RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

NBA Accredited (UG - 6 Years)

hod.ec@rvce.edu.in
www.rvce.edu.in
Tel: 080-68188178 / 8179 (O)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

Academic Year: 2024-25

Principles of Digital Communication Systems (EC353AI)

Laboratory Manual and Observation Book

(Autonomous Scheme 2022)

Rashtreeya Sikshana Samithi Trust®

Go, change the world®

RV College of Engineering®
Mysore Road, RV Vidyaniketan Post,
Bengaluru - 560059, Karnataka, India

NBA Accredited (UG - 6 Years)

hod.ec@rvce.edu.in
www.rvce.edu.in
Tel: 080-68188178 / 8179 (O)

RV COLLEGE OF ENGINEERING®, Bengaluru-59

(Autonomous Institution affiliated to VTU, Belagavi)

Department of Electronics and Communication Engineering

Laboratory Certificate

This is to certify that Mr. / Ms ______________________________ ______________________has satisfactorily completed the course of Experiments in Principles of Digital Communication Systems lab (EC353AI) course prescribed by the Department during the year ____________

USN No.: ___________________ Semester: _________________

Marks	
Maximum	**Obtained**
50	

Marks in words

Signature of the staff in-charge

Head of the Department

Date:

RV COLLEGE OF ENGINEERING®, Bengaluru-59

(Autonomous Institution affiliated to VTU, Belagavi)

Department of Electronics and Communication Engineering

SCHEME OF CONDUCT AND EVALUATION

CLASS: 5th SEMESTER
YEAR: 2024-25

CIE MARKS (Max): 50
SEE MARKS (Max): 50
SEE: 03 Hrs

Expt No	Title	Page No	Duration in Hrs	Max Marks	Marks obtained	Staff signature
Cycle-1						
1	M-PSK transceiver Design using Labview		2	10		
2	M-QAM transceiver Design using Labview		2	10		
3	ISI with PAM modulation technique using MATLAB		2	10		
4	Equalization with QAM Modulation using Labview		2	10		
Cycle-2						
5	Transmitter and Receiver Implementation of Direct Sequence Spread Spectrum and Frequency Hopped Spread Spectrum using MATLAB		2	10		
6	Symbol Timing Estimation using MATLAB		2	10		
7	Frequency selective and non-selective fading channel simulation using MATLAB		2	10		
Cycle-3						
8	Carrier Phase Estimation: Decision-Directed and Non-Decision-Directed Loops using MATLAB		2	10		
9	M-PSK transceiver Design using USRP 2920 Software Defined Radio		2	10		
10	M-QAM transceiver Design using USRP 2920 Software Defined Radio		2	10		
Total Record Marks Obtained				**100**		
Lab Record Marks				**20**		
Lab EL				**20**		
Lab Test				**10**		
Final Assessment				**50**		

11	**Experiments for practice only:** i. ASK,FSK, BPSK,QPSK modulation and demodulation using Matlab ii. M-PAM (8,16 and 32 levels) modulation and demodulation using Matlab iii. M-QAM (64 and 256 levels) modulation and demodulation using Matlab communication toolbox iv. Offset QPSK modulation and demodulation using Matlab v. DPSK modulation and demodulation using Matlab vi. Spread spectrum techniques DSSS and FHSS using Matlab vii. Gaussian MSK modulation using Matlab viii. M-PAM, offset QPSK, DPSK and MSK modulation and demodulation using Labview software The 10 experiments specified in the lab content could be implemented in Matlab or Python Programming for LAB assignment.

Course Outcomes	
1	Associate the concept of geometric basis to well-specified baseband and bandpass symbols.
2	Analyze and compute performance of detected and estimated low pass and bandpass symbols under ideal and corrupted non-band limited channels.
3	Test and validate symbol processing and performance parameters at the receiver under ideal and corrupted bandlimited channels.
4	Demonstrate by simulation and emulation bandpass signals subjected to convolution coding and symbol processed at transmitter and correspondingly demodulated and estimated at receiver after passing through a corrupted channel

Rubrics for Lab Record evaluation

Sl.No	Criteria	Excellent	Good	Average	Max Score
Data sheet					
A	Problem statement	9-10	6-8	1-5	**10**
B	Design & specifications	9-10	6-8	1-5	**10**
C	Expected output	9-10	6-8	1-5	**10**
Record					
D	Simulation/ Conduction of the experiment	14-15	11-13	1-10	**15**
E	Analysis of the result.	14-15	11-13	1-10	**15**
Viva					**40**
Total					**100**
Scale down to 10 marks for each experiment					

Experiment 1

M-PSK Transceiver Design using Labview Software

<u>Aim:</u> To implement an M-PSK transceiver implementation using Labview software

<u>Tools required:</u> Desktop or Laptop with Labview software

<u>Procedure:</u>

1. The basic block diagram of M-PSK transceiver in Labview is shown in figure 1.1.
2. The corresponding front panel in Labview is shown in figure 1.2, 1.3 and 1.4 for M=4,8 and 16 respectively.
3. Vary the M-levels in front panel (Ex M=2,4,8,16 etc) and observe the transmitted signal spectrum, eye diagram, constellation diagram and transmitted symbols for a fixed SNR.
4. Observe the received signal spectrum, eye diagram, constellation and received symbols for a fixed SNR.
5. Repeat the step-4 for various values of the SNR and frequency offset.

Labview Description:

Phase-shift keying (PSK) in a digital transmission refers to a type of angle modulation in which the phase of the carrier is discretely varied to represent data being transmitted either in relation to a reference phase, or to the phase of the immediately preceding signal element. For example, when encoding bits, the phase shift could be 0° for encoding a 0 and 180° for encoding a 1, or the phase shift could be -90° for 0 and +90° for a 1, thus making the representations for 0 and 1 a total of 180° apart. In PSK systems designed so that the carrier can assume only two different phase angles, each change of phase carries one bit of information, that is, the bit rate equals the modulation rate. If you increase the number of recognizable phase angles to four, then 2 bits of information can be encoded into each signal element; likewise, eight phase angles can encode 3 bits in each signal element.

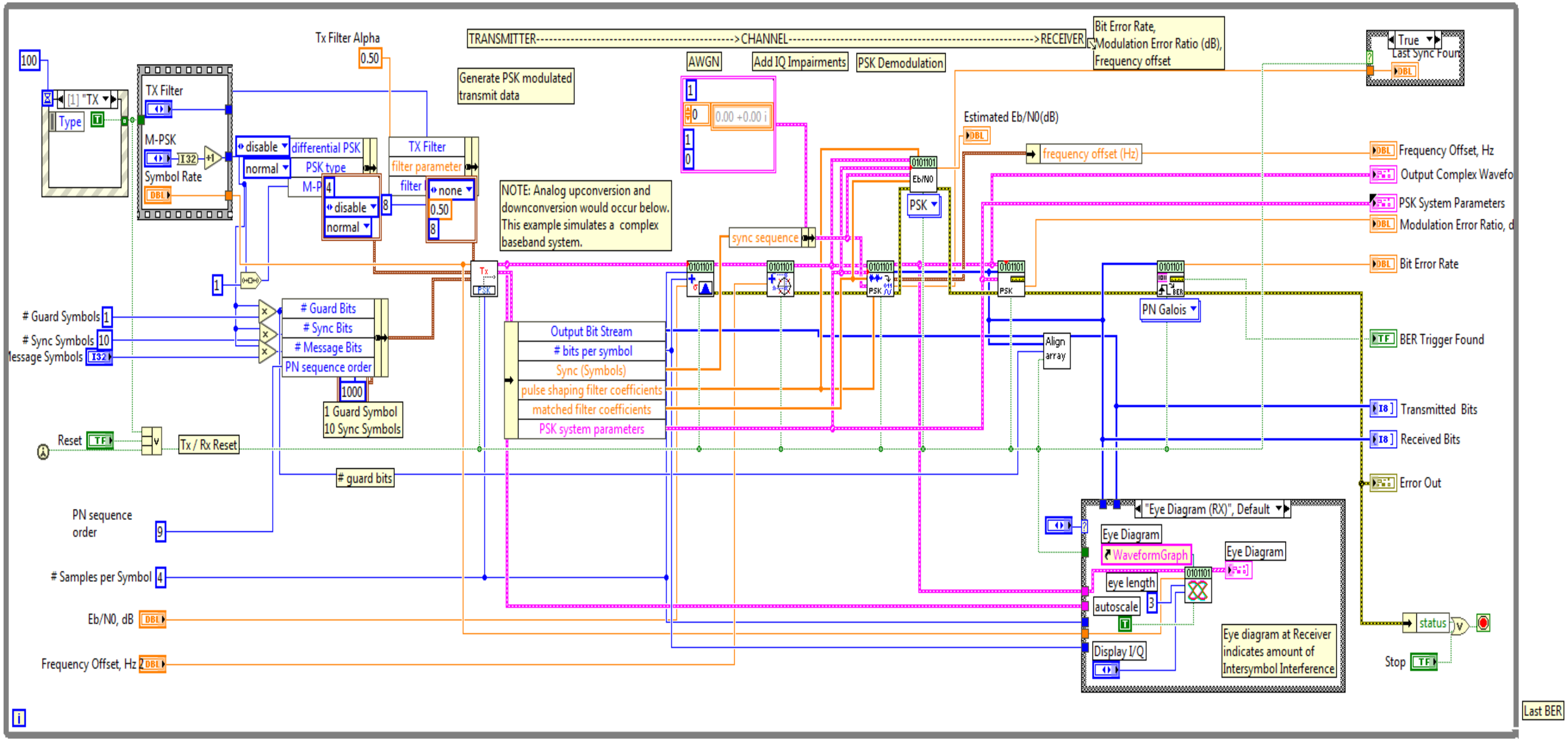


Fig1.1: Block diagram of MPSK Transceiver

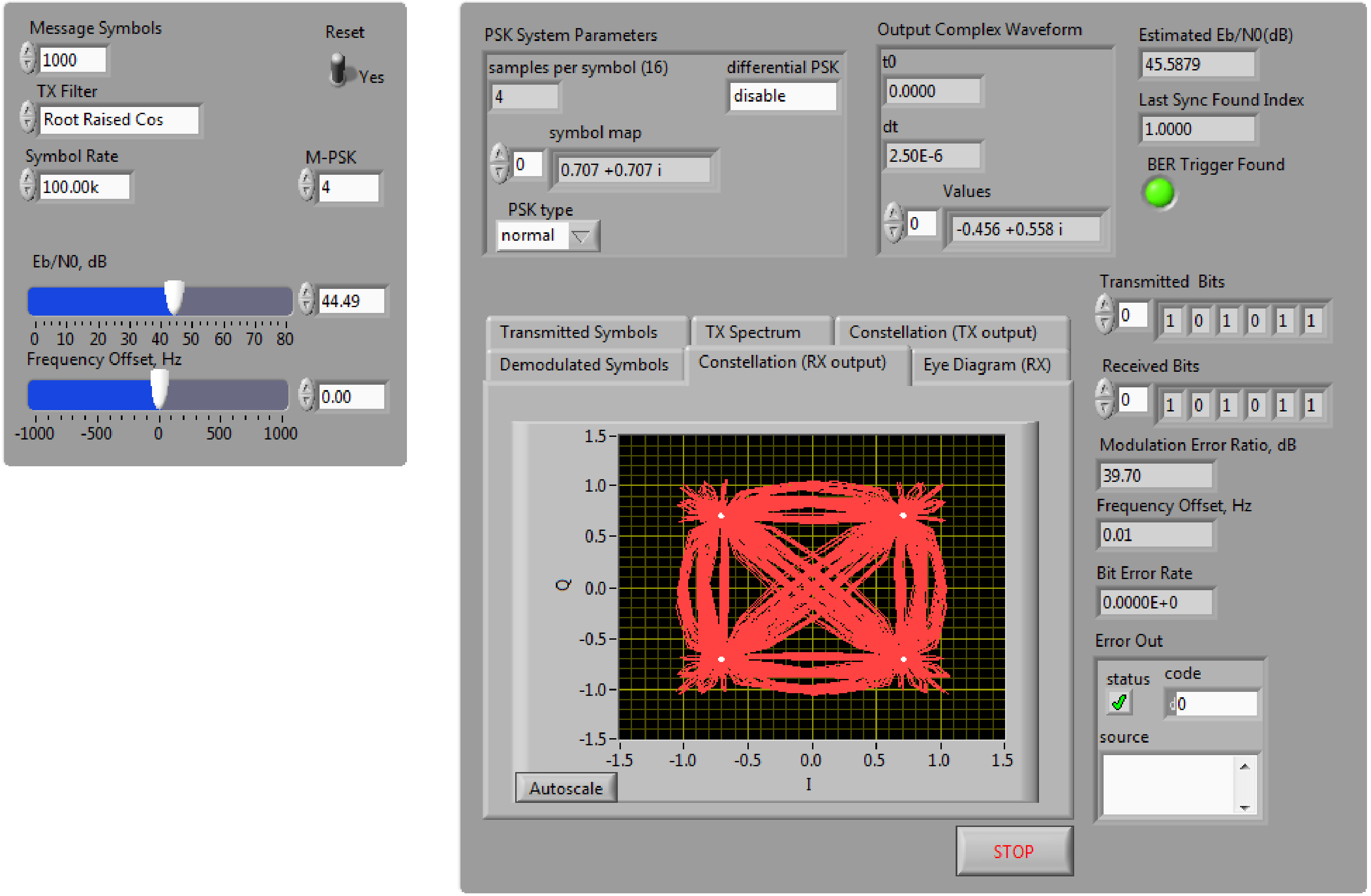


Fig1.2: Front panel for MPSK (M=4, QPSK)

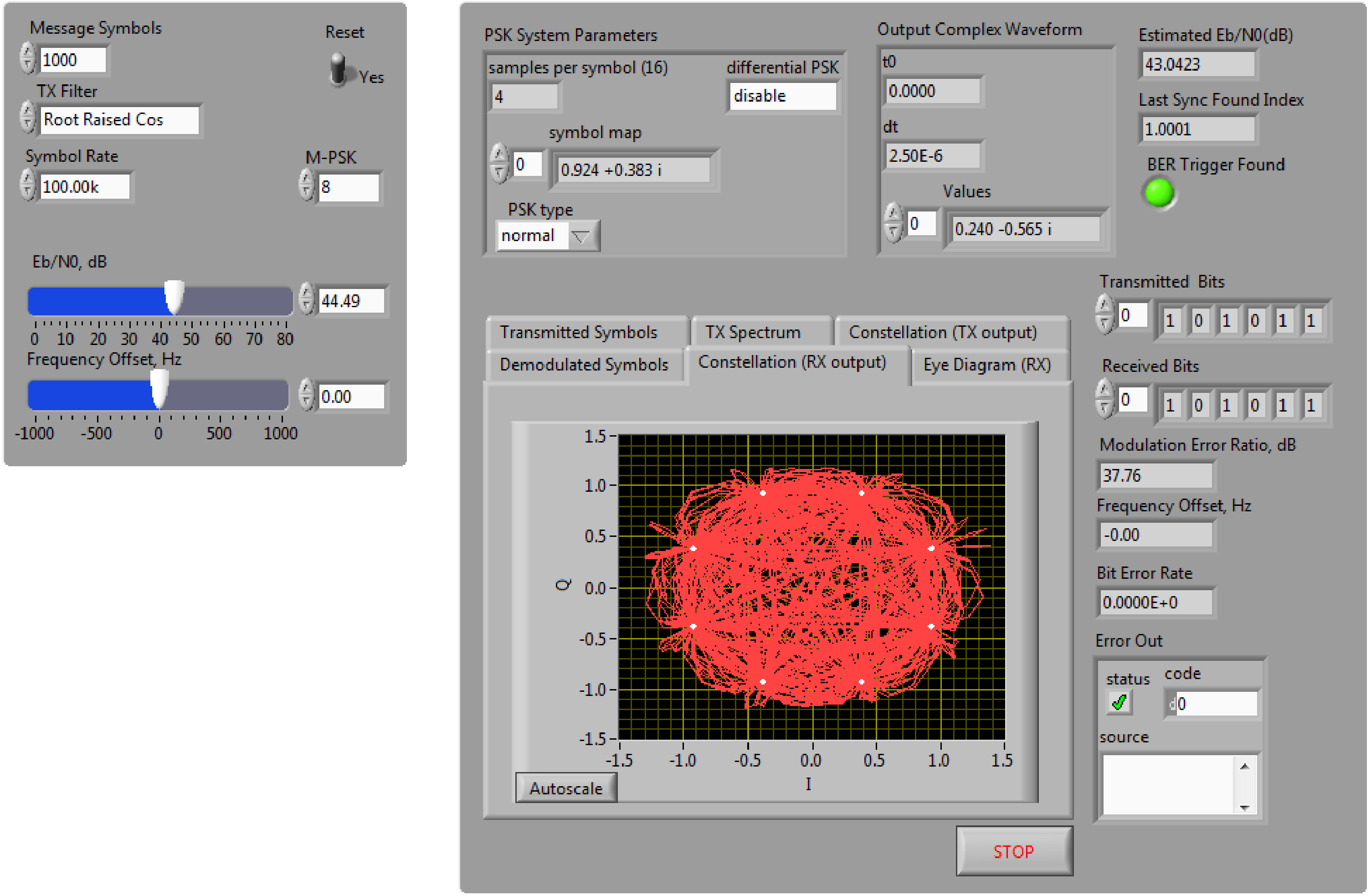


Fig1.3: Front panel for MPSK (M=8, 8-PSK)

