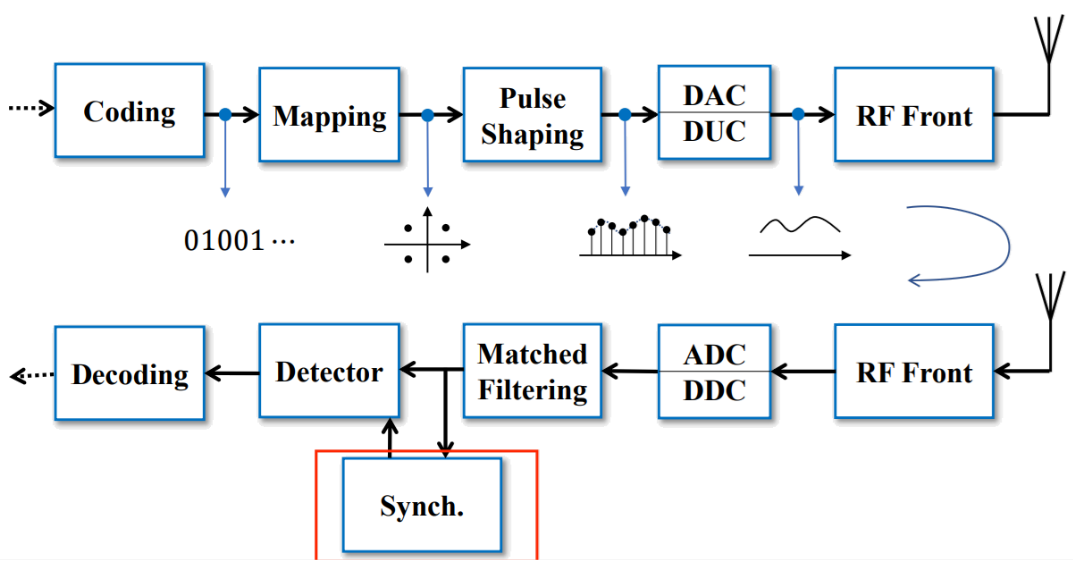
Lab 3：Symbol Synchronization

Author	Name：宋宜航　　　　　张皓东 Student ID:12112717　　　12113010

Introduction

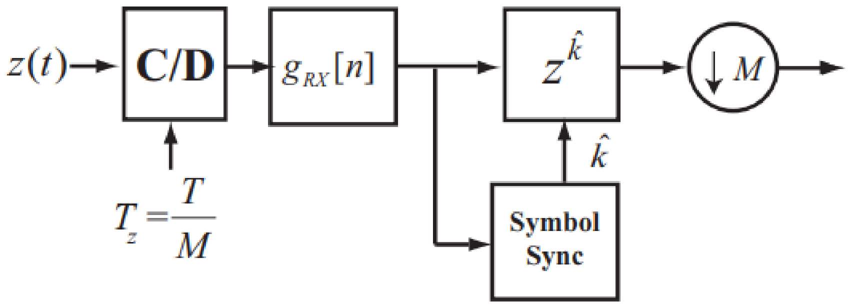


The wireless communication channel is not well modeled by simple additive white Gaussian noise. A more realistic channel model also includes attenuation, phase shifts, and propagation delays. Perhaps the simplest channel model is known as the frequency flat channel. The frequency flat channel creates the received signal

$$z(t) \quad = \quad \alpha e^{j\phi} x(t - \tau_d) + v(t),$$

where α is an attenuation, φ is a phase shift, and τd is the delay.

Two algorithms will be implemented for symbol synchronization in this lab: the maximum energy method and the Early Late gate algorithm. The maximum energy method attempts to find the sample point that maximizes the average received energy. The early–late gate algorithm implements a discrete-time version of a continuous-time optimization to maximize a certain cost function. For the best performance, both algorithms require that there is a lot of oversampling (i.e., M is large).



The Maximum Output Energy Solution

Let y(t) be the matched filtered version of z(t) in continuous-time. Given an advance τ , notice that the output energy function defined as

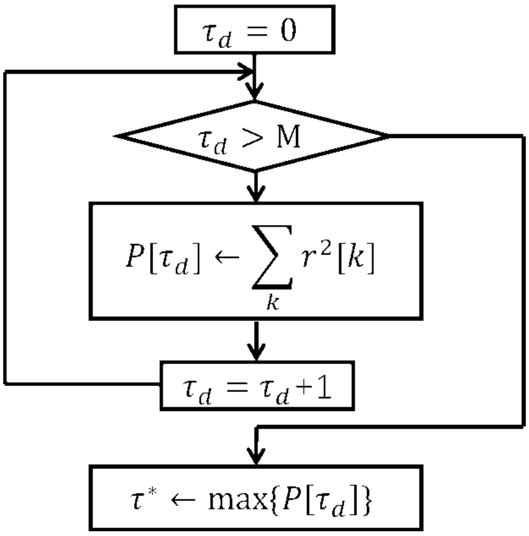
$$\begin{aligned} J(\tau) = \mathbb{E}\,|y\,(nT+\tau)|^2 &= E_x \sum_m |g\,(mT+\tau-\tau_d)\,|^2 + \sigma_v^2 \\ &\leq E_x|g(0)|^2 + \sigma_v^2. \end{aligned}$$

The maximum of the function J (τ) occurs when τ −τd is an integer multiple of the symbol rate. This value of τ is known as the maximum output energy solution. The resulting value of τ by that maximizes J (τ) is known as the maximum output energy solution. In this lab we implement two algorithms for finding a maximum output energy solution in discrete-time.

$$J_{approx}[k] = \frac{1}{p}\Sigma_0^{p-1}|r(pMT+k)|^2$$

Other criteria for performing symbol synchronization are possible, for example maximum likelihood solutions. The approach described here is robust to additive noise, certain kinds of fading channels, and small carrier frequency offsets.

Algorithmic process diagram



$$\hat{\tau}_d \approx arg\mathbf{max} \sum_i r^2(\tau_d + kT)$$

The input to the receiver has the form

$$z(t) = \alpha e^{j\phi}\sqrt{E_x}\sum_m s[m]g_{\mathrm{tx}}(t - mT - \tau_d) + v(t). \qquad (2)$$

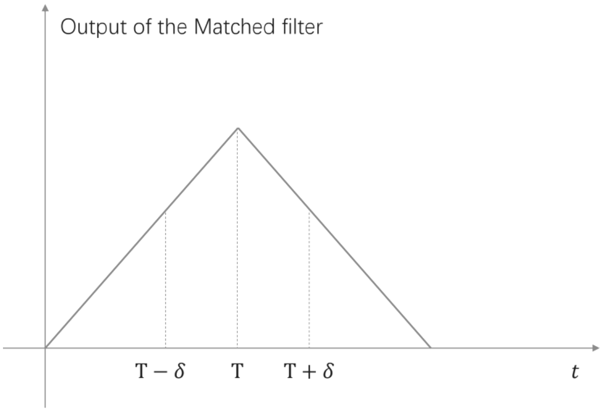
Now let g(t) be the convolution of gtx(t) and grx(t). With the model in Eq. (2) after matched filtering and sampling, the received signal is

$$y[n] = \sqrt{E_x}\alpha e^{j\phi}\sum_m s[m]g((n - m)T - \tau_d) + v[m].$$

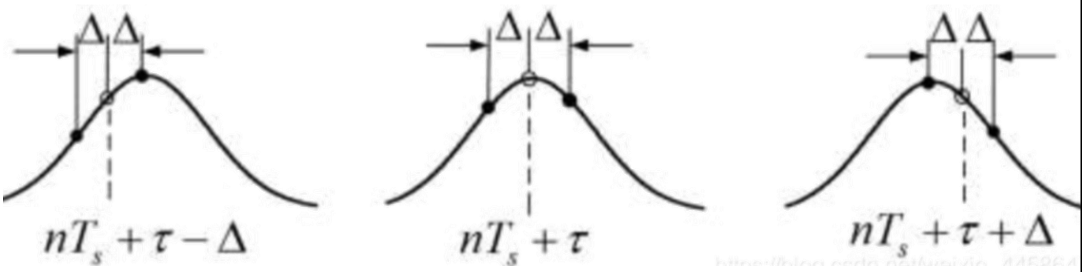
Several different impairments are possible depending on the value of the τd . Unless otherwise specified, Nyquist pulse shapes are assumed. First consider the case where τd is a fraction of a symbol period thus 0 < τd < T. This can model the effect of sample timing error, or not sampling at the right point. Under this assumption, inter symbol interference is created when the Nyquist pulse shape is not sampled exactly at nT.

$$y[n] = \underbrace{\sqrt{E_x}\alpha e^{j\phi}s[n]g(\tau_d)}_{\text{desired}} + \underbrace{\sqrt{E_x}\alpha e^{j\phi}\sum_{m\neq n} s[m]g((n - m)T - \tau_d)}_{\text{ISI}} + \underbrace{v[n]}_{\text{noise}}.$$

Early-Late Gate Algorithm



Take the baseband value of the received signal at a fixed time, and take the signal value of the adjacent time δ, make the difference and take the average, when δ is small, it can be regarded as taking the value of the derivative on the curve of the received signal, from which you can judge the trend of the received signal and introduce whether it is the extreme value point, thus increasing the SNR



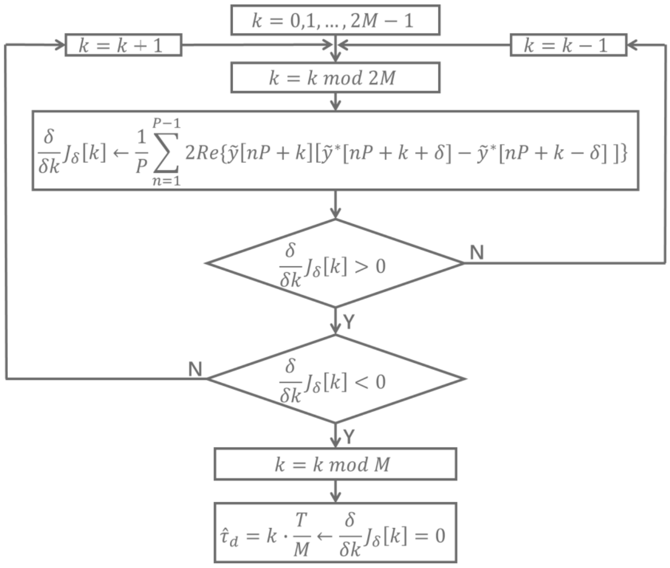
We use derivation to determine whether the peak of the target signal has been found. If the sample point is to the left of the target peak, the result should be positive, if the sample point is to the right of the target peak, the result is negative, and only if the sample point is close to the peak, the result should be 0. From this, we can get the right sampling time.

$$\frac{d}{d\tau}J_\delta(\tau) \approx \frac{1}{P}\sum_{n=1}^{P-1} 2Re\{y(nT_s+\tau)(y^*(nT+\tau+\delta)-y^*(nT+\tau-\delta))\}$$

$$\frac{d}{d\tau}J_\delta(\tau) = 0$$

The above equation gives a more accurate description and the detailed programming diagram will be shown in the analysis section.

