Quadrature Phase Shift Keying

A Project Report submitted by

Prakhar Agrawal 2023BTECH057

Priyanshu Kumar 2023BTECH062

under the supervision of

Mr. Divanshu Jain

In partial fulfillment of

the requirements for the course of

EE1123 Digital Communication

to the

Institute of Engineering & Technology (IET)

JK Lakshmipat University, Jaipur

Introduction:

Quadrature Phase Shift Keying (QPSK) is a widely used digital modulation technique designed to transmit information efficiently over communication channels that are limited by bandwidth and affected by noise. It encodes data by varying the phase of a sinusoidal carrier and assigns two bits to each symbol, which allows higher spectral efficiency compared to simpler schemes such as Binary Phase Shift Keying. By representing information through discrete phase shifts, QPSK provides a balanced combination of robustness and bandwidth utilization suitable for both wired and wireless systems.

QPSK is implemented using in-phase and quadrature components, which enables practical realization through mixers, oscillators, and baseband processing. Its resilience to common channel impairments and its moderate implementation complexity have led to widespread use in modern communication standards, including Wi-Fi, LTE, 5G, and satellite links. The technique also serves as a foundation for more advanced modulation methods, making it an essential topic in the study of digital communications.

Implementation of QPSK:

To implement a complete Quadrature Phase Shift Keying (QPSK) communication system. This lab aims to generate a QPSK signal by modulating a binary data stream onto a carrier wave and then recover the original data by demodulating the signal. The process will be verified by analyzing the modulated waveform and comparing the input and output bit sequences.

Theory:

As with binary PSK, information about the message symbols in QPSK is contained in the carrier phase. In particular, the phase of the carrier takes on one of four equally spaced phase values: π/4, 3π/4, 5π/4, and 7π/4.

- For this set of values, we may define the transmitted signal as:

$$S_i(t) = \begin{cases} \sqrt{\frac{2E}{T}} cos\left[2\pi f_c t + (2i-1)\frac{\pi}{4}\right], & \begin{cases} 0 \le \mathrm{t} \le \mathrm{T} \\ \mathrm{i} = 1,2,3,4 \end{cases} \\ & 0,\ elsewhere \end{cases}$$

- Where E is the transmitted signal energy per symbol and T is the symbol duration.
- Each possible value of the phase corresponds to a unique di-bit (i.e., a pair of bits).
- For example, we may choose the set of phase values to represent the Gray-Encoded set of de-bits: **10, 00, 01, 11**, where only one single bit is changed from one di-bit to the next.

Signal Space Representation (Constellation) of QPSK:

Time Interval	Signal ($S_i(t)$)
$0 \le t \le T$	$\sqrt{\frac{2E}{T}} cos\left[2\pi f_c t + (2i-1)\frac{\pi}{4}\right]$, i = 1,2,3,4
Elsewhere	0

The "coordinates" of the signal in the constellation diagram are determined by the coefficients s_{i1} and s_{i2}:

$$s_{i1} = \sqrt{E}\cos(\theta_i) \quad \text{and} \quad s_{i2} = \sqrt{E}\sin(\theta_i)$$

For the standard phase shifts of $\pm\pi/4$ and $\pm 3\pi/4$, the coordinates for the four symbols become:

Symbol (Bits)	Phase (θi)	Coordinates (s_{i1},s_{i2})
11	$\pi/4$ (45°)	$\left(+\sqrt{E/2}, +\sqrt{E/2}\right)$
01	$3\pi/4$ (135°)	$\left(-\sqrt{E/2}, +\sqrt{E/2}\right)$
00	$5\pi/4$ (225°)	$\left(-\sqrt{E/2}, -\sqrt{E/2}\right)$
10	$7\pi/4$ (315°)	$\left(+\sqrt{E/2}, -\sqrt{E/2}\right)$

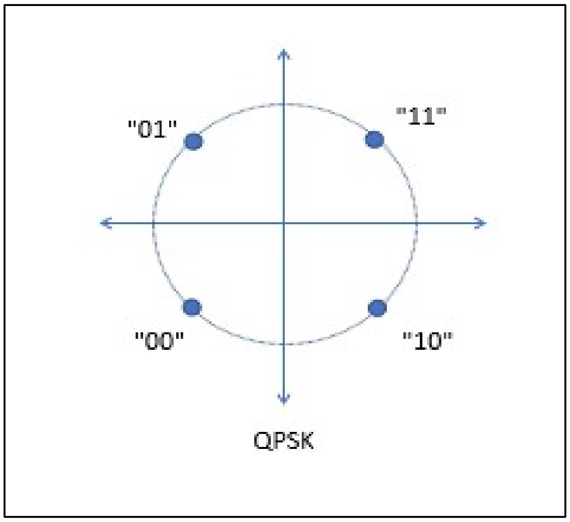

Figure 1: Constellation Diagram of QPSK

Quadrature Phase Shift Keying (QPSK) modulation process:

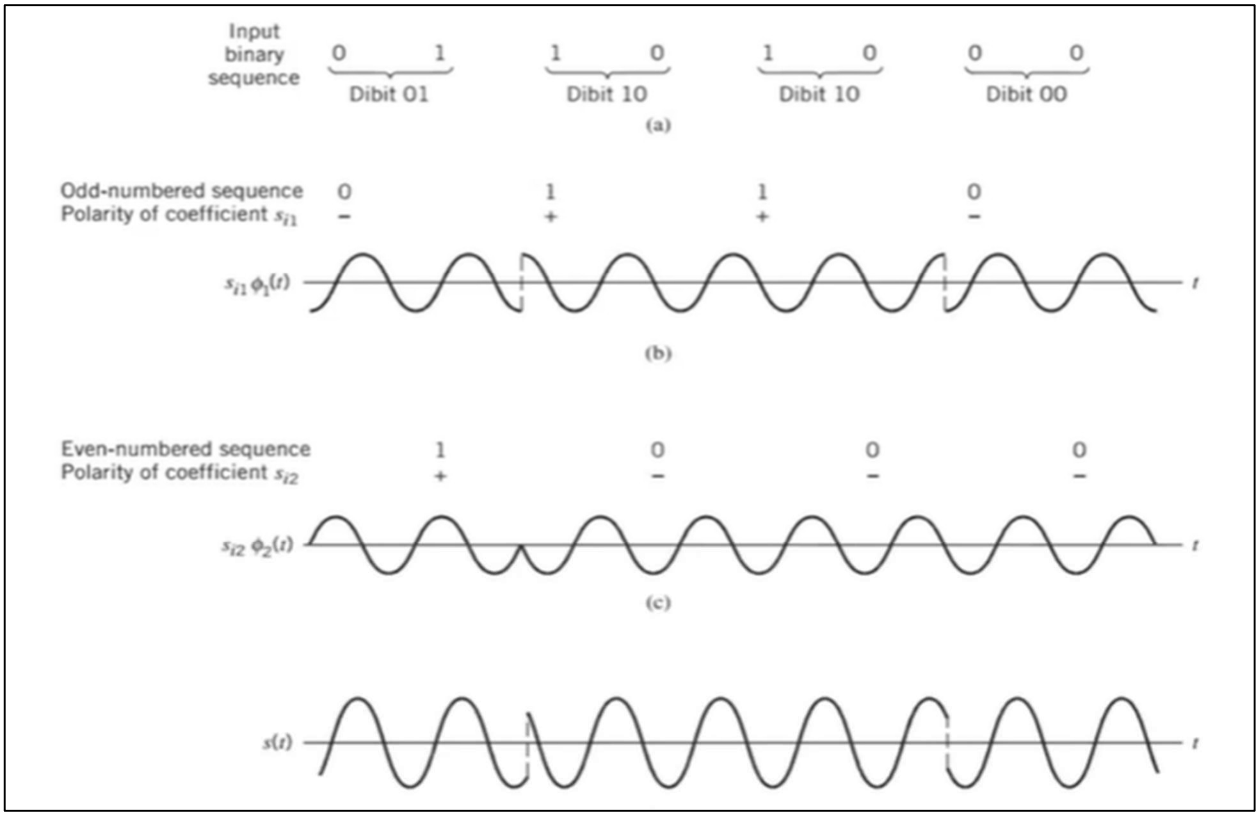

Figure 2: Example Sequence of QPSK modulation

This image illustrates Quadrature Phase Shift Keying (QPSK) modulation, which encodes digital data by varying the phase of a carrier signal. Here's how the process works based on the diagram:

1. **Input Data Grouping**
   The input bit sequence (shown at top) is divided into **di-bits** - pairs of consecutive bits. In the example: 01, 10, 10, 00. Each di-bit represents one symbol that will be transmitted.
2. **Parallel Signal Generation**
   The di-bits are split into two parallel streams based on bit position:
   a) **Odd-numbered sequence ($s_{i1}\varphi_1(t)$)** - Contains the first bit of each di-bit (positions 1, 3, 5, 7...)

      - Shows polarities: 0, 1, 1, 0 corresponding to the first bits

   b) **Even-numbered sequence ($s_{i2}\varphi_2(t)$)** - Contains the second bit of each di-bit (positions 2, 4, 6, 8...)

      - Shows polarities: 1, 0, 0, 0 corresponding to the second bits

3. **Quadrature Modulation**

Each sequence modulates a carrier wave, but the two carriers are **90° out of phase** (in quadrature):

- The odd sequence modulates the in-phase (I) component
- The even sequence modulates the quadrature (Q) component
- Binary 1 and 0 map to opposite polarities of the sinusoidal carrier

4. **Signal Combination**

The final QPSK signal **s(t)** (bottom waveform) is the sum of these two quadrature components. The resulting signal can have **four possible phase states** (0°, 90°, 180°, 270°), with each phase representing one unique di-bit combination (00, 01, 10, 11). This approach doubles the data rate compared to binary PSK since each symbol carries 2 bits of information instead of 1.

Transmitter-Receiver Architecture:

