



HAL
open science

Sécurité des moyens de test des SoC

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

► **To cite this version:**

Mathieu da Silva, Marie-Lise Flottes, Giorgio Di Natale, Bruno Rouzeyre. Sécurité des moyens de test des SoC. Journée thématique des GDR SoC² et Sécurité Informatique: Sécurité des SoC complexes hétérogènes – de la TEE au matériel, Sep 2018, Paris, France. 2018. lirmm-01882552

HAL Id: lirmm-01882552

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-01882552>

Submitted on 27 Sep 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



SÉCURITÉ DES MOYENS DE TEST DES SoC

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

Journée thématique des GDR SoC² et Sécurité Informatique

Sécurité des SoC complexes hétérogènes – de la TEE au matériel

PROJET TEEVA

- Travaux réalisés dans le cadre du projet TEEVA:
Trusted Environment Execution eVALuation

- Partenaires



TRUSTONIC



**LABORATOIRE
HUBERT CURIEN**
UMR • CNRS • 5516 • SAINT-ETIENNE



SUMMARY

- 1) Context of testing
- 2) Threats related to the test infrastructures
- 3) Proposed countermeasures: Scan Encryption
- 4) Application of the proposed countermeasures
- 5) Conclusion



SUMMARY

1) Context of testing

- Design-for-Testability (DfT)
- Test standards

2) Threats related to the test infrastructures

3) Proposed countermeasures: Scan Encryption

4) Application of the proposed countermeasures

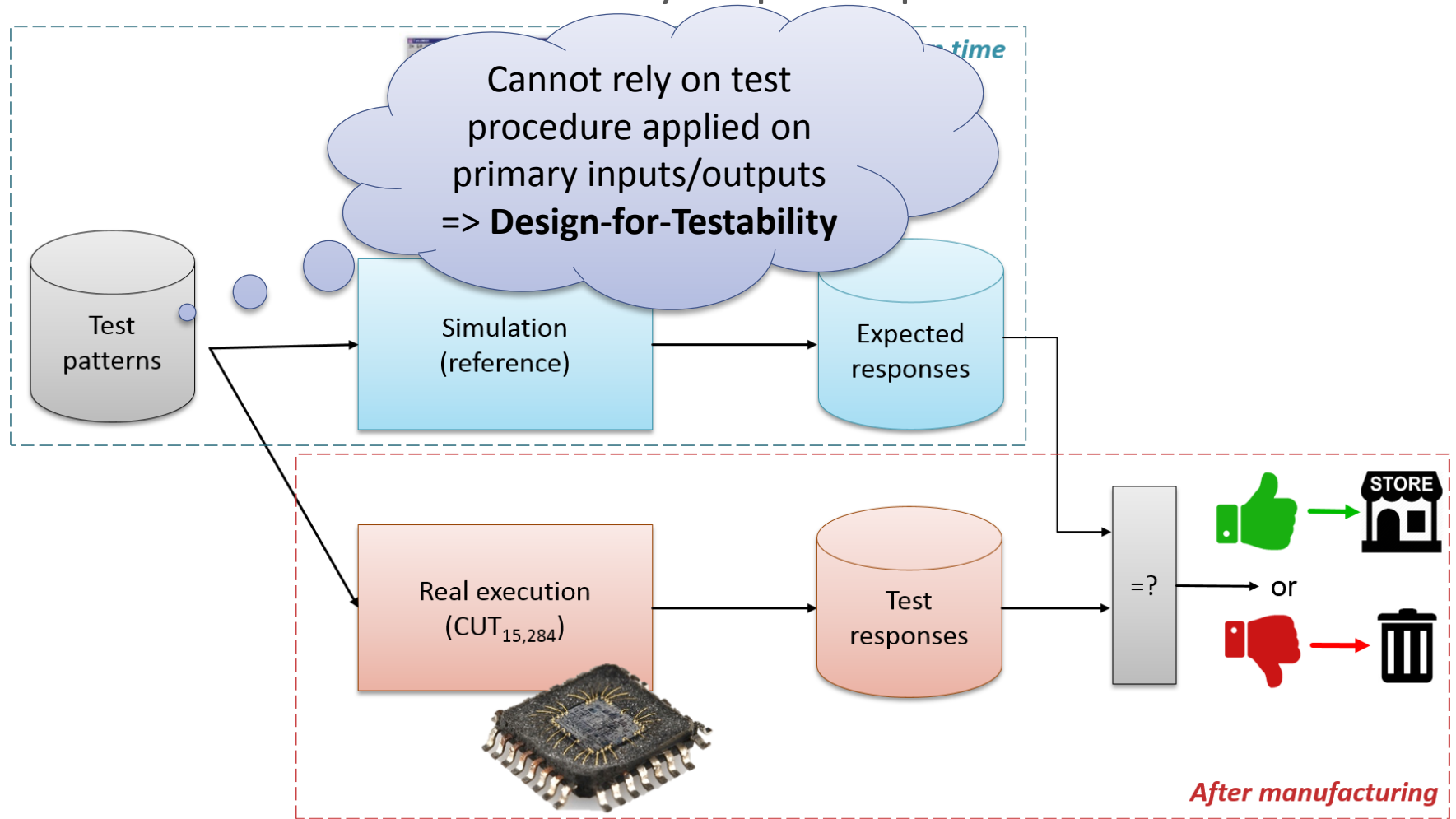
5) Conclusion



CONTEXT OF TESTING

- DESIGN-FOR-TESTABILITY (DFT)
- TEST STANDARDS

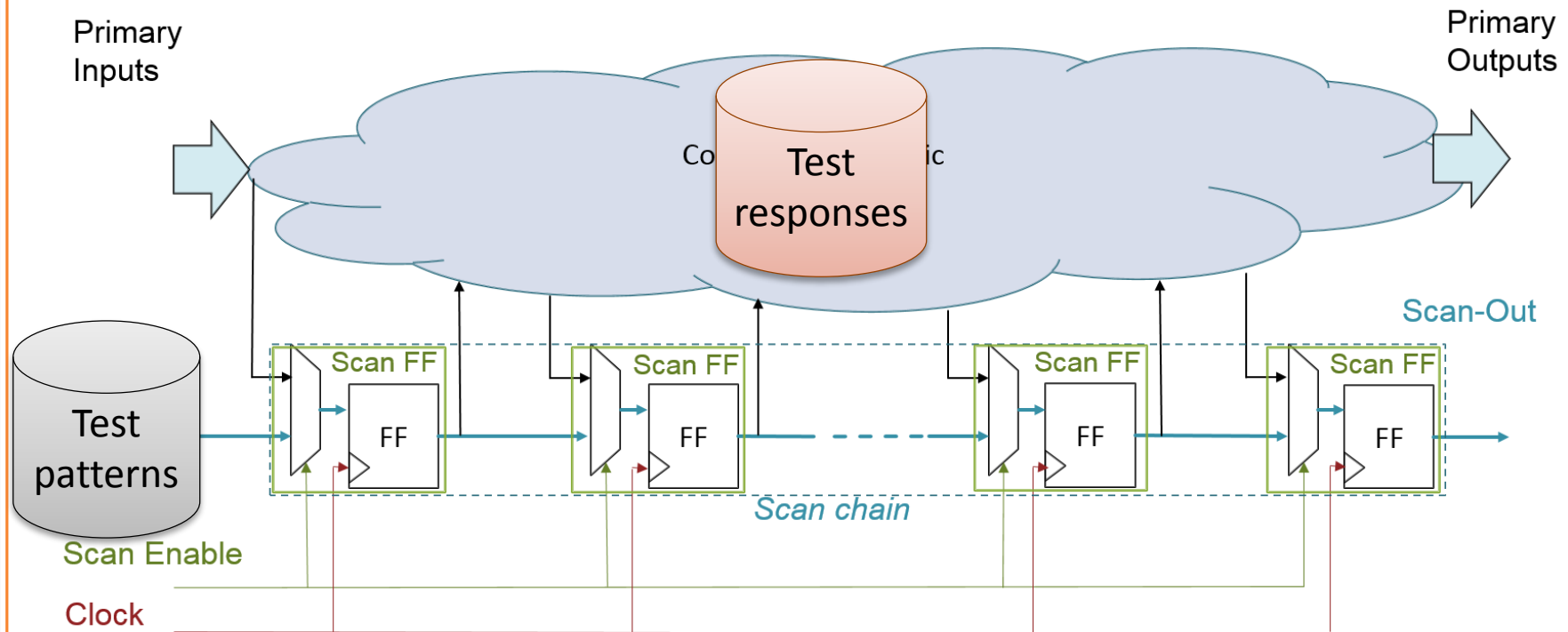
- Test of circuit is a mandatory step in IC production



SCAN CHAINS

- DESIGN-FOR-TESTABILITY (DFT)
- TEST STANDARDS

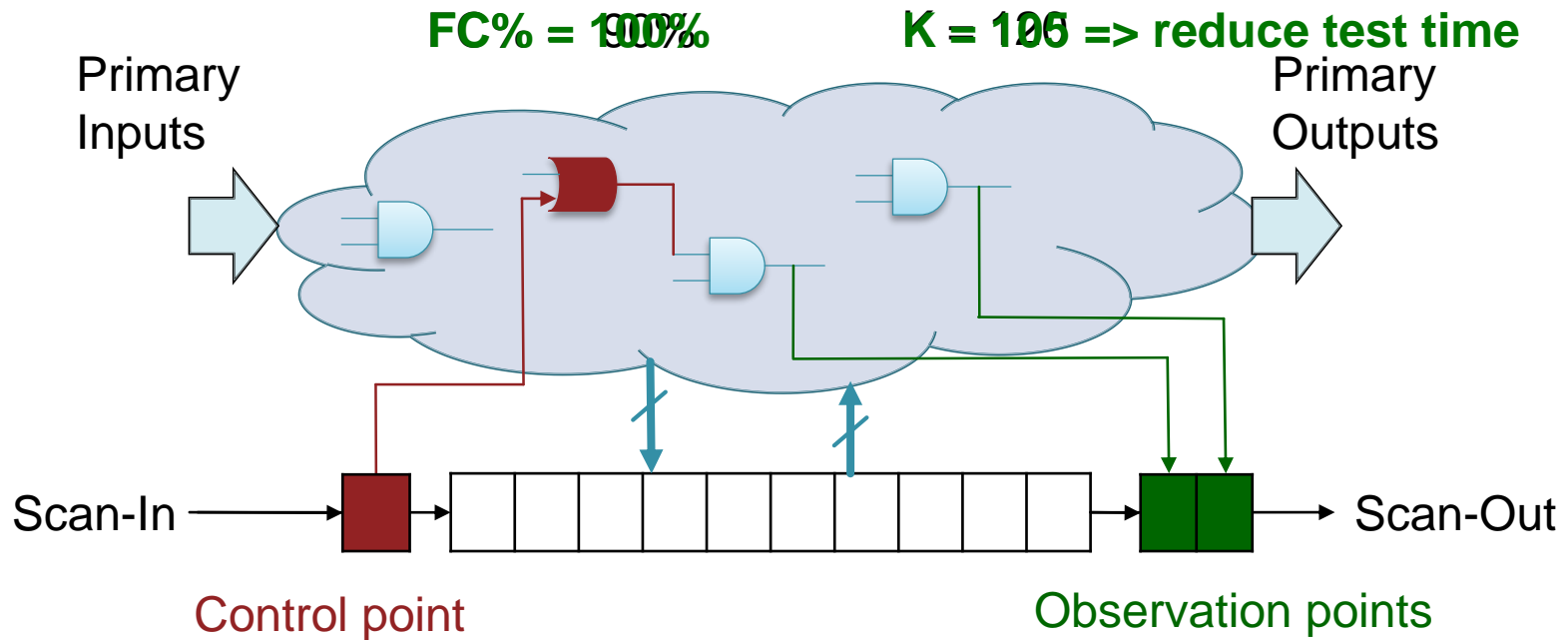
- Most popular method for Design-for-Test = Scan chains
 - Replace original FF by Scan FF connected serially together
 - Extra port « Scan-In » => total control on internal states
 - Extra port « Scan-Out » => total observation on internal states



INSERTION OF TEST POINTS

- DESIGN-FOR-TESTABILITY (DFT)
- TEST STANDARDS

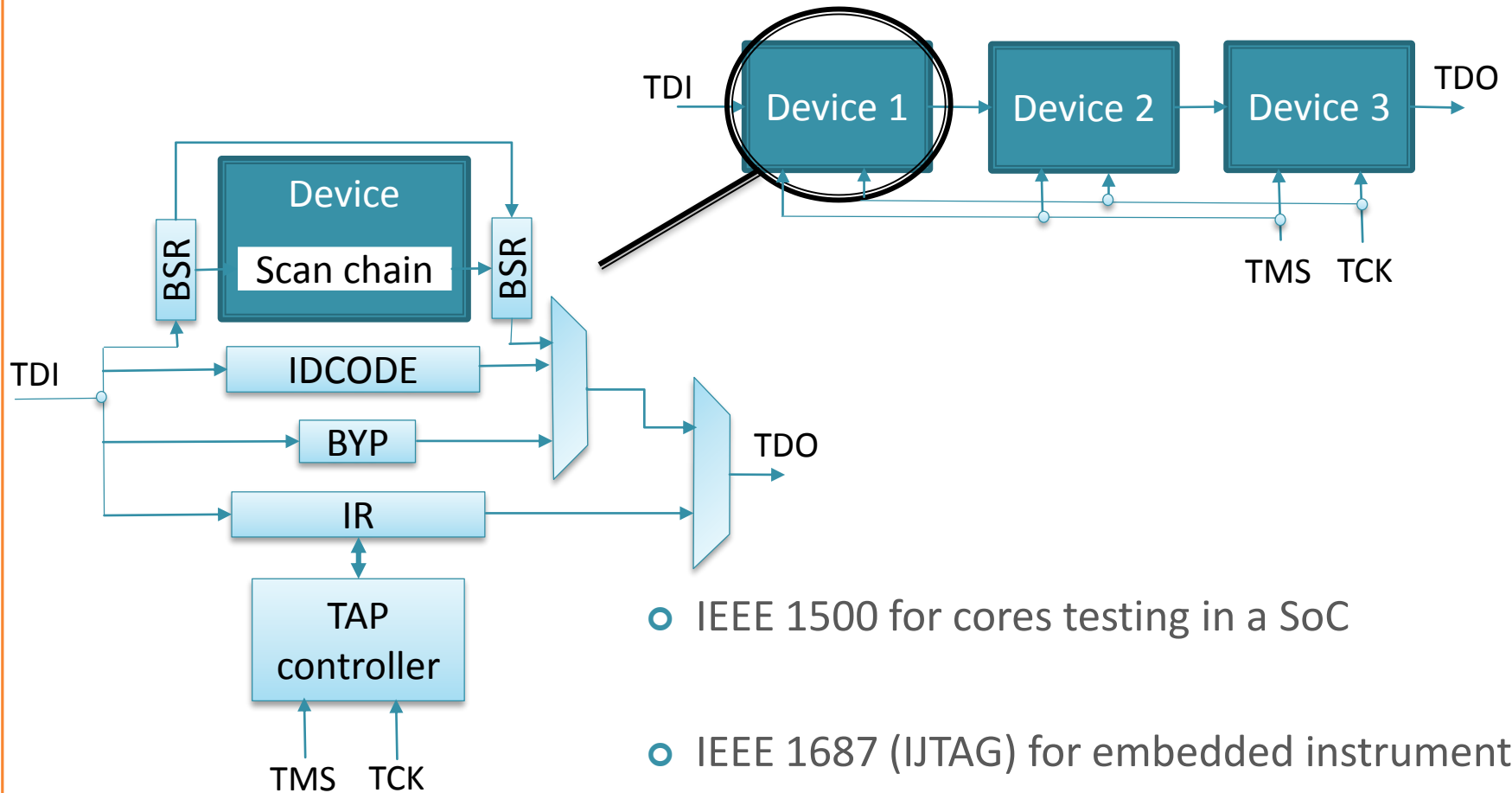
- Extra-DfT: insertion of test points
- Goal: increase the fault coverage FC% and/or reduce the number of patterns K



