Physical Unclonable Functions

Coded Modulation, Shaping, and Helper Data Schemes

Robert F.H. Fischer

NACHRICHTENTECHNIK Universität Ulm

universität ulm

Acknowledgment

Many Thanks to my Co-Workers and Colleagues:

Sven Müelich
Institute of Communications Engineering

Holger Mandry
Institute of Microelectronics

Maurits Ortmanns
Institute of Microelectronics

This work was supported in parts by the
Deutsche Forschungsgemeinschaft (DFG, German Research Foundation)
under grands FI 982/15-1 and OR 245/16-1

Introduction

Introduction

Physical Unclonable Functions (PUFs): [PRTG'02], [GCDD'02], [SN'00], [BH'13], [Mae'13]

- physical hardware object
- unique, unpredictable, and uncontrollable
  due to random physical processes at the time of production
- cannot be duplicated or cloned, i.e., are physically unclonable

Modes of Operation:

- "strong" PUFs: the response is dependent on a challenge
- "weak" PUFs: a unique fingerprint is delivered (considered here)
  maybe better: *physical unclonable "object" / physical unclonable "fingerprint"*

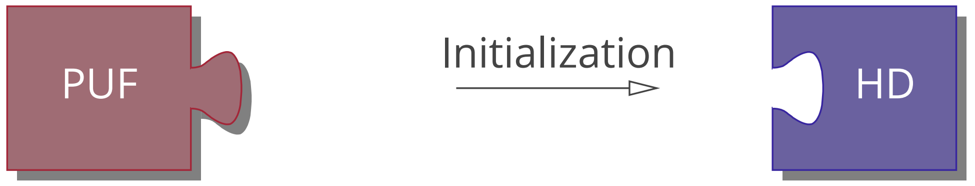
Observation and Approach:

- repeated PUF readout vary (slightly)
  due to variations in temperature, supply voltage, aging, . . .
- readout has to be stabilized — channel coding has to be applied

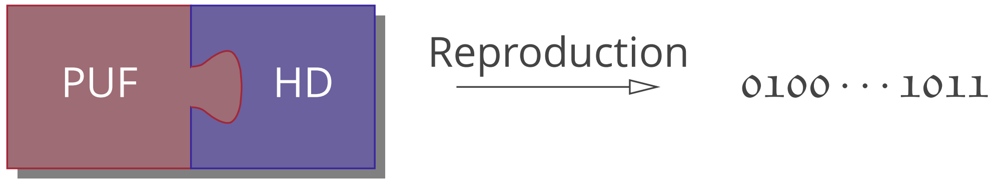
Introduction (II)

Procedure: fuzzy extractors / secure sketch [DRS'07]

- ***Initialization / Enrollment***
  based on the PUF readout ***helper data (HD) is generated***
  the helper data must not reveal any information about the PUF readout and may be public



- ***Reproduction***
  based on the noisy PUF readout and the helper data a stable (binary) word / key is generated



Applications:

- derivation of cryptographic keys / inherent key storage
  the PUF is private and the helper data may be public
- identification / countermeasure against product piracy
  the PUF is public and the helper data is private

Introduction (III)

Research Areas and Directions:

Microelectronics	***Computer Science***
more stable PUF architectures, efficient implementation of coding schemes, . . .	protocols, security, attacks, . . .
e.g., [MHV'12], [HBO'16], [MHK+'19], [KFPW'22]	e.g., [GCDD'02], [DRS'07], [MSSS'12], [Teb'22]
Information Theory	***Coding Theory***
fundamental procedures and limits, . . .	design suited channel coding schemes
e.g., [AC'93], [Mau'93], [CN'00], [IW'09], [GFBP'23]	e.g., [Mae'13], [PMBHS'15], [Müe'19], [FM'22]

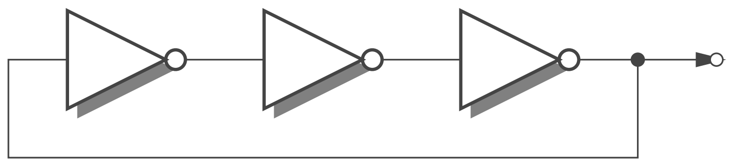
Overview

- *Classical Binary PUFs and Problem Statement*
- *Soft-Output PUFs*
- *Coded Modulation and Shaping*
- *Helper Data for Improved Decoding*
- *FPGA Implementation*

Classical Binary PUFs

Ring Oscillator PUFs

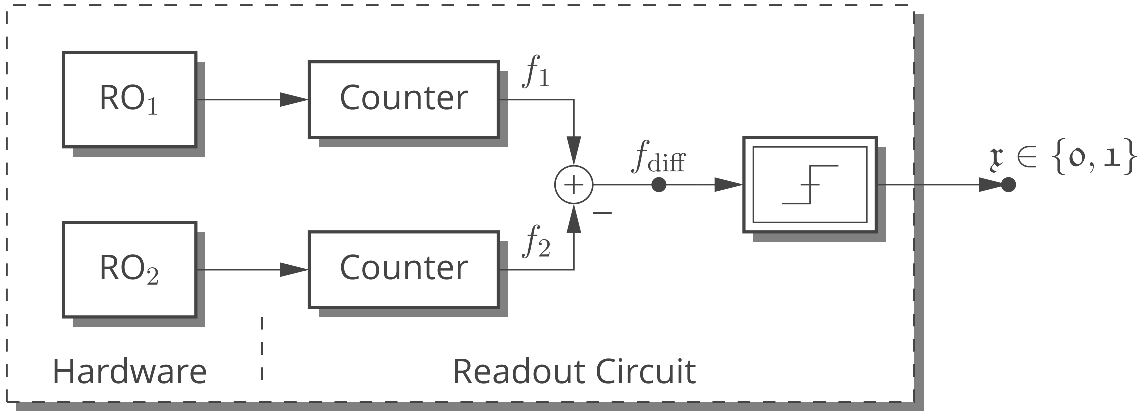
Ring Oscillator: ("silicon PUF")



- loop of an odd number of inverters (NOT gates)
- the circuit oscillates with a certain frequency
  actual value depends on uncontrollable variations within the manufacturing process

Classical Ring Oscillator PUF (ROPUF): [GCDD'02]

- sign of frequency difference f_{diff} is extracted



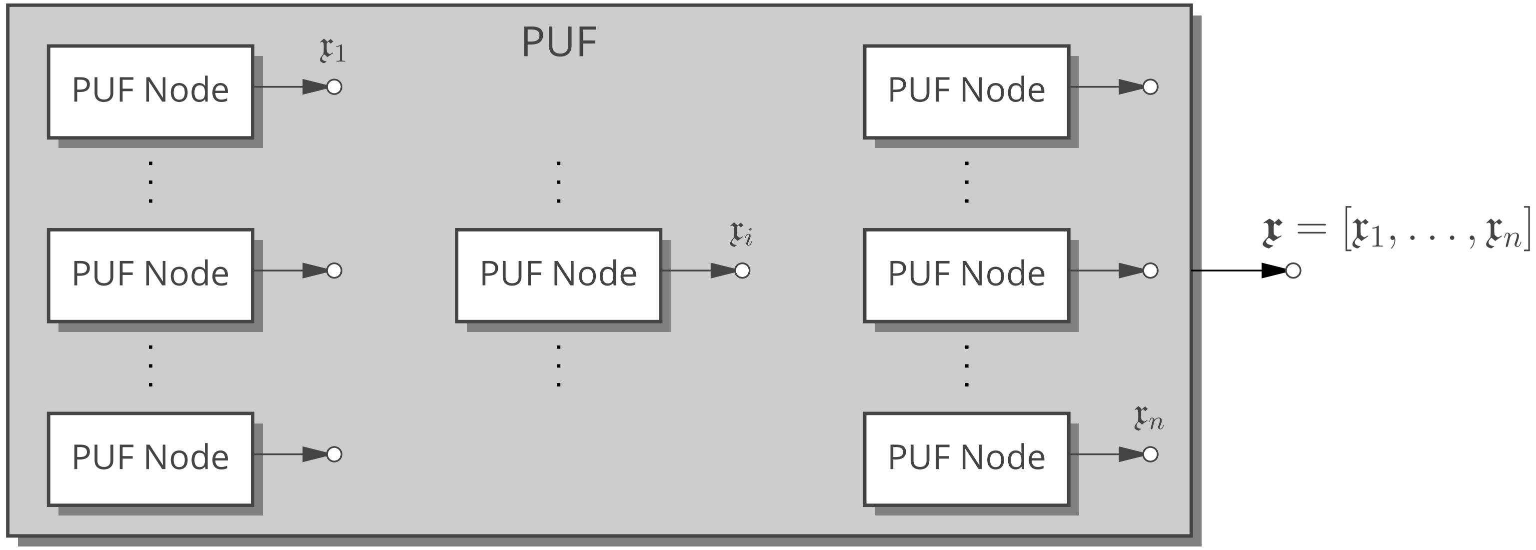
- basic block for generating a single random variable — ***PUF node***, ***PUF cell***, or ***PUF unit***

Notation: quantities over $\mathbb{R}$ are typeset as $\boldsymbol{x}$, $\boldsymbol{e}$, … — quantities over $\mathbb{F}_2$ are typeset in Fraktur font; $\mathfrak{x}$, $\boldsymbol{\mathfrak{x}}$, …

Classical PUFs

Extracted Information / Entire PUF:

- n ***independent*** PUF nodes constitute the PUF



- PUF readout vector $\mathfrak{x} = [\mathfrak{x}_1, \ldots, \mathfrak{x}_n] \in \mathbb{F}_2^n$
- $\mathfrak{x}_i$ ***uniformly*** and ***independently*** distributed
- each PUF instance has a unique readout $\mathfrak{x}$

Classical PUFs (II)

Extracted Information:

- each PUF instance has a unique *reference readout* $\mathfrak{x}_{\text{puf}}$ ⇒ *randomness in the manufacturing process*

$$\mathfrak{x}_{\text{puf},1} \; \mathfrak{x}_{\text{puf},2} \; \mathfrak{x}_{\text{puf},3} \; \cdots\cdot \; \mathfrak{x}_{\text{puf},i} \; \cdots\cdot \; \mathfrak{x}_{\text{puf},n}$$

Problem:

- repeatedly requested readouts will vary (slightly) ⇒ *randomness in the readout process*
  due to variations in temperature, supply voltage, aging, ...

- instability is traditionally modeled by a binary symmetric channel (BSC)

$$\boldsymbol{\mathfrak{y}}_{\text{puf}} = \boldsymbol{\mathfrak{x}}_{\text{puf}} \oplus \boldsymbol{\mathfrak{e}}_{\text{puf}}$$

  error pattern $\boldsymbol{\mathfrak{e}}_{\text{puf}}$ — usual assumption: bit error probability $p_{\text{BSC}} \approx 0.14$ e.g., [MHV'12], [MPMHS'14], [PMBHS'15]

  ⇒ *employ channel coding*

- However: the reference PUF readout $\boldsymbol{\mathfrak{x}}_{\text{puf}}$ is not a valid code word

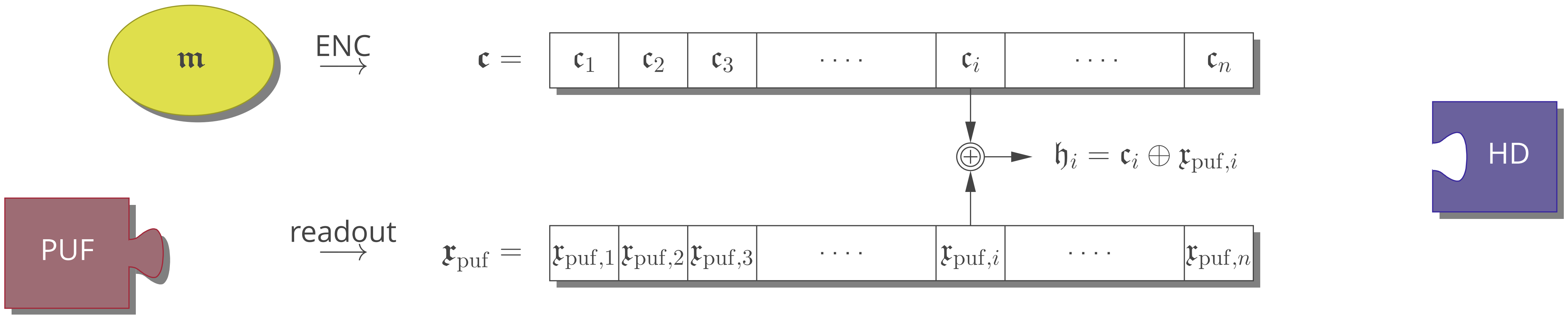
Classical PUFs (III)

Initialization / Enrollment Phase:

- the ***reference PUF readout*** $\mathfrak{x}_{\text{puf}}$ is measured
- choice: – binary channel code (rate k/n)
  – binary message word $\mathfrak{m}$ of length k — the corresponding code word $\mathfrak{c}$ is generated
- ***helper data*** is calculated as — ***code-offset algorithm*** e.g., [JW'99], [LT'03], [DRS'04]

$$\mathfrak{h} \overset{\text{def}}{=} \mathfrak{c} \oplus \mathfrak{x}_{\text{puf}}$$

- visualization



Reproduction Phase:

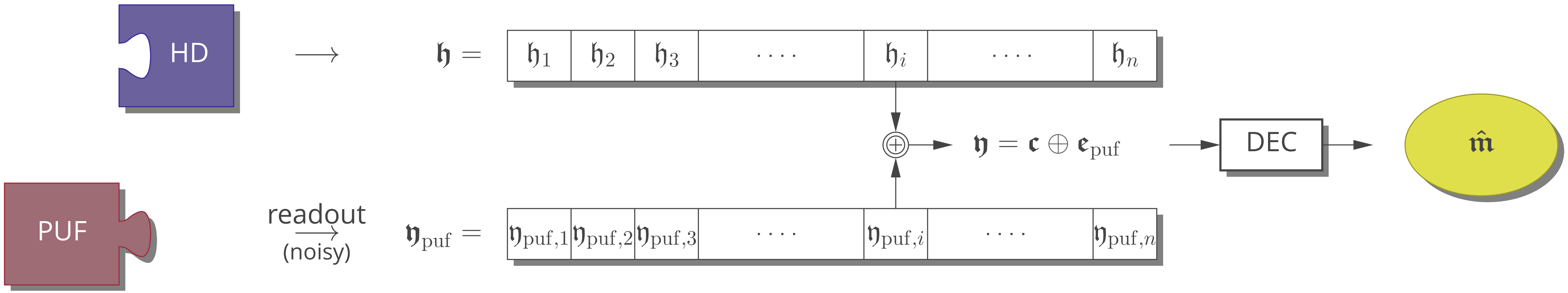
- noisy PUF readout

$$\boldsymbol{\mathfrak{y}}_{\text{puf}} = \boldsymbol{\mathfrak{x}}_{\text{puf}} \oplus \boldsymbol{\mathfrak{e}}_{\text{puf}} = \boldsymbol{\mathfrak{c}} \oplus \boldsymbol{\mathfrak{h}} \oplus \boldsymbol{\mathfrak{e}}_{\text{puf}}$$

- application of helper data

$$\boldsymbol{\mathfrak{y}} \stackrel{\text{def}}{=} \boldsymbol{\mathfrak{y}}_{\text{puf}} \oplus \boldsymbol{\mathfrak{h}} = \boldsymbol{\mathfrak{c}} \oplus \boldsymbol{\mathfrak{e}}_{\text{puf}}$$

- visualization

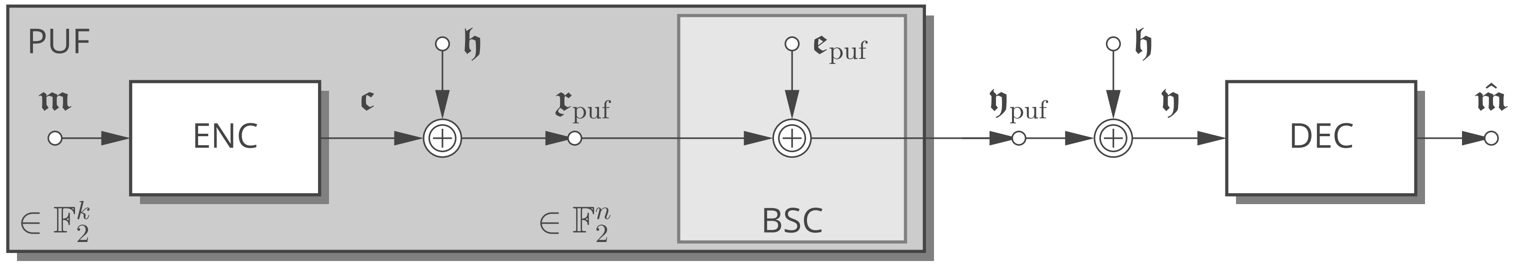


- standard (hard-decision) channel decoding reveals the message $\boldsymbol{\mathfrak{m}}$

Classical PUFs (V)

Model of the PUF:

- visualization



- randomness in the manufacturing process — $\mathfrak{x}_{\mathrm{puf}}$
- randomness in the readout process — $\mathfrak{e}_{\mathrm{puf}}$

Imagination of a Digital Communication Scheme:

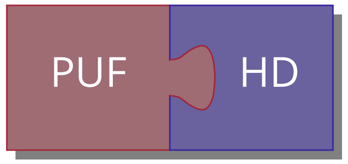
- randomly selected message $\mathfrak{m}$ of length k
- encoding and application of helper data gives $\mathfrak{x}_{\mathrm{puf}}$
- secret (key) to be retrieved: *message* $\mathfrak{m}$

Security of PUFs

Requirements: ($\mathrm{I}(\cdot;\ \cdot)$: mutual information)

- the PUF (reference) readout $\mathfrak{x}_{\text{puf}}$ and the helper data $\mathfrak{h}$ are known

  ⇒ *the message $\mathfrak{m}$ has to be decodable*



$$\mathrm{I}(\mathfrak{m};\ \{\mathfrak{x}_{\text{puf}}\ , \mathfrak{h}\}) = k$$

- only the PUF (reference) readout $\mathfrak{x}_{\text{puf}}$ is known

  ⇒ *no leakage must occur*

$$\mathrm{I}(\mathfrak{m};\ \mathfrak{x}_{\text{puf}}) = 0$$

- only the helper data $\mathfrak{h}$ is known

  ⇒ *no leakage must occur*

$$\mathrm{I}(\mathfrak{m};\ \mathfrak{h}) = 0$$

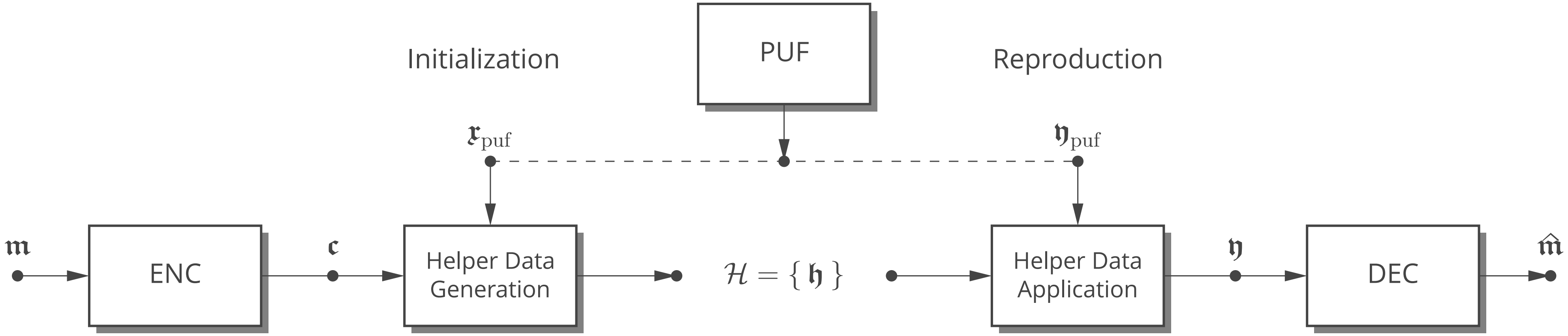
Interpretation:

- the readout $\mathfrak{x}_{\text{puf}}$ is a *one-time pad* for the codeword $\mathfrak{c}$ and vice versa

Interpretation

Channel Coding Problem:

- generation of and communication via ***helper data***

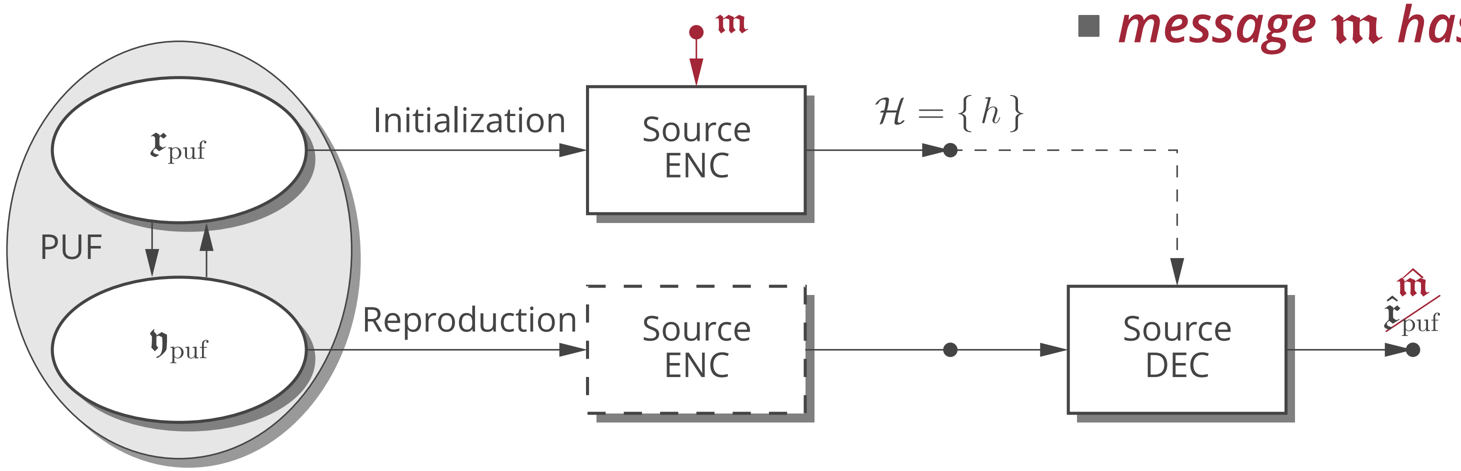


Source Coding Problem:

e.g., [GISK'19]

- Slepian–Wolf / Wyner–Ziv encoding

- ***message $\mathfrak{m}$ as additional randomness***
- ***message $\mathfrak{m}$ has to be recovered***



Numerical Examples

Word Error Ratio (WER) over the BSC Error Probability:

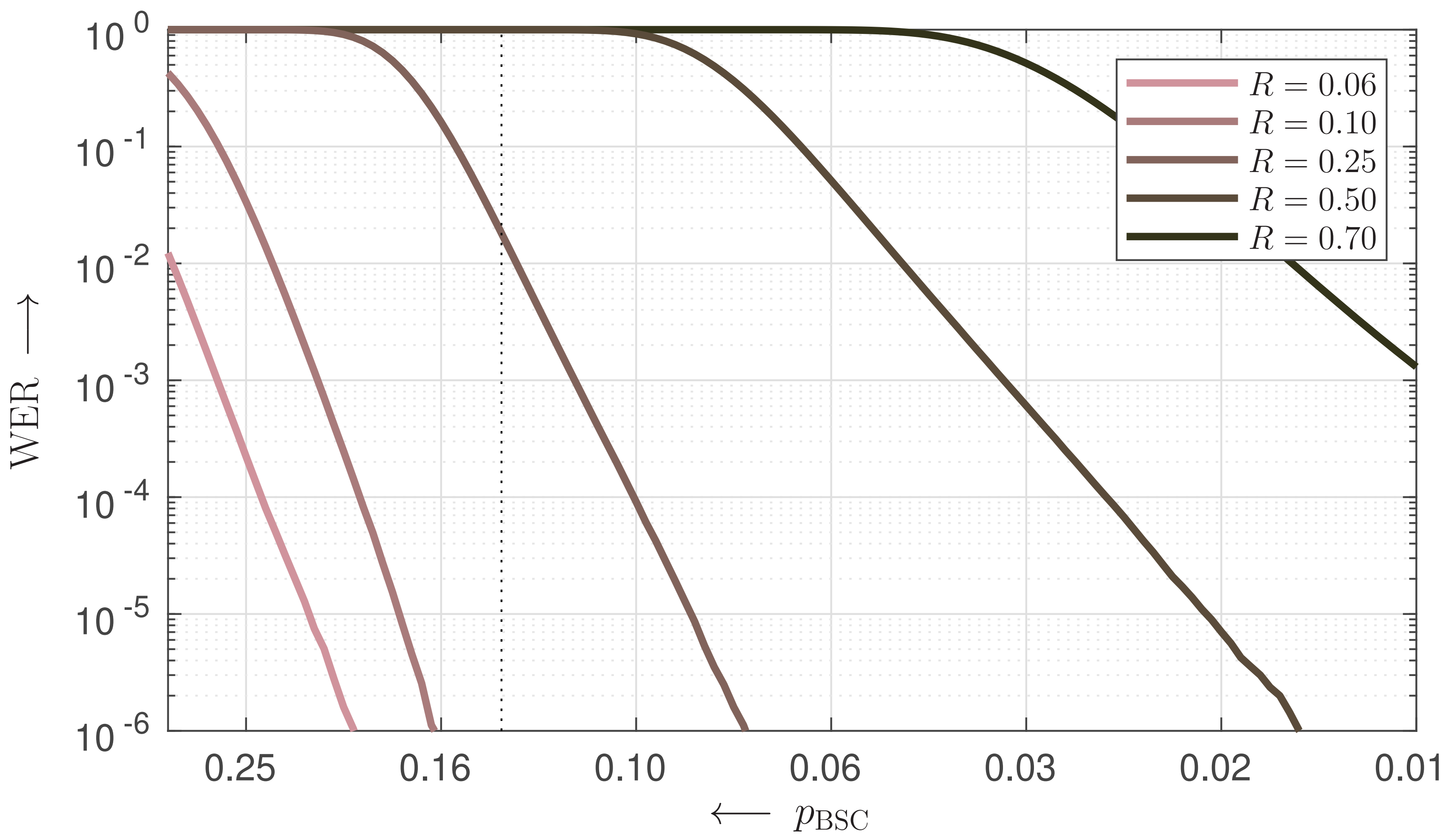
- PUF nodes: 1024

  mess. length: 61 to 717

  rate: $R = 0.06$ to 0.7 $\left[\frac{\text{bit}}{\text{node}}\right]$

- hard-decision decoding

- Polar code
  - codelength $n = 1024$
  - rate $R = 0.06$ to 0.7



Problem Statement

Channel Coding:

- ***Situation:*** vast majority of the literature is on ***binary*** codes and ***hard-decision*** decoding
- ***However:*** PUFs extract randomness from analog sources

Improvements: (the number n of PUF nodes is fixed)

- longer messages
  extract more than one bit of entropy per readout symbol $(k > n)$

  e.g., [TSB+'06], [BNCF'14], [GI'14], [WHGS'16] [ZPK+'16], [CBD+'17], [IOK+'18], [MHM+20]

  ⇒ ***multi-valued PUFs / coded modulation***

- higher reliability

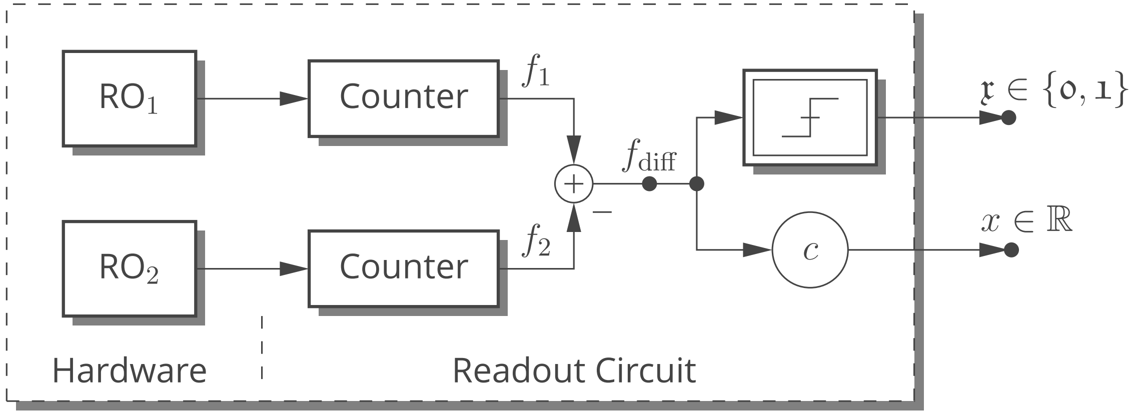
  e.g., [MTV'09], [MPSB'19], [MMOF'21], [KFPW'22]

  ⇒ ***utilize the soft output / advanced helper schemes***

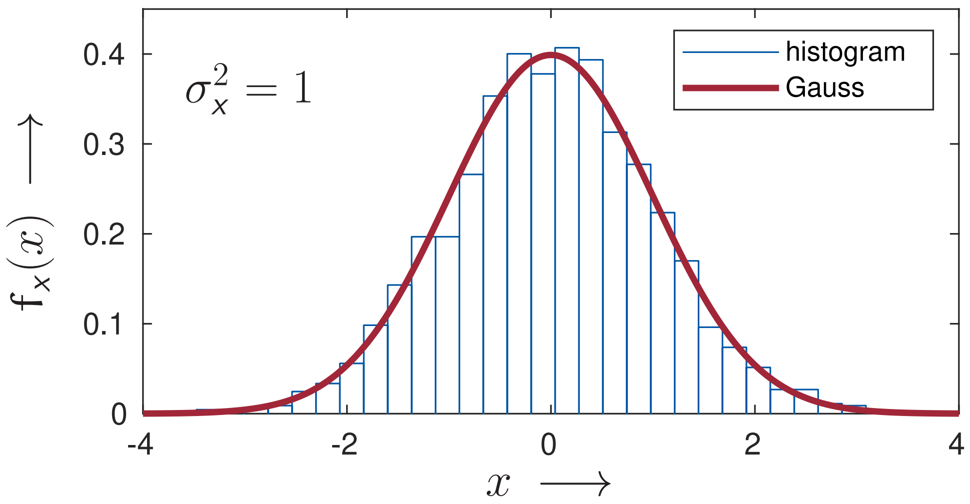
Soft-Output PUFs

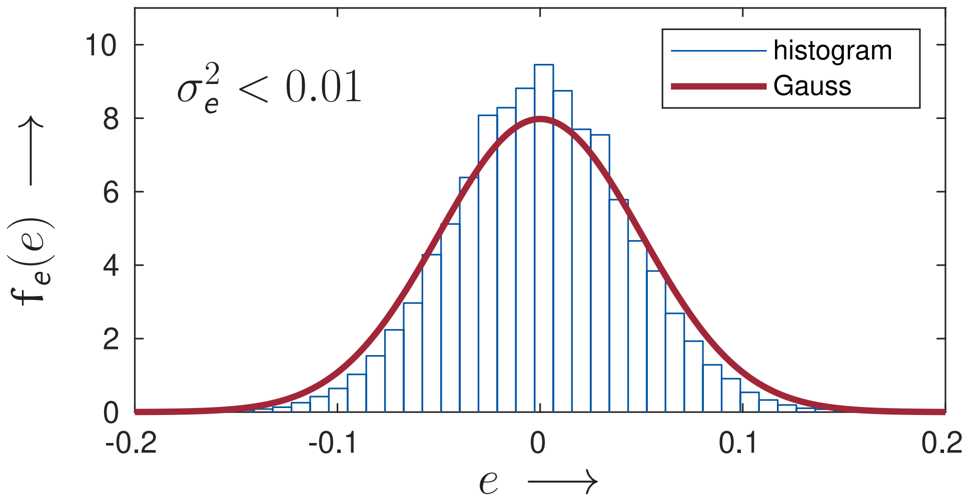
Soft-Decision Decoding:

- the real-valued frequency difference f_{diff} is utilized directly — reliability information



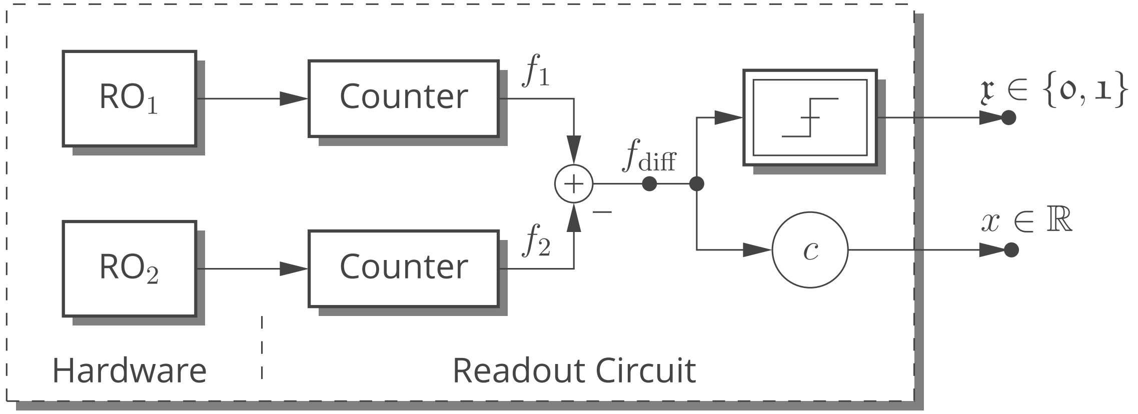
- measurement campaign at the Institute of Microelectronics using FPGA ROPUFs





Soft-Decision Decoding:

- the real-valued frequency difference f_{diff} is utilized directly — reliability information



- AWGN model

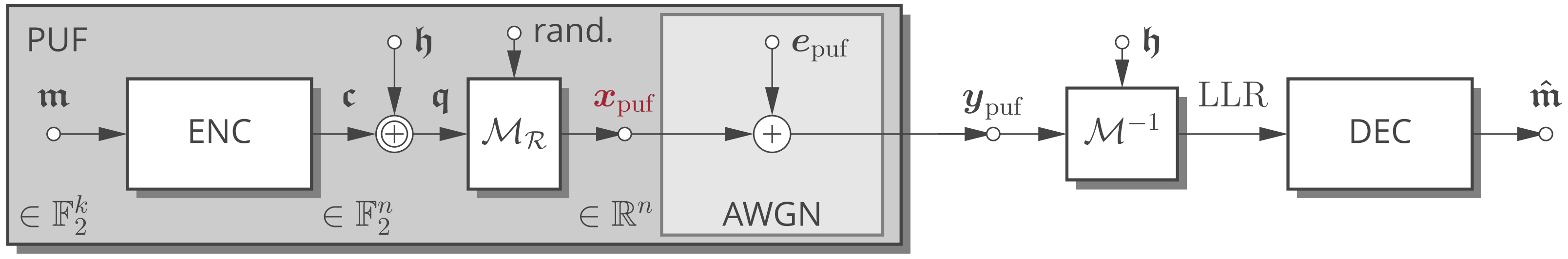
$$\boldsymbol{y}_{\mathrm{puf}} = \boldsymbol{x}_{\mathrm{puf}} + \boldsymbol{e}_{\mathrm{puf}}$$

  - reference/nominal readout $\boldsymbol{x}_{\mathrm{puf}}$ and error $\boldsymbol{e}_{\mathrm{puf}}$ are approx. zero-mean Gaussian distributed
  - scaling factor c such that $\sigma_x^2 = 1$ (per element)
  - error variance: $\sigma_e^2 < 0.01$

Soft-Output PUFs

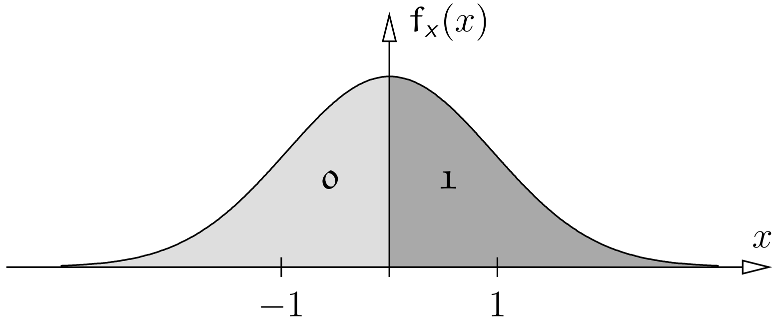
Model of the PUF:

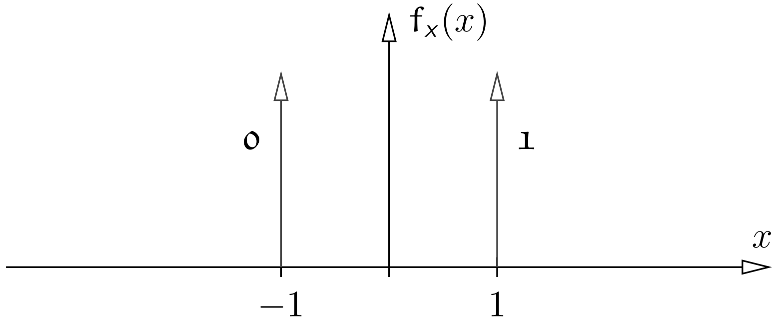
- we *imagine* a digital communication scheme — soft-decision



- random mapping — ***mapping bits to regions***
  - randomness at the transmitter
  - $\mathfrak{q}_i$ determines the region — the actual number $x_{\mathrm{puf},i}$ is drawn randomly according to a Gaussian pdf
  - individual but fixed for each PUF node (instance and position i in the codeword)

- Gaussian PUF readout vs. BPSK signaling

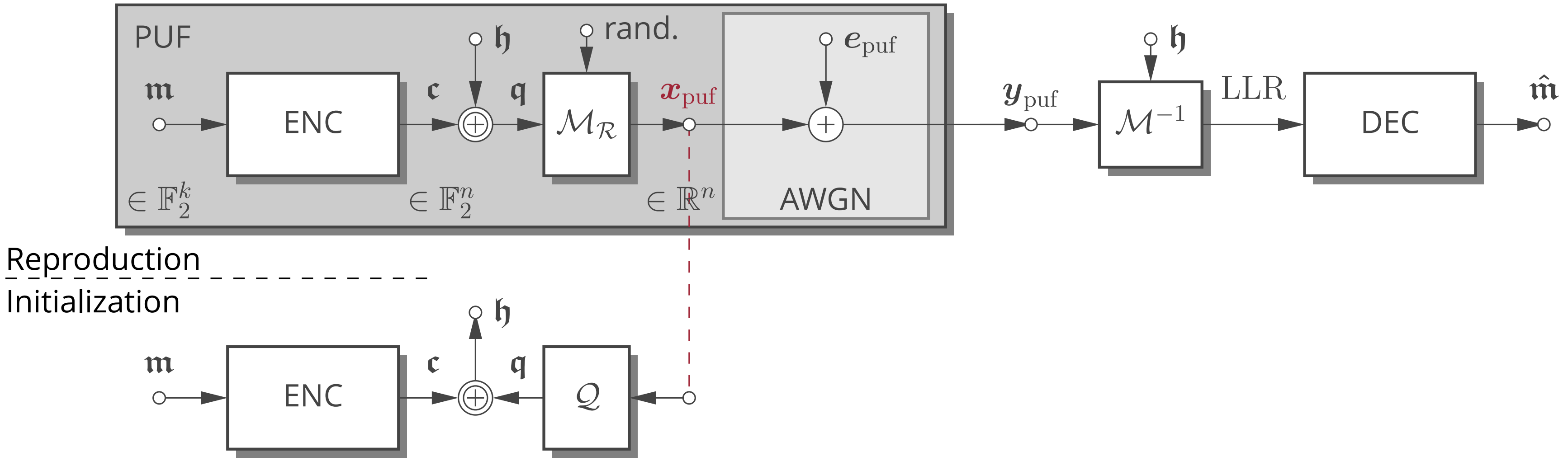




Soft-Output PUFs

Model of the PUF:

- we *imagine* a digital communication scheme — soft-decision



Initialization:

- determination of the actual region $\mathfrak{q}$
- encoding of the message to $\mathfrak{c}$
- calculation of helper data
  - $\mathfrak{c}$: desired region
  - $\mathfrak{q}$: actual region

$$\Rightarrow \quad \mathfrak{h} = \mathfrak{c} \oplus \mathfrak{q}$$

Soft-Output PUFs (II)

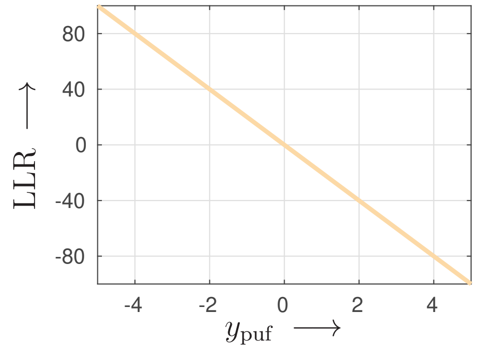
Soft-Decision Decoding:

- decoding metric: *log-likelihood ratio* (LLR)

$$\mathrm{LLR} = \log\left(\frac{\Pr\{\mathfrak{c}=0 \mid y_{\text{puf}}\}}{\Pr\{\mathfrak{c}=1 \mid y_{\text{puf}}\}}\right) = \log\left(\frac{f_y(y_{\text{puf}} \mid \mathfrak{c}=0)}{f_y(y_{\text{puf}} \mid \mathfrak{c}=1)}\right)$$

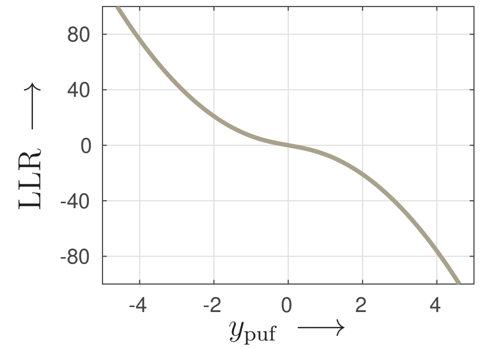
- BPSK over the AWGN channel

$$\mathrm{LLR} = \mp \frac{2}{\sigma_e^2}\, y_{\text{puf}}\,, \qquad \begin{array}{l} -,\ \mathfrak{h}=0 \\ +,\ \mathfrak{h}=1 \end{array}$$



