

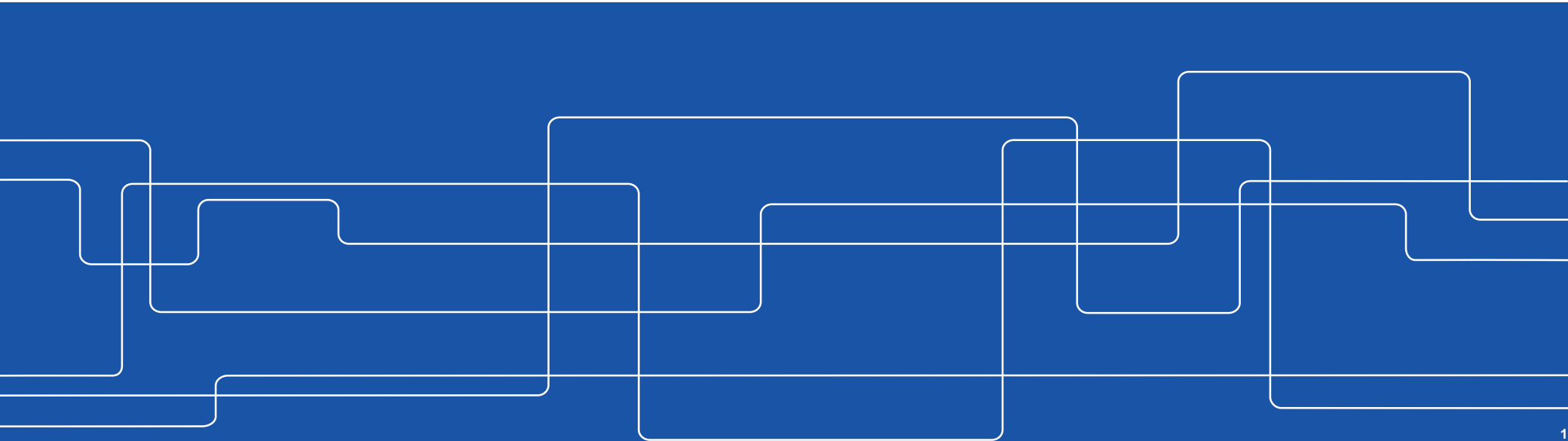


# Stronger Together: Bridging Logic Synthesis and Side-Channel Analysis

Elena Dubrova

Department of Electrical Engineering

EECS/KTH

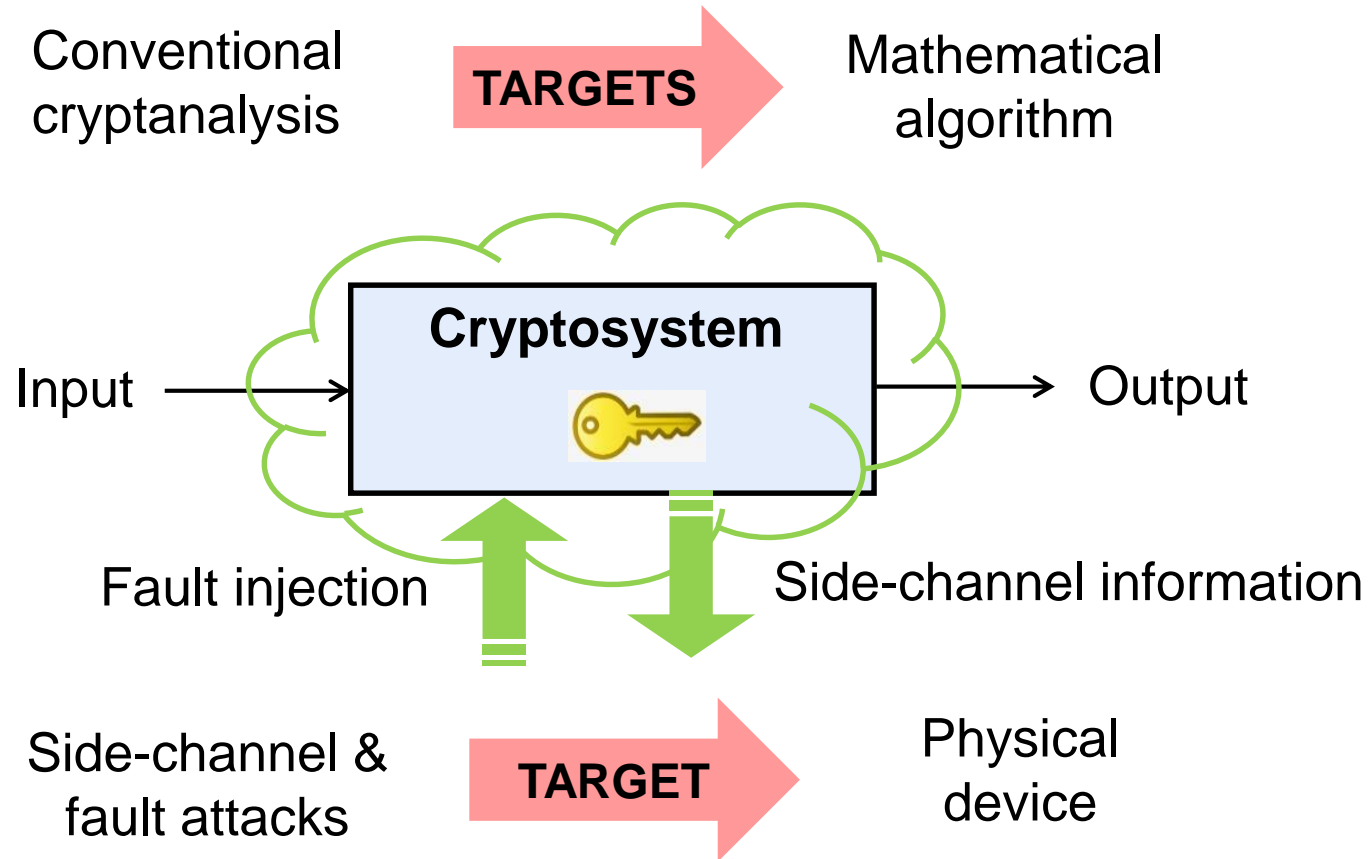




# Outline

- Motivation
- Introduction to side-channel analysis
- Attacks examples:
  - Power analysis of USIM card
  - RF-based analysis of nRF52832 Bluetooth system-on-chip
  - SAT-assisted power analysis of AES
- Summary & Future work

# What is side-channel analysis?



# Motivation: In the near future ...

- Millions **not so well protected** Internet-connected devices will be involved in services related to confidential data
  - Wearables
  - Connected cars
  - Smart home



source: <http://www.wearables.com/5-baby-monitors-wearable-infant-tech/>

source: <http://www.dqindia.com/cognizant-is-betting-big-on-connected-cars/>

source: <https://blog.econocom.com/en/blog/smartbuilding-and-bms-a-little-glossary/>

# THE FBI WARNS THAT CAR HACKING IS A REAL RISK



ANDY GREENBERG SECURITY 07.21.15 6:00 AM

## HACKERS REMOTELY KILL A JEEP ON THE HIGHWAY —WITH ME IN IT



ANDY GREENBERG SECURITY 08.11.15 7:00 AM

## HACKERS CUT A CORVETTE'S BRAKES VIA A COMMON CAR GADGET



# FBI: Hacker claimed to have taken over flight's engine controls

By [Evan Perez, CNN](#)

🕒 Updated 0119 GMT (0919 HKT) May 19, 2015



**(CNN)** — A cybersecurity consultant told the FBI he hacked into computer systems aboard airliners up to 20 times and managed to control an aircraft engine during a flight, according to federal court documents.



SECURITY

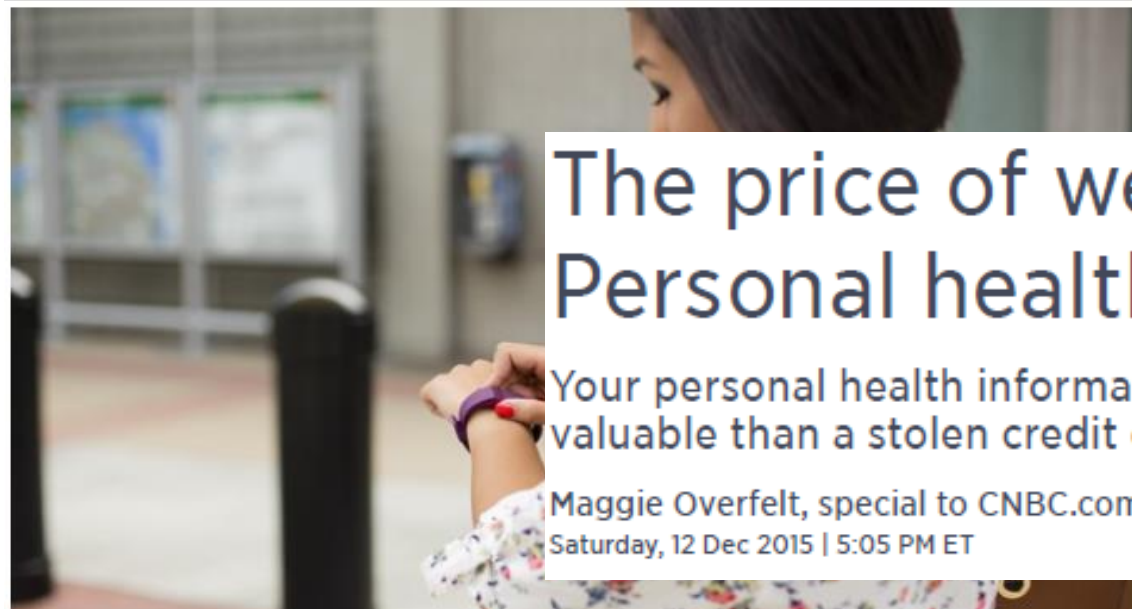
# Hacker looks to sell 9.3 million alleged patient healthcare records on the dark web

By James Rogers

Published June 28, 2016

## What does Fitbit hacking mean for wearables and IoT?

BY STEPHEN COBB POSTED 12 JAN 2016 - 02:49PM



## The price of wearable craze: Personal health data hacks

Your personal health information is about 10 times more valuable than a stolen credit card number on the black market.

