Physically Unclonable Functions
PUFs
Principle, Advantages, Limitations

Jean-Luc DANGER

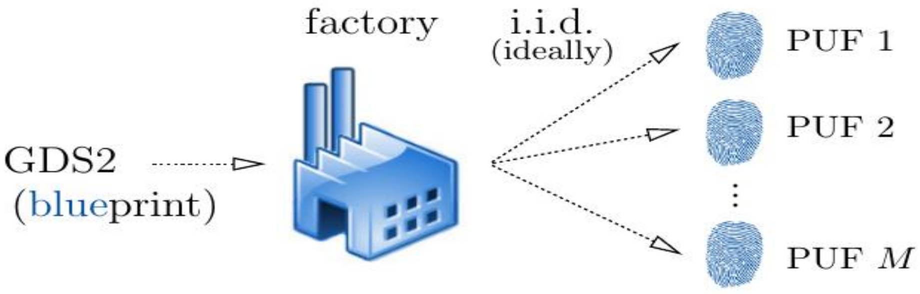
December 2019

Outline

- What and Why a PUF ?
- PUF Architectures
- PUF Reliability
- PUF Security
- Conclusions

Physically Unclonable Function: PUF

- **Function returning the fingerprint of a device**
  - Physical function,
  - which exploits material randomness, during fabrication (mismatch)
  - and is unclonable: same structure for each device

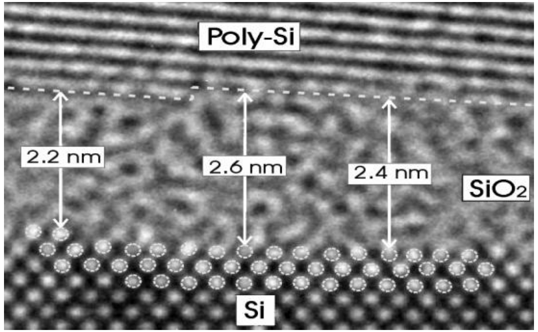


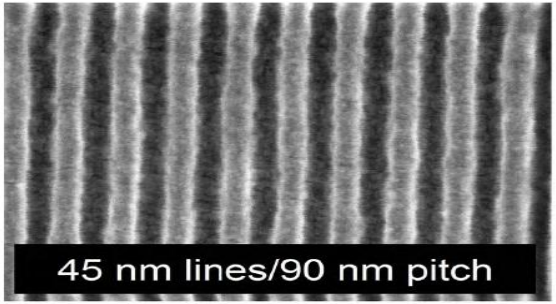
PUFs are instanciations of blueprints by a fab plant

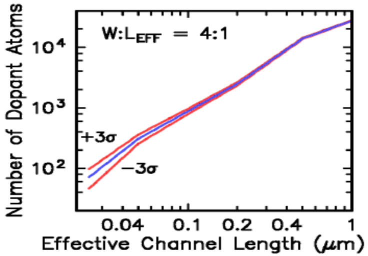
a PUF ID is unique to each device

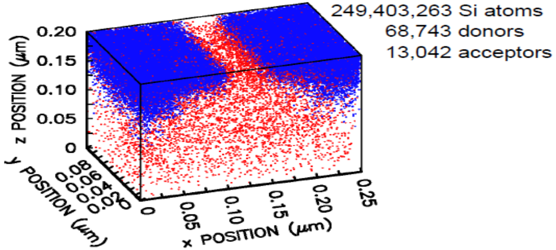
Mismatch: CMOS process variation

- **Examples**
  - Oxide thickness
  - Metal line edge roughness
  - Random dopant fluctuation









[D. J. Frank, et al., 1999 Symp. VLSI Tech.]

PUF Fingerprint: 2 types

- **List of pairs challenges / responses**

Many challenges =>The PUF is "strong"

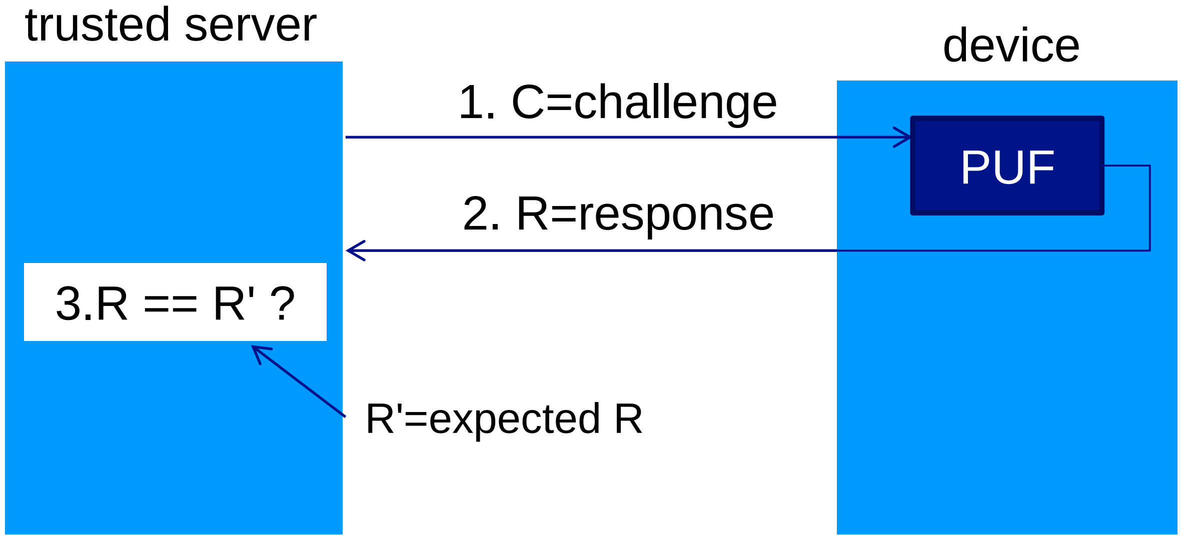
- **Unique identifier**

No challenges =>The PUF is "weak"

can be used as a
cryptographic key !

