



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Μάθημα: Ψηφιακές Επικοινωνίες Ι

Ονοματεπώνυμο: Ειρήνη Δόντη

Α.Μ: 03119839

4^η Σειρά Ασκήσεων

Αθήνα 2022

Περιεχόμενα

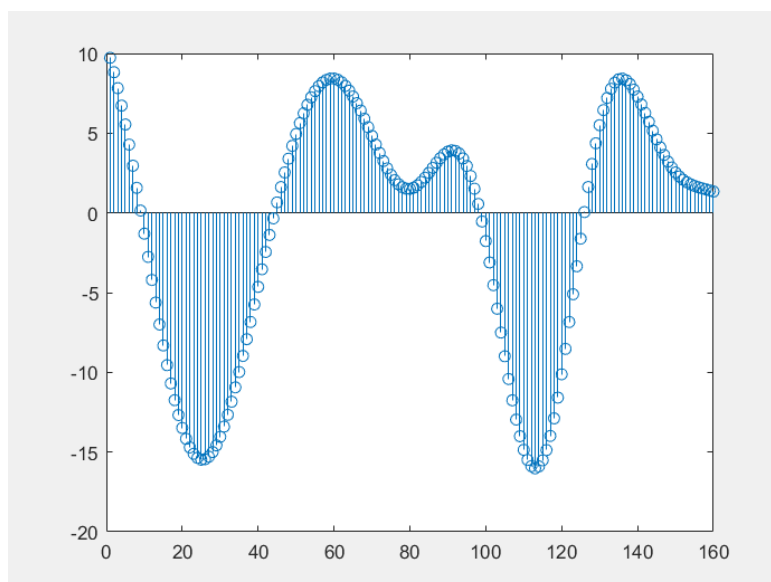
1. Παραγωγή σήματος με Φίλτρα Nyquist – Διαγράμματα χρόνου και συχνότητας	1
2. Υπολογισμός επίδοσης BER vs Eb/No. Μελέτη της επίδρασης των παραμέτρων: τάξη φίλτρου Nyquist και roll-off.....	4
3. Επίδραση του τρόπου κωδικοποίησης: Gray ή άλλη.....	7
4. Υπολογισμός παραμέτρων συστήματος	9

1. Παραγωγή σήματος με Φίλτρα Nyquist – Διαγράμματα χρόνου και συχνότητας

Παράγουμε τυχαία δυαδική ακολουθία $N_{\text{symb}}=10000$ bits και στη συνέχεια αντίστοιχο σήμα 16-ASK βασικής ζώνης με κωδικοποίηση Gray, σηματοδότηση root raised cosine με roll-off=0.35 και τάξη φίλτρου πομπού 128 (8 περιόδων, $\text{group_delay}=4T$).

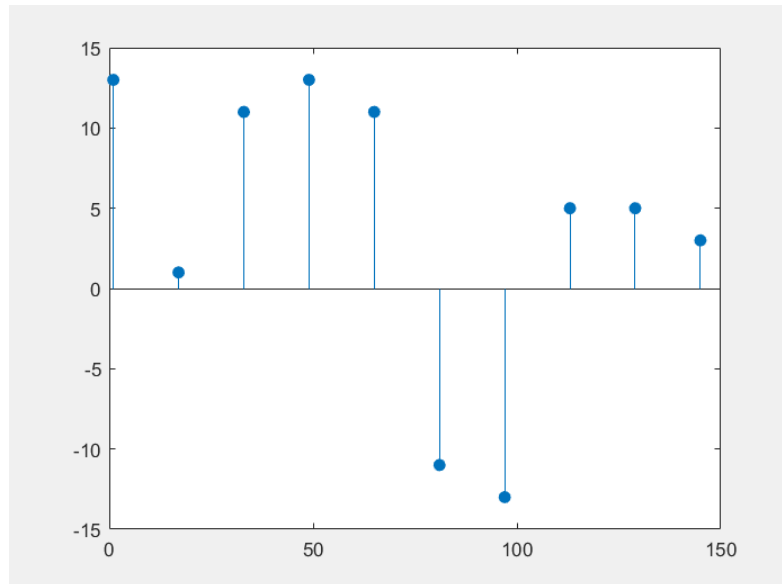
a.

Υπολογίζουμε το σήμα στην έξοδο προσαρμοσμένου φίλτρου στο δέκτη. Δείχνουμε τμήμα του σήματος αυτού (με την εντολή plot) διάρκειας $10T$.



b.