Maggie Overfelt, special to CNBC.com

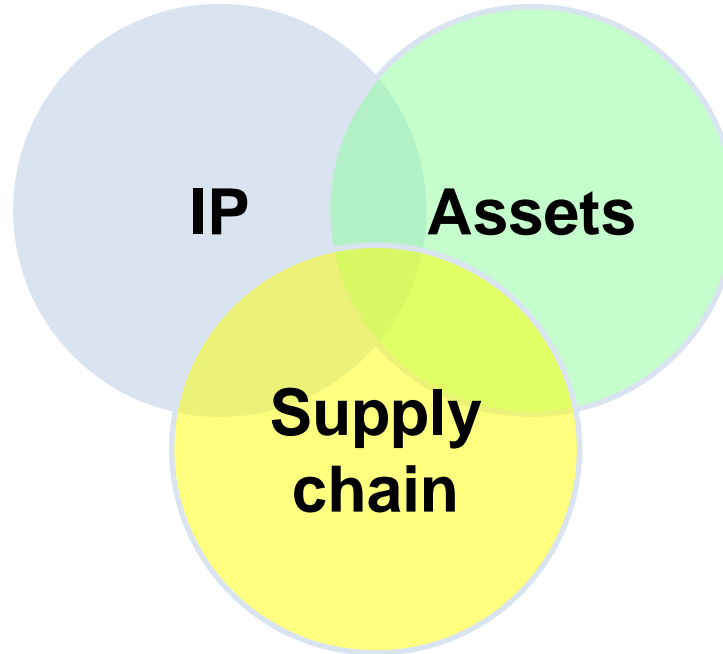
Saturday, 12 Dec 2015 | 5:05 PM ET

# What needs protection?

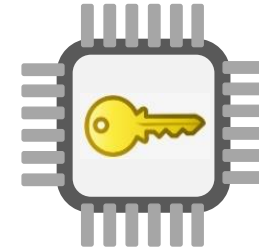
Saab@MarcusWandt



Proprietary designs  
Proprietary algorithms  
Proprietary bitstreams



source: <http://www.publicintegrity.org/2011/11/07/7323/counte>



On-device data  
On-device keys  
TRNGs

Preventing Hardware Trojans,  
counterfeit, overproduction

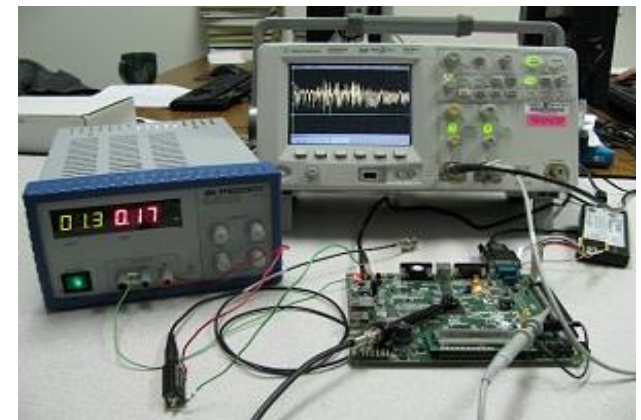


# A Typical Path from CAD to Hardware Security



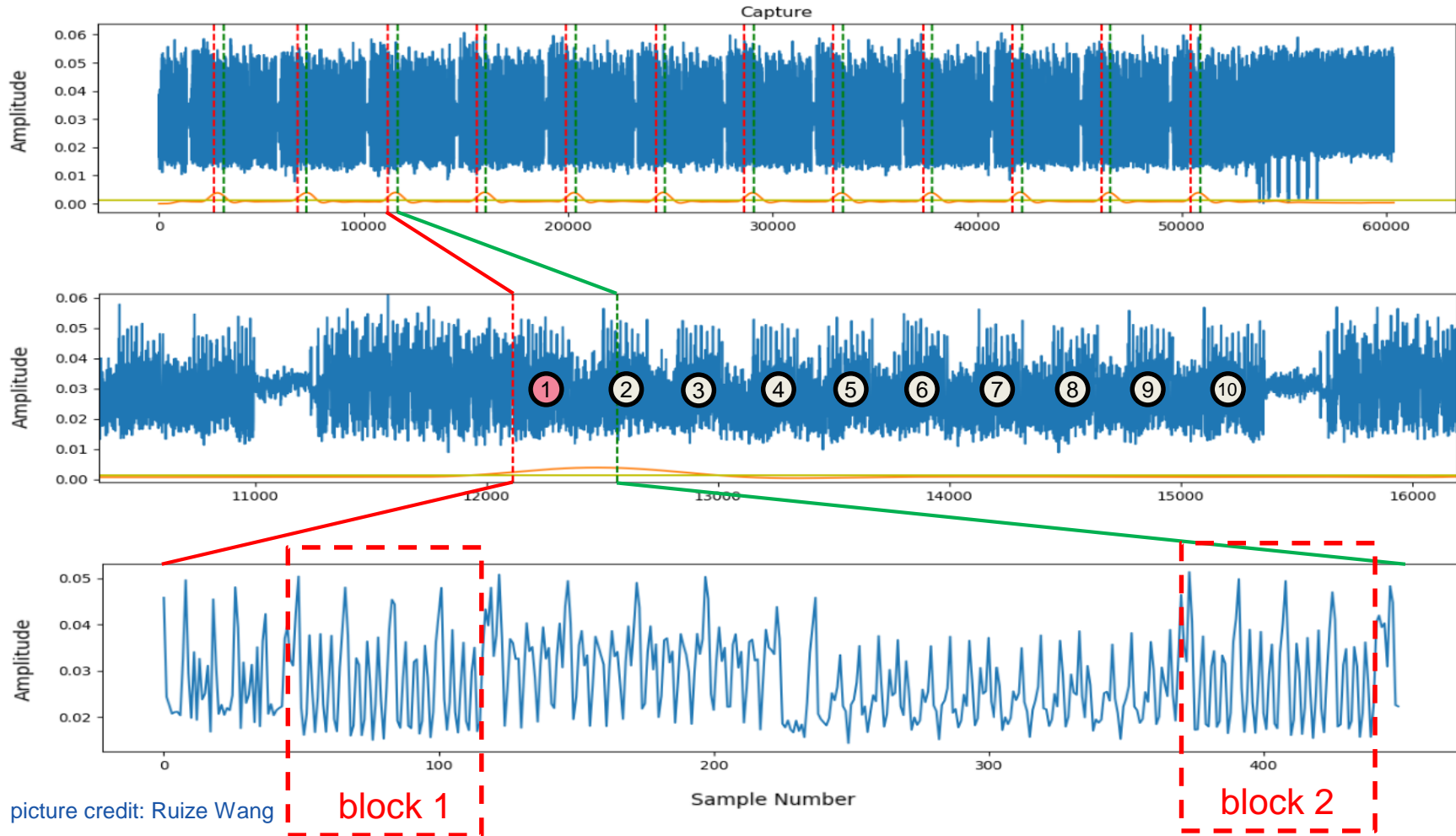
# How side-channel attacks work

- Algorithms are implemented in CPUs, FPGAs, ASICs, ...
- Different operations may consume different amount of power/time
- The same operation executed on different data may consume different amount of power/time
- It is possible to recognize which **operations and data are processed** from power/EM traces/timing
  - ML methods are useful

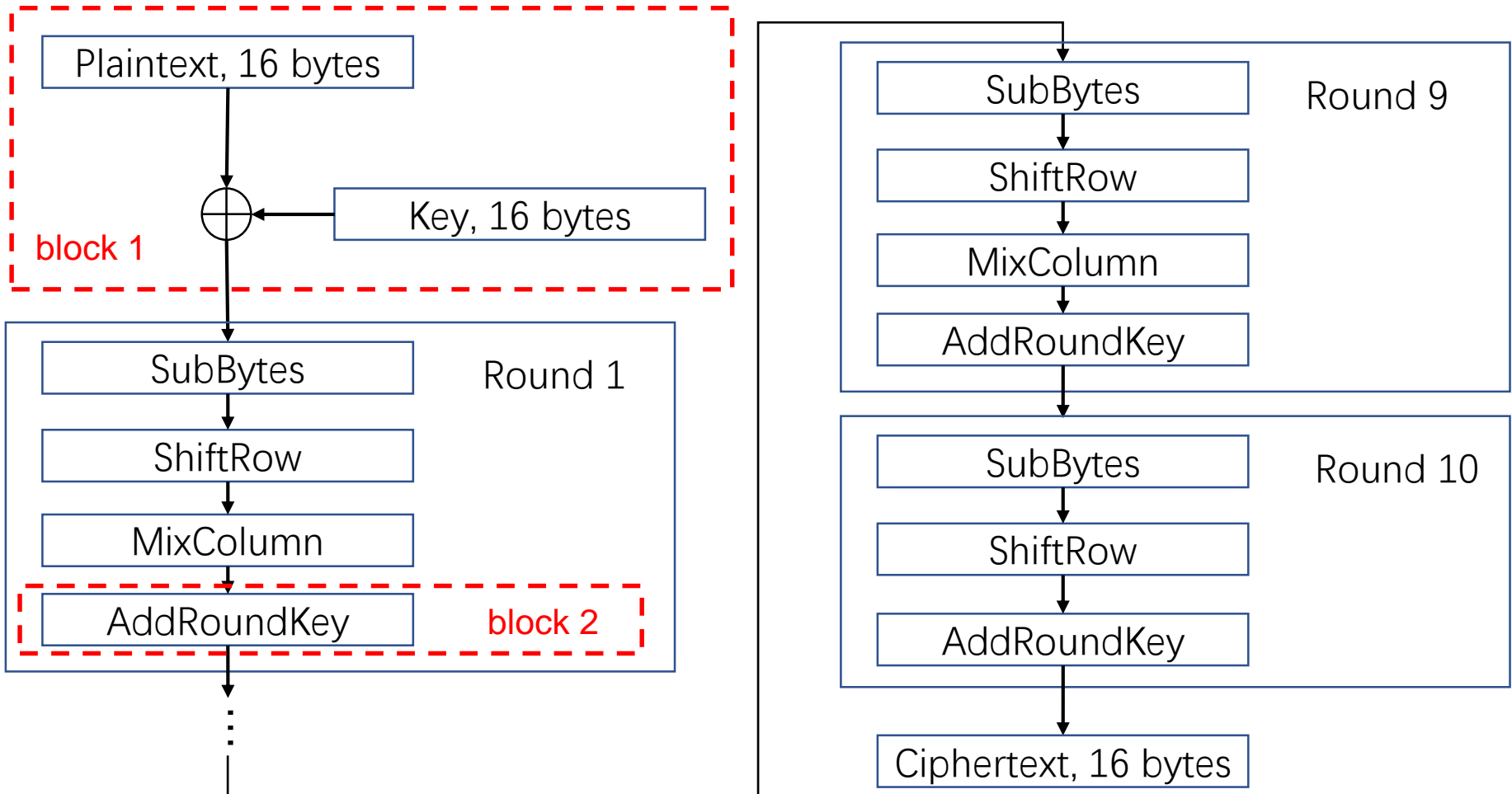


source: [hackaday.com](https://hackaday.com)

