ΗΜΜΥ ΠΟΛΥΤΕΧΝΕΙΟΥ ΚΡΗΤΗΣ

ΤΗΛ 301-ΤΗΛΕΠΙΚΟΙΝΩΝΙΑΚΑ ΣΥΣΤΗΜΑΤΑ Ι

ΧΕΙΜΕΡΙΝΟ ΕΞΑΜΗΝΟ 2017-2018

Εργαστηριακή Άσκηση 3

Φοιτητής
ΚΑΛΟΓΕΡΑΚΗΣ ΣΤΕΦΑΝΟΣ
AM:2015030064

Διδάσκων
Α. ΛΙΑΒΑΣ

Technical
University
of Crete

Περιεχόμενα

Α′ Μέρος Α

Α′.1 Δημιουργία δυαδικής ακολουθίας N Bit

Στο πρώτο ερώτημα της τρίτης άσκησης κληθήκαμε να δημιουργήσουμε δυαδική ακολουθία με στοιχεία 4N ισοπίθανα bits όπως κάναμε και στις δύο προηγούμενες εργαστηριακές ασκήσεις. Το N που επιλέξαμε ήταν $N = 300$.

```
%A.1
%create demanded 4n bit series for random N
N_bits = 300;
b = (sign(randn(4*N_bits,1))+1)/2;
```

Σχήμα 1: Κώδικας μέρους Α.1

Α′.2 Δημιουργία συνάρτησης bits to 4 PAM

Στην συνέχεια, δημιουργήσαμε την συνάρτηση bits to 4 PAM(bit seq,A). Η λειτουργία της συνάρτησης αυτής, είναι ότι απεικονίζει την δυαδική ακολουθία εισόδου στις Xi και Xq ακολουθίες 4 PAM χρησιμοποιώντας την κωδικοποίηση Gray. Η υλοποίηση της συνάρτησης φαίνεται στην παρακάτω εικόνα

```
function [X] = bits_to_4_PAM(b,A)
    %counter so that we don't have zeros
    counter=1;
    %create array of all possible outcomes
    Y = [-3*A, -1*A, A , 3*A];
    %creating space from the beginning for length(b)/2 since we have 4PAM
    X=zeros(1,length(b)/2);
    for i=1:2:length(b)
        if(b(i)==0 && b(i+1)==0)
            X(counter) = Y(1);
        elseif(b(i)==0 && b(i+1)==1)
            X(counter) = Y(2);
        elseif(b(i)==1 && b(i+1)==1)
            X(counter) = Y(3);
        elseif(b(i)==1 && b(i+1)==0)
            X(counter) = Y(4);
        end
        counter=counter+1;
    end
end
```

Σχήμα 2: Συνάρτηση ερωτήματος Α.2

Α′.3 Δημιουργία X_i, X_q

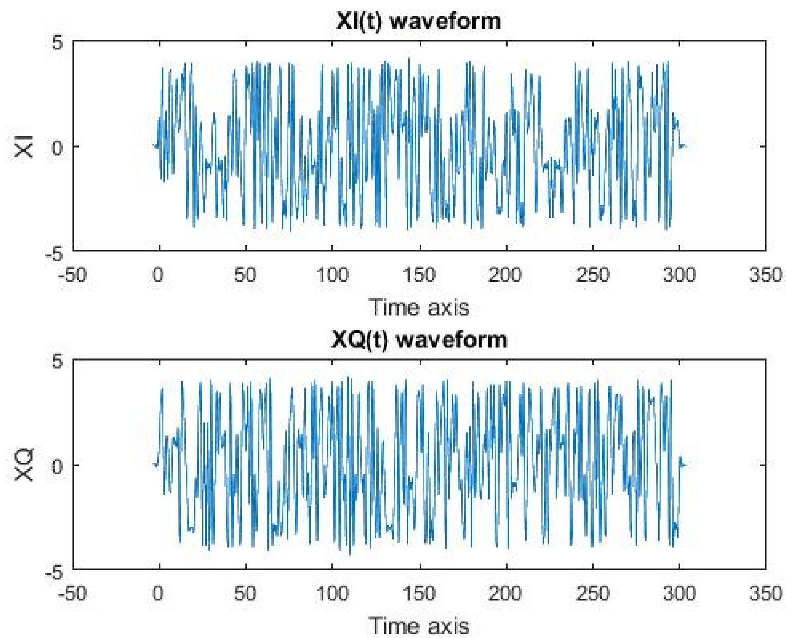
Έπειτα δημιουργήσαμε και απεικονίσαμε τα πρώτα 2 Nbits της ακολουθίας X_i και τα επόμενα 2 Nbits της X_q

```
%A.3
%design first 2N_bits at 4PAM symbols Xi and next 2N_bits at 4PAM symbols
Xi = X(1:N_bits);
Xq = X(N_bits+1:2*N_bits);
```

Σχήμα 3: Κώδικας μέρους Α.3

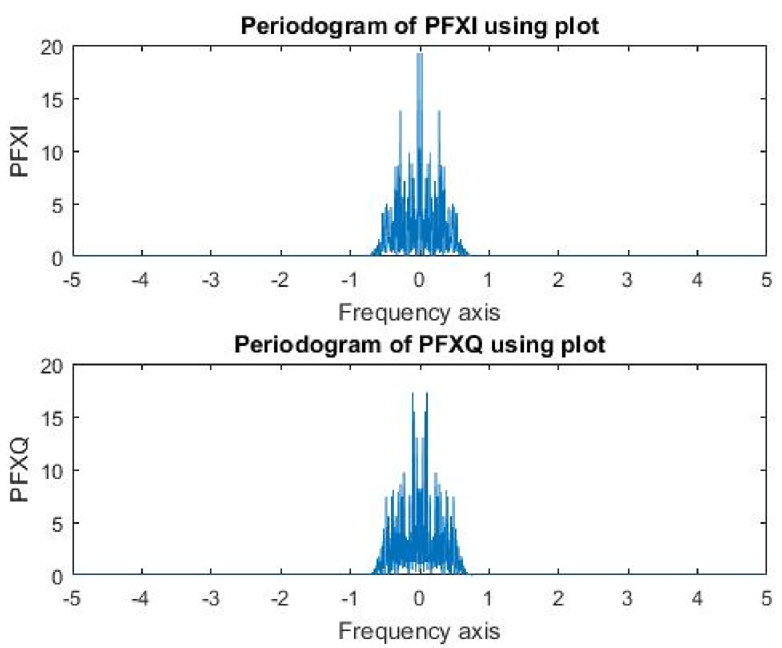
Α΄.4 Σχηματισμός κυματομορφών εξόδου X_i, X_q

Φιλτράραμε τις ακολουθίες που δημιουργήσαμε στο προηγούμενο ερώτημα από SRRC φίλτρο για $T = 1sec$, $over = 10$,$Ts = 1/Ts$, $a = 0.5$, $A = 4$. Ακολουθούν οι κυματομορφές:

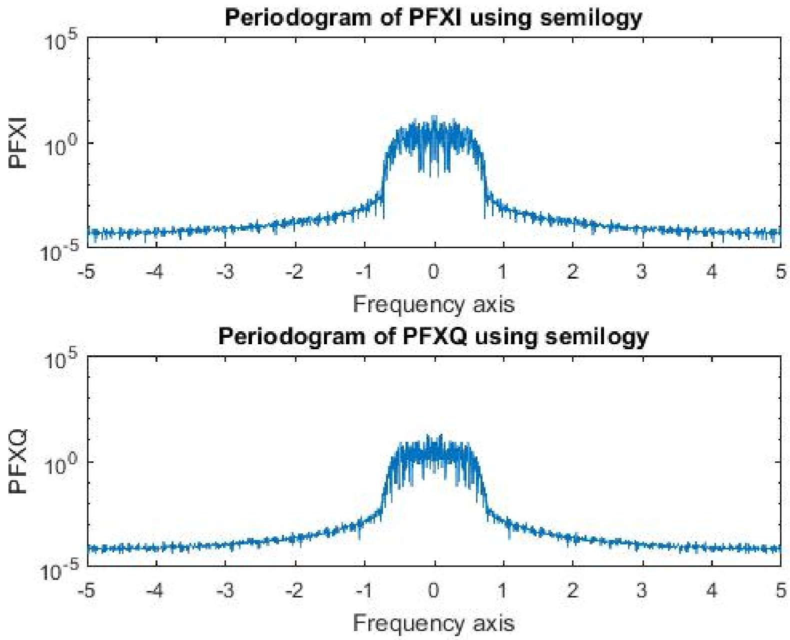


Σχήμα 4: Κυματομορφές X_i, X_q

Οι επόμενες κυματομορφές είναι τα περιοδογράμματα των X_i, X_q με την χρήση plot και semilogy:



Σχήμα 5: Περιοδογραμμα X_i, X_q με χρήση plot



Σχήμα 6: Περιοδογραμμα X_i, X_q με χρήση semilogy

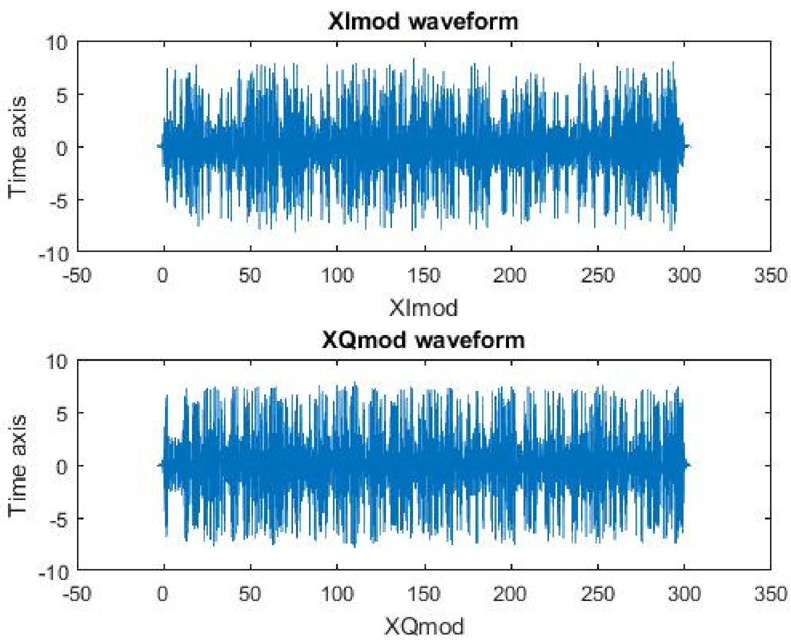
Παρακάτω επισυνάπτεται ενδεικτικός κώδικας ερωτήματος A.4

```
%A.4
%given values
T=1;
over=10;
Ts = T/over;
Fs = 1/Ts;

%some random values for a and A to create phi signal
A1 = 4;
a = 0.5;
N = 2048; %we use a large N

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

[phi,t] = srrc_pulse(T, Ts, A1 ,a);%Time will cover from 0 until time of N symbol

%first create signals
Xi_delta = 1/Ts * upsample(Xi,over);
Xi_delta_conv = conv(Xi_delta, phi)*Ts;
Xq_delta = 1/Ts * upsample(Xq,over);
Xq_delta_conv = conv(Xq_delta, phi)*Ts;

%then define time vector
T_plot = 0:Ts:N_bits-Ts;
ti_conv = linspace(T_plot(1)+t(1), T_plot(end)+t(end),length(Xi_delta_conv));%generates length(X_delta_conv) points
tq_conv = linspace(T_plot(1)+t(1), T_plot(end)+t(end),length(Xq_delta_conv));%generates length(X_delta_conv) points
```

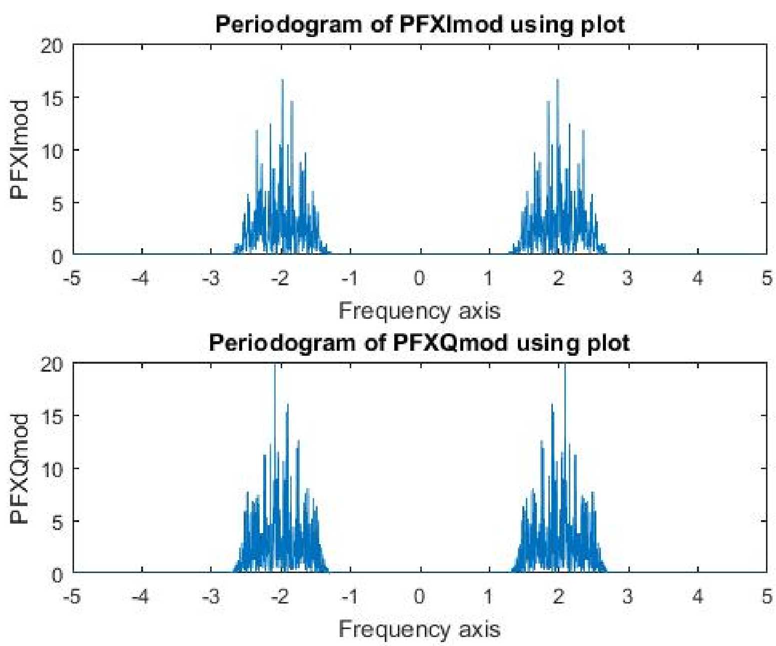
Σχήμα 7: Κώδικας μέρους A.4

Α′.5 Σχηματισμός κυματομορφών X_imod, X_qmod

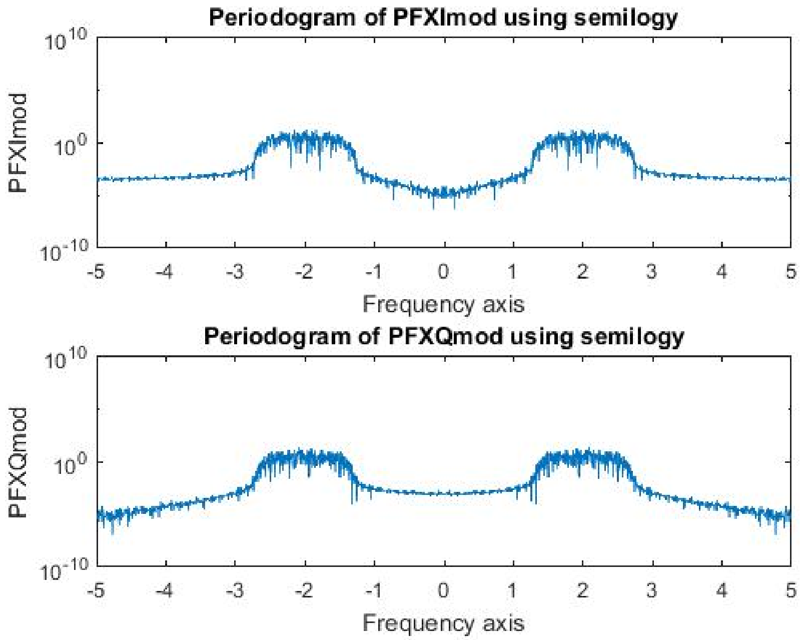
Για τον σχηματισμό των κυματομορφών των X_imod, X_qmod πολλαπλασιάσαμε τις ακολουθίες X_i, και X_q ,με τους φορείς τους $2cos(2 * pi * Fo * t)$ και $-2sin(2 * pi * Fo * t)$ αντίστοιχα για $Fo = 2Hz$ με σκοπό την διαμόρφωσή τους. Ακολουθούν όπως το προηγούμενο ερώτημα οι κυματομορφές των τελικών σημάτων όπως και τα περιοδογράμματα σε plot και semilogy