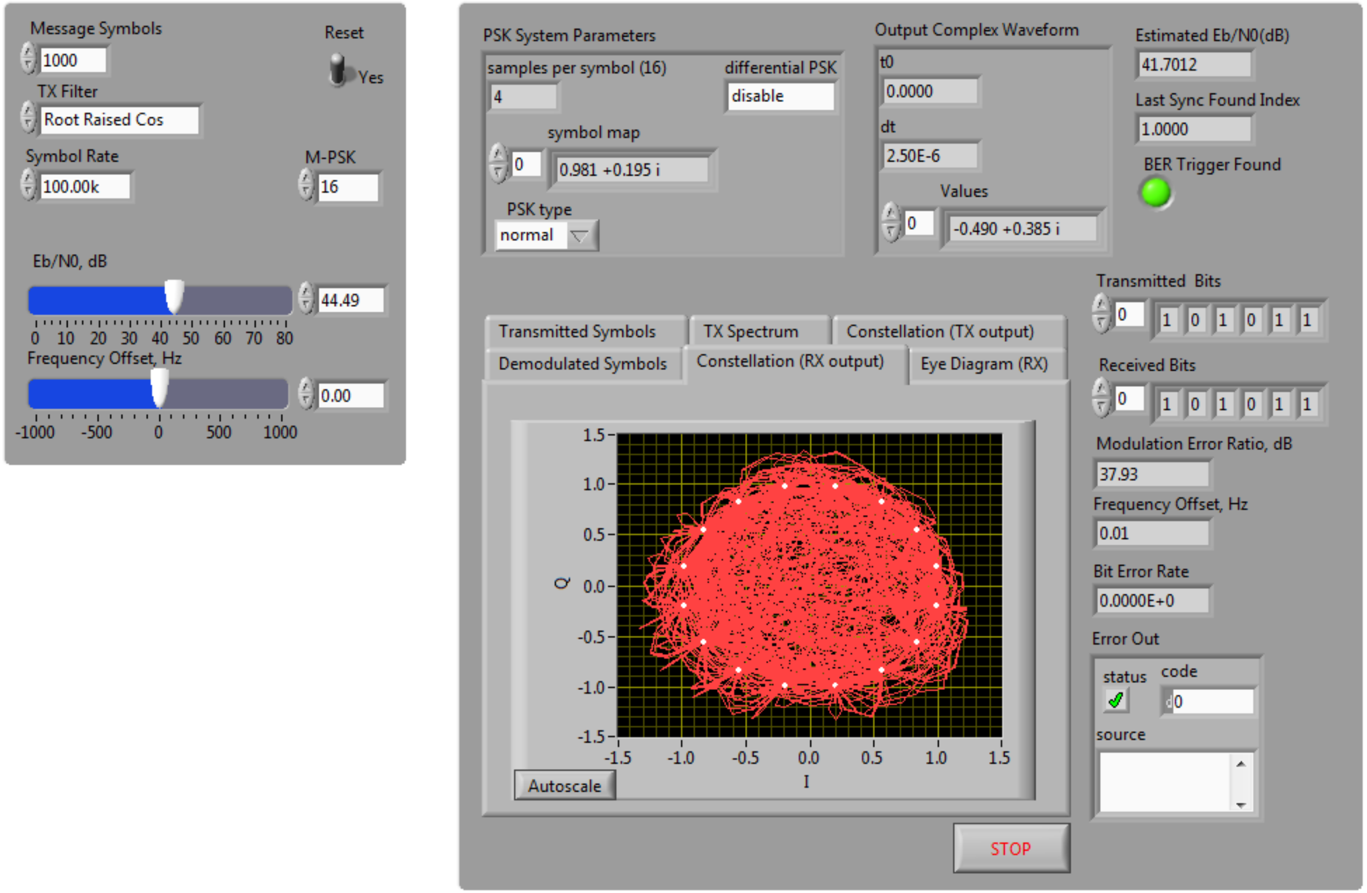


Fig1.4: Front panel for MPSK (M=16, 16-PSK)

Conclusion:

Sl.No	Criteria	Max Marks	Marks obtained
	Data sheet		
A	Problem statement	10	
B	Design & specifications	10	
C	Expected output	10	
	Record		
D	Simulation/ Conduction of the experiment	15	
E	Analysis of the result.	15	
	Viva	40	
	Total	100	
Scale down to 10 marks			

Experiment 2
M-QAM transceiver Design using Labview software

Aim: To implement M-QAM transceiver implementation using Labview software

Tools required: Desktop or Laptop with Labview software

Procedure:

1. The basic block diagram of M-QAM transceiver in Labview is shown in figure 2.1.
2. The corresponding front panel in Labview is shown in figure 2.2, 2.3 and 2.4 for M=4,8 and 16 respectively.
3. Vary the M-levels in front panel (Ex M=2, 4,8,16 etc) and observe the transmitted signal spectrum, eye diagram, constellation diagram and transmitted symbols for a fixed SNR.
4. Observe the received signal spectrum, eye diagram, and constellation and received symbols for a fixed SNR.
5. Repeat the step-4 for various values of the SNR and frequency offset.

Labview Description:

Quadrature Amplitude Modulation

A variety of communication protocols implement quadrature amplitude modulation (QAM). Current protocols such as 802.11b wireless Ethernet (Wi-Fi) and digital video broadcast (DVB), for example, both utilize 64-QAM modulation. In addition, emerging wireless technologies such as Worldwide Interoperability for Microwave Access (WiMAX), 802.11n, and HSDPA/HSUPA (a new cellular data standard) will implement QAM as well. Thus, understanding QAM is important because of its widespread use in current and emerging technologies.

QAM involves sending digital information by periodically adjusting the phase and amplitude of a sinusoidal electromagnetic wave. Each combination of phase and amplitude is called a symbol and represents a digital bitstream. This tutorial first covers the hardware implementation required to constantly adjust the phase and amplitude of a carrier wave. The tutorial also discusses the binary value associated with each symbol.

Quadrature amplitude modulation (QAM) requires changing the phase and amplitude of a carrier sine wave. One of the easiest ways to implement QAM with hardware is to generate and mix two sine waves that are 90 degrees out of phase with one another. Adjusting only the amplitude of either signal can affect the phase and amplitude of the resulting mixed signal.

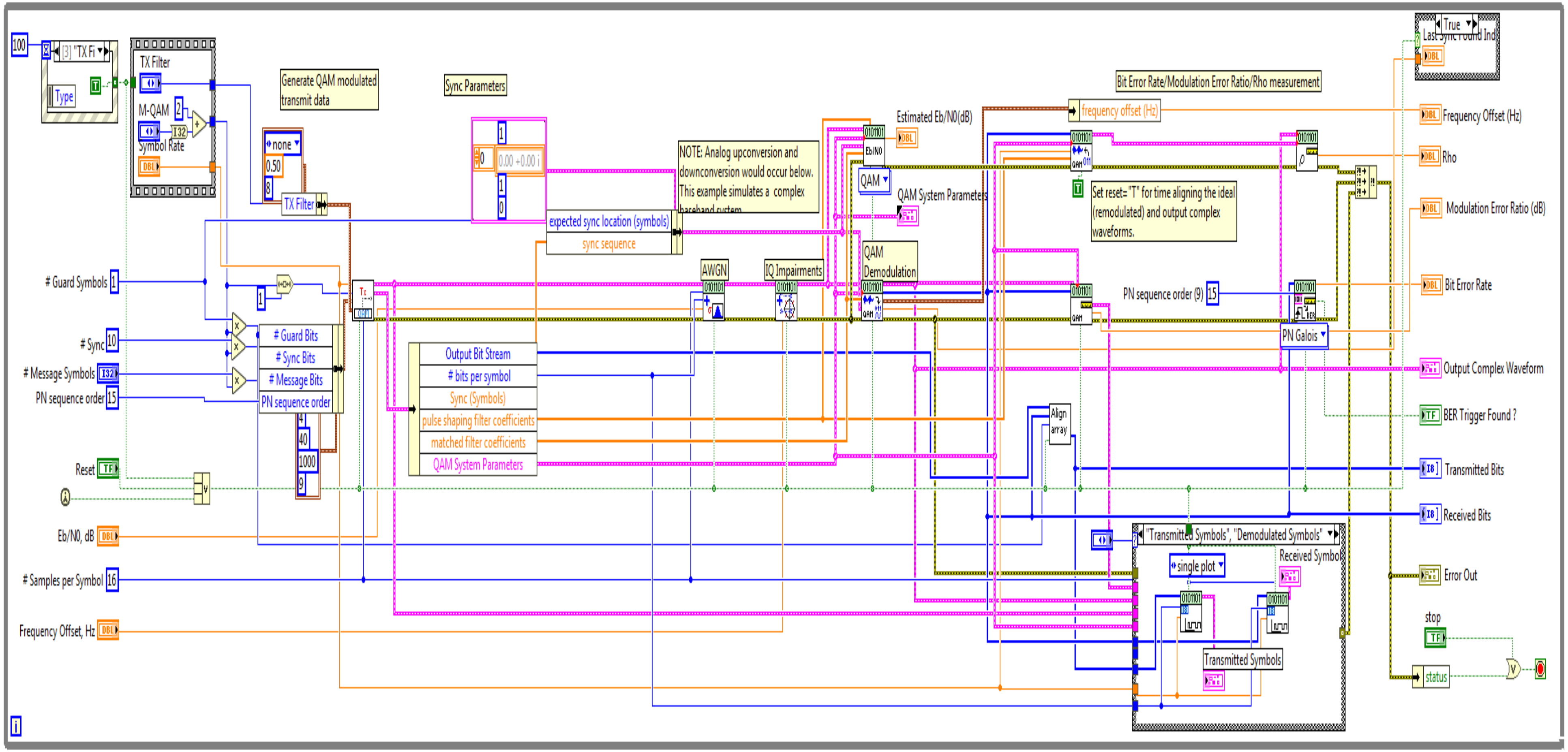


Fig2.1: Block diagram of MQAM Transceiver

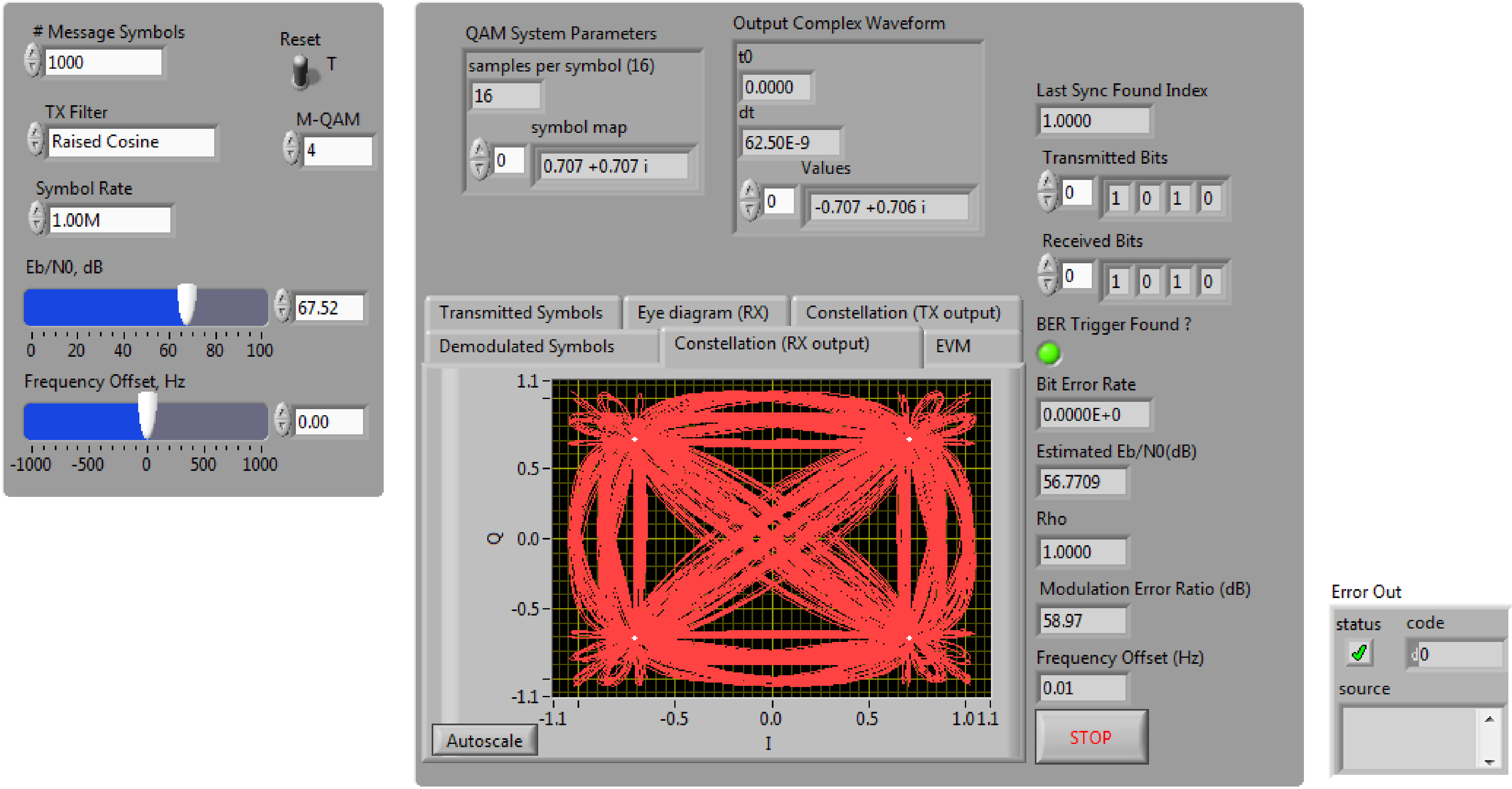


Fig2.2: Front panel for MQAM (M=4, 4-QAM)

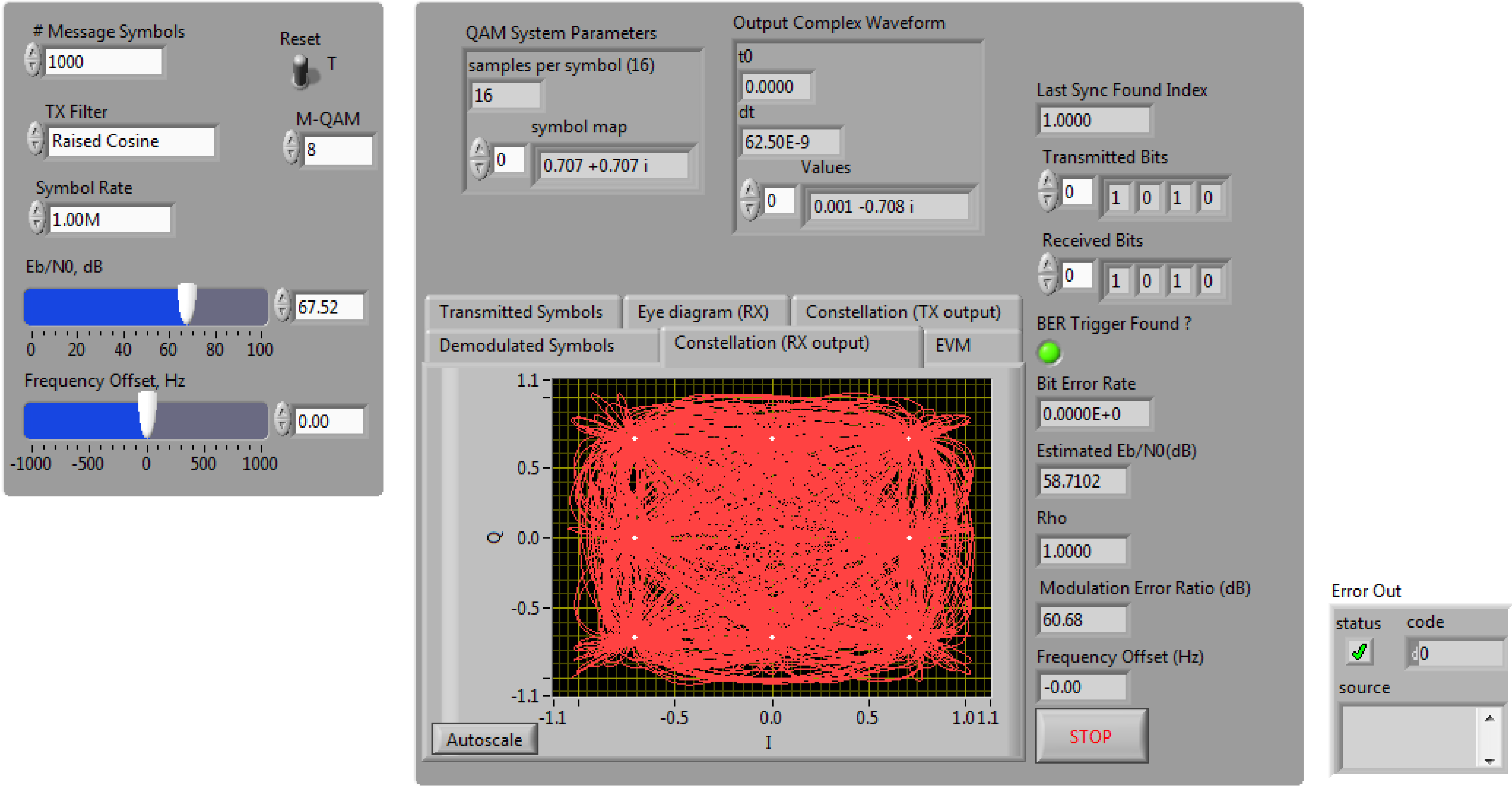


Fig2.3: Front panel for MQAM (M=8, 8-QAM)

