Indian Journal of Science and Technology, *Vol 8(35), DOI: 10.17485/ijst/2015/v8i35/86798, December 2015*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Implementation of a Novel Data Scrambling based Security Measure in Memories for VLSI Circuits

R. Vijay Sai*, S. Saravanan and V. Anandkumar

School of Computing, SASTRA University, Thanjavur – 613401, Tamil Nadu, India;
vijaysai@it.sastra.edu, saran@core.sastra.edu, anandkumarv1992@gmail.com

Abstract

This article shows the importance of security in memory for VLSI circuits based on data scrambling and overcome attacks. Security information stored in memory is very valuable. The model should not be prone to intruder attacks. The proposed method provides scrambling of information by data scrambling vectors. Instead of using some extra table for scrambling the data in cache memories, the data is divided into two halves and scrambled within to overcome extra hardware and memory requirement. This method is implemented in Verilog HDL using Model Sim which has improvement in area and memory requirement. This method is more suitable for value added applications such as smart cards and bio metric applications.

Keywords: Cache Read and Write Operations, Scrambling Vectors, Security in Memory

1. Introduction

Information is stored in cache and main memory which can be retrieved whenever needed. Intruders have always the possibility of stealing such secret information by their intelligence in hacking. Memory attacks have now become very common to decode the vital data stored. Consumer usage devices such as smart cards and credit cards are in danger of their data being maliciously stolen, asking for protection against such attacks. Specifically, information stored in main memory is always under threat of being intercepted by intruders. Cache and main memory store information, which can be retrieved whenever needed. The information which is sensitive has the threat of being intercepted by malicious readers. Attacks aiming at stealing the secret information stored in memory have been reported to be growing very large and protection against such potent attacks should be invented and employed. Cold boot attacks target to recover these secret data. This paper presents a novel security measure for any memory system: scrambling the stored plain text data with random generated vectors. Detection, correction and evaluation of the proposed concept are widely discussed in the forthcoming discussions.

2. Existing Methods

Shuai Mu, Yandong Deng1 et al.[1] conducted a quantitative analysis on the characteristics of L2 cache by simulating an extensive set of GPGPU applications. The experimental results lead to key observations. The traditional cache scheme can be inefficient upon a massively large number of on-the-fly memory requests because of severe cache line conflicts and disruption of the temporal locality.

Alenka Zajic et al.[2] conducted experiments which were performed on three laptop systems and one desktop system with different processors showing that both active (program deliberately tries to cause emanations at a particular frequency) and passive (emanations at different frequencies happen as a result of system activity) EM side-channel attacks are possible on all the systems. Furthermore, this paper showed that EM information

*Author for correspondence

leakage can reliably be received at distances that vary from tens of centimeters to several meters including the signals that have propagated through cubicle or structural walls. C. Ramya, S. Saravanan[3] proposed a scheme, where a new approach of flipped on-chip comparison is described for security issues. Based on the concept of holding the confidential information within the chip, proposed method is more secure than other countermeasures with less controllability to unknown users. It compares both the input response and the expected response without relying on the cost of the design. Flipped scan chain increases with negligible area overhead and design changes.

F. Paget et al.[4] presents an onsite of fraudulent spotting technology to enhance rate of precision and minimize frauds with lesser expenditure and reduced risk which is the requirement. J. G. Ooi, K. H. Kam et al.[5] implements hardware rather than software design, bringing more safety rather comparatively. The disadvantages of the system are performance reduction and cost. J. A. Halderman et al.[6] describes lucid software transformations holding positive and negative benefits. Being portable, the risk is heavier in laptops. DRAM may be untrusted and sensitive data should be avoided to be captured on such devices.

W. Enck et al.[7] proposes a novel MECU technique resulting in steady state encryption of main memory with no additional expenditure. Non volatility in memory provides advantages in such cases. J. Kong et al.[8] presents a fixed ECC management scheme to change ECC protection strength to the age of PRAM, which is conveniently drawn from the encryption counters.

