QPSK Transmission from PC to PC Using GNU Radio

Signal Processing with MATLAB & Python Project Paper
Bachelor of Science
im Studiengang Elektrotechnik
an der Fakultät für F07
der Technischen Hochschule Köln

Submitted by:	Mohamed Aziz Gam & Abdeljalil Biad
Student Nr.:	11147679 - 11113756
Email:	mohamed_aziz.gam@smail.th-koeln.de
	abdeljalil.biad@smail.th-koeln.de

Submitted to: Prof. Dr. Ing. Harald Elders-Boll

Cologne, 05/03/2024

Table of Contents

List of Illustration

1 Introduction

This paper presents the implementation and evaluation of a Quadrature Phase Shift Keying (QPSK) transmission system using GNU Radio software for software-defined radio (SDR) communication. Building upon the GNU Radio tutorial, the project aimed to establish reliable QPSK transmission from one PC to another utilizing the audio interface on each PC and a wired transmission setup. The methodology involved configuring GNU Radio blocks for QPSK modulation, transmitting the modulated signal over a simulated channel, and demodulating it at the receiving end. Results demonstrate successful transmission with considerations for signal quality and error rates. Additionally, discussions encompass the practical implications of the project's findings and avenues for future enhancements in digital communication systems.

2 Transmitting a QPSK Signal

In the initial phase of this project, the first task is to construct the flowgraph for the transmitter. This involves generating a signal and subsequently modulating it onto a complex constellation. Subsequently, the specific requirements and corresponding solutions for this process will be explored.

2.1 Generating a GLFSR Bit Stream (Galois LFSR pseudo-random source)

While random sources offer true randomness, making them suitable for certain scenarios, they present challenges in terms of reproducibility, testing, performance evaluation, algorithm development and resource efficiency. In contrast, GLFSR sources provide deterministic data sequences based on a seed value, offering advantages such as reproducibility, simplified testing and debugging, accurate performance evaluation with known reference sequences, ease of algorithm development by isolating effects, and efficient use of resources [3].

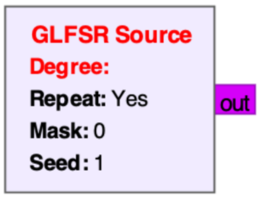


Figure 1: GLFSR Block

The GLFSR Block has 4 Parameters:

Degree: Degree of shift register must be in [1, 32]. If mask is 0, the degree determines a default mask (see digital_impl_glfsr.cc for the mapping).

Repeat: Set to repeat sequence.

Mask: Allows a user-defined bit mask for indexes of the shift register to feed back.

Seed: Initial setting for values in shift register.[3]

The block generates 1 bit at a time. The issue arises because the output type of this block is byte, causing each bit to be generated within a byte. To address this concern, a "Stream Mux" block and the "Pack K bits" block are employed. Within the system, the stream mux block is configured with two inputs, with the length parameter set to (m_sequence_length, 1). This configuration facilitates the generation of an extended m-sequence with a length equal to the FFT size, determined as 2^d (where d= degree_glfsr). Subsequently, the Pack K bits block is utilized to pack these individual bits or groups of bits into bytes. The parameter of the "Pack K bits" block operates independently of the degree of the feedback polynomial in the shift register. Instead, it focuses on consolidating 8 bits of the extended m-sequence into a single byte, as necessitated by the downstream constellation modulator block. Thus, the parameter K is set to 8 to effectively meet this byte packing requirement. [4][5]

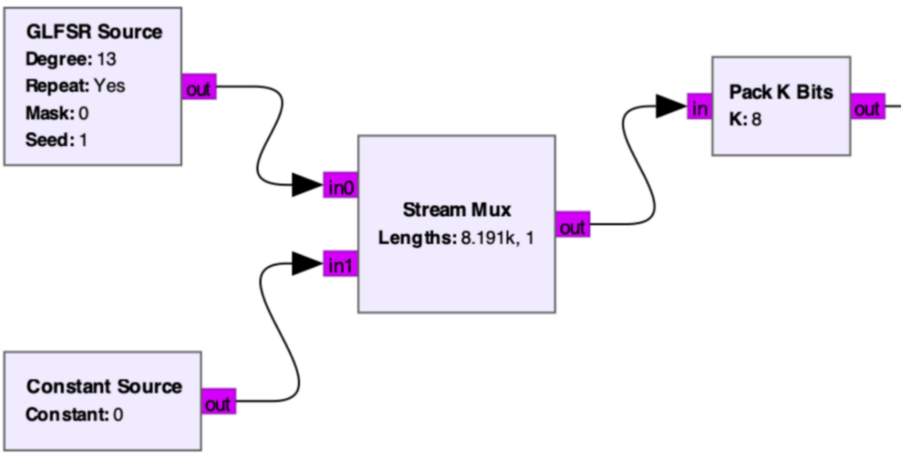


Figure 2: Flowgraph of generating the signal and packing the bits

2.2 Modulating onto a Complex QPSK Constellation

2.2.1 Small introduction to Modulation types

Different Modulation Techniques such as Amplitude modulation (AM), phase modulation (PM), and frequency modulation (FM) stand out as prevalent modulation techniques extensively employed within communication systems. AM entails the modulation of a carrier signal's amplitude in correspondence with the amplitude variations present in the modulating signal, which carries the information. Conversely, PM revolves around the manipulation of the phase of the carrier signal, adjusting it in accordance with alterations in the modulating signal.

