国立大学法人
電気通信大学
The University of Electro-Communications

崎山·李·宮原 研究室

All You Need Is Fault: Zero-Value Attacks on AES and a New λ-Detection M&M

Haruka Hirata[1)], Daiki Miyahara[1)], Victor Arribas[3,4)]

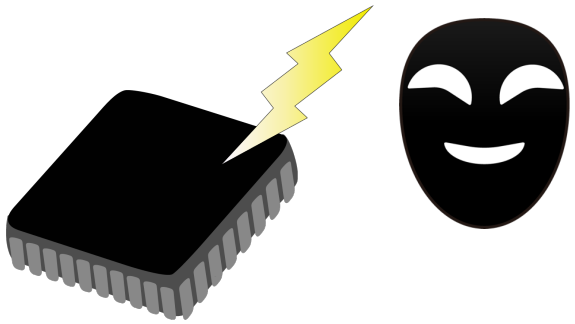
Yang Li[1)], Noriyuki Miura[2)], Svetla Nikova[3)], Kazuo Sakiyama[1)]

[1)]The University of Electro-Communications, [2)]Osaka University, [3)]KU Leuven, [4)]Rambus Inc.

This work was supported by JSPS KAKENHI
(grant numbers JP18H05289, JP20K19798, JP23H03364, and JP23H03393) and
by CyberSecurity Research Flanders with reference number VR2019220

Why combined countermeasure?

- ✓Side-channel analysis and fault analysis pose a significant threat against cryptographic hardware
- ✓Countermeasures considering **both SCA and FA** have been studied
  e.g., CAPA [RMB+18], Combined Threshold Implementation [FRS+24] and Masks and Macs [MAN+19]
- ✓Masks and Macs (M&M) was proposed at TCHES 2019 (by KU Leuven)
  - ➢ They implemented 2nd order security AES with M&M as a case study



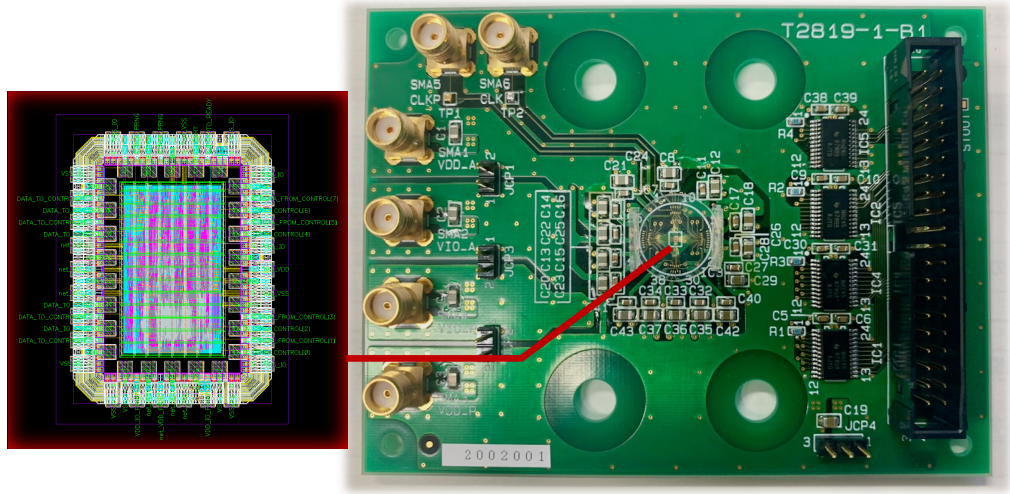
Motivation

✓We discuss the security of M&M theoretically and experimentally

1. KU Leuven proposed M&M AES
2. Osaka Univ. developed ASIC evaluation boards
3. UEC evaluated M&M AES by experiments

Table 1: Information of ASIC design and fabrication.

Foundry	TSMC
Technology	CMOS Process
Library	TSMC Standard Cell Library
Design Tool	Synopsys IC Compiler



Implemented ASIC (28nm CMOS process) board

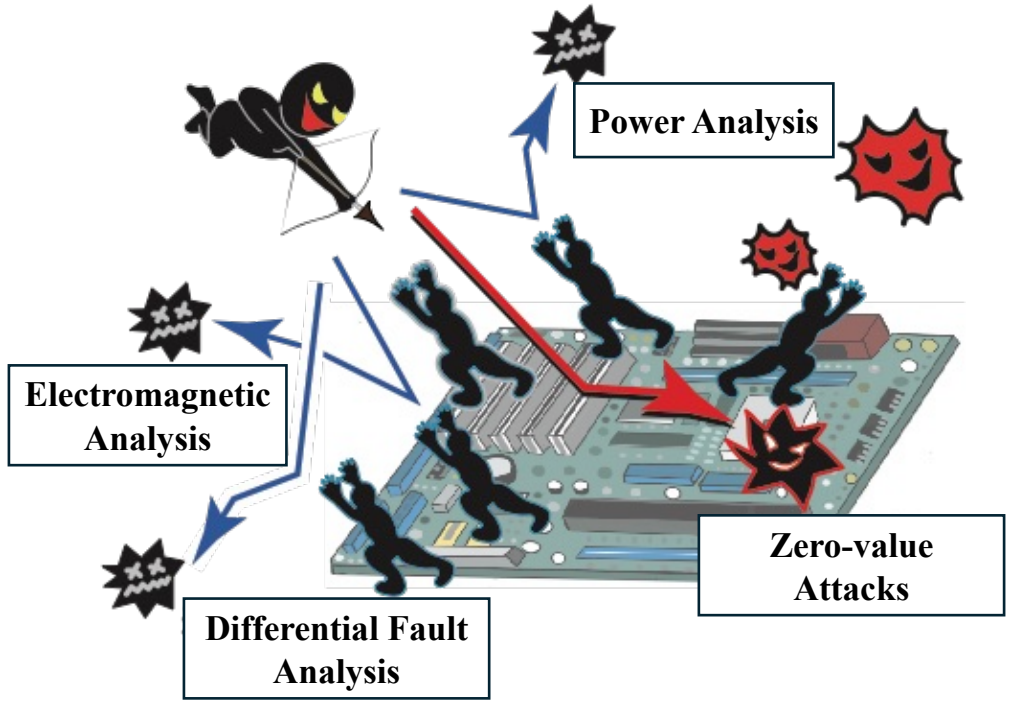
Our Contributions

Evaluation and Attack

- ✓Develop ASIC boards and evaluate M&M-AES with different experiments
- ✓Point out a vulnerability in M&M-AES Sbox that follows Canright's design [Can05]
- ✓Demonstrate SIFA-2 like attacks, named zero-value attacks