TEST STANDARDS

- DESIGN-FOR-TESTABILITY (DFT)
- TEST STANDARDS

- IEEE 1149 (JTAG) for board testing + diagnosis & debug facilities



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



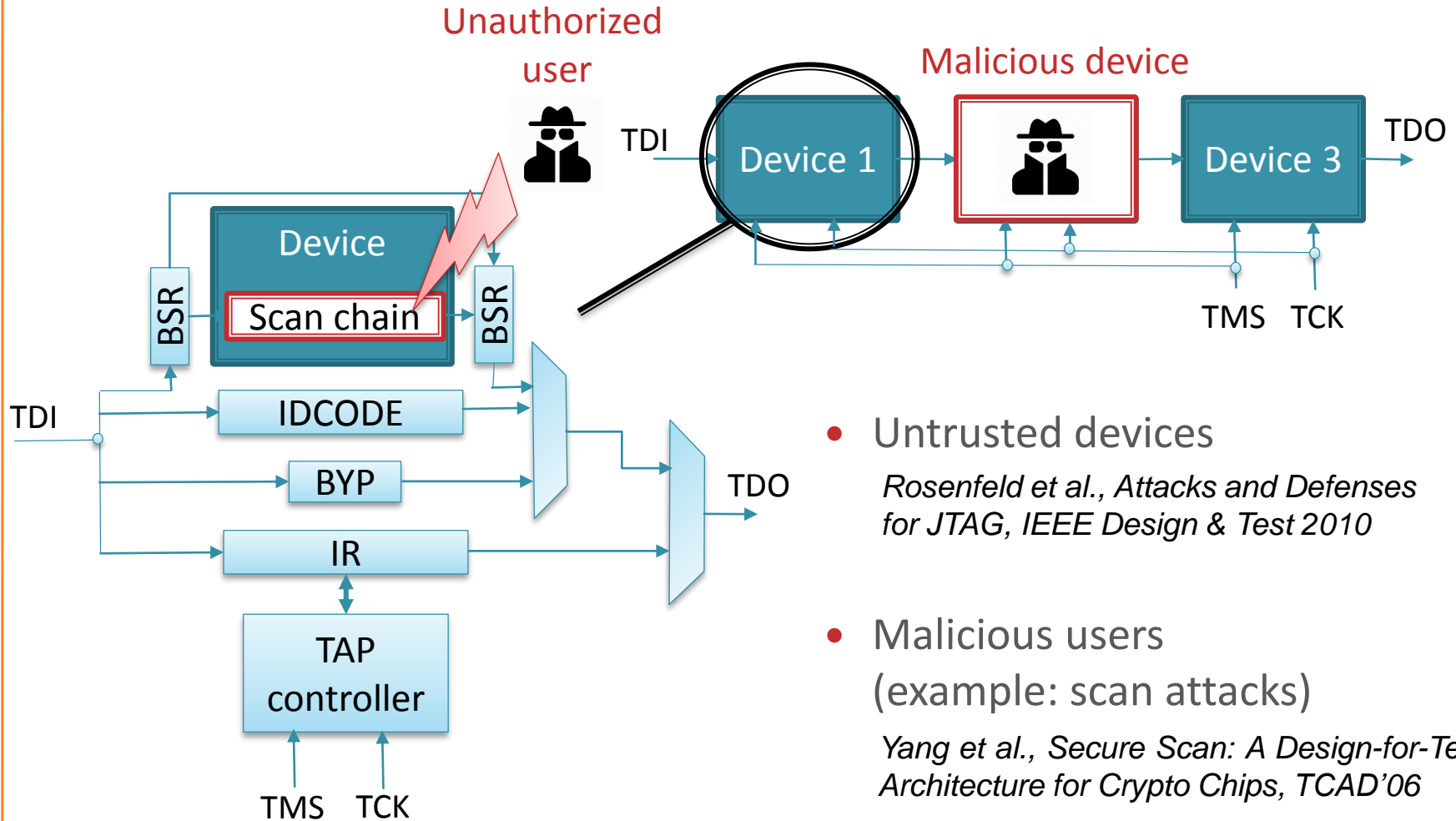
SUMMARY

- 1) Context of testing
- 2) Threats related to the test infrastructures**
 - Overview of the threats
 - Scan attacks
 - Security analysis on TEE
- 3) Proposed countermeasures: Scan Encryption
- 4) Pros and cons of the proposed countermeasures
- 5) Conclusion



THREATS

- OVERVIEW OF THE THREATS
- SCAN ATTACKS
- SECURITY ANALYSIS ON TEE



- Untrusted devices

Rosenfeld et al., Attacks and Defenses for JTAG, IEEE Design & Test 2010

- Malicious users (example: scan attacks)

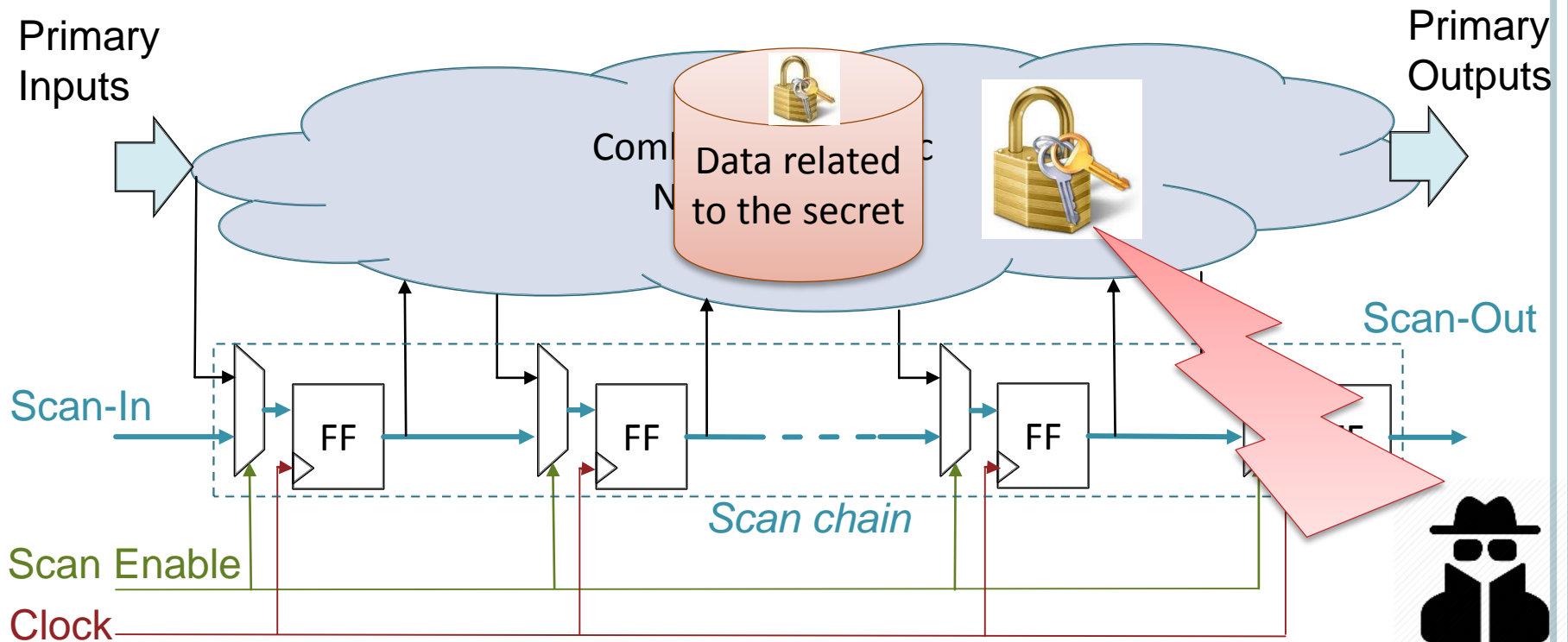
Yang et al., Secure Scan: A Design-for-Test Architecture for Crypto Chips, TCAD'06



SCAN ATTACK PRINCIPLE

- Exploit the scan chain by an attacker => Scan attacks

- Goal: Retrieve embedded secret data
- Exploit observability or controllability offered by scan chains
- Principle: switch between functional and scan modes



SCAN ATTACK ON AES

- OVERVIEW OF THE THREATS
- SCAN ATTACKS
- SECURITY ANALYSIS ON TEE

Advanced Encryption Standard (AES)

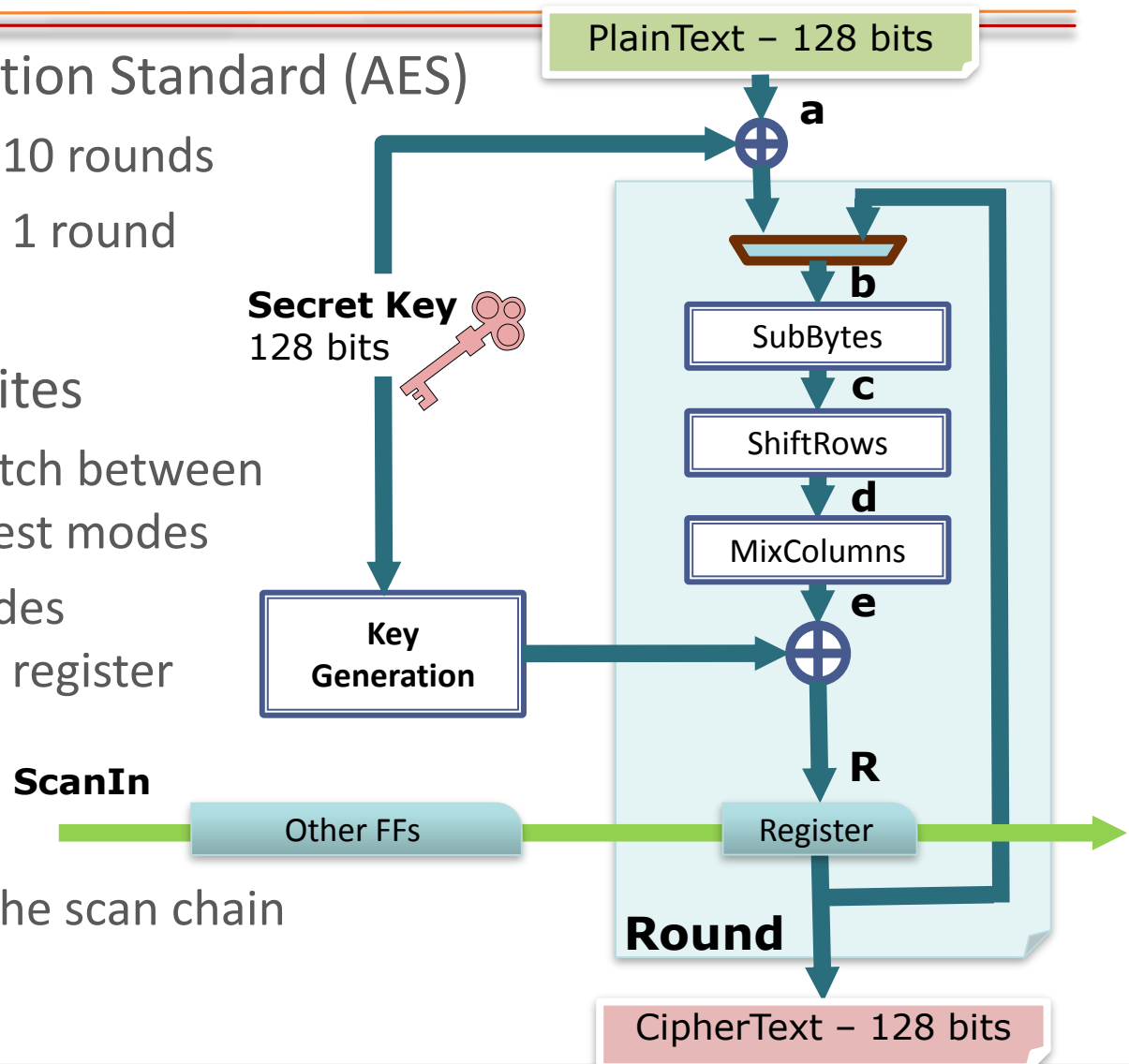
- Ciphertext after 10 rounds
- Not secure after 1 round

