Laboratoire
Informatique
Robotique
Microélectronique
Montpellier

Hardware Security and Trust

Giorgio Di Natale, CNRS

License Information

Disclaimer

- We disclaim any warranties or representations as to the accuracy or completeness of this material.
- Materials are provided “as is” without warranty of any kind, either express or implied, including without limitation, warranties of merchantability, fitness for a particular purpose, and non-infringement.
- Under no circumstances shall we be liable for any loss, damage, liability or expense incurred or suffered which is claimed to have resulted from use of this material.

Scenario

- Security is all around…
  - Identification, private communication, electronic signature, access control, electronic purse, digital right management, software protection...

Motivation

- Hardware security and trust now play critical roles as computing is intimately integrated into many infrastructures that we depend on

- Hardware Security
  - dealing with (secret) data in hardware devices
- Hardware Trust
  - dealing with design and manufacturing of devices

Outline

- Hardware Security
  - Side-Channel Attacks
  - Fault Attacks
  - Manufacturing Test Issues

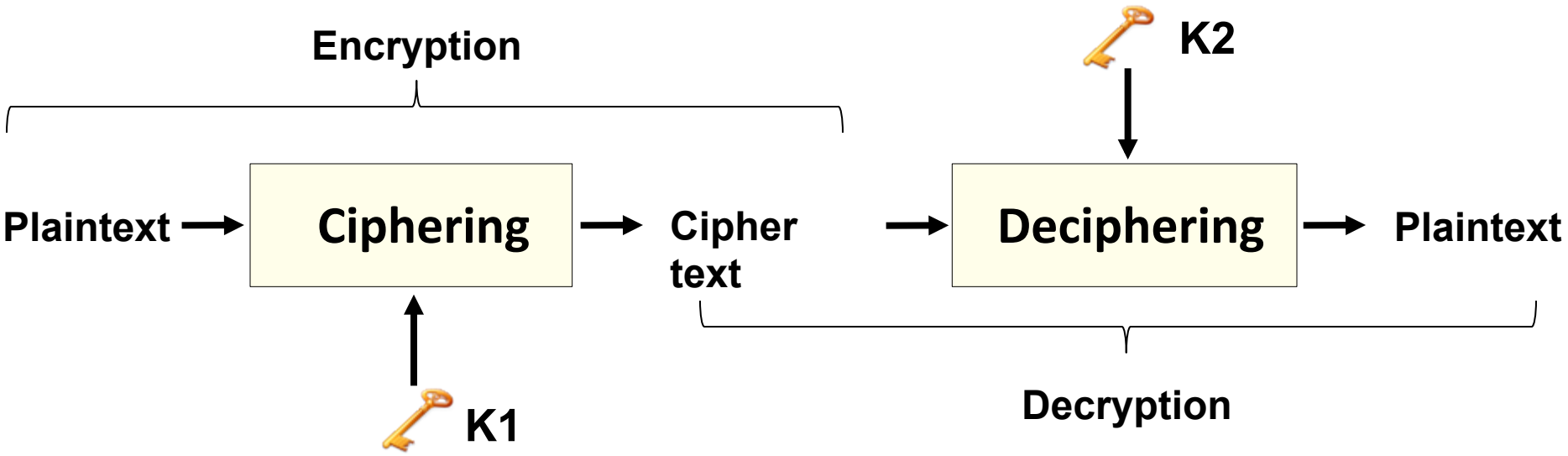
- Hardware Trust
  - Counterfeiting
  - Hardware Trojan Horses

HARDWARE SECURITY

Scenario

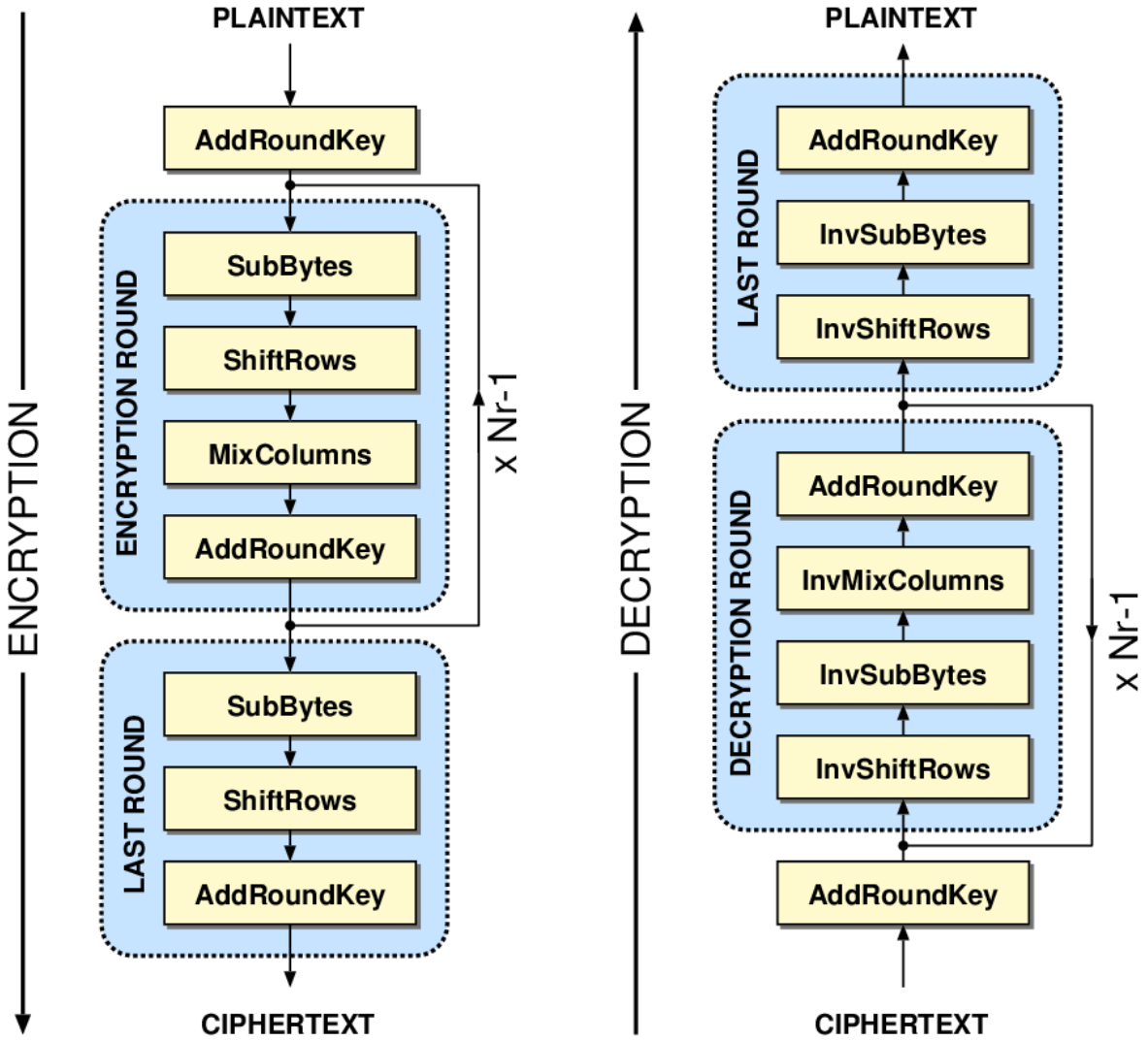
- How to protect a (digital) secret:
  - Secure storage of confidential data
  - Cryptographic capabilities
- Implementation:
  - Crypto algorithms integrated as hardware devices
  - E.g., smartcards, crypto-cores, crypto-processors, hardware security module

Encryption/Decryption

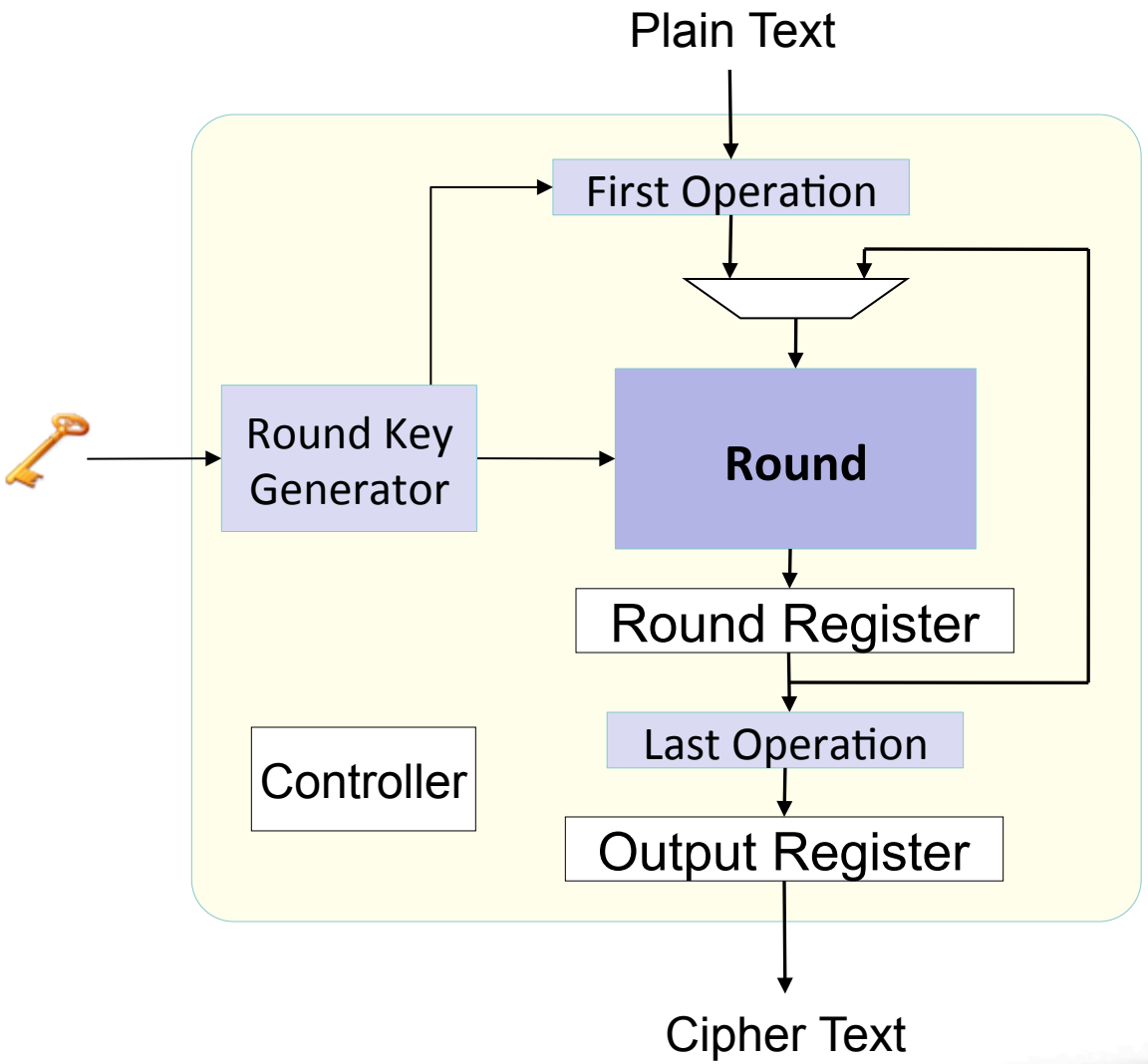


- Algorithms:
  - Symmetric: K1=K2
  - Asymmetric: K1≠K2

Advanced Encryption Standard (AES) - 2001

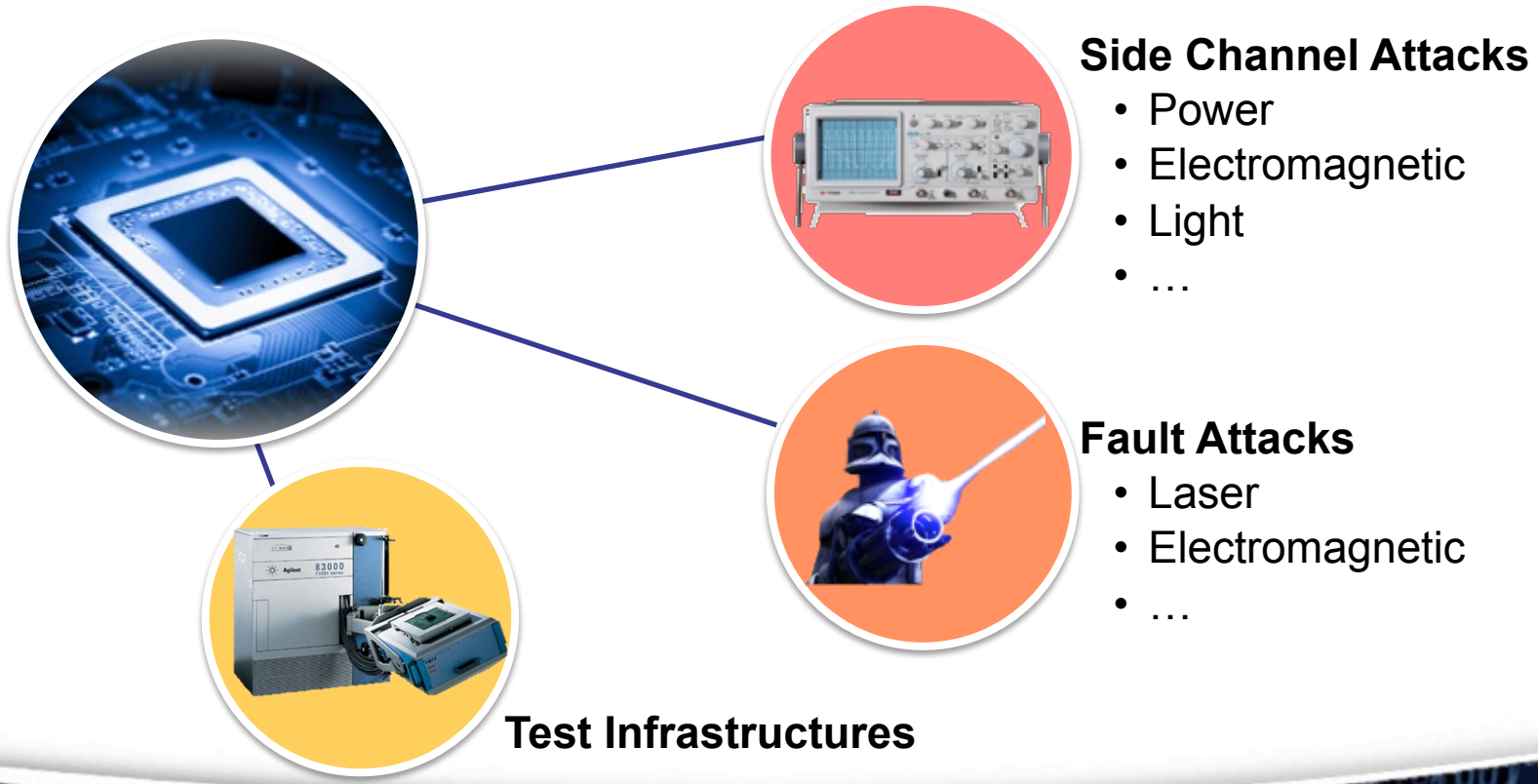


AES: Generic HW Implementation



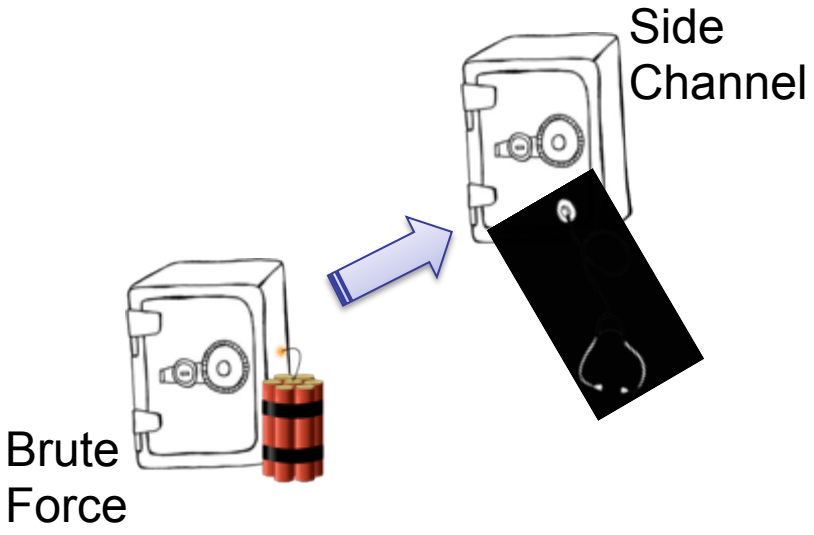
Motivation

- Crypto-algorithms very strong, but…
- New threat: hardware implementation!!



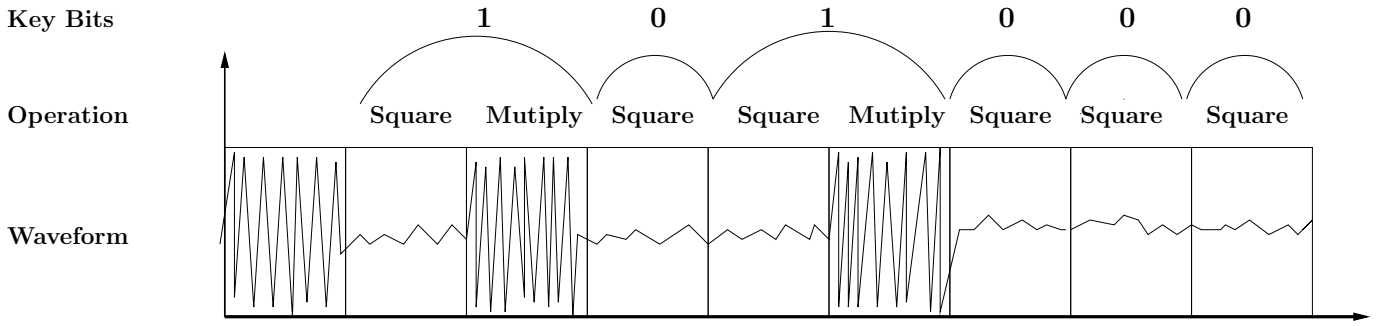
Side-Channel Attacks

- Based on information gained from the non-primary interface of the physical implementation of a cryptosystem
  - Timing information
  - Power consumption
  - Electromagnetic leaks
  - Sound
  - Light
  - …



Simple Power Analysis on RSA

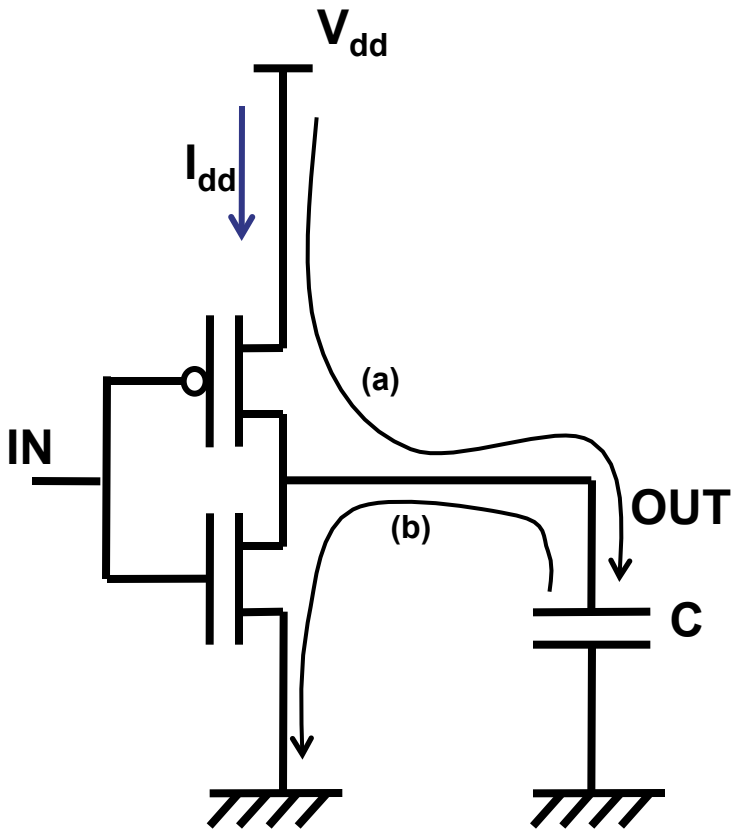
```
Input:    X, N, K=(k_{j-1}, …, k_1, k_0)_2
Output:   Z = X^K mod N

1:    Z = 1;
2:    for i=j-1 downto 0 {
3:        Z = Z * Z mod N //Square
4:        if (k_i==1) {
5:            Z = Z * X mod N //Multiply
6:        }
7:    }
```

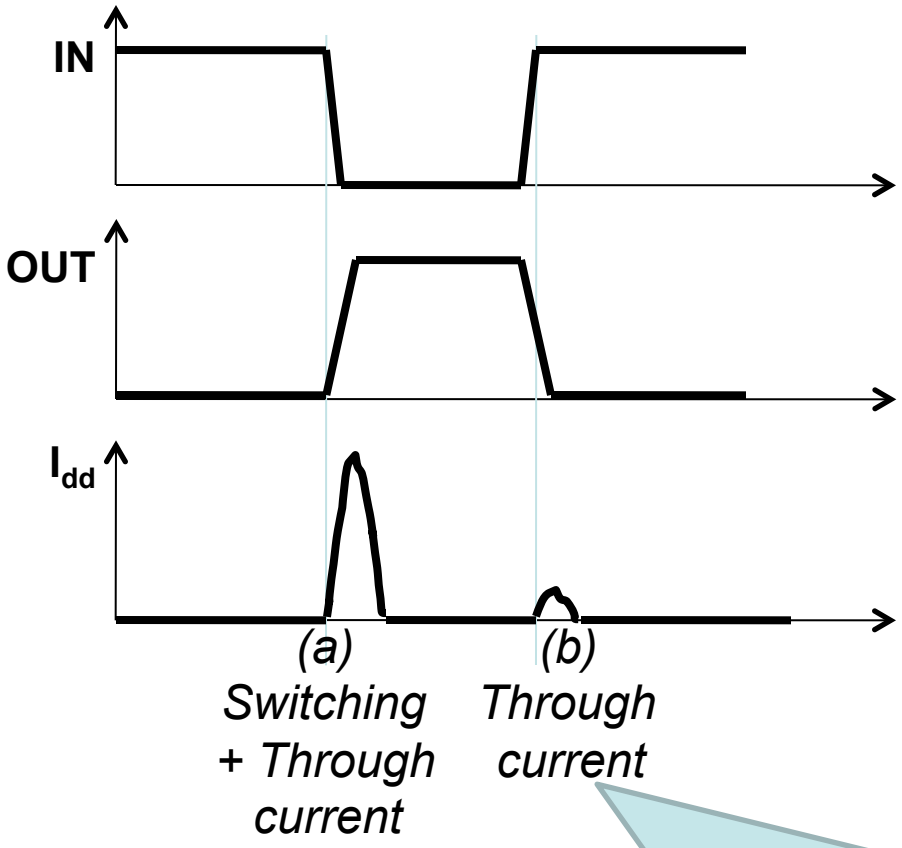


Differential Power Analsys

- Differential Power Analysis (DPA) attacks
  - it requires the knowledge of the algorithm but not its physical implementation
  - it is cheap and easy to perform
- The basic idea is to correlate the power consumed by the device and the encryption data including the key

