

PHISIC 2018
Workshop on Practical Hardware
Innovation in Security and Characterization

May 23-24th 2018
Campus Georges Charpak Provence
phisc2018.emse.fr

PHISIC'2018 – ABSTRACTS

// KEYNOTE // DEEP LEARNING FOR EMBEDDED SECURITY EVALUATION

Emmanuel Prouff // ANSSI

To provide insurance on the resistance of a system against side-channel analysis, several national or private schemes are today promoting an evaluation strategy, common in classical cryptography, which is focusing on the most powerful adversary who may train to learn about the dependency between the device behavior and the sensitive data values. Several works have shown that this kind of analysis, known as Template Attacks in the side-channel domain, can be rephrased as a classical Machine Learning classification problem with learning phase.

Following the current trend in the latter area, recent works have demonstrated that deep learning algorithms were very efficient to conduct security evaluations of embedded systems and had many advantages compared to the other methods. During the proposed presentation, I will come back on these recent works and will identify some avenues for further research on this topic.

1. SCATTER: A NEW DIMENSION IN SIDE-CHANNEL

Benoit Feix // eSHARD

Side-channel techniques have been progressing over the last few years, leading to the creation of a variety of statistical tools, aiming at extracting secrets handled in cryptographic algorithms. In the same time Integrated Circuit manufacturers and final product developers were improving the strength of their countermeasures. In particular their products were facing more and more enhanced attacks during security certifications.

Noticeably, the vast majority of side-channel techniques requires to get the traces aligned together prior to applying statistics. This prerequisite turns out to be challenging in the practical realization of attacks as implementations tend to include hardware or software countermeasures to increase this difficulty. This can be typically achieved by adding random jitters, random clock dividers, or random executions with fake operations, etc.

In this paper, we introduce the new side-channel technique scatter, whose potential is to tackle alignment issues. By construction, scatter brings an additional dimension and opens the door to a large set of potential new attack techniques.

The effectiveness of scatter has been proven on both simulated traces and real world secure products. In summary scatter is a new side-channel technique offering a valuable alternative when the trace alignment represents an issue. Furthermore, scatter represents a suitable option for low-cost attacks, as the requirements in terms of equipment and expertise are significantly reduced.

We will present during our talk the principles of scatter as well as practical results through live demonstrations.

NB: from the publication: SCATTER: A New Dimension in Side-Channel. Hugues Thiebeauld, Georges Gagnerot, Antoine Wurcker and Christophe Clavier. COSADE 2018. To appear.

2. AUTOMATED SOFTWARE PROTECTION FOR THE MASSES AGAINST SIDE-CHANNEL ATTACKS

Nicolas Belleville // Université Grenoble Alpes, CEA Tech, Damien Courousse, Henri-Pierre Charles // CEA Tech, Karine Heydemann // Sorbonne Universités, LIP6

Side-channel attacks represent a major threat for IOT devices and embedded systems. As the number of such devices is currently fastly growing, there is an urgent need for securing them against side-channel attacks. In this presentation, we will present an automated approach to increase the security of any program against power/EM side channel attacks. The approach is lightweight and can be used with constrained devices. Our approach relies on a combination of static compilation and dynamic code generation in order to make the countermeasure as efficient as possible. The protection is based on code polymorphism, which corresponds to the capability of regularly changing the behaviour of a component at runtime without altering its functional properties. More precisely, several transformations are gathered to make the code vary, such as register shuffling, instruction shuffling, semantic variants and insertion of dummy instructions.

The level of security is completely configurable as each transformation can be enabled/disabled, and as some of them provide a tunable level of variability.

In this talk,

- We present how to use compilation to generate specialized runtime generators of polymorphic code, and how the specialization of such generators helps to mitigate some JIT security issues.
- Then, we describe how the runtime generators make the code vary. We discuss the code transformations involved with filtering/resynchronization attacks in mind.
- We give experimental results on security aspects (CPA and t-test) as well as performance aspects for several configurations. We show that one can reach strong security levels using our approach.

// SESSION // ADVANCED CRYPTOGRAPHIC PRIMITIVES

3. FHE AND MPC PRIVACY PRESERVING COMPUTATIONS IN THE CLOUD

Nicolas Gama // EPFL

In this talk, we briefly introduce *fully homomorphic encryption* (FHE) and *multiparty computations* (MPC), and give an overview of the main constructions, as well as the underlying security assumptions or adversarial models behind these models.

For homomorphic encryption, we explain the original concept introduced by Gentry in 2009 and explain the current state of the art, both in theory (with the underlying Approx-GCD and RingLWE lattice problem), and in practice by introducing a few open source libraries.

For MPC, we briefly present the concept of masking, secret sharing, oblivious transfer and garbled circuits, and illustrate a few examples.

Finally, we compare both approaches, if the final goal is to achieve privacy preserving computations in the cloud.

4. STARTING TRANSITION TOWARDS PRODUCTS EMBEDDING POST-QUANTUM CRYPTOGRAPHY

Yannick Teglia, Aline Gouget // GEMALTO

In 1996, the cryptographers Shor and Grover separately imagined how a quantum computer could defeat some classical cryptographic algorithms and then proposed procedures to do so. Prototypes of such machines exist today and the digital world giants are pouring a lot of money in this field. This speeds up the need for having intrinsically resistant algorithms as emphasized by the recent NIST initiative.

We will do an overview of where we're standing today in term of threats regarding quantum computing against cryptography. We will also discuss about the current projects, initiatives and technical proposals on Post Quantum Cryptography, in the academic world and in the industrial world as well.

Finally we will discuss about the implications on the implementations and the final products.

5. IMPACT OF FREQUENCY LOCKING ON RING OSCILLATING CELLS IN FPGA

Ugo Mureddu, Nathalie Bochart, Lilian Bossuet, Viktor Fischer // Univ Lyon, UJM-Saint-Etienne, CNRS, Laboratoire Hubert Curien UMR 5516

Electronic oscillators are key elements in many applications. They are used in most of digital circuits as time reference for synchronizing operations. In the security field, oscillators serve as entropy source for True Random Number Generator (TRNG) or Physical Unclonable Functions (PUF).

Among all the available structures, CMOS ring oscillating cells are particularly interesting in the context of digital Integrated Circuits (IC) since they are fully digital which make them very easy to integrate. However, a phenomena, very little studied and taken into account, that influence CMOS ring oscillating cells is their ability to easily lock a signal with a frequency close to their natural oscillation frequency. Interaction between two oscillatory systems spatially close operating at frequency which are close to each other is a well know effect. This phenomena is called locking.

Although some particular applications like frequency divider are taking advantage of it, in most cases this is something the designers want to avoid. Indeed, locking on ring oscillating cells used for security applications could be very prejudicial. Let's consider a ring oscillating cells based TRNG used to generate encryption keys for secure communication between two people, keys generated from a locked ring oscillating cell will be fully deterministic and thus completely insecure.

This paper presents a complete study of locking phenomena on the main types of CMOS ring oscillating cells: Ring Oscillator (RO), Transient Effect Ring Oscillator (TERO) and Self Time Ring (STR).

First of all, to the best of our knowledge, it is demonstrated for the first time that locking is affecting specific configurations of ring oscillating cells like TERO and STR. Moreover, while some work has been done on the influence of locking on classical RO, there is at the moment no exhaustive evaluation. That's why, a full study of locking phenomena on different FPGA families is carried. The advantage of carrying this study on FPGA instead of dedicated IC is double. First, it permits to enlarge the scope of experimentation by reconfiguring the FPGAs. Then, the oscillations frequency of studied cells in FPGA is lower than in ASIC and allow us to output them on measurement devices. The three main FPGA families, namely Xilinx, Intel (SRAM based FPGA) and Microsemi (FLASH based FPGA), are targeted in this work.

// SESSION // FAULT INJECTION ANALYSIS: ATTACKS AND PROTECTIONS (I)

6. EXPERIMENTAL COMPARISON AND ANALYSIS OF THE SENSITIVITY TO LASER FAULT INJECTION OF CMOS FD-SOI AND CMOS BULK TECHNOLOGIES

Jean-Max Dutertre // EMSE

Laser illumination may be used to inject faults into the computations of secure ICs for the purpose of retrieving secret data. The CMOS FD-SOI technology is expected to be less sensitive than the more usual CMOS bulk technology to such laser fault injection attacks. We report in this work an experimental assessment of the interest of using FD-SOI rather than CMOS bulk to decrease laser sensitivity. Our experiments were conducted on test chips at the 28 nm node for both technologies with laser pulse durations in the picosecond and nanosecond ranges.

7. LOW-COST SETUP FOR LOCALIZED SEMI-INVASIVE OPTICAL FAULT INJECTION ATTACKS

Michael Gruber, Fabrizio De Santis // Technische Universität München, Oscar M. Guillen // Giesecke & Devrient GmbH Munich

Localized semi-invasive optical fault attacks are nowadays considered to be out of reach for attackers with a limited budget. For this reason, they typically receive lower attention and priority during the security analysis of low-cost devices. Indeed, an optical fault injection setup typically requires expensive equipment which includes at least a laser station, a microscope, and a programmable X-Y table, all of which can quickly add up to several thousand euros. Additionally, a careful handling of toxic chemicals in a protected environment is required to decapsulate the chips under test and gain direct access to the die surface. In this work, we present a low-cost fault injection setup which is capable of producing localized faults in modern 8-bit and 32-bit microcontrollers, does not require handling hazardous substances or wearing protective eyewear, and would set back an attacker only a couple hundred euros. Finally, we show that the type of faults which are obtained from such a low-cost setup can be exploited to successfully attack real-world cryptographic implementations, such that of the NSA's Speck lightweight block cipher.

8. FAULTS ATTACKS ON SoCs

Thomas Troughkine, Guillaume Bouffard // ANSSI, Jessy Clédière // CEA Tech

Systems on Chip (SoC) are complex integrated circuits which contain a many components (processors, controllers...). They are present in a lot of common systems like smartphones or IoT devices. These systems manipulate sensitive data (bank, health or ID). To ensure the confidentiality of these data, several security mechanisms (mostly software) are implemented in them such as Secure Boot, Trusted Environment Execution or Memory Partitioning.

Despite the fact SoCs are assumed to be secured from a software point of view they don't integrate any mitigation against hardware attacks (side-channels and faults). For this presentation, we'll focus on fault injection attacks. Due to this lack of security, several faults based attacks on SoCs are emerging. These attacks explore new attack paths mixing fault injection and software exploitation to undermine security mechanisms.

To execute sensitive operations, the chip designers are looking to certify their SoCs. In order to understand the underlying challenges, this presentation will focus on the issues of the hardware security on SoCs. We will present a modelization of this new attack paths and some remarkable attacks which put forward the complexity of mixing a hardware fault injection with a software exploitation. Our final goal is to be able to evaluate the resistance of SoCs against this new threats.

9. COMPILER-ASSISTED LOOP HARDENING AGAINST FAULT ATTACKS

Julien Proy // INVIA

Secure elements widely used in smartphones, digital consumer electronics, payment systems are subject to fault attacks. To thwart such attacks, software protections are manually inserted requiring experts and time. The explosion of the Internet of Things (IoT) in home, business and public spaces motivates the hardening of a wider class of applications, and the need to offer security solutions to non-experts. This paper addresses the automated protection of loops at compilation time, covering

the widest range of control- and data-flow patterns, in both shape and complexity. The security property we consider is that a sensitive loop must always perform the expected number of iterations, otherwise an attack must be reported. We propose a generic compile-time loop hardening scheme based on the duplication of termination conditions and of the computations involved in the evaluation of such conditions. We also investigate how to preserve the security property along the compilation flow while enabling aggressive optimizations. We implemented this algorithm in LLVM 4.0, at the Intermediate Representation (IR) level in the backend. On average, the compiler automatically hardens 95% of the sensitive loops of typical security benchmarks, and 98% of these loops are shown to be robust to simulated faults. Performance and code size overhead remain quite affordable, at 12.5% and 14% respectively.

// SESSION // SIDE-CHANNEL ANALYSIS: ATTACKS & PROTECTIONS (II)

10. DIVIDING THE THRESHOLD: MULTI-PROBE LOCALIZED EM ANALYSIS ON THRESHOLD IMPLEMENTATIONS

Georg SIGL // Technische Universität München

Cryptographic implementations typically need to be secured to retain their secrets in the presence of attacks. As a countermeasure to prevent side-channel attacks, threshold implementations are a commonly encountered concept. They resemble a multi-party computation, where the value is split in independent shares and processed separately.

In this work, we challenge the underlying security assumption that observing these individually processed values is difficult. We observe leakage by spatially separating the shares on an FPGA using multiple electro-magnetic (EM) probes simultaneously for localized EM analysis. We experimentally verify that the security gain is 238 times less with this method when compared to the power side-channel. In total, we only need 4,300 traces to break a second-order secure implementation.

Moreover, such a reduction in protection level is only possible when using multiple probes and applying our attack strategy which is based on state-of-the-art template attacks. This attack can easily be carried out by any attacker at the expense of buying more probes which emphasizes the danger of such attacks.

11. SIDE-CHANNEL ROBUSTNESS ANALYSIS OF MASKED PROGRAMS

Inès Ben EL Ouahma, Quentin Meunier, Karine Heydemann, Emmanuelle Encrenaz // Sorbonne Universités, LIP6.

Masking is a popular countermeasure against side-channel attacks, which randomizes secret data with random and uniform variables called masks. At software level, masking is usually added in the source code and its effectiveness needs to be verified. Such an analysis must be performed at the assembly level because the compilation flow may alter the added protections.

In this talk, we present a method to check side-channel robustness of masked programs using a symbolic approach. The aim is to verify that the result of each intermediate computation is statistically independent from secret variables. We propose to infer the distribution type of the result

by analyzing the expression of the computation using a set of inference rules. Our inference system is sound, yet not complete, and is enriched with a bit level analysis that allows to state statistical independency in more cases. Currently our method can analyze compiled and annotated codes which are masked at first order. The talk will give experimental results comparing analyzed programs at source and assembly level, as well as on a masked AES.

// SESSION // ADVANCES IN SECURE SYSTEMS

1. LIGHTWEIGHT AND SECURE SCHEME TO MITIGATE DENIAL-OF-SERVICE ON WAKE-UP RADIOS FOR IOT DEVICES

Maxime Montoya, Simone Bacles-Min, Anca Molnos, Jacques Fournier // CEA Tech

Wake-up radios are mechanisms that control the sleep and active modes of energy-constrained Internet of Things (IoT) nodes which have to be operational during a long time. These radios detect pre-determined wake-up tokens and switch the devices to an active state. Such systems are vulnerable to a kind of Denial-of-Service attacks called Denial-of-Sleep, where attackers continuously send wake-up tokens to deplete the battery of the nodes. We propose a protocol to mitigate these attacks that includes a novel solution to generate hard-to-guess wake-up tokens at every wake-up. This protocol can be used together with many common communication protocols for the IoT such as LoRa, Sigfox, or ZigBee. Simulations show that under standard operating conditions, it has a negligible energy overhead (0.03%), while it increases the lifetime of an IoT node by more than 40 times under Denial-of-Sleep attack. Finally, we compare our protocol to related work against Denial-of-Sleep attacks, and explain why it is both more resilient and more energy-efficient than existing approaches.

2. ML-ENHANCED FIA DETECTION

Robert NGuyen, Nicolas Bruneau, Xuan Thuy Ngo, Michel Le Rolland, Adrien Facon // Secure-IC

Today, chips are equipped with multiple sensors, of different kinds, some with primary goal data acquisition functionality, but also sensors for adaptation to the environment (like battery level, temperature, wireless activity in the neighborhood, etc.). All those can advantageously complement security sensors. Security sensors watch events which would hint for attack conditions. It can consist in:

- Abnormal physical operating conditions in terms of temperature, voltage, clock frequency, reset line stability, etc.
- Abnormal activity (detection of port scanning, unexpected data flowing out the device, load of processor, strange failure signals such as multiple segmentation violations within short period of time, etc.)

All these pieces of information can be processed to decide whether the device shall be considered in a nominal or in an unsafe environment. This is where AI comes into play. Indeed, AI is the solution to analyze fuzzy information arising from "big data" measurements collections. Merging heterogeneous signals allows to leverage unexploited sources of information such as randomness quality, tiny clock modulation, noise statistic moments modification for security event detection. Moreover for security chips, this AI shall run within the chips for integrity and data reduction efficiency. In this case, even data rate and data volume are high, quick decisions must be taken. Indeed, a laser or a malware

attack requires few clock cycles to exploit the chip: installing a backdoor is a matter of kilobytes of payload.

Secure-IC's Smart Monitor is an AI-enabled on-chip security supervisor improving both security event detection, diagnostic and decision-making process. This on-chip security headquarter creates collective intelligence and coherence between IPs (analogic or digital) and other whistleblowers and weak signals (software or hardware) improving both security event detection, analysis, diagnostic and decision-making process. It allows to gain advantage over attackers and interestingly to perform rich Business Intelligence.

3. SCAN CHAIN ENCRYPTION, A COUNTERMEASURE AGAINST SCAN ATTACKS

Mathieu Da Silva // LIRMM

Security in Integrated Circuits (ICs) domain is an important challenge, especially with regard to the side channel offered by test infrastructures. Test interfaces allow accessing the internal states of the IC by mean of the scan chains for testing and debugging purposes. In terms of security, scan chains are however a potential source of leakage that can be exploited by attackers. For instance, they can steal secret information, analyze the internal states of the circuit in order to help reverse-engineering, or modify the circuit operations. A countermeasure against such attacks is to on-chip encrypt the data flowing through the scan chains. Two types of ciphers can be used: either stream ciphers or block ciphers. Both have pros and cons in terms of performance and security.

// SESSION // SECURITY ASSESSMENT, CHARACTERIZATION & TEST

4. INTEGRATED CIRCUIT MODIFICATION WITH FOCALIZED X-RAYS BEAM

Stéphanie Anceau // CEA Tech

The understanding and conception of electronic devices require tools to modify these devices after their fabrication. A device failure or a functional bug require to modify the component behavior in order to find the corresponding problem. This modification of the integrated circuit is classically done with a Focused Ion Beam (FIB). This equipment permits to etch materials and deposit conductive or insulator materials, which allows to modify the interconnections of the integrated circuit. This operation is called circuit edit because the device can be reconfigured after his conception.

Previous experimentations on ESRF focalized beam line in Grenoble allowed to demonstrate that a focalized XRays perturbation changes the state of a single NMOS transistor and a single memory cell of SRAM-EEPROM and Flash memories block in a semi-permanent manner in the electronic device. This proof of concept has been realized on a new CMOS technology device (45 nm). The 50 nm focalization allows to modify one single NMOS transistor. The most aggressive technologies (<20nm) can be addressed with this technique even with a 50nm focalization diameter. Contrary to the FIB the interconnections of the device are not modified: the state of one (or several if necessary) transistor(s) is modified. This modification is semi-permanent because it is reversible with a simple heating treatment. This new circuit edit technique is very promising.

5. AUTOFOCUS IN INFRARED MICROSCOPY

Raphaël Abéle // STMicroelectronics

In the context of the secured Integrated Circuit (IC) characterization, many physical disruptions exists. One of these consists in a laser shot targeted on internal structures of the IC. Accuracy of the shot relies on the precision of the laser power calibration and its 3D positioning inside the IC. An automatic focusing mechanism on the conductive tracks of the IC could help a part of these generally handmade adjustments.

Autofocus (AF) is a widely investigated problem for natural scenes images, industrial assembly and biologic microscopy. A new effective AF method is proposed for infrared (IR) microscopy in the context of Integrated Circuit industry (IC). The proposal operates in the wavelet domain using a custom orthogonal wavelet for the 2D Discrete Wavelet Transform (DWT). The quality criterion of our AF algorithm relies on the standard deviance of the DWT coefficients, computed per subband and per level. Tested on several optical magnifying lenses, our method is robust and time-efficient, usable on the fly in IC location system.

6. EVALUATION OF BIOMETRIC SYSTEMS: WHO SAID STRAIGHTFORWARD?

Claude Barral // Bactec

The current rapid growth of biometric authentication features within consumer devices paves the way to security and performance certification needs. For more than a decade, IT security players tried to map certification schemes from smart cards and ICT systems to the area of Biometrics without success. Why? We will discuss all the issues one may face while setting up an evaluation environment for any biometric systems: which biometric data is targeted? How many different security settings? What is a representative database? Public vs Private databases? Target database size and architecture? How many authentication tests needed to claim a 0,001% false acceptance rate? Which evaluation target: Compliance? Interoperability? Security ? Performances? Well, definitely not straightforward indeed! You will see...

// SESSION // FAULT INJECTION ANALYSIS: ATTACKS AND PROTECTIONS (II)

7. EM FAULT INJECTION SUSCEPTIBILITY CRITERION AND ITS APPLICATION TO THE LOCALIZATION OF HOTSPOTS

Maxime Madau // STMicroelectronics / LIRMM

Electromagnetic (EM) fault injection has been proven efficient in attacking targets such as system-on-chip (SoC) or smartcards. Nonetheless, security characterizations, performed either by certification laboratories or by firms, are time consuming and this impacts on the final result. Indeed complete tests of integrated circuits (ICs) require trying numerous parameters, from pulse polarity to probes geometry and coupling, hence many maps are required to test all surface of Devices Under Test (DUT) and are unfortunately rarely performed. We propose a criterion to find zones with a high susceptibility to EM Fault Injection (EMFI). By using preprocessing tools based on both the effects of

EMFI on circuits and the analysis of EM emission traces, we are able to speed up the search of zones where faults are more likely to occur hence reducing the time required for security characterizations

8. BACKSIDE SHIELD BASED ON PACKAGING TECHNOLOGIES FOR CHIPS OR SYSTEMS SECURING

Stephan Borel // CEA Tech

A structure intended to protect Integrated Circuits (IC) against physical attacks will be presented. Located on the backside of a chip, it complements the countermeasures usually available on the front side of secure components. It aims at preventing attacks such as fault injection by laser illumination and can trigger an alert in case of invasive attacks by circuit edit or micro-probing. Weakening structures have been designed so as to cause the breakage of the die in case of thinning, and a metallic serpentine used as an attack witness has been thought with a maximal complexity so that an attacker cannot skirt it. These elements can be fabricated using standard packaging techniques in a wafer level integration, whether at chip or system scale. The concept of a secure System in Package (SiP) using unsecured chips is proposed, opening the perspective of components fully “secured by packaging”.



Deep Learning for Embedded Security Evaluation

Emmanuel Prouff

Joint work with Ryad Benadjila, Eleonora Cagli (CEA LETI), Cécile Dumas (CEA LETI), Housseem Maghrebi (UL), Thibault Portigliatti (ex SAFRAN), Rémi Strullu and Adrian Thillard

Laboratoire de Sécurité des Composants, ANSSI, France
Partially funded by **REASSURE** H2020 Project

May, PHYSIC 2018



Contents

1. Context and Motivation
 - 1.1 Illustration
 - 1.2 Template Attacks
 - 1.3 Countermeasures
 - 1.4 State of the Art
2. A machine learning approach to classification
 - 2.1 Introduction
 - 2.2 The Underlying Classification Problem
 - 2.3 Convolutional Neural Networks
 - 2.4 Training of Models
3. Building a Community Around The Subject
 - 3.1 ASCAD Open Data-Base
 - 3.2 Leakage Characterization and Training
 - 3.3 Results
4. Conclusions



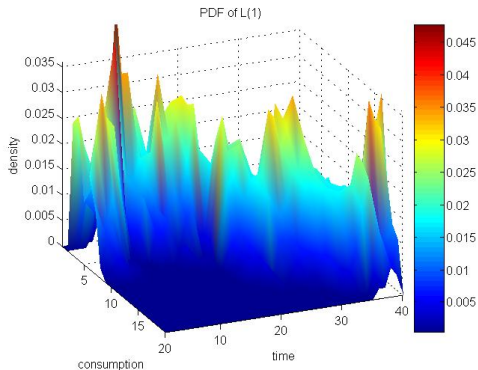
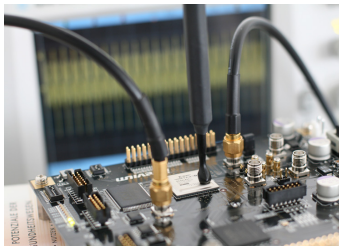
Probability distribution function (pdf) of Electromagnetic Emanations

Cryptographic Processing with a secret $k = 1$.



Probability distribution function (pdf) of Electromagnetic Emanations

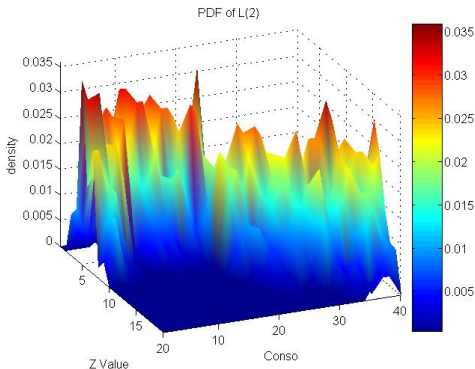
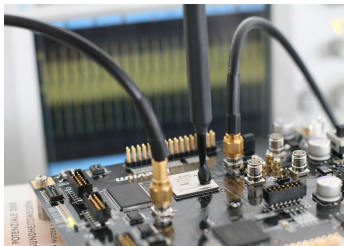
Cryptographic Processing with a secret $k = 1$.





Probability distribution function (pdf) of Electromagnetic Emanations

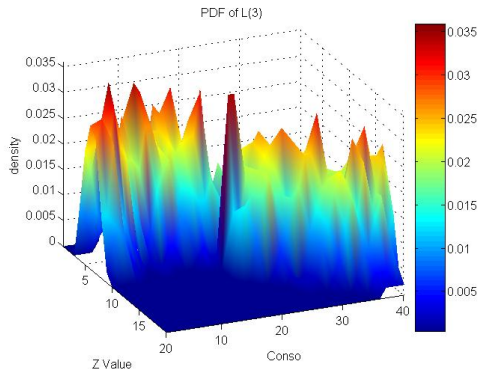
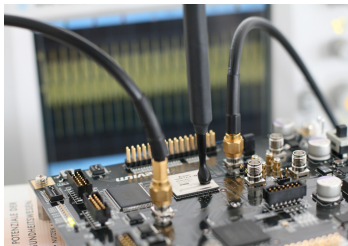
Cryptographic Processing with a secret $k = 2$.





Probability distribution function (pdf) of Electromagnetic Emanations

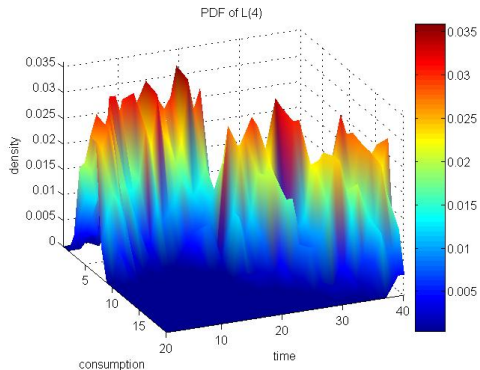
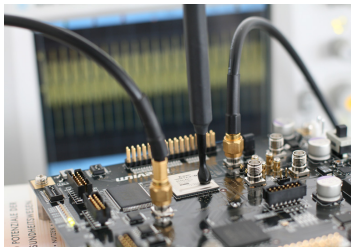
Cryptographic Processing with a secret $k = 3$.





Probability distribution function (pdf) of Electromagnetic Emanations

Cryptographic Processing with a secret $k = 4$.





Deep Learning for Embedded Security Evaluation

Context:



Target Device



Clone Device



Context:

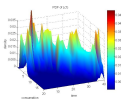


Target Device

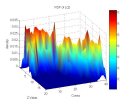


Clone Device

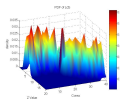
► [On Clone Device] For every k estimate the pdf of $\vec{X} \mid K = k$.



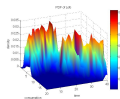
$k = 1$



$k = 2$



$k = 3$



$k = 4$

.....



Context:

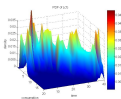


Target Device

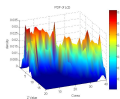


Clone Device

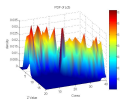
► [On Clone Device] For every k estimate the pdf of $\vec{X} \mid K = k$.



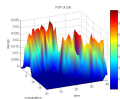
$k = 1$



$k = 2$



$k = 3$



$k = 4$

.....



Context:

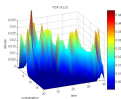


Target Device

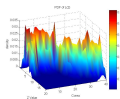


Clone Device

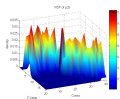
- ▶ [On Clone Device] For every k estimate the pdf of $\vec{X} \mid K = k$.



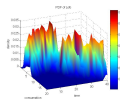
$k = 1$



$k = 2$



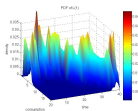
$k = 3$



$k = 4$

.....

- ▶ [On Target Device] Estimate the pdf of \vec{X} .



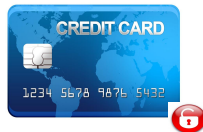
$k = ?$



Context:

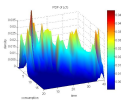


Target Device

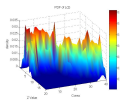


Clone Device

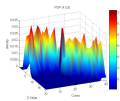
- ▶ [On Clone Device] For every k estimate the pdf of $\vec{X} \mid K = k$.



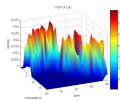
$k = 1$



$k = 2$



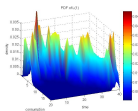
$k = 3$



$k = 4$

.....

- ▶ [On Target Device] Estimate the pdf of \vec{X} .



$k = ?$

- ▶ [Key-recovery] Compare the pdf estimations.



Side Channel Attacks (Classical Approach)

Notations

- ▶ \vec{X} observation of the device behaviour
- ▶ P public input of the processing
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}



Side Channel Attacks (Classical Approach)

Notations

- ▶ \vec{X} observation of the device behaviour
- ▶ P public input of the processing
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}

$\Pr[Z|\vec{X}]$



Side Channel Attacks (Classical Approach)

Notations

- ▶ \vec{X} observation of the device behaviour
- ▶ P public input of the processing
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}

$\Pr[Z|\vec{X}]$

Template Attacks

- ▶ Profiling phase (using profiling traces under known Z)

- ▶ Attack phase (N attack traces \vec{x}_i , e.g. with known plaintexts p_i)



Side Channel Attacks (Classical Approach)

Notations

- ▶ \vec{X} observation of the device behaviour
- ▶ P public input of the processing
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}

$\Pr[Z|\vec{X}]$

Template Attacks

- ▶ Profiling phase (using profiling traces under known Z)
 - ▶ estimate $\Pr[\vec{X}|Z = z]$ by simple distributions for each value of z
- ▶ Attack phase (N attack traces \vec{x}_i , e.g. with known plaintexts p_i)



Side Channel Attacks (Classical Approach)

Notations

- ▶ $\vec{\mathbf{X}}$ observation of the device behaviour
- ▶ P public input of the processing
- ▶ \mathbf{Z} target (a cryptographic sensitive variable $\mathbf{Z} = f(P, K)$)

Goal: make inference over \mathbf{Z} , observing $\vec{\mathbf{X}}$

$\Pr[\mathbf{Z}|\vec{\mathbf{X}}]$

Template Attacks

- ▶ Profiling phase (using profiling traces under known \mathbf{Z})
 - ▶ estimate $\Pr[\vec{\mathbf{X}}|\mathbf{Z} = \mathbf{z}]$ for each value of \mathbf{z}
- ▶ Attack phase (N attack traces $\vec{\mathbf{x}}_i$, e.g. with known plaintexts p_i)
 - ▶ Log-likelihood score for each key hypothesis k

$$d_k = \sum_{i=1}^N \log \Pr[\vec{\mathbf{X}} = \vec{\mathbf{x}}_i | \mathbf{Z} = f(p_i, k)]$$



Side Channel Attacks (Classical Approach)

Notations

- ▶ \vec{X} observation of the device behaviour
- ▶ P public input of the processing
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}

$\Pr[Z|\vec{X}]$

Template Attacks

- ▶ Profiling phase (using profiling traces under known Z)
 - ▶ mandatory dimensionality reduction
 - ▶ estimate $\Pr[\vec{X}|Z = z]$ for each value of z
- ▶ Attack phase (N attack traces \vec{x}_i , e.g. with known plaintexts p_i)
 - ▶ Log-likelihood score for each key hypothesis k

$$d_k = \sum_{i=1}^N \log \Pr[\vec{X} = \vec{x}_i | Z = f(p_i, k)]$$



Side Channel Attacks (Classical Approach)

Notations

- ▶ \vec{X} observation of the device behaviour
- ▶ P public input of the processing
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}

$\Pr[Z|\vec{X}]$

Template Attacks

- ▶ Profiling phase (using profiling traces under known Z)
 - ▶ manage de-synchronization problem
 - ▶ mandatory dimensionality reduction
 - ▶ estimate $\Pr[\varepsilon(\vec{X})|Z = z]$ for each value of z
- ▶ Attack phase (N attack traces \vec{x}_i , e.g. with known plaintexts p_i)
 - ▶ Log-likelihood score for each key hypothesis k

$$d_k = \sum_{i=1}^N \log \Pr[\varepsilon(\vec{X}) = \varepsilon(\vec{x}_i) | Z = f(p_i, k)]$$



Defensive Mechanisms



Misaligning Countermeasures

- ▶ Random Delays, Clock Jittering, ...
- ▶ In theory: assume to be insufficient to provide security
- ▶ In practice: one of the main issues for evaluators
- ▶ \Rightarrow **Need for efficient resynchronization techniques**



Defensive Mechanisms



Misaligning Countermeasures

- ▶ Random Delays, Clock Jittering, ...
- ▶ In theory: assume to be insufficient to provide security
- ▶ In practice: one of the main issues for evaluators
- ▶ \implies **Need for efficient resynchronization techniques**

Masking Countermeasure

- ▶ Each key-dependent internal state element is randomly split into **2 shares**
- ▶ The crypto algorithm is adapted to always manipulate shares at \neq times
- ▶ The adversary needs to recover information on the two shares to recover K
- ▶ \implies **Need for efficient Methods to recover tuple of leakage samples that jointly depend on the target secret**

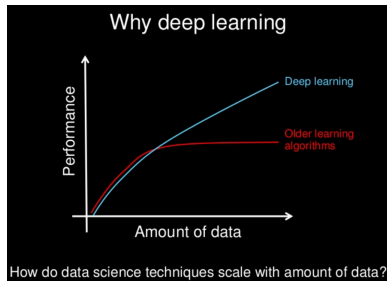


Contents

1. Context and Motivation
 - 1.1 Illustration
 - 1.2 Template Attacks
 - 1.3 Countermeasures
 - 1.4 State of the Art
2. A machine learning approach to classification
 - 2.1 Introduction
 - 2.2 The Underlying Classification Problem
 - 2.3 Convolutional Neural Networks
 - 2.4 Training of Models
3. Building a Community Around The Subject
 - 3.1 ASCAD Open Data-Base
 - 3.2 Leakage Characterization and Training
 - 3.3 Results
4. Conclusions

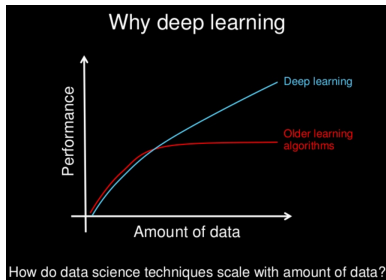


Motivating Conclusions





Motivating Conclusions



Now:

- ▶ preprocessing to prepare data
 - ▶ Traces resynchronisation
 - ▶ Selection of Pols
- ▶ make strong hypotheses on the statistical dependency
 - ▶ e.g. Gaussian approximation
- ▶ characterization to extract information
 - ▶ e.g. Maximum Likelihood

The proposed perspective:

- ▶ ~~preprocessing to prepare data~~
 - ▶ ~~Traces resynchronisation~~
 - ▶ ~~Selection of Pols~~
- ▶ ~~make strong hypotheses on the statistical dependency~~
 - ▶ ~~e.g. Gaussian approximation~~
- ▶ **Train** algorithms to directly extract information



Side Channel Attacks

Notations

- ▶ \vec{X} side channel trace
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}

$$\Pr[Z|\vec{X}]$$

~~Template Attacks~~ Machine Learning Side Channel Attacks

- ▶ Profiling phase (using profiling traces under known Z)
 - ▶ manage de-synchronization problem
 - ▶ mandatory dimensionality reduction
 - ▶ estimate $\Pr[\vec{X}|Z = z]$ for each value of z
- ▶ Attack phase (N attack traces, e.g. with known plaintexts p_i)
 - ▶ Log-likelihood score for each key hypothesis k

$$d_k = \sum_{i=1}^N \log \Pr[\vec{X} = \vec{x}_i | Z = f(p_i, k)]$$



Side Channel Attacks **with a Classifier**

Notations

- ▶ \vec{X} side channel trace
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}

$$\Pr[Z|\vec{X}]$$

~~Template Attacks~~ Machine Learning Side Channel Attacks

- ▶ Profiling phase (using profiling traces under known Z)
 - ▶ manage de-synchronization problem
 - ▶ mandatory dimensionality reduction
 - ▶ estimate $\Pr[\vec{X}|Z = z]$ for each value of z
- ▶ Attack phase (N attack traces, e.g. with known plaintexts p_i)
 - ▶ Log-likelihood score for each key hypothesis k

$$d_k = \sum_{i=1}^N \log \Pr[\vec{X} = \vec{x}_i | Z = f(p_i, k)]$$



Side Channel Attacks **with a Classifier**

Notations

- ▶ \vec{X} side channel trace
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}

$$\Pr[Z|\vec{X}]$$

~~Template Attacks~~ Machine Learning Side Channel Attacks

- ▶ Training phase (using training traces under known Z)
 - ▶ manage de-synchronization problem
 - ▶ mandatory dimensionality reduction
 - ▶ estimate $\Pr[\vec{X}|Z = z]$ for each value of z
- ▶ Attack phase (N attack traces, e.g. with known plaintexts p_i)
 - ▶ Log-likelihood score for each key hypothesis k

$$d_k = \sum_{i=1}^N \log \Pr[\vec{X} = \vec{x}_i | Z = f(p_i, k)]$$



Side Channel Attacks **with a Classifier**

Notations

- ▶ \vec{X} side channel trace
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}

$$\Pr[Z|\vec{X}]$$

~~Template Attacks~~ Machine Learning Side Channel Attacks

- ▶ Training phase (using training traces under known Z)
 - ▶ manage de-synchronization problem
 - ▶ mandatory dimensionality reduction
 - ▶ construct a classifier F s.t. $F(\vec{x})[z] = y \approx \Pr[Z = z|\vec{X} = \vec{x}]$
- ▶ Attack phase (N attack traces, e.g. with known plaintexts p_i)
 - ▶ Log-likelihood score for each key hypothesis k

$$d_k = \sum_{i=1}^N \log \Pr[\vec{X} = \vec{x}_i | Z = f(p_i, k)]$$



Side Channel Attacks **with a Classifier**

Notations

- ▶ \vec{X} side channel trace
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}

$$\Pr[Z|\vec{X}]$$

~~Template Attacks~~ Machine Learning Side Channel Attacks

- ▶ Training phase (using training traces under known Z)
 - ▶ manage de-synchronization problem
 - ▶ mandatory dimensionality reduction
 - ▶ construct a classifier F s.t. $F(\vec{x})[z] = y \approx \Pr[Z = z|\vec{X} = \vec{x}]$
- ▶ Attack phase (N attack traces, e.g. with known plaintexts p_i)
 - ▶ Log-likelihood score for each key hypothesis k

$$d_k = \sum_{i=1}^N \log F(\vec{x}_i)[f(p_i, k)]$$



Side Channel Attacks **with a Classifier**

Notations

- ▶ \vec{X} side channel trace
- ▶ Z target (a cryptographic sensitive variable $Z = f(P, K)$)

Goal: make inference over Z , observing \vec{X}

$$\Pr[Z|\vec{X}]$$

~~Template Attacks~~ Machine Learning Side Channel Attacks

- ▶ Training phase (using training traces under known Z)
 - ▶ manage de-synchronization problem
 - ▶ mandatory dimensionality reduction
 - ▶ construct a classifier F s.t.
 $F(\vec{x})[z] = y \approx \Pr[Z = z | \vec{X} = \vec{x}]$
- } **Integrated approach**
- ▶ Attack phase (N attack traces, e.g. with known plaintexts p_i)
 - ▶ Log-likelihood score for each key hypothesis k

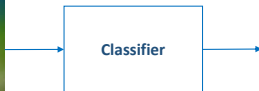
$$d_k = \sum_{i=1}^N \log F(\vec{x}_i)[f(p_i, k)]$$



Classification

Classification problem

Assign to a datum \vec{X} (e.g. an image) a label Z among a set of possible labels $Z = \{\text{Cat}, \text{Dog}, \text{Horse}\}$

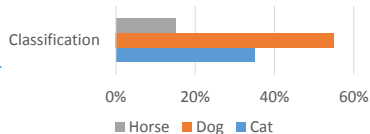
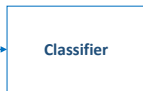




Classification

Classification problem

Assign to a datum \vec{X} (e.g. an image) a label Z among a set of possible labels $Z = \{\text{Cat}, \text{Dog}, \text{Horse}\}$

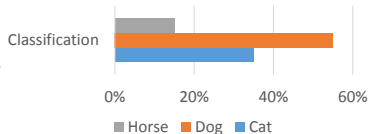




Classification

Classification problem

Assign to a datum \vec{X} (e.g. an image) a label Z among a set of possible labels $\mathcal{Z} = \{\text{Cat}, \text{Dog}, \text{Horse}\}$



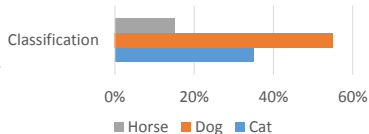
$$\Pr[Z|\vec{X}]$$



Classification

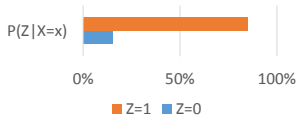
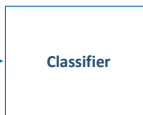
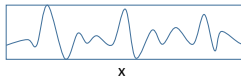
Classification problem

Assign to a datum \vec{X} (e.g. an image) a label Z among a set of possible labels $\mathcal{Z} = \{\text{Cat}, \text{Dog}, \text{Horse}\}$



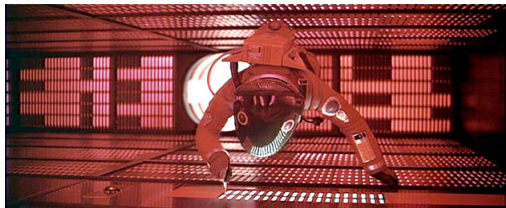
$$\Pr[Z|\vec{X}]$$

SCA as a Classification Problem





Machine Learning Approach



Overview of Machine Learning Methodology

Human effort:

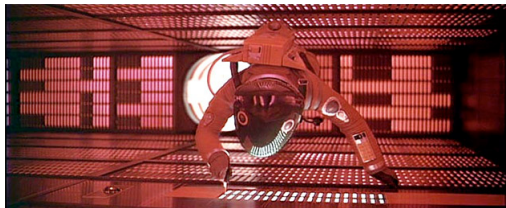
- ▶ choose a class of algorithms
- ▶ choose a model to fit + tune **hyper-parameters**

Automatic training:

- ▶ automatic tuning of **trainable parameters** to fit data



Machine Learning Approach



Overview of Machine Learning Methodology

Human effort:

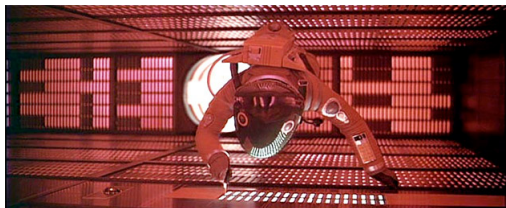
- ▶ choose a class of algorithms
Neural Networks
- ▶ choose a model to fit +
tune **hyper-parameters**

Automatic training:

- ▶ automatic tuning of
trainable parameters
to fit data
Stochastic Gradient Descent



Machine Learning Approach



Overview of Machine Learning Methodology

Human effort:

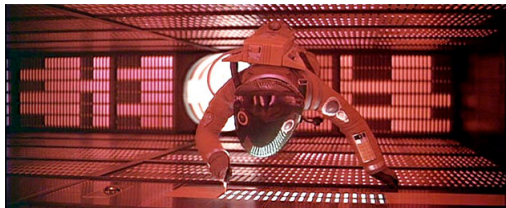
- ▶ choose a class of algorithms
Neural Networks
- ▶ choose a model to fit +
tune **hyper-parameters**
MLP, ConvNet

Automatic training:

- ▶ automatic tuning of
trainable parameters
to fit data
Stochastic Gradient Descent



Machine Learning Approach



Overview of Machine Learning Methodology

Human effort:

- ▶ choose a class of algorithms
Neural Networks
- ▶ choose a model to fit +
tune **hyper-parameters**
MLP, ConvNet

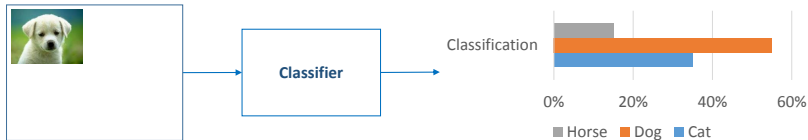
Automatic training:

- ▶ automatic tuning of
trainable parameters
to fit data
Stochastic Gradient Descent



Convolutional Neural Networks

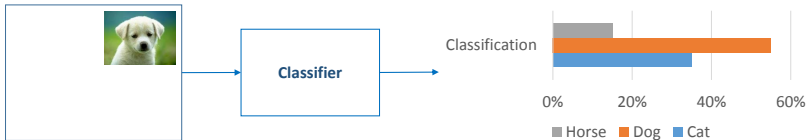
An answer to translation-invariance





Convolutional Neural Networks

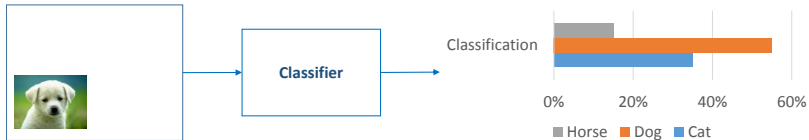
An answer to translation-invariance





Convolutional Neural Networks

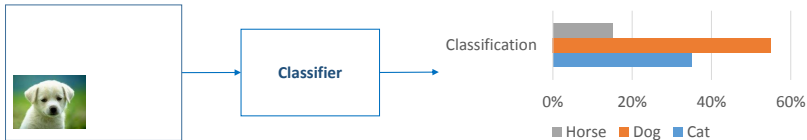
An answer to translation-invariance





Convolutional Neural Networks

An answer to translation-invariance

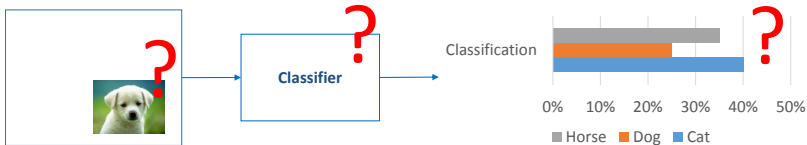


It is important to explicit the data translation-invariance



Convolutional Neural Networks

An answer to translation-invariance



It is important to explicit the data translation-invariance



Convolutional Neural Networks

An answer to translation-invariance



It is important to explicit the data translation-invariance



Convolutional Neural Networks

An answer to translation-invariance



It is important to explicit the data translation-invariance



Convolutional Neural Networks

An answer to translation-invariance



It is important to explicit the data translation-invariance



Convolutional Neural Networks

An answer to translation-invariance



It is important to explicit the data translation-invariance
Convolutional Neural Networks: share weights across space



Convolutional Neural Networks

An answer to translation-invariance



It is important to explicit the data translation-invariance
Convolutional Neural Networks: share weights across space

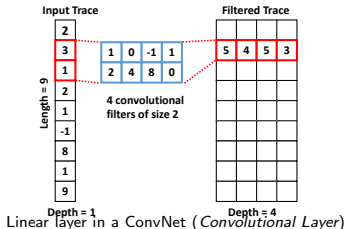


Convolutional Neural Networks

An answer to translation-invariance



It is important to explicit the data translation-invariance
Convolutional Neural Networks: share weights across space



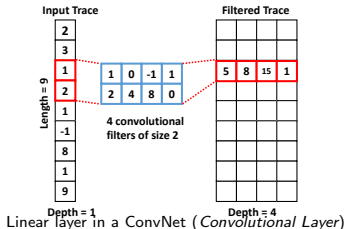


Convolutional Neural Networks

An answer to translation-invariance



It is important to explicit the data translation-invariance
Convolutional Neural Networks: share weights across space



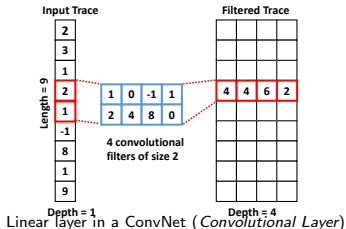


Convolutional Neural Networks

An answer to translation-invariance



It is important to explicit the data translation-invariance
Convolutional Neural Networks: share weights across space



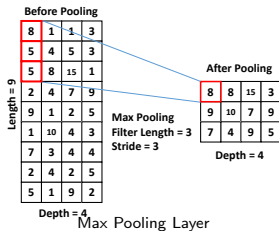
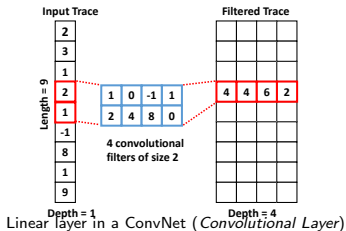


Convolutional Neural Networks

An answer to translation-invariance



It is important to explicit the data translation-invariance
Convolutional Neural Networks: share weights across space





Training of Neural Networks

Trading Side-Channel Expertise for Deep Learning Expertise or huge computational power!

Training



Training of Neural Networks

Trading Side-Channel Expertise for Deep Learning Expertise or huge computational power!

Training

Aims at finding the **parameters** of the function modelling for the dependency btw the target value and the leakage.



Training of Neural Networks

Trading Side-Channel Expertise for Deep Learning Expertise or huge computational power!

Training

Aims at finding the **parameters** of the function modelling for the dependency btw the target value and the leakage.

The search is done by solving a minimization problem with respect to some metric (aka loss function)



Training of Neural Networks

Trading Side-Channel Expertise for Deep Learning Expertise or huge computational power!

Training

Aims at finding the **parameters** of the function modelling for the dependency btw the target value and the leakage.

The search is done by solving a minimization problem with respect to some metric (aka loss function)

The training algorithm has itself some **training hyper-parameters**:
the number of iterations (aka **epochs**) of the minimization procedure,
the number of input traces (aka **batch**) treated during a single iteration.



Training of Neural Networks

Trading Side-Channel Expertise for Deep Learning Expertise or huge computational power!

Training

Aims at finding the **parameters** of the function modelling for the dependency btw the target value and the leakage.

The search is done by solving a minimization problem with respect to some metric (aka loss function)

The training algorithm has itself some **training hyper-parameters**:

- the number of iterations (aka **epochs**) of the minimization procedure,
- the number of input traces (aka **batch**) treated during a single iteration.

The trained model has **architecture hyper-parameters**:

- the size of the layers, the nature of the layers, the number of layers, etc.



Training of Neural Networks

Trading Side-Channel Expertise for Deep Learning Expertise or huge computational power!

Training

Aims at finding the **parameters** of the function modelling for the dependency btw the target value and the leakage.

The search is done by solving a minimization problem with respect to some metric (aka loss function)

The training algorithm has itself some **training hyper-parameters**:

- the number of iterations (aka **epochs**) of the minimization procedure,
- the number of input traces (aka **batch**) treated during a single iteration.

The trained model has **architecture hyper-parameters**:

- the size of the layers, the nature of the layers, the number of layers, etc.

Tricky Points

Find sound hyper-parameters is the main issue in Deep Learning: **this can be done thanks to a good understanding of the underlying structure of the data and/or access to important computational power.**



Contents

1. Context and Motivation
 - 1.1 Illustration
 - 1.2 Template Attacks
 - 1.3 Countermeasures
 - 1.4 State of the Art
2. A machine learning approach to classification
 - 2.1 Introduction
 - 2.2 The Underlying Classification Problem
 - 2.3 Convolutional Neural Networks
 - 2.4 Training of Models
3. Building a Community Around The Subject
 - 3.1 ASCAD Open Data-Base
 - 3.2 Leakage Characterization and Training
 - 3.3 Results
4. Conclusions



Creation of an open database for Training and Testing

ANSSI recently publishes

- ▶ source codes of secure implementations of AES128 for public 8-bit architectures (<https://github.com/ANSSI-FR/secAES-ATmega8515>)
 - ▶ **first version:** 10-masking + processing in random order
 - ▶ **second version:** affine masking + processing in random order (plus other minor tricks)
- ▶ data-bases of electromagnetic leakages (<https://github.com/ANSSI-FR/ASCAD>)
- ▶ example scripts for the training and testing of models in SCA contexts

Goal

- ▶ Enable fair and easy benchmarking
- ▶ Initiate discussions and exchanges on the application of DL to SCA
- ▶ Create a community of contributors on this subject



Nature of the Observations/Traces

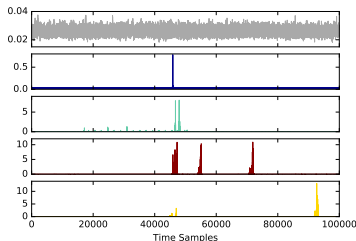
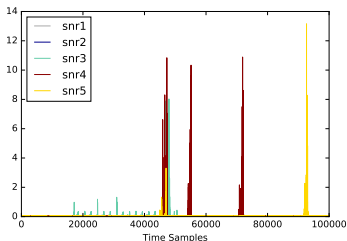
Side-channel observations in ASCAD correspond to the masked processing of a simple cryptographic primitive
Information leakage validated thanks to SNR characterization



Nature of the Observations/Traces

Side-channel observations in ASCAD correspond to the masked processing of a simple cryptographic primitive

Information leakage validated thanks to SNR characterization

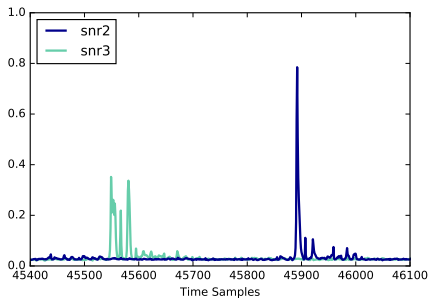


<i>snr1</i>	unmasked sbx output	$SBox(p \oplus k)$
<i>snr2</i>	masked sbx output	$SBox(p \oplus k) \oplus r_{out}$
<i>snr3</i>	common sbx output mask	r_{out}
<i>snr4</i>	masked sbx output in linear parts	$SBox(p \oplus k) \oplus r_{lin}$
<i>snr5</i>	sbx output mask in linear parts	r_{lin}



Nature of the Observations/Traces

Side-channel observations in ASCAD correspond to the masked processing of a simple cryptographic primitive
Information leakage validated thanks to SNR characterization

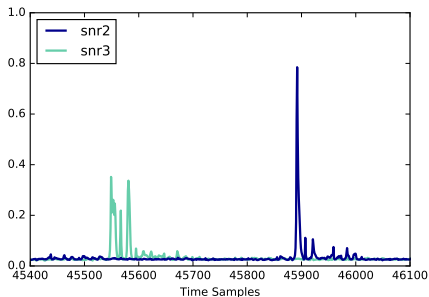


Validate that shares are **manipulated at different times**



Nature of the Observations/Traces

Side-channel observations in ASCAD correspond to the masked processing of a simple cryptographic primitive
Information leakage validated thanks to SNR characterization



Validate that shares are **manipulated at different times**
Scripts are also proposed to add **artificial signal jittering**



Our Training Strategy



Our Training Strategy

Find a **base model architecture** and find training hyper-parameters for which a convergence towards the good key hypothesis is visible



Our Training Strategy

Find a **base model architecture** and find training hyper-parameters for which a convergence towards the good key hypothesis is visible

Fine-tune all the hyper-parameters one after another to get the best efficiency/effectiveness trade-off



Our Training Strategy

Find a **base model architecture** and find training hyper-parameters for which a convergence towards the good key hypothesis is visible

Fine-tune all the hyper-parameters one after another to get the best efficiency/effectiveness trade-off

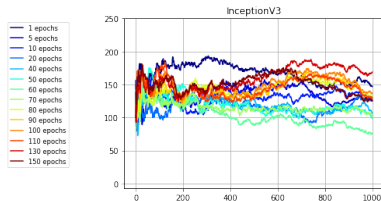
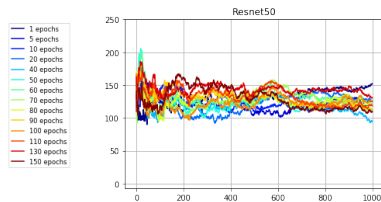
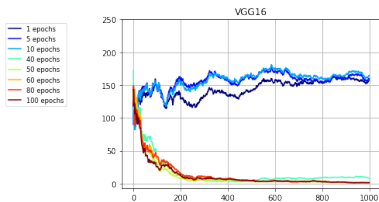
Table: Benchmarks Summary

Parameter	Reference	Metric	Range	Choice
Training Parameters				
Epochs	-	rank vs time	10, 25, 50, 60, . . . , 100, 150	up to 100
Batch Size	-	rank vs time	50, 100, 200	200
Architecture Parameters				
Blocks	n_{blocks}	rank, accuracy	[2..5]	5
CONV layers	n_{conv}	rank, accuracy	[0..3]	1
Filters	$n_{\text{filters},1}$	rank vs time	$\{2^i; i \in [4..7]\}$	64
Kernel Size	-	rank	{3, 6, 11}	11
FC Layers	n_{dense}	rank, accuracy vs time	[0..3]	2
ACT Function	α	rank	ReLU, Sigmoid, Tanh	ReLU
Pooling Layer	-	rank	Max, Average, Stride	Average
Padding	-	rank	Same, Valid	Same



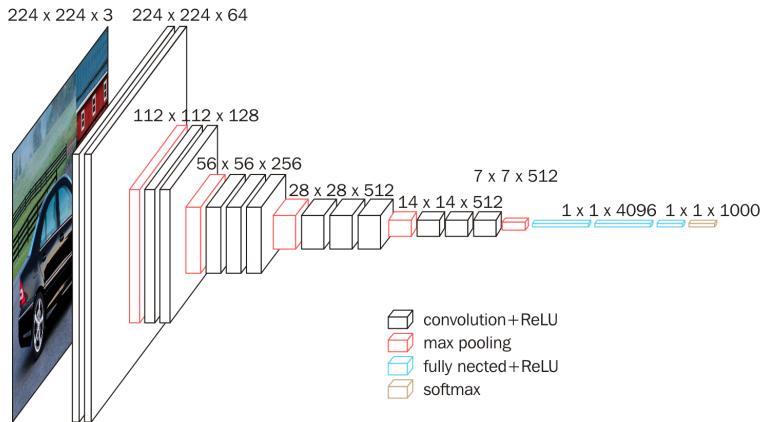
The Base Architecture

h Mean rank of the good-key hypothesis obtained with **VGG-16**, **ResNet-50** and **Inception-v3** w.r.t. different epochs:



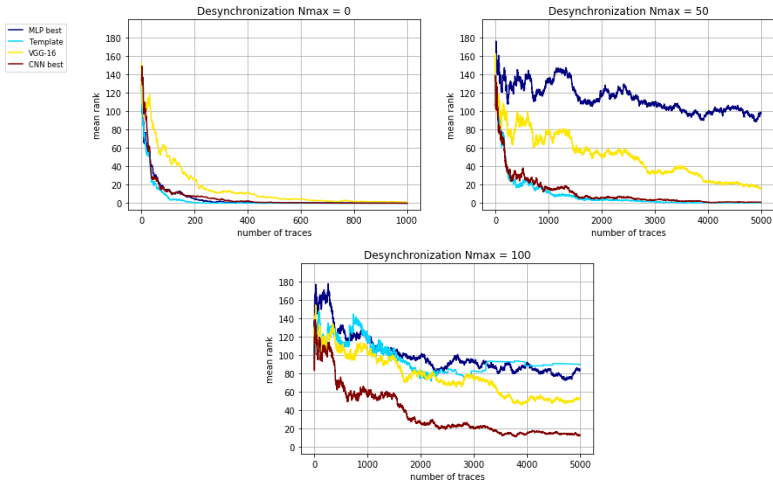


VGG-16 Architecture





Comparisons with State-Of-the-Art Methods





Feedbacks & Open Issues

Feedbacks



Feedbacks & Open Issues

Feedbacks

- ▶ The number of epochs for the training is between 100 and 1000



Feedbacks & Open Issues

Feedbacks

- ▶ The number of epochs for the training is between 100 and 1000
- ▶ Model architectures are relatively complex (more than 10 layers)



Feedbacks & Open Issues

Feedbacks

- ▶ The number of epochs for the training is between 100 and 1000
- ▶ Model architectures are relatively complex (more than 10 layers)
- ▶ Data-bases for the training must be large



Feedbacks & Open Issues

Feedbacks

- ▶ The number of epochs for the training is between 100 and 1000
- ▶ Model architectures are relatively complex (more than 10 layers)
- ▶ Data-bases for the training must be large
- ▶ Require important processing capacities (several GPUs, RAM memory, etc.)



Feedbacks & Open Issues

Feedbacks

- ▶ The number of epochs for the training is between 100 and 1000
- ▶ Model architectures are relatively complex (more than 10 layers)
- ▶ Data-bases for the training must be large
- ▶ Require important processing capacities (several GPUs, RAM memory, etc.)
- ▶ Importance of cross-validation



Feedbacks & Open Issues

Feedbacks

- ▶ The number of epochs for the training is between 100 and 1000
- ▶ Model architectures are relatively complex (more than 10 layers)
- ▶ Data-bases for the training must be large
- ▶ Require important processing capacities (several GPUs, RAM memory, etc.)
- ▶ Importance of cross-validation

Open Issues



Feedbacks & Open Issues

Feedbacks

- ▶ The number of epochs for the training is between 100 and 1000
- ▶ Model architectures are relatively complex (more than 10 layers)
- ▶ Data-bases for the training must be large
- ▶ Require important processing capacities (several GPUs, RAM memory, etc.)
- ▶ Importance of cross-validation

Open Issues

- ▶ Models are trained to recover manipulated values (e.g. sbx outputs) **but** not the key itself



Feedbacks & Open Issues

Feedbacks

- ▶ The number of epochs for the training is between 100 and 1000
- ▶ Model architectures are relatively complex (more than 10 layers)
- ▶ Data-bases for the training must be large
- ▶ Require important processing capacities (several GPUs, RAM memory, etc.)
- ▶ Importance of cross-validation

Open Issues

- ▶ Models are trained to recover manipulated values (e.g. sbx outputs) **but** not the key itself
- ▶ Current loss functions measure the accuracy of pdf estimations **but** not the efficiency of the resulting attack



Feedbacks & Open Issues

Feedbacks

- ▶ The number of epochs for the training is between 100 and 1000
- ▶ Model architectures are relatively complex (more than 10 layers)
- ▶ Data-bases for the training must be large
- ▶ Require important processing capacities (several GPUs, RAM memory, etc.)
- ▶ Importance of cross-validation

Open Issues

- ▶ Models are trained to recover manipulated values (e.g. sbox outputs) **but** not the key itself
- ▶ Current loss functions measure the accuracy of pdf estimations **but** not the efficiency of the resulting attack
- ▶ Adaptation to get (very) efficient key enumeration algorithms



Contents

1. Context and Motivation
 - 1.1 Illustration
 - 1.2 Template Attacks
 - 1.3 Countermeasures
 - 1.4 State of the Art
2. A machine learning approach to classification
 - 2.1 Introduction
 - 2.2 The Underlying Classification Problem
 - 2.3 Convolutional Neural Networks
 - 2.4 Training of Models
3. Building a Community Around The Subject
 - 3.1 ASCAD Open Data-Base
 - 3.2 Leakage Characterization and Training
 - 3.3 Results
4. Conclusions



Conclusions

- ▶ State-of-the-Art Template Attack separates resynchronization/dimensionality reduction from characterization



Conclusions

- ▶ State-of-the-Art Template Attack separates resynchronization/dimensionality reduction from characterization
- ▶ Deep Learning provides an integrated approach to directly extract information from rough data (no preprocessing)



Conclusions

- ▶ State-of-the-Art Template Attack separates resynchronization/dimensionality reduction from characterization
- ▶ Deep Learning provides an integrated approach to directly extract information from rough data (no preprocessing)
- ▶ Many recent results validate the practical interest of the Machine Learning approach



Conclusions

- ▶ State-of-the-Art Template Attack separates resynchronization/dimensionality reduction from characterization
- ▶ Deep Learning provides an integrated approach to directly extract information from rough data (no preprocessing)
- ▶ Many recent results validate the practical interest of the Machine Learning approach
- ▶ We are in the very beginning and we are still discovering how much Deep Learning is efficient



Conclusions

- ▶ State-of-the-Art Template Attack separates resynchronization/dimensionality reduction from characterization
- ▶ Deep Learning provides an integrated approach to directly extract information from rough data (no preprocessing)
- ▶ Many recent results validate the practical interest of the Machine Learning approach
- ▶ We are in the very beginning and we are still discovering how much Deep Learning is efficient
- ▶ New needs:
 - ▶ big data-bases for the training,
 - ▶ platforms to enable comparisons and benchmarking,
 - ▶ create an open community "ML for Embedded Security Analysis",
 - ▶ encourage exchanges with the Machine Learning community,
 - ▶ understand the efficiency of the current countermeasures



Conclusions

- ▶ State-of-the-Art Template Attack separates resynchronization/dimensionality reduction from characterization
- ▶ Deep Learning provides an integrated approach to directly extract information from rough data (no preprocessing)
- ▶ Many recent results validate the practical interest of the Machine Learning approach
- ▶ We are in the very beginning and we are still discovering how much Deep Learning is efficient
- ▶ New needs:
 - ▶ big data-bases for the training,
 - ▶ platforms to enable comparisons and benchmarking,
 - ▶ create an open community "ML for Embedded Security Analysis",
 - ▶ encourage exchanges with the Machine Learning community,
 - ▶ understand the efficiency of the current countermeasures

Thank You!

Questions?

FHE AND MPC PRIVACY PRESERVING COMPUTATIONS IN THE CLOUD

Nicolas Gama // EPFL

Slides sur <http://lab.algonics.net/fc18/index-phisc.html>

Starting transition towards products embedding post-quantum cryptography



Yannick Teglia
Aline Gouget

Phisic 2018 – Ecole des Mines de Saint-Etienne - Gardanne

Gemalto – Security Consulting & Services

Outlook


- ✦ Quantum threat to today's cryptography
 - ✦ Theoretical resistance of current cryptographic algorithms
 - ✦ Quantum algorithms & quantum computers
 - ✦ Current status
- ✦ Post-Quantum Cryptography
 - ✦ Running initiatives
 - ✦ Different families and noticeable characteristics
- ✦ Foreseen consequences in current and coming products
- ✦ Conclusion

Quantum threat to today's cryptography

Theoretical Resistance of Cryptographic Algorithms

- ✘ The resistance of a cryptographic algorithm is relative to the best known attack
- ✘ For Public Key cryptography, those attacks consist in solving the underlying mathematical « hard » problem
 - ✘ Factorization for RSA
 - ✘ The best known algorithm is sub exponential
 - ✘ Discrete logarithm for Elliptic curves
 - ✘ The best known algorithm is fully exponential
- ✘ For secret key cryptography, attacks are considered on a 'per case' basis
 - ✘ For AES-128, the best known attack (Bogdanov et al. [1]) would require $2^{126.2}$ operations instead of 2^{128} for brute force
- ✘ Advances are monitored by
 - ✘ Academia/industry (Lenstra et al [2], Ecrypt [3], ...)
 - ✘ Governmental agencies (ANSSI[4], NIST[5],BSI[6])
- ✘ The keylength has then to be set accordingly to ensure the correct security level
- ✘ The industry has to comply to requested security levels

4 | Starting transition towards products embedding post-quantum cryptography



Choix de la méthode

- Équations de Lenstra et Verheul (2000)
- Équations plus récentes de Lenstra (2004)
- Recommandations d'ECRYPT II (2012)
- Recommandations du NIST (2016)
- Recommandations de l'ANSSI (2014)
- CNSA Suite de l'IAD-NSA (2016)
- Network Working Group RFC3766 (2004)
- Recommandations du BSI (2017)

Comparer les méthodes

Ce papier [5] correspond aux recommandations de l'agence nationale de la sécurité des systèmes d'information (ANSSI). Il représente l'expression du gouvernement français en termes de qualité cryptographique.

Date	Symétrique	Factorisation Module	Logarithme discret Clef	Logarithme discret Groupe	Courbe elliptique GF(p) GF(2 ⁿ)		Hash
2014 - 2020	100	2048	200	2048	200	200	200
2021 - 2030	128	2048	200	2048	256	256	256
> 2030	128	3072	200	3072	256	256	256

Les tailles de clef sont exprimées en bit. Ces résultats garantissent une sécurité minimale. **Cliquer sur une valeur pour la comparer avec les autres méthodes.**

Remarques et algorithmes recommandés pour les systèmes symétriques :

La taille recommandée pour les systèmes symétriques est de 128 bits.
 La taille minimale des blocs de chiffrement par bloc est de 64 bits (128 bits recommandés et obligatoires après 2020).
 Il est recommandé d'employer des algorithmes par bloc et non des algorithmes de chiffrement par flot.
 Algorithme de chiffrement : AES-CBC (FIPS 197)

www.keylength.com



Quantum Algorithms for Cryptography

- ✦ In the middle 1990s, Grover [7] and Shor [8] published algorithms that could defeat some cryptographic primitives using a Quantum Computer
 - ✦ Shor
 - ✦ Specific use: factoring large integers and finding discrete logarithms
 - ✦ Cost reduced: from sub exponential to quadratic
 - ✦ Grover
 - ✦ Generic use: provides polynomial speed-up in unstructured search
 - ✦ Could be applied to symmetric cryptography
 - ✦ Cost reduced by a square root factor
- ✦ A Quantum Computer is expected to increase the computational power beyond the usual limits
 - ✦ Quantum systems could be in multiple states at the same time
 - ✦ From sequential computation to « holistic computation » (Breton [14])
- ✦ Bringing together those algorithms and real Quantum Computers could then theoretically jeopardize today's cryptography

Threats on current cryptography: where do we stand?

✘ Symmetric Cryptography

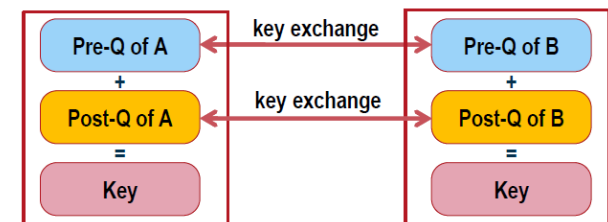
- ✘ AES-256 secure beyond 2050 (ETSI [15])
- ✘ AES-128, possibly secure yet
 - ✘ 2^{81} operations expected instead of the 2^{64} predicted by the rule of thumb (Grassl [17])
 - ✘ The cost of the algorithm as to be taken into account as well
- ✘ 3-DES with 3 keys
 - ✘ No academic publication
 - ✘ NIST says that 3-DES with 3 keys is ok up to 2030
- ✘ Several MAC and authenticated encryption modes can be broken (Kaplan [18])

✘ Hash Functions

- ✘ SHA-256 secure beyond 2050 (ETSI [15])
 - ✘ Attacks in 2^{166} for SHA-2 and SHA-3 (instead of 2^{128})
- ✘ Hash based signatures scheme safe against quantum computing

✘ ECC and RSA

- ✘ A Post Quantum RSA would require a 1 TB key from 4kbit primes (Bernstein [26])
- ✘ Replacement is needed for “long term” ⇔ when quantum computers are available
- ✘ No short-term endorsement of PQ cryptosystems in RGS Annex B1 of ANSSI (Gilbert [40])
 - ✘ Potential exception for hash-based signatures
 - ✘ Hybrid systems combining pre-quantum and post-quantum likely to be endorsed



Source [40]

Current Quantum Computers ...

