Project1：SDR Digital transceivers

Author	Name：宋宜航12112717 张皓东12113010

Introduction

1. Theoretical Knowledge

Complex baseband signal representation

In digital communication, the complex baseband waveform is a fundamental concept used to represent and transmit digital information over a communication channel. It is based on the principle of complex representation, which allows us to conveniently manipulate and analyze real-valued signals using complex mathematics.

The complex baseband waveform is represented as a complex-valued signal, where the in-phase (I) and quadrature (Q) components are combined into a single complex number. The I component represents the real part of the complex waveform, and the Q component represents the imaginary part. Together, they capture the amplitude, phase, and frequency characteristics of the original signal.

The I/Q signal is typically multiplied by the sine and cosine functions (or quadrature carriers) to modulate and demodulate the signal. In modulation, the original signal is frequency converted and phase modulated, and multiplied by the sine and cosine functions to obtain an I/Q signal. In demodulation, the received I/Q signal is multiplied by the local sine and cosine functions, and the original signal is restored by filtering and sampling.

$$\mathrm{I} \cdot cos(2\pi f_c t) - \mathrm{Q} \cdot sin(2\pi f_c t)$$

Packet encapsulation procedure

Packet encapsulation refers to the packaging of raw data into packets (also known as frames or messages) in a specific format in network communication for transmission and reception in a network. Through packet encapsulation, network communication can realize reliable data transmission, error detection and correction, routing and other functions, so that data can be effectively transmitted and exchanged between network nodes.

The format of a data packet encapsulation typically includes several components: guard band, sync sequence, packet number, data, and cyclic pad.

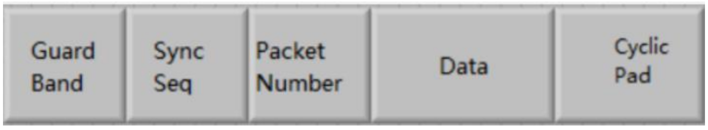
Guard band: The guard band is an additional blank space located at the start and end of the data packet. Its purpose is to provide redundancy to combat noise, multipath interference, and other transmission impairments, ensuring data accuracy and integrity.

Sync sequence: The sync sequence is a predetermined special bit sequence used to aid the receiver in synchronizing when it receives the data packet. The sync sequence helps the receiver correctly determine the start position of the data packet, enabling proper decoding of the subsequent data.

Packet number: The packet number is a unique identifier used to label the data packet. It can be an incrementing counter or another unique identifier. The packet number serves as a reference for the receiver to sort and reassemble the data packets, ensuring their order and integrity.

Data: The data component is the actual digital information to be transmitted. It can be any form of digital data, such as text, audio, images, etc. The data part contains the relevant information to be transmitted, which is encoded, compressed, and encapsulated within the data packet.

Cyclic pad: The cyclic pad is a cyclic redundancy check (CRC) used for error detection and correction during data transmission. It involves adding redundant check bits to the data packet, which are calculated using the CRC technique. The receiver performs the CRC check to ensure data integrity and accuracy.



PSK modulation

The phase shift keying (PSK) modulator is a modulator commonly used in digital communication systems to convert digital data into a phase-modulated signal. The PSK modulator maps each discrete digital symbol to a different phase state, enabling the transmission of data. It includes the procedures as follows:

Determine the phase state: The PSK modulator determines the phase state corresponding to each digital sign based on the PSK modulation scheme used. For example, in BPSK, two phase states, usually 0 degrees and 180 degrees are used representing two different number signs.

Map symbol to phase: Splits the input digital data stream into discrete symbols, each corresponding to a specific numeric value. Depending on the PSK modulation scheme employed, each symbol is mapped to the corresponding phase state. For example, for BPSK, the binary number "0" maps to a 0-degree phase state, while the binary number "1" maps to a 180-degree phase state.

Phase modulation: Depending on the phase state of the map, the PSK modulator modulates the continuous carrier signal to the corresponding phase. Sine waves are often used as carrier signals, expressing different number signs by changing phase. In BPSK, if the current symbol is a 0-degree phase state, then the phase does not change; If the current symbol is a 180-degree phase state, then the phase is reversed.

Filtering and transmitting: To meet bandwidth requirements and suppress noise, PSK modulators typically filter and adjust the modulated signal. The modulated signal is then sent to the receiver via the channel.

PSK demodulation

The phase shift keying (PSK) demodulator is a device or module used in a digital communications system to convert a received phase-modulated signal back to digital data. It demodulates and decodes phase changes during transmission, thereby recovering the original digital data. It includes the procedure as follows:

Receive signal: The PSK demodulator receives a phase-modulated signal from the channel. These signals may experience noise interference, attenuation, and distortion during transmission.

Phase demodulation: The PSK demodulator converts the phase change of the signal into discrete digital symbols by phase demodulating the received signal. The demodulation process involves detecting changes in phase and mapping each phase state to a corresponding numeric value.

Decoding symbols: Demodulated digital symbols are mapped back to the corresponding digital data. Depending on the PSK modulation scheme employed, the demodulator maps each digital symbol back to the corresponding binary digit or other digital representation.

Output digital data: The decoded digital data is provided to the receiving device or system for further processing or application

Principle of frame synchronization

Frame synchronization in digital communication refers to the process of establishing and maintaining alignment with the boundaries of data frames in a received signal. It is crucial for accurately extracting and interpreting the transmitted data. The principle of frame synchronization involves detecting and locating the start and end points of each frame within the received signal. It includes these procedures as follows:

Frame Structure: In digital communication, data is typically organized into frames, which consist of a fixed number of bits or symbols. Each frame contains a header, payload (data), and sometimes a trailer or error detection/correction codes.

Sync Patterns: Frames are often preceded by sync patterns or sync words. These are predetermined bit sequences with known patterns that serve as unique markers to identify the start of a frame.

Frame Detection: The receiver continuously samples and examines the incoming signal to identify the sync pattern. This is achieved by comparing the received signal against the expected sync pattern using correlation or pattern matching techniques.

Error Tolerance: Due to noise, channel impairments, or timing variations, the received signal may experience distortions or slight misalignments. Therefore, frame synchronization algorithms are designed to tolerate a certain degree of errors or deviations in the detected sync patterns.

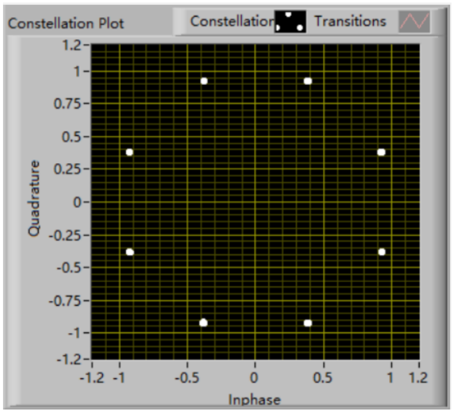
Synchronization Maintenance: Once the initial frame synchronization is achieved, the receiver continuously monitors and adjusts its timing based on the detected sync patterns within subsequent frames. This ensures that the receiver remains synchronized with the incoming data stream over time.

Frame synchronization is essential for various operations in digital communication, including demodulation, decoding, error detection, and payload extraction. By establishing accurate frame boundaries, it enables reliable and efficient processing of the transmitted data, ensuring the correct interpretation and utilization of the received information.

Constellation chart and eye chart

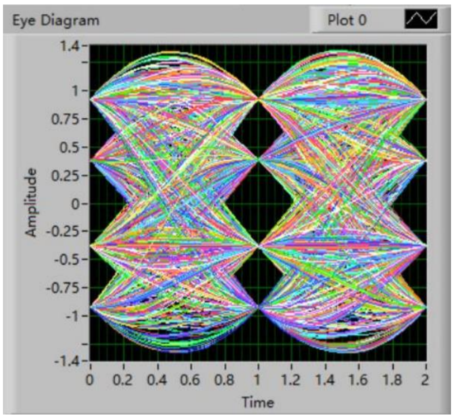
Constellation chart and eye chart are all the common tools used in communication system analysis to present the performance of the communication system.

The constellation chart, also known as a constellation diagram or a signal space diagram, is a graphical representation of the signal points or symbols used in a digital communication system. It is commonly used in modulation schemes such as Quadrature Amplitude Modulation (QAM) and Phase Shift Keying (PSK). In a constellation chart, each signal point represents a unique combination of amplitude and phase (for QAM) or phase (for PSK). The chart typically consists of a grid or a set of axes, where the horizontal axis represents the in-phase component and the vertical axis represents the quadrature component of the signal. The signal points are plotted at specific locations on the chart based on their amplitude and phase values.



Eye diagram is a graphical representation of a digital signal's quality and characteristics. It provides valuable insights into the signal's integrity and helps evaluate the performance of the communication system. The eye diagram is commonly used to analyze and diagnose various issues such as noise, jitter, and distortions in the received signal. The eye diagram is constructed by overlaying multiple samples of the transmitted signal in a synchronized manner. The samples are typically taken at the receiver and correspond to one bit period of the digital signal. The horizontal axis represents time, and the vertical axis represents the signal voltage or amplitude.

The resulting pattern resembles an "eye" shape, hence the name "eye diagram." The open area in the center of the eye represents the most probable levels or voltage thresholds for correct signal detection. This region is commonly referred to as the "eye opening" or "eye width" and indicates the signal's reliability. By examining the eye diagram, various signal characteristics can be assessed. For instance, a wide and well-defined eye opening suggests a clean and robust signal, indicating good signal quality. On the other hand, a narrow or distorted eye opening indicates signal impairments such as noise, inter-symbol interference (ISI), or timing errors.

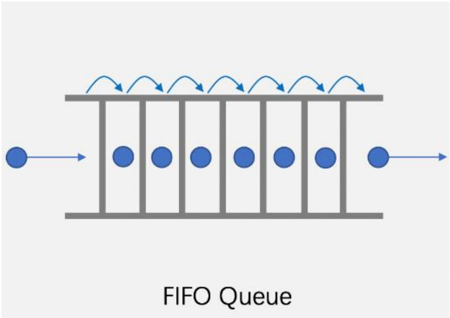


Principle of Queue

A queue is a common data structure that organizes and stores data according to the principle of First-In-First-Out (FIFO). A queue can be seen as a finite-length linear list, where data enters the queue from one end (known as the rear or tail) and exits from the other end (known as the front or head).

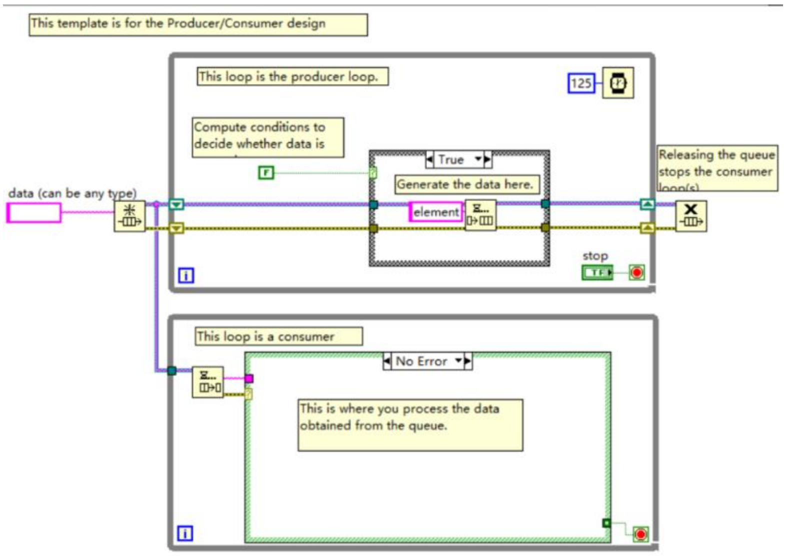
In a queue, the insertion operation of data items is called enqueue, while the removal operation is called dequeue. The enqueue operation adds a new data item to the rear of the queue, while the dequeue operation removes and returns the data item at the front.

