

ΤΗΛ 301 – ΤΗΛΕΠΙΚΟΙΝΩΝΙΑΚΑ ΣΥΣΤΗΜΑΤΑ 1

ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2019-2020

Εργαστηριακή Άσκηση 2

ΜΑΝΕΣΗΣ ΑΘΑΝΑΣΙΟΣ

ΑΜ:2014030061

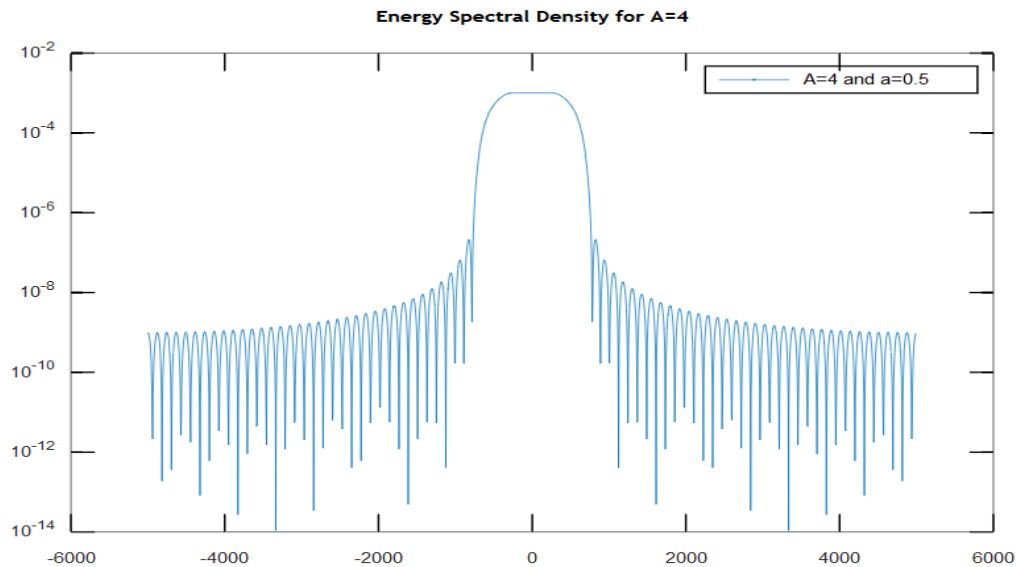


Σχολή
Ηλεκτρολόγων
Μηχανικών &
Μηχανικών
Υπολογιστών

A. Μελέτη φασματικού περιεχομένου PAM κυματομορφών

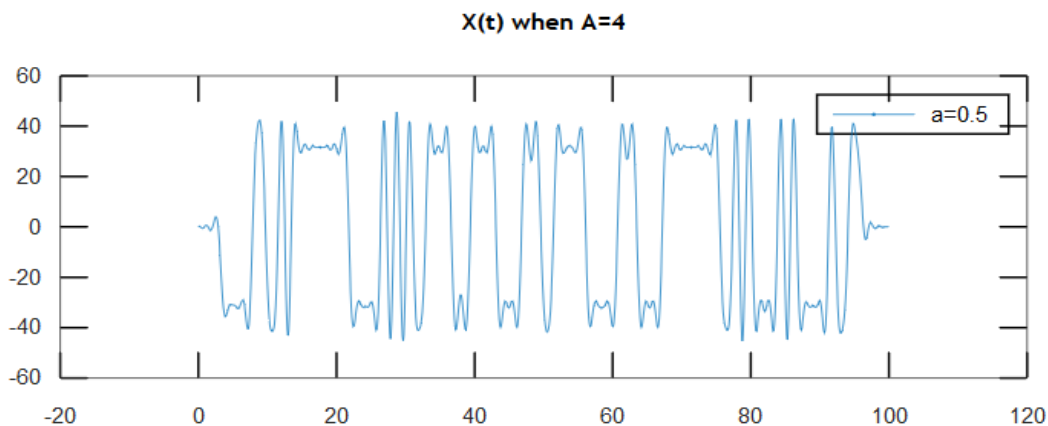
A1. Δημιουργία αποκομμένων παλμών SRRC

Δημιουργήθηκε παλμός SRRC για $T=0.001$, $\text{over}=10$, $A=4$ και $a=0.5$. Με την χρήση των `fftshift`, `fft` υπολογίστηκε ο μετασχηματισμός Fourier και σχεδιάστηκε η φασματική πυκνότητα ενέργειας σε ημιλογαριθμική κλίμακα.



