AN ENERGY-EFFICIENT FAULT-TOLERANT METHOD IN NONVOLATILE MAIN MEMORY

R Vyshnavi[1] , Meghraj Meghana[2], Mathangi Preethi[3], Bairoju Rishika Viswanath[4]

[1]Assistant Professor, Department of ECE, Bhoj Reddy Engineering College for Women, Hyderabad, India

[2,3,4]UG scholar students Department of ECE, Bhoj Reddy Engineering College for Women, Hyderabad, India

Abstract

An energy- and area-efficient solution for tolerating the stuck-at faults induced by an endurance problem in secure-resistive main memory. A large number of memory locations with stuck-at faults might be used in the suggested technique to appropriately store the data by using the rotational shift operation and the random properties of the encrypted data encoded by the Advanced Encryption Standard (AES). The suggested method's energy usage is much lower than that of other previously presented approaches because of its straightforward hardware implementation. The error correction code (ECC) and error correction pointer (ECP) are two more error correction techniques that may be used in conjunction with this one. The suggested approach is put into practice in a main memory system based on phase-change memory (PCM) and contrasted with three error-tolerating techniques in order to determine its effectiveness. The findings show that the suggested approach provides 82% energy savings over the state-of-the-art technique for a stuck-at fault incidence rate of 10−2 and an uncorrected bit error rate of 2 × 10−3. More broadly, we demonstrate that the fault coverage of the suggested approach is comparable to the state-of-the-art method using a simulation analysis methodology.

Introduction

Human life has changed drastically due to semiconductor industry initiatives. Advanced technology has created many apps, gadgets, appliances, and devices. Portable gadgets make multitasking simpler anytime, anyplace. Nanotechnology allows this competence. Many sophisticated functionalities are now accessible on watches and pens due to the rising density of integrated circuits. Advances in deep submicron technologies have reduced feature size.

Memory density, the most important aspect of semiconductor production and applications, has increased due to decreasing feature size. It is now essential to system on chip. Life in micro-miniature is full with portable technologies and gadgets. The most applications depend only on semiconductor electronics. Memory must be fault-tolerant for such an important application portion. DRAM's utilization in various applications across domains is the subject of this paper.

A real-time computer system has temporal restrictions. Thus, a real-time system can calculate or decide quickly. Deadlines exist for these crucial computations. Practically, a missed deadline is the same as an incorrect response. Processor cores double every two years, while DRAM DIMM capacity double every three. A wide gap exists between core count and memory density. However, typical DRAM chips need server power. DRAM scaling for higher memory density faces high leakage current, reduced memory cell reliability, and more

complicated fabrication processes. To address scale and power consumption, volatile (DRAM-based) and nonvolatile (resistive-based) memory systems have emerged.

Memory is critical to SoCs. Memory test must be done properly when system-on-chip memory occupies increasing space. Fast testing with minimal gear is key for testing technologies. System-on-chip yield, reliability, and quality depend on memory testing. Traditional DC or AC parametric testing techniques are ineffective because the memory pins are firmly integrated in the semiconductor. Thus, complicated memory testing requires an effective test method.

Literature Survey

Federal Information Processing Standards Publication 197, "Announcing the Advanced Encryption Standard (AES)," November 2001. This standard describes how to use cipher keys of lengths of 128, 192, and 256 bits to process data blocks of 128 bits using the symmetric block cipher Rijndael algorithm. Although Rijndael was intended to support different block sizes and key lengths, this standard does not use them. The method described below shall be called “the AES algorithm” during the duration of this standard. "AES-128, AES-192, and AES-256" are three distinct "flavors" that may be named after the three different key lengths that the method can be employed with.

"Accompact rijndael hardware architecture with S-box optimization," in A. Satoh, S. Morioka, K. Takano, and S. Munetoh, Springer-Verlag Berlin Heidelberg, 2001. For more than 20 years, the de facto norm for US government information processing standards has been DES (Data Encryption norm), a common-key block cipher. The latest Advanced Encryption Standard (AES) is Rijndael, according to NIST (National Institute of Standard Technology).

Fast Implementations of Secret-Key Block Ciphers Using Mixed Inner- and Outer-Round Pipelining, P. Chodowiec, P. Khuon, and K. Gaj, Proc. ACM/SIGDA Int. Symposium on Field Programmable Gate Arrays, FPGA'01, Monterey, CA., 2001.

