ISSN 1840-4855
e-ISSN 2233-0046

Original scientific article
http://dx.doi.org/10.70102/afts.2026.1835.732

ACAPE-FID ADAPTIVE CONVOLUTION-ASSISTED POLAR ENCODER WITH FLEXIBLE ITERATIVE DECODING FOR HIGH-EFFICIENCY FPGA WIRELESS COMMUNICATION

T. Ranjitha Devi[1], Dr.C. Kamalanathan[2*]

[1]Research Scholar, Electrical Electronics and Communication Engineering, GITAM Deemed to be University, Bengaluru, Karnataka, India; Assistant Professor, Electronics and Communication Engineering, G.Pullaiah College of Engineering and Technology, Jawaharlal Nehru Technological University Anantapur, Kurnool, Andhra Pradesh, India.
e-mail: rthammis@gitam.in,
orcid: https://orcid.org/0009-0004-5248-7153
[2]Associate Professor, Electrical Electronics and Communication Engineering, GITAM Deemed to be University, Bengaluru, Karnataka, India. e-mail: kchandra@gitam.edu,*
Orcid: https://orcid.org/0000-0003-1579-5670

Received: January 27, 2026; Revised: March 09, 2026; Accepted: April 29, 2026; Published: May 29, 2026

SUMMARY

Reliable wireless communication needs a highly efficient Forward Error Correction (FEC) technique in order to counter the effects of noise, interference, and losses. Most existing FEC techniques add too much redundancy and create extra latency, thereby reducing the efficiency of bandwidth utilization. Hence, the purpose of the current research is to design a new model, called Adaptive Convolution-Assisted Polar Encoder with Flexible Iterative Decoding (ACAPE-FID). This hybrid FEC method uses a combination of Adaptive Frozen Polar Coding, Convolutional Encoding, and Flexible Iterative Decoding using Reed-Solomon Euclidean. ACAPE-FID adapts the number of frozen bits based on the number of input bits, so there is no unnecessary redundancy in the system, and it uses sequential Turbo-based iterative decoding to achieve correct error correction in case of congestion in channels. The developed algorithm was designed and implemented using MATLAB HDL Coder on Xilinx Zynq-7000 FPGA, and its performance evaluation was conducted to show that ACAPE-FID had superior results in terms of BER=10^{-9}, BLER=10^{-7}, FER=0.001, latency=0.01 μs, loss=1%, efficiency=0.94%, SNR=3.1 dB, PSNR=33, and throughput=12 Mb/s compared to traditional Polar, LDPC, TBCC, and Turbo coding schemes.

Key words: *forward error correction, convolution codes, polar codes, euclidean algorithm, reed-solomon code, turbo coding schemes.*

INTRODUCTION

FEC is a crucial technique used in wireless communication systems to enhance the precision and consistency of data delivery [1]. It gets over the challenges presented by the unreliable and loud wireless channels by adding redundancy to the broadcast data, which also makes it possible for the receiver to detect and resolve issues. In wireless communication, data is transmitted over wireless channels, which are susceptible to a variety of losses, such as fading, noise, interference, and signal attenuation [2]. These

vulnerabilities introduce errors into the data that is being sent, which lowers performance and reliability. FEC systems make an effort to prevent these errors by introducing error-correcting codes at the transmitter. By adding redundant information to the data, these codes allow the receiver to restore the original data even in the event of errors. By increasing redundancy, errors are detected and corrected at the receiver, eliminating the need for retransmission, according to the FEC guiding concept [3]. Using an error-correcting code to encode the data at the transmitter and produce extra redundant bits is the FEC technique. Together with the encrypted data, the unnecessary bits are also sent via the wireless channel. Error detection and repair are made possible at the receiver by using the same error correction algorithm to decode the received data [4].

The use of FEC in wireless communication systems has several benefits, mainly the enhancement of data reliability during transmission by minimizing errors and compensating for the effects of the channel [5]. In cases where poor signal reception causes challenges, the use of FEC guarantees that the original information can still be read. This method is highly applicable when it becomes impossible to retransmit the data because of lack of time, energy, or other important factors [6]. There are different methods used for FEC in wireless communications, namely Polar Codes, Reed-Solomon, Low-Density Parity Check, and Convolutional Codes. Every coding scheme features a unique set of efficiencies, effectiveness, and complexities [7]. FEC is incorporated into many wireless communication schemes, such as broadcasting services, Wi-Fi, 3G/4G/5G communications, satellite communication, and wireless sensor networks [8].

Retransmission helps overcome the limitations and challenges caused by defective wireless channels, which is necessary for achieving reliable communication in wireless networks. By detecting and fixing lost or broken packets, retransmission improves overall data transport reliability and ensures that the destination receives the proper data [9]. On the other hand, excessive retransmissions may cause further delays and decreased system throughput. Retransmission process optimization is essential for striking a balance between reliability and efficiency in wireless communication systems. Many communication protocols, such as the Automatic Repeat request (ARQ) protocol, require retransmission when the recipient fails to respond with a positive acknowledgment (ACK) or a negative acknowledgment (NAK) [10]. Error correction systems need to be adaptable enough to change course in the event of unstable channel conditions in order to guarantee reliable communication. Adapting decoding settings and algorithms in real-time to account for channel variability is challenging. Errors can also be produced by multipath propagation or interference from neighboring signals, which makes error-correcting systems less effective [11]. Interference must be handled with sophisticated decoding techniques that can accurately distinguish between multipath or interference components and intended signal components. Finding a balance between error correction capability and processing efficiency is challenging since more powerful error-correcting techniques may require large amounts of resources.

FEC finds its application in high-speed communications, which operate at a very fast rate. The management of high-speed data links becomes very difficult in the case of FEC systems based on FPGA (Field Programmable Gate Array) technology, if proper consideration is not provided for clock domains, data serialization/ deserialization, and synchronization processes. As far as FPGA implementation is concerned, the designs tend to consume much energy when compared to those of the ASIC (application specific integrated circuits) [13]. In order to ensure that power consumption becomes optimum in FPGA-based FEC designs, proper power optimization strategies have to be considered. These include reducing redundant computations, leveraging low-power features, and implementing power gating systems. Verification of the correctness and performance of FEC systems in FPGAs can become problematic due to the concurrent nature of FPGA designs [14]. The development of proper test benches and simulation execution to verify the designs under various failure conditions can be a cumbersome process [15]. Wireless communication systems face a lot of difficulties in ensuring data transmission without any loss of information. Many research activities are underway, yet there are lots of improvements required in the systems. Thus, to overcome the challenges with the current methodologies, a novel way of FEC-based wireless communication is required. The following are this paper's main contributions:

- In order to minimize the extra information that will be injected into the input, the Polarized Convolutional Encoder is used as it ensures effective utilization of the data and efficient use of available bandwidth.
- In order to avoid any loss in data packets and to increase the efficiency of data transmission, a new Flexible Turbo Decoder has been introduced. This decoder not only locates but also corrects errors through an iterative procedure.

The proposed study is organized in the following sections. Section I discusses the introduction, while Section II describes the literature survey. Section III details the proposed methodology and also discusses various methodologies applied in this particular research work. Performance analysis of the proposed work has been discussed in Section IV. Section V concludes the complete work.

LITERATURE SURVEY

Another interesting TEC scheme was proposed by Vinodhini et al. [16] specifically for the robust and low power consumption data link layer of Network on Chip (NoC). This scheme managed to correct the errors with a minimal amount of hardware resources utilized. The authors relied on simulation results for validating their TEC concept and conducted tests based on practical traffic flows to assess its performance. In turn, due to the smaller number of residual errors obtained by the authors' scheme compared to the Hamming product code, it allowed for efficient data transmission even in low voltage regimes. Furthermore, this technique resulted in power savings in NoC links of up to 71% compared to the Hamming product code; however, as a consequence, it resulted in additional delays for the codec and the router.

The fundamental hardware structure of the GRAND-MO algorithm has been stated by Syed Mohsin Abbas et al [17]. The GRAND-MO algorithm represents an improved version of the GRAND algorithm, where the decoding of linear block codes through memory channels has been carried out. Several improvements have been made to the algorithm in order to facilitate its hardware realization and decoding ease. According to the simulation results from the ASIC synthesis approach, the average information throughput attained for a code length of 128 bits and FER objective of 10^{-5} was around 52 Gbps along with a 3 dB enhancement in the decoding efficiency compared to that of the GRANDAB (AB=3) algorithm. Moreover, it gave 2 dB enhancement in terms of decoding efficiency and 33% improvement in the worst-case throughput compared to the (79,64) BCH decoder.

Matrix-based error correction and parity sharing, which uses just 14 parity bits to conveniently find out and correct neighboring faults, has been proposed by Konda Nandan Kumar et al. [18]. Using the proposed technique, up to four bits and some two-bit errors can be corrected. The computation of bits M8 and M9 in this technique is different and based on the bits present in the DNA-curve in the matrix. The proposed technique reduces the power, area, and latency considerably. The performance analysis carried out on the Vertex-6 indicated that there was a reduction of 1.26% in power, 5.55% in area, and 21.17% in time delay. From the above, it can be observed that the proposed code was suitable for memories but some errors persisted.

Research conducted by Qilin Zhang et al. [19] proposed a decoding method for CRC codes using deep learning and SPA with data-driven approaches. This decoding process incorporated internal parameter optimization training using learning procedures. In relation to the way soft-decision decoding operates, a new loss function based on negentropy, which is a crucial metric for evaluating the nearness of a probability distribution to a Gaussian distribution, and binary cross-entropy (BCE) was proposed. As per the findings of the study, the decoding method employing deep unfolding and the new loss function incorporating negentropy was successful in improving BER performance. However, the method failed to ensure the interoperability of different systems.

