Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

GNU Radio-Based Performance Evaluation of Confidence-Based Soft-decision Recovery for Noisy BPSK Communication

Jenifer Darling Rosita Prince Devaraj

j.prince-devaraj@uwtsd.ac.uk

University of Wales Trinity Saint David https://orcid.org/0009-0006-7265-462X

Stalin Jacob Wilson Packia Dhas

University of Wales Trinity Saint David

Research Article

Keywords: GNU Radio, BPSK communication, Channel Model, Binary Slicer, soft-decision recovery, BER analysis, noise voltageKeywords: GNU Radio, BPSK communication, Channel Model, Binary Slicer, soft-decision recovery, BER analysis, noise voltage

Posted Date: July 15th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10344035/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Abstract

Reliable bit recovery is an important requirement in noisy digital communication systems, especially when short binary messages must be recovered accurately under channel distortion. This paper presents a GNU Radio-based performance evaluation of a confidence-based soft-decision recovery method for noisy BPSK communication. The proposed recovery approach uses repeated bit transmission, real-valued received BPSK samples, soft-decision recovery, and confidence estimation to improve message recovery compared with conventional hard-decision Binary Slicer majority voting.

A structured bit frame consisting of a preamble, a 24-bit message, repetition-coded message bits, and padding was transmitted through a GNU Radio Channel Model at different noise voltage levels. Three different 24-bit input messages were tested under multiple noise conditions. The receiver output was evaluated using both hard-decision majority voting and the proposed confidence-based soft-decision method. Experimental results showed that the proposed method reduced bit errors in most tested cases compared with the raw Binary Slicer majority approach. For some tested inputs, the proposed method achieved zero-bit errors at noise voltages where the raw method produced multiple errors.

The results demonstrate that confidence-based soft-decision recovery can improve noisy BPSK bit detection and provide useful reliability information through weak-bit identification. The method is suitable as a low-complexity recovery layer for robust low-data-rate communication experiments and future extensions toward end-to-end deep learning frameworks for robust communication under realistic channel conditions.

1. Introduction

Accurate bit recovery is a fundamental requirement in digital communication systems. During transmission, digital data may be affected by additive channel noise and other impairments, which can cause incorrect bit decisions at the receiver [1], [2]. In noisy digital communication systems, as the received signal becomes weaker or the channel noise increases, the probability of receiver error also increases [1], [2].

Binary Phase Shift Keying (BPSK) is one of the simplest and most reliable digital modulation techniques. In BPSK, binary bits are represented using two opposite signal states, commonly mapped as 0 to -1 and 1 to + 1. Because of its simplicity and robustness, BPSK is widely used as a baseline modulation scheme for studying noisy channel performance and bit recovery techniques [1], [3]. BPSK has also been used in satellite navigation and other robust communication contexts because of its simple two-state modulation structure [3]. However, even with BPSK, high channel noise can cause received symbols to cross the detection threshold, producing bit errors [1], [2].

Conventional hard-decision BPSK receivers are simple because they convert each received sample directly into a binary 0 or 1 [4]. However, this process can discard useful reliability information contained in the received signal amplitude. Repetition coding can improve bit recovery by transmitting each bit

multiple times and applying majority voting, but hard-decision majority voting still depends only on final binary decisions. Therefore, soft-decision recovery is considered in this work because it can use real-valued received signal samples as reliability information before hard slicing [5], [6].

This paper evaluates a confidence-based soft-decision recovery method for noisy BPSK communication. The method uses a structured bit frame with a preamble, repeated message bits, and padding. At the receiver, both the hard-decision Binary Slicer output and the real-valued received BPSK signal are captured. The proposed recovery method applies soft group summation over repeated signal samples to recover the message bits and calculates confidence values to identify weak or uncertain bit positions.

The system is implemented using GNU Radio and Python. GNU Radio is used to model the BPSK transmitter, noisy channel, and receiver, while Python is used to perform soft-decision recovery, confidence estimation, and bit error rate analysis. GNU Radio provides a software-defined radio environment for building and evaluating communication signal processing systems [7]. The performance of the proposed method is compared with raw Binary Slicer majority voting under different noise voltage levels and different 24-bit input messages.