Pipelining is a well recognized method that expedites the functioning of digital systems by concurrently processing many data blocks. Secret-key block ciphers, like the Data Encryption Standard (DES), were originally intended to be hardware-implemented. They so had a very easy and quick cipher round. This characteristic suggested that there was only one feasible kind of pipelining, which included unrolling the rounds and inserting registers in between cipher rounds. This kind of pipelining is known as outer-round pipelining. This design technique became suboptimal due to the introduction of new block ciphers that were based on simple instructions from current microprocessors and designed for software implementations. Modern ciphers like Rijndael, RC6, and Two Fish have quite complicated fundamental cipher rounds, which restricts the maximum clock frequency of their non-pipelined iterative hardware implementations. Simultaneously, the space required within an integrated circuit to repeat a single round of these ciphers may make it impossible to do the loop unrolling needed for outer-round pipelining.

Consequently, it became possible to implement a novel kind of pipelining in which pipeline registers were placed within a cipher round. This construction is known as inner-round pipelining. With just a little increase in the circuit area, the inner-round pipelining offers a significant boost in cipher speed. Furthermore, the inner-

round pipelining and outer-round pipelining may be coupled with ease to provide the quickest feasible architecture for a particular block cipher if the space available on the integrated circuit is larger than the area needed by the iterative design.

Block Diagram

Block Diagram-1

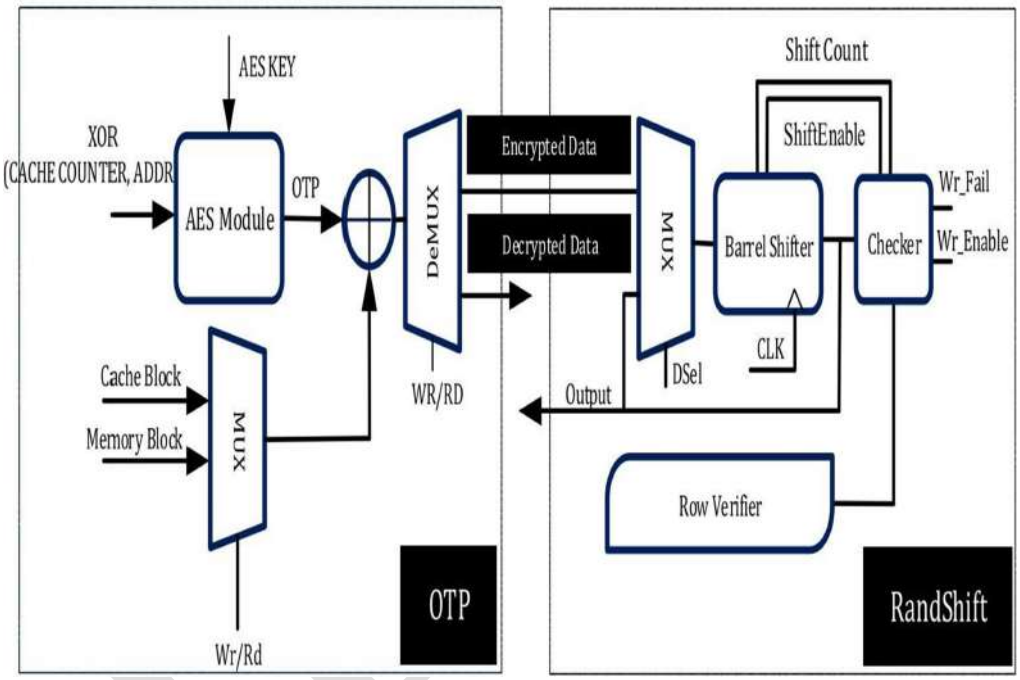

Fig. 2.1 Block Diagram Using Barrel Shifter

The RandShift system, which includes Encrypt/Decrypt and Shifter components, is shown in the diagram. Once the OTP is used to generate encrypted data in the Encrypt/Decrypt unit, the data is then sent to the Shifter unit. The Row Verifier unit transmits the precise location and corresponding value of the defects to the Checker unit. The Checker unit verifies the correspondence between the shifted data bit values and the required fault values determined by the Row Verifier. The Shifter unit is a basic barrel shifter that is constructed using multiplexers. The RandShift approach may be employed either at the row-level or word-level. Regarding the row level, all the data within a row are completely shifted, but at the word level, each word (either 64 or 128 bits as mentioned in this article) inside a row is shifted individually.

