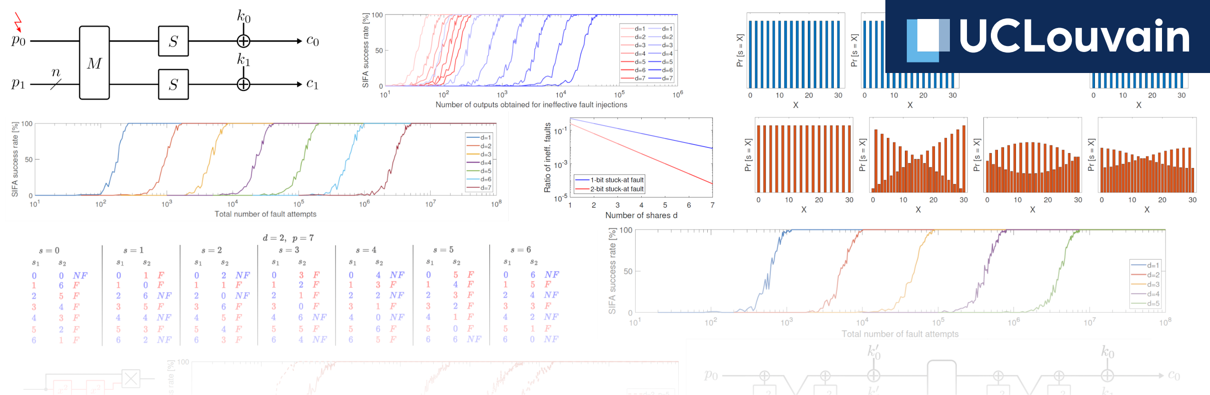


$d = 2,\ p = 7$

$s=0$ s_1	s_2		$s=1$ s_1	s_2		$s=2$ s_1	s_2		$s=3$ s_1	s_2		$s=4$ s_1	s_2		$s=5$ s_1	s_2		$s=6$ s_1	s_2	
0	0	NF	0	1	F	0	2	NF	0	3	F	0	4	NF	0	5	F	0	6	NF
1	6	F	1	0	F	1	1	F	1	2	F	1	3	F	1	4	F	1	5	F
2	5	F	2	6	NF	2	0	NF	2	1	F	2	2	NF	2	3	F	2	4	NF
3	4	F	3	5	F	3	6	F	3	0	F	3	1	F	3	2	F	3	3	F
4	3	F	4	4	NF	4	5	F	4	6	NF	4	0	NF	4	1	F	4	2	NF
5	2	F	5	3	F	5	4	F	5	5	F	5	6	F	5	0	F	5	1	F
6	1	F	6	2	NF	6	3	F	6	4	NF	6	5	F	6	6	NF	6	0	NF

UCLouvain

UCLouvain Institute of Information and Communication Technologies, Electronics and Applied Mathematics (ICTEAM)

Prime Masking vs. Faults - Exponential Security Amplification against Selected Classes of Attacks

Thorben Moos, Sayandeep Saha, François-Xavier Standaert

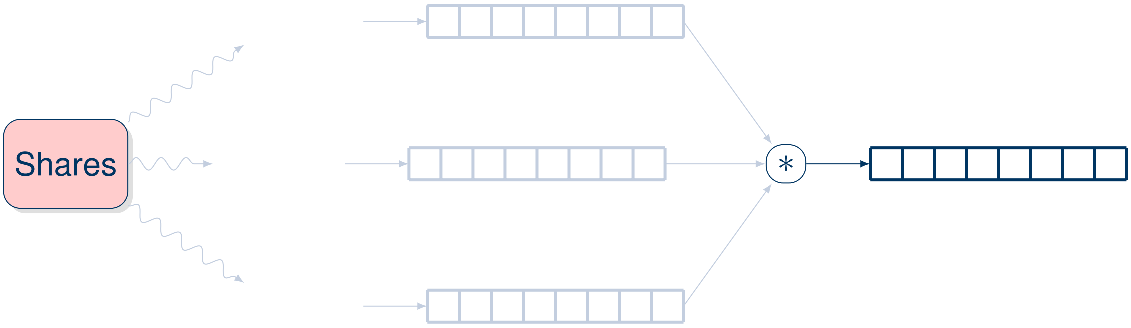
UCLouvain, IIT Bombay

Sep 5th, 2024

European Research Council

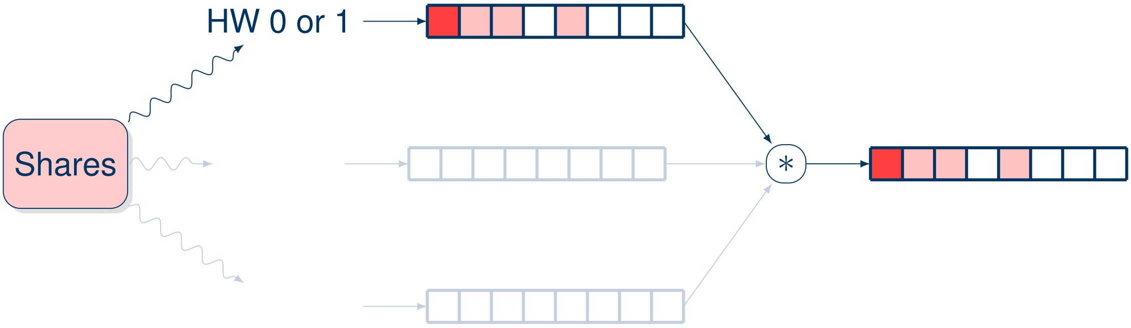
Masking against Side Channel Adversaries

