ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Generating STI Bits for Fault Free Cache Memory Using System Generator

Araveti Pragna
PG Scholar,
Department of ECE,
Global College of Engineering & Technology,
Kadapa, YSR (Dt), AP, India.

S.Farooq Anwar
Assistant Professor & HOD,
Department of ECE,
Global College of Engineering & Technology,
Kadapa, YSR (Dt), AP, India.

Abstract:

With the trend of increasing transient error rate, it is becoming important to prevent transient errors and provide a correction mechanism for hardware circuits, especially for SRAM cache memories. Caches are the largest structures in current microprocessors and, hence, are most vulnerable to the transient errors. Tag bits in cache memories are also exposed to transient errors but a few efforts have been made to reduce their vulnerability. In this paper, we propose to exploit prevalent warcapability of the tag bits in the caches. When data are fetched from the main memory, it is checked if adjacent cache lines have the same tag bits as those of the data fetched. This same tag bit information is stored in the caches as extra bits to be used later. When an error is detected in the tag bits, the same tag bit information is used to recover from the error in the tag bits.

I.INTRODUCTION:

Transient errors (also called soft errors, or single-event upsets) caused by external radiation events have become an important consideration for microprocessor design.Recent research shows that the uncorrected soft errors induce a failure rate higher than all the reliability mechanisms combined [1]. Following the trends of shrinking feature sizes, low supply voltage and high frequency, future microprocessors will become increasingly vulnerable to soft errors.Modern microprocessors typically employ cache memories to bridge the speed gap between the processor and the memory. Cache memories, however, are particularly susceptible to particle strikes since they consume a large fraction of on-chip area. In addition, the leakage control techniques aggressively used today for reducing cache leakage energy make the cache reliability problem even more severe [2]. The accumulated charge from external particle strikes can invert the state of the SRAM cell, which can be easily propagated to the processor or lower-level memory, resulting in erroneous computation or system crash. Consequently, cache memories must be protected against soft errors to ensure dependable computing.A number of approaches exist to improve cache reliability against soft errors, ranging from information redundancy (parity, ECC[3]) to space redundancy (N Modular Redundancy). However, all these techniques come at additional costs in performance, energy, area or design time (which is called reliability cost in this paper). For microprocessors or embedded systems that are increasingly used in reliability-critical applications but with stringent cost constraints, it is a necessity to develop novel cost-effective fault tolerant techniques or to select the most cost-effective mechanism to meet the reliability goal. To achieve this goal, the first step is to understand and measure cache vulnerability to soft errors accurately and quantitatively. While over-estimation can result in excessive protection and thus higher reliability cost; under-estimation can lead to inadequate protection and hence is useless.

II.LITERATURE REVIEW:

Design of high speed computers has always remained a research challenge for computer engineers. High end gigahertz ranged processors have been successfully designed and marketed since last decade. The performance of a computer is necessarily dependent on the relatively slower storages its processor has to work with. Bulk amount of data gets transferred between the processor and the relatively slower hierarchy of memory. This causes delay in the overall response time of the processor. It is known as Von-Neumann bottleneck.Several design steps were taken to eliminate the speed gap between processor and memory. Cache memory was introduced as a high speed intermediate storage between the primary memory and the processor [1]. Cache memories are widely used in von-neumann machines which has simple low speed scalar processors [1],

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

[2] but became unimpressive in harvard computers that have high speed vector processors. Changes in storage technology to design high speed memory devices, increase of memory wordlength and width of databus [3] etc. can be used as alternative ways for increasing throughputs of memory. Again, these design changes had their own limitations due to the predictable saturation of moore's law and chances of inappropriate power dissipation in low power VLSI circuits. The most formidable alternative to these approaches is an interleaved storage system. Here the memory is segmented into equi-banks of storage modules which are connected in an interleaved fashion [4], [5].Detailed study, analysis and verification of structure and speed of interleaved memories have been done in many previous research articles [6] [7] [8] [9]. The speed of interleaved memories depends on how the memory modules are addressed [8].

In this paper, a fault tolerant design of low-order interleaved cache memory is proposed and the structure has been implemented on FPGA. One or more faulty cache lines can easily be bypassed in the reconfigurable platform with reallocation of the address space among the fault free cache lines. With such a line level fault tolerant scheme which is proposed in this paper, only the faulty cache lines are removed from the address space rather than a cache set as a whole. This new method also includes the speed advantage of low-order interleaving and brings in fault tolerance property within the cache memory by insignificantly reducing the addressable cache memory space. The proposed cache memory structure is not only unique but also advantageous over the already existing cache architectures.

III.IMPLEMENTATION:

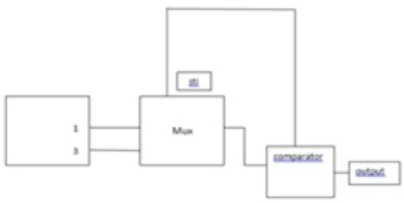
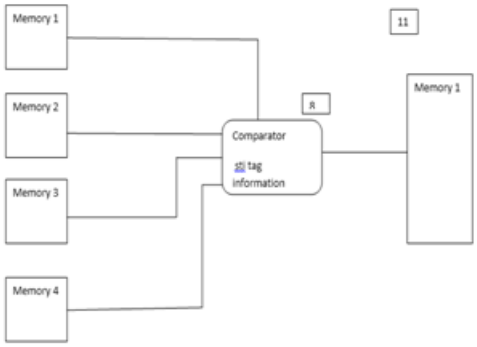


Fig 1 encoder block diagram in implementation

The first two boxes are the memory arrays of different memories given by the mux for comparison and then STI tag bit has been appended.



