Project 1

Digital communications

ID	BN	Sec	Name
9202162	24	1	أحمد محمد أحمد شيبة
9203081	1	4	كلارا عيسى اسحاق عبدالمسيح
9203349	3	4	محمد ناصر محمد كمال
9203591	27	4	مينا صبحي كامل السيد
9203687	41	4	هنا ايهاب خاطر يونس

THE CODE

```
%Initialization of variables
A = 4;  %trasmitted voltage level
ensemble_size = 20000;  %number of realizations in the ensemble
num_bits = 100; %number of bits in each realization
bit_time = 70;  %pulse width to transmite one bit
sampling_time = 10; %DAC sampling time
num_samples_per_bit = floor(bit_time / sampling_time);  %numbers of samples in
on bit (7)
num_samples = num_samples_per_bit*num_bits; %total number of samples in one
realization

%Formation of the signals according to the required line code:
%fistly, generating the data as a matrix with nubmer of rows equal to the
%number of realization and number of coloumns represent the bits for each one
adding a bit for delay
data_bits = randi([0,1], ensemble_size, num_bits+1);
%secondly, generate an empty matrix to store the data after transforming it
%form bits to samples
data_expanded = zeros(ensemble_size, num_samples+num_samples_per_bit);
%thirdly, Determining from the user the type of line coding to sample the bits
Line_code = input("1. Unipolar NRZ\n2. Polar NRZ\n3. Polar RZ\nEnter The
Required Line Code: ");
%According to the input, the line code will be chosen
switch Line_code
    case 1  %Unipolar NRZ
        %transmit '0' as 0 and '1' as A
        data_symbols = data_bits * A;
        %transform bits to samples in all the realizations
        for i = 1:ensemble_size
            data_expanded(i,:) = repelem(data_symbols(i,:),num_samples_per_bit);
        end
    case 2  %Polar NRZ
        %transmit '0' as -A and '1' as A
        data_symbols = (2*data_bits-1)*A;
        %transform bits to samples in all the realizations
        for i = 1:ensemble_size
            data_expanded(i,:) = repelem(data_symbols(i,:),num_samples_per_bit);
        end
    case 3  %Polar RZ
        %transmit '0' as -A and '1' as A for the ceil of half the bit
        %period (4 out of 7 in our case)
        for i = 1:ensemble_size
            for j = 1:num_bits+1
                % Index of first sample for current bit
                start_idx = ((j - 1) * num_samples_per_bit) + 1;
                if data_bits(i,j) == 1 % Transmit '1' as +A followed by zero for
the remaining bit period

data_expanded(i,start_idx:start_idx+floor(num_samples_per_bit/2)) = A;
                else % Transmit '0' as -A followed by zero for the remaining bit
period

data_expanded(i,start_idx:start_idx+floor(num_samples_per_bit/2)) = -A;
                end
            end
```

```
        end
end


%Adding a random dalay at the start of each realization
%firstly, generating a delay less than the number of samples per bit for all
realizations
Td = randi([0,(num_samples_per_bit-1)],ensemble_size,1);
%defining a matrix to store the data after the delay
data = zeros(ensemble_size, num_samples);
%looping on all the realization and store the data from the end of the
%delay until getting all the samples
for i = 1:ensemble_size
    data(i,:)= data_expanded(i, Td(i)+1 : num_samples + Td(i));
end

%Ploting the final Realizations generated
figure('Name','Realizations');
%define a variable (t) represent the x-axis start from 0 and has a value with
%each sampling time until reaching the number of samples-1 as we start from 0
t = 0:sampling_time:(num_samples-1)*sampling_time;
%looping 5 times to plot first 5 realization
for i = 1 : 5
    %plot the 5 realization in 1 figure inorder in a vertical position
    subplot(5,1,i);
    %set the value of x-axis as (t) and the corresponding y-axis is the
generated data
    plot(t, data(i,:));
    %set a title for the graph
    str = sprintf("Realization %d",i);
    title(str);
    %set a limets from -6 to 6 on the y-axis
    ylim([-6 6]);
    %setting labels for x-axis and y-axis
    xlabel("Time(ms)");
    ylabel("Volts(V)");
end


%Q1: Getting the statistical Mean
%getting the sum of elements in each column and divided by their numeber which
in the
%number of realization to get the mean
stat_mean = sum(data, 1) / ensemble_size;
%plotting the statistical mean
figure ('Name', 'Mean');
plot(t, stat_mean);
title ("Statistical Mean");
ylim([-6,6]);
xlabel("Time(ms)")
ylabel("Mean(V)");

%Q2: Determine if the random process is stationary
%Firstly, to say that the process is stationary, the mean must be constant
%which is previously checked
%Secondly, we will compare autocorrelations to check if it is dependent on
%the time difference, so we have to get the autocorrelation in Q3 first
%Q3: Statistical Autocorrelation calculations
%generate an empty matrix to store the result in it
stat_acf = zeros(size(data(1,1:end)));
```

```
%loop on the coulmns of the data matrix to multiply the first coloumn by all
%the other coulmns and take their sum and divided by the number of elemnts
%in that coloumn which is represented by the ensemble size to get the right
%sided autocorrelation from 0 to the number of samples
for i = 1 : num_samples
    stat_acf(1,i) = sum((data(:,1) .* data(:,i))) / ensemble_size;
end
%get the left side by fliping the previously calculated autocorrelation and
%concatenate them to get the final statistical autocorrelation
stat_acf = cat (2, fliplr(stat_acf(2:num_samples)), stat_acf);

%Statistical Autocorrelation for another sample to check stationary
%choose a random number from the first quarter of our samples
random_check = randi([1,floor(num_samples/4)]);
%make an empty matrix to store the result
stat_acf2 = zeros(size(data(1,1:num_samples-random_check)));
%multiply from the random column with its seld and all it's consecutive and
%divided by the ensemble size to get the autocorrelation for the right side
