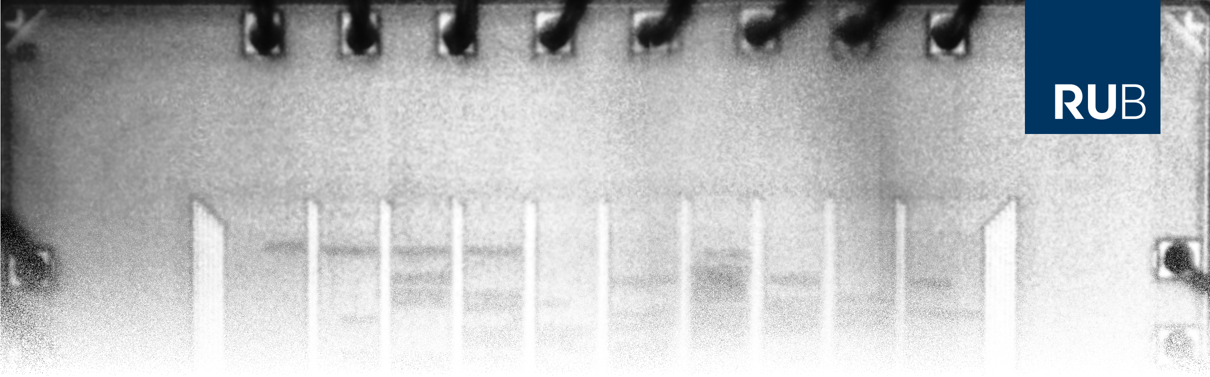


RUHR-UNIVERSITÄT BOCHUM

Unrolled Cryptography on Silicon

A Physical Security Analysis

Thorben Moos

Ruhr University Bochum, Horst Görtz Institute for IT Security, Germany

September 15th, 2020

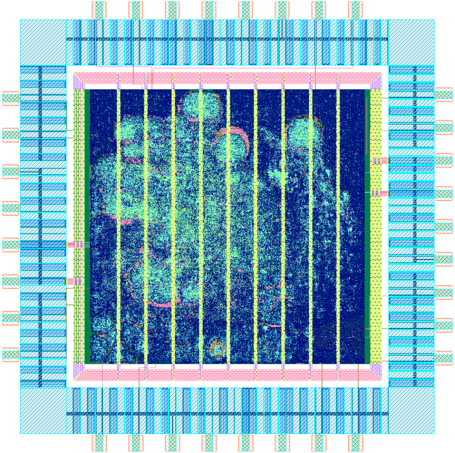
Deutsche Forschungsgemeinschaft

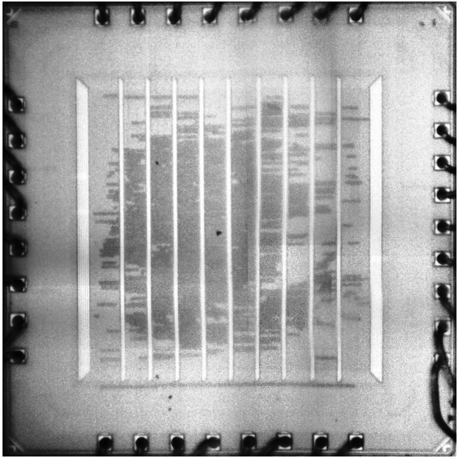
Section 1

Introduction

Target

Introduction





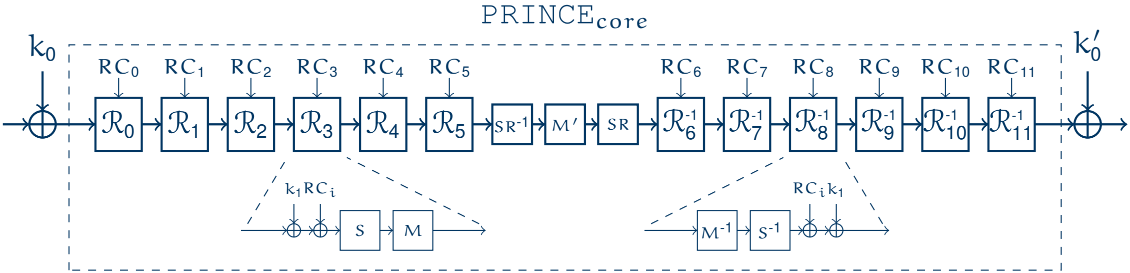
Background

Introduction

- Cryptographic primitives with high-speed (low-latency) performance in hardware have received growing attention in the last decade
- This design goal requires a short critical path as a fully-unrolled combinatorial circuit without memory elements
- PRINCE has been developed for high-speed single-cycle encryption and decryption at moderate hardware cost
- Tempting for many different applications, e.g., memory encryption