On the other hand, Linfang Wang et al. [20] employed CRC for supporting PAS in TCM in meeting the RCU condition. The messages were first encoded by the transmitter's distribution matcher to create symbols with the right distribution. The binary outputs from the distribution matcher were then encrypted using CRC. After that, the CRC-encrypted bits were encoded and modulated using Unger

Boeck's TCM scheme, which combines a systematic tail-biting convolutional code with a mapping function that converts the coded bits into signals that can reach the maximum channel capacity. However, there was a balance to strike between the amount of redundancy and the error correction performance.

The research conducted by Svitlana Matsenko et al [21] on indivisible codes for BSC was done through the average probability method, an approach that is simple, flexible, and dependable with estimations that can closely resemble the practical environment.

This study also involved the implementation of fractal decoding hardware developed with FPGA software, where multilayer pulse amplitude modulation (PAM-M) with grey code mapping was implemented for communication with fiber-optical media in short distances. Using the average probability method, indivisible codes were analyzed according to standards of exact transmission and error detection of both visible and concealed forms. Indivisible codes assist in the management of data end-to-end through a uniform structure and high-speed transmission in digital machines and apparatuses. Nonetheless, this mode of error control could not effectively solve burst errors, and code mapping should be improved.

Another JRC technique, namely spectrum-spreading phase-coding-based JRC using photonic technologies operating at the millimeter wave band, was proposed in Bai et al. [22]. In particular, spectrum-spreading multiplexing and photonic microwave phase-coding approaches were used to develop the system capable of producing the wideband mm Wave JRC signal with a communication data rate exceeding 1 Gb/s and high-resolution radar ranging capabilities with the range resolution better than 3.5 cm. Due to the separation of communication and radar signals in terms of the frequency domain, spectrum-spreading multiplexing was used to minimize the interference between the communication and radar parts of the system. At the same time, the technique increased the capability of the communication system to resist interference/noise and improved the PSR for radar applications. The developed technique experienced the problem associated with the nonlinearities of PA operation.

A technique using the H-Matrix as a tool for detecting and correcting several occurrences of double adjacent faults was suggested by Bhargavi et al. [23]. For 32 bits of input data, it was able to rectify up to eight erroneous bits. Double adjacent fault errors are those errors where each burst contains not more than four erroneous bits. Three codes using the H-Matrix with different parity bits have been formulated and evaluated based on several criteria. In order to make the design of encoders and decoders easier, the H-Matrix had fewer one's and minimal redundancy. Nonetheless, the technique only considered errors arising from double adjacent faults up to eight bits.

Prathyusha et al. [24] presented a matrix-based model for error detection and correction within memory systems using power and stoppage for focusing on equality bits. The memory protection technique used by Prathyusha et al. [24] was successful, and its performance was optimal. In particular, this method would prove helpful where there was limited speed and parity bits. The address control unit had a row decoder and a column decoder that decoded data addresses, respectively. Both reading and writing activities were executed by the address control unit. However, high complexity led to high power consumption and slower speeds.

Wang et al. [25] put forth deep error-correcting output codes (DeepECOCs), a technique for deep learning that integrates concepts of deep learning, online learning, and ensemble learning. The connections between two network layers could be viewed as incremental SVMs, while each ECOC module would be considered a layer in the network. Supervisory information, including class labels, was provided during the pretraining phase of the network. Their model proved efficient and successful, particularly when considering large-scale applications. However, it does not provide enough scalability and computational efficiency.

On balance, it appears that the literature reveals that Vinodhini et al. [16] had an imbalance between error correction and bandwidth efficiency; Abbas et al. [17] performed poorly on various channels; Kumar et al.[18] could not reduce the residual error; Zhang et al. [19] could not ensure interoperability;

Wang et al. [20] revealed a trade-off between redundancy and error correction; and Matsenko et al. [21] could not manage burst errors, while the code mapping approach required further improvements.

Bai et al., [22] identifies the source of nonlinear distortion as the radar's power amplifier (PA), while Bhargavi et al. [23] manages errors arising only up to 8 bits through double adjacent errors. Prathyusha et al. [24] identifies complexities as sources of high-power usage and increased delay, and Wan et al. [25] cannot address issues of scalability and computation efficiency. As such, a new strategy needs to be adopted for dealing with these weaknesses and improving the efficiency of wireless communication.

ADAPTIVE CONVOLUTION ASSISTED POLAR ENCODER WITH FLEXIBLE ITERATIVE DECODING

The additional number of bits introduced due to the coding scheme and the block length impacts the efficiency of FEC in dealing with errors. The current FEC schemes consume a larger amount of time and power resources, reducing the data rates and causing delays. In order to overcome these challenges and enhance the efficiency of communications in wireless channels, ACAPE-FID is proposed. The existing approaches apply large code blocks for enhanced correction of errors and continuous communication flow in wireless devices; however, they waste unnecessary bits and reduce the amount of effective bandwidth. Therefore, a Polarized Convolutional Encoder is introduced. The first step in this new encoder involves the process of Adaptive Frozen Polar Coding. During polarization, encoding is done by adding original data bits and other frozen bits according to the data length without having to employ unnecessarily lengthy blocks. After polarization, the transformed bits are subject to convolution codes. This coding process employs generator polynomials that regulate the output of bits and provide a form of redundancy that helps detect and repair any errors in the received data. As a result of this new encoder, the number of redundancies found in the data will be minimized.

Moreover, in times of congestion in the network, the dropping of packets leads to the loss of data in the receiver. The current method used to detect such an error involves a cyclic redundancy check (CRC). CRC finds it hard to cope with congested network systems, resulting in data loss and errors. CRC sometimes comes up with identical results for various data packets. This leads to data collision at the receiver end since the error correction process involving lower degree polynomials ends up misidentifying errorless data packets. In order to solve this problem, a Flexible Turbo Decoder will be presented in this paper. This decoder uses Turbo codes and a revised version of the Reed-Solomon (RS) codes. After receiving the data, the Reed-Solomon Euclid (RSE) Code can detect errors in the data. In the decoding process, the data obtained is first decoded to generate syndromes by employing the generator polynomial for the RS code, and subsequently locating the error positions using the Euclidean algorithm. In the correction process, the errors are corrected by Sequential Concatenated Turbo Coding. This scheme is iterative in nature with respect to the previous scheme (RSE). This will result in the elimination of packet losses during the transmission of data, especially on congested networks. Finally, adaptive polar codes using the reversed polarization transform are used.

The figure 1 depicts the block diagram of the developed ACAPE-FID. The architecture consists of three major components, namely, the sender side, the communication medium, and the receiver side. At the sender side, the data is fed into the Polarized Convolutional Encoder. In this case, the Polarized Convolutional Encoder is characterized by two techniques, namely, Adaptive Polar Coding and Convolution Coding. These two coding schemes help in removing the redundant information that is attached to the input data, thus improving the data rates and bandwidth efficiency. From the Polarized Convolutional Encoder, the encoded information passes to the communication channel up to the Flexible Turbo Decoder on the receiver side, where errors can be detected and corrected. In this case, the RSE code is used in the detection of errors, while the Sequential Concatenated Turbo codes are used in the correction of errors.

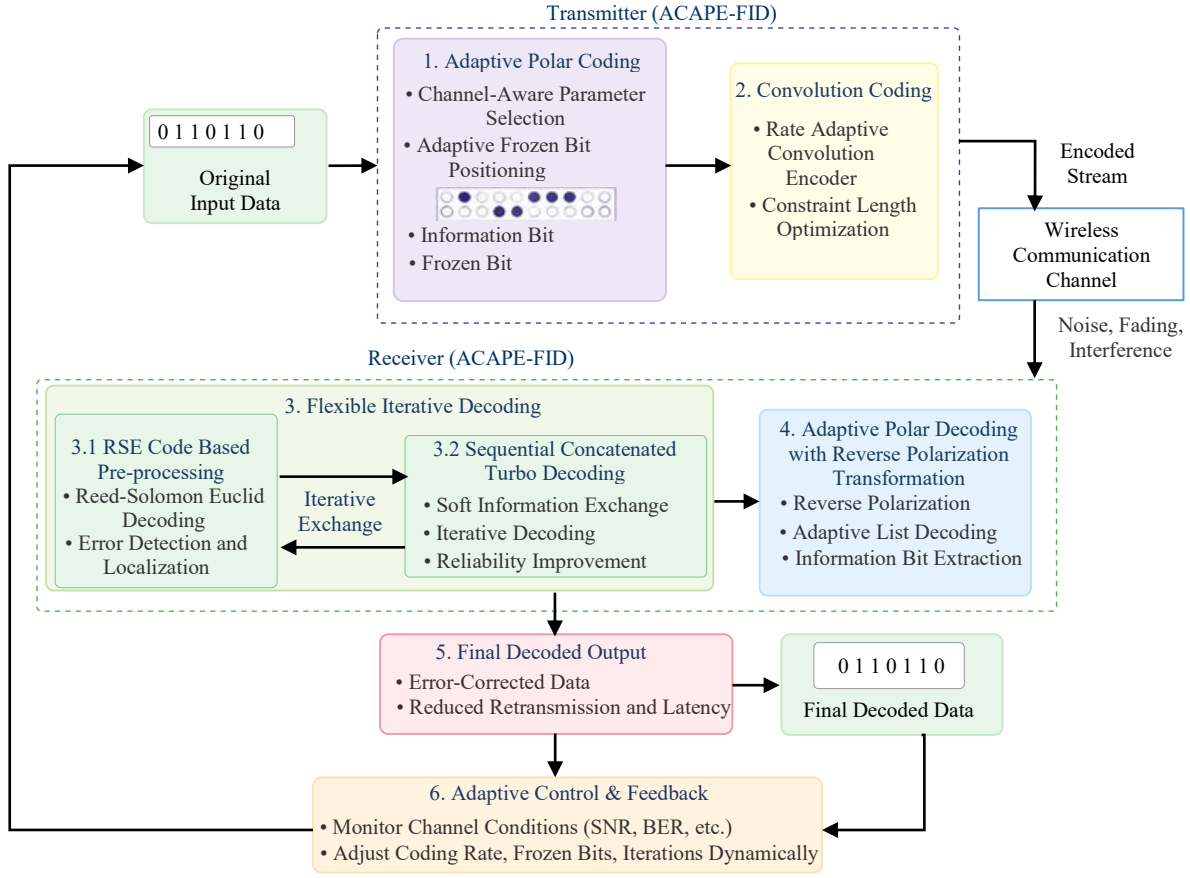


Figure 1. ACAPE-FID model showing transmitter, channel, receiver, and adaptive feedback

Polarized Convolutional Encoder

In wireless communications, one technique called Forward Error Correction (FEC) provides an additional layer of reliability by including extra information within the data transmitted. In doing so, the receiver can correct any errors in the received data without having to request retransmission of the data from the sender. Such a technique becomes highly relevant in addressing problems such as interference and noise in the wireless channel. Initially, the input data is fed into the Polarized Convolutional Encoder, consisting of Adaptive Frozen Polar Coding and convolutional coding techniques. Such coding schemes play the role of reducing excessive redundancy of the input data.

