P3 – Hardware-entangled cryptography

Cryptography based on hardware characteristics

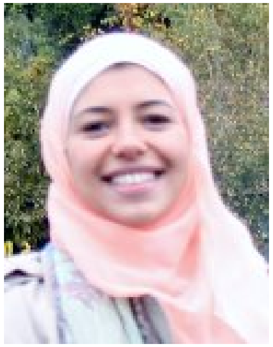
Who we are

Stefan Katzenbeisser

Ahmad-Reza Sadeghi

André Schaller

Nikolaos Athanasios Anagnostopoulos


Ghada Dessouky

CROSSING

Deutsche Forschungsgemeinschaft

TECHNISCHE UNIVERSITÄT DARMSTADT

Motivational use cases

- Authentication and identification

BUNDESREPUBLIK DEUTSCHLAND
FEDERAL REPUBLIC OF GERMANY / REPUBLIQUE FEDERALE D'ALLEMAGNE
PERSONALAUSWEIS
IDENTITY CARD / CARTE D'IDENTITE

T22000129

Name/Surname/Nom
MUSTERMANN
GEB. GABLER

Vornamen/Given names/Prénoms
ERIKA

Geburtstag/Date of birth/ Date de naissance
12.08.1964

Staatsangehörigkeit/Nationality/ Nationalité
DEUTSCH

Geburtsort/Place of birth/Lieu de naissance
BERLIN

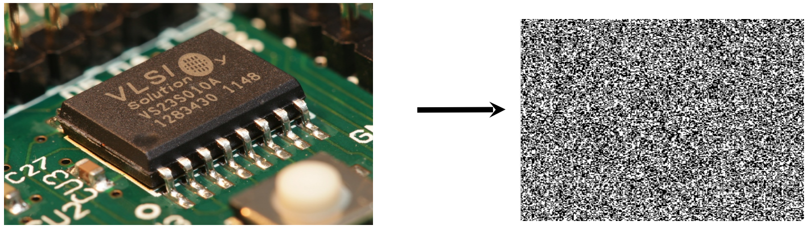
Gültig bis/Date of expiry/ Date d'expiration
31.10.2020

938568

Unterschrift der Inhaberin/des Inhabers - Signature of bearer - Signature de la titulaire/du titulaire

- Integrity of devices
  - Anti-counterfeiting
  - Tamper-evidence

- Lightweight security



CROSSING

DFG Deutsche Forschungsgemeinschaft

TECHNISCHE UNIVERSITÄT DARMSTADT

Hardware-entangled cryptography

Physical(ly) Unclonable Functions (PUFs)

- Functions embedded into physical objects
- Manufacturing process variations
  → unique identity for ICs
- Primitives similar to those employed in biometrics
  - "Hardware biometrics"

Stefan Katzenbeisser

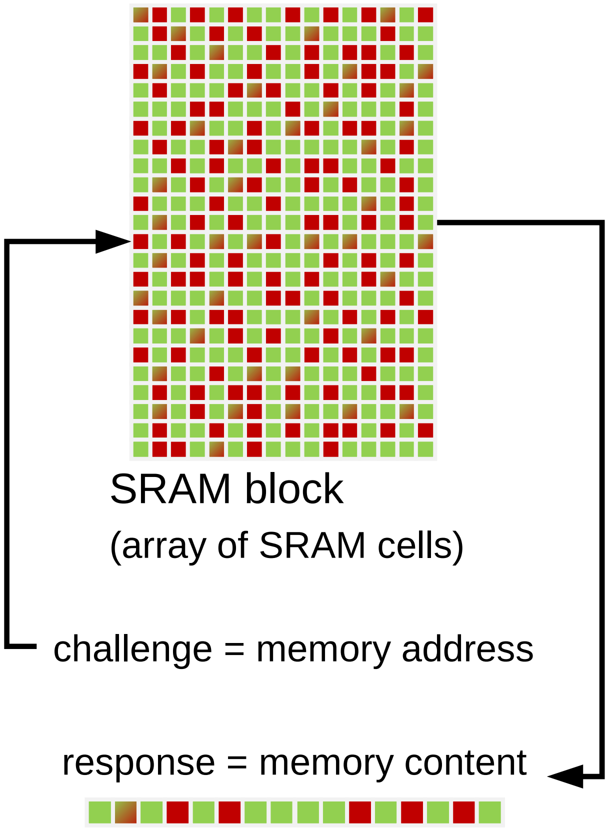
André Schaller

Nikolaos Athanasios Anagnostopoulos

Principles of PUFs

Physical(ly) Unclonable Functions (PUFs)

