Design and VLSI Implementation of Hybrid LDPC-STBC Codes for Enhanced Data Transmission in Satellite Communications

Bandari Nithya[1], Kunsoth Umarani[1], Bandari Srilekha[1]
[1]Department of Electronics and Communication Engineering
[1]Sree Dattha Group of Institutions, Sheriguda, Hyderabad, Telangana.

Abstract

Satellite communication systems play a crucial part in promoting worldwide connection. As a result, the development of robust error correction codes is required in order to guarantee the transfer of data in a reliable manner. As a result of the desire for larger data rates and improved spectral efficiency in modern communication networks, new error correction algorithms have been incorporated into these systems. On the other hand, existing systems frequently struggle to find a balance between the complexity of their operation, the amount of power they consume, and their overall performance. The purpose of this work is to solve the constraints of existing systems by putting forward a hybrid solution that combines the benefits of LDPC (Low-Density Parity-Check) codes with STBC (Space-Time Block Coding) approaches. In order to improve the error correction capabilities of satellite communication systems and guarantee reliable data transmission even when channel impairments are present, the hybrid LDPC-STBC codes have been developed. In spite of the progress that has been made in error correction coding, the systems that are currently in use have significant shortcomings. These downsides include increased computational complexity, higher power consumption, and the possibility of performance loss under adverse channel circumstances. These problems are alleviated by the hybrid LDPC-STBC codes that have been devised. These codes take advantage of the advantages that both coding schemes offer, so establishing a synergistic balance between the performance of error correction and the efficiency of computing.

Keywords: Space-Time Block Coding, Low-Density Parity-Check, Error Correction Codes, Very Large-Scale Integration.

DOI Number: 10.48047/nq.2020.18.10.NQ20249 NeuroQuantology 2020; 18(10):208-215

1. Introduction

ECC is the technique most used to mitigate memory failures. It was initially proposed by Hamming to avoid errors on a relay-based machine at Bell Labs. The ECC basis adds check bits to a set of data bits to perform a codeword for detecting and correcting errors using Boolean logic relations, usually XOR operations. This technique requires an encoder to calculate the check bits and associate them to the data bits producing a codeword; a decoder analyses the codeword, detecting and correcting possible errors. Device miniaturization increasingly leads to MCU that many times one-dimensional (1D)-ECCs are inefficacious to correct. Thus, n-dimensional codes have been used, mainly two-dimensional (2D)-ECCs like product codes, because they provide higher detection and correction power with proportionally less energy and area consumption.ECC represent a crucial aspect of modern data transmission and storage systems, providing robust mechanisms for detecting and correcting errors that may occur during communication or retrieval processes. Essentially, ECC

techniques involve adding redundancy to the transmitted or stored data in a structured manner, allowing for the detection and correction of errors caused by various factors such as noise, interference, or hardware malfunctions. By employing sophisticated algorithms and encoding schemes, ECC can not only identify errors but also reconstruct the original data, ensuring data integrity and reliability. These codes find wide application in diverse fields including telecommunications, digital storage devices, wireless networks, and satellite communications, where data accuracy and reliability are paramount. Through continuous research and development, ECC continue to evolve, offering increasingly efficient and resilient solutions to address the challenges of modern data communication and storage environments.

2. Literature Survey

Singh, S. Pratap, et al. [1] developed Molecular Communication (MC) as a multidisciplinary branch that lay at the junction of nano, bio, and communication technology. MC evolved to serve almost every field of humanity, be it biomedical, environmental, or security against NBC attack. Silva, Felipe, et al. [2] explored how the evolution of microelectronics boosted more scalable and complex circuit designs, providing high processing speed and greater storage capacity. However, reliability issues grew significantly as electronic devices scaled down, increasing the fault rate, mainly in critical applications exposed to radiation. Memories were sensitive to charged particles, which could corrupt data due to transient effects. Chu, Syuna-A Ke, et al. [3] introduced Guessing random additive noise decoding (GRAND) as a recently proposed code-agnostic decoding technique for linear block codes, which attempted to guess the possible error pattern applied on the received word to check if the result was a valid codeword. The GRAND with abandonment (GRANDAB) served as a hard-detection decoder by limiting the number of generated test error patterns. Kuo, Yao-Ming, et al. [4] illustrated the recommendation by the Consultative Committee for Space Data Systems (CCSDS) to utilize short-block length Bose–Chaudhuri–Hocquenghem and binary LDPC. Despite the significant error-correction capacity of nonbinary low-density parity-check (NB-LDPC) codes, their consideration had been lacking due to the complexity associated with decoding. Grurl, Thomas, Christoph Pichler, et al. [5] discussed the challenges posed by frequent noise effects in real quantum computers due to the fragility of quantum mechanical effects, resulting in errors during computations. Quantum error-correcting codes were proposed as a solution to identify and correct these errors.