for i = 0 : num_samples-random_check
    stat_acf2(1,i+1) = sum((data(:,random_check) .* data(:,i+random_check))) /
ensemble_size;
end
%flip the previous generated result and concatenate it to get the final
%result
stat_acf2 = cat (2, fliplr(stat_acf2(2:end)), stat_acf2);

%polting the two different statistical autocorrelation to compare them so
%that if they are approximatly the same then the process is stationary
%defining a new variable tau to represent the x-axis
tau = (-num_samples+1)*sampling_time :sampling_time: (num_samples-
1)*sampling_time;
%ploting the acf of the first column in the left side of the figure
figure ('Name', 'Stationary check');
subplot(1,2,1);
plot(tau, stat_acf);
title ("Statistical ACF for first column");
ylim([-2,20]);
xlabel("Tau(ms)")
ylabel("Autocorrelation");

tau2 = (-num_samples+random_check)*sampling_time :sampling_time: (num_samples-
random_check)*sampling_time;
%ploting the acf of the random column in the left right of the figure
subplot(1,2,2);
plot(tau2, stat_acf2);
title ("Statistical ACF for random column");
ylim([-2,20]);
xlabel("Tau(ms)")
ylabel("Autocorrelation");

%Q4: Time Mean and autocorrelation function for one waveform
%1. Time mean:
%choose a random realization to get it's time mean
random_realization = randi([1,ensemble_size]);
%get the mean by getting the sum of the row of the random realization and
%divided by the number of elements which is represented this time by the
%number of samples
time_mean = sum(data(random_realization,:)) / num_samples;
%replicate the mean over the number of samples to be plotted
time_mean = repelem(time_mean,num_samples);
```

```
%2. Time Autocorrelation
%firstly create to empty matrix, one to store the shifted data and the
%other to store the result of the time autocorrelation function
shifted = zeros(size(data(random_realization,:)));
time_acf = zeros(size(data(random_realization,:)));

%secondly, according to the chosen line code a random bit will be chosen
%to be put when the waveform is shifted
switch Line_code
    case 1
        %put A or 0 for Unipolar NRZ
        random_bit = randi([0,1]) * A;
    case 2
        %put A or -A for Polar NRZ
        random_bit = (2*(randi([0,1]))-1)*A;
    case 3
        %put A or -A for Polar RZ
        random_bit = (2*(randi([0,1]))-1)*A;
end
%define a new variable to calculate the number of the sample in the bit (from 1
to 7)
flag = 1;

%loop on the number of samples to multiply the waveform by itself and
%shifted waveform from being shifted by 1 to the number of samples to cover
%it all
for i = 1 : num_samples
    %firstly store the waveform itself to be initially multiply by itself
    if i == 1
        shifted = data(random_realization,:);
        flag = flag+1; %increment the flag after the first sample in the bit is
used
    %check if we are using polar return to zero to add the shifted sample
    %firstly equal to zero for three samples then a random value between A
    %or -A for the 4 remaining samples in the bit
    elseif Line_code==3
        if flag == 1 || flag == 2 || flag == 3
            shifted (2:end) = shifted(1: end-1);
            shifted (1) = 0;
        else
            shifted (2:end) = shifted(1: end-1);
            shifted (1) = random_bit;
        end
        flag = flag+1; % increament the flag each sample to determine its
position in the bit
    else
        % for unipolar or polar NRZ the shifted bit will be the random bit
        % that has been set before
        shifted (2:end) = shifted(1: end-1);
        shifted (1) = random_bit;
    end

    %multiply the waveform by its shifted version then sum and divide it to get
the
    %acf and then store it
    time_acf(i) = sum((data(random_realization,:) .* shifted(1:end)), 'all') /
num_samples;

    %check if the bit has been ended and its 7 samples has already shitfted
```

```
    %to generate a new random value
    if mod(i,7) == 0
        switch Line_code
            case 1
                random_bit = randi([0,1]) * A;
            case 2
                random_bit = (2*(randi([0,1]))-1)*A;
            case 3
                random_bit = (2*(randi([0,1]))-1)*A;
        end
        flag = 1;
    end
end
%after getting the right sided acf, flip it and concatenate to get the
%final graph
time_acf = cat (2, fliplr(time_acf(2:num_samples)), time_acf);

%Q5: now we can determine if it is ergodic by comparing the figure of the
%time mean with the statistical mean and the time autocorrelation with the
%statistical autocorrelation if both are equal so we can call it an ergodic
%process
figure ('Name', 'Time mean and autocorrelation');
%plotting the time mean in the left half of the figure
subplot(1,2,1);
plot(t, time_mean);
title ("Time Mean");
ylim([-6,6]);
xlabel("Time(ms)")
ylabel("Mean(V)");
%plotting the time autocorrelation funtion in the other half
subplot(1,2,2);
plot(tau, time_acf);
title ("Time Autocorrelation");
ylim([-2,20]);
xlabel("Tau(ms)")
ylabel("Autocorrelation");

%Q6: calculating bandwidth
%to calculate the bandwindth a graph of the PSD has to be plotted to get
%the bandwidth from it
%firstly, to get the x-axis correctly discribe the value of the frequansy
%a variable k has been defined with the size of our acf and then through
%the equation k*fs/(2*num_samples) we can get the right representaion
k = -num_samples + 1: num_samples - 1;
fs = 100;  %Sampling frequency
figure('Name','PSD');
%on the other hand the y-axis is represented by the absolute value of the
%shifted fast fourier transform of the acf to represent the PSD
plot(k*fs/(2*num_samples),abs(fftshift(fft(stat_acf))));
title("PSD");
%limiting the y-axis in unipolar NRZ as it goes to infinity at frequensy = 0
if Line_code == 1
    ylim([0,150]);
end
xlabel("Frequency(Hz)");
ylabel("PSD");
```