Σχήμα 8: Κυματομορφές X_imod, X_qmod



Σχήμα 9: Περιοδογραμμα X_imod, X_qmod με χρήση plot



Σχήμα 10: Περιοδογραμμα X_imod, X_qmod με χρήση semilogy

Παρακάτω επισυνάπτεται ενδεικτικός κώδικας ερωτήματος A.5

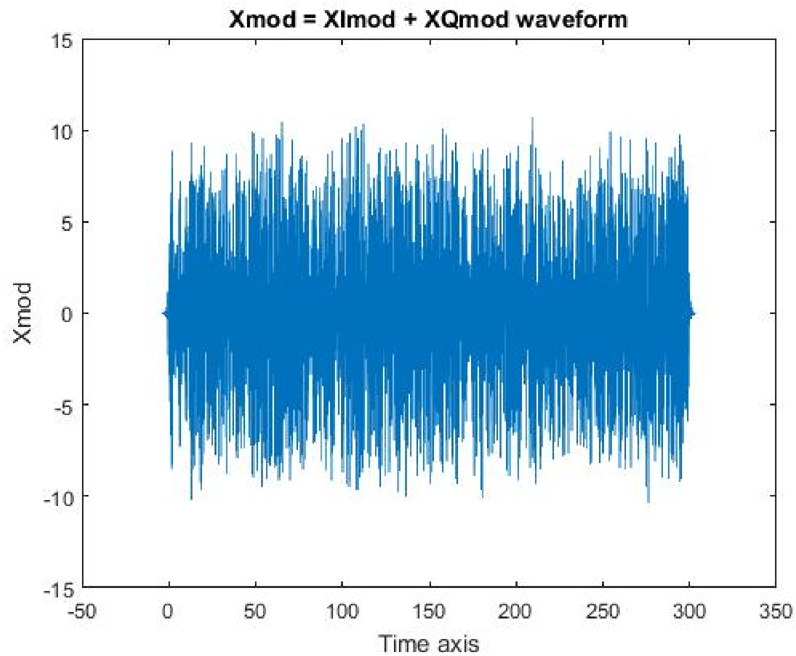
```
%A.5
%data given from exercise
Fo = 2;
Ximod = 2*Xi_delta_conv.*cos(2*pi*Fo*ti_conv);
Xqmod = -2*Xq_delta_conv.*sin(2*pi*Fo*tq_conv);

%design Ximod and Xqmod
figure;
subplot(2,1,1);
plot(ti_conv,Ximod);
title('XImod waveform');
xlabel('XImod');
ylabel('Time axis');
subplot(2,1,2);
plot(tq_conv,Xqmod);
title('XQmod waveform');
xlabel('XQmod');
ylabel('Time axis');

%find periodogram of those signals
PXFimod = ((abs(fftshift(fft(Ximod,N))).^2)*Ts)./Ti_total;
PXFqmod = ((abs(fftshift(fft(Xqmod,N))).^2)*Ts)./Tq_total;
```

Σχήμα 11: Κώδικας μέρους A.5

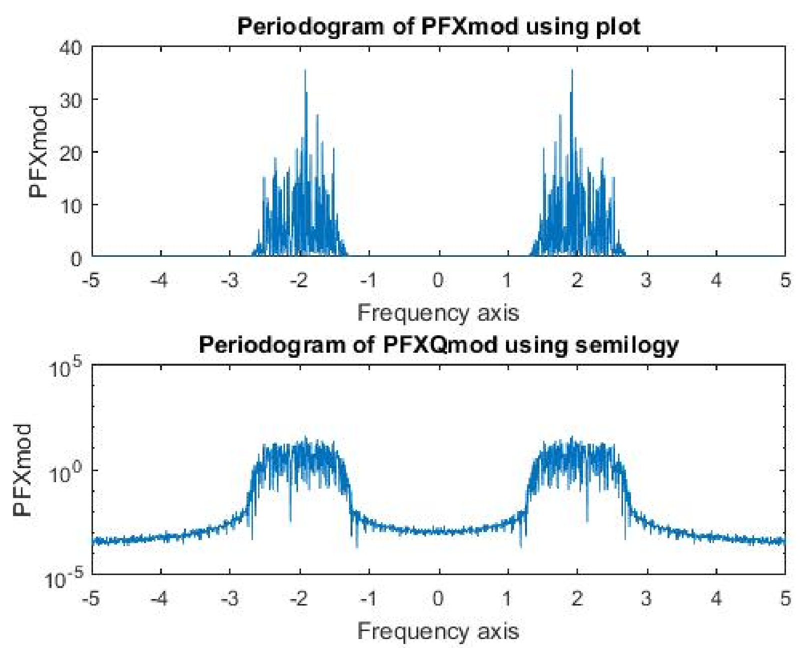
Παρατηρώντας τις κυματομορφές βλέπουμε μετακίνηση του φάσματος στην περιοχή $-2Hz$ και $2Hz$ και δύο μέγιστα αντί για ένα. Τα παραπάνω είναι φυσιολογικά αφού τα σήματα πολλαπλασιάστηκαν με δύο ημιτονοειδή σήματα των οποίων το φάσμα είναι στην περιοχή $-2Hz$ και $2Hz$

Α′.6 Άθροιση κυματομορφών X_imod, X_qmod

Στο έκτο ερώτημα ζητήθηκε να σχηματίσουμε και να σχεδιάσουμε την είσοδο $Qmod$ του καναλιού. Η είσοδος του καναλιού είναι το άθροισμα των X_imod, X_qmod που υπολογίσαμε στο προηγούμενο ερώτημα. Η κυματομορφή εισόδου και τα περιοδογράμματα ακολουθούν παρακάτω:



Σχήμα 12: Κυματομορφή εισόδου $Xmod$



Σχήμα 13: Περιοδογραμματα $Xmod$

Παρατηρείται ένα αρκετά πυκνό φάσμα στην $Xmod$ που οφείλεται στο άθροισμα των Q_i και Q_q ενώ αυξάνεται και το πλάτος μετά το άθροισμα. Επίσης, το φάσμα παραμένει μετατοπισμένο γύρω από το $Fo = 2Hz$

Ενδεικτικός κώδικας ερωτήματος Α.6

```
%A.6
XmodTotal = Ximod + Xqmod;

figure;
%ti_conv is the same as tq_conv so it does not matter which one we choose
plot(ti_conv ,XmodTotal);
title('Xmod = XImod + XQmod waveform');
ylabel('Xmod');
xlabel('Time axis');

PXFmodTotal = ((abs(fftshift(fft(XmodTotal,N))).^2)*Ts)./Ti_total;
```

