Cairo University

Faculty of Engineering

Electronics and Electrical Communication Department

Digital Communication Project 1

Submitted by:

Name: Ahmed Mokhtar Mahfouz Emam

Sec: 1

BN: 31

Email: Ahmed.mokhtar5483@gmail.com

Table of Contents

Table of Figures

Generating the data

```
A=5;

n = 100;                                % However many numbers you want.
ret_0=4;                                % to knows the number of ones in RZ waveform
1111000
Tx_out = zeros(500,7*n,3);
out_data1 = zeros(500,7*n);             % Initialize output data for Non-Return Zero
Waveforms
out_data2 = zeros(500,7*n);             % Initialize output data for Return Zero
Waveforms
del_no= zeros(500,1);                   % Initialize columns vector for delays

for i = 1:500       % Generate the Data
    Data = randi([0, 1], [1, n]);       % Generate random int data 0 or 1 with length
of n
    Tx=repmat(Data,7,1);                % Repete each opservation 7 times
    del_no(i) = randi([1 7]);
    delay = repmat(randi([0 1]),del_no(i),1);
    Tx_delayed=[delay ; Tx(:)];
    Tx_final = Tx_delayed(1:700,1);
    out_data1(i,:) = Tx_final';         % the output data for Non Return Zero
Waveforms 500*700
    %   -------------------------------
    Tx = A*((2*Tx)-1);
    Tx(1+ret_0:7,:)=zeros(7-ret_0,100); % map deta bits 0 to -A and 1 to A
    delay = A*((2*delay)-1);            % map delay bits 0 to -A and 1 to A
    Tx_delayed=[delay ; Tx(:)];         % Join Data and delay (delay_bits + 700)*500
    Tx_final = Tx_delayed(1:700,1);     % Take the first 700*500
    out_data2(i,:) = Tx_final';         % the output data for Return Zero Waveforms
500*700
end
% Adding the 3 linecodes in 3D matrix (Tx_out) after mapping them
Tx_out(:,:,1) =  A*out_data1;            %1- Unipolar Signaling ->map deta bits 0 to
-A and 1 to A
Tx_out(:,:,2) =  A*((2*out_data1)-1);    %2- Polar Non-Return to Zero ->map delay
bits 0 to -A and 1 to A
Tx_out(:,:,3) =  out_data2;              %3- Polar Return to Zero

for i = 1:500 % handel the delay in Polar_return_zero to follow the sequance 1111000
    if(del_no(i) < ret_0)
        Tx_out(i,:,3)=[zeros(1,del_no(i)) out_data2(i,(1+del_no(i)):700)];
    else
        start =del_no(i)-ret_0+1;
        Tx_out(i,1:del_no(i),3)=[out_data2(i,1:start) zeros(1,del_no(i)-(start))];
    end
end
```

Processing on Unipolar Signaling

```
stat_mean = zeros(1,700,3);

for i = 1:700
    stat_mean(i,:,1) = sum(Tx_out(:,i,1))/500;
end


% Time Mean for one wave form
time_mean = zeros(1,3);
i = randi([0 500]);
time_mean(1,1) = sum(Tx_out(i,:,1))/700;


%Time Mean for all WaveForms
Time_mean_all=zeros(500,3);
for i=1:500
    Time_mean_all(i,1)=sum(Tx_out(i,:,1))/700;
end


% Statistical ACF
stat_acf =zeros(1,700,3);
for i=1:700
    stat_acf(1,i,1)=sum(Tx_out(:,1,1).*Tx_out(:,i,1))/500;
end

% Time ACF
TimeAcf=zeros(700,3);
TimeAcf(1,1)=Tx_out(1,1,1)*Tx_out(1,1,1)*(1/700);
for i=0:699
    for j=1:700-i
        t1=Tx_out(1,j,1);
        t2=Tx_out(1,j+i,1);
        TimeAcf(i+1,1)=TimeAcf(i+1,1)+t1*t2*(1/(700-i));
    end
end

% Bandwidth
PSD = zeros(1,690,3);
PSD(:,:,1) = fft(stat_acf(:,11:700,1));
N = length(stat_acf) ;
M=length(PSD);
freq_axis = ((-0.5*M:0.5*M-1)*100)/M;
```