FM, on the other hand, revolves around the adjustment of the carrier signal's frequency, aligning it with the characteristics exhibited by the modulating signal. These modulation techniques play vital roles in adapting the transmitted signals to suit the diverse requirements and characteristics of various communication channels. [6]

As mentioned in the introduction what interests us in this project is The Phase Modulation. And exactly QPSK Modulation which stands for Quadrature Phase shift keying.

2.2.2 What is QPSK and how is it Modulated?

In digital modulation techniques a set of basis functions are chosen for a particular modulation scheme. Generally, the basis functions are orthogonal to each other. the basis function are chosen, that any vector in the signal space can be represented as a linear combination of the basis functions. In Quadrature Phase Shift Keying (QPSK) two sinusoids (sin and cos) are taken as basis functions for modulation. Modulation is achieved by varying the phase of the basis functions depending on the message symbols. In QPSK, modulation is symbol based, where one symbol contains 2 bits.[7]

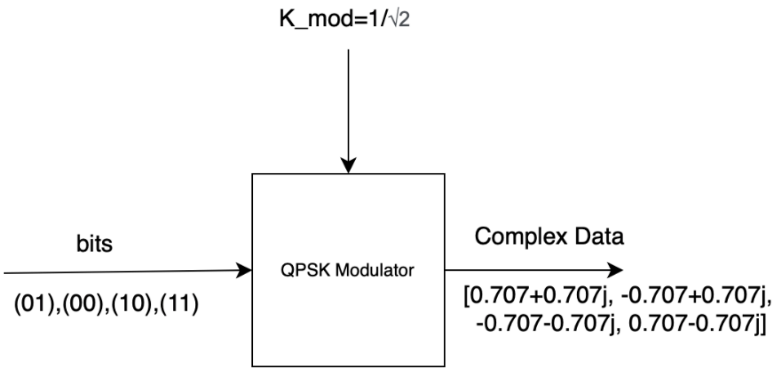
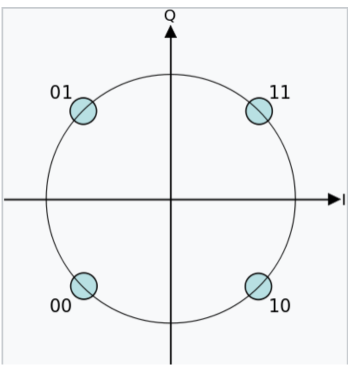


Figure 3: Example of a QPSK Modulator



Encoding Rule Example:

00 → (-1-1j)*KMOD

01 → (-1+1j) * KMOD

10 → (1-1J)*KMOD

11 →→ (1+1J)*KMOD

Figure 4: Example of a π/4 QPSK Constellation

2.2.3 Modulating The Transmitted Signal

2.2.3.1 Challenges

a) Handling Samples per Symbol

The parameter "Samples per Symbol" in the Constellation Modulator block determines the number of samples used to represent each symbol in the modulation scheme. Setting this parameter appropriately ensures that the modulation process aligns with the desired sample rate and maintains consistency across the flowgraph.[8]

b) Excess Bandwidth

To control the bandwidth of the transmitted signal, a root raised cosine (RRC) pulse shaping filter was employed within the Constellation Modulator block. This filter, controlled by the "Excess BW" parameter, helps contain the spectral content of the transmitted signal within desired limits, ensuring efficient spectrum utilization and compliance with communication standards.[8]

c) Root Raised Cosine (RRC) Pulse Shaping Filter

The RRC filter shapes the transmit signal to limit spectral splatter within desired bandwidth limits. However, it also introduces inter-symbol interference (ISI) due to its finite impulse response. To address this, a matched filter (MF) is employed at the receiver end, taking the form of another RRC filter. By convolving the received signal with this filter (Nyquist), ISI is mitigated, ensuring the fidelity of the received QPSK signal.[8]

2.2.3.2 Modulation Solution: Leveraging Constellation Rect. Object and Constellation Modulator Blocks for QPSK Modulation

To complete the construction of the transmitter and achieve the project's goal of establishing communication between two PCs using QPSK transmitter and receiver, the "Constellation Rect. Object" and "Constellation Modulator" blocks serve as integral components in the QPSK modulation scheme. These blocks facilitate the modulation of digital data into a complex baseband signal suitable for transmission over the channel.[8]

The "Constellation Rect. Object" block is utilized to define the QPSK constellation points, mapping digital symbols to complex numbers. In the setup, the

Symbol Map defines the digital symbols [0, 1, 2, 3], while the Constellation Points specify the corresponding complex values [0.707+0.707j, -0.707+0.707j, -0.707-0.707j, 0.707-0.707j]. These parameters, along with Rotational Symmetry, Real Sectors, Imaginary Sectors, and sector widths, enable accurate representation of the QPSK constellation.[9]

Once the constellation is defined, the "Constellation Modulator" block is utilized to modulate the digital data. This block applies differential encoding to the defined constellation, converting the input byte stream into a complex modulated signal at baseband. Parameters such as Samples per Symbol (sps), Excess BW, and logging options ensure efficient and accurate modulation.[10]

After generating and packing the bits, these blocks are utilized to perform QPSK modulation for several reasons. Firstly, by defining the constellation points, the mapping between digital symbols and complex numbers is established, enabling efficient signal representation. Secondly, the modulator applies differential encoding and filtering, enhancing the robustness and performance of the transmitted signal. Overall, these blocks play a crucial role in converting digital data into a modulated signal ready for transmission, ensuring reliable communication in the QPSK system.