H. Yuemei et al.[9] showcases cache-based timing attacks and installs a new defending technique against such brutal attacks. Security from side channel attacks can be monitored. Secret information leakage may be vindictive to make system to be of no use. J. Kong et al.[10] stressed that software cache-based side-channel attacks can cause serious threat to the privacy of computer systems. Data cache attacks and instruction cache attacks scan the drawbacks of presently available hardware and software techniques.

Z. Wang, R. B. Lee et al.[11] encompasses that it is ever perilous to put into use cache-based side channel attacks. Shared caches in computing systems are always vulnerable to these attacks. Being software attacks, such attacks are very easy to perform, without the need of special equipments. G. Bertoni et al.[12] imposes a bright method to overcome encryption algorithm implemented in a system with cache memory. The attack is watched and tested against the AES but it can be tested against other block ciphers also.

D. Samyde et al.[13] proves that the hacker can read the data using optical or electromagnetic probing methods. Compared to full invasive attacks, equipment installation and skill is predominantly lower. The necessity will be on hardware countermeasures for providing resistance against such attacks. Wenjing et al.[14] proposes two security techniques which can protect from physical attack in SRAM when the power supply has been switched off. Using HHNEC 0.25um CMOS technology, two SRAMs implementing different security techniques are designed. Successful elimination of data remnants are the outcome of valid test results. When comparing with traditional SRAM, the power consumption of operation is only improved by 4%. Further, write and read operations of two SRAMs are identical to the conventional one.

V. Rozic et al.[15] highlights that the dual-rail precharge principle, which is implemented in secure logic styles can also be applied to design SRAM cells with enhanced side-channel security. Cutting the power supply or cutting the feedback loop can be used without high short circuit elements to apply this principle. One of the drawbacks of the power-cut cell is the requirement of one additional vertically routed wire per column which is activated in every write cycle and operates in full-swing mode, which results in high dynamic power consumption. Because of the reduced number of vertically routed signal wires, shared PMOS structure avoids this issue to some level. Unusable thin-cell layout is the apparent disadvantage o this structure.

K. Soontae et al.[16] made a point that reduced supply voltage, high frequencies and low capacitive values of circuits make them prone to transient errors. Particularly, cache memories are weak yielding soft errors because of their large transistor counts. As a result, modern processors such as Power4, Itanium processors, MIPS R10000 and ARM processors observe error protection techniques for cache memories. Fundamental results stresses that a high degree of protection of the data transferred on the bus can be attained without any power cost. L. Benini et al.[17] submit a set of techniques to improve the security of information transmitted on a communication bus by putting together data scrambling with energy-efficient bus encoding with various trade-offs between the attained degree of security and energy consumption.

3. Proposed Security Measures

The data stored in the cache memory is in encrypted form and so even if any attack occurs, the data retrieved remains unusable for the adversary. In order to maintain the security level, the proposed encryption is designed such that the data is encrypted in write cycle and decrypted during read cycle.

4. Proposed Encryption and Decryption

In the proposed encryption, the data is first divided into two halves. Taking the first bit in the first half as reference, XOR operations for the remaining bits in first half are carried out. This results in encrypted first half of data. The result from the first half is then XORed with the remaining second half to get the encrypted second half data. The encrypted data is stored in the memory in write cycle.

During decryption, the first and second half of encrypted data is XORed to get the decrypted second half data. Taking the first bit in first half as reference and doing XOR operation for the remaining bits in the first half provides the decrypted first half of data.

5. Write Cycle

As shown in Figure 1 during write cycle, CPU generates address and data. The address contains tag, index and word field. This address bit portioning depends on the cache memory size, block size, no. of ways of associativity in a set-associative cache. Data is encrypted and stored in the memory in corresponding address location. The mux/demux is used to select the encryption block during write cycle.

