CIRCUITS AND DESIGN TECHNIQUES FOR SYSTEM-ON-A-CHIP INFORMATION SECURITY

HÉCTOR IVÁN GÓMEZ ORTIZ

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS
ESCUELA DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES
BUCARAMANGA
2019

CIRCUITS AND DESIGN TECHNIQUES FOR SYSTEM-ON-A-CHIP INFORMATION SECURITY

HÉCTOR IVÁN GÓMEZ ORTIZ

Trabajo de grado presentado para optar al título de Doctor en Ingeniería, área Ingeniería Electrónica

Director:
ÉLKIM FELIPE ROA FUENTES
INGENIERO ELECTRÓNICO. PhD.
Co-director:
ÓSCAR MAURICIO REYES TORRES
INGENIERO ELECTRÓNICO. PhD.

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FISICOMECÁNICAS
ESCUELA DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES
BUCARAMANGA
2019

ACKNOWLEDGMENTS

There are plenty of people I feel indebted to during this extraordinary and long journey – my sincere gratitude to all them.

I am grateful to Professor Elkim Roa for the invaluable opportunity of being part of this excellent team OnChip, and for always demanding the best of me, which allowed me to develop new skills, especially when it comes to a foreign language.

I am also thankful to Professor Óscar Reyes who gave me the opportunity at first instance to initiate my doctoral program.

I am indebted to Professor Robert Mullins for his invaluable support during the amazing research stay at the University of Cambridge, which at the same time was one of the most challenging moments of my life for being far away from my family. Professor Mullins introduced me to some great researchers, and he always, and he always supported me during my stay.

A very special gratitude goes out to the entire at the research lab, especially to Andres Amaya and Javier Ardila for the technical discussions and proofreading assistance. My appreciation goes to Ckristian Duran for sharing and helping me with all the programming tasks. Julian Arenas for his hard work during the final chip measurement stage of my research. David Reyes for helping me with layout tasks. Rolando Torres, Camilo Rojas, Juan Moya, Luis Rueda, Wilmer Ramirez for the enlightening coffee breaks.

I would like to thank Maria Helena, Alberto and Maria Fernanda for being again like a family during the last year. I also want to thank Oliberth Reyes for his entrepreneurialism and enthusiasm that always motivated me to see a silver lining. I am grateful to my father and mother in law for being like my second parents.

My gratitude also goes to the professors who served in my committee. Professor Alfonso Chacon for reviewing in detail the whole document and assisting to my defense. Professor Alfredo Arnaud that took the time to review my dissertation even with the tight schedule. Professor Jose Amaya, Luis Nuñez and Henry Lamos for their feedback.

Last but not least, I would like to thank my closest and most important family members. My wife Jennifer who spent uncountable times by my side in the long nights of work and always set the limit and put me to bed when she knew it was time, as well as for her unconditional support

and encouragement every time. I am eternally grateful to my daughter Valeria for comfortable playing times and for making me see life differently. I think I am more than indebted with my mother Arleth for all her support and philosophical discussions. I am grateful to my father Hector for the distractive talks about cuisine. Also, I wish to thank my brother Diego and sister Mabel for sharing with me their life experiences and seeing me as a living example.

I am grateful to Colciencias for awarding me a scholarship during the first years of my doctoral program. I am also thankful to the Newton Fund Industry-Academia Partnership Programme for funding my research stay.

This work is dedicated to my daughter Valeria who literally took me to run several times when I was stuck writing this dissertation.

CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

RESUMEN

TÍTULO: CIRCUITOS Y TÉCNICAS DE DISEÑO PARA SEGURIDAD DE LA INFORMACIÓN EN UN SISTEMA INTEGRADO *

AUTOR: HECTOR IVAN GOMEZ ORTIZ †

PALABRAS CLAVE: Seguridad, sistema integrado, hardware, números aleatorios, establecimiento de llave, martilleo de filas.

La tendencia al aumento de dispositivos electrónicos interconectados crea varios desafíos técnicos. Uno de ellos es garantizar la privacidad de la información, considerando como la información podría extenderse a través de diferentes canales antes de llegar a su destino final. La garantía de privacidad implica diferentes primitivas de seguridad de hardware / software, como la generación y el establecimiento de claves. Estas dos primitivas desempeñan un papel fundamental, ya que las operaciones en cualquier algoritmo criptográfico se basan en claves de alta calidad y en el establecimiento de una clave de sesión, o tener una clave secreta común. La protección de la propiedad intelectual es otra preocupación de la industria actual; el acceso físico a los dispositivos permite un escenario ideal para desarrollar ingeniería inversa. La ingeniería inversa podría conducir a la extracción sin la autorización adecuada de funcionalidades o datos confidenciales que podrían usarse para explotar vulnerabilidades y obtener acceso restringido, o para realizar ataques sofisticados.

Este trabajo presenta tres contribuciones probadas en el área de seguridad de la información a nivel de circuito abordando los desafíos mencionados anteriormente. La primera contribución comprueba la implementación de un establecimiento de clave ligero con una función de cambio de clave rápido, donde se proponen dos implementaciones a nivel de circuito para acelerar la función de cambio de clave y proporcionar ofuscación. La segunda contribución se enfoca

*Trabajo de Investigación.

†Facultad de Ingenierías Fisicomecánicas. Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones. Director: Élkim Felipe Roa Fuentes. Co-director: Óscar Mauricio Reyes Torres.

en generadores de verdaderos números aleatorios (TRNG) totalmente sintetizables para la generación de claves de costo bajo. Presentamos dos nuevas arquitecturas para TRNG totalmente sintetizadas junto con un método para captar la entropía, utilizando dos fuentes de entropía, considerando que un avance reciente prueba que la extracción de números verdaderamente aleatorios requiere más de una fuente de entropía. La tercera contribución propone una técnica para evitar la ingeniería inversa mediante el uso de la ofuscación a nivel de layout.

Finalmente, más allá del alcance del trabajo propuesto, también abordamos un problema de seguridad en las memorias DRAM. Algunos chips DRAM modernos experimentan el llamado error de martilleo de filas que permite infracciones de seguridad. Para contrarrestar este ataque desarrollamos una estrategia novedosa a nivel de hardware para mitigar los ataques de martilleo de filas basados en una celda ficticia. La estructura propuesta ofrece un mecanismo de alerta para activar las operaciones de actualización del controlador de memoria, evitando el cambio de bits o la pérdida de información, mientras se realiza un ataque de martilleo de filas.

ABSTRACT

TITLE: CIRCUITS AND DESIGN TECHNIQUES FOR SYSTEM-ON-A-CHIP INFORMATION SECURITY [‡]

AUTHOR: HECTOR IVAN GOMEZ ORTIZ [§]

KEYWORDS: Security, system-on-a-chip, hardware, random numbers, key establishment, row-hammering.

The up-trending increase of interconnected electronic devices brings into existence several technical challenges. One of them is to ensure information privacy, considering how information might spread throughout different channels before reaching its final destination. Privacy assurance implies different security hardware/software primitives such as key generation and establishment. These two primitives play a critical role since operations on any cryptographic algorithm are based on high-quality keys and the establishment of a session key, or have a common secret key. Intellectual property protection is another concern of today's industry; physical access to devices enables an ideal scenario to perform reverse engineering. Reverse engineering could lead to unauthorized extraction of functionality or sensitive data that might be used to exploit vulnerabilities and get restricted access, or to perform elaborated sophisticated attacks.

This work introduces three proven contributions in the area of information security at the circuit level addressing the above mentioned challenges. The first contribution validates the implementation of a light-weight key establishment core with a fast rekeying feature, where two circuit level implementations are proposed to accelerate a rekeying feature and to provide obfuscation. The second contribution focuses on offering fully-synthesizable true random number generators (TRNG) for low-cost key generation. Two new fully synthesized TRNG architectures are introduced along with a method to harvest entropy, using two entropy sources -considering that a

[‡] Research Work.

[§] Facultad de Ingenierías Fisicomecánicas. Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones. Advisor: Élkim Felipe Roa Fuentes. Co-advisor: Óscar Mauricio Reyes Torres.

recent breakthrough proves that extraction of truly random numbers requires more than one entropy source-. The third contribution proposes a technique to prevent reverse engineering using layout design obfuscation.

Finally, beyond the scope of the proposed work, we also address a security issue in DRAM memories. Some modern DRAM chips experiment the so-called row-hammering bug which enables security breaches. As a countermeasure, we develop a novel hardware strategy to mitigate row hammering attacks based on a dummy cell. The proposed structure offers an alert mechanism to trigger refresh operations, avoiding bit-flipping or information loss, while a row hammering attack is performed.

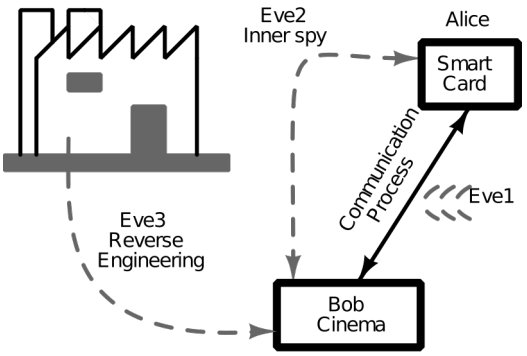
INTRODUCTION

The idea of interconnecting devices was foreshadowed decades ago, foreseeing different human-controlled machines connected to the internet: from radio frequency identification (RFID) technology to today's wireless devices featuring different sensing capabilities. The interconnection of these devices is called the internet of things (IoT), and it offers many applications from smart home appliances, to wearable transportation and health-care gadgets. The number of interconnected devices capturing and sharing data without human intervention is constantly increasing, with projections of hundreds of billions of connected devices.

Yet, within the IoT context, a significant challenge remains: how to ensure that the information that billions of devices capture, process and transmit is secured? Designing low-cost IoT systems that are robust against information leakage requires to take into account that many interconnected devices operate with limited energy, memory, and processing capabilities. Besides, IoT security solutions must comprise integration at hardware, software and application levels, because of the heterogeneous nature of devices and networks. This integration should take into account that security is not a priority in many products, and in such a scenario, many vulnerabilities might appear. Also, time to market pressure reduces development and verification time; therefore, some products are not developed with strong security considerations.

Needless to say, IoT privacy and security are hot research topics, aiming at preventing personal and business information leakage from interconnected devices [1], [2], [3]. Lack of security may not only result in stolen information but poses serious risks to critical systems if, for instance, devices for health-care and transportation are induced to fail or are susceptible to tampering. A security system implemented in a cinema, for instance, could face three attack scenarios as Fig. 1 shows. First scenario—Eve1— happens when an attacker hears or manipulates the communication process to extract sensitive information. Second scenario—Eve2— relates to the smart card provider or programmer. In this case, an attacker could try to find security breaches by manipulating the reader and transponder [4]. Third scenario—Eve3— may be linked to a possible system attack during the third-party fabrication process. An attacker in a semiconductor foundry can analyze the physical implementation trying to reproduce the hardware employing reverse engineering and implant backdoors.

Figure 1. Security system under different attack scenarios.



Encryption, on the other hand, caters a mechanism to achieve privacy targets [5], providing hardware/software security primitives such as key generation and establishment. These two primitives play a critical role since any cryptographic algorithm bases its operations on high-quality keys and the establishment of a session key, or a common secret key.

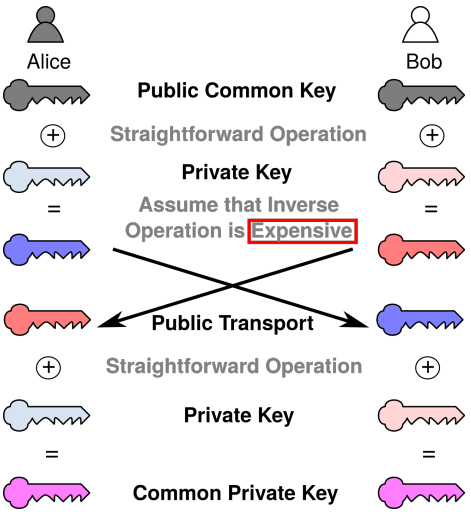
TRNGS: TRUE RANDOM NUMBER GENERATORS

Development of hardware for cryptography is a challenge considering the need for energy and area efficient circuits. Cryptographic protocols require many security primitives where key generation and establishment are some of the most critical objectives. A security system requires high-quality and unpredictable keys where their generation relies on true random number generators TRNGs. Therefore, a TRNG is essential security primitive that must have a consistent statistical behavior as close as possible to uniform distribution to fulfill high-quality standards.

A TRNG uses physical sources as device noise to extract entropy. Physical sources exhibit non-uniform or biased statistical behavior, which represents a challenge in generating high-quality keys. A classical method to alleviate the associated bias in entropy sources is to use intensive computational post-processing, at the expense of reduced throughput. However, restrictive applications require more straightforward solutions which have motivated designers to propose fully-digital approaches.

Digital circuit implementation is commercially attractive, considering its inherent robustness to process and environmental variations providing high yield low-cost circuits. Although digital TRNGs have been reported [6, 7], there is still a need to apply computing intensive calibration algorithms [7–9], or the limited application of output bits. Based upon reported results so far,

Figure 2. Classical public scheme based key establishment.



the design of TRNGs is still an open field which can be explored to bring improved methods and architectures for robust key generation.

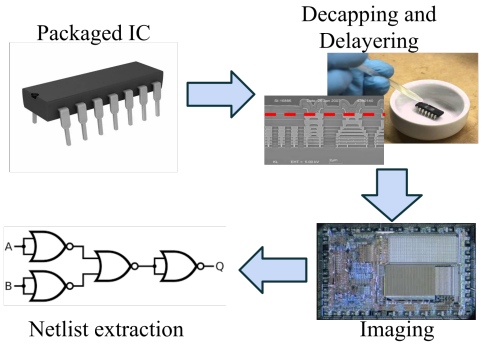
LIGHT-WEIGHT SYMMETRIC KEY EXCHANGE

Once a security system generates a key, a secure sharing system delivers it to the entities involved in the transactions. This sharing or establishment is also critical in a symmetrical system; if an attacker can discover the key, the security system is broken. Hence, a robust and efficient establishment system must deliver the key.

Traditional crypto-systems use public key cryptography based on elliptic and hyperelliptic curves to establish a session key [10]. Fig. 2 is a classical key establishment using a public key scheme where two parties in a communication process establish a common secret key without revealing the latter. This kind of crypto-systems are robust, but require computational power not available in resource-restricted systems. Therefore, a fixed key based system or symmetric algorithms are preferred in order to deal with the low availability of resources.

Nonetheless, fixed key based systems are too rigid, and once the key is exposed, the whole security system is broken. Therefore, there exists the need for flexible cryptosystems that use light-weight algorithms. One alternative is to use symmetric cryptography, which is less complex than public cryptography, providing low-cost security systems. However, the trade-off between complexity and security is still open to discussion, requiring efficient hardware implementations

Figure 3. Revese engineering process using decapping and image processing tools.



with optimized classical algorithms or using new light-weight algorithm-based encryption.

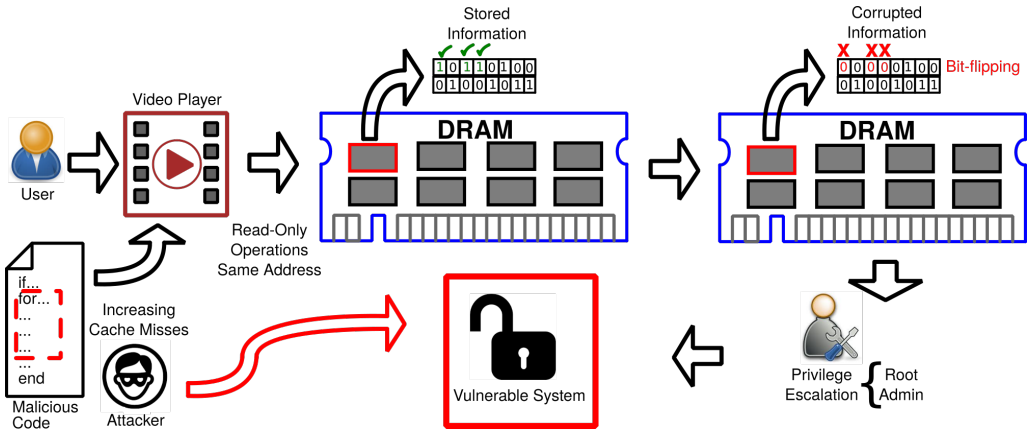
HARDWARE OBFUSCATION

Strong security primitives are not the only objective that must be fulfilled. Obfuscation is another requirement due to reverse engineering. With the advance of new technologies, reverse engineering has become feasible and profitable. A securely stored key, a TRNG IP, or possible flaws in the system are targets for reverse engineering.

Intellectual property (IP) protection is another concern of today's industry, due to the increased popularity of reverse engineering. An attacker who aims at performing reverse engineering, makes use of high-resolution imaging, micro-probing, and side-channel examination [11], as Fig. 3 shows, to extract functionality or sensitive data that can be used to exploit vulnerabilities and get restricted access or develop sophisticated attacks. The mentioned processes prove how difficult it is for a designer to protect an IP when an attacker has physical access to the chip.

Hardware obfuscation [12] is an alternative solution which relies on two main approaches: structural obfuscation and physical obfuscation. The purpose of the structural approach is to hide the actual functionality of the device-under-attack through techniques such as the insertion of key-gates [13]. Traditional methods to physical obfuscation exploit camouflaging by designing look-alike gates with metal and contact dummies [14]. However, using dummy structures implies area and timing overheads that might impede the adoption of camouflaging in high-performance applications.

Figure 4. Security vulnerabilities due to hardware flaws.



HARDWARE VULNERABILITIES

Market demands and industry competition impact security on consumer electronic devices. A shrinking time to market pushes hardware design to the point where verification time is reduced, resulting in hardware vulnerabilities that are not detected in pre-marketing phases. For instance, DRAM memories go through time-consuming verification processes under complex fault models that are continuously updated due to technology shrinking undesired effects. However, the question remains whether it is feasible to model all possible scenarios, including those that do not correspond to regular operation, and how much verification would render a positive cost-benefit [15, 16].

A recent DRAM bug proves that unexplored verification scenarios can produce failures. When a specific row or address is repeatedly opened (or activated), coupling noise increases leakage current among adjacent addresses or memory rows. In such condition, the information stored in capacitance cells leaks faster than refresh operation intervals, resulting in information loss or privilege escalation [16], as Fig. 4 depicts. These flaws appear due to the lack of a detailed verification and the strong interaction between memory cells that integration density brings on. Current verification procedures cannot detect such a failure since the cause does not correspond to a regular operation case.

The phenomenon of increased leakage current in adjacent rows cells is called row hammering. Many recent works report this bug in modern DRAM chips. Also, some researches prove this bug not only allows for intentional information corruption but may also provide privilege esca-

lation or unrestricted access, which represent severe security issues [17]. This security breach has driven the development of different hardware and software countermeasures [15, 16, 18, 19]. However, current solutions face a lack of compatibility or unacceptable overheads, and notwithstanding such countermeasures DRAM chips area still vulnerable to the bug. Therefore, finding feasible solutions remains an ongoing challenge that requires further work and research.

CONTRIBUTIONS

This dissertation proposes circuits and design techniques that offer low-performance overhead fulfilling new security requirements. For instance, a robust key establishment and sharing generally imply public key cryptography which requires power and area hungry implementations. In contrast, This dissertation proposes a light-weight symmetric key establishment core based on tree parity machines with a fast rekeying feature [20, 21]. A circuit implementation for a rekeying feature has not been discussed in detail. Hence, a PRBS with variable length and a masked output is here proposed to perform the rekeying operation. This PRBS is later reused to generate initial weights and to create pseudo-random inputs in the synchronization process, reducing the implementation cost. A post-processed seed using the variable length PRBS adds obfuscation to the rekeying, increasing the hardening of the system. Besides, a figure-of-merit (FoM) is proposed, based on a cyclic redundancy check (CRC) implementation in order to estimate TPM implementation cost ratio, showing the scalability of the circuital architecture. A CRC implementation is appropriate as a measurement reference for size and power specifications since common low-cost systems usually use well-known CRC algorithms to detect errors.

Key generation is another critical objective in security systems due to the need for high-quality and near impossible to predict numbers. This generation relies on TRNGs, and their vulnerability to process and environmental variations makes TRNG design a challenge, even using fully-digital approaches. Therefore, high-quality keys implementation generally require high-complexity TRNG circuits with post-processing and calibration stages. To address the mentioned issues with low-complexity circuits, this dissertation proposes two fully-digital architectures and a methodology to boost entropy. First, a fully-synthesized TRNG design with a fast post-processing stage is offered [22, 23]. A pure-combinational parallel cellular automaton based post-processing stage is implemented in contrast to conventional sequential strategies. This dissertation demon-

strates the proposed post-processing stage alleviates truncation of output bits, enhancing statistical data properties. Secondly, a fully-synthesized TRNG architecture is proposed, offering a multi-throughput operation by way of a new multi-throughput multi-mode ring oscillator that uses an inverter cell bypassing strategy [24]. The proposed TRNG achieves a multi-throughput operation with similar features to the original multi-mode approach. Finally, a methodology to boost entropy in low-cost TRNG chips is introduced [24]. Due to non-idealities, physical entropy sources do not achieve enough quality to pass security standards. Hence, a TRNG needs more than one entropy source to extract high-quality numbers. A two-source extraction methodology is adopted in this dissertation in order to propose low-complexity circuit implementations of low-cost two-sources TRNG chips.

An additional concern of today's industry is the exposure to reverse engineering of security primitives or other intellectual property. Current countermeasures used to provide intellectual property protection still present flaws or unacceptable overheads. Instead, as a low-overhead solution, this dissertation shows the development of a method to obfuscate digital circuits at a layout level [25, 26]. An on-fly standard cell generator is used to feed an obfuscation algorithm that includes random layouts of the same standard cell. The algorithm performs this feeding for different cells depending on the desired obfuscation level. In this way, reverse engineering will take too much time as an attacker does not know which layouts correspond to a specific logic function. The proposed algorithm achieves one of the lowest overhead compared to the state of the art physical obfuscation techniques. This low overhead is achieved due to the obfuscated designs occupy the same area as the originals, and it does not modify the critical path, reducing the timing overhead. This method also achieves a similar security level to other state-of-the-art layout-level camouflage techniques.

As a complement of the leading research line, this dissertation addresses a current hardware bug in the DRAM memories not fixed yet. The bug, called row-hammering, is still present in modern DRAM chips using current countermeasures whereas other countermeasures are prohibited due to the associated overhead. As an alternative solution, a hardware strategy is developed in order to mitigate the so-called bug using a dummy cell that does not alter the memory architecture adding low overhead [27–29]. This dissertation shows that a dummy cell with increased susceptibility to leakage current could work as a row hammering attack indicator. We proved that a dummy cell always loses its information before a standard cell under a row hammering attack.

Thus a proposal to attach dummy cells to each DRAM row is shown, storing the same information. In this way, a memory controller can verify the information in dummy cells, refreshing a specific row when it detects a dummy cell loses its charge. This approach achieves a low-area overhead due to the DRAM array only requiring one dummy per row. This solution also provides with low-performance overhead since the additional refresh operation only appears in the presence of a row hammering attack.

DISSERTATION PREVIEW

This dissertation is organized into six chapters. Chapter 1 discusses the proposed key establishment architecture with a fast rekeying feature. Chapter 2 discusses two fully-synthesized TRNG architectures and a methodology to boost entropy using two entropy sources. Chapter 3 presents a method to camouflage digital circuits at a layout level. Chapter 4 discusses a hardware strategy to mitigate the row hammering bug in DRAM chips. Chapter 5 concludes this dissertation.

1. LIGHT-WEIGHT KEY ESTABLISHMENT ARCHITECTURE

A low-area ASIC implementation of a fully-synthesized symmetric key establishment architecture based on tree parity machines (TPMs) in 130nm and 65nm standard-cell CMOS technologies is presented. The proposed circuit architecture has a rekeying characteristic enabled by two new circuit implementations. The behavioral simulations shows that synchronization time can be reduced from 1.25ms to less 0.7ms with a weight misalignment of 20% in rekeying mode. Relative area and power consumption are studied by comparing synthesized TPMs with an implementation of a CRC16 error detection code used within security applications. Scalability of the architecture is shown by mean of a proposed figure of merit. Further verification is applied for fabrication in 130nm CMOS technology.

1.1. INTRODUCTION

The growth of the amount of data that is stored, transmitted and shared has imposed serious challenges in protecting sensitive information. This increase results in more security flaws because the protection itself resides in the networks and in the ability to have complete control of data. Therefore, the way of implementing information security is changing to data-centric because the data travel longer distances and can be accessed by different users [30].

The need for modern security systems is more evident with the increasing interest in the Internet of Things (IoT). The connectivity of things increases the data flow and thereby the information that needs to be protected. Moreover, the security must be integrated with these systems without affecting their implementation cost while considering that this integration should be at three levels: hardware, software, and applications.

The cost of security integration should be taken into account to decide what is the proper implementation. Software-based implementations are flexible but they consume excessive resources. Solutions in hardware level are important alternatives to avoid the resource consumption of their software counterpart. Therefore, reduced hardware implementations help to optimize the resources available for cryptographic primitives in restrictive systems [31].

Protection of sensitive information faces other problems such as physical attacks. Hardware-

based solutions provide inherent protection against this kind of attacks, increasing the information security. Part of this important information is called “keys” and they determine the performance of the security system.

Keys need to be generated and shared with the intended users. Both processes imply special considerations in order to have the proper level of security. Key generation is performed through primitive functions such as true random number generators that need to accomplish standard statistical qualities. Accordingly, once the key is generated, it is ready to be shared.

In many cases, the associated cost of sharing or establishing a key result in the usage of fixed keys. Security in a system with a fixed key depends on the secrecy of this key. Popular attacks are focused on finding flaws in the key management, that can reveal the secret key [32]. In addition, the implementation of cryptosystems in hardware level limits their flexibility, eliminating the possibility of updating the fixed key or any other security feature.

The establishment of a shared key pair is generally performed through an insecure channel. Therefore, key establishment is based on public cryptography using elliptic curves and hyperelliptic curves. This establishment is very common in embedded systems in order to impede an unintended person extracting information from the channel [10]. However, public cryptography is a software-based solution making the integration in low-cost systems difficult.

Neural cryptography presents a hardware-friendly alternative to implementing key establishment in a completely symmetric way [33]. The usage of tree parity machine (TPM) neural networks in synchronization mode provides an algorithm to perform key establishment with low computational resources. Only a few implementations are reported with incomplete information of final resource usage such as in [10], [32], [34], and [35]. Volkmer [10], [32], proposes an architecture for FPGA in which power consumption is not reported. On the other hand, Mühlbach [34], [35] proposes a synthesized-architecture on FPGA and 180nm CMOS technology in which area is not reported. In addition, Volkmer and Mühlbach do not report experimental results with resources budget.

One interesting feature of the establishment based on neural cryptography is the ability to update the final key very fast. Once, the key is established, the neural networks remain synchronized, updating the key continuously. The way of implementing this process, called rekeying, is open to different proposals and give the flexibility that many hardware-based systems do not provide.

This dissertation proposes a more secure key establishment algorithm with a rekeying feature enabled with two new circuit implementation techniques [21]. First, full control over stored information in a memory device is provided by a modified datapath. Second, additional obfuscation is added by a proposed variable-length pseudo random binary sequence (PRBS) generator to increase the complexity for attackers without significant impact on the resources required. Furthermore, a detailed implementation of a fully-synthesized TPM core to perform key establishment in 130nm and 65nm CMOS technologies is presented. Analysis of resource consumption is made by comparing TPMs implementations with CRC16 error code detection showing the relative consumption to a commonly used circuit in communication systems. A figure of merit (FoM) is proposed to set criteria for comparing different TPM implementations including the key length.

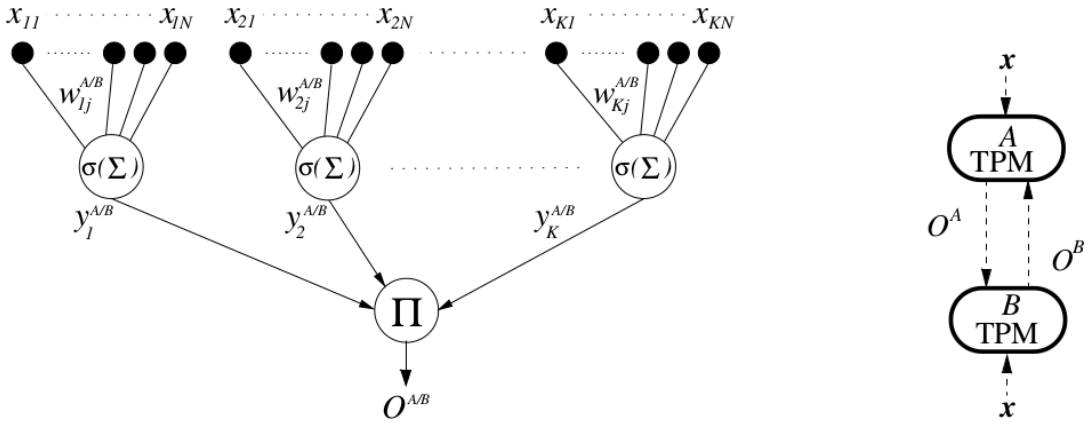
1.2. NEURAL CRYPTOGRAPHY: TPM ALGORITHM

Neural cryptography is an alternative to achieve key establishment in light-weight applications using a special property: two randomly initialized neural networks learning from each other can get synchronized among themselves [36]. These particular feed-forward neural networks, having one layer of hidden units, are called parity machines (PM). Both networks have common inputs and exchange information about their outputs. In case of agreement, the two PMs are trained by a Hebbian rule based on their mutual outputs, resulting in many cases in a complete-synchronization state of synaptic weights. This behavior can work as a possible key exchange protocol for data transmission because it is difficult for an attacker to reveal the common parameters after synchronization [33].

Multi-layer feedforward PMs (tree PM or TPM) show better performance in synchronization mode. A TPM structure has three layers as shown in Fig. 5. The input layer has N non-overlapping binary inputs that are random and common to the networks. There is also one hidden layer with K units (with an input field of K $\cdot$ N) and bounded discrete weights $\mathrm{w}_{kj}^{A/B} \in [-L, L]$. Finally, one binary output $O^{A/B}(t) \in \{-1, 1\}$ is calculated by a parity function of the signs of summations [10].

Through the process of synchronization it is possible to obtain a common key transmitting information that is not directly related to the key itself. Therefore, a key exchange based on TPM can be used as a complement to another type of encryption to increase security [10]. The final

Figure 5. Tree parity machine(TPM) neural network.



key length depends on the TPM parameters and is given by:

$$KEY_{length} = \log_2((2 \times L + 1)^{KN}). \tag{1}$$

The synchronization state remains once the TPMs establish common synaptic weights. However, these weights are continuously updated due to the pseudo-random behavior of the neural networks, resulting in new and equal synaptic weights every synchronization step. Therefore, a fast and easy process of key updating (rekeying) can be carried with this type of establishment.

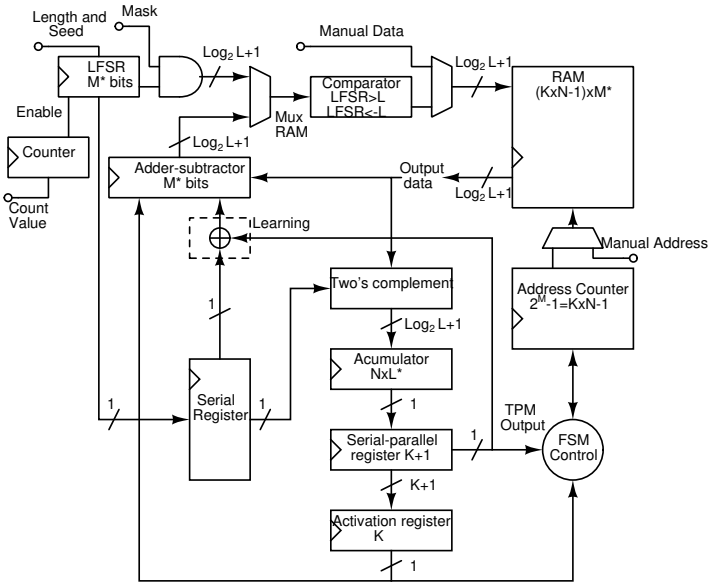
The usage of the rekeying feature has a disadvantage. If more information is transmitted, the possibility of predicting the key is increased. Therefore, the rekeying process should be carefully implemented to avoid revealing sensitive information to an attacker.

1.3. SYMMETRIC KEY ESTABLISHMENT ARCHITECTURE BASED ON TPMS

Hardware-based encryption engine in wireless applications make use of symmetric key algorithms —due to their light-weight calculations— to provide security with low resources. Because of what has been discussed up to this point regarding hardware based encryption, an architecture based on TPMs is an interesting alternative due to the associated light-weight calculations.

The datapath of the developed architecture is shown in Fig. 6. A serial data flow is used to compute the output. The architecture uses a PRBS based on a variable-length linear feedback shift register (LFSR), as shown in Fig. 7. The variable characteristic is implemented using a mux to select what registers are used as feedback, and the serial output of the PRBS is selected in the

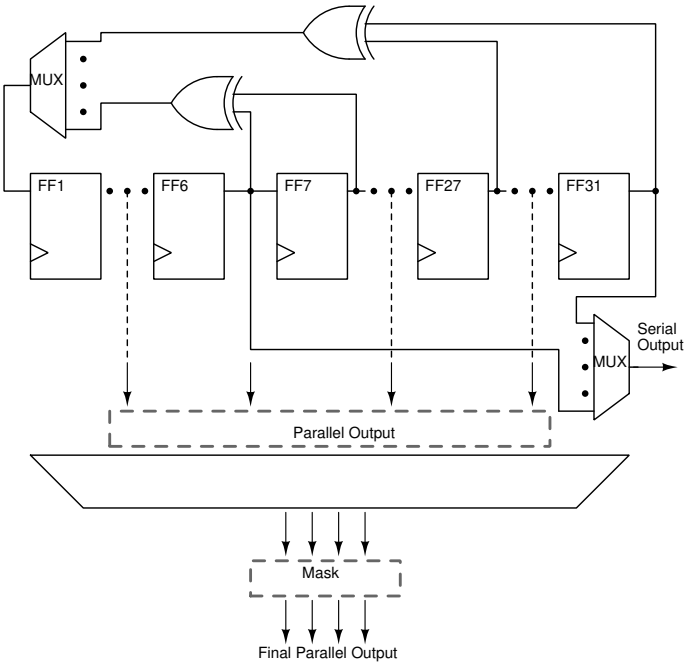
Figure 6. TPM's serial datapath architecture. The datapath features a pseudo-random number generator that establish the initial weights using a secret seed and it is reused to generate the necessary common inputs. Obfuscation is added by complementing the pseudo random generator with a variable length feature and masking its output.



same way as the feedback signals (depending on the required length). Furthermore, a random count enables the PRBS output in order to set the initial synaptic weights From any part of the generated sequence, the final output is selected and masked with logical gates. It is possible to choose all of these options externally and randomly. Besides, the LFSR provides pseudo-random inputs with fixed options using the serial output feature in order to develop the synchronization process.