- Gaussian PUF readout

$$\mathrm{LLR} = \pm \log\left(\frac{Q(+F\, y_{\text{puf}})}{Q(-F\, y_{\text{puf}})}\right)\,, \qquad \begin{array}{l} +,\ \mathfrak{h}=0 \\ -,\ \mathfrak{h}=1 \end{array}$$



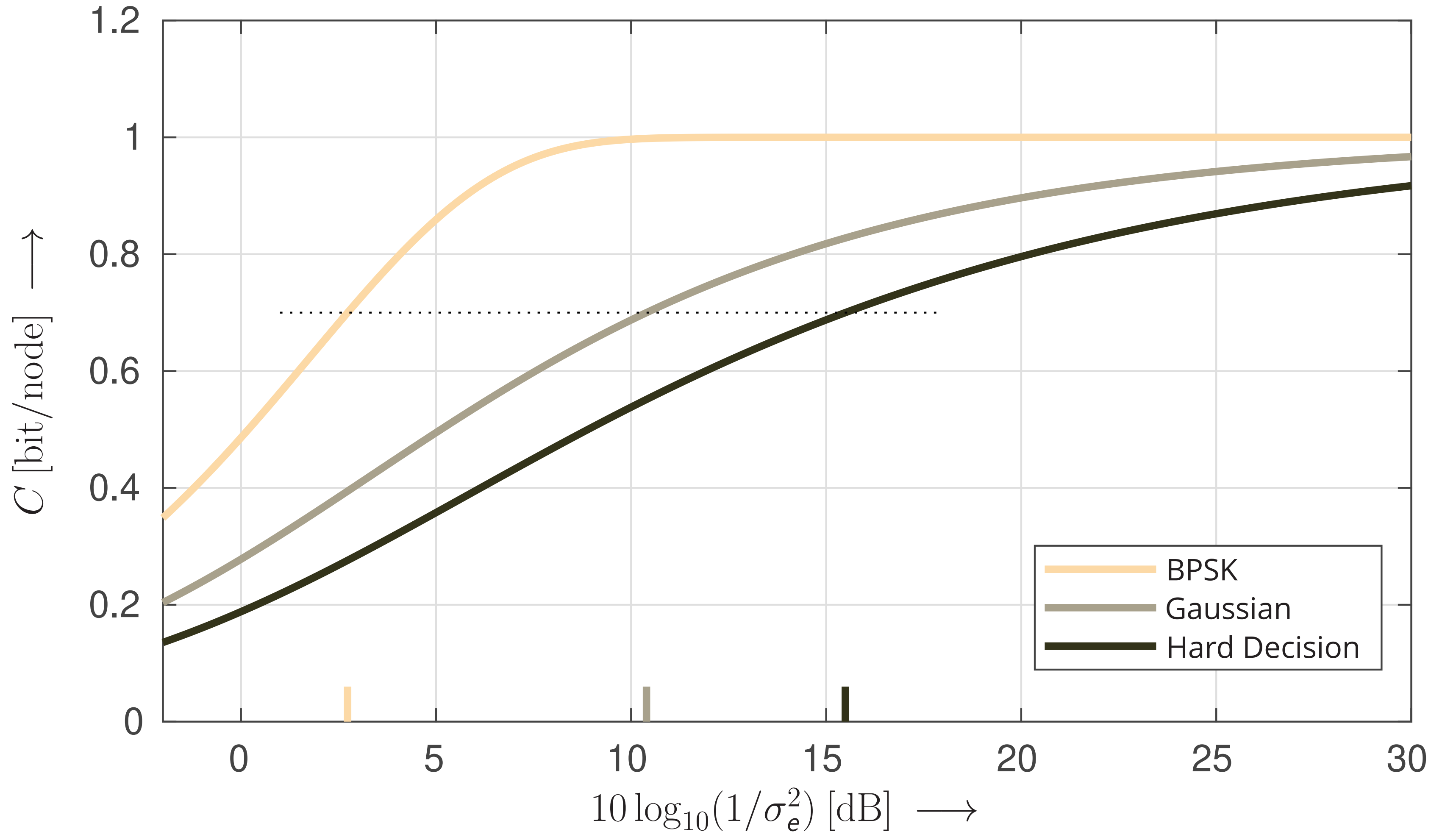
with $F \overset{\text{def}}{=} \frac{1}{\sqrt{1+\sigma_e^2}\,\sigma_e}$

$$Q(x) \overset{\text{def}}{=} \int_x^\infty \frac{1}{\sqrt{2\pi}}\, e^{-\frac{z^2}{2}}\, dz$$ (complementary Gaussian integral function)

Numerical Examples

Capacities over the Signal-to-Noise Ratio (in dB):

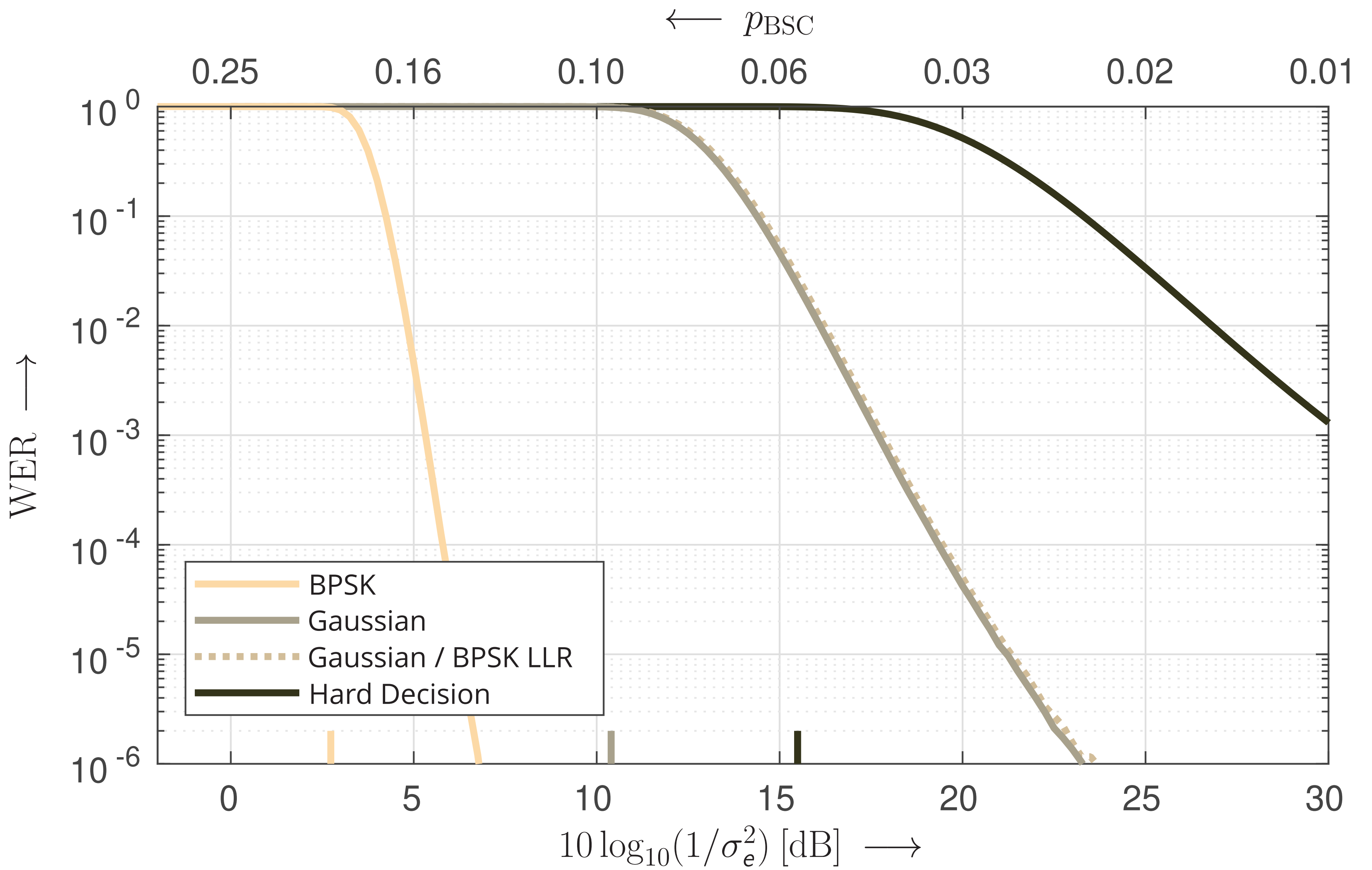
- BPSK
- Gaussian readout



Numerical Examples (II)

Word Error Ratio (WER) over the Signal-to-Noise Ratio (in dB):

- PUF nodes: 1024
  mess. length: 717
  rate: $R = 0.7$ $\left[\frac{\text{bit}}{\text{node}}\right]$
- Polar code
  - codelength $n = 1024$
  - rate $R = 0.7$

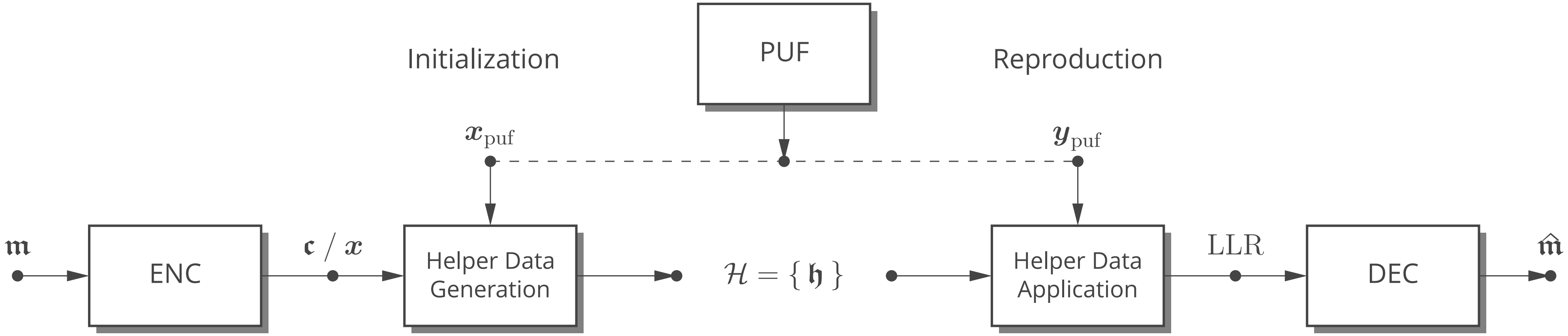


Coded Modulation and Shaping

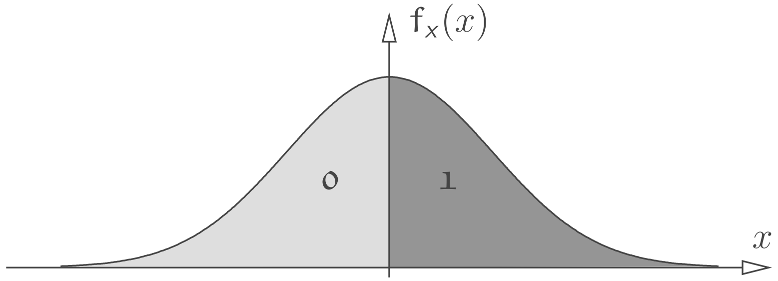
Situation

Binary Soft-Output PUF:

- generation of and communication via ***helper data***



- mapping bits to ***regions***



Challenge:

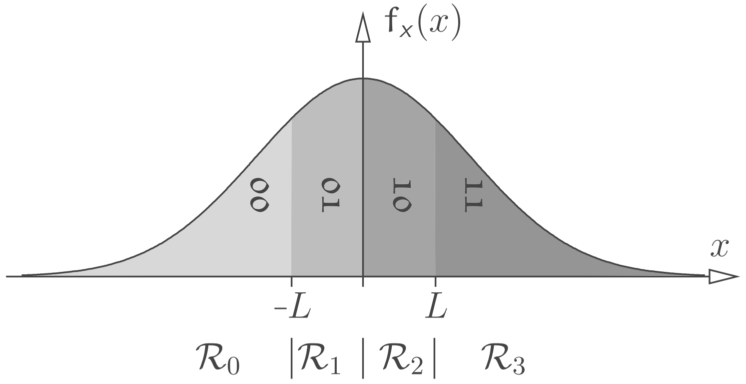
- increase code rate / size of the message $\mathfrak{m}$ — extract more than one bit per PUF node

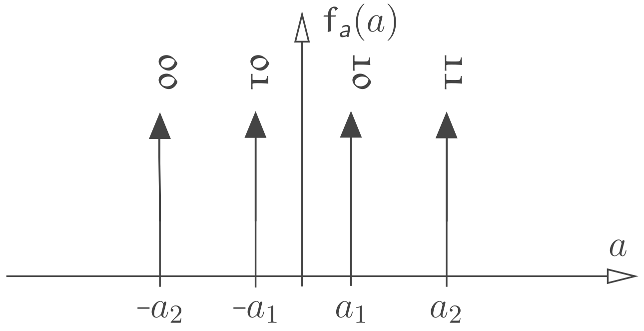
  ⇒ ***employ higher-order modulation / coded modulation***

Regions and Schemes

PUF Readout and Regions:

- regions for 4-ary signaling





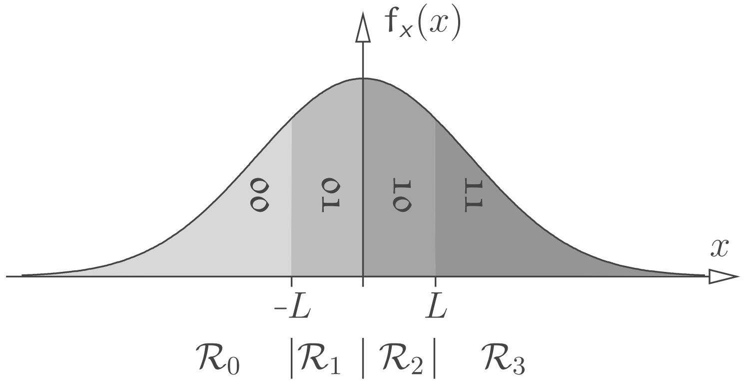
- regions $\mathcal{R}_\rho$
- natural labeling:
  - label $\mathfrak{c} = [\mathfrak{c}_1 \mathfrak{c}_0]$
  - region number $\rho = [\mathfrak{c}_1 \mathfrak{c}_0]_2$
- for $L = 0.675$ the regions are drawn with the same probability

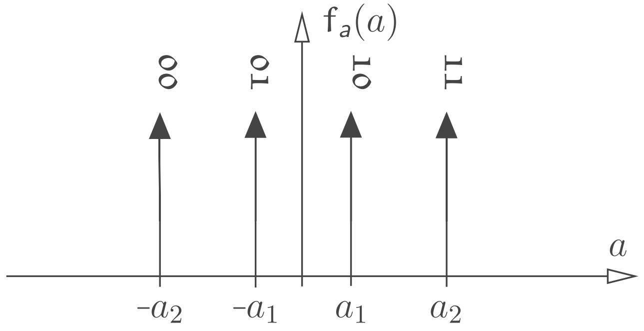
⇒ ***4-ary uniform scheme***

Regions and Schemes

PUF Readout and Regions:

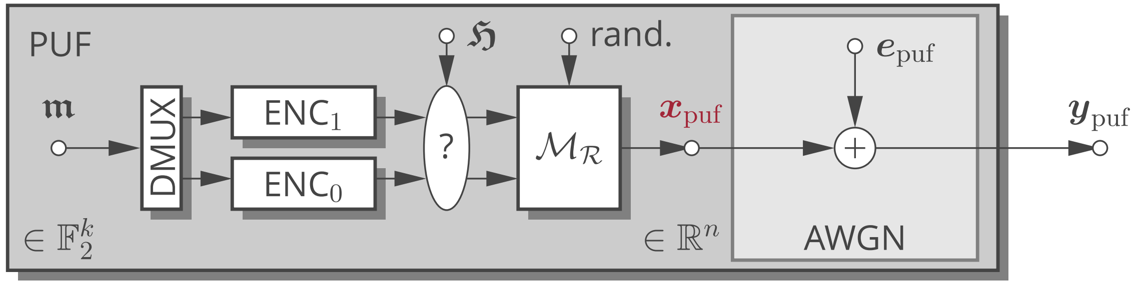
- regions for 4-ary signaling





Model of the PUF:

- we *imagine* a digital communication scheme

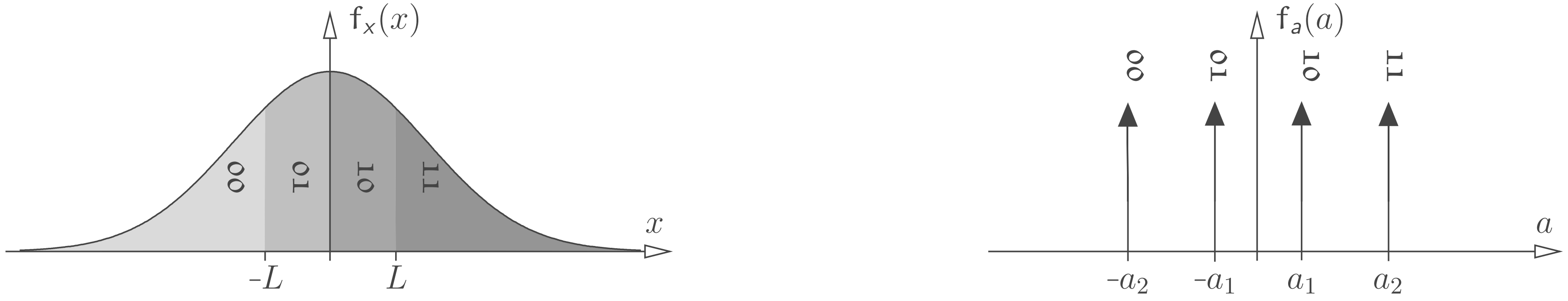


- ***mapping bits to regions*** — the actual number is drawn randomly according to a Gaussian pdf
- suited helper data scheme required

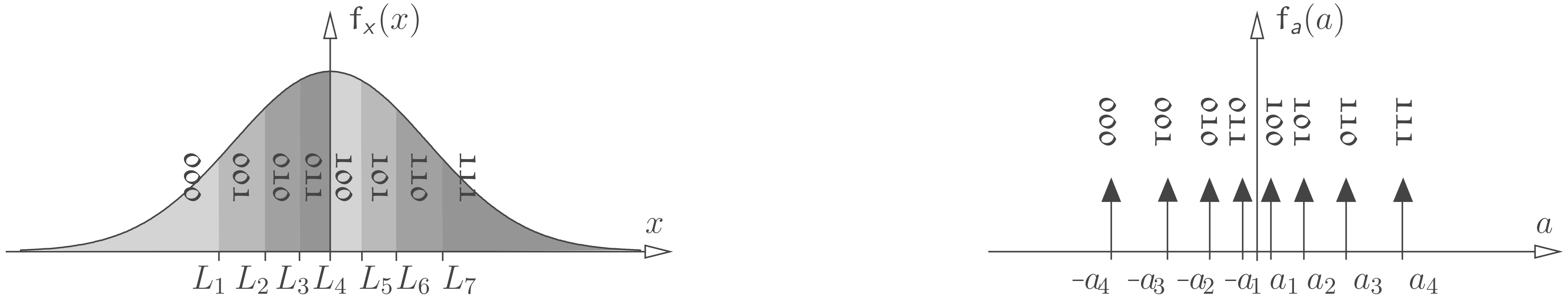
Regions and Schemes (II)

PUF Readout and Regions:

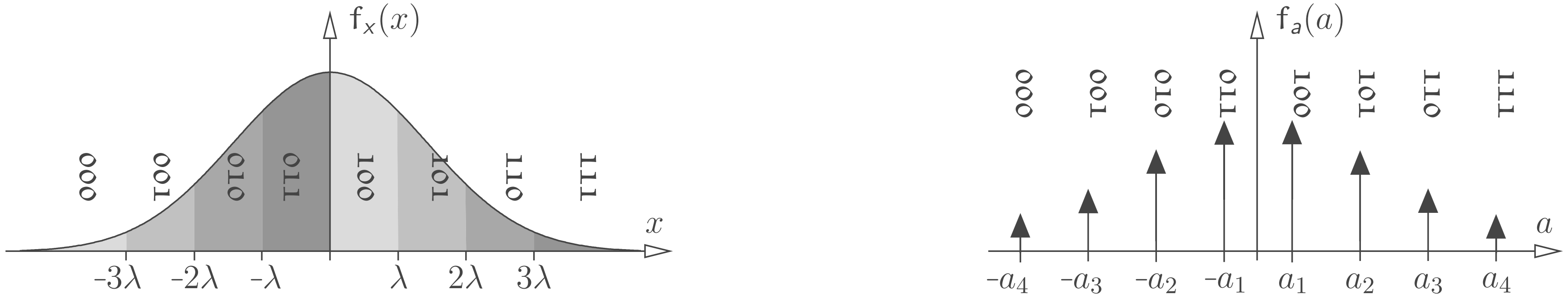
- 4-ary uniform signaling $(L = 0.675)$



- 8-ary uniform signaling



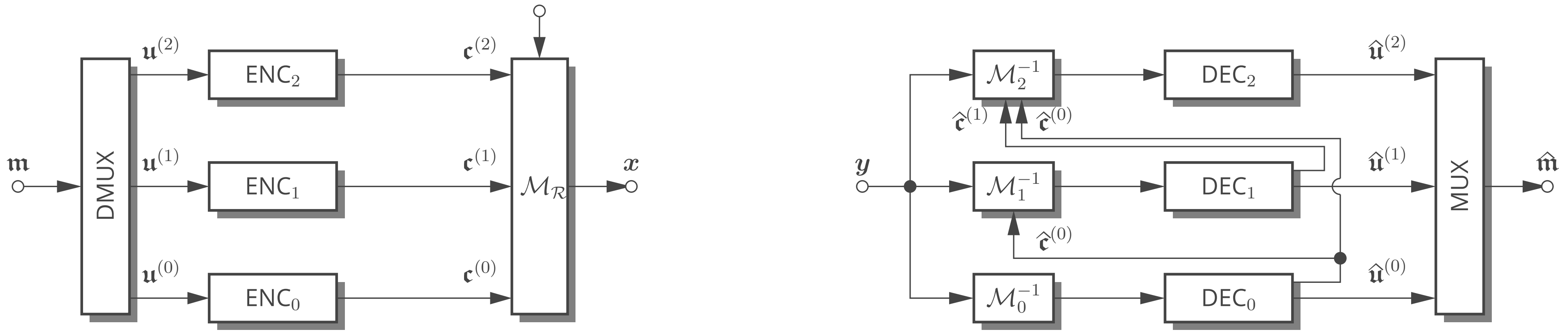
- 8-ary non-uniform signaling / shaping



Regions and Schemes (III)

Multilevel Encoder and Multistage Decoding: here: $M = 8$, $\mu = \log_2(M)$

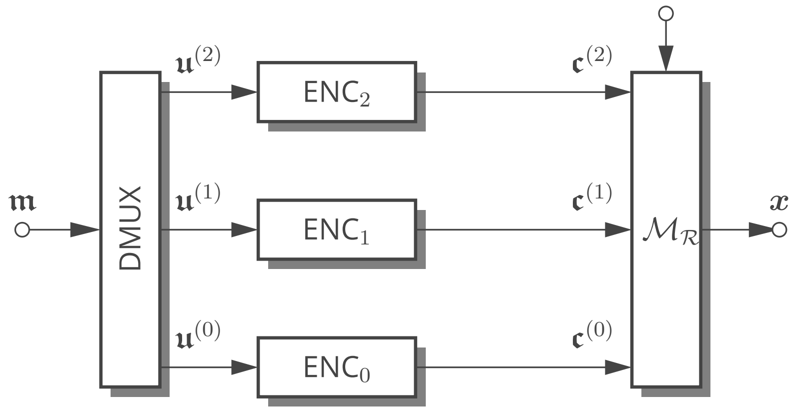
- scheme for uniform signaling

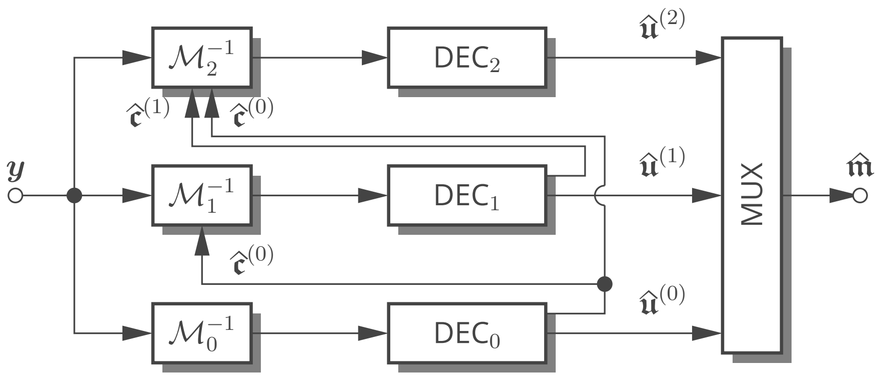


Regions and Schemes (III)

Multilevel Encoder and Multistage Decoding: here: $M = 8,\ \mu = \log_2(M)$

- scheme for uniform signaling





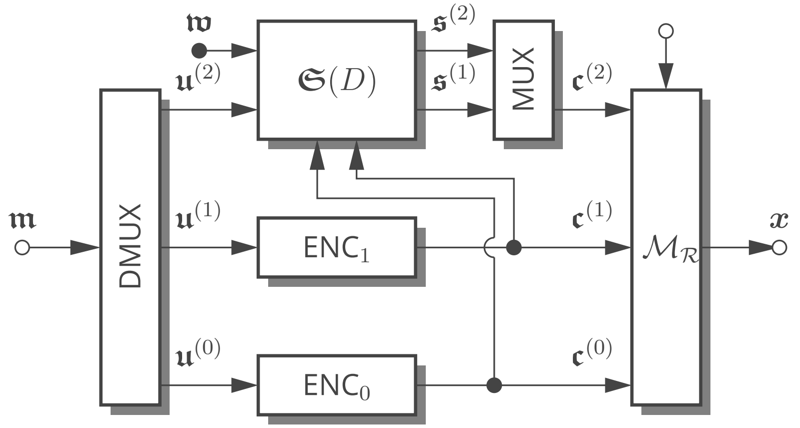
- specification by ***codematrix*** (code of length n)

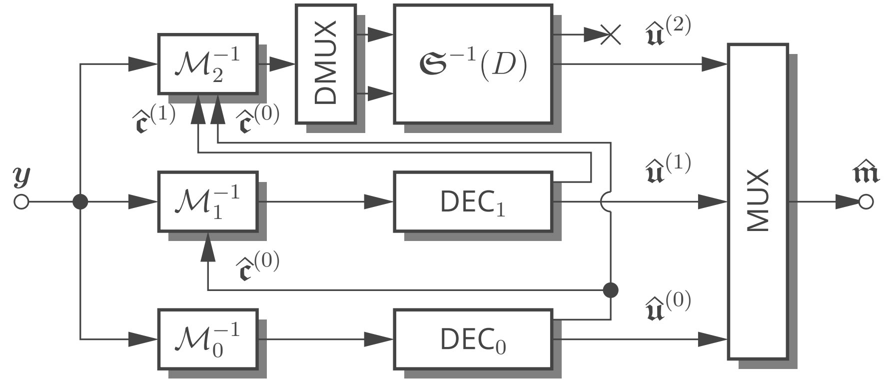
$$\mathfrak{m} \xrightarrow{\text{ENC}} \mathfrak{C} = \begin{array}{|c|c|c|c|c|c|c|}\hline \mathfrak{c}_{\mu\text{-}1,1} & \mathfrak{c}_{\mu\text{-}1,2} & \mathfrak{c}_{\mu\text{-}1,3} & \cdots\cdot & \mathfrak{c}_{\mu\text{-}1,i} & \cdots\cdot & \mathfrak{c}_{\mu\text{-}1,n} \\ \hline \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \hline \mathfrak{c}_{0,1} & \mathfrak{c}_{0,2} & \mathfrak{c}_{0,3} & \cdots\cdot & \mathfrak{c}_{0,i} & \cdots\cdot & \mathfrak{c}_{0,n} \\ \hline \end{array} = \begin{bmatrix} \mathfrak{c}^{(\mu-1)} \\ \vdots \\ \mathfrak{c}^{(0)} \end{bmatrix}$$

Region Number $\rho_1 \quad \rho_2 \quad \rho_3 \quad \cdots \quad \rho_i = [\mathfrak{c}_{\mu\text{-}1,i} \cdots \mathfrak{c}_{0,i}]_2 \quad \cdots \quad \rho_n$

Multilevel Encoder and Multistage Decoding: here: $M = 8$, $\mu = \log_2(M)$

- scheme with trellis shaping (highest level has rate $1/2$)





- specification by ***codematrix*** (code of length n)

$\mathfrak{m} \xrightarrow{\text{ENC}} \mathfrak{C} =$

$\mathfrak{c}_{\mu\text{-}1,1}$	$\mathfrak{c}_{\mu\text{-}1,2}$	$\mathfrak{c}_{\mu\text{-}1,3}$	$\cdots$	$\mathfrak{c}_{\mu\text{-}1,i}$	$\cdots$	$\mathfrak{c}_{\mu\text{-}1,n}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathfrak{c}_{0,1}$	$\mathfrak{c}_{0,2}$	$\mathfrak{c}_{0,3}$	$\cdots$	$\mathfrak{c}_{0,i}$	$\cdots$	$\mathfrak{c}_{0,n}$

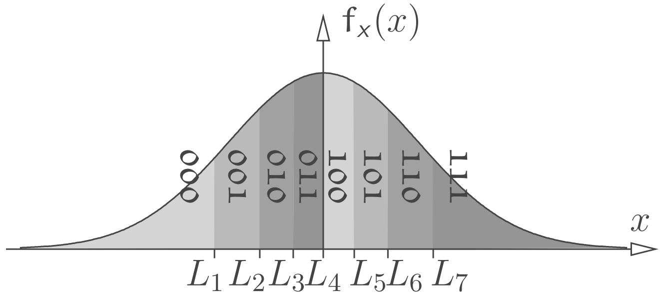
$= \begin{bmatrix} \mathfrak{c}^{(\mu-1)} \\ \vdots \\ \mathfrak{c}^{(0)} \end{bmatrix}$

Region Number $\rho_1 \quad \rho_2 \quad \rho_3 \quad \cdots \quad \rho_i = [\mathfrak{c}_{\mu\text{-}1,i} \cdots \mathfrak{c}_{0,i}]_2 \quad \cdots \quad \rho_n$

Numerical Examples

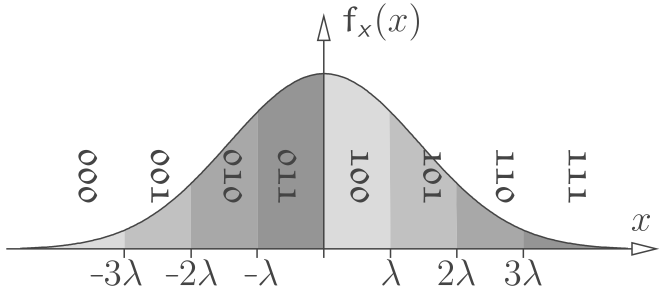
Capacities over the Signal-to-Noise Ratio (in dB):

- uniform:

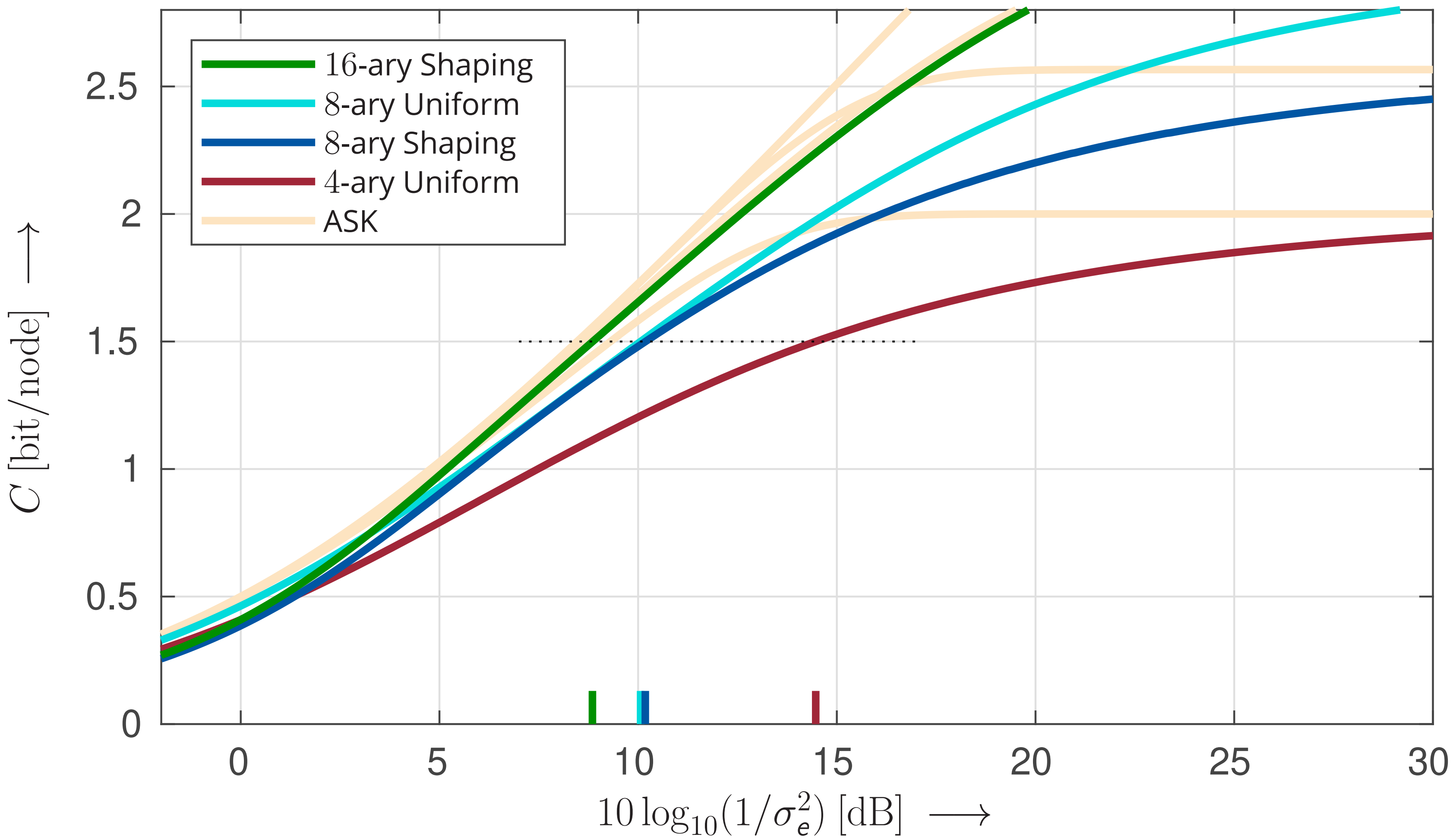


  - 4-ary Uniform
  - 8-ary Uniform

- shaping:

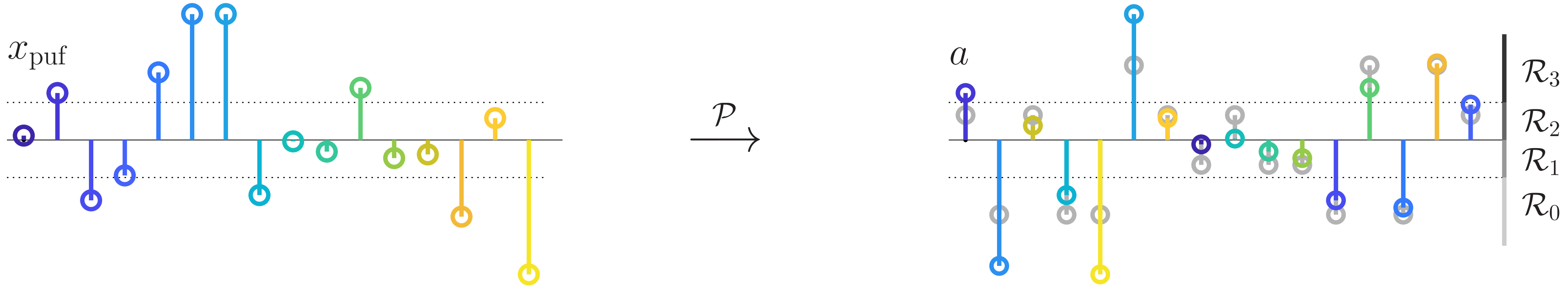


  - 8-ary Shaping: $\lambda = 0.70$
  - 16-ary Shaping: $\lambda = 0.35$
  - highest level: $R_{\mu\text{-}1} = 0.5$, hard decision



First Approach: generate a valid codeword in signal space

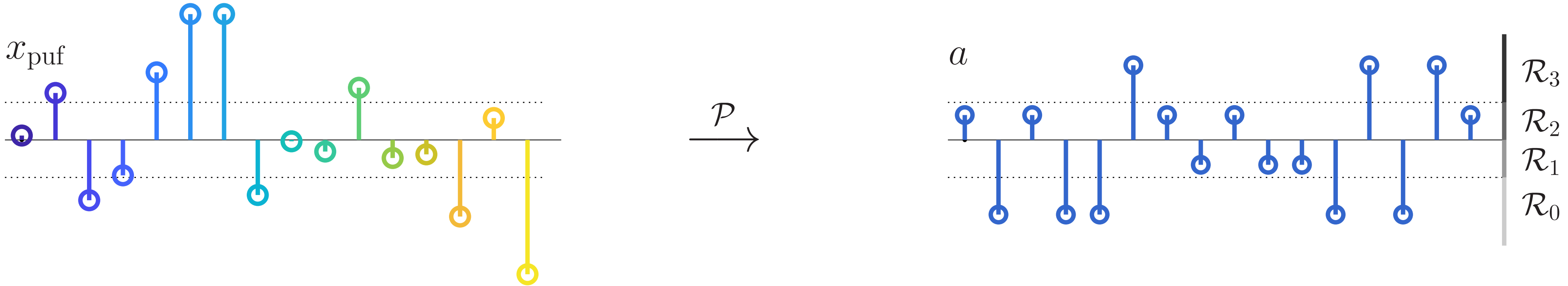
- employ ***permutation*** and ***sign flip*** [FM'22]
  - easy to implement
  - large number of bits required to store the helper data: $\approx n(1 + \log_2(n))$
  - no perfect match possible



Helper Data Scheme

First Approach: generate a valid codeword in signal space

- employ ***permutation*** and ***sign flip*** [FM'22]
  - easy to implement
  - large number of bits required to store the helper data: $\approx n(1+\log_2(n))$
  - no perfect match possible



$$\mathfrak{Q} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \xleftarrow{\mathcal{H}} \mathfrak{C} = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}$$

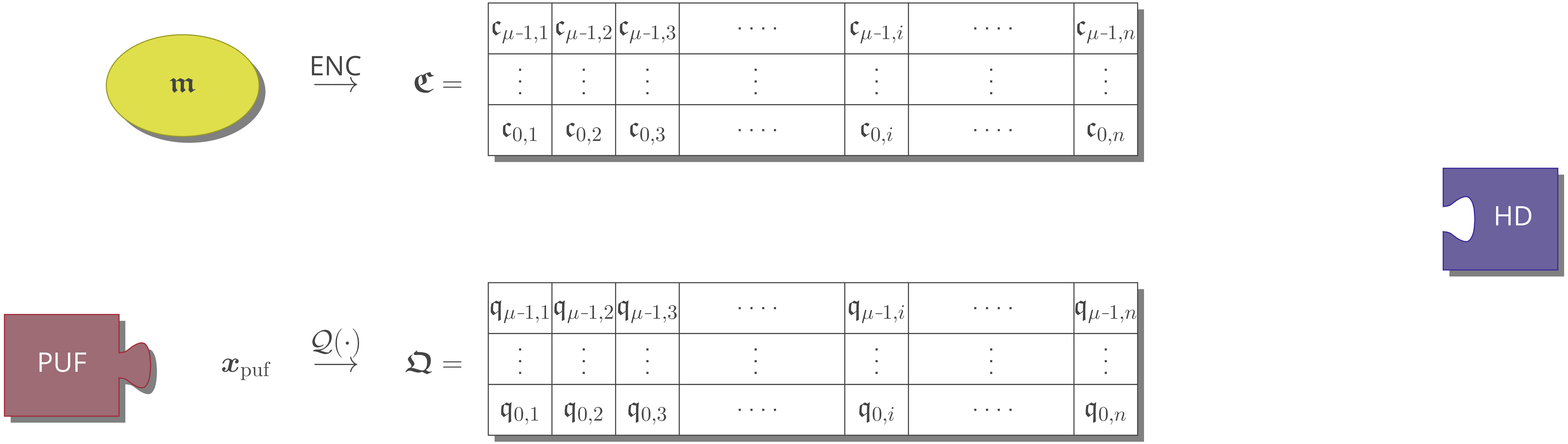
Better Approach: adapt LLR calculation

- employ a ***conversion*** of the region labels
  - applied element-wise
  - small number of bits required to store the helper data: $n\ \log_2(M)$
  - ideal LLR calculation

Helper Data Scheme (II)