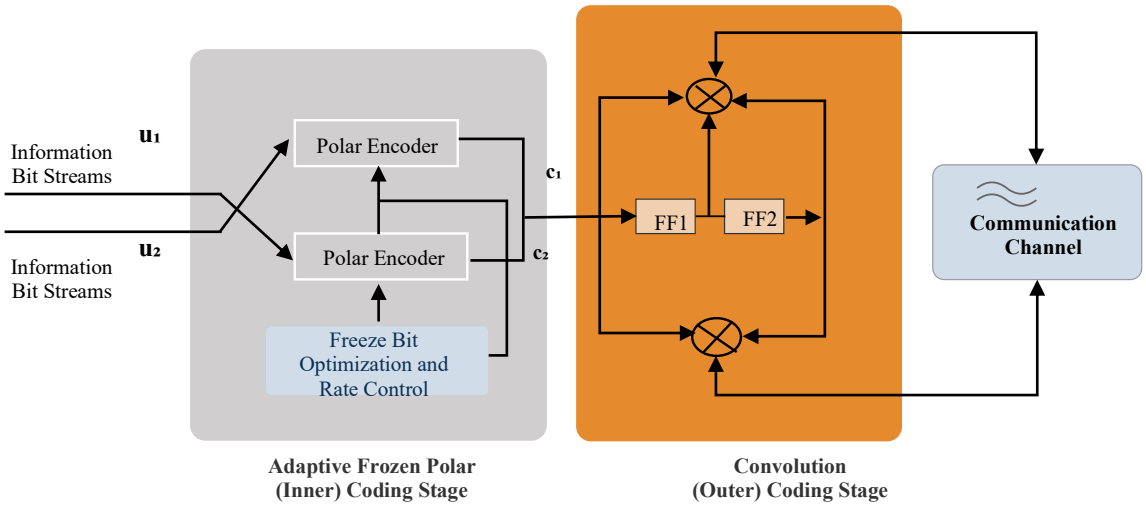


Figure 2. Polarized convolutional encoder architecture

The structure of the Polarized Convolutional Encoder of the ACAPE-FID method is illustrated in figure 2. In the process of transmitting data, an output from the polar encoder is employed for generating a signal in the convolutional encoder. This signal is transmitted via a communication channel and decoded to retrieve the initial information at the receiving end.

Firstly, Adaptive Frozen Polar Coding is used to encode the original data. The input information is coded using polar coding to create polarized bits of information that can either be considered "frozen" or "information". With the use of frozen bits and by turning on/off particular bits within the original input sequence of bits, a polarized sequence of bits is created through the polarization stage. The novelty of the polarization process lies in its ability to dynamically determine how many frozen bits are inserted within the polarization stage based on the size of the original information sequence. This is contrary to static approaches that rely on the same quantity of frozen bits for each original sequence (Equation 1).

$$x = u \times F + f \qquad (1)$$

In the above equation, u is the original input bits, x is the encoded bits after polarization, f is the frozen bits, and F is the matrix that polarizes the original bits. Through this process, there is no need for excessive block sizes, thus making encoding efficient for every particular set of data. After the bits are polarized, convolutional encoding is done on the polarized bits to create encoded bits with redundant information. The generator polynomials help create some redundant information, control the output bit stream size, and also control the degree of redundancy. This convolutional encoding algorithm is explained below using Equation (2).

$$c = x' \times G \qquad (2)$$

In this case, x' are the bits that have been encoded by means of the polar coding technique, while c are the bits after being encoded through the process of convolutional coding, and G stands for the generator polynomial matrix. It is preferable to use convolutional codes in wireless communication since they have good error correction capabilities, even when noise exists in the channel. The whole process of coding through the convolutional and polar coding technique is given in equation (3).

$$c = (u \times F + f) \times G \qquad (3)$$

In consequence, the Polarized Convolutional Encoder reduces unnecessary redundancy to the inputted data. The reduction in the number of unnecessary information contributes to proper utilization of the data and better efficiency in using the available bandwidth. Since unnecessary redundancy will not be included in the data stream, the performance of the communication system becomes much better. Encoded data then goes to the receiver side where it is decoded by the Flexible Turbo Decoder, which is discussed in 3.2.

Flexible Turbo Decoder

Flexible Turbo Decoder is a decoder that is used in wireless communication systems for implementing forward error correction (FEC). Flexi Turbo Decoder uses modified Reed-Solomon (RS) code and turbo code decoding algorithms for FEC. With one design, the decoder is able to work on multiple FEC algorithms such as Turbo, LDPC, and Polar codes.

In figure 3 displays the Architecture of the turbo decoder of the ACAPE-FID, which utilizes the concept of iterative decoding, which is one of the components of Turbo Decoding. Iterative decoding uses two decoders that repeatedly exchange information with each other through a number of cycles. Both decoders use the output produced by another decoder and the received signal to improve the estimate of the transmitted data. In this way, the accuracy of decoding increases with each new iteration.

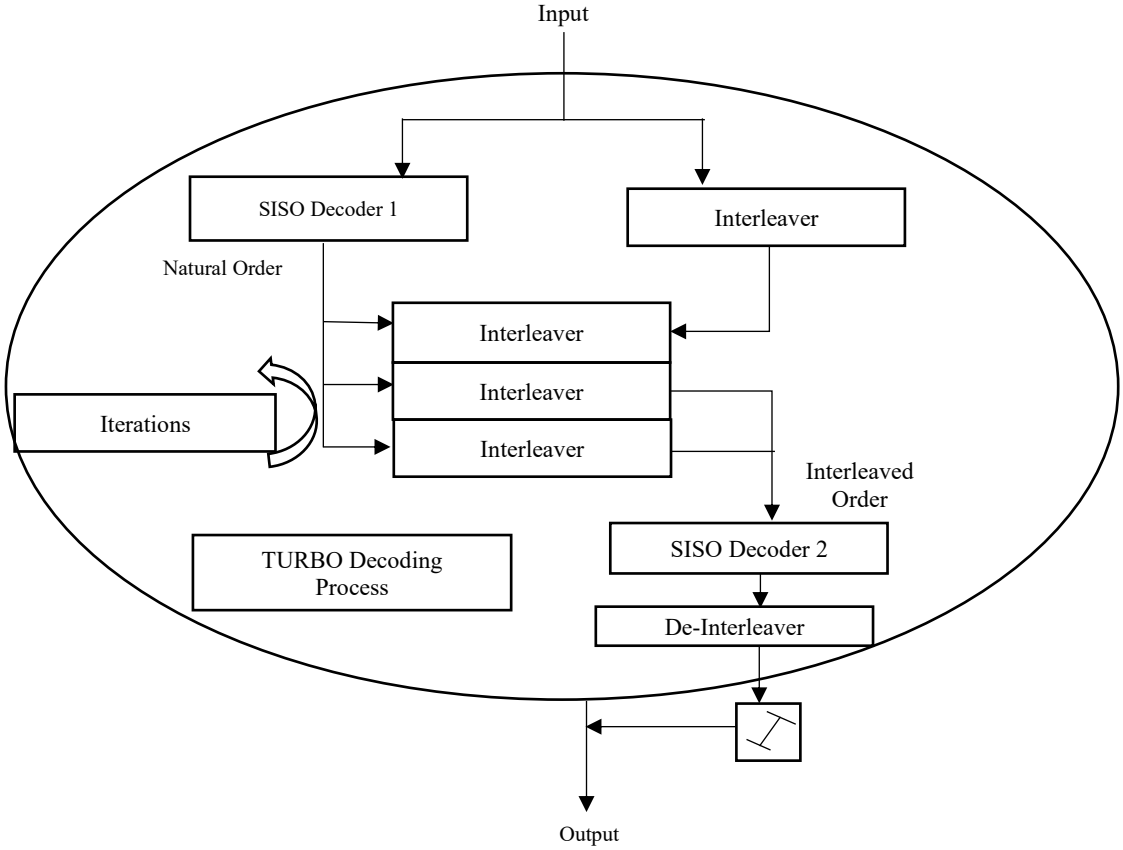


Figure 3. Architecture of the turbo decoder

At the receiving end, the data is decoded using the RSE code. The RSE code uses the generator polynomial of the Reed-Solomon (RS) code for syndrome calculation. The RS codeword $c(x)$ can be mathematically expressed as shown in Equation (4).

$$c(x) = g(x) \times m(x) + e(x) \qquad (4)$$

In case $g(x)$ represents the generator polynomial for an RS code, then $m(x)$ represents the message polynomial, and $e(x)$ represents the error polynomial. After obtaining the syndromes, the next step is to use the Euclidean approach to detect and correct any errors in the received code. The Euclidean approach is based on the division process carried out to a point where there is no more remainder. This means that the receiver will be able to know whether there are errors and which data requires correction (equation 5).

$$gcd(a, b) = gcd(b, a \bmod b) \qquad (5)$$

As such, there would be a reduced risk of labeling perfectly good data as erroneous and, therefore, minimizing unnecessary effort in the subsequent process. After identifying the errors through the RSE technique, the next phase will be to correct these errors. This will be through Sequential Concatenated Turbo Coding, a coding technique that constantly improves on error correction as long as it is applied.