# Power trace of AES-128 execution on Cortex-M4

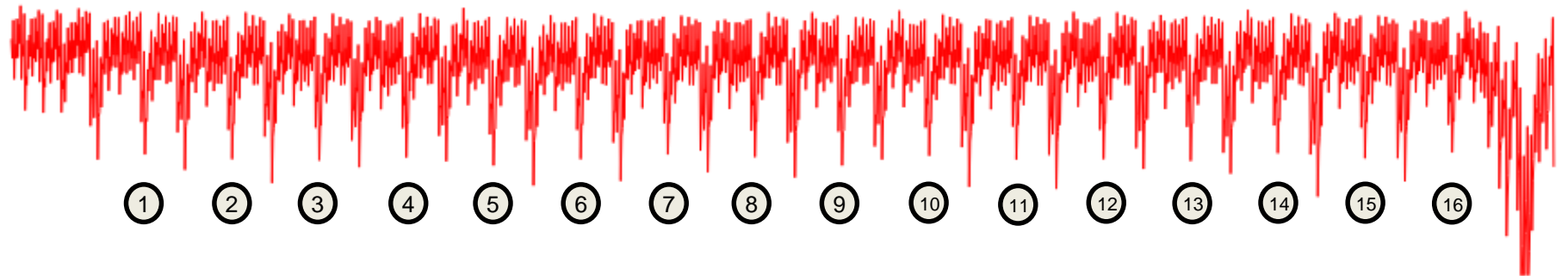


# AES-128



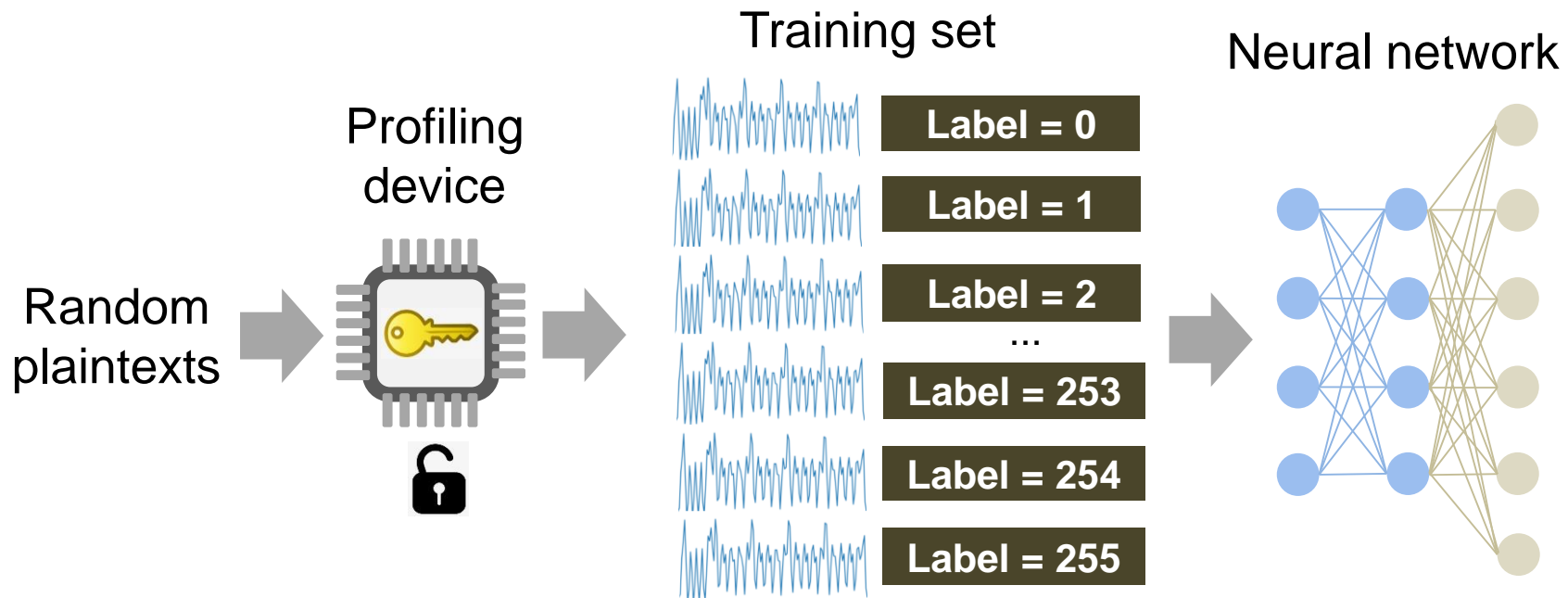
picture credit: Ruize Wang

# Power trace of S-Box execution on 8-bit MCU



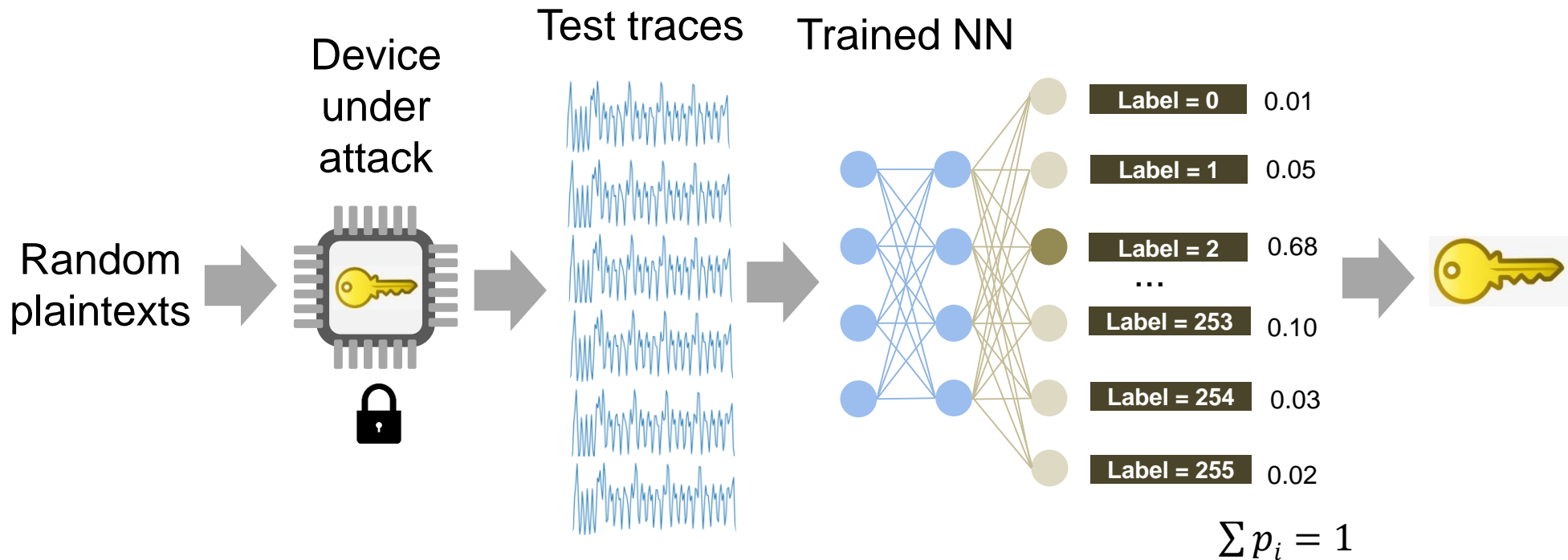
# Deep learning in side-channel analysis

**Profiling stage:** Train a neural network (NN) using traces from profiling devices

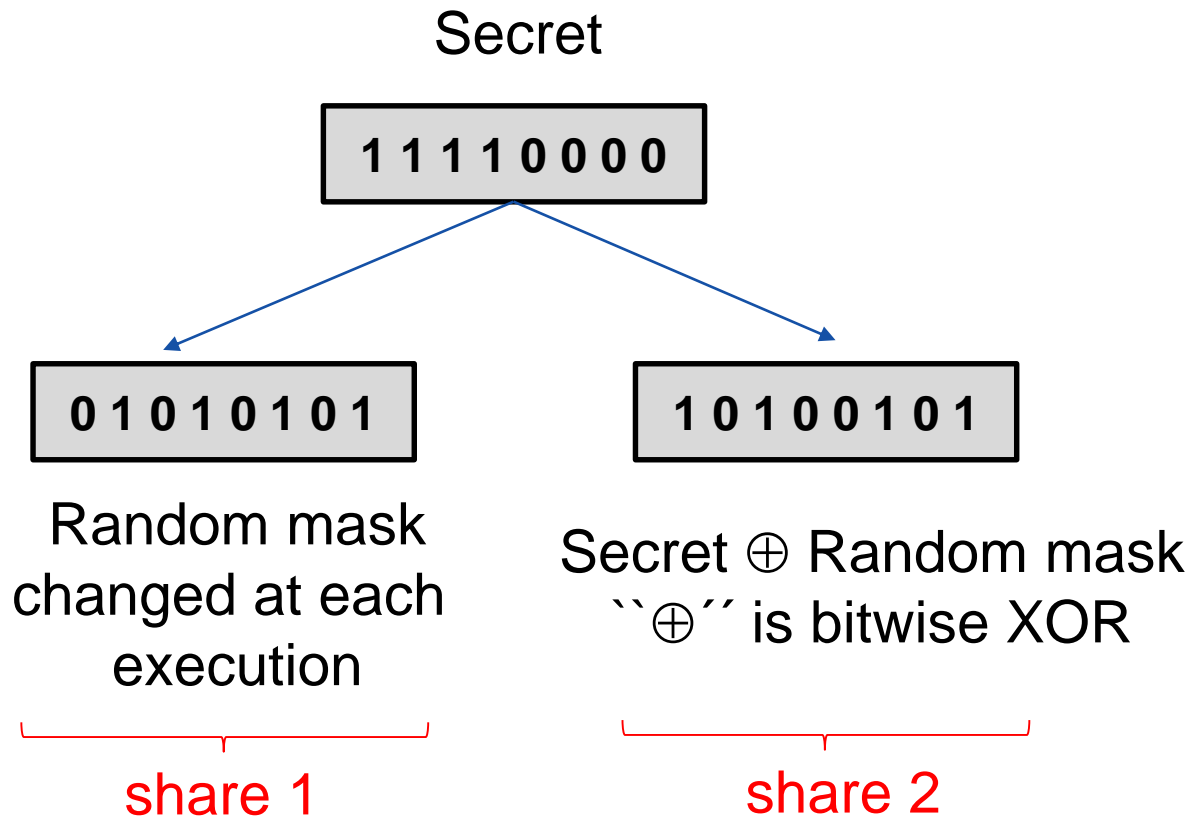


# Deep learning in side-channel analysis, cont.

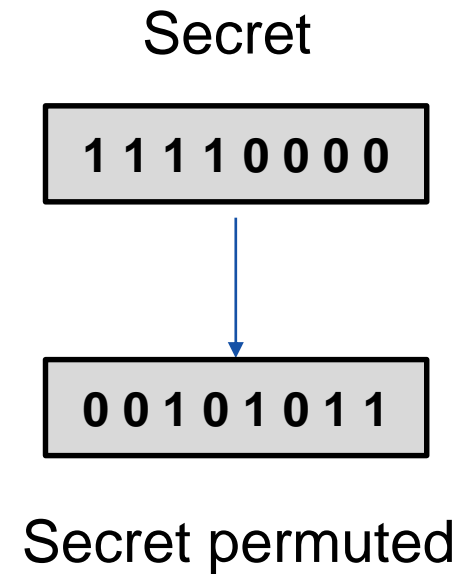
**Attack stage:** Use the trained NN to classify traces from the device under attack