1. UNIPOLAR NRZ

GENERATING THE REALIZATIONS

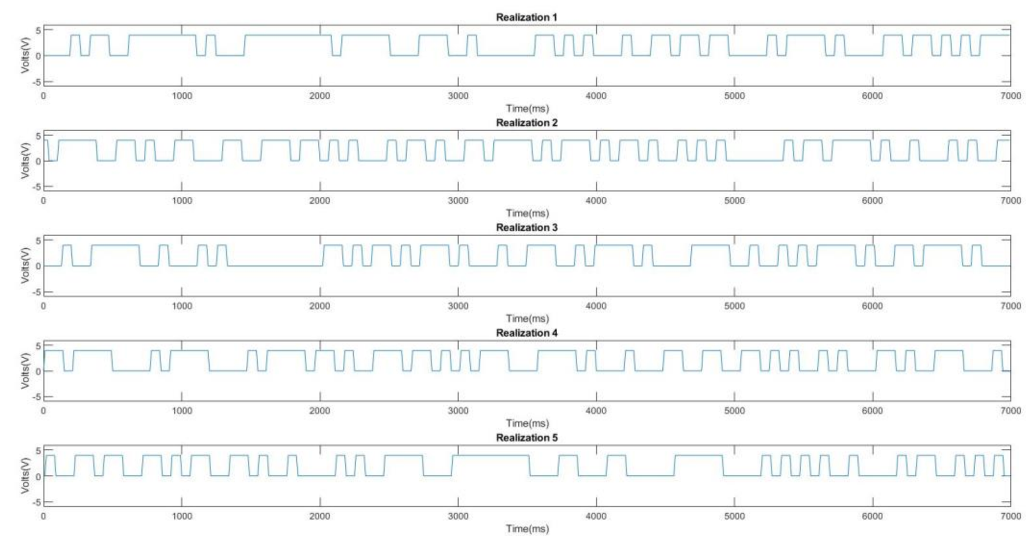

Figure 1: realizations of Unipolar NRZ line code

Q1: STATISTICAL MEAN

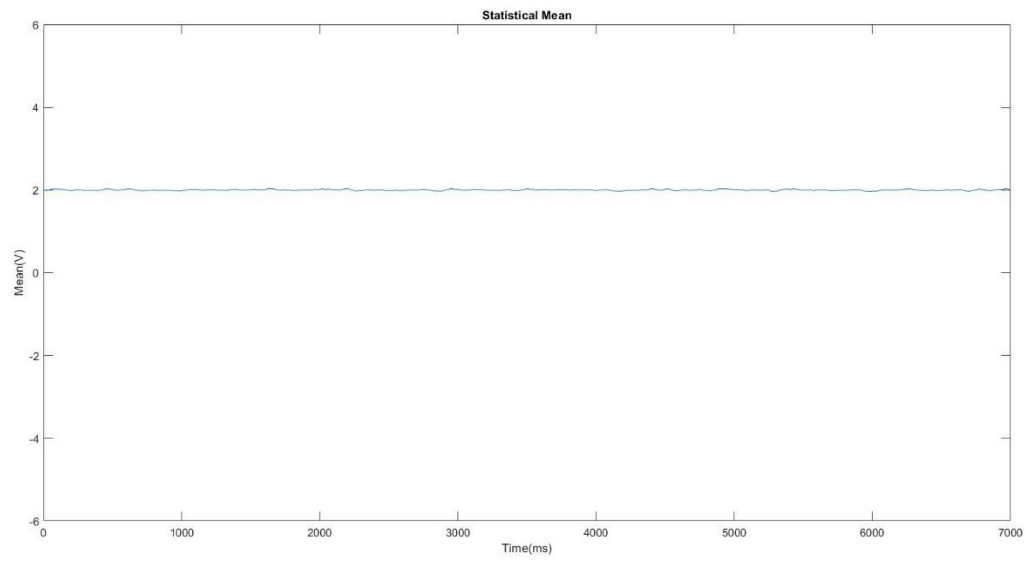

Figure 2: statistical mean of Unipolar NRZ line code

From figure 2, we can observe that statistical mean is constant and approximately equal to **2**.

$$E\big(X(t)\big) = \left(\frac{A}{2}\right) + \left(\frac{0}{2}\right) = \frac{A}{2} = 2$$

Q3: ENSEMBLE AUTOCORREALTION FUNCTION

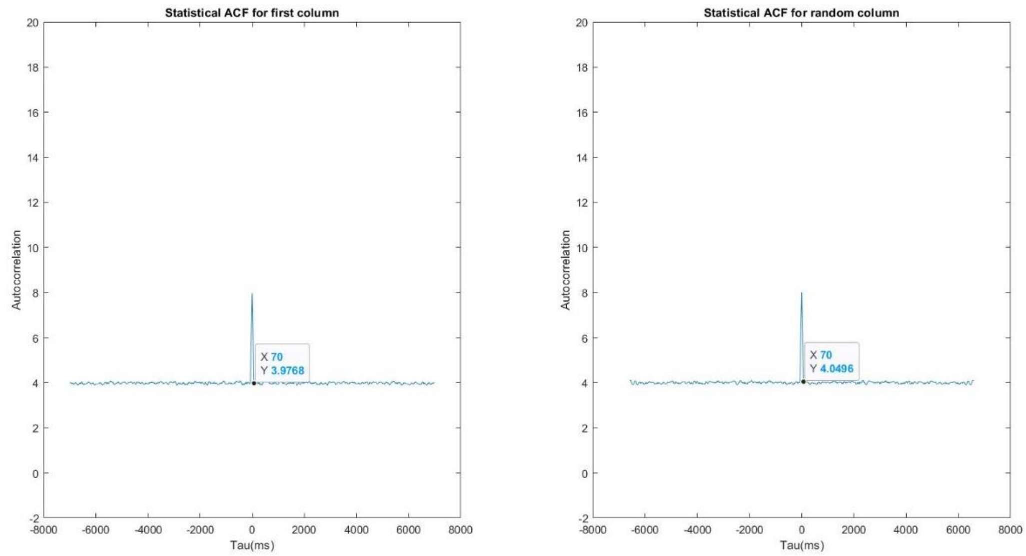

Figure 3: statistical autocorrelation of two different times

The autocorrelation indicates how much the signal at t1 is correlated to the signal at t2, hence the autocorrelation must be maximum at time difference (τ) equals to zero because the correlation of the signal and itself at any time t must be the highest comparing with the signal at time t and the signal at any other time instance, and by increasing the time difference (τ), the autocorrelation decreases till become approximately constant

$R_x(\tau) = \begin{cases} \frac{A^2}{4}\left(2 - \frac{|\tau|}{T_b}\right), & |\tau| \leq T_b \\ \frac{A^2}{4}, & |\tau| > T_b \end{cases}$, where A = 4. $R_x(0) = \frac{A^2}{4} \times 2 = 8.$

The graph shows that the autocorrelation of two different times is the same which means the **autocorrelation is not function in time absolute but function in time difference**, and its maximum at $\boldsymbol{\tau = 0}$ and starts to decreases until become constant equal to **4** from $\boldsymbol{\tau = 70}$ which is the time bit duration (Tb).

Q2: STATIONARY CHECK

After calculating the statistical mean and statistical autocorrelation, we can now get back to check the stationarity of the process, as the WSS process has constant statistical mean, and the statistical autocorrelation is only function in the time difference.

1. Back to figure 2, we can find that the mean is constant over the time.
2. From figure 3, we compared the statistical autocorrelation at only two different times just to simplify instead of comparing with all times, so we can find that the autocorrelation is approximately the same in the two graphs.

From (1) and (2), the conditions of the WSS process are verified, so we can say that a **unipolar non-return to zero process is a WSS process**.