Adversaries Make Imprecise Observations, Masking Amplifies Imprecision:



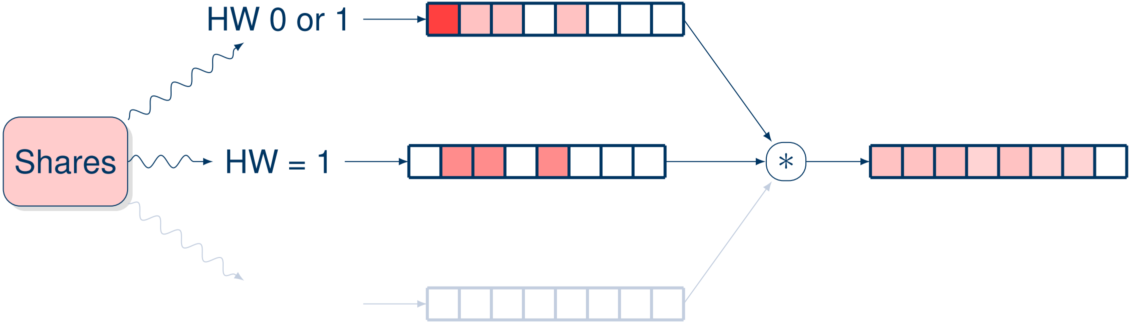
Masking against Side Channel Adversaries

Adversaries Make Imprecise Observations, Masking Amplifies Imprecision:



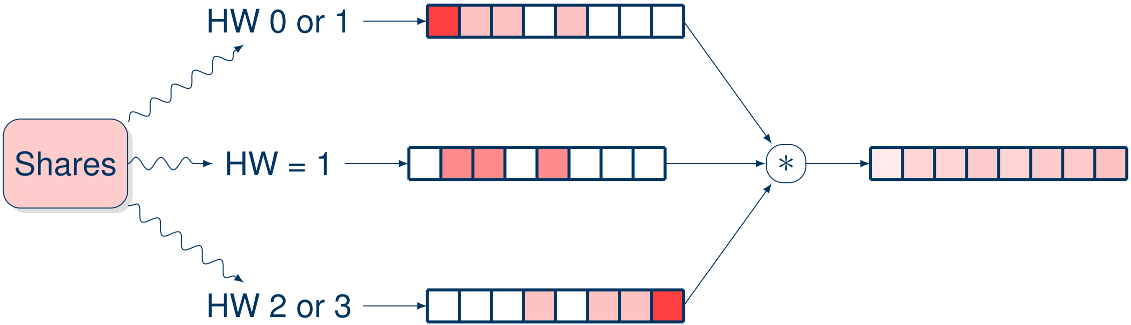
Masking against Side Channel Adversaries

Adversaries Make Imprecise Observations, Masking Amplifies Imprecision:

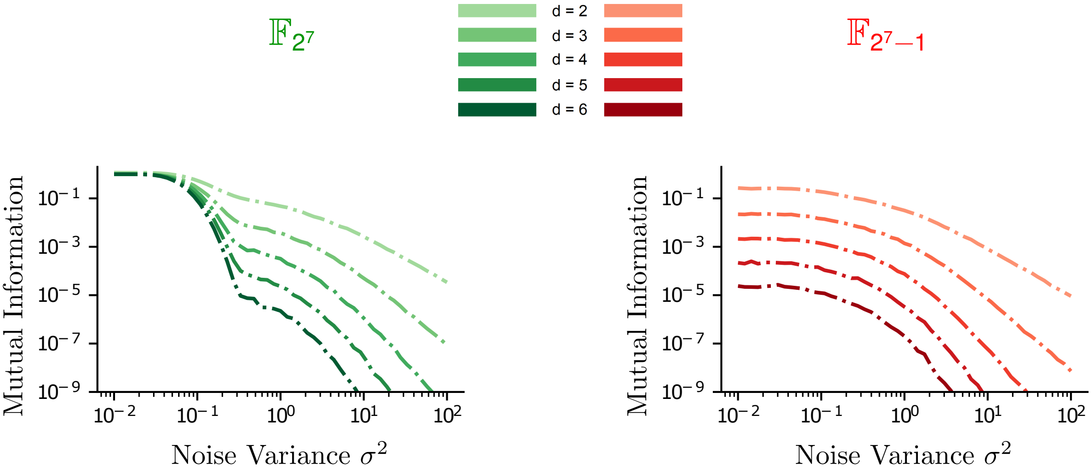


Masking against Side Channel Adversaries

Adversaries Make Imprecise Observations, Masking Amplifies Imprecision:



Prime-Field Masking



State of the Art

- Dziembowski et al., TCC 2016 [1]:
  - Masking in groups of prime order can amplify arbitrarily low noise levels (lack of subgroups)
  - Exponential security in the number of shares in presence of any non-injective leakage function
- Masure et al., Eurocrypt 2023 [2]:
  - Information theoretic evaluation + first practical results + toy AES-prime cipher
- Cassiers et al., TCHES 2023 [3]:
  - Efficient arbitrary-order composable masked gadgets for *squaring*, half as costly as multipl.
- Faust et al., Eurocrypt 2024 [4]:
  - For Hamming-weight-like leakage functions, security of prime-field masking is $\approx \log(\mathrm{p})^{\mathrm{d}}$
- Grassi et al., Eurocrypt 2024 [5]:
  - FPM (Feistel for Prime Masking) family of tweakable block ciphers, small — pSquare for HW

[1] Dziembowski, Faust and Skórski, Optimal Amplification of Noisy Leakages, TCC 2016
[2] Masure, Méaux, Moos and Standaert, Effective and Efficient Masking with Low Noise using Small-Mersenne-Prime Ciphers, Eurocrypt 2023
[3] Cassiers, Masure, Momin, Moos and Standaert, Prime-Field Masking in Hardware and its Soundness against Low-Noise SCA Attacks, TCHES 2023
[4] Faust et al., Connecting Leakage-Resilient Secret Sharing to Practice: Scaling Trends and Physical Dependencies of Prime Field Masking, Eurocrypt 2024
[5] Grassi et al., Generalized Feistel Ciphers for Efficient Prime Field Masking, Eurocrypt 2024

This Work

In this work we explore whether prime-field masking can provide similarly beneficial properties against fault attackers.

Background Fault Attacks

Three classes of fault attacks are relevant for this talk:

1. **DFA**: Differential Fault Analysis (DFA) attacks rely on faulting data or operations and then exploiting the differentials that can be observed between correct and faulty outputs.
2. **SIFA-1**: Statistical Ineffective Fault Attacks (SIFA) Type 1, rely on faulting data or operations and then exploiting the dependency between the effectiveness of the fault injection and the value to be faulted.
3. **SIFA-2**: Statistical Ineffective Fault Attacks (SIFA) Type 2, rely on faulting data or operations and then exploiting the conditional propagation of a faulted value based on a sensitive intermediate.

Observation 1 - Single Share LSB Toggle/Bit-Flip

Flipping the LSB of a single share of a **Boolean** encoding (top) always leads to the same bit being flipped in the secret intermediate after recombination. Doing the same on a **Prime-field** encoding (bottom) can lead to different faults in the secret intermediate.

Correct	Faulty
$\text{0x23} \oplus \text{0x48} = \text{0x6B}$	$\text{0x22} \oplus \text{0x48} = \text{0x6A}$
$\text{0x5A} \oplus \text{0x31} = \text{0x6B}$	$\text{0x5B} \oplus \text{0x31} = \text{0x6A}$
$\text{0x72} \oplus \text{0x19} = \text{0x6B}$	$\text{0x73} \oplus \text{0x19} = \text{0x6A}$
$\text{0x4F} \oplus \text{0x24} = \text{0x6B}$	$\text{0x4E} \oplus \text{0x24} = \text{0x6A}$

Correct	Faulty
$\text{0x23} + \text{0x48} \bmod \text{0x7F} = \text{0x6B}$	$\text{0x22} + \text{0x48} \bmod \text{0x7F} = \text{0x6A}$
$\text{0x5A} + \text{0x11} \bmod \text{0x7F} = \text{0x6B}$	$\text{0x5B} + \text{0x11} \bmod \text{0x7F} = \text{0x6C}$
$\text{0x72} + \text{0x78} \bmod \text{0x7F} = \text{0x6B}$	$\text{0x73} + \text{0x78} \bmod \text{0x7F} = \text{0x6C}$
$\text{0x4F} + \text{0x1C} \bmod \text{0x7F} = \text{0x6B}$	$\text{0x4E} + \text{0x1C} \bmod \text{0x7F} = \text{0x6A}$

Observation 1 - Single Share LSB Toggle/Bit-Flip

Result:

- Faults injected into the secret intermediate cannot be fully controlled even by a perfect adversary in case of prime-field masking
- This makes DFA attacks on duplicated circuits which require the same fault injection in multiple instances less likely to succeed
- The effect appears to be exponential in the number of redundancy domains

Observation 2 - All LSBs Stuck-at-0

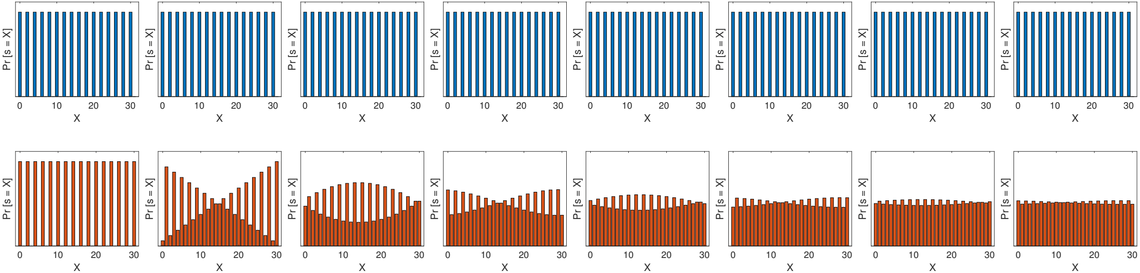
Keeping the LSBs of all shares of a **Boolean** encoding (top) stuck-at-0 always leads to ineffective/effective faults for even/odd secret intermediate values. Doing the same on a **Prime-field** encoding (bottom) can lead to effective and ineffective faults for any value.