Queues play a crucial role in the simulation of digital communication systems, primarily for simulating and processing real-time data streams. In this experiment, the queues are frequently used in the simulation of LabVIEW to transform the data among different modules and programs.



Producer-consumer module

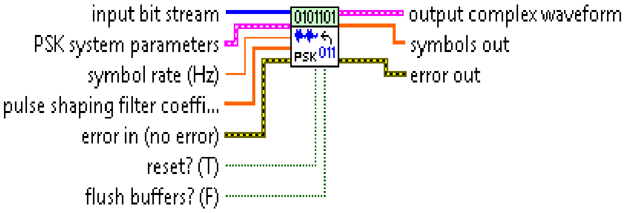
In LabVIEW, the Producer-Consumer module is a design pattern used for communication and coordination between multiple parallel tasks or modules. It is commonly employed in simulation and modeling of digital communication systems to handle the flow of data between different components. The Producer-Consumer module involves two main entities: the producer and the consumer. The producer generates or produces data, while the consumer consumes or processes the data. The communication between them is typically facilitated through a shared buffer or queue. In LabVIEW, the producer module continuously generates data and places it in the shared queue. The consumer module, on the other hand, retrieves data from the queue and performs specific tasks or calculations on it. This decoupling of the producer and consumer modules allows them to operate independently and at their own pace. The Producer-Consumer module is especially useful in digital communication system simulations because it enables the simulation of real-time data flow and processing. It allows for the efficient handling of data streams, mimicking the behavior of actual communication systems. By employing the Producer-Consumer module, LabVIEW simulations can accurately replicate the flow of data between different components of a digital communication system. It helps in evaluating system performance, analyzing bottlenecks, and optimizing the overall system design.



2. LabVIEW Modules

1) MT Modulate PSK VI

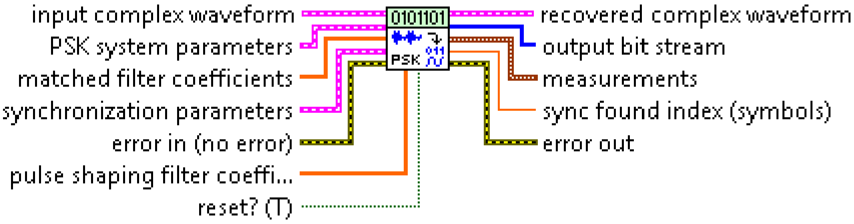
MT Modulate PSK VI is a module in LabVIEW, a graphical programming environment, designed to perform PSK modulation. With the MT Modulate PSK VI, users can specify the modulation parameters, such as the modulation order (e.g., BPSK, QPSK), the symbol rate, and the carrier frequency. The VI then takes the digital input data and maps it to the corresponding PSK symbols according to the selected modulation scheme. It generates the modulated signal by varying the phase of the carrier signal based on the PSK symbols. The MT Modulate PSK VI also allows users to configure additional parameters, such as the amplitude of the carrier signal and the choice of polar or non-polar PSK.



2) MT DeModulate PSK VI

MT DeModulate PSK VI is another module in LabVIEW that is used for demodulating PSK-modulated signals. It provides a comprehensive set of tools for recovering the original digital data from the received PSK signal.

The MT DeModulate PSK VI takes the received PSK signal as input and performs signal processing operations to extract the phase information embedded in the signal. It employs various demodulation techniques, such as phase detection and symbol synchronization, to accurately demodulate the signal and recover the original data symbols.

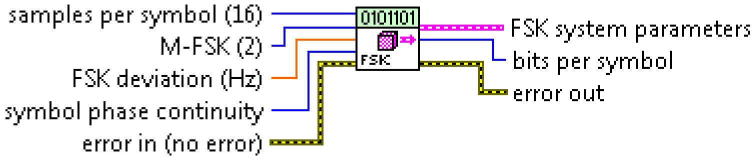


3) MT Generate System Parameters VI

MT Generate System Parameters VI is a module in LabVIEW used to set up and define the parameters of a digital communication system. This VI allows users to configure essential system parameters required for the operation of the communication system.

The MT Generate System Parameters VI enables users to specify parameters such as the carrier frequency, symbol rate, modulation type (e.g., BPSK, QPSK), and coding scheme. Users can also define other system-specific parameters, including channel characteristics, noise levels, and synchronization methods. By customizing these parameters, users can create a system that matches their specific communication requirements.

The output of the MT Generate System Parameters VI is a set of system parameters that can be used as input for other modules or VIs in LabVIEW for further signal processing and analysis.

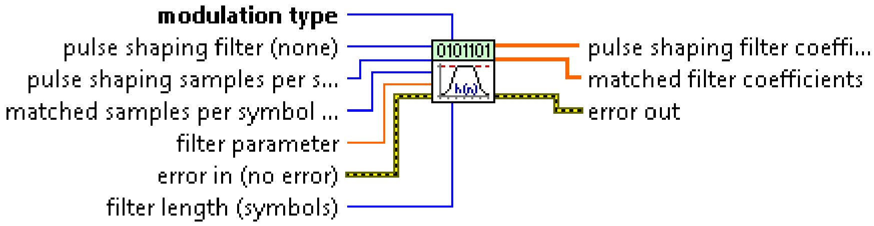


4) MT Generate Filter Coefficients VI

MT Generate Filter Coefficients VI is a module in LabVIEW used to generate filter coefficients for digital filters. This VI provides a convenient and efficient way to design and create filter coefficients based on user-defined filter specifications.

With the MT Generate Filter Coefficients VI, users can specify filter parameters such as filter type (e.g., low-pass, high-pass, band-pass), cutoff frequency, filter order, and desired filter characteristics (e.g., pass-band ripple, stop-band attenuation). The VI then calculates the filter coefficients using various filter design algorithms, such as windowing, Parks-McClellan, or least squares methods.

The generated filter coefficients can be used to implement digital filters for tasks such as signal filtering, equalization, or modulation/demodulation in a digital communication system.

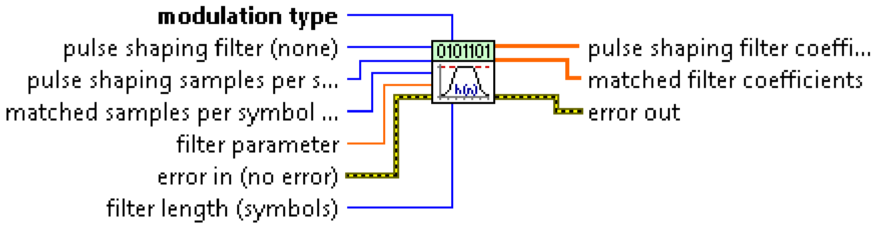


5) MT Generate Synchronization Parameters VI

MT Generate Synchronization Parameters VI is a module in LabVIEW used to generate and configure synchronization parameters for digital communication systems. This VI allows users to define and optimize parameters related to synchronization, which is crucial for accurate signal detection and demodulation.

The MT Generate Synchronization Parameters VI enables users to set parameters such as timing synchronization, carrier frequency synchronization, and symbol timing recovery. These parameters can be customized based on specific system requirements and signal characteristics. The VI provides options for different synchronization algorithms, such as matched filtering, correlation, or maximum likelihood estimation.

By using the MT Generate Synchronization Parameters VI, users can fine-tune the synchronization process to ensure reliable and accurate signal detection, demodulation, and data recovery in digital communication systems.

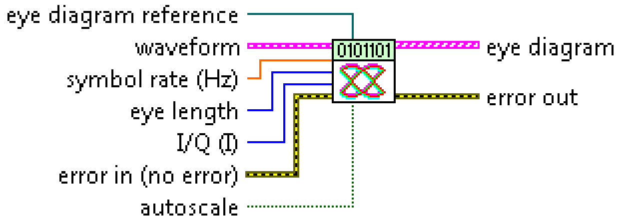


6) MT Format Eye Diagram VI

MT Format Eye Diagram VI is a module in LabVIEW used for visualizing and analyzing the eye diagram of a digital communication system. The eye diagram provides a graphical representation of signal quality, including noise, distortion, and timing jitter.

The MT Format Eye Diagram VI takes the received digital signal as input and processes it to generate the eye diagram. It samples the signal at different time instants and plots the sampled values on a graph. By repeating this process for multiple signal periods, an eye-shaped pattern is formed, which represents the overlapped waveforms of the signal.

The eye diagram allows users to evaluate various system parameters, such as signal quality, timing offset, and noise margin. It provides insights into the integrity of the transmitted signal, making it easier to detect and diagnose issues related to synchronization, distortion, or noise.



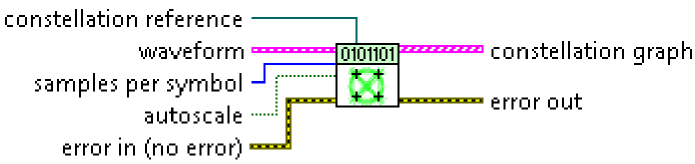
7) MT Format Constellation VI

MT Format Constellation VI is a module in LabVIEW used to visualize and analyze the constellation diagram of a modulated signal in a digital communication system. The constellation diagram displays the amplitude and phase relationships of the signal's symbols.

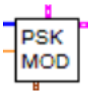
The MT Format Constellation VI takes the received modulated signal as input and processes it to generate the constellation diagram. It maps the received symbols onto a two-dimensional graph, where the x-axis represents the in-phase component and the y-axis represents the quadrature component. Each symbol is plotted as a point on the graph, indicating its amplitude and phase.

The constellation diagram provides valuable information about the signal quality, such as signal distortion, noise interference, and symbol spacing. It helps users analyze and optimize modulation and demodulation schemes, identify performance issues, and evaluate the overall system performance.

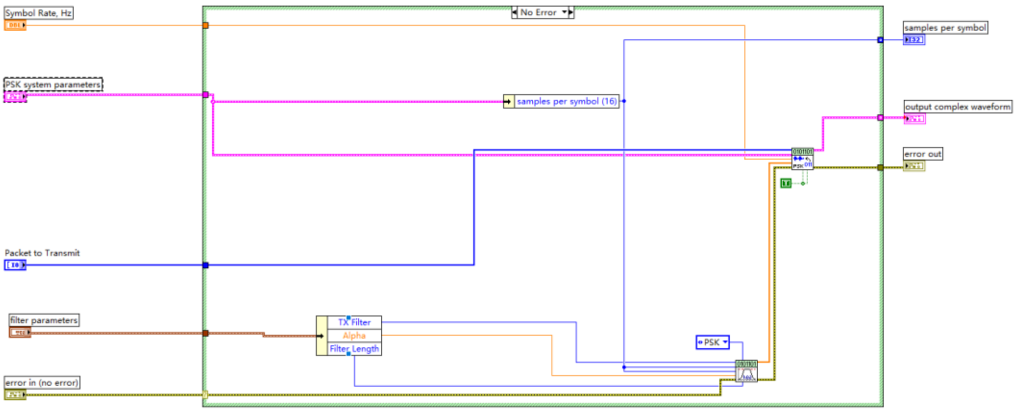
By using the MT Format Constellation VI, users can visually examine and interpret the characteristics of the modulated signal, enabling effective troubleshooting and improvement of digital communication systems.



