A 90nm CMOS Cryptographic Core with Improved Fault-Tolerance in Presence of Massive Defect Density

Milos Stanisavljevic, Frank Kagan Gürkaynak,
Alexandre Schmid, Yusuf Leblebici
Microelectronic Systems Laboratory LSM, Station 11
Swiss Federal Institute of Technology EPFL
CH – 1015 Lausanne Switzerland

Maria Gabrani
IBM Zurich Research Laboratory
Säumerstrasse 4
CH – 8803 Rüschlikon Switzerland

ABSTRACT

This paper presents the development methodology, circuit realization and measurement of a cryptographic core intended to operate reliably in the presence of massive defect density. A circuit-level voter based on averaging and thresholding has been implemented, and is measured to offer superior reliability in comparison with standard techniques.

Categories and Subject Descriptors

B.8.1 [**Hardware**]: Performance and Reliability – *Reliability, Testing, and Fault-Tolerance.*

General terms

Reliability.

Keywords

Fault-tolerant architecture, high defect density, reliability of submicron and nanoelectronic systems.

1. INTRODUCTION

The ever-growing field of applications, and the pervasiveness of consumer electronic products has largely benefited from the very stable and reliable CMOS fabrication technology. Over the past thirty years, the widespread use of Electronic Design Automation (EDA) software has guaranteed the successful development of a vast variety of novel electronic applications, throughout the many subsequent generations of the scaled fabrication technologies.

Nevertheless, warning signs are prevailing regarding the reliability of future very-deep submicron and nanoelectronic technologies. The dramatic dimensional scaling of the fabrication technologies is expected to impacts on the proper operation of individual transistors, showing up as current leakage, hot electron degradation, and device parameter fluctuations. Moreover, future systems based on nanoelectronic devices are expected to suffer from low reliability due to the constraints imposed by the fabrication technologies, and due to nondeterministic parasitic effects such as background charge, which may disrupt correct operation of single devices both in time and space in a random way. Higher frequencies pose strict limits to timing and therefore also add the probability of timing errors. The increased integration of devices on a single die raises the probability of erroneous components in a die.

The increasing miniaturization of CMOS technology is causing the chip failure rate to increase as both the number of devices and the individual device failure rates increase. It is expected that the probability of failure during manufacturing of nanoscale or molecular scale devices will be orders of magnitude higher than that of present-day transistors. Since it should eventually be possible to put more than 10^{12} molecular-sized devices on a 1cm^2 chip, it is evident that advanced fault-tolerant strategies will have to be devised [1]. Novel strategies enabling the development of fault-tolerant electronics in presence of massive defect density must be prepared in order to allow the advent of embedded systems applied in safety-critical fields such as in-situ medical prosthetic microelectronic circuits or space applications, where the methods applied so far, for example triple redundancy (TMR) with majority voting are going to face their limits. Moreover, the limitation of redundancy factor R to low values (R<5) is a key issue that could not be addressed using earlier methods (NAND-multiplexing).

We propose to address fault-tolerance at several levels of abstraction, also including EDA, and focus in this paper on the presentation of circuit-level developments, using averaging and thresholding in complement of redundancy. A many-core system including several incarnations of a cryptographic core has been developed and measured as a proof-of-concept, including hardware sites for fault emulation. In Section II, we recall to earlier work by authors, where the superiority of the proposed method could be demonstrated based on numerical simulations. The circuit, architecture and integrated circuit development of the fault-tolerant cryptographic core are presented in Section III. Measurement results of the fabricated chip have confirmed the effectiveness of our approach, and are presented in Section IV, also including comparisons with TMR-protected and unprotected cores.

Nano-Net 2007 September 24-26, 2007, Catania, Italy.

DOI 10.4108/ICST.NANONET2007.2142

2. EARLIER WORK ON FAULT-TOLERANCE

2.1 A. Layered Fault-Tolerant Architecture

A four-layer fault-tolerant hardware architecture, named 4LRA (Figure 1) is used in order to offer a solution to the previously presented issues [2]. The architecture described in the following has been applied at the gate, or extended gate level. It can be applied hierarchically in a bottom-up way, and combined with other high-level fault absorption techniques. It consists of four layers that process data in a feed-forward manner, yet the operation is quite different from the classical majority-based redundancy. The details of this architecture have already been presented in [2], [3].