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

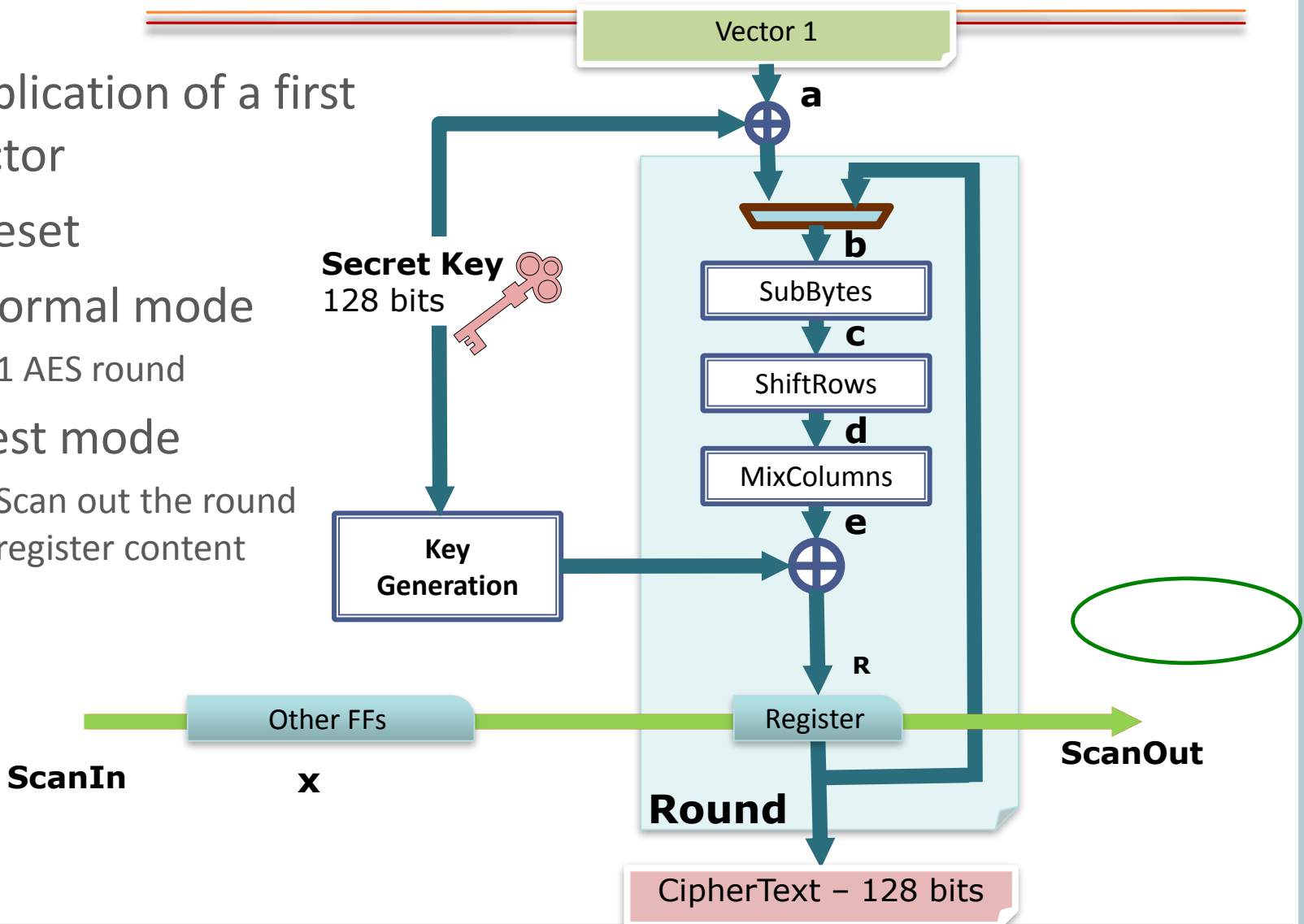


- OVERVIEW OF THE THREATS
- SCAN ATTACKS
- SECURITY ANALYSIS ON TEE

DIFFERENTIAL ATTACK

○ Application of a first vector

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

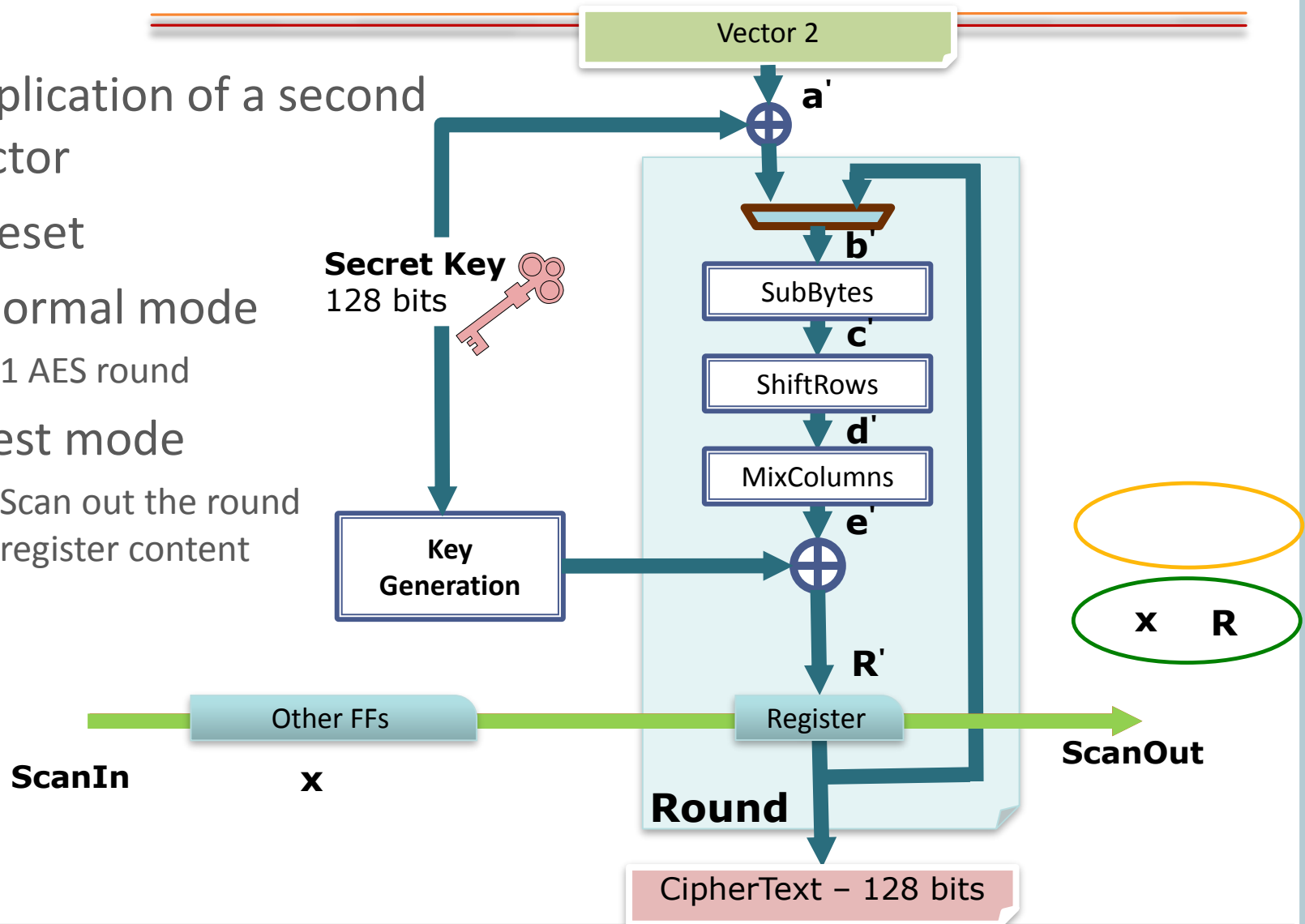


DIFFERENTIAL ATTACK

- OVERVIEW OF THE THREATS
- SCAN ATTACKS
- SECURITY ANALYSIS ON TEE

Application of a second vector

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



DIFFERENTIAL ATTACK

- OVERVIEW OF THE THREATS
- SCAN ATTACKS
- SECURITY ANALYSIS ON TEE

○ Hamming distance



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

○ On average, 32 trials

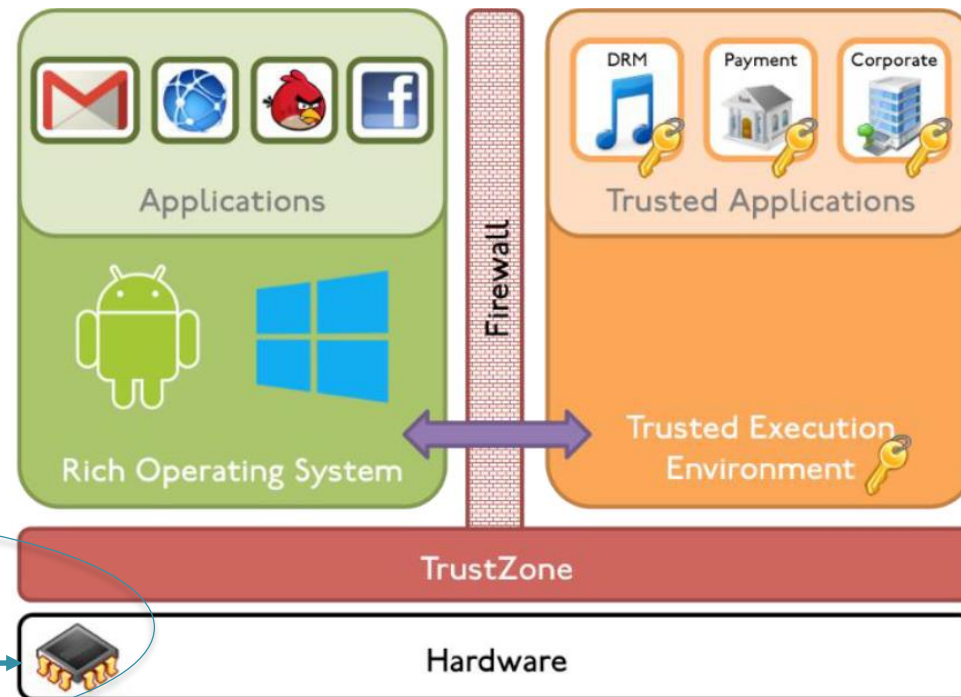
⇒ 512 trials to retrieve the whole 128-bit key



THREATS ON TEE?

- OVERVIEW OF THE THREATS
- SCAN ATTACKS
- SECURITY ANALYSIS ON TEE

- Accessing the scan chains => no differentiation between data processed and saved in Non Secure and Secure world
- Test & Debug access = an open door for attacks



©Prove & Run S.A.S



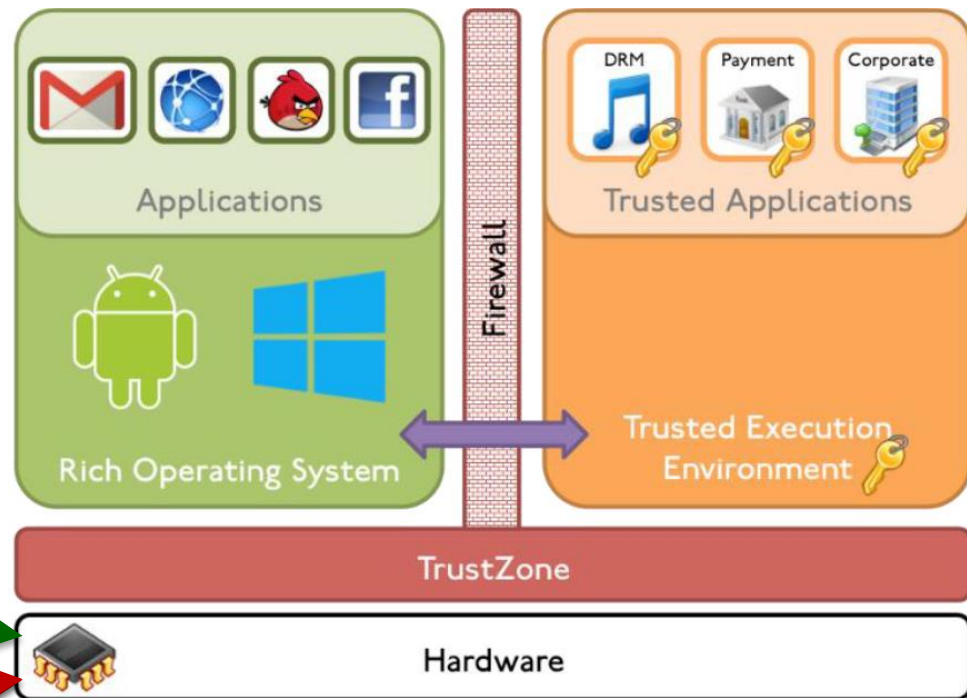
SECURITY ANALYSIS ON TEE

- OVERVIEW OF THE THREATS
- SCAN ATTACKS
- SECURITY ANALYSIS ON TEE

Industrial practice to ensure protection:
disconnection of the access to the scan chains

Disadvantages:

- In-field diagnosis and debug impossible
 - Probing on disconnected access
- ⇒ Circumvent the countermeasure



©Prove & Run S.A.S.



20/09/2018

SUMMARY

- 1) Context of testing
- 2) Threats related to the test infrastructures
- 3) Proposed countermeasures: Scan Encryption**
 - Principle of Scan Encryption
 - Implementation with block cipher
 - Implementation with stream cipher
- 4) Application of the proposed countermeasures
- 5) Conclusion



SCAN ENCRYPTION

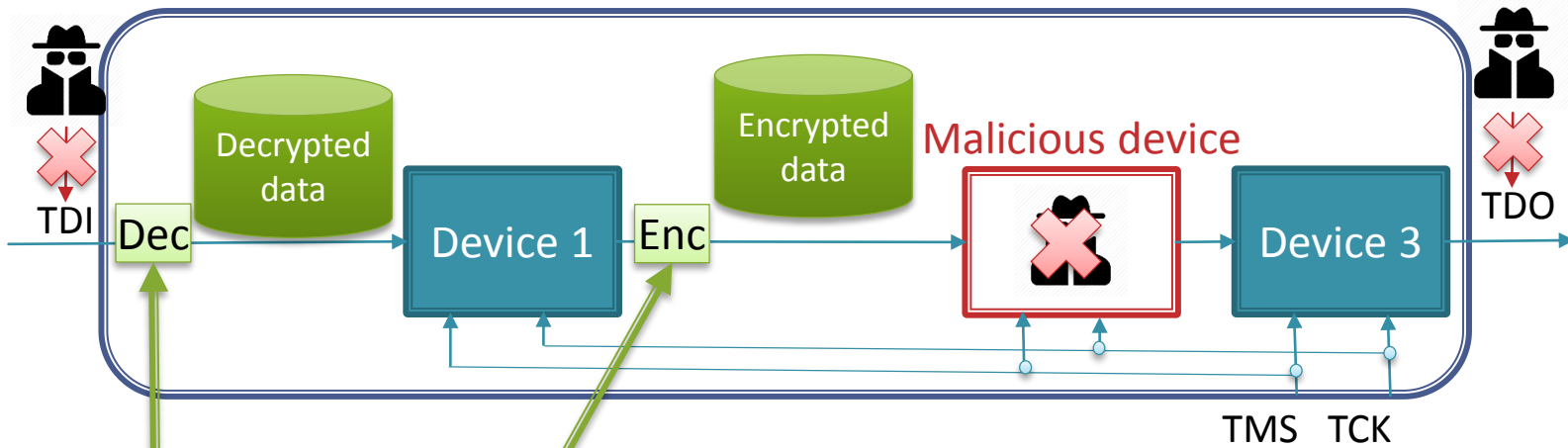
- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

○ Solution: test communication encryption

Unauthorized user

Chip

Unauthorized user



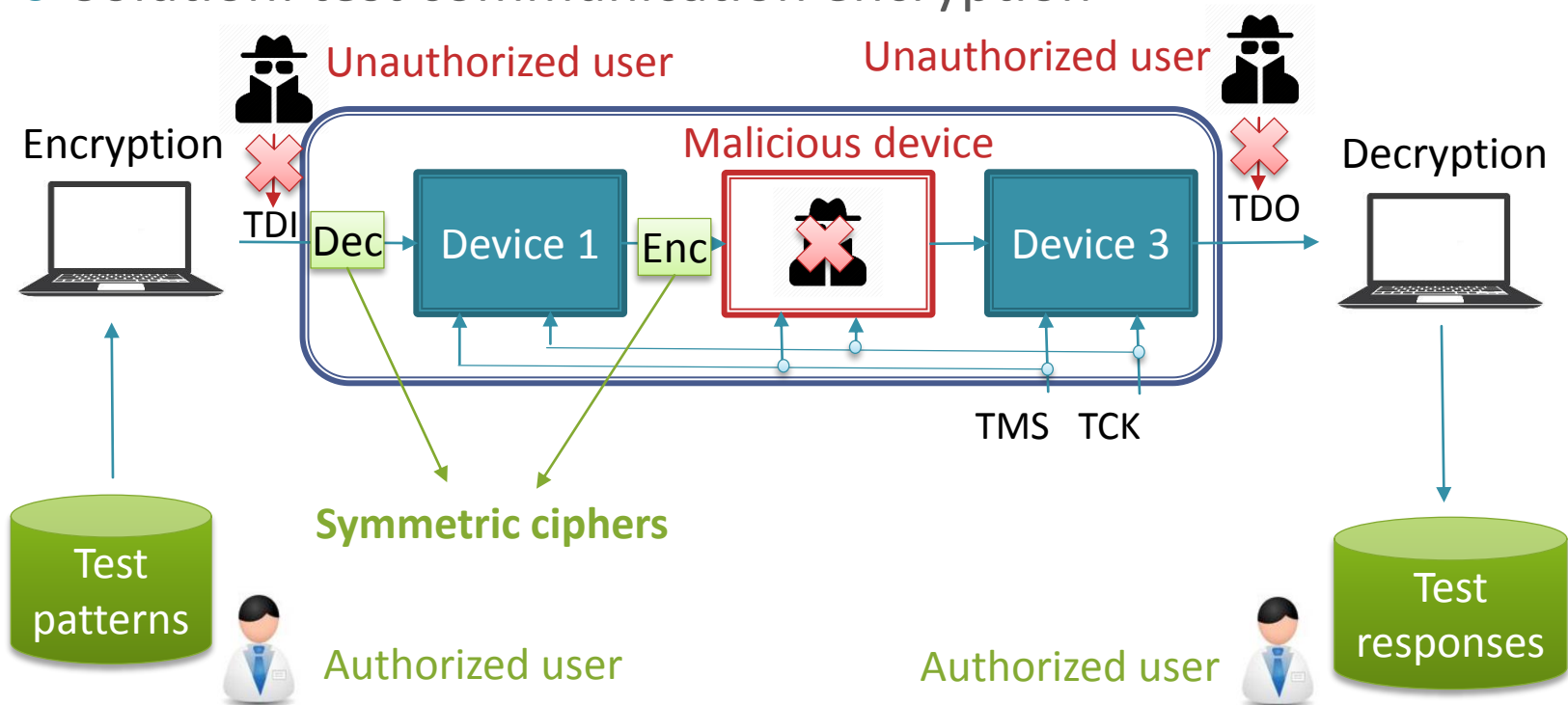
- **Input decryption** prevents sending desired test data
- **Output encryption** prevents reading plain test responses