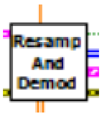
8) sub_PSKMod_Temp VI



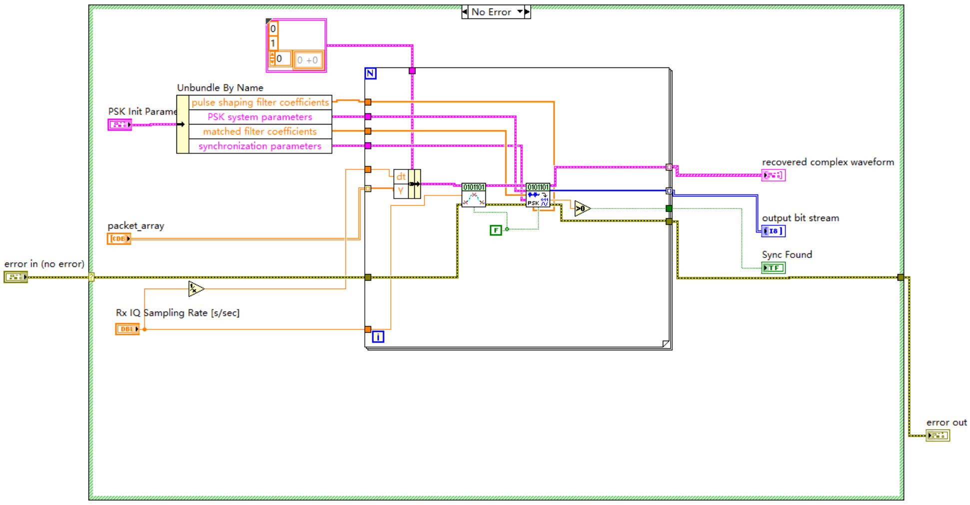
This module is an important part of modulation, which is used to modulate and pulse shape the data in the transmitted data packet. The input is a variety of parameters, which are encapsulated in a cluster structure. The data packets that need to be transmitted are also imported into the input, and then MT Generate Filter Coefficients VI is used to generate a function based on the input parameters, usually using the raised cosine roll down function, and then the convolution operation is performed in MT Modulate PSK VI. The pulse forming function is formed and output as a complex baseband waveform. The above modules together constitute this module.



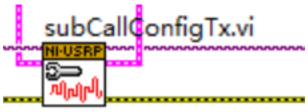
9) sub_resample_and_demodulate_Temp VI



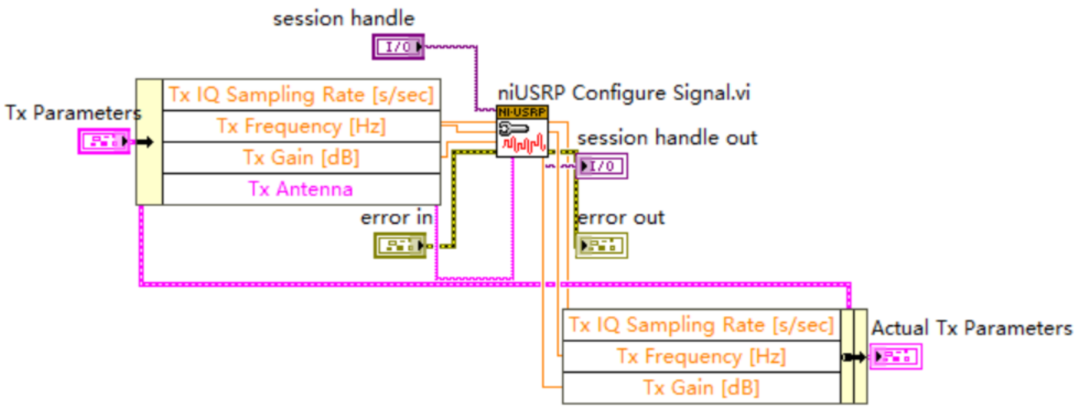
This module is an important part of the demodulation process. Firstly, the parameters are input at the input end, which is encapsulated in a cluster structure. After that, a Resample operation is required. MT Resample (Complex Cluster) VI is used for resample operation, the resample waveform is output, and then the next MT Demodulate PSK VI is imported for demodulation operation. Output the demodulated waveform and bit stream. The above modules and operations together constitute the module.

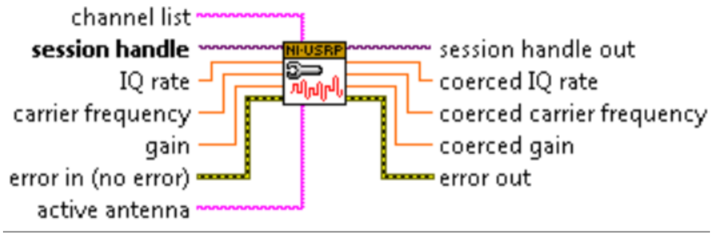


10) subCallConfigTx VI

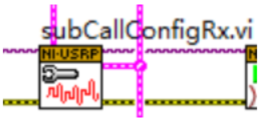


The module is used to configure various parameters of the transmit (TX) side of the USRP. Its core is based on the official NIUSRP Configure Signal VI, which is a LabVIEW module used to configure signal parameters of the USRP device. However, unlike the official VI, this module packages all the required input and output parameters into a cluster structure, making it convenient for parameter input.

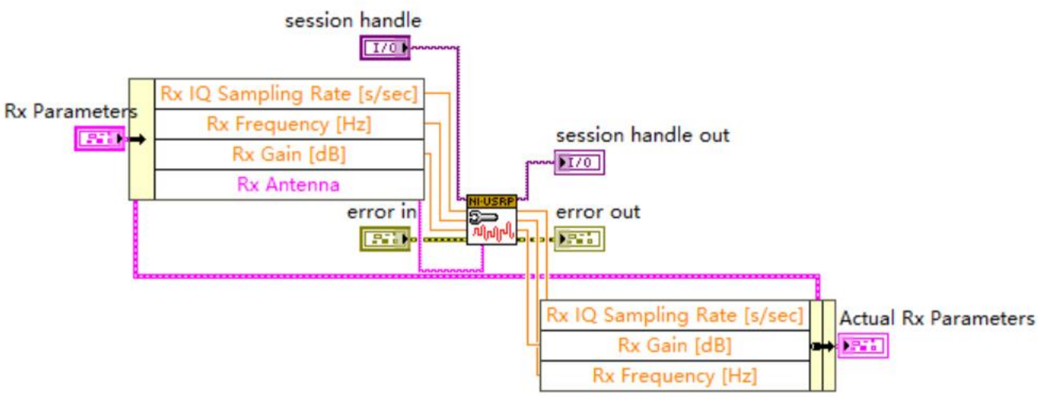


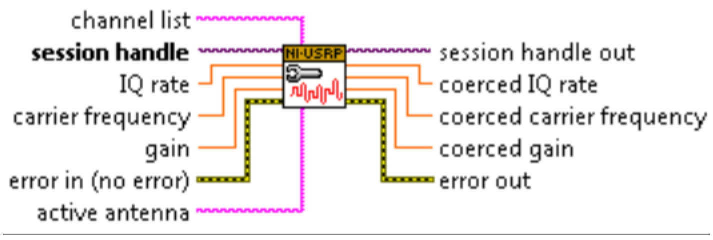


11) subCallConfigRx VI



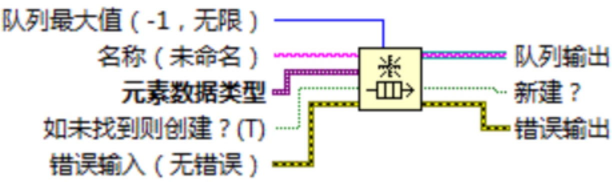
The module is used to configure various parameters of the receiver (RX) side of the USRP. Its core is based on the official NIUSRP Configure Signal VI, which is a LabVIEW module used to configure signal parameters of the USRP device. However, unlike the official VI, this module packages all the required input and output parameters into a cluster structure, making it convenient for parameter input.





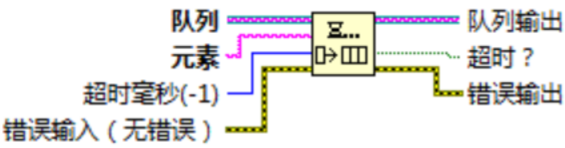
12) Obtain Queue VI

The "Obtain Queue" module can be used to create a new queue in LabVIEW. At the input side, data type can be specified according to the maximum length of the queue. This allows to create a queue of the specified length with the data type determined by the input. At the output side, the created queue can be exported for further operations in subsequent steps.



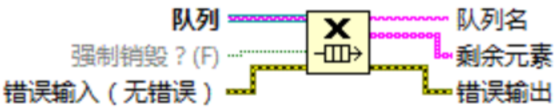
13) Enqueue Element VI

The "Enqueue Element" module can be used to add new elements to the rear end of an existing queue. The module takes as inputs the element to be added and the queue to which it should be added. This allows the element to be inserted at the rear end of the queue, maintaining the FIFO (First-In-First-Out) rule of the queue. At the output side, the updated queue with the added element can be exported for further operations in subsequent steps.



14) Dequeue Element VI

The "Dequeue Element" module can be used to retrieve elements from a queue while adhering to the FIFO (First-In-First-Out) rule of the queue. This module ensures that elements are dequeued in the order they were enqueued, allowing for a gradual removal of elements from the queue and the release of associated resources and references.



Lab results & Analysis：

1. Complex baseband waveform of a data packet

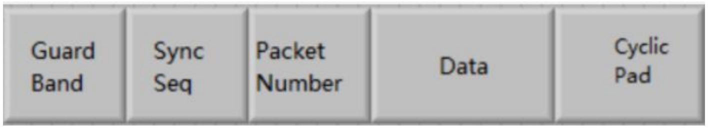
In this experiment, data is transmitted in the form of data packets. In this digital communication experiment, data packets usually have a fixed format, which usually includes the following parts:

Guard band: The guard band is an additional blank space located at the start and end of the data packet. Its purpose is to provide redundancy to combat noise, multipath interference, and other transmission impairments, ensuring data accuracy and integrity.

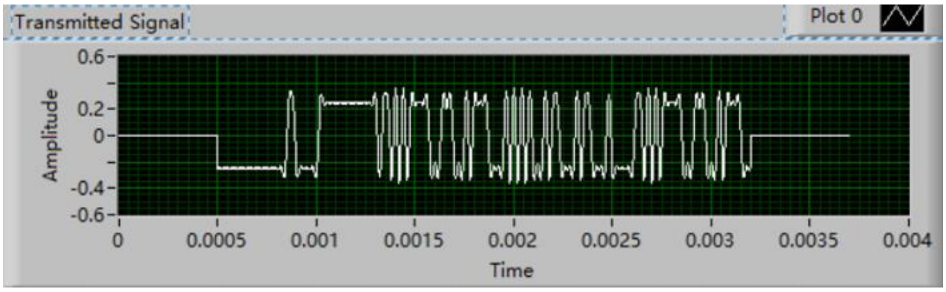
Sync sequence: The sync sequence is a predetermined special bit sequence used to aid the receiver in synchronizing when it receives the data packet. The sync sequence helps the receiver correctly determine the start position of the data packet, enabling proper decoding of the subsequent data.

Packet number: The packet number is a unique identifier used to label the data packet. It can be an

incrementing counter or another unique identifier. The packet number serves as a reference for the receiver to sort and reassemble the data packets, ensuring their order and integrity.
Data: The data component is the actual digital information to be transmitted. It can be any form of digital data, such as text, audio, images, etc. The data part contains the relevant information to be transmitted, which is encoded, compressed, and encapsulated within the data packet.
Cyclic pad: The cyclic pad is a cyclic redundancy check (CRC) used for error detection and correction during data transmission. It involves adding redundant check bits to the data packet, which are calculated using the CRC technique. The receiver performs the CRC check to ensure data integrity and accuracy.