- ✗ The real computational power of a quantum computer is difficult to estimate
- ✗ The qubit could be seen as a metric to quantify the power of a Quantum Computer
 - ✗ 1998 – 2 qubits
 - ✗ ...
 - ✗ 2018 - 72 qubits
 - ✗ NB: D-wave is claiming 128 qubits but with a different architecture and a limited scope
- ✗ Several recent announcements regarding Quantum Computers
 - ✗ Intel Tangle Lake : 49 qubit
 - ✗ IBM Q: 50 qubit
 - ✗ Google Bristlecone: 72 qubit
- ✗ Different technologies and architectures
 - ✗ There are qubits and qubits
 - ✗ Silicon qubits (Intel) vs regular qubits
 - ✗ Universal Quantum Computers vs dedicated quantum annealing (D-Wave)
 - ✗ The latter cannot process Shor's algorithm

... And their limits

- ✗ Regarding factorization, quantum computers are currently only able to factor 15
- ✗ Quantum Computing is a huge engineering challenge
 - ✗ Nanotechnology, quantum electronics, ...
- ✗ Quantum computers are not yet to replace classical ones
 - ✗ Bob Sutor, VP of IBM Q Strategy & Ecosystem [16]:

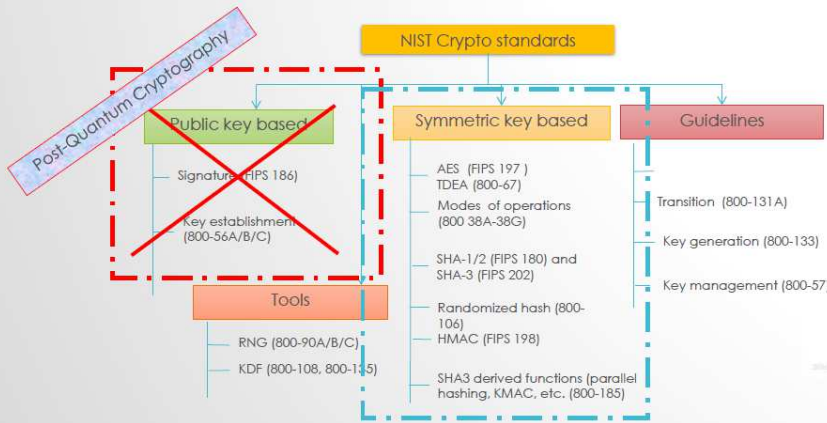
And if you're waiting for today's quantum computers to be able to compete with modern supercomputers anytime soon, you shouldn't hold your breath. "We need to get several orders of magnitude better than we are now to probably move into that period where we're solving the really super hard problems," he said.

- ✗ Some people even think that it cannot ever be reached, as mathematician Gil Kalai [9] recently mentioned:
 - ✗ **Quantum computing is** like any similar process in nature — **noisy**, with random fluctuations and **errors**. [...]
 - ✗ « We need what's known as **quantum error correction** [...] **the amount of noise has to go below a certain level, or threshold.** »
 - ✗ « [...] **our first result shows that the noise level cannot be reduced, because doing so will contradict an insight from the theory of computing** »
 - ✗ « So I don't need to be certain, **I can simply wait and see.** »

Current status & Emergency Level

THE SKY IS FALLING?

- If a large-scale quantum computer could be built then....



Dustin Moody, NIST [10]

The sky is falling?

- When will a quantum computer be built that breaks current crypto?
 - 15 years, \$1 billion USD, nuclear power plant (to break RSA-2048) (PQCrypto 2014, Matteo Mariantoni)

Dustin Moody, NIST [39]



Only a rash person would declare that there will be no useful quantum computers by the year 2050, but only a rash person would predict that there will be.

David Mermin [46]

"1 in 7 chance key cryptography tools will be broken by 2026 and a 50% chance by 2031 "

Author: Michele Mosca,
Institute for Quantum Computing & Special Advisor on Cyber Security to the Global Risk Institute

level of risk determination

y = Migration Time

x = Security Shelf Life

z = Time to Compromise

secret keys compromised

$$x + y > z$$

Mosca Theorem, source Medium [19]

Running Initiatives for Post Quantum Cryptography

NSA announcement & NIST call for submissions

- ✘ Back in august 2015, NSA explicitly talked about the threat of quantum computers
- ✘ "Unfortunately, the growth of **elliptic curve** use has bumped up against the fact of continued progress in the research on quantum computing, **necessitating a re-evaluation of our cryptographic strategy.**"
- ✘ "we recommend not making a significant expenditure to [make the transition to Suite B] at this point but **instead to prepare for the upcoming quantum resistant algorithm transition.**"
- ✘ In 2016: announcement of NIST's call for submissions on post-quantum public-key cryptography
- ✘ NB: Post-Quantum Cryptography is not Quantum Cryptography
 - ✘ Quantum Cryptography, Bennett and Brassard [45]
 - ✘ Use quantum mechanics to perform cryptographic operations
 - ✘ Post-Quantum Cryptography
 - ✘ Process quantum resistant cryptographic algorithms on regular devices

Timeline

**This is a tentative timeline, provided for information, and subject to change.*

Date	Event
Feb 24-26, 2016	NIST Presentation at PQCrypto 2016: <i>Announcement and outline of NIST's Call for Submissions (Fall 2016)</i> , Dustin Moody
April 28, 2016	NIST releases NISTIR 8105, Report on Post-Quantum Cryptography
Dec 20, 2016	Formal Call for Proposals
Nov 30, 2017	Deadline for submissions
Dec 4, 2017	NIST Presentation at AsiaCrypt 2017: <i>The Ship Has Sailed: The NIST Post-Quantum Crypto "Competition"</i> , Dustin Moody
Dec 21, 2017	Round 1 algorithms announced (69 submissions accepted as "complete and proper")
Apr 11, 2018	NIST Presentation at PQCrypto 2018: <i>Let's Get Ready to Rumble - The NIST PQC "Competition"</i> , Dustin Moody
April 11-13, 2018	First PQC Standardization Conference - Submitter's Presentations
2018/2019	Round 2 begins
August 2019 (tentative)	Second PQC Standardization Conference
2020/2021	Round 3 begins or select algorithms
2022/2024	Draft Standards Available

csrc.nist.gov [37]

Other initiatives: examples



Programme d'Investissements d'Avenir
GRANDS DÉFIS DU NUMÉRIQUE
RISQ
 Regroupement de l'Industrie française pour la
 Sécurité Post - Quantique
 "Gathering of the French Industry for Post-Quantum Security"

		
Industry	Certification Body	Academia
+ External Partners: 		



**PQCRYPTO
 ICT-645622**



Candidates Families for Post Quantum Cryptography

- ✘ Error Correcting Codes Cryptography
 - ✘ With Goppa codes
 - ✘ 1978, Mc Eliece [20]
 - ✘ Public keys for from 0.5 up to 1Mbyte
 - ✘ Slow key generation
 - ✘ With more structured codes (QC-MDPC)
 - ✘ Smaller Public Keys
 - ✘ Structure of code could lead to vulnerabilities
 - ✘ (Small) decryption error probability

- ✘ Hash Based Signatures
 - ✘ 1979, Lamport [25]
 - ✘ Recent proposals like SPHINCS (signatures of 41kbytes) or XMSS (signatures of 1-5 kbytes)

- ✘ Multivariate Cryptography
 - ✘ 1996, Patarin [23]
 - ✘ Large public keys (27.9 to 75 kbytes)

- ✘ Lattice Based Cryptography
 - ✘ 1996, Ajtai [21] and Regev [22]
 - ✘ NTRU, Keyber, Newhope, Frodo, ...
 - ✘ Well suited for ephemeral key exchange [41]
 - ✘ Pretty fast (equivalent to ECDH) on Cortex M4 [42] and Intel [41]

- ✘ Isogeny Based Key Exchange
 - ✘ 2006, Rostovtsev [24]
 - ✘ New field of research
 - ✘ Relatively slow

Candidates Families for Post Quantum Cryptography

Scheme		Public key size (bytes)	Data size (bytes)
Public-key signatures:			
• Hash based:			
– XMSS (stateful)	[17]	64	2,500 – 2,820
– SPHINCS (state free)	[9]	1,056	41,000
• Multivariate based:			
– HFEv-*	[51]	500,000 – 1,000,000	25 – 32
Public-key encryption:			
• Code based:			
– McEliece	[10]	958,482 – 1,046,739	187 – 194
• Lattice based:			
– NTRUEncrypt	[35, 37]	1,495 – 2,062	1,495 – 2,062
Key exchange:			
• Lattice based:			
– NewHope	[3]	—	1,824 – 2,048
• Supersingular isogenies:			
– SIDH	[21]	—	564
Classical schemes:			
• RSA:			
– RSA-2048		256	256
– RSA-4096		512	512
• ECC:			
– 256-bit		32	32
– 512-bit		64	64
• Key exchange:			
– DH		—	256 – 512
– ECDH		—	32 – 64

Source Fraunhofer [27]

× Noticeable Characteristics

- × Usually (much) bigger keys
- × Expected longer execution time with respect to current cryptography
- × Heterogeneous arithmetic/algorithmic solutions
- × Depends on family/refinement/targeted application

Consequences on implementations

- Fonctionnalités
- Security

Business as usual ?

- ✗ Post-Quantum crypto primitives will be run on classical electronic devices able to manage
 - ✗ Higher key size
 - ✗ Increased computational capabilities
 - ✗ Gaussian sampling capabilities (for lattice based crypto)
- ✗ Functionnalities requirements
 - ✗ Increased memory footprint and/or gate-count
 - ✗ Code Size
 - ✗ Key storage
 - ✗ Additional cryptographic engines or use of existing capabilities?
 - ✗ Dedicated Random Number Generators ?
 - ✗ Providing gaussian variables instead of usual uniform ones
- ✗ Some attacks have already been published for every 'old' family of PQ Cryptosystems
 - ✗ Timing Attacks (Strentzke et al. [28], [29], [31]) and SPA (Molter et al. [30], Heyse et al. [32]) on Mc Eliece
 - ✗ Timing Attack on NTRU (Silverman et al. [33])
 - ✗ Side Channel Attacks (Primas et al. [34]) and Fault Injection(Espitau et al. [35],[36]) on lattice-based schemes
 - ✗ Side Channel Attacks on Hash Based Signatures (Eisenbarth et al. [44])
- ✗ Security
 - ✗ Usual devices will leak in the usual way if no counter measure is applied
 - ✗ Possible new weaknesses / new ways for attacking

What's next in term of actual attacks?

- ✘ Well, I don't know ...
- ✘ Try to take advantage of long processing and big keys + algebraic structures / arithmetic patterns
 - ✘ Will PQ Cryptosystems be the flagship for Machine Learning: on the tracks of RSA and ECC ?
 - ✘ Time/Frequency analysis to look for patterns
 - ✘ Fault Injection to take advantage of the increased surface
- ✘ Would data be the favorite target (as for symmetric key crypto but also RSA) ?
 - ✘ Or rather treatments that reveals data (as for RSA and ECC) ?
- ✘ More complex software stacks that will have to deal with pre quantum and post quantum schemes (crypto agility)
 - ✘ The overall probability of a bug is increased
 - ✘ The attack surface is enlarged for fault injection and software attacks
- ✘ Gaussian TRNG
 - ✘ Usual (i.e. uniform) TRNG have not been widely attacked while being at the bosom of the security protocols
 - ✘ Gaussian TRNG are important for lattice based cryptography; they can then be a preferred target
 - ✘ Are gaussian TRNG more easily attackable than regular ones?
 - ✘ Successful attacks can turn into a generic trend, back to uniform TRNG targeting 'pre-quantum' cryptosystems
- ✘ Quantum computers are likely to boost machine learning, yielding advances in cryptanalysis ([38])

What is needed?

- ✘ Further studies !
 - ✘ Assess security of NIST candidates
 - ✘ Theoretical standpoint
 - ✘ With respect to side channel attacks and fault injection
 - ✘ Primitives to to efficiently & securely implement Post-Quantum schemes
 - ✘ Keep and adapt current crypto engines?
 - ✘ Develop brand new ones?
 - ✘ Ways to ensure proper & secure transition within products
 - ✘ Secure crypto agility required
 - ✘ Are current firmware upgrade mechanisms enough?
- ✘ Post-Quantum Cryptography in Gemalto
 - ✘ Follow up on theoretical and practical advances in the field
 - ✘ Involved in collaborative project RISQ
 - ✘ Several Proof of Concept running in Gemalto
- ✘ Gemalto to work on the subject through additionnal collaborative projects
 - ✘ Theoretical resistance study
 - ✘ Side Channel and Fault Injection resistance
 - ✘ Need for additionnal cryptographic bricks
 - ✘ Random Sampling capabilities

Take Away

- ✘ Quantum Computers are not there yet
 - ✘ Engineering problem to solve
 - ✘ Theoretical problems may also occur on the road
- ✘ But Post Quantum Cryptography would be needed at some point
 - ✘ AES-256, SHA-256 seems safe
 - ✘ ECC and RSA are in danger
- ✘ Probably moving in different steps as suggested by Shamir [43], depending on the maturity level
 - ✘ Production schemes (Hash Hased)
 - ✘ Development schemes (Lattice, Code Based, Multivariate)
 - ✘ Research schemes (Isogenies)
- ✘ Several proposals to emerge from the NIST initiative by 2024
- ✘ The robustness of those constructions have to be evaluated
 - ✘ Side Channel / Fault Injection
 - ✘ Software Attacks / Logical Attacks
- ✘ Provisionning needs to be defined
 - ✘ Additionnal arithmetic/algorithmic core bricks
 - ✘ Gaussian RNG
 - ✘ Counter measures for Side Channel Attacks and Fault Injection
 - ✘ Crypto agility capabilities to manage the transition

References

References

- × [1] Andrey Bogdanov; Dmitry Khovratovich & Christian Rechberger (2011). "Biclique Cryptanalysis of the Full AES«
- × [2] Selecting Cryptographic Key Sizes, Arjen K. Lenstra and Eric R. Verheul, Journal Of Cryptology, vol. 14, p. 255-293, 2001.
- × [3] Yearly Report on Algorithms and Keysizes (2012), D.SPA.20 Rev. 1.0, ICT-2007-216676 ECRYPT II, 09/2012.
- × [4] Mécanismes cryptographiques - Règles et recommandations, Rev. 2.03, ANSSI , 02/2014.
- × [5] Recommendation for Key Management, Special Publication 800-57 Part 1 Rev. 4, NIST, 01/2016.
- × [6] Kryptographische Verfahren: Empfehlungen und Schlüssellängen, TR-02102-1 v2017-01, BSI, 02/2017.
- × [7] Grover L.K. : A fast quantum mechanical algorithm for database search, Proceedings, 28th Annual ACM Symposium on the Theory of Computing, (May 1996) p. 212
- × [8] Peter W. Shor: Polynomial time algorithms for discrete logarithms and factoring on a quantum computer. ANTS 1994: 289
- × [9] The Argument Against Quantum Computers, <https://www.quantamagazine.org/gil-kalais-argument-against-quantum-computers-20180207/>
- × [10] THE SHIP HAS SAILED, The NIST Post-Quantum Crypto "Competition", Dustin Moody, NIST
- × [11] A. Hülsing, D. Butin, S.-L. Gazdag, and A. Mohaisen. XMSS: Extended Hash-based Signatures, July 2017. Work in Progress –
- × [12] Physical Attack Vulnerability of Hash-Based Signature Schemes, Master-Thesis von Matthias Julius Kannwischer, 2017
- × [13] Stefan Heyse. "Low-Reiter: Niederreiter encryption scheme for embedded microcontrollers." Pages 165–181 in: Nicolas Sendrier (editor). Post-Quantum Cryptography, Third international workshop, PQCrypto 2010. Lecture Notes in Computer Science 6061
- × [14] Aurons-nous un jour des ordinateurs quantiques ?, <https://www.franceculture.fr/emissions/la-conversation-scientifique/aurons-nous-un-jour-des-ordinateurs-quantiques>
- × [15] ETSI GR QSC 006 V1.1.1 (2017-02)
- × [16] Not even IBM is sure where its quantum computer experiments will lead, <https://www.engadget.com/2018/02/23/ibm-q-quantum-computer-experiments/>
- × [17] Markus Grassl, Brandon Langenberg, Martin Roetteler, Rainer Steinwandt: Applying Grover's Algorithm to AES: Quantum Resource Estimates. PQCrypto 2016: 29-43
- × [18] Marc Kaplan, Gaëtan Leurent, Anthony Leverrier, María Naya-Plasencia: Breaking Symmetric Cryptosystems Using Quantum Period Finding. CRYPTO (2) 2016: 207-237
- × [19] When Should You Start Worrying About Post-Quantum Cryptography?, <https://medium.com/@EncryptKen/when-should-you-start-worrying-about-post-quantum-cryptography-c881102c3f73>
- × [20] Robert J. McEliece, « A Public-Key Cryptosystem Based on Algebraic Coding Theory », Jet Propulsion Laboratory DSN Progress Report,? 1978, p. 42–44

References

- × [21] Mikiós Ajtai, « Generating hard instances of lattice problems », Symposium on Theory of Computing, 1996
- × [22] Oded Regev, « On lattices, learning with errors, random linear codes, and cryptography », Symposium on Theory of Computing, 2005
- × [23] Patarin, J. Hidden fields equations (HFE) and isomorphisms of polynomials (IP): two new families of asymmetric algorithms. In Maurer, U. M. (ed.) Advances in Cryptology - EUROCRYPT '96
- × [24] Alexander Rostovtsev et Anton Stolbunov, « Public-Key Cryptosystem based on Isogenies », *iacr ePrint Reports*, 2006
- × [25] Lamport, L. Constructing digital signatures from a one way function. Technical Report SRI-CSL-98, SRI International Computer Science Laboratory (1979)
- × [26] Daniel J. Bernstein, Nadia Heninger, Paul Lou et Luke Valenta, « Post-quantum RSA », Post-Quantum Cryptography, Springer, Cham, série Lecture Notes in Computer Science, 26 juin 2017, p. 311–329
- × [27] White Paper Practical PostQuantum Cryptography August 18, 2017 Dr. Ruben Niederhagen Department CyberPhysical Systems Security (CSS) Prof. Dr. Michael Waidner Director Fraunhofer SIT and Professor for Security in IT at TU Darmstadt Fraunhofer Institute for Secure Information Technology SIT
- × [28] Falko Strenzke, Erik Tews, H. Gregor Molter, Raphael Overbeck, Abdulhadi Shoufan: Side Channels in the McEliece PKC. PQCrypto 2008: 216-229
- × [29] Falko Strenzke: A Timing Attack against the Secret Permutation in the McEliece PKC. PQCrypto 2010: 95-107
- × [30] H. Gregor Molter, Marc Stöttinger, Abdulhadi Shoufan, Falko Strenzke: A simple power analysis attack on a McEliece cryptoprocessor. J. Cryptographic Engineering 1(1): 29-36 (2011)
- × [31] Falko Strenzke: Timing Attacks against the Syndrome Inversion in Code-Based Cryptosystems. PQCrypto 2013: 217-230
- × [32] Stefan Heyse, Amir Moradi, Christof Paar: Practical Power Analysis Attacks on Software Implementations of McEliece. PQCrypto 2010: 108-125
- × [33] Joseph H. Silverman, William Whyte: Timing Attacks on NTRUEncrypt Via Variation in the Number of Hash Calls. CT-RSA 2007: 208-224
- × [34] Robert Primas, Peter Pessl, Stefan Mangard: Single-Trace Side-Channel Attacks on Masked Lattice-Based Encryption. CHES 2017: 513-533
- × [35] Thomas Espitau, Pierre-Alain Fouque, Benoît Gérard, Mehdi Tibouchi: Side-Channel Attacks on BLISS Lattice-Based Signatures: Exploiting Branch Tracing against strongSwan and Electromagnetic Emanations in Microcontrollers. CCS 2017: 1857-1874
- × [36] Thomas Espitau, Pierre-Alain Fouque, Benoît Gérard, Mehdi Tibouchi: Loop-Abort Faults on Lattice-Based Fiat-Shamir and Hash-and-Sign Signatures. SAC 2016: 140-158
- × [37] <https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Workshops-and-Timeline>
- × [38] Job One for Quantum Computers: Boost Artificial Intelligence, <https://www.quantamagazine.org/job-one-for-quantum-computers-boost-artificial-intelligence-20180129/>
- × [39] Dustin Moody, Post-Quantum Cryptography: NIST's Plan for the Future, PQCrypto 2016
- × [40] Henri Gilbert, Selection of Cryptographic Algorithms, Post-Quantum Cryptography: ANSSI Views
- × [41] Thomas Pöppelmann, Post-Quantum Cryptography, Infineon Technologies AG, Invited Talk Cardis 2017
- × [42] Alkim, Jakubeit, Schwabe: NewHope on ARM Cortex-M. SPACE 2016
- × [43] Adi Shamir, PQ-Crypto: A New Proposed Framework, Computer Science Dept The Weizmann Institute Israel, NIST First PQC Standardization Conference, April 2018
- × [44] Thomas Eisenbarth, Ingo von Maurich, Xin Ye: Faster Hash-Based Signatures with Bounded Leakage. SAC 2013
- × [45] Charles H. Bennett, Gilles Brassard: An Update on Quantum Cryptography. CRYPTO 1984: 475-480
- × [46] David Mermin, Quantum Computation Lecture Notes and Homework Assignments Cornell, Spring 2006

Low-cost Setup for Localized Semi-invasive Optical Fault Injection Attacks

Oscar M. Guillen¹ Michael Gruber² Fabrizio De Santis²

¹ Giesecke & Devrient

² Technical University of Munich

May 23, 2018



TUM Uhrenturm

Table of contents

- 1 Introduction
 - Motivation
 - Fault Injection Techniques
- 2 Evaluation Framework
 - Fault Injection Setup
 - Preparation
 - Fault Characterization
- 3 Application to SPECK
 - SIMON and SPECK
 - Instruction Skip
 - Random Fault
- 4 Summary

- Fault Injection in practice:
 - Are local optical attacks feasible using **low cost** equipment (\sim €500)?
 - What kind of **faults** can be generated?

The cost of the equipment is important for security evaluation

- Attack rating
 - Equipment
 - Level of expertise

- Low-cost devices
 - Microcontroller-based devices
 - IoT endpoints

Technique	Accuracy (Spatial)	Accuracy (Temporal)	Cost	Risk (Damage)
Clock glitch	none	high	low	none
Voltage spike	none	high	low	low
Heat	low	none	low	low
EM Pulse	medium	medium	medium	medium
Laser beam	high	high	high	medium

Table: Summary of non-invasive fault injection techniques [1]

Optical Fault Injection

Complete fault injection stations cost up to €150k [3]

- Light source
 - Flashgun, for older technology nodes [6]
 - Laser, newer technologies
- Focusing element
 - Visible-light microscope
 - Infrared microscope and camera
- Positioning
 - X-Y Stage

Low-cost Optical Fault Injection

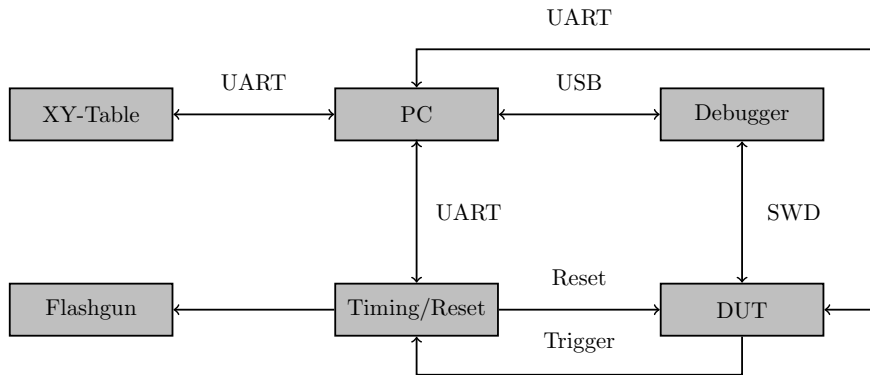
Our low-cost fault injection setup \sim €500

- Light source
 - Flashgun
- Focusing element
 - **Ball lens** 'microscope'
- Positioning
 - X-Y Micro-milling stage (5 μ m resolution)
 - Motor control using grbl [5]
 - Z-axis operated manually
- DUT's minimal setup boards
 - AVR 8-bit,
Atmega328p, 350 nm
 - ARM Cortex M0 32-bit,
STM32F030F4P6, 90 nm

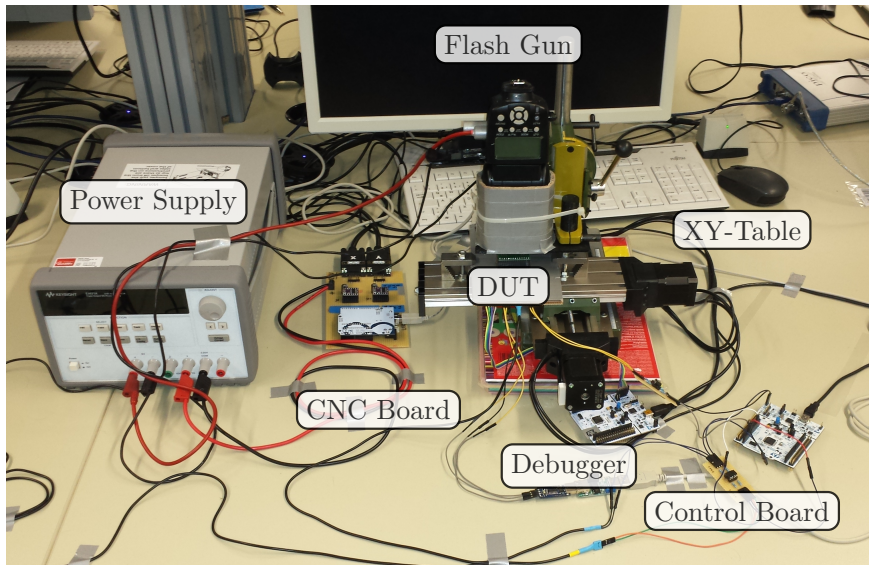
Table of contents

- 1 Introduction
 - Motivation
 - Fault Injection Techniques
- 2 Evaluation Framework
 - Fault Injection Setup
 - Preparation
 - Fault Characterization
- 3 Application to SPECK
 - SIMON and SPECK
 - Instruction Skip
 - Random Fault
- 4 Summary

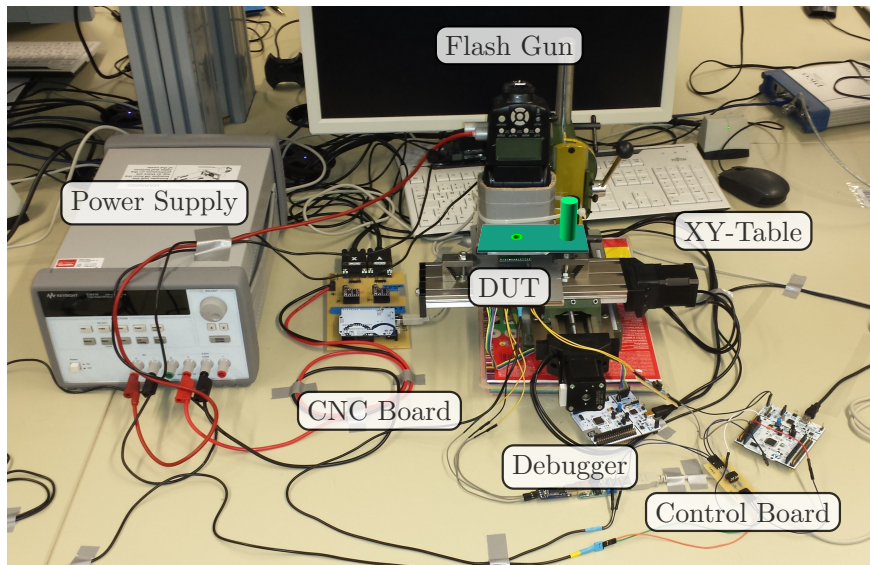
Block Diagram



Fault Injection Setup



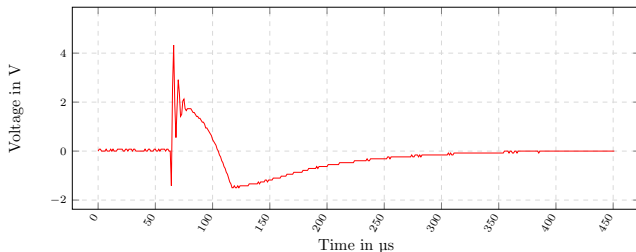
Fault Injection Setup



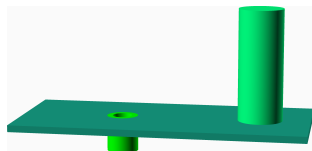
Fault Injection Setup

Light source

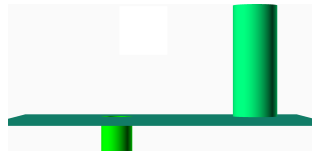
- Flashgun
- Trigger Delay of $64 \mu\text{s}$
(measured using a LED to sense emitted light)



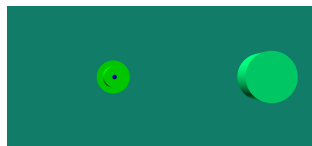
3D Printed Mount for the Optics



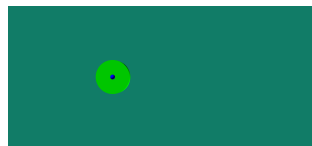
(a) Side I



(b) Side II



(c) Top



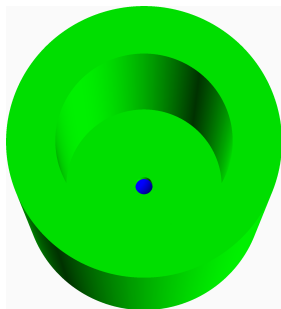
(d) Bottom

- Ball Lens
- Diameter 1 mm
- Substrate N-BK7
- Wavelength 350 nm to 2200 nm
- Diameter Tolerance $\pm 2.5 \mu\text{m}$
- Back Focal Length (BFL) 0.23 mm
- Mounted in 3d printed socket

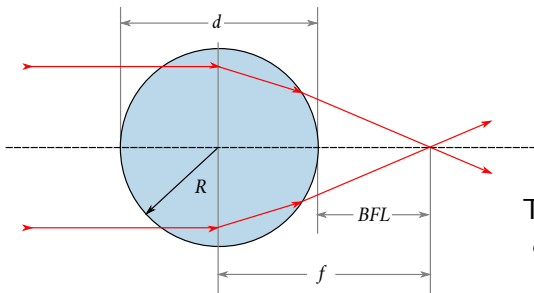


Front-View

- Ball Lens
- Diameter 1 mm
- Substrate N-BK7
- Wavelength 350 nm to 2200 nm
- Diameter Tolerance $\pm 2.5 \mu\text{m}$
- Back Focal Length (BFL) 0.23 mm
- Mounted in 3d printed socket



Top-View



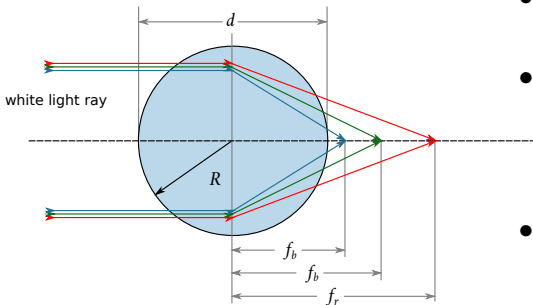
Ball lens focal point,

$$\frac{1}{f} = \frac{4(n-1)}{n \cdot d} \quad (1)$$

The magnification M of a lens compared to a human eye is:

$$M = \frac{250 \text{ mm}}{f} \quad (2)$$

for a 1.0 mm diameter, N-BK7 borosilicate-glass ball lens $n = 1.517$
 $f = 0.73356 \text{ mm}$, $BFL = 0.23356 \text{ mm}$, $M = 340\times$

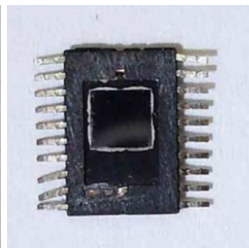
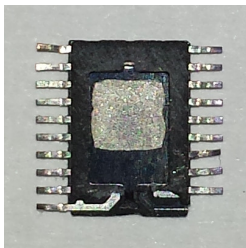
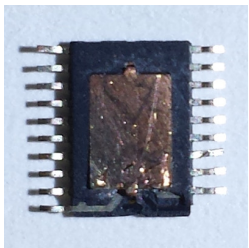
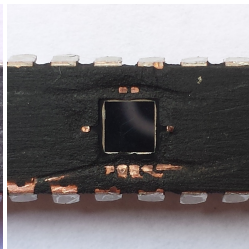
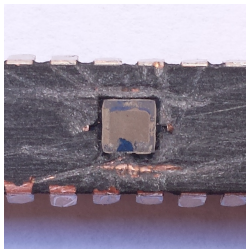
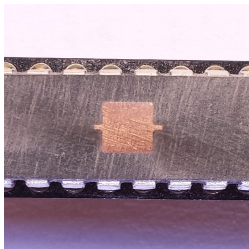


- White light is composed of different wavelengths
- Light components are dispersed according to their frequency (chromatic aberration)
- Infrared component (wavelength >715 nm) is present in the light generated by the flashgun and focused through the ball lens

Preparation I

- Semi-invasive attacks require a decapsulated DUT
 - Frontside: dangerous, using chemicals
 - Backside: easy, but no visible structures
- Decapsulation procedure:
 - 1 Grind down the backside using sandpaper
 - 2 Pry the lead frame open using a knife
 - 3 Clean the chip from glue using acetone

	ARM Cortex M0	AVR
Package	TSSOP	PDIP
Grinding	—	—
Opening	—	+
Cleaning	—	+



(a) Sanding

(b) Removing

(c) Cleaning

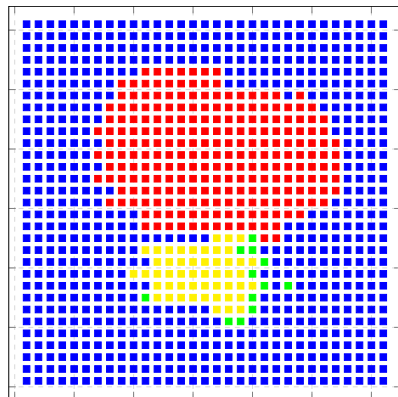
- Instruction Skip Test (global/local)
 - 1 Execute function
 - 2 Inject fault
 - 3 Check result
- RAM Faults (global/local)
 - 1 Write pattern to RAM
 - 2 Inject fault
 - 3 Check result
- Register Faults (local)
 - 1 Pre-set user accessible registers
 - 2 Inject fault
 - 3 Read back registers
- Procedure:
 - 1 Generate meander pattern
 - 2 Perform test
 - 3 Read result
 - 4 Update position
 - 5 goto #2

Fault Injection Results

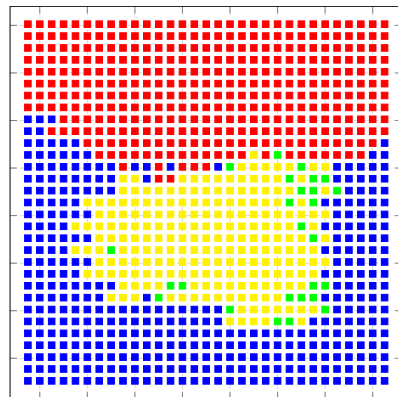
	Atmega328p (350 nm)		STM32F030F4P6 (90 nm)	
	local	global	local	global
Instruction Skip	✗	✓	✓	✗
Register Change	✗	✗	✓	✗
RAM Change	✓	✗	✗	✗

ARM Cortex M0 32-bit, 90 nm, (STM32F030F4P6)

■ Reset, ■ No change, ■ Exploitable fault, ■ Non-exploitable fault



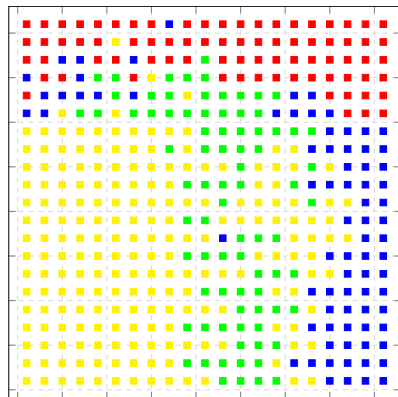
(a) Whole Chip, 0.1 mm, 3 mm \times 3 mm



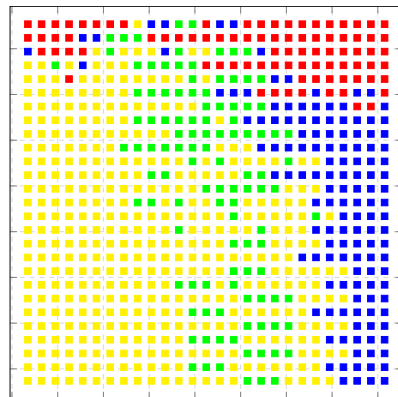
(b) ROI-1, 0.05 mm, 1.5 mm \times 1.5 mm

ARM Cortex M0 32-bit, 90 nm, (STM32F030F4P6)

■ Reset, ■ No change, ■ Exploitable fault, ■ Non-exploitable fault



(c) ROI-2, 0.02 mm, 0.4 mm \times 0.4 mm



(d) ROI-3, 0.015 mm, 0.4 mm \times 0.4 mm

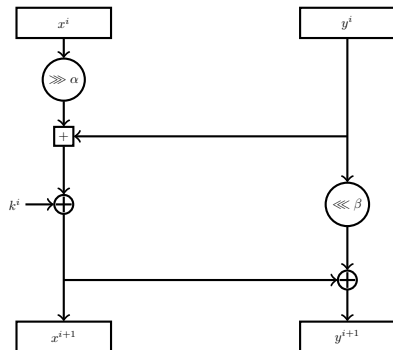
Table of contents

- 1 Introduction
 - Motivation
 - Fault Injection Techniques
- 2 Evaluation Framework
 - Fault Injection Setup
 - Preparation
 - Fault Characterization
- 3 Application to SPECK**
 - SIMON and SPECK
 - Instruction Skip
 - Random Fault
- 4 Summary

- Published by the NSA in 2013 [2]
- Lightweight block ciphers
- Perform well on resource constrained devices
- SIMON targets HW implementations
- SPECK targets SW implementations
- Each algorithm allows 10 different combinations of block/key size

block size	key size
32	64
48	72, 96
64	96, 128
96	96, 114
128	128, 192, 256

- Feistel-like structure
- ADD, ROT, XOR (ARX)
- T 22-34 rounds
- Break the 2,3,4 last rounds to recover key, depending on key size
- Key Schedule reuses the round function
- State y^{T-1} known

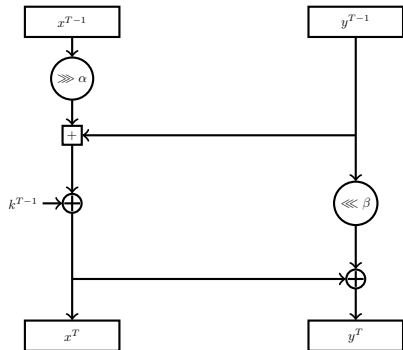


$$R(x, y) = (f(x, y) \oplus k, y \lll \beta \oplus f(x, y) \oplus k) \text{ where } f(x, y) = x \ggg \alpha + y$$

- What kind faults can we generate?
- What kind of faults can we exploit?

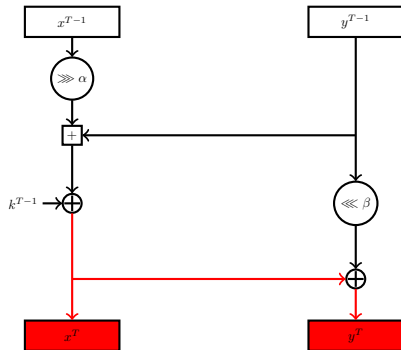
Instruction Skip

- AVR - *global setup*
- STM32 - *local setup*
- Skip XOR with k^{T-1}
- Less than 1 second
- Only 1 injection needed
- Recover k^{T-1} completely
- Same outcome in 80% of the injections



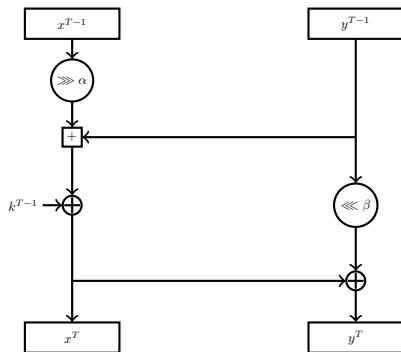
Instruction Skip

- AVR - *global setup*
- STM32 - *local setup*
- Skip XOR with k^{T-1}
- Less than 1 second
- Only 1 injection needed
- Recover k^{T-1} completely
- Same outcome in 80% of the injections



Random Fault/Register Fault [4]

- STM32 - *local setup*
- Random fault model at y^{T-1}
- Attack takes ≈ 1 h
- Attack needs $\approx 3 \times 10^3$ injections
- 46 faulty pairs recovered
- Recovers $n - 1$ bits of k^{T-1}
(MSB cannot be recovered due to the modular addition)



Random Fault/Register Fault [4]

- STM32 - *local setup*
- Random fault model at y^{T-1}
- Attack takes ≈ 1 h
- Attack needs $\approx 3 \times 10^3$ injections
- 46 faulty pairs recovered
- Recovers $n - 1$ bits of k^{T-1}
(MSB cannot be recovered due to the modular addition)

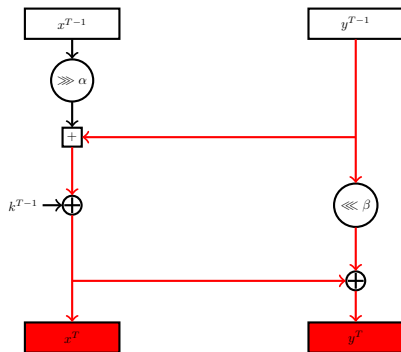


Table of contents

- 1 Introduction
 - Motivation
 - Fault Injection Techniques
- 2 Evaluation Framework
 - Fault Injection Setup
 - Preparation
 - Fault Characterization
- 3 Application to SPECK
 - SIMON and SPECK
 - Instruction Skip
 - Random Fault
- 4 Summary

Summary

- Low cost localized fault injection setup
 - <https://github.com/open-fi/fault-injector>
- Backside fault injection
 - Cheap ball lens enables backside attacks with flashgun
 - Performed in unthinned devices
- Faults observed on 90 nm MCUs
 - Register manipulation
 - Instruction skip

Implications

- High security devices might already have countermeasures in place (e.g. optical sensors)
- Low-cost, microcontroller-based, devices should consider low-cost optical attacks as a serious threat

- Different light sources
- Different types and sizes of focusing elements
- Pattern-based triggering
- EM Fault Injection

Function	Description	Price (EUR)
Optics		
Flashgun	YN560 III	60
Ball lens	1 mm N-BK7	25
Positioning		
X-Y Table	Proxxon KT 70	263
Stand	Proxxon Stand	70
Control	Arduino UNO	20
Drivers	DRV8825	18
Control and Debugging		
Control Board	STM32 Nucleo F411RE	12
Debugger	STM32 Nucleo F411RE (OpenOCD)	12
Miscellaneous		
	Sand paper, mask, latex gloves, acetone	26
		506

Thank you for your attention!

- [1] Alessandro Barenghi, Luca Breveglieri, Israel Koren, and David Naccache. Fault injection attacks on cryptographic devices: Theory, practice, and countermeasures. *Proceedings of the IEEE*, 100(11):3056–3076, 2012.
- [2] Ray Beaulieu, Douglas Shors, Jason Smith, Stefan Treatman-Clark, Bryan Weeks, and Louis Wingers. The simon and speck families of lightweight block ciphers. Cryptology ePrint Archive, Report 2013/404, 2013. <http://eprint.iacr.org/>.
- [3] Jakub Breier and Dirmanto Jap. Testing feasibility of back-side laser fault injection on a microcontroller. In *Proceedings of the WESS'15: Workshop on Embedded Systems Security*, WESS'15, pages 5:1–5:6, New York, NY, USA, 2015. ACM. ISBN 978-1-4503-3667-3. doi: 10.1145/2818362.2818367. URL <http://doi.acm.org/10.1145/2818362.2818367>.
- [4] Yuming Huo, Fan Zhang, Xiutao Feng, and Li-Ping Wang. Improved differential fault attack on the block cipher speck. In *2015 Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC)*, pages 28–34. IEEE, 2015.
- [5] Sungeun K. Jeon, 2016. <https://github.com/grbl/grbl>.
- [6] Sergei P. Skorobogatov and Ross J. Anderson. Optical fault induction attacks. In *Cryptographic Hardware and Embedded Systems - CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, August 13-15, 2002, Revised Papers*, pages 2–12, 2002. doi: 10.1007/3-540-36400-5_2. URL http://dx.doi.org/10.1007/3-540-36400-5_2.



Fault attacks on System On Chip

Thomas TROUCHKINE Guillaume BOUFFARD Jessy CLÉDIÈRE

ANSSI - Hardware Security Labs

May 22, 2018

Context



Smartcard



Mobile device

Same services, different securities

Context



Based on a Secure Element

- Simple SoC
- Designed for security
- Evaluated



Based on a Computer on Chip

- Complex SoC
- Designed for performance
- Adding TEE¹ for software security

¹Trusted Environment Execution

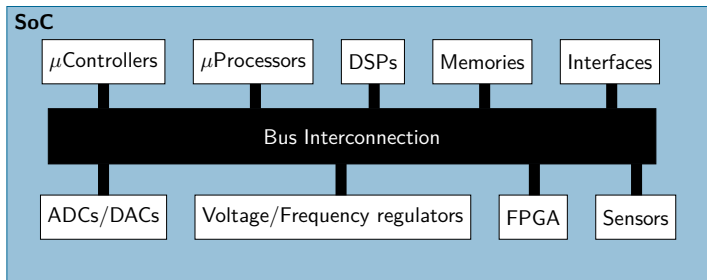
Hardware attacks ?

Fault attacks

- Laser/EM injection
- Clock glitch
- Voltage glitch
- Rowhammer
- Heating
- Body biasing

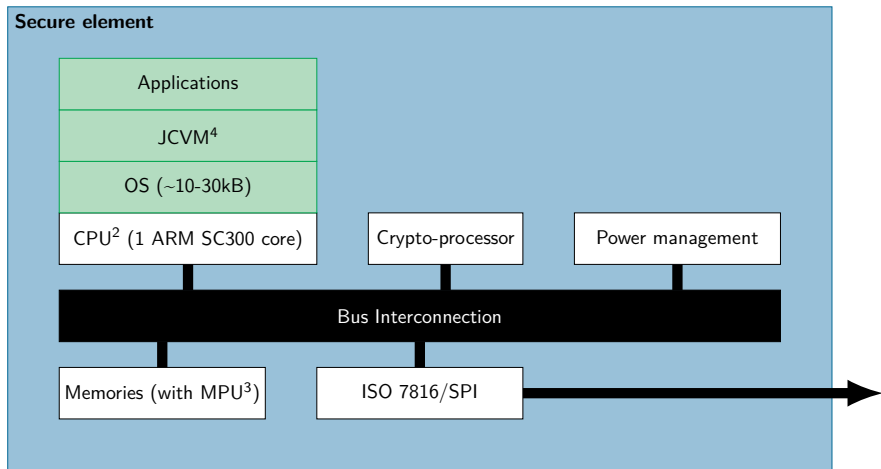


What is a System on Chip ?



- Integrate all components on the same chips
- Reduce power consumption
- Reduce chip size

What is a Secure Element ?



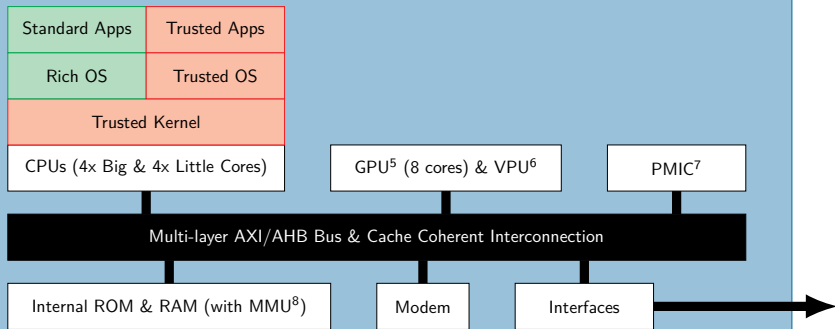
²Central Processing Unit

³Memory Protection Unit

⁴Java Card Virtual Machine

What is a Computer on Chip ?

Computer on Chip (Exynos like)



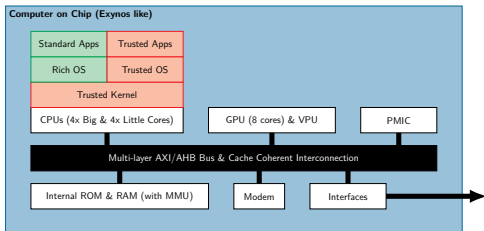
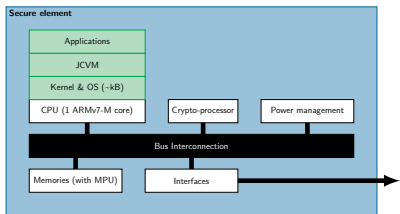
⁵ Graphical Processing Unit

⁶ Video Processing Unit

⁷ Power Management Integrated Circuit

⁸ Memory Management Unit

Secure element vs Computer on Chip

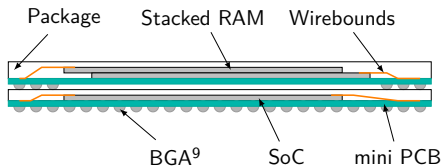


- Run at 4 to 60MHz
- Not multithreaded
- Fine engraving > 40 nm
- Constant Voltage & Frequency
- Trusted hardware & Trusted apps only
- Hardware mitigations

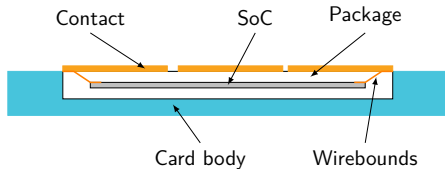
- Run at 300MHz to 3Ghz
- Multithreaded
- Fine engraving < 20 nm
- Dynamic Voltage & Frequency management
- Trusted Environment Execution
- No hardware mitigations

The packaging

Computer on Chip package on package



Secure element package



⁹Ball Grid Array

Assets to protect

- Cryptographic secrets and operations
- Secure boot
- Memory partitioning
- Execution flow integrity
- Trusted part isolation



Unknowns

- Repeatability ?
- Design impact ?
- Technology impact ?
- New attack paths ?



Soooo let's start !

- Computer on Chip → software security only
- Hardware quite similar with Secure Elements
- Some attacks already exist:
 - 1 Evaluate their difficulty
 - 2 Push some uncompleted attacks
 - 3 Find new paths

Known attacks

Injection medium	Physical target	Software target	Software security
Software	RAM	Virtual to physical translation table	Memory partitioning
Glitch voltage	Clock	Key	Cryptography
Laser	Register	Instruction	Secure boot
EM	Bus	Return value	Execution flow integrity
	Cache	Program counter	
	MMU	User rights	
	Pipeline		

Known attacks

Injection medium	Physical target	Software target	Software security
Software	RAM	Virtual to physical translation table	Memory partitioning
Glitch voltage	Clock	Key	Cryptography
Laser	Register	Instruction	Secure boot
EM	Bus	Return value	Execution flow integrity
	Cache	Program counter	
	MMU	User rights	
	Pipeline		

Project Zero attack/Drammer (2015 - 2016) [Vee+16]

Known attacks

Injection medium	Physical target	Software target	Software security
Software	RAM	Virtual to physical translation table	Memory partitioning
Glitch voltage	Clock	Key	Cryptography
Laser	Register	Instruction	Secure boot
EM	Bus	Return value	Execution flow integrity
	Cache	Program counter	
	MMU	User rights	
	Pipeline		

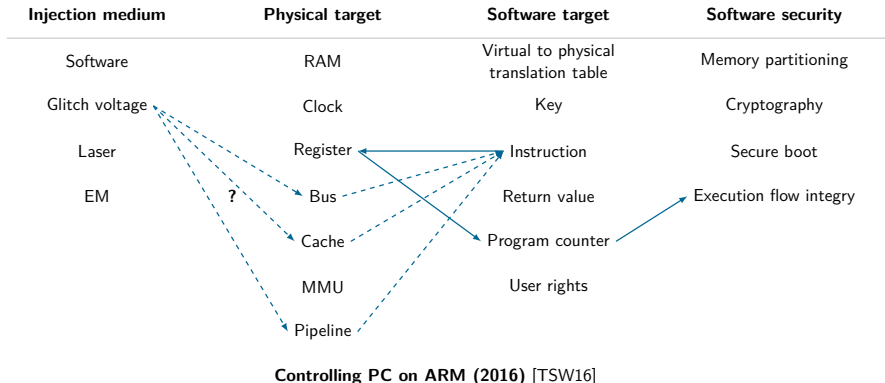
Project Zero NaCl/Rowhammer on TrustZone (2015) [Car17]

Known attacks

Injection medium	Physical target	Software target	Software security
Software	RAM	Virtual to physical translation table	Memory partitioning
Glitch voltage	Clock	Key	Cryptography
Laser	Register	Instruction	Secure boot
EM	Bus	Return value	Execution flow integrity
	Cache	Program counter	
	MMU	User rights	
	Pipeline		

ClkScrew (2017) [AS17]

Known attacks



Known attacks

Injection medium	Physical target	Software target	Software security
Software	RAM	Virtual to physical translation table	Memory partitioning
Glitch voltage	Clock	Key	Cryptography
Laser	Register	Instruction	Secure boot
EM	Bus	Return value	Execution flow integrity
	Cache	Program counter	
	MMU	User rights	
	Pipeline		

Attack on PS3

Known attacks

Injection medium	Physical target	Software target	Software security
Software	RAM	Virtual to physical translation table	Memory partitioning
Glitch voltage	Clock	Key	Cryptography
Laser	Register	Instruction	Secure boot
EM	Bus	Return value	Execution flow integrity
	Cache	Program counter	
	MMU	User rights	
	Pipeline		

Attack on Xbox 360 (2015) [Bla15]

Known attacks

Injection medium	Physical target	Software target	Software security
Software	RAM	Virtual to physical translation table	Memory partitioning
Glitch voltage	Clock	Key	Cryptography
Laser	Register	Instruction	Secure boot
EM	Bus	Return value	Execution flow integrity
	Cache	Program counter	
	MMU	User rights	
	Pipeline		

Laser induced fault on smartphone (2017) [Vas+17]

Conclusion

- Migration of services from Secure Element to Computer on Chip
- Hardware security gap
 - SE is a full trusted environment
 - Computer on chip integrate a software trusted environment
- Invasive/Semi-invasive attacks feel harder on Computer on Chip
- New attack paths

Questions?

References

- [AS17] Simha Sethumadhavan Adrian Tang and Salvatore Stolfo. *CLKSCREW: Exposing the perils of security-oblivious energy management*. Tech. rep. Columbia University, 2017.
- [Bla15] BlackHat. “XBOX 360 Glitching on fault attack”. Nov. 2015.
- [Car17] Pierre Carru. “Attack TrustZone with Rowhammer”. In: eshard. 2017.

- [TSW16] Niek Timmers, Albert Spruyt, and Marc Witteman. “Controlling PC on ARM Using Fault Injection”. In: *2016 Workshop on Fault Diagnosis and Tolerance in Cryptography, FDTC 2016, Santa Barbara, CA, USA, August 16, 2016*. IEEE Computer Society, 2016, pp. 25–35. DOI: 10.1109/FDTC.2016.18.
- [Vas+17] Aurélien Vasselle et al. “Laser-induced fault injection on smartphone bypassing the secure boot”. In: (Sept. 2017).
- [Vee+16] Victor van der Veen et al. “Drammer: Deterministic Rowhammer Attacks on Mobile Platforms”. In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, Vienna, Austria, October 24-28, 2016*. Ed. by Edgar R. Weippl et al. ACM, 2016, pp. 1675–1689. DOI: 10.1145/2976749.2978406.



Compiler-assisted Loop hardening against fault attacks

Julien PROY¹, Karine HEYDEMANN², Alexandre BERZATI¹, Albert COHEN³

¹Invia Secure Semiconductor Meyreuil, France

²Sorbonne Université, UPMC Univ Paris 06, CNRS, Lip6, Paris, France

³INRIA and DI, Ecole Normale Supérieure, Paris, France

Published in **TACO** (*Transactions on Architecture and Code Optimization*)

Presented at #HiPeac 2018, **Manchester**

May 23-24th | **Phisic 2018**

Agenda

- **Introduction & context**
 - Embedded systems
 - Physical attacks and countermeasures
- **Loop hardening scheme**
 - Principle & main algorithm
 - Implementation in LLVM
- **Experimental results**
- **Conclusion**

IoT & Embedded systems

- 2000's: Secure devices designed to be resilient to physical attacks...



IoT & Embedded systems

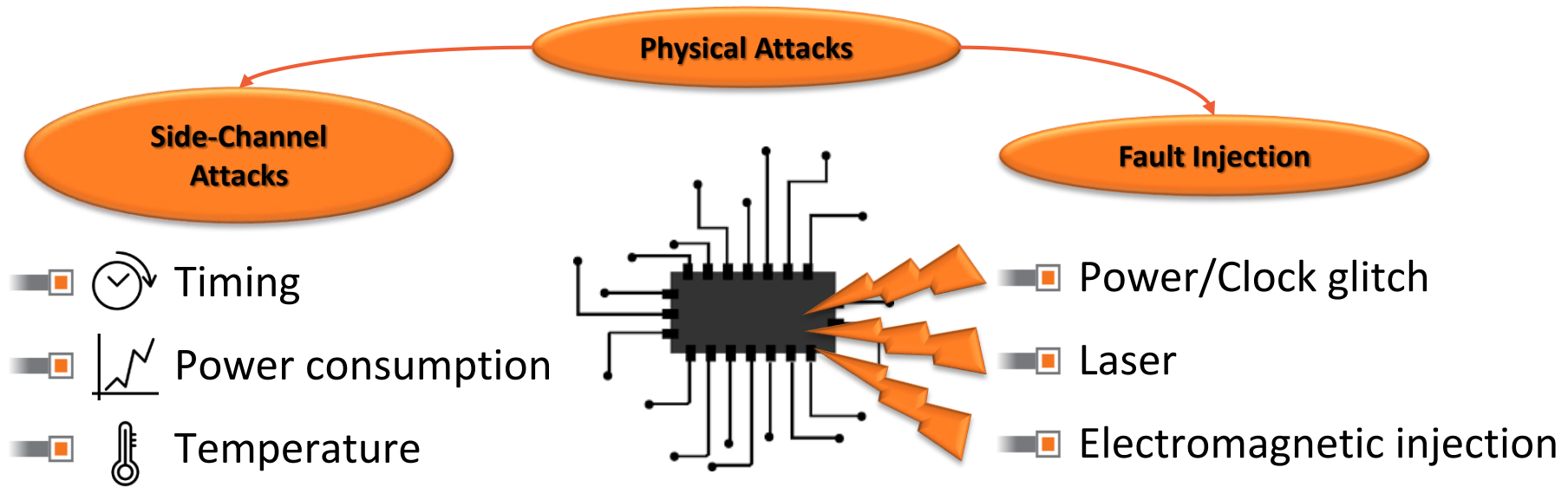
- 2000's: Secure devices designed to be resilient to physical attacks...



- 2010's: Many other devices...beyond traditional secure ones:



Physical attacks



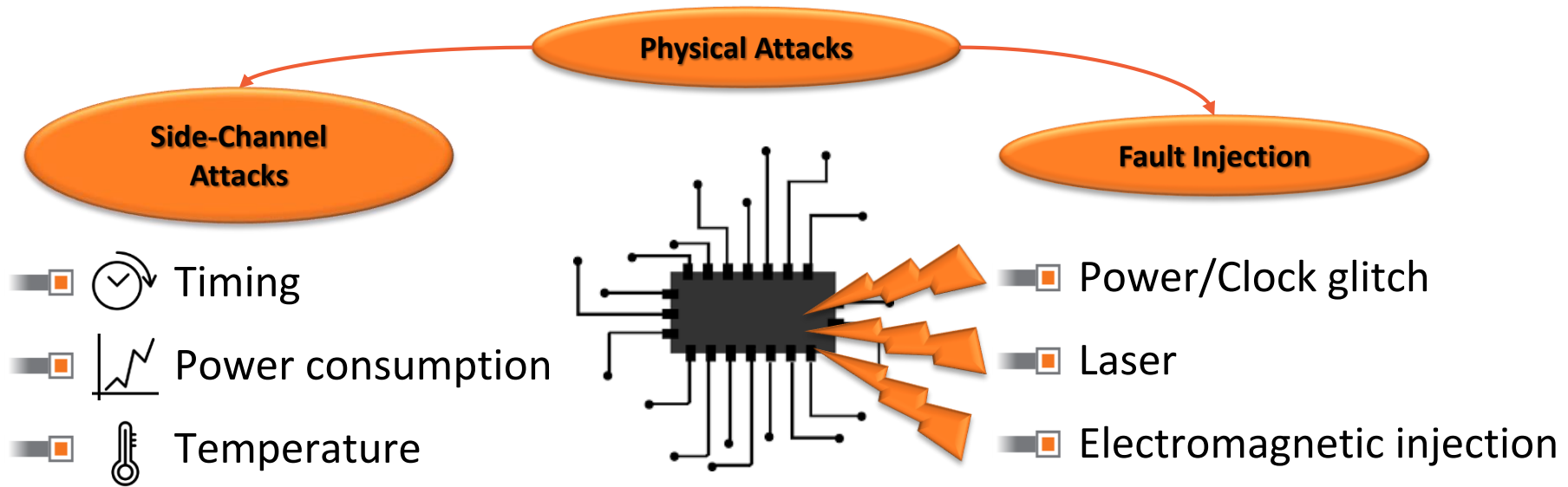
Common goal: Retrieve sensitive data / bypass protections

[1999] Kocher et al.

[2006] Bar-El et al.

[2013] Karaklajic et al.


Physical attacks



Common goal: Retrieve sensitive data / bypass protections

- Example of attack: fault during firmware update

```
i = rand;
int i = 0;
while (i != n) {
    dst[i] = src[i];
    i++;
}
```

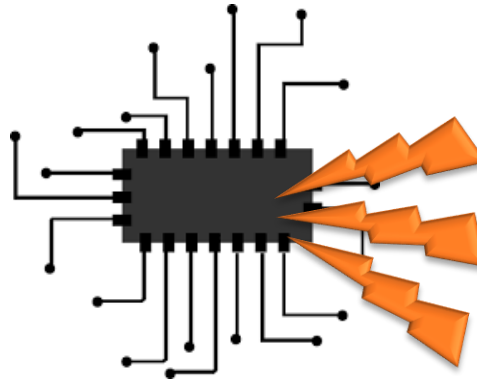


Physical attacks

Physical Attacks

Side-Channel Attacks

- ☐ Timing
- ☐ Power consumption
- ☐ Temperature



Fault Injection

- ☐ Power/Clock glitch
- ☐ Laser
- ☐ Electromagnetic injection

- ☐ Need for protection
 - ☐ **Hardware** and/or **software** countermeasures
 - ☐ **Silicon area** increase/**performances** slowdown
 - ☐ **Expensive** development cost
- ☐ Need for automation (manual today)

Focus on **automatic software** countermeasures against **fault** attacks

Targeted sensitive code

■ Several attacks target loop iteration

- On cryptographic code: **AES** [Demirci et al. 2008][Dehbaoui et al. 2013]
- On system code: **Buffer overflow** [Nashimoto et al. 2016], **PIN verification** [Dureuil et al. 2016]

■ Security properties to preserve

- Ensure the **number of iterations**
- Ensure the **right exit** is taken

```
int diff = 0;
for (int i=0; i<n; i++) {
    if (A[i] != B[i]) {
        diff = 1;
        break;
    }
}
...
```

■ Against which effects?