PRINCE

Introduction



Source: TikZ for Cryptographers, https://www.iacr.org/authors/tikz, Author Jérémy Jean

Motivation 1

Introduction

- Unrolled circuits are hard to protect against SCA attacks
- Glitch-resistant masking is arguably the most relevant class of SCA countermeasures for hardware circuits
- It can not easily be applied to unrolled circuits as it requires registers as synchronization stages
- Generic low-latency masking [1] causes an exponential increase in the circuit size when trying to avoid register stages
- However, it has been reported that the high parallelism, asynchronicity and speed of execution of unrolled circuits create an inherent resistance to side-channel attacks

Source: [1] Gross et al., Generic Low-Latency Masking in Hardware, TCHES Volume 2018 Issue 2

Motivation 2

Introduction

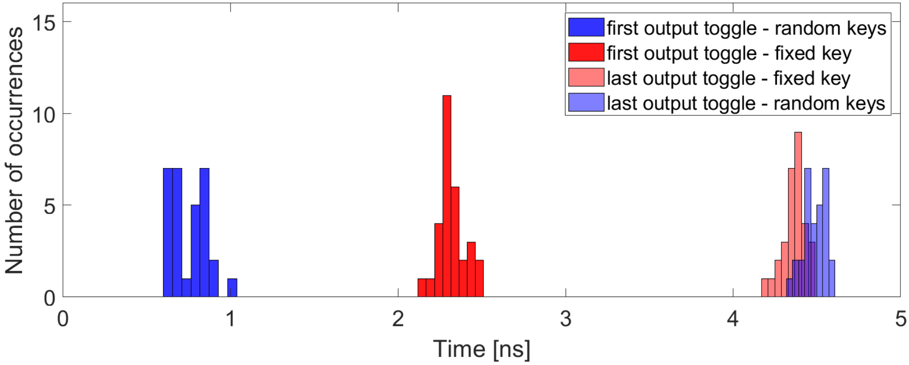
- Previous works on the physical security of unrolled `PRINCE` are all FPGA-based
- According to [2] an FPGA implementation occupies about **35**× as much area, consumes about **14**× as much dynamic power and is more than **4**× slower than an equivalent standard-cell-based ASIC design
- Hard to transfer conclusions from one platform to the other
- Static leakage of unrolled circuits has not been considered as a threat to such implementations yet

Source: [2] Kuon et al., Measuring the Gap Between FPGAs and ASICs, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD), 2007

Gate-Level Simulations

Introduction

- 9 169 logic gates corresponding to 10 036 (GE), synthesized for 200 MHz
- 114 803 gate transitions (avg) for random plaintext and key transition, 96% glitches
- 56 920 gate transitions (avg) for random plaintext transition, 92% glitches