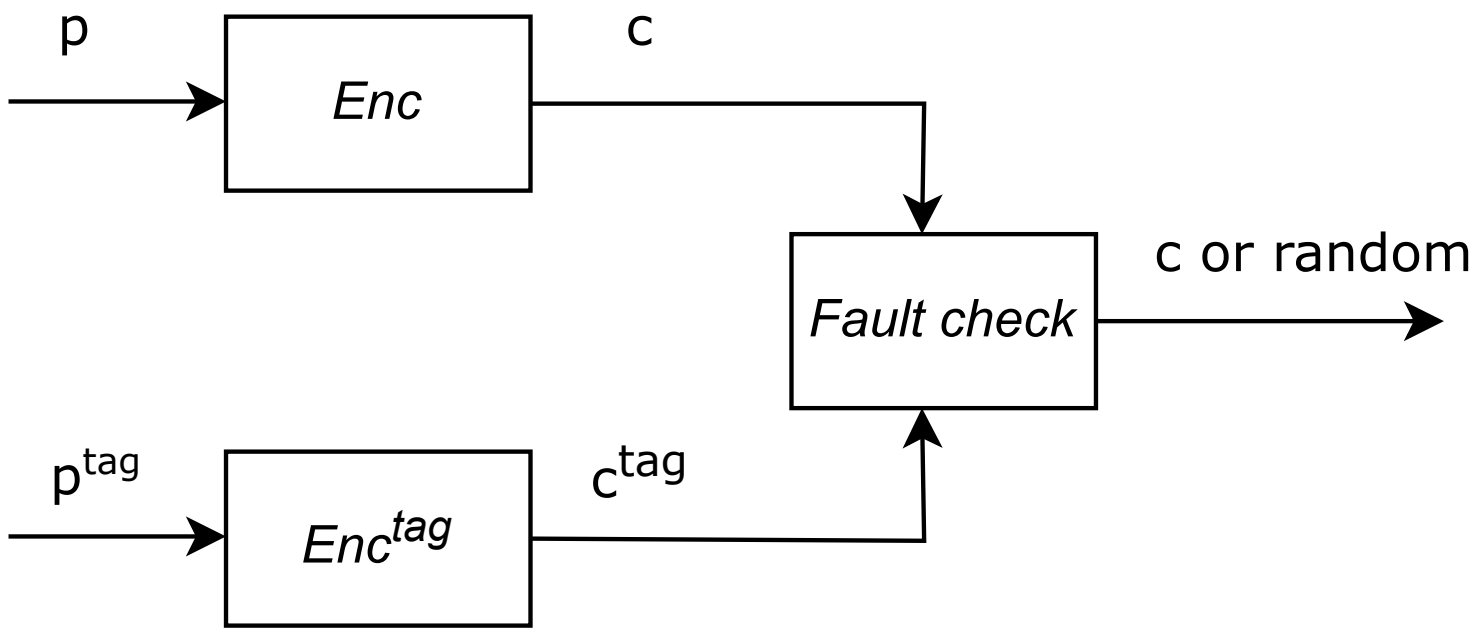
Countermeasure

- ✓Propose a new fine-grained and secure fault detection scheme
- ✓Conduct security evaluation for both fault and side-channel analysis



Overview of Masks and Macs (M&M)

- ✓M&M is a combined countermeasure against **side-channel** (by masking) and **fault** (by mac tags) analysis
  → withstands both SCA and FA
- ✓Mac tag τ^x is obtained by $\tau^x = x * \alpha$, where α is a tag key



M&M overview

Computing an AES S-box

✓AES S-box (SubBytes layer) can be realized

1. Draw a look-up table (often used in software implementation)

   Sbox = [63 7c 77 .. 54 bb 16];

   S_out = Sbox[00];

2. **Compute inversion and affine transformation**

   S_out = $\psi(x^{-1})B + c$,

   $\psi: GF(2^8) \rightarrow GF(2)^8$

   $B \in GF(2)^{8\times 8}, c \in GF(2)^8$

✓Option 2 is common in masked AES hardware implementations, but the inversion over $GF(2^8)$ has a heavy cost to compute..

Effective inversion over $GF(2^8)$

✓Inversion costs can be reduced from 8 bit to 2 bit [Can05]

Use an isomorphic mapping $\phi: GF(2^8) \rightarrow GF((2^4)^2)$,
then compute the inversion over $GF((2^4)^2)$

