

ΨΗΦΙΑΚΕΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΕΣ

Φακωτάκης Ορέστης

ΑΜ=5673

Ακαδημαϊκό Έτος 2016-2017

1η Εργαστηριακή Άσκηση

Μέρος Α

Συμπύεση Διακριτής Πηγής με Χρήση της Κωδικοποίησης DPCM

Η κωδικοποίηση **DPCM** (Differential Pulse Code Modulation) μπορεί να θεωρηθεί ως μια γενίκευση της **κωδικοποίησης Δέλτα**

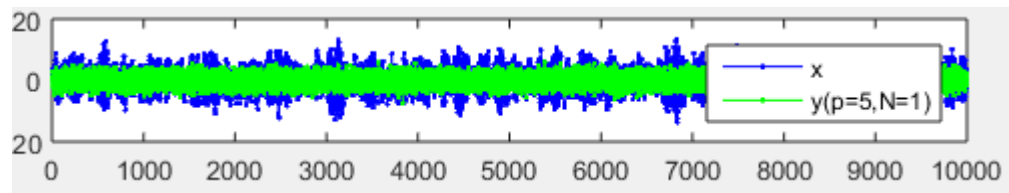
όπου το σήμα που κβαντίζεται και αποστέλλεται στο δέκτη,

είναι η διαφορά ανάμεσα στο τρέχον δείγμα (της χρονικής στιγμής n) και σε μία γραμμική πρόβλεψή του. Όταν έχουμε μια τυχαία διαδικασία περιορισμένου εύρους ζώνης και δειγματοληπτείται με ρυθμό **Nyquist** ή μεγαλύτερο τότε τα διαδοχικά δείγματα εμφανίζουν σημαντική συσχέτιση. Η συσχέτιση μπορεί να ελαττώσει τον αριθμό **bits/έξοδο** διατηρώντας την ίδια απόδοση. Η Γενική μέθοδος ονομάζεται Διαφορική Παλμοκωδική Διαμόρφωση (Differential PCM - DPCM), Η Διαμόρφωση Δέλτα (DM) είναι η απλούστερη μορφή DPCM χρησιμοποιεί μόνο το προηγούμενο δείγμα, όπως στο απλό DPCM ($p=1$). Χρησιμοποιεί κβαντιστή ενός μόνο bit.

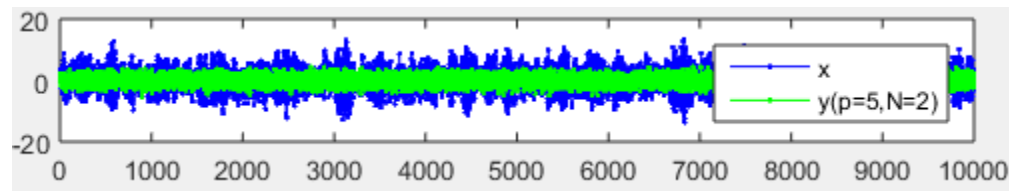
1. Αρχικά μας ζητήθηκε να υλοποιήσουμε, ένα σύστημα κωδικοποίησης/αποκωδικοποίησης DPCM. Το σύστημα αποτελείται από έναν ομοιόμορφο κβαντιστή (quantizer.m Στις παρενθέσεις είναι τα ονόματα των αντίστοιχων συναρτήσεων της MATLAB) ΚΑΙ ένα σύστημα πρόβλεψης-μνήμης (forecast.m).

