

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ : Τ.Μ.Η/Υ.Π

ΕΠΩΝΥΜΟ : ΤΡΙΑΝΤΗΣ

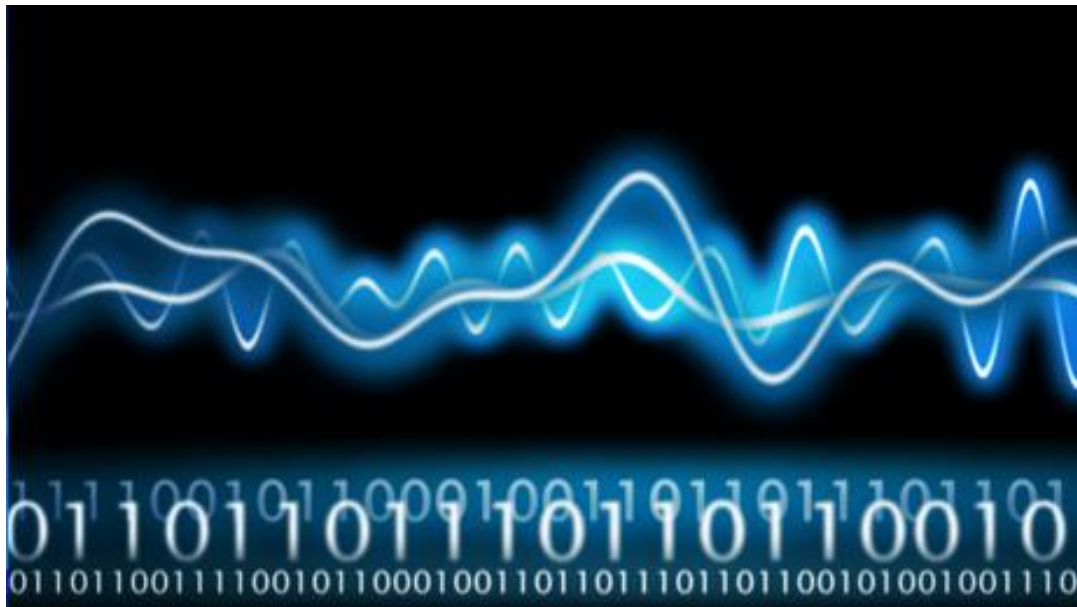
ΟΝΟΜΑ : ΠΑΝΑΓΙΩΤΗΣ

Α.Μ : 5442

ΨΗΦΙΑΚΕΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΕΣ

ΕΡΓΑΣΤΗΡΙΑΚΗ ΑΣΚΗΣΗ 2

ΣΥΓΚΡΙΣΗ ΟΜΟΔΥΝΩΝ ΣΥΣΤΗΜΑΤΩΝ (4-PSK,4-FSK)



Αρχικά παρουσιάζω όλους τους κώδικες που χρησιμοποιήθηκαν στα πλαίσια της άσκησης :

Συναρτήσεις που χρησιμοποιούνται για να δημιουργηθούν τα ομόδυνα συστήματα (4-PSK & 4-FSK).

mapper :

```
%% mapper function
% symb : symbola pou ana8etei o mapper sthn akoloy8ia mas pou
eisagoyme
% mhkos : mhkos ths akoloy8ias
% bin : dyadikh akoloy8ia
% type : kwidikopoihsh gia PSK or FSK (gia psk xrhsimopoioume Gray enw
gia
% fsk ginetai kanonika h antitoixish
function [symb,mhkos] = mapper(bin,type)

mhkos = length(bin);
rem = mod(mhkos,2);
new_len = (mhkos-rem)/2;
if rem == 0 % An o arxikos ari8mos einai zugos,
    x_new = reshape(bin,2,new_len); % anasxhmatizw to dianusma x se 2
grammes
    %kai new_len sthles.
else % An einai monos anasxhmatizw pali to dianusma x ekτος apo to
teleutaio stoixeio tou.
    x_new = reshape(bin(1:mhkos-1,1),2,new_len);
end
% Antistoixish ka8e dyadas bits se sumvola.
for i = 1:new_len
    if x_new(1,i)==0 && x_new(2,i)==0
        symb(i) = 0;
    elseif x_new(1,i)==0 && x_new(2,i)==1
        symb(i) = 1;
    elseif x_new(1,i)==1 && x_new(2,i)==0
        if type=='psk'
            symb(i) = 3; % gray
        elseif type=='fsk'
            symb(i) = 2;
        end
    elseif x_new(1,i)==1 && x_new(2,i)==1
        if type=='psk'
            symb(i) = 2; % gray
        elseif type=='fsk'
            symb(i) = 3;
        end
    end
end
% mia akolou8ia pou exei mono ari8mo bit sto teleutaio p einai mono
toy
% ana8etoume mia tyxaia timh dyadas
if rem==1
symb(new_len+1)=randsrc(1,1,[0 3]);
end
end
```

