



Birzeit University
Faculty of Engineering & Techonology
Department of Electrical & Computer Engineering
ENEE3306

“16-QAM Project”

Student : Mohamad Emad Bornat #1130842

Mohammad Hasan Ghraba

Instructor: Dr.Mohammad Jubran

Date: 5/6/2016

Building The System



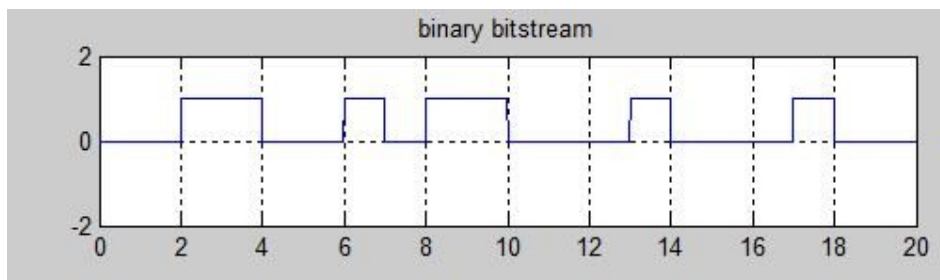
```
n=1000;
nbit=20;
msg=round(rand(nbit,1));
disp(' binary information at transmitter ');
disp(msg);
fprintf('\n\n');
f=.1;
b=msg;
n = length(b); % determine the length of bit stream
t = 0:.01:n; % time axis
x = 1:1:(n+1)*100;
for i = 1:n
    for j = i.1:i+1
        bw(x(i*100:(i+1)*100)) = b(i); % loop
    end
end
bw = bw(100:end); % binary bit stream
```

Bitstream Plot

```
figure(1);
subplot(3,1,1)
plot(t,bw)
title('binary bitstream');
grid on ; axis([0 n -2 +2])
```

SENT BITS:

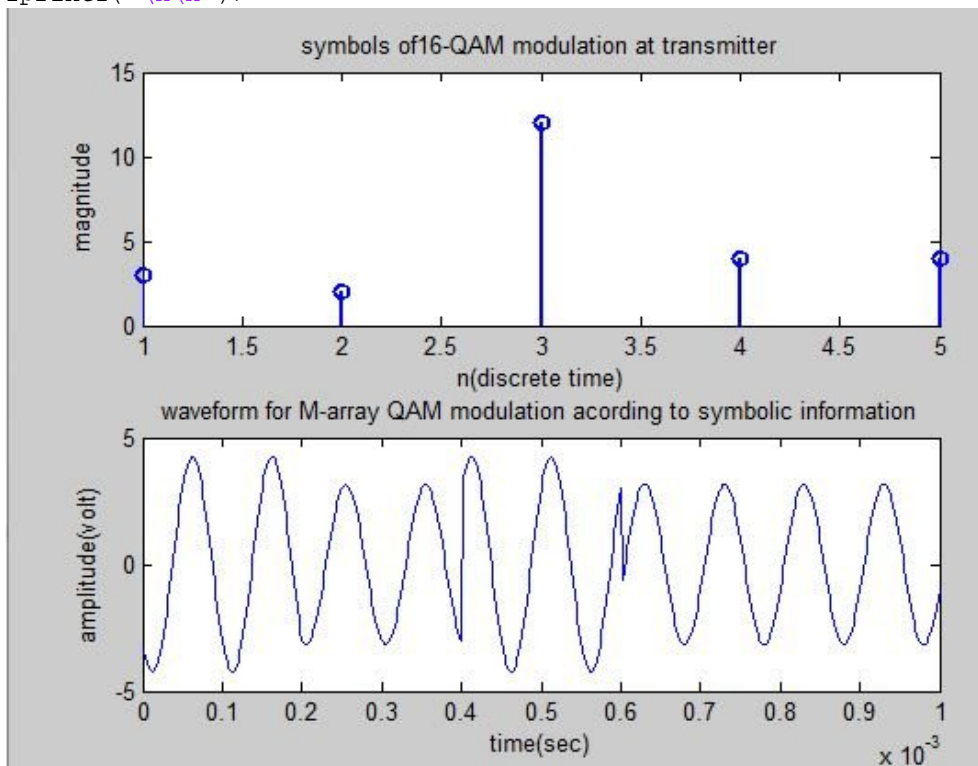
0011001011001000100



Converting Binary information to symbolic



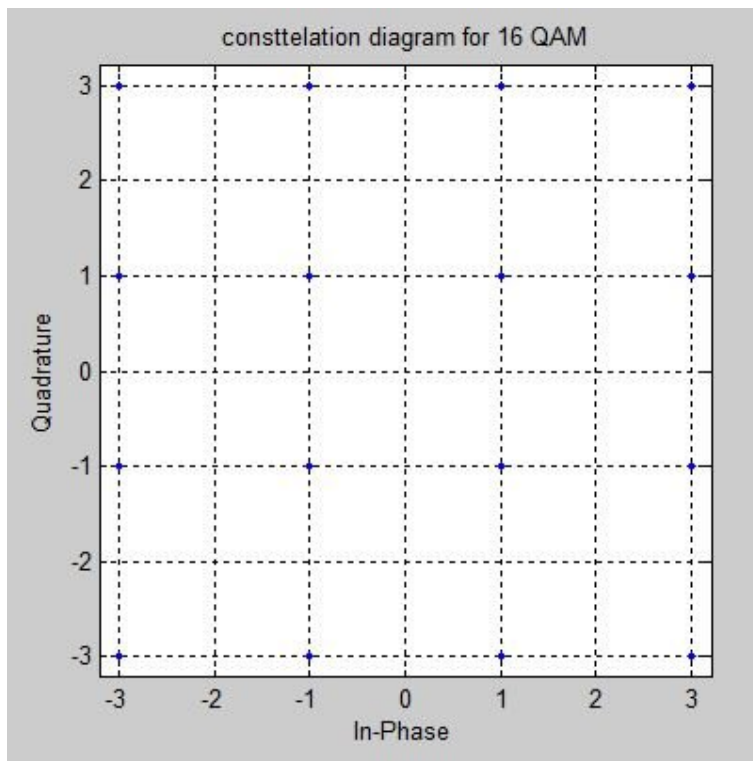
```
M=16;
msg_reshape=reshape(msg,log2(M),nbit/log2(M))';
disp(' information are reshaped for convert symbolic form');
disp(msg_reshape);
fprintf('\n\n');
size(msg_reshape);
for(j=1:1:nbit/log2(M))
    for(i=1:1:log2(M))
        a(j,i)=num2str(msg_reshape(j,i));
    end
end
as=bin2dec(a);
ass=as';
figure(5)
subplot(2,1,1);
stem(ass,'Linewidth',2.0);
title('symbols of16-QAM modulation at transmitter');
xlabel('n(discrete time)');
ylabel(' magnitude');
disp('symbolic form information for M-array QAM ');
disp(ass);
fprintf('\n\n');
```



Mapping to 16QAM Constellation Diagram



```
M=16;  
x1=[0:M-1];  
p=qammod(ass,M)  
sym=0:1:M-1;  
pp=qammod(sym,M);  
scatterplot(pp),grid on;  
title('consttellation diagram for 16 QAM');
```



16QAM Modulation



```
RR=real(p)  
II=imag(p)  
sp=bp*2;  
sr=1/sp;  
f=sr*2;  
t=sp/100:sp/100:sp;  
ss=length(t);  
m=[];  
for(k=1:1:length(RR))  
    yr=RR(k)*cos(2*pi*f*t);  
    yim=II(k)*sin(2*pi*f*t);  
    y=yr+yim;  
    m=[m y];  
end
```

```

tt=sp/100:sp/100:sp*length(RR);
figure(5);
subplot(2,1,2);
plot(tt,m);
title('waveform for M-array QAM modulation according to symbolic
information');
xlabel('time(sec)');
ylabel('amplitude(volt)');

```

16QAM Demodulation



```

m1=[];
m2=[];
for n=ss:ss:length(m)
    t=sp/100:sp/100:sp;
    y1=cos(2*pi*f*t);
    y2=sin(2*pi*f*t);
    mm1=y1.*m((n-(ss-1)):n);
    mm2=y2.*m((n-(ss-1)):n);
    z1=trapz(t,mm1);
    z2=trapz(t,mm2);
    zz1=round(2*z1/sp);
    zz2=round(2*z2/sp);
    m1=[m1 zz1];
    m2=[m2 zz2];
end

