RESEARCH ARTICLE

Coded Modulation and Shaping for Multivalued Physical Unclonable Functions

ROBERT F. H. FISCHER, (Senior Member, IEEE), AND SVEN MÜELICH
Institute of Communications Engineering, Ulm University, 89081 Ulm, Germany

Corresponding author: Robert F. H. Fischer (robert.fischer@uni-ulm.de)
This work was supported by the Deutsche Forschungsgemeinschaft (DFG), German Research Foundation under Grant FI 982/15-1.

ABSTRACT In this paper, a coded modulation scheme employing signal shaping is proposed and designed for use in multi-valued physical unclonable functions (PUFs). In the literature, usually binary PUFs are considered, where less than one bit of entropy is extracted from each fundamental building block (PUF node). This randomness in the hardware may then be used for the derivation of cryptographic keys. Meanwhile, PUFs over higher-order alphabets, so-called multi-valued PUFs (MVPUs) have been presented. Typically, symbols from a higher-order alphabet are extracted by suited quantization. In contrast, in this paper we propose a coded modulation scheme which directly works on the analog values provided by the PUF nodes. We establish an interpretation as digital transmission with randomness at the transmitter, and thereon calculate the respective capacities of bit levels for the Gaussian distributed reference PUF readout in additive Gaussian noise. In addition, the metric for optimum soft-decision decoding is derived. Thereby, the coded modulation scheme can be designed for the present situation. A helper data scheme adapted to the given situation is proposed and analyzed. Moreover, on top of the multilevel coding scheme, we employ signal shaping for the first time in the context of PUFs. In particular, trellis shaping is applied. By this, the coded modulation scheme can be perfectly matched to the PUF statistics. This is reflected by the results of conducted numerical simulations. High coding rates can be used, i.e., a large entropy per PUF node can be extracted, with very high reliabilities.

INDEX TERMS Coded modulation, signal shaping, physical unclonable functions, multi-valued PUF.

I. INTRODUCTION
Physical unclonable functions (PUFs) are hardware primitives, where randomness, which is intrinsically present in the hardware due to uncontrollable imperfections in the manufacturing process, is exploited for the derivation of cryptographic keys. It is impossible to produce a PUF that generates a pre-determined readout. We address so-called “weak” PUFs, where a unique fingerprint is delivered based on the properties of the hardware. In contrast, in “strong” PUFs the response is dependent on a challenge.

Since the exploited randomness is static over the PUFs lifetime, a PUF can be interpreted as an implicit key storage. However, extracted PUF readouts slightly vary due to external conditions like, e.g., changes of temperature. Thus, error-correcting codes are mandatory to guarantee the stability of the final keys. Helper data schemes are used to map a readout to a codeword to enable error correction. The most prominent scheme for this approach is the code-offset algorithm [1], [2], [3].

Most often, PUFs that produce binary readouts are considered in the literature, even when based on an analog source. Traditionally, a PUF node1 is modeled as a binary symmetric channel (BSC) with a fixed bit error probability, see, e.g., [8], [9], [10]. Various studies revealed that for some PUF constructions this assumption is too pessimistic, and more

1. We call the basis building block for extracting randomness, i.e., any piece of hardware delivering a single (continuous or discrete) random variable, a PUF node, cf., e.g., [4]. Other denominations are PUF cell, cf., e.g., [5], [6] (which must not be understood as “memory cell”) or PUF unit, cf., e.g., [7]. An array of PUF nodes delivers the readout, from which, by means of an error correcting procedure, the final response is derived.
suited channel models have been adopted [11], [12]. The binary PUFs in [5] directly employ the analog PUF readouts as input to a soft-decision decoding algorithm.

When PUFs extract randomness from analog sources, a single PUF node might contain more entropy than one bit. Consequently multi-valued PUFs (MVPUs), which extract a symbol from a finite alphabet of cardinality greater than two from each single PUF node have been considered in the literature.

References [13] and [14] outline how MVPUs can be constructed. The given principles and ideas can in general be applied, provided that the raw data can be accessed. References [15] and [16] exploit CMOS gate oxide breakdown locations as entropy source for proposing an MPVU construction, where each PUF node delivers an analog value in the interval [0, 1].