$\phi(x) = (a, b)$, $(c, d) := (a, b)^{-1}$, ν is a constant value in $GF(2^4)$

$$c = [ab + (a + b)^2\nu]^{-1}\, b$$

$$d = [ab + (a + b)^2\nu]^{-1}\, a$$

$\phi^{-1}((c, d)) = x^{-1} \in GF(2^8)$

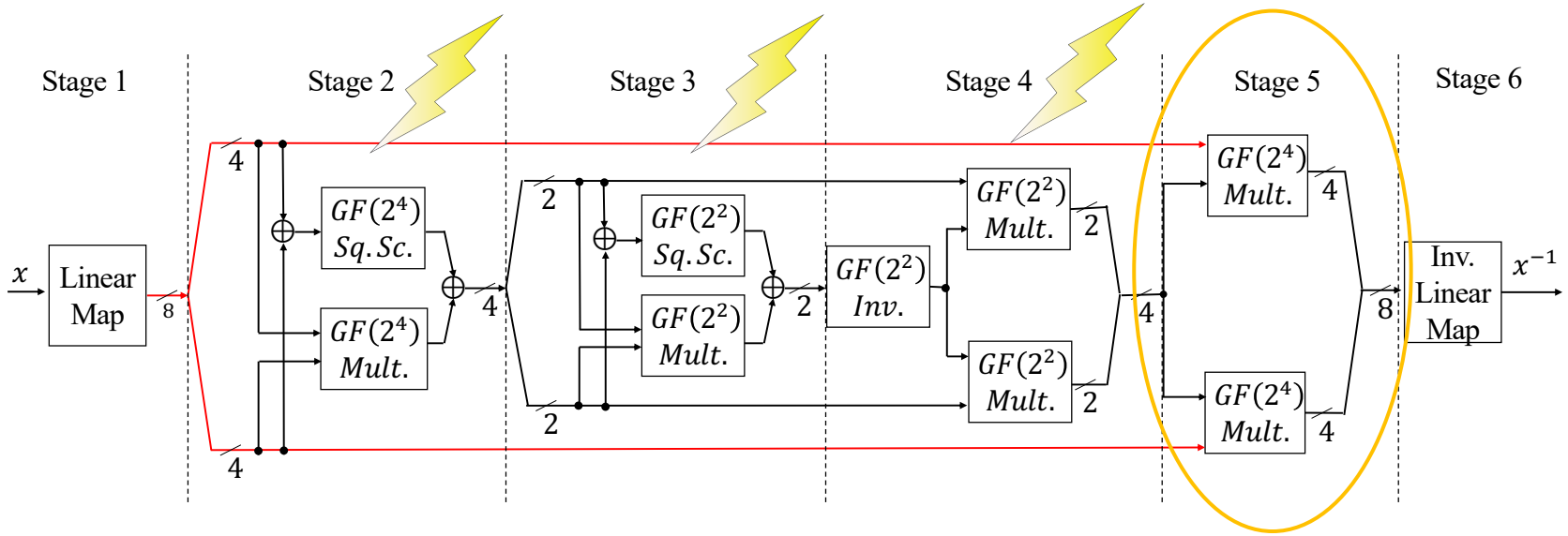
✓**if and only if $x = 0$, $(a, b) = \phi(x) = (0, 0)$ and $x^{-1} = 0$**

→ The computation of $[ab + (a + b)^2\nu]^{-1}$ is ineffective

GF(2^8) Inversion circuit and flaw

- ✓M&M AES S-box circuit has a pipelined structure with 6 stages [CRB+16]
  - ➢ Consolidating masking scheme [RBN+15] is used as a Boolean masking
- ✓Side paths (red colored) are the “critical” path against zero-value attacks

Faults on Stages 2-4
will be nullified by multiplying zero



Stage 1:

Map to $GF((2^4)^2)$ from $GF(2^8)$

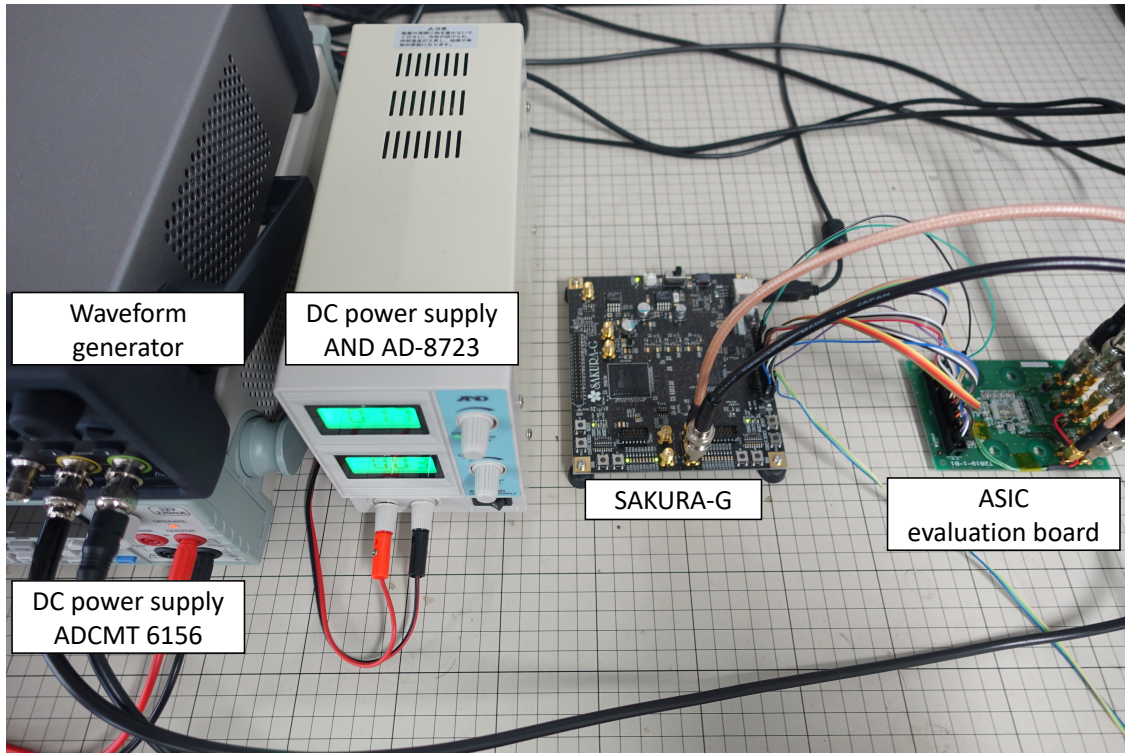
Stages 2-5:

Inversion over $GF((2^4)^2)$

Stage 6:

Map to $GF(2^8)$ from $GF((2^4)^2)$

Experiment to verify the vulnerability



Our lab's setup

Setup

✓We use a clock glitch to introduce faults

- ➢ Targeting the last round of AES

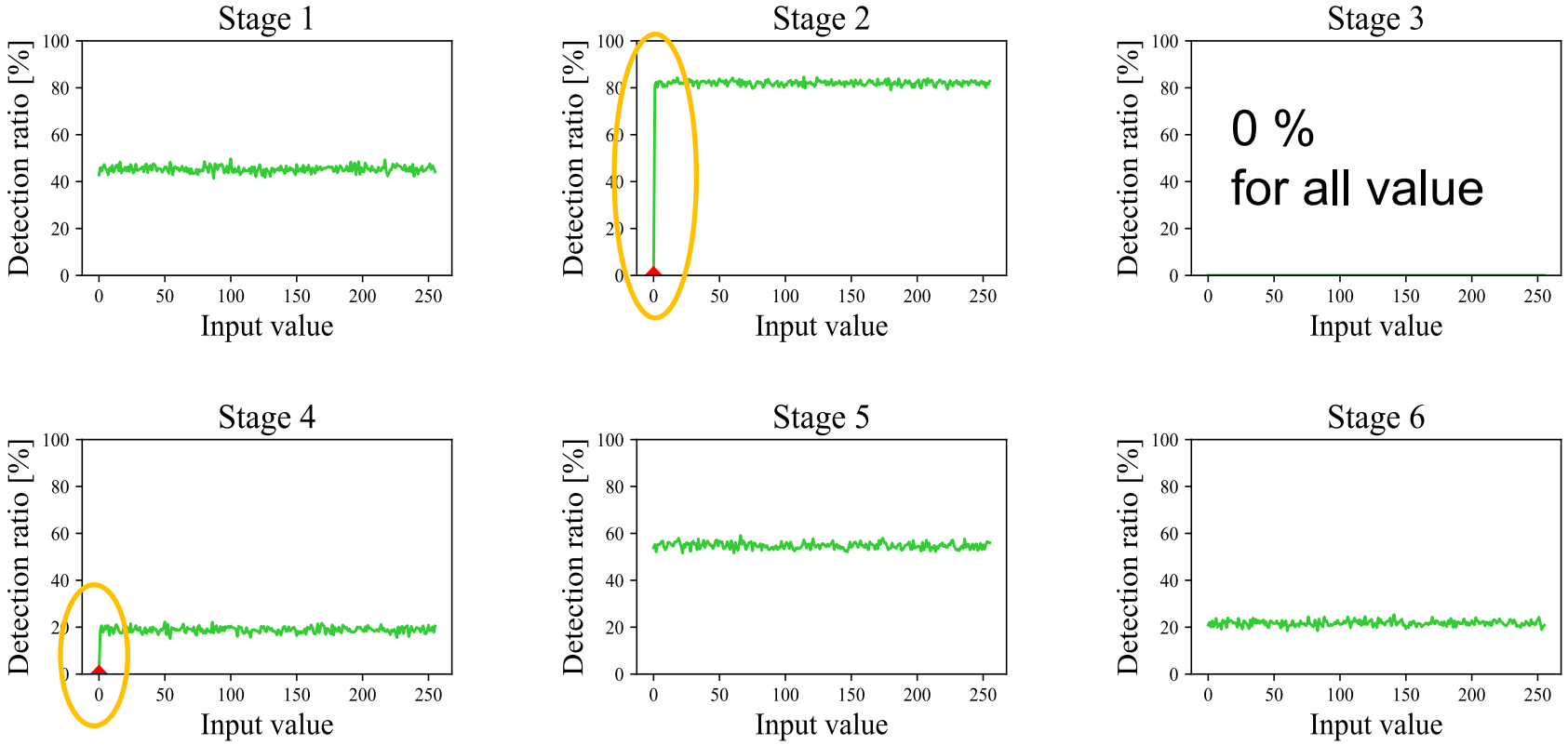
✓Calculate detection ratio at each stage with 30,000 random plaintexts *10 repetition

✓ Detection ratio is defined as

$$\frac{The\ number\ of\ fault\ occurences}{The\ number\ of\ operations}$$

Detection ratio of each input value of S-box

- ✓We focus on the ratios for input values
- ✓The detection ratio of zero is clearly 0% at Stages 2 and 4 as expected
- ✓This result indicates **SIFA2-like attacks are feasible**



Zero-value attacks procedure (last round)

