

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ

ELE 361 HABERLEŞME SİSTEMLERİ 1

MATLAB PROJESİ

SAYISAL KİPLEME VE BENZETİMİ

HAZIRLAYAN

Seçkin Burak Cengiz 06120067

PROJENİN TANIMI: 16 PSK kipleme tekniği ile bir tabanbant sinyalin kiplenmesi

PROJENİN AMACI: Bir tabanbant sinyalini 16 PSK kipleme tekniği ile kipleyerek sembol hata oranı simülasyonunu yapmak ve SNR- SER grafiklerini çizdirmek.

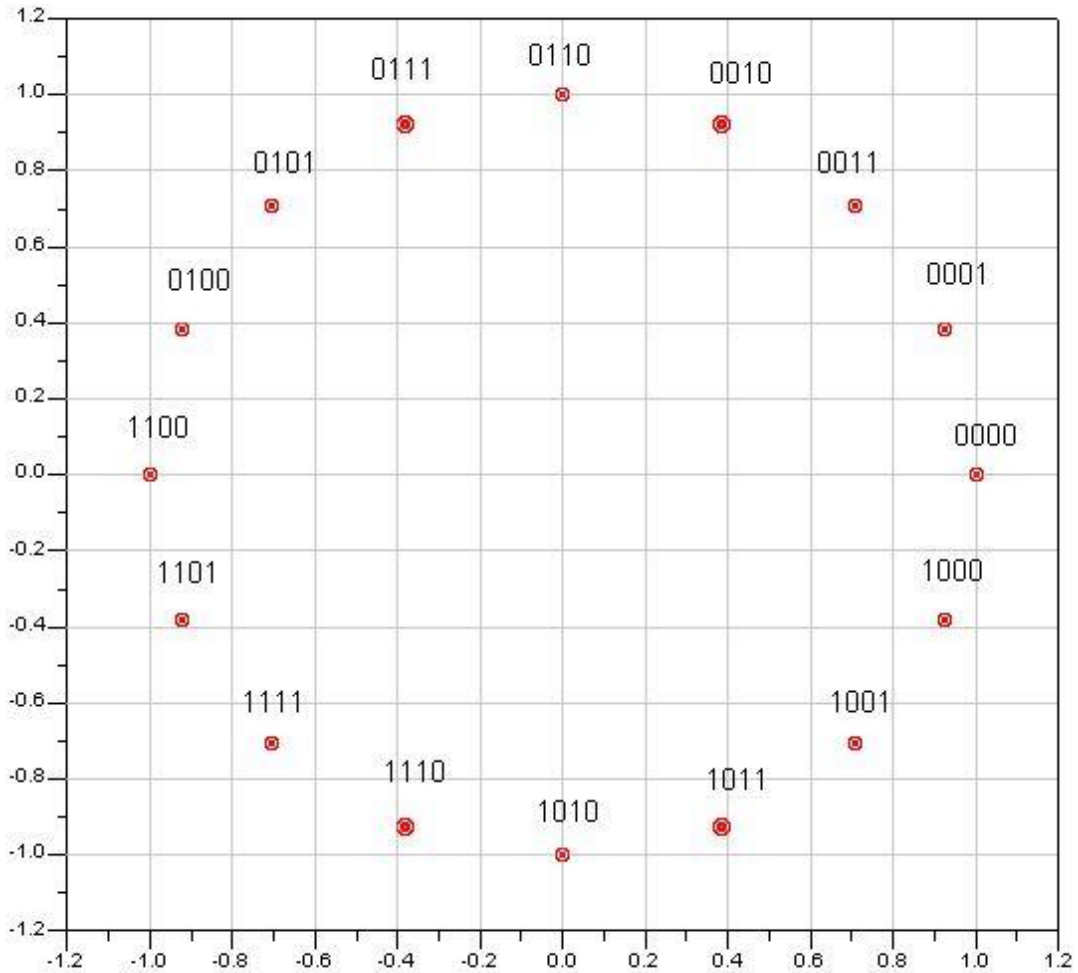
PROJE SON TESLİM TARİHİ: 03.05.12

TESLİM TARİHİ: 27.04.12

GEREKLİ ÖN TANIMLAR:

16 PSK kipleme tekniği : Dijital haberleşme de kullanılan bir kipleme tekniğidir. Bu kipleme tekniğinde iletilen sinyalin fazı çeşitlendirilerek bilginin taşınması sağlanır. 16 PSK yönteminde 16 adet farklı sembol yani faz bulunur. Bu sembollerin her biri 4 bitlik Gray kodu olarak ifade edilerek komşu fazlar arasında tek dijital farkı olacak şekilde yerleştirilmiştir. Bunun sebebi; gürültüden dolayı oluşan en olası hata iletilen faza komşu fazın seçilmesi ile yapılmıştır, bu durumda Gray kodlamayla k-bitlik dizide tek bit hatası oluşur. İki sembol arasındaki mesafe öklit kuralı ile hesaplanır. 16 PSK için 16 adet sembol olduğundan bu semboller arasındaki açı 22.5 derece olacaktır. Bu projede semboller bu kurala göre dizilmiştir ve her bir sembol için 4 bitlik gray kodu kullanılmıştır. Aşağıda Matlab da kullanılan sembol haritası ve haritaya göre dizilmiş sembollerin gösterildiği resim görülmektedir.

```
map=[-1 0;-0.92 0.39;-0.7 0.7;-0.39 0.92;0 1;0.39 0.92;0.7 0.7;0.92 0.39;1 0;0.92 -0.39;0.7 -0.7;0.39 -0.92;0 -1;-0.39 -0.92;-0.7 -0.7;-0.92 -0.39];
```



16 PSK kipleme yöntemi için kullanılan teorik bilgiler aşağıda verilmiştir. Bu teorik bilgiler yazılan ana kodda uygulanmış ve aşağıda belirtilmiştir.

Kullanılan temel formüller aşağıdaki gibidir;

$$k = \log_2(M).$$

$$\frac{E_s}{N_0} = k \frac{E_b}{N_0}.$$

$$P_b = \frac{1}{k} \operatorname{erfc} \left[\sqrt{\frac{k E_b}{N_0}} \sin \left(\frac{\pi}{M} \right) \right].$$

Matlab da simülasyon ve Teorik olarak hesaplanan BER değerleri yukardaki formüller esas alınarak düzenlenmiştir ve aşağıdaki gibidir.

```
simulatedBER(count) = log10(bitErrors/N);  
theoreticalBER(count) = log10((1/4)*erfc(sqrt(EbN0*4)*sin(pi/M)));
```

Desibel (dB) : Belirli bir referans güç ya da miktar seviyeye olan oranı belirten genelde ses şiddeti için kullanılan logaritmik ve boyutsuz bir birimdir. Desibel daima iki değer arasındaki karşılaştırmadır. Bunun sonucu olarak da, çoğu kez ölçülen güç değeri değişik olmasına rağmen desibel sayısı aynıdır. Desibelin yaygın olarak ses şiddeti birimi olduğu sanılır ama ses şiddeti karşılaştırmalarında da kullanılabilen bir karşılaştırmadır. Not: dB= 20*log10 ()

(Sinyal Gürültü Oranı)(Signal to Noise Ratio) SNR : İletilmek istenen bilgi sinyali ile bu sinyalin üzerindeki gürültü bileşenleri arasındaki orana sinyal gürültü oranı (SNR veya S/N) denilir.

MATLAB KODLARI

Binary den Gray koda çevirmek için yazılan fonksiyon

```
%Binary den Gray koda çevirim  
  
function [grayCoded]=bin2gray(binaryInput)  
  
    [rows,cols]=size(binaryInput);  
  
    grayCoded=zeros(rows,cols);  
  
    for i=1:rows
```

Gray koddan Binary koda çevirmek için yazılan fonksiyon

```
%Gray koddan Binary' e çevirim  
  
function [binaryCoded]=gray2bin(grayInput)  
  
    [rows,cols]=size(grayInput);  
  
    binaryCoded=zeros(rows,cols);  
  
    for i=1:rows  
  
        binaryCoded(i,1)=grayInput(i,1);  
  
        for i=2:cols
```