modulator :

```
%% modulator function
% signal : to en8oryvo shma
% s_m : to shma prin prote8ei o 8oryvos
% symb : h e3odos symbolwn meta ton mapper
% SNR
% Es : energeia / symbol
% Eb : energeia / bit
% T_symb : T symbol
% fc : f ferousas
% type : PSK or FSK

function [signal,s_m] = modulator(symb,SNR,Es,Eb,T_symb,fc,type)

length_symb = length(symb);%To mhkos ths akolou8ias sumvolwn.
g_t = sqrt(2*Es/T_symb);%Orthogwnios palmos.
s_m = zeros(length_symb,40); %Arxikopoiw me mhden to shma pou
prokeitai na steilw.
s = zeros(4,2);%Arxikopoiw to dianusma pou periexei tis 2 sunistwses,
%gia ka8e sumvolo.
if type=='psk'
%Ka8e sumvolo ths diamorfwsis 4-PSK orizetai apo 2 sunistwses, tis
opoies
%vriskw ws ekshs:
    for m = 1:4
        s(m,1) = cos((2*pi*(m-1))/4);
        s(m,2) = sin((2*pi*(m-1))/4);
    end
%Gia na upologisw to zwnoperato shma
    for i = 1:length_symb
        for t = 1:40 %1 ews T_symb
            k_n = symb(i)+1;%Gia na paroume tis swstes sunistwses s
            %pros8etoume to 1 se ka8e sumvolo
            s_m(i,t) = s(k_n,1)*g_t*cos(2*pi*fc*t) +
s(k_n,2)*g_t*sin(2*pi*fc*t);
        end
    end
elseif type=='fsk'
    for i = 1:length_symb
        for t = 1:40 %1 ews T_symb
            s_m(i,t) = g_t*cos(2*pi*(fc + symb(i)/40)*t);
        end
    end
end
No = Eb/(10^(SNR/10));
noise = sqrt(No/2)*randn(length_symb,40); %Ypologismos tou AWGN
8oruvou me vash
%to SNR pou eisagoume kai kanonikopoihsh gia na ginei h pros8esh me
to shma
%mas
signal = s_m + noise; % en8oruvo shma
end
```

demodulator :

```
%% demodulator function
```

```

% r : dianysma meta thn apodiamorfws
% signal : to shma mas
% Es : energeia / symbol
% T_symb : T symbol
% fc : f ferousas
% type : PSK or FSK

function [r] = demodulator(signal,Es,T_symb,fc,type)

gt = sqrt(2*Es/T_symb);%Orthogwnios palmos.
%Apodiamorfws shmatos.
if type=='psk'
%Pollaplasiazw tis 2 sunistwses ths ferousas me ton orthogwnio palmo
kai
%ta apotelesmata ta ekxwrw sta dianusmata y1 kai y2.
    for t = 1:40
        y1(t,1) = gt*cos(2*pi*fc*t);
        y2(t,1) = gt*sin(2*pi*fc*t);
    end
%Susxetizw to shma pou elava me tis 2 sunistwses opote prokuptoun
%oi r1,r2.
    r1 = signal*y1;
    r2 = signal*y2;
%Vriskw to dianusma r einai h ektimhtheisa timh tou trexontos
%sumvolou panw ston asterismo tou 4-PSK.
    r = [r1,r2];
elseif type=='fsk'
%Gia to ka8e sumvolo (0 ews 3,) pollaplasiazw thn antistoixh
sunistwsa
%ths ferousas me ton orthogwnio palmo kai to apotelesma to ekxwrw sto
%dianusma y.
    for m = 0:3
        for t = 1:40
            y(t,m+1) = gt*cos(2*pi*(fc + m/40)*t);
        end
    end
%Vriskw to dianusma r pou einai h ektimhtheisa timh tou trexontos
%sumvolou panw ston asterismo tou 4-FSK.
    r = signal*y;
end
end

```

fwraths :

```

%% function fwrath
% symb_teliko : ta symbola pou apofasizei o fwraths oti exei stalei
% r : to dianysma pou einai e3odos tou demodulator
% type : PSK || FSK

function [symb_teliko] = fwraths(r,type)

len = length(r);
if type=='psk'
    for m = 1:4
        s(m,1) = cos((2*pi*(m-1))/4);
        s(m,2) = sin((2*pi*(m-1))/4);
    end
%Ypologizw thn apostash tou s apo to r. Auto (to dianusma) me thn
mikroterh