Main function: Authentication

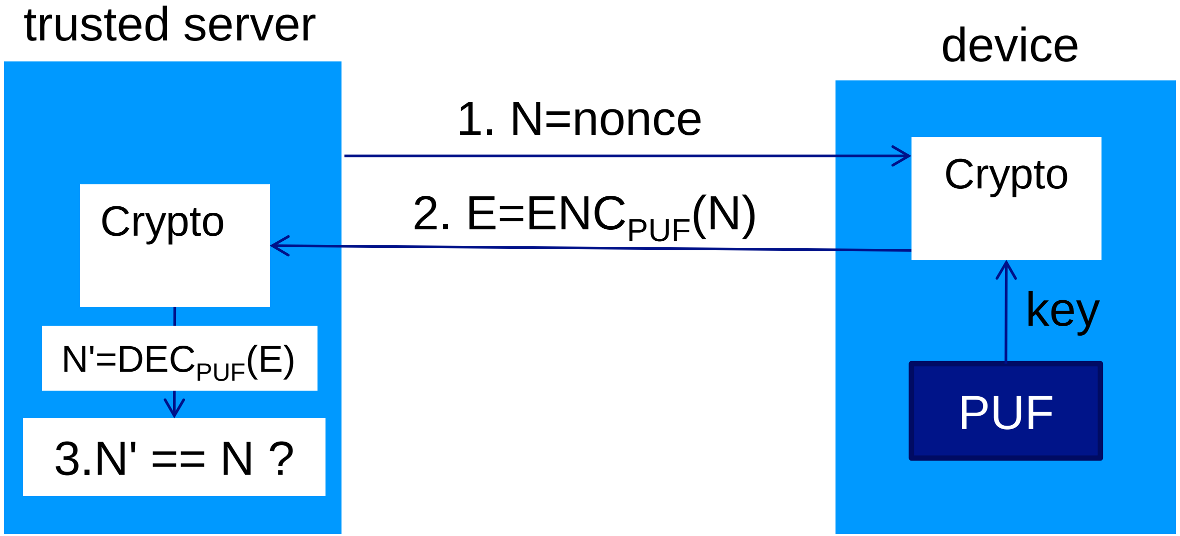
- **Use of Challenge-Response => CRP protocol**



The challenge is never sent twice to avoid replay attacks

Main function: Authentication

- **Use of the ID as a key => cryptographic protocol**



The nonce is never sent twice to avoid replay attacks

Advantages of PUF vs Non Volatile Memory "NVM"

- **PUF is self contained**
  - NVM has to be programmed with an ID, and can be tampered
- **Not clonable**
  - PUF has the same structure, NVM can be reverse engineered
- **Feasible in standard CMOS process**
  - NVM requires a specific process

Many advantages compared to an identifier stored in a NVM memory !

Important Properties to meet

related to **entropy**

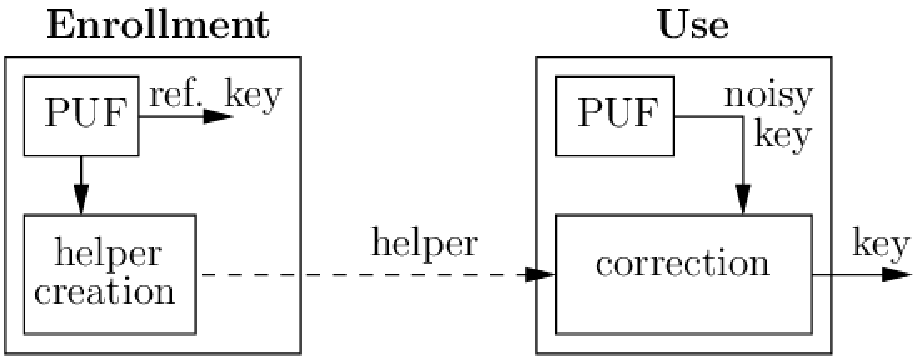
- **Reliability**
  - The PUF responses are unreliable : 1 to 15% of Bit Error Rate
- **Randomness**
  - The PUF responses can be biased: Pr(1) ≠ Pr(0)
- **Uniqueness**
  - Two devices should not have the same ID.
- **Security against attacks**
  - 2 main types: Modeling and Physical attacks
- **Latency**
- **Complexity**

PUF Application examples

- **IP block protection**
  - The IP can run only on the authorized device
- **Secure Boot**
  - The OS is loaded and deciphered only on the authorized device
- **Safe guard**
  - The data are ciphered before being stored in an untrusted device
- **RFID / NFC tag**
  - A product can be authenticated and traced (anti-counterfeiting)

PUF: Two phases of use

- **1. Enrollment**
  - To do only once after manufacturing
  - To get a "reference PUF" and a "helper data" to get it reliable
- **2. Usage or Reconstruction**
  - To obtain the PUF ID
  - The "helper data" is used to correct errors