8-PSK simülasyonunun ve teorik hesaplamalarının yapıldığı ana kod

```
%16-PSK Ana Kod
clc; clear; close all; % Bir önceki kodlardan kalan değişkenleri siler, Komut Penceresini temizler.
N=10000; %Kanal üzerinden gönderilecek toplam bit sayısı. (Parametrik bir değerdir, değiştirilebilir)
EbN0dB=-4:2:20; % SNR için -4 den 16 ya kadar 2 şer artarak değişen değerler sağlar.
N=N+rem((4-rem(N,4)),4); %16-PSK 4 bit taşıyacak şekilde ayarlanır.
x=rand(1,N)>=0.5;%Random 1 ve 0 datalar oluşturur;
%Her sembolü 4 bitlik gray kodlar ile ifade etmek için;
inputSymBin=reshape(x,4,N/4)';
g=bin2gray(inputSymBin);
%Her sembol için oluşturulan gray kodları onluk taban çevirir.
b=bin2dec(num2str(g,'%1d'))';
%8-PSK Yıldız Kümesi Haritalama
%8-PSK Haritalama Tablosu
map=[-1 0;-0.92 0.39;-0.7 0.7;-0.39 0.92;0 1;0.39 0.92;0.7 0.7;0.92 0.39;1 0;0.92 -0.39;0.7 -0.7;0.39
-0.92;0 -1;-0.39 -0.92;-0.7 -0.7;-0.92 -0.39];
s=map(b(:)+1,1)+1i*map(b(:)+1,2);
%Her Eb/N0 değeri için farklı simülasyon yaptırmak için.
M=16; %Sembol sayısı; M=2^k. 16-PSK için k=4 olacaktır.
Rm=log2(M); %Rm=log2(M). 16-PSK için M=16 olacaktır.
Rc=2; %Kodlama oranı.

simulatedBER = zeros(1,length(EbN0dB));
theoreticalBER = zeros(1,length(EbN0dB));
count=1;

figure
for i=EbN0dB
    %-----
    %Eb/N0 değerleri için kanal gürültüsü
    %-----
    %Gerekli Eb/N0 değerleri için gürültü ekler.
    EbN0 = 10.^(i/10); %Eb/N0 dB değerini doğrusal ölçekler.
    SNR = 10+10*log10(4);

    noiseSigma = sqrt(2)*sqrt(1./(2*Rm*Rc*EbN0)); %AWGN gürültüsü için standart sapma
    disp('Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir');
    disp(noiseSigma);
    %Kompleks gürültü ekler
    n = noiseSigma*(randn(1,length(s))+1i*randn(1,length(s)));
    y = s + n;

    plot(real(y),imag(y),'r*');hold on;
    plot(real(s),imag(s),'ko','MarkerFaceColor','g','MarkerSize',8);hold off;
    title(['Yıldız Kümesi SNR =',num2str(i),' dB']);legend('Gürültü eklenmiş y sinyali','16-PSK');
    pause;
    %Demodulasyon
    %Öklid teoremini kullanarak en kısa mesafe için MAP tablosundan sinyal noktalarını hesaplar.
    demodSymbols = zeros(1,length(y));
    for j=1:length(y)
        [minVal,minindex]=min(sqrt((real(y(j))-map(:,1)).^2+(imag(y(j))-map(:,2)).^2));
        demodSymbols(j)=minindex-1;
    end

    demodBits=dec2bin(demodSymbols)-'0'; %Ondalık değeri ikilik tabandaki vektörlere çevirir.

    xBar=gray2bin(demodBits)'; %Gray kodları ikilik tabana çevirir.
    xBar=xBar(:)';

    bitErrors=sum(sum(xor(x,xBar)));
    simulatedBER(count) = log10(bitErrors/N);
    theoreticalBER(count) = log10((1/4)*erfc(sqrt(EbN0*4)*sin(pi/M)));

    count=count+1;
end

figure
plot(EbN0dB,theoreticalBER,'r-*');hold on;
plot(EbN0dB,simulatedBER,'k-o');

title('16-PSK için BER - SNR (dB) grafiği ');legend('Teorik','Simülasyon');grid on;
xlabel('SNR dB');
ylabel('BER - Bit Error Rate (Bit Hata Oranı)');
grid on;
```

MATLAB SİMÜLASYON ÇIKTILARI

Bu projede matlab kodları -4 ile 20 db arasında 2 şer artarak değişen SNR değerleri için gürültü eklenmiş sinyali ve Yıldız kümesini çizer. Program koşturulunca ilk olarak -4db SNR değeri için oluşan yıldız kümesi görülecektir, daha sonra “Enter” tuşuna basılarak SNR değerlerinin artması ile oluşan yeni yıldız kümeleri sırasıyla gözlemlenir. Yıldız kümeleri gösterilirken eş zamanlı olarak Matlab Command Window penceresinde, hesaplanan sembol gürültü oranları gösterilir. Son olarak 20db için oluşan yıldız kümesi grafiği oluştuktan sonra “Enter” tuşuna basıldığında sayısal kiplemenin değişik SNR değerlerine karşılık gelen sembol hata oranlarının grafiği oluşur. Bağlı parametreler değiştirilerek bu değişimlerin sonuçları gözlemlenebilir. Bu parametreler ve oluşan çıktılar sırasıyla aşağıda gösterilmiştir.

Programda değiştirilebilir parametreler :

SNR aralığı değiştirilebilir. -4 ile 20 arasında 1 er artarak değişen SNR uygulanabilir.

`EbN0dB=-4:2:20; % SNR için -4 den 16 ya kadar 2 şer artarak değişen değerler sağlar.`

N değeri değiştirilebilir. 10000 olarak girilen değer 100000 yapılabilir.

`N=10000; %Kanal üzerinden gönderilecek toplam bit sayısı. (Parametrik bir değerdir, değiştirilebilir)`

Sembol haritası değiştirilebilir. Sembollerin koordinatları uygun değerlerle değiştirilebilir.

`map=[-1 0;-0.92 0.39;-0.7 0.7;-0.39 0.92;0 1;0.39 0.92;0.7 0.7;0.92 0.39;1 0;0.92 -0.39;0.7 -0.7;0.39 -0.92;0 -1;-0.39 -0.92;-0.7 -0.7;-0.92 -0.39];`

Formüller değiştirilebilir.

`simulatedBER(count) = log10(bitErrors/N);
theoreticalBER(count) = log10((1/4)*erfc(sqrt(EbN0*4)*sin(pi/M)));`

Oluşan çıktılar sırasıyla aşağıdaki gibidir;

