IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 5, May 2024

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.379

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205181 |

A Review on Local Memory Bus Controller with ECC

Guruprasad R [1], Nitin Manohar Mishra [1], Rajendra Gurukar [1], Swathi [1], Dr. C M Patil [2], Dr. Geethashree [3]

Students, Department of ECE, Vidyavardhaka College of Engineering, Mysuru, India[1]

Professor & Head, Department of ECE, Vidyavardhaka College of Engineering, Mysuru, India[2]

Associate Professor, Department of ECE, Vidyavardhaka College of Engineering, Mysuru, India [3]

ABSTRACT: This project focuses on enhancing data reliability in computer systems by adding error correction code (ecc) features to the local memory bus controller (lmbc). the lmbc manages data flow between the cpu and system memory, with ecc technology crucial for spotting and fixing memory errors. the project highlights the importance of strong memory systems in modern computing, pointing out the risks of data corruption, system instability, and security issues linked to memory errors. the proposed solution involves integrating a hamming code-based ecc feature into the lmbc to boost data reliability, improving overall stability and security. the approach includes careful ecc algorithm selection, smooth integration with memory transport, and effective data validation and correction methods. this solution is especially useful for critical missions and high-performance computing tasks. in summary, the project aims to fortify computer systems, ensuring better data integrity through a comprehensive ecc-enabled lmbc design.

KEYWORDS: ecc (error correction codes) ,lmbc (local memory bus controller),data reliability, strong memory systems,critical computing.

I. INTRODUCTION

Error detection and correction (ECC) is a crucial feature implemented in high-reliability applications such as enterprise data storage systems and communication technologies like satellite receivers. In enterprise storage, ECC is integrated into memory caches within controllers to enhance system reliability by preventing data loss from single-point failures. By employing ECC, these systems can detect and correct errors, thereby safeguarding customer data without solely relying on disk arrays. Similarly, in communication applications like satellite receivers, ECC is indispensable for optimizing performance and cost efficiency. Rather than resorting to retransmitting data, which can be resource-intensive and impractical in certain contexts like satellite communication, ECC allows for error correction directly, improving overall system efficiency and reducing operational costs. In summary, ECC plays a pivotal role in maintaining data integrity, minimizing disruptions, and enhancing performance in critical high-reliability environments such as enterprise storage and communication systems.

II.LITERATURE SURVEY

[1] Miguel Costa and Srikanth Beerla's propose an imaginative approach to address eFuse unwavering quality and deserting challenges by joining a custom ECC module with a BISR (Built-In Self-Repair) controller. This strategy points to upgrade blunder discovery and rectification capabilities in eFuses, promising made strides proficiency and unwavering quality in memory repair plans. By decreasing run overhead and exhibiting potential for differing industry applications, their procedure offers a groundbreaking arrangement to complexities related with eFuses, with suggestions for memory frameworks in different segments.

[2]Jungrae Kim's introduction of the All-Inclusive ECC (AIECC) marks a significant advancement in ensuring robust memory protection by addressing signal vulnerabilities and enhancing end-to-end data integrity. AIECC's ability to detect nearly 100% of CCCA errors and prevent transmission errors from compromising memory data integrity without additional overhead underscores its value as a comprehensive solution for data security.

[3] Shalini Ghosh's strategy to minimize control usage in ECC circuitry, focusing on SEC-DED codes like Hamming and Hsiao codes, represents a critical step towards optimizing memory systems. Supported by industry giants like

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205181 |

Hewlett-Packard Enterprise and the National Science Foundation, this research delves into the application of simulated annealing and genetic algorithms to reduce control while maintaining area and delay constraints. The significant control reductions demonstrated for various error-correcting codes highlight the potential for enhanced efficiency and performance in memory systems.

[4] Shalini Ghosh's approach to minimize control usage in ECC circuitry, focusing on SEC-DED codes like Hamming and Hsiao codes, is supported by industry leaders like Hewlett-Packard Enterprise and the National Science Foundation. By applying simulated annealing and genetic algorithms, this research achieves substantial reductions in control complexity while maintaining area and delay constraints, promising enhanced efficiency and performance in memory systems.