```

```

%apostash apo to r, antistoixei sto sumvolo pou stal8hke
    for i = 1:len
        min = 1.0e3; %Thetw enan polu megalo arithmo ws min
        position = 1; %Apo8hkeuei th thesh tou distance pou vre8hke h
mikroterh
%apostash k to arxikopoiw me 1
    for m = 1: 4
        distance(m, 1) = norm([r(i,1), r(i,2)] - s(m,:));
        if distance(m,1) < min
            min = distance(m,1);
            position = m-1; %Vazw m-1 giati ta sumvola pou exw einai
apo 0
%ews 3, enw oi 8eseis tou dianusmatos apo 1 ew 4
    end
    end
%H thesh tou distance pou periexei th mikroterh apostash,
antiprosopeuei
%kai to sumvolo pou stal8hke
    symb_teliko(i,1) = position;
    end
elseif type=='fsk'
    for i = 1:len
        max = -1.0e3; %Thetw enan polu mikro arithmo ws max
        position = 1; %Apo8hkeuei th thesh tou r, pou vre8hke h
megaluterh timh
%k to arxikopoiw me 1
        for j = 1:4
            if r(i,j) > max
                max = r(i,j);
                position = j-1;
            end
        end
        symb_teliko(i,1) = position;
    end
end
end
end

```

demapper :

```

%% function demmapper
% received : shma pou lhf8hke
% symb_teliko : ta symbola pou apofasise o fwraths
% mhkos : mhkos ths arxikhs akolou8ias bits
% type : PSK || FSK (gia ton psk xrhsimopoioume edw apokwdikopoioume
me
% bash to kwdika gray enw gia fsk apokwdikopoioume kanonika

function [received] = demapper(symb_teliko,mhkos,type)

len = length(symb_teliko); %To megethos tw n sumvolwn.
rem = mod(mhkos,2); %Vriskw an einai zugos h' monos ari8mos h arxikh
akolou8ia.
x_tel = zeros(2,len-rem); %Arxikopoihsh toy dianusmatos
for i=1:len
    if symb_teliko(i)==0
        x_tel(1,i)=0; x_tel(2,i)=0;
    elseif symb_teliko(i)==1

```

```

        x_tel(1,i)=0; x_tel(2,i)=1;
elseif symb_teliko(i)==2
    if type=='psk'
        x_tel(1,i)=1; x_tel(2,i)=1; %gray
    elseif type=='fsk'
        x_tel(1,i)=1; x_tel(2,i)=0;
    end
else
    if type=='psk'
        x_tel(1,i)=1; x_tel(2,i)=0; %gray
    elseif type=='fsk'
        x_tel(1,i)=1; x_tel(2,i)=1;
    end
end
end
%Anasxhmatizw pali to dianusma x_tel se 2*len grammes kai 1 sthlh,gia
na
%dhmiourghsw to received
received = reshape(x_tel,2*len,1);
if rem==1
    received(2*len+1) = randsrc(1,1,[0 1]);
    received=received(1:2*len+1);
end
end

```

Συναρτήσεις που χρησιμοποιούνται για να δημιουργηθούν τα ομόδυνα συστήματα (4-PSK & 4-FSK).

ypologismos ber (συνάρτηση που υπολογίζει το BER) :

```

%% function ypologismos_ber : ypologizei to BER
% BER : to bit error rate pou zhtame
% x : h arxikh ακολου8ia
% received : h lhf8eisa ακολου8ia
% SNR : to SNR
% type : theoritical(gia ypologismo toy 8ewrhtikou BER gia to PSK)...
% experimental(gia ypologismo tou BER peiramatika, dhladh posa la8os
bit
% lhf8hkan)
function [BER] = ypologismos_ber(x,mhkos,received,SNR,type)

arxikopoihsh
No = Eb/(10^(SNR/10));
EbNo = Eb/No;
if strcmp(type,'theoritical')==1
    BER = (1/2)*erfc(sqrt(EbNo));%Ypologizw theorhtika to BER
elseif strcmp(type,'experimental')==1
    error_bit_counter = 0;%bazoyme ena counter gia na metrhsoyme me
sugkrish ta la8os mpit
    for i=1:mhkos
        if x(i)~=received(i)%sygkrish stal8entos kai lhf8entos
akolou8ias
            error_bit_counter = error_bit_counter+1;
        end
    end
    BER = error_bit_counter/mhkos; %Ypologizw peiramatika to BER
(la8os bits)
end