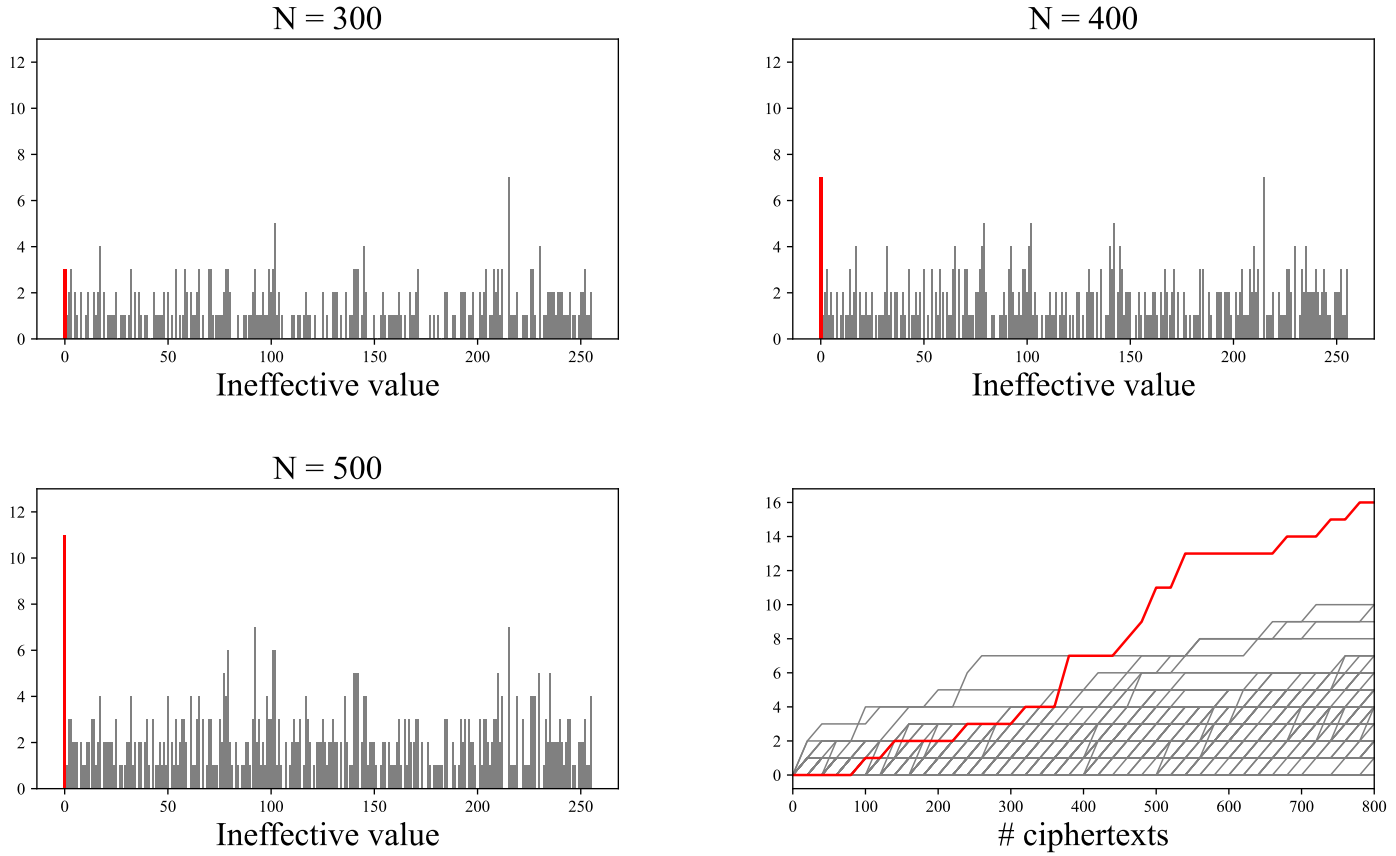
Assumption: The attacker knows **no error observed when the input of S-box is zero**

Attack Procedure:

1. Collecting fault-free ciphertexts while introducing a fault
   - ➢ Plaintexts are randomly generated
   - ➢ The fault is injected into the last round
2. Calculate the inverse S-box with candidate key (0, 1, .., 255)
   $S^{-1}(C_i \oplus K_{cand})$ and make a histogram for the value
3. Obtain the correct key where the value of zero is the highest in the histogram

Make histograms then obtain keys

✓500 operations is enough to obtain the correct key



Zero-value attacks procedure (first round)

✓SIFA attacks are conventionally targeting latter round of AES

✓**Our insight is applicable to the 1st round**

Attack Procedure (chosen plaintext attack)

1. Choose a plaintext P (sweeping from 0 to 255)
2. Do encryption and inject a fault into the first round
3. Check that the output is correct or faulty

   Correct output: the chosen value $P = Key \rightarrow$ success!!

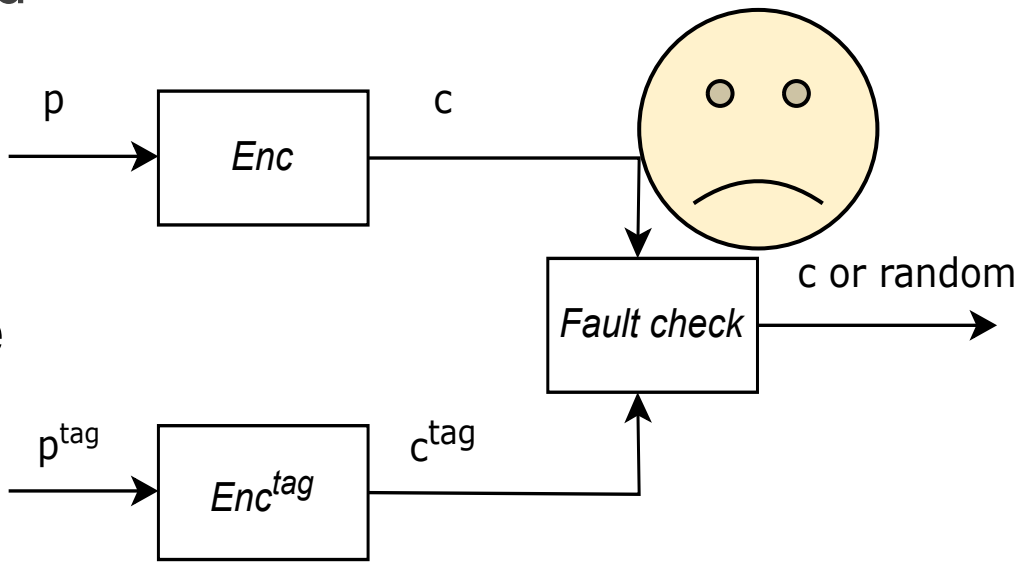
   Faulty output: the chosen value $P \neq Key \rightarrow$ go to the next value

How M&M failed the protection?

- ✓Fault checks are conducted after encryption
  - ➢ **The fault for zero-value has been already nullified**
- ✓Encryptions for data and tag are computed in parallel
  - ➢ **The tag for zero-value is also zero ($\tau^0 = 0 * \alpha$, where α is a tag key)**
  - ➢ Errors on both tag and data path were nullified

Can we prevent the attack? Yes!!