# Masking and shuffling countermeasures



First-order Boolean masking

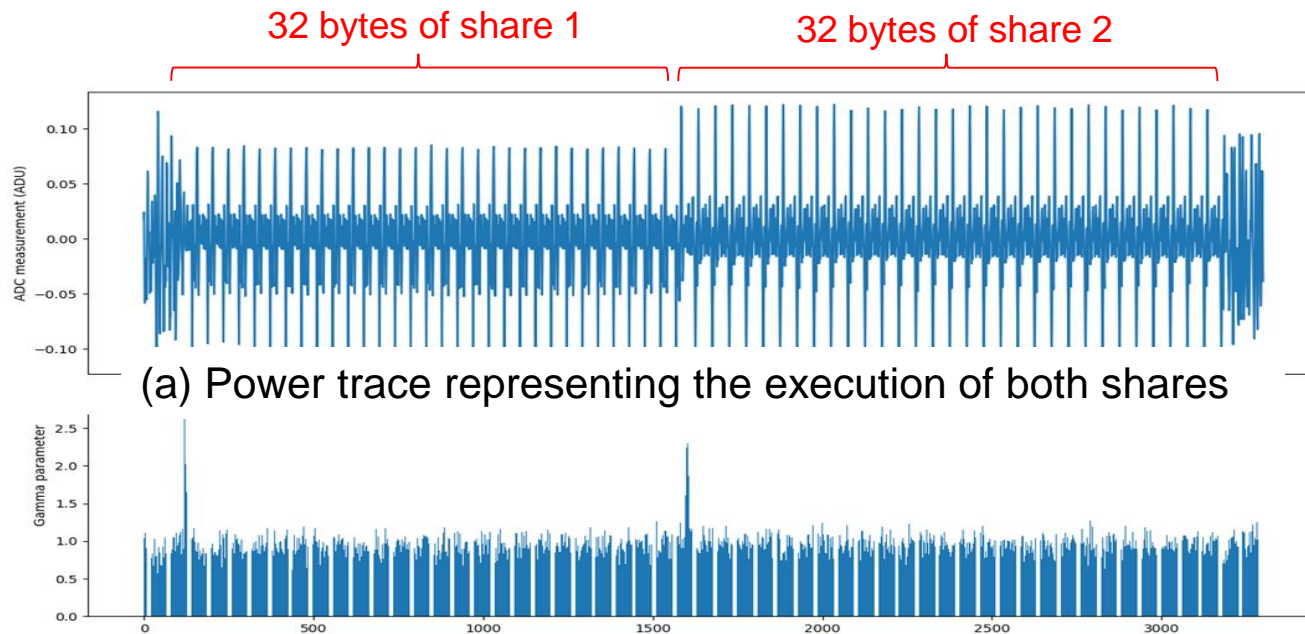


Shuffling

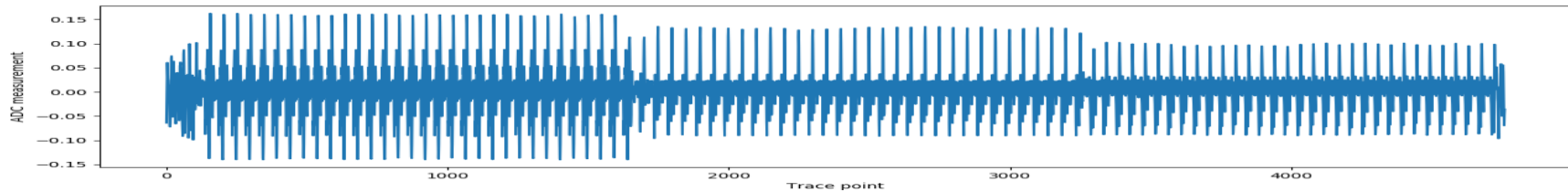
# How masking can be defeated

- If all shares are given as input to the NN, it may learn XOR

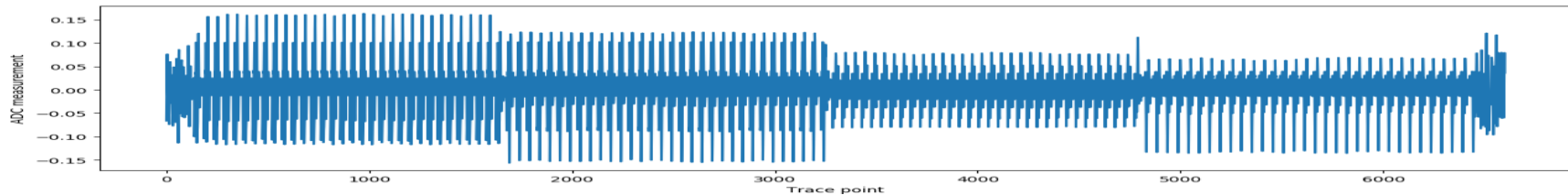
*Breaking a Fifth-Order Masked Implementation of CRYSTALS-Kyber by Copy-Paste,*  
Dubrova, E., Ngo, K., Gärtner, J., Real World Crypto'2023



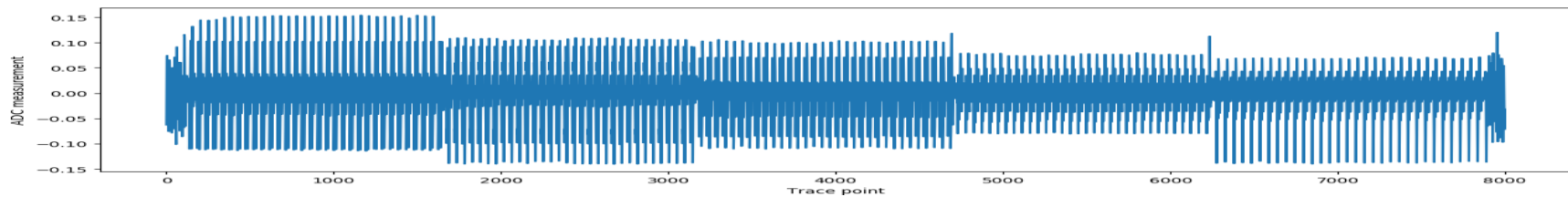
# More shares $\Rightarrow$ more 32-byte blocks



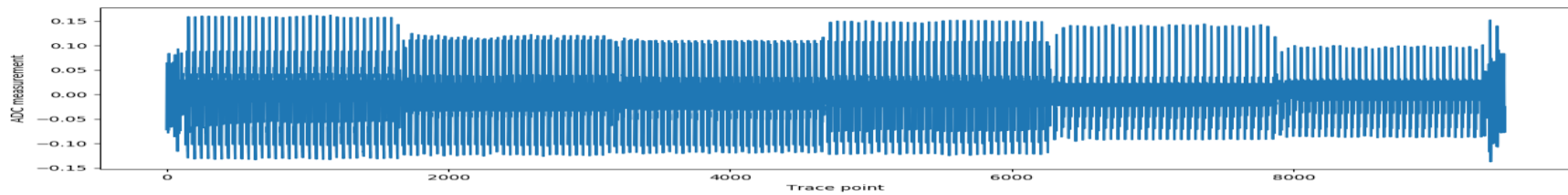
3



4



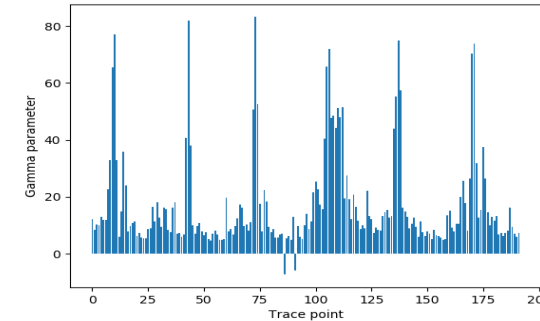
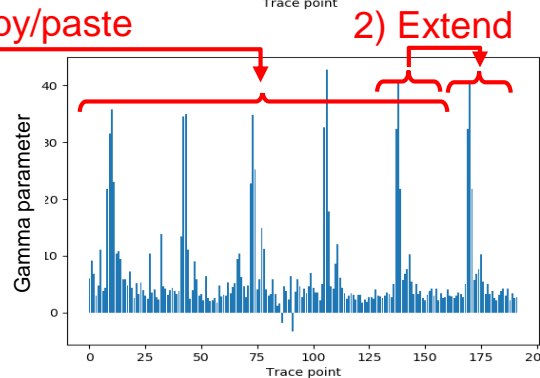
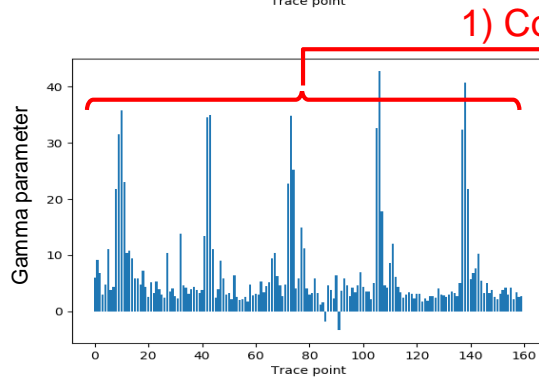
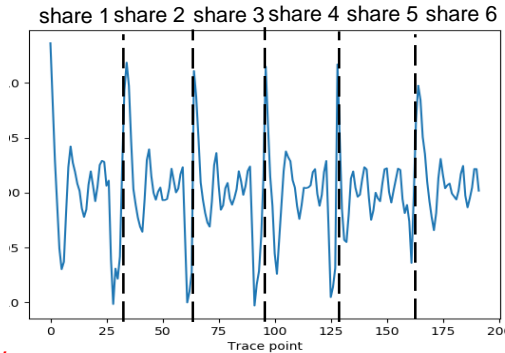
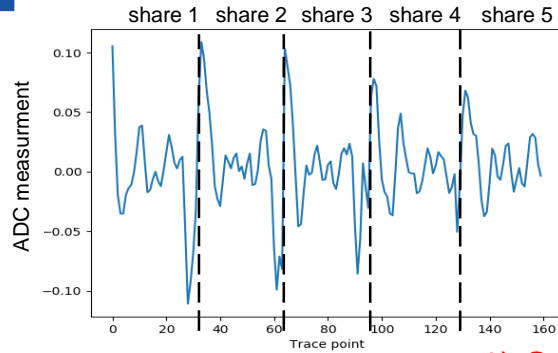
5



