

Original Article

Design Approaches for SECDED ECC Architectures in Automotive Systems

VenkataKrishna Brahmam Pogadadanda

Senior Silicon Design Engineer at AMD Xilinx, India.

Received: 08-03-2022

Revised: 08-26-2022

Accepted: 09-02-2022

Published: 09-04-2022

ABSTRACT

With vehicles nowadays heavily relying on electronic control systems for their operation, securing the reliability and safety of automotive memory architectures has become a major design challenge. In fact, the different automotive features like Advanced Driver Assistance Systems (ADAS), autonomous driving platforms, engine control units, and in-vehicle infotainment do normally function in the very harsh vehicular environments where radiation-induced soft errors, noise, and hardware faults could disrupt their operability. Especially in safety-critical environments, a single bit error in the memory might not only affect the system's normal operation or reduce its performance but even cause a tragedy. The common response to the aforementioned problems is the use of fault-tolerant design techniques, and among these, the role of Single Error Correction and Double Error Detection (SECDED) Error Correction Code (ECC) structures in enhancing memory reliability and data integrity cannot be underrated. This paper showcases a variety of design techniques of SECDED ECC architectures for automotive systems, mainly concentrating on leveraging reliability, hardware complexity, latency, and power consumption. The paper discusses different encoding and decoding options, parity generation methods, syndrome computation techniques, and architectural improvements for automotive embedded memories. Conducted experiments continue to focus on fault coverage, as well as on the costs associated with the implementations and performances expected for automotive scenarios with stringent real-time requirements. Introduced solutions preserve compactness and high operating speed and at the same time, they improve error detection and correction abilities which make them more than suitable for next generation automotive electronics. The results further confirm that well-designed SECDED ECC structures do a great job in reinforcing system failure resilience and supporting the fulfillment of automotive safety standards such as ISO 26262. Basically, this study is a step towards the design of dependable and efficient memory protection mechanisms for reliable automotive computing systems.

KEYWORDS

SECDED ECC, Automotive Systems, Error Correction Codes, Functional Safety, Memory Reliability, Fault-Tolerant Architectures, FPGA/ASIC Design, ISO 26262.

1. INTRODUCTION

1.1. Background

The automotive sector has been rapidly changing for the last ten years due to the integration of sophisticated electronic systems in new vehicles. While in the past cars were mainly composed of mechanical parts that were complemented by some electronic functionalities, nowadays a single car can have hundreds of electronic control units (ECUs), embedded processors, sensors, and communication interfaces to perform various functions. Advanced Driver Assistance Systems (ADAS), self-driving, adaptive cruise-control, infotainment systems, battery management systems, and vehicle-to-vehicle communication are just a few examples of technologies that significantly increased the reliance on electronic hardware and embedded memory subsystems. In fact, reliable data storage and processing are must-haves in automotive computing environments.

Today's automotive systems rely heavily on communication and data exchange. The main storage components of an embedded system include SRAM, DRAM, Flash memory, and cache units. These types of memories are, however, vulnerable to transient faults and soft errors caused by cosmic rays, electromagnetic interference, temperature, and voltage fluctuations, etc. The presence of such faults can modify data bits, thereby causing data corruption, ECU malfunction, or even unpredictable system behavior. A memory error alone can pose a serious risk to safety of passengers and reliability of vehicle in safety-critical automotive applications.

One of the most common methods of dealing with these problems is through the use of Error Correction Codes (ECCs) that can detect and fix faults in the memory. To implement ECCs, there are different techniques that one can use. Among them, Single Error Correction and Double Error Detection (SECCDED) ones are most often used by the industry since on the one hand, they make the system reliable and on the other hand, they are not too complicated or expensive to produce. With SECCDED ECCs automotive systems get the possibility to detect double-bit errors and to correct single-bit errors faultlessly, thus increasing memory integrity and safety of operations in modern ECUs.

1.2. Challenges in Automotive ECC Design

One of the issues that makes it difficult to develop efficient ECC architectures for automotive systems is technical problems and the surrounding environment. The car's electronic control systems will be exposed to environmental extremes, such as very high or low temperatures, EMI, mechanical vibrations, humidity, and radiation. Possibilities for random faults and soft errors in semiconductor memories also arise with such conditions as they greatly increase the probabilities of errors; therefore, highly fault-tolerant and robust mechanisms that are capable of withstanding environmental changes for reliable operations are a must. On top of that, memory cells become more radiation-induced bit flips and charge leakage effects susceptible to the effects of errors when memory sizes are scaled down during the manufacturing process, which leads to further increase in the errors.

Meeting a high level of operational real-time requirements remains a big challenge. Functions of a car like brakes, steering, collision detection, and navigation will have to process sensor data on the spot with very little latency. ECC architectures that delay encoding and decoding operations to a great extent might end up degrading system performance and responsiveness. As a result, the need for faster error detection and correction that also retains low latency remains a critical design target.

The foremost concerns with implementation of ECC in automotive applications are power consumption and size of hardware. Adding parity bits and performing syndrome computation and decoding require more logic gates, which will increase the area and power consumption of the chip. As a result, the final system cost will be higher and thermal efficiency of the system will be compromised. Therefore, car manufacturers generally prefer fault-tolerant ECC techniques that do not lead to a substantial increase in hardware complexity.