Calculation of Helper Data: uniform signaling

- visualization

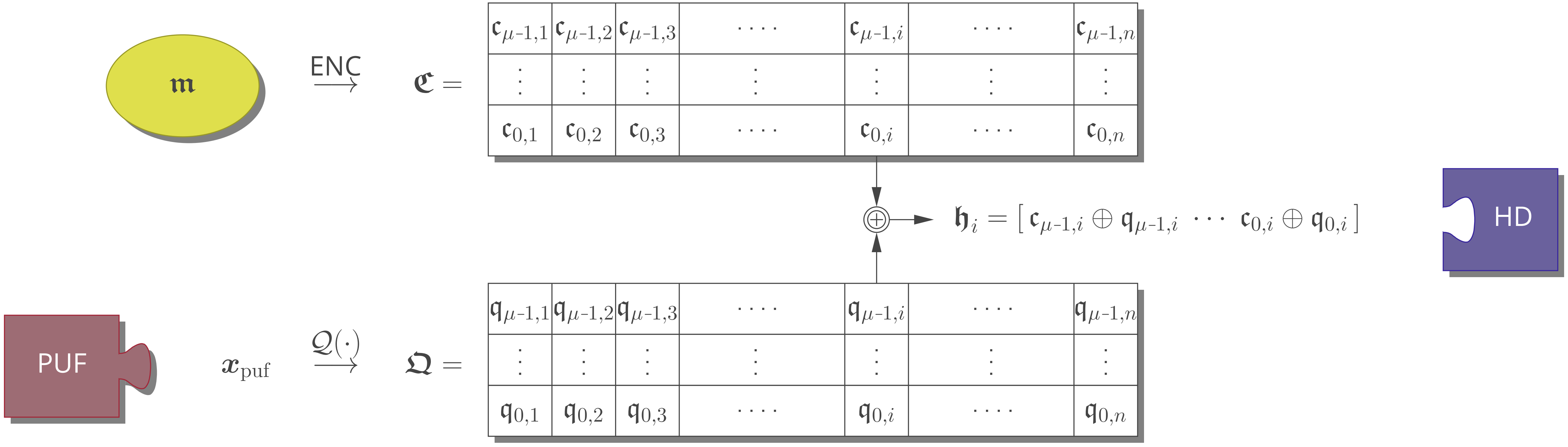


- $[\mathfrak{c}_{\mu\text{-}1,i} \ \cdots \ \mathfrak{c}_{0,i}]_2$: desired codesymbols
  $[\mathfrak{q}_{\mu\text{-}1,i} \ \cdots \ \mathfrak{q}_{0,i}]_2$: obtained by quantization $\mathcal{Q}(\cdot)$

Helper Data Scheme (II)

Calculation of Helper Data: uniform signaling

- visualization



$\mathfrak{c}_{\mu\text{-}1,1}$	$\mathfrak{c}_{\mu\text{-}1,2}$	$\mathfrak{c}_{\mu\text{-}1,3}$	$\cdots$	$\mathfrak{c}_{\mu\text{-}1,i}$	$\cdots$	$\mathfrak{c}_{\mu\text{-}1,n}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathfrak{c}_{0,1}$	$\mathfrak{c}_{0,2}$	$\mathfrak{c}_{0,3}$	$\cdots$	$\mathfrak{c}_{0,i}$	$\cdots$	$\mathfrak{c}_{0,n}$

$\mathfrak{q}_{\mu\text{-}1,1}$	$\mathfrak{q}_{\mu\text{-}1,2}$	$\mathfrak{q}_{\mu\text{-}1,3}$	$\cdots$	$\mathfrak{q}_{\mu\text{-}1,i}$	$\cdots$	$\mathfrak{q}_{\mu\text{-}1,n}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\mathfrak{q}_{0,1}$	$\mathfrak{q}_{0,2}$	$\mathfrak{q}_{0,3}$	$\cdots$	$\mathfrak{q}_{0,i}$	$\cdots$	$\mathfrak{q}_{0,n}$

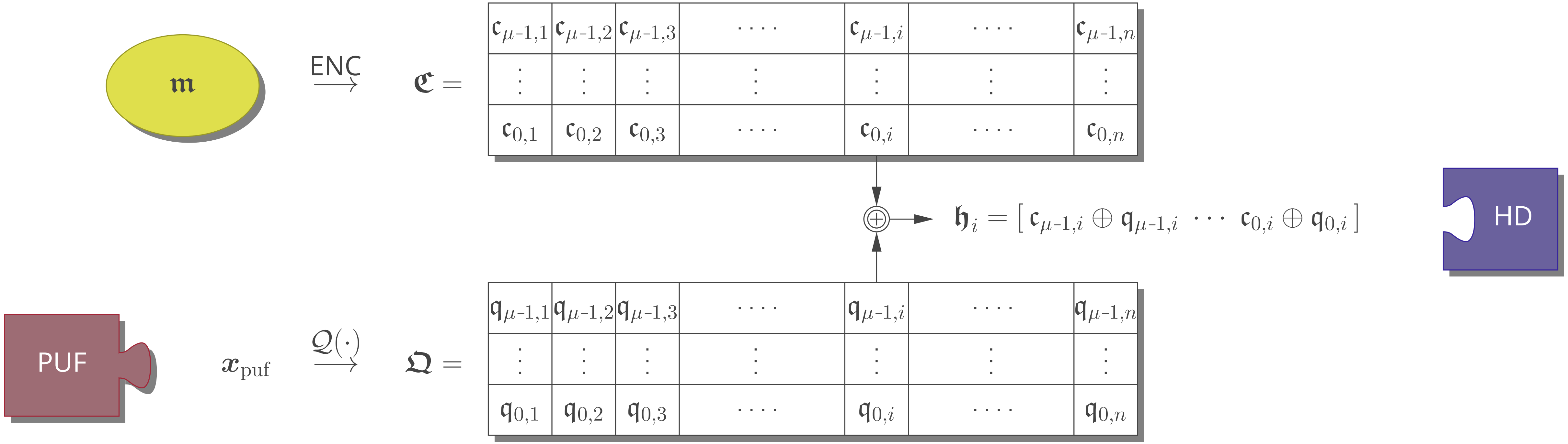
- $[\mathfrak{c}_{\mu\text{-}1,i} \;\cdots\; \mathfrak{c}_{0,i}]_2$: desired codesymbols
  $[\mathfrak{q}_{\mu\text{-}1,i} \;\cdots\; \mathfrak{q}_{0,i}]_2$: obtained by quantization $\mathcal{Q}(\cdot)$

- helper data: $\mathfrak{H} = \mathfrak{C} \oplus \mathfrak{Q}$

Helper Data Scheme (II)

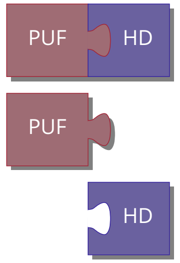
Calculation of Helper Data: uniform signaling

- visualization



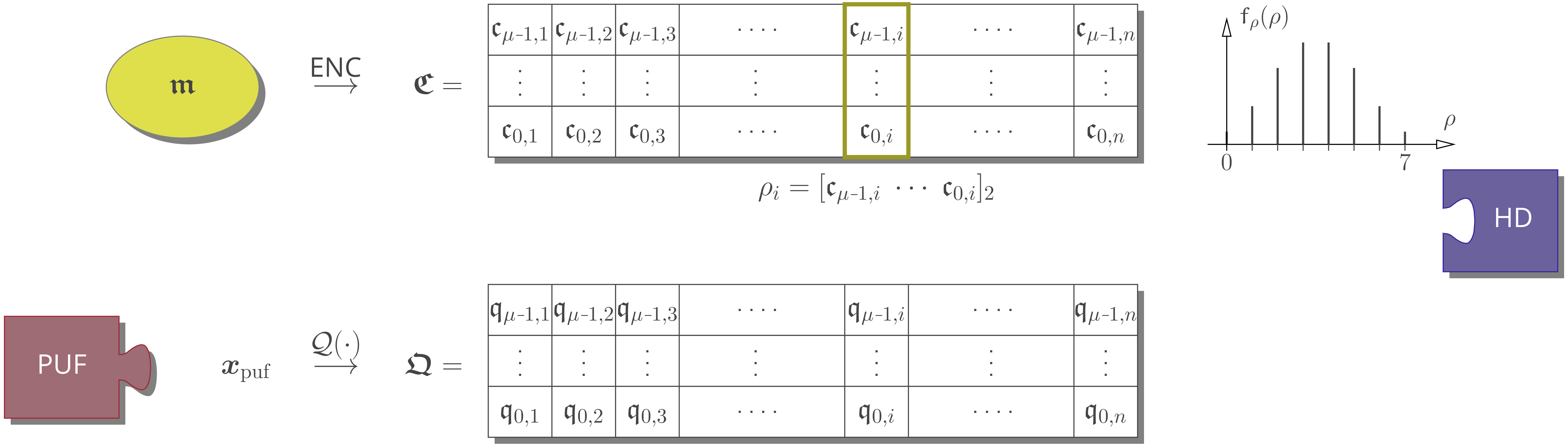
Security: it can be shown

- message can be decoded when knowing the PUF readout and the helper data
- no leakage when knowing the PUF readout only
- no leakage when knowing the helper data only



Calculation of Helper Data: shaped signaling

- visualization

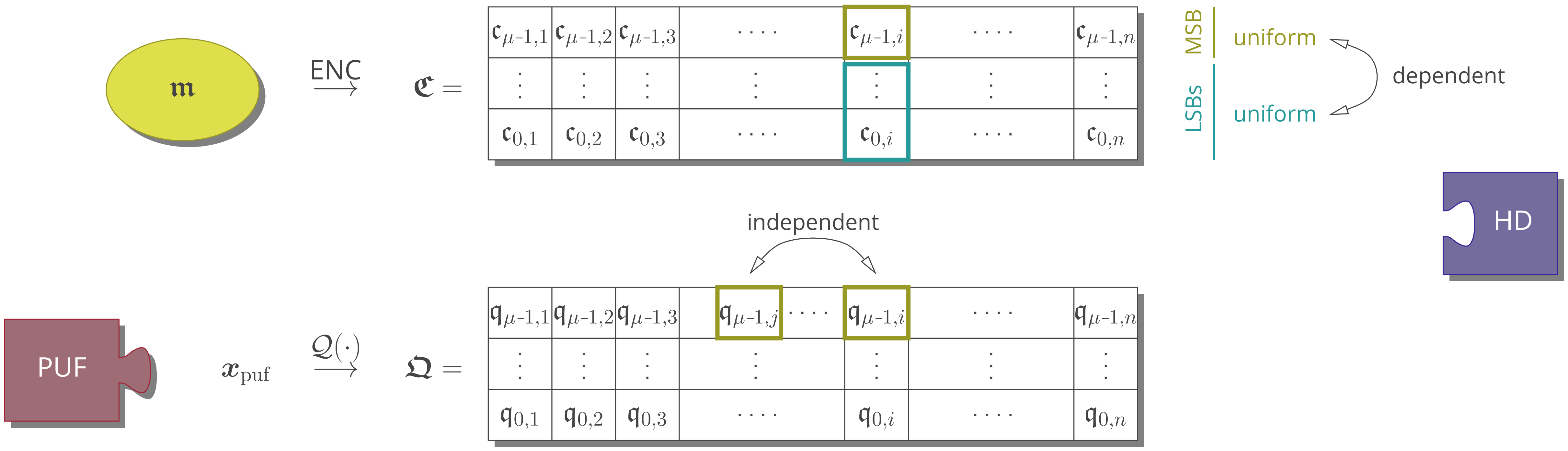


Problem:

- region numbers not uniformly distributed — leakage

Calculation of Helper Data: shaped signaling

- visualization



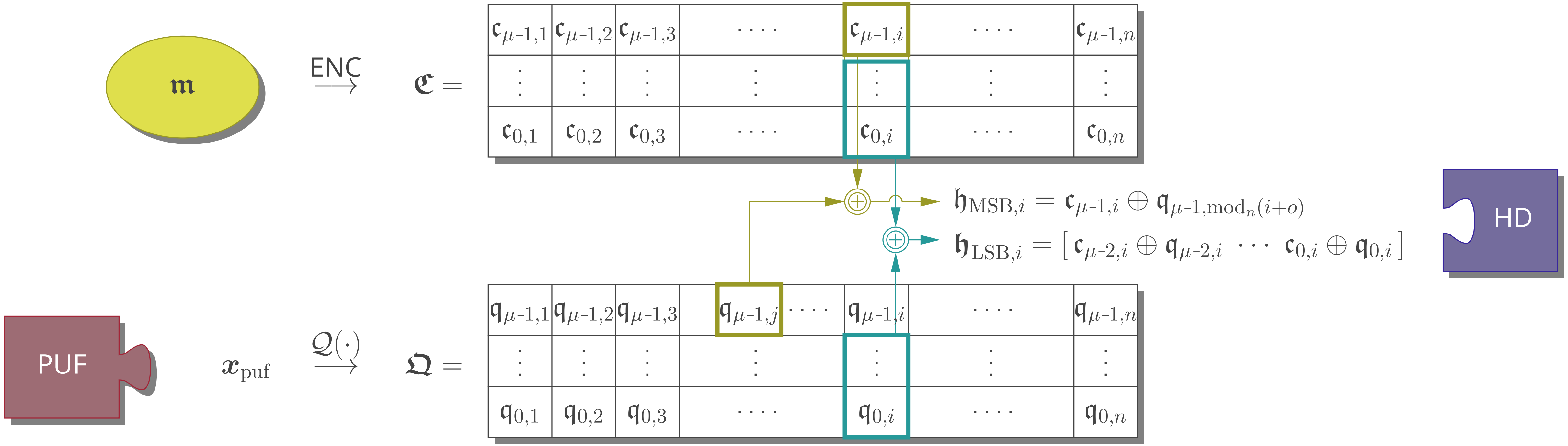
Problem:

- region numbers not uniformly distributed — leakage

Helper Data Scheme (III)

Calculation of Helper Data: shaped signaling

- visualization



Problem:

- region numbers not uniformly distributed — leakage

Solution:

- $\mathfrak{c}_{\mu-1,i} \oplus \mathfrak{q}_{\mu-1,i+o}$ independent on $[\mathfrak{c}_{\mu-2,i} \;\cdots\; \mathfrak{c}_{0,i}]$

Optimum Decoding

LLR Calculation: conversion helper scheme

- PUF readout $\boldsymbol{y}_{\text{puf}} = [y_{\text{puf},1}, \ldots, y_{\text{puf},n}]$
- LLR for label bit $\mathfrak{c}_{0,i}$

$$\text{LLR}(\mathfrak{c}_{0,i}) \;=\; \log\left(\frac{\sum_{\forall \mathfrak{q},\, \mathfrak{q}_{0,i}=\mathfrak{0} \,\oplus\, \mathfrak{h}_{0,i}} \Delta\text{Q}(y_{\text{puf},i}, \mathcal{R}_{\mathfrak{q}})}{\sum_{\forall \mathfrak{q},\, \mathfrak{q}_{0,i}=\mathfrak{1} \,\oplus\, \mathfrak{h}_{0,i}} \Delta\text{Q}(y_{\text{puf},i}, \mathcal{R}_{\mathfrak{q}})}\right)$$

- definition

$$\Delta\text{Q}(y, \mathcal{R}_{\mathfrak{c}}) \overset{\text{def}}{=} \text{Q}(D\,L_\rho - F\,y) - \text{Q}(D\,L_{\rho+1} - F\,y)$$

with $\mathcal{R}_{\mathfrak{c}} = \mathcal{R}_{[\,\mathfrak{c}_{\mu\text{-}1}\,\cdots\,\mathfrak{c}_0\,]}$ — lower limit L_ρ; upper limit $L_{\rho+1}$, $\rho = [\,\mathfrak{c}_{\mu\text{-}1}\cdots\mathfrak{c}_0\,]_2$

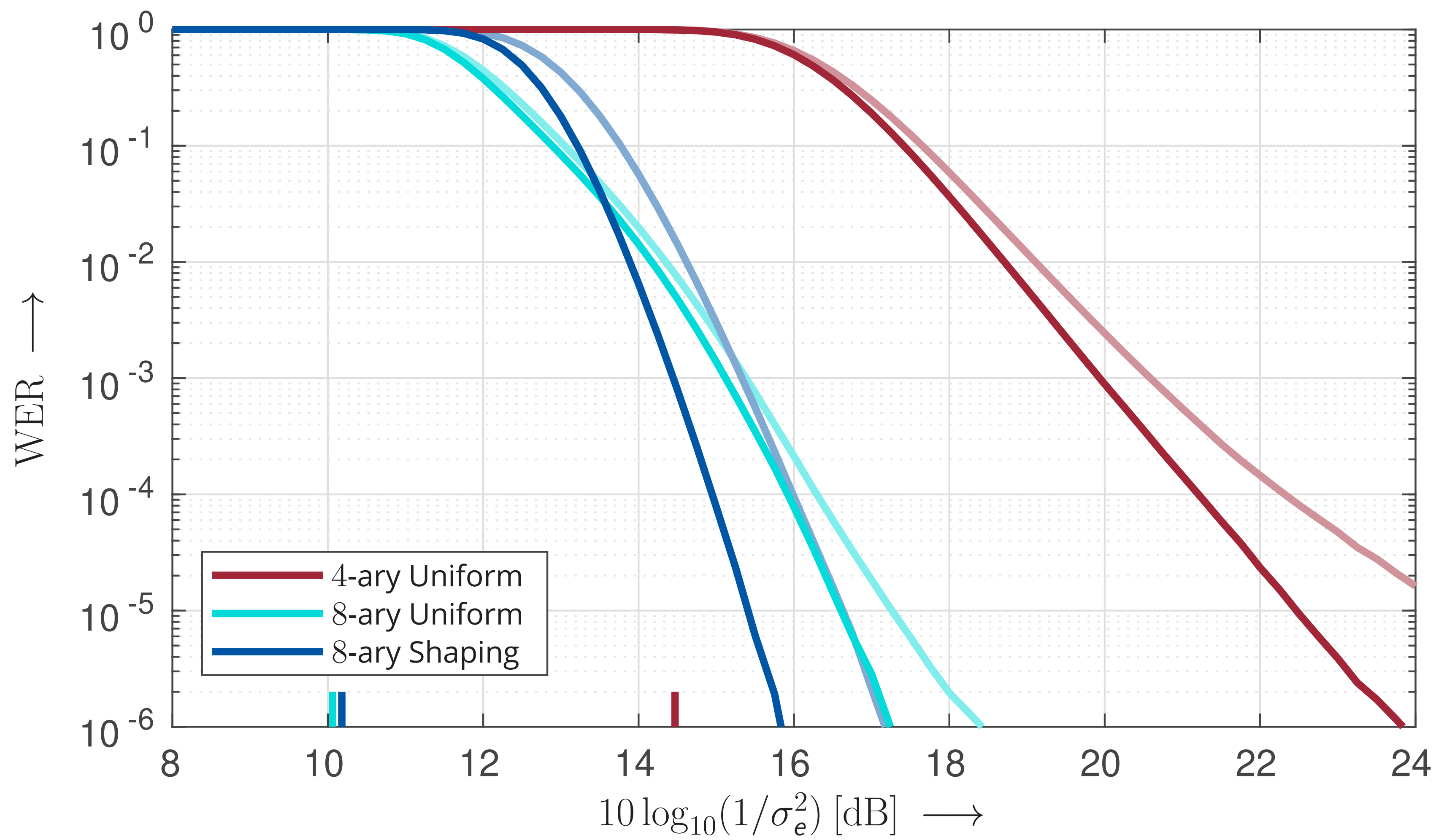
$$F \overset{\text{def}}{=} \frac{1}{\sqrt{1+\sigma_e^2}\,\sigma_e}, \qquad D \overset{\text{def}}{=} \frac{\sqrt{1+\sigma_e^2}}{\sigma_e}$$

$$\text{Q}(x) \overset{\text{def}}{=} \int_x^\infty \frac{1}{\sqrt{2\pi}}\,\text{e}^{-\frac{z^2}{2}}\,\text{d}z$$ (complementary Gaussian integral function)

Numerical Examples

Word Error Ratio (WER) over the Signal-to-Noise Ratio (in dB):

- PUF nodes: 1024
  mess. length: 1536
  rate: $R = 1.5 \; [\frac{\text{bit}}{\text{node}}]$
- Polar code
  - codelength $n = 1024$
- MLC
  - 4-ary Uniform:
    $k_0 = 523,\; k_1 = 1013$
  - 8-ary Uniform:
    $k_0 = 106,\; k_1 = 439,\; k_2 = 991$
  - 8-ary Shaping:
    $k_0 = 100,\; k_1 = 924,\; k_2 = 512$
- permutation vs. conversion

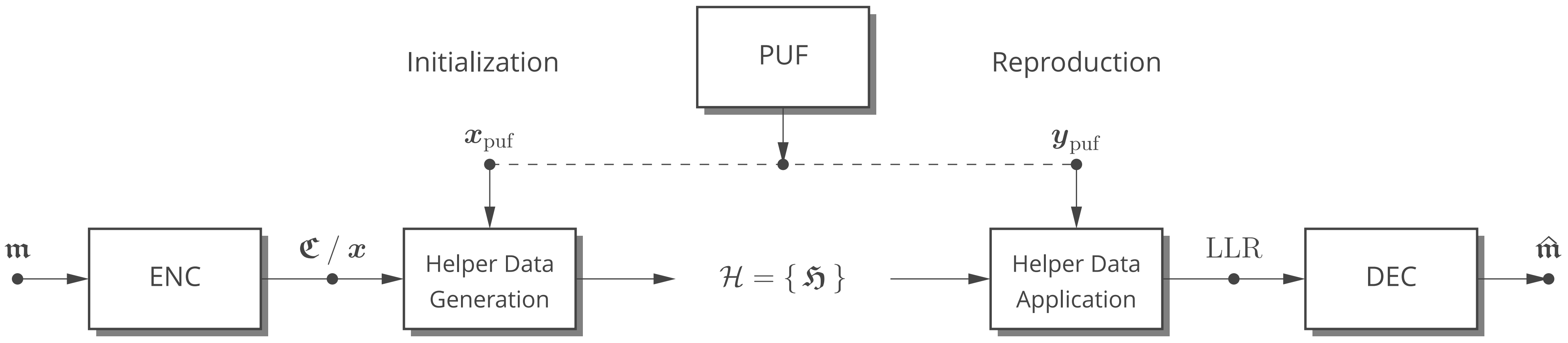


Helper Data for Improved Decoding

Situation

Coded Modulation / Shaping for PUFs:

- generation of and communication via ***helper data***



- helper data enables decoding in the first place

Improvement:

- recently, a *two-metric helper data scheme* was proposed [DGS'19], [TKDP'21]
  - two possible quantizers are available at reconstruction (uncoded case)
  - reference PUF readout determines which quantizer should be used (per PUF node)
  - these binary flags establish the helper data

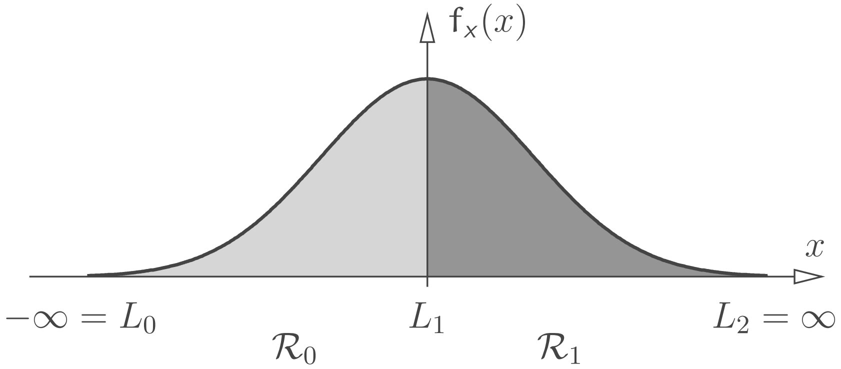
⇒ ***generalization to M-ary coded modulation***

Regions

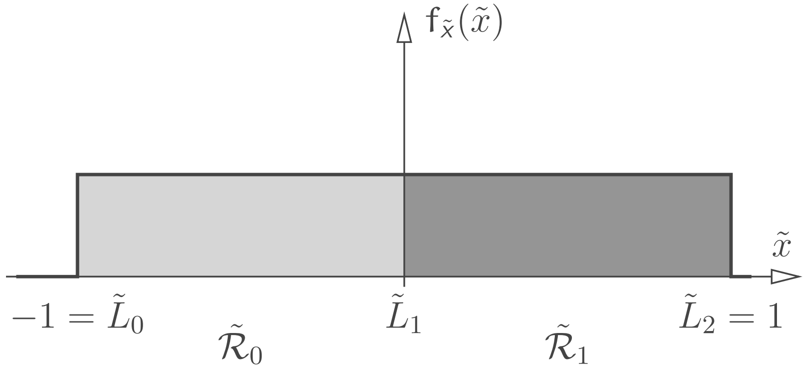
$\mathrm{erf}(z) \overset{\text{def}}{=} \frac{2}{\sqrt{\pi}} \int_0^z \mathrm{e}^{-t^2}\,\mathrm{d}t$

Regions for Uniform Signaling:

- visualization — $M = 2,\ S = 1$



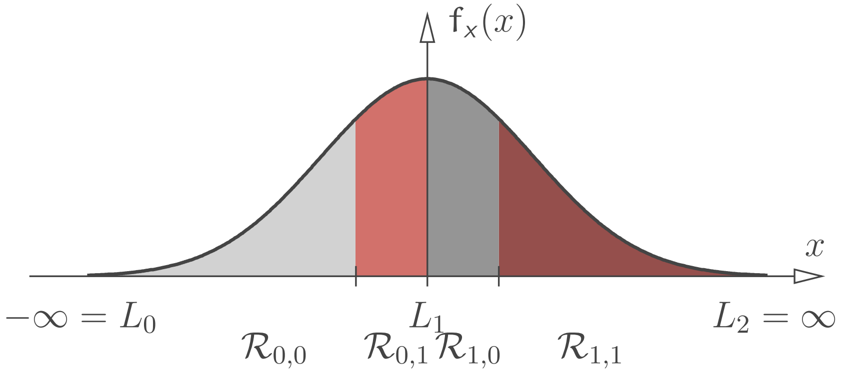
$$\tilde{x} = \mathrm{erf}(x/\sqrt{2})$$
$$x = \sqrt{2}\,\mathrm{erf}^{-1}(\tilde{x})$$

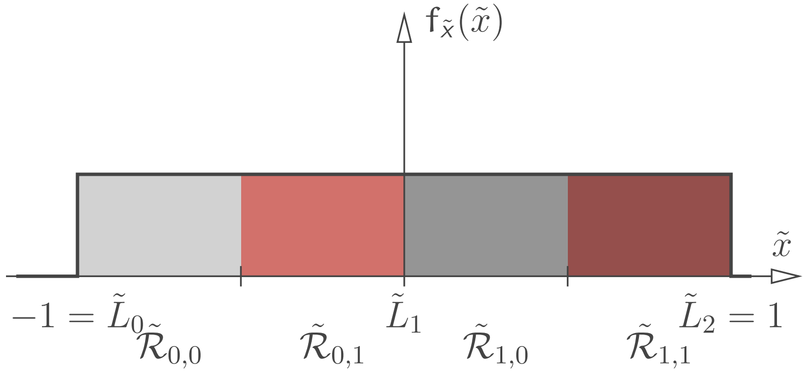


Regions

Regions for Uniform Signaling:

- visualization — $M = 2,\ S = 2$

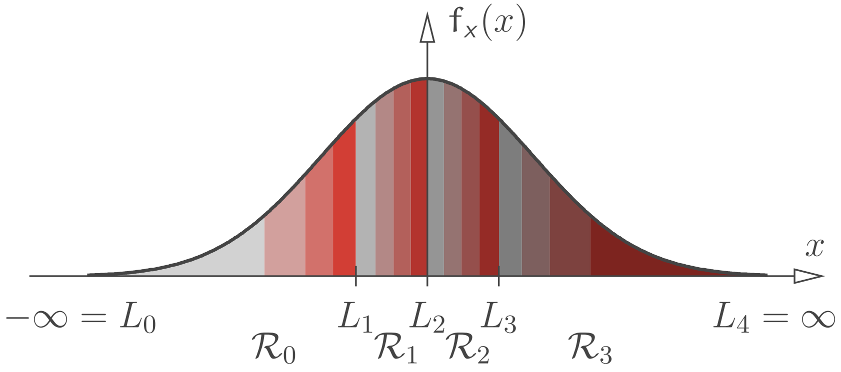


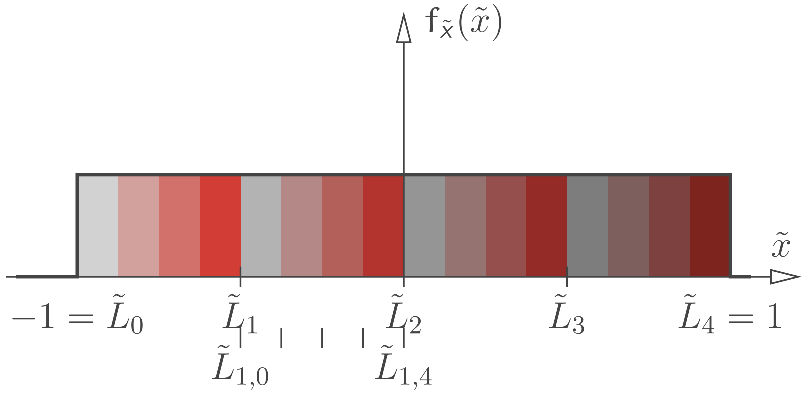


Regions

Regions for Uniform Signaling:

- visualization — $M = 4,\ S = 4$





- region limits for M-ary S-metric scheme

$$\tilde{L}_{\rho,s} = \tilde{L}_\rho + \frac{\tilde{L}_{\rho+1} - \tilde{L}_\rho}{S}\, s\,, \qquad \begin{array}{c} \rho = 0, \ldots, M-1 \\ s = 0, \ldots, S \end{array}$$

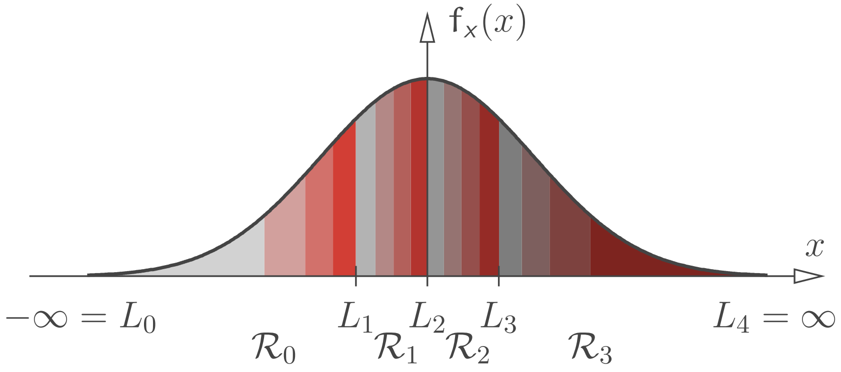
Initialization Phase:

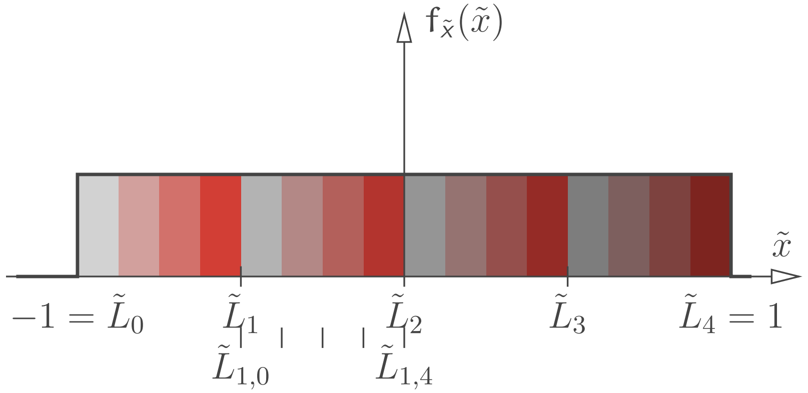
- quantization of the reference PUF readout $\boldsymbol{x}_{\text{puf}}$ (limits $L_{\rho,s}$) $\Rightarrow$ ***region ρ and subregion s***
- total helper data $\mathcal{H} = \{\, \boldsymbol{\mathfrak{H}},\ \boldsymbol{s} \,\}$ $\Rightarrow$ $n\left(\log_2(M) + \log_2(S)\right)$ bits

Regions

Regions for Uniform Signaling:

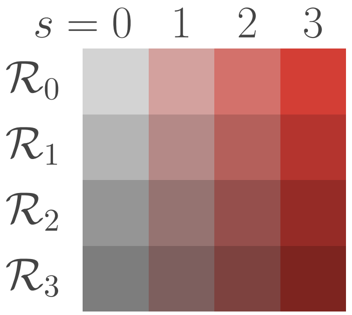
- visualization — $M = 4, \; S = 4$





Security:

- due to construction



$$\Pr\{s\} = \frac{1}{S}$$

and

$$p_{\rho,s} = \Pr\{x \in \mathcal{R}_{\rho,s}\} = \Pr\{x \in \mathcal{R}_\rho\} \, \frac{1}{S}$$

⇒ subregion number s is uniformly distributed

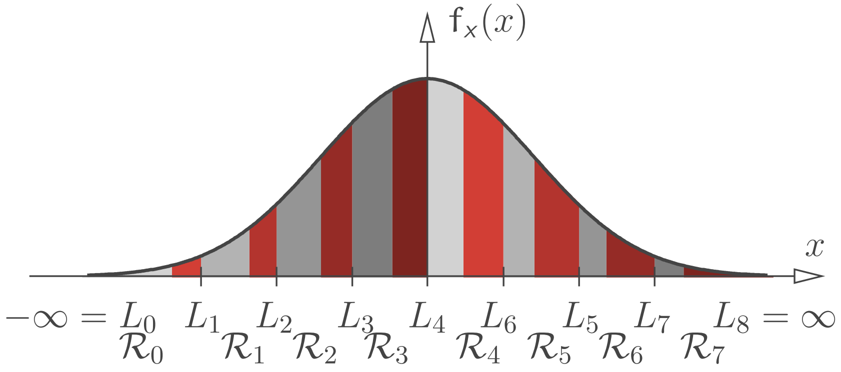
⇒ region number ρ and subregion number s are independent

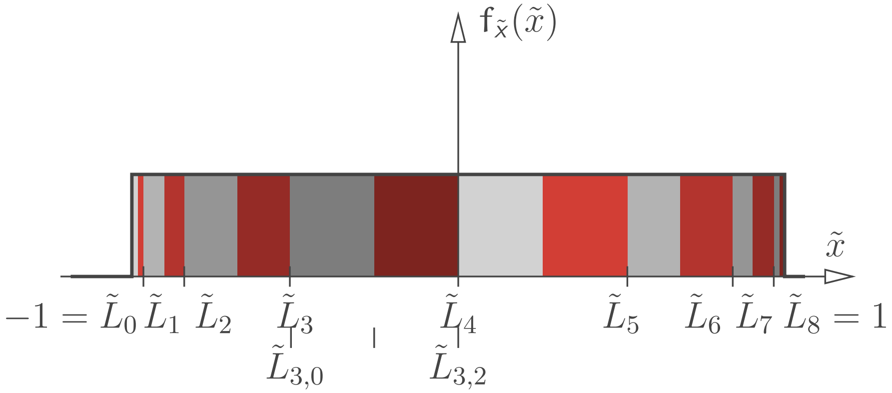
⇒ ***no leakage***

Regions

Regions for Shaping:

- visualization — $M = 8,\ S = 2$





Security:

- due to construction

$$\Pr\{s\} = \frac{1}{S}$$

and

$$p_{\rho,s} = \Pr\{x \in \mathcal{R}_{\rho,s}\} = \Pr\{x \in \mathcal{R}_{\rho}\}\,\frac{1}{S}$$

⇒ subregion number s is uniformly distributed

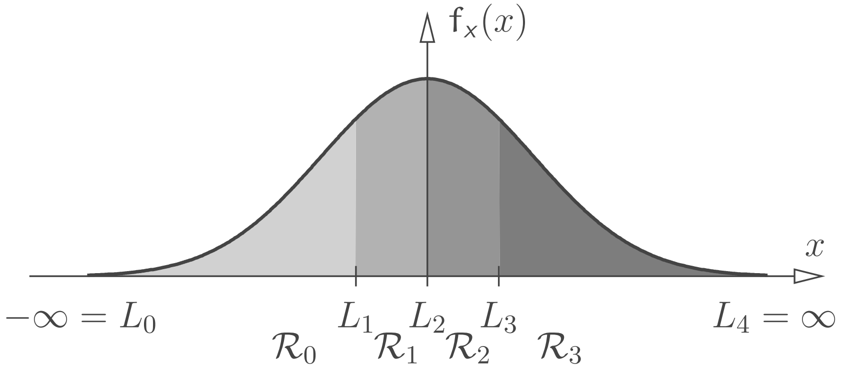
⇒ region number ρ and subregion number s are independent

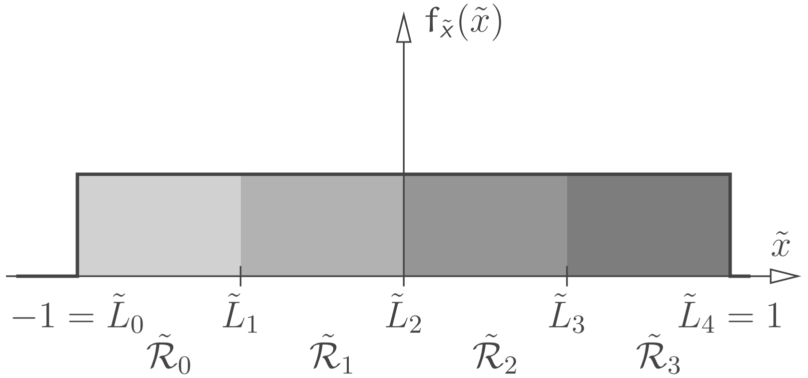
⇒ ***no leakage***

Constellations

Active Constellation: $M = 4$

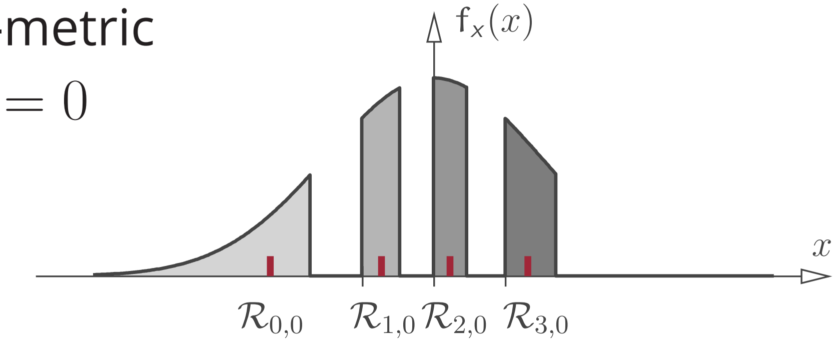
- conventional

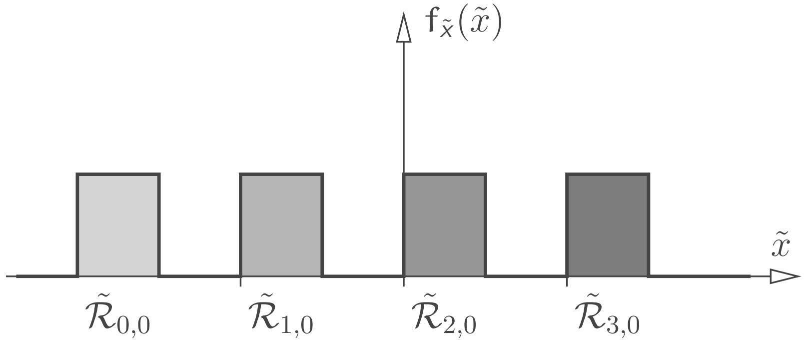




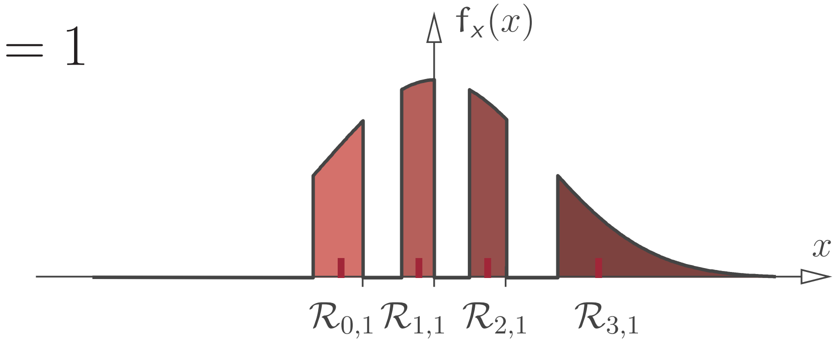
- 2-metric

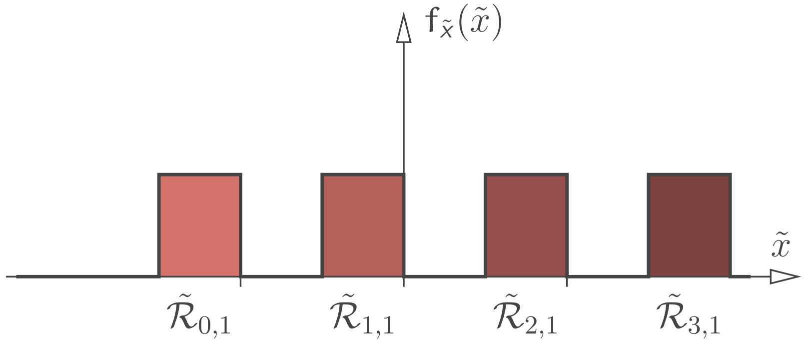
$s = 0$





$s = 1$

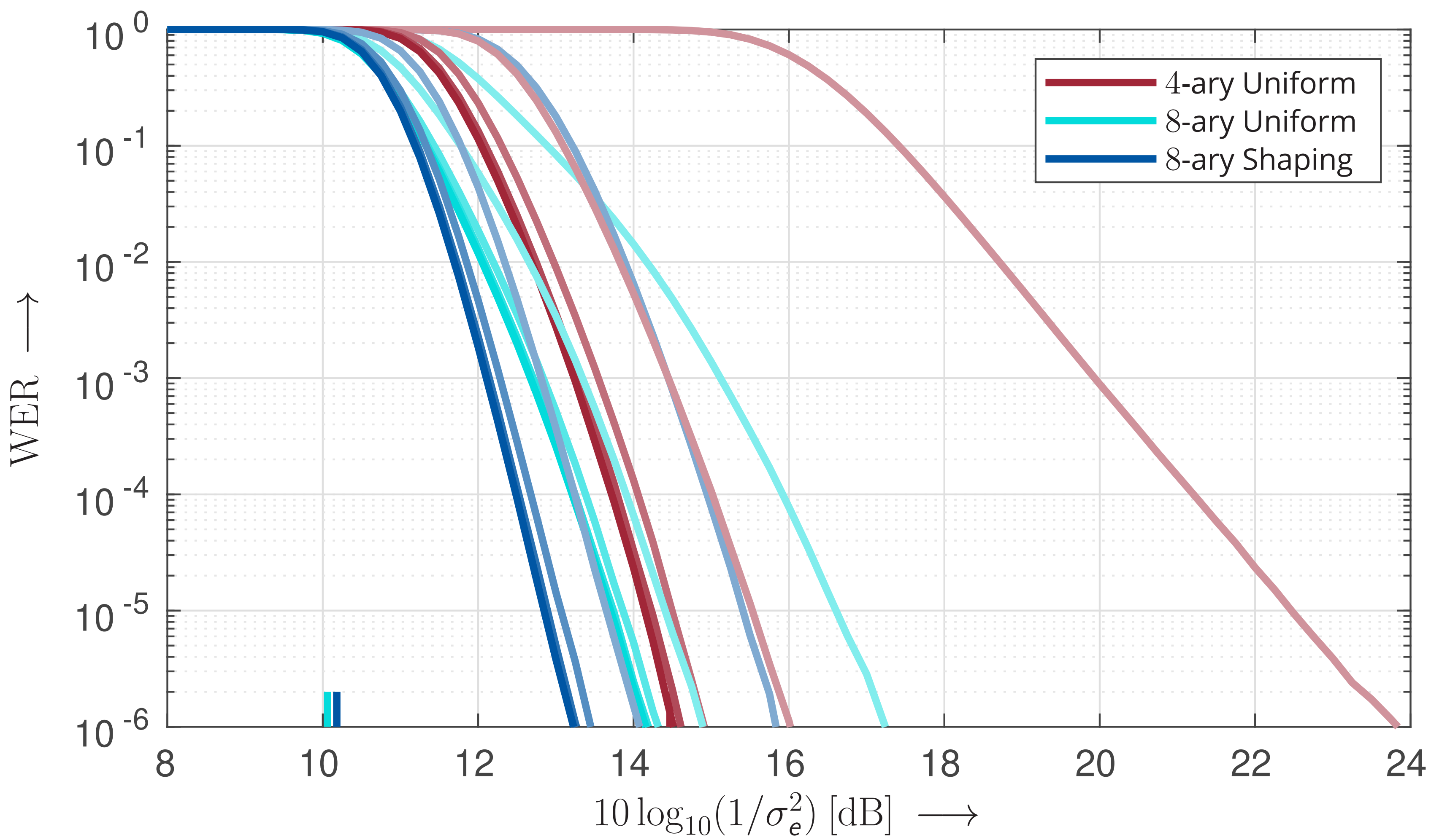




Numerical Examples

Word Error Ratio (WER) over the Signal-to-Noise Ratio (in dB):

