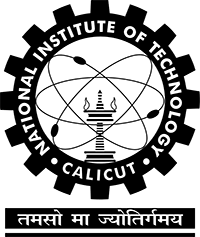
**NATIONAL INSTITUTE OF TECHNOLOGY CALICUT**



**Department of E.C.E**

**EC4092: DIGITAL COMMUNICATION LAB**

**EXP-8 : 8 QAM WITH DIFFERENT CONSTELLATIONS**

**(SOFTWARE)**

**SUBMITTED BY**

Sowmya Bitra-B150821EC

Tejaswi Manukonda-B150571EC

V Prasanna-B150738EC

Vundavalli Aswini-B150519EC

Expt-8

01-10-2018

**Aim:**

Simulate a baseband 8QAM communication system in the presence of AWGN and compare BER performance for three different constellations (rectangular, circular1 and circular2)

* Implementation of 8QAM modulation without using built-in MATLAB functions.
* Constructing three different constellation diagrams.
* Plotting of ideal and practical BER curves for different constellations.

**Theory:**

**QAM:** Quadrature amplitude modulation (QAM) is both an analog and a digital [modulation](https://en.wikipedia.org/wiki/Modulation) scheme. It conveys two analog message signals, or two digital [bit streams](https://en.wikipedia.org/wiki/Bit_stream), by changing (*modulating*) the [amplitudes](https://en.wikipedia.org/wiki/Amplitude) of two [carrier waves](https://en.wikipedia.org/wiki/Carrier_wave), using the [amplitude-shift keying](https://en.wikipedia.org/wiki/Amplitude-shift_keying) (ASK) digital modulation scheme or [amplitude modulation](https://en.wikipedia.org/wiki/Amplitude_modulation) (AM) analog modulation scheme. The two carrier waves of the same frequency, usually [sinusoids](https://en.wikipedia.org/wiki/Sine_wave), are [out of phase](https://en.wikipedia.org/wiki/Out_of_phase) with each other by 90° and are thus called [quadrature](https://en.wikipedia.org/wiki/Quadrature_phase" \o "Quadrature phase) carriers or quadrature components — hence the name of the scheme. The modulated waves are summed, and the final waveform is a combination of both [phase-shift keying](https://en.wikipedia.org/wiki/Phase-shift_keying) (PSK) and [amplitude-shift keying](https://en.wikipedia.org/wiki/Amplitude-shift_keying) (ASK), or, in the analog case, of phase modulation (PM) and amplitude modulation. In the digital QAM case, a finite number of at least two phases and at least two amplitudes are used.

**Signal Constellation:**

Signal constellation is a graphical representation of the complex envelope of each possible signal .

The x-axis represents the in-phase component and the y-axis represents the quadrature component of the complex envelope

The distance between signals on a constellation diagram relates to how different the modulation waveforms are and how well a receiver can differentiate between them when random noise is present.

**BIT ERROR AND SYMBOL ERROR:**

In [digital transmission](https://en.wikipedia.org/wiki/Digital_transmission), the number of bit errors is the number of received [bits](https://en.wikipedia.org/wiki/Bit) of a [data stream](https://en.wikipedia.org/wiki/Data_stream) over a [communication channel](https://en.wikipedia.org/wiki/Communication_channel) that have been altered due to [noise](https://en.wikipedia.org/wiki/Noise_(telecommunications)), [interference](https://en.wikipedia.org/wiki/Interference_(communication)), [distortion](https://en.wikipedia.org/wiki/Distortion) or [bit synchronization](https://en.wikipedia.org/wiki/Bit_synchronization) errors.

The bit error rate (BER) is the number of bit errors per unit time. The bit error ratio (also BER) is the number of bit errors divided by the total number of transferred bits during a studied time interval. Bit error ratio is a unit less performance measure, often expressed as a [percentage](https://en.wikipedia.org/wiki/Percentage). The bit error probability *pe* is the [expectation value](https://en.wikipedia.org/wiki/Expectation_value) of the bit error ratio.