The datapath uses two's complement notation, hence the necessary word length for the weights is $\log_2(L) + 1$. A comparator is used after the PRBS output to ensure the weights boundaries. A RAM memory stores all the synaptic weights. Being the dataflow serial, weights are modified one at time, according to their sign and the inputs' sign in the two's complement block. Then the accumulator adds every result from each hidden node in the network. The serial-parallel register stores the resulting sign of every unit in the hidden layer and calculates the final parity (TPM output). The activation register verifies which of the hidden nodes have to be modified producing an activation signal for the adder-subtracter in the learning phase where the weights are updated based on a Hebbian rule easily implemented with an XOR operation. Finally, the serial

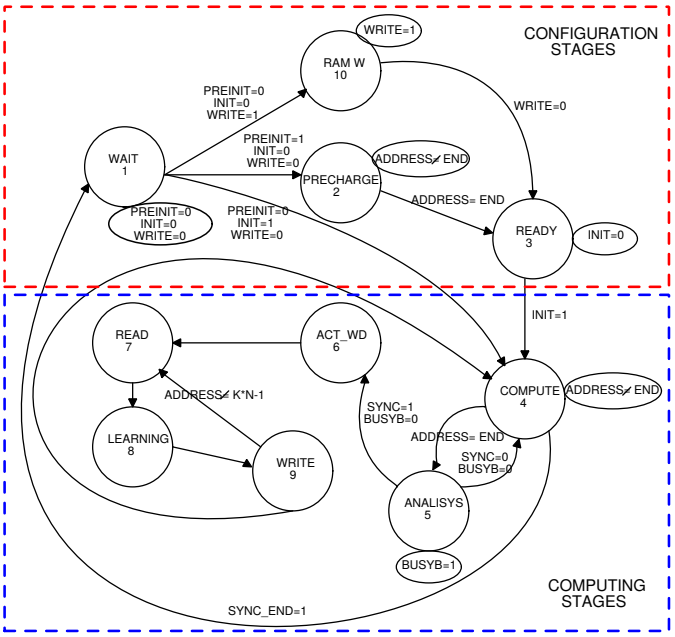
Figure 7. Proposed PRBS with variable length and masked output. The PRBS bases its operation in an LFSR with a muxing strategy in order to primitive polynomials of different degree. PRBS's parallel output is used for weights initialization and includes a combinational mask. The PRBS's serial output delivers the binary inputs for the serial dataflow.



register stores all of training inputs producing the adequate output for the computing and learning stages.

The objective of the architecture, once synchronization occurs, is the fast rekeying. In order to protect the key, the proposed rekeying consists in randomly modifying a part of the synchronized-synaptic weights and rerunning the synchronization process; as consequence, the synaptic weights can be aligned faster than in the initial procedure. Previous works ([32, 35]) are not clear about their rekeying strategy and implementation or directly use the inherent successive key generation. In order to further hindering attacks from hackers, this dissertation proposes additional inputs to access the memory device, providing with full control over stored information in order to change partial weights and re-starting the synchronization process. Moreover, the proposed PRBS allows to pseudo-randomly change these weights, choosing at random the length of the primitive polynomial and the combinational mask. These features improve the statistical properties of rekeying and also obfuscate the process.

Figure 8. TPM's finite state machine. The FSM has four configuration stages that set the initial weights and prepare the PBRS for delivering the binary inputs. The FSM also has six computing stages that carry on the synchronization process obtaining the parity of the units of the hidden layer and updating the weights in case of agreement with the other party.

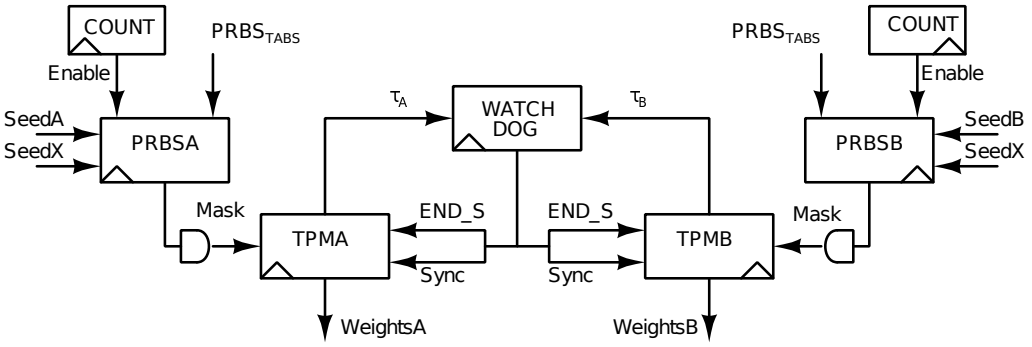


A ten stages finite state machine (FSM) is used as control unit. The objective of the FSM is to allow data serial calculation besides providing control over the proposed-rekeying feature. The FSM uses a pre-initialization to store the initial pseudo-random weights and after the synchronization process the FSM allows the modification of the aligned weights in two ways: a completely arbitrary modification or using the proposed PRBS to provide with better statistical properties.

The FSM can be divided into two parts. The first part, framed in red in Fig. 8, consists of the states one, two, three and ten. The three states control compose the necessary configuration of the TPM in order to perform a key establishment, by setting the seed of the PRBS, loading the synaptic weights and defining the inputs of the network. State ten enables the rekeying characteristic, and reviews and updates the synaptic weights once a synchronization process is done.

The second part of the FSM consists of the states four to nine in the blue box in Fig. 8. These states guide the serial calculations of the datapath. The count of the synchronization steps taken is stored in order to stop the computation process if the alignment does not occur in a determined number of steps. Once the synchronization process is done or interrupted, the FMS waits for a

Figure 9. Complete TPM link with built-in testing platform using a serial peripheral interface(SPI). The circuit uses two independent PRBS for each TPM and the whole process is controlled and verified with the SPI. Final weights are always stored in SPI registers.



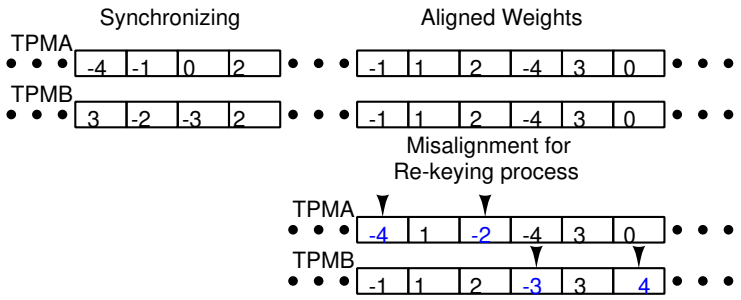
complete new synchronization or a rekeying process.

The synchronization is verified by implementing a TPM link with two different networks learning from each other. The checking of the TPMs alignment is performed by counting successive equal outputs in a sufficiently large number of iterations as indicated in [32]. Figure 9 shows the complete architecture for verification indicating the usage of two different PRBS for each network. A watch module is added to limit the maximum number of synchronization steps in case that TPMs does not achieve the alignment of their synaptic weights. The way of controlling and verifying the synchronization process is through a serial peripheral interface (SPI) which by means of a register modifies the necessary inputs and loads the seeds required by the PRBS. The SPI interface can capture at any time the output information of the TPMS such as synaptic weights, the binary outputs, and even the PRBS sequence in order to fully verify the proper operation of the link.

1.4. RESULTS

Testing of the architecture features was carried out through the execution of 100 synchronization iterations operating at 350MHz. The synchronization process is validated by randomly initializing the two TPMs. The initialization is achieved by loading two different seeds for pseudo-random generators every time the process is re-started. The watchdog count is fixed to 400 synchronization steps as Volkmer [32] suggests, in order to interrupt the process when there is a high probability of having the weights properly aligned.

Figure 10. Example of misalignment in the rekeying process. Some TPM's stored weights are randomly changed for both TPMs, once synchronization happens. The new value, as well as the address of the updated weights, are randomly chosen.



Once the watchdog interrupts the synchronization process, the rekeying characteristic is tested. The testing process was carried out by misalignment the weights, as shown in Fig. 10. Initially, the TPMs have different synaptic weights in the synchronizing stage. Once the weights in both TPMs become equal, the alignment is achieved. Then, the weights are randomly misaligned by about 20 % as shown by the example with blue numbers in Fig. 10, meaning that at least four of the 42 positions of the weights are changed in both TPMs. This change implies different address of the datapath memory for the four positions of each TPM.

During this process, the different features of the PRBS are tested. The testing procedure randomly chooses a specific PBRS polynomial degree, and it applies a different mask to the output, selecting different output bits since the PRBS output always has more bits than required for weights representation.

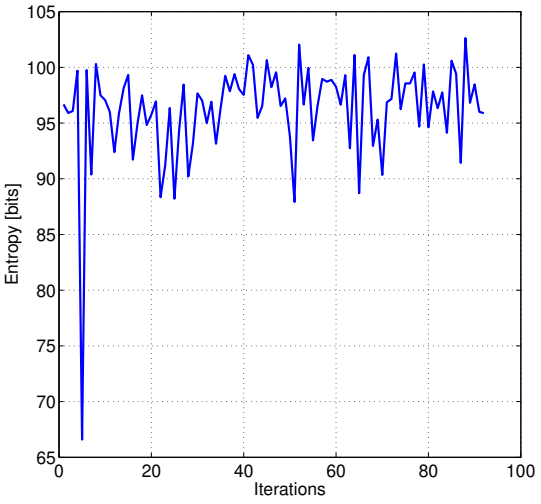
Figure 11 shows the entropy for the resulting weights in the synchronization and rekeying process thought iterations. For the synchronization process, Fig. 11a depicts the resulting entropy with a total of 92 % successful synchronizations. In average, the effective length of the final key is 96 bits of the 132 bits predicted by the equation (1).

Figure 11b shows the entropy after the rekeying process. In average the effective length of the key is 97 bits, better than after the synchronization process. The rekeying process considering a 20% of misalignment has a percentage of successful synchronization of 100%. In this case, only the 92 iterations with a successful initial synchronization were considered.

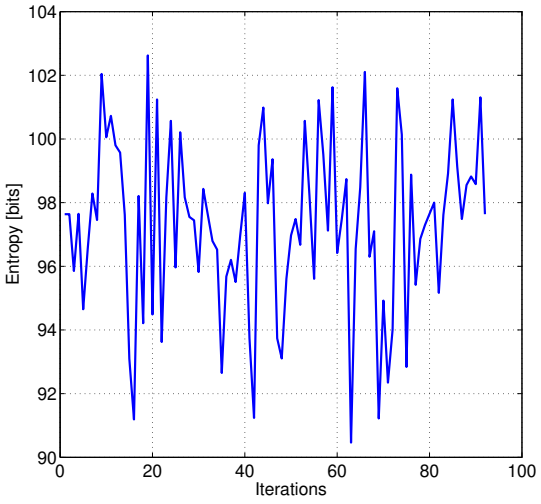
The extraction of the synchronization time in the initial alignment and in the rekeying process is shown in Fig. 12. The distribution of the initial synchronization times in the 92 iterations of the

Figure 11. Entropy of the established key through different synchronizations and rekeying process. Fig. 11a shows an entropy of 96bits for the final key in 92 different events of synchronization each of them with initial weights chosen at random. Fig. 11b shows the key entropy after the rekeying process using a synaptic weights misalignment of 20 % approximately. The results show how the weights' entropy is not affected for the rekeying process showing a value of 97bits.

(a)



(b)



study are shown in Fig. 12a . In this case, the distribution can be adjusted to a normal distribution with a mean of 1.25ms and standard deviation of 0.49ms.

Rekeying features are also analyzed with the same case study with 92 iterations. Figure 12b shows how the distribution of synchronization time is biased. In this case, the synchronization time has more than 90 % of probability of being less than 0.7ms. With these conditions, the alignment process takes about half of a typical one.

Another case of a rekeying process is shown in the Fig. 13. Here, the misalignment is changed to study the influence of this parameter. As expected, reducing the misalignment to 15% implies that synchronization is achieved in less time with a 90% of probability of being under 0.6ms. Therefore, the control of this parameter can be used to estimate more precisely the time that the link needs to establish the key.

The number of used steps to achieve synchronization is about 230. When the rekeying process is carried on, the number of steps is reduced, agreeing with the time analysis done. Using a

Figure 12. Distribution of the synchronization and rekeying time. Fig. 12a shows the distribution of the synchronization time during 92 iterations. Fig. 12b shows how the synchronization time is reduced if only a partial misalignment in rekeying process is applied. The distribution is biased to less than 0.75ms.

(a) (b)

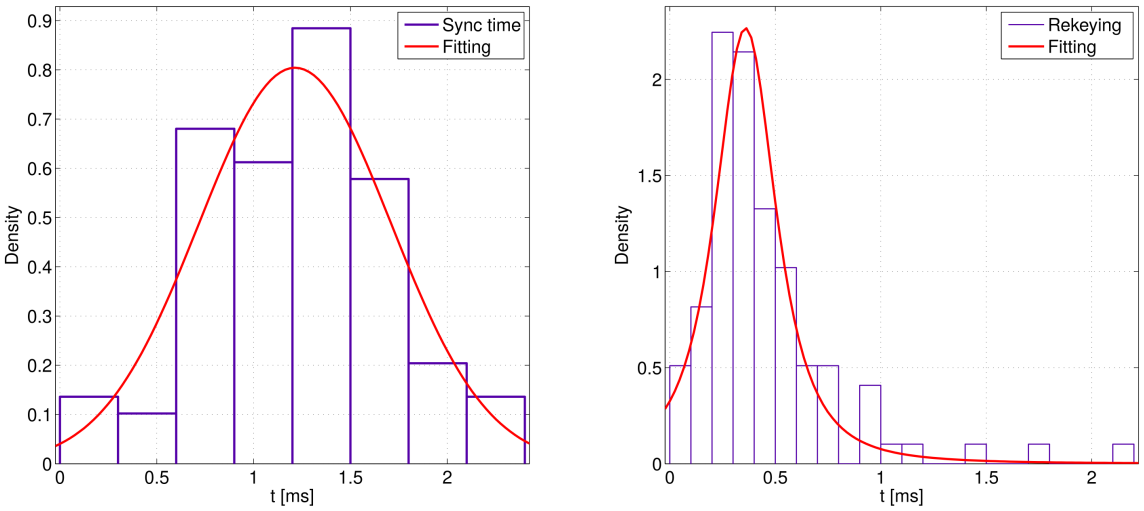
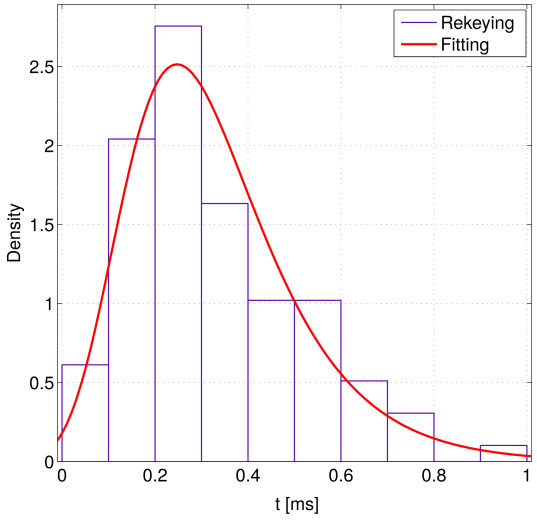


Figure 13. Distribution of the synchronization time after rekeying using a misalignment of 15%.



misalignment of 20% the number of steps is reduced to 88 on average with a similar distribution

Table 1. Library corners.

Corner	Process	Voltage [V]	Temperature [°C]
Worst case	Slow-Slow	V_{DD}-10%	125
Typical case	Typical-Typical	V_{DD}	25
Best case	Fast-Fast	V_{DD}+10%	-40

Table 2. TPM used cells.

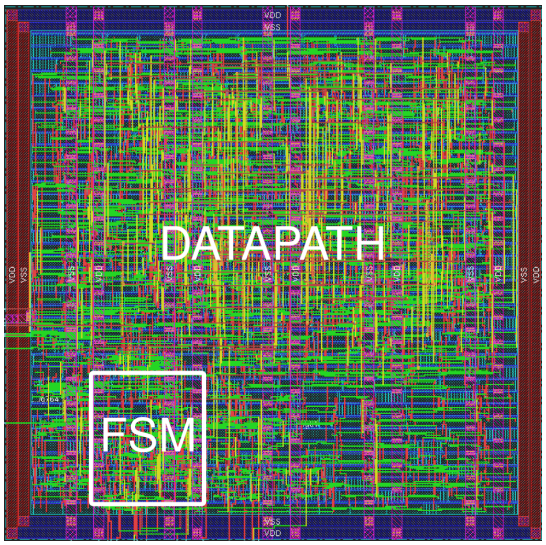
Process	Sequential	Inverters + buffers	logic
Worst case			
130nm	347	100	573
65nm	347	116	562
Typical case			
130nm	347	104	570
65nm	347	120	568
Best case			
130nm	347	107	562
65nm	347	113	566

of the synchronization time. When the misalignment is 15%, the number of steps is reduced to 65, therefore, being the results congruent with the distribution of synchronization time.

1.4.1. Results from synthesized circuit Two different TPMs have been fully-synthesized and verified in two different technologies implementing the RAM memory with registers. The synthesis process is performed by using the different corners available in both technologies 130nm@V_{DD}=1.2V and 65m@V_{DD}=1.2V, corresponding to those exposed in Table 1. Synthesized TPMs have an input field of 42 (N=14 and K=3) with synaptic weights of four given an approximate key length of $K_L = \log_2((2 \times 4 + 1)^{42}) \approx 133\text{bits}$. Concordance of results is evident with the number of sequential cells, which is 347 for both technologies, as shown in Table 2. Moreover, logic cells and the usage of buffers and inverters are similar.

Timing constraints considering the non-ideal clock network provide the maximum operating frequency for the three corner cases and Table 3 shows that this frequency corresponds for 65nm

Figure 14. TPM layout in 130nm. Final area consumption is 160μm$\times$160μm with a double power ring of 3μm of width.



and 130nm to 434MHz and 350MHz respectively in the worst-case condition. More accurate measurement of power consumption is achieved by extracting the activity factor of cells. Table 4 summarizes the energy and area specifications in order to verify how the performance scales with frequency.

Table 3. TPM synthesis results.

Process	Area [μm^2]	Energy [μW/MHz]	Max. Freq. [MHz]
Worst case			
130nm	16600	19.4	350
65nm	4800	7.6	434
Typical case			
130nm	16600	22.4	500
65nm	4800	8.7	667
Best case			
130nm	16600	31.5	770
65nm	4780	10.2	800

A post-synthesis evaluation of the synchronization process is carried out with the same strategy as the behavioral simulations. In both technologies, the TPMs achieve the alignment of their weights in about 200 steps with 40000 clock cycles resulting in about 200 clock periods per train-

Figure 15. TPM layout in 65nm. Final area consumption is 90μm$\times$90μm with a double power ring of 3μm of width.

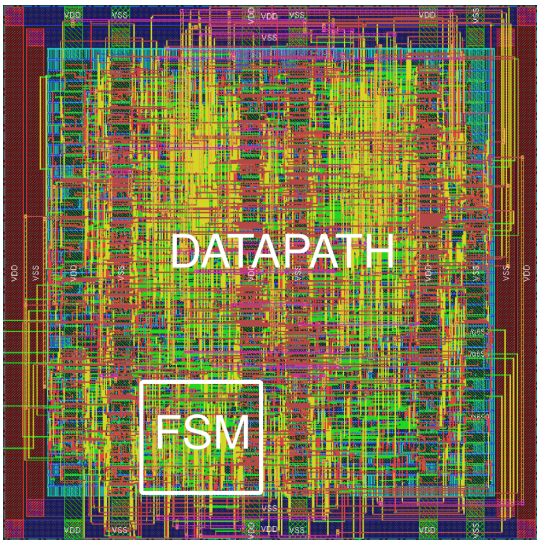


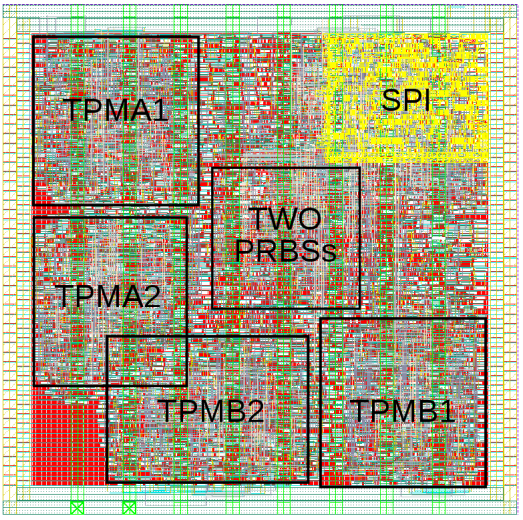
Table 4. Power and speed performance for different frequencies on a 130nm and 65nm process, using typical corners.

Process	Energy [μW/MHz]	Freq. [MHz]
130nm	39	1
65nm	19	1
130nm	28.5	10
65nm	15.3	10
130nm	22.4	100
65nm	10.2	100

ing. The resulting cycles and steps agree with those reported by [32] and [35] for keys around 128 bits considering an ideal communication channel. Also, the rekeying procedure allows for the reduction of the synchronization steps to 80, since the weights are partially aligned at the time of restarting the system.

1.4.2. Chip layout Layout for both technologies is built by using the synthesis results of worst-case library and taking into account the clock tree specification. Figures 14 and 15 show the final layout for both technologies with power rings. Besides, figures highlight the area and placement of datapath and FSM, which help to verify the accomplishment of constraints. Final layouts areas are 160μm$\times$160μm and 90μm$\times$90μm for 130nm and 65nm respectively wherein

Figure 16. The final layout in 130nm of the complete TPM link. In the upper high corner the SPI interface is shown. Final area consumption is 400μm$\times$400μm with a double power ring of 10μm width.



both cases the datapath occupies most of the area.

A chip was sent to fabrication in 130nm. Fig. 16 shows the layout in 130nm technology for complete testing. The area is 400μm$\times$400μm including the SPI interface. Two different links compose the final chip. The links consist of two pairs of TPMs, each one with different key lengths. The expected key of the two links is over 128 bits that is a typical value used in cryptographic standards such as the Advanced Encryption Standard (AES). The difference between the implemented TPMs is the input length, odd input length (N=11) and even input length (N=14), and the bounded weights ($[-4, 4]$ and $[-7, 7]$).

Figure 17 shows chip's sign-off layout. A power grid was created to minimize IR drop and electro-migration issues. Moreover, empty space is used to put decoupling capacitors.

An RC extraction of the chip's layout was performed to verify further that there was nor IR drop neither electromigration issues. This extraction was also used to carry on transient simulations in order to compare these results against behavioral and post-place and route results. Simulations result shown a proper operation, achieving the expected maximum operating frequency.

1.4.3. A cyclic redundancy check (CRC) implementation

Communication systems typically use error correction and/or detection coding strategies such as CRC and Hamming.

Figure 17. Complete sign-off layout of a whole TPM link. A power grid and decoupling capacitors are used to improve reliability of the chip.

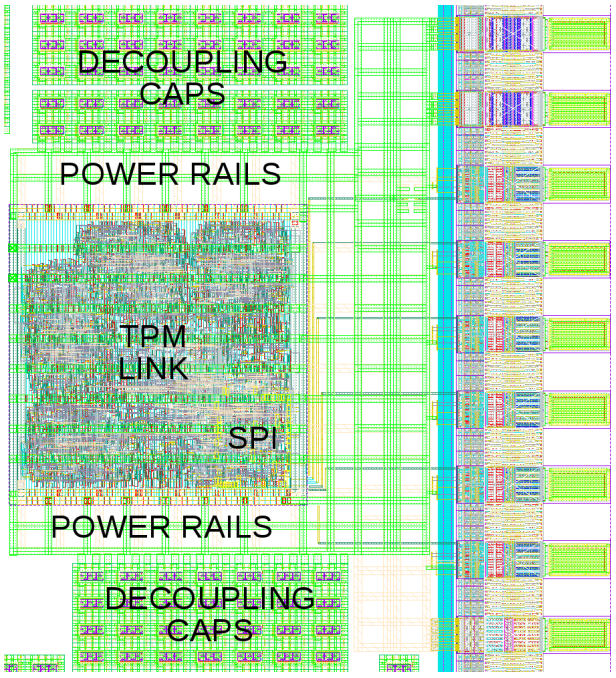
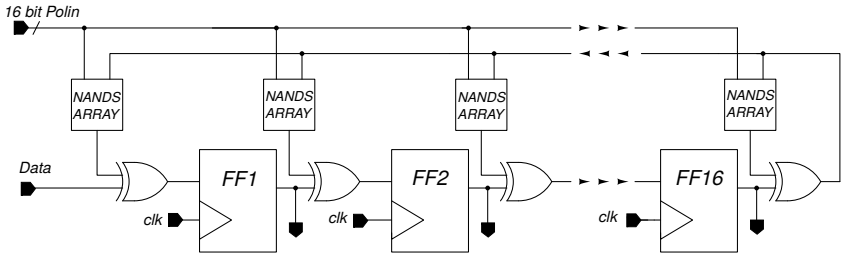


Figure 18. CRC16 block diagram used as a reference.



Wireless systems such as smart cards use CRC due to the algorithm's computational efficiency to detect errors in data transmission. For example, Lee [37] proposes the use of a CRC16 in an RFID passive tag with cryptographic functions where the main limitation is that power supply that comes from the tag antenna. Moreover, this kind of tag should be not only power efficient but also area efficient to reduce chip cost. The impact of cryptographic functions such as the TPM algorithm in area utilization can be estimated by making a comparison with other components implemented within security applications, as for example, CRC16.

Communication systems typically use error correction and/or detection coding strategies such as CRC and Hamming. Wireless systems such as smart cards use CRC due to the algorithm's

Table 5. Number of leaf cells used to implement a CRC block.

Process	Sequential	Inverters + buffers	logic
130nm	16	7	66
65nm	16	8	66

Table 6. CRC synthesis results and comparison with TPMs using the proposed FoM at worst corner case.

Process	Area [μm^2]	Energy [uW/MHz]	Freq. [MHz]	FoM [bit^{-1}]
130nm	930	1.7	350	1.53
65nm	282	0.93	350	1.34*

*Energy for TPM at 350MHz is 9.8uW/MHz.

computational efficiency to detect errors in data transmission. For example, Lee [37] proposes the use of a CRC16 in an RFID passive tag with cryptographic functions where the main limitation is that power supply that comes from the tag antenna. Moreover, this kind of tag should be not only power efficient but also area efficient to reduce chip cost. The impact of cryptographic functions such as the TPM algorithm in area utilization can be estimated by making a comparison with other components implemented within security applications, as for example, CRC16.

Figure 18 shows the implemented CRC16 block used as a reference. The CRC's features programmable characteristic in order to check different generator polynomials. Table 5 gives a summary of total leaf cells needed to implement the CRC with 16 sequential cells for both 130nm and 65nm technologies using the worst case library. Area and power specification are given in Table 6, including operation frequencies.

Quantification of comparison between TPM and CRC16 implementation is made through the proposed figure of merit (FoM) given by equation(2). The FoM includes area and energy consumption, besides taking into account the key length of the establishment process. This FoM allows the comparison of different implementations based on the resources consumption and the effective key length, being better the implementation with the minimum FoM value. The FoM for related works such as those reported by Volkmer [10], [32] and Mühlbach [34], [35] cannot be calculated due to the absence of used resources. Moreover, neither system integration nor layout is reported by Volkmer and Mühlbach.

Figure 19. Micro-photograph of fabricated TPM.

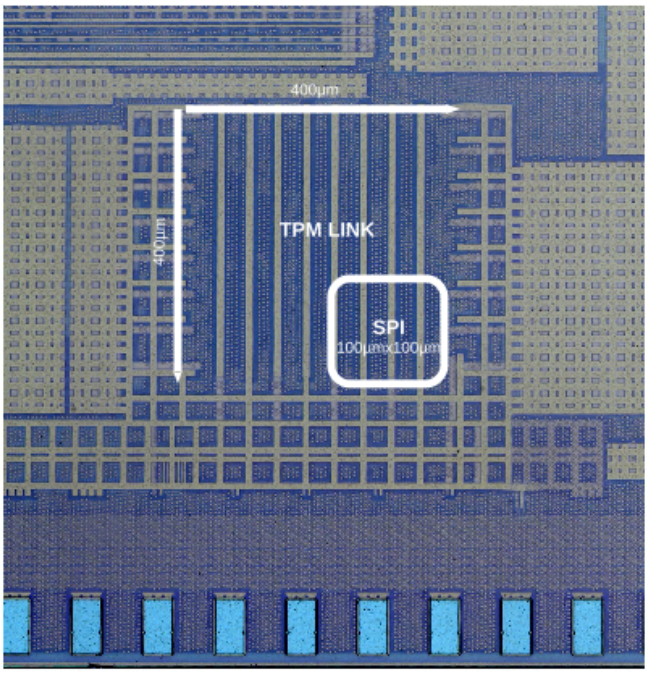


Table 7. Elliptic curve establishment core vs TPM core.

	logic gates	Energy [uW/MHz]	Freq. [MHz]
[38]	>6000	60	0.5
This work	∼1000	54.7	0.5

$$FoM = \left(\frac{Engine_{Energy}}{CRC16_{Energy}} \times \frac{Engine_{area}}{CRC16_{area}} \times \frac{1}{Key_{length}} \right) \quad (2)$$

Table 7 summarizes a performance comparison of the proposed estabishlment core and the work presented in [38]. Houss et al. [38] present different establishment cores based on elliptic curve in a 130nm CMOS technology. Table shows that the core here proposed uses less logic gates with a similar energy efficiency, illustrating how the low complexity algorithm used enables a compact core implementation.

1.5. ON THE KEY ESTABLIHSMENT ENERGY EFFICIENCY FOR WIRELESS IOT NODES

As mentioned before, security schemes for restricted devices such as most IoT nodes often use fixed key systems to avoid the associated resources that key updating implies, but physical attacks have demonstrated that is possible to reveal a stored fixed key. Flexibility is another reason why key updating is an interesting option due to cryptographic algorithms may use different key lengths for encryption. However, a key establishment implies information exchange, and not only the energy efficiency of the algorithm must be taken into account but also the associated cost of transmitting/receiving information.

For instance, Janhunen et al. [39] offer an IoT wireless node for harvesting applications. The communication system uses a commercial chip developed by Texas Instruments. The chip has a Bluetooth transceiver with a payload of 13bytes and a microcontroller. The energy during transmission is about 31μJ in a time window of 2.5ms that includes package preparation, radio transmission and radio deactivation. Assuming the proposed TPM requires 200 data exchanges, the energy cost for a complete synchronization is 6.2mJ. This energy may be reduced if a modified TPM data exchange is used [40], [32]. Instead of using only one bit per transmission, the system can generate a package of ”b” bits TPM generated outputs. For instance, if a package of 32bits is used, the synchronization time increases(number of TPM single output bit) but due to every transmitted package composes of 32 bits the number information exchanges is reduced. With a package of 32 bits, the synchronization time might increase about three times (600 for the proposed TPM) [40], but the information exchanged is reduced by the package bit-width, resulting in the need for only $\frac{600}{32}$ $\approx$7 packages exchanges and corresponding energy consumption of 217μJ. Finally, the TPM package variant also improves the usage of transceiver payload considering in the one bit per transmission version, most of the payload is unused.

1.6. CHIP MEASURMENT ISSUES

Fig. 19 shows the micro-photograph of the fabricated chip. Despite having fully verified the behavior of the post-synthesis and post-place-and-route circuit, no measurement was possible. Integration issues caused that some inputs were left open such as the main clock and one con-

figuration register. Besides, pad configuration pins were shorted to a supply voltage in a way that some pads were always disabled. As a consequence, the prototype could not be tested.

1.7. CHAPTER SUMMARY

A fully-synthesized TPM core in 65nm and 130nm technology has been presented. The proposed TPM implementation enables the key establishment and rekeying using a serial datapath and a FSM to control the process of synchronization and rekeying. A post-processed seed using variable length PRBS adds obfuscation to the rekeying, increasing the hardening of the system. Two proposed circuits techniques were implemented to reduce implementation cost. An improved rekeying feature is achieved by using a PRBS with variable length and a masked output. This PRBS is reused to generate initial weights and in synchronization process to generate pseudo random inputs in order to reduce the implementation cost. Simulation results show the proper operation of synchronization and rekeying features of the proposed architecture.

The entropy of the final key shows that TPMs achieve an effective length close to the estimated value. Moreover, the synchronization time is studied for 100 hundred iterations resulting in a probability of weights alignment of 92%, taking 1.25ms in average to synchronize. The rekeying characteristic is checked by restarting the synchronization process with a misalignment in the synaptic weights. The misalignment is applied in different memory addresses at random, obtaining the new weights from the PRBS. The features of the PRBS are activated also at random in order to verify the whole functionality of the architecture. The simulations results show that with a misalignment of 15 and 20% the synchronization time is less than 0.7ms in more than 90% of cases. The synchronization steps are reduced from 220 (typical establishment) to less than 100.

This proposed implementation can be used in low-cost systems such as in radio-frequency identification (RFID) where symmetric cryptography is mandatory for security purposes, implying the need of a shared key. Common low-cost systems usually use well-known CRC algorithms to detect errors. Therefore, a CRC implementation is appropriate to be used as a measurement reference for size and power specifications. A FoM is proposed to study resources consumption relative to CRC16 implementation showing the scalability of the circuital architecture.

2. INTEGRATED RANDOM NUMBER GENERATOR

In this chapter, we present two fully-synthesized TRNG architectures. First, a TRNG enhanced with a cellular automaton based post-processing stage is shown. The TRNG is tested in an FPGA platform. Then, a multi-throughput TRNG using a multi-throughput multi-mode oscillator is proposed. Finally, measurement results of the fabricated chip implementing the multi-throughput approach on a 180nm technology are presented.

2.1. INTRODUCTION

The internet of things (IoT) may be defined as the interconnection among things/devices that allows them to exchange data. IoT applications are many, from smart home appliances to wearables, transportation, and health-care. Recent projections foresee hundreds of billions of connected IoT devices, with a market worth billions of dollars by 2025 [41].

But interconnecting devices raises challenges such as: creating knowledge or extracting meaningful information from the vast amount of raw data captured; or ensuring privacy in devices that most of the time use wireless communication, besides having a minimal computational capacity for implementing robust security solutions [1]. The mentioned issues related to security might bring vulnerabilities, making systems hackable or susceptible to information leakage. Such broad exposure results in the necessity of replacing the classical security approaches.

Data-centric security has appeared as an alternative to replace traditional security schemes. With a data-centric approach, a security system can maintain better control of the users that can access or change the information. Besides, this type of security model requires the use of encryption with high-quality keys. Generation of these keys involves security primitives called True Random Number Generators (TRNG). TRNGs base their operation on entropy sources from a natural physical phenomenon, such as device noise [42].