The main contribution of this work is the experimental evaluation of a confidence-based soft-decision repetition recovery method for noisy BPSK bit detection. The study compares the proposed method with hard-decision Binary Slicer majority recovery and analyses its ability to reduce bit errors and identify unreliable recovered bits under different noise conditions.

The objective of this paper is to evaluate a confidence-based soft-decision recovery method for noisy BPSK communication by comparing it with hard-decision Binary Slicer majority recovery under different noise voltage levels and input messages. The evaluation focuses on recovered bit errors, BER, and weak-bit identification using confidence values.

2. Literature Review

Binary Phase Shift Keying (BPSK) is commonly presented in digital communication literature as a basic digital modulation technique for transmitting binary information. BPSK represents binary data using two signal states with opposite phase, making it suitable for analysing noisy channel behaviour and bit error performance. Because of its simple two-state signal structure, BPSK is often used in communication studies to evaluate receiver performance under noisy channel conditions [1], [2]. BPSK has also been used in satellite navigation, where it was one of the first modulation methods used for satellite navigation signals and is still used because of its simple structure [3].

GNU Radio is a free and open-source software development toolkit that provides signal processing blocks for implementing software radios. It can be used with external radio-frequency hardware or without hardware in a simulation-like environment. These features make GNU Radio suitable for building flexible transmitter, channel, and receiver workflows for communication experiments [7]. In this work, GNU Radio is used to model noisy BPSK communication and to compare different bit recovery methods.

In simple BPSK receiver implementations, hard-decision detection can be used to convert received signal values into binary outputs. A Binary Slicer is an example of this type of block because it slices a float value to produce a one-bit output. In the GNU Radio Binary Slicer, positive input values produce binary 1, while negative input values produce binary 0 [4]. Although this approach is simple, it keeps only the final binary decision after slicing. Therefore, reliability information contained in the received signal amplitude is lost, so samples close to the decision threshold and samples far from the threshold may be treated as the same binary value after hard slicing [5].

Repetition coding is one of the simplest error recovery techniques. A repetition code encodes one information bit into an (n)-bit sequence by repeating the logical information several times. Majority decoding can then be used to recover the information, with odd (n) commonly required to avoid ties [8]. In a hard-decision implementation of repetition recovery, majority voting is applied to final binary decisions only. Hard-decision decoding is generally simpler, but it may lose useful channel reliability information compared with soft-decision decoding [9]. Therefore, it does not consider whether each received sample was strong, weak, or close to the decision boundary.

Soft-decision detection addresses this limitation by using reliability information from the received signal value before hard slicing. Instead of treating every received sample as only 0 or 1, soft-decision methods use real-valued channel information to estimate how reliable a received symbol is. For BPSK over an additive white Gaussian noise (AWGN) channel, the channel log-likelihood value is proportional to the received sample value, as shown by $L_{ch(s|y)} = \frac{2}{\sigma^2} y$, where y is the received sample and σ² is the noise variance. This means that received samples with larger magnitude provide stronger evidence about the transmitted symbol, while samples close to zero indicate lower reliability or uncertainty [5]. This principle supports the use of real-valued BPSK samples for confidence-based bit recovery [5], [10].

In many BPSK and bit error rate (BER) performance studies, common evaluation parameters include the modulation scheme, channel model, noise level, receiver decision method, and BER. Conventional communication texts often evaluate BPSK performance under AWGN channel conditions using bit error probability or BER as an important performance measure [1], [2]. In the present work, these common evaluation ideas are adapted into a GNU Radio and Python workflow, where noise voltage, recovered bit errors, and BER are used to compare hard-decision majority recovery with confidence-based soft-decision recovery.

Based on the reviewed literature, hard-decision Binary Slicer detection and repetition-code majority decoding are simple and useful methods, but a hard-decision implementation does not fully use the reliability information available in real-valued received BPSK samples. Soft-decision detection provides a way to use this additional information [5]. Therefore, this paper evaluates a confidence-based soft-decision repetition recovery method that combines repeated bit transmission, soft signal summation, and confidence estimation for noisy BPSK communication.

3. Methodology

3.1 GNU Radio BPSK System Model

The experimental system was implemented using GNU Radio and Python. GNU Radio was used to construct the BPSK transmitter, noisy channel, and receiver flowgraph, while Python was used for post-processing, confidence-based soft-decision recovery, and bit error rate analysis [7].