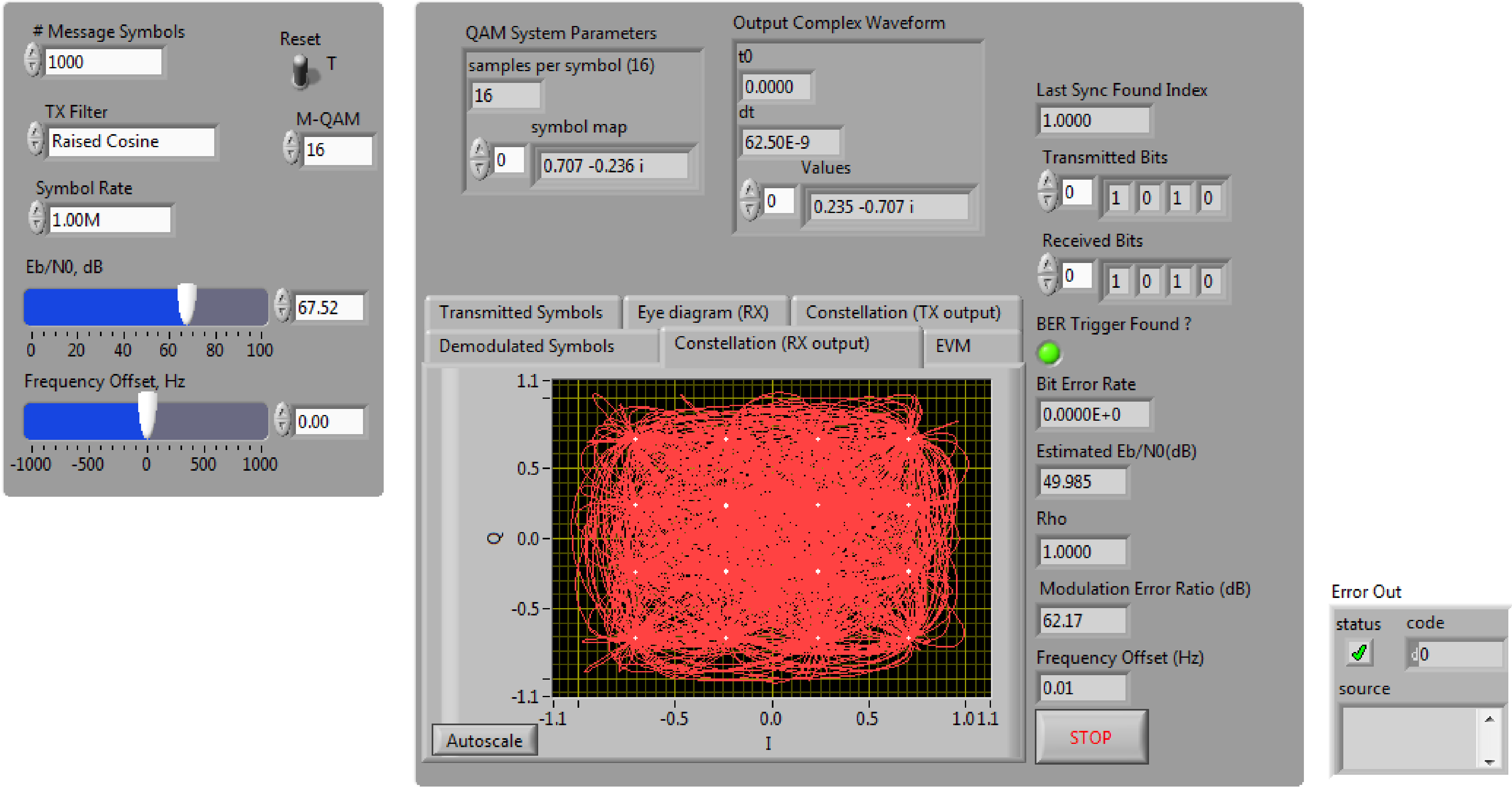


Fig2.4: Front panel for MQAM (M=16, 16-QAM)

Conclusion:

Sl.No	Criteria	Max Marks	Marks obtained
Data sheet			
A	Problem statement	10	
B	Design & specifications	10	
C	Expected output	10	
Record			
D	Simulation/ Conduction of the experiment	15	
E	Analysis of the result.	15	
	Viva	40	
	Total	100	
Scale down to 10 marks			

Experiment 3
ISI with PAM modulation technique using MATLAB

Aim: To implement ISI with PAM modulation technique using MATLAB

Tools required: Desktop or Laptop with MATLAB software

Procedure:

1. Generate PAM symbols and modulate
2. Upsample the modulated symbols and pass it through SRRC pulse shaping
3. Add AWGN noise for given SNR value
4. Matched filter with SRRC pulse shape
5. Symbol rate sampler and demodulation
6. Generating eye plot at the matched filter output

Theory:

The basic pulse amplitude modulation (PAM) system is enhanced in Figure 3.1 by adding an upsampler (↑ L), pulse shaping function (p[n]) at the transmitter and a matched filter (g[n]), down-sampler (↓ L) combination at the receiver. In this model, a random stream of source bits is first segmented into k-bit wide symbols that can take any value from the set m ∈ {1, 2, ..., M}. The simulation code directly starts by generating a random set of symbols, that goes into the modulation mapper. Pulse amplitude modulation (MPAM) mapping and de-mapping, are considered here for simulation. An MPAM modulator maps the k-bit information symbols to one of the $M = 2^k$ distinct signaling levels. The MPAM modulated symbols are shown in Figure 3.2. The sample waveforms at each stage of figure 3.1 are shown for reference in Fig3.3 onwards.

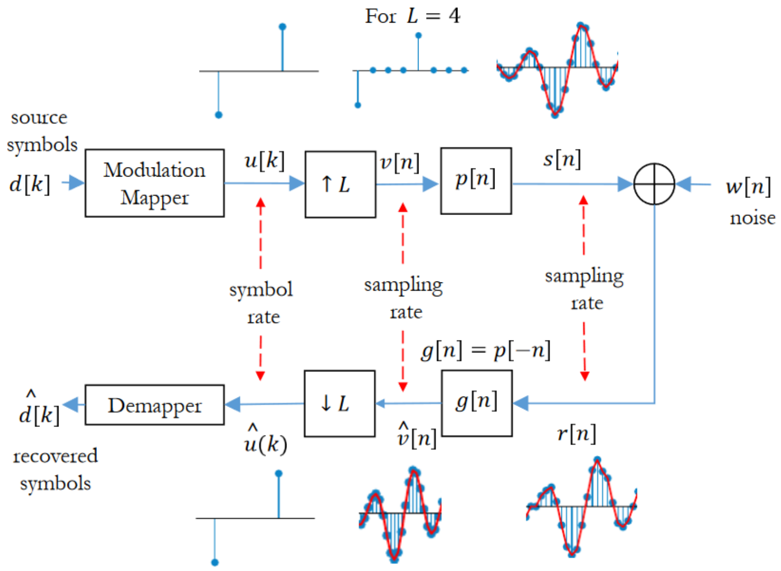


Fig. 3.1: DSP implementation of a PAM modulation system with pulse shaping and matched filtering

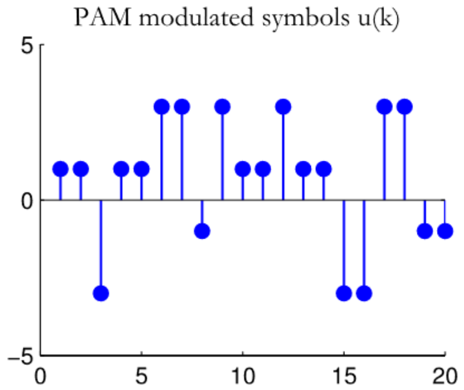


Fig3.2: MPAM modulated symbols

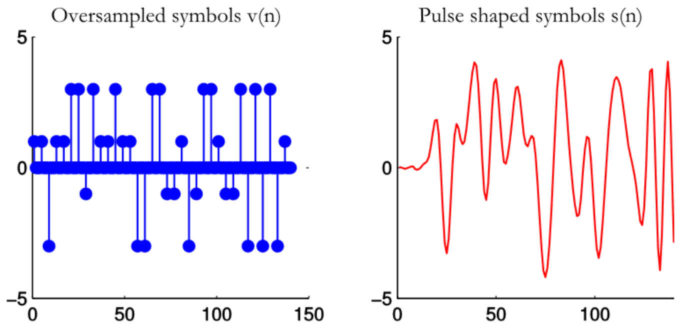


Fig3.3: Modulated symbols upsampled by 4 (left) and the SRRC pulse shaping filter output (right)

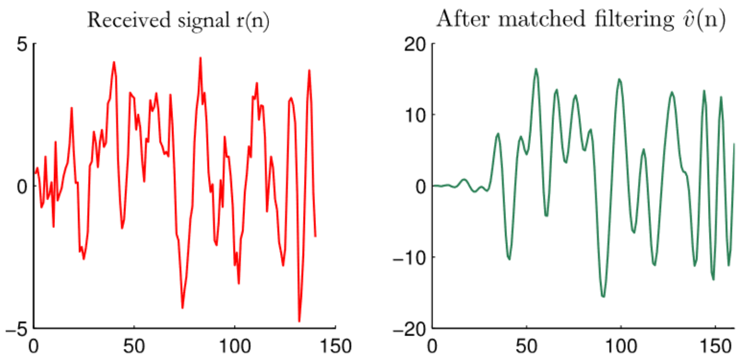


Fig. 3.4: Received signal with AWGN noise (left) and the output of the matched filter (right)

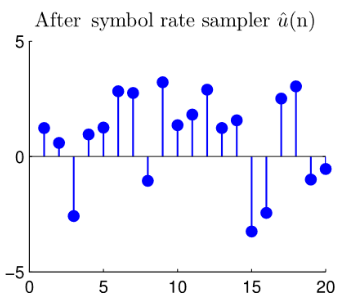


Fig3.5: Down sampling - output of symbol rate sampler

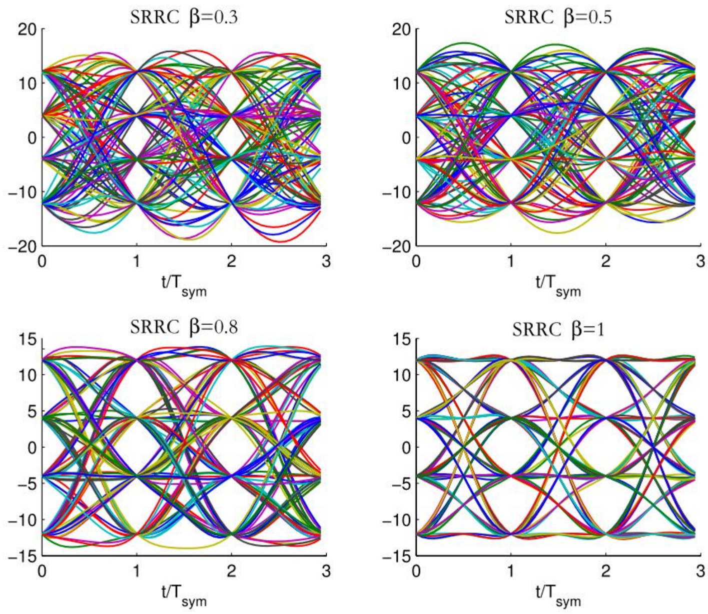


Fig. 3.6: Eye plot at the matched filter output just before the symbol sampler at the receive

Conclusion:

Sl.No	Criteria	Max Marks	Marks obtained
Data sheet			
A	Problem statement	10	
B	Design & specifications	10	
C	Expected output	10	
Record			
D	Simulation/ Conduction of the experiment	15	
E	Analysis of the result.	15	
	Viva	40	
	Total	100	
Scale down to 10 marks			

Experiment 4

Equalization with QAM Modulation using Labview

Aim: Demonstrate the implementation of Equalization in an QAM Modulation-Demodulation System

Tools required: Desktop or Laptop with Labview software

Theory:

Equalization in QAM modulation is used to combat channel impairments like inter-symbol interference (ISI) and multipath fading, which distort the signal during transmission. It restores the signal's integrity, ensuring correct data detection. Channel Effects (noise, fading, multipath) distort the QAM signal, causing symbol detection errors. Equalization mitigates these effects by compensating for channel-induced distortions. Common techniques used are:

- Linear Equalizers: Zero Forcing (ZF) and Minimum Mean Square Error (MMSE).
- Decision Feedback Equalizers (DFE): Uses previous symbols to remove ISI.
- Adaptive Equalizers: Dynamically adjusts to changing conditions.
- MLSE: Optimal but computationally heavy.

Equalization is Critical for high-order QAM (64-QAM, 256-QAM) in wireless and high-speed communication systems like 5G, Wi-Fi, and DSL. Equalization ensures reliable data transmission in QAM systems by counteracting channel distortions.

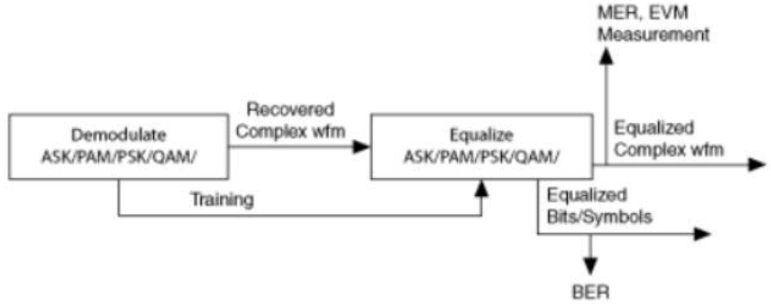
Procedure:

1. The front panel in Labview is shown in figure 4.1, for M=16.
2. The basic block diagram of M-QAM with equalization in Labview is shown in figure 4.2.
3. Vary the M-levels in front panel (Ex M=4,8,16,256 etc) and observe the Equalizer Taps, LMS Error, constellation diagram before equalization and constellation diagram after equalization.
4. Observe the received data, BER and modulation error ratio.
5. Repeat the step-4 for various values of the SNR and equalizer length.

LabVIEW Description:

This program applies adaptive feedforward software equalization using the least-mean-squared (LMS) algorithm to the QAM-demodulated input complex waveform. Use the Specify Coefficients instance of this polymorphic VI if the equalizer coefficients are already known or when the initial conditions for the equalizer coefficients must be specified. Use the Specify Length polymorphic instance when only the length of the QAM feedforward equalizer (in symbols) must be specified.

The following flow chart illustrates demodulator-to-equalizer data flow.