```

end

spectrum (συνάρτηση που υπολογίζει το φάσμα ισχύος που ζητείται) :

```
%% function spectrum
% s : spectrum
% ypologismos fasmatos isxyos
% x : to arxiko mas binary
% Es : energeia / symbol
% Eb : energeia / bit
% T_symb : T symbol
% fc : f ferousas
% type : PSK || FSK

function [spec] = spectrum(x,Es,Eb,T_symb,fc,type)

if type=='psk'
    [symb,~]=mapper(x,'psk');
    [~,s_m]=modulator(symb,10,Es,Eb,T_symb,fc,'psk');
elseif type=='fsk'
    [symb,~]=mapper(x,'psk');
    [~,s_m]=modulator(symb,10,Es,Eb,T_symb,fc,'fsk');
end
length_s_m = length(s_m);
s_m_reshaped = reshape(s_m',1,length_s_m*40);
length_reshaped = length(s_m_reshaped);
r = 2048 - mod(length_s_m*40,2048);%Gia na vrw tis epipleon 8eseis
pou prepei
%na eisagw mhdenika sto dianusma s_m_reshaped.
for i = length_reshaped:length_reshaped+(r-1) %Sumplhrwnw me mhdenika
tis upoloipes 8eseis
    s_m_reshaped(i+1) = 0;%tou neou dianusmatos
end
%Arxikopoiw to prwto sector apo ta 2048 me apeu8eias timh
sector(1:2048,1) = abs(fft(s_m_reshaped(1:2048))).^2;
for i = 2:(length_reshaped/2048)-1%Ypologizw ta upoloipa 2047 tmhmata
kai ta vazw sto
    %dianusma sector

sector(1:2048,i)=abs(fft(s_m_reshaped((2048*i)+1:2048*(i+1)))).^2;
end
for i = 1:2048 %Gia ta 2048 tmhmata pou upologisame parapanw
vriskoume thn
    %mesh timh tous
    Athroisma = 0;
    for j = 1:length_reshaped/2048-1
        athroisma = sector(i,j)+athroisma;
    end
    spec(i,1) = athroisma/2048;
end
end
```

Ακολούθως παρατίθενται τα script που καλούν τις παραπάνω συναρτήσεις για να απαντηθούν τα ερωτήματα 3 & 4.

3^ο ερώτημα script :

```

%% script ylopoihshs gia to erwthma 3

close all; clear all; clc

% oi arxikopoihseis pou xreiazontai
T_symb=40;           %T symbol
T_samp=1;            %T sample
Tc=4;                %T ferousas
fc=1/Tc;             %f ferousas
Es=1;                %Energeia / symbol
Eb=Es/2;             %Energeia / bit

x = randsrc(1.0e5,1,[0 1]); % h binary akolou8ia mas
SNR = [0:2:8];

[psk_symb,mhkos1] = mapper(x,'psk');
[fsk_symb,mhkos2] = mapper(x,'fsk');

for i = 1:length(SNR)
    %% psk
    [psk_signal,~] =
modulator(psk_symb,SNR(i),Es,Eb,T_symb,fc,'psk');
    [psk_r] = demodulator(psk_signal,Es,T_symb,fc,'psk');
    [psk_symb_teliko] = fwraths(psk_r,'psk');
    [psk_received] = demapper(psk_symb_teliko,mhkos1,'psk');
    psk_BER_exp(i) =
ypologismos_ber(x,mhkos1,psk_received,SNR(i),'experimental');
    psk_BER_theor(i) =
ypologismos_ber(x,mhkos1,psk_received,SNR(i),'theoretical');

    %% fsk
    [fsk_signal,~] =
modulator(fsk_symb,SNR(i),Es,Eb,T_symb,fc,'fsk');
    [fsk_r] = demodulator(fsk_signal,Es,T_symb,fc,'fsk');
    [fsk_symb_teliko] = fwraths(fsk_r,'fsk');
    [fsk_received] = demapper(fsk_symb_teliko,mhkos2,'fsk');
    fsk_BER_exp(i) =
ypologismos_ber(x,mhkos2,fsk_received,SNR(i),'experimental');

end

semilogy(SNR',psk_BER_exp,'r:>',SNR',fsk_BER_exp,'m-
*',SNR',psk_BER_theor,'b-o');
legend('BER psk','BER fsk','BER psk theor');
xlabel('SNR(dB)');
ylabel('BER');
title('Erotima 3 - BER');

```

4^ο ερωτημα script:

```

%% script ylopoihshs gia to erwthma 4

close all; clear all; clc

% oi arxikopoihseis pou xreiazontai
T_symb=40;           %T symbol
T_samp=1;            %T sample

```



```

Tc=4; %T ferousas
fc=1/Tc; %f ferousas
Es=1; %Energeia / symbol
Eb=Es/2; %Energeia / bit

x = randsrc(1.0e5,1,[0 1]);

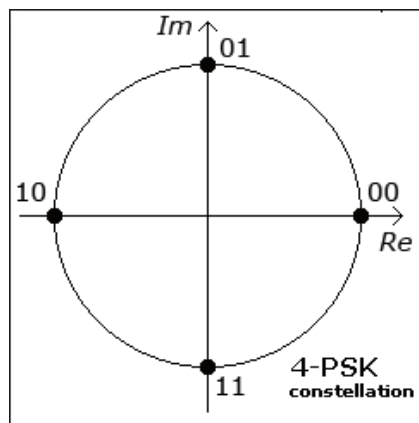
[psk_spec] = spectrum(x,Es,Eb,T_symb,fc,'psk');
[fsk_spec] = spectrum(x,Es,Eb,T_symb,fc,'fsk');

n = [1:1:2048]; %oi times pou 8elw na valw ston x'x a3ona
semilogy(n,psk_spec, 'r',n,fsk_spec,'b:');
title('Erotima 4 - Spectrum');
xlabel('samples');
ylabel('power spectrum');
legend('PSK Power Spectrum','FSK Power Spectrum');

