CT111- Introduction to Communication Systems

SUBMISSION: Lab-6

NAME: SHAH NILAY JAYESH

ID:201901026

Topic: Modulation Schemes

Index

❖ Honor Code:

I declare that,

The work that I have done is my own work. I have not copied the work (the code, the results etc.) that someone else has done-> concepts and insights I will be describing is my own. I make this pledge truthfully. I know that violation of this solemn pledge can carry grave consequences.

I have not taken any help from my classmates.

Signature:

N.J. Shah

❖ Simplified Model implementation:

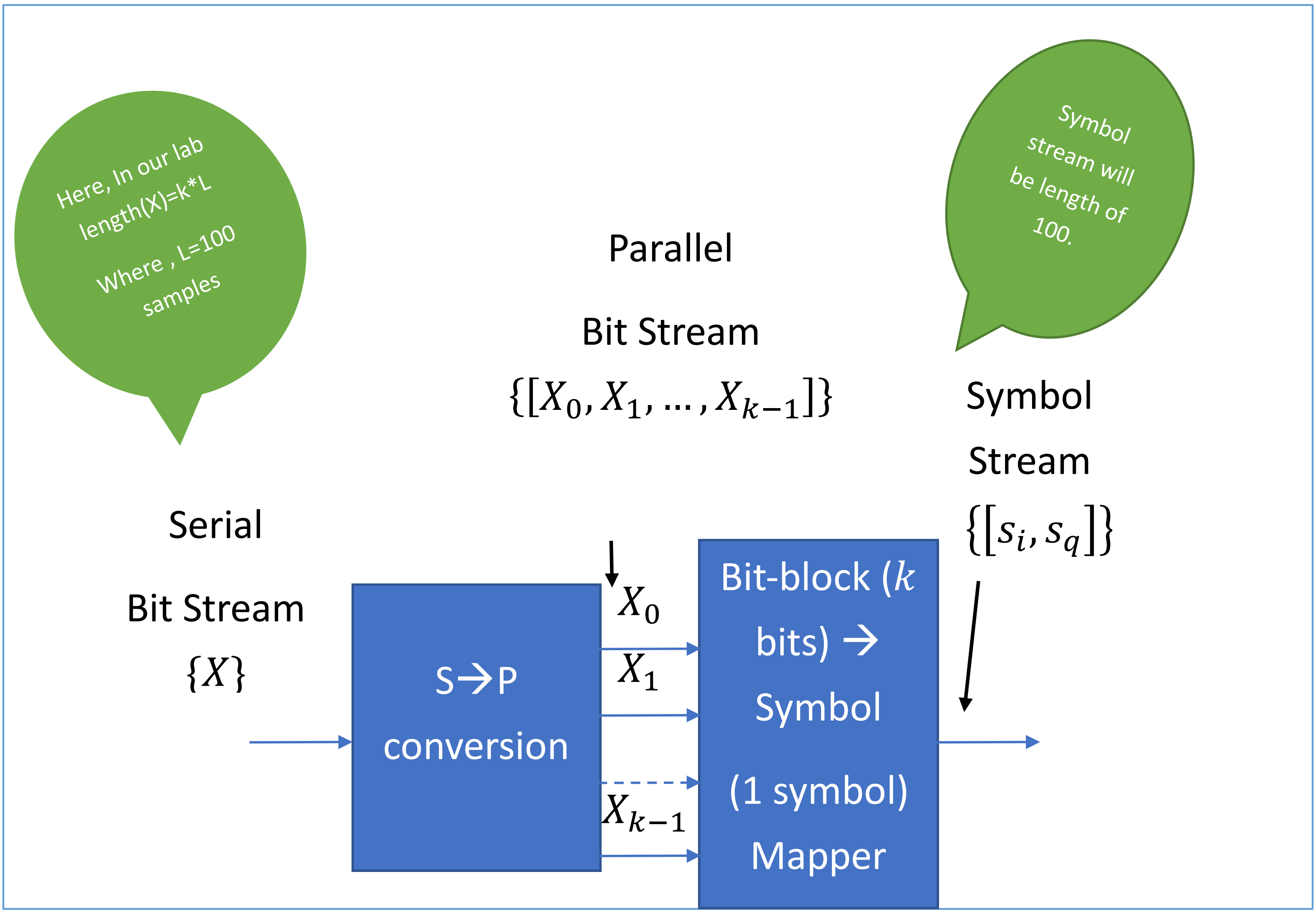
❖ Here, I have implemented the modem for the following higher-order M = 2^k modulation schemes:

1. QPSK (k=2)
2. 8-PSK(k=3)
3. 16-APSK(k=4)
4. 32-QAM(k=5)

- **The basic idea is designated as diagram:**

At the modulator,

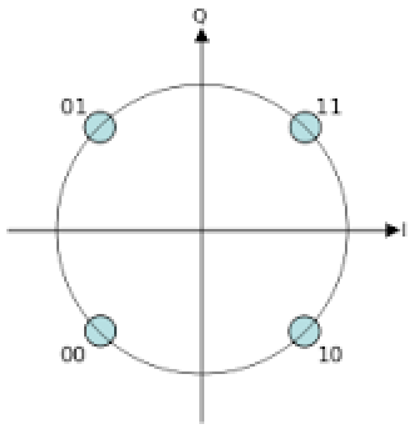
- A serial bit stream is converted to a parallel bit stream
- The parallel branches together form a block of k sequential bits.
- Each such block is transformed to one of $M = 2^k$ symbols.

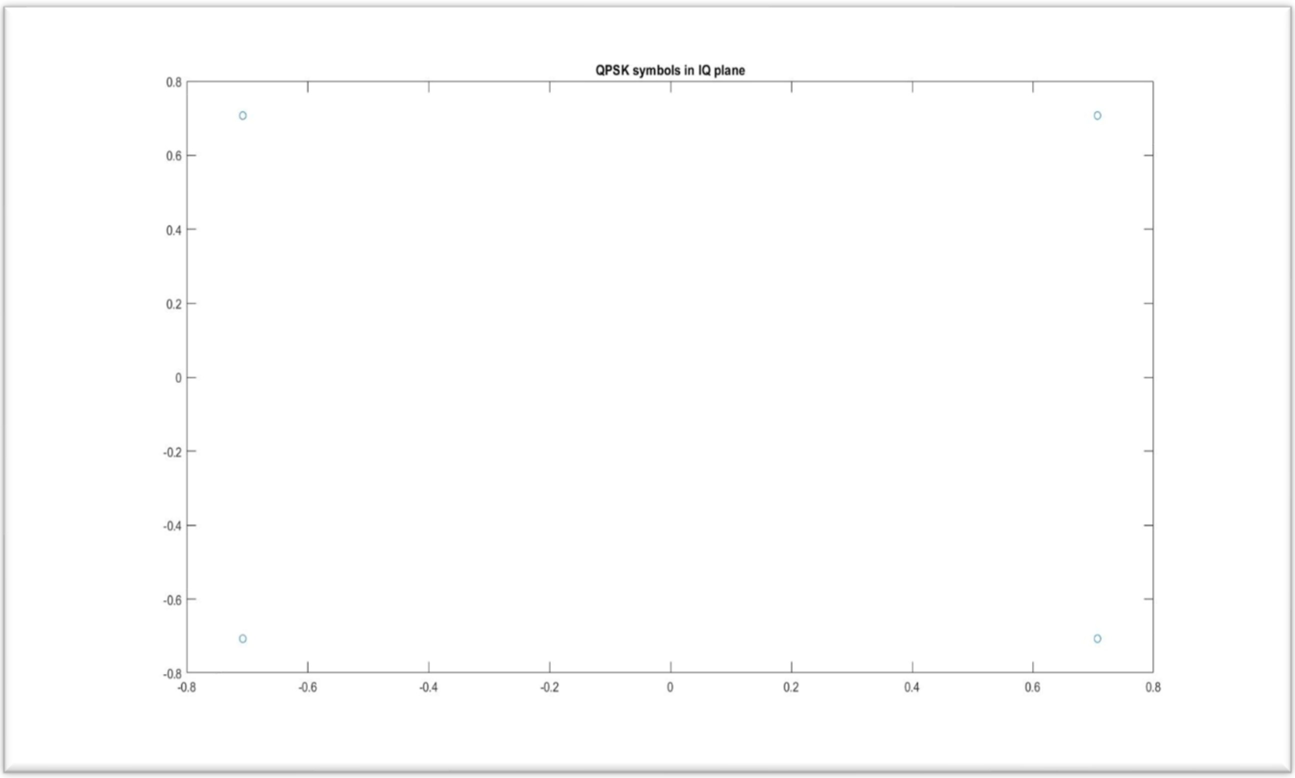

Here, In our lab length(X)=k*L Where , L=100 samples
Symbol stream will be length of 100.
Parallel
Bit Stream
$\{[X_0, X_1, \dots, X_{k-1}]\}$
Symbol
Stream
$\{[s_i, s_q]\}$
Serial
Bit Stream
$\{X\}$
S→P conversion
X_0
X_1
X_{k-1}
Bit-block (k bits) → Symbol
(1 symbol) Mapper

1. Implementation of QPSK Modem:

- For QPSK scheme, first we need to define M=2^k symbols where here k=2. Therefore M=4 symbols. We can designate these symbols in IQ plane shown below:

QPSK: $(\mathrm{M}, \mathrm{k} = 4{,}2)$





- I have set a=1 for these symbols.

X_0, X_1

QPSK: An Example Bits to Symbol Mapping

$$X_0, X_1 = 1,\ 1 \rightarrow s_i,\ s_q = +\frac{1}{\sqrt{2}},\ +\frac{1}{\sqrt{2}}\ \text{volts}$$

$$X_0, X_1 = 1,\ 0 \rightarrow s_i,\ s_q = +\frac{1}{\sqrt{2}},\ -\frac{1}{\sqrt{2}}\ \text{volts}$$

$$X_0, X_1 = 0,\ 1 \rightarrow s_i,\ s_q = -\frac{1}{\sqrt{2}},\ +\frac{1}{\sqrt{2}}\ \text{volts}$$

$s = s_i, s_q$

* **Derivation of Es in QPSK:**

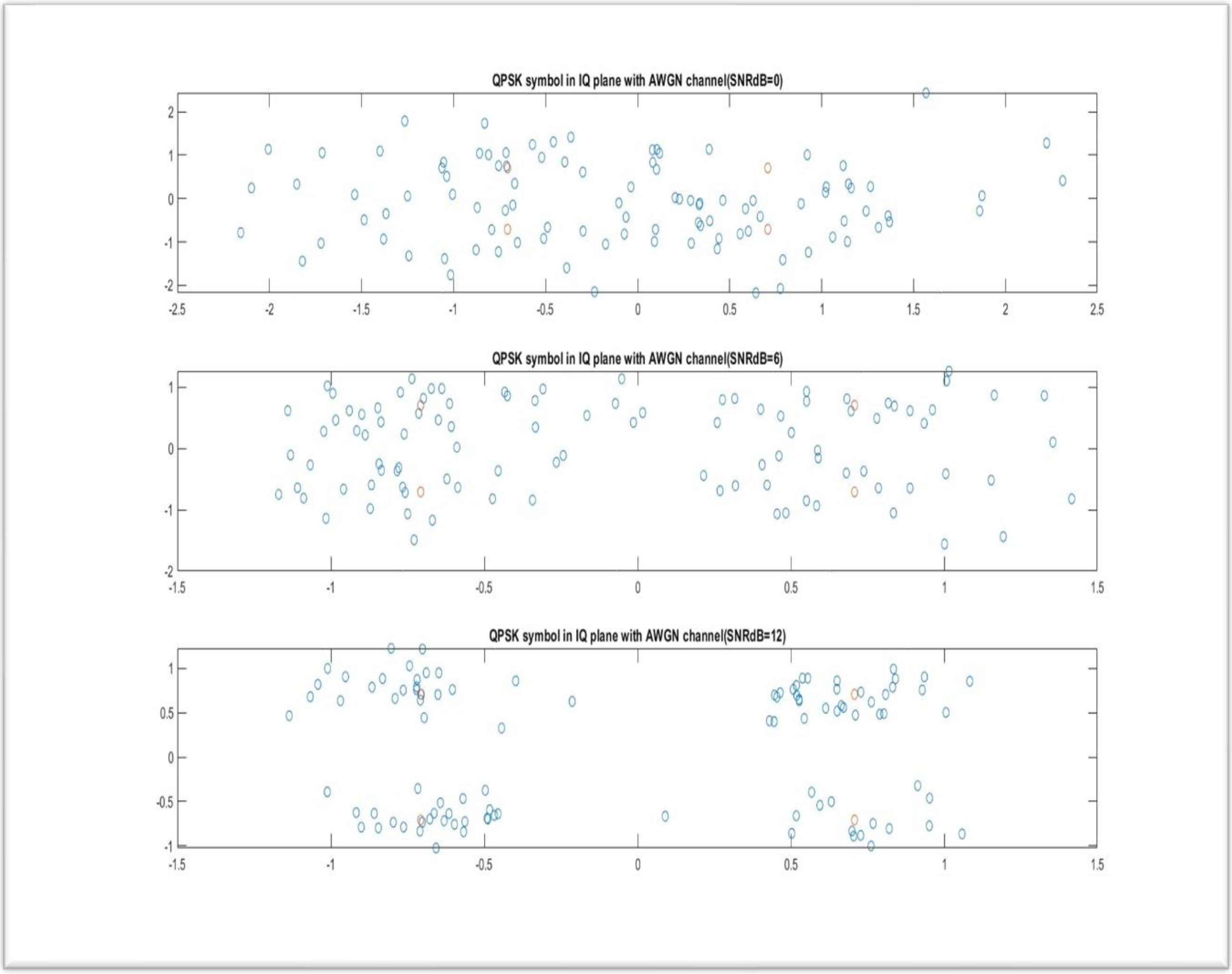
∴ For QPSK, $E_b = a^2$ and Here in my case $E_b = 1$.

∴ Average $E_b = \boxed{E_s = \frac{4E_b}{4} = 1}$ ①

∴ $\boxed{E_s = 1}$

- Our modulator is ready. Now, we add AWGN channel to it.
- Thus, we will get noise because we enable this AWGN channel and this will be added to Symbol vector S.
- After addition operation we will get received signal r.

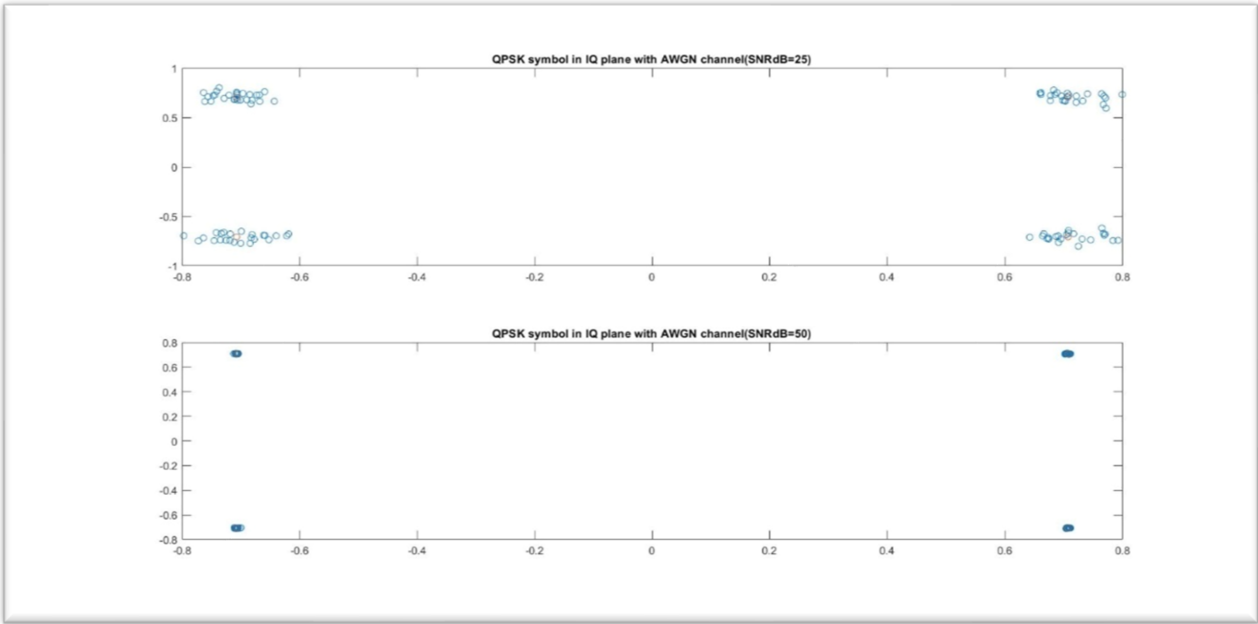
❖ **Here, I have mapped this signal for SNRdb values 0,6 and 12 db.**



1. Observation:

❖ By observing above plot, I can say that by increasing the value of SNR the radius of the cloud around each transmitted symbol decreases and another observation is that if we increase furthermore than the radius is further decrease.

❖ This is shown in below plot:



❖ Thus for 50 dB SNR, I got both received and transmitted signal approximately same.

❖ I have implemented minimum-distance based symbol detection by calculating Euclidean distance method.

❖ Now by implementing Monte-Carlo-Simulator a result showing the symbol error probability at the output of the Monte Carlo simulator.

