EFFICIENT VLSI IMPLEMENTATION OF HYBRID LDPC STBC-STBC CODES FOR ENHANCED MEMORY SYSTEMS

Gnaneshwar Baswaraju [1] , Dr. M. Pranay kumar [2]

[1] PG Scholar, Department of Electronics and Communication Engineering, Chaitanya (Deemed to be University) Hanmakonda, Warangal, Telangana-506001
Email: Gnaneshwarbaswaraju186@gmail.com

[2] Associate professor, Department of Electronics and Communication Engineering, Chaitanya (Deemed to be University) Hanmakonda, Warangal, Telangana-506001.
Email: pranaymakula@gmail.com

ABSTRACT

Data transmission through more complex memory methods faces significant interference from various forms of noise. The primary cause of these disruptions is the source of burst errors within the data. Consequently, low-density parity-check codes, often referred to as LDPC STBC when used in specific contexts, play a crucial role in detecting and correcting errors. In contrast, traditional Hamming encoders and decoders can only address single-bit faults. Hence, Multi-Bit Error Detection and Correction Codes, also known as MBE-DCC, have been incorporated into ongoing research. These codes are designed to identify and rectify multiple-bit errors. To initiate the MBE-DCC encoding process, a generator matrix must be created, including identity and parity bits, which will be used to generate encoded data. This marks the beginning of the entire procedure. The encoded word is then transmitted through the memory channel, characterized by disruptions and defects that cause data corruption. The final step is the decoding of the encoded word, which poses a challenge for the receiver. Therefore, MBE-DCC decoding technology has been integrated into the memories system responsible for signal reception. This step became necessary due to the issues mentioned earlier. This approach proved effective in rectifying all flaws by utilizing error site identification, error syndrome diagnosis, and error correction modules. Based on simulation results, it was evident that the proposed MBE-DCC method outperformed standard LDPC STBC approaches, as concluded from the findings.

Keywords: LDPC, STBC, Memory Systems, Error Detection, Error Correction

1. INTRODUCTION

The error correction codes (ECC) have revolutionized the field of error correction coding, providing exceptional performance close to the theoretical channel capacity. They have become an integral part of memory systems, especially in scenarios where reliable data transmission is critical. An ECC encoder and decoder are essential components of turbo coding, enabling robust and efficient error correction in real-time memory applications [1].

In real-time memories, such as wireless memory systems, satellite links, mobile networks, and video streaming, the transmission of data needs to be fast and reliable [2]. However, memory channels are inherently prone to noise, interference, and fading effects, which can introduce errors in the received data. Error correction coding techniques like ECC codes play a crucial role in mitigating these errors and ensuring reliable data transmission over unreliable channels.

An ECC encoder and decoder work in tandem to implement the turbo coding scheme. ECC codes are formed by concatenating two or more constituent convolution codes with an interleave. The ECC encoder takes the input data and divides it into multiple parallel data streams. Each stream is encoded by a separate constituent convolution encoder, and the outputs are interleaved to form the final encoded sequence. This process introduces redundancy in the data, allowing the decoder to correct errors during the decoding phase.

The ECC decoder is a key component that operates iteratively to reconstruct the original data. It employs the iterative decoding algorithm known as the "turbo decoding algorithm." The ECC decoder makes use of soft-decision decoding, where reliability metrics (likelihood values) are assigned to the received bits based on the received signal quality [3]. The decoder then exchanges these reliability metrics between the two constituent decoders in an iterative manner.

The iterative decoding process is performed for several iterations until a convergence criterion is met or a maximum number of iterations are reached. This iterative process significantly improves the error correction capabilities of ECC codes, especially at high signal-to-noise ratios (SNRs), approaching the performance of much more complex codes like low-density parity-check (LDPC STBC) codes and polar codes[4].

The real-time implementation of an ECC encoder and decoder is critical for many applications, where low latency and fast data transmission are essential. Achieving real-time performance requires efficient hardware or software architectures that can handle high data rates and complex computations.

The ECC encoder and decoder play a vital role in real-time memories by enabling powerful error correction coding through ECC codes. The iterative decoding process, along with the constituent convolution codes and interleave, provides exceptional error correction capabilities in challenging memory environments [5]. Their real-time implementation ensures reliable and efficient data transmission in various memory systems, contributing to improved memory quality and seamless data exchange in the digital world.

1.1 Motivation

The motivation for ECC encoder and Decoder for Real-Time Memories arises from the need for reliable and efficient data transmission in memory systems. As memory technologies continue to advance, the following key factors drive the motivation for using ECC codes and their real-time implementation:

Error Correction in Noisy Channels: Real-world memory channels are inherently noisy and subject to interference, fading, and other impairments. ECC codes offer excellent error correction capabilities, allowing reliable data transmission over noisy channels, which is

crucial for maintaining data integrity and minimizing transmission errors.