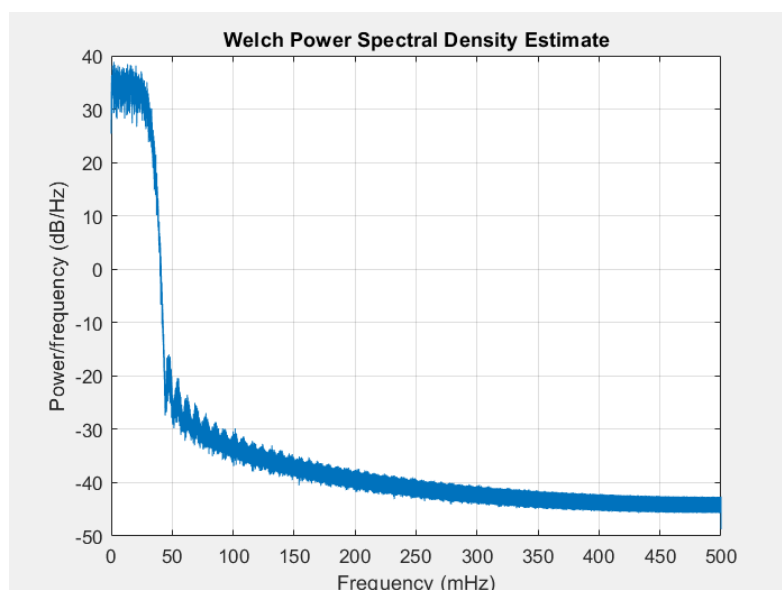
Με την εντολή `stem`, δημιουργούμε στο τμήμα αυτό αντίστοιχα δείγματα του σήματος εισόδου στο πλέγμα περιόδου T (πριν τη δειγματοληψία).



Σε περίπτωση απουσίας θορύβου, τα δείγματα αυτά συμπίπτουν με εκείνα της εξόδου του δέκτη.

c.

Σχεδιάζουμε (με την εντολή `pwelch`) το φάσμα του σήματος δέκτη, όπως απεικονίζεται παρακάτω:



Παρατηρούμε ότι το εύρος συχνοτήτων που καταλαμβάνει το φάσμα του σήματος στο δέκτη είναι ίσο με 0.5 Hz. Επίσης, καθώς αυξάνεται η συχνότητα, η τιμή power/frequency μειώνεται ραγδαία και συνεπώς η μορφή του παραπάνω διαγράμματος να είναι μία φθίνουσα καμπύλη. Η μορφή της γραφικής παράστασης, είναι η μορφή ενός βαθυπερατού φίλτρου. Το παραπάνω είναι λογικό, καθώς το φίλτρο Nyquist είναι, εν γένει, ένα βαθυπερατό φίλτρο.

Παρακάτω, παρατίθεται ο κώδικας που εκτελέστηκε για αυτό το ερώτημα:

```
function errors=ask_Nyq_filter(k,Nsymb,nsamp,EbNo)
% Εξομοιώνει την παραγωγή και αποκωδικοποίηση
% θορυβώδους ακολουθίας L-ASK και μετρά τα λαθμεμένα σύμβολα,
% με μορφοποίηση παλμών μέσω φίλτρου τετρ. ρίζας Nyquist.
% % % % % ΠΑΡΑΜΕΤΡΟΙ % % % % %
% k είναι ο αριθμός των bits ανά σύμβολο, έτσι L=2^k
% Nsymb είναι το μήκος της εξομοιούμενης ακολουθίας συμβόλων L-ASK
% nsamp είναι ο συντελεστής υπερδειγμάτωσης, δηλ. #samples/Td
% EbNo είναι ο ανηγμένος σηματοθορυβικός λόγος, Eb/No, σε db
clear all; % 16-QASK
k = 4; L=2^k; nsamp = 16; EbNo = 10; Nsymb = 10000; step=2;
SNR=EbNo-10*log10(nsamp/2/k); % SNR ανά δείγμα σήματος
% Διάνυσμα τυχαίων ακεραίων από το σύνολο {±1, ±3, ... ±(L-1)}
% x=2*floor(L*rand(1,Nsymb))-L+1;
% Ορισμός παραμέτρων φίλτρου
delay = 4; % Group delay (# of input symbols)
filtorder = delay*nsamp*2; % τάξη φίλτρου
rolloff = 0.35; % Συντελεστής πτώσης -- rolloff factor
% κρουστική απόκριση φίλτρου τετρ. ρίζας ανυψ. συνημιτόνου
rNyquist=rcosine(1,nsamp,'fir/sqrt',rolloff,delay); % κρουστική απόκριση φίλτρου
%figure; freqz(rNyquist);
%-----
% Για φίλτρο γραμμικής πτώσης να χρησιμοποιηθεί η επόμενη εντολή
% (με την rtrapezium του Κώδικα 4.1 στο current directory)
% Πρέπει delay>5 για καλά αποτελέσματα στην αναγνώριση.
% rNyquist=rtrapezium(nsamp,rolloff,delay);
% -----
% Gray Mapping
%step=4;
mapping=[step/2; -step/2]; % mapping is vector of ask points in gray-coding order
% Για L=8 και step=2: mapping=[7 5 1 3 -7 -5 -1 -3]

if(k>1)
    for j=2:k
        mapping=[mapping+2^(j-1)*step/2; -mapping-2^(j-1)*step/2];
    end
end
% %
% Without Gray coding
%mapping=-(L-1):step:L-1;
% Demo κωδικοποίησης Gray
%ask_scatterplot_gray(k,step);
% %
x=randi(2,1,k*Nsymb)-1;
xsym=bi2de(reshape(x,k,length(x)/k).','left-msb');
y1=[]; % ακολουθία παλμών στο φίλτρο πομπού
for i=1:length(xsym);
    y1=[y1 mapping(xsym(i)+1)];
end

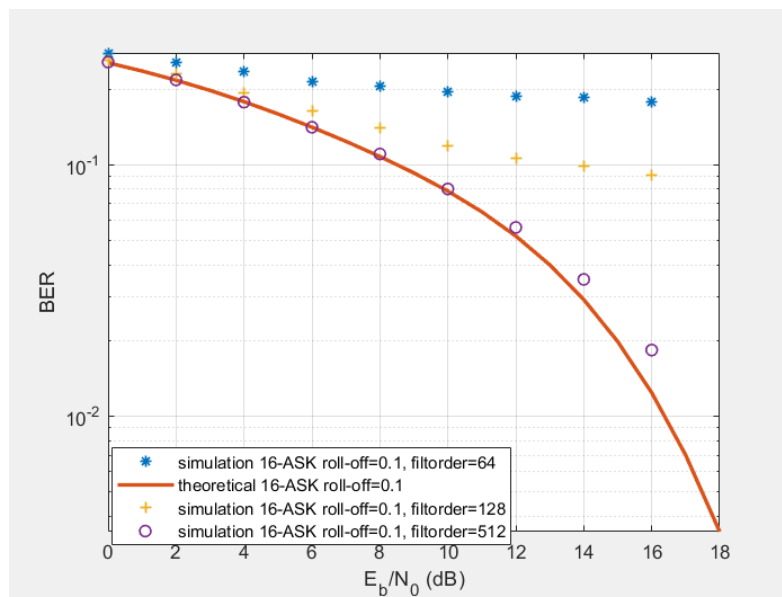
% % ΕΚΠΕΜΠΟΜΕΝΟ ΣΗΜΑ
% Υπερδειγμάτωση και εφαρμογή φίλτρου rNyquist
y=upsample(y1,nsamp); %x <-
ytx = conv(y,rNyquist,'same'); % εκπεμπόμενο σήμα - μετά την υπέρθεση - παραγόμενο σήμα - στην έξοδο πομπού
% Διάγραμμα οφθαλμού για μέρος του φιλτραρισμένου σήματος
% eyediagram(ytx(1:2000),nsamp*2);
ynoisy=awgn(ytx,SNR,'measured'); % θορυβώδες σήμα
figure; stem(ytx(1:20*nsamp))
% -----
figure; stem(ynois(1:20*nsamp))
% % ΛΑΘΑΝΟΜΕΝΟ ΣΗΜΑ
% Φιλτράρισμα σήματος με φίλτρο τετρ. ρίζας ανυψ. συνημ.
ytx=conv(ynois,rNyquist);
yr = downsample(ytx,nsamp); % Υποδειγμάτωση - δείγματα κατά τις στιγμές kT
ytx = ytx(2*delay*nsamp+1:end-2*delay*nsamp); % περικοπή, λόγω καθυστέρησης
% Ανιχνευτής ελάχιστης απόστασης L πλατών
l=[-L+1:2:L-1];
%xr=[]; % γειτονικές τιμές του yr στον αστερισμό
for i=1:length(yr)
    [m,j]=min(abs(mapping-yr(i)));
    %xr=[xr de2bi(j-1,k,'left-msb')];
end
%errors=sum(not(x==xr));
% %
%a.
figure; stem(ytx(1:10*nsamp));
%b.
figure; stem([1:nsamp:nsamp*10],y1(1:10), 'filled');
%c.
figure; pwelch(ytx,[],[],[],1);
end
```