❖ UUB Calculation:

* UUB calculation for QPSK:

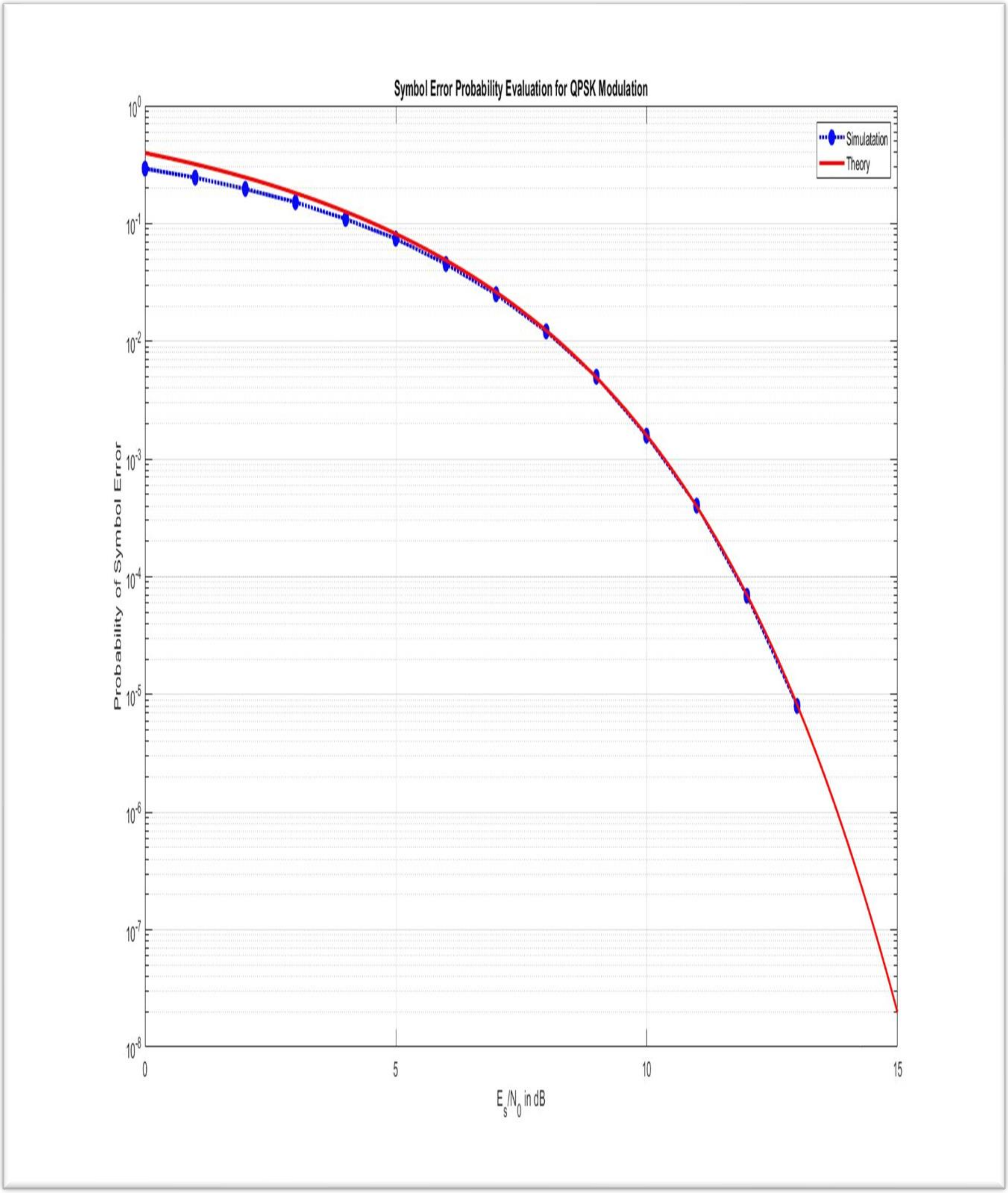
⇒ As we know, $P_{berr} = Q\left[\frac{d}{2\sigma_n}\right]$

∴ By my I-Q plot, I got UUB:

∴ $$\boxed{UUB = 2 * qfunc\left[\frac{1.412}{2\sigma_n}\right] + qfunc\left[\frac{1}{\sigma_n}\right]}$$

$\leftarrow$ (2)

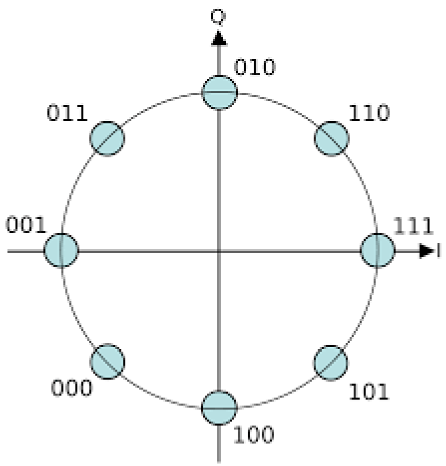
1. **For the QPSK simulator, a result showing the symbol error probability at the output of the Monte Carlo simulator and the corresponding analytical (UUB) result:**

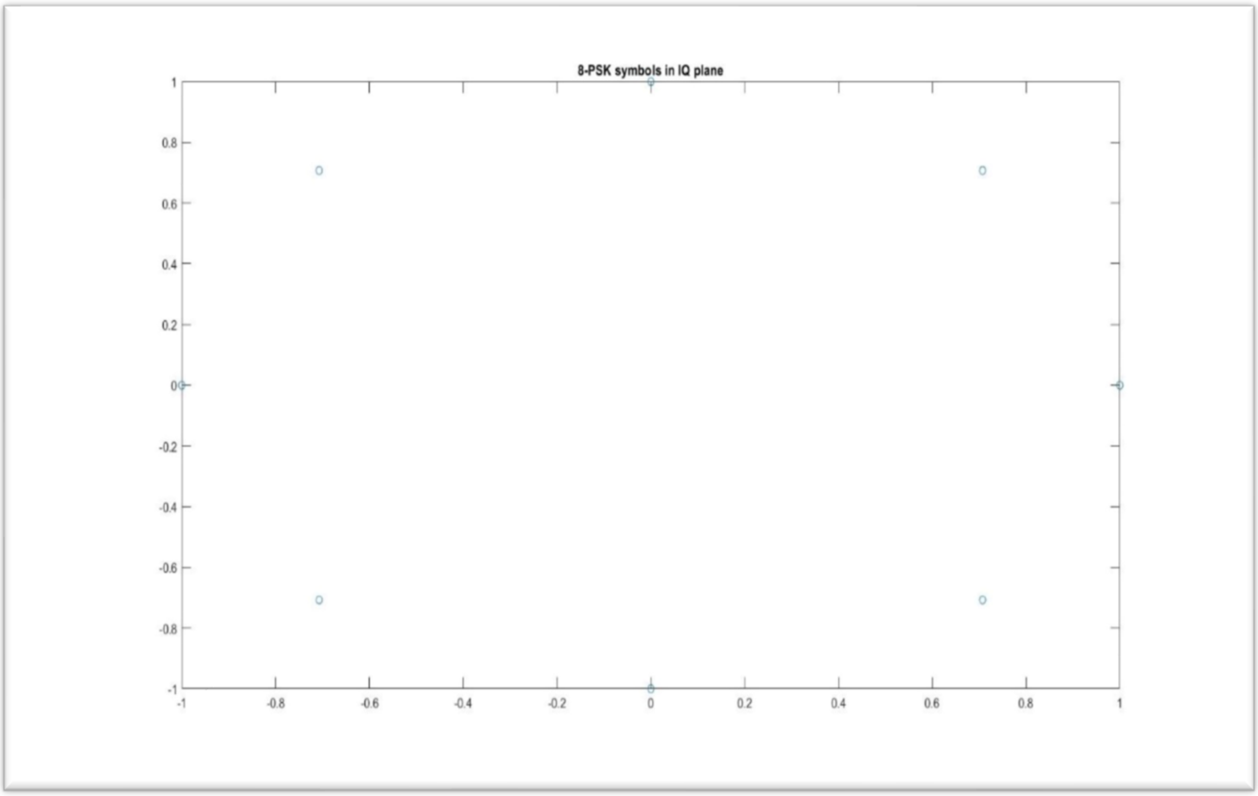


❖ Implementation of 8-PSK Modem:

2. For 8-PSK scheme, first we need to define M=2^k symbols where here k=3. Therefore M=8 symbols. We can designate these symbols in IQ plane shown below:

8-PSK: $(\mathrm{M}, \mathrm{k} = 8{,}3)$





- I have set a=1 for these symbols.

X_0, X_1, X_2

8PSK: An Example Bits to Symbol Mapping

$X_0, X_1, X_2 = 1,\ 1, 1 \rightarrow s_i,\ s_q = +1,\ 0$ volts

$X_0, X_1, X_2 = 1, 0, 1 \rightarrow s_i,\ s_q = +\frac{1}{\sqrt{2}},\ \ -\frac{1}{\sqrt{2}}$ volts

$X_0, X_1, X_2 = 0, 1, 1 \rightarrow s_i,\ s_q = -\frac{1}{\sqrt{2}},\ \ +\frac{1}{\sqrt{2}}$ volts

$X_0, X_1, X_2 = 0, 0, 0 \rightarrow s_i,\ s_q = -\frac{1}{\sqrt{2}},\ \ -\frac{1}{\sqrt{2}}$ volts

$X_0, X_1, X_2 = 0, 1, 0 \rightarrow s_i,\ s_q = 0,\ 1$ volts

$X_0, X_1, X_2 = 1, 1, 0 \rightarrow s_i,\ s_q = +\frac{1}{\sqrt{2}},\ \ +\frac{1}{\sqrt{2}}$ volts

$X_0, X_1, X_2 = 1,0, 0 \rightarrow s_i,\ s_q = 0,\ \ -1$ volts

$X_0, X_1, X_2 = 0, 0,1 \rightarrow s_i,\ s_q = -1,\ 0$ volts

$s = s_i, s_q$

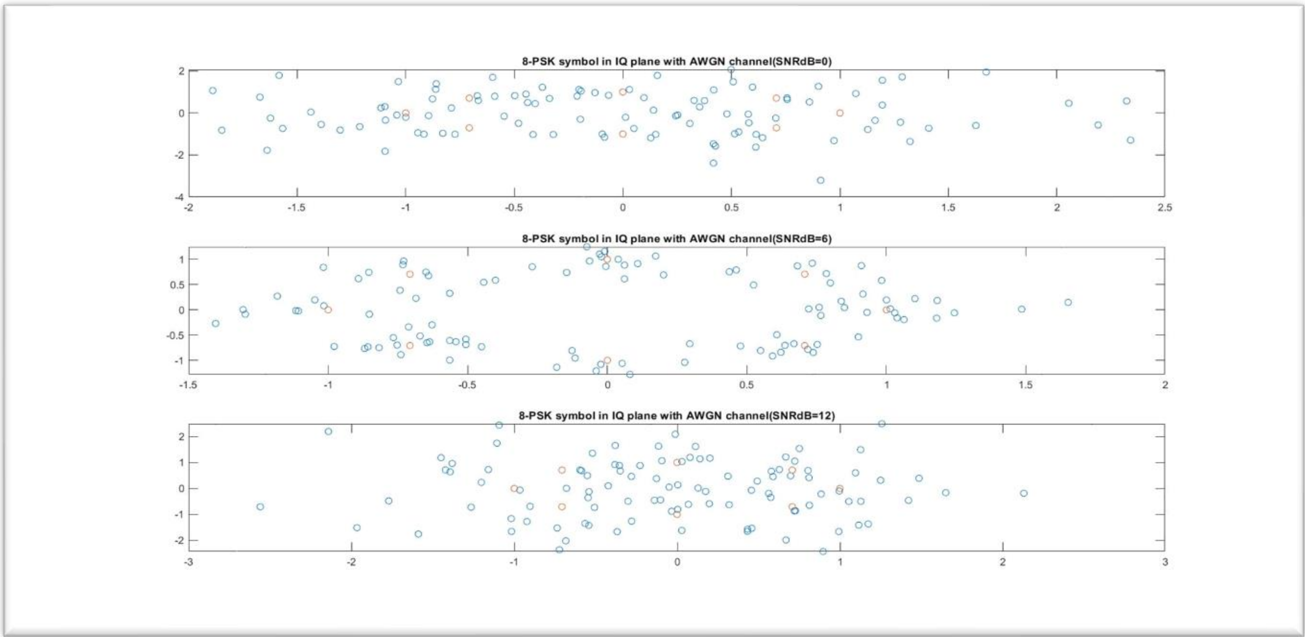
* Derivation of E_s in 8-PSK Scheme:

⇒ Here, I have take $a = 1$ & likewise in QPSK we got $E_s = a^2 = 1$ & here also we got for all eight symbols $E_b = 1$.

∴ Average $E_b = E_s = 1$

∴ $\boxed{E_s = 1}$ —(1)

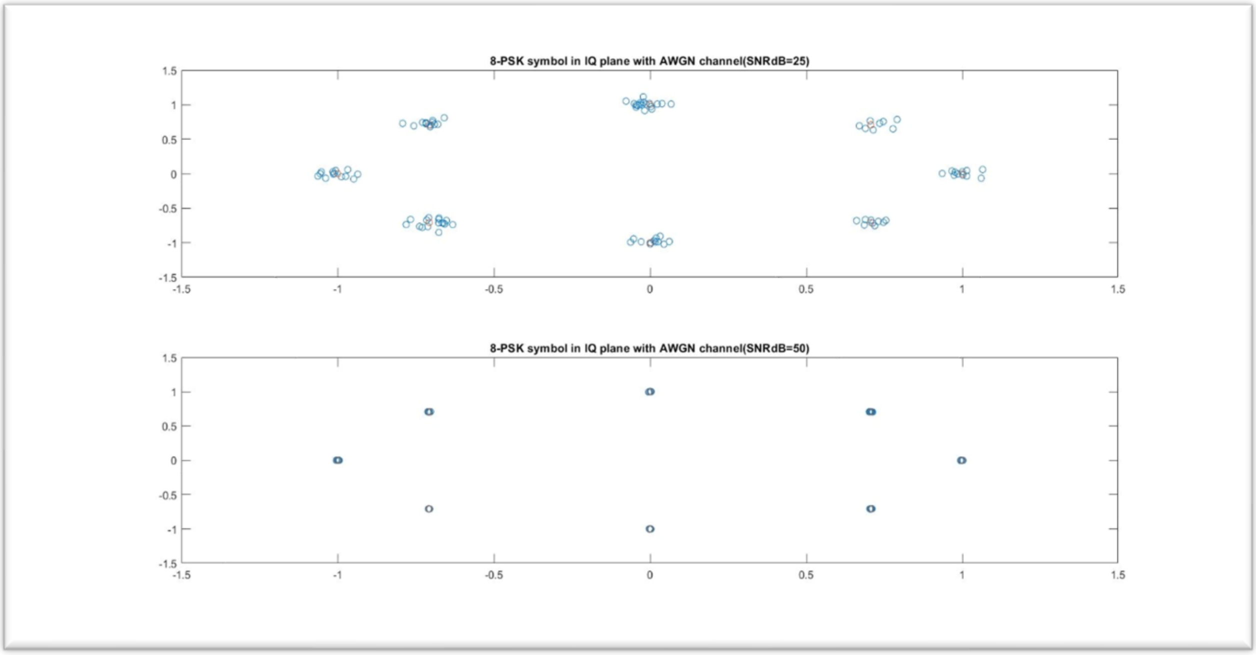
- **Here, I have mapped this signal for SNRdb values 0,6 and 12 db.**



3. Observation:

As we observe in QPSK that result has also seen by this plot.