**RECTANGULAR QAM**

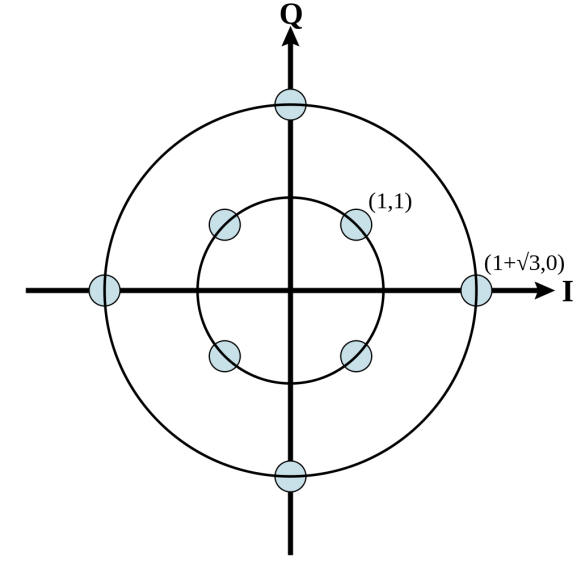
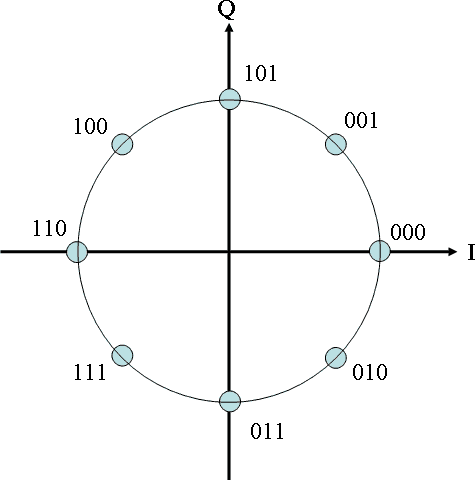
Rectangular QAM constellations are, in general, sub-optimal in the sense that they do not maximally space the constellation points for a given energy. However, they have the considerable advantage that they may be easily transmitted as two [pulse amplitude modulation](https://en.wikipedia.org/wiki/Pulse_amplitude_modulation) (PAM) signals on quadrature carriers, and can be easily demodulated. The non-square constellations, dealt with below, achieve marginally better bit-error rate (BER) but are harder to modulate and demodulate.

{\displaystyle \,P\_{s}=1-\left(1-P\_{sc}\right)^{2}}

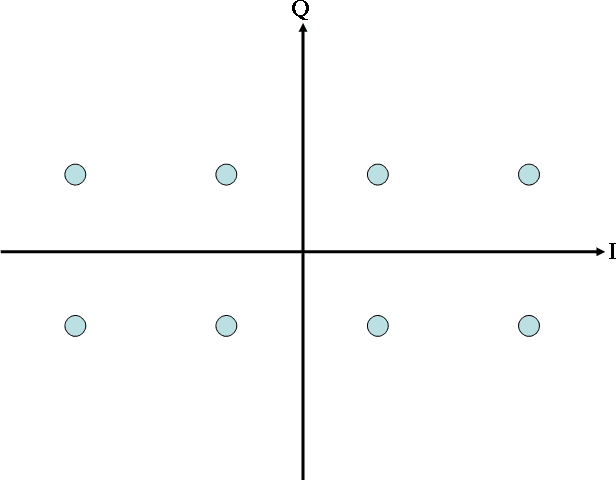
**CIRCULAR QAM**

The circular 8-QAM constellation is known to be the optimal 8-QAM constellation in the sense of requiring the least mean power for a given minimum Euclidean distance.

**Fig1: Circular1 Constellation Fig2:Circular2 Constellation**



**Fig3:Rectangular constellation**



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.

**SIMULATION METHODOLOGY**

1. Define constellation points for rectangular, circular1, circular2 constellations and normalize it with average symbol energy and plot the same.
2. Map QAM constellations through index values and add AWGN before transmitting.
3. Find the minimum Euclidean distance in all three cases to demodulate the transmitted signal.
4. Find BER theoretically and that of rectangular, circulra1 and circular2 constellations.
5. Compare BER plots of all three constellations.