Outline

- What and Why a PUF ?
- PUF Architectures
- PUF Reliability
- PUF Security
- Conclusions

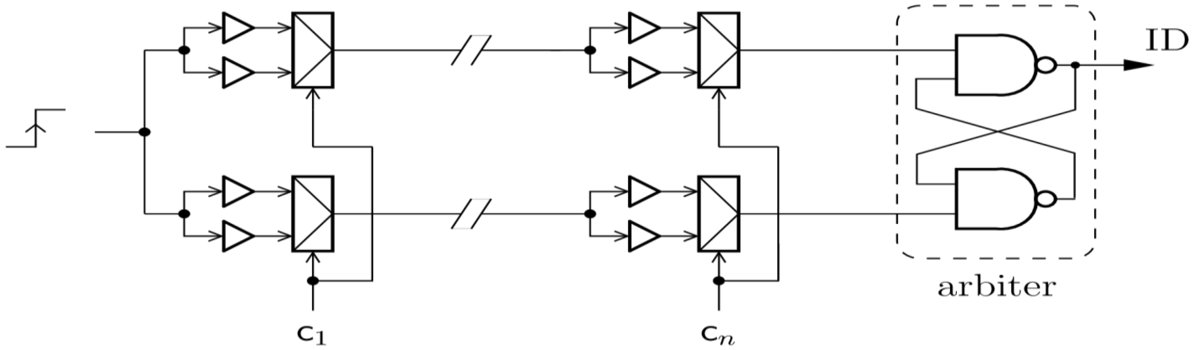
Main Classes of PUF in silicon

Two main types

- **Delay-PUF**
  - Exploits the delay difference between 2 identical delay lines.
- **Memory-PUF**
  - Exploits the difference between two inverters in an SRAM cell
- **Many other types in the literature:**
  - GLITCH PUF
  - MECCA PUF
  - VIA PUF
  - RRAM PUF
  - TERO-PUF
  - …

Delay-PUF: Arbiter-PUF

- **Delay difference between two identical pathes:**

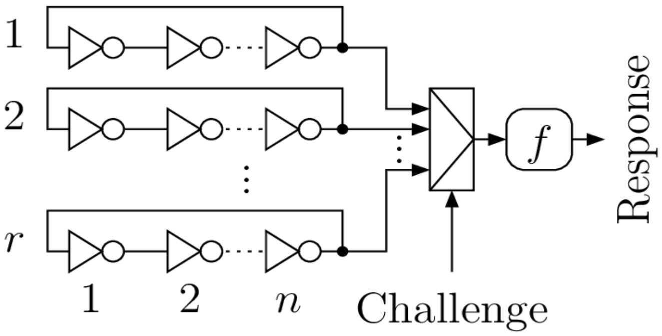


  - "Strong" PUF: many challenges for CRP protocol
  - Sensitive to Mathematical attacks: Modeling Attacks

Delay-PUF: RO-PUF

- **Frequency difference between two identical Ring Oscillators:**

Ring-oscillator PUF (RO-PUF):
(r rings of n inverters)

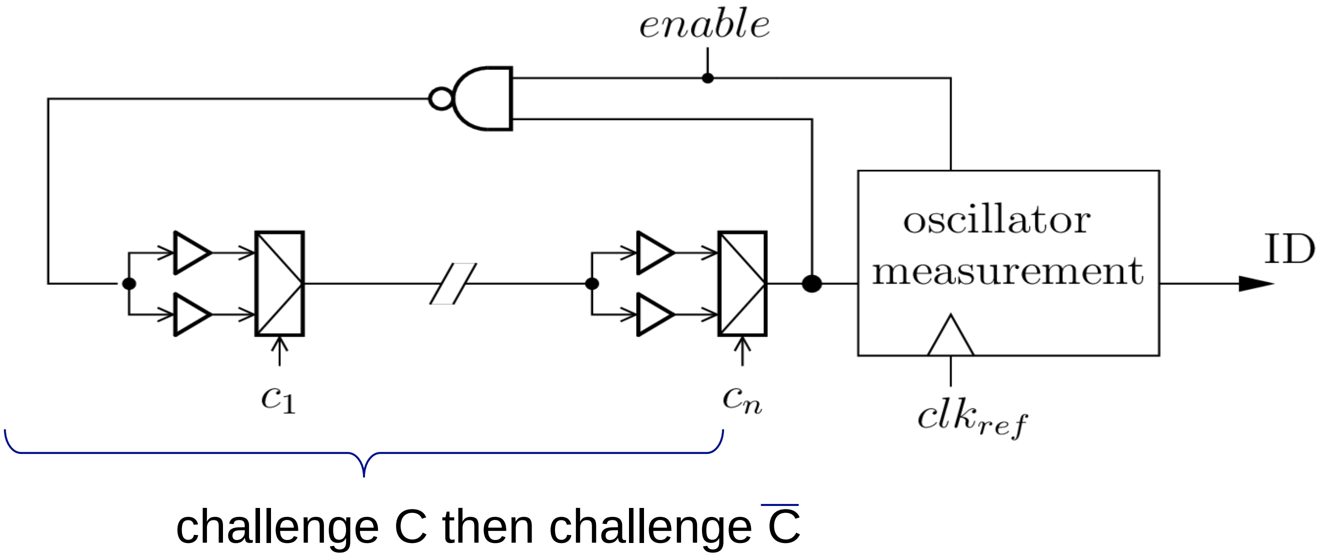


Rationale:

Challenge selects a pair i, j, $1 \leq i \neq j \leq r$.