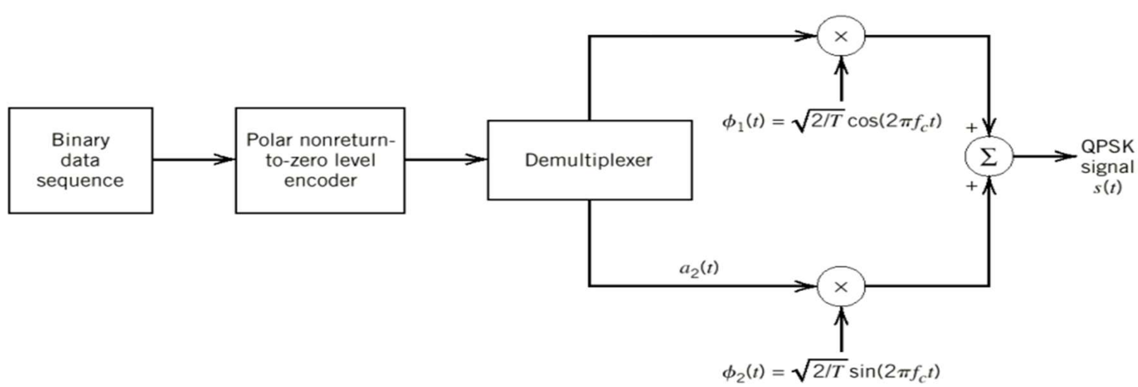

Figure 3: QPSK Transmitter

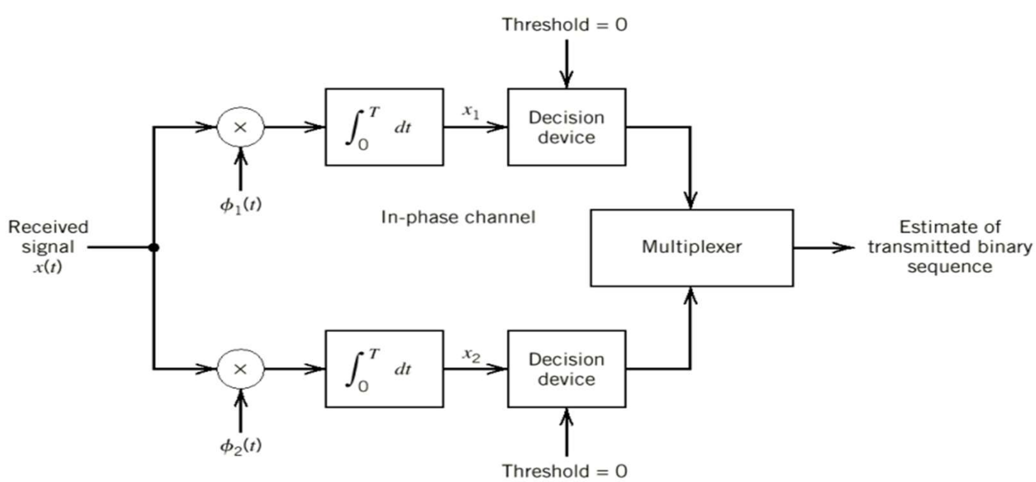

Figure 4: QPSK Receiver

Bit Error Rate (BER) Analysis of QPSK in AWGN Channel

1. Mathematical Representation of the Transmitted Signal

The analysis of QPSK performance in an Additive White Gaussian Noise (AWGN) channel begins with the time-domain definition of the signal. A QPSK signal modulates the phase of the carrier to one of four equiprobable values. The transmitted signal $s(t)$ is defined as:

$$s(t) = \sqrt{\frac{2E_s}{T_s}} \cos(2\pi f_c t + \phi_i), \quad i = 1,2,3,4$$

Where:

- E_s is the energy per symbol.
- T_s is the symbol duration.
- f_c is the carrier frequency.
- ϕ_i represents the phase corresponding to the input bits (typically $\frac{\pi}{4}, \frac{3\pi}{4}, \frac{5\pi}{4}, \frac{7\pi}{4}$).

Using trigonometric identities, this signal can be decomposed into two orthogonal components: the In-phase (I) and Quadrature (Q) components. This allows the signal to be expressed in Canonical Form:

$$s(t) = I(t)\cos(2\pi f_c t) - Q(t)\sin(2\pi f_c t)$$

This decomposition reveals that a QPSK signal is effectively two independent Binary Phase Shift Keying (BPSK) signals transmitted simultaneously on orthogonal carriers ($\cos$ and sin).

2. Euclidean Distance Analysis

The noise immunity of the system is determined by the Euclidean distance between constellation points. In the signal space diagram, the minimum distance (d) between adjacent points determines the probability of a decision error.

For QPSK, the minimum Euclidean distance is given by:

$$d = 2\sqrt{P_s T_s} \sin\left(\frac{\pi}{4}\right)$$

Since the symbol duration T_s transmits 2 bits, we have the relationship $T_s = 2T_b$ and $E_s = 2E_b$. Substituting these values, the distance can be expressed in terms of bit parameters:

$$d = 2\sqrt{P_s T_b}$$

Crucially, this distance is identical to that of a BPSK system with the same energy per bit (E_b).

3. Probability of Bit Error (BER)

Because the In-phase and Quadrature components are orthogonal, the noise in one channel does not affect the detection of the other. Therefore, the Bit Error Rate (BER) for QPSK is calculated by analyzing one of the constituent BPSK streams.

The probability of bit error (P_b) in an AWGN channel is a function of the distance d and the noise power spectral density N_0. Using the standard BPSK error probability formula:

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$$

Expressing this using the Complementary Error Function (erfc), we arrive at the final expression for QPSK BER:

$$P_b = \frac{1}{2}\text{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$$

4. **Probability of Symbol Error (SER)**

While the Bit Error Rate describes the reliability of individual bits, the Symbol Error Rate (P_s) describes the probability that the received symbol (pair of bits) is incorrect. A symbol error occurs if either the In-phase bit or the Quadrature bit (or both) are received in error.

For sufficiently high Signal-to-Noise Ratios (SNR), the SER is approximated as:

$$P_s \approx 2Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$$

5. **Conclusion**

The analysis demonstrates that the Bit Error Rate of QPSK is identical to that of BPSK for the same E_b/N_0:

$$BER_{QPSK} = BER_{BPSK}$$

However, since QPSK transmits two bits per symbol ($T_s = 2T_b$), it achieves twice the bandwidth efficiency of BPSK. Consequently, QPSK is often preferred in bandwidth-constrained applications as it doubles the data rate without compromising error performance.