Correct	Faulty
$\text{0x23} \oplus \text{0x48} = \text{0x6B}$	$\text{0x22} \oplus \text{0x48} = \text{0x6A}$
$\text{0x72} \oplus \text{0x19} = \text{0x6B}$	$\text{0x72} \oplus \text{0x18} = \text{0x6A}$
$\text{0x22} \oplus \text{0x48} = \text{0x6A}$	$\text{0x22} \oplus \text{0x48} = \text{0x6A}$
$\text{0x73} \oplus \text{0x19} = \text{0x6A}$	$\text{0x72} \oplus \text{0x18} = \text{0x6A}$

Correct	Faulty
$\text{0x23} + \text{0x48} \bmod \text{0x7F} = \text{0x6B}$	$\text{0x22} + \text{0x48} \bmod \text{0x7F} = \text{0x6A}$
$\text{0x72} + \text{0x78} \bmod \text{0x7F} = \text{0x6B}$	$\text{0x72} + \text{0x78} \bmod \text{0x7F} = \text{0x6B}$
$\text{0x22} + \text{0x48} \bmod \text{0x7F} = \text{0x6A}$	$\text{0x22} + \text{0x48} \bmod \text{0x7F} = \text{0x6A}$
$\text{0x73} + \text{0x76} \bmod \text{0x7F} = \text{0x6A}$	$\text{0x72} + \text{0x76} \bmod \text{0x7F} = \text{0x69}$

Observation 2 - All LSBs Stuck-at-0

Probability distributions of secrets shared over fields $\mathbb{F}_{2^5}$ (top) and $\mathbb{F}_{2^5-1}$ (bottom) when given knowledge of outputs obtained for ineffective fault injections when LSBs stuck-at-0:



Observation 2 - All LSBs Stuck-at-0

Result:

- Even a perfect adversary cannot exclude any potential values for the secret intermediate with certainty in case of prime-field masking
- This makes SIFA-1 attacks less likely to succeed using the same amount of fault attempts
- effect appears to be exponential in the number of shares

Fault Model

The Heisenberg Adversary Model

This adversary model allows the insertion of an almost arbitrary number of stuck-at-0, stuck-at-1 or toggle/bit-flip faults into the encoded state with perfect accuracy and precision at the same time. Alternatively, the adversary may insert an arbitrary number of said faults, but only with almost perfect accuracy and precision.

- Details about the meaning of the keyword almost are found in the paper
- Intuitively, we allow multi-bit faults with perfect accuracy and precision in each share while at the same time prohibiting the trivial attack of simply faulting all bits
- In the following we assume that a redundancy-based fault detection countermeasure is present and working

Fault Model

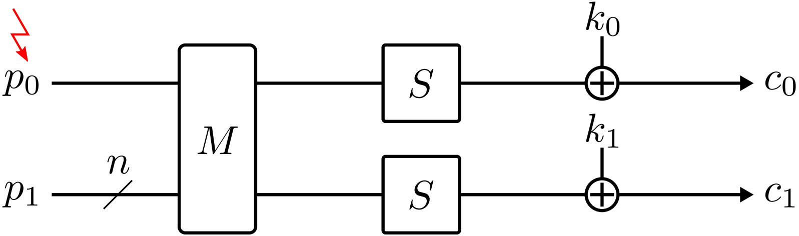
How realistic is an adversary that reliably faults all shares of an encoding?

- In hardware implementations, the same bit of all shares is often processed or stored by gates in very close proximity to each other [6].
- In hardware implementations, the same bit of all shares may be stored in memory elements that are clocked by the same clock buffer [6].

Fault attacks that target only a limited region may still realistically be capable of faulting all shares at once (at a certain probability).

[6] Sadhukhan, Saha, and Mukhopadhyay, Antisifa-cad: A framework to thwart SIFA at the layout level, ICCAD 2022

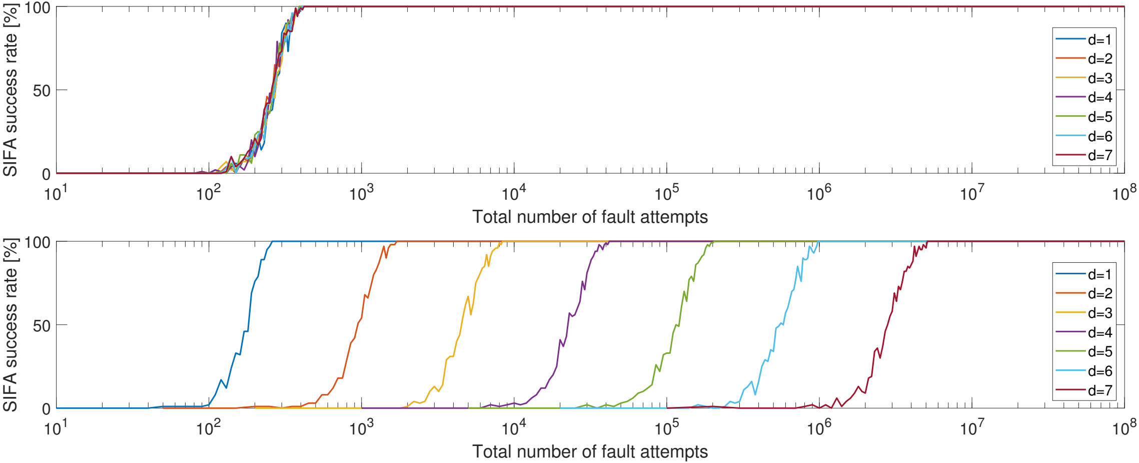
Toy Circuit for SIFA-1 Attack Simulations



