# Mid-Size Primes for Symmetric Cryptography with Strong Embedded Security

(*Low-Noise Masking and Hard Physical Learning Problems*)

*François-Xavier Standaert*

UCLouvain, Crypto Group

**STAP 2023**

**Lyon, France, April 23, 2023**

# Outline

- Side-channel analysis & the need of masking
- Boolean masking and the need of noise
- Prime masking and design challenges

- Fresh re-keying & basic models
- Hard physical learning problems

- General conclusions for symmetric crypto

# Outline

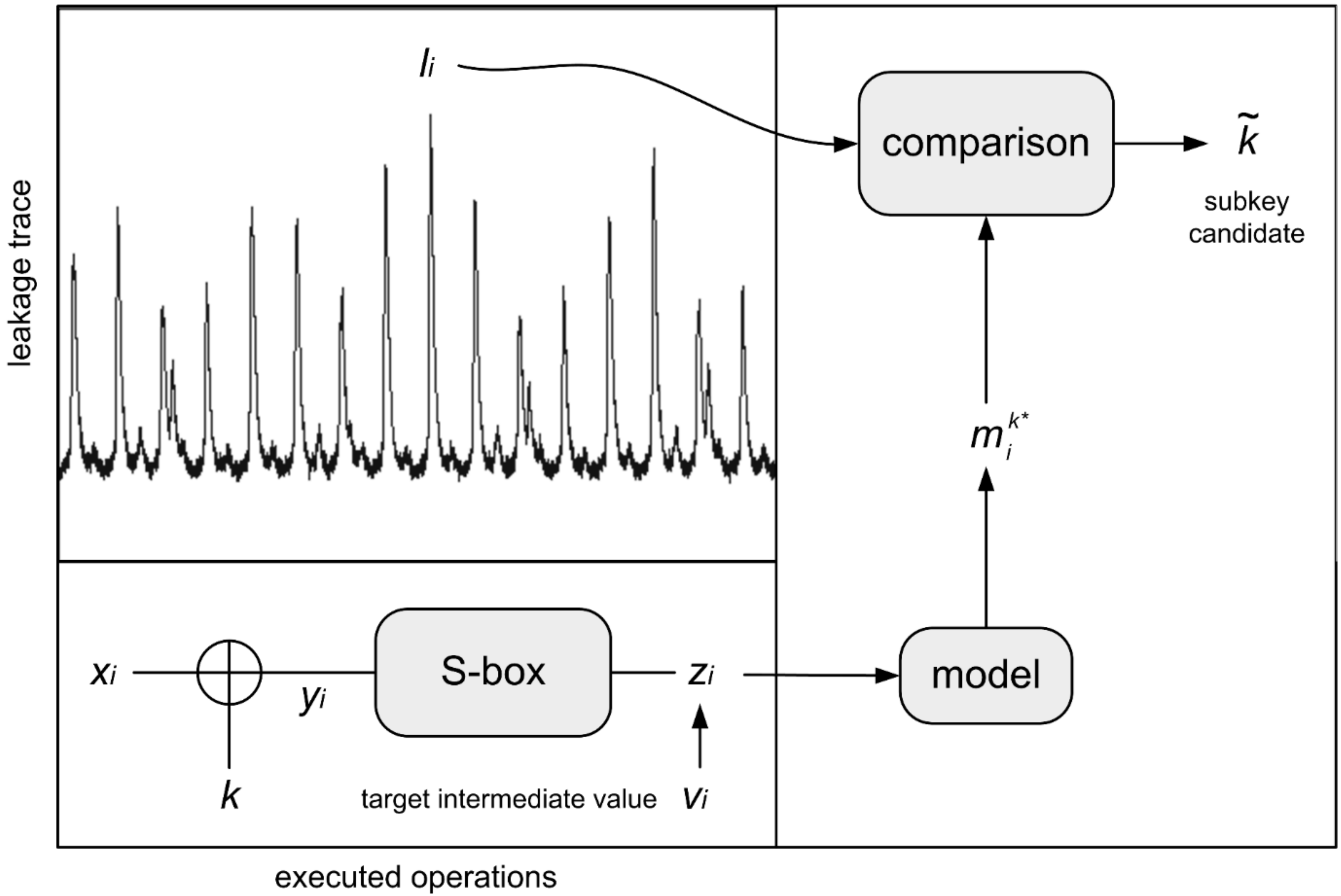
- **Side-channel analysis & the need of masking**
- Boolean masking and the need of noise
- Prime masking and design challenges

- Fresh re-keying & basic models
- Hard physical learning problems

- General conclusions for symmetric crypto

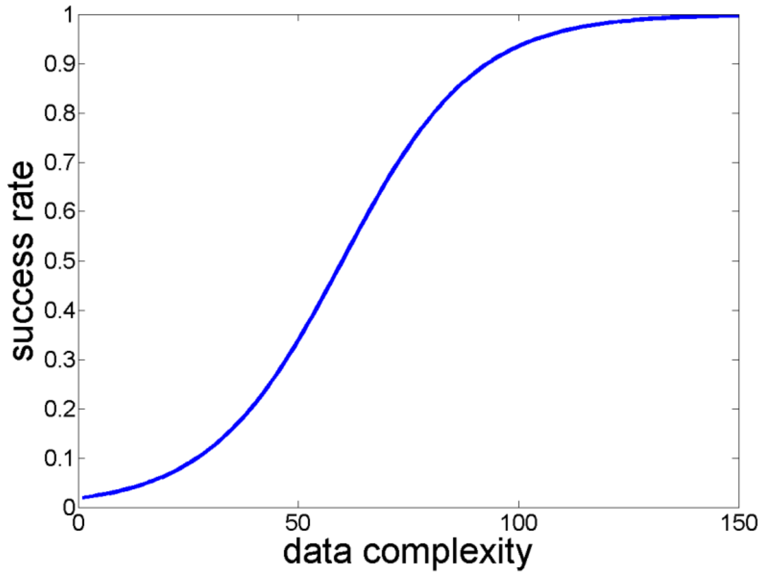
# Power Analysis Analysis [KJJ99]





- Differential Power Analysis (many-traces attacks)

$$\Pr\left[A_{\mathrm{KR}}\left(x_1, \boldsymbol{L}(x_1, K), \ldots, x_q, \boldsymbol{L}(x_q, K)\right) \rightarrow K | K \leftarrow \$\right] \approx 2^{-128+q\cdot\lambda}$$



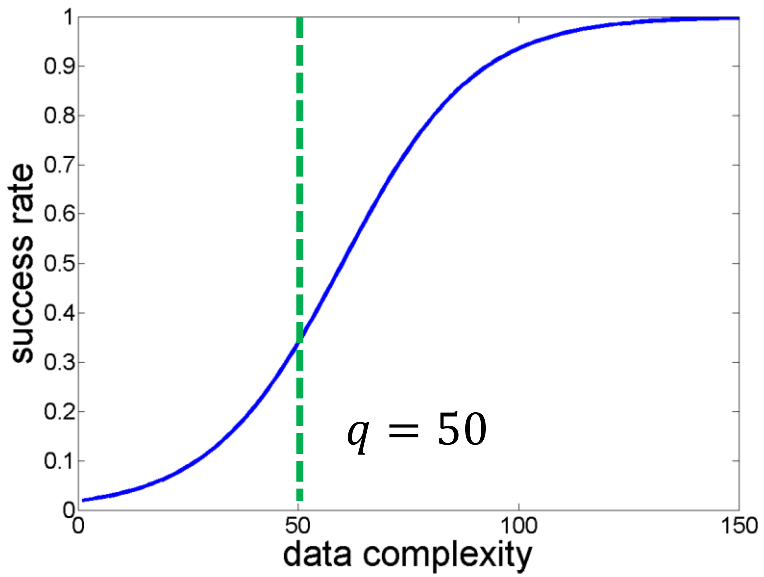

$$\lambda \approx \mathrm{MI}(Z; \boldsymbol{L})$$

# DPA vs. SPA taxonomy



- Differential Power Analysis (many-traces attacks)

$$\Pr\left[A_{\mathrm{KR}}\left(x_1, \boldsymbol{L}(x_1, K), \ldots, x_q, \boldsymbol{L}(x_q, K)\right) \rightarrow K \mid K \leftarrow \$\right] \approx 2^{-128+q\cdot\lambda}$$

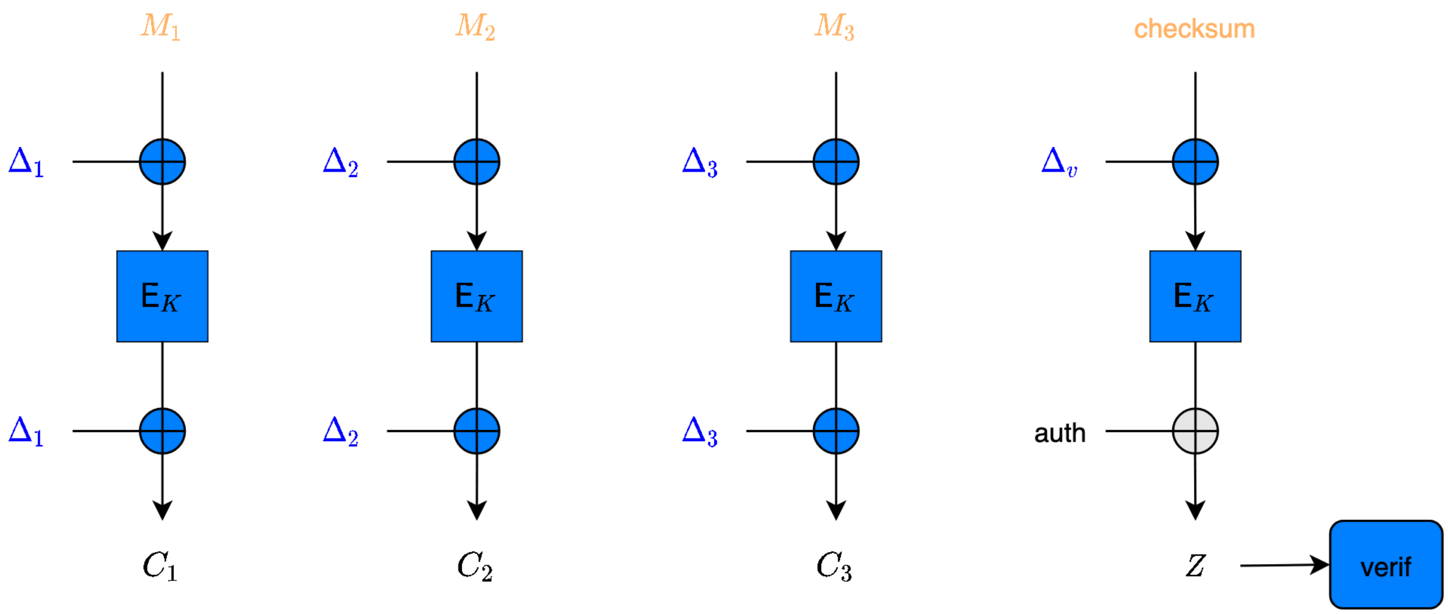



$$\lambda \approx \mathrm{MI}(Z; \boldsymbol{L})$$

- Simple Power Analysis (few-traces attacks)