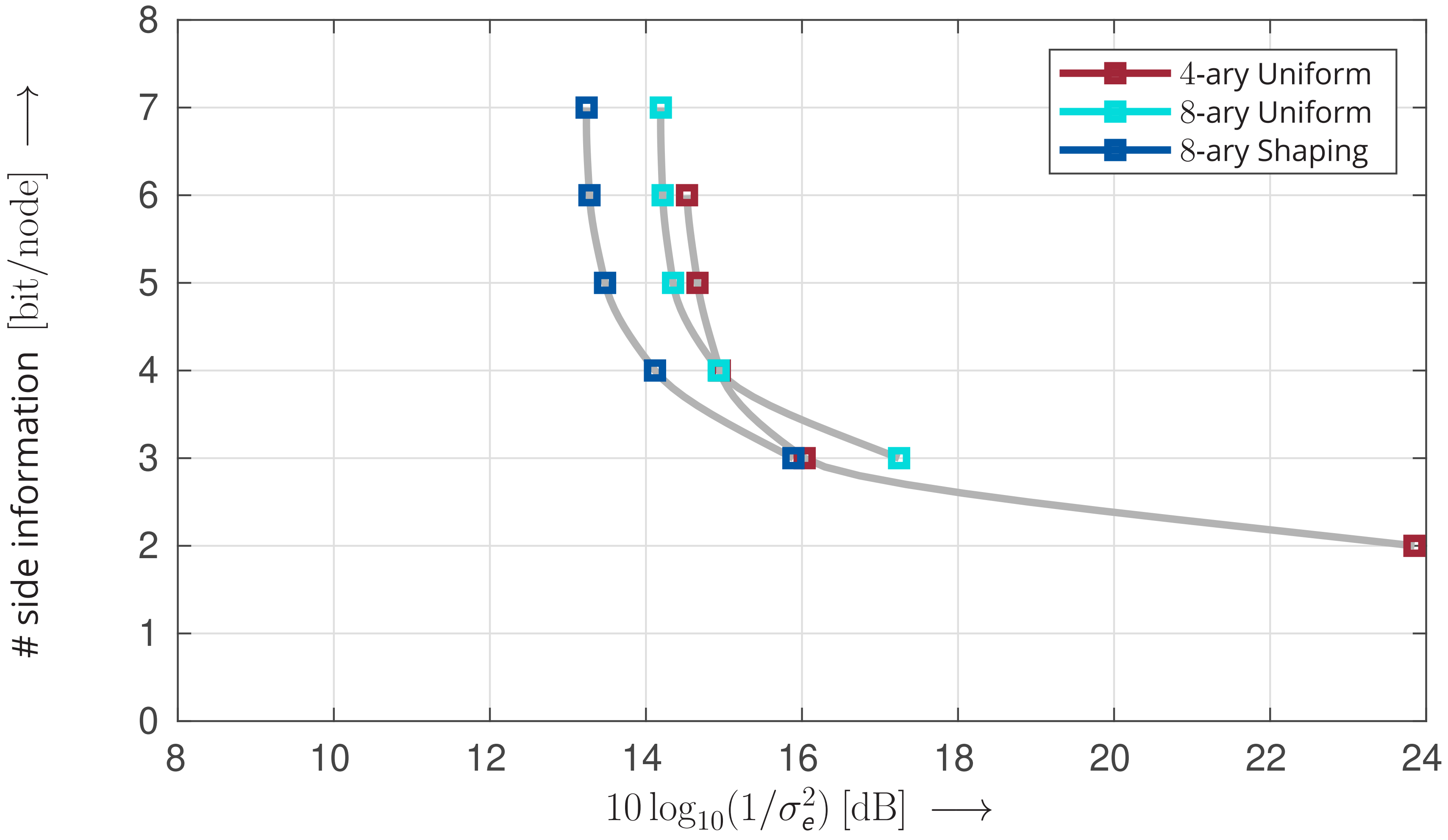
- PUF nodes: 1024
  mess. length: 1536
  rate: $R = 1.5 \ [\frac{\text{bit}}{\text{node}}]$
- Polar code
  - codelength $n = 1024$
- MLC
- conversion helper scheme
- $S = 1,\ 2,\ 4,\ 8,\ 16$



Numerical Examples (II)

 # Side Information $[\mathrm{bit/node}]$ over Required Signal-to-Noise Ratio (in dB):

- PUF nodes: 1024
  mess. length: 1536
  rate: $R = 1.5 \quad [\frac{\mathrm{bit}}{\mathrm{node}}]$
- $\mathrm{WER} = 10^{-6}$



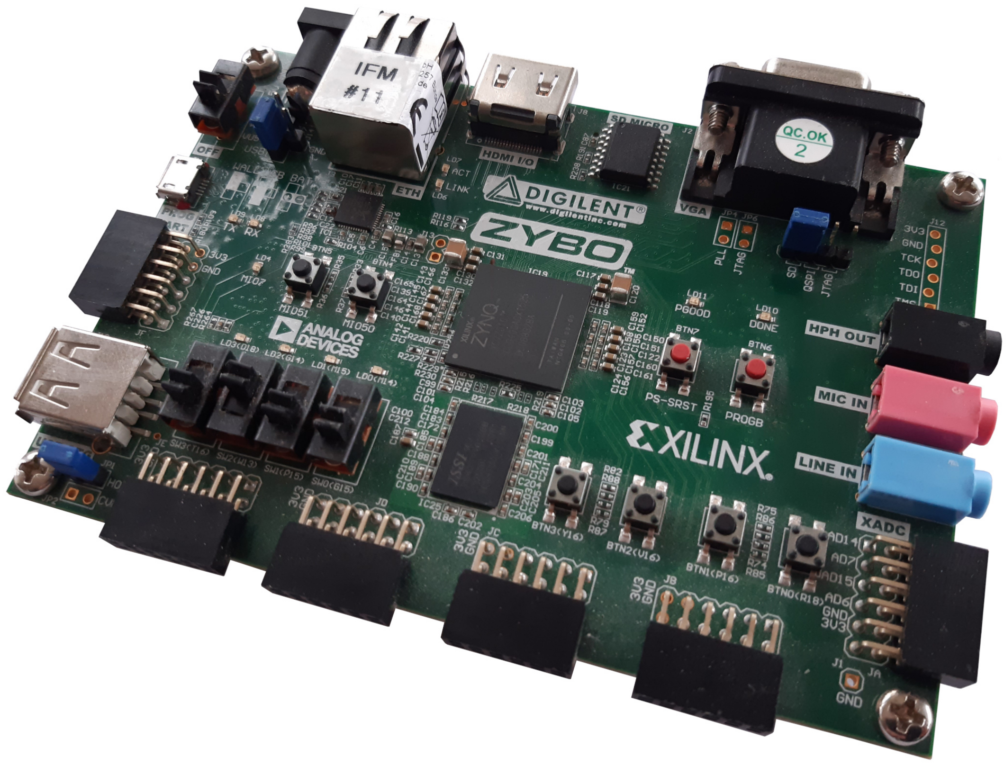
FPGA Implementation

FPGA Implementation

Specification:

[HMBO'19]

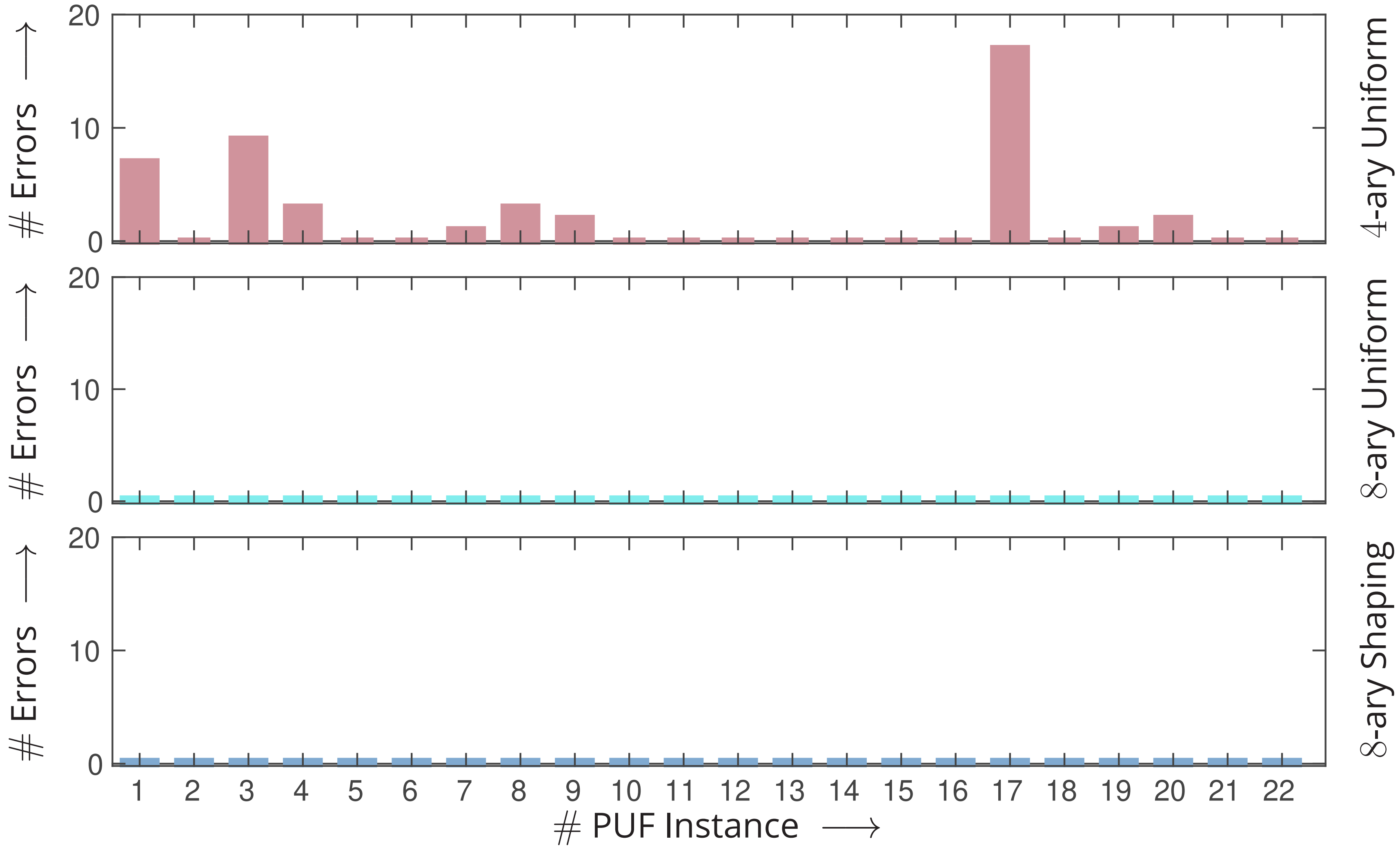
- ROPUFs implemented on XILINX FPGAs
  at the Institute of Microelectronics
- 22 instances (evaluation boards) available
- each comprising 3800 ROs
- $n = 1024$ disjoint pairs of ROs randomly selected
- temperature from $-10\ °\mathrm{C}$ to $50\ °\mathrm{C}$
  (in steps of $10\ °\mathrm{C}$)
- reference readout $\boldsymbol{x}_{\mathrm{ref}}$:
  average of 10 readouts at $20\ °\mathrm{C}$
- $10{,}000$ readouts per PUF instance and temperature
  (in total $70{,}000$ readouts per PUF instance)
- schemes
  - 4-ary uniform
  - 8-ary uniform
  - 8-ary shaping



FPGA Implementation (II)

Number of Word Errors: per 70,000 readouts

- PUF nodes: 1024
  mess. length: 1536
  rate: $R = 1.50$ $\left[\frac{\text{bit}}{\text{node}}\right]$
- $S = 1$



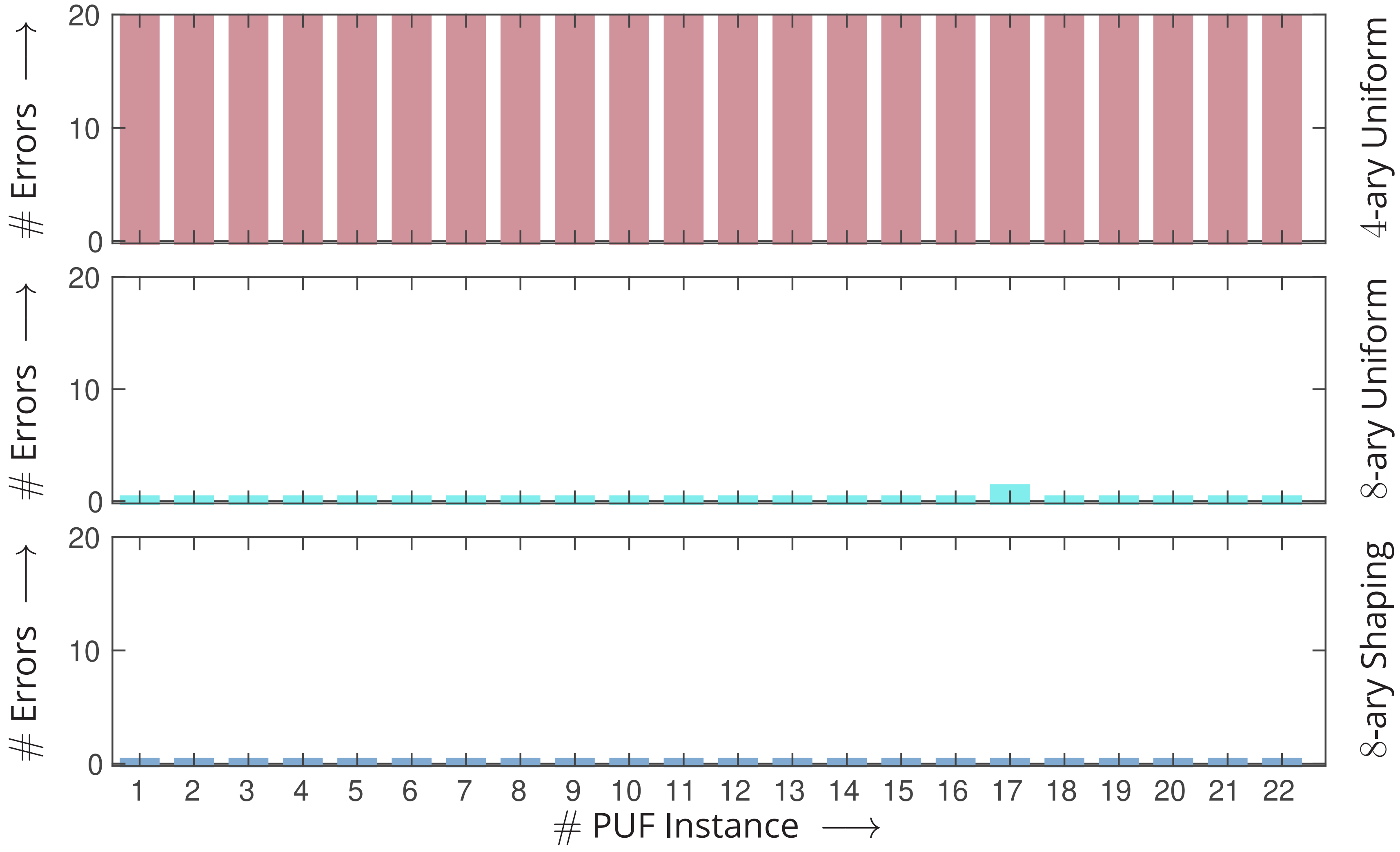
FPGA Implementation (II)

Number of Word Errors: per 70,000 readouts

- PUF nodes: 1024

  mess. length: 1792

  rate: $R = 1.75 \ \left[\frac{\text{bit}}{\text{node}}\right]$
- $S = 1$



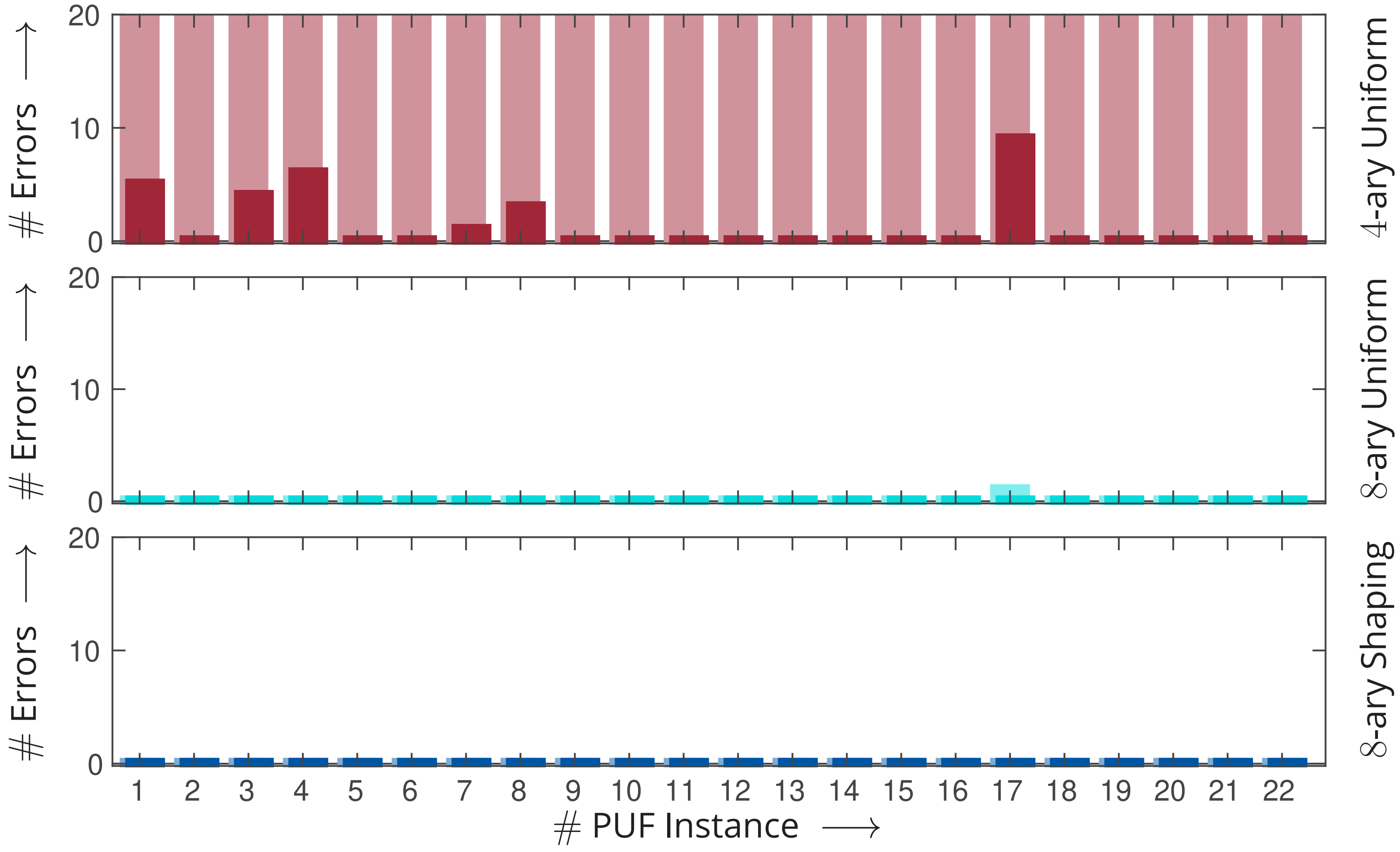
FPGA Implementation (II)