6. Read Cycle

As shown in Figure 2 during read cycle, CPU generates only address. The data in the corresponding address is fetched from the memory and decrypted. The decrypted data is sent to CPU. The mux/demux is used to select the decryption block during write cycle.

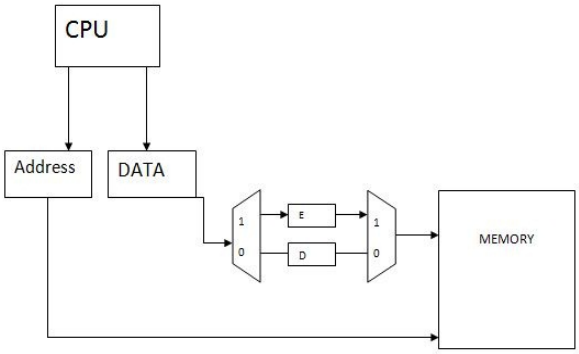


Figure 1. Write cycle.

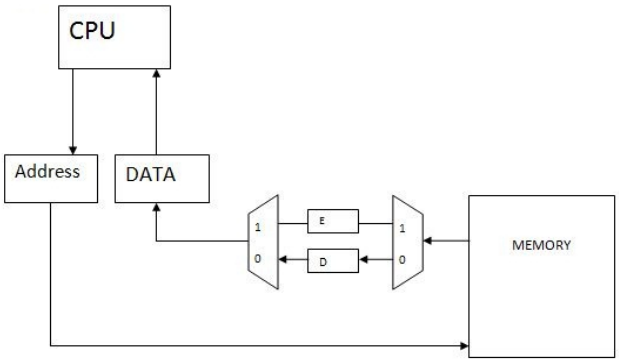
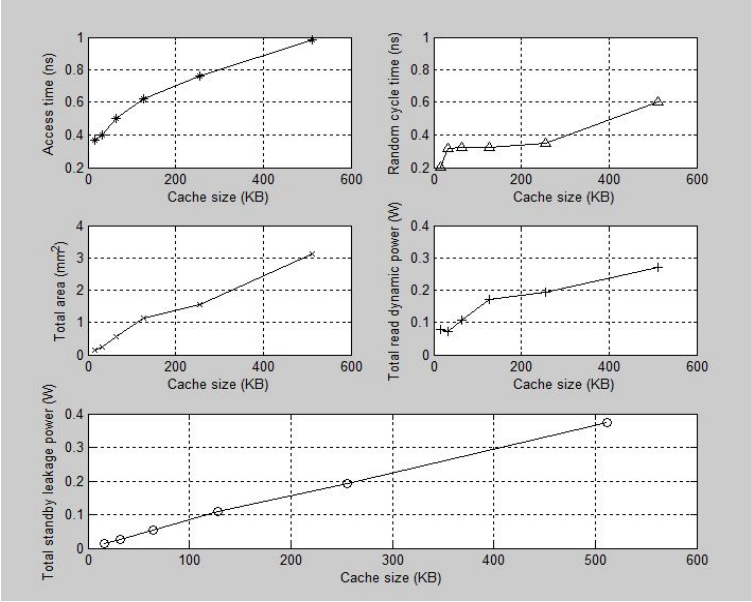