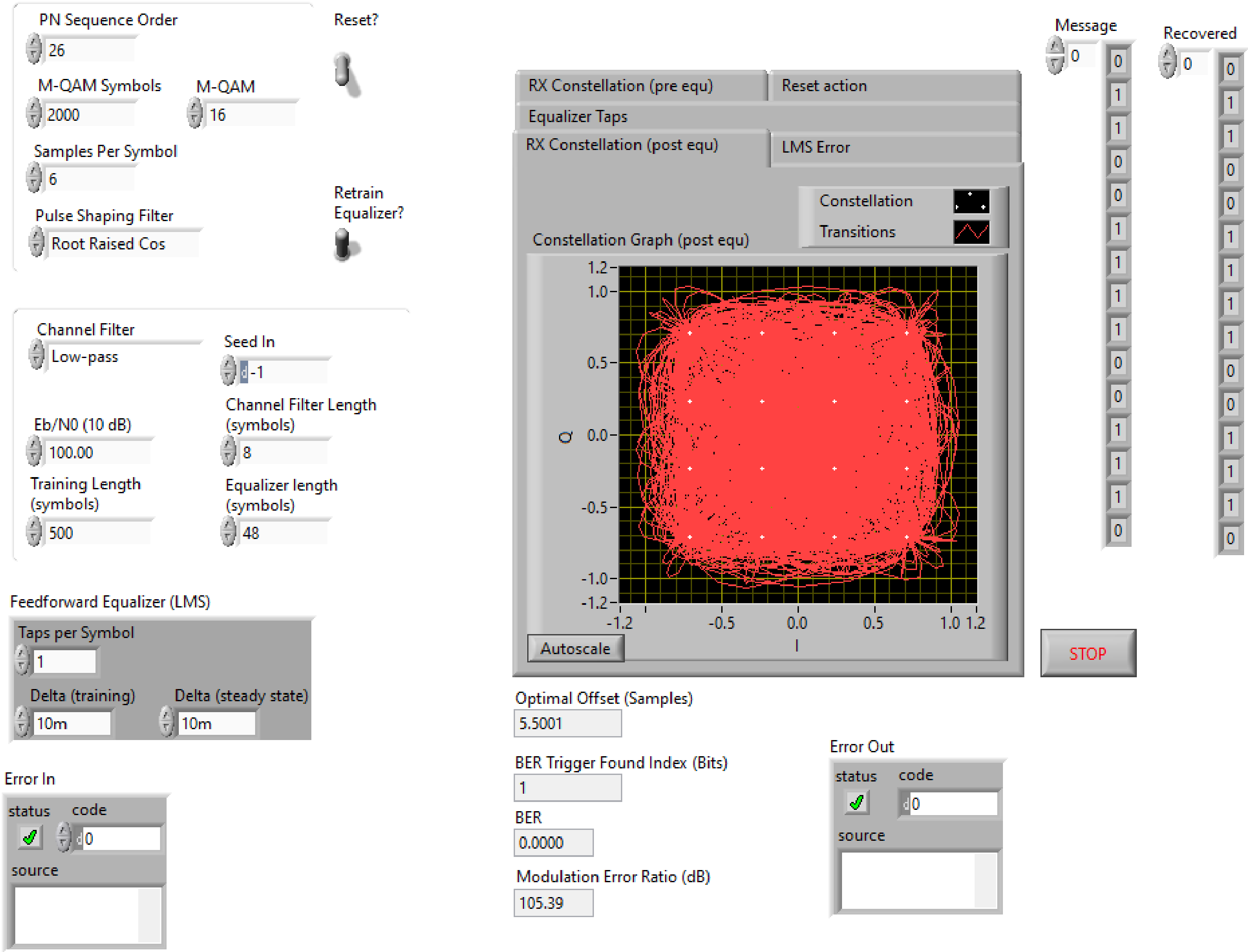


Fig4.1: Front panel for MQAM with Equalizer

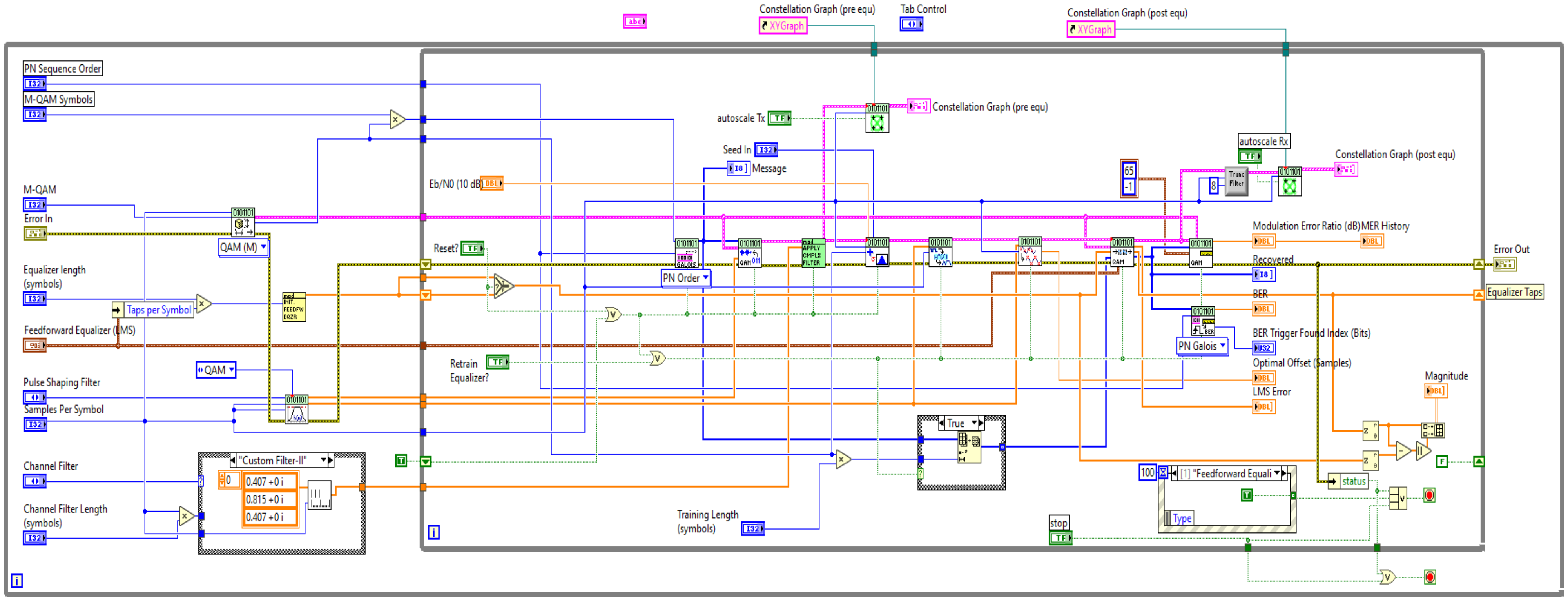


Fig4.2: Block diagram for MQAM with Equalizer

Conclusion:

Sl.No	Criteria	Max Marks	Marks obtained
Data sheet			
A	Problem statement	10	
B	Design & specifications	10	
C	Expected output	10	
Record			
D	Simulation/ Conduction of the experiment	15	
E	Analysis of the result.	15	
	Viva	40	
	Total	100	
Scale down to 10 marks			

Experiment 5
Transmitter and Receiver Implementation of Direct Sequence Spread Spectrum and Frequency Hopped Spread Spectrum

Aim: To implement Transmitter and Receiver for Direct Sequence Spread Spectrum and Frequency Hopped Spread Spectrum

Tools required: Desktop or Laptop with Matlab software

Theory:

Spread-spectrum (SS) technology uses a sequential noise-like signal to spread the normally narrowband information signal over a relatively wide band of radio frequencies. The receiver correlates the received signals to retrieve the original information signal. Using wideband, noise-like signals to "spread the spectrum" used makes the transmissions hard to detect, intercept, jam, or demodulate. This is why the military has used spread-spectrum methods for many years.

The Direct Sequence Spread Spectrum (DSSS) Technique

In the Direct Sequence Spread Spectrum (DSSS) technique, the signal being transmitted is divided and injected with multiple frequencies within a particular frequency band as shown in figure 7.1a The original data is mixed with redundant data bits or code, called chips or chipping code, and the ratio of the chips to information is called the spreading ratio. A high spreading ratio indicates a wider bandwidth. As per Shannon's theorem, the higher the bandwidth, the more immune the signal transmission is to crosstalk, interference, and noise.

The DSSS technique helps maintain secure signal transmission with a high signal-to-noise ratio (SNR) at the receiving end. And, the DSSS technique helps recover the original data even when a part of the transmitted data is corrupted. The DSSS method of modulation has a higher rate of transmission than the FHSS method. In terms of the signal transmission rate, DSSS is superior. However, DSSS is vulnerable to electromagnetic interference and noise produced by devices operating at the same frequency band. Output for DSSS at transmitter and receiver are shown in figures 5.2.

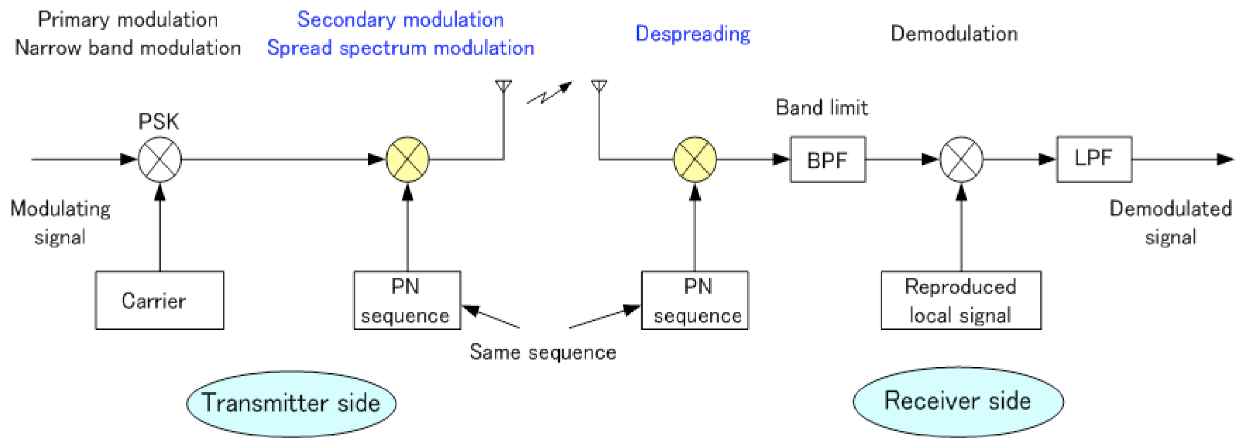


Fig 5.1a- Transmitter and Receiver Implementation of Direct Sequence Spread Spectrum

The Frequency Hopping Spread Spectrum (FHSS) Technique

The spread spectrum technique in which the carrier frequency is changed according to the pseudo-random noise or sequence injected is called the frequency hopping spread spectrum (FHSS) technique as shown in figure 5.1b. The pseudo-random sequence is only available with the transmitter and receiver. The signal transmitted is injected with noise, increasing the bandwidth of the original information. At the receiving end, the pseudo-random sequence is removed to retrieve the original information. The FHSS technique prevents the loss of data and limits noise, crosstalk, and electromagnetic interference, preserving the signal integrity and reliability of communications.

According to the frequency at which the RF frequency varies in signal communications, the FHSS technique is classified into either slow frequency hopping or fast frequency hopping. In fast frequency hopping, the hopping is carried out at a faster rate than the information bit rate. When the frequency hopping rate is less than the message bit rate, the spread spectrum technique is called slow frequency hopping. The frequency hopping spread spectrum technique is often implemented in wireless local area networks. Output for FHSS at transmitter and receiver are shown in figures 5.3.

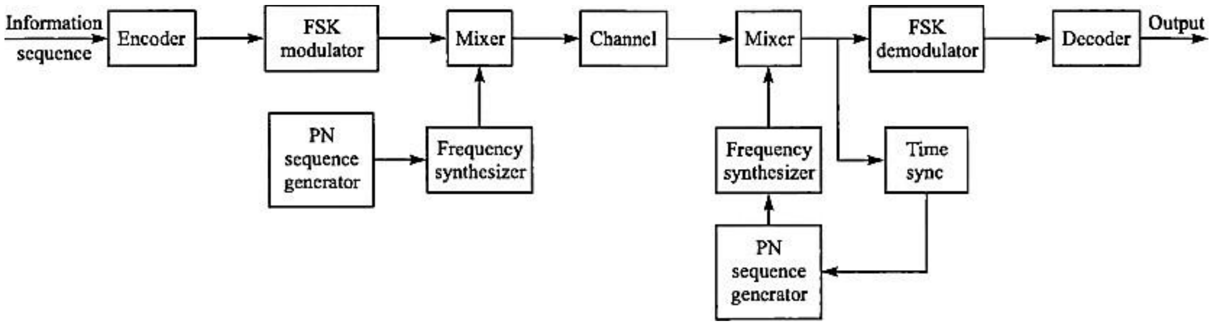


Figure 5.1b: Transmitter and Receiver Implementation of Frequency Hopped Spread Spectrum

Output:

1. **Direct Sequence Spread Spectrum (DSSS) Technique**

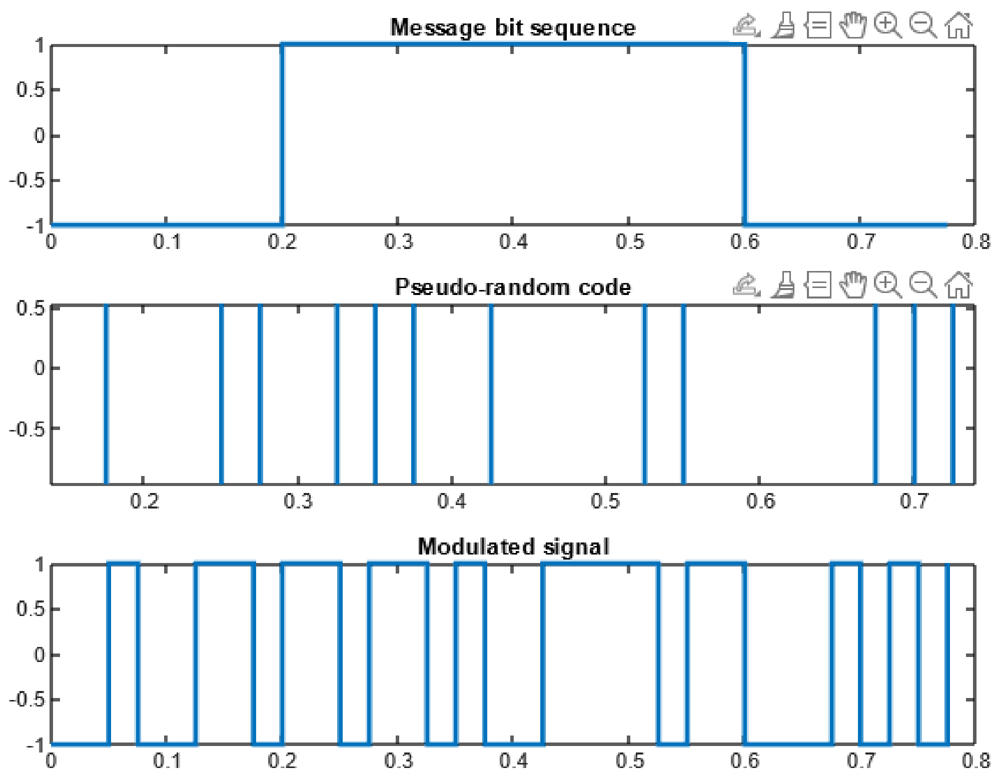


Figure 5.2: Direct sequence spread spectrum signals at transmitter in time domain

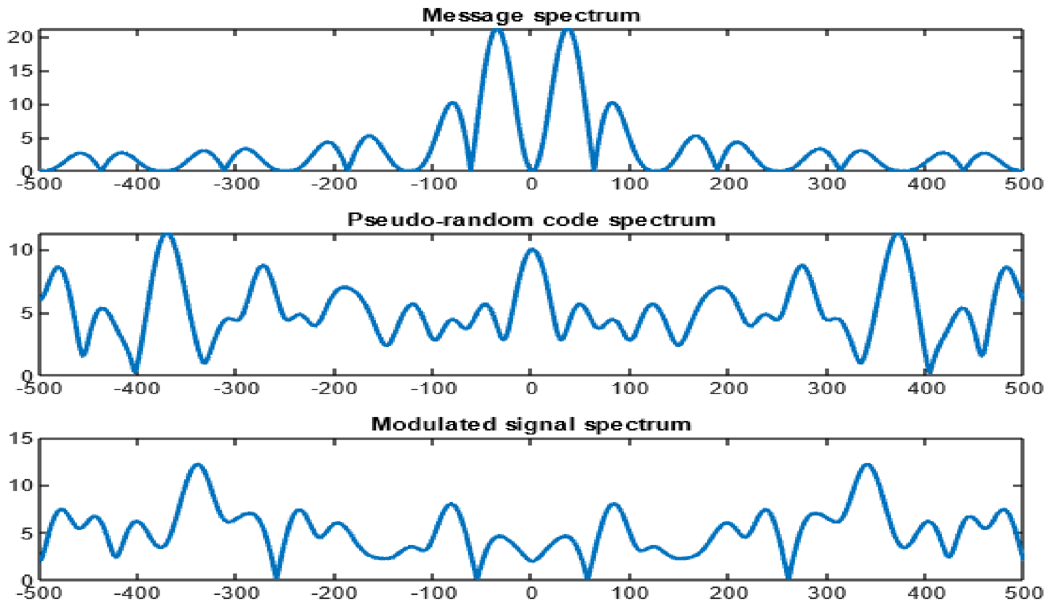


Figure 5.2 b: Direct sequence spread spectrum output at transmitter in frequency domain