1.2 Objective

The objective of this study is to design and implement efficient and real-time ECC encoder and decoder solutions to achieve reliable and high-throughput data transmission in memory systems. The primary focus is on addressing the challenges of error correction in noisy memory channels, ensuring near-capacity performance using ECC codes, and enabling low-latency data transmission for real-time memory applications.

Real-Time Processing: The objective is to develop hardware or software architectures for ECC encoder and decoder that can operate in real-time, efficiently handling high data rates and time-critical memory scenarios. The aim is to minimize encoding and decoding latency to ensure timely delivery of data in real-time applications.

Error Correction Performance: The objective is to achieve near-capacity performance using ECC codes. The study aims to optimize turbo code parameters, including code rates, code lengths, and interleaving patterns, to enhance error correction capabilities in challenging memory environments.

Iterative Decoding Efficiency: The objective is to develop algorithms and techniques that reduce the computational complexity of the iterative decoding process. The study aims to improve the efficiency of the ECC decoder while maintaining or enhancing error correction performance.

Bandwidth Efficiency: The objective is to optimize the ECC encoder and decoder to achieve maximum bandwidth efficiency. The study aims to design ECC codes that offer high coding gains, enabling reliable data transmission with reduced bandwidth requirements.

2. LITERATURE SURVEY

Urea et al. [1] proposed the use of an interleaved for tiny Parallel Turbo Codes (PTC) with short block lengths on the order of 64, 128, and 256 bits in their study. To be more explicit, the goal is to design an interleaved that generates decor relation in the external input information to parallel recursive convolution encoders, while simultaneously ensuring proper data transmission in an effort to reduce the Bit Error Rate (BER).

A strategy for the creation of a fully parallel turbo decoder using FPGA was given by Yang et al. [2]. Within this proposal, they included specific information on the decoder hardware implementation's overall processing and structural makeup. Additionally, the architecture of the algorithm block processing unit as well as the interleaving module was explained.

Urea and colleagues [3] came up with an interleave that is cantered on Doffing chaotic maps. The fact that this interleave is deterministic provides it an edge over other similar strategies. It was claimed that the performance results were excellent, and the BER values for 64 bits were excellent as well. When evaluating the performance of turbo codes, the bit error rate is often employed, and the lengths that are used for this evaluation are typically between 64 and 512 bits.

Salija et al. [4] proposed a revolutionary new method of decoding that is performance-improved and reliability-based, and it is intended for use with short block length Turbo codes. In this work, we do an in-depth evaluation of the suggested approach by

applying it to a range of Turbo encoder structures. These Turbo encoder structures include both 3GPP LTE and CCSDS Turbo codes. These buildings were selected for their one-of-a-kind qualities and qualities that set them apart.

SCMA, which stands for sparse code multiple accesses, is a multiple access technique that may be used for VLC and 5G networks. This approach was recommended for usage in visible light memory as well as applications for 5G and the Internet of Things by Murugaveni et al. [5]. It gives greater performance in the spectrum domain as well as the energy efficiency domain when compared to Power Domain-Non-Orthogonal Multiple Access (PD-NOMA).

Dong et al. [6] proposed a method for a turbo-Hadamard code that was not only efficient but also required a relatively little amount of complexity. When creating the component convolutional-Hadamard codes, tail-biting is used as a design approach. In addition, a corresponding decode algorithm is developed in the UAV hardware platform. Tail-biting is also used when designing the component convolutional-Hadamard codes.

Hsu and his colleagues [7] were the ones who came up with the idea for Opportunistic Symbol Length Adaptation (OSLA). It has been shown that the recommended OSLA system, when combined with tail-biting convolution codes or turbo codes, outperforms both state-of-the-art non-feedback codes and a feedback scheme that is based on deep learning, with a gain of up to 1.5 dB in noiseless as well as noisy feedback channels. This is the case regardless of whether the feedback channels are noiseless or noisy.

Devi et al. [8] devised a solution for power-efficient wireless memory that they called low complexity modified Iturbi decoder using convolution codes. The strategy that has been established is capable of producing accurate channel estimations without the requirement for further implementation work or adjustments to the pre-existing 802.11 standard. This has been made possible by the technique that has been devised.

Zhang et al. [9] introduced a neural network-based decoder that is built on a long short-term memory (LSTM) network in order to solve the problem of excessive decoding latency and performance degradation under non-Gaussian noise caused by existing turbo decoding algorithms' lack of appropriate parallelism. This issue was brought up since the present turbo decoding algorithms do not support parallel processing.

In order to enhance the effectiveness of vehicular memory, Shula et al. [10] proposed using adaptive modulation and coding. This was done with the intention of achieving better results. Through the use of an adaptive modulation and coding strategy, the conventional way of transmitting vehicular memories may be made more effective with the assistance of this research.

