
Αναφορά δεύτερης άσκησης

Στοιχεία Ομάδας

Αλέξανδρος Σεργάκης: 2017030199

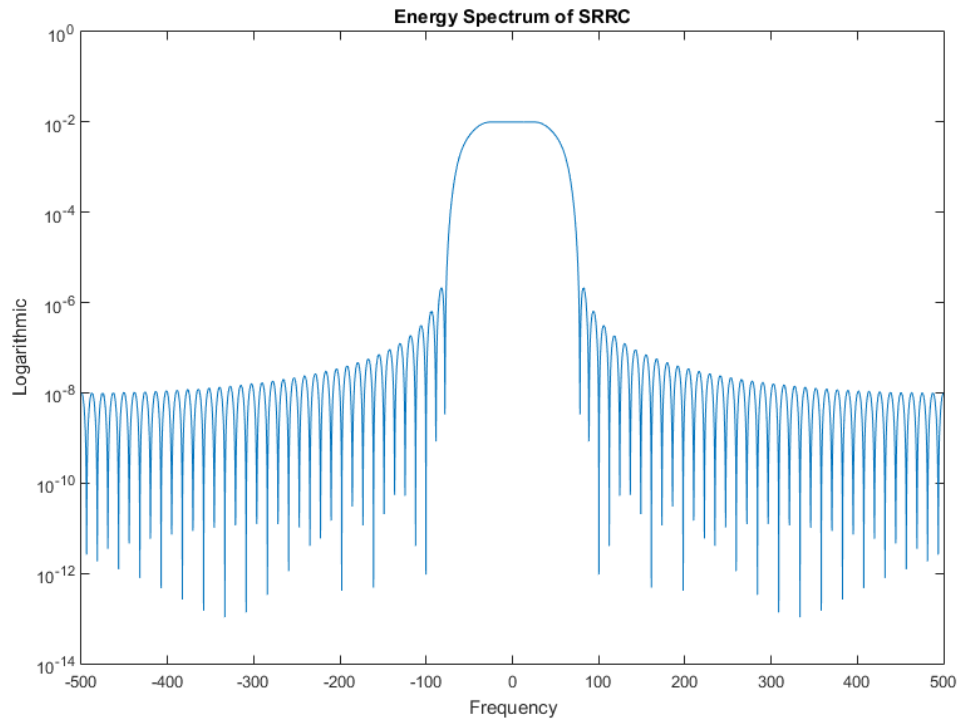
Ευάγγελος Κατσούπης: 2017030077

A (10) Στο πρώτο μέρος της άσκησης, το οποίο είναι, κυρίως, πειραματικό, θα μελετήσουμε το φασματικό περιεχόμενο PAM κυματομορφών βασικής ζώνης.

A.1 Να δημιουργήσετε παλμό SRRC $\phi(t)$ με τιμές $T = 10^{-2}$, $over = 10$, $Ts = \frac{T}{over}$, $A = 4$ και $a = 0.5$.

(10) Μέσω των συναρτήσεων `fftshift` και `fft`, να υπολογίσετε το μέτρο του μετασχηματισμού Fourier της $\phi(t)$, $|\phi(F)|$, σε N_f ισαπέχοντα σημεία στο διάστημα $[-\frac{Fs}{2}, \frac{Fs}{2}]$. 1. Να σχεδιάσετε τη φασματική πυκνότητα ενέργειας $|\phi(F)|^2$ στον κατάλληλο άξονα συχνοτήτων με χρήση της εντολής `semilogy`.

Μέσω της συνάρτησης SRRC δημιουργούμε την $\phi(t)$ και μέσω των συναρτήσεων `fft` και `fftshift` παράγουμε τον μετασχηματισμό Fourier $\phi(F)$ της $\phi(t)$ στο διάστημα $[-\frac{Fs}{2}, \frac{Fs}{2}]$. Ορίζεται ως φασματική πυκνότητα ενέργειας της $\phi(t)$ η $|\phi(F)|^2$.



Ο κώδικας υπολογισμού του A.1 είναι ο εξής:

```
% Exercise A.1
clear all;
close all;

T = 0.01; %Given Period
over = 10; %Oversampling factor
A = 4; %Half duration of the pulse in symbol periods
a = 0.5; %Roll-off factor
Ts = T / over; %sampling period
Fs = 1/Ts; % Sampling Frequency

% Number of samples
K = 1000;

% Step
% Nf = 1024;
Nf = 4096;

% Generate the pulse
```

```

[phi, t] = srrc_pulse(T, over, A, a);

% Frequency range
f_axis = linspace(-Fs/2, (Fs/2-Fs/Nf), Nf);

% Fourier Transform of Phi
phi_F = fftshift(fft(phi, Nf)*Ts);
abs_phi_F = abs(phi_F);
Energy_phi = power(abs_phi_F, 2);
% Set figure
figure('Name', 'Energy.SRRC.logScale');
semilogy(f_axis, Energy_phi);
hold on;
title('Energy Spectrum of SRRC');
xlabel('Frequency');
ylabel('Logarithmic');
xlim([-Fs/2 Fs/2]);
hold off;

```

A.2 Να δημιουργήσετε ακολουθία $N = 100$ ανεξάρτητων και ισοπίθανων (b_0, \dots, b_{N-1}) . Χρησιμοποιώντας την απεικόνιση

$$0 \rightarrow +1$$

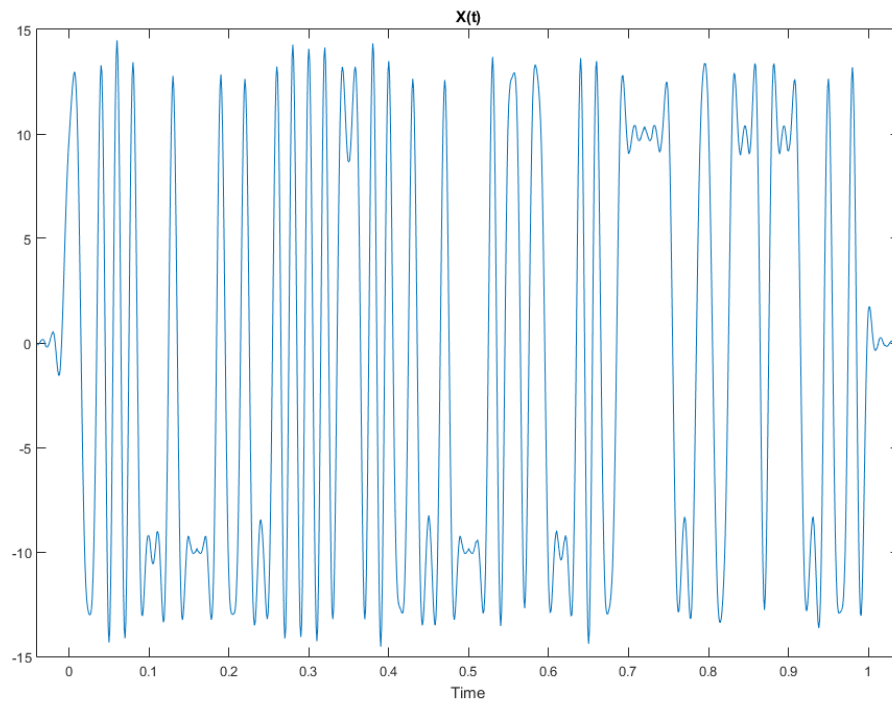
$$1 \rightarrow -1,$$

να απεικονίσετε τα bits σε σύμβολα X_n για $n = 0, \dots, N - 1$.

Να κατασκευάσετε την κυματομορφή:

$$X(t) = \sum_{n=0}^{N-1} X_n \phi(t - nT)$$

Στην αρχή δημιουργήθηκαν 100 τυχαία ανεξάρτητα και ισοπίθανα bits τα οποία μετατρέπονται μέσω της συνάρτησης *bits_to_2PAM* σε 2-PAM σύμβολα. Στην συνέχεια κάνουμε upsampled την ακολουθία συμβόλων και γίνεται η συνέλιξη με την $\phi(t)$ με την χρήση της συνάρτησης *conv()* έτσι ώστε να δημιουργήσουμε την συνάρτηση $X(t)$.



Ο κώδικας υπολογισμού του A.2 είναι ο εξής:

```
N = 100;
% Bits generation
b = (sign(randn(N, 1)) + 1)/2;

% Generate 2-PAM
X_2pam = bits_to_2PAM(b);

% Upsample X_2pam
X_Upsampled = (1/Ts) * upsample(X_2pam, over);
X_Upsampled_time = 0:Ts:N*T-Ts;

% The Sum is the convolution X and phi
X = conv(X_Upsampled, phi)*Ts;
X_time = X_Upsampled_time(1)+t(1):Ts:X_Upsampled_time(end
    )+t(end);

% Set figure
figure('Name','2pamConvPhi');
plot(X_time,X);
hold on;
title('X(t)');
xlabel('Time');
xlim([X_time(1) X_time(end)]);
hold off;
```

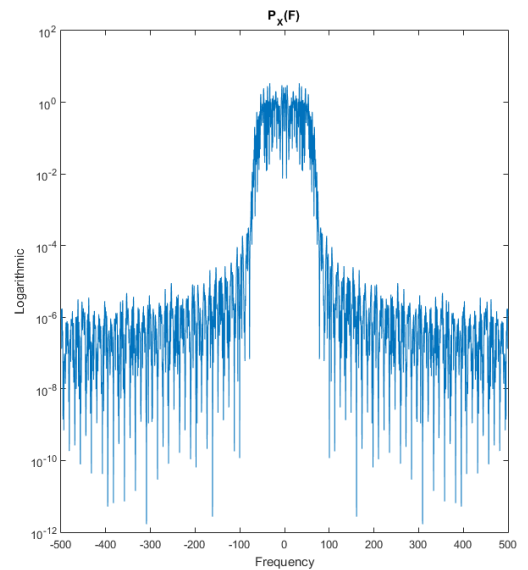
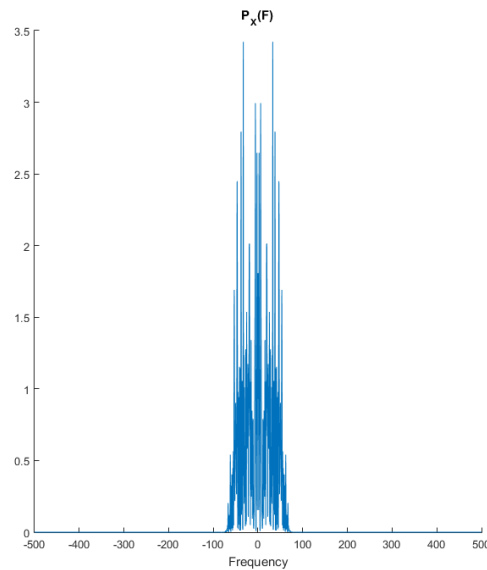
Υποθέτοντας ότι το πλήθος των συμβόλων είναι άπειρο, γνωρίζουμε πως η φασματική πυκνότητα ισχύος της $X(t)$ είναι:

$$S_x(F) = \frac{\sigma_X^2}{T} \cdot |\phi(F)|^2 = \frac{1}{10^{-2}} \cdot |\phi(F)|^2$$

A.3 (10) Με χρήση των συναρτήσεων `fft` και `fftshift` να υπολογίσετε το περι-
οδόγραμμα μίας υλοποίησης της $X(t)$.