Fault model

- The design of countermeasures requires a precise fault model
- **Characterizes** effects of a fault attack

- Fault models found in literature
 - Single/multiple bit set/bit reset/bit flip in register
 - Register random corruption
 - Instruction replacement / instruction skip
 - Random alteration of transfers between the CPU and non-volatile memory

Fault model

- The design of countermeasures requires a precise fault model
- **Characterizes** effects of a fault attack

- Fault models found in literature
 - Single/multiple bit set/bit reset/bit flip in register
 - Register random corruption
 - Instruction replacement / instruction skip
 - Random alteration of transfers between the CPU and non-volatile memory

- Popular fault models considered in this work
 - **Instruction skip**
 - Global purpose **register** random **corruption**

Software countermeasures against fault attacks



Compilation



Software countermeasures against fault attacks



Compilation



[2014] *Lalande et al.*

Source code level

+ More human friendly

- Compiler optimization compromise security

- Still require manual assembly check

- No correspondence to fault model

```
int i = 0, j = 0;
while (i != n) {
    dst[i] = src[i];
    i++;
    j++;
}
```

Software countermeasures against fault attacks



Compilation



[2014] *Lalande et al.*

Source code level

+ More human friendly

- Compiler optimization compromise security

- Still require manual assembly check

- No correspondence to fault model

```
int i = 0, j = 0;
while (i != n) {
    dst[i] = src[i];
    i++;
    j++;
}
```

Software countermeasures against fault attacks



Compilation



[2014] *Lalande et al.*

[2016] *De Keulenaer et al.*

Source code level

- + More human friendly
- Compiler optimization compromise security
- Still require manual assembly check
- No correspondence to fault model

```
int i = 0, j = 0;
while (i != n) {
    dst[i] = src[i];
    i++;
    j++;
}
```

Binary level

- + More realistic fault model
- + Source code not required
- Lack of semantic information
- How to find available registers

```
...
10010100100101011001
11001010010100010101
11001010010100010101
11001010101111001010
...
```

Software countermeasures against fault attacks



Compilation

[2014] *Lalande et al.*

[2016] *Barry et al.*

[2016] *De Keulenaer et al.*

Source code level

- + More human friendly
- Compiler optimization compromise security
- Still require manual assembly check
- No correspondence to fault model

Compilation time

- + Fully automatic
- + Leverage compiler analysis
- + Limited overhead
- + Promising and less studied
- Downstream optimizations

Binary level

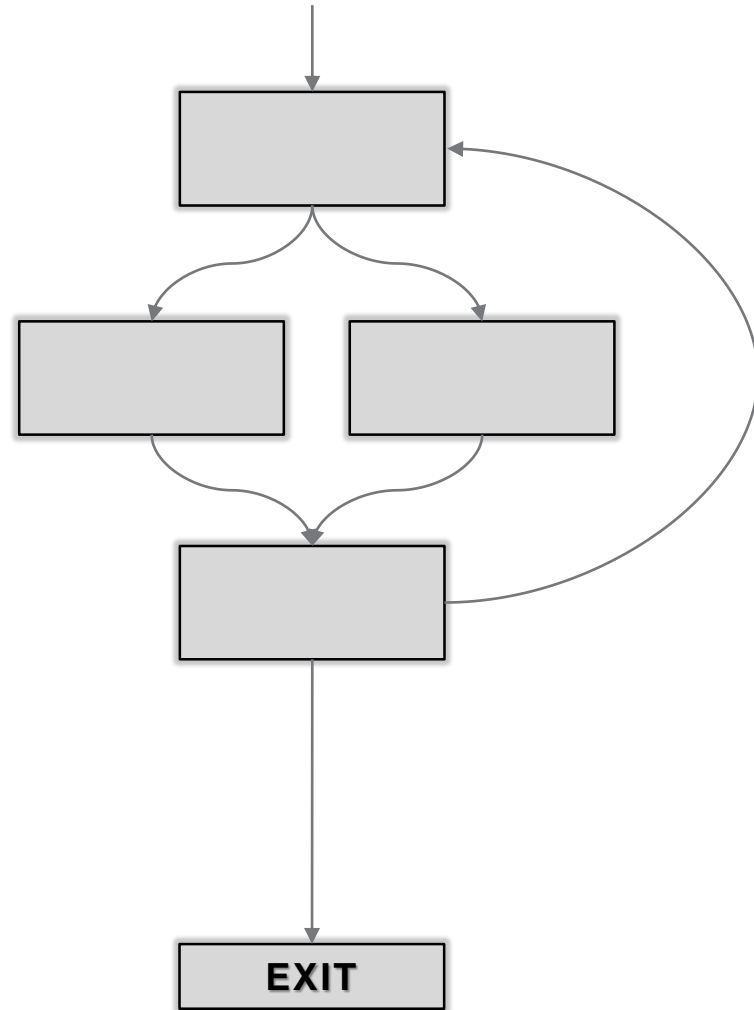
- + More realistic fault model
- + Source code not required
- Lack of semantic information
- How to find available registers

Maximize the benefits of higher level representation, static analysis, reduced overhead while minimizing interference with compiler optimizations

-
- Introduction & context
 - Embedded systems
 - Physical attacks and countermeasures
 - **Loop hardening scheme**
 - Principle & main algorithm
 - Implementation in LLVM
 - Experimental results
 - Conclusion

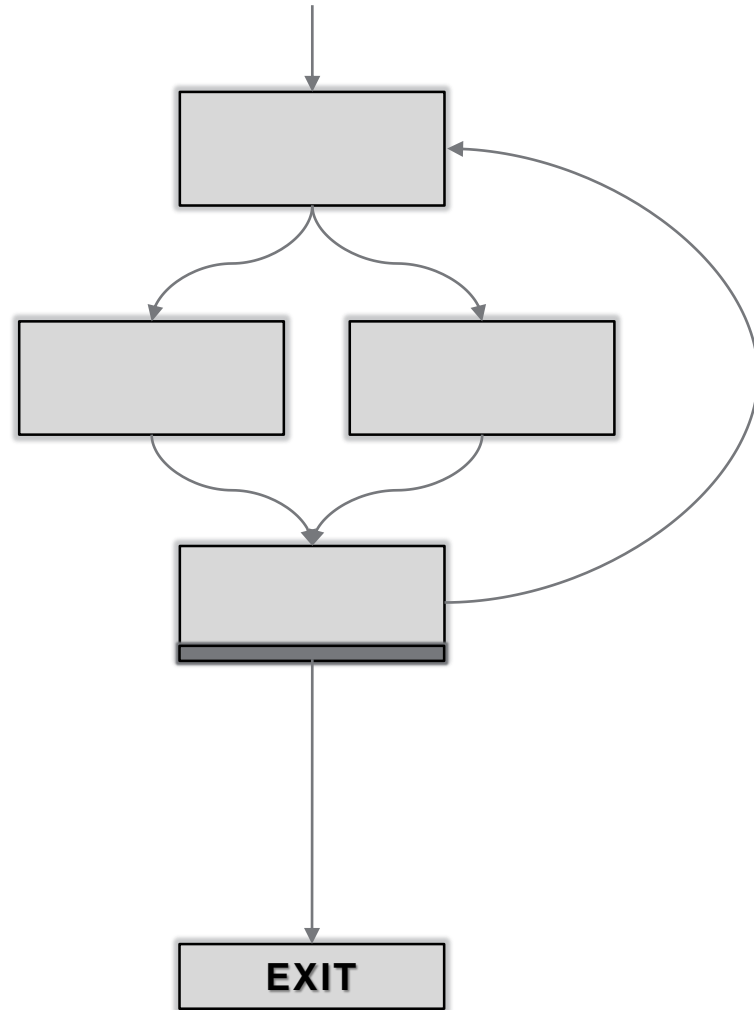
Loop hardening: principle

- **Security property**
Ensure the iteration count and proper loop exit or detect an attack
- **Principle**
Duplicate instructions involved in the computation of exit conditions and add checking blocks
- **Originality**
Minimizes overhead by duplicating only necessary instructions



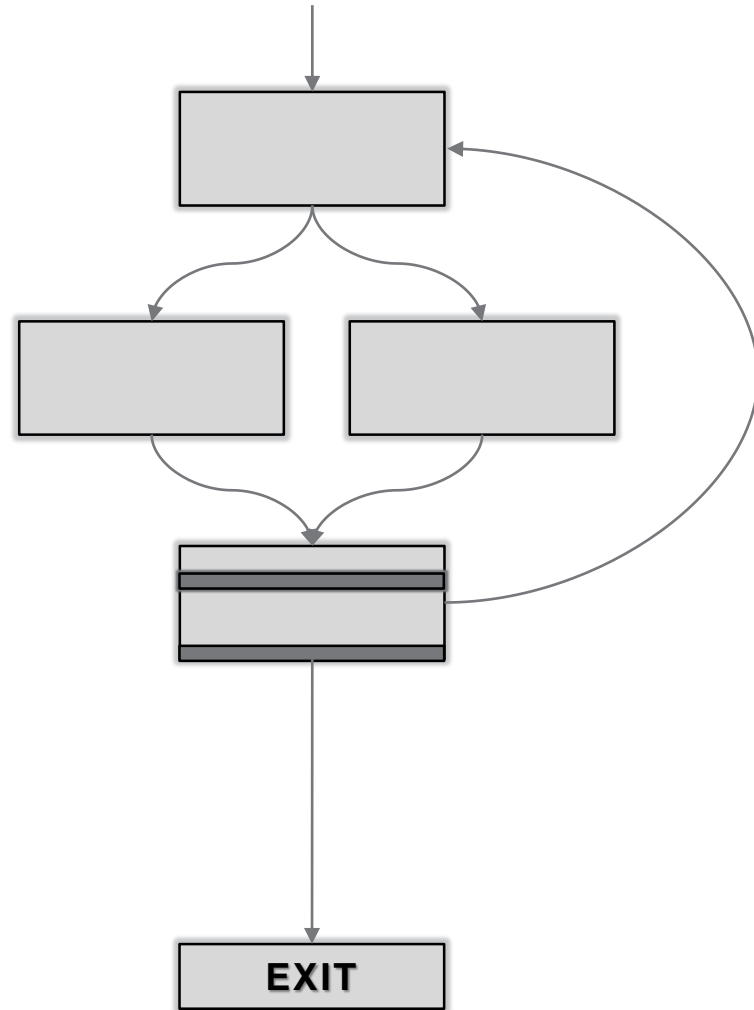
Loop hardening: principle

- Find exit condition



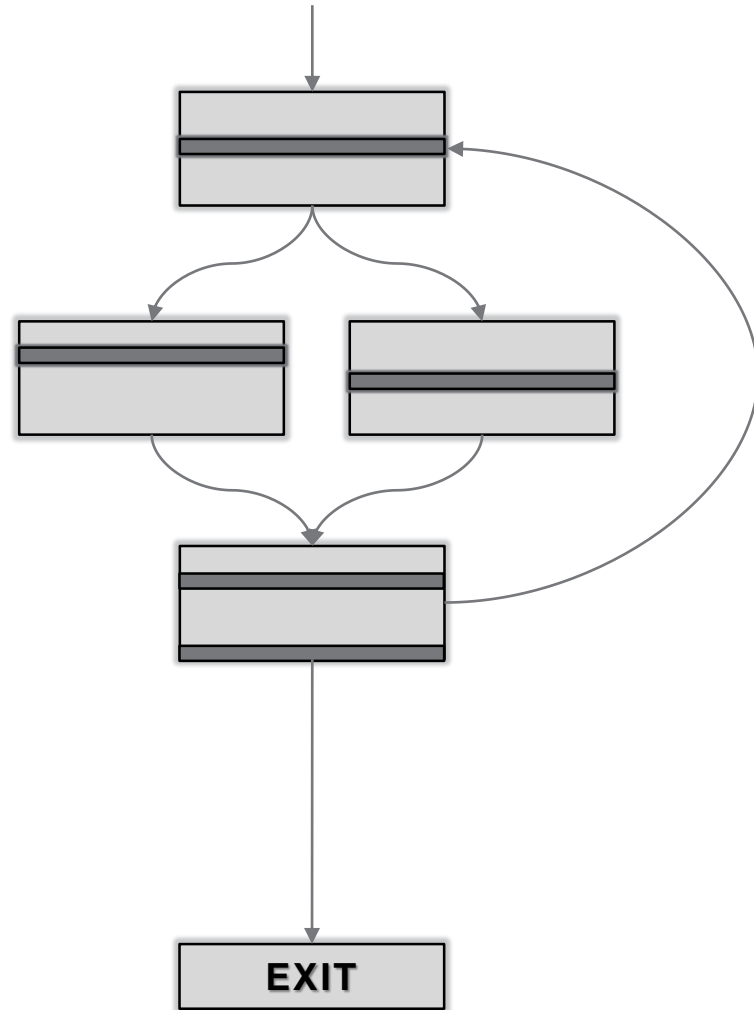
Loop hardening: principle

- Find exit condition
- Backward analysis to find instructions involved in its computation



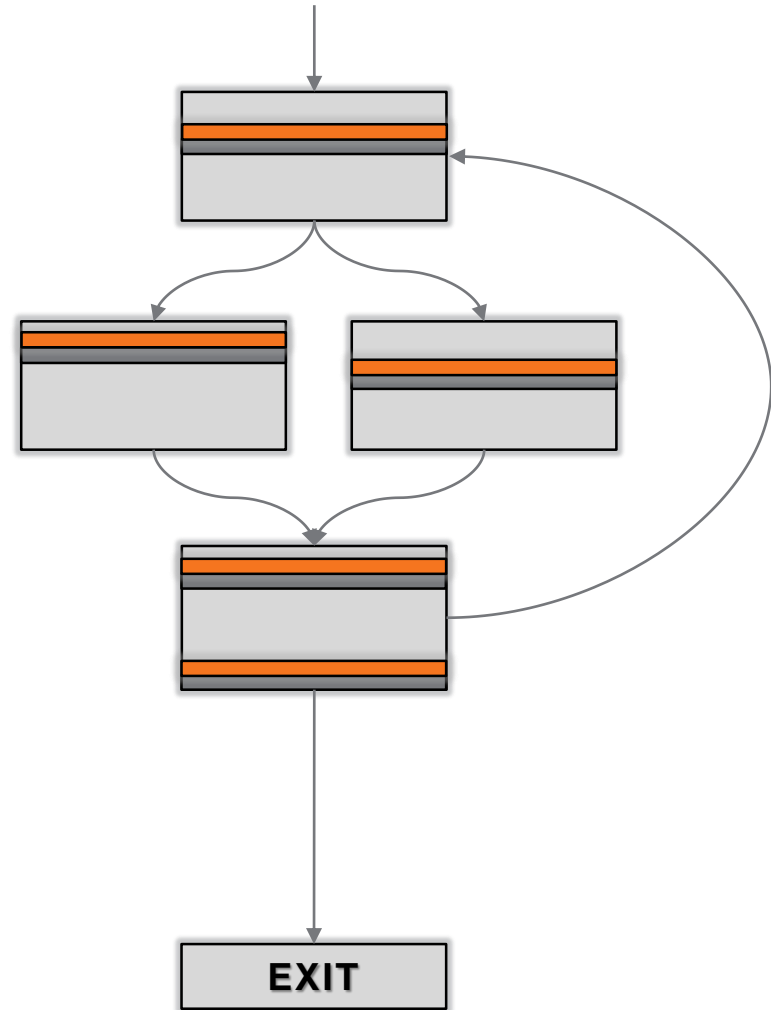
Loop hardening: principle

- Find exit condition
- Backward analysis to find instructions involved in its computation



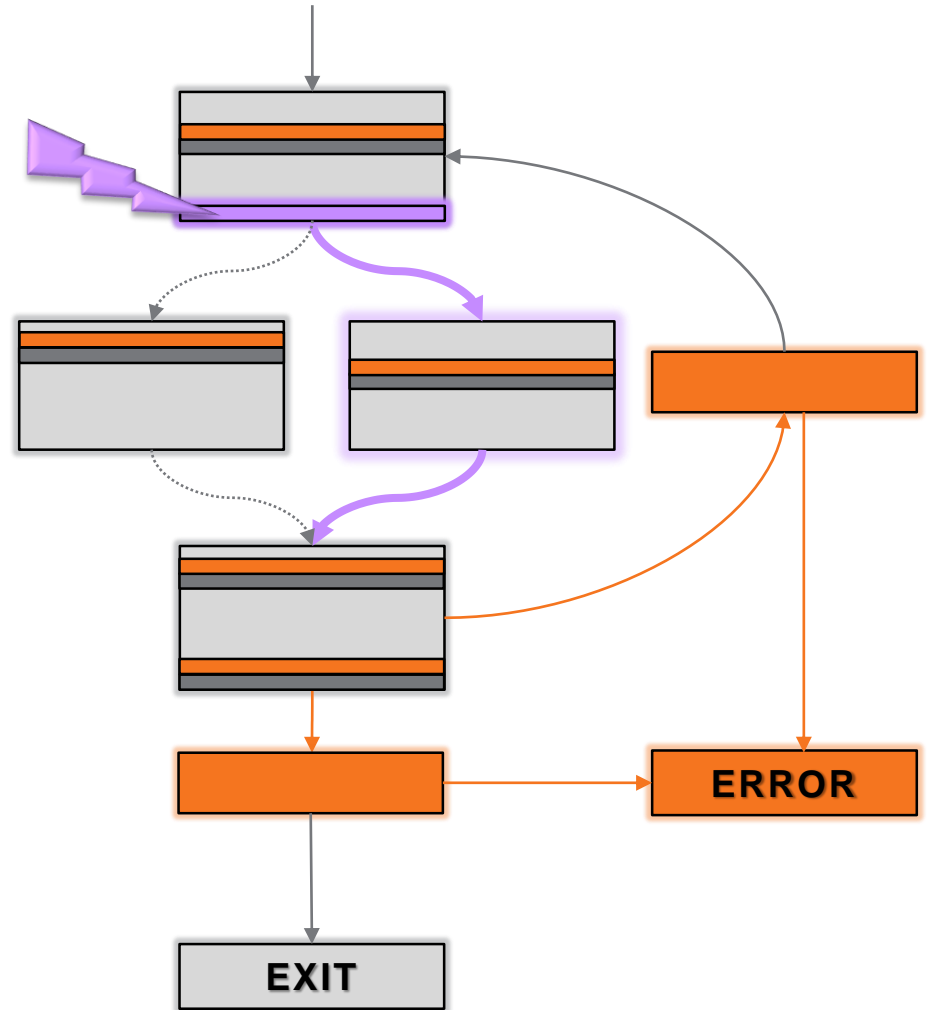
Loop hardening: principle

- Find exit condition
- Backward analysis to find instructions involved in its computation
- Duplicate selected instructions



Loop hardening: principle

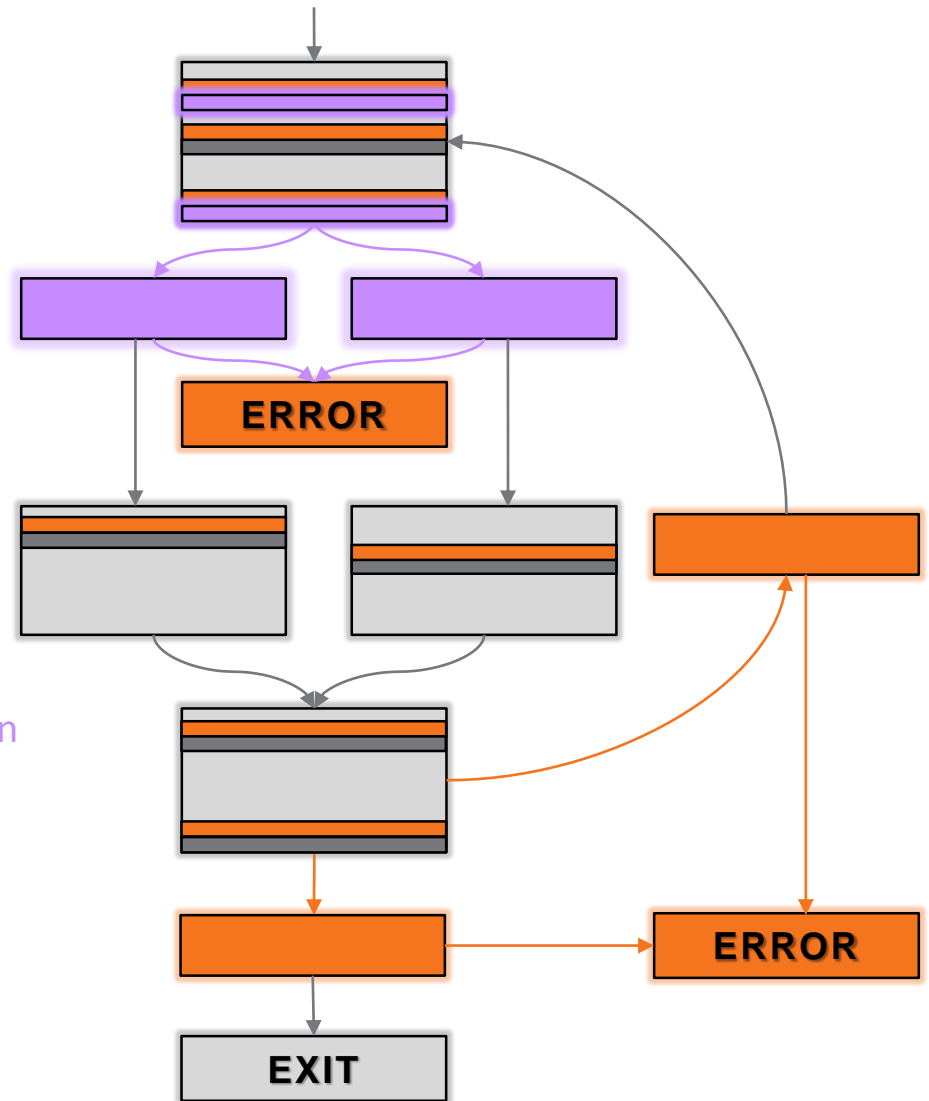
- A fault targeting **an internal branch** can compromise both data flows



Loop hardening: main algorithm

- For each loop and each exiting,

- Find exit condition
- Backward analysis to find instructions involved in its computation or in a branch condition influencing it
- Duplicate selected instructions
- Create blocks:
 - Checking the exit condition
 - Checking internal branch condition
 - Error handler



Integration in compilation flow



- Middle-end level is more portable and easily retargetable
- Algorithm designed for an SSA-based Intermediate Representation (IR)

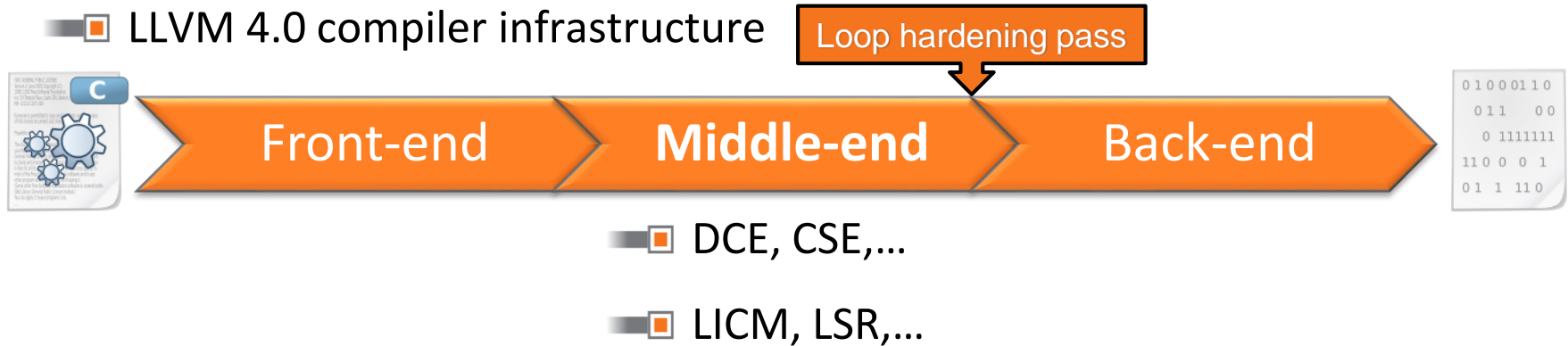
Integration in compilation flow

- LLVM 4.0 compiler infrastructure



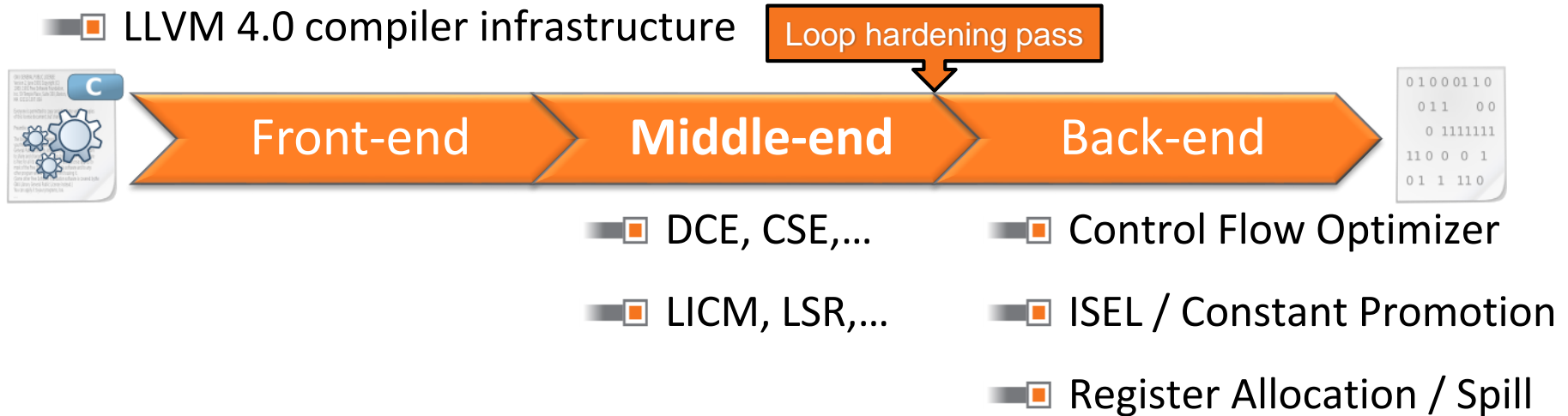
- DCE, CSE, ...
 - LICM, LSR, ...
- Middle-end level is more portable and easily retargetable
 - Algorithm designed for an SSA-based Intermediate Representation (IR)
 - Placement to avoid interactions with the compiler

Integration in compilation flow



- Middle-end level is more portable and easily retargetable
- Algorithm designed for an SSA-based Intermediate Representation (IR)
- Placement to avoid interactions with the compiler

Interactions with backend



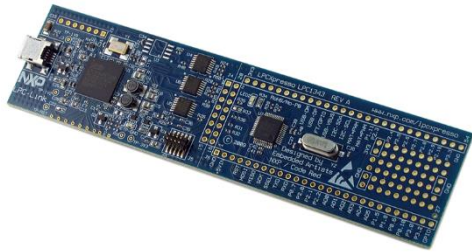
- Back-end interactions?
- Dealing with interfering passes
 - **Adapt**: Instruction Selection, Register Allocation
 - Or **Deactivate**: Control Flow Optimizer

-
- Introduction & context
 - Embedded systems
 - Physical attacks and countermeasures
 - Loop hardening scheme
 - Principle & main algorithm
 - Implementation in LLVM
 - **Experimental results**
 - Conclusion

Experimental setup

- Two ARM target boards

- NXP LPCXpresso 1343 with ARM **Cortex M3** (ARMv7m/Thumb2)



- Raspberry Pi3 with ARM **Cortex A53** (ARMv7a)



Experimental setup

- Two ARM target boards

- NXP LPCXpresso 1343 with ARM **Cortex M3** (ARMv7m/Thumb2)



- Raspberry Pi3 with ARM **Cortex A53** (ARMv7a)



- 18 Benchmarks selected covering different kind of target code

- SPEC CPU2006 INT Benchmark (*general purpose reference*)
- MiBench pgp & blowfish (*security oriented reference*)
- 3 Homemade cryptographic codes (*aes, ecc, sha*)
- gzip, gmp lib, openssl, sqlite (*large and complex loops*)

Experimental evaluation

■ Objectives

- Functional correctness
- Automatic loop coverage
- Security analysis
- Overhead evaluation

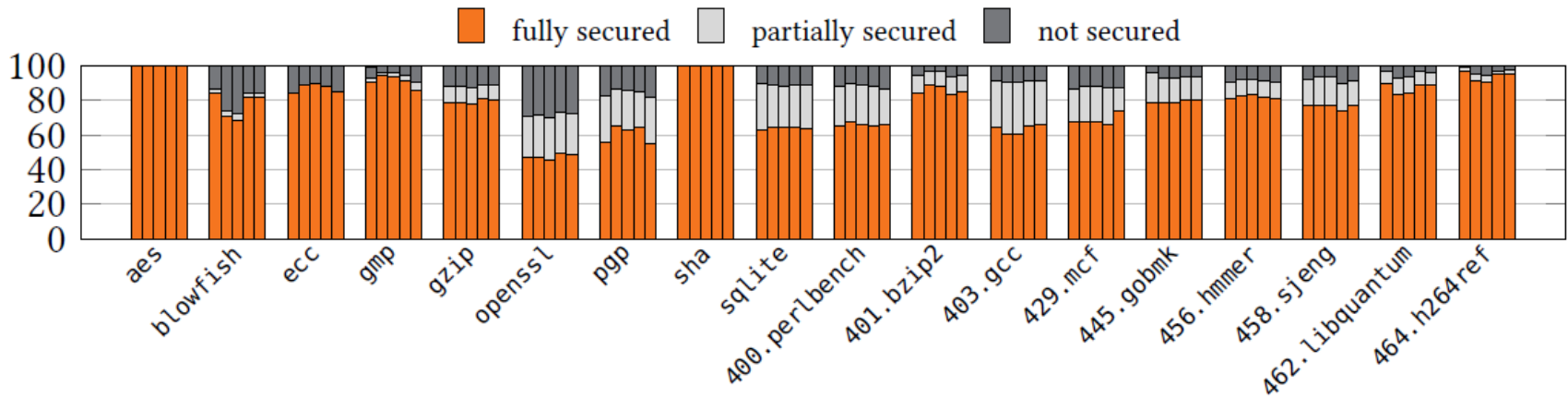
■ Methodology for functional correctness from benchmark multiplicity

- Self-testing libraries & benchmarks
- CIPHERING/decIPHERING procedure for cryptographic code

■ Compilation at various optimization levels

Automatic loop hardening

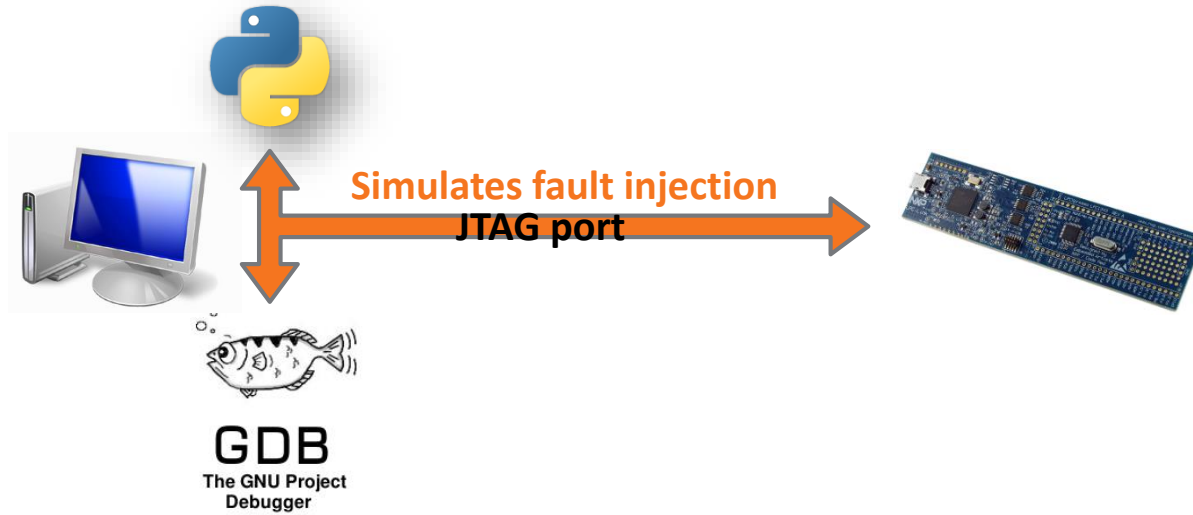
Loop coverage



- Results obtained by default option hardening all loops (sensitive or not)
- Compiled with 5 standard optimization levels (-O1, -O2, -O3, -Os, -Oz)
- Fail if slice contains unduplicable instruction (ex. call, volatile memory access...)
- Warning generated in this case

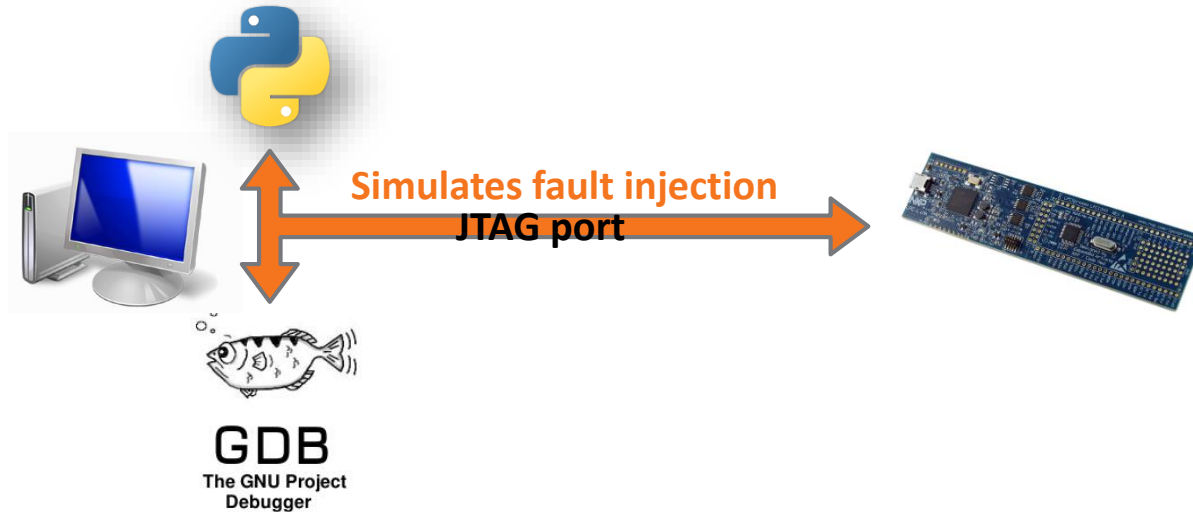
Security analysis

- Security evaluation from simulation with **python** based **gdb** scripting



Security analysis

- Security evaluation from simulation with **python** based **gdb** scripting



- ARM Loop Hardening scheme's robustness
 - Simulation of *instruction skip* and *register corruption*
 - Original code: $\approx 600\,000$ injections, $\approx 30\,000$ effective faults
 - Secure code: $\approx 900\,000$ injections, $\approx 85\,500$ effective faults, $\approx 85\,000$ detected, i.e. $\approx 99.4\%$
 - Up to **100%** with back-end modifications implemented

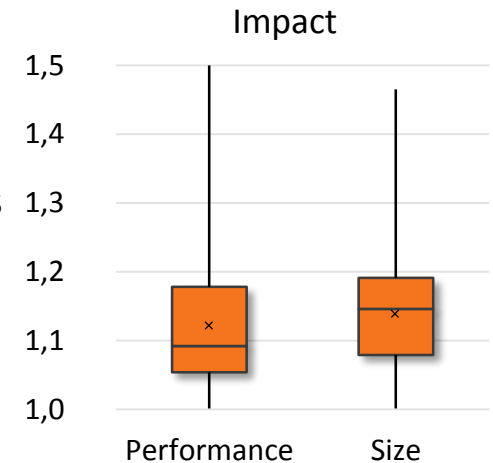
Overhead introduced

Performance overhead

- From **0% to 50%** depending on execution time spent in loops
- Average **12.5%**, median **9,2%**

Code size overhead

- From **1% to 43%** depending on number and complexity of loops
- Average **14%**, median **14,5%**
- High variance due to different benchmarks/loops



Overhead introduced

Performance overhead

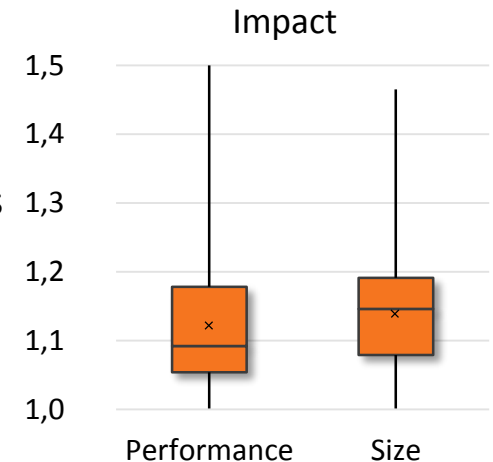
- From **0% to 50%** depending on execution time spent in loops
- Average **12.5%**, median **9,2%**

Code size overhead

- From **1% to 43%** depending on number and complexity of loops
- Average **14%**, median **14,5%**
- High variance due to different benchmarks/loops

Overhead limited if loop is the sensitive part

- Compared to fully duplication schemes ($\geq x2$)
- Focus on sensitive parts only



-
- Introduction & context
 - Embedded systems
 - Physical attacks and countermeasures
 - Loop hardening scheme
 - Principle & main algorithm
 - Implementation in LLVM
 - Experimental results
 - **Conclusion**

Conclusion & Perspectives

- Loop hardening countermeasure at compilation time
 - Ensure loop iteration count & proper exit taken
 - 99% of simulated attacks detected with low overhead & fine tuned security

Conclusion & Perspectives

- Loop hardening countermeasure at compilation time
 - Ensure loop iteration count & proper exit taken
 - 99% of simulated attacks detected with low overhead & fine tuned security
 - Answer to need for automation vs. manual application today
 - Compilation warning to user when not applicable
 - Benefits from compiler optimizations...

Conclusion & Perspectives

- Loop hardening countermeasure at compilation time
 - Ensure loop iteration count & proper exit taken
 - 99% of simulated attacks detected with low overhead & fine tuned security
 - Answer to need for automation vs. manual application today
 - Compilation warning to user when not applicable
 - Benefits from compiler optimizations...
- ...but requires to patch the compiler
 - To guarantee security properties preservation up to the end
 - Highlights incompatibility between compiler optimizations and security properties

Conclusion & Perspectives

- Loop hardening countermeasure at compilation time
 - Ensure loop iteration count & proper exit taken
 - 99% of simulated attacks detected with low overhead & fine tuned security
 - Answer to need for automation vs. manual application today
 - Compilation warning to user when not applicable
 - Benefits from compiler optimizations...
- ...but requires to patch the compiler
 - To guarantee security properties preservation up to the end
 - Highlights incompatibility between compiler optimizations and security properties
- Open questions
 - Combination of protections?
 - How to automate multiple protections together?

INVIA



Invia
INVENTEURS DU MONDE NUMÉRIQUE



Thanks!
Questions?

Dividing the Threshold: Multi-Probe Localized EM Analysis on Threshold Implementations

Robert Specht, Vincent Immler, Florian Unterstein, Johann Heyszl, Georg Sigl

Technical University of Munich

Department for Electrical and Computer Engineering

Fraunhofer Institute AISEC

Applied and Integrated Security

May 2018



Uhrenturm der TUM

Goals of our work

Questions to be answered:

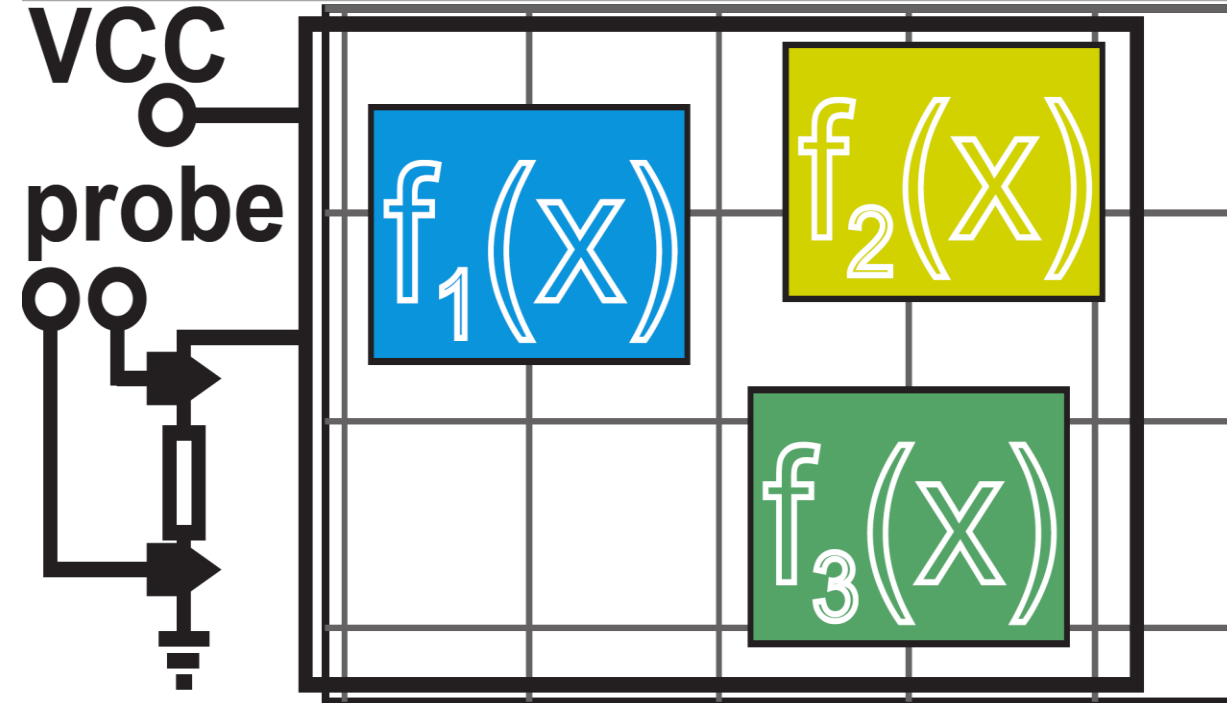
- Are Threshold Implementations (TI) a strong protection against SCA?
- What is better:
 - Global power measurements or
 - Local electromagnetic emission (EM) measurements?
 - Single EM probe
 - Multiple EM probes

Approach

- Implementation of a TI protected S-Box on an FPGA
 - 3 share TI according to *
- Side channel analysis of masked implementation
 - Moments correlating DPA with power
 - LDA based template attack with power
 - LDA based template attack with 1 EM probe
 - LDA based template attack with 3 EM probe
- Compare those side channel analyses

* T. De Cnudde, O. Reparaz, B. Bilgin, S. Nikova, V. Nikov, and V. Rijmen, “Masking AES with $d+1$ shares in hardware,” in International Conference on Cryptographic Hardware and Embedded Systems. Springer

The principle of TI implementation

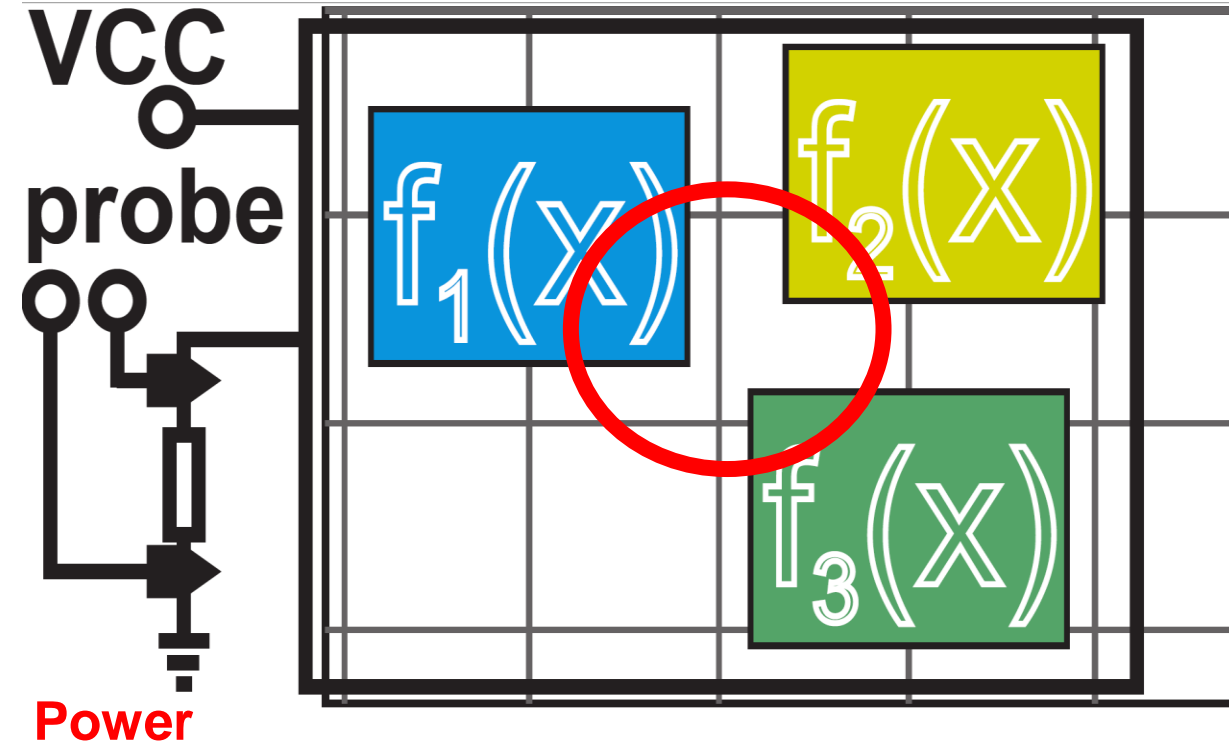


$$f_1: x = \text{Sbox_in} \oplus m_1 \oplus m_2$$

$$f_2: x = m_1$$

$$f_3: x = m_2$$

Single Measurement



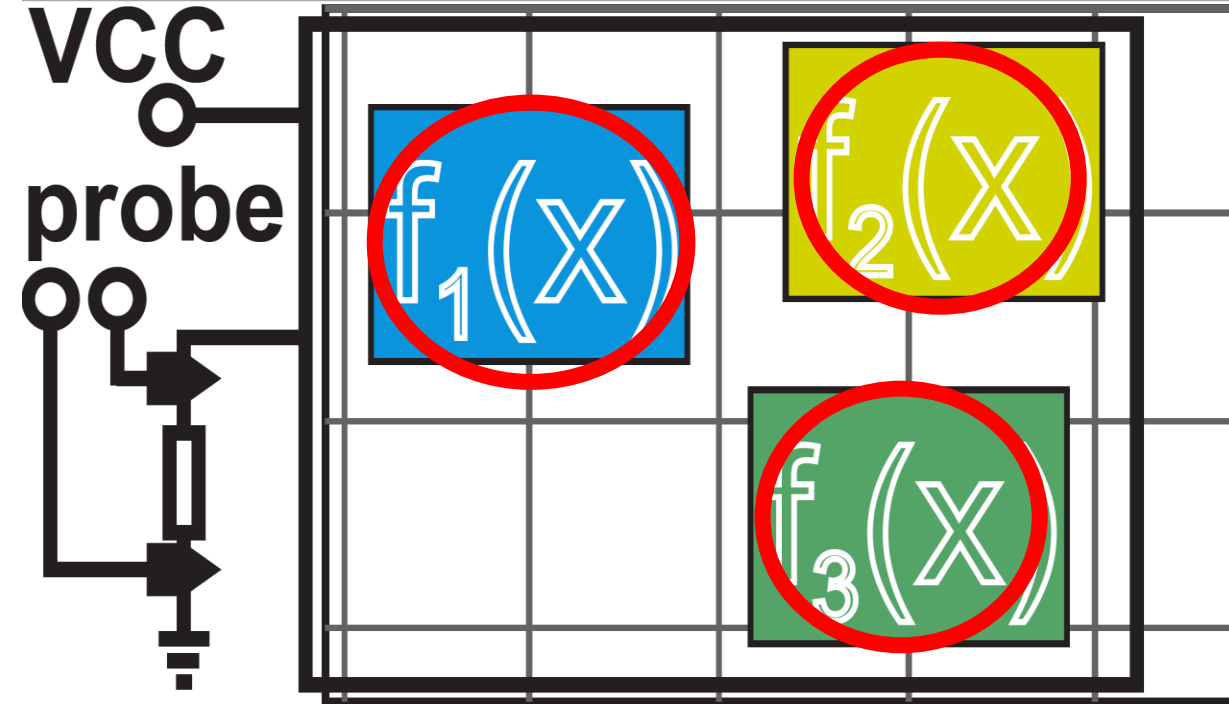
$$f_1: x = \text{Sbox_in} \oplus m_1 \oplus m_2$$

$$f_2: x = m_1$$

$$f_3: x = m_2$$

1 EM Probe

Measurement with 3 Probes



$$f_1: x = \text{Sbox_in} \oplus m_1 \oplus m_2$$

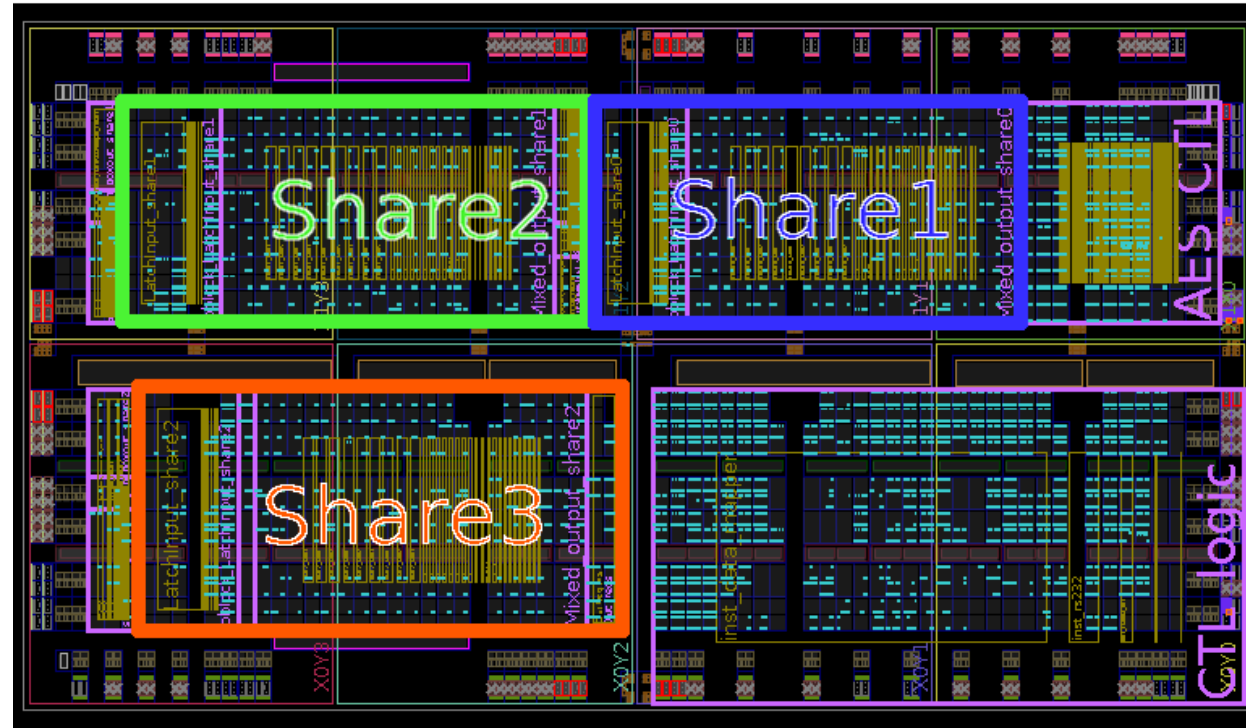
$$f_2: x = m_1$$

$$f_3: x = m_2$$

3 EM Probes

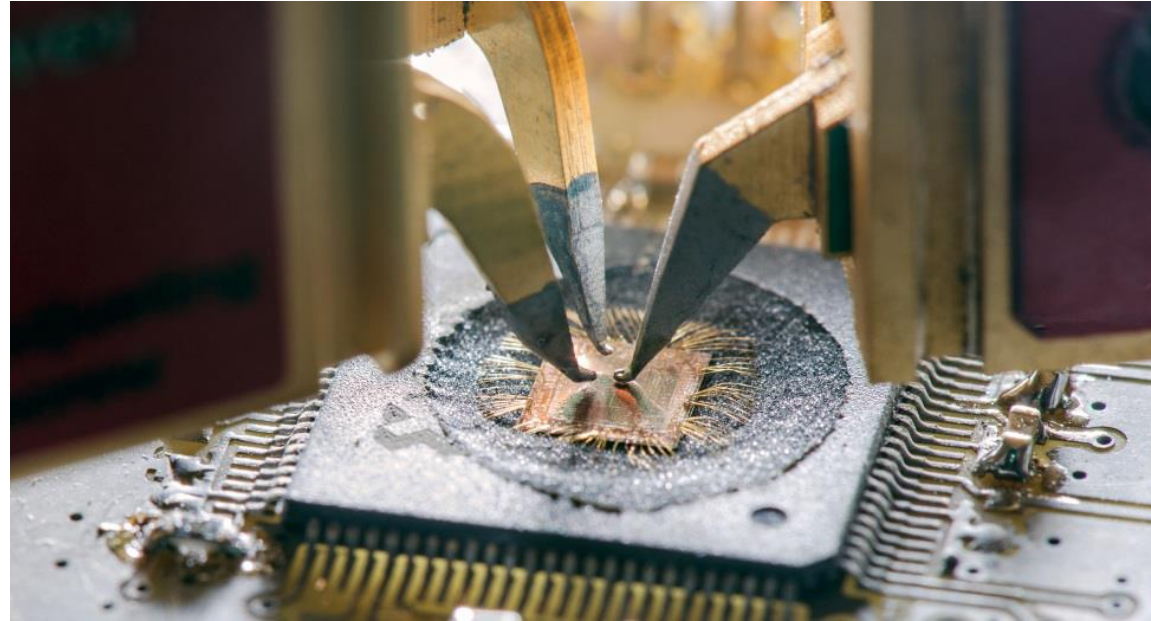
FPGA Layout

- Shares must be separated to avoid coupling
- Coupling violates the idea of independent shares
- This could result in first order weaknesses



Measurement Setup

- De-capped Spartan 6 (45nm Technology)
- Clock frequency: 8MHz
- DUT mounted on XY-table for probe positioning
- 3 Langer near field probes
 - 2x HH 150 μ m
 - 1x HH 100 μ m



Attack Steps

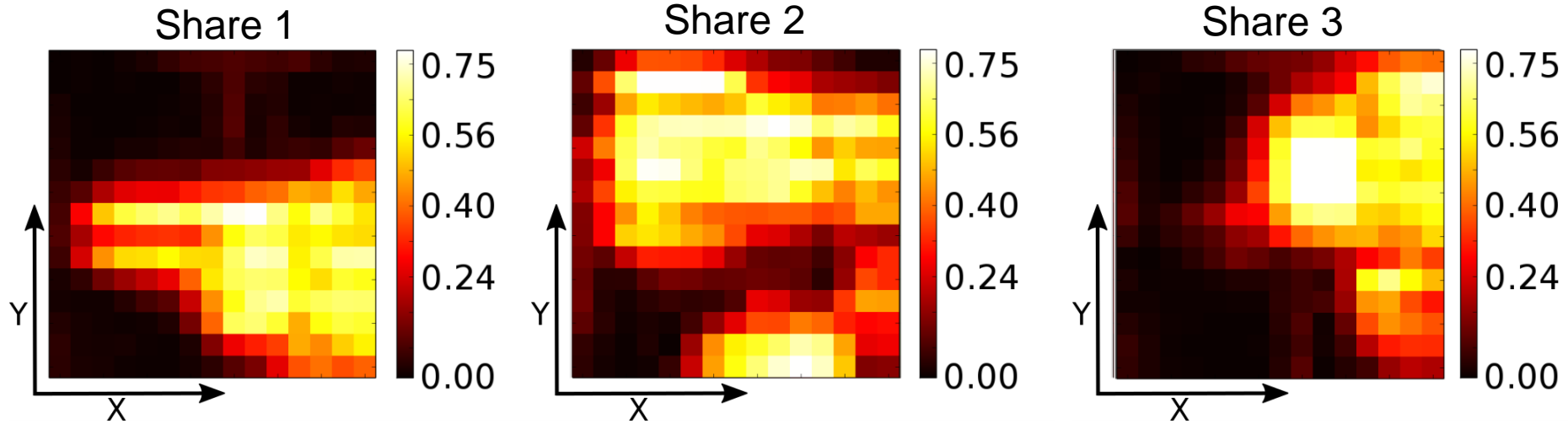
Steps:

1. Determine Location of Interest
2. Determine (Timing) Points of Interest
3. Profiling using templates and prepare LDA
4. Attack execution using LDA

Assumptions:

- Profiling Phase: Access to plaintext, key, and all masks
- Attack Phase: Access to plaintext

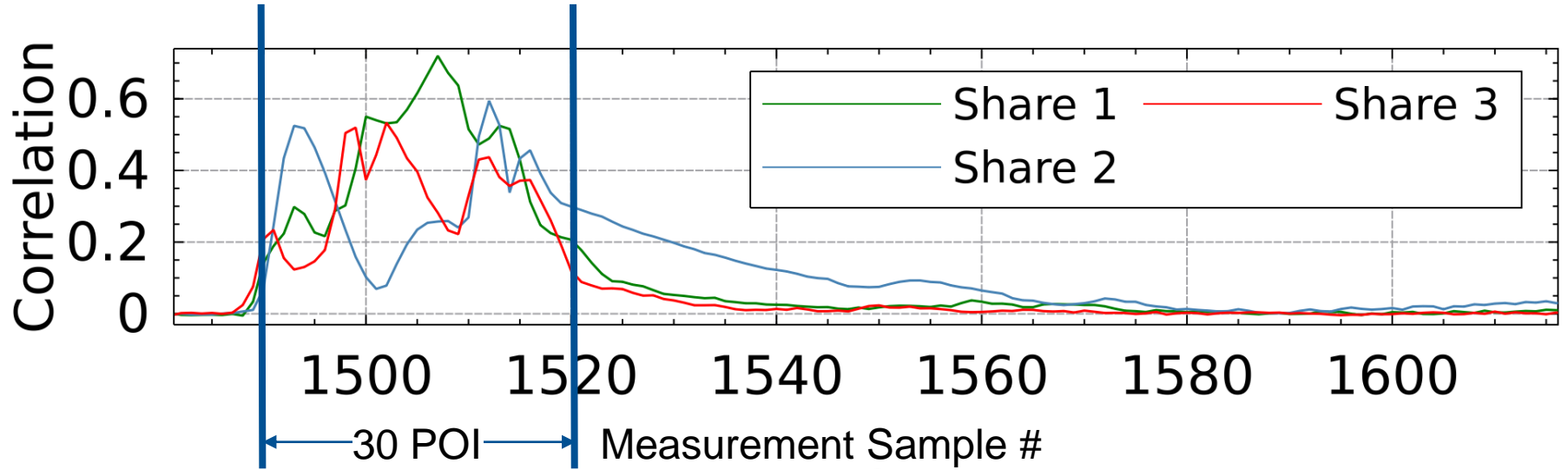
Step 1: Location of Interest



- Correlation based leakage* test with known values for m_1 , m_2 , $S_{\text{box_in}} \oplus m_1 \oplus m_2$
- Shares are spatially separable

* F. Durvaux and F.-X. Standaert, From Improved Leakage Detection to the Detection of Points of Interests in Leakage Traces. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 240–262.

Step 2: Timing Points of Interest



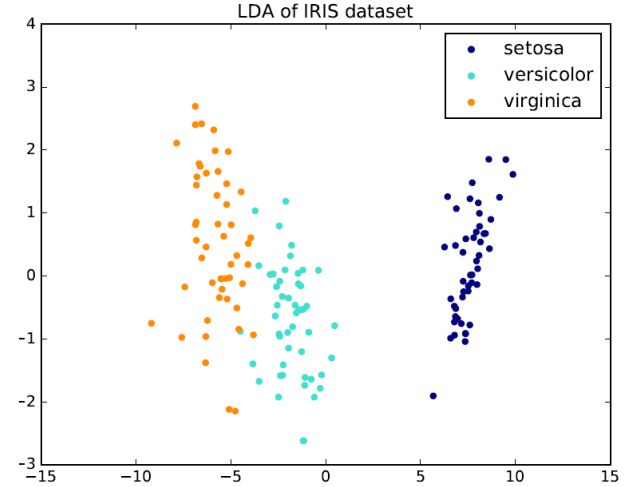
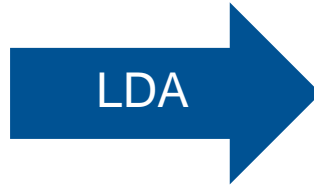
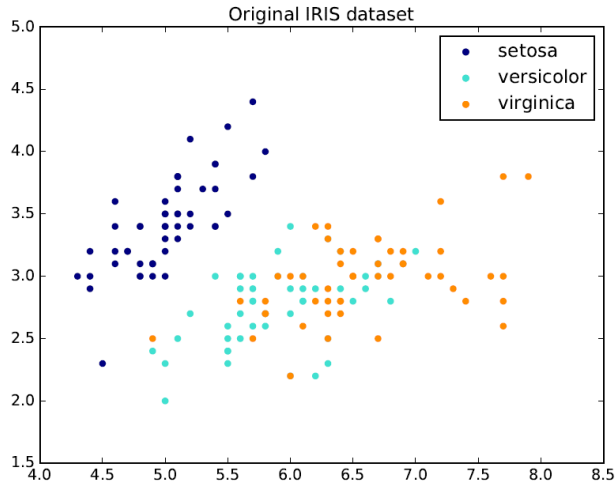
- 30 timing POI contain main leakage
- These 30 POI are used for the next step

Step 3: Profiling using templates and LDA preparation

- Profiling with 500 k traces
- Known plaintext, key, and masks
- 30 input dimensions and 15 output dimensions for LDA

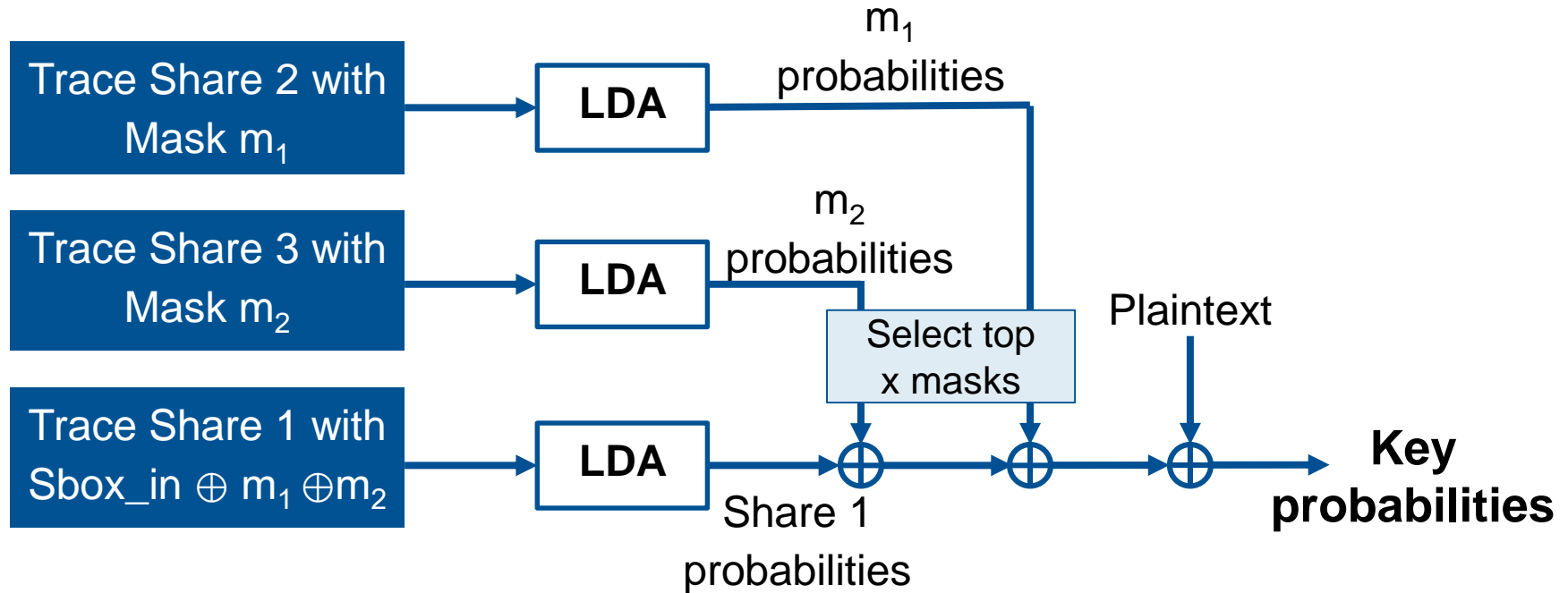
- Result:
 - Gaussian templates for all 3 intermediate values
 - LDA transformation matrices for 3 shares
 - Goal: separate all 256 possible input byte values of each share

Side Note: LDA



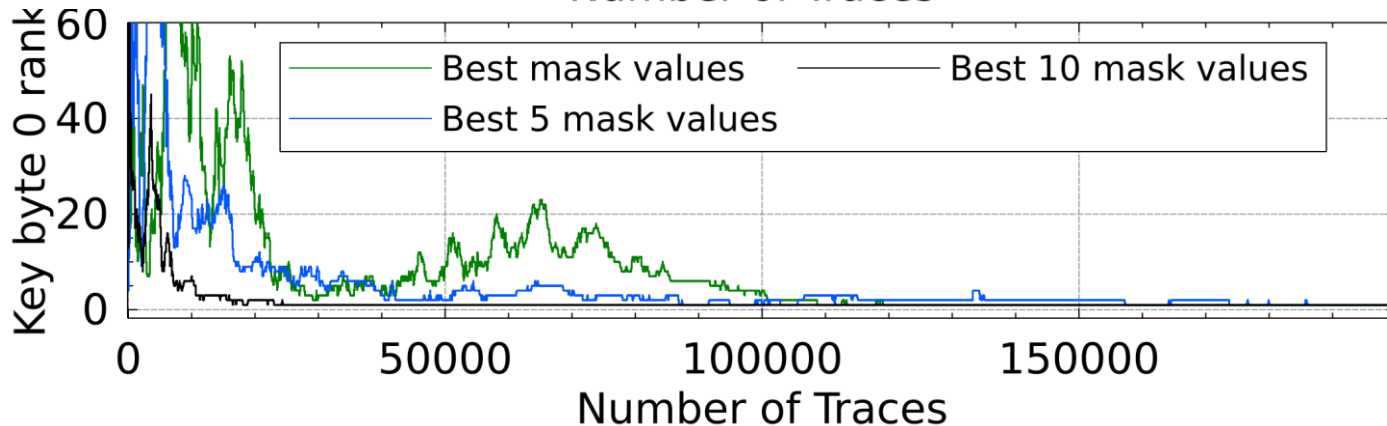
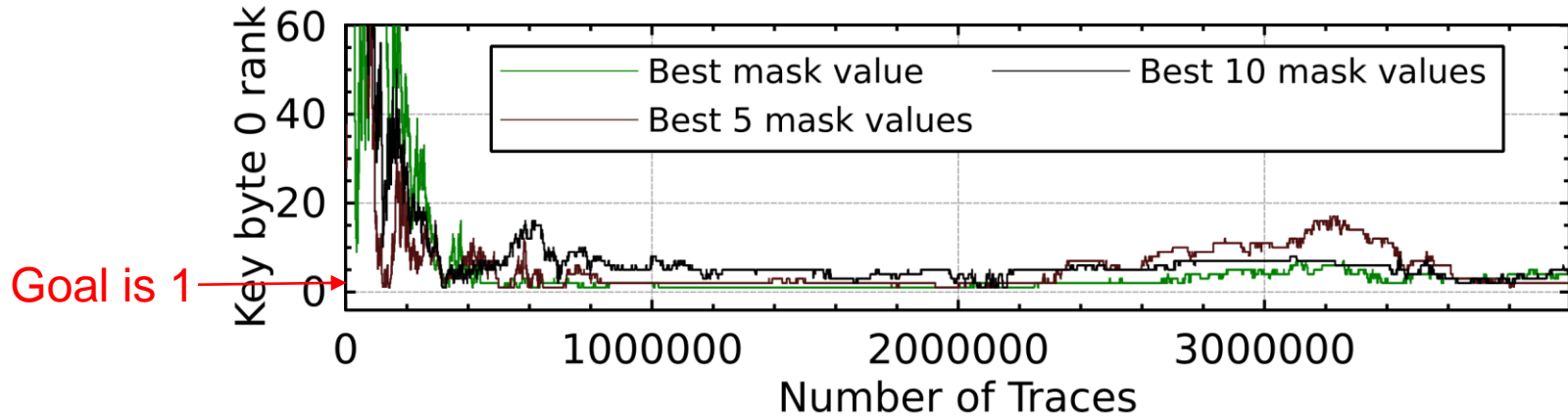
- Goal: Project to subspace, where classes are better separable
- Calculate:
 - Inter-class variance
 - Intra-class variance
- Maximize interclass variance and minimize intraclass variance

Step 4: Attack Execution

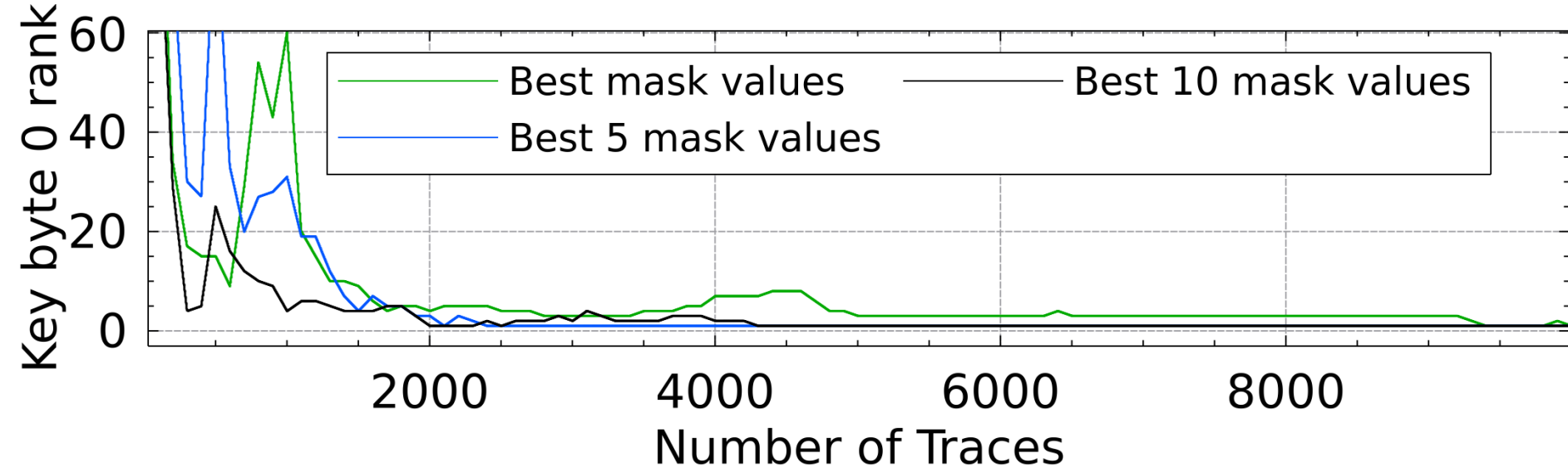


- For each trace we get a key probability list weighted with mask probabilities
- The key probabilities are accumulated for all traces → **key ranks**

Results: Power versus EM 3 probes



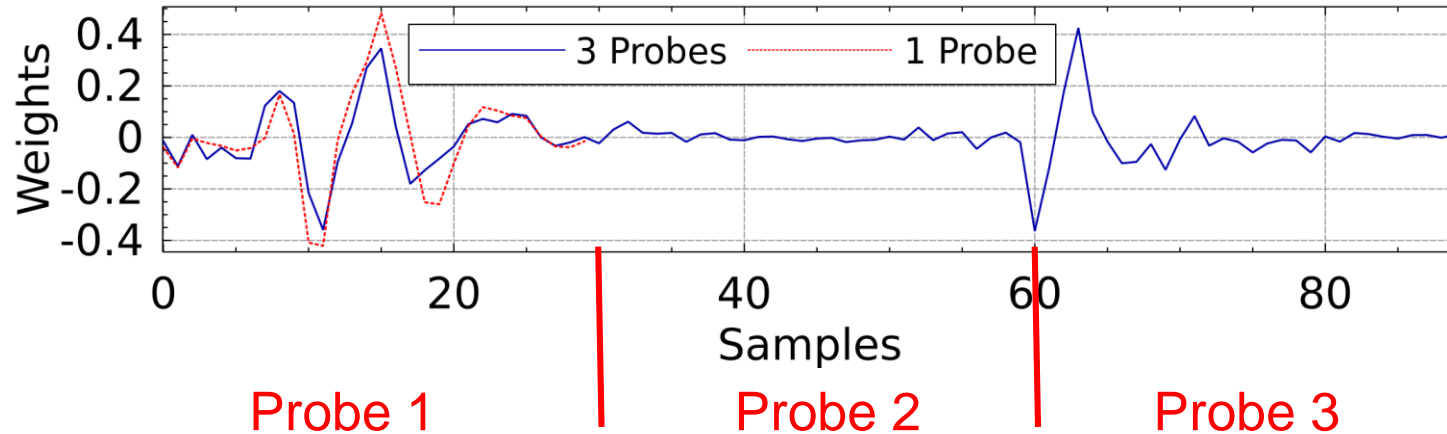
Combined EM traces



Idea:

- Combine 3 traces with 30 values (POI) by concatenation
- Perform LDA and reduce from 90 dimensions to 15
- Conclusion: Probes receive information from different shares

Details on Combining Probes



- Measurement of 2 probes is accumulated, resulting in more exploitable leakage.
- electrical noise and thermal noise, etc. can be easier eliminated by measuring the S-box-input with multiple independent observations.

Result Table

		# Best Masks	# Traces
Power Measurement	LDA	1/10	1 025 400
	MCP-DPA		600 000
EM measurement with 1 probe	LDA		Attack fails
	MCP-DPA		1 300 000
EM measurement with 3 probes	LDA separate	1	119 300
		10	24 600
	LDA combined	1	18 200
		10	4 300

MCP-DPA: A. Moradi and F.-X. Standaert, “Moments-correlating dpa,” Cryptology ePrint Archive, Report 2014/409, 2014, <http://eprint.iacr.org/2014/409>.

Conclusion

- Multiple probes can capture local EM with high SNR simultaneously
- Multiprobe EM vs power results in less traces (here factor 238)
- Multiple probes can significantly weaken the security of masking schemes

Symbolic Approach for Side-Channel Resistance Analysis of Masked Assembly Codes

PHISIC 2018

Inès Ben El Ouahma, Quentin Meunier, Karine Heydemann and
Emmanuelle Encrenaz

Sorbonne Université, LIP6, CNRS, UMR 7606, F-75005, Paris, France

May 24th, 2018, Gardanne, France



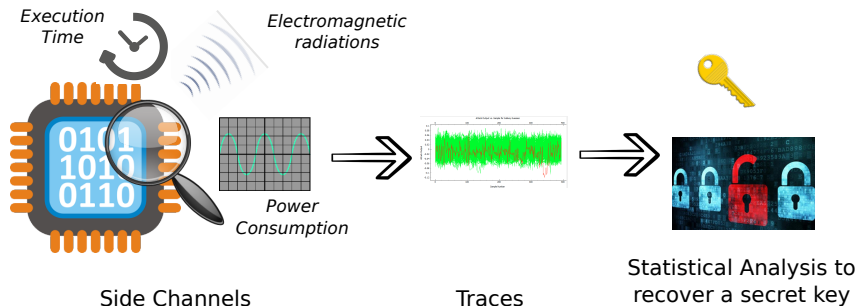
Introduction / Motivation

Symbolic Method

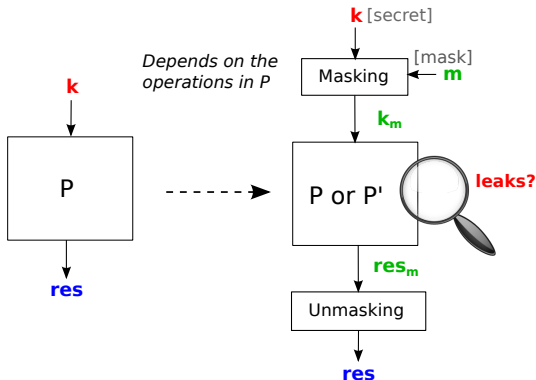
Experiments

Conclusion

Side-Channel Attacks



The Masking Countermeasure (1/2)



At order d : the observation of d intermediate computations does not reveal information on the secret k

The Masking Countermeasure (2/2)

Properties

- Splits a secret x in $d+1$ shares using random uniform independent variables called *masks*
- At the software level, added in the source code
- Depends on the algorithm, e.g. boolean masking: $x = x_1 \oplus x_2$
- In practice, masked programs are often written directly in assembly by experts

Problems

- Need to ensure that a masked program is leakage free in practice
- Compilation flow and optimizations (reordering, removal...) may affect masking effectiveness

Masked Programs Security: Existing Formal Verifications Tools and Methods

- [Bayrak13] SAT verification of *sensitivity*: an operation on a secret must involve a random variable which is not a *don't care* variable (i.e. it affects the result)
 - ✓ Low level: LLVM programs
 - ✗ Security property not sufficient
- [Eldib14] SMT verification of *perfect masking*, i.e statistical independency of intermediate computations from secrets
 - ✓ Strong security property
 - ✗ C level & Bit-blasted programs (could be applied to low level)
 - ✗ Lack of scalability (combinatorial blow-up of the enumeration)
- [Barthe15] *t-non-interference*: the joint probability distribution of any t intermediate expression is independent from secrets
 - ✓ Strong security property
 - Good scalability?
 - ✗ Cannot conclude for some cases

Our Goal

To verify side channel resistance:

- Of first order **masked** programs
- At **assembly** level
- In the **value-based model**: instruction result leaks
- Considering that: leakage-free instruction \iff result is **statistically independent** from secrets
- With a **symbolic approach** that infers the distribution type of instruction expressions

Plan

Introduction / Motivation

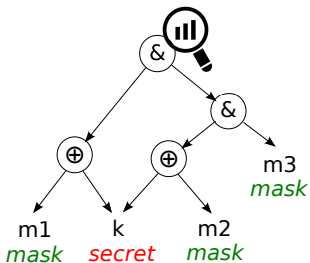
Symbolic Method

Experiments

Conclusion

Verification Scheme

```
# r0 ← k; r1 ← m1; r2 ← m2; r3 ← m3
1 eor r4, r0, r1 # k ⊕ m1
2 eor r5, r0, r2 # k ⊕ m2
3 and r5, r5, r3 # (k ⊕ m2) & m3
4 and r5, r5, r4 # (k ⊕ m1) & ((k ⊕ m2) & m3)
```



Data dependency graph of the last instruction

Is the root distribution statistically independent from k ?

- ▶ Inputs tagged with a distribution type
- ▶ Bottom-up combination of distribution types using defined inference rules

Symbolic Approach

4 distribution types for variables and expressions:

- Random Uniform Distribution (**RUD**)
- Unknown Distribution (**UKD**)
- Constant (**CST**)
- (Statistically) Independent from Secrets Distribution (**ISD**): not necessarily uniform but identical for all values of the secrets.

k: secret

m_1, m_2 : masks

$e = (k \oplus m_1) \& m_2$

$e' = (k \oplus m_1) \& m_1$

k	m_1	m_2	e
0	0	0	0
	0	1	0
	1	0	0
	1	1	1
1	0	0	0
	0	1	1
	1	0	0
	1	1	0

$$\left. \begin{array}{l} P(e=0) = \frac{3}{4} \\ P(e=1) = \frac{1}{4} \end{array} \right\} \quad \left. \begin{array}{l} P(e=0) = \frac{3}{4} \\ P(e=1) = \frac{1}{4} \end{array} \right\}$$

e'
0
0
1
1
0
0
0
0

$$\left. \begin{array}{l} P(e'=0) = \frac{1}{2} \\ P(e'=1) = \frac{1}{2} \end{array} \right\} \quad \left. \begin{array}{l} P(e'=0) = 1 \\ P(e'=1) = 0 \end{array} \right\}$$

Independence Notions

Which distribution types assert that an expression is statistically independent from secrets?

Dependence between an expression e and a variable v :

- *structural* $\implies v$ appears in e
- *statistical* \implies the distribution of the result of e depends on v

\implies Need to keep track of structural dependencies: $(k \oplus m) \& m$

Safe tags:

- $e \sim \text{RUD}$
- $e \sim \text{ISD}$
- $e \sim \text{UKD}$ with no structural dependency on any secret

Unsafe tag:

- $e \sim \text{UKD}\{\text{dep}\}$ with structural dependency on some secret variable: $\text{dep} \cap S \neq \emptyset$

Dominant Masks

Aim: to find a mask that randomizes the whole expression

Dom Rule

- expression $e = e' \oplus m$ or $e = e' + m \bmod 2^n$
- $m \sim \text{RUD}\{m\}$
- $m \notin \text{dep}(e')$

$\implies e \sim \text{RUD}$ and m is a **dominant mask** of e .

