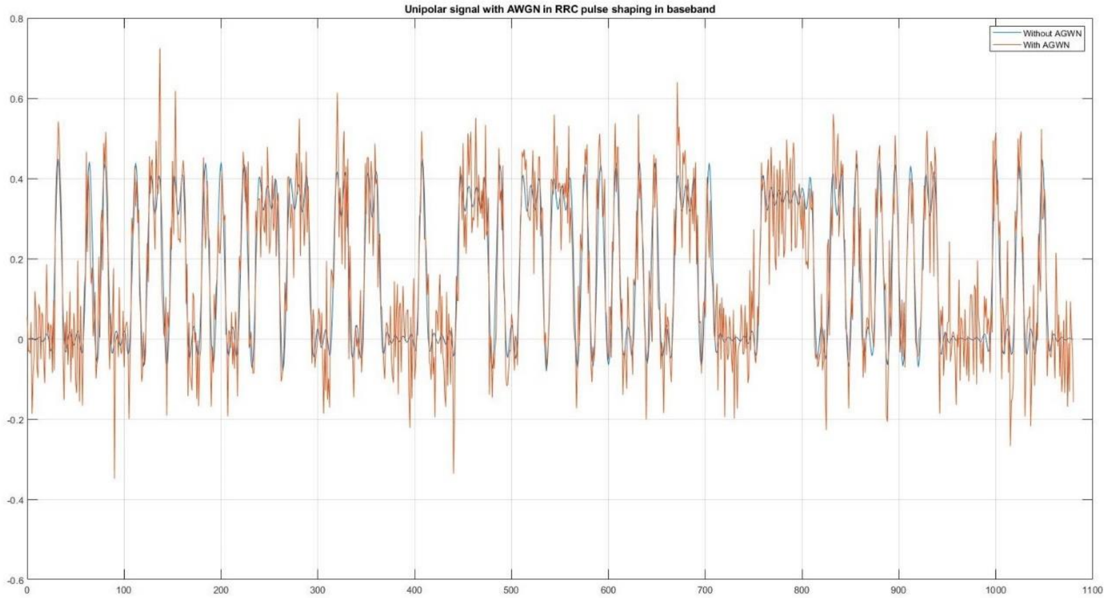

Figure 9: Baseband original unipolar signal with AWGN in time domain (RRC).

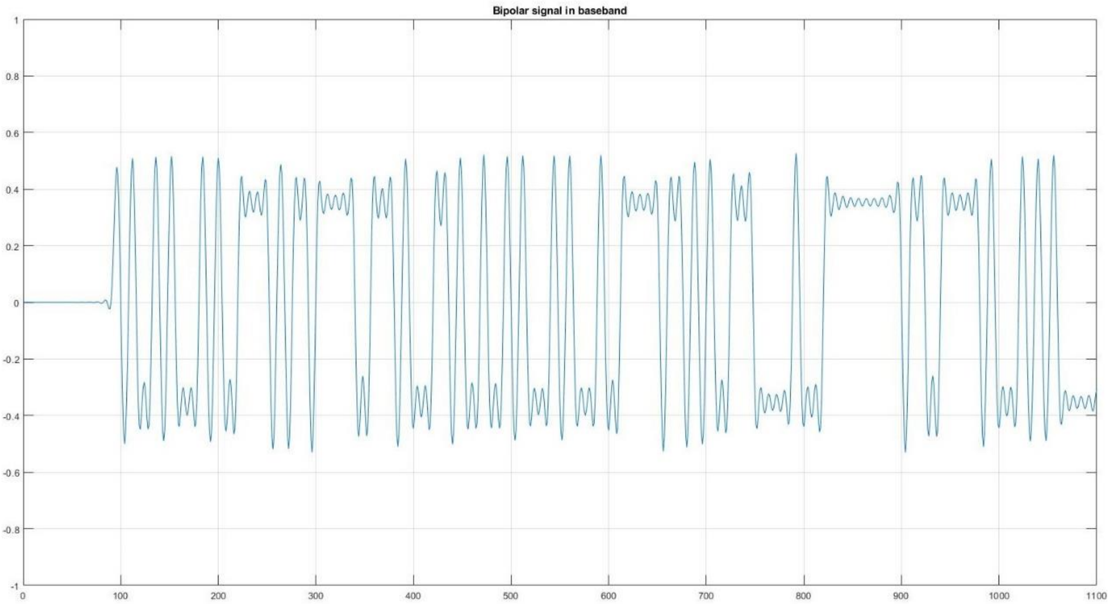

Figure 10: Baseband original bipolar signal in time domain (RRC).