The technique goes hand-in-hand with the error detection procedure that was implemented (RSE) to correct the erroneous data bit by bit, hence improving the error correction process every time it is implemented. Sequential Concatenated Turbo Coding is able to tolerate packet losses effectively while increasing the quality of data transmitted by refining the error correction from previous processes, especially during times when the network traffic is high. In the final stage after correcting the errors, the next procedure will be the decoding of the corrected data through the implementation of Adaptive Polar Coding with Reverse Polarization Transformation.

Decoding of the modified code sequence is done through Adaptive Polar Coding, depending on the nature of the received signal. In order to have exact recovery of the original signal data, the reverse process of polarization transformation is performed, and this is indicated in Equation (6).

$$u' = x' \times F' \qquad (6)$$

Therefore, the final decoding makes sure that the transmitted information has been decoded accurately for transmission or further processing. The Flexible Turbo Decoder implements the use of RSE code to detect errors in the information, sequential concatenated turbo coding to correct errors iteratively, and adaptive polar coding to decode the final information. Overall, the combination of all these techniques helps overcome transmission errors, minimize latency, and prevent packet loss.

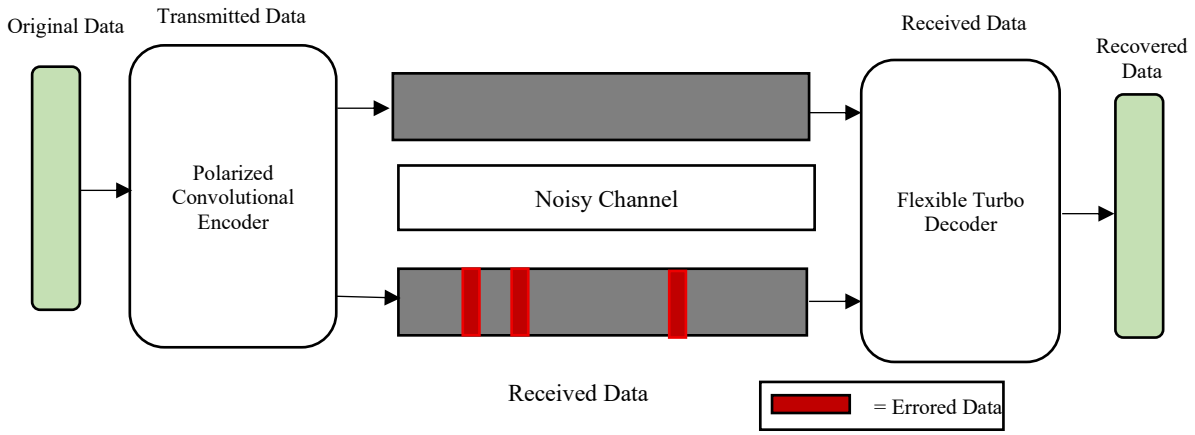


Figure 4. ACAPE-FID model overall architecture

The overall architecture of the ACAPE-FID model is depicted in figure 4. In the beginning, the data enters the Polarized Convolutional Encoder, where adaptive polar coding and convolutional coding take place. This data gets transferred through the noisy channel to the Flexible Turbo Decoder. In this case, the received data gets error detection by the RSE code and gets corrected by Sequential Concatenated Turbo Coding.

This error correction technique can help boost the effective data rate, expand the bandwidth availability, minimize retransmissions, and enhance data transfer efficiency. The subsequent section discusses the results and findings from this model.

RESULT AND DISCUSSION

In this section, an extensive review is provided of the implementation results, and the performance of the ACAPE-FID framework is presented. Additionally, a comparative study section is included to prove that the designed scheme performs efficient forward error correction in the wireless communication environment, as demonstrated on an FPGA platform.

System Configuration

This system is implemented in MATLAB with the help of the HDL coder (Table 1). The HDL coder tool is used in order to create an FPGA for implementing the system according to the architecture of the Xilinx Zynq-7000 series. This aspect of work provides a thorough evaluation of the performance of the system through the parameters such as its efficiency and response speed. Both of these characteristics have been optimized specifically for implementation on the FPGA of the chosen platform. The selection of this FPGA has a direct impact on the efficiency of operation as the algorithm can operate faster due to the capability of processing several requests simultaneously and the optimal allocation of logical devices on the board. Moreover, there will be a comparison of results achieved by this system with results achieved by other methods of image processing.

Table 1. Implementation environment and software configuration

Software	: MATLAB R2023a
OS	: Windows 10 (64-bit)
Processor	: Intel(R) Core (TM) i3-4130 CPU @ 3.40GHz 3.40 GHz
RAM	: 8GB RAM

Performance Metrics of the Proposed System

This section provides an analysis of the proposed technique and the outcomes achieved.

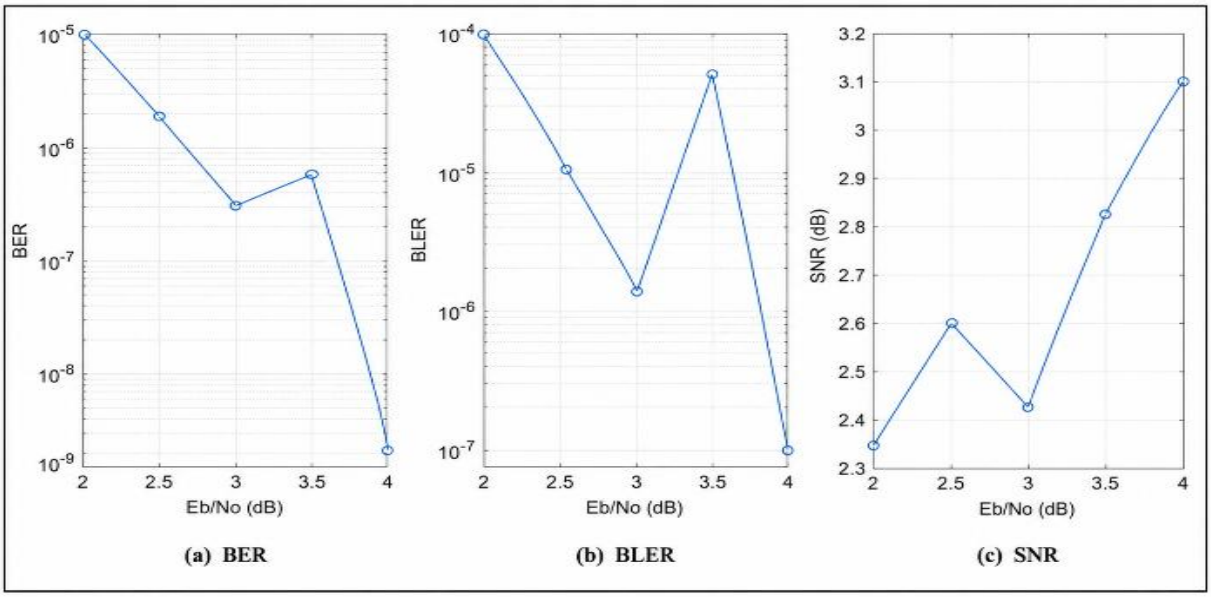
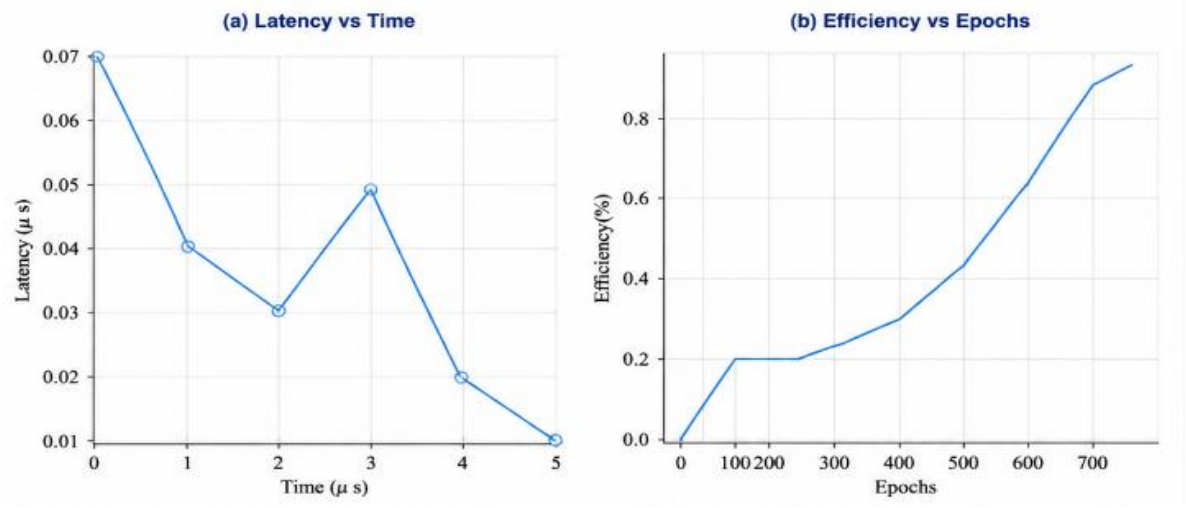


Figure 5(a-c). BER, BLER and SNR of the proposed system

In figure 5(a-c) present the enhanced performance of the proposed system based on BER, BLER, and SNR parameters for various Eb/N0 ratios. From figure 5a, the decrease in BER is from 10^{-5} at 2 dB to 10^{-9} at 4 dB due to Adaptive Polar Coding with the reverse polarization method. From figure 5b, the decrease in BLER is from 10^{-4} at 2 dB to 10^{-7} at 4 dB due to the effective error detection and correction by the Reed-Solomon Euclid Code at the block level. However, there are minor changes in the performance caused by adaptive coding, turbo coding, and channel noise at moderate SNR ranges. Furthermore, figure 5c indicates the effective increase in SNR by the Flexible Turbo Decoder from 2.3 dB to 3.1 dB.