Σχήμα 14: Κώδικας μέρους Α.6

Α′.7 Προσθήκη Gaussian θορύβου

Υποθέτοντας ότι το κανάλι είναι ιδανικό, προσθέτουμε λευκό Gaussian θόρυβο $W(t)$ στην έξοδο του καναλιού με διασπορά σ_W^2 όπως δίνεται από την εκφώνηση. Χρησιμοποιούμε την συνάρτηση rand για να μεταβάλουμε τυχαία το πλάτος του σήματος και να προσομοιωθεί έτσι η επίδραση του θορύβου.

```
%A.7 & A.8
%Add gaussian noise
SNR = 22;
var_w = (10*A^2)/(Ts*(10^(SNR/10)));
W_sig = sqrt(var_w)*randn(1, length(XmodTotal));
W = W_sig + XmodTotal;
```

Σχήμα 15: Κώδικας μέρους Α.8

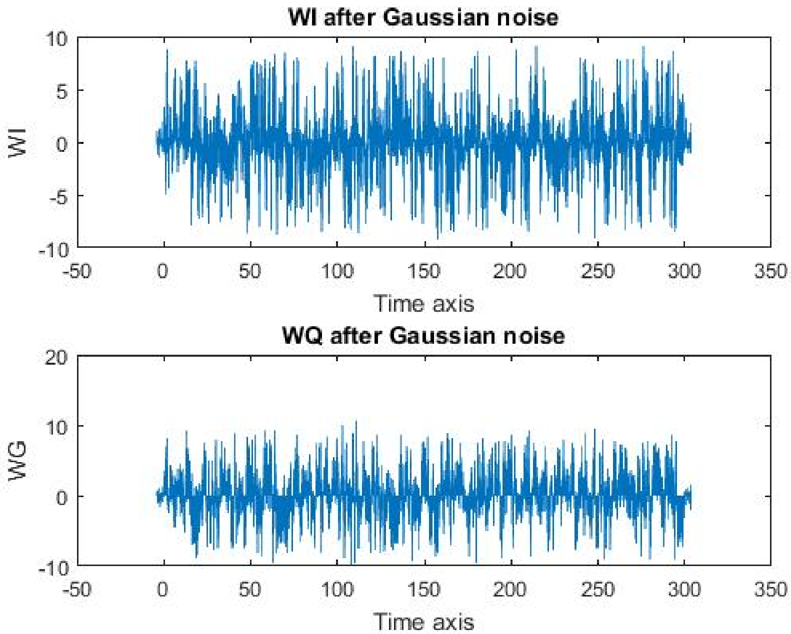
Α′.8 Διακλάδωση ενθόρυβης κυματομορφής

Στο συγκεκριμένο ερώτημα, διακλαδώνουμε την ενθόρυβη κυματομορφή και την πολλαπλασιάζουμε με τους φορείς $\cos(2pFot)$ και $-\sin(2pFot)$ αντίστοιχα με στόχο την λήψη του σήματος από τον δέκτη.

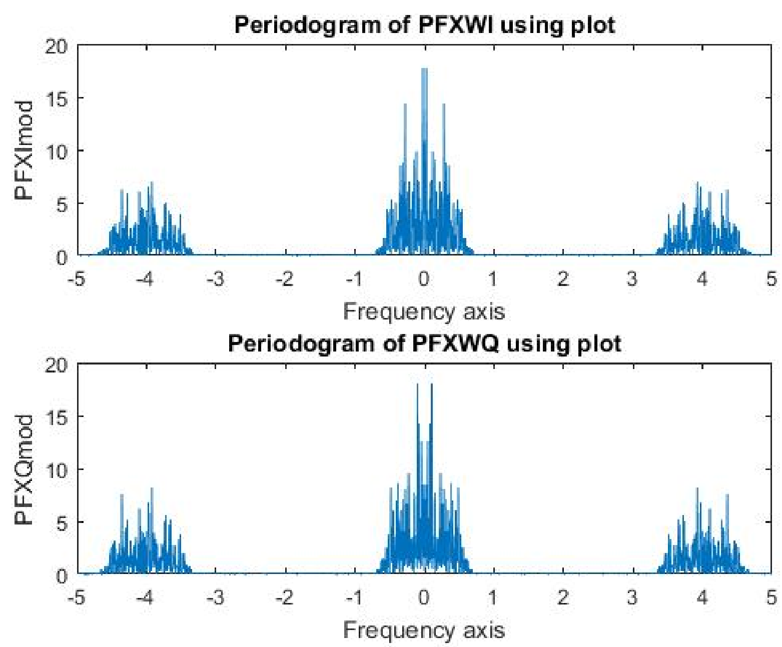
```
%A.9
WI = W.*cos(2*pi*Fo*ti_conv);
WQ = W.*(-sin(2*pi*Fo*tq_conv));

%show both WI and WQ
figure;
subplot(2,1,1);
plot(ti_conv ,WI);
title('WI after Gaussian noise');
ylabel('WI');
xlabel('Time axis');
subplot(2,1,2);
plot(ti_conv ,WQ);
title('WQ after Gaussian noise');
ylabel('WG');
xlabel('Time axis');

PXFWI = ((abs(fftshift(fft(WI,N))).^2)*Ts)./Ti_total;
PXFWQ = ((abs(fftshift(fft(WQ,N))).^2)*Ts)./Tq_total;
```

Σχήμα 16: Κώδικας μέρους Α.9

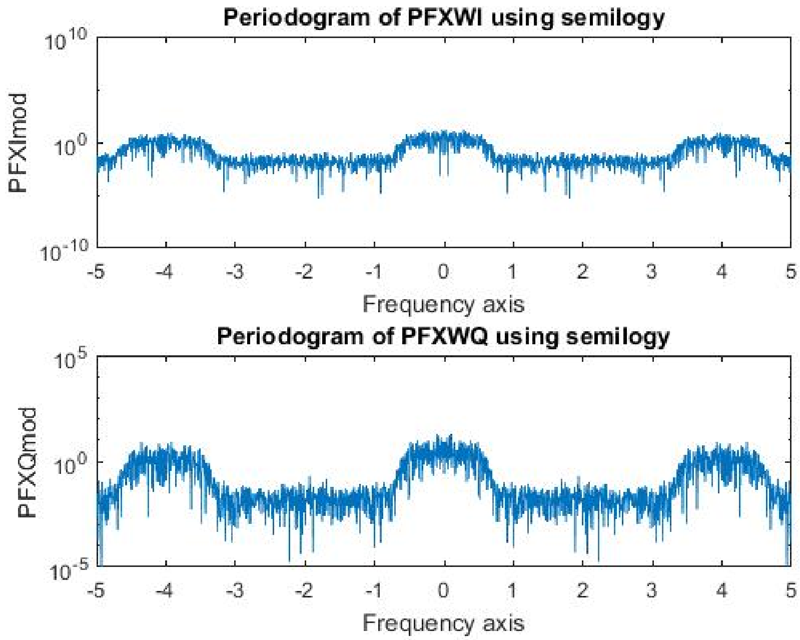
Ακολουθούν οι κυματομορφές και τα περιοδογράμματα που προέκυψαν