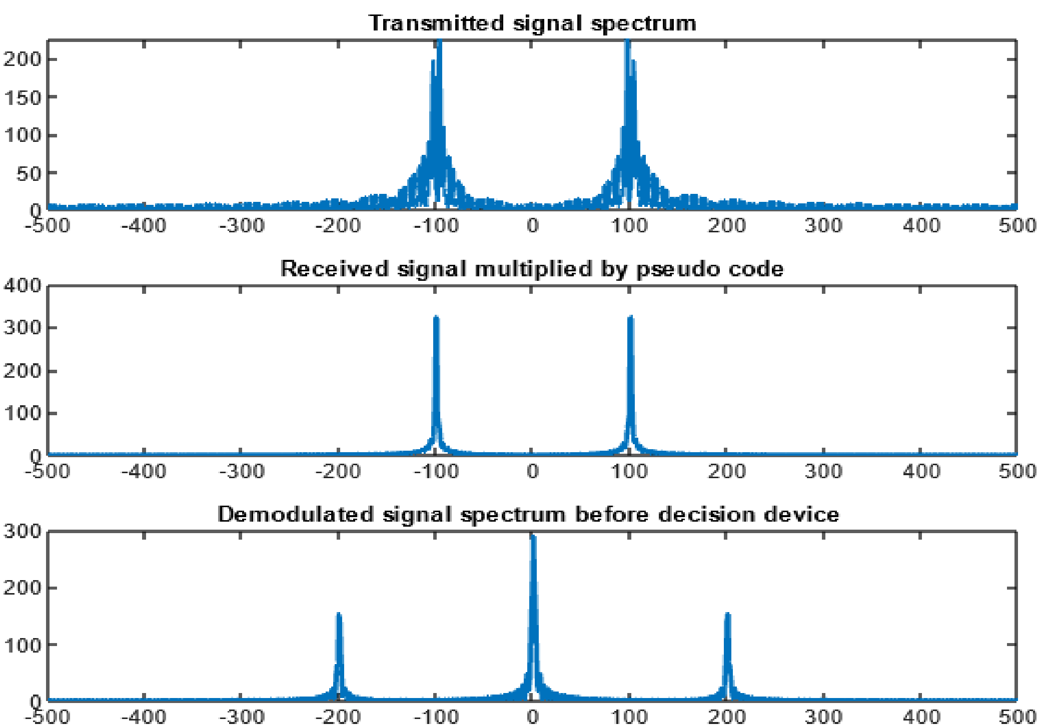


Figure 5.2 c: Direct sequence spread spectrum output at receiver in frequency domain

2. Frequency Hopping Spread Spectrum (FHSS) Technique

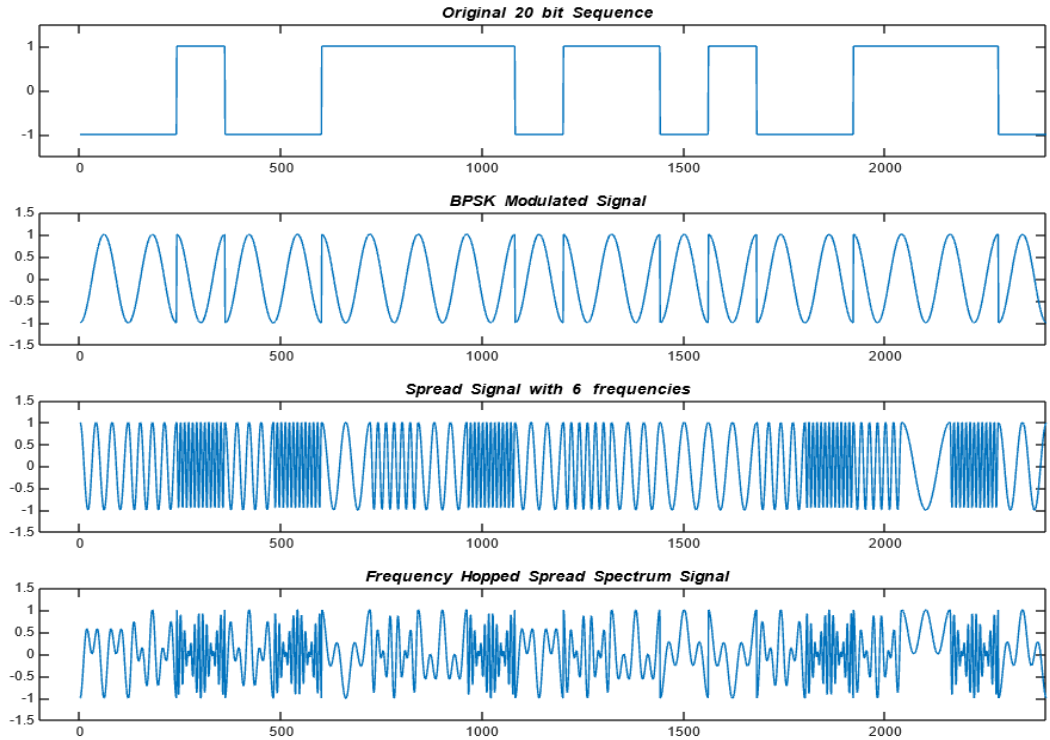


Figure 5.3 a: Frequency spread spectrum signals at transmitter

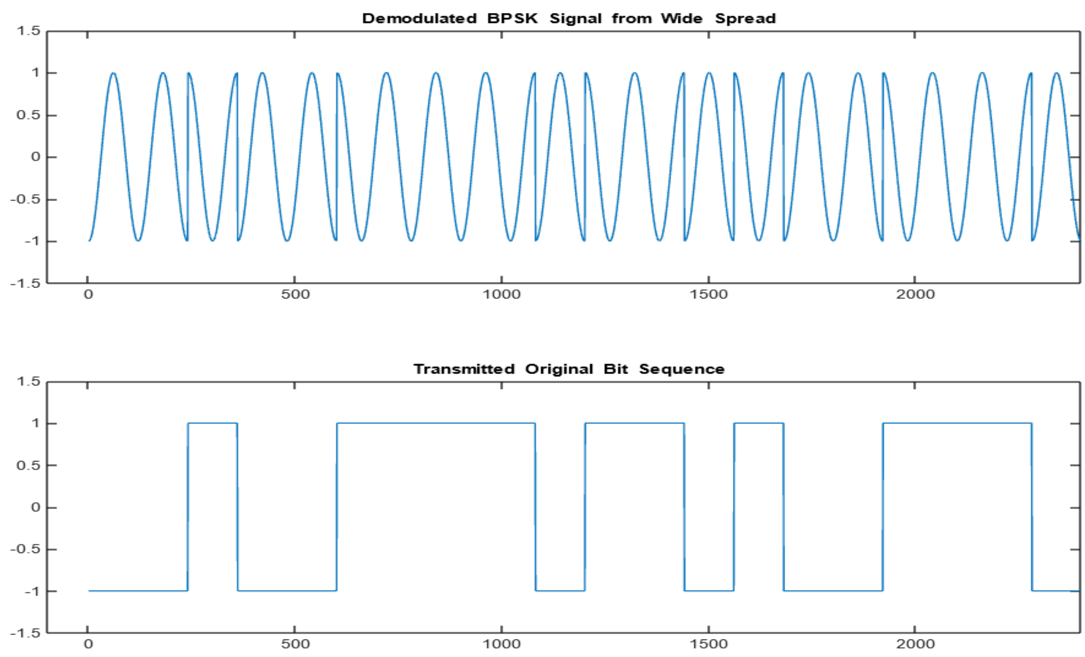


Figure 5.3 b: Frequency spread spectrum signals at receiver

Conclusion:

Sl.No	Criteria	Max Marks	Marks obtained
Data sheet			
	Problem statement	10	
	Design & specifications	10	
	Expected output	10	
Record			
D	Simulation/ Conduction of the experiment	15	
E	Analysis of the result.	15	
	Viva	40	
	Total	100	
Scale down to 10 marks			

Experiment 6
Symbol Timing Estimation using MATLAB

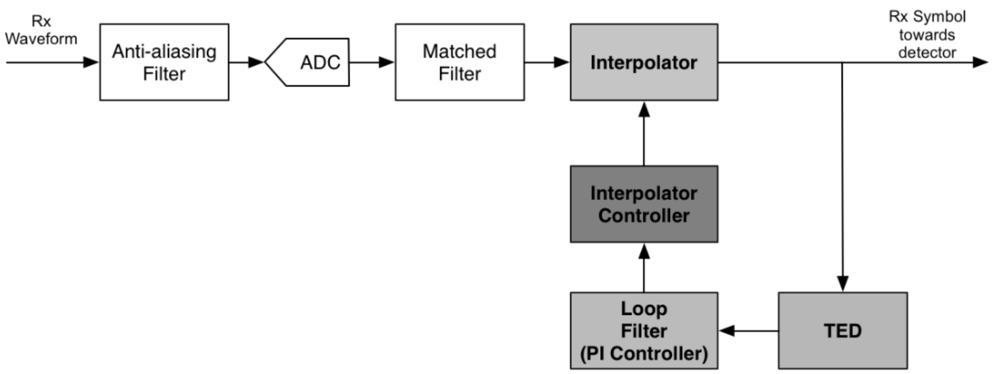
Aim: To demonstrate the symbol timing estimation

Tools required: Desktop or Laptop with Matlab software

Theory:

Symbol timing synchronization works by selecting the best sample from each group of samples to represent the transmitted symbol. The goal is to produce L samples per symbol at the matched filter output, with one of those samples aligned with the maximum eye opening

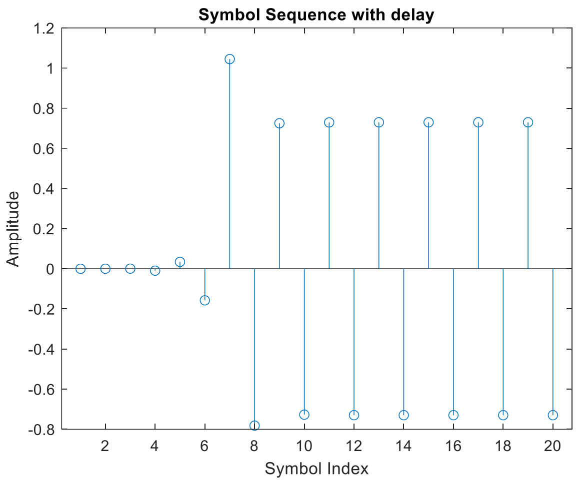
i. The block diagram of Symbol timing synchronization consists of
ii. Timing Error Detector
iii. Loop Filter
iv. Interpolator Controller and Interpolator

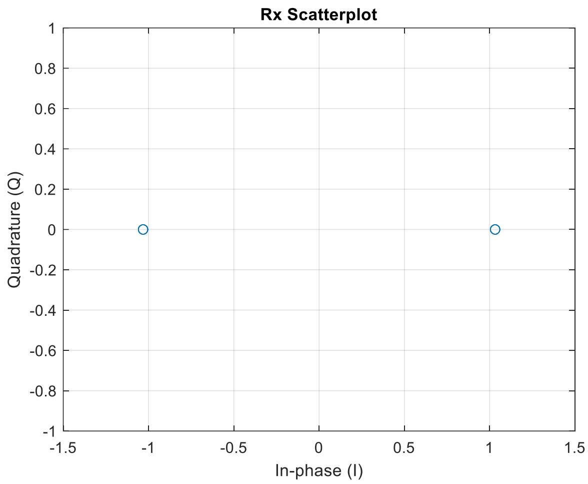


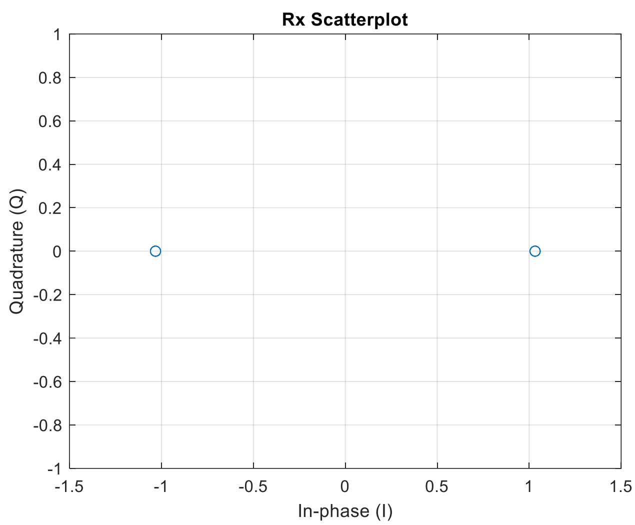
Algorithm:

1. Design a Square-root Raised Cosine filter for pulse shaping
2. Generate the transmission of a 2-PAM/4-PAM symbol sequence
3. Add a random channel propagation delay in units of sampling intervals
4. Develop a receiver that does not perform symbol timing synchronization. (The matched filter block precedes the symbol timing recovery loop).
5. Add an arbitrary timing used by the receiver to select samples from the incoming sequence and pass them to the decision module
6. Plot the scatter plots
7. Design symbol timing offset exactly and plot the received symbols and scatter plots.

Expected plots







Conclusion:

Sl.No	**Criteria**	Max Marks	Marks obtained
Data sheet			
A	Problem statement	10	
B	Design & specifications	10	
C	Expected output	10	
Record			
D	Simulation/ Conduction of the experiment	15	
E	Analysis of the result.	15	
	Viva	40	
	Total	100	
Scale down to 10 marks			

Experiment 7

Frequency selective and non-selective fading channel simulation using MATLAB

Aim: To observe the impact on the received signals after passing through frequency selective fading and frequency non-selective fading channel.

Tools required: Desktop with Matlab

Theory:

Frequency nonselective fading channel: If all frequency components of a transmitted signal that fall within the coherence bandwidth Bcb will fade simultaneously. If the transmitted signal has a bandwidth W < Bcb, the channel is called frequency

nonselective. Thus, at any instant in time, all frequency components of the transmitted signal fade simultaneously. On the other hand, if the transmitted signal has a bandwidth W > Bcb, the frequencies in the signal separated by an amount greater than Bcb will be affected differently by the channel. Hence, at any instant, some frequency components in the transmitted signal may fade, whereas other frequency components may not. In such a case, the channel is said to be frequency selective.

$$B_{cb=}\ \frac{1}{T_m}$$

Where, channel coherence bandwidth (Bcb) and multipath (time) spread of the channel (Tm)

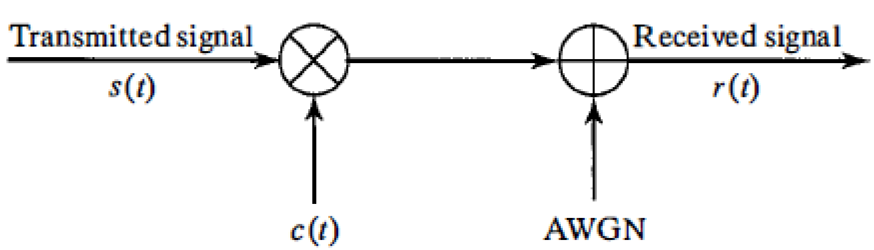


Fig-7.1 Model of a frequency nonselective time-varying channel with AWGN.

The frequency nonselective, slowly fading channel model as shown in fig 7.1 applies to many physical radio channels used for digital communications when the bandwidth W of the transmitted signal satisfies the condition W « Bcb· However, there are communication systems in which the transmitted signal bandwidth W » Bcb• so that the channel is frequency selective. In such a case, a more complex channel model must be employed as shown in fig 7.2.

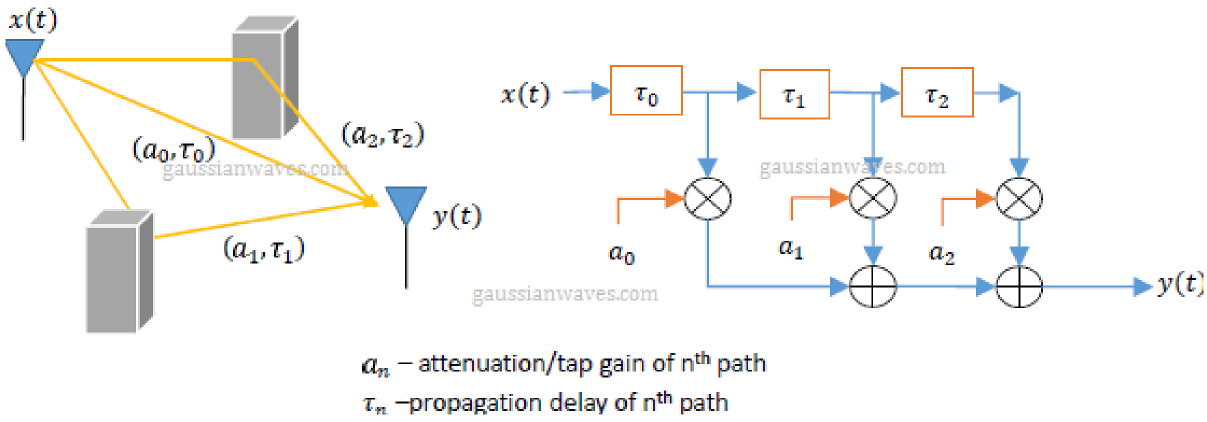


Fig7.2- Model of a frequency nonselective time-varying channel with AWGN.

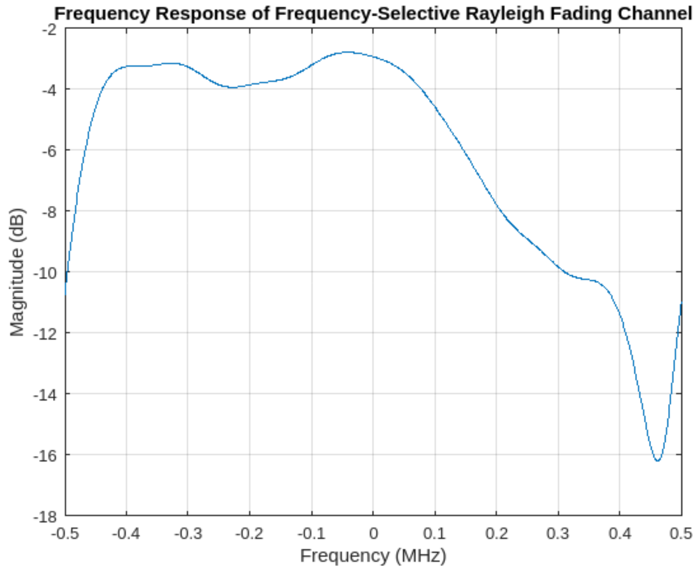
A **frequency selective fading channel** is a type of fading in wireless communications where different frequencies of the transmitted signal experience varying amounts of attenuation. This happens cause the signal's different frequency components take multiple paths to the receiver (multipath propagation), causing some to interfere constructively while others interfere destructively.