Wu, Yujun, Bin Wu, et al. [6] developed the QC-LDPC code, with its excellent error correction performance and hardware friendliness, and was identified as one of the channels encoding schemes by Wi-Fi 6. Pokhrel, Nabin Kumar, et al. [7] presented a class of integer codes capable of correcting burst asymmetric errors. The presented codes were constructed with the help of a computer and had the potential to be used in various practical systems, such as optical networks and VLSI memories. Saini, Madan Lal, et al. [8] proposed: "In digital communication, various single- and double-bit error correcting and detecting codes were available. The efficiency of an error correcting code was evaluated by its error correction capabilities and redundancy. Ponmalar, VJ Beulah Sherin, et al. [9] developed a convolution code namely LDPC which was proposed and implemented in VLSI architectures (FPGA). In general, a digital communication system suffered from errors due to noise, distortion, and interference during data transmission and various algorithms commonly used to correct the errors. Boncalo, Oana, et al. [10] proposed a novel iterative decoding method - Gradient Descent Symbol Update - for real number parity-based ECC, as well as its corresponding hardware architecture. The decoding process was based on the gradient descent optimization technique, as well as binary maximum likelihood error correction decoding.

Dutta, Shruti, et al. [11] proposed the applications involving machine learning and neural networks had become increasingly essential in the AI revolution. Emerging trends in Resistive RAM technologies provided high-speed, low-cost, scalable solutions for such applications. These RRAM cells provided efficient and sophisticated memory hardware structures for machine-learning applications. Dhandapani, et al. [12] proposed the implementation of a BCH Encoder in the ZYNQ-7000 AP SOC to guarantee that the sensitive data acquired from capsule endoscopy was transmitted without any errors through a wireless medium. The BCH Encoder and Decoder were designed, synthesized, and simulated using Xilinx Vivado. Ajmal, Muhammad, et al. [13] presented the first memory-less transition bus encoding technique for low power dissipation, crosstalk avoidance, and error correction simultaneously. They constructed optimal or asymptotically optimal constant weight codes that eliminated each kind of crosstalk. Maity, Raj Kumar, et al. [14] addressed the increasing importance of Error Correcting Codes (ECCs) for protecting memories from localized errors, which are primarily caused by radiation-induced soft errors corrupting data stored in a single cell or multiple cells of a memory. Initially, Single Error Correction (SEC) and Single Error Correction-Double Error Detection (SEC-DED) codes were widely used to protect memories against soft errors. Gracia-Morán, L. J. Saiz-Adalid, et al. [15] observed that during these last years, the use of embedded systems has grown exponentially, mainly due to the expansion of the Internet of Things (IoT). Data collected by IoT devices were sent to the cloud to be processed in datacentres. Edge Computing philosophy aimed to change this “passive” behavior of IOT devices.

3. Proposed Methodology

LDPC codes are a class of forward error correction codes known for their exceptional performance and low decoding complexity. LDPC codes gained renewed interest in the late 1990s due to their near-capacity performance when decoded using iterative message-passing algorithms. These codes are defined by sparse parity-check matrices, which contribute to efficient encoding and decoding processes. LDPC codes have been widely adopted in modern communication systems, including wireless, optical, and satellite communications, due to their ability to approach the Shannon limit, providing reliable data transmission over noisy channels.The versatility of LDPC codes extends to various communication standards and applications, offering flexibility in coding rates and adaptability to different channel conditions. Their robust error correction capabilities make them particularly suitable for applications where reliability is paramount, such as digital broadcasting, deep-space communication, and high-speed data transmission. Ongoing research and advancements in LDPC code design and decoding algorithms continue to enhance their performance and expand their applicability in emerging communication technologies, ensuring their relevance in the ever-evolving landscape of digital communication systems.