```

Ερώτημα 1&2 (PSK & FSK)

Σχήμα διαμόρφωσης μετατόπισης φάσης, PSK, χρησιμοποιείται ευρέως αυτές τις μέρες μέσα σε μια ολόκληρη σειρά από συστήματα ραδιοεπικοινωνιών. Στην άσκηση μας υπάρχουν τέσσερις φάσεις ($m = 4$) - συγκεκριμένα η λειτουργία MPSK με $M=4$ ονομάζεται τεταρτοταγής μετατόπισης φάσης (QPSK) - ακόμη η ενέργεια όλων των καταστάσεων-φάσεων είναι η ίδια, και κάθε μετατόπιση φάσης αντιπροσωπεύει δύο στοιχεία του σήματος. Γεωμετρικά αυτό αναπαρίσταται με ένα κύκλο και οι φάσεις καταλαμβάνουν τις θέσεις $0^\circ, 90^\circ, -90^\circ, 180^\circ$ και με ακτίνα κύκλου $\sqrt{E_s}$.

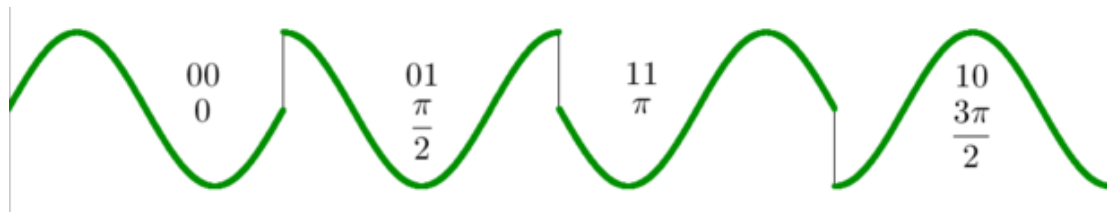


Στην άσκηση το σήμα μας θα έχει τη μορφή :

$$u_m(t) = g_T(t) * \cos\left(2\pi f_c t + \frac{2\pi m}{4}\right)$$

Όπου το g_T είναι ο ορθογώνιος παλμός που χρησιμοποιείται για τη μετάδοση των συμβόλων.

Ένα παράδειγμα QPSK :

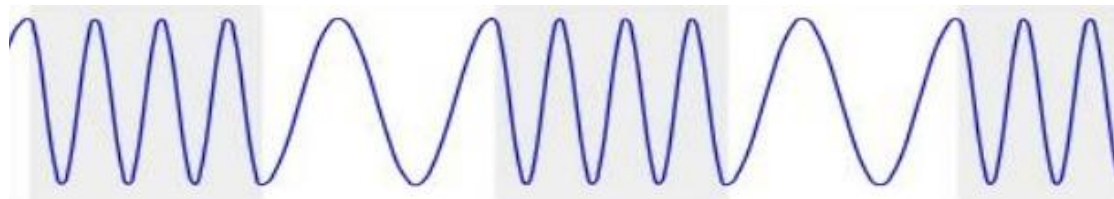


Σχήμα συχνοτικής διαμόρφωσης , FSK. Σε αυτό η πληροφορία μας (η δυαδική ακολουθία μας δηλαδή) διαδίδεται μέσω διακριτών αλλαγών στη συχνότητα του φέροντος σήματος. Η ενέργεια όλων των κυματομορφών και πάλι είναι ίση με 1 , ίδια σε όλες τις κυματομορφές και χρησιμοποιούμε πάλι τον ορθογώνιο παλμό g_T για τη μετάδοση του σήματος. Στο ζωνοπερατό σήμα μεταδίδονται $k = \log_2 4$ bits ανά κυματομορφή και οι κυματομορφές είναι της μορφής :

$$u_m(t) = g_T(t) * \cos(2\pi f_c + m\Delta f)t$$

όπου Δf : η διαφορά μεταξύ 2 διαδοχικών συχνοτήτων

Ένα παράδειγμα 4-FSK :



mapper

Στο σύστημα 4-psk ακόμη να προσθέσουμε ότι χρησιμοποιείται η κωδικοποίηση GRAY στον mapper με βάση το πίνακα που παρατίθεται για να αντιστοιχιστεί κάθε δυάδα bit σε ένα σύμβολο ενώ στο σύστημα 4-fsk στο mapper χρησιμοποιείται η κανονική κωδικοποίηση (αφού τα σύμβολα είναι γειτονικά μεταξύ τους, οπότε δεν έχει νόημα η κωδικοποίηση GRAY) όπως φαίνεται και παρακάτω στον πίνακα.