Workspace				
Stack: Base Select data to plot				
Name	Value	Min	Max	
EbN0	100	100	100	
EbN0dB	<1x13 double>	-4	20	
M	16	16	16	
N	10000	10000	10000	
Rc	2	2	2	
Rm	4	4	4	
SNR	16.0206	16.0206	16.0206	
b	<1x2500 double>	0	15	
bitErrors	0	0	0	
count	14	14	14	
demodBits	<2500x4 double>	0	1	
demodSymbols	<1x2500 double>	0	15	
g	<2500x4 double>	0	1	
i	20	20	20	
inputSymBin	<2500x4 logical>			
j	2500	2500	2500	
map	<16x2 double>	-1	1	
minVal	0.0118	0.0118	0.0118	
minindex	5	5	5	
n	<2500x1 complex do...	2.2001...	0.1325...	
noiseSigma	0.0354	0.0354	0.0354	
s	<2500x1 complex do...	-0.700...	-1	
simulatedBER	<1x13 double>	-Inf	-0.5414	
theoreticalBER	<1x13 double>	-8.0669	-0.7401	
x	<1x10000 logical>			
xBar	<1x10000 double>	0	1	
y	<2500x1 complex do...	-0.326...	0.0359...	

```

Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.

Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.5603

Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.4451

Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.3536

Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.2808

Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.2231

Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.1772

Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.1408

Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.1118

Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.0888

Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.0705

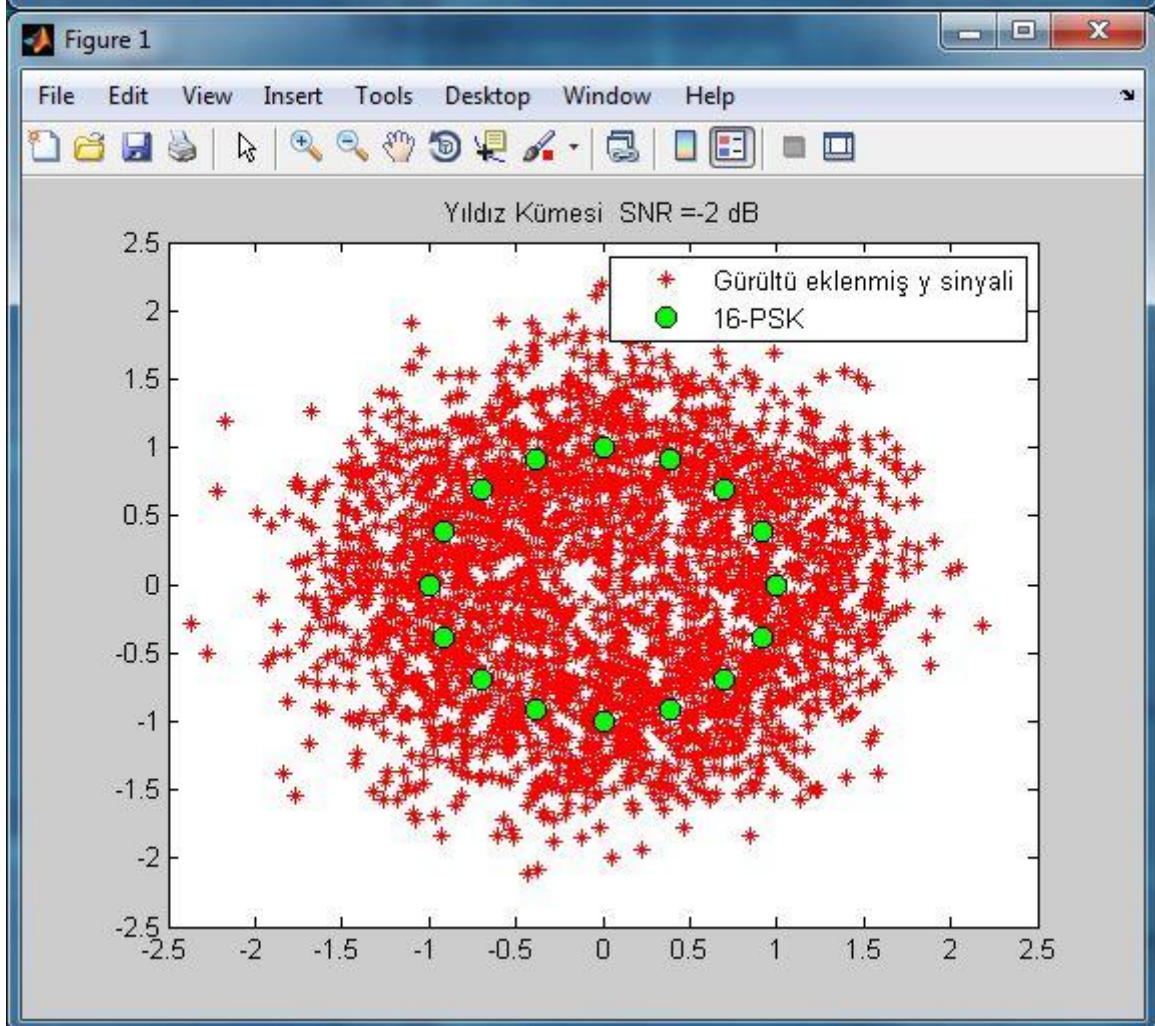
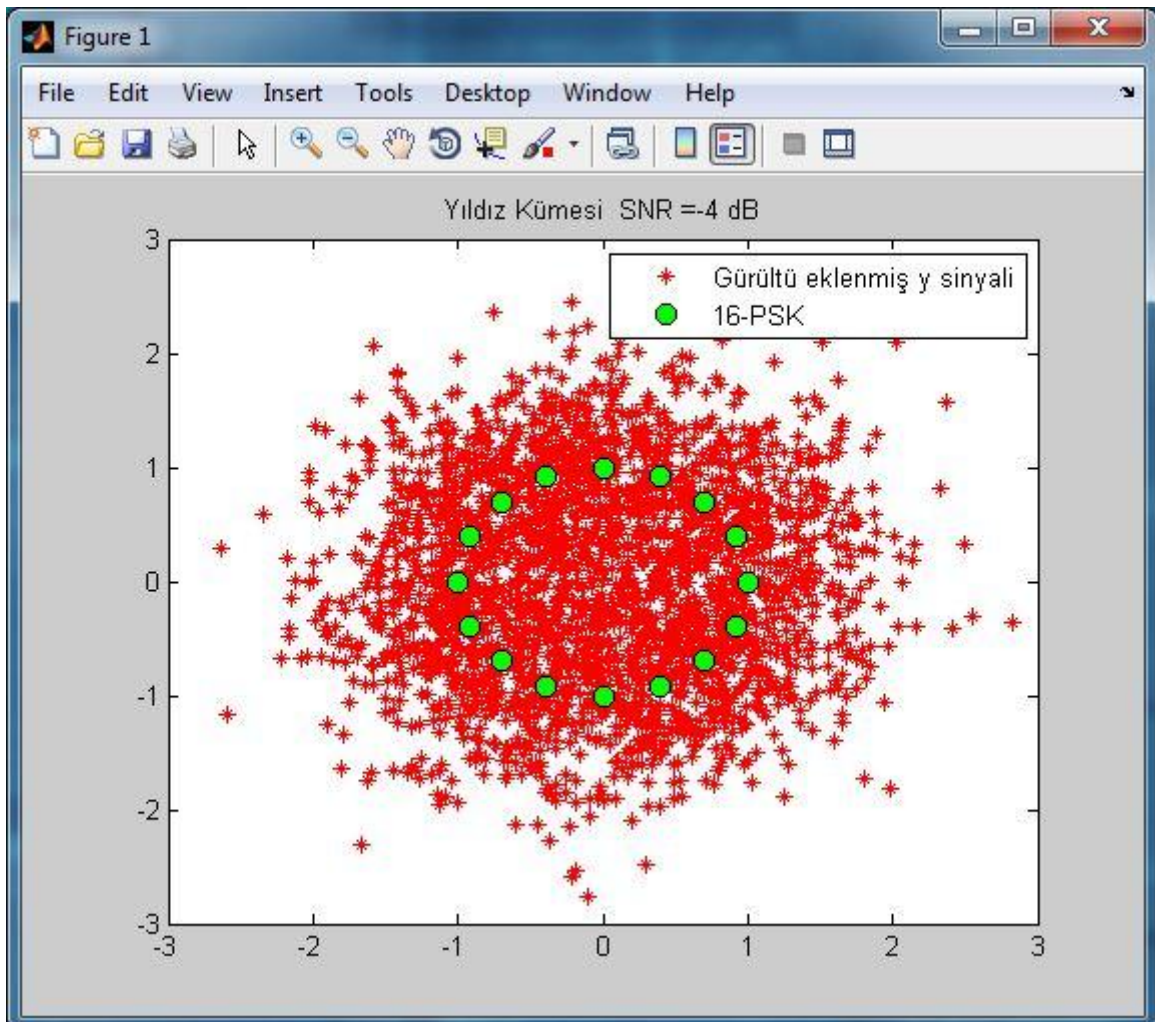
Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.0560

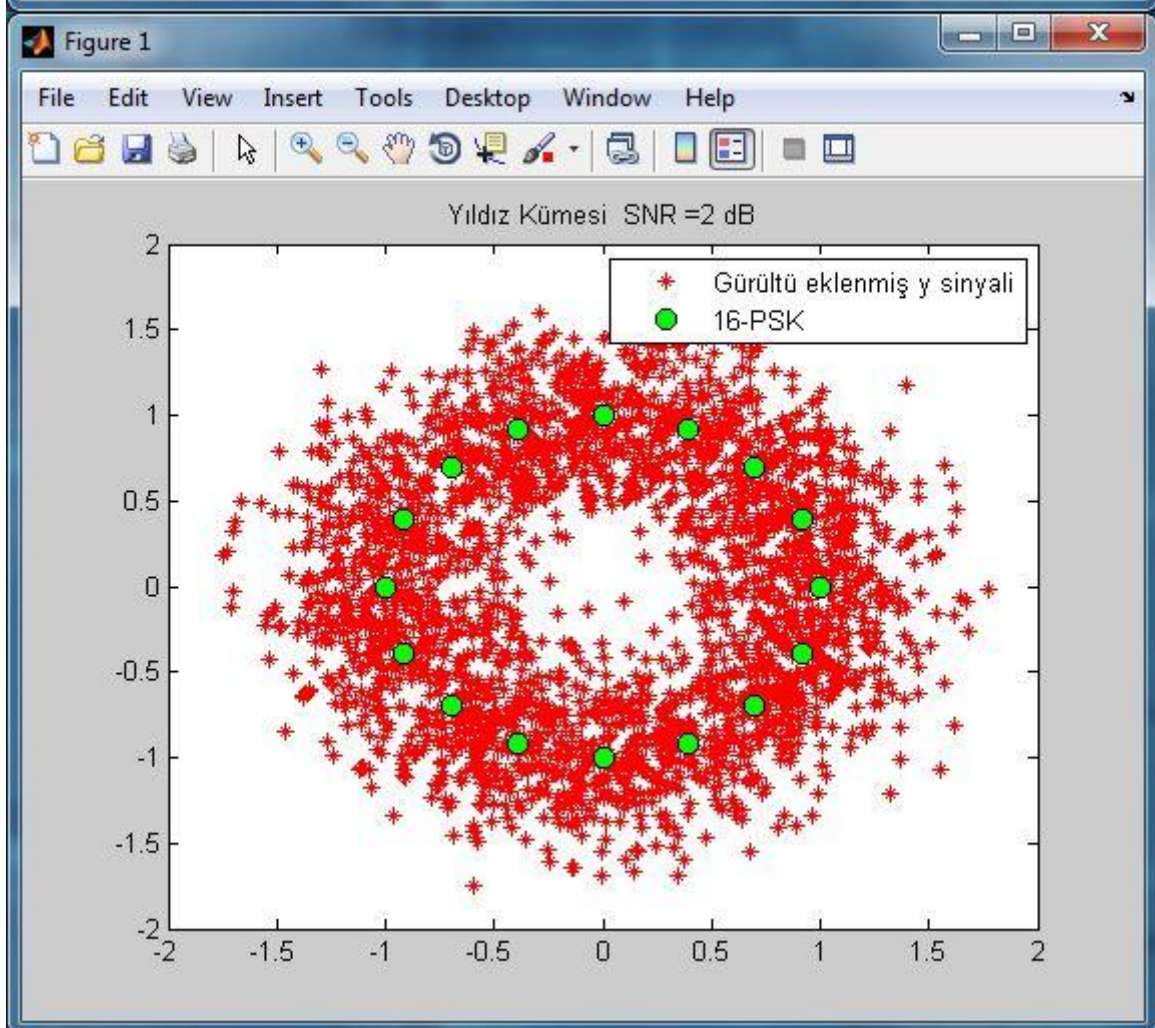
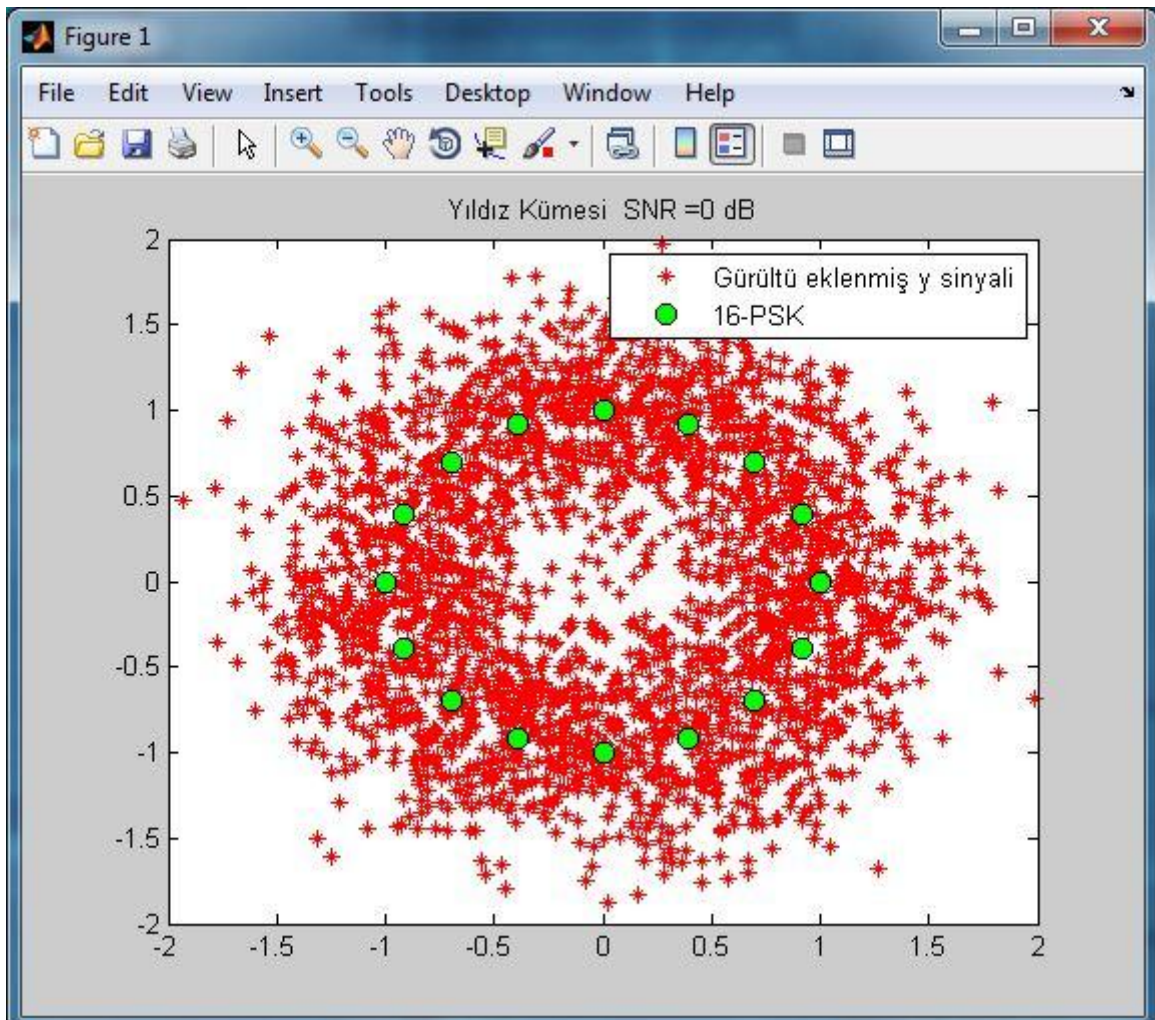
Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.0445

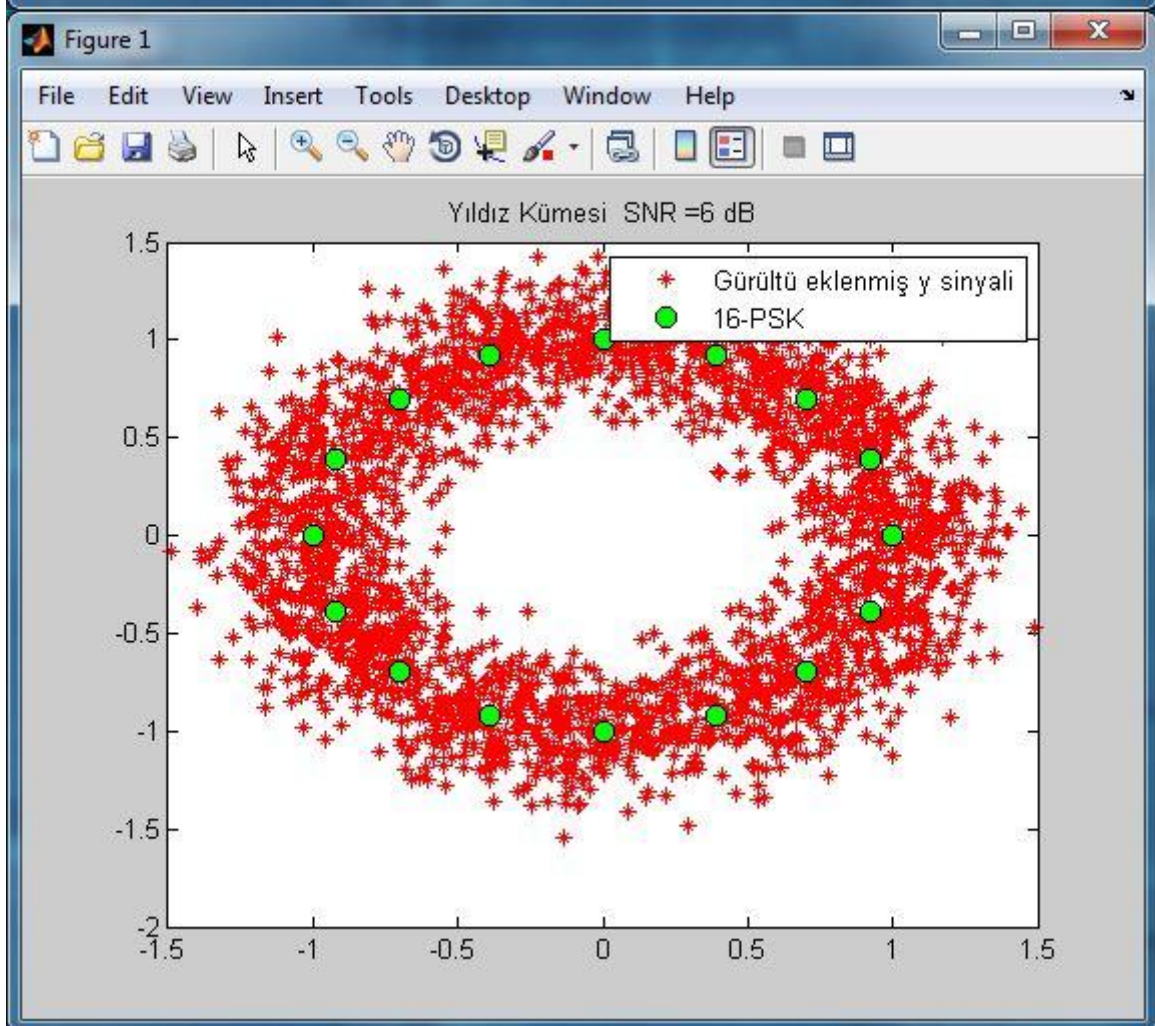
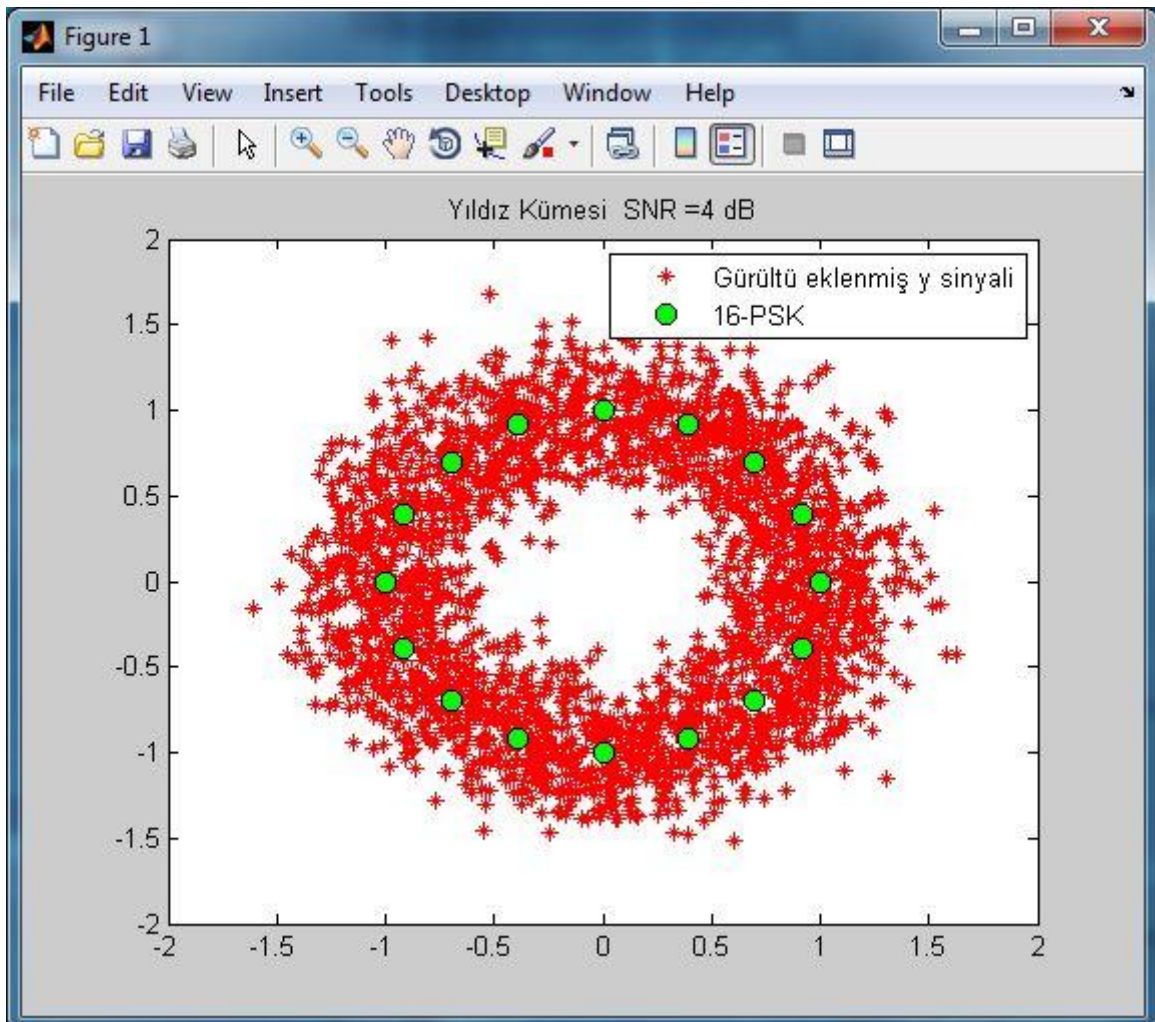
Sembol gürültü oranı şu andaki grafik için aşağıdaki gibidir
0.0354

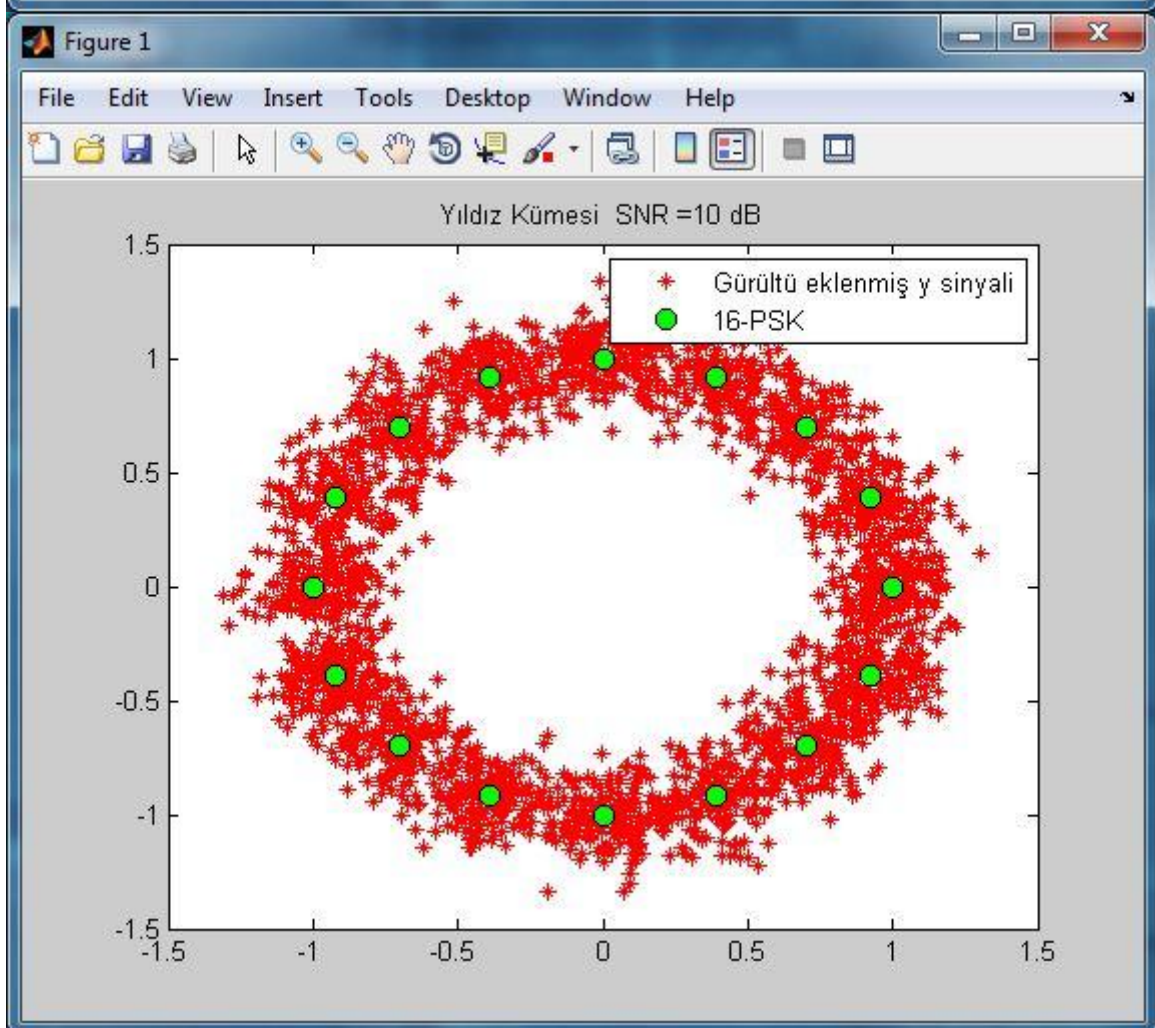
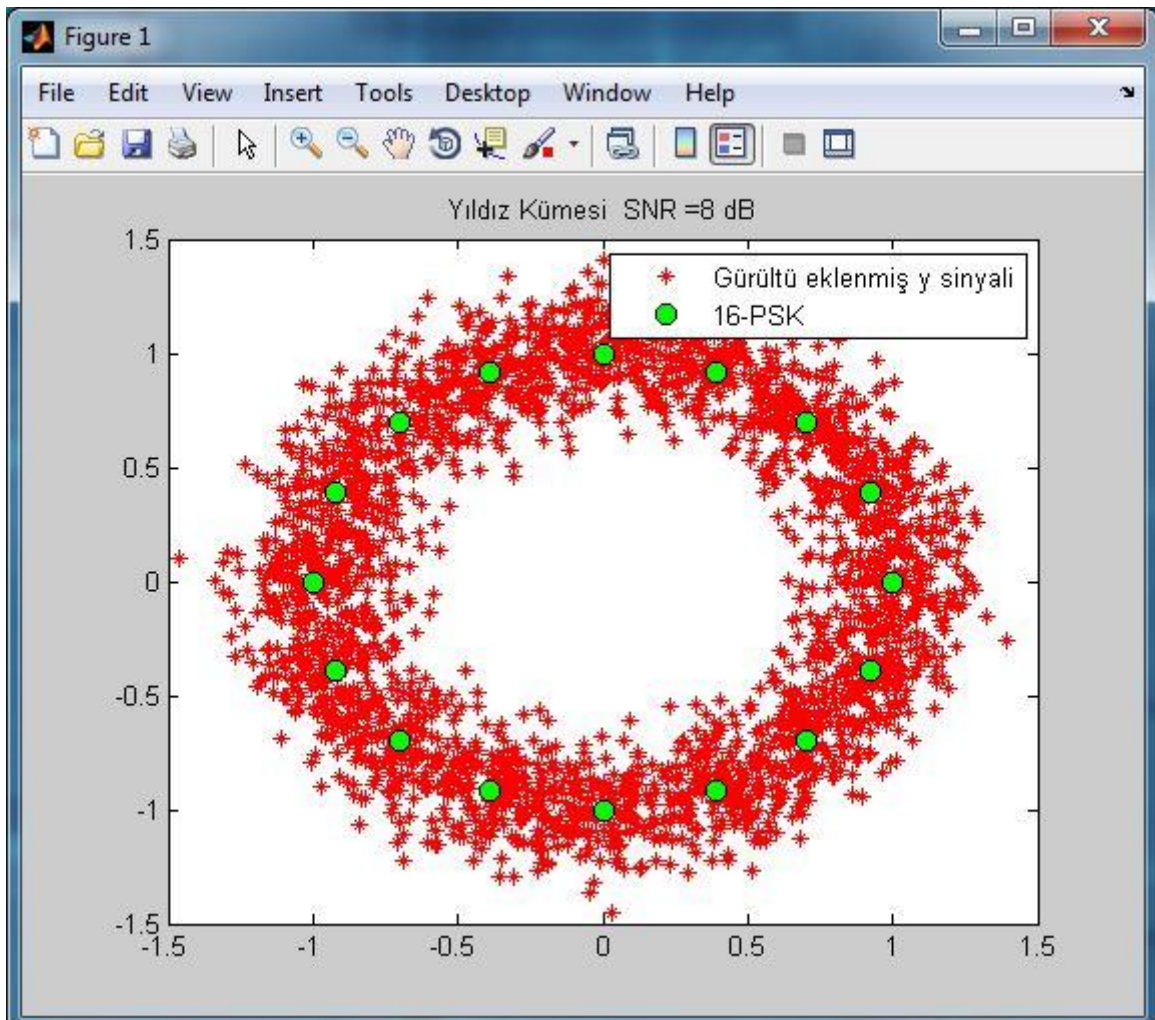
fx >> |

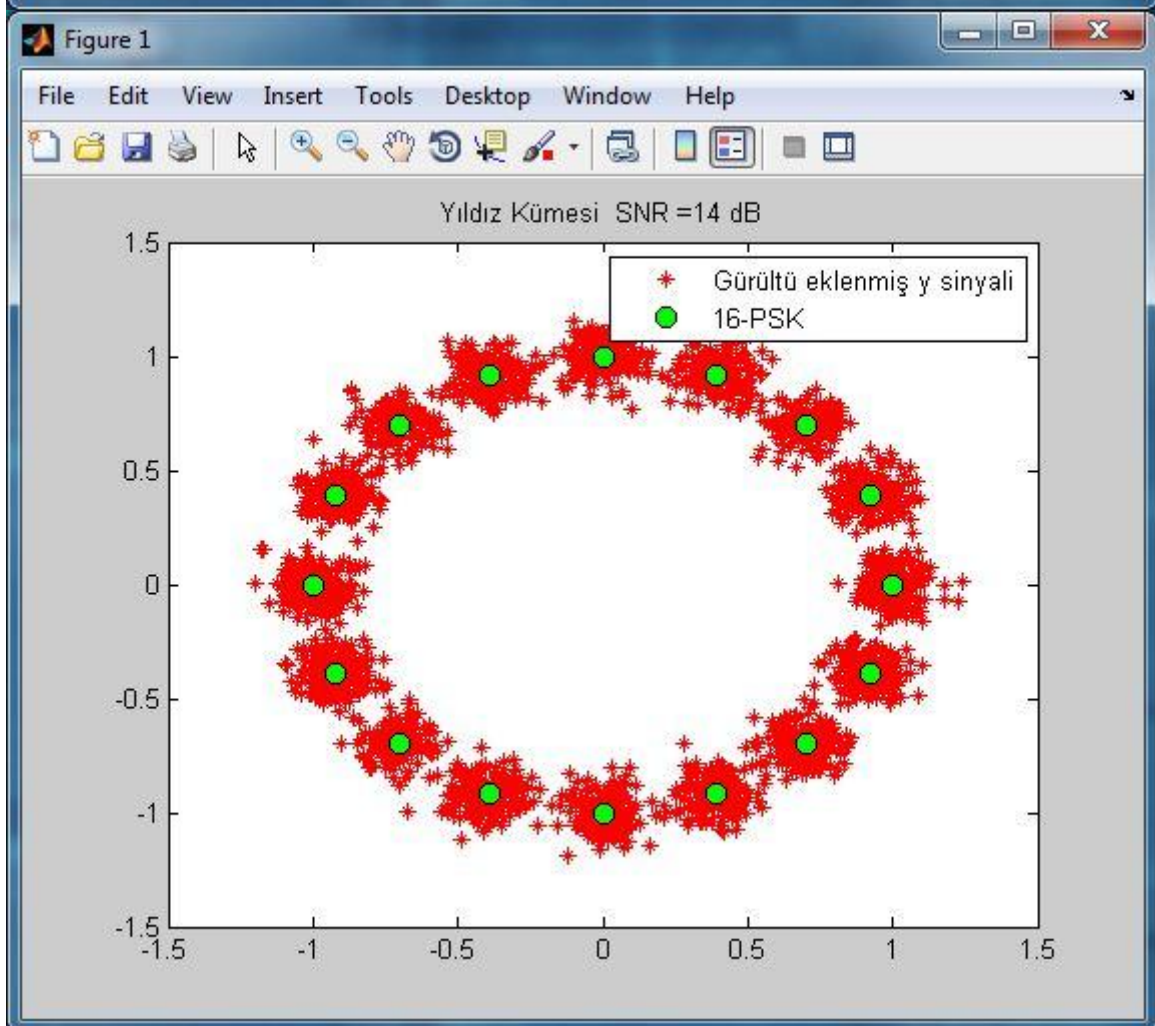
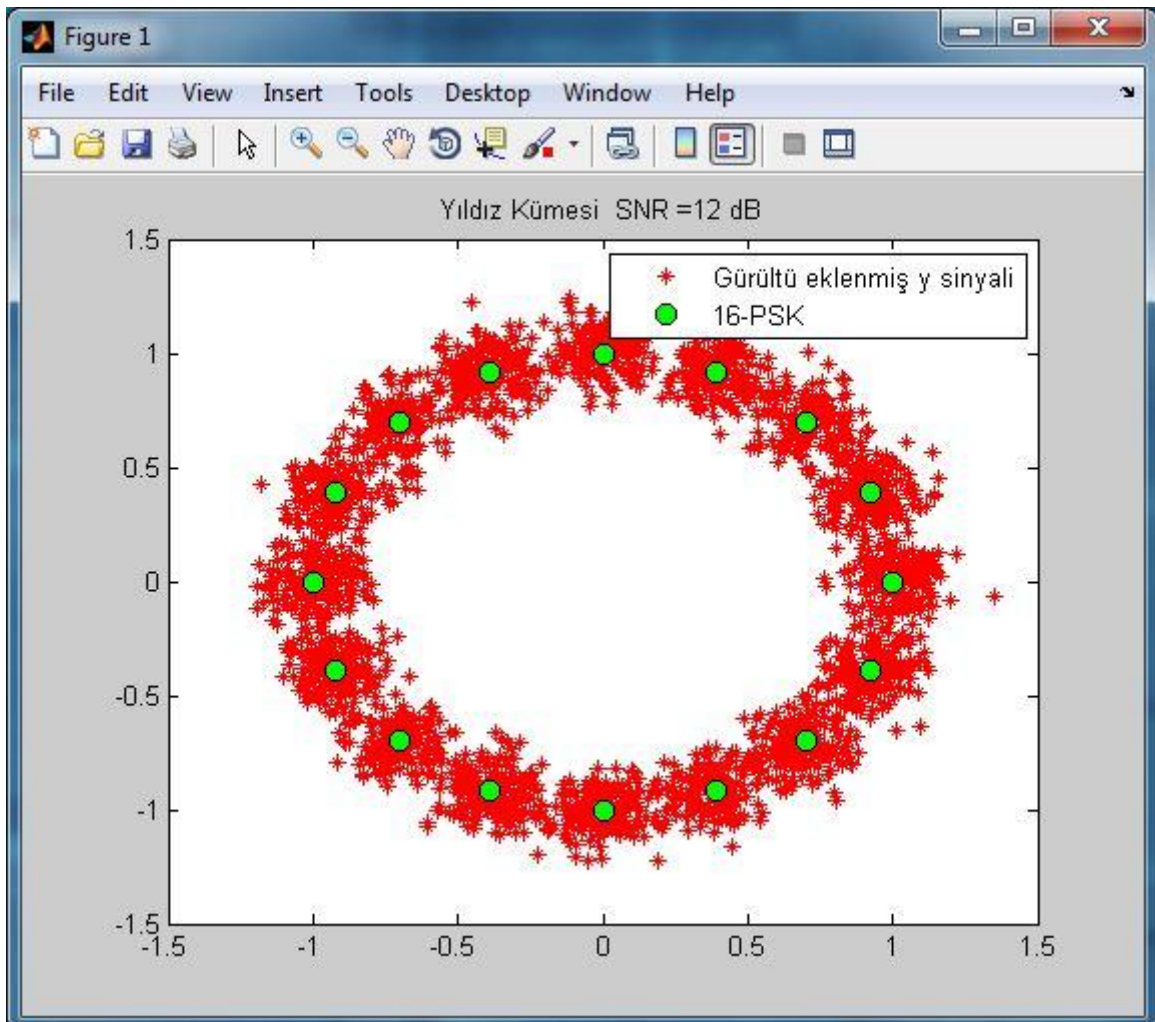
```