SCAN ENCRYPTION

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

○ Solution: test communication encryption



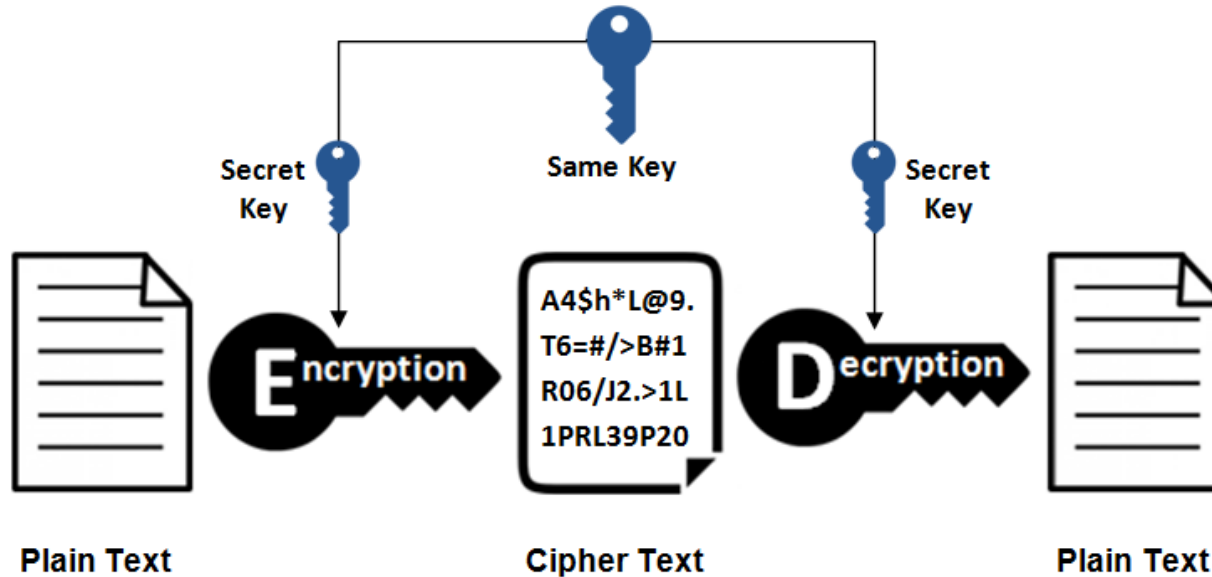
- Input decryption prevents sending desired test data
- Output encryption prevents reading plain test responses
- Test/debug only possible by authorized user knowing the secret key



SYMMETRIC CIPHER

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

Symmetric Encryption



- 2 types of symmetric cipher: stream and block ciphers

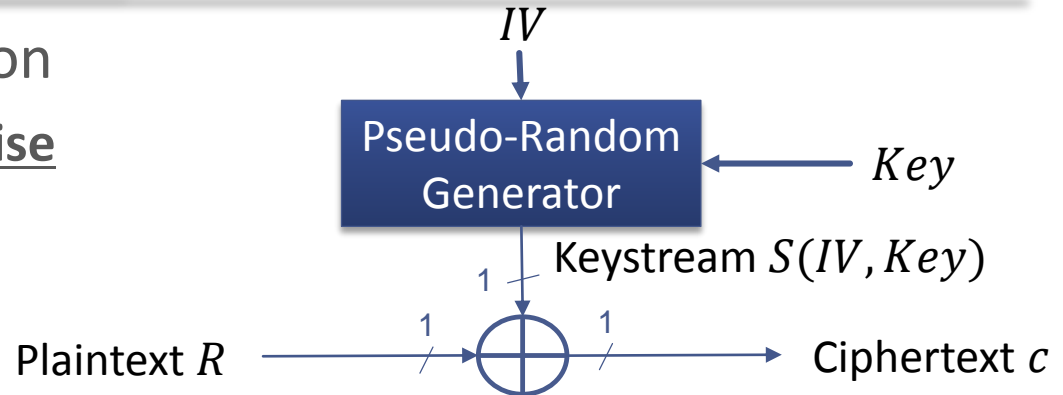


STREAM CIPHER / BLOCK CIPHER

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

Stream cipher encryption

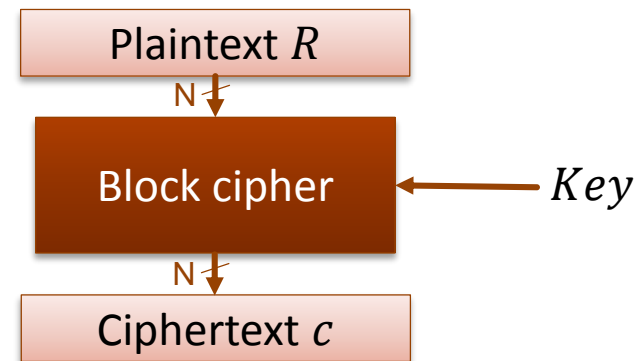
- Keystream XORed **bitwise** with the plaintext



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

Block cipher encryption

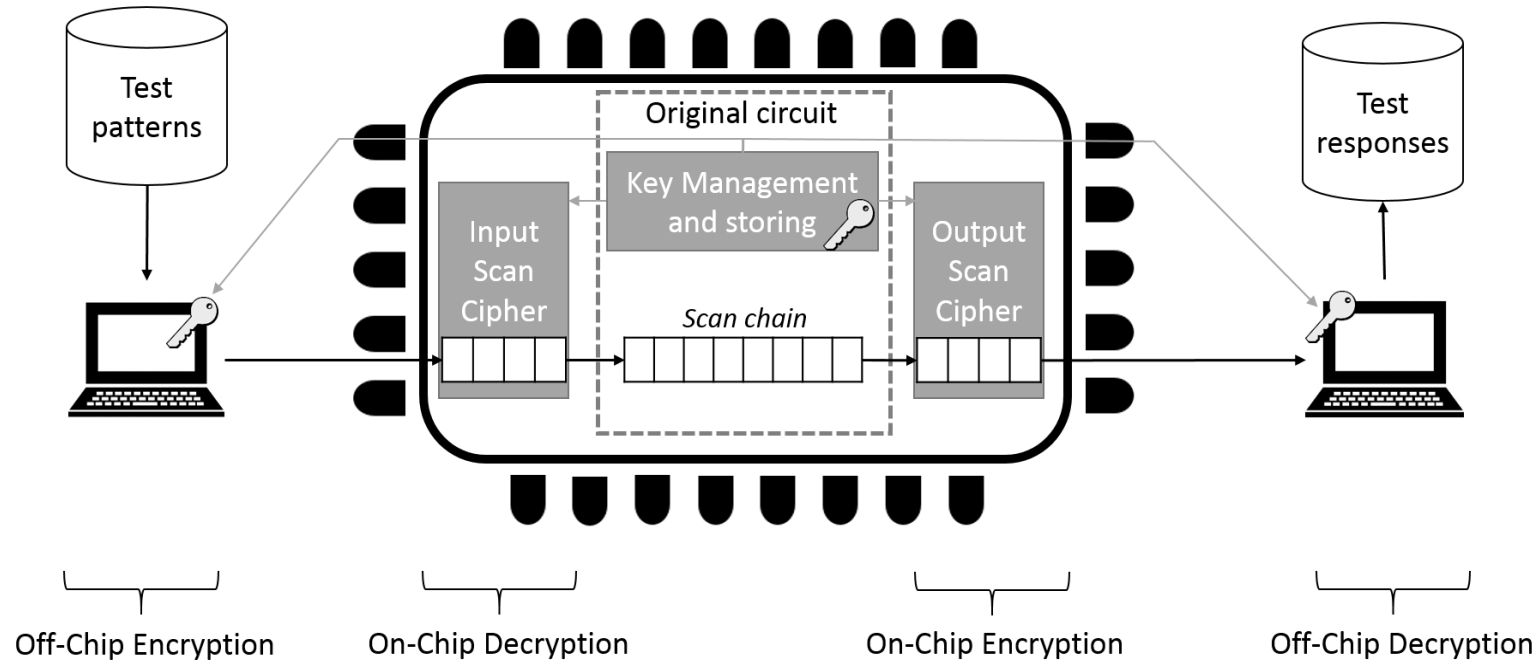
- Confusion and diffusion on a **block** of plaintext
- + Strong security
- But cost?



BASIC SCHEME

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

- Study of both solutions (block cipher and stream cipher)



- 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

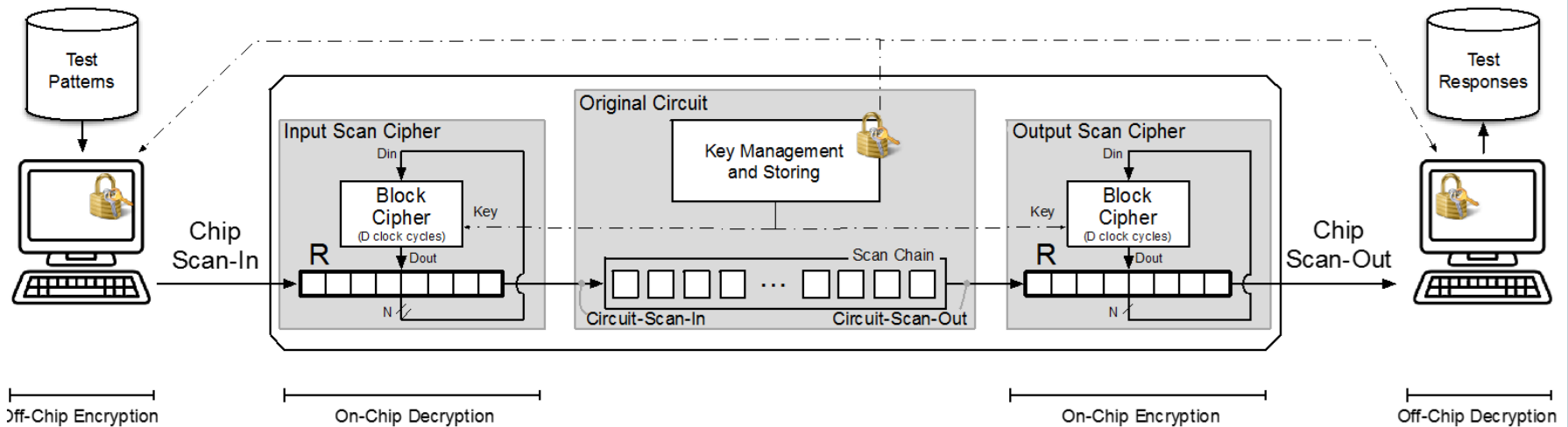


BLOCK CIPHER-BASED SOLUTION

- PRINCIPLE OF SCAN ENCRYPTION
- **IMPLEMENTATION WITH BLOCK CIPHER**
- IMPLEMENTATION WITH STREAM CIPHER

Implementation on scan chain with 2 PRESENT block ciphers:

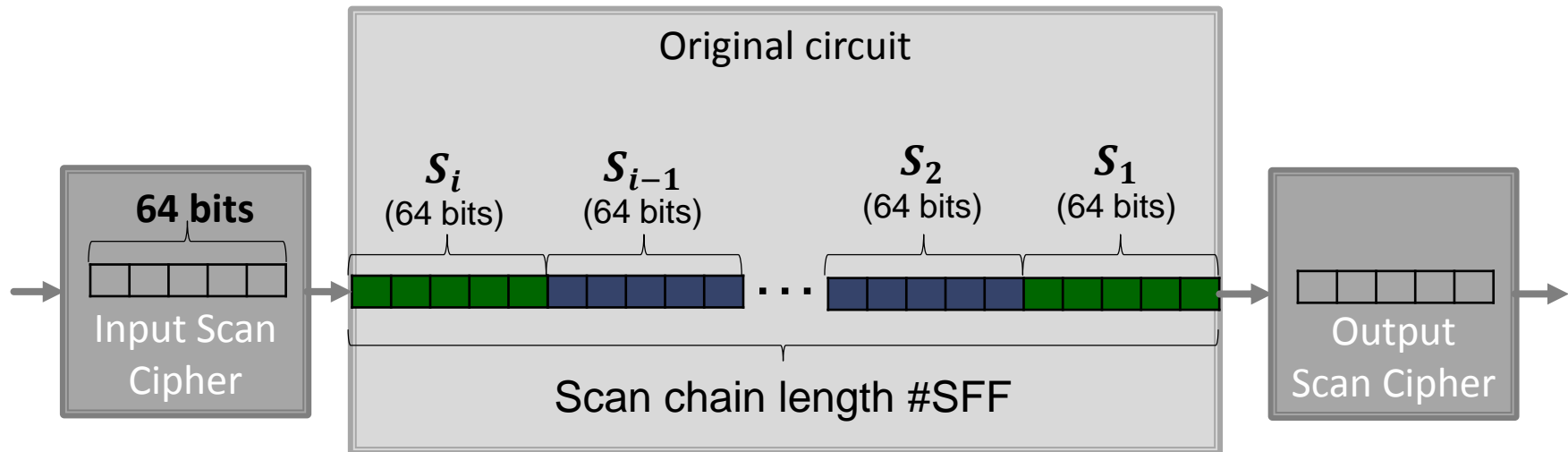
- Lightweight (1 PRESENT = 2 139 GE)
- Encryption by 64-bits block size



MODE OF OPERATIONS

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

- 64 bits encrypted every 32 clock cycles



⇒ **#SFF = P x 64**

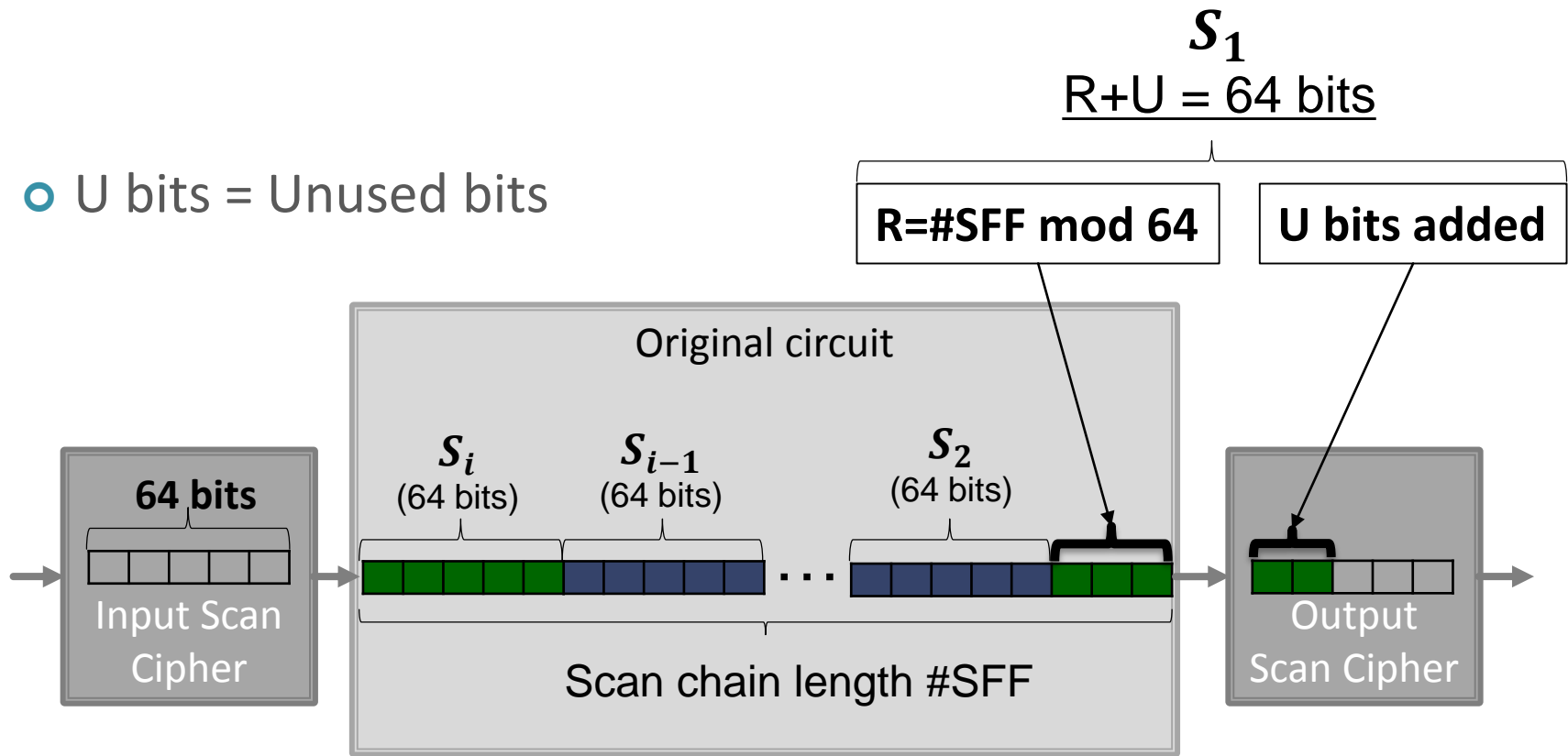
⇒ **No test time overhead on each pattern**