- Functions embedded into physical objects
- Manufacturing process variations
  → unique identity for ICs
- When queried with a challenge, a PUF generates a response (Challenge-Response Pair; CRP)
- The response depends on
  - the challenge **and**
  - specific physical properties of the object

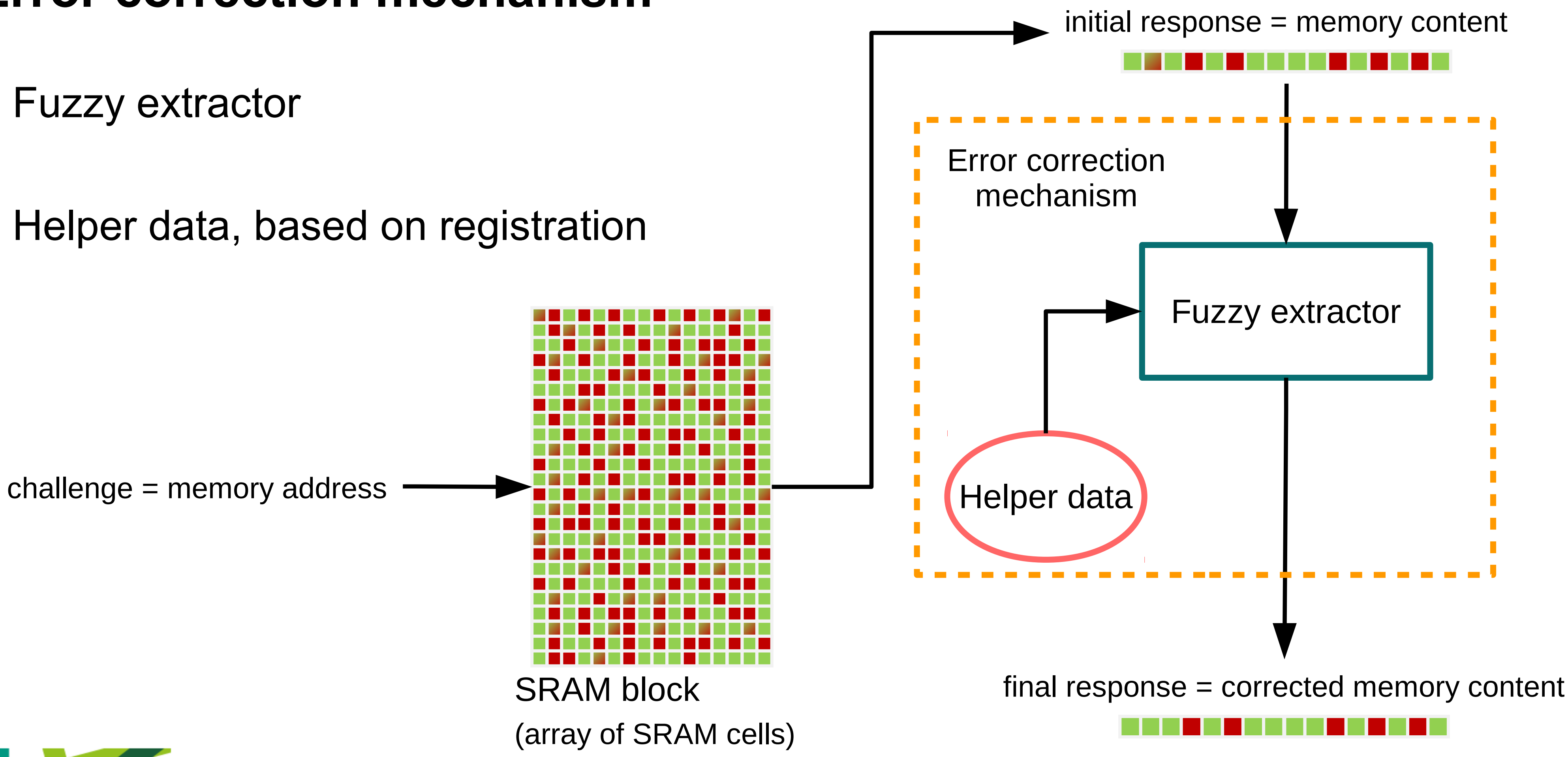


Principles of PUFs

Error correction mechanism

- Fuzzy extractor

- Helper data, based on registration



challenge = memory address

SRAM block
(array of SRAM cells)