The block diagram depicts two crucial registers, namely key_reg_reg[0][0] and key_reg_reg[0][1]. Every register is equipped with a clock (C) input, a clear (CLR) input, an enable (CE) input, and a data (D) input. The clock input is used to synchronize the register with the system.

clock. The clear input is used to reset the register to zero. The enable input is used to control when the data on the data input is written to the register. The data input is used to provide thedata to be written to the register. The block diagram also shows a connection between the key registers and a coreblock. The core block is likely the part of the system that uses the cryptographic key. The connection between the key registers and the core block is labeled core_key [255:0], which suggests that the key is 256 bits wide.

Block Diagram-2

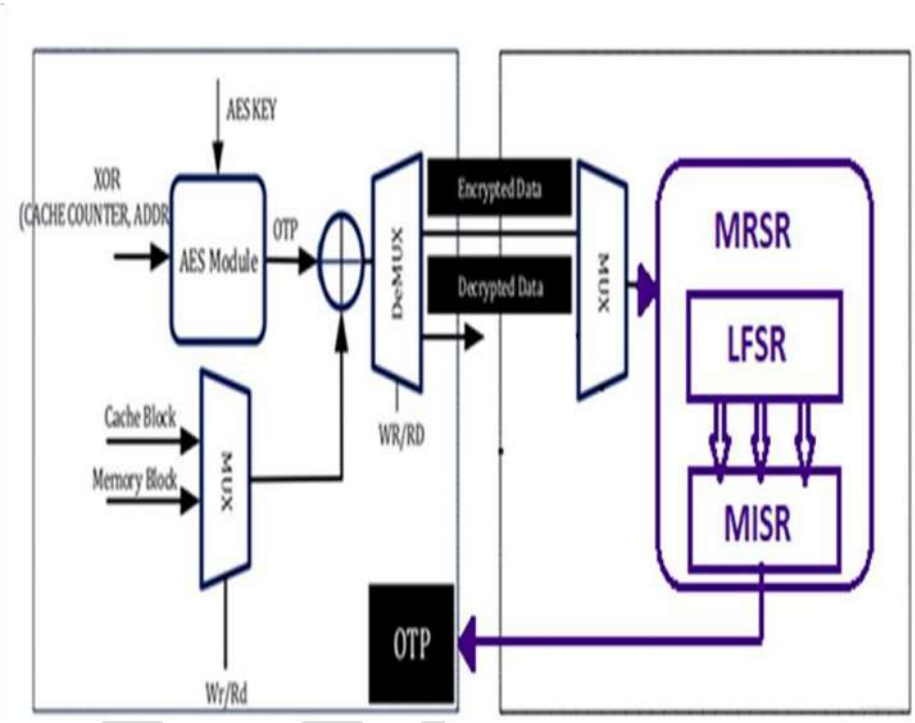

Fig. 2.2 Block Diagram Using MRSR

Multi-rate shifters can be optimized for different shifting speeds, potentially improving efficiency in terms of speed and resource usage. The term "multi-rate shifter" is nota standard term in the context of commonly known digital logic components like barrel shifters or regular shifters. Fault-tolerant systems often incorporate redundancy and error-detection mechanisms. A multi-rate shifter could be part of a strategy to introduce redundancy or to adapt the shifting operation based on error-detection feedback.

Results

In this chapter, it includes the procedure and also discuss about the results of An Energy-Efficient Fault Tolerant Method in Nonvolatile Main Memory. It requires different types of Hardware and Software Components which concludes the proper result and output ofthe project. According to the proposed plan the final outcome of this leads to the security of the data. Through this project, the data is encrypted and stored in memory. Due to stuck-at- faults in the process some amount of data is lost. For this reason, the data is divided into bits and encrypted using Randshift method. With this simple technology in use, data is secure. Apart from this all ,these systems

are more efficient and Communication is between user's device and remote server

Output 1:

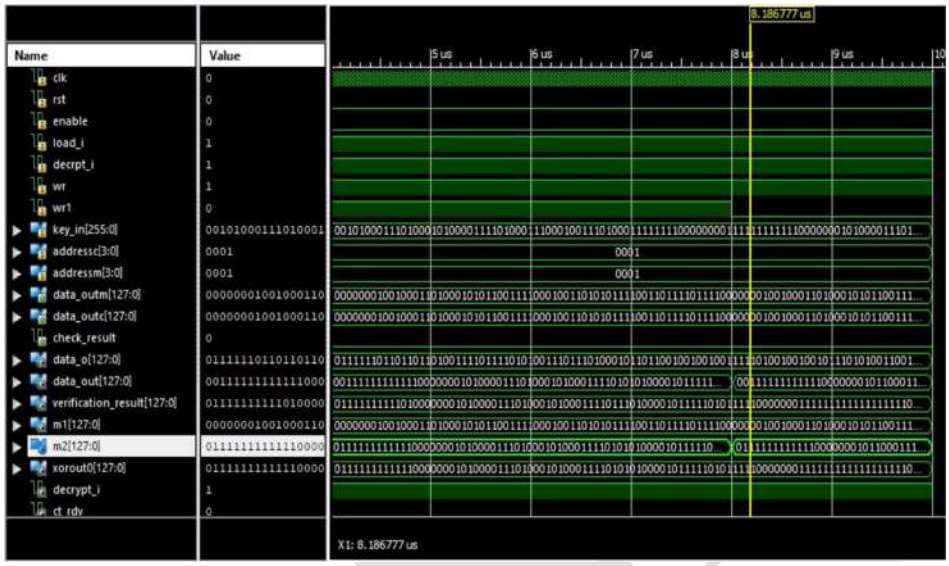

Fig. 5.1 Simulation of Verilog code

- cycle_ctr [31:0]: This register is a 32-bit counter that keeps track of the number of clock cycles. The value in the screenshot is 170.
- error_ctr [31:0]: This register is a 32-bit counter that keeps track of the number of errors. The value in the screenshot is 0, which means there are no errors.
- tc_ctr[31:0]: This register is a 32-bit counter that appears to keep track of the number of transfers that have been completed. The value in the screenshot shows 8 transfers have been completed.
- tb_cik: This signal is currently inactive (low).
- tb_reset_n: This signal is currently active (high).
- to encdec: This signal is currently inactive (low).
- tb_init: This signal is currently inactive (low).
- to_next: This signal is currently inactive (low).
- to_ready: This signal is currently inactive (low).
- to_key [255:0]: This is a 256-bit register that stores the encryption key. The value in the screenshot is partially obscured, but it appears to be a valid key.
- tb_result [127:0]: This is a 128-bit register that stores the result of the encryption or decryption operation. The value in the screenshot is partially obscured, but it appears to bevalid data.
- tb_result_valid: This signal is currently inactive (low), which means the result is not yet valid.
- DEBUG [31:0]: This register is a 32-bit register that is used for debugging purposes. Thevalue in the screenshot is 0.

- DUMP WAIT [31:0]: This register is a 32-bit register that is used for debugging purposes.The value in the screenshot is 0.
- CLK HALF PERIOD [31:0]: This register is a 32-bit register that stores the half period ofthe clock signal. The value in the screenshot is 1.
- CLK_PERIOD [31:0]: This register is a 32-bit register that stores the period of the clock signal. The value in the screenshot is 2.
- AES_128_BIT_KEY: This signal is inactive (low), which means a 128-bit key is not beingused.
- AES_256_BIT_KEY: This signal is active (high), which means a 256-bit key is beingused.
- AES DECIPHER: This signal is inactive (low), which means the device is currently inencryption mode.
- AES ENCIPHER: This signal is active (high), which means the device is currently inencryption mode.

The decrypted plaintext is obtained, which should match the original input data. The application of the RandShift AES algorithm provides enhanced security compared to thestandard AES algorithm by introducing randomness into the encryption process. This randomness makes it more challenging for attackers to analyze patterns in the encrypted data and mount successful attacks. However, it's essential to ensure that the random shifting operation does not compromise the security or integrity of the encryption process. Additionally, like any cryptographic algorithm, the strength of the encryption depends on therandomness of the key and the quality of the random number generator used.

The outputs of Verilog code are data_out, verification_result, and m2 are simulated and faults of each bit has been verified.