Probability of error for QPSK System Derivation:

1. **QPSK Signal Transmission:**
   The transmitted QPSK signal is written on orthogonal carriers:

$$s(t) = \sqrt{\frac{2E_s}{T}}\left(I\cos(2\pi f_c t) - Q\sin(2\pi f_c t)\right)$$

   where $I, Q \in \left\{+\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{2}}\right\}$

   - E_s = energy per symbol
   - T = symbol duration

- I = in-phase component (carries 1 bit)
- Q = quadrature component (carries 1 bit)

2. **Matched Filter Reception:**
   The received signal is: $r(t) = s(t) + n(t)$
   where $n(t)$ is additive white Gaussian noise (AWGN).
   Project onto orthogonal basis functions:
   In-phase correlator: $r_I = \int_0^T r(t)\cos(2\pi f_c t)\,,dt$
   Quadrature correlator: $r_Q = -\int_0^T r(t)\sin(2\pi f_c t)\,,dt$
3. **Correlator Outputs:**
   Substituting $r(t) = s(t) + n(t)$:

$$r_I = s_I + n_I$$

$$r_Q = s_Q + n_Q$$

   where:

   - $s_I, s_Q \in \left\{\pm\sqrt{\frac{E_s}{2}}\right\}$ are the signal components
   - $n_I, n_Q \sim \mathcal{N}\left(0, \frac{N_0}{2}\right)$ are independent Gaussian noise components

4. **Decision Rule:**
   Threshold at 0 for sign detection:

$$\boldsymbol{I} = \begin{cases} +\frac{\mathbf{1}}{\sqrt{\mathbf{2}}} \; \boldsymbol{if\ r_I > 0} \\ -\frac{\mathbf{1}}{\sqrt{\mathbf{2}}} \; \boldsymbol{if\ r_I < 0} \end{cases}$$

$$\boldsymbol{Q} = \begin{cases} +\frac{\mathbf{1}}{\sqrt{\mathbf{2}}} \; \boldsymbol{if\ r_Q > 0} \\ -\frac{\mathbf{1}}{\sqrt{\mathbf{2}}} \; \boldsymbol{if\ r_Q < 0} \end{cases}$$

5. **Conditional Error Probability (I-channel):**
   Given $s_I = +\sqrt{E_s/2}$, an error occurs when $r_I < 0$:

$$P_{I|+} = P(r_I < 0) = P\left(n_I < -\sqrt{\frac{E_s}{2}}\right)$$

Since $n_I \sim \mathcal{N}(0, N_0/2)$, we use its probability density function:

$$P_{I|+} = \int_{-\infty}^{-\sqrt{E_s/2}} \frac{1}{\sqrt{\pi N_0}} \exp\left(-\frac{n^2}{N_0}\right) dn$$

6. **Standardize the Integral:**
   Let $x = \frac{n}{\sqrt{N_0/2}}$, then:

$$P_{I|+} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{-\sqrt{E_s/N_0}} e^{-x^2/2}\, dx$$

This is expressed using the Q-function:

$$P_{I|+} = Q\left(\sqrt{\frac{E_s}{N_0}}\right)$$

where $Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-t^2/2} dt$

7. **Convert to Bit Energy:**
   Since QPSK transmits 2 bits per symbol:

$$E_s = 2E_b$$

   Substituting:

$$P_I = Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$$

This is identical to BPSK bit error probability.

8. **Symbol Error Probability:**
   A symbol error occurs if the I-channel is wrong OR the Q-channel is wrong (or both).
   Since I and Q are independent:

$$P_s = P(\text{I wrong or Q wrong})$$
$$P_s = 1 - P(\text{I correct and Q correct})$$
$$P_s = 1 - (1 - P_I)^2$$
$$P_s = 2P_I - P_I^2$$

Expanding: $P_s = 2Q\left(\sqrt{\frac{E_s}{N_0}}\right) - Q^2\left(\sqrt{\frac{E_s}{N_0}}\right)$

$$P_s = 2Q\left(\sqrt{\frac{2E_b}{N_0}}\right) - Q^2\left(\sqrt{\frac{2E_b}{N_0}}\right)$$

9. **Bit Error with Gray Mapping:**

With Gray mapping, adjacent QPSK symbols differ by only one bit.
When a symbol error occurs:

- Most likely error is to the nearest neighbor
- This causes only 1 bit error out of 2 bits

Therefore, the bit error probability is:

$$\boxed{P_b = P_I = \mathrm{Q}\left(\sqrt{\frac{2E_b}{N_0}}\right)}$$

10. Alternative Forms:

Using the complementary error function:

$$Q(x) = \frac{1}{2}\mathrm{erfc}\left(\frac{x}{\sqrt{2}}\right)$$

Therefore:

$$P_b = \frac{1}{2}\mathrm{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right)$$

11. High SNR Approximation:

For large x, the Q-function can be approximated:

$$Q(x) \approx \frac{1}{\sqrt{2\pi}}\frac{e^{-x^2/2}}{x}$$

Thus, for high SNR:

$$P_b \approx \frac{1}{\sqrt{4\pi E_b/N_0}}\exp\left(-\frac{E_b}{N_0}\right)$$

Key Result: $\boxed{BER_{QPSK} = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) = BER_{BPSK}}$

Why QPSK has the same BER as BPSK:

1. Each dimension (I and Q) operates independently.
2. Each dimension has the same geometry as BPSK.
3. Gray mapping ensures most symbol errors cause only single-bit errors.

4. QPSK achieves 2× spectral efficiency with no BER penalty.

MATLAB Simulation for QPSK Modulation:

Code:

```
clear; clc; close;
Fc = 1e6;          // carrier frequency
Tb = 1e-6;         // bit duration
Fs = 100*Fc;       // sampling frequency
t_symbol = 0:1/Fs:Tb-1/Fs;   // time for 1 symbol
bits = [0 1 1 0 1 1 0 0];   // user input bits
N = length(bits);
// convert bits → bipolar form (0→-1 , 1→+1)
msg = 2*bits - 1;
I_bits = msg(1:2:$);   // odd bits
Q_bits = msg(2:2:$);   // even bits
////   MODULATION
I_wave = [];
Q_wave = [];
QPSK_wave = [];
for k = 1:length(I_bits)
    Ik = I_bits(k) * cos(2*%pi*Fc*t_symbol);
    Qk = Q_bits(k) * sin(2*%pi*Fc*t_symbol);
    I_wave = [I_wave Ik];
    Q_wave = [Q_wave Qk];
    QPSK_wave = [QPSK_wave (Ik + Qk)];
end
// time axis for full signal
t = 0:1/Fs:(length(I_wave)-1)/Fs;
////PLOTTING
clf();
subplot(4,1,1);
```

```
plot(0:1/Fs:4*Tb, cos(2*%pi*Fc*(0:1/Fs:4*Tb)));
title("Unmodulated COSINE carrier wave");
xlabel("time(sec)"); ylabel("amplitude"); xgrid();
subplot(4,1,2);
plot(t, I_wave);
title("Waveform for in-phase component");
xlabel("time(sec)"); ylabel("amplitude"); xgrid();
subplot(4,1,3);
plot(t, Q_wave);
title("Waveform for quadrature component");
xlabel("time(sec)"); ylabel("amplitude"); xgrid();
subplot(4,1,4);
plot(t, QPSK_wave);
title("Modulated QPSK signal");
xlabel("time(sec)"); ylabel("amplitude"); xgrid();
//// add bit-pair labels on modulated signal
for k = 1:length(I_bits)
   text((k-0.5)*Tb, max(QPSK_wave)*0.8, string(bits(2*k-1)) + string(bits(2*k)));
end
```

Output:

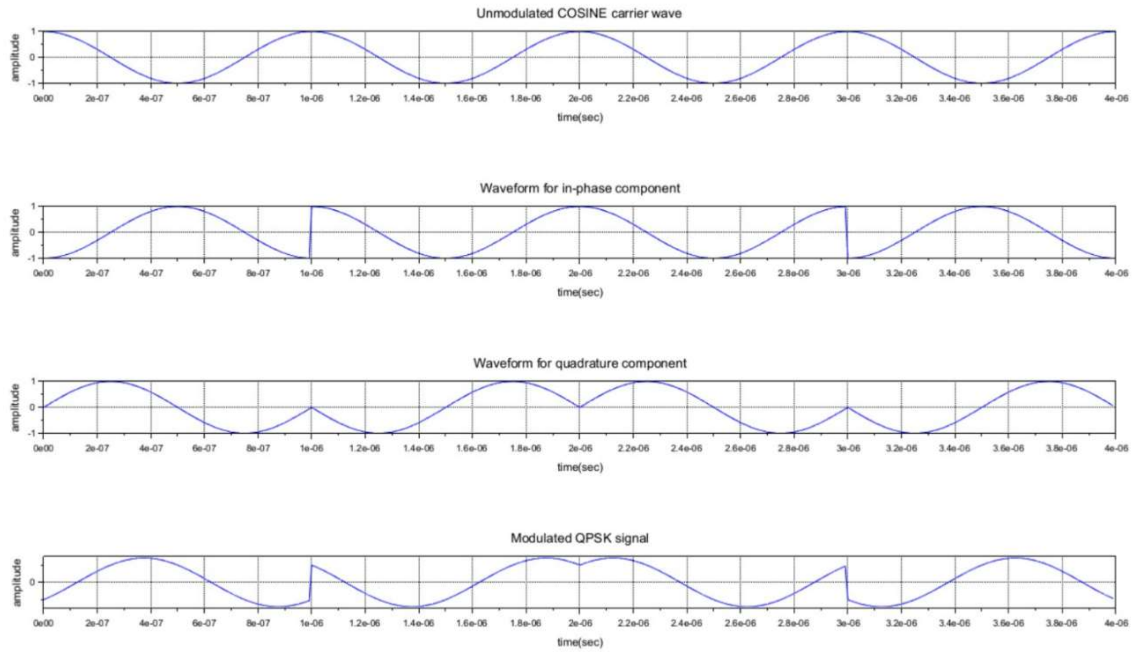

Figure 5: MATLAB output waveform of QPSK modulation

The first graph shows the odd-bit sequence mapped onto the cosine basis signal $\varphi_1(t)$. The second graph shows the even-bit sequence mapped onto the sine basis signal $\varphi_2(t)$. Each bit pair (di-bit) controls the amplitudes of these two orthogonal waves. In the third graph, both components add together to form the final QPSK signal s(t). The sudden jumps in the waveform represent phase changes according to the input di-bits this is the key idea of QPSK modulation.

MATLAB Simulation of Theoretical vs Simulated BER for QPSK in AWGN:

MATLAB Code:

```
clear;
close all;
clc;

N = 1000000;            % Number of bits (large for accuracy)
EbN0dB = 0:1:15;        % SNR range
ber_sim = zeros(1,length(EbN0dB));
ber_theory = zeros(1,length(EbN0dB));

% ---------- QPSK Modulation ----------
% Generate random bits
bits = randi([0 1], 1, N);

% Group bits into pairs for QPSK
bits_I = bits(1:2:end);     % odd bits → I component
bits_Q = bits(2:2:end);     % even bits → Q component

% Map: 0→-1 , 1→+1  (BPSK mapping on both I and Q)
symbols_I = 2*bits_I - 1;
symbols_Q = 2*bits_Q - 1;

% Form QPSK symbols
s = symbols_I + 1j*symbols_Q;

% ---------- Loop over SNR ----------
for k = 1:length(EbN0dB)
   % Convert Eb/N0 to linear
   EbN0 = 10^(EbN0dB(k)/10);
   % For QPSK: Es = 2Eb  →  Noise variance:
   N0 = 1/EbN0;
   noise_sigma = sqrt(N0/2);
   % AWGN noise
   noise = noise_sigma*(randn(size(s)) + 1j*randn(size(s)));
   % Received signal
   r = s + noise;
   % Detection (decision on real and imag)
   bits_I_hat = real(r) > 0;
   bits_Q_hat = imag(r) > 0;
   % Combine detected bits
   bits_hat = zeros(1,N);
   bits_hat(1:2:end) = bits_I_hat;
   bits_hat(2:2:end) = bits_Q_hat;
   % BER calculation
   ber_sim(k) = sum(bits ~= bits_hat) / N;
```

```
    % ---------- QPSK BER Theory ----------
    ber_theory(k) = 0.5 * erfc( sqrt(EbN0) );
end

% ---------- Plot ----------
semilogy(EbN0dB, ber_sim, 'r*-', 'LineWidth', 1.5); hold on;
semilogy(EbN0dB, ber_theory, 'ko-', 'LineWidth', 1.5);
grid on;
xlabel('Eb/N0 (dB)');
ylabel('Bit Error Rate (BER)');
legend('Simulated BER', 'Theory (erfc)');
title('QPSK BER: Simulation vs Theory (AWGN)');
```

