Aim ↬ To perform QPSK [Quadrature Phase Shift Keying] using a trainer kit and verify by MATLAB.

Apparatus Required ↬ MATLAB.

Theory ↬

Quadrature Phase Shift Keying (QPSK) is a digital modulation technique where binary data is encoded into the phase changes of a carrier signal. Unlike Frequency Shift Keying (FSK) or Amplitude Shift Keying (ASK), where frequency or amplitude variations represent data, QPSK uses phase shifts to represent pairs of bits, making it highly bandwidth-efficient.

In QPSK, the carrier signal undergoes four distinct phase shifts to encode two bits per symbol:

Bit pair 00 is represented by a phase of 0°.

$$s(t) = Acos(2\pi f_c t + 0)$$

Bit pair 01 is represented by a phase of 90° (π/2).

$$s(t) = Acos(2\pi f_c t + \pi/2)$$

Bit pair 11 is represented by a phase of 180° (π).

$$s(t) = Acos(2\pi f_c t + \pi)$$

Bit pair 10 is represented by a phase of 270° (3π/2).

$$s(t) = Acos(2\pi f_c t + 3\pi/2)$$

This encoding allows QPSK to transmit twice as much data as Binary Phase Shift Keying (BPSK) over the same bandwidth, making it more efficient.

QPSK achieves higher spectral efficiency because it encodes two bits per symbol. The required bandwidth is approximately:

$$B \approx \frac{bit\ rate}{2}$$

This efficiency makes QPSK ideal for applications requiring high data throughput.

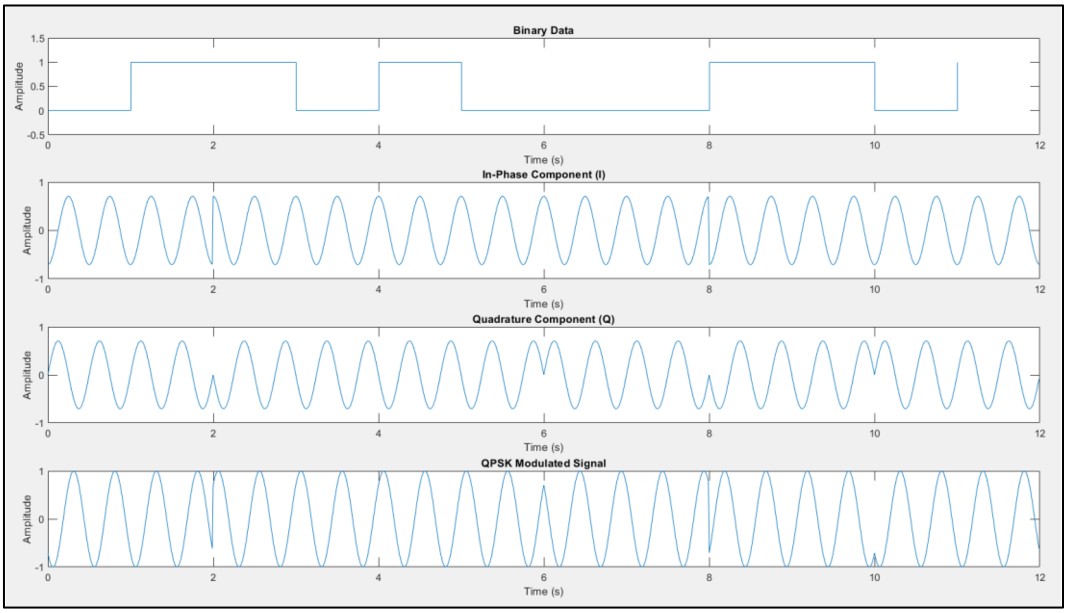
QPSK is robust against amplitude noise because it encodes data in phase changes. However, it is susceptible to phase noise and requires precise synchronization between the transmitter and receiver to ensure accurate demodulation.

QPSK is widely used in Satellite communication systems, Digital TV broadcasting, 4G LTE and Wi-Fi standards, RFID systems, etc. QPSK balances bandwidth efficiency, noise immunity, and implementation complexity, making it a cornerstone of modern digital communication systems.

Code ↬

```
%Quadrature Phase Shift Keying
clc;
clear;
close all;

Fs = 100;
f0 = 2;
bit_rate = 1;
Tb = 1/bit_rate;
Ts = 2 * Tb;
t = 0:1/Fs:Ts-1/Fs;

data = [0 1 1 0 1 0 0 0 1 1 0 1];
N = length(data);

if mod(N, 2) ~= 0
    error('Number of input bits must be even for QPSK.');
end

qpsk_signal = [];
I_component = [];
Q_component = [];

for i = 1:2:N
    bits = data(i:i+1);
    if isequal(bits, [0 0])
        phase = pi/4;
    elseif isequal(bits, [0 1])
        phase = 3*pi/4;
    elseif isequal(bits, [1 1])
        phase = 5*pi/4;
    elseif isequal(bits, [1 0])
        phase = 7*pi/4;
    end

    I = cos(phase);
    Q = sin(phase);

    I_signal = I * cos(2 * pi * f0 * t);
    Q_signal = Q * sin(2 * pi * f0 * t);
    qpsk_symbol = I_signal - Q_signal;
    qpsk_signal = [qpsk_signal qpsk_symbol];

    I_component = [I_component I_signal];
    Q_component = [Q_component Q_signal];
end

t_total = 0:1/Fs:Ts*(N/2)-1/Fs;

figure;
subplot(4,1,1);
stairs([0:N-1]*Tb, data);
xlabel('Time (s)');
ylabel('Amplitude');
title('Binary Data');
ylim([-0.5 1.5]);

subplot(4,1,2);
plot(t_total, I_component);
xlabel('Time (s)');
ylabel('Amplitude');
title('In-Phase Component (I)');

subplot(4,1,3);
plot(t_total, Q_component);
xlabel('Time (s)');
ylabel('Amplitude');
title('Quadrature Component (Q)');

subplot(4,1,4);
plot(t_total, qpsk_signal);
xlabel('Time (s)');
ylabel('Amplitude');
title('QPSK Modulated Signal');
```

Output ↬



Result ↬

The QPSK modulation was successfully demonstrated in MATLAB, where distinct phase shifts in the carrier signal effectively represented binary data pair.

Conclusion ↬

The experiment confirms that QPSK provides a robust and bandwidth-efficient way to modulate binary data by varying the phase, making it suitable for high-speed communication and resilient to amplitude-based noise.

Precautions ↬

- Ensure that the phase shifts (0°, 90°, 180°, 270°) are correctly implemented to represent the intended bit pairs.
- Verify that the binary data and modulation settings are accurately inputted into MATLAB for precise signal generation.
- Check the MATLAB plots for proper time and phase scales to visualize the modulated signal clearly.
- Synchronize the transmitter and receiver accurately to prevent errors in phase detection.