4. For 25 dB and 50 dB SNR:



❖ UUB Calculation:

* UUB derivation for 8-PSK scheme:

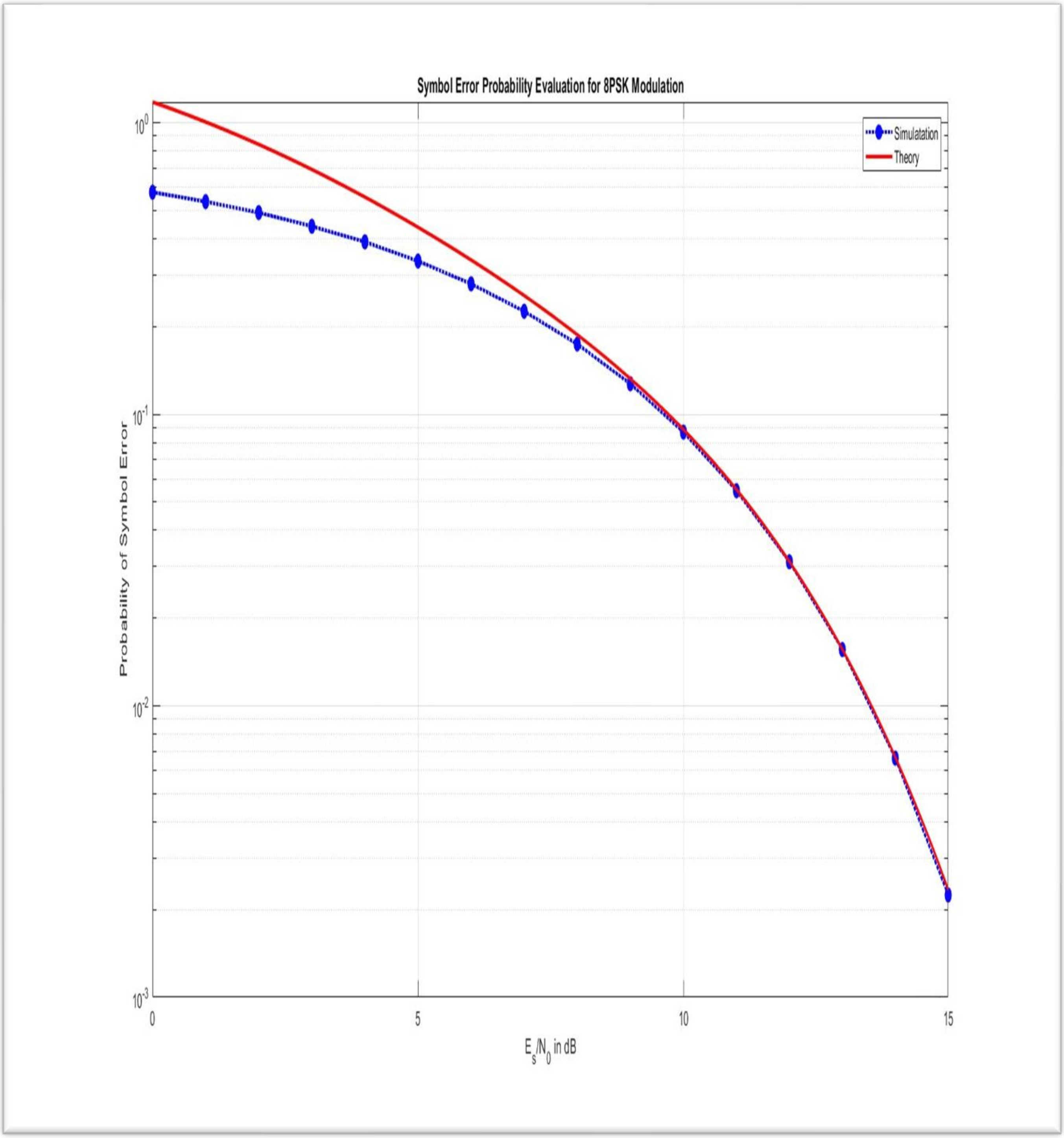
⇒ As, we know $P_{berror} = Q\left[\frac{d}{2\sigma_n}\right]$

∴ By above formula & looking at I-Q plane we can define UUB as follow:

$$\boxed{\begin{aligned} UUB = {} & 2 * qfunc(1.412/2\sigma_n) + 2 * qfunc(0.7654/2\sigma_n) \\ & + 2 * qfunc(1.8478/2\sigma_n) + qfunc\left(\frac{1}{\sigma_n}\right) \end{aligned}}$$

(2)

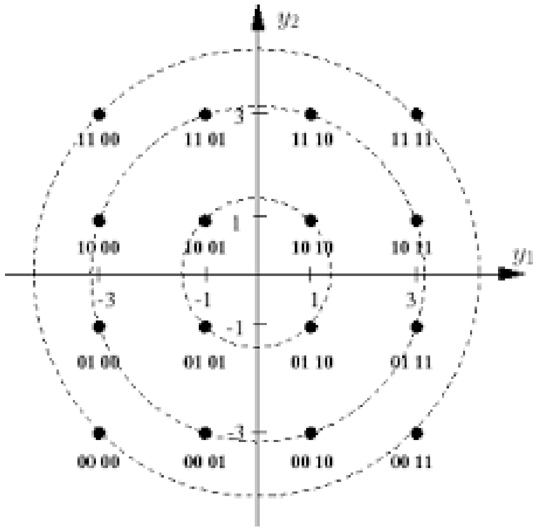
2. **For the 8-PSK simulator, a result showing the symbol error probability at the output of the Monte Carlo simulator and the corresponding analytical (UUB) result:**

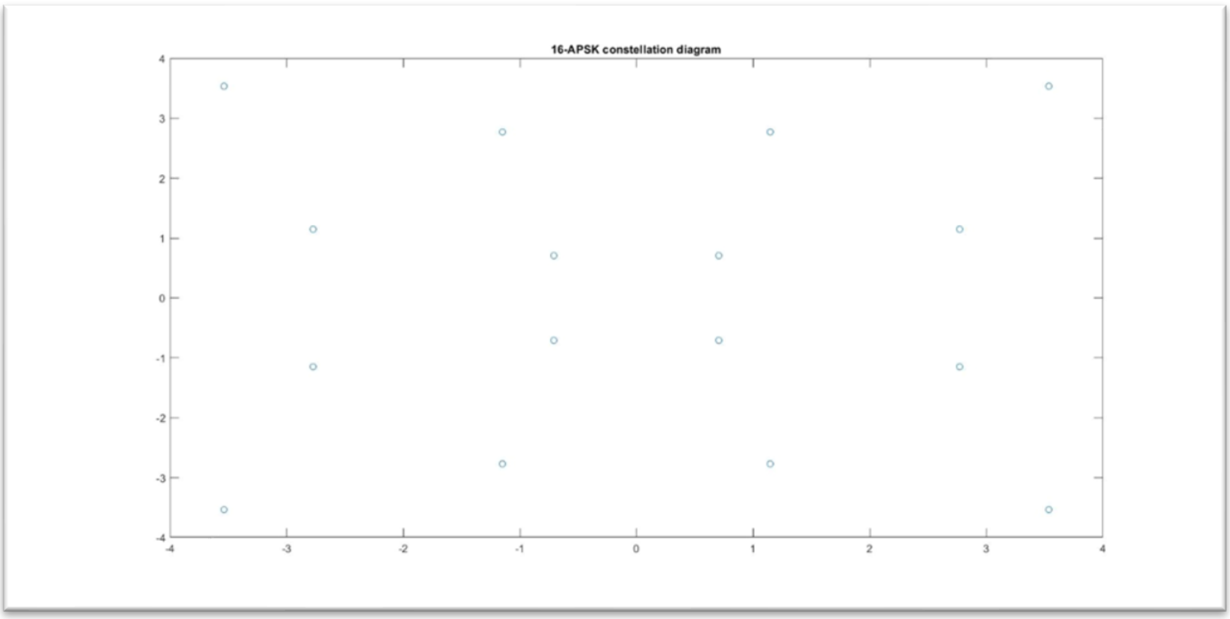


❖ Implementation of 16-APSK Modem:

3. For 16-APSK scheme, first we need to define M=2^k symbols where here k=4. Therefore M=16 symbols. We can designate these symbols in IQ plane shown below:

16-APSK: $(\mathrm{M}, \mathrm{k} = 16,4)$





4.Here, for 16-APSK Scheme we shift amplitude and phase.

* Derivation of E_s for 16-APSK:

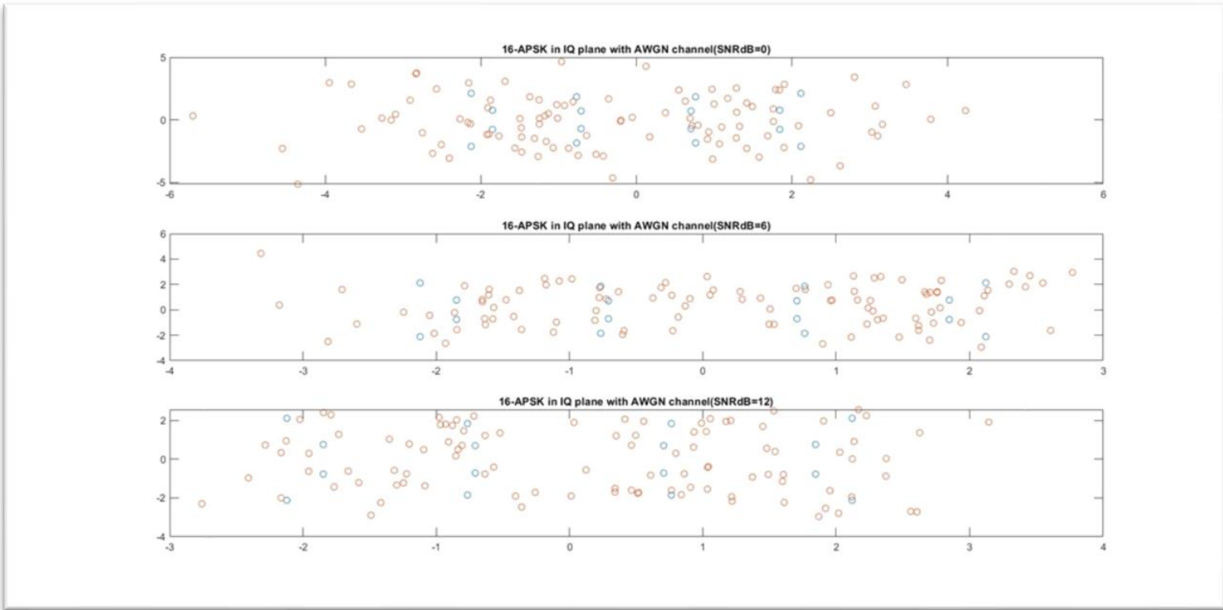
⇒ By looking at each circle we can define $E_{s_1} = 1$, $E_{s_2} = 9$ & $E_{s_3} = 25$ for all circles.

∴ By looking at I-Q plane:

$$\therefore \boxed{E_s = \frac{1\times4 + 9\times8 + 25\times4}{16} = \frac{4+72+100}{16} = 11}$$

L (1)

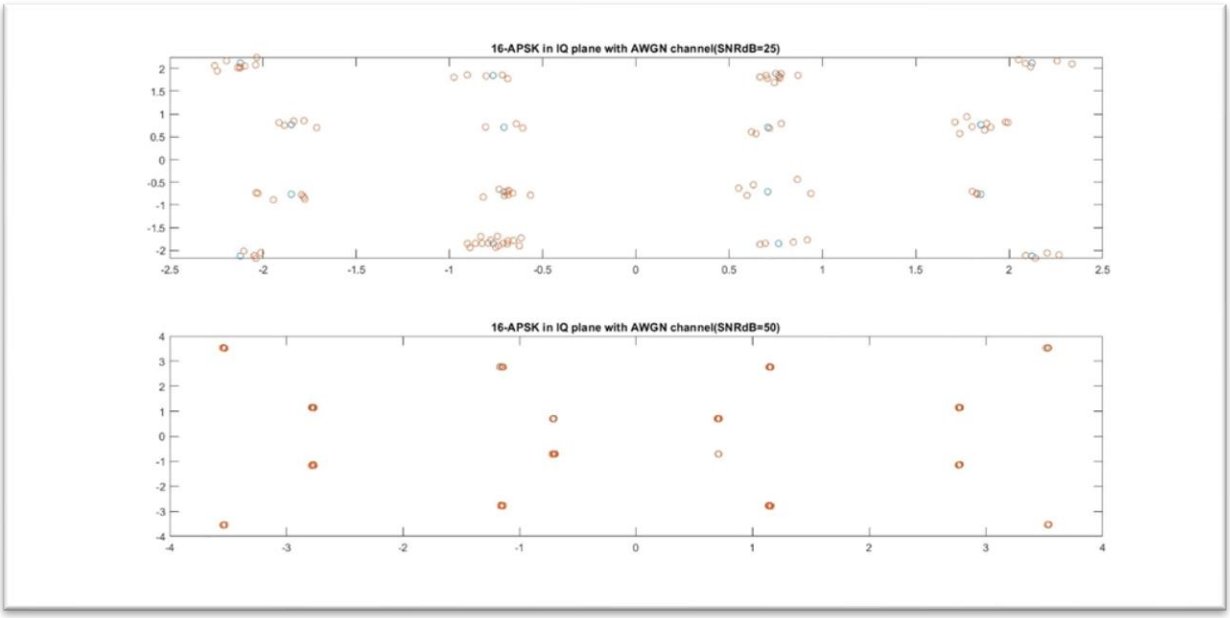
❖ **Here, I have mapped this signal for SNRdb values 0,6 and 12 db.**



❖ Observation:

As we observe in QPSK that result has also seen by this plot.

- **For 25 dB and 50 dB SNR:**



❖ UUB Calculation:

* UUB Calculation for 16-APSK:

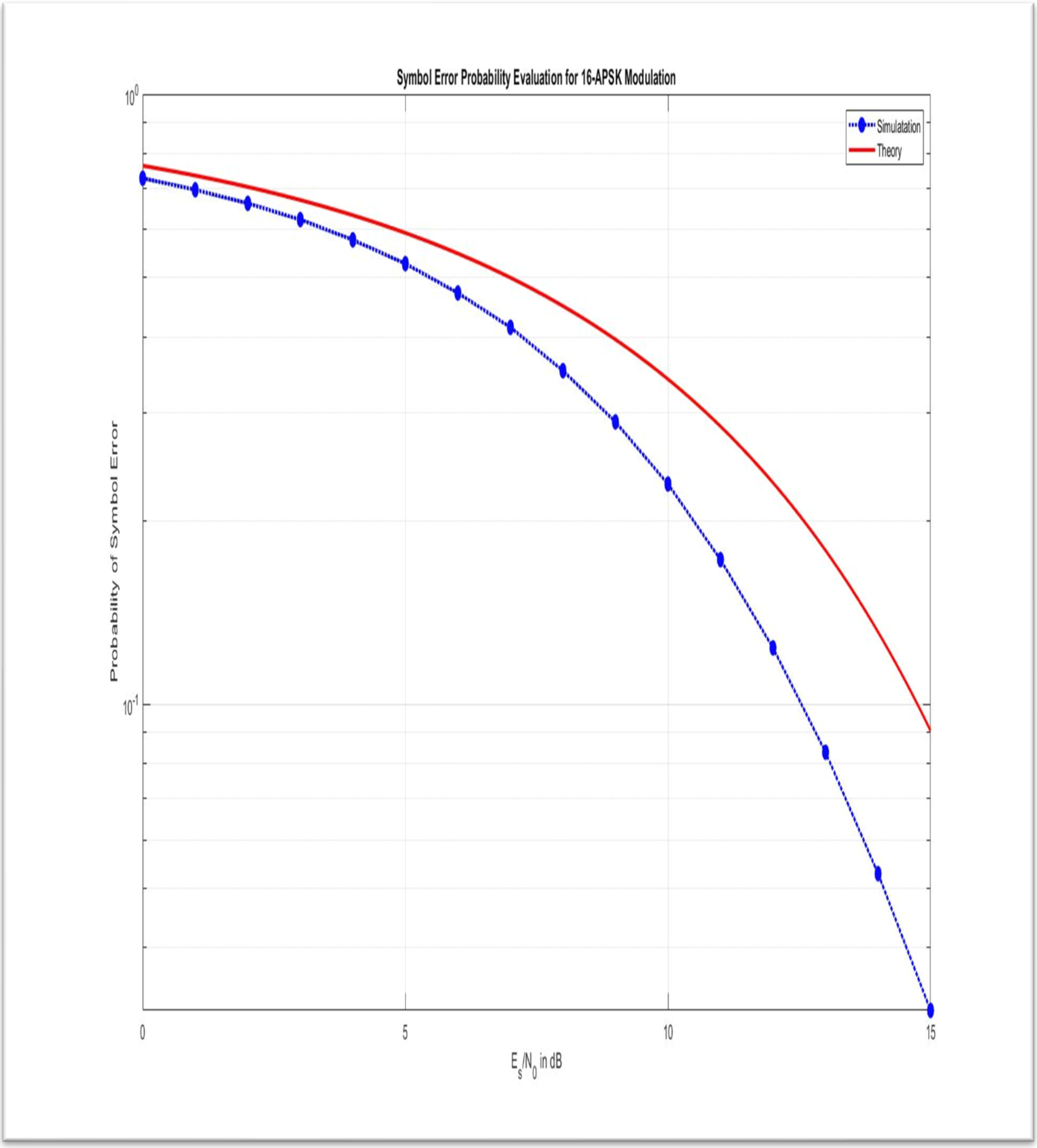
∴ Now, for I-UUB [Improved UUB] we will take only adjacent symbols in our calculation.

$$\therefore P_{berr} = Q\left(\frac{d}{2\sigma_n}\right)$$

$$\therefore \boxed{UUB = 2 * qfunc\left(\frac{1.412}{2\sigma_n}\right)}$$

(2)

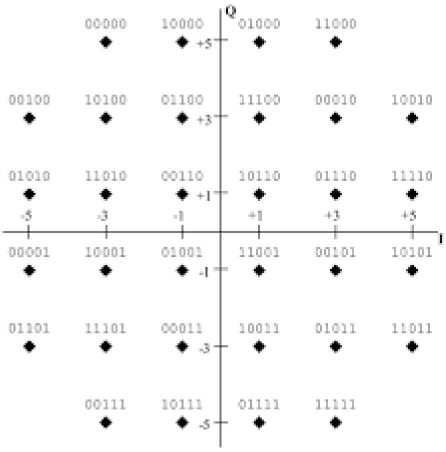
3. For the 16-APSK simulator, a result showing the symbol error probability at the output of the Monte Carlo simulator and the corresponding analytical (UUB) result:

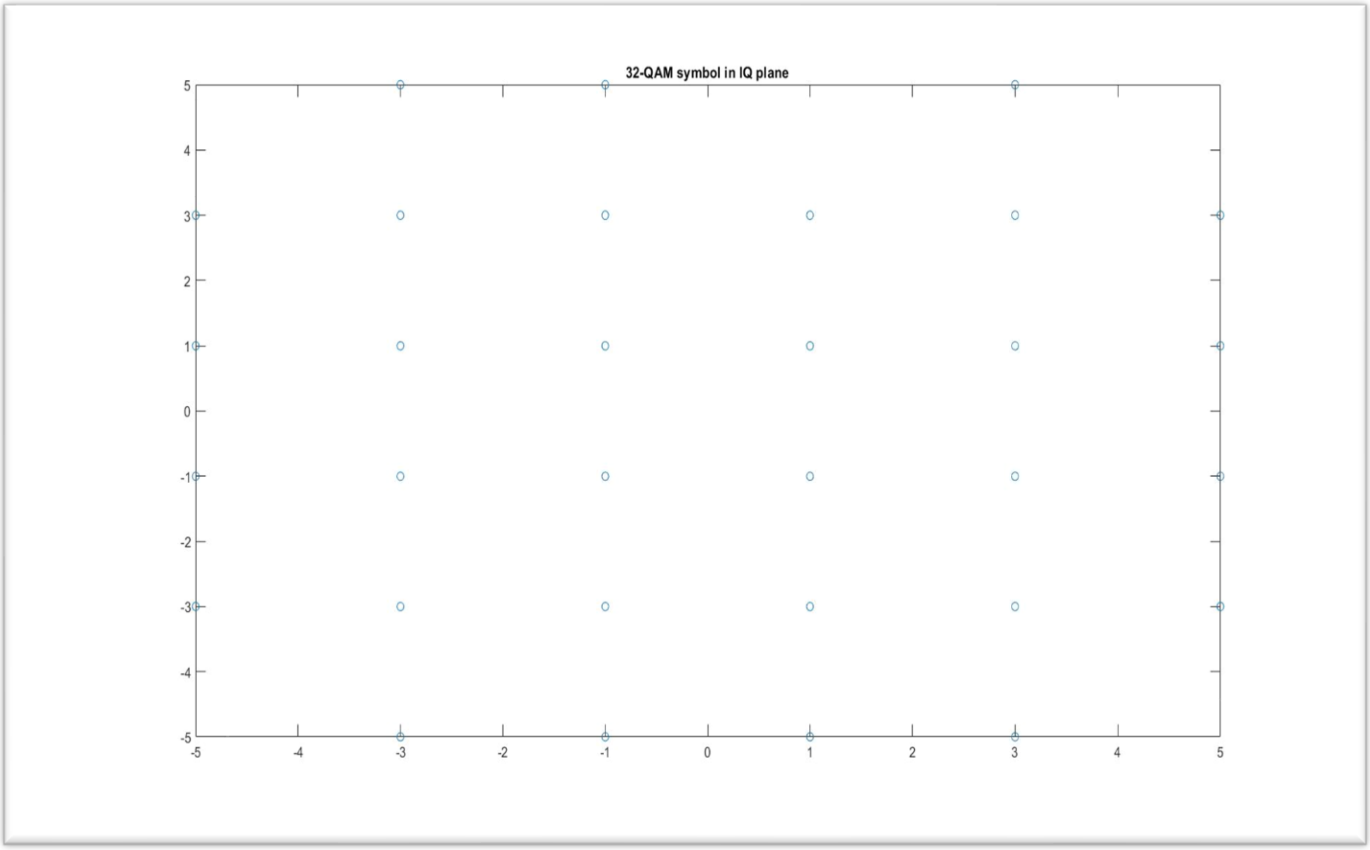


❖ Implementation of 32-QAM Modem:

1. For 32-QAM scheme, first we need to define M=2^k symbols where here k=5. Therefore M=32 symbols. We can designate these symbols in IQ plane shown below:

32-QAM: $(\mathrm{M}, \mathrm{k} = 32{,}5)$





* Derivation of E_s for 32-QAM scheme:

⇒ Now, looking at I-Q plot, we can obtain E_s by this eight E_s calculation.

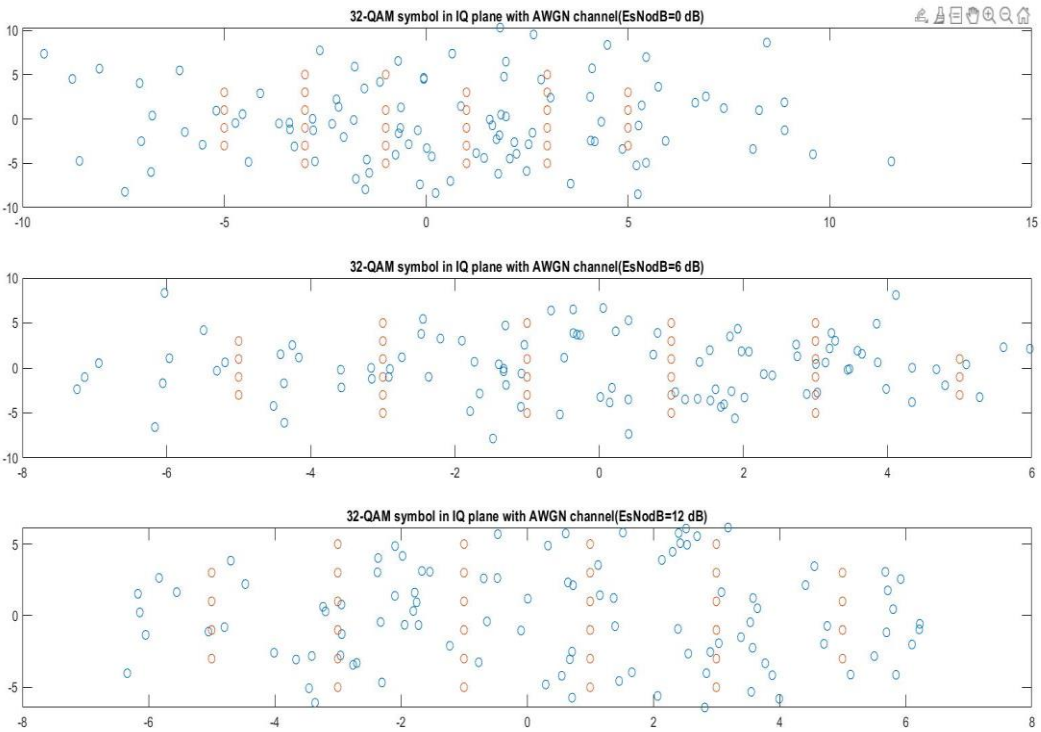
$$\therefore \; E_{s1} = 2 \;\&\; E_{s2} = 10 \;\&\; E_{s3} = 26 \;\&\; E_{s4} = 10$$

$$\& \; E_{s5} = 18 \;\&\; E_{s6} = 34 \;\&\; E_{s7} = 34 \;\&\; E_{s8} = 26$$

$$\therefore \; E_s = \frac{2\times4 + 10\times4 + 26\times4 + 10\times4 + 18\times4 + 34\times4 + 34\times4 + 26\times4}{32}$$

$$\therefore \; \boxed{E_s = 20} \quad \leftarrow (1)$$

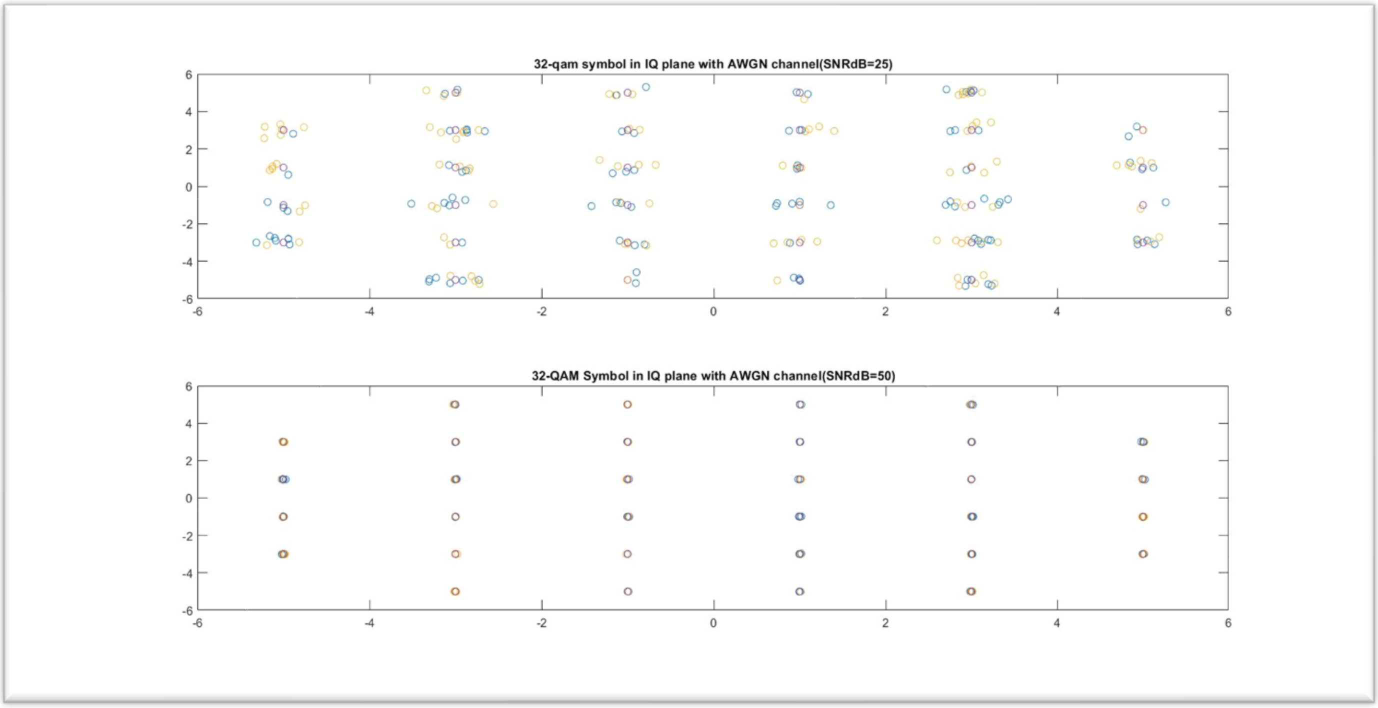
❖ **Here, I have mapped this signal for SNRdb values 0,6 and 12 db.**



2. Observation:

As we observe in QPSK that result has also seen by this plot.

- **For 25 dB and 50 dB SNR:**



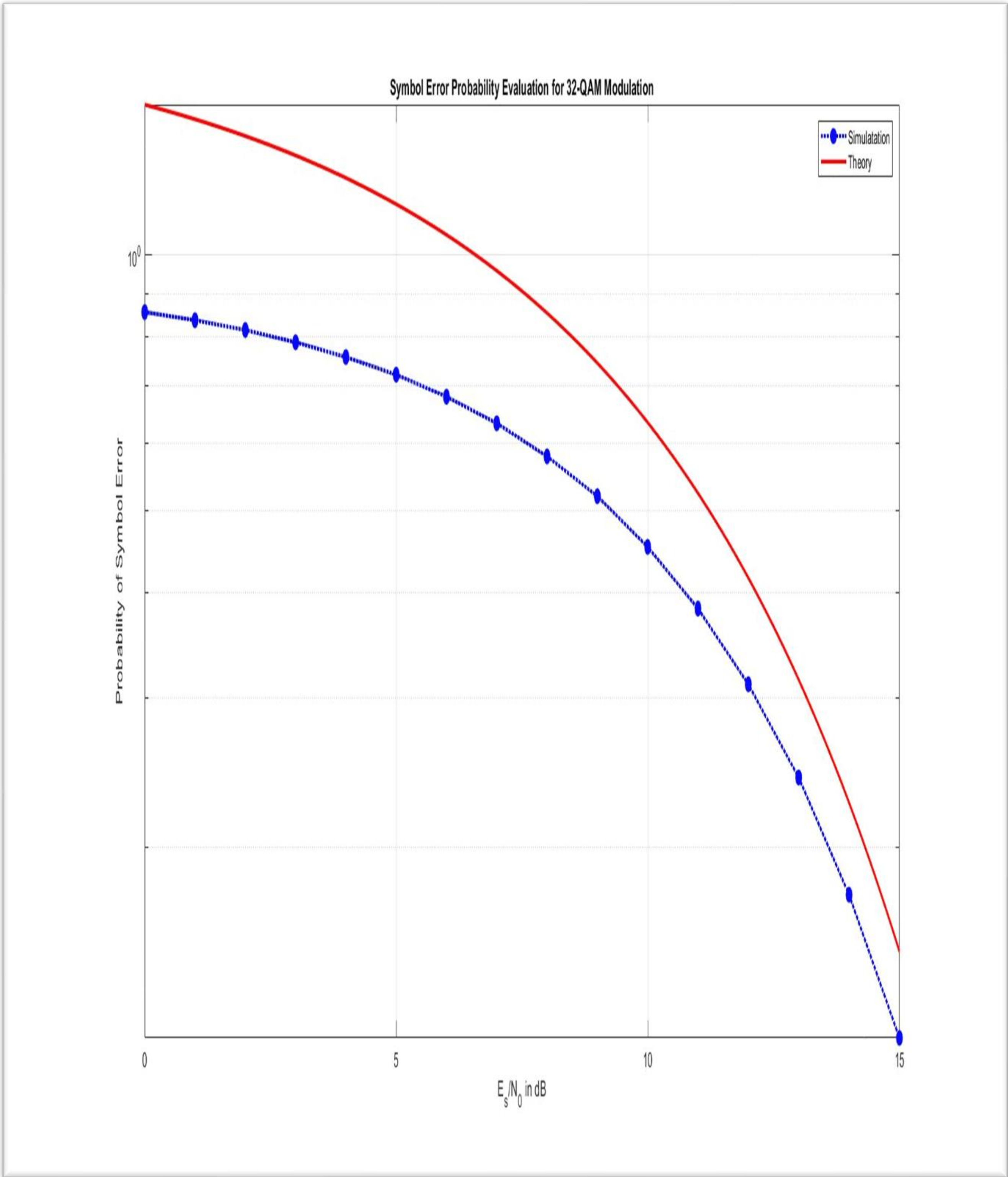
❖ **UUB Calculation:**

* Derivation for 32-QAM of UUB:

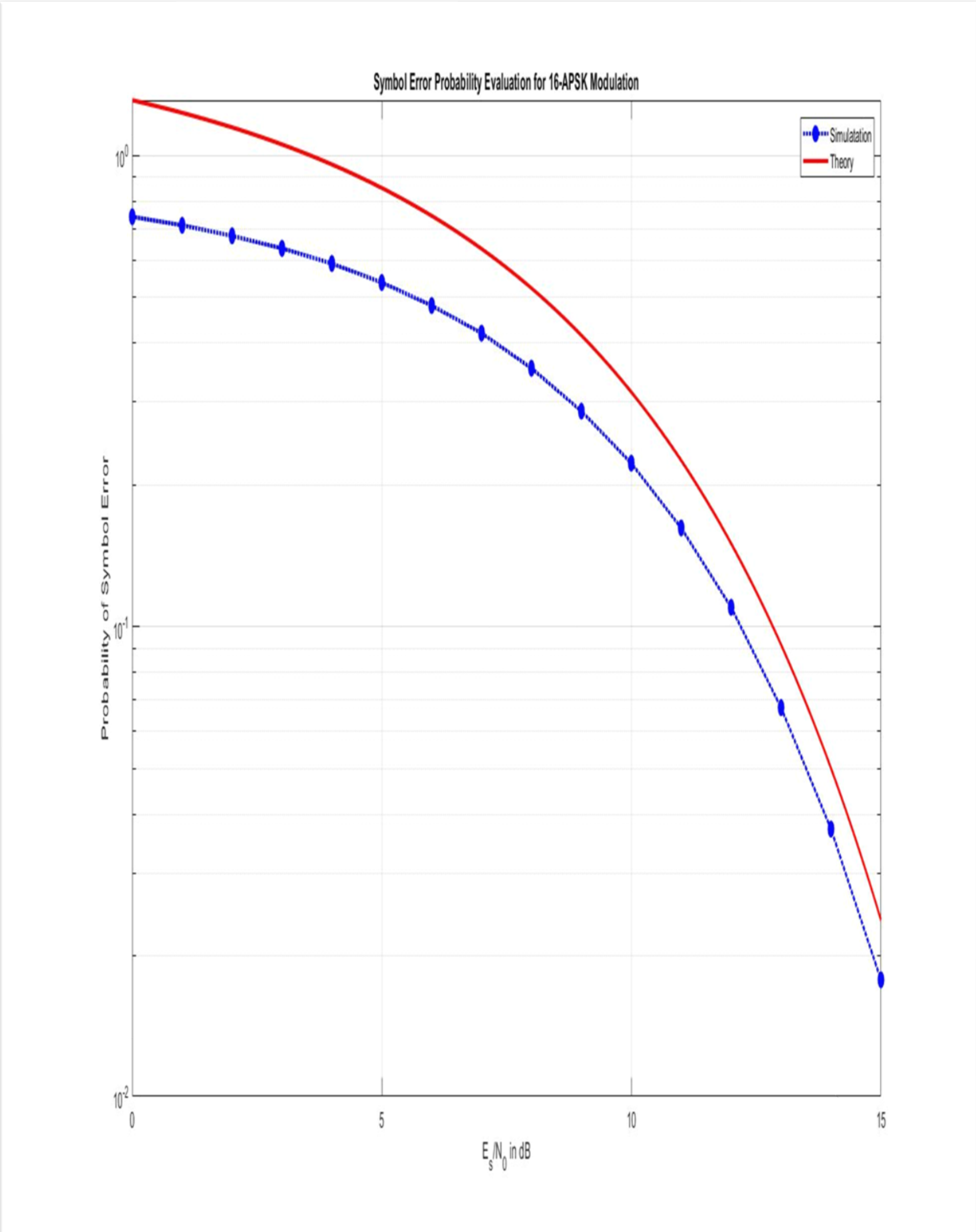
⇒ We can obtain UUB calculation likewise that we have done in 16-APSK.

$$\therefore \boxed{UUB = 4 * qfunc\left[\frac{1}{\sigma_n}\right]} \quad (2)$$

4. **For the 32-QAM simulator, a result showing the symbol error probability at the output of the Monte Carlo simulator and the corresponding analytical (UUB) result:**