# DPA security is needed

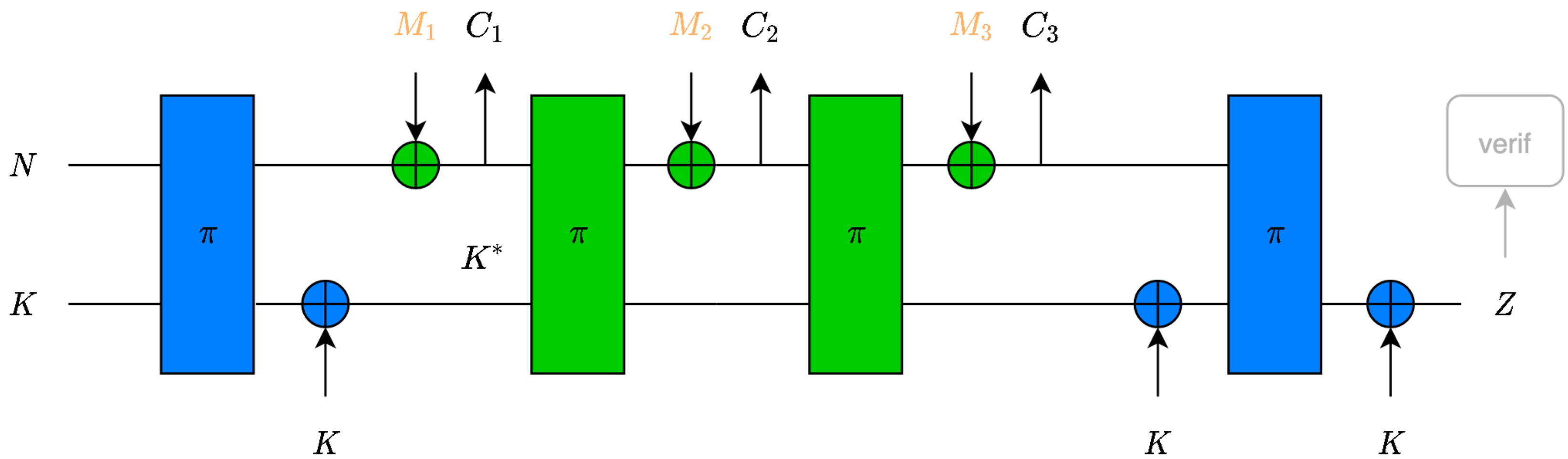
- Everywhere for standard modes of operation




OCB

# DPA security is needed

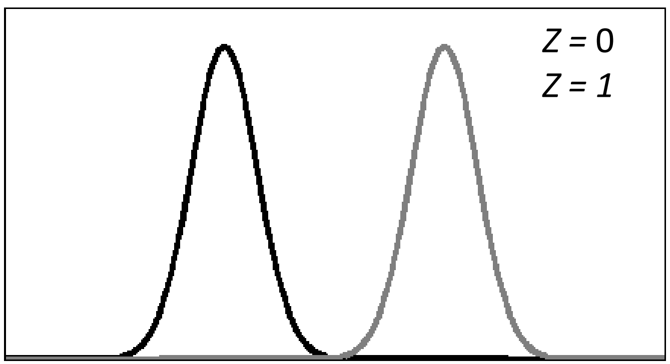
- Everywhere for standard modes of operation

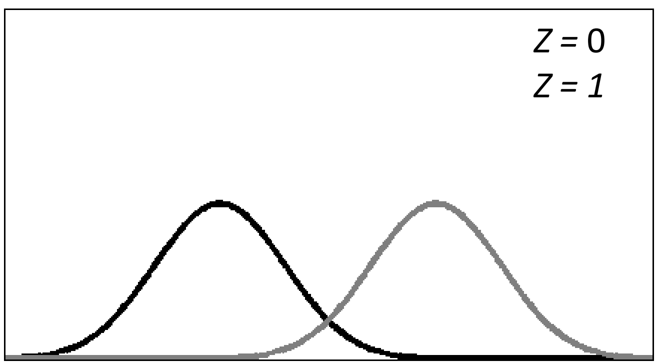



Ascon

- Mildly for leakage-resistant modes of operation
  - ∝ requirements (e.g., integrity, confidentiality)

# Noise is not enough for DPA security 


- Additive noise ≈ cost × 2 ⇒ security × 2
  ⇒ not a good (crypto) security parameter
- ≈ same holds for all hardware countermeasures

# Outline

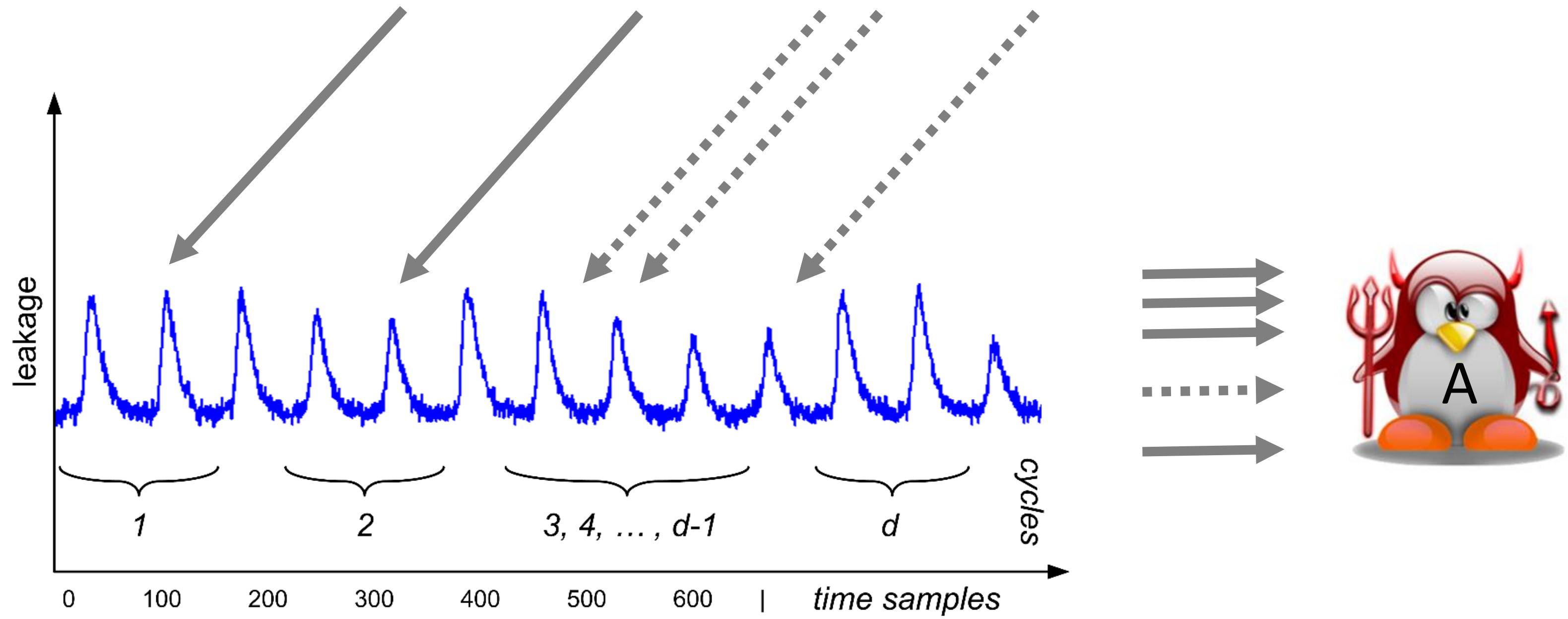
- Side-channel analysis & the need of masking
- **Boolean masking and the need of noise**
- Prime masking and design challenges

- Fresh re-keying & basic models
- Hard physical learning problems

- General conclusions for symmetric crypto

- Private circuits / probing security [ISW03]

$$z = z_1 \oplus z_2 \oplus \cdots \oplus z_{d-1} \oplus z_d$$



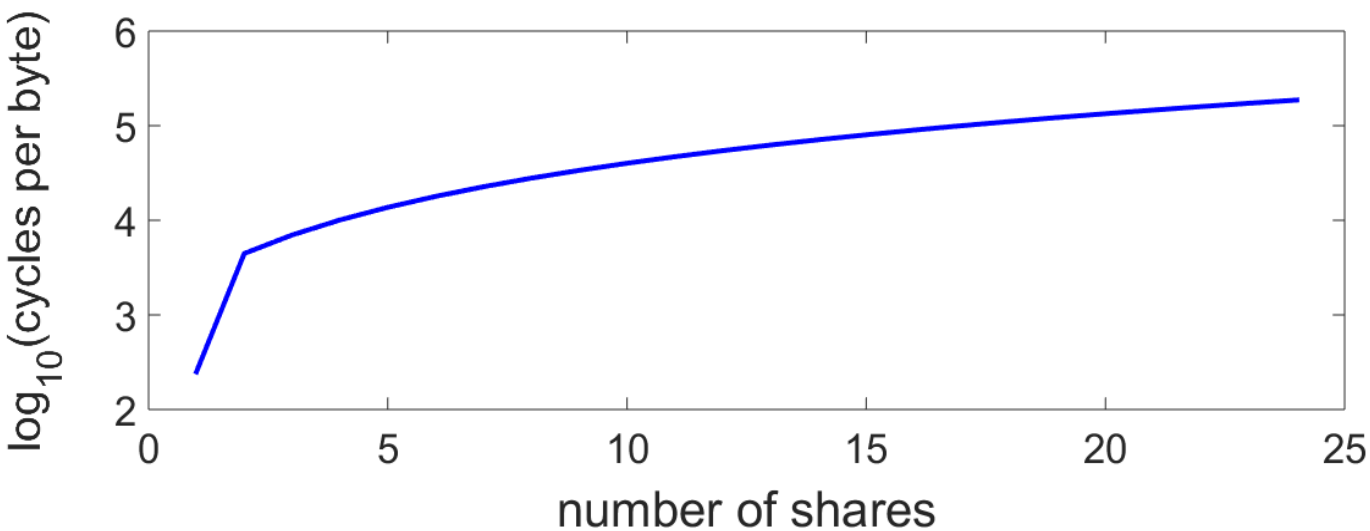

serial implementation.