We propose a fine-grained fault check scheme



Original M&M

Fine-grained fault check with λ

✓Recall the inversion for S-box: the calculation on Stage 2

$$(a,b) \in GF((2^4)^2),\ \ \lambda((a,b)) := ab + (a+b)^2\nu$$

✓λ is **a homomorphic function** (proof is shown in our paper)

$$\therefore \lambda(data) * \lambda(\alpha) = \lambda(tag), \text{tag} := data * \alpha$$

Homomorphism on Stages 3 and 4 can be similarly proven

✓We can detect faults by just comparing

$$\lambda(data) * \lambda(\alpha) \oplus \lambda(tag) = 0?$$

Accumulate the result of λ-checks

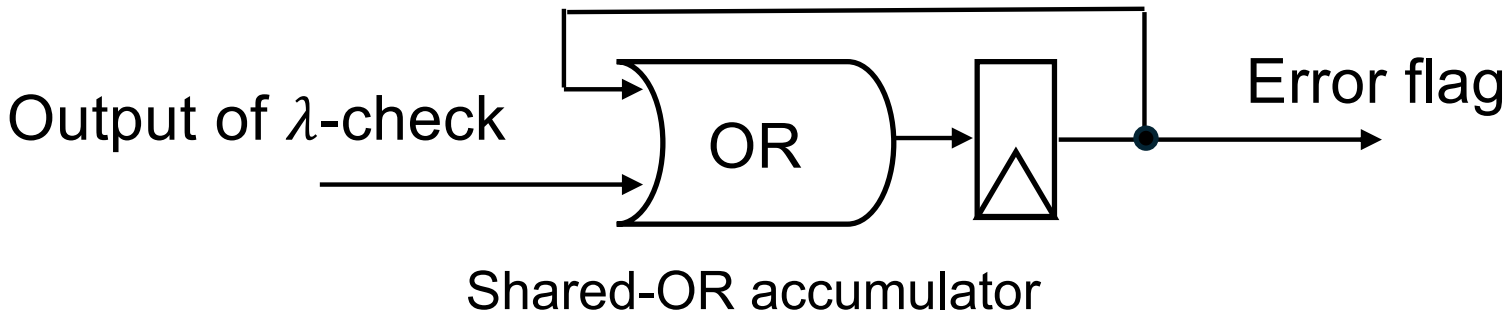
When should we refer the output of λ-checks?

✓What if we stop the encryption when faults detected..

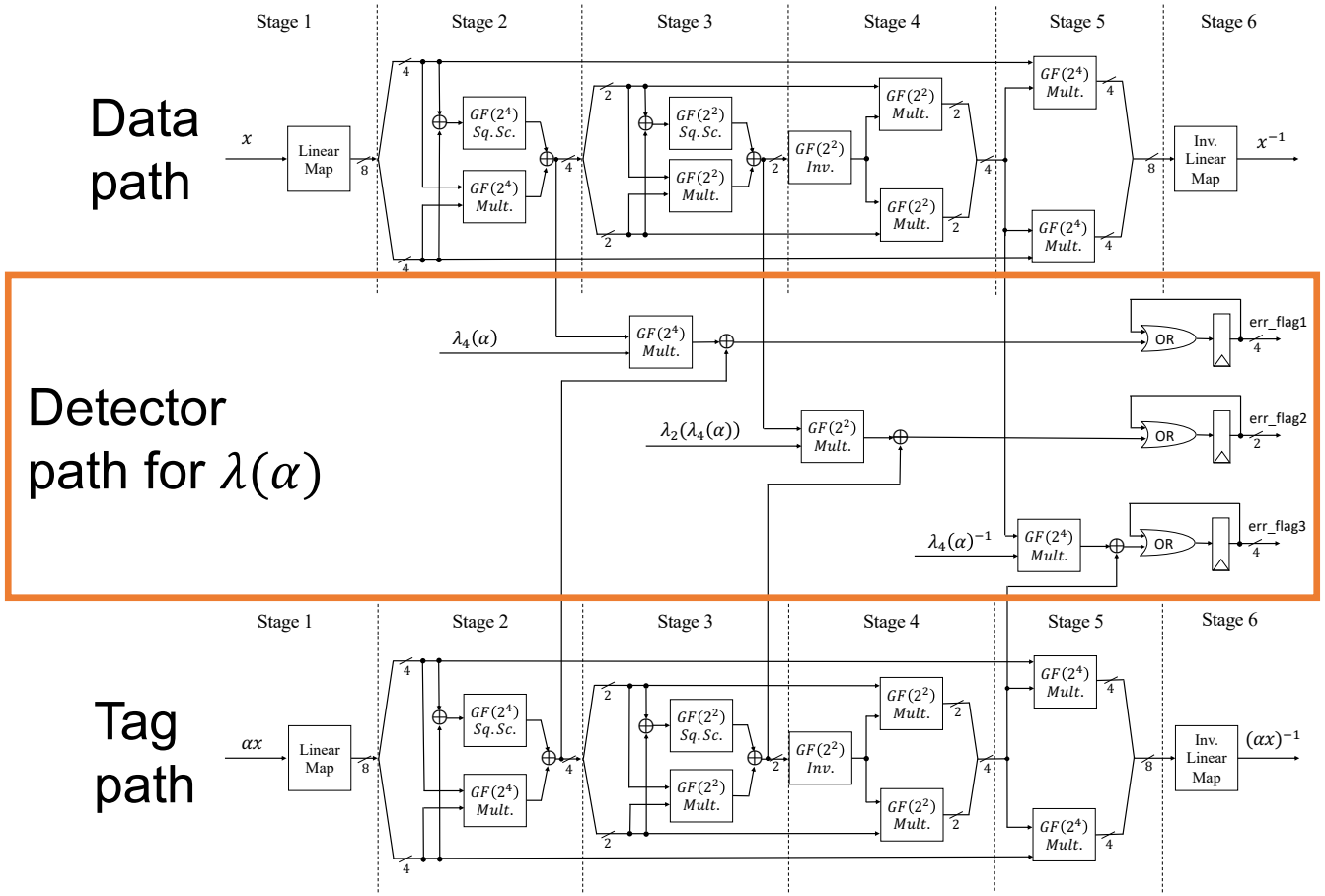
→ **An attacker would know timing of fault occurrence (when) or an exact faulty byte (where)**

✓We accumulate the result of λ-checks and refer them after the encryption finished

✓Moreover, these values are kept shared form



New S-box design with detectors



- ✓We add detectors on Stages 2-4, where are susceptible to zero-value attacks
- ✓Detectors can be placed on all Stage if needed

Delta function: Prevents combined attacks

崎山·李·宮原 研究室

✓After the encryption we compute final fault checks, named ***match check***

- Match ckech: $e_i = (\alpha * c_i) \oplus \tau_i^c$ for each byte $(i = 1,2,..,16)$

✓What if an attacker tries a combined attack?