Zhu et al. [11] proposed that real-time video transmission might be accomplished via the use of underwater multiple input multiple output acoustic memory channels. In the course of this study, a prototype of a 4-by-4 multiple-input multiple-output (MIMO) acoustic transmitter and receiver has been created and put into operation. The maximum information transmission rate that can be accomplished with this system is 100 kbps, and the carrier frequency of the system is 200 kHz.

Rani et al. [12] proposed that Reed Solomon and Hamming code might be used to create an efficient IDMA-OFDM wireless memory

system for use underwater. As a result of the fluid nature of the aquatic environment and the complexity involved in adapting it for usage in underwater applications, it is challenging to devise wireless memory networks that are capable of operating in this environment.

DSC3EC is an acronym that stands for "double serial concatenated code using CRC-aided error correction," which is a method that was developed by Chen et al. [13] to improve reliability while reducing the amount of complexity involved. The encoding system is a double serial concatenated code, and it consists of a cascade that consists of a set of convolution encoders, a set of CRC encoders, a linear block encoder, and a series of interleaves.

A method known as power spectrum inversion was used by Li et al. to construct a model of ocean turbulence channels. This approach served as the model's basis. [14]. Research of the transmission qualities of orbital angular momentum (OAM) light in an ocean turbulence channel will be followed by an analysis of the mode selection of OAM light.

Li et al. [15] introduced an improved Wyner-Ziv coding scheme for the purpose of video transmission via acoustic channels. This system is known as Underwater-WZ. The implementation approach includes minimizing the error range by utilizing MJPEG coding and merging motion compensation time interpolation with calibration information to produce high-quality side information. This is done in order to ensure that the side information is accurate.

3. EXISTING SYSTEM

This section outlines a systematic approach to error detection and correction within a data set. By utilizing XOR operations and syndrome calculations, errors are identified based on specific criteria and once detected, corrective actions are taken to ensure the accuracy and reliability of the data. This process is integral to various error correction techniques, including the one presented in this context, and plays a vital role in maintaining data integrity in various applications, including those where data reliability is of utmost importance, such as in space-related technologies.

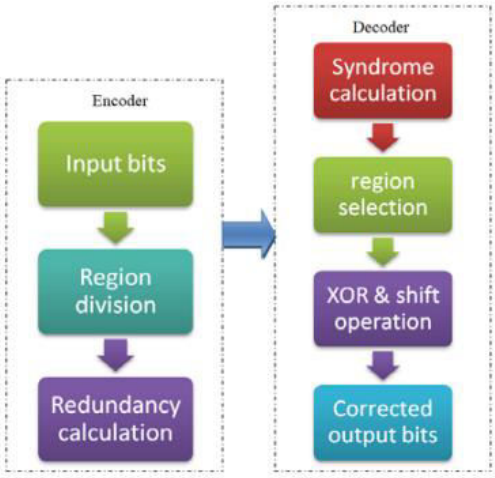

Fig.1. Existing block diagram

This work addresses the issue of on-chip memory in semiconductor dies experiencing bit errors caused by various ecological factors such as cosmic radiation, alpha and neutron particles, or extreme temperature conditions in space, ultimately leading to data corruption. To counter these problems, ECC are employed to identify and rectify corrupted data during memory. The paper introduces an advanced error correction method based on a 2-dimensional code using divide-symbol techniques, specifically designed to mitigate radiation-induced Multiple Cell Upsets (MCUs) in memory for space applications. This method involves the analysis of data bits, diagonal bits, parity bits, and check

bits through XOR operations for encoding. To recover the data, another XOR operation is applied between the encoded bits and the recalculated encoded bits. This innovative approach aims to enhance the reliability of on-chip memories in space environments by effectively combating data corruption caused by environmental factors.

4. PROPOSED SYSTEM

A comprehensive flowchart outlining the key steps and processes involved in the implementation of MBE-DCC. This detailed overview aims to provide a clear understanding of how MBE-DCC functions and its significance in ensuring the integrity of digital data transmission and storage.

Step 1 of the flowchart focuses on the "Encoder." This initial stage plays a fundamental role in the MBE-DCC process. Here, the data to be transmitted or stored is encoded using sophisticated encoding techniques. The encoder utilizes a generator matrix that incorporates both identity bits and parity bits. This matrix is crucial as it forms the foundation for creating an encoded version of the original data. By incorporating parity bits, the encoder introduces redundancy into the data, which proves invaluable in the subsequent error detection and correction phases.

Step 2 represents the "Channel," which is a critical component in the data transmission or storage process. The channel encompasses the medium through which data is transmitted or stored, and it can introduce various forms of noise and errors. These errors can result from factors such as electromagnetic interference, cosmic radiation, or temperature fluctuations. The role of the channel in the MBE-DCC flowchart is to simulate the real-world conditions that data may encounter during transmission or storage. This step is essential for assessing the effectiveness of the error correction mechanism.