The transmitter begins with a Vector Source block that provides a known binary input frame. The binary bits are mapped into BPSK symbols, where bit 0 is represented by -1 and bit 1 is represented by + 1 [1]. The BPSK signal is then passed through a GNU Radio Channel Model block to introduce channel noise [11]. Different noise voltage values were used to evaluate the behaviour of the recovery method under noisy conditions.

At the receiver side, two types of outputs were captured for comparison. First, the received signal was passed through a Binary Slicer to produce hard-decision binary values [4]. This output was saved as the raw received bit stream and used as the baseline recovery input. Second, the real-valued received BPSK signal before Binary Slicer detection was saved separately. This pre-slicer signal contains soft information about the received symbols, including the strength and reliability of each sample. The proposed confidence-based recovery method used this soft signal output to support bit recovery, since soft-decision decoding can use reliability information that is not available in final hard 0 or 1 decisions [5],[6].

The GNU Radio flowgraph therefore produced three main output files: the original transmitted bit frame, the hard-decision received bit stream, and the real-valued received signal samples. These files were then processed in Python to compare hard-decision majority recovery with the proposed confidence-based soft-decision recovery method.

3.2 Frame Structure and Repetition Coding

A structured binary frame was used in the experiment to allow controlled transmission and recovery analysis. In communication systems, framing is commonly used to organize transmitted data into identifiable units and support receiver processing [12]. In this experiment, each transmitted frame consisted of a preamble, a 24-bit message, repetition-coded message bits, and padding bits. The preamble was used to mark the beginning of the frame, while the padding bits were added at the end to complete the frame structure used in the GNU Radio experiment.

The frame structure used in this experiment is illustrated in Fig. 1. The original message length was 24 bits, and each message bit was repeated seven times using repetition coding. Therefore, the message section was expanded into 168 repeated message bits. With a 6-bit preamble and 3 padding bits, the complete transmitted frame length became 177 bits. The preamble was fixed as 101010, and the padding bits were fixed as 000.

The frame consists of a 6-bit preamble, 168 repeated message bits, and 3 padding bits. The repeated message section is organized into 24 groups, where each group represents one original message bit

repeated seven times. For example, a message bit of 1 is transmitted as 1111111, while a message bit of 0 is transmitted as 0000000. In the baseline method, each group is decoded using hard-decision majority voting. In the proposed confidence-based soft-decision recovery method, the same groups are recovered using real-valued BPSK samples instead of only hard 0 or 1 decisions.

3.3 Baseline Hard-Decision Recovery

The baseline recovery method used in this experiment was hard-decision repetition decoding. The Binary Slicer output was divided according to the known frame structure, and the preamble and padding sections were ignored during message recovery. The repeated message section was then extracted for decoding.

Since the repetition factor was 7, every seven received binary values represented one original message bit. For each group of seven hard-decision bits, majority voting was applied. If the number of ones in a group was greater than or equal to the number of zeros, the recovered bit was classified as 1; otherwise, it was classified as 0.

The recovered 24-bit message was compared with the original transmitted 24-bit message to calculate the number of bit errors and the bit error rate. This result was used as the baseline for comparison with the proposed confidence-based soft-decision recovery method.

3.4 Confidence-Based Soft-Decision Recovery

The proposed recovery method uses the real-valued received BPSK signal samples before hard-decision slicing. This follows the general soft-decision principle, where the received sample values provide reliability information before they are converted into hard binary decisions [5], [6]. Unlike the baseline Binary Slicer method, which converts each received value directly into either 0 or 1, this work uses the sign and magnitude of grouped received samples as part of a confidence-based repetition recovery process.

After removing the preamble and padding sections, the repeated message portion of the received signal was divided into groups of seven samples. Each group represented one original message bit. Instead of applying majority voting to hard binary values, the proposed method summed the seven real-valued BPSK samples in each group. The sign of this group sum was then used to determine the recovered bit according to the following rule:

- If the group sum was greater than or equal to zero, the recovered bit was classified as 1.
- If the group sum was less than zero, the recovered bit was classified as 0.