- For **Boolean** masking, we use the field $\mathbb{F}_{2^n}$ with $n = 8$,
  $M = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$, $M^{-1} = \begin{pmatrix} 82 & 164 \\ 164 & 82 \end{pmatrix}$, S the AES S-box.
- For **Prime** masking, we use the field $\mathbb{F}_{2^n-1}$ with $n = 7$,
  $M = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$, $M^{-1} = \begin{pmatrix} 42 & 43 \\ 43 & 42 \end{pmatrix}$, S the AES-prime S-box.

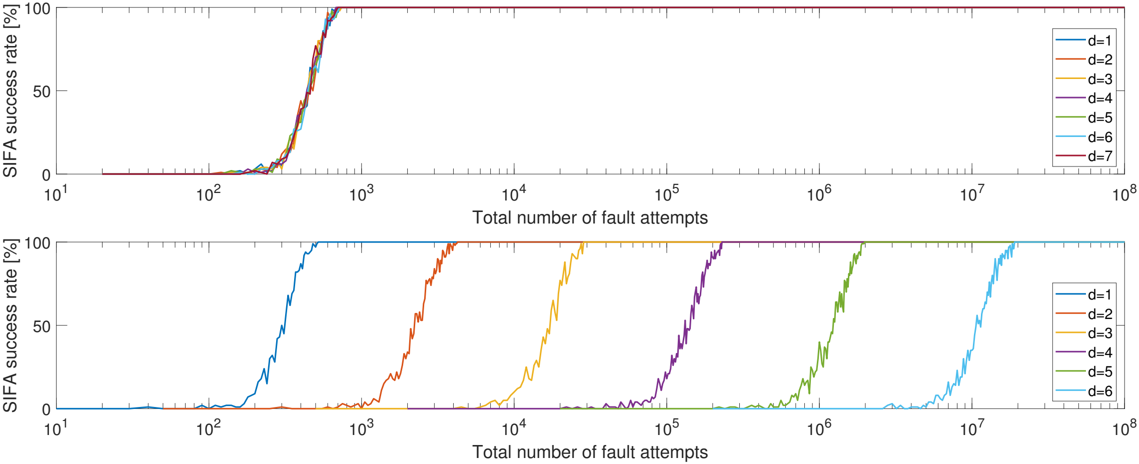
LSBs Stuck-at-0 SIFA-1 – 100% Biasing Success

Boolean masking over $\mathbb{F}_{2^8}$ (top) vs. **Prime** masking over $\mathbb{F}_{2^7-1}$ (bottom).



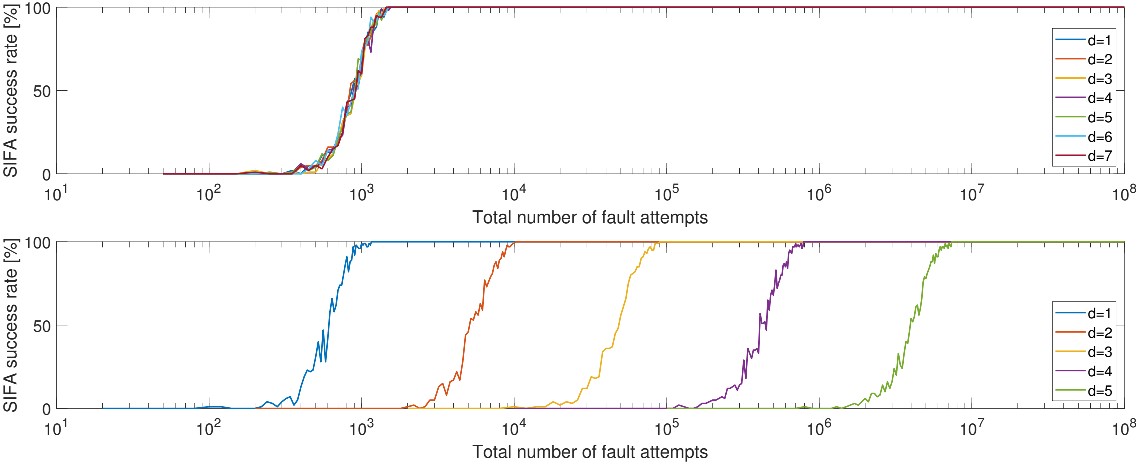
LSBs Stuck-at-0 SIFA-1 – 80% Biasing Success

Boolean masking over $\mathbb{F}_{2^8}$ (top) vs. **Prime** masking over $\mathbb{F}_{2^7-1}$ (bottom).