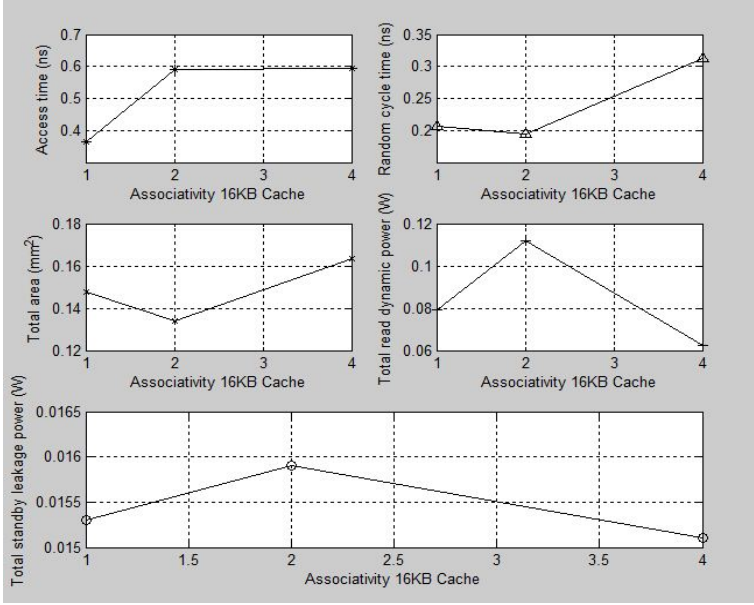
Figure 2. Read cycle.

7. Evaluation of the Proposed Technique

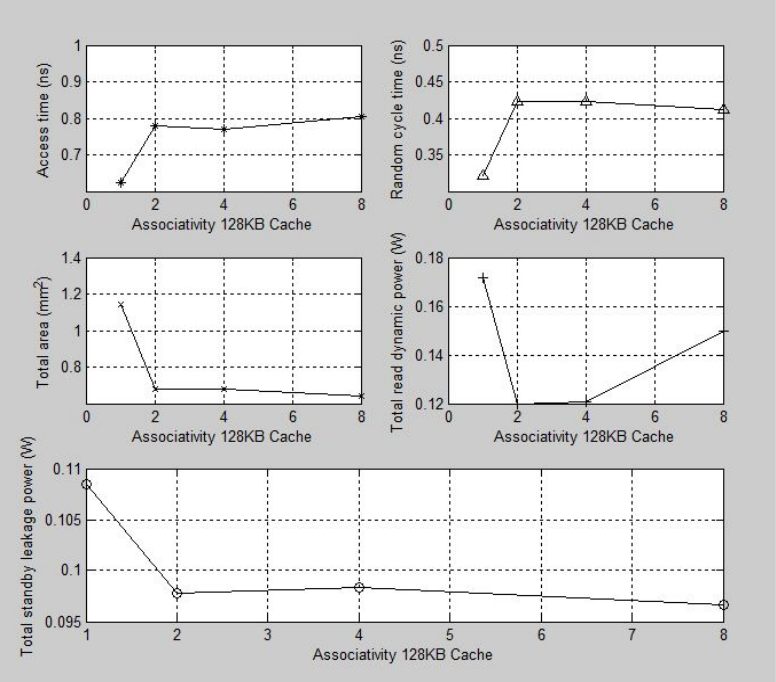
The system taken for evaluation of the proposed technique contains the memory, mux/demux, encryption and decryption block. For evaluation, a 16KB memory with line size of 16 byte is taken. The address field is of 32 bits and data is of 128 bits (four words per line). For modelling of cache memory, cacti tool is used to generate the design parameters like access time, random cycle time, total area, total read dynamic power and total standby leakage power etc. For evaluation we take cache memory size from 16 to 512 KB with a line size of 16 bytes, 1-way set associative with one bank for 45nm technology is shown in Graph 1. We considered the cache memory size of 16 KB, 16 byte line size, one bank and 45nm technology with respect to different ways of associatively like 1-way, 2-way and 4-way as shown in the Graph 2. For evaluation,



Graph 1. 1-way set associative with one bank for 45nm technology.



Graph 2. 1-way, 2-way, 4-way set associative with one bank for 45nm technology.



Graph 3. 1-way, 2-way, 4-way,8-way set associative with one bank for 45nm technology.

cache memory size of 128 KB, 16 byte line size, one bank and 45nm technology with respect to different ways of associatively like 1-way, 2-way, 4-way and 8-way are shown in Graph 3.