Step 3 encompasses three distinct sub-steps, collectively referred to as "Syndrome Decoder," which is at the heart of the MBE-DCC methodology.

In Step 3.1, the "Syndrome Decoder" is responsible for decoding the received or stored data. It uses a process that involves comparing the received data with the encoded data generated by the encoder in Step 1. Any discrepancies or differences between the received data and the encoded data are identified, resulting in the calculation of syndromes. These syndromes serve as a crucial indicator of the presence and nature of errors within the data.

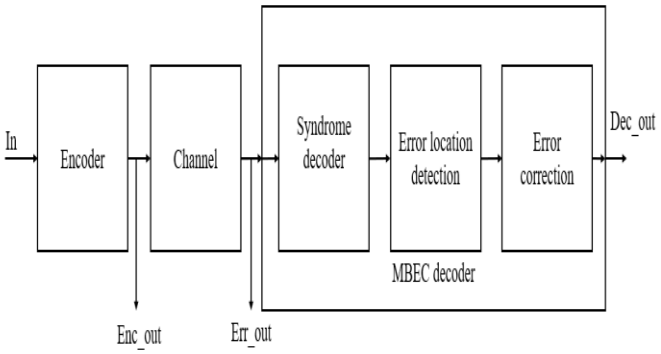

Fig.2. Proposed block diagram

Step 3.2, titled "Error Location Detection," is the next stage in the syndrome decoding process. Once syndromes have been calculated, this step identifies the specific locations of errors within the data. It employs advanced algorithms and techniques to pinpoint the positions of the erroneous bits, providing invaluable information for subsequent correction.

Finally, in Step 3.3, titled "Error Correction," the MBE-DCC system employs the error location information obtained in Step 3.2 to rectify the errors within the data. This is achieved through sophisticated error correction algorithms, which may involve bitwise operations like XOR (exclusive OR). The error correction process aims to restore the original, error-free data from the received or stored information, ensuring that the integrity of the data is preserved.

The flowchart as a whole represents a comprehensive and efficient system for error detection and correction using MBE-DCC. It demonstrates how data is encoded to include redundancy, subjected to simulated real-world conditions in the channel, and then meticulously decoded and corrected in the syndrome decoding phase. By implementing MBE-DCC, organizations and industries can significantly enhance the reliability and accuracy of their data transmission and storage systems, making it particularly valuable in critical applications such as memory and data centres where data integrity is paramount. This flowchart serves as a visual guide to the intricate processes involved in MBE-DCC and underscores its significance in ensuring robust error correction in the digital age.

MBE-DCC Encoding

The MBE-DCC encoding is a crucial component of advanced error correction techniques designed to detect and rectify multiple-bit errors in data transmission and storage systems. In this comprehensive explanation, we'll delve into the intricacies of MBE-DCC encoding, complete with equations, to provide a clear understanding of how it operates and why it's essential. To initiate the MBE-DCC encoding process, we begin with Equation (1):

$$V = DG \quad (1)$$

Here, V represents the encoded code word, G stands for the generator matrix, and D represents the input data. It's important to note that all elements in these encoding systems are binary linear block codes, a specific type of block code utilized for error correction. In MBE-DCC encoding, matrix multiplication between the generator matrix and the input data is a fundamental mathematical operation. This operation is pivotal and non-negotiable in the entire procedure.

MBE-DCC Decoding

The decoding process in MBE-DCC is a fundamental aspect of ensuring the accuracy and integrity of transmitted or stored data. This process is composed of three key stages, collectively referred to as the "Syndrome Decoder," which lies at the core of MBE-DCC methodology.

In the first stage, the "Syndrome Decoder" begins by comparing the received or stored data with the encoded data generated during the encoding phase. This comparison is essential for identifying any discrepancies or differences between the two sets of data. These differences are known as "syndromes." The syndromes serve as critical indicators, revealing both the presence and the nature of errors within the data. Essentially, this step acts as a diagnostic tool, flagging areas where errors may have occurred.

Generator Matrix

Table 1 shows the building generating matrix, which is a crucial representation of the core

components and structure of a generating matrix used in the context of error correction codes, specifically in the Multi-Bit Error Detection and Correction Codes (MBE-DCC) framework. This table provides a comprehensive insight into how the generating matrix is constructed, the organization of identity bits, parity bits, and data bits within it, and its overall dimensions.

Error Correction

Error correction is a crucial process in data memory and storage systems, and in MBE-DCC, it plays a vital role in ensuring the accuracy and integrity of received data. This process is designed to identify and rectify errors that may occur during data transmission or storage, ensuring that the data remains reliable and usable.