6

## 5 shares

## 6 shares



Power traces  
(cut & concatenated  
 $i^{\text{th}}$  bits of shares)

Weights of input  
BatchNorm layer  
before training

3) Train

Weights of input  
BatchNorm layer  
after training

# How shuffling can be defeated

photo credit: Sönke Jendral

Initialization

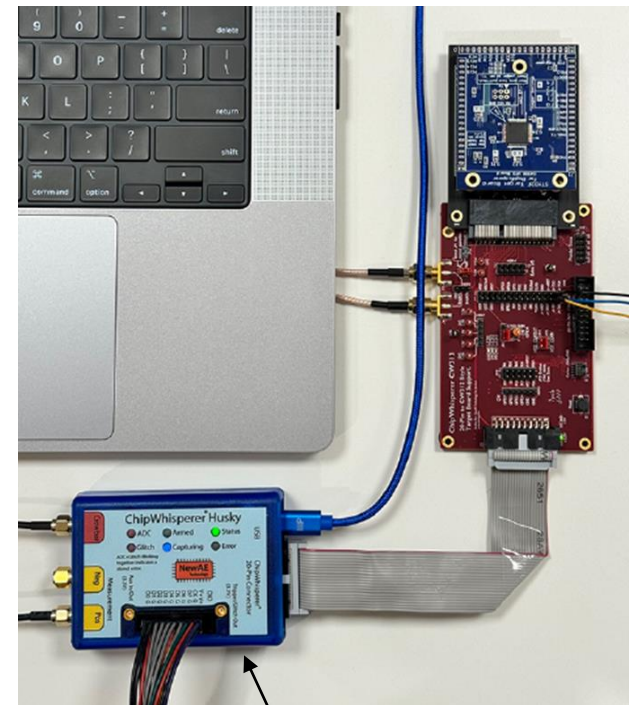
```

init_loop: ←
    strb.w    r3, [r2, #1]!
    adds     r3, #1
    cmp.w    r3, #256
    bne.n    init_loop
    add     r3, sp, #32
    addw    r6, sp, #287
    rsb     r4, r3, #1
  
```

Shuffling

```

shuffle_loop: ←
    bl      rng_get_random_blocking
    adds    r3, r4, r6
    udiv    r2, r0, r3
    mls     r0, r2, r3, r0
    add     r3, sp, #32
    add     r1, sp, #32
    ldrb    r3, [r3, r0]
    ldrb    r2, [r6, #0]
    strb    r2, [r1, r0]
    strb.w  r3, [r6], #-1
    cmp     r6, r1
    mov     r3, r1
    bne    shuffle_loop
  
```



ChipWhisperer-Husky

Voltage glitch injection  
(0.89 success rate)

# Example 1: USIM card attack

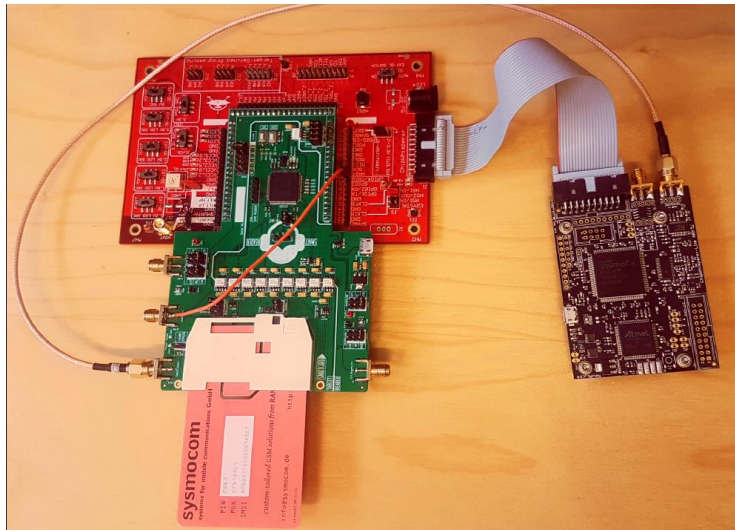
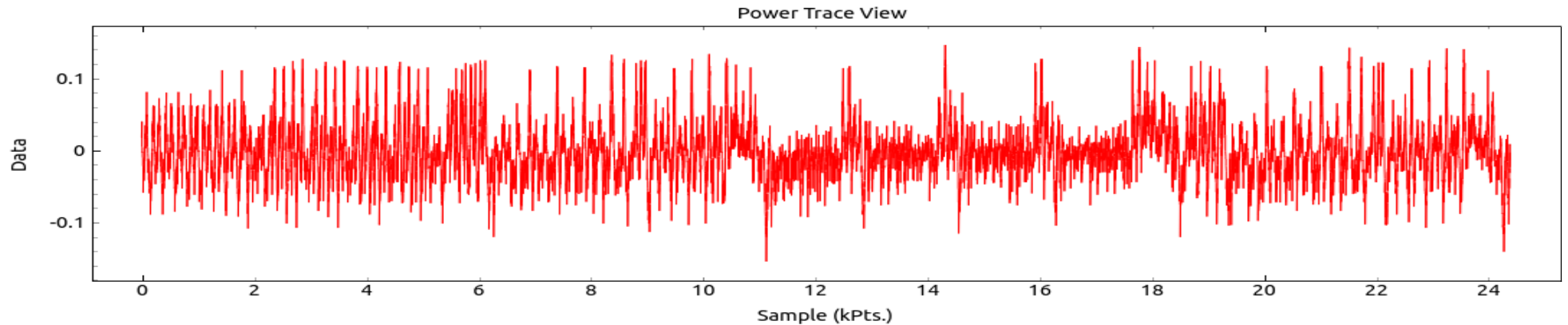


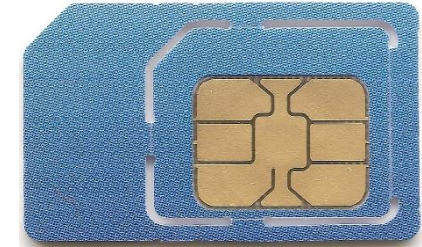
photo credit: Martin Brisfors

USIM's long-term key can be extracted from the USIM using 4 power traces on average (max 20)

*How Deep Learning Helps Compromising USIM,*  
M. Brisfors, S. Forsmark, E. Dubrova,  
CARDIS'2020, Nov. 18-19, 2020

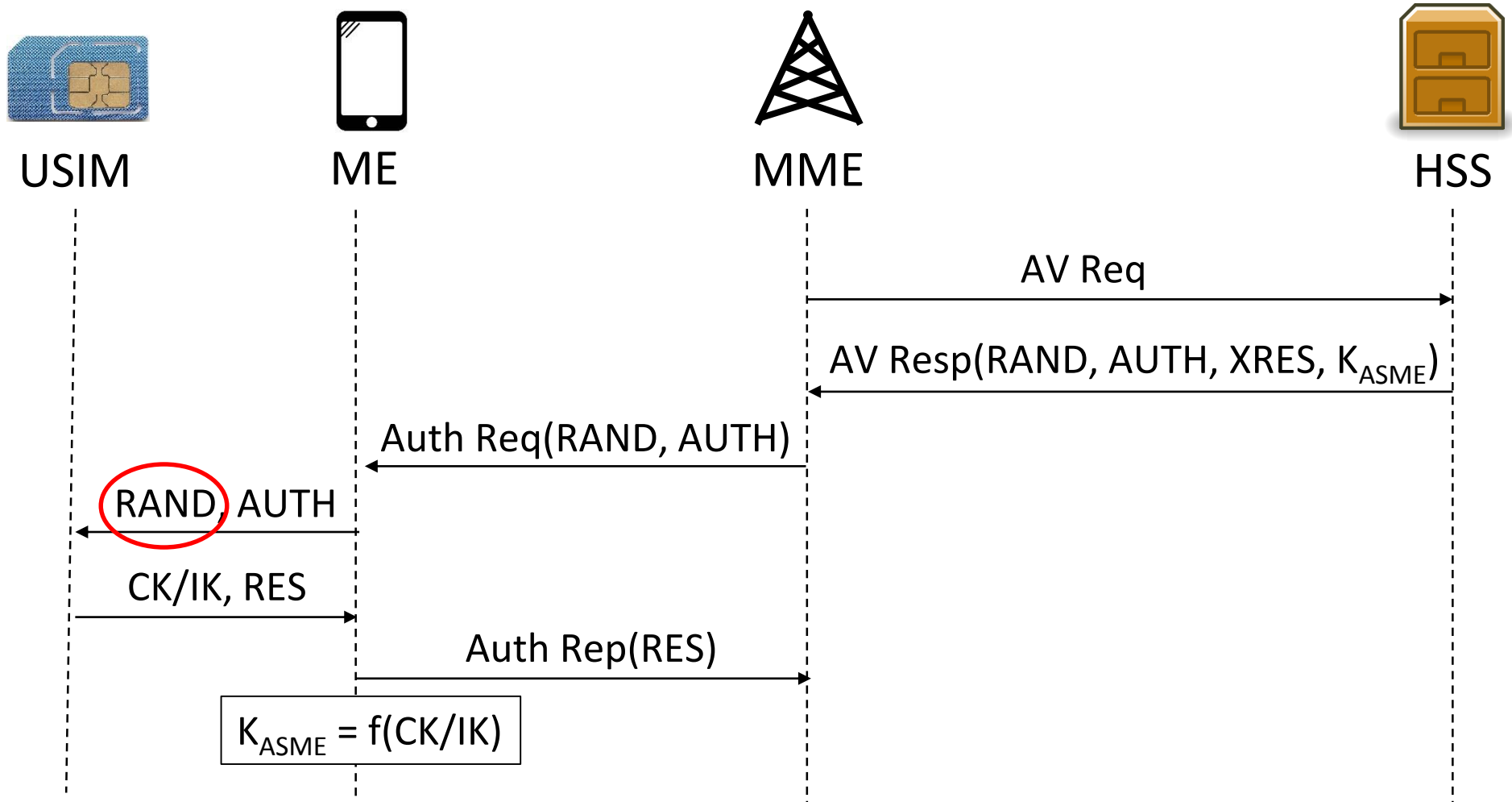
# Universal Subscriber Identity Module (USIM)

- USIM is a type of smart card
- Contains:
  - Secret key  $K$  pre-shared with home subscriber server
  - International Mobile Subscriber Identity (IMSI)
  - Operator Variant Algorithm Configuration Field (OP)
- All cryptographic operations involving  $K$  are carried out within the USIM