MODE OF OPERATIONS

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

- U bits = Unused bits



⇒ **#SFF = P x 64 + R**

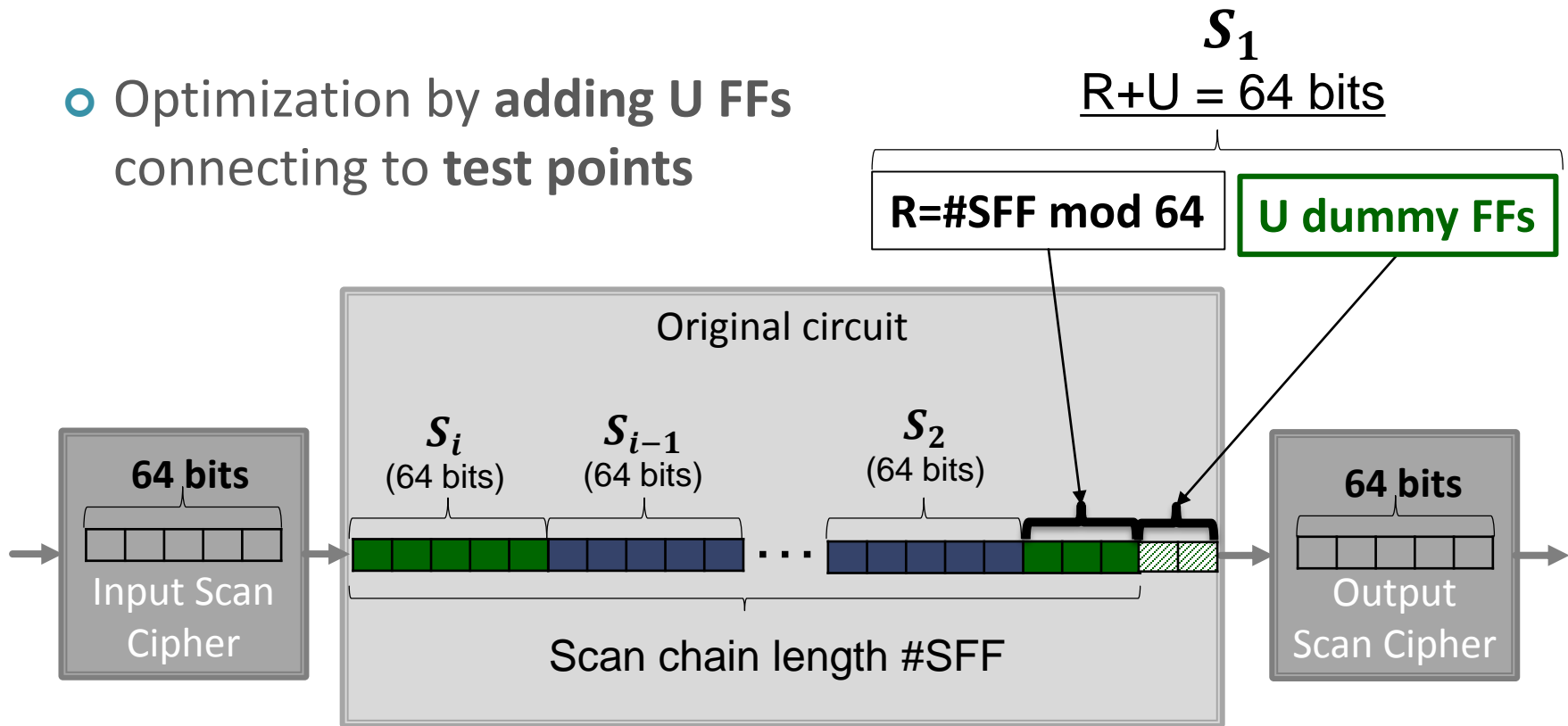
⇒ **Loss of U clock cycles per pattern**



TEST TIME OPTIMIZATION

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

- Optimization by adding **U** FFs connecting to **test points**



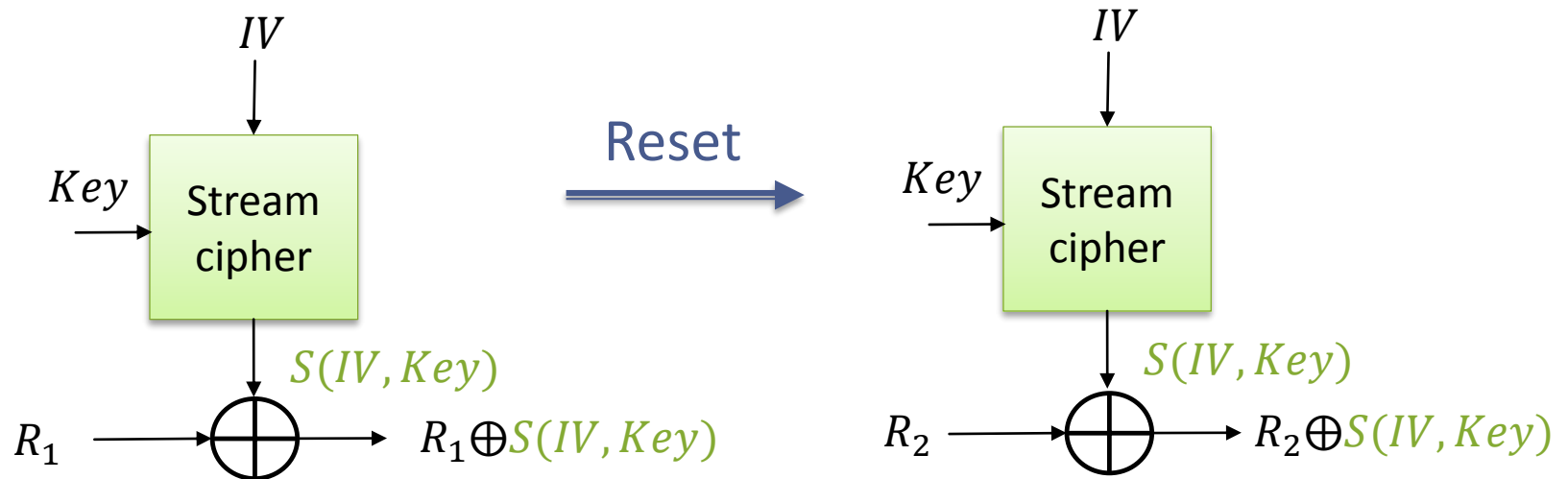
⇒ **Reduce test time overhead**



STREAM CIPHER SECURITY?

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

- **Two-times pad:** same key and *IV* re-used => same keystream generated to encrypt different data



⇒ Possible to carry out attacks if requirement is not fit



⇒ Solution: *IV* generated randomly at each circuit reset

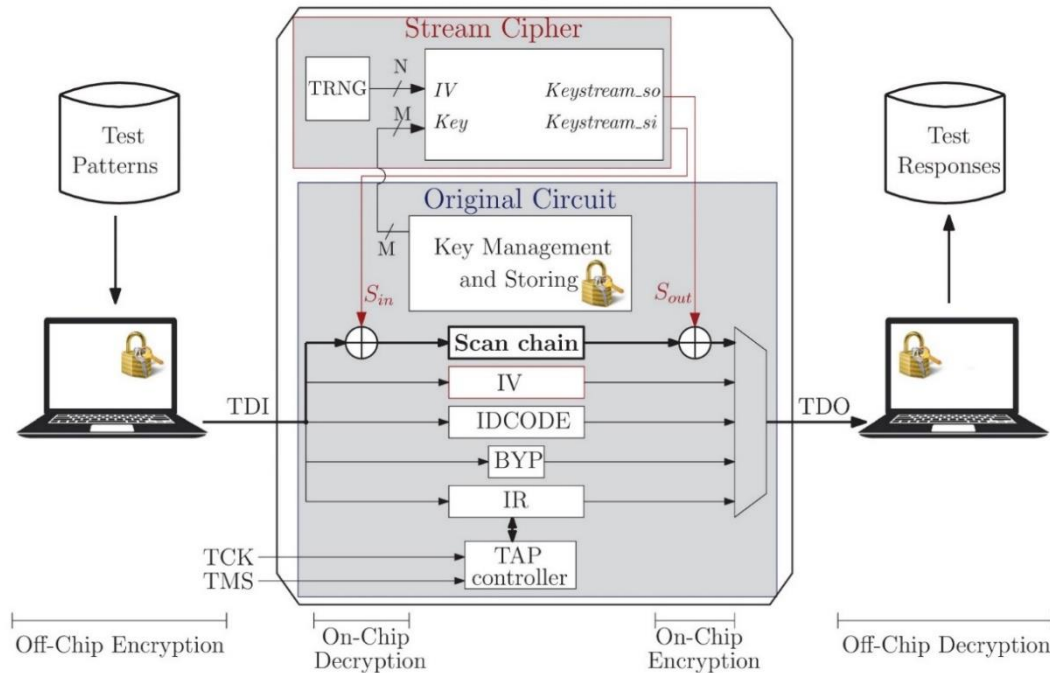
$$R_1 \oplus S(IV_1, Key) \oplus R_2 \oplus S(IV_2, Key)$$



STREAM CIPHER-BASED SOLUTION

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

- 1 TRIVIUM stream cipher (2 016 GE)
 - 2 Keystreams
 - True Random Number Generator (TRNG) to generate random IV
- E.g. on JTAG, new instruction *GetIV* with a test data register IV



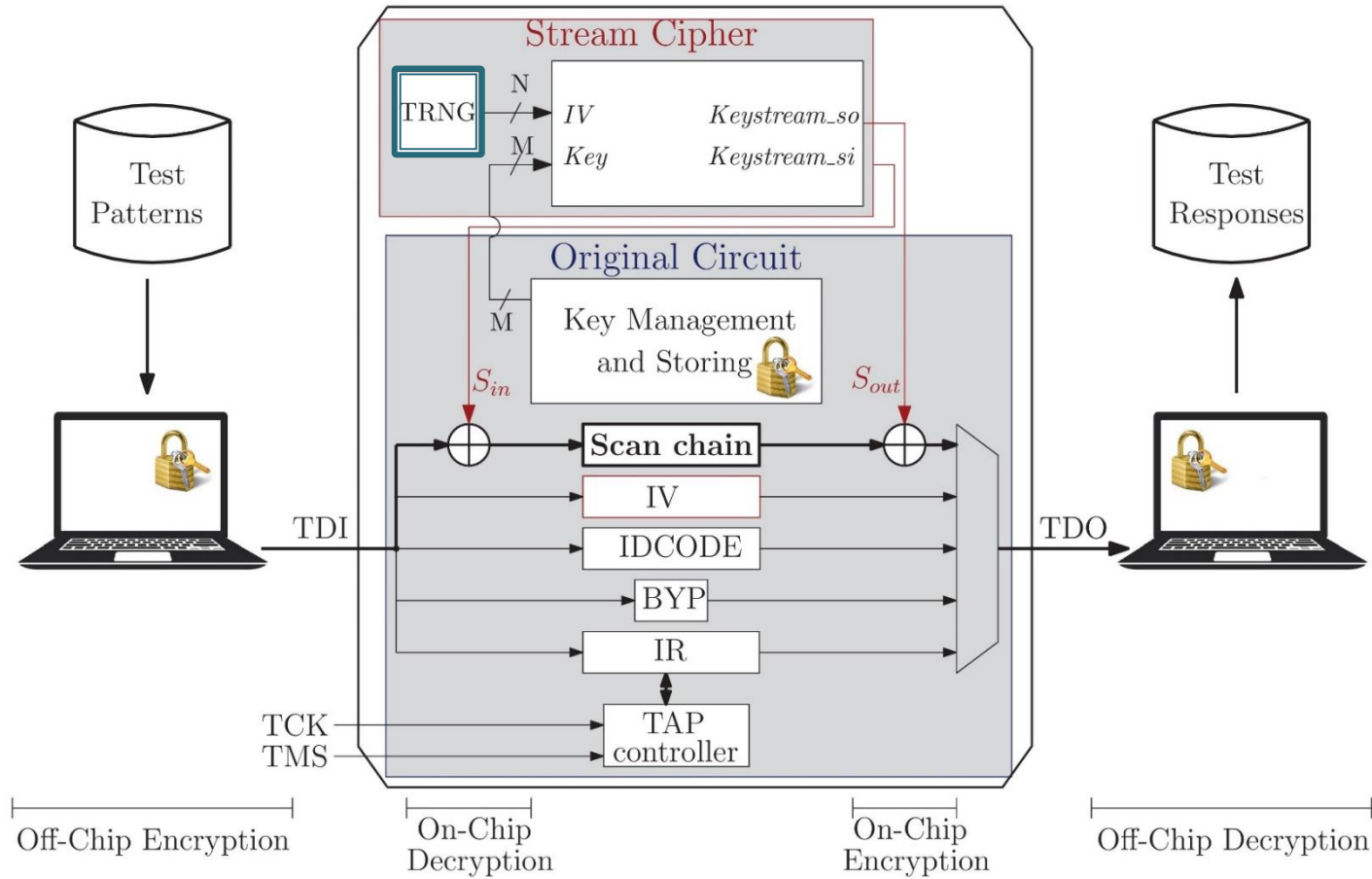
- Mode of operations in 2 phases: initialization and encryption