Number of Word Errors: per 70,000 readouts

- PUF nodes: 1024

  mess. length: 1792

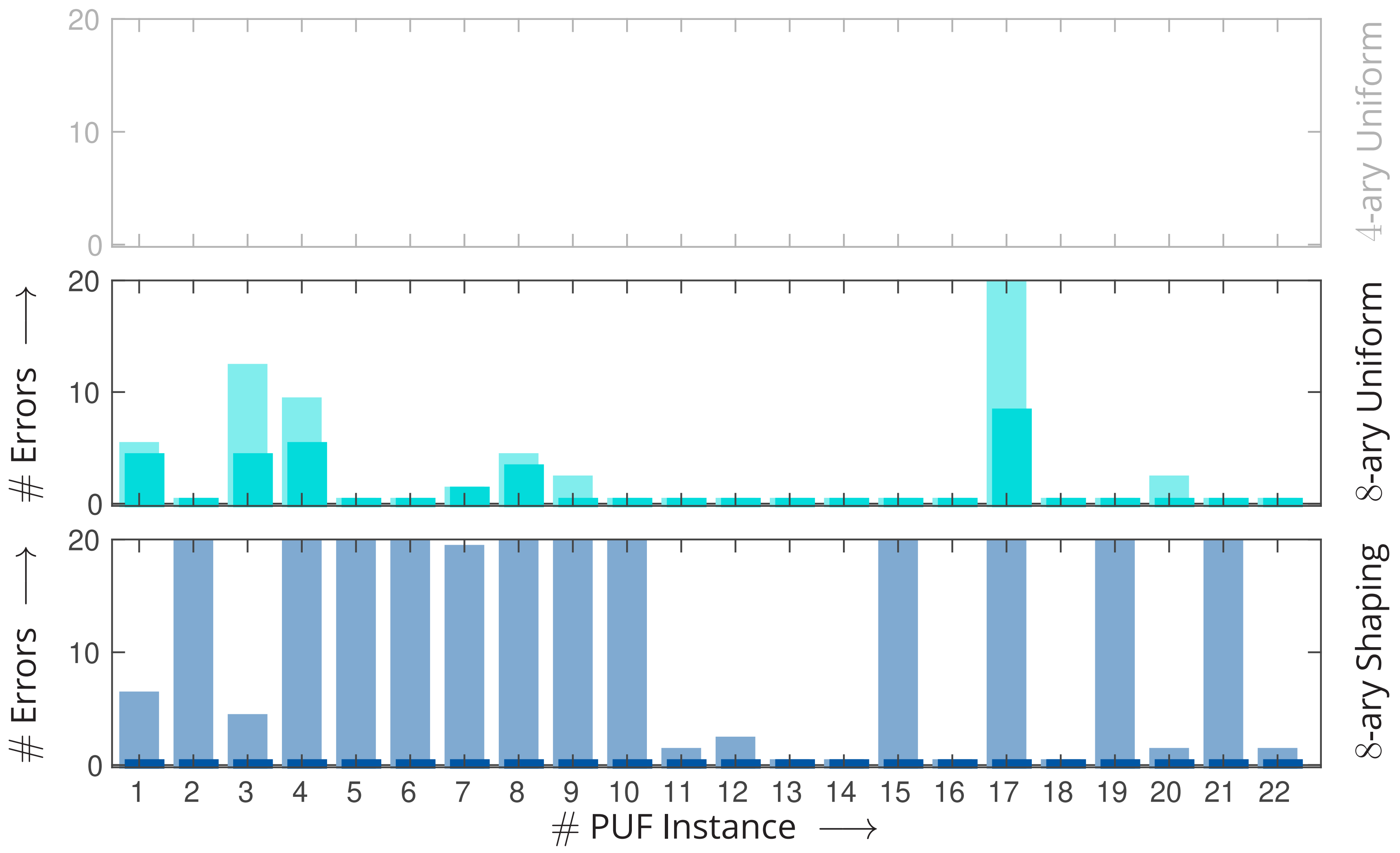
  rate: $R = 1.75$ $\left[\frac{\text{bit}}{\text{node}}\right]$
- $S = 1$ and 4



FPGA Implementation (II)

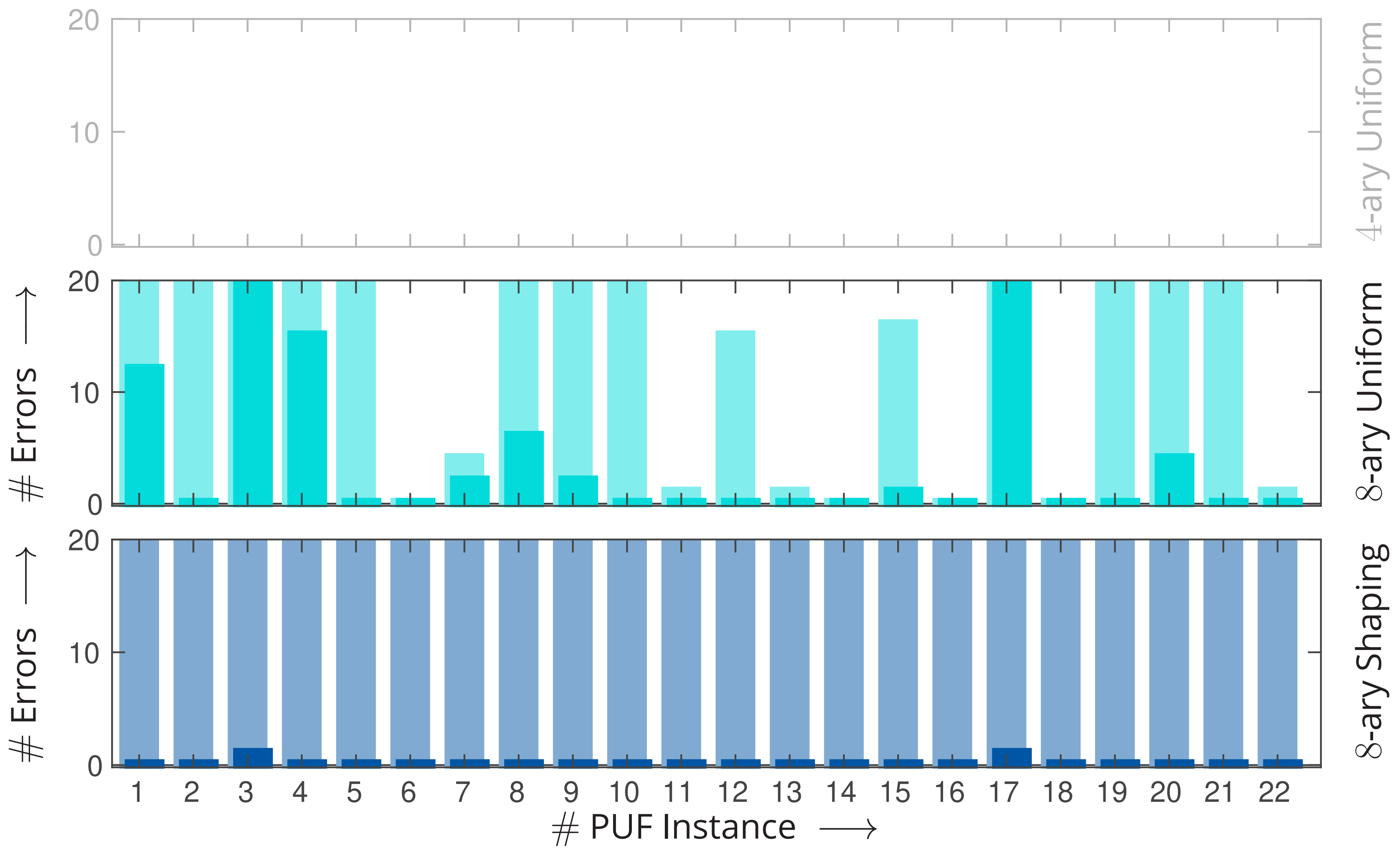
Number of Word Errors: per 70,000 readouts

- PUF nodes: 1024
  mess. length: 2048
  rate: $R = 2.00 \; [\frac{\text{bit}}{\text{node}}]$
- $S = 1$ and 4



Number of Word Errors: per 70,000 readouts

- PUF nodes: 1024
  mess. length: 2304
  rate: $R = 2.25 \; [\frac{\text{bit}}{\text{node}}]$
- $S = 1$ and 4



Summary and Outlook

Summary and Outlook

Error Correction for PUFs:

- utilizing the analog readout is rewarding
- PUF model: digital transmission with randomness at the transmitter
- design of coded modulation and shaping techniques ⇒ *increase in rate per PUF node*
- design of suited helper data ⇒ *increase in reliability*

Further Directions:

- here: Gaussian model for signal and error ⇒ *adaptation to real-world data*
- here: (silicon) PUF as hardware device ⇒ *application to “channel PUFs”*
- here: practical designs (coded modulation / helper data) ⇒ *fundamental finite-length limits*

References

References

[AC'93] R. Ahlswede, I. Csiszar. Common Randomness in Information Theory and Cryptography. I. Secret Sharing. *IEEE Transactions on Information Theory*, vol. 39, no. 4, pp. 1121–1132, July 1993.

[BH'13] C. Böhm , M. Hofer. *Physical Unclonable Functions in Theory and Practice*. Springer Science+Business Media, New York, 2013.

[BNCF'14] L. Bossuet, X. T. Ngo, Z. Cherif, V. Fischer. A PUF Based on a Transient Effect Ring Oscillator and Insensitive to Locking Phenomenon. *IEEE Transactions on Emerging Topics in Computing*, vol. 2, no. 1, pp. 30–36, March 2014.

[CBD$^+$'17] K.-H. Chuang, E. Bury, R. Degraeve, B. Kaczer, G. Groeseneken, I. Verbauwhede, D. Linten. Physically Unclonable Function Using CMOS Breakdown Position. In *IEEE Int. Reliability Physics Symposium (IRPS)*, Monterey, CA, USA, pp. 4C-1.1–4C-1.7, 2017.

[CN'00] I. Csiszar, P. Narayan. Common Randomness and Secret Key Generation with a Helper. *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 344–366, Mar. 2000.

[DGS'19] J.-L. Danger, S. Guilley, A. Schaub. Two-Metric Helper Data for Highly Robust and Secure Delay PUFs. In *IEEE International Workshop on Advances in Sensors and Interfaces (IWASI)*, pp. 184–188, 2019.

[DRS'07] Y. Dodis, L. Reyzin, A. Smith. Fuzzy Extractors: A Brief Survey of Results from 2004 to 2006. In *Security with Noisy Data: On Private Biometrics, Secure Key Storage and Anti-Counterfeiting*, pp. 79–99, Editors P. Tuyls, B. Skoric, T. Kevenaar, Springer, London, 2007.

[DRS'04] Y. Dodis, L. Reyzin, A. Smith. Fuzzy Extractors: How to Generate Strong Keys from Biometrics and other Noisy Data. In: C. Cachin, J.L. Camenisch (eds) *Advances in Cryptology – EUROCRYPT 2004*, Lecture Notes in Computer Science, vol 3027, Springer, Berlin, Heidelberg, pp. 523–540, 2004.

[FM'22] R.F.H. Fischer, S. Müelich. Coded Modulation and Shaping for Multivalued Physical Unclonable Functions. *IEEE Access*, vol. 10, pp. 99178–99194, 2022.

[Fis'24] R.F.H. Fischer. Helper Data Schmes for Coded Modulation and Shaping in Physical Unclonable Functions. *Submitted*. 2024.

[GCDD'02] B. Gassend, D. Clarke, M. van Dijk, S. Devadas. Silicon Physical Random Functions. Proceedings of the ACM Computer and Communications Security Conference, pp. 148–160, Nov. 2002.

[GI'14] O. Günlü, O. İşcan. DCT Based Ring Oscillator Physical Unclonable Functions. In *EEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, pp. 8198–8201, 2014.

[GISK'19] O. Günlü, O. Iscan, V. Sidorenko, G. Kramer. Code Constructions for Physical Unclonable Functions and Biometric Secrecy Systems. *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 11, pp. 2848–2858, Nov. 2019.

[GFBP'23] O. Günlü, R.F. Schaefer, H. Boche, H.V. Poor. Secure and Private Distributed Source Coding With Private Keys and Decoder Side Information. *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 3803–3816, 2023.

References

[HBL+'17] G. He, J.-C. Belfiore, I. Land, G. Yang, X. Liu, Y. Chen, R. Li, J. Wang, Y. Ge, R. Zhang, W. Tong. Beta-Expansion: A Theoretical Framework for Fast and Recursive Construction of Polar Codes. In *IEEE Global Communications Conference*, Singapore, 2017.

[HBO'16] A. Herkle, J. Becker, M. Ortmanns. Exploiting Weak PUFs from Data Converter Non-Linearity — E.g. A Multibit CT $\Delta\Sigma$ Modulator. *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 7, pp. 994–1004, Juli 2016.

[HMBO'19] A. Herkle, H. Mandry, J. Becker, M. Ortmanns. In-depth Analysis and Enhancements of RO-PUFs with a Partial Reconfiguration Framework on Xilinx Zynq-7000 SoC FPGAs. In *Int. Symposium on Hardware Oriented Security and Trust (HOST)*, pp. 238–247, 2019.

[IW'09] T. Ignatenko, F.M.J. Willems. Biometric Systems: Privacy and Secrecy Aspects. *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 4, pp. 956–973, Dec. 2009.

[IOK+'18] V. Immler, J. Obermaier, M. König, M. Hiller, G. Sigl. B-TREPID: Batteryless Tamper-Resistant Envelope with a PUF and Integrity Detection. In *IEEE Int. Symposium on Hardware Oriented Security and Trust (HOST)*, Washington, DC, USA, pp. 49–56. 2018.

[JW'99] A. Juels, M. Wattenberg. A Fuzzy Commitment Scheme. In *ACM Conference on Computer and Communications Security*, pp. 28–36, Nov. 1999.

[KFPW'22] C. Kestel, C. Frisch, M. Pehl, N. Wehn. Towards More Secure PUF Applications: A Low-Area Polar Decoder Implementation. In *IEEE International System-on-Chip Conference (SOCC)*, Belfast, United Kingdom, 2022.

[LT'03] J.-P. Linnartz, P. Tuyls. New Shielding Functions to Enhance Privacy and Prevent Misuse of Biometric Templates. In: J. Kittler, M.S. Nixon (eds) *Audio- and Video-Based Biometric Person Authentication*, Lecture Notes in Computer Science, vol. 2688, Springer, Berlin, Heidelberg, pp. 393–402, 2003.

[MTV'09] R. Maes, P. Tuyls, I. Verbauwhede. A Soft Decision Helper Data Algorithm for SRAM PUFs. In *IEEE International Symposium on Information Theory*, Seoul, Korea (South), pp. 2101–2105, 2009.

[MHV'12] R. Maes, A. Van Herrewege, I. Verbauwhede. PUFKY: A Fully Functional PUF-based Cryptographic Key Generator. In: E. Prouff, P. Schaumont, P. (eds) *Cryptographic Hardware and Embedded Systems (CHES), Lecture Notes in Computer Science*, vol 7428. Springer, Berlin, Heidelberg. 2012.

[Mae'13] R. Maes. *Physically Unclonable Functions: Constructions, Properties and Applications*. Springer Science & Business Media, 2013.

[MHK+'19] H. Mandry, A. Herkle, L. Kürzinger, S. Müelich, J. Becker, R.F.H. Fischer, M. Ortmanns. Modular PUF Coding Chain with High-Speed Reed-Muller Decoder. In *IEEE International Symposium on Circuits and Systems (ISCAS)*, Sapporo, Japan, Mai 2019.

[MHM+20] H. Mandry, A. Herkle, S. Müelich, J. Becker, R.F.H. Fischer, M. Ortmanns. Normalization and Multi-Valued Symbol Extraction from RO-PUFs for Enhanced Uniform Probability Distributions. *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 67, no. 12, pp. 3372–3376, Dec. 2020.

[Mau'93] U.M. Maurer. Secret Key Agreement by Public Discussion from Common Information. *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 733–742, May 1993.

References

[MSSS'12] D. Merli, D. Schuster, D. Stumpf, G. Sigl. Side Channel Analysis of PUFs and Fuzzy Extractors. In: J.M. McCune, B. Balacheff, A. Perrig, A.R. Sadeghi, A. Sasse, Y. Beres (eds) *International Conference on Trust and Trustworthy Computing, Lecture Notes in Computer Science*, vol. 6740. Springer, Berlin, Heidelberg, 2011.

[MPMHS'14] S. Müelich, S. Puchinger, M. Bossert, M. Hiller, G. Sigl. Error Correction for Physical Unclonable Functions Using Generalized Concatenated Codes. In *International Workshop on Algebraic and Combinatorical Coding Theory*, 2014.

[MPSB'19] S. Müelich, S. Puchinger, V. Stukalov, M. Bossert. A Channel Model and Soft-Decision Helper Data Algorithms for ROPUFs. In *International ITG Conference on Systems, Communications and Coding (SCC)*, Rostock, Germany, 2019.

[Müe'19] S. Müelich. *Channel Coding for Hardware-Intrinsic Security*. Ph.D. Dissertation, Universität Ulm, 2019.

[MMOF'21] S. Müelich, H. Mandry, M. Ortmanns, R.F.H. Fischer. A Multilevel Coding Scheme for Multi-Valued Physical Unclonable Functions. *IEEE Trans. Inf. Forensics Security*, vol. 16, pp. 3814–3827, 2021.

[PRTG'02] R. Pappu, B. Recht, J. Taylor, N. Gershenfeld. Physical one-way functions" (PDF). Science, vol. 297, no. 5589, pp. 2026–2030, Sep. 2002.

[PMBHS'15] S. Puchinger, S. Müelich, M. Bossert, M. Hiller, G. Sigl. On Error Correction for Physical Unclonable Functions. In *ITG Conference on Systems, Communications and Coding (SCC)*, Hamburg, Germany, 2015.

[SN'00] A.-R. Sadeghi, D. Naccache. *Towards Hardware-Intrinsic Security*. Springer, Berlin, Heidelberg, 2010.

[TKDP'21] L. Tebelmann, U. Kühne, J.-L. Danger, M. Pehl. Analysis and Protection of the Two-metric Helper Data Scheme. Cryptology ePrint Archive, Paper 2021/830, 2021. In: S. Bhasin, F. De Santis (eds) *Constructive Side-Channel Analysis and Secure Design*, Lecture Notes in Computer Science, vol. 2910, Springer, Cham, pp. 279–302, 2021.

[Teb'22] L. Tebelmann. *Side-Channel Analysis and Countermeasures for Physical Unclonable Functions*. Ph.D. Thesis, Technische Universität München, 2023.

[LLB'13] C. Ling, L. Luzzi, M.R. Bloch. Secret Key Generation from Gaussian Sources Using Lattice Hashing. In *IEEE International Symposium on Information Theory*, Istanbul, Turkey, pp. 2621–2625, 2013.

[TSB+'06] P. Tuyls, G.-J. Schrijen, B. Škorić, J. Van Geloven, N. Verhaegh, R. Wolters. Read-Proof Hardware from Protective Coatings. In *Int. Workshop on Cryptographic Hardware and Embedded Systems*, pp. 369–383, 2006.

[WHGS'16] O. Willers, C. Huth, J. Guajardo, H. Seidel. MEMS Gyroscopes as Physical Unclonable Functions. In *ACM SIGSAC Conference on Computer and Communications Security*, pp. 591–602, 2016.

[ZPK+'16] S.S. Zalivaka, A.V. Puchkov, V.P. Klybik, A.A. Ivaniuk, C.-H. Chang. Multi-Valued Arbiters for Quality Enhancement of PUF Responses on FPGA Implementation. In *IEE Asia and South Pacific Design Automation Conference (ASP-DAC)*, Macao, China, pp. 533–538, 2016.