2. Υπολογισμός επίδοσης BER vs E_b/N_0 .

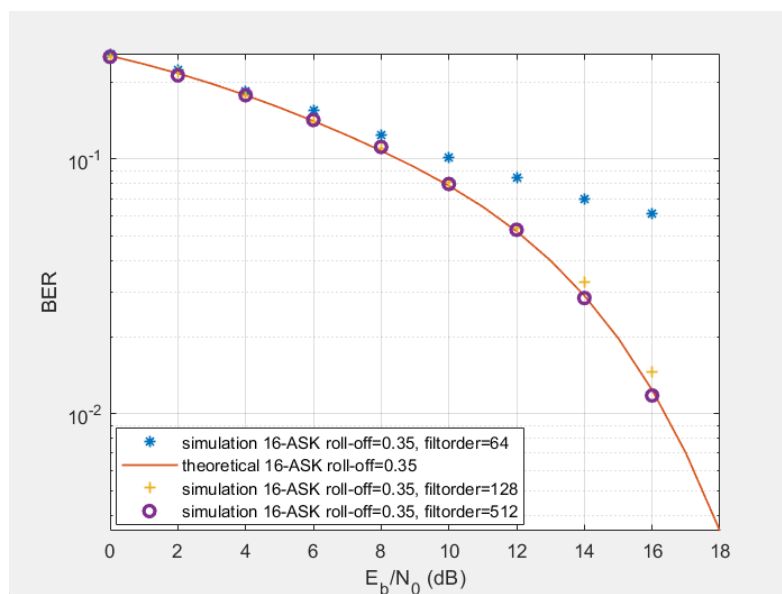
Μελέτη της επίδρασης των παραμέτρων: τάξη φίλτρου Nyquist και roll-off.