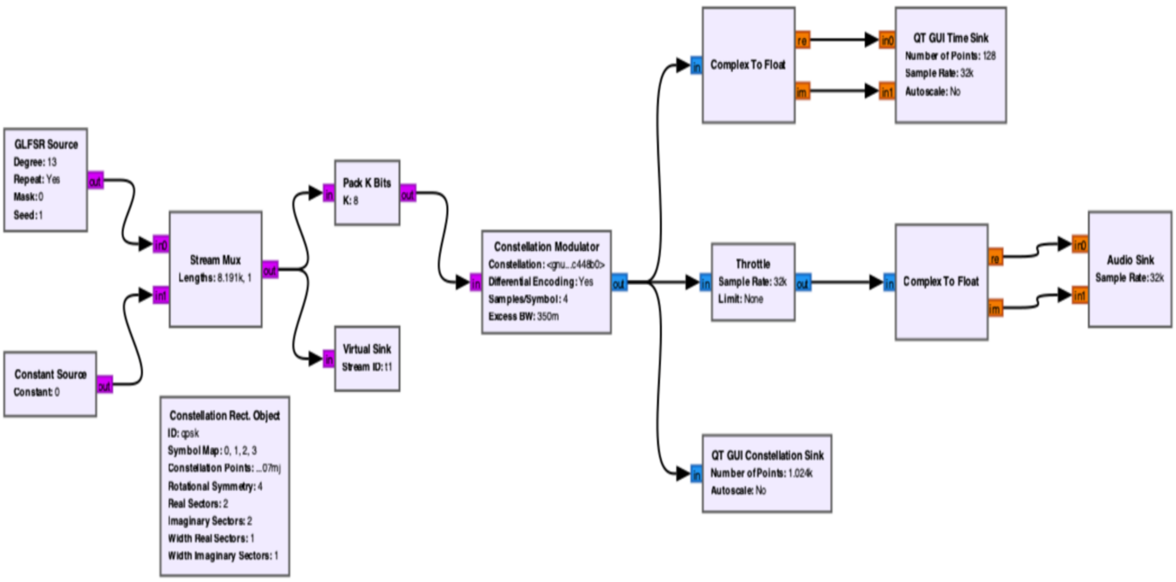


Figure 5: Transmitter flowgraph (Tx)