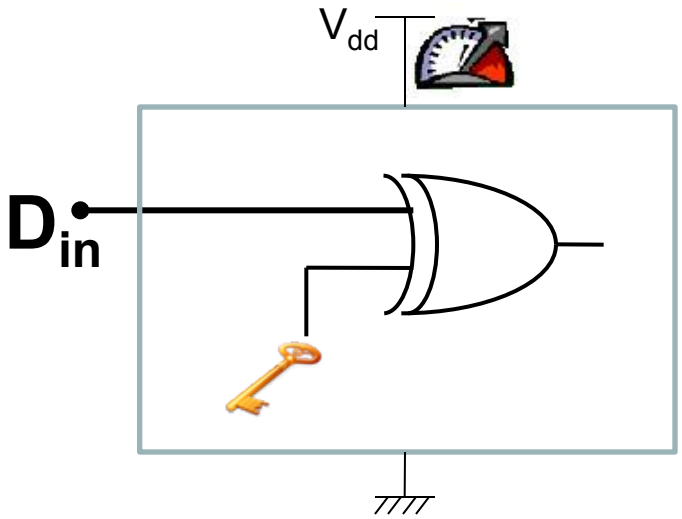
Power Consumption of CMOS

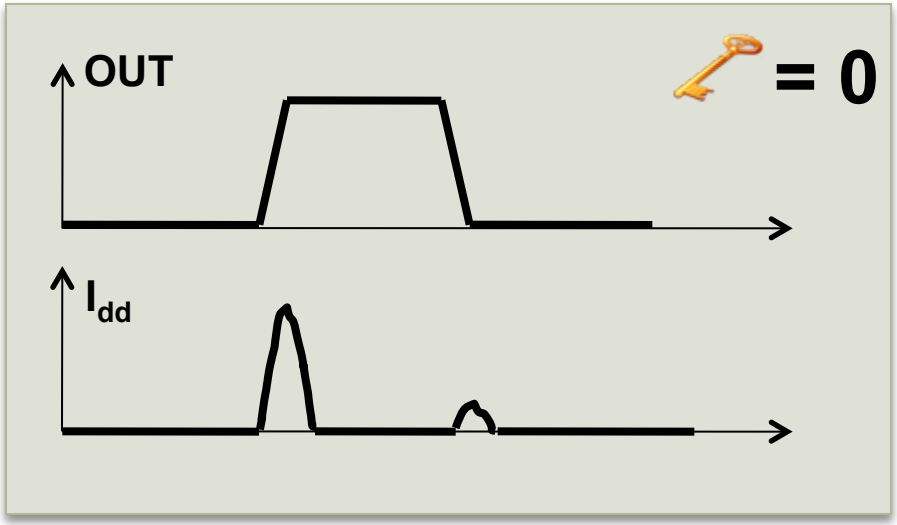


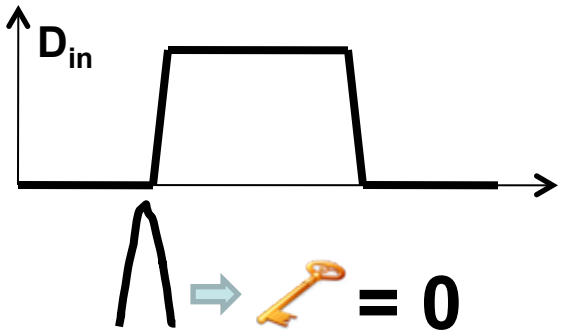


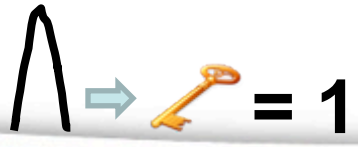
Current that flows from Vdd to GND when the p-channel transistor and n-channel transistor turn on briefly at the same time during the logic transition

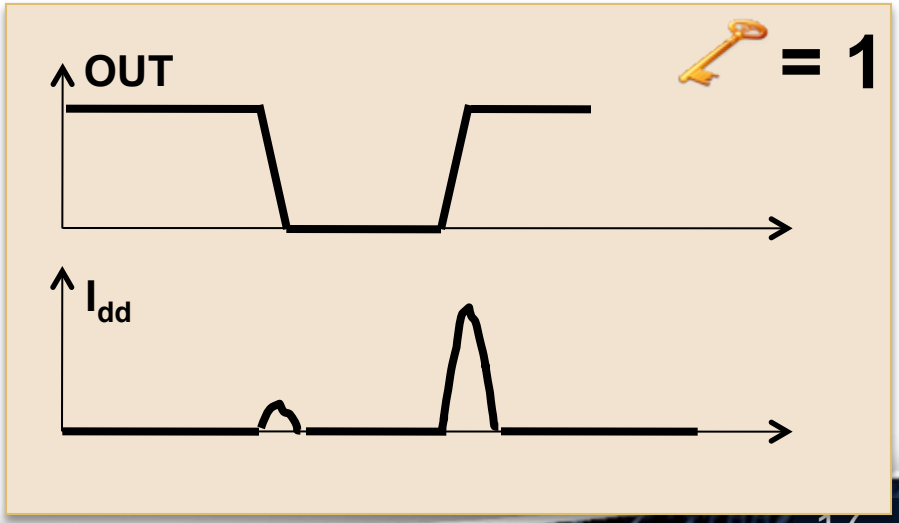
Power attack on XOR gate









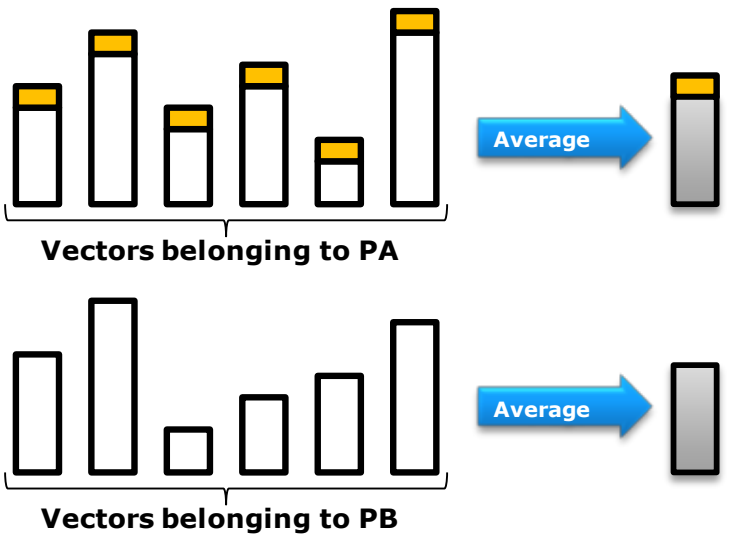


Input patterns

- **Target node**: output of a gate within the circuit
- Sequence of input patterns:
  - $P_0, P_1, \ldots, P_n$
  - $T_1(P_0 \rightarrow P_1), T_2(P_1 \rightarrow P_2), \ldots, T_n(P_{n-1} \rightarrow P_n)$
- Logic simulation to create two sets:
  - (PA) transitions that make the target node to commute from 0 to 1 and therefore that make the target gate to consume
  - (PB) transitions that do not lead the target gate to participate to the power consumed by the circuit

DPA: Basic Principle

- We classified the two sets in such a way that the set PA always leads to a component of power consumption that is not present in the set PB

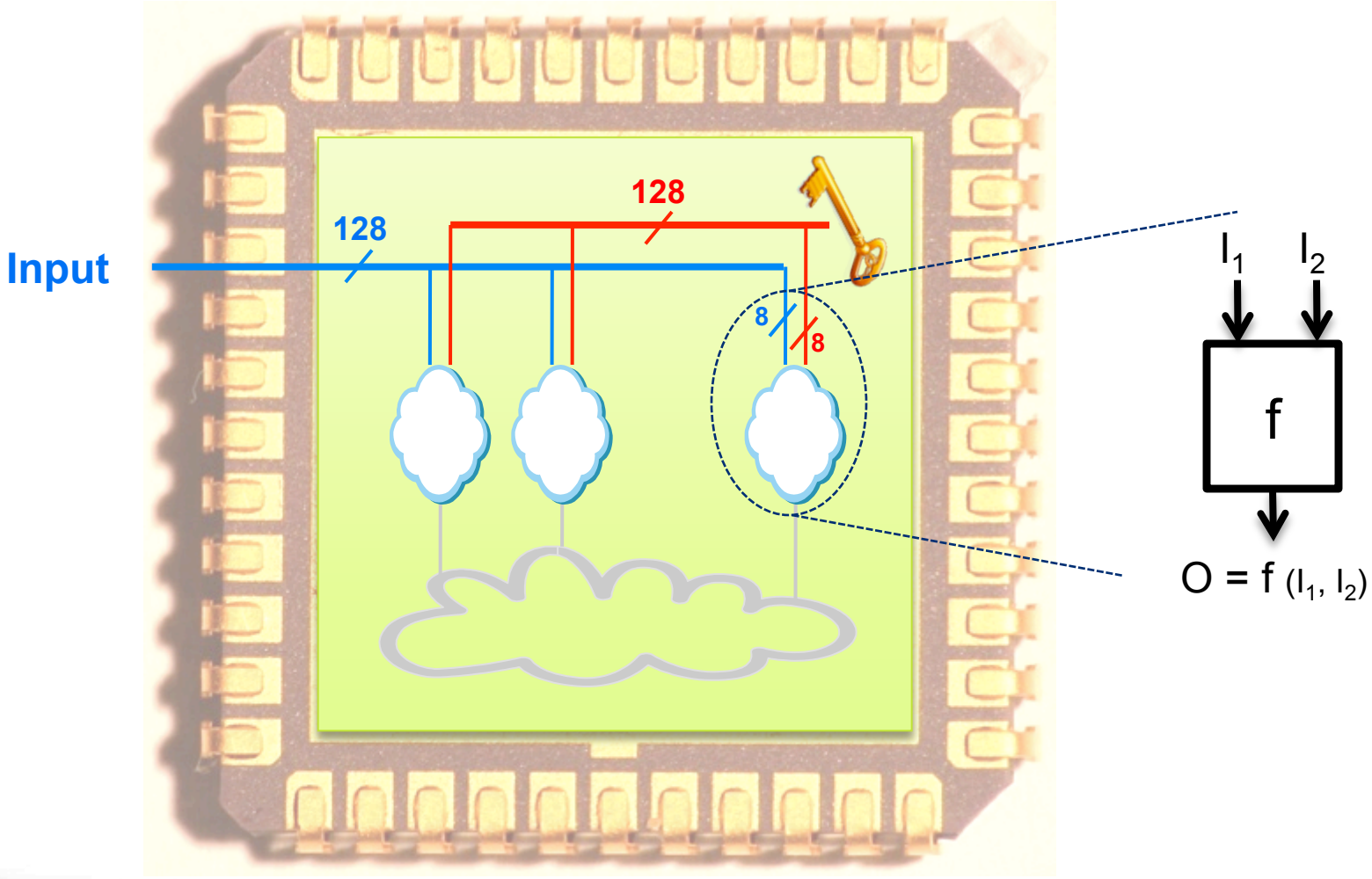


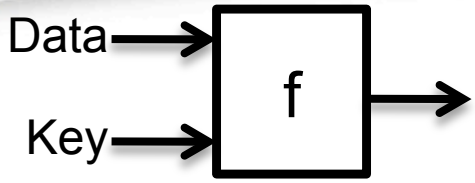
→ Power(PA) > Power(PB)

DPA: Performing the Attack

- Target node:
  - output of a gate whose value depends on both the plain text under ciphering (primary inputs) and the secret key
- Basic idea:
  - All the key suppositions
  - For each key guess, creation of the two sets PA and PB

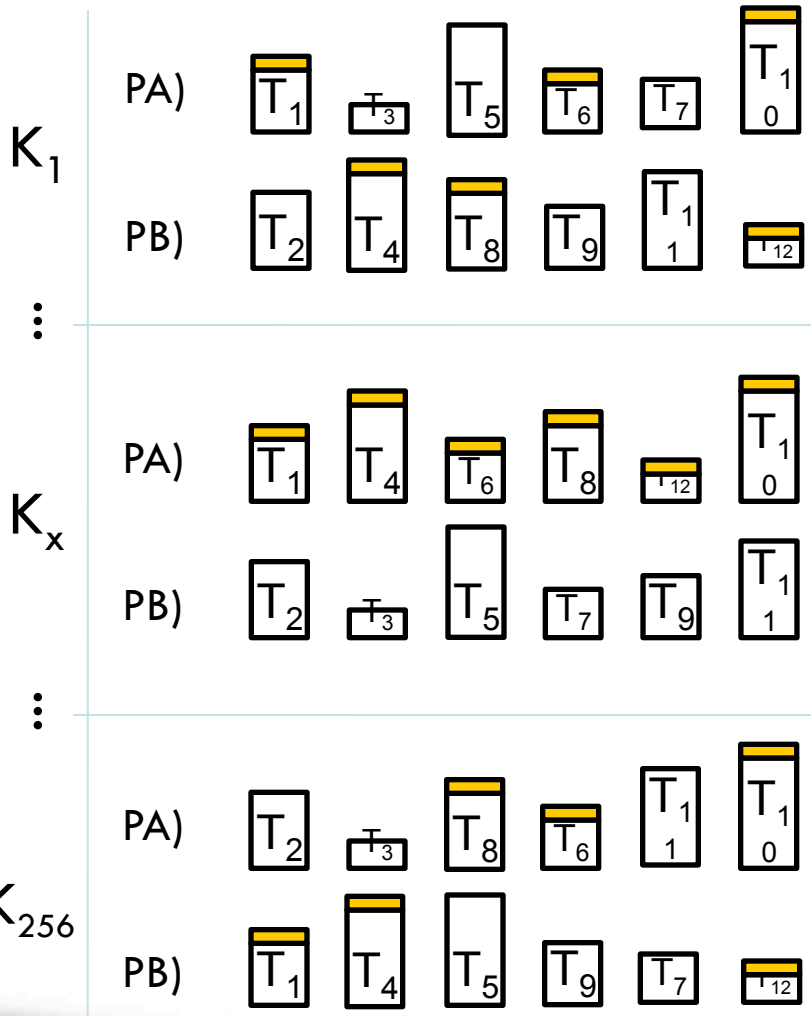
DPA: Target Node





O = f (Key, Data)

Inputs: T_1, T_2, …, T_{12}
Key: K_x (8 bits)

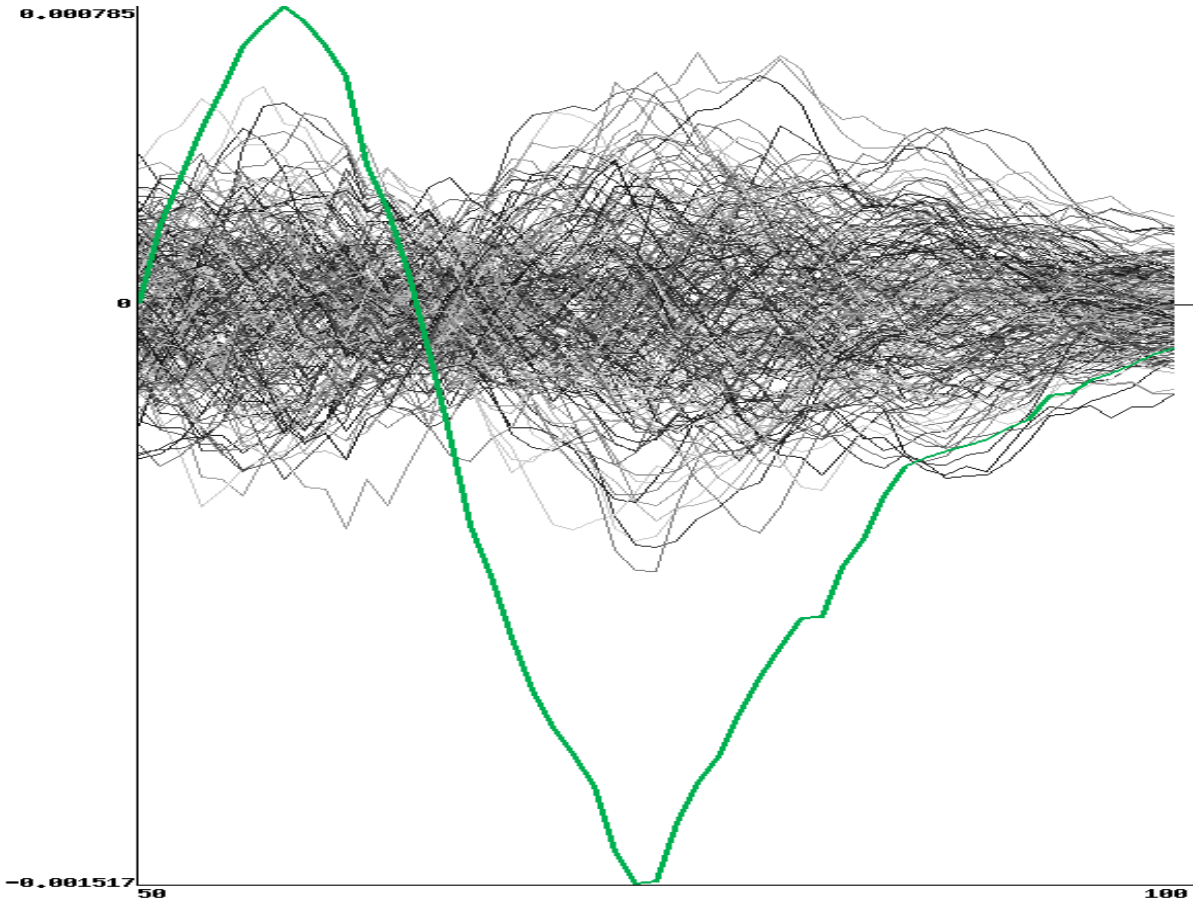


$\overline{PA} - \overline{PB} \cong 0$

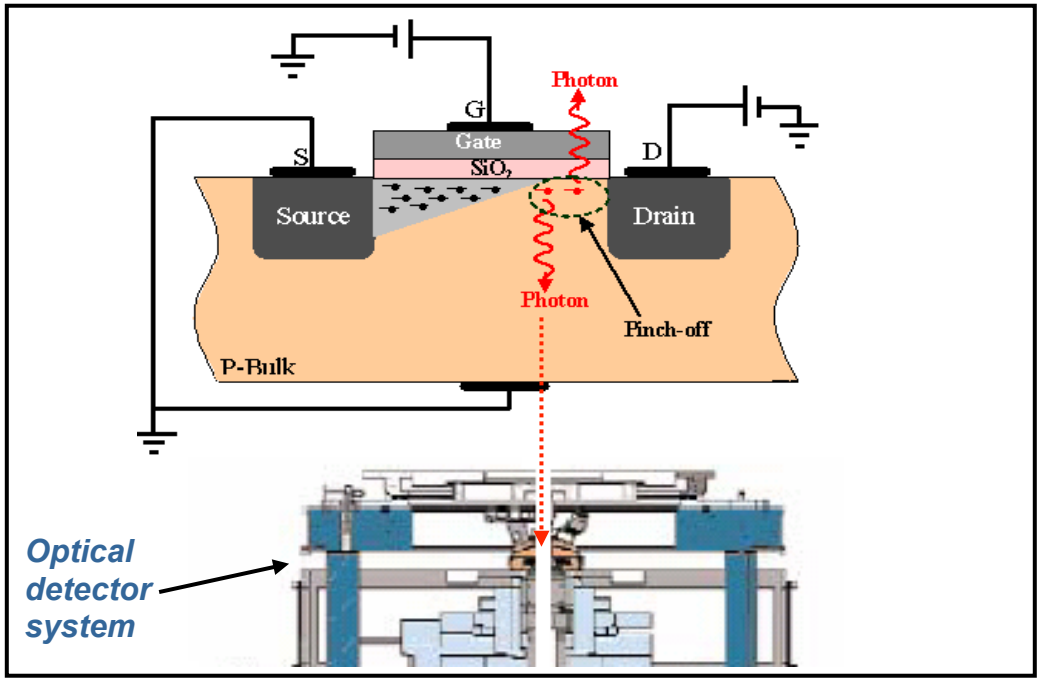
$\overline{PA} - \overline{PB} =$

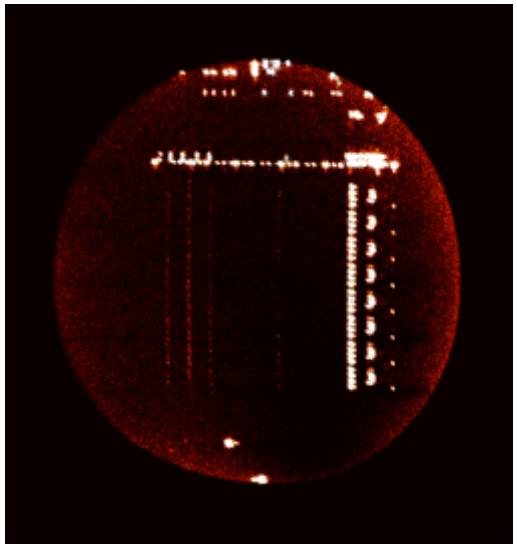
$\overline{PA} - \overline{PB} \cong 0$

DPA: Result



Light Emission





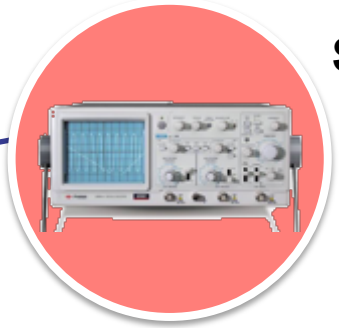
http://www.vvku.eu/cv/efa/background.html

Countermeasures

- Goal: removing the correlation between processed data and the physical interface
- Methods:
  - Adding noise (random power consumption, clock jitter, logic masking, random dummy calculations, …)
  - Making constant the physical interface (dual logic, triple logic, …)

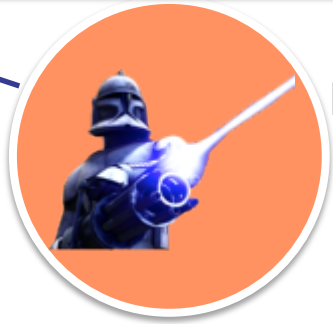
Motivation

- Crypto-algorithms very strong, but…
- New threat: hardware implementation!!



