Soft Decision Error Correction for Compact Memory-Based PUFs using a Single Enrollment

Vincent van der Leest (Intrinsic-ID)
Bart Preneel (KU Leuven and IBBT)
Erik van der Sluis (Intrinsic-ID)

CHES workshop 2012, Leuven
Tuesday, September 11, 2012

Introduction

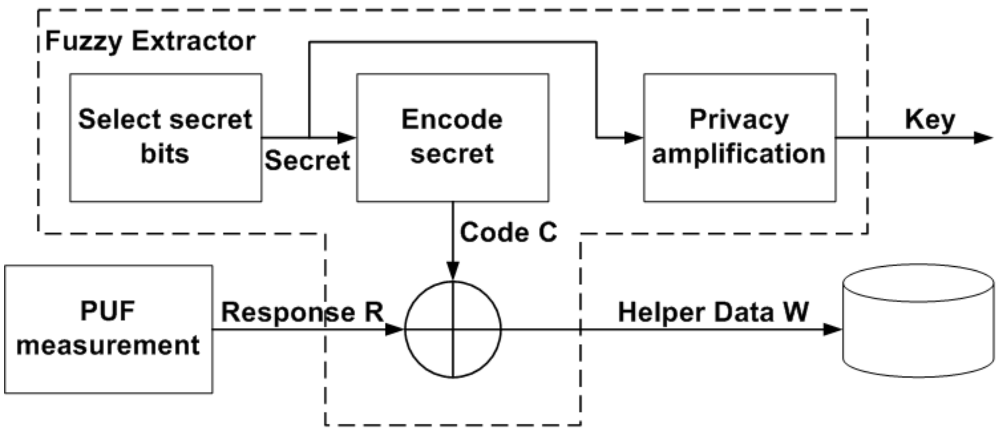
- PUFs
  - IC identification based on physical characteristics
  - Measurements are noisy and require error correction
- Use Case: Secure Key Storage
  - Error correct noisy PUF to produce stable key
- Error correction
  - Overhead on PUF size, efficient codes are required
  - Soft decision decoding is more efficient than hard decision
  - Soft decision algorithms with **multiple** measurements exist
  - We introduce soft decision using a **single** measurement

Memory-based PUFs

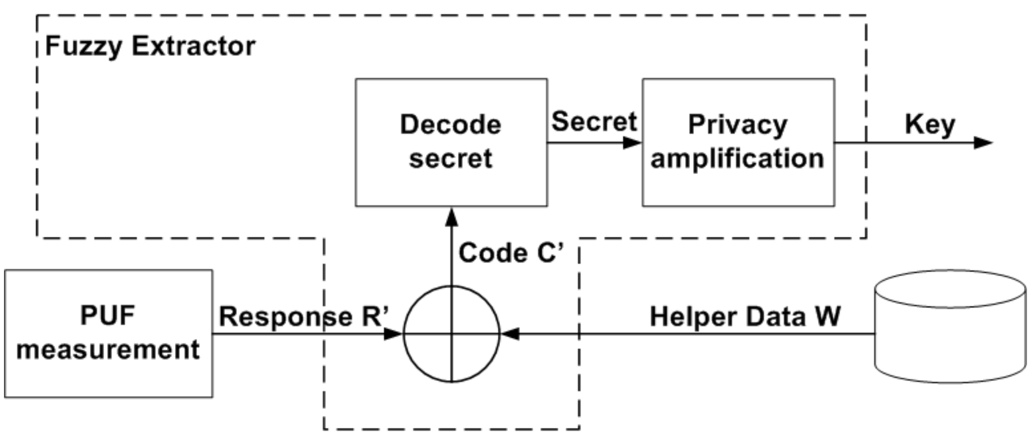
- Memory-based PUFs: deriving PUF fingerprint from start-up pattern of (standard-cell) memory in IC
- Examples: SRAM, D Flip-Flop, Latch, Buskeeper…
- Startup patterns are required to be:
  - **Robust** (stable under different operating conditions)
  - **Unique** (random and unpredictable)
- Memory-based PUF used here: SRAM PUF