After the bit decision, a confidence value was calculated by dividing the absolute group sum by the sum of the absolute values in the group. The confidence value was calculated as:

$$C_i = \frac{\left|\Sigma_{k=1}^{7} r_i, k\right|}{\Sigma_{k=1}^{7} |r_i, k|}$$

where C is the confidence value of the i-th recovered bit, and r, k is the k-th real-valued BPSK sample in the i-th repeated group.

Recovered bits with confidence values below the selected threshold were marked as weak or uncertain. In this experiment, a confidence threshold of 0.35 was used to identify uncertain bit positions. These weak-bit positions provide additional information about which recovered bits may require further checking, correction, or analysis.

3.5 Performance Metrics

The performance of the recovery methods was evaluated by comparing each recovered 24-bit message with the original transmitted message. The baseline hard-decision majority voting method and the proposed confidence-based soft-decision recovery method were compared under different noise voltage levels and input bit patterns.

The first performance measure was the number of bit errors, defined as the number of recovered bits that did not match the corresponding transmitted bits. The second performance measure was the bit error rate (BER), calculated as:

$$BER = \frac{N_e}{N}$$

where N_e is the number of bit errors and N is the total number of transmitted message bits. Since the message length used in this experiment was 24 bits, $N = 24$.

In addition to bit errors and BER, the proposed method produced confidence values for the recovered bits. Bits with confidence values below the selected threshold of 0.35 were marked as weak or uncertain. Therefore, the evaluation considered three outputs: bit errors, BER, and weak-bit positions.

4. Results and Discussion

This section presents the experimental recovery results obtained from the hard-decision Binary Slicer majority method and the proposed confidence-based soft-decision recovery method.

Table 1
Selected performance comparison for Input 3 at 5 V noise voltage.

Performance metric	Raw Binary Slicer + Majority	Confidence-based soft-decision recovery
Recovered message errors	12	5
BER	0.5000	0.2083
Error positions	0, 2, 3, 4, 5, 10, 12, 13, 16, 17, 20, 23	8, 9, 10, 13, 16
Confidence threshold	Not applicable	0.35
Weak/uncertain bit positions	Not applicable	1, 4, 6, 7, 8, 10, 11, 13, 14, 16, 17, 18, 20, 22, 23

For the selected test case, Table 1 shows that the confidence-based soft-decision recovery method reduced the number of bit errors from 12 to 5. The BER decreased from 0.5000 for the raw Binary Slicer majority method to 0.2083 for the proposed method. In addition, the proposed method identified weak or uncertain bit positions using a confidence threshold of 0.35.

Table 2 further summarizes the hard-decision and soft-decision error counts across all tested input messages and noise voltage levels. The results show that the proposed confidence-based soft-decision recovery method reduced the number of recovered bit errors in most test cases compared with the hard-decision Binary Slicer majority method.

Table 2
Error comparison across noise levels.

Noise voltage	Input 1 Hard errors	Input 1 Soft errors	Input 2 Hard errors	Input 2 Soft errors	Input 3 Hard errors	Input 3 Soft errors
0.5 V	0	0	0	0	0	0
1.0 V	6	0	2	0	5	0
2.5 V	9	2	3	0	9	1
3.0 V	9	2	4	1	9	2
3.5 V	9	2	5	2	11	3
4.0 V	10	3	5	2	12	4
5.0 V	12	4	6	3	12	5

Note: Error reduction can be calculated as hard-decision errors minus soft-decision errors.

As shown in Table 2, the confidence-based soft-decision recovery method reduced the number of bit errors in most tested cases compared with the hard-decision Binary Slicer majority method. For Input 1,

the number of errors decreased from 6 to 0 at 1.0 V, from 9 to 2 at 2.5 V, 3.0 V, and 3.5 V, from 10 to 3 at 4.0 V, and from 12 to 4 at 5.0 V. For Input 2, the proposed method achieved zero errors up to 2.5 V and continued to reduce the number of errors at higher noise voltage levels. For Input 3, the proposed method achieved zero errors at 0.5 V and 1.0 V and reduced the error count at all higher tested noise voltage levels. These results show that the proposed method improved recovery performance for the tested input messages and noise conditions.

The BER values calculated from the error counts in Table 2 are presented in Table 3. BER was calculated by dividing the number of recovered bit errors by the 24-bit message length.

Table 3
BER comparison across noise levels.