Source:Telefónica O<sub>2</sub> Europe

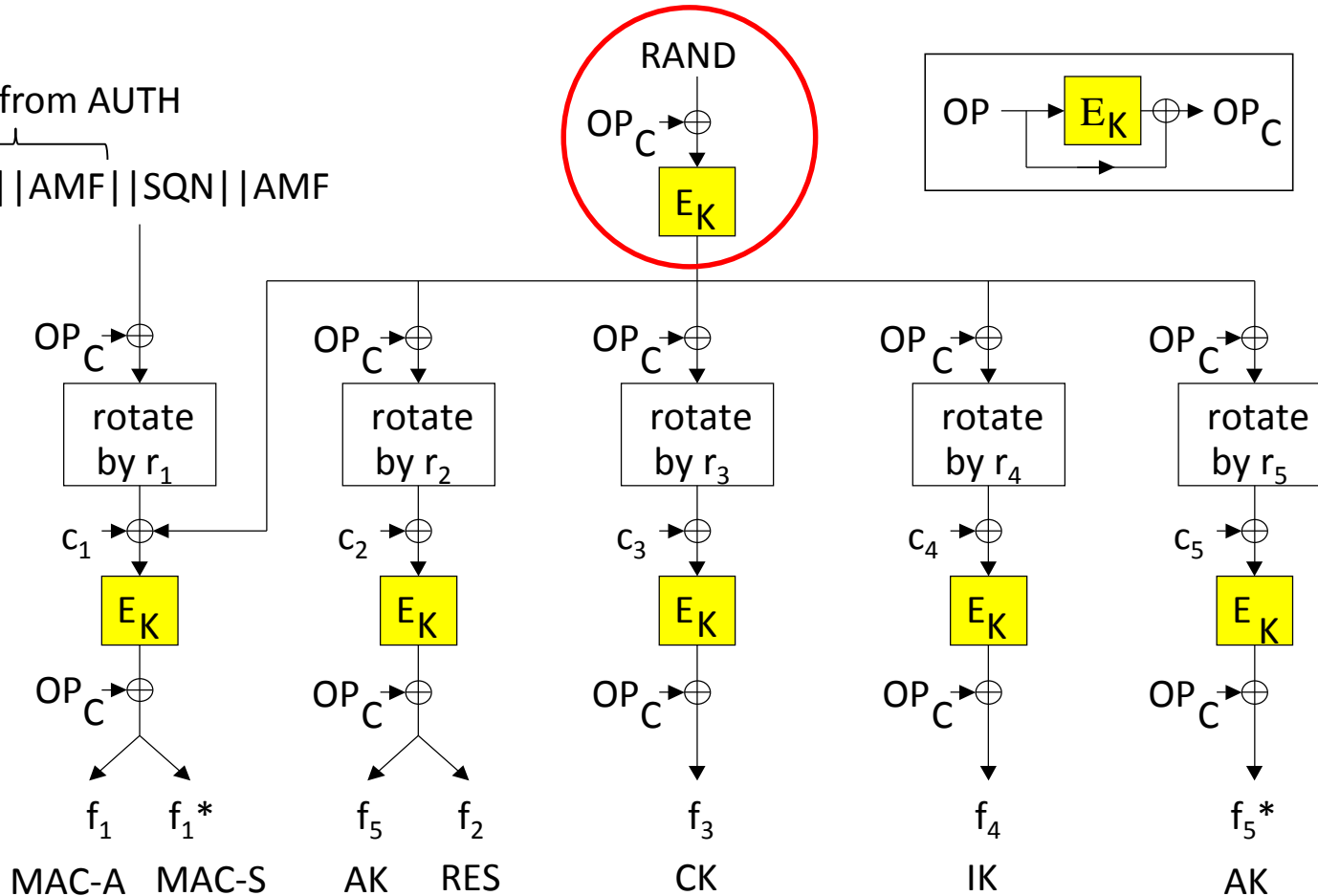
# Authentication and Key Agreement (AKA) in 4G



# MILENAGE algorithm

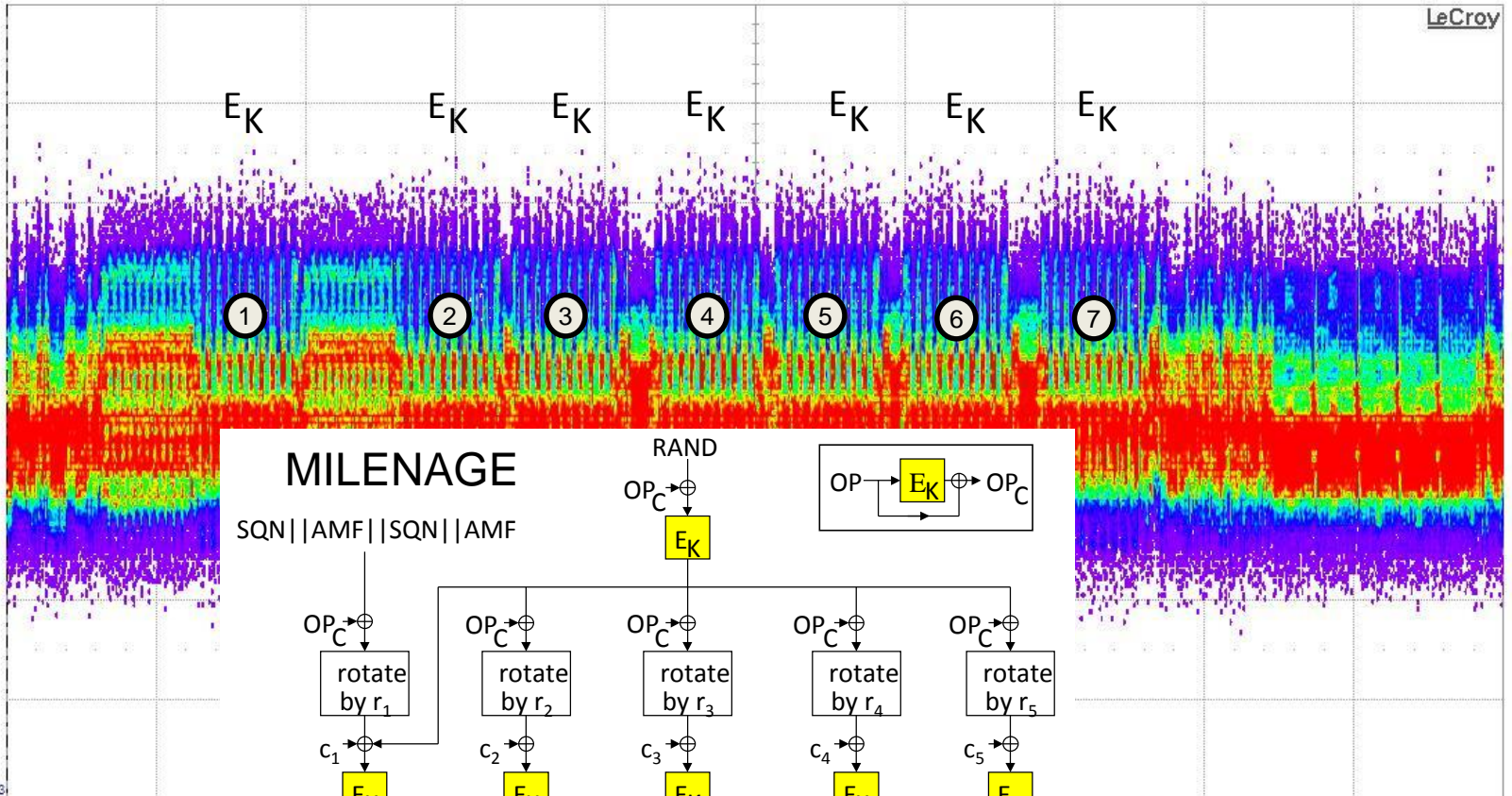
Derived from AUTH

SQN || AMF || SQN || AMF



# Zoomed interval of MILENAGE execution

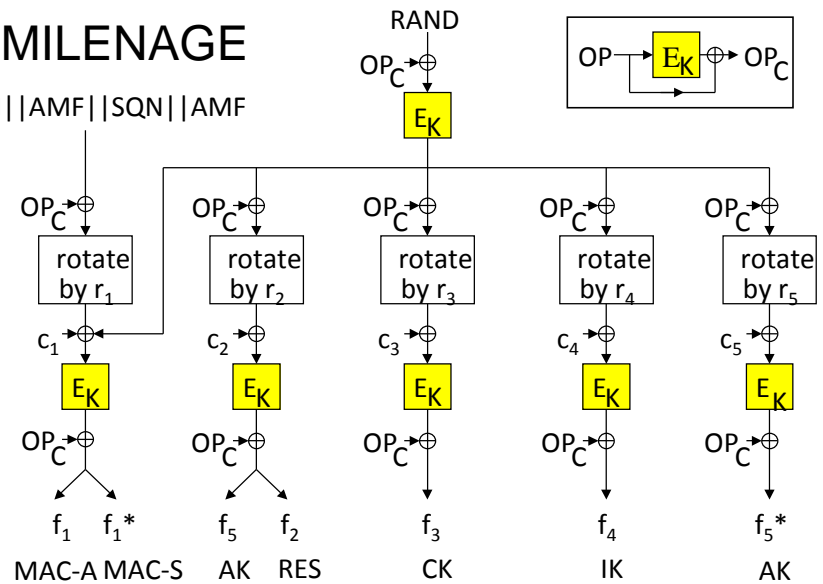
Idx Edge Time  
No. ... No Data...