Response is 1 if RO_i rotates $f$aster than RO_j, and 0 otherwise.

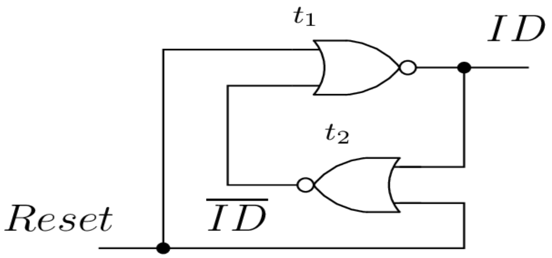
Delay-PUF: Loop-PUF

- **Frequency difference between a controlled Ring Oscillator driven by two complementary challenges**
- 

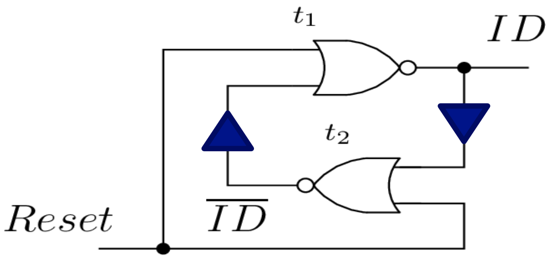
Memory-PUF: Latch-PUF

- **Imbalance between two elements of a latch**

A:

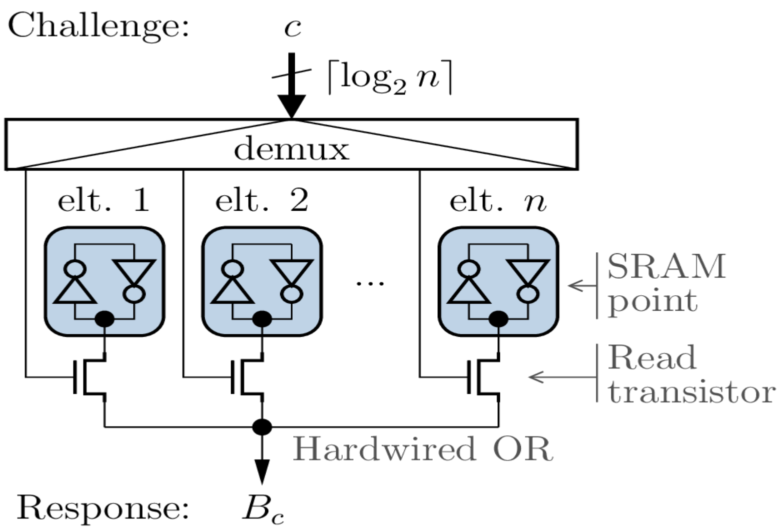


B:



Memory-PUF: SRAM-PUF

- **Imbalance between two inverters of an SRAM cell**



Outline

- What and Why a PUF ?
- PUF Architectures
- PUF Reliability
- PUF Security
- Conclusions

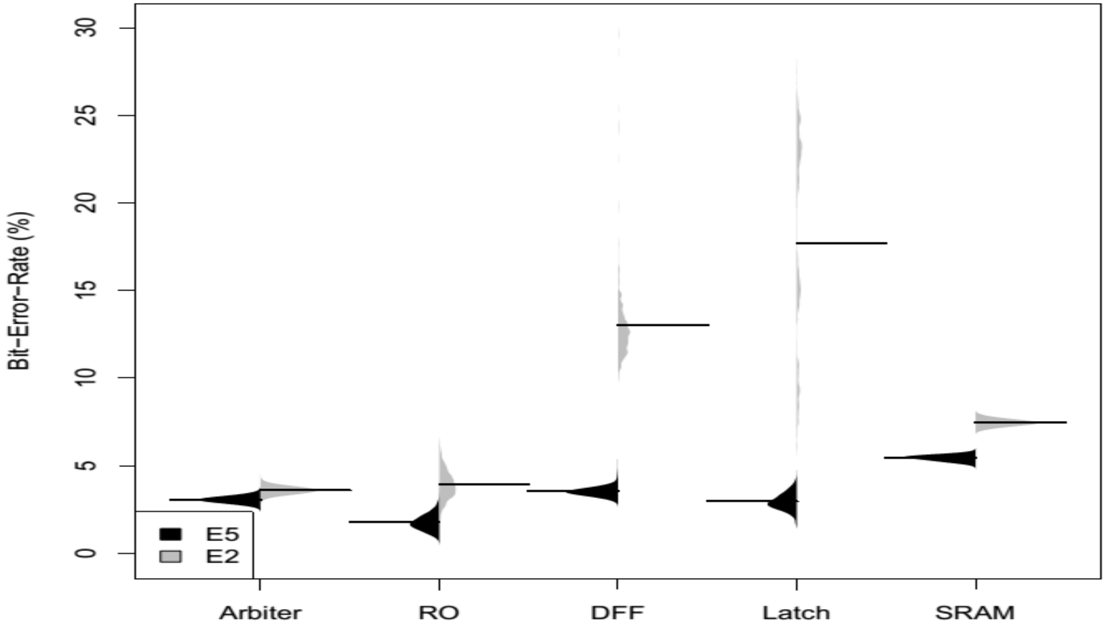
Main Properties to meet

Relatively less problematical

- **Reliability**
  - The PUF response is sensitive to:
    - Noise
    - Environmental change T°C, Vdd
    - Aging
- **Entropy**
  - Inter device: Uniqueness: Each device must have a unique fingerprint
  - Intra device: Randomness: as many bits at 0 and 1
- **Security**
  - Robustness against physical attacks: SCA, FIA
  - Robustness against modeling attacks

reliability estimate

BER = 1- steadiness



Big Lack of Reliability !