Besides, compliance with automotive functional safety standards such as ISO 26262 requires electronic systems to have a very high level of reliability. ECC systems must provide dependable fault detection and should happen to operate in a predictable manner even when faults occur. As a result, the combination of reliability, area efficiency, power, and timing performance is still a major challenge in the automotive SECDED ECC design for future embedded systems.

1.3. Problem Statement

Most of the standard ECC implementations that are used in various embedded systems fall short of the stringent requirements that automotive applications demand, especially when it comes to performance optimization. Earlier SECDED architectures mainly centered around the error-correcting feature only without giving due weightage to the constraints like lower latency, power consumption, hardware size, and so on. Now with the increased complexity and data volume of automotive systems, these constraint issues are posing difficulties in simultaneously ensuring system reliability and real-time operations performance.

Besides that, present ECC methods also compromise heavily on one or more aspects of the reliability, chip area, complexity of calculation, and speed of decoding. Sturdy ECC methods may call for more redundancy bits, and decoding may even become a non-trivial task leading to more power consumption and thus, more area utilization. Conversely, lightweight ECCs might not be upgraded enough to multiple fault coverages in the case of safety-critical automotive systems which are worse even going through very adverse environmental conditions and TTC radiation-induced off errors.

Hence, a demand is progressively being built up for SECDED ECC architectures that are fully optimized and tailored from the ground up for the automotive market. These architectures not only should be capable of performing efficient single-bit correction and double-bit detection but also must be designed in such a way that they produce the minimum possible levels of latency, complexity of implementation, and consumption of energy. Developing automotive ECC solutions able to meet the

requirements of functional safety along with performance constraints is still an open research problem in the field of embedded system design.

1.4. Motivation

Dependent electronics have gained more importance as driver assistance systems (ADAS) are being incorporated in numerous car designs alongside self-driving technologies, electric vehicles, and automotive platforms based on artificial intelligence. Modern cars rely heavily on high-speed processors, embedded memories, sensors, and communication networks that all work together to perform complex real-time tasks. Since these systems are directly responsible for vehicle safety and reliability, the importance of designing automotive memory components that can be cleared of transient faults and soft errors has become a top priority in the industry.

With the ongoing shrinking of semiconductor devices to smaller nanometer technologies, memory cells are increasingly susceptible to radiation-induced bit flips, noise disturbances, and environmental variations. Such faults may lead to the corruption of essential data and, as a result, to malfunction of safety systems, erratic behavior of ECUs, or even system crashes. In view of this, automotive companies are looking for strong and fault-tolerant memory protection methods that can guarantee data integrity even in extreme operating conditions.

Another very important reason for producing this paper is that the request for low-latency, highly energy-efficient error correction code (ECC) architectures is constantly increasing. Most of the time, due to their imposed timing constraints, car systems are in a need for very fast detection and correction of errors for them to be responsive in real-time situations. On the contrary, reducing hardware resources and power consumption is of a high priority for improving thermal efficiency and lowering production costs at the same time.

In fact, even though SECDED ECCs are standard memory protection methods, very few studies have actually looked into how to adapt these architectures optimally specifically to the automotive context. The main goal of this paper is to fill this gap by figuring out design methods of SECDED that are capable of balancing reliability, performance, hardware efficiency, and acquiring the necessary functional safety levels for present automotive solutions.

2. LITERATURE REVIEW

2.1. Overview of Error Correction Codes

Two main purposes of ECC (Error Correction Codes) in digital systems are to provide data integrity assurance and to increase the system reliability during storage and communication. ECC methods provide ways of detecting and correcting errors that may be introduced by noise, radiation, interference, or even faulty hardware. To list just one type of the very simple coding, parity codes are widely used for detecting a single-bit error. In parity coding, an extra parity bit is attached to the data word which shows whether the number of logical ones is even or odd. Being simple and minimal in

hardware requirement, the parity codes fail, at the same time, to correct the errors and they aren't guaranteed to detect multiple-bit faults.

In single-bit error correction, Hamming codes are still one of the regularly used ECC methods. The redundancy of parity bits is exploited in these codes and placing them at the designated positions within the data word makes it possible to identify and correct single-bit errors. Adding to the double-error detection capability, extended Hamming codes have increased reliability and this is known as SECDED functionality. BCH (Bose, Chaudhuri, Hocquenghem) codes are capable of correcting multiple random bit errors thus can be used in communication systems as well as high-reliability applications. On the other hand, BCH decoding usually demands complicated hardware and a lot of computational power.

Reed, Solomon codes represent a category of error-correcting codes (ECC) that operate on discrete blocks of data by working with symbols rather than bits. One of the high points of these codes is their ability to handle burst errors efficiently. The use of Reed, Solomon codes is quite common in various areas such as data storage, wireless communication, and satellite systems. However, a downside is that the Reed, Solomon structures significantly increase both the delay and the hardware requirements.

Out of the different ECC techniques, SECDED stands out as the most reasonable one for embedded automotive memory systems in terms of trade-off among reliability, complexity, and costs of implementation. By design, SECDED schemes are capable of correcting one-bit errors and detecting two-bit errors which make them perfect for automotive embedded memory systems where operation under faults, and meeting live performance requirements have been identified as key concerns.