Σχήμα 17: Κυματομορφές WI, WQ



Σχήμα 18: Περιοδογραμμα WI, WQ με χρήση plot

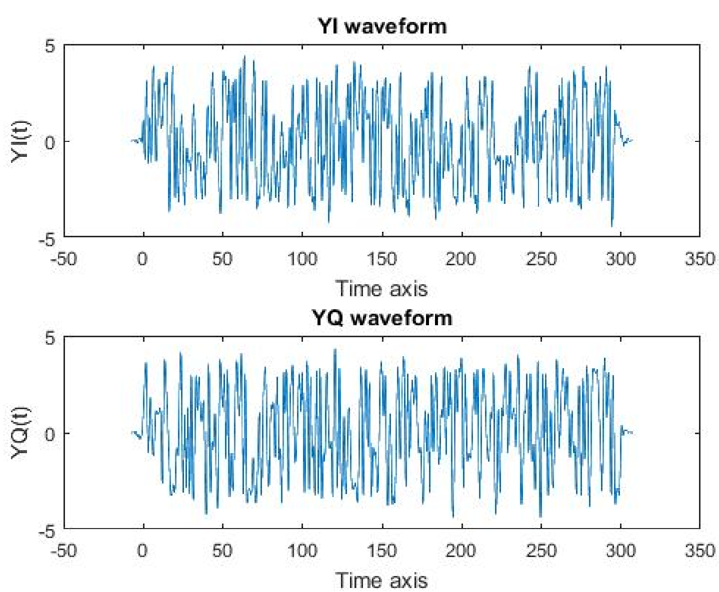


Σχήμα 19: Περιοδογραμμα WI, WQ με χρήση semilogy

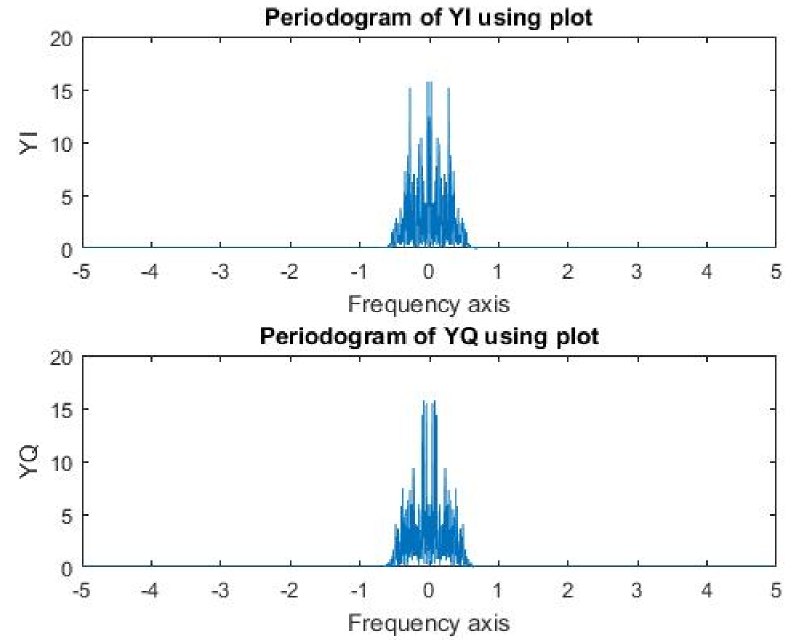
Παρατηρούμε ότι μετά τον θόρυβο έχει μεταβληθεί το εύρος φάσματος παρουσιάζοντας σύγκλιση στις συχνότητες $-4Hz$ και $4Hz$

A′.9 Πέρασμα ενθόρυβων κυματομορφών από SRRC φίλτρα

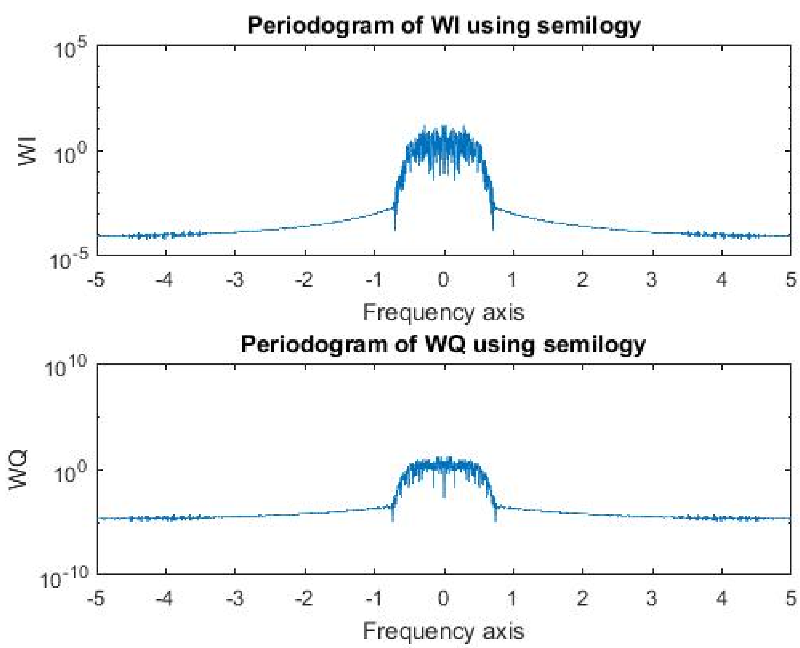
Ως συνέχεια της άσκησης περάσαμε τις υπολογισμένες κυματομορφές από προσαρμοσμένα φίλτρα (SRRC). Οι κυματομορφές που προέκυψαν και ακολούθως τα περιοδογράμματα τους ακολουθούν:



Σχήμα 20: Κυματομορφές YI, UQ



Σχήμα 21: Περιοδογραμμα UI, UQ με χρήση plot



Σχήμα 22: Περιοδογραμμα UI, UQ με χρήση semilogy

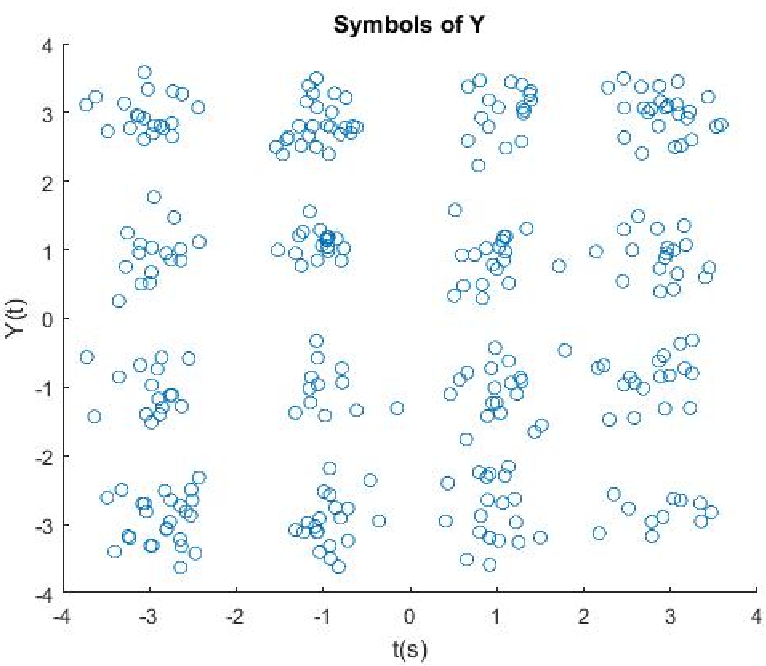
Είναι εμφανές ότι το προσαρμοσμένο φίλτρο (SRRC) επιτρέπει μόνο τις συχνότητες βασικής ζώνης να περάσουν γεγονός που συμβαίνει και στα δύο σήματα.

```
%A.10
sigWI = Ts*conv(WI,phi);
sigWQ = Ts*conv(WQ,phi);

ti_conv_sig = linspace(ti_conv(1)+t(1), ti_conv(end)+t(end),length(sigWI));
tq_conv_sig = linspace(tq_conv(1)+t(1), tq_conv(end)+t(end),length(sigWQ));

figure;
subplot(2,1,1);
plot(ti_conv_sig, sigWI);
title('YI waveform');
ylabel('YI(t)');
xlabel('Time axis');
subplot(2,1,2);
plot(tq_conv_sig, sigWQ);
title('YQ waveform');
ylabel('YQ(t)');
xlabel('Time axis');

%compute periodogramms
PXFYI = ((abs(fftshift(fft(sigWI,N))).^2)*Ts)./Ti_total;
PXFYQ = ((abs(fftshift(fft(sigWQ,N))).^2)*Ts)./Tq_total;
```