Device noise provides entropy sources that can achieve high-quality standards. However, entropy sources may produce biased streams due to environmental and process variations. Biasing makes TRNG's output predictable, which means vulnerabilities. The need for a biasing mitigation strategy and low complexity circuits has increased the interest in digital domain entropy sources.

Digital circuits are more straightforward to implement than analog ones with inherent robustness to process, voltage and temperature (PVT) variations [7, 8, 43]. Inherent robustness helps reducing complex calibration schemes. However, TRNGs by themselves do not always achieve high-quality standards, requiring a post-processing stage [44]. Computational intensive post-processing can improve statistical qualities, but this solution might not be longer valid in restricted environments such as wearable or sensor nodes. Therefore, limited systems require low-cost TRNG IPs, which still must pass statistical standards.

Another issue of using post-processing is throughput limitation. Extracting few bits will guarantee better statistical performance, but reducing the effective amount of generated bits. Although digital TRNGs have been reported without post-processing [6,7], there is still a need to apply computing intensive calibration algorithms [7–9], or the limited application of output bits. Based upon mentioned issues and current approaches to mitigate them, the design of TRNGs still needs more exploration to bring robust circuit implementations, fulfilling the required quality standards.

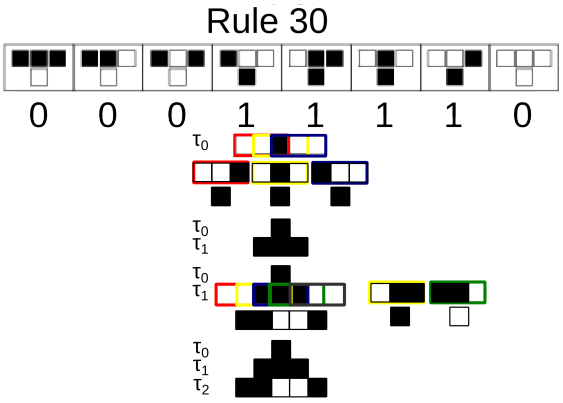
2.2. TRNG WITH A CELLULAR AUTOMATON BASED POST-PROCESSIN STAGE

This section reports an all digital TRNG with a low overhead post-processing stage. The entropy source corresponds to multi-mode oscillators organized in four pairs of different number of inverter stages to analyze the trade-off between throughput and statistical qualities. The post-processing stage bases its operation on a pure combinational implementation of an array of cellular automaton, operating over 1GHz without affecting TRNG throughput. The proposed scheme enables high throughput operation using ring oscillators of 15 inverter stages with good statistical qualities after post-processing. An FPGA implementation enables a verification applying NIST tests, passing 12 tests with reduced truncation. The TRNG is synthesized in a 130nm technology;details about the integration are also provided in this section.

2.2.1. Cellular automata background

Cellular Automata are discrete differential equations that develop a series of binary numbers through iterations. There are many ways of arranging the equation coefficients, resulting in different cellular automata behaviors [45]. As discrete models, these equations might be useful to predict the behavior of physical, biological, chemical and other systems.

Figure 20. Example of the definition of a rule an its output calculation.



A one-dimensional cellular automata has three coefficients resulting in eight different combinations with a single output bit. These combinations can be arranged in 256 different sets called rules. Table 8 illustrates one possible combination, called rule 30 by its binary notation. Every rule has different behavior and is classified into four types of classes [46]. However, there only exists a group of independent rules that can reproduce, by linear transformations, the behavior of remaining rules.

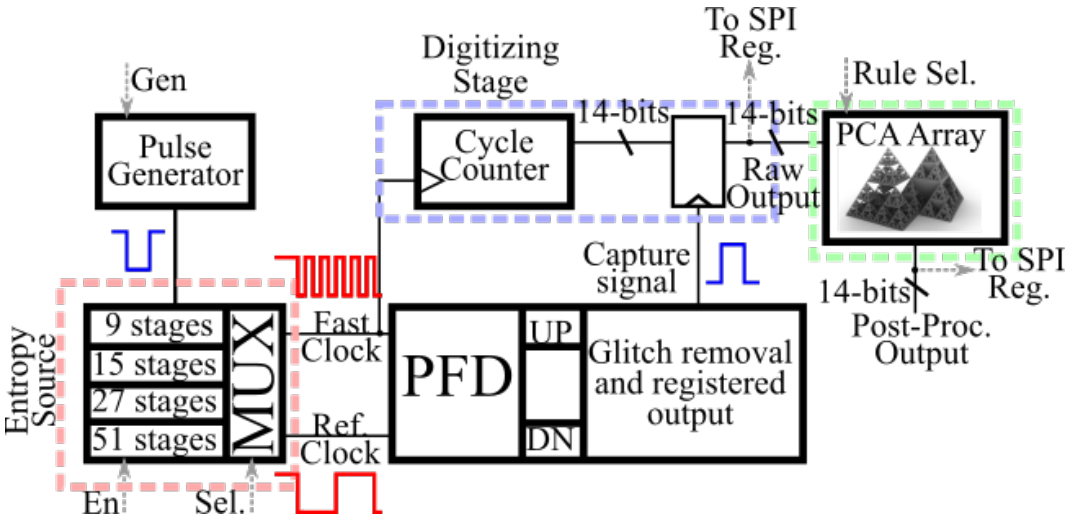
Table 8. Definition of Rule 30.

State	111	110	101	100	011	010	001	000
Output	0	0	0	1	1	1	1	0
30 dec	128	64	32	16	8	4	2	1

The disordered behavior of some rules enables their use in pseudo-random number generators. The main characteristic of these rules is a chaotic or aperiodic behavior during time evolution, which groups them in class III [46]. Nevertheless, only some rules have proved to be useful in a pseudo-random generation such as 30, 45, 60, 90, 105, 150 and 165.

Figure 20 shows an example of how to calculate each possible output using rule 30. The number 30 in binary is constructed by the arrange of the three input coefficients in the eight possible ways. In Fig. 20, the black boxes represent a logic one, and white boxes represent a logic zero. The initial condition is set to $\tau_0 = 010$, using zero boundary conditions. Time evolution of rule 30 is as follows: First, using τ_0 and zero boundary conditions, output τ_1 has three possibilities. The coefficients' arrangement corresponds to 001(red), 010 (yellow) and 100 (blue) which produce

Figure 21. Proposed implementation of the TRNG using programmable cellular automata as post-processing stage.



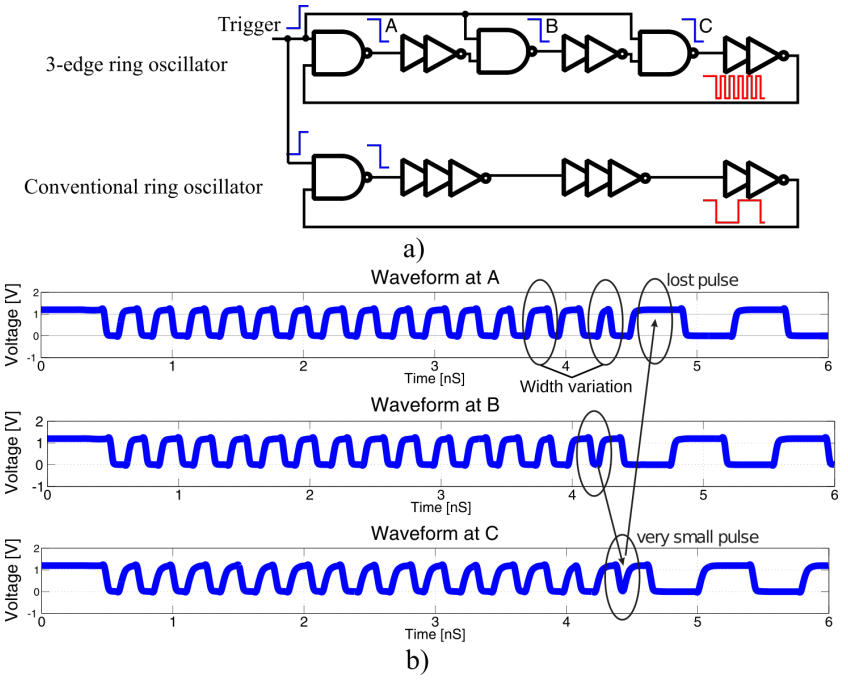
three logic ones i.e., τ_1=111. In the next step, the output has five possibilities different from zero, which lead to τ_2=11001. The resulting pyramid presents a pseudo-random behavior with infinite possible outputs, maintaining zero boundary conditions.

2.2.2. Proposed random generator scheme

Figure 21 shows the TRNG system block diagram consisting of a multiplexed entropy source, a digitizing stage, and a post-processing stage. The entropy source has four pairs of multiplexed ring oscillators with a pulse or trigger generator. The digitizing stage bases its operation on a cycle counter with a registered output where a phase and frequency detector (PFD) triggers the raw register data. The post-processing stage, based on a programmable cellular automata(PCA), processes the raw data in a combinational way. Finally, an SPI interface controls the TRNG, selecting the ring oscillators length, programming the desired rule in the PCA array and storing the raw data and the post-processed data. The following subsections describe in detail the main blocks of the proposed TRNG.

Entropy source and digitization The proposed entropy source comes from jitter accumulation in a multi-mode ring oscillator (RO). Figure 22a) illustrates the concept: A pair of nine-inverter-stages ROs has a synchronized trigger that injects edges in different points. In a conventional RO (bottom of Fig. 22a)), a NAND gate enables and disables the oscillation. When

Figure 22. Time graph of oscillations generated at different nodes of a multi-mode ring oscillator: Node A (*a*), Node B (*b*) and node C (*c*).



a logical one is applied to the enable input (Trigger), an edge is injected, spreading all along the chain of inverters, causing an oscillation with a nominal frequency. Similarly, when a logical is applied to the RO trigger input (bottom of Fig. 22a)), three edges simultaneously appear in the NAND gates outputs, generating an oscillation at 3$\times$ nominal frequency spreading along the ring.

The high-frequency oscillation in the multi-mode RO is finite over time since the injected pulses accumulate jitter from thermal noise. This accumulation causes a duty cycle variation that weakens the edges. Fig. 22b) depicts this effect with a transient noise simulation from signals A, B and C of the multi-mode RO (Fig. 22a)) . Here, the duty cycle varies over time at 2.1Ghz. Black circles indicate how two adjacent pulses (signals B and C) weaken with each completed cycle. After some cycles, two adjacent edges collapsing making the oscillator to operate at a nominal frequency of 700MHz.

Fig. 23 shows a transient simulation with a longer collapsing time (bottom waveform) where the top waveform shows the frequency variation over time. The top figure shows how the frequency goes to zero when the *trigger* input is deasserted, and then goes up to 2.1GHz. This frequency mode weakens with every completed cycle and finally collapses to 700MHz. The bot-

Figure 23. Time and frequency graph of proposed 3-edge ring oscillator: frequency (*top*), time (*bottom*).

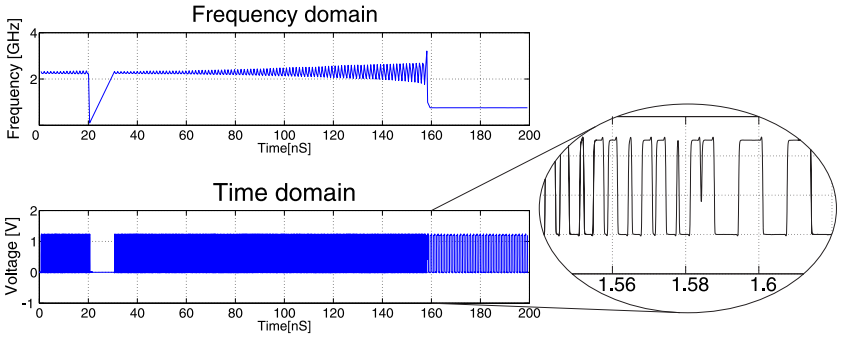
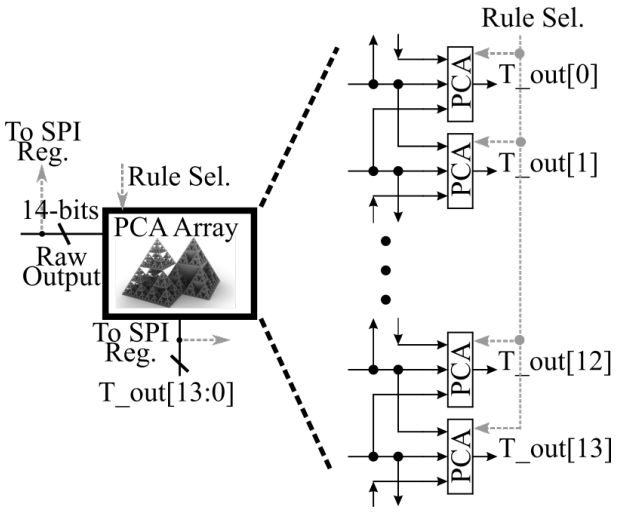


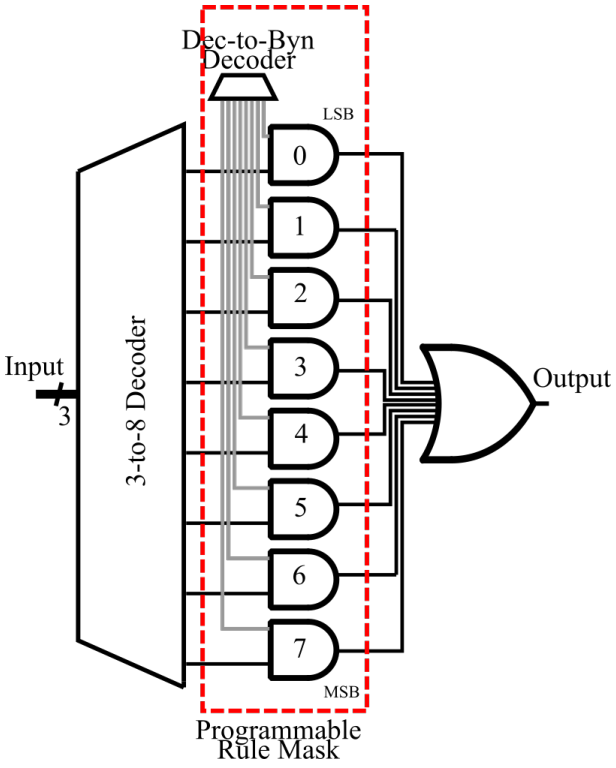
Figure 24. PCA array with SPI control. Each cellular automata uses cycle boundary conditions as initial state. A combinational function produces the 14-bit output.



tom figure illustrates the same behavior in the time domain, where the collapsing event occurs similarly than in Fig. 22. Collapsing time is unpredictable due to jitter accumulation, which increases with the number of stages. Finally, pulse variance also increases with the number of stages, enhancing thus the statistical properties of entropy source [47].

The proposed entropy source consists of four pairs of multiplexed ROs. Each pair of ROs has different stages number providing different statistical properties. As the number of its stages increases, an RO operates at a lower frequency, reducing throughput. However, statistical properties improve [7]. An SPI control signal can disable the ROs, and a shift-register based pulse generator triggers the oscillation. A clock mux provides one fast clock and one reference clock

Figure 25. Proposed programmable cellular automata (PCA). The PCA can be programmed to implement 255 rules.



from the four available possibilities. This scheme enables a comparison between sources with different statistical properties without the need to replicate the whole system.

The uncertainty in a collapse event provides the unpredictable behavior of the TRNG. The time to collapse is converted to a digital number through a digitizing stage based on a cycle counter as Fig. 21 shows. The cycle counter uses the fast frequency oscillation as the input clock to provide a binary number that is then registered. A phase and frequency detector (PFD) based stage generates a trigger signal for the register. The PFD UP signal is always active during the high-frequency oscillation and then goes to zero. To avoid false triggers, some extra logic —that requires the activation of the fourth less significant bit of the counter—provides a glitch-free pulse to the output register. This stage delivers raw data that is the input of the post-processing stage.

Proposed post-processing stage Fig. 24 shows the proposed cellular automata based post-processing stage. The stage consists of an array of programmable blocks that compute the 14 output bits using a combinational function, which means the stage only consumes

Table 9. Area comparison between the proposed PCA and fixed rule cellular automata.

Name	Cells	Area[μm^2]
PCA array	340	2.319
Fixed Rule_105	140	1.284
Fixed Rule_150	112	1.283
Fixed Rule_85	126	1.045

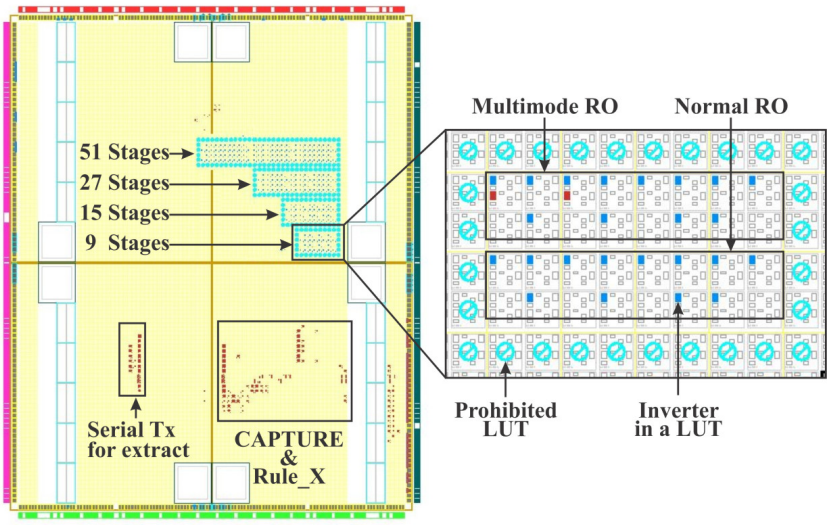
power with new input data. The array receives 14-bits of raw data in such a way that each programmable block (PCA) input can fulfill three bits of the initial state forming cycle boundary conditions. Thus, each PCA shares the adjacent bits to complete the state such that first PCA block uses 13, 0 and 1 bit of the raw data and the second PCA block uses 0, 1 and 2 bits, for instance. Besides, the SPI controls the array, selecting the rule and also storing the array output.

Figure 25 shows the internal structure of a PCA block. The block consists of a 3-to-8 decoder, a programmable rule mask and an output OR. The input decoder uses the 3-bits initial state to active the corresponding AND gates in the rule mask, one at a time. Following the example of rule 30 in Fig. 20, an initial state of 001 will activate the first AND in the rule mask. The corresponding rule mask must be pre-charged with the number 30 in binary. Then, a logic one in the output of first AND will appear, as the rule 30 actives from the first to the fourth AND gate. An OR gate computes the final output, resulting in a logic one as corresponding in rule 30 to the initial state 001.

To save area and to enable testing under different rules, a programmable approach is here proposed. Different approaches were synthesized on a 130nm technology to illustrate how the stage is area efficient. Table 9 shows that the proposed PCA array, though bigger than a simple fixed rule implementation, is still area efficient as it can implement different rules with the same structure, enabling high flexibility. For example, the total area occupied by the sum of 105, 150 and 85 rules implementations is 3612μm^2 which is bigger than the one occupied by the proposed array.

2.2.3. Measurements results We tested the TRNG under an FPGA platform, also generating a circuit implementation on a 130nm standard technology. A guided placement guarantees each pair of ROs nominal frequency are similar. Further placement restrictions were use

Figure 26. Annotated layout in FPGA. Layout shows the placement restrictions of the four entropy sources. Final implementation uses 233 slices, 36 flip flops and 409 LUTs.



in order to reduce external interference. This subsection discusses the statistical analysis of the PCA implementation carried out on FPGA platform and gives details about the final integration in 130nm. In order to extract the statistical qualities of the system, a set of test given by the United States National Institute of Standards and Technologies (NIST) were carried out [48].

NIST publication [48] includes a total of 15 tests. A sequence is considered random when all tests are passed, for example, having an equal number of ones and zeros. However, that is not the only feature the tests evaluate. The tests also evaluate a sequence in blocks in order to look for the longest run of ones expected in a random sequence which might fail even though the whole sequence has an equal number of ones and zeros. Besides, NIST publication provides software that evaluates a sequence over the 17 tests, offering all of tests results at a time or individually. Passing rate depends on the number of sequences tested, corresponding to 96 when 100 input sequences are used. Moreover, the software calculates a parameter called *P-value* that indicates the strength of the evidence against the hypothesis that the tested sequence is not random. A *P-value* of 1 corresponds to a perfect random number generator. A level of significance level(α) is set such that if *P-value*$\geq \alpha$ the sequence appears to be random. Typical values for the significance level are in the range [0.001,0.01]. Both *P-value* significance level and passing rate must be fulfilled by the tested sequence to be considered as random.

Table 10. Some measured NIST randomness test results with *P-values* for no ruled applied data and post-processed data for the 9-stage ring oscillator based TRNG.

NIST Pub 800-22, rev 1a. 2010 Randomness Tests	No Rule Applied		Rule 30		Rule 150	
	P-value	RATE	*P-value*	RATE	*P-value*	RATE
Frequency	0.000000	38	0.000000	0	0.000000	0
Block Frequency	0.000000	0	0.000000	0	0.000000	0
Cumulative Sums	0.000000	15	0.000000	0	0.000000	0
Runs	0.000000	0	0.000000	0	0.000000	0
Longest Run	0.000000	0	0.000000	0	0.000000	0
Rank	0.000000	6	0.000000	0	0.000000	0
FFT	0.000000	0	0.000000	0	0.000000	0
Non-Overlapping Template*	0.000000	135	0.000000	120	0.000000	118
Overlapping Template	0.000000	0	0.000000	0	0.000000	0
Serial	0.000000	0	0.000000	0	0.000000	0
Linear Complexity	0.324301	96	0.213309	97	0.153763	98
Approximate Entropy	0.000000	0	0.000000	0	0.000000	0

* This test composes of 148 sub-tests. The sequence must pass all sub-tests.

Table 11. Measured NIST randomness test results for the 15-stage ring oscillator based TRNG.

NIST Pub 800-22, rev 1a. 2010 Randomness Tests	No Rule Applied	51	75	85	105	150	180
Frequency	94	94	98	97	100	100	98
Block Frequency	95	95	100	100	99	99	100
Cumulative Sums	93	93	99	97	100	100	99
Runs	98	98	90	96	99	99	90
Longest Run	99	98	100	98	99	100	100
Rank	99	99	99	97	97	99	98
FFT	99	99	100	99	100	100	100
Non-Overlapping Template	146	146	142	146	148	148	142
Overlapping Template	100	99	98	100	98	99	98
Serial	98	98	97	97	99	99	97
Linear Complexity	96	99	100	98	97	99	99
Approximate Entropy	91	91	98	97	90	90	98
Sum	1388	1389	1414	1415	1424	1430	1412

FPGA implementation The proposed FPGA implementation uses look-up tables (LUTs) based inverters and NANDs. LUTs enable the use of *x,y* coordinates for placement, facilitating the definition of forbidden and specific location for each pair of ROs. Fig. 26 shows a final layout of TRNG in a Xilinx SPARTAN 3AN. The figure highlights the entropy source and the logic for digitizing and post-processing. Finally, a detailed view of the shorter pair of ROs in the right side shows blue areas around that correspond to the LUTs with prohibited locations.

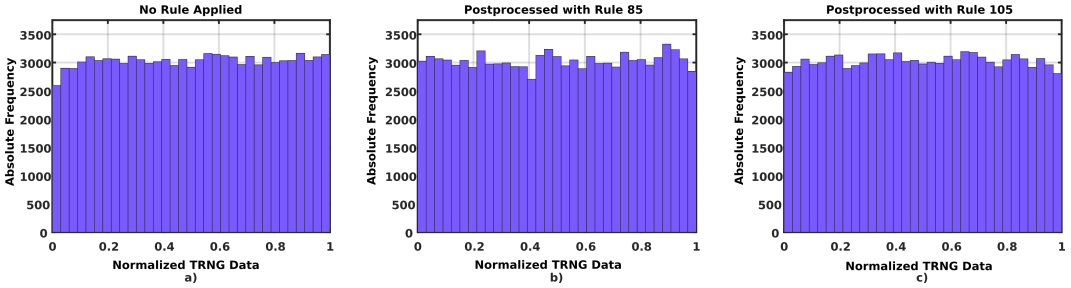
Programmability of the proposed TRNG allows us to test all the independent cellular automata rules. We capture a total of 78Mbits per rule for the four different pairs of ROs. One of the aims here was reducing data truncation, a strategy followed by Yang et al. [47], while also using shorter ROs. Thus, the emphasis is given to the results of the 15-stages and 9-stages pair of ROs.

Table 10 shows NIST tests results for the shortest ring. Different rules were applied to post-process the output data. Even though, the table shows how no significant changes in tests output

Table 12. Measured NIST randomness test results with *P-values* for no ruled applied data and post-processed with rules 105 and 150 for the 15-stage ring oscillator based TRNG.

NIST Pub 800-22, rev 1a. 2010	No Rule Applied		Rule 105		Rule 150	
Randomness Tests	*P-value*	RATE	*P-value*	RATE	*P-value*	RATE
Frequency	0.000000	94	0.000818	100	0.000818	100
Block Frequency	0.000001	95	0.983453	99	0.983453	99
Cumulative Sums	0.000000	93	0.867692	100	0.867692	100
Runs	0.202268	98	0.145326	99	0.145326	99
Longest Run	0.162606	99	0.798139	99	0.304126	100
Rank	0.145326	99	0.129620	97	0.005358	99
FFT	0.275709	99	0.275709	100	0.275709	100
Non-Overlapping Template	0.514124	146	0.699313	148	0.779188	148
Overlapping Template	0.455937	100	0.137282	98	0.759756	99
Serial	0.000145	98	0.595549	99	0.595549	99
Linear Complexity	0.213309	96	0.020548	97	0.366918	99
Approximate Entropy	0.000000	91	0.000000	90	0.000000	90

Figure 27. Resulting Histogram for a) raw data, b) rule 85 and c) rule 105.



can be achieved. Table 11 summarizes the results for the 15-stages ROs, comparing the raw data and the post-processed data for 6 of the most relevant post-processed results. In this case, the results show how the post-processing enhance statistics compared when no rule is applied. The final row presents the sum of all applied tests to identify the improvement over prior output and to select the rules with better performance.

Table 11 shows that rules 85, 105 and 150 obtained the best punctuation and that rules 75 and 180 are the best for the Approximate Entropy test. Rule 150 obtained a perfect pass in six of the NIST tests and rules 105 and 150 provided a full pass of the 148 sub-tests, failing in approximate entropy test. Moreover, Table 12 shows the *P-value* for raw and post-processed data using rules 105 and 150. Despite improving the past rate, *P-value* are not adequate in some tests indicating that the output numbers are at the edge of non-randomness and data bit-width might need to be reduced.

Figure 27 presents histograms for the output data evaluated in Table 12. Raw data in Fig. 27a) is redistributed applying post-processing with rules 85 (Fig. 27b)) and 150 (Fig. 27c)).

Figure 28. Entropy in bits for different output data-width.

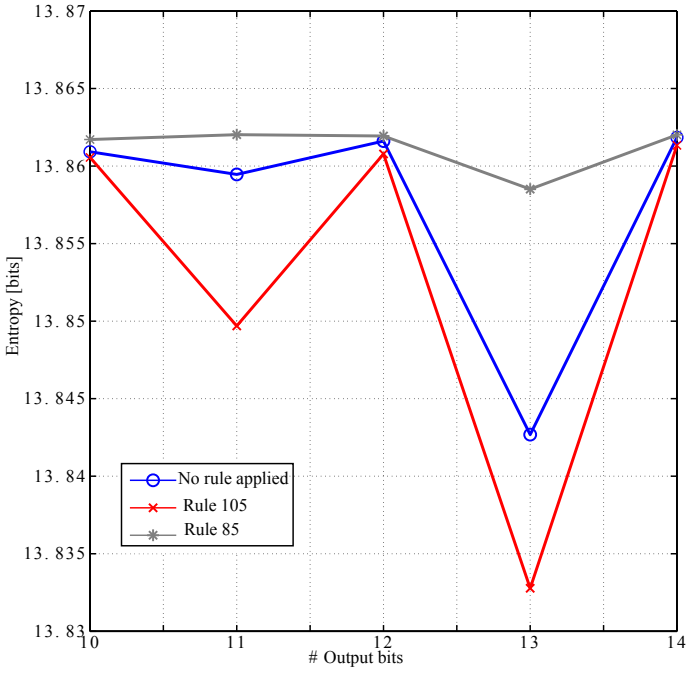


Table 13. Comparison with another FPGA implementation.

Specifications	[44]	[49]	This work
LUTs	308	272	409
FFs	380	807	36
$f_{real}[MHz]$	275	-	50

As the proposed post-processing does not reduce data-with the resulting effect is evidenced in the number of ones and zeros in the sequences. As consequence, an improvement in the data mean is achieve without significant changes in the standard deviation. The mean and standard deviation for data when no rule is applied correspond to 0.5048 and 0.2873*, respectively. For rule 85, the mean is 0.5016 and standard deviation is 0.2896 and for rule 105, the mean is 0.5004 and standard deviation is 0.2865.

Figure 28 shows extraction of entropy with words of 15 bits for a different number of output bits. In the case of the post-processed data with rule 105, the entropy value is always over the

*Mean and standard deviation for an ideal uniform distribution —ideal random sequence— correspond to 0.5 and 0.2886.

Figure 29. Entropy vs Number of outputs bits: Approximated Entropy from NIST test results for raw data, rules 85, 105 and 150.

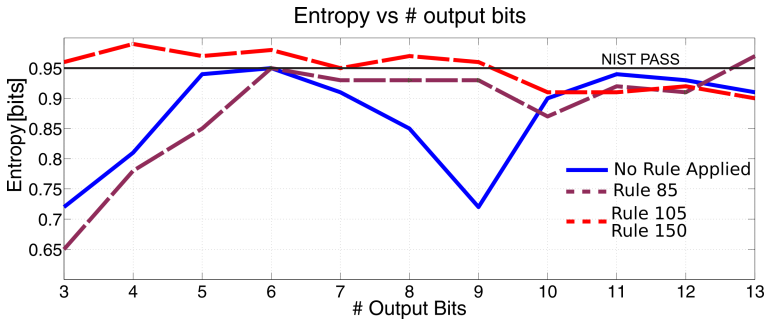
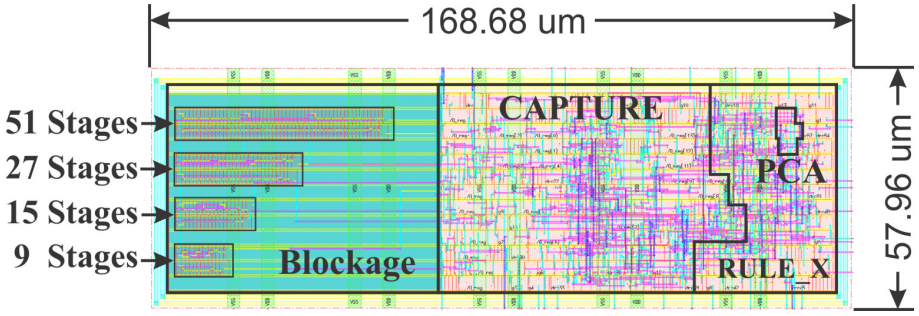


Figure 30. Standard cell based layout in 130nm, total area: 171μm$\times$59μm.



one for raw data whereas with rule 85 is always under the raw data, showing that rules should be carefully chosen to avoid deteriorating output statistics.

Results show post-processed data do not pass approximated entropy for rules 105 and 150 using the complete output data-width. Fig. 29 shows approximated entropy extraction for rules 85, 105 and 150. Rules 105 and 150 are the same and thus are super-placed on the red graphic. This figure shows that output data-widths less than or equal to nine allow for post-processing to enhance output data, passing approximated entropy test.

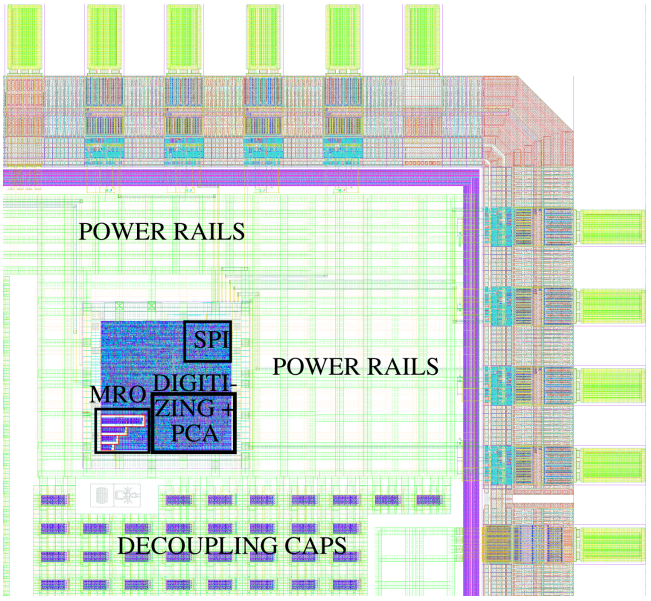
Table 13 compares this work to other post-processing stages for TRNGs implemented on FPGA. The proposed TRNG is compact, requiring 36 flip-flops (FFs) since the cellular automata array is completely combinational. Comparing this TRNG to the one proposed by Yang et al. [47], as shown in Table 14, one can readily see that the post-processing stage alleviates the need for output truncation, which is useful in enhancing short ROs, with low area and timing overhead. The proposed cellular automata array performs operations up to 1GHz which introduces no limitations to the TRNG core generation speed.

Table 14. Statistical comparison results with the work presented by yang et al. [47].

	21 stage RO* [47] 65nm CMOS	This work, 15 stage RO, FPGA
Output word data size [bits]	up to 7	up to 9
Passed NIST Tests	All	12
Capture freq. [MHz]	2.8	0.5

* A 15 stage TRNG is presented with no information about the whole NIST tests.

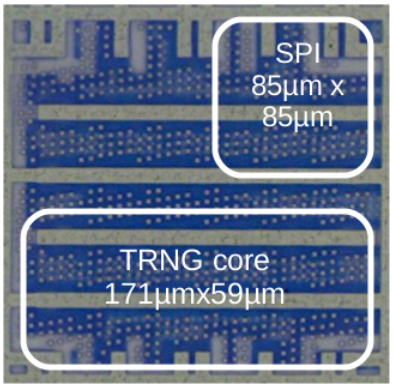
Figure 31. Layout for the signoff of the TRNG core including a digital monitor based SPI.



ASIC implementation The TRNG was fabricated in a 130nm CMOS standard technology with a final area of 170μm$\times$58μm. The power consumption of processing and capturing stage obtained from synthesis up to 1GHz is 2.66mW using 969 gates. Fig. 30 shows the four ROs located on the left side, after an user guided placement. The design includes additional blockages around ROs to avoid coupling between them, and protecting the circuits from external frequency noise sources.