Side Channel Attacks

- Power
- Electromagnetic
- Light
- …

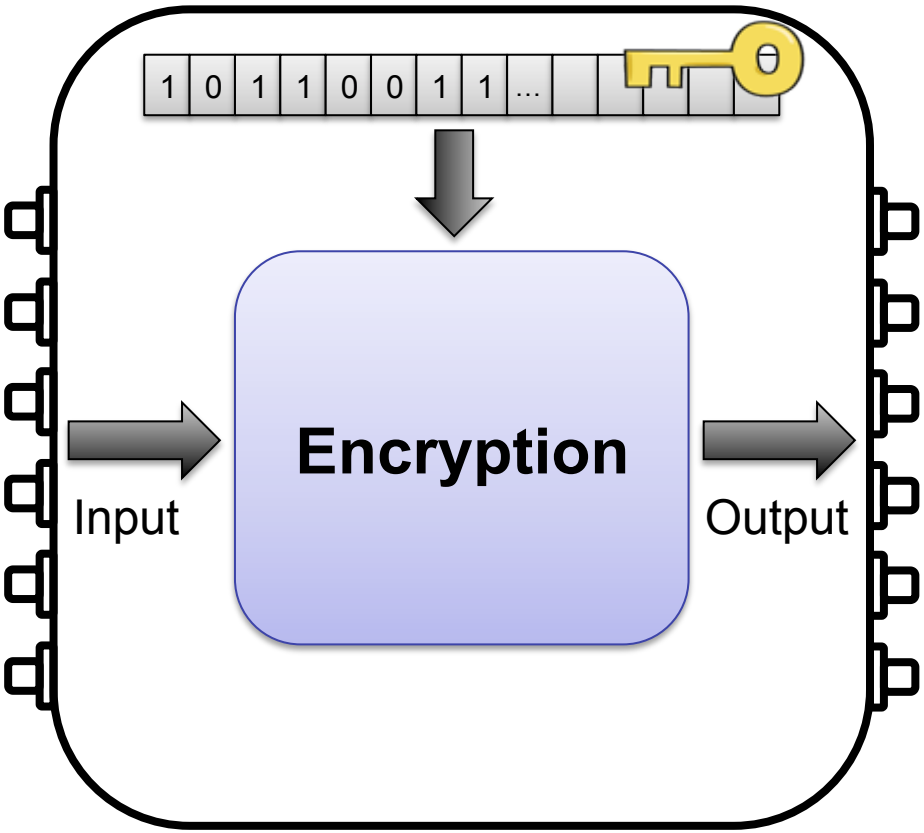


Fault Attacks

- Laser
- Electromagnetic
- …

Test Infrastructures

Fault Attacks



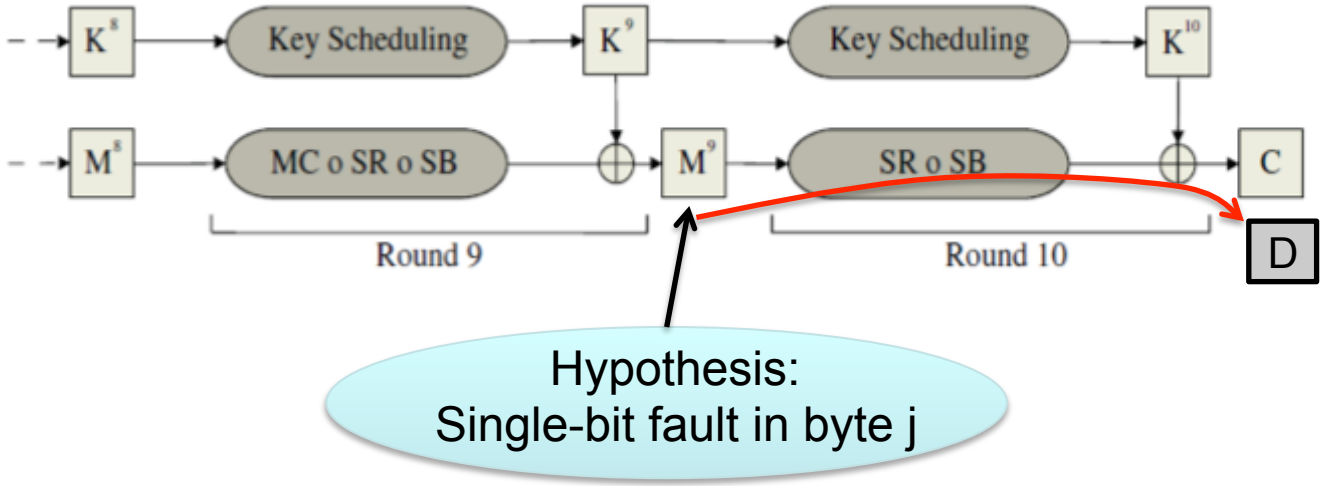
Hypothesis: Injection forces a '0' on a single bit of the secret key

1) $C_{OK}=E(P)$

2) Calculate C'=E(P), while injecting a fault

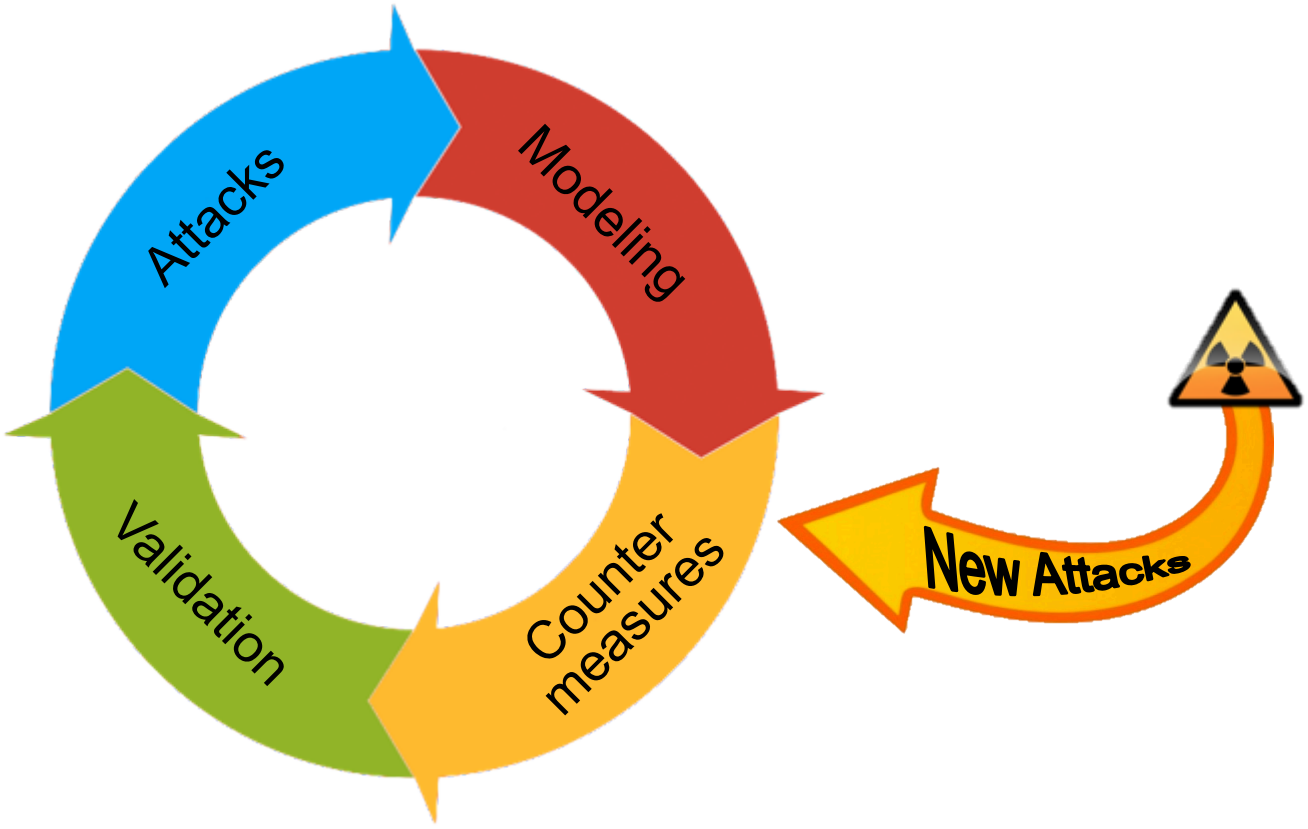
3) If C' = C_{OK} → target bit is '0' else → target bit is '1'

Differential Fault Attacks

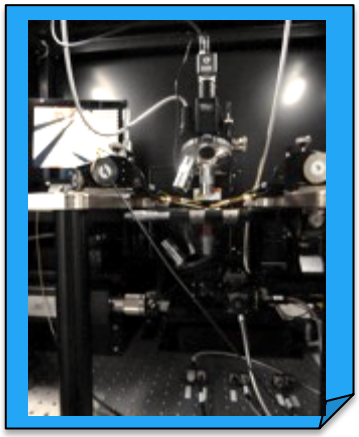


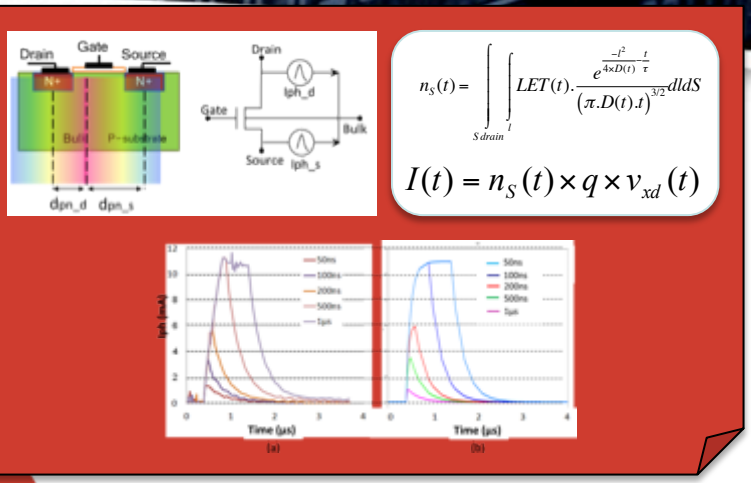
$$C_{SR(j)} \oplus D_{SR(j)} = SubByte\left(M_j^9\right) \oplus SubByte\left(M_j^9 \oplus e_j\right)$$

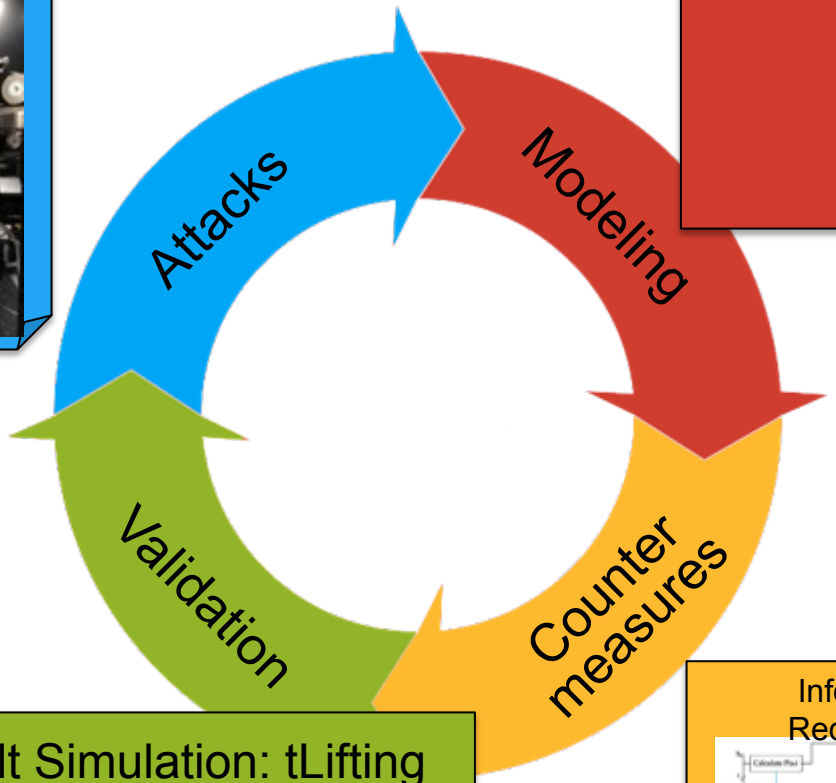
Fault Attacks



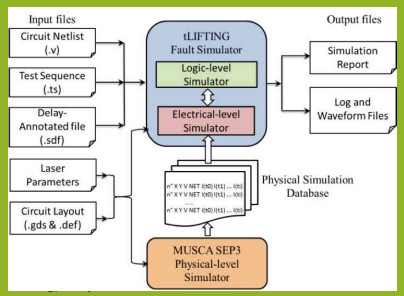
Fault Attac

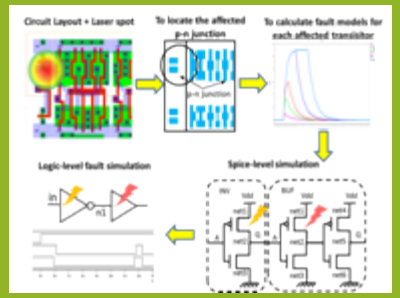






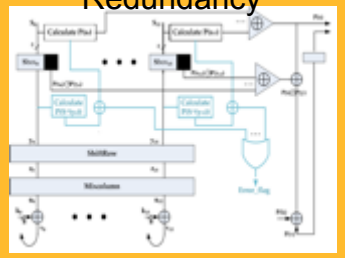
Validation via Fault Simulation: tLifting

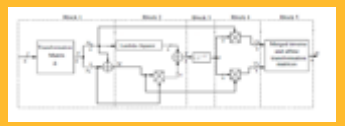




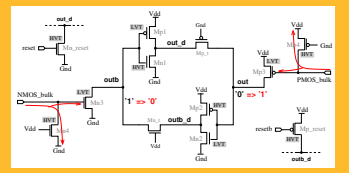
New Attacks

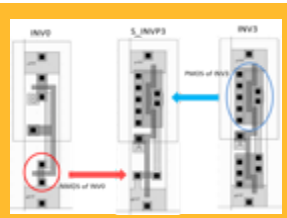
Information Redundancy





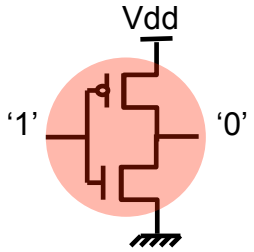
Detectors

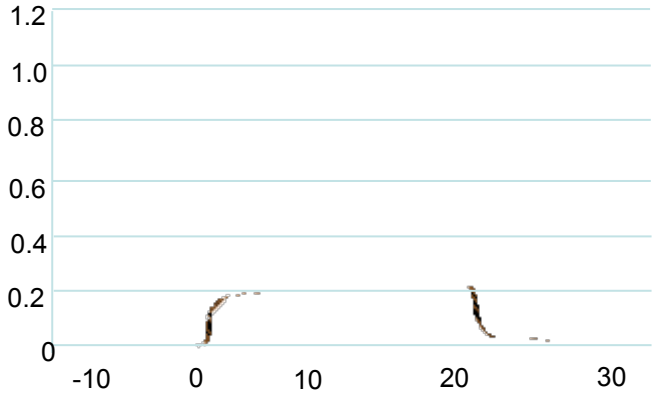




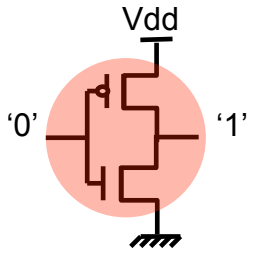
Laser Detector

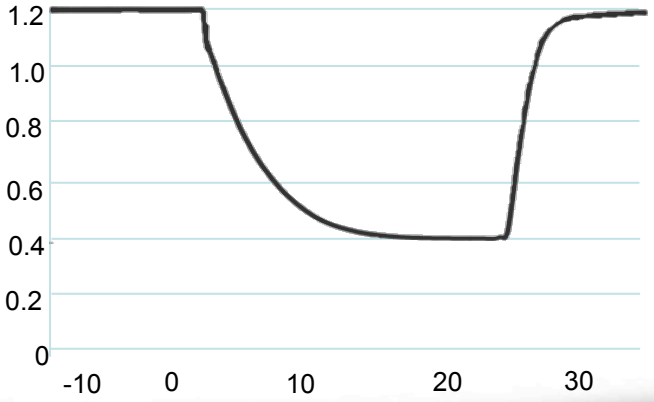
- Basic idea: using inverters to detect laser injections