initial response = memory content

Error correction mechanism

Fuzzy extractor

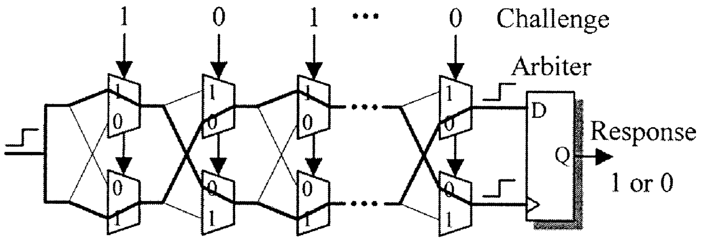
Helper data

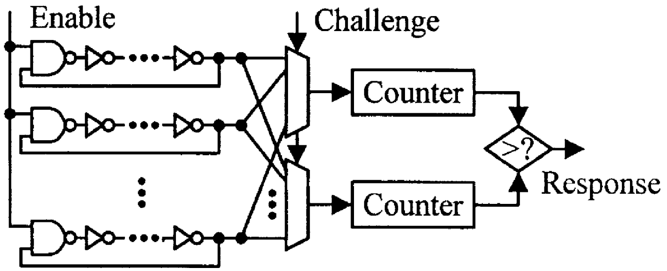
final response = corrected memory content

Strong and weak PUFs

Strong PUFs

- Multiple challenge-response pairs
- Delay-based PUFs
- Still on the prototype stage