Use Case: Secure key storage

Enrollment



Reconstruction



In secure environment:

- "Program" key
- Derive helper data
- Store helper data

During operation:

- Retrieve secret key using helper data and PUF response
- Secret reproducible with error correction

Soft decision decoding: state of the art

- Soft decision decoding for memory-based PUFs*:
  - **Enrollment:**
    - Perform **multiple** measurements
    - Derive error probability of each PUF bit
    - Store error probability with helper data (= soft information)
  - **Reconstruction:**
    - Use error probabilities as confidence level for each bit
    - Less PUF bits required to reconstruct secret

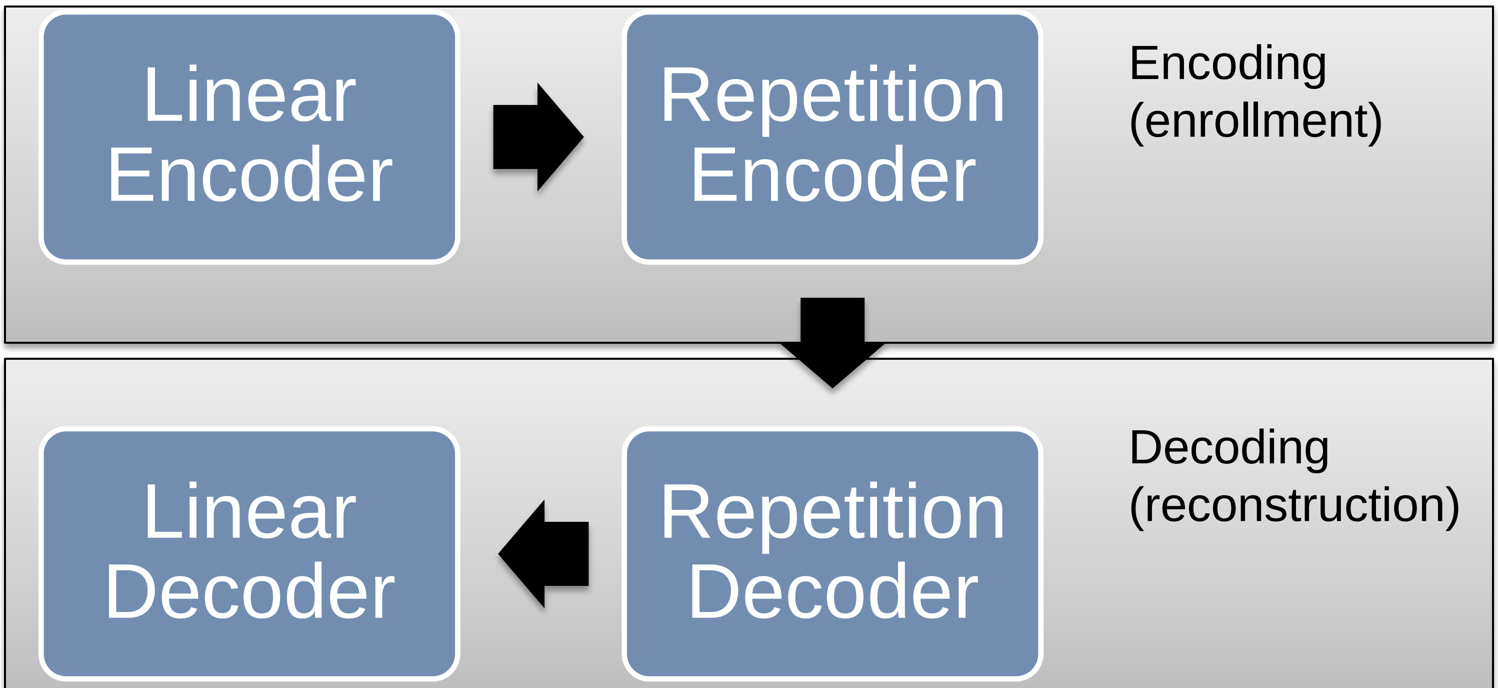
* [Maes-Tuyls-Verbauwhede'09]