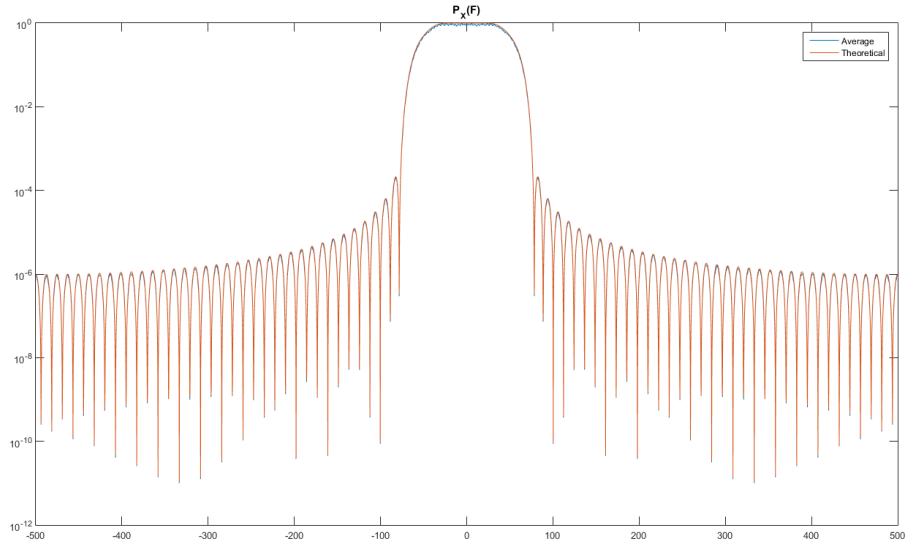
$$P_x(F) = \frac{|\phi(F)|^2}{T_{total}}$$

όπου T_{total} είναι ο συνολικός χρόνος διάρκειας της $X(t)$ σε sec. Να
σχεδιάσετε το $P_X(F)$ με χρήση `plot` και `semilogy`.



Για τον σχεδιασμό του περιοδόγραμματος σε ημιλογαριθμική και γραμμική
κλίμακα έγινε χρήση της παραπάνω σχέσης.

(10) Να εκτιμήσετε τη φασματική πυκνότητα ισχύος υπολογίζοντας αρ-
ιθμητικές μέσες τιμές πάνω σε K υλοποιήσεις περιοδογραμμάτων. Να
σχεδιάσετε σε κοινό `semilogy` την εκτίμηση και τη θεωρητική φασματική
πυκνότητα ισχύος.



Για τον υπολογισμό της θεωρητικής φασματικής πυκνότητας ισχύος στην αρχή έγινε ο υπολογισμός της μέσης τιμής:

Ισχύει ότι X_n ισοπίθανα σύμβολα λόγου του bit array με

$$(P_X(X = 1) = P_X(X = -1) = \frac{1}{2})$$

Έτσι έχουμε:

$$E(X_n) = \sum_x x p_x = 1 \cdot \frac{1}{2} + (-1) \cdot \frac{1}{2} = 0$$

Στην συνέχεια έγινε ο υπολογισμός της διασποράς με:

$$\sigma_X^2 = E[(X_n - E(X_n))^2] = E[X_n^2] = 1$$

$$\text{Διότι } E(X_n^2) = \sum_x x^2 p_x = (1)^2 \cdot \frac{1}{2} + (-1)^2 \cdot \frac{1}{2} = 1$$

Άρα η θεωρητική φασματική πυκνότητα ισχύος είναι ίση με:

$$S_x(F) = \frac{\sigma_X^2}{T} \cdot |\phi(F)|^2 = \frac{1}{10^{-2}} \cdot |\phi(F)|^2 = 100 \cdot |\phi(F)|^2$$

Κατα την σύγκριση των θεωρητικών και πειραματικών αποτελεσμάτων σε κοινή ημιλογαριθμική κλίμακα παρατηρείται πως το πειραματικό γράφημα αποκλίνει σε διάφορα σημεία από το θεωρητικό με απόσταση της τάξης 10^{-2} , 10^{-3} . Την πειραματική αναπαράσταση την υπολογίζουμε από τον μέσο όρο των υλοποιήσεων της $X(t)$. Συνεπώς με την αύξηση του αριθμού των

υλοποιήσεων και των δειγμάτων η πειραματική, προσεγγίζει την πραγματική.

Ο κώδικάς για τον υπολογισμό της πειραματικής και της θεωρητικής φασματικής ποικνότητας ισχύος είναι ο εξής:

```
% Fourier transform of X
X_F = fftshift(fft(X, Nf)*Ts);
% Periodogram Numerator - Denominator
Px_F_num = power(abs(X_F),2);
PX_F_denom = X_time(end) - X_time(1);

% Periodogram of 2_PAM
Px_F_2PAM = Px_F_num/PX_F_denom;

%% Separate the plots
% Set figure
figure('Name','Periodograms');
subplot(1,2,1);
hold on;
plot(f_axis, Px_F_2PAM);
title('P_X(F)');
xlabel('Frequency');
xlim([-Fs/2 Fs/2]);
hold off;
hold on;
subplot(1,2,2);
semilogy(f_axis, Px_F_2PAM);
title('P_X(F)');
xlabel('Frequency');
ylabel('Logarithmic');
xlim([-Fs/2 Fs/2]);
hold off;

% A.3.2

Px_F_total = 0;
```



```

% K different bit arrays
for i=1:K
    % Bit array
    b = (sign(randn(N, 1)) + 1)/2;

    % 2-PAM
    X_2pam = bits_to_2PAM(b);

    % Upsample x_delta and upsampled time
    X_Upsampled = (1/Ts) * upsample(X_2pam, over);
    X_Upsampled_time = 0:Ts:N*T-Ts;

    % The Sum is the convolution X and phi
    X = conv(X_Upsampled, phi)*Ts;
    X_time = X_Upsampled_time(1)+t(1):Ts:X_Upsampled_time
(end)+t(end);

    % Fourier transform of X
    X_F = fftshift(fft(X, Nf)*Ts);
    % Periodogram
    Px_F_num = power(abs(X_F), 2);
    PX_F_denom = X_time(end) - X_time(1);
    Px_F_total = Px_F_total + (Px_F_num./PX_F_denom);
end

Px_F_2PAM_average = Px_F_total./K;
Px_F_theoretical = 1/T .* Energy_phi;

% Set figure
figure('Name', 'Px_F_Average');
semilogy(f_axis, Px_F_2PAM_average);
hold on;
semilogy(f_axis, Px_F_theoretical);
title('P_X(F)');
xlim([-Fs/2 Fs/2]);
legend('Average', 'Theoretical');

```

A.4 Χρησιμοποιώντας την απεικόνιση

$$00 \rightarrow +3$$

$$01 \rightarrow +1$$

$$11 \rightarrow -1$$

$$10 \rightarrow -3,$$

Να κατασκευάσετε την ακολουθία 4-PAM X_n , για $n = 0, \dots, \frac{N}{2} - 1$.

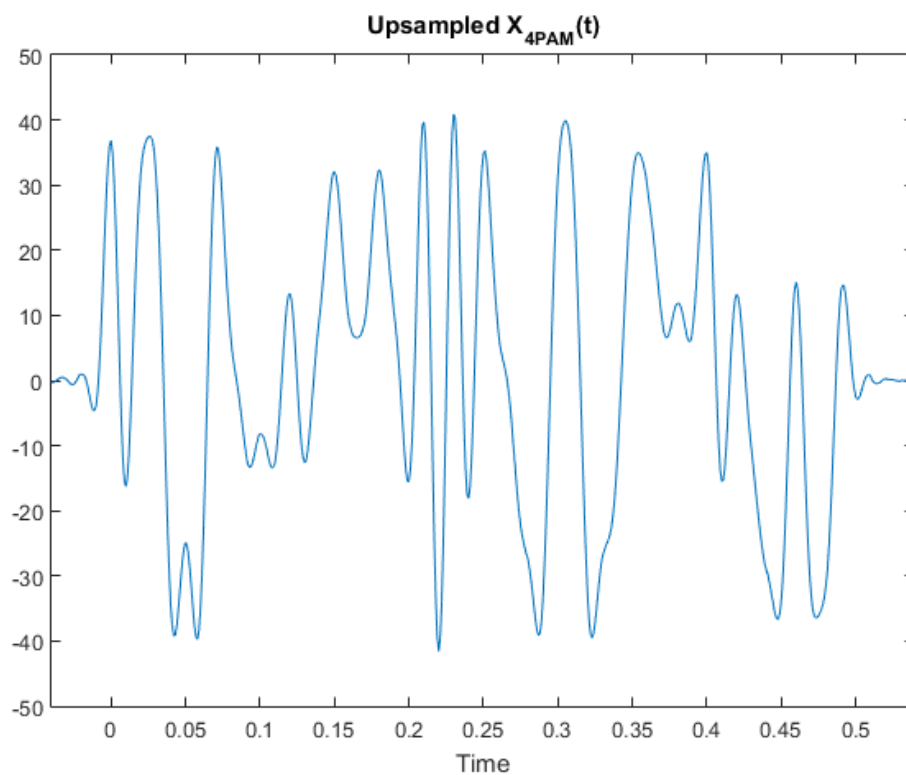
Παρατηρήστε ότι, αν τα bits είναι ισοπίθανα, τότε και τα σύμβολα X_n είναι ισοπίθανα! Να κατασκευάσετε την κυματομορφή:

$$X(t) = \sum_{n=0}^{\frac{N}{2}-1} X_n \phi(t - nT)$$

χρησιμοποιώντας την ίδια περίοδο T με το ερώτημα A.2.