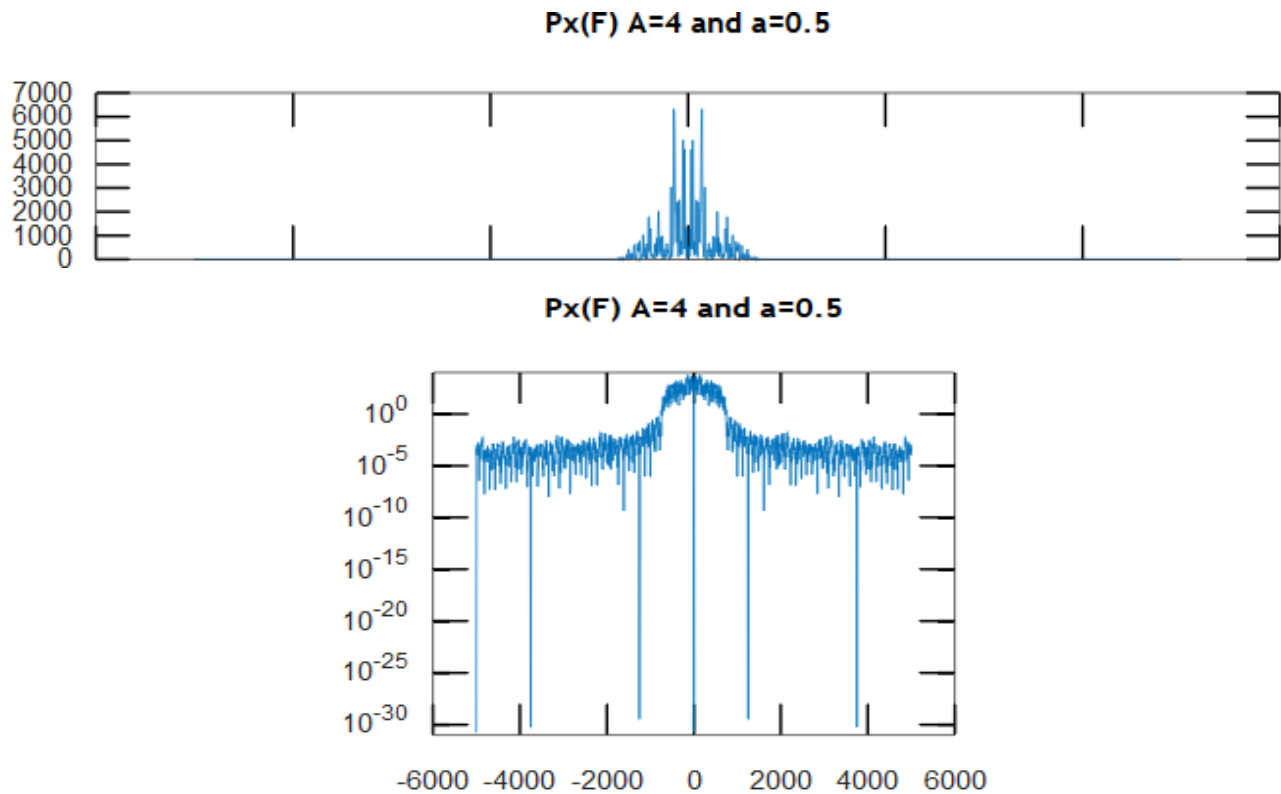
A2. Μετατροπή δυαδικής σειράς σε 2-PAM

Το ερώτημα βασίζεται στις γνώσεις της προηγούμενης άσκησης. Δημιουργήθηκαν ανεξάρτητα και ισοπίθανα `bits` χρησιμοποιώντας την συνάρτηση `bits_to_2PAM` για να απεικονίσουμε τα `bits` σε συμβολα X_n . Τέλος έπρεπε να μετατοπιστεί κατάλληλα η συνάρτηση ϕ , του A1 ερωτήματος, ώστε να παραχθεί το ζητούμενο σήμα.