2.2. SECDED ECC Architectures

SECDED ECC architectures are the first choice in embedded systems to enhance memory reliability and provide fault-tolerant mechanisms against transient errors. Typically these architectures consist of an encoder and a decoder. The encoder unit is responsible for producing parity bits using predefined parity-check equations. It then stores or transmits parity bits along with the data word that allows error monitoring during future read operations.

The main task of a decoder is to pinpoint the parts of the memory that are wrong and fix the errors. At the time when data is taken out of memory, a decoder works out parity data again and compares it with the stored parity bits. The result of this comparison is a syndrome vector. In SECDED ECC methods, generating the syndrome is crucial as it discloses the location and the nature of the error in the data word. When the syndrome points to the location of a valid bit, the decoder corrects the single-bit error by simply changing the bit. However, if multiple-bit errors occur, a SECDED system will flag such an error as an uncorrectable double-bit error and will issue an error signal.

Many hardware design methods to enhance the effectiveness of SECDED ECC have been put forward. Encoding and decoding become faster with the logic-only implementation but it often comes with the cost of greater area filling. Pipeline schemes raise the system's data processing rate by breaking down ECC tasks into stages that reduce the overall delay issue, a major factor in the case of very-high-speed devices. The designs based on FPGAs have the advantage of being flexible and the implementation relatively quick, however, ASIC designs are more suitable to lower power usage and get silicon utilized in a more efficient way.

Contemporary SECDED scheme designs have also become equipped with low latency syndrome computations, the ability to generate parities in parallel, and decoding algorithms that have been fine-tuned to ensure that they remain highly reliable while satisfying real-time automotive electronic systems performance requirements.

2.3. ECC in Automotive Applications

ECC systems play a very critical role in ensuring that the electronic systems in vehicles are reliable and safe. The latest car models rely heavily on the embedded memories located in Electronic Control Units (ECUs), processors, sensors, and communication modules to perform their functions live. Safety-critical data processing car applications are like ADAS, autonomous driving systems, braking control, steering systems, and battery management units that wander continuously among other things. A mistake in the memory content can lead to wrong decision-making, system instability, or even the vehicle behaving dangerously.

Therefore, car makers use ECC designs extensively in their memory components like SRAM, DRAM, cache, and Flash memory. Among various ECC schemes, the one that is used widely in the industry is SECDED because it gives the ability to correct single-bit errors and detect double-bit errors effectively with a relatively small hardware cost. Such capabilities are important in preserving data integrity in the environments that are prone to radiation-induced soft errors, electromagnetic interference, temperature variations, and power supply disturbances.

Additionally, it is necessary for automotive systems to follow functional safety standards like ISO 26262 that are very strict and require reliable fault detection and protection mechanisms. Therefore, it is these ECC designs that play a major role in meeting the Automotive Safety Integrity Level (ASIL) criteria. The trend is that the automotive processors and microcontrollers of today are equipped with hardware ECC modules that help to improve reliability while also preserving low latency and high-speed performance in real-time vehicle operations.

3. PROPOSED METHODOLOGY

3.1. System Architecture

This paper presents a method for a reliable and efficient Single Error Correction and Double Error Detection (SECDED) Error Correction Code (ECC) implementation suitable for automotive memory systems. The whole concept is aimed at equipping automotive electronic environments with

fault-resilient memory thus protecting against memory transient faults, soft errors, and bit flips through radiation exposure. The design involves a CODING and DECODING phase combining embedded memory units such as SRAM, DRAM, cache memory, and automotive Electronic Control Units (ECUs). The main goal is to guarantee data integrity without compromise in terms of low latency, hardware overhead reduction, and high-speed performance that are essential for automotive real-time applications.

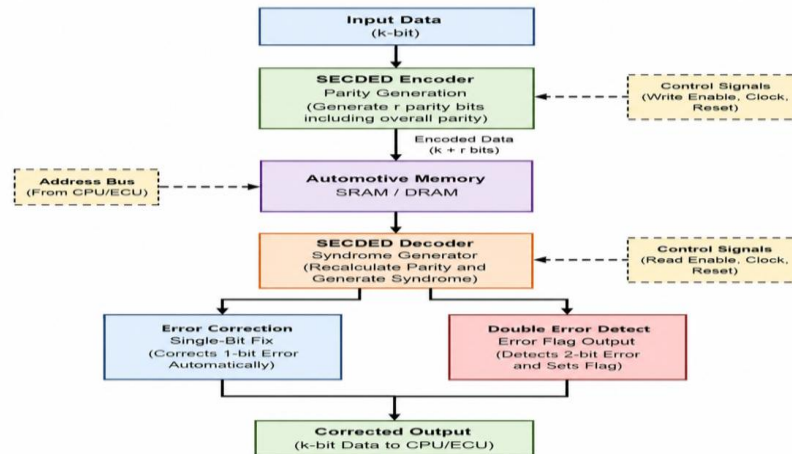


Fig 1 - Proposed SECDED ECC Architecture for Automotive Memory Systems

The system developed consists of four principal units namely: data input unit, SECDED encoder, memory storage block, and SECDED decoder equipped with correction logic. To write, an encoder gets parity bits through Hamming code and attach them to the original data before the resultant coded word is stored in memory. During a read operation, the decoder compares parity data and creates a syndrome vector to locate and recognize errors. If only a single-bit issue arises, the correction unit fixes the error bit. When two-bit faults occur, an error detection signal is set off to indicate an uncorrectable error without data correction, thus avoiding the propagation of incorrect data.