INITIALIZATION PHASE

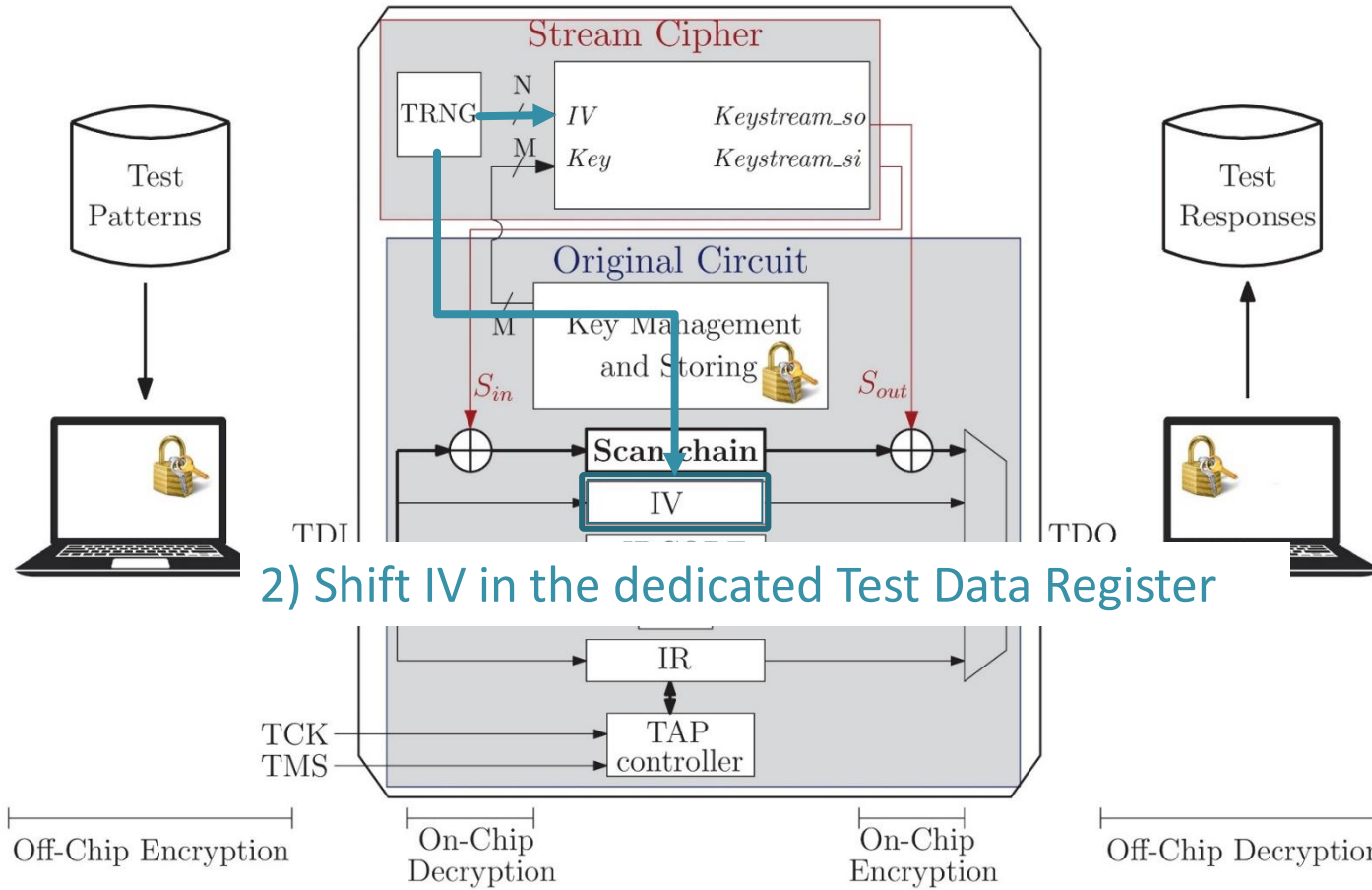
- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- **IMPLEMENTATION WITH STREAM CIPHER**

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



INITIALIZATION PHASE

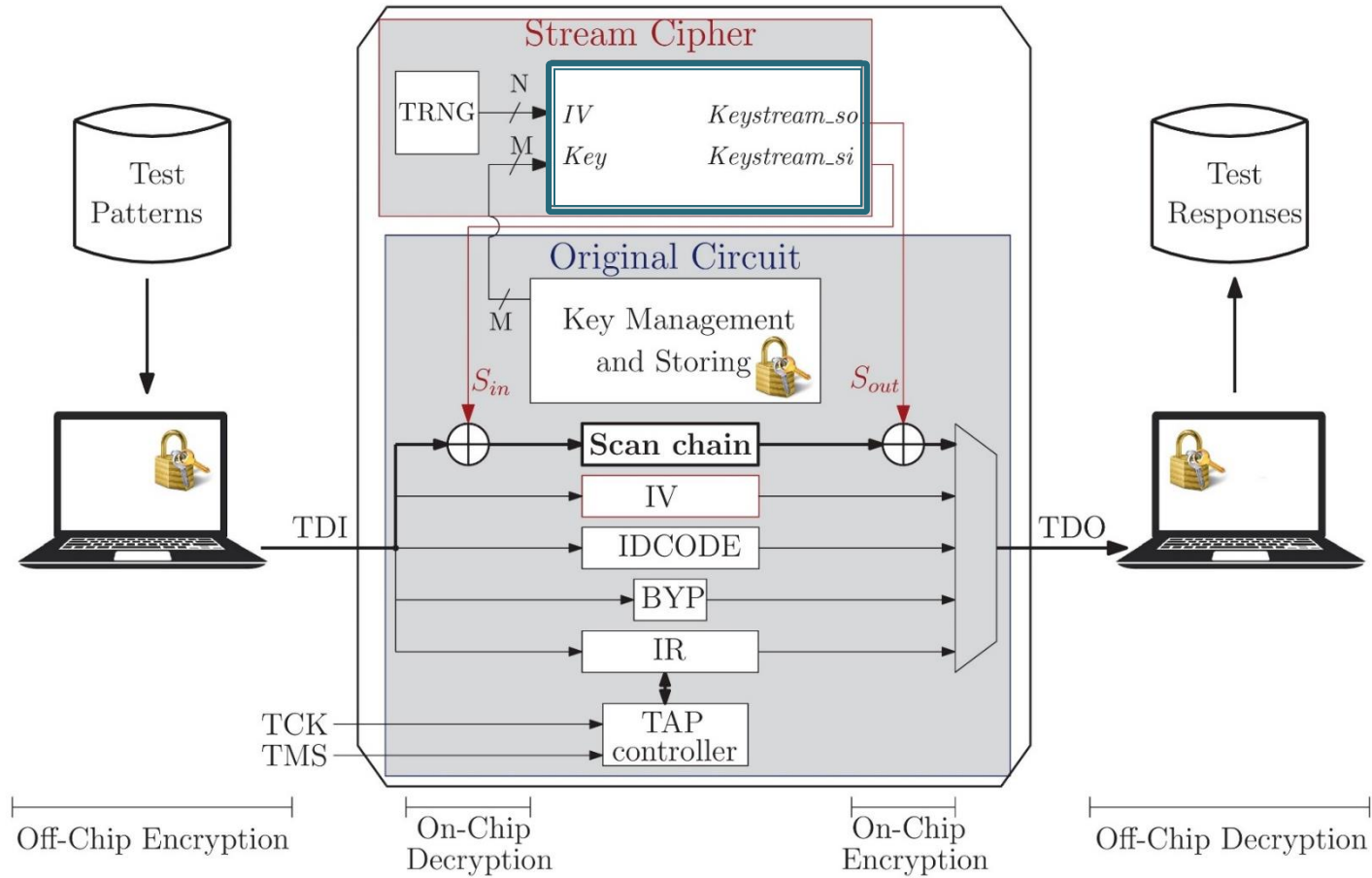
- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- **IMPLEMENTATION WITH STREAM CIPHER**



INITIALIZATION PHASE

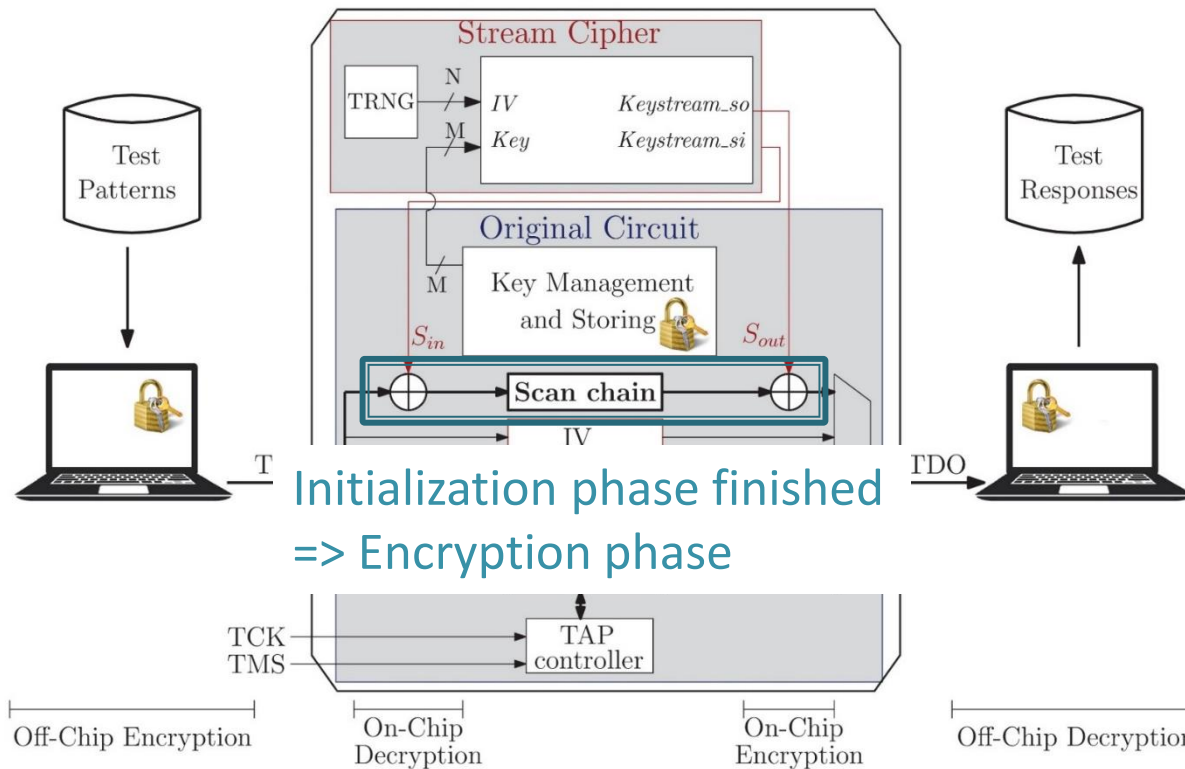
- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- **IMPLEMENTATION WITH STREAM CIPHER**

3) Stream cipher setup



INITIALIZATION PHASE

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER



Test time overhead:

- 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

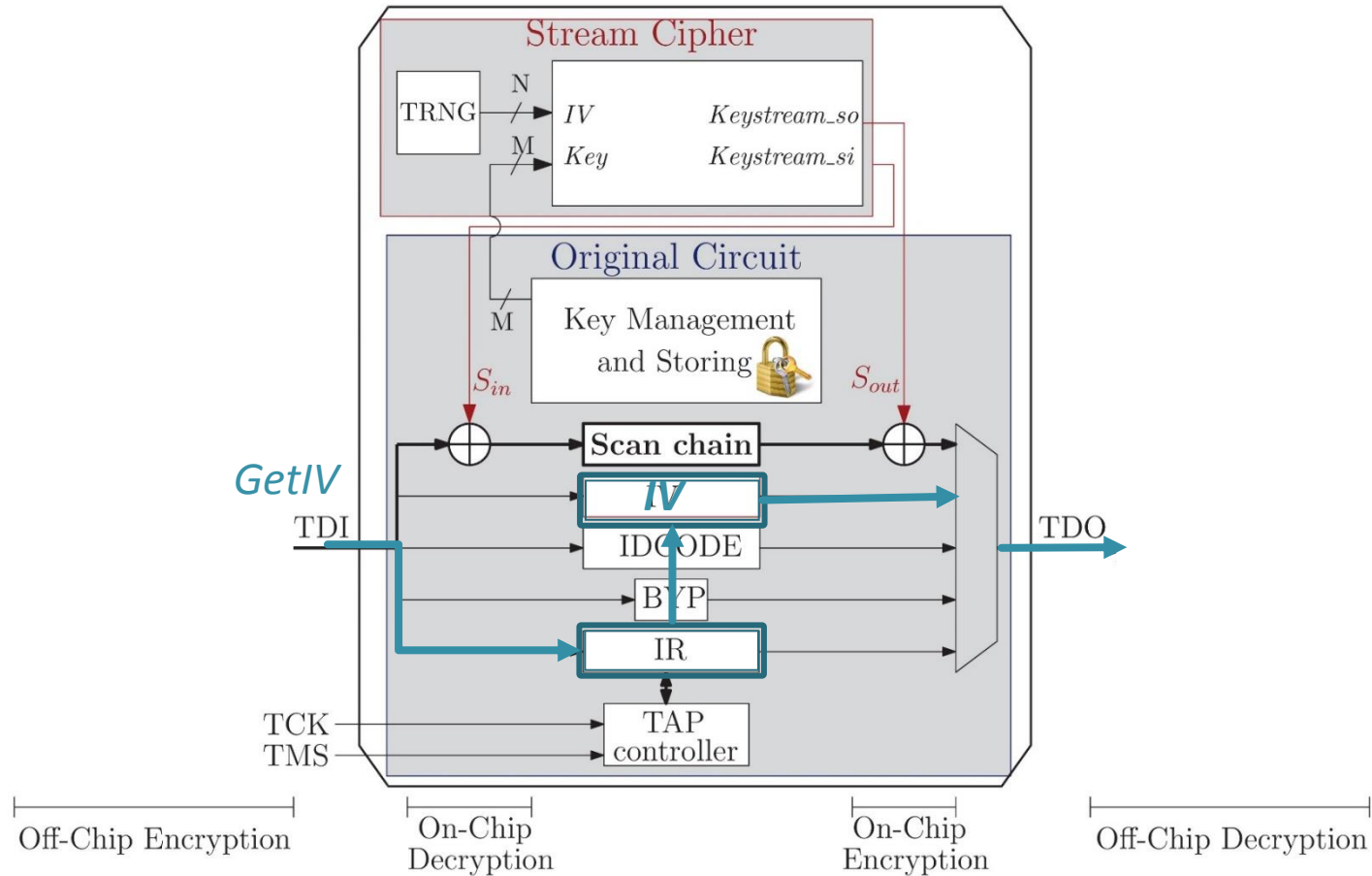


ENCRYPTION PHASE

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

○ Send *GETIV* instruction

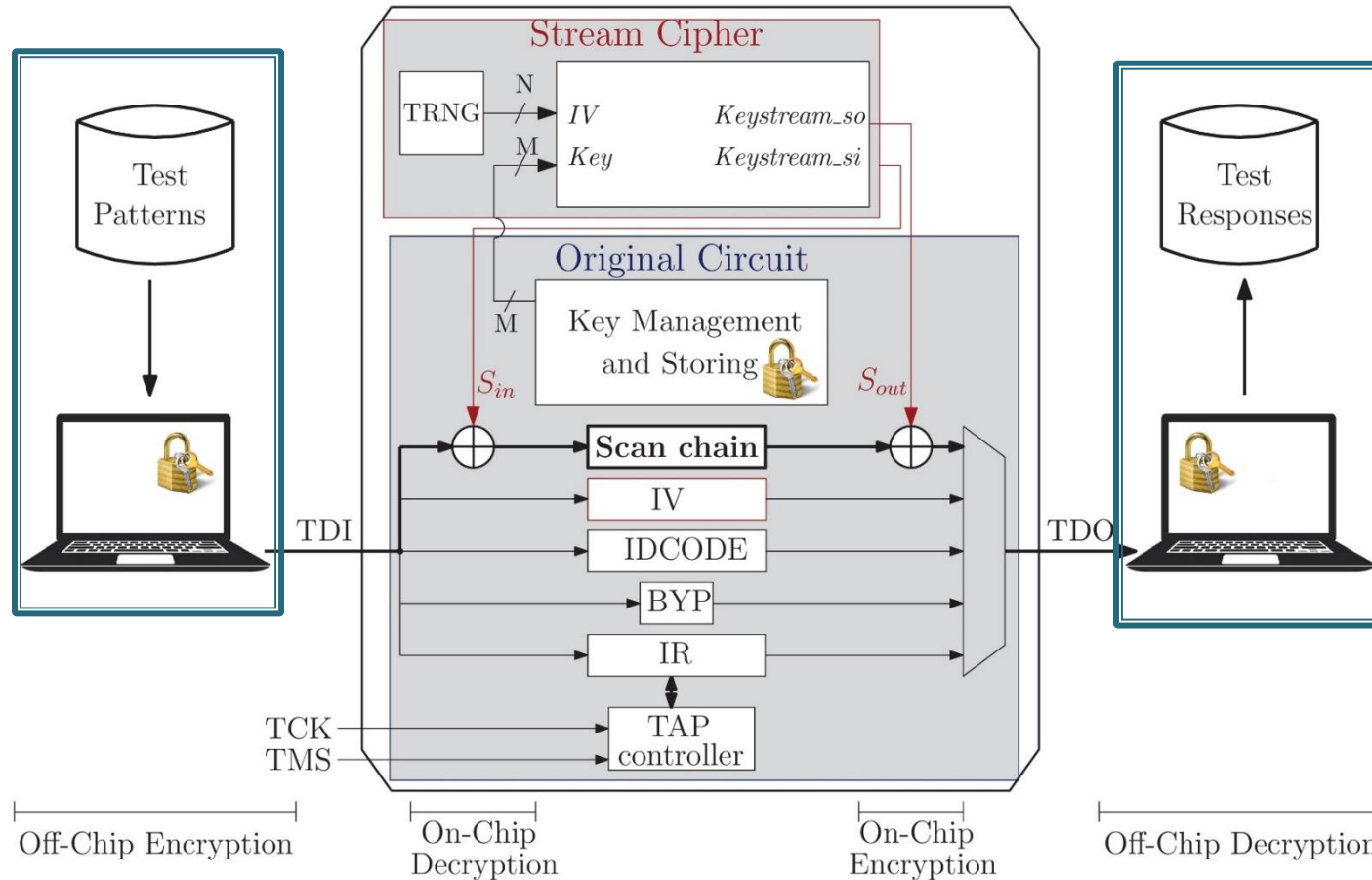
⇒ Shift the content of the IV register out the circuit