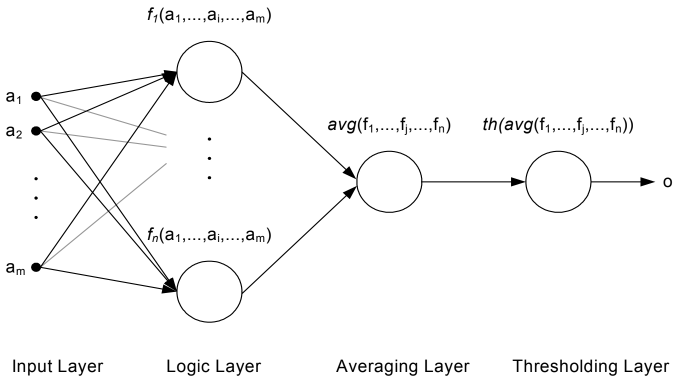


Figure 1. Four-layer fault-tolerant architecture.

The averaging/thresholding circuit used in layers three/four exhibits analog behavior. Adaptable thresholding is necessary to adapt the 4LRA to the actual faulty transfer function surface as illustrated in Figures 8 and 9 in [2]. Static errors can be recovered with the proposed architecture. This work has extended to the study of delay faults which can also be recovered using the proposed 4LRA.

2.2 Electronic Design Automation Design Process

A complete tool for a-priori functional fault tolerance analysis was developed by authors to support the development of fault-tolerant libraries of standard cells. It is a statistical Monte Carlo based tool that induces different failure models, and does subsequent evaluation of system reliability under realistic constraints. The system reliability analysis tool, together with the structured fault modeling architecture, represent a considerable improvement of existing IC design methodologies for fault tolerance.

Moreover, the analysis of reliability of different circuits has been undertaken starting from the simple NOR Boolean gate as depicted in Figure 2, where Monte Carlo (SPICE) analysis have been applied using the developed tool [3]. Here, the use of the 4LRA enables correct circuit operation over a plateau reaching out to 25% of probability of failure per gate, and gracefully degrades, where TMR shows a clearly weaker resistance to transistor fault. Only averaging without adaptable threshold (AVG) also shows better performance then TMR.

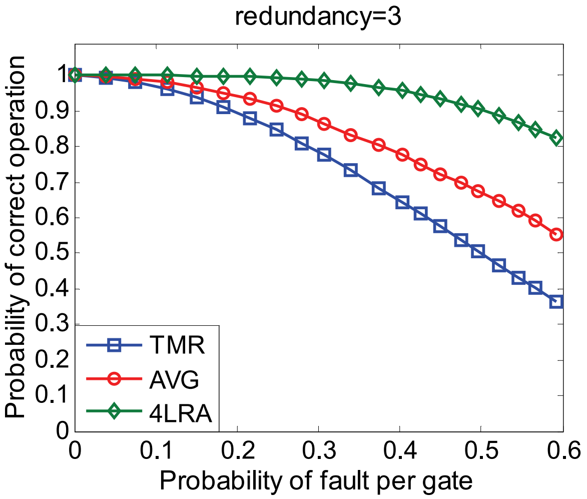


Figure 2. Comparative analysis of the 2-input NOR gate in case of four-layer architecture with adaptable threshold (4LRA) and without it, i.e. fixed threshold (AVG), and triple modular redundancy (TMR) with a redundancy factor of 3.

3. RELIABLE ARCHITECTURE IMPLEMENTATION OF A 128-BIT AES PROCESSOR CORE

A standard cell design of moderate complexity using a 90nm technology [4] has been implemented in order to compare the efficiency of the reliable design methodology. The implemented system consists of an implementation of the well-known Advanced Encryption Standard (AES) cryptographic algorithm using 128-bit keys [5]. This choice of design makes fault masking very improbable, which is important for testing of reliable architectures.

In order to evaluate different fault-tolerant architectures, the design has been partitioned as shown in Figure 3. The implementation consists of four encryption (encrypt) module cores, one key generation (keygen) module, input/output registers, and one additional module that has been integrated for the purpose of controlling induced faults. To minimize the I/O requirements external I/O interfaces have been limited to 8 bits. Separate input and output controllers handle the communication and store 128-bit copies of the plaintext, cipherkey, and the ciphertext. This pipelining allows simultaneous encryption, and data I/O. A parallel on-the-fly key generator is used to derive keys used in AES from the cipherkey, in every round, iteratively.

The full development process has been inserted into an established industrial design flow, involving a number of adaptation scripts to be developed. In order to limit the number of cells to be added into the standard cell libraries, a generic averaging and thresholding circuit (ATC) with the ability to accommodate three redundant units has been designed. Several different drive strength versions of the ATC cell are designed and characterized. Consequently, the ATC can be treated by place and route tools as a regular cell. The design is synthesized in a standard fashion, and the resulting netlist is modified using the set of custom scripts.