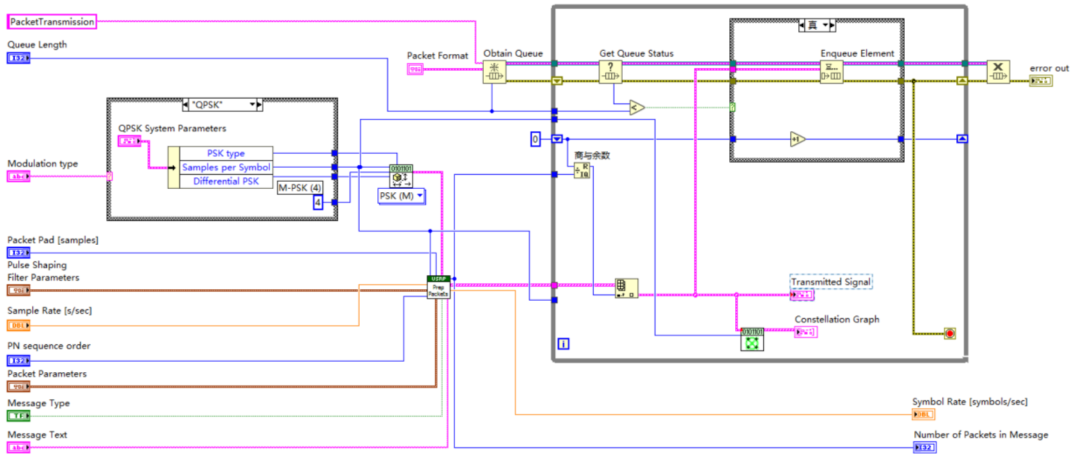


The data in these packets will be mapped into complex baseband waveforms, through which we can see the presence of different components in the packet. The first half of the waveform is a line with a amplitude equal to zero, because this represents the guard band. After this bit of zero is a synchronization sequence, which we can also see in the waveform diagram. The following section is the effective data bit, and the waveform obtained after the convolution of the filter function can be seen as the shape of the raised cosine roll down function. At the end is another line of zero amplitude, which is the zero bits of the zero pad. Increasing the value of guard band and sync sequence shows that the waveform will be more stable and less susceptible to interference.

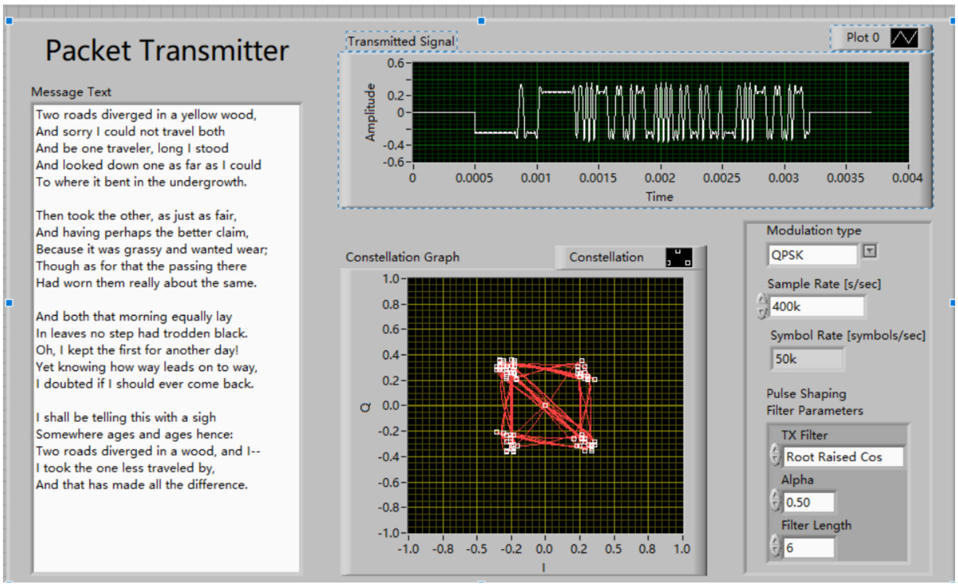


2. Queue-based text transfer emulation

QPSK
TX:

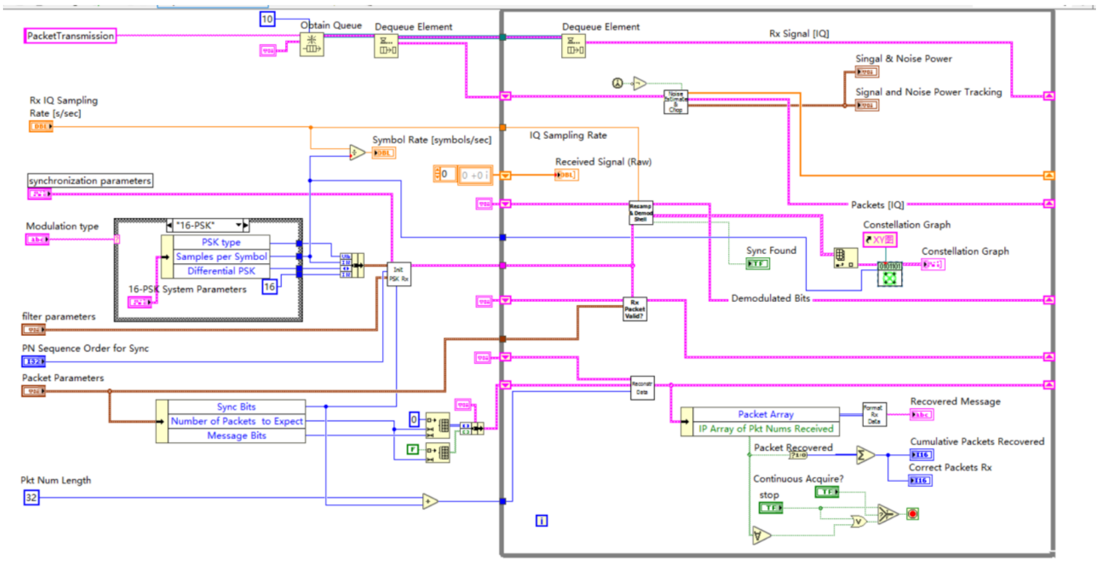


The block diagram of the queue-based text transfer program is shown in the figure above, using the sub-vi module of the queue in the synchronous function to build a transmitter. First, symbol mapping is performed through the given PSK modulation method, and then the pulse forming function is generated, encapsulated in the packet, and transmitted to the queue in the form of a packet. The queue arranges these packets and transmits them in the manner of FIFO.

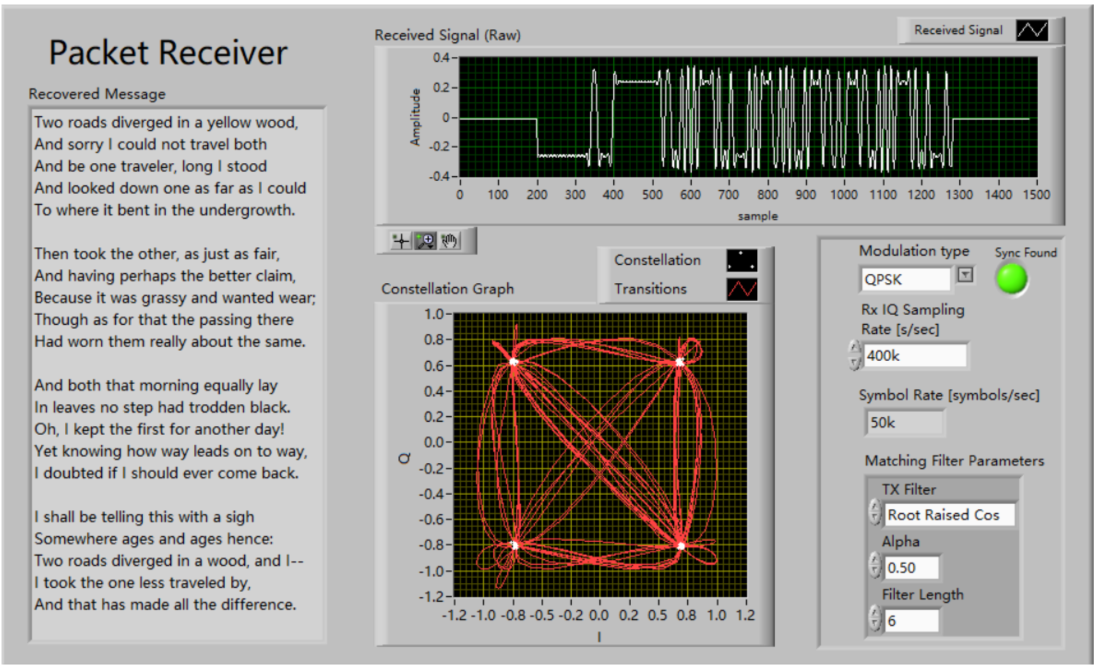


The setting parameters are shown in the figure, set to QPSK modulation, the transmitted signal waveform and constellation diagram are as above, and the TX side uses Root Raised Cosine for filtering, and then pulse molding.

RX:



The RX block diagram interface is shown in the figure. Queue names are consistent and queue calls can be made between different programs on the same PC. The receiver uses the Dequeue function to extract packets in the queue and demodulate them according to the original modulation mode



The text decoding result obtained is as shown in the figure, the decoding rate is completed with 100% accuracy, and the position of the point in the constellation diagram is in line with QPSK modulation

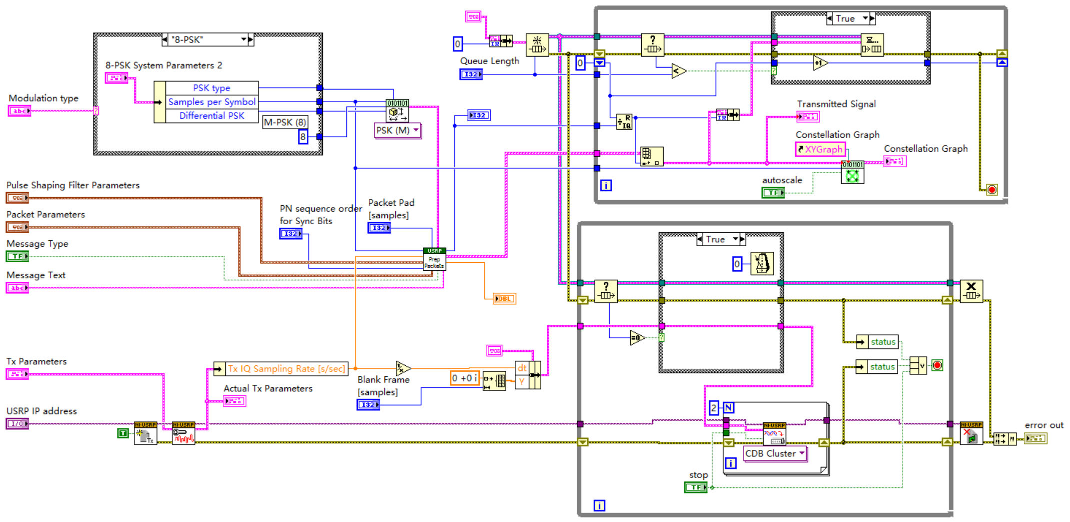
Producer-consumer model

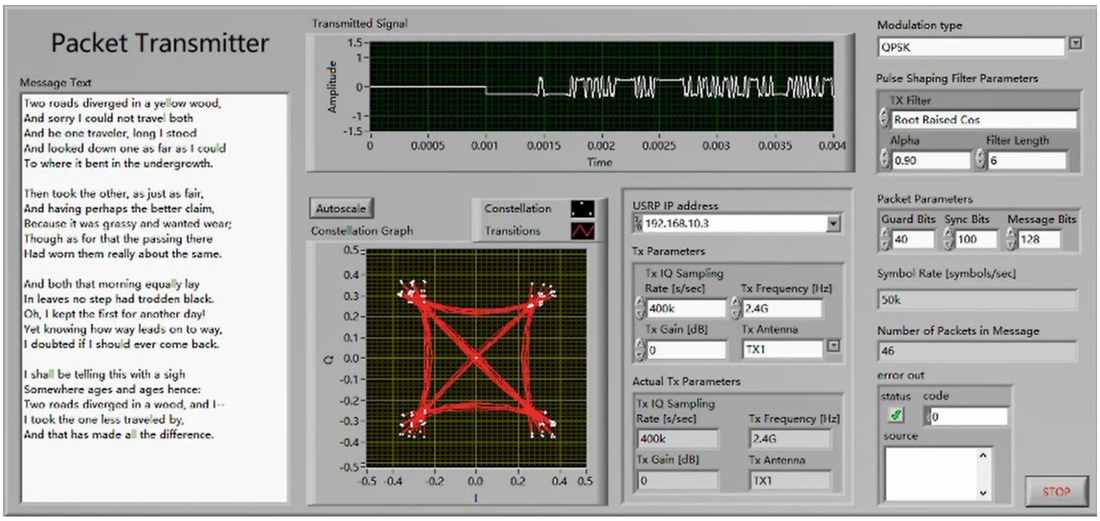
TX is partially equivalent to the producer model, and the enqueue function passes packets into the queue, so that data is bloomed and transmitted, and can generate a signal to stop consumers from accepting data. RX is equivalent to the consumer model, and the dequeue function extracts packets from the queue for processing in turn according to the principles of FIFO.