[5] Aniruddha N. Udipi's proposal LOT-ECC, a unused memory mistake redress strategy by Aniruddha N. Udipi, progresses unwavering quality over existing strategies whereas utilizing less control and decreasing idleness. It isolates blunder discovery and redress utilizing basic codes, works with standard memory and frameworks, and can be amplified to more extensive memory components. LOT-ECC offers critical control investment funds, adaptability, and mistake redress whereas keeping up execution.

In general, a writing study on nearby memory transport controllers utilizing ECC plays a imperative part in progressing inquire about, cultivating development, and tending to the challenges related with guaranteeing information keenness and unwavering quality in memory frameworks.

III. METHODOLGY

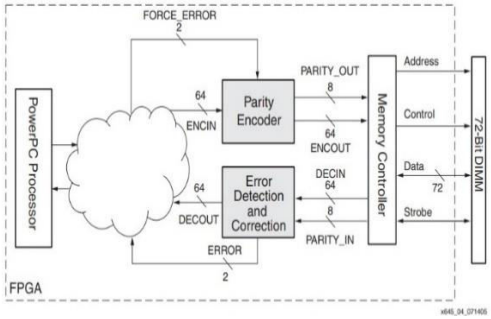

Figure 1: ECC in a Memory system *[Single Error Correction and Double Error Detection – Simon Tam]*

Error Correction Code (ECC) is used in digital communication and storage systems to detect and correct errors that occur during data transmission or storage. ECC involves adding redundant bits to the data to enable error detection and, in some cases, error correction without the need for retransmission. The primary purpose of ECC is to ensure data integrity and reliability by employing mathematical techniques such as parity checks, checksums, or more advanced methods like Reed-Solomon codes or Hamming codes. These methods vary in complexity and capability, with some able to detect errors only and others capable of both error detection and correction, making ECC a critical component in maintaining data accuracy in digital systems.

A. Hamming Code: The ECC capacities depicted in this application note are made conceivable by Hamming code, a moderately straightforward however effective ECC code. It includes transmitting information with different check bits (equality) and translating the related check bits when accepting information to identify blunders. The check bits are parallel equality bits produced from XORing certain bits within the unique information word. In the event that bit error(s) are presented within the codeword, a few check bits appear equality blunders after interpreting the recovered codeword. The combination of these check bit blunders show the nature of the mistake. In expansion, the position of any single bit mistake is distinguished from the check bits **.**

The Hamming codeword could be a concatenation of the initial information and the check bits (equality). It is depicted by an requested set (d + p,d) where d is the width of the information and p is the width of the equality. The equality framework [P] can be communicated as:

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205181 |

[P] = [D] • [G]

Where [D] is the information lattice and [G] is the generator network.

[C]. [G] = [I:C]

Table:1 -7-Bit parity bits for 32 bits data :

Participating Data Bits	Generated Check Bits						
	CB0	CB1	CB2	CB3	CB4	CB5	CB6
0	√	√					√
1	√		√				√
2		√	√				√
3	√	√	√				√
4	√			√			√
5		√		√			√
6	√	√		√			√
7			√	√			√
8	√		√	√			√
9		√	√	√			√
10	√	√	√	√			√
11	√				√		√
12		√			√		√
13	√	√			√		√
14			√		√		√
15	√		√		√		√
16		√	√		√		√
17	√	√	√		√		√
18				√	√		√
19	√			√	√		√
20		√		√	√		√
21	√	√		√	√		√
22			√	√	√		√
23	√		√	√	√		√
24		√	√	√	√		√
25	√	√	√	√	√		√
26	√					√	√
27		√				√	√
28	√	√				√	√
29			√			√	√
30	√		√			√	√
31		√	√			√	√

***B. Data Correction*:** Within the information rectification arrange, the mask is XOR'd at the side the first approaching information to flip the blunder bit to the proper state, in the event that required. When there are no bit errors or twofold bit blunders, all the cover bits are zeros. As a result, the approaching information goes through the ECC unit without changing the first information.

***C. Error Diagnostics*:** In expansion to showing the blunder sort, the reference plan moreover bolsters symptomatic mode. Single, numerous, and triple bit mistakes can be presented to the yield codeword.