Fig. 31 shows the final layout of the chip sent to fabrication. The layout includes decoupling capacitors, power rails, and I/O pads. The SPI monitoring and configuration interface occupies an area of 85μm$\times$85μm. The same SPI can capture the output bits with and without post-processing up to tens of MHz.

Figure 32. TRNG Fabricated chip.



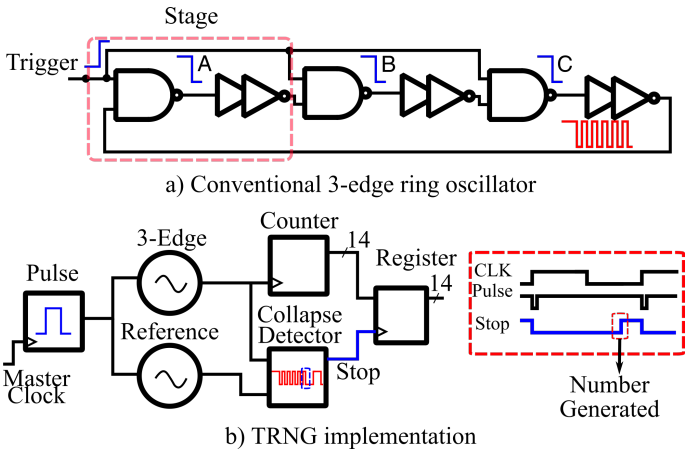
2.2.4. Chip measurement issues Fig. 32 shows the micro-photograph of the fabricated chip. Since the TRNG bases its operation in ring oscillators, a digital verification is not feasible. In an academic license for TSMC fabrication process, the foundry never provides layout and schematics of standard cells. Hence. a verification of this kind of circuit is a challenge. The TRNG was integrated into the same chip as the TPM prototype, that as explained in chapter 1 had integration problems. A short-circuit with the supply grid affected the configuration of some input pads. Besides, the multiplexing scheme of ring oscillators led to a frequency coupling that deteriorated the entropy quality.

2.2.5. Summary An all-digital TRNG, verified in FPGA and synthesized in a 130nm technology has been presented. The proposed TRNG passes 12 NIST tests using up to nine output bits and a post-processing stage based on cellular automata. The post-processing stage uses a programmable mask rule, enabling the testing for all possible rules. This programmable characteristic allows a designer to change rules during operation which can improve the output statistical behavior. The results show an implementation with high flexibility, improving throughput due to the use of short ROs. Furthermore, combinational operations in post-processing add a low timing overhead with an operating frequency up to 1Ghz.

Further transient noise simulations show a possible coupling of ROs in the entropy sources. Hence a redesign on entropy source can help with the reduction in output bias. Another improvement can be to redesign the post-processing stage applying a multi-stage approach, exploiting the cellular automation random behavior over time. Although this approach can improve the statistical

properties, a designer must address the trade-off between throughput and resources overhead.

Figure 33. Multi-mode ring based TRNG [47]. a) Conventional multi-mode ring oscillator, b)TRNG's extracting scheme and temporal behavior.



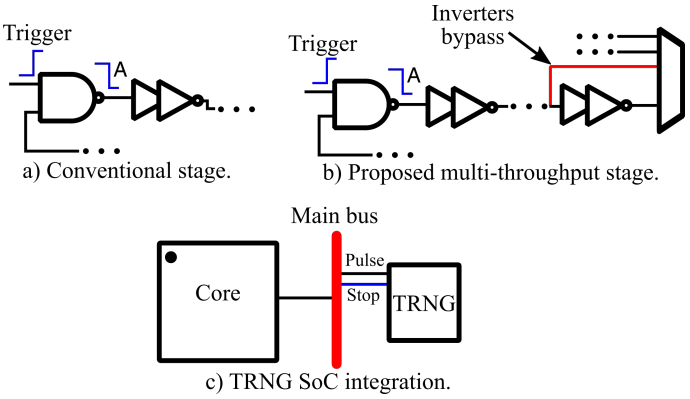
2.3. MULTI-THROUGHPUT TRNG ARCHITECTURE

This section presents a low-cost multi-throughput TRNG IP based on a variable-length multi-mode ring oscillator. The proposed TRNG implements a multi-throughput feature by bypassing inverter cells in the ring oscillator reducing the loop delay. This multi-throughput feature offers the advantage of high-performance or low-power operation when needed. These options make the proposed TRNG suitable for end-to-end encryption in highly restricted devices such as IoT sensor nodes. Measurement results show that the proposed TRNG passes NIST test for different throughput operation. The TRNG achieves an energy efficiency of 92pJ at 3.7Mbps, occupying 58μm$\times$150μm. In addition, this section presents a system technique to implement entropy enhanced TRNGs, using multiple entropy sources. An extraction system provides high-quality random numbers with a sampling method that takes one entropy output to sample other entropy sources. The system few resources, using low-cost TRNG IPs as entropy sources.

2.3.1. Proposed multi-throughput TRNG

In many TRNGs, a master clock samples their output so that if the master clock changes within the operation limits, the throughput will change as well [43, 50]. Yang et al. [47] report another TRNG that depends on a master clock, using a multi-mode ring oscillator as Fig. 33a) shows. A trigger signal injects three edges which propagate through the ring causing an oscillation frequency three times greater than nominal. The high-frequency oscillation collapses to nominal frequency due to jitter accumulation, making

Figure 34. Proposed multi-throughput inverter cell. a) Classical stage, b) proposed multi-throughput stage and c) proposed TRNG SoC integration.



that time unpredictable. Besides, the TRNG extracts random numbers with a cycle counter and a capture register that is triggered by a phase and frequency detector (PFD) as Fig. 33b) depicts. The PFD works as a collapse detector, identifying the time when the frequency of the multi-mode oscillator and a reference ring are similar.

The TRNG uses a master clock to trigger a pulse generator which determines the start time of active or generation phase. Every active phase starts with a positive clock edge and ends with a positive edge of *stop* signal as Fig. 33b) shows, registering the random number. Therefore, there is a high probability that the TRNG generates a number before the whole clock cycle ends.

One particular characteristic of Yang et al. [47] work is that the statistical properties of output bits depend on the number of inverter cells in the ring oscillators. With short rings, the collapsing time is faster, but TRNG outcomes require higher truncation to extract enough entropy. With large rings, TRNG outcomes need less truncation but with reduced throughput. Hence, there exists a trade-off between output bits quality and TRNG throughput.

In this dissertation, a prototype based on [47], using a multiplexed scheme [22] is proposed. Instead of using many TRNGs with rings of different inverter cells as proposed in [47], this work shows how multiplexing four pairs of ring oscillators of a different number of inverter cells in order to reuse the remaining logic, results in avoiding a large extra area penalty, with a resulting TRNG that may be tested for different entropy sources. However, using such a multiplexed scheme has revealed high-frequency interference due to the shortest ring, causing output data biasing.

For the above reasons, a modification is introduced in the multi-mode ring oscillator in order to achieve different operating frequencies and throughput with a single ring. The conventional

Figure 35. Extraction scheme to capture the highest entropy number between two multi-mode ring oscillator.

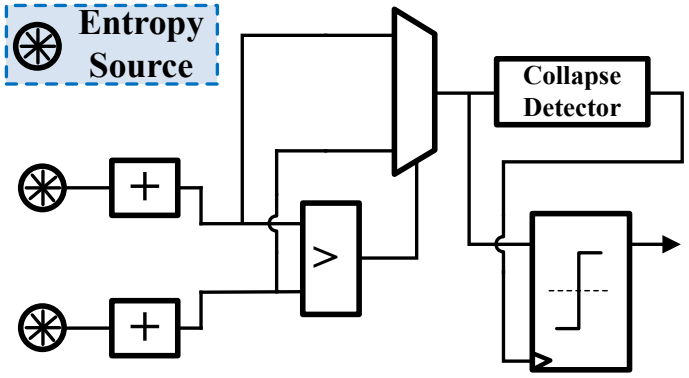
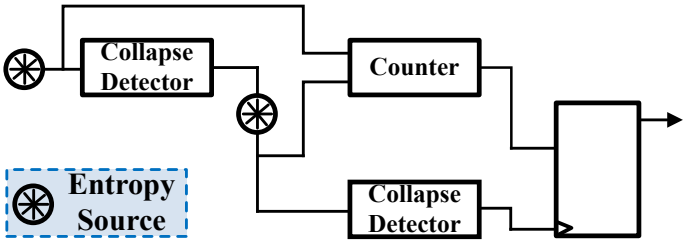


Figure 36. Scheme to enhance the entropy of a multi-mode ring oscillator by creating a dependent trigger from a second multi-mode ring oscillator.



multi-mode oscillator uses three similar stages composed by one NAND and an even number of inverters as Fig 34a) shows. A bypassing strategy is set to change the number of inverters per stage as Fig. 34b) depicts. The edge thus skips some inverter cells (as the red line in Fig. 34b) indicates) using a bypassing mux that reduces the effective delay.

The proposed TRNG is integrated along a 32bits RISC-V core in a SoC for IoT sensors nodes. Fig 34c) shows the TRNG connected to the core's main bus, through which the core can control the pulse generation and reads the *stop* signal. This *stop* signal acts as a flag so that the core can identify when the TRNG ends the active phase, considering that the number generation is asynchronous. In this way, the throughput is not limited to a master clock but only to the TRNG's collapsing time, saving additional clock circuitry.

The proposed TRNG has a total of 45 inverter cells per stage. The shortest ring uses a total of nine inverters including the NANDs. However, as previously reported in [47], a ring of 15 stages does not get enough entropy to pass all NIST test. We include more than one mux per stage to emulate a higher number of stages, increasing the collapsing event time. Therefore, the number

Figure 37. General scheme to sample one entropy source using a second entropy source. Choosing a pair of entropy sources can be done in a random way to obfuscate the design or hindering a possible attack.

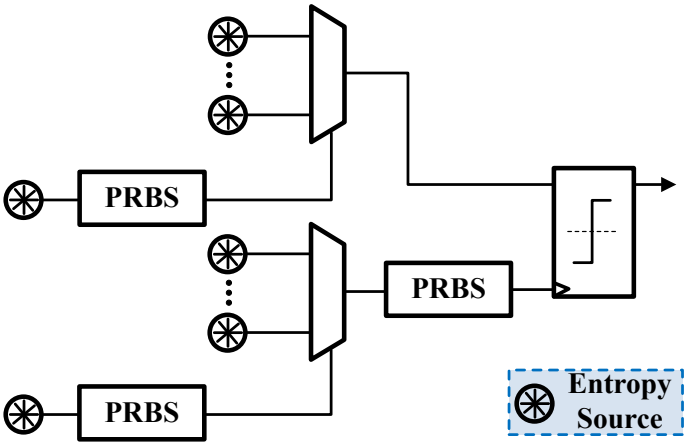
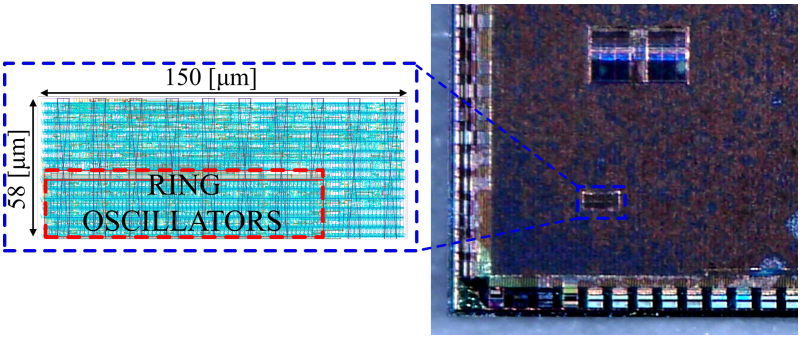


Figure 38. Micro-photograph of the taped-out multi-throughput TRNG.



of effective delays in the shortest ring is more than 15 including inverters, NANDs and muxes. Nevertheless, achieving high-quality output data requires to use a reduced set of output bits [47].

The next subsection discusses how the entropy of low-cost TRNGs can be boosted using at least two entropy sources. Theoretically, uniformly-distributed random numbers require at least two entropy sources as proposed by Chattopadhyay and Zuckerman [51]. They developed a theoretical method to extract the best approximation of uniform random bits. However, a circuit implementation has not been presented yet, since it would be too complex for restricted environments such as IoT devices. Therefore, alternative methods are here proposed in order to accomplish a circuital implementation using two or more entropy sources.

Figure 39. Histogram for a fixed throughput using different amount of output bits: a) 14 bits per sample, μ=0.4240 and σ=0.2681 b) 12 bits per sample, μ=0.5340 and σ=0.3016 and c) 10 bits per sample, μ=0.5044 and σ=0.2883. Figure depicts how the numbers distribution tends to be uniform with output bits reduction(ideal uniform distribution, μ=0.5 and σ=0.2886).

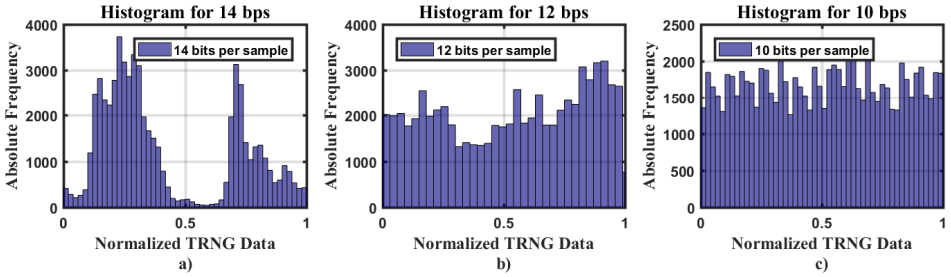
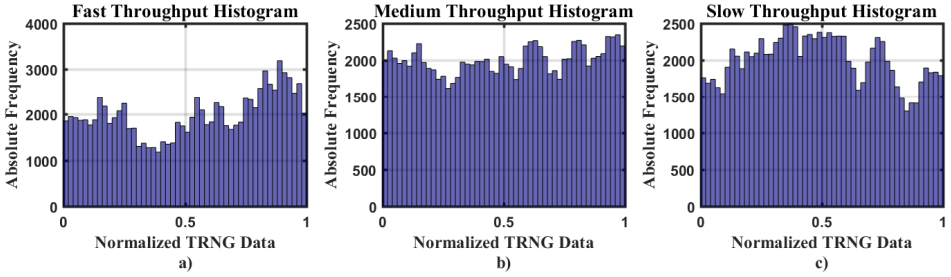


Figure 40. Histogram for three different throughputs using a fixed number of output bits: a) Fast throughput, μ=0.4240 and σ=0.2681 b) medium throughput, μ=0.5344 and σ=0.2858 and c) slow throughput, μ=0.4376 and σ=0.2951. Fast throughput histogram offers a more biased distribution.



2.3.2. Enhancing entropy quality with a two sources sample system

The use of two or more entropy sources has been considered before, but their use in randomness extractors remains in a theoretical or software level. This dissertation presents some approaches of how to build a TRNG using two or more independent entropy sources at hardware level.

Another important consideration in the implementation of a randomness extractor is the selection of entropy sources. For a high-quality source, a physical source of noise offers the best alternative. A physical source can be digital based or analog-based, but designers prefer digital entropy sources due to their low-complex implementation and inherent robustness against process and environmental variations [47]. Besides, if the digital entropy source is fully-synthesizable, the design would achieve portability to other technologies without negative performance impact.

Figure 35 shows a proposed scheme of a two-source based TRNG. In this work, the approach

proposed by Chattopadhyay and Zuckerman is followed, but boosting its deployability with fully-synthesized entropy sources as the offered in the subsection 2.3.1. Figure 35 depicts how to use the collapsing time of two independent multi-mode oscillators in order to boost output quality. One of the counters will go faster than the other, considering the collapse event of both rings is different. A comparator identifies which counter is faster, using it as the reference for a collapse event. A more extended collapse event provides with better entropy while the comparison provides a way to filter which ring produces a better quality number.

Figure 36 shows an alternative circuit. The boosting strategy is to trigger a multi-mode oscillator based on the collapsing event of a first oscillator. When the first oscillator collapses, a collapse detector triggers a second multi-mode ring. A counter provides the information on the collapsing time of the first ring oscillator plus the information of the collapsing time of the second ring oscillator. In this way, the effective collapsing time increases as the resulting values depends on both oscillators.

Fig. 37 shows a general scheme to boost entropy. If the entropy sources are of low complexity and portable, one can implement a TRNG with many independent entropy sources options. The main idea is to use one entropy source to sample a second one. A sample register captures numbers from the first entropy source based on a pseudo-random bit sequence (PRBS) as the clock, assuming the muxes already selected a pair of entropy sources. The second entropy source seeds the PRBS generator to provide the sampling clock. An additional alternative is to choose both entropy sources randomly. In the proposed scheme, one can select a pair of entropy sources at random using two other sources. In this way, the system avoids correlations between each couple of entropy sources while keeping the implementation low cost.

2.3.3. Measurement results

Figure 38 shows the micro-photograph of the fabricated multi-throughput TRNG in a 180nm technology. The TRNG is fully-synthesizable and occupies an area of 150μm$\times$ 58μm. Three different throughput configurations are measured, extraction a total of 100Mbits for each configuration in order to apply NIST tests [48].

It remains now to show that the proposed multi-throughput scheme does not affect the statistical behavior of the original design. Hence, the data distribution is studied using the histogram of a fixed throughput configuration. Figure 39 presents the histogram of captured data with a fixed throughput configuration and different output sizes. The aim here is to show that despite

Table 15. Measured NIST randomness test results for fast-throughput under different power supply values for the multi-throughput TRNG. Nominal is V_{DD} =1.8V

NIST Pub 800-22, rev 1a. 2010	Nominal		V_{DD} =2V		V_{DD} =1.6V	
Randomness Tests	*P-value*	RATE	*P-value*	RATE	*P-value*	RATE
Frequency	0.699313	99	0.350485	100	**FAIL**	**FAIL**
Block Frequency	0.122325	100	0.534146	100	0.911413	100
Cumulative Sums	0.883171	99	0.122325	100	**FAIL**	**FAIL**
Cumulative Sums	0.964295	99	0.008879	100	**FAIL**	**FAIL**
Runs	0.080519	98	0.419021	99	0.911413	100
Longest Run	0.055361	99	0.911413	99	0.066882	100
Rank	0.137282	100	0.574903	99	0.401199	100
FFT	0.437274	99	.334538	98	0.419021	100
Non-Overlapping Template	PASS*	PASS*	PASS*	PASS*	**FAIL**	**FAIL**
Overlapping Template	0.514124	99	0.075719	100	0.534146	100
Approximate Entropy	**FAIL**	**FAIL**	**FAIL**	**FAIL**	**FAIL**	**FAIL**
Universal	0.534146	99	0.017912	98	**FAIL**	**FAIL**
Random Excursions	PASS*	PASS*	PASS*	PASS*	PASS*	PASS*
Random Excursions Variant	PASS*	PASS*	PASS*	PASS*	PASS*	PASS*
Serial	**FAIL**	**FAIL**	**FAIL**	**FAIL**	**FAIL**	**FAIL**
Serial	0.017912	99	0.004301	99	0.000474	98
Linear Complexity	0.383827	97	0.275709	100	0.911413	99

*"PASS" means all sub tests pass minimum requirements

the multi-throughput modification, the use of fewer output bits actually improves the data statistic distribution. Figure 39a) gives the histogram, indicating a biased distribution and, as the number of used output bits decreases (Fig. 39b) 12 bits and Fig. 39c) 10 bits) the histograms tend to be uniform. An ideal standard uniform distribution presents a mean(μ) of 0.5 and standard deviation (σ) of 0.2886, and the data of the histogram shown in Fig. 39c) offers the closest approximation with μ=0.5044 σ=0.2883. Therefore, data distribution presents a behavior similar to the original multi-mode oscillator.

Fig. 40 shows histograms for a) the fastest throughput, b) a medium throughput and c) the slowest throughput, using ten outputs bits. As expected, the fastest throughput presents a more biased distribution, having the following order statistic parameters: μ=0.4240 σ=0.2681. Although, the medium throughput data offers better order statistic parameters than the slowest one, the proposed multi-throughput approach has a similar statistic behavior as the original proposal by Yang et al. [47].

The statistical properties of the proposed circuits, though, are not enough to pass NITS tests, even when a reduced amount of output bits seems to show a more uniform distribution in the histograms. This biased data could be introduced by the muxes, which produce an unbalanced load in some inverters. Hence, we applied a simple linear corrector based on an XOR function with a compression ratio of 1/2 [52], for three different throughput configurations, varying the power supply.

Table 16. Measured NIST randomness test results for medium-throughput under different power supply values for the multi-throughput TRNG. Nominal is V_{DD} =1.8V

NIST Pub 800-22, rev 1a. 2010	Nominal		V_{DD} =2V		V_{DD} =1.6V	
Randomness Tests	*P-value*	RATE	*P-value*	RATE	*P-value*	RATE
Frequency	0.350485	100	0.534146	99	0.171867	99
Block Frequency	0.350485	100	0.319084	100	0.534146	100
Cumulative Sums	0.122325	98	0.304126	96	0.637119	99
Cumulative Sums	0.017912	98	0.911413	99	0.991468	99
Runs	0.007694	96	0.834308	96	0.978072	97
Longest Run	0.657933	98	0.000145	99	0.026948	97
Rank	0.759756	100	0.401199	99	0.037566	98
FFT	0.867692	98	0.102526	100	0.181557	97
Non-Overlapping Template	PASS	PASS	**FAIL**	**FAIL**	PASS	PASS
Overlapping Template	0.455937	100	0.075719	100	0.834308	99
Approximate Entropy	0.017912	96	**FAIL**	**FAIL**	0.534146	99
Universal	0.534146	100	0.122325	100	**FAIL**	**FAIL**
Random Excursions	PASS	PASS	PASS	PASS	PASS	PASS
Random Excursions Variant	PASS	PASS	PASS	PASS	PASS	PASS
Serial	0.145326	99	0.002043	98	0.122325	98
Serial	0.224821	96	0.236810	100	0.759756	99
Linear Complexity	0.574903	97	0.574903	96	0.162606	99

One expected that for the fastest throughput configuration the statistical data distribution were good enough to pass all NIST tests. However, output data for the fastest throughput failed to pass at least two test for the different power supply values. Tables 15 summarizes the results for the fastest throughput configuration after applying the bias correction. Even in the nominal power supply, TRNG outcomes failed to pass the approximate entropy test and one of the serial tests.

Tables 16 and 17 summarize the NIST tests results for a medium throughput and the slowest throughput after applying the bias correction. The results show how the data for these two throughput configurations entirely pass all NIST tests in one of the operating conditions. These results agree with the histograms which show a significant biased distribution of data at the fastest throughput compared to the slowest and medium throughput. However, Data at medium throughput pass all NIST test in nominal operation and data at slowest throughput do not. On the contrary, outcomes at slowest throughput operating at V_{DD} =1.6V pass all NIST tests, and data at medium throughput do not.

Tables show that there exists a difference between the original behavior in the multi-mode oscillator compared to the proposed multi-mode oscillator. In the original scheme, a slow throughput always presents better statistical performance than a faster. In the proposed work, the applied tests show that in nominal operation a medium throughput offers better statistical qualities. Besides, power supply variations significantly affect the TRNG performance as in other regular ring oscillator based TRNG [8, 53] which was not reported by Yang et al. [47].

Table 17. Measured NIST randomness test results for low-throughput under different power supply values for the multi-throughput TRNG. Nominal is V_{DD} =1.8V

NIST Pub 800-22, rev 1a. 2010	Nominal		V_{DD} =2V		V_{DD} =1.6V	
Randomness Tests	*P-value*	RATE	*P-value*	RATE	*P-value*	RATE
Frequency	0.334538	100	0.319084	99	0.181557	99
Block Frequency	0.000199	100	0.004301	100	0.129620	100
Cumulative Sums	0.964295	99	0.867692	100	0.834308	100
Cumulative Sums	0.595549	99	0.010237	99	0.739918	99
Runs	0.181557	99	0.816537	99	0.005358	99
Longest Run	0.003447	96	0.978072	98	0.779188	99
Rank	0.637119	100	0.657933	100	0.224821	100
FFT	0.066882	99	0.213309	100	0.851383	98
Non-Overlapping Template	**FAIL**	**FAIL**	**FAIL**	**FAIL**	PASS	PASS
Overlapping Template	0.066882	100	0.834308	99	0.289667	99
Approximate Entropy	**FAIL**	**FAIL**	**FAIL**	**FAIL**	0.739918	100
Universal	**FAIL**	**FAIL**	**FAIL**	**FAIL**	0.534146	100
Random Excursions	PASS	PASS	PASS	PASS	PASS	PASS
Random Excursions Variant	PASS	PASS	PASS	PASS	PASS	PASS
Serial	0.004301	99	0.008879	99	0.437274	97
Serial	0.058984	99	0.304126	98	0.719747	100
Linear Complexity	0.935716	96	0.867692	98	0.983453	99

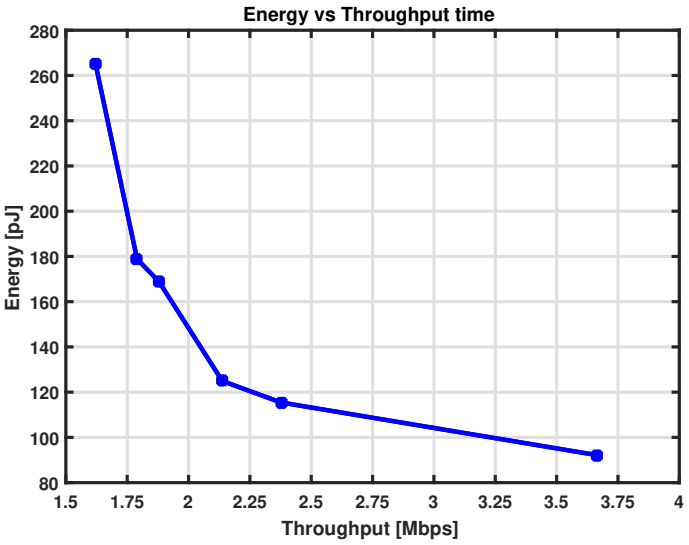
Table 18. Comparison of the proposed TRNG with the classical multi-mode based TRNG.

	ISSCC 2014 [47]	This work
Technology	65nm	180nm
Bit Rate [Mb/s]	2.8	**3.7**
NIST PASS	ALL	ALL
AREA[μm^2]	960	8700
Eficiency[nJ/bit]	0.057	**0.092**

The throughput is not a fixed value for a fixed configuration. This variation happens since the number generation relies on the non-deterministic behavior of the collapsing time, resulting in different generation times for each number. An average value is extracted for the three throughput configurations studied. 100Mbits were captured and the time measured with a real-time counter available in the SoC. Also, a similar set-up is used to estimate the average current during number generation. Figure 41 illustrates the energy consumption versus different TRNG throughputs. The TRNG achieves the best energy efficiency for a throughput of 3.7Mbits with an energy consumption of 92pJ.

Table 18 presents a comparison of the proposed multi-throughput multi-mode TRNG with the classical multi-mode based TRNG. The circuit here proposed is not limited by a reference clock, enabling optimization of the extraction rate. This work achieves a similar energy efficiency with a higher bit rate, validating the multi-throughput approach.

Figure 41. Energy consumption versus different throughput.



2.3.4. Summary This work demonstrates a multi-throughput TRNG using a bypassing approach to change the effective loop delay in a multi-mode oscillator. In the conventional multi-mode based approach, the authors used many TRNGs to test different throughputs. In contrast, the proposed TRNG offers a multi-throughput operation reusing the remaining logic which reduces area overhead. The TRNG was taped-out in a 180nm technology occupying 58μm$\times$150μm. The TRNG achieves a peak energy efficiency of 0.69nJ at 2.2Mbits. The TRNG is integrated along a 32 bits RISC-V based SoC exploiting the asynchronous nature of the proposed TRNG. A simple linear corrector with a compression ratio of 1/2 was introduced to reduce bias in the output data. However, the TRNG is sensitive to power supply variations, only passing all NIST tests in some specific operating conditions. This dissertation also proposes a method to boost entropy for low-cost TRNGs due to physical entropy sources do not provide the required quality. A minimum of two sources are required to extract enough entropy, but powerful extractors remain in software or theoretical level. Alternative methodologies of how to sample one entropy source with a second one in a fully-synthesizable way are here proposed. These methods offer an alternative to classical post-processing schemes with a low-cost implementation.

3. COUNTERING REVERSE ENGINEERING

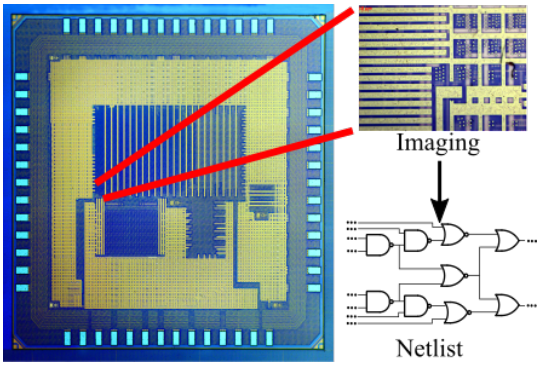
Intellectual property protection techniques face a challenging task in countering a physical attack by reverse engineering of an embedded integrated circuit. An attacker can extract sensitive information with image tools by processing microphotographs at low metal layers of delayered chips. This dissertation proposes a low performance and zero-area impact method to obfuscate circuits by using a current digital design flow with a layout standard cell generator that produces different physical versions of the same logic cell. Results indicate that timing and power overhead, introduced by the obfuscation method, can be mitigated. After applying the method to a set of benchmark circuits and a 32-bit RISC based microprocessor, results show a 40%-to-50% average obfuscation with zero area penalty and less than 2% timing and power penalty for system level blocks. Considering that most attacks direct reverse engineering to key cryptographic functions, experimental obfuscation results indicate a timing penalty of 4% with a strong obfuscation level for a synthesized key establishment core.

Although results are good and the software used for verification works well under a typical reverse engineering task, the results evidence some limitations in the software. The testing tasks that were here performed show how the software can identify no-obfuscated cells and fail when trying to identify the obfuscated layouts with the proposed method. In some cases, though, the software tends to inadequately resolve the obfuscated layouts when for example, identifies NAND cells as NOR cells. These results might indicate some weakness in the proposed algorithm and an exhaustive verification with stronger processing tools might be required. However, the use of third-party tools is desirable for a fair evaluation, limiting this verification to software availability.

3.1. INTRODUCTION

The pervasiveness of consumer electronic (CE) devices has brought to existence different challenges for intellectual property (IP) protection [54, 55]. These challenges increase when time-to-market pressure forces the reuse of IP cores, leading to additional vulnerabilities [56–58]. Among different threats, reverse engineering attacks have become popular, enabling extraction of functionality or sensitive information. An attacker may use of techniques such as depackag-

Figure 42. Process of reverse engineering integrated circuits. A delayered chip is analyzed using an image tool that is trained with some common patterns. Final netlist extraction is achieved by identifying existing cells and connections.



ing/delayering, high-resolution imaging, probing, and side-channel examination to extract useful information from ICs [11]. The extracted information can be used in different ways, such as discovering IP violations or faulty products, but may also serv for cloning/copying IP or exploiting vulnerabilities in order to get restricted accessed or elaborate sophisticated attacks [12].

Figure 42 shows a typical image processing-based reverse engineering starting by decapping and delayering a chip [59]. An image processing tool —capable of rebuilding a gate netlist by extracting routing layers—is trained with known layout patterns of the standard cells to be recognized. At the end of a repetitive and time-consuming process, the recognized cells along with the extracted routing information enable the software to rebuild the whole netlist. The described process proves how difficult it is for o a designer to protect an IP when an attacker has physical access to the chip.

Developing a successful strategy to protect IP blocks is a concern for the semiconductor industry. Hardware obfuscation [12] is an alternative with two main approaches that may offer the necessary protection: structural obfuscation and physical obfuscation. The purpose of a structural approach is to hide the true functionality of the device-under-attack through techniques such as the insertion of key-gates [13]. The application of key-gates enables the possibility of creating unique secret keys such that the logic function is corrupted if the correct key is not applied. Although logic level obfuscation by using key-gates is defeated by using test patterns and observing their output in order to decipher the keys [12], its strength against attackers is considered larger than current physical level camouflage techniques.

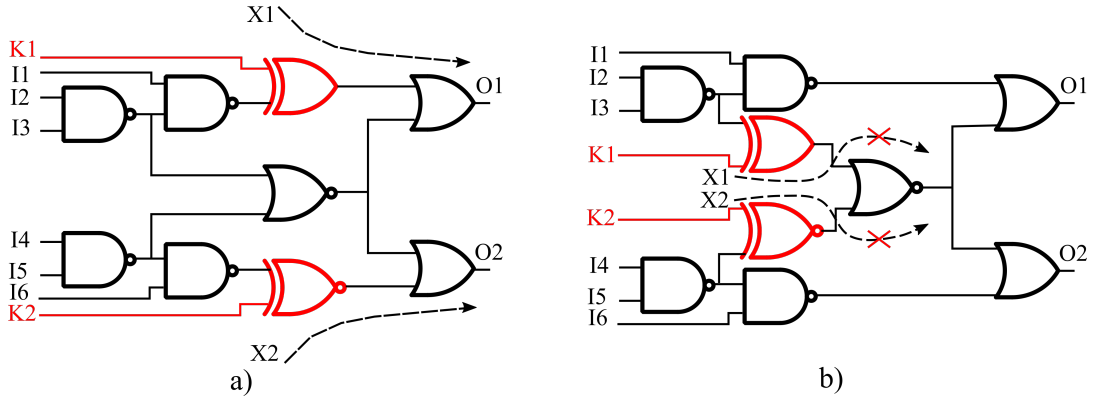
Traditional approaches to physical obfuscation exploits camouflaging by designing look-alike

gates with metal and contact dummies [14]. Using dummy structures implies area and timing overheads that might impede the adoption of camouflaging in high-performance applications. Other methods of look-alike cells exploit different voltage thresholds to define gates with identical layouts [11]. However, composing gates with multiple voltage thresholds translates to larger areas and higher performance impact due to process variations.

This work develops a physical obfuscation technique based on the usage of on-the-fly cell generation during synthesis. Compared to current reported approaches, the proposed technique incurs in a zero-area penalty with a small performance impact. The proposed methodology is transparent to commercial flows used in high-performance nodes, where it can be easily implemented considering that dynamic cell generation during synthesis is a common approach.

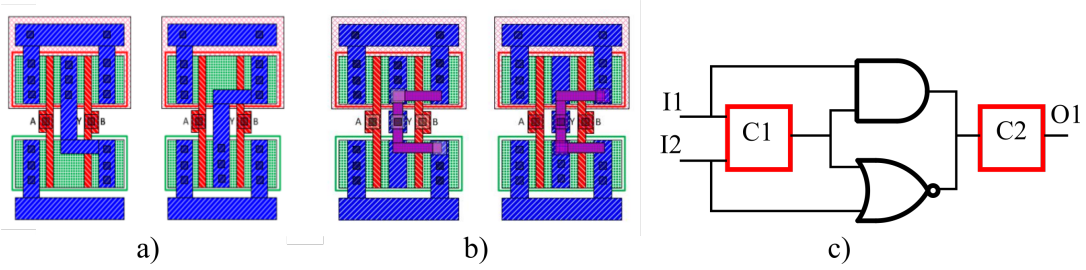
3.2. RELATED WORK

Figure 43. Structural obfuscation using key patterns [13]. a) Mutable keys, determined by input patterns. b) Non-mutable keys, strong logic obfuscation.