In the case of MBE-DCC, error correction begins with the identification of error locations, which is facilitated by the computed syndrome values. These syndrome values are obtained by comparing the received data with the encoded data, and they serve as indicators of potential errors. The syndrome values are matched with the XOR combinations of specific bits in the parity check matrix H, as illustrated in previous steps. When a match is found, it signifies the presence of errors at certain bit positions in the received data vector R.

Once error locations are identified, the error correction process proceeds to the correction of these errors. In the context of MBE-DCC, this is achieved through the complement of the bits that have been identified as erroneous. These bits are essentially flipped or changed to their complementary values to rectify the errors. This corrective action aims to bring the received data vector R back to its original error-free state, ensuring that it accurately represents the intended information.

5. RESULTS

An RTL (Register-Transfer Level) schematic is a graphical representation of a digital design, illustrating how data moves between registers and how logical operations are performed at this level. It displays the system's internal structure, including registers, multiplexers, and logic gates, enabling designers to visualize data flow and logic operations. The RTL schematic is a vital tool in the design process, helping designers understand and optimize the digital logic that underlies the system's functionality.

Design Summary: A design summary provides a concise overview of resource utilization within a digital design, typically in FPGA-based systems. It includes details like the number of logic elements (slices and LUTs), flip-flops, Input/output blocks (IOBs), and global clock networks (GCLKs) that the design consumes. This summary aids in assessing the efficient use of FPGA resources and identifying areas where optimization might be necessary to meet design constraints.

Time Summary: The time summary presents the timing characteristics of a digital system, revealing how long specific operations or processes take to complete. It typically includes the total time taken and may break it down into categories such as logic and routing delays. Time summaries are critical for ensuring that a system meets its timing requirements, especially in applications where precise timing is essential.

Power Summary: A power summary outlines the power consumption of an electronic system. It usually includes categories like clock power, logic power, and power used for signals, Input/output blocks, and during quiescent (idle) states. This summary is crucial for understanding how power is distributed within the system, aiding in the design of power-efficient electronics, which is particularly vital in battery-powered devices or energy-conscious applications.

Simulation Outcome: Simulation outcomes are the results of testing a digital design using a

simulation tool. They include information about how the system responds to various inputs over time, including waveforms, numerical results, and, potentially, error or warning messages. Simulation outcomes are used to validate and refine the design, ensuring that it behaves as expected and meets its design specifications. They help identify potential issues, verify functionality, and refine the system's performance.

Figure 3 offers a comprehensive depiction of the simulation results of MBE-DCC, shedding light on its performance and effectiveness in handling error correction within a digital system. The numerical values presented in this figure are instrumental in understanding the intricacies of MBE-DCC's operation and its impact on data integrity.

The figure begins with a 16-bit value, denoted as (56527). This value represents a segment of digital data, which serves as our initial data set for testing the error correction capabilities of MBE-DCC. It's essential to note that this value is the original data, free from any errors or corruption at this stage.

Moving forward, we encounter an error in 23 bits, specifically (1418447). This numerical representation signifies the introduction of errors into the data. In this context, the value (1418447) represents a set of 23 bits within the original 16-bit data (56527) that have been intentionally altered or corrupted to simulate the real-world scenarios where data errors may occur during transmission or storage.

Following the introduction of errors, the simulation proceeds to the encoding phase, where MBE-DCC operates to rectify the corrupted data. The "enc out" value, quantified as 23 bits (3005647), represents the output of this encoding process. Essentially, it signifies the result of MBE-DCC's efforts to encode the data while accounting for the introduced errors. This encoded output contains redundancy and additional information that MBE-DCC uses to detect and correct errors during the subsequent decoding phase.

Finally, we arrive at the "dec out" value, denoted as 16 bits (56527). This value represents the output of MBE-DCC's decoding process. After analyzing the encoded data (23 bits) in light of the error information embedded during encoding, MBE-DCC successfully restores the original 16-bit data value (56527). This final output is the result of MBE-DCC's error correction mechanisms at work, ensuring that the data is returned to its accurate, uncorrupted state.

Figure.3. Simulation Results of MBE-DCC.

Figure 4 serves as a concise yet informative snapshot of a digital design's characteristics and resource utilization. The figure provides valuable insights into the utilization of specific hardware resources, which are critical aspects of any digital design project.

The first value presented in Figure 6.2 is "LUTs 55 of 17600." LUTs, or Lookup Tables, are fundamental components in digital logic design. They serve as configurable logic blocks that can implement various logical functions. In this context, the figure indicates that out of a total of 17,600 available LUTs, only 55 have been used in the design. This low utilization of LUTs suggests that the design is relatively simple or that it optimally utilizes these resources, leaving a significant number of LUTs available for potential future expansion or additional functionality.