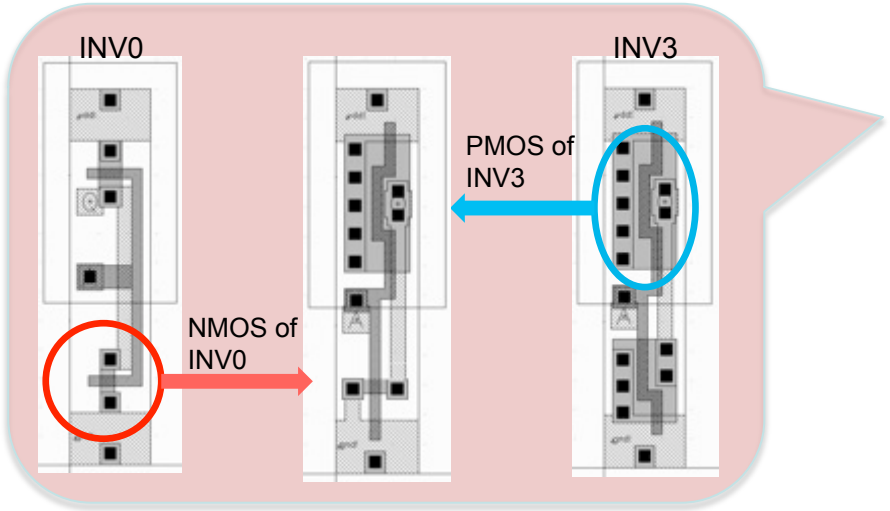


Laser spot = 3.25µm
Laser power = 1.0w
Technology = 90nm ST



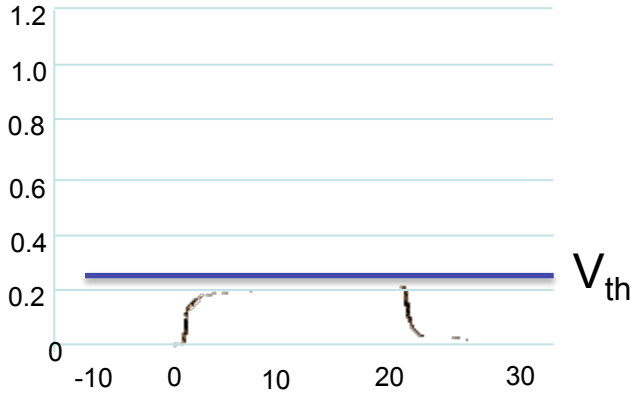


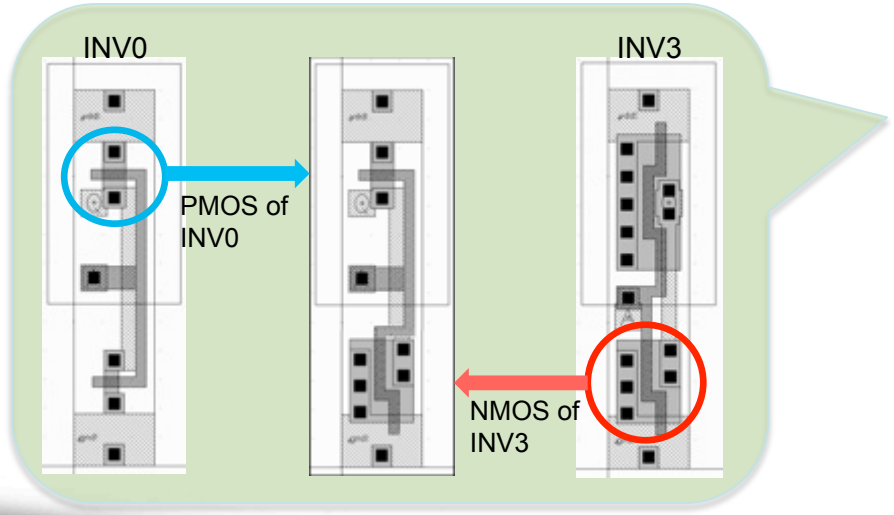
Laser Detectors



S_INVP3

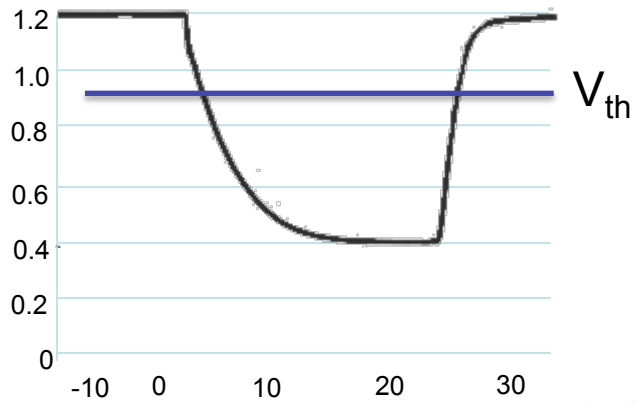
1



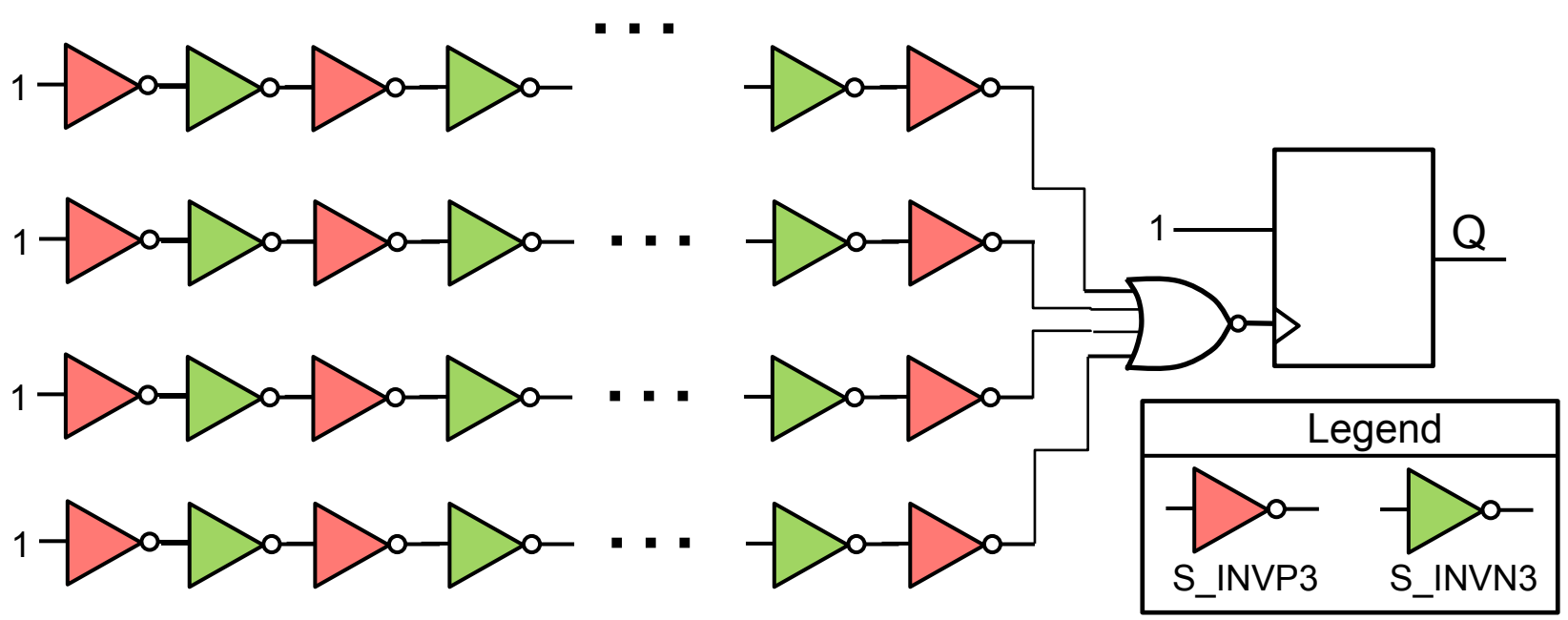


S_INVN3

0

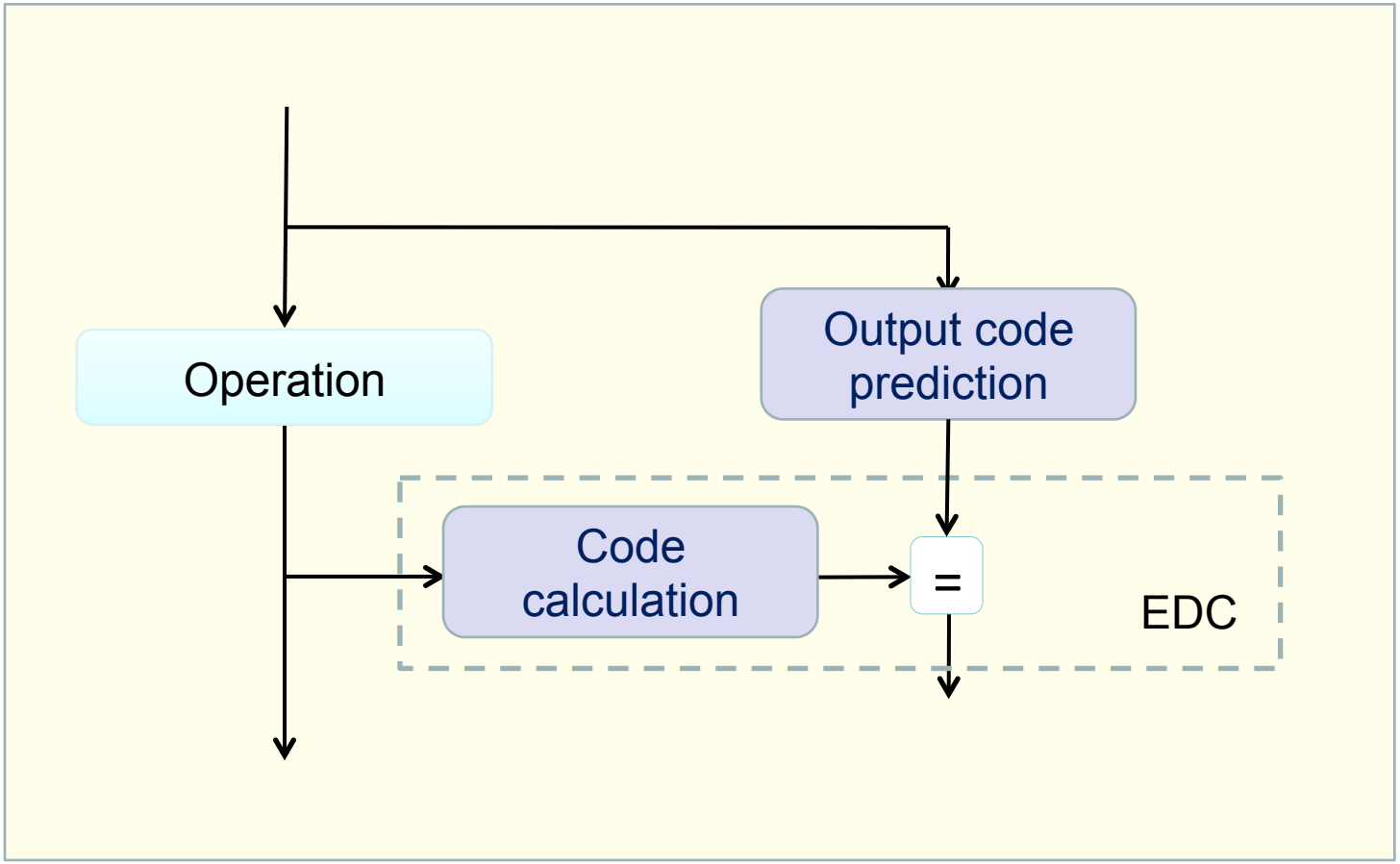


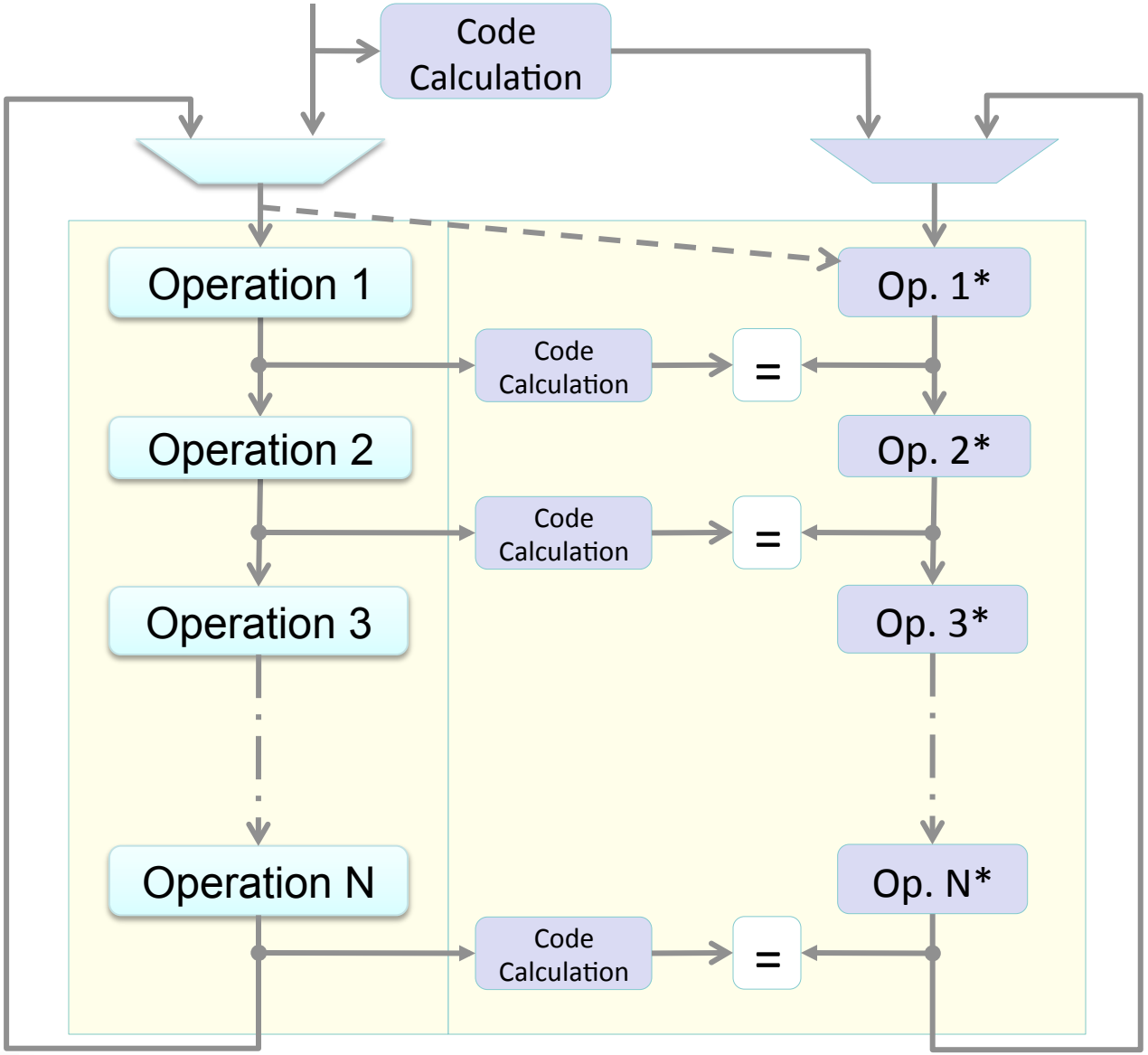
Proposed Solution



Feng LU, Giorgio DI NATALE, Marie-Lise FLOTTES, Bruno ROUZEYRE,
Customized Cell Detector for Laser-Induced-Fault Detection,
IEEE IOLTS 2014

Error Detection Codes



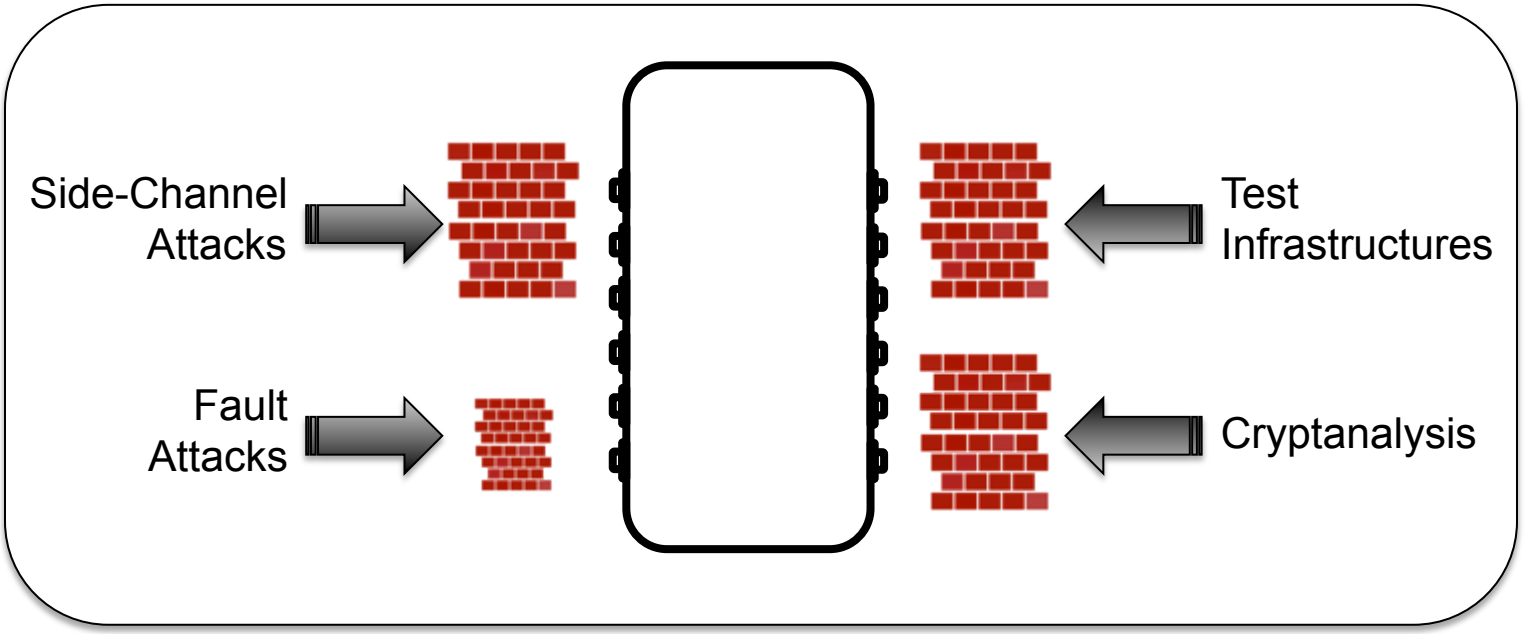

Code Calculation
Operation 1
Op. 1*
Code Calculation
=
Operation 2
Op. 2*
Code Calculation
=
Operation 3
Op. 3*
Operation N
Op. N*
Code Calculation
=

Which code ?

- Linear EDC: parity bits
  - 1 parity bit for 128 data bits
  - 1 parity bit for each data byte
  - 1 parity bit for each state "column"

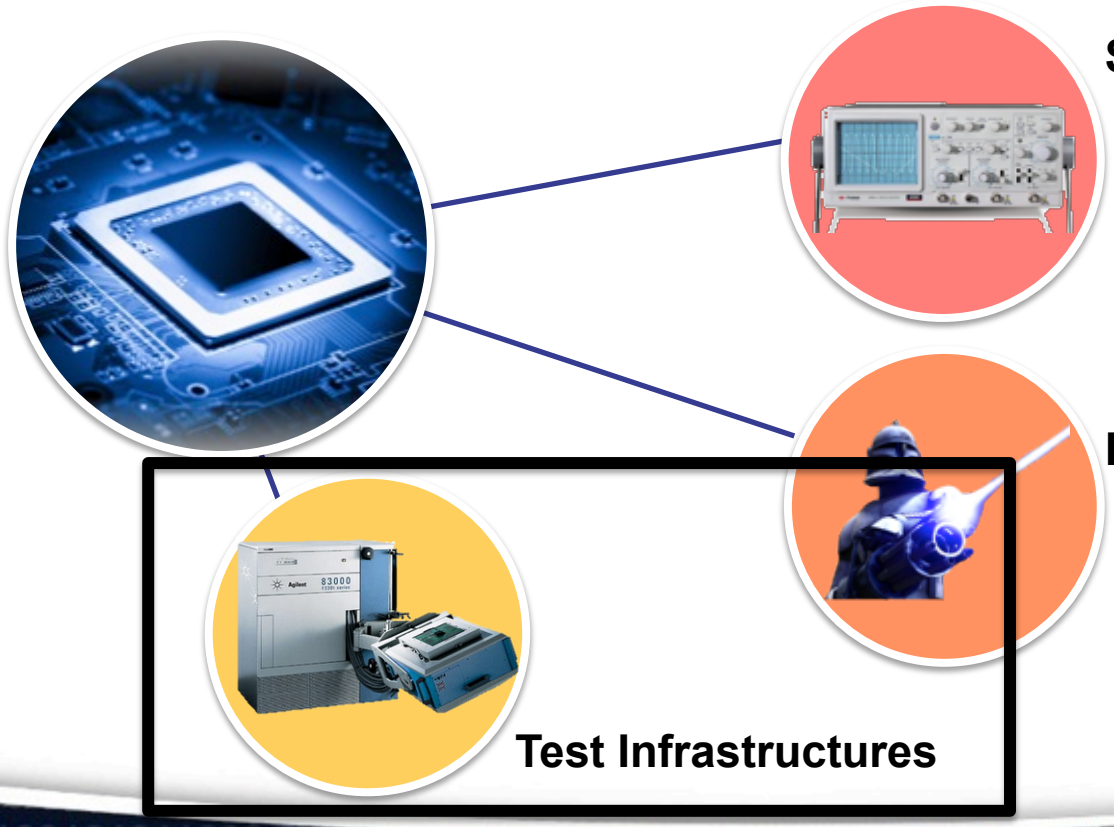
  - SubByte non linear

Fault Attacks vs Side-Channel



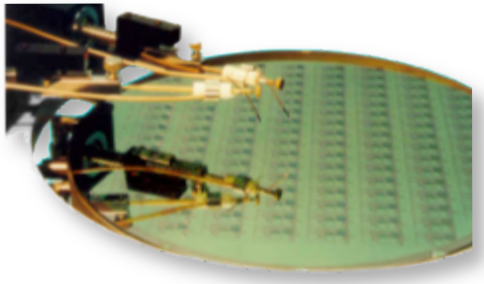
Motivation

- Crypto-algorithms very strong, but…
- New threat: hardware implementation!!

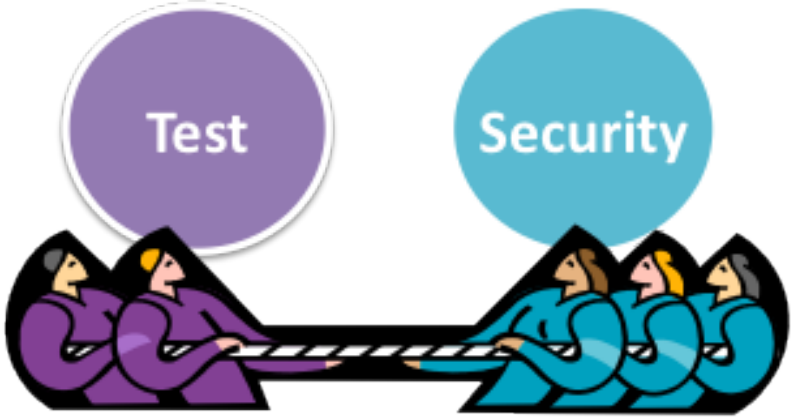


Manufacturing Test

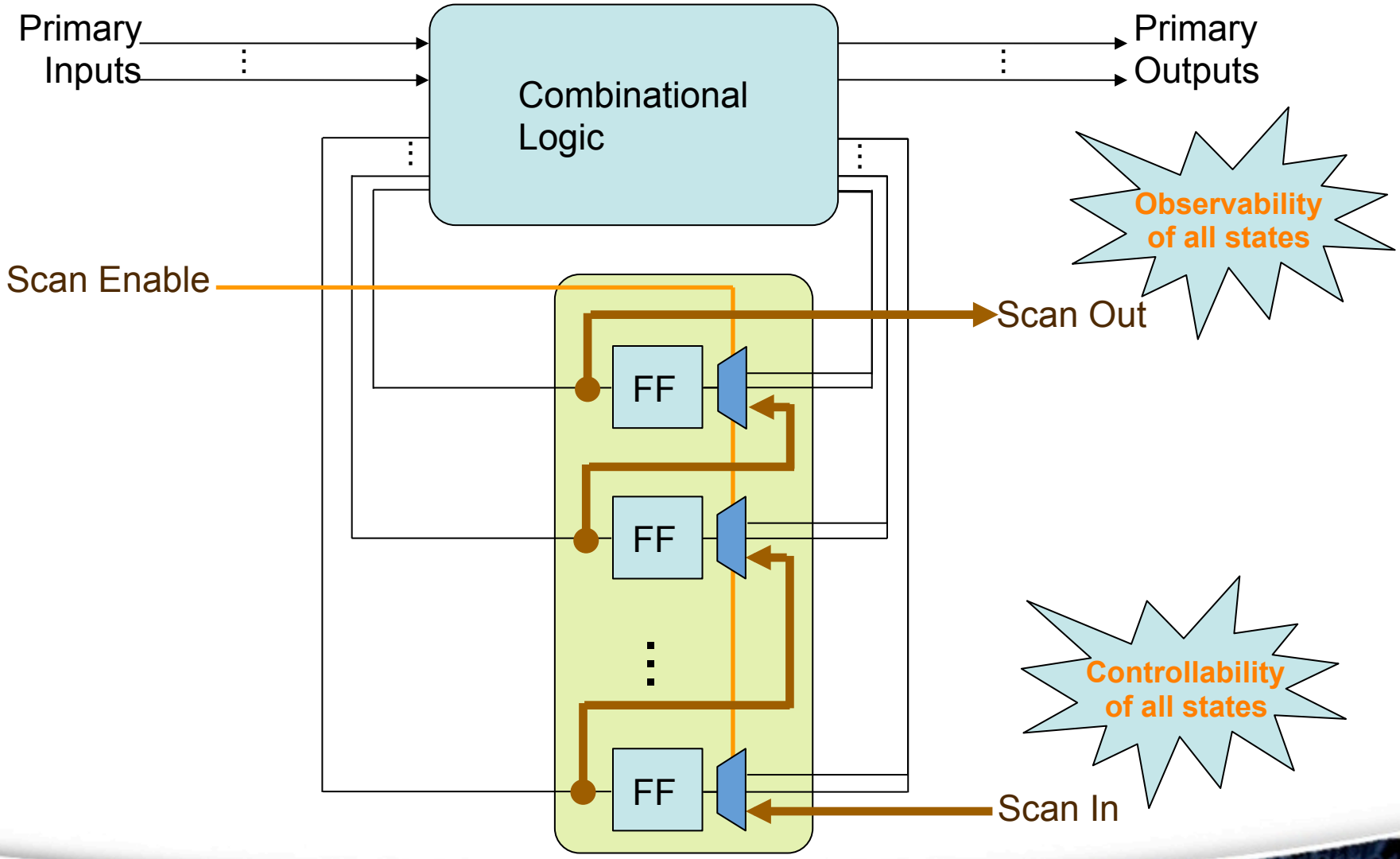
- Circuit testing is mandatory to guarantee high quality of digital ICs
  - Increase controllability and observability



- On the contrary, security fears testability
  - Test infrastructure used for attacks