```

De-Mapping & Representation of received signal



```

clear i;
clear j;
for (k=1:1:length(m1))
    gt(k)=m1(k)+j*m2(k);
end
gt

ax=qamdemod(gt,M);
figure(3);
subplot(2,1,1);
stem(ax,'linewidth',2);
title(' 16QAM demodulation symbols ');
xlabel('n(discrete time)');
ylabel(' magnitude');

disp('re-obtain symbol after M-array QAM demodulation ');
disp(ax);
fprintf('\n\n');

bi_in=dec2bin(ax);
[row col]=size(bi_in);

```

```

p=1;
for(i=1:1:row)
    for(j=1:1:col)
        re_bi_in(p)=str2num(bi_in(i,j));
        p=p+1;
    end
end
disp('re-obtain binary information after M-array QAM demodulation');
disp(re_bi_in)
fprintf('\n\n');

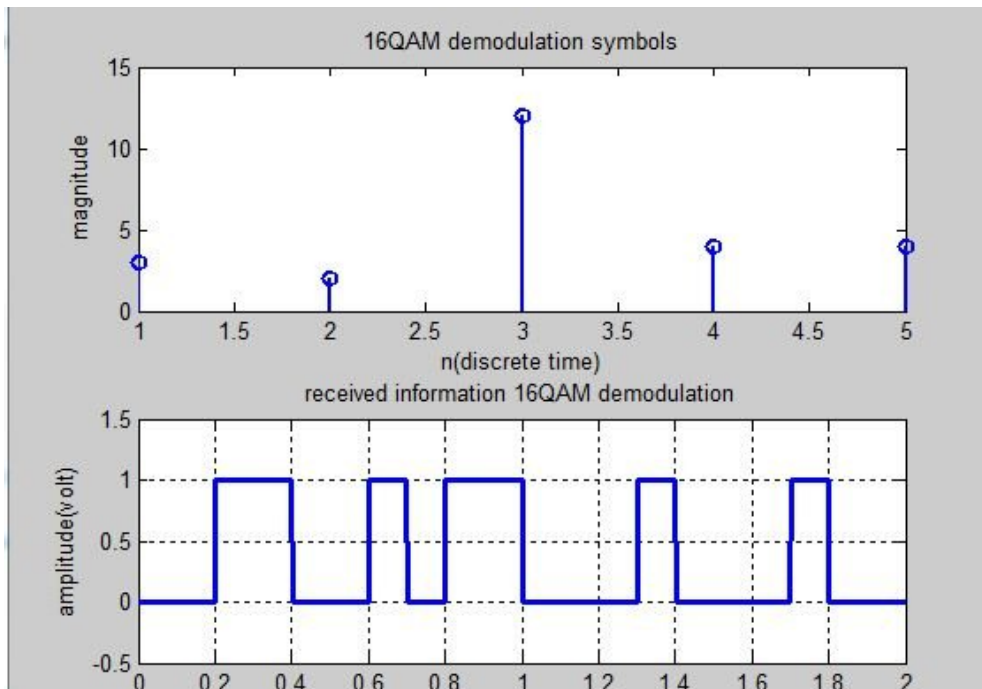
x=re_bi_in;
bp=.0001;
bit=[];
for n=1:1:length(x)
    if x(n)==1;
        se=ones(1,100);
    else x(n)==0;
        se=zeros(1,100);
    end
    bit=[bit se];
end

end
t1=bp/100:bp/100:100*length(x)*(bp/100);
figure(3)
subplot(2,1,2);
plot(t1,bit,'lineWidth',2.5);grid on;
axis([ 0 bp*length(x) -.5 1.5]);
ylabel('amplitude(volt)');
xlabel(' time(sec)');
title('received information 16QAM demodulation');

```

RECEIVED BITS:

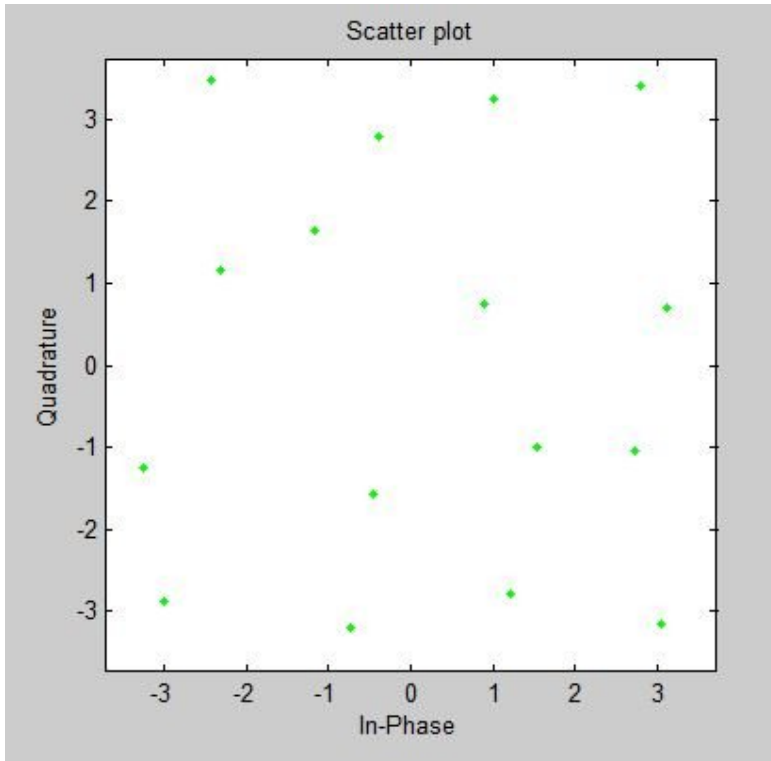
0011001011001000100



Adding AWGN



```
numSamplesPerSymbol = 1; % Oversampling factor
EbNo = 10;
snr = EbNo + 10*log10(k) - 10*log10(numSamplesPerSymbol);
receivedSignal = awgn(pp,snr,'measured');
sPlotFig = scatterplot(receivedSignal,1,0,'g');
```



Computing Power & Energy



```
fb=10000;
for snr=0:10
eb = sum(msg.^2)/(length(msg)*fb)
disp(eb);
ebn0= 10^(snr/10)
disp(ebn0)
No=eb/ebn0
disp(No)
end
```

ANSWER for some values :

```
eb =
    6.0000e-05
    6.0000e-05

ebn0 =
    1.5849
    1.5849

No =
    3.7857e-05
    3.7857e-05

eb =
    6.0000e-05
    6.0000e-05

ebn0 =
    1.9953
    1.9953

No =
    3.0071e-05
    3.0071e-05
```

 New to MATLAB? Watch this [Video](#), see [Examples](#), or read [Getting Started](#).

```
end
```

```
eb =
```

```
6.0000e-05
```

```
6.0000e-05
```

```
ebn0 =
```

```
1
```

```
1
```

```
No =
```

```
6.0000e-05
```

```
6.0000e-05
```

```
eb =
```

```
6.0000e-05
```

```
6.0000e-05
```

```
ebn0 =
```

```
1.2589
```

```
1.2589
```

```
No =
```

```
4.7660e-05
```

```
4.7660e-05
```