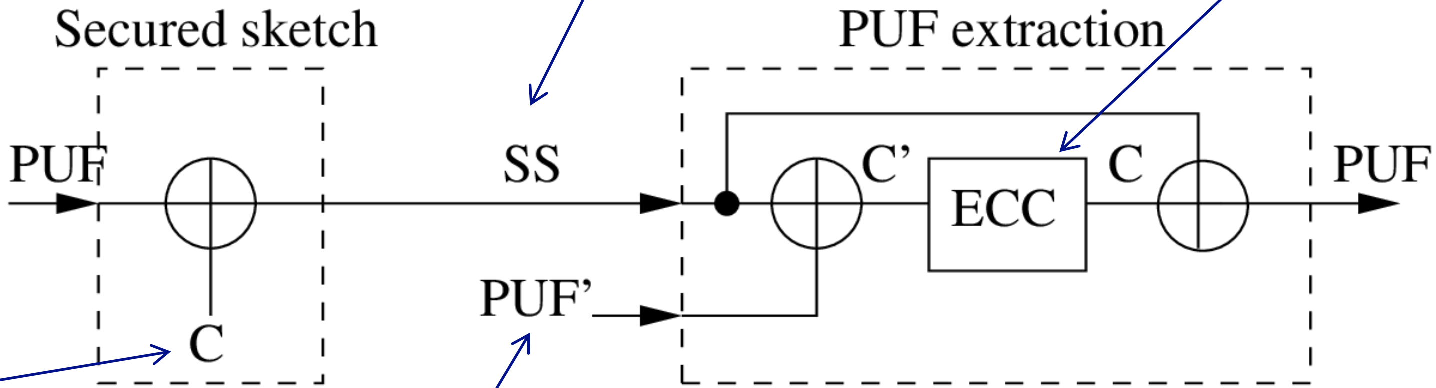
Black: 25°C
Grey: -40C

Results from the european "UNIQUE" project

Secure sketch to correct PUF

Helper Data = "Secure Sketch"