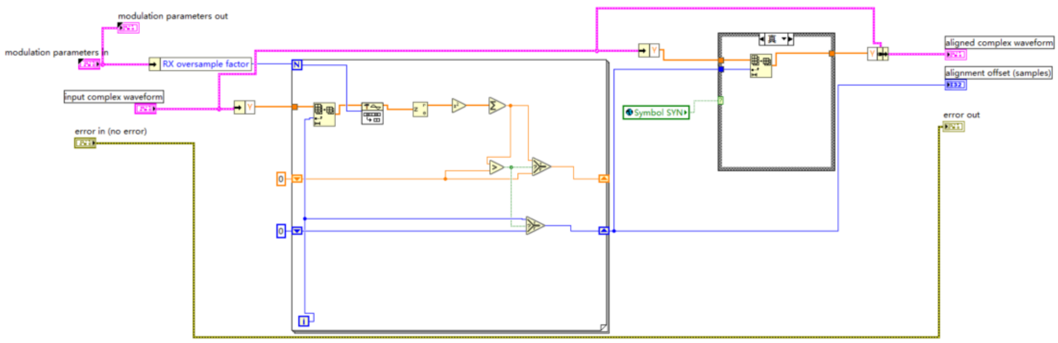
Algorithmic process diagram

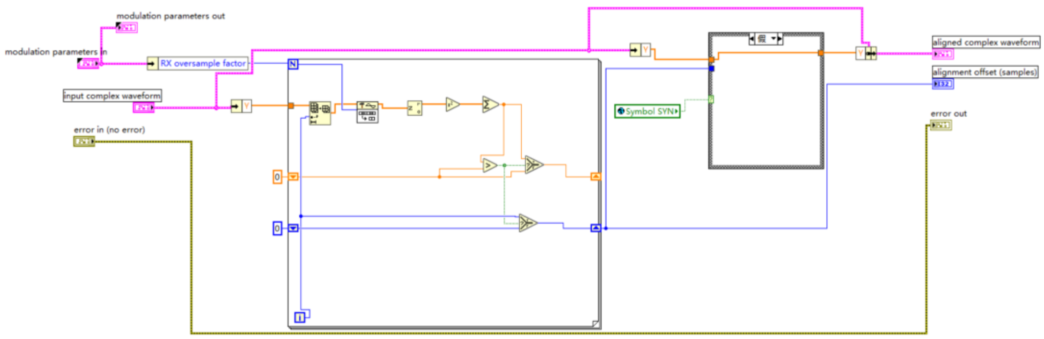


Lab results & Analysis:

Maximum Energy Algorithm

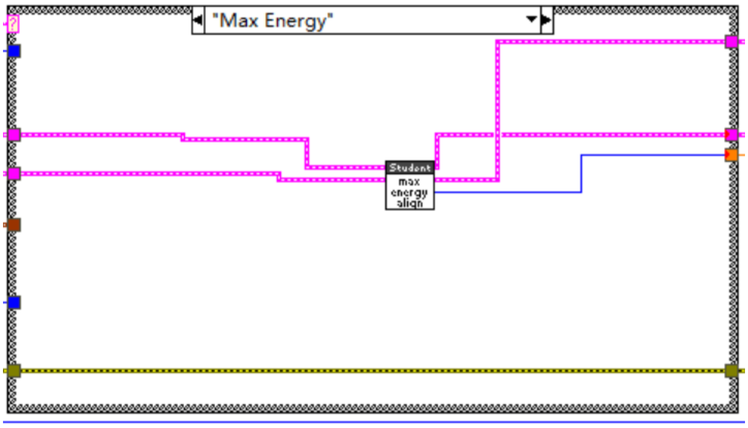
The program diagram for symbol synchronization using the maximum energy method is as follows:



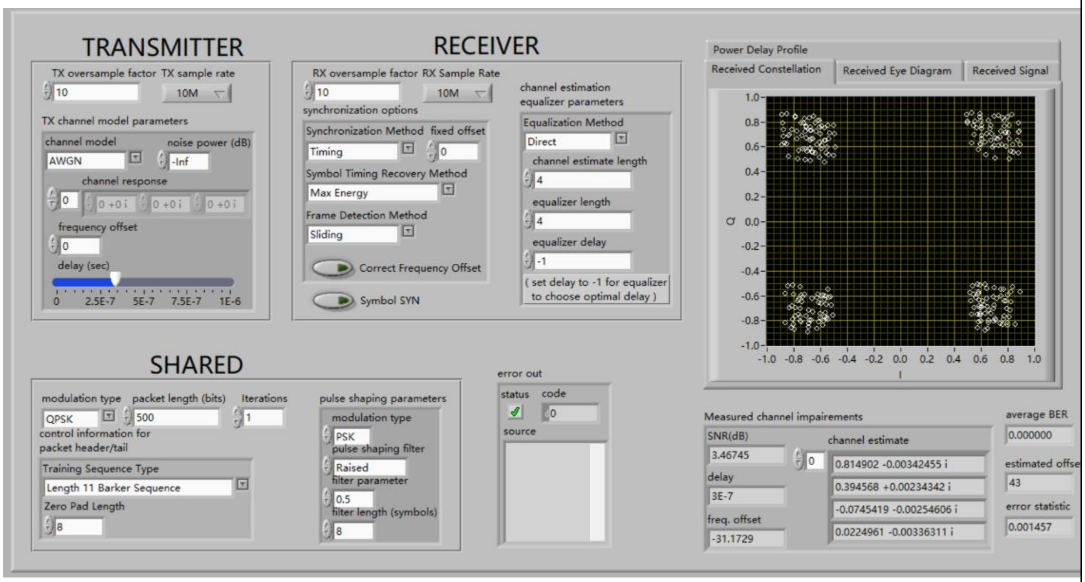


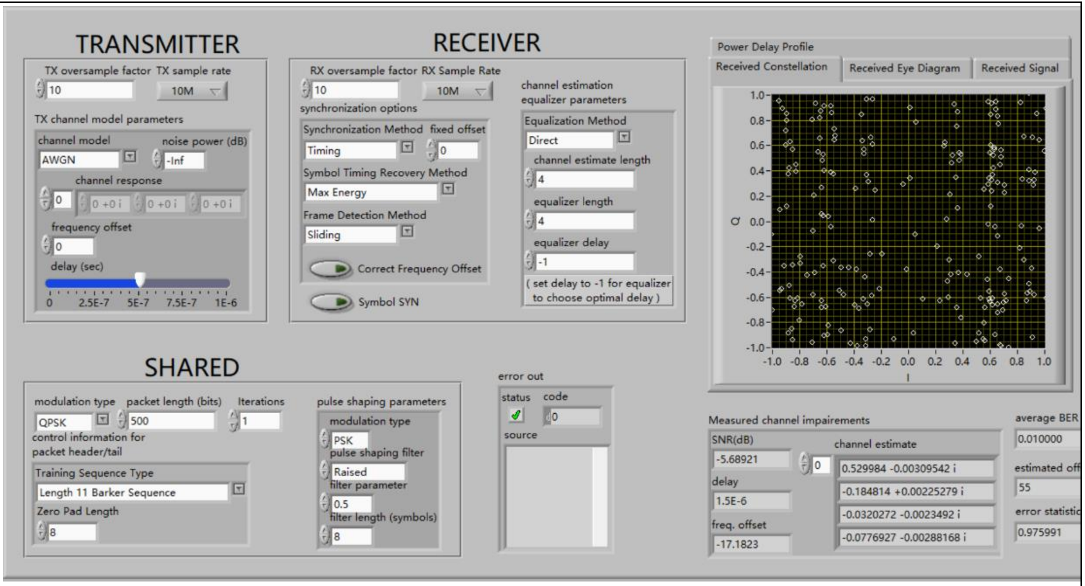
Its basic approach aligns with the fundamental principles introduced earlier. The program logic is as follows: first, convert the input waveform into an array, select the subarray after a specified index, then perform down-sampling and calculate the total energy, which involves squaring the amplitudes and summing them. Simultaneously, within a loop and using conditional statements, update the maximum value and record its index. If symbol synchronization is activated, select the subarray after the maximum index; otherwise, output the original waveform array.

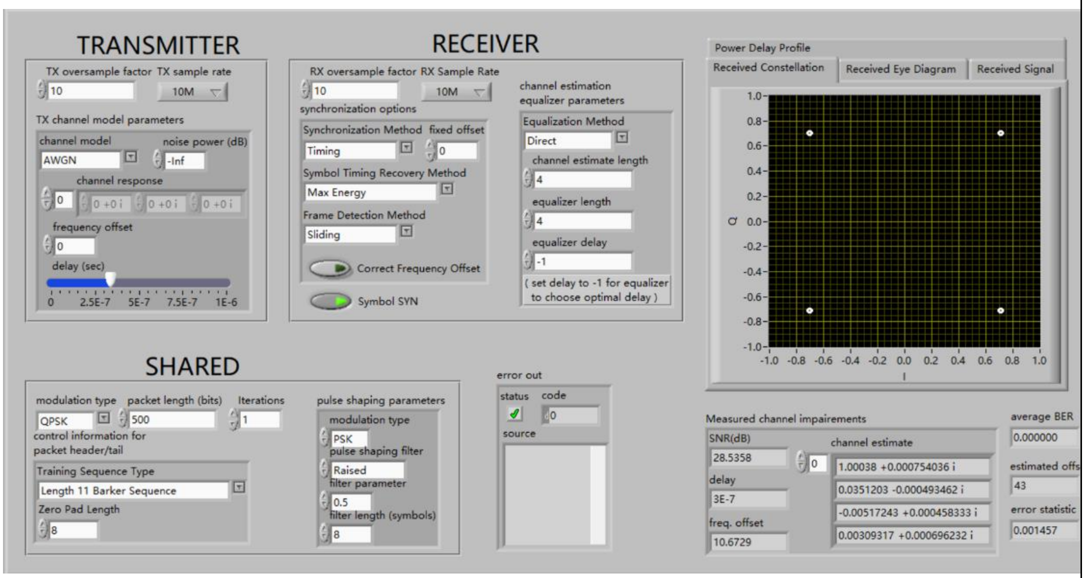
Next, we replace the module we have written into the program and verify that the written program has no errors:



Then, we can observe the simulation results. By dragging the slider to select different delay values, we can observe that as the delay increases, the points on the constellation diagram will periodically diverge. This indicates that the delay indeed affects the performance of the transceiver. As shown in the figure below, the delay at this point has a significant impact on the reception performance. However, when we enable Symbol Sync, we observe that the constellation points return to normal. This implies that our algorithm and program for symbol synchronization using the maximum energy method are correct.