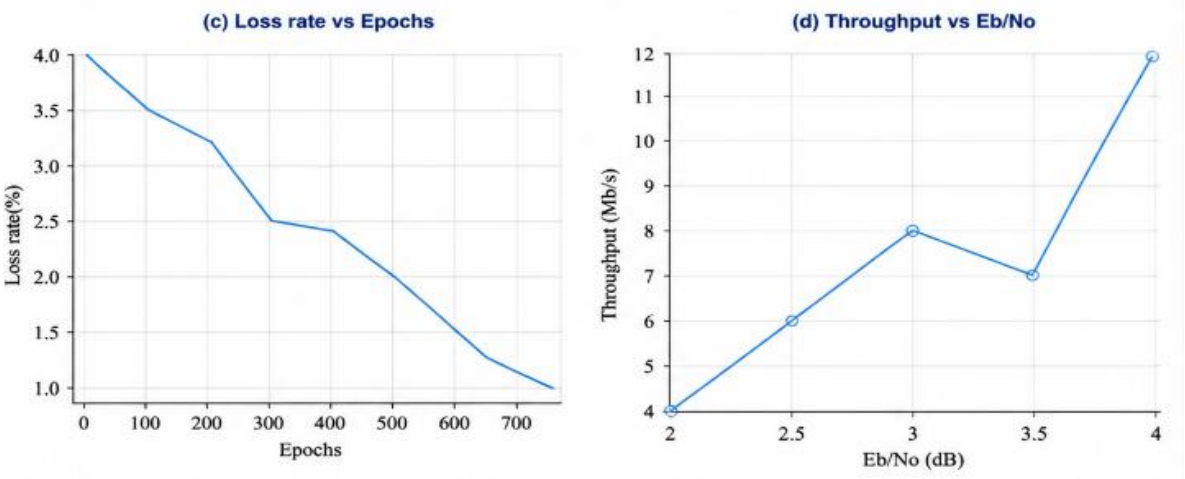


Figure 6(a-d). Latency, efficiency, loss rate and throughput of the proposed system

The latency performance, efficiency, loss rate, and throughput of the proposed model are represented in figure 6(a-d) respectively. From figure 6a, latency drops from 0.04 µs to 0.01 µs between 2 µs and 5 µs as the latency results due to Convolution-Assisted Polar Encoding with Flexible Iterative Decoding is more efficient. The graph shown in figure 6b denotes an improvement in the efficiency from 0.45% to 0.94% within the period of 500 and 700 epochs, respectively, by optimizing the bandwidth usage through iterations. The loss rate reduces from 2% to 1% in figure 6c using Adaptive Frozen Polar Coding and Convolutional Encoding. Figure 6d depicts an increase in throughput from 4 Mb/s to 12 Mb/s between 2 dB and 4 dB.

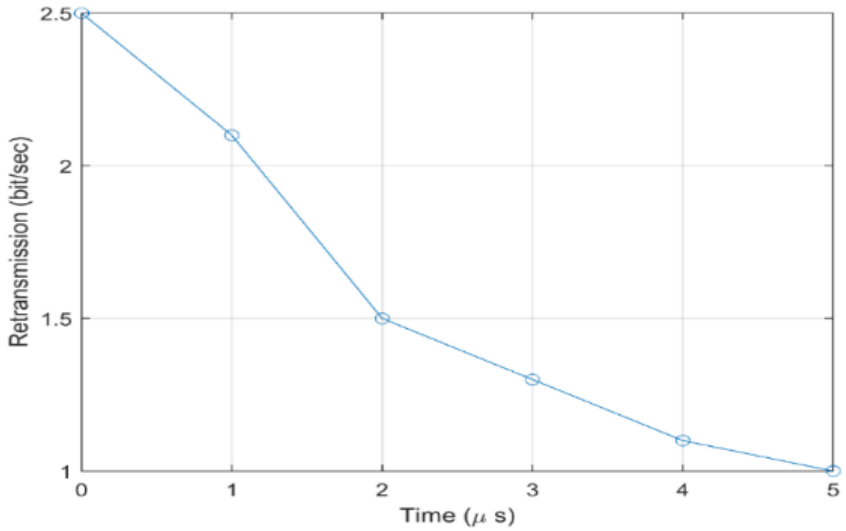


Figure 7. Retransmission of the proposed system

As illustrated in figure 7, this depicts the efficiency of the proposed model based on the retransmission rates during various time durations. The maximum retransmission rate is recorded at 1 microsecond, representing 2.1 bits per second, whereas the minimum rate is recorded at 5 microseconds, indicating 1 bit per second. The reason behind this result can be attributed to Sequential Concatenated Turbo Coding, where error correction capability is enhanced via various stages, improving data accuracy. However, as time durations are reduced, processing speed is increased, resulting in transmission errors.

In figure 8 demonstrates the number of clock cycles of the system within various time intervals. As demonstrated, the largest number of clock cycles takes place at five microseconds and amounts to 24, while the least number of clock cycles is 14 and occurs within one microsecond. Thus, one can conclude

that the performance of the system improves as time reduces. The increase in performance is possible due to using such techniques as Polarized Convolutional Encoder and Flexible Turbo Decoder.

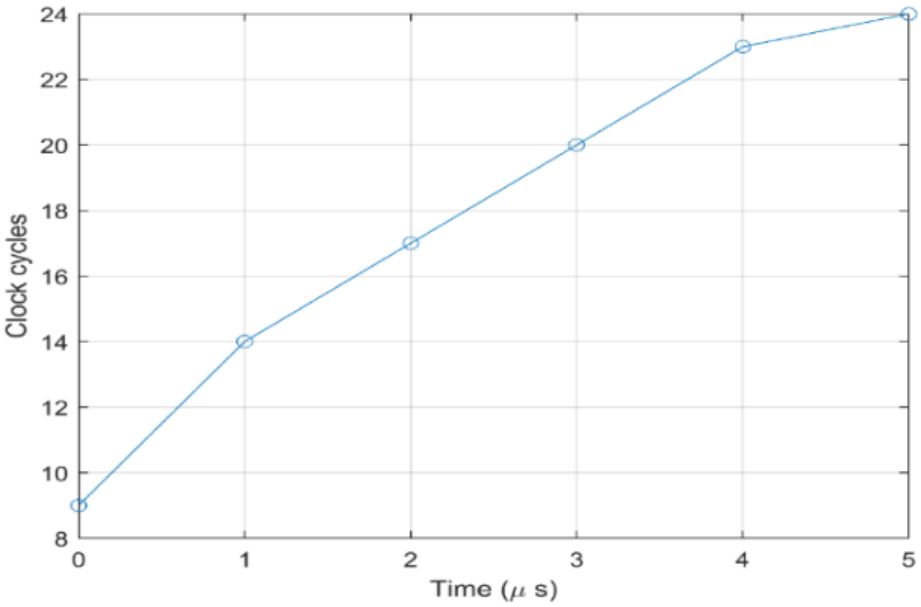


Figure 8. Clock cycle of the proposed system

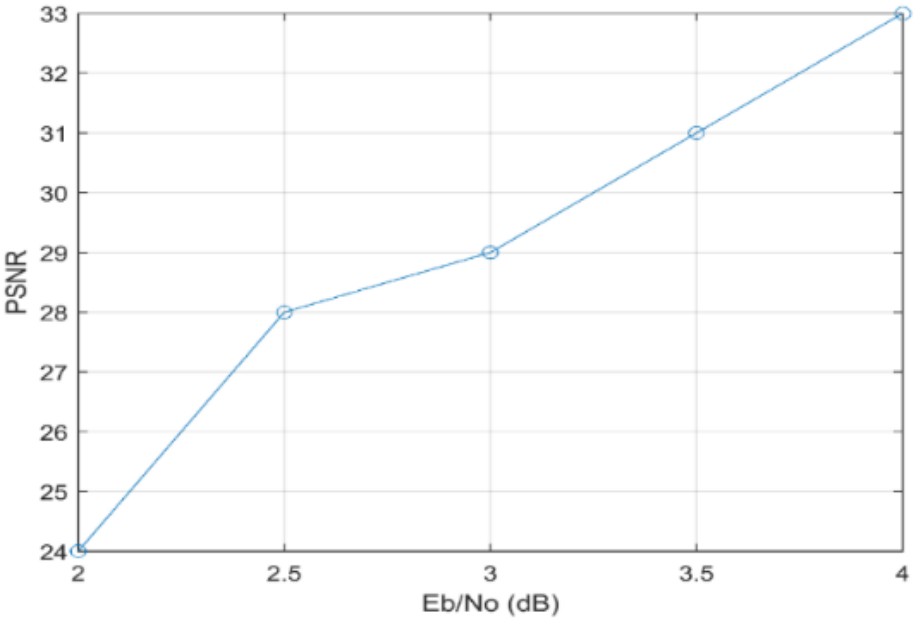


Figure 9. PSNR of the proposed system

The relationship between the Peak Signal to Noise Ratio (PSNR) and SNR per Bit (Eb/N0) in the proposed model is shown in figure 9. The highest PSNR value that can be achieved in the proposed model is 33 at 4 dB while the lowest is 24 at 2 dB. The PSNR measures the quality of the received signal with relation to the interfering signal, where high values mean accurate transmission. The increase in the SNR per Bit results in improved PSNR values due to efficiency in the transmission of signals without any errors. This can mainly be attributed to the employment of the modified RS code.