Output for Unipolar Signaling

```
figure

subplot(3,2,1)

plot(linspace(0,7000,700),Tx_out(1,1:700,1),'r','LineWidth',1,'MarkerSize',50)
title('Unipolar NonReturnZero')
xlabel('time T(msec)')
ylabel('Amp (Volts)')
ylim([0 6])
grid on
subplot(3,2,2)
plot(freq_axis,abs(fftshift(PSD(:,:,1))),'r','LineWidth',1,'MarkerSize',50)
title('Unipolar NonReturnZero BW')
xlabel('Freq f(Hz)')
ylabel('Power Spectral Density')
grid on
subplot(3,2,3)
plot(linspace(1,7000,700),stat_mean(:,:,1),'r','LineWidth',1,'MarkerSize',50)
title('Unipolar NRZ Statistical Mean')
xlabel('time T(msec)')
ylabel('Statistical Mean')
ylim([0 5])
grid on
subplot(3,2,4)
plot(linspace(0,10*300,300),stat_acf(:,1:300,1),'r',...
    -
linspace(0,10*300,300),stat_acf(:,1:300,1),'r','LineWidth',1,'MarkerSize',50)
title('Unipolar NRZ Statistical ACF')
xlabel('time T(msec)')
ylabel('Statistical ACF')
grid on
subplot(3,2,5)
plot(linspace(1,500,500),Time_mean_all(:,1),'r','LineWidth',1,'MarkerSize',50)
title('Unipolar NRZ Time Mean')
xlabel('No of waveform')
ylabel('Time Mean')
ylim([0 5])
grid on
subplot(3,2,6)
plot(linspace(1,300,300),TimeAcf(1:300,1),'r',-
linspace(1,300,300),TimeAcf(1:300,1),'r','LineWidth',1,'MarkerSize',50)
title('UniPolar NRZ Time AutoCorrelation Function')
xlabel('Samples')
ylabel('Time ACF')
grid on
```