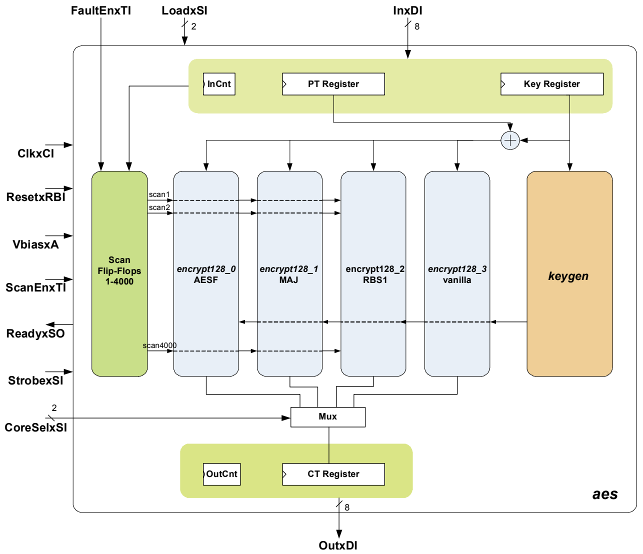


Figure 3. System architecture of the four implemented crypto-cores.

The first step in the development takes care of selecting circuit portions of appropriate size, and instantiating them according to the desired redundancy factor. Then, adding reliability feature is performed by taking the output from the synthesizer or a preplaced netlist, and generating the final netlist for place and route tools. Redundant units (layer two in 4LRA) are placed separately as well as the ATC.

After this, a step of fault insertion is performed. Fault emulation has been introduced to provide full controllability over the faults to be injected into the AES core. The goal of the design is to demonstrate the performance of fault-tolerant architectures. The 90nm design process that has been used is a commercial fabrication process, and does not have the high/massive defect density required to justify the fault-tolerant design methodology. Therefore, artificial faults are inserted into the system at 4000 spatially uniform locations. 4000 has been chosen in order to be able to simulate a wide variety of fault patterns concerning that the basic circuit of the encryption core contains more than 5000 gates. Two types of faults are supported. In a hard-0 fault, the node is driven to logic-0 by the addition of a AND gate whose second input is used to control the insertion of a fault. A hard-1 fault can be inserted by adding an OR gate in a similar manner. To control all faults, a shift register with 4000 locations is used. This organization allows any combination of 4000 faults to be active at any given time. After this, a standard back-end design flow is used. However, the optimization rules of the back-end design are modified to prevent logic optimization and critical path resynthesis. Another modification is ensuring that the analog bias voltage is routed as an analog signal without buffers.

The ATC is realized as a standard library cell. A differential to single-ended version of the ATC has been selected as a way to keep static current below the 20μA limit that has been derived as the maximal current dissipation per ATC from the permitted overall static current dissipation, and increase linearity figures. The schematic and layout of the implemented circuit are depicted in Figure 4. The unbalanced aspect ratio of the standard cell is due to the need to respect the regular height of standard cells, as well as the need to decrease the current density through the averager. The cell was characterized and included in the standard design flow. The analog biasing voltage, V_{bias} is set to 300mV causing a static current dissipation of 18μA. The gate delay is equal to 280ps with a load of 6fF. Dynamic current dissipation is equal to 8μA. Relative differential and integral linearity are both under 10% for the whole input range.

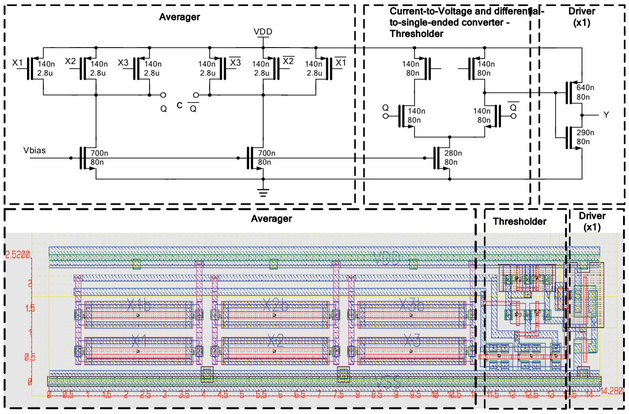


Figure 4. Schematic and layout of the ATC cell that has been designed and inserted in the library of standard cells.