Δημιουργήθηκε νέα συνάρτηση *bits_to_4PAM* μέσω της οποίας κατασκευάστηκε η ακολουθία 4-PAM για $n = 0, \dots, \frac{N}{2} - 1$ όπου $N=100$.

Η κυματομορφή του $X(t)$ (υλοποίηση όμοια με αυτή του A2):



Ο κώδικας 4-PAM:

```
function [X] = bits_to_4PAM(b)
% Conversion of bit array to 4-PAM Symbols
%
%
% Input Bit b(i) / 00 / 01 / 11 / 10 /
%-----
% Output 2-PAM / +3 / +1 / -1 / -3 /
% Symbol X(i) /

for i = 1:2:length(b)
    bits = b(i:i+1);
    index = (i+1)/2;

    % B is type of (x1;x2;x3;x4;x5)
    if isequal(bits, [0;0])
        X(index) = +3;
    elseif isequal(bits, [0;1])
        X(index) = +1;
    elseif isequal(bits, [1;1])
        X(index) = -1;
    elseif isequal(bits, [1;0])
        X(index) = -3;
    end
end
end
```

Ο κώδικας για την υλοποίηση του $X(t)$:

```
% 4PAM symbols
X_4pam = bits_to_4PAM(b);

% Upsample x and upsampled time
X_Up = (1/Ts) * upsample(X_4pam, over);
X_Up_time = 0:Ts:(N/2)*T-Ts;

% The Sum is convolution X and phi
X = conv(X_Up, phi)*Ts;
X_time = X_Up_time(1)+t(1):Ts:X_Up_time(end)+t(end);

figure('Name','4pamConvPhi');
plot(X_time, X);
hold on;
title('Upsampled X_{4PAM}(t)');
xlabel('Time');
xlim([X_time(1) X_time(end)]);
hold off;
```

(10) Να υπολογίσετε το περιοδόγραμμα και να εκτιμήσετε τη φασματική πυκνότητα ισχύος μέσω αριθμητικών μέσων τιμών υλοποιήσεων περιοδογραμμάτων της $X(t)$. Να σχεδιάσετε την πειραματική και την θεωρητική φασματική πυκνότητα ισχύος στο ίδιο semilogy. Τι παρατηρείτε;

Για τον υπολογισμό της θεωρητικής φασματικής πυκνότητα ισχύος έγιναν οι εξής μαθηματικές πράξεις:

Αρχικά υπολογίστηκε η μέση τιμή: Ισχύει, όμοια, ότι X_n ισοπίθανα σύμβολα λόγω του bit array με

$$P_X(X = 1) = P_X(X = -1) = P_X(X = 3) = P_X(X = -3) = \frac{1}{4}$$

Έτσι έχουμε:

$$E(X_n) = \sum_x x^2 p_x = (1) \cdot \frac{1}{4} + (-1) \cdot \frac{1}{4} + (3) \cdot \frac{1}{4} + (-3) \cdot \frac{1}{4} = 0$$

Στην συνέχεια έγινε ο υπολογισμός της διασποράς:

$$\sigma_X^2 = \mathcal{E} [(X_n - E(X_n))^2] = \mathcal{E} [X_n^2] = 5$$

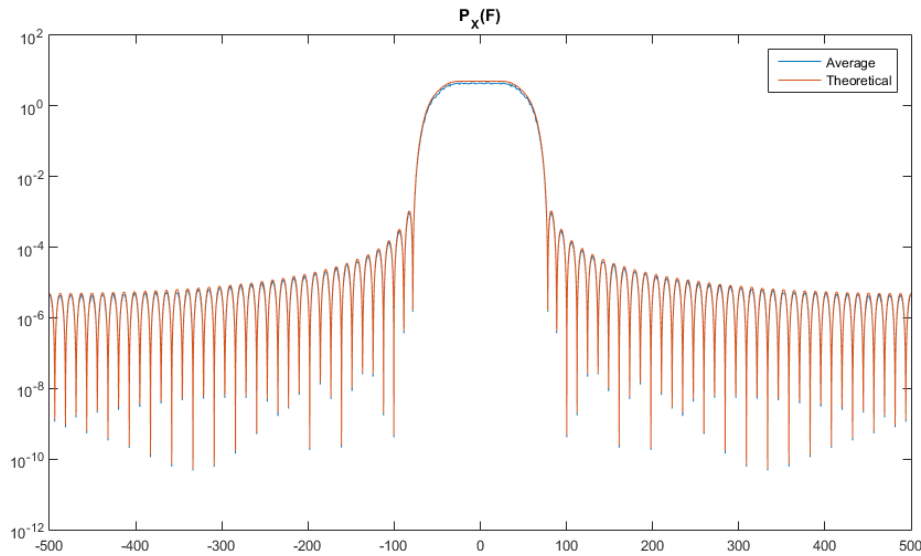
$$\Delta\acute{o}\tau\iota, E(X_n^2) = \sum_x x^2 p_x = (1)^2 \cdot \frac{1}{4} + (-1)^2 \cdot \frac{1}{4} + (3)^2 \cdot \frac{1}{4} + (-3)^2 \cdot \frac{1}{4} = 5$$

Άρα η θεωρητική φασματική πυκνότητα ισχύος είναι ίση με:

$$S_x(F) = \frac{\sigma_X^2}{T} \cdot |\phi(F)|^2 = \frac{5}{10^{-2}} \cdot |\phi(F)|^2 = 500 \cdot |\phi(F)|^2$$

Είναι εύκολα παρατηρήσιμο πως ισχύει $S_{X_{4pam}}(F) = 5S_{X_{2pam}}(F)$

Έπειτα υπολογίζουμε πειραματικά την φασματική πυκνότητα ισχύος, για 1000 υλοποιήσεις της $X(t)$.



Κατα την σύγκριση των θεωρητικών και πειραματικών αποτελεσμάτων σε κοινή ημιλογαριθμική κλίμακα παρατηρείται πως το πειραματικό γράφημα αποκλίνει σε διάφορα σημεία απο το θεωρητικό με απόσταση της τάξης $\simeq 5 \cdot 10^{-1}$. Την πειραματική αναπαράσταση την υπολογίζουμε και πάλι απο τον μέσο όρο των υλοποιήσεων της $X(t)$. Συνεπώς με την αύξηση του αριθμού των υλοποιήσεων και των δειγμάτων, η πειραματική, προσεγγίζει την πραγματική.

Ο κώδικας που υπολογίζει την πειραματική τιμή της S_x και σχεδιάζει τις φασματικές πυκνότητες ισχύος.:

```
PxF_4pamTotal = 0;

% K different bit arrays
for i=1:K
    % Bit array
    b = (sign(randn(N, 1)) + 1)/2;

    % 2-PAM
    X_4pam = bits_to_4PAM(b);

    % Upsample
    X_Up = (1/Ts) * upsample(X_4pam, over);
    X_Up_time = 0:Ts:(N/2)*T-Ts;

    % The Sum is the convolution X and phi
    X = conv(X_Up,phi)*Ts;
    X_time = X_Up_time(1)+t(1):Ts:X_Up_time(end)+t(end);

    % Fourier transform of X
    X_F = fftshift(fft(X, Nf)*Ts);
    % Periodogram
    PxF_num = power(abs(X_F),2);
    PX_F_denom = X_time(end) - X_time(1);
    PxF_4pamTotal = PxF_4pamTotal + (PxF_num/PX_F_denom)
;
end

PxF_4PAM_average = PxF_4pamTotal/K;
PxF_theoretical = 5/T * Energy_phi;

% Set figure
figure('Name','Px4pamF.Average');
semilogy(f_axis,PxF_4PAM_average);
hold on;
semilogy(f_axis,PxF_theoretical);
```

```

title('P_X(F)');
xlim([-Fs/2 Fs/2]);
legend('Average','Theoretical');

figure('Name','Periodograms');
subplot(1,2,1);
hold on;
plot(f_axis, Px_F_2PAM);
title('P_X(F)');
xlabel('Frequency');
xlim([-Fs/2 Fs/2]);
hold off;
hold on;
subplot(1,2,2);
semilogy(f_axis, Px_F_2PAM);
title('P_X(F)');
xlabel('Frequency');
ylabel('Logarithmic');
xlim([-Fs/2 Fs/2]);
hold off;

%figure Px_2PAM and Px_4PAM
figure('Name','Px_F.2-4.PAM');
subplot(1,2,1);
semilogy(f_axis, Px_F_2PAM_average);
hold on;
semilogy(f_axis, Px_F_4PAM_average);
title('P_X(F)');
xlim([-Fs/2 Fs/2]);
ylabel('Logarithmic');
legend('P_X(F)_{2PAM}','P_X(F)_{4PAM}');

subplot(1,2,2);
plot(f_axis, Px_F_2PAM_average);
hold on;

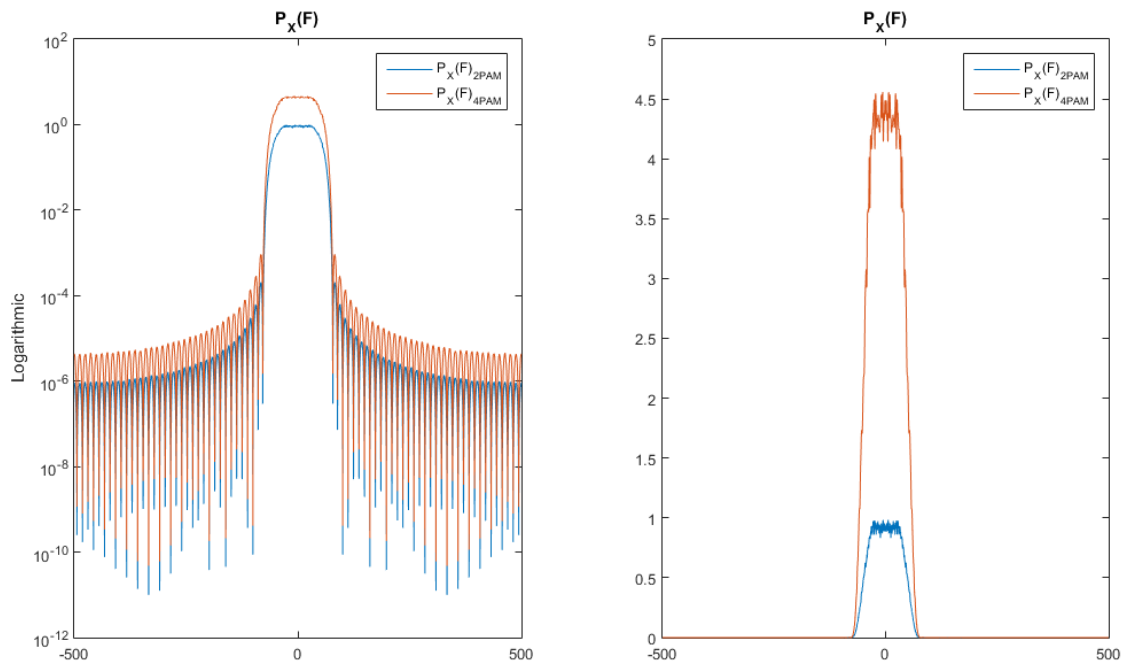
```

```

plot(f_axis, PxF_4PAM_average);
title('P_X(F)');
xlim([-Fs/2 Fs/2]);
legend('P_X(F)_{2PAM}', 'P_X(F)_{4PAM}');