- The attacker injects a fault Δ in α and probing the output of match check e

$$e = (\alpha \oplus \Delta) \cdot c \oplus \tau^c$$
$$\Rightarrow e = \Delta c \oplus \alpha c \oplus \tau^c$$
$$\Rightarrow e = \mathbf{\Delta c}$$

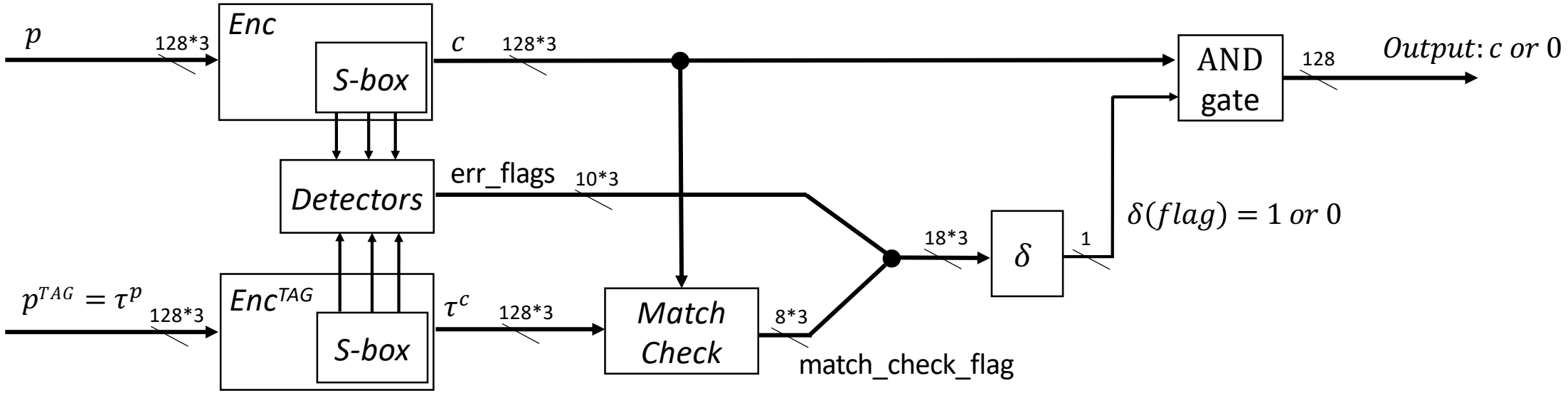
- It leaks the ciphertext c

✓We overcome this problem by using Kronecker's delta $\delta(x) = \begin{cases} 1 \text{ if } x = 0, \\ 0 \text{ otherwise} \end{cases}$

✓The advantage of this method is reducing data e to a single shared bit

Overview of our countermeasure

- ✓All values are kept shared form until the end of the operation
- ✓Our countermeasure outputs correct ciphertext or zero while M&M outputs correct or random value
  - ➢ We do not need an additional randomness
- ✓**The attacker obtains no information about faults**

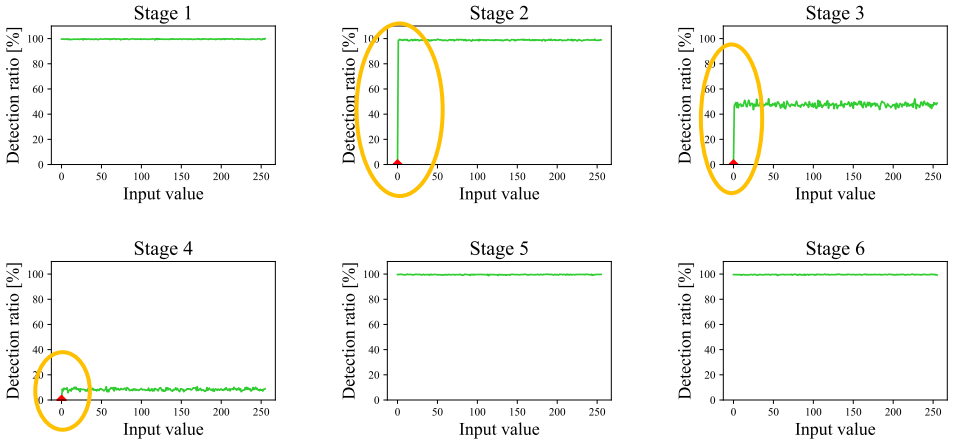


Performance comparison to the original M&M AES

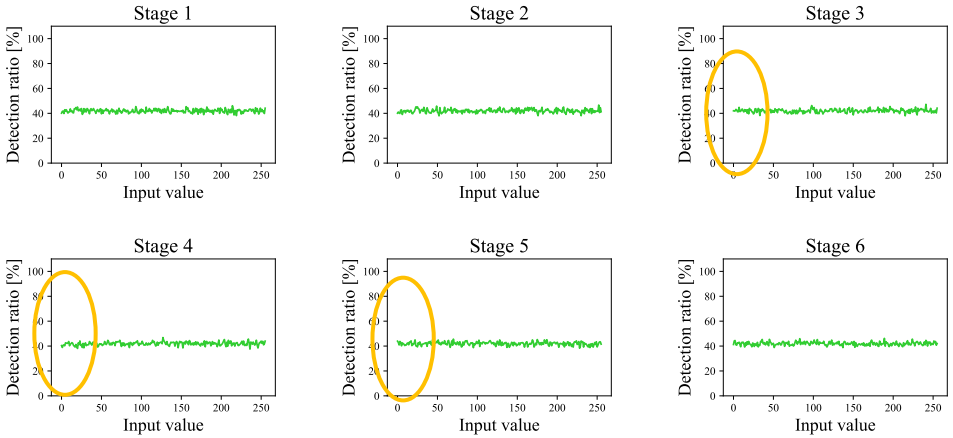
- ✓Compare implementation costs: randomness, latency, and circuit area
- ✓λ-detection M&M requires **5 additional clock cycles to compute**
  Overhead: x1.33 area and x1.62 randomness
- ✓Realized with reasonable implementation costs