As mentioned by [57], CE literature focuses on characteristics such as energy efficiency, high performance, and area efficiency. Only a few authors, Sang-ho *et al.* [60] discuss the importance of security characteristics, but recently the interest in IP protection has increased due to the feasibility of implementing different reverse engineering attacks [11, 12, 14]. This section advances some related works in the consumer electronics area and two main techniques considered to obfuscate circuits. The first one involves corrupting digital behavior through a fixed key bitstream commonly known as structural obfuscation [13, 61]. The second one is a physical approach where multiple cells have the same layout, or dummy routes hide the actual connections [11, 62].

Figure 44. Layout level camouflage using look-alike gates [14]. Two different logic cells have similar geometric shapes.



3.2.1. Consumer electronics related work Reusable IP cores are the main concern in terms of IP protection in CE devices, since reduced time-to-market scenarios have led to the reuse of IPs such as DSP, MPEG, and JPEG [56–58]. One approach used to protect IPs is watermarking, which includes a signature in the IP in order to identify ownership. Sengupta *et al.* [56] proposed a low-cost technique in high-level synthesis to include a watermark in reusable IPs. Despite reducing implementation cost, watermark does not necessarily avoid copying or cloning functionality, stressing the need to find alternative techniques.

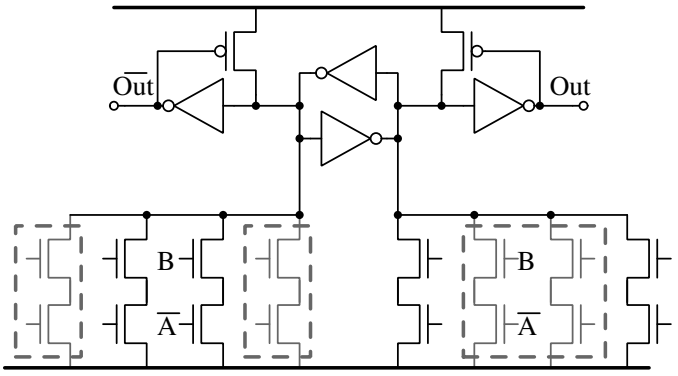
Sengupta *et al.* [57] also propose a structural obfuscation using algorithm transformation hiding the functionality during synthesis, along with a method to optimize specifications such as area and power consumption. Optimization led to increase proposal complexity and cannot totally avoid area overhead over the original design. Another approach is provided by Sengupta *et al.* [58], using locking gates or key gates. This approach entails blocking functionality with a secret key, only available to the final user. Sengupta recommends IP functional locking blocks (ILBs) attaining the same functionality with different secret keys. An extra optimization reduces the impact of including locking gates not only in terms of area but timing and power consumption. Despite reported approaches that optimize the number of resources used, performance penalties, in term of area, can not be avoided. Subsections 3.2.2 and 3.2.3 describe structural obfuscation as well as physical obfuscation in detail as other alternatives for IP protection.

3.2.2. Structural obfuscation Structural obfuscation uses keys to prevent leakage of information in untrusted phases of the design flow. Only a valid key can activate correct functionality in the post-fabrication phase. In this way, an attacker is not able to extract the actual functionally without knowing the key. However, this key can be revealed statistically by applying

some input patterns that propagate key validations to the circuit's output [12]. Figure 43a) [13] shows an example of structural obfuscation where key inputs and outputs correspond to I[1-6], K[1,2] and O[1,2]. A key {K1,K2}=01 activates functionality, but any other combination will corrupt all outputs. Although functionality can be hidden, a binary input pattern 100000 can reveal the key. This input pattern puts a zero logic and one logic in XOR and XNOR gates respectively, making the value of keys available in OR gates at the output. The common input of OR gates results in a zero logic, hence X1 and X2 signals, that have a correct key value, move to the output.

Rajendran *et al.* consider how to prevent key leakage due to redundant or isolated keys [13]. Their method proposes a strong logic obfuscation by inserting key gates that create complex interferences between all of the keys as Fig 43b) shows. An interference between two keys means that an attacker would need to control one key to propagate the other key. However, an attacker would never have access to any of the keys, if an anti-tamper memory stored the keys. Therefore, an attacker would need to use a brute force attack to decipher the functionally. Even so—given enough time and resources—an adversary could successfully perform reverse engineering [12]. Furthermore, Increasing even more the required time for a brute force attack implies area and power penalties that might be prohibited for high restricted designs.

Figure 45. Threshold voltage defined (TVD) logic cell [11]. Multiple differential cells are designed to maintain the same connections using transistors with different threshold voltages. Different logic functions are implemented with the same layout when transistors' placement is exchanged.



3.2.3. Physical obfuscation Among alternative physical approaches, the look-alike approach offers a feasible strategy avoiding expensive modifications in the fabrication process.

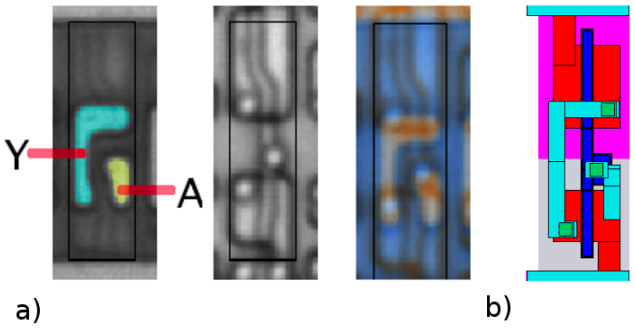
Cocchi *et al.* [62] propose different methodologies to design solutions as shown in Fig. 44. Figure 44a) shows typical NAND and NOR layouts where an image processing tool can differentiate both cells. However, Fig. 44b) shows a layout of the two cells using the look-alike method where both cells look similar for an image processing tool. Camouflaged standard cells with a look-alike method can be differentiated by using certain input patterns in two different ICs, and comparing their outputs [14]. Rajendran *et al.* [14] offer a solution to increase difficulty for an attacker. This method introduces look-alike cells to interfere with each other as shown in Fig. 44c), causing ambiguity when someone is trying to extract functionality. A general drawback of the look-like method is the extra routing which imposes performance penalties.

Erbagci *et al.* [11] propose another alternative to achieve look-alike layouts by using a threshold voltage defined (TVD) logic family. The author designs different logic cells with identical schematic by using different threshold transistors in a differential cell. Fig. 45 shows an example of a differential cell that implements an XOR/XNOR function. The cell uses low threshold (LVT) transistors (shown in gray) and high threshold (HVT) transistors where each branch and its complement have the same inputs. This connection causes two complementary branches to activate at the same time, but due to the threshold difference, a differential behavior occurs with zero static consumption at steady state. The differential cell can implement different logic functions by changing the position of LVT and HVT transistors and maintaining the same connections. As a consequence, all the layouts of the logic family look the same. Although similar layouts are desired to counter an image processing based attack, it is not possible for a standard cell-based design flow to include different thresholds transistor in the same standard cell, increasing design complexity and reducing scalability. Furthermore, considering design for manufacturing guidelines, combining multiple threshold devices close together increases the chance of larger process variations that are usually not considered in device models.

3.3. PROPOSED CAMOUFLAGING METHOD

IP protection techniques generally imply area or power overhead. These additional resources provide proper security levels without altering functionality. Achieving functionality and minimizing resources usage are goals that enhance deployability of any obfuscation technique. This section focuses on presenting the proposed solution, quantifying possible overheads and discussing how

Figure 46. a) Different metal1 microphotographs of an INVD1 from a 9-track standard library, b) and its layout.



to mitigate them.

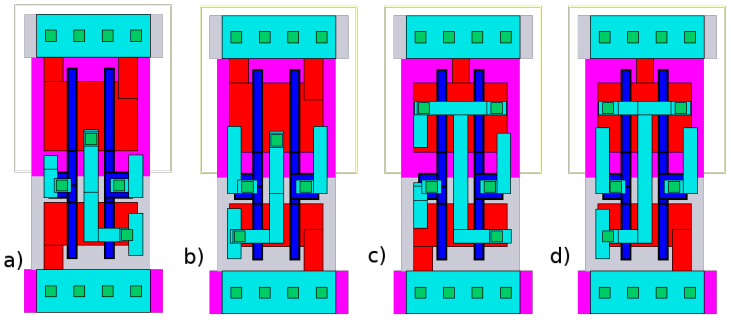
3.3.1. Standard cell camouflage method There are mainly two digital flows used in the industry. The most common one is to use standard cell libraries with regular gate sizes. For instance, a regular standard library would have inverters sized as multiples of an inverter capable to drive a load of fan-out of four (FO4), usually called INVD1. Inverters are usually sized twice to eight times the size of an INVD1. These libraries are generated and tested by foundries and delivered to foundry users. The other flow, that is uncommon to small and mid-sized companies, is the flow with dynamic libraries. A library is populated dynamically during synthesis to reduce power and improve timing. For instance, a critical path might require an inverter sized 1.6X and, instead of using the common 2X inverter from a fixed library (INVD2), an on-fly library generator creates the customized 1.6X inverter. This flow generates gates on-fly synthesis, requiring standard cell layout generators which are usually restricted to companies focused on high volume consumer electronics.

A flow capable of generating on-fly gates enables the generation of non-optimal cells with different placement and routing, compared to an optimal cell which has been drawn for optimal timing and power performance. By exploiting the capability to get additional geometries for the same cell, we have developed a method to obfuscate digital blocks by the use of multiple cells for the same function. An imaging tool, such as Chipworks' ICWorks and Degate [12], trained with a database of regular optimal cells would not be able to identify gates that have a non-optimal

geometry shape and therefore a full netlist cannot be extracted, disrupting the attack.

Imaging tools are fed with known gate geometries to train them to detect regular gates [59]. Figure 46 shows an example of well-known layout patterns of an INVD1. Different microphotographs of fabricated inverters in Fig 46a) show similarities to a typical INVD1 layout in Fig. 46b). Shape similarities are easily identified on metal 1 and polysilicon layers even without a processing tool, making it suitable for pre-trained software to extract useful information to rebuild a complete digital block. However, the use of gates with different geometry shapes on metal 1 and polysilicon layers could fool the imaging tool since an INVD1 usually has an optimal well-known shape.

Figure 47. Four different layout representations of a NAND2D1 layout.



Different layout geometries can be requested from an automatic cell generation tool considering that many non-optimal layouts might be generated before finding the optimal case. For instance, Fig. 47 shows four different NAND2D1 flavors where Fig. 47a) corresponds to the optimal case. Also, Layouts shown in Fig. 47b), Fig. 47c) and Fig. 47d) are alternative geometries, some of them quite far from the optimal placement, but with similar routing accessibility to their inputs.

3.3.2. Standard cells characterization and comparison

In order to verify the impact on non-optimal gates, a characterization of generated cells is performed for different environmental and process conditions, using 9-tracks standard cells. The characterization extracts power and timing specifications for typical and worst cases. Finally, the area for all different shapes of the same standard cell remains constant, resulting in no penalties for this specification.

Table 19 depicts performance extraction for the typical case and worst case scenarios. In the worst case, timing performance overhead rises up to a 13% for NAND cell non-optimal alterna-

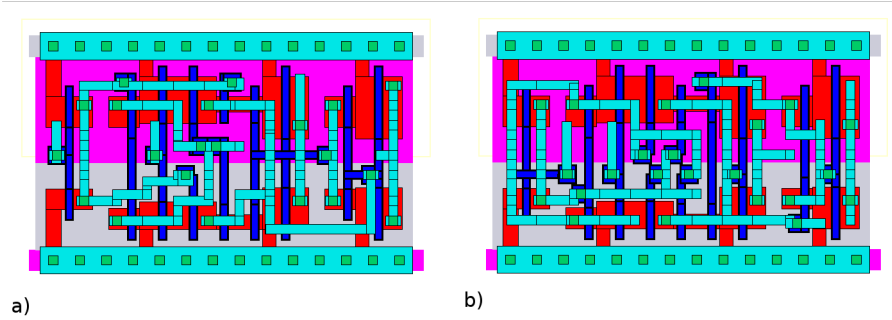
tives, whereas timing penalties for inverter variants remain under 5%. Non-optimal cells are also up to 16% more power hungry. Such power and timing overheads in non-optimal cells are due to additional routing usage that increases inner cell capacitances, degrading performance. Besides, using non-optimal placement can result in increased diffusion capacitances. Although this table presents a reduced number of gates, such as inverter, NOR and NAND gates for one column size, a larger cell library has been used for complex circuits demonstrating experimentally that these three cells are a representative sample of the large set. Average performance indicates that a large timing penalty occurs if non-optimal cells appear inside a critical path, thus limiting their usage to non-critical paths.

Table 19 summarizes performance as well for a latch LND1. Considering performance overhead of latches, results show that penalties are smaller than combinational cells. Low overhead can be achieved due to additional margin to create different layouts from basic combinational cells considering a larger cell width. Consequently, few changes produce different geometries without requiring large additional routes. As an example, Fig. 48 shows two different geometric representations for a latch where, maintaining the same placement, the routing significantly changes.

Table 19. Basic gates characterization at typical case (TC) and at low voltage supply, slow-slow process corner, and 125C (WC). This characterization is presented as the ratio between the specifications of cells with non-optimal layouts and cells with the optimal ones.

	INVD1							
	A		B		C		D	
CORNER	TC	WC	TC	WC	TC	WC	TC	WC
POWER	1	1	1.04	1.03	1.03	1.03	1.09	1.08
TIMING	1	1	1.01	1.02	1.01	1.02	1.04	1.05
	NAND2D1							
	A		B		C		D	
POWER	1	1	1	1.005	1.15	1.15	1.16	1.16
TIMING	1	1	1	1.004	1.11	1.12	1.12	1.13
	NOR2D1							
SPEC	A		B		C		D	
POWER	1	1	0.99	0.99	1.14	1.14	1.15	1.15
TIMING	1	1	0.99	0.99	1.10	1.12	1.11	1.12
	LND1							
SPEC	A		B		C		D	
POWER	1	1	1.072	1	1.053	1.073	1.019	1.009
TIMING	1	1	0.997	1.004	1.038	1.038	1.012	1.012

Figure 48. Two different geometric representations of a LND1 layout with same placement and different routing.



3.3.3. Obfuscation and standard cell generation algorithms

The algorithm 1 summarizes the procedure to obfuscate a circuit including many layout cells. The algorithm requires the original standard cell library (LIB), the number of desired obfuscation levels (OBLevels), and hardware description of a circuit to be obfuscated. First, the circuit is synthesized using the standard library to obtain an initial netlist and the corresponding critical path. Then, the library is populated with on-the-fly generated cells aiming at the obfuscation of the previous netlist. The algorithm inserts new cells, blocking the ones in the critical path, resulting in a netlist that includes two layout versions per logic gate. If the new circuit presents the same critical path, a first obfuscation level is achieved(Lvl. 1), avoiding cell restoring. The algorithm saves the netlist for the next iteration and then, generates new cells to increase the obfuscation level(Lvl. 2). The critical path is always verified, performing a restoration of the original cells in case of finding a timing penalty. New cells are generated and inserted repeating the process until the desired obfuscation level is achieved. Although the critical path is always verified, the algorithm can incur in timing penalties, hence, the cell restoration procedure includes a pre-defined value of tolerance to avoid permanent loops. This might alter the critical path in the final netlist if the counter reaches greatest user-defined tolerance.

The algorithm prevents reverse engineering processes by keeping a unique generated library per run. This library includes a set of random layouts per standard cell according to “OBLevels”. A designer can keep or destroy this library without affecting the repeatability of any obfuscation process. Also, the number of obfuscation levels “OBLevels” does not affect final circuit operation. In a traditional reverse engineering process, the known information is generally related to the original standard cell library that a foundry provides and an approximate digital behavior that the

Algorithm 1 Obfuscation algorithm

```
1: procedure StdCellOb(LIB,OBLevels,HDL)
2:     [netlist, critPath] ← RTL(LIB, HDL)
3:     save(netlist, "NORMAL");
4:     for each integer i in OBLevels do
5:         netlist_i ← netlist
6:         for each object cell in LIB.cells do                ▷ Generate a set of stdcells
7:             StdCell(cell.netlist, cell.info);
8:         end for
9:         for each object inst in netlist_i.insts do          ▷ Obfuscate using the new set of stdcells
10:            cell ← LIB.FindCell(inst.name_inst)
11:            if inst.name_inst = cell.geom.first.name then
12:                inst.name_inst ← RandomFrom(cell.geom).name;
13:            end if
14:        end for
15:        [netlist_new, newCritPath] ← RTL(LIB, netlist_i)
16:        count ← 0
17:        while newCritPath ≠ critPathANDcount ≤ tolerance do
18:            for each object inst in newCritPath.insts do    ▷ Restore cells
19:                cell ← LIB.FindCell(inst.name_inst)
20:                ninst ← netlist_i.FindInst(inst.name)
21:                ninst.name_inst ← cell.geom.first.name;
22:            end for
23:            [netlist_new, newCritPath] ← RTL(LIB, netlist_i)
24:            count ← count + 1
25:        end while
26:        name ← format("OBS[1]", i)
27:        save(netlist_new, name)
28:        netlist ← netlist_new
29:        critPath ← newCritPath
30:    end for
31: end procedure
```

circuit performs. If an attacker tries to perform reverse engineering with the original standard cell library, random layouts prevent recognition of different cells across the image, thwarting the RTL extraction.

Algorithm 2 describes an extra procedure for generating obfuscated layouts for standard cells. The algorithm requires the circuit netlist (Netlist), and a master managed cell info, which carries geometries and previous solutions information (CellInfo). First, placement constraints generate satisfiability boolean problems (SAT) in a conjunctive normal form based as proposed by [63]. A SAT solver is used in procedure SATSolve, which determines placement feasibility with the current configuration. If placement is possible, the algorithm tries to find a feasible routing solution which is also based on an SAT problem. If the algorithm finds a routing solution for the corresponding placement, a new layout geometry in standard cell format can be generated. Characterization

Algorithm 2 Standard cell generation algorithm

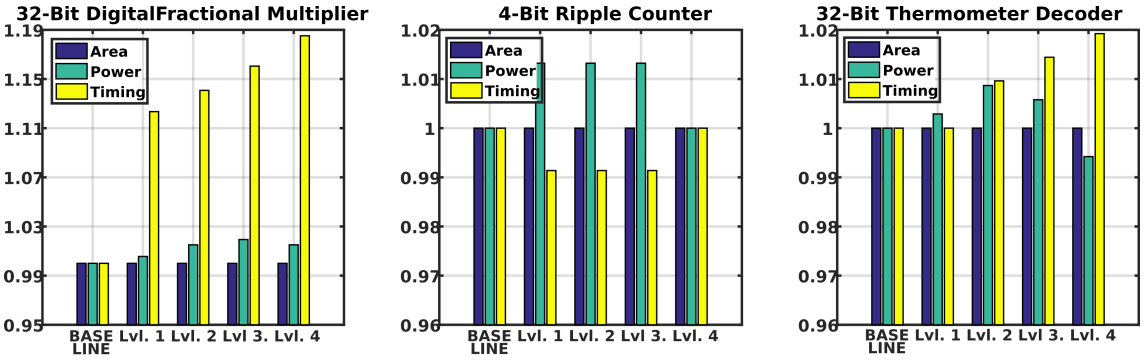
```
1: procedure StdCell(Netlist,CellInfo)
2:     found ← false
3:     pAnalysis ← AnalysisPlace(Netlist)
4:     BlockPlace(pAnalysis, CellInfo.pSolutions)
5:     IncrementColumns(pAnalysis, CellInfo.col)
6:     while !found do
7:         satProb ← PlaceConstr(pAnalysis)
8:         [pSolved, pSolution] ← SATSolve(satProb)
9:         if pSolved then
10:            rAnalysis ← AnalysisRoute(pSolution)
11:            satProb ← RouteConstr(rAnalysis)
12:            [rSolved, rSolution] ← SATSolve(satProblem)
13:            if rSolved then
14:                found ← true
15:                Push(CellInfo.geom)
16:                Push(CellInfo.char)
17:                DoGeometry(CellInfo.geom.last,
18:                pSolution, rSolution)
19:                CellInfo.char.last ← Characterize(CellInfo.geom.last)
20:            end if
21:            Append(CellInfo.pSolutions ← pSolution)
22:            BlockPlace(pAnalysis, pSolution)
23:        else
24:            IncrementColumns(pAnalysis, 1)
25:            EraseBlocking(pAnalysis)
26:            CellInfo.col ← pAnalysis.col
27:            Clear(CellInfo.pSolutions)
28:        end if
29:    end while
30: end procedure
```

is performed and stored after new geometry generation. Whether routing is solvable or not, the algorithm blocks current placement solution guarantying that a placement will not be used more than once. If SAT constraints in placement fail, the algorithm increases transistor spacing in columns, erasing any previous blocking. Optimizations in both placement and routing are done using pseudo-boolean SAT (PB-SAT) constraints, all embedded into SATSolve procedures.

3.4. RESULTS AND DISCUSSION

This section carries out an analysis of performance specifications for different obfuscated circuits. We extract timing and power specifications for different circuits where the proposed design flow is applied. Performance summaries are presented in order to study how cells with non-optimal layouts penalize a camouflaged circuit and how efficient a mitigation strategy might be. The final

Figure 49. Performance summary of camouflaged circuits with the proposed technique. These measurements are given by per-unit comparison with the initial synthesis performance. Specifications are presented as follows: blue bars indicate area, red bars indicate power, and orange bars indicate timing. a) Results on a 32-bit digital fractional multiplier. b) 4-bit ripple counter. c) Thermometer decoder.



a) b) c)

area is also extracted but as the camouflaged standard cells were designed to have the same size, the performance penalty is always void.

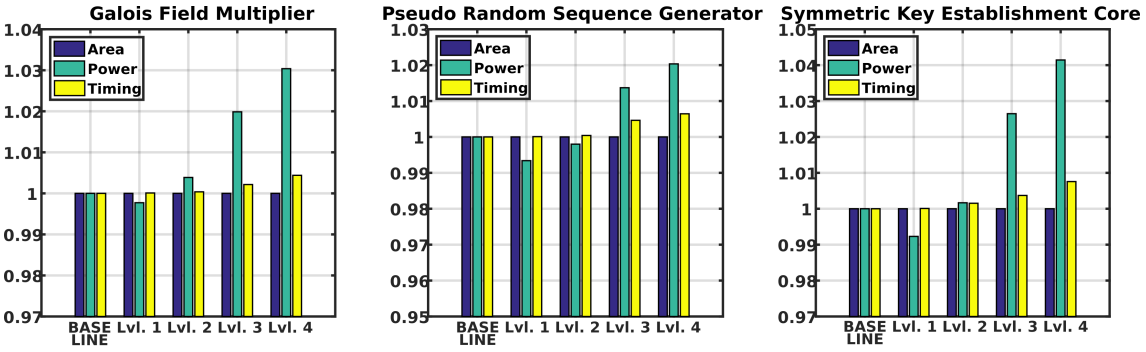
3.4.1. Low-area block-level benchmarks

Fig. 49 shows results of three synthesized circuits, a fractional multiplier, a ripple counter and a thermometer decoder from an ISCAS synthesis benchmark [64]. Fig. 49a) shows the performance of the synthesized multiplier including the camouflaged cells with worst timing. The left-most bars state the results of the synthesized multiplier using the foundry provided standard library. In the first iteration (the group of bars number 1), only one camouflaged cell is randomly included, alternating its use with the foundry-provided one. Synthesis results show a timing overhead although the critical path is not modified. Possible penalizations appear due to fanout changes, even if gates in critical path remain the same. Further iterations are required to minimize this effect.

Fig. 49b) shows the results for the ripple counter. For this circuit, the algorithm runs with an increased number of iterations in order to optimize timing performance. Results from a set of solutions show enhanced timing performance or null overhead when proper cells are generated. In contrast, power consumption increases in three of the four cases.

Fig. 49c) shows results associated with the thermometer decoder synthesis. The algorithm achieves less than 2% timing penalization for all cases and a zero penalty for the first obfuscation,

Figure 50. Performance summary of some cryptography related circuits when obfuscation is applied. These measurements are given by per-unit comparison with the initial synthesis performance. Specifications are presented as follows: blue bars indicate area, red bars indicate power, and orange bars indicate timing. a) Results on a galois field multiplier (GF8). b) Pseudo random sequence generator (PRBS). c) Key establishment core (KEC).



a) b) c)

considering large tolerance on critical paths. Besides, the algorithm achieves low timing penalty with no penalization in power performance for the last level of obfuscation.

3.4.2. Obfuscation of some cryptographic related circuits.

Cryptographic functions are one of the most critical security implementations. These functions process sensitive data from top-level executions. Some cryptographic algorithm examples are involved, such as key handshaking and encryption. Reverse engineering focuses on extracting digital behavior of data ciphering in order to have an algorithmic approximation of these functions.

Although cryptographic algorithms are generally well known, many different implementations are possible depending on what a designer wants to optimize. These multiple options should be carefully designed to avoid implementation vulnerabilities. However, an attacker can unveil backdoors [65] through a reverse engineering process enabling cryptographic function manipulation or even possibilities of adding/disabling functionality [12].

We present three circuits involving cryptographic functions as examples. The first circuit is an 8-bit Galois field multiplier (GF8), that can be used in Rjindael and advanced encryption standard (AES) scheduling schemes [66,67]. The second one is a pseudo-random binary sequence generator (PRBS which generates an Nth-order cyclic random bit stream) which is based on a variable length linear feedback shift register(LFSR) [21]. PRBSs can be used as post-processing stage in

true random number generators (TRNG) [68] or also as a source of cryptographic keys by using a random seed for low level security protocols. Finally, the third circuit is a lightweight symmetric key establishment core. This core provides synchronization of private random key across a public channel [21].

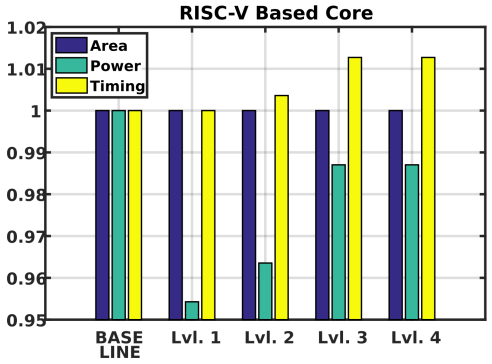
Considering AES algorithm, best threatens still remain computationally infeasible. However, if an attacker finds a backdoor in the implementation, the attacker can exploit the behavior or inject backdoors at foundry-level. A similar situation is present in a PRBS used as post-processing stage. Even if the entropy source is well distributed, a PRBS manipulation could result in a bias of output data. A similar scenario can apply to key establishment core. Consequently, an obfuscation in a physical level adds a level of security by hardening the process of knowing the actual implementation.

The establishment core is based on synchronizing property of neural networks learning from each other. The block has a serial data flow and weights are loaded in parallel at random using a PRBS generator. After initialization, a serial input is applied and one output bit is computed based on parity of hidden units. When two neural networks have their corresponding outputs, a comparison is performed, updating weights only if outputs agree. Synchronization is achieved in time establishing a common key, which means weights are aligned or equal. An advantage of this synchronization feature is that enables a fast update of the common key when the security system requires it.

Fig. 50 shows synthesized results for cryptographic circuits for four different versions of obfuscation where circuits sequential behavior enables the use of obfuscated latches. Fig 50a), Fig. 50b) and Fig. 50c) show results for Galois field multiplier, PRBS, and key establishment core respectively. For all circuits, the first obfuscated version shows a reduction in power consumption but it is not consistent in remaining versions.

Despite increasing obfuscation level, timing overhead is maintained as low as 1% for all cases. In contrast, power overhead increases as obfuscation level do, with 4% in the worst case. One interesting result is that power overhead for the PRBS can also be mitigated not only in the first obfuscation level but also in the second version implementation. A similar power reduction is observed in Fig. 49 for the thermometer decoder case.

Figure 51. Performance summary of camouflaged processor circuit using RISC-V ISA with the proposed technique. These measures are given by per-unit comparison with the initial synthesis performance. Blue bars indicate area, red bars indicate power, and orange bars indicate timing.



3.4.3. Top level RISC-V implementation

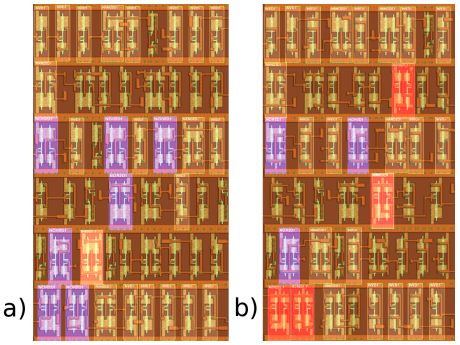
Top level execution mentioned in the previous subsection typically involves a general purpose processor. In security schemes, architectural extraction of cryptographic interfaces inside a processor is necessary to understand execution usage. Consequently, a security system requires obfuscation of application-specific circuits as well as of the general purpose processor.

A processor using an RISC-V RV32IM instruction set architecture is obfuscated with proposed design flow. The processor architecture was implemented in a 32-bit microcontroller [69]. The datapath is composed by an instruction decoder, a register file, an ALU and a multiplication/division module, and a basic interrupt controller. Memory interface controls memory operations and fetches through an AXI-4 lite master interface.

Synthesized results for different obfuscated versions of the RISC-V processor are shown in Fig. 51. In these results, power consumption is decreased up to 5% using lower obfuscation complexity. As expected, timing is gradually compromised when obfuscation complexity increases reaching values around 2%. Although results are promising, results also indicate that larger digital circuits impose a higher effort in optimization of timing overhead, resulting in possible higher overheads as indicated in Fig. 51.

Based on the results shown in section 3.4.2 and current section, the algorithm achieves a power consumption reduction for the first obfuscation level in all circuits. This reduction is due to that some generated cells might present lower power consumption than the ones in a original

Figure 52. Thermometer decoder obfuscated layouts using camouflaged cells. Detection software uses one type of NOR cell layout for circuit detection. Successfully detected cells are highlighted in blue and not detected ones are in red. a) Layout using one NOR cell, which is mostly detected. b) Layout using multiple camouflaged NOR cells with less than half detected cells.

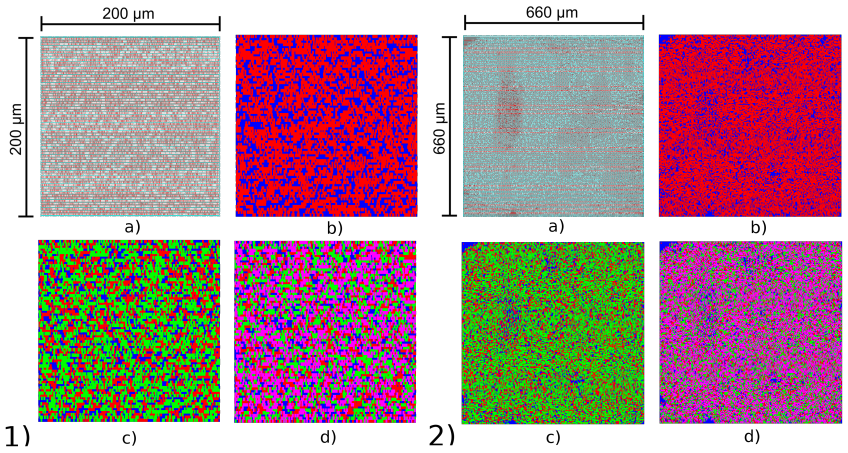


standard cell library usually designed for speed or density, which means an obfuscated design could have better power performance than an original considering the new mixing of speed- and power-based cells. Although there is possible to get better power performance, non-critical paths using the original speed-based cells can get their timing impacted. For a better timing-power tradeoff, the algorithm could compare the performance of generated cells vs original ones after characterization and add power and timing constraints to keep a balanced tradeoff. In this way, we could optimize an obfuscated design in both timing and power considering the new available cells.

3.4.4. Processing obfuscated layouts Layouts for all synthesized circuits are generated for processing purposes, using the original netlist and the four obfuscated ones. Training patterns for image processing tool used [70], are the optimal layouts of every standard cell. Optimal layouts are chosen considering information is usually available for an attacker. Detection ratio of the processing tool is obtained by comparing the number of existent standard cells in the original layout and the ones detected by software.

Some pictures are added to graphically describe the obfuscation concept. In the layout in Fig. 52a), two different kinds of NOR2D1 are used and the software is not able to identify all existing NORs in the circuit. A promising result is obtained by increasing the number of different layouts for the same NOR as shown in Fig. 52b). In this case, the image tool identifies only 3 NOR cells

Figure 53. Obfuscation highlighting on KEC (1) and RISC-V processor (2). a) is a layout that is generated using a standard library. b) shows an obfuscated layout using two different kind of standard cell layouts per gate, c) uses three, and d) uses four different layouts.



leaving 4 NOR cells unidentified proving that the method can obfuscate the hardware resulting in a detection ratio less than 50%.

Fig. 53 presents layouts for KEC and RISC-V processor graphically exemplifying the result of increased obfuscation level. Original layouts are appreciated on version a). Different levels of obfuscation are applied to the layouts and highlighted in different colors. Optimal standard cells are shown in blue, second alternative non-optimal cells in red, third alternative in green, and fourth alternative in pink. Version labeled b), for both circuits, is obfuscated with only two different types of layouts for every cell. Layouts became fuzzy when obfuscation level increased resulting in a random distribution, as indicated in Fig. 53 c) and d). Final level is observed at versions d) where the position of optimal cells is almost imperceptible.

Table 20 and Table 21 show detection results for the thermometer decoder and the 4-bit counter. Since the thermometer decoder is a combinational circuit, Table 20 shows results for INVD NAND and NOR. The detection ratio is clearly affected by obfuscation decreasing to less than 50% in some cases. For the 4-bit counter, Table 21 additionally shows results for DFD (flip-flop) due to its sequential behavior. The obfuscation process is still consistent and the detection ratio is reduced to less than 50% in many cases.

Table 22 summarizes detection results for cryptographic circuits. Table 22a) and Table 22b) show how detected cells are always less than existent for Galois multiplier and KEC respectively.

Table 20. Thermometer decoder detection results

	Thermo 1		
	Detected	**Existent**	**Ratio (%)**
INVD1	16	28	57.14%
NAND2D1	4	9	44.44%
NOR2D1	7	7	100%
	Thermo 2		
	Detected	**Existent**	**Ratio (%)**
INVD1	16	28	57.14%
NAND2D1	4	9	44.44%
NOR2D1	4	7	57.14%
	Thermo 3		
	Detected	**Existent**	**Ratio (%)**
INVD1	15	28	53.57%
NAND2D1	7	9	77.78%
NOR2D1	3	7	42.86%

These cryptographic circuits include latch cell LND which presents a similar detection ratio compared to combinational cells. However, increasing level of obfuscation seems to negatively affect detection ratio for more complex circuits such as KEC.