```

(10) Πώς συγκρίνεται, ως προς το εύρος φάσματος και ως προς το μέγιστο πλάτος τιμών, η φασματική πυκνότητα ισχύος της $X(t)$ σε σχέση με αυτή της $X(t)$ του βήματος Α.2· Μπορείτε να εξηγήσετε τα αποτελέσματα της σύγκρισης·



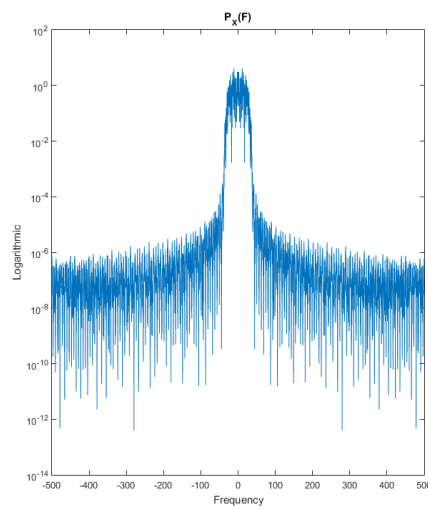
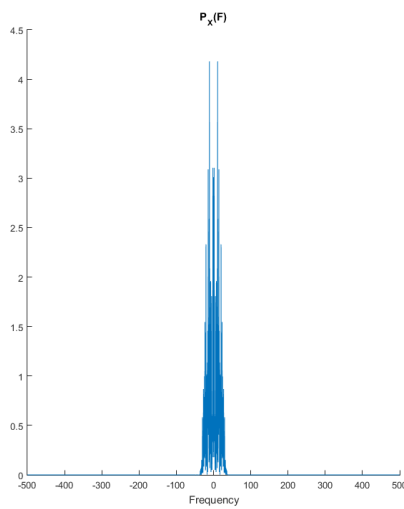
Κάναμε κοινό plot και semilogy τις φασματικές πυκνότητες ισχύος της 2-PAM και της 4-PAM για να δούμε τις διαφορές τους. Αρχικά παρατηρήθηκε ότι το εύρος φάσματος ($BW = \frac{1+a}{2T}$) και των 2 είναι το ίδιο. Αυτό συμβαίνει διότι έχουμε κοινή συχνότητα συμβόλου και κοινό a . Στην συνέχεια διαπιστώθηκε ότι το μέγιστο πλάτος τιμών της 4-PAM είναι σχεδόν 5 φορές μεγαλύτερο από αυτό της 2-PAM διότι η διασπορά του 4-PAM είναι μεγαλύτερη.

A.5 (10) Να επαναλάβετε το βήμα A.3, θέτοντας περίοδο συμβόλου $T' = 2T$ (να διατηρήσετε την περίοδο δειγματοληψίας Ts ίση με αυτή των προηγούμενων βημάτων, άρα, θα πρέπει να διπλασιάσετε την παράμετρο *over*).

Άρα το $T' = 2 \cdot 10^{-2}$ και *over'* = 20

Για τον σχεδιασμό του περιοδόγραμματος σε ημιλογαριθμική και γραμμική κλίμακα έγινε χρήση της παρακάτω σχέσης.

$$P_x(F) = \frac{|\phi(F)|^2}{T_{total}}.$$



Για τον υπολογισμό της θεωρητικής φασματικής πυκνότητας ισχύος στην αρχή έγινε ο υπολογισμός της μέσης τιμής:

Ισχύει ότι X_n ισοπίθανα σύμβολα λόγου του bit array με

$$P_X(X = 1) = P_X(X = -1) = \frac{1}{2}$$

$$E(X_n) = \sum_x x p_x = 1 \cdot \frac{1}{2} + (-1) \cdot \frac{1}{2} = 0$$

Όμοια με A.3 ισχύει

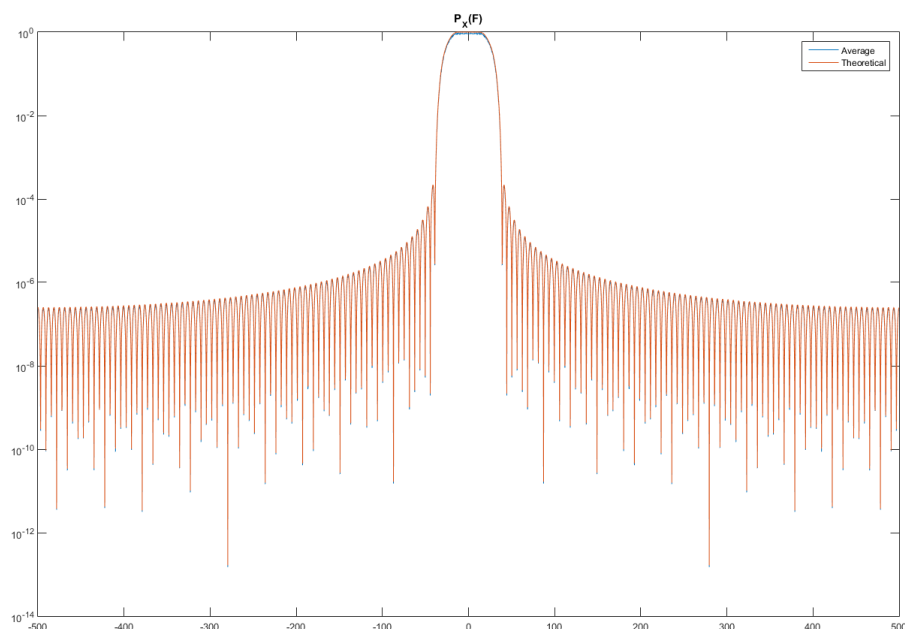
$$\sigma_X^2 = \mathcal{E}[(X_n - E(X_n))^2] = \mathcal{E}[X_n^2] = 1$$

Άρα η θεωρητική φασματική πυκνότητα ισχύος είναι ίση με:

$$S_x(F) = \frac{\sigma_X^2}{T'} \cdot |\phi(F)|^2 = \frac{1}{2 \cdot 10^{-2}} \cdot |\phi(F)|^2 = 50 \cdot |\phi(F)|^2$$

Ο κώδικας για τον υπολογισμό και την υλοποίηση των plots είναι όμοιος με αυτόν του A.3.

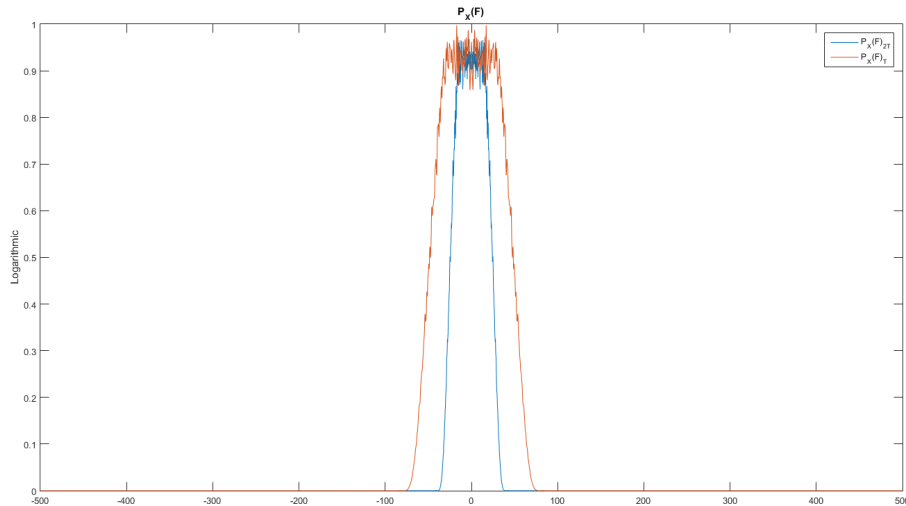
Κοινά γραφήματα:



Κατα την σύγκριση των θεωρητικών και πειραματικών αποτελεσμάτων σε κοινή ημιλογαριθμική κλίμακα παρατηρείται πως το πειραματικό γράφημα αποκλίνει σε διάφορα σημεία απο το θεωρητικό με απόσταση της τάξης όμοια με $\simeq 10^{-2}$.

Συνεπώς με την αύξηση του αριθμού των υλοποιήσεων και των δειγμάτων η πειραματική, προσεγγίζει την πραγματική.

(5) Τι παρατηρείτε σχετικά με το εύρος φάσματος των κυματομορφών σε αυτή την περίπτωση σε σχέση με αυτό των κυματομορφών του βήματος A.3· Μπορείτε να εξηγήσετε το φαινόμενο·



Κάναμε κοινό plot των περιοδογραμμάτων για $T = 10^{-2}$, $over = 10$ και $T' = 2 \cdot T$, $over' = 2 \cdot over$ για να δούμε τις διαφορές τους. Παρατηρήθηκε ότι το εύρος φάσματος της $T' = 2 \cdot T$ με $BW' = \frac{1+a}{2T'}$ είναι μικρότερο της $T = 10^{-2}$ διότι το $T' = 2T$. Άρα $(BW' = \frac{1+a}{2T'} = 37.5) < (BW = \frac{1+a}{2T} = 75)$.