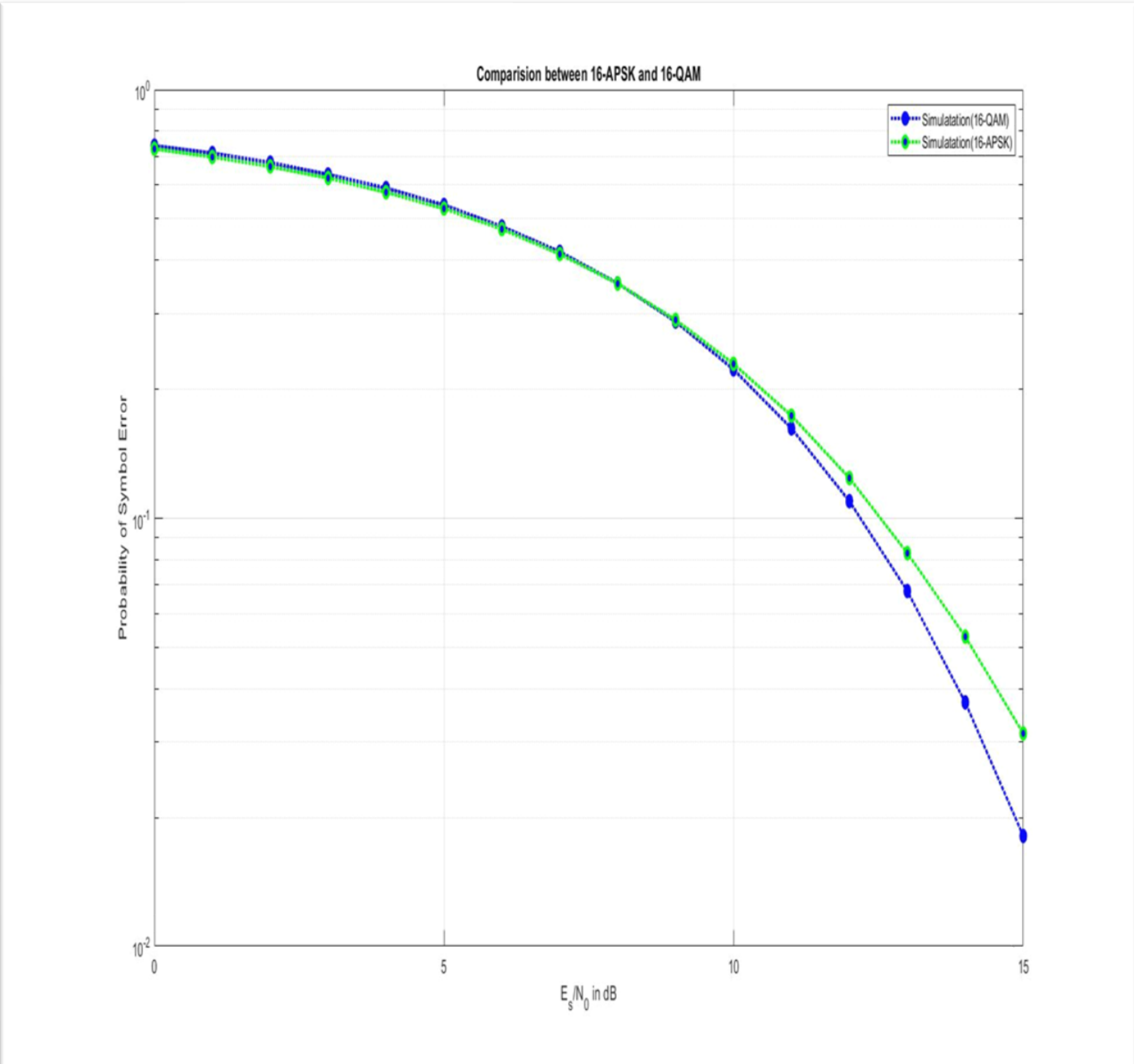


5. For the 16-QAM simulator, a result showing the symbol error probability at the output of the Monte Carlo simulator and the corresponding analytical (UUB) result:

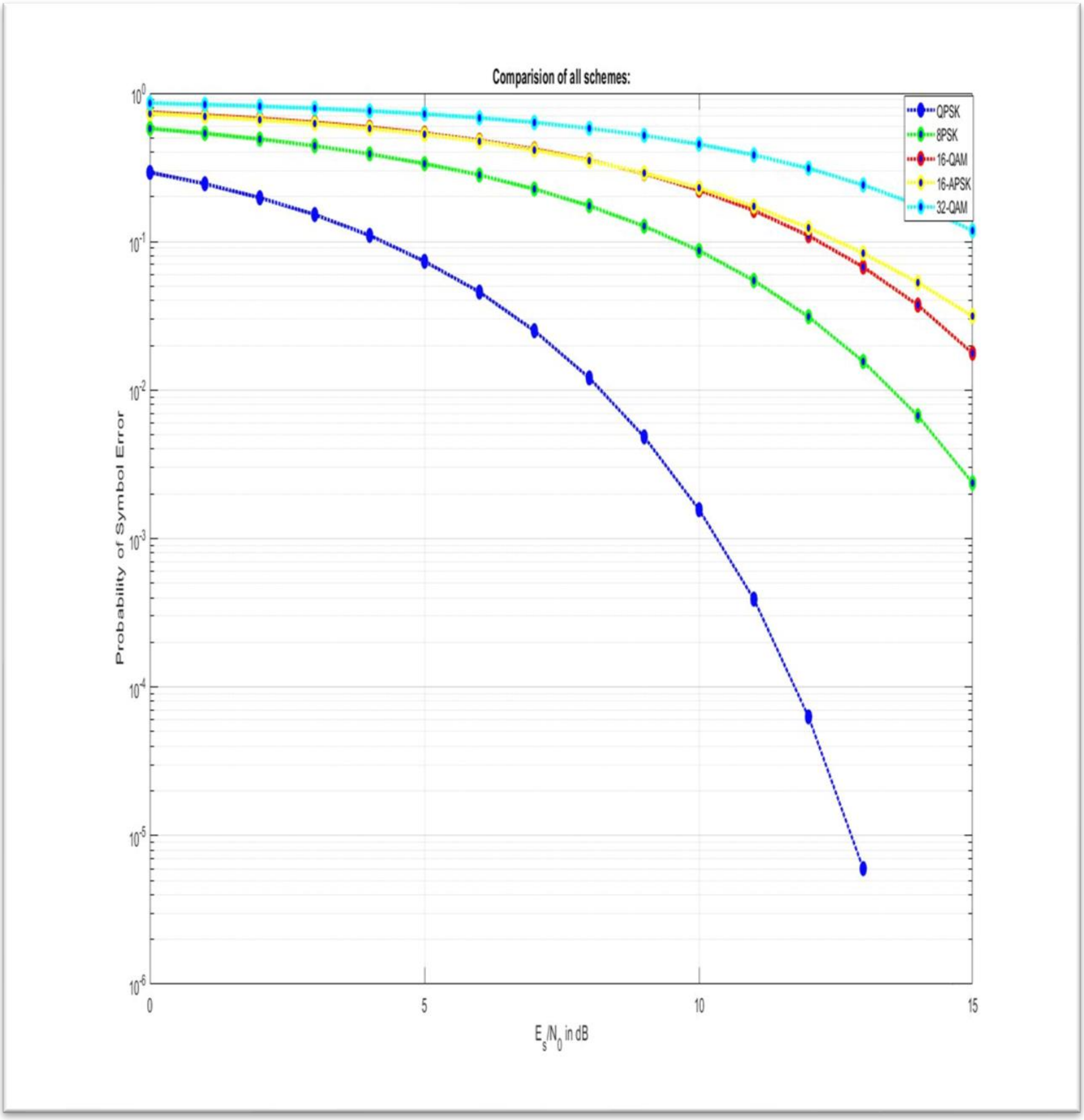


❖ Comparisons:

1. 16-QAM VS 16-APSK:

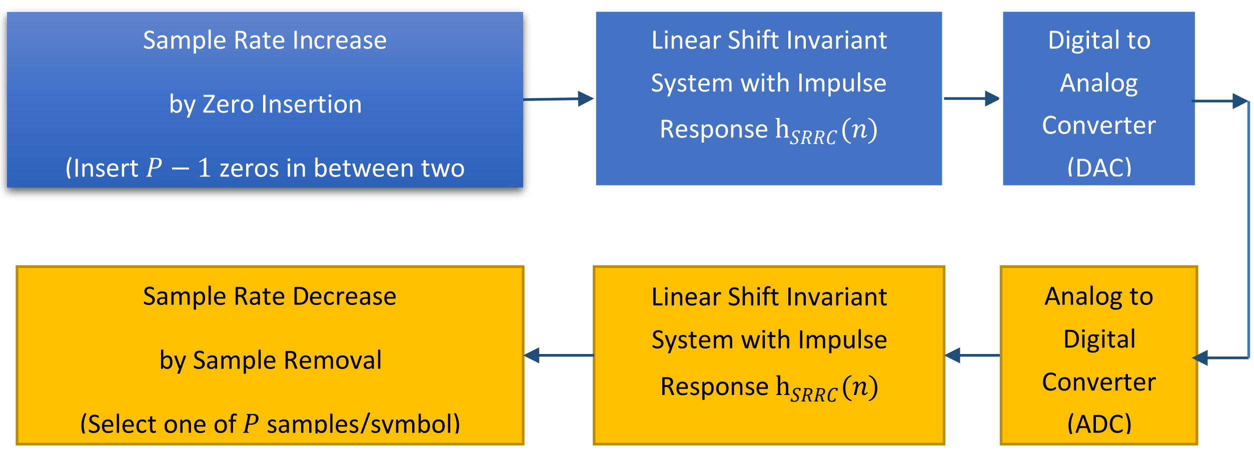


2. All Modulation Schemes:



❖ Realistic Model Implementation:

- ❖ We show all the four modulation schemes but in simplified modulation schemes there are certain limitations. Therefore, we can't implement it in a practical life.
- ❖ we have seen that the impulse sequence in the time-domain extends over an infinite range in the frequency-domain
- ❖ The spectral occupancy of the transmitted impulse sequence is unnecessarily large.
- ❖ Therefore ,we use pulse shaping filter at transmitter.

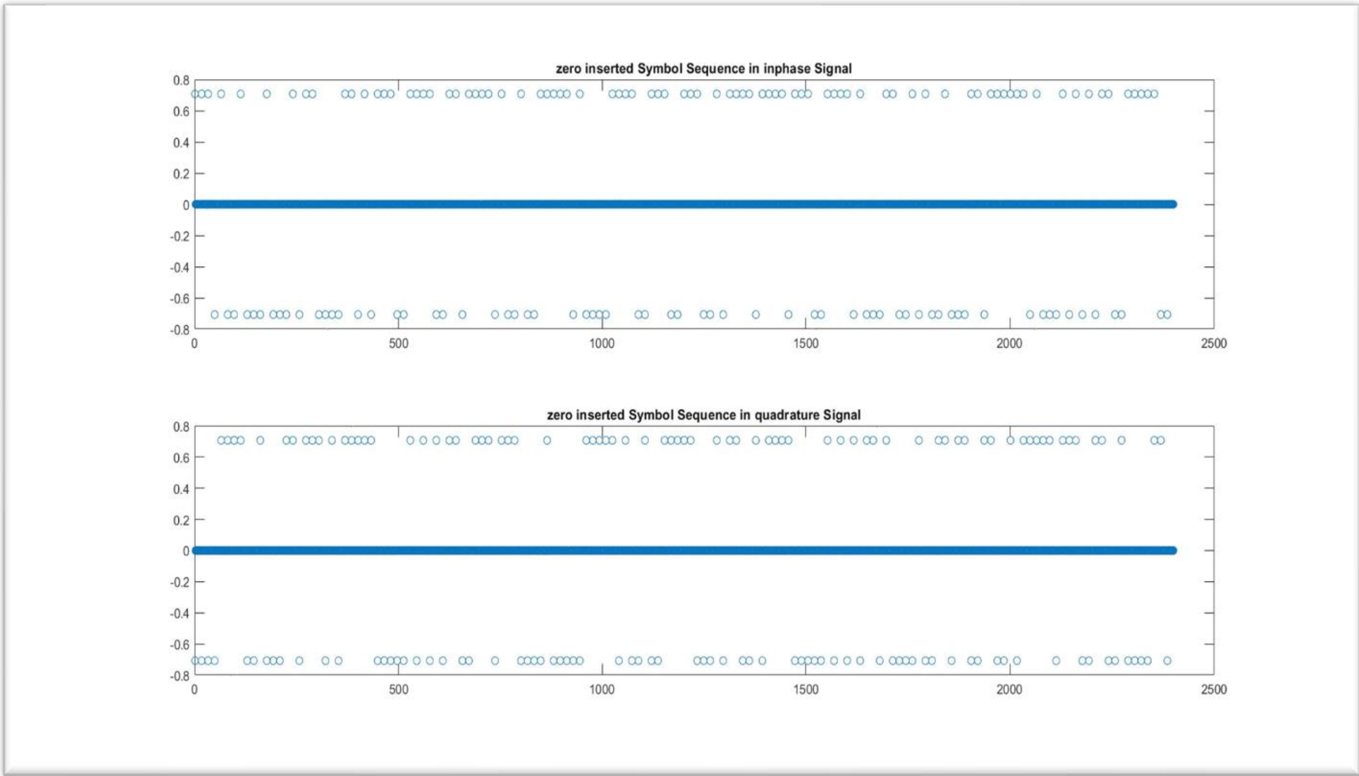


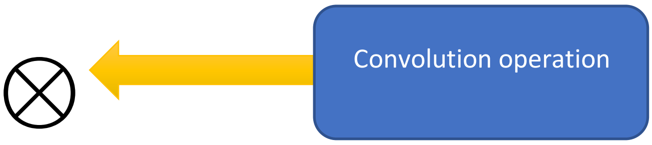
Realistic Model design

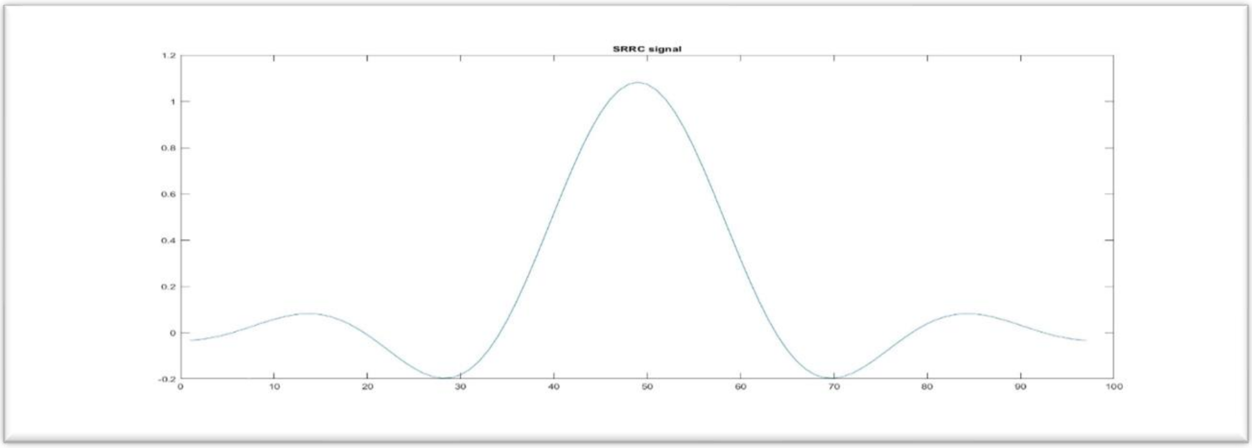
- **Baseband simulation with pulse-shaping and receiver LSI filter:**

  ❖ QPSK symbol for realistic model:

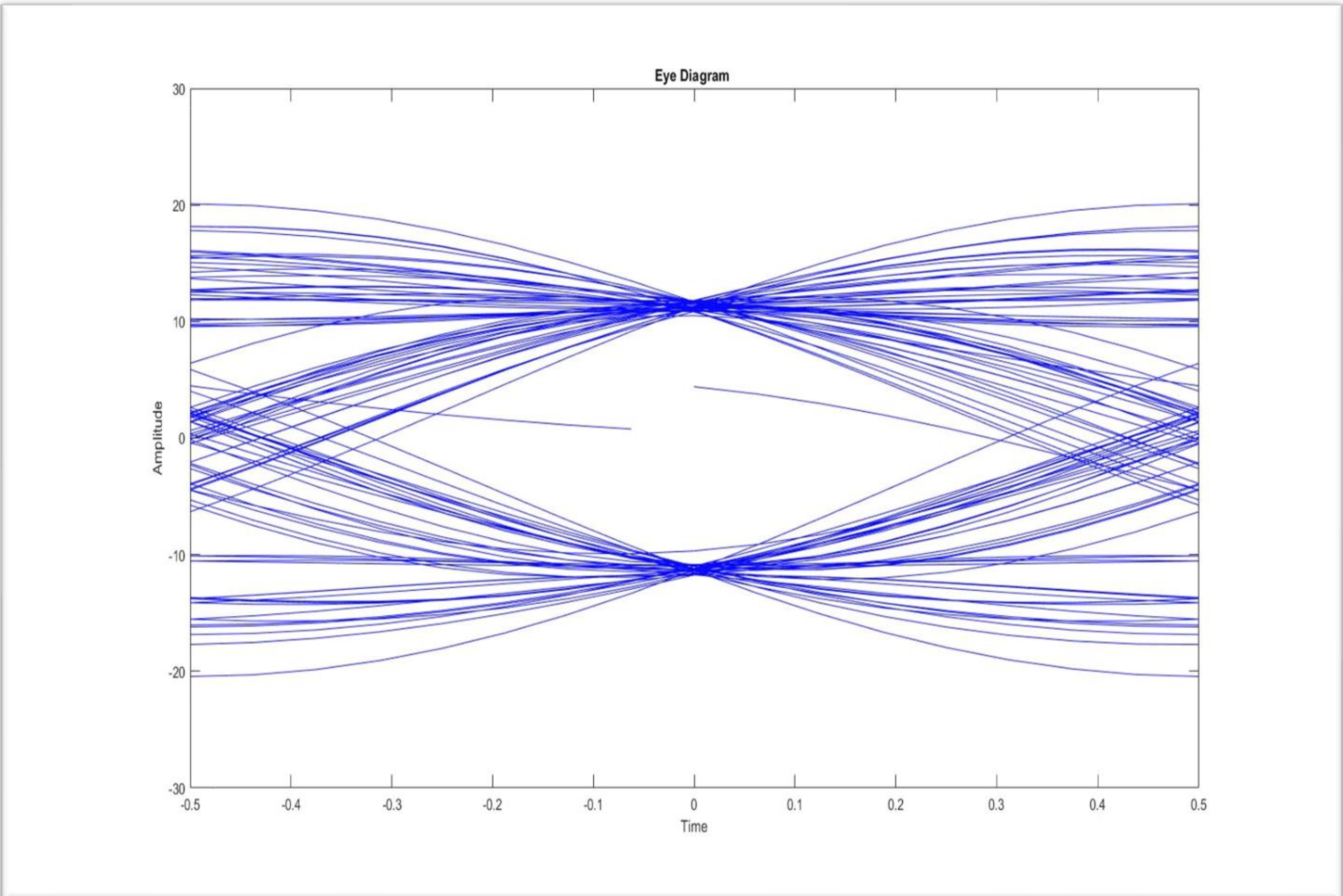
- Here, we send the In-phase and Quadrature symbol streams through the SRRC filter.