- **Force error = 00:** This is the normal operation mode. No bit error has been imposed on the output of the encoder.
- **Force error = 01:** Single bit mistake mode. One bit is turned around (0 to 1 or 1 to 0) within the codeword at each rising edge of the clock.
- **Force error = 10:** Named two fold bit mistake mode. Two continuous bits are switched, within the codeword at each rising edge of the clock

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205181 |

IV. IMPLEMENTATION

A. ***ECC Encoder:*** Commence with the plan of an ECC encoder that takes 32 bit information as input and yields a 39-bit encoded information word, which incorporates a 7-bit mistake redressing code. This requires selecting an fitting ECC calculation, ordinarily a Hamming code or for encoding the information.

B. ***ECC Decoder:*** Make an ECC decoder that can interface with the memory controller to get the 39-bit information, which incorporates the ECC. The decoder's rationale will utilize the ECC to adjust single-bit mistakes and distinguish double-bit blunders, shown by the Mistake flag. It's pivotal that the decoder works in real-time to preservet all information throughput and unwavering quality.

C. ***Configure of Memory Model:*** Implement a memory module that supports a 39 bit word length for each memory cell, accommodating the 32-bit data along with the 7 bit ECC. The module should be capable of interfacing with the memory controller to receive encoded data for storage (ENCOUT) and send out data for error checking (DECIN). Communication System.

D. ***Development of Error Status Monitoring:*** Develop a monitoring system that can track the error status of each data operation. This system will need to interpret the ERROR signal from the decoder and provide feedback to the memory controller, which can then take appropriate action based on the type of error detected.

E. ***Integration of Memory Controller:*** Coordinated all the planned modules beneath the control of the Memory Controller. The controller ought to oversee the stream of information to and from the memory module, the encoding and interpreting forms, and the mistake dealing with rationale. It ought to moreover be able to prepare FORCE_ERROR signals to mimic blunders for testing purposes.

F. ***Final Deployment and Monitoring:*** Deploy the system into its intended operational environment. Monitor its performance over time to ensure it meets the required specifications and to identify any areas for further improvement. Feedback from this phase may lead to subsequent iterations of design refinement.

V. RESULTS

A. ***Design of ECC Encoder*** :The system shapes 32-bit data at the side a 1-bit ECC engage hail. When the ECC engage hail is set to basis 1, the system performs a uniformity calculation. On the off chance that the uniformity calculation comes approximately in 0, the balance bits are set to 0. The ECC surrender comprises of 39 bits, comprising the beginning 32-bit data and an additional 7 bits for uniformity.

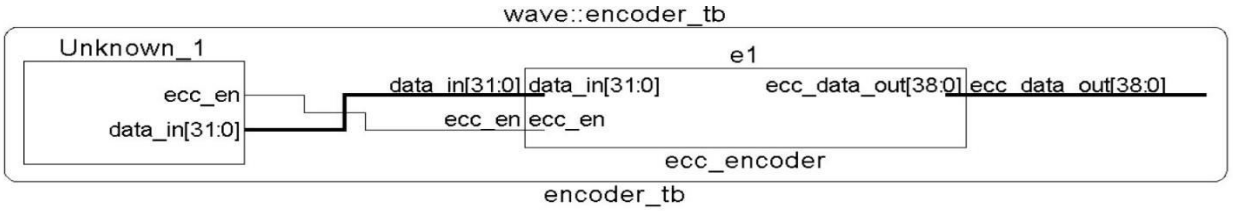

Fig 2: ecc encoder module

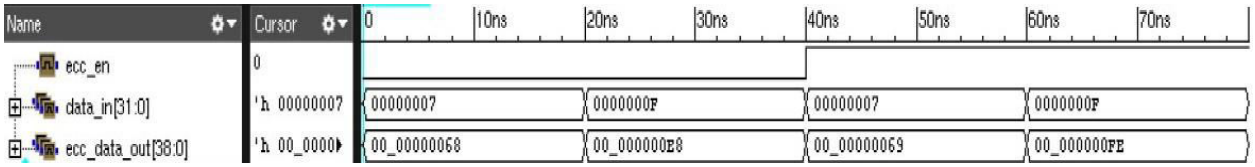

Fig 3: ecc encoder waveform

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205181 |

B. Configure of Memory Model: The memory square incorporates a few control and information signals: clock (offbeat), a reset (rst), perused empower (read_en), type in empower (write_en), compose address (write_address), and studied address (read_address). When the reset flag (rst) is tall, the memory information is reset to zero, and operations such as composing and perusing are debilitated until the reset flag returns to moo. When the compose empower flag (write_en) is declared (tall), the memory composes the information display at the desired type in address (write_address) into the memory piece. So also, when the examined empower flag (read_en) is attested (tall), the memory recovers the information from the memory square based on the desired studied address (read_address). The output of the memory piece gives the information examined from the desired address.