3. USRP-based packet transmission

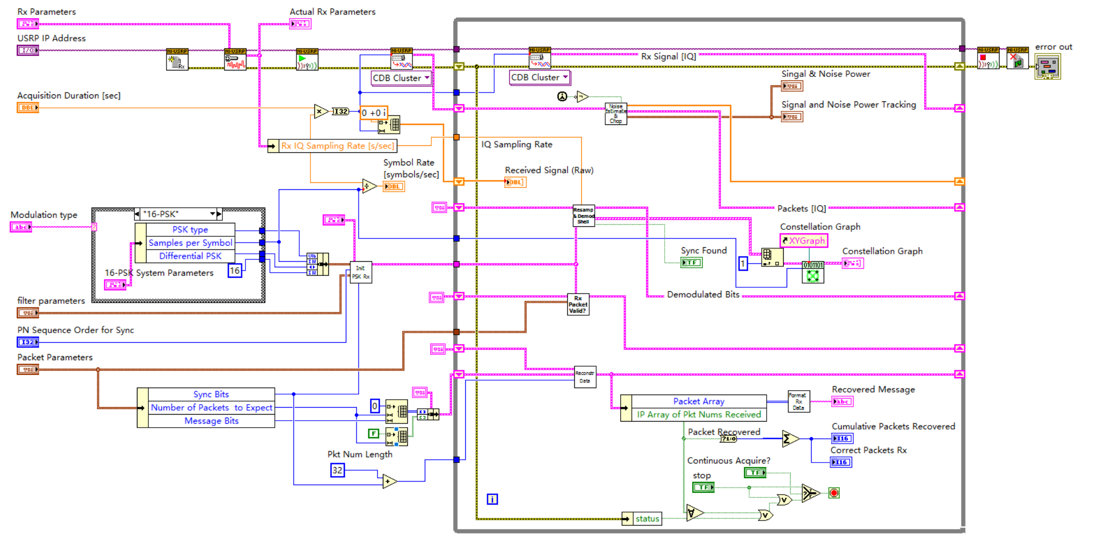
QPSK

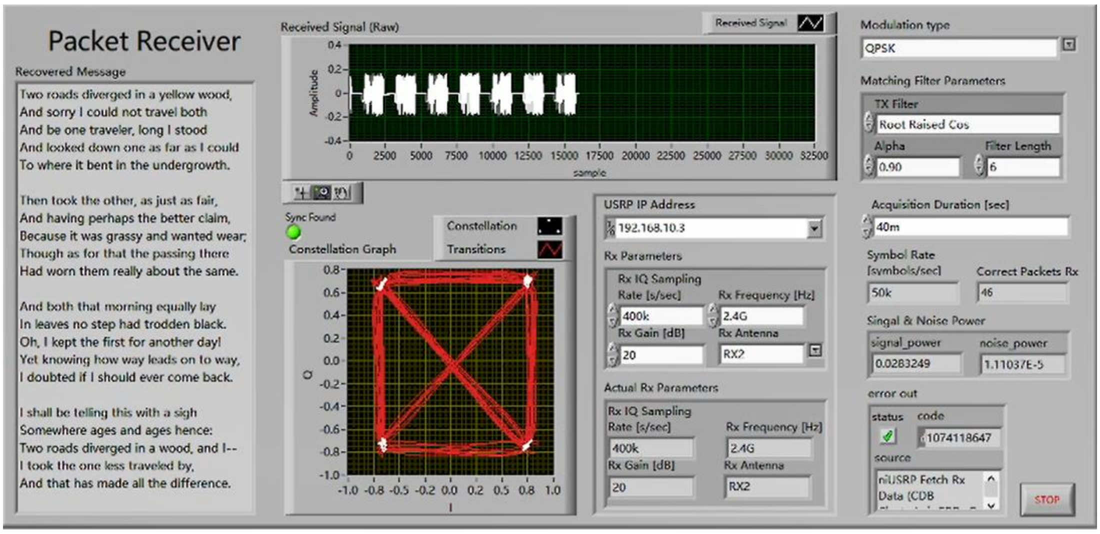
TX:





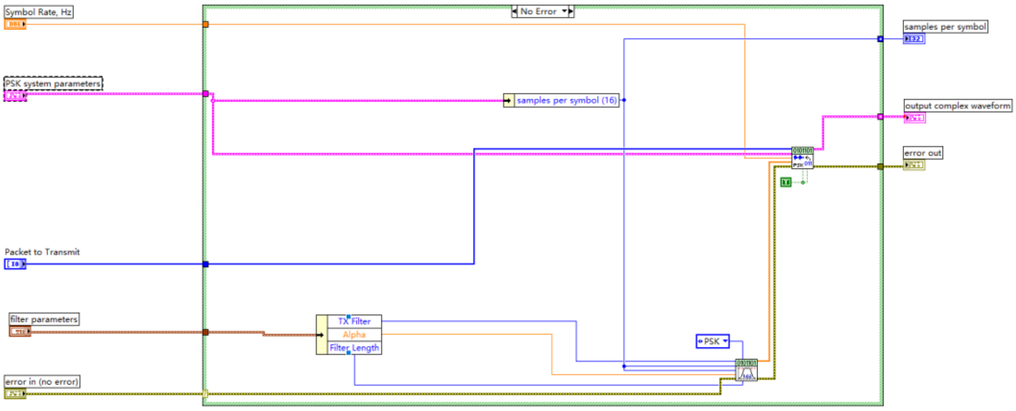
RX

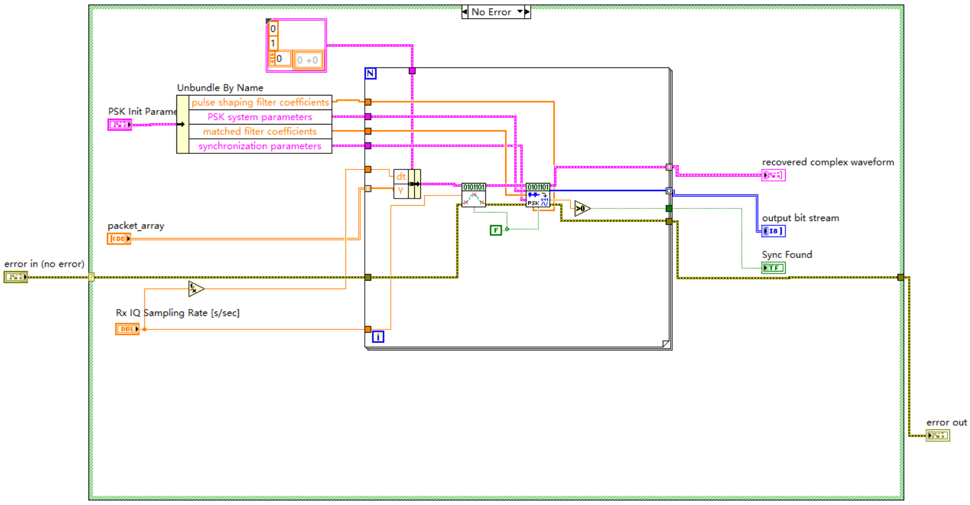




Add the USRP module to the original queue TX program, so that the packet in the queue is pulse shaped, passed through the DAC, transmitted by the antenna of the actual USRP platform, and then converted by the ADC after the antenna received by the RX program, and the received packet passes through the Match Filter and synchronously passes through the Detector, and finally the original Message text can be decoded.

As shown in the figure above, QPSK completes the decoding of the text data, and the bit error rate is 0, and the recovered constellation map also satisfies QPSK modulation





The architecture of the two subVIs is completed, and the construction of subVI in the TX program involves two modules, the pulse forming filter and the PSK modulation module, and the construction can be completed after the input parameters are given. Similarly, in the RX program, the connection of the resampling module and the PSK demodulation module can be completed.

Producer-consumer model

TX is partially equivalent to the producer model, and the enqueue function passes packets into the queue, so that data is bloomed and transmitted, and can generate a signal to stop consumers from accepting data. RX is equivalent to the consumer model, and the dequeue function extracts packets from the queue for processing in turn according to the principles of FIFO.

4. Advantages and disadvantages of the PSK modulation system

PSK (Phase Displacement Keying) modulation is a commonly used digital modulation technique that has some advantages and disadvantages in digital communication systems. Below is an analysis of the advantages and disadvantages of PSK modulation systems and their value in social applications.

Advantages:

Anti-interference: PSK modulation has good anti-interference performance. The phase change is the main information carrier of PSK modulation, while the amplitude is relatively fixed. This makes the PSK modulation system relatively insensitive to amplitude changes in noise and

interference, enabling reliable data transmission under poor channel conditions.
Efficient spectrum utilization: PSK modulation enables high spectral efficiency. By mapping multiple bits into each symbol, PSK modulation can transmit more data within a limited bandwidth. For example, QPSK modulation carries 2 bits per symbol, and 16-QAM modulation carries 4 bits per symbol, thus providing higher data transfer rates.
Simple demodulator design: The demodulator design for PSK modulation is relatively simple. Demodulators are primarily concerned with phase demodulation, which restores the original digital data based on the phase state of the received signal. Phase demodulators are designed to be relatively easy to implement and have low hardware complexity.

Shortcoming:

Phase offset sensitive: PSK modulation is sensitive to phase offset. Since phase is critical information for PSK modulation, phase shifts at the receiver can lead to demodulation errors or data loss. Therefore, PSK modulation systems require effective phase tracking and correction techniques to counteract the effects of phase offset.
Lower noise immunity: PSK modulation has lower noise immunity compared to some other modulation schemes. PSK modulation is sensitive to noise and attenuation, especially under poor channel conditions, which can lead to higher bit error rates.

Social applications

Wireless communication: PSK modulation is widely used in the field of wireless communication, such as mobile communication, satellite communication and wireless local area network. It can provide efficient spectrum utilization and better anti-interference performance, enabling wireless communication systems to achieve high-speed and reliable data transmission. This is critical to the wireless communication needs of modern society, enabling people to make voice calls, video communications, mobile Internet access, and more.
Digital radio and television: PSK modulation can be used in digital radio and television systems by converting audio, video, and data information into digital signals and transmitting them using PSK modulation. This digital transmission method has better anti-interference performance and transmission quality, can provide clearer and more stable radio and television signals, and provide users with higher-quality entertainment and information services.
Optical fiber communication: PSK modulation is also widely used in optical fiber communication systems. Optical fiber communication uses optical signals for high-speed data transmission, and PSK modulation can convert digital data into optical signals with phase changes to realize data transmission and demodulation. Optical fiber communication has the advantages of high bandwidth, low loss, and anti-interference, enabling information to be transmitted at the speed of light, and is widely used in long-distance communications, Internet backbone networks, and data centers.
Satellite Communications: PSK modulation also plays an important role in satellite communications. Satellite communication is a key means of long-distance communication and broadcasting, and PSK modulation can provide efficient spectrum utilization and better anti-interference performance, enabling satellite communication systems to achieve global data transmission and communication coverage, supporting satellite phones, remote monitoring , disaster relief and other applications.
Overall, the PSK modulation system has a wide range of applications in modern communication and transmission fields, providing people with reliable and efficient data transmission and

communication services. It plays an important social application value in promoting the development of the information society, improving social productivity, and improving people's quality of life.