- Goal: bounded information $\mathrm{MI}(Z;\boldsymbol{L})<\mathrm{MI}(Z_i;\boldsymbol{L}_{Z_i})^d$

# Masking is expensive (e.g., ARM Cortex-M4) 

- Multiplications ≈ quadratic overheads

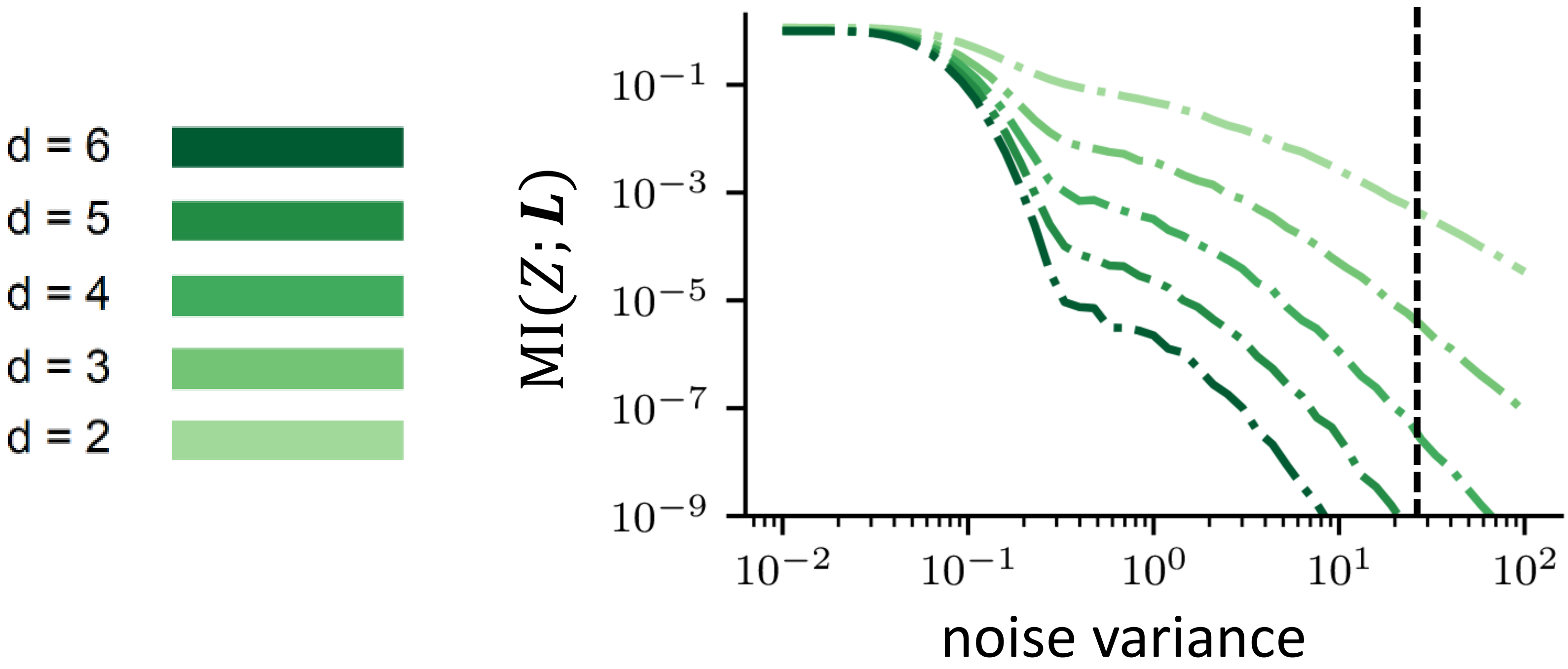
$$\begin{bmatrix} a_1b_1 & a_1b_2 & a_1b_3 \\ a_2b_1 & a_2b_2 & a_2b_3 \\ a_3b_1 & a_3b_2 & a_3b_3 \end{bmatrix} + \begin{bmatrix} 0 & r_1 & r_2 \\ -r_1 & 0 & r_3 \\ -r_2 & -r_3 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix}$$