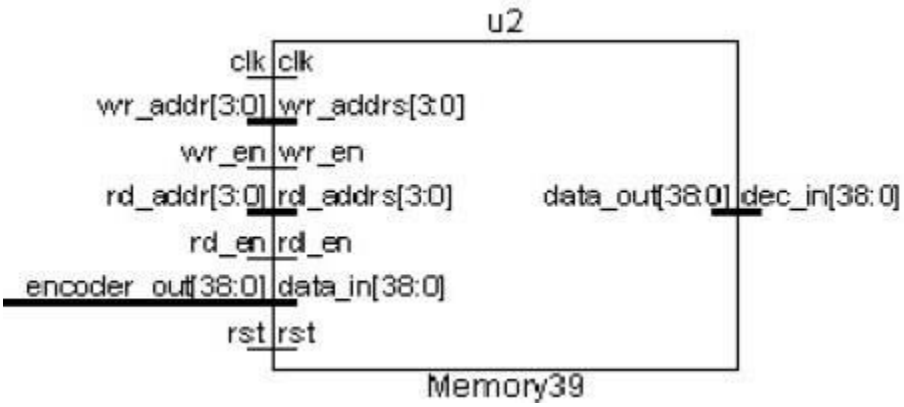

Fig 4: memory module

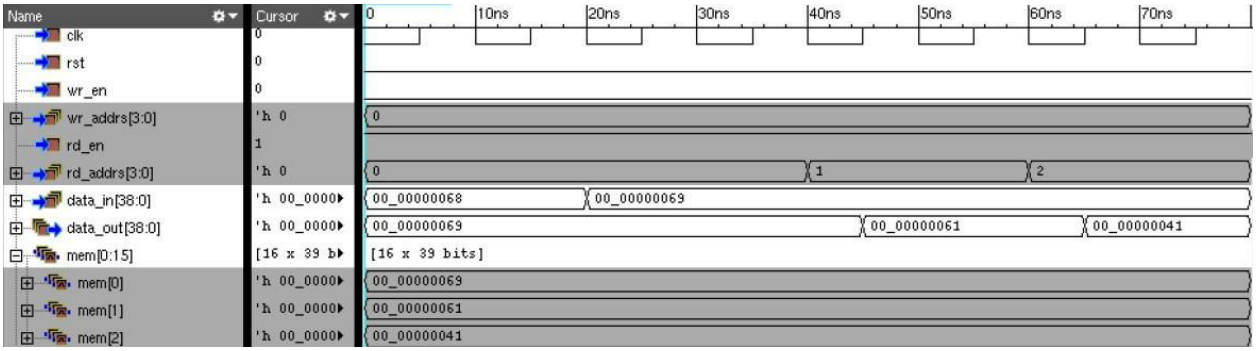

Fig 5: memory module waveform

C. Construction of ECC Decoder: In an ECC decoder square, the input is sourced from the yield of the memory piece at the side an extra input, ecc_en. When ecc_en is moo, no operations are performed by the decoder. In any case, when ecc_en is tall, the decoder forms the input and generates two yields:

dec_out and gen. dec_out could be a 32-bit data output speaking to the redressed or decoded information from the memory. On the other hand, gen may be a 7-bit output that gives data around the blunder recognized amid interpreting. This data demonstrates whether a 1-bit mistake or a 2-bit mistake was recognized within the input information stream. The ECC decoder is significant for mistake location and rectification in computerized frameworks, improving information unwavering quality and astuteness.