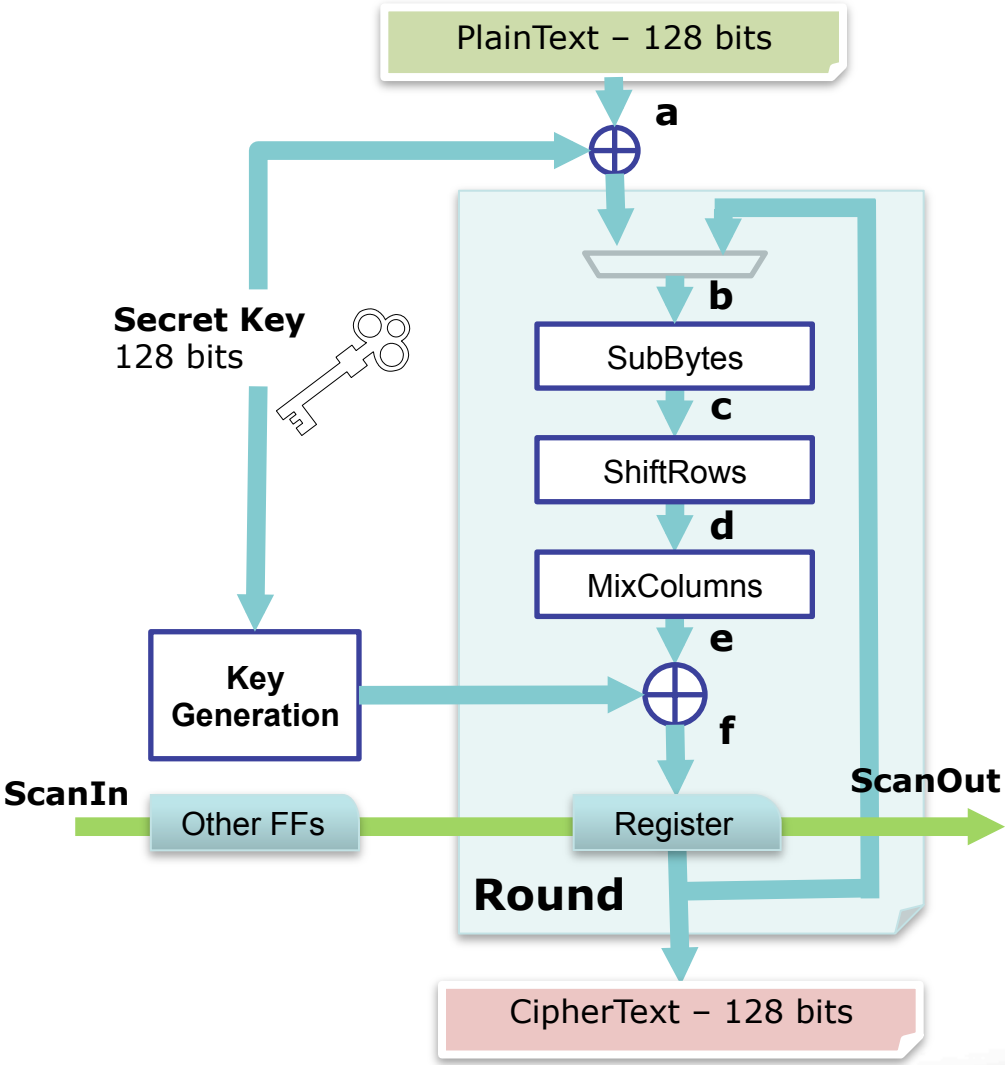


Scan-based Design



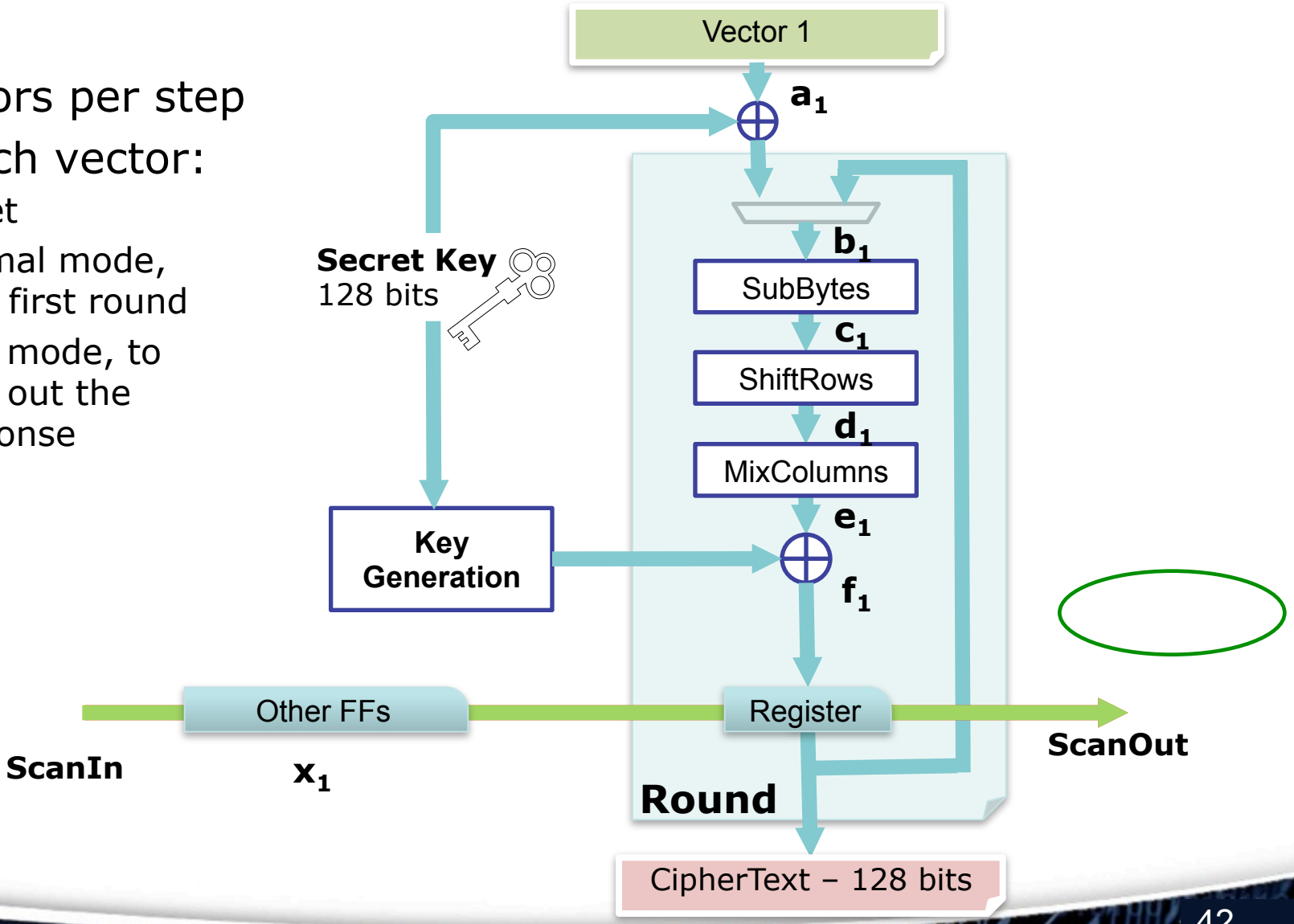
Scan-chain attack on AES

- AES
  - Secure after 10 rounds
- Hypothesis:
  - Round-register belongs to the scan chain
  - Attacker controls test mode
  - Deterministic circuit
  - Other FFs belong to the scan chain
- Attack:
  - Accessing the scan chain after the first round



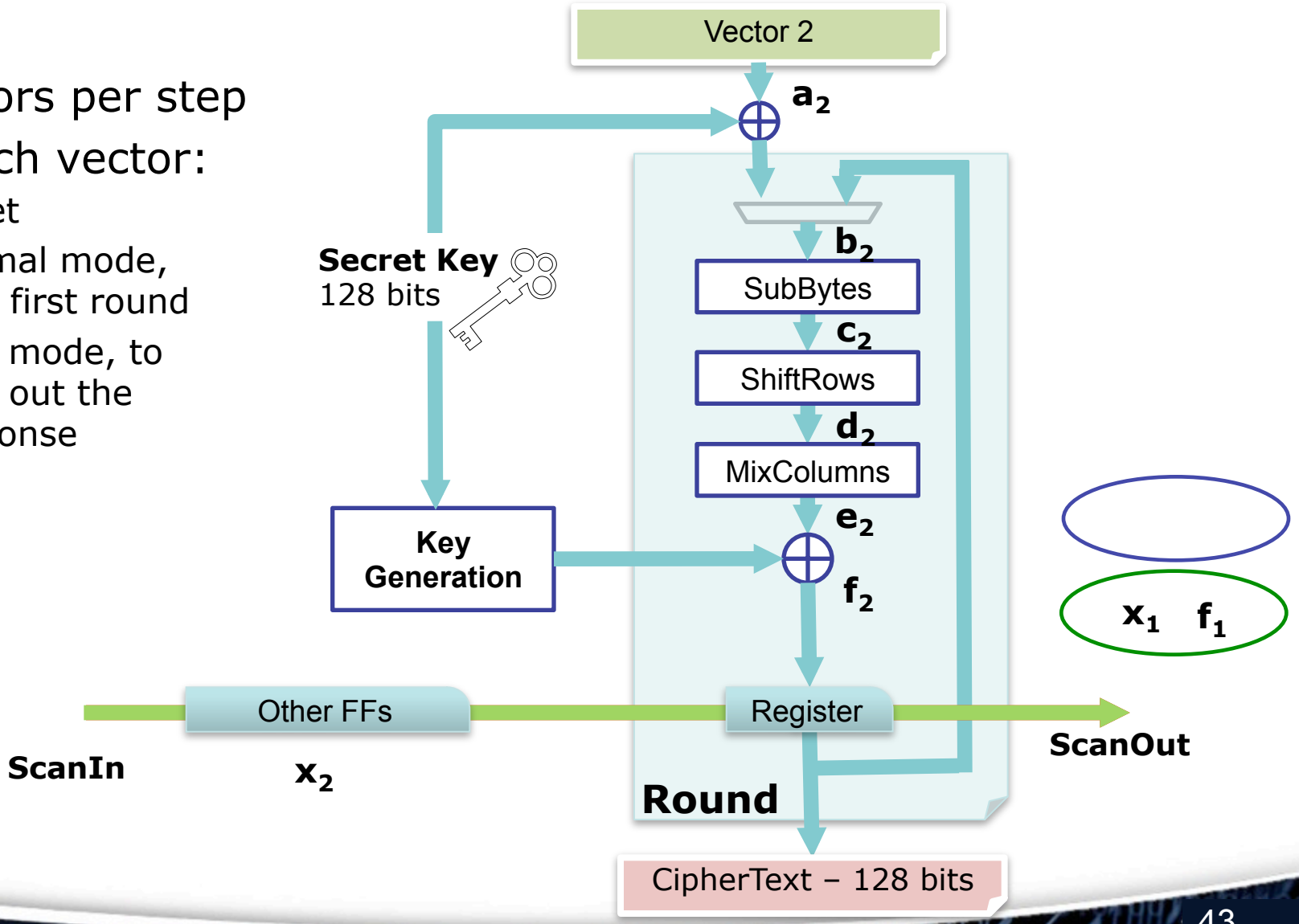
Differential Scan Attack

- 2 vectors per step
- For each vector:
  - Reset
  - Normal mode, until first round
  - Test mode, to shift out the response



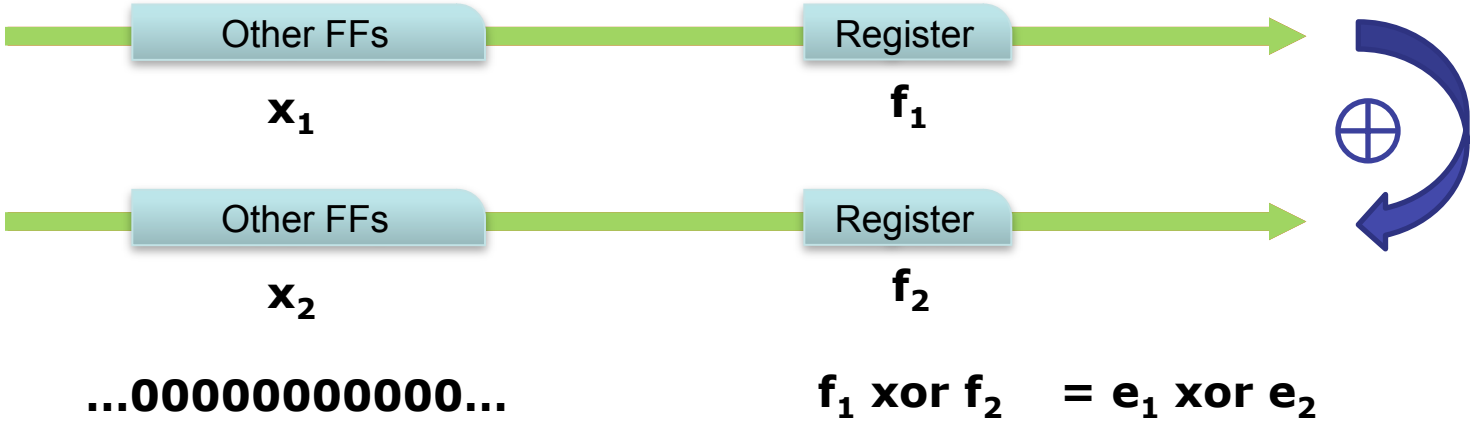
Differential Scan Attack

- 2 vectors per step
- For each vector:
  - Reset
  - Normal mode, until first round
  - Test mode, to shift out the response



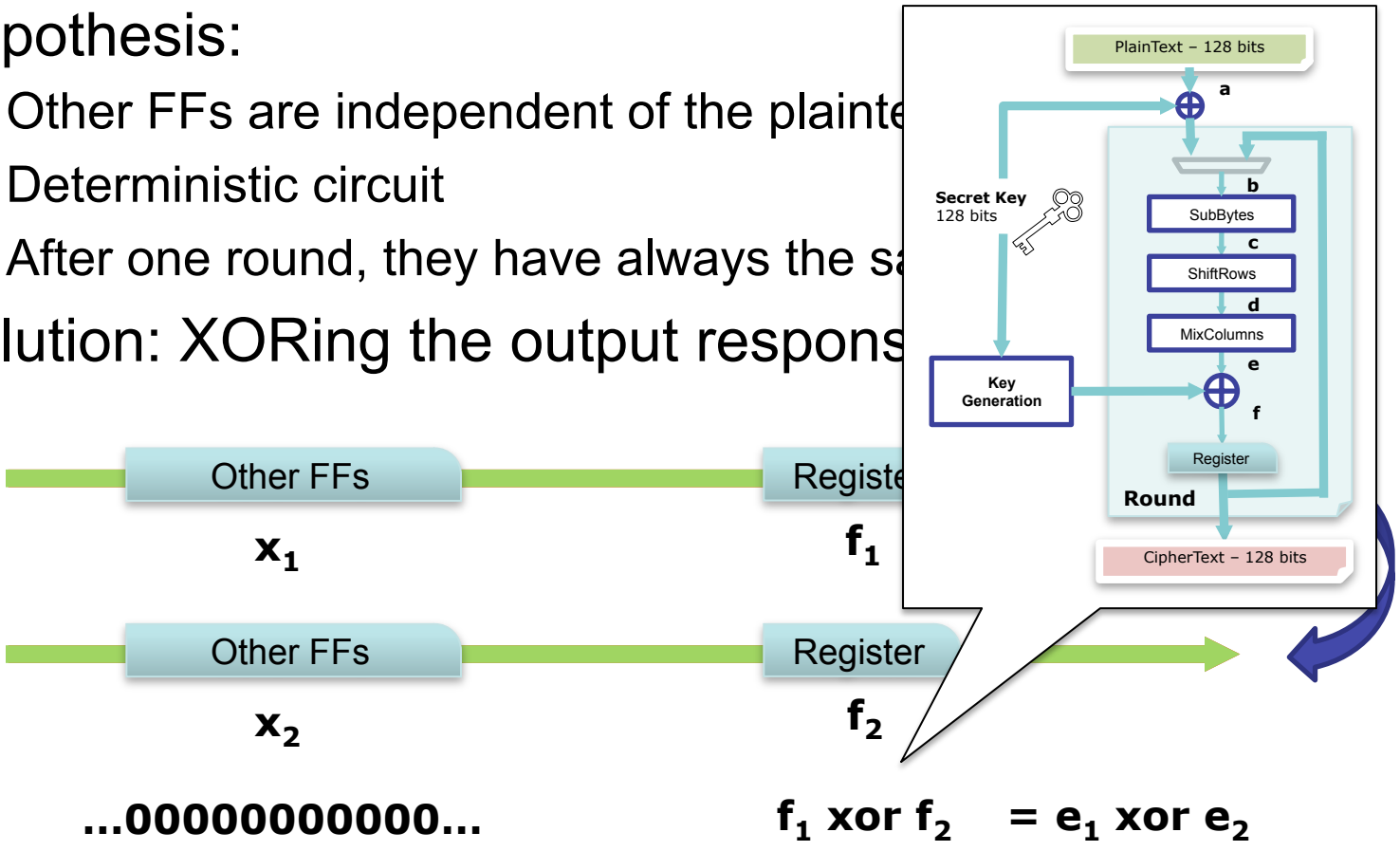
Avoiding other FFs

- Hypothesis:
  - Other FFs are independent of the plaintext
  - Deterministic circuit
  - After one round, they have always the same value
- Solution: XORing the output response



Avoiding other FFs

- Hypothesis:
  - Other FFs are independent of the plainte
  - Deterministic circuit
  - After one round, they have always the sa
- Solution: XORing the output respons



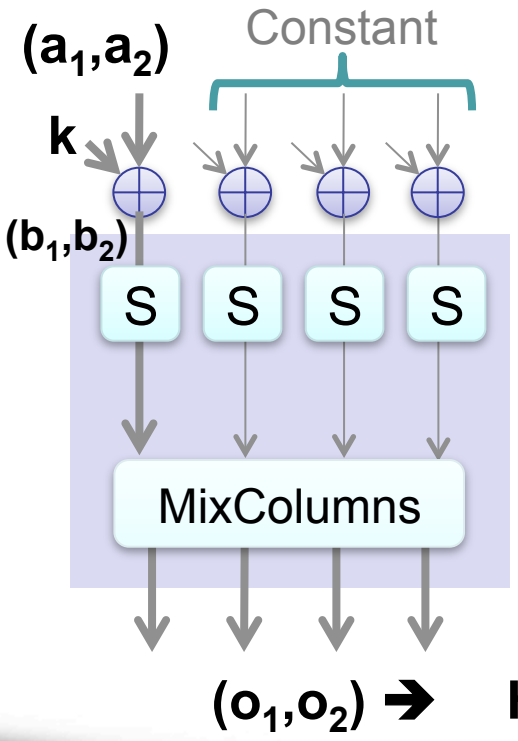
Avoiding the chain

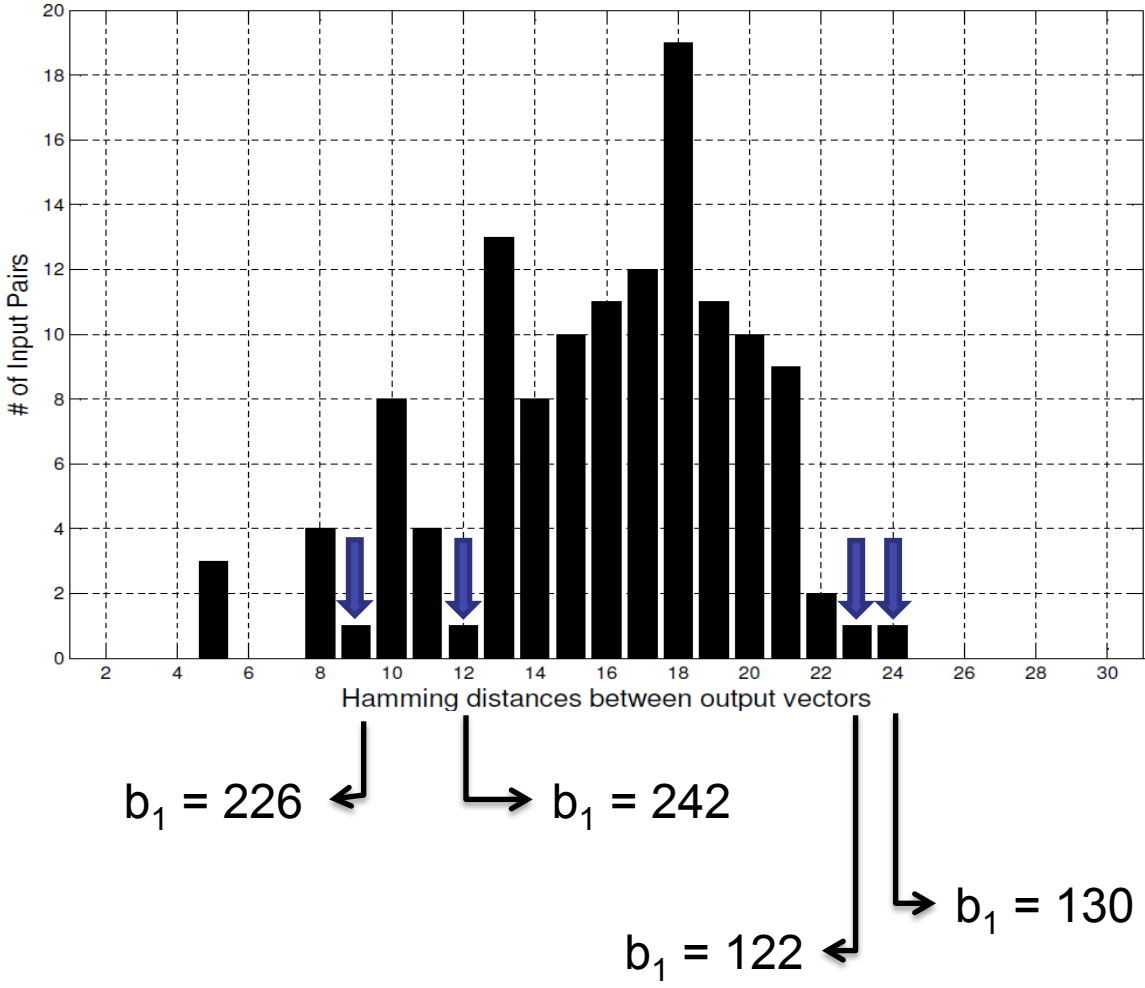
- Position of FFs within the scan chain is not known

- OutScanChain = array [N]
- We need a function such that:
  - p = f (OutScanChain)
  - f : array → number