**MATLAB CODE**

clear all;

close all;

clc;

snrdB = [0:2:18]; *%SNR values*

snr = 10.^(snrdB/10);

ber\_r=zeros(1, length(snrdB));

ber\_c1=zeros(1, length(snrdB));

ber\_c2=zeros(1, length(snrdB));

ber\_rth = zeros(1, length(snrdB));

ber\_c1th=zeros(1, length(snrdB));

ber\_c2th=zeros(1, length(snrdB));

N=30000; *%total no.of bits*

iter\_max = 50;

rec = [-3+j -1+j 1+j 3+j -3-j -1-j 1-j 3-j]; %*points on rectangular constellation.*

cir1=[1 -1 j -j (1+j)/sqrt(2) (-1+j)/sqrt(2) (1-j)/sqrt(2) (-1-j)/sqrt(2)]; %*points on circular constellation1.*

cir2 = [1+j 1-j -1+j -1-j 1+sqrt(3) -(1+sqrt(3)) (1+sqrt(3))\*j -(1+sqrt(3))\*j]; %*points on circular constellation2.*

rec\_bit = [0 0 0; 0 1 0; 1 0 0; 1 1 0; 0 0 1; 0 1 1; 1 0 1; 1 1 1];

cir1\_bit=[1 1 1; 0 0 1; 0 1 0; 1 0 0; 1 1 0; 0 1 1; 1 0 1; 0 0 0];

cir2\_bit=[0 0 0; 0 1 1; 0 0 1; 0 1 0; 1 0 0; 1 1 0; 1 0 1; 1 1 1];

for i=1:1:length(snrdB)

error = zeros(1, iter\_max);

for k=1:1:iter\_max

signal = round(rand(1,N));

const\_rec = zeros(1, N/3);

const\_cir1 = zeros(1, N/3);

const\_cir2 = zeros(1, N/3);

for m=1:3:length(signal)

symbol=signal(m:m+2);

if(symbol == [0 0 0])

const\_rec((m-1)/3+1)=-3+1i;

const\_cir1((m-1)/3+1)=(-1-1i)/sqrt(2);

const\_cir2((m-1)/3+1)=1+1i;

elseif(symbol==[0 0 1])

const\_rec((m-1)/3+1)=-3-1i;

const\_cir1((m-1)/3+1)=-1;

const\_cir2((m-1)/3+1)=-1+1i;

elseif(symbol==[0 1 0])

const\_rec((m-1)/3+1)=-1+1i;

const\_cir1((m-1)/3+1)=1i;

const\_cir2((m-1)/3+1)=-1-1i;

elseif(symbol==[0 1 1])

const\_rec((m-1)/3+1)=-1-1i;

const\_cir1((m-1)/3+1)=(-1+1i)/sqrt(2);

const\_cir2((m-1)/3+1)=1-1i;

elseif(symbol==[1 0 0])

const\_rec((m-1)/3+1)=1+1i;

const\_cir1((m-1)/3+1)=-1i;

const\_cir2((m-1)/3+1)=1+sqrt(3);

elseif(symbol==[1 0 1])

const\_rec((m-1)/3+1)=1-1i;

const\_cir1((m-1)/3+1)=(1-1i)/(sqrt(2));

const\_cir2((m-1)/3+1)=(1+sqrt(3))\*1i;

elseif(symbol==[1 1 0])

const\_rec((m-1)/3+1)=3+1i;

const\_cir1((m-1)/3+1)=(1+1i)/sqrt(2);

const\_cir2((m-1)/3+1)=-(1+sqrt(3));

elseif(symbol==[1 1 1])

const\_rec((m-1)/3+1)=3-1i;

const\_cir1((m-1)/3+1)=1;

const\_cir2((m-1)/3+1)=-(1+sqrt(3))\*1i;

end

end

received\_rec=awgn(const\_rec, snrdB(i)+4.771, 'measured');

received\_cir1=awgn(const\_cir1, snrdB(i)+4.771, 'measured');