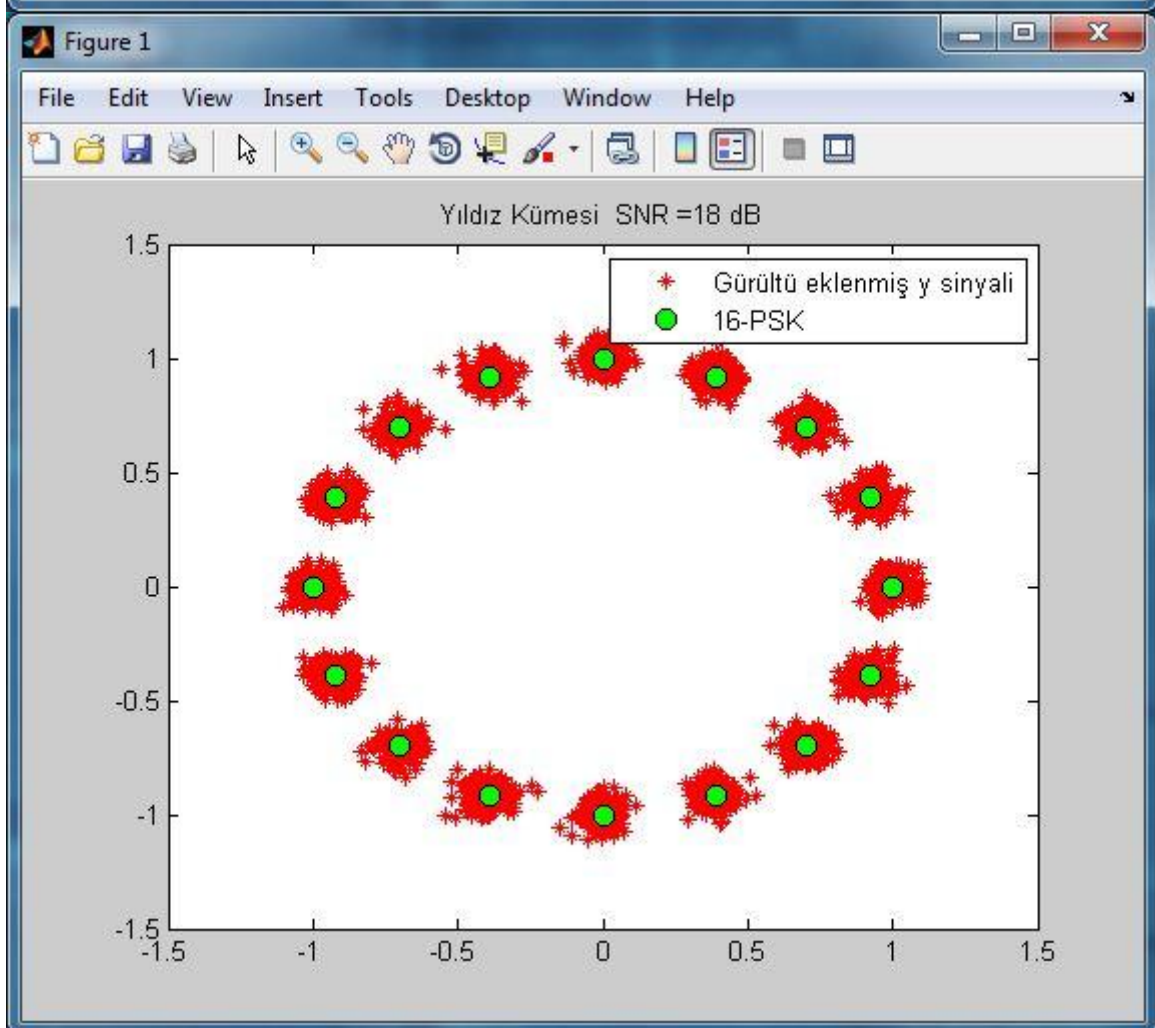
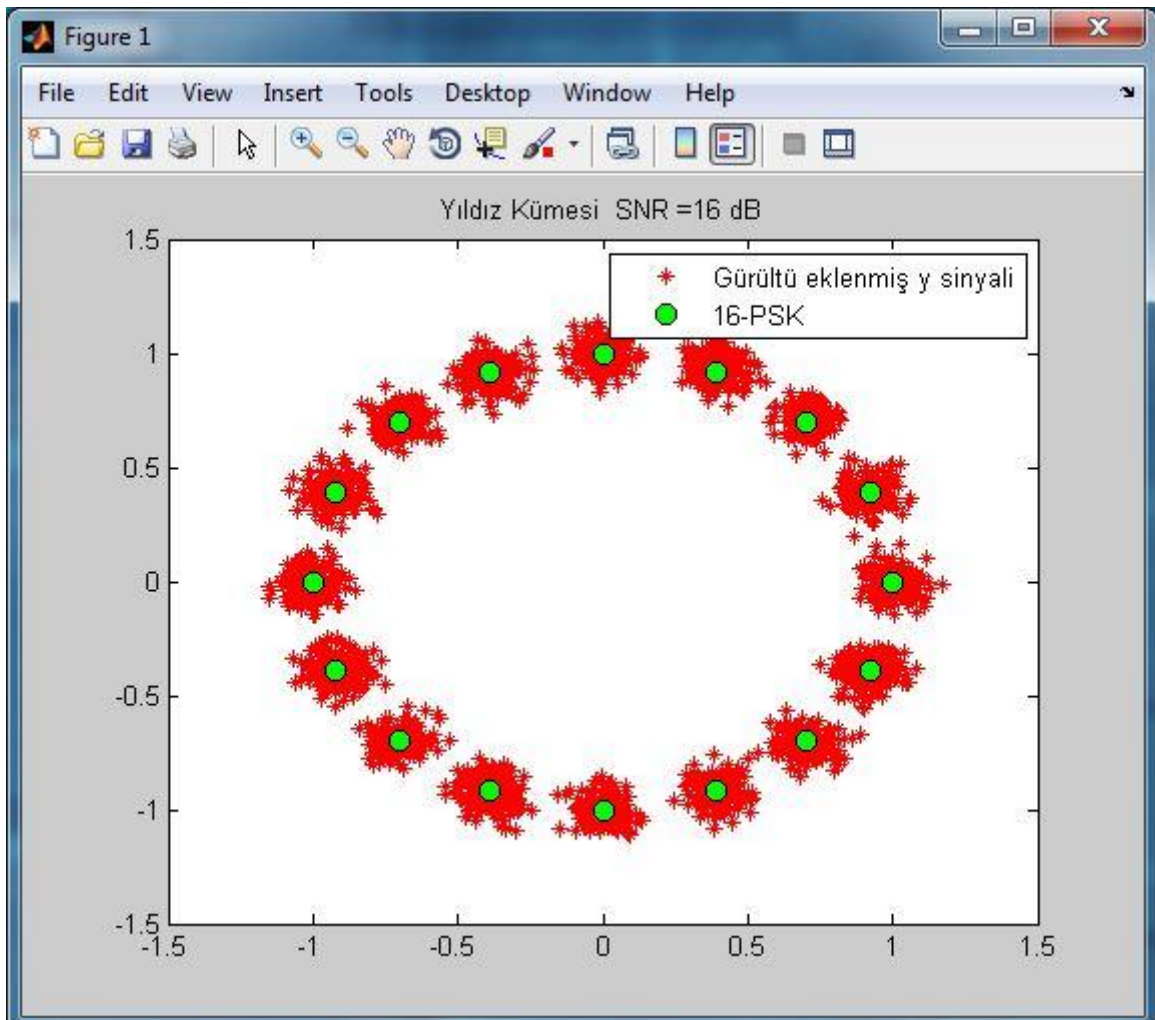