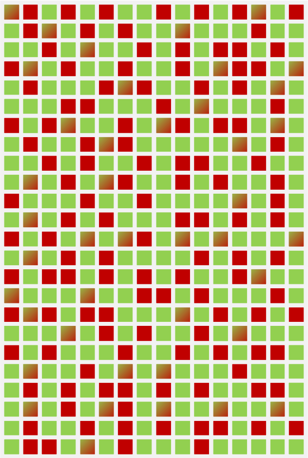




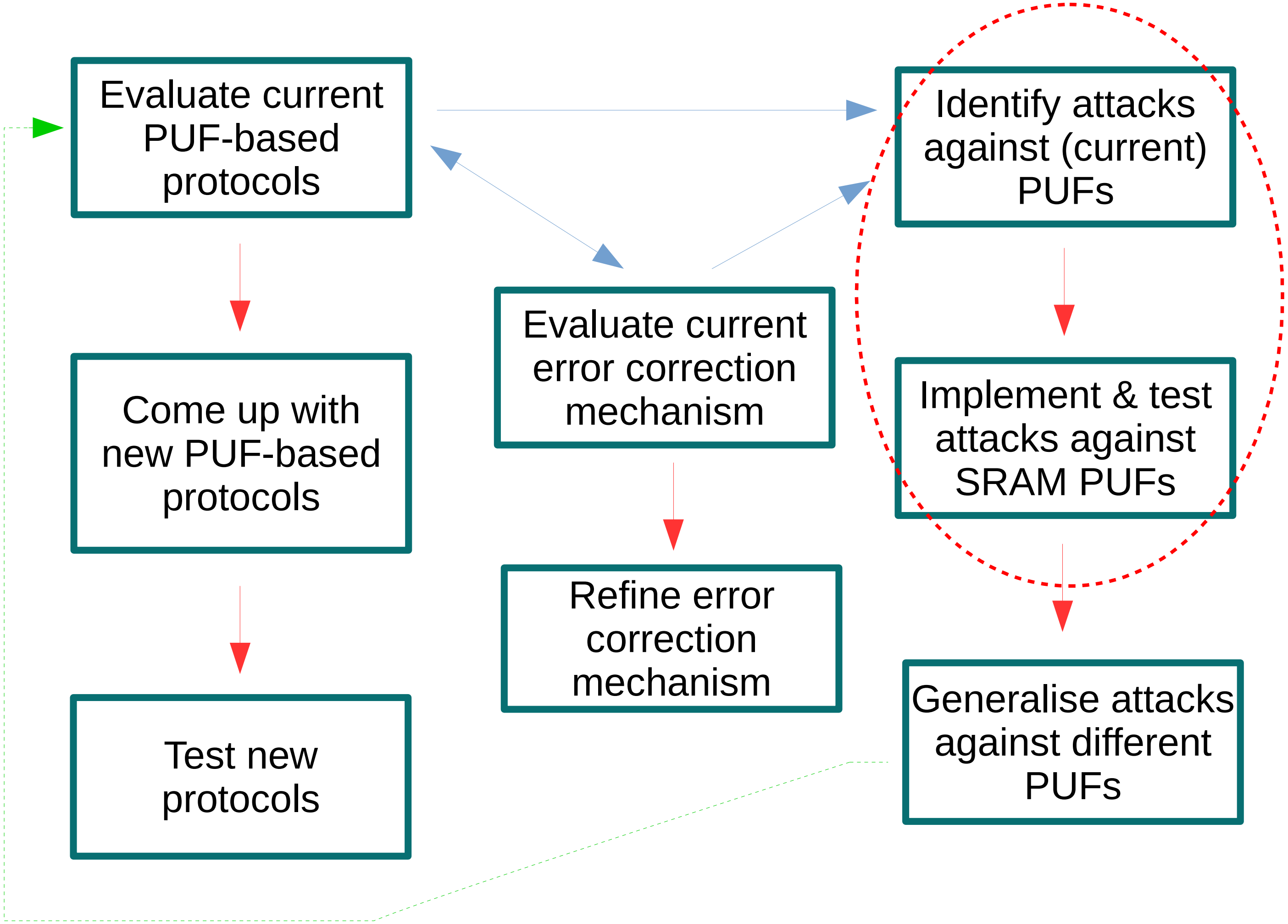
Strong and weak PUFs

Weak PUFs

- A single or very few challenge-response pairs
- Memory-based PUFs
- In production stage



What we do


Evaluate current PUF-based protocols
Come up with new PUF-based protocols
Test new protocols
Evaluate current error correction mechanism
Refine error correction mechanism
Identify attacks against (current) PUFs
Implement & test attacks against SRAM PUFs
Generalise attacks against different PUFs

Current PUF schemes

- ~~Optical PUFs~~
- Delay-based PUFs
  - Arbiter PUFs
  - Ring Oscillator PUFs
- Memory-based PUFs
  - Latch PUFs
  - Flip-Flop PUFs
  - **SRAM PUFs**

Attacks on PUFs

Desired effect

- Get/Predict/Model challenge-response pair
  - Man in the middle
  - Physical access
  - Logical access
- Disable/Make unavailable/Break PUF
  - Destroy PUF
  - Bypass PUF
- Force PUF into producing specific result
  - Physical access
  - Logical access

Attacks on PUFs

Means and ways of attacks

- Hardware
  - Side-channel
  - Invasive
- Software

- Internal
- External
  - Man in the middle
  - Cloning (Guessing + error correction)

- Target
  - PUF structure itself
  - Error-correction mechanism

Attacks on PUFs

Reasons

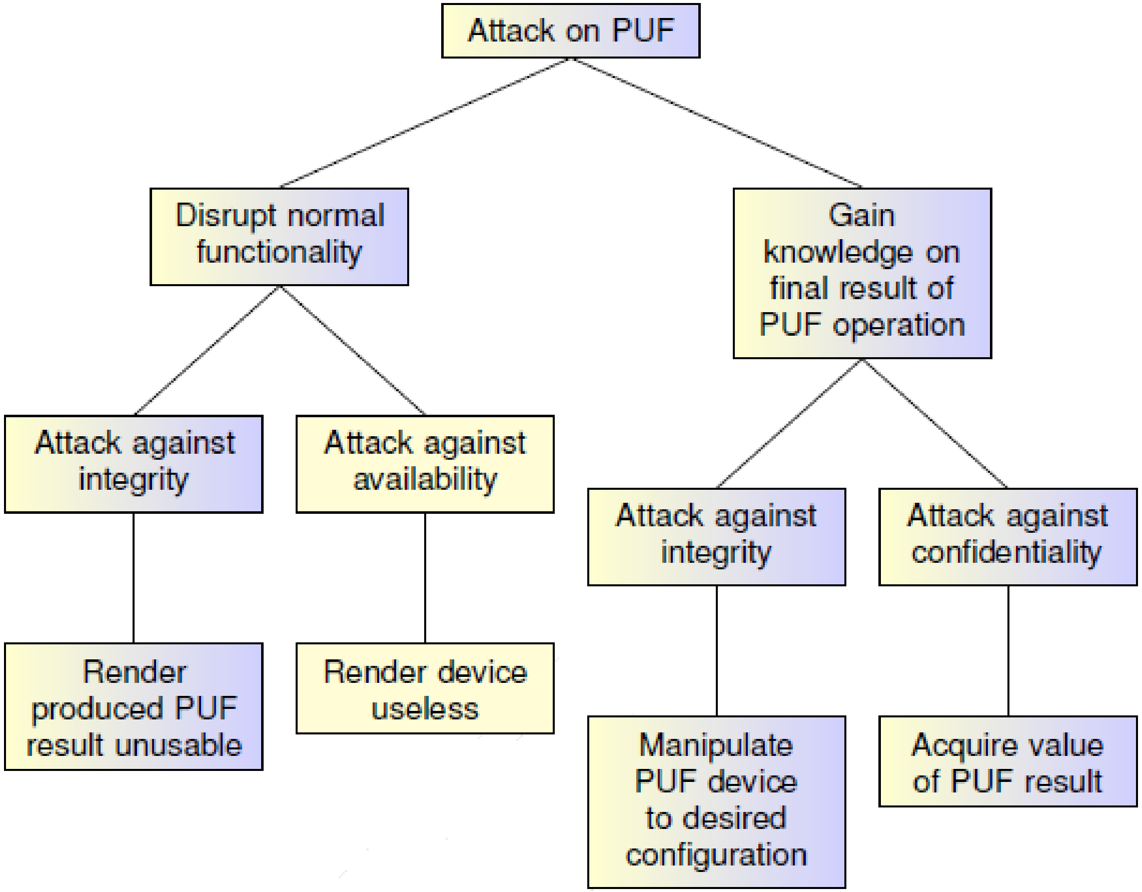
- Availability
- Integrity
- Confidentiality