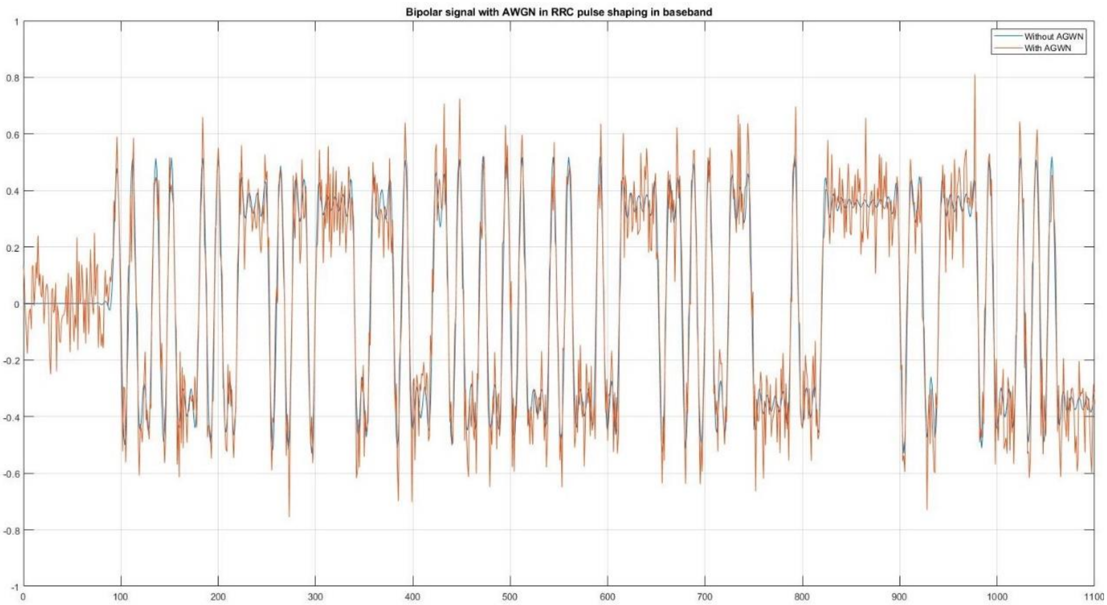

Figure 11: Baseband original bipolar signal in time domain (RRC).

Power spectrum density of the baseband signal:

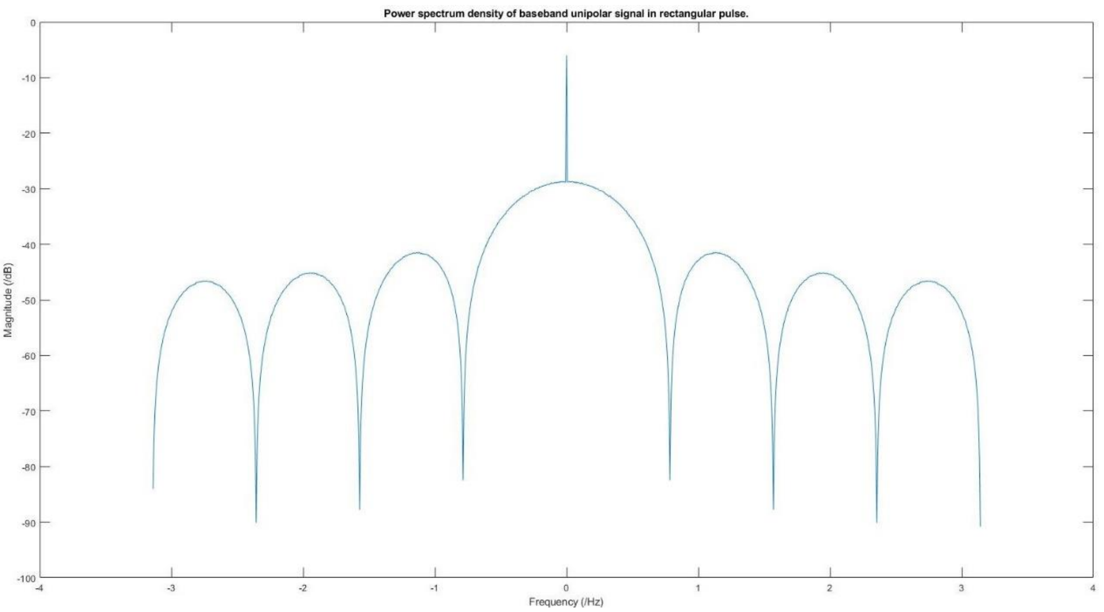

Figure 12: Power spectrum density of baseband unipolar signal in rectangular pulse.

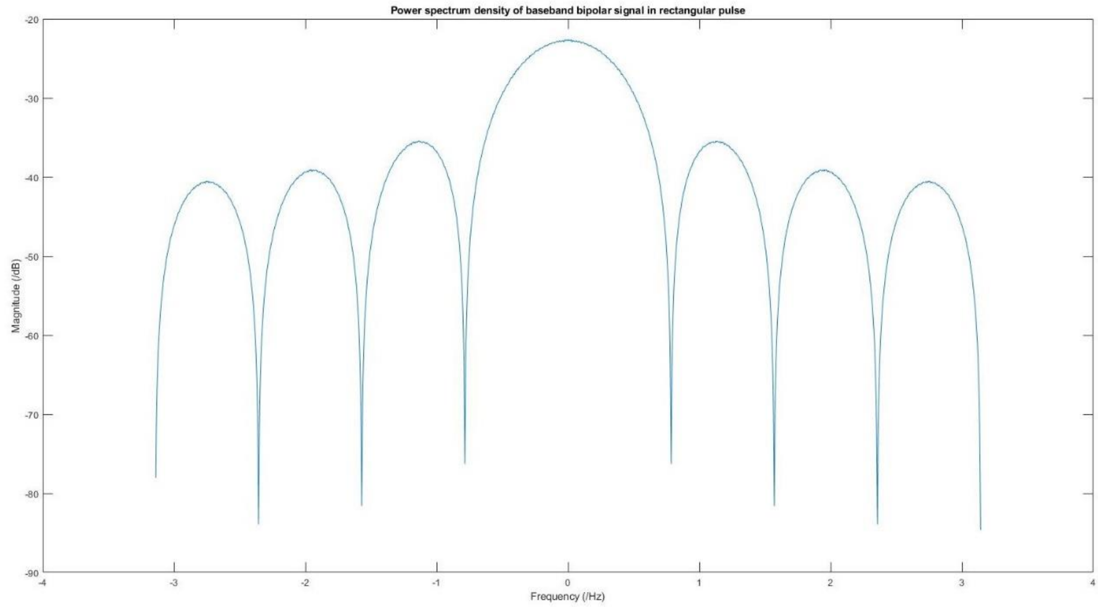

Figure 13: Power spectrum density of baseband bipolar signal in rectangular pulse.