LeCroy

## MILENAGE

SQN | AMF | SQN | AMF



3) P11:--- P12:---

Measure value status  
10.0 mV/div  
-42.40 mV

P1:ampl(C3) 49.6 mV  
P2:freq(C3) 1.92793 MHz  
P3:freq(C3) 1.92793 MHz

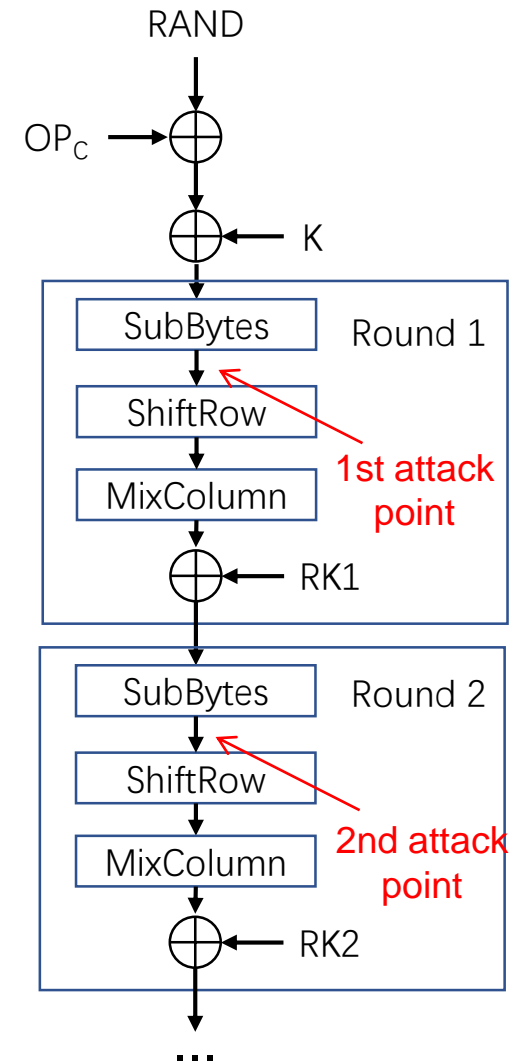


Tbase -49.48 ms Trigger C4 DC  
1.00 ms/div Stop 1.10 V  
2.50 MS 250 MS/s Edge Positive  
X1= 44.480000 ms

picture credit: Martin Brisfors

# Secret key recovery method

- In MILENAGE,  $RAND \oplus OP_C$  is first computed and then the result is encrypted
- If  $E_k$  is AES-128, the key  $K$  can be recovered in two steps:
  1. Recover  $OP_C \oplus K$  using S-box output in the 1st round as the attack point
  2. Recover the 1st round key,  $RK1$ , using the S-box output in the 2nd round as the attack point
  3. Compute  $K$  from  $RK1$
  4.  $OP_C = (OP_C \oplus K) \oplus K$





## Cost of USIM attack

- The attack can be done with a low-cost equipment

ChipWhisperer Lite	250 USD
ChipWhisperer UFO board	240 USD
USIM reader (LEIA)	3780 SEK

**< 1000 USD**

- If trained NN models are available, the attack does not require expert-level skills



Realistic threat

## Example 2: nRF52832 Bluetooth SoC attack

- Fully parallel ASIC implementation of AES in CBC-MAC authenticated encryption mode
- Uses RF signals as a side channel
  - Demodulated using a SDR
  - No physical modification of device
    - ⇒ No tamper-evidence
  - Remote attack
- Full key can be extracted from 450 M traces captured at 1 m distance



photo credit: Katerina Gurova

*Is Your Bluetooth Chip Leaking Secrets via RF Signals?*, Y. Ji, E. Dubrova, R. Wang, Real World Crypto'2025, <https://eprint.iacr.org/2025/559>

# Attack setup

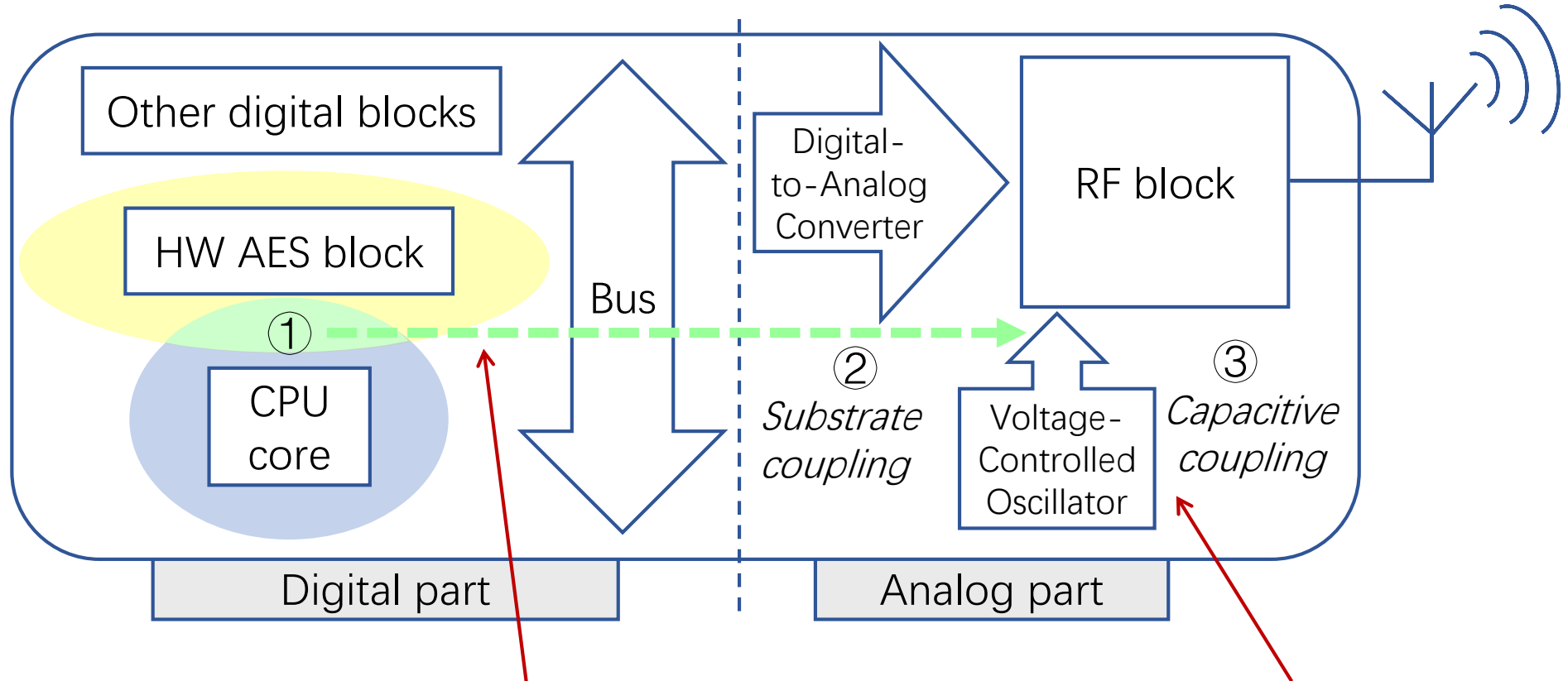
- Equipment:
  - Antenna at 1m distance captures RF signals
  - Software-defined radio demodulates RF signals at  $2.4\text{GHz} + n \cdot 16\text{ MHz}$

Bluetooth band  
frequency

HW AES  
clock frequency



# Why mixed-signal circuits may leak secrets via RF?

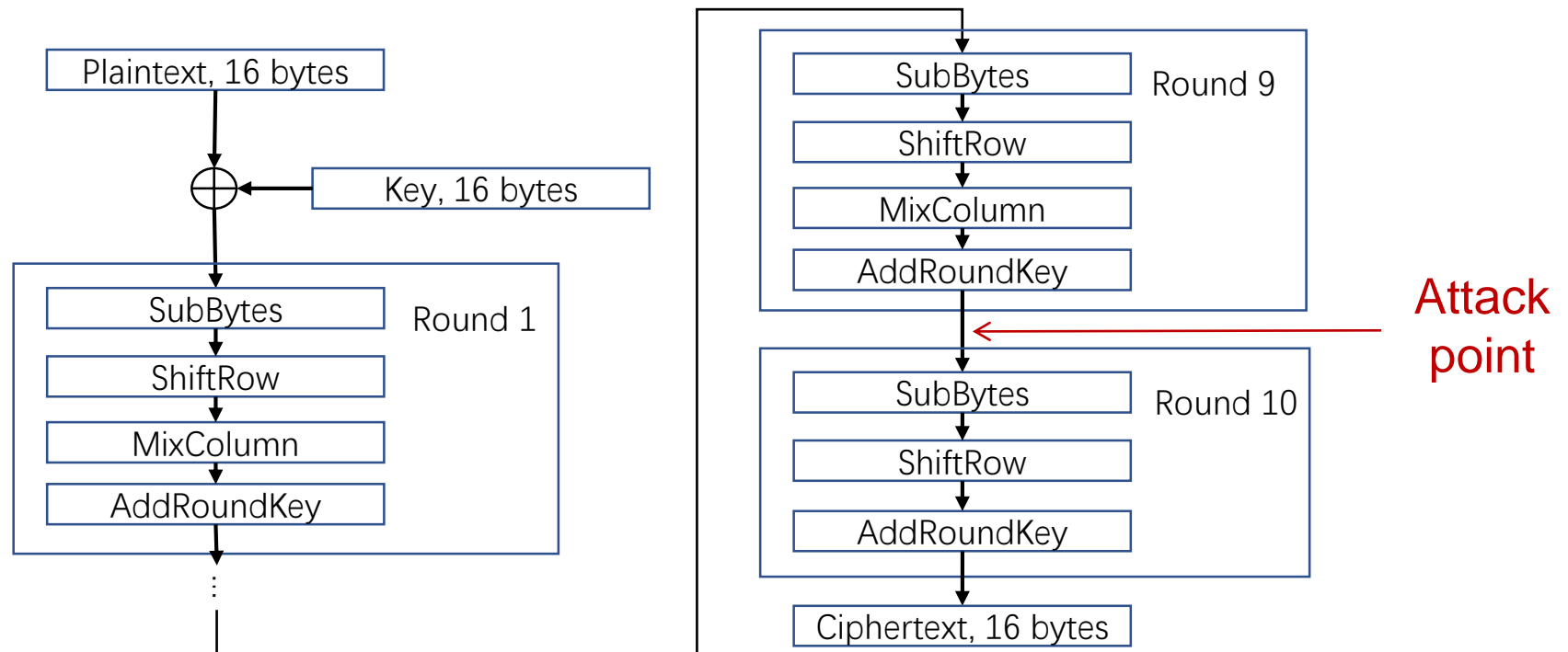


Current fluctuations in digital circuits induce noise in the shared silicon substrate, affecting analog components

RF signal generated by VCO may be affected

# Secret key recovery methods

- Correlation analysis (CA), non-profiled attack
- Deep learning (DL), profiled attack



## Results for 450 M traces (4.7 days of capture)

Capture method	Key #	Device #	Attack method	# Incorrect key bytes, $k$	# Enum.	Enum. time
Cable	Key 1	Device 1	CA	4	$2^{42.8}$	21.7 hrs
	Key 2	Device 2	CA	5	$2^{52.1}$	1.53 yrs
Antenna	Key 1	Device 1	CA	11	$2^{102.1}$	-
	Key 2	Device 2	CA	12	$2^{106.8}$	-

For CA attack, positions of incorrect bytes are unknown  $\Rightarrow$  #Enum. =  $\binom{16}{k} 2^{8k}$

## Results for 450 M traces (4.7 days of capture)

Capture method	Key #	Device #	Attack method	# Incorrect key bytes, $k$	# Enum.	Enum. time
Cable	Key 1	Device 1	CA	4	$2^{42.8}$	21.7 hrs
			<b>DL</b>	<b>1</b>	<b><math>2^8</math></b>	<b>0.003 ms</b>
	Key 2	Device 2	CA	5	$2^{52.1}$	1.53 yrs
			<b>DL</b>	<b>2</b>	<b><math>2^{16}</math></b>	<b>0.66 ms</b>
Antenna	Key 1	Device 1	CA	11	$2^{102.1}$	-
			<b>DL</b>	<b>2</b>	<b><math>2^{16}</math></b>	<b>0.66 ms</b>
	Key 2	Device 2	CA	12	$2^{106.8}$	-
			<b>DL</b>	<b>2</b>	<b><math>2^{16}</math></b>	<b>0.66 ms</b>