2. Για $N=1,2,3$ και για $p=5, p=7$ παρατηρούμε το σφάλμα πρόβλεψης

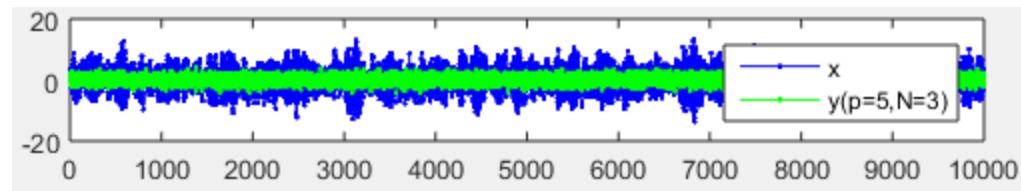
$p=5, N=1$



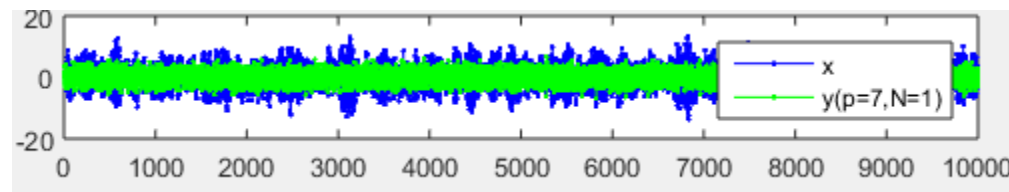
$p=5, N=2$



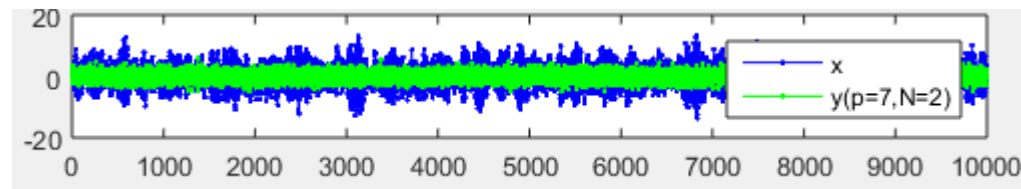
$p=5, N=3$



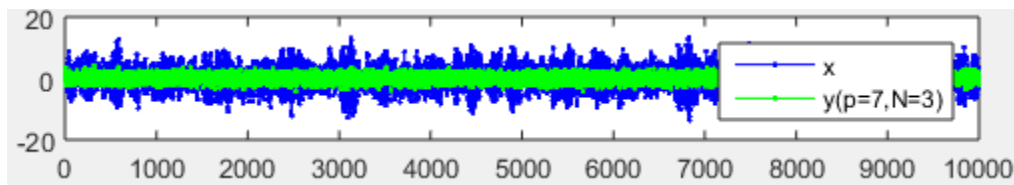
$p=7, N=1$



$p=7, N=2$

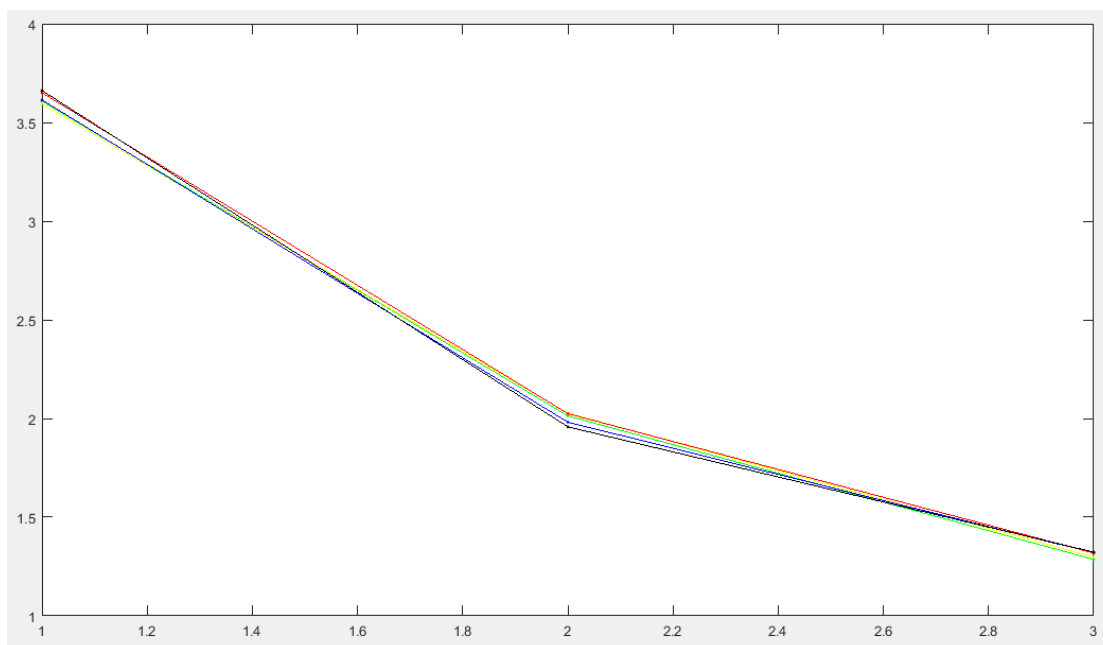


p=7,N=3



Όσο αυξανοντε τα bits της κβαντισης από 1 έως 3, παρατηρούμε αυξάνετε η ακρίβεια στην κβαντιση. Επίσης το σφάλμα κβαντισης παρατηρούμε οτι με την αύξηση του N μειώνετε το πλάτος του $y(n)$.