Για 16-ASK, θα ληφθεί η καμπύλη BER- E_b/N_0 θεωρητικά και με εξομοίωση:

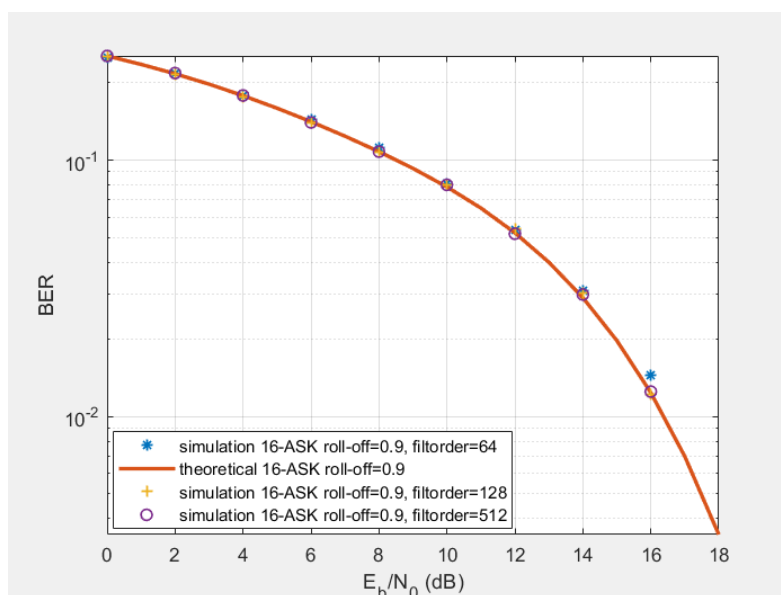
(I) με roll-off=0.1 και τάξη φίλτρου 64 (4 περιόδων, group_delay=2T), 128 (8 περιόδων, group_delay=4T) και 512 (32 περιόδων, group_delay=16T):



(II) με roll-off=0.35 και τάξη φίλτρου όπως πριν:



(III) με roll-off=0.9 και τάξη φίλτρου όπως παραπάνω:



Παρατηρούμε ότι, όσο αυξάνεται η τιμή της τάξης φίλτρου `filtorder` (για δεδομένη τιμή του `roll-off`), τόσο περισσότερα τα σημεία που προκύπτουν από την προσομοίωση τείνουν να ταυτιστούν με τα σημεία της θεωρητικής γραφικής παράστασης. Επίσης, όσο αυξάνεται η τιμή της `roll-off` (για σταθερή τιμή της τάξης φίλτρου `filtorder`), τόσο περισσότερα πλησιάζουν τα σημεία της προσομοίωσης με εκείνα της θεωρητικής γραφικής παράστασης.

Παρακάτω, παρατίθεται ο κώδικας ο οποίος δημιουργεί τη συνάρτηση `ask_Nyq_filter1()`:

```
function errors=ask_Nyq_filter1(k,Nsymb,nsamp,EbNo)
% Εξομοιώνει την παραγωγή και αποκωδικοποίηση
% θορυβώδους ακολουθίας L-ASK και μετρά τα λαθεμένα σύμβολα,
% με μορφοποίηση παλμών μέσω φίλτρου τετρ. ρίζας Nyquist.
% PARAMETROI %%%%%%%%%
% k είναι ο αριθμός των bits ανά σύμβολο, έτσι L=2^k
% Nsymb είναι το μήκος της εξομοιούμενης ακολουθίας συμβόλων L-ASK
% nsamp είναι ο συντελεστής υπερδειγμάτισης, δηλ. #samples/Td
% EbNo είναι ο ανηγμένος σηματοθορυβικός λόγος, Eb/No, σε db
%clear all; %16-ASK
%k = 4; L=2^k; nsamp = 16; step=2; Nsymb = 10000; EbNo = 10;
step=2; L=2^k;
SNR=EbNo-10*log10(nsamp/2/k); % SNR ανά δείγμα σήματος
% Ορισμός παραμέτρων φίλτρου
delay = 16; % Group delay (# of input symbols)
filtorder = delay*nsamp*2; % τάξη φίλτρου
rolloff = 0.9; % Συντελεστής πτώσης -- rolloff factor
% κρουστική απόκριση φίλτρου τετρ. ρίζας ανυψ. συνημιτόνου
rNyquist=rcosine(1,nsamp,'fir/sqrt',rolloff,delay);% κρουστική απόκριση φίλτρου
```

```

%figure; freqz(rNyquist);
%-----
% Για φίλτρο γραμμικής πτώσης να χρησιμοποιηθεί η επόμενη εντολή
% (με την rtrapezium του Κώδικα 4.1 στο current directory)
% Πρέπει delay>5 για καλά αποτελέσματα στην αναγνώριση.
% rNyquist=rtrapezium(nsamp,rolloff,delay);
% -----
% Gray Mapping
step=4;
mapping=[step/2; -step/2]; % mapping is vector of ask points in gray-coding order
% Για L=8 και step=2: mapping=[7 5 1 3 -7 -5 -1 -3]

if(k>1)
    for j=2:k
        mapping=[mapping+2^(j-1)*step/2; -mapping-2^(j-1)*step/2];
    end
end

% Without Gray coding
%mapping=-(L-1):step:L-1;
% Demo κωδικοποίησης Gray
%ask_scatterplot_gray(k,step);
%
x=randi(2,1,k*Nsymb)-1;
xsym=bi2de(reshape(x,k,length(x)/k).','left-msb');
y1=[]; % ακολουθία παλμών στο φίλτρο πομπού
for i=1:length(xsym);
    y1=[y1 mapping(xsym(i)+1)];
end

% ΕΚΠΕΜΠΟΜΕΝΟ ΣΗΜΑ
% Υπερδειγμάτωση και εφαρμογή φίλτρου rNyquist
y=upsample(y1,nsamp);
ytx = conv(y,rNyquist,'same'); % εκπεμπόμενο σήμα - μετά την υπέρθεση - παραγόμενο σήμα - στην έξοδο πομπού
% Διάγραμμα οφθαλμού για μέρος του φιλτραρισμένου σήματος
% eyediagram(ytx(1:2000),nsamp*2);
ynoisy=awgn(ytx,SNR,'measured'); % θορυβώδες σήμα
%figure; stem(ytx(1:20*nsamp))
% -----
%figure; stem(ynoisy(1:20*nsamp))
% ΛΑΜΒΑΝΟΜΕΝΟ ΣΗΜΑ
% Φιλτράρισμα σήματος με φίλτρο τετρ. ρίζας ανυψ. συνημ.
yrx=conv(ynoisy,rNyquist,'same');
yr = downsample(yrx,nsamp); % Υποδειγμάτωση - δείγματα κατά τις στιγμές κΤ
%yrx = yrx(2*delay*nsamp+1:end-2*delay*nsamp); % περικοπή, λόγω καθυστέρησης
% Ανιχνευτής ελάχιστης απόστασης L πλατών
l=[-L+1:2:L-1];
xr=[]; % γειτονικές τιμές του yr στον αστερισμό
for i=1:length(yr)
    [m,j]=min(abs(mapping-yr(i)));
    xr=[xr de2bi(j-1,k,'left-msb')];
end
errors=sum(not(x==xr));
end