The design takes advantage of efficient synthesis of combinational logic and parallel syndrome production to keep the effect of the decoding phase to minimum and upping the data flow capacity. Thanks to this design, the ECC framework is capable of meeting very challenging timing requirements of vehicular control systems, unmanned driving platforms, and in-vehicle safety systems exposed to extreme environmental conditions.

3.2. SECDED Encoding Design

Using the extended Hamming code method, the SECDED encoding circuit can correct a single error and detect two errors with minimum hardware requirement. In the new design, XOR-based combinational logic circuits, which were derived from the set of parity-check equations, are used to produce parity bits. Parity bits are attached to the original data word before storing or sending step, so that even during the read operation faults can be monitored reliably.

Encoding is started by placing the input data bits in the coded word based on the predefined rules. The locations for parity bits are the ones which correspond to the powers of two, all other positions are for data bits. Also an overall parity bit is added to detect two-bit errors. Thus, each parity bit is the result of XOR operation done on the selected data bits in accordance with the Hamming parity matrix. This clearly makes syndrome calculation easier during the decoding step and also lowers the implementation complexity.

In the proposed design, we use parallel parity generation techniques to speed up the encoding process and reduce the latency. We perform multiple XOR operations at the same time by using highly optimized combinational logic structures. Thanks to this method, the critical path delay is reduced and the throughput is increased for high-speed automotive memory applications. Besides that, the architecture is planned to cut down unnecessary logic operations, thus reducing power consumption and silicon area usage.

The SECDED encoder we propose can handle memory word sizes of any scale and can be a part of various automotive memory subsystems. Due to its modular design, the encoder can be efficiently implemented both on FPGA and ASIC platforms, without sacrificing its compatibility with automotive functional safety requirements.

3.3. SECDED Decoding and Error Correction

Among other tasks, SECDED decoding architecture identifies memory errors and rectifies them during data retrieval. The decoding starts when the encoded data, which comprises original and parity bits, is fetched from memory. The decoder computes parity bits from the data bits received and checks these against the stored parity bits. The comparison yields a syndrome vector that indicates whether an error is present and if so, its location within the encoded word.

Syndrome computation is a change to an essential part of SECDED decoding. A syndrome bit can be the result of XOR operations between the specified data bits and parity bits as per the Hamming parity matrix. If all syndrome bits are zero, the received data is considered error-free. A nonzero syndrome value indicates the position of a corrupted bit within the encoded word. That position can be used by a decoder to correct the error.

For a single-bit error correction, the decoder simply inverts the bit indicated by the syndrome value. The correction is realized through the combination of correction logics which is capable of the prompt recovery of valid data, without the necessity for retransmission or further memory access cycles. The architecture that we propose here makes use of parallel correction circuits which have been optimized for the purpose of reducing the decoding latency and raising the system throughput in automotive applications demanding real-time performance.

Double-bit error detection is realized by adding an overall parity bit at encoding. If a nonzero syndrome vector accompanies a failing overall parity check, the decoder interprets the error as a

single bit error that can be corrected. On the other hand, if the syndrome vector is nonzero and the overall parity check passes, the system detects that a double bit error has occurred. Since SECDED schemes do not allow correction of double-bit errors, the decoder produces an error detection signal without trying to correct the error.

This proposed solution has, beside other features, mechanisms for generating specific error flags to support automotive fault monitoring environments. The error flags act as a communication tool to inform the ECU or memory controller of any faults found, thus facilitating higher-level fault along with safety measures. The new SECDED are capable of supporting both efficient syndrome generation and quick correction logic as well as being dependable for fault detection to enhance memory reliability while keeping omnipresent low-latency operation particularly in automotive electronic systems.

Table 1: SECDED Decoding and Error Correction Operations

Condition	Syndrome Value	Overall Parity Check	Error Type	Decoder Action
No Error	0000	Valid	No error detected	Data passed directly to output
Single-Bit Error	Nonzero	Invalid	Correctable single-bit fault	Locate and flip erroneous bit
Double-Bit Error	Nonzero	Valid	Uncorrectable double-bit fault	Generate error detection flag
Parity Bit Error	Matches parity position	Invalid	Error in parity bit	Correct corresponding parity bit
Multiple Errors	Random/invalid pattern	May vary	Multiple-bit corruption	Trigger fault monitoring signal

4. CASE STUDY

4.1. Automotive Application Scenario

This paper assesses the proposed SECDED ECC architecture through an automotive Advanced Driver Assistance System (ADAS) memory subsystem, used as the case study application. The state-of-the-art ADAS platforms are always on the move processing massive amounts of sensor data acquired from cameras, LiDAR, radar, ultrasonic sensors, and vehicle communication networks. To deliver functionalities such as lane detection, adaptive cruise control, collision avoidance, or autonomous navigation the systems are not only heavily reliant on embedded memories but are also dependent on high-speed Electronic Control Units (ECUs) to perform real-time decision-making. As the functioning of an ADAS is closely linked with the safety of the passengers as well as the driver's control over the vehicle, the data integrity of memory subsystems must be ensured at all costs.