Figure of Unipolar Non-return Zero

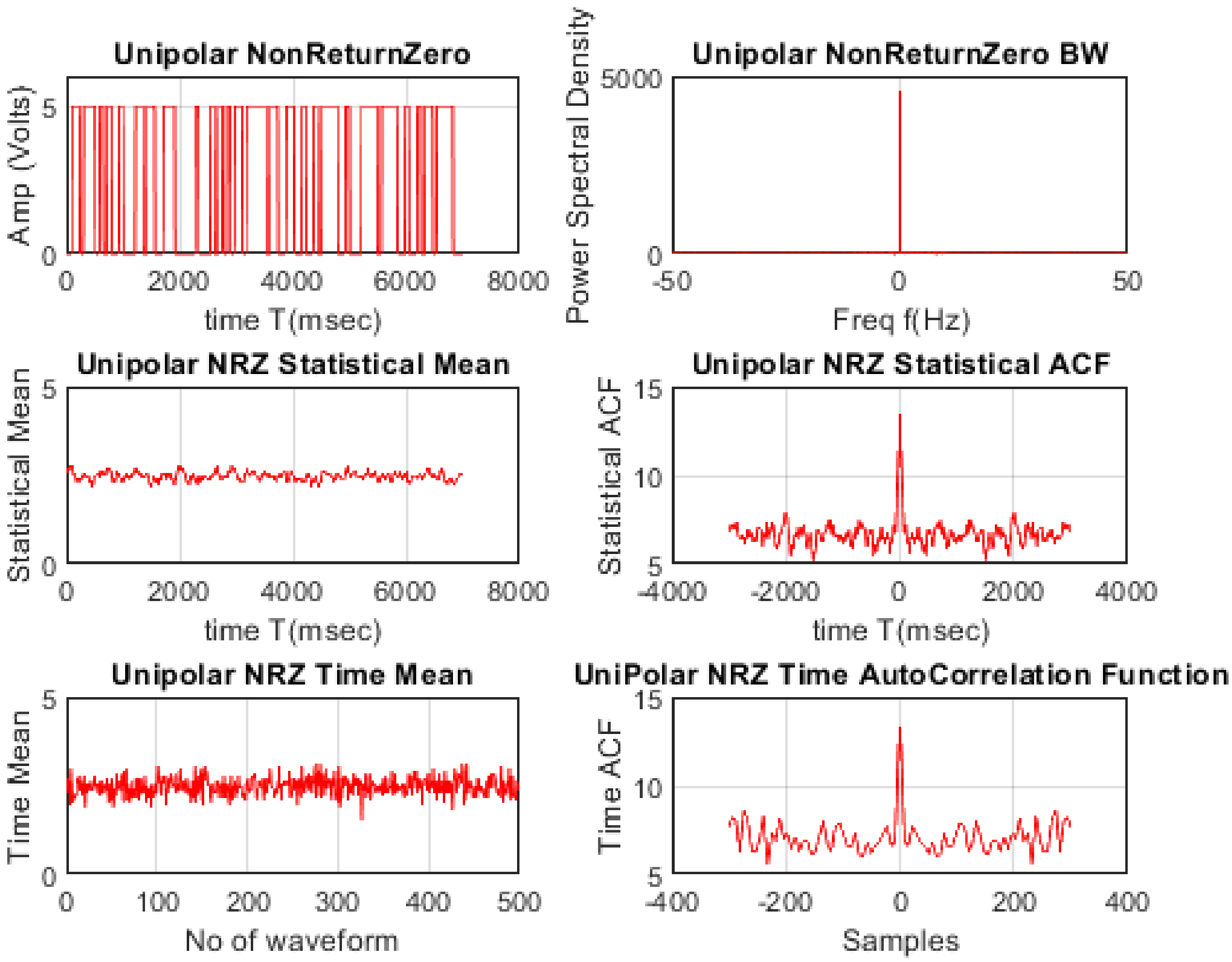

Figure 1:Unipolar NRZ

Comment:

- ❖ The statical mean is varying around (2.24 -> 2.78).
- ❖ The Time mean is varying around (2.1 -> 2.9).
- ❖ The waveform in frequency domain for this line code is an impulse signal.

Processing Polar Non-Return to Zero

```
for i = 1:700
    stat_mean(i,:,2) = sum(Tx_out(:,i,2))/500;
end

% Time Mean for one wave form
i = randi([0 500]);
time_mean(1,2) = sum(Tx_out(i,:,2))/700;

%Time Mean for all WaveForms
for i=1:500
    Time_mean_all(i,2)=sum(Tx_out(i,:,2))/700;
end

% Statistical ACF
for i=1:700
    stat_acf(1,i,2)=sum(Tx_out(:,1,2).*Tx_out(:,i,2))/500;
end

% Time ACF
TimeAcf(1,2)=Tx_out(1,1,2)*Tx_out(1,1,2)*(1/700);
for i=0:699
    for j=1:700-i
        t1=Tx_out(1,j,2);
        t2=Tx_out(1,j+i,2);
        TimeAcf(i+1,2)=TimeAcf(i+1,2)+t1*t2*(1/(700-i));
    end
end

% Bandwidth
PSD(:,:,2) = fft(stat_acf(:,11:700,2));
```

Output for Polar Non-Return to Zero

```
figure

subplot(3,2,1)

plot(linspace(0,7000,700),Tx_out(1,1:700,2),'y','LineWidth',1,'MarkerSize',1)
title('Polar NonReturnZero')