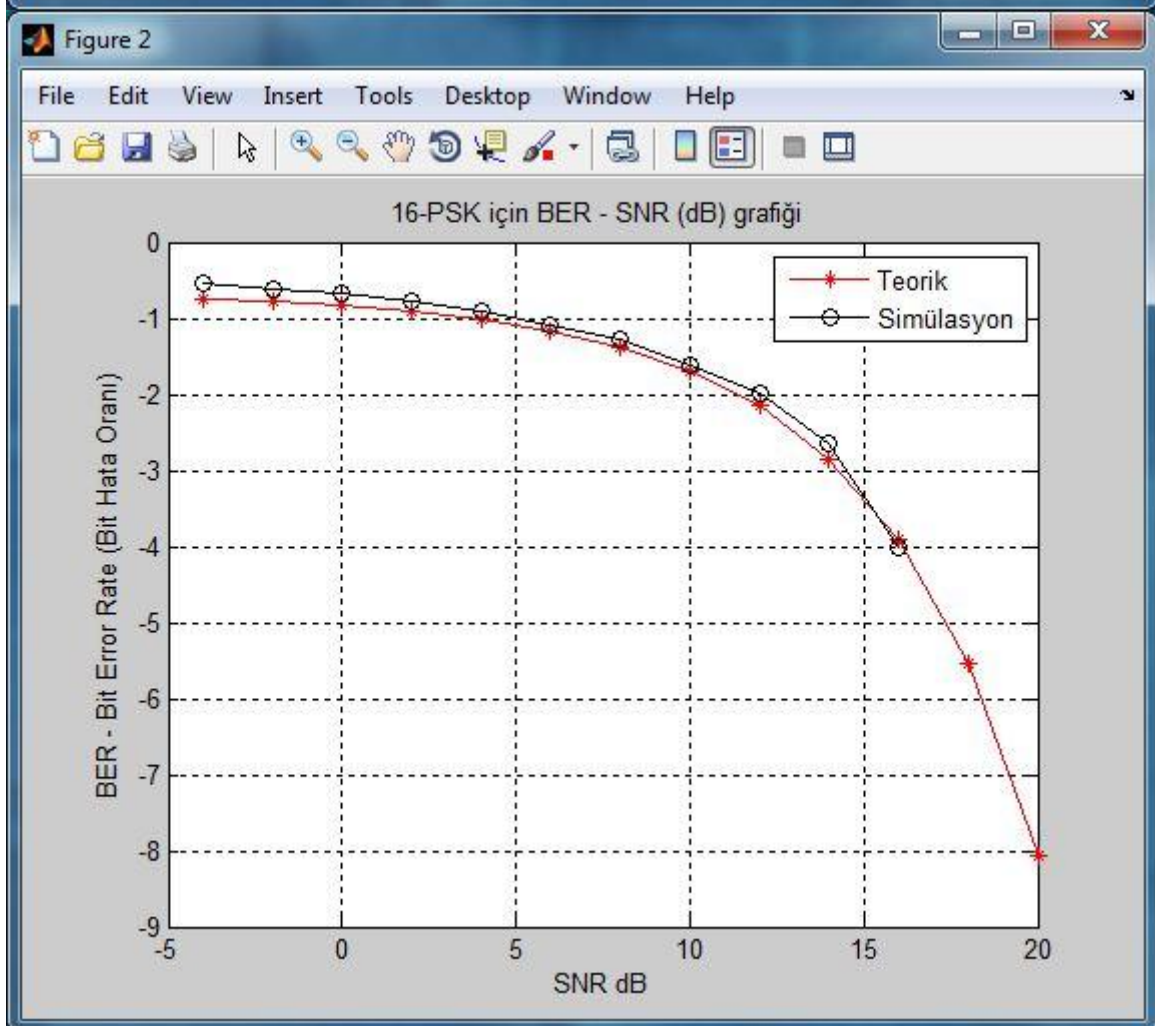
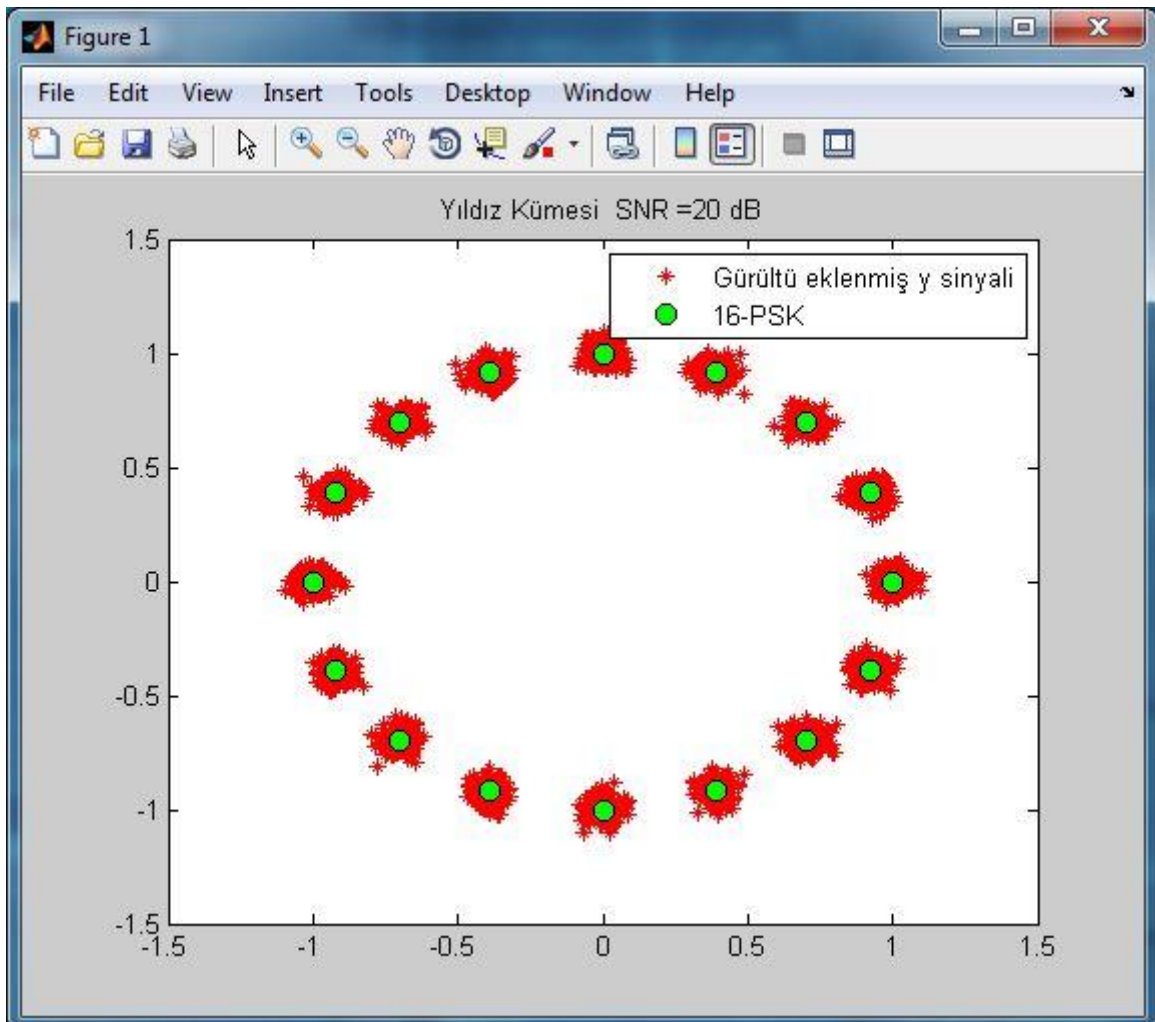












KAYNAKLAR

Communication Systems Fifth Edition – Simon Haykin, Michael Moher

Fundamentals of Communication Systems – John G. Proakis, Masoud Salehi

MATLAB R2011b