Motivation for new construction

- Using multiple enrollment measurements leads to:
  - Requiring **non-volatile memory** during enrollment
  - Growing **footprint** with number of measurements
  - Additional **enrollment time** in production line
- Drawbacks make soft decision decoding for PUFs practically and commercially inapplicable

Our proposal (high level)

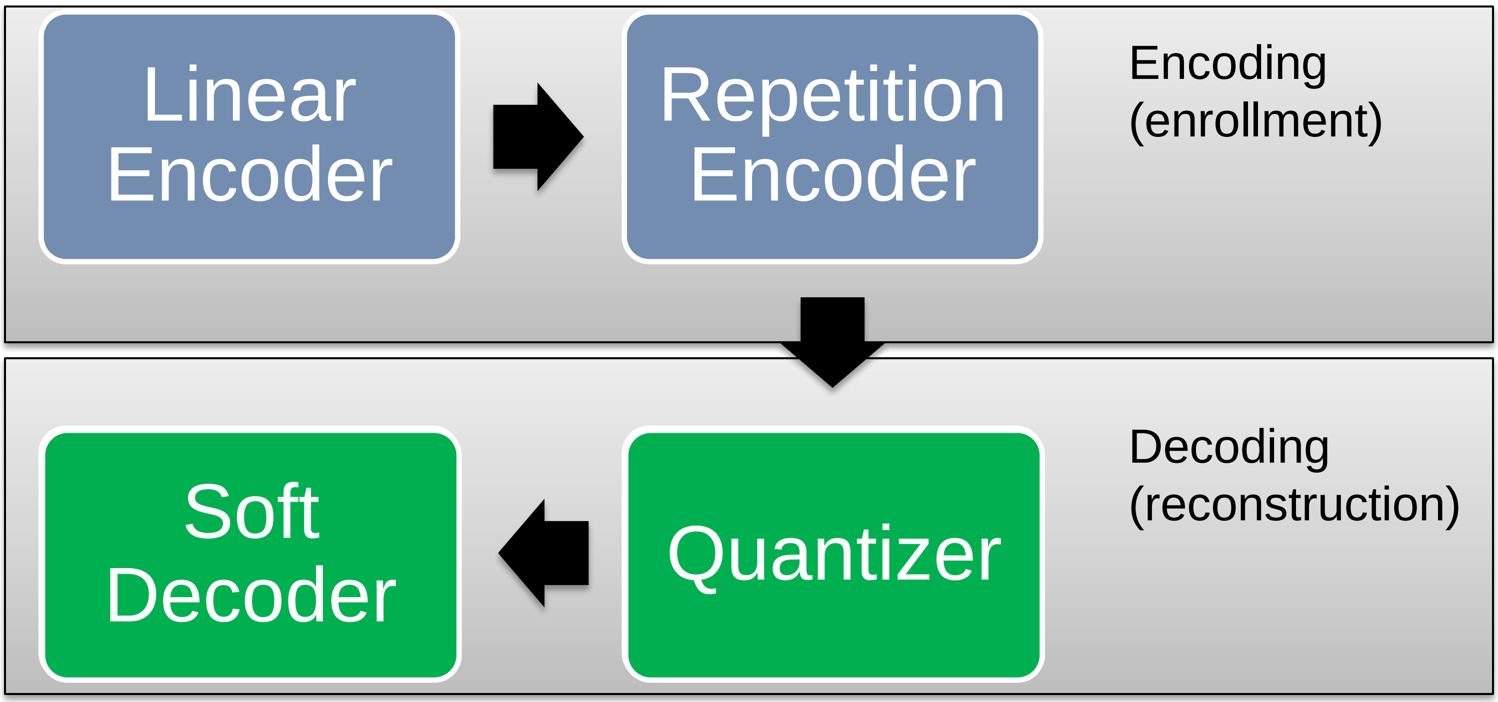
- Hard decision decoding using concatenated codes*



* [Bösch-Guajardo-Sadeghi-Shokrollahi-Tuyls’08]

Our proposal (high level)