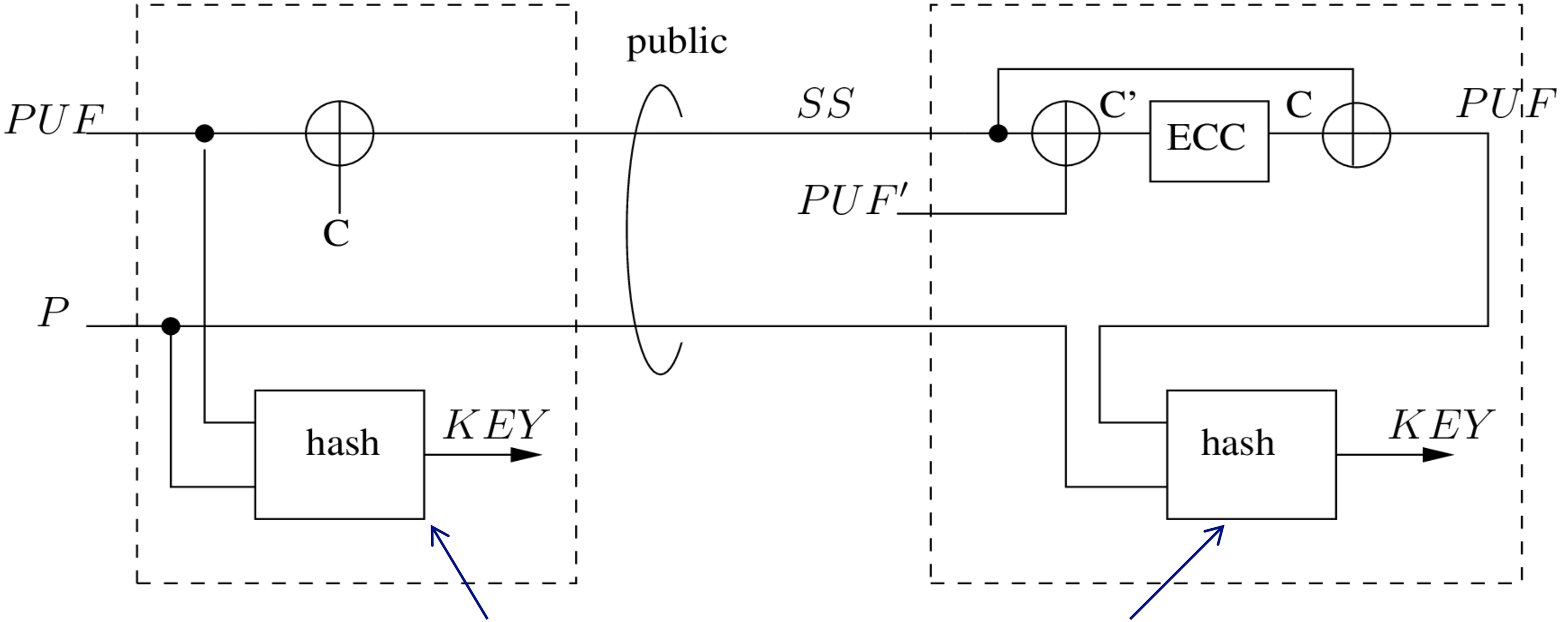
ECC decoder



Codeword

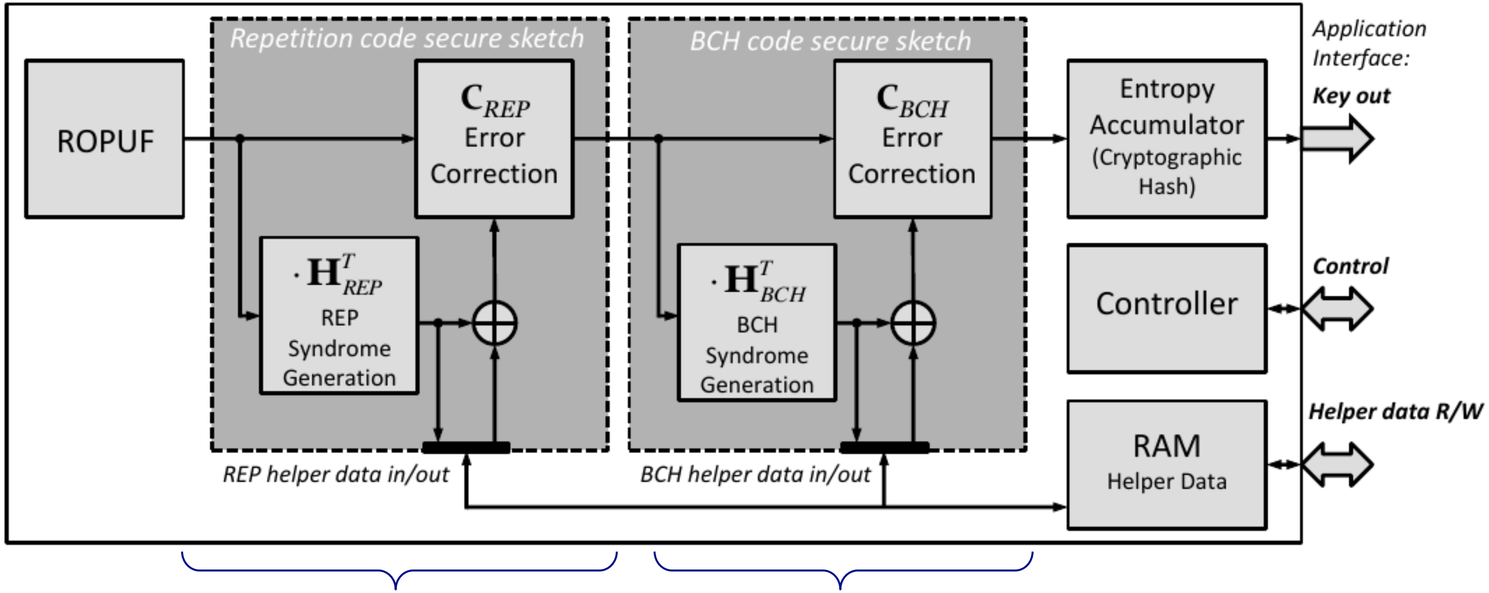
PUF'=PUF+error

Fuzzy extraction for Key generation



The Key changes with Hash(P||PUF)