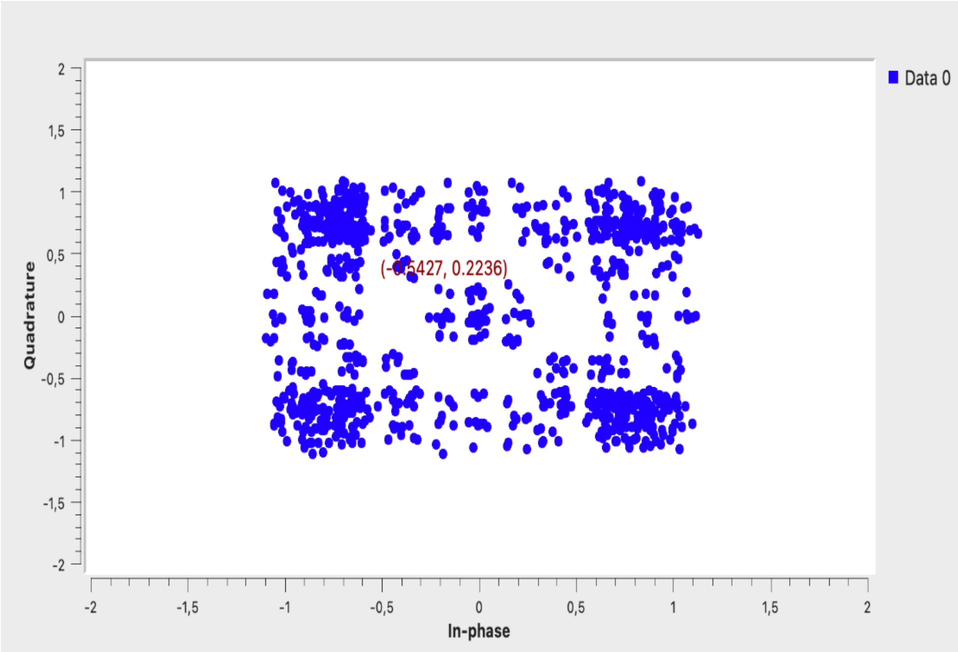


Figure 6: QT GUI Constellation Sink of the transmit signal

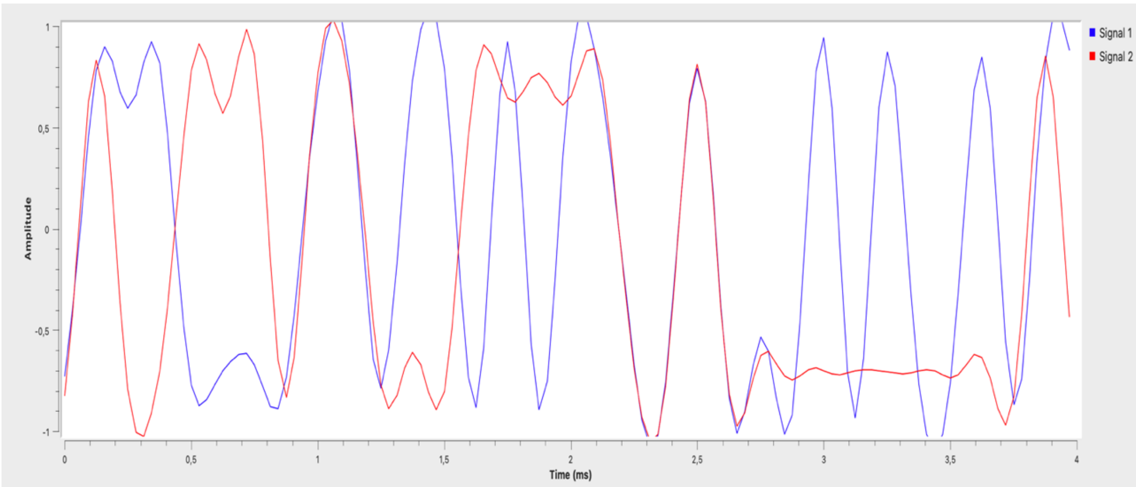


Figure 7: QT GUI time Sink of the Transmit Signal (In phase und Quadrature phase)

3 Receiving a QPSK Signal

3.1 Polyphase Clock Sync Block

The Polyphase Clock Sync block serves multiple functions in the reception of a QPSK signal. Firstly, it facilitates clock recovery, ensuring synchronization with the received signal. Secondly, it applies a receiver matched filter to eliminate inter-symbol interference (ISI). Lastly, it down samples the signal, reducing the number of samples per symbol. The block configuration typically involves setting the number of filters and loop bandwidth parameters to optimize performance. By processing the output of the channel model through the Polyphase Clock Sync block, the received signal undergoes crucial synchronization and filtering stages, preparing it for further processing. [8][11]

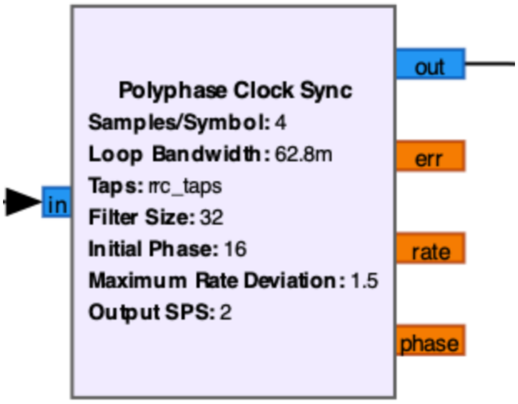


Figure 8: Polyphase Clock Sync Block with the used parameters

3.2 Multipath & Equalizer

3.2.1 Multipath

Multipath is a common phenomenon in communication environments where signals take multiple paths from transmitter to receiver due to reflections off surfaces. This results in signal distortions caused by constructive and destructive interference. To mitigate multipath effects, advanced equalization techniques are employed to adaptively compensate for channel distortions and ensure reliable signal reception.[8]

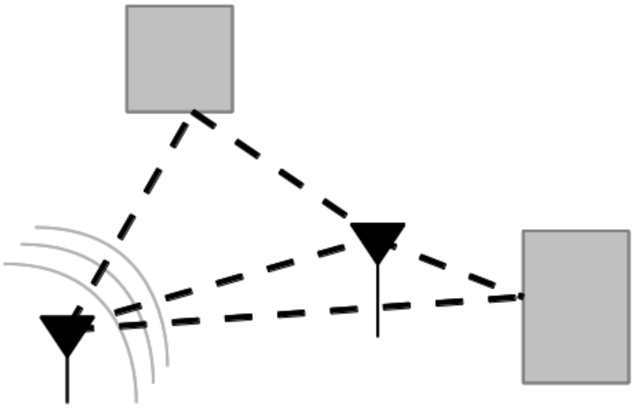


Figure 9: Multipath Illustration

3.2.2 Equalizer

The Equalizer block, integrated with an Adaptive Algorithm such as the Constant Modulus Algorithm (CMA), assumes a pivotal role in the reception of QPSK signals by addressing the adverse effects of channel distortions induced by multipath propagation and frequency-selective fading. Employing an FIR filter, the Equalizer executes linear equalization on the incoming signal, dynamically adjusting its coefficients to compensate for fluctuations in the channel's characteristics. In this configuration, the adoption of the Constant Modulus Algorithm (CMA) underscores a blind equalization approach, well-suited for signals exhibiting consistent amplitude or modulus, a characteristic often observed in digital signals like QPSK.[8][12]