	Random bits/cycle	Latency [# cycles]	Area [kGE]
S-box	564	6	18.7
Detectors	180	3	4.4
Match check	96	2	3.9
Delta function	63	5	2.3
Total			
λ-Detection M&M	564	244	**44.0**
M&M [MAN+19]	348	239	33.2

Fault Detection verification for zero-value attacks



M&M (on FPGA)



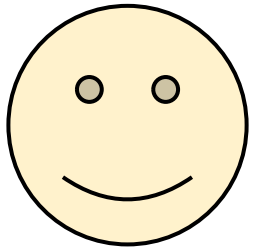
λ-detection M&M

✓Implement λ-detection M&M on SAKURA-G

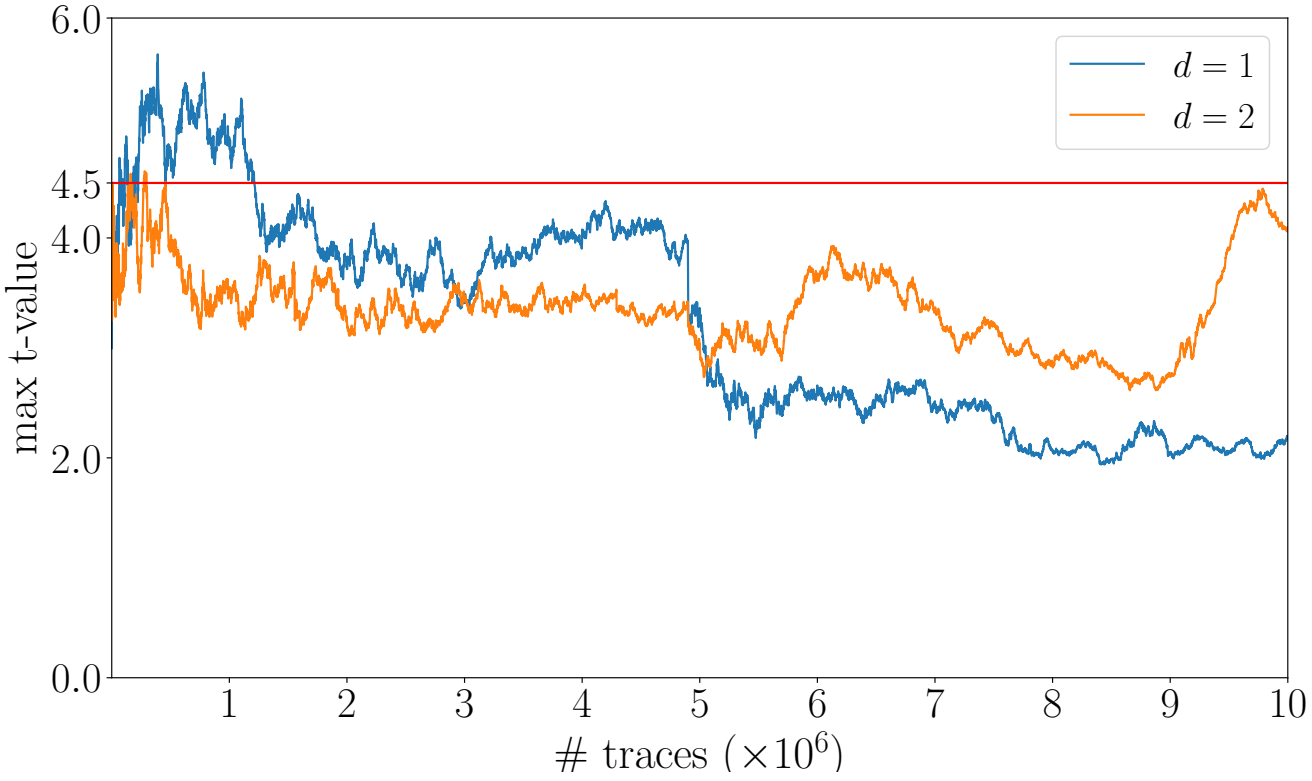
✓Evaluate the security for zero-value attacks

➢ The attack is also feasible on FPGA

✓Our countermeasure **removed biases of detection ratio at all stage**

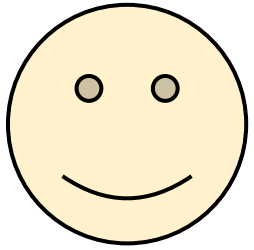


Power Leakage Detection by t-test (TVLA)



✓Conduct the 1st and 2nd –order t-test

✓No leakage detected up to 10 M traces



Conclusion

- ✓Developed ASIC evaluation boards
- ✓Pointed out the flaw of Canright's AES S-box design
- ✓Demonstrate SIFA2-like attacks against M&M AES
  - ➢ The attack can be applicable to other masked AES implementations
- ✓Proposed a fine-grained fault check scheme λ-detection
  - ➢ Our countermeasure does not give the attacker any information about the fault
- ✓Conducted security evaluation
  - ➢ No leakage found so far for bath FA and SCA

Thank you so much!!

崎山·李·宮原 研究室

Any Questions?

h.haruka@uec.ac.jp