Attacks as a means of protection

- Deniability
- Denial of access for third parties

- PUF as a single (unique) point of failure

Classification of attacks in the form of an attack tree


Attack on PUF
Disrupt normal functionality
Gain knowledge on final result of PUF operation
Attack against integrity
Attack against availability
Attack against integrity
Attack against confidentiality
Render produced PUF result unusable
Render device useless
Manipulate PUF device to desired configuration
Acquire value of PUF result

Classification of attacks in the form of an attack tree

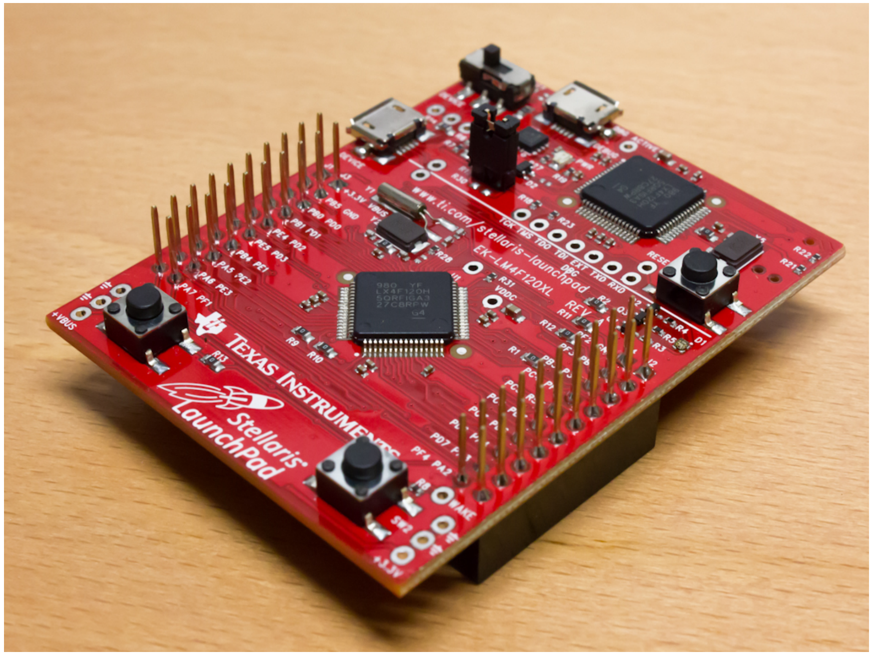
Advantages

- Classification according to previous criteria
- Means of calculating cost and appropriateness
  - Thus, also, a way to identify possible vulnerabilities and assess security
- Can lead to an estimation of *acceptable* risk and thus to assessment of PUFs as security mechanisms