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205181 |

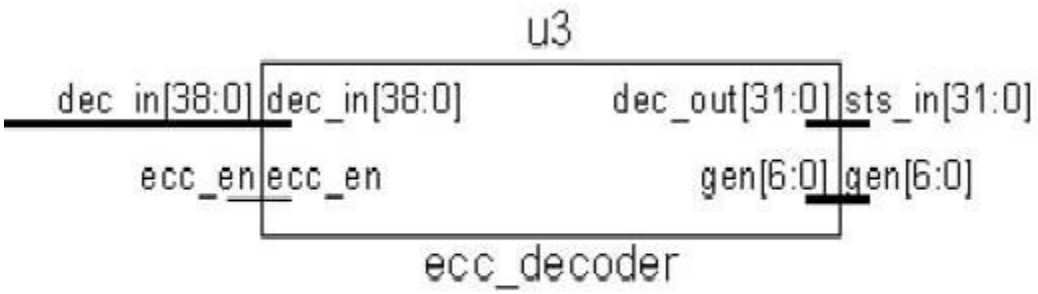

Fig 6 : decoder module

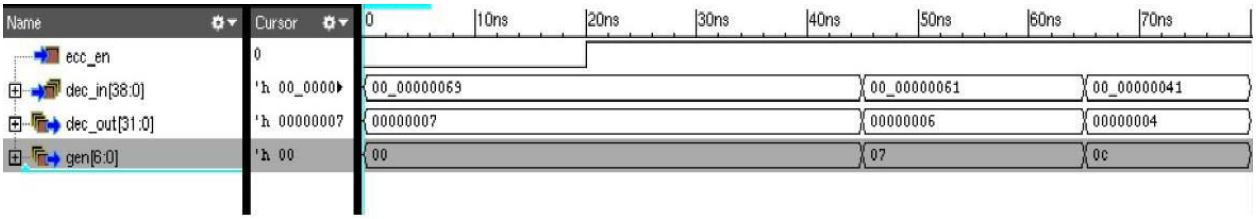

Fig 7: decoder module waveform

D. Development of Error Status Monitoring: The ecc status module has three inputs:
two inputs taken from the decoder yields, dec_out and gen, and an extra input ecc_en. In the event that ecc_en is zero, the module basically passes the input information through without adjustment. In any case, in case ecc_en is tall, the module performs blunder rectification by checking for single-bit blunder discovery and adjustment or two-bit mistake discovery inside the 32-bit information.

The output from the ecc status module includes two components: sts_out and ind.

- sts_out : This is the ECC status output, representing the corrected or unchanged data based on error detection and correction
- ind : This is a 2-bit output that indicates the error status as follows:

00: No error detected.
01: Single-bit error detected and corrected.
10: Two-bit error detected
11: Invalid state.