received\_cir2=awgn(const\_cir2, snrdB(i)+4.771, 'measured');

for l=1:1:length(received\_rec)

dist\_rec = zeros(1, 8);

dist\_cir1 = zeros(1, 8);

dist\_cir2 = zeros(1, 8);

for ll=1:1:8

dist\_rec(ll)=abs(received\_rec(l)-rec(ll));

dist\_cir1(ll)=abs(received\_cir1(l)-cir1(ll));

dist\_cir2(ll)=abs(received\_cir2(l)-cir2(ll));

end

[v, minrec] = min(dist\_rec);

[v, mincir1] = min(dist\_cir1);

[v, mincir2] = min(dist\_cir2);

decoded\_rec(l, :) = rec\_bit(minrec, :);

decoded\_cir1(l, :) = cir1\_bit(mincir1, :);

decoded\_cir2(l, :) = cir2\_bit(mincir2, :);

end

temp = decoded\_rec';

rec\_stream = temp(:);

temp = decoded\_cir1';

cir1\_stream = temp(:);

temp = decoded\_cir2';

cir2\_stream = temp(:);

error\_rec = abs(rec\_stream'-signal);

errorno\_rec(k) = sum(error\_rec);

ber\_r(i) = ber\_r(i) + (errorno\_rec(k)/iter\_max);

error\_cir1 = abs(cir1\_stream'-signal);

errorno\_cir1(k) = sum(error\_cir1);

ber\_c1(i) = ber\_c1(i) + (errorno\_cir1(k)/iter\_max);

error\_cir2 = abs(cir2\_stream'-signal);

errorno\_cir2(k) = sum(error\_cir2);

ber\_c2(i) = ber\_c2(i) + (errorno\_cir2(k)/iter\_max);

end

end

ber\_r = ber\_r/N;

ber\_c1 = ber\_c1/N;

ber\_c2 = ber\_c2/N;

ber\_rth = 2\*erfc(sqrt((9/14)\*snr));

ber\_c1th = erfc(sqrt(3\*snr/14))\*sin(pi/8);

ber\_c2th= 3.5\*erfc(sqrt(snr));

scatterplot(const\_rec);

xlabel('In-phase');

ylabel('Quadrature');

title('Constellation diagram of rectangular 8-QAM(Theoretical)');

scatterplot(const\_cir1);

xlabel('In-phase');

ylabel('Quadrature');

title('Constellation diagram of Circular-1 8-QAM (Theoretical)');

scatterplot(const\_cir2);

xlabel('In-phase');

ylabel('Quadrature');

title('Constellation diagram of Circular-2 8-QAM (Theoretical)');

scatterplot(received\_rec);

xlabel('In-phase');

ylabel('Quadrature');

title('Constellation diagram of rectangular 8-QAM at decoder');

scatterplot(received\_cir1);

xlabel('In-phase');

ylabel('Quadrature');

title('Constellation diagram of received Circular-1 8-QAM at decoder');

scatterplot(received\_cir2);

xlabel('In-phase');

ylabel('Quadrature');

title('Constellation diagram of received Circular-2 8-QAM at decoder');

figure;

semilogy(snrdB, ber\_r, 'k\*-', snrdB, ber\_c1, 'b-', snrdB, ber\_c2, 'c+-');

xlabel('----> Eb/No(dB)');

ylabel('----> Bit Error Rate');

legend('Rectangular constellation', 'Circular constellation 1', 'Circular constellation 2');

title('BER plots - Practically observed');

figure;

semilogy(snrdB, ber\_rth, 'k\*-', snrdB, ber\_c1th, 'b-', snrdB, ber\_c2th, 'c+-');