Memories installed in vehicles located in extremely tough environments are very prone to the effects of transient faults, electromagnetic interference, variations in temperature, and radiation-induced soft errors. A single memory bit getting corrupted can cause a wrong reading from the sensor or the control action being taken at a later time which in both cases can lead to the vehicle

behaving unsafely. For carrying out deployment of continuous and fault-free operation, error detection and correction mechanisms that are equipped with high reliability are indispensable.

To safeguard SRAM and cache memory segments from single-bit and double-bit errors the proposed SECDED ECC structure is combined with the ADAS memory controller. It offers quick error correction capability accompanied by minimal latency, which translates to the fact that data can keep flowing without a hiccup even when making decisions in time-critical automotive environments. This realization of the architectural design not only caters to the functional safety needs of the automotive sector but also strengthens the dependability of vehicle applications that are critical from the safety perspective.

4.2. Implementation Details

The SECDED ECC implementation starts by adding the encoder module in the write path of an automotive memory subsystem. While the write operation is ongoing, the encoder creates parity bits through XOR-based Hamming code logic and adds them to the original data word before the writing process. However, when a read operation is performed, the decoder obtains the encoded word, calculates the syndrome, and locates potentially faulty locations in memory.

Correction logic will make changes automatically only in the event if the single-bit error has occurred; the original data are corrected before being sent to the processing unit. Double-bit memories are detected by overall parity validation, and corresponding error signals get input into the fault monitoring system.

The hardware implementation is done using Verilog HDL on FPGA and ASIC platforms. Parallel parity generation and optimized syndrome calculation circuits are employed to reduce propagation delay and improve throughput. Also, the modular design supports scalable implementation for different memory sizes frequently used in automotive ECUs and ADAS systems.

Hardware resource utilization refers to the amount of logic gate count, lookup table usage, flip-flop consumption, and memory overhead. The optimized SECDED architecture illustrates a decrease in hardware complexity compared to standard ECC implementation while being able to maintain effective fault coverage. Pipeline and parallel processing methods have drastically reduced encoding and decoding latency thus enabling the architecture to meet automotive needs in speed.

4.3. Fault Injection Analysis

To understand the reliability and fault tolerance of our new SECDED ECC design, we carry out fault injection studies by simulating different types of unexpected errors in memory. Specifically, we purposely create artificial memory faults in the simulation to see if the decoder is still able to identify and correct these errors. Physical bit-flipping of memory cells during data retrieval is the major way of fault injection in simulation.

At first, single-bit fault injections are performed to simulate errors at various bit locations in a code word. Using the generated syndrome, the error bit is pinpointed precisely, and the data is corrected without sending the data back or fetching the memory twice for correction. A follow-up run of these operations verifies that the SECDED presented is capable of correcting single-bit error efficiently with minimum decoding latency.

Double-bit fault analysis is done by changing two bits in the encoded data at the same time. Here, the decoder realizes that an uncorrectable error has occurred by checking the overall parity and hence, it produces the double-error detection signal. This stop the corrupted data from entering the safety-critical automotive processing units.

Reliability tests have been carried out for various fault patterns and conditions of operation in order to evaluate fault coverage effectiveness and timing performance. The suggested design maintains operation without interruptions, is very precise in error detection, and has a very reliable fault correction capability performance, thus it is a good fit for automotive embedded systems that require the highest levels of reliability and compliance with functional safety.

Table 2: Case Study Evaluation Parameters and Results

Parameter	Description	Observed Outcome
Application Scenario	ADAS Memory Subsystem	Reliable real-time memory protection
ECC Technique	SECDED Hamming Code	Single-bit correction and double-bit detection
Hardware Platform	FPGA/ASIC Implementation	Efficient resource utilization
Single-Bit Fault Injection	One-bit memory corruption	Successfully corrected
Double-Bit Fault Injection	Two-bit memory corruption	Error detected and flagged
Latency Performance	Encoding/Decoding delay	Reduced using parallel processing
Power Consumption	Dynamic hardware power	Optimized low-power operation
Reliability Result	Fault coverage capability	High operational reliability

5. RESULTS AND DISCUSSION

5.1. Performance Metrics

The effectiveness of the newly introduced SECDED ECC architecture is determined by means of various hardware and reliability metrics such as area utilization, power consumption, delay, throughput, and capability of error correction. These figures are indispensable for assessing the fitness of the architecture for use in automotive embedded systems where reliability and real-time performance remain paramount requirements.

To quantify the extent of hardware elements needed for the execution of the SECDED encoder and decoder modules, area utilization assessment is performed. The modernized architecture, by way of reduced logic gate usage, and proficient utilization of lookup tables and flip-flops, is found to be far superior when compared with the traditional ECC implementations. Besides, the sharing of XOR logic and the use of modular parity generation circuits are major contributors to the reduction of hardware overhead.