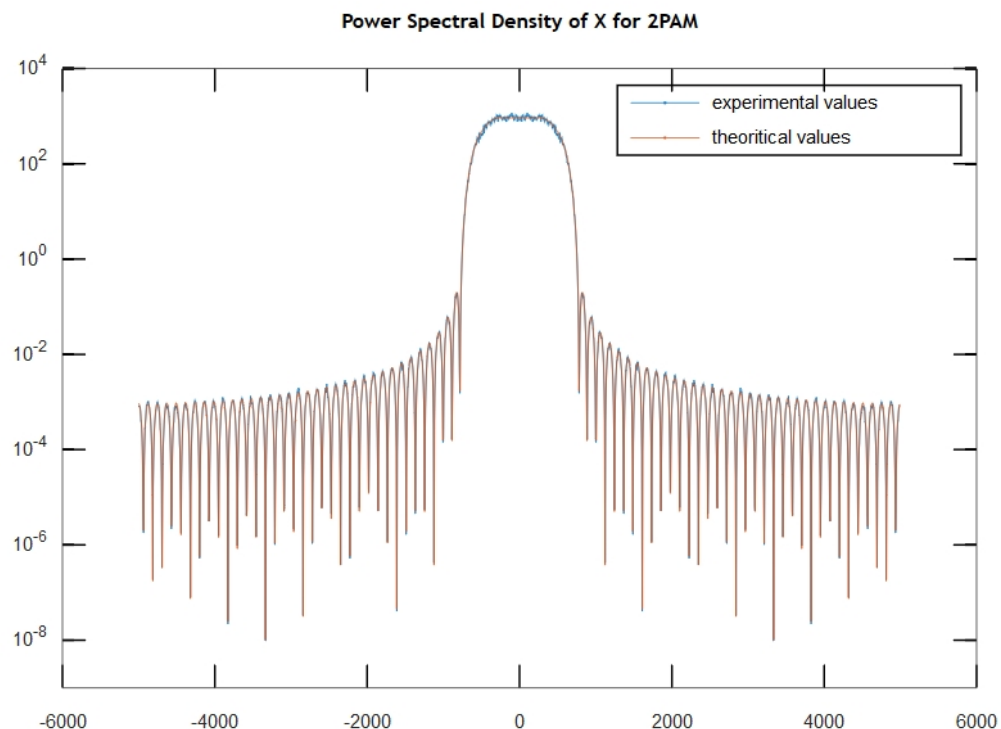


A3. Υπολογισμός περιοδογράμματος και φασματικής πυκνότητας ισχύος για κωδικοποίηση 2-PAM

Η κατανομή ισχύος στο πεδίο της συχνότητας απεικονίζεται μέσω του περιοδογράμματος. Ακολουθούν οι κυματομορφές που ζητήθηκαν με χρήση των συναρτήσεων plot και semilogy.



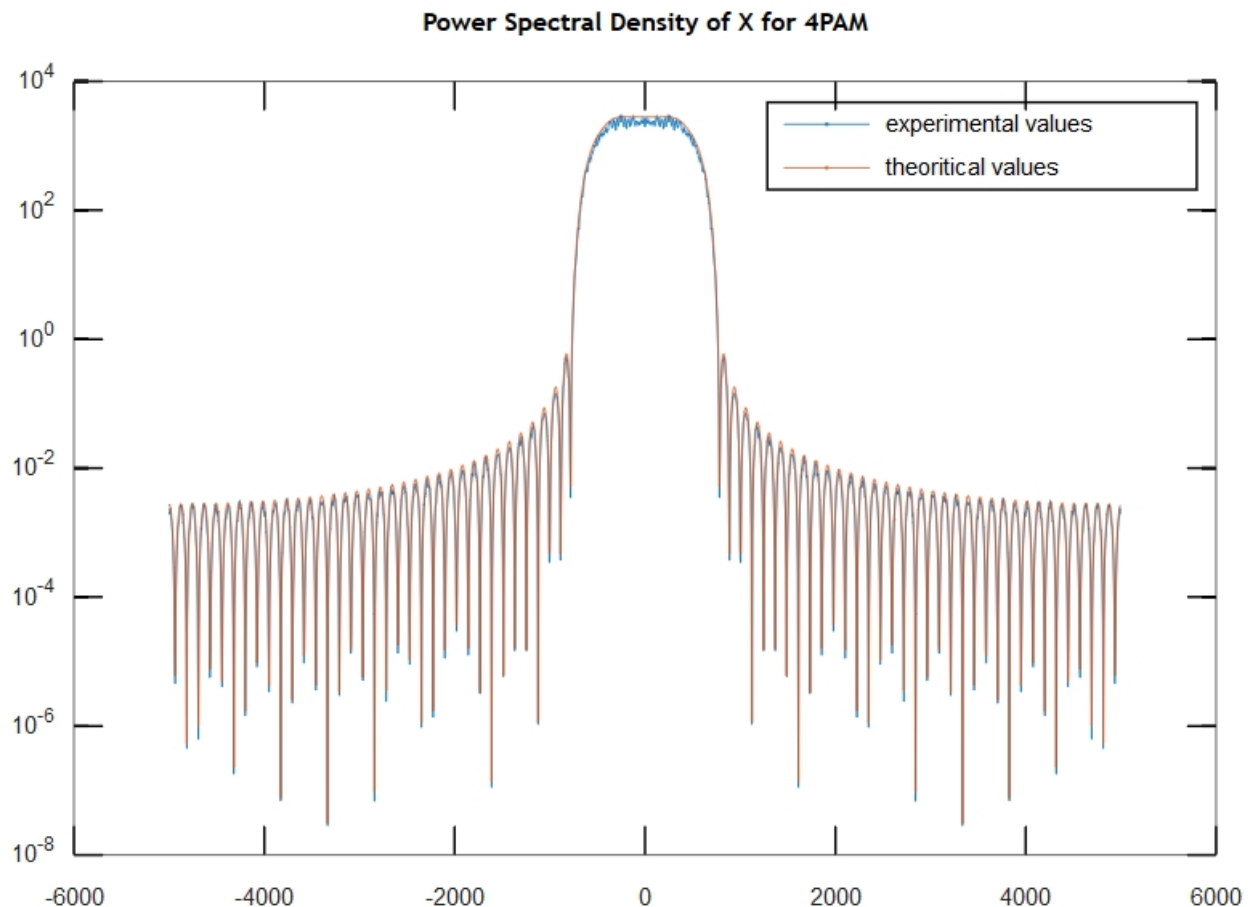
Η κυματομορφή που ακολουθεί είναι ημιλογαριθμικής κλίμακας με την θεωρητική και την πειραματική προσέγγιση να περιλαμβάνονται.