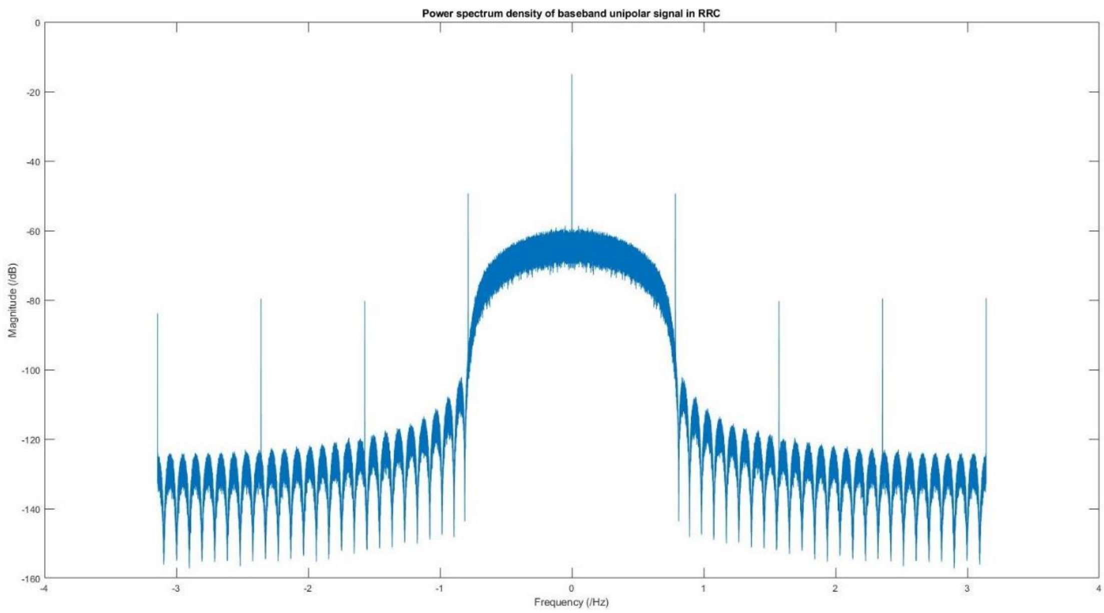

Figure 14: Power spectrum density of baseband Unipolar signal in RRC.

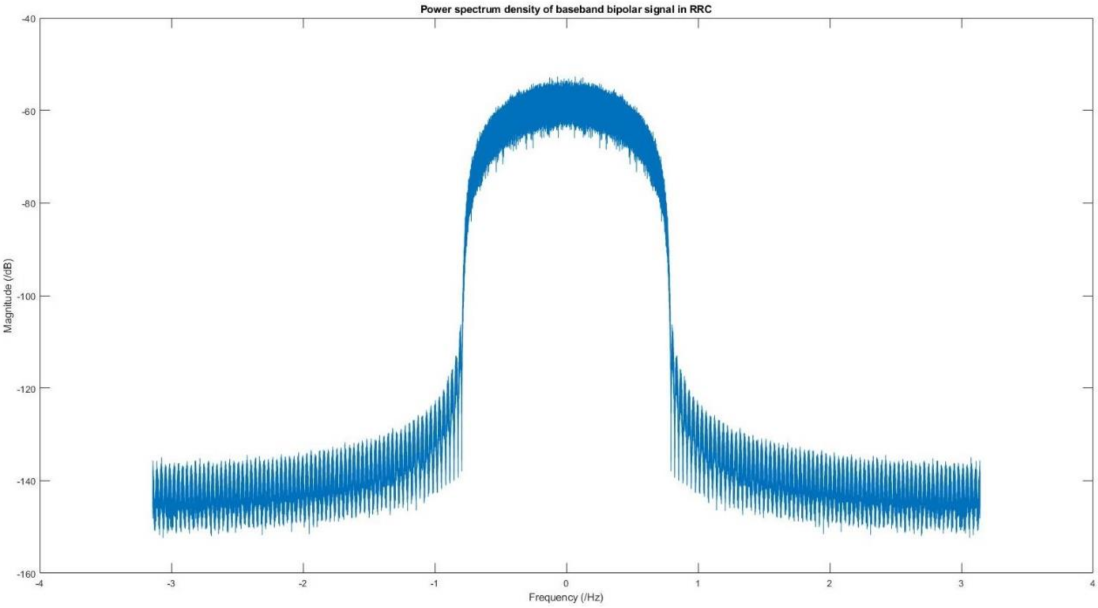

Figure 15: Power spectrum density of baseband bipolar signal in RRC.

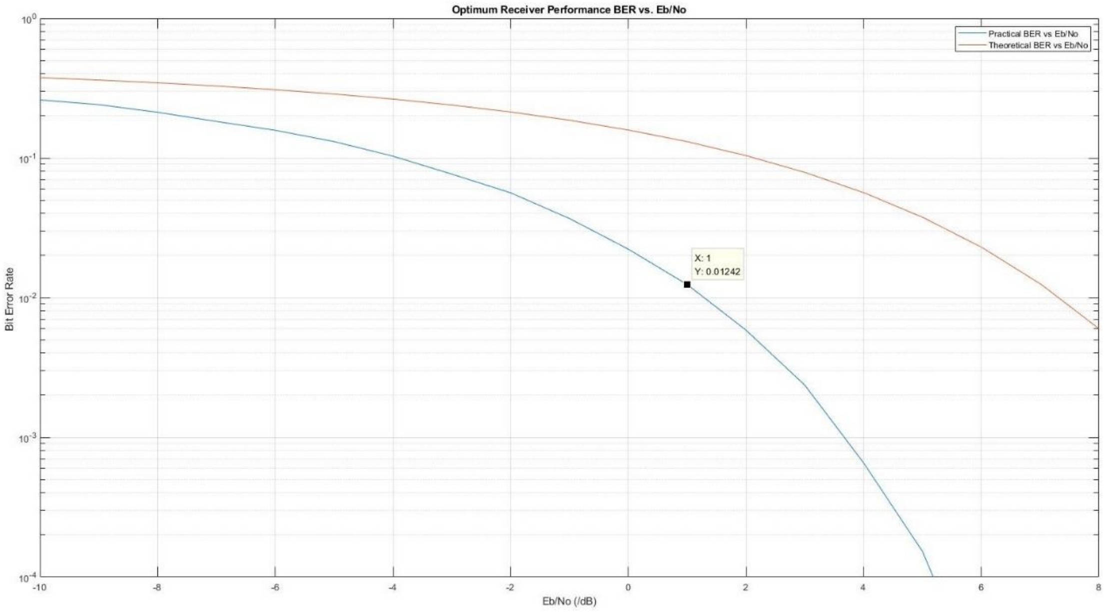

Figure 16: BER vs. E_b/N_0 for unipolar signal in rectangular pulse.