2 sets of dominant masks:

- $\text{dom}_{\oplus}(e)$ the set of xor dominant masks of e
- $\text{dom}_{+}(e)$ the set of additive dominant masks of e

Examples:

- $\text{dom}_{\oplus}((k + m1) \oplus (k \oplus m1 \oplus m2)) = m2$
- $\text{dom}_{+}((k + m1) \oplus 0) = \text{dom}_{+}(k + m1) = m1$

Other Inference Rules

By distribution types:

- Set of rules for \oplus , $+$ mod 2^n
- Set of rules for AND and OR

Disjoint rule for binary operators

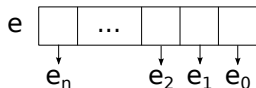
- $u \sim \text{ISD}\{\text{dep0}\}$ and $v \sim \text{ISD}\{\text{dep1}\}$
- No masks in common: $\text{dep0} \cap \text{dep1} \cap M = \emptyset$

$\implies (u \text{ op } v) \sim \text{ISD}\{\text{dep0} \cup \text{dep1}\}$ for every binary operation op

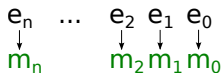
▷ More details in a PROOFS'17 article

Bit Level Analysis

When no conclusion is possible at word level:
 \implies split the expression into several expressions at bit level

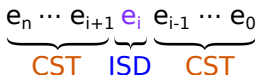


▷ case 1:



$e_i \sim$ RUD and different dominant masks for each e_i

▷ case 2:



Concatenation of an ISD bit with CST bits

▷ case 3:



Duplicated ISD bit and concatenation with CST bits

Example from mix columns in AES:

$$e = ((LSR(mt1 \oplus mp \oplus sbx5, 7) \oplus LSR(mt2 \oplus mp \oplus sbx10, 7)) + ((LSR(mt1 \oplus mp \oplus sbx5, 7) \oplus LSR(mt2 \oplus mp \oplus sbx10, 7)) \ll 1)$$

$$b_7 = mt1_7 \oplus mp_7 \oplus sbx5_7 \oplus mt2_7 \oplus mp_7 \oplus sbx10_7$$

$$e \implies 0000\ 00b_7b_7 \implies ISD$$

Plan

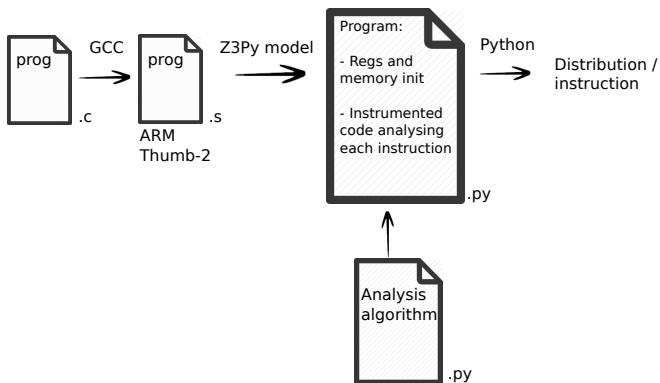
Introduction / Motivation

Symbolic Method

Experiments

Conclusion

Implementation



Benchmarks

- Compiled in O2 with gcc

Program	# ASM insts.	Size in bits	# Masks	# secrets	Secure in literature
C Boolean programs					
P6 [Eldib14]	8	1	3	3	×
Masked Khi [Eldib14]	8	1	2	3	✓
Algorithms for switching between boolean and arithmetic maskings					
Goubin Conversion [Goubin01]	8	4	2	1	✓
Coron Conversion [Coron15]	37	4	3	1	✓
Cryptographic algorithms					
Masked AES 1st round [Herbst06]	422	8	6	16 + 16	✓
Simon TI 1st round [Shahverdi17]	15	32	5	3 + 2	✓

Analysis Results

Comparison with *Enum-C*: C program which enumerates the combinations of values (secrets and masks) and seeks 2 values of a secret with different distributions

Program	# RUD	# ISD	# CST	# UKD	Time	Enum-C # leaks
P6	6	2	0	0	<1s	0
Masked Khi	2	2	0	4	<1s	4
Goubin Conversion	5	0	0	3	<1s	0
Coron Conversion	14	10	0	13	2s	7
Masked AES (1st round)	238	64	120	0	22s	-
Simon TI round 1	7	4	1	3	8.5s	-

Impact of Bit-Level Analysis

Program	#RUD _w	#RUD _b	#RUD total	#ISD _w	#ISD _b	#ISD total	#CST	#UKD _w	#UKD _b	#UKD total
P6	6	0	6	2	0	2	0	0	0	0
Masked Khi	2	0	2	2	0	2	0	4	4	4
Goubin Conv.	0	0	5	0	0	0	0	3	3	3
Coron Conv.	10	4	14	6	4	10	0	21	13	13
Masked AES 1st round	222	16	238	0	64	64	120	80	0	0
Simon TI 1st round	7	0	7	0	3	3	1	7	4	4

- Coron Conv. & Simon TI: allows to conclude for 40% of UKD results

Currently Ongoing Work

- Automating the generation of the python code representing the assembly code
- Explore the transition leakage model (based on the xor between two consecutive values in a register)
- Compare the performance and results to those of [Barthe15](#)

Plan

Introduction / Motivation

Symbolic Method

Experiments

Conclusion

Conclusion

We propose a symbolic method:

- For verifying side channel robustness of 1st order masked programs at assembly level
- Using type inference of expression distributions
- Scalable and sound, but not complete

Perspectives for future work:

- Refine the set of rules / bit level analysis
- Combine with the other approaches (enumerative, NI) at bit level
- Extend to other leakage models (e.g general transition-based model) / higher masking orders
- Take into account architecture specificities in the model
- Criteria to conclude on the leakage of some UKD?

References

- [Bayrak13] Ali Galip Bayrak, Francesco Regazzoni, David Novo, Paolo Ienne. Sleuth: Automated Verification of Software Power Analysis Countermeasures. *CHES 2013: 293-310*
- [Eldib14] Hassan Eldib, Chao Wang, Patrick Schaumont. SMT-Based Verification of Software Countermeasures against Side-Channel Attacks. *TACAS 2014: 62-77*
- [Barthe15] Gilles Barthe, Sonia Belaïd, François Dupressoir, Pierre-Alain Fouque, Benjamin Grégoire, Pierre-Yves Strub. Verified Proofs of Higher-Order Masking. *EUROCRYPT (1) 2015: 457-485*
- [Goubin01] Louis Goubin. A sound method for switching between boolean and arithmetic masking. In *Cryptographic Hardware and Embedded Systems CHES 2001*, pages 3–15. Springer, 2001.
- [Coron15] Jean-Sébastien Coron, Johann Großschädl, Mehdi Tibouchi, and Praveen Kumar Vadnala. Conversion from arithmetic to boolean masking with logarithmic complexity. In *International Workshop on Fast Software Encryption*, pages 130–149. Springer, 2015.
- [Herbst06] Christoph Herbst, Elisabeth Oswald, and Stefan Mangard. An aes smart card implementation resistant to power analysis attacks. In *ACNS*, volume 3989, pages 239–252. Springer, 2006.
- [Shahverdi17] Aria Shahverdi, Mostafa Taha, and Thomas Eisenbarth. Lightweight side channel resistance. Threshold implementations of simon. *IEEE Transactions on Computers*, 66(4):661–671, 2017.

Thank you for your attention!

Backup Slide 1

Algorithm 1 Distribution inference algorithm

```
1: function INFER( $E$ )
2:   if  $e$  is a leaf then
3:     if  $e \in S$  then return UKD $\{e\}$ 
4:     else if  $e \in M$  then return RUD $\{e\}$ 
5:     else return CST
6:   else
7:      $le\{ld\} = \text{infer}(e.\text{left\_child})$ 
8:      $re\{rd\} = \text{infer}(e.\text{right\_child})$ 
9:     if  $\exists$  rule for  $(le\{ld\} \ e.\text{op} \ re\{rd\})$  that returns RUD $\{dep\}$ 
10:    then
11:      return RUD $\{dep\}$ 
12:    else if  $\exists$  rule for  $(le\{ld\} \ e.\text{op} \ re\{rd\})$  that returns
13:     $ISD\{dep\}$  then
14:      return ISD $\{dep\}$ 
15:    else return UKD $\{dep\}$ 
```



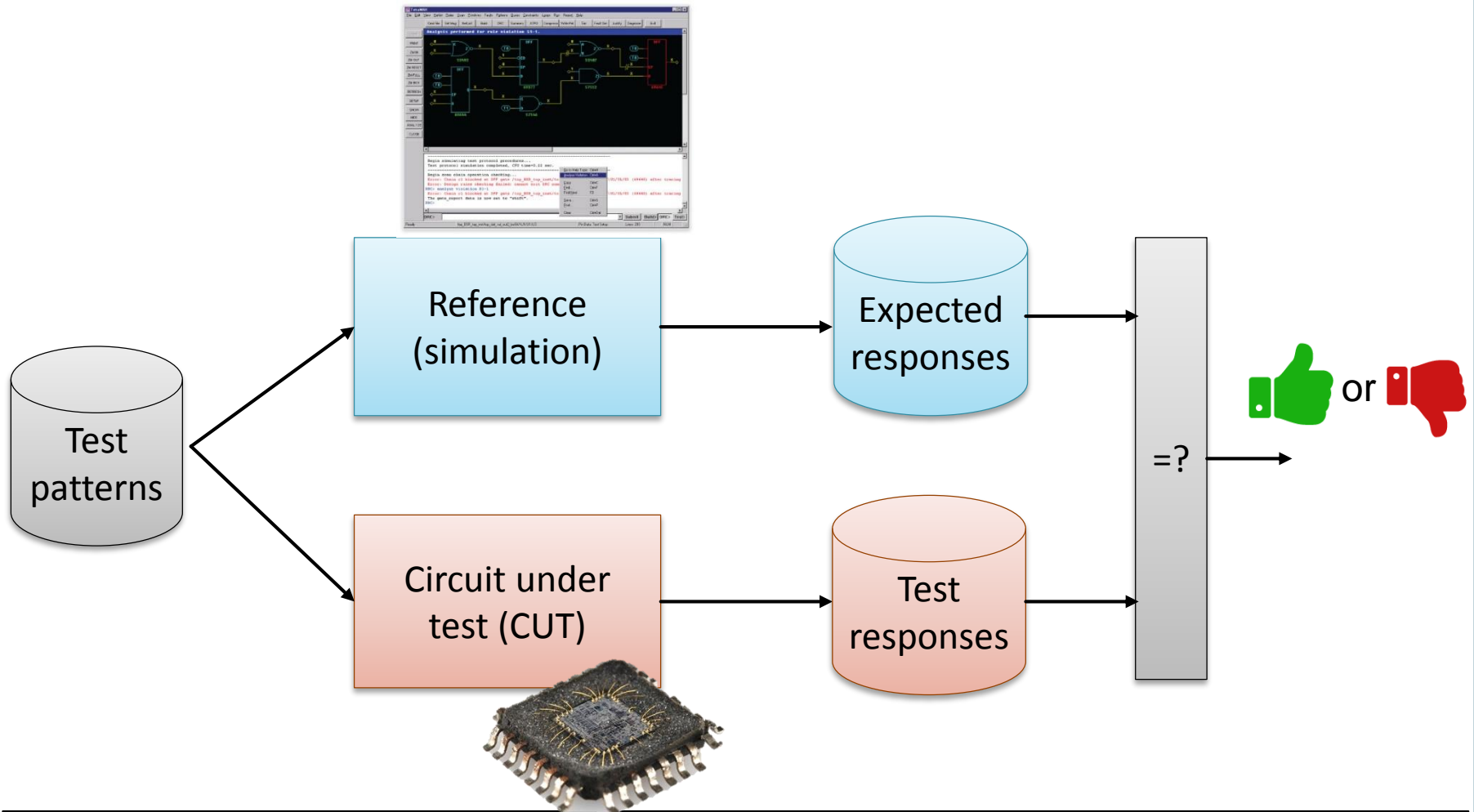
SCAN CHAIN ENCRYPTION, A COUNTERMEASURE AGAINST SCAN ATTACKS

Mathieu Da Silva, Marie-Lise Flottes, Giorgio Di Natale, Bruno Rouzeyre

PHISIC 2018

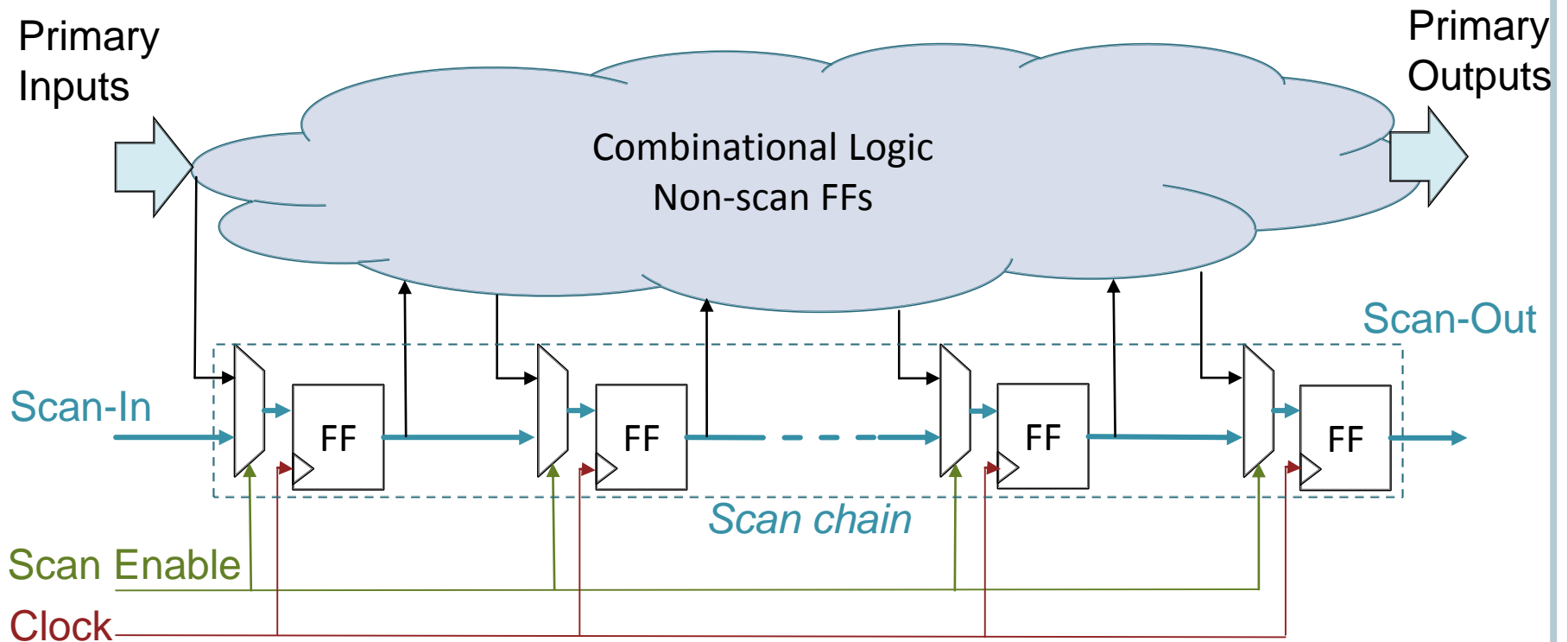
CONTEXT

- Test of circuit is a mandatory step in IC production



CONTEXT

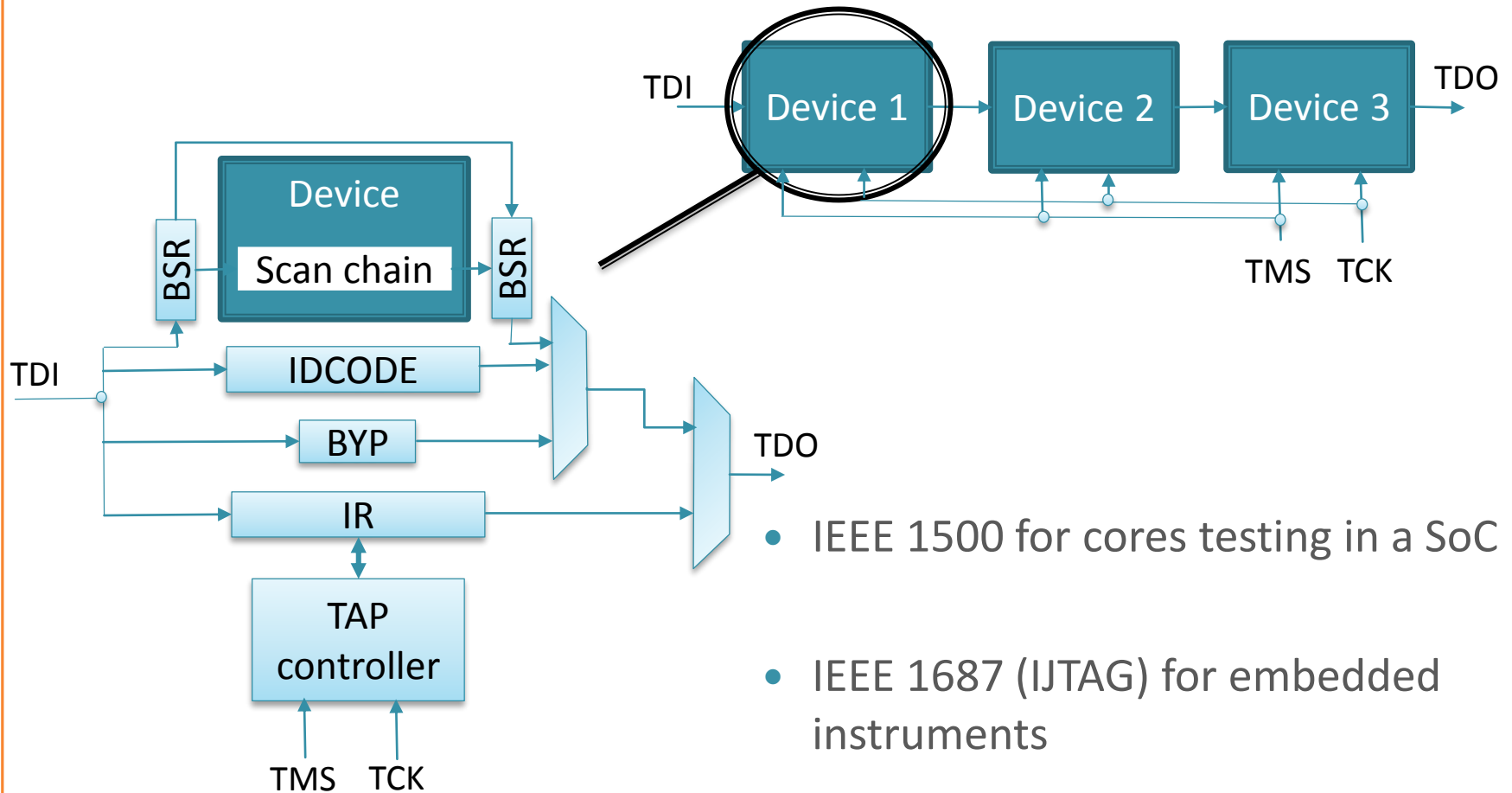
- Most popular method for Design-for-Test = Scan chains
 - Replace original FF by Scan FF connected serially together
 - Extra port « Scan-In » => controllability on internal states
 - Extra port « Scan-Out » => observability on internal states



CONTEXT

- Test standards

- IEEE 1149 (JTAG) for board testing

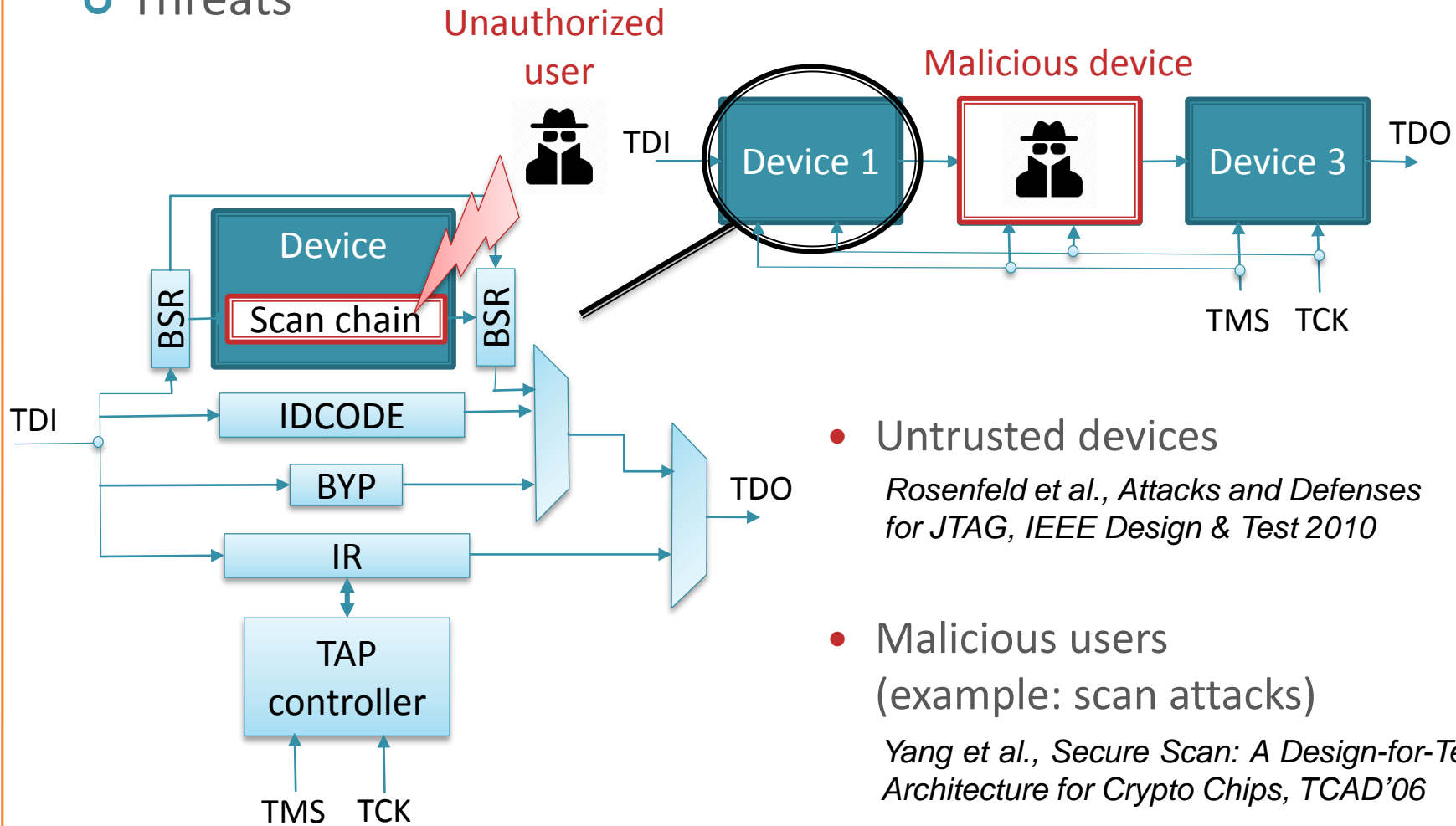


- IEEE 1500 for cores testing in a SoC
- IEEE 1687 (IJTAG) for embedded instruments



CONTEXT

Threats



- Untrusted devices
Rosenfeld et al., Attacks and Defenses for JTAG, IEEE Design & Test 2010
- Malicious users
(example: scan attacks)
Yang et al., Secure Scan: A Design-for-Test Architecture for Crypto Chips, TCAD'06



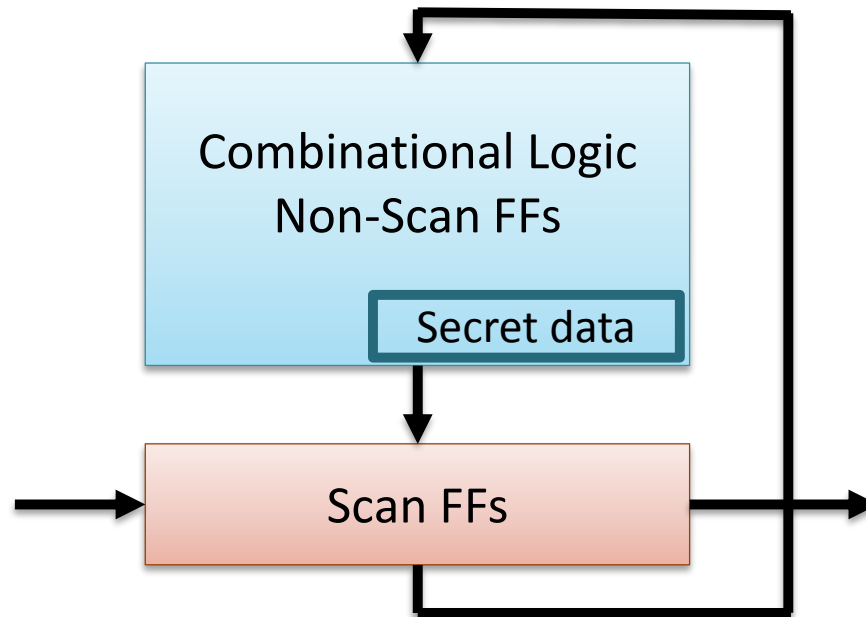
SUMMARY

- 1) **Scan attacks**
- 2) A new countermeasure: Scan chain encryption
- 3) Implementation with block cipher
- 4) Implementation with stream cipher
- 5) Conclusion



SCAN ATTACK PRINCIPLE

- Goal: Retrieve embedded secret data
- Exploit observability or controllability offered by scan chains
- Principle: switch between functional and scan modes
- Main target: secret key of crypto-processors (example: AES)



SCAN ATTACK ON AES

Advanced Encryption Standard (AES)

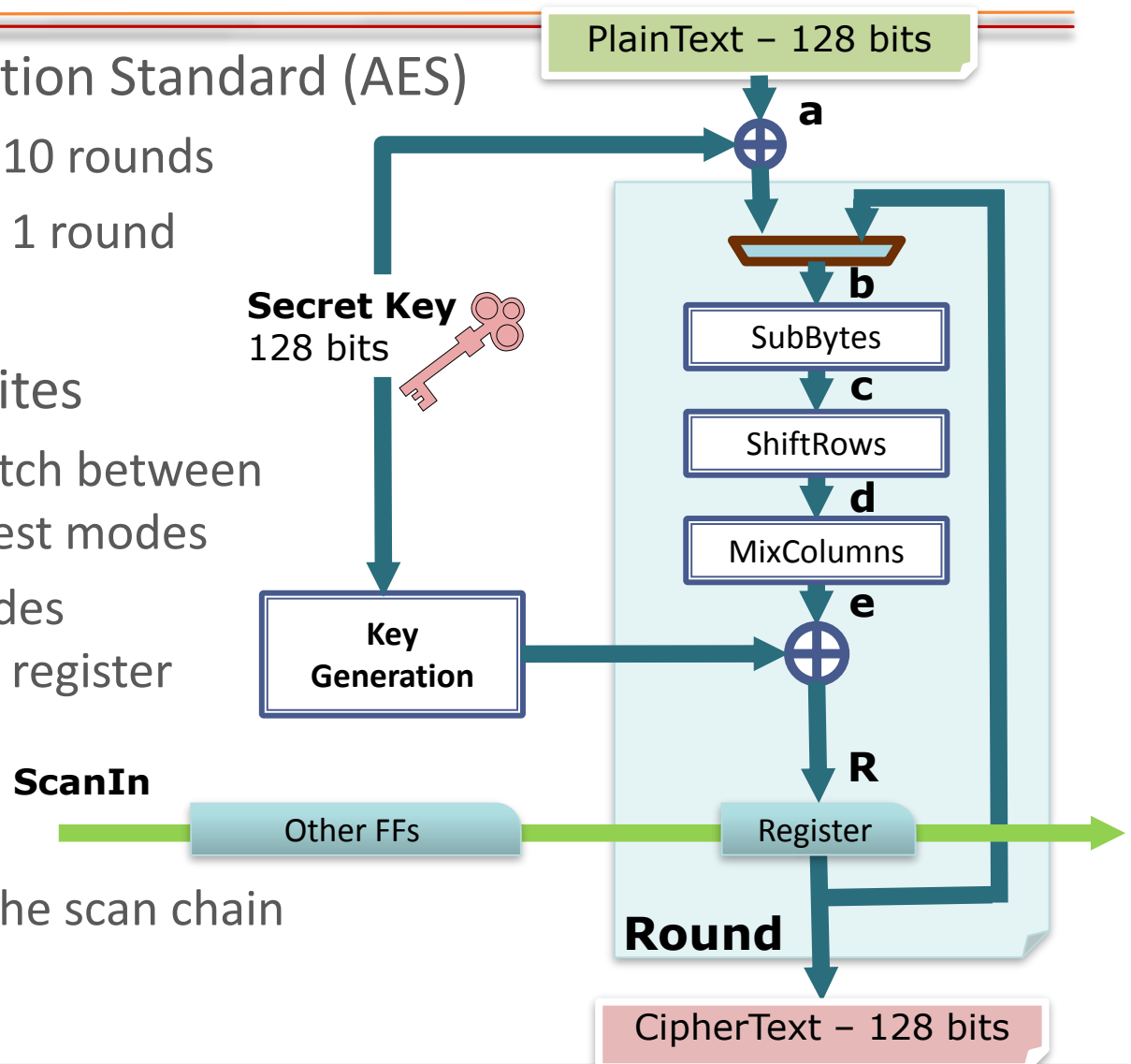
- Ciphertext after 10 rounds
- Not secure after 1 round

Attack pre-requisites

- Attacker can switch between functional and test modes
- Scan chain includes FFs of the round register

Attack principle

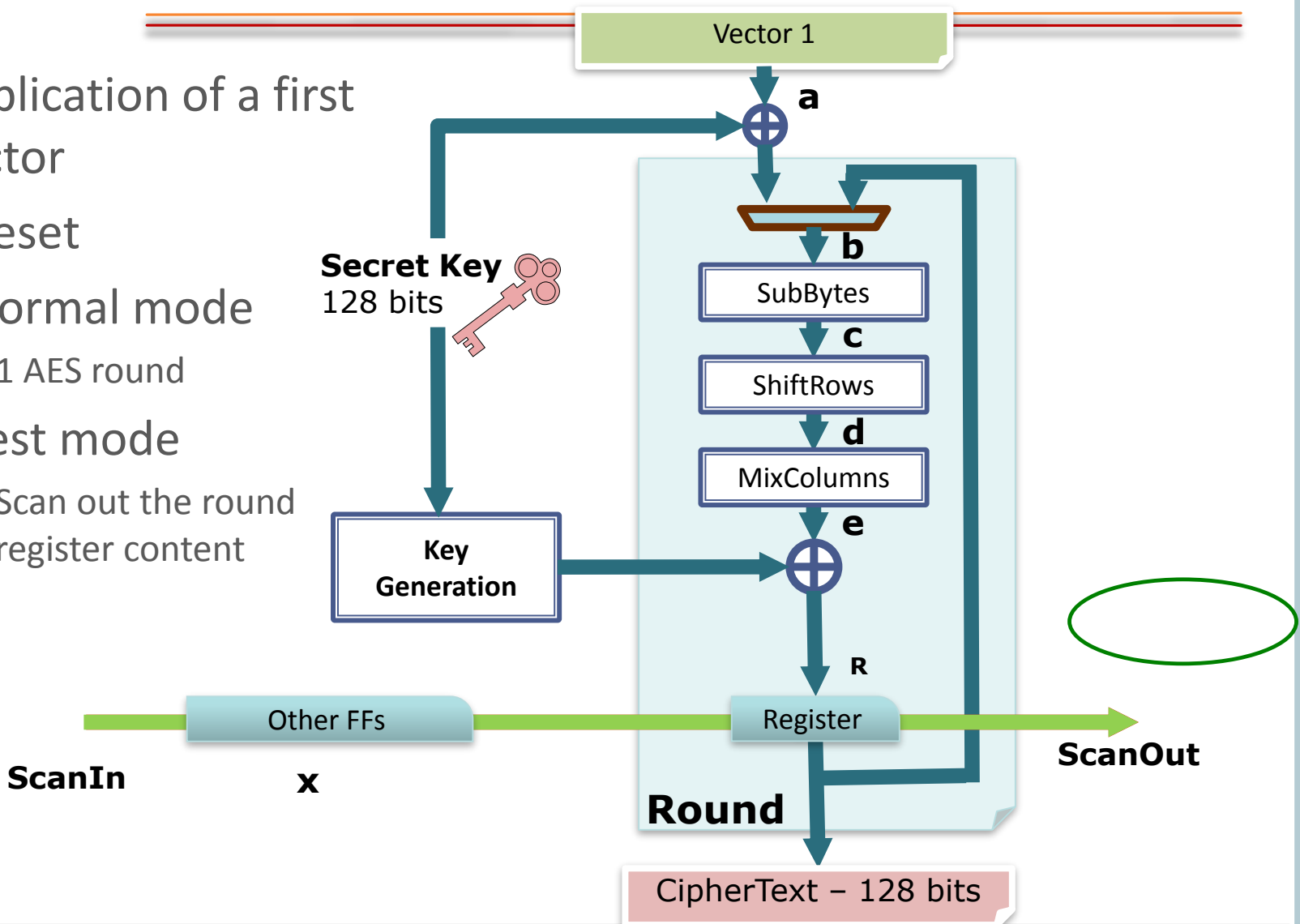
- Observation of the scan chain after 1 round



DIFFERENTIAL ATTACK

○ Application of a first vector

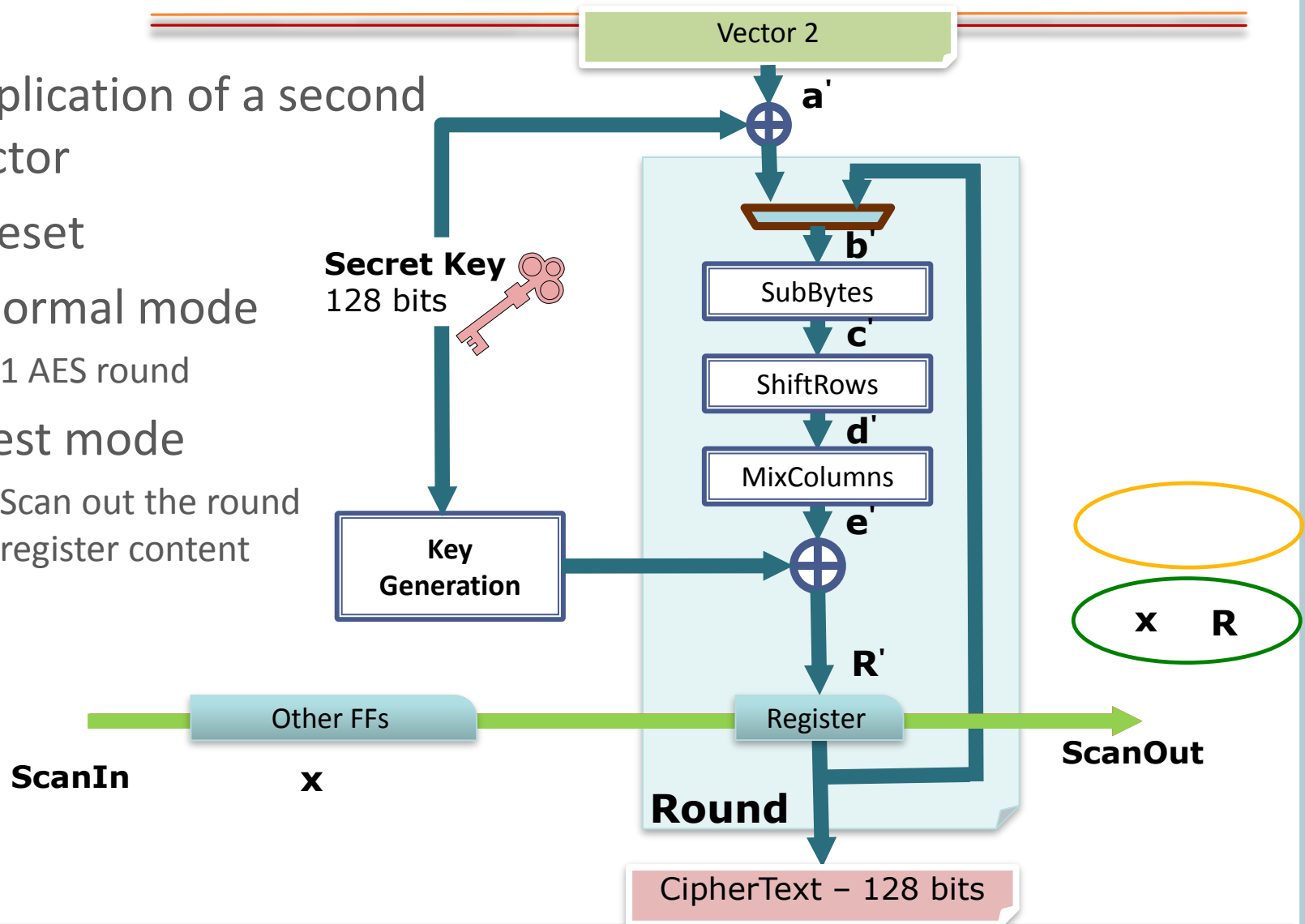
- 1) Reset
- 2) Normal mode
 - 1 AES round
- 3) Test mode
 - Scan out the round register content



DIFFERENTIAL ATTACK

- Application of a second vector

- 1) Reset
- 2) Normal mode
 - 1 AES round
- 3) Test mode
 - Scan out the round register content



DIFFERENTIAL ATTACK

- Hamming distance



- Attacker applies pairs of input values until hamming distance equal to specific values => key byte revealed

- On average, 32 trials

⇒ 512 trials to retrieve the whole 128-bit key



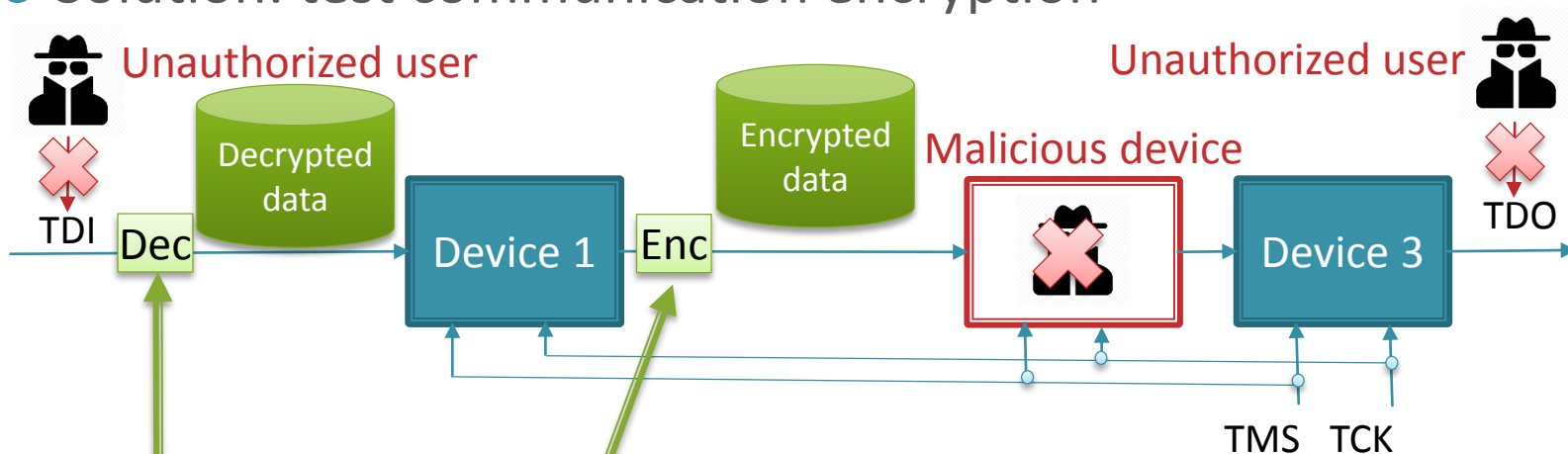
SUMMARY

- 1) Scan attacks
- 2) A new countermeasure: Scan chain encryption**
- 3) Implementation with block cipher
- 4) Implementation with stream cipher
- 5) Conclusion



SCAN CHAIN ENCRYPTION

- Solution: test communication encryption

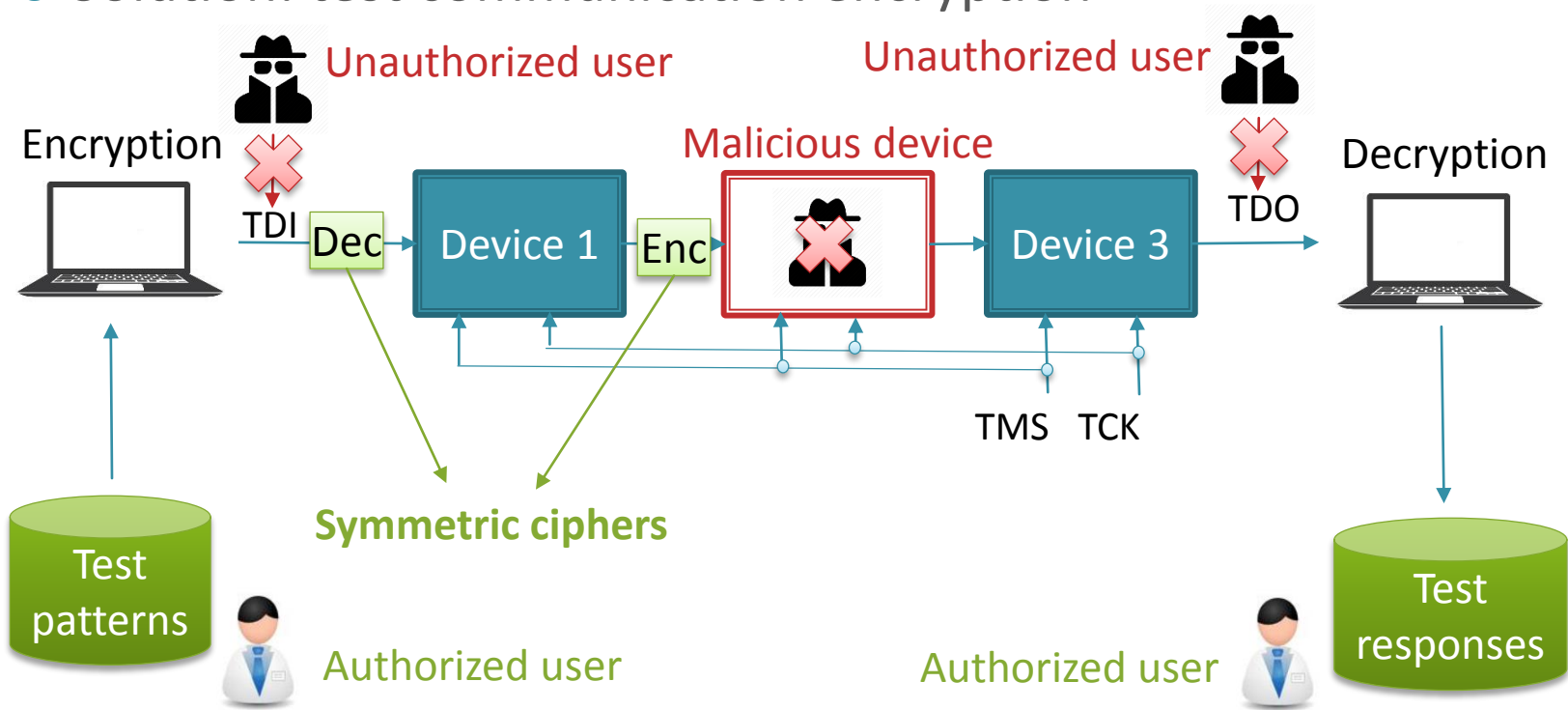


- **Input decryption** prevents sending desired test data
- **Output encryption** prevents reading plain test responses



CONTEXT

○ Solution: test communication encryption

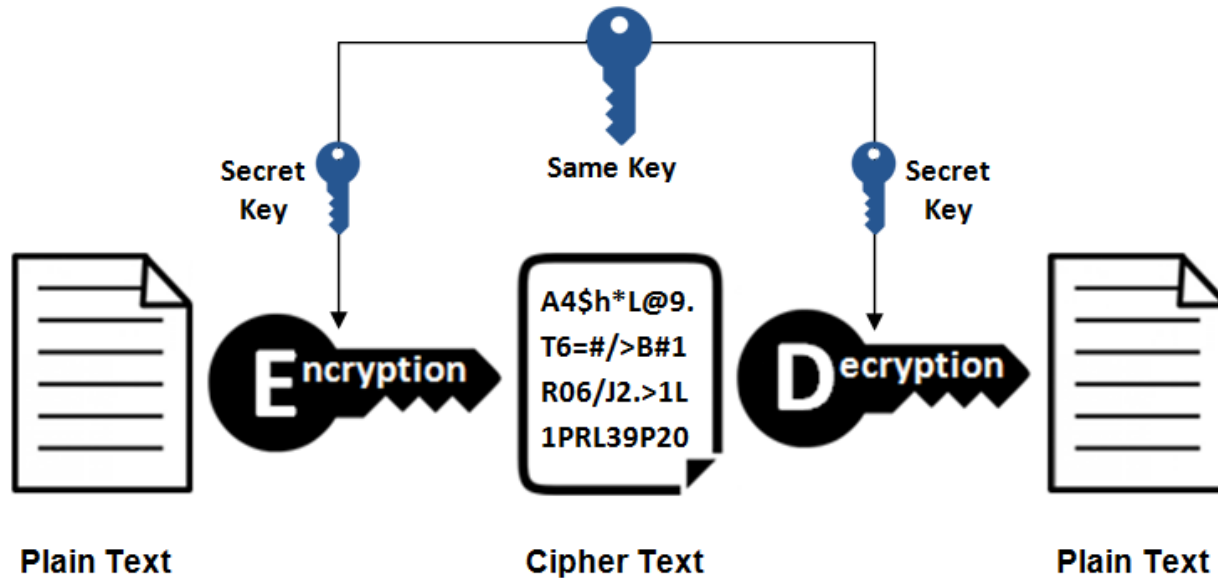


- Input decryption prevents sending desired test data
- Output encryption prevents reading plain test responses
- Test/debug only possible by authorized user knowing the secret key



SYMMETRIC CIPHER

Symmetric Encryption



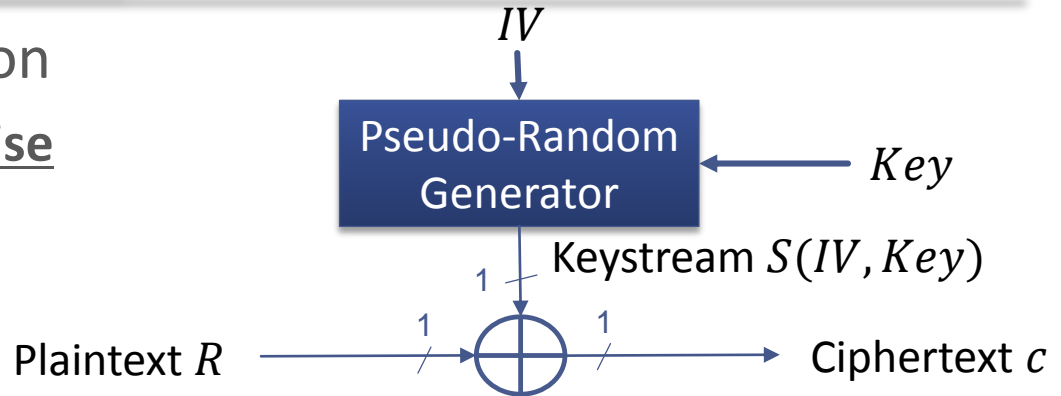
- 2 types of symmetric cipher: stream and block ciphers



STREAM CIPHER / BLOCK CIPHER

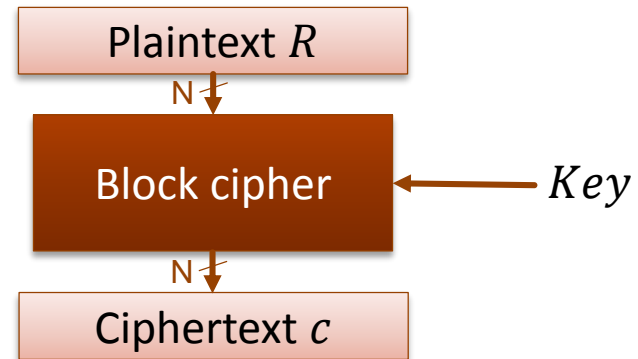
Stream cipher encryption

- Keystream XORed **bitwise** with the plaintext



Block cipher encryption

- Confusion and diffusion on a **block** of plaintext



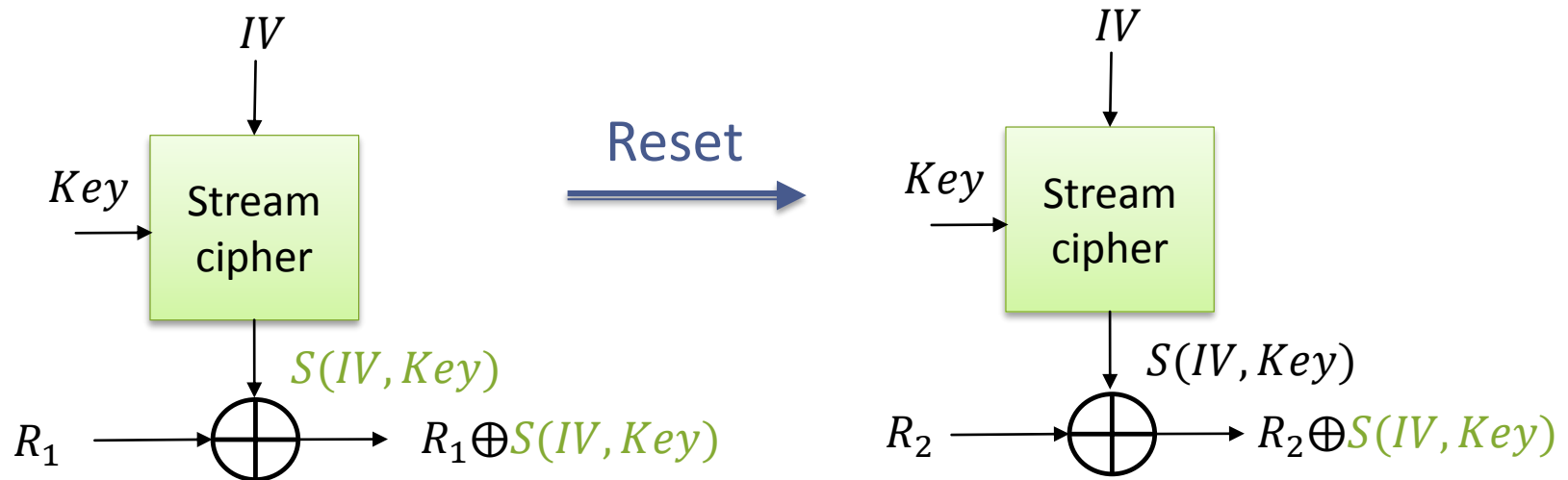
Preference for stream ciphers

- "Naturally" adapted to serial test communication (JTAG, IEEE 1500, IJTAG)
- Smaller area footprint compared to block ciphers
- But ..



TWO-TIMES PAD: STREAM CIPHER REQUIREMENT

- Two-times pad: same key and IV re-used \Rightarrow same keystream generated to encrypt different data



\Rightarrow Possible to carry out attacks if requirement is not fit

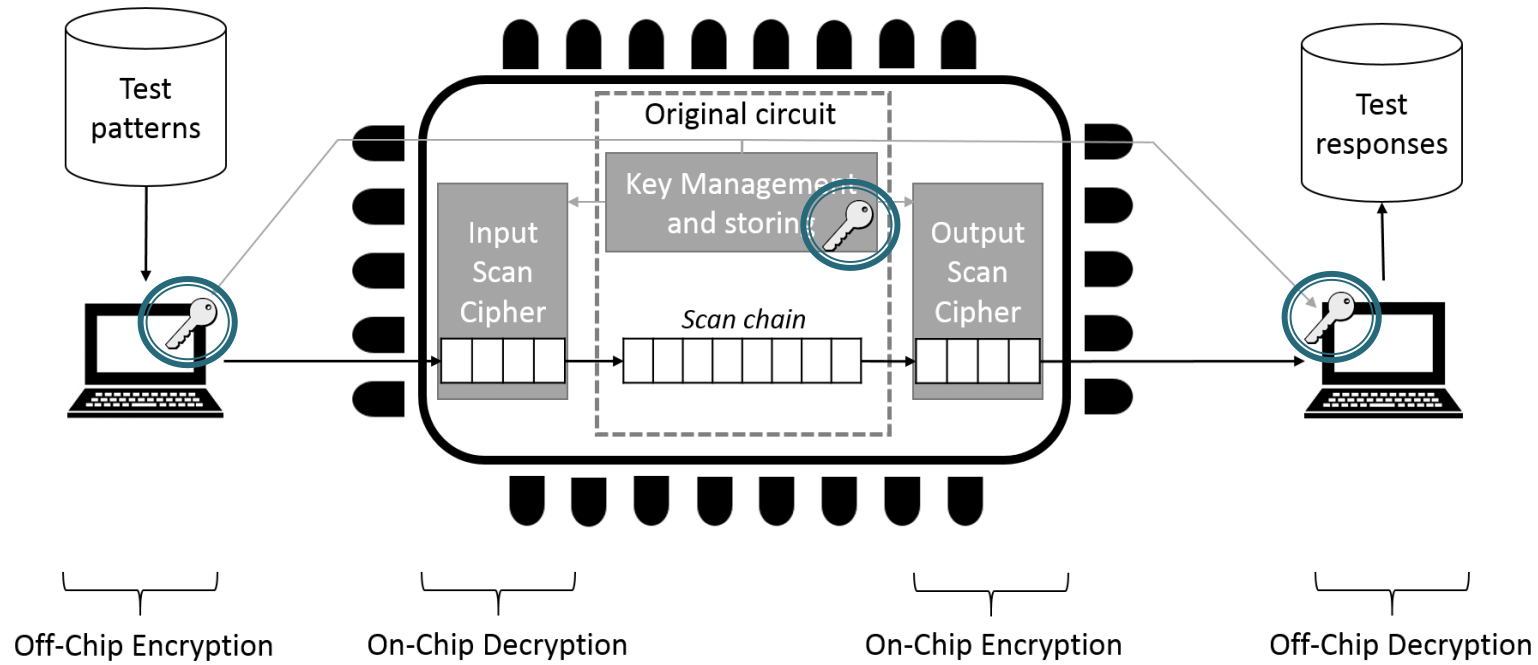
$$R_1 \oplus \cancel{S(IV, Key)} \oplus R_2 \oplus \cancel{S(IV, Key)}$$

\Rightarrow Solution: IV generated randomly at each circuit reset

$$R_1 \oplus S(IV_1, Key) \oplus R_2 \oplus S'(IV_2, Key)$$



BASIC SCHEME



- Assumption: original circuit embedded a crypto-core with its key management and storing
- Scan chain encryption solution shares the key management and storing already implemented



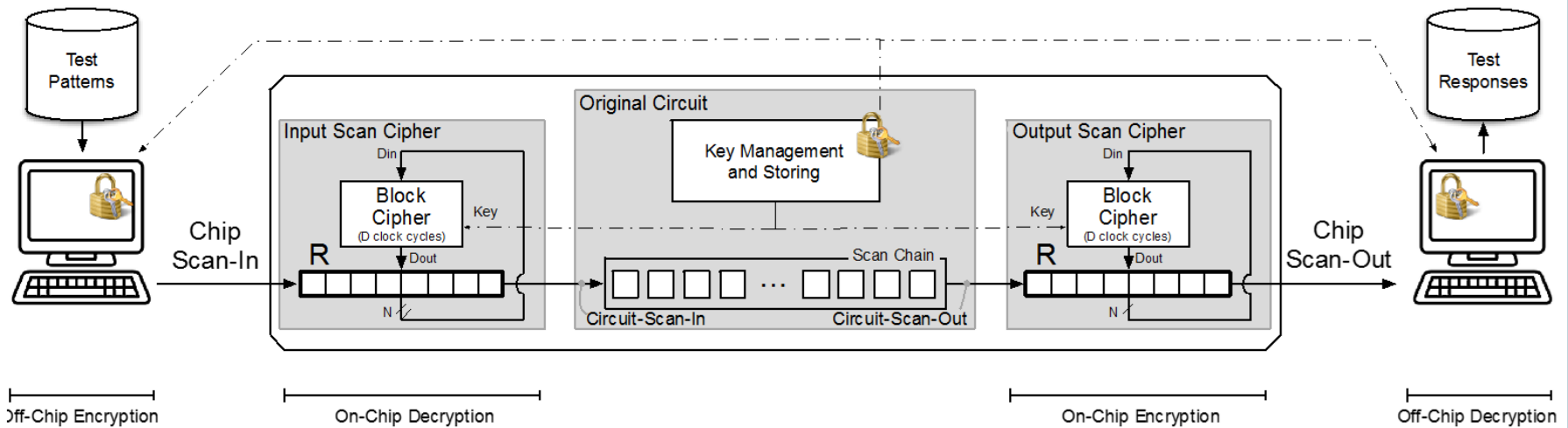
SUMMARY

- 1) Scan attacks
- 2) A new countermeasure: Scan chain encryption
- 3) Implementation with block cipher**
- 4) Implementation with stream cipher
- 5) Conclusion



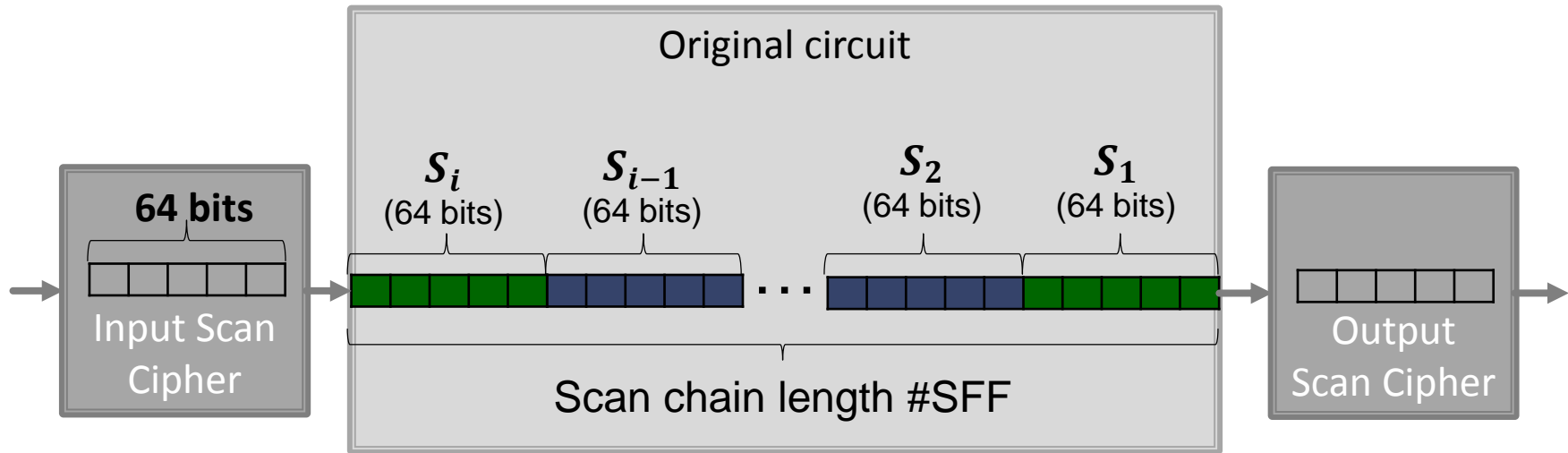
BLOCK CIPHER-BASED SCAN ENCRYPTION

- Implementation on scan chain with 2 PRESENT block ciphers:
 - Lightweight (1 PRESENT = 2 139 GE)
 - Encryption by 64-bits block size



MODE OF OPERATIONS

- 64 bits encrypted every 32 clock cycles



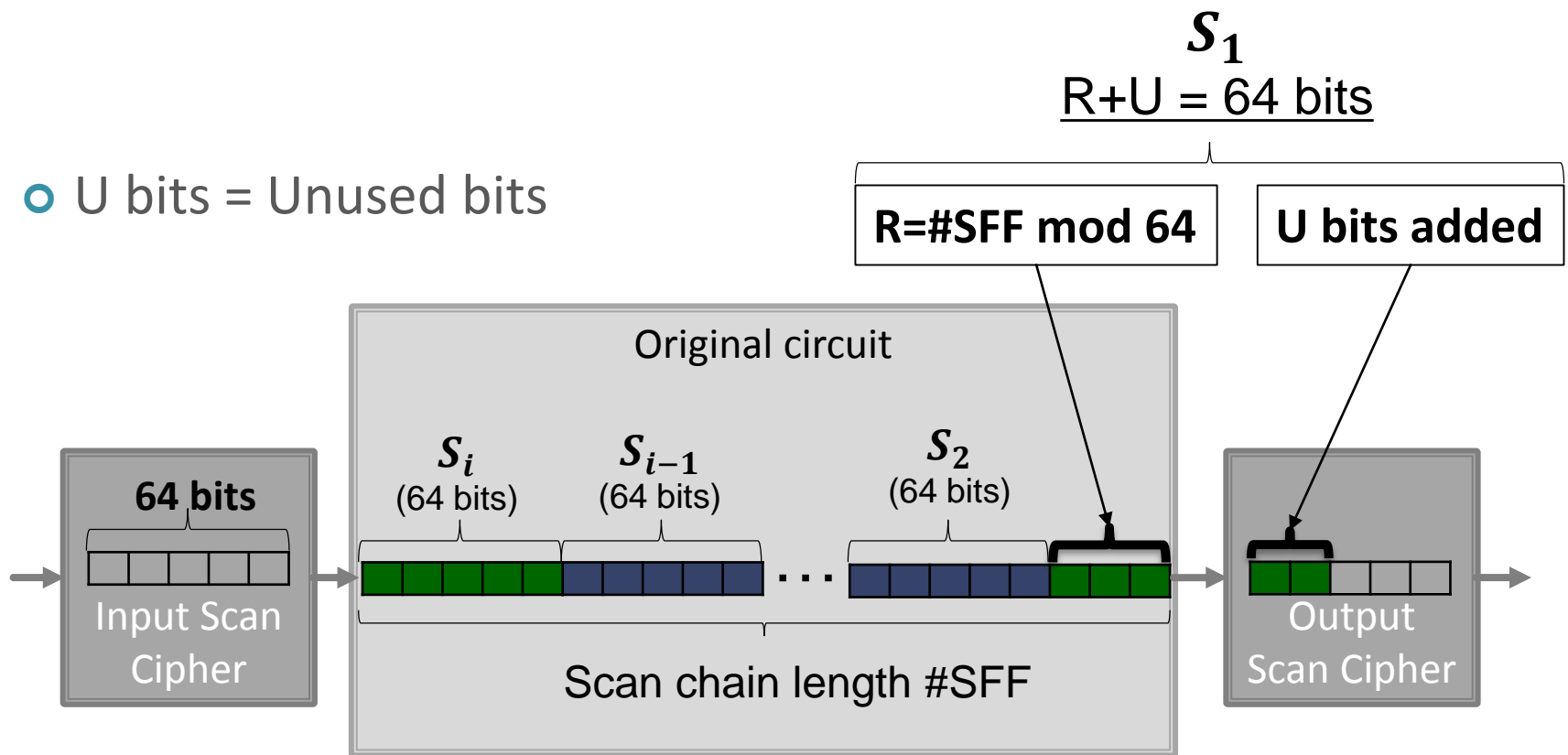
⇒ **$\#SFF = P \times 64$**

⇒ **No test time overhead on each pattern**



MODE OF OPERATIONS

- U bits = Unused bits



$\Rightarrow \#SFF = P \times 64 + R$

\Rightarrow Loss of U clock cycles per pattern



SUMMARY

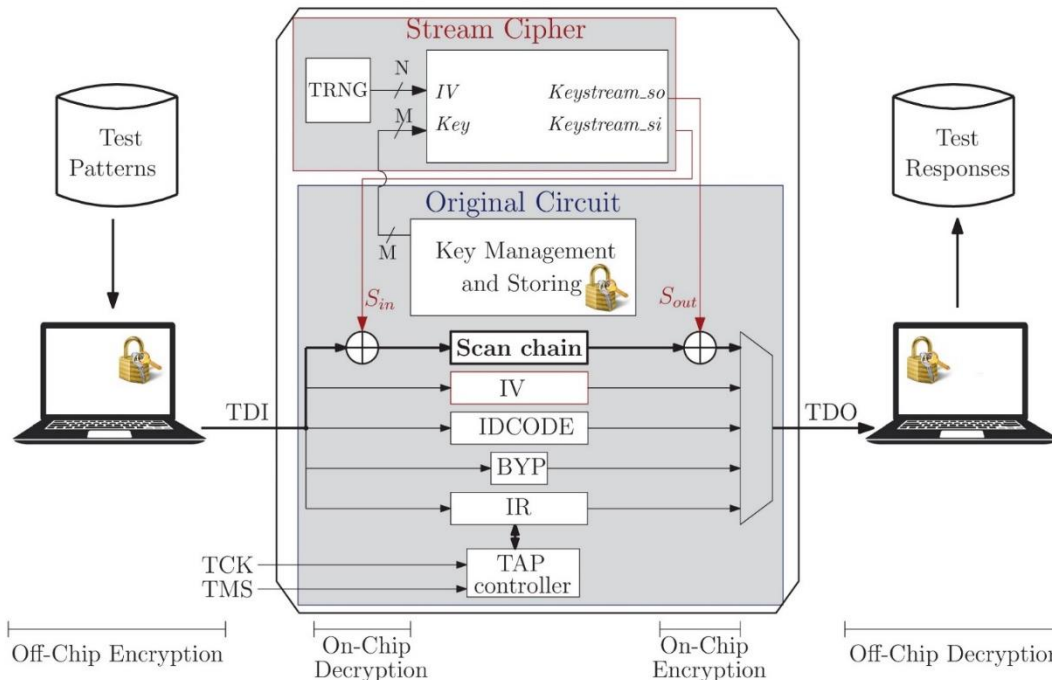
- 1) Scan attacks
- 2) A new countermeasure: Scan chain encryption
- 3) Implementation with block cipher
- 4) Implementation with stream cipher**
- 5) Conclusion



STREAM CIPHER-BASED SCAN ENCRYPTION

Implementation on JTAG:

- 1 TRIVIUM stream cipher (2 016 GE)
- TRNG to generate random IV
- New instruction *GetIV* with a test data register IV