Q4: TIME MEAN AND AUTOCORRELATION OF ONE WAVEFORM

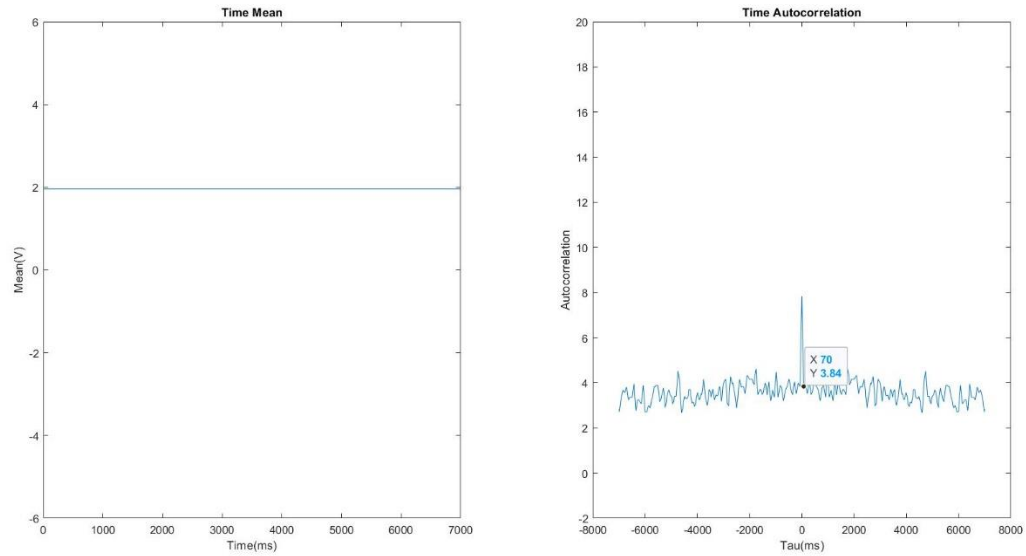

Figure 4: time mean and time autocorrelation of one waveform

The time mean is **constant** and approximately equal **2**.

The time autocorrelation starts to be **constant** at Tb.

Q5: ERGODIC CHECK

To say that a random process is Ergodic process the statistical mean and the time mean of any waveform must be equal, also the statistical autocorrelation and the time autocorrelation of any waveform must be equal, in general the statistics over the realizations must be equal the statistics over time of any waveform.

1. Comparing the mean in figure (2) and (4), the statistical mean is equal to the time mean which is equal to 2.
2. Comparing the autocorrelation in figure (3) and (4), the statistical autocorrelation is equal to the time autocorrelation which is approximately varying around 0.

From (1) and (2), the conditions of the Ergodic process are verified, so we can say that a **unipolar non-return to zero process is an Ergodic process**.

Knowing that by increasing the number of samples, the accuracy will increase, and the time statistics will be closer to the statistics of the whole ensemble.
Also, by increasing the number of realizations, the statistics will be more accurate, so the time statistics and the ensemble statistics will be the same.

Q6: BANDWIDTH OF THE TRANSMITTED SIGNAL

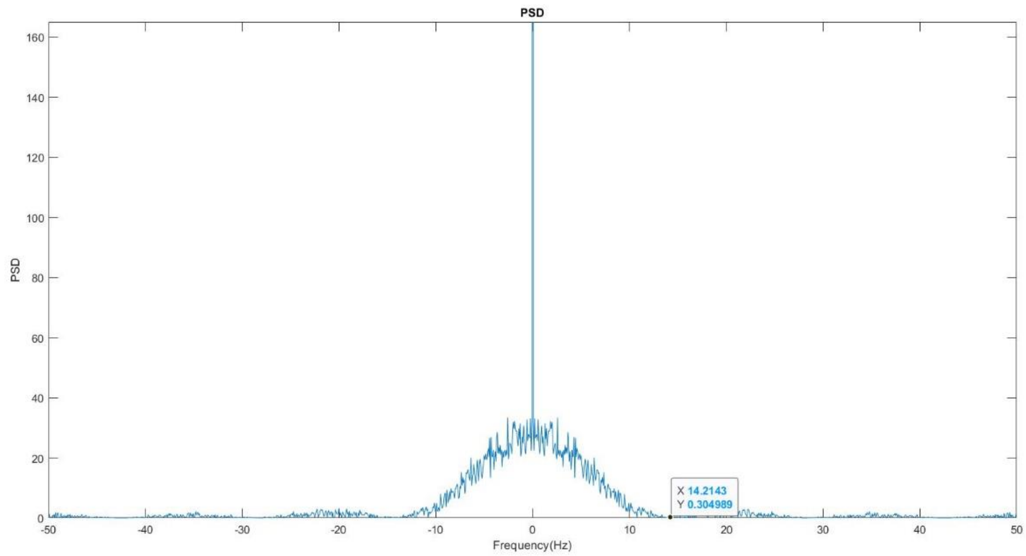

Figure 5: PSD of Unipolar NRZ

Theoretically, the bandwidth is approaching infinity, but we can take approximately till the first null which is laying at 1/bit duration ($\frac{1}{T_b} = 14.2857\ Hz$).

From the graph, we can find that we achieve the required bandwidth at **x = 14.21**, y approximately equal zero.

So, the bandwidth of the signal at the baseband = **14.21 Hz**.

