Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 13, Issue. 2 : 2022
ISSN : **1906-9685**

MULTIPLE BIT ERROR TOLERANT SRAM-BASED TCAM FOR HIGH-SPEED NETWORK OPERATIONS

[#1]KAMA KANISHKA, ***Asst. Professor,***
[#2]KOTICHINTHALA NEETHIKA, ***Asst. Professor,***
Department of Electronics Communication Engineering,
Sree Chaitanya Institute of Technological Sciences, Karimnagar, Ts.

Abstract: Ternary Content Addressable Memory (TCAM) is frequently used in software that requires quick searching. Unlike SRAM-based TCAMs, which are made with a field-programmable gate array (FPGA), conventional TCAMs are made with an application-specific integrated circuit (ASIC). You can use CAM to access data without knowing its exact location or filename. CAM is frequently described as a binary format that only uses 0s and 1s. In addition to CAM, there is ternary CAM (TCAM), a more complicated form of memory. Bits of any value can be stored, including zero and one. When using FPGA for fast applications, there is a chance of making tiny mistakes. Such flaws may diminish the method's effectiveness. Soft mistakes in SRAM-based TCAMs are difficult to avoid without considerably slowing down the search process. The current technique employs single-bit upset detection and repair, which entails adding a single parity bit to each word to detect faults. The proposed method simplifies the detection and correction of MBUs (multiple bit upsets). Multi-bit defects can be discovered using one of three parities: row, column, or diagonal parity. They are totaled as a group. Mistakes are automatically corrected so that the search can continue to function properly.
***Keywords*:** Field-programmable gate array (FPGA), soft errors, static random access memory (SRAM)-based ternary content- addressable memory (TCAM).

1. INTRODUCTION

Memory is a form of brain in the same way that the mind's brain is. It serves as a central location for maps, directions, and other essential information. The computer's memory stores information and instructions for processing it. There are numerous memory subsets to consider. Each location or cell has its own address, which can range from 0 to the memory capacity minus 1. Data access memory is classified into four types. The four major types of computer memory are RAM (Random Access Memory), DAM (Direct Access Memory), CAM (Content Addressable Memory), and SAM (Sequential Access Memory). RAM is another name for read-write memory. Because RAM is a "fragile memory," data stored in it is lost if the power is quickly turned off. RAM comes in several varieties, the most popular of which are DRAM (Dynamic Random Access Memory) and SRAM (Static Random Access Memory). SAM retrieves data from storage during an operation. RAM, on the other hand, permits data access in any order. Sequential access devices are often compatible with both magnetic and optical recording media. The DAM stores data in material pieces. These features are simple to use. However, the information required for these areas is freely accessible. There are memories of driving on a magnetic tape and in a blind spot. Associative memory is another name for CAM. The two most common types are Ternary CAM and Binary CAM.

Content-addressable memory (CAM) has high search speed because it can search for several pieces of information in a single cycle. A binary CAM can only store or retrieve data in the "0" or "1" states. A ternary CAM (TCAM), which is commonly used in network systems for

traffic sorting and filtering, displays information in three states: "0," "1," and "x," which stands for "do not care." Figure 1 demonstrates a simple NOR-based architecture for a 4 x 5-bit ternary CAM. A CAM holds the routing table, which is used when an address is required. Four horizontal words are formed by the CAM core cells. Each word consumes five bits. Core cells have both memory and comparison circuits. The vertical search lines in the diagram send search data to the CAM cells. The horizontal lines that span the length of the array represent search results that are relevant to the term in each row. A match is indicated by a green line in the CAM texts, whereas a mismatch is represented by a red line. An encoder accepts match lines as input and generates addresses that can be utilized to find things like the match site.