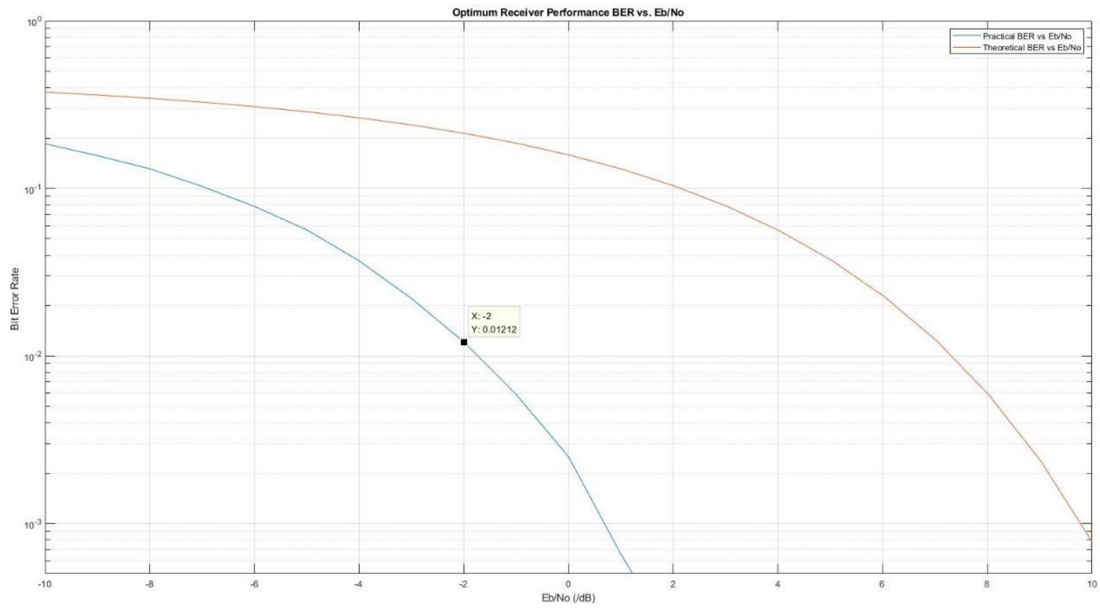

Figure 17: BER vs. E_b/N_0 for bipolar signal in rectangular pulse.

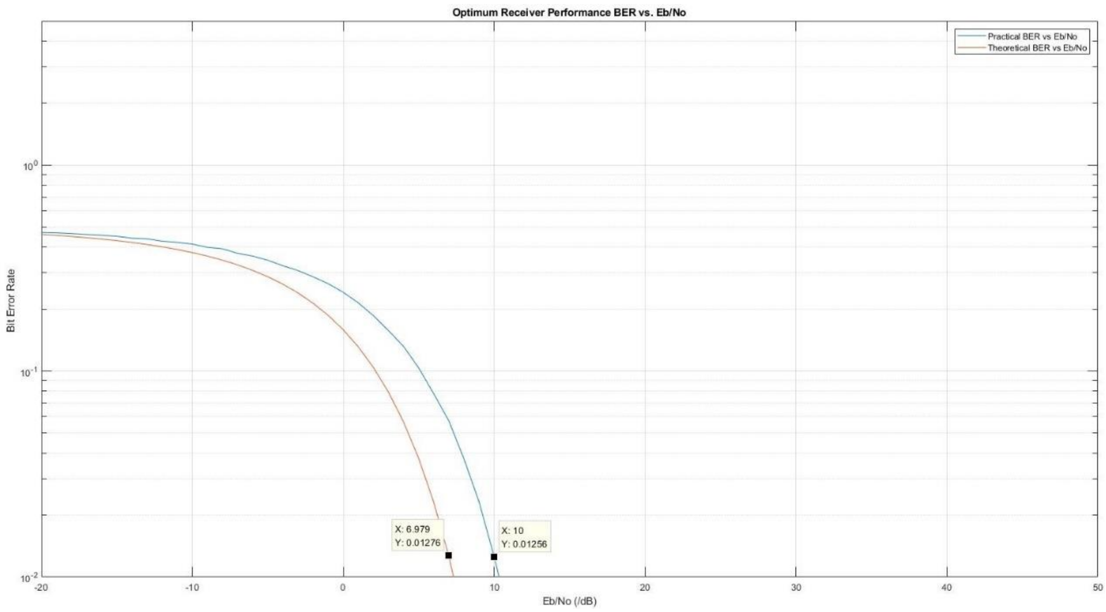

Figure 18: BER vs. E_b/N_0 for unipolar signal in RCC pulse.