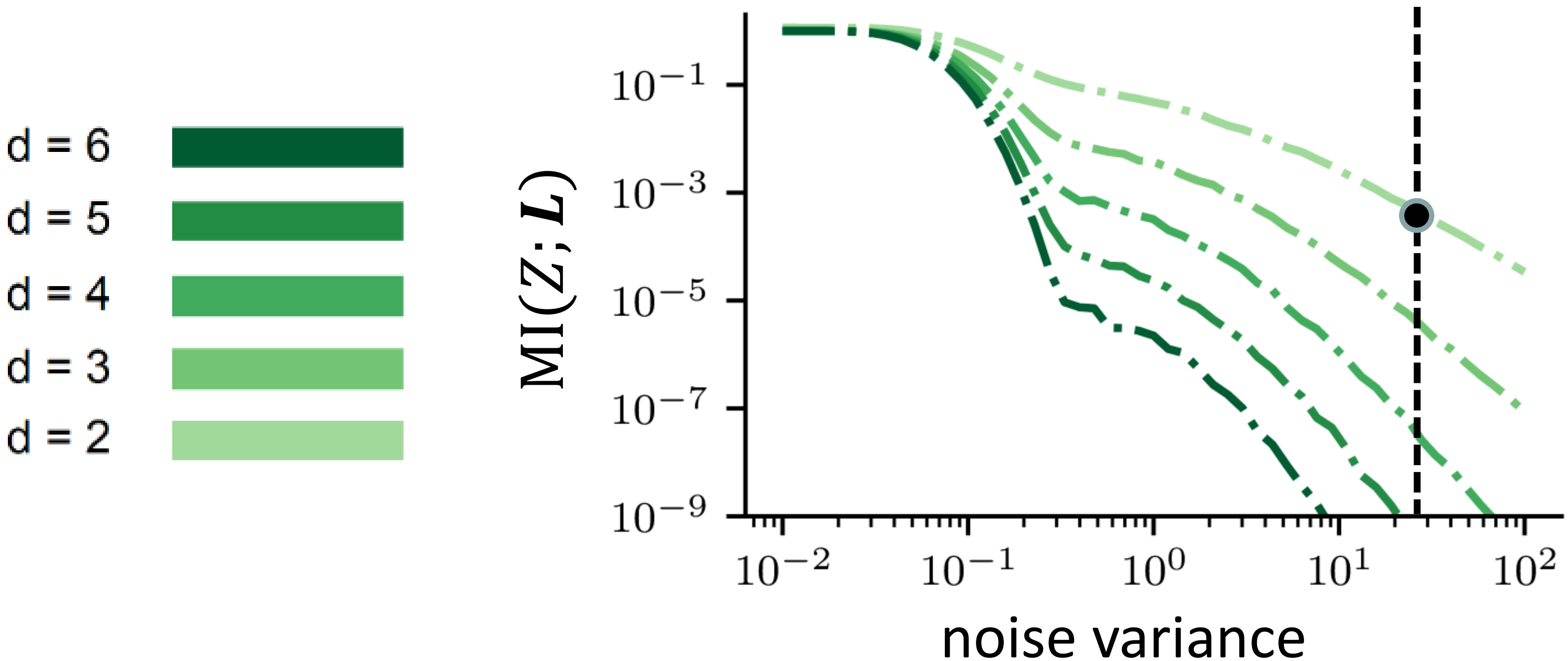

⇒ Current approach: bitslice ciphers + noise

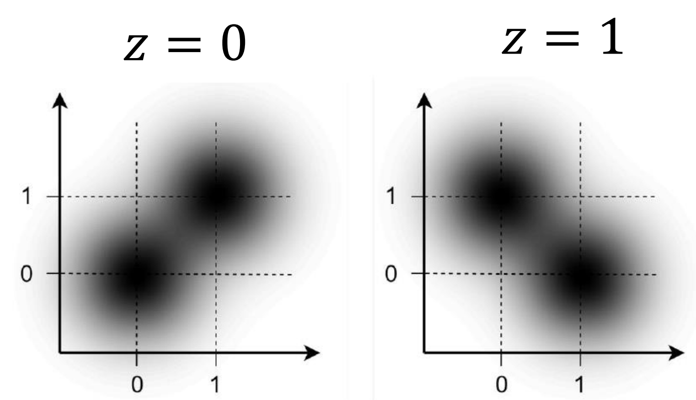
# Boolean masking with noise: OK



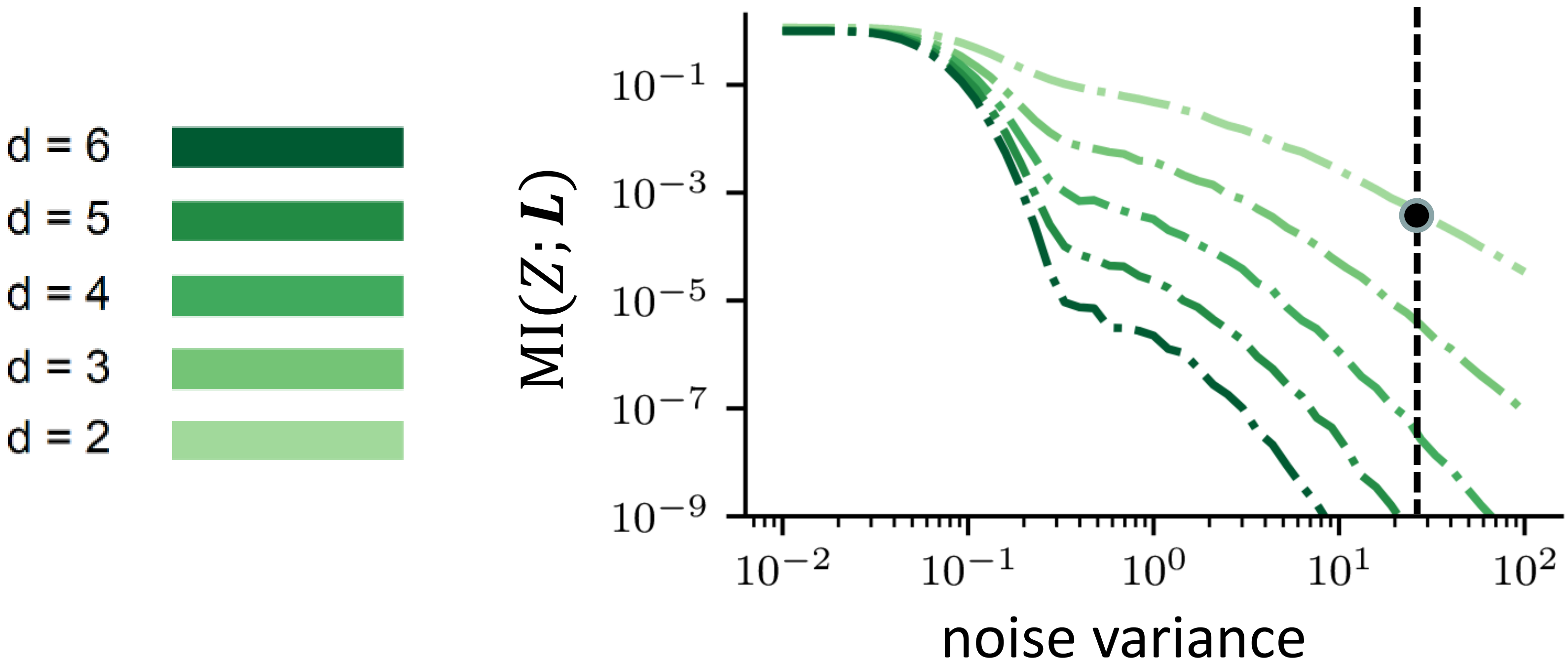


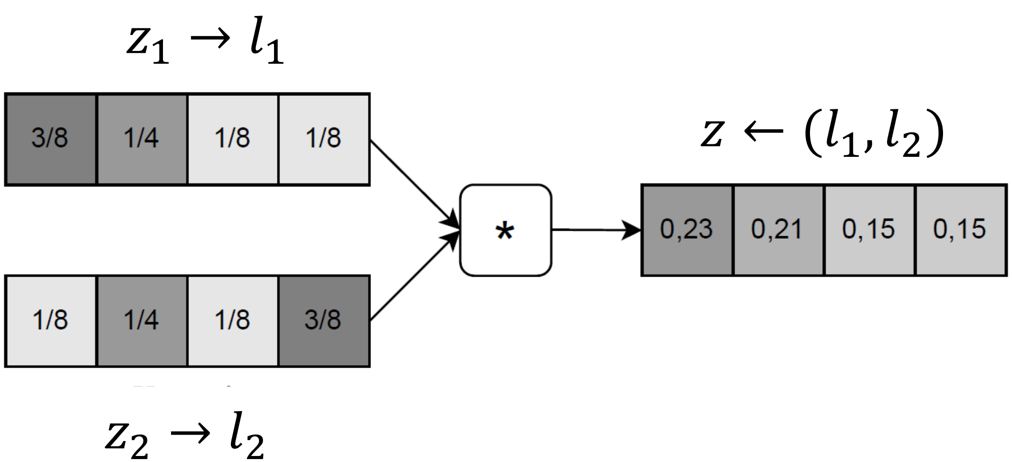
# Boolean masking with noise: OK

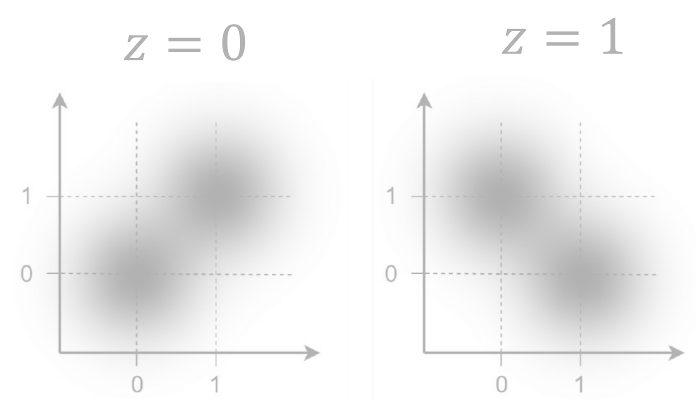


# Boolean masking with noise: OK

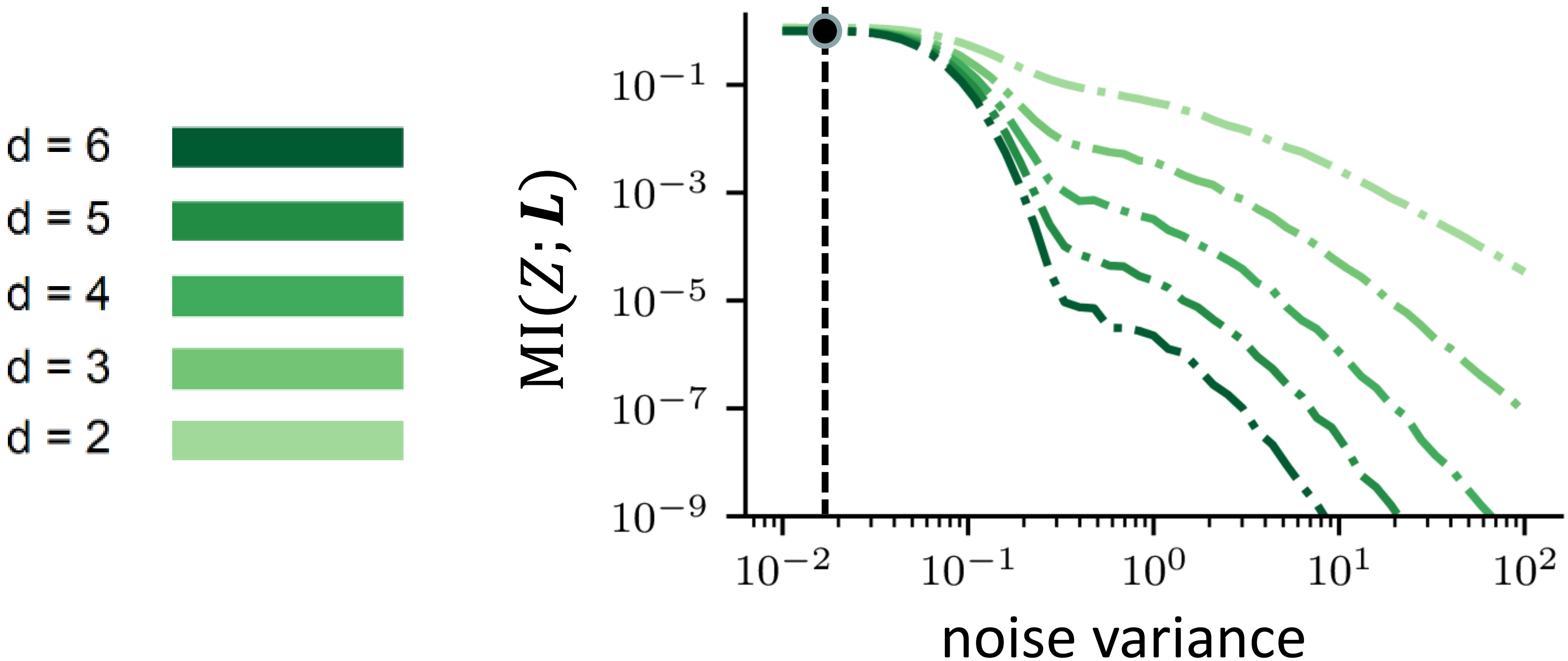


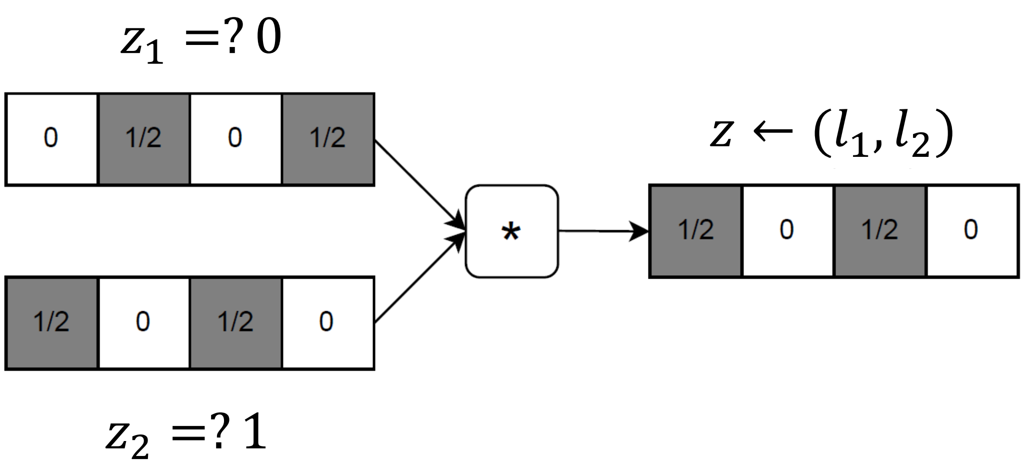



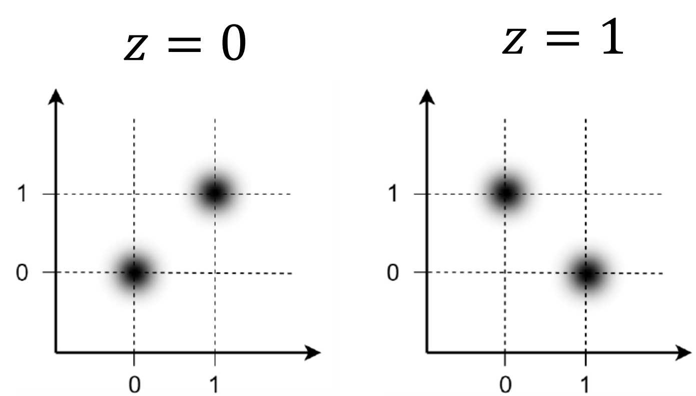