3.Το Μέσο Τετραγωνικό Σφάλμα Πρόβλεψης $E(y(n)^2)$ για $p=4:8$ και για $N=1,2,3$, παρατηρούμε οτι όσο το p μεγαλώνει τα N συγκλίνουν σε κοντινές μεσες τετραγωνικές τιμές η αλλιώς όσο το μεγαλώνει το p δεν προσφέρει μεγάλη βελτίωση στην πρόβλεψη



-->Επιπλέον, για κάθε p , οι τιμές των συντελεστών του προβλέπτη

Για $p=5$ και για $N=1,2,3$ bits

aq= 1.3828

-1.5234

1.2109

-0.3047

-0.0078

Για p=4 N=1,2,3 bits

aq=

1.3828

-1.5234

1.2109

-0.3047

Για p=6 και N=1,2,3 bits

aq=

1.3828

-1.5234

1.2109

-0.3047

0.0078

-0.0078

Για p=7 και N=1,2,3 bits

aq=

1.3828

-1.5234

1.2109

-0.3047

-0.0078

0.0078

-0.0078

Για $p=8$ και $N=1,2,3$ bits

$aq=$

1.3828

-1.5234

1.2109

-0.3047

0.0078

-0.0078

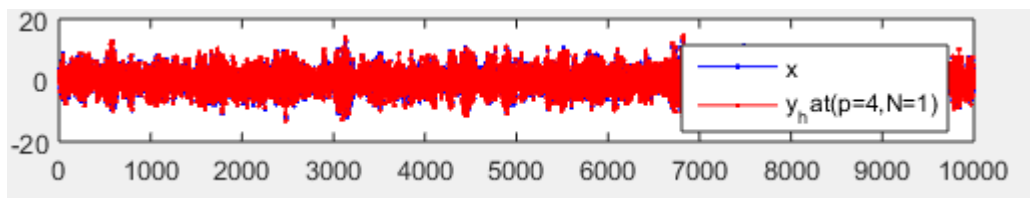
0.0078

-0.0078

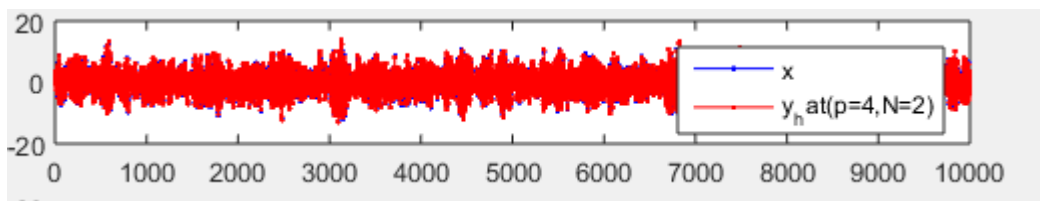
Οι τιμές των συντελεστών του προβλέπτη, παρατηρούμε ότι δεν επηρεάζονται από την αλλαγή των bits της κβαντισμού (N) του σφάλματος προβλεψής, αφού στην κβαντισμό του α (φίλτρου προβλεψής) το $N=8$ σταθερά.