STBC is a technique used in wireless communications to improve the reliability of data transmission over multiple-input multiple-output (MIMO) channels. STBC encodes data across multiple antennas at the transmitter, creating a signal that is transmitted simultaneously from each antenna. This technique utilizes the spatial diversity inherent in MIMO systems to combat fading and interference. The STBC works by transmitting a linear combination of data symbols from multiple antennas in a specific pattern known as a space-time block code. At the receiver, these transmitted signals are decoded to recover the original data, taking advantage of the diversity provided by the multiple antennas. By exploiting the spatial dimension of the communication channel, STBC can improve the reliability and performance of wireless communication systems, particularly in scenarios with multipath propagation and fading. STBC has found applications in various wireless

communication standards, including Wi-Fi, LTE, and WiMAX, enabling more robust and efficient data transmission in diverse environments.

3. Proposed Methodology

Hybrid LDPC-STBC combines LDPC coding for error correction and STBC for diversity gain in wireless communications. LDPC corrects errors efficiently, while STBC enhances signal reliability through diversity. This hybrid approach optimizes spectral efficiency and robustness in modern communication systems.

Figure 1 shows the proposed LDPC-STBC encoder architecture. Initially, LDPC Encoder operates by partitioning a message into blocks and generating parity bits to enhance error correction capabilities. It begins by creating a parity check matrix with a low density of ones. The message bits are multiplied by this matrix, resulting in parity bits appended to the original message. These parity bits are calculated based on predefined rules dictated by the LDPC code structure. The encoder ensures a balanced distribution of ones in the output codeword, optimizing error correction efficiency. This process provides redundancy for error detection and correction, crucial in reliable communication systems such as wireless networks and storage devices. The STBC is a technique used in multiple-input multiple-output (MIMO) wireless communication systems to improve reliability. In STBC encoding, data symbols are distributed across multiple antennas and transmitted over multiple time intervals. The encoder creates a matrix of symbol vectors, where each vector represents the symbols transmitted simultaneously from different antennas. By carefully designing these matrices, STBC ensures that the transmitted signals have diversity properties, enhancing the system's resilience to fading and interference. At the receiver, these encoded signals are decoded to recover the transmitted symbols, exploiting the diversity introduced by the encoding process to improve reliability.

Concatenating LDPC codes with STBC involves encoding data in two stages. Firstly, LDPC encodes the input data using a sparse parity-check matrix to create redundancy for error correction. Secondly, STBC encodes the LDPC-coded symbols across multiple antennas and time slots, exploiting diversity to enhance reliability in wireless communication. The LDPC-coded symbols are spread across the STBC matrix, facilitating robust transmission against fading channels. Concatenating LDPC with STBC thus leverages the error-correction capability of LDPC codes with the diversity gain of STBC, offering improved performance in noisy and fading wireless channels.

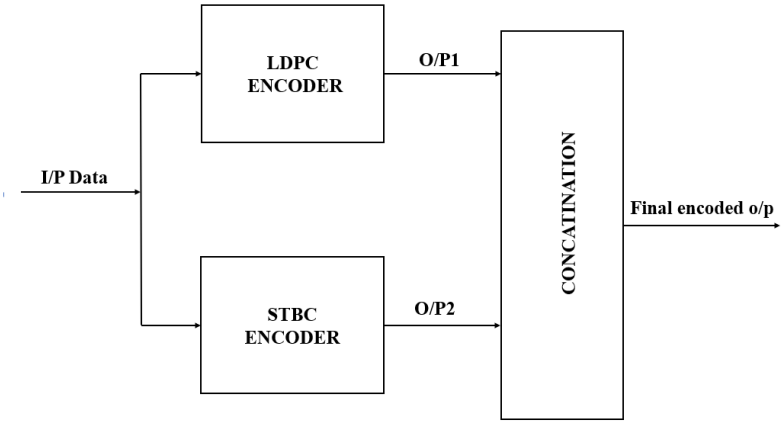

Figure 1.Proposed LDPC-STBC Encoder

Figure 2 shows the proposed LDPC-STBC decoder architecture. The LDPC-STBC data splitting involves dividing input bits into LDPC codewords. Each codeword undergoes STBC encoding, then the resulting symbols are arranged into blocks for transmission over multiple antennas, facilitating robustness against fading. Decoding at the receiver

