Interactive Simulation Framework for Digital Encoding, Modulation, CRC-Based Error Detection and Sliding Window Protocol Using Python and Streamlit

1st Prof. Supriya Telsang
Dept of Computer Science
Vishwakarma Institute of Technology
Pune,India
supriya.telsang@vit.edu

2nd Shreyas Mulavekar
Dept. of Computer Science
Vishwakarma Institute of Technology
Pune,India
shreyas.mulavekar23@vit.edu

3rd Ritesh Nimbalkar
Dept. of Computer Science
Vishwakarma Institute of Technology
Pune,India
ritesh.nimbalkar231@vit.edu

4th Sachin Prasad
Dept. of Computer Science
Vishwakarma Institute of Technology
Pune,India
sachin.prasad23@vit.edu

5th Palash Poojary
Dept. of Computer Science
Vishwakarma Institute of Technology
Pune,India
palash.poojary23@vit.edu

***Abstract*—Digital communication systems require robust encoding, modulation, and noise resilience to ensure accurate transmission of information across unreliable channels. Understanding the behavior of these techniques in real time is essential for students and practitioners in networking, wireless communication, and embedded systems. However, most educational resources remain theoretical or lack visual interactivity.**

This research presents an interactive simulation framework built using Python, NumPy, Matplotlib, Plotly, and Streamlit to demonstrate real-time signal encoding, modulation, noise simulation, bit error calculation, and communication protocols. The system supports multiple digital encoding methods including NRZ, NRZI, AMI, Manchester, Differential Manchester, and 4B/5B, as well as digital modulation schemes such as ASK, FSK, PSK, and QPSK. The simulator includes AWGN channel modeling, live decoding, BER computation, and visualization.

Furthermore, the framework implements CRC-based error detection utilizing polynomial long division and Sliding Window ARQ protocol, demonstrating packet retransmission, acknowledgment timing, and reliability mechanisms. The platform serves as an effective teaching, experimentation, and demonstration tool for communication engineering concepts.

***Index Terms*—Digital Encoding, Modulation, CRC, BER, Sliding Window, AWGN, Streamlit, Python, Signal Processing, Simulation.**

I. INTRODUCTION

Digital communication forms the backbone of modern computer networks, IoT systems, mobile communication, satellite communication, and data transmission electronics. In such systems, raw digital data must be converted into physical waveforms, transmitted across noisy mediums, and reconstructed accurately at the receiver.

While theoretical models exist in textbooks, actual visualization of how encoding schemes, noise, modulation, and error-handling protocols interact during transmission is limited. Many existing tools require expensive hardware (oscilloscopes, FPGA boards, SDR modules) or paid software (MATLAB, Simulink).

To address this accessibility gap, this work introduces a **fully interactive communication simulator** capable of real-time visualization of encoding, modulation, noise effects, BER tracking, CRC verification, and ARQ retransmission behavior.

The system enables learners to:

- Observe waveform transitions for different encoding and modulation schemes.
- Understand the impact of SNR on recovered signal quality.
- View live CRC error detection and retransmission steps.
- Compare BER performance across encoding and modulation.

This approach supports experiential learning and research prototyping.

II. RELATED WORK

Recent research has focused extensively on evaluating digital modulation, encoding, and channel performance under noisy communication environments. Sharma et al. [1] analyzed the performance of BPSK, QPSK, and QAM under AWGN conditions and demonstrated that lower-order modulation schemes exhibit significantly lower Bit Error Rates (BER) at low Signal-to-Noise Ratios (SNR). Similarly, Alazab et al. [2] investigated PSK schemes under both AWGN and Rayleigh fading models, concluding that fading environments increase error probability and demand additional synchronization and filtering.

Comparative analysis of line coding approaches has also gained attention. Verma and Singh [3] evaluated NRZ, Manchester, and MLT-3 coding techniques for wired communication and reported that Manchester encoding improves synchronization capabilities while MLT-3 offers better bandwidth efficiency. Further studies by Jha and Gupta [6] expanded this comparison to include NRZI, AMI, and Bipolar RZ encoding, highlighting that bipolar and transition-based coding techniques generally provide improved noise resilience and signal integrity.

BER-focused modulation simulations have also been explored. Work by More et al. [4] demonstrated that BPSK consistently outperforms ASK and FSK in noisy channels due to its phase robustness. Kumar et al. [5] supported these findings through comparative modulation experiments, confirming phase-based modulation schemes maintain better BER stability relative to amplitude-based techniques.

Foundational literature in digital communication reinforces these results. Proakis [7] provides mathematical underpinnings showing why coherent modulation schemes demonstrate superior error performance over AWGN channels. Tanenbaum [8] and Forouzan [9] highlight how encoding strategies directly affect synchronization, bandwidth utilization, and channel tolerance. Educational resources such as MIT OpenCourseWare tutorials [10] further emphasize visualization-based learning for modulation and communication channel behavior.