2. POLAR NON-RETURN TO ZERO

GENERATING THE REALIZATIONS

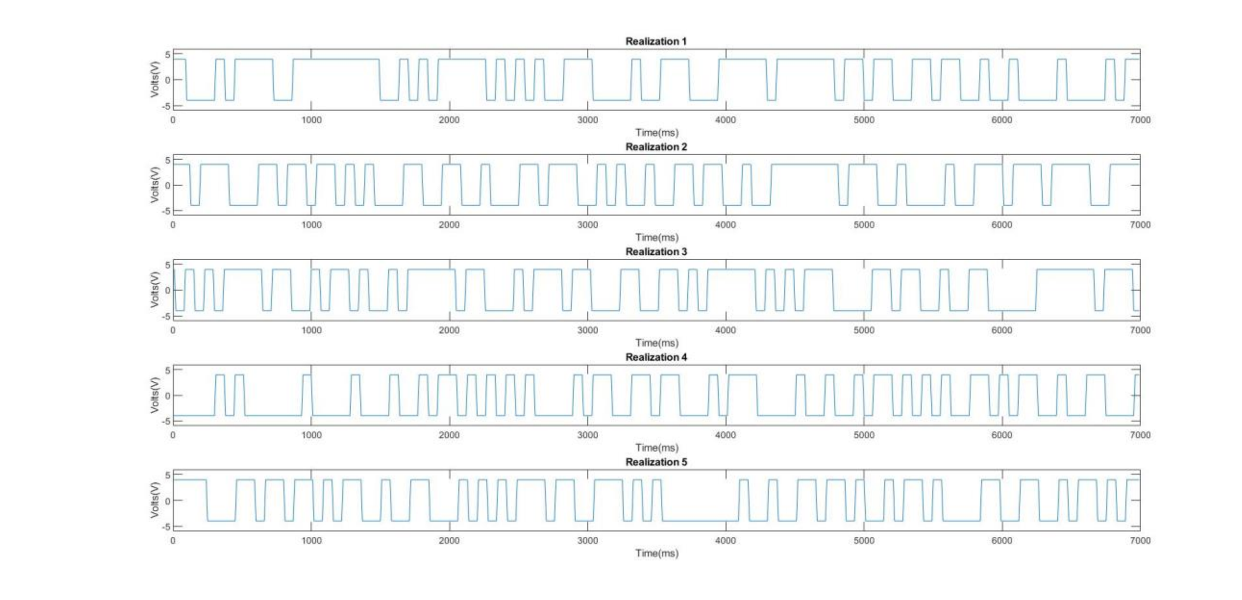

Figure 6: realizations of Polar NRZ line code

Q1: STATISTICAL MEAN

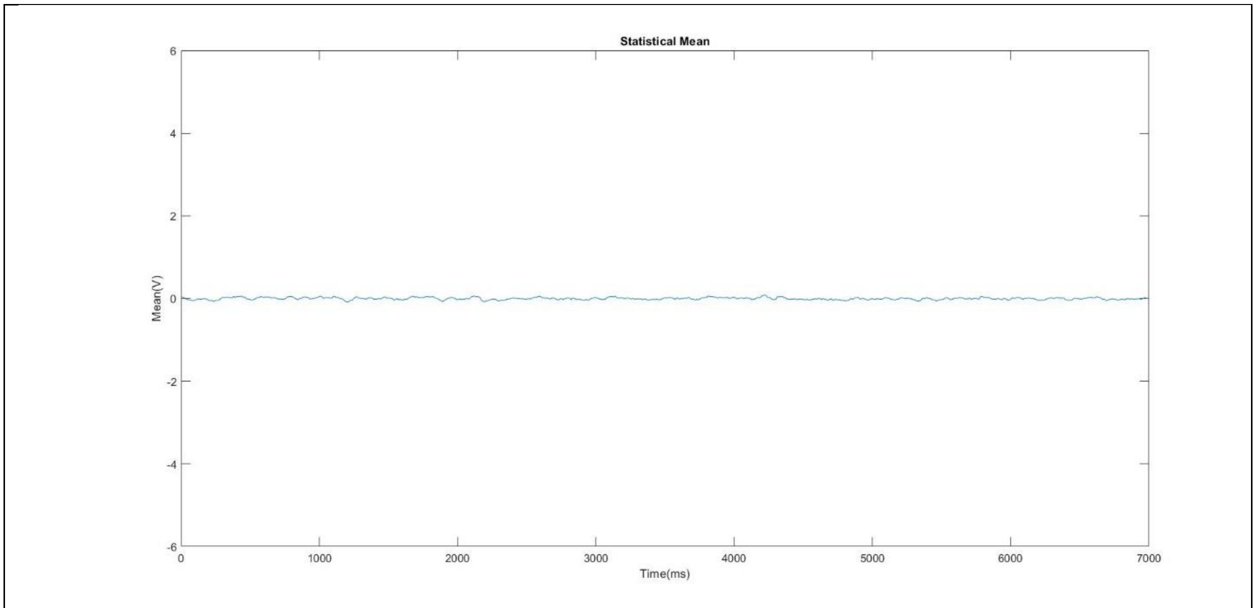

Figure 7: statistical mean of Polar NRZ line code

From figure 7, we can observe that statistical mean is constant and approximately equal to **0**.

$$E\big(X(t)\big) = \left(\frac{A}{2}\right) + \left(\frac{-A}{2}\right) = 0$$

Q3: ENSEMBLE AUTOCORREALTION FUNCTION

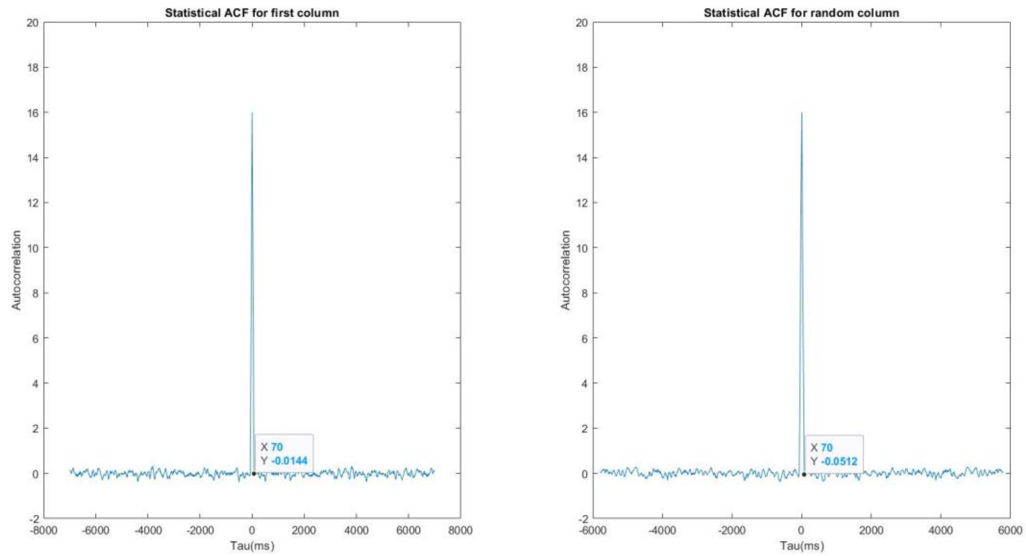

Figure 8: statistical autocorrelation of two different times

The autocorrelation indicates how much the signal at t1 is correlated to the signal at t2, hence the autocorrelation must be maximum at time difference (τ) equals to zero because the correlation of the signal and itself at any time t must be the highest comparing with the signal at time t and the signal at any other time instance, and by increasing the time difference (τ), the autocorrelation decreases till become approximately constant

$R_x(\tau) = \begin{cases} A^2\left(1 - \frac{|\tau|}{T_b}\right), & |\tau| \leq T_b \\ 0, & |\tau| > T_b \end{cases}$, where A = 4. $\qquad R_x(0) = A^2 = 16.$

The graph shows that the autocorrelation of two different times is the same which means the **autocorrelation is not function in time absolute but function in time difference**, and its maximum at $\boldsymbol{\tau = 0}$ and starts to decreases until become constant equal to **0** from $\boldsymbol{\tau = 70}$ which is the time bit duration (Tb).