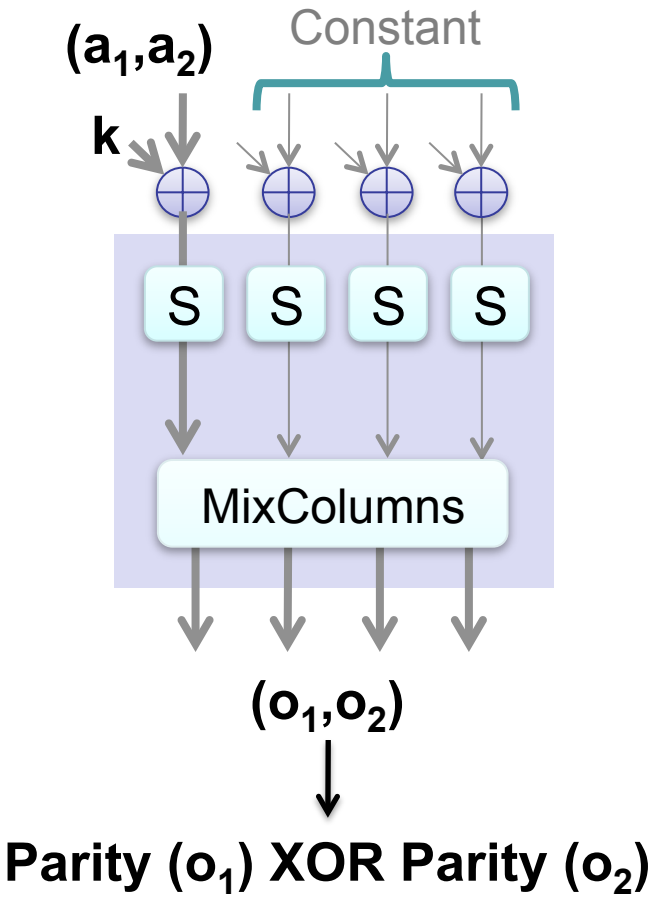
Hamming Distance

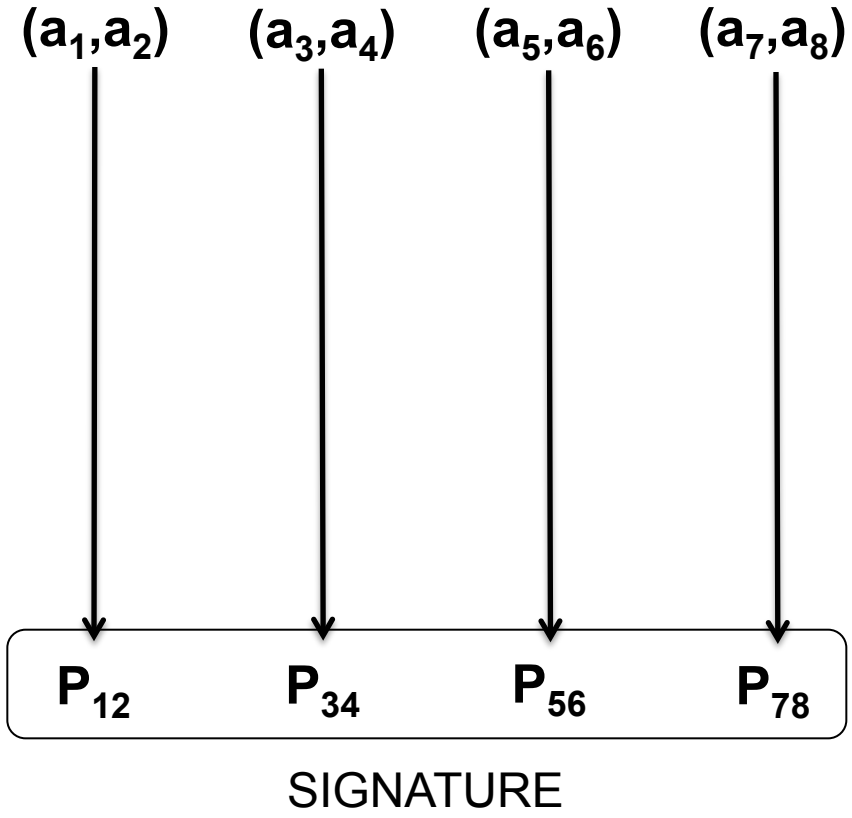
$(b_1, b_1 \text{ XOR } 1)$





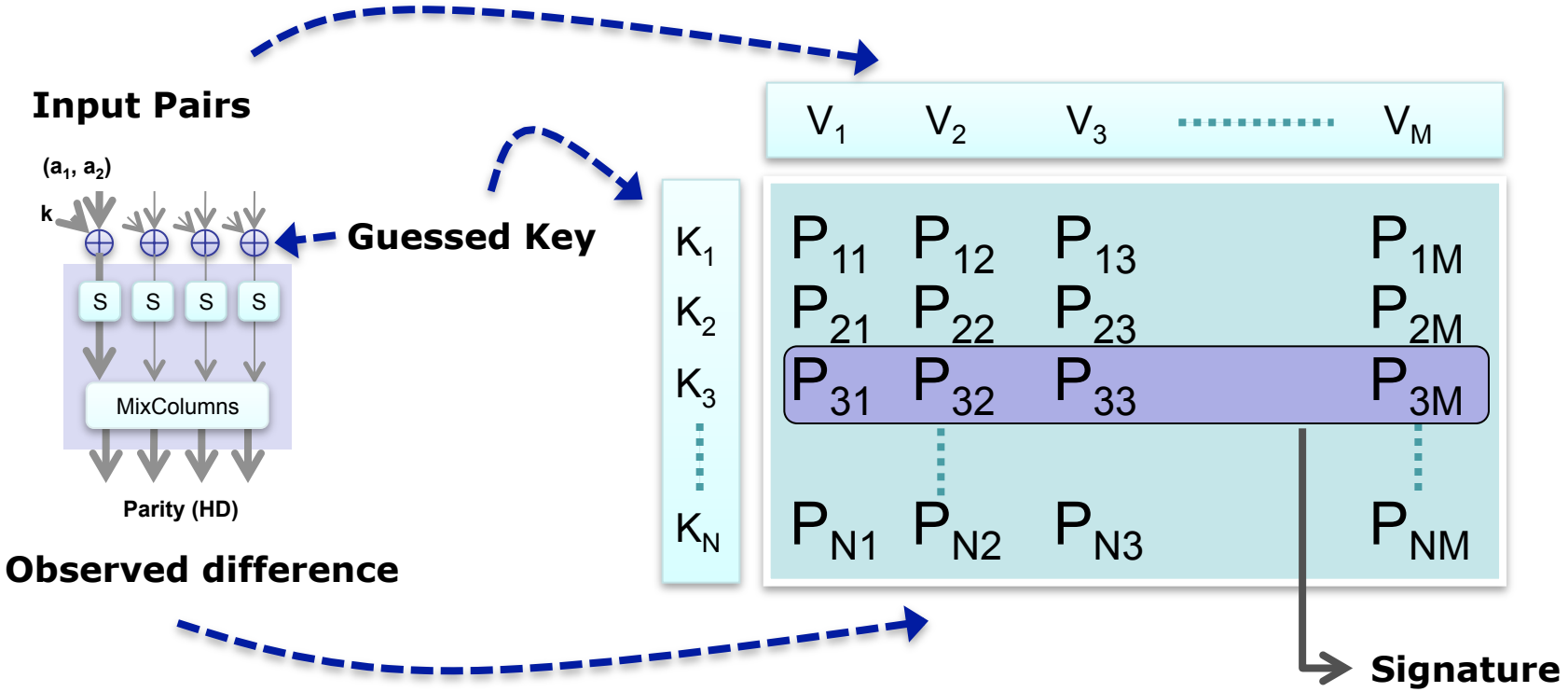
Signature Attack





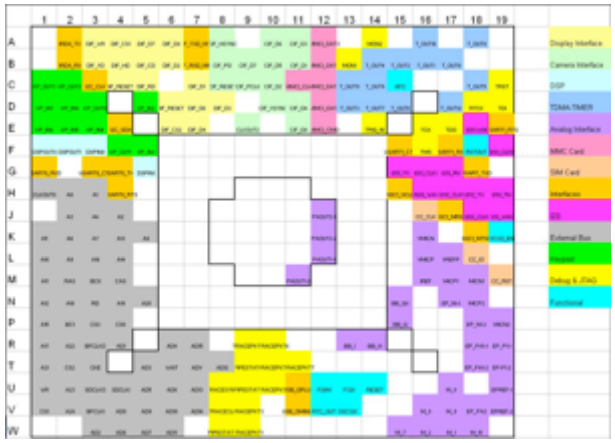
Signature Table

- Signature depends on the key guess
- Signature table consists of the signature for each key

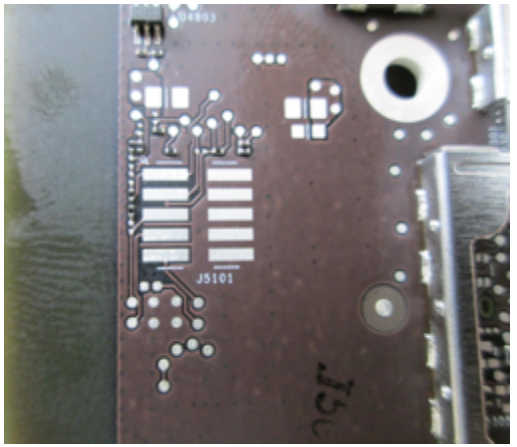


Goal: have unique signatures!

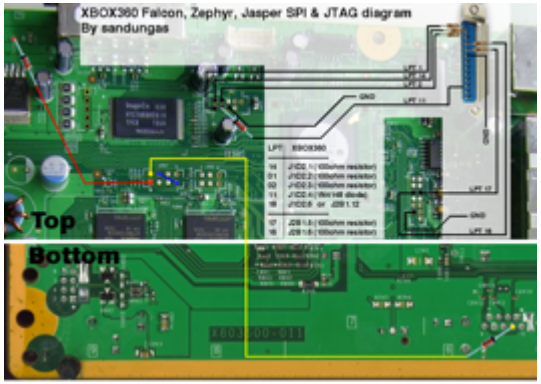
A Real Threat???



iPhone 4 Pinout



Apple TV – Access to JTAG



XBOX 360

Countermeasures

- Leave the scan chain unbound (fuses)
- Built-In Self-Test
- On-chip test comparison
- Secure Test Access Mechanism

Countermeasures

- Leave the scan chain unbound (fuses)
- **Built-In Self-Test**
- On-chip test comparison
- Secure Test Access Mechanism

Built-In Self-Test (BIST)

- Motivation:
  - Avoid scan-based testing
  - Allow at-speed testing
  - Reduced ATE cost
- But:
  - Area overhead ?
  - Fault coverage ?

Properties of Crypto-Algorithms

- Confusion
  - making the relationship between the key and the ciphertext as complex and involved as possible
- Diffusion
  - a change in a single bit of the plaintext should result in changing the value of many ciphertext bits

Testability

- Diffusion vs testability:
  - every input bit influences many output bits (i.e. every input bit is in the logic cone of many output bits)
    - the circuit is very observable
  - the input logic cone of every output contains many inputs
    - each fault is highly controllable

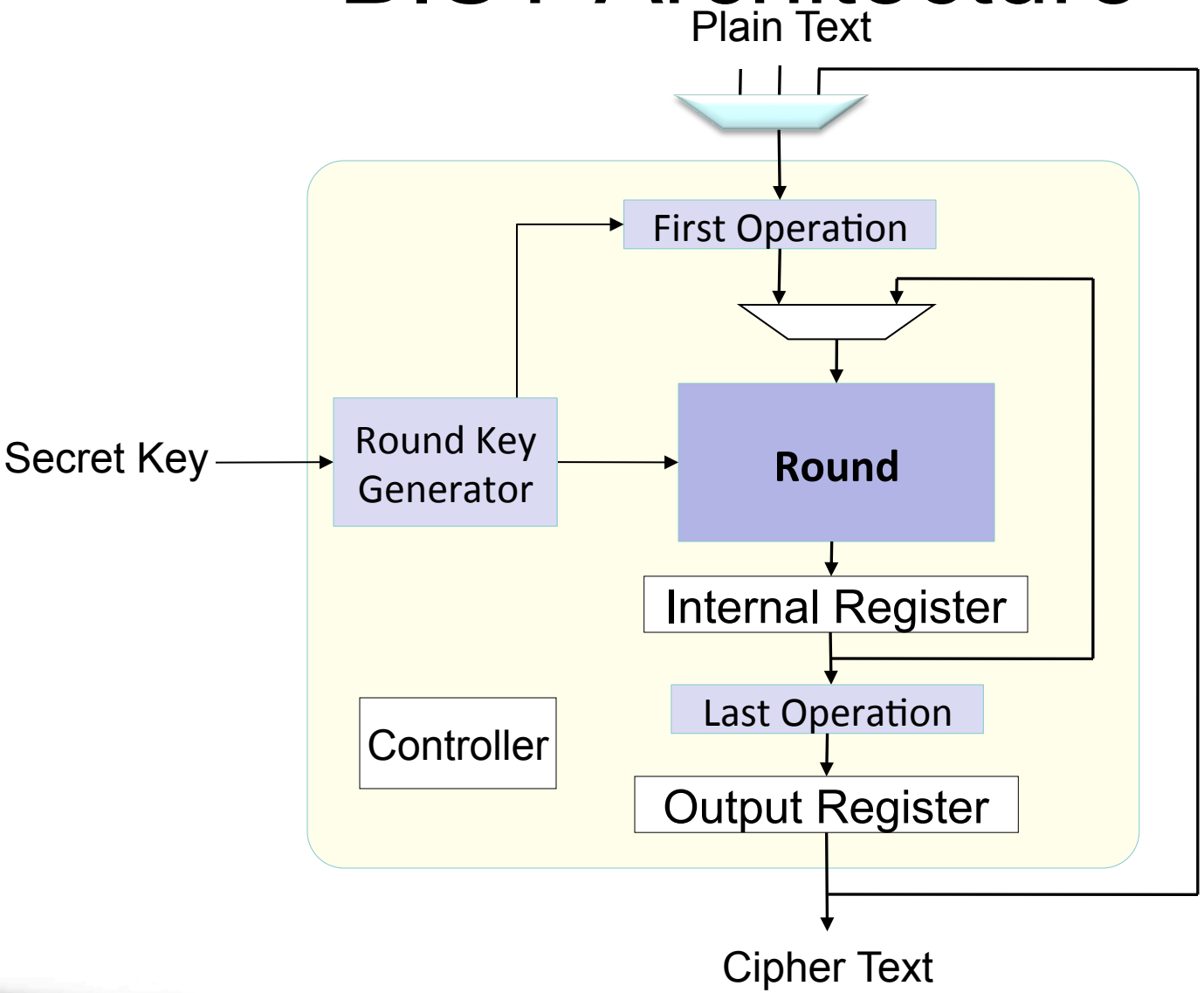
- **Therefore these circuits are highly testable by nature no matter the implementations**

Randomness

- Diffusion and confusion de-correlate the output values from the input ones (both at encryption and round levels)
- It has been demonstrated that by feeding back the output to the input, the generated output sequence has random properties

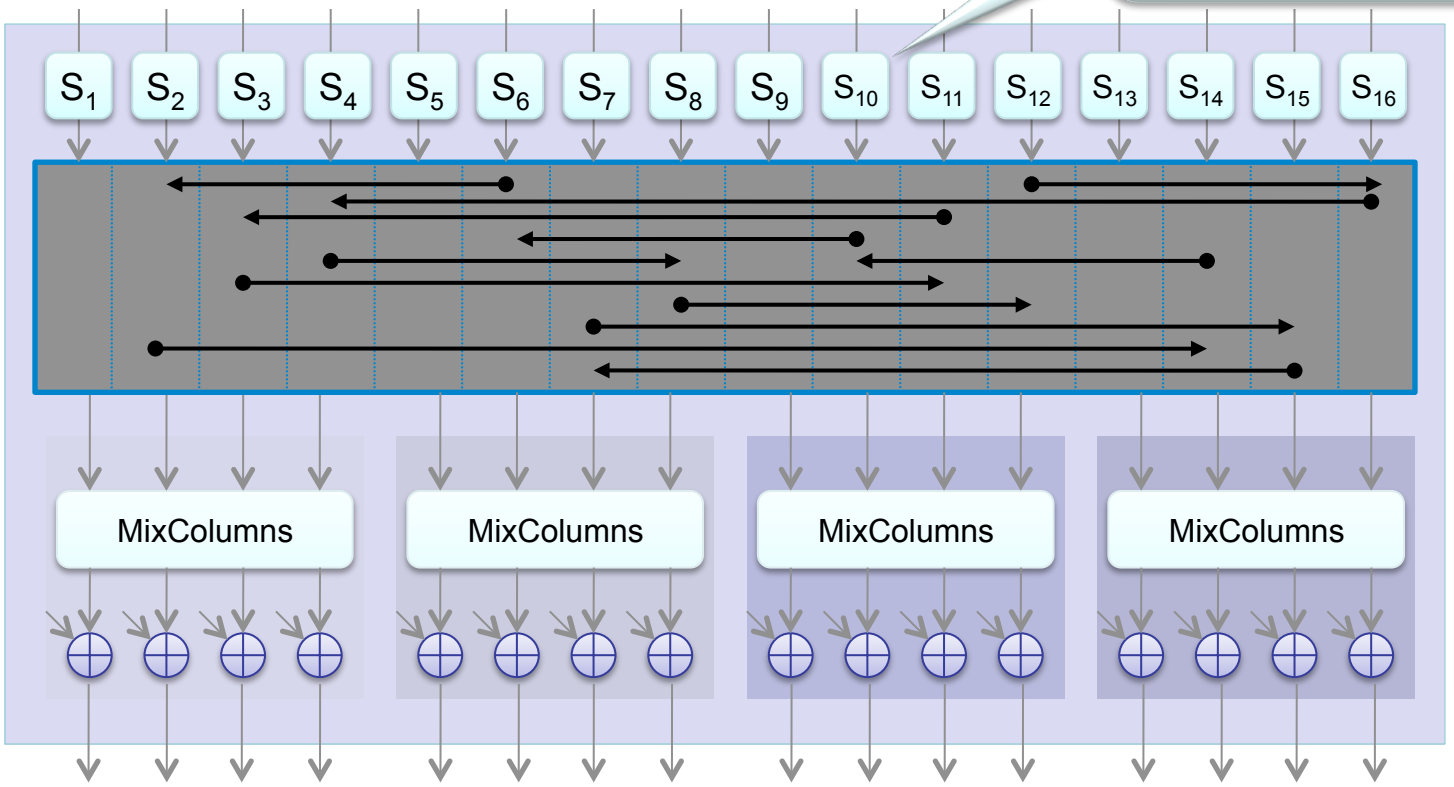
  [B. Schneier, Applied Cryptography, Second Ed., John Wiley and Sons, 1996]
- AES/DES better than LFSR (proved by NIST tests)

BIST Architecture



AES Round Testability

Sbox Implementations:
- ROM → 256 patterns
- Logic → 200 – 220 patterns



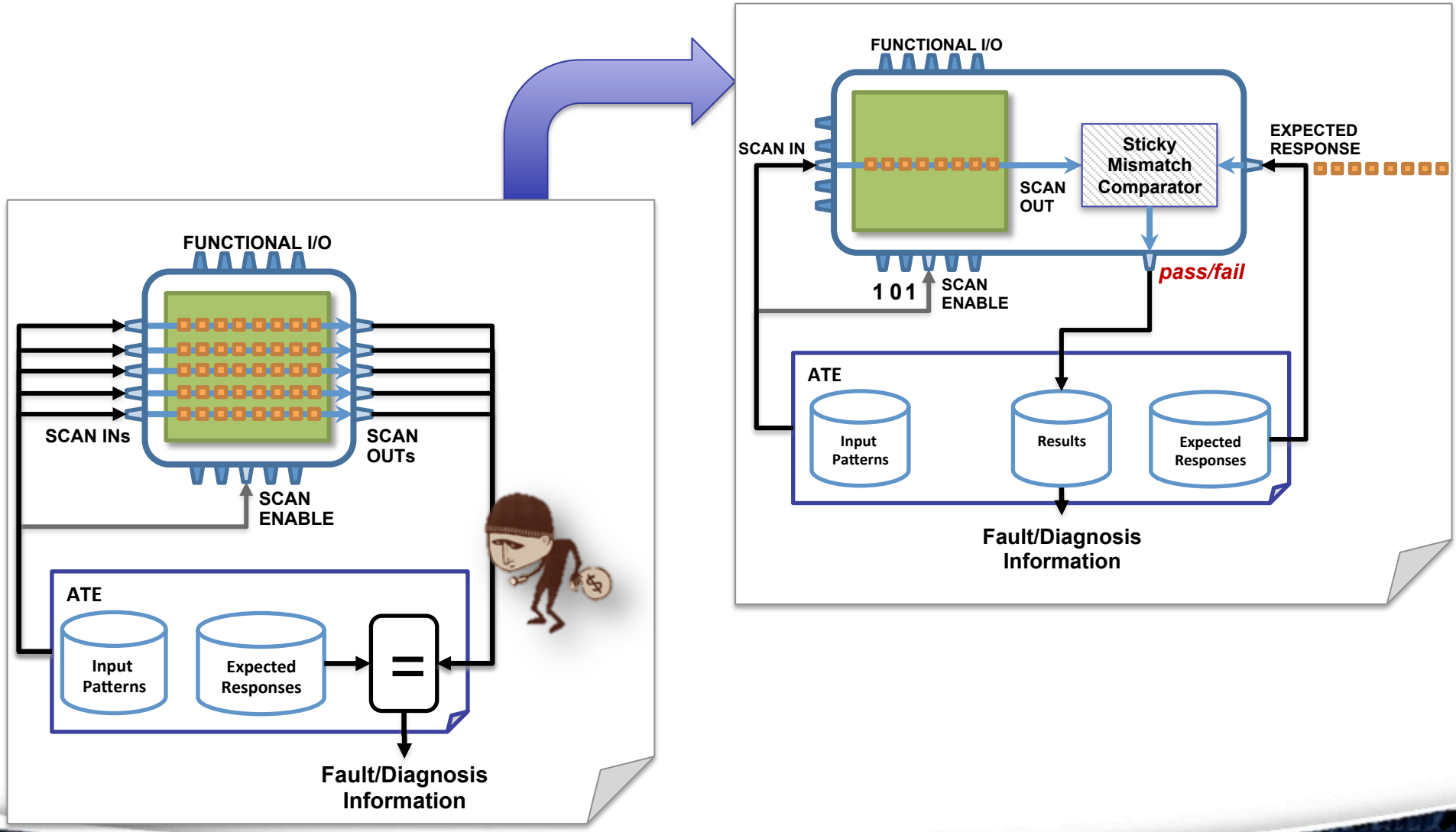
Faults in ShiftRows, MixColumns, Addkey, and Register are also tested by the 200 patterns