The flexibility required by software defined networking (SDN) and OpenFlow network accelerators for big data is provided by static random-access memory (SRAM)-based field programmable gate array (FPGA) technology. Single-event disturbances (SEUs) can be generated by high-energy neutron particle disruptions in SRAM-based FPGA-based circuits [3]. On-chip embedded memory is regarded as the most risky SEU technology due to its small size and close proximity of memory cells. An SEU in embedded memory will corrupt data until the corrupted data is cleared. SBUs (single-bit upsets) and MBUs (multiple-bit upsets) are two types of single-event shocks. Cuckoo hashing can be used to quickly and cheaply build binary CAMs on FPGAs. The TCAM function is typically determined by the block RAM (BRAM) and scattered RAM (distRAM) of an SRAM-based FPGA. It is possible that the incorrect match address will be returned owing to a temporary error, resulting in a false match or mismatch. When a soft error occurs, the offending SRAM word must be changed so that the correct data from lookups may be accessed. It is difficult to maintain the critical path search speed or latency in SRAM-based TCAM systems.

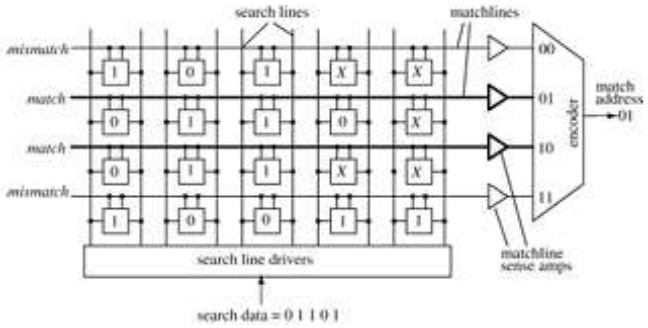

Fig. 1 TCAM Architecture

The paper describes a rapid, low-cost, and simple way for protecting SRAM-enabled TCAMs without slowing down search speed. The suggested method is a low-cost solution for detecting and repairing errors induced by multibit upsets (MBUs). In addition to the typical row, column, and diagonal tests, multi-bit mistakes are verified using three alternative parities. To maintain a fast search rate, error correcting occurs automatically in the background. It offers a low-cost, low-effort, and low-complexity means of protecting SRAM-enabled TCAMs in the existing system without severely reducing search performance. To detect errors with minimal overhead and processing time, a simple single-bit parity check is used. The error-correction technique can cure soft faults by utilizing the supplemental binary-encoded TCAM table stored for updating in SRAM-based TCAM solutions. Because of its fast search performance, the error-correction system can be left running in the background, allowing for concurrent searches.

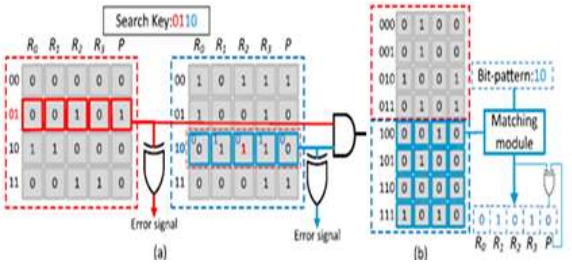

Fig. 2 Currently available ER-TCAM that aids in mistake detection include (a)

Making amends for previous mistakes

Figure 2 depicts the basic notion of the ER-TCAM technique. The TCAM table is stored in a binary-encoded 8-by-4 SRAM that is split across two SRAMs to ensure that matching data is always available. To locate an SBU, the ER-TCAM adds a parity bit to each SRAM word.

As illustrated in Figure 2(a). Error checking is performed on SRAM bits read during lookups. By accessing the redundant information recorded in the binary-coded TCAM table, the ER-TCAM corrects misspelled words. The second (00101) and third (01110) words are read when the search key (0110) is pressed on the SRAMs indicated in figure (a). An SBU exists because the match bits of the second SRAM word (011010) do not match the anticipated parity, as illustrated in Figure 2(a). The binary-encoded data from each TCAM segment is stored in SRAM words that are read progressively by the ER-TCAM. The error-correction vector (ECV) is formed by combining the proper bit pattern from the read SRAM words with the parity (01010). After determining the ECV, it is overwritten on top of the corrupted SRAM word.