- Soft decision decoding using concatenated codes

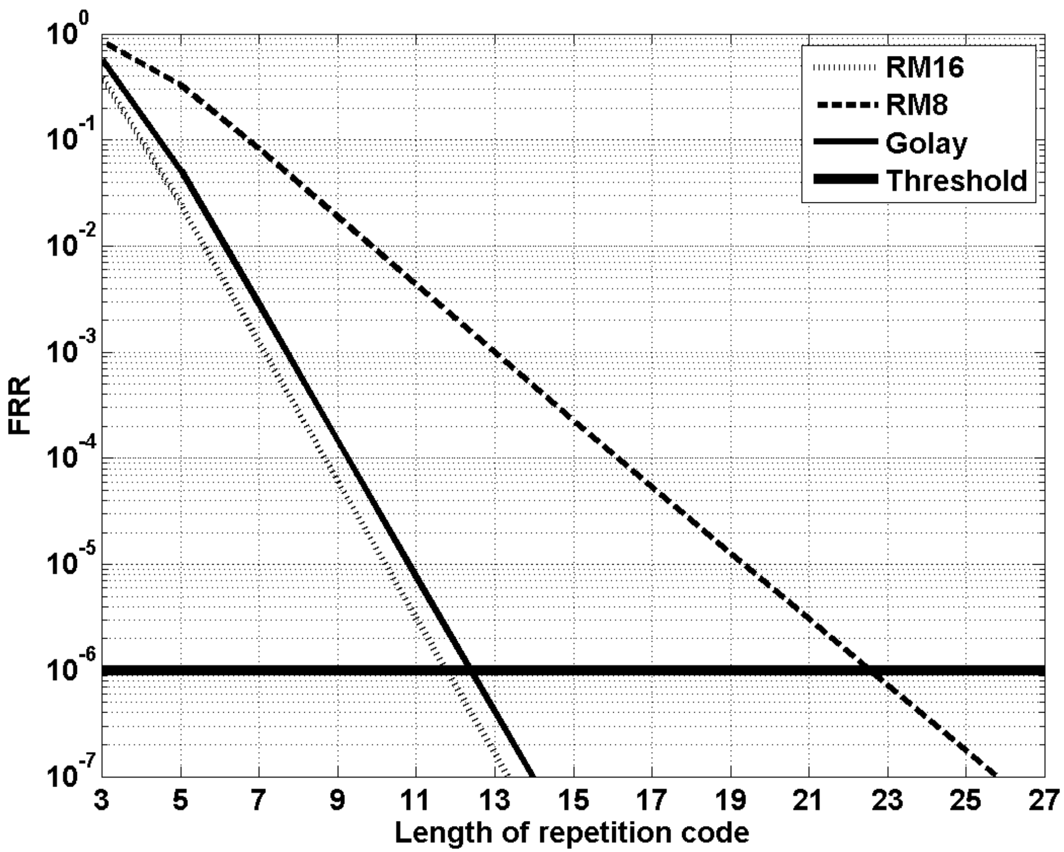


- Quantizer: only a **single** enrollment measurement required

Soft decoder examples

- Decoders with efficient hardware implementation
- Brute force decoder:
  - Codes with limited set of codewords
  - Calculate Euclidean Distance input to all codewords
  - Select most likely codeword for decoding
  - Examples: Reed-Muller [16,5,8] and [8,4,4]
- Hackett decoder:
  - Golay [24,12,8] decoder with soft input
  - Hard decision decoding with 8 different input patterns
  - Input patterns selected based on soft information
  - Most likely output selected based on Euclidean Distance

Calculating hard decision performance



Hard decision FRR can be calculated based on length of repetition code (equations available for concatenated codes)