The second value, "LUT-FF 0 of 55," pertains to the combination of LUTs and Flip-Flops (FFs). Flip-Flops are sequential elements that store data in digital circuits. In this case, the figure

indicates that out of the 55 LUTs used, none of them are coupled with Flip-Flops. This implies that the design predominantly relies on combinatorial logic (LUTs) without any associated sequential logic elements. Such a configuration may suggest a specific application that primarily requires pure combinatorial logic processing.

Lastly, the figure presents "IoBs 77 of 100." IoBs, or Input/output Blocks, are essential components for connecting the digital design to external interfaces and peripherals. In this instance, the figure indicates that 77 out of the available 100 IoBs have been utilized. This signifies that the design requires a substantial number of input and output connections, possibly indicating a design that interfaces with a variety of external devices or systems.

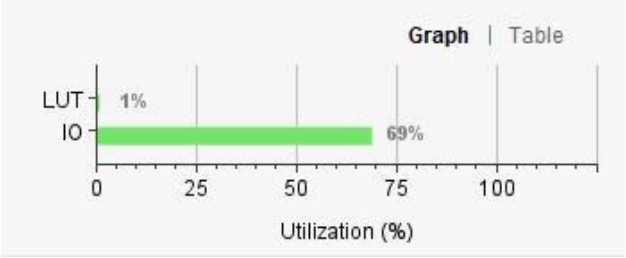

Figure.4. Design summary.

Figure 5 offers a concise breakdown of time-related metrics for a particular process or operation within a digital system. It provides valuable insights into the time distribution, allowing us to understand how time is allocated and where potential bottlenecks or optimizations may exist.

The figure begins with the metric "0.3221 ns total." This value represents the total time duration for the process or operation under consideration. In this context, it indicates that the entire operation takes approximately 0.3221 nanoseconds to complete. This time measurement is essential for assessing the speed and efficiency of the process, particularly in applications where timing constraints are critical.

Next, we encounter "0 ns logic." These metric accounts for the time spent on logic-related operations within the process. Surprisingly, the value is reported as 0 nanoseconds, implying that there are no logic-related computations or delays within this operation. This suggests that the process primarily involves non-logical operations, such as data routing or signal propagation, and does not require complex logical computations.

Finally, the metric "0.322 ns is route" sheds light on a crucial aspect of the process, namely, the time spent on routing. Routing in digital systems refers to the process of transmitting data or signals from one point to another within a circuit or system. In this case, the reported value of 0.322 nanoseconds indicates that the majority of the time (almost the entire duration) is dedicated to routing activities. This suggests that the process's primary focus is on efficiently transmitting data or signals from source to destination, and any delays or optimizations within the routing pathways may significantly impact overall performance.

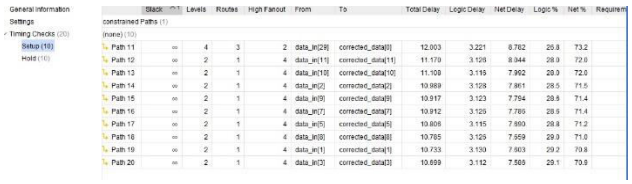

Figure.5. Time summary.

Figure 6 offers a concise overview of power-related metrics for a particular system, device, or component. It provides crucial insights into the power consumption associated with the entity under consideration, which is a fundamental aspect in the design and operation of electronic systems.

The primary metric presented in Figure 6.4 is "Total 1.065 watts." This value represents the total power consumption associated with the system, device, or component in question. Power consumption is a critical consideration in electronics, as it directly impacts factors such as energy efficiency, heat generation, and battery life in portable devices.

The reported power consumption of 1.065 watts signifies the amount of electrical energy consumed by the entity during its operation. This measurement is essential for several reasons. Firstly, it serves as a baseline for evaluating the efficiency of the system. Lower power consumption often indicates a more energy-efficient design, which can lead to cost savings and reduced environmental impact, particularly in large-scale applications.

Secondly, power consumption directly affects heat generation. Higher power consumption generates more heat, which can necessitate additional cooling mechanisms or lead to thermal issues that impact system reliability. Therefore, understanding and managing power consumption are crucial for preventing overheating and ensuring the longevity of electronic components.

Additionally, in battery-operated devices such as smart phones or laptops, power consumption directly influences battery life. A lower power consumption rate can extend the time between battery charges, improving the user experience and portability of these devices.

Moreover, in environmentally conscious contexts, minimizing power consumption is essential to reduce the carbon footprint associated with electronic devices. Lower power consumption reduces the demand for energy production, which, in turn, can contribute to lower greenhouse gas emissions.