Collectively, these studies indicate that phase-based modulation (PSK/BPSK/QPSK) and transition-based line coding (Manchester, NRZI, AMI) offer improved reliability under noise, while bandwidth-efficient schemes (NRZ, MLT-3, QAM) require additional synchronization mechanisms or filtering to maintain stability. However, despite extensive analytical research, most prior work lacks accessible real-time visualization platforms that demonstrate bit-level behavior, decoding, error introduction, CRC validation, and ARQ retransmission. This gap is addressed by the proposed simulation framework.

III. Methodology

The simulation workflow follows the complete digital communication pipeline shown in Fig. 1.

1) **Bit Input:** User manually enters or uploads a binary bitstream.
2) **Encoding Layer:** Line coding scheme selected (NRZ, NRZI, AMI, Manchester, etc.)
3) **Modulation Layer (optional):** If ASK/FSK/PSK/QPSK selected, encoded binary values map to carrier signal.
4) **Channel Simulation:** Additive White Gaussian Noise (AWGN) applied based on adjustable SNR.
5) **Decoding Layer:** Threshold decision or carrier phase demodulation extracts received bits.
6) **Error Detection:** CRC remainder validation detects corruption.**
7) **Error Recovery:** Sliding Window ARQ retransmits missing or erroneous packets.**
8) **Performance Metrics:** BER, retransmission count, and latency displayed.**

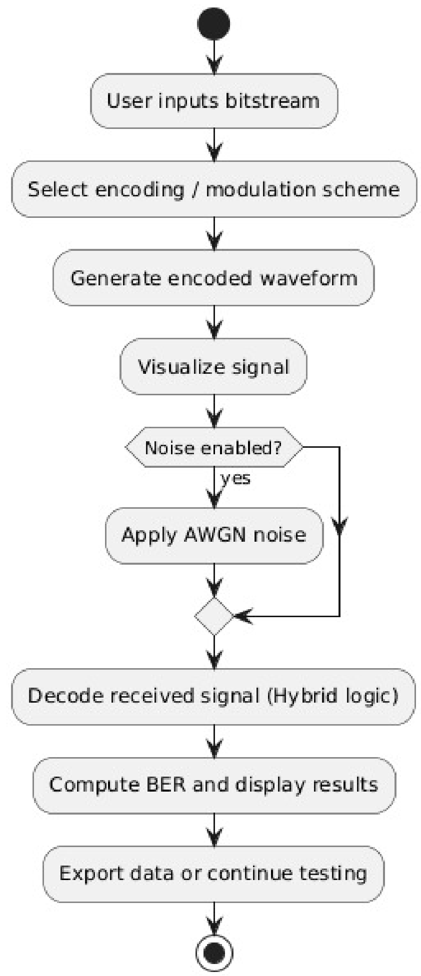

Fig. 1. Methodology Process Flow Diagram

IV. System Architecture

The system architecture shown in Fig. 2 consists of four main layers:

A. User Interface Layer

Built using Streamlit supporting real-time sliders, buttons, waveform controls, and live playback.

B. Signal Processing Engine

Implements encoding, modulation, filtering, and sampling using NumPy/Matplotlib.

C. Communication Core

Includes CRC logic, AWGN channel, decision thresholds, demodulation, BER calculation.

D. Protocol and Control Layer

Implements sliding window ARQ, acknowledgment logic, timeout processing, and retransmission.

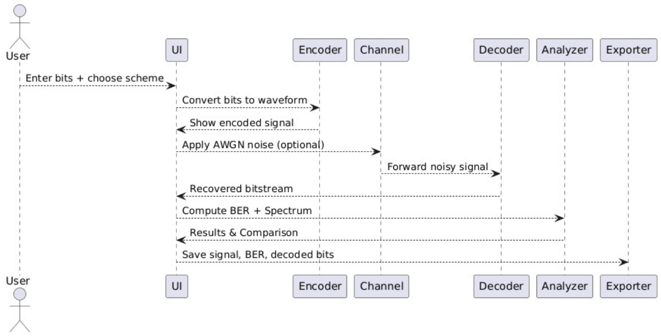

Fig. 2. System Architecture Overview

V. Mathematical Modeling

Digital encoding and modulation methods are represented mathematically below.

A. Line Encoding

Let $b_i \in \{0,1\}$ be the bitstream.

NRZ:

$$s(t) = \begin{cases} +1 & b_i = 1 \\ 0 & b_i = 0 \end{cases}$$

NRZI:

$$s_i = \begin{cases} s_{i-1}, & b_i = 0 \\ -s_{i-1}, & b_i = 1 \end{cases}$$

Manchester:

$$s(t) = \begin{cases} +1 \to -1, & b_i = 1 \\ -1 \to +1, & b_i = 0 \end{cases}$$

AMI:

$$s(t) = \begin{cases} 0, & b_i = 0 \\ (-1)^k, & b_i = 1 \end{cases}$$

B. Digital Modulation

Let A be amplitude and f_c carrier frequency.