The manufactured chip supports four separate incarnations of the encryption core. The first core is a straightforward implementation of the encryption (codenamed: Vanilla). It has been optimized to run at 250MHz and is able to reach a throughput of 2.9 Gb/s. However, the 8-bit I/O interface limits the throughput to 2.0 Gb/s. The second core uses the well-known majority voter approach (MAJ) with each gate triplicated and added a simple two-out-of-three majority voter. The third core implements the proposed multi-level fault-tolerant architecture (RBS1). In this version a simplified model is used, where the logic layer consists of a single logic gate. Due to the ATC cell implementation, thresholding is adjustable, but not adaptable, therefore this realization is similar to the one named AVG in Figure 2. The fourth core is once again a standard implementation (AESF). However, it also includes the artificial emulated fault locations. Note that, for each gate in the original netlist there are three gates in the redundant architecture. Therefore in some cases multiple faults in the redundant architecture map to a single fault in this architecture. In total, redundant architectures have 4000 fault locations, while this core has only 2668.

The critical component in this design is the ATC cell. Our first version of the ATC was designed as a proof of principle and was designed as a standard cell that is compatible to the rest of the standard cell library and was not optimized for speed and area. As a result there are noticeable delay penalties for the core that uses the ATC cell. The critical path delay is approximately 4ns for cores Vanilla and AESF, 6ns for MAJ, and over 15ns for RBS1 due to a slow ATC cell implementation that is common in the path. The maximum operating frequency is 66MHz.

The test chip has 27 I/O pins and occupies 1.5x1.5mm total area out of which 0.7mm2 is core area. The total number of gate-equivalents is 67000 gates. No restriction has been assigned to the exact position of the four cores during placement and routing, which spread over the full area. The ATC cell has also a significant impact on size which makes the RBS1 core approximately twice larger then the MAJ core. Extra hardware logic included for fault emulation (AND/OR gates) and additional wiring occupy 1.7% of the total useful area in each core.

Gate count and area statistics are summarized in Table 1.

Table 1. Gate Count and Area Statistics

Modul	N. of non-replicated gates	Num. of replicated gates	Area of replicated gates (um^2)	Area for ATC/MAJ (um^2)	Area for fault scan g. (um^2) (N. of gates)	**Total gates**	**Total area**(um^2)
Vanilla	5073	-	-	-	-	5073	**27095**
RBS1	1539	10602	67259	127173	14113 (4000)	19675	**213991**
MAJ	1539	10602	67259	32417	14113 (4000)	19675	**119235**
AESF	5073	-	-	-	9413 (2668)	9073	**36508**
Fault scan						4000	**70559**
Top+keyg.						9237	**66781**
Total						**66733**	**534160**

The analysis of the maximal fault densities for each integrated core is given in Table 2.

Table 2. Overall Fault Rates

Module	Faults/μm^2	Faults/total gates
RBS1	0.019	0.2
MAJ	0.034	0.2
AESF	0.073	0.29

4. MEASUREMENT RESULTS AND DISCUSSION

After the chip has been fabricated, a suitable packaging has been chosen and testing has been performed. Beside functional tests, electrical and reliability tests were carried out. The overall static power dissipation is under 70mA. For the best fault tolerance, the optimal analog bias voltage V_{bias} is determined to be equal to 370mV. The maximum measured operation frequency for different cores is 100MHz for AESF, 60MHz for MAJ and 35MHz for RBS1 core.

The reliability testing is performed using a sampling technique. Sample of 1000 patterns with equal number of faults per pattern, but different uniform distribution, has been generated for every value of fault density, until the overall probability of correct operation of the chip was not negligible. The sample of 1000 patterns is sufficient for the sampling error smaller then 1.5%. Measurement results showing the probability of correct operation of the whole chip versus the probability of fault per gate, for every core, are depicted in Figure 5.

The core without reliability architecture (AESF) has zero fault tolerance, due to the absence of fault-masking in the crypto design. Despite the fact, that induced faults are only hard-0 and hard-1 type which makes the result of the averaging operation equal to majority voter, a clear advantage of RBS1 over MAJ core can be observed, as an improvement of about 10% especially for higher fault rates of 0.015 and larger. The cases when both ATC and MAJ circuits can extract a correct value are one-input stuck-at 0 or stuck-at 1; one-input stuck-at 0 and another input stuck-at 1. The case where both inputs are stuck-at 1 can also be corrected by ATC circuit in certain number of occurrences, thanks to the adjustable threshold; this is the reason of the observed improvement between RBS1 and MAJ core. A tradeoff in this case is between reliability and speed. In order to have a real advantage of RBS1 core demonstrated (as shown in the Figure 2), it is necessary to have analog faults interfering, which means that the outputs of the cells in logic layer in Figure 1 can be at arbitrary levels between ground and supply in the presence of faults. This is a situation which is expected to happen in case of realistic defects.