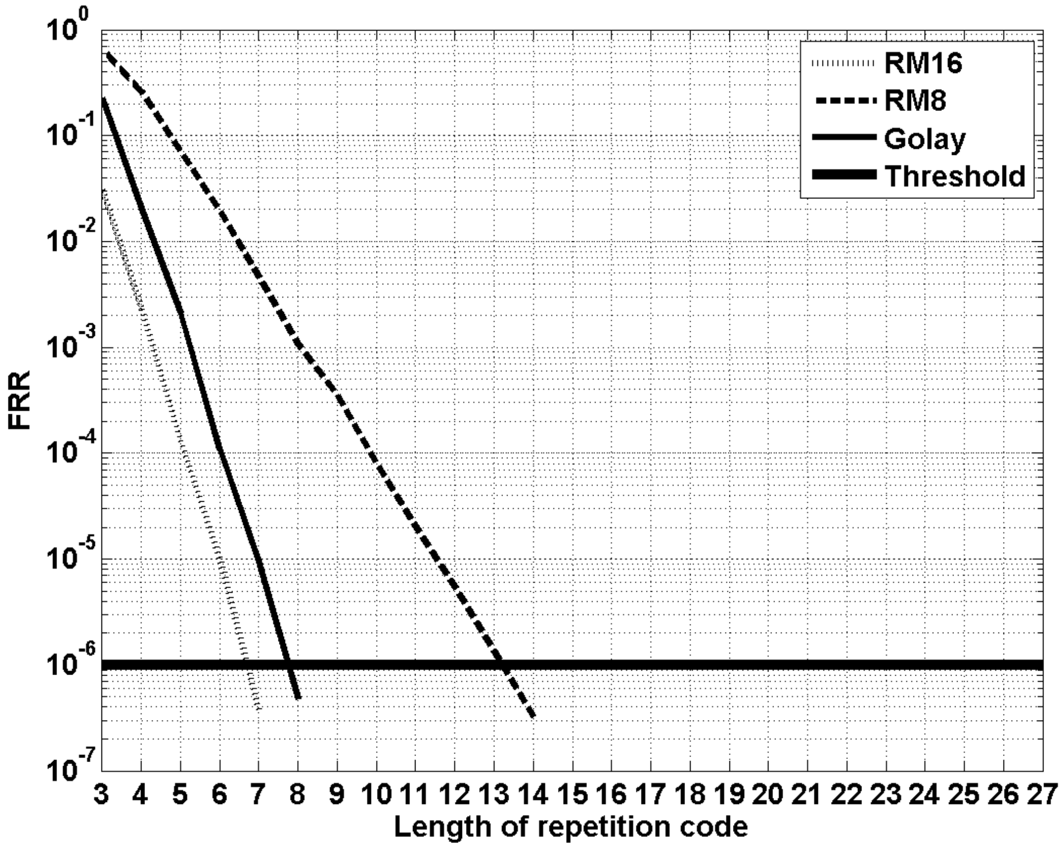
Based on results, codes require repetition length:

RM[16,5,8] : 13 bits

RM[8,4,4] : 23 bits

Golay[24,12,8] : 13 bits

Simulating soft decision performance



No equations available for calculating FRR of soft decision codes → simulations performed

Based on simulations, codes require repetition/ quantizer length:

RM[16,5,8] : 7 bits

RM[8,4,4] : 14 bits

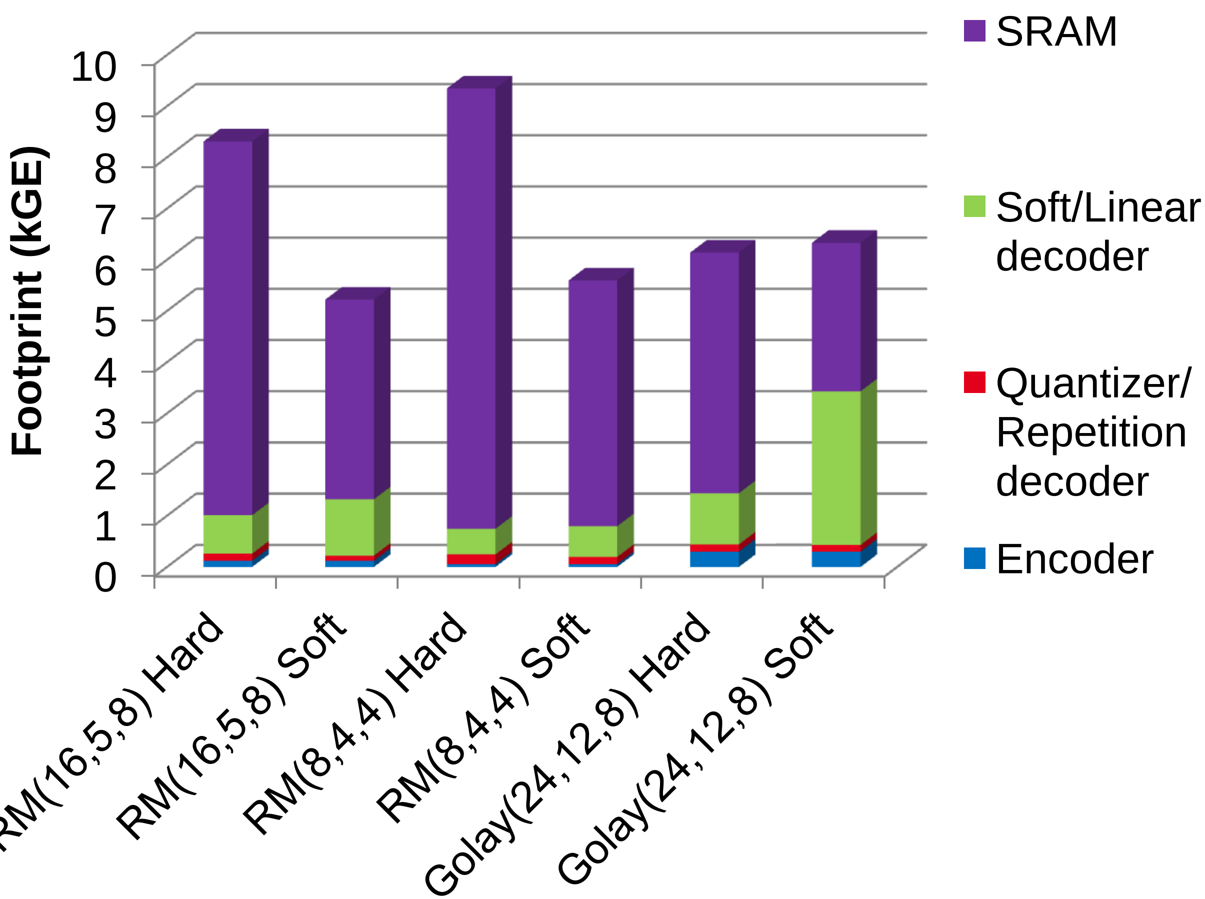
Golay[24,12,8] : 8 bits

Comparing amount of SRAM required

Code	Type	Repetition length	FRR	SRAM (bytes)
RM[16,5,8]	Hard	13	$1.6 \cdot 10^{-7}$	**910**
RM[16,5,8]	Soft	7	$3.7 \cdot 10^{-7}$	**490**
RM[8,4,4]	Hard	25	$3.4 \cdot 10^{-7}$	**1075**
RM[8,4,4]	Soft	14	$3.3 \cdot 10^{-7}$	**602**
Golay[24,12,8]	Hard	13	$4.0 \cdot 10^{-7}$	**585**
Golay[24,12,8]	Soft	8	$4.8 \cdot 10^{-7}$	**360**

Results show: soft decision decoding decreases amount of SRAM required 38 - 47% in these examples

Comparing total footprint



Impact of SRAM changes with:

- FRR
- Noise rate
- Key length
- Number of keys
- …

In this example:
SRAM cell ≈ 1GE

Conclusions

- New soft decoding method for memory-based PUFs**:**
  - Using only single enrollment measurement
  - Requires 38 - 47% less PUF bits than hard decoding
  - Solves issues from old method (NVM, footprint, enrollment time)
  - All example codes implemented efficiently in hardware

- New method comes at a limited cost in resources
- Size of PUF more dominant in footprint → cost decreases

- Decoder implementation to be chosen based on:
  - What to minimize: PUF size, footprint, …
  - Values of FRR, noise rate, key length, number of keys, …

Questions?