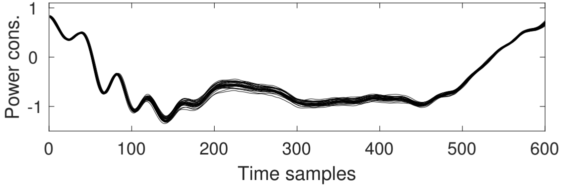


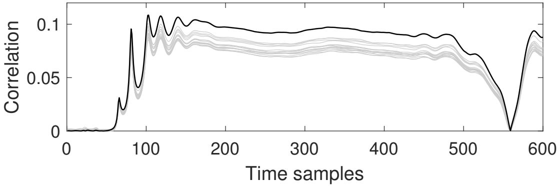
Section 2

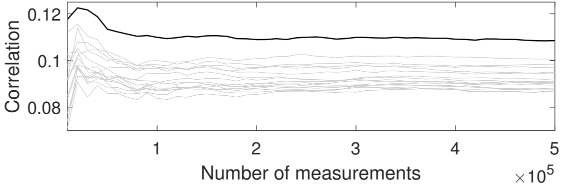
Experimental Results

No Reset

Dynamic Power Analysis

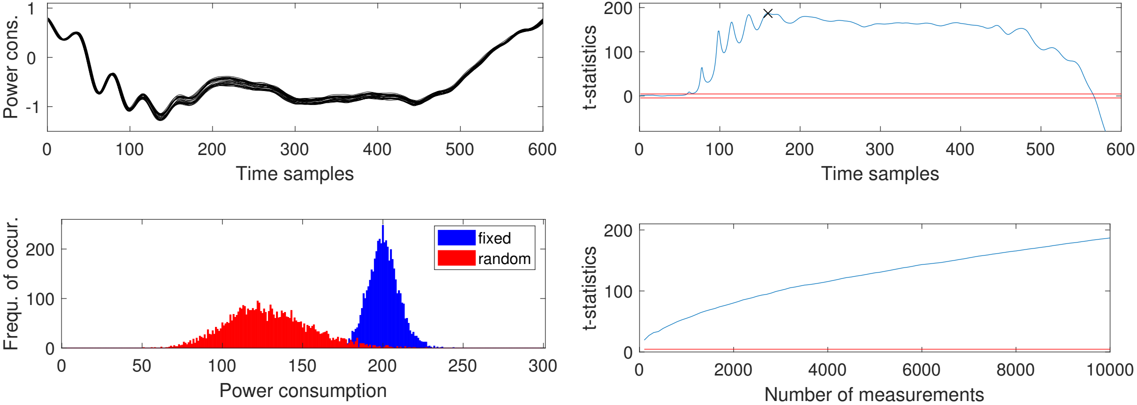






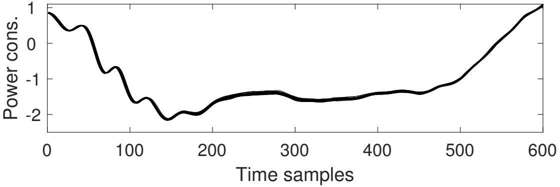
Plaintext Reset to Zero

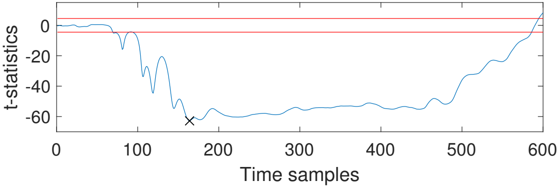
Dynamic Power Analysis

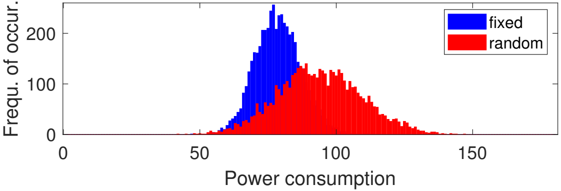


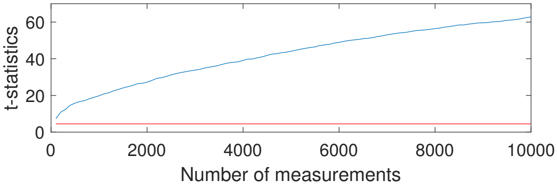
Plaintext and Key Reset to Zero

Dynamic Power Analysis



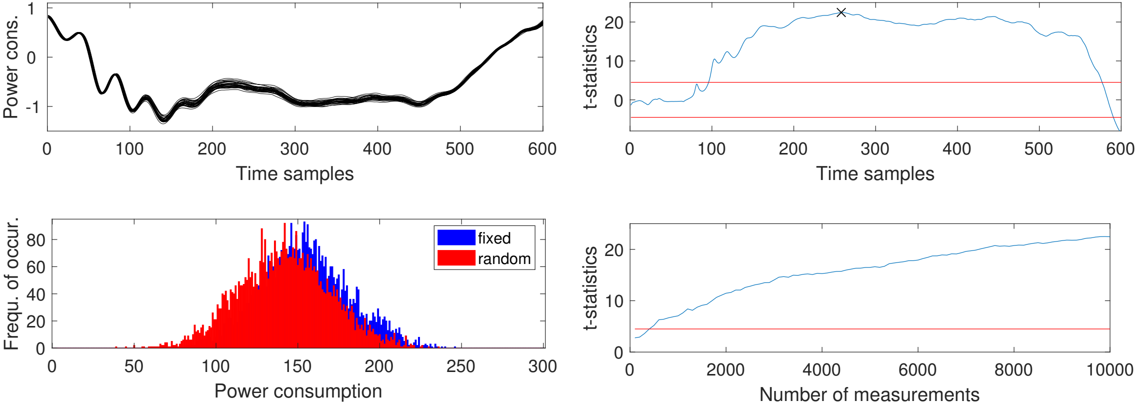






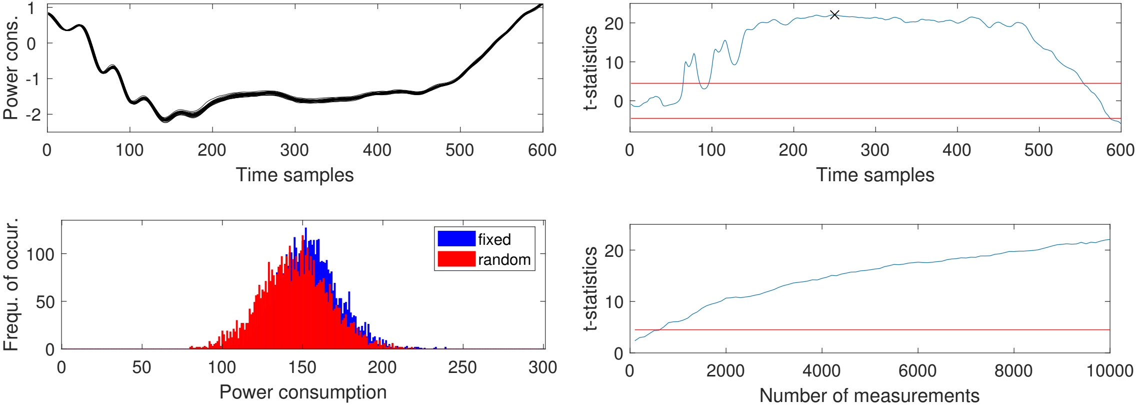
Plaintext Reset to Random Value

Dynamic Power Analysis



Plaintext and Key Reset to Random Value

Dynamic Power Analysis



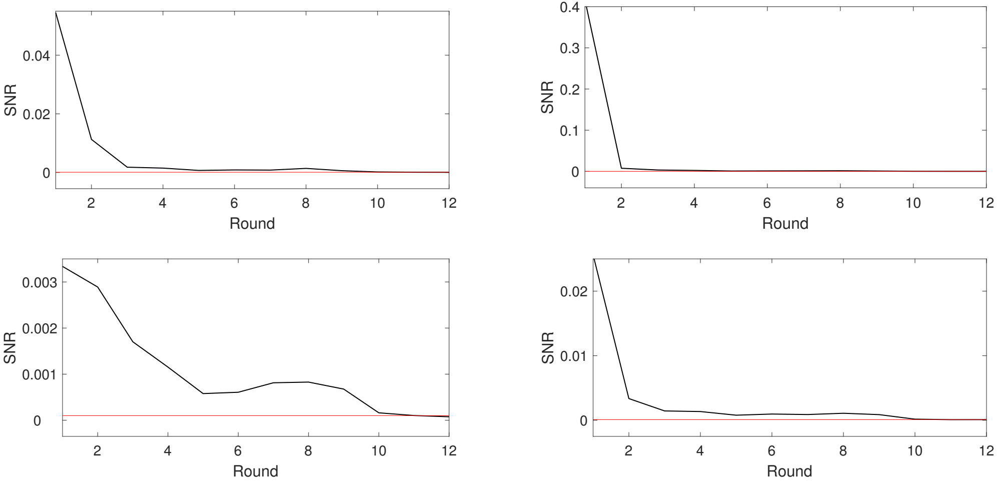
Plaintext and Key Reset to Random Value

Dynamic Power Analysis