Q2: STATIONARY CHECK

After calculating the statistical mean and statistical autocorrelation, we can now get back to check the stationarity of the process, as the WSS process has constant statistical mean, and the statistical autocorrelation is only function in the time difference.

3. Back to figure 7, we can find that the mean is constant over the time.
4. From figure 8, we compared the statistical autocorrelation at only two different times just to simplify instead of comparing with all times, so we can find that the autocorrelation is approximately the same in the two graphs.

From (1) and (2), the conditions of the WSS process are verified, so we can say that a **polar non-return to zero process is a WSS process**.

Q4: TIME MEAN AND AUTOCORRELATION OF ONE WAVEFORM

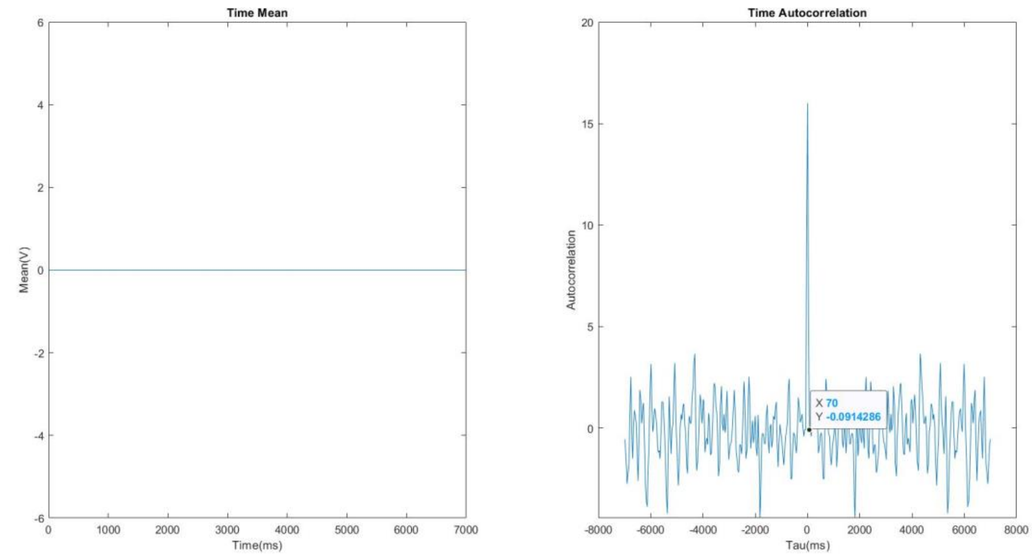

Figure 9: time mean and time autocorrelation of one waveform

The time mean is **constant** and approximately equal **0**.

The time autocorrelation starts to be **constant** at Tb.

Q5: ERGODIC CHECK

To say that a random process is Ergodic process the statistical mean and the time mean of any waveform must be equal, also the statistical autocorrelation and the time autocorrelation of any waveform must be equal, in general the statistics over the realizations must be equal the statistics over time of any waveform.

1. Comparing the mean in figure (7) and (9), the statistical mean is equal to the time mean which is equal to 0.
2. Comparing the autocorrelation in figure (8) and (9), the statistical autocorrelation is equal to the time autocorrelation which is approximately varying around 4.

From (1) and (2), the conditions of the Ergodic process are verified, so we can say that a **polar non-return to zero process is an Ergodic process**.

Knowing that by increasing the number of samples, the accuracy will increase, and the time statistics will be closer to the statistics of the whole ensemble.
Also, by increasing the number of realizations, the statistics will be more accurate, so the time statistics and the ensemble statistics will be the same.

Q6: BANDWIDTH OF THE TRANSMITTED SIGNAL

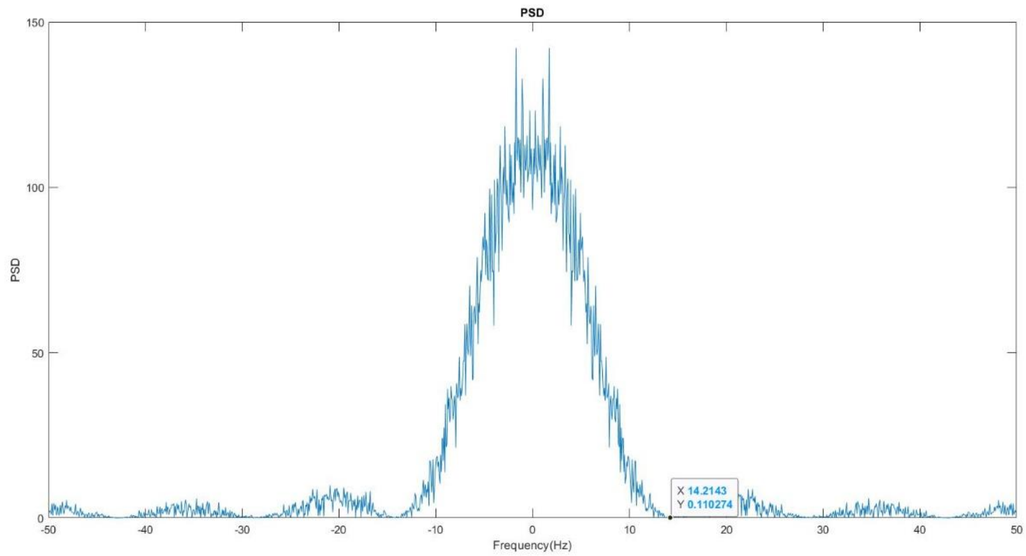

Figure 10: PSD of Polar NRZ

Theoretically, the bandwidth is approaching infinity, but we can take approximately till the first null which is laying at 1/bit duration ($\frac{1}{T_b} = 14.2857\ Hz$).

From the graph, we can find that we achieve the required bandwidth at **x = 14.21**, y approximately equal zero.

So, the bandwidth of the signal at the baseband = **14.21 Hz**.