4. Το ανακατασκευασμένο σήμα, δηλαδή το κβαντισμένο σφάλμα προβλεψής $y_q(n)$

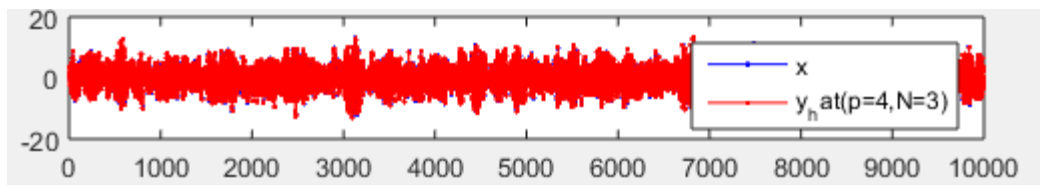
$p=4$ $N=1$



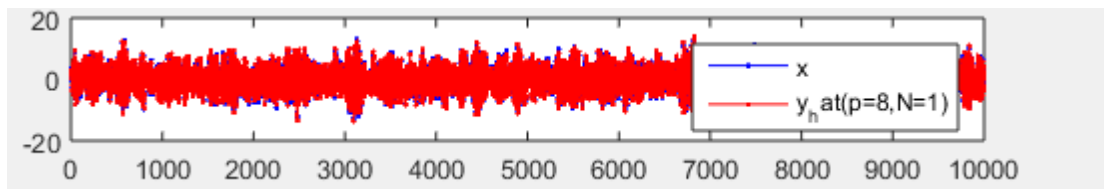
$p=4$ $N=2$



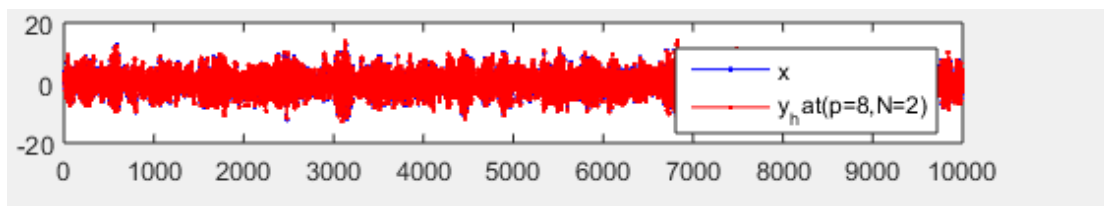
$p=4$ $N=3$



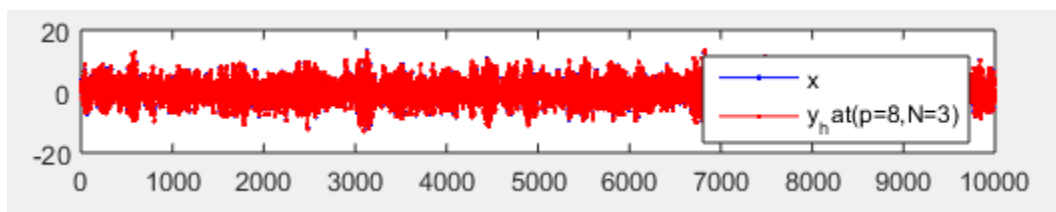
p=8 N=1



p=8 N=2



p=8 N=3



Το ανακατασκευασμένο σήμα, δηλαδή το κβαντισμένο σφάλμα προβλεψης $y_q(n)$ παρατηρούμε ότι όσο αυξανετε το N τινή να γινή ίδιο με το αρχικό της πηγής

Μέρος Β

Προσομοίωση Ομόδυνου Ζωνοπερατού

Συστήματος M-PSK

Ο πομπός του συστήματος M-PSK δέχεται ως είσοδο μια $\log_2(M)$ ακολουθία, την μετατρέπει σε σύμβολα, την πολλαπλασιάζει με τον ορθογώνιο παλμό, και κατόπιν το σήμα μεταφέρεται στην ζώνη μετάδοσης μέσω του διαμορφωτή. Στο σήμα που στάλθηκε προστίθεται AWGN θόρυβος, και φτάνει στο δέκτη του συστήματος. Εκεί αποδιαμορφώνεται και προκύπτει ένα δισδιάστατο διάνυσμα το οποίο εισάγεται στο φωρατή όπου και αποφασίζεται ποίο σύμβολο στάλθηκε. Τέλος, ο demapper κάνει την αντίστροφη αντιστοίχιση από σύμβολα σε δυαδικά ψηφία.

—

—

Η είσοδος του συστήματος είναι μια ακολουθία δυαδικών ψηφίων, όπου οι τιμές 0 και 1 εμφανίζονται ισοπίθانا. Ο mapper είναι ένας μετατροπέας δυαδικών ψηφίων σε σύμβολα. Κάθε σύμβολο αντιστοιχεί σε μια συγκεκριμένα δυάδα δυαδικών ψηφίων. Επομένως, ο mapper θα πρέπει για κάθε ζευγάρι δυαδικών ψηφίων να εξάγει και ένα από τα M σύμβολα της διαμόρφωσης. Αντίστοιχα, ο demapper δέχεται ως είσοδο το σύμβολο που έχει ανιχνεύσει ο φωρατής του δέκτη, και βγάζει το αντίστοιχο ζευγάρι δυαδικών ψηφίων. Η επιλογή των συμβόλων δεν είναι τυχαία. Υιοθετώντας την κωδικοποίηση Gray, αν δύο σύμβολα είναι γειτονικά στο δισδιάστατο χώρο σημάτων, τότε σε αυτά ανατίθενται διατάξεις δυαδικών ψηφίων που διαφέρουν μόνο κατά ένα δυαδικό ψηφίο μεταξύ τους.