Symbol synchronization accuracy with up-sampling factor

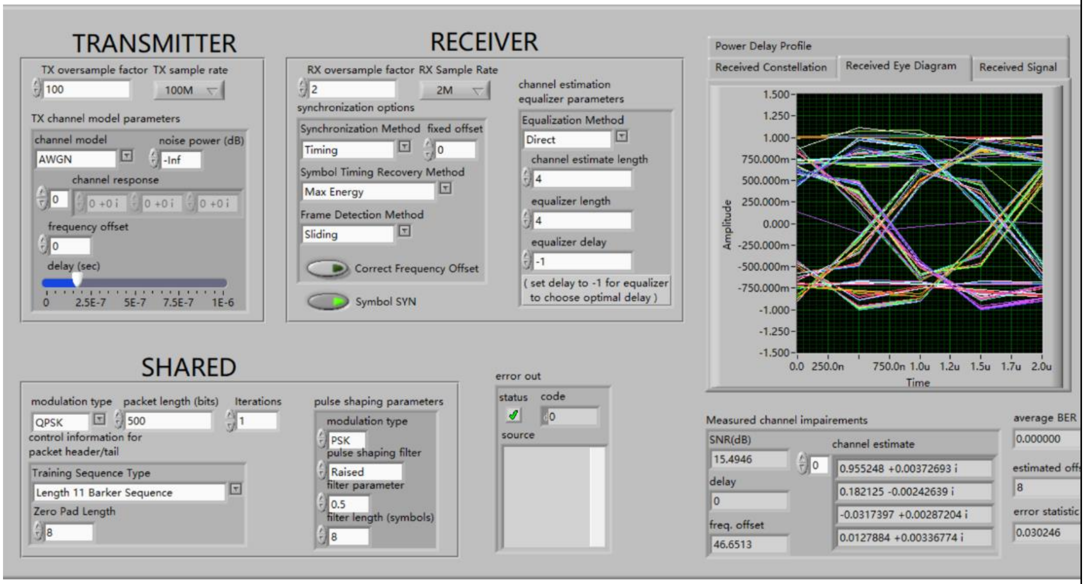
The up-sampling factor has a certain impact on the accuracy of symbol synchronization. We will explore how the up-sampling factor is related to symbol synchronization and plot a curve to represent this relationship. Other parameters are set as follows:

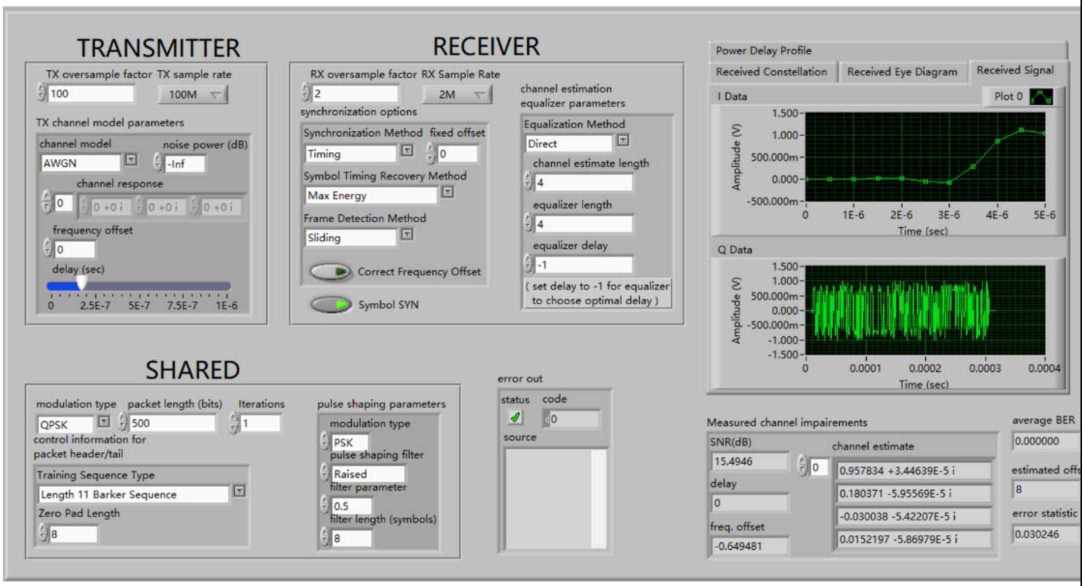
参数名	参数值
信道延迟τ_d	$0.17\mu s$
符号速率f_s	$1MHz$
发射端过采样因子M_{tx}	100
接收端过采样因子$M(M_{rx})$	2,4,10,20,50
传输信道	AWGN
信道信噪比(dB)	$-\infty$
符号定时同步方法	Max Energy

Because the TX oversample factor is 100 and the symbol rate is 1MHZ, so the TX sample rate is equal to 100*1MHZ = 100MHZ. Then we can change the RX sample rate and observe the difference of the simulation results. Similarly, the RX oversample factor*RX sample rate is also equal to 1MHZ.

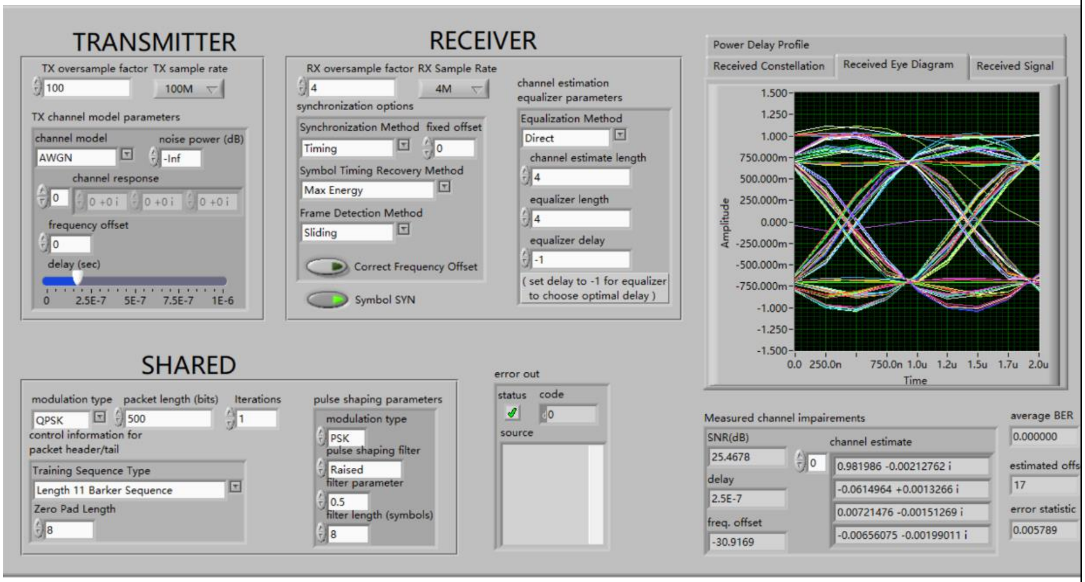
The simulation results are that:

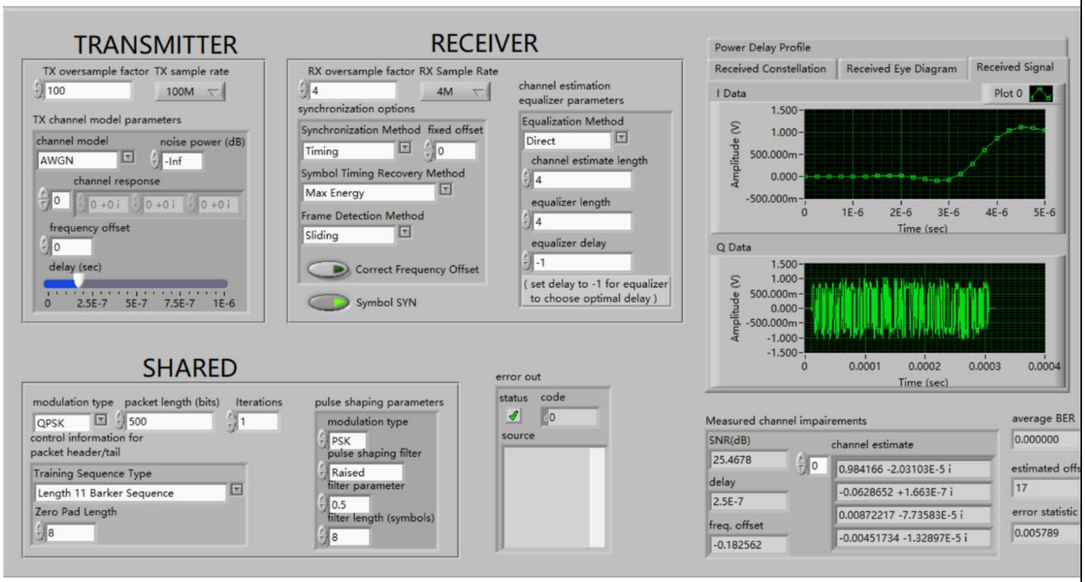
When RX sample rate = 2, and RX sample rate = 2MHZ:



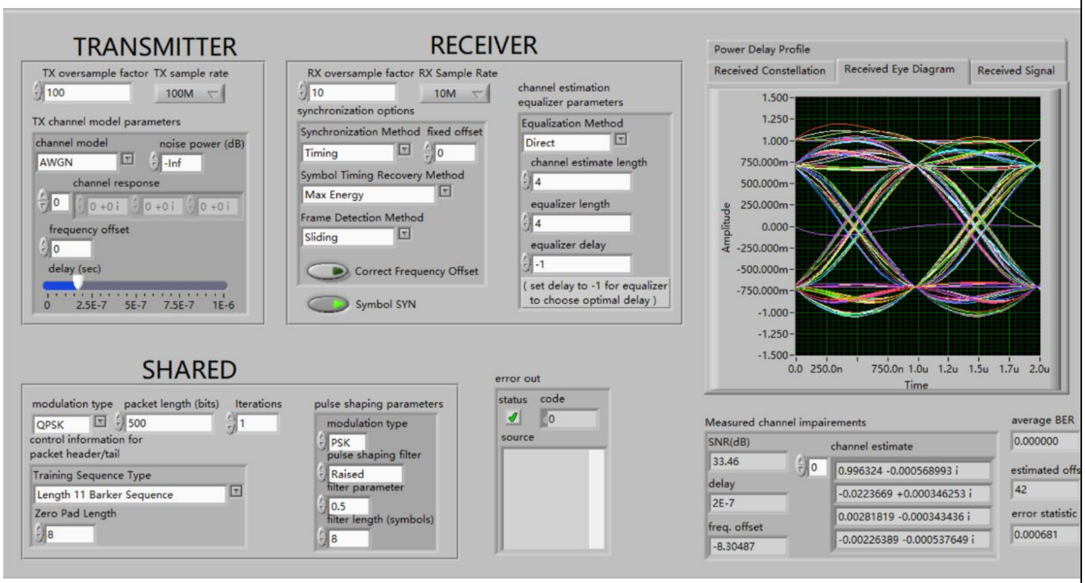


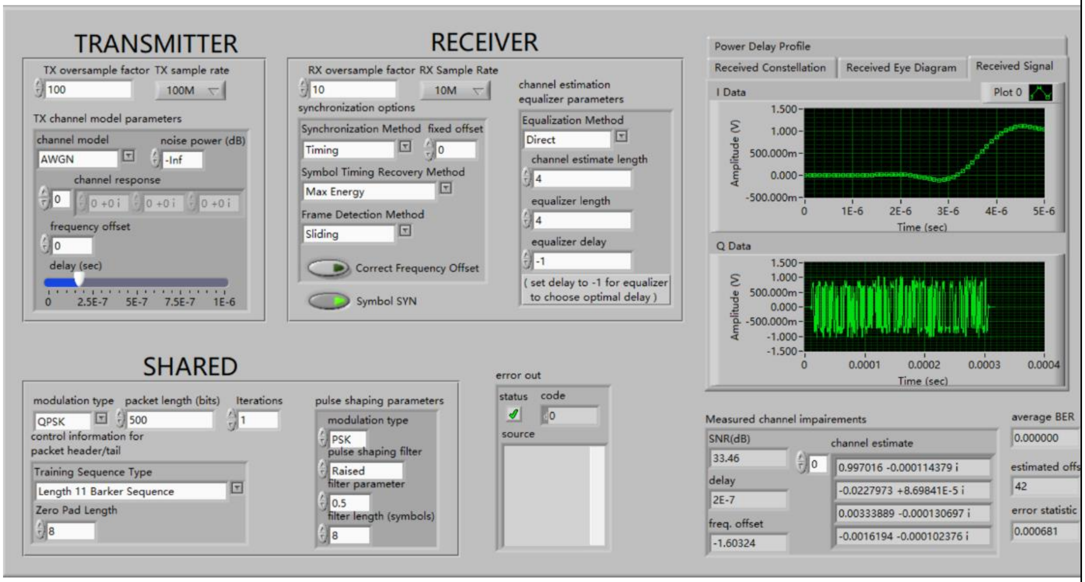
When RX sample rate = 4, and RX sample rate = 4MHZ:



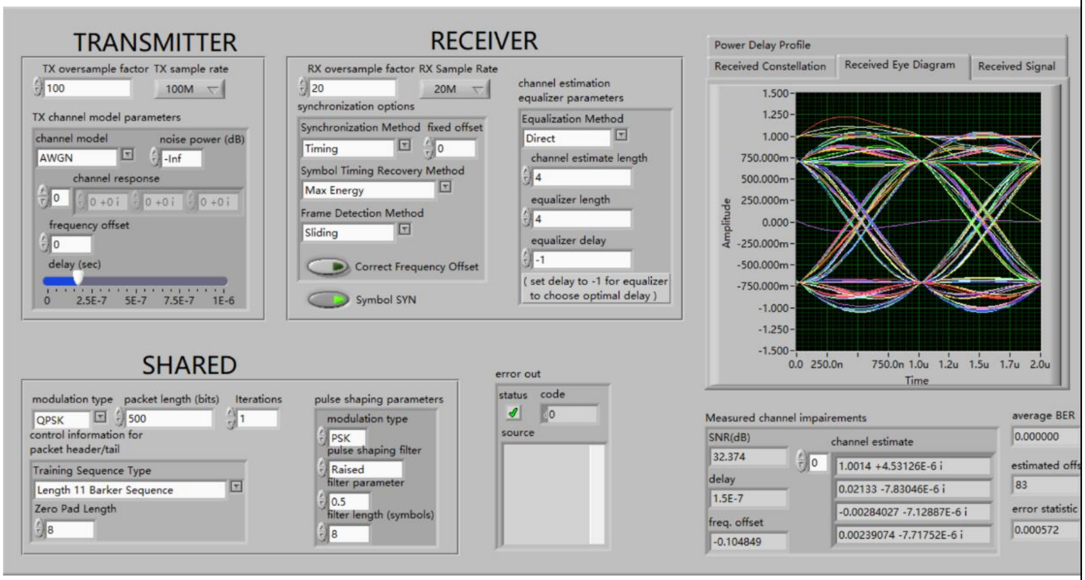


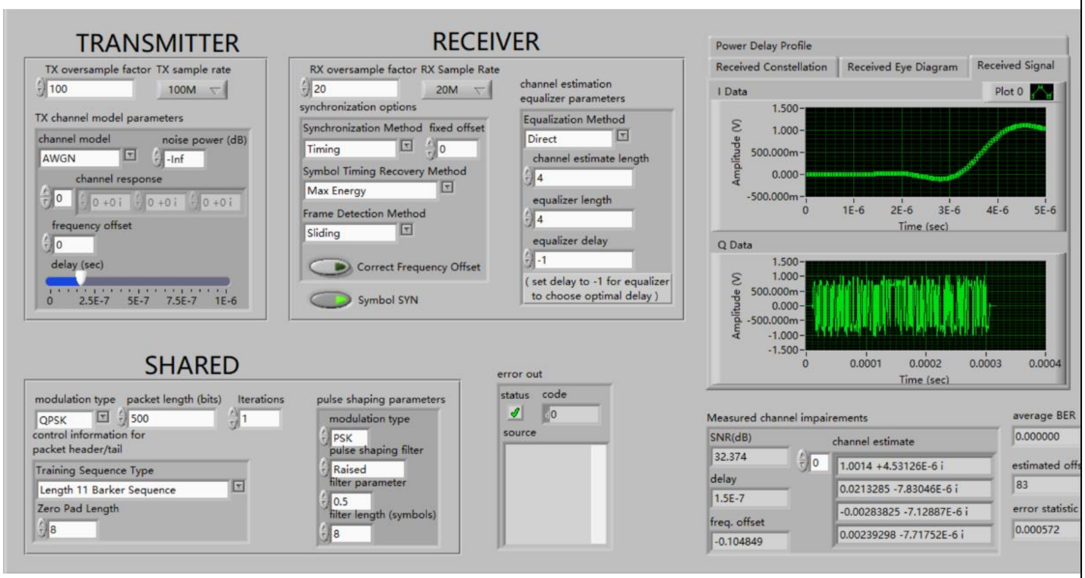
When RX sample rate = 10, and RX sample rate = 10MHZ:



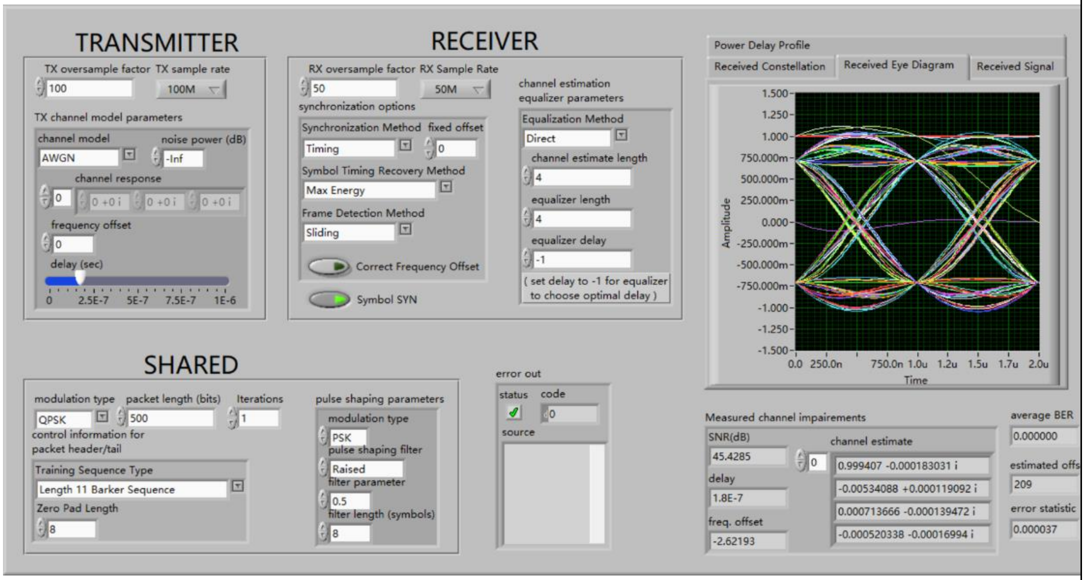


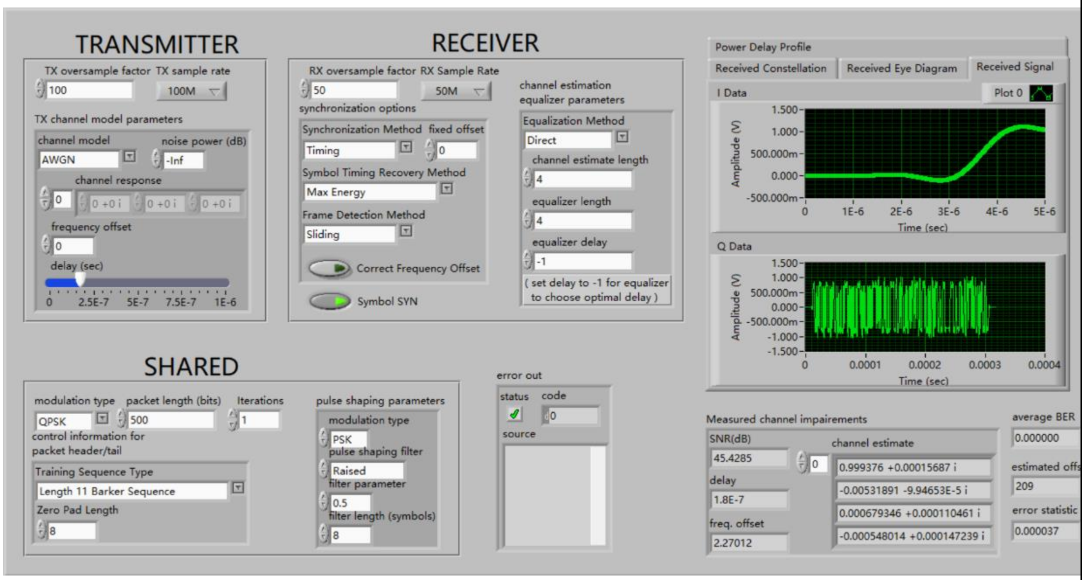
When RX sample rate = 20, and RX sample rate = 20MHZ:





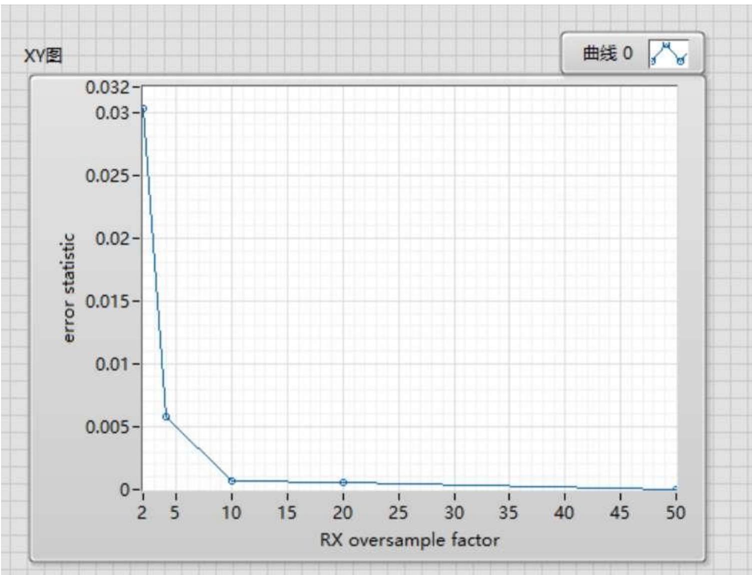
When RX sample rate = 50, and RX sample rate = 50MHZ:



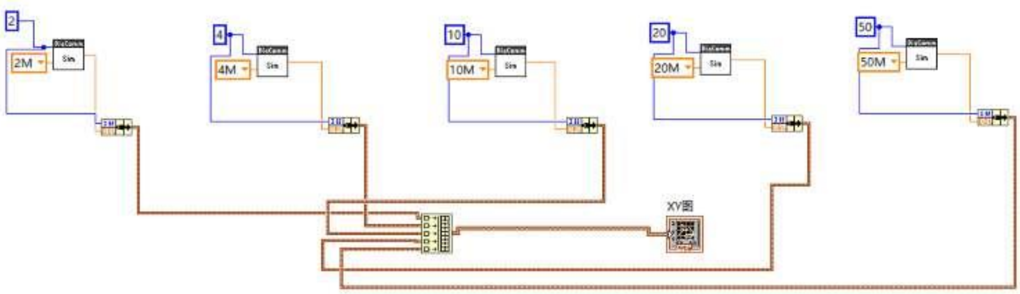


From the simulation results above, it can be observed that continuously increasing the sampling factor of RX results in more points within each cycle, as the number of points becomes denser in the received signal plot. Increasing the sampling factor also affects the transceiver performance. It can be noticed that with the increase in the sampling factor, the error probability tends to decrease, and the eye diagram gradually becomes smoother, closer to the ideal eye diagram. This indicates an improvement in performance.

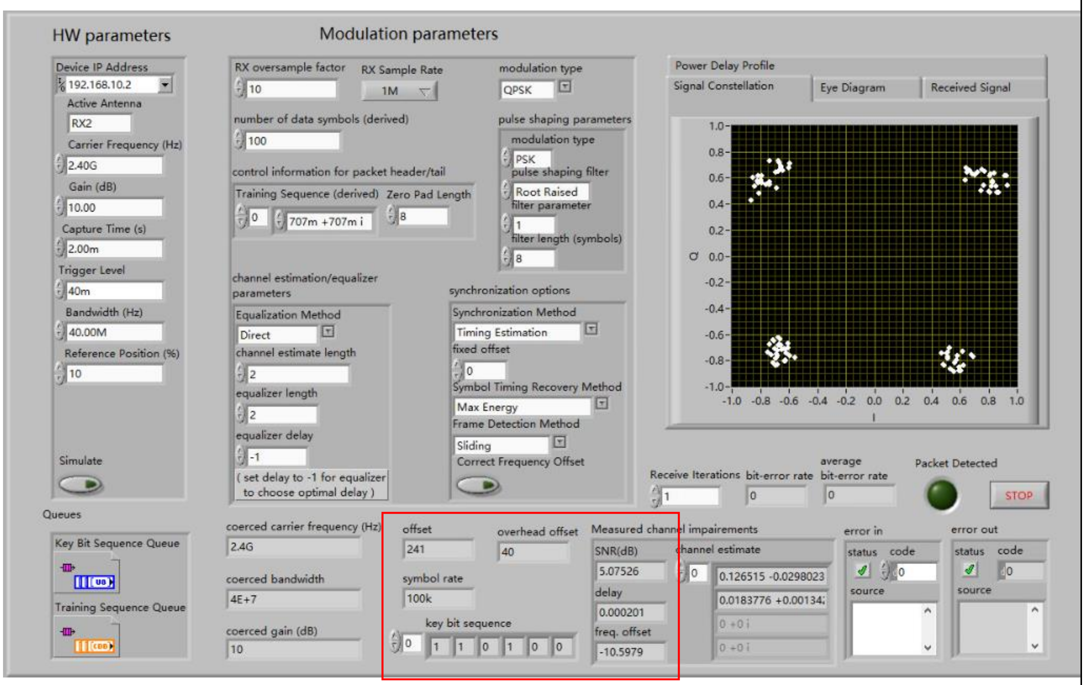
From the graph, it can be observed that as the RX oversample factor increases, the error probability rapidly decreases. Therefore, increasing the RX oversample factor can effectively reduce the error probability and enhance communication performance.



The implementation process in programming is also straightforward. It involves encapsulating the simulation VI as a subVI, with inputs being the RX sample rate and RX oversample factor. By inputting different values, the entire curve can be plotted.



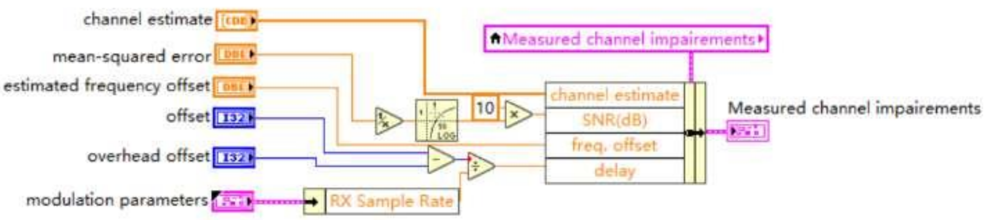
USRP Implementation



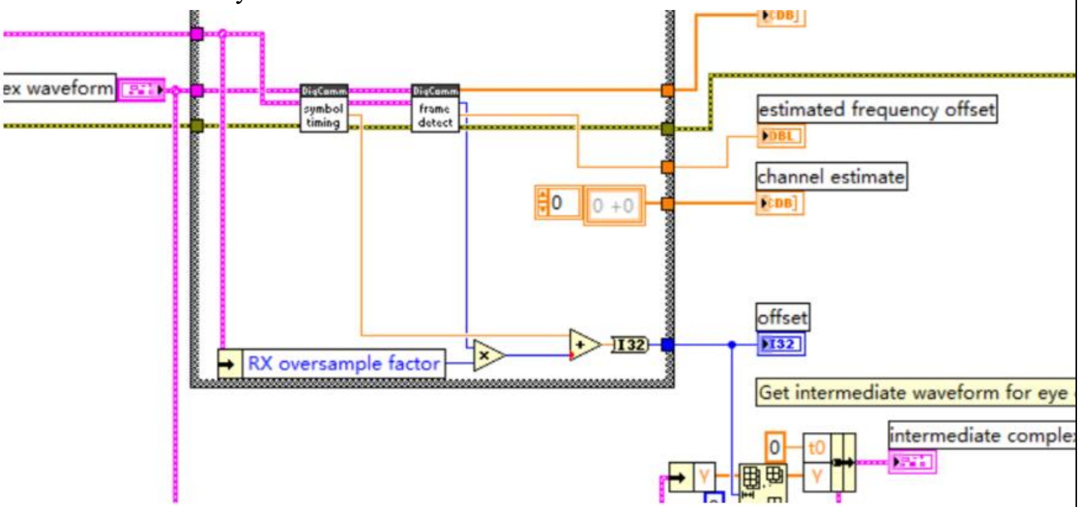
In USRP, we can implement the above communication transceiver process. We pay particular attention to three parameters: offset, overhead offset, and delay. Based on the configured parameters, we can observe a pattern among these three:

$$delay = \frac{offset - overhead_{offset}}{RX_{samplerate}}$$

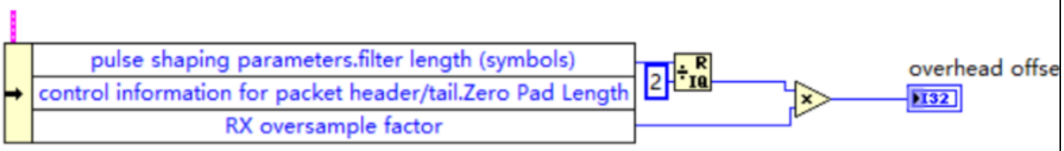
We can find the calculation method for delay in the LabVIEW program, thereby obtaining the theoretical relationship among the three parameters. The program is shown as follows:



From which we can find that the relationship we observed among three parameters is correct. Offset is calculated by:



Overhead is decided by the modulation parameter, following picture shows the way to calculate it:



So far, we have gained a deeper understanding of the calculation way and relationship among the three parameters and have validated our mathematical conclusions regarding their interdependence.

Early-Late Gate Algorithm test

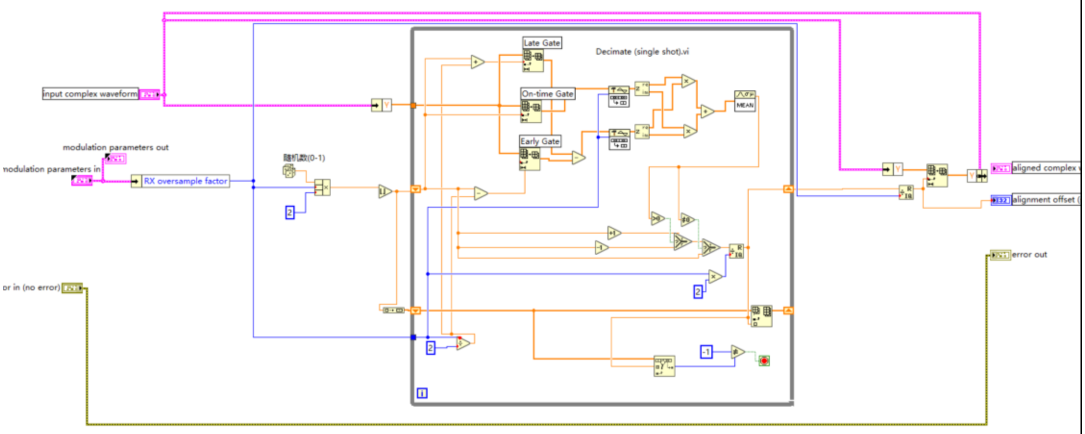
The Early-Late Gate algorithm follows the same principles as described earlier. The programming approach is as follows. The goal is to find the appropriate k, which make the $\delta/\delta k$

$J\delta[k]$ is equal to the zero. And the energy is max. When it is equal to zero, we get the Maximization of the Output Energy in Discrete Time at the delay $\tau = k$T/M.