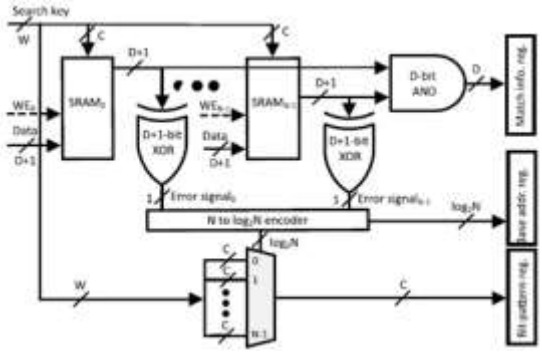

Fig. 3 Look at existing problems in the ER-TCAM layout.

Figure 3 shows an example of error detection in ER-TCAM. When a search key is pressed, the bits of the read SRAM words are EX-ORed together to give an error signal. The TCAM design turns the error information from the N SRAMs into a log2N-bit error code. This code is used in a certain way to locate any SRAM fault. The error code and search key bit patterns are transmitted to the error-correction module. Schematic of an error-correcting tethered camera array (ER-TCAM). The binary-encoded TCAM table data is stored in an SRAM, which is linked to a read/write controller, an address generating unit (AGU), and an ECV compute unit. The most significant bits of the SRAM ID, the log2N bits, indicate the start of the sub-block in SRAM that corresponds to the address. The AGU can utilize this to access the binary-encoded words in the TCAM table.

Deducing the match bits and parity bit from the C-bit structure of the read TCAM words takes D clock cycles. These two bytes make up the ECV. The write enable high signal is subsequently sent by the read/write controller to the associated SRAM, which writes the computed ECV over the corrupted SRAM word. SRAMs must be aware that the TCAM feature can also be used for retrieval operations during the error-correction process in order to do search operations; this is where the ER-TCAM comes in. The ER-TCAM configures these SRAMs as simple dual-port RAM with simultaneous reading and writing. The error correcting procedure in the ER-TCAM fully overlaps with the search operations after the ECV is computed and written to the write port of SRAM. Soft errors are conceivable in the SRAM that stores the binary-encoded TCAM table, but they are far less prevalent than in SRAMs that achieve ternary content addressable memory.

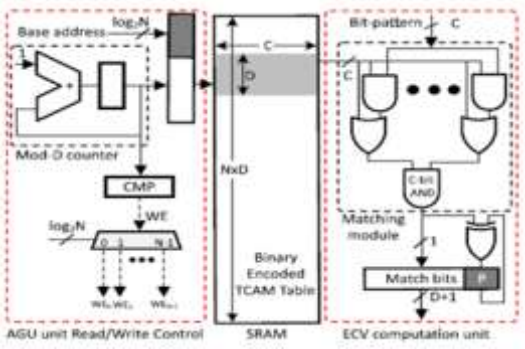

Fig. 4 The ER-TCAM tool is now being used to correct errors.

2. TERNARY CONTENT ADDRESSABLE MEMORY

TCAMs (Ternary Content Addressable Memories) are often used by network nodes to classify data packets. They are required for the establishment of software-defined networks (SDN), the transport of data packets, and the maintenance of security. The bits held in memory are susceptible to corruption as a result of soft mistakes, which is a problem. The memory are safeguarded by using a parity check to detect mistakes and a mistake correction mechanism to rectify them. This does, however, demand more memory per word. This research looks into the topic of safeguarding TCAM simulation memories. This approach is utilized to ensure that the

SRAM-Based TCAM Design is perfect by protecting against soft faults and offering an error repair mechanism with a fast response time, cheap cost, and powerful search performance.

SRAM based TCAM on FPGAs

Because of their on-chip SRAM memories, modern FPGAs use TCAM systems. When you put a "1" in RAM[0], the TCAM is in the "0" state; when you put a "1" in SRAM[1," the price is "1," and when you put a "1" in both SRAM[0] and SRAM spots, the TCAM is in the "x" state. A 1-bit SRAM with 2C locations is frequently used to generate a C-bit TCAM pattern. Each TCAM table word is compared to all conceivable C-bit patterns in the SRAM words, and a match or mismatch is determined. As a result, a table of B-words in a TCAM with a C-bit width can be built using a 2C-position B-bit wide SRAM. SRAM-based TCAMs require more memory than large TCAM bit patterns, hence researchers have partitioned them. TCAM can be created on FPGAs in a variety of ways.