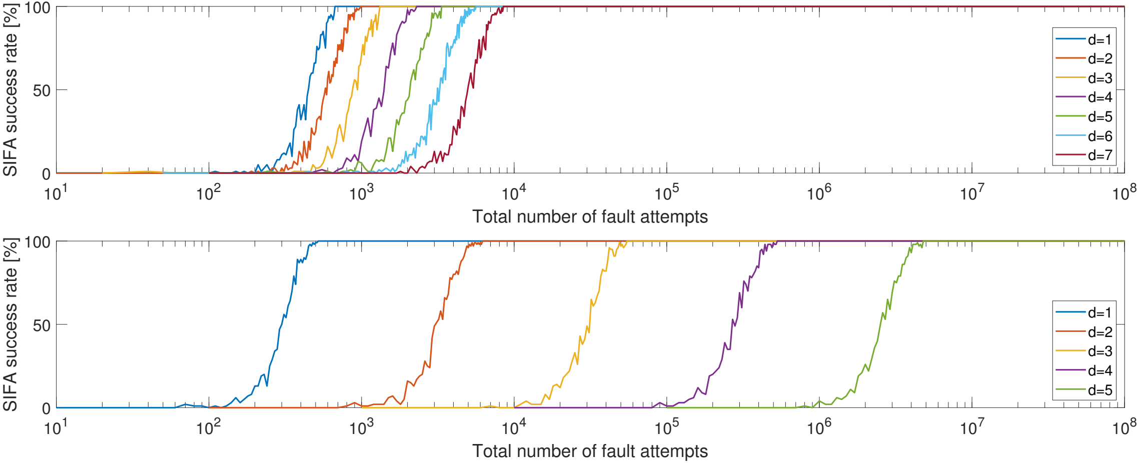
LSBs Stuck-at-0 SIFA-1 – 60% Biasing Success

Boolean masking over $\mathbb{F}_{2^8}$ (top) vs. **Prime** masking over $\mathbb{F}_{2^7-1}$ (bottom).



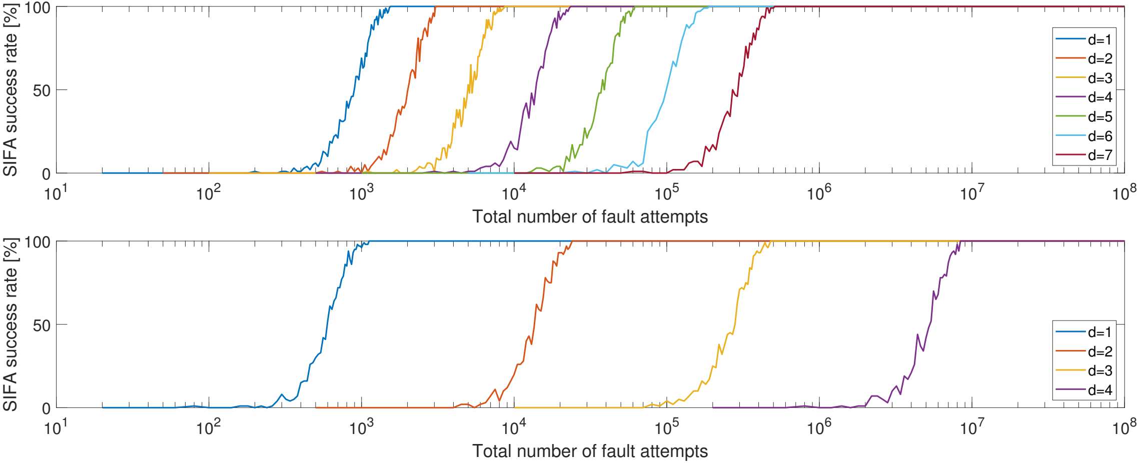
LSBs Stuck-at-0 SIFA-1 – 80% Injection Prob.

Boolean masking over $\mathbb{F}_{2^8}$ (top) vs. **Prime** masking over $\mathbb{F}_{2^7-1}$ (bottom).



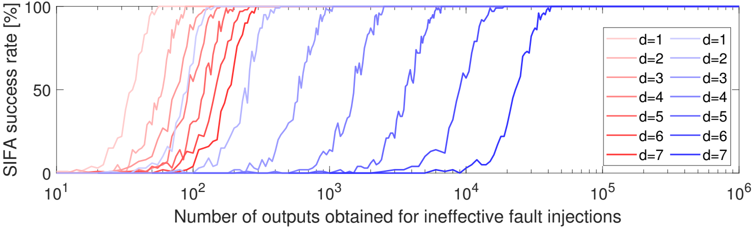
LSBs Stuck-at-0 SIFA-1 – 60% Injection Prob.

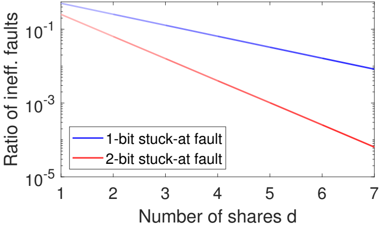
Boolean masking over $\mathbb{F}_{2^8}$ (top) vs. **Prime** masking over $\mathbb{F}_{2^7-1}$ (bottom).