involves LDPC decoding followed by STBC decoding. The LDPC decoding begins by initializing variable nodes with received symbols. Messages are passed iteratively between variable and check nodes. At each iteration, variable nodes compute check node messages based on received symbols and previous messages. Check nodes then update variable node messages according to parity check equations. This process iterates until either a maximum number of iterations is reached, or all parity check equations are satisfied. Finally, the decoder outputs the estimated codeword based on the variable node messages. This iterative process aims to minimize the discrepancy between received symbols and the estimated codeword, achieving reliable decoding of LDPC codes.

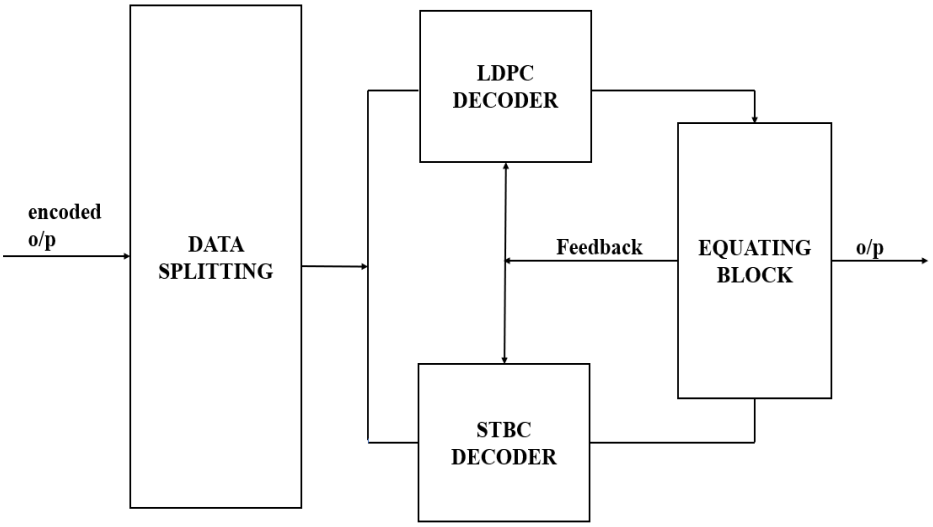

Figure 2. Proposed LDPC-STBC Decoder.

Further, STBC decoding involves recovering transmitted symbols from multiple antennas to improve reliability in wireless communication. The decoder utilizes received signals from different antennas, applies matrix operations such as maximum likelihood or minimum mean square error, and estimates the original transmitted symbols. By exploiting spatial diversity, STBC decoding enhances signal integrity, mitigating fading effects, and enhancing overall communication performance.In LDPC-STBC decoding, feedback involves iteratively refining soft information about transmitted symbols. The decoder utilizes soft information from received symbols to update the reliability of symbols, incorporating feedback from previously decoded symbols. Equated output combines LDPC decoding with STBC decoding, jointly optimizing symbol detection. It iteratively refines soft information using LDPC decoding techniques while exploiting STBC diversity. This process enhances symbol reliability, mitigating errors introduced by fading channels and noise. Ultimately, equated output produces improved symbol estimates by leveraging both LDPC and STBC decoding advantages in a synergistic manner.