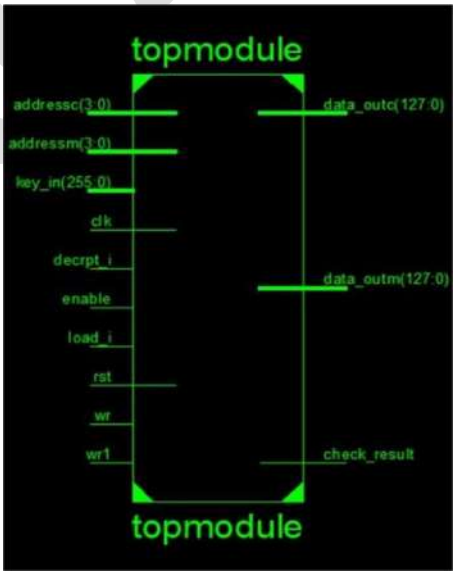

Fig. 5.2 Top Level Hierarchy

The schematic diagram of the synthesized verilog code can be considered by means of doubleclicking View RTL Schematic Synthesize-XST menu in the approach Window. This is competent to be a useful procedure to debug the code if the output just isn't assembly our requirements inside the proto kind board. Via utilising double clicking it opens the easiest stagemodule displaying handiest input(s) and output(s) as proven.

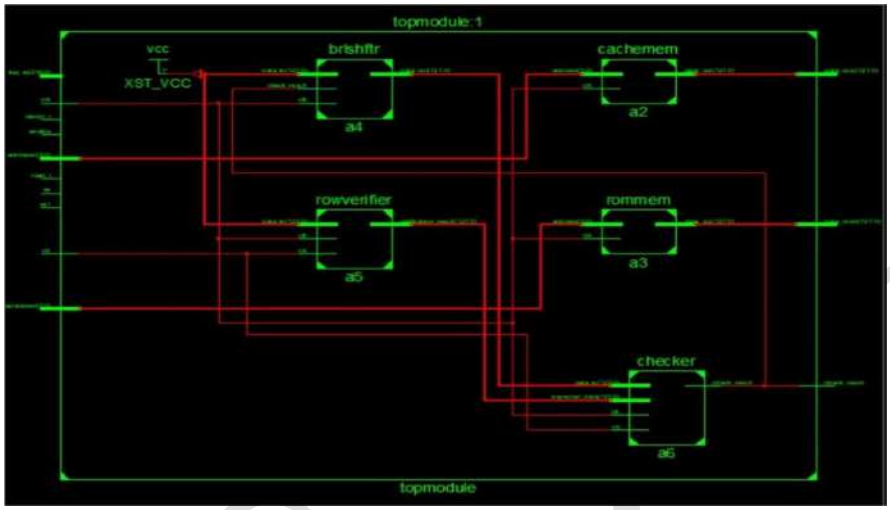

Fig. 5.3 Realized logic for Verilog code-1

By double clicking the rectangle, it opens the realized internal logic.

topmodule Project Status (12/20/2023 - 01:26:22)			
Project File:	aes.xise	Parser Errors:	No Errors
Module Name:	topmodule	Implementation State:	Synthesized
Target Device:	xc7z045-3ffg900	• Errors:	No Errors
Product Version:	ISE 14.3	• Warnings:	283 Warnings (1 new)
Design Goal:	Balanced	• Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	• Timing Constraints:	
Environment:	System Settings	• Final Timing Score:	

Device Utilization Summary (estimated values)			[-]
Logic Utilization	Used	Available	Utilization
Number of Slice Registers	3	437200	0%
Number of Slice LUTs	73	218600	0%
Number of fully used LUT-FF pairs	1	75	1%
Number of bonded IOBs	267	362	73%
Number of BUFG/BUFGCTRLs	1	32	3%

Fig. 5.4 Area Parameters of the Design-1

The area and power are recorded from topmodule project status and logic utilization isrecorded from device utilization summary.

```
Timing Summary:
---------------
Speed Grade: -3

   Minimum period: 1.628ns (Maximum Frequency: 614.341MHz)
   Minimum input arrival time before clock: 0.564ns
   Maximum output required time after clock: 0.515ns
   Maximum combinational path delay: No path found
```

Fig. 5.5 Timing Summary-1

The Time Period has been recorded in Timing Summary and gives minimum input arrivaltime before clock and maximum output required time after clock.