IV.RESULTS:

cache memory

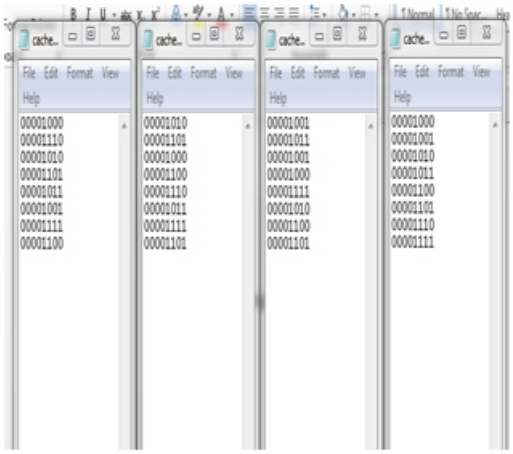
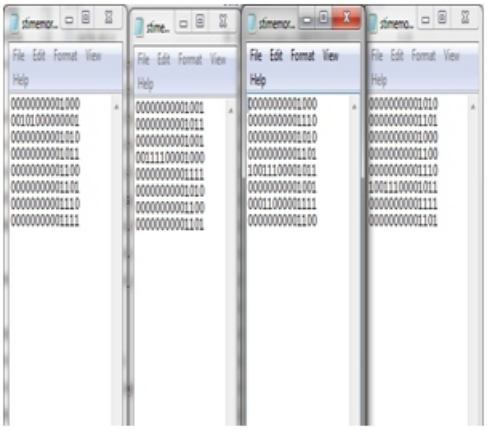


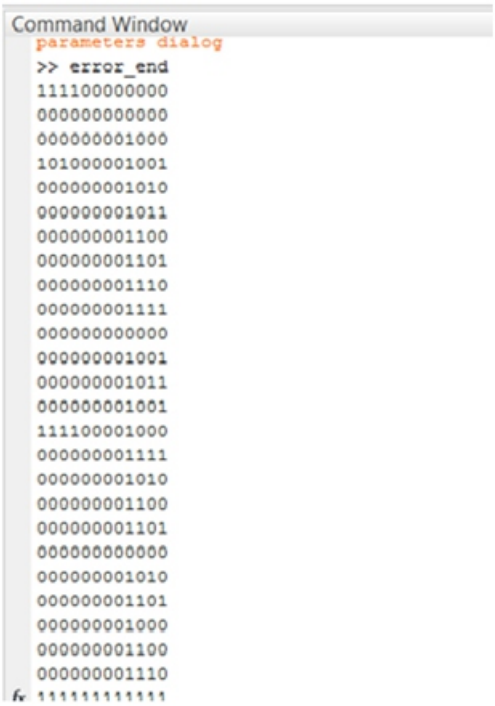
Fig 3 : cache memory data in notepad as input

STI memory



ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Error correction result:



Simulation results
Error corrector

V.CONCLUSION:

With the trend of increasing soft error rate, it is becoming important to provide error detection and correction capability for hardware circuits, especially for cache memories. However, most of the previous techniques focus only on data bits without considering tag bits corruption. Most tag bits in the data caches have their replica in adjacent cache sets from our experiments. We exploit this tag bits similarity against transient errors. Faulty tag bits are simply replaced with correct tag bits from the adjacent cache lines for error correction.

Reference:

[1] Jeongkyu Hong, Jesung Kim, and Soontae Kim," Exploiting Same Tag Bits to Improve the Reliability of the Cache Memories " IEEE Trans. Very Large Scale Integr. (VLSI) Syst,2014.

[2] W. Zhang, S. Gurumurthi, M. Kandemir, and A. Sivasubramaniam,"ICR: In-cache replication for enhancing data cache reliability," in Proc.Int. Conf. Dependable. Syst. Netw., 2003, pp. 291–300.

[3] C. Slayman, "Cache and memory error detection, correction, and reduction techniques for terrestrial servers and workstations," IEEE Trans. Device Math. Rel., vol. 5, no. 3, pp. 397–404, Sep. 2005.

[4] L. Hung, M. Goshima, and S. Sakai, "Mitigating soft errors in highly associative cache with CAM-based tag," in Proc. IEEE Int. Conf. Comput. Des., Oct. 2005, pp. 342–347.

[5] N. Wang and S. Patel, "ReStore: Symptom-based soft error detection in microprocessors," IEEE Trans. Dependable Sec. Comput., vol. 3, no. 3,pp. 188–201, Jul. 2006.

[6] O. Ergin, O. Unsal, X. Vera, and A. Gonzaez, "Exploiting narrow values for soft error tolerance," IEEE Comput. Archit. Lett., vol. 5, no. 2, pp. 1–12, Dec. 2006.

Author's Profile:

Araveti Pragna, Pursuing M.Tech (DECS), Global College of Engineering & Technology, Kadapa, Andhra Pradesh. India.

S.Farooq Anwar, M.Tech , Assistant Professor at Global College of Engineering & Technology, Kadapa, Andhra Pradesh, India.