4. Results and Discussion

Figure 3 displays the simulation results of the proposed LDPC-STBC method for N=32. Figure 4illustrates the area estimation of implementing the proposed LDPC-STBC method for N=32. It may detail the hardware resources required, such as the number of logic elements, registers, or other components, indicating the area footprint of the method.Figure 5 presents a power estimation for implementing the proposed LDPC-STBC method for N=32. It may include metrics such as static power (power consumed when idle) and dynamic power (power consumed during operation), providing insights into the energy efficiency of the method.Figure 6 depicts the setup delay estimation for implementing the proposed LDPC-STBC method for N=32. It may detail the timing characteristics related to the setup

time requirement, ensuring that data is stable and valid before processing.

Figure 7 illustrates the hold delay estimation for implementing the proposed LDPC-STBC method for N=32. It may detail the timing characteristics related to the hold time requirement, ensuring that data remains stable and valid throughout the entire processing cycle.Table 1 compares the performance of the existing LDPC-STBC method with the proposed method for N=32 across various metrics. These metrics include resource utilization (LUTs, IO), total power consumption (static and dynamic), and total delay (logic delay, net delay). The comparison highlights the differences and improvements achieved by the proposed method compared to the existing one, expressed as a percentage change.

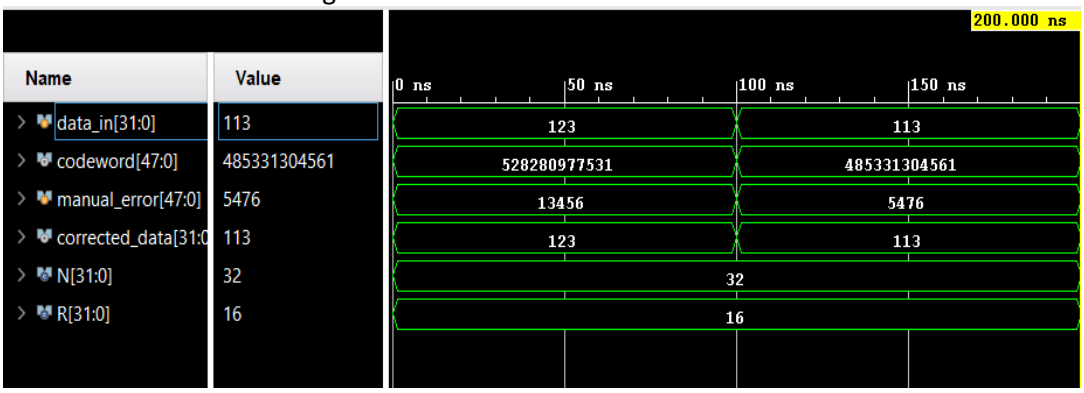

Figure 3. Proposed Simulation Results for N=32

Resource	Estimation	Available	Utilization %
LUT	6	134600	0.01
IO	112	500	22.40

Figure 4. Proposed Area for N=32.