8. Conclusion

Low power consumption, low area overhead and small impact over performance are the essential requirements while taking memory security into account. This paper presents a unique security measure which can be applied to any type of memory flexibly. The concept of scrambling is utilized efficiently. Evaluation results positively underlines security in memory, best suited to latest technologies.

9. References

1. Mu S, Deng Y, Chen Y, Li H, Pan J, Zhang W. Orchestrating cache management and memory scheduling for GPGPU applications. IEEE Transaction on Very Large Scale Integration (VLSI) Systems. 2014 Aug; 22(8):1803–14.
2. Zaji A, Prvulovic M. Experimental demonstration of electromagnetic information leakage from modern processor-memory systems. IEEE Transaction on Very Large Scale Integration (VLSI) Systems. 2014 Aug; 56(4):885–93.
3. Ramya C, Saravanan S. Rectifying various scan based attacks on secure IC's. Indian Journal of Science and Technology. 2015 Jul; 8(13):1–6.
4. Paget F. Financial fraud and internet banking: Threats and countermeasures. Report; McAffee Avert Labs. 2009.
5. Ooi JG, Kam KH. A proof of concept on defending cold boot attack. 1st Asia Symposium on Quality Electronic Design (ASQED); Kuala Lumpur. 2009 Jul 15-16. p. 330–5.
6. Halderman JA. Lest we remember: Cold-boot attacks on encryption keys. Communications of the ACM – Security in the Browser. 2009 May; 52:91–8.
7. Enck W, Butler K, Richardson L, McDaniel P, Smith A. Defending against attacks on main memory persistence. Proceedings of the 2008 Annual Computer Security Applications Conference (ACSAC); Anaheim, CA. 2008 Dec 8-12. p. 65–74.
8. Kong J, Zhou H. Improving privacy and lifetime of PCM-based main memory. Proceedings of the 2010 IEEE/IFIP International Conference on Dependable Systems and Networks (DSN); 2010. p. 333–42.
9. Yuemei H, Haibing G, Kai C, Alei L. A new software approach to defend against cache-based timing attacks. Proceedings of the International Conference on Information Engineering and Computer Science (ICIECS); Wuhan. 2009 Dec 19-20. p. 1–4.
10. Kong J, Aciicmez O, Seifert JP, Zhou H. Architecturing against software cache-based side-channel attacks. IEEE Transactions on Computers. 2013 Jul; 62(7):1276–88.

11. Wang Z, Lee RB. New cache designs for thwarting software cache based side channel attacks. Proceedings of the 34th Annual International Symposium on Computer Architecture (ISCA); 2007. p. 494–505.
12. Bertoni G, Zaccaria V, Breveglieri L, Monchiero M, Palermo G. AES power attack based on induced cache miss and countermeasure. Proceedings of the International Conference on Information Technology: Coding and Computing; 2005 Apr 4-6. p. 586–91.
13. Samyde D, Skorobogatov S, Andreson R, Quisquater JJ. On a new way to read data from memory. Proceedings of First IEEE International Security in Storage Workshop; 2002. p. 65–9.
14. Wenjing K, et al. Novel security strategies for SRAM in powered off state to resist physical attack. Proceedings of the 2009 12th International Symposium on Integrated Circuits; Singapore. 2009 Dec 14-16. p. 298–301.
15. Rozic V, et al. Design solutions for securing SRAM cell against power analysis. Proceedings of 2012 IEEE International Symposium on Hardware-Oriented Security and Trust; San Francisco: CA. 2012 Jun 3-4. p. 122–7.
16. Kim S. Reducing area overhead for error-protecting large L2/L3 Caches. IEEE Transactions on Computers. 2009 Mar; 58(3):300–10.
17. Benini L, Galati A, Macii A. Energy-efficient data scrambling on memory-processor interfaces. Proceedings of the 2003 International Symposium on Low Power Electronics and Design (ISLPED); 2003. p. 26–9.