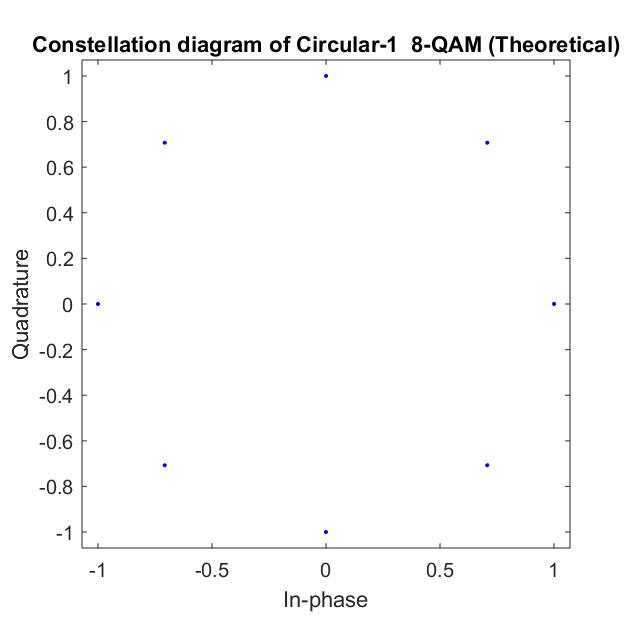
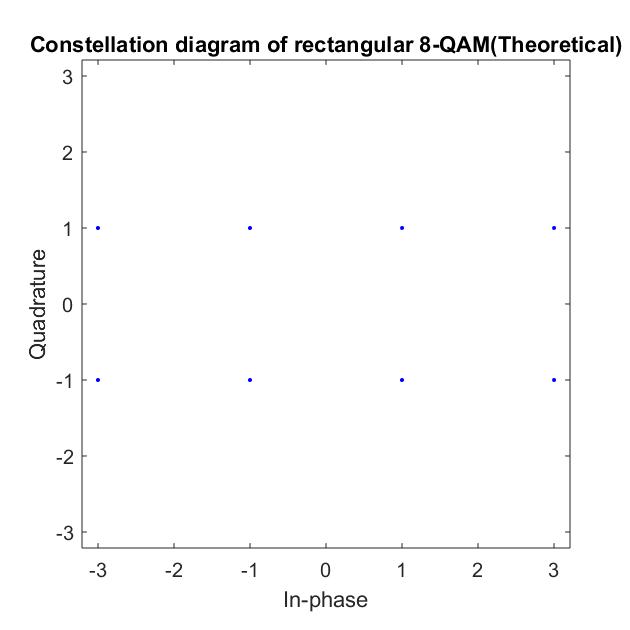
xlabel('----> Eb/No(dB)');

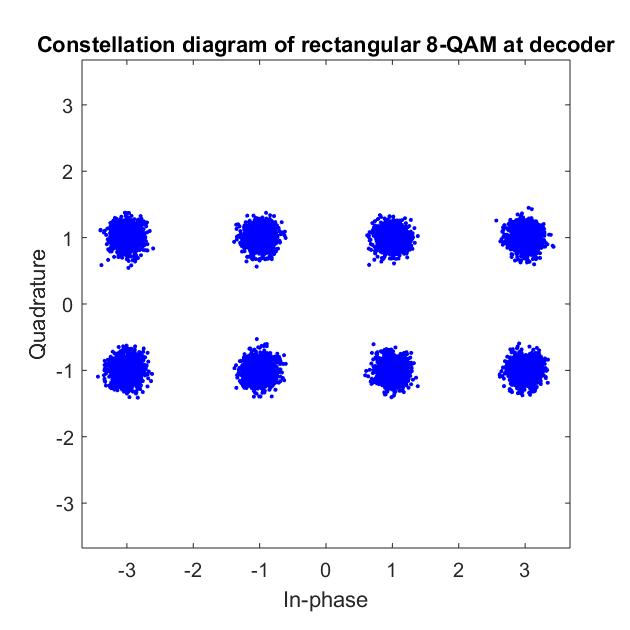
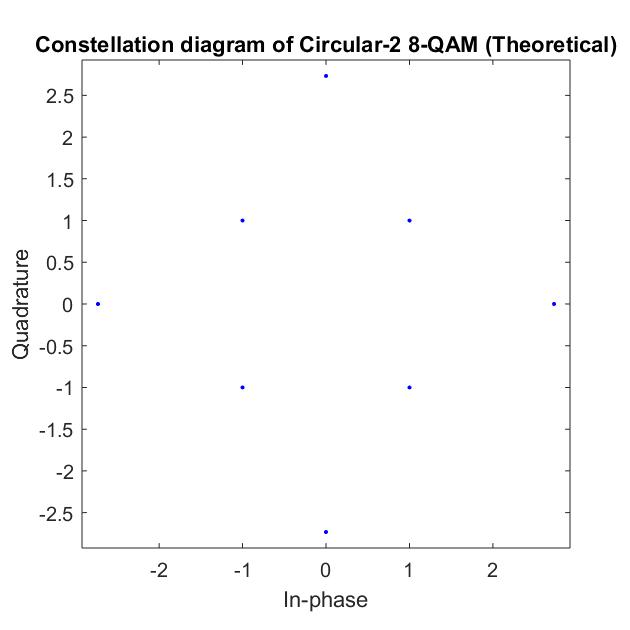
ylabel('----> Bit Error Rate');