# Boolean masking without noise: KO



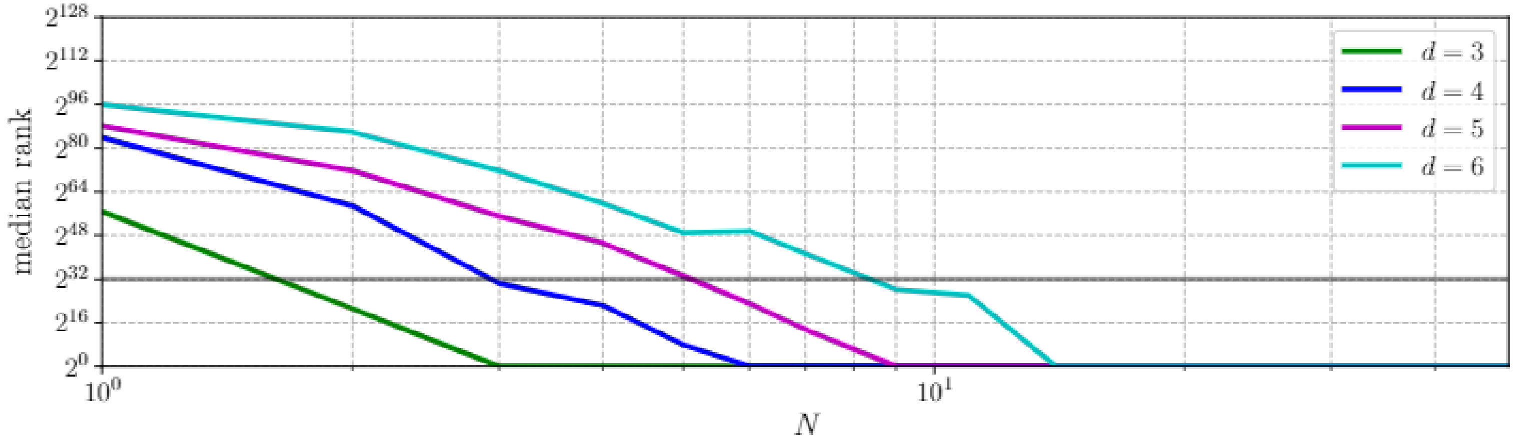


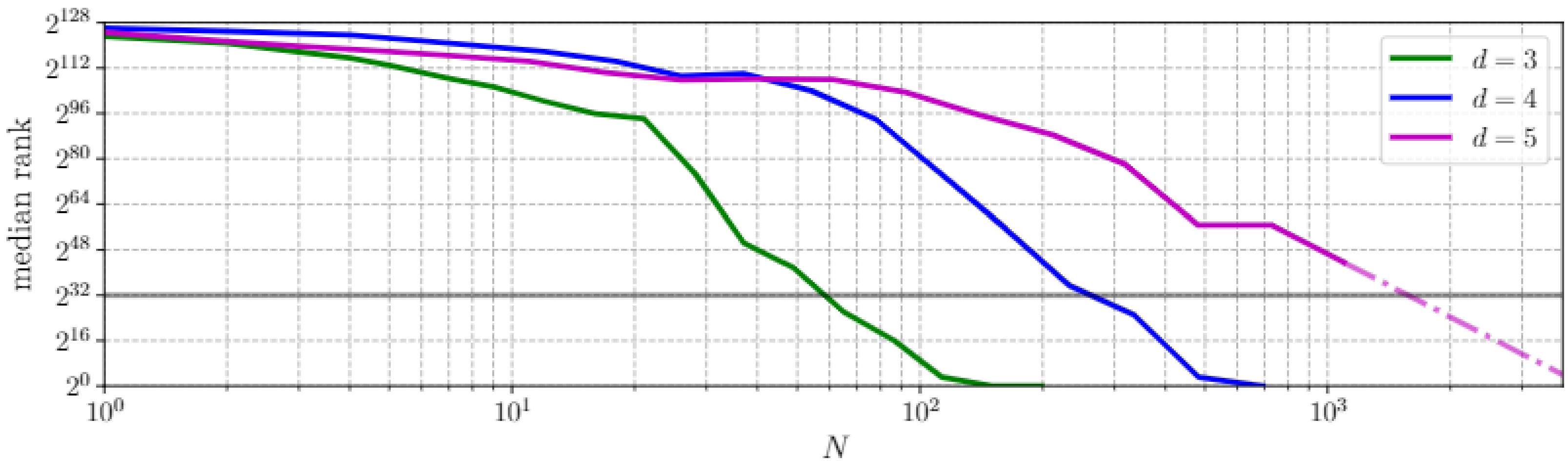



# Noise issue in practice

- Masked bitslice AES implementation
  - ARM Cortex-M0




  - ARM Cortex-M3



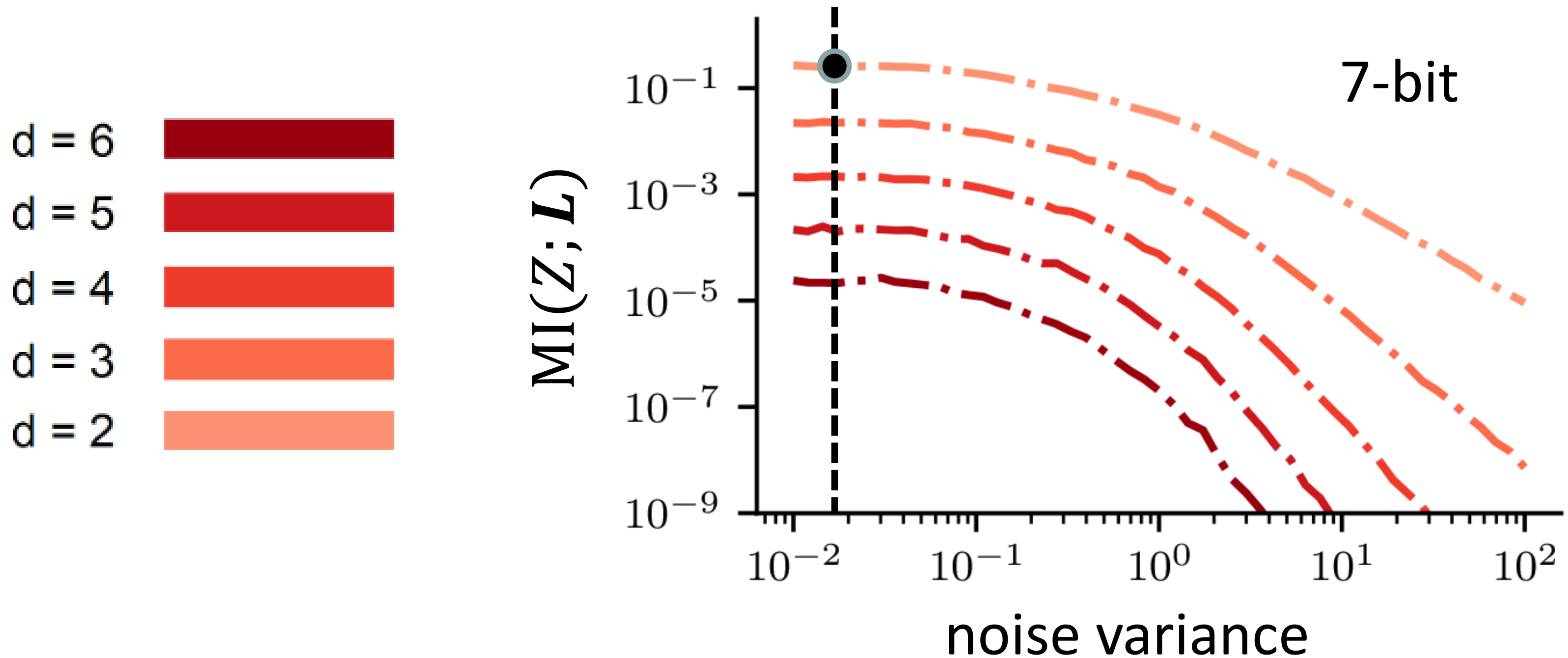
# Outline

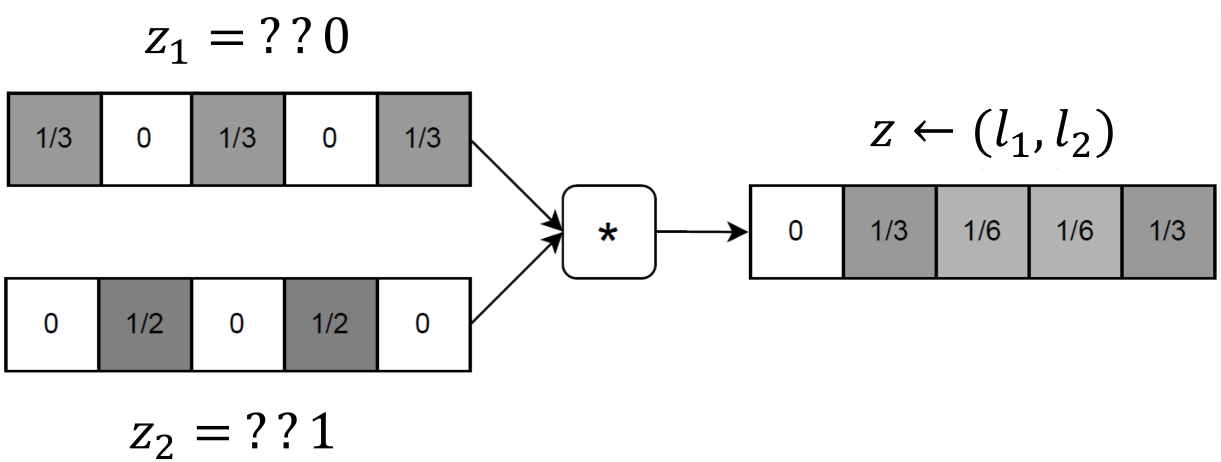
- Side-channel analysis & the need of masking
- Boolean masking and the need of noise
- **Prime masking and design challenges**