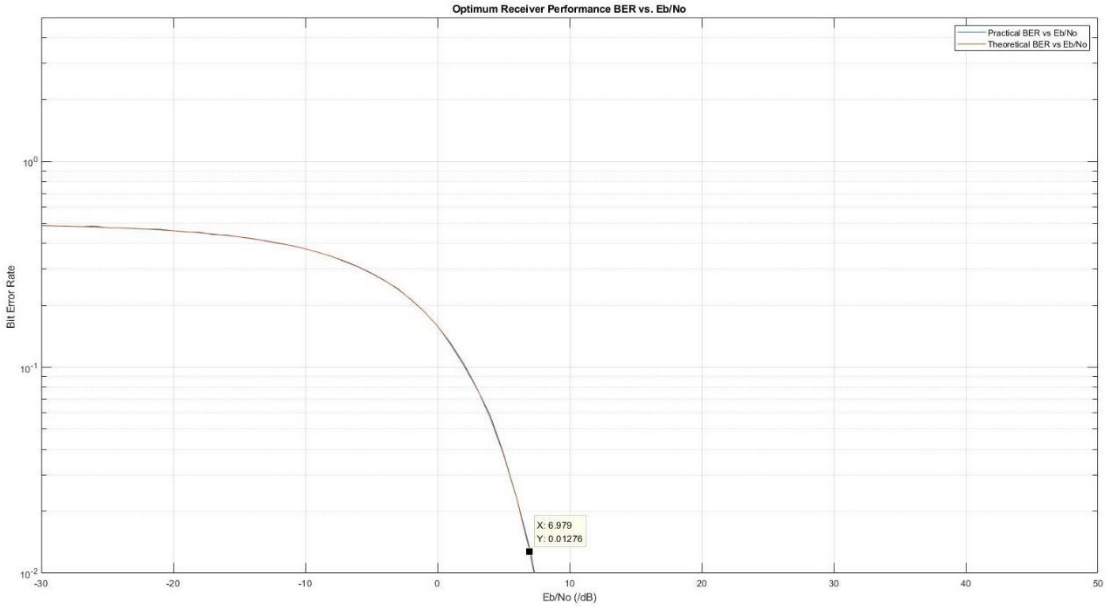

Figure 19: BER vs. E_b/N_0 for bipolar signal in RCC pulse.

6. Discussion

a) Why pulse shaping is necessary?

In telecommunication systems, the wireless channels are commonly required to use a pulse shaping filter. There are two significant reasons:

1) In practice, numerous working environments require data transmission systems to transmit data in limited bandwidth. By limiting the bandwidth of the channel, efficiency of the bandwidth would be increased, which could be referred to the formula:

$$Bandwidth\ Efficiency = \frac{data\ rate}{bandwidth}(bits/sec/Hz)$$

In real application, it allows more users to transmit information. However, the limited bandwidth would cause increasing BER in receiver.

2) Inter symbol Interference (ISI) is a common issue in data transmission. As discussed in 1), the limited bandwidth will cause high BER in receiver. Therefore, the limited bandwidth system usually applies pulse shaping to minimize BER in receiver.

RRC is the practical pulse shaping because it has limited bandwidth and ISI could be reduced. The discussion of RRC has been presented in previous sections.

b) What is the appropriate oversampling rate?

According to the Nyquist sampling theory, the operate oversampling could be verified as:

$$f_s \geq 2f_{max}$$

where f_{max} is the highest frequency in the transmitted signal.

For rectangular pulse, it is a sinc function in frequency domain which is shown in Figure 20.

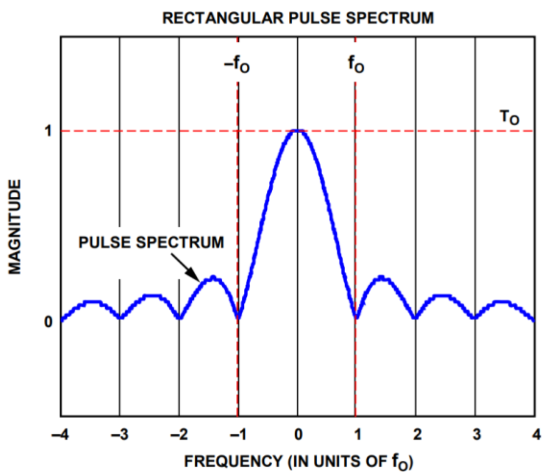

Figure 20: Rectangular pulse in frequency domain.

From Figure 20, it could be observed that the null points are always in the points $N * f_0$, where N is integer number. Hence, in theory, the frequency could be expanded to $\pm\infty$. In conclusion, the highest frequency of the rectangular pulse is infinite.

For root raised cosine (RRC) pulse, the bandwidth depends on the roll-off β. During this assignment, the roll-off β is considered as 1, therefore $f_{max} = \frac{1}{T_b} = 1$. According to Nyquist sampling theory,

$$f_s \geq 2f_{max} = 2 * \frac{1}{T_b} = 2\ Hz$$

Therefore, the appropriate oversampling rate for the RRC is 2 Hz. In this assignment, the oversampling rate will be 8 Hz.

c) What is matched filter and why it is necessary?

The matched filter is an optimum receiver achieving the minimum BER whose values is the maximum signal to noise (SNR) in receiver and it is usually used in baseband transmission. The main purpose of the matched filter is to recover the digital signals and obtain minimum BER.

In receiver, it will recover the transmitted data. During transmission, the original signal might be interfered by external noise, which will cause errors in receiver. After applying the matched filter, useful components of the signal could be relatively strong and the external noise could be eliminated.