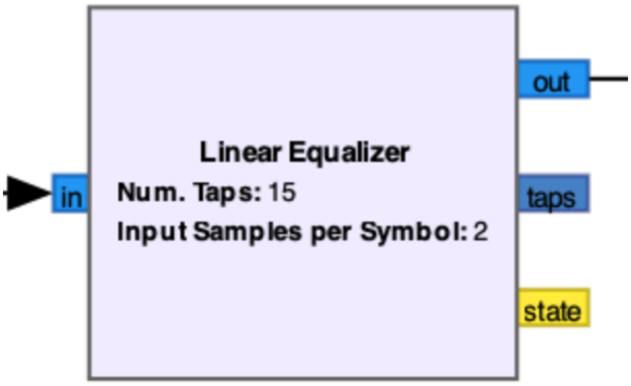


Figure 10: Linear Equalizer Block

During operation, the Equalizer dynamically adjusts its coefficients based on the received signal, aiming to invert and cancel out channel distortions effectively.

The number of taps for the FIR filter, set to 15 in the configuration, determines the Equalizer's ability to adapt to varying channel conditions. Furthermore, the Equalizer decimates the signal to the symbol rate specified by the "Samples per Symbol" (SPS) parameter, ensuring synchronization with the symbol timing.[8][12]

After receiving the signal and utilizing the Polyphase Clock Sync and Linear Equalizer Block, here is how the flowgraph looks:

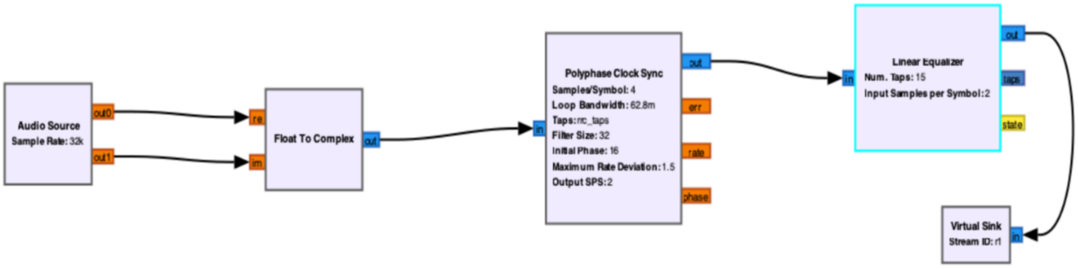


Figure 11: Stage 1 of Receiving the Signal

3.3 Phase and frequency correction

Having successfully addressed channel distortions through equalization, the next challenge lies in rectifying phase and frequency offsets present in the received signal to ensure precise demodulation. Phase and frequency synchronization are paramount for accurate symbol recovery, especially in complex modulation schemes like QPSK.

3.3.1 Costas Loop for Frequency and Phase Synchronization

To address the challenge of phase and frequency synchronization in the QPSK reception process, the Costas Loop block is utilized with a 4th-order loop configuration. This setup enables the Costas Loop to synchronize effectively to the center frequency of the received signal and downconvert it to baseband. Operating by tracking both phase and frequency deviations over time, the Costas Loop ensures precise alignment of the received signal's constellation points. This meticulous synchronization process is crucial for ensuring the accurate recovery of symbols from the received signal. Leveraging the “phase_est tag” generated by the Correlation Estimator, the Costas Loop corrects phase offsets, further enhancing synchronization accuracy. Ultimately, this pivotal step guarantees the fidelity of the demodulated signal, establishing a solid foundation for subsequent decoding processes with heightened precision and reliability.[8][13]