Procedure:

Steps to Simulate Frequency Selective Fading Channel in MATLAB

1. Define Simulation Parameters
First, define basic parameters like the signal properties, number of paths, path delays, and gains.
2. Generate a Test Signal
You can use a simple modulated signal such as BPSK, QPSK, or OFDM for testing.
3. Create Frequency Selective Fading Channel
MATLAB provides built-in functions to simulate a frequency selective fading channel. One common way is to use the rayleighchan or comm.RayleighChannel function.
4. Pass the Signal Through the Channel
Transmit the signal through the channel to observe the effects of frequency selective fading.
5. Visualize the Results
You can compare the transmitted and received signals to observe the impact of the frequency selective fading
6. Equalization (Optional)
If you want to mitigate the effect of frequency selective fading, you can use equalization techniques such as Zero Forcing (ZF) or MMSE.

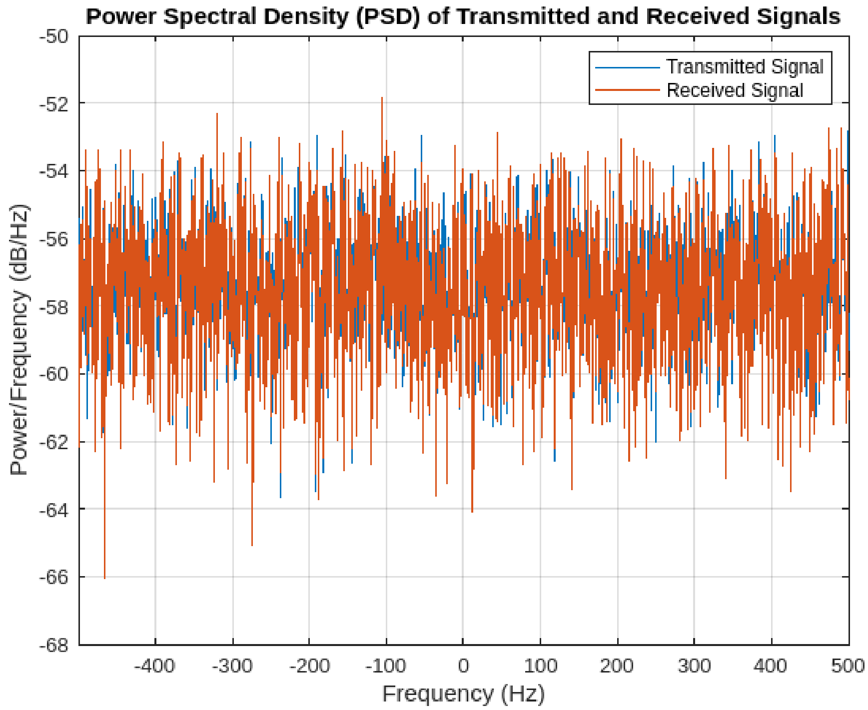
Expected output:



Steps to Simulate Frequency Non-Selective Fading Channel in MATLAB

1. Define Simulation Parameters
Start by defining the signal and channel parameters.
2. Create Frequency Non-Selective Fading Channel
A frequency non-selective fading channel can be simulated using a Rayleigh fading model, where the coherence bandwidth is larger than the signal bandwidth. The comm.RayleighChannel function can also be used for this purpose.
3. Pass the Signal Through the Channel
Now, pass the signal through the non-selective fading channel and observe the results.
4. Plot Transmitted and Received Signals
You can visualize the effect of the flat fading on the signal.
5. Equalization (Optional)
Since the fading is flat, the entire signal is uniformly attenuated. In this case, equalization is straightforward (e.g., a simple amplitude correction).

Expected output



Conclusion:

Sl.No	Criteria	Max Marks	Marks obtained
Data sheet			
	Problem statement	10	
	Design & specifications	10	
	Expected output	10	
Record			
D	Simulation/ Conduction of the experiment	15	
E	Analysis of the result.	15	
	Viva	40	
	Total	100	
Scale down to 10 marks			

Experiment 8
Carrier Phase Estimation: Decision-Directed and Non-Decision-Directed Loops

Aim: Carrier Phase Estimation - ML Carrier Phase Estimation, The PLL, Effect of Additive Noise on the Phase Estimate, Decision-Directed and Non-Decision-Directed Loops.

Tools required: Desktop with Matlab

Theory: The various stages of this experiment is described as follows.

1. QPSK Symbol Generation

QPSK Modulation:

The transmitted QPSK symbols can be represented mathematically as:

$$s[n] = A \cdot e^{j(\theta[n]+\phi)}$$

where:

- A is the amplitude of the signal,
- $\theta[n]$ is the phase corresponding to the transmitted symbol (taking values from $\{0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}\}$ for QPSK),
- ϕ is the true phase offset.

2. Received Signal with Noise

The received signal, including noise, is given by:

$$r[n] = s[n] \cdot e^{j\phi} + w[n]$$

where:

- $r[n]$ is the received signal,
- $w[n]$ represents the additive noise, modeled as $w[n] \sim \mathcal{CN}(0, \sigma^2)$, where $\sigma^2 = \frac{1}{2 \cdot 10^{\frac{SNR}{10}}}$ (variance of the complex Gaussian noise).

3. Maximum Likelihood (ML) Carrier Phase Estimation

The ML estimate of the phase can be computed as:

$$\hat{\phi}_{ML} = \arg\left(\sum_{n=1}^{N} r[n] \cdot e^{-j\theta[n]}\right)$$

where:

- $\hat{\phi}_{ML}$ is the estimated phase,
- N is the number of received symbols,
- $\arg(\cdot)$ is the argument (phase) function.

4. Phase-Locked Loop (PLL)

The PLL iteratively estimates the phase with the following equations:

Phase Error Calculation:

$$\text{Phase Error}[n] = \arg\left(r[n] \cdot e^{-j\hat{\phi}[n-1]}\right)$$

where:

- $\hat{\phi}[n-1]$ is the estimated phase from the previous sample.

Update Rule:

$$\hat{\phi}[n] = \hat{\phi}[n-1] + K \cdot \text{Phase Error}[n]$$

where:

- K is the loop gain or bandwidth (loop bandwidth).

5. Corrected Received Signal

The corrected received signal after applying phase estimation is:

$$r_{corrected}[n] = r[n] \cdot e^{-j\hat{\phi}[n]}$$

6. **Effects of Additive Noise on Phase Estimation**

The variance of the phase estimation error can be calculated as:

$$\mathrm{Var}(\text{Phase Error}) = \mathrm{E}\left[(\angle r[n] - \phi)^2\right]$$

where:

- $\angle r[n]$ is the phase of the received signal,
- ϕ is the true phase.

7. Decision-Directed and Non-Decision-Directed Loops

Decision-Directed Phase Estimation:

- The detected symbol is obtained from:

$$\hat{s}_{DD}[n] = e^{j\left(\frac{2\pi}{M}\cdot\mathrm{round}\left(\frac{\angle r[n]}{2\pi}\cdot M\right)\right)}$$

- The phase error is updated similarly:

$$\text{Phase Error}_{DD}[n] = \arg\left(\hat{s}_{DD}[n]\cdot e^{-j\hat{\phi}_{DD}[n-1]}\right)$$

- Update rule:

$$\hat{\phi}_{DD}[n] = \hat{\phi}_{DD}[n-1] + K\cdot\text{Phase Error}_{DD}[n]$$

Non-Decision-Directed Phase Estimation:

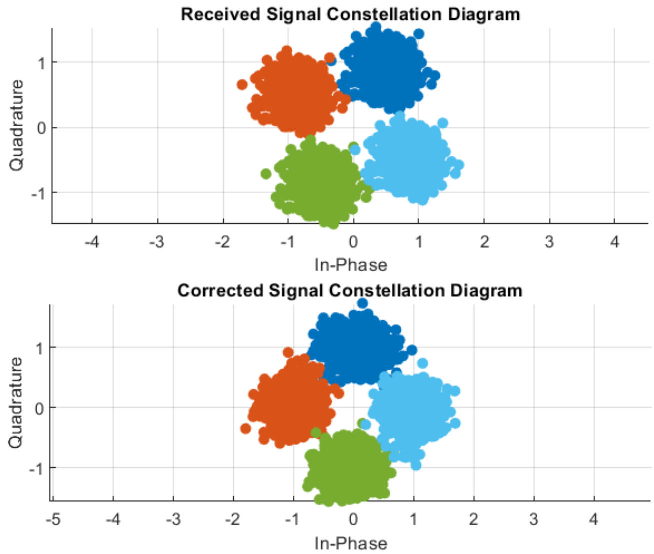
- Phase error calculation is the same as in the PLL:

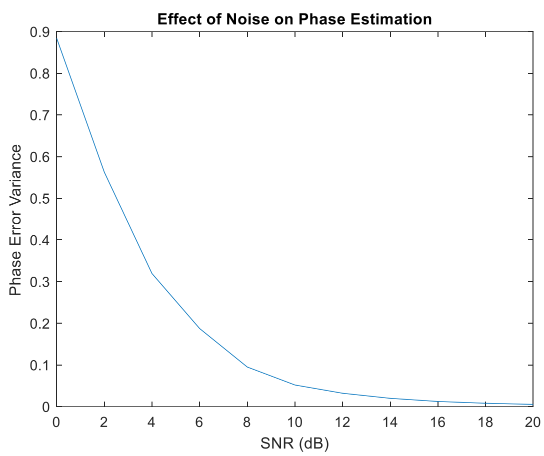
$$\text{Phase Error}_{NDD}[n] = \arg\left(r[n]\cdot e^{-j\hat{\phi}_{NDD}[n-1]}\right)$$

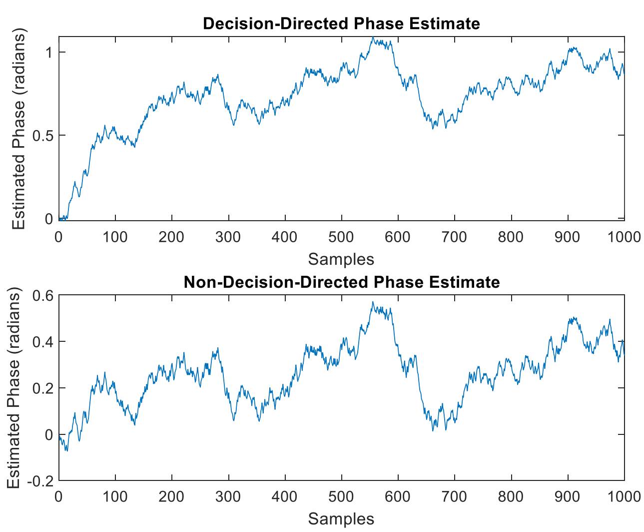
- Update rule:

$$\hat{\phi}_{NDD}[n] = \hat{\phi}_{NDD}[n-1] + K\cdot\text{Phase Error}_{NDD}[n]$$

The sample simulation results are here:







Conclusion:

Sl.No	**Criteria**	Max Marks	Marks obtained
Data sheet			
	Problem statement	10	
	Design & specifications	10	
	Expected output	10	
Record			
D	Simulation/ Conduction of the experiment	15	
E	Analysis of the result.	15	
	Viva	40	
	Total	100	
Scale down to 10 marks			

Experiment - 9
M-ary Phase Shift Keying Modulation and Demodulation using USRP 2920 Software Defined Radio

Aim: To implement M-ary PSK transceiver using USRP 2920 SDR kit.

Tools required: Desktop or Laptop with labview software and USRP 2920 SDR kits.

Theory: Phase shift keying (PSK) is a one of most frequently efficient modulation technique for wireless data transmission for its high noise immunity and narrow frequency bandwidth. Binary phase shift keying (BPSK) is the simplest type of phase keying. It is normally used for high speed data transfer application, provides a 3dB power advantage over the BASK modulation technique and is robust and simple in implementation but proves to be an inefficient user of the provided bandwidth and is normally termed as a non-linear modulation scheme. Fig9.1 shows the block diagram for BPSK transmitter and receiver.

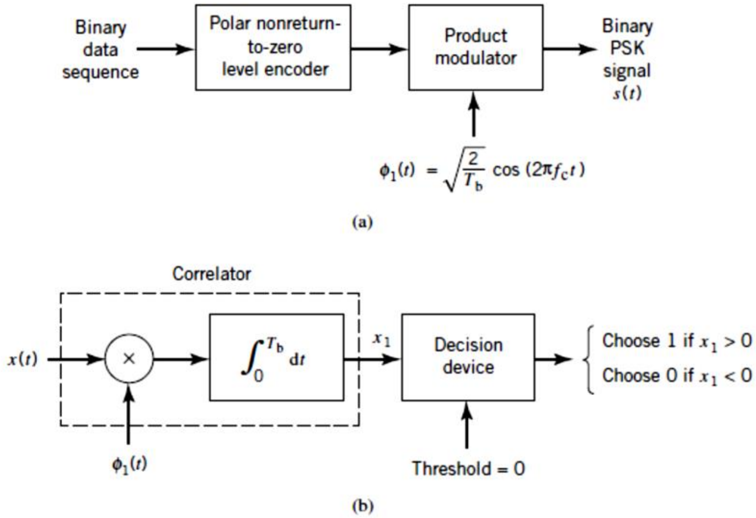


Fig9.1: BPSK (a) Transmitter (b) Receiver

In QPSK, the data bits to be modulated are grouped into symbols, each containing two bits, and each symbol can take on one of four possible values: 00, 01, 10, or 11. During each symbol interval, the modulator shifts the carrier to one of four possible phases corresponding to the four possible values of the input symbol. In the ideal case, the phases are each 90 degrees apart, and these phases are usually selected such that the signal constellation matches the configuration shown in Figure 9.2. and Figure 9.3 shows the message and corresponding QPSK signal.

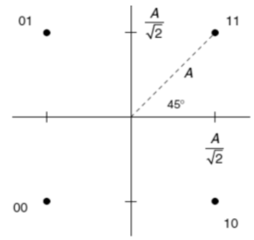


Fig9.2 : QPSK constellation.

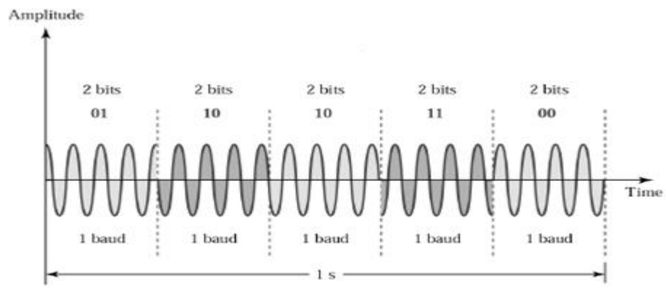


Fig 9.3: QPSK signal and the message

Procedure:

1. The block diagram and front panel for MPSK transmitter with Labview is shown in fig 9.4 and 9.5 respectively. Vary the input parameters and observe the results to understand the significance and applications of MPSK.
2. The block diagram and front panel for MPSK receiver with Labview is shown in fig 9.6 and 9.7 respectively. Vary the input parameters and observe the results to understand the significance and applications of MPSK.