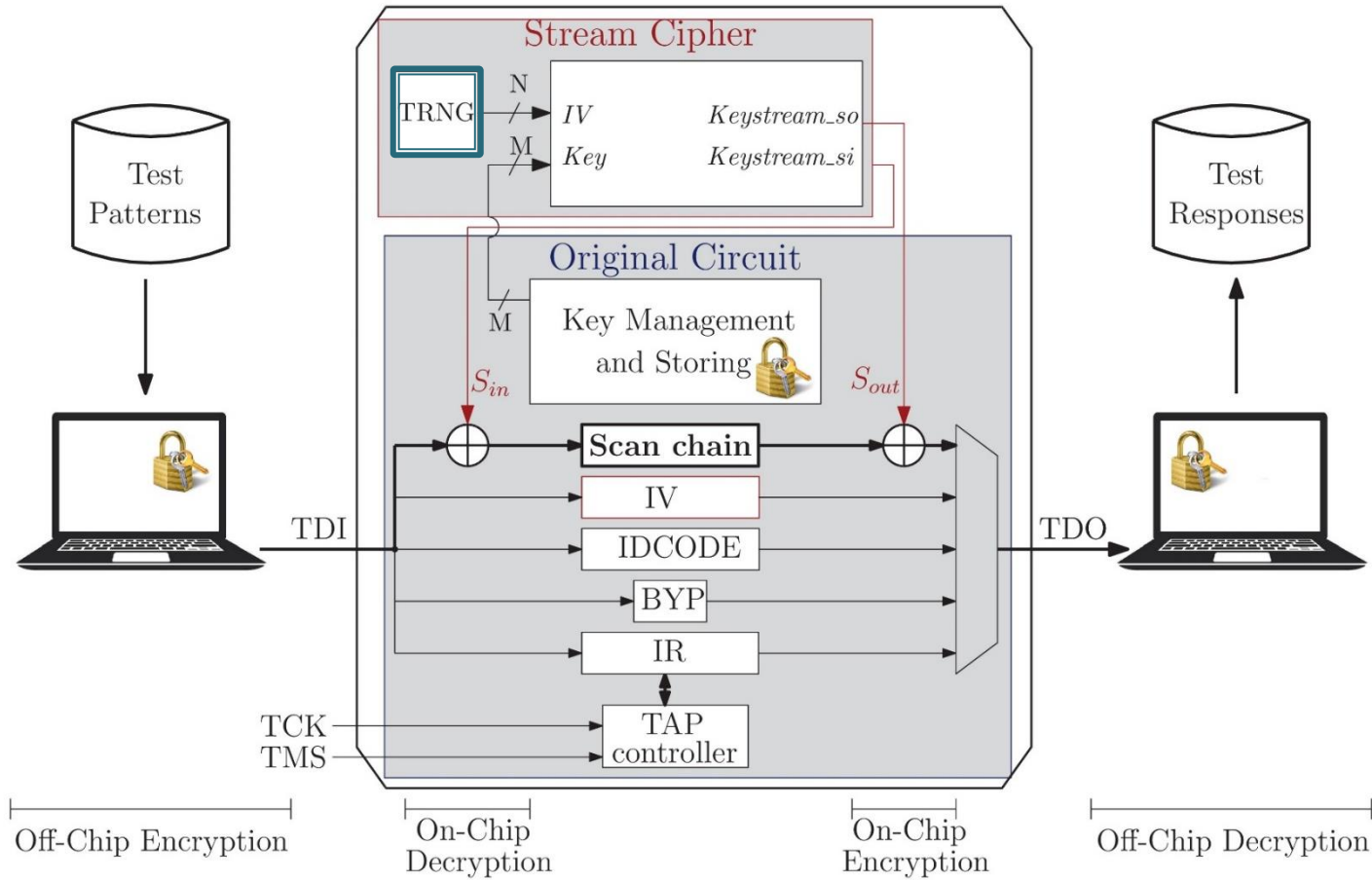


Mode of operations in 2 phases: initialization and encryption

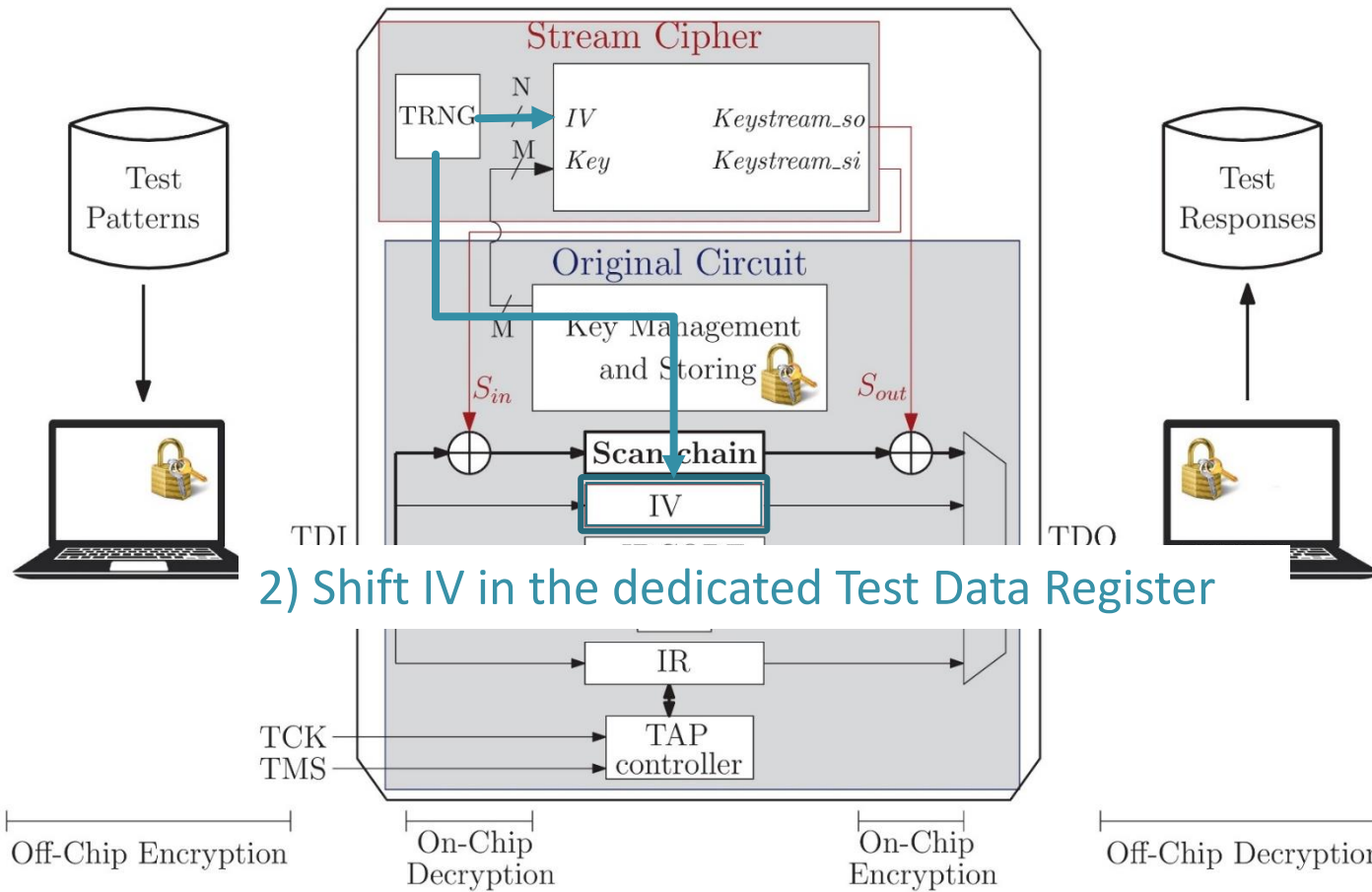


INITIALIZATION PHASE

1) TRNG initialization: reach sufficient entropy to generate random number



INITIALIZATION PHASE

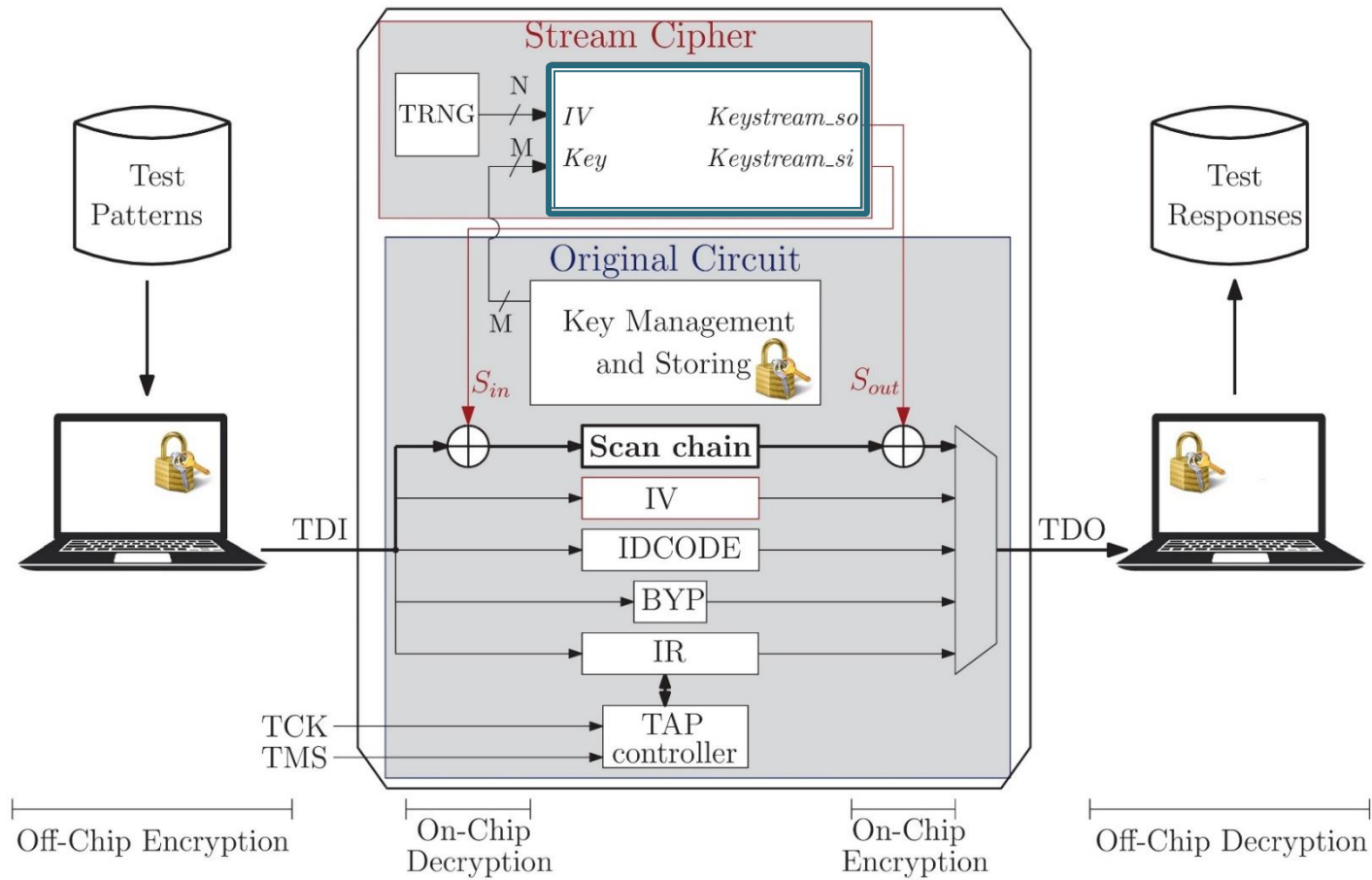


2) Shift IV in the dedicated Test Data Register

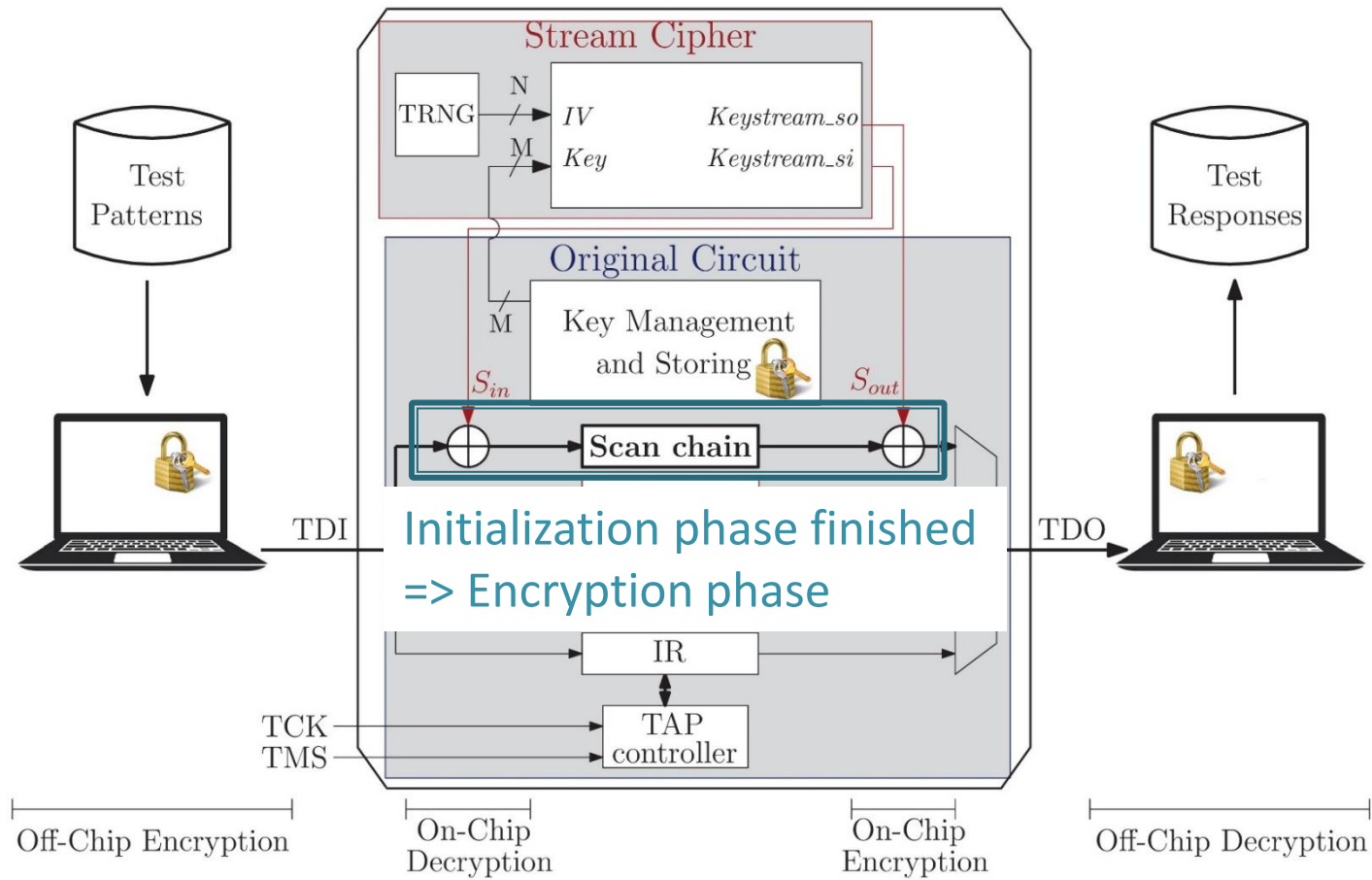


INITIALIZATION PHASE

3) Stream cipher setup

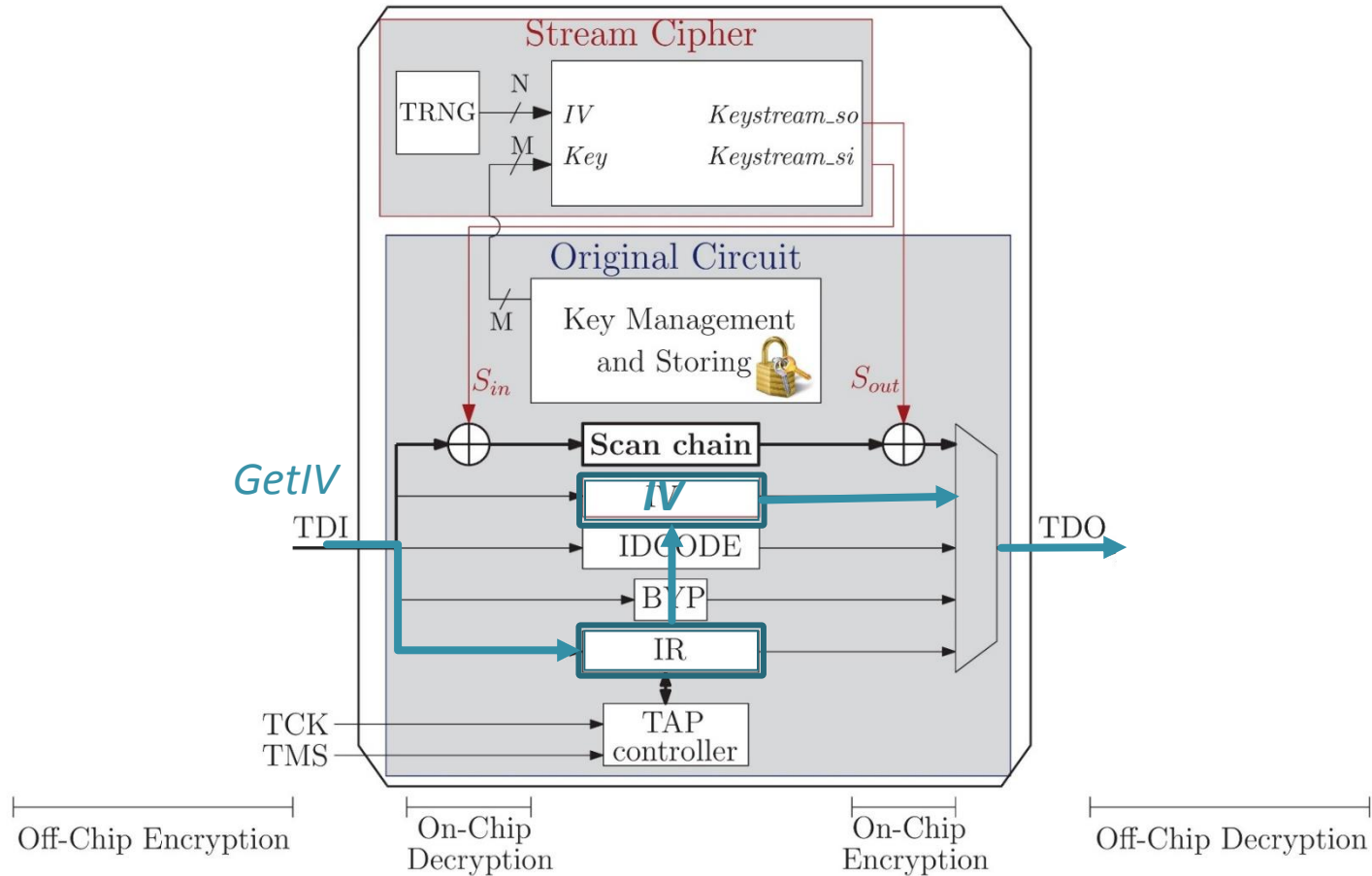


INITIALIZATION PHASE



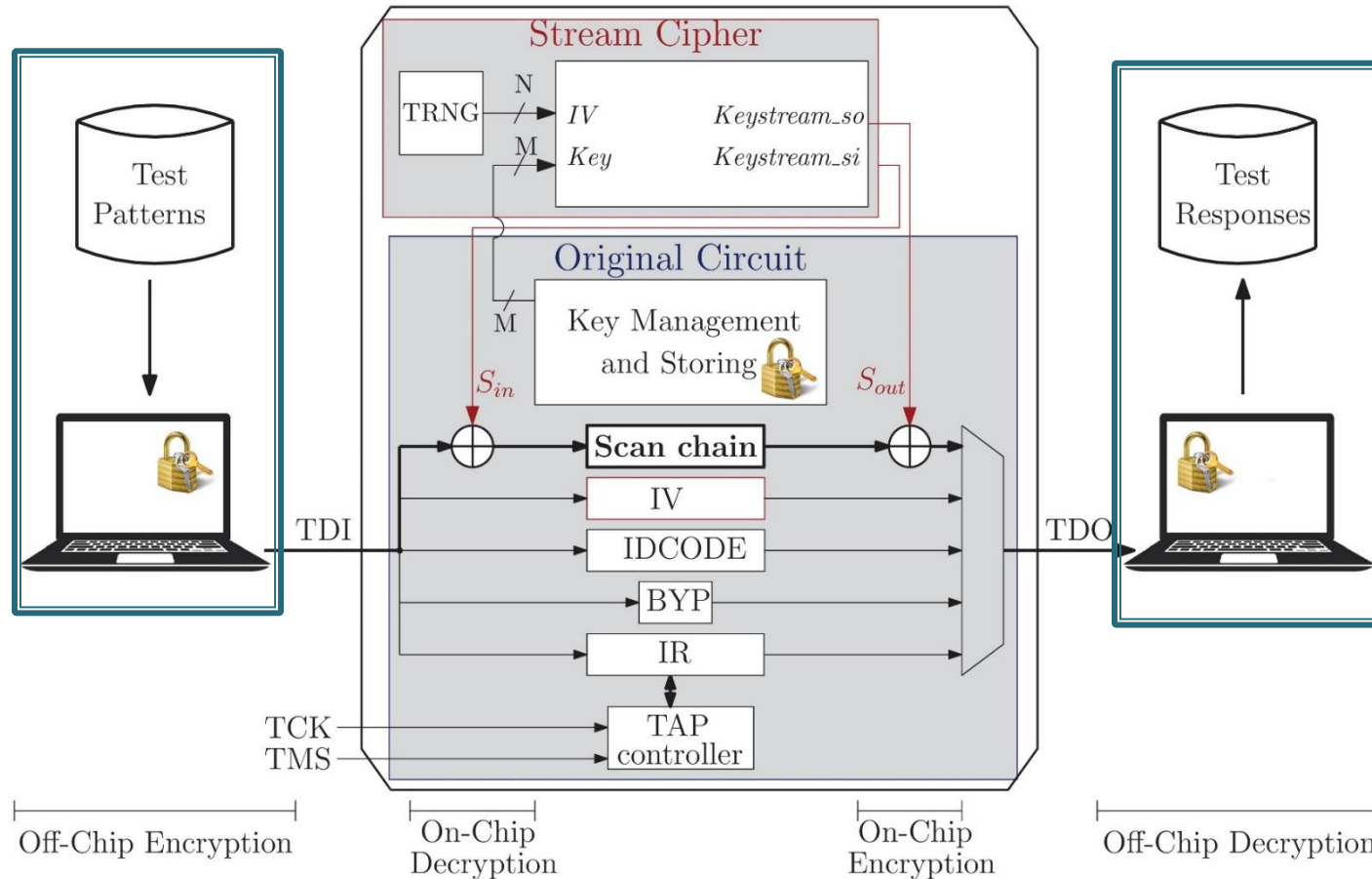
ENCRYPTION PHASE

- Send *GETIV* instruction
- ⇒ Shift the content of the IV register out the circuit



ENCRYPTION PHASE

- User can encrypt and decrypt test data with the **obtained IV** and the **shared secret key**



TIME FOR THE INITIALIZATION PROCESS

- T_{TRNG_init} to initialize the TRNG
- 80 clock cycles to shift the IV in the register
- 1 152 clock cycles for the stream cipher setup

Original circuit	Triple-DES	Pipelined AES-128	Pipelined AES-256	RSA 1024	LEON3
Test time* (clock cycles)	687 101	1 944 877	4 559 845	39 405 239	11 612 051
<i>Test time overhead</i>					
Block-based solution (%)	+0.31	+0.81	+0.006	+0.33	+0.004
Stream-based solution (%)**	+0.18	+0.06	+0.03	+0.003	+0.01

*: Test time considered for a fault coverage of 100%, except for LEON3 where it reaches 70%

** : test time overhead without the initialization of the TRNG



SUMMARY

- 1) Scan attacks
- 2) A new countermeasure: Scan chain encryption
- 3) Implementation with block cipher
- 4) Implementation with stream cipher
- 5) **Conclusion**



COMPARISON BETWEEN BOTH SOLUTIONS

	Block cipher-based solution (PRESENT)	Stream cipher-based solution (TRIVIUM)
Security		
- Scan attacks	Protected	Protected (two times pad not possible)
- Malicious core	Protected	Protected
Cost		
- Area	10 658.96 μm^2	5 408.52 μm^2 (+ 31 200 μm^2 for TRNG)
- Test time	Depends on the scan length (multiple or not of the block size)	Clock cycles required for the initialization phase
Integration		
- Diagnosis & debug	Still possible in-field	
- Key management	Re-use key management already implemented	
- Integration in test daisy-chain	Possible issue with the padding of test data	No issue



Thank You



PRESENT & TRIVIUM



PRESENT BLOCK CIPHER

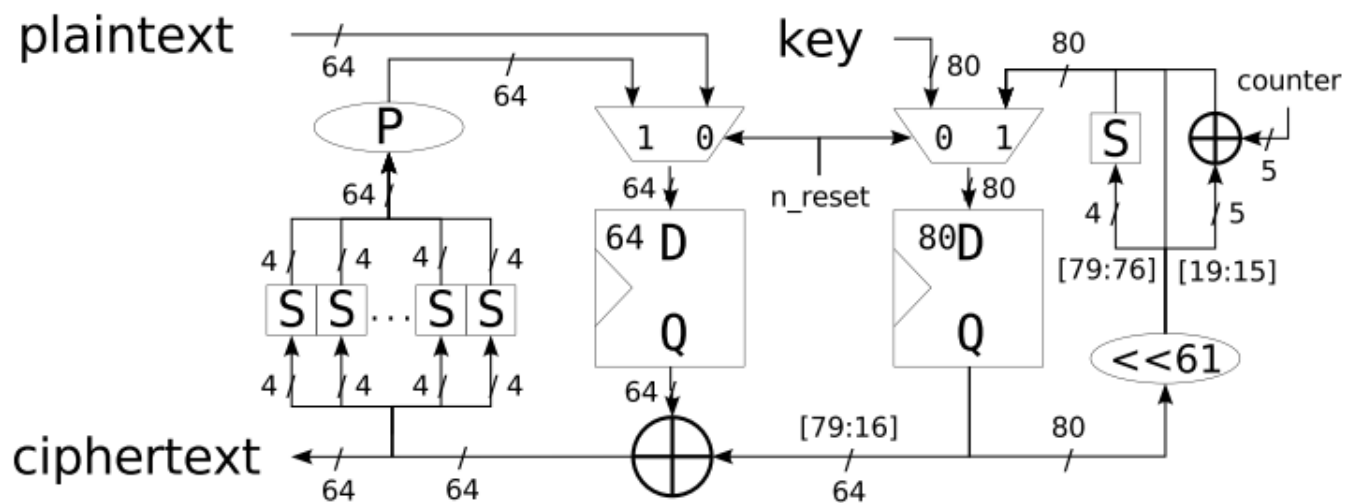


Fig. 4. The datapath of an area-optimized version of PRESENT-80.



TRIVIUM STREAM CIPHER

- Non-Linear Feedback Shift Register (NLFSR) as PRG
- 80-bit secret key Key
- 80-bit Initialization Vector (IV)
- All existing countermeasures based on TRIVIUM stream cipher

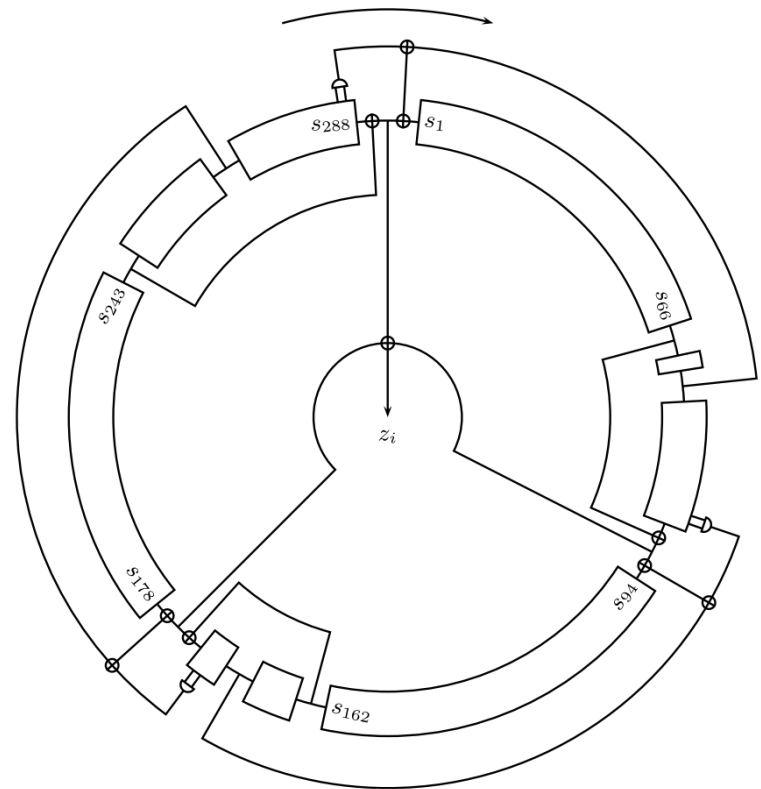


Fig. 1. TRIVIUM

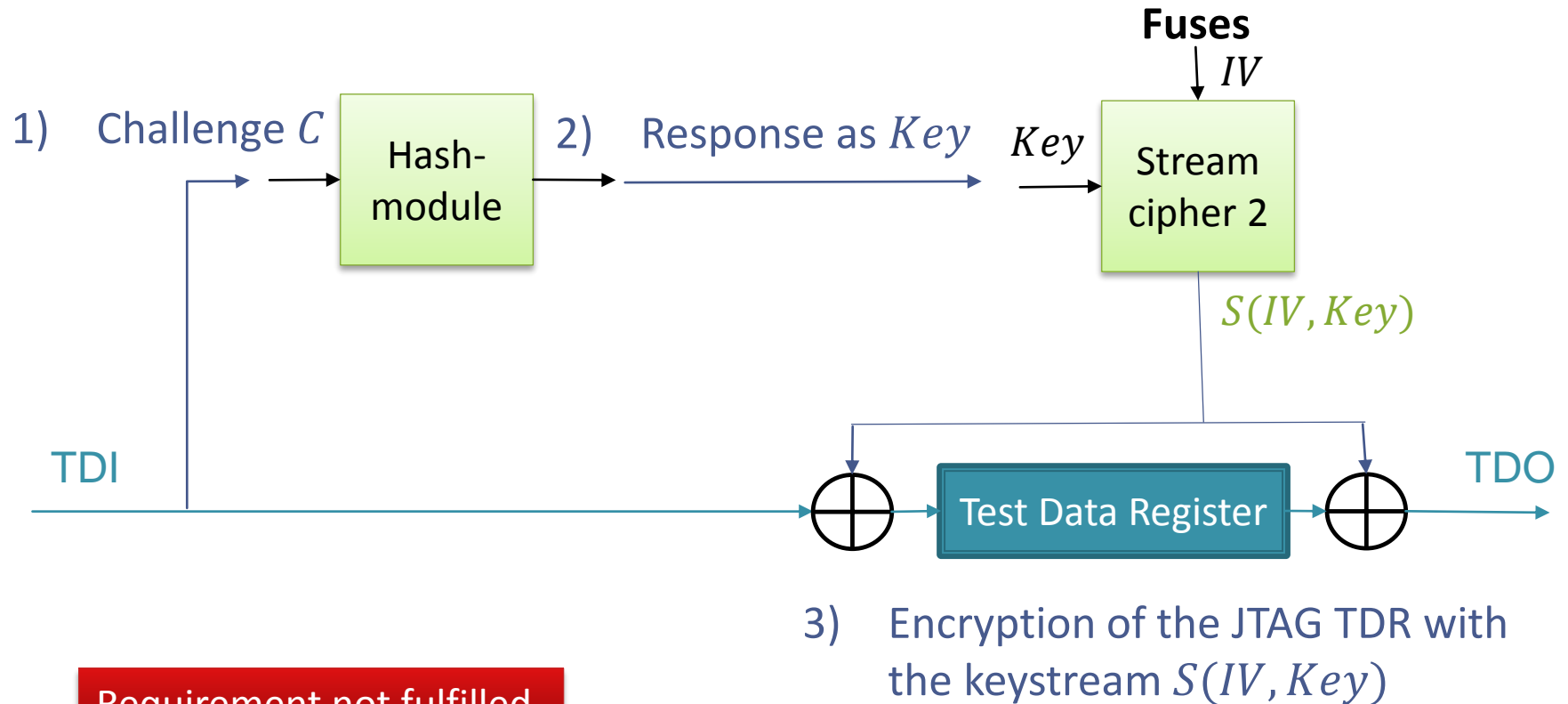


STATE-OF-THE-ART OF THE STREAM CIPHER-BASED SCAN ENCRYPTION



STREAM-BASED ENCRYPTION ON JTAG INTERFACE

- Challenge/Response protocol to encrypt JTAG test communication



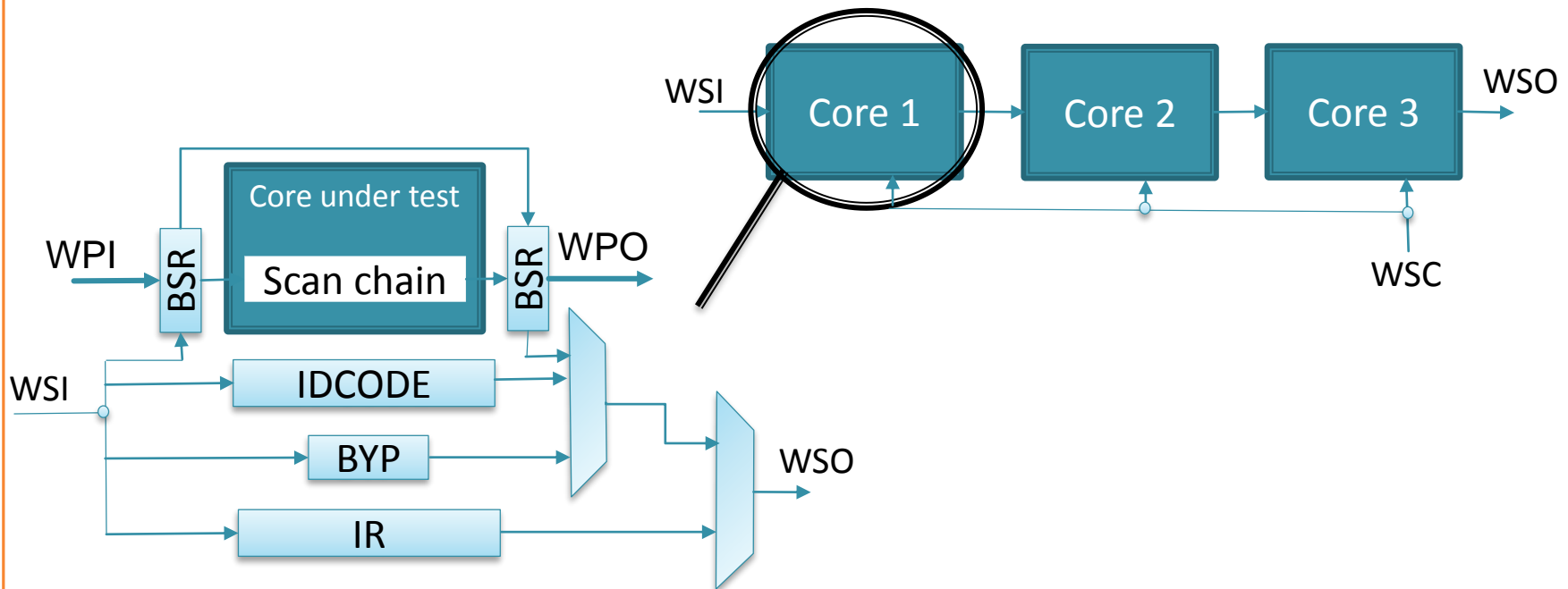
Requirement not fulfilled



STREAM-BASED ENCRYPTION ON IEEE 1500 INTERFACE

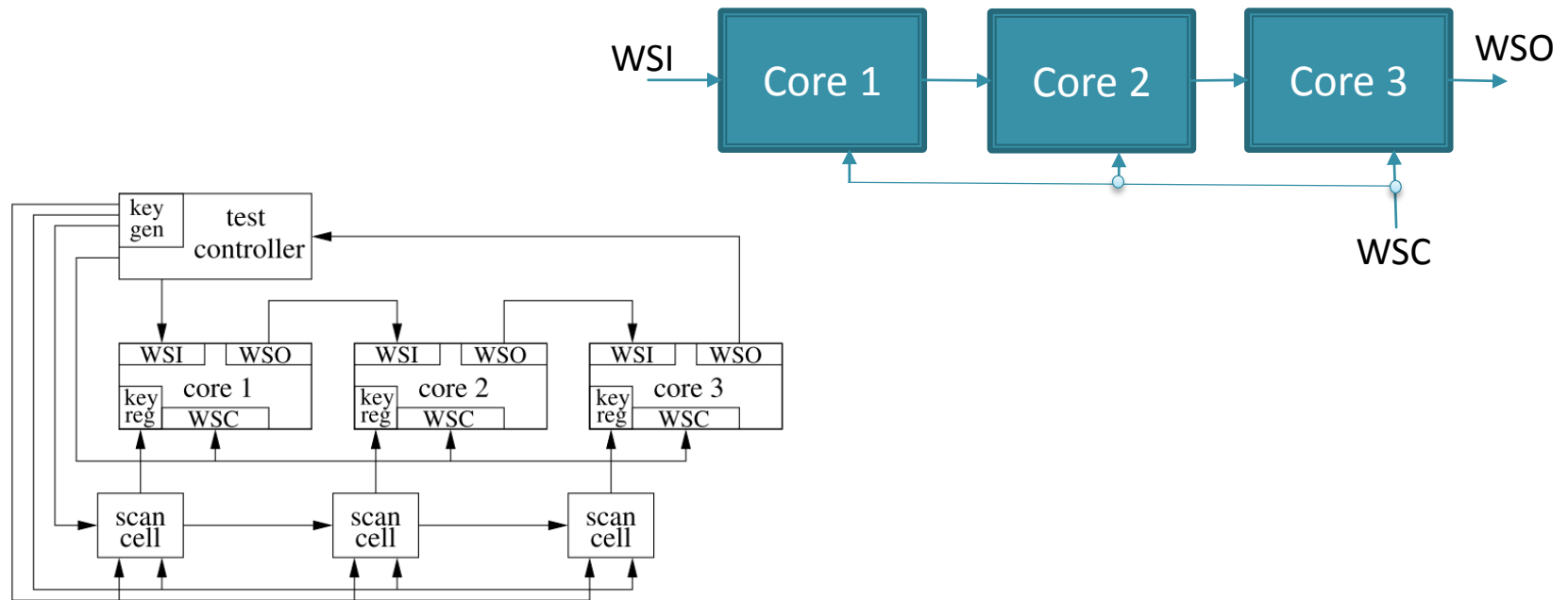
- IEEE 1500 standard

- Similar as JTAG standard, but for SoC wrappers
- Parallel test inputs WPI and parallel test outputs WPO



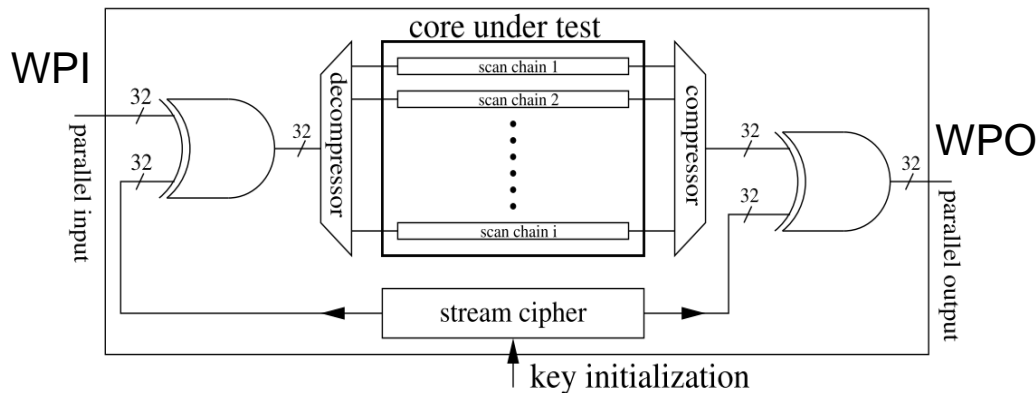
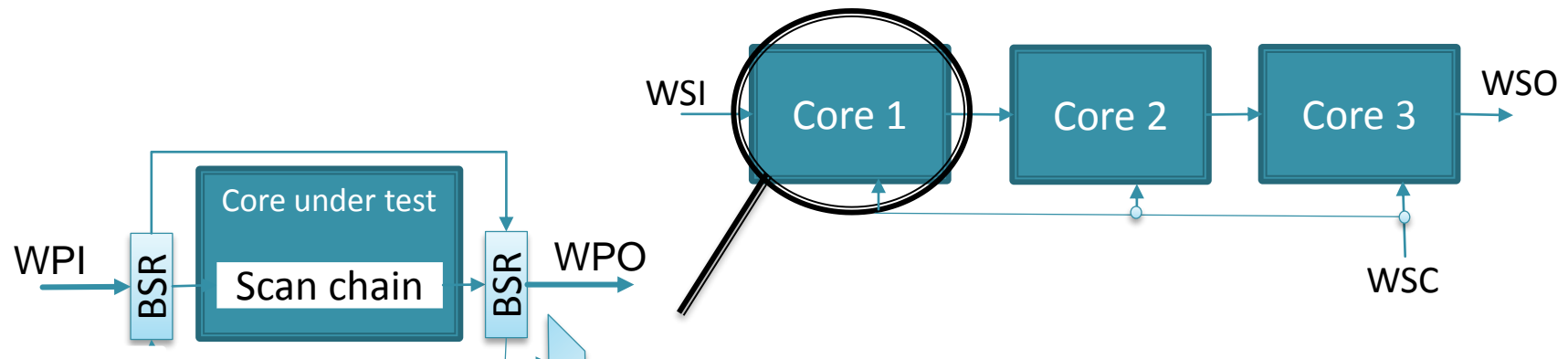
STREAM-BASED ENCRYPTION ON IEEE 1500 INTERFACE

- Encrypt test data on a targeted core (IEEE 1500)
 - 1) Send the key to the core via specific scan chain non-visible from the others cores



STREAM-BASED ENCRYPTION ON IEEE 1500 INTERFACE

- Encrypt test data on a targeted core (IEEE 1500)
 - 1) Encrypt the parallel input/output (WPI and WPO)

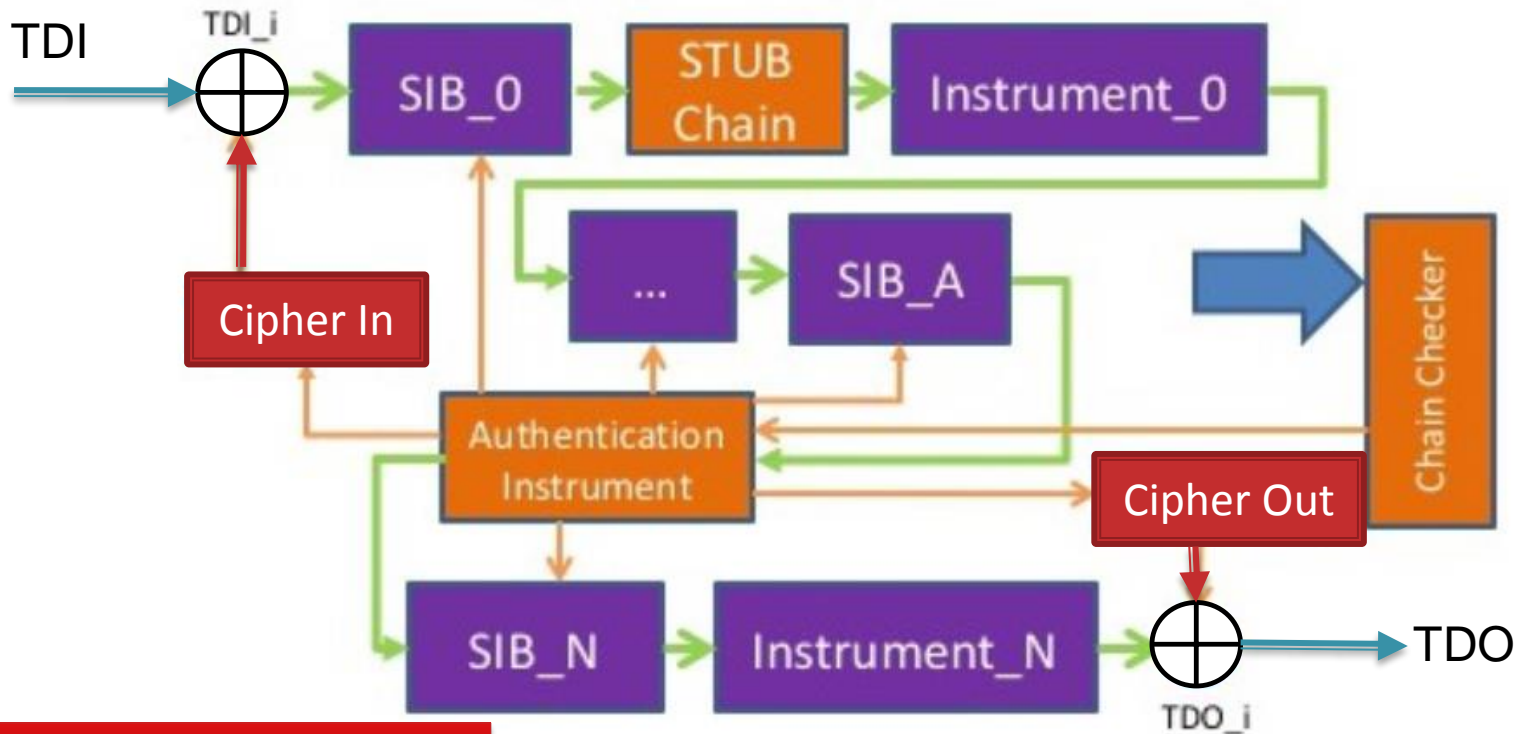


Requirement not fulfilled



STREAM-BASED ENCRYPTION ON JTAG INTERFACE

- Encryption of Test Data Register associated to Instruments in the JTAG network



Requirement not fulfilled



OPTIMIZATION OF THE BLOCK-CIPHER-BASED SCAN ENCRYPTION



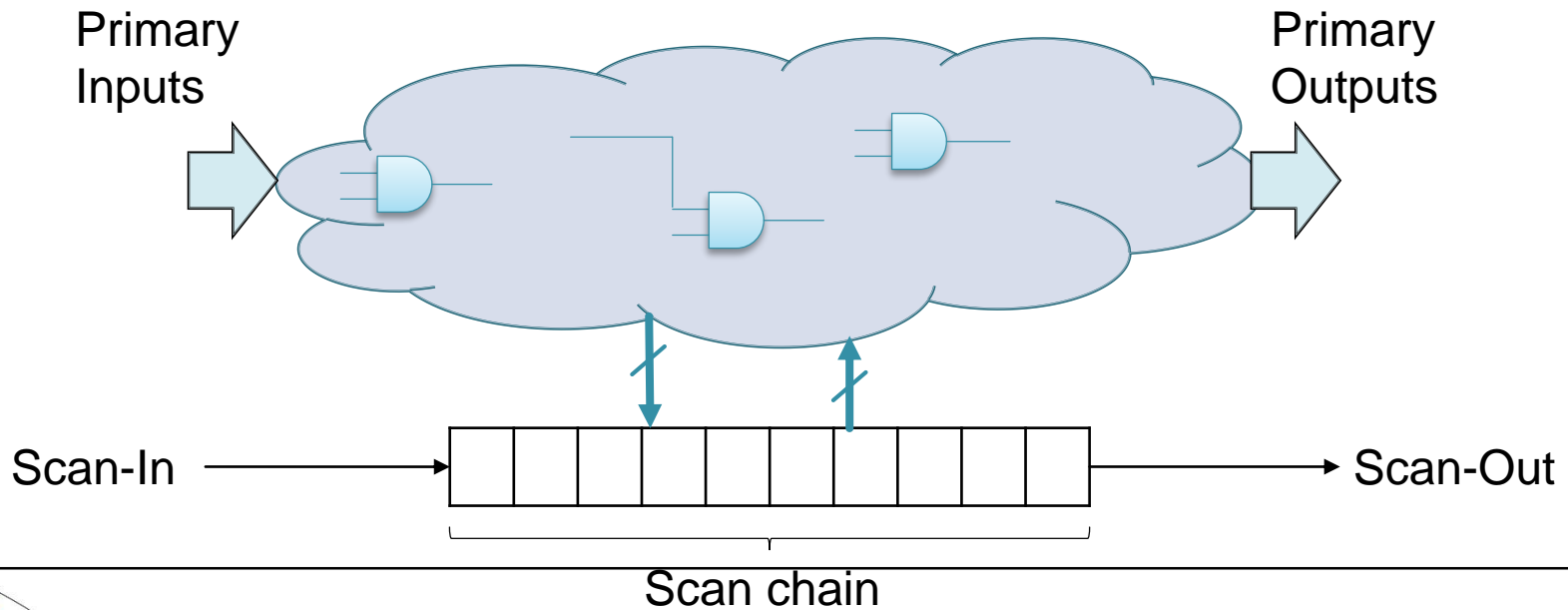
TEST TIME OPTIMIZATION

- Problem

- U additional clock cycles per patterns

- 2 solutions

- 1) Do nothing (=> test time overhead)
- 2) Optimization: **reduce the number of patterns**
 - Extra DfT: insertion of test points



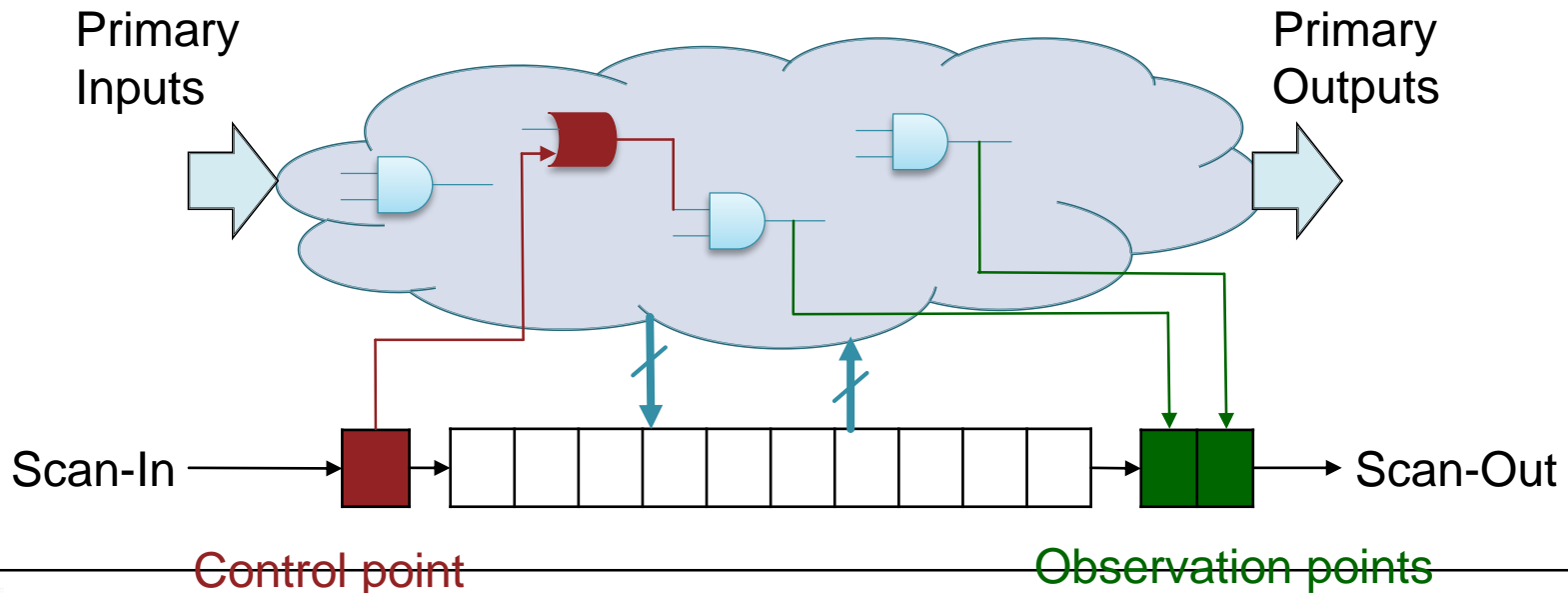
TEST TIME OPTIMIZATION

- Problem

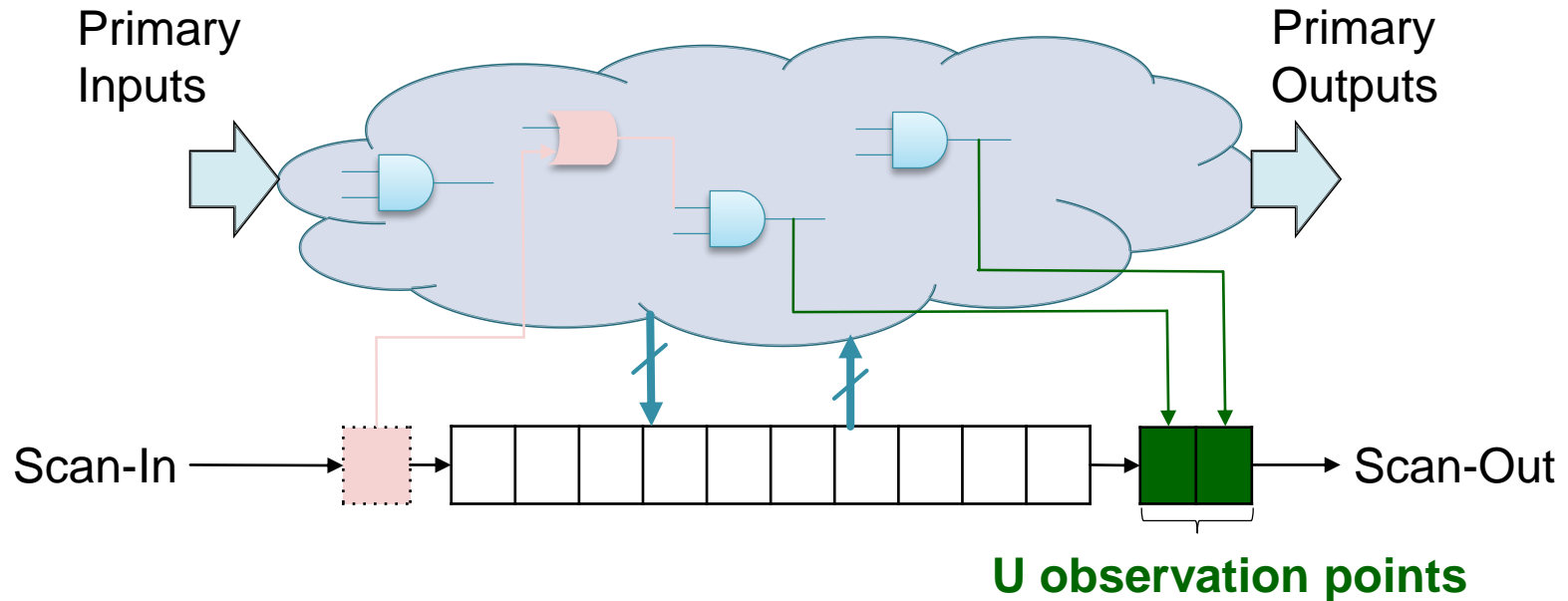
- U additional clock cycles per patterns

- 2 solutions

- 1) Do nothing (=> test time overhead)
- 2) Optimization: **reduce the number of patterns**
 - Extra DfT: insertion of test points



TEST TIME OPTIMIZATION



○ Observation points

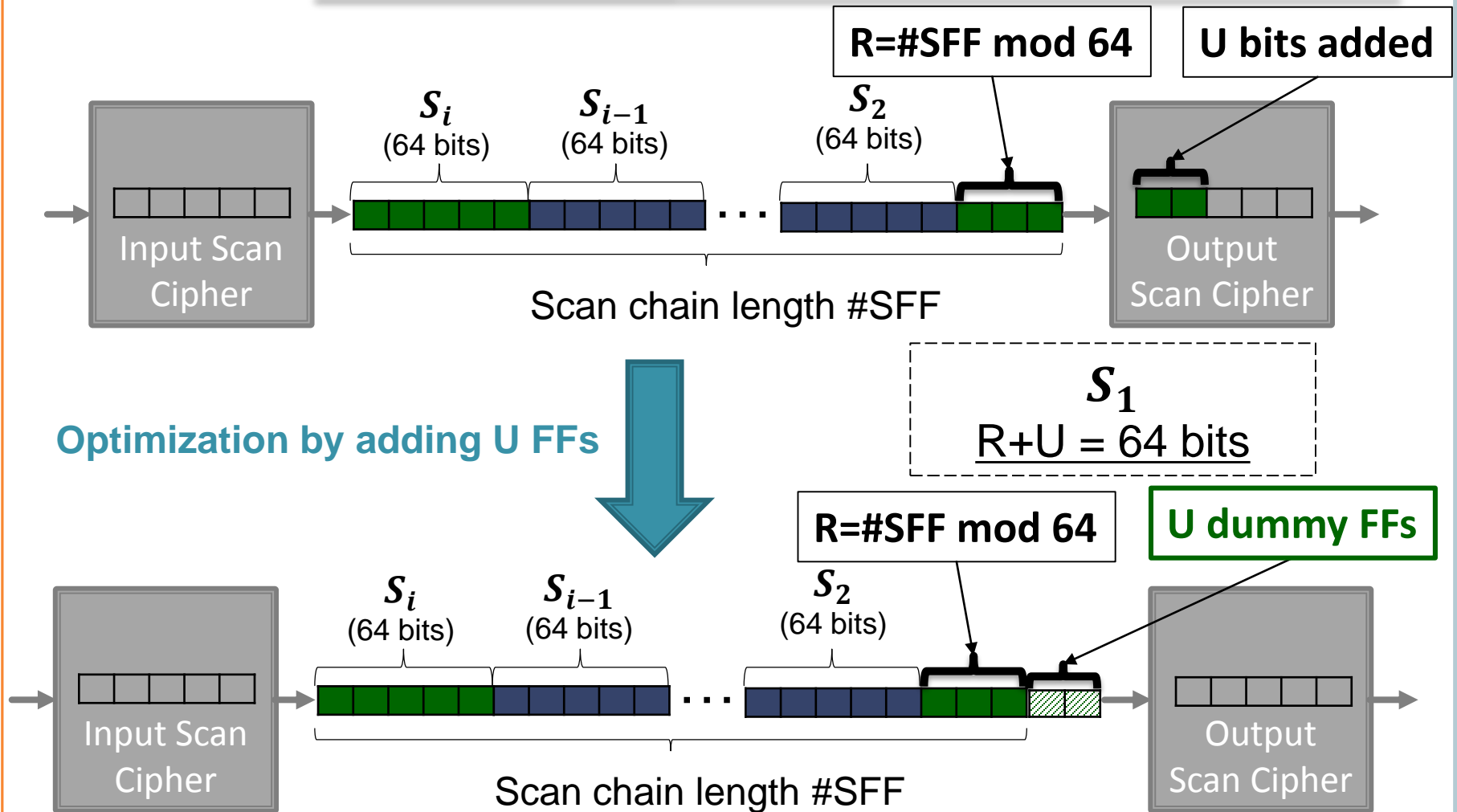


- No impact on original circuit
- ATPG tool* targets the reduction of number of patterns

*: Synopsys, TetraMax



ADDING DUMMY FF DOESN'T IMPACT TEST TIME



EXPERIMENTAL RESULTS & TEST COVERAGE



EXPERIMENTAL RESULTS

Original circuit		Triple-DES		Pipelined AES-128		Pipelined AES-256		RSA 1024		LEON3	
		Area (μm^2)	Test time (clock cycles)	Area (μm^2)	Test time (clock cycles)	Area (μm^2)	Test time (clock cycles)	Area (μm^2)	Test time (clock cycles)	Area (μm^2)	Test time* (clock cycles)
		187,494	687,101	367,926	1,944,877	669,193	4,559,845	468,415	39,405,239	1,902,095	11,612,051
Area (μm^2)	Overhead (%)	Overhead (%)	Overhead (%)	Overhead (%)	Overhead (%)	Overhead (%)	Overhead (%)	Overhead (%)	Overhead (%)		
Scan encryption with Stream cipher (without TRNG implementation)											
TRIVIUM	5,408.5 2	+2.88	+0.18	+1.47	+0.06	+0.81	+0.03	+1.15	+0.003	+0.28	+0.01
Scan encryption with Block cipher											
PRESENT-128	10,658.96	+5.74	+0.31	+2.92	+0.81	+1.61	+0.006	+2.30	+0.33	+0.57	+0.004



SCAN CIPHERS TESTED FOR FREE

	Triple-DES	Pipelined AES 128	Pipelined AES 256	RSA 1024	LEON 3
#Scan_FF	8 808	7 873	12 736	16 459	107 518
#Patterns	77	246	357	2 393	107
Original circuit Test coverage	100%	100%	100%	100%	70%
#Encryption (64-bits block size)	$138 \times 77 =$ 10 626	$124 \times 246 =$ 30 504	$199 \times 357 = 7$ 1 043	$258 \times 2393 =$ 617 394	$1680 \times 107 =$ 179 760
Scan ciphers Test Coverage	100%	100%	100%	100%	100%

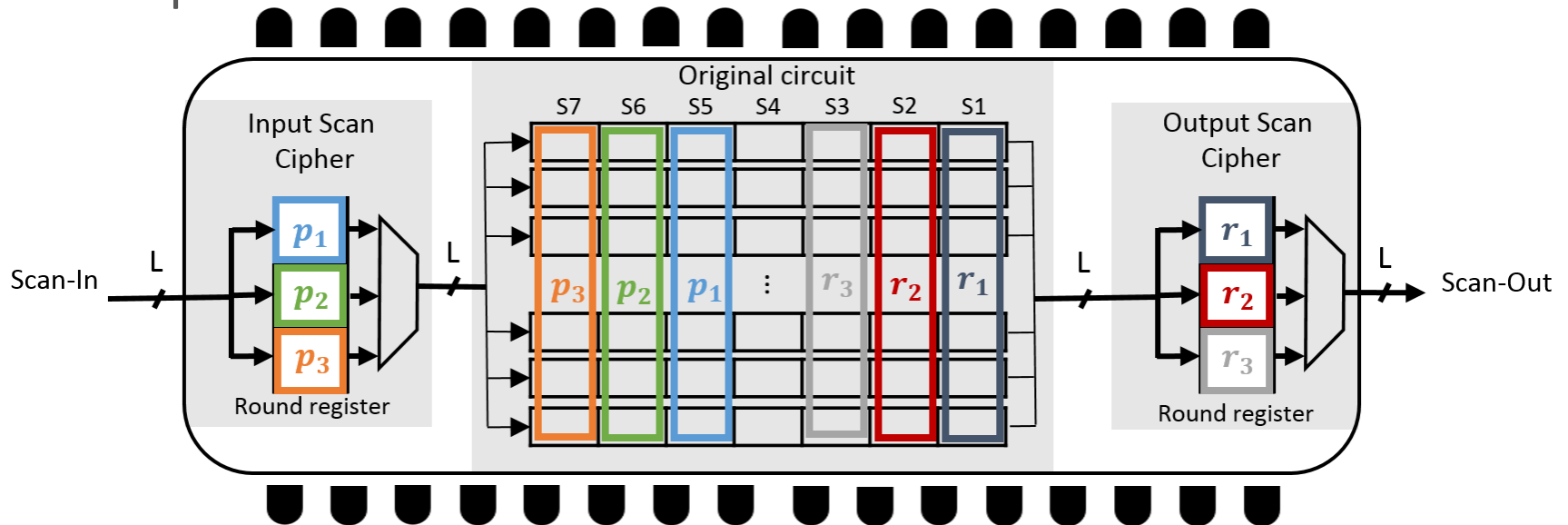


MULTIPLE SCAN CHAINS

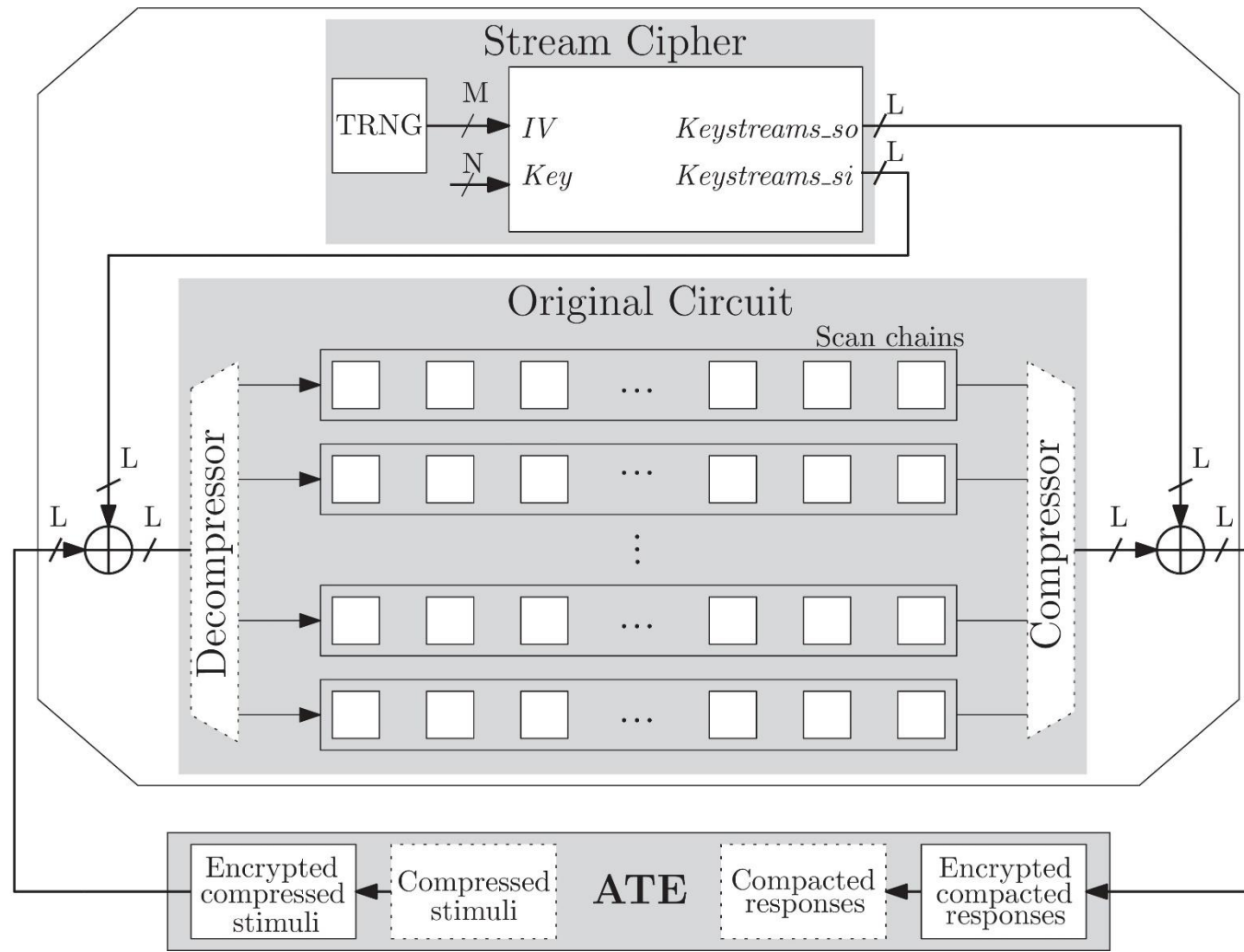


BLOCK-BASED SOLUTION APPLIED ON MULTIPLE SCAN CHAINS

- Objective 1: apply the solution on multiple scan chains
- Objective 2: no impact on test time due to the scan encryption
- Solution: encrypt several scan slices at high frequency (>10 MHz) during the basic scan operations at low frequency (10 MHz)
- Example:



STREAM-BASED SOLUTION APPLIED ON MULTIPLE SCAN CHAINS



EXPERIMENTAL RESULTS

Scan encryption	Stream cipher (without TRNG implementation)			Block Cipher		
Ciphers	TRIVIUM			PRESENT-128		
# scan chains	Area (μm^2)	Power (μW)	Freq. (MHz)	Area (μm^2)	Power (μW)	Freq. (MHz)
1	5,408.52	34.01	10	10,658.96	73.80	10
2	5,553.60	34.02	10	11,877.84	147.6	20
4	5,851.04	34.16	10	11,652.16	221.3	30
8	6,453.20	34.55	10	11,532.56	368.9	50
16	7,615.92	35.14	10	11,520.08	664.0	90
32	9,999.60	36.37	10	11,215.36	1,254.2	170
64				11,183.12	2,434.7	330



NANOFOCUSED X-RAY BEAM TO REPROGRAM SECURE CIRCUITS

24/05/2018

BLEUET Pierre
CLEDIERE Jessy
MAINGAULT Laurent
RAINARD Jean-Luc
TUCOULOU Rémi

CEA/LETI

ESRF

Anceau Stéphanie
(Research Engineer at Leti ITSEF)
Stephanie.anceau@cea.fr

leti



1 -Context and laboratory experiment setup

2 -Synchrotron experiment setup

3 -Attacks experiments results on :

- Flash (350 nm)
- Flash NOR (110-90 nm)
- SRAM (45nm)

4 -Physical Phenomenon explanation

- Transistors N and P
- Flash memory cell

5 -Conclusion

1 -Context and laboratory experiment setup

2 -Synchrotron experiment setup

3 -Attacks experiments results on :

- Flash (350 nm)
- Flash NOR (110-90 nm)
- SRAM (45nm)

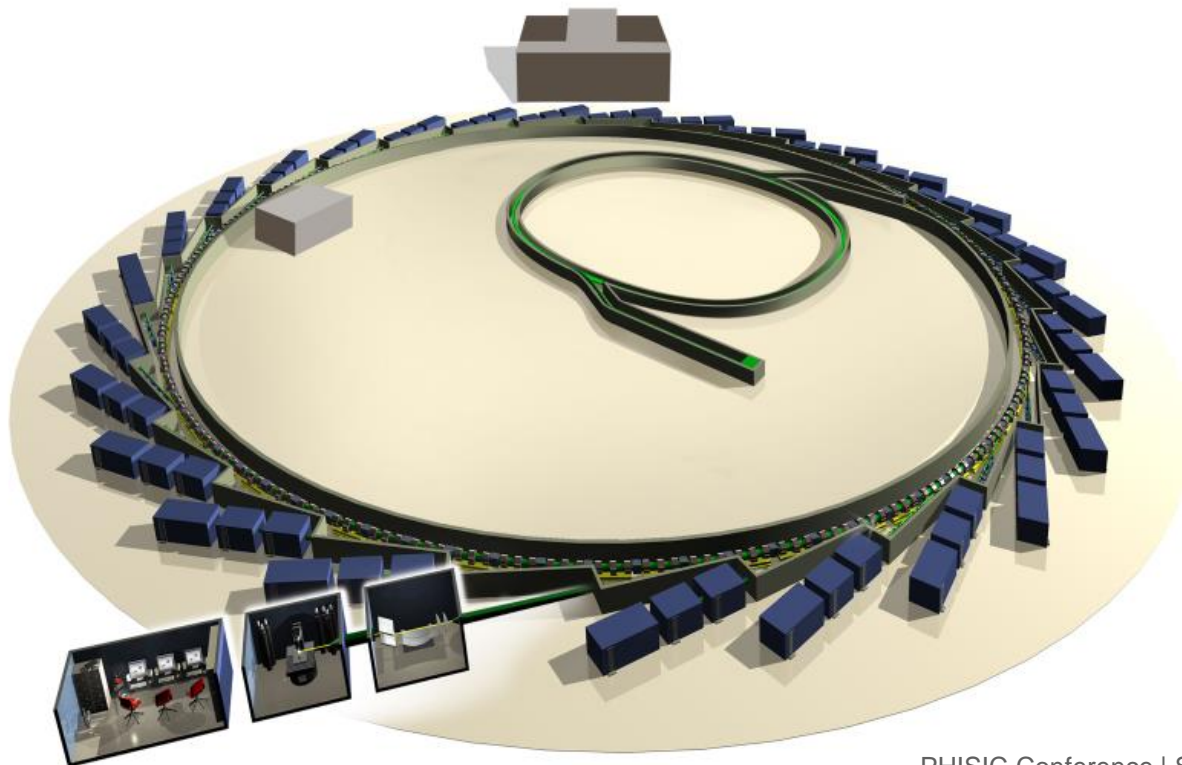
4 -Physical Phenomenon explanation

- Transistors N and P
- Flash memory cell

5 -Conclusion

LET'S SPEAK ABOUT X-RAYS

- Ionizing radiations are often mentioned in literature, in failure analysis and space literature
- A new method of perturbation?
- We propose using a nanofocused X-ray beam of a synchrotron



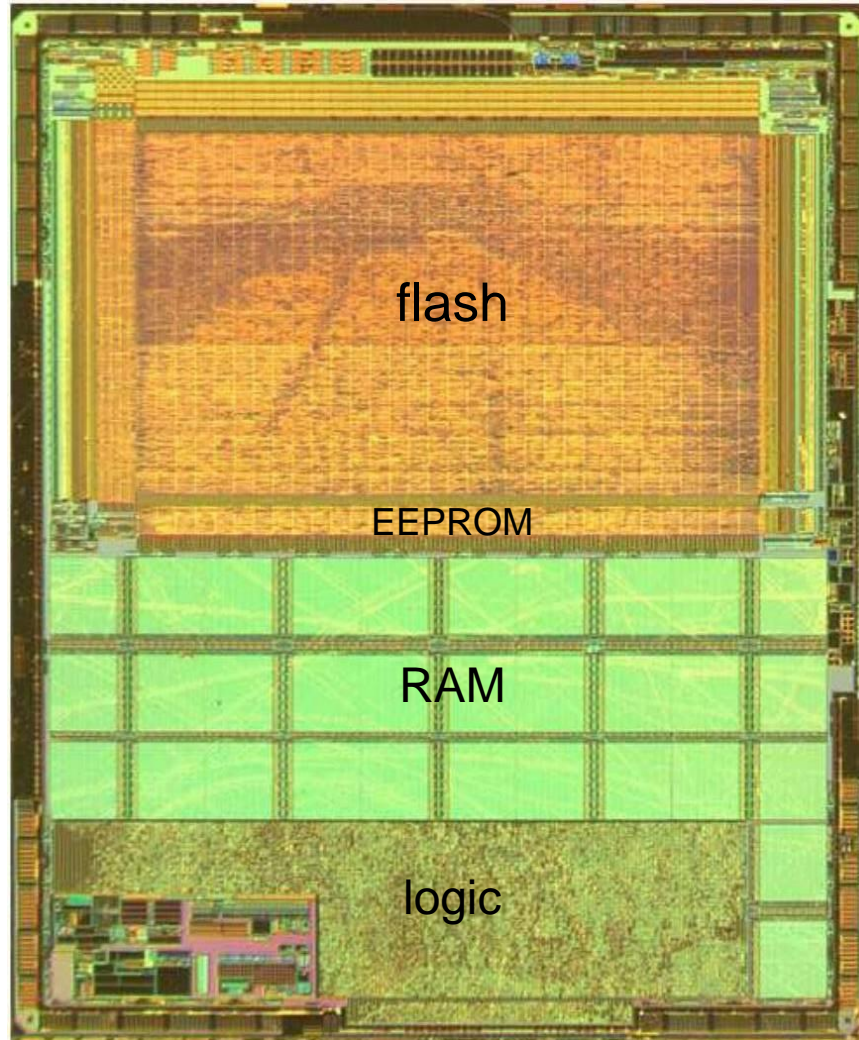
HOW DID WE GET TO A SYNCHROTRON ?

...after doing some preliminary tests on simple equipment



A fairly old circuit (350 nm) but useful to investigate new attacks

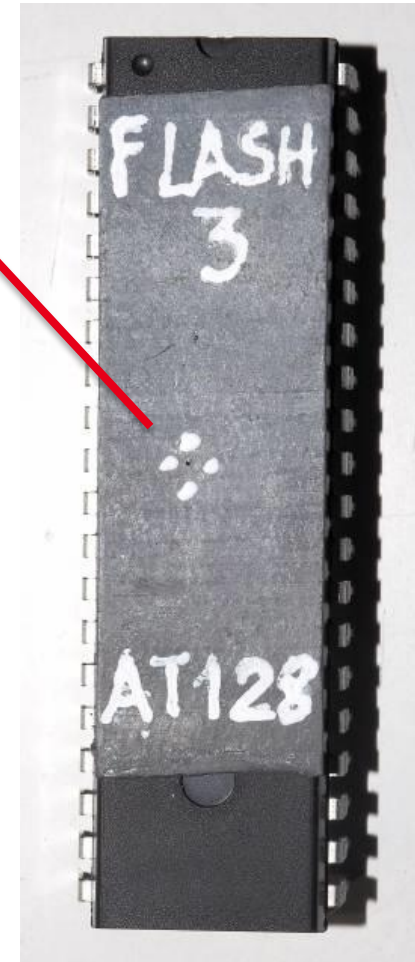
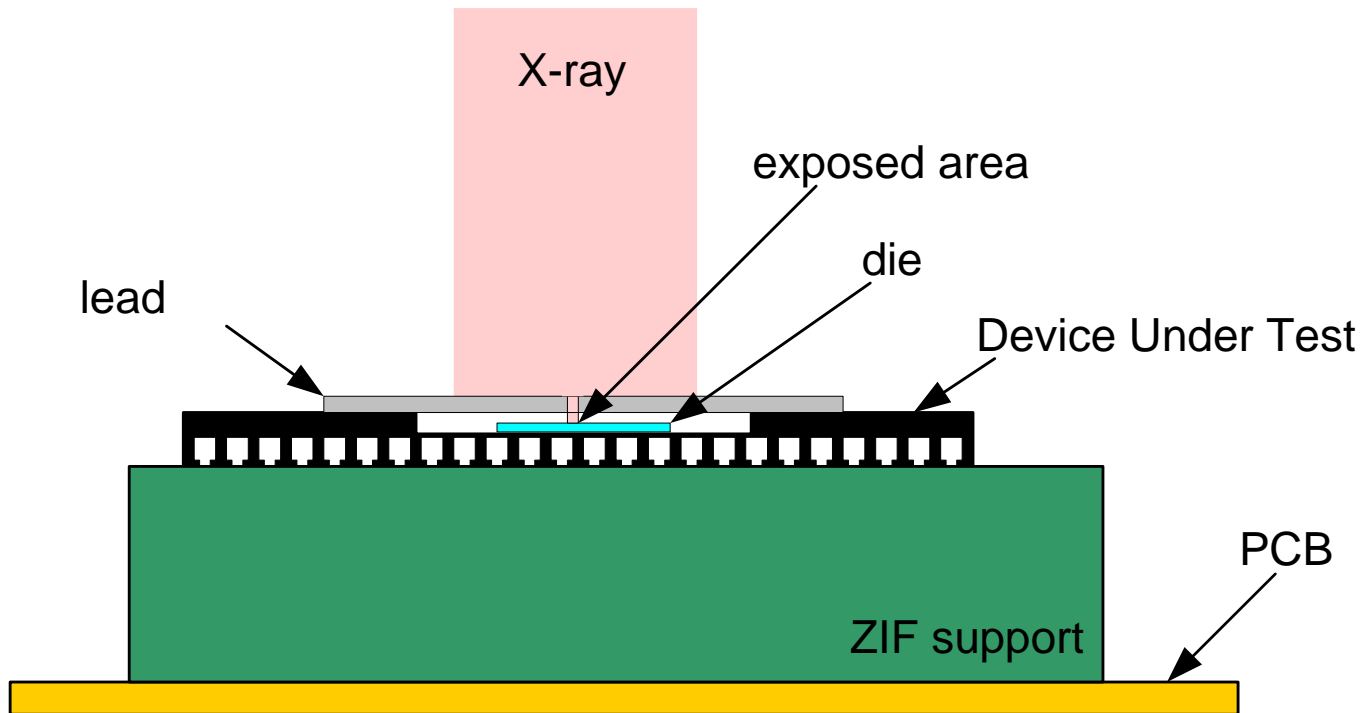
ATMEGA LAYOUT



500 μm

WITH SOME BASIC FOCUSING...

