**Project 2: Text Transmission using USRP**

|  |  |
| --- | --- |
| **Author** | Name: 杨佳怡 孙逸涵 张旭东  Student ID: 12012146 12012128 12011923 |
| **Introduction**  In this project, we implement the text transmission using USRP and LabVIEW. First of all, we analyze the way to cut the text into packets, and the packet format we use in this project to cut the text. Then we use the Producer-Consumer Design Pattern to implement the launch delay, and use FIFO queue, USRP, and global variable to implement the text transmission between different program. After that, we use 16-PSK, 8-PSK, 4-PSK, and 2-PSK for digital modulation, and discuss the relationship between SNR and BER, alpha and BER, and filter length and BER, i.e., bandwidth and BER. Finally, we make a tentative exploration of ASK and FSK.  **Theoretical analysis**  For the signal which is transferred, the first step is to encode into bit streams and then encapsulate bit streams into packets. After that, packets is input into PSK modulator and then perform symbol mapping and pulse generation.   1. **Modulation**   Consider a sinusoidal carrier wave given by where means the Amplitude of carrier signal, means the frequency of the carrier signal.  For the n bits which need to transmit, divide equally to parts. If n is equal to 1, then the transmitted signal is . If n is greater than 1, the transmitted signal is .   1. **Demodulation**   The received signal is , where represents white gaussian noise. The received signal goes through two processes to the signal detector. One process is that the received signal is firstly multiplied by and then integrated over a period. Another process is that the received signal is firstly multiplied by and then integrated over a period. In the signal detector, there are thresholds to recover the transmitted bits.    Receiver 1   1. **Packet format**  * **Cut the text into packets**     In the process of packet sending, the data is first encapsulated into packets for digital modulation, and the complex baseband signal is obtained through pulse forming, and finally sent to USRP through sending queue.  In this project, we use **sub\_Generate\_Packet\_Array.vi** subVI to package packets according to packet format.    In the process of packet receiving, we first get the complex baseband signal from the receiving queue, and then get the packet bits after matching filtering and digital demodulation. Then, according to the packet format, we take out the data in the packet bits, and finally restore it to text through decoding.   * **Packet format**     The role of the fields in the packet format are as follows:   * Guard Band: Head protection interval; * Sync Seq: Locate the starting position of data; * Packet Number: Locate the packet in the text; * Data: Text content, i.e., message bits; * Cyclic Pad / Pad Data: Tail protection interval.   And the role of protection interval is to avoid the interference between adjacent packets during transmission.   1. **Implement transmission delay**     To implement transmission delay, we design our block diagram based on Producer-Consumer Design Pattern which is because each different loop in Producer-Consumer Design Pattern can run synchronously at different rates.  During text transmission, the rate at which the PC generates data packets is inconsistent with the rate at which the USRP sends data packets. If the rate at which the PC generates data packets is too fast and the rate at which the USRP sends data packets is too slow, packet loss may occur. Thus, we need use Producer-Consumer Design Pattern to separate the established response from the data to implement transmission delay.  In the Producer-Consumer Design Pattern, we use FIFO queues as buffer pools to control data input and output rates. If the queue is empty, new data cannot enqueue. When the queue is full, new data can only enqueue after the previous data dequeue. In this case, the data input rate and output rate are the same.   1. **Transmission between different program** 2. **Use FIFO queue**       We can use queue to implement transmission between different program. When the program is running, LabVIEW will first look up whether the queue already exists in the lookup table according to the name of the input queue. If so, it will directly return the queue reference. If not, create a new queue and add the queue reference to the lookup table for maintenance. In any program subVI, as long as the name of the queue is known, the queue with the corresponding name can be used. And we use **Obtain Queue** module’s Name terminal to name the queue.   1. **Use USRP**       In this project, we mainly use USRP to implement transmission between different program in wireless channel.  Note: Before we use USRP, we need to set the PC's IP address to 192.168.10.1.   1. **Use global variable**         We also can use global variable to implement transmission between different program. Global variables are stored in a VI file, but it only has the front panel, and no block diagram, in its front panel can be placed multiple controls, corresponding to multiple global variables, i.e., in a VI file can store multiple global variables.  **Lab results**  **Transmission and receiving of sine wave using USRP**   1. **Transmitter**     The figure shown above is the block diagram of the transmitter of the transmission of sine wave using USRP.  In this process, we need to use USRP driver functions to implement the design of transmitter. First, create **niUSRP Open Tx Session.vi** module, which can be used to set the IP address of USRP. And create **niUSRP Configure Signal.vi** module, and create input controls and output indicators respectively. Then create **niUSRP Write Tx Data (poly).vi** module in while loop, which can be used to transmit the complex baseband signal to USRP. And then create **niUSRP Close Session.vi** module, and create an error output pipeline. Finally, create a **Initialize Array** module, create a complex constant 1+0i at its element terminal, and create a dimension input control at its dimension terminal, and input the initialized array to the data terminal of **niUSRP Write Tx Data (poly).vi** module.   1. **Receiver**     The figure shown above is the block diagram of the receiver of the transmission of sine wave using USRP.  In this process, we need to use USRP driver functions to implement the design of receiver. First, create **niUSRP Open Rx Session.vi** module, **niUSRP Configure Signal.vi** module, and **niUSRP Initiate.vi** module, respectively, and create corresponding input controls and output indicators of them. And create **niUSRP Fetch Rx Data (poly).vi** module in while loop, which can be used to get USRP received data. In while loop, create **Complex To Re/Im** module to get the real and imaginary part of received waveform, and combine the extracted real and imaginary parts and input them into the waveform graph. Finally create **niUSRP Abort.vi** module and **niUSRP Close Session.vi** module, and create an error output pipeline.  **Text transmission using USRP**   1. **Transmitter**     First, slicing text into packets and then each packet is encoded as bit streams. After that, the PSK modulator maps and pluses shape the bit streams into the analog signal, which is (n=1) or (n>1).   1. **Receiver**     In the receiver, the first step is to transfer the analog signal to the digital signal by analog-to-digital converter. Then matching filtering and synchronization detection are performed. After that, decode bitstreams into text.     1. **Results** 2. **One USRP transmits the signal and the same one receives the signal**     BPSK    QPSK    8PSK    16PSK   1. **One USRP transmits the signal and another one receives the signal**     BPSK Transmitter    BPSK Receiver    QPSK Transmitter    QPSK Receiver    8PSK Transmitter    8PSK Receiver    16PSK Transmitter    16PSK Receiver  **Evaluation**  The following is the introduction of the result evaluation and analysis part.  Firstly, we explored the relationship between bit error rate and signal-to-noise ratio. In this exploration process, taking QPSK as an example, the distance between the transmitter and the receiver and the gain of the transmitter and the receiver are adjusted to change the transmission channel. The size of the noise is changed to change the signal-to-noise ratio, and the bit error rate is calculated by comparing the recovered bit stream with the bit stream at the transmitter at the receiving end. In the acquisition of experimental data, the data acquisition of the same channel environment and gain is averaged for many times, thereby increasing the stability of the data and avoiding the spikes in the drawing of the experimental results.  **Experiment results:**    **Theoretical Analysis:**    Above the first figure is the image of the bit error rate versus the signal-to-noise ratio drawn by the measurement results of this experiment. From the figure, we can find that the bit error rate decreases as SNR increases. When the SNR<5dB, the bit error rate is about 0.4. When the SNR>10dB, the bit error rate is small, about 0dB. The experimental results are basically consistent with the graph trend drawn by the theoretical calculation above.  Then we explore the relationship between alpha, bandwidth and bit error rate.  Below is the time waveform and frequency-domain waveform of raised-cosine signal with different roll-off factor alpha.    When the pulse signal is a raised cosine signal, the roll-off factor alpha determines the waveform of the pulse shaping function and the bandwidth in the frequency domain. It can be seen from the time domain waveform that when the alpha is closer to 0, the raised cosine function is closer to the sinc function, and the corresponding bandwidth in the frequency domain is narrower; when the alpha is closer to 1, the time domain waveform of the raised cosine function has less tailing effect, but wider the bandwidth occupied in the corresponding frequency domain.  **BER to alpha figure:**    Under the situation of SNR=20dB, QPSK modulation and demodulation mode, the relationship between BER and alpha measured by experiments can be seen above, BER decreases with the increase of alpha, and when alpha tends to 1, the system The better the signal recovery. This is consistent with our theoretical analysis. When the alpha value is 0, the bandwidth utilization rate is the highest, but at this time, the waveform will cause inter-symbol crosstalk due to the tailing effect, resulting in an increase in the bit error rate. Similarly, alpha determines the bandwidth of the signal. The closer alpha is to 0, the smaller the bandwidth. Therefore, the relationship between the pulse waveform bandwidth and the bit error rate can also be seen from the left figure.  **BER to filter length figure:**    In the graph of BER versus filter width, it can be seen that when the filter width is too small or too large, the bit error rate increases. Such experimental results are also in line with theoretical analysis. When the filter width is small and smaller than the bandwidth of the pulse waveform, the filtered waveform will be distorted and the recovery effect will be deteriorated; when the filter width is large, it will lead to the component of noise increases, thereby increasing the bit error rate. | |
| **Experience**   1. In this project, I was responsible for the analysis of packet format, the way to cut the text into packets in this project, three ways to implement transmission between different program, and implement the 16-PSK, 8-PSK, 4-PSK, and 2-PSK using one USRP and using two USRPs with teammates. When I first started working on this project, I was completely new to the software radio technology, and I also cannot understand why we need to add so many fields to the packet when wrapping data into packets. After I look up some materials online, and open the subVI designed by Dr. Wu to learn the process of the text transmission, so now I have some superficial cognition of the application of software radio and realize the powerful function of USRP.   孙逸涵: Packet format, transmission between different program, 16-PSK, 8-PSK, 4-PSK, and 2-PSK, PPT making, report writing.   1. In this project, I complete the construction of the receiver, help complete the construction of the transmitter, help study the method of cutting text into packets, and tentative exploration of ASK/FSK. From the progress, I learned three methods of cutting text into packets, including using queues, using USRP and using global variables. What’s more, I have a deep understanding of the relationship between the signal-noise ratio and the bit error rate and he relationship between alpha, bandwidth and bit error rate.   张旭东: Construction of the receiver, Implementation of 2/4/8/16 PSK, PPT making, report writing.   1. In this project, I was responsible for the implement transmission for sine using USRP and evaluate and explore the relationship between BER and SNR, also the connection between BER, alpha and bandwidth of the transmitting raised-cosine signal. From this progress, finding a specific way to describe quantitatively SNR is a challenge for me. However, after trying so many times and reading the transmission and receiving VI, I have better understand of this program. This is a great experience for me to have a cognition of the powerful programmable device USRP and implementation of software radio.   杨佳怡: transmission for sine signal using USRP and evaluate and explore the relationship between BER and SNR, also the connection between BER, alpha and bandwidth of the transmitting raised-cosine signal., PPT making, report writing.  Note: the Video about sine signal transmission not recorded as the lab202 is locked during Weekend. | |
| **Score** | 100 |