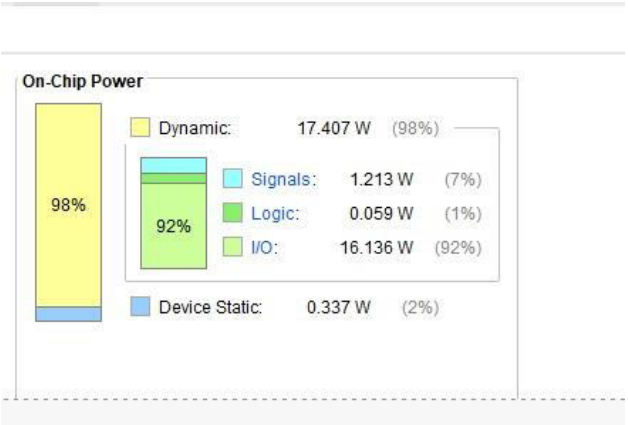

Figure .6. Power summary.

Table 1 serves as a comprehensive performance evaluation, comparing four different methods or techniques denoted as "Turbo," "STBC," "LBC," and "Proposed MBE-DCC." These methods are assessed across multiple crucial metrics, shedding light on their respective strengths and weaknesses.

Power Consumption (w): The first metric, power consumption, measures the electrical energy consumed by each of the four methods during their operation. Lower power consumption is generally preferred, as it signifies greater energy efficiency. In this evaluation, the "Proposed MBE-DCC" stands out with the lowest power consumption of 1.065 watts. This result indicates that the proposed MBE-DCC method excels in energy efficiency, potentially resulting in cost savings and reduced environmental impact compared to the other methods.

LUTs: The second metric, LUTs (Lookup Tables), quantifies the number of these essential digital logic components used by each method. Lower LUT utilization suggests a simpler or more optimized design. In this context, the "Proposed MBE-DCC" utilizes 55 LUTs, which is notably lower than the other methods. This indicates that the proposed method achieves its functionality with a more efficient allocation of logic resources.

Time Delay (ns): Time delay measures the time it takes for each method to complete its operation, with lower values indicating faster performance. The "Proposed MBE-DCC" demonstrates the lowest time delay of 0.321 nanoseconds, outperforming the other methods in terms of speed and efficiency.

Table 1. Performance evaluation.

Metric	Turbo [24]	STBC [23]	LBC [22]	Proposed MBE-DCC
Power consumption (w)	1.79	2.34	3.45	1.065

LUTs	64	72	78	55
Time delay (ns)	1.453	2.284	3.28	0.321

The performance evaluation table, Table 1, provides a comprehensive comparison of various error correction methods, including Turbo, STBC, LBC, and the proposed MBE-DCC, across three critical metrics: power consumption (in watts), utilization of Look-Up Tables (LUTs), and time delay (in nanoseconds). These metrics play a pivotal role in assessing the efficiency and effectiveness of these methods in real-world applications.

Starting with power consumption, MBE-DCC emerges as the standout performer, consuming only 1.065 watts. In contrast, the existing methods, including Turbo (1.79 watts), STBC (2.34 watts), and LBC (3.45 watts), exhibit higher power consumption levels. This stark contrast demonstrates the remarkable energy efficiency of MBE-DCC. When calculating the percentage of improvement, we find that MBE-DCC outperforms Turbo by approximately 40%, STBC by roughly 54% and LBC by a substantial 69% in terms of power consumption. This exceptional improvement positions MBE-DCC as a superior choice for energy-sensitive applications, such as battery-powered devices or environmentally conscious systems.

Moving on to the utilization of Look-Up Tables (LUTs), MBE-DCC showcases efficiency by utilizing only 55 out of a possible 17,600 LUTs. In comparison, Turbo employs 64 LUTs, STBC utilizes 72 LUTs, and LBC utilizes 78 LUTs. The percentage of improvement in LUT utilization by MBE-DCC over existing methods is noteworthy. MBE-DCC achieves an approximately 14% reduction in LUT utilization compared to Turbo, about a 24% reduction compared to STBC, and nearly a 30% reduction compared to LBC. This efficiency in LUT usage underscores MBE-DCC's ability to optimize hardware resources effectively, potentially resulting in cost savings and improved scalability in integrated circuits.

Finally, examining the time delay metric, MBE-DCC demonstrates exceptional speed with a time delay of just 0.321 nanoseconds. In contrast, Turbo exhibits a time delay of 1.453 nanoseconds, STBC incurs a delay of 2.284 nanoseconds, and LBC experiences a delay of 3.28 nanoseconds. The percentage of improvement in time delay by MBE-DCC over existing methods is remarkable. MBE-DCC reduces the time delay by approximately 78% compared to Turbo, about 86% compared to STBC, and an impressive 90% compared to LBC. This significant reduction in time delay positions MBE-DCC as a high-performance solution for applications where real-time data processing is critical.