Reference [17, Fig. 5.1] divides methods for reliability enhancement into error correction and error reduction. Methods for error correction have been published in [18], [19], and [4]. References [18] and [19] combine an equidistant quantization with a variable-length bit mapping. The probability density function describing the PUF readouts is divided into 14 equidistant intervals, each representing a symbol. Each symbol is mapped to a binary sequence employing a variable-length prefix-free code, which is designed such that the Levenshtein distance between neighboring symbols is 1. The XOR-operation in the code-offset algorithm is replaced by a parity approach, since it does not support sequences of different lengths, which might occur in case of an insertion or deletion of a symbol. For error correction a Varshamov–Tenengolts code is applied.

In [4], equi-probable quantization is applied to the raw PUF readouts. A binary Gray code and classical error correction over the Hamming metric is used. Alternatively, equi-distant quantization of the raw PUF readouts is applied and different scenarios therefore are considered. As further method, [4] proposes to model the quantized symbols (alphabet of cardinality \( q \)) as a \( q \)-ary symmetric channel and to apply \( q \)-ary error-correcting code over the Hamming metric and over the Lee metric. In [20], equi-probable and discrete equi-probable quantization are applied and analyzed, while error correction is not treated.

In [21], spatial context tree weighting was introduced for assessing unpredictability by means of entropy estimation. [22] and [23] deal with ternary PUFs, based on binary Latch PUFs. PUF nodes are classified into always 0 cells, always 1 cells, and random cells, where the classification into one of these categories is determined by performing multiple readouts. In [22], the binary code-offset scheme is extended to a ternary version where addition/subtraction over the Galois field \( \mathbb{F}_3 \) replaces the XOR operation and ternary codes instead of binary codes are used. In [23], a ternary extended von Neumann corrector is proposed for debiasing the responses.

Other approaches for deriving multiple bits from each PUF node by quantisation, can be found, e.g., in [24], [25], [26], [27], and [28]; the list is not exhaustive. Note that also approaches exists which group bits extracted from several PUF nodes to one multi-valued symbol, e.g., [29], [30].

In contrast to the schemes discussed above, our approach is to directly utilize the analog, non-quantized output of a PUF node. The term “multi-valued” is then used synonymously for extracting more than one bit of entropy per PUF node. This is done implicitly by the (soft-decision) decoding procedure and not by quantization. To that end, we interpret the readout process in PUFs as digital transmission schemes (cf. [5, Fig. 6]) and, based on that interpretation, coded modulation/signal shaping are applied.

The contribution of the present paper is as follows:

i) We establish an interpretation as digital transmission with randomness at the transmitter. Based on this, the capacities of the bit levels for the Gaussian distributed reference PUF readout in additive Gaussian noise are calculated and the sum capacity, which is the ultimate limit for the information rate which can be extracted, are derived. Thereby, the differences to conventional digital transmission (as used in [5]) are enlightened and the design of the coded modulation scheme can be matched to the present situation. This also includes the derivation of the metric for optimum soft-decision decoding.

ii) A new helper data scheme, which operates with the shaping scheme, is proposed and its security is analyzed.

iii) A coded modulation scheme with signal shaping [31] is presented; specifically, as an initial example, multilevel codes [32], [33] in combination with trellis shaping [34] are employed. For the first time, signal shaping is applied in the context of PUFs. By this, the (envisaged) useful signal is Gaussian distributed and, thus, reflects the statistics of the actual PUF readout. As an alternative to signal shaping, uniform signaling with a non-uniformly spaced constellation is employed. Please note that the paper focuses on the theoretical interpretation and capacity limits, and the proposal of a particular well-suited coded modulation/shaping scheme. A comprehensive comparison with other coded modulation/shaping schemes is part of future research. Moreover, attacks and hardware/implementation aspects such as long-term stability/reliability of the PUF cells are beyond the scope of the paper.

The paper is structured as follows: Sec. II reviews MVPUs based on ring oscillators and the statistics of the analog readout. A new helper data scheme for coded modulation is proposed and its security is proven. In Sec. III, a new interpretation as digital transmission scheme is given; signal and channel model are established and coded modulation/shaping schemes are designed. To that end, the capacities of the coding levels are studied. Sec. IV presents results from numerical simulations. The statements derived in Sec. III based on capacity arguments are covered by word-error-rate
simulations employing practical codes. The paper is concluded in Sec. V. The derivation of the channel capacity based on a new interpretation of what is considered as useful signal is given in Appendix A.