xlabel('time T(msec)')
ylabel('Amp (Volts)')
ylim([-6 6])
grid on
subplot(3,2,2)
plot(freq_axis,fftshift(abs(PSD(:,:,2))),'y','LineWidth',1,'MarkerSize',50)
title('Polar NonReturnZero BW')
xlabel('Freq f(Hz)')
ylabel('Power Spectral Density')
% ylim([0 400])
grid on
subplot(3,2,3)
plot(linspace(1,7000,700),stat_mean(:,:,2),'y','LineWidth',1,'MarkerSize',50)
title('Polar NRZ Statistical Mean')
xlabel('time T(msec)')
ylabel('Statistical Mean')
ylim([-2 2])
grid on
subplot(3,2,4)
plot(linspace(0,10*300,300),stat_acf(:,1:300,2),'y',...
    -
linspace(0,10*300,300),stat_acf(:,1:300,2),'y','LineWidth',1,'MarkerSize',50)
title('Polar NRZ Statistical ACF')
xlabel('time T(msec)')
ylabel('Statistical ACF')
grid on
subplot(3,2,5)
plot(linspace(1,500,500),Time_mean_all(:,2),'y','LineWidth',1,'MarkerSize',50)
title('Polar NRZ Time Mean')
xlabel('No of waveform')
ylabel('time Mean')
ylim([-2 2])
grid on
subplot(3,2,6)
plot(linspace(1,300,300),TimeAcf(1:300,2),'y',-
linspace(1,300,300),TimeAcf(1:300,2),'y','LineWidth',1,'MarkerSize',50)
title('Polar NRZ Time AutoCorrelation Function')
xlabel('Samples')
ylabel('Time ACF')
grid on
```