<u>PSK</u>			<u>FSK</u>	
Bin	Gray	Symbol	Bin	Symbol
00	00	0	00	0
01	01	1	01	1
10	11	2	10	2
11	10	3	11	3

modulator

4-PSK

Για τη διαμόρφωση 4-psk (αφού στην άσκηση μας έχουμε ίδια ενεργεια ,ίση με 1, για όλες τις κυματομορφές) το ζωνοπερατό σήμα που προκύπτει τελικά, έπειτα από τη διαμόρφωση, είναι το :

$$s_m(t) = \cos\left(\frac{2\pi m}{4}\right) g_T(t) \cos(2\pi f_c t) + \cos\left(\frac{2\pi m}{4}\right) g_T(t) \sin(2\pi f_c t)$$

4-FSK

Έπειτα από τη διαμόρφωση το ζωνοπερατό σήμα που προκύπτει για το 4-fsk είναι το :

$$s_m(t) = g_T(t) \cos\left(2\pi\left(f_c + \frac{m}{T_{symbol}}\right)t\right)$$

Noise

Στον modulator στο τέλος της συνάρτησης που υλοποιήθηκε προστίθεται και ο θόρυβος. Για το θόρυβο έχουμε ότι :

$$10\log_{10}\left(\frac{E_b}{N_o}\right) = SNR \Rightarrow N_o = \frac{E_b}{10^{\frac{SNR}{10}}}$$

Και ότι $noise = \text{sqrt}(N_o / 2) * \text{randn}((L_b / 2) * 40, 1)$

Τελικά το σήμα παίρνει τη μορφή : $signal = s_m + noise$, όπου s_m τα σήματα από πιο πάνω.

demodulator

Εδώ γίνεται η αποδιαμόρφωση του σήματος όταν αυτό φτάσει στο δέκτη , πράγμα που για να γίνει χρειάζεται ο πομπός και ο δέκτης να είναι πλήρως συγχρονισμένοι. Συσχετίζει το ληφθέν σήμα με τις συνιστώσες της φέρουσας και προκύπτει το διάνυσμα r που αποτελεί την εκτιμώμενη τιμή του συμβόλου.

Στο **PSK** το $r=[r1,r2]$, όπου $r1$ το γινόμενο του σήματος επί τη 1^{η} συνιστώσα και $r2$ το γινόμενο του σήματος με τη 2^{η} συνιστώσα.

Στο **FSK** το $r=[signal*\gamma]$, όπου γ το διάνυσμα που περιέχει για κάθε σύμβολο (0,1,2,3) και για κάθε δείγμα (1...40) το γινόμενο της αντίστοιχης συνιστώσας της φέρουσας με τον g_T .

Fwraths

Η έξοδος του αποδιαμορφωτή οδηγείται στο φωρατή ο οποίος αποφασίζει ποιο είναι το σύμβολο που έχει σταλθεί.

Για το **PSK** υπολογίζεται η μικρότερη απόσταση του διανύσματος s_m από το r με τη βοήθεια της ευκλείδειας νορμας.

Για το **FSK** υπολογίζεται η μέγιστη τιμή $\max(r)$. Η θέση που περιέχει τη μέγιστη τιμή αντιπροσωπεύει το σύμβολο που έχει σταλεί .

demapper

Η έξοδος πλέον του φωρατή οδηγείται στο demapper που κάνει την αντιστοίχιση- αποκωδικοποίηση των συμβόλων πλέον σε δυαδική ακολουθία και πάλι. Αυτό αντιστοιχα με το mapper γίνεται για το PSK με κώδικα GRAY ενώ στο FSK με κανονική δυαδική.

Ερωτημα 3 (BER-bit error rate)