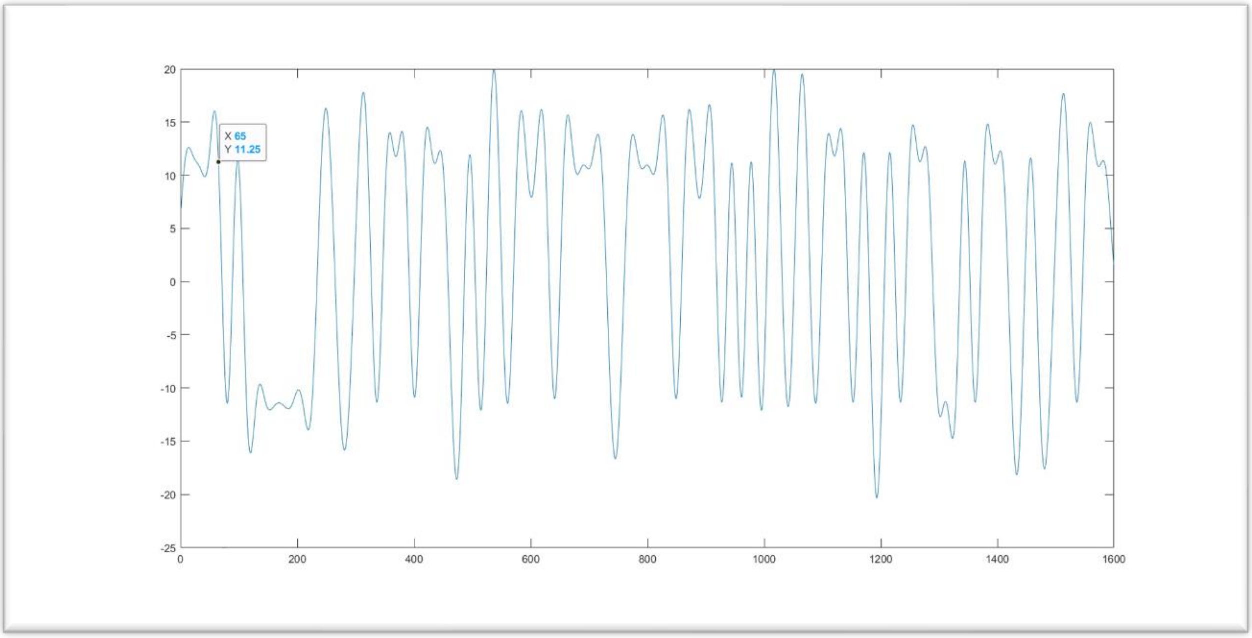




- Now if we send this pulse-shaped signal to same LSI filter, then we will get received signal.
- Actual design can't achieve an ideal inverse function at receiver. This is due to inter symbol interference.
- We can reduce the effect of ISI by choosing an accurate one of P sample.
- This can be done by eye diagram.
- The eye diagram is obtained by overlapping multiple randomly-generated symbol waveforms (the in-phase or the quadrature branch) on top of each other.



- Here, as you can see that maximum eye opening is at 11.25 amplitude.
- **So, the question is which sample out of 16 sample to choose?**
- This can be shown by graph.

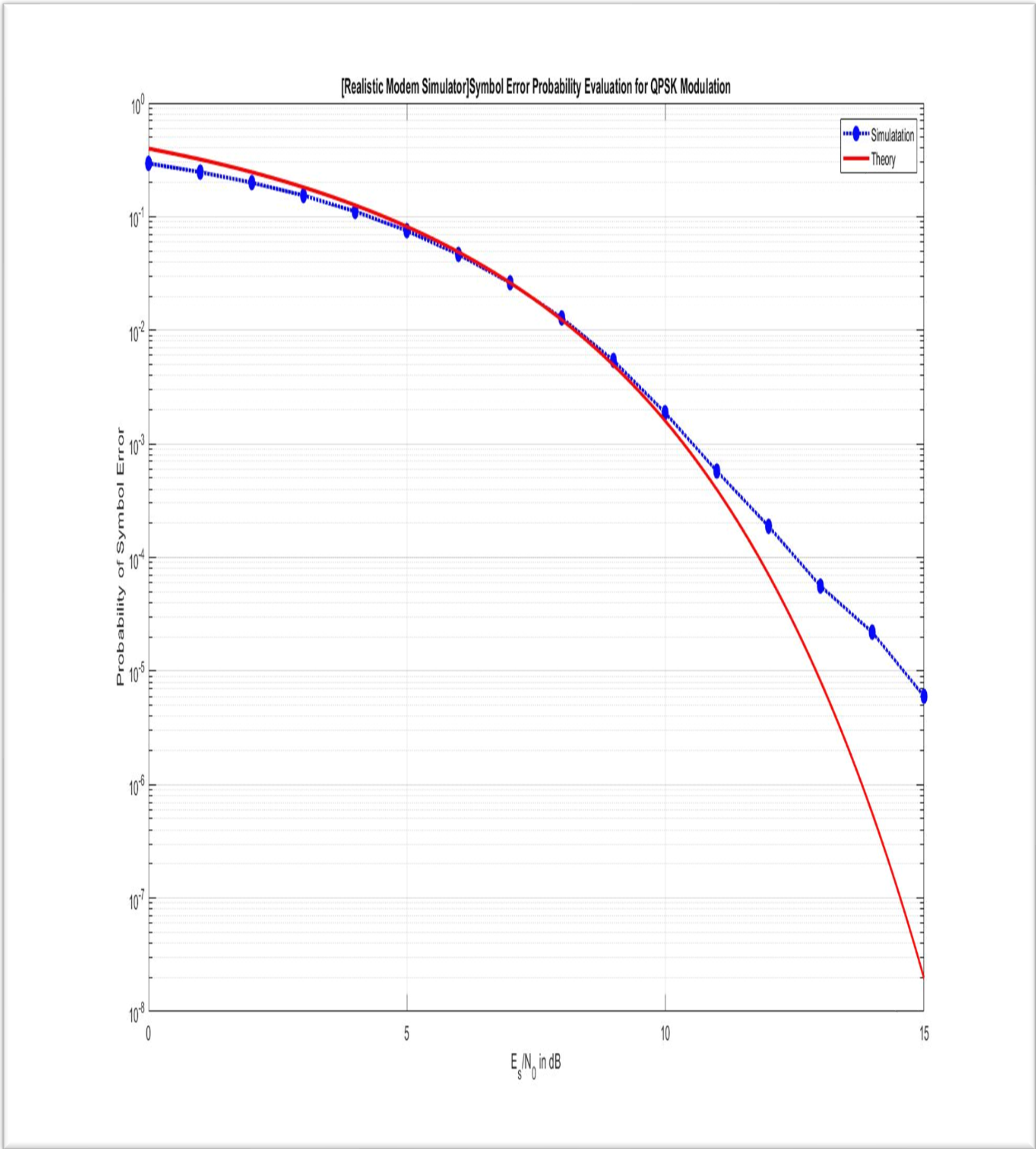


- By above graph, we find that for maximum eye opening instance 11.25 we got 65^{th} sample means mod (65,16)=1.
- Therefore, we can choose every first sample out of 16 sample from received signal.
- Thus, we will reduce the effect of ISI from our signal.
- Now, we can add AWGN channel to it. But the main thing is that we need to generate P*L noise samples as we have P samples.
- As the sample rate has increased by a factor of P, we need to increase the variance of the generated noise also by P.

➢ **The question is why it is so necessary to multiply the variance by factor of P?**

  - As we know, $P_N = N_0 * B = 2\sigma_n^2$. So as we increase B by factor of P without changing either Es or N0 and therefore we need to P_N by factor of P and variance is need to be increased by this viewpoint.

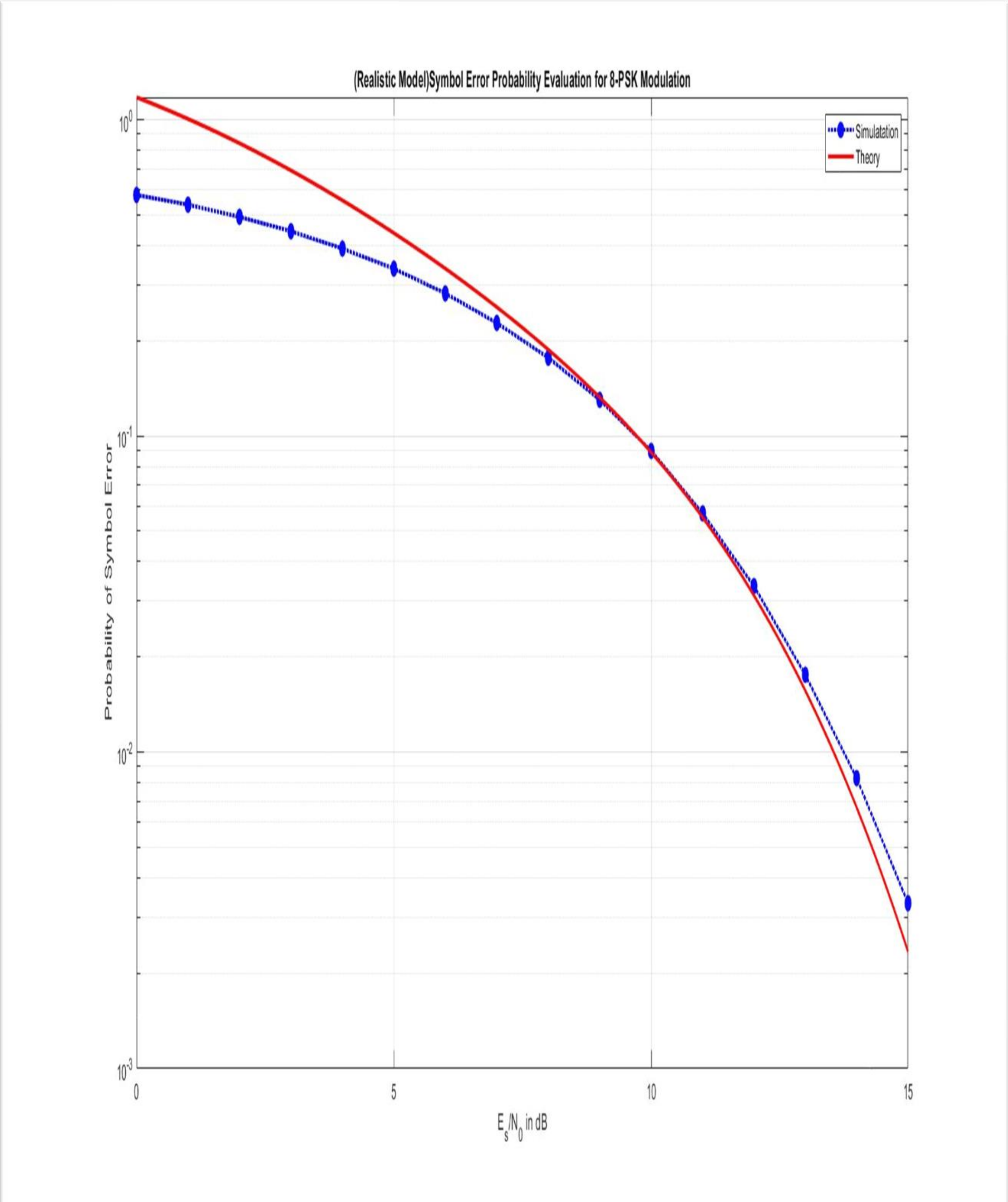
3. For the QPSK baseband simulator with the pulse-shaping, a result showing the symbol error probability at the output of the Monte Carlo simulator and the corresponding analytical (UUB) result:



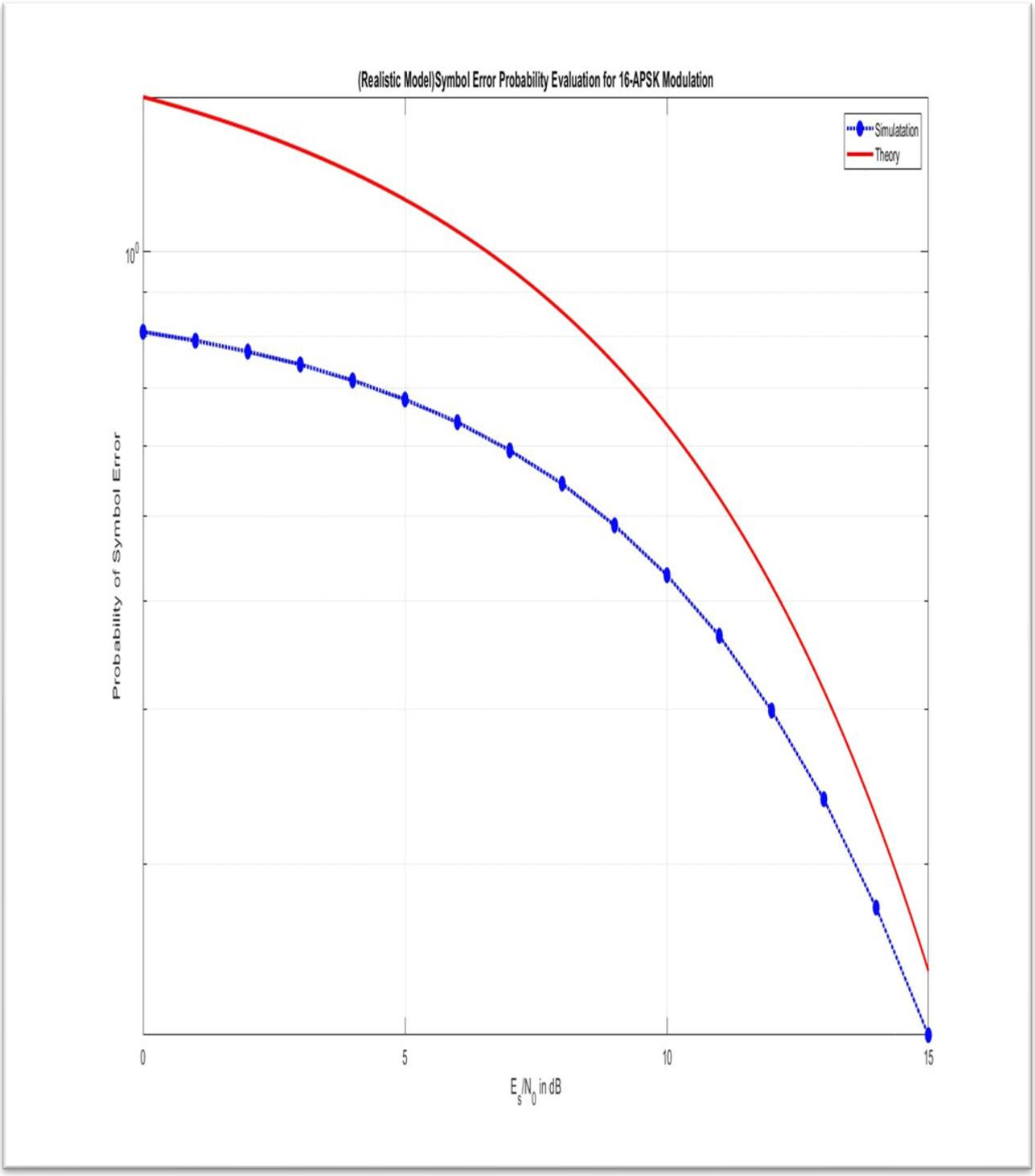
- You will notice that SNR>10, our upper-bound will cross simulation plot, this is unusual so **why it happens?**

- The reason behind this is that the SRRC filter is not ideal. Only with the ideal SRRC filter, the ISI becomes zero. When the SRRC filter is non-ideal, there is a small but nonzero ISI that remains. At SNR > 10 dB, this small ISI becomes more dominant than the AWGN (the AWGN starts becoming small compared to the ISI at SNR>10 dB) and therefore the performance of the demodulator saturates beyond 10 dB and exhibits a symbol error floor.