ο κωδικας του β μερους ειναι μια προσπαθια ,ειναι ελατοματικος

ΚΩΔΙΚΕΣ MATLAB Α ΜΕΡΟΥΣ

QUANTIZER

```
function [yhat]=quantizer(x,N,maxvalue,minvalue)
```

```

L=2^N;
D=((maxvalue - minvalue) / L);

akra=zeros(L,1);
center=zeros(L,1);
yhat=zeros(length(x),1);

for i=1:length(x)
    if x(i)<= minvalue
        yhat(i)=minvalue;
    elseif x(i) >= maxvalue
        yhat(i) = maxvalue;
    end
end

for i=1:L+1
    akra(i)=minvalue+(i-1)*D;
end

center(1)=minvalue+(D/2);

for i=2:L
    center(i)=center(i-1)+ D;
end

for i = 1:length(x)
    for t= 1 : L
        if ((x(i)<akra(t+1) && x(i) >= akra(t)))
            yhat(i)= center(t);
        end
    end
end

```

FORECAST

```

function [a]=forecast(x,p)

r=zeros(p,1);
R=zeros(p,p);

for i=1:p
    for n=p+1:length(x)
        r(i)=x(n)*x(n-i)+r(i);
    end
    r(i)=r(i)*(1/(length(x)-p));
end
for i=1:p
    for j=1:p
        for n=p+1:length(x)+1
            R(i,j)= x(n-i)*x(n-j)+R(i,j);
        end

        R(i,j)= R(i,j)*(1/(length(x)-p+1));
    end
end

```



```
a = mldivide(R,r);  
end
```

MESO

```
function [meso ] = Meso(y)  
meso=0;  
yoliko=0;  
for n=1:10000  
    yoliko= yoliko+y(n)^2;  
end  
meso=yoliko/10000;  
end
```

MAIN

```
function [y,y_hat]=main(x,p,NA)  
N=10000;  
y_=zeros(N,1);  
y=zeros(N,1);  
y_hat=zeros(N,1);  
yhat=zeros(N,1);  
a=forecast(x,p);  
aq=quantizer(a,8,2,-2);  
aq  
for n=1:p  
    y(n)=x(n);  
    yhat(n)=quantizer(y(n),NA,3,-3);  
    y_hat(n)=yhat(n);  
end  
  
for n=p+1:N  
    for i=1:p  
        y_(n)=aq(i)*y_hat(n-i)+y_(n);  
    end  
    y(n)=x(n)-y_(n);  
    yhat(n)=quantizer(y(n),NA,3,-3);  
    y_hat(n)=y_(n)+yhat(n);  
end
```

```
l=[1:10000];  
plot(l,x,'b.-',l,y_hat,'r.-')  
end
```

CONTROL

```
function [] = control(x)  
N=10000;  
p = 4:8;  
NA = 1:3;  
yoliko=0;  
TetraSfalm =zeros(5,3);  
ya=zeros(N,1);  
y_a=zeros(N,1);
```

```

yb=zeros(N,1);
y_b=zeros(N,1);

yc=zeros(N,1);
y_c=zeros(N,1);

yd=zeros(N,1);
y_d=zeros(N,1);

ye=zeros(N,1);
y_e=zeros(N,1);

yf=zeros(N,1);
y_f=zeros(N,1);

[ya,y_a]=main(x,5,1);
[yb,y_b]=main(x,5,2);
[yc,y_c]=main(x,5,3);
[yd,y_d]=main(x,7,1);
[ye,y_e]=main(x,7,2);
[yf,y_f]=main(x,7,3);
[ys,y_s]=main(x,8,1);
[yt,y_t]=main(x,8,2);
[yk,y_k]=main(x,8,3);
[yq,y_q]=main(x,4,1);
[yw,y_w]=main(x,4,2);
[yn,y_n]=main(x,4,3);
for i = 1:5
    for j = 1:3
        [y,y_]= main(x,p(i),NA(j));
        TetraSfalm(i,j) = Meso(y);
    end
end

l=[1:3];

plot(l,TetraSfalm(1,:), 'b.-', l,TetraSfalm(2,:), 'r.-',
l,TetraSfalm(3,:), 'k.-', l,TetraSfalm(4,:), 'g.-', l,TetraSfalm(5,:), 'c.-')

%B erotima
l=[1:10000];
subplot(6,2,1);
plot(l,x, 'b.-', l,ya, 'g.-')
legend('x', 'y(p=5,N=1)')

subplot(6,2,2);
plot(l,x, 'b.-', l,yd, 'g.-')
legend('x', 'y(p=7,N=1)')