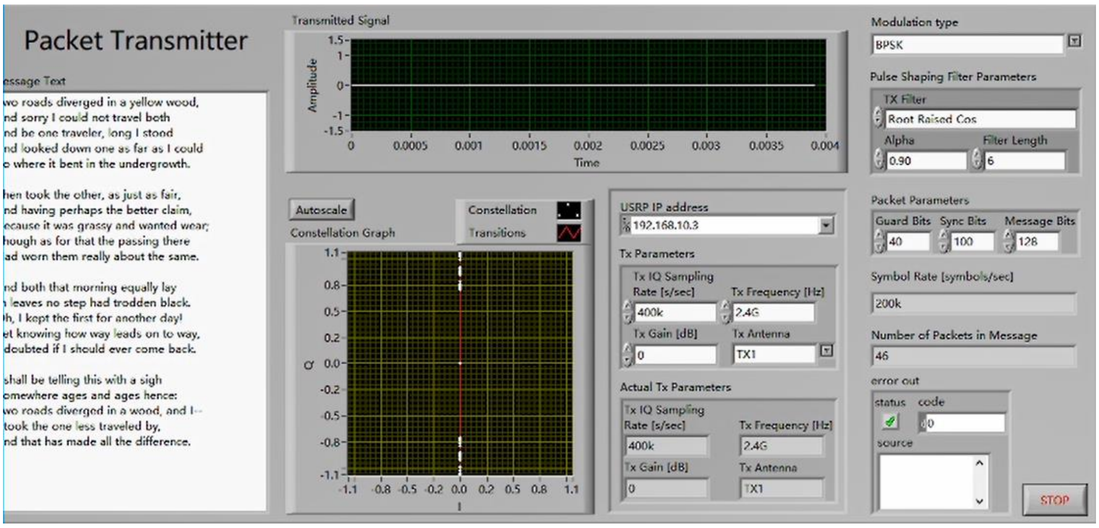
5. Extension

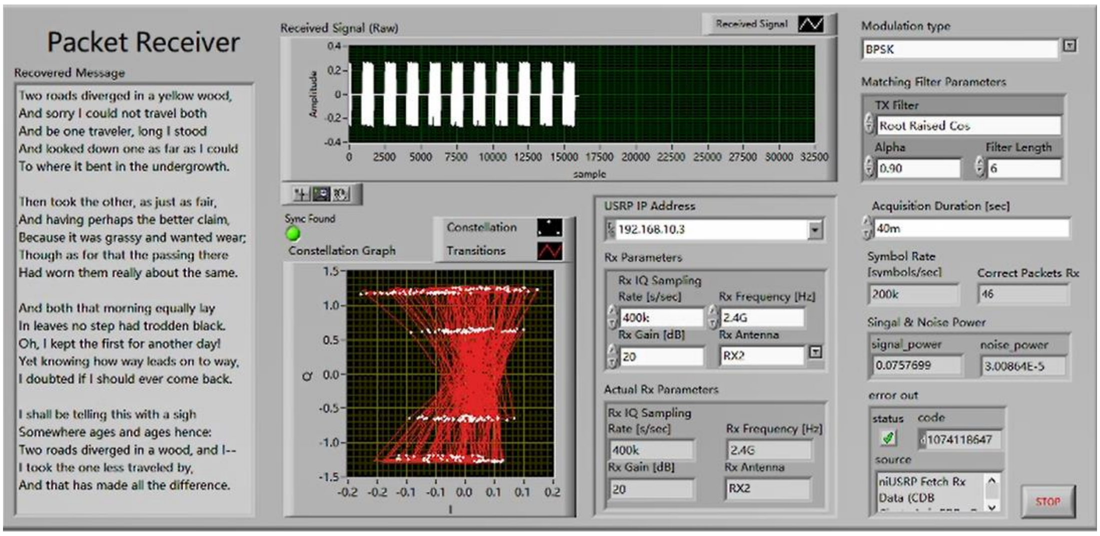
1. Realize 16-PSK, 8-PSK, QPSK, BPSK four digital modulation

To complete MPSK, we only need to change the packet structure parameters and pulse forming filter parameters in TX and RX previously built using the USRP platform

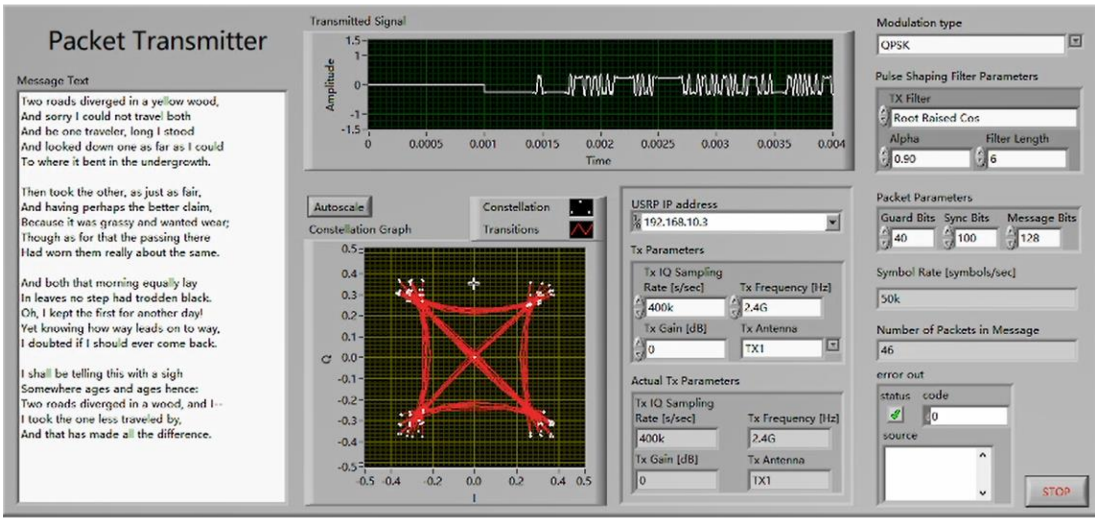
Adjust the sync sequence to 100 bits and message bits to 128 bits

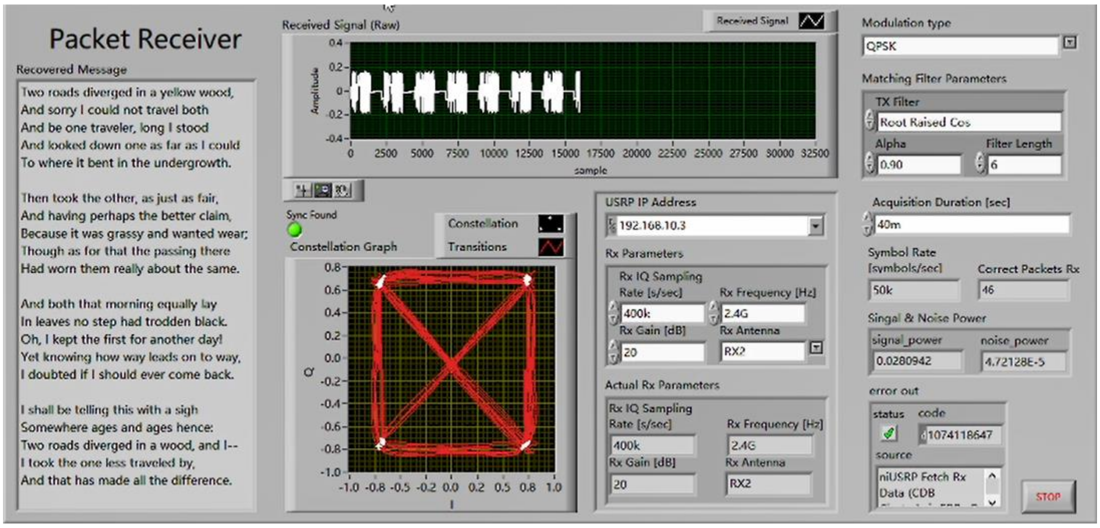
BPSK:



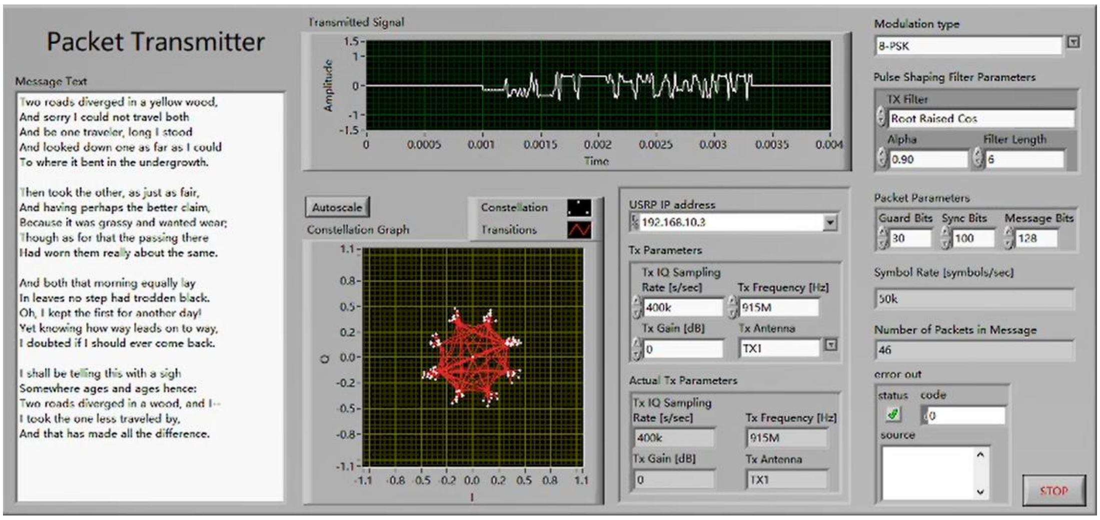


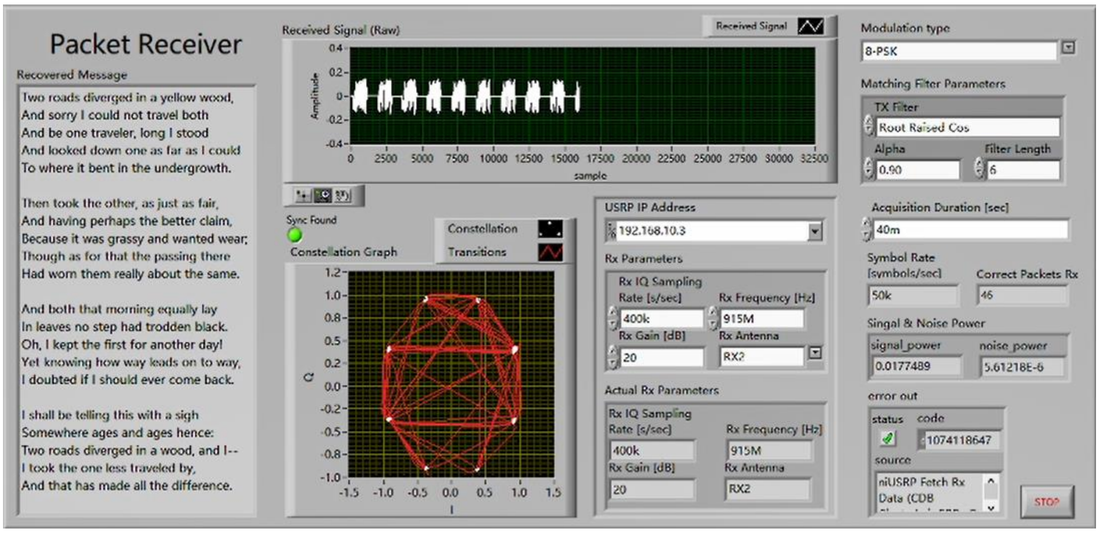
QPSK:


Packet Transmitter
Message Text
Two roads diverged in a yellow wood,
And sorry I could not travel both
And be one traveler, long I stood
And looked down one as far as I could
To where it bent in the undergrowth.
Then took the other, as just as fair,
And having perhaps the better claim,
Because it was grassy and wanted wear;
Though as for that the passing there
Had worn them really about the same.
And both that morning equally lay
In leaves no step had trodden black.
Oh, I kept the first for another day!
Yet knowing how way leads on to way,
I doubted if I should ever come back.
I shall be telling this with a sigh
Somewhere ages and ages hence:
Two roads diverged in a wood, and I--
I took the one less traveled by,
And that has made all the difference.
Transmitted Signal
Amplitude
Time
Autoscale
Constellation Graph
Constellation
Transitions
Modulation type
QPSK
Pulse Shaping Filter Parameters
TX Filter
Root Raised Cos
Alpha
0.90
Filter Length
6
USRP IP address
192.168.10.3
Tx Parameters
Tx IQ Sampling Rate [s/sec]
400k
Tx Frequency [Hz]
2.4G
Tx Gain [dB]
0
Tx Antenna
TX1
Actual Tx Parameters
Packet Parameters
Guard Bits
40
Sync Bits
100
Message Bits
128
Symbol Rate [symbols/sec]
50k
Number of Packets in Message
46
error out
status
code
0
source
STOP


Packet Receiver
Recovered Message
Two roads diverged in a yellow wood,
And sorry I could not travel both
And be one traveler, long I stood
And looked down one as far as I could
To where it bent in the undergrowth.
Then took the other, as just as fair,
And having perhaps the better claim,
Because it was grassy and wanted wear;
Though as for that the passing there
Had worn them really about the same.
And both that morning equally lay
In leaves no step had trodden black.
Oh, I kept the first for another day!
Yet knowing how way leads on to way,
I doubted if I should ever come back.
I shall be telling this with a sigh
Somewhere ages and ages hence:
Two roads diverged in a wood, and I--
I took the one less traveled by,
And that has made all the difference.
Received Signal (Raw)
Received Signal
Amplitude
sample
Sync Found
Constellation Graph
Constellation
Transitions
Modulation type
QPSK
Matching Filter Parameters
TX Filter
Root Raised Cos
Alpha
0.90
Filter Length
6
USRP IP Address
192.168.10.3
Rx Parameters
Rx IQ Sampling Rate [s/sec]
400k
Rx Frequency [Hz]
2.4G
Rx Gain [dB]
20
Rx Antenna
RX2
Actual Rx Parameters
Acquisition Duration [sec]
40m
Symbol Rate [symbols/sec]
50k
Correct Packets Rx
46
Singal & Noise Power
signal_power
0.0280942
noise_power
4.72128E-5
error out
status
code
1074118647
source
niUSRP Fetch Rx Data (CDB
STOP

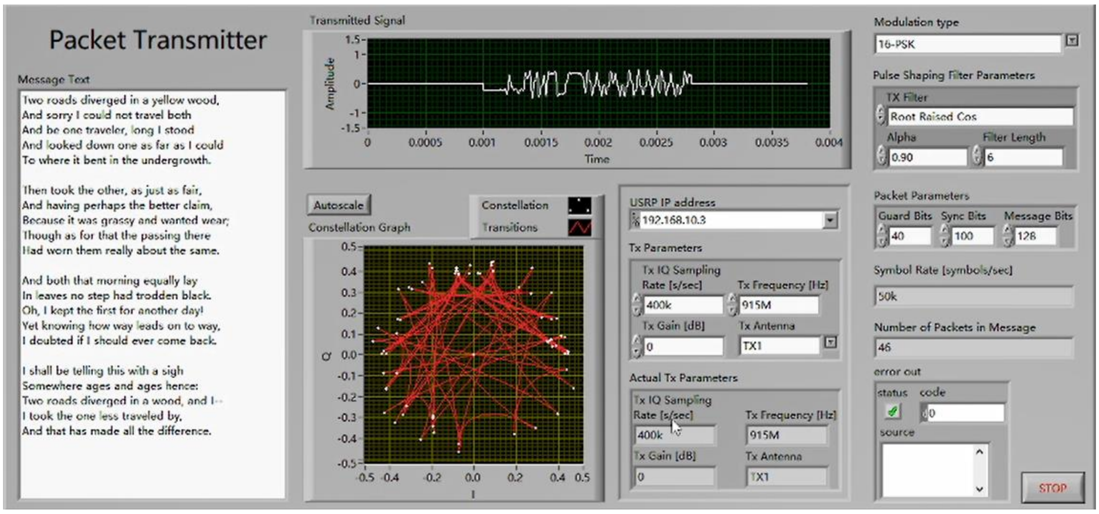
8PSK:

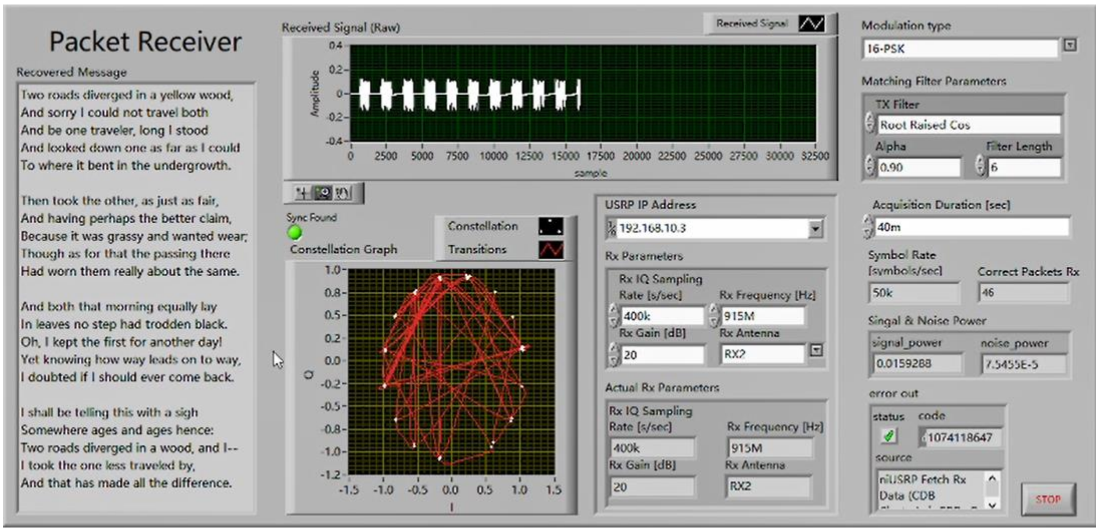

Packet Transmitter
Message Text
Two roads diverged in a yellow wood,
And sorry I could not travel both
And be one traveler, long I stood
And looked down one as far as I could
To where it bent in the undergrowth.
Then took the other, as just as fair,
And having perhaps the better claim,
Because it was grassy and wanted wear;
Though as for that the passing there
Had worn them really about the same.
And both that morning equally lay
In leaves no step had trodden black.
Oh, I kept the first for another day!
Yet knowing how way leads on to way,
I doubted if I should ever come back.
I shall be telling this with a sigh
Somewhere ages and ages hence:
Two roads diverged in a wood, and I--
I took the one less traveled by,
And that has made all the difference.
Transmitted Signal
Amplitude
Time
Autoscale
Constellation Graph
Constellation
Transitions
Modulation type
8-PSK
Pulse Shaping Filter Parameters
TX Filter
Root Raised Cos
Alpha
0.90
Filter Length
6
USRP IP address
192.168.10.3
Tx Parameters
Tx IQ Sampling Rate [s/sec]
400k
Tx Frequency [Hz]
915M
Tx Gain [dB]
0
Tx Antenna
TX1
Actual Tx Parameters
Packet Parameters
Guard Bits
30
Sync Bits
100
Message Bits
128
Symbol Rate [symbols/sec]
50k
Number of Packets in Message
46
error out
status
code
0
source
STOP



Since the guard bits in the packet structure occupy a certain space, 8PSK modulation is analyzed every three bits, so the guard bits need to be a multiple of 3, and here it is adjusted to 30 bits

16PSK:



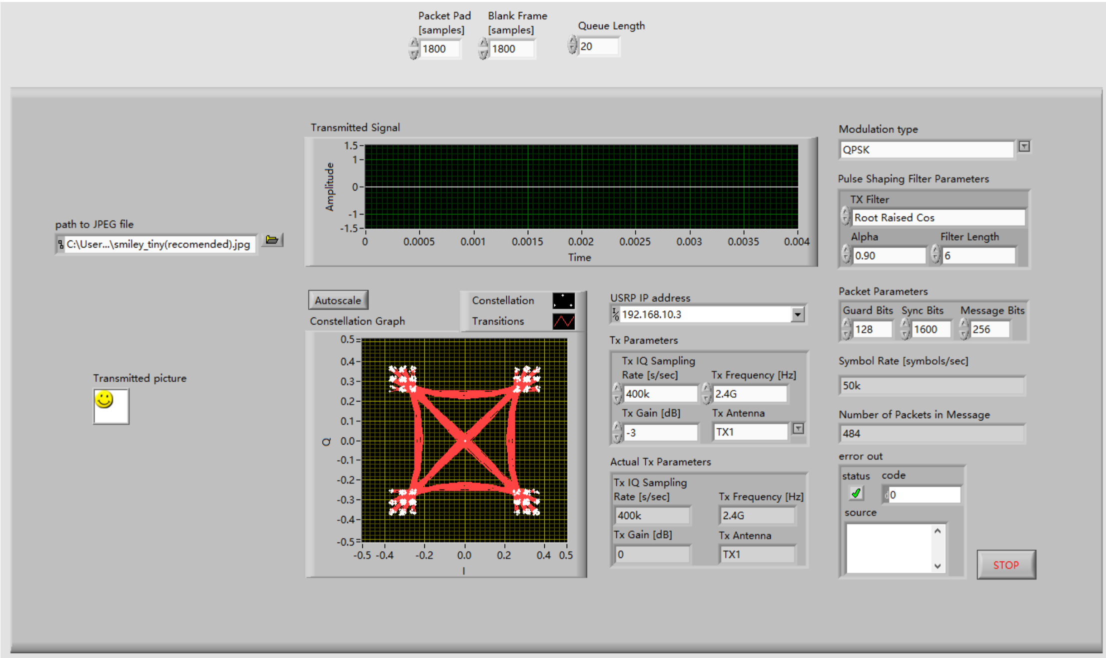


From the above experimental result figure, it can be seen that MPSK modulation achieves the effect of fully restoring the original transmitted text based on the USRP platform to achieve both transmission and reception, and the bit error rate is 0, and the coordinates of the constellation plot points conform to the corresponding modulation.