II. MULTI-VALUED PHYSICAL UNCLONABLE FUNCTIONS

We consider PUF architectures, where access to an analog, i.e., real-valued, physical quantity is possible. A particular example are ring oscillator PUFs (ROPUFs), introduced in [35], and also studied in [36] and [5]. First, the basic principle is briefly reviewed and the error model is established. Then, a helper data scheme suited for analog readouts and the envisaged coded modulation scheme (to be detailed in Sec. III) is introduced. Finally, the security of this approach is analyzed.

A. DATA AND ERROR MODEL OF RING OSCILLATOR PUFs

In ROPUFs, each PUF node consists of two ring oscillators (ROs); an RO is a loop consisting of an odd number of inverters. Due to this construction, the circuit oscillates with a frequency which, due to uncontrollable variations when manufacturing the device, varies around an average frequency. Usually, cf. [35], the sign of the frequency difference of the ROs determines the information extracted from a PUF node. However, it is more appropriate to consider the frequency difference as an analog quantity.

As described in [5], at the Institute of Microelectronics at Ulm University, a set of ROPUFs has been implemented and an exhaustive measurement campaign has been conducted. Detailed information about the structure of the underlying hardware can be found in [36]. By averaging over a (fixed, small) number of readouts, the nominal difference frequency of each PUF node is determined. The derived statistics show that the nominal frequency difference (when going over the PUF nodes) is nearly Gaussian distributed. Removing a possible mean and normalizing the nominal frequency differences suitably (for details see [5]), the normalized readout (reference frequency), denoted as $r_{\text{ref}}$, can thus be modeled to be zero-mean Gaussian with variance $\sigma_r^2 = 1$.

Given a particular PUF node, repeated readouts will not give exactly the same real number of the nominal frequency. Instead, over a wide range of physical conditions (e.g., the environmental temperature), the readout is given as its nominal value—which we defined to be the error-free signal—plus an superimposed error. This error, denoted as $e_{\text{ref}}$, is very well modeled by a zero-mean Gaussian random variable which is independent of the nominal (error-free) readout. Hence, we have a reproduced readout

$$r = r_{\text{ref}} + e_{\text{ref}}.$$  

Combining the readouts $r_i, i = 1, \ldots, n$, of $n$ PUF nodes into a vector, we have the readout word

$$r = [r_1, \ldots, r_n]^T \in \mathbb{R}^n,$$

which is given by

$$r = r_{\text{ref}} + e_{\text{ref}}. \tag{1}$$

Thereby, we assume that the elements of the vectors are i.i.d., i.e., all PUF nodes are independent of each other and have the same statistics. All processing and coding is done on this PUF readout.

We remark, that the subsequent discussion is based on this generic model. It might not be restricted to ROPUFs and the setting of [5]. For any PUF architecture where the analog source of randomness is studied, due to the law of large numbers, a Gaussian signal model and a Gaussian error model are reasonable assumptions. Any mean can be removed as it conveys no information [38].

B. HELPER DATA SCHEME

As it is well-known in the PUF literature, in order to provide a stable, unique PUF readout, i.e., to combat the error $e_{\text{ref}}$, some form of channel coding has to be applied. Since the nominal/reference readout $r_{\text{ref}}$ will (in general) never be a valid codeword from a chosen code, a so-called helper data scheme (HDS) is employed.

1) BASIC PRINCIPLE

For binary, hard-decision readout, most often the code-offset algorithm [1], [2], [3] or the syndrome construction [2], [3], are applied. Also pointer-based schemes [39], [40], [41], which store pointers to reliable PUF nodes, hence, improving the overall channel, have been proposed. Schemes, based on the code-offset algorithm, which are suited for $M$-ary signaling are given in [5].

In view of the coded modulation/shaping scheme to be discussed in Sec. III, a different approach is required. To that end, Fig. 1 visualizes the processing in the two phases initialization and reproduction.

FIGURE 1. Block diagram of the initialization phase (top), i.e., generation of the helper data, and the reproduction phase (bottom), i.e., application of the helper data.

$^3$In [5], different to the present paper, the nominal readout was normalized to a variance $\sigma_r^2 = 5$. This also affects the variance of the error.

$^4$We distinguish between scalars (normal font) and (column) vectors (bold font).