Noise voltage	Input 1 raw BER	Input 1 soft BER	Input 2 raw BER	Input 2 soft BER	Input 3 raw BER	Input 3 soft BER
0.5 V	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1.0 V	0.2500	0.0000	0.0833	0.0000	0.2083	0.0000
2.5 V	0.3750	0.0833	0.1250	0.0000	0.3750	0.0417
3.0 V	0.3750	0.0833	0.1667	0.0417	0.3750	0.0833
3.5 V	0.3750	0.0833	0.2083	0.0833	0.4583	0.1250
4.0 V	0.4167	0.1250	0.2083	0.0833	0.5000	0.1667
5.0 V	0.5000	0.1667	0.2500	0.1250	0.5000	0.2083

Note: BER is calculated by dividing the number of recovered bit errors by the 24-bit message length.

Overall, the results show that using real-valued BPSK samples and soft-decision recovery improves bit recovery in most tested conditions. The confidence-based method also provides weak-bit information, which is useful for identifying less reliable recovered bit positions under noisy channel conditions.

The BER values in Table 3 are plotted in Fig. 2 for visual comparison.

Figure 2 compares the BER performance of the hard-decision Binary Slicer majority recovery method and the confidence-based soft-decision recovery method for three input messages across different noise voltage levels. The results show that the confidence-based soft-decision recovery method generally produced lower BER values than the raw Binary Slicer majority method. The improvement is especially visible at higher noise levels, where the hard-decision method produced more bit errors. For Input 1, the BER decreased from 0.5000 to 0.1667 at 5.0 V. For Input 2, the BER decreased from 0.2500 to 0.1250 at 5.0 V. For Input 3, the BER decreased from 0.5000 to 0.2083 at 5.0 V. This confirms that using real-valued BPSK samples before hard slicing improves bit recovery under noisy channel conditions.

5. Conclusion

This paper presented a GNU Radio-based performance evaluation of a confidence-based soft-decision recovery method for noisy BPSK communication. The proposed method used repeated bit transmission, real-valued received BPSK samples, soft-decision recovery, and confidence estimation to improve message recovery compared with hard-decision Binary Slicer majority voting.

The experimental results showed that the confidence-based soft-decision method reduced bit errors and BER in most tested noise conditions and input messages. Overall, the reduction in bit errors and BER shows that the proposed method had a positive effect on message recovery, as it reduced many of the errors produced by the hard-decision Binary Slicer majority method under noisy BPSK channel conditions. The method also identified weak or uncertain bit positions using a confidence threshold, which provides useful reliability information for further checking or correction. These results demonstrate that using real-valued signal information before hard slicing can improve bit recovery in noisy low-data-rate communication experiments.

Although the proposed method improved recovery performance in most cases, it did not guarantee error reduction for every individual test condition. Future work may optimize the confidence threshold, test additional channel conditions, and combine the method with stronger error-control coding techniques. This work can also support future robust covert or steganographic communication systems by improving the reliability of the underlying bit recovery stage.

Abbreviations

AWGN
Additive White Gaussian Noise
BER
Bit Error Rate
BPSK
Binary Phase Shift Keying
GNU Radio
GNU Radio software-defined radio toolkit
V
Volt

References

1. J. G. Proakis and Masoud. Salehi, *Digital communications*, 5th ed. McGraw-Hill, (2008)
2. Sklar B (2001) Digital Communications: Fundamentals and Applications, 2nd edn. Prentice Hall, Upper Saddle River, NJ, USA