Reset Type	Attack	Best Power Model Found	Rec. Nib.
no reset	CPA	$\mathrm{HD}(\mathrm{S}(p_{i-1,j} \oplus \hat{k}_j), \mathrm{S}(p_{i,j} \oplus \hat{k}_j))$	16/16
plain zero	CPA	$\mathrm{HD}(\mathrm{S}(0 \oplus \hat{k}_j), \mathrm{S}(p_{i,j} \oplus \hat{k}_j))$	7/16
plain and key zero	CPA	$\mathrm{HD}(\mathrm{S}(0 \oplus 0), \mathrm{S}(p_{i,j} \oplus \hat{k}_j))$	5/16
plain random	CPA	$\mathrm{HW}(\mathrm{S}(p_{i,j} \oplus \hat{k}_j))$	2/16
plain and key random	CPA	$\mathrm{HW}(\mathrm{S}(p_{i,j} \oplus \hat{k}_j))$	3/16

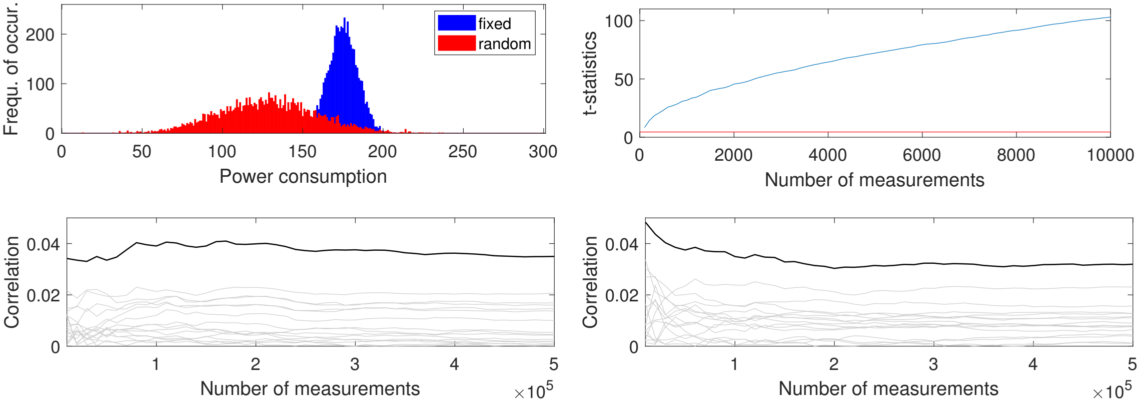
Signal-to-Noise-Ratio (SNR)

Dynamic Power Analysis



Static Power Results

Static Power Analysis



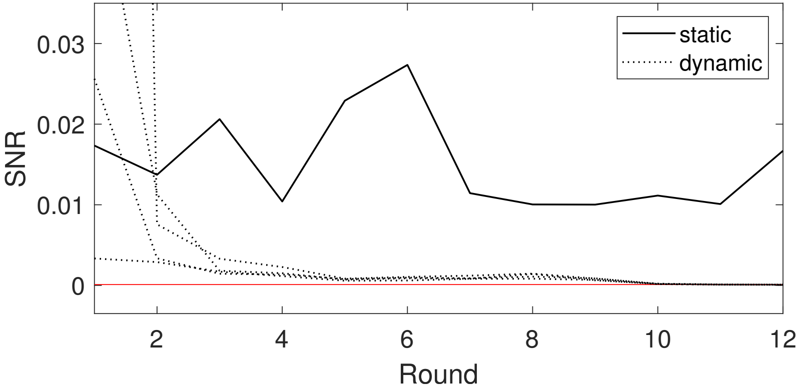
Static Power Results

Static Power Analysis

Round	Attack	Best Power Model Found	Rec. Nib.
first	CPA	LSB(S($p_{i,j} \oplus \hat{k}_j$))	15/16
last	CPA	LSB(S($c_{i,j} \oplus \hat{k}'_j$))	16/16

Signal-to-Noise-Ratio (SNR)

Static Power Analysis



Conclusion

- Protecting unrolled circuits without causing severe area or latency penalties is hard
- Some simple usage principles deliver promising results
- Resetting the plaintext input of an unrolled cipher to a random value between encryptions makes is effective against information leakage through the dynamic power
- Static power adversaries can remain dangerous in such a scenario if clock control is an option or if other mistakes are made
- Due to its nature the static power consumption is often the easiest way to extract the full 128-bit key of unrolled PRINCE because each round can be targeted with the same effort

Thank you for your attention.

Any questions?