...a hole in a lead sheet

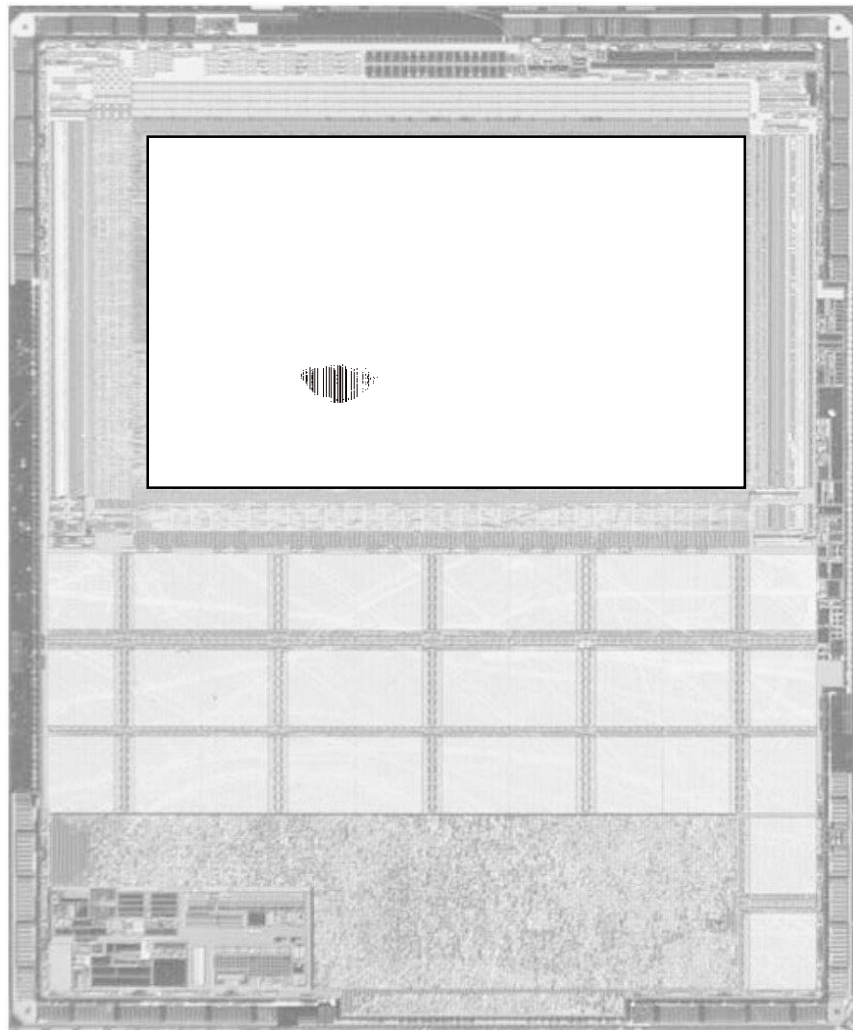


ATMEGA + LEAD SHEET AND HOLE



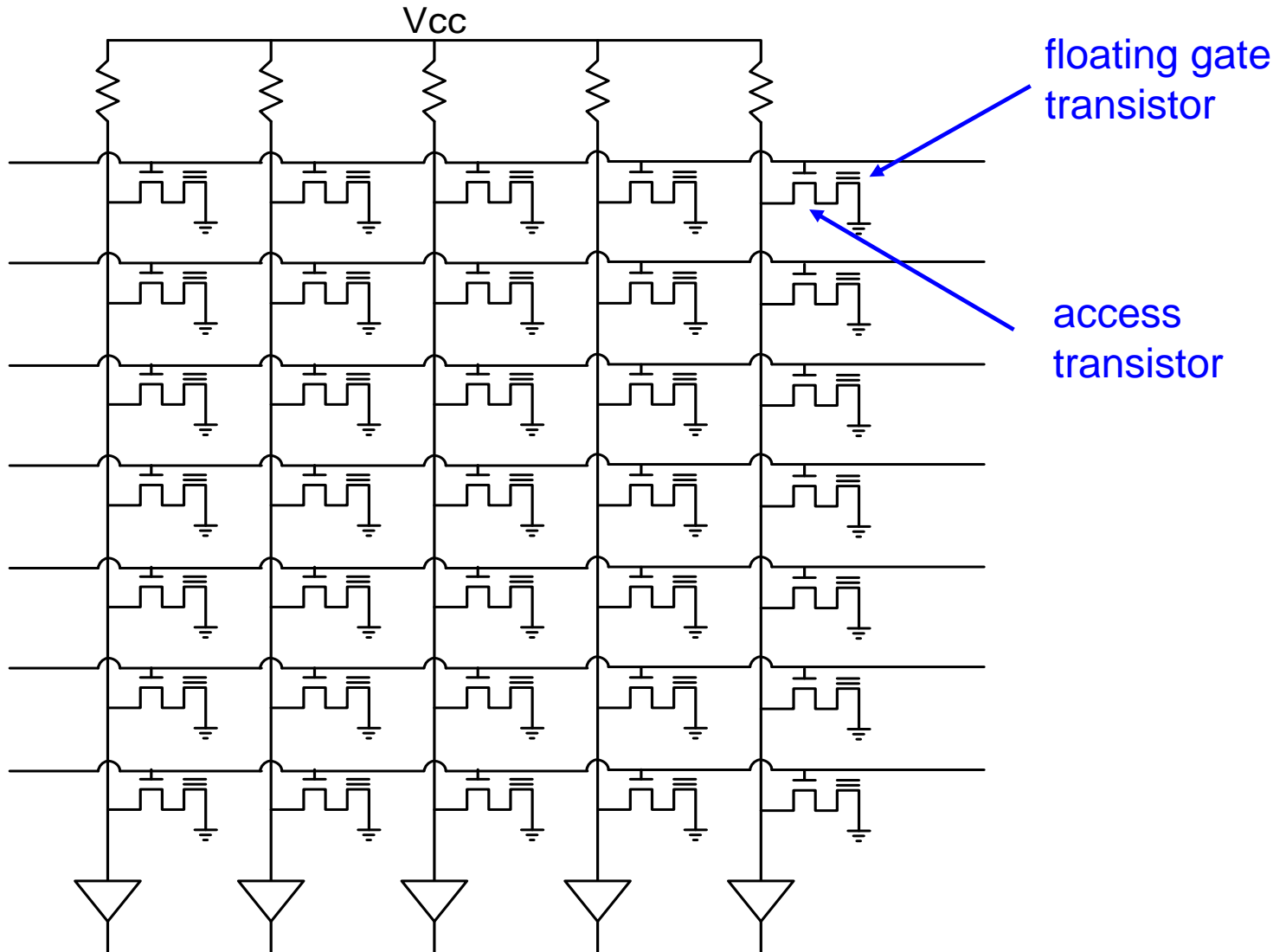
we fill FLASH memory
with value `0x55`

FIRST FAULTS OBTAINED AFTER 210 SECONDS OF EXPOSURE

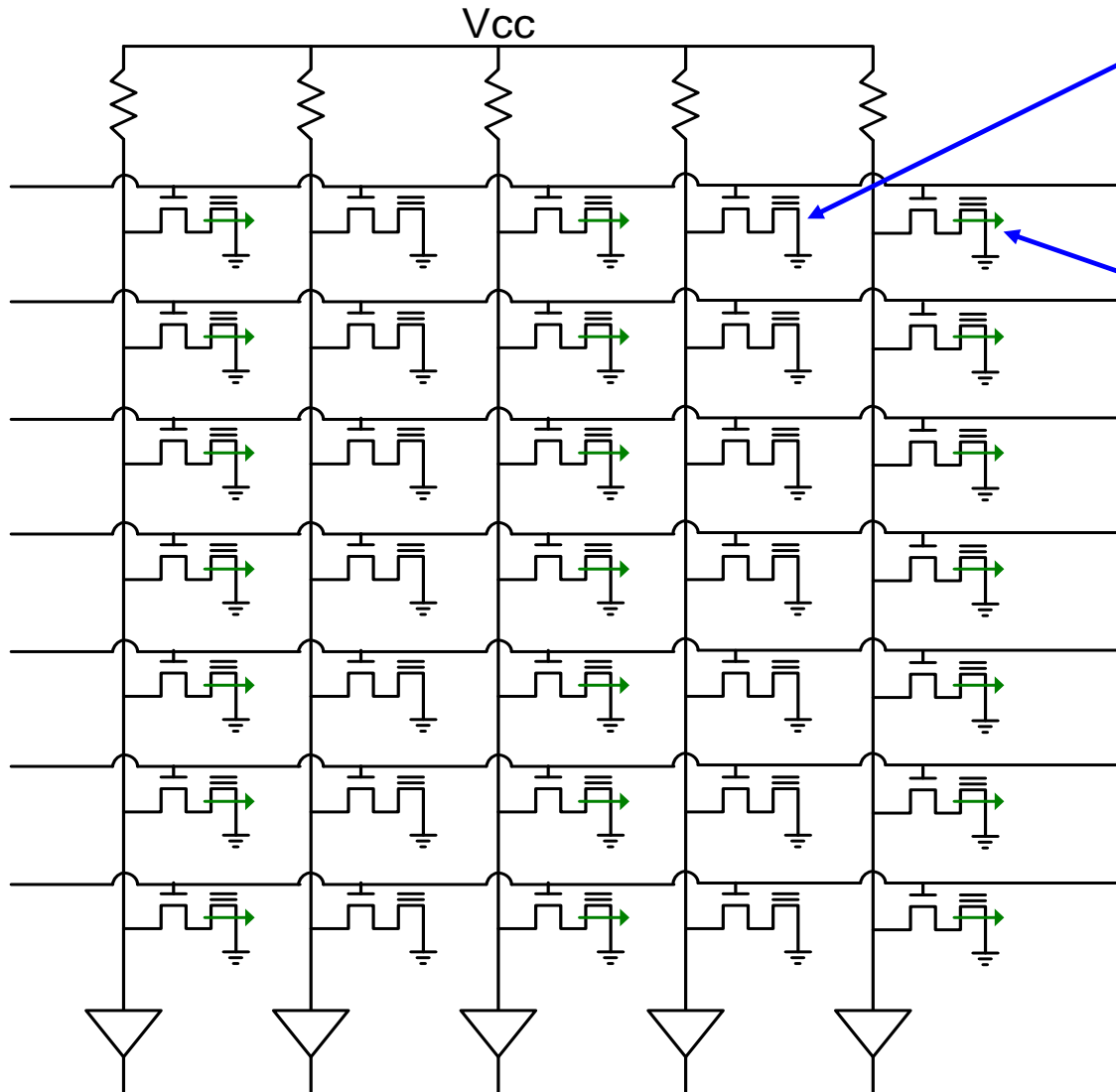


red: "1" to "0" corruption

WHAT HAPPENED?



DATA IS STORED IN THE FLOATING GATES



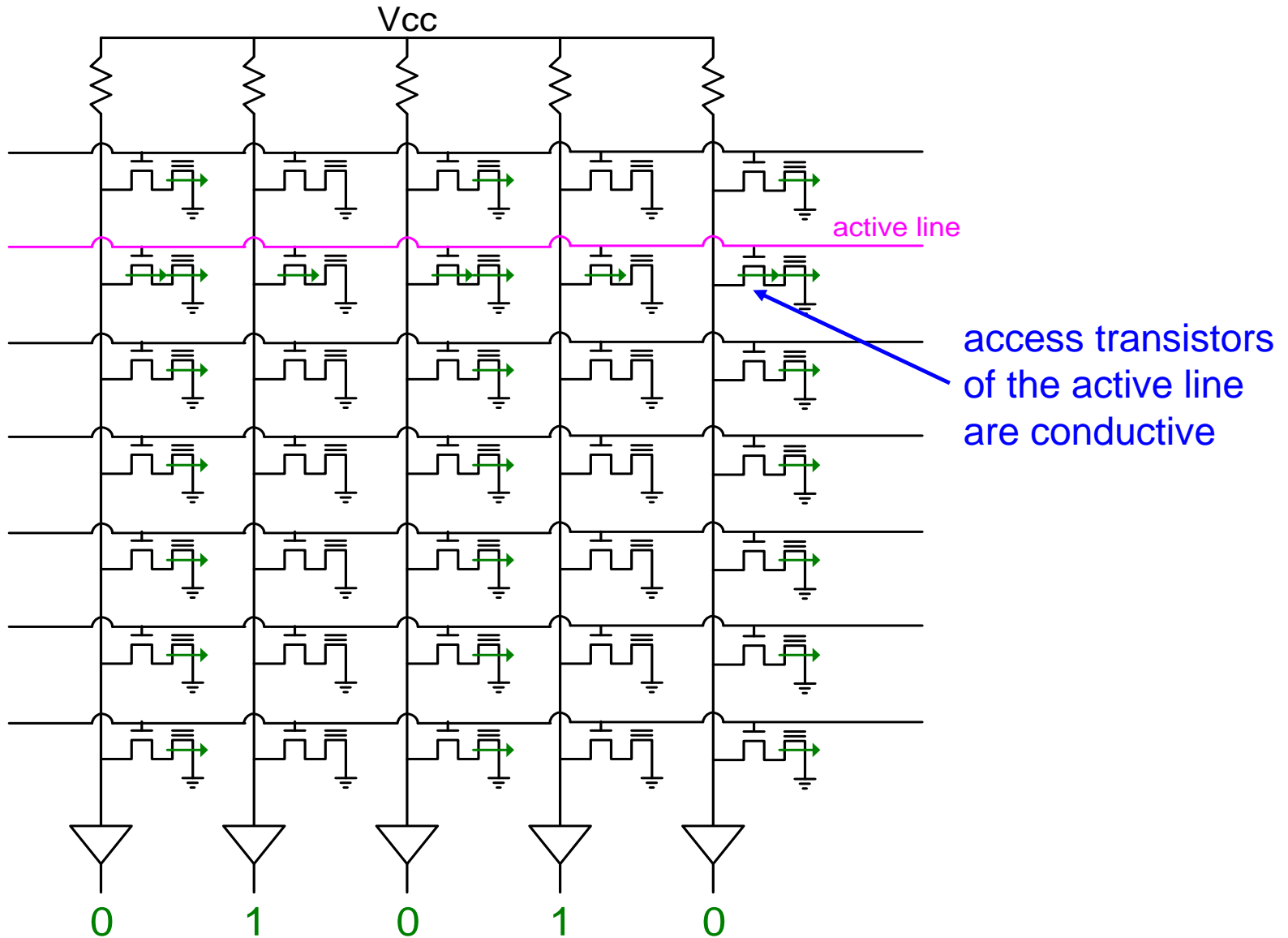
charge in the floating gate:

- transistor is blocked
- Value 1 is stored

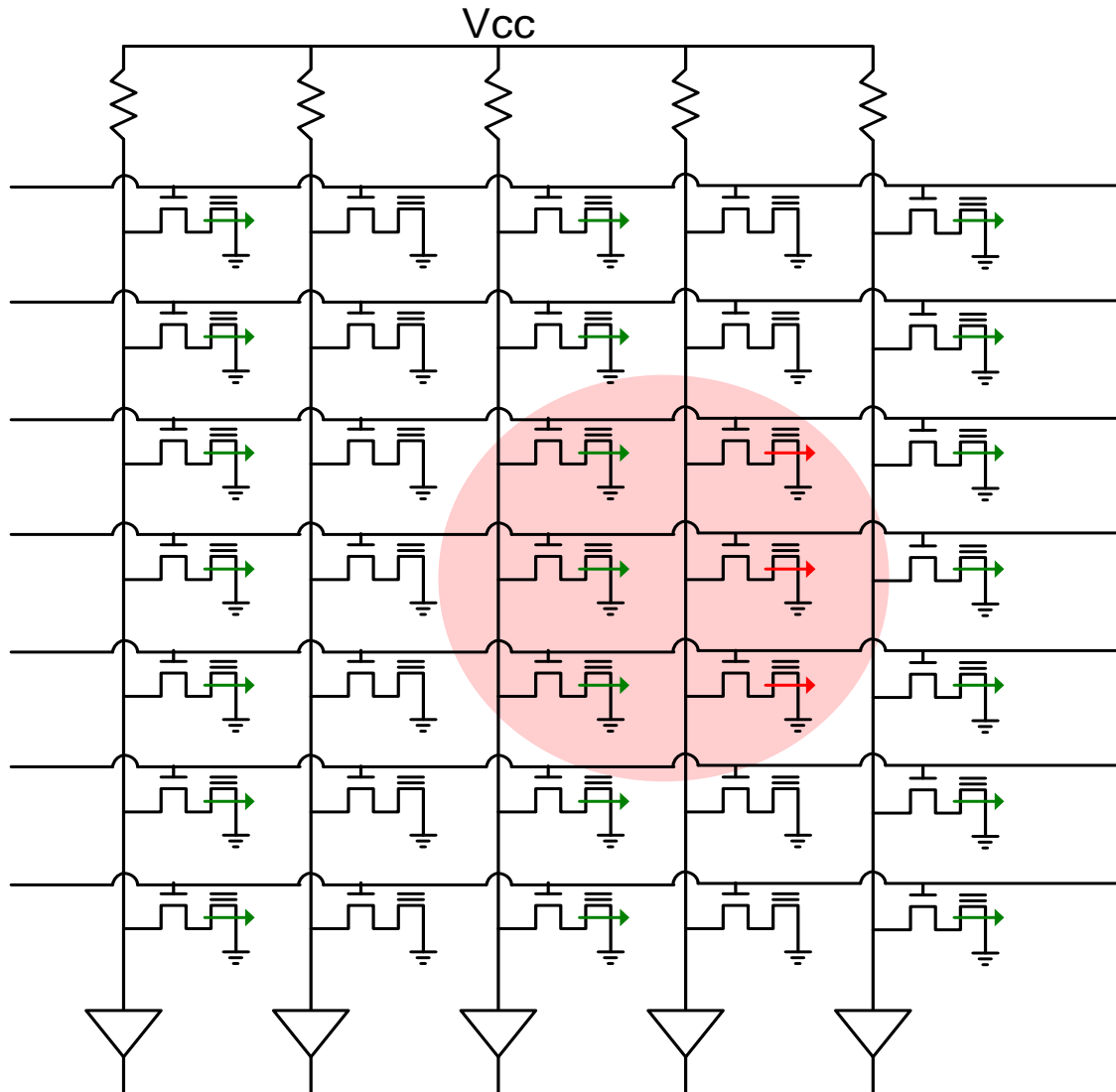
no charge in the floating gate:

- transistor is conductive
- Value 0 is stored

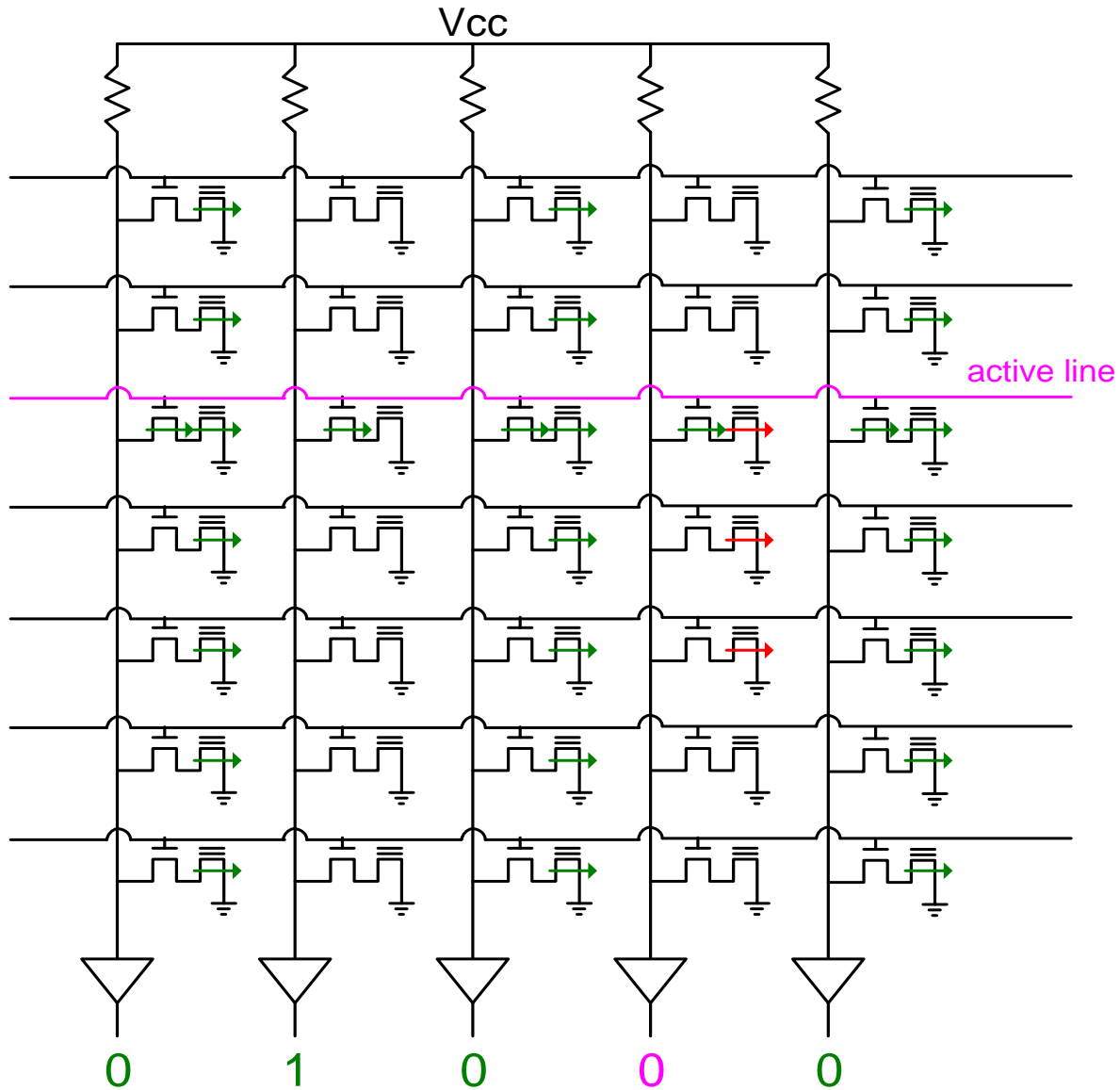
ACCESS TO THE FLOATING GATES



X-RAY EXPOSURE: WE DISCHARGE THE FLOATING GATES



ACCESS TO THE DATA



- We empty floating gates of carriers
 ➔ **we could modify (1 to 0) FLASH and EEPROM**



- We modify transistors semi-permanently
 NMOS are made conductive
 it is **reversible** with a heat treatment (150°C, 1 hour)

The last result applied to logic area of the circuit :

➔ **we could reconfigure circuits : circuit edit**

- These effects are described in the space literature and are very interesting for our activity



let's focus x-rays down to the nano-scale **to target a single transistor!**

1 -Context and laboratory experiment setup

2 -Synchrotron experiment setup

3 -Attacks experiments results on :

- Flash (350 nm)
- Flash NOR (110-90 nm)
- SRAM (45nm)

4 -Physical Phenomenon explanation

- Transistors N and P
- Flash memory cell

5 -Conclusion

GRENOBLE, FRANCE

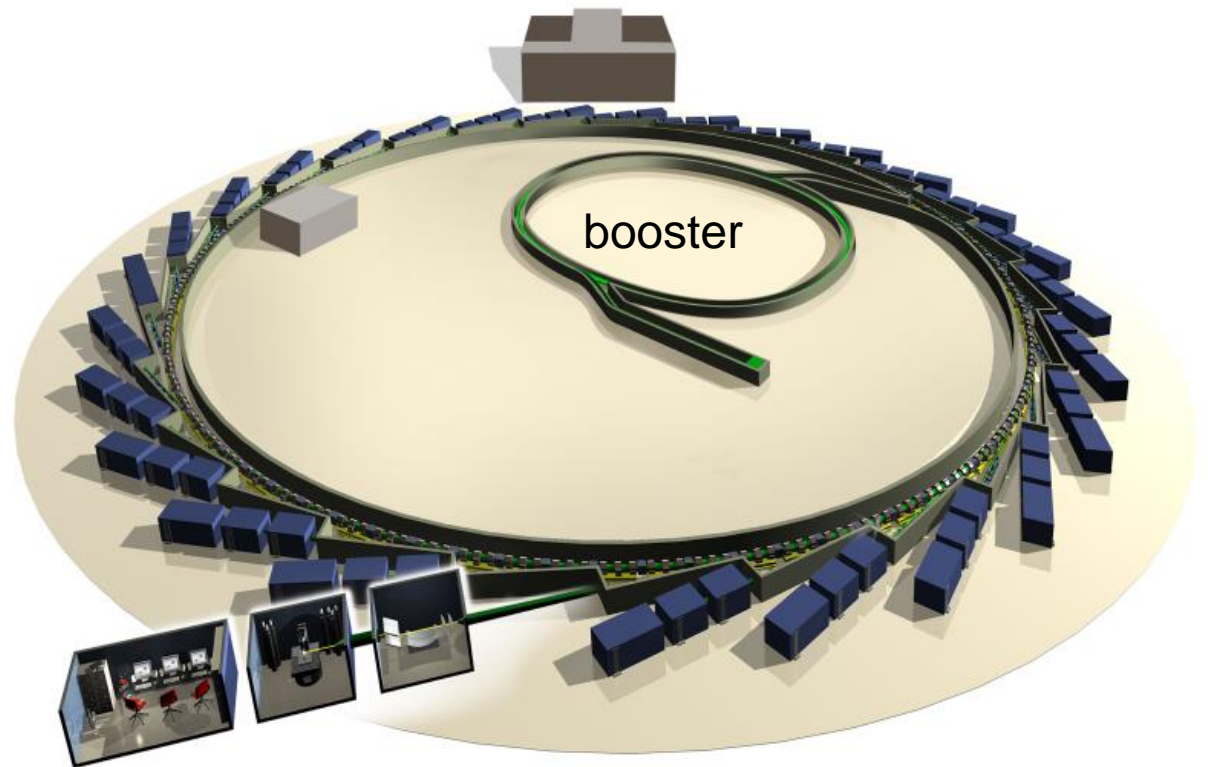
Léti ITSEF

**European Synchrotron Radiation Facility
(ESRF)**



500 m

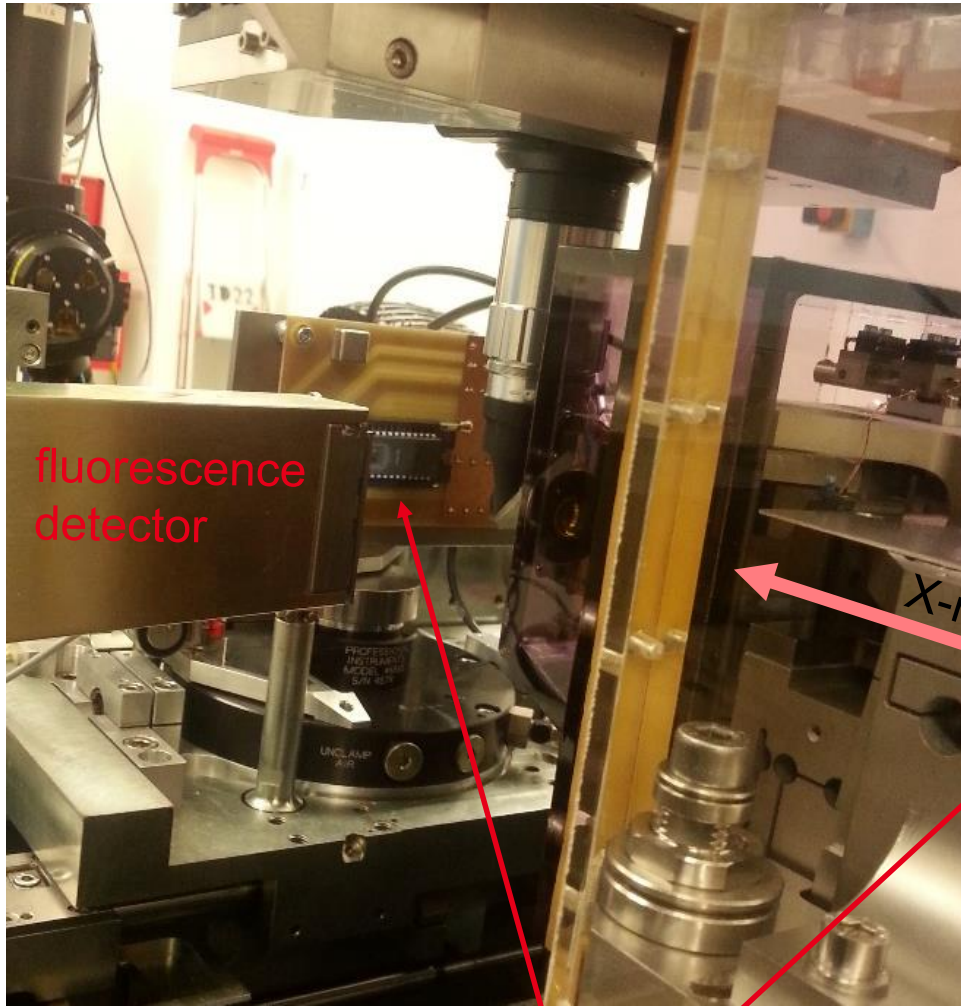
SYNCHROTRON EXPERIMENT SETUP



INSIDE THE DONUT



FOCUSING TO THE NANO SCALE : 60 NM XRAY SPOT



long focal length optic



ATMEGA at the focal point of X-ray optic

NANOFOCUSED X-RAY BEAM TO REPROGRAM SECURE CIRCUITS

1 -Context and laboratory experiment setup

2 -Synchrotron experiment setup

3 -Attacks experiments results on :

- Flash (350 nm)
- Flash NOR (110-90 nm)
- SRAM (45nm)

4 -Physical Phenomenon explanation

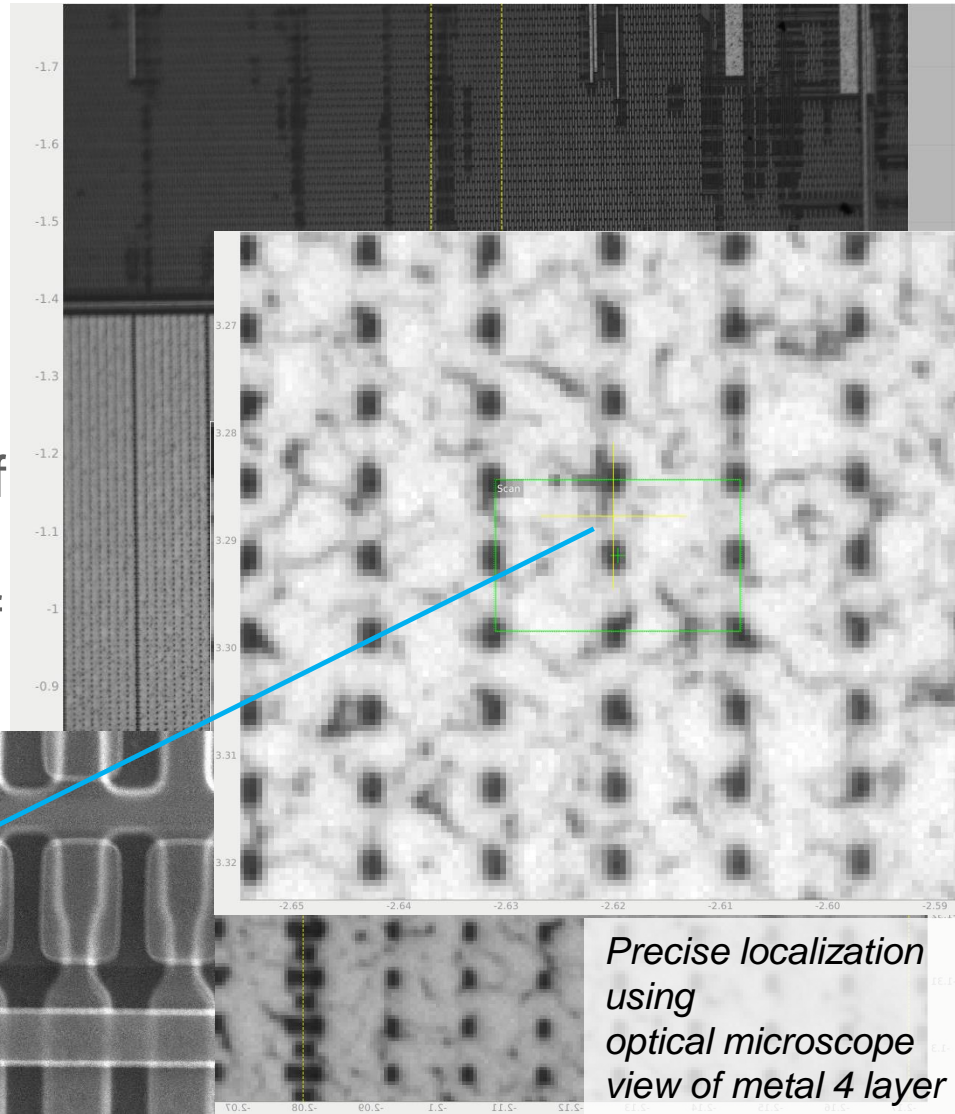
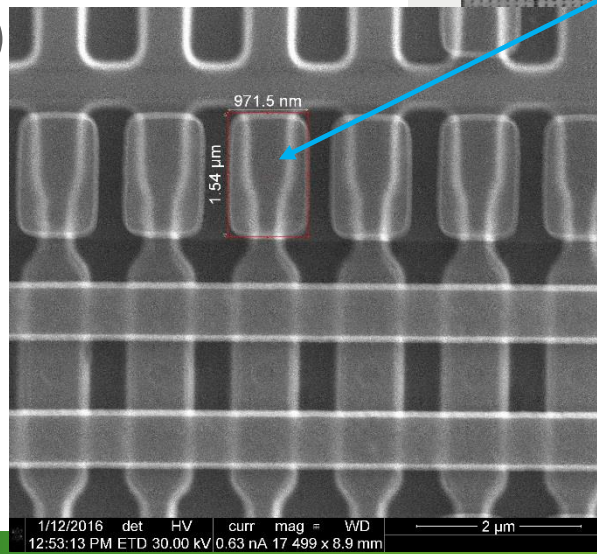
- Transistors N and P
- Flash memory cell

5 -Conclusion

ATTACKS RESULTS ON FLASH

Localization of each **Flash** memory cells using :

- Preliminary high resolution Fluorescence mappings of (tungsten vias (precision of ~50 nm) (destructive)
- Optical microscope view of metal 4 layer (non destructive)



Permanent erase of the memory cell : 0 -> 1

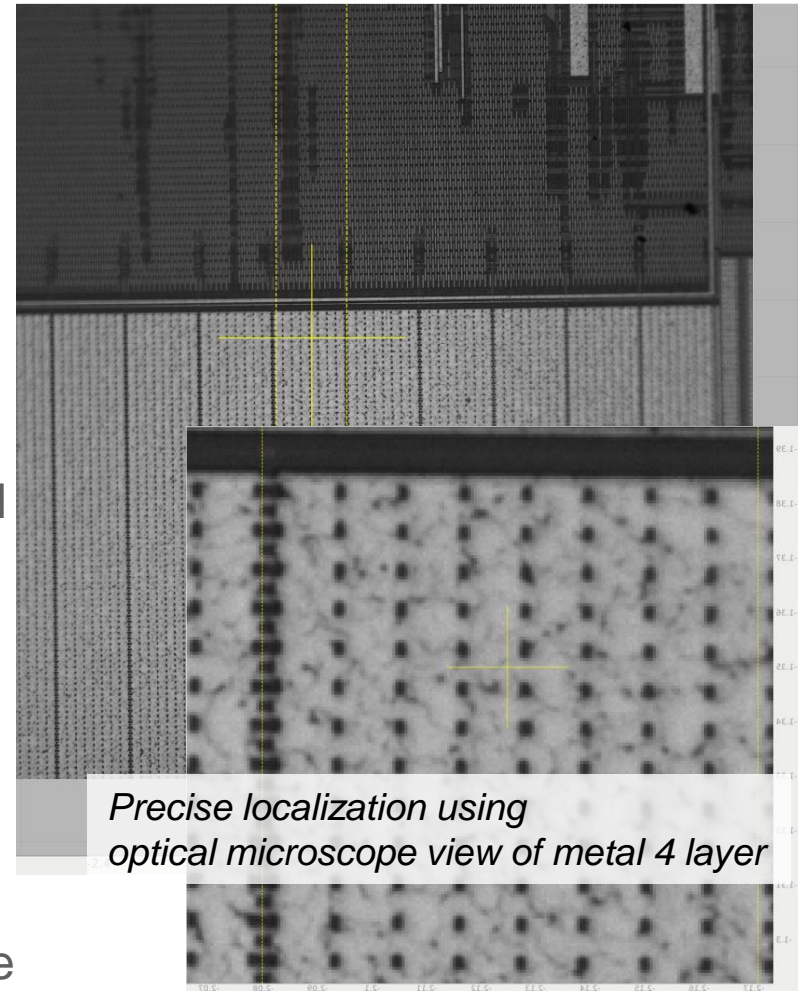
Attack Experiment

- Local x-ray attack of a single **Flash** memory cell :
- Instructions are stored in the flash memory cells.
- One instruction has been modified with single bit permanent erase with the Xray focalized beam: BRanch if Not Equal op code become BRanch if Equal op code and accept 9999 erroneous PIN (rejecting the genuine PIN).



After the attack the chip accepts every authentication code.

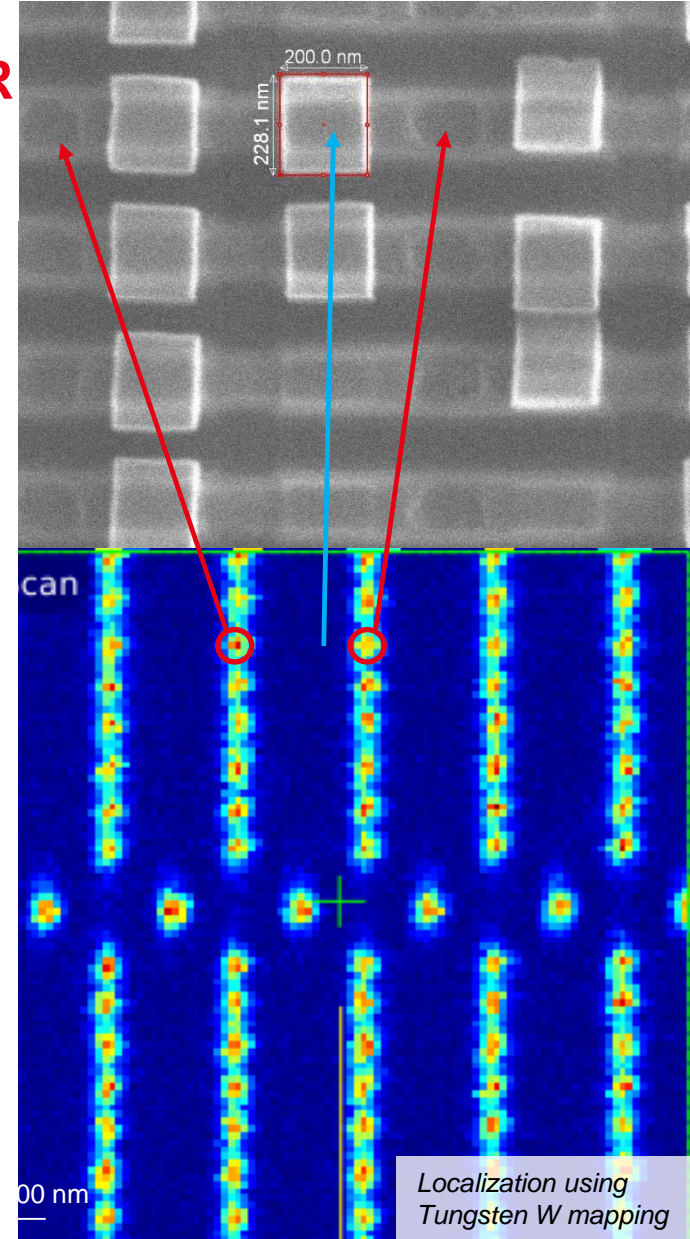
- The precise address of the single bit can be retrieved before the attack
- The attack can be done during or before or after a simple reading of the block memory



Permanent erase of the memory cell: 0 -> 1

Experiment

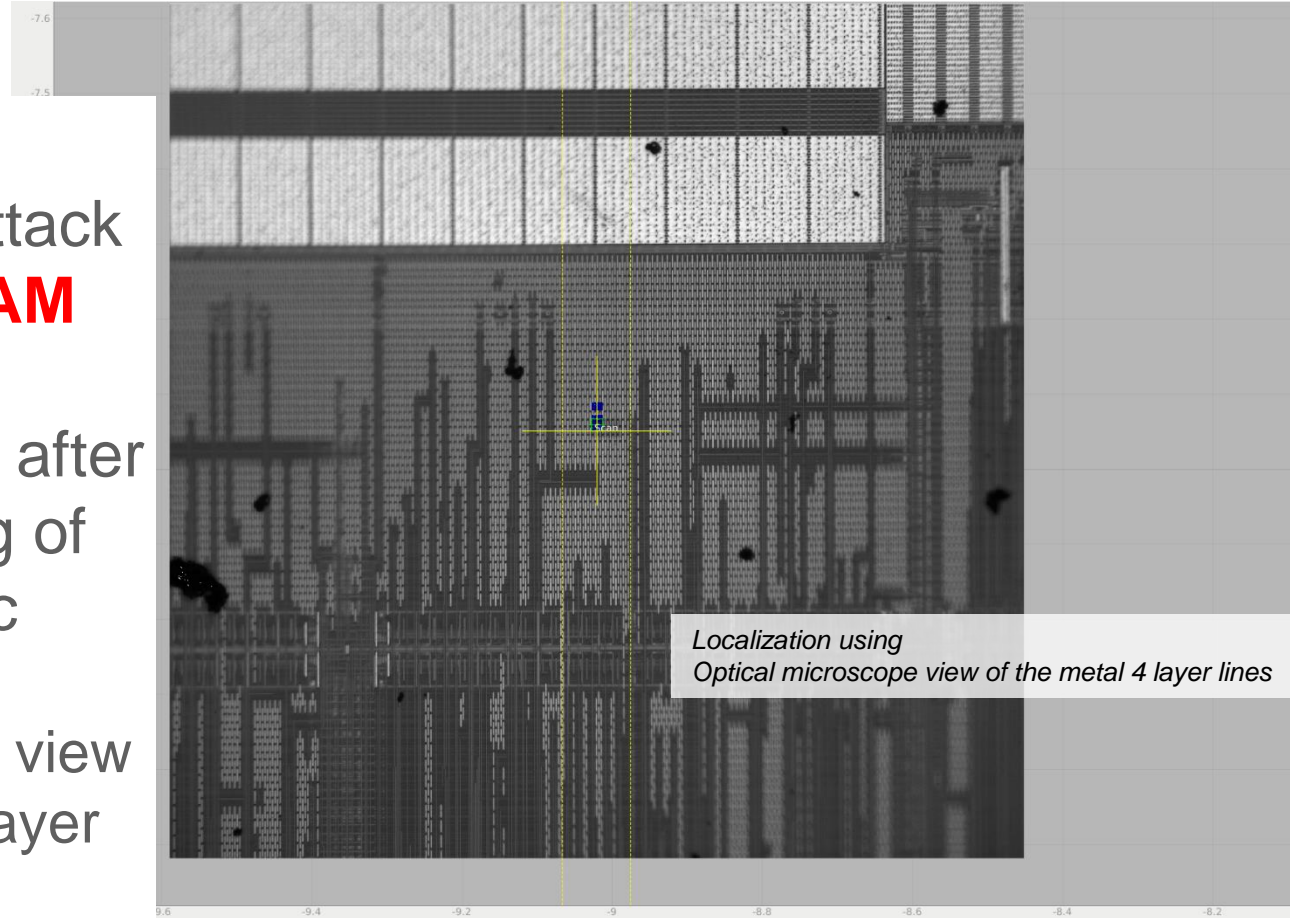
- Local x-ray attack of a single **Flash Nor** memory cell before or after a simple reading of the memory bloc
- The precise address of the single bit can be retrieved before the attack
- Localization of each memory cell using high resolution fluorescence mappings of tungsten vias (precision of ~ 50 nm)
- The technology node of the device is ≥ 90 nm



Permanent erase of the memory cell :1 -> 0

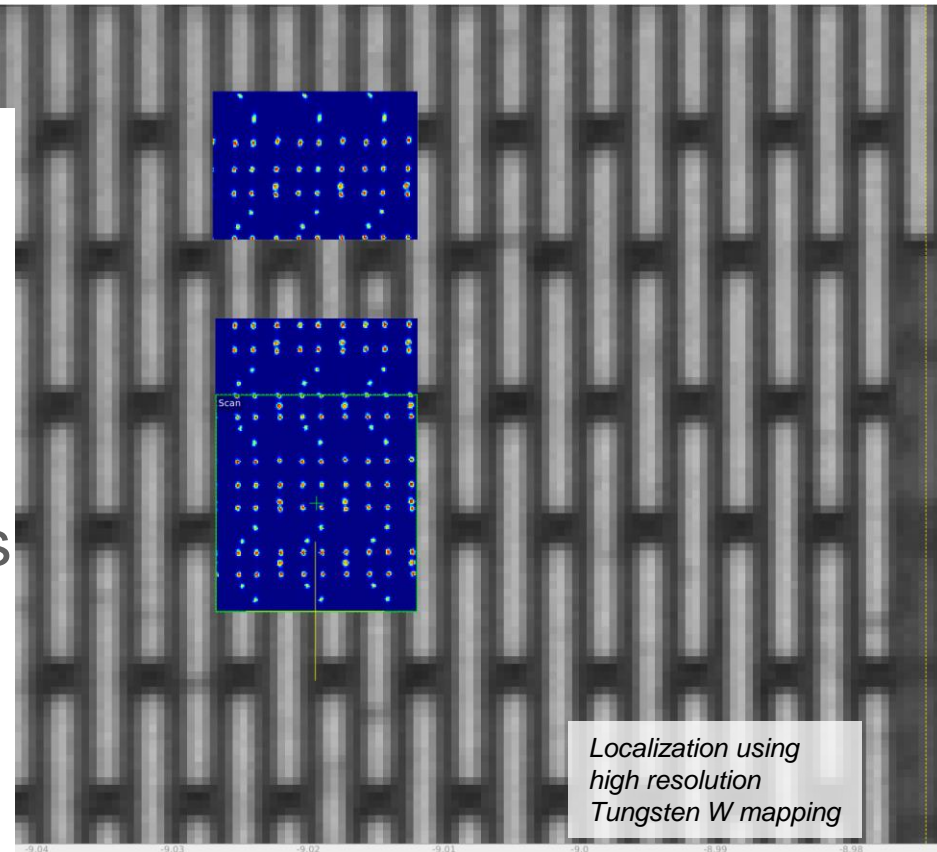
- **Experiment**

- Local x-ray attack of a single **RAM** memory cell
→ before or after a simple reading of the memory bloc
- Optical microscope view of metal 4 layer



- **Experiment**

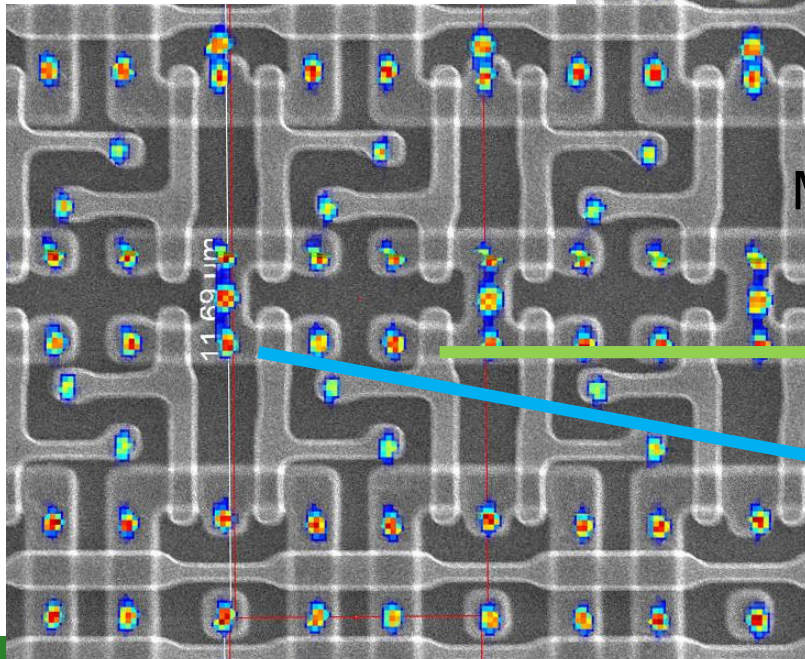
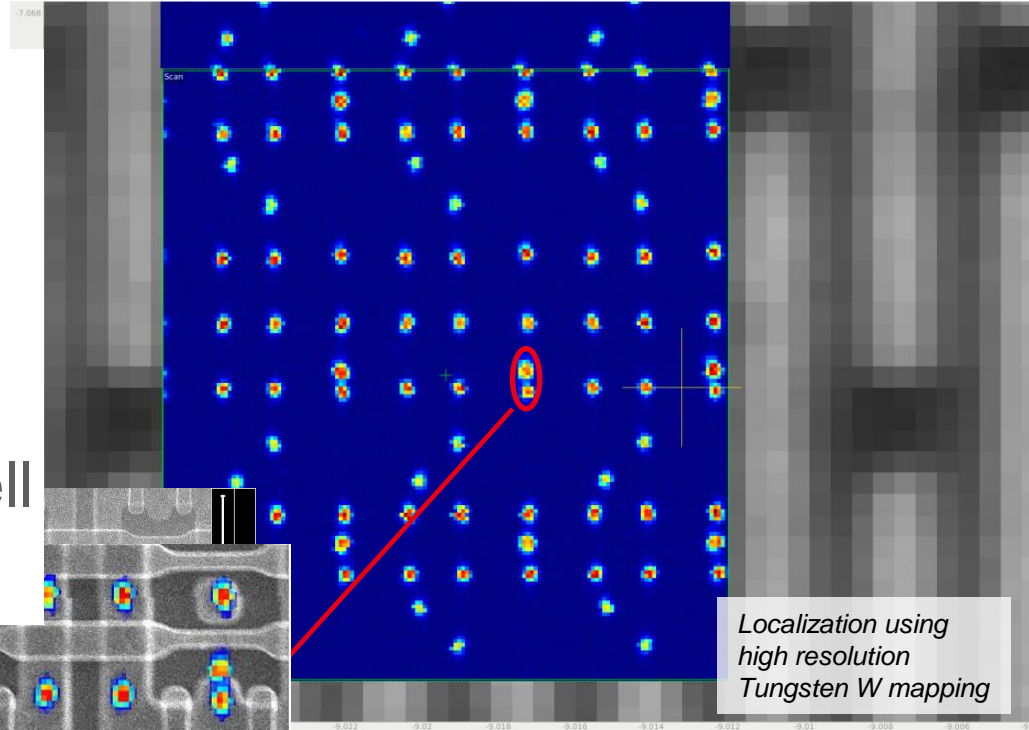
- Local x-ray attack of a single **RAM** memory cell :
- Preliminary high resolution fluorescence mappings of tungsten vias (precision of ~50 nm)
 - **Non destructive**
- The attack can be done during or before the device functioning



ATTACKS EXPERIMENTS ON SRAM

Experiment

- Local x-ray attack of a single **RAM** memory cell
- The precise address of the single bit can be retrieved Each memory cell can be set or reset



MOS-N=> Permanent conductor

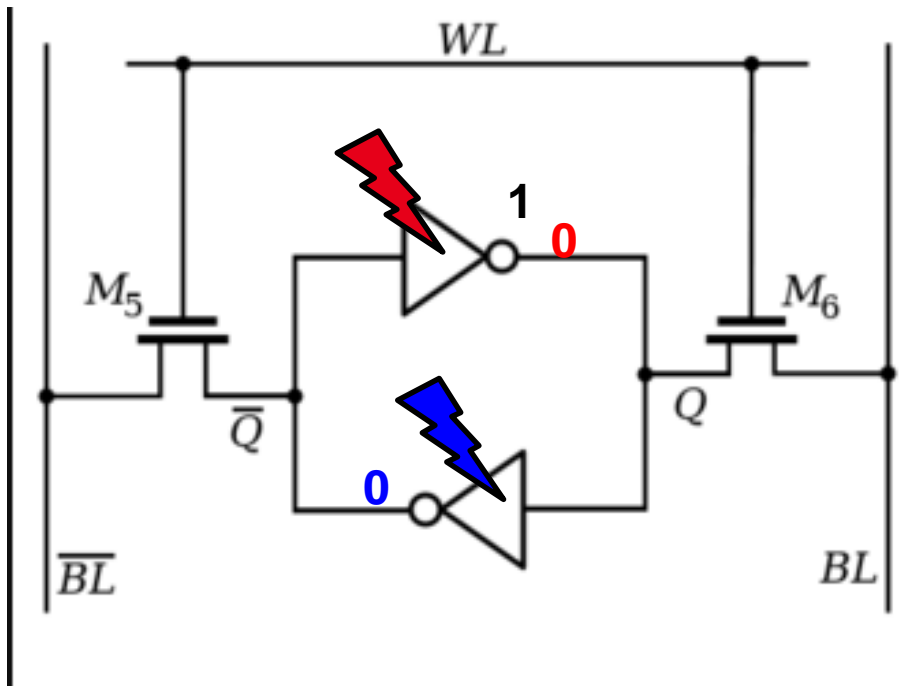
Set = output stick to 1

Reset = output stick to 0

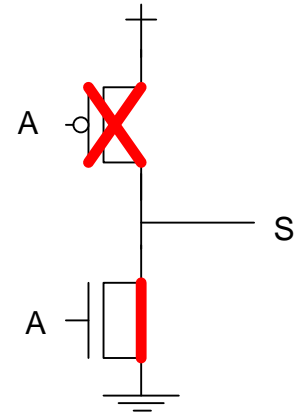
ATTACKS EXPERIMENTS ON SRAM

N MOS TRANSISTOR => CONDUCTOR

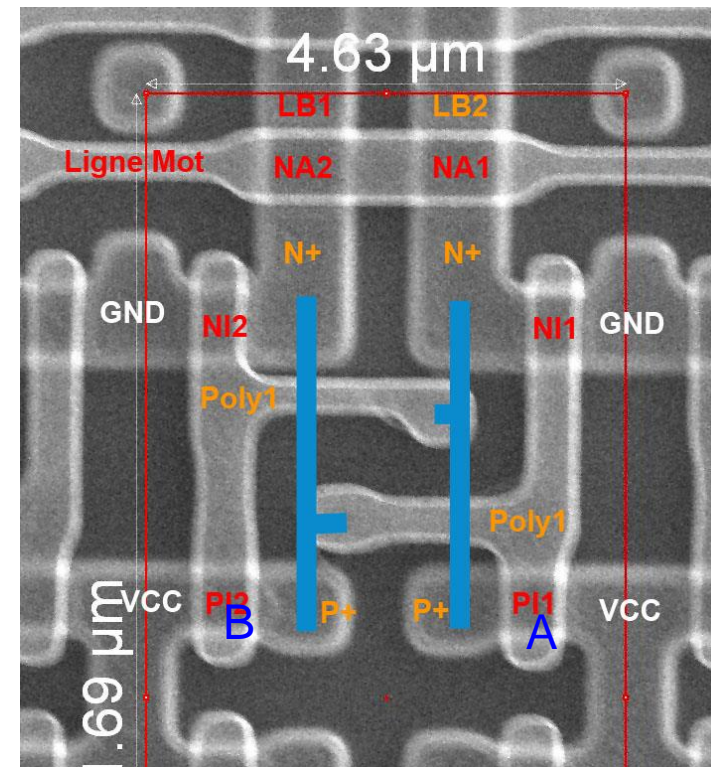
- Picture : Active areas N and P and Polysilicon lines
- Metal M1 (blue)



PMOS IS
Weak
compared to
the NMOS

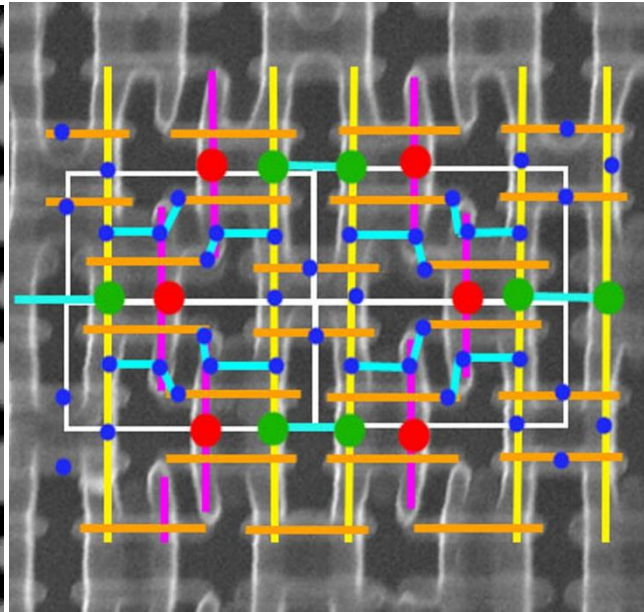
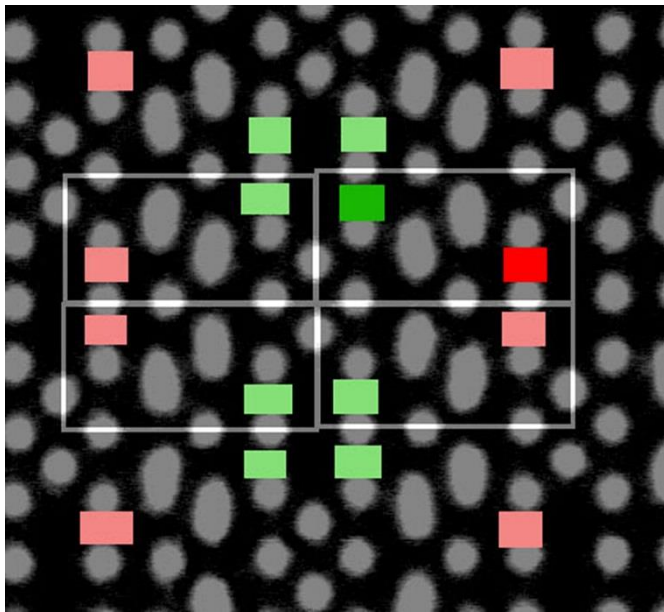


Inverseur



SRAM 55 NM MICROCONTROLLER RESULTS

Modification of a single transistor state in an SRAM memory cell
 => Memory cell stuck at 1 or 0 depending on which transistor is irradiated.



yellow : diff N
 violet : diff P
 orange : poly
 Light blue : M1
 Dark blue : contact
 red : contact alim-Vcc
 green : contact alim-GND
 white : SRAM cell

MEB Observation at contact under metal level 1 (left picture) at polysilicon level (right picture) of a 6 transistors memory cell.

The irradiated transistors is green for NMOS (1 -> 0) and red for NMOS (0->1).

One single transistor irradiated gives a single bit faulted result on the SRAM memory bloc.

1 -Context and laboratory experiment setup

2 -Synchrotron experiment setup

3 -Attacks experiments results on :

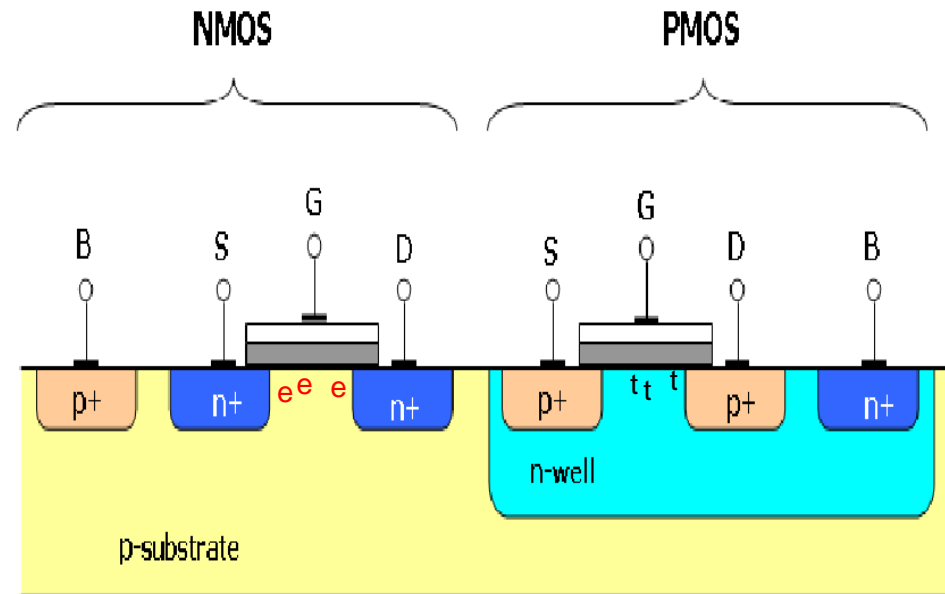
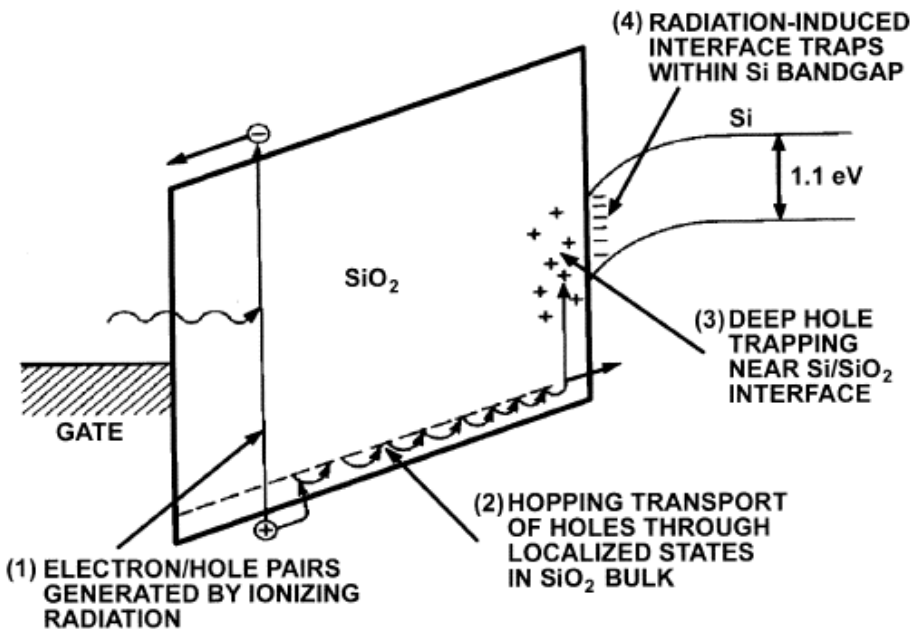
- Flash (350 nm)
- Flash NOR (110-90 nm)
- SRAM (45nm)

4 -Physical Phenomenon explanation

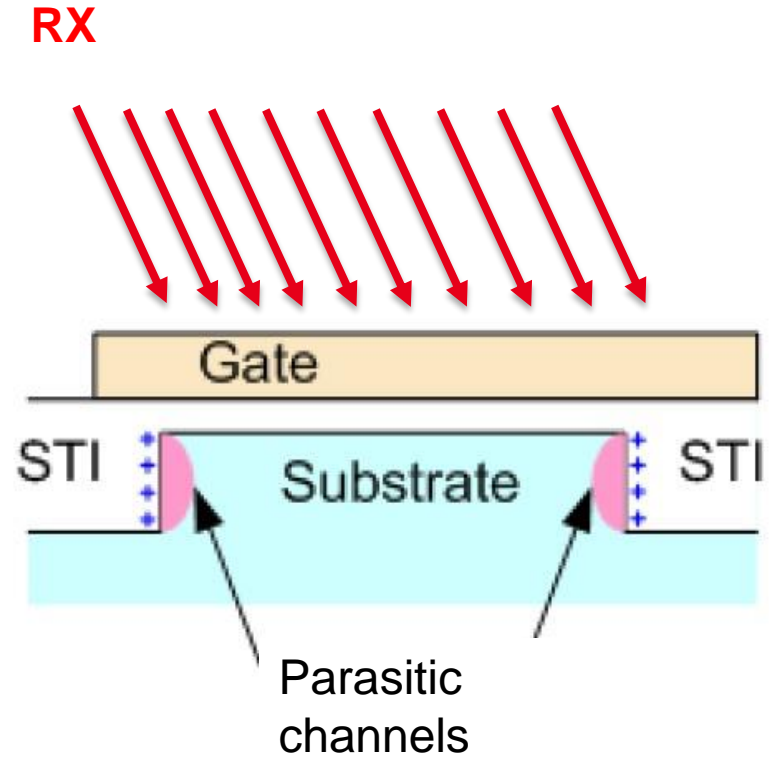
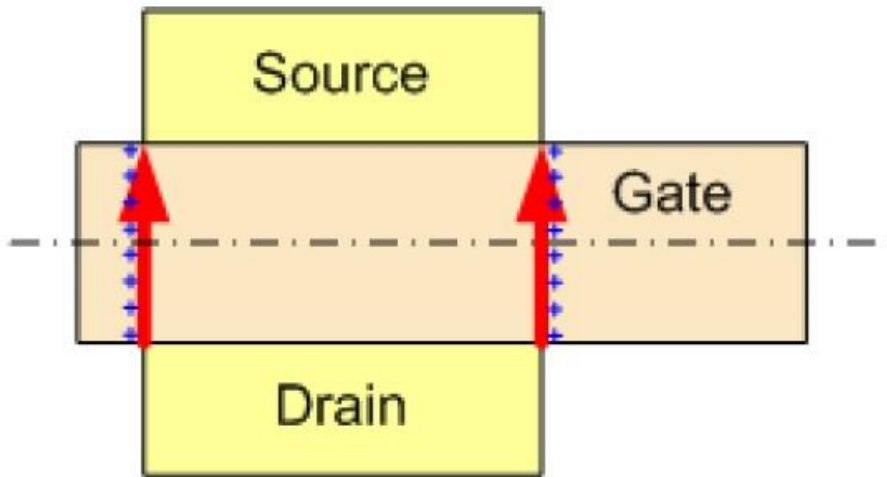
- Transistors N and P
- Flash memory cell

5 -Conclusion

- Possibility to modify a single bit at any given memory address !
- Physics behind:
 - Carriers in the oxide decrease NMOS transistor V_T
 - Carriers in the oxide Increase PMOS transistor V_T



WHAT HAPPENS ?



NMOS transistor

PHYSICAL PHENOMENON ON FLASH MEMORY CELL

- Possibility to modify a single bit at any given memory address !
- $\sim 10^{10}$ ph/s on 1800 nm^2 to change the state of a single bit within 1s

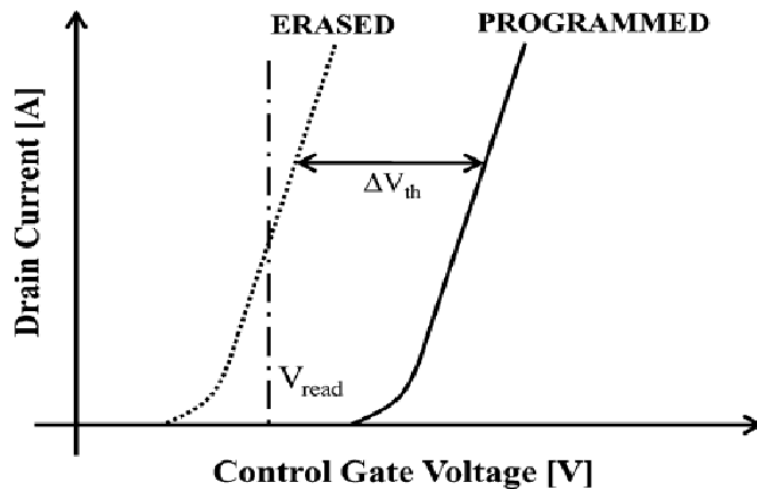
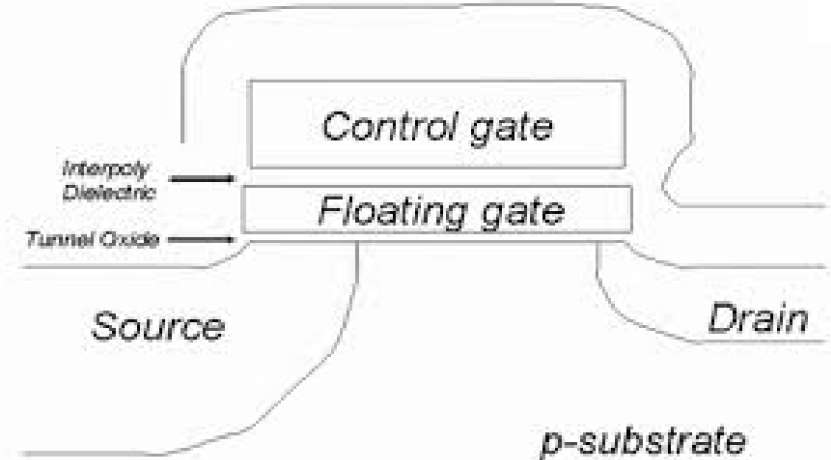


Fig. 2. Typical characteristic curves for drain current as a function of gate voltage for transistors in cases of erased and programmed floating gates.

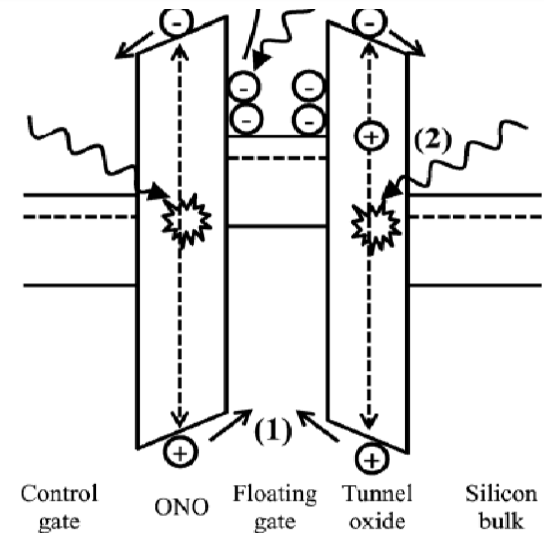


Fig. 8. Basic mechanisms for total ionizing dose induced threshold voltage shifts in floating gate cells. After [23].

NANOFOCUSED X-RAY BEAM TO REPROGRAM SECURE CIRCUITS

1 -Context and laboratory experiment setup

2 -Synchrotron experiment setup

3 -Attacks experiments results on :

- Flash (350 nm)
- Flash NOR (110-90 nm)
- SRAM (45nm)

4 -Physical Phenomenon explanation

- Transistors N and P
- Flash memory cell

5 -Conclusion

OBTAINED RESULTS

- Fluorescence mapping allows **accurate positioning at the transistor level**
- FLASH and EEPROM can be **modified (1 to 0) at the bit level** : code of a circuit can be changed



CHES Best paper award 2017 article : S. Anceau, P. Bleuët, J. Clédière, L. Maingault, J.L. Rainard, R. Tucoulou :

“Nanofocused X-Ray Beam To Reprogram Secure Circuits”, CHES 2017, Taiwan

- **Single SRAM cells** can be semi-permanently **stuck at 0 or 1** by corrupting NMOS transistors

PERSPECTIVES :

- Logic can be modified at the transistor level : **circuit edit**
this could be used to:
 - change the behavior of the circuit
 - remove hardware countermeasures...
- **No need to open the package of the die**

CONCLUSION ON NANOFOCUSED X-RAY

- A new technique to attack circuits and to perform circuit-editing
- “Extreme” resolution with accurate positioning with the use of fluorescence mapping
- Tool with a difficult access, **but not so expensive!**
- Experiments are still ongoing.