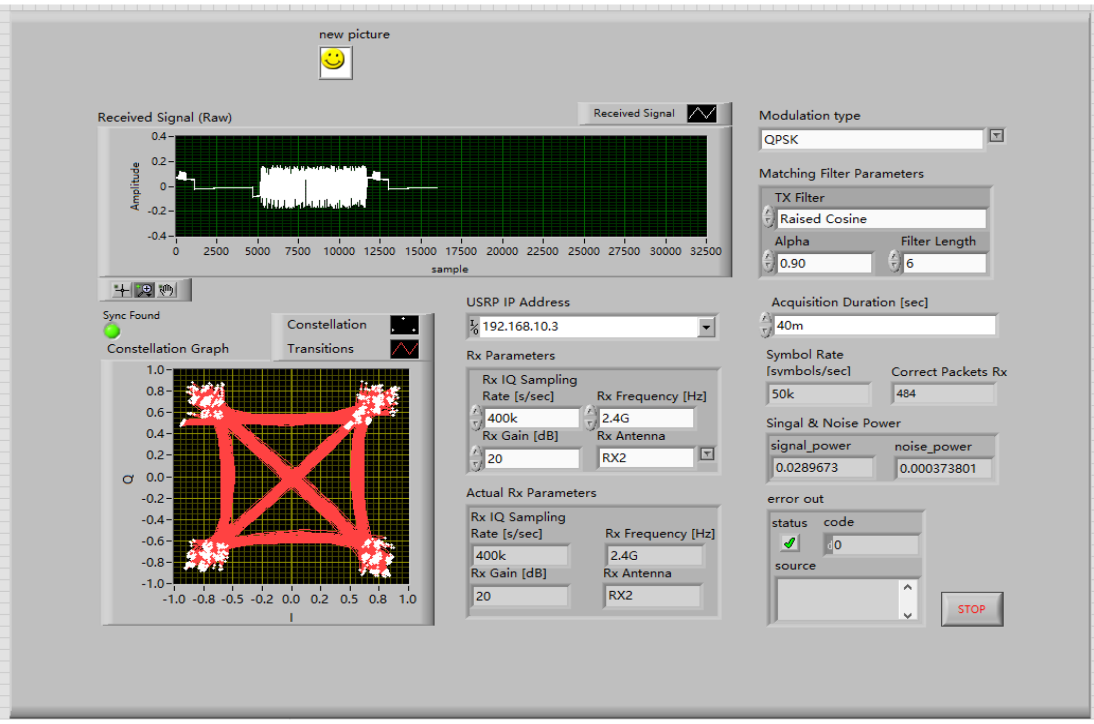
2. The image transmission by MPSK

We have also implemented image transmission based on USRP, and the resulting implementation is as follows (taking the QPSK as example, other PSK types is similar in essence):

TX:



RX:



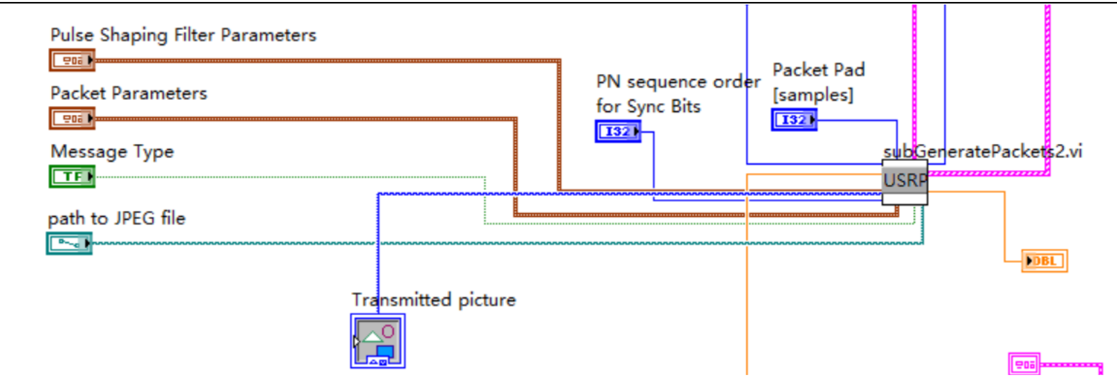
From the above image, it can be observed that the modulation effect of the transmitted data at the transmission end is good, forming a relatively standard constellation diagram. The image to be transmitted is displayed on the left side. In the time-domain signal plot, a straight line is displayed. This is because we set a large number of zero bits in the preceding guard band, which prevents the signal from being displayed. All the parameters are displayed on the rear panel, and these parameters have a significant impact on the performance of the system transmission.

Since image transmission involves a much larger amount of data compared to text transmission, to prevent collisions and interference between such a large amount of data, we have set relatively large values for the guard band and padding at the rear end. Additionally, the synchronization sequence also has a significant impact. We adjusted the synchronization sequence appropriately to ensure the transmission performance of the system and continuously experimented to select the most suitable value.

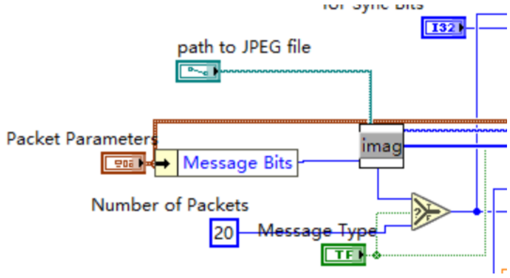
We chose 256 for the message signal because it corresponds to exactly 484 packets. The image size is 22x22, which means there are 484 pixels. Having 484 packets ensures that all the pixels can be recovered, although the recovery process may be slower.

Below, we will introduce the changes we made to the program and its implementation principles:

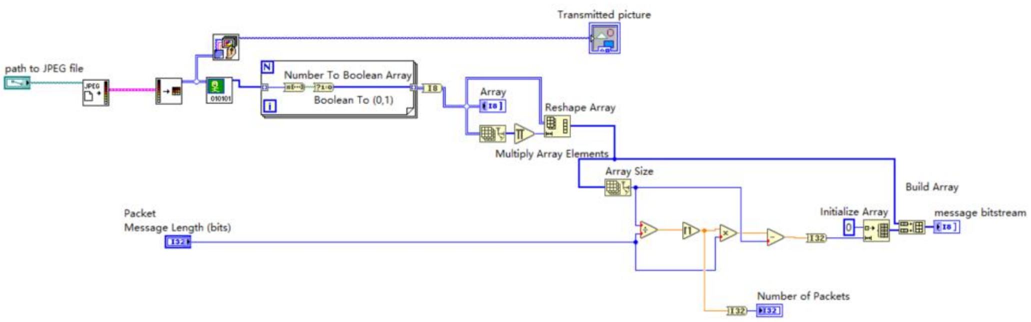
In TX, we adjust the subGeneratePackets.vi whose position is present as below:



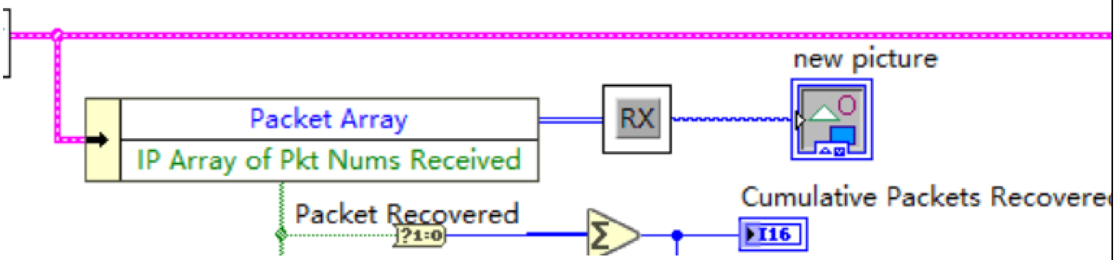
The core function of this module is to take the selected image, convert it into a series of bits, and then divide these bits into a series of packets for transmission. Inside this module, we primarily made modifications to the "image" module, which is positioned as follows:



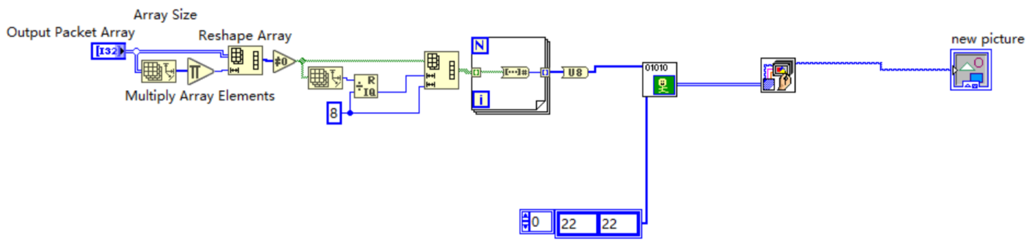
The purpose of this module is to select a file path for an image, convert the image into a series of bits, and divide the bits into multiple packets for transmission. The program block diagram for this module is as follows:



At the RX end, we also made modifications and adjustments to some modules in the original program. The main changes were made to the RX module, which is positioned as follows:



The purpose of this module is to reassemble the bit streams from each packet, which have been demodulated, based on the packet numbers. Then, the bit stream is converted back into image data, and the recovered image is displayed. It is important to note that the parameters here should be set to 22 and 22, as the image size is 22x22. The program block diagram for this module is as follows:



Experience

1. Problems and experience

(1) During the design of our first extension, we encountered a decoding failure issue in 16PSK modulation. After investigating the program, we found no issues with the code itself. Since different types of PSK modulation only involve a small portion of the program where the modulation type is selected, we suspected that the problem might lie in the parameter settings.

After analysis and careful consideration, we realized that the guard band in the packet header has a significant impact. Since 16PSK processes data in units of 4 bits, the preceding zero pad in the guard band must be a multiple of 4. This ensures that the zero pad can be processed together without being mixed with the actual data, which would otherwise disrupt the decoding process. This was the reason for the decoding failure.

Similarly, for 8PSK, the zero pad should be set as a multiple of 3, and for QPSK, it should be set as a multiple of 2. We conducted experiments to validate our analysis and conclusions, and successfully resolved the issue accordingly.

(2) In our experiments, the parameter settings played a crucial role in the system's performance. We invested significant effort in parameter tuning, which helped us gain a better understanding of the various components in the packet header. Through practical validation, we obtained a deeper and more intuitive understanding.

Both the guard band and the padding in the packet header serve to protect the data and prevent data collisions. Increasing their size to a certain extent enhances system stability. By increasing their size, we observed a reduction in noise in the constellation diagram, resulting in a more visually pleasing and cleaner constellation plot.

The synchronization sequence also plays a crucial role. In theory, a larger synchronization sequence is better as it improves the accuracy of data recovery. However, increasing its size also introduces a series of challenges, requiring continuous debugging to select an appropriate value.

(3) During the extension of image transmission, we encountered several challenges. Initially, we attempted to directly convert the image into bits for transmission, based on the method used in lab 7. However, we realized that the data needed to be divided into multiple packets for successful transmission. Therefore, we added operations to split the data into different packets in the program, allowing for proper display of the image.

Parameter settings played a crucial role in this process as well. With the increased amount of data, it was necessary to adjust the number of zero pads for data protection to ensure stable and effective data transmission. The value of "message bits" was also significant in this context. Since our program converts and transmits data on a per-pixel basis, and the image contains a total of 484 pixels, dividing it into 484 packets ensures the recovery of each pixel. Hence, "message bits" needs to be set to 256. Setting it to other values would result in garbled output. However, it should be noted that recovering individual pixels can be time-consuming. An alternative approach is to convert all the bits into a string of characters and transmit them using text-based transmission. This method requires minimal modifications to the program and allows for quick image recovery.

2. Contribution

We two finish the whole LabVIEW program together. Zhang Haodong has completed the writing of the introduction section, providing explanations on the theory and modules, and has also completed Extension 2. Song Yihan has completed the programming part and Extension 1.

Contribution ratio :50%,50%

Score	97