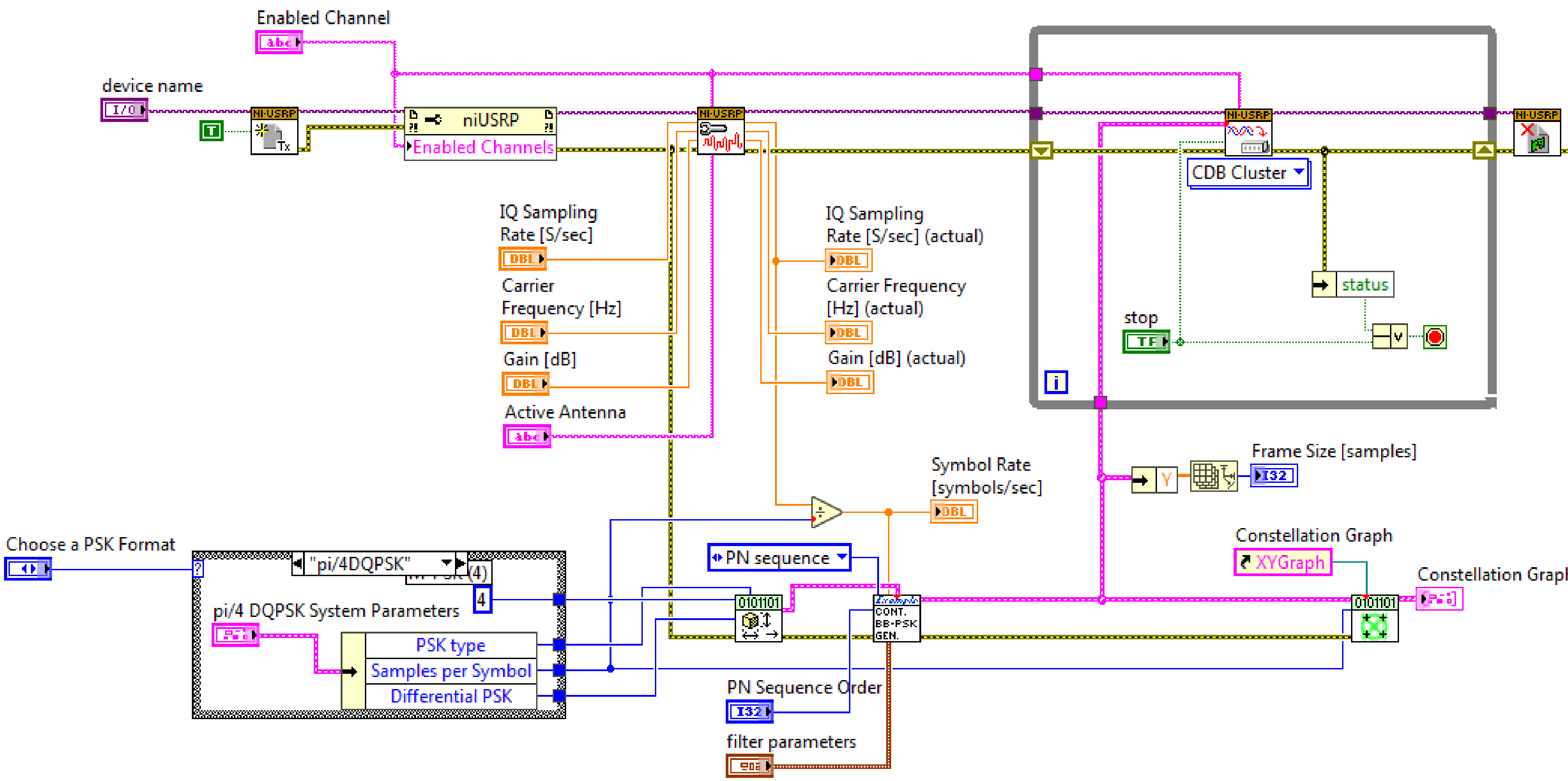


Fig9.4: Block diagram of MPSK transmitter using USRP

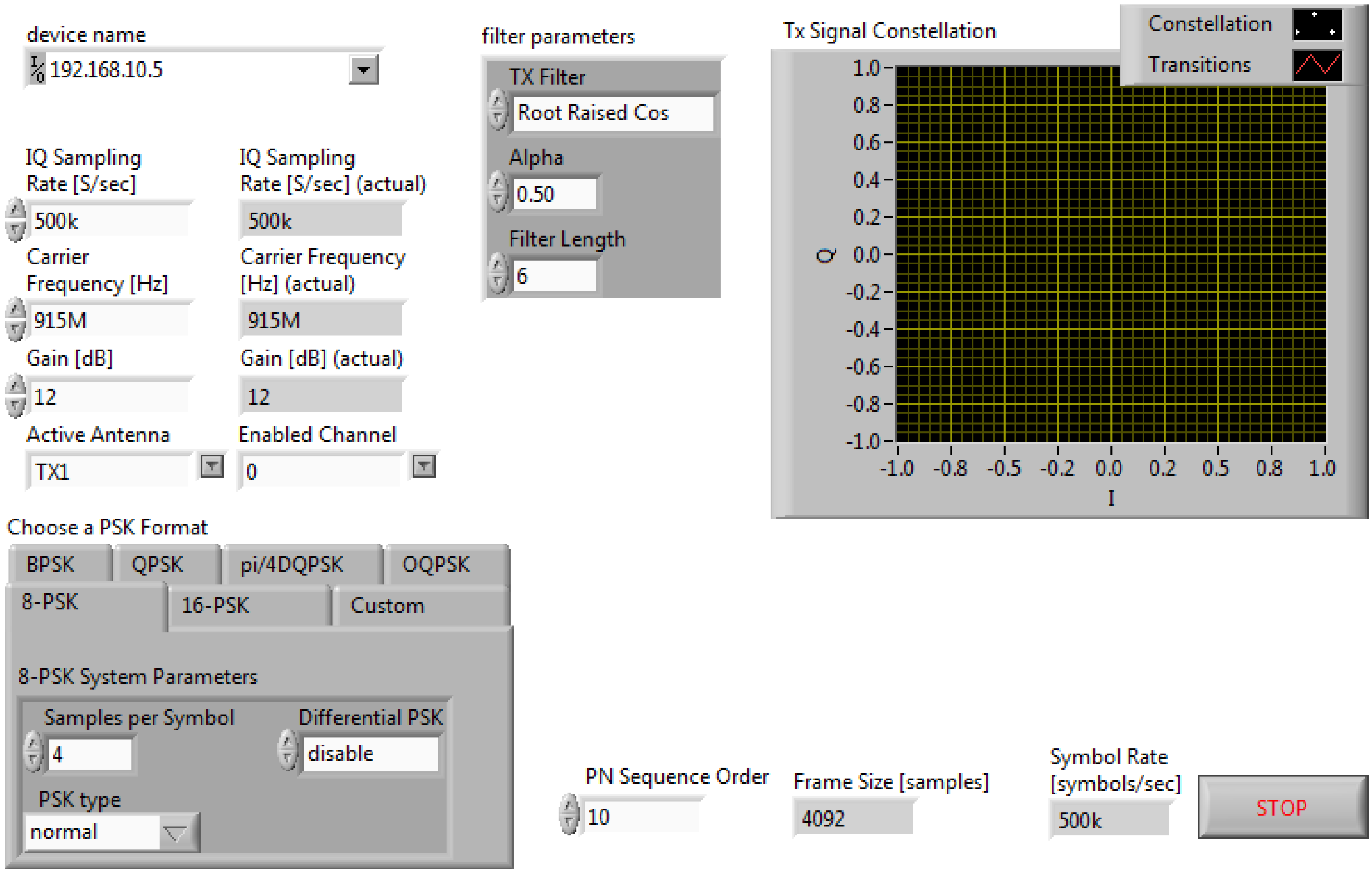


Fig9.5: Front panel of MPSK transmitter using USRP

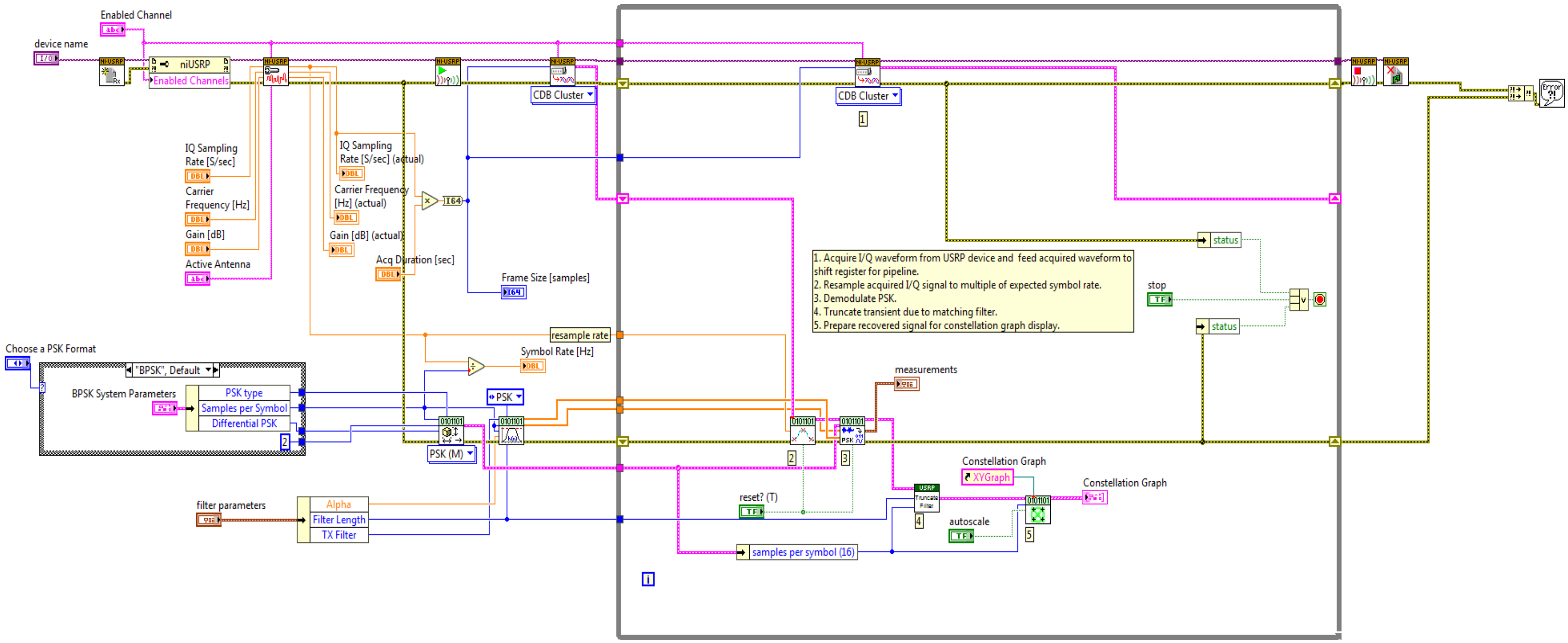


Fig9.6: Block diagram of MPSK receiver using USRP

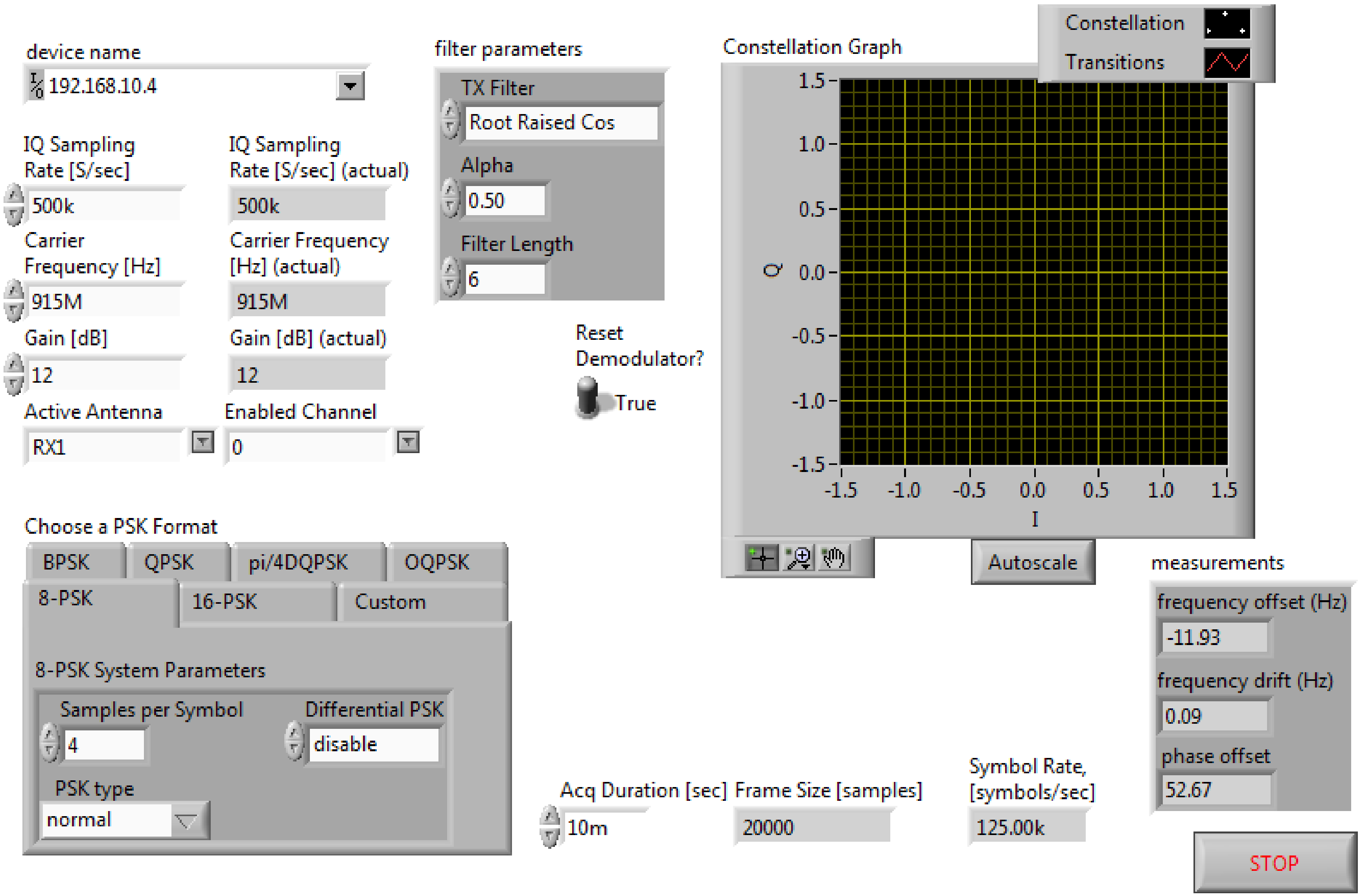


Fig9.7: Front panel of MPSK receiver using USRP

Conclusion:

Sl.No	Criteria	Max Marks	Marks obtained
Data sheet			
A	Problem statement	10	
B	Design & specifications	10	
C	Expected output	10	
Record			
D	Simulation/ Conduction of the experiment	15	
E	Analysis of the result.	15	
	Viva	40	
	Total	100	
Scale down to 10 marks			

Experiment - 10

M-ary QAM Modulation and Demodulation using USRP 2920 Software Defined Radio

Aim: To implement M-ary QAM transceiver using NI-USRP2920 kit

Tools required: Desktop or Laptop with labview software and USRP 2920 SDR kits.

Theory: In an M-ary PSK system, the in-phase and quadrature components of the modulated signal are interrelated in such a way that the *envelope is constrained to remain constant*. This constraint manifests itself in a circular constellation for the message points. However, if this constraint is removed so as to permit the in-phase and quadrature components to be independent, we get a new modulation scheme called *M-ary* QAM. The QAM is a *hybrid* form of modulation, in that the carrier experiences amplitude as well as phase-modulation. In *M*-ary PAM, the signal-space diagram is one-dimensional. *M*-ary QAM is a two-dimensional generalization of *M*-ary PAM. The constellation diagram for QAM with M=16 is shown in fig 10.1.

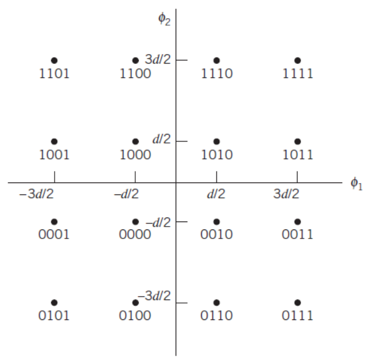


Fig 10.1: Signal-space diagram of *M*-ary QAM for *M* = 16

Procedure:

1. The block diagram and front panel for MPSK transmitter with Labview is shown in fig 10.2 and 10.3 respectively. Vary the input parameters and observe the results to understand the significance and applications of MPSK.
2. The block diagram and front panel for MPSK receiver with Labview is shown in fig 10.4 and 10.5 respectively. Vary the input parameters and observe the results to understand the significance and applications of MPSK.