Παρατηρούμε ότι η πειραματική προσέγγιση ακολουθεί την θεωρητική. Όσο αυξάνονται οι επαναλήψεις του πειράματος η μέση τιμή θα προσεγγίζει την ιδανική προσέγγιση. Σε θεωρητικό επίπεδο επίσης λαμβάνουμε υπόψιν ότι τα σύμβολα είναι άπειρα, άρα με την αύξηση του N τα αποτελέσματα τείνουν στη θεωρητική προσέγγιση. Στο πειραματικό σκέλος παρατηρείται ότι μετά από αύξηση συμβόλων βελτιώθηκε περισσότερο η προσέγγιση σε σχέση με την αύξηση των επαναλήψεων.

A4. Υπολογισμός περιδιογράμματος και φασματικής πυκνότητας ισχύος για κωδικοποίηση 4-PAM

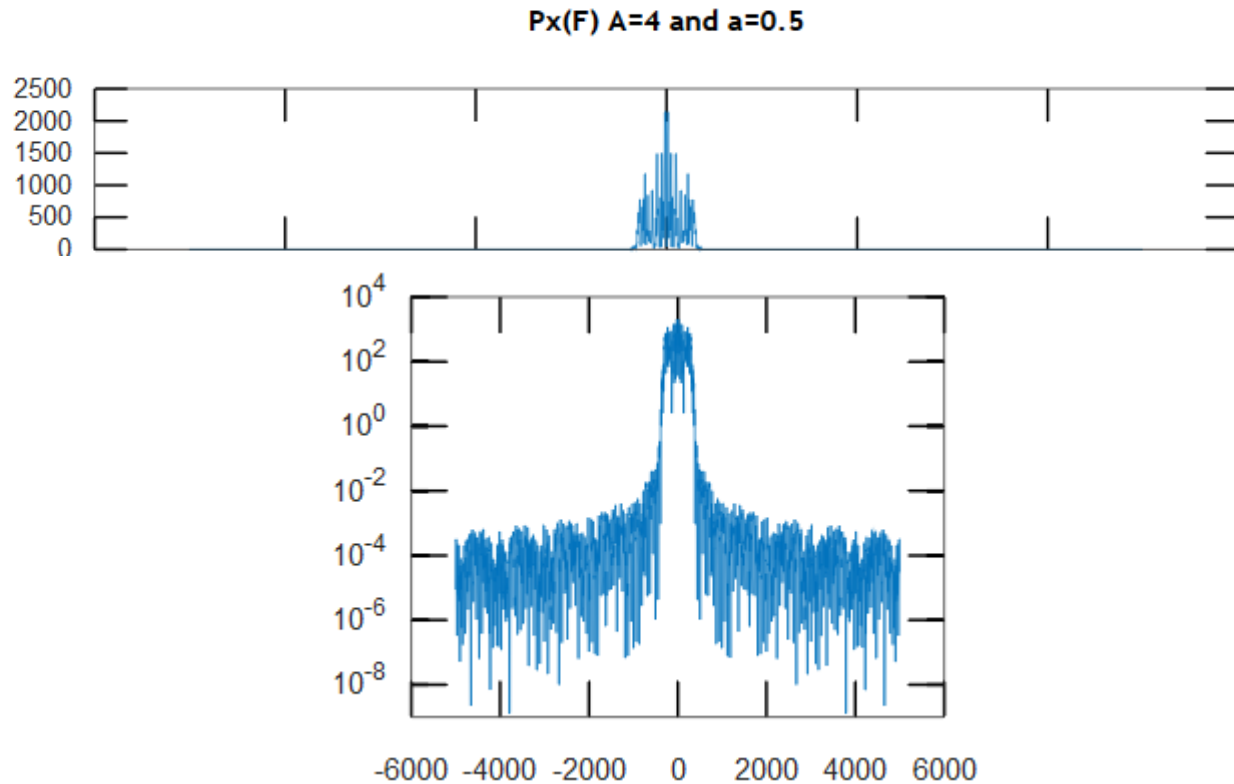
Όμοια με τα ερωτήματα A2, A3 αλλά με την χρήση της συνάρτησης `bits_to_4PAM`. Η συνάρτηση αυτή μετατρέπει ακολουθίες `bits` σε `+3,+1,-1,-3`.



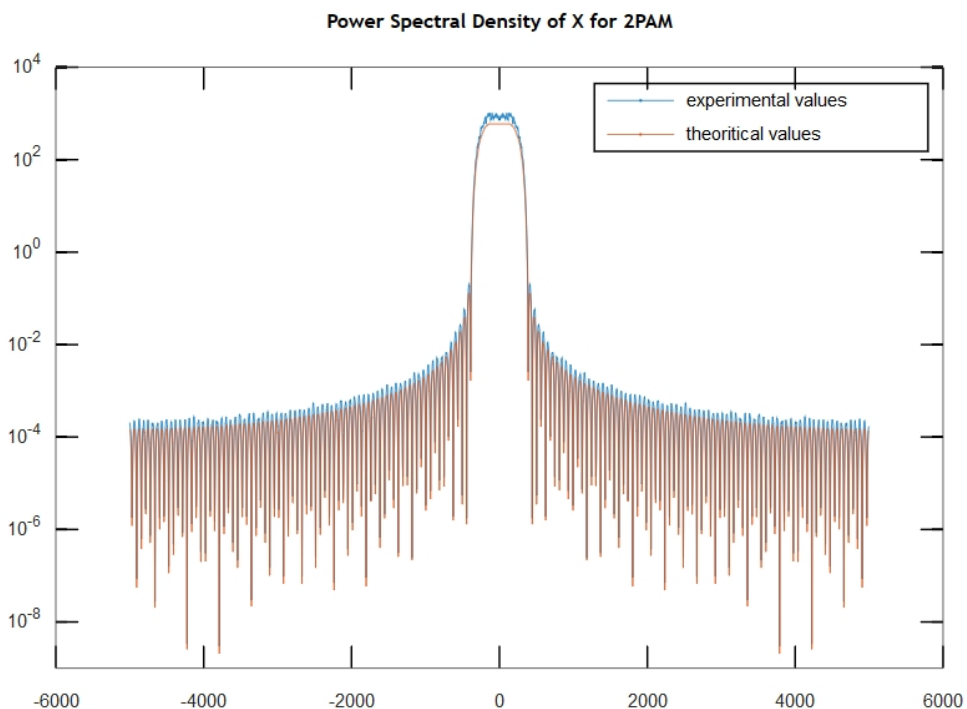
Συγκρίνοντας τις φασματικές πυκνότητες ισχύος των υλοποιήσεων 4-PAM και 2-PAM βλέπουμε ότι το εύρος φάσματος είναι το ίδιο και αυτό οφείλεται στην χρήση της ίδιας συνάρτησης με καθορισμένο εύρος φάσματος, αλλά το πλάτος είναι μεγαλύτερο στην 4PAM λόγω της περισσότερη ισχύ που απαιτείται για να σταλθούν τα σύμβολα.