```

```

subplot(6,2,3);
plot(1,x,'b.-',1,yb,'g.-')
legend('x','y(p=5,N=2)')

subplot(6,2,4);
plot(1,x,'b.-',1,ye,'g.-')
legend('x','y(p=7,N=2)')

subplot(6,2,5);
plot(1,x,'b.-',1,yc,'g.-')
legend('x','y(p=5,N=3)')

subplot(6,2,6);
plot(1,x,'b.-',1,yf,'g.-')
legend('x','y(p=7,N=3)')

%D erothma
%apeikonisi x,y'kapelo
subplot(6,2,7);
plot(1,x,'b.-',1,y_q,'r.-')
legend('x','y_hat(p=4,N=1)')

subplot(6,2,8);
plot(1,x,'b.-',1,y_w,'r.-')
legend('x','y_hat(p=4,N=2)')

subplot(6,2,9);
plot(1,x,'b.-',1,y_n,'r.-')
legend('x','y_hat(p=4,N=3)')

subplot(6,2,10);
plot(1,x,'b.-',1,y_s,'r.-')
legend('x','y_hat(p=8,N=1)')

subplot(6,2,11);
plot(1,x,'b.-',1,y_t,'r.-')
legend('x','y_hat(p=8,N=2)')

subplot(6,2,12);
plot(1,x,'b.-',1,y_k,'r.-')
legend('x','y_hat(p=8,N=3)')

end

```

ΚΩΔΙΚΕΣ MATLAB Β ΜΕΡΟΣ

POMPOS

```

function output = Pompos(input,M)
M=log2(M)
Es = 1;
fc = 1/4;

```

```

Tsample = 1;
Tsymbold = 40;
T = Tsymbold/Tsample;
gt = sqrt(2*Es/Tsymbold);
L = length(input)/2;
ms = zeros(L,2);
symbols = zeros(M,2);
input = reshape(input, 2, L)';
ms=gray2bin(input, 'psk',M)
symbols(:,1) = sqrt(Es)*(cos(2*pi/4*(0:(M-1))))';
symbols(:,2) = sqrt(Es)*(sin(2*pi/4*(0:(M-1))))';
M1 = repmat(gt * ms(:,1),1,T)' .* repmat(gt * ms(:,2),1,T)';
M2 = repmat(cos(2*pi*fc*(0:T-1)),L,1)' .* repmat(sin(2*pi*fc*(0:T-1)),L,1)';
output = reshape(M1 + M2, L * T, 1);

```

KANALI

```

function 8oribos = WGN(n,SNR)
Tsymbold = 40;
Tsample = 1;
T = Tsymbold/Tsample;
Eb = 1/2;
Nok = Eb/(10^(SNR/10));
8oribos = sqrt(Nok/2) * randn(T*(n/2),1);

```

DEKTIS

```

function exodos = Dektis(input,mode)
M=log2(mode)
Es = 1;
fc = 1/4;
Tsample = 1;
Tsymbold = 40;
T = Tsymbold/Tsample;
gt = sqrt(2*Es/Tsymbold);
L = length(input)/T;
output = zeros(2,L);
symbolSamples = reshape(input, T, L);
symbols(:,1) = sqrt(Es)*(cos(2*pi/4*(0:(M-1))))';
symbols(:,2) = sqrt(Es)*(sin(2*pi/4*(0:(M-1))))';

dd(1:T,:) = diag(gt * cos(2*pi*fc*(0:T-1)));
dd(T+1:2*T,:) = diag(gt * sin(2*pi*fc*(0:T-1)));
received = zeros(L,2);
for i = 1 : 2
    a = T*(i-1) + 1;
    b = T * i;
    received(:,i) = sum(dd(a:b,:) * symbolSamples);
end
decSymbols = zeros(L,2);
for i = 1 : L
    A = repmat(received(i,:),M,1) - symbols;
    [x,indx] = min(max(abs(A),[],2));
    decSymbols(i,:) = symbols(indx,:);
end
end
g(k,:) = sum(gt * cos(a'*(0:T-1)) * resinput(:,k),2);
end

```

```
decSymbols = zeros(L,2);  
for i = 1:L  
    [x,indx] = max(g(i,:));  
    decSymbols(i,:) = symbols(indx,:);  
end
```

9

```
% Demapper  
exodos=gray2bin(decSymbols,'psk',M)  
output = reshape(output, 2*L,1);
```