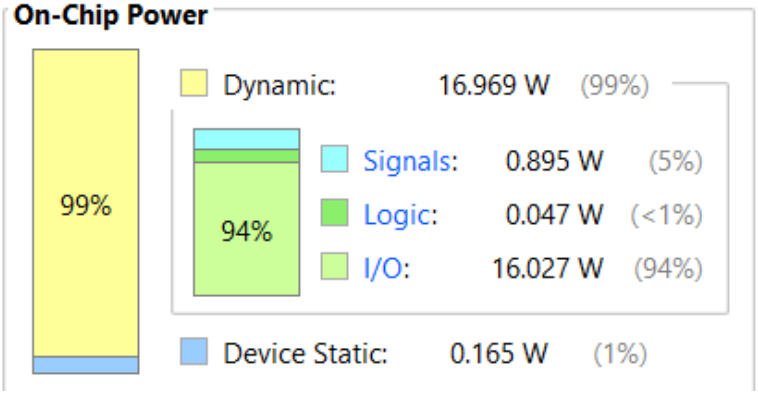

Figure 5. Proposed Power for N=32

Name	Slack	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock	Destination
Path 1	∞	4	3	4	data_in[7]	corrected_data[0]	14.766	4.270	10.496	∞	input port clock	
Path 2	∞	2	1	4	data_in[9]	corrected_data[9]	13.225	3.972	9.253	∞	input port clock	
Path 3	∞	2	1	2	data_in[0]	codeword[32]	13.050	3.902	9.148	∞	input port clock	
Path 4	∞	2	1	4	data_in[14]	corrected_data[14]	12.989	3.972	9.016	∞	input port clock	
Path 5	∞	2	1	4	data_in[10]	corrected_data[10]	12.719	3.974	8.745	∞	input port clock	
Path 6	∞	2	1	4	data_in[8]	corrected_data[8]	12.617	3.960	8.658	∞	input port clock	
Path 7	∞	2	1	4	data_in[5]	corrected_data[5]	12.597	3.973	8.625	∞	input port clock	
Path 8	∞	2	1	4	data_in[1]	corrected_data[1]	12.541	3.988	8.553	∞	input port clock	
Path 9	∞	2	1	4	data_in[4]	corrected_data[4]	12.433	3.954	8.479	∞	input port clock	
Path 10	∞	2	1	2	data_in[23]	codeword[23]	12.373	3.908	8.465	∞	input port clock	

Figure 6. Proposed Set Up delay for N=32

Name	Slack ^1	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock	Destination Cloc
Path 11	∞	2	1	4	data_in[11]	codeword[43]	2.471	1.753	0.718	-∞	input port clock	
Path 12	∞	2	1	4	data_in[12]	codeword[44]	2.486	1.768	0.718	-∞	input port clock	
Path 13	∞	2	1	4	data_in[13]	codeword[45]	2.489	1.771	0.718	-∞	input port clock	
Path 14	∞	2	1	4	data_in[15]	codeword[47]	2.494	1.787	0.707	-∞	input port clock	
Path 15	∞	2	1	4	data_in[11]	codeword[11]	2.797	1.744	1.053	-∞	input port clock	
Path 16	∞	2	1	4	data_in[12]	codeword[12]	2.814	1.761	1.053	-∞	input port clock	
Path 17	∞	2	1	4	data_in[15]	codeword[15]	2.826	1.783	1.043	-∞	input port clock	
Path 18	∞	2	1	4	data_in[13]	codeword[13]	2.850	1.796	1.053	-∞	input port clock	
Path 19	∞	2	1	4	data_in[14]	codeword[46]	2.937	1.746	1.191	-∞	input port clock	
Path 20	∞	2	1	4	data_in[2]	codeword[34]	3.131	1.778	1.352	-∞	input port clock	

Figure 7. Proposed Hold delay for N=32.

Table 1. Performance comparison of existing and proposed methods for N=32

Metric	Existing Method	Proposed Method	Comparison
LUT	104	6	94%
IO	140	112	20%
TOTAL POWER	34.692 W	17.134 W	50.6%
STATIC POWER	0.301 W	0.165 W	45.18%
DYNAMIC POWER	34.391 W	16.969 W	50.65%
TOTAL DELAY	18.142	14.766	18.6%
LOGIC DELAY	4.522	4.270	5.57%
NET DELAY	13.620	10.496	22.93%

5. Conclusion

The hybridization of LDPC and STBC codes presents a promising avenue for enhancing satellite communication systems, particularly in terms of improving error correction capabilities and mitigating the effects of fading channels. Through the comprehensive review and analysis conducted in this paper, several important insights have emerged.Furthermore, the integration of LDPC-STBC coding within satellite communication systems offers significant practical advantages. By enhancing the robustness of transmissions against channel impairments, such as multipath fading and interference, the proposed scheme enables reliable communication over long-distance satellite links. This is particularly important for applications requiring high data rates and stringent error rate requirements, such as multimedia streaming, remote sensing, and telemedicine.Moreover, the scalability and flexibility of LDPC-STBC coding make it well-suited for future satellite communication standards and deployments. As communication technologies continue to evolve, there is a growing need for efficient and adaptable error control techniques that can accommodate changing channel conditions and network requirements. The inherent parallelism and low-complexity decoding algorithms of LDPC codes, combined with the diversity and coding gain of STBC schemes, make the hybrid approach a compelling solution for next-generation satellite systems.

References

[1] Singh, S. Pratap, Ruchi Rai, Shashank Awasthi, Dinesh Kumar Singh, and M. Lakshmanan. "VLSI Implementation of Error Correction Codes for Molecular Communication." Wireless Personal Communications 130, no. 4 (2023): 2697-2713.