```

Επίσης, παρακάτω απεικονίζεται και η συνάρτηση ask_ber_func() η οποία καλείται για την προσομοίωση BERTOOL:

```

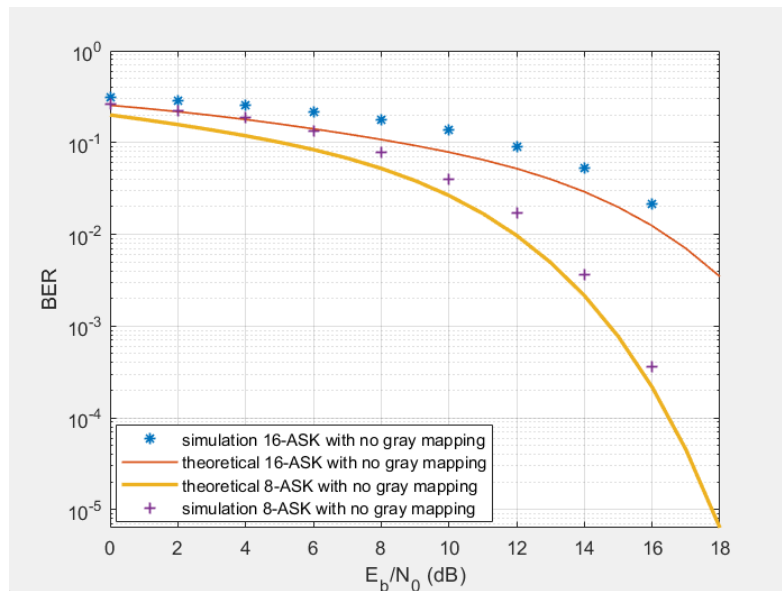
function [ber,numBits] = ask_ber_func(EbNo, maxNumErrs, maxNumBits,varargin)
% Import Java class for BERTool.
import com.mathworks.toolbox.comm.BERTool.*;
% Initialize variables related to exit criteria.
totErr = 0; % Number of errors observed
numBits = 0; % Number of bits processed
% A. --- Set up parameters. ---
% --- INSERT YOUR CODE HERE.
k=3; % number of bits per symbol
Nsymb=10000; % number of symbols in each run
nsamp=16; % oversampling,i.e. number of samples per T
% Simulate until number of errors exceeds maxNumErrs
% or number of bits processed exceeds maxNumBits.
while((totErr < maxNumErrs) && (numBits < maxNumBits))
    % Check if the user clicked the Stop button of BERTool.
    if (isBERToolSimulationStopped(varargin{:}))%BERTool.getSimulationStop
        break;
    end
    % B. --- INSERT YOUR CODE HERE.
    %errors=ask_errors(k,Nsymb,nsamp,EbNo);
    errors=ask_Nyq_filter1(k,Nsymb,nsamp,EbNo);
    % Assume Gray coding: 1 symbol error ==> 1 bit error
    totErr=totErr+errors;
    numBits=numBits + k*Nsymb;
end % End of loop
% Compute the BER

```


3.Επίδραση του τρόπου κωδικοποίησης: Gray ή άλλη

Παράγουμε τις καμπύλες BER-Eb/No για τις 16-ASK και 8-ASK με κωδικοποίηση mapping=-(L-1):step:(L-1), μέσω της προσομοίωσης BERTOOL.

Παρακάτω, απεικονίζονται τα ζητούμενα διαγράμματα:



Παρατηρούμε ότι, τα διαγράμματα που προκύπτουν από τις προσομοιώσεις έχουν απόκλιση από τις θεωρητικές γραφικές παραστάσεις BER-Eb/No των εκάστοτε L-ASK.