Σχήμα 23: Ενδεικτικός κώδικας μέρους Α.10

Α′.10 Δειγματοληψία εξόδου προσαρμοσμένων SRRC φίλτρων

Για να είναι εμφανής ο $16-QAM$ αστερισμός μας ζητήθηκε να δειγματοληπτήσουμε την έξοδο των προσαρμοσμένων φίλτρων και σχεδιάσουμε την ακολουθία χρησιμοποιώντας την συνάρτηση scatterplot της matlab. *Αξίζει να σημειωθεί πραγματοποιήθηκε 'tail-cutting' αποκόπτοντας τις τιμές που είναι ικανές να δημιουργήσουν πρόβλημα κατά την εκτίμηση των σφαλμάτων*



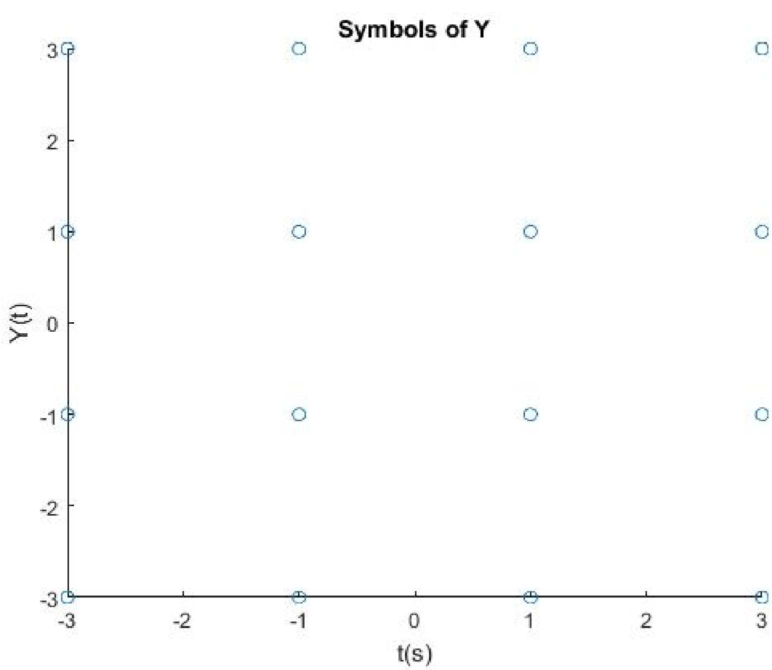
Σχήμα 24: Scatterplot μετά από δειγματοληψία

```
%A.11
%find all the negative values in time domain with the following loop
cutoffCounter=0;
for p=1:length(ti_conv_sig)
    if(ti_conv_sig(1,p)<0)
        cutoffCounter = cutoffCounter+1;
    else
        break;
    end
end

%take only the positive values
YIAfteCutoff = sigWI(cutoffCounter:(length(ti_conv_sig) - (cutoffCounter+1)));
YQAfterCutoff = sigWQ(cutoffCounter:(length(tq_conv_sig) - (cutoffCounter+1)));

%decreases sample rate by over=10
YIdecresed = downsample(YIAfteCutoff,over);
YQdecreased = downsample(YQAfterCutoff,over);

figure();
scatter(YIdecresed,YQdecreased);
title('Symbols of Y');
xlabel('t(s)');
ylabel('Y(t)')
```

Σχήμα 25: Ενδεικτικός κώδικας μέρους A.11

A′.11 Δημιουργία συνάρτησης detect 4 PAM

Σκοπός του συγκεκριμένου ερωτήματος ήταν η δημιουργία της συνάρτησης detect 4 PAM. Η συγκεκριμένη συνάρτηση υπακούει στο κανόνα του εγγύτερου γείτονα για να αποφασίσει για την ακολουθία εισόδου σύμβολο προς σύμβολο.Εφαρμόσαμε την συνάρτηση και στις δυο ακολουθίες Υ που χρησιμοποιήθηκαν προηγουμένως και προέκυψαν δύο εκτιμώμενες ακολουθίες τις οποίες και απεικονίσαμε με την εντολή scatterplot. Με τον τρόπο αυτό υπολογίστηκε και ο αριθμός σφαλμάτων απόφασης στα επόμενα ερωτήματα.



Σχήμα 26: Scatterplot μετά από detect 4PAM

```
function [PAMsymbols] = detect_4_PAM(data, A)
%we need to find which is point closer to our data input
X = [-3*A, -1*A, A , 3*A];

%initialize with zeros
PAMsymbols=zeros(1,length(data));

for i=1:length(data)
    %we find for each element of data the distance from each X
    ds1 = norm(X(1)-data(1,i));
    ds2 = norm(X(2)-data(1,i));
    ds3 = norm(X(3)-data(1,i));
    ds4 = norm(X(4)-data(1,i));

    %find the minimum value
    min_val = min([ds1,ds2,ds3,ds4]);
    %check every time which distance is the shortest
    if(ds1== min_val)
        PAMsymbols(1,i) = X(1);
    elseif(ds2 == min_val)
        PAMsymbols(1,i) = X(2);
    elseif(ds3 == min_val)
        PAMsymbols(1,i) = X(3);
    elseif(ds4 == min_val)
        PAMsymbols(1,i) = X(4);
    end
end

end
```

Σχήμα 27: Ενδεικτικός κώδικας συνάρτησης detect 4PAM

Α΄.12 Αριθμός σφαλμάτων απόφασης συμβόλου αστερισμού 16-QAM

Για να υπολογίσουμε τον αριθμό σφαλμάτων απόφασης συμβόλου 16-QAM ελέγχθηκαν τα σύμβολα εξόδου αν είναι ίδια με αυτά της εισόδου. Μετά από την υλοποίηση του κώδικα προέκυψαν 0 σφάλματα απόφασης συμβόλου.

```
%A.13
ErrorI = 0;
ErrorQ = 0;

%Xi and Xq have the same length so it does not really matter which one we
%choose
for i=1:length(Xi)
    %compute I and Q errors seperately
    %also use round function to scale the number into the nearest int and
    %the comparison can be done easier
    if(YIestimate(i) ~= round(Xi(i)))
        ErrorI = ErrorI + 1;
    end
    if(YQestimate(i) ~= round(Xq(i)))
        ErrorQ = ErrorQ + 1;
    end
end
%compute total Errors
totalError = ErrorI+ErrorQ;

disp(['Total Errors ',num2str(totalError),' with ',num2str(ErrorI),' I Errors and ',num2str(ErrorQ)  ,' Q Error']);
```