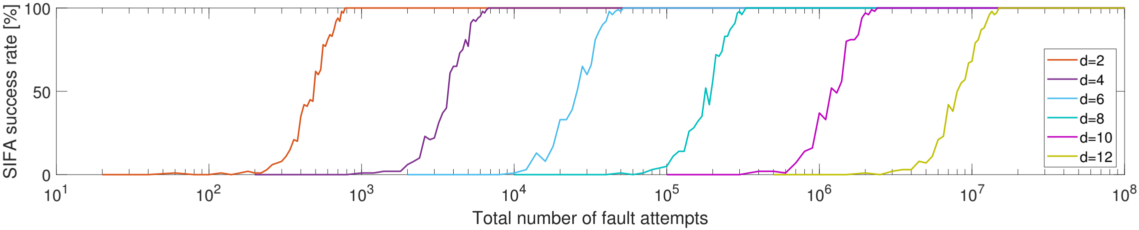
LSBs Stuck-at-0 SIFA-1 – Two Separate Effects

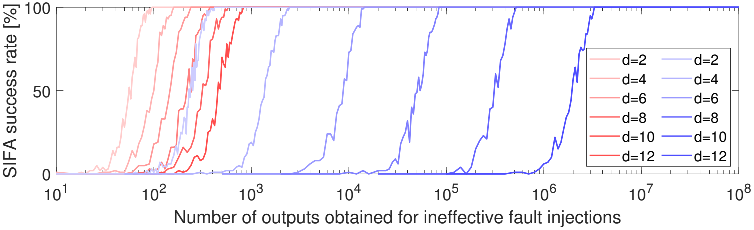
Separate depiction of the number of ineffective faults needed to perform a successful SIFA-1 attack and the probability to obtain an ineffective fault over the number of shares:

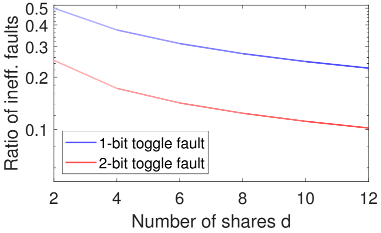




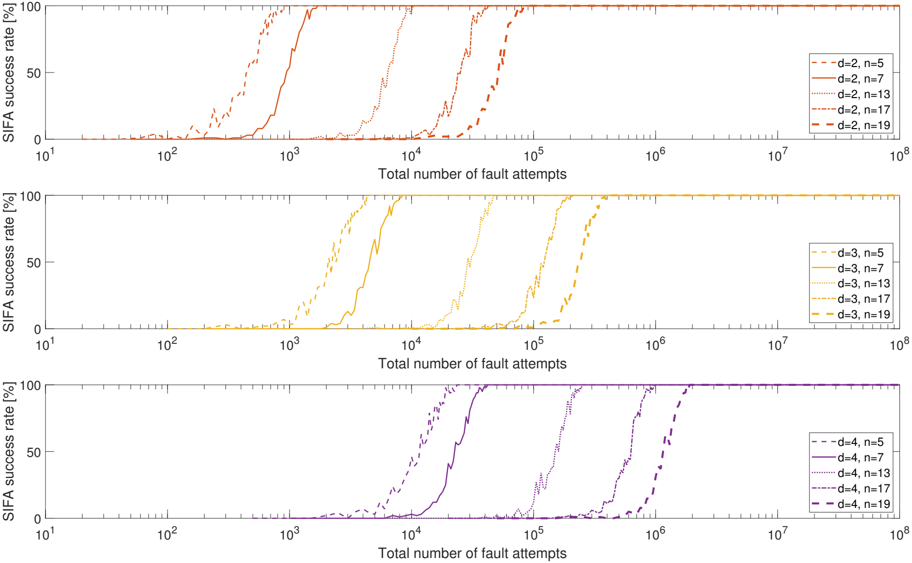
LSBs Toggle/Bit-Flip SIFA-1, Even Number of Shares