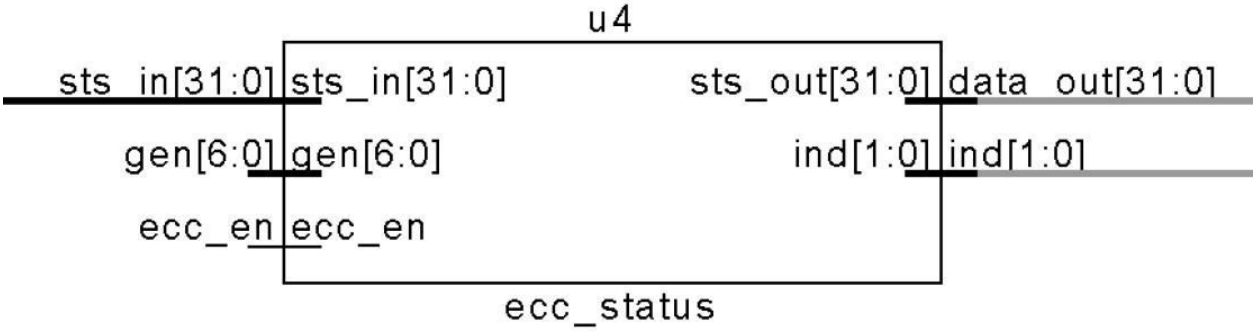

Fig 8: ecc_status module

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205181 |

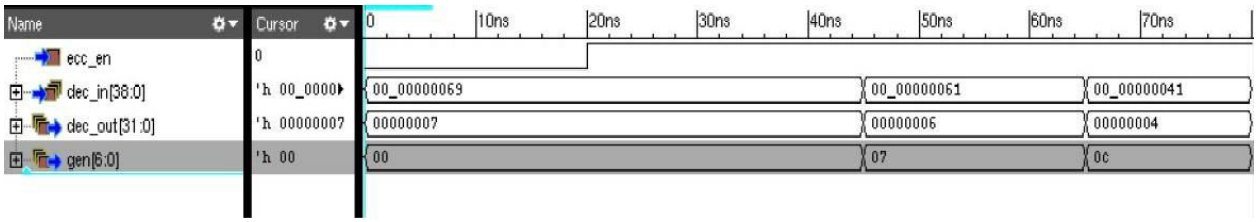

Fig 9 : ecc status module waveform

E. Integration of Memory Controller: The memory controller module coordinating information capacity and recovery operations inside a framework utilizing different input signals. It reacts to 'data_in' for compose operations ('wr_en' and 'wr_addr') to store information at indicated memory addresses and peruses information from memory addresses ('read_addr' and 'read_en'), yielding the recovered information as 'data_out'. The controller moreover bolsters ECC (Blunder Redress Code) usefulness, actuated by 'ecc_en', which can distinguish and possibly rectify blunders amid studied operations, giving status criticism through the 'ecc_sts' yield and 'ind' flag to demonstrate blunder conditions. This module plays a basic part in overseeing memory operations, guaranteeing information astuteness and unwavering quality inside the framework.

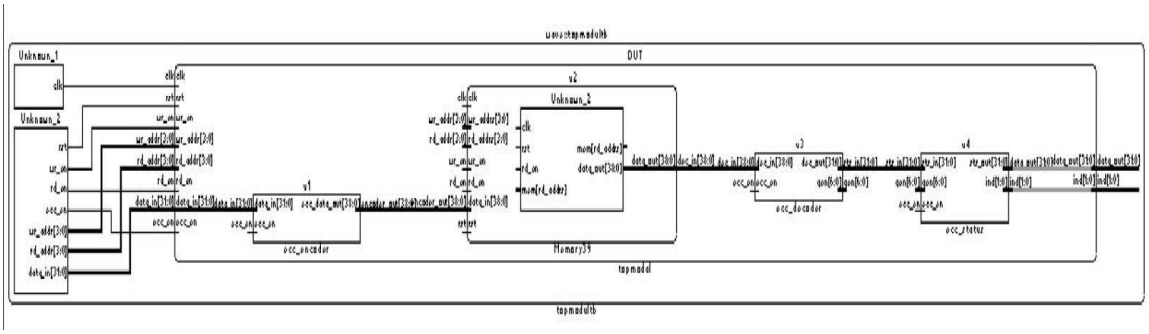

Fig 10: Top module

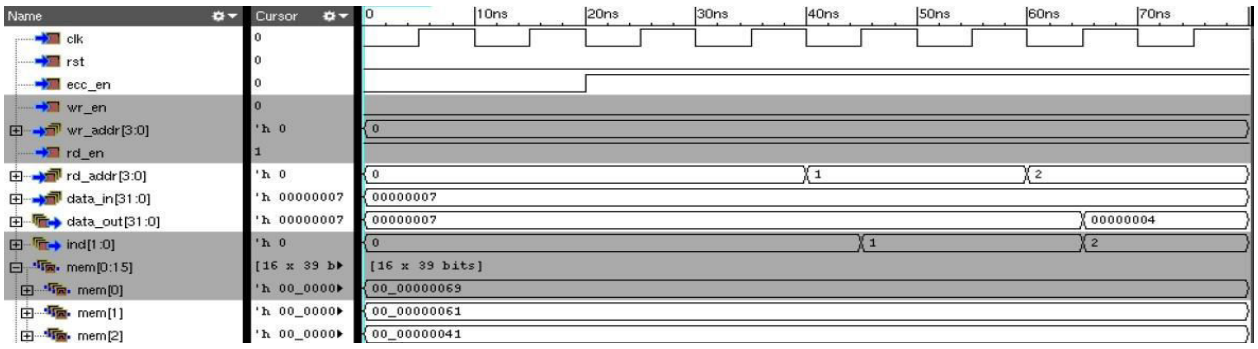

Fig 11 : memory controller waveform3

VI. CONCLUSION

Coordinated all the planned modules beneath the control of the Memory Controller. The controller ought to oversee the stream of information to and from the memory module, the encoding and interpreting forms, and the mistake taking care of rationale. It taught to moreover be able to handle FORCE_ERROR signals to recreate mistakes for testing purposes.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205181 |

VII. FUTURE WORK

Looking ahead, the project could expand into several promising areas. Future work might involve the development of more advanced ECC algorithms to handle multiple bit errors more efficiently, possibly integrating machine learning to predict and preempt error patterns. Additionally, reducing the latency introduced by ECC checks could further enhance system performance. Energy efficiency is another avenue, where future iterations could aim for a balance between error correction robustness and power consumption. Finally, adapting the design for emerging memory technologies like MRAM or 3D XPoint could ensure the project's relevance in the next generation of computing hardware.