Error Detection and Correction in TCAM

Modern electronics, particularly memories, are prone to a wide range of unintentional malfunctions. If a minor error alters the data in the memory's bits, the system may crash. Soft error rates are often quite low in terrestrial applications. A 65-nm static random-access memory (SRAM) memory was expected to have 109 bit errors per year.

Even a very small error rate is reason for concern in mission-critical applications such as communication networks, where network components such as routers must be extremely trustworthy and available at all times. As soft flaws are a key concern, measures for reducing errors are used when developing routers and other network components. Error checking and repair codes are usually required to avoid forgetting.

PROPOSED ERROR TOLERANT TCAM

Ternary Content Addressable Memory (TCAM) is a type of memory built primarily for fast searching. Unlike SRAM-based TCAMs, which are made with a field-programmable gate array (FPGA), conventional TCAMs are made with an application-specific integrated circuit (ASIC). You can use CAM to access data without knowing its exact location or filename. Because of its restricted storage capacity, CAM is also known as binary CAM. Ternary CAM (TCAM) is a higher-level CAM memory that can hold values other than 0 and 1. Soft mistakes can develop when FPGA is used in rapid systems. Such flaws may diminish the method's effectiveness. Soft mistakes in SRAM-based TCAMs are difficult to avoid without considerably slowing down the search process.

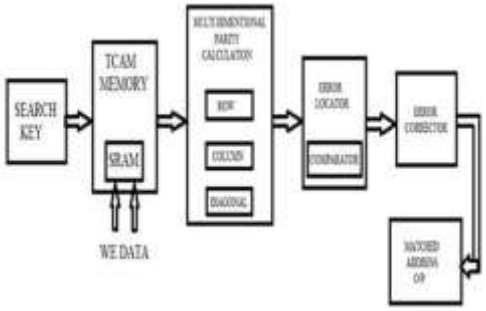

Fig.5 TCAM block error-proof layout recommended

This method makes use of single-bit disturbance detection and rectification. The proposed method simplifies the detection and correction of MBUs (multiple bit upsets). To identify faults across many bits, row parity, column parity, and diagonal parity are all employed. To maintain a fast search rate, error correcting occurs automatically in the background. Figure 5 depicts a block schematic of the suggested approach. When the search key is pressed to initiate a search, the memory unit scans the SRAM for matching words. After that, the corresponding parities are determined. When there is a discrepancy, the problem is detected. When mistakes are found, the words are routed to a section that alters them. The update is then applied on top of the repaired memory.

3. MULTIBIT UPSETS PROTECTION

In a normal microprocessor, parity bits are inserted to each memory location (word) accessed. This is done in order to detect any errors. Each memory location contains an error checking code that detects errors and speeds up access. In this situation, each operation would have to check every possible additional entry for the required common parity bit(s). Despite

the fact that FPGA faults are frequently discovered during maintenance. The error checking system can see everything within a configuration frame in this case. The proposed error detection approach involves significant parity computations. Errors can be found by taking advantage of these parity flaws. If the calculated parity differs from the produced parity, this is considered an error. When errors are found, the text is reported to the appropriate department for rectification. The lines that have been repaired are saved in memory.

4. RESULTS AND DISCUSSION

Simulated result of existing TCAM

An error-resistant TCAM is created to detect and rectify single-bit mistakes. To begin, an error-detection module must be created. The error correction system is activated when an error is identified. SRAMs must be aware that the TCAM feature can also be used for retrieval operations during the error-correction process in order to do search operations; this is where the ER-TCAM comes in. The error correcting procedure in the ER-TCAM fully overlaps with the search operations after the ECV is computed and written to the write port of SRAM. However, for the time being, this method can only detect and rectify single bit problems. In other words, this technique fails if more than one error happens.