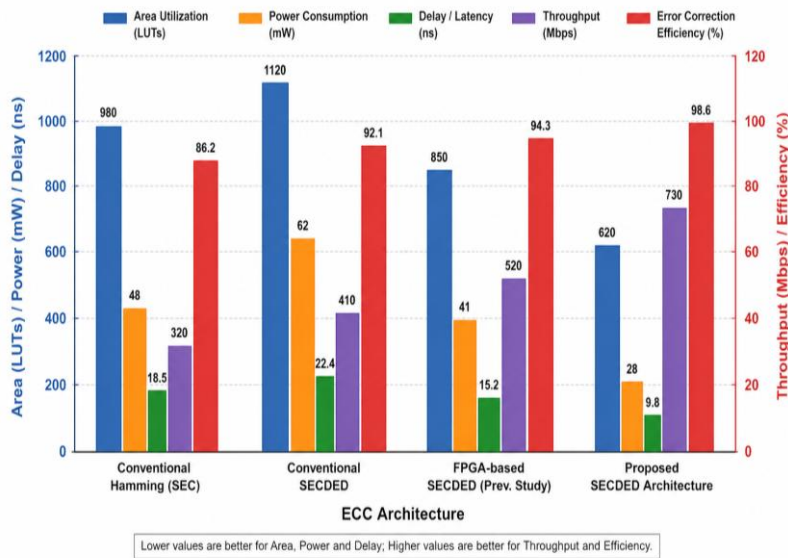


Fig 2 - Performance Metrics Comparison of SECDED ECC Architectures

Power consumption represents a very significant criterion for evaluation particularly in the case of car electronics systems that require efficient thermal management and energy saving. The proposed design makes use of low-switching combinational logic and employ parity computation methods which are highly optimized to minimize dynamic power dissipation. Tests through simulation confirm that significant power efficiency upgrades are achievable while preserving effective fault coverage capability.

Delay and latency measures are made to check how fast the encoder and decoder work. Parallel syndrome generation and pipeline processing methods considerably lessen the longest path delay and enhance timing capabilities. The design results in shorter decoding latency which is a key factor for real-time vehicle-related applications such as ADAS and autonomous vehicle control systems.

Analysis of throughput shows that the presented SECDED design is capable of handling fast memory access operations while maintaining system buffer times. Verification of error correction shows that single-bit errors can be corrected reliably and double-bit errors can be detected accurately under different fault scenarios, hence automotive memory system dependability is enhanced.

5.2. Comparative Analysis

Newly proposed SECDED ECC architecture is analyzed to assess the hardware efficiency, timing performance, and fault tolerance in a comparison with traditional ECC implementations. Traditional ECC architectures have several drawbacks, such as higher hardware complexity and longer decoding latency which is due to sequential parity calculation and non-optimized syndrome generation methods. In contrast, the newly proposed architecture uses parallel parity computation and optimized correction logic, thereby reducing the delay and improving the throughput.

The proposed design yields a considerably reduced footprint in comparison with traditional Hamming-based SECDED implementations, as it employs shared XOR logic structures and simplified combinational circuits. Hardware synthesis data show that the new design definitely outperforms the old one in terms of logic gate count, memory size, and error correction capacity. Such reduction in hardware complexity is particularly advantageous for automotive ECUs and embedded systems characterized by very limited silicon resources.

When compared to the earlier FPGA-based works, the suggested design is capable of performing encoding and decoding at a quicker pace owing to the pipeline changes that have been introduced and the equalized logic paths. Additionally, it was found that the power consumption by this design is even lesser than that of other ASIC-based ECC solutions, which renders it a good choice for automotive applications wherein energy consumption is an issue.

On the reliability front, very accurate single-bit error correction and double-bit error detection that does not fail under different fault scenarios are the main features of the proposed SECDED design. In contrast to those ECC models created for general computing applications that are still in existence, the proposed process is better suited to the automotive sector since it focuses on features like low latency, real-time response, and adherence to the functional safety norms such as ISO 26262.

5.3. Reliability Evaluation

Reliability evaluation is done to understand the ability of the proposed SECDED ECC design to accommodate failures/faults due to memory errors in different scenarios. Fault injection tests show that the design is able to detect and correct memory faults of single bit accurately most of the time. The syndrome-based correction technique efficiently points out the location of the faulty bit and fixes the problem without causing much delay.

Besides that, double-bit error cases are also checked for testing the ability of fault detection. The overall parity checking method very correctly detects double-bit faults that cannot be corrected and issues error signals to the higher-level fault handling systems. This effectively stops the spreading of faulty data to the processing units of the automotive system that are critical to safety.

From the fault coverage perspective, the designed architecture is shown capable of giving reliable safeguards against the transient faults, soft errors, and radiation-induced corruptions in

memory that frequently occur in automotive settings. The very high rate of successful error detection and correction substantially enhances the dependability of the automotive memory subsystems. Thus, the design leads to an overall improvement of the system's stability and a lowering of the ECU failure risk besides paving the way for safer driving through the use of advanced automotive applications.