ENCRYPTION PHASE

- PRINCIPLE OF SCAN ENCRYPTION
- IMPLEMENTATION WITH BLOCK CIPHER
- IMPLEMENTATION WITH STREAM CIPHER

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



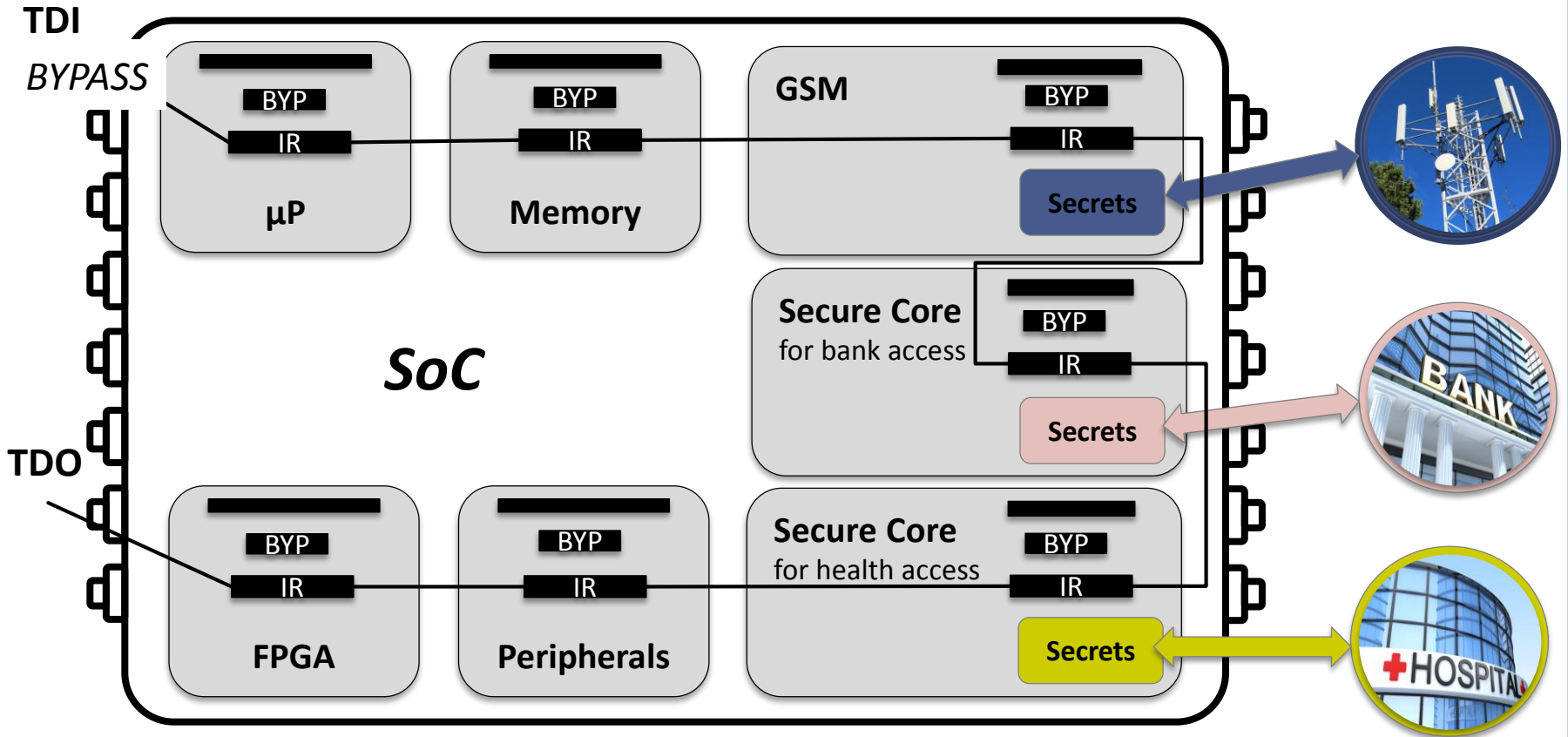
SUMMARY

- 1) Context of testing
- 2) Threats related to the test infrastructures
- 3) Proposed countermeasures: Scan Encryption
- 4) Application of the proposed countermeasures**
 - Integration in a SoC design
 - General advantages
 - Comparison between both implementations
- 5) Conclusion



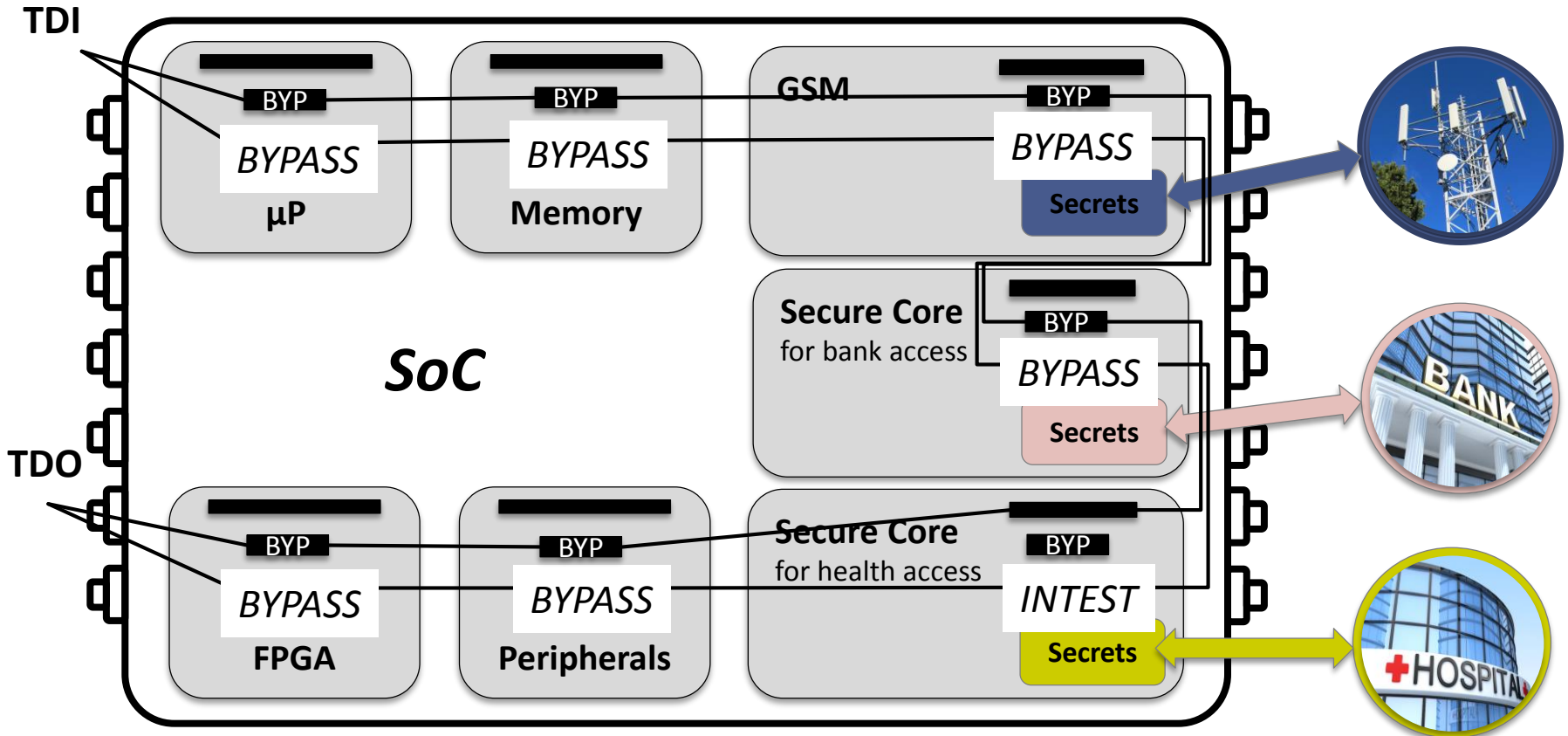
EXAMPLE OF SoC DESIGN

- INTEGRATION IN A SoC DESIGN
- GENERAL ADVANTAGES
- COMPARISON BETWEEN BOTH IMPLEMENTATIONS



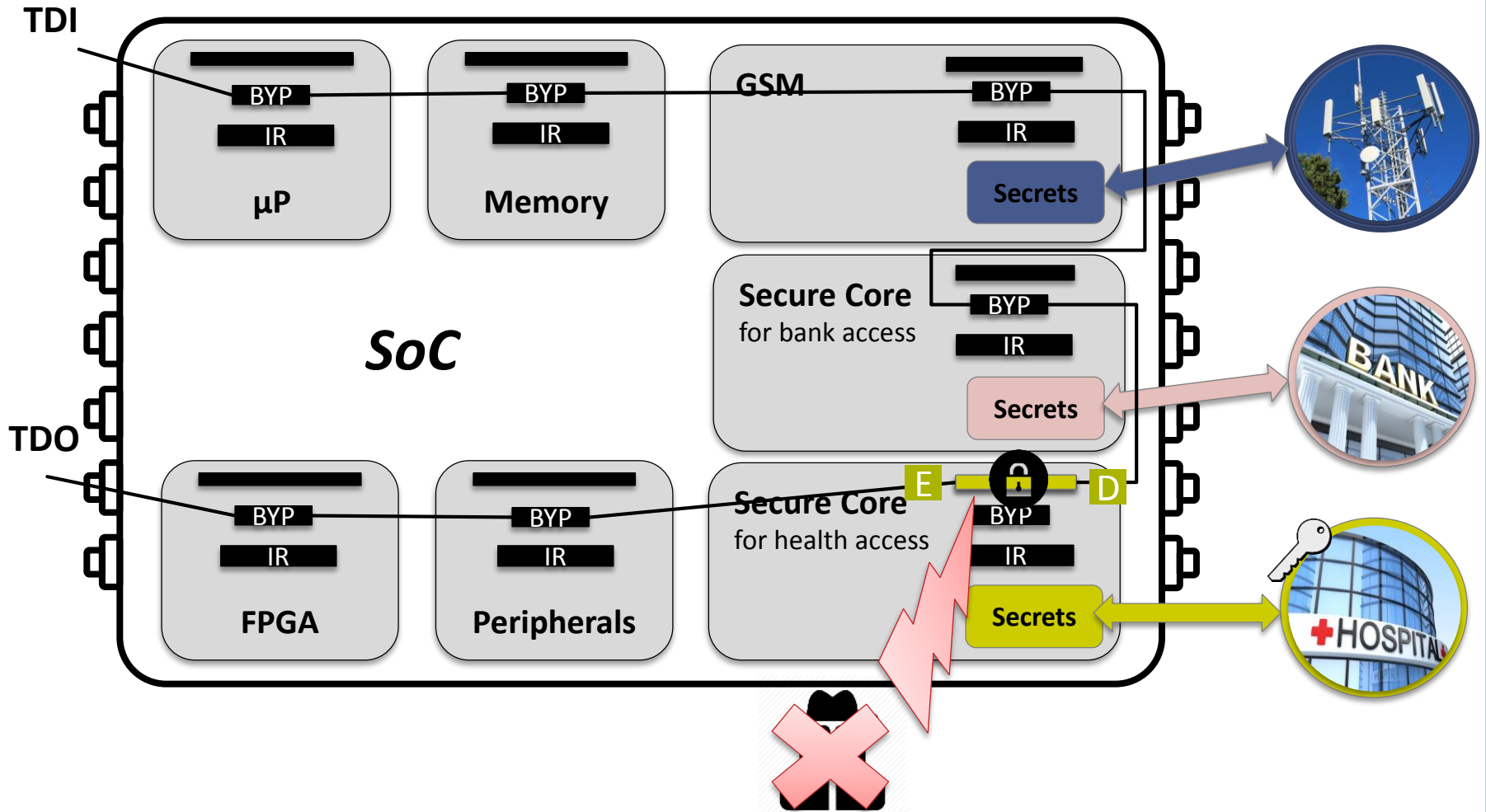
INSTRUCTIONS SHIFTED IN IR REGISTERS

- INTEGRATION IN A SOC DESIGN
- GENERAL ADVANTAGES
- COMPARISON BETWEEN BOTH IMPLEMENTATIONS



INTEGRATION OF SCAN ENCRYPTION

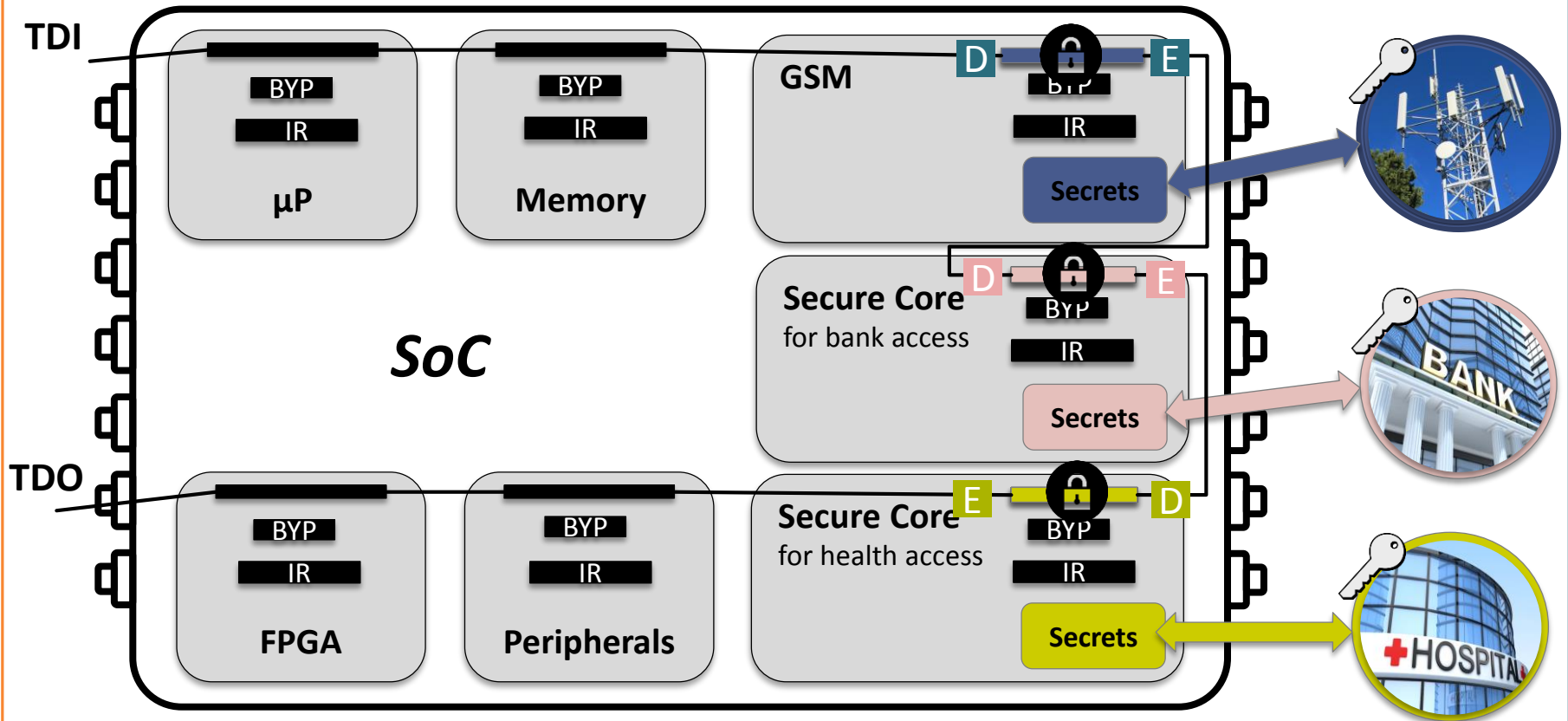
- INTEGRATION IN A SOC DESIGN
- GENERAL ADVANTAGES
- COMPARISON BETWEEN BOTH IMPLEMENTATIONS