A5. Υπολογισμός περιοδογράμματος και φασματικής πυκνότητας ισχύος για διπλάσια περίοδο συμβόλου

Όμοια με τα προηγούμενα ερωτήματα για διπλάσια περίοδο συμβόλου.



Διπλασιάζοντας την περίοδο συμβόλου, υποδιπλασιάζεται το εύρος φάσματος. Ακολουθεί η κυματομορφή θεωρητικής-πειραματικής προσέγγισης σε ημιλογαριθμική κλίμακα.



A6. Σύγκριση αποτελεσμάτων

1. Υποθέτοντας ότι λαμβάνουμε σύμβολα με το ίδιο T και ότι η επεξεργασία σήματος είναι αμνηλεής θα επιλέγαμε την 4-PAM επειδή μεταφέρει 2-bits ανά σύμβολο και στο ίδιο bandwidth μεταδίδεται ταχύτερα η πληροφορία.
2. Επιλέγω $T'=2T$, παρόλο που απαιτείται μεγαλύτερη ενέργεια για μετάδοση, το bandwidth υποδιπλασιάζεται.

B.1

$$X(t) = \sum_{n=-\infty}^{\infty} X_n \phi(t - nT)$$

$$E[X_n] = 0$$

$$Y(t) = X(t) \cos(2\pi f_0 t + \Theta)$$

$$E[X_n^2] = \sigma^2 X \quad \Theta \text{ ομοιόμορφα κατανεμημένη } [0, 2\pi)$$

$$E_{X\Theta} m_1 = E[Y(t)] = E[X(t) \cdot \cos(2\pi f_0 t + \Theta)]$$

$$E[X_n] = \sum_n X_n \phi(t - nT) = \sum_n E[X_n \phi(t - nT)] = \sum_n \phi(t - nT) E[X_n] = 0$$

$$R_{XX}(t+z, t) = E[X(t+z) X(t)]$$

$$= E\left[\sum_n X_n \phi(t+z-nT) \cdot \sum_n X_n \phi(t-nT)\right]$$

$$= E\left[\sum_n X_n \phi(t+z-nT) \cdot X_n \phi(t-nT)\right]$$

$$= \sum_n E[X_n \phi(t+z-nT) \cdot X_n \phi(t-nT)]$$

$$= \sum_n E[X_n^2 \phi(t+z-nT) \phi(t-nT)]$$

$$= \sum_n \phi(t+z-nT) \cdot \phi(t-nT) \cdot \sigma^2 X$$

$$\text{Επομένως } E[Y(t)] = E[X(t) \cdot \cos(2\pi f_0 t + \Theta)] = E[X(t)] \cdot E[\cos(2\pi f_0 t + \Theta)] = 0$$

$$R_{YY}(t+z, t) = E[Y(t+z) Y(t)]$$

$$= E[X(t+z) \cdot \cos(2\pi f_0(t+z) + \Theta) \cdot X(t) \cdot \cos(2\pi f_0 t + \Theta)]$$

$$= E[X(t+z) X(t)] \cdot E[\cos(2\pi f_0(t+z) + \Theta) \cdot \cos(2\pi f_0 t + \Theta)]$$

$$= R_{XX}(t+z, t) \cdot E[\cos(2\pi f_0 t) \cdot \cos(2\pi f_0(t+z) + \Theta)]$$

$$= R_{XX}(t+z, t) \cdot E\left[\frac{1}{2} \cos(2\pi f_0 t) + \cos(2\pi f_0(t+z) + 2\Theta)\right]$$

B.2

$$E[Y(t)] = 0 \rightarrow Y(t) \text{ ααθερη}$$

οπως $R_{yy}(t+z, t)$ ειναι εφ'αραται για το ειναι το t
 $\rightarrow \Delta \epsilon \eta$ ειναι ααθερη και ανω ευρεα ευρεα.

$$E[Y(t)] = 0 \Rightarrow E[Y(t+T)] = 0$$

$$R_{yy}(t+z+T, t+T) = \frac{1}{2} \cos(2\pi f_0 z) \cdot \sum_n G_x^2 \phi(t+z+T-nT) \cdot \phi(t+T-nT)$$

$$= \frac{1}{2} \cos(2\pi f_0 z) \cdot \sum_n G_x^2 \phi(t+z-T(n-1)) \cdot \phi(t-T(n-1))$$

$$= \frac{1}{2} \cos(2\pi f_0 z) \cdot \sum_n G_x^2 \phi(t+z-nT) \cdot \phi(t-nT)$$

$$= R_{yy}(t+z, t)$$

Εννεπωι $Y(t)$ ειναι κωιθωταγορη
 και ανω ευρεα ευρεα.