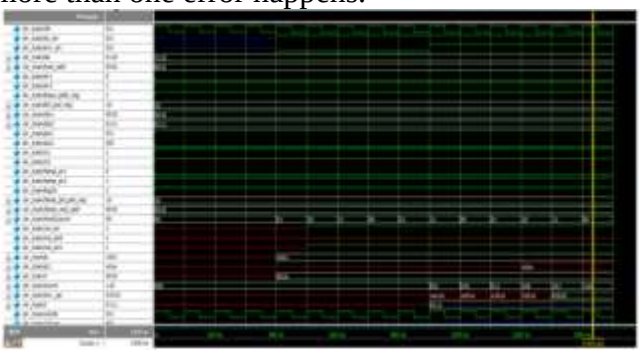

Fig.8 Simulated result of existing TCAM

Simulated result of proposed ER-TCAM

A TCAM that is incapable of committing multi-bit errors has been created to aid in the detection and rectification of such problems. This is the first of its kind error detection tool. Row parity, column parity, and diagonal parity are the three types of parity computations used in this situation. The error correction system is activated when an error is identified. We employ the error locator and gate processes to correct errors. Multibit errors can be identified and repaired using the supplied method.

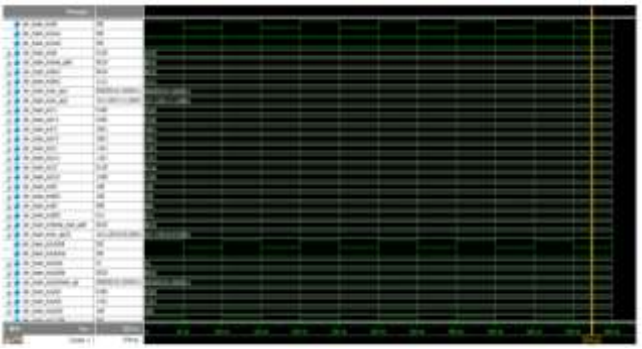

Fig.9 Simulated result of proposed ER-TCAM

5. CONCLUSION

Because the original TCAM data is saved on the chip for updating, it can be utilized to detect and correct errors in SRAM-based TCAMs. The current error detection method employs single-bit parity to reduce critical path latency and circuit complexity. This technology, also known as ER-TCAM, corrects faults in SRAM-based TCAMs by using the binary-encoded TCAM table. SRAMs with the TCAM capability perform the error repair method in the background. This assures that the chosen error correction strategy has no effect on the data route. The proposed method may detect and correct multibit faults in SRAM-based TCAM using three parities. Fixing bugs in the background helps search engines perform properly.

REFERENCES

P. He, W. Zhang, H. Guan, K. Salamatian, and G. Xie, “Partial order theory for fast TCAM updates,” IEEE/ACM Trans. Netw., vol. 26, no. 1, pp. 217–230, Feb. 2018.

1. W. Fu, T. Li, and Z. Sun, “FAS: Using FPGA to accelerate and secure SDN software switches,” Secur. Commun. Netw., vol. 2018, Jan. 2018, Art. no. 5650205.
2. S. T. Li, H. Liu, and H. Yang, “Design and characterization of SEU hardened circuits for SRAM-based FPGA,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 27, no. 6, pp. 1276–1283, Feb. 2019.
3. T. Li, H. Yang, H. Zhao, N. Wang, Y. Wei, and Y. Jia, “Investigation into SEU effects and hardening strategies in SRAM based FPGA,” in Proc. 17th Eur. Conf. Radiat. Effects Compon. Syst. (RADECS), 2019, pp. 1–5.

4. A. Ramos, R. G. Toral, P. Reviriego, and J. A. Maestro, "An ALU protection methodology for soft processors on SRAM based FPGAs," IEEE Trans. Comput., vol. 68, no. 9, pp. 1404–1410, Mar. 2019.
5. S. Pontarelli, P. Reviriego, and J. A. Maestro, "Parallel D-pipeline: A cuckoo hashing implementation for increased throughput," IEEE Trans. Comput., vol. 65, no. 1, pp. 326–331, Mar. 2015.
6. P. Reviriego, A. Ullah, and S. Pontarelli, "PR-TCAM: Efficient TCAM emulation on Xilinx FPGAs using partial reconfiguration," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 27, no. 8, pp. 1952– 1956, Mar. 2019.