- Fresh re-keying & basic models
- Hard physical learning problems

- General conclusions for symmetric crypto

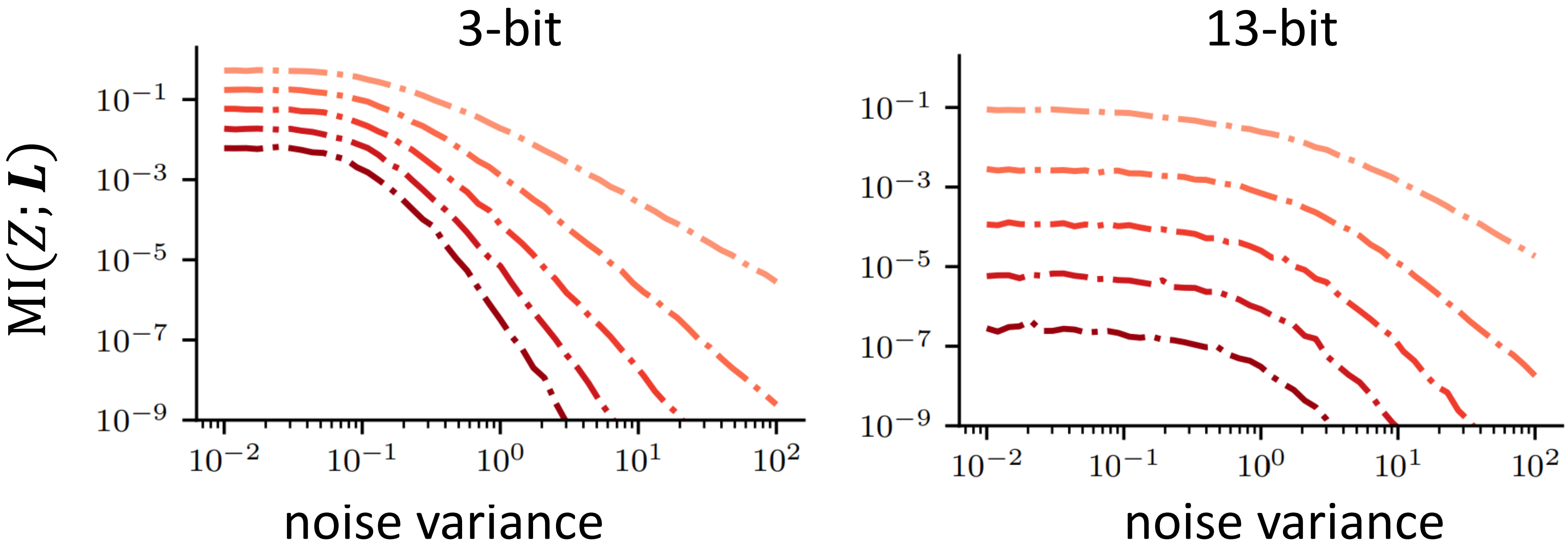
# Prime masking with noise: OK



# Cost vs. security tradeoff (I)

- Increasing the field size (sometimes) helps
  - Example for Hamming weight leakages
  - And Mersenne primes for efficiency



# Cost vs. security tradeoff (II)



- Prime computations overheads can be mild
  - In software and hardware implementations

Cycle Counts (ARM Cortex-M3):

| d | Field Arith. $\mathbb{F}_{2^n}$ | Field Arith. $\mathbb{F}_{2^n-1}$ | log/alog $\mathbb{F}_{2^n}$ | log/alog $\mathbb{F}_{2^n-1}$ |
|---|---|---|---|---|
| 2 | 1321 | 189 | 232 | 282 |
| 3 | 2902 | 334 | 448 | 535 |
| 4 | 5213 | 600 | 800 | 912 |
| 5 | 8255 | 1125 | 1340 | 1581 |
| 6 | 12038 | 1692 | 1988 | 2283 |

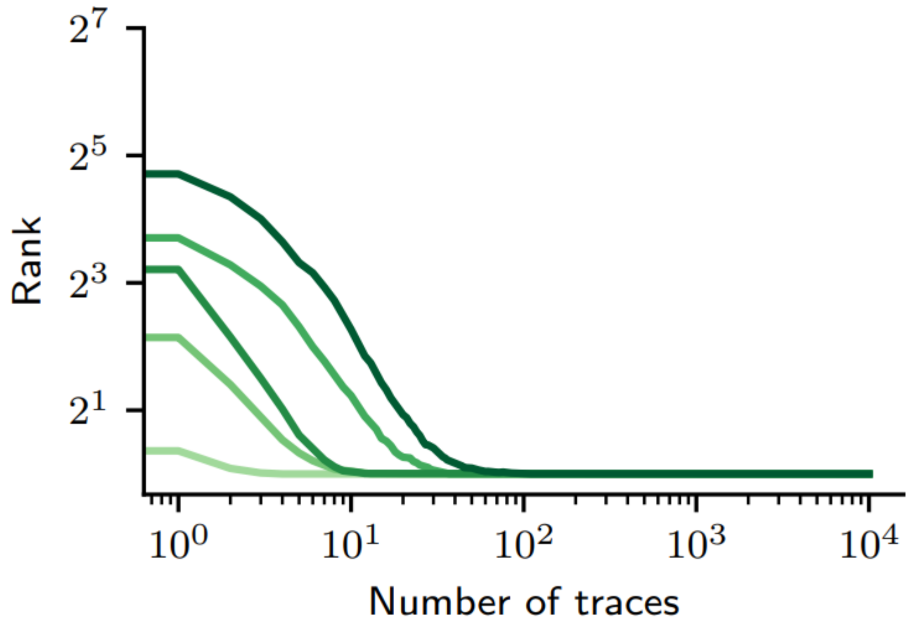
Resource Utilization (Xilinx Spartan-6):

| d | Binary Field $\mathbb{F}_{2^n}$ LUTs | Slic. | DSPs | Prime Field $\mathbb{F}_{2^n-1}$ LUTs | Slic. | DSPs |
|---|---|---|---|---|---|---|
| 2 | 26 | 15 | 0 | 20 | 11 | 1 |
| 3 | 126 | 77 | 0 | 131 | 70 | 4 |
| 4 | 285 | 161 | 0 | 348 | 160 | 9 |
| 5 | 539 | 293 | 0 | 710 | 306 | 16 |
| 6 | 848 | 486 | 0 | 1096 | 515 | 25 |

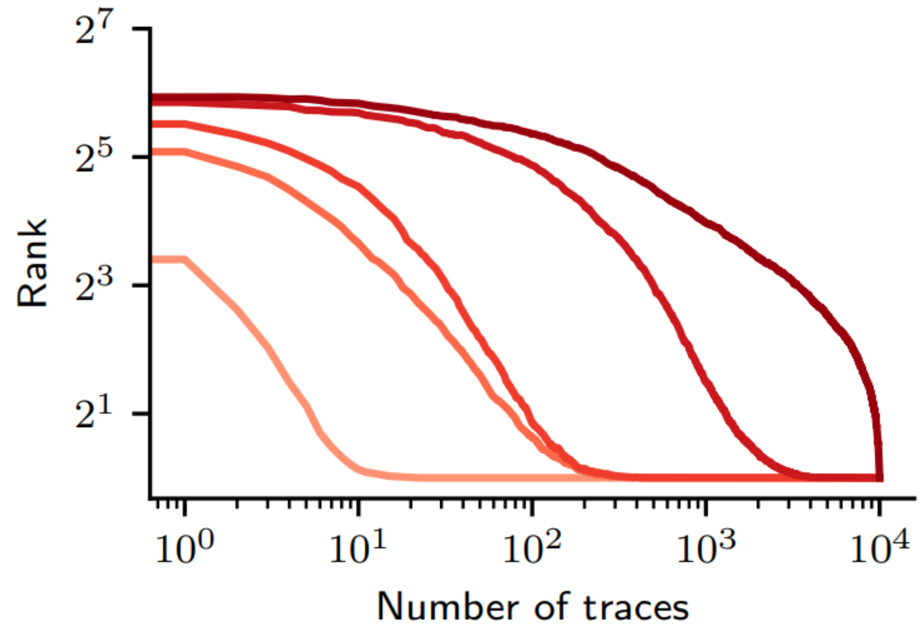
- Especially if efficient arithmetic operations (in SW) and DSP blocks (in HW) are available

- Theoretical gains are observed in the field
  - Example of attacks against an ARM Cortex-M3

$x^5 + 2$ in $\mathbb{F}_{2^7}$




$x^5 + 2$ in $\mathbb{F}_{2^7-1}$




- And seem to increase with the # of shares

# Conclusions for part #1



- Prime field masking can significantly increase side-channel security in low-noise contexts
- At the cost of manageable overheads
- Gains are maintained in high-noise context!

⇒ Next: show cost vs. security gains for full ciphers

# Conclusions for part #1 

- Prime field masking can significantly increase side-channel security in low-noise contexts
- At the cost of manageable overheads
- Gains are maintained in high-noise context!

⇒ Next: show cost vs. security gains for full ciphers

- This requires ciphers adapted to prime masking
  - $2^7 - 1$ for hardware, $2^{31} - 1$ for software ?
  - Taking advantage of secure squaring (CHES 2023)
- To be compared with the best bitslice ciphers

# Conclusions for part #1



- Prime field masking can significantly increase side-channel security in low-noise contexts
- At the cost of manageable overheads
- Gains are maintained in high-noise context!