The most representative case can be seen in number of inverters in Table 22 b) where in the first version the detection percentage is 58.55% whereas in the third version ti goes up to 70.78%. This apparent increment in detection ratio is caused due to wrong detections of software. Fig. 54 illustrates this issue, where an actual inverter in red, a latch in blue and a wrong detected inverter strike through in blue are highlighted. The processing tool detects the inverter at the output of the latch as a single inverter causing a larger number of detected inverters. This wrong detection is also observed in NOR cells, detected as NAND but maintaining a detection ratio around 50%. This result can be seen as an additional obfuscation where similar layouts produce a look-alike effect in the processing tool without affecting original obfuscation level.

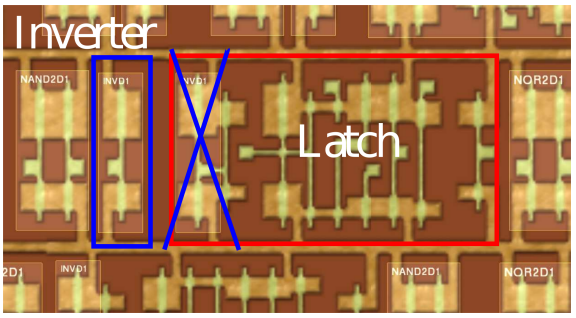
An analysis of the layout of RISC-V processor is also carried out. Table 22c) partially summarizes detection ratio where similar results to previous experiments are observed. Only a portion of layouts is analyzed due to resolution limitations in available tools. Results show inverter count increases with the level of obfuscation due to the look-alike effect and general detection ratio remains near 50 %. Even though partial layout analysis is performed, outcomes are representative considering detection ratio is congruent when different parts of the layout are processed.

Some interesting conclusions can be drawn from discussed experiments. Partial analysis

Table 21. 4-bit counter detection results

	4bitrip 1		
	Detected	**Existent**	**Ratio (%)**
INVD1	7	12	58.33%
NAND2D1	2	8	25.00%
NOR2D1	6	14	42.86%
DFD1	2	4	50.00%
	4bitrip 2		
	Detected	**Existent**	**Ratio (%)**
INVD1	5	12	41.67%
NAND2D1	3	8	37.50%
NOR2D1	11	14	78.57%
DFD1	1	4	25.00%
	4bitrip 3		
	Detected	**Existent**	**Ratio (%)**
INVD1	7	12	58.33%
NAND2D1	4	8	50.00%
NOR2D1	11	14	78.57%
DFD1	1	4	25.00%

Figure 54. Example of wrong inverter detection. Latch output stage is mistaken by an inverter during netlist extraction phase.



of processor layout causes an increment over 100% of the number of inverters as indicated in Table 22c): due to the layouts of some NAND and NOR cells are truncated at the image edges. This truncation causes an incomplete NAND layout to look like an inverter layout, resulting in an incorrect detection. In addition, the proposed algorithm does not achieve the expected reduction in detection percentage, with an increment of obfuscation level caused by software mistakenly recognizing some cells, which can be easily verified in small circuits. n complex circuits such as the processor or the establishment core, there is no easy way to verify an actual detection, highlighting the need to explore other software for further verification and development.

Table 23 presents a brief comparison with some related works. The proposed alternative adds

Table 22. Detection results on cryptographic circuits and RISC-V processor

	a) Galois field multiplier			**b)** Key establishment core (KEC)			**c)** RISC-V RV32I processor (partial)		
	GF8 1			**KEC 1**			**RISC-V 1**		
	Detected	**Existent**	**Ratio (%)**	**Detected**	**Existent**	**Ratio (%)**	**Detected**	**Existent**	**Ratio (%)**
INVD1	117	163	71.77%	948	1619	58.55%	1083	1097	98.72%
NAND2D1	51	105	48.57%	977	1853	53.8%	484	1179	41.05%
NOR2D1	117	264	44.31%	1733	3452	50.20%	1183	1916	61.74%
LND1	23	48	47.97%	334	690	48.4%	245	408	60.04%
	GF8 2			**KEC 2**			**RISC-V 2**		
	Detected	**Existent**	**Ratio (%)**	**Detected**	**Existent**	**Ratio (%)**	**Detected**	**Existent**	**Ratio (%)**
INVD1	109	163	66.87%	1061	1619	65.53%	1505	1080	137.19%
NAND2D1	53	105	50.47%	1023	1853	55.2%	397	1191	33.67%
NOR2D1	115	264	43.56%	1617	3452	46.84%	941	1896	49.11%
LND1	32	48	66.66%	297	690	43.04%	247	420	60.53%
	GF8 3			**KEC 3**			**RISC-V 3**		
	Detected	**Existent**	**Ratio (%)**	**Detected**	**Existent**	**Ratio (%)**	**Detected**	**Existent**	**Ratio (%)**
INVD1	104	163	63.76%	1146	1619	70.78%	1657	1090	151.04%
NAND2D1	68	105	64.76%	1026	1853	55.36%	338	1183	28.66%
NOR2D1	122	264	46.21%	1562	3452	45.24%	833	1898	43.47%
LND1	34	48	70.83%	320	690	46.37%	210	413	51.47%

Table 23. Comparison with some related works.

Spec.	**% Overhead(worst case)**		
	Area	**Delay**	**Power**
[71]	>20	-	>17
[14]	>12	>20	>100
[58]	Null	> 17**	> 40
[72]*	>50	> 50	> 50
This work	Null	1	4

$*$ Using 25% of obfuscated cells.
$**$ Cost function, delay and power consumption included.

one of the smallest overhead in terms of delay and power consumption. As the other approaches can provide a stronger security level, the proposed camouflage can be used as an addition in order to increase the security level without impacting resources usage or even as a complement in order to maintain security level but reducing the associated overhead.

Rajendran et. al [14] studies the security level in a layout level camouflage technique. Rajendran et. al uses a look-alike technique to camouflage NOR and NAND gates achieving a high-security level. The authors estimate the security level considering an attacker can only extract only part of the netlist due to the function of camouflage cells cannot be revealed. With this assumption, the work achieves a security level of 2^{92} but with a high overhead particularly in power consumption with a value over 100%. Our proposal works in a similar way due to we use a layout level camouflage. The main difference is that we camouflage not only NAND and NOR but latches, flip-flops, and inverters. In this way, when an attacker tries to extract the netlist, more unknown functions will appear. Therefore, our proposal can achieve a similar security level but increased by the number of camouflaged cells as follows (N-1)*2^{92} where N is the number of

camouflaged cells greater or equal to two. In addition, the associated overhead in our proposal is one of the lowest in the state of the art.

3.5. CHAPTER SUMMARY

Several obfuscation techniques are found in literature, providing alternatives to counter different threats in IP protection. These alternatives should not only provide protection but performance and cost efficiency to enhance deployability. The proposed method mitigates timing performance by keeping away slower cells from critical paths with a zero area penalty.

The proposed algorithm enables one of the smallest power and delay overhead, in the state of the art, when applied over cryptographic primitives and in a 32-bits RISC-V processor. Moreover, experiments with a large cell library and complex circuits proved the small set of used camouflaged cells are representative of the large set. Finally, the obfuscation results demonstrate that large VLSI IPs can be protected including a reduced overhead.

Obfuscation results show a reduced detection ratio as obfuscation level increases. In complex circuits, detection ratio seems to remain unaffected by the obfuscation level due to a look-alike effect. However, this effect does not reduce method effectiveness but it could be favorable considering a detection of NOR cells as NAND cells will produce an incorrect netlist, for instance.

4. ROW HAMMER ATTACK REDUCTION USING DUMMY CELLS

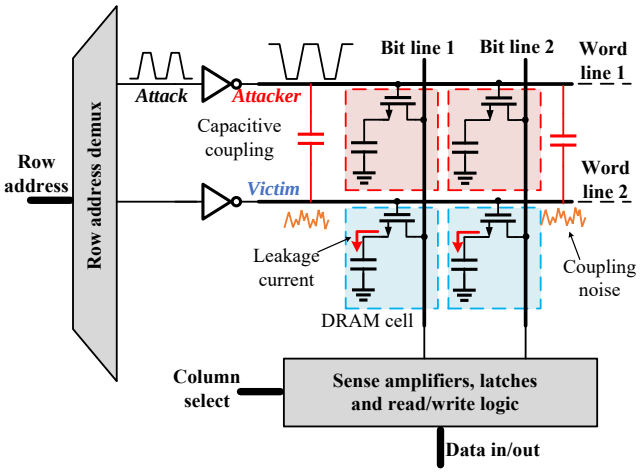
This chapter describes a low-cost and low-complexity alternative to reduce the occurrence of Row-Hammer attacks. The detection of an undesired attack is based on the use of an additional memory cell —called dummy cell—, bearing a larger leakage current and therefore a higher sensitivity to cross-talk and coupling noise. The proposed scheme achieves detection based on larger pass transistors and smaller storage capacitors. Dummy cells might be distributed across whole memory to hinder possible association attacks. Simulations on a 65nm CMOS process were performed in order to validate the proposed alternative. Process variations for coupling and interconnections were taken into account in a 64x64 memory array, such that the results are congruent with a memory dedicated, state-of-art, 28nm process. One of the most outstanding relevant aspects of the proposed solution is the reduced additional hardware space required, since it occupies less than 0.1% of the whole memory.

4.1. INTRODUCTION

Reliability in integrated circuits has become a key concern due to undesired technology scaling effects [15]. Testing solutions for integrated devices could guarantee the required reliability, but time-to-market pressure and competition reduce available time to exercise exhaustive verification [73]. As a result, shortened testing times increase unexpected failures in final products triggering a reduced market life or security issues [74]. The memory shrinking trend has brought to existence higher and more complex interactions among memory cells and reduced retention times. DRAM memories, as a case in point, have been affected by device miniaturization and reduced testing time [74, 75]. This scenario increases failure mechanisms associated with static and dynamic leakage currents and cell-to-cell couplings due to cell state transitions [15], implying the need for more complex models [74]. However, the question remains whether it is feasible to model all possible scenarios, including those that do not correspond to a regular operation [15, 16].

Fig. 55 illustrates how consecutively reading a specific row in DRAM memories, the stored information can be modified [15]. Coupling noise in adjacent rows increases the leakage current of memory cells when a specific memory address is repeatedly opened (or activated), read and

Figure 55. Coupling noise effect in high-density DRAM memories. Technology scaling has increased coupling effects which are exacerbated by PVT variations.

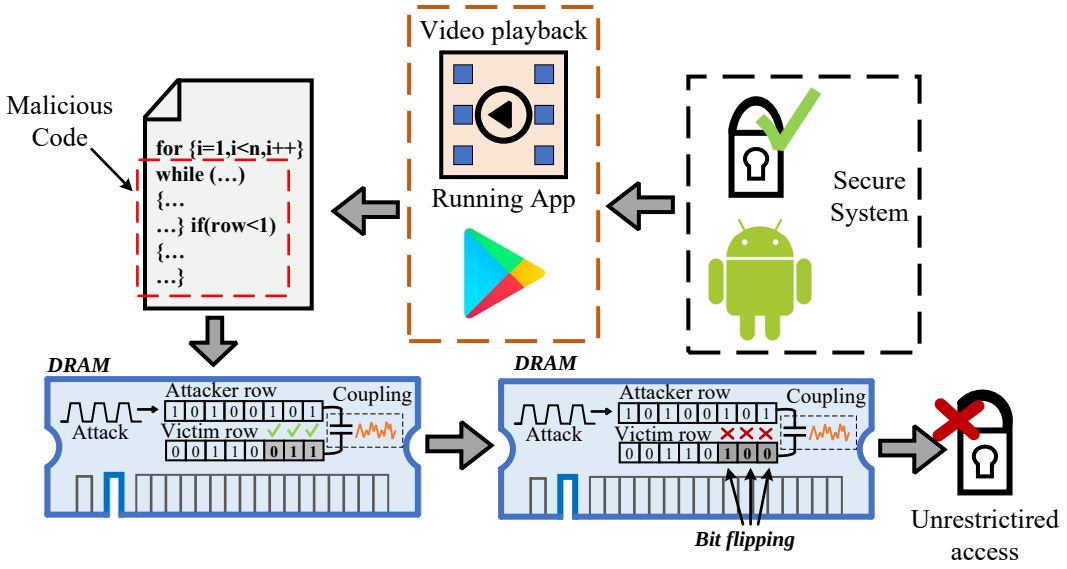


closed. This increased leakage current can cause disturbance errors in memory devices. In fact, the coupling between word lines can cause bit flips or data corruption, considering the read word line as an attacker, and the adjacent one as a victim. The latter disturbance is often considered to be the worst defect on computer system quality expectations [76].

The phenomenon of increased leakage current in adjacent rows cells is called row hammering. Recently, many works have shown how row hammering causes bit flips in modern DRAM chips. Kim et al. [16], found that 110 out of 129 DRAM modules fabricated between 2012 and 2013, which were under analysis, presented high vulnerability to row hammering. Therefore, they constructed a user level-program that flushes the cache-line while accessing a specific address, showing the existence of disturbance errors in DRAM memories used for field programmable gate array (FPGA) platforms.

At a higher system level, a team of security analysts employed by Google (Project Zero) [17] studied the row hammering phenomenon in an x86 platform. In this case, Project Zero used a refined technique based on the work of Kim et al., demonstrating that is possible to achieve privilege escalation either to escape from native client sandbox or to access all physical memory. Besides, Gruss et al. [77] developed remote attacks based on software using a website with JavaScript. Gruss et al. carried out an automated attack that remotely gained unrestricted access to systems of website visitors. More vulnerabilities such as the illustrated in Fig. 2 have been unveiled by Goodin [78]. This publication deals with the possibility to trigger row hammer attacks

Figure 56. Illustration of a DRAM row hammering attack. Booby-trapped videos or documents can be used to trigger row hammering attacks and enable privilege escalation.



using booby-trapped videos and documents. These alternatives might enable attackers to create bit flips to exploit buffer overflows and other software weaknesses. This discovery means that to get privilege escalation, an attacker can manipulate a video as Fig. 56 shows.

Literature has shown row hammering related bugs in DDR3 silicon since 2010, but the computer industry has claimed that the row hammer bug had been resolved in DDR4 memories. However, Lanteigne et al. [76] introduced a brute force attack test that applies to any computer system. Lanteigne et al. found faults in DDR3 memories and also designed a test for DDR4 memories, finding flaws in 8 out of 12 DRAM modules.

Actually, the row hammering bug has not yet been solved although literature shows different proposals to protect the legacy and future devices. In the case of legacy DRAMs, an initial approach is to increase the refreshing rate at the expense of resources overhead, but devices can still present flaws. Considering that the main idea of an attack is to bypass the cache, other solutions at the software level include reducing the time to activate error correction algorithms or restricting access to *CFLUSH* instruction. Other software level complex strategies involve the creation of kernel modules and additions to the bootloader [18, 19] but solutions can be dependent on architecture or can include huge overheads for low-performance devices.

In regards to mitigation hardware approaches, proposed solutions include deterministic and

probabilistic row refreshing depending on how often a specific row is activated to prevent bit flips in adjacent rows [15, 16]. Some of these solutions are proposed as optional modules in DDR4 memories, which implies some devices are unprotected. The reason to use these solutions as optional models is to avoid the addition of new hardware and the subsequent unacceptable overheads.

This work adopts a hardware approach by including a dummy cell which has a higher susceptibility to leakage current than standard memory cells. The information in dummy cells remains the same under normal operation so that the memory controller can identify a row-hammering attack if the information changes. Simulation results in a 65nm CMOS standard technology prove the viability of the concept. Such results can be extended to a state-of-art RAM dedicated process [79, 80].

4.2. RELATED WORK

Since the initial reports on row hammering attacks, different mitigation strategies have been proposed. Countering strategies have to consider existing and future memory devices, which implies that solutions should vary depending on the application. For existent memories, an initial software strategy consists of reducing the refresh interval to avoid data loss or corruption. In this solution, the memory controller operates with a refresh interval of 32ms instead of the standard value of 64ms. However, many works report the row hammering bug is still present in devices using this solution. Moreover, a refresh operation is expensive, and some works have proposed smart refreshing strategies to even increase the refresh interval [81, 82]. A second alternative is reducing the time to activate error correction algorithms. The interval of error corrections affects the error rate, thus diminishing the time interval can mitigate induced errors by the row hammer bug. Nonetheless, these software strategies can introduce large performance overhead [83, 84], reducing their effectiveness in practical applications.

Flexibility and easy implementation of software strategies have motivated some authors to propose alternatives to reduce possible overheads and enhance deployability. One strategy consists of blocking *CFLUSH* instructions limiting the number of cache misses. However, Aweke et al. [18] have demonstrated a *CFLUSH*-free attack that makes *CFLUSH*-based strategies no longer valid. The authors carried out several other types of attacks in order to defeat some exist-

ing protections, and end up proposing a Linux-based strategy in which a new kernel module for Intel x86 architectures is used to track the location of DRAM row access out of the last-level cache. The proposed solution reduces the number of false positives and introduces only 1% performance overhead, but, the kernel operates in Intel architectures only, and there are no developments for other ones.

Dealing with the issue of an operative system (OS) compatibility, Brasser et al. [19] propose a bootloader-based strategy. Brasser et al. designed a bootloader extension that makes vulnerable memory regions unavailable to the OS. This solution is compatible with different OSs, and reduced bit flipping probabilities but at the expense of losing an amount of memory that may be unacceptable for embedded applications. In addition, aging and extreme operating conditions lead to new vulnerable memory regions making the protection no longer valid. To overcome this limitation, Brasser et al. propose a strategy at a kernel level, that supports x86 and ARM architectures, which focus on mitigating the effects of bit flipping instead of avoiding bit flipping itself. This strategy ensures that exits at least one row between a memory row with high-security level and an attacker one, avoiding critical bit flips. All these software strategies imply a high complexity implementation and none of them are universally compatible, resulting in the need for more strategies to secure unprotected scenarios.

Hardware strategies proposed in the literature, meanwhile, divide into two approaches: counter-based and probabilistic-based row refresh. The counter-based approach, such as target row refresh (TRR), and counter-based row activation (CRA) [15], repose on counting the number of times a row (aggressor) is opened. When the count surpasses a threshold, a control signal indicates that some rows (victims) must be refreshed. This threshold based on the row hammering threshold (RH_{th}) deals with the technology used and can be expressed as follows [15]:

$$RH_{th} = \frac{\beta - 1}{\alpha - 1} \times M_{max} \tag{3}$$

where β is a factor used to guarantee an adequate refresh time, α is a constant that represents the number of times the leakage current of a memory cell increases under the row hammer attack, and M_{max} corresponds to the total possible number of activation in a refresh rate which can be 1.3 million considering the common refresh interval of 64ms. Furthermore, assuming $\alpha = 11$ (α can vary from 4 to 11.7) and $\beta = 2$, the RH_{th} results in 130K which agrees with the work reported

Figure 57. Dummy cell with reduced capacitance and increased transistor size. A dummy cell with these attributes experiments higher leakage in pass transistor and reduced storage capacitor which in turn it enables a decreased retention time.

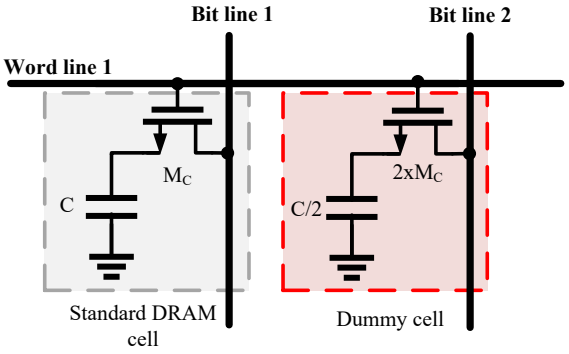
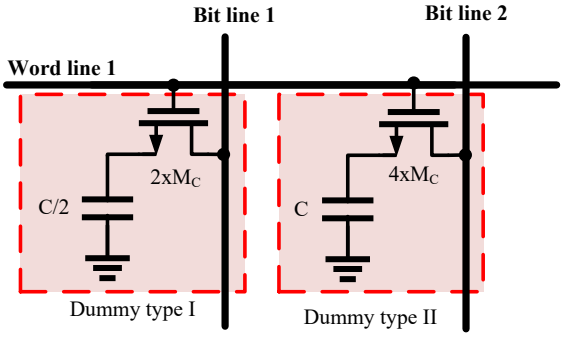


Figure 58. Different types of dummy cells. Similar susceptibility can be achieved in a cell with doubled-sized transistor and reduced capacitance than in a cell where capacitance remains unchanged and transistor is even wider.



by Kim et al. [16]. Besides, Kim et al. [15] state that for future technologies, this threshold can be in the range of tens of thousands.

The main disadvantage of counting to activate a control signal is the high-performance overhead due to the need of including one counter per row and extra memory storage. Seyedzadeh et al. [85] address this issue using a counter-based tree (CBT) that determines ”hot” rows or aggressor rows. In this way, the authors optimize the number of counters, assigning counters only to the rows that require tracking and triggering a refresh operation. Although this CBT technique reduces the number of counters, the need for extra storage results in considerable overhead.

Reducing the need for more hardware is crucial to enhance deployability of hardware-based strategies. A reduced hardware overhead approach consists of using a probabilistic approach

instead of a deterministic one. Some proposals such as pseudo-target row refresh (pTRR), probabilistic row activation (PRA) [15] and probabilistic adjacent row activation (PARA) [16] base their operation on the activation of the victim rows with low probability each time an aggressor row is activated. The activation of an aggressor row launches a random number generator to determine the activation of victim rows. A probabilistic approach avoids the need for extra storage; however, it also generates unnecessary row activation [85], besides needing a reliable random number generator.

4.3. DUMMY CELL BASED MITIGATING STRATEGY

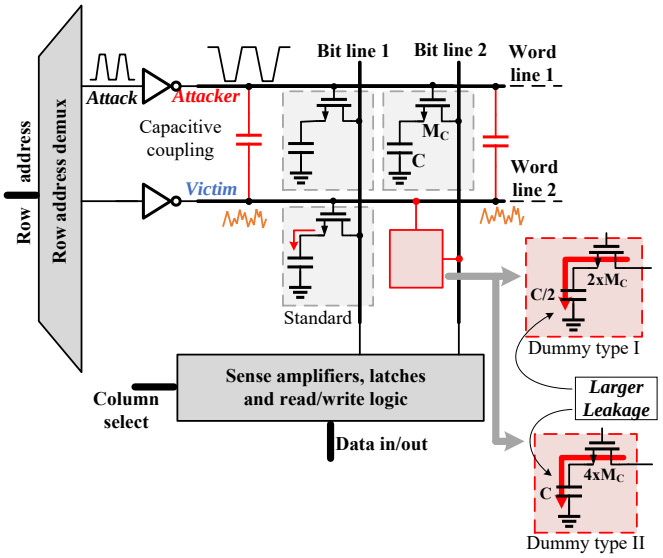
This work proposes an alternative at circuit level to reduce the effectiveness of row hammering attacks allowing software and hardware compatibility. The proposed strategy consists in connecting a dummy cell to the word line of a victim row, as Fig. 57 shows. This extra cell, on the right side in Fig. 57, is similar to a standard memory cell but more susceptible to leakage. This characteristic can be achieved by implementing a transistor of twice its nominal value size, and a capacitor of a half its capacitance. A wider transistor has a larger leakage current that, combined with the reduced storage capacity, results in a cell with a decreased retention time.

A cell with this particular attribute is useful as an indicator of a possible attack because: first, the dummy cell must be pre-charged to a logic level one (1) during the refresh phase; in this case, the memory controller must ignore the dummy cell in the attacker word line. Then, if an attacker carries out a malicious memory access, the dummy cell in the victim row will experiment a larger leakage current than a standard cell, its capacitor discharging at an increased rate. Therefore, the information stored in the dummy cell is corrupted before that of the standard cell.

If the memory controller senses the output data of dummy cells, the respective victim row or those nearby can be refreshed when the controller detects a logic zero. This new refresh avoids a possible bit flipping, preventing data corruption. Moreover, this method avoids the need for an increased refresh rate because the memory controller only refreshes the critical rows.

One important characteristic of the proposed alternative is the fact that it does not include more complex hardware, because it only uses slightly modified cells. Also, dummy cells can be placed in specific memory locations to protect critical information, or in a random way across the whole memory. It is important to highlight that the memory structure is not altered, and thus it is

Figure 59. Accelerated discharge process using a high-leakage dummy cell. During a row activation, coupling noise in near rows induces a higher leakage current in dummy cells, causing a faster lost of capacitor charge that in standard memory cells.



difficult for a hacker to find unusual hardware that might catch his attention.

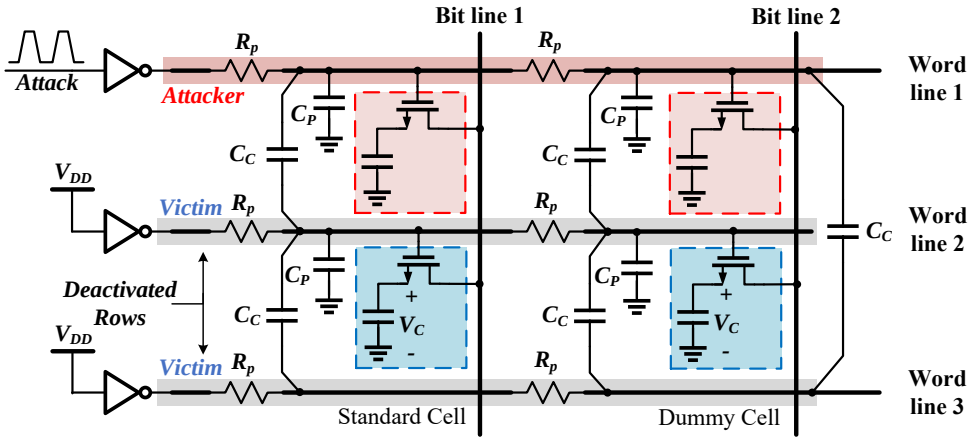
4.3.1. Enhancing deployability of the dummy cell approach

Capacitor retention time is of the utmost concern in modern DRAMs. Variability in this specification makes testing routines time-consuming and complex [74, 75]. Hence, DRAM cells' capacitor go through an optimization process in order to give robustness to their retention time, reducing susceptibility to leakage, for instance.

Considering the optimization process over a cell capacitor, a modification in its value can result in an excessive performance degradation. Therefore, the dummy cell proposal requires an exhaustive verification under different test patterns, although simulations with a transistor twice its size and lower capacitance show consistence.

Figure 58 depicts an alternative to enhance robustness of a dummy cell. The figure shows a cell equivalent to the original proposal, maintaining the capacitor size but with a wider transistor. A transistor four times its original width creates a similar leakage susceptibility in the cell. The advantage is that a cell of this type can consist of standard transistors in parallel which means no need for additional sizing. In this way, we guarantee an early alert monitoring cell avoiding

Figure 60. Circuit level designed testbench to emulate row hammering attacks. The induced leakage phenomenon is introduced with an RC coupling model and all memory cells are pre-charged to VDD. Cells capacitor voltage of near rows are monitored when a specific row is repeatedly activated.



incompatibilities with standard RAM-dedicated fabrications processes.

Figure 59 shows how the dummy cells experiment a larger leakage current under a row-hammering attack. The figure illustrates how a memory array can include any type of dummy cell to monitor a specific row. As both dummy cells experiment similar leakage susceptibility, the dummy cell type II can provide the same functionality as the original idea, but using a standard capacitor value.

4.4. RESULTS

In order to validate the proposed technique, transient simulations were carried out on an array of 64×64 memory cells, including one dummy cell per row. The array includes a distributed RC network that simulates coupling between adjacent rows, as Fig. 60 shows [86]. Moreover, a capacitance between word lines one and three was added, aiming to simulate coupling due to higher metal interconnections —the value of coupling elements was obtained from technology files—.

The DRAM array was implemented in a 65nm CMOS standard technology, and the results could be extended to a state-of-art RAM dedicated process [79, 80]. A RAM process is optimized to reduce CMOS high order effects, so cell transistors are designed to have low leakage

Figure 61. Induced leakage current in the pass transistor of a standard memory cell and two different sized dummy cells. Leakage current is higher in dummy cells and it can be increased with wider pass transistors.

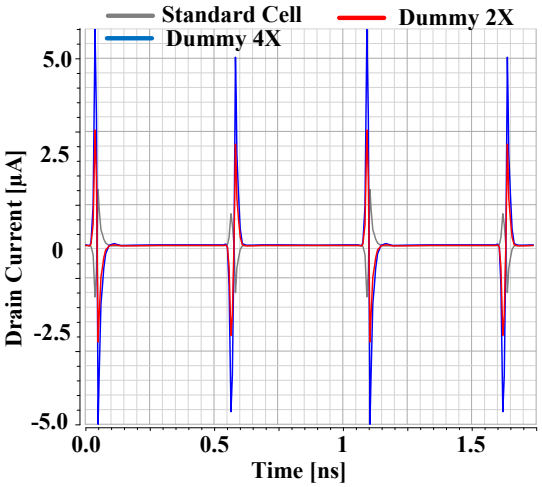
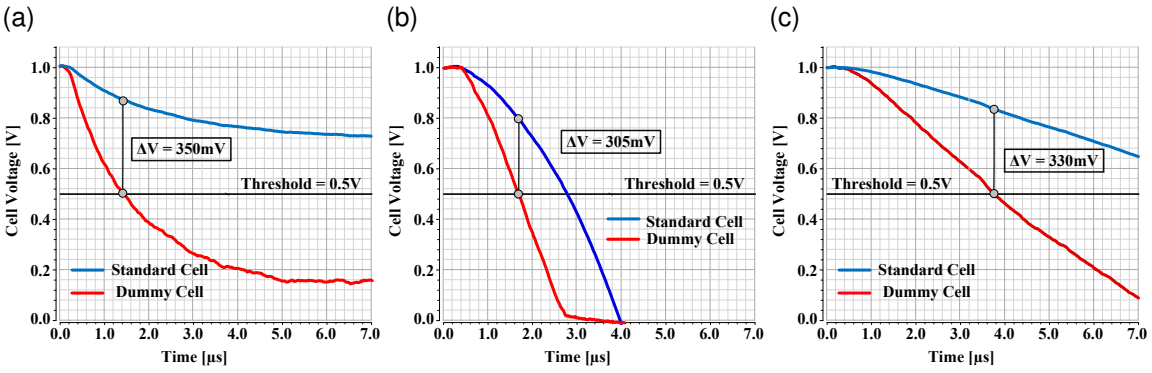


Figure 62. Discharge process under the influence of coupling noised in a DRAM array. Transient discharge behavior is extracted under different process corner: a) Typical case b) Fast NMOS and capacitor corners, at high temperature, and c) Slow NMOS and capacitor corner, and low temperature.



current (as a first alternative to mitigate row hammering). Therefore, memory-dedicated technologies have devices with reduced current-capability and performance regarding conventional CMOS processes. Furthermore, in spite of a conventional 65nm process having a larger parasitic capacitance than a 28nm process, the former has higher current density, leading to an increment of the total performance.

The proposed method is based on increasing cell leakage current sensitivity due to coupling noise under a read operation. Accordingly, the first simulation is related to the quantification of the

Figure 63. Discharge process in different sized dummy cells due to coupling noise in a repeatedly read operation. An accelerated capacitor discharge rate is observed as the cell-transistor's width increases.

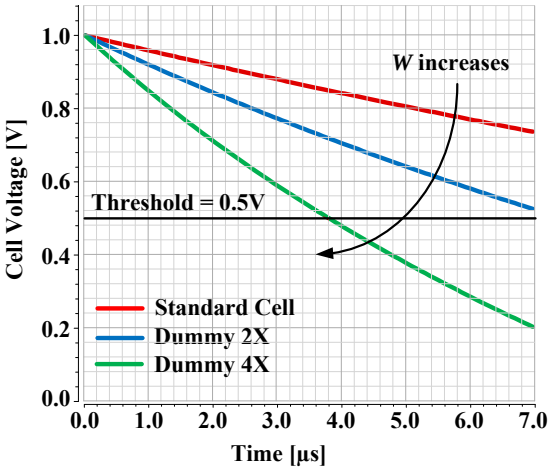
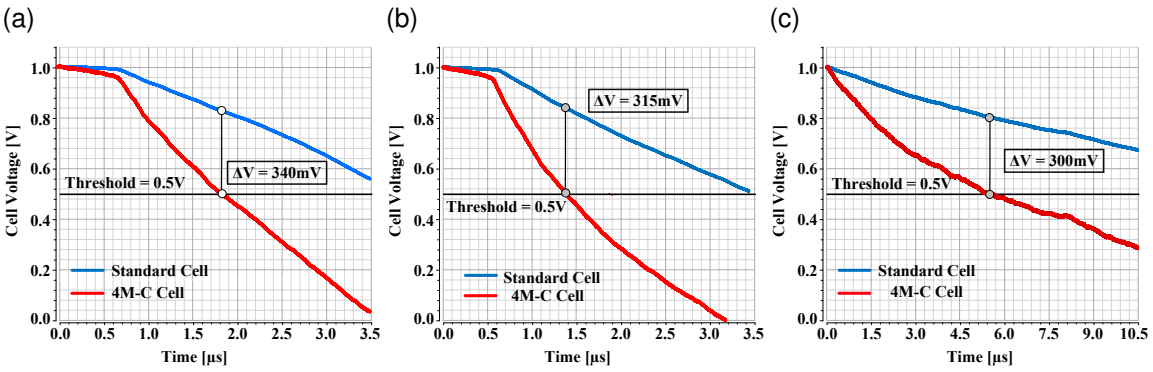


Figure 64. Discharge process under the influence of coupling noised in a DRAM array. Transient discharge behavior is extracted under different process corner: a) Typical case b) Fast NMOS and capacitor corners, at high temperature, and c) Slow NMOS and capacitor corner, and low temperature.



leakage current through the pass transistors of a standard cell and different sized dummy cells, as Fig. 61 shows. For the standard cell, dimensions of the pass transistor and the storage capacitor are 800nm$\times$65nm and 56fF respectively. For 2$\times$ and 4$\times$ dummy cells, the pass transistor has twice and four times the width of the standard cell's, while the storage capacitor is half the former.

When an attacker row is accessed (word line one), cells on victims rows are partially activated through coupling capacitances C_C. Each transition of row one will produce a peak voltage on victims rows, that increases the sub-threshold current of each access transistor. As a result,

storage capacitors are gradually discharged until bit flipping is achieved. As Fig. 61 shows, maximum leakage for dummy cells doubles every time the width of its pass transistor is doubled, such that when an attack is carried out the discharge process of the storage capacitor is faster. Access frequency is 1GHz, producing a maximum leakage current of 1μA for a standard cell while reaching 3μA and 5.8μA for the 2$\times$ and 4$\times$ dummy cells respectively.