Σχήμα 28: Ενδεικτικός κώδικας ερωτήματος Α.13

Α΄.13 Δημιουργία συνάρτησης PAM 4 to bits

Στο συγκεκριμένο ερώτημα δημιουργήθηκε η συνάρτηση PAM 4 to bits η οποία χρησιμοποιεί την αντίστροφη απεικόνιση Gray. Πιο συγκεκριμένα, μετατρέπει τα σύμβολα σε δυάδες bits, και υπολογίζει απο τις αποφάσεις για τις ακολουθίες συμβόλων εισόδου την εκτιμώμενη δυαδική ακολουθία εισόδου.Αποτελεί ουσιαστικά την αντίστροφη συνάρτηση της bits to 4 PAM που χρησιμοποιήσαμε στην αρχή της άσκησης.

```
function [revBits] = PAM_4_to_bits(X,A)
%the opposite from bits_to_4_PAM
%transforms PAM_4 decoding to bits sequence

Y = [-3*A, -1*A, A , 3*A];

counter=1;
%again create default space
revBits = zeros(1,2*length(X));
for i=1:length(X)
    if(X(i) == Y(1))
        revBits(counter) =0;
        revBits(counter+1) = 0;
    elseif(X(i) == Y(2))
        revBits(counter) =0;
        revBits(counter+1) = 1;
    elseif(X(i) == Y(3))
        revBits(counter) =1;
        revBits(counter+1) = 1;
    elseif(X(i) == Y(4))
        revBits(counter) =1;
        revBits(counter+1) = 0;
    end
    %the counter increases by 2 for
    counter = counter+2;
end

end
```

Σχήμα 29: Συνάρτηση PAM 4 to bits

Α΄.14 Υπολογισμός αριθμού σφαλμάτων bit

Στο τελευταίο ερώτημα του πρώτου μέρους κληθήκαμε να υπολογίσουμε τον αριθμό σφαλμάτων bit. Ουσιαστικά ελέγξαμε αν η εκτιμώμενη ακολουθία εξόδου που υπολογίστηκε στο προηγούμενο ερώτημα είναι ίδια με αυτή που δημιουργήσαμε στην αρχή. Το αποτέλεσμα που προκύπτει είναι και πάλι 0 πράγμα που δείχνει ότι ανακτήθηκαν επιτυχώς όλα τα σύμβολα.

```
%A.15
bit_error = 0;
for i=1:length(b)
    if(b(i) ~= est_bit(i))
        bit_error = bit_error + 1;
    end
end
disp(['Bit error: ',num2str(bit_error)]);
```

Σχήμα 30: Ενδεικτικός κώδικας ερωτήματος Α.15

Β′ Μέρος Β

Β′.1 Εκτίμηση πιθανότητας σφάλματος συμβόλου/bit με πείραμα ανεξάρτητων επαναλήψεων (μέθοδος Monte Carlo)

Στο δεύτερο μέρος της άσκησης ζητήθηκε να εκτιμήσουμε την πιθανότητα σφάλματος συμβόλου και bit με την χρήση της μεθόδου Monte Carlo.

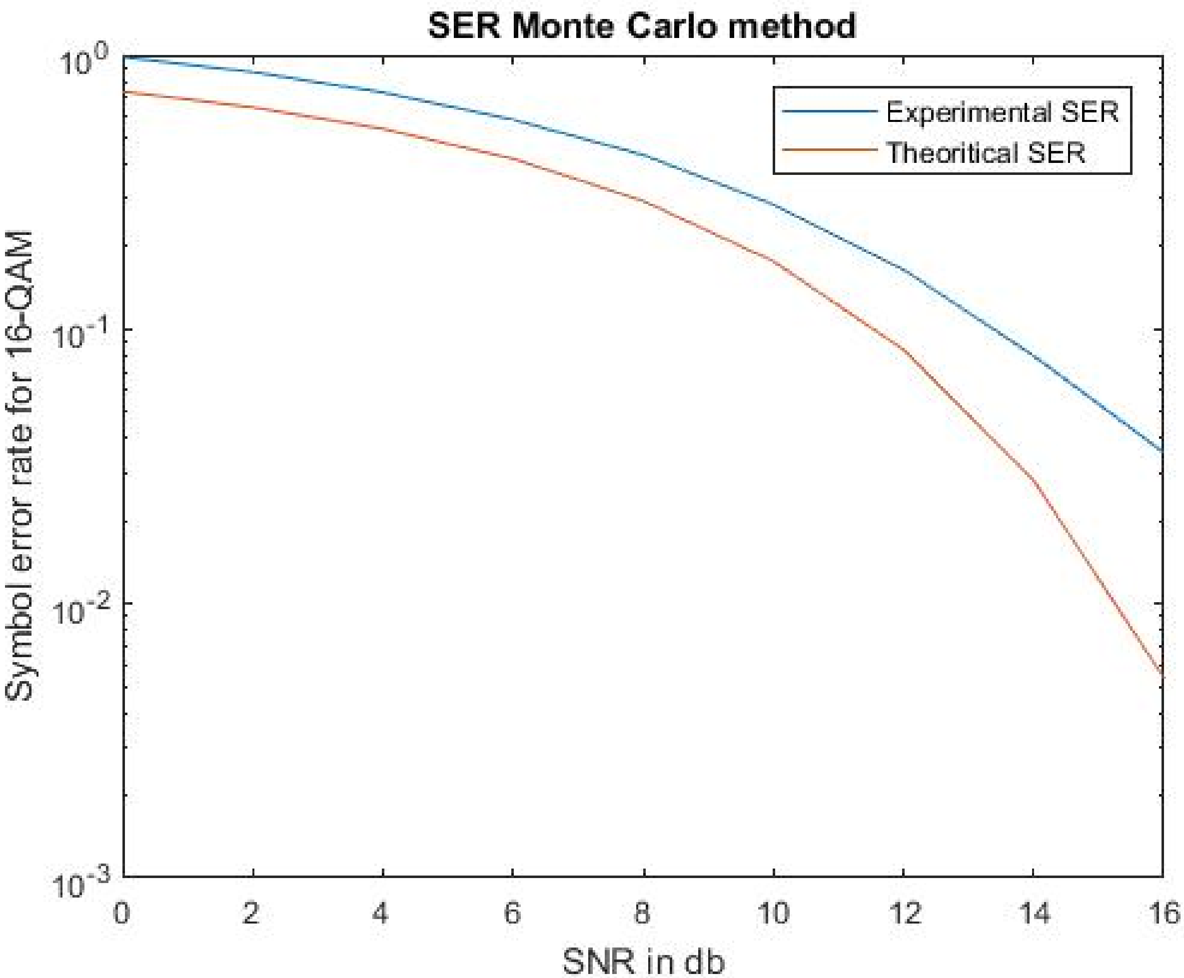
Ο κώδικας που χρησιμοποιήθηκε ήταν αντίστοιχος με το Α μέρος με την διαφορά ότι αυτή την φορά υπολογίσαμε τα σφάλματα για διαφορετικές τιμές του SNR που δίνονταν από την εκφώνηση. Επιπροσθέτως, υπολογίστηκε η θεωρητική προσέγγιση των σφαλμάτων συμβουλευόμενοι πηγές που φαίνεται στο μέρος του κώδικα που επισυνάπτεται παρακάτω

```
        ErrorI = 0;
        ErrorQ = 0;
        %Xi and Xq have the same length so it does not really matter which one we
        %choose
        for i=1:length(Xi)
        %compute I and Q errors seperately
        %also use round function to scale the number into the nearest int and
        %the comparison can be done easier
            if(YIestimate(i) ~= round(Xi(i)))
                ErrorI = ErrorI + 1;
            end
            if(YQestimate(i) ~= round(Xq(i)))
                ErrorQ = ErrorQ + 1;
            end
        end
        %compute total Errors
        totalError = totalError+ ErrorI+ErrorQ;

        est_bitI = PAM_4_to_bits(YIestimate, A);
        est_bitQ = PAM_4_to_bits(YQestimate, A);

        est_bit = zeros(1,4*N_bits);
        est_bit(1:2*N_bits) = est_bitI;
        est_bit(2*N_bits+1:4*N_bits) = est_bitQ;
        for i=1:length(b)
            if(b(i) ~= est_bit(i))
                bit_error = bit_error + 1;
            end
        end
    end
      IQerr_exp(1,n) =totalError/(N_bits*K);
      ber_exp(1,n) = bit_error/(N_bits*K*4);
end

%sources for theoritical T_symbol and T_bit
%https://www.mathworks.com/matlabcentral/fileexchange/30074-ber-curve-for-qam-16-in-gaussian-environment?focusec
%http://www.academia.edu/5770384/Simulation_of_OFDM_and_BERvsSNR_plots_in_matlab
%Check if correct
T_symbol = 3/2.*erfc(sqrt(0.1*(10.^(SNRdb/10))))-(1/4)*3/2.*erfc(sqrt(0.1*(10.^(SNRdb/10))));
T_bit = (1/4)*3/2.*T_symbol;
```