FINE-GRAINED ACCESS

- INTEGRATION IN A SOC DESIGN
- GENERAL ADVANTAGES
- COMPARISON BETWEEN BOTH IMPLEMENTATIONS

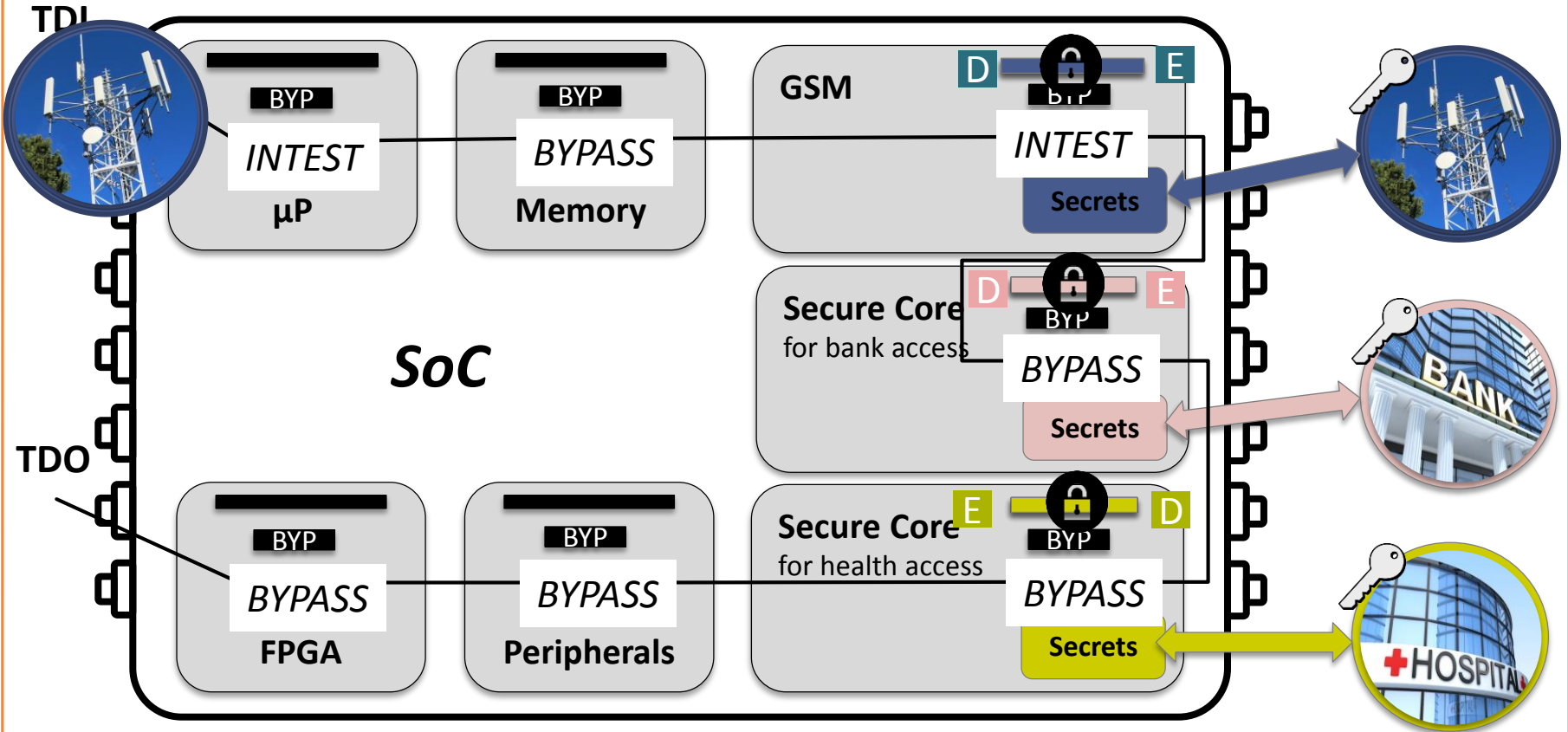
- Allow to distinguish between different group of users



EXAMPLE

- INTEGRATION IN A SOC DESIGN
- GENERAL ADVANTAGES
- COMPARISON BETWEEN BOTH IMPLEMENTATIONS

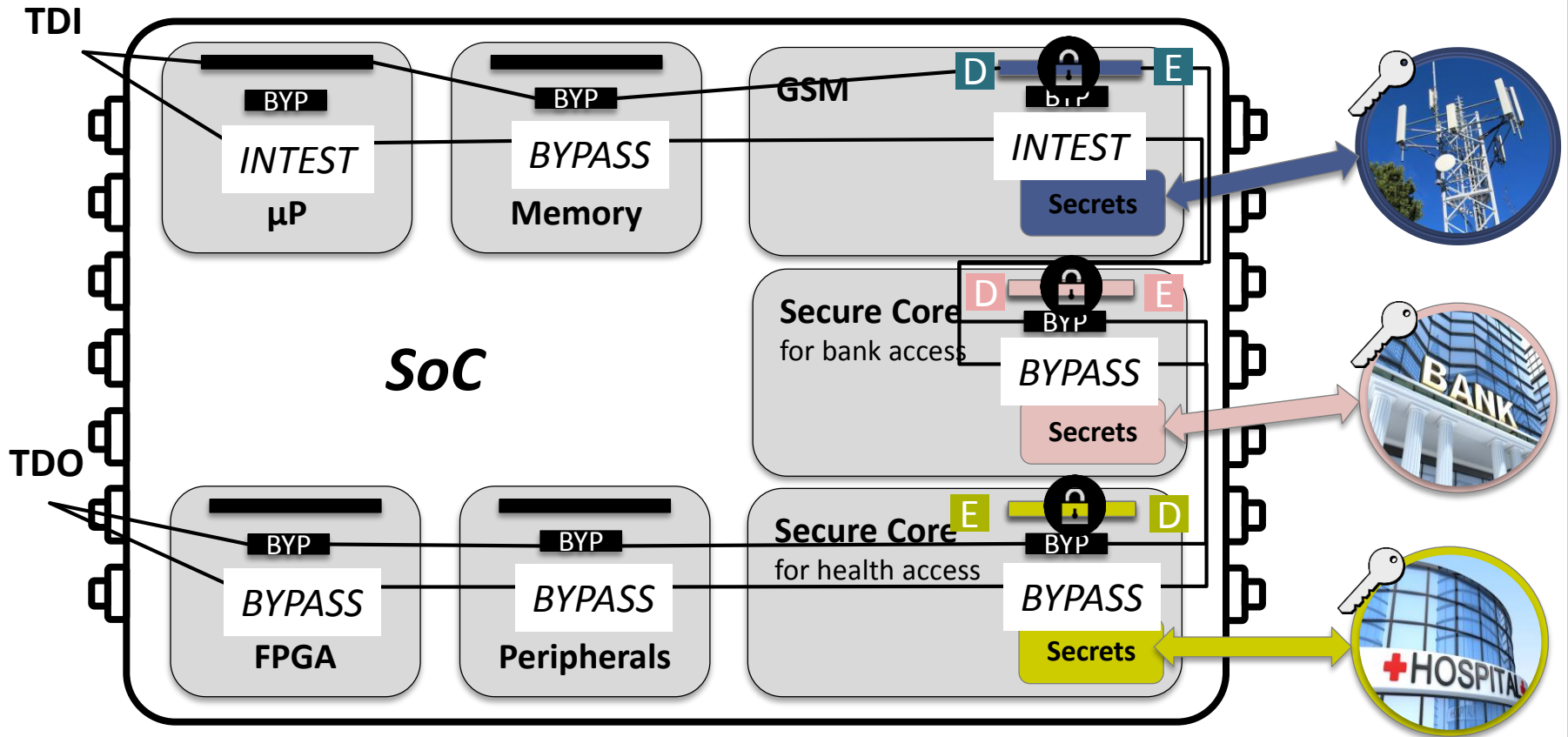
- Test in the SoC of μ P and GSM module by GSM operator



EXAMPLE

- INTEGRATION IN A SOC DESIGN
- GENERAL ADVANTAGES
- COMPARISON BETWEEN BOTH IMPLEMENTATIONS

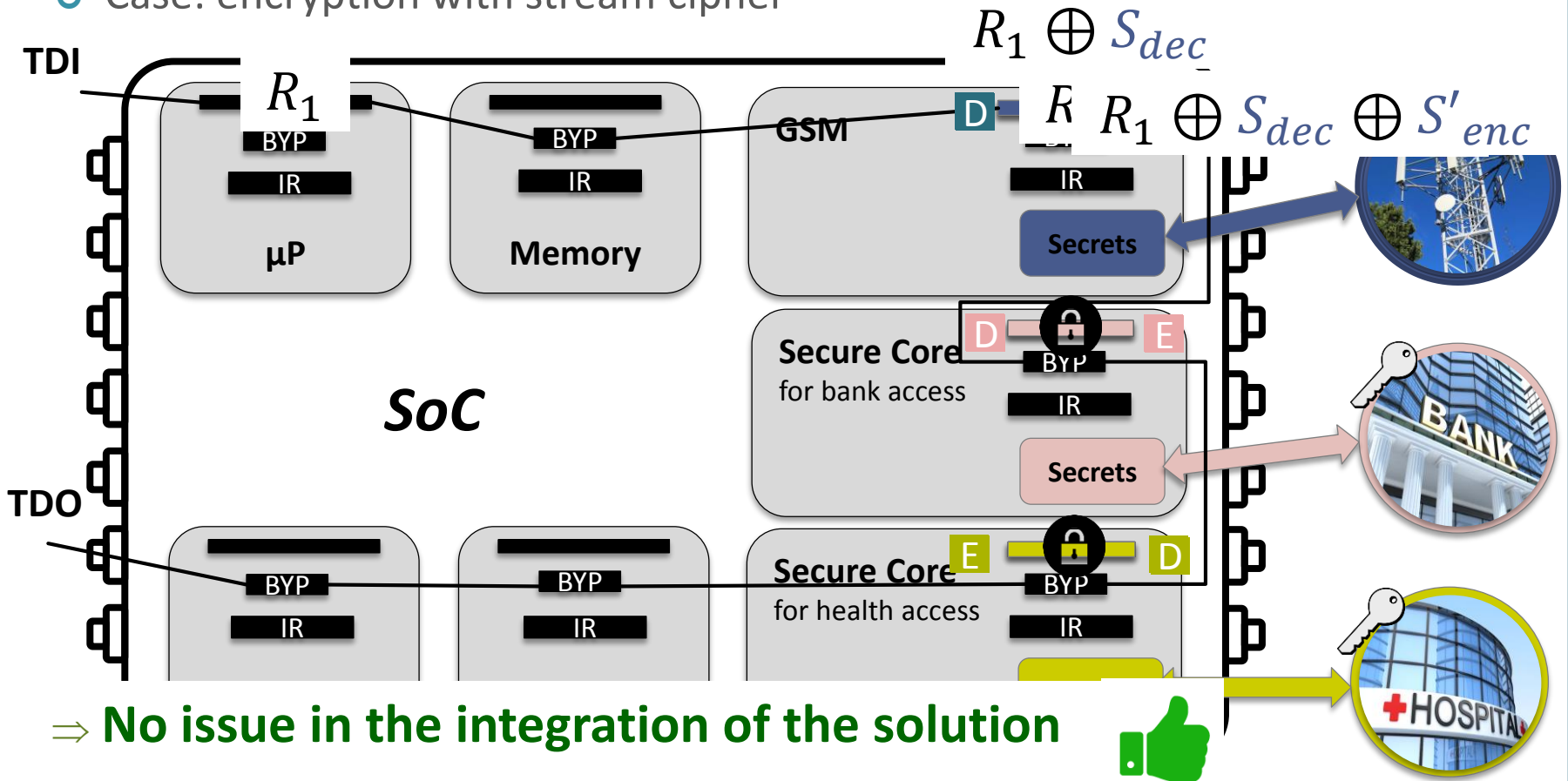
- Test in the SoC of μ P and GSM module by GSM operator



EXAMPLE

- INTEGRATION IN A SOC DESIGN
- GENERAL ADVANTAGES
- COMPARISON BETWEEN BOTH IMPLEMENTATIONS

- Test in the SoC of μP and GSM module by GSM operator
- Case: encryption with stream cipher



GENERAL ADVANTAGES

- INTEGRATION IN A SoC DESIGN
- **GENERAL ADVANTAGES**
- COMPARISON BETWEEN BOTH IMPLEMENTATIONS

○ Advantages of scan encryption solutions (both stream and block encryption):

+ Security

- Protected against scan attacks
- Protected against malicious core

+ Key management

- Re-use key management already implemented

+ Diagnosis and debug preserved

- Still possible in-field

+ Integration in a SoC design









- No issue



COMPARISON

- INTEGRATION IN A SOC DESIGN
- GENERAL ADVANTAGES
- COMPARISON BETWEEN BOTH IMPLEMENTATIONS

○ Block cipher vs stream cipher

	Stream cipher-based solution		Block cipher-based solution	
Conditions on the original circuit	TRNG already implemented	No TRNG implemented	Scan chain length not multiple of 64	Scan chain multiple of 64 (insertion of test points)
Cost				
- Area				
- Test time				



SUMMARY

- 1) Context of testing
- 2) Threats related to the test infrastructures
- 3) Proposed countermeasures: Scan Encryption
- 4) Application of the proposed countermeasures
- 5) Conclusion**



CONCLUSION

- Need a protection on the test infrastructures (even with TEE)
 - ⇒ Data saved and processed in Secure world can be controlled and observed through the scan chains
- Solution consisting in disconnecting test accesses
 - ⇒ Important issues with in-field diagnosis and debug
 - ⇒ Security threats with probing attacks
- Proposition of Scan Encryption countermeasures
 - ⇒ Preserve diagnosis and debug only for authorized users
 - ⇒ Prevents both external and internal attacks exploiting test infrastructures
 - ⇒ Study of two implementations (block cipher and stream cipher)



Thank You