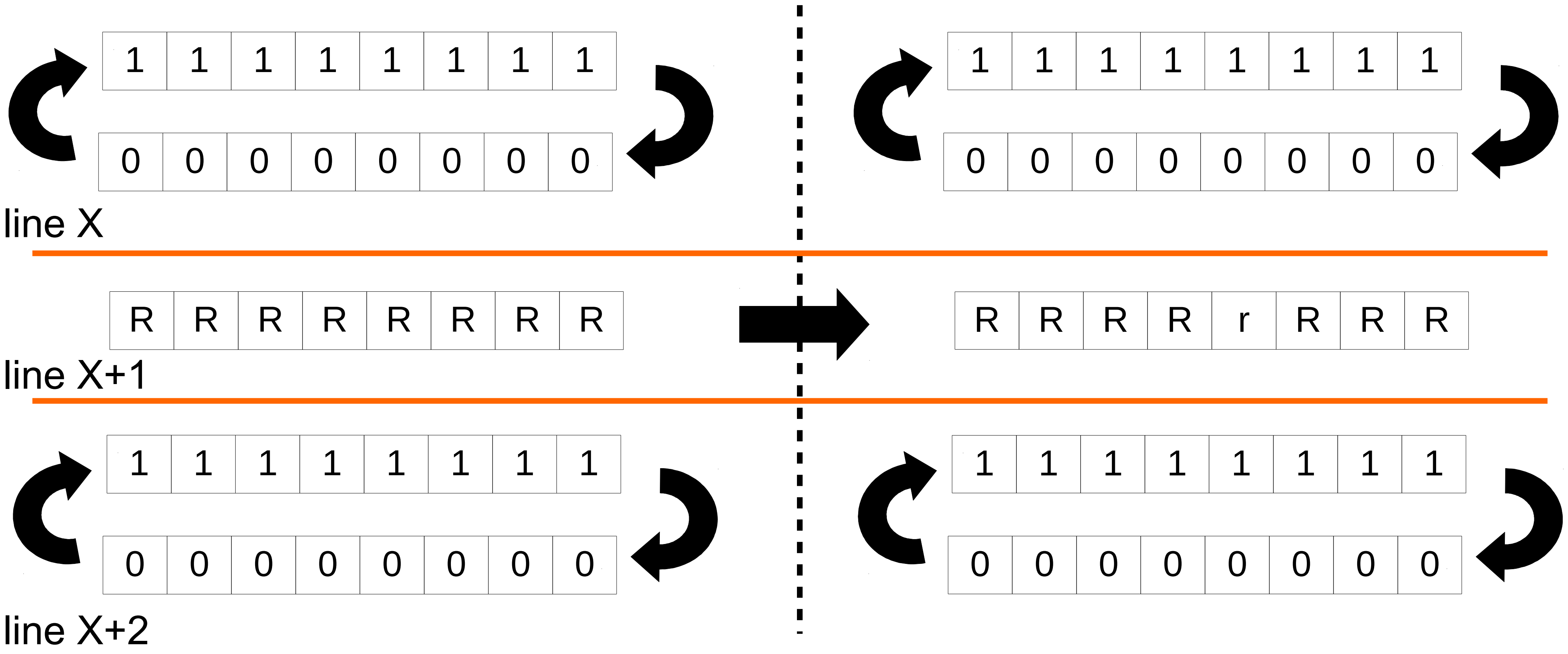
Implementation and testing of attacks

- Work in progress

- Selected attacks are being implemented against SRAM PUFs
  - Aging
  - Data remanence
  - Manipulation of neighbouring cells



Manipulation of neighbouring cells



1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0

1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0

line X

R	R	R	R	R	R	R	R

R	R	R	R	r	R	R	R

line X+1

1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0

1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0

line X+2

Row hammering on SRAM PUFs

Implementation and testing of attacks

- The attacks selected are easily implementable
  - They do not have extensive requirements
  - Are accessible to inexperienced attackers

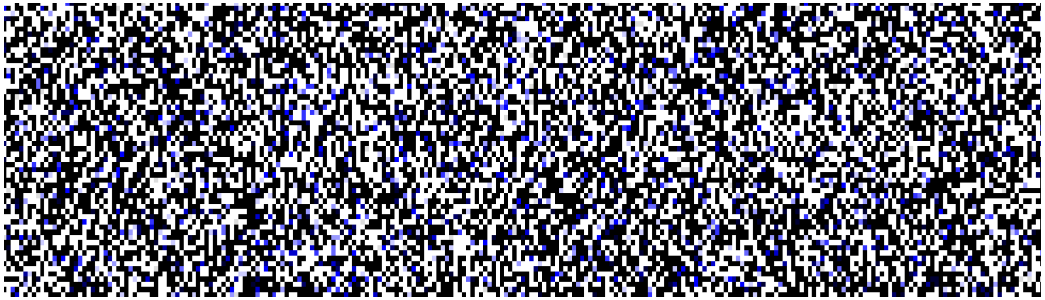
- They target SRAM PUFs which are already in production

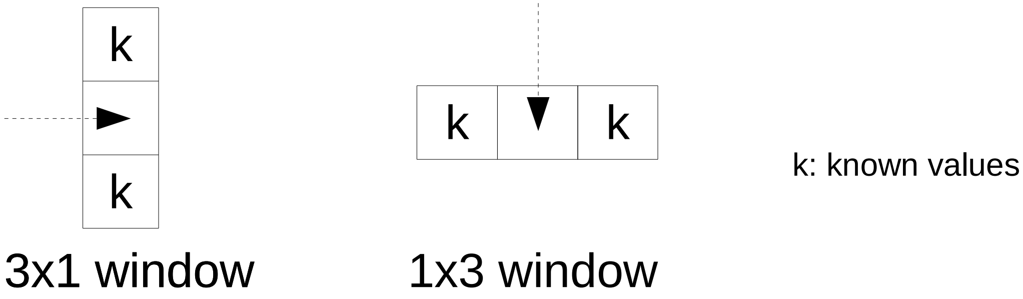
- Can therefore serve to determine if current PUF **products** can actually be considered as an *acceptable* security mechanism