7. Verification of the BER performance with analytical results

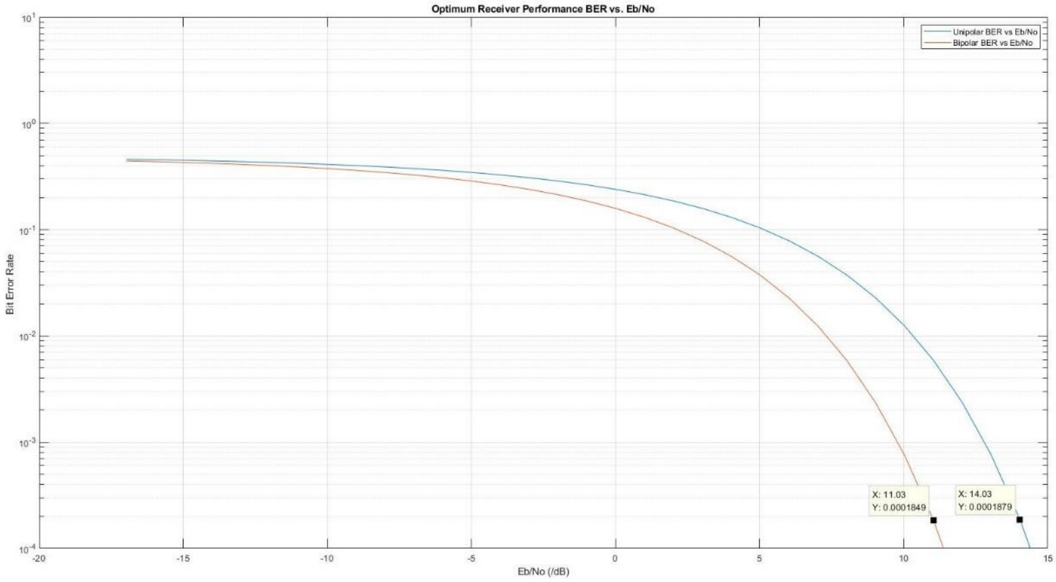

Figure 20: The comparison of unipolar and bipolar BER vs. E_b/N_0 in theory

Figure 20 is the comparison of unipolar and bipolar BER vs. E_b/N_0 in theory and it could be discovered that the BER difference between bipolar signal and unipolar signal is 3 dB when the SNR are the same.

In practice, it could be observed that the BER performance of rectangular pulse in unipolar and bipolar signals are below the theoretical values. As discussed previously, the rectangular pulse will cause more ISI and BER because its bandwidth in frequency domain is infinite, which is contrary to the practice. Assuming the noise values in dB are equal, from Figure16 and Figure 17, the BER difference is$1-(-2)=3\ dB$, which is corresponding to the theory. Therefore, the verification of the BER performance has been done and it could be concluded that the MATLAB simulation was correct.

Similarly, from Figure 18 and Figure 19, the BER difference is$10-7=3\ dB$, which is corresponding to the theory. As mentioned previously, RRC filter is commonly used in pulse shaping in real application, because it can limit the bandwidth and reduce ISI. Comparing from Figure 16 to Figure 19, it could be stated that the BER has been highly improved.

8. Conclusion

In conclusion, the verification in last section indicated the practices are corresponding to the theories, it is therefore, this assignment has been finished correctly. Through this assignment, students have gained skills of using MATLAB software to simulate the process of a baseband transceiver for binary digital signals, which help students learn the theory and practice in deeper way and is quite useful for future study. In this report, it has presented:

1) The overview of the designed baseband signal transceiver.
2) Baseband signal waveforms in the time domain and corresponding power spectrum density.
3) 4 cases of optimum receiver performance.
4) Several significant discussions (pulse shaping, appropriate oversampling rate and matched filter).
5) Verification of the BER performance with analytical results

9. Appendix

RECTUNI.m

```
close all;
Tb1=1; % bit duration
fs = 8; % sampling rate
dt=Tb1/fs; %increament
N1=8192; %number of data bits

t1=0:dt:(Tb1*N1-dt);
t2=0:dt:(fs*N1-dt);
t3=0:dt:(Tb1*N1);
signal = randi([0 1], fs*N1, 1); %generate discrete bit 0 or 1.

rect_pulse = rectpulse(signal,fs); %pulse shaping and oversampling.
figure;
plot(t2, rect_pulse);%plot baseband signal waveform in time domain.
grid on;
axis([0 fs*Tb1*N1 -3 3]);
title('Unipolar signal in rectangular pulse shaping in baseband');

% example_mix=awgn(rect_pulse, 20); %example: adding noise which is 20dB.
% figure; %baseband signal waveform in time domain.
% plot(t2, rect_pulse);
% hold on;
% plot(t2, example_mix);
% grid on;
% axis([0 fs*Tb1*N1 -3 3]);
% title('Unipolar signal with AWGN in rectangular pulse shaping in baseband');
% legend('Without AGWN','With AGWN');

noise_index =-39:1:26; %noise in dB, it is from -80~52.
ber=[];%for storing every ber when noise_index varies

for i=1:length(noise_index) %obtain a series of ber when noise_index varies
   noise = wgn(size(rect_pulse, 1), 1, noise_index(i));
   ber_mix = awgn(rect_pulse, noise_index(i), 'measured');%adding noise
   ber_matched_filter = ones(fs,1); %initialize matched filter
   ber_matched_output = conv(ber_mix, ber_matched_filter); %convolution of signal with
noise and matched filter
   ber_matched_output1=[]; % initialize the ber_matched_output1

   for c=1:length(ber_matched_output) %transposition
      ber_matched_output1(1, c) = ber_matched_output(c);
   end
```

```
    ber_matched_output2= circshift(ber_matched_output1, [0,-7]); %remove delay

    output = downsample(ber_matched_output2,8); %downsample

    output_series=[];% used for storing recovered digits
    for j=1:length(output)
        if (output(j)>4) %threshold is 4
            output_series(j)=1;
        else
            output_series(j)=0;
        end
    end

    e=0;
    for k=1:length(signal) %calculate number of error
        if (signal(k)~=output_series(k))
            e = e + 1;
        end
    end
    ber(i) = e/length(output_series); %calculate BER
end

semilogy(noise_index,ber); %plot practical BER vs Eb/No and theoretical BER vs Eb/No
tBER1=qfunc(sqrt((4./(10.^(noise_index./10)))));
hold on;
semilogy(10*log10(4./(10.^(noise_index./10))),(tBER1));
grid on;
xlim([-10 8]);
ylim([0.0001 1]);
legend('Practical BER vs Eb/No','Theoretical BER vs Eb/No')
xlabel('Eb/No (/dB)');
ylabel('Bit Error Rate ');
title('Optimum Receiver Performance BER vs. Eb/No');

% figure;
% nfft=2^15; %number of FFT
% [pxx, f] = pwelch(rect_pulse, nfft, nfft/2, 'centered','power');
% plot(f,10*log10(pxx));  %plot power spectrum density
% xlabel('Frequency (/Hz)');
% ylabel('Magnitude (/dB)');
% %axis([-8 8 -30 30]);
% title('Power spectrum density of baseband unipolar signal in rectangular pulse.');
```