3. Ávila Rodríguez JÁ (2011) Binary Phase Shift Keying Modulation (BPSK), *European Space Agency*. [Online]. Available: https://gssc.esa.int/navipedia/index.php/Binary_Phase_Shift_Keying_Modulation_(BPSK)
4. Radio GNU Binary Slicer. [Online]. Available: https://wiki.gnuradio.org/index.php/Binary_Slicer
5. Oestreich K, Fellhauer F Soft-Demapping for AWGN Channel: Log-Likelihood Ratios (L-values) for BPSK. [Online]. Available: https://webdemo.inue.uni-stuttgart.de/webdemos/02_lectures/communication_3/soft_demapping/index.php?id=2
6. Huffman WC, Pless V (2003) Fundamentals of Error-Correcting Codes. Cambridge University Press, Cambridge. 10.1017/CBO9780511807077
7. Project GNUR, Radio GNU The Free and Open Software Radio Ecosystem. [Online]. Available: https://www.gnuradio.org/
8. Zoo EC (2023) Repetition code, [Online]. Available: https://errorcorrectionzoo.org/c/repetition
9. Bere PS, Khan MZA, Hanzo L (2023) A Low-Complexity Diversity-Preserving Universal Bit-Flipping Enhanced Hard Decision Decoder for Arbitrary Linear Codes, [Online]. Available: https://arxiv.org/abs/2304.03561
10. Liu R, Wang P, Wang X (2024) Design and simulation of soft decision decoding based on chaotic M-ary spread spectrum system. Discover Appl Sci 6:185. 10.1007/s42452-024-05849-7
11. Radio GNU Channel Model. [Online]. Available: https://wiki.gnuradio.org/index.php/Channel_Model
12. Tanenbaum AS, Wetherall DJ (2011) Computer Networks, 5th edn. Pearson, Boston, MA, USA

Figures

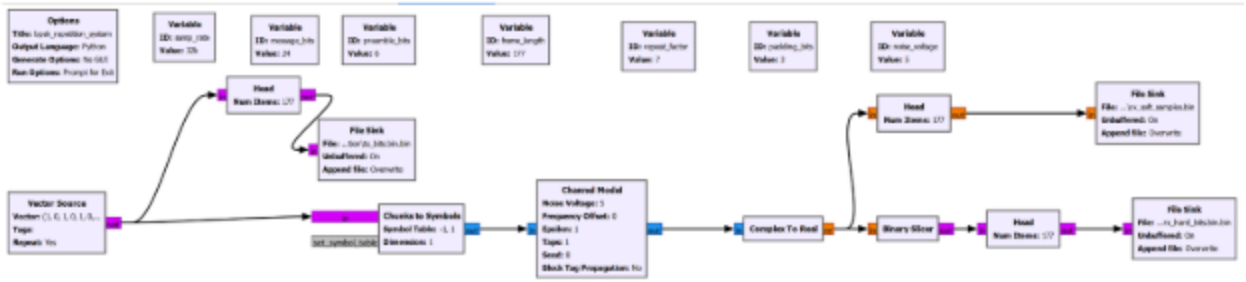


Figure 1

GNU Radio BPSK repetition system flowgraph.

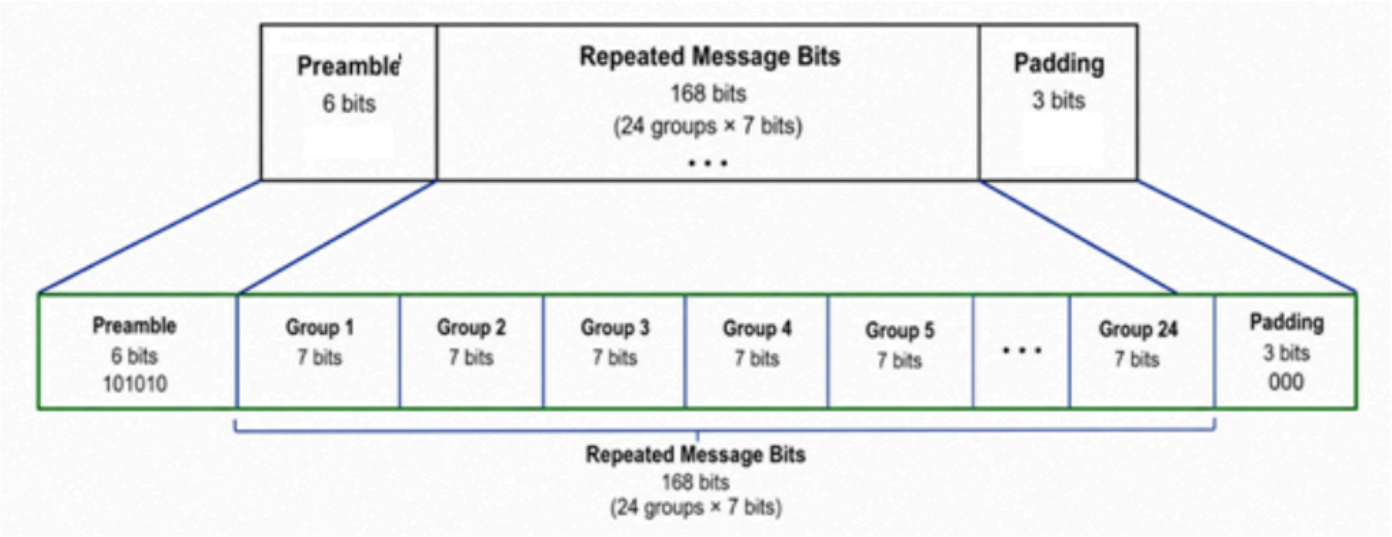


Figure 2

Transmitted BPSK bit-frame structure.