Metrics to examine dependencies between nearby SRAM cells

Entropy in SRAM responses

- Has already been investigated for logical layout
- There is a need to prove or disprove if SRAM PUFs can be modelled based on the response of neighbouring cells (+error correction)



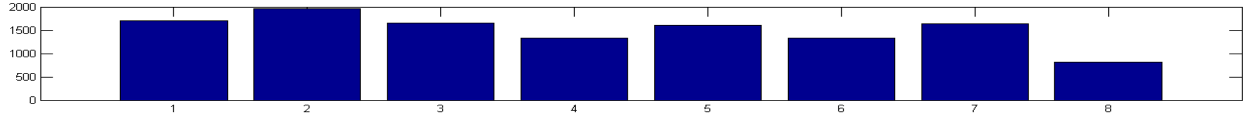


- We examine the physical layout
  - Estimate the value of a central cell in windows of different sizes, when values of all other cells are known
  - Data obtained by TU Berlin
  - 2 chips tested

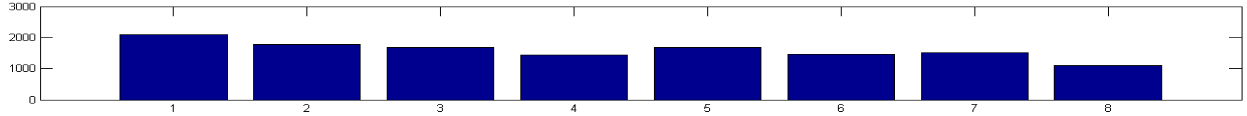
Metrics to examine dependencies between nearby SRAM cells