A.6 (2.5) Αν θέλατε να στείλετε δεδομένα όσο το δυνατό ταχύτερα έχοντας διαθέσιμο το ίδιο εύρος φάσματος, θα επιλέγατε 2-PAM ή 4-PAM, και γιατί·

Εφόσον έχουμε κοινό εύρος φάσματος και θέλουμε να μεταδώσουμε bit με την μεγαλύτερη δυνατή ταχύτητα θα επιλέγαμε 4-PAM. Γιατί σε 2-PAM κάθε bit γίνεται 1 σύμβολο με αποτέλεσμα να χρειάζεται ένας χρόνος τ για να στείλουμε έναν αριθμό από bit. Ενώ σε 4-PAM η αντιστοιχία είναι 2 προς 1 δηλαδή 2 bits μετατρέπονται σε ένα σύμβολο και έτσι μπορούμε να στείλουμε τον ίδιο αριθμό bit στον μισό χρόνο ($\tau/2$).

(2.5) Αν το διαθέσιμο εύρος φάσματος είναι πολύ ακριβό, θα επιλέγατε περίοδο συμβόλου T ή $T' = 2T$, και γιατί·

Αν το διαθέσιμο εύρος φάσματός είναι πολύ ακριβό θα διαλέγαμε $T' = 2T$ διότι όπως δείξαμε στο Α.5. για $T' > T \Rightarrow BW' < BW$ (\simeq μισό εύρος φάσματος).

Β. Έστω η κυματομορφή:

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

όπου X_n είναι ανεξάρτητα τυχαία σύμβολα με $E[X_n]$ και $E[X_n^2] = \sigma_X^2$ και $T > 0$ η περίοδος του συμβόλου. Η $X(t)$ διαμορφώνει ένα ημιτονοειδές σήμα. Το διαμορφωμένο σήμα είναι:

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

όπου Θ είναι τυχαία μεταβλητή, ομοιόμορφα κατανοημένη στο $[0, 2\pi)$, ανεξάρτητη των $X_n, \forall n$.

Β.1 Να υπολογίσετε αναλυτικά τις ποσότητες $E[Y(t)]$ και $E[Y(t + \tau)Y(t)]$.

$$\begin{aligned} \text{ι) } E[Y(t)] &= E[X(t) \cos(2\pi f_o t + \Theta)] = E[X(t)] \cdot E[\cos(2\pi f_o t + \Theta)] \\ E[X(t)] &= E[\sum_{n=-\infty}^{\infty} X_n \cdot \phi(t - nT)] = \sum_{n=-\infty}^{\infty} E[X_n] \cdot \phi(t - nT) = \\ &= \sum_{n=-\infty}^{\infty} 0 \cdot \phi(t - nT) = 0 \\ E[\cos(2\pi f_o t + \Theta)] &= \int_0^{2\pi} \frac{1}{2\pi} \cos(2\pi f_o t + \Theta) d\Theta = 0 \\ \text{Άρα } E[Y(t)] &= 0 \end{aligned}$$

ιι) Για τον υπολογισμό του $E[Y(t + \tau)Y(t)]$ θα θέσουμε με $t + \tau = t_1$ και $t = t_2$ Άρα έχουμε:

$$\begin{aligned} E[Y(t_1)Y(t_2)] &= E[X(t_1) \cos(2\pi f_o t_1 + \Theta)] \cdot E[X(t_2) \cos(2\pi f_o t_2 + \Theta)] = \\ &= E[X(t_1)X(t_2) \cos(2\pi f_o t_1 + \Theta) \cdot \cos(2\pi f_o t_2 + \Theta)] = E[X(t_1)X(t_2)] \cdot \\ &= E[\cos(2\pi f_o t_1 + \Theta) \cdot \cos(2\pi f_o t_2 + \Theta)]. \end{aligned}$$

$$\begin{aligned} E[X(t_1)X(t_2)] &= E[\sum_{n=-\infty}^{\infty} X_n \cdot \phi(t_1 - nT) \cdot \sum_{m=-\infty}^{\infty} X_m \cdot \phi(t_2 - mT)] = \\ &= E[\sum_{n=-\infty}^{\infty} X_n^2 \cdot \phi(t_1 - nT) \phi(t_2 - nT)] = \sum_{n=-\infty}^{\infty} E[X_n^2] \cdot \phi(t_1 - nT) \phi(t_2 - nT) \end{aligned}$$

$$\begin{aligned}
& nT)\phi(t_2 - nT)] \\
& = \sum_{n=-\infty}^{\infty} \sigma_n^2 \cdot \phi(t_1 - nT)\phi(t_2 - nT)] = \\
& \sum_{n=-\infty}^{\infty} \sigma_n^2 \cdot \phi(t + \tau - nT)\phi(t - nT)
\end{aligned}$$

$$\begin{aligned}
& \underline{E[\cos(2\pi f_o t_1 + \Theta) \cdot \cos(2\pi f_o t_2 + \Theta)]} = \\
& E[\frac{1}{2} \cos(2\pi f_o(t_1 - t_2)) + \frac{1}{2} \cos(2\pi f_o(t_1 + t_2) + 2\Theta)] = \\
& E[\frac{1}{2} \cos(2\pi f_o(t_1 - t_2))] + E[\frac{1}{2} \cos(2\pi f_o(t_1 + t_2) + 2\Theta)] \\
& = \frac{1}{2} \cos(2\pi f_o(t_1 - t_2)) + 0 \\
& = \frac{1}{2} \cos(2\pi f_o(t + \tau - t)) = \frac{1}{2} \cos(2\pi f_o(\tau)).
\end{aligned}$$

Άρα

$$E[Y(t+\tau)Y(t)] = \left(\sum_{n=-\infty}^{\infty} \sigma_n^2 \phi(t + \tau - nT)\phi(t - nT) \right) \cdot \frac{1}{2} \cos(2\pi f_o(\tau))$$

B.2 Να χαρακτηρίσετε την $Y(t)$ ως προς τη (κυκλο)-στασιμότητα, υπό την ευρεία έννοια.

Για να μπορεί να χαρακτηριστεί η $U(t)$ ως στάσιμη υπό την ευρεία έννοια θα πρέπει η μέση τιμή να είναι ανεξάρτητη του t , και η συνάρτηση αυτο-συσχέτισης να εξαρτάται μόνο από τη διαφορά $\tau = t_1 - t_2$.

Από το ερώτημα B.1 γνωρίζουμε πως:

$$\alpha) m_Y(t) = E[Y(t)] = 0$$

$$\beta) E[Y(t_1)Y(t_2)] = \left(\sum_{n=-\infty}^{\infty} \sigma_n^2 \phi(t_1 - nT)\phi(t_2 - nT) \right) \cdot \frac{1}{2} \cos(2\pi f_o(t_1 - t_2)) = R_{YY}(t_1, t_2)$$

Όμως:

$$\begin{aligned}
R_{YY}(t_1 + T, t_2 + T) &= \left(\sum_{n=-\infty}^{\infty} \sigma_n^2 \phi(t_1 + T - nT)\phi(t_2 + T - nT) \right) \cdot \\
&\frac{1}{2} \cos(2\pi f_o(t_1 + T - t_2 - T)) = \left(\sum_{n=-\infty}^{\infty} \sigma_n^2 \phi(t_1 + T - nT)\phi(t_2 + T - nT) \right) \cdot \\
&\frac{1}{2} \cos(2\pi f_o(t_1 + T - t_2 - T)) = \left(\sum_{n=-\infty}^{\infty} \sigma_n^2 \phi(t_1 - (n-1)T)\phi(t_2 - (n-1)T) \right) \cdot \\
&\frac{1}{2} \cos(2\pi f_o(t_1 - t_2)) = R_{YY}(t_1, t_2)
\end{aligned}$$

Επομένως έχουμε

$$\textcircled{1} R_{YY}(t1, t2)$$

$$\textcircled{2} m_Y(t) = m_Y(t + T) = 0$$

Άρα η $Y(t)$ μπορεί να χαρακτηριστεί ως κυκλο-στάσιμη υπό την ευρεία έννοια.

B.3 Να υπολογίσετε τη φασματική πυκνότητα ισχύος της $Y(t)$, $S_Y(F)$, συναρτήσει της $S_X(F)$ και της συχνότητας διαμόρφωσης, f_0 .

Η Φασματική πυκνότητα ισχύος της $Y(t)$ ορίζεται ως $S_Y(F) = \mathcal{F}\{\bar{R}_Y\}$.

Γνωρίζουμε ότι η $Y(t)$ είναι κυκλοστασιμη υπό την ευρεία έννοια, άρα μπορούμε να πούμε πώς $\bar{R}_Y = \frac{1}{T_0} \int_{T_0} R_{YY}(t + \tau, t) dt$