For CA attack, positions of incorrect bytes are unknown  $\Rightarrow$  #Enum. =  $\binom{16}{k} 2^{8k}$

For DL attack, positions of incorrect bytes are known  $\Rightarrow$  #Enum. =  $2^{8k}$

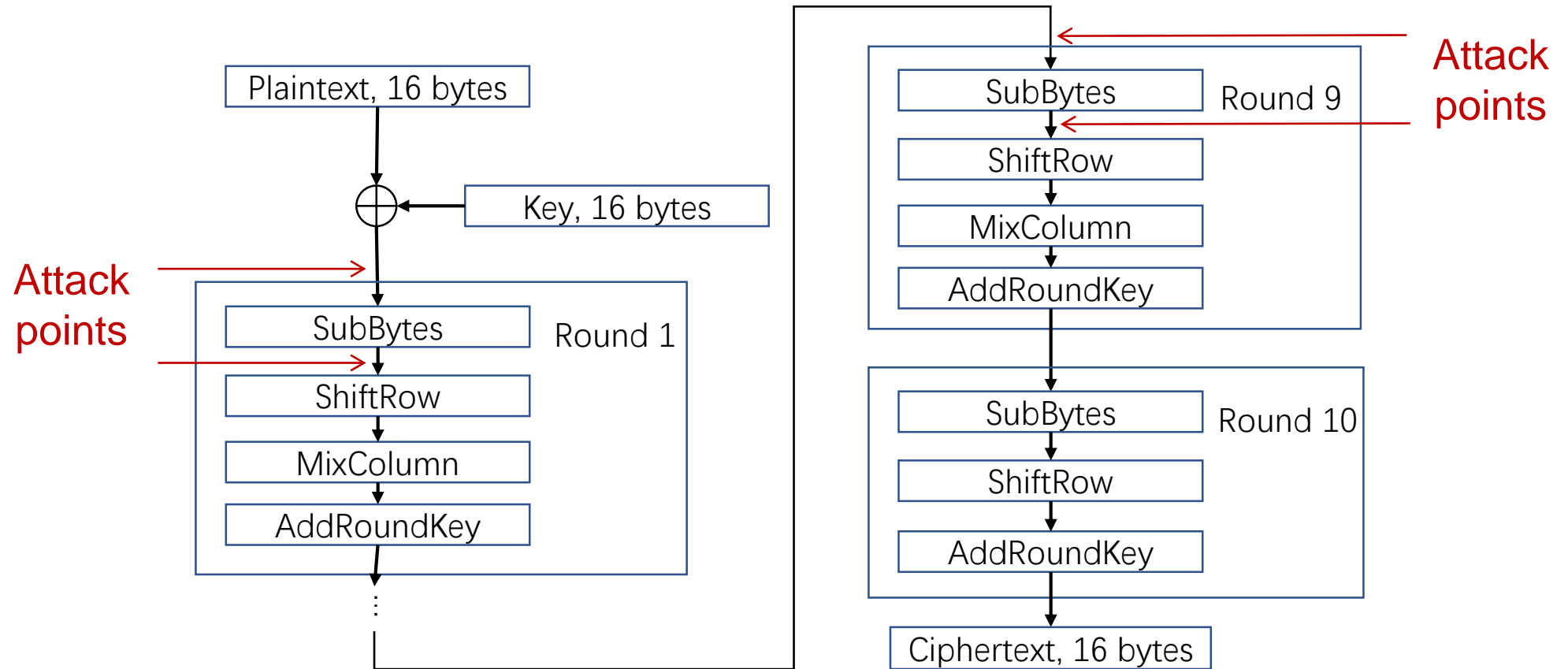


## Example 3: SAT-assisted attack on AES

- Software implementation of AES-128 in an 8-bit MCU
  - Uses power as a side channel
- Hybrid attack: deep-learning power analysis + SAT solving
- First study on real devices with cross-device evaluation
- Recovers full key from a single trace in 20 min with a high probability

*Solving AES-SAT Using Side-Channel Hints: A Practical Assessment*, E. Dubrova,  
IEEE International Symposium on Multiple-Valued Logic, June 5-6, 2025

# AES-128 algorithm



$9 \times 16 = 144$  S-Box I/Os in 9 rounds

# SAT background

- SAT problem: Find variable assignment satisfying CNF
  - solved by MiniSat solver
- AES encoded using 9,152 literals and 188,664 clauses
  - in DIMACS format typically used by SAT solvers
  - CNF generation tool <https://github.com/meelgroup/aes-cnf-gen/>
- Example: two-variable Boolean XOR  $a \oplus b = c$

a	b	c
0	0	0
0	1	1
1	0	1
1	1	0

$\neg a \wedge \neg b \wedge \neg c$   
 $\neg a \wedge b \wedge c$   
 $a \wedge \neg b \wedge c$   
 $a \wedge b \wedge \neg c$   
**CNF clauses**

p cnf 3 4  
 -1 -2 -3 0  
 -1 2 3 0  
 1 -2 3 0  
 1 2 -3 0  
 DIMACS format



## Related work

- SAT solvers have been used in cryptanalysis for finding preimages and collisions for hash functions, forging RSA signatures, identifying weak keys, etc.
  - Attacking non-reduced crypto algorithms is typically difficult
- SAT solvers have been applied in cold boot attacks for recovering secret keys from decayed RAM images of expanded round keys
  - Redundancy of the key scheduling algorithm is exploited
- Previous SCA employing SAT use simulated data and typically recover the key partially
- This work: real devices, cross-device evaluation, full key recovery

## Attack overview: Profiling stage

- Capture training traces from two profiling devices
- Train MLPs using S-Box I/Os as labels and combine in ensemble: 5 MLPs for inputs + 5 MLPs for outputs

Table I: Profiling stage duration

<b>Time to capture 10K traces</b>	<b># Devices</b>	<b>Total time</b>
1 hour	2	2 hours

<b>Time to train 1 MLP</b>	<b># MLPs</b>	<b>Total time</b>
3 hours	10	30 hours



## Attack overview: Attack stage

- Capture a single trace from each device under attack
  - Three devices with different aging or wear-out characteristics are evaluated
- Use MLP ensembles to predict S-box I/Os
  - Add top  $n$  predictions as unitary clauses to AES CNF
- Use MiniSat to solve the CNF
  - Full 128-bit key is recovered



# SAT modeling of AES

- *SubBytes*: Boolean minimization of 8-bit S-box
  - Permutation of type  $\{0,1\}^8 \rightarrow \{0,1\}^8$
  - Espresso can find a minimal sum-of-product form
- *ShiftRows* & *MixColumns*: linear mappings over  $GF(2^8)$ 
  - Auxiliary variables are used to split into smaller functions
- *AddRoundKey*: bitwise XOR of the AES state and round key
- *Key schedule algorithm*: derives 128-bit round keys from the original 128-bit key  $K$ 
  - Involves S-box substitutions, one-byte left circular shifts, XOR operations, and round constants



# Experimental setup

- Target device
  - ATxmega128D4 MCU running C AES-128 in Electronic Codebook (ECB) mode
- Equipment for trace capture:
  - ChipWhisperer-Lite
- Profiling set:
  - 20 K traces captured from two profiling devices
  - Expanded to  $20\text{ K} \times 9\text{ rounds} \times 16\text{ Sboxes} = 2.88\text{ M traces}$

## Results: Power analysis only

- Full key can be recovered from a single trace:
  - with probability 14.4 % without enumeration
  - with probability 79 % after 26.1 hrs of enumeration

Table II: Mean probability to recover  $n$  S-box outputs in round R1

$n$	Probability	Enumeration time
12	0.885	1.89 years
13	0.790	26.1 hours
14	0.626	1.31 min
15	0.383	0.04 sec
16	0.144	0

## Results: Power analysis + SAT

- Full key can be recovered from a single trace:
  - in 20.2 min with 96.7% probability if SAT timeout is 1 min
  - In 8.9 hrs with 100% probability if SAT timeout is 1 hour

Table III: Mean probability to recover the full key using S-Box I/Os

Rounds	% Recovered keys	Time to recover full key
R1	60%	53.3 sec
R1-R2	80%	99.1 sec
R1-R5	90%	17.6 min
R1-R9	96.7%	20.2 min



## Key takeaways

- Combing Boolean reasoning methods with side-channel analysis may dramatically reduce key search time
  - This strengthens side-channel analysis methods
  - But also opens new opportunities for attackers
- Further research on combined attacks is needed to understand their possibilities and limitations
  - Important for designing countermeasures



# Using side-channel analysis methods for run-time side-channel monitoring

Several desirable properties:

- Non-invasive monitoring of device, can be retrofitted
- Monitor does not need to know internal state of device - only what is “normal operation”
- Can be used to prove a certain process has been performed on a device
- Can complement other security mechanisms as trusted boot and malware detection
  - *Machine-Learning Assisted Side-Channel Analysis for Software Integrity Verification*, Lindskog, N., Englund, H., Sternby, J., Dubrova, E., IEEE European Test Symposium, May 26-30, 2025



# What is possible today?

Given access to a physical side-channel:

- Determine application from pre-determined set of applications
- Detect counterfeit hardware and Trojans

Given reasonable determinism in processor and some oversampling:

- Detect malicious alterations and failures in real-time
- Detect non-conforming input to ML models
- Verify proof-of-work

Given simpler processors and severe oversampling:

- Disassemble instructions and registers



SXQgaXMgcG9zc2libGUgdG8g  
 aW52ZW50IGEgc2luZ2xlIG1h  
 Y2hpbmUgd2hpY2ggY2FuIGJl  
 IHVzZWQgdG8gY29tcHV0ZSBh  
 bnkgY29tcHV0YWJzZSBzZXFl  
 ZW5jZS4gSWYgdGhpcyBtYWNo  
 aW5lIHRlc3wuc3pZWQg  
 d210aShIRhEglGgdGhl  
 IGJlZmlyeW9kaGlj  
**CDIS**  
 aCBpcyB3cm10dGVuIHRoZSBT  
 LkQgb2Ygc29tZSBjb21wdXRp  
 bmcgbWFjaGluZSBnLlCB0aG  
 VuIFUgd21sbCBjb21wdX  
 RlIHRoZSBzYW1lIH  
 NlcXVlbnNlIG  
 FzIEOuCG  
 ==



Myndigheten för  
 samhällsskydd  
 och beredskap

# Thank you!

TECOSA

VINNOVA