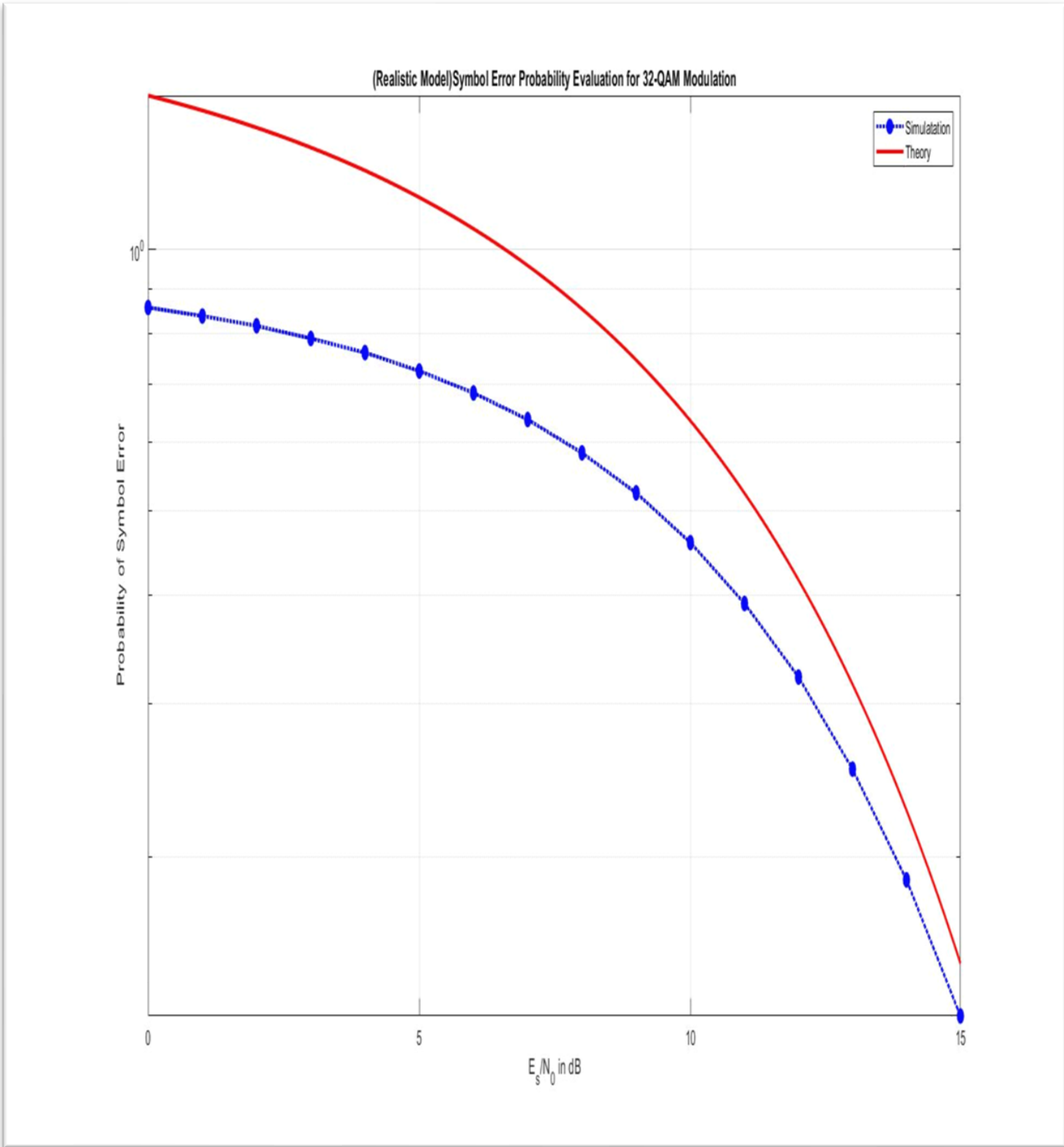
❖ **For the 8-PSK baseband simulator with the pulse-shaping, a result showing the symbol error probability at the output of the Monte Carlo simulator and the corresponding analytical (UUB) result:**



- **For the 16-APSK baseband simulator with the pulse-shaping, a result showing the symbol error probability at the output of the Monte Carlo simulator and the corresponding analytical (UUB) result:**

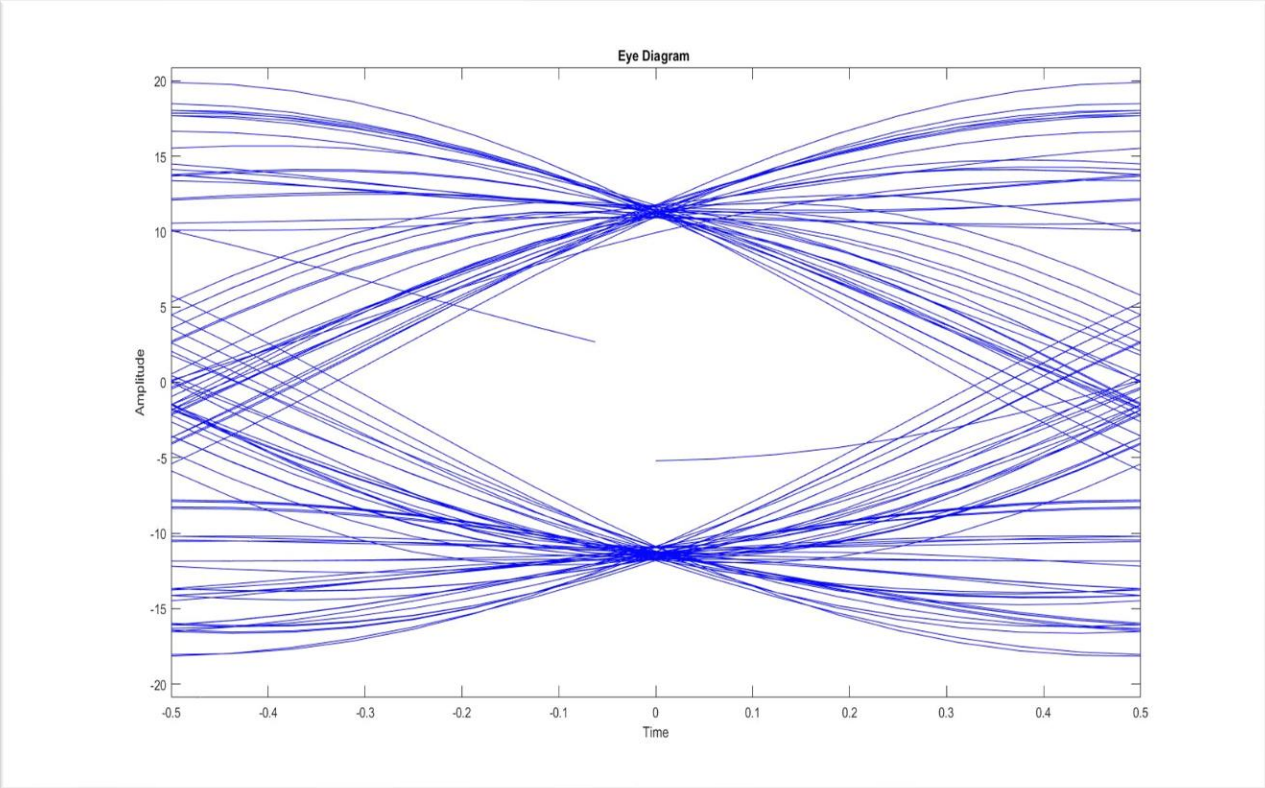


❖ **For the 32-QAM baseband simulator with the pulse-shaping, a result showing the symbol error probability at the output of the Monte Carlo simulator and the corresponding analytical (UUB) result:**

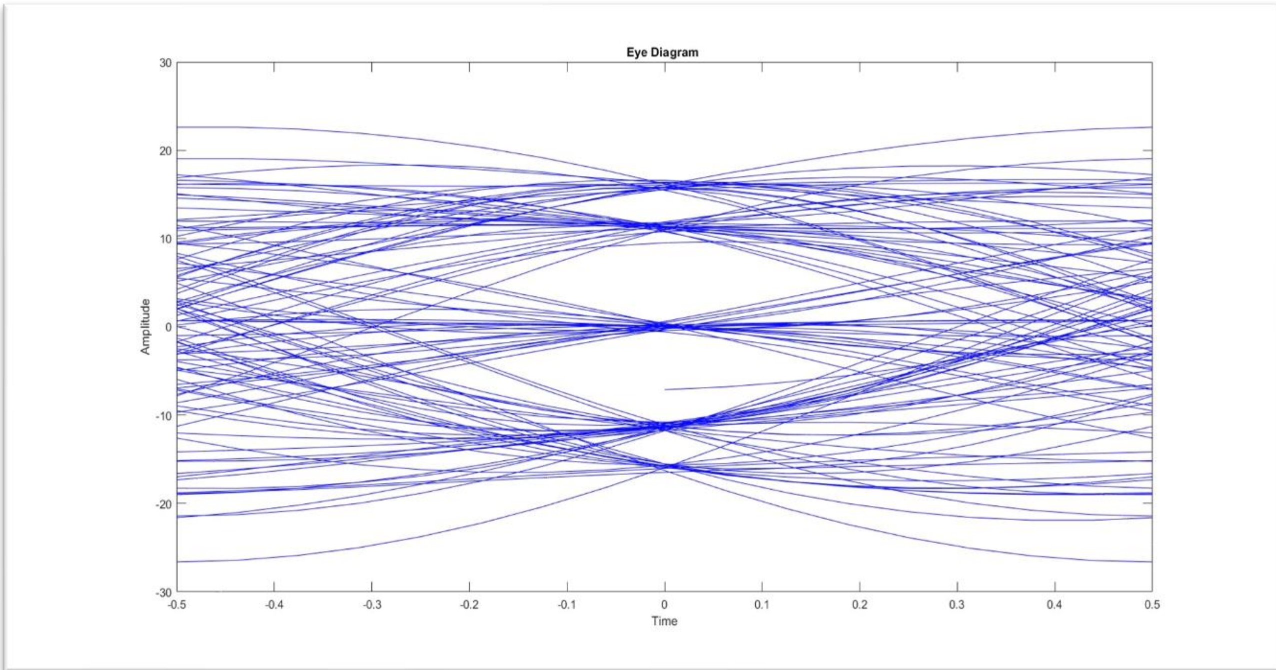


❖ Eye diagram for all schemes:

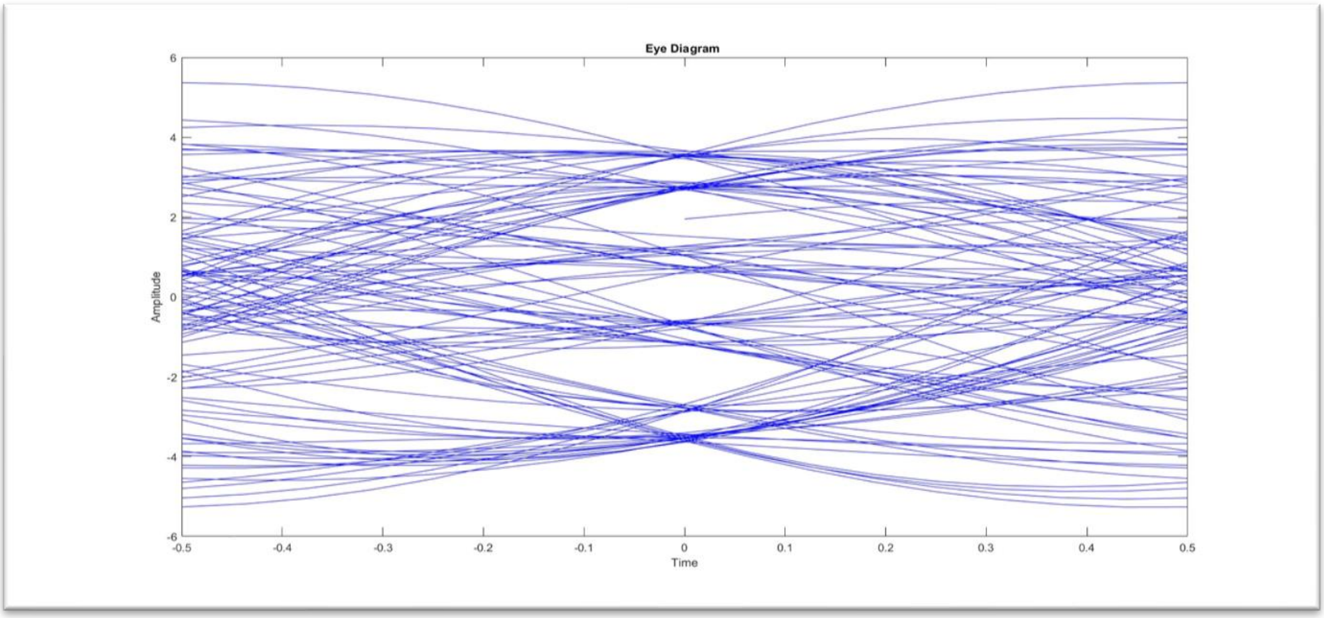
i. QPSK:



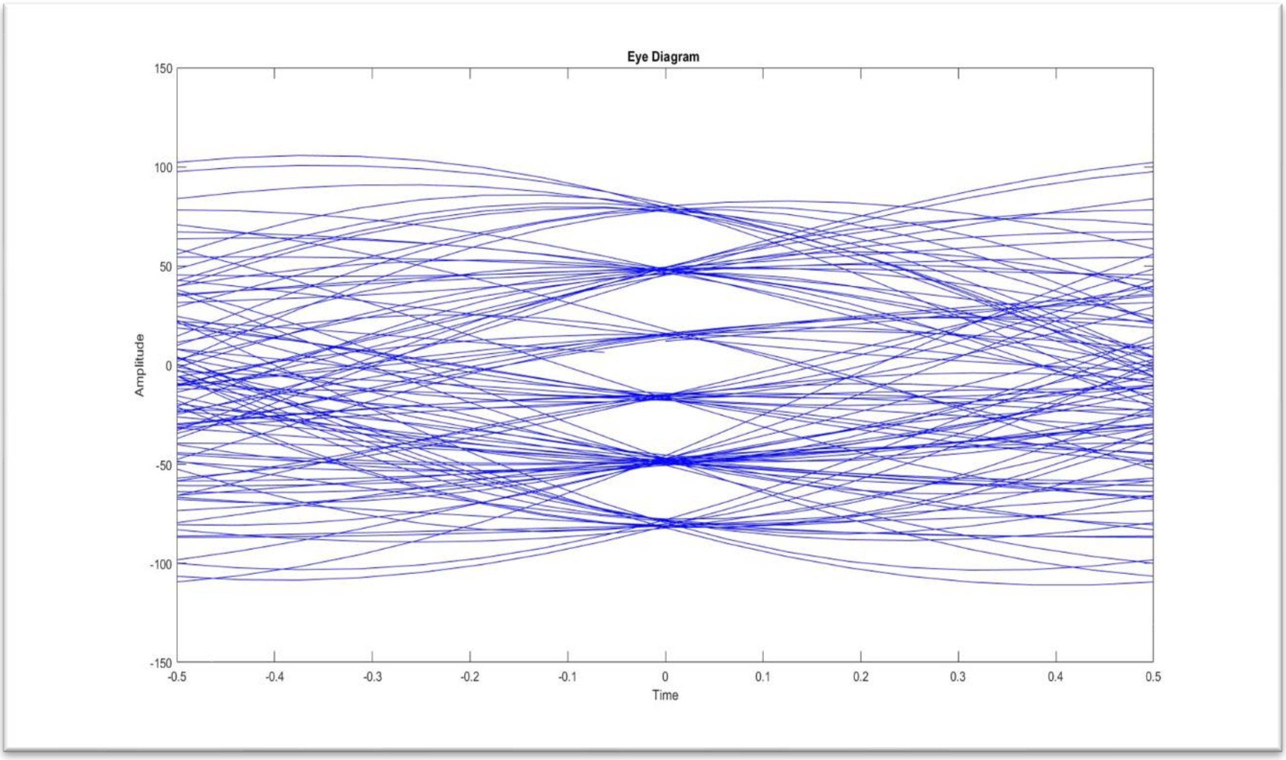
ii. 8-PSK:



iii.16-APSK:



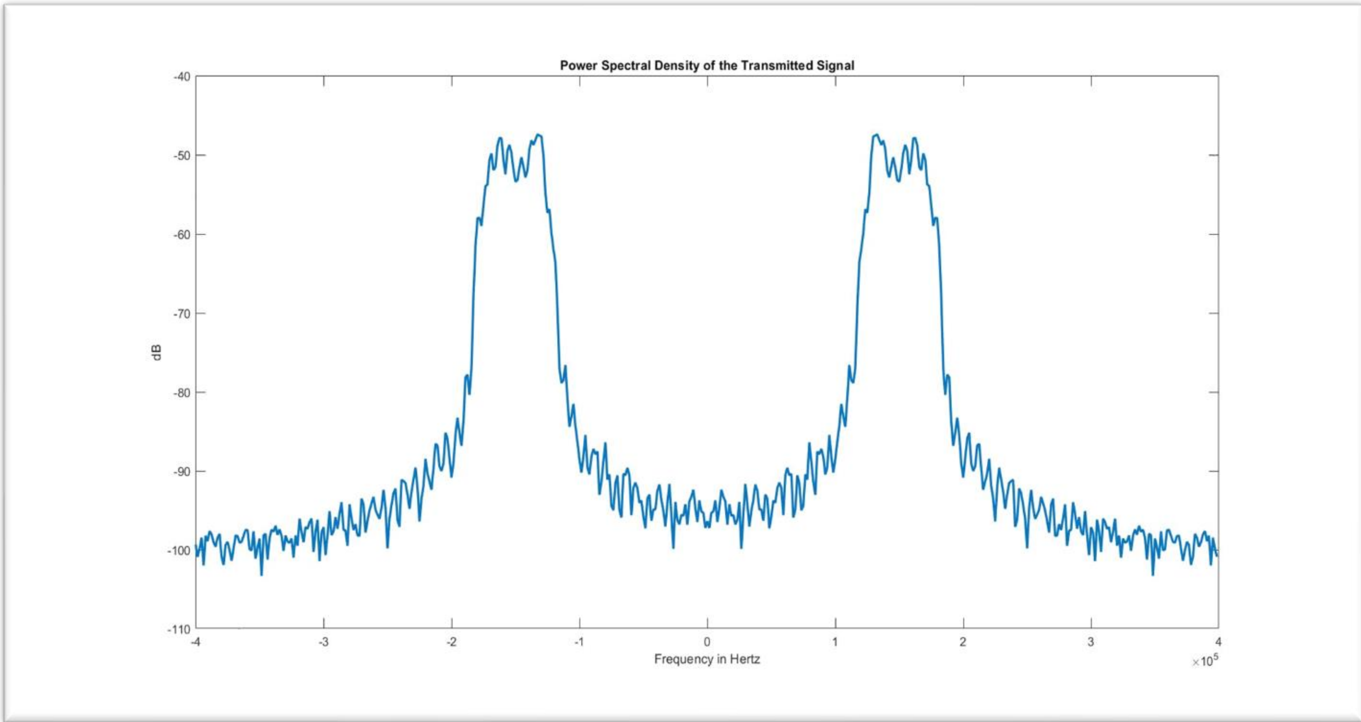
iv.32-QAM:



4. Radio frequency simulation with frequency upconversion and downconversion for QPSK signal:

- A widely popular scheme for transmission of communication messages uses sinusoidal carrier waveforms
- The inphase and the quadrature branches' output after the pulse shaping at the transmitter are given to this RF simulator.
- This simulator converts this signal to Quadrature notation given in ppt.