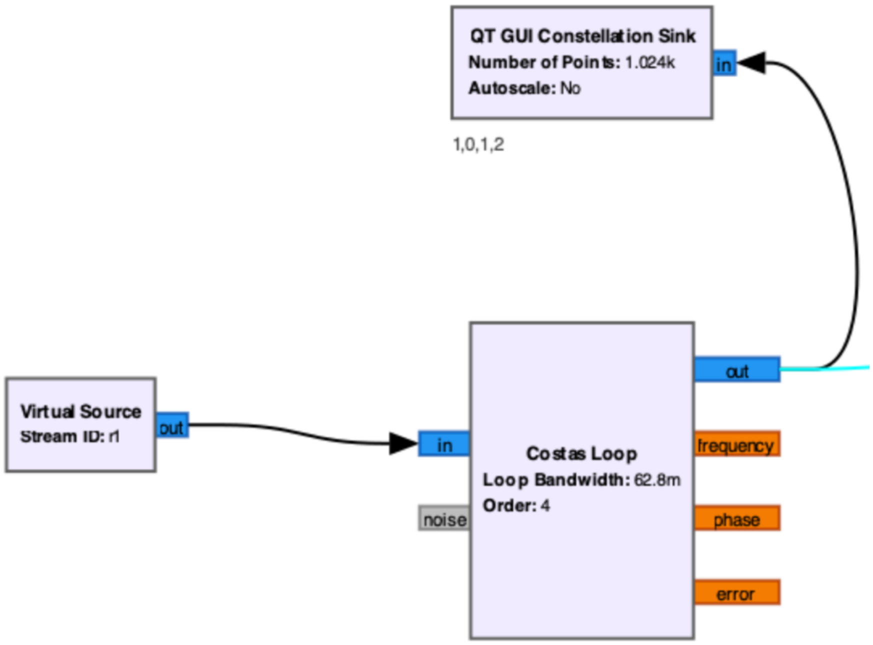


Figure 12: Using the Costas Loop

3.4 Decoding

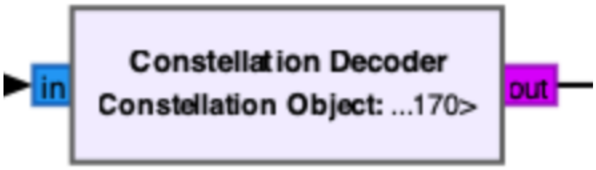
Upon reception of the demodulated signal, the decoding process commences, aiming to accurately recover the original transmitted data.

3.4.1 Constellation decoder

Following the Costas loop, the incorporation of a Constellation Decoder represents a crucial step in the process. However, this step presents a notable challenge. As the decoder interprets the received symbols and maps them back to their respective complex constellation points, a pivotal challenge arises: ensuring precise alignment between the symbols and their corresponding constellation points from the transmission phase.[8]

This requires a meticulous approach when looking at how the symbols, which range from 0 to 3 in a QPSK scheme, are mapped to the constellation points. It becomes clear that the decoding process must take into account the potential ambiguity inherent in the symbol-to-constellation mapping. This ambiguity, often manifested as a 90-degree phase difference within the constellation, could compromise the accuracy of the decoding if not properly addressed[8].

To address this challenge, a strategic approach is adopted: differential symbol transmission. Instead of directly transmitting the constellation itself, symbols are encoded and transmitted as the difference between successive symbols within the constellation. This is accomplished by activating the Differential setting in the Constellation Modulator block. By employing this technique, transmitted symbols are differentially encoded, mitigating the risk of misalignment between the symbols and their corresponding constellation points during decoding. As a result, the integrity and accuracy of the decoding process are upheld, safeguarding against potential ambiguities and enhancing the reliability of the communication system.[8]



As a parameter:

Constellation Object: qpsk (the constellation rect. Object with ID: qpsk that we declared while modulating the signal)

Figure 13: Constellation Decoder block

3.4.2 Differential Decoding & Mapping

After utilizing the Constellation decoder, the Differential Decoder is employed. This component works to interpret the differential-coded symbols, restoring them to their original states based on the phase transitions they undergo, while considering the modulus of the code's alphabet, which in this case is 4. The modulus value of 4 indicates the size of the code's alphabet, corresponding to the number of symbols used in the modulation scheme. In the context of QPSK modulation, where each symbol represents two bits of information, the alphabet size is 4, allowing for the representation of four distinct symbols (0, 1, 2, and 3).[8][14]

Following differential decoding, the symbols are restored to their original values. However, further adjustment is required to ensure compatibility with the transmission mapping. The Map block reconciles the decoded symbols with the original constellation mapping, aligning them appropriately for subsequent processing.[8]