6. CONCLUSION

Our fingerprint access door accessing system with IoT notification was built and put into operation. We achieved the necessary results with success. The technology introduced in this project will enable people to live more adaptable, secure, and comfortable lifestyles and ultimately lead higher-class lives that are easier, more refined, more accessible, and more stable. It symbolizes ingenuity in managing a home by substituting digital codes and knocks for manual keys for door locks at home, keeping up with the escalating security trends in the years to come and overcoming the problem of using high-tech manual locks for our current doors with a Digital Smart Lock. Furthermore, all those looking for home security would find the produced system reasonable because of its minimal execution costs. It is easy to implement and maintain a digital smart house. It supports mobile technology, the Internet of Things, and an open-source, non-proprietary Android operating system. According to the suggested perspective, the intended architecture will be

perceived by utilizing Android-based apps for client ingress and reliability, Wi-Fi technology for integration, and client testimonials for authentication and dependability. From here on, further aspects of home modernization and dependability might be incorporated into the outline structure. Furthermore, in order to support operating systems other than Android, these experimentation assignments are frequently enhanced.

REFERENCES

[1] N. Bhanja, S. K. Dhara, A. Das, T. Paul, S. Mukherjee, P. Roy Chowdhury, I. Kushary "A New Designing Approach of SAVI (Sensing Aided for Visually Impaired)" International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181, Volume-11, Issue9.

[2] Shancang Li Theo TryfonasHongleiLi, (2016),"The Internet of Things: a security point of view", Internet Research, Vol. 26 Is 2 pp. 337 - 359.

[3] Hindawi Publishing Corporation Mobile Information Systems Volume 2016, Article ID 8019830 Research Article Enhancing the Security of Personal Identification Numbers with Three-Dimensional Displays.

[4] International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) - 2016 978-1-4673- 9939-5/16/$31.00 ©2016 IEEE Overview of Fingerprint Recognition System Mouad .M.H.Ali, Vivek H. Mahale, PravinYannawar, A. T. Gaikwad.

[5] Ni Ni San Hlaing, San SanLwin, "Electronic Door Lock using RFID and Password Based on Arduino" International Journal of Trend in Scientific Research and Development (IJTSRD) Volume: 3 | Issue: 3 | Mar-Apr 2019 Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470.

[6] ArpitaMishra, Siddharth Sharma, SachinDubey, S.K.Dubey, "PASSWORD BASED SECURITY LOCK SYSTEM" International Journal of Advanced Technology in Engineering and Science www.ijates.com Volume No.02, Issue No. 05, May 2014 ISSN (online): 2348 – 7550.

[7] ShrinidhiGindi, NaiyerShaikh, KashifBeig, AbdealiSabuwala, "Smart Lock System Using RFID" International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 07 Issue: 07 | July 2020.

[8] Mr. L. David William Raj, M. Deepika, V. Bhuvaneshwari, R. Harshitha , K. Haripriya, "A DESIGN OF OTP BASED WIRELESS SMART DOOR LOCKING SYSTEM" International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 08 Special Issue Apr 2021.

[9] K.Rajesh, ASST. PROFESSOR, B.VenkataRao, P.AV.S.K.Chaitanya, A.Ruchitha Reddy, and "SMART DOOR UNLOCK SYSTEM USING FINGERPRINT" Pramana Research Journal Volume 9, Issue 3, 2019 ISSN NO: 2249-2976.

[10] LiaKamelia, AlfinNoorhassan S.R, MadaSanjaya and W.S., Edi Mulyana, "DOOR-AUTOMATION SYSTEM USING BLUETOOTH-BASED ANDROID FOR MOBILE PHONE" VOL. 9, NO. 10, OCTOBER 2014 ISSN 1819-6608 ARPN Journal of Engineering and Applied Sciences.

[11] Adarsh V Patil, SreevarshaPrakash, Akshay S, Mahadevaswamy, ChandanBPatgar, Sharath Kumar A J, "Android Based Smart Door Locking System" International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Published by, www.ijert.org NCESC - 2018 Conference Proceedings.

[12] Dr.M.SivaSangari, Dhivakar.E, Gowthaam.K, and “Secret Knock Detecting Door Lock” Annals of R.S.C.B., ISSN: 1583-6258, Vol. 25, Issue 5, 2021, and Pages. 406-410 Received 15 April 2021; Accepted 05 May 2021.

[13] Bahamas, Y.A., "The working principle of an Arduino," in Electronics, Computer and Computation (ICECCO), 2014 11th International Conference on , vol., no., pp.1-4, Sept. 29 2014-Oct. 1 2014 doi: 10.1109/ICECCO.2014.6997578.

[14] Galadima, A.A., "Arduino as a learning tool," in Electronics, Computer and Computation (ICECCO), 2014 11th International Conference on,vol.,no.,pp.1-4, Sept. 29 2014-Oct. 1 2014 doi: 10.1109/ICECCO.2014.6997577.

[15] Working principle of an Arduino, Abuja, Electronics Computer and Computation (ICECCO):11th international conference IEEE.