REFERENCES

[1] Miguel Costa, Srikanth Beerla,"Enabling ECC and Repair Features in an eFuse Box for Memory Repair Applications, "Published in: 2021 22nd International Symposium on Quality Electronic Design (ISQED), https://doi.org/10.1109/ISQED51717.2021.9424327

[2] Jungrae Kim, Michael Sullivan, Sangkug Lym, Mattan Erez, "All-Inclusive ECC: Thorough End-to-End Protection for Reliable Computer Memory," ACM SIGARCH Computer Architecture News, Volume: 44, Issue 3, pp 622–633, https://doi.org/10.1145/3007787.3001203

[3] S. Ghosh, S. Basu, N.A. Touba, "Reducing power consumption in memory ECC checkers," Published in: 2004 International Conference on Test
https://DOI:10.1109/TEST.2004.1387407

[4] R. Männer, O. Stucky, "Fault-Tolerant Data Transfer in a Multiprocessor System by Forward and Backward Hardware Error Recovery," The Computer Journal, Volume 35, Issue 4, August 1992, Pages 361–368 ; https://doi.org/10.1093/comjnl/35.4.361

[5]Aniruddha N. Udipi, Naveen Muralimanohar, Rajeev Balsubramonian, Al Davis, Norman P. Jouppi, "LOT-ECC: localized and tiered reliability mechanisms for commodity memory systems, " ACM SIGARCH Computer Architecture News, Volume 40, Issue-3, pp 285–296, https://doi.org/10.1145/2366231.2337192.

[6] Xiaochen Guo, Mahdi Nazm Bojnordi, Qing Guo; Engin Ipek, "Sanitizer: Mitigating the Impact of Expensive ECC Checks on STT-MRAM Based Main Memories ," Published in: IEEE Transactions on Computers (Volume: 67, Issue: 6, 01 June 2018), https://doi:10.1109/TC.2017.277915

[7] Sheng Li, Doe Hyun Yoon, Ke Chen, Jishen Zhao, Jung Ho Ahn, Jay B. Brockman, Yuan Xie, "Sanitizer: Mitigating the Impact of Expensive ECC Checks on STT-MRAM Based Main Memories ," Published in: IEEE Transactions on Computers (Volume: 67, Issue: 6, 01 June 2018), https://doi:10.1109/TC.2017.2779151

[8] Lucian Cojocar, Kaveh Razavi, Cristiano Giuffrida, Herbert Bos, "Exploiting Correcting Codes: On the Effectiveness of ECC Memory Against Rowhammer Attacks ," Published in: 2019 IEEE Symposium on Security and Privacy (SP) , https://doi:10.1109/SP.2019.00089

[9] Seong-Lyong Gong, Minsoo Rhu, Jungrae Kim, Jinsuk Chung, Mattan Erez , "CLEAN-ECC: high reliability ECC for adaptive granularity memory system ," MICRO-48: Proceedings of the 48th International Symposium on Microarchitecture , December-2015 , Pages-611–622 , https://doi.org/10.1145/2830772.2830799

[10] S. Pontarelli, M. Ottavi, A. Salsano , "Error Detection and Correction in Content Addressable Memories ," Published in: 2010 IEEE 25th International Symposium on Defect and Fault Tolerance in VLSI Systems , https://doi:10.1109/DFT.2010

[11] Salvatore Pontarelli, Marco Ottavi, " Error Detection and Correction in Content Addressable Memories by Using Bloom Filters ," Published in: IEEE Transactions on Computers (Volume: 62, Issue: 6, June 2013) , Page(s): 1111 - 1126 , https://doi:10.1109/TC.2012.56

[12] Nandivada Sridevi, K. Jamal, Kiran Mannem ,"Implementation of Error Correction Techniques in Memory Applications ,"Published in: 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), https://DOI:10.1109/ICCMC51019.2021.9418432

[13] Simon Tam "Single Error Correction and Double Error Detection" Published in August 9, 2006 (Virtex-II Pro, Virtex-4, and Virtex-5)
www.xilinx.com

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details