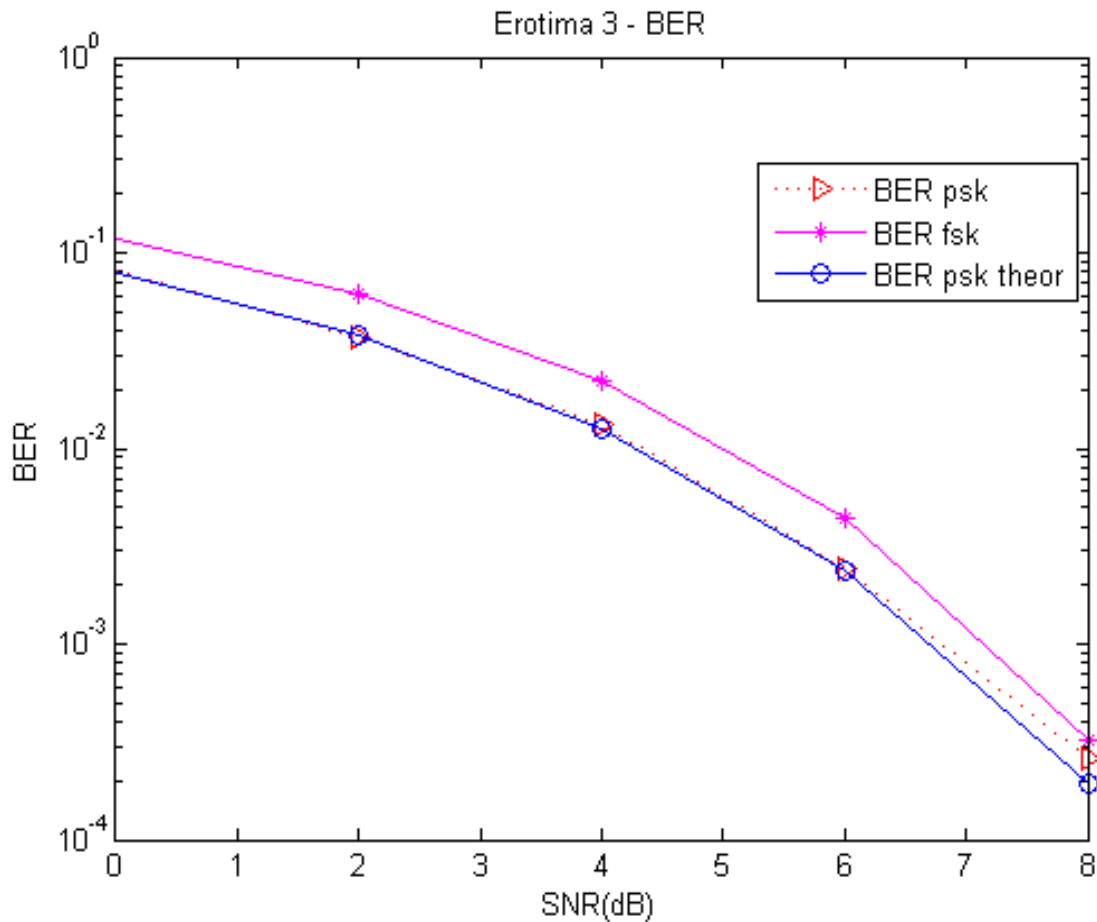
Η ακολουθία που χρησιμοποιήθηκε είχε μήκος 100.000(1.0e5) για αξιόπιστες μετρήσεις όπως υποδείχθηκε. Το θεωρητικό BER για το PSK μετρήθηκε ως εξής:

$$BER_{theor} = \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{E_b}{N_0}} \right)$$

και το πειραματικό για PSK & FSK αμφότερα μετρήθηκε ως:

$$BER_{exp} = \frac{false_bits}{L_b}$$

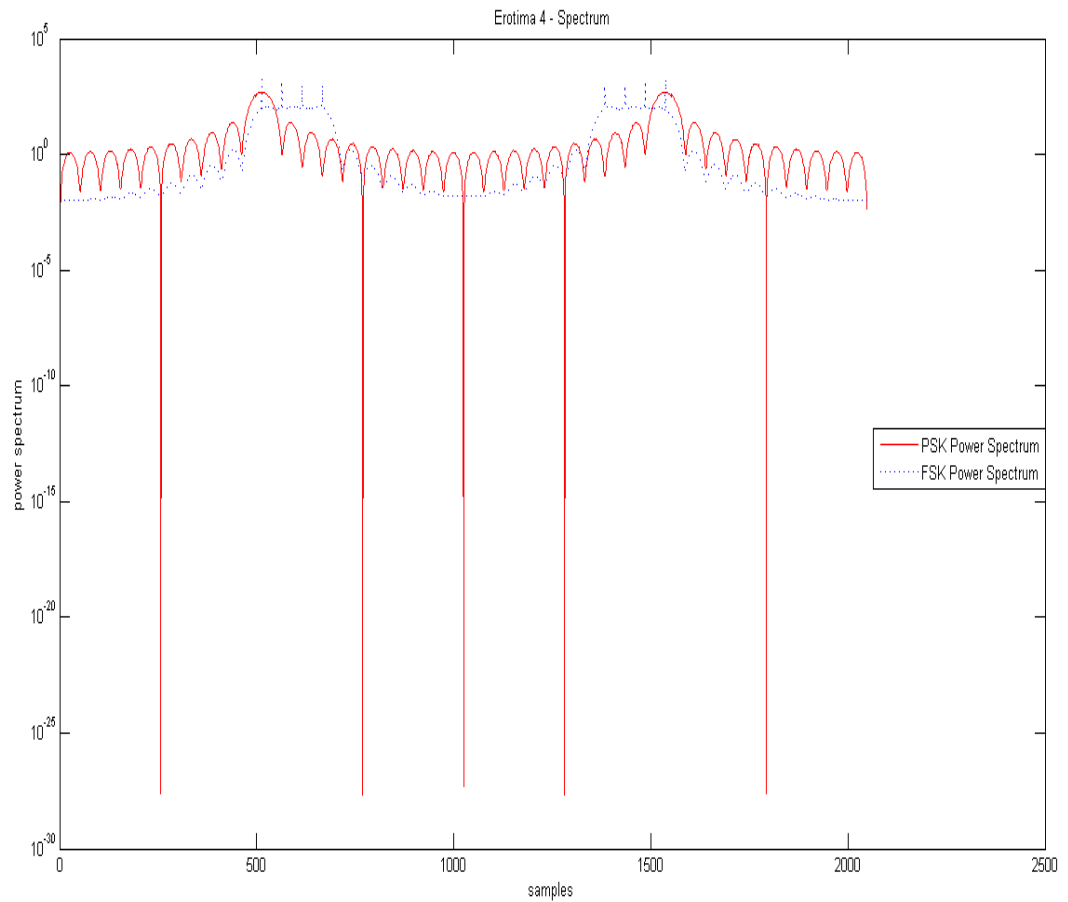
Τώρα τρέχοντας το κώδικα που δίνεται στην αρχή στη matlab παίρνουμε την εξής γραφική παράσταση.