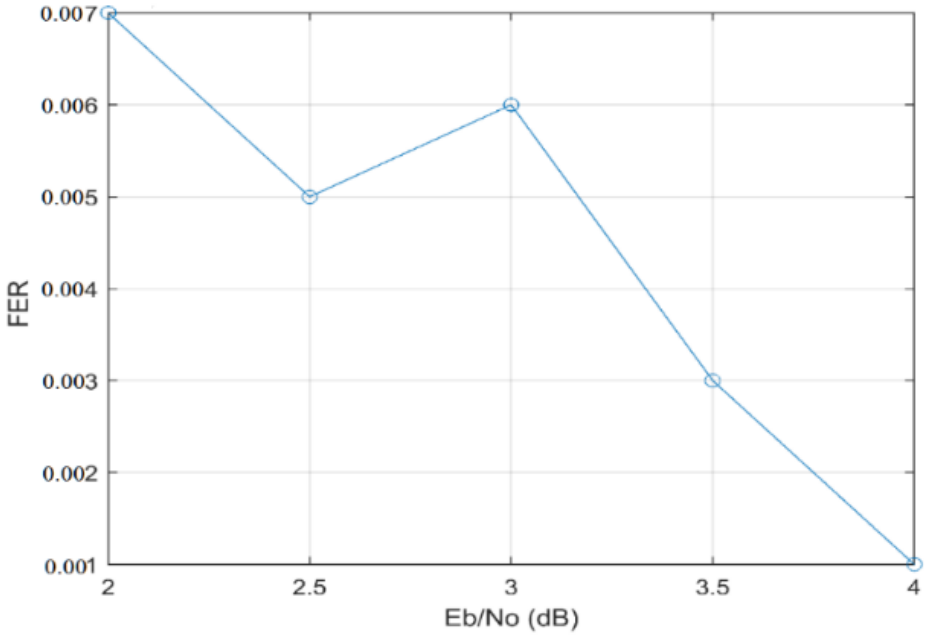


Figure 10. FER of the proposed system

The performance of the proposed model with respect to the FER for different SNR per bit values is demonstrated in figure 10. It can be seen that the FER drops from 0.007 for 2 dB to 0.001 for 4 dB, with some fluctuations observed for 2.5 and 3 dB. These results were obtained using the Flexible Turbo Decoder along with the modified version of Reed-Solomon codes, which can detect and correct errors during transmission.

Comparison with Previous Models

This part will determine how effective the ACAPE-FID model is through comparison with existing coding models using various parameters to evaluate their effectiveness. These parameters include BER, BLER, FER, latency, efficiency, loss, SNR, PSNR, clock cycles, and throughput. The ACAPE-FID model is compared with other models, including Polar codes [26][27], LDPC codes, TBCC, Turbo codes [26], RS codes [27], UEP, EEP [28], and Packet-Level FEC [12].

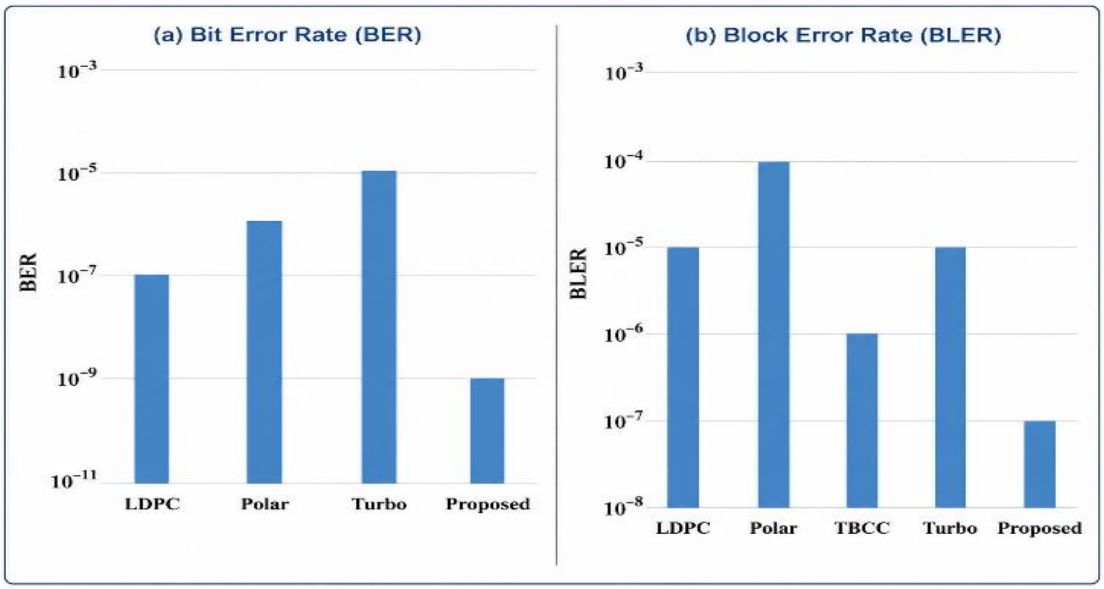


Figure 11 (a&b). Comparison of the BER and BLER of the proposed model

In figure 11(a), the Bit Error Rate (BER) for the proposed scheme is compared to that of other coding models. The values for the existing models LDPC, Polar, and Turbo are 10^{-7}, 10^{-6}, and 10^{-5}, respectively. The proposed model, on the other hand, performs better since it gives an even lower BER value of 10^{-9}. Figure 11(b) illustrates the comparison of the Block Error Rate (BLER). The BLER for LDPC, Polar, Turbo, and TBCC codes are 10^{-5}, 10^{-4}, 10^{-5}, and 10^{-6}, respectively.

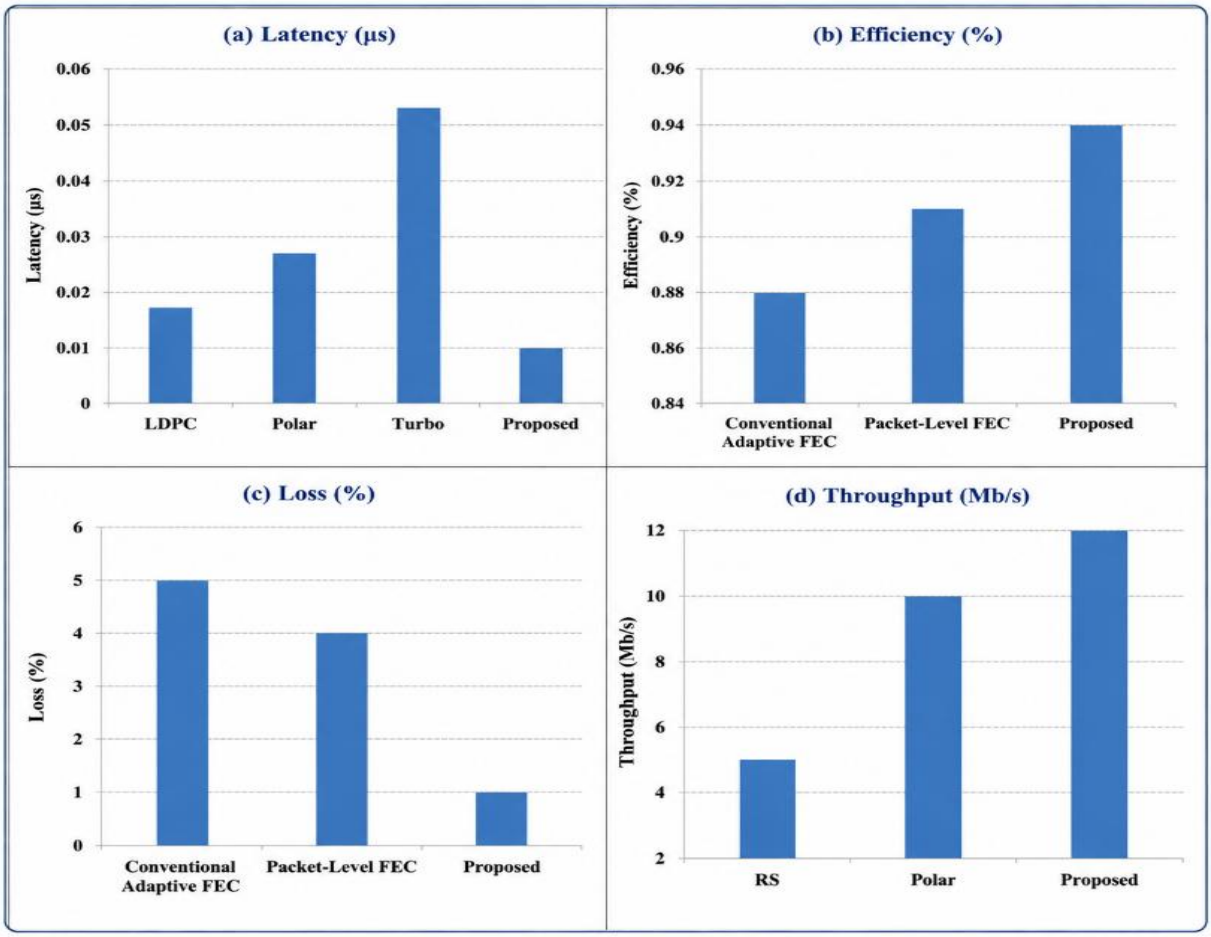


Figure 12(a-d). Comparison of the latency, efficiency, throughput and loss of the proposed model

Figure 12a displays the latency comparisons between the proposed ACAPE-FID and other coding methods. The existing models of LDPC, Polar, and Turbo have latencies of 0.017 µs, 0.027 µs, and 0.053 µs, respectively, while the latency in the proposed model is 0.01 µs. Figure 12b shows the efficiency of the models, where Conventional Adaptive FEC and Packet-Level FEC have efficiencies of 0.88% and 0.91%, respectively. The proposed model shows an efficiency of 0.94%. Figure 12c displays the loss rate comparison, where the loss rates for the existing models are 5% and 4%, and the loss rate for the proposed model is 1%. Finally, figure 12d displays the throughput comparisons, where the throughputs for the existing RS and Polar models are 5 Mb/s and 10 Mb/s, respectively, while that of the proposed model is 12 Mb/s.