Independently of the operation mode, DRAM cell retention time has to be higher than a refresh interval. In a state of the art DRAM memory, a distribute refresh operation is generally adopted which leads to a refresh interval cycles of 7,8μs [87]. Therefore, proper operation of a DRAM memory cell implies a retention time greater than this interval. A dummy cell under a row hammer attack needs to have a retention time less than the refresh interval cycles. Simulations of the retention time of standard and dummy memory cells in victim rows under process and temperature variations are shown in Fig. 62. Blue wave-forms correspond to the capacitor voltage of the standard cell, and red lines represent dummy cell discharge. Process variations cover NMOS', and capacitor's fast and slow corners and the temperature was varied from -40 to 120 °C.

The threshold for the memory controller alert is set to $V_{DD}/2$ because in a read operation, the bit line is pre-charged to this value. A logic high is sensed when the voltage is greater than $V_{DD}/2$ and vice-versa [88]. Fig. 62 shows that the dummy cell loses its charge with an increased rate, ensuring that capacitor voltage V_c crosses the threshold before the standard cell one does, in less than 7,8μs. It means that the information stored in dummy cells is corrupted with enough time in advance, such that the memory controller can refresh the row (or the whole array) preventing information loss. Moreover, when the dummy cell crosses the threshold, the voltage of the standard cell is 300mV larger for the worst case, thus ensuring that information is not corrupted when refreshing or reading.

The concept of the cell increased leakage susceptibility considering a wider transistor and a lower capacitance can be applied without affecting the capacitor value. If the cell's capacitor remains unchanged, an equivalent dummy cell can be designed by raising even more the cell-transistors width. Figure 63 shows the discharge process of the dummy cell capacitor for various cell's transistor widths. As the dimensions of the access device increase the leakage current rises and the time required to cross the memory threshold decreases. Therefore, a dummy cell with four times its transistor size alerts of an attack in a similar time than a double sized cell with reduced capacitance.

This strategy of using a standard capacitor while having a wider transistor is also validated in the 64×64 array. Voltage and temperature variations are carried out in order to extract the voltage and time difference as in the original approach. Figure 64 shows the discharge process for a dummy cell in red and a standard cell's in blue. Simulations show consistency over the different operating conditions, resulting in a minimum voltage difference of approximately 300mV. Hence, the modified dummy cell also guarantees an early alert with enough time in advance for the controller to trigger a refresh operation.

A highlight of the proposed strategy is the associated low-hardware overhead. A typical DDR4 memory has 512 bytes (4096 cells) per row [89]. So, if only a dummy cell is added to each row, there is an increment of less than 0.1% in the whole memory area. Moreover, adding one dummy cell for each 4096 makes it difficult for a hacker to identify extra security.

4.5. CHAPTER SUMMARY

Row hammering attacks are still possible in modern DRAM chips. Attack mitigation strategies are not universal, and in some cases, cannot be feasible. This work demonstrates a hardware-based solution by slightly modifying a memory cell with the addition of an extra dummy in order to increase leakage susceptibility and using it as an attack indicator. Increased leakage susceptibility ensures that a dummy cell loses its charge faster enough than a standard memory cell, which can be used as a refresh indicator when an attack is being carried out.

Main highlights of the proposed method are: no modifications to memory architecture, implementation flexibility which means dummy cells do not need to be placed in specific areas, and robustness against process, voltage and temperature variations. The method's flexibility also allows random placement in order to avoid easy identification by an attacker. Besides, low area overhead is added with —around 0.1%— considering that only one memory cell needs to be added per row.

The proposed method can be used along with other mitigation strategies so to enhance security. For instance, embedded performance counters in modern DRAM chips allow for the extraction of additional information in order for the memory controller to perform accurate refreshes. Finally, since memory architecture is not modified, the proposed method may be applied in tandem with software strategies.

5. SUMMARY OF CONTRIBUTIONS AND CONCLUSIONS

5.1. SUMMARY OF CONTRIBUTIONS

The key contributions of this dissertation are the following:

- *A detailed implementation of a fully-synthesized key-establishment core* with fast rekeying, featuring two proposed circuit techniques [20, 21].
- *A fully-synthesized TRNG architecture with a post-processing stage based on a cellular automaton* that achieves fast post-processing and high reconfigurability [22, 23].
- *A low-cost fully-synthesized multi-throughput TRNG* based on a proposed adjustable throughput oscillator, using an inverter bypass technique [24].
- *A methodology to harvest entropy in a fully-synthesizable way* from at least two low-cost entropy sources, offering low-resources extraction (patent request) [24].
- *A methodology to camouflage standard cells* in order to prevent reverse engineering with one of the lowest overheads reported in terms of area and delay, featuring an on-fly standard cell generator [25, 26].
- *A hardware strategy to mitigate row hammering attacks* in DRAM memories with only a 0.2% area overhead and enhanced compatibility with standard DRAM processes (patent request) [27–29].

All these contributions required extensive verification in software and hardware and an exhaustive revision of the state-of-the-art. Additionally, this research work proposed a relaxation oscillator architecture described in the appendix. The proposed architecture offers low-power consumption, high start-up energy efficiency and a wide-band operation in contrast to traditional relaxation oscillators.

5.2. CONCLUSIONS

A secure key establishment is fundamental in any crypto-system. This dissertation has proposed enhancement in security with two circuits techniques by implementing a variable length PRBS

with a masked output for a TPM core [20, 21]. An advantage of using a TPM core is the low-cost rekeying associated, which is enhanced with an additional random operation using the proposed PRBS. The proposed core achieves fast and inexpensive rekeying with low-area and low-power consumption. A proposed FoM offers the possibility to compare resources usage with state-of-the-art cores.

Key generation is also important since the security level is measured concerning the entropy of the key. There are two general approaches to extract entropy in order to generate those keys: those based on analog circuits and those in the digital domain. Digital domain approaches offer low-complexity implementations and inherent robustness to process and environmental variations. This dissertation has proposed different circuits and techniques to achieve a full-synthesizable architecture that generates high-quality random numbers. The first architecture uses jitter accumulation and ring oscillator multiplexing technique as an entropy source enabling multi-throughput operation. This dissertation also proposed a post-processing stage aiming at enhancing the effective throughput of the TRNG. For high-quality random numbers, the ring oscillators require a large number of stages, increasing the amount of jitter accumulation. Hence this limits the throughput considering that even with larger rings, the output bits need to be sampled or truncated. The cellular automaton post-processing stage here proposed enhances output quality with a pure combinational operation [22, 23]. However, a frequency coupling between rings due to the multiplexing scheme introduced a bias in data, degrading output quality. This issue could have been avoided, using independent enables for the ring pairs.

This dissertation has also proposed a second architecture that maintains the objective of operating with different throughput [24]. Using a modified multi-mode ring oscillator with a bypassing inverter stage feature, the TRNG provides random numbers at different rates. The proposed TRNG was taped-out and measured under power supply variations. The multi-throughput architecture operates with similar statistical behavior to the classical multi-mode approach, passing all NIST tests. The shortest or fastest ring still present low-quality output numbers. Hence, a next prototype should include longer stages.

Avoiding reverse engineering requires to obfuscate designs. State-of-the-art obfuscation techniques introduce considerable resources overhead that prevent their implementation. This work has proposed a layout level camouflage technique featuring an on-the-fly standard cell generator [25, 26]. The proposed method uses timing restrictions to mitigate any possible overhead, and

it includes null area overhead as the generated standard cell layouts occupy the same area as the original ones. Results show that the proposed technique can protect IPs with a high-security level. A possible enhancement is to include power constraints in the obfuscation algorithm to optimize for power consumption since generated cells might consume less power than original standard cells. A power consumption reduction is possible, considering the new mixing of speed- and power-based cells.

A recently reported DRAM memory bug showed potential security issues. The bug appears when a word-line or address is repeatedly read causing bit flips in adjacent rows. The operation of continuously reading a specific address cause increased current leakage, triggering the bit-flipping. Although the industry has proposed some solutions, row hammering is still prevalent in some modern DRAM chips. Therefore, this dissertation has offered a hardware solution to mitigate this bus using dummy or monitoring cells. Two alternatives have been discussed considering deployability in a standard DRAM process. The first one is to use a monitoring cell with a transistor cell width twice that of a conventional transistor cell and a capacitor of half the value of a standard capacitor [27,28]. A cell with these characteristics presents an increased susceptibility to leakage current so that when an attacker aims to produce a bit flipping the capacitor of a dummy cell discharges before a standard capacitor. If the memory controller systematically stores a logic high in dummy cells, any change in that information may be used to trigger an attack alert. In this way, the memory controller can perform a refresh operation before any information loss in the standard cells.

The proposed row hammering protection has been validated with extensive simulations in a 65nm CMOS standard technology. The proposed solution is robust against the process and environmental variations and offers less than 0.2% overhead considering one dummy cell per row. However, a dedicated DRAM process optimizes the capacitor size to reduce leakage susceptibility. Reducing capacitor value could generate an excessive leakage susceptibility affecting the normal operation. This issue could not be verified due to lack of access to a dedicated DRAM process. Therefore, a second alternative was proposed where the dummy cell experiments similar leakage susceptibility but keeping the standard capacitor [29]. The second proposal uses a transistor four times wider than a standard transistor cell which can be implemented using four standard transistors in parallel. In this way, the dummy cell will only use standard capacitor and transistor cell sizes, enhancing proposal deployability. The second approach was validated in the same

way as the first approach with similar results, which indicates that the second approach is also feasible.

A possible approach to further enhance compatibility with DRAM processes is to use the cells that a typical DRAM verification process chooses to make unusable due to their high leakage susceptibility. These unusable or weak cells could be used as an early alert indicator instead of using the proposed dummy cells. In this way, the monitoring system might use only standard DRAM cells. Implementation of this mechanism requires testing of fabricated DRAM cells using standard procedures. Instead of marking weak cells as unusable, the memory controller always stores the same information in those cells. This information should remain unchanged while memory operation, hence a row-hammering attack alert appears when those cells lose their information. In this way, the memory controller can trigger a refresh operation to avoid the bit-flipping in the regular cells.

5.3. PERSPECTIVES

A particular aim in this dissertation has been to propose security solutions at the hardware level for restricted cryptosystems. As the number of interconnected devices increase, the need for robust, flexible and resources efficient cryptosystems is more evident. Recent studies have developed key establishment hardware approaches using elliptic curve cryptography. In contrast, this dissertation has proposed a key establishment core using tree parity machine as a light-weight alternative. However, the recent developments in elliptic curve hardware-friendly implementations have shown that algorithm optimization can achieve high energy-efficient circuits with low-cost deployments. Therefore, this field of study requires to be explored to identify if a TPM core achieves better specifications that new elliptic curve implementations. This study is necessary since elliptic curves are used within well-defined protocols, propelling thus further research.

Development of all-digital TRNGs is a clear trend in the literature. All-digital TRNGs offer low-complexity implementations, still requiring calibration schemes and even post-processing so to achieve high-quality standards. This work has proposed two different fully-synthesized TRNGs which require further research: biased output data evidence low-robustness and the need for digital calibration to alleviate post-processing and output truncation. Another alternative is to use an even number of inverter cells which relaxes circuit complexity and enables simpler calibration

schemes.

Additional analysis is also required in the multi-throughput TRNG. Although the multi-throughput approach offers output numbers with the required quality for a fixed supply voltage, results showed how the TRNG is affected by supply variations. Therefore, exhaustive testing for different voltage supplies must be carried on, checking different operating rates. In addition, temperature variations must be included to have a complete verification of the TRNG robustness.

Countering reverse engineering is a field that needs continuous research. This dissertation has proposed a method to camouflage digital circuits at the layout level. The technique requires further verification with post-fabrication measurements. Besides, the approach here offered can boost other camouflage methods. Hence, new research combining different obfuscation methods is needed.

The prevalence of row hammering bug continuous to be a reliability and security threat in modern DRAM chips. This dissertation has offered a hardware strategy to mitigate it, but further validations are required. Although the method works well under simulations, it needs additional verification with a dedicated DRAM process. Compatibility issues are the main topic to evaluate, and post-fabrication validation is a must. Moreover, proof of concept of the compatibility enhancement is required. A standard DRAM testing for fabricated commercial chips is necessary to identify the targeted cells as well as the design of software compatible with the memory controller to fully deploy the approach.

5.4. CONTRIBUTION LIST

Conference papers

1. **H. Gomez**, C. Duran and E. Roa, ”Standard Cell Camouflage Method to Counter Silicon Reverse Engineering,” 2018 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, 2018, pp. 1-4. doi: https://doi.org/10.1109/ICCE.2018.8326300.

2. A. Amaya, **H. Gomez** and E. Roa, ”Mitigating Row Hammer Attacks Based on Dummy Cells in DRAM,” 2017 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, 2017, pp. 442-443. doi: https://doi.org/10.1109/ICCE.2017.7889389.

3. **H. Gomez**, A. Amaya and E. Roa, "DRAM Row-Hammer Attack Reduction Using Dummy Cells," 2016 IEEE Nordic Circuits and Systems Conference (NORCAS), Copenhagen, 2016, pp. 1-4. doi: https://doi.org/10.1109/NORCHIP.2016.7792886.

4. J. Cartagena, **H. Gomez** and E. Roa, "A Fully-Synthesized TRNG with Lightweight Cellular-Automata Based Post-Processing Stage in 130nm CMOS," 2016 IEEE Nordic Circuits and Systems Conference (NORCAS), Copenhagen, 2016, pp. 1-5. doi: https://doi.org/10.1109/NORCHIP.2016.7792898

5. **H. Gómez**, Ó. Reyes and E. Roa, "A Fully Synthesized Key Establishment Core Based on Tree Parity Machines in 65nm CMOS," 2016 12th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME), Lisbon, 2016, pp. 1-4. doi: https://doi.org/10.1109/PRIME.2016.7519517.

Journal papers

1. **H. Gomez**, C. Duran, E. Roa, "Defeating Silicon Reverse Engineering Using a Layout-Level Standard Cell Camouflage," IEEE Transactions on Consumer Electronics, 2019. doi: https://doi.org/10.1109/TCE.2018.2890616, Early Access.

2. **Hector Gomez**, Óscar Reyes, Elkim Roa, "A 65nm CMOS Key Establishment Core Based on Tree Parity Machines", Integration, Volume 58, 2017, Pages 430-437, ISSN 0167-9260, doi: https://doi.org/10.1016/j.vlsi.2017.01.010.

Papers in submission/revision process

1. **H. Gomez**, A. Amaya and E. Roa, "A Low-Cost Technique to Reduce Vulnerability Against DRAM Row-Hammering Attacks", submitted to Microelectronics Reliability.

2. **H. Gomez** and E. Roa, "An All-Digital Cellular Automaton Based True Random Number Generator in 130nm CMOS", submitted to IEEE Transactions on Very Large Scale Integration (VLSI) Systems.

3. **H. Gomez**, J. Arenas and E. Roa, "Low-Cost TRNG IPs", submitted to IEEE Transactions on Very Large Scale Integration (VLSI) Systems.

Other publications

1. **H. Gomez**, J. Arenas, C. Rojas, D. Reyes, A. Mantilla and E. Roa, "A 68/36ppm/°C TC 32.768kHz-to-1MHz RC-based Oscillator with 72/6pJ Start-up Energy," in 2019 IEEE Custom Integrated Circuits Conference (CICC). Accepted for oral presentation.

2. W. Ramirez, **H. Gomez**, E. Roa, "On UVM Reliability in Mixed-Signal Verification," 2019 IEEE 10th Latin American Symposium on Circuits & Systems (LASCAS), Armenia, Colombia, 2019, pp. 233-236. doi: https://doi.org/10.1109/LASCAS.2019.8667543.

3. Ckristian Duran, Luis E. Rueda G., Andres Amaya, Rolando Torres, Javier Ardila, Luis Rueda D., Giovanny Castillo, Anderson Agudelo, Camilo Rojas, Luis Chaparro, Harry Hurtado, Juan Romero, Wilmer Ramirez, **Hector Gomez**, Hugo Hernandez and Elkim Roa, "A System-on-Chip Platform for the Internet of Things Featuring a 32-bit RISC-V Based Microcontroller," 2017 IEEE 8th Latin American Symposium on Circuits & Systems (LASCAS), Bariloche, 2017, pp. 1-4. doi: https://doi.org/10.1109/LASCAS.2017.8126878.

4. A. Amaya, **H. Gomez** and E. Roa, "A Digital Offset Correction Method for High Speed Analog Front-Ends," 2016 29th Symposium on Integrated Circuits and Systems Design (SBCCI), Belo Horizonte, 2016, pp. 1-4. doi: https://doi.org/10.1109/SBCCI.2016.7724077.

5. Ckristian Duran, Luis Rueda D., Giovanny Castillo, Anderson Agudelo, Camilo Rojas, Luis Chaparro, Harry Hurtado, Juan Romero, Wilmer Ramirez, **Hector Gomez**, Javier Ardila, Luis Rueda, Hugo Hernandez, Jose Amaya and Elkim Roa, "A 32-bit RISC-V AXI4-Lite Bus-Based Microcontroller with 10-bit SAR ADC," 2016 IEEE 7th Latin American Symposium on Circuits & Systems (LASCAS), Florianopolis, 2016, pp. 315-318. doi: https://doi.org/10.1109/LASCAS.2016.7451073.

6. Edwin G. Carreño, Christian D. Hernandez, Oscar M. Diaz, **Hector Gomez**, Carlos Fajardo, Hugo Hernandez, Wilhelmus Van Noije and Elkim Roa, "A 3.9 Compression-Ratio Huffman Encoding Scheme for the Large Ion Collider on 65nm and 130nm CMOS technologies," 2016 IEEE 7th Latin American Symposium on Circuits & Systems (LASCAS), Florianopolis, 2016, pp. 347-350. doi: https://doi.org/10.1109/LASCAS.2016.7451081.

7. Amaya, Andrés F.; **Gomez, Héctor I.**; Espinosa, Guillermo. An Area Efficient High Speed,

Fully On-Chip Low Dropout -LDO- Voltage Regulator. Ing. compet., Cali , v. 17, n. 1, p. 153-160, June 2015 .

Patents

1. **H. Gomez**, E. Roa, ”Generador de Cadena Aleatoria de Bits Usando Fuentes Débiles”. At Colombian Patent Office in examination process.

2. A. Amaya, **H. Gomez** and Elkim Roa, ”Método y Aparato para la Protección de Memorias RAM contra Ataques Informáticos ”. At Colombian Patent Office in examination process.

Bibliography

[1] J. A. Stankovic, “Research Directions for the Internet of Things,” *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 3–9, Feb 2014.

[2] R. Hussain and I. Abdullah, “Review of different encryptionand decryption techniques used for security and privacy of iot in different applications,” in *2018 IEEE International Conference on Smart Energy Grid Engineering (SEGE)*, Aug 2018, pp. 293–297.

[3] D. Minoli, K. Sohraby, and J. Kouns, “Iot security (iotsec) considerations, requirements, and architectures,” in *2017 14th IEEE Annual Consumer Communications Networking Conference (CCNC)*, Jan 2017, pp. 1006–1007.

[4] M. Meriac, “Heart of Darkness - exploring the uncharted backwaters of HID iCLASS security,” in *27TH CHAOS COMMUNICATION CONGRESS*, no. December, Berlin, 2010, pp. 1–13.

[5] U. Banerjee, C. Juvekar, A. Wright, Arvind, and A. P. Chandrakasan, “An energy-efficient reconfigurable dtls cryptographic engine for end-to-end security in iot applications,” in *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, Feb 2018, pp. 42–44.

[6] K. Yang, D. Fick, M. B. Henry, Y. Lee, D. Blaauw, and D. Sylvester, “16.3 A 23Mb/s 23pJ/b fully synthesized true-random-number generator in 28nm and 65nm CMOS,” in *Digest of Technical Papers - IEEE International Solid-State Circuits Conference*, vol. 57, San Francisco, CA, 2014, pp. 280–281.

[7] K. Yang, D. Blaauw, and D. Sylvester, “An All-Digital Edge Racing True Random Number Generator Robust Against PVT Variations,” *IEEE Journal of Solid-State Circuits*, vol. 51, no. 4, pp. 1022–1031, April 2016.

[8] S. Srinivasan, S. Mathew, R. Ramanarayanan, F. Sheikh, M. Anders, H. Kaul, V. Erraguntla, R. Krishnamurthy, and G. Taylor, “2.4GHz 7mW all-digital PVT-variation tolerant True Random Number Generator in 45nm CMOS,” in *2010 Symposium on VLSI Circuits*, June 2010, pp. 203–204.

[9] S. K. Mathew, S. Srinivasan, M. a. Anders, H. Kaul, S. K. Hsu, F. Sheikh, A. Agarwal, S. Satpathy, and R. K. Krishnamurthy, “2.4 Gbps, 7 mW All-Digital PVT-Variation Tolerant True

Random Number Generator for 45 nm CMOS High-Performance Microprocessors," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 11, pp. 2807–2821, Nov. 2012.

[10] M. Volkmer and S. Wallner, "A Key Establishment IP-Core for Ubiquitous Computing," in *16th International Workshop on Database and Expert Systems Applications (DEXA'05)*. Copenhagen: IEEE, 2005, pp. 241–245.

[11] B. Erbagci, C. Erbagci, N. E. C. Akkaya, and K. Mai, "A Secure Camouflaged Threshold Voltage Defined Logic Family," in *2016 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, May 2016, pp. 229–235.

[12] A. Vijayakumar, V. C. Patil, D. E. Holcomb, C. Paar, and S. Kundu, "Physical Design Obfuscation of Hardware: A Comprehensive Investigation of Device and Logic-Level Techniques," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 1, pp. 64–77, Jan 2017.

[13] J. Rajendran, Y. Pino, O. Sinanoglu, and R. Karri, "Security Analysis of Logic Obfuscation," in *DAC Design Automation Conference 2012*, June 2012, pp. 83–89.

[14] J. Rajendran, O. Sinanoglu, and R. Karri, "VLSI Testing Based Security Metric for IC Camouflaging," in *2013 IEEE International Test Conference (ITC)*, Sept 2013, pp. 1–4.

[15] D. H. Kim, P. J. Nair, and M. K. Qureshi, "Architectural Support for Mitigating Row Hammering in DRAM Memories," *IEEE Computer Architecture Letters*, vol. 14, no. 1, pp. 9–12, Jan 2015.

[16] Y. Kim, R. Daly, J. Kim, C. Fallin, J. H. Lee, D. Lee, C. Wilkerson, K. Lai, and O. Mutlu, "Flipping bits in Memory without Accessing Them: An Experimental Study of DRAM Disturbance Errors," in *2014 ACM/IEEE 41st International Symposium on Computer Architecture (ISCA)*, June 2014, pp. 361–372.

[17] P. Zero, "Exploiting the DRAM Row-Hammer Bug to Gain Kernel Privileges." [Online]. Available: http://googleprojectzero.blogspot.com.co/2015/03/exploiting-dram-rowhammer-bug-to-gain.html

[18] Z. B. Aweke, S. F. Yitbarek, R. Qiao, R. Das, M. Hicks, Y. Oren, and T. Austin, "Anvil: Software-based protection against next-generation rowhammer attacks," *SIGARCH Comput. Archit. News*, vol. 44, no. 2, pp. 743–755, Mar. 2016. [Online]. Available: http://doi.acm.org/10.1145/2980024.2872390

[19] F. F. Brasser, L. Davi, D. Gens, C. Liebchen, and A. Sadeghi, “Can’t touch this: Practical and generic software-only defenses against rowhammer attacks,” *CoRR*, vol. abs/1611.08396, 2016. [Online]. Available: http://arxiv.org/abs/1611.08396

[20] H. Gomez, O. Reyes, and E. Roa, “A Fully Synthesized Key Establishment Core Based on Tree Parity Machines in 65nm CMOS,” in *2016 12th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME)*, June 2016, pp. 1–4.

[21] ——, “A 65nm CMOS Key Establishment Core Based on Tree Parity Machines,” *Integration, the VLSI Journal*, vol. 58, pp. 430 – 437, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S016792601730055X

[22] J. Cartagena, H. Gomez, and E. Roa, “A Fully-Synthesized TRNG with Lightweight Cellular-Automata based Post-processing Stage in 130nm CMOS,” in *2016 IEEE Nordic Circuits and Systems Conference (NORCAS)*, Nov 2016, pp. 1–5.

[23] H. Gomez and E. Roa, “An All-Digital Cellular Automaton Based True Random Number Generator in 130nm CMOS,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2019.

[24] J. A. H. Gomez and E. Roa, “Low-Cost TRNG IPs,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2019.

[25] H. Gomez, C. Duran, and E. Roa, “Standard Cell Camouflage Method to Counter Silicon Reverse engineering,” in *2018 IEEE International Conference on Consumer Electronics (ICCE)*, Jan 2018, pp. 1–4.

[26] H. Gomez, C. Duran, and E. Roa, “Defeating Silicon Reverse Engineering Using a Layout-Level Standard Cell Camouflage,” *IEEE Transactions on Consumer Electronics*, vol. 65, no. 1, pp. 109–118, Feb 2019.

[27] A. Amaya, H. Gomez, and E. Roa, “Mitigating Row Hammer attacks based on dummy cells in DRAM,” in *2017 IEEE International Conference on Consumer Electronics (ICCE)*, Jan 2017, pp. 442–443.

[28] H. Gomez, A. Amaya, and E. Roa, “DRAM row-hammer attack reduction using dummy cells,” in *2016 IEEE Nordic Circuits and Systems Conference (NORCAS)*, Nov 2016, pp. 1–4.

[29] ——, “A Low-Cost Technique to Reduce Vulnerability Against DRAM Row-Hammering Attacks,” *Microelectronics Reliability*, 2019.

[30] G. Pardo, “Industrial IoT needs data security,” http://www.eetimes.com/author.asp?section_id=36&doc_id=1329480, accessed: 2016-09-28.

[31] S. Kerckhof, F. Durvaux, C. Hocquet, D. Bol, and F.-X. Standaert, *Towards Green Cryptography: A Comparison of Lightweight Ciphers from the Energy Viewpoint*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 390–407.

[32] M. Volkmer and S. Wallner, “Tree parity machine rekeying architectures,” *IEEE Transactions on Computers*, vol. 54, no. 4, pp. 421–427, Apr. 2005.

[33] A. Ruttor, W. Kinzel, L. Shacham, and I. Kanter, “Neural cryptography with feedback,” *Physical Review E*, vol. 69, no. 4, p. 46110, Apr. 2004.

[34] S. Muhlbach and S. Wallner, “Secure and Authenticated Communication in Chip-Level Microcomputer Bus Systems with Tree Parity Machines,” in *2007 International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation*. IEEE, Jul. 2007, pp. 201–208.

[35] S. Mühlbach and S. Wallner, “Secure communication in microcomputer bus systems for embedded devices,” *Journal of Systems Architecture*, vol. 54, no. 11, pp. 1065–1076, 2008.

[36] I. Kanter and W. Kinzel, “Cryptography based on neural networks - analytical results,” *Journal of Physics A: Mathematical and General*, vol. 35, no. 47, pp. 1–4, 2002.

[37] J.-w. Lee, D. H. T. Vo, Q.-h. Huynh, and S. H. Hong, “A Fully Integrated HF-Band Passive RFID Tag IC Using 0.18-um CMOS Technology for Low-Cost Security Applications,” *IEEE Transactions on Industrial Electronics*, vol. 58, no. 6, pp. 2531–2540, 2011.

[38] H. Houssain, M. Badra, and T. F. Al-Somani, “Hardware implementations of elliptic curve cryptography in wireless sensor networks,” in *Internet Technology and Secured Transactions (ICITST), 2011 International Conference for*, Dec 2011, pp. 1–6.

[39] J. P. J. Janhunen, K. Mikhaylov and M. Sonkki, “Wireless Energy Transfer Powered Wireless Sensor Node for Green IoT: Design, Implementation and Evaluation,” *Sensors*, vol. 19, 2018.

[40] J. Blanco, “Implementación sobre FPGA de un algoritmo de intercambio de llave simétrica basado en redes neuronales,” Project Degree Universidad Industrial de Santander, 2015.

[41] J. Manyika *et al.*, “The Internet of Things: Mapping the Value Beyond the Hype,” *McKinsey Global Institute*, June 2015.

[42] S. K. Mathew, D. Johnston, S. Satpathy, V. Suresh, P. Newman, M. A. Anders, H. Kaul, A. Agarwal, S. K. Hsu, G. Chen, and R. K. Krishnamurthy, “μRNG: A 300–950 mV, 323 Gbps/W All-Digital Full-Entropy True Random Number Generator in 14 nm FinFET CMOS,” *IEEE Journal of Solid-State Circuits*, vol. 51, no. 7, pp. 1695–1704, July 2016.

[43] S. K. Mathew, S. Srinivasan, M. A. Anders, H. Kaul, S. K. Hsu, F. Sheikh, A. Agarwal, S. Satpathy, and R. K. Krishnamurthy, “2.4 Gbps, 7 mW All-Digital PVT-Variation Tolerant True Random Number Generator for 45 nm CMOS High-Performance Microprocessors,” *IEEE Journal of Solid-State Circuits*, vol. 47, no. 11, pp. 2807–2821, Nov 2012.

[44] P. Dabal and R. Pelka, “An Efficient Post-processing Method for Pipelined Pseudo-random Number Generator in SoC-FPGA,” in *2015 22nd International Conference Mixed Design of Integrated Circuits Systems (MIXDES)*, June 2015, pp. 607–611.

[45] S. Wolfram, “Cellular Automata as Models of Complexity,” *Nature*, vol. 311, pp. 419–424, 1984.

[46] K. Salman, “ANALYSIS OF ELEMENTARY CELLULAR AUTOMATA CHAOTIC RULES BEHAVIOR,” *International Journal of Security, Privacy and Trust Management (IJSPTM)*, vol. 2, no. 6, pp. 1–22, 2013.

[47] K. Yang, D. Fick, M. B. Henry, Y. Lee, D. Blaauw, and D. Sylvester, “16.3 A 23Mb/s 23pJ/b Fully Synthesized True-Random-Number Generator in 28nm and 65nm CMOS,” in *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, Feb 2014, pp. 280–281.

[48] L. E. Bassham *et al.*, “A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications,” Tech. Rep. April, 2010.

[49] D. Mocanu, A. Gheolbanoiu, R. Hobincu, and L. Petrica, “Global Feedback Self-programmable Cellular Automaton Random Number Generator,” vol. 39, pp. 1–9, 04 2016.

[50] Q. Tang, B. Kim, Y. Lao, K. K. Parhi, and C. H. Kim, “True random number generator circuits based on single- and multi-phase beat frequency detection,” in *Proceedings of the IEEE 2014 Custom Integrated Circuits Conference*, Sept 2014, pp. 1–4.

[51] E. Chattopadhyay and D. Zuckerman, “Explicit Two-Source Extractors and Resilient Functions,” in *Electronic Colloquium on Computational Complexity*, no. 119, Mar 2016.

[52] P. Lacharme, “Post-Processing Functions for a Biased Physical Random Number Generator,” in *Fast Software Encryption*, K. Nyberg, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 334–342.

[53] A. Cherkaoui, V. Fischer, A. Aubert, and L. Fesquet, “Comparison of Self-Timed Ring and Inverter Ring Oscillators as Entropy Sources in FPGAs,” in *Design Automation and Test in Europe (DATE 2012)*, no. 11.4_2, Dresden, Germany, Mar. 2012, pp. 1–6. [Online]. Available: https://hal-ujm.archives-ouvertes.fr/ujm-00667639

[54] H. Ju, Y. Kim, Y. Jeon, and J. Kim, “Implementation of a Hardware Security Chip for Mobile Devices,” *IEEE Transactions on Consumer Electronics*, vol. 61, no. 4, pp. 500–506, November 2015.

[55] Y. Shi, X. Wang, and H. Fan, “Light-Weight White-Box Encryption Scheme with Random Padding for Wearable Consumer Electronic Devices,” *IEEE Transactions on Consumer Electronics*, vol. 63, no. 1, pp. 44–52, February 2017.

[56] A. Sengupta and S. Bhadauria, “Exploring Low Cost Optimal Watermark for Reusable IP Cores During High Level Synthesis,” *IEEE Access*, vol. 4, pp. 2198–2215, 2016.

[57] A. Sengupta, D. Roy, S. P. Mohanty, and P. Corcoran, “DSP Design Protection in CE through Algorithmic Transformation based Structural Obfuscation,” *IEEE Transactions on Consumer Electronics*, vol. 63, no. 4, pp. 467–476, November 2017.

[58] A. Sengupta, D. Kachave, and D. Roy, “Low Cost Functional Obfuscation of Reusable IP Cores used in CE Hardware through Robust Locking,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, pp. 1–1, 2018.

[59] K. Nohl and Starbug, “Silicon Chips: No More Secrets,” in *11th PACific SECurity annual conference (PACSEC)*, 2009, pp. 1–27.

[60] S. ho Park, J. Jeong, and T. Kwon, "Contents Distribution System based on MPEG-4 IS-MACryp in IP Set-Top Box Environments," *IEEE Transactions on Consumer Electronics*, vol. 52, no. 2, pp. 660–668, May 2006.

[61] J. A. Roy, F. Koushanfar, and I. L. Markov, "EPIC: Ending Piracy of Integrated Circuits," in *2008 Design, Automation and Test in Europe*, March 2008, pp. 1069–1074.

[62] R. P. Cocchi, L. W. Chow, J. P. Baukus, and B. J. Wang, "Method and Apparatus for Camouflaging a Standard Cell Based Integrated Circuit with Micro Circuits and Post Processing," Jun. 7 2012, uS 8510700B2. [Online]. Available: https://patents.google.com/patent/US8510700

[63] T. Iizuka, M. Ikeda, and K. Asada, "Exact Minimum-Midth Multi-Row Transistor Placement for Dual and Non-Dual CMOS Cells," in *2006 IEEE International Symposium on Circuits and Systems*, May 2006, pp. 4 pp.–.

[64] F. Brglez, D. Bryan, and K. Kozminski, "Combinational Profiles of Sequential Benchmark Circuits," in *IEEE International Symposium on Circuits and Systems,*, May 1989, pp. 1929–1934 vol.3.

[65] X. T. Ng, Z. Naj, S. Bhasin, D. B. Roy, J. L. Danger, and S. Guilley, "Integrated Sensor: A Backdoor for Hardware Trojan Insertions?" in *2015 Euromicro Conference on Digital System Design*, Aug 2015, pp. 415–422.

[66] C.-W. Chiou, C.-Y. Lee, and J.-M. Lin, "Finite Field Polynomial Multiplier with Linear Feedback Shift Register," *Tamkang Journal of Science and Engineering*, vol. 10, 12 2004.

[67] Y. Zhang, K. Yang, M. Saligane, D. Blaauw, and D. Sylvester, "A Compact 446 Gbps/W AES Accelerator for Mobile SoC and IoT in 40nm," in *2016 IEEE Symposium on VLSI Circuits (VLSI-Circuits)*, June 2016, pp. 1–2.