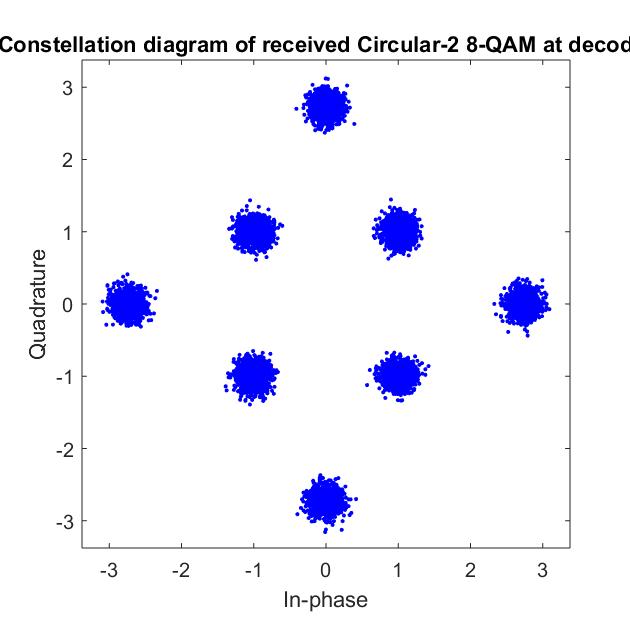
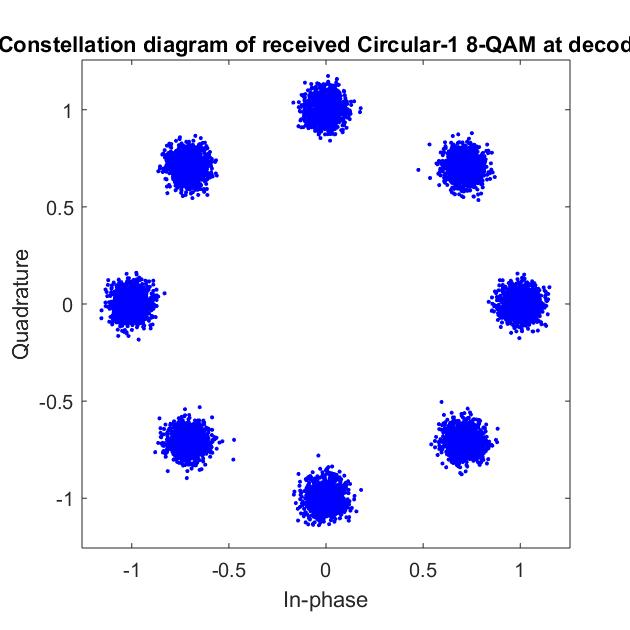
legend('Rectangular constellation', 'Circular constellation 1', 'Circular constellation 2');