Output 2:

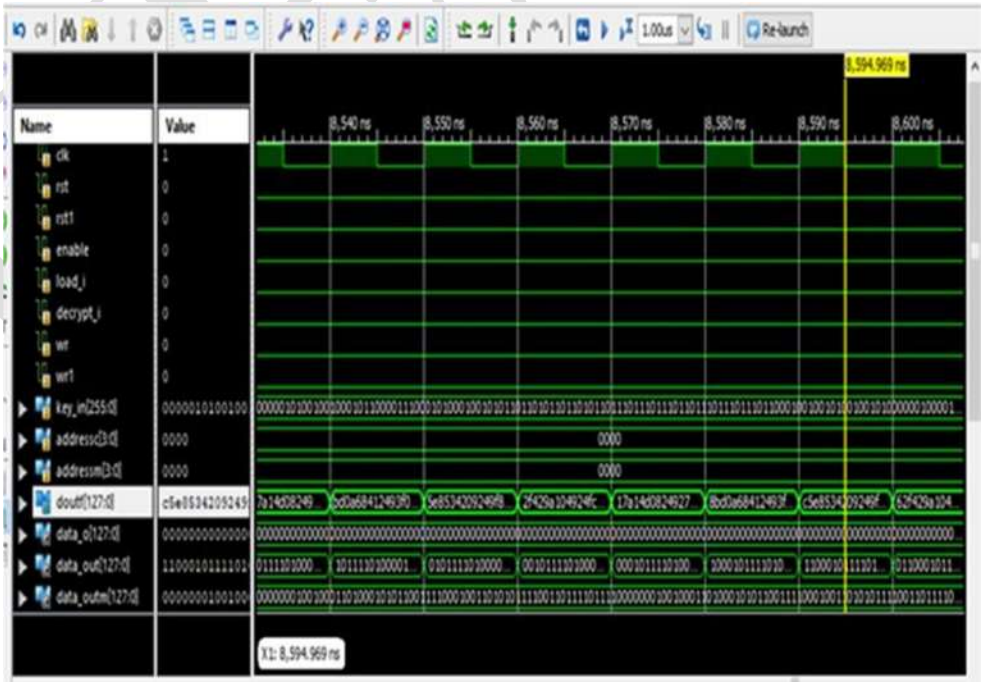

Fig 5.6 Simulation of Verilog Code-2

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slice Registers	1276	408000	0%
Number of Slice LUTs	2384	204000	1%
Number of fully used LUT-FF pairs	1121	2539	44%
Number of bonded IOBs	399	600	66%
Number of BUFG/BUFGCTRLs	1	32	3%

Fig. 5.7 Area Parameters of the Design-2

The area and power are recorded from topmodule project status and logic utilization is recordedfrom device utilization summary.

```
Minimum period: 5.134ns (Maximum Frequency: 194.792MHz)
Minimum input arrival time before clock: 4.298ns
Maximum output required time after clock: 0.521ns
Maximum combinational path delay: No path found
```

Fig. 5.8 Timing Summary-2

The Time Period has been recorded in Timing Summary and gives minimum input arrivaltime before clock and maximum output required time after clock.

Conclusion

The project "An Energy-Efficient Fault-Tolarant Method in Nonvolatile Main Memory" has been successfully designed and tested. This is a method employing the randomness feature of AES encryption as well as rotational shift operation to tolerate hard faults in nonvolatile memory cells. This method, which was called RandShift, enjoyed the simple hardware implementation and low energy consumption. It limited the need for exploiting powerful error correction methods, such as ECC and ECP. The results of our comparative study showed up to 82% lower energy consumption for RandShift when obtainingabout the same fault coverage as that of the state-of-the-art technique. This method is to tolerate the faults in the PCM main memory when encrypted data are

stored on it. improved reliability and reduced power consumption.

They also come with potential drawbacks like increased complexity and possible trade-offs between energy efficiency and performance. It's important to carefully evaluate these factors based on the specific application and requirements. This method results shows that the data is securely encrypted and decrypted and stored in memory without any faults.

The decoding operation with the transmitted data becomes difficult for data attacker, as it is a tedious process to match which combination of bits will work out. Even if the bit sequence is intercepted while transferring of data, discovering the correct combinations of 8-bit sequence (stored in ROM and to be XOR-ed with incoming bit sequence) for generating plain text from the cipher text is not at all feasible. Further it does not have a conventional polynomial feedback function but a mixture of various combinational operations.

One need not worry about hacking of e-mail accounts as even if the password is known to the hacker, he or she will only be able to see the sender's name and sequence of 0's and 1's in text area. Unless and until the user works on his or her system, the original message can't be viewed. Once the message is viewed it is restored back to its encoded form instead of

saving it in its decoded format (to view the original message later user must save it in his/her PC and avoid repeated decoding of same data). When the user operates via his account for decoding he is asked for device authentication prior to decoding. If the password (device sensitive) recognises the device id then only decoding process gets activated else it shows a failure message,decoding not possible'Word time signal ensures shifting of bits in the shift register(in figure 5) thereby restoring the original bit sequence at the end of operation.