[2] Silva, Felipe, Alan Pinheiro, Jarbas AN Silveira, and César Marcon. "A Triple Burst Error Correction Based on Region Selection Code." IEEE Transactions on Very Large-Scale Integration (VLSI) Systems (2023).

[3] Chu, Shao-I, Syuan-A Ke, Sheng-Jung Liu, and Yan-Wei Lin. "An Efficient Hard-Detection GRAND Decoder for Systematic Linear Block Codes." IEEE Transactions on Very Large-Scale Integration (VLSI) Systems (2023).

[4] Kuo, Yao-Ming, Mark F. Flanagan, Francisco Garcia-Herrero, Oscar Ruano, and Juan Antonio Maestro. "Integration of a Real-Time CCSDS 410.0-B-32 Error-Correction Decoder on FPGA-Based RISC-V SoCs Using RISC-V Vector Extension." IEEE Transactions on Aerospace and Electronic Systems (2023).

[5] Grurl, Thomas, Christoph Pichler, Jurgen Fus, and Robert Wille. "Automatic Implementation and Evaluation of Error-Correcting Codes for Quantum Computing: An Open-Source Framework for Quantum Error Correction." In 2023 36th International Conference on VLSI Design and 2023 22nd International Conference on Embedded Systems (VLSID), pp. 301-306. IEEE, 2023.

[6] Wu, Yujun, Bin Wu, and Xiaoping Zhou. "High-Performance QC-LDPC Code Co-Processing Approach and VLSI Architecture for Wi-Fi 6." Electronics 12, no. 5 (2023): 1210.

[7] Pokhrel, Nabin Kumar, Pankaj Kumar Das, and Aleksandar Radonjic. "Integer codes capable of correcting burst asymmetric errors." Journal of Applied Mathematics and Computing 69, no. 1 (2023): 771-784.

[8] Saini, Madan Lal, Vivek Kumar Sharma, and Ashok Kumar. "An Efficient Single and Double Error Correcting Block Codes with Low Redundancy for Digital Communications." In 2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT), pp. 827-831. IEEE, 2023.

[9] Ponmalar, VJ Beulah Sherin, and M. Ruth Jenila. "Low Density Parity Check used in VLSI Circuits Providing Optimal Circuits."

[10] Boncalo, Oana, and Alexandru Amaricai. "Gradient Descent Iterative Correction Unit for Fixed Point Parity Based Codes." In 2023 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), pp. 1-4. IEEE, 2023.

[11] Dutta, Shruti, Sai Charan Rachamadugu Chinni, Abhishek Das, and Nur A. Touba. "Highly Efficient Layered Syndrome-based Double Error Correction Utilizing Current Summing in RRAM Cells to Simplify Decoder." In 2023 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), pp. 1-4. IEEE, 2023.

[12] Dhandapani, N., M. Z. Mohamed Ashik, Kalthi Reddy Bhargav, N. Achyuth, and Deepa Jose. "VLSI Implementation of BCH Encoder with Triple DES Encryption for Baseband Transceiver." In Mobile Radio Communications and 5G Networks: Proceedings of Third MRCN 2022, pp. 329-341. Singapore: Springer Nature Singapore, 2023.

[13] Ajmal, Muhammad, Masood Ur Rehman, and B. G. Rodrigues. "Improved asymptotically optimal error correcting codes for avoidance crosstalk type-IV on-chip data buses." Computational and Applied Mathematics 42, no. 4 (2023): 150.

[14] Maity, Raj Kumar, Jagannath Samanta, and Jaydeb Bhaumik. "An Improved Single and Double-Adjacent Error Correcting Codec with Lower Decoding Overheads." Journal of Signal Processing Systems (2023): 1-13.

[15] Gracia-Morán, J., L. J. Saiz-Adalid, J. C. Baraza-Calvo, D. Gil-Tomás, and P. J. Gil-Vicente. "Comparison of the overheads provoked by the inclusion of different Error Correction Codes in Embedded Systems."