Example: PUFKY



Concatenated code = Repetition code + BCH code

BER: 10^{-1} => 10^{-4} => 10^{-9}

Reliability enhancement by filtering

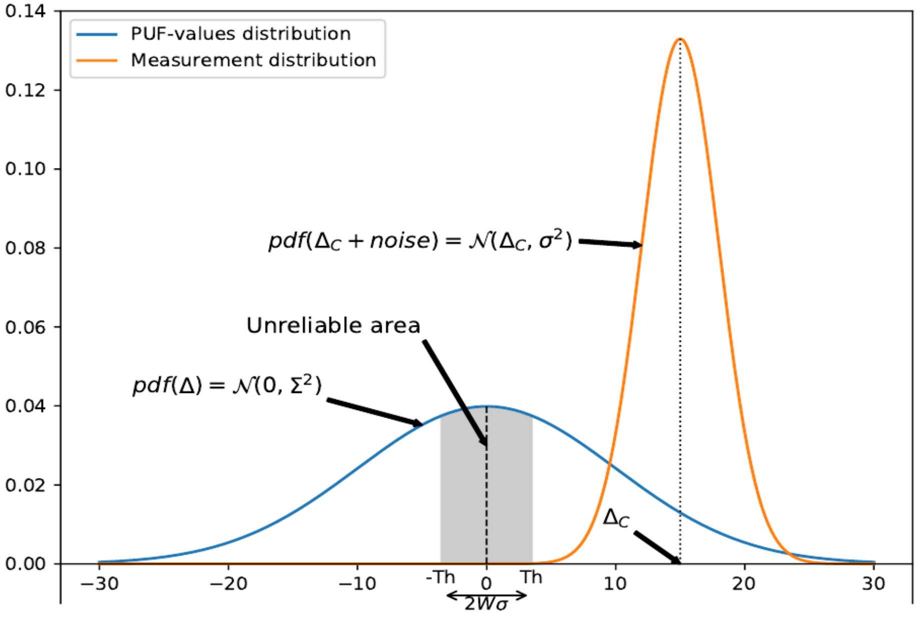


Figure 7: Unreliable area vs Distributions of Δ and noise

Applies for delay PUF having the precise delay information

Bit **unreliable** ⇔ |delay| < Th

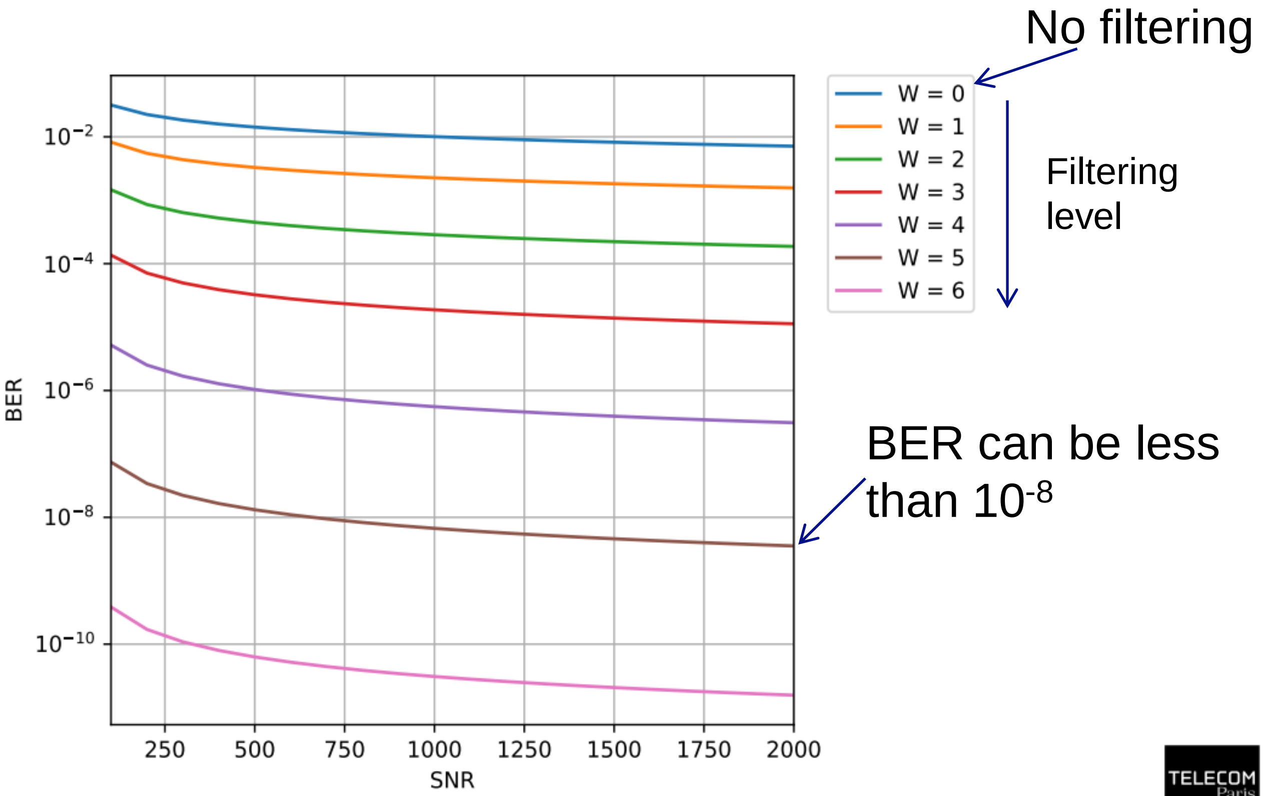
Th= Ws

The bits in the unreliable area are **discarded**

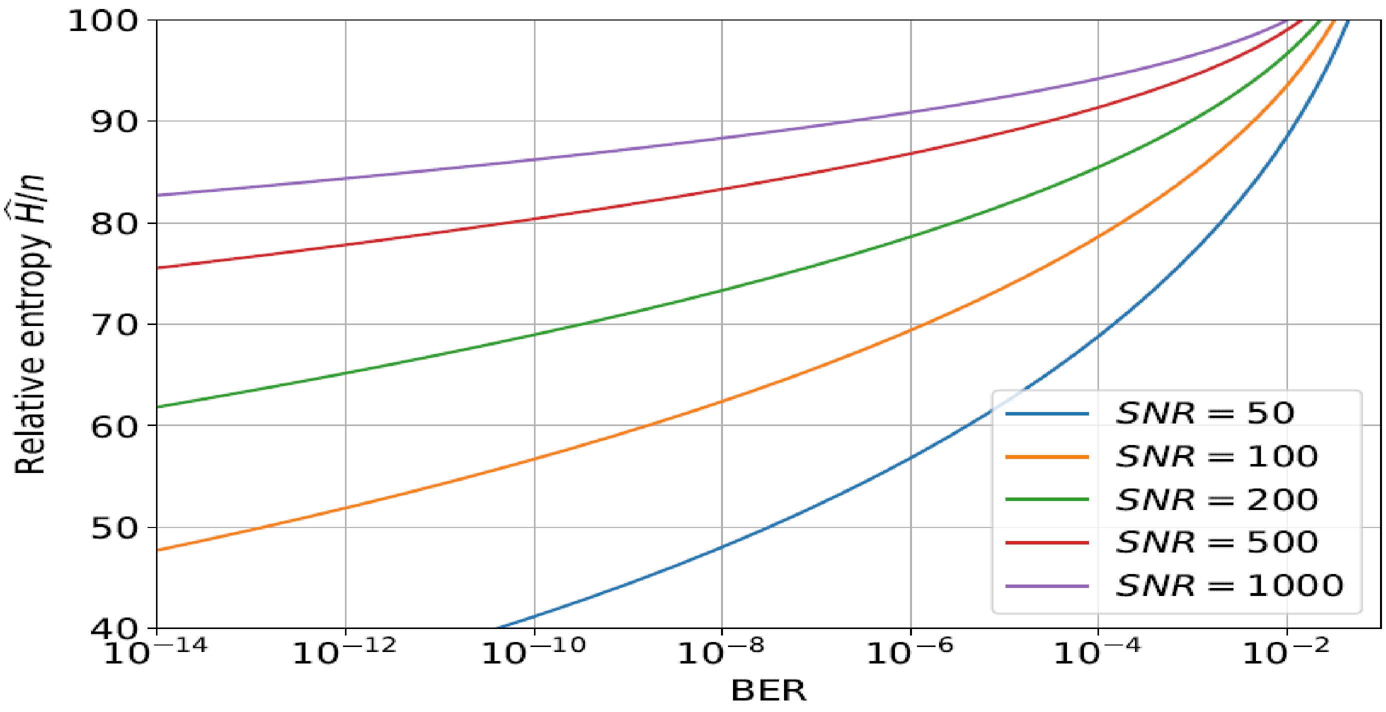
The helper data indicates the unreliable bits, but gives **no information** on the bit value