Figure of Polar Non-Return to Zero

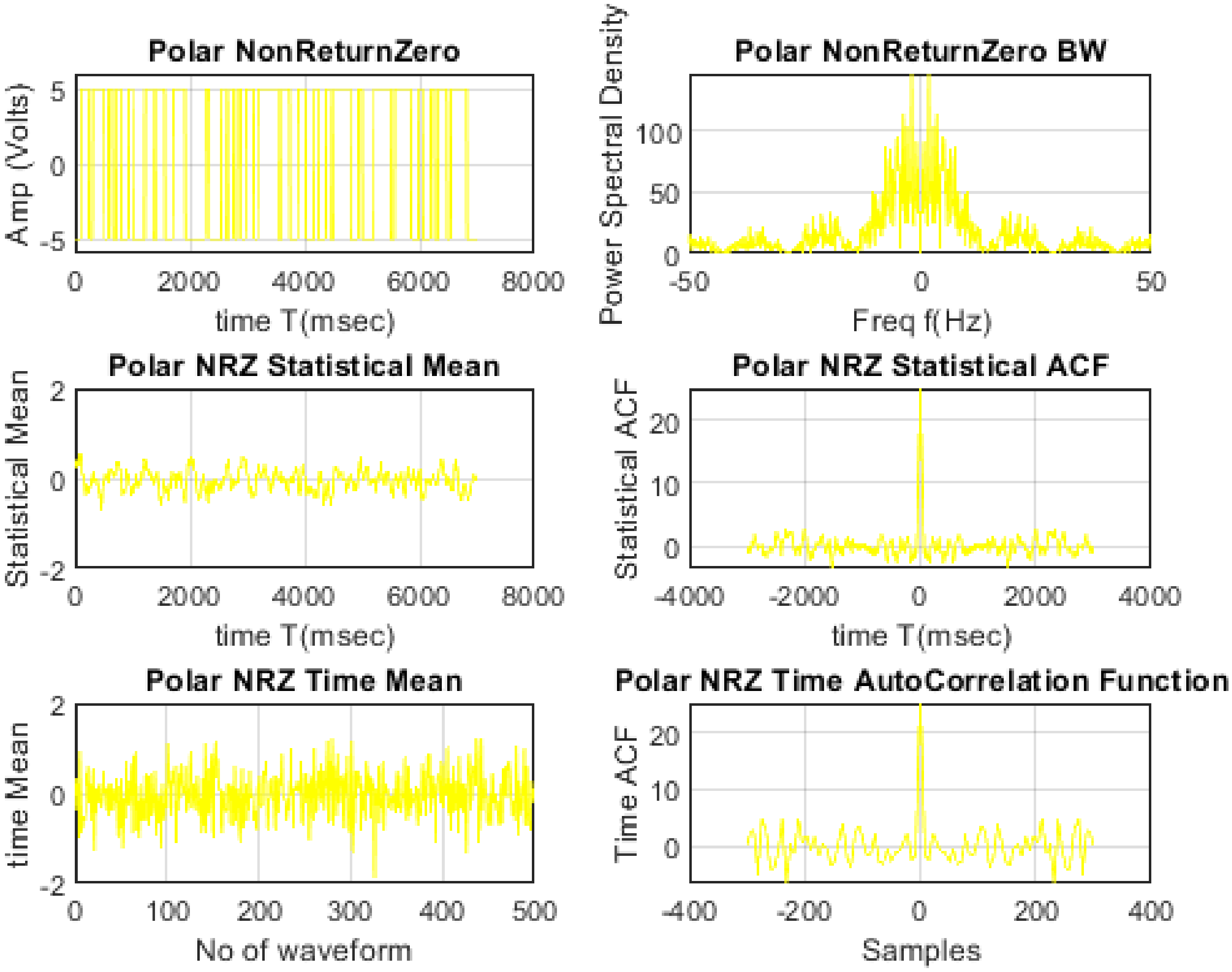

Figure 2:Polar NRZ

Comment:

- The statical mean is varying around (-0.5 -> 0.5).
- The Time mean is varying around (-0.8 -> 0.9).
- The bandwidth is 20 Hz.

Processing on Polar Return to Zero

```
for i = 1:700
    stat_mean(i,:,3) = sum(Tx_out(:,i,3))/500;
end

% Time Mean for one wave form
i = randi([0 500]);
time_mean(1,3) = sum(Tx_out(i,:,3))/700;

%Time Mean for all WaveForms
for i=1:500
    Time_mean_all(i,3)=sum(Tx_out(i,:,3))/700;
end

% Statistical ACF
for i=1:700
    stat_acf(1,i,3)=sum(Tx_out(:,1,3).*Tx_out(:,i,3))/500;
end

% Time ACF
TimeAcf(1,3)=Tx_out(1,1,3)*Tx_out(1,1,3)*(1/700);
for i=0:699
    for j=1:700-i
        t1=Tx_out(1,j,3);
        t2=Tx_out(1,j+i,3);
        TimeAcf(i+1,3)=TimeAcf(i+1,3)+t1*t2*(1/(700-i));
    end
end

% Bandwidth
PSD(:,:,3) = fft(stat_acf(:,11:700,3));
```

Output for Polar Return to Zero

```
figure

subplot(3,2,1)

plot(linspace(0,7000,700),Tx_out(1,1:700,3),'k','LineWidth',1,'MarkerSize',1)
title('Polar ReturnZero')
xlabel('time T(msec)')
ylabel('Amp (Volts)')
ylim([-6 6])
grid on
subplot(3,2,2)
plot(freq_axis,fftshift(abs(PSD(:,:,3))),'k','LineWidth',1,'MarkerSize',50)
title('Polar ReturnZero BW')
xlabel('Freq f(Hz)')
ylabel('Power Spectral Density')
grid on
subplot(3,2,3)
plot(linspace(1,7000,700),stat_mean(:,:,3),'k','LineWidth',1,'MarkerSize',50)
title('Polar RZ Statistical Mean')
xlabel('time T(msec)')
ylabel('Statistical Mean')
ylim([-2 2])
grid on
subplot(3,2,4)
plot(linspace(0,10*300,300),stat_acf(:,1:300,3),'k',...
    -
linspace(0,10*300,300),stat_acf(:,1:300,3),'k','LineWidth',1,'MarkerSize',50)
title('Polar RZ Statistical ACF')
xlabel('time T(msec)')
ylabel('Statistical ACF')
grid on
subplot(3,2,5)
plot(linspace(1,500,500),Time_mean_all(:,3),'k','LineWidth',1,'MarkerSize',50)
title('Polar RZ time Mean')
xlabel('No of waveform')
ylabel('Time Mean')
ylim([-2 2])
grid on
subplot(3,2,6)
plot( linspace(1,300,300),TimeAcf(1:300,3),'k',-
linspace(1,300,300),TimeAcf(1:300,3),'k','LineWidth',1,'MarkerSize',50)
title('Polar RZ Time AutoCorrelation Function')
xlabel('Samples')
ylabel('Time ACF')
grid on
```