Επίσης, είναι λογικό να υπάρχουν περισσότερες αποκλίσεις στα διαγράμματα σε σχέση με εκείνα της κωδικοποίησης Gray. Η κωδικοποίηση Gray δημιουργεί γειτονικά στοιχεία με διαφορά μόνο κατά ένα bit. Όσον αφορά την κωδικοποίηση με άλλον τρόπο εκτός από Gray, τα γειτονικά σημεία μπορεί να μην κατανέμονται με διαφορά ψηφίου κατά ένα bit και συνεπώς να υπάρχει μεγαλύτερη απόκλιση της θεωρητικής από την πειραματική προσομοίωση.

Παρακάτω, παρατίθεται ο κώδικας ο οποίος δημιουργεί τη συνάρτηση

ask_Nyq_filter1():

```
function errors=ask_Nyq_filter1(k,Nsymb,nsamp,EbNo)
% Εξομοιώνει την παραγωγή και αποκωδικοποίηση
% θορυβώδους ακολουθίας L-ASK και μετρά τα λαθεμένα σύμβολα,
% με μορφοποίηση παλμών μέσω φίλτρου τετρ. ρίζας Nyquist.
% % % % % ΠΑΡΑΜΕΤΡΟΙ % % % % %
% k είναι ο αριθμός των bits ανά σύμβολο, έτσι L=2^k
% Nsymb είναι το μήκος της εξομοιούμενης ακολουθίας συμβόλων L-ASK
% nsamp είναι ο συντελεστής υπερδειγμάτωσης, δηλ. #samples/Td
% EbNo είναι ο ανηγμένος σηματοθορυβικός λόγος, Eb/No, σε db
% clear all; %16-ASK
% k = 4; L=2^k; nsamp = 16; step=2; Nsymb = 10000; EbNo = 10;
% step=2; L=2^k;
SNR=EbNo-10*log10(nsamp/2/k); % SNR ανά δείγμα σήματος
% Ορισμός παραμέτρων φίλτρου
delay = 16; % Group delay (# of input symbols)
filtorder = delay*nsamp*2; % τάξη φίλτρου
rolloff = 0.9; % Συντελεστής πτώσης -- rolloff factor
% κρουστική απόκριση φίλτρου τετρ. ρίζας ανυψ. συνημιτόνου
rNyquist=rcosine(1,nsamp,'fir/sqrt',rolloff,delay);% κρουστική απόκριση φίλτρου
%figure; freqz(rNyquist);
%-----
% Για φίλτρο γραμμικής πτώσης να χρησιμοποιηθεί η επόμενη εντολή
% (με την rtrapezium του Κώδικα 4.1 στο current directory)
% Πρέπει delay>5 για καλά αποτελέσματα στην αναγνώριση.
% rNyquist=rtrapezium(nsamp,rolloff,delay);
% -----
% Gray Mapping
%step=4;
% mapping=[step/2; -step/2]; % mapping is vector of ask points in gray-coding order
% % % % % Για L=8 και step=2: mapping=[7 5 1 3 -7 -5 -1 -3]
% if(k>1)
%     for j=2:k
%         mapping=[mapping+2^(j-1)*step/2; -mapping-2^(j-1)*step/2];
%     end
% end
% Without Gray coding
mapping=-(L-1):step:L-1;
% Demo κωδικοποίησης Gray
%ask_scatterplot_gray(k,step);
%
x=randi(2,1,k*Nsymb)-1;
xsym=bi2de(reshape(x,k,length(x)/k).','left-msb');
y1=[]; % ακολουθία παλμών στο φίλτρο πομπού
for i=1:length(xsym);
    y1=[y1 mapping(xsym(i)+1)];
end

% ΕΚΠΕΜΠΟΜΕΝΟ ΣΗΜΑ
% Υπερδειγμάτωση και εφαρμογή φίλτρου rNyquist
y=upsample(y1,nsamp);
ytx = conv(y,rNyquist,'same'); % εκπεμπόμενο σήμα - μετά την υπέρθεση - παραγόμενο σήμα - στην έξοδο πομπού
% Διάγραμμα οφθαλμού για μέρος του φιλτραρισμένου σήματος
% eyediagram(ytx(1:2000),nsamp*2);
ynoisys=awgn(ytx,SNR,'measured'); % θορυβώδες σήμα
%figure; stem(ytx(1:20*nsamp))
%-----
%figure; stem(ynoisys(1:20*nsamp))
% ΛΑΜΒΑΝΟΜΕΝΟ ΣΗΜΑ
% Φιλτράρισμα σήματος με φίλτρο τετρ. ρίζας ανυψ. συνημ.
yrx=conv(ynoisys,rNyquist,'same');
yr = downsample(yrx,nsamp); % Υποδειγμάτωση - δείγματα κατά τις στιγμές kT
%yrx = yrx(2*delay*nsamp+1:end-2*delay*nsamp); % περικοπή, λόγω καθυστέρησης
% Ανιχνευτής ελάχιστης απόστασης L πλατών
l=[-L+1:2:L-1];
xr=[];% γειτονικές τιμές του yr στον αστερισμό
for i=1:length(yr)
    [m,j]=min(abs(mapping-yr(i)));
    xr=[xr de2bi(j-1,k,'left-msb')];
end
errors=sum(not(x==xr));
end
```

Κάνουμε χρήση της ίδιας συνάρτησης ask_ber_func(), η οποία είναι απαραίτητη για την προσομοίωση μέσω bertool.