Output:

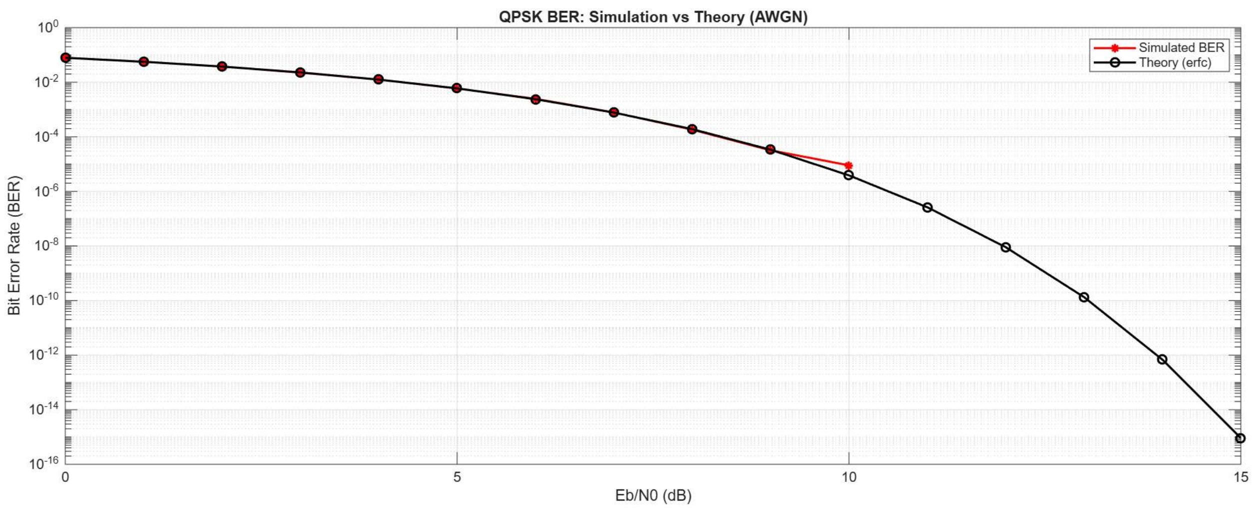

Figure 6: MATLAB BER Simulation of QPSK

Explanation:

This graph compares the simulated BER of QPSK with the theoretical BER over an AWGN channel. As Eb/N0 increases, the noise effect reduces, so both curves show a decreasing BER. The simulated red curve closely matches the theoretical black curve, which proves that our QPSK modulation, detection, and noise model are implemented correctly. The small difference at higher SNR is due to random noise and finite simulation size. Overall, the graph confirms that QPSK performance follows theory very accurately.

QPSK Simulation Using Simulink:

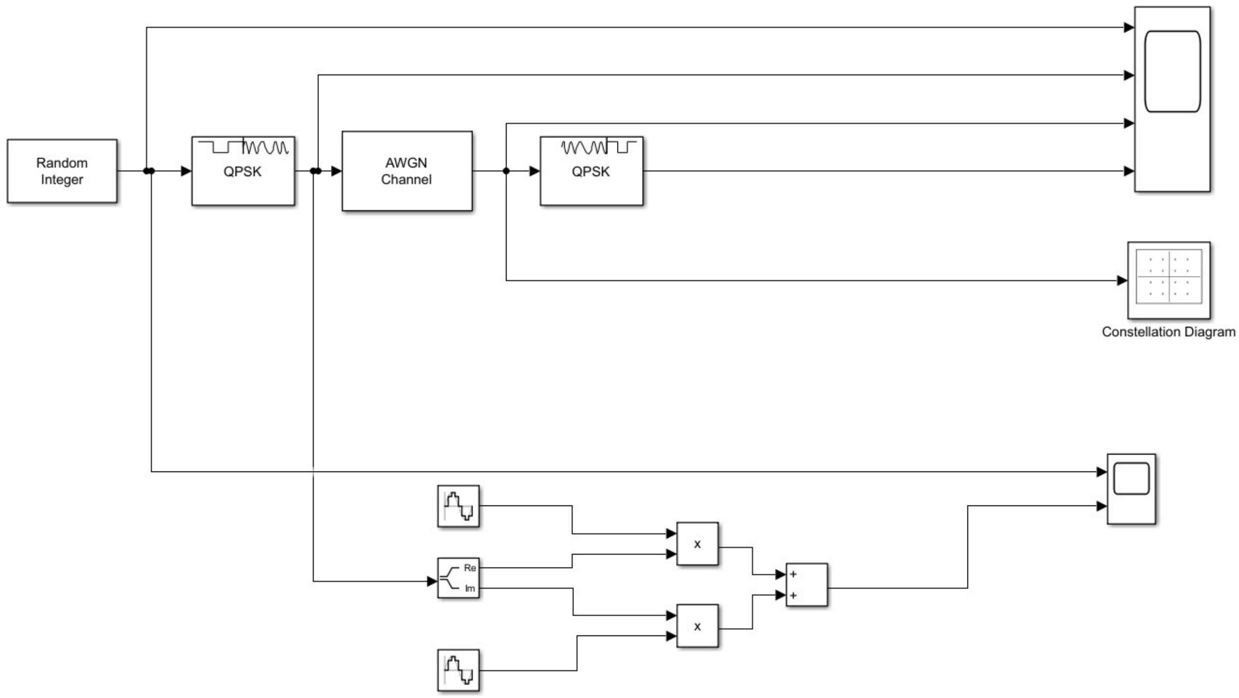

Figure 7: Simulink model of QPSK

Explanation of the Simulink Model components:

Component	Output	Purpose
Random Integer Generator	Integer symbols (0–3)	Generates random data symbols for QPSK.
QPSK Modulator Baseband	Complex baseband signal (I + jQ)	Maps symbols to QPSK constellation points.
AWGN Channel	Noisy complex signal	Adds channel noise based on Eb/No.
QPSK Demodulator Baseband	Recovered symbols (0–3)	Converts received constellation points back to data.
Scope (Bits Comparison)	Display of transmitted vs received symbols	Verifies correctness of modulation and demodulation.
Complex to Real-Imag	I (real) and Q (imag) components	Splits the complex baseband signal into I/Q paths.

Sine Wave Generator	sin(ωt)	Carrier for the I branch.
Sine Wave Generator (phase shifted)	cos(ωt)	Carrier for the Q branch.
Product (Multiplier) – I branch	I·sin(ωt)	Modulates the I component with the sine carrier.
Product (Multiplier) – Q branch	Q·cos(ωt)	Modulates the Q component with the cosine carrier.
Add Block	QPSK passband waveform	Combines I and Q components into final modulated signal.
Scope (Waveform)	Passband waveform display	Shows phase shifts in the QPSK carrier.
Constellation Diagram	Scatter plot of received symbols	Visualizes the constellation and noise impact.

Scope (Waveform) Output from Simulink:

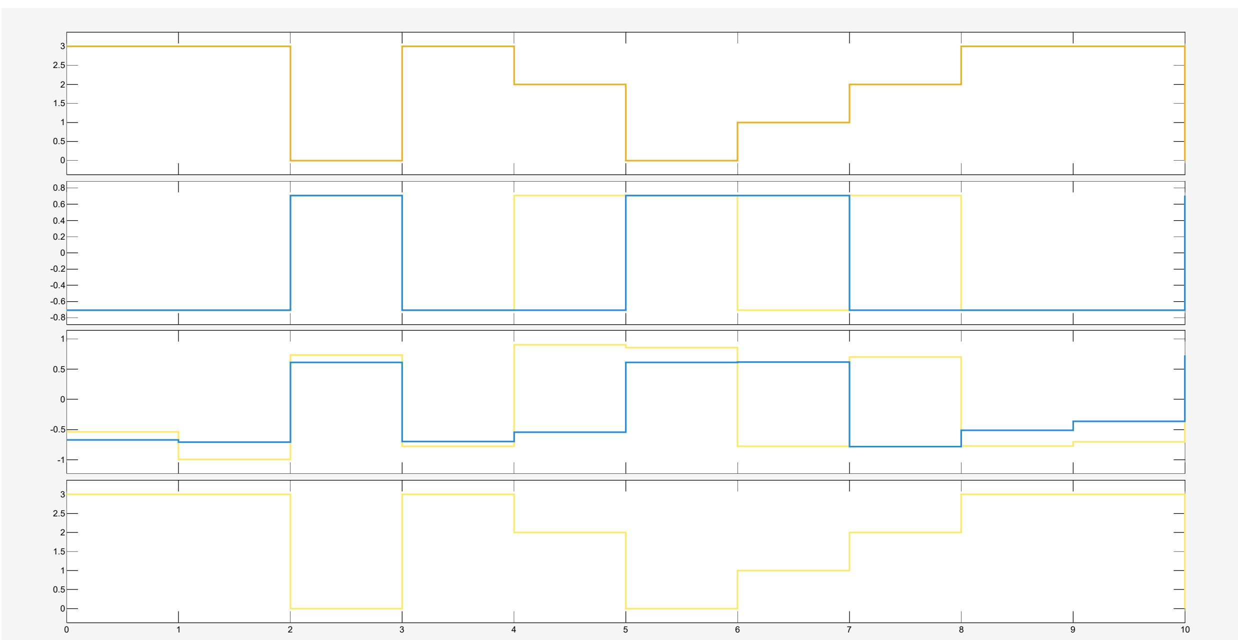

Figure 8: Scope 1 Output

This Scope-1 plot explains:

Top plot: Input data stream (2-bit symbols: 00, 01, 10, 11 → levels 0, 1, 2, 3)

Middle plots: Separated into I-channel (In-phase) and Q-channel (Quadrature) bits

Bottom plot: Reconstructed symbols or received data

Here, we can see that input and received data streams match perfectly which means that our Simulink model works correctly.