RECTBI.m

```
close all;
Tb1=1; % bit duration
fs = 8; % sampling rate
dt=Tb1/fs; %increament
N1=8192; %number of data bits

t1=0:dt:(Tb1*N1-dt);
t2=0:dt:(fs*Tb1*N1-dt);
t3=0:dt:(Tb1*N1);
signal = 2*(randi([0 1], Tb1*fs*N1, 1))-1;%generate discrete bit -1 or 1.

rect_pulse = rectpulse(signal,fs);%pulse shaping and oversampling
figure;
plot(t2, rect_pulse);%plot baseband signal waveform in time domain.
grid on;
axis([0 fs*Tb1*N1 -4.5 4.5]);
title('Bipolar signal in rectangular pulse shaping in baseband');

% example_mix=awgn(rect_pulse, 20); %example: adding noise which is 20dB.
% figure;
% plot(t2, rect_pulse);%baseband signal waveform in time domain.
% hold on;
% plot(t2, example_mix);
% grid on;
% axis([0 fs*Tb1*N1 -3 3]);
% title('Bipolar signal with AWGN in rectangular pulse shaping in baseband');
% legend('Without AGWN','With AGWN');

noise_index =-39:1:26;%noise in dB, it is from -80~52.
ber=[];%for storing every ber when noise_index varies

for i=1:length(noise_index)%for storing every ber when noise_index varies
   noise = wgn(size(rect_pulse, 1), 1, noise_index(i));
   ber_mix = awgn(rect_pulse, noise_index(i), 'measured');%adding noise
   ber_matched_filter = ones(fs,1);%initialize matched filter
   ber_matched_output = conv(ber_mix, ber_matched_filter);%convolution of signal with
noise and matched filter
   ber_matched_output1=[];% initialize the ber_matched_output1

   for c=1:length(ber_matched_output) %transposition
      ber_matched_output1(1, c) = ber_matched_output(c);
   end
```

```
    ber_matched_output2= circshift(ber_matched_output1, [0,-7]);%remove delay

    output = downsample(ber_matched_output2,8);%downsample

    output_series=[];% used for storing recovered digits
    for j=1:length(output)
        if (output(j)>0)%threshold is 0
            output_series(j)=1;
        else
            output_series(j)=-1;
        end
    end

    e=0;
    for k=1:length(signal) %calculate number of error
        if (signal(k)~=output_series(k))
            e = e + 1;
        end
    end

    ber(i) = e/length(output_series);

end

semilogy(noise_index,ber); %plot practical BER vs Eb/No and theoretical BER vs Eb/No
tBER1=qfunc(sqrt((8./(10.^(noise_index./10)))));
hold on;
semilogy(10*log10(8./(10.^(noise_index./10))),(tBER1));
grid on;
xlim([-10 10]);
ylim([0.0005 1]);
legend('Practical BER vs Eb/No','Theoretical BER vs Eb/No')
xlabel('Eb/No (/dB)');
ylabel('Bit Error Rate ');
title('Optimum Receiver Performance BER vs. Eb/No');

% figure;
% nfft=2048; %number of FFT
% [pxx, f] = pwelch(rect_pulse, nfft, nfft/2, 'centered','power');
% plot(f,10*log10(pxx));   %plot power spectrum density
% xlabel('Frequency (/Hz)');
% ylabel('Magnitude (/dB)');
% %axis([-8 8 -30 30]);
% title('Power spectrum density of baseband bipolar signal in rectangular pulse');
```

RRCUNI.m

```
close all;
Tb1=1; % bit duration
fs = 8; % sampling rate
dt=Tb1/fs; %increament
N1=8192; %number of data bits

t1=0:dt:(Tb1*N1-dt);
t2=0:dt:(fs*Tb1*N1-dt);
signal = randi([0 1], Tb1*fs*N1, 1); %generate discrete bit 0 or 1.

rolloff = 1; %Roll-off
span = 8;  %Filter span
sps = 8; %Samples per symbol

RRC_filter = rcosdesign(rolloff, span, fs,'sqrt'); %RRC filter
RRC_pulse = upfirdn(signal, RRC_filter, sps); % oversample

t3=0:1:size(RRC_pulse, 1)-1;
figure;
plot(t3, RRC_pulse);%plot baseband signal waveform in time domain.
grid on;
axis([0 1100 -0.3 0.6]);
title('Unipolar signal in RRC pulse shaping in baseband');

% example_mix=awgn(RRC_pulse, 20);%example: adding noise which is 20dB.
% figure;
% plot(t3, RRC_pulse);%baseband signal waveform in time domain.
% hold on;
% plot(t3, example_mix);
% grid on;
% axis([0 1100 -0.6 0.8]);
% title('Unipolar signal with AWGN in RRC pulse shaping in baseband');
% legend('Without AGWN','With AGWN');

noise_index =-39:1:26;%noise in dB, it is from -39~26.
ber=[];%for storing every ber when noise_index varies
SNR=[];

for i=1:length(noise_index)%for storing every ber when noise_index varies
   %noise = wgn(size(RRC_pulse, 1), 1, noise_index(i));
   SNR = noise_index(i)- 10*log10(sps); % real SNR
   ber_mix = awgn(RRC_pulse, SNR, 'measured');%adding noise
   output_filter = upfirdn(ber_mix, RRC_filter, 1, sps);%convolution of signal with noise
```

```
and matched filter and downsample
    output_filter = output_filter(span+1:end-span); % remove delay

    output_series=[];
    for j=1:length(signal) % used for storing recovered digits
        if (output_filter(j)>0.5)%threshold is 0.5
            output_series(j)=1;
        else
            output_series(j)=0;
        end
    end

    e=0;
    for k=1:length(signal) %calculate number of error
        if (signal(k)~=output_series(k))
            e = e + 1;
        end
    end

    ber(i) = e/length(output_series);

end

semilogy(noise_index,ber);%plot practical BER vs Eb/No and theoretical BER vs Eb/No
tBER1=qfunc(sqrt((0.5./(2.*10.^(noise_index./10)))));
hold on;
semilogy(10*log10(0.5./(2.*10.^(noise_index./10))),(tBER1));
grid on;
xlim([-20 50]);
ylim([0.01 5]);
legend('Practical BER vs Eb/No','Theoretical BER vs Eb/No')
xlabel('Eb/No (/dB)');
ylabel('Bit Error Rate ');
title('Optimum Receiver Performance BER vs. Eb/No');

% figure;
% nfft=2^15; %number of FFT
% [pxx, f] = pwelch(RRC_pulse, [], nfft/2, 'centered','power');
% plot(f,10*log10(pxx));  %plot power spectrum density
% xlabel('Frequency (/Hz)');
% ylabel('Magnitude (/dB)');
% %axis([-8 8 -30 30]);
% title('Power spectrum density of baseband unipolar signal in RRC');
```