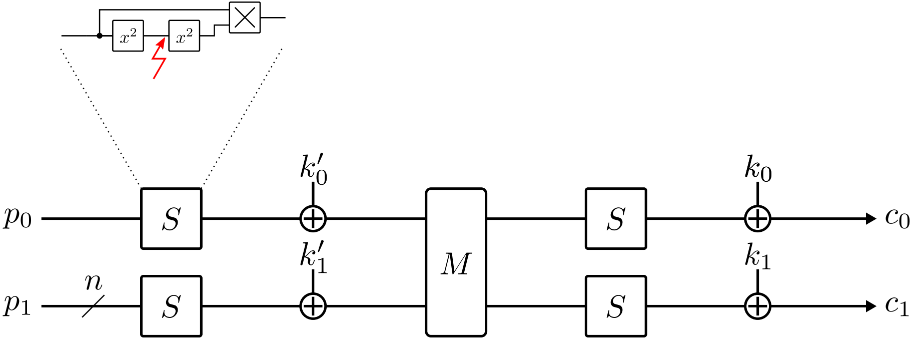




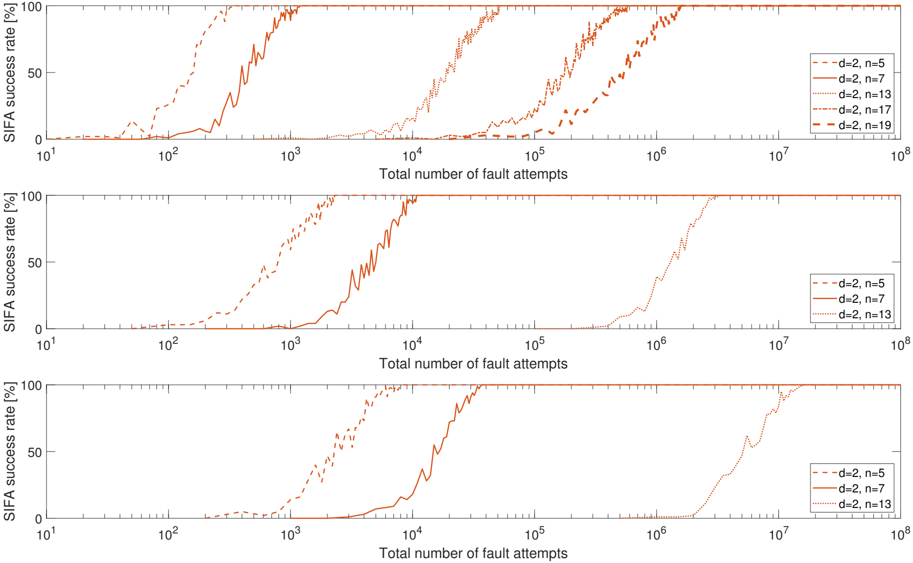
LSBs Stuck-at-0 SIFA-1 – Field Size Dependence



Toy Circuit for SIFA-2 Attack Simulations



LSB Toggle/Bit-Flip SIFA-2 – Field Size Dependence

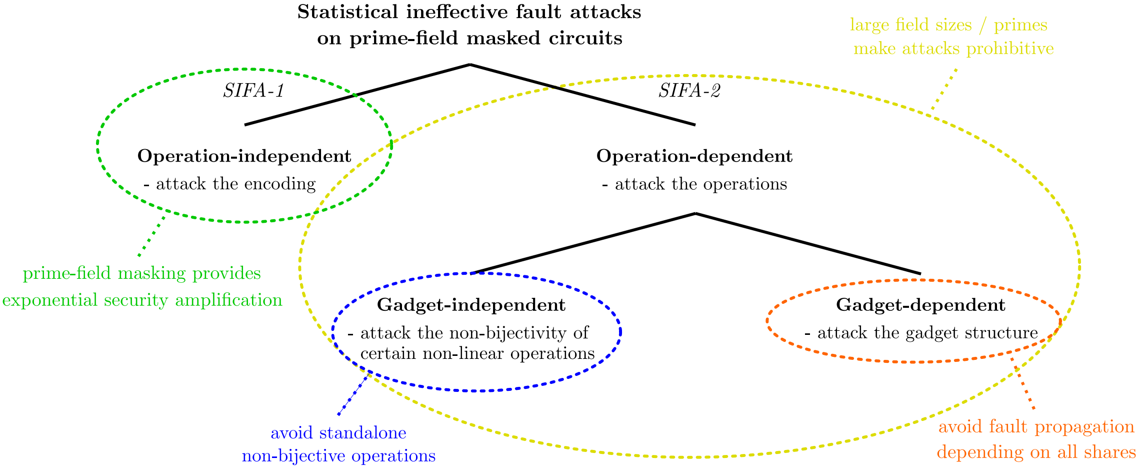


There's More

The paper provides additional results that have been omitted in this talk for brevity, incl.:

- Closed-form expressions to predict the likelihood of ineffective faults under multiple fault models (stuck-at/toggle) and estimate the number of outputs needed for key recovery
- Concrete SIFA-1 complexities for masked AES and AES-prime implementations
- Analysis of the problems arising from the non-bijectivity of multiplications and squarings
- Guidelines on how to construct fault resistant cryptographic primitives over prime fields

Overview: SIFA Security of Prime Masking



Statistical ineffective fault attacks on prime-field masked circuits

SIFA-1

Operation-independent

- attack the encoding

prime-field masking provides exponential security amplification

SIFA-2

Operation-dependent

- attack the operations

Gadget-independent

- attack the non-bijectivity of certain non-linear operations

avoid standalone non-bijective operations

Gadget-dependent

- attack the gadget structure

avoid fault propagation depending on all shares

large field sizes / primes make attacks prohibitive

Conclusion

- Prime-field masking offers a natural resistance to SIFA-1 attacks (and somewhat weaker resistance to some DFA attacks)
- This resistance is obtained for free for masked algorithms over prime fields
- To the best of our knowledge this is the first countermeasure that can deliver exponential security amplification in such a strong adversary model
- Prime-field primitives over larger fields or based on bijective non-linear operations may provide security against SIFA-2 and FTA as well
- Next: Building cryptographic primitives that best leverage the SCA and FIA advantages of prime-field masking