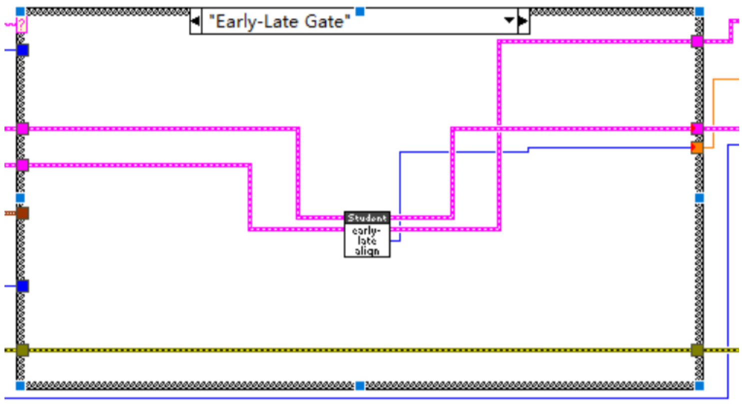
According to the equation:

$$J_\delta[k] = \sum_{n=0}^{P-1} 2Re\{r[nP+k](r^*[nP+k+\delta] - r^*[nP+k-\delta])\}$$

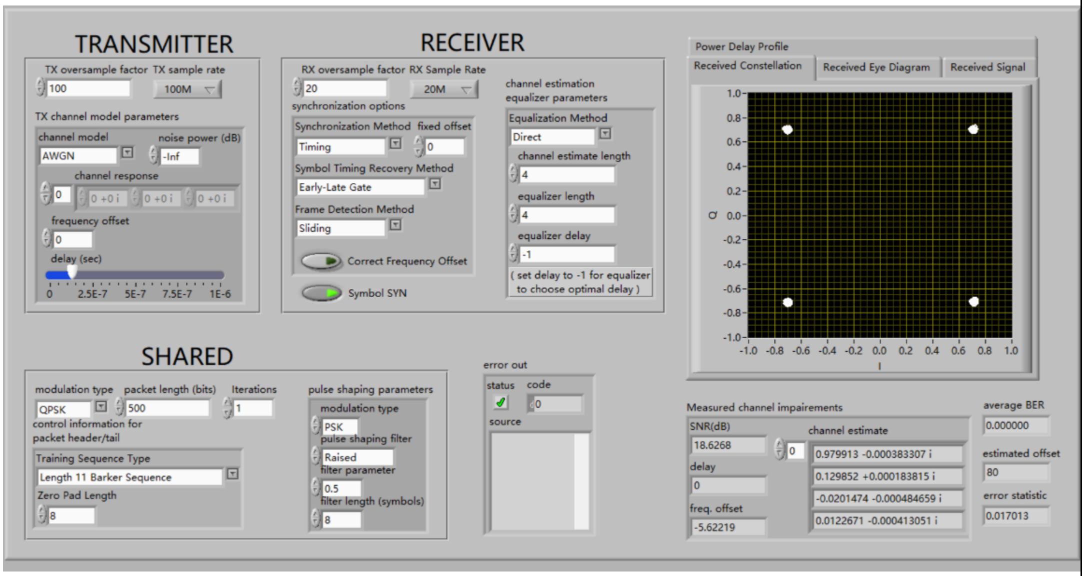
We can design the upper part program, and then we can determine whether $\delta/\delta k\ J\delta[k]$ is 0. We use the Select.vi to achieve it. If it is less than 0, k-1. If it is larger than 0, k+1. If it is equal to 0, end the loop. At the end, this section is used to provide the termination condition for the program. When a repeated 'k' occurs, the entire loop can be terminated to prevent the situation where 0 cannot appear, and the program remains stuck in an infinite loop without breaking out.

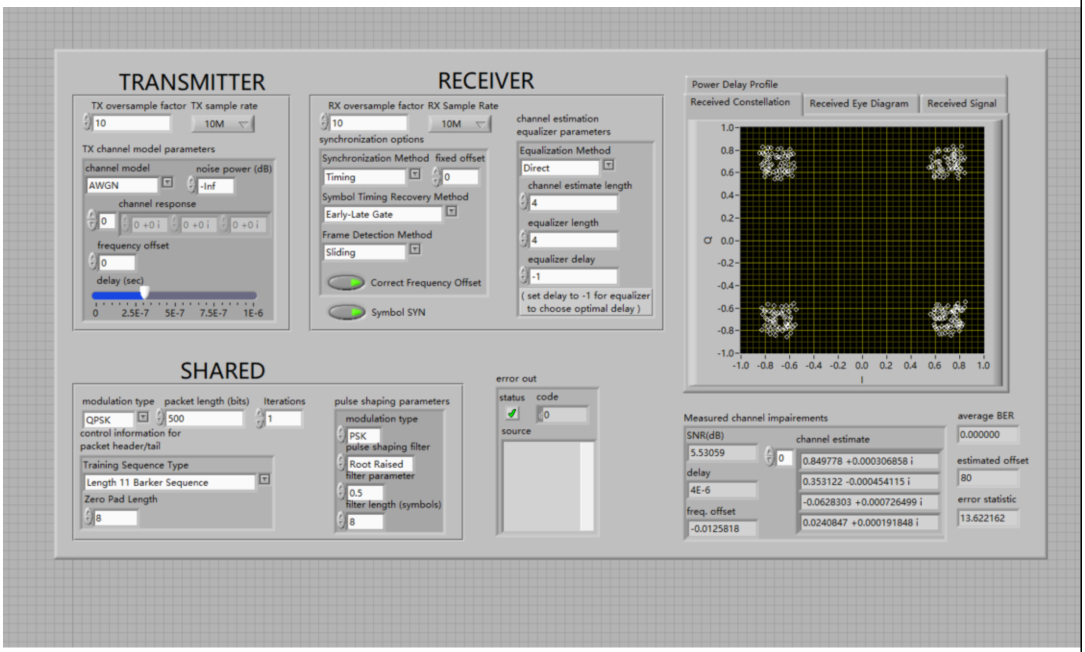


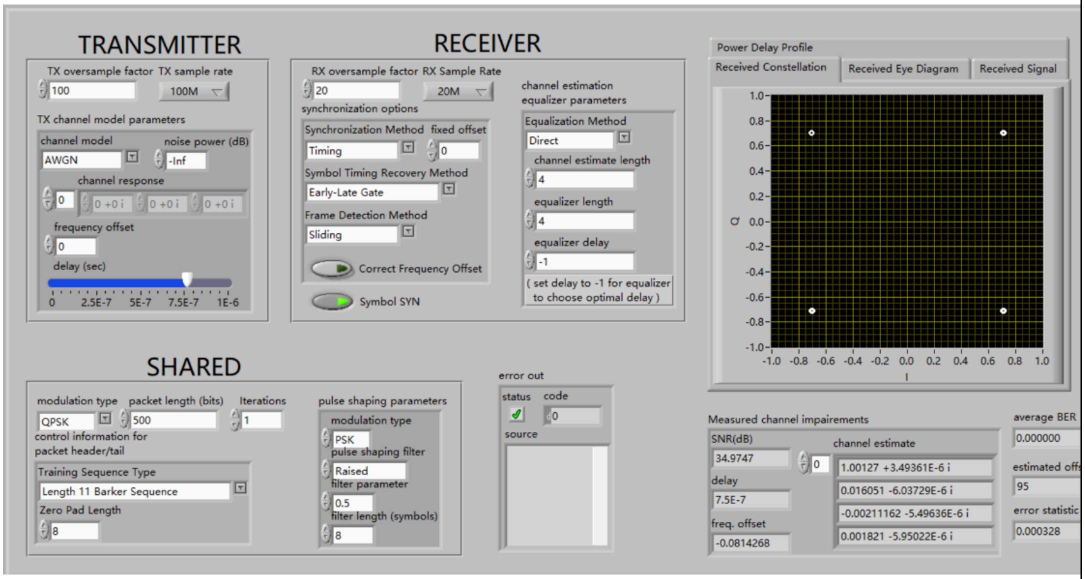
We encapsulate the above program as a subVI and then replace it in our symbol synchronization program. We validate its effectiveness through simulation and check for any programming errors.



The simulation results are as follows:


TRANSMITTER
TX oversample factor
TX sample rate
100
100M
TX channel model parameters
channel model
noise power (dB)
AWGN
-Inf
channel response
frequency offset
delay (sec)
RECEIVER
RX oversample factor
RX Sample rate
20
20M
synchronization options
Synchronization Method
fixed offset
Timing
Symbol Timing Recovery Method
Early-Late Gate
Frame Detection Method
Sliding
Correct Frequency Offset
Symbol SYN
channel estimation
equalizer parameters
Equalization Method
Direct
channel estimate length
equalizer length
equalizer delay
(set delay to -1 for equalizer
to choose optimal delay)
Power Delay Profile
Received Constellation
Received Eye Diagram
Received Signal
SHARED
modulation type
packet length (bits)
Iterations
QPSK
500
control information for
packet header/tail
Training Sequence Type
Length 11 Barker Sequence
Zero Pad Length
pulse shaping parameters
modulation type
PSK
pulse shaping filter
Raised
filter parameter
0.5
filter length (symbols)
error out
status
code
source
Measured channel impairements
SNR(dB)
18.6268
delay
freq. offset
-5.62219
channel estimate
0.979913 -0.000383307 i
0.129852 +0.000183815 i
-0.0201474 -0.000484659 i
0.0122671 -0.000413051 i
average BER
0.000000
estimated offset
80
error statistic
0.017013