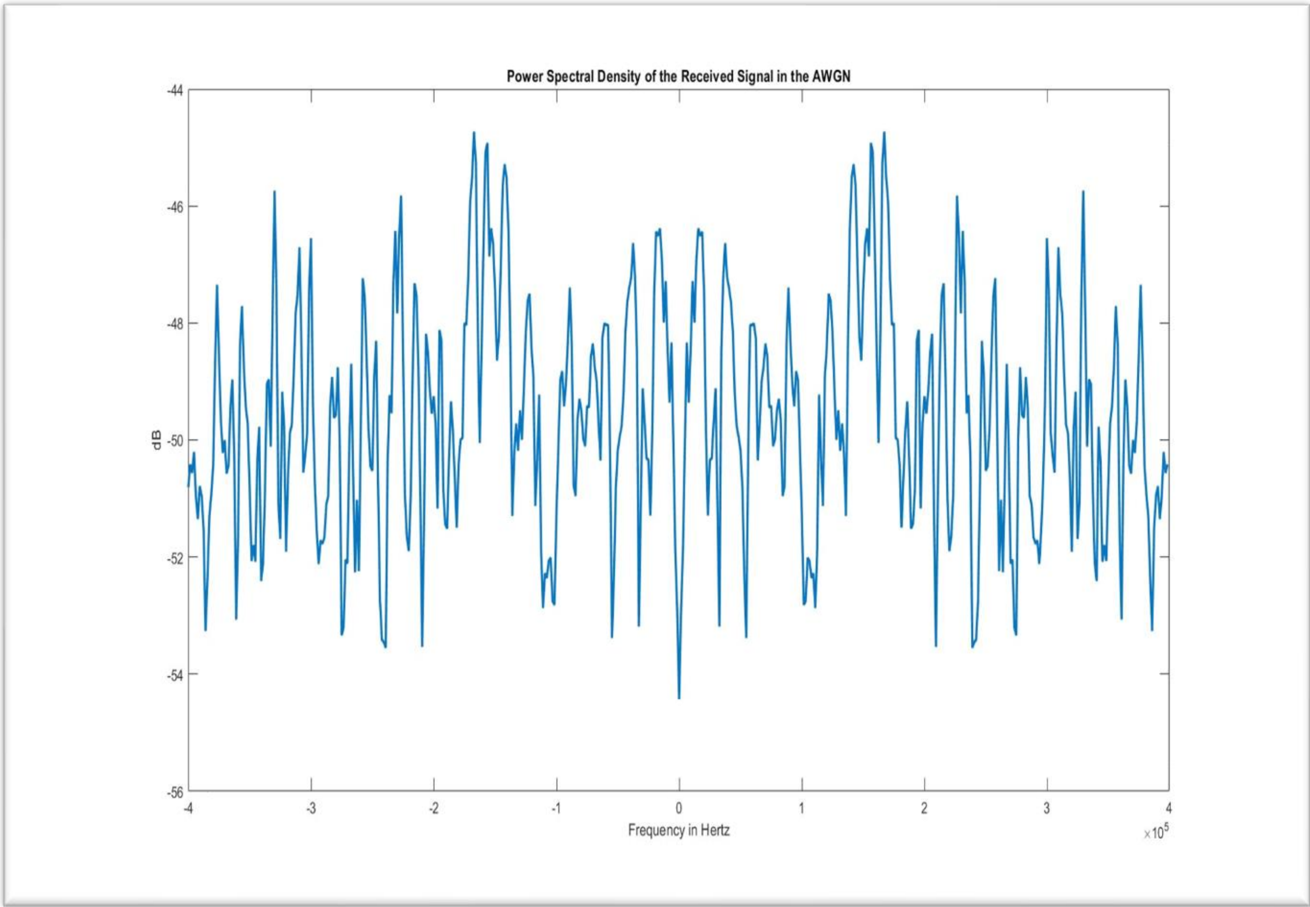
Here, I got power spectral density of the transmitted signal as follows:



Here, Carrier frequency is 150 kHz.

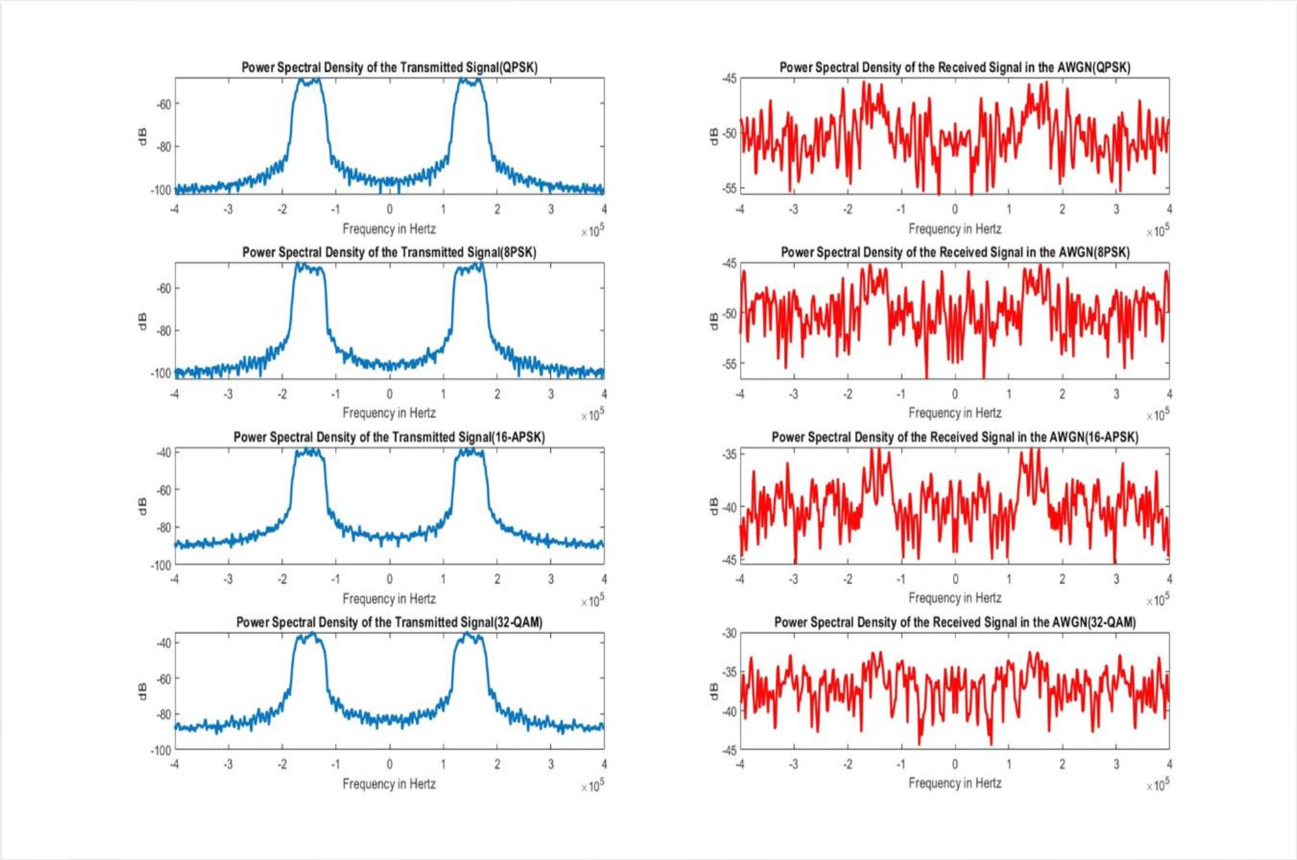
- We got conjugate symmetric spectrum because a real valued time domain signal has conjugate symmetric spectrum.
- Also, real valued-ness in one domain translates to symmetry in another domain. Therefore, we got mirror image at f=0 Hz.

Now, I added AWGN channel to it and got the power spectral density of the received signal in the presence of the AWGN.

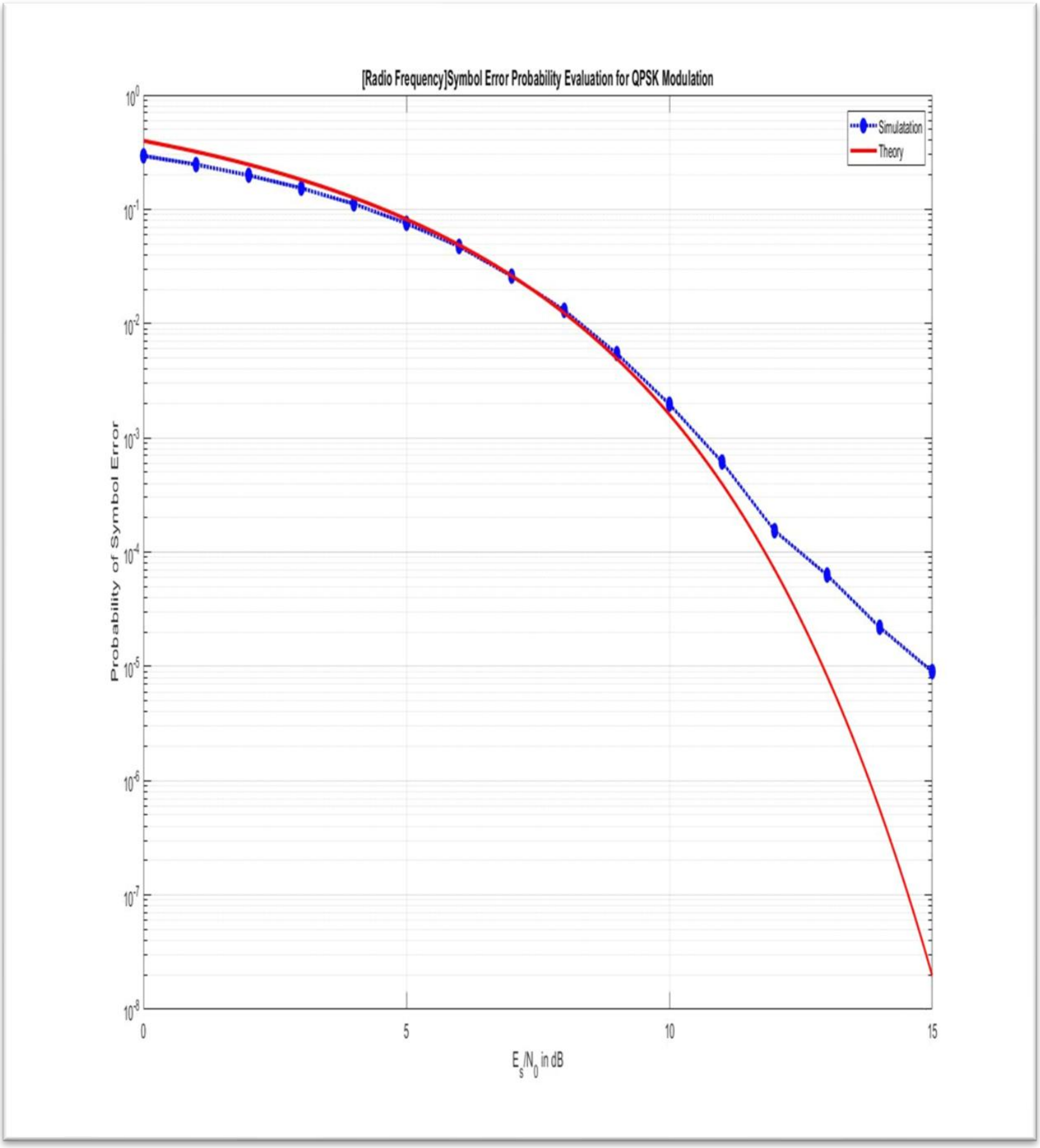


- Here also we got mirror image at f=0 Hz which is symmetric.

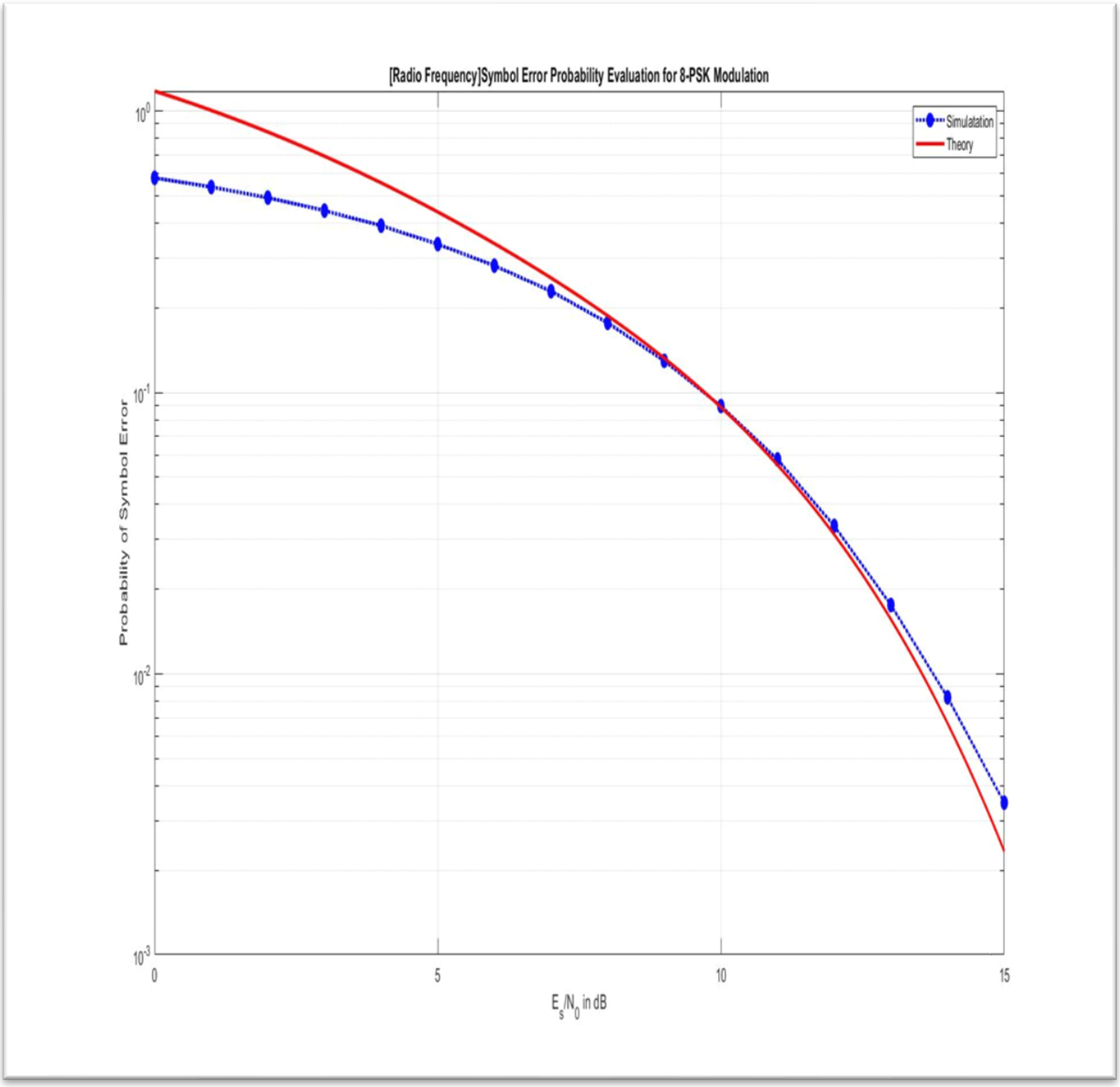
- **For all modulation schemes:**



6. For the QPSK simulator at Radio Frequency (RF) with the pulse-shaping, a result showing the symbol error probability at the output of the Monte Carlo simulator and the corresponding analytical (UUB) result:



- **For the 8-PSK simulator at Radio Frequency (RF) with the pulse-shaping, a result showing the symbol error probability at the output of the Monte Carlo simulator and the corresponding analytical (UUB) result:**



❖ Appendix:

- **Code Snippets:**https://drive.google.com/drive/folders/1o-mx3KxTZhEkSCepy8HTgxOKaxI4NeJF?usp=sharing