Σχήμα 31: Ενδεικτικός κώδικας ερωτήματος Β.1

Β΄.2 Σχεδιασμός θεωρητικής και πειραματικής πιθανότητας σφάλματος συμβόλου

Στην συνέχεια, ζητήθηκε ο σχεδιασμός θεωρητικής και πειραματικής πιθανότητας σφάλματος συμβόλου συναρτήσει του SNR σε κοινό semilogy. Όλα τα ζητούμενα υπολογίστηκαν στο προηγούμενο ερώτημα με το διάγραμμα που προκύπτει να είναι

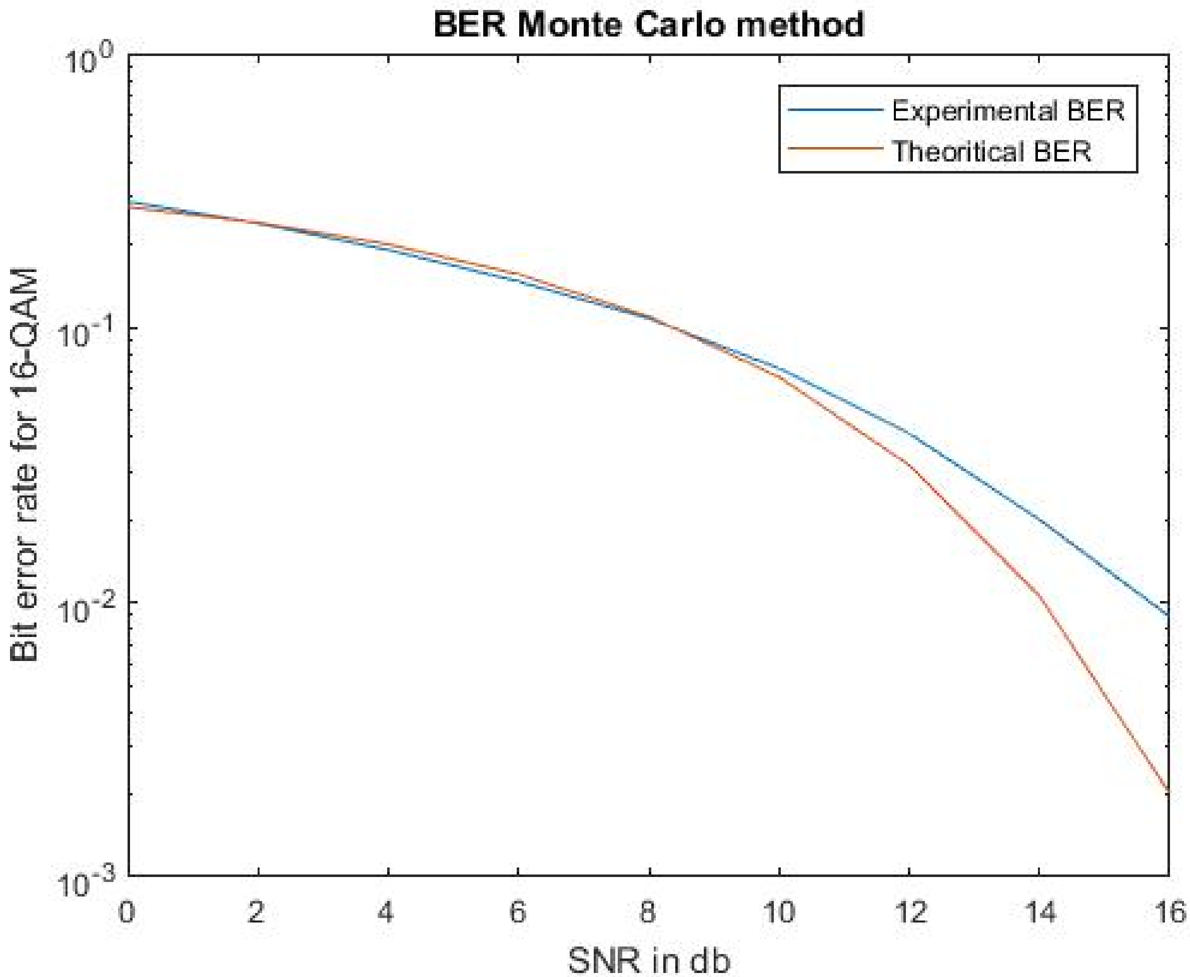


Σχήμα 32: Θεωρητική και πειραματική πιθανότητας σφάλματος συμβόλου

Παρατηρείται, όπως βλέπουμε, απόκλιση μεταξύ θεωρητικών και πειραματικών κυματομορφών. Κανονικά θα περιμέναμε οι δύο κυματομορφές να συμπίπτουν γεγονός που δεν συμβαίνει με το σφάλμα που προκάλεσε το λάθος να μην είναι δυνατό να εντοπιστεί. Αξίζει να σημειωθεί όμως, ότι οι κυματομορφές είχαν αναμενόμενη συμπεριφορά καθώς παρατηρήθηκε μείωση των τιμών όσο αυξάνεται το SNR , όπως γνωρίζουμε και από τις ιδιότητες της κανονικοποιημένης τυχαίας μεταβλητής.

Β΄.3 Σχεδιασμός θεωρητικής και πειραματικής πιθανότητας σφάλματος bit

Τέλος, υπολογίστηκε η θεωρητική και πειραματική πιθανότητα σφάλματος bit συναρτήσει του SNR σε κοινό semilogy. Το διάγραμμα της εικόνας 33 είναι το διάγραμμα που προέκυψε.



Σχήμα 33: Θεωρητική και πειραματική πιθανότητας σφάλματος bit

Παρατηρούμε ότι στο συγκεκριμένο ερώτημα η προσέγγιση μας ήταν καλύτερη με τις κυματομορφές σχεδόν να συμπίπτουν. Για την συμπεριφορά των κυματομορφών ισχύουν τα ίδια που ίσχυσαν και στο προήγουμενο ερώτημα.