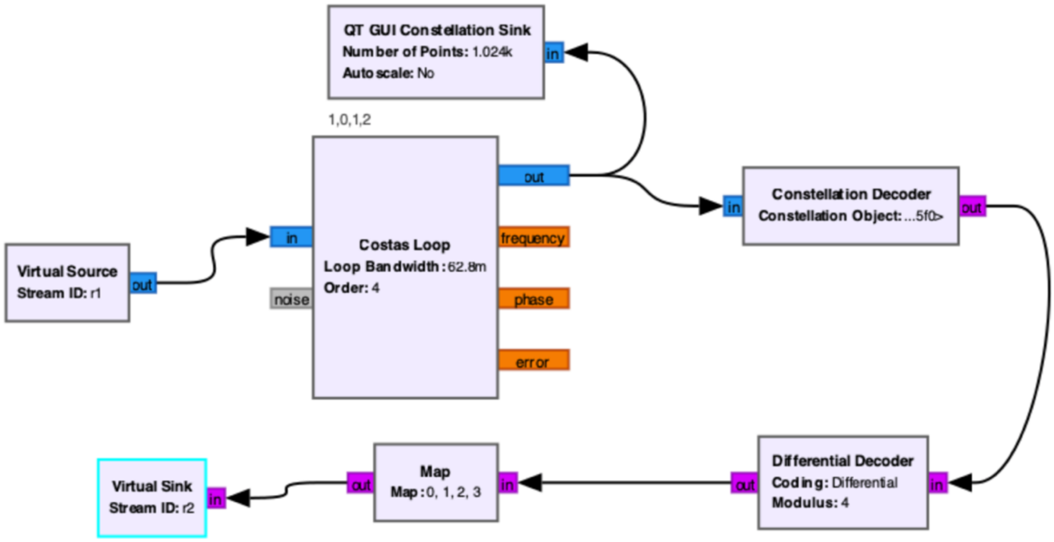


Figure 14: Stage 2 - Costas Loop and Decoding the Received Signal

3.4.3 Unpacking Bits and Synchronization/Comparison with Transmitted Data

To initiate the comparison between the received and transmitted signals, the transmitted signal was first regenerated using the “GLFSR Source” Block with the same degree used during signal transmission. Next, the bits were unpacked using the Unpack K Bits block to extract individual bits from the packed bytes, preparing them for further processing. Subsequently, the unpacked bits were converted into float values using the Map and Char to Float blocks, facilitating smoother signal manipulation in subsequent stages. The transformed signal then underwent a Fast Fourier Transform (FFT) with the size 2^d, with d representing the degree of the GLFSR source, transitioning from the time domain to the frequency domain.

Next, the Multiply Conjugate block was applied to perform complex conjugate multiplication, a critical operation for correlation-based synchronization techniques. This operation facilitated correlation between the received signal and reference patterns, essential for accurate synchronization and timing recovery. The signal was then transformed back from the frequency domain to the time domain using the Inverse Fast Fourier Transform (IFFT) Block. To enhance the processed signal, it was transformed from a complex format to a float format using the Complex to Real block. Subsequently, the signal went through the "Fast Multiply Const" Block and the "Absolute (ABS)" Block, preparing it for peak detection.

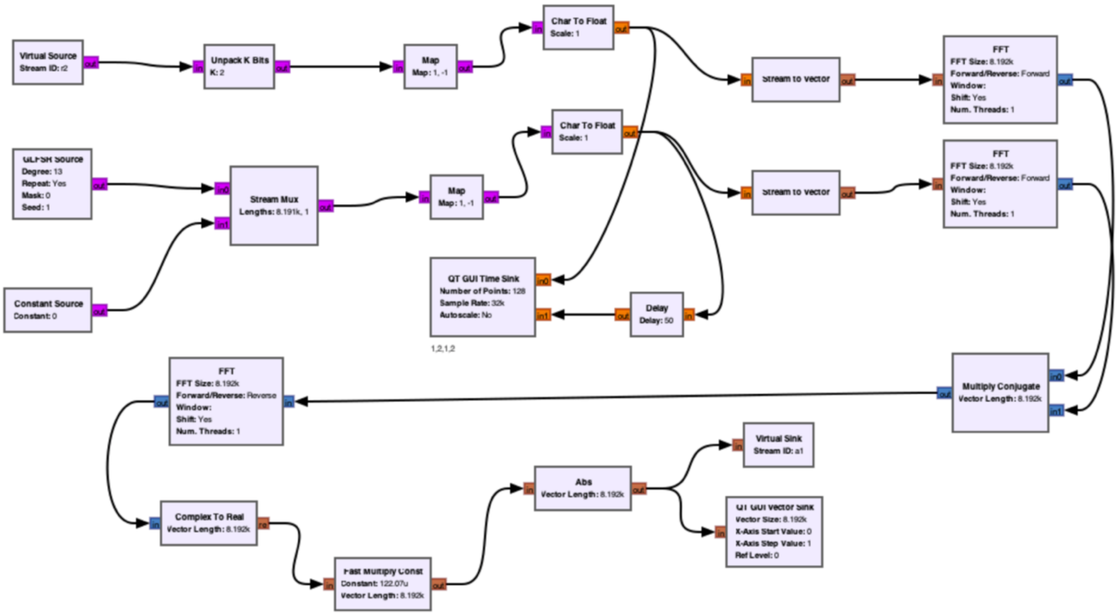


Figure 15: Stage 3 - Comparing the Received Signal with the Transmitted Signal and preparation for Peak Detection

The final step involves peak detection using the Max and Argmax blocks. Configured with a vector length equal to the FFT size, the Max block compares vectors from multiple streams and determines the maximum value from each vector over all streams, providing insights into the signal's strength or amplitude. Similarly, the Argmax block compares vectors and identifies the index in the vector and the stream number where the maximum value occurs. By determining the exact location of the peak, correlation between the received signal and the transmitted signal is achieved.[15][16]