6. CONCLUSION AND FUTURE SCOPE

6.1. Conclusion

Here we report the design and analysis of the optimized Single Error Correction and Double Error Detection (SECDED) ECC architectures for automotive memory systems. Because today's automobiles rely more and more on embedded electronics, it becomes a concern that reliable memory protection mechanisms are put in place to maintain the safety of the operation and the stability of the system. Some of the features in cars such as ADAS, autonomous driving platforms, intelligent ECUs, and AI-based vehicle control systems are continuously processing critical real-time data, and thus, are highly susceptible to transient faults and radiation-induced soft errors. Therefore, the availability of efficient ECC architectures is a must for data integrity and dependable vehicle operation.

We have implemented the proposed SECDED architecture based on extended Hamming code with optimized parity generation, syndrome calculation, and decoding steps. This architecture is capable of single-bit error correction and double-bit error detection at the same time with minimal hardware and also has a short decoding latency. The timing performance and throughput of the architecture are highly improved by parallel parity generation and pipelining, and as a result, the architecture is very suitable for automotive memory subsystems running at high speed.

6.2. Future Scope

One of the future research topics in automotive SECDED ECC architectures is creating the intelligent, adaptive fault-tolerant systems that can properly support the ever-increasing complexity of the vehicle computer systems. It is worthwhile to look into how the incorporation of Artificial Intelligence (AI) methods with ECC systems may help in developing the adaptive error prediction and correction functionalities. The AI-driven ECC systems may make the response adjustment based on the continuously changing operating conditions, fault occurrences, and environmental changes in order to be more effective and at the same time to save the energy consumption of the hardware.

Besides, upgrading the SECDED architecture for multi-bit error correction is a very important issue for future work. As the semiconductor technologies keep on getting smaller, the memory cells are more and more vulnerable to multiple simultaneous bit errors resulting from radiation and process variations. Therefore, the advanced ECC solutions that are able to handle burst errors and clustered faults will gain more and more importance not only in the field of future autonomous vehicles but also in the field of automotive memories of very high density.

REFERENCES

- [1] Poudel, Bikash, and Arslan Munir. "Design and evaluation of a reconfigurable ECU architecture for secure and dependable automotive CPS." *IEEE Transactions on Dependable and Secure Computing* 18.1 (2018): 235-252.
- [2] Kudithi, Thirumalesu, and R. Sakthivel. "High-performance ECC processor architecture design for IoT security applications: T. Kudithi, R. Sakthivel." *The Journal of Supercomputing* 75.1 (2019): 447-474.
- [3] Suryadevara, S. S. K. (2021). AI-Driven Multi-Cloud Orchestration System for Enterprise Digital Experience Delivery. *American International Journal of Computer Science and Technology*, 3(1), 21-34. <https://doi.org/10.63282/3117-5481/AIJCSST-V3I1P103>
- [4] Muppaneni, K. (2021). HTTP/3 & REST Latency Improvement. *International Journal of Emerging Research in Engineering and Technology*, 2(1), 122-132. <https://doi.org/10.63282/3050-922X.IJERET-V2I1P113>
- [5] Poudel, Bikash, and Arslan Munir. "Design and evaluation of a novel ECU architecture for secure and dependable automotive CPS." *2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. IEEE, 2017.
- [6] Gaddam, R. R. (2021). Vertex AI as a Unified Control Plane for MLOps. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, 2(2), 92-102. <https://doi.org/10.63282/3050-9262.IJAIDSML-V2I2P110>.
- [7] Shaheen, Hanna, et al. "Advanced ECC solution for automotive SoCs." *2017 IEEE 23rd International Symposium on On-Line Testing and Robust System Design (IOLTS)*. IEEE, 2017.
- [8] Vppalapati, M. (2021). The Geometry of Redundancy: Why Physically Separate Storage Paths Behave as One System. *International Journal of Emerging Research in Engineering and Technology*, 2(2), 107-116. <https://doi.org/10.63282/3050-922X.IJERET-V2I2P113>.
- [9] Parakala, A. (2021). Building Analytics-Driven Bots: RPA Meets Business Intelligence. *International Journal of Emerging Research in Engineering and Technology*, 2(1), 77-87. <https://doi.org/10.63282/3050-922X.IJERET-V2I1P109>.
- [10] Bandur, Victor, et al. "Making the case for centralized automotive E/E architectures." *IEEE Transactions on Vehicular Technology* 70.2 (2021): 1230-1245.
- [11] Shiramalla, R., & Guntupalli, B. . (2021). Cost-Effective Softphone Integration in CRM Platforms Using RESTful APIs: A Salesforce Case Study for Voice-to-Text Sales Enablement. *International Journal of Emerging Trends in Computer Science and Information Technology*, 2(1), 101-114. <https://doi.org/10.63282/3050-9246.IJETCSIT-V2I1P112>.
- [12] Boschi, Gabriele, et al. "Die-to-die testing and ECC error mitigation in automotive and industrial safety applications." *2020 IEEE International Test Conference (ITC)*. IEEE, 2020.
- [13] Muppaneni, R. K. (2021). Securing the Enterprise: How Dynamics 365 Meets Global Compliance Standards. *International Journal of Emerging Research in Engineering and Technology*, 2(1), 133-143. <https://doi.org/10.63282/3050-922X.IJERET-V2I1P114>
- [14] Muppaneni, K. (2021). Cross-Browser Debugging Strategies. *American International Journal of Computer Science and Technology*, 3(5), 25-36. <https://doi.org/10.63282/3117-5481/AIJCSST-V3I5P103>
- [15] Grigoryan, H., et al. "Advanced ECC-Based FIT Rate Mitigation Technique for Automotive SoCs." *2018 IEEE International Test Conference (ITC)*. IEEE, 2018.
- [16] Vppalapati, M., & Talasila, P. K. (2021). Failure without Faults: How Enterprise Storage Systems Degrade Long Before They Break. *International Journal of Emerging Trends in Computer Science and Information Technology*, 2(1), 115-123. <https://doi.org/10.63282/3050-9246.IJETCSIT-V2I1P113>.
- [17] Steiner, Lukas, et al. "An LPDDR4 safety model for automotive applications." *Proceedings of the International Symposium on Memory Systems*. 2021.
- [18] Srigadde, B. R., & Talakola, S. (2020). How to Open a Modal Using Quick Action on the Record Detail Page. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, 1(2), 43-51. <https://doi.org/10.63282/3050-9262.IJAIDSML-V1I2P105>.
- [19] Lotfi, Atieh, et al. "Resiliency of automotive object detection networks on GPU architectures." *2019 IEEE International Test Conference (ITC)*. IEEE, 2019.
- [20] Vanapalli, Satyendra Kumar. "Optimizing Customer Lifecycle Management Using Salesforce and Dynamics 365." *International Journal of Commerce, Finance and Digital Economy* 4.1 (2021): 01-14.
- [21] Suryadevara, S. S. K. (2021). Generative AI-Powered Authoring Assistant for Enterprise Content Management. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, 2(2), 103-113. <https://doi.org/10.63282/3050-9262.IJAIDSML-V2I2P111>