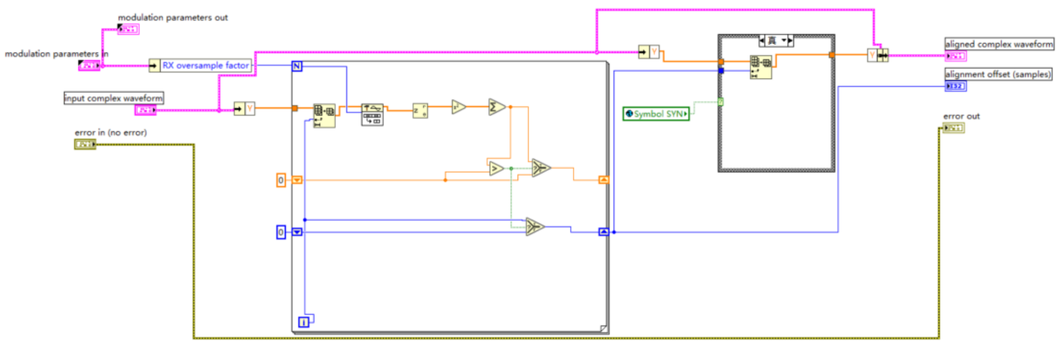

TRANSMITTER
TX oversample factor
TX sample rate
10
10M
TX channel model parameters
channel model
noise power (dB)
AWGN
-Inf
channel response
frequency offset
delay (sec)
RECEIVER
RX oversample factor
RX Sample Rate
10
10M
synchronization options
Synchronization Method
fixed offset
Timing
Symbol Timing Recovery Method
Early-Late Gate
Frame Detection Method
Sliding
Correct Frequency Offset
Symbol SYN
channel estimation
equalizer parameters
Equalization Method
Direct
channel estimate length
equalizer length
equalizer delay
(set delay to -1 for equalizer
to choose optimal delay)
Power Delay Profile
Received Constellation
Received Eye Diagram
Received Signal
SHARED
modulation type
packet length (bits)
Iterations
QPSK
500
control information for
packet header/tail
Training Sequence Type
Length 11 Barker Sequence
Zero Pad Length
pulse shaping parameters
modulation type
PSK
pulse shaping filter
Root Raised
filter parameter
0.5
filter length (symbols)
error out
status
code
source
Measured channel impairements
SNR(dB)
5.53059
delay
4E-6
freq. offset
-0.0125818
channel estimate
0.849778 +0.000306858 i
0.353122 -0.000454115 i
-0.0628303 +0.000726499 i
0.0240847 +0.000191848 i
average BER
0.000000
estimated offset
80
error statistic
13.622162

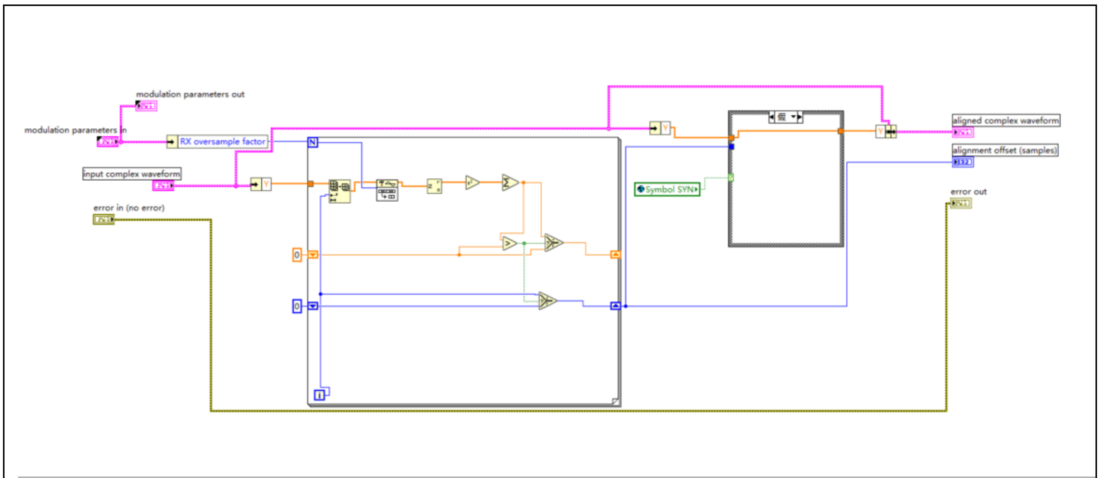


The simulation results are as follows. It can be observed that even with symbol synchronization enabled, the transceiver performance is not perfect under certain delay conditions. In some delay situations, the signal reception is almost negligible. Therefore, compared to the maximum energy method, the Early-Late Gate algorithm's performance is not ideal, but it has a smaller cost. Additionally, we observe an interesting phenomenon: when the delay is relatively high, the algorithm remains relatively stable, experiencing minimal variation within the given delay range.

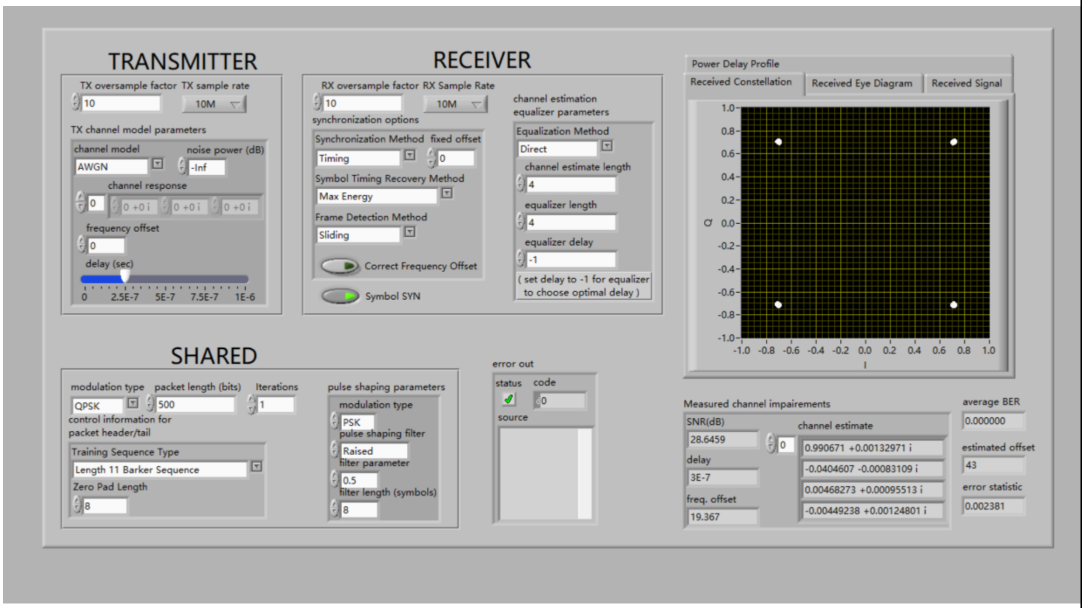
Experience

1. In-class lab screenshot

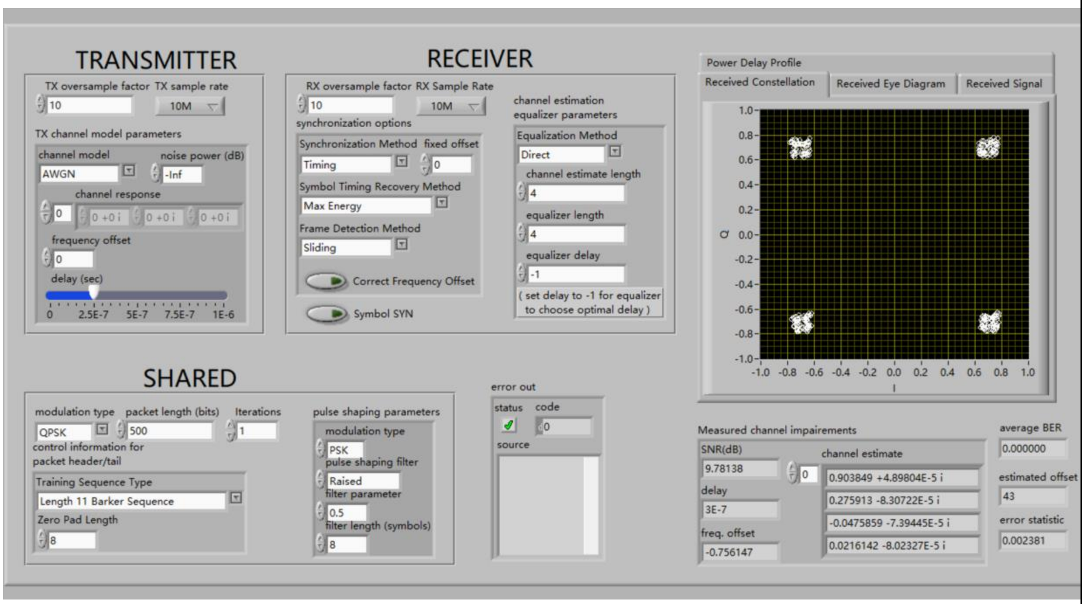




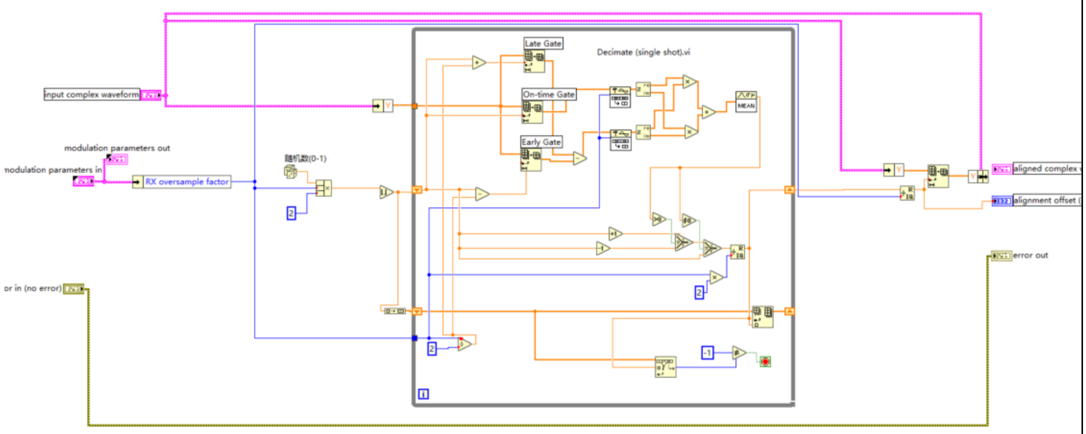
modulation parameters out
modulation parameters in
RX oversample factor
input complex waveform
error in (no error)
Symbol SYN
aligned complex waveform
alignment offset (samples)
error out