The statistics provided above is just based on theoretical calculations and needs experimentation to compute the data transfer time and whether any further modifications could be added (like data compression) for reducing the complexity of algorithm. As of now we can't claim it to be a 100% secured data transfer technique and can be applied only to text files attachments, or text mails. Picture files (graphics, RGB files)/AV files will not follow the above process while encoding.

Future Scope

Encryption mechanisms are getting increasingly complex in the face of evolving cyber threats. From key security and management to navigating complex regulatory landscapes, the road to comprehensive data protection is anything but straightforward. Although methods like SSL/TLS encryption, API keys, and passwords help keep data safe, they can only protect the data if they are kept secret. No matter how complex the encryption algorithm is, cybercriminals can quickly decrypt private data if the encryption key is no longer kept secret. The study carried out in this project can be extended to many other potential fields. Major possibility is to develop an error correcting system which can provide better performance, with less delay overhead, lower power requirements and less area consumption.

The study can be carried out by pipelining the existing codes in to an efficient form so that the delay overhead is reduced. Also, by changing the adders and other elements used in realization, the area can be reduced, by the proper implementation of the above indicated two ideas the power consideration can also be considerably reduced.

The Rand Shift AES algorithm introduces an innovative approach to enhance the security of data encryption by incorporating random shifting operations. Its future scope lies in several areas:

1. Security Enhancement: As cyber threats continue to evolve, there is a constant need for more robust encryption techniques. The RandShift AES algorithm offers an additional layer of security by introducing randomness into the encryption process, making it harder for attackersto exploit patterns in encrypted data. Its future lies in being further analyzed, tested, and potentially standardized for use in various security-critical applications.
2. Quantum Computing Resistance: With the emergence of quantum computing, traditional encryption algorithms like AES are at risk due to the potential future ability of quantum computers to break them using algorithms like Shor's algorithm. Techniques like RandShift AES could potentially offer resistance against quantum attacks by introducing additional complexity and randomness into the encryption process.
3. IoT and Edge Computing: With the proliferation of Internet of Things (IoT) devices and edge computing systems, there's a growing need for lightweight yet secure encryption algorithms. The RandShift AES algorithm could be optimized for resource-constrained devices while still providing a high level of security, making it suitable for securing communications and data inIoT and edge computing environments.
4. Standardization and Adoption: For widespread adoption, cryptographic algorithms often undergo standardization processes through organizations like NIST (National Institute of Standards and Technology). The future scope of RandShift AES involves further research, analysis, and potential standardization efforts to ensure its compatibility, interoperability, andsecurity across different platforms and applications.
5. Post-Quantum Cryptography: As researchers explore post-quantum cryptographic algorithms that are resistant to quantum attacks, RandShift AES could be considered as a candidate for inclusion in the suite of post-quantum cryptographic techniques. Its resistance toclassical cryptanalysis combined with its potential resistance to quantum attacks makes it an interesting candidate for securing data in a post-quantum computing era.
6. Cryptographic Research: The RandShift AES algorithm opens avenues for further cryptographic research. Researchers can explore variations, optimizations, and potential weaknesses of the algorithm to continuously improve its security and efficiency.

Additionally, it can inspire the development of new encryption techniques that leverage randomness and shifting operations for enhanced security.

In conclusion, the future scope of the RandShift AES algorithm is promising, with potential applications in enhancing security, resisting quantum attacks, securing IoT devices, and contributing to the advancement of cryptographic research and standardization efforts.

References

[1] K. Lim, J. Chang, T. Mudge, P. Ranganathan, S. K. Reinhardt, and T. F. Wenisch, “Disaggregated memory for expansion and sharing in blade servers,” ACM SIGARCH Comput. Archit. News, vol. 37, no. 3, pp. 267–278, 2009.

[2] M. Ware et al., “Architecting for power management: The IBM power7 approach,” in Proc. High

Perform. Comput. Archit. (HPCA), Jan. 2010, pp. 1–11.

[3] O. Mutlu, "The RowHammer problem and other issues we may face as memory becomes denser," in Proc. Conf. Design, Autom. Test Eur., 2017, pp. 1116–1121.