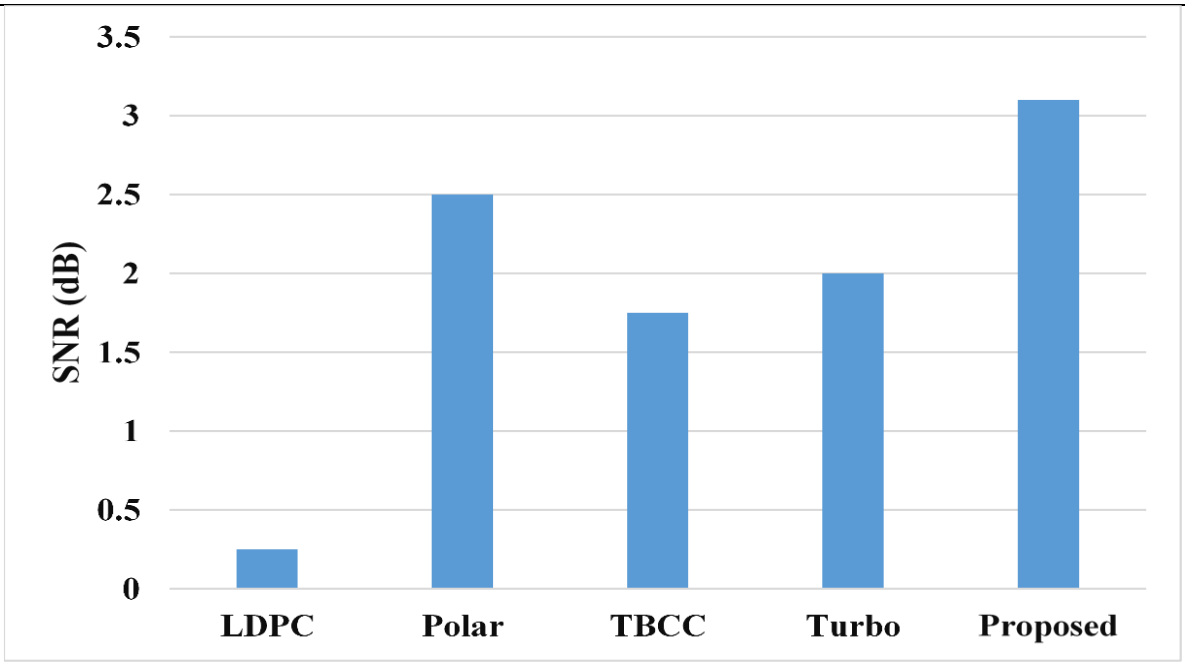


Figure 13. Comparison of the SNR of the proposed model

In figure 13 shows the comparison between SNR of the proposed model and those of the present models. The SNR of the present models, that is, LDPC, Polar, TBCC, and Turbo, is found to be 0.25dB, 2.5dB, 1.75dB, and 2dB, respectively. However, the SNR of the suggested model is higher than those values and is 3.1dB.

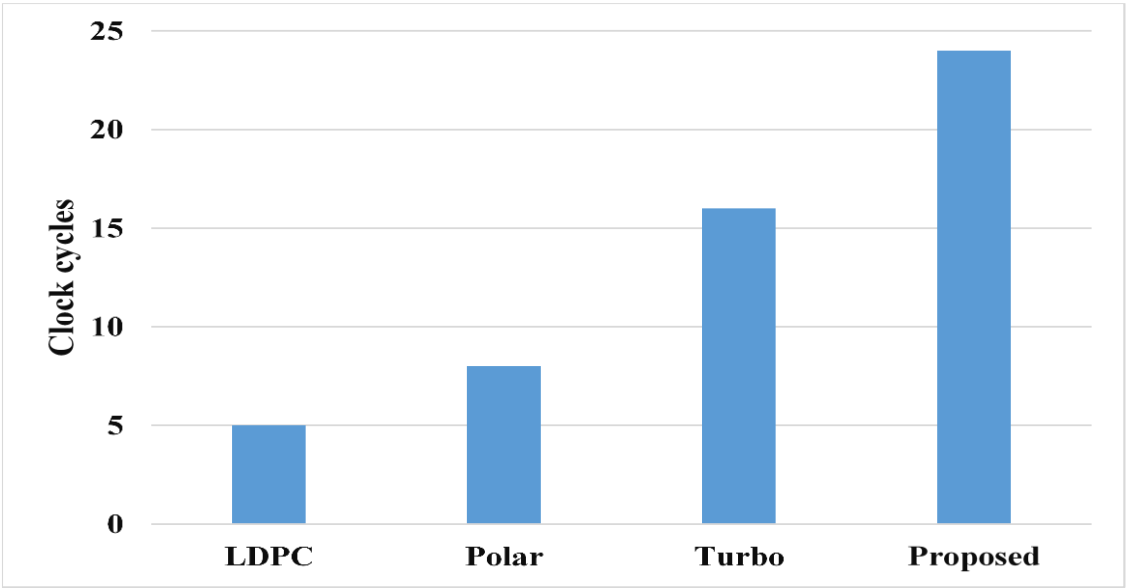


Figure 14. Comparison of the clock cycle of the proposed model

The figure 14 shows a comparison of the clock cycle of the proposed model with the existing models. The present models, including LDPC, Polar, and Turbo, have clock cycles of 5, 8, and 16. Compared with the existing models, the proposed model achieves a maximum clock cycle of 24.

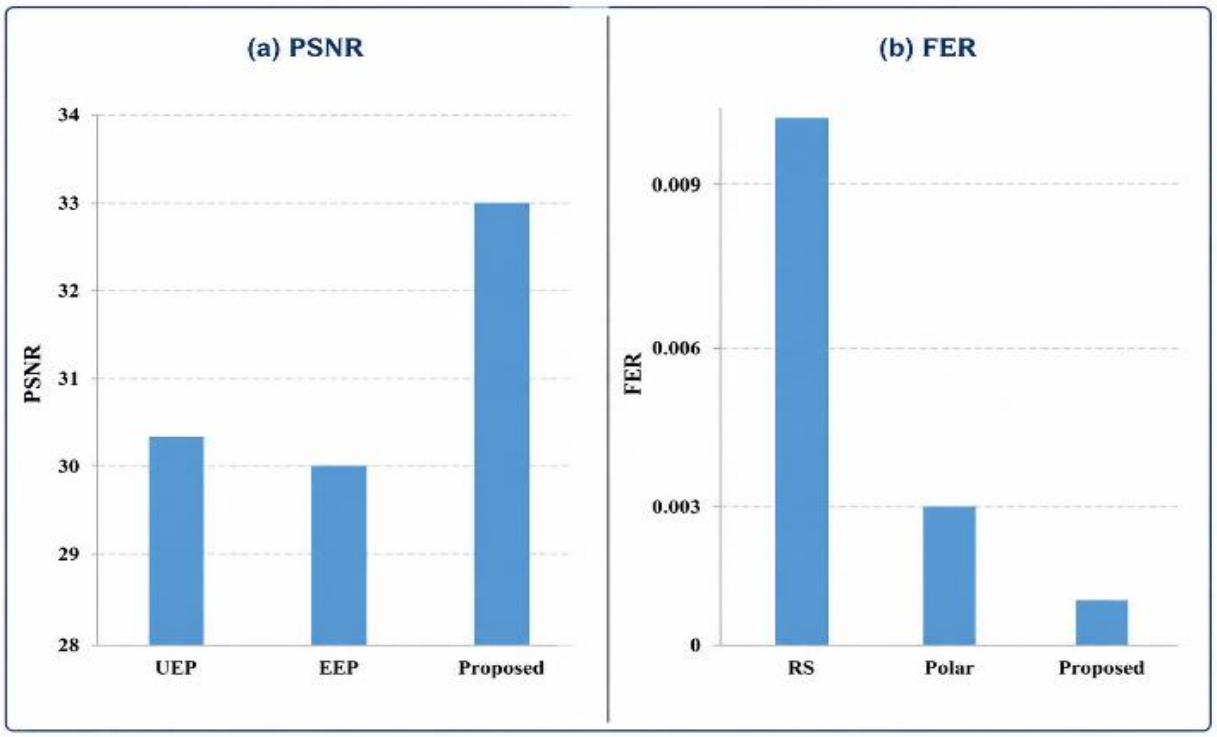


Figure 15(a&b). Comparison of the PSNR and FER of the proposed model

The PSNR of the proposed model and that of current models are depicted in figure 15a. The PSNR for the current models is 30.4 for UEP and 30 for EEP, while the PSNR of the proposed model is 33. The FER of the proposed model is compared with that of current models and is illustrated in figure 15 b. The FER for the existing models, such as RS and Polar, is 1 and 0.003, respectively. The proposed model gives a considerably low FER of 0.001.