QAM modulation and demodulation implementation on NI-USRP2920 kit

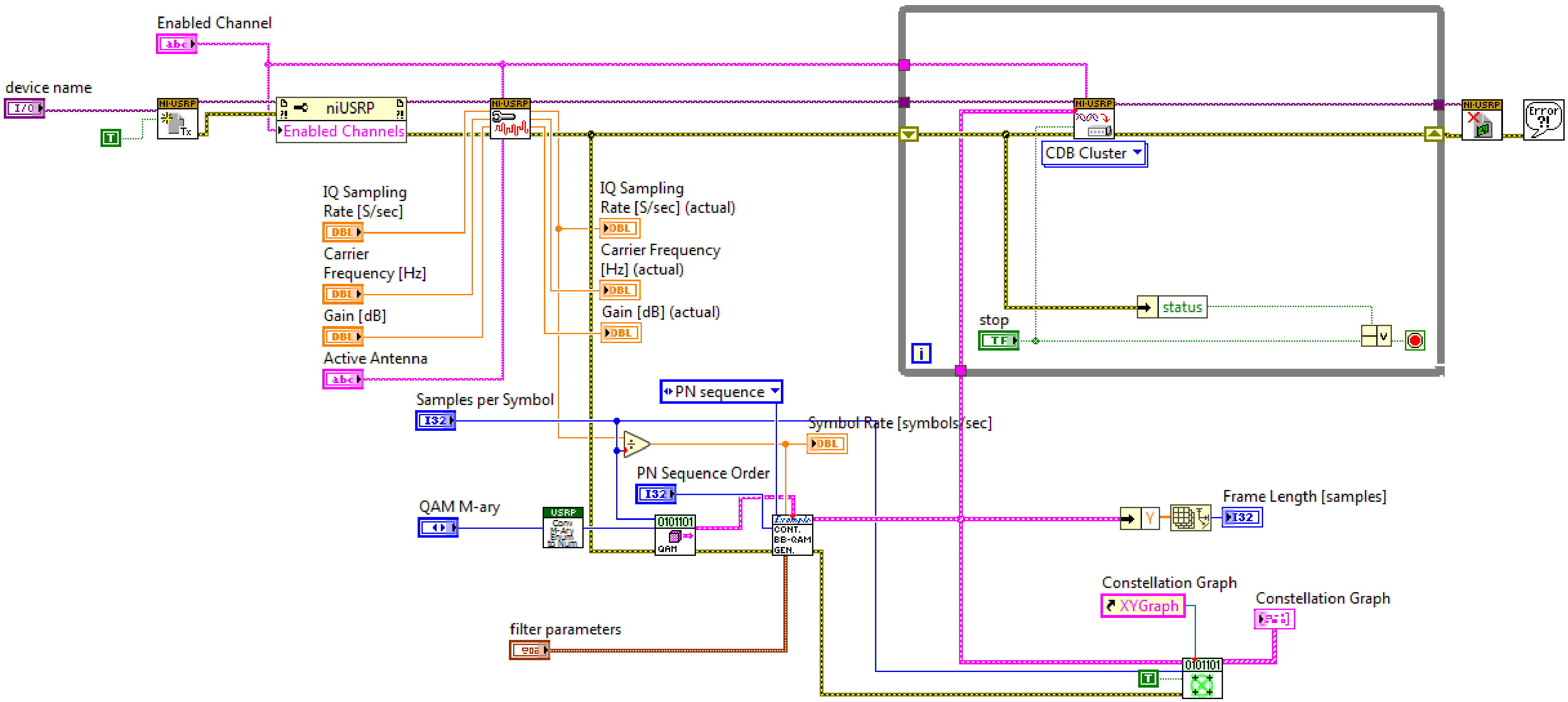


Fig10.2: Block diagram of M-QAM transmitter

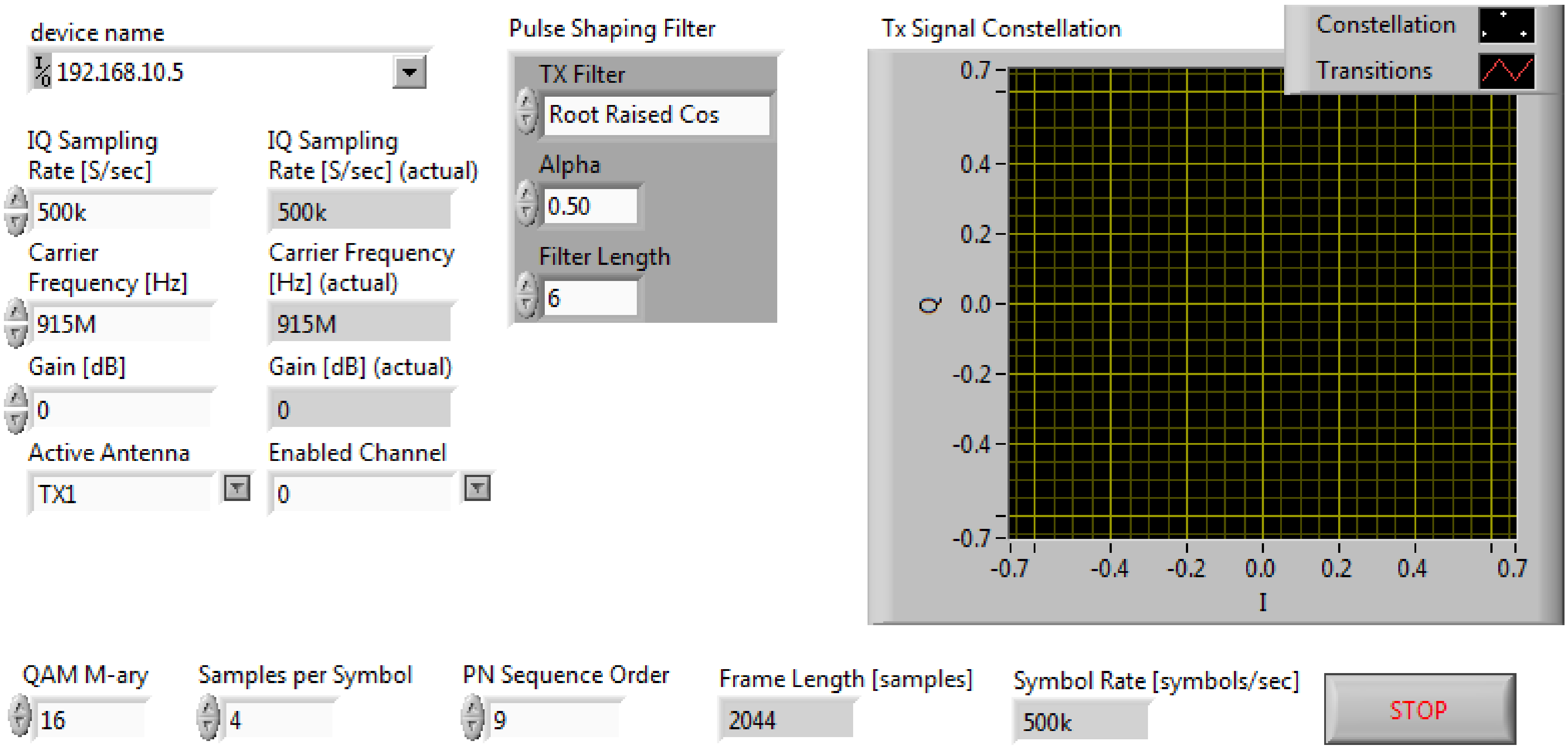


Fig10.3: Front panel of a M-QAM transmitter

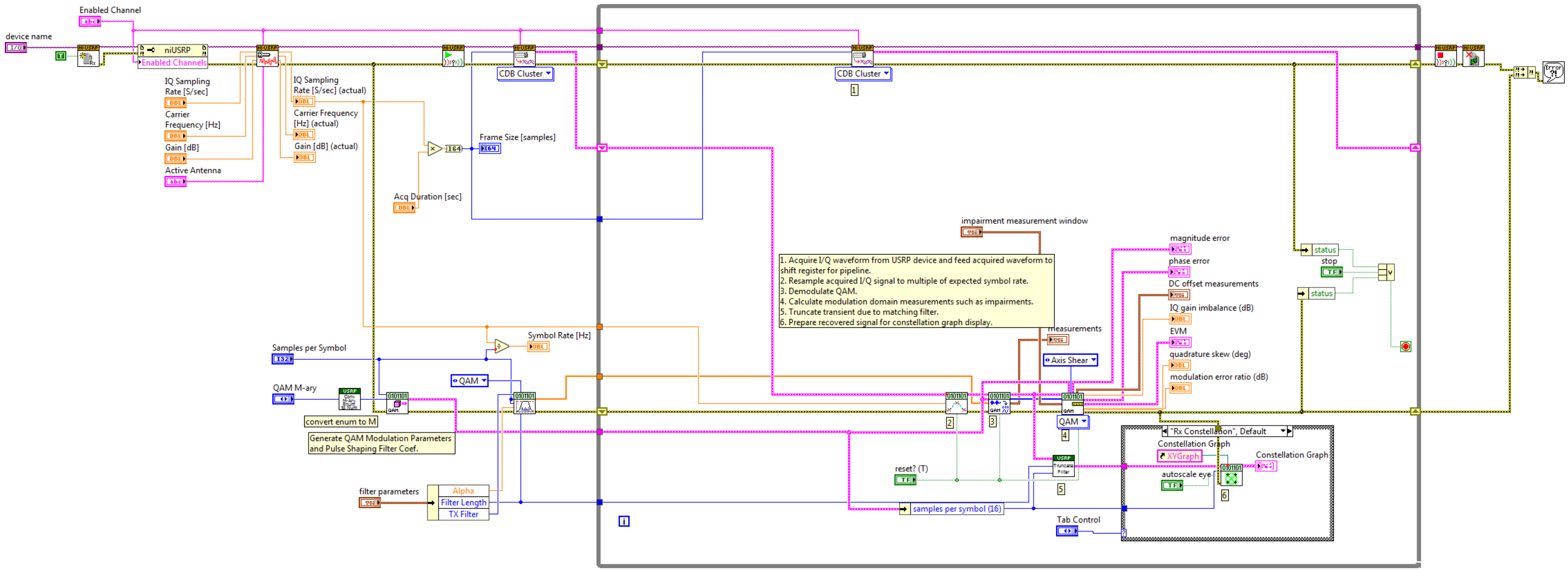


Fig10.4: Block diagram of M-QAM Rx

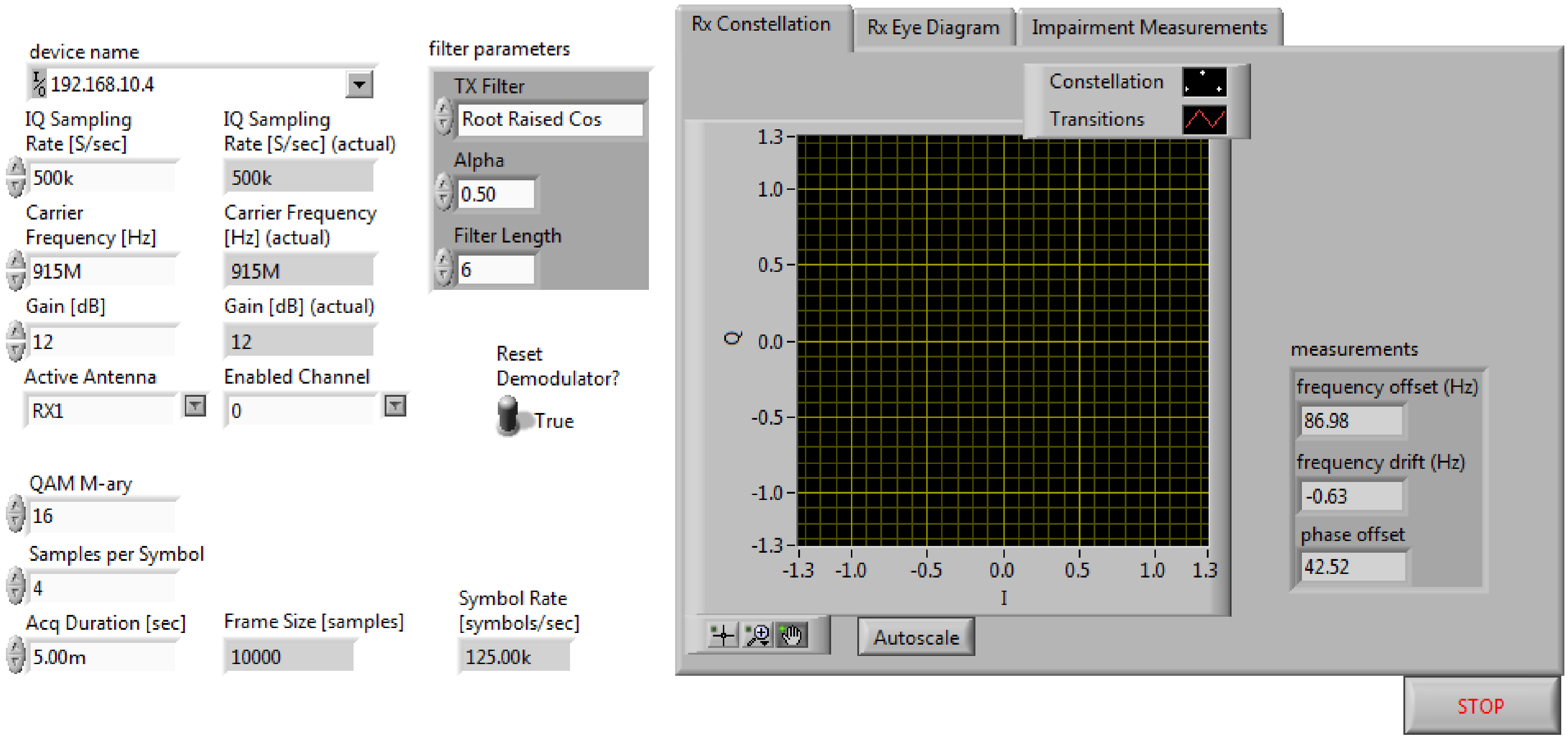


Fig10.5: Front panel of a M-QAM receiver

Conclusion:

Sl.No	Criteria	Max Marks	Marks obtained
Data sheet			
A	Problem statement	10	
B	Design & specifications	10	
C	Expected output	10	
Record			
D	Simulation/ Conduction of the experiment	15	
E	Analysis of the result.	15	
	Viva	40	
	Total	100	
Scale down to 10 marks			

Appendix: Labview Procedure for MPSK and M-QAM

To build an MPSK transceiver, the required sub vi's and their paths in the Labview directory are illustrated in Table1. It is assumed that the Labview software is installed in the path- C:\Program files\ National Instruments\Labview2020. The sub vi's of Table1 are inside this directory. For example, to find the **mod_psk transmitter.vi,** one can use the path as C:\Program files\National Instruments\ Labview2020\ vi.lib\ addons\ Modulation\ Digital\ Modulation. These vi's are used sequentially to construct a block diagram of the M-PSK transceiver as shown in Figure 1. The vi's and their paths required to construct M-QAM are illustrated in Table2. The impairments of a communication system like AWGN (Additive White Gaussian Noise) and IQ(In phase - Quadrature phase) impairments are considered for determining BER, estimating SNR at receiver, estimating demodulated symbols, plotting the transmit spectrum, receive spectrum, constellation diagram and eye diagram.

Table 1: Labview sub-vi paths for MPSK transceiver

Sub vi's	**Path**
Main Blocks	
mod_psk transmitter.vi	vi.lib\addons\Modulation\Digital\Modulation
MT Add AWGN.vi	vi.lib\addons\Modulation\Digital\Impairments
MT Apply IQ Impairments.vi	
MT Demodulate PSK.vi	vi.lib\addons\Modulation\Digital\Demodulation
MT measure PSK Quadrature Impairments.vi	
MT Calculate BER.vi	
MT Calculate EbNo.vi	
Mod_align Tx and Rx bit sequence	vi.lib\addons\Modulation\Digital\support\examples
Case Structure	
Transmitted Symbols\Demodulated Symbols: MT bit stream to digital graph.vi	vi.lib\addons\Modulation\Digital\General
Constellation (Tx Output): MT format Constellation.vi	
MT format Constellation.vi	
Eye Diagram(Rx): MT Format_Eye Diagram.vi	
TX Spectrum: Mod_complex FFT(Mag_Phase).vi	vi.lib\addons\Modulation\Analog\support
Constellation (Rx Output): mod_truncate filter transient.vi	vi.lib\addons\Modulation\Digital\support\Examples

Table 2: Labview sub-vi paths for MQAM transceiver

Sub vi's	Path
Main Blocks	
mod_QAM Transmitter.vi	vi.lib\addons\Modulation\Digital\Support\Examples
MT Add AWGN.vi	vi.lib\addons\Modulation\Digital\Impairments
MT Apply IQ Impairments.vi	
MT Demodulate QAM.vi	vi.lib\addons\Modulation\Digital\Demodulation
MT measure QAM Quadrature Impairments.vi	
MT Calculate BER.vi	
MT Calculate EbN0.vi	
MT Modulate QAM	vi.lib\addons\Modulation\Digital\Modulation
MT Measure Rho	vi.lib\addons\Modulation\Digital\Demodulation
mod_align TX and RX bit sequence	vi.lib\addons\Modulation\Digital\support\examples
Case Structure	
Transmitted Symbols\Demodulated Symbols: MT Bit Stream to Digital Graph.vi	vi.lib\addons\Modulation\Digital\General
Constellation (Tx Output)	
Constellation (Rx Output) mod_Truncate Filter Transient.vi	vi.lib\addons\Modulation\Digital\support\Examples
MT Format Constellation.vi	vi.lib\addons\Modulation\Digital\General
Eye Diagram (Rx) MT Format_Eye Diagram.vi	

<u>**For More Details Pl refer:**</u>

Raghunandan B H, Mahesh A, Mahesha Babu M.P, "Wireless Communication System Design Using Labview and Software Defined Radio", 7th International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), 2023.
(Link to download: https://ieeexplore.ieee.org/document/10334108)

References

1. Simon Haykin, “Digital Communication Systems”, 1st Edition, Wiley, 2013.
2. S. Pasupathy, “Minimum Shift Keying: A Spectrally Efficient Modulation”, IEEE Communications Magazine, vol. 17, no. 4, pp. 14- 22, July 1979.
3. John G. Proakis, Masoud Salehi, Gerhard Bauch, “Contemporary Communication Systems Using MATLAB”, 3rd Edition, Cengage learning, 2013.
4. Sklar, Bernard, Digital Communications, Prentice Hall, 2001.
5. http://www.gaussianwaves.com
6. https://in.mathworks.com/products/communications.html
7. http://www.ni.com/academic/students/learn-labview/
8. https://documentation.help/NI-LabView-Modulation-Toolkit/HelpTitle.html
9. https://www.3blue1brown.com/