- Nanofocused X-rays could be compared to laser perturbation or to **Focused Ion Beam** (invasive attack, circuit edit)
- Implementation is like a laser setup but no sample preparation is required (package opening, thinning...). But very small spot (60 nm or less): reverse engineering is required!
- **Effects are like invasive attacks but totally non invasive!**
FIB: modification of metal layers of the circuit
X-rays: modification of the transistors of the circuit

Explore and develop this new technique of integrated circuit modification by using an X Rays focalized beam at ESRF

- Precise localization of the transistor to attack with the help of fluorescence scan and GDS layout of the circuit

- Modification of single PMOS transistors

- Adaptation of the technique to more aggressive technology

- Exploration of the perturbation possibilities with laboratory X Rays beam (without synchrotron)

During the PhD the candidate will have access to beam line shift at ESRF

END

Anceau Stéphanie
(Research Engineer at Leti ITSEF)
Stephanie.anceau@cea.fr

leti



Raphaël
ABELÉ

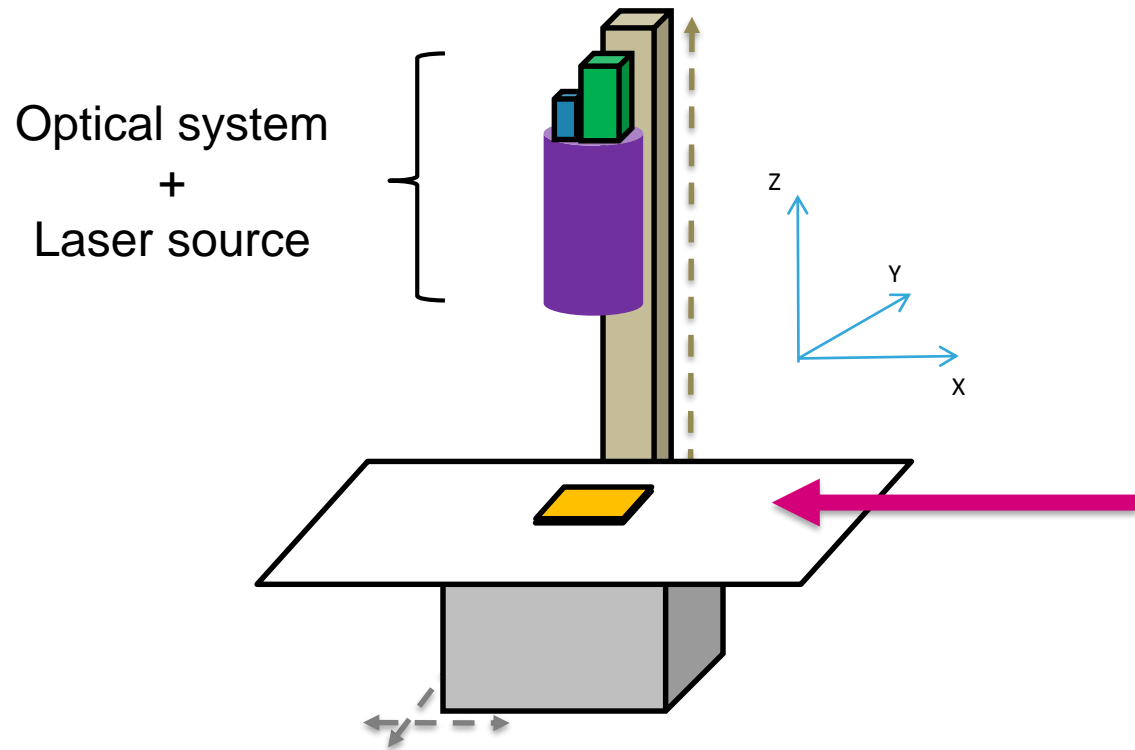
Autofocus in infrared microscopy

PHISIC 2018

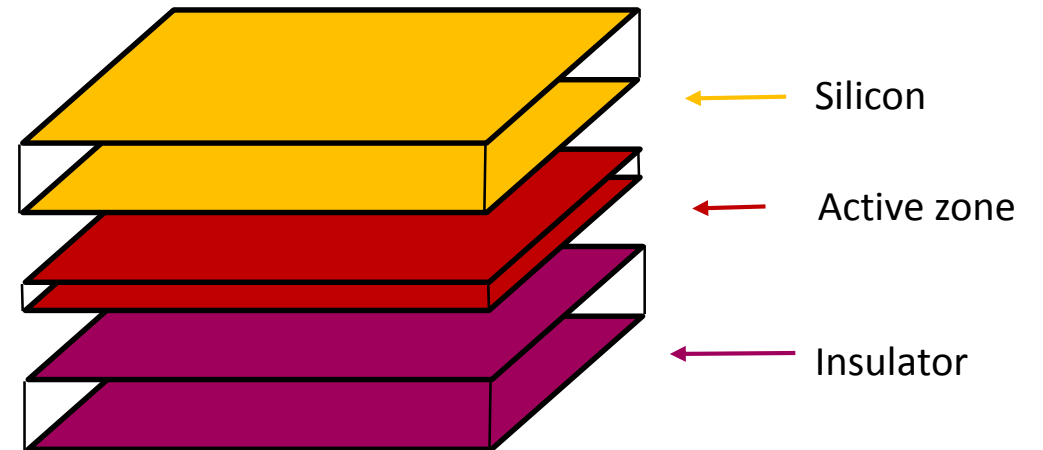


Why autofocus ?

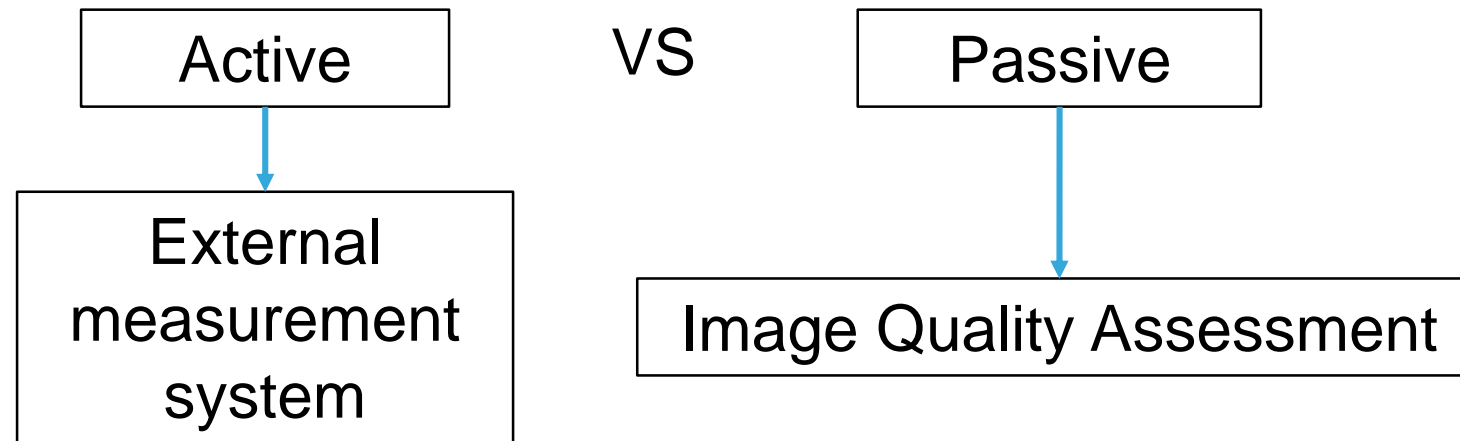
Laser injection bench



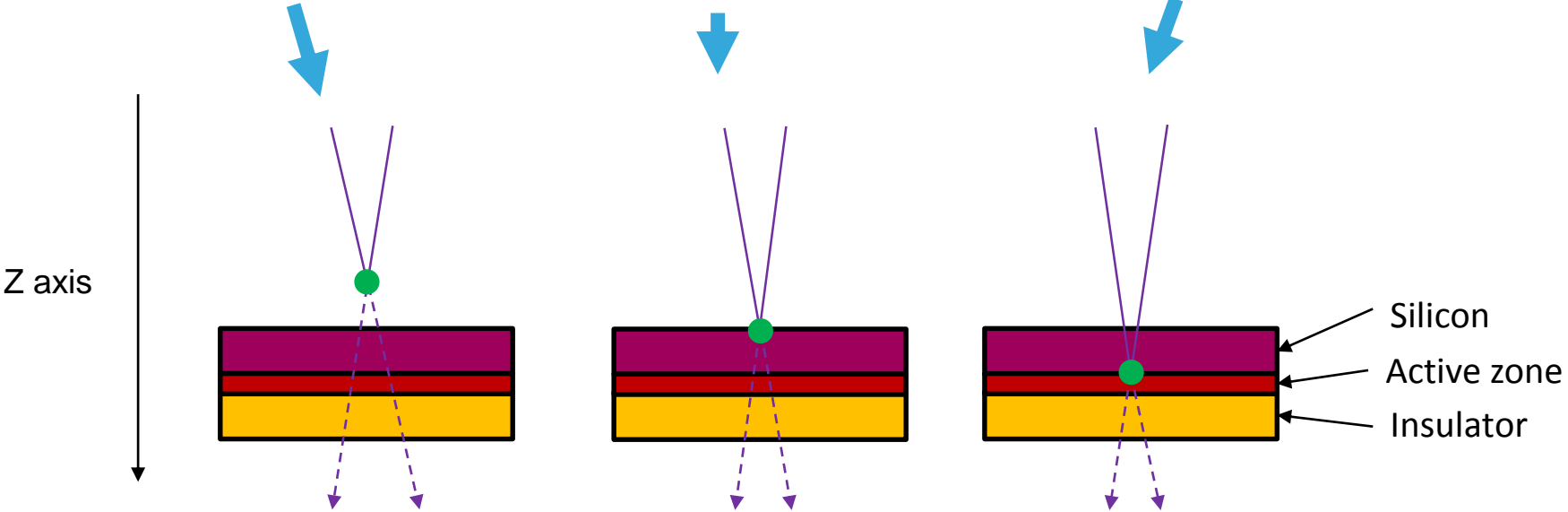
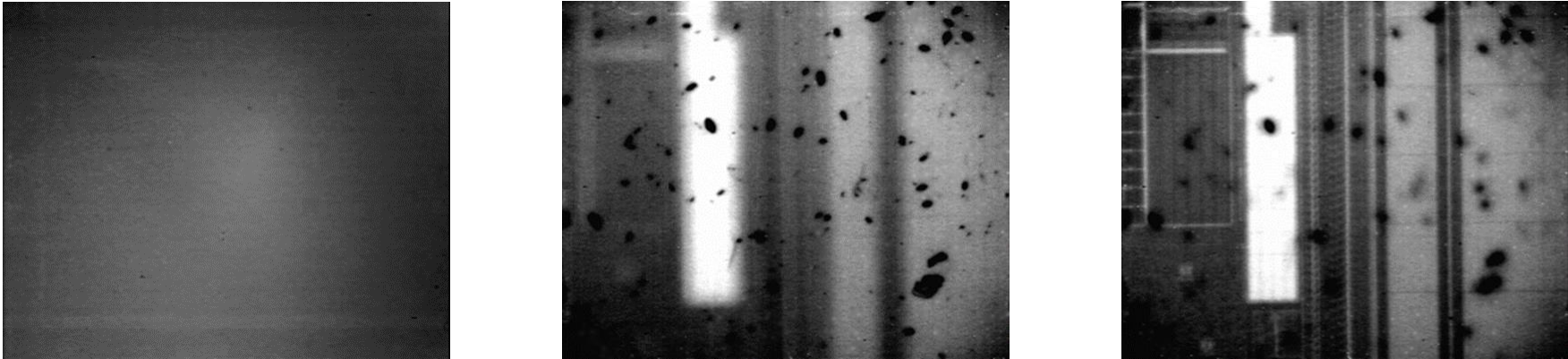
Concept mapping of a microchip



Around autofocus approaches



1.

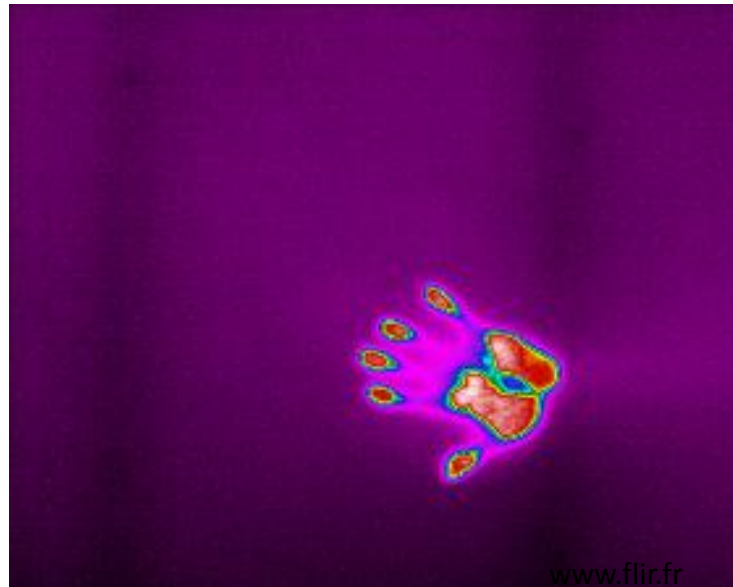


Focal point (green) according the Z height. Out of focus component (left), focus on the silicon surface (middle), and focus on the conductive tracks (right).

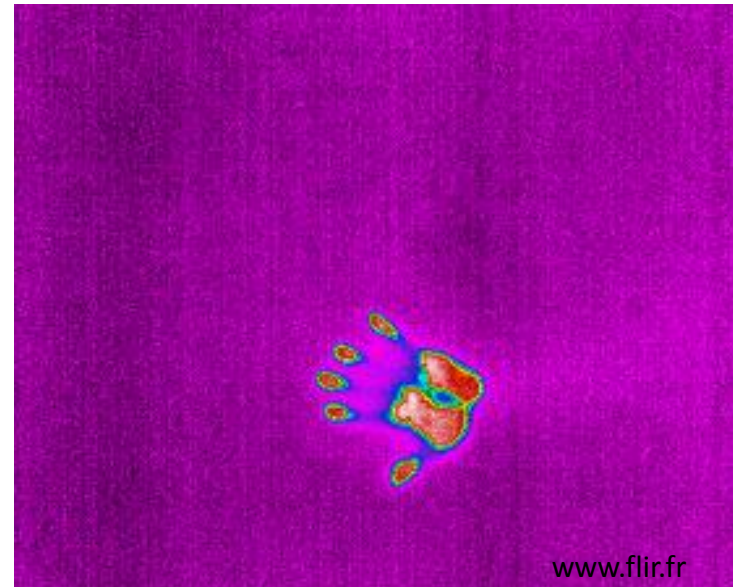
1. Directional information VS sharpness
2. Multiple scale capability

1. Directional information VS sharpness
2. Multiple scale capability
3. Thermal noise from the camera sensor

Cooled infrared camera



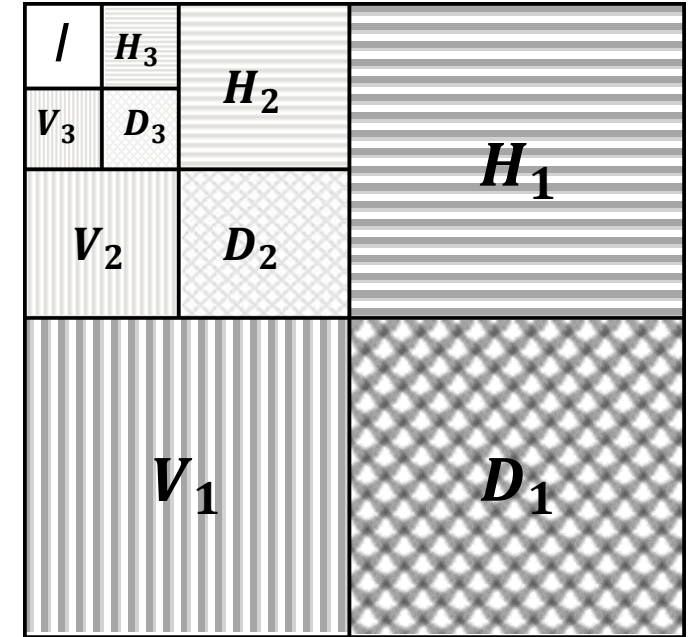
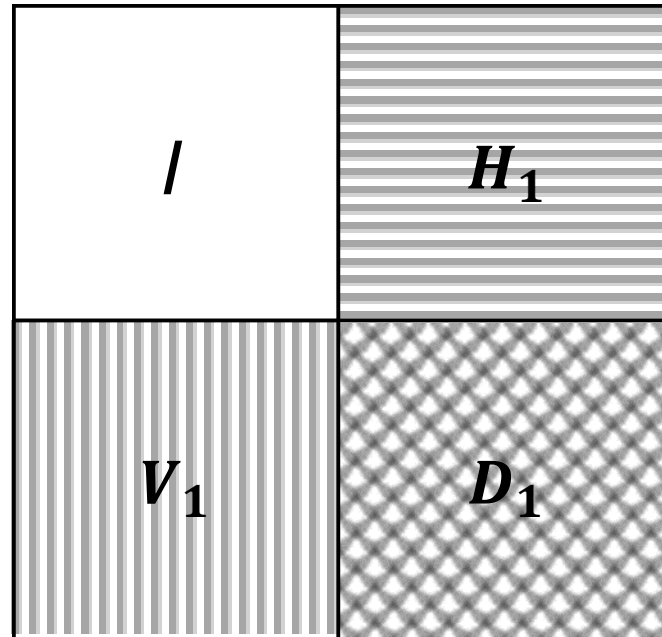
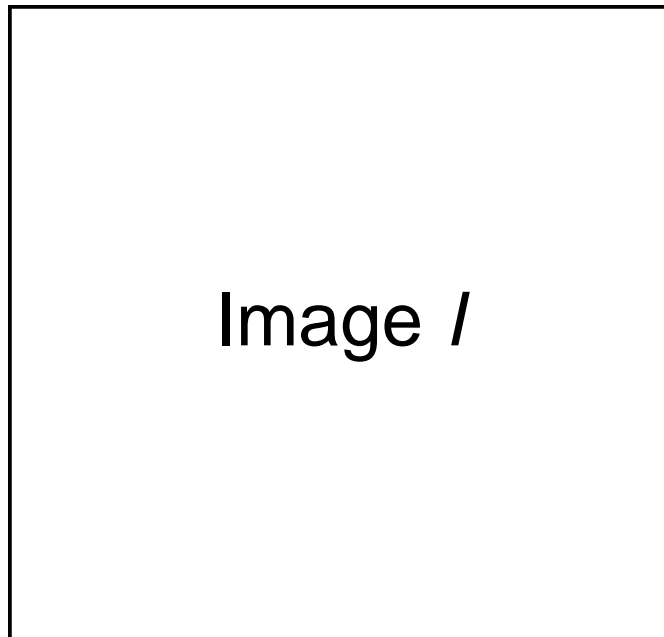
Uncooled infrared



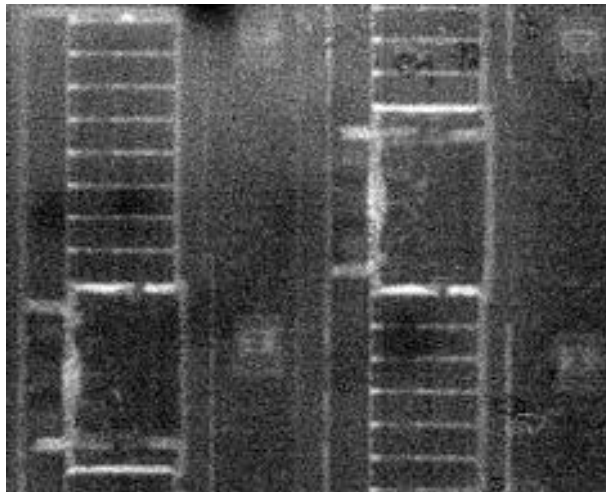
1. Directional information VS sharpness
2. Multiple scale capability
3. Thermal noise from the camera sensor
4. Reduced time consumption

Autofocus criterion: wavelet decomposition

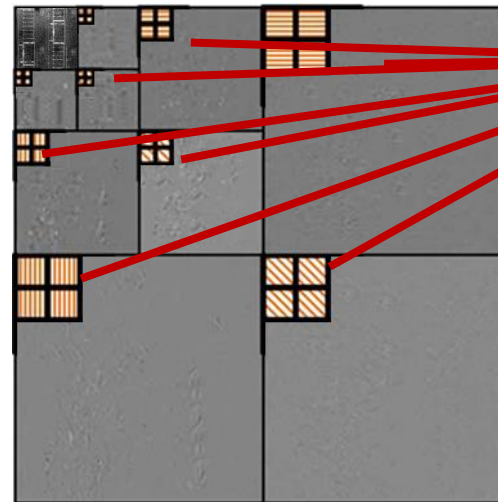
2D Discrete Wavelet Transform



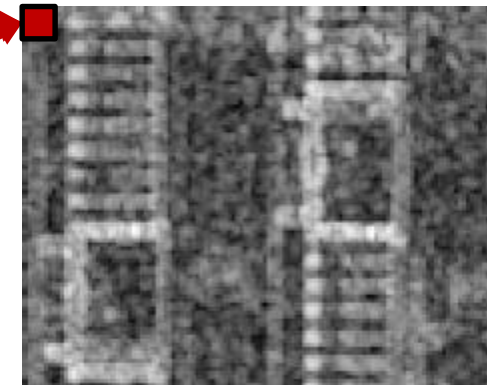
Autofocus criterion : wavelet choice



A component photo

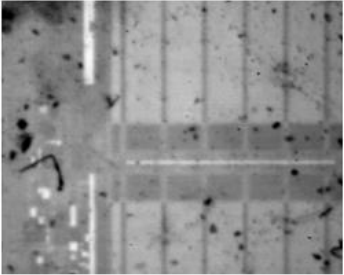
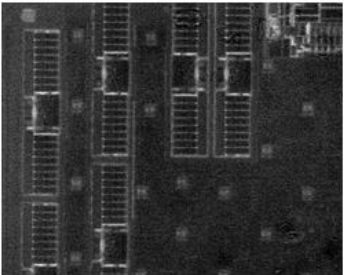
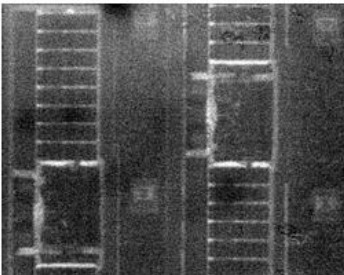


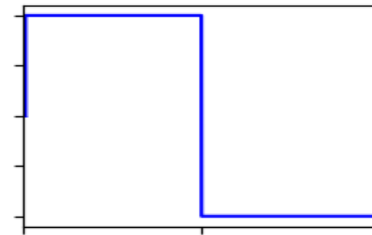
Wavelet decomposition coefficients



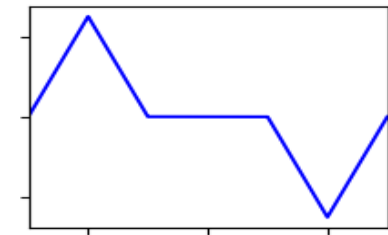
Coefficients map [2]

Autofocus criterion : wavelet choice

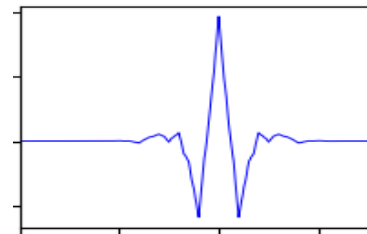
Zoom factor	Original acquisition
5x	
20x	
50x	



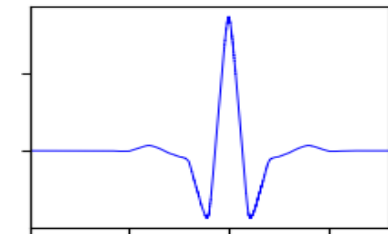
Haar



Custom Orthogonal Wavelet





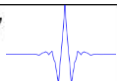
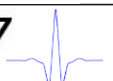
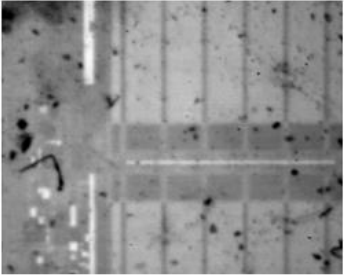
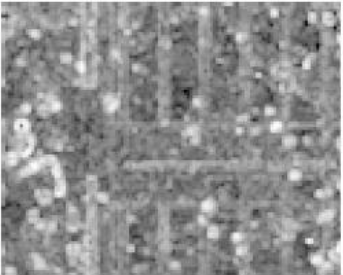
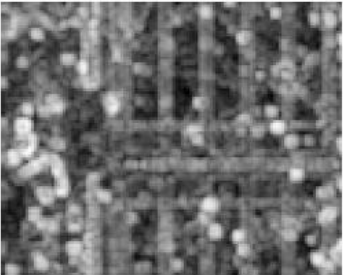
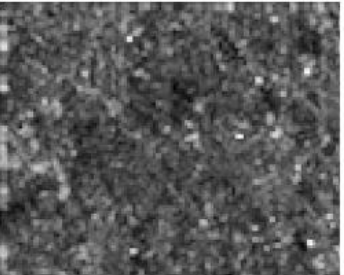

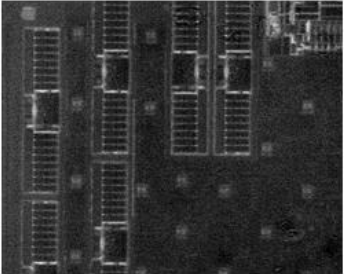
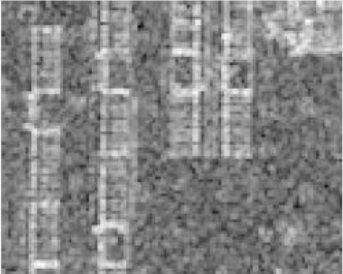
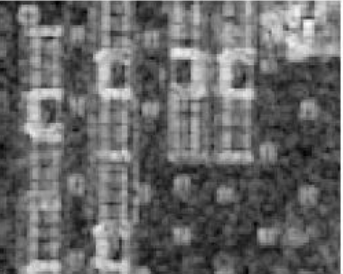
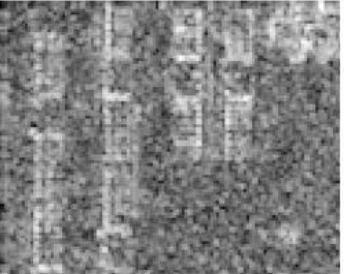
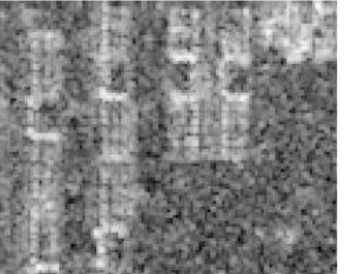
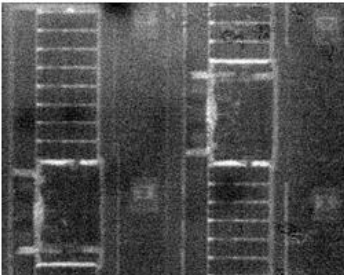
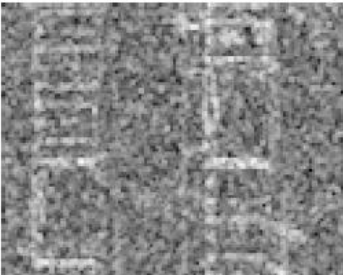
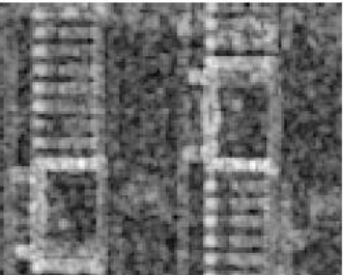
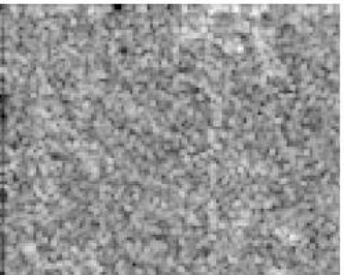
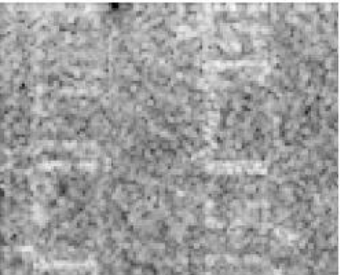
Cohen-Daubechies-Fauveau 9/7



Optimized CDF 9/7 [1]

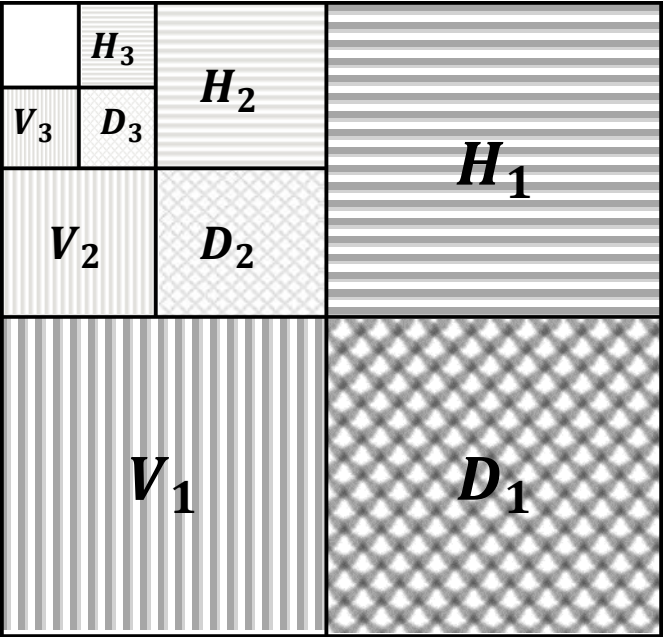
[1] ZHANG Songjun, *et al.* **A Novel 9/7 Wavelet Filter banks For Texture Image Coding.** *International Journal of Advanced Research in Artificial Intelligence*, 2012, vol. 1, no 6, p. 7-14.

Autofocus criterion: wavelet choice

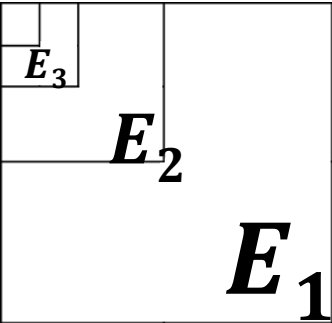
Zoom factor	Original acquisition	Coefficients maps depending on the wavelet used			
		Haar 	COW 	CDF 9/7 	OCDF 9/7 
5x					
20x					
50x					

Autofocus criterion: our contribution

FISH [2]

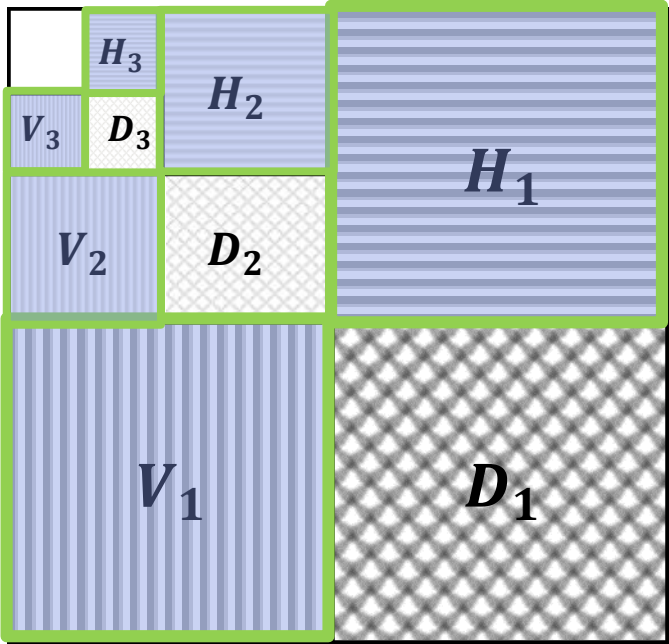


CDF 9/7

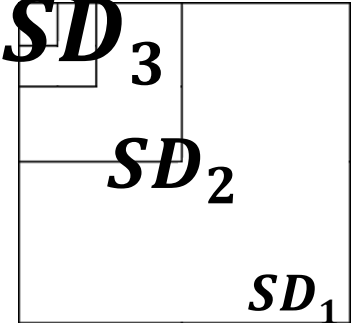


Energy levels

Our method



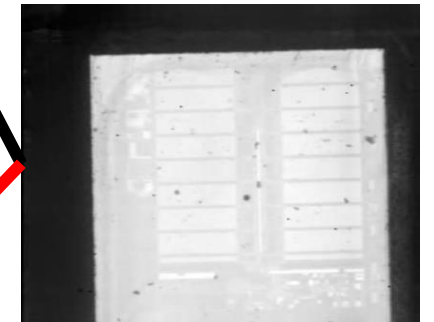
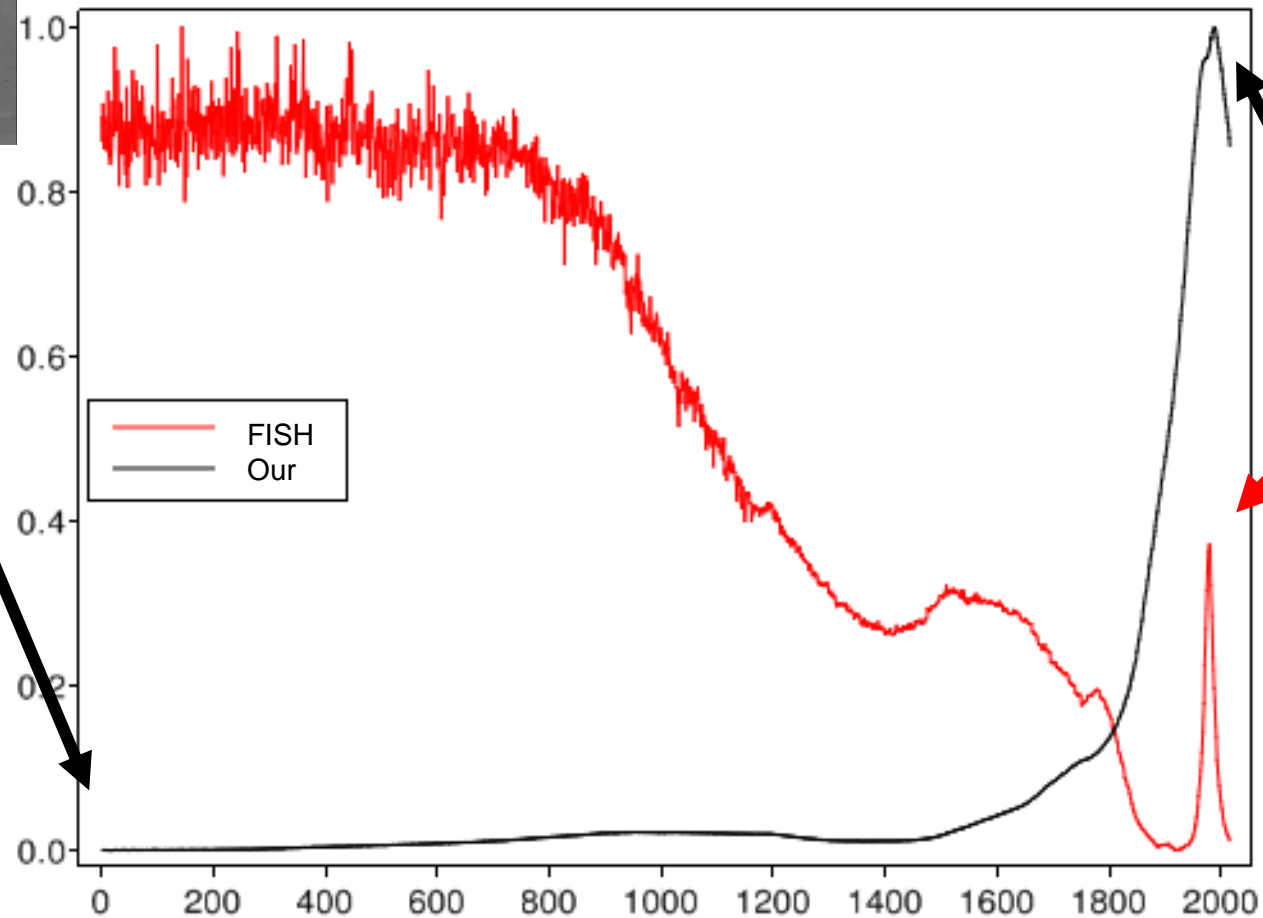
COW



Standard Deviation levels

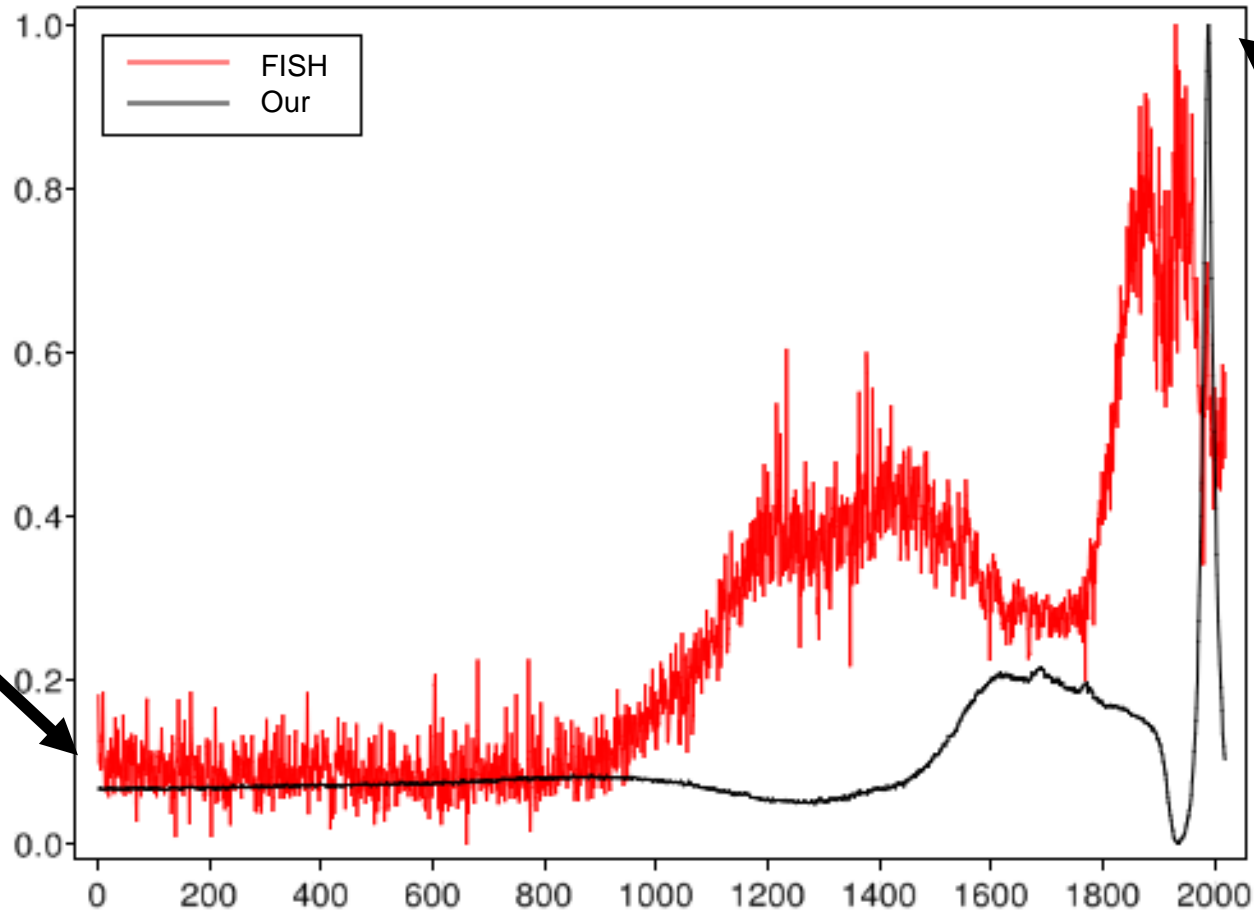
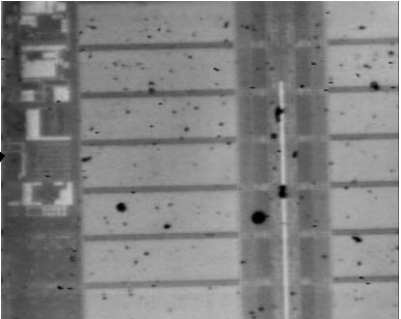


Images focus metric depending to Z axis (2.5x)

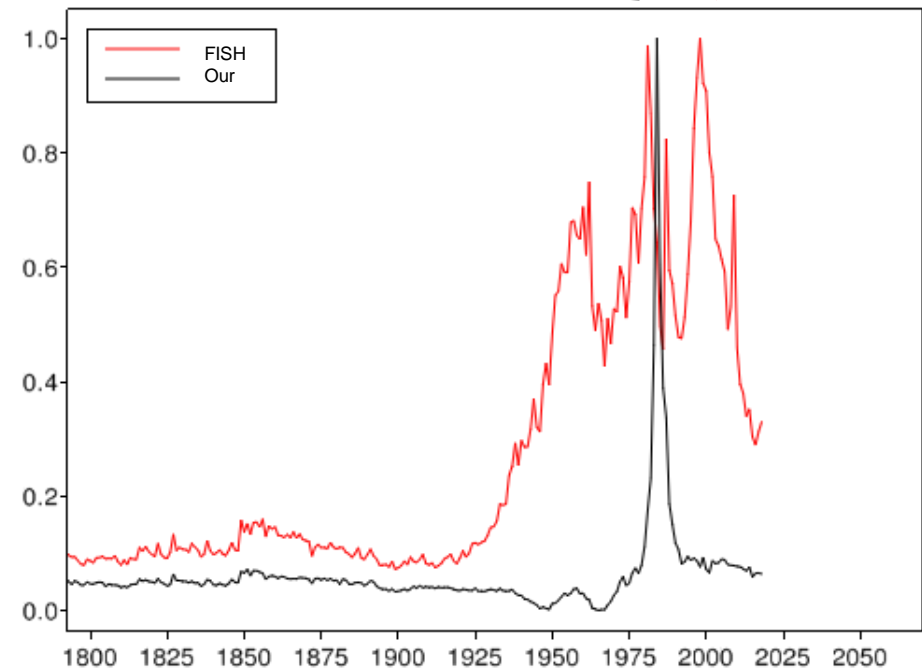
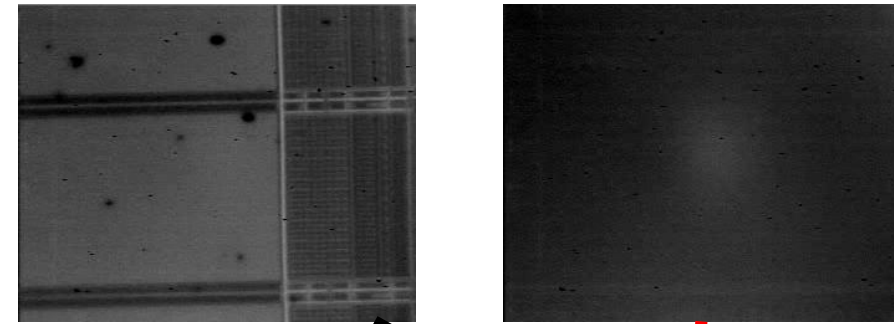
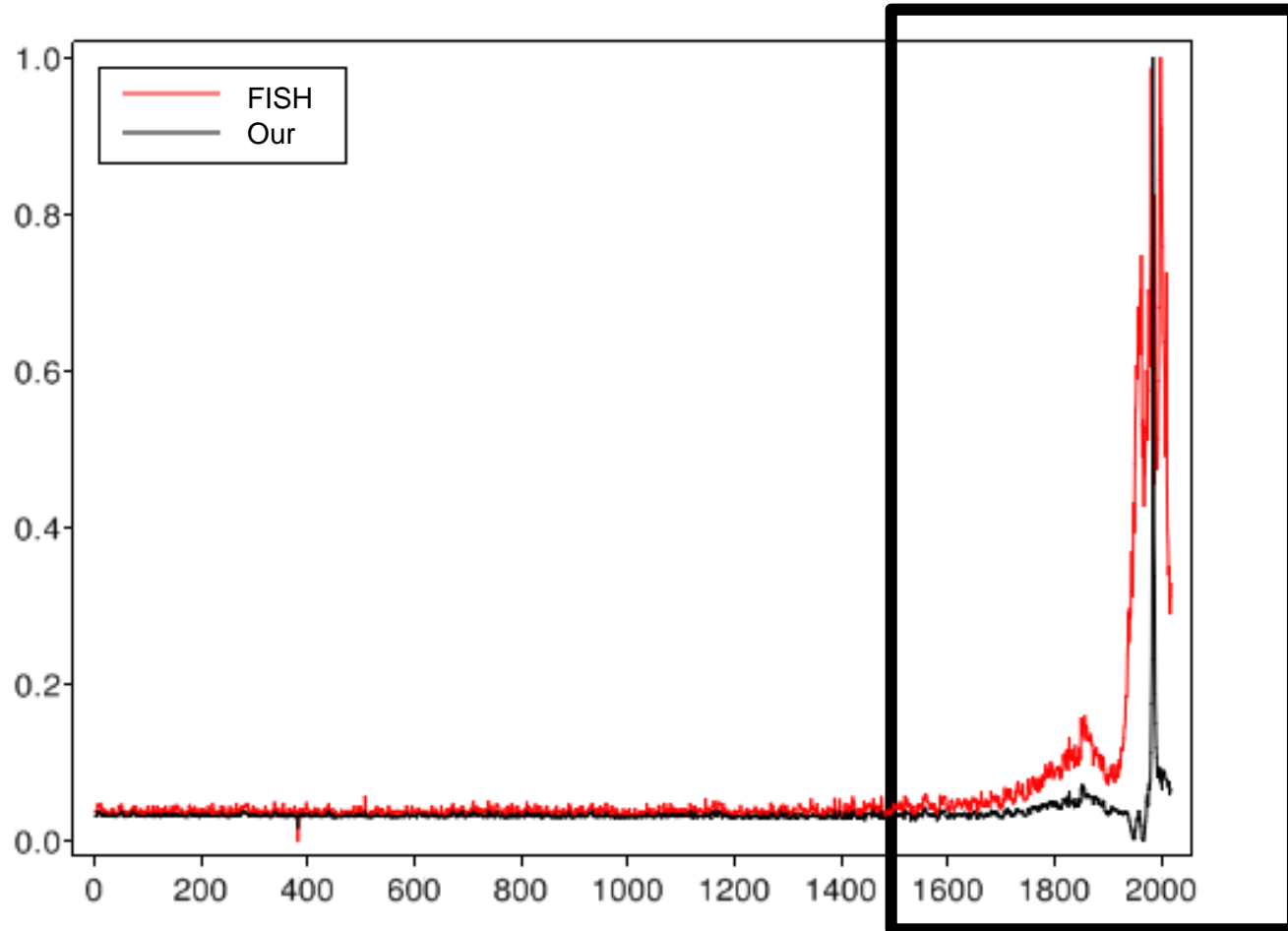


Experiment and results

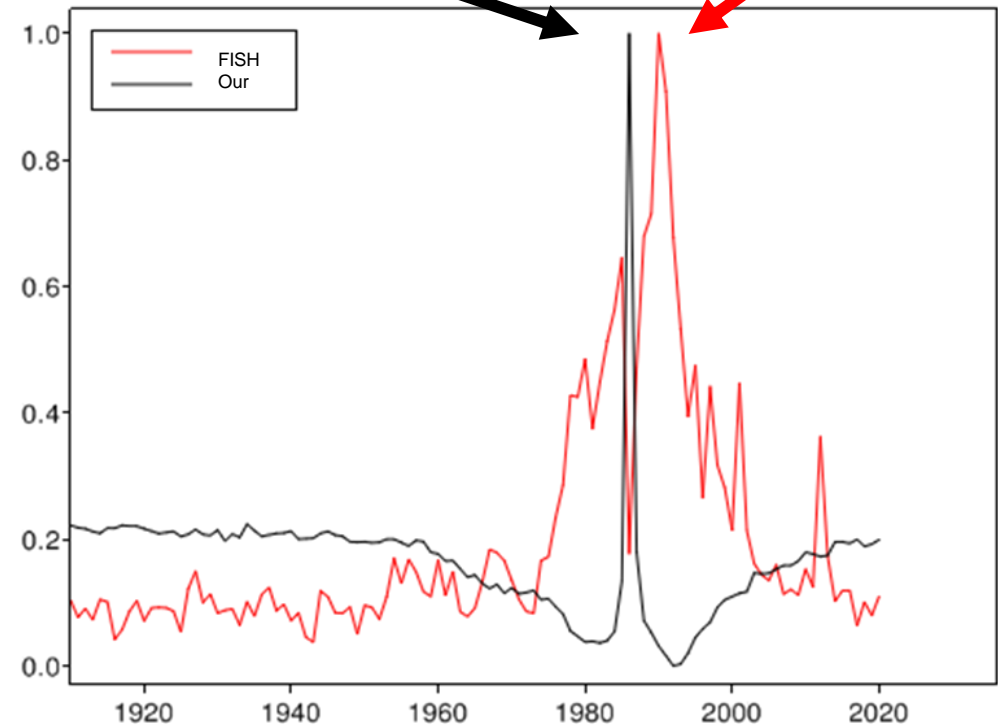
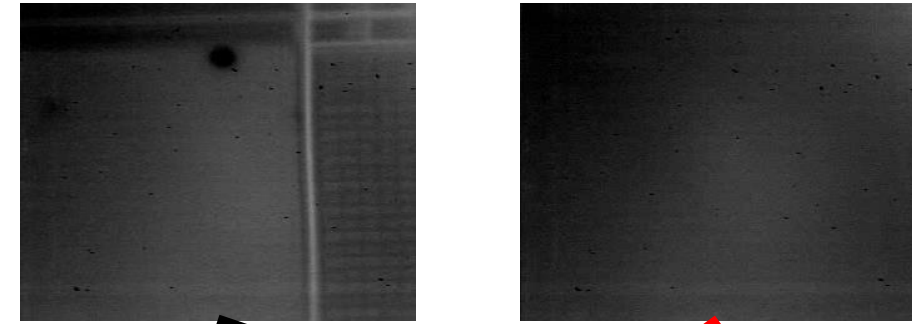
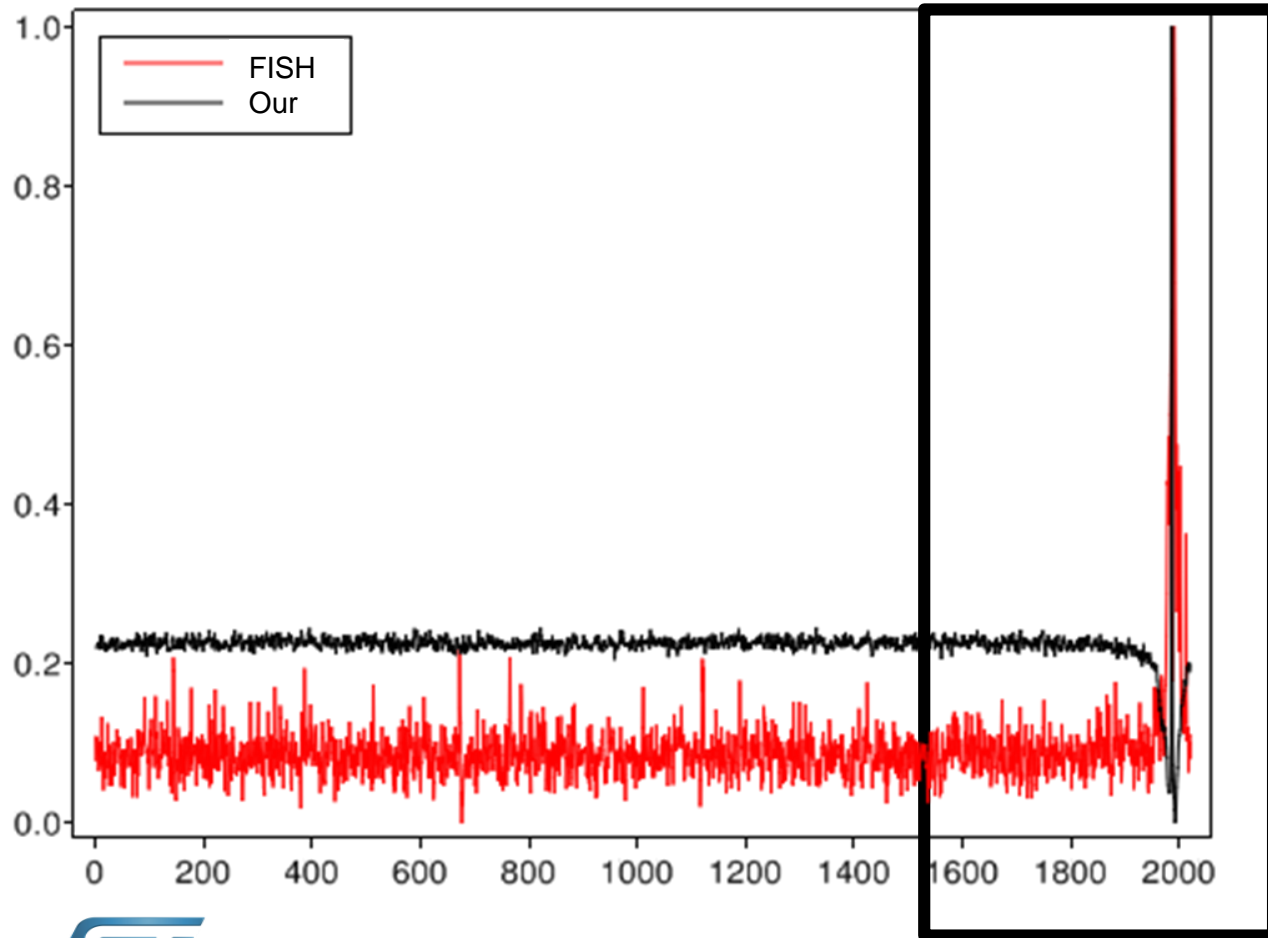
Images focus metric depending to Z axis (5x)



Images focus metric depending to Z axis (20x)



Images focus metric depending to Z axis (50x)

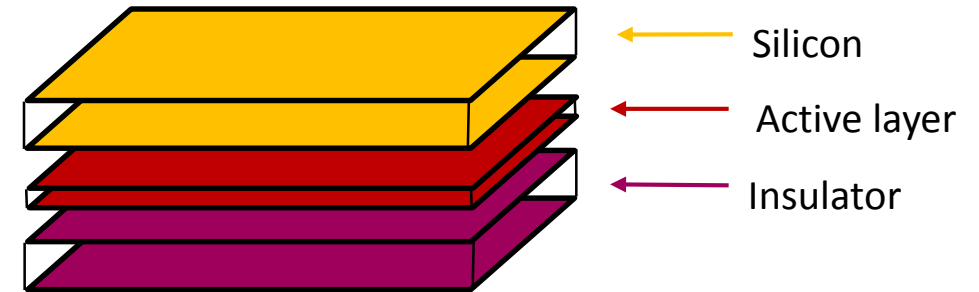


To recap

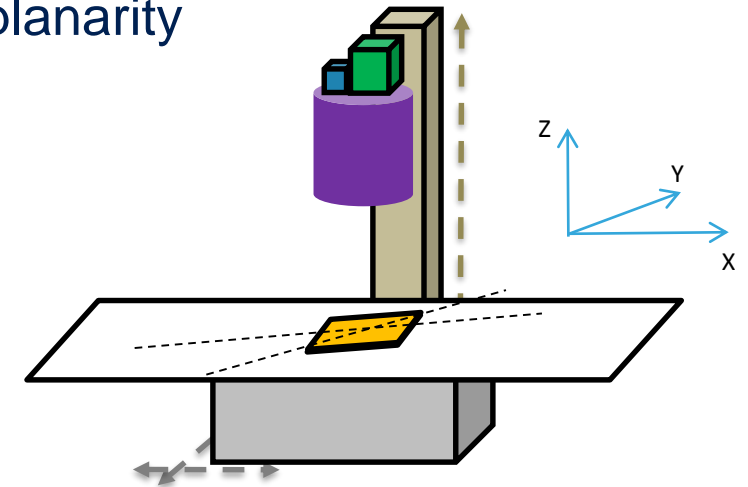
- Autofocus to detect active surface
- Conform with infrared vision & robust to thermal noise
- Multi-scale capable
- Computational time: 2.24ms/image
64-bit PC with an Intel Core i5 PC at 2.30GHz and 8Gb of RAM
Images resolution: 320x256 pixels

Perspectives

- Silicon thickness estimation



- Component planarity





Evaluation of Biometric Systems

Who said straightforward?

PHISIC – 2018, May 23rd – 24th



EURL au capital de 15.000€
SIREN 789 985 843 - RCS Aix-en-Provence - FRANCE

claude.barral@bactech.fr

www.bactech.fr

Parc de la Duranne
255, avenue Galilée
13857 Aix-en-Provence cedex 3
FRANCE

Agenda

- Current buzz about attacks and eval. + short introduction to Biometrics
- Private & Public biometric evaluation as of today
- Evaluation needs
 - Conformance
 - Performances
 - will be our focus: measuring False Acceptance Rate & False Rejection Rate
 - Security
- Best practices & ISO/IEC 19795
- Evaluation issues
- Conclusion



Current attack and evaluation buzz...



EURL au capital de 15.000€
SIREN 789 985 843 - RCS Aix-en-Provence - FRANCE

claude.barral@bactech.fr

www.bactech.fr

Parc de la Duranne
255, avenue Galilée
13857 Aix-en-Provence cedex 3
FRANCE

Current buzz...

- Smartphone
 - Face selfie
 - Standard front camera vs. Dedicated sensors (e.g. iPhone X FaceID)
 - Voice
 - Iris
 - Fingerprint not dead: « underglass » sensors and ultrasonic re-born
- Smartcards
 - Embedded fingerprint sensors
- Wearables
 - Easy liveness detection + constant monitoring of liveness

Public evaluation: NIST MINEX III, April 2018

- Fingerprint Match-on-Card
- FRR ranking at FAR = 1/10.000
 - Neurotechnology : 0,6%
 - Idemia : 0,9%
 - NEC : 0,9%
 - Innovatrics : 1,0%
 - AA Technologies : 1,2%
 - Gemalto/Cogent : 1,3%
 - ID3 Technologies : 1,5%
 - Beijing Highsign : 1,6%
 - Griaule Biometrics : 1,97%

Recent attack examples

- iPhone X FaceID
 - 3D mask + 2D images (eyes, nose, lips)
- Synthetizing voice
 - « 'Deep Voice' Software Can Clone Anyone's Voice With Just 3.7 Seconds of Audio »
- And still...
 - Synthetizing fingerprint
 - SFinGe software
 - Synthetic Fingerprint Generator
 - Fake fingers, as usual
 - Vascular pattern recognition
 - Not so perfect...





Very short introduction about Biometrics



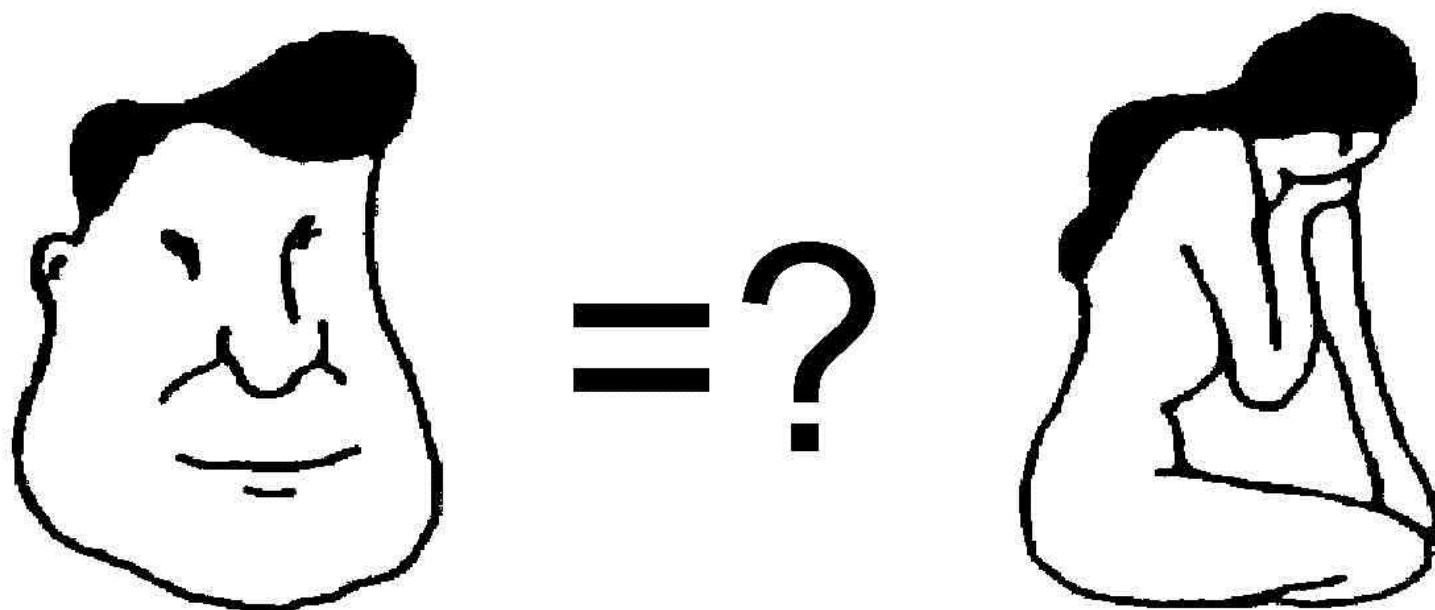
EURL au capital de 15.000€
SIREN 789 985 843 - RCS Aix-en-Provence - FRANCE

claude.barral@bactech.fr

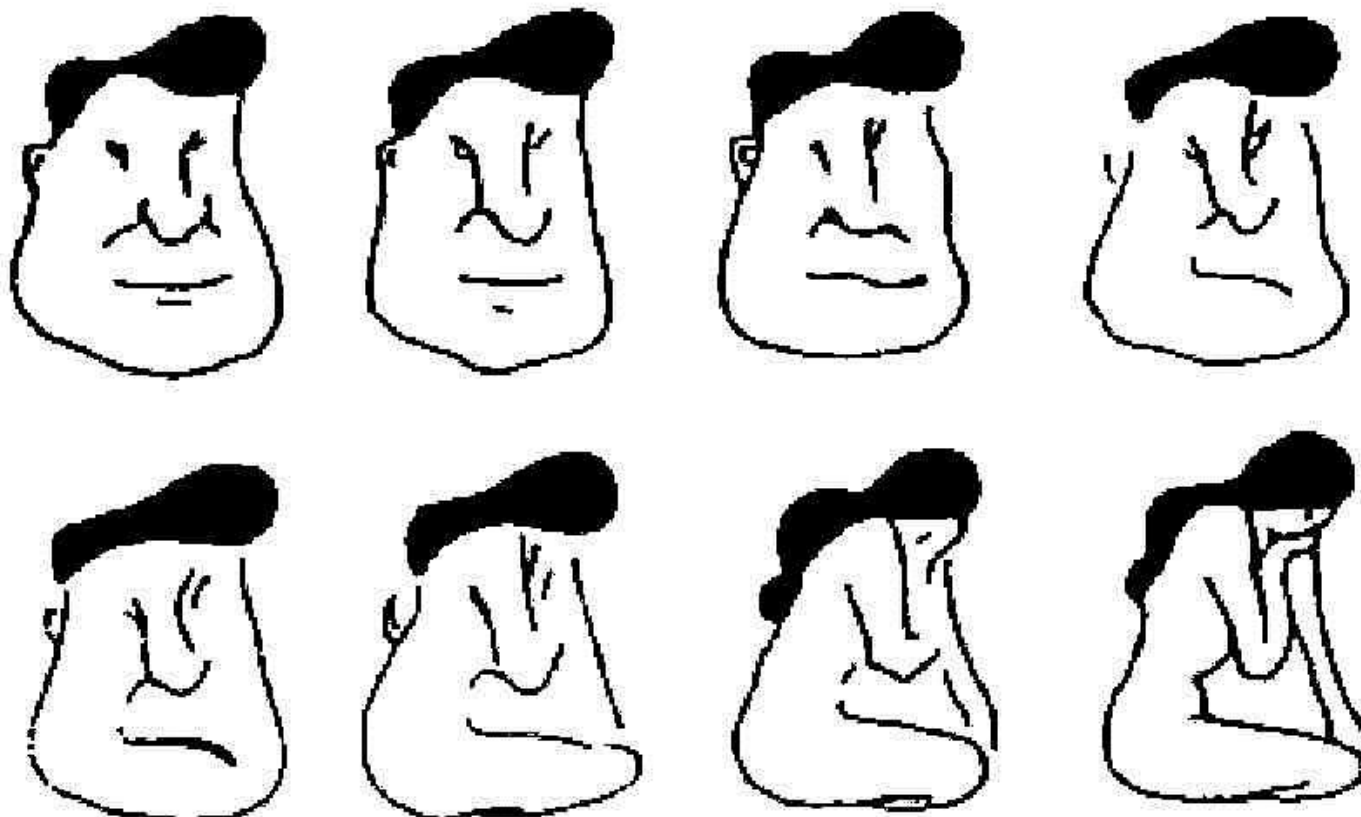
www.bactech.fr

Parc de la Duranne
255, avenue Galilée
13857 Aix-en-Provence cedex 3
FRANCE

Biometric Comparison Issue (1/2)



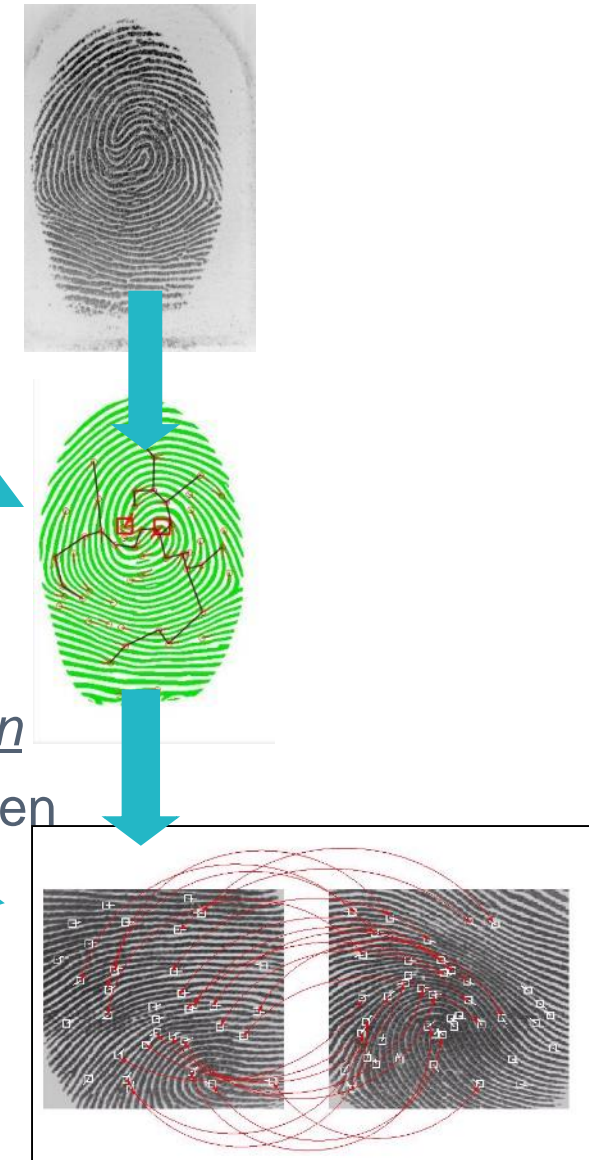
Biometric Comparison Issue (2/2)



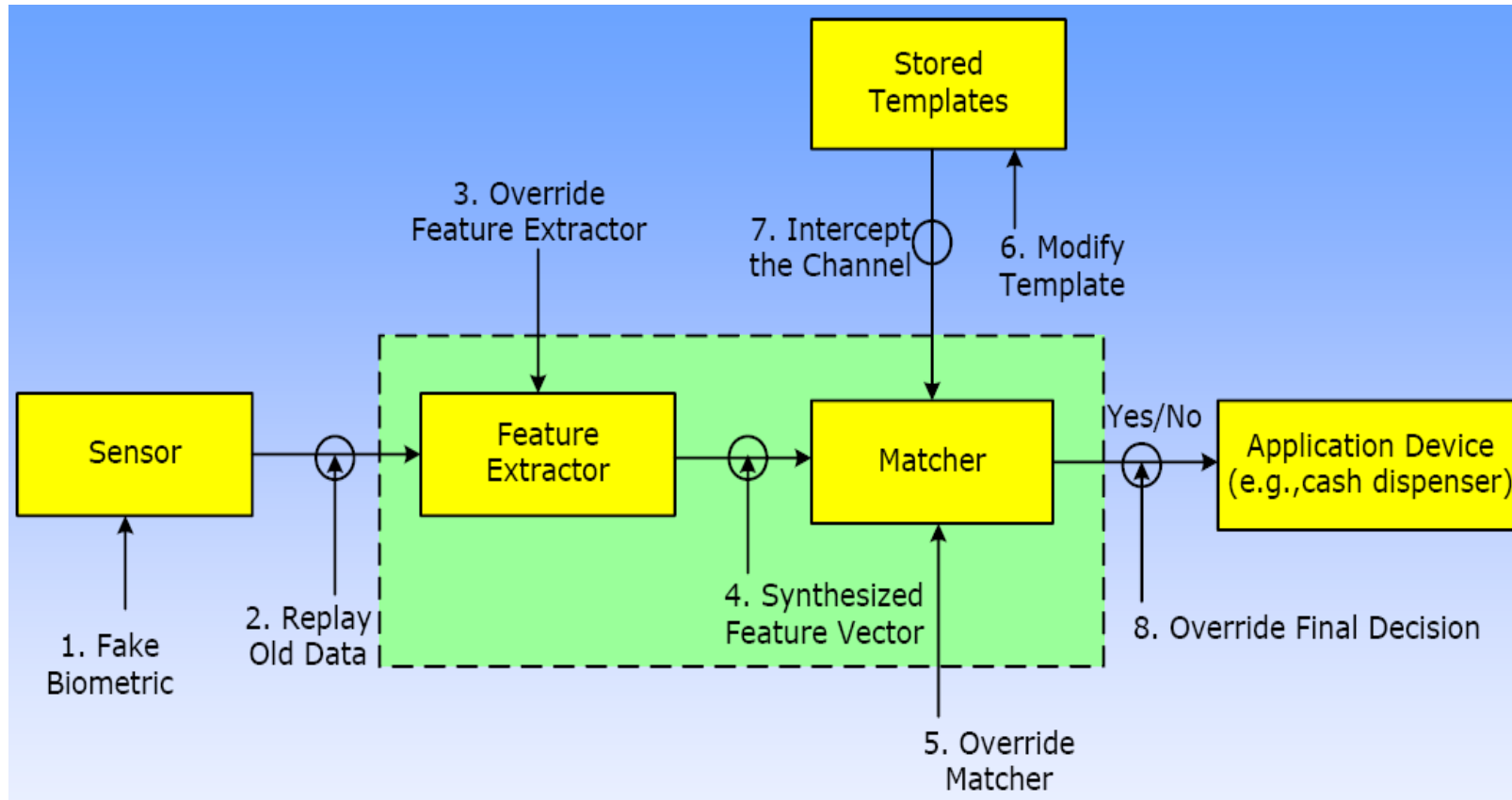
It's only a question of **threshold**

Biometric Process

- Enrollment: reference template
 - Image *capture*
 - Reference template extraction
 - Reference template *storage*
- Authentication: candidate template
 - Image *capture*
 - Candidate (or *Probe*) template extraction
 - Templates matching (comparison between reference and probe)



System Architecture (and its weaknesses...)





Biometric systems evaluation



EURL au capital de 15.000€
SIREN 789 985 843 - RCS Aix-en-Provence - FRANCE

claude.barral@bactech.fr

www.bactech.fr

Parc de la Duranne
255, avenue Galilée
13857 Aix-en-Provence cedex 3
FRANCE

Which evaluation for biometric systems?

- Conformance to standards / Interoperability
 - ISO / ANSI interoperability of templates
 - Interfaces conformance (BioAPI, CBEFF)
- Security / Attacks resilience
 - Physical and logical attacks detection
 - PADER: Presentation Attack Detection Error Rate
 - Liveness detection
- Performances assessment
 - FRR: False rejection rate
 - FAR: False acceptance rate
 - FTE: Failure to enroll
 - Enrollment / Authentication timings & algorithmic complexity (min. run. plat.)