Θα υπολογίσουμε την R_{XX} ως προς την κυκλο-στασιμότητα (θέτουμε $t1 = t + \tau$ και $t2 = t$):

$$\begin{aligned} \underline{R_{XX}(t1 + T, t2 + T)} &= (\sum_{n=-\infty}^{\infty} \sigma_n^2 \phi(t1 + T - nT) \phi(t2 + T - nT)) \\ &= (\sum_{n=-\infty}^{\infty} \sigma_n^2 \phi(t1 - (n - 1)T) \phi(t2 - (n - 1)T)) = \underline{R_{XX}(t1, t2)}. \end{aligned}$$

$$\text{Επίσης } \underline{m_X(t) = E[X(t)] = 0}$$

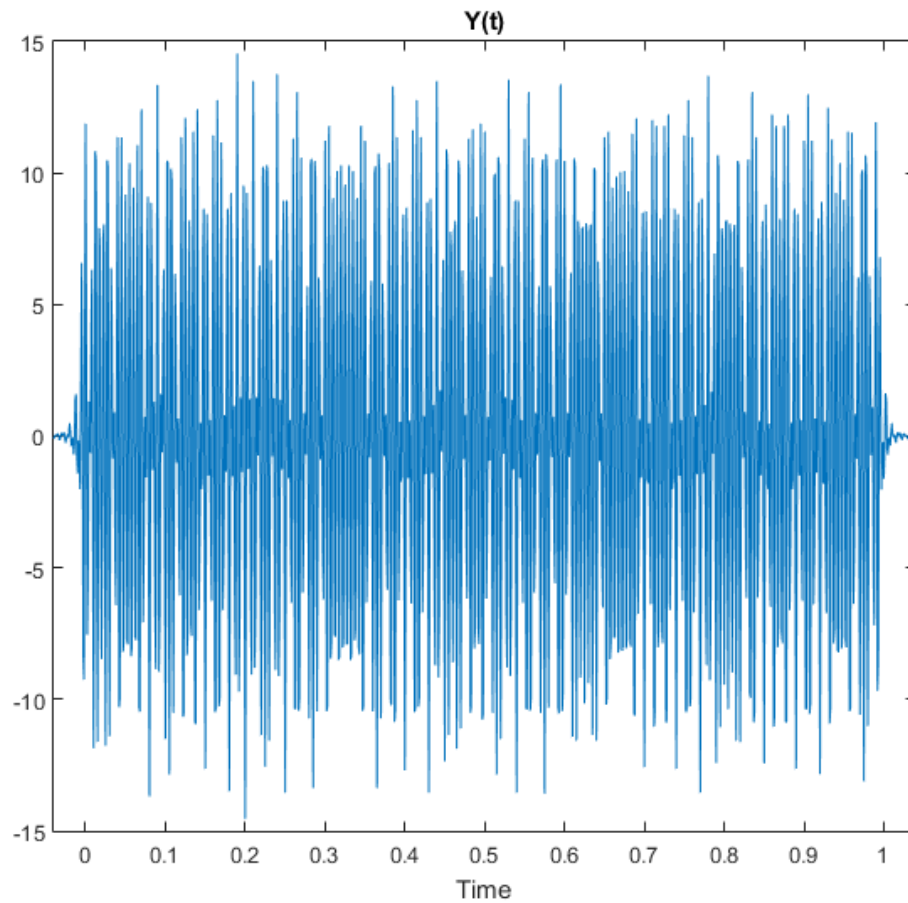
Με χρήση των παραπάνω:

$$\begin{aligned} \bar{R}_Y &= \frac{1}{2T_0} \int_{T_0} R_{XX}(t + \tau, t) \cos(2\pi f_0(t1 - t2)) dt = \\ &\frac{1}{2T_0} \cos(2\pi f_0(t1 - t2)) \int_{T_0} R_{XX}(t + \tau, t) dt = \\ &\frac{1}{2T_0} \cos(2\pi f_0(t + \tau - t)) \bar{R}_X = \\ &\frac{1}{2T_0} \cos(2\pi f_0(\tau)) \bar{R}_X \end{aligned}$$

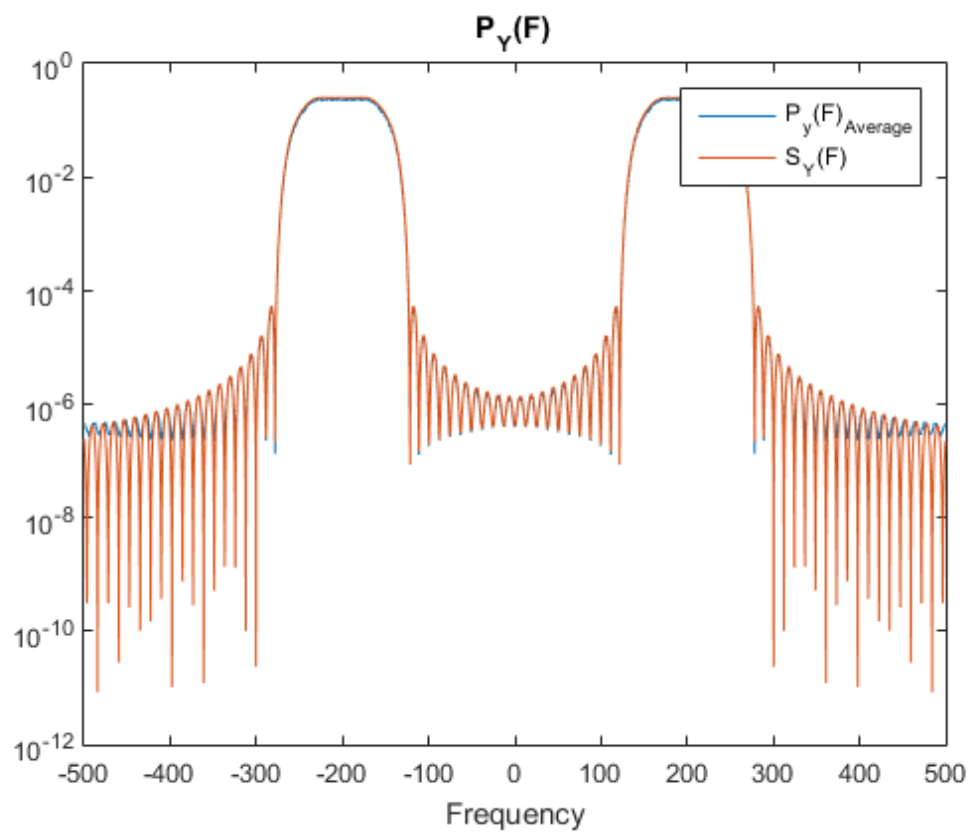
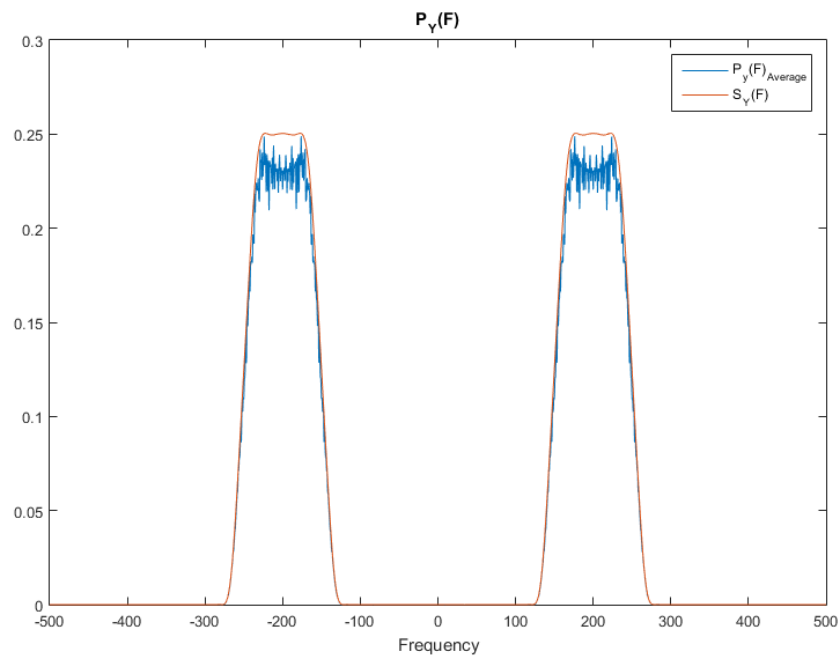
$$\text{Τέλος μπορούμε εύκολα να πούμε πως } S_Y(F) = \mathcal{F}\{\bar{R}_Y\} = \mathcal{F}\{\frac{1}{2T_0} \cos(2\pi f_0(\tau)) \bar{R}_X\} =$$

$$S_Y(F) = \frac{1}{4} S_X(F - F_0) + \frac{1}{4} S_X(F + F_0)$$

B.4 Με την χρήση των A ερωτημάτων δημιουργούμε το σήμα $X(t)$ και το $Y(t)$ με $f_0 = 200Hz$, Θ τυχαία μεταβλητή, ομοιόμορφα κατανομημένη στο $[0, 2\pi)$ και γραφική αναπαράσταση:



Έπειτα υπολογίζουμε την θεωρητική και την πειραματική πυκνότητα ισχύος και την για $K = 1000$ δείγματα με χρήση των παραπάνω και έχουμε:



Μπορούμε να παρατηρήσουμε πως η θεωρητική και η πειραματική $S_Y(F)$ είναι όμοιες με εξαίρεση τα διαστήματα δεξιά και αριστερά των peaks καθώς για την δημιουργία του $S_Y(F)$ χρησιμοποιήσαμε zero padding για να προσομοιώσουμε την μετατόπιση των $S_X(F)$ με $-f_0$ και $+f_0$ αντίστοιχα. Τέλος προσθέσαμε τα δύο σήματα με αποτέλεσμα να δημιουργηθεί το παραπάνω αποτέλεσμα. Βέβαια η διαφορά αυτή είναι αντιληπτή μόνο στην λογαριθμική κλίμακα.

Κώδικας:

```
%B
clear all;
close all;

T = 0.01; %Given Period
over = 10; %Oversampling factor
A = 4; %Half duration of the pulse in symbol periods
a = 0.5; %Roll-off factor
Ts = T / over; %sampling period
Fs = 1/Ts; % Sampling Frequency
K = 1000;
Nf = 4096;
% Generate the pulse
[phi, t] = srrc_pulse(T, over, A, a);

% Frequency range
f_axis = linspace(-Fs/2,(Fs/2-Fs/Nf), Nf);

% Fourier Transform of Phi
phi_F = fftshift(fft(phi, Nf)*Ts);
abs_phi_F = abs(phi_F);
Energy_phi = power(abs_phi_F,2);

% Exercise A.2
N = 100;
% Bits generation
```

```

b = (sign(randn(N, 1)) + 1)/2;

% Generate 2-PAM
X_2pam = bits_to_2PAM(b);

% Upsample X_2pam
X_Upsampled = (1/Ts) * upsample(X_2pam, over);
X_Upsampled_time = 0:Ts:N*T-Ts;

% The Sum is the convolution X and phi
X = conv(X_Upsampled, phi)*Ts;
X_time = X_Upsampled_time(1)+t(1):Ts:X_Upsampled_time(end)+t(
    end);

% % Set figure
% figure('Name', '2pamConvPhi');
% plot(X_time, X);
% hold on;
% title('X(t)');
% xlabel('Time');
% xlim([X_time(1) X_time(end)]);
% hold off;
% 50 < f < 450
f = 200;
theta = 2*pi*rand(1);
Y1=X.*cos(2*pi*f*X_time + theta);

% Plot Y(t)
figure('Name', 'Y(t)');
plot(X_time, Y1);
hold on;
title('Y(t)');
xlabel('Time');
xlim([X_time(1) X_time(end)]);
hold off;
% Average Periodogram of Y(t)
PyF_total = 0;