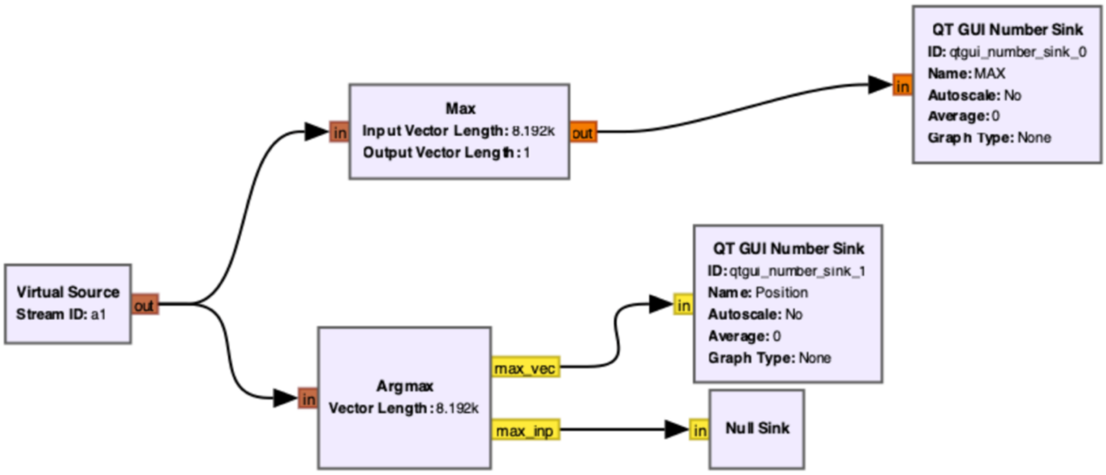


Figure 16: Peak Detection with “Argmax” Block for the Position of the Peak and “Max” Block for the Peak Amplitude

4 Conclusion

In summary, this project aimed to translate the theoretical concepts of QPSK modulation into a practical communication system between two computers using GNU Radio. Using various blocks such as the Constellation Modulator, Costas Loop and Differential Decoder, successfully modulated, transmitted, received and demodulated QPSK signals. The project involved overcoming challenges such as channel distortion, frequency, phase synchronisation, and signal decoding. In addition, an implementation of peak detection using Max and Argmax blocks for correlation, facilitating synchronisation and timing recovery took Place. By implementing signal processing techniques and making effective use of GNU Radio blocks, the Output was to achieve a reliable communication between the two computers. This project not only provided valuable hands-on experience in digital communication systems, but also demonstrated the practical application of theoretical concepts in real-world scenarios.

Sources

[1] Anderson, J.B., Aulin, T. and Sundberg, C.E. (no date) 'Why Digital Transmission', in *Digital Phase Modulation*. Springer Science+Buisness Media, LLC, pp. 1–1. Available at: https://books.google.de/books?hl=de&lr=&id=yf0ICAAAQBAJ&oi=fnd&pg=PA1&dq=phase+modultion&ots=VRJT9sInRo&sig=rsoWc9bfibAy2erQPLN12pqx4Ks&redir_esc=y#v=onepage&q=phase%20modulation&f=false.

[3] *GLFSR source* (no date) *GLFSR Source - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php/GLFSR_Source (Accessed: 05 March 2024).

[4] *Stream mux* (no date) *Stream Mux - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=Stream_Mux (Accessed: 05 March 2024).

[5] *Pack K bits* (no date) *Pack K Bits - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=Pack_K_Bits (Accessed: 05 March 2024).

[6] Prof .Dr.-Ing Uwe Dettmar, . '*Modulationsverfahren*', in Übertragungstechnik. Köln.

[7] Mathuranathan (2023) *QPSK - quadrature phase shift keying*, *GaussianWaves*. Available at: https://www.gaussianwaves.com/2010/10/qpsk-modulation-and-demodulation-2/ (Accessed: 05 March 2024).

[8] *QPSK Mod and Demod QPSK Mod and Demod - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=QPSK_Mod_and_Demod (Accessed: 05 March 2024).

[9] *Constellation rect. object* (no date) *Constellation Rect. Object - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=Constellation_Rect._Object (Accessed: 05 March 2024).

[10] *Constellation Modulator* (no date) *Constellation Modulator - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=Constellation_Modulator (Accessed: 05 March 2024).

[11] *Polyphase Clock sync* (no date) *Polyphase Clock Sync - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=Polyphase_Clock_Sync (Accessed: 05 March 2024).

[12] *Linear Equalizer* (no date) *Linear Equalizer - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=Linear_Equalizer (Accessed: 05 March 2024).

[13] *Costas Loop* (no date) *Costas Loop - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=Costas_Loop (Accessed: 05 March 2024).

[14] *Differential Decoder* (no date) *Differential Decoder - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=Differential_Decoder (Accessed: 05 March 2024).

[15] *Max* (no date) *Max - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=Max (Accessed: 05 March 2024).

[16] *Argmax* (no date) *Argmax - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=Argmax (Accessed: 05 March 2024).

Illustrations Source

Figure 3 : *QPSK modulation-quadrature phase shift keying modulation*. Available at: https://www.rfwireless-world.com/Terminology/QPSK.html (Accessed: 06 March 2024).

Figure 4: *Quadraturphasenumtastung* (2023) *Wikipedia*. Available at: https://de.wikipedia.org/wiki/Quadraturphasenumtastung (Accessed: 06 March 2024).

Figure 9 : *QPSK Mod and Demod* (no date) *QPSK Mod and Demod - GNU Radio*. Available at: https://wiki.gnuradio.org/index.php?title=QPSK_Mod_and_Demod (Accessed: 06 March 2024).

Erklärung

Ich versichere, die von mir vorgelegte Arbeit selbstständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer oder der Verfasserin/des Verfassers selbst entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

Anmerkung: In einigen Studiengängen findet sich die Erklärung unmittelbar hinter dem Deckblatt der Arbeit.

Köln, 06.03.2024 | Gam Biad
Ort, Datum | Unterschrift