Public / Private evaluations

- NIST
 - Fingerprints, face, iris, voice
 - Both standardized & proprietary templates
 - Both authentication (1 to 1) & identification (1 to n) schemes
- Academics
 - Many EU funded projects

=> Focuses on performances and interoperability

- No mature certification schemes at Common Criteria, ITSEFs...

Reference standard

- ISO/IEC 19795
 - Biometrics Performance Testing and Reporting
 - « Technology evaluation »
 - « Scenario evaluation »
 - « Operational evaluation »
 - « rule-of-3 », « rule-of-30 »

ISO 19795

- Biometrics Performance Testing and Reporting (WG5)
- -1 : Principles and framework (IS – 2006)
- -2 : Testing Methodologies (IS – 2007)
- -3 : Specific Testing Methodologies (IS - 2007)
- -4 : Interoperability Testing of Data Formats (IS – 2008)
- -5 : Performance Evaluation for Access Control (CD3)
- -6 : Testing Methodologies for Operational Evaluation (CD3)
- -7 : On-Card Biometric Comparison Algorithm (FCD)

* standards state status (CD / FCD / IS) may not be up to date



Conformance / Interoperability



EURL au capital de 15.000€
SIREN 789 985 843 - RCS Aix-en-Provence - FRANCE

claude.barral@bactech.fr

www.bactech.fr

Parc de la Duranne
255, avenue Galilée
13857 Aix-en-Provence cedex 3
FRANCE

Standardization Bodies

- ISO SC37 - Biometrics
- ISO SC17 – Identification Cards
- ISO SC27 – IT Security Techniques
- CEN (EU)
- Afnor (Fr)
- ICAO – Travel Documents
- ANSI/NIST (US)

ANSI/NIST

- Clear and practical (generally a subset of ISO version)
- Published a little before ISO version from INCITS M1
- ANSI/INCITS 358-2002 BioAPI
- NISTIR 6529-A, "Common Biometric Exchange Formats Framework (CBEFF)"
- ANSI/NIST-ITL 1-2000 Data Format for the Interchange of Fingerprint, Facial, & Scar Mark & Tattoo (SMT)
- NISTIR 7151 Fingerprint Image Quality

ANSI/NIST

- ANSI/INCITS 378-2004 Finger Minutiae Format for Data Interchange
- SP 800-76 Biometric Data Specification for Personal Identity Verification (related to FIPS 201)
- AAMVA DL/ID-2000 National Standard for the Drivers License/Identification Card

ISO JTC1 SC17, Identification Cards

- WG1: Physical Characteristics & Test Methods
- WG2: *deprecated*
- WG3: Machine Readable Travel Document
 - e.g. Passport, Visa, National ID
- WG4: Contact Cards
 - e.g. Communication Interface and Protocols
- WG5: Registration Management Group
 - e.g. Issuer Identification Numbers, Application Provider Identifiers

ISO JTC1 SC17, Identification Cards

- WG6: *deprecated*
- WG7: *deprecated*
- WG8: Contactless Cards
 - e.g. Communication Interface and Protocols
- WG9: Optical Memory Cards & Devices,
 - *almost deprecated?*
- WG10: Motor Vehicle Driving Licence & related doc.
- WG11: On-Card Biometrics & Personal Identification

ISO JTC1 SC27, IT Security

- WG1: Information Security Management Systems
- WG2: Cryptography and Security Mechanisms
 - e.g. Encryption, Signature, Hash, MAC, RNG
- WG3: Security Evaluation, Testing & Specification
 - e.g. Common Criteria
- WG4: Security Controls & Services
 - e.g. Intrusion Detection Systems, Access Control
- WG5: Privacy & Identity management
 - e.g. Security issues with biometrics and other personal data

ISO SC37 - Biometrics

- WG1: Harmonized Biometric Vocabulary
- WG2: Technical Interfaces
 - e.g. BioAPI, CBEFF
- WG3: Data Interchange Formats
 - e.g. Fingerprint minutiae, finger image, facial image
- WG4: Functional Architecture and Related Profiles
- WG5: Performance Testing & Reporting
- WG6: Cross-Jurisdictional and Societal Aspects

ISO 19794 (1/4)

- Biometric Data Interchange Formats (WG3)
- ISO process to published standard is very long
 - 2,5 years at best!
- -1 : Framework
 - Level : IS - 2005
- -2 : Finger Minutiae Data
 - Level : IS - 2005
 - Close to DIN66400V (Germ.), ANSI/NIST-ITL 1-2000 (US)
- -3 : Finger Pattern Spectral Data
 - Level : IS - 2005

ISO 19794 (2/4)

- -4 : Finger Image Data
 - Level : IS - 2005
- -5 : Face Image Data
 - Level : IS – 2005 + A2:2009
- -6 : Iris Image Data
 - Level : IS - 2005
- -7 : Signature/Sign Time Series Data
 - Level : IS - 2007
- -8 : Finger Pattern Skeletal Data
 - Level : IS - 2006

ISO 19794 (3/4)

- -9 : Vascular Image Data
 - Level : IS - 2007
- -10 : Hand Geometry Silhouette Data
 - Level : IS – 2007
- -11 : Signature/Sign Processed Dynamic Data
 - Level : IS – 2013
- -12 : Feature-Based Face Recognition Data
 - Deprecated?
- -13 : Voice Data
 - Level : CD

ISO 19794 (4/4)

- -14 : DNA Data
 - Level : IS - 2013
- Recent proposals:
 - 19794-15 : Palm Line features (NWI)

ISO 19795

- Biometrics Performance Testing and Reporting (WG5)
- -1 : Principles and framework (IS – 2006)
- -2 : Testing Methodologies (IS – 2007)
- -3 : Specific Testing Methodologies (IS - 2007)
- -4 : Interoperability Testing of Data Formats (IS – 2008)
- -5 : Performance Evaluation for Access Control (CD3)
- -6 : Testing Methodologies for Operational Evaluation (CD3)
- -7 : On-Card Biometric Comparison Algorithm (FCD)
 - NIST: MINEX II

ISO 29109

- Information technology -- Conformance testing methodology for biometric data interchange formats defined in ISO/IEC 19794
- One to one matching with ISO 19794

Biometrics in ICAO

- ICAO specifies the use of contactless electronic chip
- ICAO specifies the use of Biometrics
 - Mandatory: face recognition
 - Optional: fingerprint recognition, iris recognition
- ICAO specifies the use of cryptography
 - Both symmetric & asymmetric
 - For chip and terminal authentication
 - Challenge-response
 - For session key agreement
 - Diffie-Hellman
 - For integrity and authenticity of data
 - Digital signature

Biometrics in ICAO - Logical Data Struct.

- DG1: MRZ data
- DG2: Face, 15-20kB
- DG3: Fingerprint(s), 10-12kB per finger
- DG4: Iris(es), 30kB per eye
- DG5: Portrait image (the printed one)
- DG7: Signature image (the printed one)
- DG15: Chip Public Key for Active Authentication
- Reserved for Future Use (RFU):
 - DG6, DG14

Dedicated Biometrics&Security stds

- ISO 24745 Information Technology – Security Techniques – Biometric Information Protection
- ISO 19092 Financial Services – Biometrics – Security framework
- ISO 19792 Information Technology – Security Techniques – Security Evaluation of Biometrics
- ISO 24760 Information Technology – Security Techniques – A framework for identity management
- ISO 29100 Information Technology – Security Techniques – A privacy framework

Common Criteria

- ISO SC27 WG3
- Evaluation Assurance Level
 - EAL1 = Functionally Tested
 - EAL2 = Structurally Tested
 - EAL3 = Methodically Tested & Checked
 - EAL4 = Methodically Designed, Tested & Reviewed
 - EAL5 = Semiformally Designed & Tested
 - EAL6 = Semiformally Verified, Design & Tested
 - EAL7 = Formally Verified, Design & Tested
- Authorizes intermediate levels
 - e.g. EAL4+
- Current initiatives to map smartcard certification procedures onto biometric systems



Security



EURL au capital de 15.000€
SIREN 789 985 843 - RCS Aix-en-Provence - FRANCE

claude.barral@bactech.fr

www.bactech.fr

Parc de la Duranne
255, avenue Galilée
13857 Aix-en-Provence cedex 3
FRANCE

Keywords

- Presentation attack
 - presenting a fake to the biometric sensor
- Spoofing
 - bypassing the biometric system with presentation attack
- a Real
 - the living biometric trait
- a Fake / a Dummy
 - a copy of the biometric trait
- Positive attack
 - the attack succeeded

Let's focus on « presentation attacks »

- So-called « presentation attacks »
 - Presenting your biometric trait, or *something else*, to the biometric sensor...
- Attack efficiency and convenience level depend on the security context
 - Facing an authority? Remote monitoring? All alone?
- Consumer applications and personal mobile devices
 - Mainly in the « all alone » context

Attacks efficiency levels

- Level 1 (the most critical)
 - Enroll a *real*, copy it, authenticate with the *fake* (for how long?)
 - Issue is critical full system security
- Level 2
 - Enroll a *fake*, authenticate with the *fake* (for how long?)
 - Issue is critical liveness detection
- Level 3
 - Enroll a *fake*, failed authentication with the *fake*
 - Issue is non-critical liveness detection
- Level 4 (the less critical)
 - Can't even enroll a *fake*! Perfect?

Dummy fingers...



... and targeted sensors



Beyond fingerprint spoofing...



Figure 5-27: Vascular pattern spoofs effective with liveness detection turned off. a) The copy of a vascular pattern stuck on a bottle. b) The bottle spoof verified as an authorised user. c) The copy of a vascular pattern stuck on a hand (left), next to the original hand (right) of an authorised user. d) The hand spoof verified as the authorised user



Spoofing Biometrics, Solutions

- Counter-measures: so-called "liveness detection"
 - Sensor ability to detect fake biometric traits:
 - Temperature, skin conductivity –elec., thermal...-
 - Heart beat, Blood pressure
 - ...
 - Face to face or video check
- Issues
 - Tolerance range vs. environment

Logical attacks

- Random template generation
 - Statistical attack in relation with the FAR of the system
- Hill-climbing attack if access to the score value (0-100%)
 - Starts with random template and process fine-tuning depending on score value
- Random Image/Signal synthesizing
- Cheating with extraction algorithm, matching algorithm, final decision
- Cheating with the reference template
 - Changing reference template data linked with the targeted ID within the database



Performances



EURL au capital de 15.000€
SIREN 789 985 843 - RCS Aix-en-Provence - FRANCE

claude.barral@bactech.fr

www.bactech.fr

Parc de la Duranne
255, avenue Galilée
13857 Aix-en-Provence cedex 3
FRANCE

Performance evaluation issues

- In « technology mode » (usual focus and first step)
 - Algorithm performances against large databases
 - Many hundreds to many thousands of samples needed
 - Public databases available or not depending on the biometric modality under test
 - Sometimes need to build our own database
 - Low impact of the human factor
- In « scenario mode »
 - Enrolment and authentication with live capture in environmental conditions
 - Run on several days with several tens of recurring users – high impact of human factor
- In « operational mode »
 - Active survey of system performances over several weeks on the field of deployment

Technology mode evaluation issues

- Only real measurement at large scale of FAR, FRR, FTE
- Integration of the system under test with the database and test sequencer server => need to develop interfaces with API/SDK of the target system
- What about the target OS?
 - Windows? Mac? Linux? iOS? Android?...
- What about the target platform?
 - On-server? On-client? On-Card?
- What about the needed target size and architecture of the database?
 - Many capture of a same sample from a same person for FRR eval. (\neq time and location)
 - Many capture of a same sample from many other persons for FAR eval.
 - Grow the DB in coherence with the FAR/FRR to certify (x3 to x30 tests more than target)

Technology mode evaluation issues

- Claiming 1/100.000 FAR requests a minimum of 300.000 tests with a maximum of 3 errors
 - 300.000 tests need about 550 different biometric samples
 - ... and a lot of resources (time, storage, processing and analysis capabilities)
- Several security setting levels to evaluate, many secondary thresholds
 - Exponential growth of possible settings combination
- Few biometrics are more « wide » than others: Voice
 - Text-dependent vs text-independent, language-dependent vs language-independent, natives vs non-natives speakers, accents...
- Ability to detect possible bias if too much tuned for a specific target corpus
 - « developers » approach vs. « scientists » approach during test and dev.



Conclusion



EURL au capital de 15.000€
SIREN 789 985 843 - RCS Aix-en-Provence - FRANCE

claude.barral@bactech.fr

www.bactech.fr
Parc de la Duranne
255, avenue Galilée
13857 Aix-en-Provence cedex 3
FRANCE

Well, not so straightforward...

- Size and representativeness of databases
 - public databases vs. Private databases and GDPR regulation
- Automation of large scale tests (>>100k), test pairs sequencer & on-the-fly result interpretation (false/real positives/negatives)
- Side effect: thousands of tests per second heavily charge the system under test and assess about the operational reliability of the solution
 - especially important for embedded systems to track bad memory management ;-)
- Lots of initiatives for a dozen of years from many institutional entities such as Common Criteria, ANSSI, ITSEF labs... But still not mature enough!

www.bactech.fr, your biometric evaluation lab



BACTECH
Biometrics And Cryptography TECHNOLOGIES (made easy for you)

Parc de la Duranne - 255 avenue Gallié - 13857 Aix-en-Provence cedex 3 - FRANCE
M. +33 (0)6 86 83 19 55

HOMEPAGE TRAININGS CONSULTING DEVELOPMENT EXPERTISE AREAS CONTACT

ABOUT BACTECH

Bactech helps you to understand and use complex IT security features such as Biometrics, Cryptography and electronic embedded systems such as Smart Cards.

Bactech is proposing many trainings and consultancy services, participates to research projects, delivers academic courses to MS students and engineers. With more than fifteen years of experience in electronics and IT, especially in Smart Cards, our consultants may also help you to develop proof-of-concepts of your future products.

Bactech is member of the French national body for standardization (Afnor) and participates to ISO SC37 (Biometrics), ISO SC27 (IT Security) and ISO SC17 (Identification Cards). Our consultants are effective in standardization for more than ten years now.

BACTECH - Biometrics and Security training, consulting, development and support.

EESTEL
SOLUTIONS COMMUNICANTES SECURISEES
POLE DE COMPETENCE MONDIALE

IEEE Certified Biometrics Professional® (CBP) Program

BACTECH - EURL au capital de 15.000€ - SIREN 789 985 843 - RCS Aix-en-Provence - FRANCE
WEB DESIGN NOON GRAPHIC DESIGN



Thank you.



EURL au capital de 15.000€
SIREN 789 985 843 - RCS Aix-en-Provence - FRANCE

claude.barral@bactech.fr

www.bactech.fr
Parc de la Duranne
255, avenue Galilée
13857 Aix-en-Provence cedex 3
FRANCE

An EM Fault Injection Susceptibility Criterion and its application to the localisation of hotspots

M. MADAU, M. AGOYAN, P. MAURINE

Laboratoire d'Informatique de Robotique et de Microélectronique de Montpellier
(LIRMM), STMicroelectronics

PHISIC 2018



Fault injection in practice

Fault injection main **drawback**
in practice
→ combinatory complexity of
its parameters.

Aim:

How to decrease it?

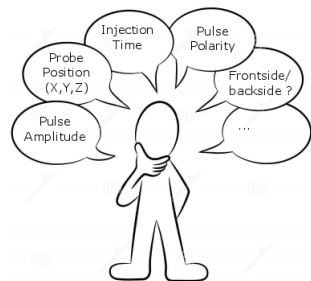


Figure: EMFI parameters example



LIRMM



💡 Using injection channel for analysis

- ▶ EMFI: (*EM emission*)
 - + **spatial.**
 - + temporal.
 - + same setup → no extra cost.

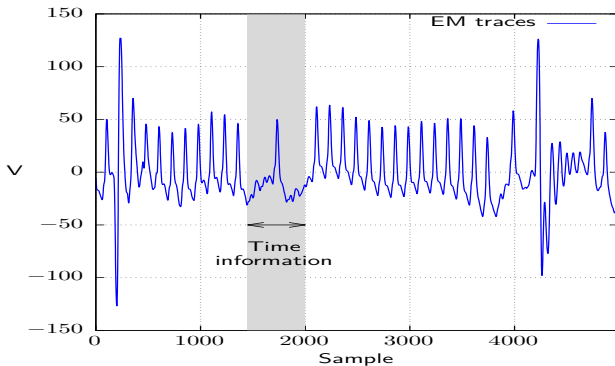


Figure: Example EM traces at X,Y position

Benefits of binding EM emission to injection

Time efficiency¹:

Analysis map → **8 hours**.

Injection map (**fixed parameters**)

→ **three days** × exhaustive search ...



LIRMM



¹ timing are relative to our setup

Table of Contents

Criterion principles

EMFI hotspots definition

Designing the criterion

Results

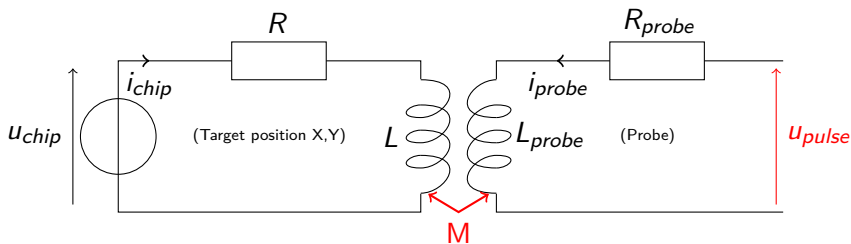
Conclusion



LIRMM



EM coupling



Coupling: (*injection case*)

$$u_{chip} = Ri_{chip} + L \frac{di_{chip}}{dt} + M \frac{di_{probe}}{dt}$$

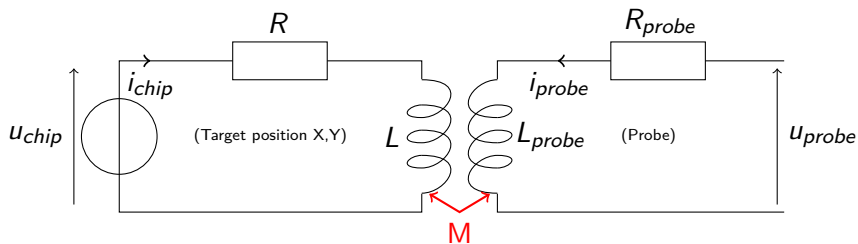


LIRMM



life.augmented

EM coupling



Coupling: (*analysis case*)

$$u_{probe} = R i_{probe} + L_{probe} \frac{di_{probe}}{dt} + M \frac{di_{chip}}{dt}$$

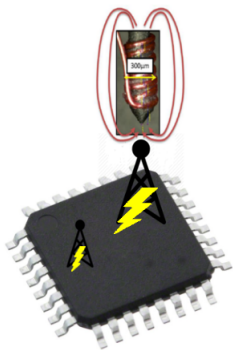


LIRMM



life.augmented

Antenna reciprocity



Antenna reciprocity:

Antenna's receiving efficiency is as important as its transmitting efficiency.

Conclusion:

Finding high emission antennas

→ best coupling positions on circuits.

Warning:

High emission antenna \neq best entry point

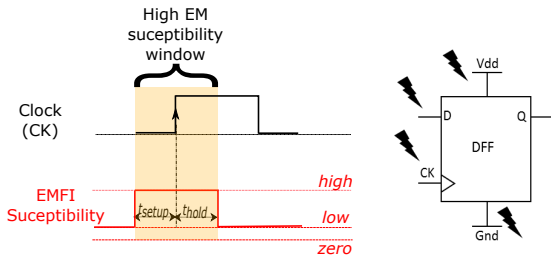
→ not necessarily linked to algorithm.



LIRMM



Sampling fault model²



Target:

- ▶ DFF are more likely to be faulted by EM injection.
- ▶ Target events occurring at f_{CK} .



LIRMM



²EM injection: fault model and locality S. Ordas, L.Guillaume-Sage, P. Maurinne FDTC 2015.

EMFI Criterion definition

Area to target are positions:

- ▶ (*guideline 1*) emitting the strongest signal (in terms of power) associated to the clock signal or clock tree.
 → tool: Power Spectral Density $PSD(f_{CK})$
- ▶ (*guideline 2*) emitting signal tightly bind to both targeted algorithm and clock frequency (f_{CK}).
 → tool: **incoherence**(f_{CK})



LIRMM



Guideline 2 tools:

$$inc_{s_1, s_2}(f) = 1 - \frac{psd_{s_1, s_2}(f)^2}{psd_{s_1, s_1}(f) \cdot psd_{s_2, s_2}(f)}$$

Notation:

s_1 = EM emission for input 1.

s_2 = EM emission for input 2.

$inc_{s_1, s_2}(f)$ = incoherence between s_1 and s_2 at f .

Aim

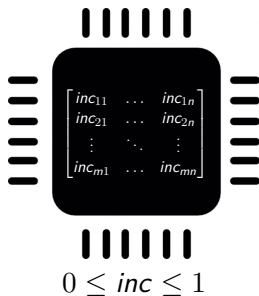
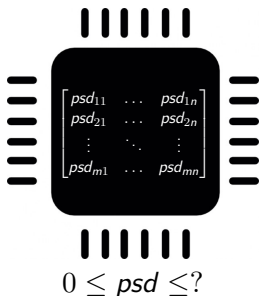
- ▶ Look for differences in spectrum occurring at f_{CK} i.e. DFF used by algorithm.



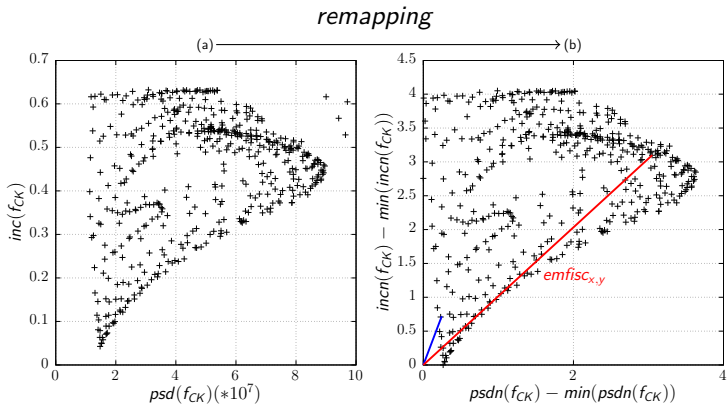
LIRMM



How to combine and weight those two measures?



Raw data: PSD, Incoherence view



LIRMM



life.augmented

EMFISC's degrees of freedom

1. Weight psd and incoherence:

$$emfisc_{x,y} = \sqrt{(1 - a) * psdn_{x,y}^2 + a * incn_{x,y}^2}$$

2. Percent of the chip area rejected for injection (α):

$$quantile(emfisc_{x,y}, \alpha)$$

Table of Contents

Criterion principles

EMFI hotspots definition

Designing the criterion

Results

Conclusion



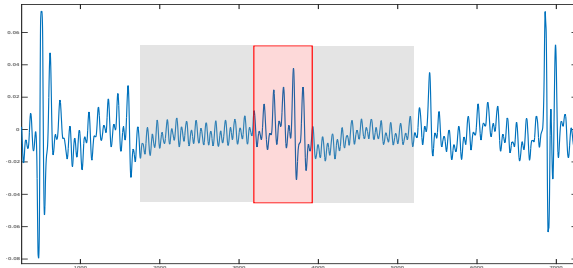
LIRMM



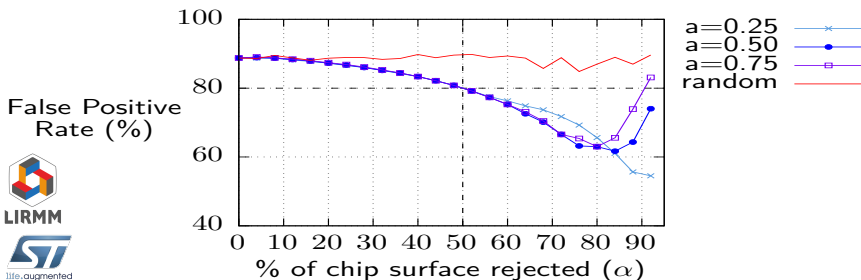
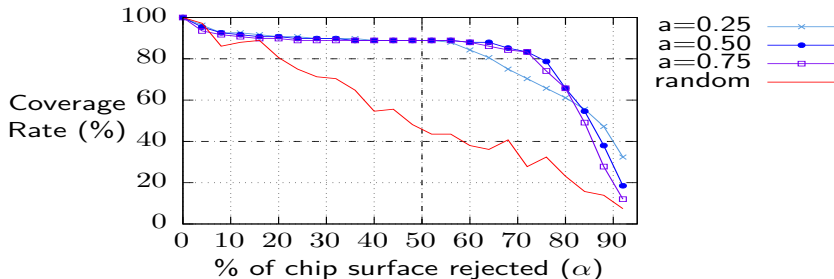
Target algorithm:

Algorithm 1 Pattern (AddrSRAM1 (R0), AddrSRAM2 (R1))

- 1: ADD R0,R0,#0; 11 times
 - 2: LDR R2,[R0]; read AddrSRAM1
 - 3: STR R2,[R1]; write at AddrSRAM2
 - 4: LDR R3,[R1]; read back AddrSRAM2
 - 5: ADD R0,R0,#0; 11 times
-



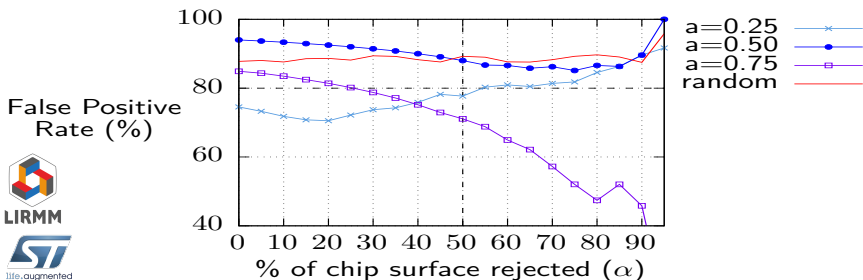
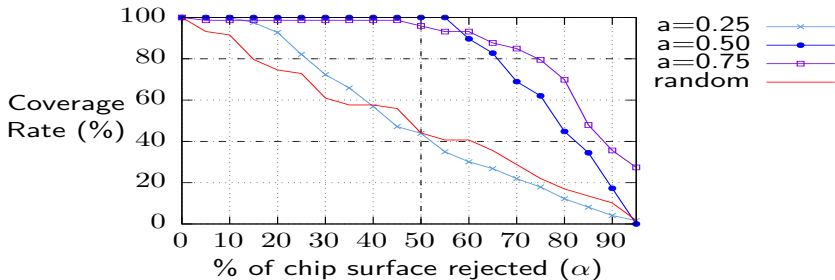
EMFISC figures of merit (target A pulse: 198V)



False Positive Rate (%)



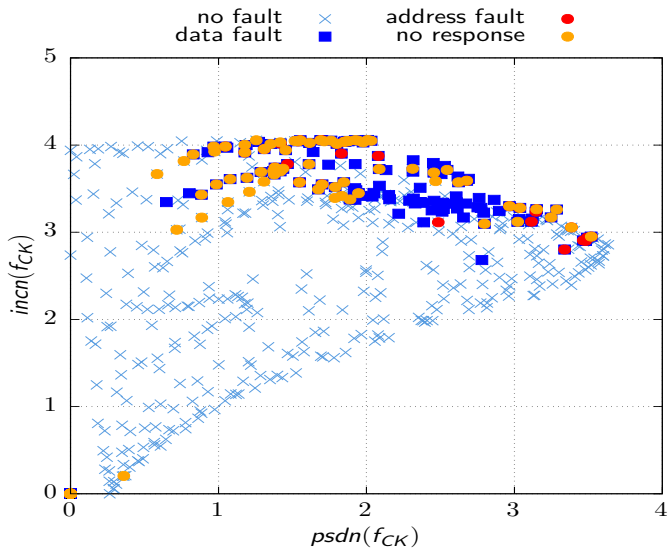
EMFISC figures of merit (target B pulse: 130V)



False Positive Rate (%)



Fault repartition (target 1)



Results:

- ▶ There is a link between EM emissions and EMFI.
- ▶ This link can be used to ease EMFI characterisation.

Refining the criterion:

- ▶ Try the method on masked implementation
- ▶ Adding a criterion more target specific, such as a better measurement of M parameter (measure of Vdd drop).
- ▶ Add some strategy instead of doing a full EM analysis map
→ genetic algorithm?

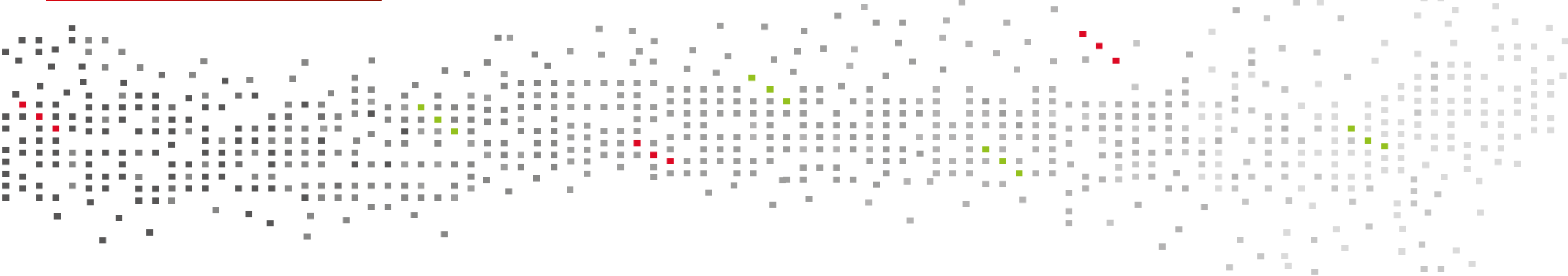


LIRMM



Thanks
Any questions?





BACKSIDE SHIELD BASED ON PACKAGING TECHNOLOGIES FOR CHIPS OR SYSTEMS SECURING

- 1** Context
- 2** Backside protection concept
- 3** New implementations
- 4** Wafer level integration
- 5** Conclusion

- **Hackers**

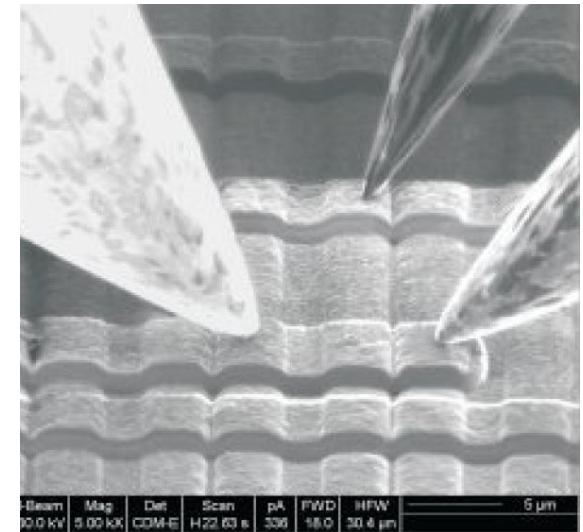
- Clever Outsiders
- Knowledgeable Insiders
- Funded Organizations

- **Physical attacks**

- Non-invasive (side channel)
- Semi-invasive (fault injection)
- Invasive (reverse engineering, microprobing)

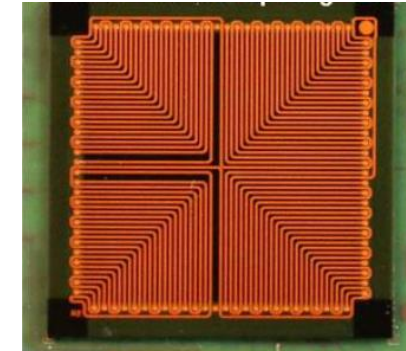
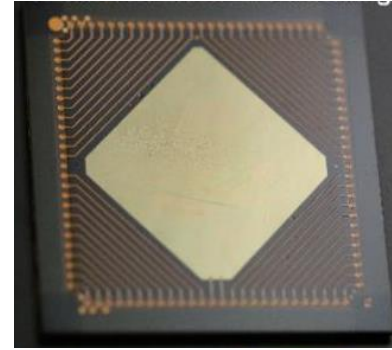
- **Secure components**

- Security level adapted to the threat



• Prevent

- Passive shield



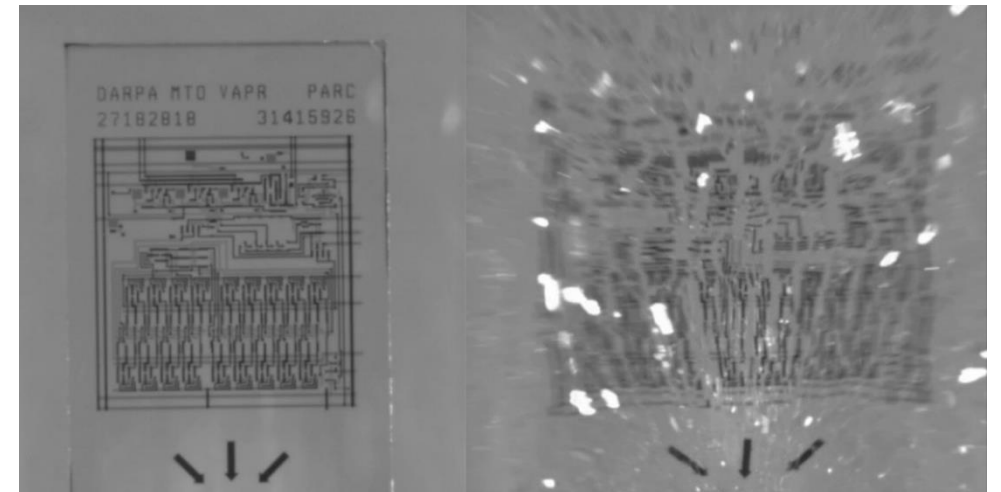
• Detect

- Active shield
- IR / UV / X-rays photodiodes
- Temperature sensors
- ...

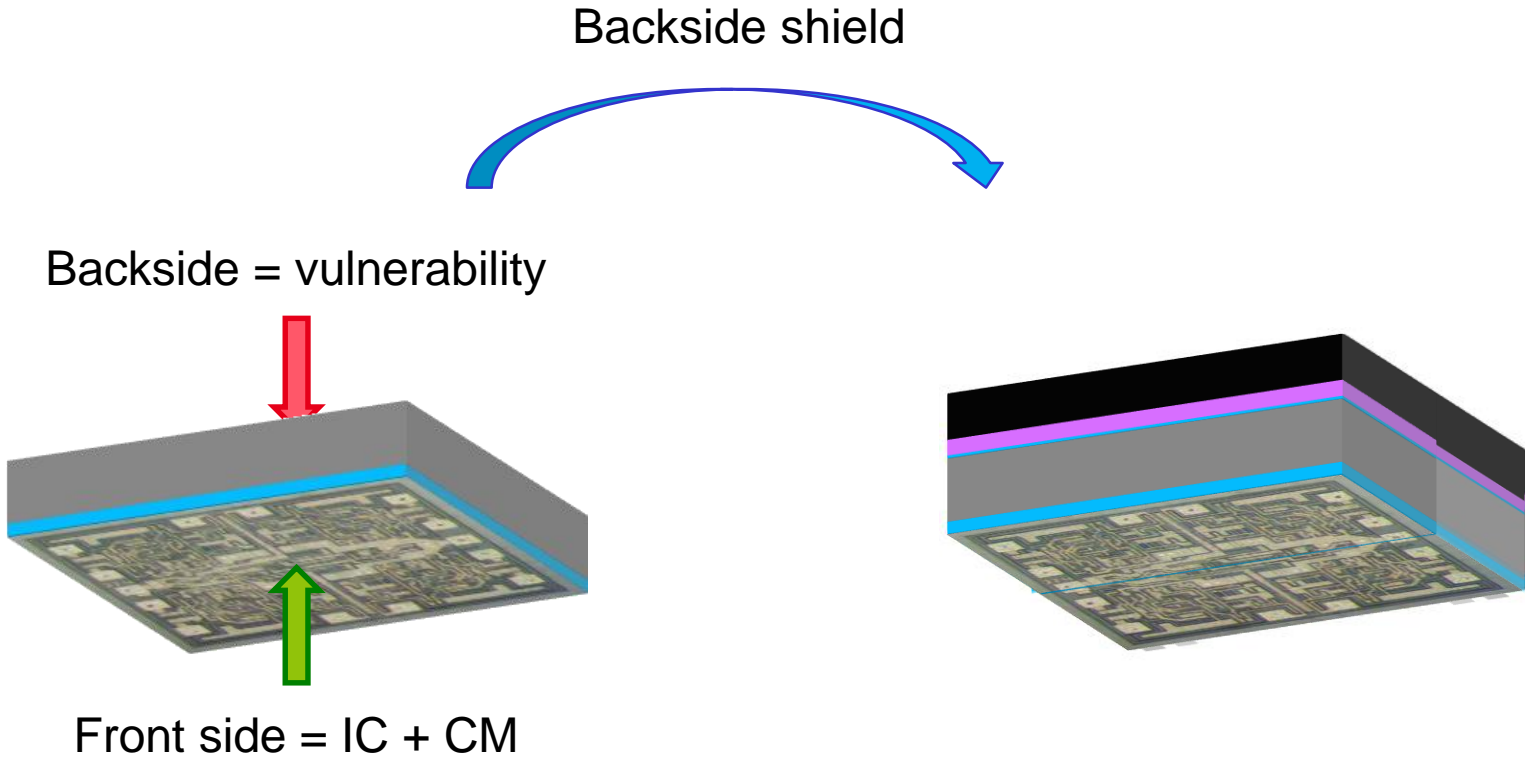


• Trigger a reaction

- Erase the data
- Deactivate the chip
- Destruct the chip physically

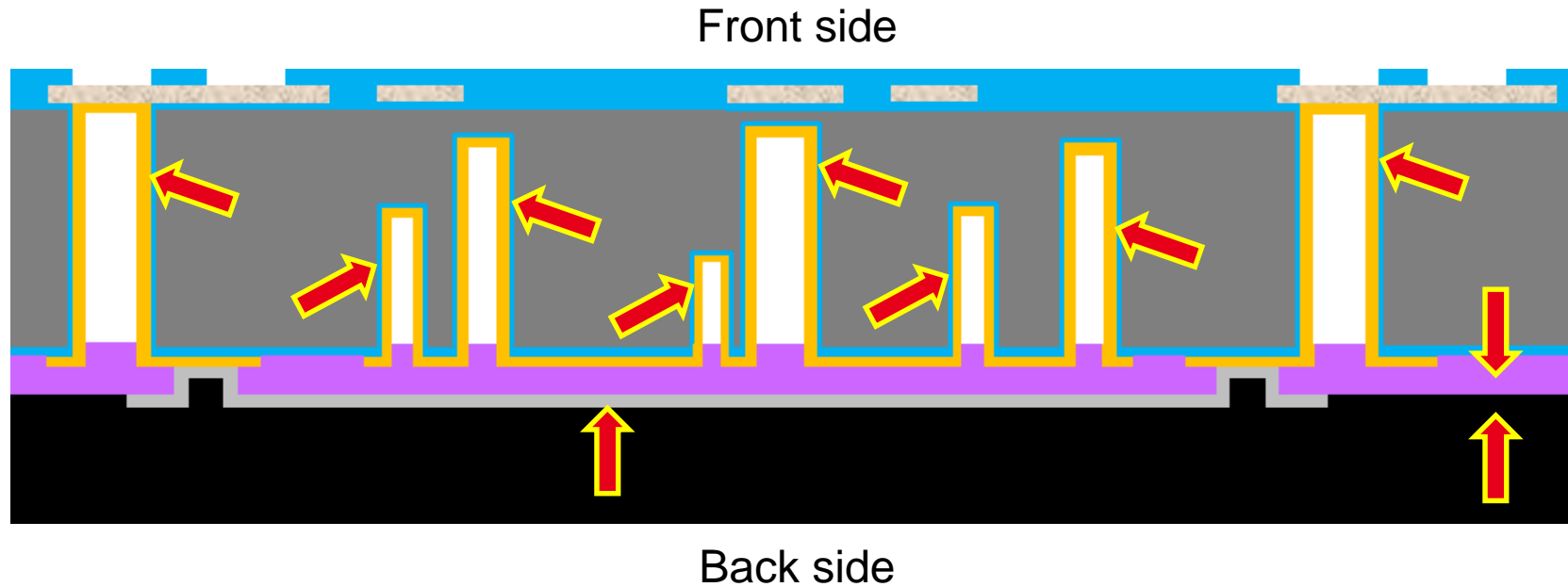


CONTEXT : COUNTERMEASURES



- 1 Context
- 2 **Backside protection concept**
- 3 New implementations
- 4 Wafer level integration
- 5 Conclusion

BACKSIDE PROTECTION : CONCEPT



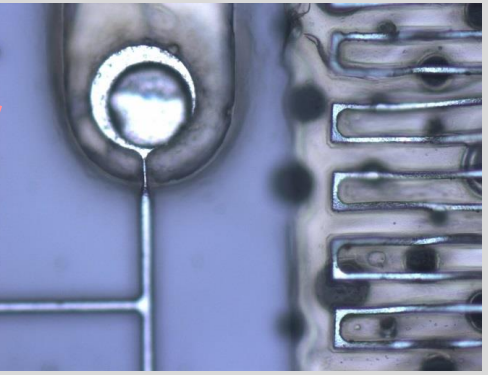
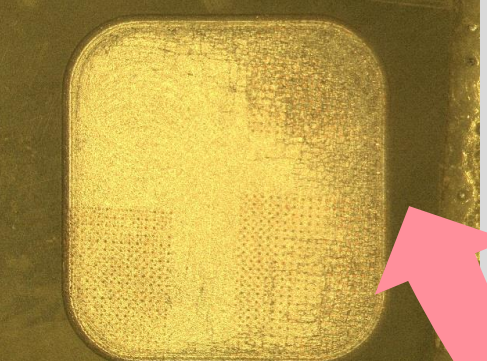
- ✓ Metal serpentine (attack witness)
- ✓ TSVs (for integrity check by the circuit)
- ✓ Blind holes (weakening + optical shield)
- ✓ Polymers (laser, FIB, chemicals)

BACKSIDE PROTECTION : DEMONSTRATOR + EVALUATION

μ milling

↓

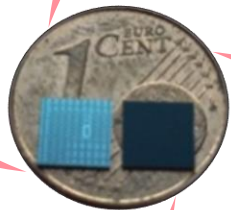
Cracks in the Si crystal



Molding polymer
chemical removal

↓

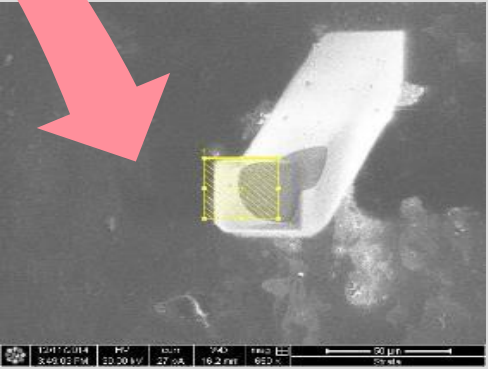
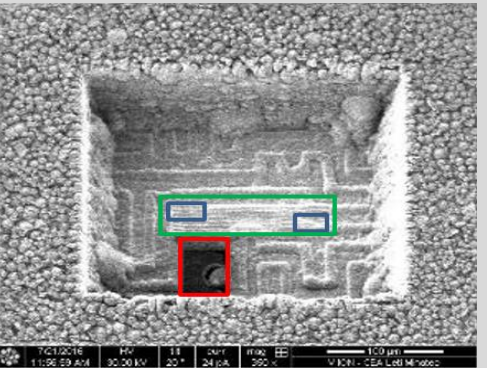
Serpentine
breakage



PFIB +
circuit edit

↓

Increase of R



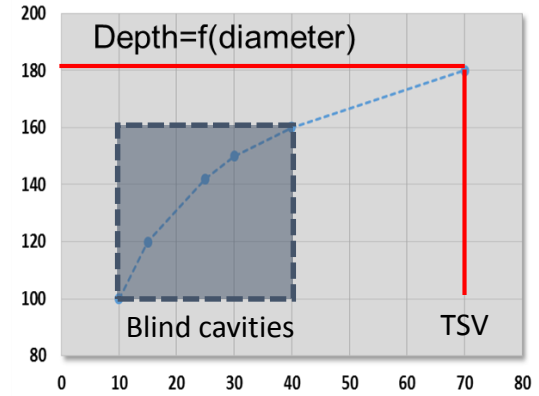
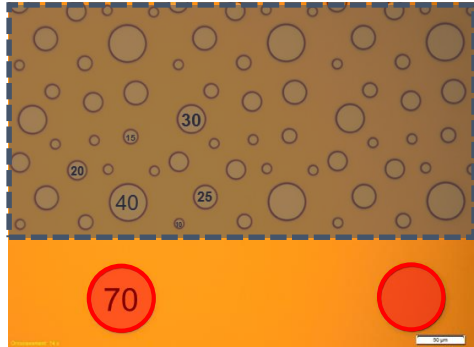
Local removal
by FIB

↓

Drift of the beam

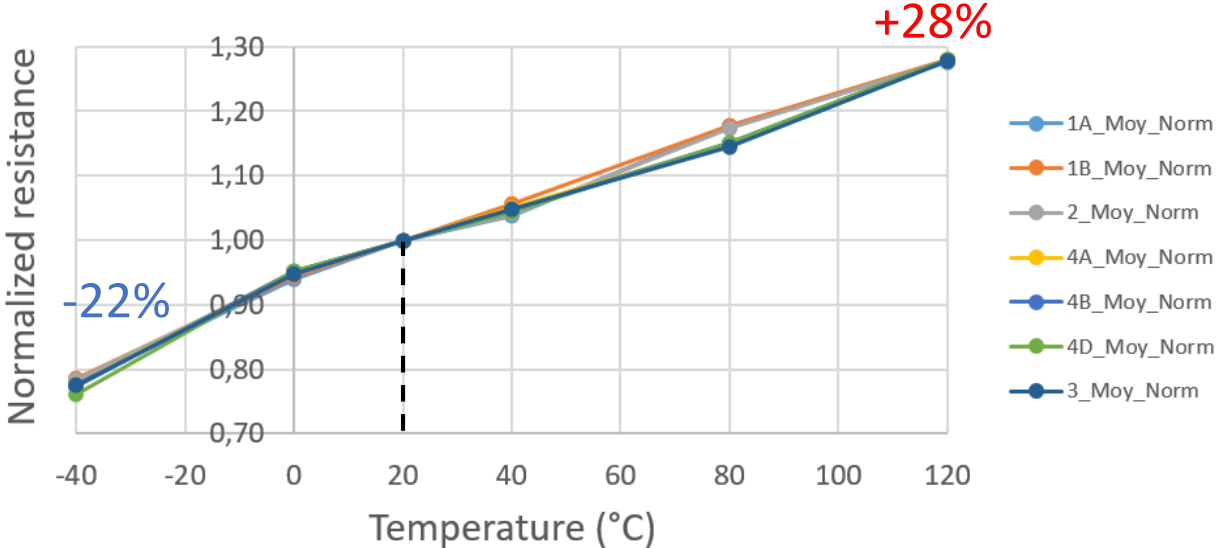
BACKSIDE PROTECTION : LIMITATIONS

- TSV must be hidden
 - Largest patterns ($\varnothing 70\mu\text{m}$) can be identified (but they are covered with a black polymer...)



- Complexity is mandatory
 - Limited for a 2D serpentine \rightarrow “easy” to shunt

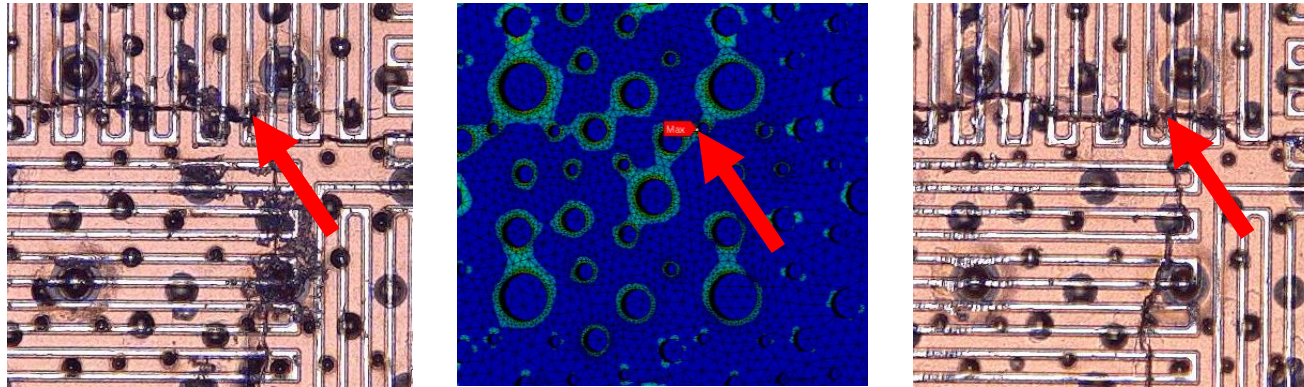
- R measurement vs. T
 - ΔT can be considered as an attack



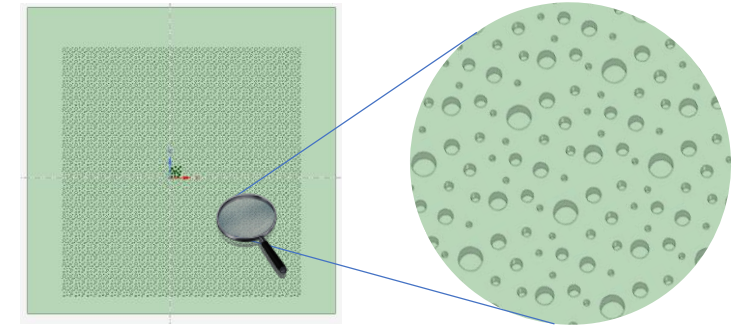
- 1 Context
- 2 Backside protection concept
- 3 New implementations**
- 4 Wafer level integration
- 5 Conclusion

• Weakening structures (1/2)

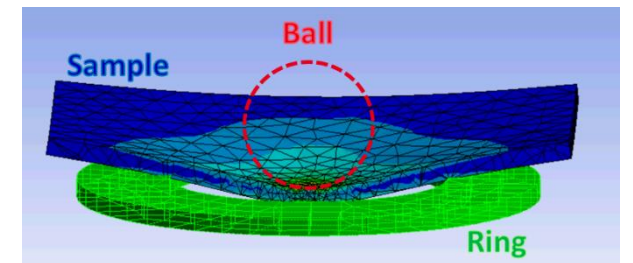
- FEM of blind holes with different diameters (and depths)
- Maximum of tensile stress (in a Ball on Ring configuration)



- Good correlation with fractures initialization
- Prominent role of the smallest structures (Ø10µm)
- ... but they are also the shallowest

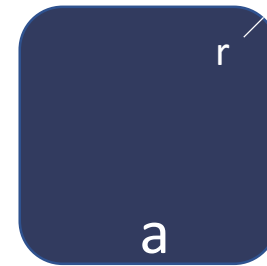
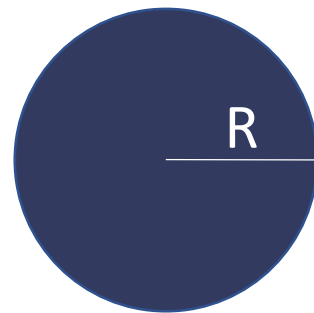


$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{xz} \\ \sigma_{xy} \end{bmatrix} = \begin{bmatrix} 166 & 64 & 64 & 0 & 0 & 0 \\ 64 & 166 & 64 & 0 & 0 & 0 \\ 64 & 64 & 166 & 0 & 0 & 0 \\ 0 & 0 & 0 & 80 & 0 & 0 \\ 0 & 0 & 0 & 0 & 80 & 0 \\ 0 & 0 & 0 & 0 & 0 & 80 \end{bmatrix} \begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ \epsilon_{yz} \\ \epsilon_{xz} \\ \epsilon_{xy} \end{bmatrix}$$



- Weakening structures (2/2)

- Large patterns (for depth)
- Small radius of curvature (for stress concentration)



$$S = \pi R^2$$

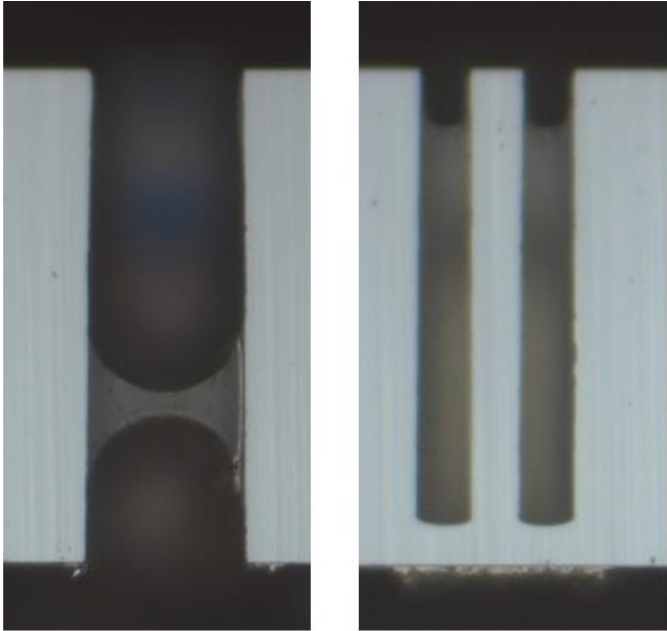
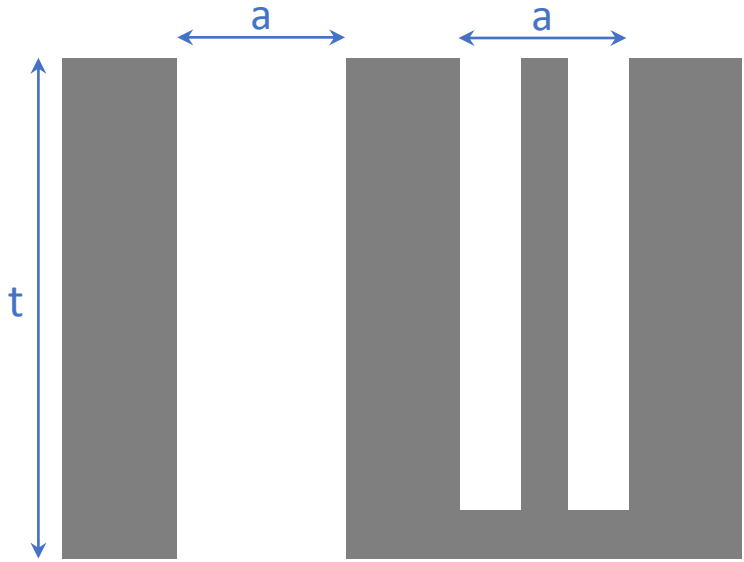
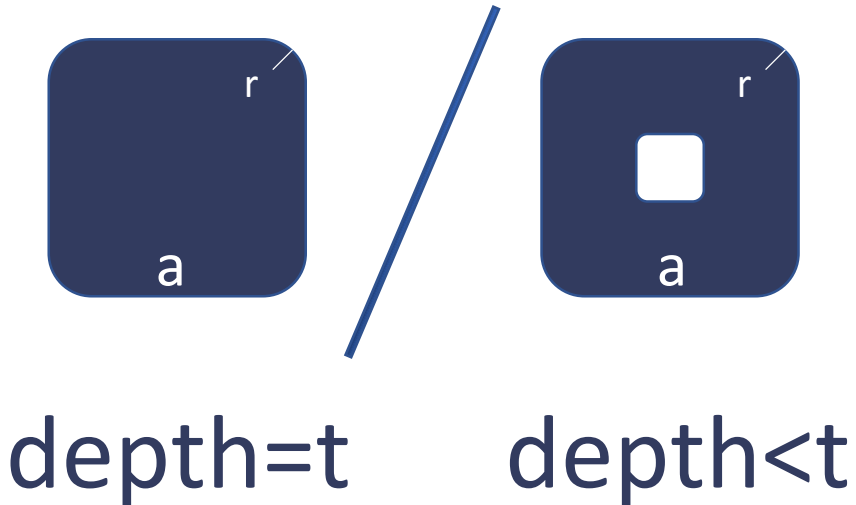
$$\sigma = f(R)$$

$$S' = a^2$$

$$\sigma = f(r)$$

• **Indistinguishable TSV (1/2)**

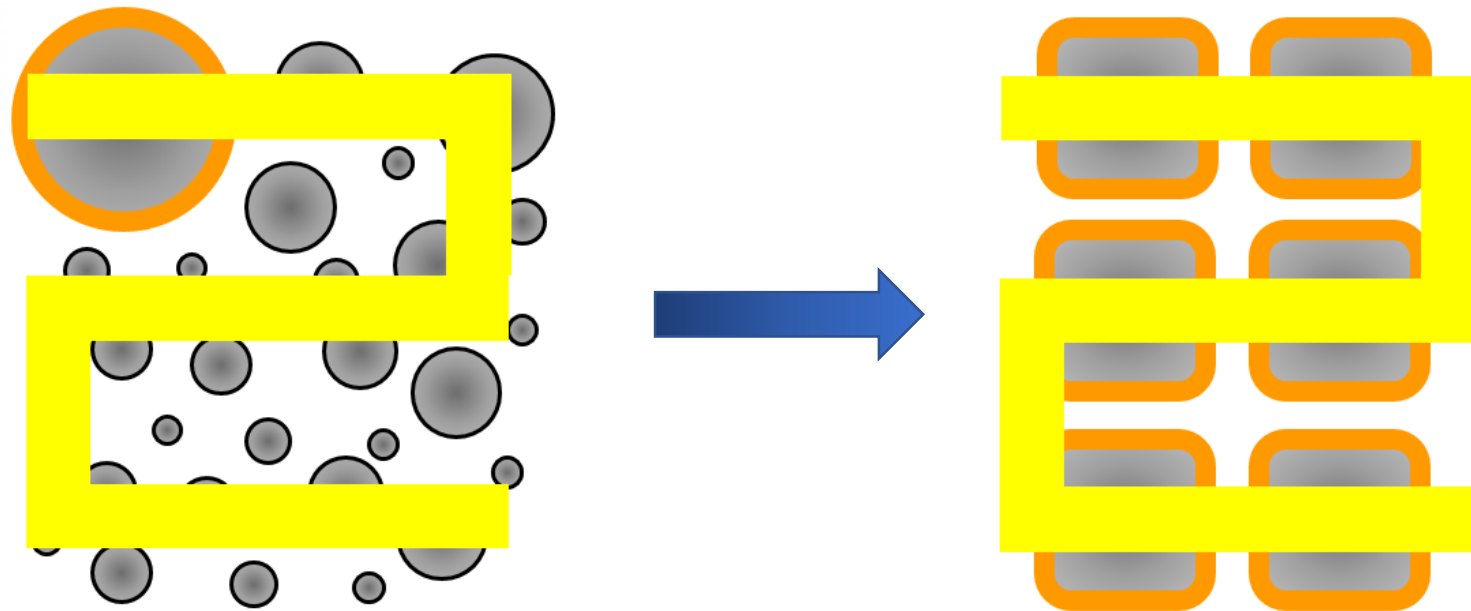
- Deeper patterns (access to the front side) with same aspect
- No additional process step



→ Necessity to hide this pillar

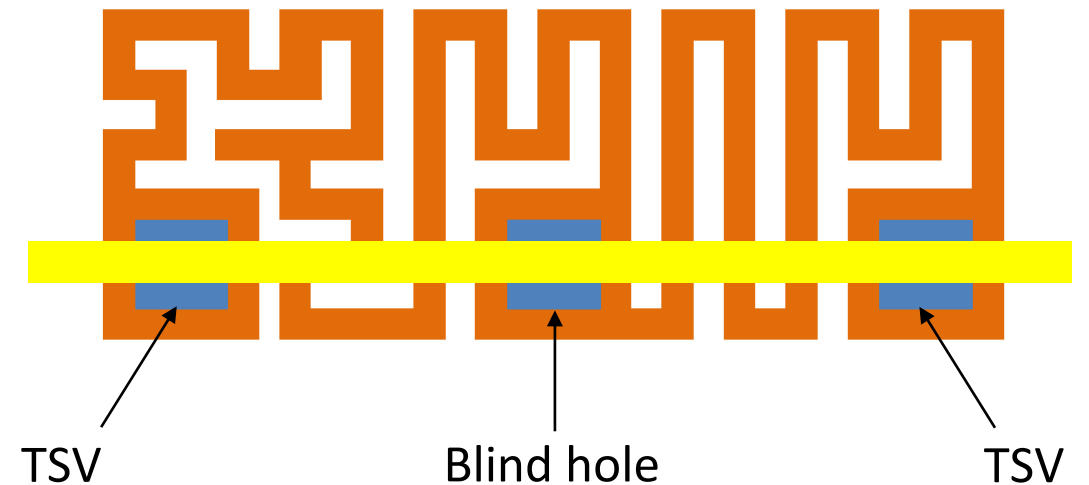
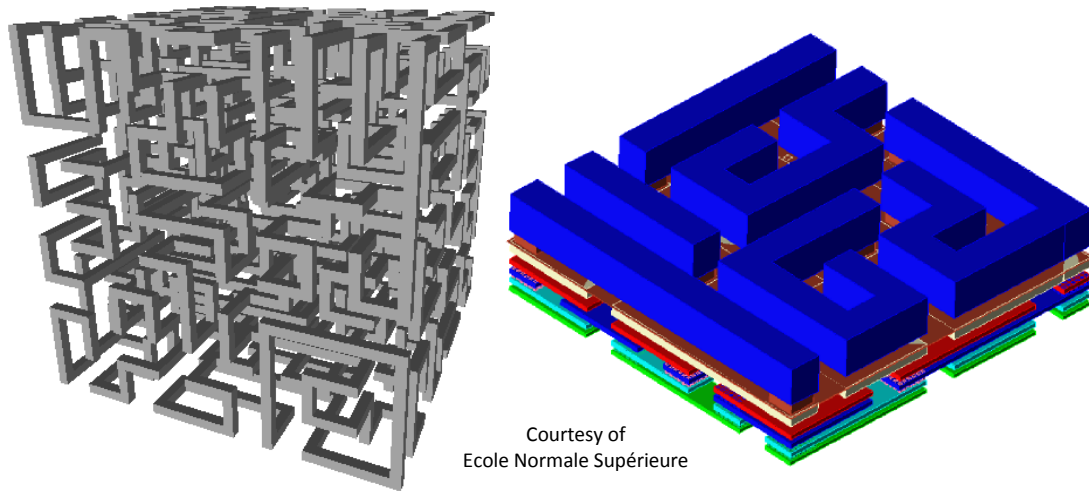
- **Indistinguishable TSV (2/2)**

- The serpentine is used to cover the pillars
- A novel distribution is considered for the blind holes (grid-like arrangement)
- Blind holes must be plated as well



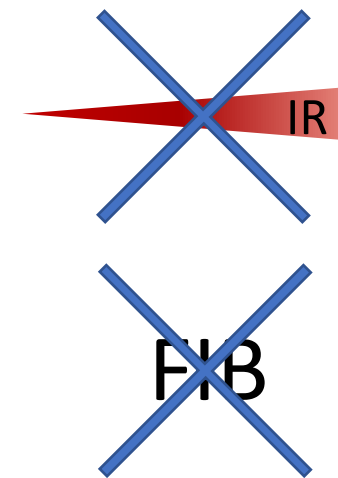
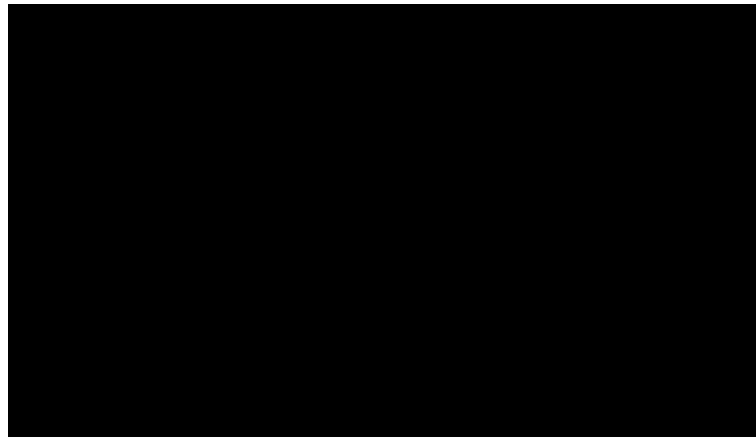
• Multilayer serpentine (1/2)

- Complexity can be drastically increased with a 3D serpentine
- A 1st level of serpentine is obtained by patterning the seed layer (+ Cu plating)
- Confusion is increased by making the serpentine transit through blind holes



- **Multilayer serpentines (2/2)**

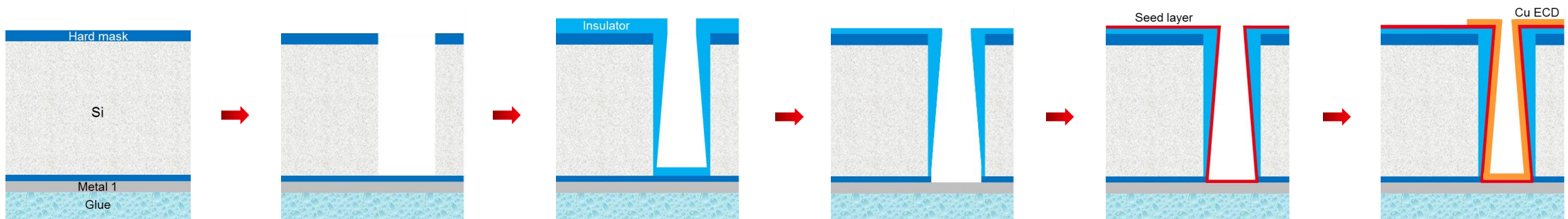
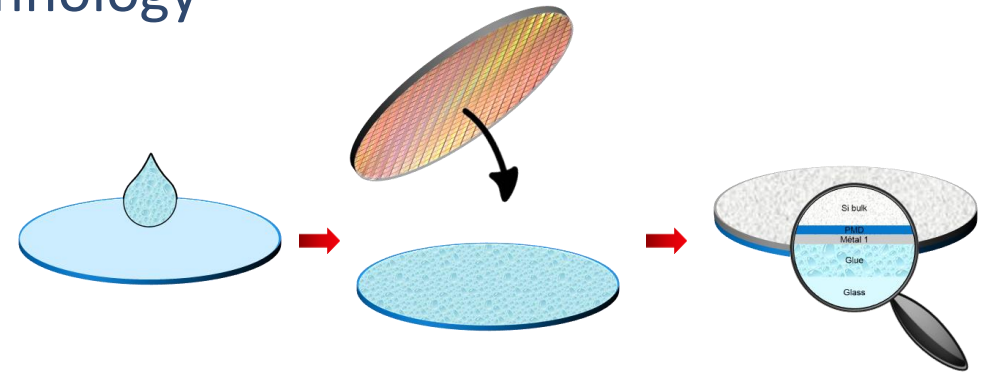
- The 2nd level of serpentine is unchanged in terms of integration, but its design is “free”
- A second serpentine (for independence with T) can be intermingled
- The hacker cannot identify the serpentines through the molding polymer
- The serpentines can have a different patterns on each die of a wafer



- 1 Context
- 2 Backside protection concept
- 3 New implementations
- 4 Wafer level integration**
- 5 Conclusion

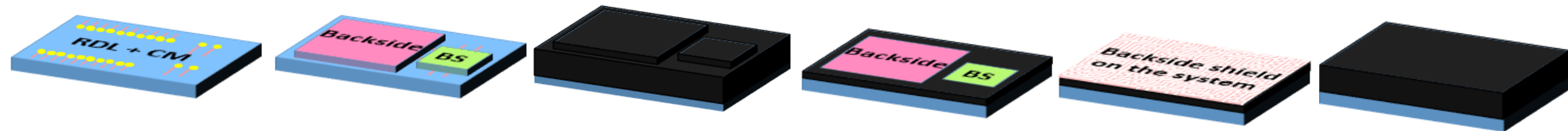
• Chip scale

- Generic brick that is compatible with any technology
- Processes used for 3D integration :
 - Glass carrier, temporary bonding
 - TSV last + BRDL processes
 - Encapsulation molding (lamination)



- System scale

- Use of a smart interposer with embedded countermeasures
- Flip chip of several (non-secured) dies
- Front sides are protected by the interposer
- Molding
- Backside process
- Back sides are collectively protected by the shield



- 1** Context
- 2** Backside protection concept
- 3** New implementations
- 4** Wafer level integration
- 5** Conclusion

- Backside needs to be protected
- A shield based on packaging technologies (3D integration) is proposed
 - Compatible with any technology (bulk / SOI)
 - Wafer level
 - Effective against fault injection / probing
 - New implementations to be evaluated
- Secure SiP
 - Smart interposer with countermeasures
 - Non-secured dies flip-chipped on it
 - Front sides are protected by the CM embedded in the interposer
 - Backside shield : backsides are protected collectively

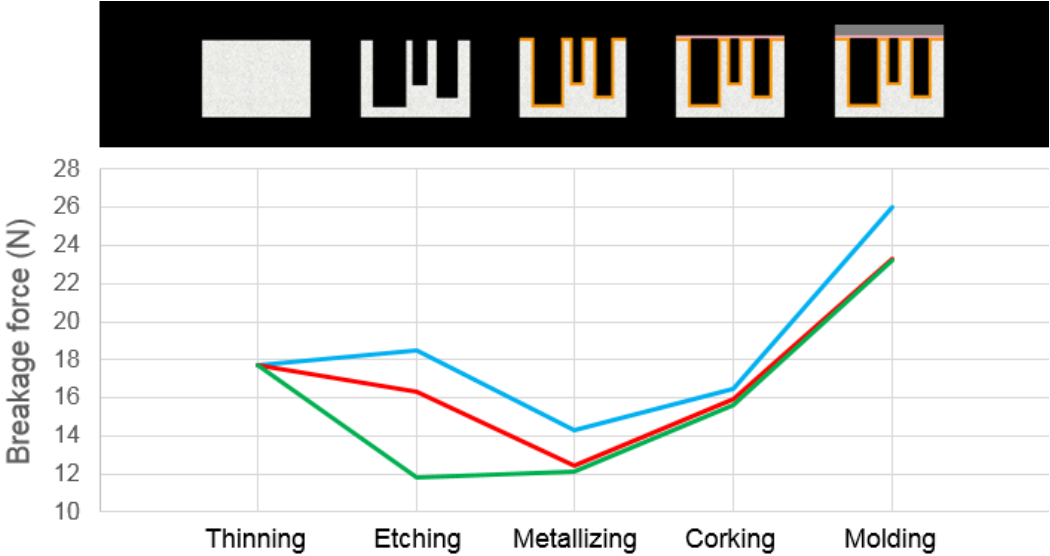
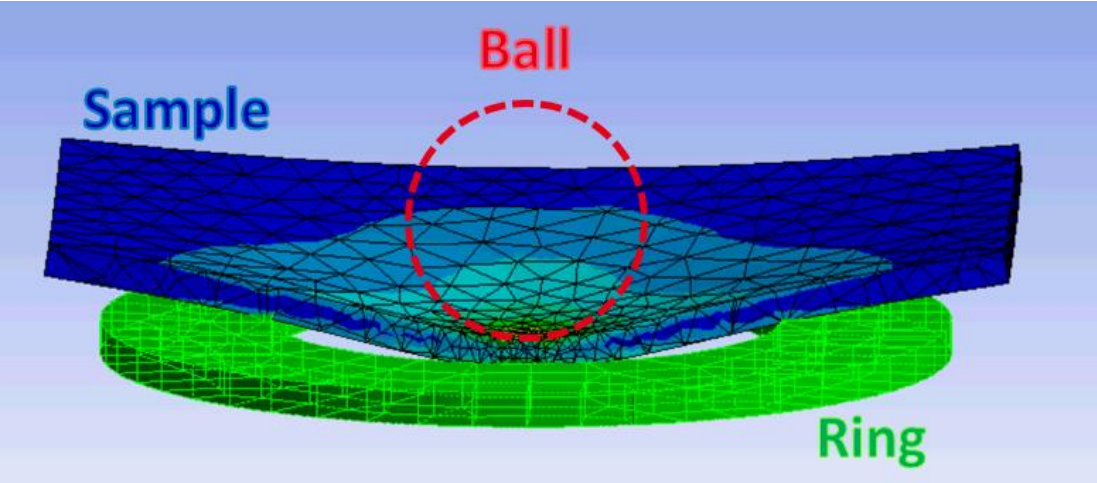
Thanks to the contributors :

Lucas DUPERREX
Edouard DESCHASEAUX
Jean CHARBONNIER
Philippe MEDINA
Stéphanie ANCEAU
Jessy CLEDIERE
Romain WACQUEZ
Jacques FOURNIER
Eric JALLAGUIER
Christophe PLANTIER
Gilles SIMON
Alain MERLE
Bruno CHARRAT

Thank you for your attention

Any questions ?

stephan.borel@cea.fr



BACKUP SLIDES