In the Results part, the performance of the proposed model is compared with that of existing models by explaining the graphically represented data. This implies that the proposed Convolution Assisted Polar Encoder with Flexible Iterative Decoding model gives low BER, BLER, and FER of 10^-9, 10^-7, and 0.001, respectively, while the latency is 0.01 microseconds, loss is 1%, and clock cycle is 24. Also, the performance gives high efficiency, SNR, PSNR, and throughput of 0.94%, 3.1dB, 33, and 12MB/s respectively.

CONCLUSION

ACAPE-FID is introduced in this work as an advanced scheme for FEC in wireless communication with high performance levels. ACAPE-FID achieves high performance through integrating adaptive frozen polar coding, convolutional coding, and iterative decoding using the Turbo method. This results in eliminating any redundant data, lowering re-transmission rates, increasing error detection and correction ability. The performance level of ACAPE-FID after FPGA implementation was assessed using the Xilinx Zynq-7000 series board, proving the capability of maintaining ultra-low latencies and high throughput. The performance analysis between ACAPE-FID and other common FEC methods, including Polar, LDPC, TBCC, and Turbo, showed great improvements in parameters like BER, BLER, FER, latency, throughput, SNR, and PSNR. The use of an adaptive coding scheme allows ACAPE-FID to adjust its coding scheme depending on both the length of data inputs and the wireless channels' characteristics. The use of an iterative decoding scheme allows correcting errors in received data, while the RSE-based detection scheme enables detecting errors in transmitted data and correcting them. In conclusion, the proposed ACAPE-FID can be considered as a scalable and effective FEC scheme suitable for modern wireless communication needs.

DECLARATIONS

Funding

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Competing Interests

The authors have no relevant financial or non-financial interests to disclose.

Data Availability

Data sharing is not applicable to this article.

Conflict of Interest

None

REFERENCES

[1] Luvisotto M, Pang Z, Dzung D. High-performance wireless networks for industrial control applications: New targets and feasibility. Proceedings of the IEEE. 2019 Mar 5;107(6):1074-93. https://doi.org/10.1109/JPROC.2019.2898993

[2] Seijo Ó, López-Fernández JA, Val I. W-SHARP: Implementation of a high-performance wireless time-sensitive network for low latency and ultra-low cycle time industrial applications. IEEE Transactions on Industrial Informatics. 2020 Jul 7;17(5):3651-62. https://doi.org/10.1109/TII.2020.3007323

[3] Tandon B, Thakur M. An overview of adaptive signal processing methods for 6G wireless communication networks. International Academic Journal of Science and Engineering. 2025;12(1):12-5. https://doi.org/10.71086/IAJSE/V12I1/IAJSE1203

[4] Zhan M, Pang Z, Dzung D, Xiao M. Channel coding for high performance wireless control in critical applications: Survey and analysis. IEEE Access. 2018 May 31;6:29648-64. https://doi.org/10.1109/ACCESS.2018.2842231

[5] Rezaei F, Galappaththige D, Tellambura C, Herath S. Coding techniques for backscatter communications—A contemporary survey. IEEE Communications Surveys & Tutorials. 2023 Mar 22;25(2):1020-58. https://doi.org/10.1109/COMST.2023.3259224

[6] Devi TR, Kamalnathan C. Convolution Assisted Polar Encoder with Flexible Iterative Decoding for Forward Error Correction in Wireless Communication Incorporated in Fpga. Infocommunications Journal. 2025;17(3):54-61. https://doi.org/10.36244/ICJ.2025.3.7

[7] Kanapala S, Hussain SJ. Optimized FBMC-OQAM-CDMA framework for low-complexity interference mitigation in MIMO systems with space-time coding. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications. 2026;17(1):120-141. https://doi.org/10.58346/JOWUA.2026.I1.008

[8] Cavalcanti D, Perez-Ramirez J, Rashid MM, Fang J, Galeev M, Stanton KB. Extending accurate time distribution and timeliness capabilities over the air to enable future wireless industrial automation systems. Proceedings of the IEEE. 2019 Mar 21;107(6):1132-52. https://doi.org/10.1109/JPROC.2019.2903414

[9] Hailes P, Xu L, Maunder RG, Al-Hashimi BM, Hanzo L. A survey of FPGA-based LDPC decoders. IEEE Communications Surveys & Tutorials. 2015 Dec 18;18(2):1098-122. https://doi.org/10.1109/COMST.2015.2510381

[10] Perdomo-Campos A, Vega-González I, Ramírez-Beltrán J. ESP32 based low-power and low-cost wireless sensor network. InThe conference on Latin America Control Congress 2020 Nov 30 (pp. 275-285). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-26361-3_24

[11] Sappa A. Mitigating Propagation Faults in Real-Time Content Streaming for Low-Bandwidth Learning Environments. Archives for Technical Sciences. 2025 Dec;34(3):807-14.

https://doi.org/10.70102/afts.2025.1834.807

[12] Nafaa A, Taleb T, Murphy L. Forward error correction strategies for media streaming over wireless networks. IEEE Communications Magazine. 2008 Jan 7;46(1):72-9. https://doi.org/10.1109/MCOM.2008.4427233

[13] Hajiyat ZR, Sali A, Mokhtar M, Hashim F. Channel coding scheme for 5G mobile communication system for short length message transmission. Wireless Personal Communications. 2019 May 1;106(2):377-400. https://doi.org/10.1007/s11277-019-06167-7

[14] Yang SS, Liu JQ, Lu ZG, Bai ZL, Wang XY, Li YM. An FPGA-based LDPC decoder with ultra-long codes for continuous-variable quantum key distribution. IEEE Access. 2021 Mar 12;9:47687-97. https://doi.org/10.1109/ACCESS.2021.3065776

[15] Ferraz O, Subramaniyan S, Chinthala R, Andrade J, Cavallaro JR, Nandy SK, Silva V, Zhang X, Purnaprajna M, Falcao G. A survey on high-throughput non-binary LDPC decoders: ASIC, FPGA, and GPU architectures. IEEE Communications Surveys & Tutorials. 2021 Nov 8;24(1):524-56. https://doi.org/10.1109/COMST.2021.3126127

[16] Vinodhini M, Murty NS, Ramesh TK. Transient error correction coding scheme for reliable low power data link layer in NoC. IEEE Access. 2020 Sep 22;8:174614-28. https://doi.org/10.1109/ACCESS.2020.3025770

[17] Abbas SM, Jalaleddine M, Gross WJ. High-throughput VLSI architecture for GRAND Markov order. In: 2021 IEEE Workshop on Signal Processing Systems (SiPS). IEEE; 2021. p. 158-163. https://doi.org/10.1109/SiPS52927.2021.00036

[18] Kumar KN, Reddy NA, Shanmukh P, Vinodhini M. Matrix based error detection and correction using minimal parity bits for memories. In2020 IEEE international conference on distributed computing, VLSI, electrical circuits and robotics (DISCOVER) 2020 Oct 30 (pp. 100-104). IEEE. https://doi.org/10.1109/DISCOVER50404.2020.9278030

[19] Zhang Q, Ibi S, Takahashi T, Iwai H. Deep Unfolding-Aided Sum-Product Algorithm for Error Correction of CRC Coded Short Message. In2022 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC) 2022 Nov 7 (pp. 1737-1743). IEEE. https://doi.org/10.23919/APSIPAASC55919.2022.9979875

[20] Wang L, Song D, Areces F, Wesel RD. Achieving short-blocklength RCU bound via CRC list decoding of TCM with probabilistic shaping. InICC 2022-IEEE International Conference on Communications 2022 May 16 (pp. 2906-2911). IEEE. https://doi.org/10.1109/ICC45855.2022.9838498.

[21] Matsenko S, Borysenko O, Spolitis S, Udalcovs A, Gegere L, Krotov A, Ozolins O, Bobrovs V. FPGA-Implemented Fractal Decoder with Forward Error Correction in Short-Reach Optical Interconnects. Entropy. 2022 Jan 13;24(1):122. https://doi.org/10.3390/e24010122

[22] Bai W, Zou X, Li P, Ye J, Yang Y, Yan L, Pan W, Yan L. Photonic millimeter-wave joint radar communication system using spectrum-spreading phase-coding. IEEE Transactions on Microwave Theory and Techniques. 2022 Jan 13;70(3):1552-61. https://doi.org/10.1109/TMTT.2021.3138069

[23] Bhargavi C, Nishanth DV, Nikhita P, Vinodhini M. H-matrix based error correction codes for memory applications. In2021 International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT) 2021 Feb 19 (pp. 1-5). IEEE. https://doi.org/10.1109/ICAECT49130.2021.9392574

[24] Devi TR, Kamalnathan C. Convolution Assisted Polar Encoder with Flexible Iterative Decoding for Forward Error Correction in Wireless Communication Incorporated in Fpga. Infocommunications Journal. 2025;17(3):54-61. https://doi.org/10.36244/ICJ.2025.3.7

[25] Wang LN, Wei H, Zheng Y, Dong J, Zhong G. Deep error-correcting output codes. Algorithms. 2023 Dec 4;16(12):555. https://doi.org/10.3390/a16120555

[26] Fanari L, Iradier E, Bilbao I, Cabrera R, Montalban J, Angueira P, Seijo O, Val I. A survey on FEC techniques for industrial wireless communications. IEEE Open Journal of the Industrial Electronics Society. 2022 Nov 4;3:674-99. https://doi.org/10.1109/OJIES.2022.3219607

[27] Spyrou ED, Kappatos V. Application of Forward Error Correction (FEC) Codes in Wireless Acoustic Emission Structural Health Monitoring on Railway Infrastructures. Infrastructures. 2022 Mar 16;7(3):41. https://doi.org/10.3390/infrastructures7030041

[28] Legrand A, Macq B, De Vleeschouwer C. Forward error correction applied to JPEG-XS codestreams. In2022 IEEE International Conference on Image Processing (ICIP) 2022 Oct 16 (pp. 3723-3727). IEEE. https://doi.org/10.1109/ICIP46576.2022.9897287