Computing Probability of Error per Symbol & per Bit



$$PES = (3/2) * \text{erfc}(\sqrt{\text{snr}/10})$$

$$PEB = PES/k$$

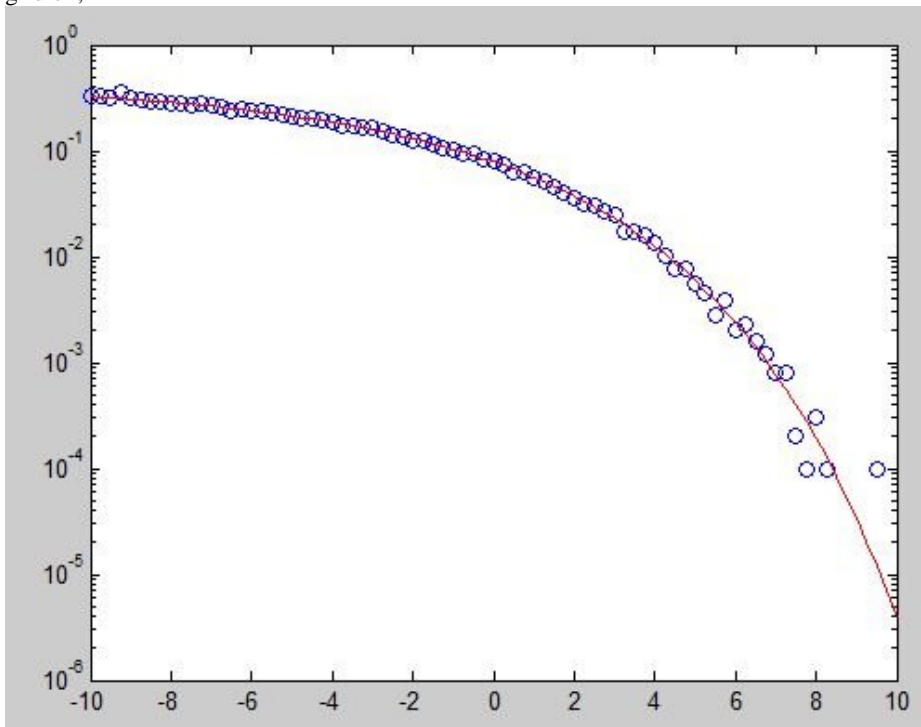
Answer:

```
PES =
Columns 1 through 16
    1.3313    1.3109    1.2880    1.2625    1.2340    1.2022    1.1667    1.1273    1.0836    1.0353    0.9821    0.9237    0.8601    0.7914    0.7177    0.6397
Columns 17 through 21
    0.5583    0.4751    0.3919    0.3113    0.2359
>> PEB=PES/k
PEB =
Columns 1 through 16
    0.3328    0.3277    0.3220    0.3156    0.3085    0.3005    0.2917    0.2818    0.2709    0.2588    0.2455    0.2309    0.2150    0.1978    0.1794    0.1599
Columns 17 through 21
    0.1396    0.1188    0.0980    0.0778    0.0590
```

Plotting Probability of Error VS SNR



```
Pb(k)=mean(abs((pp)-(receivedSignal)));
SNR_db=-10:0.25:10;
SNR=10.^(SNR_db/10);
P_theory=(0.5)*erfc(sqrt(2.*SNR)/sqrt(2));
semilogy(SNR_db,Pb,'o',SNR_db,P_theory,'r-')
title('SNR Vs Probability of Error');
xlabel('SNR (dB)');
ylabel('Probability of Error');
legend('practical curve','theoretical curve');
grid on;
```





Quadrature Amplitude Modulation or QAM is a form of modulation which is widely used for modulating data signals onto a carrier used for radio communications. It is widely used because it offers advantages over other forms of data modulation such as PSK, although many forms of data modulation operate alongside each other.

Quadrature Amplitude Modulation, QAM is a signal in which two carriers shifted in phase by 90 degrees are modulated and the resultant output consists of both amplitude and phase variations. In view of the fact that both amplitude and phase variations are present it may also be considered as a mixture of amplitude and phase modulation.

The advantage of using QAM is that it is a higher order form of modulation and as a result it is able to carry more bits of information per symbol. By selecting a higher order format of QAM, the data rate of a link can be increased.

Although QAM appears to increase the efficiency of transmission for radio communications systems by utilizing both amplitude and phase variations, it has a number of drawbacks. The first is that it is more susceptible to noise because the states are closer together so that a lower level of noise is needed to move the signal to a different decision point. Receivers for use with phase or frequency modulation are both able to use limiting amplifiers that are able to remove any amplitude noise and thereby improve the noise reliance. This is not the case with QAM.

The second limitation is also associated with the amplitude component of the signal. When a phase or frequency modulated signal is amplified in a radio transmitter, there is no need to use linear amplifiers, whereas when using QAM that contains an amplitude component, linearity must be maintained. Unfortunately linear amplifiers are less efficient and consume more power, and this makes them less attractive for mobile applications.

While higher order modulation rates are able to offer much faster data rates and higher levels of spectral efficiency for the radio communications system, this comes at a price. The higher order modulation schemes are considerably less resilient to noise and interference.