4. Υπολογισμός παραμέτρων συστήματος

Προσαρμόζουμε τις παραμέτρους του συστήματος μετάδοσης 16-ASK στα εξής πραγματικά δεδομένα:

Εύρος ζώνης διαύλου $W=1\text{MHz}$, Πυκνότητα φάσματος θορύβου $N_0=100$ picowatt/Hz, Ρυθμός μετάδοσης 6 Mbps και ανεκτό BER=2 Kbps.

Επαληθεύουμε τις προδιαγραφές του εύρους ζώνης και BER:

Το απαιτούμενο εύρος βασικής ζώνης με σηματοδοσία Nyquist είναι ίσο με:

$W = \frac{1}{2T} (1 + \alpha) (*)$, όπου α ο συντελεστής εξάπλωσης (roll-off) του φίλτρου Nyquist

και $\frac{1}{T}$ ο ρυθμός μετάδοσης συμβόλων (Baud Rate). Ο ρυθμός μετάδοσης R (bits/s)

συνδέεται με το $1/T$ και το μέγεθος αστερισμού L με τη σχέση $\frac{R}{\log_2(L)} = \frac{1}{T} (**)$

Για χρήση οριακής απόκρισης Nyquist, χρειαζόμαστε εύρος ζώνης $\frac{W}{2} = 500 \text{ KHz}$

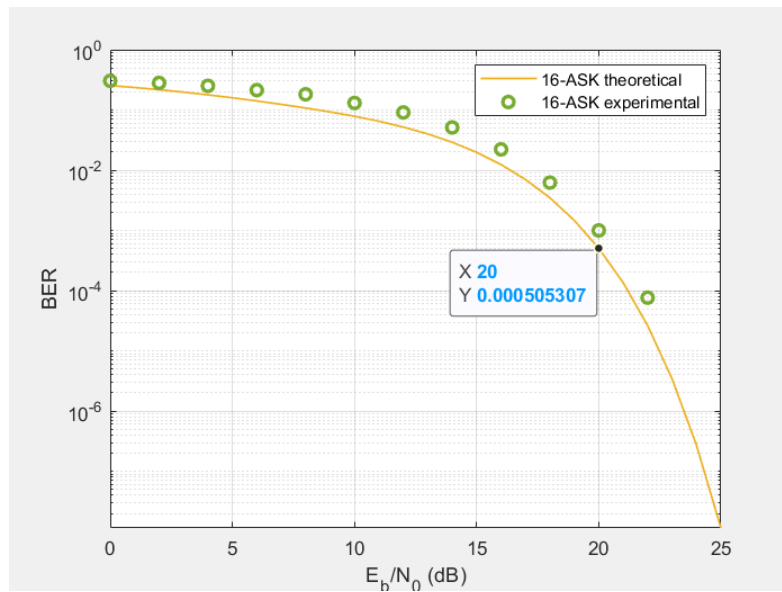
(για $\alpha=0$). Λύνοντας την εξίσωση (*) ως προς $\frac{1}{T}$ τότε, $\frac{1}{T} = \frac{2W}{(1+\alpha)}$.

Για $\alpha=0$, τότε $\frac{1}{T} = \frac{2 \cdot 1 \text{ MHz}}{1} = 2 \text{ MHz}$. Επίσης, για $k=3$, ο ρυθμός μετάδοσης από τη σχέση (**) είναι $R = \frac{1}{T} \log_2(L) = 6 \text{ Mbps}$. Δηλαδή, επαληθεύεται η δοσμένη τιμή του ρυθμού μετάδοσης. Ωστόσο, αυτή η περίπτωση δεν είναι ευσταθής, καθώς είναι οριακή περίπτωση. Οπότε, οι τιμές του λόγου $\frac{1}{T}$ πρέπει να είναι $\frac{1}{T} < 2 \text{ MHz}$, και συνεπώς $k > 3$. Οπότε, η ελάχιστη τιμή που μπορεί να πάρει η τιμή k που αντιπροσωπεύει τον αριθμό bits ανά σύμβολο είναι $k=4$. Οπότε, το σύστημα μετάδοσης που πρέπει να χρησιμοποιηθεί είναι το 16-ASK, οπότε επαληθεύεται η δοθείσα παράμετρος.

Η μορφή των παλμών δεν επηρεάζουν το BER, καθώς το BER δεν εξαρτάται από το α και συνεπώς από το εύρος ζώνης.

Από τη σχέση $W = \frac{1}{2T}(1 + \alpha)$ και τη σχέση $\frac{R}{\log_2(L)} = \frac{1}{T}$, συμπεραίνουμε ότι $W = \frac{R}{2 \log_2(L)}(1 + \alpha)$ ή $\alpha = \frac{2 \log_2(L)}{R} W - 1 = \frac{2 \log_2(16)}{6 \cdot 10^6} 1 \cdot 10^6 - 1 = \frac{1}{3} = 0.3333$

Πρέπει να βρούμε ποιος είναι ο σηματοθορυβικός λόγος EbNo, ώστε το BER να είναι ίσο με 2 kbps ή $\text{BER}_{\text{bertool}} = \frac{\text{BER}}{R} = \frac{2 \cdot 10^3}{6 \cdot 10^6} = 0.33 \cdot 10^{-3}$ μέσω της προσομοίωσης BERTOOL. Εκτελούμε την προσομοίωση και συνεπώς λαμβάνουμε τις παρακάτω θεωρητικές και πειραματικές γραφικές για 16-ASK:



Παρατηρούμε, από το παραπάνω διάγραμμα, ότι ο σηματοθορυβικός λόγος, ο οποίος ικανοποιεί τις παραπάνω συνθήκες, είναι περίπου 20 dB.