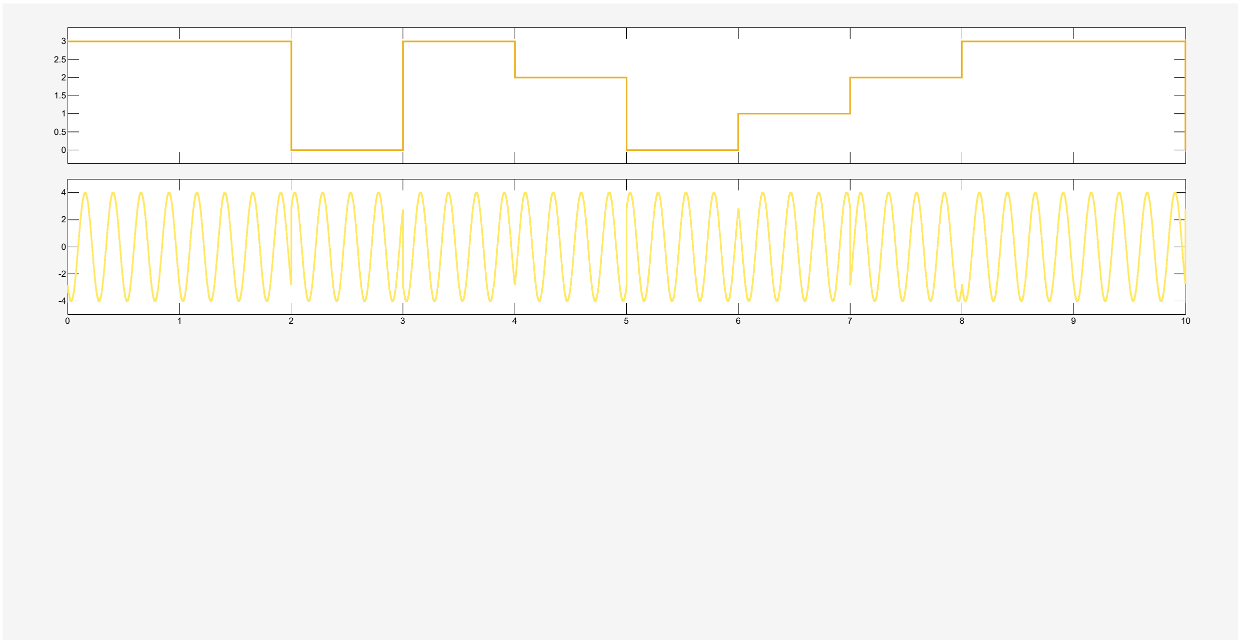

Figure 9: Scope 2 Output

The plot of Scope-2 tells the input and modulated signal using QPSK Simulink model.

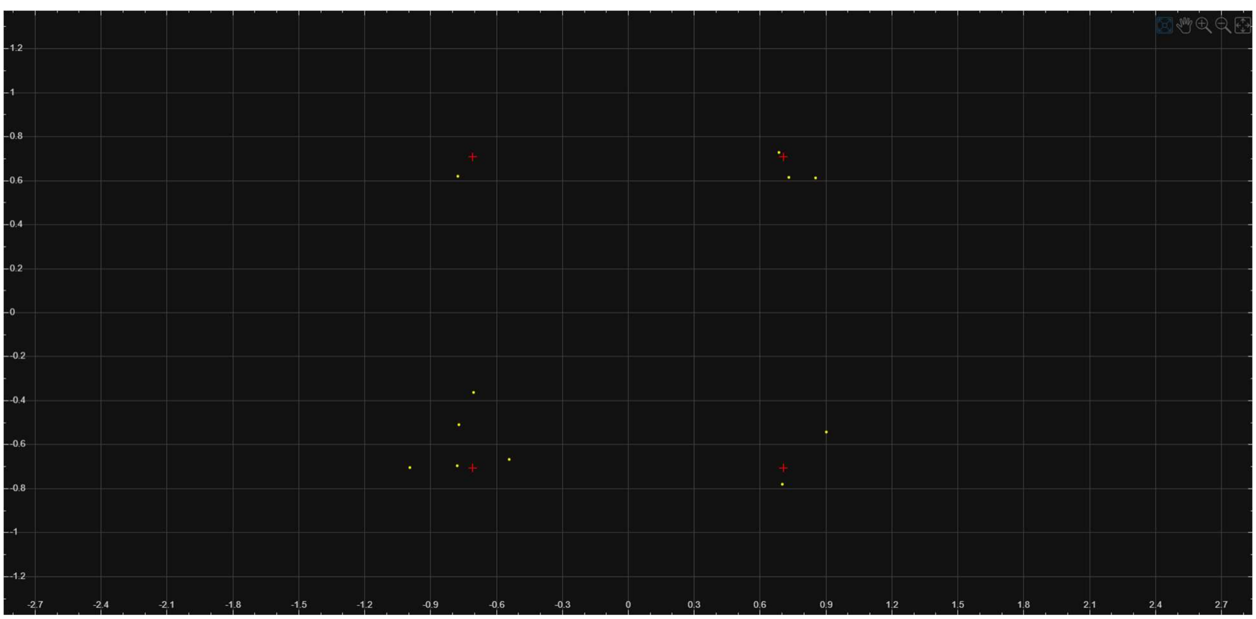

Figure 10: Constellation Diagram Output from Simulink

The constellation diagram shows four clusters, one for each QPSK symbol. The red dots are the ideal points where the symbols should land, and the yellow dots are the actual received points after noise is added by the AWGN channel. The yellow dots are scattered around each red point because of noise. The slight tilt or uneven spread means there might be small phase or scaling errors, which is normal in passband QPSK systems without perfect correction. Overall, the plot shows that the QPSK modulation, transmission, and demodulation are working correctly.

Applications of QPSK:

- **Wi-Fi:** Used for reliable data transmission at lower modulation levels.
- **LTE / 5G:** Used in control channels and poor signal conditions for stable communication.
- **Satellite Communication:** Common in DVB-S, VSAT systems due to good performance at low SNR.
- **GPS / GNSS:** Used in modern navigation signals for better accuracy and interference resistance.
- **Cable Modems:** Used in DOCSIS upstream channels for stable transmission over noisy cables.

Conclusion:

This project successfully demonstrated the implementation and performance analysis of Quadrature Phase Shift Keying (QPSK). Through mathematical derivation and simulations in both MATLAB and Simulink, we verified that QPSK achieves double the bandwidth efficiency of BPSK while maintaining the exact same Bit Error Rate (BER) performance. The simulation results closely matched theoretical predictions, with constellation diagrams confirming effective symbol clustering even in the presence of AWGN noise. Ultimately, the project confirms that QPSK is a robust and spectral-efficient modulation technique, justifying its widespread adoption in modern communication standards such as LTE, 5G, and satellite networks.