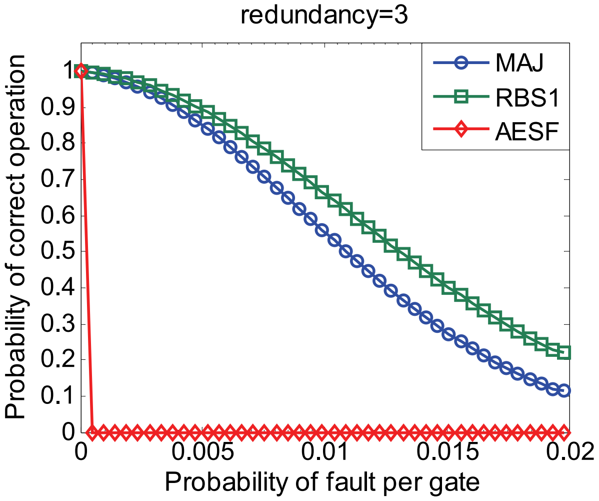
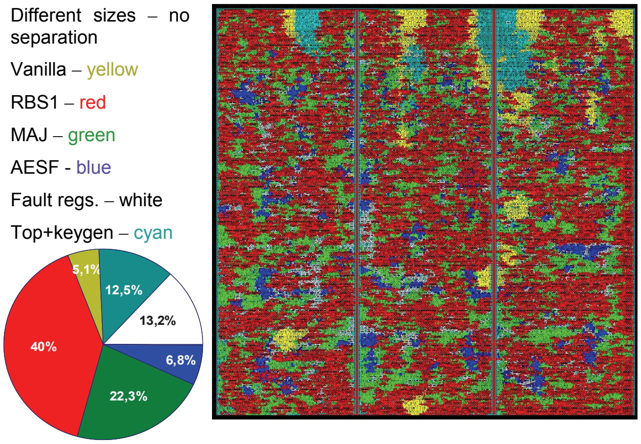


Figure 5. Measured reliability figures.



(a) (b)

Figure 6. (a) Final cores layout figures, and (b) layout of the integrated circuit.

The methodology and supporting software tool presented in this article have been applied in the development of a common cryptographic 128-bit AES core, demonstrating the practical feasibility of the approach, and its possible integration within an established industrial design-flow.

In our approach, the amount of engineering work devoted to the adaptation of a regular design into a design with high level of fault-tolerance is very limited. Typically, the designer is expected to provide the tool information related to the expected defect density, in addition to standard design database. The software system will extract the related redundancy factor, and accordingly adapt the netlists by clustering the design and inserting ATCs.

The cost to be paid for a fault-tolerant design remains in terms of extra hardware and consequently, additional delay and power dissipation. These issues should primarily be solved at circuit level. Specifically, the ATC which needs to be massively replicated will be redesigned, to increase switching speed, while improving the power dissipation figures. The area constraint remains from our perspective the least stringent constraint, considering redundancy factors used in this method as low. Moreover, from an application oriented perspective, savings in engineering workload are efficiently re-invested into increased silicon area, which is very acceptable in safety-critical developments.

5. ACKNOWLEDGMENTS

This work has been conducted with the support of the Swiss SNF grant 200021-112291, the EPFL Centre SI project NANO-PLA, and the CCMX-MMNS project.

6. REFERENCES

[1] K. Nikolic, A. Sadek, M. Forshaw, Fault-Tolerant Techniques for Nanocomputers, *Nanotechnology*, 13, 2002, 357-362.

[2] Schmid and Y. Leblebici, Robust Circuit and System Design Methodologies for Nanometer-Scale Devices and Single-Electron Transistor, *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 12, No. 11, Nov. 2004, 1156-1166.

[3] M. Stanisavljevic, A. Schmid, and Y. Leblebici, A Methodology for Reliability Enhancement of Nanometer-Scale Digital Systems Based on A-Priori Functional Fault-Tolerance Analysis, *Proc. IFIP VLSI-SoC*, October 2005.

[4] Publicly available parameters for the IBM 90nm technology http://www-03.ibm.com/chips/asics/products/stdcell.html

[5] Advanced Encryption Standard, Federal Information Processing Standards 197 (FIPS 197), National Institute of Standards and Technology (NIST), November 2001.