```

```

for i=1:K
    % bit array
    b = (sign(randn(N, 1)) + 1)/2;
    % bit 2pam
    X_2pam = bits_to_2PAM(b);

    X_Upsampled = (1/Ts) * upsample(X_2pam, over);
    % X(t)
    X = conv(X_Upsampled, phi)*Ts;
    % Generate theta
    theta = 2*pi*rand(1);
    % Y(t)
    Y1 = X.*cos(2*pi*f*X_time + theta);

    Y = fftshift(fft(Y1, Nf)*Ts);

    % Periodogram
    Py_F_TotalTime= X_time(end) - X_time(1);
    PyF_total = PyF_total + (power(abs(Y),2)./Py_F_TotalTime)
;
end

PyF_average = PyF_total./K;
% Sx(f)
SxF = 1/T.*Energy_phi;

indexf0 = round(f/(Fs/Nf));

% SxF delayed
SxF1 = [zeros(1,indexf0) SxF(1:end-indexf0)];
SxF2 = [SxF(indexf0+1:end) zeros(1,indexf0)];

% SyF
SyF = 1/4*(SxF1 + SxF2);
figure;
semilogy(SyF)

```

```

figure('Name','Py_Average')
semilogy(f_axis,PyF_average);
hold on;
title('P_Y(F)');
xlabel('Frequency');
hold on;
semilogy(f_axis,SyF);

```

```

figure('Name','Py_Average')
plot(f_axis,PyF_average);
hold on;
title('P_Y(F)');
xlabel('Frequency');
hold on;
plot(f_axis,SyF);
legend('','')

```

Κώδικας matlab:

```
%A
% Exercise A.1
clear all;
close all;

T = 0.01; %Given Period
over = 10; %Oversampling factor
A = 4; %Half duration of the pulse in symbol periods
a = 0.5; %Roll-off factor
Ts = T / over; %sampling period
Fs = 1/Ts; % Sampling Frequency

% Number of samples
K = 1000;

% Step
% Nf = 1024;
Nf = 4096;

% Generate the pulse
[phi, t] = srrc_pulse(T, over, A, a);

% Frequency range
f_axis = linspace(-Fs/2,(Fs/2-Fs/Nf), Nf);

% Fourier Transform of Phi
phi_F = fftshift(fft(phi, Nf)*Ts);
abs_phi_F = abs(phi_F);
Energy_phi = power(abs_phi_F,2);
% Set figure
figure('Name','Energy.SRRC.logScale');
semilogy(f_axis,Energy_phi);
hold on;
title('Energy Spectrum of SRRC');
xlabel('Frequency');
ylabel('Logarithmic');
```

```

xlim([-Fs/2 Fs/2]);
hold off;

% Exercise A.2
N = 100;
% Bits generation
b = (sign(randn(N, 1)) + 1)/2;

% Generate 2-PAM
X_2pam = bits_to_2PAM(b);

% Upsample X_2pam
X_Upsampled = (1/Ts) * upsample(X_2pam, over);
X_Upsampled_time = 0:Ts:N*T-Ts;

% The Sum is the convolution X and phi
X = conv(X_Upsampled, phi)*Ts;
X_time = X_Upsampled_time(1)+t(1):Ts:X_Upsampled_time(end)+t(end);

% Set figure
figure('Name', '2pamConvPhi');
plot(X_time, X);
hold on;
title('X(t)');
xlabel('Time');
xlim([X_time(1) X_time(end)]);
hold off;

% Exercise A.3
% Fourier transform of X
X_F = fftshift(fft(X, Nf)*Ts);
% Periodogram Numerator - Denominator
Px_F_num = power(abs(X_F), 2);
PX_F_denom = X_time(end) - X_time(1);

% Periodogram of 2-PAM
Px_F_2PAM = Px_F_num/PX_F_denom;

```

```

%% Separate the plots
% Set figure
figure('Name','Periodograms');
subplot(1,2,1);
hold on;
plot(f_axis, Px_F_2PAM);
title('P_X(F)');
xlabel('Frequency');
xlim([-Fs/2 Fs/2]);
hold off;
hold on;
subplot(1,2,2);
semilogy(f_axis, Px_F_2PAM);
title('P_X(F)');
xlabel('Frequency');
ylabel('Logarithmic');
xlim([-Fs/2 Fs/2]);
hold off;

% A.3.2

Px_F_total = 0;

% K different bit arrays
for i=1:K
    % Bit array
    b = (sign(randn(N, 1)) + 1)/2;

    % 2-PAM
    X_2pam = bits_to_2PAM(b);

    % Upsample x_delta and upsampled time
    X_Upsampled = (1/Ts) * upsample(X_2pam, over);
    X_Upsampled_time = 0:Ts:N*T-Ts;

    % The Sum is the convolution X and phi

```

```

X = conv(X_Upsampled,phi)*Ts;
X_time = X_Upsampled_time(1)+t(1):Ts:X_Upsampled_time(end)+t(
end);

% Fourier transform of X
X_F = fftshift(fft(X, Nf)*Ts);
% Periodogram
Px_F_num = power(abs(X_F),2);
PX_F_denom = X_time(end) - X_time(1);
Px_F_total = Px_F_total + (Px_F_num./PX_F_denom);
end

Px_F_2PAM_average = Px_F_total./K;
Px_F_theoretical = 1/T .* Energy_phi;

% Set figure
figure('Name','Px_F.Average');
semilogy(f_axis,Px_F_2PAM_average);
hold on;
semilogy(f_axis,Px_F_theoretical);
title('P_X(F)');
xlim([-Fs/2 Fs/2]);
legend('Average','Theoretical');

% Oso afxananete to K kai to N dld plisiazoume
% sto apeiro toso tha

% A.4
% 4PAM symbols
X_4pam = bits_to_4PAM(b);

% Upsample x and upsampled time
X_Up = (1/Ts) * upsample(X_4pam, over);
X_Up_time = 0:Ts:(N/2)*T-Ts;

% The Sum is convolution X and phi
X = conv(X_Up,phi)*Ts;

```



```

X_time = X_Up_time(1)+t(1):Ts:X_Up_time(end)+t(end);

% Fourier transform of X
X_F = fftshift(fft(X, Nf)*Ts);
% Periodogram of 4_PAM
Px_F_4PAM = power(abs(X_F),2)/(X_time(end) - X_time(1));

% Set figure Px_F
figure('Name', 'Periodograms');
subplot(1,2,1);
hold on;
plot(f_axis, Px_F_4PAM);
title('P_X(F)');
xlabel('Frequency');
xlim([-Fs/2 Fs/2]);
hold off;
hold on;
subplot(1,2,2);
semilogy(f_axis, Px_F_4PAM);
title('P_X(F)');
xlabel('Frequency');
ylabel('Logarithmic');
xlim([-Fs/2 Fs/2]);
hold off;

% Set figure X(t)
figure('Name', '4pamConvPhi');
plot(X_time, X);
hold on;
title('Upsampled X_{4PAM}(t)');
xlabel('Time');
xlim([X_time(1) X_time(end)]);
hold off;

Px_F_4pamTotal = 0;

% K different bit arrays

```

```

for i=1:K
    % Bit array
    b = (sign(randn(N, 1)) + 1)/2;

    % 2-PAM
    X_4pam = bits_to_4PAM(b);

    % Upsample
    X_Up = (1/Ts) * upsample(X_4pam, over);
    X_Up_time = 0:Ts:(N/2)*T-Ts;

    % The Sum is the convolution X and phi
    X = conv(X_Up, phi)*Ts;
    X_time = X_Up_time(1)+t(1):Ts:X_Up_time(end)+t(end);

    % Fourier transform of X
    X_F = fftshift(fft(X, Nf)*Ts);
    % Periodogram
    Px_F_num = power(abs(X_F), 2);
    PX_F_denom = X_time(end) - X_time(1);
    Px_F_4pamTotal = Px_F_4pamTotal + (Px_F_num/PX_F_denom);
end

Px_F_4PAM_average = Px_F_4pamTotal/K;
Px_F_theoretical = 5/T * Energy_phi;

% Set figure
figure('Name', 'Px4pamF.Average');
semilogy(f_axis, Px_F_4PAM_average);
hold on;
semilogy(f_axis, Px_F_theoretical);
title('P_X(F)');
xlim([-Fs/2 Fs/2]);
legend('Average', 'Theoretical');

figure('Name', 'Periodograms');

```

```

subplot(1,2,1);
hold on;
plot(f_axis, Px_F_2PAM);
title('P_X(F)');
xlabel('Frequency');
xlim([-Fs/2 Fs/2]);
hold off;
hold on;
subplot(1,2,2);
semilogy(f_axis, Px_F_2PAM);
title('P_X(F)');
xlabel('Frequency');
ylabel('Logarithmic');
xlim([-Fs/2 Fs/2]);
hold off;

%figure Px_2PAM and Px_4PAM
figure('Name','Px_F.2-4.PAM');
subplot(1,2,1);
semilogy(f_axis, Px_F_2PAM_average);
hold on;
semilogy(f_axis, Px_F_4PAM_average);
title('P_X(F)');
xlim([-Fs/2 Fs/2]);
ylabel('Logarithmic');
legend('P_X(F)_{2PAM}', 'P_X(F)_{4PAM}');

subplot(1,2,2);
plot(f_axis, Px_F_2PAM_average);
hold on;
plot(f_axis, Px_F_4PAM_average);
title('P_X(F)');
xlim([-Fs/2 Fs/2]);
legend('P_X(F)_{2PAM}', 'P_X(F)_{4PAM}');

```

```

% Max amplitudes
% max_2PAM = max(PxF_2PAM_average);
% max_4PAM = max(PxF_4PAM_average);
% fprintf = ('Max of 2PAM periodogram is %.3f and of 4PAM
    periodogram %.3f', max_2PAM, max_4PAM )

%A
% Exercise A.5

T = 0.02; %Given Period
over = 20; %Oversampling factor
A = 4; %Half duration of the pulse in symbol periods
a = 0.5; %Roll-off factor
Ts = T / over; %sampling period
Fs = 1/Ts; % Sampling Frequency
K = 1000;
Nf = 4096;
% Generate the pulse
[phi, t] = srrc_pulse(T, over, A, a);

% Frequency range
f_axis = linspace(-Fs/2,(Fs/2-Fs/Nf), Nf);

% Fourier Transform of Phi
phi_F = fftshift(fft(phi, Nf)*Ts);
abs_phi_F = abs(phi_F);
Energy_phi = power(abs_phi_F,2);

% Exercise A.2
N = 100;
% Bits generation
b = (sign(randn(N, 1)) + 1)/2;

% Generate 2-PAM
X_2pam = bits_to_2PAM(b);