Από τη γραφική παρατηρούμε ότι το μεγαλύτερο BER έχει το 4-FSK όπως και ήταν αναμενόμενο, το θεωρητικό με το πειραματικό BER για το 4-PSK είναι αρκετά κοντά για SNR 0:6 και από κει και πάνω υπάρχει μία μικρή απόκλιση που δείχνει ότι θεωρητικά το BER είναι λίγο χαμηλότερο->καλύτερο. Καταλήγοντας, το 4-PSK είναι καλύτερο ως προς τη πιθανότητα σφάλματος ως προς το ίδιο SNR από το 4-FSK και για να έχει το χειρότερο (4-FSK) το ίδιο BER με το 4-FSK απαιτείται το διπλάσιο SNR.

Ερώτημα 4 Φάσμα Ισχύος

Η γραφική που παίρνουμε για 2048 samples είναι :



Βλέπω ότι στο PSK απαιτείται μεγαλύτερο εύρος ζώνης. Το πλάτος των κύριων λοβών στο PSK είναι πάντα διπλάσιο από αυτό των δευτερευόντων, ενώ στο FSK το πλάτος των κύριων είναι 4-πλάσιο των δευτερευόντων που έχουν το μέγιστο πλάτος γιατί απ'ότι βλέπουμε φθίνουν οι δευτερεύοντες λοβοί.

Ερώτημα 5

Τα σχήματα είναι τα :

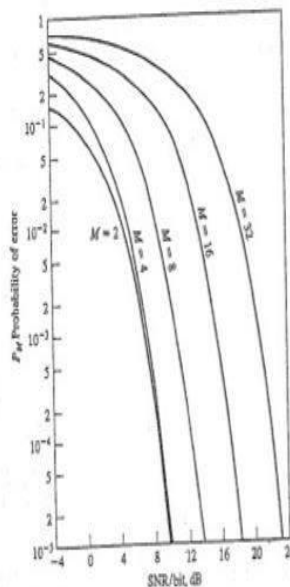


Figure 7.57 Probability of a symbol error for PSK signals.

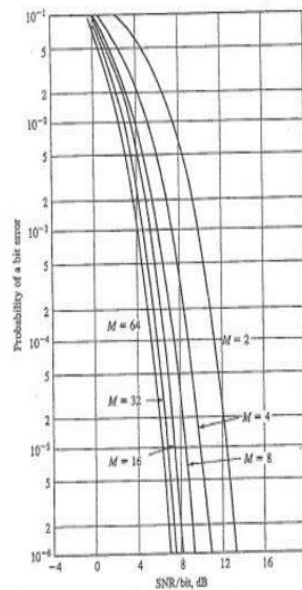


Figure 7.63 Probability of a bit error for coherent detection of orthogonal signals.

Παρατηρούμε από τα παραπάνω σχήματα ότι για τα δυαδικά ορθογώνια (M-FSK) όσο μεγαλώνει το M τόσο μικραίνει η απαίτηση SNR/bit και η καμπύλη BER βελτιώνεται ενώ για τα (M-PSK) παρατηρούμε ότι όσο μεγαλώνει το M τόσο μεγαλώνει και η απαίτηση SNR/bit (4-6(db) για την ακρίβεια) για να έχουμε το ίδιο SER και κατ'επέκταση BER αφού έχουν την ίδια μορφή αυτά τα διαγράμματα.

Όταν αυξάνει το M τα οι διαμορφώσεις M-PSK και M-FSK συμπεριφέρονται ως εξής:

PSK: αυξάνεται η απόδοση εύρους ζώνης αλλά μειώνεται η απόδοση ισχύος (αυξάνεται η πιθανότητα Ber)

FSK: αντίθετα στα ορθογώνια όταν το M αυξάνεται μειώνεται η απόδοση του εύρους ζώνης (αυξάνεται το απαιτούμενο εύρος κατά $M/(2\log_2 M)$) και αυξάνεται η απόδοση ισχύος (αφού μειωνεται η πιθανότητα Ber).