Delay PUF: filtering out unreliable challenges

RO- PUF and Loop-PUF gives the reliability level
=> Unreliable challenges can be filtered out



Entropy loss after bit filtering



Need for standard tests and/or stochastic model

Active discussion at ISO sub-committee 27: (**ISO 20897**)

ISO IEC

ISO/IEC JTC 1/SC 27/WG 3 **N1233**

REPLACES:

ISO/IEC JTC 1/SC 27/WG 3
Information technology - Security techniques - Security evaluation, testing and specification
Convenorship: AENOR, Spain, Vice-convenorship: JISC, Japan

DOC TYPE: working draft

TITLE: Text for ISO/IEC 1st WD 20897 — Information technology — Security requirements and test methods for physically unclonable functions for generating non-stored security parameters

SECURE-IC TELECOM Paris IP PARIS

Outline

PUF Attacks

- **Reverse Engineering Attack**
  - Virtually impossible: same bueprint
- **Brute force**
  - Virtually impossible to store all challenge/responses (CRP)
- **Replay**
  - Sniffing CRPs and play them back
  - Can be countered at protocol level
- **Mathematical**
  - Reconstruct the PUF model: Modeling Attack (CRP only)
- **Physical attack**
  - Side-channel
  - Faults

Main threats

Modeling Attacks

- **Based on Machine Learning algorithms**
  - Take advantage of equations defining the Response from the Challenge
  - Very powerful to attack delay-PUFs
  - Applies only to CRP protocol
- **Countermeasures**
  - Combination of delay-PUFs
  - Do not used PUF in CRP protocol but for key generation

Side-Channel Attack

- **Observation of raw oscillating frequency**
  - Applies to RO-PUF and Loop PUF
  - Countermeasures:
    - RO-PUF: interleave the placement of the RO banks
    - RO and Loop PUF: Use random sequential measurement
- **Attack on the Fuzzy extractor**
  - Simple Power Analysis has been carried out on a FE
  - Template attacks have been implemented on ECC
  - Countermeasures: masking, as cryptographic blocks

Enhanced SCA

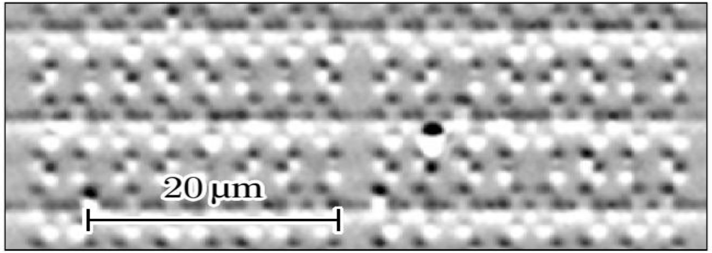
- **Combination with Machine Learning algorithms**
  - Use of noise distribution of the arbiter PUF
  - Use unsupervised ML- techniques [2]
    - SCA is performed first
    - The ML technique proposes a model for classification (like for instance the "k-means" algorithm).

Fault Injection Attack

- **Applies on Delay PUF**
  - Pulse attack (laser, EMI,…)
    - The PUF output is forced
  - Harmonics attack
    - RO PUF: The PUF frequency can be locked on external EM carrier injection
- **Countermeasures**
  - Detection
    - Use embedded sensors to detect disturbances
    - Measure online the entropy of the PUF response

PUF invasive attack

- **Applies on SRAM PUF**
  - Laser stimulation techniques exploiting the Seebeck effect
    - the off-transistor becomes to conduct under laser shot
    - Provides a current increase
  - Attack performed on AVR microcontrollers



SRAM content read out

Outline

- What and Why a PUF ?
- PUF Architectures
- PUF Reliability
- PUF Security
- Conclusions

Conclusions

- **A specific fingerprint for each IC**
- **Used for authentication and key generation**
- **Use two phases: enrollment (with helper data) + reconstruction**
- **Main advantages**
  - Self-generated by the device
  - No reverse engineering and limited tampering
- **Main limitations**
  - Lack of reliability
    - Necessary post-processing
  - Can be attacked physically and mathematically
    - Protections required
- **ISO Standard for PUF validation**

A few PUF providers and users