3. POLAR RETURN TO ZERO

GENERATING THE REALIZATIONS

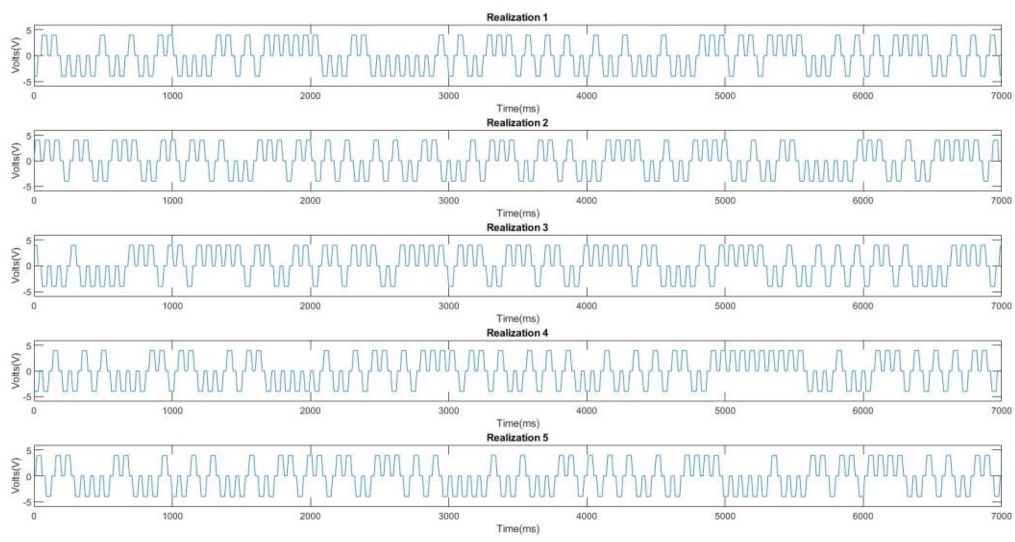

Figure 11: realizations of Polar RZ line code

Q1: STATISTICAL MEAN

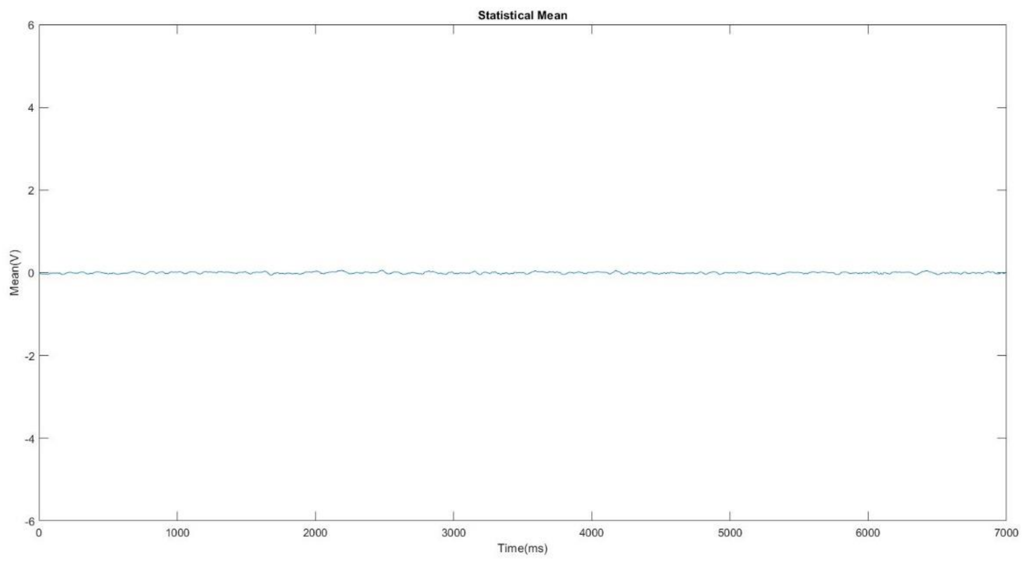

Figure 12: statistical mean of Polar RZ line code

From figure 12, we can observe that statistical mean is constant and approximately equal to **0**.

$E\big(X(t)\big) = \left(\frac{A}{2} \times \frac{4}{7}\right) + \left(\frac{-A}{2} \times \frac{4}{7}\right) = 0$

Q3: ENSEMBLE AUTOCORREALTION FUNCTION

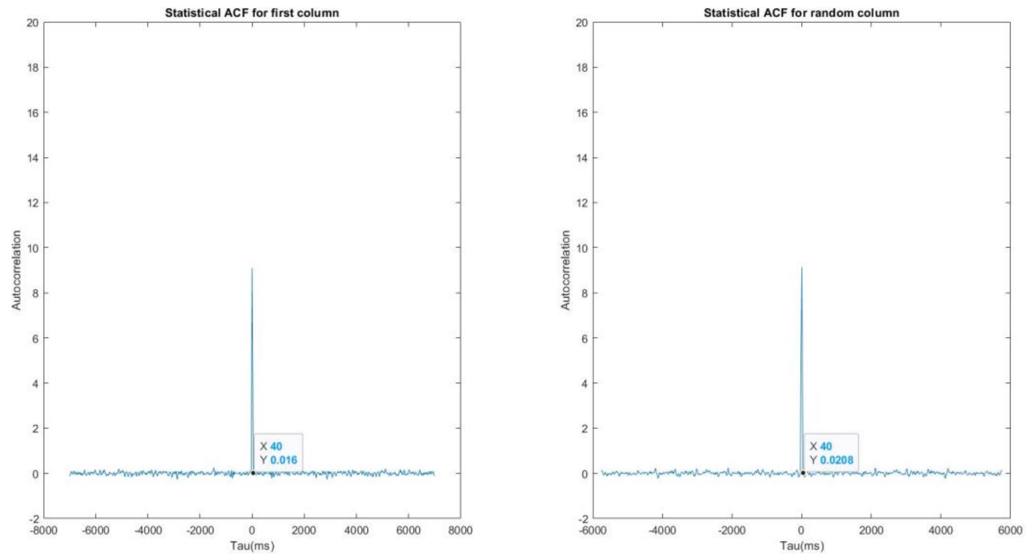

Figure 13: statistical autocorrelation of two different times

The autocorrelation indicates how much the signal at t1 is correlated to the signal at t2, hence the autocorrelation must be maximum at time difference (τ) equals to zero because the correlation of the signal and itself at any time t must be the highest comparing with the signal at time t and the signal at any other time instance, and by increasing the time difference (τ), the autocorrelation decreases till become approximately constant

$R_x(\tau) = \begin{cases} A^2 \times \frac{4}{7}\left(1 - \frac{|\tau|}{T_b}\right), & |\tau| \leq T_b \\ 0, & |\tau| > T_b \end{cases}$, where A = 4. $R_x(0) = A^2 \times \frac{4}{7} = 9.142857143.$

The graph shows that the autocorrelation of two different times is the same which means the **autocorrelation is not function in time absolute but function in time difference**, and its maximum at $\boldsymbol{\tau = 0}$ and starts to decreases until become constant equal to **0** from $\boldsymbol{\tau = 40}$ which is the time bit duration ($\frac{4}{7}$ ×Tb) as we choose to send the data in four samples and return to zero in the remaining three samples in each bit.