⇒ Next: show cost vs. security gains for full ciphers

- This requires ciphers adapted to prime masking
  - $2^7 - 1$ for hardware, $2^{31} - 1$ for software ?
  - Taking advantage of secure squaring (CHES 2023)
- To be compared with the best bitslice ciphers

- More details this Monday at Eurocrypt 2023

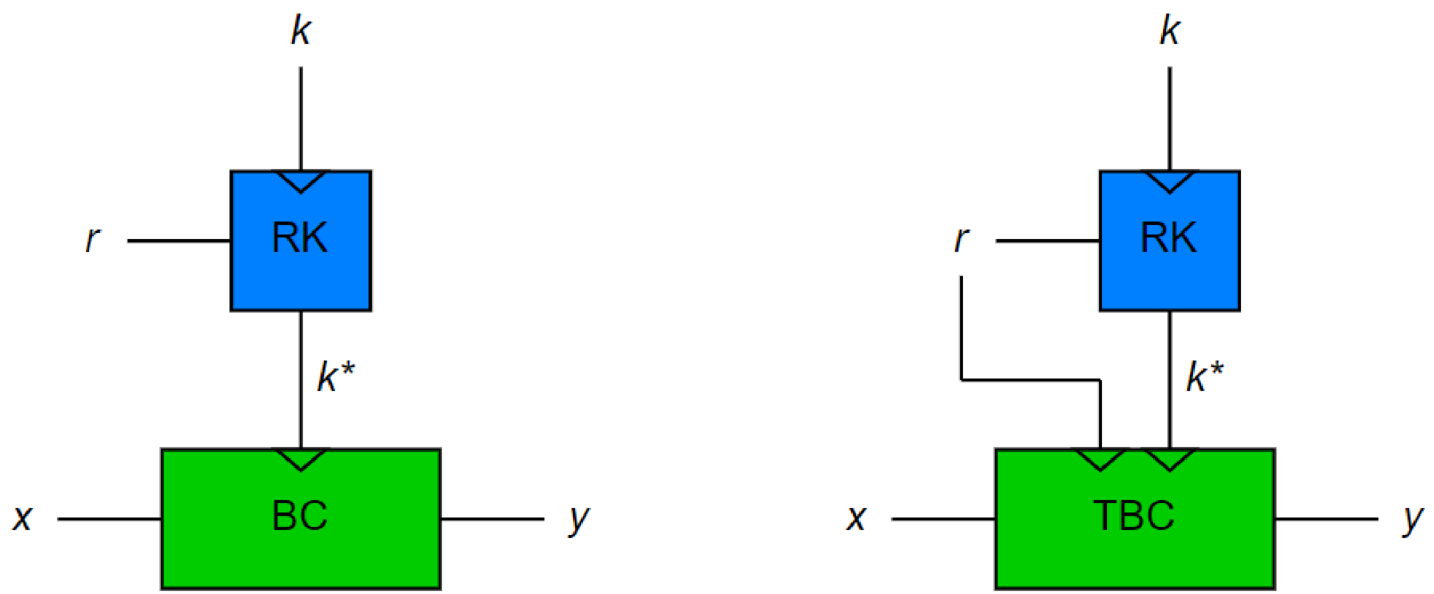
# Outline

- Side-channel analysis & the need of masking
- Boolean masking and the need of noise
- Prime masking and design challenges

- **Fresh re-keying & basic models**
- Hard physical learning problems

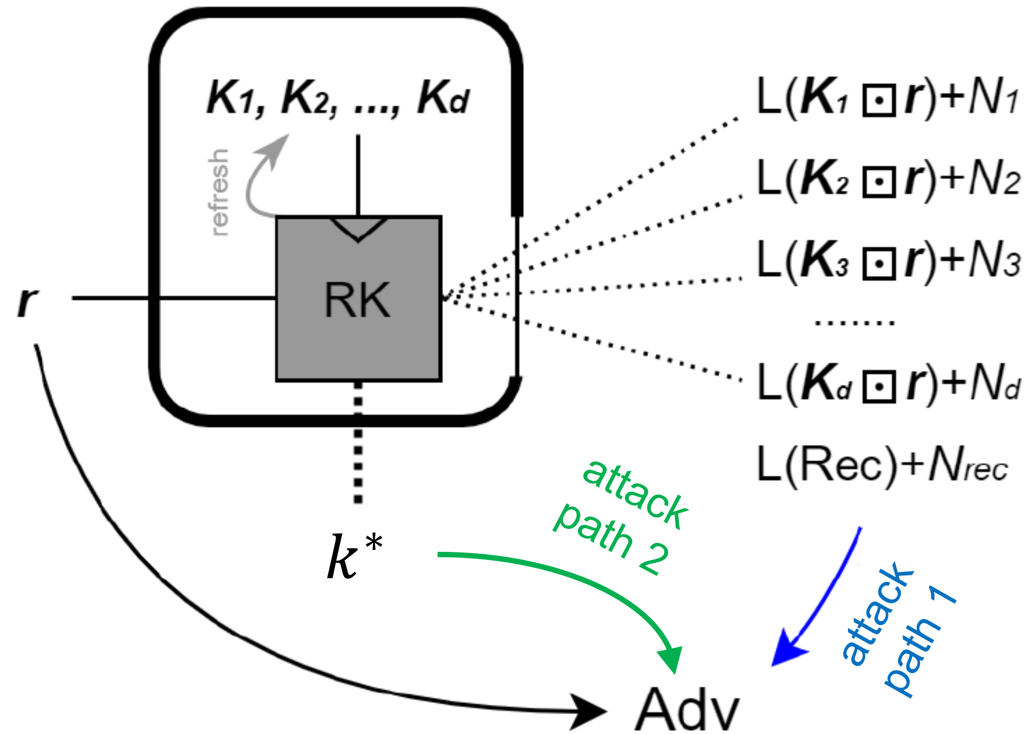
- General conclusions for symmetric crypto

# Fresh re-keying principle



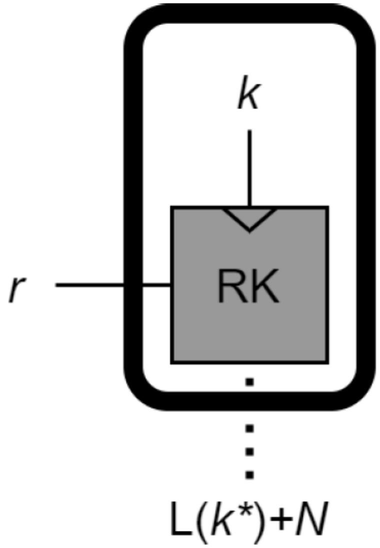

- Find a re-keying function that is easy to protect against DPA (e.g., key homomorphic, …)
  - Main question: how to formalize RK security?

# Security requirements




- Avoiding attack path #1 is well understood
- Avoiding attack path #2 much less ($\neq$ models)

# Model 1: Medwed et al.

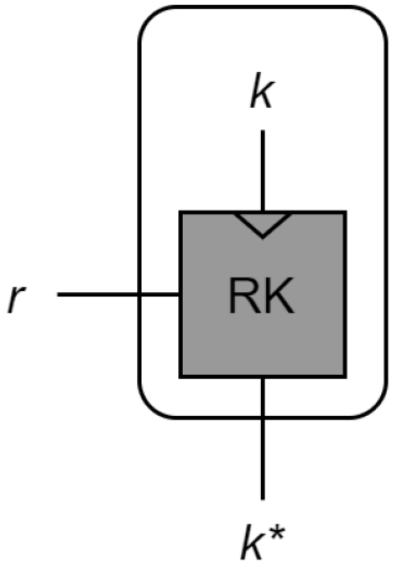



- Noisy leakages
- Proposed instance
  - $k^* = r \cdot k$ over $\mathbb{F}_{2^\kappa}$
  - Key homomorphic
- Efficient but insecure w/o noise

- Somewhat similar to Boolean masking
  - LSB of Hamming weight leakage is linear in $\mathbb{F}_{2^\kappa}$

# Model 2: Dziembowski et al.






- Unbounded leakages on $k^*$
- Proposed instance (wPRF)
  - $k^* = \lfloor \langle \boldsymbol{r}, \boldsymbol{k} \rangle \rceil_p$, with $\boldsymbol{k}, \boldsymbol{r} \in \mathbb{Z}_{2q}^n$
  - Nearly key-homomorphic

$\Rightarrow$ Needs $\log(d)$ bits of error correction

- Very large key requirements
  - Poor performances in software & hardware

# Outline

- Side-channel analysis & the need of masking
- Boolean masking and the need of noise
- Prime masking and design challenges

- Fresh re-keying & basic models
- **Hard physical learning problems**

- General conclusions for symmetric crypto

# Model 3: Duval et al.

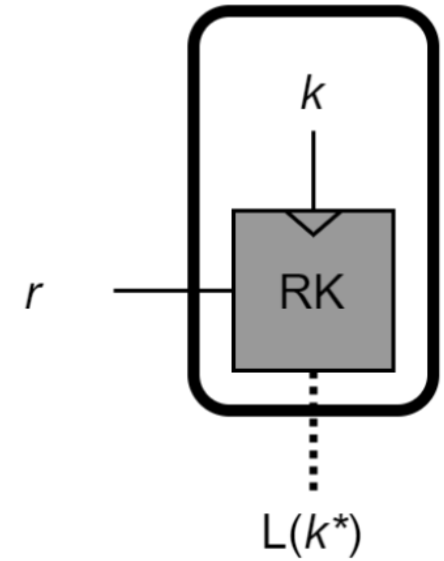


- Noise-free (compressive) leakages
- Similar to “crypto dark matter”
  - $\mathrm{F}_{\boldsymbol{K}}(\boldsymbol{r}) = \mathrm{map}(\boldsymbol{r} \cdot \boldsymbol{K})$

≈ security by combining different fields

- But assumes a physical mapping L

  ⇒ Crypto-physical dark matter

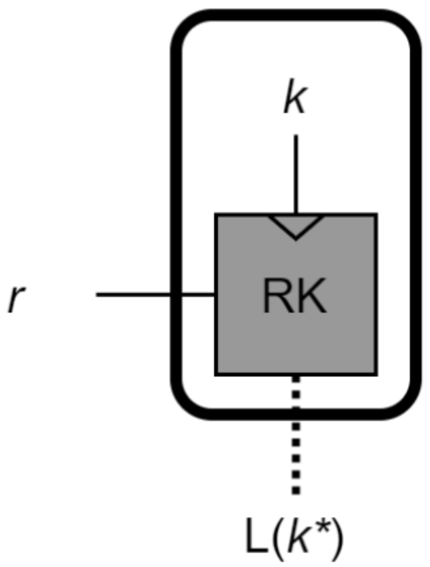


- Noise-free (compressive) leakages
- Similar to "crypto dark matter"
  - $\mathrm{F}_{\boldsymbol{K}}(\boldsymbol{r}) = \mathrm{map}(\boldsymbol{r} \cdot \boldsymbol{K})$