RRCBI.m

```
close all;
Tb1=1; % bit duration
fs = 8; % sampling rate
dt=Tb1/fs; %increament
N1=8192; %number of data bits

t1=0:dt:(Tb1*N1-dt);
t2=0:dt:(10*Tb1*N1-dt);
signal = 2*(randi([0 1], Tb1*fs*N1, 1))-1;%generate discrete bit -1 or 1.

rolloff = 1; %Roll-off
span = 8;  %Filter span
sps = 8; %Samples per symbol

RRC_filter = rcosdesign(rolloff, span, fs,'sqrt'); %RRC filter
RRC_pulse = upfirdn(signal, RRC_filter, sps); % oversample

t3=0:1:size(RRC_pulse, 1)-1;
figure;
plot(t3, RRC_pulse);%plot baseband signal waveform in time domain.
grid on;
axis([0 1100 -1 1]);
title('Bipolar signal in baseband');

% example_mix=awgn(RRC_pulse, 20);%example: adding noise which is 20dB.
% figure;
% plot(t3, RRC_pulse);%baseband signal waveform in time domain.
% hold on;
% plot(t3, example_mix);
% grid on;
% axis([0 1100 -1 1]);
% title('Bipolar signal with AWGN in RRC pulse shaping in baseband');
% legend('Without AGWN','With AGWN');

noise_index =-39:1:26;%noise in dB, it is from -39~26.
ber=[];%for storing every ber when noise_index varies
SNR=[];

for i=1:length(noise_index)%for storing every ber when noise_index varies
   noise = wgn(size(RRC_pulse, 1), 1, noise_index(i));
   SNR = noise_index(i)- 10*log10(sps);% real SNR
   ber_mix = awgn(RRC_pulse, SNR, 'measured');%adding noise
   output_filter = upfirdn(ber_mix, RRC_filter, 1, sps);%convolution of signal with noise
```

```
and matched filter and downsample
    output_filter = output_filter(span+1:end-span);% remove delay

    output_series=[];% used for storing recovered digits
    for j=1:length(signal)
        if (output_filter(j)>0)%threshold is 0
            output_series(j)=1;
        else
            output_series(j)=-1;
        end
    end

    e=0;
    for k=1:length(signal)  %calculate number of error
        if (signal(k)~=output_series(k))
            e = e + 1;
        end
    end

    ber(i) = e/length(output_series);

end

semilogy(noise_index,ber);%plot practical BER vs Eb/No and theoretical BER vs Eb/No
tBER1=qfunc(sqrt((0.5./(2.*10.^(noise_index./10)))));
hold on;
semilogy(10*log10(0.5./(2.*10.^(noise_index./10))),(tBER1));
grid on;
xlim([-30 50]);
ylim([0.01 5]);
legend('Practical BER vs Eb/No','Theoretical BER vs Eb/No')
xlabel('Eb/No (/dB)');
ylabel('Bit Error Rate ');
title('Optimum Receiver Performance BER vs. Eb/No');

% figure;
% nfft=2^15; %number of FFT
% [pxx, f] = pwelch(RRC_pulse, [], nfft/2, 'centered','power');
% plot(f,10*log10(pxx));  %plot power spectrum density
% xlabel('Frequency (/Hz)');
% ylabel('Magnitude (/dB)');
% %axis([-8 8 -30 30]);
% title('Power spectrum density of baseband bipolar signal in RRC');
```

theory.m

```
noise_index =-39:1:26;
tBER1=qfunc(sqrt((4./(10.^(noise_index./10))))); %for unipolar
tBER2=qfunc(sqrt((8./(10.^(noise_index./10))))); %for bipolar

semilogy(10*log10(8./(10.^(noise_index./10))),(tBER1)); %plot unipolar BER vs Eb/No and
                                                         %bipolar BER vs Eb/No
hold on;
semilogy(10*log10(8./(10.^(noise_index./10))),(tBER2));
grid on;
% xlim([-10 8]);
 ylim([0.0001 10]);
legend('Unipolar BER vs Eb/No','Bipolar BER vs Eb/No');
xlabel('Eb/No (/dB)');
ylabel('Bit Error Rate ');
title('Optimum Receiver Performance BER vs. Eb/No');
```