B.3

κωιθωταγορη $\rightarrow S_y(F) = F \{ \bar{R}_y(z) \}$

$$\bar{R}_y(z) = \frac{1}{T} \int_T R_{yy}(t+z, t) dt = \frac{1}{T} \int_T R_{xx}(t+z, t) \cdot \frac{1}{2} \cos(2\pi f_0 z) dt$$

$$= \frac{1}{2T} \cos(2\pi f_0 z) \cdot \int_T R_{xx}(t+z, t) dt = \frac{1}{2} \cos(2\pi f_0 z) \cdot \bar{R}_x(z)$$

τελμα:

$$S_y(F) = F \{ \bar{R}_y(z) \} = \frac{1}{2} F \{ \cos(2\pi f_0 z) \cdot \bar{R}_x(z) \} = \frac{1}{4} (S_x(F+f_0) + S_x(F-f_0))$$

Κώδικας

```
-----  
%This function takes a sequence of n bits and transforms it into 4PAM  
%signal  
function [S] = bits_to_4PAM(b)
```

```
%every time we check a pair of numbers  
for k=1:2:length(b)  
    if b(k)==0 && b(k+1)==0  
        S(k) = 3;  
    elseif(b(k)==0 && b(k+1)==1)  
        S(k)=1;  
    elseif(b(k)==1 && b(k+1)==1)  
        S(k)=-1;  
    elseif(b(k)==1 && b(k+1)==0)  
        S(k)=-3;  
    else  
        disp('Error')  
        return  
    end  
end
```

```
-----  
  
clc  
clear all;  
close all;
```

```
-----  
% --- Part A ---  
-----
```

```
%A.1  
T = 10^-3;  
over = 10;  
Ts = T/over;  
A = 4;  
a = 0.5;  
N = 2048;%choose a large N as requested
```

```
[phi,t] = srrc_pulse(T, Ts, A, a);  
PHI_f = fftshift(fft(phi,N)*Ts); %Fourier for phi function
```

```
Fs = 1/Ts; %Ts is the sampling period so Fs is the sampling frequency
```

```
%Frequency vector  
F= -Fs/2:Fs/N:Fs/2-Fs/N;
```

```
spectrum_f = abs(PHI_f).^2;
```

```
figure;  
semilogy(F,spectrum_f,'DisplayName','A=4 and a=0.5');  
legend('show');  
title('Energy Spectral Density for A=4');
```

```
-----  
%A.2  
-----
```

```
%create demanded bits  
N_bits = 100;  
%Create n bits series  
b = (sign(randn(N_bits,1))+1)/2;
```

```

X = bits_to_2PAM(b);% the demanded decoding

T_plot = 0:Ts:N_bits-Ts;%Time will cover from 0 until time of 1 symbol
X_delta = 1/Ts * upsample(X,over);

X_delta_conv = conv(X_delta, phi)*Ts;

t_conv = linspace(T_plot(1)+t(1), T_plot(end)+t(end),length(X_delta_conv));%generates
length(X_delta_conv) points

Sx = (var(X_delta_conv)/T).*spectrum_f;

figure;
subplot(2,1,1);
plot(t_conv,X_delta_conv,'DisplayName', 'a=0.5');
title('X(t) when A=4');
legend('show');

-----
%A.3.a
-----
T_total = length(t_conv)*T;
PXF = ((abs(fftshift(fft(X_delta_conv,N)))).^2)*Ts)./T_total;

%PXF = FX./T_total;

figure;
subplot(4,1,1)
plot(F,PXF);
hold on;
title('Px(F) A=4 and a=0.5');

%i choose to use phi pulse that has the default values and not the ones i
%randomly used
%A.3.b
reps = 100;
for n = 1:reps
    b = (sign(randn(N_bits,1))+1)/2;    %generate different symbols
    X1 = bits_to_2PAM(b);% the demanded decoding
    X_delta1 = 1/Ts * upsample(X1,over);
    X_delta_conv1 = conv(X_delta1, phi)*Ts;
    PFX_keep(n,:) = ((abs(fftshift(fft(X_delta_conv1,N))))).^2)./T_total;    %save for different
symbols
end

PSD_exp = sum(PFX_keep, 1)*Ts./reps;

figure;
semilogy(F, PSD_exp,'DisplayName', 'experimental values');
hold on;
semilogy(F, Sx,'DisplayName', 'theoretical values');
title('Power Spectral Density of X for 2PAM');
legend('show');

-----
%A.4
-----
%N_bits = 100 and b = (sign(randn(N_bits,1))+1)/2 from A.2;

X2 = bits_to_4PAM(b);% the demanded decoding
TX2_plot = 0:Ts:N_bits/2-Ts;%Time will cover from 0 until time of 1 symbol
XA4_delta = 1/Ts * upsample(X2,over);