Q2: STATIONARY CHECK

After calculating the statistical mean and statistical autocorrelation, we can now get back to check the stationarity of the process, as the WSS process has constant statistical mean, and the statistical autocorrelation is only function in the time difference.

1. Back to figure 12, we can find that the mean is constant over the time.
2. From figure 13, we compared the statistical autocorrelation at only two different times just to simplify instead of comparing with all times, so we can find that the autocorrelation is approximately the same in the two graphs.

From (1) and (2), the conditions of the WSS process are verified, so we can say that a **polar return to zero process is a WSS process**.

Q4: TIME MEAN AND AUTOCORRELATION OF ONE WAVEFORM

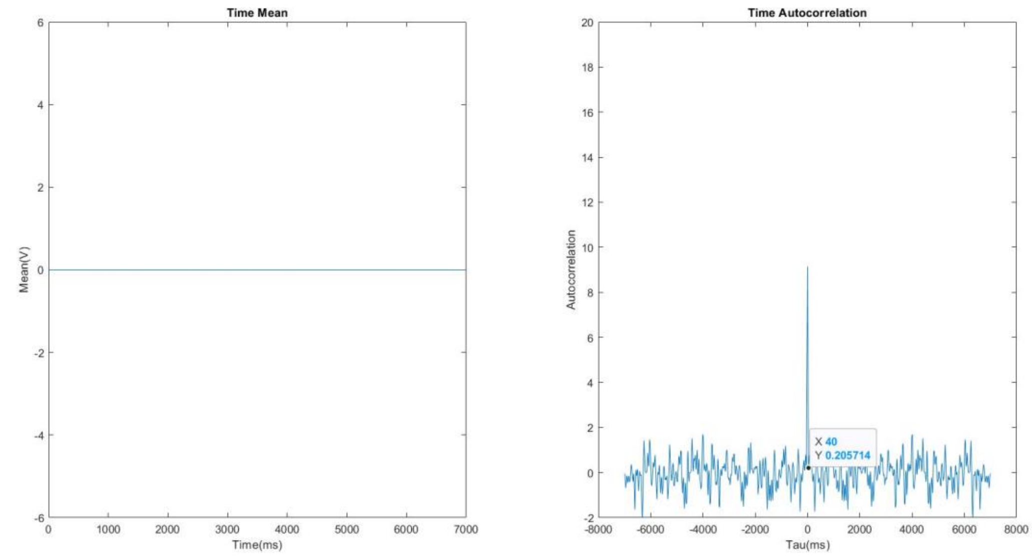

Figure 14: time mean and time autocorrelation of one waveform

The time mean is **constant** and approximately equal 0.

The time autocorrelation starts to be **constant** at Tb*4/7 as we take only **four** samples from the seven samples in the bit.

Q5: ERGODIC CHECK

To say that a random process is Ergodic process the statistical mean and the time mean of any waveform must be equal, also the statistical autocorrelation and the time autocorrelation of any waveform must be equal, in general the statistics over the realizations must be equal the statistics over time of any waveform.

1. Comparing the mean in figure (12) and (14), the statistical mean is equal to the time mean which is equal to 0.
2. Comparing the autocorrelation in figure (13) and (14), the statistical autocorrelation is equal to the time autocorrelation which is approximately varying around 0.

From (1) and (2), the conditions of the Ergodic process are verified, so we can say that a **polar return to zero process is an Ergodic process**.

Knowing that by increasing the number of samples, the accuracy will increase, and the time statistics will be closer to the statistics of the whole ensemble.
Also, by increasing the number of realizations, the statistics will be more accurate, so the time statistics and the ensemble statistics will be the same.

Q6: BANDWIDTH OF THE TRANSMITTED SIGNAL

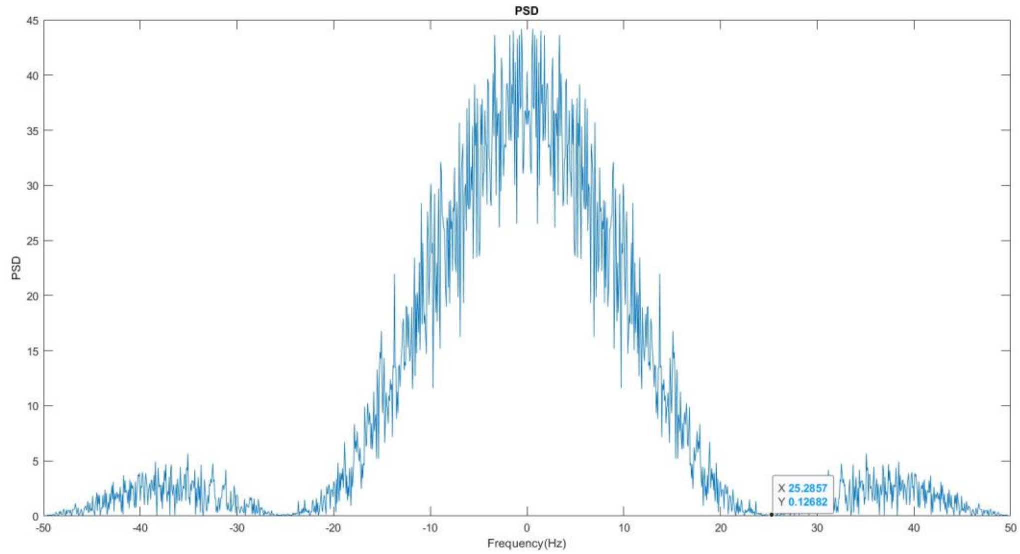

Figure 15: PSD of Polar RZ

Theoretically, the bandwidth is approaching infinity, but we can take approximately till the first null which is laying at 7/(4*bit duration) then ($\frac{7}{4T_b} = 25$ Hz).

From the graph, we can find that we achieve the required bandwidth at **x = 25.28**, y approximately equal zero.

So, the bandwidth of the signal at the baseband = **25.28 Hz**.