ASK:

$$s(t) = b_i A \sin(2\pi f_c t)$$

FSK:

$$s(t) = \begin{cases} A\sin(2\pi f_1 t), & b_i = 1 \\ A\sin(2\pi f_0 t), & b_i = 0 \end{cases}$$

PSK:

$$s(t) = A\cos(2\pi f_c t + \pi b_i)$$

QPSK:

$$s(t) = A\cos\left(2\pi f_c t + \frac{\pi}{2}(2b_{2i} + b_{2i+1})\right)$$

C. Noise and BER

$$r(t) = s(t) + n(t), \quad n(t) \sim \mathcal{N}(0, \sigma^2)$$

$$BER = \frac{\text{bit errors}}{\text{total bits}}$$

D. CRC Polynomial Division

$$R(x) = M(x)x^k \mod G(x), \quad F(x) = M(x)x^k + R(x)$$

VI. Results and Observations

The simulator produced multiple visual and numerical outputs.

Fig. 3. Noisy vs. Clean Signal Representation

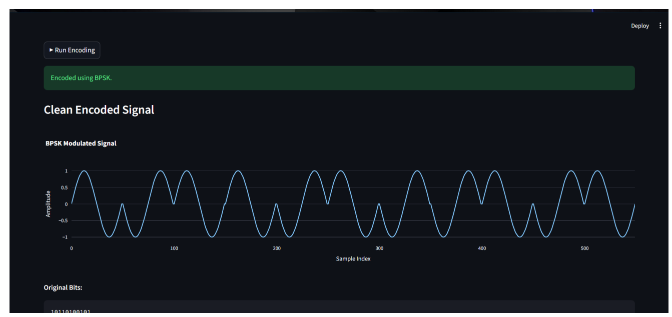

Fig. 4. PSK Signal Output Visualization

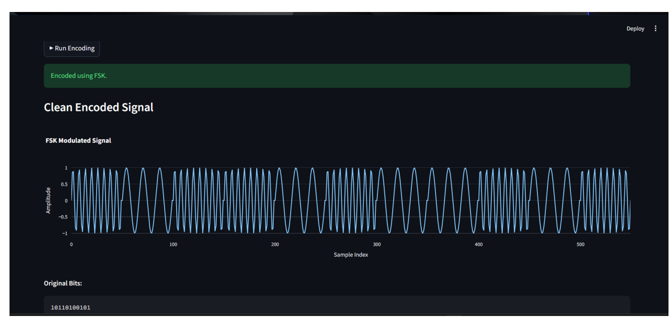

Fig. 5. FSK Signal Output Visualization

Key observations:

- Phase-based schemes (PSK/QPSK) outperform ASK under high noise.
- Manchester resulted in lower error propagation due to synchronization property.
- CRC detected all corrupted frames and triggered ARQ retransmission.
- Sliding Window significantly reduced latency compared to stop-and-wait transmission.

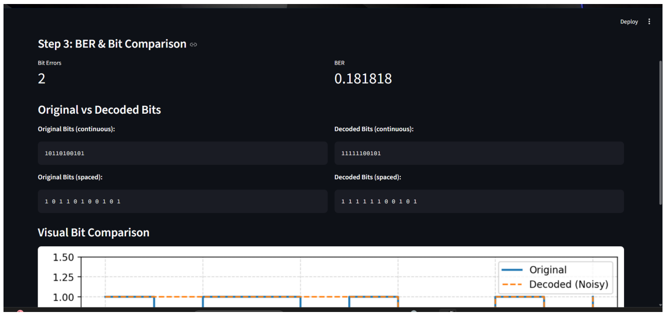

Fig. 6. BER vs. Scheme Comparison

VII. Conclusion

The developed simulation system successfully demonstrates and animates digital encoding, modulation, noise modeling, decoding, CRC calculation, and retransmission using sliding window. The interactive environment helps visualize theoretical concepts in an intuitive and engaging manner.

VIII. Future Enhancements

- Support for OFDM, DSSS, and adaptive modulation.
- Machine learning noise prediction and dynamic EQ.
- Export as standalone EXE for offline classroom use.
- 5G PHY framing and TCP packet streaming simulation.

Acknowledgment

The author thanks mentors, faculty and peers for guidance.

References

[1] S. Sharma et al., "Performance Analysis of BPSK, QPSK & QAM over AWGN Channel," International Journal of Communication Systems, 2022.
[2] M. Alazab et al., "Digital Modulation Simulation through AWGN Channel and Rayleigh Fading using PSK," IEEE Access, 2022.
[3] P. Verma and R. Singh, "Comparative Study of Line Coding Techniques for Wired Communication," International Journal of Emerging Technologies, 2019.
[4] S. More et al., "BER Performance of ASK, FSK & BPSK in AWGN Channel," International Journal of Electronics and Communication, 2024.
[5] R. Kumar et al., "Modulation Comparison for Wireless Digital Systems under AWGN," Proc. Intl. Conf. on Wireless Networks, 2021.
[6] A. Jha and M. Gupta, "Performance Comparison of Line Coding Techniques for Digital Data Transmission," Journal of Communication Engineering, 2020.
[7] J. Proakis, *Digital Communications*, McGraw-Hill, 2007.
[8] A. Tanenbaum, *Computer Networks*, Pearson, 2020.
[9] B. Forouzan, *Data Communications and Networking*, McGraw-Hill, 2017.
[10] MIT OpenCourseWare, "Digital Modulation Tutorials." Available: https://ocw.mit.edu