[4] A. Chen, "A review of emerging non-volatile memory (NVM) technologies and applications," Solid-State Electron., vol. 125, pp. 25–38, Nov. 2016.

[5] O. Mutlu, "Rethinking memory system design for data-intensive computing," in Proc. SAMOS, 2015, p. 1.

[6] N. H. Seong, D. H. Woo, V. Srinivasan, J. A. Rivers, and H.-H. S. Lee, "SAFER: Stuck-at-fault error recovery for memories," in Proc. 43rd Annu. IEEE/ACM Int. Symp. Microarchitecture, Dec. 2010, pp. 115–124.

[7] D. Strukov, "The area and latency tradeoffs of binary bit-parallel BCH decoders for prospective nanoelectronic memories," in Proc. Signals, Syst. Comput. (ACSSC), Oct. 2006,pp. 1183–1187.

[8] R. C. Bose and D. K. Ray-Chaudhuri, "On a class of error correcting binary group codes,"Inf. Control, vol. 3, no. 1, pp. 68–79, Mar. 1960.

[9] S. Schechter, G. H. Loh, K. Strauss, and D. Burger, "Use ECP, not ECC, for hard failures in resistive memories," ACM SIGARCH Comput. Archit. News, vol. 38, no. 3, pp. 141–152,2010.

[10] R. Maddah, R. Melhem, and S. Cho, "RDIS: Tolerating many stuckat faults in resistive memory," IEEE Trans. Comput., vol. 64, no. 3, pp. 847–861, Mar. 2015.

[11] R. Maddah, S. Cho, and R. Melhem, "Symbol shifting: Tolerating more faults in PCMblocks," IEEE Trans. Comput., vol. 65, no. 7, pp. 2270–2283, Sep. 2016.

[12] D. Kline, Jr., R. G. Melhem, and A. K. Jones, "Counter advance for reliable encryption in phase change memory," IEEE Comput. Archit.Lett., vol. 17, no. 2, pp. 209–212, Jul. 2018.

[13] M. K. Qureshi, A. Seznec, L. A. Lastras, and M. M. Franceschini, "Practical and securePCM systems by online detection of malicious write streams," in Proc. High Perform. Comput. Archit. (HPCA), Feb. 2011, pp. 478–489.

[14] S. Chhabra and Y. Solihin, "i-NVMM: A secure non-volatile main memory system with incremental

encryption," in Proc. 38th Annu. Int. Symp. Comput. Archit. (ISCA), Jun. 2011,pp. 177–188.

[15] M. Jalili and H. Sarbazi-Azad, "Endurance-aware security enhancement in non-volatile memories using compression and selective encryption," IEEE Trans. Comput., vol. 66, no. 7, pp. 1132–1144, Dec. 2017.

[16] S. Cho and H. Lee, "Flip-N-Write: A simple deterministic technique to improve PRAM write performance, energy and endurance," in Proc. IEEE/ACM Int. Symp. Microarchitecture, Dec. 2009, pp. 347–357.

[17] S. Mathew et al., "53 Gbps native GF (24) 2 composite-field AES-encrypt/decrypt accelerator for content-protection in 45 nm high-performance microprocessors," in Proc. VLSI Circuits (VLSIC), Jun. 2010, pp. 169–170.

[18] S. Haber and P. K. Manadhata, "Improved security for non volatile main memory," Tech.Discl. Commons, Feb. 2017. [Online]. Available: https://www.tdcommons.org/dpubs_series/396

[19] Z. Zhang, W. Xiao, N. Park, and D. J. Lilja, "Memory module-level testing and error behaviors for phase change memory," in Proc. 30th Int. Conf. Comput. Design (ICCD), Dec.2012, pp. 358–363.

[20] M. Soltani, M. Ebrahimi, and Z. Navabi, "Prolonging lifetime of nonvolatile last level caches with cluster mapping," in Proc. Int. Great Lakes Symp. VLSI, May 2016, pp. 329– 334.

[21] N. Binkert et al., "The gem5 simulator," ACM SIGARCH Comput. Archit. News, vol.39, no. 2, pp. 1–7, 2011.

[22] J. L. Henning, "SPEC CPU2006 benchmark descriptions," ACM SIGARCH Comput. Archit. News, vol. 34, no. 4, pp. 1–17, Sep. 2006.