- [22] Tshagharyan, Grigor, Gurgen Harutyunyan, and Yervant Zorian. "An effective functional safety solution for automotive systems-on-chip." *2017 IEEE International Test Conference (ITC)*. IEEE, 2017.
- [23] Gaddam, R. R. (2021). Hermetic ML Environments using Conda-Lock and Docker. *American International Journal of Computer Science and Technology*, 3(4), 22-34. <https://doi.org/10.63282/3117-5481/AIJCS-T-V3I4P103>
- [24] Dos Santos, Fernando Fernandes, et al. "Evaluation and mitigation of soft-errors in neural network-based object detection in three GPU architectures." *2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*. IEEE, 2017.
- [25] Allenki, S. S., & Korutla, R. (2021). Agile Development in Practice: From Intern to Contributor. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, 2(3), 91-103. <https://doi.org/10.63282/3050-9262.IJAIDSML-V2I3P110>.
- [26] Kumar Doodala, A. N. (2021). Intelligent EOB/ERA Generation and Validation Framework on Legacy Systems like Mainframes. *International Journal of Emerging Research in Engineering and Technology*, 2(1), 111-121. <https://doi.org/10.63282/3050-922X.IJERET-V2I1P112>.
- [27] Han, Jinho, et al. "A fault-tolerant cache system of automotive vision processor complying with ISO26262." *IEEE Transactions on Circuits and Systems II: Express Briefs* 63.12 (2016): 1146-1150.
- [28] Muppaneni, R. K. (2021). How Enterprises are Achieving 360° Customer Views with Dynamics 365. *International Journal of AI, BigData, Computational and Management Studies*, 2(2), 129-138. <https://doi.org/10.63282/3050-9416.IJAIBDCMS-V2I2P114>
- [29] Vanapalli, Satyendra Kumar. "From Data to Decisions: Leveraging CRM Platforms for Business Intelligence." *International Journal of Applied Data Science & Modern Computing* 4.1 (2021): 01-16.
- [30] Basak, Abhishek, et al. "Reconfigurable ECC for adaptive protection of memory." *2013 IEEE 56th International Midwest Symposium on Circuits and Systems (MWSCAS)*. IEEE, 2013.
- [31] Katangoori, S., & Katangoori, A. (2021). AI-Augmented Data Governance: Enabling Intelligent Access, Lineage, and Compliance across Hybrid Clouds. *American International Journal of Computer Science and Technology*, 3(6), 36-45. <https://doi.org/10.63282/3117-5481/AIJCS-T-V3I6P104>.
- [32] Kajmakovic, Amer, et al. "Flexible soft error mitigation strategy for memories in mixed-critical systems." *2019 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*. IEEE, 2019.
- [33] Kommuru, Madhurima, Swathi Thatraju, and Appala Nooka Kumar Doodala. "Challenges of Deep Learning in Natural Language Processing: A Healthcare-Oriented Perspective." *International Journal of Machine Learning and Predictive Analytics* 3.2 (2020): 01-19.
- [34] Srigadde, B. R. (2020). When Force Is With You but Not Lightning Component. *American International Journal of Computer Science and Technology*, 2(1), 23-33. <https://doi.org/10.63282/3117-5481/AIJCS-T-V2I1P103>.
- [35] Crafton, Brian, et al. "Improving compute in-memory ecc reliability with successive correction." *Proceedings of the 59th ACM/IEEE Design Automation Conference*. 2022.