≈ security by combining different fields

- But assumes a physical mapping L

  ⇒ Crypto-physical dark matter




- Interest for re-keying: L never has to be computed explicitly by the leaking device (and therefore masked), the physics does it
- Challenge: L is not controlled by the designer

# Learning with Physical Rounding (LWPR) 

- Adv. gets samples $(\boldsymbol{r}, \mathsf{L}(\boldsymbol{K} \cdot (\boldsymbol{r}, \mathbf{1})))$ with $\boldsymbol{r} \in \mathbb{F}_p^n$ and $\boldsymbol{K} \in \mathbb{F}_p^{m\times(n+1)}$ and tries to recover $\boldsymbol{K}$

- Requires an embedding $\mathrm{g}\colon \mathbb{F}_p \to \{0,1\}^{\lceil \log(p) \rceil}$
- And a physical assumption on the mapping L

- Adv. gets samples $(\boldsymbol{r}, \mathrm{L}(\boldsymbol{K} \cdot (\boldsymbol{r}, \mathbf{1})))$ with $\boldsymbol{r} \in \mathbb{F}_p^n$ and $\boldsymbol{K} \in \mathbb{F}_p^{m \times (n+1)}$ and tries to recover $\boldsymbol{K}$

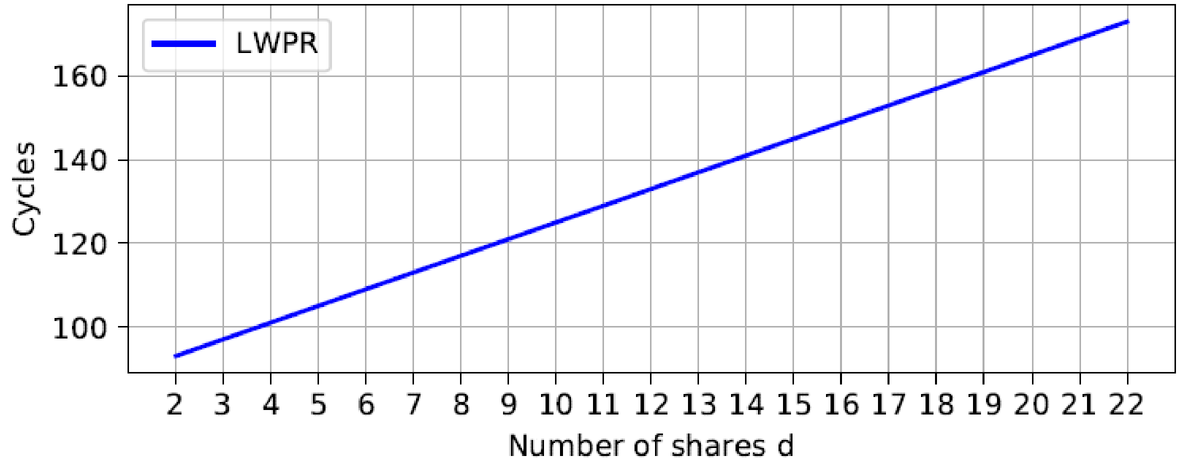
- Requires an embedding g: $\mathbb{F}_p \to \{0,1\}^{\lceil \log(p) \rceil}$
- And a physical assumption on the mapping L

- CHES 2021: Hamming weight (HW) assumption
  - First instance: $m = 4, n = 4, p = 2^{31} - 1$
  - Parallel implem.: if $\boldsymbol{k}_{\boldsymbol{i}}^* = \boldsymbol{K} \cdot (\boldsymbol{r}, \mathbf{1})$, adversary gets HW(g($\boldsymbol{k}_{\mathbf{1}}^*$))+HW(g($\boldsymbol{k}_{\mathbf{2}}^*$))+HW(g($\boldsymbol{k}_{\mathbf{3}}^*$))+HW(g($\boldsymbol{k}_{\mathbf{4}}^*$))
    - Lower bound on algebraic degree and degree-1 approximations in $\mathbb{F}_p$, MELP/MEDP in $\mathbb{F}_2$

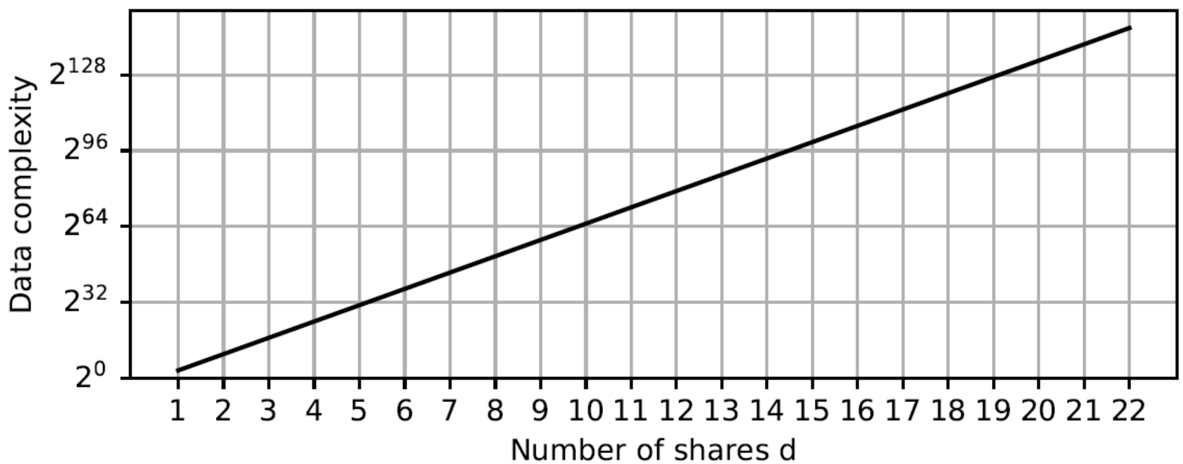
# Hardware implementation results



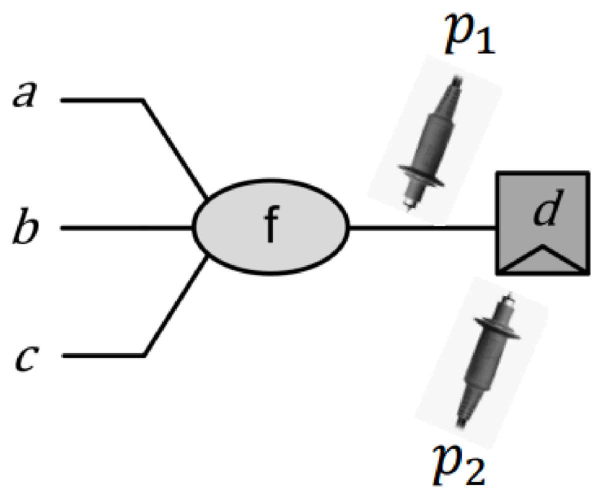
- 128-bit FPGA implementation




Latency



- Other advantages (improved security against glitches, …




**Glitch-extended probes:** probing any output of a combinatorial sub-circuit allows the adversary to observe all the sub-circuit inputs

Example: $p_1$ gives $a, b$ and $c$

# Conclusions for part #2



- Other advantages (improved security against glitches, trivial composition, linear refreshing)

# Conclusions for part #2 

- Other advantages (improved security against glitches, trivial composition, linear refreshing)
- If secure, game changer for embedded security
- Concrete relevance requires generalization
  - From Hamming weight leakages to linear, …
  - From univariate to multivariate leakages
    - Will possibly require noise again!
      - Or considering errors in measurements

# Conclusions for part #2



- Other advantages (improved security against glitches, trivial composition, linear refreshing)
- If secure, game changer for embedded security
- Concrete relevance requires generalization
  - From Hamming weight leakages to linear, …
  - From univariate to multivariate leakages
    - Will possibly require noise again!
      - Or considering errors in measurements
- Also raises important theoretical challenges
  - Learning with Leakage reduces to LPN
  - What about LWPR, LWPE? Can we connect them?

# Outline

- Side-channel analysis & the need of masking
- Boolean masking and the need of noise
- Prime masking and design challenges

- Fresh re-keying & basic models
- Hard physical learning problems

- **General conclusions for symmetric crypto**

# General conclusions



- The reduced "compatibility" between physical leakages and prime computations is a source of improved security for masking & re-keying
  - Yet the meaning of "compatible" differs for both

# General conclusions



- The reduced "compatibility" between physical leakages and prime computations is a source of improved security for masking & re-keying
  - Yet the meaning of "compatible" differs for both
- Leakage in symmetric crypto so far drove
  - Bitslice primitives with low AND complexity
  - Modes of operation for levelled implementations
- Could also drive new (prime) ciphers & the integration of hard physical learning problems in modes of operation (with the same primes?)

# General conclusions



- The reduced "compatibility" between physical leakages and prime computations is a source of improved security for masking & re-keying
  - Yet the meaning of "compatible" differs for both
- Leakage in symmetric crypto so far drove
  - Bitslice primitives with low AND complexity
  - Modes of operation for levelled implementations
- Could also drive new (prime) ciphers & the integration of hard physical learning problems in modes of operation (with the same primes?)
- Both have application in PQ asymmetric crypto!

# THANKS!

https://perso.uclouvain.be/fstandae/

**We are hiring on these topics** erc

**Proposition 3 (Properties of $s$-bounded pseudo-linear functions).** *Let $f \in \mathsf{C}_1^s$ with $ts < p$, where $t = \lceil \log p \rceil$, then the following holds:*

- $\mathsf{v}_f \geq \lceil \frac{p}{ts+1} \rceil$,
- $\mathsf{w}_f \geq p - ts - 1$.

*And assuming $\mathsf{v}_f \neq p$, we further have:*

- $\mathsf{deg}(f) \geq \lceil \frac{p}{ts+1} \rceil$,
- $\mathsf{nl}(f) \geq \min\left(p - \mathsf{v}_f, \max\left(\lceil \frac{p}{ts+1} \rceil - 1, p - ts - 1\right)\right)$.