```

```

% Upsample X_2pam
X_Upsampled = (1/Ts) * upsample(X_2pam, over);
X_Upsampled_time = 0:Ts:N*T-Ts;

% The Sum is the convolution X and phi
X = conv(X_Upsampled,phi)*Ts;
X_time = X_Upsampled_time(1)+t(1):Ts:X_Upsampled_time(end)+t(end);

% Set figure
figure('Name','Energy.SRRC');
semilogy(f_axis,Energy_phi);
hold on;
title('Energy Spectrum of SRRC');
xlabel('Frequency');
xlim([-Fs/2 Fs/2]);
hold off;

% Fourier transform of X
X_F = fftshift(fft(X, Nf)*Ts);
% Periodogram Numerator - Denominator
Px_F_num = power(abs(X_F),2);
PX_F_denom = X_time(end) - X_time(1);

% Periodogram of 2_PAM
Px_F_2PAM = Px_F_num/PX_F_denom;

%% Separate the plots
% Set figure
figure('Name','Periodograms');
subplot(1,2,1);
hold on;
plot(f_axis, Px_F_2PAM);
title('P_X(F)');
xlabel('Frequency');
xlim([-Fs/2 Fs/2]);
hold off;
hold on;

```

```

subplot(1,2,2);
semilogy(f_axis, Px_F_2PAM);
title('P_X(F)');
xlabel('Frequency');
ylabel('Logarithmic');
xlim([-Fs/2 Fs/2]);
hold off;

% A.3.2

Px_F_total = 0;

% K different bit arrays
for i=1:K
    % Bit array
    b = (sign(randn(N, 1)) + 1)/2;

    % 2-PAM
    X_2pam = bits_to_2PAM(b);

    % Upsample x_delta and upsampled time
    X_Upsampled = (1/Ts) * upsample(X_2pam, over);
    X_Upsampled_time = 0:Ts:N*T-Ts;

    % The Sum is the convolution X and phi
    X = conv(X_Upsampled, phi)*Ts;
    X_time = X_Upsampled_time(1)+t(1):Ts:X_Upsampled_time(end)+t(
end);

    % Fourier transform of X
    X_F = fftshift(fft(X, Nf)*Ts);
    % Periodogram
    Px_F_num = power(abs(X_F), 2);
    PX_F_denom = X_time(end) - X_time(1);
    Px_F_total = Px_F_total + (Px_F_num./PX_F_denom);
end

```

```

PxF_2PAM_av_2T = PxF_total./K;
PxF_theoretical = 1/T .* Energy_phi;

% Set figure
figure('Name','PxF.Average');
semilogy(f_axis,PxF_2PAM_av_2T);
hold on;
semilogy(f_axis,PxF_theoretical);
title('P_X(F)');
xlim([-Fs/2 Fs/2]);
legend('Average','Theoretical');

% Set figure
figure('Name','T' = 2T');
semilogy(f_axis,PxF_2PAM_av_2T);
hold on;
semilogy(f_axis,PxF_2PAM_average);
title('P_X(F)');
xlim([-Fs/2 Fs/2]);
legend('P_X(F)_{2T}','P_X(F)_{T}');

figure('Name','T' = 2T');
plot(f_axis,PxF_2PAM_av_2T);
hold on;
plot(f_axis,PxF_2PAM_average);
title('P_X(F)');
ylabel('Logarithmic');
xlim([-Fs/2 Fs/2]);
legend('P_X(F)_{2T}','P_X(F)_{T}');

% Oso afxananete to K kai to N dld plisiazoume
% sto apeiro toso tha

%B
clear all;
close all;

```

```

T = 0.01; %Given Period
over = 10; %Oversampling factor
A = 4; %Half duration of the pulse in symbol periods
a = 0.5; %Roll-off factor
Ts = T / over; %sampling period
Fs = 1/Ts; % Sampling Frequency
K = 1000;
Nf = 4096;
% Generate the pulse
[phi, t] = srrc_pulse(T, over, A, a);

% Frequency range
f_axis = linspace(-Fs/2,(Fs/2-Fs/Nf), Nf);

% Fourier Transform of Phi
phi_F = fftshift(fft(phi, Nf)*Ts);
abs_phi_F = abs(phi_F);
Energy_phi = power(abs_phi_F,2);

% Exercise A.2
N = 100;
% Bits generation
b = (sign(randn(N, 1)) + 1)/2;

% Generate 2-PAM
X_2pam = bits_to_2PAM(b);

% Upsample X_2pam
X_Upsampled = (1/Ts) * upsample(X_2pam, over);
X_Upsampled_time = 0:Ts:N*T-Ts;

% The Sum is the convolution X and phi
X = conv(X_Upsampled,phi)*Ts;
X_time = X_Upsampled_time(1)+t(1):Ts:X_Upsampled_time(end)+t(end);

% % Set figure
% figure('Name','2pamConvPhi');

```



```

% plot(X_time,X);
% hold on;
% title('X(t)');
% xlabel('Time');
% xlim([X_time(1) X_time(end)]);
% hold off;
% 50 < f < 450
f = 200;
theta = 2*pi*rand(1);
Y1=X.*cos(2*pi*f*X_time + theta);

% Plot Y(t)
figure('Name','Y(t)');
plot(X_time,Y1);
hold on;
title('Y(t)');
xlabel('Time');
xlim([X_time(1) X_time(end)]);
hold off;
% Average Periodogram of Y(t)
PyF_total = 0;
for i=1:K
    % bit array
    b = (sign(randn(N, 1)) + 1)/2;
    % bit 2pam
    X_2pam = bits_to_2PAM(b);

    X_Upsampled = (1/Ts) * upsample(X_2pam, over);
    % X(t)
    X = conv(X_Upsampled,phi)*Ts;
    % Generate theta
    theta = 2*pi*rand(1);
    % Y(t)
    Y1 = X.*cos(2*pi*f*X_time + theta);

    Y = fftshift(fft(Y1, Nf)*Ts);

```

```

    % Periodogram
    PyF_TotalTime= X_time(end) - X_time(1);
    PyF_total = PyF_total + (power(abs(Y),2)./PyF_TotalTime);
end

PyF_average = PyF_total./K;
% Sx(f)
SxF = 1/T.*Energy_phi;

indexf0 = round(f/(Fs/Nf));

% SxF delayed
SxF1 = [zeros(1,indexf0) SxF(1:end-indexf0)];
SxF2 = [SxF(indexf0+1:end) zeros(1,indexf0)];

% SyF
SyF = 1/4*(SxF1 + SxF2);
figure;
semilogy(SyF)

figure('Name','Py_Average')
semilogy(f_axis,PyF_average);
hold on;
title('P_Y(F)');
xlabel('Frequency');
hold on;
semilogy(f_axis,SyF);

figure('Name','Py_Average')
plot(f_axis,PyF_average);
hold on;
title('P_Y(F)');
xlabel('Frequency');
hold on;
plot(f_axis,SyF);

```

```
legend('','')
```

```
function [X] = bits_to_2PAM(b)
% Conversion of bit array to 2-PAM Symbols
%
%
%      Input Bit b(i) /      0      /      1
%      -----
%      Output 2-PAM   /      +1     /      -1
%      Symbol X(i)   /
%
for i =1:length(b)
%      if b(i)~=0 or b(i)~=1
%          disp(['Bit in position' num2str(i-1) 'is not binary']);
%          continue;
%      end
      X(i)=(-1)^b(i);
end

% for i =1:length(b)
%      if (b(i) == 0)
%          X(i) = 1;
%      else
%          X(i) = -1;
%      end
% end
end
```

```
function [X] = bits_to_4PAM(b)
% Conversion of bit array to 4-PAM Symbols
%
%
%      Input Bit b(i) /  00  /  01  /  11  /  10  /
%      -----
%      Output 2-PAM   /  +3  /  +1  /  -1  /  -3  /
```

```
% Symbol  $X(i)$       /
```

```
for i = 1:2:length(b)
    bits = b(i:i+1);
    index = (i+1)/2;

    % B is type of (x1;x2;x3;x4;x5)
    if isequal(bits, [0;0])
        X(index) = +3;
    elseif isequal(bits, [0;1])
        X(index) = +1;
    elseif isequal(bits, [1;1])
        X(index) = -1;
    elseif isequal(bits, [1;0])
        X(index) = -3;
    end
end
end
end
```

```
function [phi, t] = srrc_pulse(T, over, A, a)
```

```
%
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% [phi, t] = srrc_pulse(T, over, A, a)
```

```
%
```

```
% OUTPUT
```

```
%
```

```
%      phi: truncated SRRC pulse, with parameter T,
```

```
%
```

```
%      roll-off factor a, and duration  $2*A*T$ 
```

```
%
```

```
%      t:   time axis of the truncated pulse
```

```
%
```

```
%
```

```
%
```

```
% INPUT
```

```

%
%      T: Nyquist parameter or symbol period (positive real
%      number)
%
%      over: positive integer equal to T/T_s (oversampling factor)
%
%
%      A: half duration of the pulse in symbol periods (positive
%      integer)
%
%      a: roll-off factor (real number between 0 and 1)
%
%
%
%
%      A. P. Liavas, Oct. 2020
%
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

Ts=T/over;

% Create time axis
t = [-A*Ts:A*Ts] + 10^(-8); % in order to avoid division by zero
% problems at t=0.

if (a>0 && a<=1)
    num = cos((1+a)*pi*t/T) + sin((1-a)*pi*t/T) ./ (4*a*t/T);
    denom = 1-(4*a*t./T).^2;
    phi = 4*a/(pi*sqrt(T)) * num ./ denom;
elseif (a==0)
    phi = 1/(sqrt(T)) * sin(pi*t/T)./(pi*t/T);
else
    phi = zeros(length(t),1);
    disp('Illegal value of roll-off factor')
    return
end

```