Figure of Polar Return to Zero

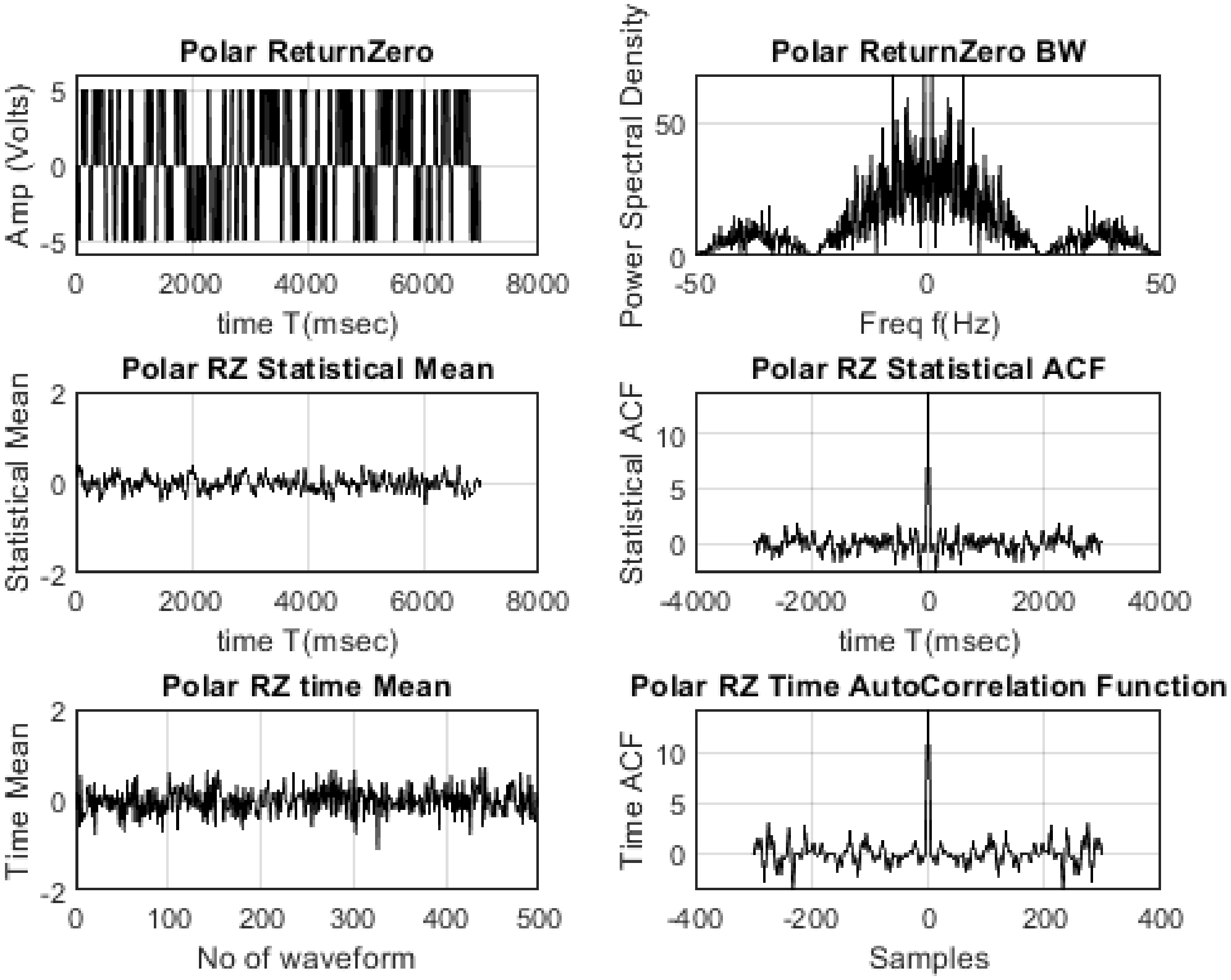

Figure 3:Polar RZ

Comment:

- ❖ The statical mean is varying around (-0.4 -> 0.5).
- ❖ The Time mean is varying around (-0.5 -> 0.5).
- ❖ The bandwidth is 50 Hz.

Conclusion

- This random process is stationary for all line codes due to it satisfied the two condition for the WSS:

  1) The mean doesn’t depend on time.
  2) The Statistical ACF depends on τ only.

- This random process is Ergodic for all line codes except Polar RZ due to it satisfies the two condition:

  1) Time mean is close enough to the statistical mean.
  2) The Time Auto Correlation Function for one realization is close enough to the Auto Correlation Function for ensemble.