Ο κώδικας που χρησιμοποιήθηκε για το παραπάνω υποερώτημα, είναι ο παρακάτω:

```
% Eirini Donti 03119839
function errors=ask_Nyq_filter1(k,Nsymb,nsamp,EbNo)
% Εξομοιώνει την παραγωγή και αποκωδικοποίηση
% Θορυβώδους ακολουθίας L-ASK και μετρά τα λαθεμένα σύμβολα,
% με μορφοποίηση παλμών μέσω φίλτρου τετρ. ρίζας Nyquist.
% % % % % ΠΑΡΑΜΕΤΡΟΙ % % % % %
% k είναι ο αριθμός των bits ανά σύμβολο, έτσι L=2^k
% Nsymb είναι το μήκος της εξομοιούμενης ακολουθίας συμβόλων L-ASK
% nsamp είναι ο συντελεστής υπερδειγμάτωσης, δηλ. #samples/Td
% EbNo είναι ο ανηγμένος σηματοθορυβικός λόγος, Eb/No, σε db
% clear all; %16-ASK
% k = 4; L=2^k; nsamp = 16; step=2; Nsymb = 10000; EbNo = 10;
% step=2; L=2^k;
% SNR=EbNo-10*log10(nsamp/2/k); % SNR ανά δείγμα σήματος
% Ορισμός παραμέτρων φίλτρου
% delay = 8; % Group delay (# of input symbols)
% filtorder = delay*nsamp*2; % τάξη φίλτρου
% rolloff = 1/3; % Συντελεστής πτώσης -- rolloff factor
% κρουστική απόκριση φίλτρου τετρ. ρίζας ανυψ. συνημιτόνου
% rNyquist=rcosine(1,nsamp,'fir/sqrt',rolloff,delay);% κρουστική απόκριση φίλτρου
% figure; freqz(rNyquist);
% -----
% Για φίλτρο γραμμικής πτώσης να χρησιμοποιηθεί η επόμενη εντολή
% (με την rtrapezium του Κώδικα 4.1 στο current directory)
% Πρέπει delay>5 για καλά αποτελέσματα στην αναγνώριση.
% rNyquist=rtrapezium(nsamp,rolloff,delay);
% -----
% Gray Mapping
% step=4;
% mapping=[step/2; -step/2]; % mapping is vector of ask points in gray-coding order
% % % % % Για L=8 και step=2: mapping=[7 5 1 3 -7 -5 -1 -3]
if(k>1)
    for j=2:k
        mapping=[mapping+2^(j-1)*step/2; -mapping-2^(j-1)*step/2];
    end
end

% Without Gray coding
% mapping=-(L-1):step:L-1;
% Demo κωδικοποίησης Gray
% ask_scatterplot_gray(k,step);
%
% x=randi(2,1,k*Nsymb)-1;
% xsym=bi2de(reshape(x,k,length(x)/k).','left-msb');
% y1=[]; % ακολουθία παλμών στο φίλτρο πομπού
% for i=1:length(xsym);
%     y1=[y1 mapping(xsym(i)+1)];
% end

% ΕΚΠΕΜΠΟΜΕΝΟ ΣΗΜΑ
% Υπερδειγμάτωση και εφαρμογή φίλτρου rNyquist
% y=upsample(y1,nsamp);
% ytx = conv(y,rNyquist,'same'); % εκπεμπόμενο σήμα - μετά την υπέρθεση - παραγόμενο σήμα - στην έξοδο πομπού
% Διάγραμμα οφθαλμού για μέρος του φιλτραρισμένου σήματος
% eyediagram(ytx(1:2000),nsamp*2);
% ynoisy=awgn(ytx,SNR,'measured'); % θορυβώδες σήμα
% figure; stem(ytx(1:20*nsamp))
% -----
% figure; stem(ynois(1:20*nsamp))
% ΛΑΜΒΑΝΟΜΕΝΟ ΣΗΜΑ
% Φιλτράρισμα σήματος με φίλτρο τετρ. ρίζας ανυψ. συνημ.
% yrx=conv(ynois,rNyquist,'same');
% yr = downsample(yrx,nsamp); % Υποδειγμάτωση - δείγματα κατά τις στιγμές kT
% yrx = yrx(2*delay*nsamp+1:end-2*delay*nsamp); % περικοπή, λόγω καθυστέρησης
% Ανιχνευτής ελάχιστης απόστασης L πλατών
% l=[-L+1:2:L-1];
% xr=[];% γειτονικές τιμές του yr στον αστερισμό
% for i=1:length(yr)
%     [m,j]=min(abs(mapping-yr(i)));
%     xr=[xr de2bi(j-1,k,'left-msb')];
% end
% errors=sum(not(x==xr));
end
```