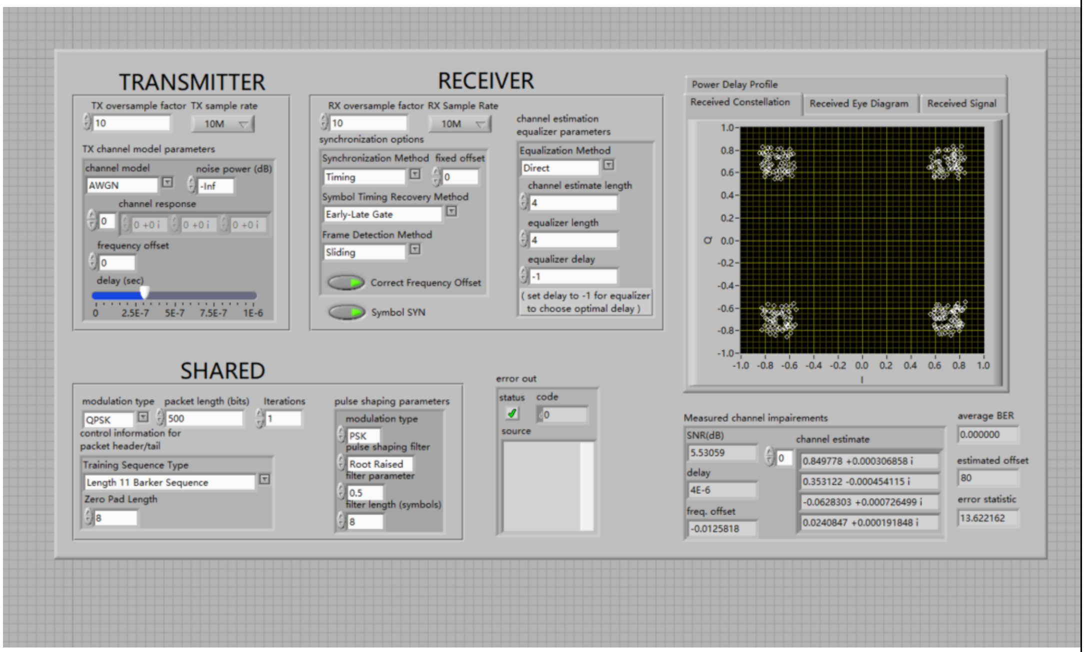


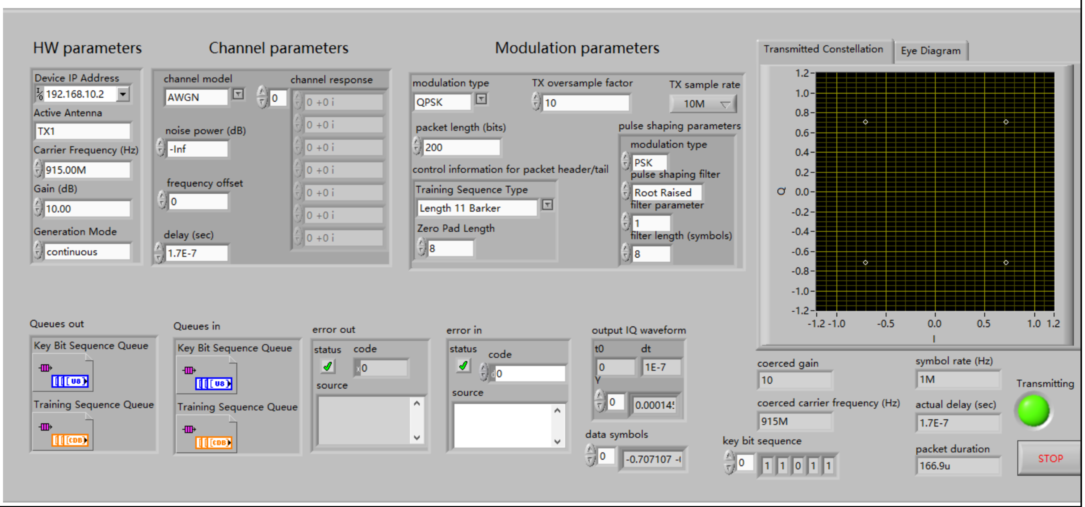
TRANSMITTER
TX oversample factor
TX sample rate
10
10M
TX channel model parameters
channel model
noise power (dB)
AWGN
-Inf
channel response
frequency offset
delay (sec)
RECEIVER
RX oversample factor
RX Sample Rate
synchronization options
Synchronization Method
fixed offset
Timing
Symbol Timing Recovery Method
Max Energy
Frame Detection Method
Sliding
Correct Frequency Offset
Symbol SYN
channel estimation equalizer parameters
Equalization Method
Direct
channel estimate length
equalizer length
equalizer delay
(set delay to -1 for equalizer to choose optimal delay)
Power Delay Profile
Received Constellation
Received Eye Diagram
Received Signal
SHARED
modulation type
packet length (bits)
Iterations
QPSK
500
control information for packet header/tail
Training Sequence Type
Length 11 Barker Sequence
Zero Pad Length
pulse shaping parameters
PSK
pulse shaping filter
Raised
filter parameter
0.5
filter length (symbols)
error out
status
code
source
Measured channel impairements
SNR(dB)
28.6459
delay
3E-7
freq. offset
19.367
channel estimate
0.990671 +0.00132971 i
-0.0404607 -0.00083109 i
0.00468273 +0.00095513 i
-0.00449238 +0.00124801 i
average BER
0.000000
estimated offset
43
error statistic
0.002381

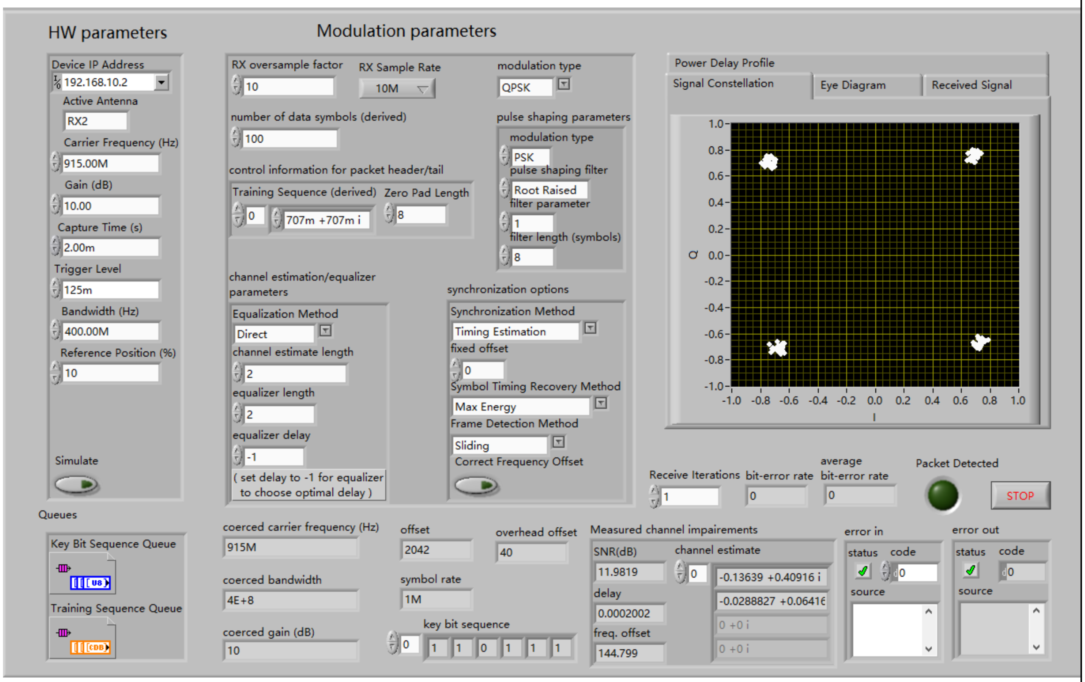


TRANSMITTER
RECEIVER
Power Delay Profile
Received Constellation
Received Eye Diagram
Received Signal
SHARED
Measured channel impairements
SNR(dB)
9.78138
delay
3E-7
freq. offset
-0.756147
channel estimate
0.903849 +4.89804E-5 i
0.275913 -8.30722E-5 i
-0.0475859 -7.39445E-5 i
0.0216142 -8.02327E-5 i
average BER
0.000000
estimated offset
43
error statistic
0.002381


input complex waveform
modulation parameters out
modulation parameters in
RX oversample factor
error in (no error)
Late Gate
On-time Gate
Early Gate
Decimate (single shot).vi
MEAN
aligned complex
alignment offset
error out


TRANSMITTER
TX oversample factor
TX sample rate
10
10M
TX channel model parameters
channel model
noise power (dB)
AWGN
-Inf
channel response
frequency offset
delay (sec)
RECEIVER
RX oversample factor
RX Sample Rate
synchronization options
Synchronization Method
fixed offset
Timing
Symbol Timing Recovery Method
Early-Late Gate
Frame Detection Method
Sliding
Correct Frequency Offset
Symbol SYN
channel estimation equalizer parameters
Equalization Method
Direct
channel estimate length
equalizer length
equalizer delay
(set delay to -1 for equalizer to choose optimal delay)
Power Delay Profile
Received Constellation
Received Eye Diagram
Received Signal
SHARED
modulation type
packet length (bits)
Iterations
QPSK
500
control information for packet header/tail
Training Sequence Type
Length 11 Barker Sequence
Zero Pad Length
pulse shaping parameters
PSK
pulse shaping filter
Root Raised
filter parameter
filter length (symbols)
error out
status
code
source
Measured channel impairements
SNR(dB)
5.53059
delay
4E-6
freq. offset
-0.0125818
channel estimate
0.849778 +0.000306858 i
0.353122 -0.000454115 i
-0.0628303 +0.000726499 i
0.0240847 +0.000191848 i
average BER
0.000000
estimated offset
80
error statistic
13.622162


HW parameters
Device IP Address
192.168.10.2
Active Antenna
TX1
Carrier Frequency (Hz)
915.00M
Gain (dB)
10.00
Generation Mode
continuous
Channel parameters
channel model
AWGN
channel response
noise power (dB)
-Inf
frequency offset
delay (sec)
1.7E-7
Modulation parameters
modulation type
QPSK
TX oversample factor
TX sample rate
10M
packet length (bits)
200
control information for packet header/tail
Training Sequence Type
Length 11 Barker
Zero Pad Length
pulse shaping parameters
PSK
pulse shaping filter
Root Raised
filter parameter
filter length (symbols)
Transmitted Constellation
Eye Diagram
Queues out
Key Bit Sequence Queue
Training Sequence Queue
Queues in
error out
error in
status
code
source
output IQ waveform
t0
dt
1E-7
data symbols
-0.707107
coerced gain
coerced carrier frequency (Hz)
915M
key bit sequence
symbol rate (Hz)
1M
actual delay (sec)
1.7E-7
packet duration
166.9u
Transmitting
STOP



2. Problem we meet and experience

1. When conducting experiments with USRP, it's common to encounter situations where the received signal is mostly noise. This occurs because the trigger level is set too low, causing the noise to overwhelm the desired signal. To address this issue, one can increase the trigger level to some extent.
2. An important issue which we should pay attention to during this experiment is that when we set oversample factor and corresponding sample rate, we should always keep their ratios to a fixed Ts, which in this experiment is $1\mu s$. Otherwise, the constellation may become a mass and the simulation will fail.
3. In USRP experiment, we studied deeply on the relationship between the offset and the delay, first we came up with our own assumption. Finally, we proved it through the results of USRP experiment, which ensures the correction of our experiment.

3. Respective contributions

We two finish the whole LabVIEW program together. Zhang Haodong complete procedure design including maximum energy algorithm design, plotting curve of symbol synchronization accuracy with changes in the oversampling factor, measuring signal transmission delay with USRP and early-late gate algorithm test. The introduction of basic principle in maximum energy algorithm and early-late gate algorithm were elaborated by Song Yihang.

Score	98