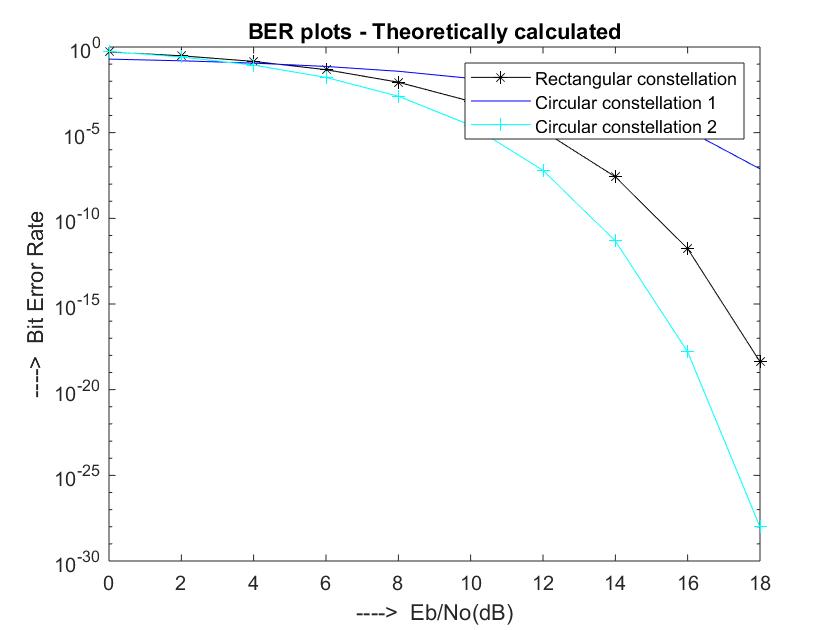
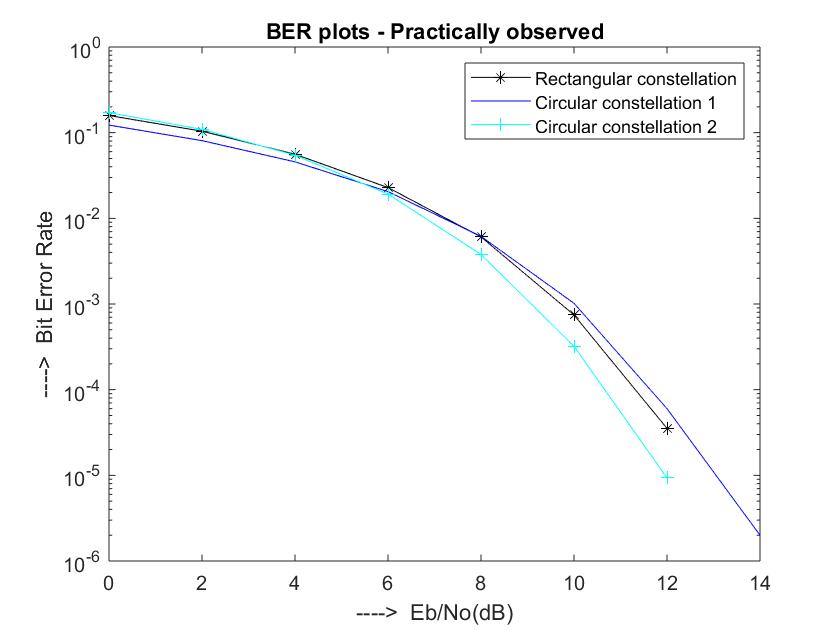
title('BER plots - Theoretically calculated');

**OBSERVATIONS AND OUTPUTS:**

****

****

****

****

**INFERENCE**

1. Rectangular constellation points are selected at distances √2 and √10 from the origin and plotted the constellation.
2. Circular1 constellation points are selected on a unit circle and plotted.
3. Circular2 constellation points on circles of radius √2 units and 1+√3 units are selected and plotted.
4. The constellation diagram of the received signal is spread due the addition of noise.
5. In BER plot the bit error rate came down when the SNR ratio Eb/No was increased.
6. Circular 2 constellation was found to be better when compared to other constellations considering the low BER for higher SNR values.
7. In practical plot, rectangular constellation gave similar results to circular1 constellation plot.

**RESULT**

8QAM modulation technique was simulated in MATLAB and their BER plots and different constellations were studied.