Fault coverage: 100% always before 2400 clock cycles (several keys, several plaintexts)

Countermeasures

- Leave the scan chain unbound (fuses)
- Built-In Self-Test
- **On-chip test comparison**
- Secure Test Access Mechanism

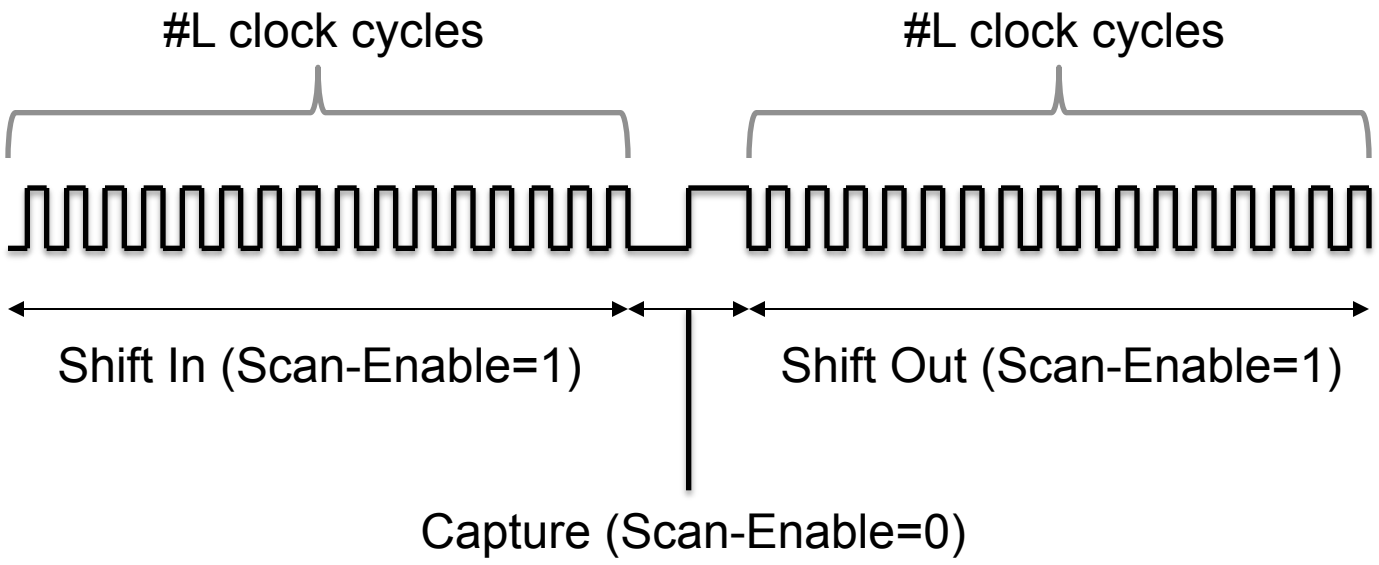
On-Chip Test Comparison



Countermeasures

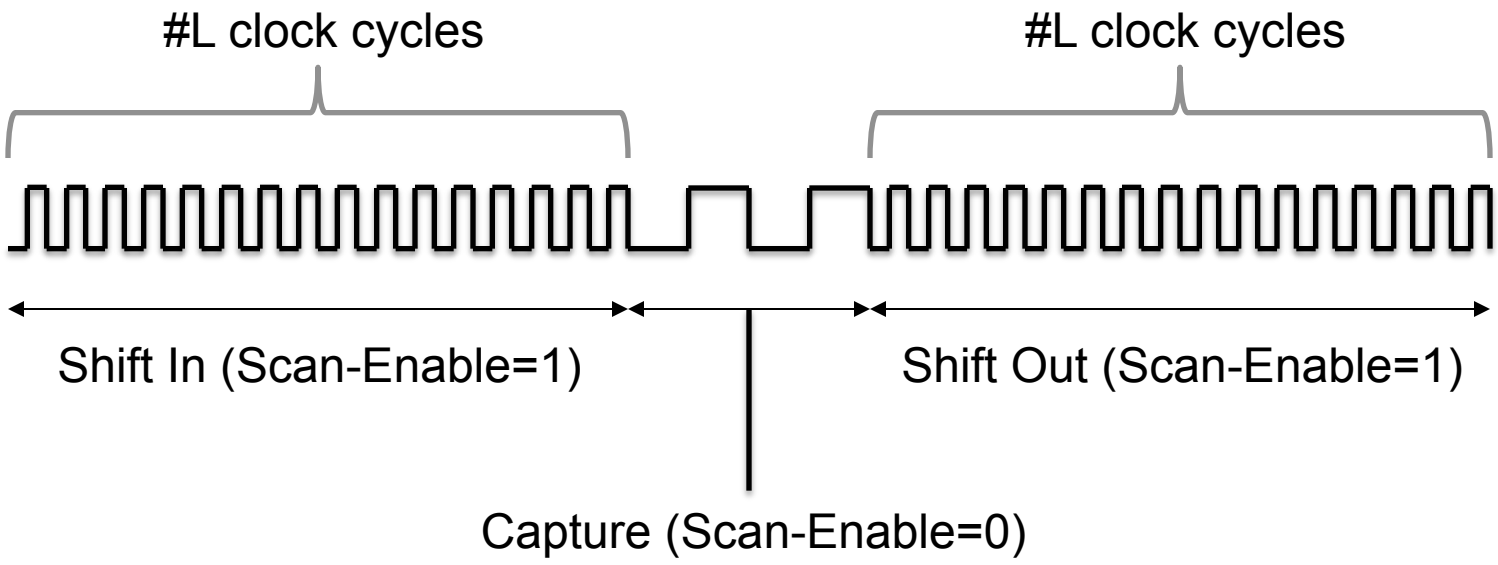
- Leave the scan chain unbound (fuses)
- Built-In Self-Test
- On-chip test comparison
- **Secure Test Access Mechanism**

Scan Chain Testing



Scan Chain Testing

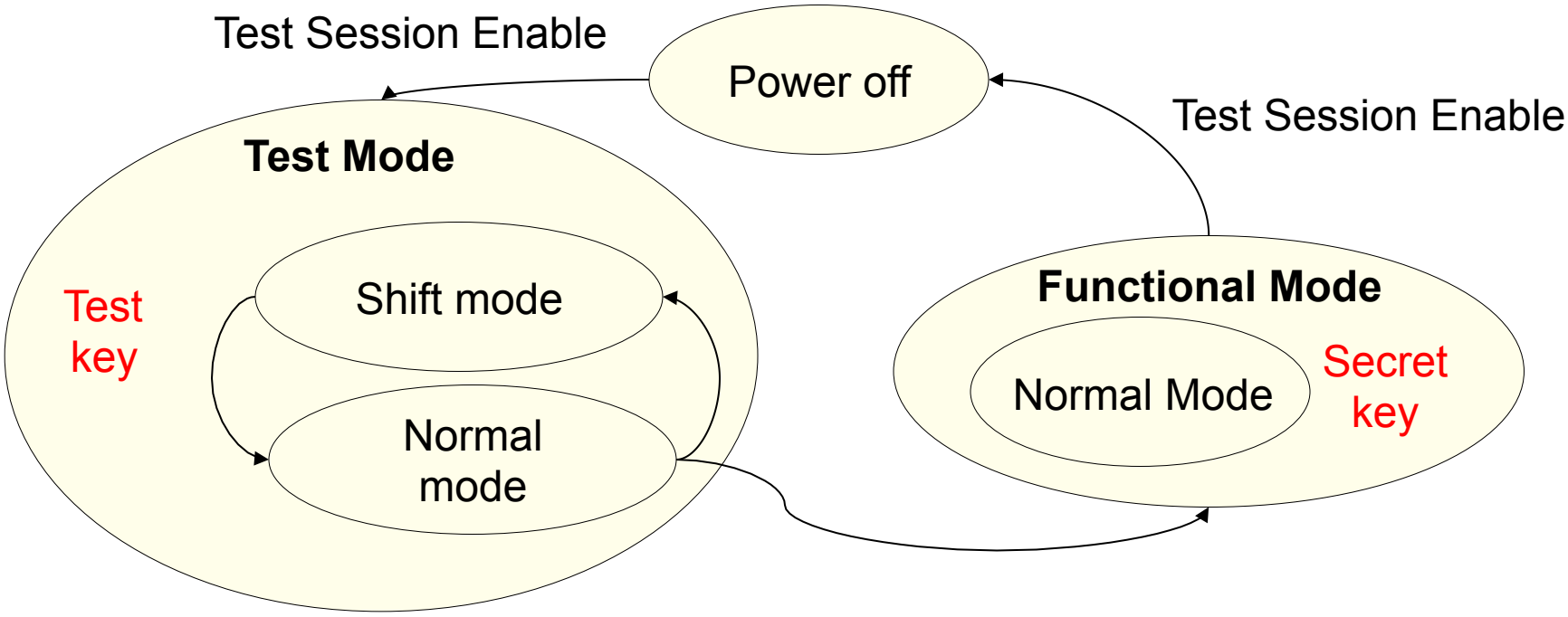
(delay fault testing with Launch On Capture)



Secure Test Access Mechanism

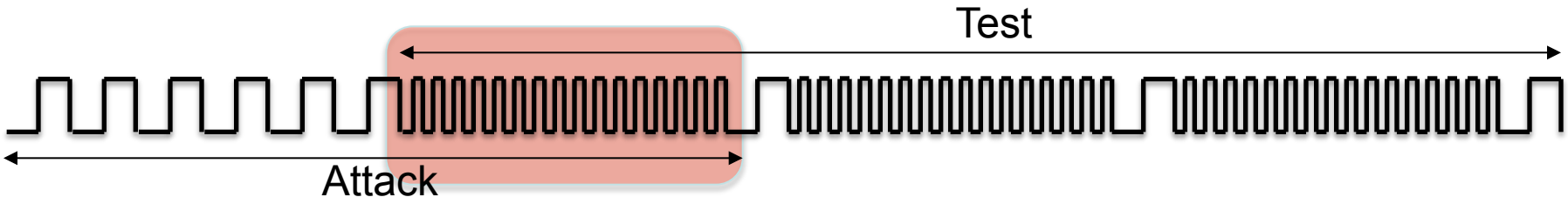
- How to start a test session
  - Additional signal
  - Power on
  - Authentication
- What to do in functional mode
  - e.g., scrambling, avoiding shift mode, spy Flip Flops
- What to do when switching to test mode
  - e.g., reset FFs
- What to do in test mode
  - e.g., test key instead of actual secret key

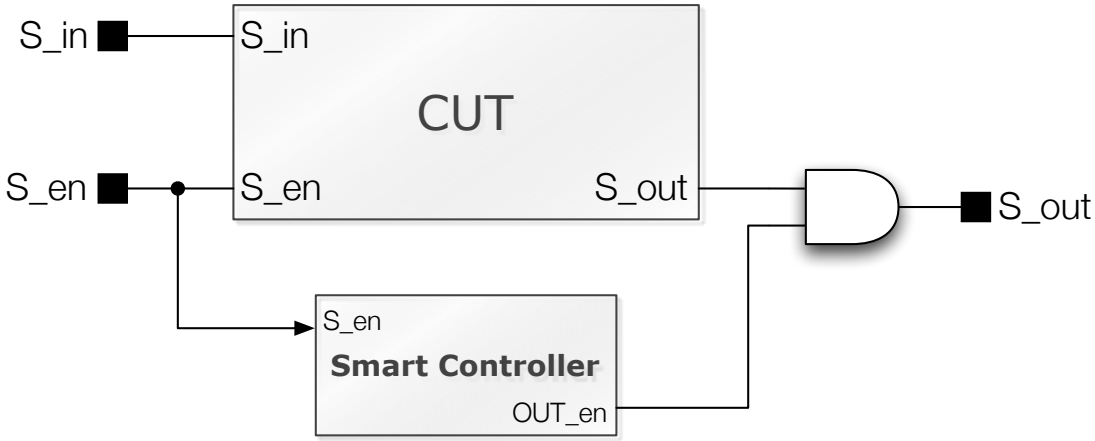
Control



[B. Yang et al., IEEE TRANS. ON CAD, oct. 2006]

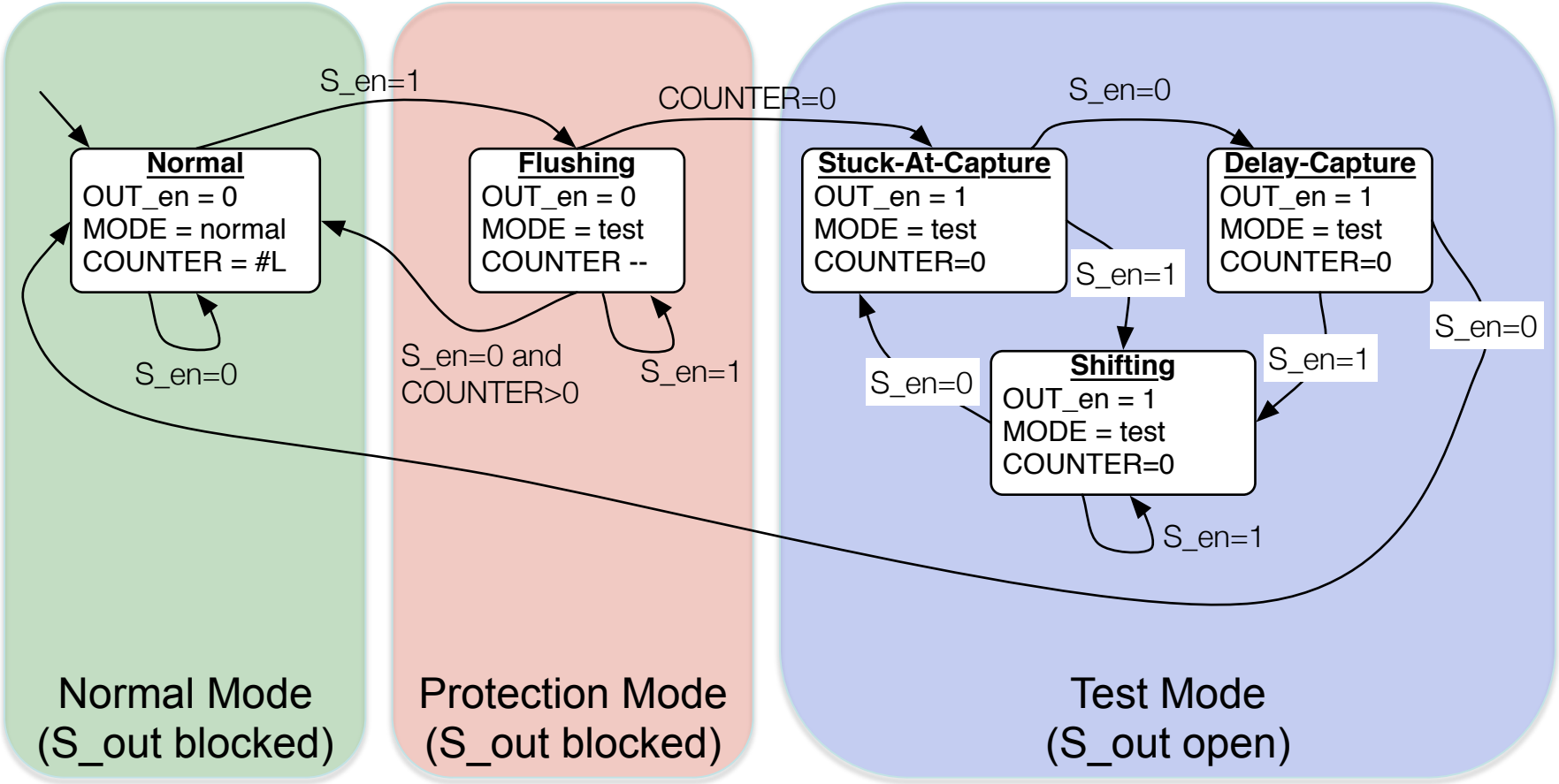
The smart test controller



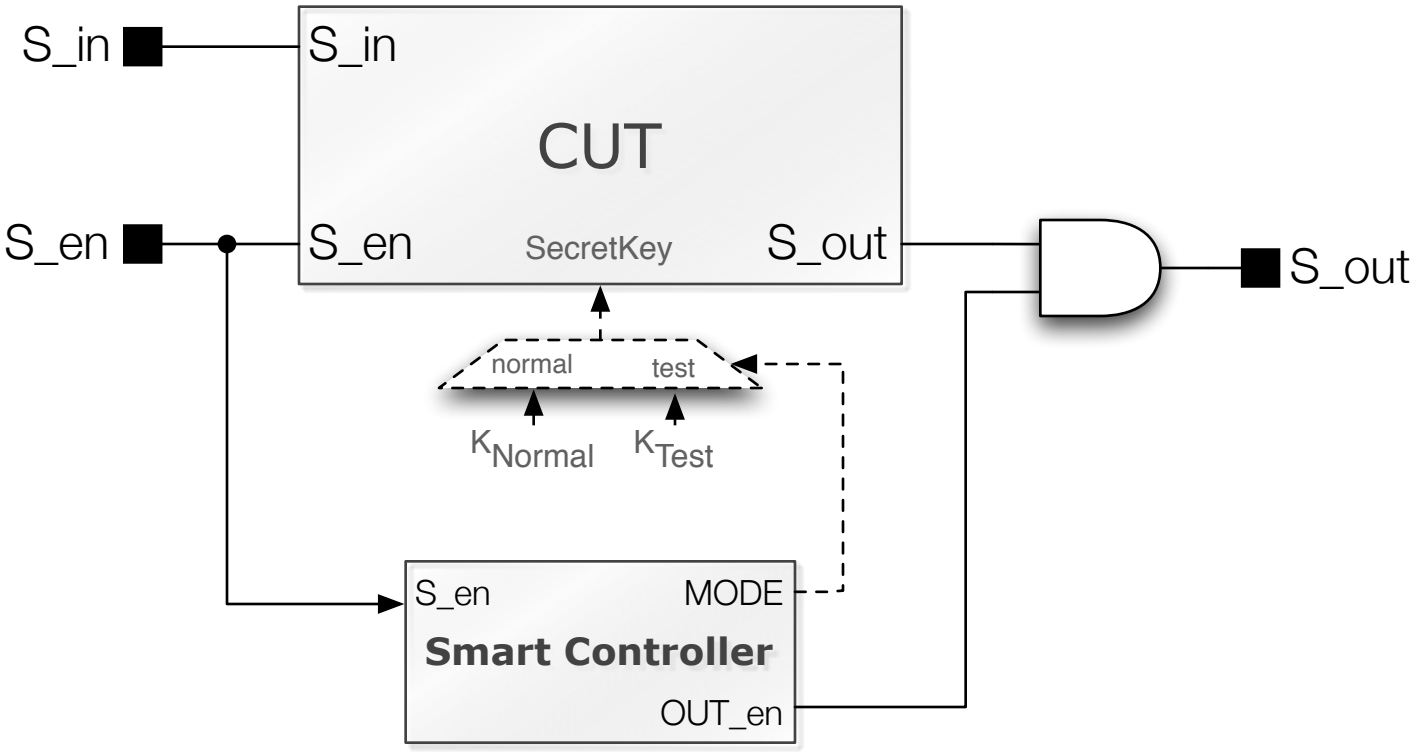


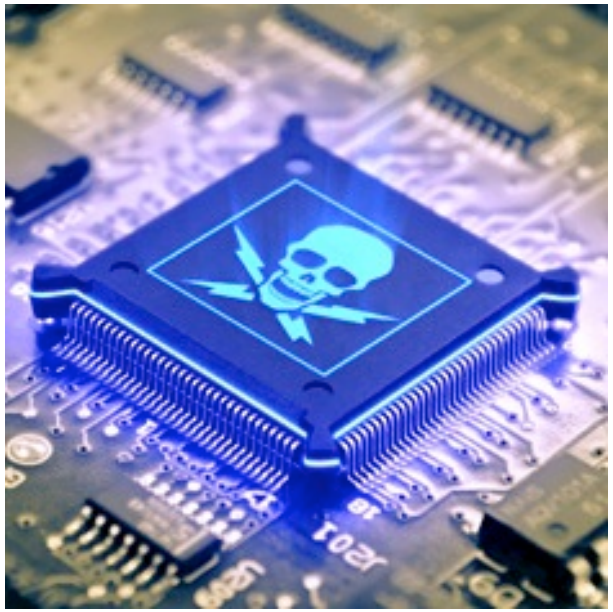
[J. Darolt et al., IEEE IOLTS, 2013]

Detection of the test mode



The smart test controller





HARDWARE TRUST

Counterfeiting

- Counterfeiting of integrated circuits has become a major challenge
  - deficiencies in the existing test solutions
  - lack of low-cost and effective avoidance mechanisms in place
- Numbers:
  - $75 billion lost in 2013 (in semiconductors)
  - E.g., 186 million fake mobile phones in 2013

Source: http://www.havocscope.com/tag/counterfeit-electronics/

Which is the cause?