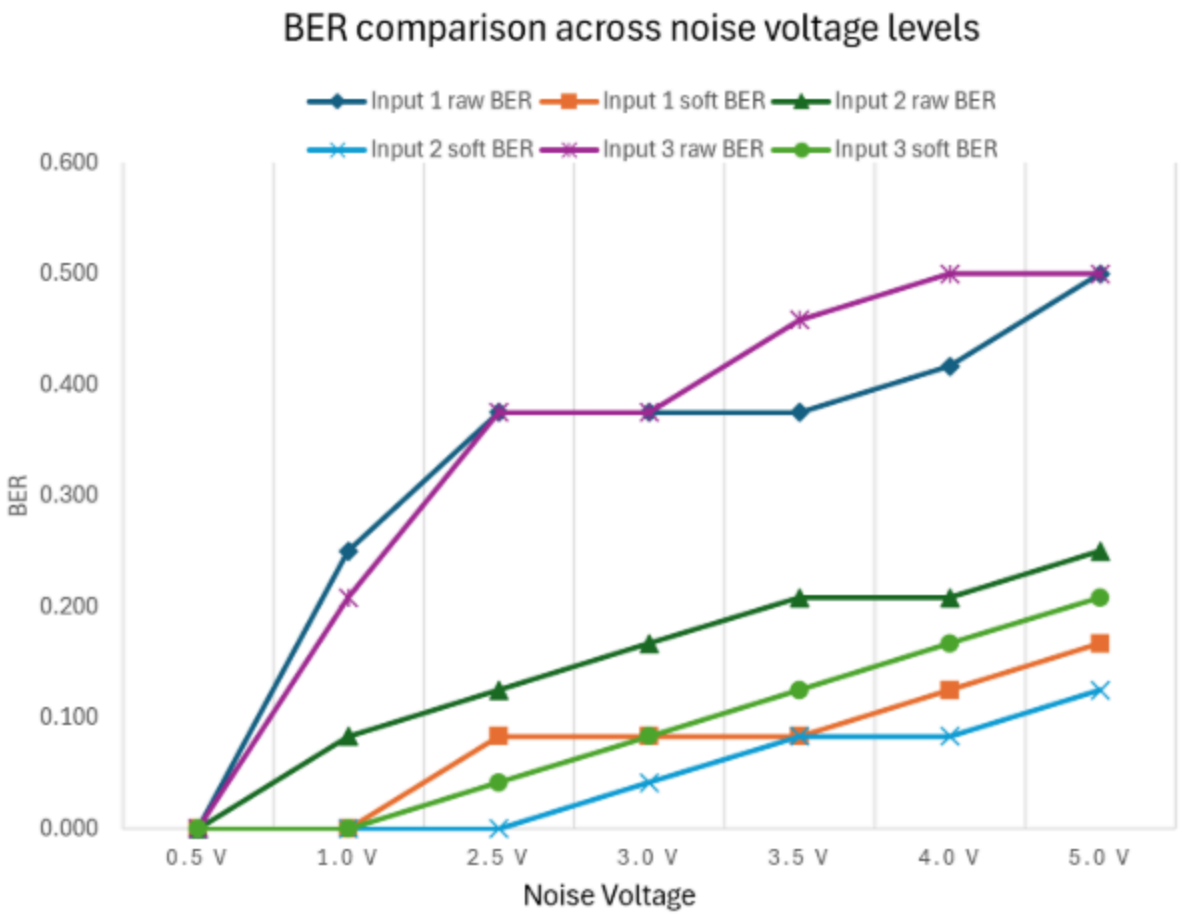


Figure 3

Figure 2. BER comparison across noise voltage levels.