```

```

XA4_delta_conv = conv(XA4_delta, phi)*Ts;

tA4_conv = linspace(TX2_plot(1)+t(1), TX2_plot(end)+t(end),length(X_delta_conv));%generates
length(X_delta_conv) points
%figure;
%plot(t_conv,X_delta_conv)
T_total_A4 = length(tA4_conv)*T;

FX4 = (abs(fftshift(fft(XA4_delta_conv,N))).^2)*Ts;
PXF4 = FX4./T_total_A4;

SxA4 = (var(XA4_delta_conv)/T).*spectrum_f;

for n = 1:reps
    b = (sign(randn(N_bits,1))+1)/2;    %generate different symbols

    XA4 = bits_to_4PAM(b);% the demanded decoding

    %T_plot = 0:Ts:N_bits-Ts;%Time will cover from 0 until time of 1 symbol
    X_deltaA4 = 1/Ts * upsample(XA4,over);

    X_delta_convA4 = conv(X_deltaA4, phi)*Ts;
    %t_conv = linspace(T_plot(1)+t(1), T_plot(end)+t(end),length(X_delta_conv));%generates
length(X_delta_conv) points
    PFX_keepA4(n,:) = ((abs(fftshift(fft(X_delta_convA4,N))).^2)./T_total_A4;    %save for
different symbols

end

PSD_expA4 = sum(PFX_keepA4, 1)*Ts./reps;

figure;
semilogy(F, PSD_expA4,'DisplayName', 'experimental values');
hold on;
semilogy(F, SxA4,'DisplayName', 'theoretical values');
title('Power Spectral Density of X for 4PAM');
legend('show');
-----
%A.5
-----
Tt = 2*T;    %as requested
%create new signals for new time
[phi,t] = srnc_pulse(Tt, Ts, A, a);

PHI_f = fftshift(fft(phi,N)*Ts); %Fourier for phi function
spectrum_f = abs(PHI_f).^2;

N_bits = 100;
%Create n bits series
b = (sign(randn(N_bits,1))+1)/2;

X = bits_to_2PAM(b);% the demanded decoding

T_plot = 0:Ts:N_bits-Ts;%Time will cover from 0 until time of 1 symbol
X_delta = 1/Ts * upsample(X,over);

X_delta_conv = conv(X_delta, phi)*Ts;
Sx = (var(X_delta_conv)/T).*spectrum_f;

t_conv = linspace(T_plot(1)+t(1), T_plot(end)+t(end),length(X_delta_conv));%generates
length(X_delta_conv) points

T_total = length(t_conv)*Tt;

```

```

PXF = ((abs(fftshift(fft(X_delta_conv,N)))).^2)*Ts)./T_total;
figure;
subplot(4,1,1)
plot(F,PXF);
hold on;
title('Px(F) A=4 and a=0.5');

%i choose to use phi pulse that has the default values and not the ones i
%randomly used
-----
%A.5.b
-----
reps = 100;
for n = 1:reps
    b = (sign(randn(N_bits,1))+1)/2;    %generate different symbols
    X1 = bits_to_2PAM(b);% the demanded decoding
    X_delta1 = 1/Ts * upsample(X1,over);
    X_delta_conv1 = conv(X_delta1, phi)*Ts;
    PFX_keep(n,:) = ((abs(fftshift(fft(X_delta_conv1,N))))).^2)./T_total;    %save for different
symbols
end

PSD_exp = sum(PFX_keep, 1)*Ts./reps;

figure;
semilogy(F, PSD_exp,'DisplayName', 'experimental values');
hold on;
semilogy(F, Sx/2,'DisplayName', 'theoretical values');
title('Power Spectral Density of X for 2PAM');
legend('show');

```