- Complexity of the electronic systems significantly increased over the past few decades
  - To **reduce production cost**, they are mostly fabricated and assembled globally
- This **globalization** has led to an illicit market willing to undercut the competition with counterfeit and fake parts

Stories

- November 8, 2011, the United States Committee on Armed Services held a hearing on an investigation of counterfeit electronic parts in the **defense supply chain**
- The investigation had revealed alarming facts: materials used to make counterfeit electronic (a.k.a. e-waste) parts are shipped from the United States and other countries

http://www.industryweek.com/procurement/ticking-time-bomb-counterfeit-electronic-parts

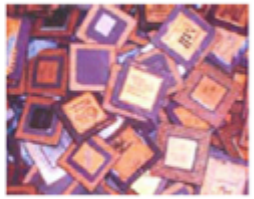
Stories (2)

- The e-waste is sent to cities like Shantou, China, where:
  - It is disassembled by hand, washed in dirty river water, and dried on the city sidewalk
  - It is sanded down to remove the existing part number or other markings that indicate its quality or performance
  - False markings are placed on the parts that lead the average person to believe they are new or high-quality parts

Stories (3)

Millions of Scrap Boards

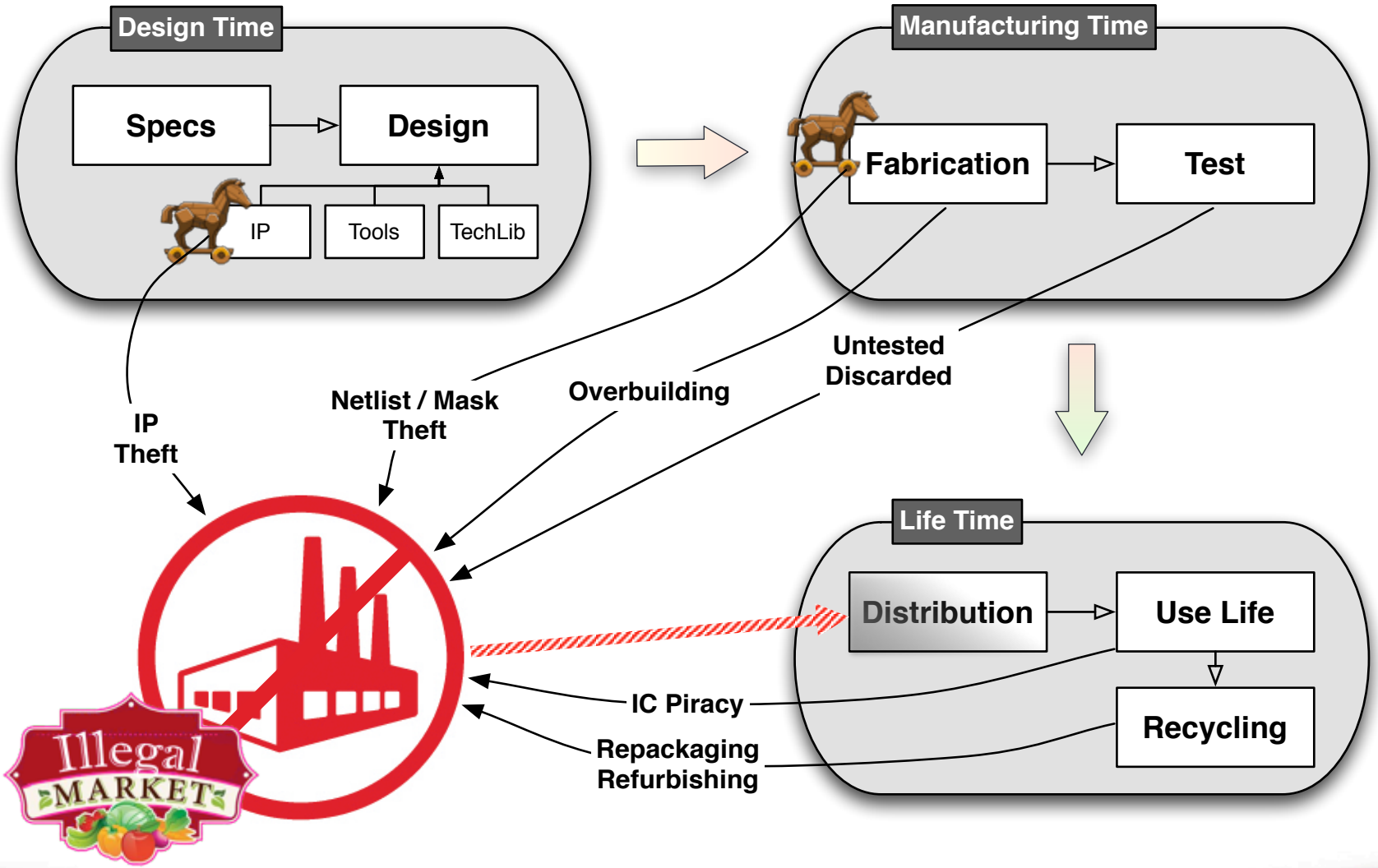
Component Removal



Sorted by size, similarity and lead count

Re-processed

The Untrusted Chain



Counterfeit types: Recycled

- Electronic component that is recovered from a system and then modified to be misrepresented as a new component.
- Problems:
  - lower performance
  - shorter lifetime
  - damaged component, due to the reclaiming process (removal under very high temperature, aggressive physical removal from boards, washing, sanding, repackaging, etc.)

Aging Detectors

- Sensor in the chip to capture the usage of the chip in the field
  - It relies on aging effects of MOSFETs to change a ring oscillator frequency in comparison with the golden one embedded in the chip.
- Antifuse-based Technology for Recording Usage Time

Counterfeit types: Remarked

- Chemically or physically removing the original marking
- Goal:
  - to drive up a component's price on the open market
  - to make a dissimilar lot fraudulently appear homogeneous

Counterfeit types: Overproduced

- Overproduction occurs when foundries sell components outside of contract with the design house
- Problems:
  - loss in profits for the design and IP owner
  - reliability threats since they are often not subjected to the same rigorous testing as authentic parts

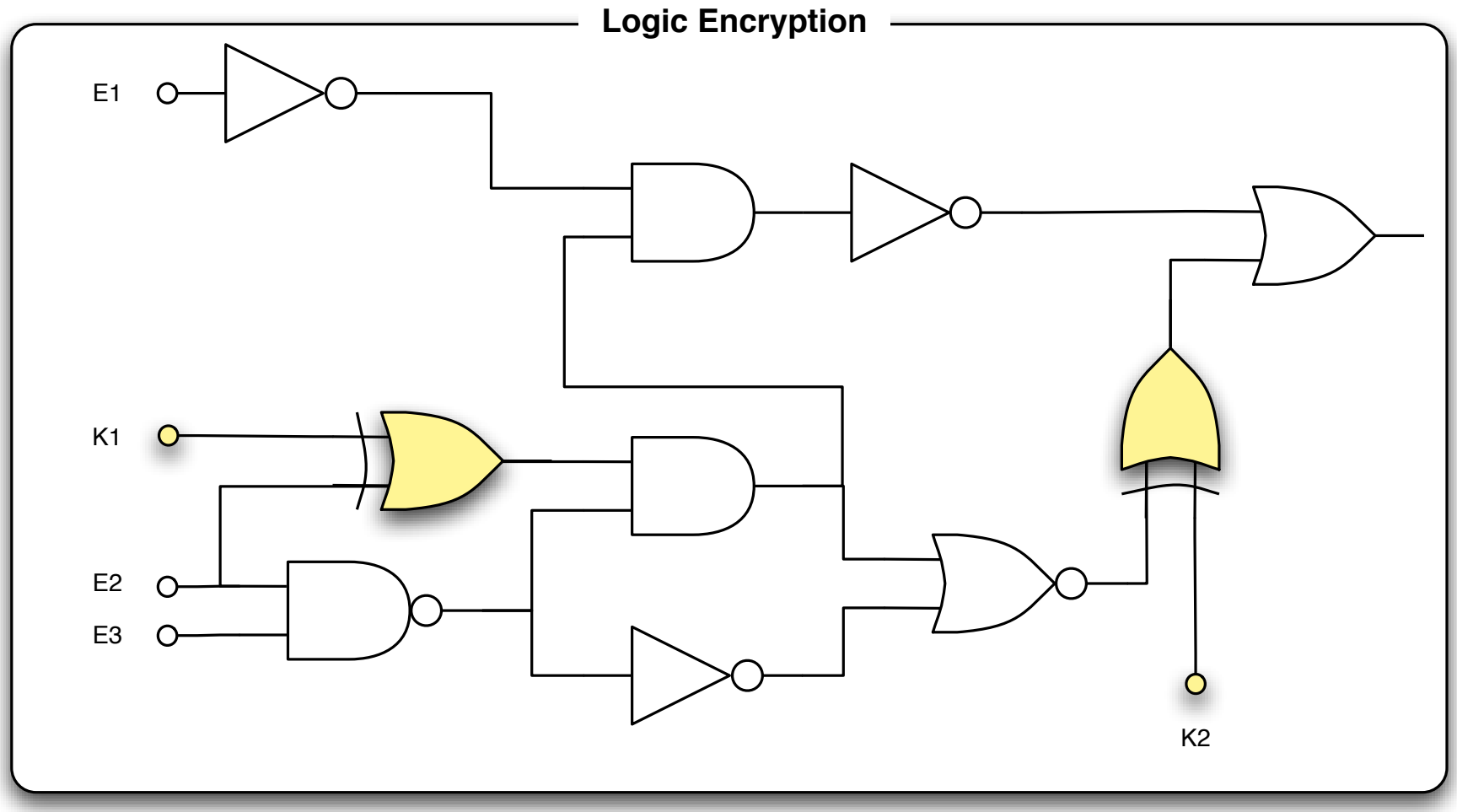
Hardware Metering

- A set of security protocols that enable the design house to achieve the post-fabrication control of the produced ICs to prevent overproduction
  - Physically Unclonable Functions
  - Post-Manufacturing Activation
  - Adding a Finite-State Machine (FSM) which is initially locked and can be unlocked only with the correct sequence of primary inputs
  - Logic Encryption

Physically Unclonable Functions

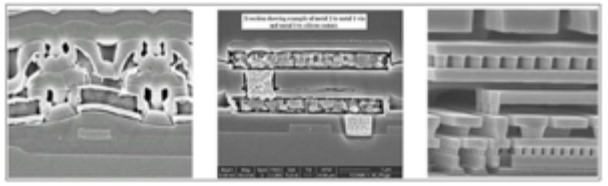
- Silicon PUFs exploit inherent physical variations (process variations) that exist in modern integrated circuits
  - variations are uncontrollable and unpredictable, making PUFs suitable for IC identification and authentication
- Based on:
  - Content of SRAMs at power-up
  - Delay of some networks

Logic Encryption



Counterfeit types: Defective

- A part is considered defective if it produces an incorrect response to post-manufacturing tests
- These parts should be destroyed, downgraded, or otherwise properly disposed of
- However, if they can be sold on the open markets, either knowingly by an untrusted entity or by a third party who has stolen them



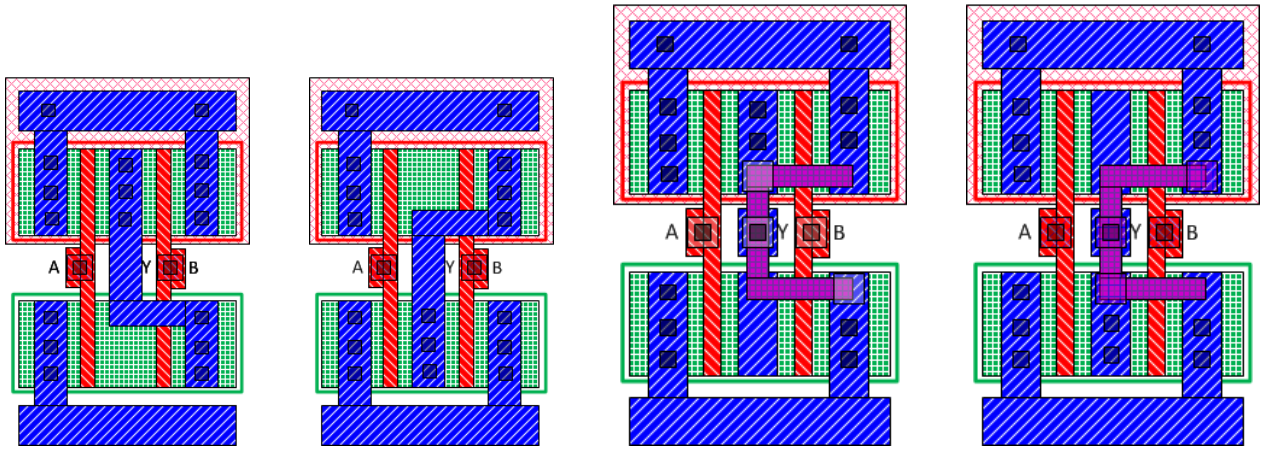
Source: http://www.bradreese.com/blog/2-16-2014.htm

Counterfeit types: Cloned

- A copy of a design, in order to eliminate the large development cost of a part
- Methods:
  - Reverse engineering,
  - By obtaining IP illegally (also called IP theft)
  - With unauthorized knowledge transfer from a person with access to the part design

IC Camouflage

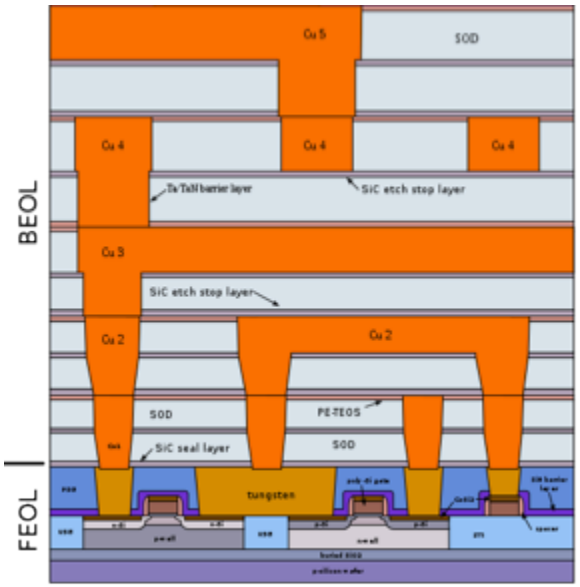
- Standard-cells are re-designed not to disclose their identity



NAND NOR Camouflaged NAND Camouflaged NOR

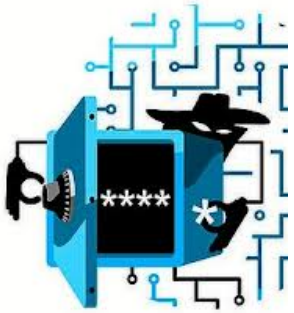
Split Manufacturing

- Front End Of Line (FEOL) layers (transistor and lower metal layers) are fabricated in an untrusted foundry
- Back End Of Line (BEOL) in a trusted low-end fab
- It is considered secure against reverse engineering as it hides the BEOL connections from an attacker in the FEOL foundry



Counterfeit types: Tampered

- Components modified in order to cause damage or make unauthorized alterations
- Examples:
  - time bombs that stop the circuit functionality at a critical moment
  - Backdoors that give access to critical system functionality or leak secret information

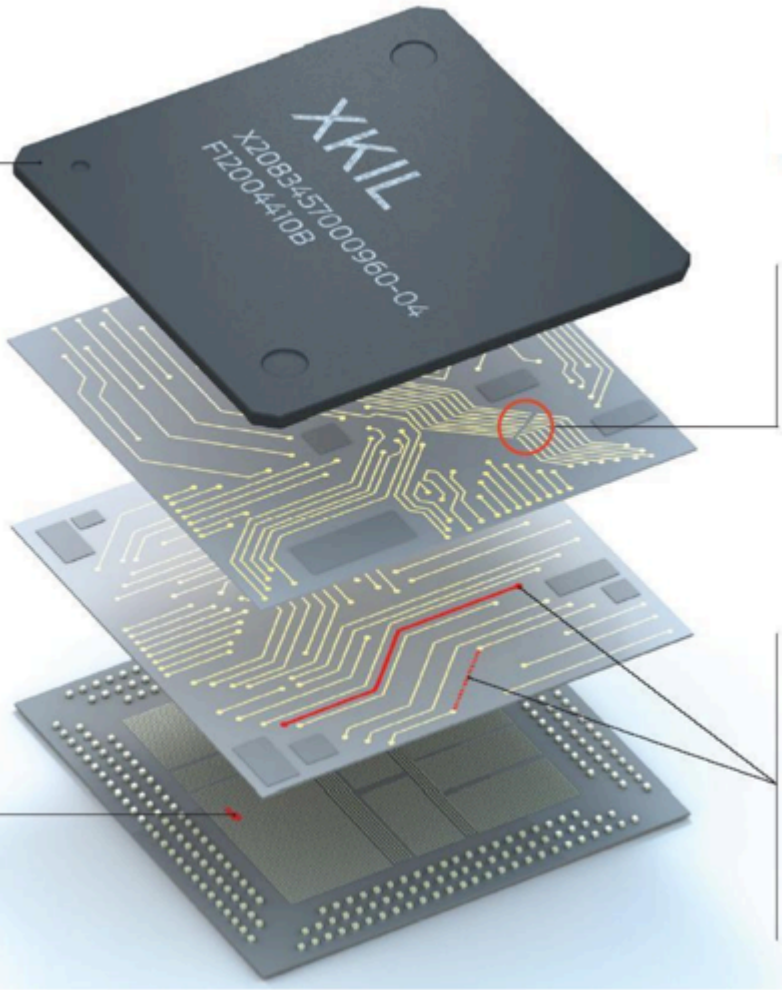


Hardware Trojan Horses

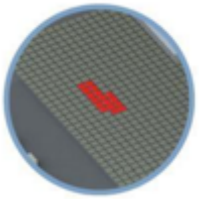
- A Hardware Trojan Horse is a malicious modification of an integrated circuit
  - Performed at any design and/or manufacturing step
- A real threat?
  - Few real cases
  - A big fear!!

Trojan at Manufacturing Time

BAKE
Baking a chip for 24 hours after fabrication could shorten its life span from 15 years to a scant 6 months



NICK THE WIRE
A notch in a few interconnects would be almost impossible to detect but would cause eventual mechanical failure as the wire became overloaded.



ADD EXTRA TRANSISTORS
Adding just 1000 extra transistors during either the design or the fabrication process could create a kill switch or a trapdoor. Extra transistors could enable access for a hidden code that shuts off all or part of the chip.

ADD OR RECONNECT WIRING
During the layout process, new circuit traces and wiring can be added to the circuit. A skilled engineer familiar with the chip's blueprints could reconnect the wires that connect transistors, adding gates and hooking them up using a process called circuit editing.

Source: IEEE Spectrum

Hardware Trojan Horses: Solutions

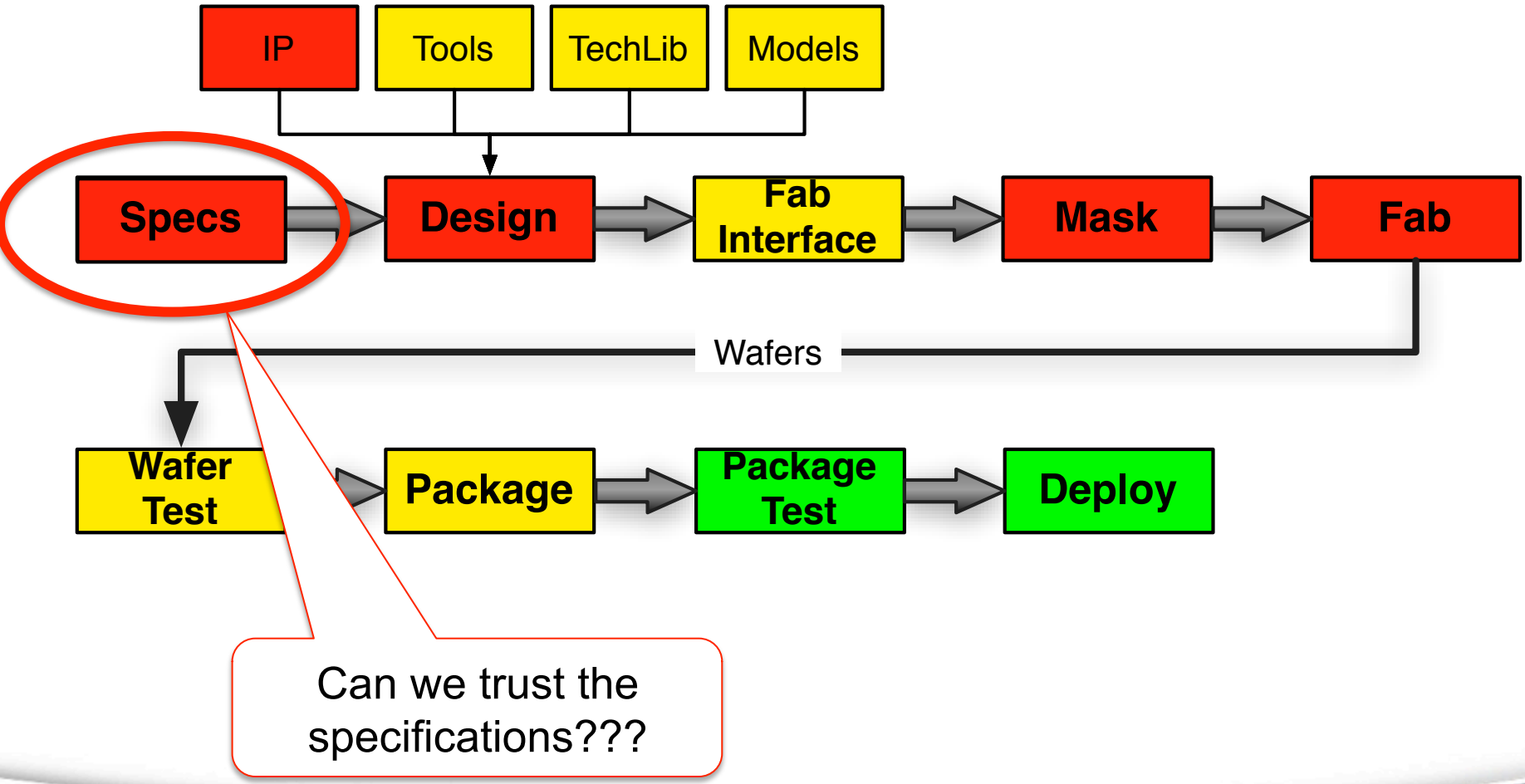
- Detection:
  - Test methods
  - Side-Channel (power, EM, timing)
  - Destructive methods (reverse engineering)
- Prevention:
  - Split Manufacturing
  - Logic Encryption

Summary

	Overproduction	Recycling	Cloning	Trojans
HW Metering	Y		Y	y
IC Camouflage			Y	y
Aging Detectors		Y		
PUFs	y		Y	
Split Manufacturing	Y		Y	Y
Test methods		y		Y
Side-Channel				Y
Reverse Engineering			As mean	Y

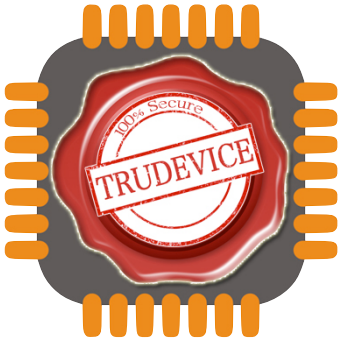
Ujjwal Guin · Daniel DiMase · Mohammad Tehranipoor,
Counterfeit Integrated Circuits: Detection, Avoidance, and the Challenges Ahead,
J Electron Test (2014) 30:9–23 DOI 10.1007/s10836-013-5430-8

Evolution of IC Supply Chain



Advertisement

- TRUDEVICE
  - COST Action IC1204
  - Summer School: Lisbon, 14th-18th July
  - http://www.trudevice.com



Thank you!!