[68] S. K. Mathew, D. Johnston, S. Satpathy, V. Suresh, P. Newman, M. A. Anders, H. Kaul, A. Agarwal, S. K. Hsu, G. Chen, and R. K. Krishnamurthy, "uRNG: A 300-950 mV, 323 Gbps/W All-Digital Full-Entropy True Random Number Generator in 14 nm FinFET CMOS," *IEEE J. Solid-State Circuits*, vol. 51, no. 7, pp. 1695–1704, July 2016.

[69] C. Duran, L. E. R. G., A. Amaya, R. Torres, J. Ardila, L. R. D., G. Castillo, A. Agudelo, C. Rojas, L. Chaparro, H. Hurtado, J. Romero, W. Ramirez, H. Gomez, H. Hernandez, and

E. Roa, “A System-On-Chip Platform for the Internet of Things Featuring a 32-bit RISC-V Based Microcontroller,” in *2017 IEEE 8th Latin American Symposium on Circuits Systems (LASCAS)*, Feb 2017, pp. 1–4.

[70] M. Schobert. (2012) Reverse engineering integrated circuits with degate. [Online]. Available: https://degate.org

[71] R. S. Chakraborty and S. Bhunia, “Security Against Hardware Trojan Attacks Using Key-Based Design Obfuscation,” *Journal of Electronic Testing*, vol. 27, no. 6, pp. 767–785, Dec 2011. [Online]. Available: https://doi.org/10.1007/s10836-011-5255-2

[72] N. E. C. Akkaya, B. Erbagci, and K. Mai, “A Secure Camouflaged Logic Family Using Post-Manufacturing Programming with a 3.6GHz Adder Prototype in 65nm CMOS at 1V Nominal VDD,” in *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, Feb 2018, pp. 128–130.

[73] H. Y. Yang, C. M. Chang, M. C. T. Chao, R. F. Huang, and S. C. Lin, “Testing methodology of embedded drams,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 9, pp. 1715–1728, Sept 2012.

[74] Y. Sfikas and Y. Tsiatouhas, “Testing neighbouring cell leakage and transition induced faults in drams,” *IEEE Transactions on Computers*, vol. 65, no. 7, pp. 2339–2345, July 2016.

[75] M. White, J. Qin, and J. B. Bernstein, “A study of scaling effects on dram reliability,” in *2011 Proceedings - Annual Reliability and Maintainability Symposium*, Jan 2011, pp. 1–6.

[76] M. Lanteigne, “ How Row-Hammer Could Be Used to Exploit Weaknesses in Computer Hardware,” 2016.

[77] D. Gruss, C. Maurice, and S. Mangard, “Rowhammer.js: A Remote Software-Induced Fault Attack in JavaScript,” in *arXiv*, 2016.

[78] D. Goodin, “DRAM Bitflipping Exploits that Hijack Computers just Got Easier.” [Online]. Available: http://arstechnica.com/security/2016/04/dram-bitflipping-exploits-that-hijack-computers-just-got-easier/

[79] K. C. H. et. al, “A High-Performance, High-Density 28nm eDRAM Technology with High-K Metal-Gate,” in *Electron Devices Meeting (IEDM), 2011 IEEE International*, Dec 2011, pp. 24.7.1–24.7.4.

[80] S. K. h. Fung et. al, “65nm SOI CMOS Technology for High Performance Microprocessor Application,” in *2006 International Symposium on VLSI Technology, Systems, and Applications*, April 2006, pp. 1–2.

[81] Y. H. Gong and S. W. Chung, “Exploiting refresh effect of dram read operations: A practical approach to low-power refresh,” *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1507–1517, May 2016.

[82] I. Bhati, M. T. Chang, Z. Chishti, S. L. Lu, and B. Jacob, “Dram refresh mechanisms, penalties, and trade-offs,” *IEEE Transactions on Computers*, vol. 65, no. 1, pp. 108–121, Jan 2016.

[83] J. Hong and S. Kim, “Flexible ecc management for low-cost transient error protection of last-level caches,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 6, pp. 2152–2164, June 2016.

[84] Y. Riho and K. Nakazato, “Partial access mode: New method for reducing power consumption of dynamic random access memory,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 7, pp. 1461–1469, July 2014.

[85] S. M. Seyedzadeh, A. K. Jones, and R. Melhem, “Counter-based tree structure for row hammering mitigation in dram,” *IEEE Computer Architecture Letters*, vol. 16, no. 1, pp. 18–21, Jan 2017.

[86] P. Heydari and M. Pedram, “Analysis and reduction of capacitive coupling noise in high-speed VLSI circuits,” in *Computer Design, 2001. ICCD 2001. Proceedings. 2001 International Conference on*, 2001, pp. 104–109.

[87] “Double Data Rate(DDR) SDRAM Specification,” JEDEC Solid State Technology Association, Virginia, USA, Standard, Mar. 2003.

[88] Y. H. Gong and S. W. Chung, “Exploiting refresh effect of dram read operations: A practical approach to low-power refresh,” *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1507–1517, May 2016.

[89] D. Wang, “Why migrate to DDR4?” [Online]. Available: http://www.eetimes.com/document.asp

[90] M. Ding, Z. Zhou, Y. Liu, S. Traferro, C. Bachmann, K. Philips, and F. Sebastiano, “A 0.7-V 0.43-pJ/cycle Wakeup Timer Based on a Bang-Bang Digital-Intensive Frequency-Locked-Loop for IoT Applications,” *IEEE Solid-State Circuits Letters*, vol. 1, no. 2, pp. 30–33, Feb 2018.

[91] A. Savanth, J. Myers, A. Weddell, D. Flynn, and B. Al-Hashimi, “A 0.68nW/kHz Supply-Independent Relaxation Oscillator With $\pm$0.49%/V and 96ppm/°C Stability,” in *2017 IEEE International Solid-State Circuits Conference (ISSCC)*, Feb 2017, pp. 96–97.

[92] Ç. Gürleyük, L. Pedala, F. Sebastiano, and K. A. A. Makinwa, “A CMOS Dual-RC Frequency Reference With $\pm$250ppm Inaccuracy From -45°C to 85°C,” in *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, Feb 2018, pp. 54–56.

[93] Y. Tokunaga, S. Sakiyama, A. Matsumoto, and S. Dosho, “An On-Chip CMOS Relaxation Oscillator With Power Averaging Feedback Using a Reference Proportional to Supply Voltage,” in *2009 IEEE International Solid-State Circuits Conference - Digest of Technical Papers*, Feb 2009, pp. 404–405,405a.

[94] Y. Cao, P. Leroux, W. D. Cock, and M. Steyaert, “A 63,000 Q-factor Relaxation Oscillator With Switched-Capacitor Integrated Error Feedback,” in *2013 IEEE International Solid-State Circuits Conference Digest of Technical Papers*, Feb 2013, pp. 186–187.

BIBLIOGRAPHY

ABAYANEH, M. R. S. (2012). Security analysis of lightweight schemes for rfid systems (University of Bergen).

AUER, M, EHRLICH, E., MISSONI, A., KARGL, W., HOLWEG, G., & PRIBYL, W. (2008). Design and development of a mixed signal prototyping system to achieve very high data rates for contactless applications. E & i Elektrotechnik Und Informationstechnik, 125(4), 138–142.

AUER, MARKUS, MISSONI, A., & KARGL, W. (2010). HF RFID transponder with phase demodulator for very high bit-rates up to 13.56 Mbit/s. 2010 IEEE International Conference on RFID (IEEE RFID 2010), 69–76.

BO, F., YUJIE, D., XIAOXING, Z., & YINGJIE, L. (2009). Low power clock recovery circuit for passive HF RFID tag. Analog Integrated Circuits and Signal Processing, 59(2), 207–214.

ENGELHARDT, M., PFEIFFER, F., FINKENZELLER, K., & BIEBL, E. (2013). Extending ISO/IEC 14443 Type A Eavesdropping Range using Higher Harmonics. Smart Objects, Systems and Technologies (SmartSysTech), Proceedings of 2013 European Conference On, 1–8.

FELDHOFER, M., AIGNER, M., BAIER, T., HUTTER, M., PLOS, T., & WENGER, E. (2010). Semi-passive RFID development platform for implementing and attacking security tags. Internet Technology and Secured Transactions (ICITST), 2010 International Conference For, 1–6.

FINKENZELLER, K. (2010). RFID HANDBOOK. Great Britain: Wiley.

HANCKE, G. (N.D.). A Practical Relay Attack on ISO 14443 Proximity Cards. (January 2005), 1–13.

INTERNATIONAL STANDARD ISO/IEC 9798. (2010). Part 1: Information technology -Security techniques - Entity authentication.

KANTER, I., & KINZEL, W. (2002). Cryptography based on neural networks - analytical results. Journal of Physics A: Mathematical and General, 35(47), 1–4.

KI, T., KIM, H., CHUNG, C., KIM, Y., BAE, K., & KIM, J. (2013). Design of a Low-power Digital Processor for a Security Passive RFID Tag. 5450–5454.

LEE, J., VO, D. H. T., HUYNH, Q., & HONG, S. H. (2011). A Fully Integrated HF-Band Passive RFID Tag IC Using 0.18-um CMOS Technology for Low-Cost Security Applications. IEEE Transactions on Industrial Electronics, 58(6), 2531–2540.

MAN, A. S. W., ZHANG, E. S., CHAN, H. T., LAU, V. K. N., TSUI, C. Y., & LUONG, H. C. C. (2007). Design and Implementation of a Low-power Baseband-system for RFID Tag. 1585–1588.

MAN, A. S. W., ZHANG, E. S., LAU, V. K. N., TSUI, C. Y., & LUONG, H. C. (N.D.). Low Power VLSI Design for a RFID Passive Tag baseband System Enhanced with an AES Cryptography Engine.

MENEZES, A. J., OORSCHOT, P. C. VAN, & VANSTONE, S. A. (1997). Handbook of Applied Cryptography.

MERIAC, M. (2010). Heart of Darkness - exploring the uncharted backwaters of HID iCLASS security. 27TH CHAOS COMMUNICATION CONGRESS, (December), 1–13.

MUHLBACH, S., & WALLNER, S. (2008). Secure communication in microcomputer bus systems for embedded devices. Journal of Systems Architecture, 54(11), 1065–1076.

PEBAY-PEYROULA, F., & REVERDY, J. (2011). A true full-duplex communication between HF contactless reader and card. 2011 IEEE International Conference on RFID-Technologies and Applications, 473–478.

PERIS-LOPEZ, P., HERNANDEZ-CASTRO, J. C., ESTEVEZ-TAPIADOR, J. M., & RIBAGORDA, A. (2006). RFID Systems: A Survey on Security Threats and Proposed Solutions. In Personal Wireless Communications (pp. 159–170). Springer.

PFEIFFER, F., FINKENZELLER, K., & BIEBL, E. (2012). Theoretical Limits of ISO/IEC 14443 type A RFID Eavesdropping Attacks. Smart Objects, Systems and Technologies (SmartSysTech), Proceedings of 2012 European Conference On, 1–9.

REYES, O. M., & ZIMMERMANN, K.-H. (2010). Permutation parity machines for neural cryptography. Physical Review E, 81(6), 066117.

RFID Security. (2008). In Kitsos Paris and Zhang Yan (Ed.), Vasa.

RUTTOR, A., KINZEL, W., & KANTER, I. (2005). Neural cryptography with queries [Disordered Systems and Neural Networks]. Journal of Statistical Mechanics: Theory and Experiment, 2005(01), P01009.

RUTTOR, A., KINZEL, W., SHACHAM, L., & KANTER, I. (2004). Neural cryptography with feedback [Disordered Systems and Neural Networks]. Physical Review E, 69(4), 46110.

SHI, W., & CHOY, C. (2012). A Process-Compatible Passive RFID Tag's Digital Design for Subthreshold Operation. 528–531.

VOLKMER, M., & WALLNER, S. (2005). A Key Establishment IP-Core for Ubiquitous Computing. 16th International Workshop on Database and Expert Systems Applications (DEXA'05), 241–245.

VOLKMER, M., & WALLNER, S. (2005). Tree parity machine rekeying architectures. Computers, IEEE Transactions On, 54(4), 421–427.

VOLKMER, MARKUS, & SCHAUMBURG, A. (2004). Authenticated tree parity machine key exchange [Cryptography and Security; Disordered Systems and Neural Networks]. CoRR, cs.CR/0408, 1–11.

VOLKMER, MARKUS, & WALLNER, S. (2005a). Lightweight Key Exchange and Stream Cipher based solely on Tree Parity Machines. IACR Eprint Archive.

VOLKMER, MARKUS, & WALLNER, S. (2005b). Tree Parity Machines Rekeying Architectures for Embedded Security. In IACR Eprint archive.

WANG, J., LI, H., & YU, F. (2007). Design of Secure and Low-Cost RFID Tag Baseband. 2066–2069.

WITSCHNIG, H., PATAUNER, C., MAIER, A., LEITGEB, E., & RINNER, D. (2007). High speed RFID lab-scaled prototype at the frequency of 13.56 MHz. E & i Elektrotechnik Und Informationstechnik, 124(11), 376–383.

XUECHENG, Z., HUAN, L., HUI, L., DONGSHENG, L., LIANG, G., & KE, Y. (2014). Design and implementation of an analog front-end circuit for semi-passive HF RFID tag. 2014 IEEE Radio Frequency Integrated Circuits Symposium, 389–392.

APPENDIX

Appendix A. A 32.768kHz-to-1MHz RC-based Oscillator

This appendix describes a 32.768kHz-to-1MHz RC-based oscillator (RCO) suitable for high- and low-duty-cycle sleep-mode timers in emerging sensor node applications. In contrast to crystal oscillators (XO) based frequency references which employ long and energy intensive restarts, the proposed RCO achieves a 1000X reduction in restarting time. Measurement shows that the implemented RCO achieves a temperature stability of 68.5ppm/°C @32.768kHz and 37.5ppm/°C @1MHz. A proposed autonomous digital scheme uses coarse and fine tuning to compensate for process and temperature variations. Measurements report an energy efficiency of 1.46nW/kHz@1MHz occupying an area of 0.055mm2 in a pure digital 180nm CMOS process node.

Introduction

Emerging IoT sensor nodes demand low-power wake-up timers that perform system duty cycling so that, during long sleeping periods, the system power decreases. Systems usually employ a real-time counter (RTC) XO as an always-on (AON) frequency reference for the wake-up timers. Although off-chip XO offers superior frequency stability, area-constrained and battery-powered sensor nodes restrict the adoption of integrated oscillators. In addition, each shutdown of the XO in low-duty-cycle applications is energy intensive, with an approximated consumption of 100mW over a 100ms start-up time [90].

Fig. 1 shows an RCO within the clock distribution of a current system-on-chip (SoC) for IoT sensor nodes. The RCO and an XO provide the low-frequency clocks for the AON domain. Based upon our measurement results for faster start-up times, the RCO may replace the low-frequency XO as a wake-up timer in future applications. Periodic wake-ups enable the processor core to perform computation/communication operations at higher frequencies where even an RCO can be applied. This work proposes the use of the same RCO enabling low-power low-duty-cycle mid-frequency operations by triggering events of the system to check sensor inputs for data rates

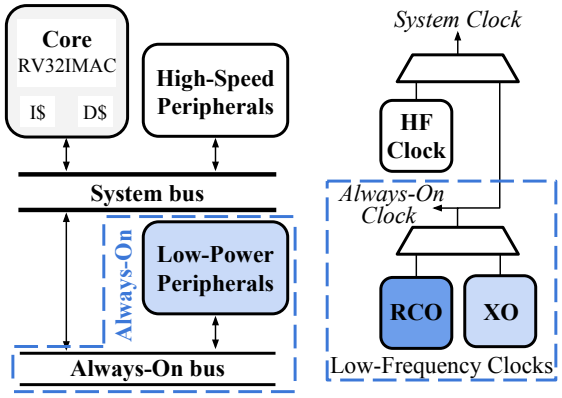


Figure 65. RC relaxation oscillator as reference clock in an always-on domain.

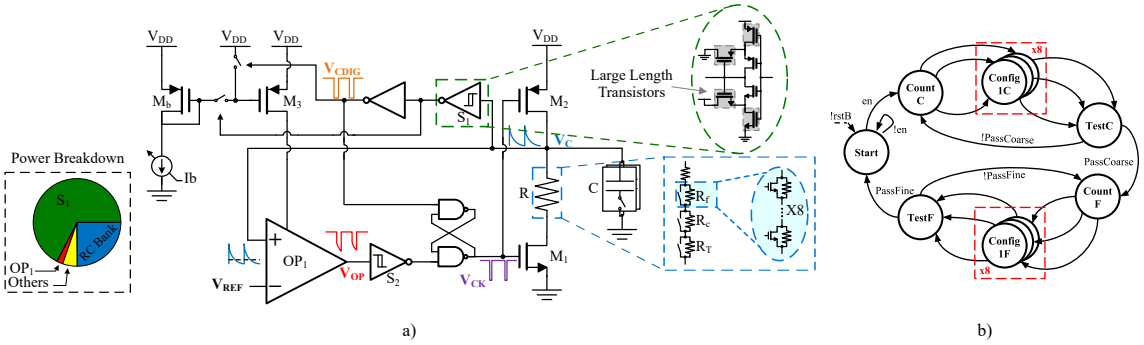


Figure 66. a) Proposed relaxation oscillator architecture b) States diagram of the digital assisted compensation loop.

as low as 1Mb/s.

This low-power wide-band RCO reduces the start-up energy down to three scales lower than the required start-up energy in an XO. The measured RCO can operate at different frequencies ranging from 32.768kHz-to-1MHz, with sub-100ppm temperature coefficients. The proposed circuit has a power control loop with optimized Schmitt trigger inverters to reduce the dominant large DC-current associated with their slow input slope. A wide-band tuning scheme uses two resistor banks for fine and coarse adjustment, plus a capacitor bank to provide a coarser variation. A proposed self-calibration scheme enables the RCO operation over a temperature range from -20°C to 125°C with high temperature and voltage stability.

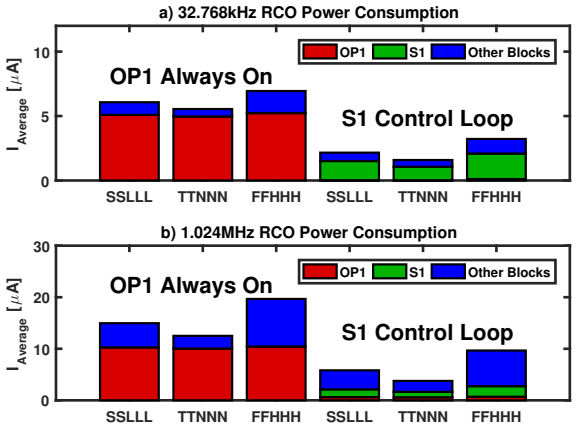


Figure 67. a)Simulation results of power consumption for different blocks at 32.768kHz. b) Simulation results of power consumption of different blocks at 1MHz.

Relaxation Oscillator Architecture and circuit implementation

A regular relaxation oscillator uses an error amplifier to generate a pulse that charges and discharges an RC bank [91–94]. The oscillator frequency depends on the capacitor discharge through the resistance and the comparison time with a voltage reference made by an error amplifier. However, the amplifier is idle during most of the oscillator cycle and comparison operation requires a short time. Thus, the amplifier can be turned-off during most of the idle time and turned-on before the comparison time and, therefore, saving power during almost the whole period [91].

Fig. 66 shows the implemented oscillator with the states diagram of autonomous digital scheme. Most of the reported RCOs are limited to a specific operation frequency, and here we propose a wide-band oscillator operating from 32.768kHz-to-1MHz with reduced start-up time. A power down loop based Schmitt trigger (S_1) controls the on and off error amplifier(OP_1) cycle. In contrast to [91], we propose a fixed Schmitt trigger threshold and a tuning digital scheme that allows faster calibration time.

On the Limits of Power Consumption There are two main circuits that consume most of the total current. The error amplifier (OP_1) and the Schmitt trigger (S_1) that controls the amplifier power-down signal. Due to the control over amplifier operation, the current consumption in this circuit is less than 1μA over all process variations. Hence, the major part of the current corresponds to the Schmitt trigger where the slow slope of the RC discharge creates a large DC

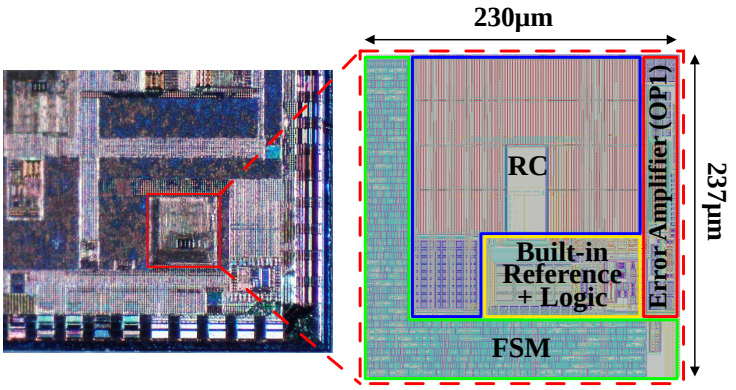


Figure 68. Die micro-photograph and layout details of the proposed RCO.

current.

Savanth et. al. [91] proposes an automatic control for the Schmitt trigger threshold with the aim to optimize the amplifier power consumption. The control requires to estimate the period of OP_1 operation and to adjust a variable Schmitt trigger threshold. They also use the estimations to adjust the frequency with pre-charged values of resistance tuning. In contrast, we optimize the Schmitt trigger performance, sizing it to reduce as much of the associated DC current. In addition, we design a fixed Schmitt trigger threshold for a robust performance over process, temperature and voltage (PVT) variations, ensuring the amplifier is always active for a precise comparison.

Fig. 67 shows a power break-down of the oscillator when OP_1 power control loop operates and when OP_1 is always working. Fig. 67a) and Fig. 67b) show the power break-down at 32.768kHz and 1MHz. For both operation frequencies, OP_1 experiments the higher power consumption when is always on. When the power control loop operates, the OP_1 power consumption is negligible compared to the Schmitt trigger and other blocks. In addition, the behavior is consistent for both operation frequencies and across PVT variations.

For a further power consumption optimization, the implemented amplifier has a bandwidth control feature. As Fig. 66a) shows, the digital word Ib varies the bias current turning-on and off some current mirrors. This digital trimming enables a current consumption of less than 1μA in low-frequency operation (kHz) but it also enables a higher frequency operation with reduced propagation delay when using the extra current bias.

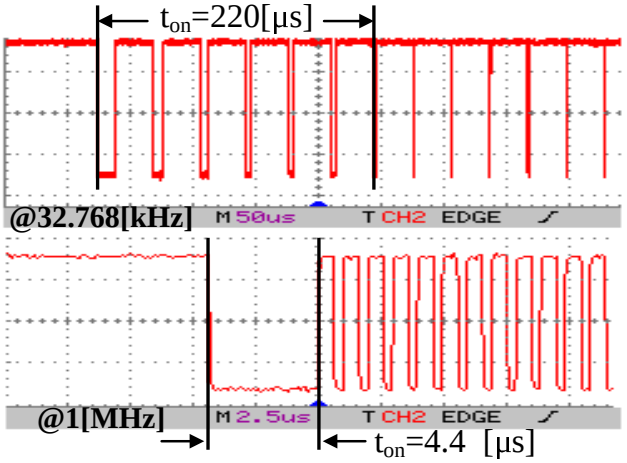


Figure 69. Settling time (start-up) measurement until RCO reaches its steady state.

RC Bank Tuning and Wide-Band Operation RCO output frequency relies on the time constant $\tau = RC$. However, resistance and capacitance variations across process and temperature change the RC time constant –and so the output frequency–, also affecting the temperature stability [94]. The proposed RC bank feature three types of tuning, including a temperature coefficient tuning.

Fig. 66a) shows the three resistance banks where R_f, R_c and R_T correspond to fine tuning, coarse tuning and temperature tuning . Two 8 thermometer buses adjust the coarse and fine tuning and an extra resistance control provides the temperature tuning. R_f and R_c banks are poly resistors and R_T are well resistors to help with temperature compensation. Temperature compensation is a major issue considering the resistance bank change over the frequency range. We address this issue by tuning the well resistor bank as well.

Considering that the implementation is performed in a pure digital process flavor, MiM-capacitors are not provided. We implement a C bank with NMOS capacitors occupying a lower area at the expense of increased sensitivity to process variations and leakage. In addition, a capacitance control enables a higher frequency range with only a coarse resistance step. Hence, two control bits provide different operation frequencies with a minimum value of 300fF for the MHz range.

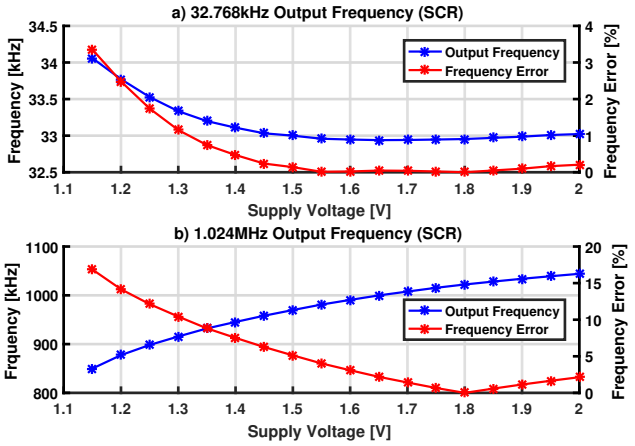


Figure 70. a) 32.768kHz free-running frequency against supply voltage. b) 1MHz free-running frequency against supply voltage.

Measurement Results

Supply Voltage Robustness The discharge time must be constant to ensure an invariant frequency. If OP_1 compares with a constant reference under supply variations, the discharge time will change due to the comparison can occur after or before the expected time [91]. Thus, the ideal voltage reference should be invariant to temperature and process but not to supply variations.

We use a switched capacitor based reference (SCR) which emulates a resistor voltage divider [91]. The advantage is that an SCR has a better temperature coefficient, occupies less area and avoids any static current consumption. Moreover, we add a diode reference (DREF) and an external reference (Bandgap) as alternative options for debugging purposes.

Self-Calibration Feature Savanth et al. [91] uses a calibration method with a pre-loaded tuning. A counter estimates the time that V_{CDIG} signal lasts in zero with a reference clock based on a ring oscillator. The calibration system changes the resistance tuning and the Schmitt trigger threshold if the count is out of a predefined-range. In contrast, we propose an FSM to calibrate for process and temperature variations counting over the whole cycle of the oscillator. Fig 66b) shows the states diagram where the reference clock comes from a starved-inverter based oscillator. The proposed calibration performs an initial count and then adjusts the coarse resistance bank. Once the count fits within a pre-defined range, the FSM switches to the fine resistance bank and

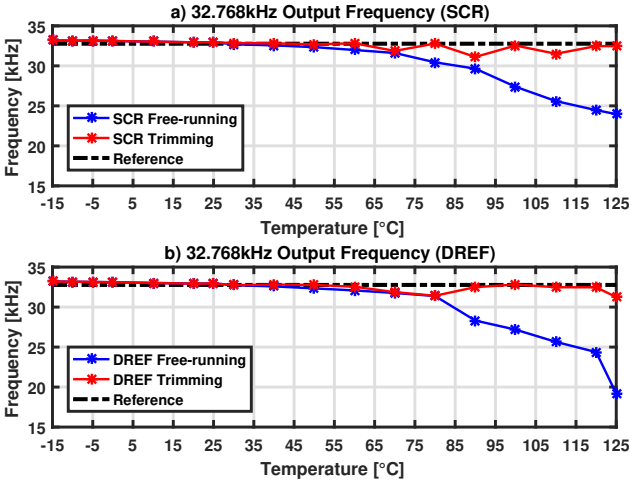


Figure 71. a) SCR measured results for 32.768kHz over temperature. b) DREF measured results for 32.768kHz over temperature.

performs a similar process. The pre-defined range depends on the desired frequency, so we can adjust them according to the targeted operation frequency. In addition, capacitance tuning provides the coarsest step, with a fixed value for a given frequency.

The RCO is implemented in a pure logic 180nm CMOS technology node. Fig. 68 shows the die micro-photograph and annotated layout of the proposed oscillator. Fig 69 illustrates the start-up transient behavior of the implemented oscillator, showing a settling time of 200μs@32.768kHz and 4.4μs@1MHz. This fast settling time guarantees a low start-up energy of 72pJ and 6pJ. In addition, the results show a 1000X reduction in restarting time compared to traditional XO circuits. This feature enables low-power low-duty-cycle mid-frequency operations at data rates as low as 1Mb/s. Although the RCO frequency is adjustable to operate between 32.768kHz and 1MHz, we report measurement results for voltage and temperature stability for these extreme frequencies to summarize performance.

Voltage Stability Unlike Savanth in [91] that implemented variable threshold Schmitt trigger, we implement robust Schmitt trigger inverters which high-to-low threshold is always greater than reference voltage over PVT variations. Results or show similar voltage stability performance with reduced complexity. Fig. 70 shows the results for 32.768kHz and 1.768MHz varying the voltage supply from 1.1 to 2 V. Fig 70a) shows the stability for 32.768kHz where frequency error is less than $\pm$1% for supply voltages over 1.3V. For supply values less than 1.3 V the SCR reference

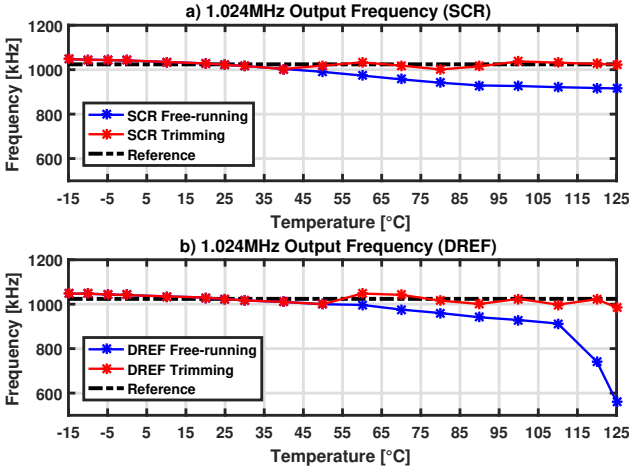


Figure 72. a) SCR measured results for 1MHz over temperature. b) DREF measured results for 1MHz over temperature.

loses the proportionality, hence, increasing the frequency variation. Fig. 70b) shows a similar behavior for 1MHz but with higher variation due to the optimization of the threshold at 32.768kHz.

Temperature Stability As expected, variations over temperature are larger than a mim-cap based RC bank since we based our implementation on a pure standard logic node. We measured oscillator performance from -20 to 125 °C for 32.768kHz and 1MHz using SCR and DREF references. Fig. 71 shows the performance at 32.768kHz at free running and after frequency tuning. The increased sensitivity to temperature mitigates with the tuning feature. Frequency error at high temperatures reduces after tuning, resulting in an average error less than 0.96 %. As expected, DREF reference presents a higher sensitivity under temperature variations.

Fig.72 shows the performance at 1MHz for SCR and DREF references. At 1MHz, operation with DREF reference at high temperature presents a stronger sensitivity than at 32.768kHz. Finally, the tuning feature can reduce high temperature variations to less than 0.5% with SCR reference and 0.9% with DREF reference.

Performance Comparison Table 24 shows a summary of measurement results and a comparison of the proposed oscillator to some state-of-the-art circuits. This work achieves 68.57ppm/°C in temperature stability with frequency tuning and 0.5% voltage stability without any adjustment at 32.768kHz. The energy efficiency $(Power/Freq)$ at 1MHz is 1.46nW/kHz, which is

Table 24. Performance summary and comparison.

	[91]	[92]	[93]	[94]	This Work
Tech (nm)	65	180	180	65	180
Power(μW)	0.92* 12**	750	43.2	98.4	**0.33^{+}@32.768kHz 1.5@$^{+}$1MHz**
Freq(kHz)	1350	8000	14000	12600	**32.768–1024**
Area (mm^2)	0.005	1.594	0.04	0.01	0.05
Start-up Time (μs)	10	N./A.	N./A.	N./A.	**220@32.768kHz 4.4@1MHz**
Start-up Energy (pJ)	120**	N./A.	N./A.	N./A.	**72.64@32.768kHz 6.6@1MHz**
T. Range (°C)	0 to 150	-45 to 85	-40 to 125	0 to 80	-20 to 125
T. Coeff (ppm/°C)	96	3.85	1900	8200	68.57@32.768kHz 35.714@1MHz
V. Range	0.9 to 1.9	1.7 to 2	1.7 to 1.9	1.1 to 1.5	1.1 to 2
V.Stab (%△T/V)	0.49	0.18	0.16	0.07	0.5@32.768kHz 10.23@1MHz
FOM (nW/kHz)	0.68* 8.89**	93.75	3.08	7.8	**10.07^{+}@32.768kHz 1.46^{+}@1MHz**
Start-up FOM (pJ/kHz)	0.09**	N./A.	N./A.	N./A.	**2.21^{+}@32.768kHz 0.006^{+}@1MHz**

* Measured at 0.9V ** Measured at 1.4V $^{+}$ Measured at 1.1V

the second best after the reported by [91]. Energy efficiency is not well scaled with frequency due to the lower slope at the Schmitt trigger input, however, measurement results show an efficiency of 10.07nW/kHz at 32.768kHz.

Few papers in the literature report the start-up time of the RCO. This feature is important since a fast and energy efficient start-up enables low-energy duty-cycle applications by replacing the slower XO circuits. The conventional figure of merit (FOM) is not able to compare performance in terms of the start-up energy and settling time. Therefore, we include the start-up time complementing the original FOM as equation (4) shows. Using this new FOM, we obtain a performance of 2.1pJ/kHz @32.768kHz and 6fJ/kHz @1MHz, being the best reported result considering the performance comparison of Table 24.

$$FOM = \frac{Power[nW] \cdot Startup[ms]}{Freq[kHz]} \tag{4}$$

Summary

This work demonstrates a wide-band RCO with a fast and energy efficient start-up. The oscillator includes a Schmitt trigger-based amplifier power control loop and a bandwidth control to optimize power consumption at different operation frequencies. Measurement results show an energy efficiency of 10.07nW/kHz@33kHz and 1.46nW/kHz@1MHz; as well as 2.1nW $\cdot$ ms/kHz@32.768kHz and 0.006nW $\cdot$ ms/kHz@1MHz start-up energy efficiency. The results show a restarting time of

220μs@33kHz and 4.4μs@1MHz, which is 1000X less than the required of XO circuits, only consuming 72pJ and 6pJ energy at start-up. The measured start-up energies enable the application of the proposed RCO in high- and low-duty-cycle sensor node applications. The circuit was fabricated in a pure digital 180nm technology occupying an area of 0.055mm^2 and achieving a temperature stability of 68.5ppm/$^\circ$C @32.768kHz and 37.5ppm/$^\circ$C @1MHz with the autonomous digital scheme enabled.