3x1 window

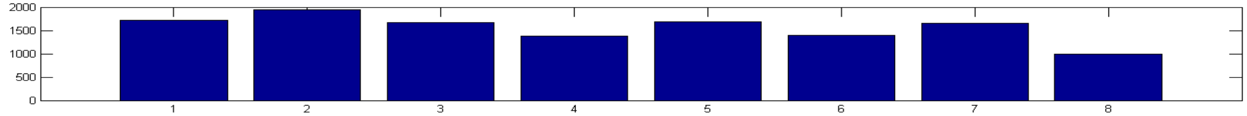
Chip 1 physical layout



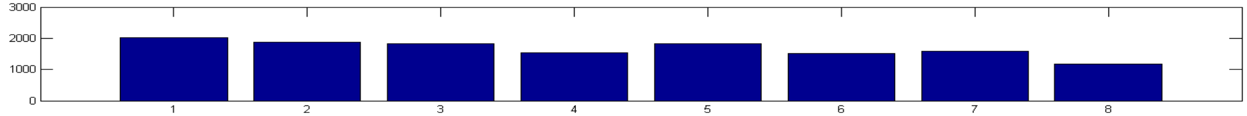
Chip 1 logical layout



Chip 2 physical layout

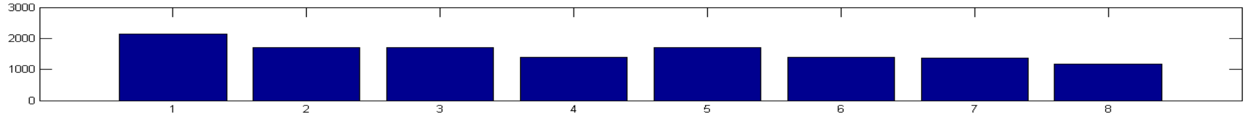


Chip 2 logical layout

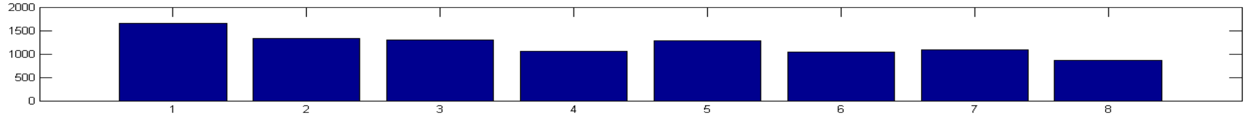


1x3 window

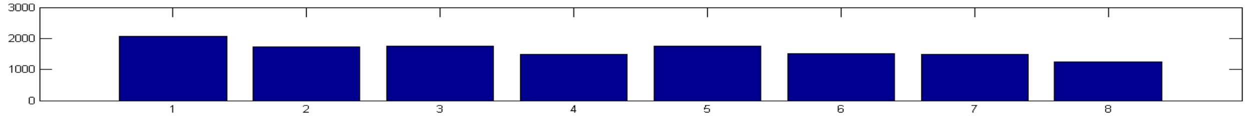
Chip 1 physical layout



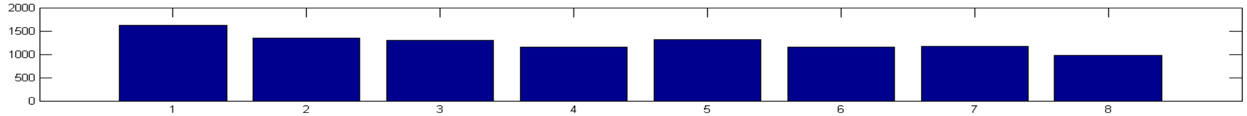
Chip 1 logical layout



Chip 2 physical layout



Chip 2 logical layout



First experiments indicate a good entropy

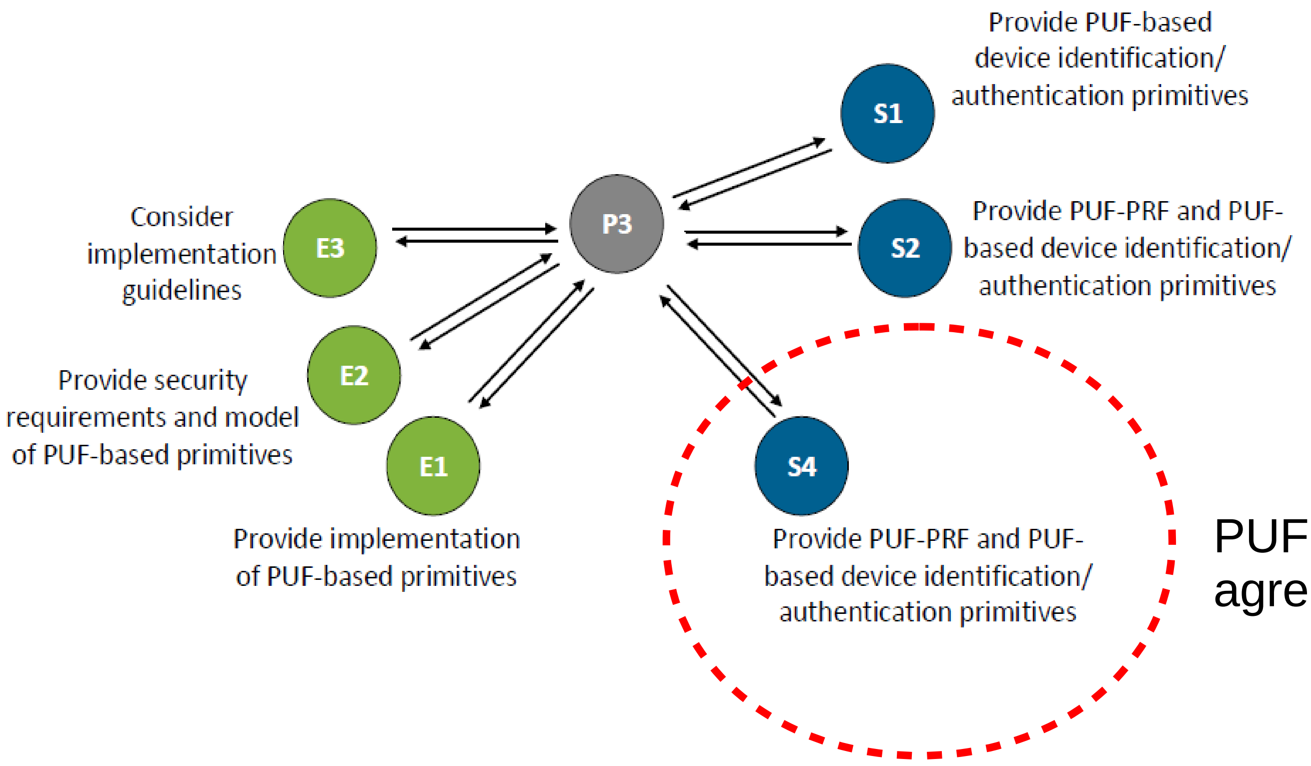
Future objectives

- Complete assessment of current PUF solutions regarding their security
- Assess and improve the error correction mechanism
- Identify possible new PUF solutions
- Pick and implement better PUF solutions and protocols

Future collaborations

- PUF-based attestation (internal)
- Novel PUF solutions (external)
- Side-channel attacks on PUFs (internal & external)
- PUF-based communication protocols (internal & external)

Future collaborations


Provide PUF-based device identification/ authentication primitives
S1
Consider implementation guidelines
E3
P3
S2
Provide PUF-PRF and PUF-based device identification/ authentication primitives
E2
Provide security requirements and model of PUF-based primitives
E1
S4
Provide implementation of PUF-based primitives
Provide PUF-PRF and PUF-based device identification/ authentication primitives

PUF-based key agreement