$^5$Memory effects (as, e.g., considered in [37]) are beyond the scope of the paper as we deal with weak PUFs based on ROs and infrequent readouts.
In the initialization (or enrollment) phase, which is done once in a secure environment at rollout of the device, first, the reference readout \( r_{\text{ref}} \) is extracted. Second, a \( k \)-bit secret message\(^6 \) \( m \in \mathbb{F}_2^k \) is generated at random, uniformly over all possible binary vectors of length \( k \). This message is encoded using the desired coded modulation/shaping scheme into a codeword \( a \in s \mathcal{A}^n \subset \mathbb{R}^n \) in signal space, where \( \mathcal{A} = \{ \pm 1, \pm 3, \ldots, \pm (M - 1) \} \) is an \( M \)-ary bipolar amplitude-shift keying (ASK) signal constellation and \( s \) a suitably chosen scaling factor (details are given in Sec. III). Finally, having \( r_{\text{ref}} \) and \( a \), the helper data (HD) \( \mathcal{H} \) is generated; \( \mathcal{H} \) is stored publicly.

In the reproduction phase, the (noisy) PUF readout \( r \) is extracted. The known helper data is applied to produce the word \( y \)—the HDS is constructed such that this word is the sum of the valid codeword \( a \) in signal space and additive Gaussian noise. Hence, the subsequent decoder can decode the word and provide an estimate \( \hat{m} \) of the message.

In the classical code-offset algorithm for binary coding, the helper data is a binary word of length \( n \) which is added (in the arithmetic of the finite field \( \mathbb{F}_2 \)) to the hard-decision PUF readout. Using binary codes with soft-decision decoding, the helper data may be a word from \( \{ \pm 1 \}^n \) which is multiplied (element-wise, over the real numbers) to the real-valued PUF readout \( y \), i.e., performing a sign flip. Employing coded modulation/shaping methods, such a procedure is not sufficient in order to guarantee that \( \hat{m} \) can be decoded based on \( y \).

### 2) HELPER DATA SCHEME FOR CODED MODULATION

We propose to use a signed permutation matrix \( P \) as helper data \( \mathcal{H} \). Such a matrix contains a single plus or minus one in each row and column. It represents a permutation (reordering) of a vector together with a possible sign change of the elements. Since different combinations of permutation and sign flipping may lead to a desired result, the choice of \( P \) (contrary to the classical code-offset algorithm) may be not unique—from the set of possible permutation matrices, the actual may be chosen at random. Hence, in the initialization a further source of randomness may be present. In the reproduction phase, simply \( y = Pr \) is calculated. The complexity of this operation is negligible and \( P \) can be stored very compactly.\(^7 \)

A suited procedure for establishing the signed permutation matrix is visualized in Fig. 2. On the one hand, the elements of the codeword \( a \in \mathcal{A}^n \) are sorted ascending according to their magnitude; the respective signed permutation matrix (negative elements in \( a \) are multiplied by \(-1\); positive by \(+1\)) is denoted as \( P_a \). Hence, this permutation matrix is a deterministic function of \( a \). Within the groups of equal magnitude (dashed lines), a further (random, unsigned) permutation is done; the respective permutation matrix \( P_b \) has block diagonal structure.

On the other hand, the elements of the reference readout are sorted ascending according to their magnitude (signed permutation matrix \( P_r \)). Both sorted versions, \( P_a, P_b, a \) and \( P_r r_{\text{ref}}, \) respectively, provide the best match. Since the processed PUF readout \( y \) in the reproduction phase should mimic a (noisy) codeword in signal space, the best choice for the requested signed permutation matrix \( P \) is going the cascade from bottom to top. As the inverse of a (signed) permutation matrix is its transpose, the cascade

\[
P = P_a^T P_b^T P_r
\]

permutes and possibly inverts the elements in the requested way and establishes the entire permutation matrix.

### 3) CHANNEL MODEL

When applying the signed permutation matrix to the noisy PUF readout we have

\[
y = Pr = P(r_{\text{ref}} + e_{\text{ref}}) = Pr_{\text{ref}} + Pe_{\text{ref}}
\]

with the obvious definitions of the useful (error-free) signal \( x \) and the error \( e \). Note that the elements of both vectors are still zero-mean Gaussian as the elements of \( r_{\text{ref}} \) and \( e_{\text{ref}} \) are.

The signed permutation does not change the statistics. The elements of \( x \) have variance \( \sigma_x^2 = 1 \) and that of \( e \) have variance \( \sigma_e^2 \).

### C. SECURITY OF THE HELPER DATA SCHEME

We now analyze the security of the proposed HDS. As described above, the \( k \)-bit message \( m \) is chosen uniformly at random. Encoding (and mapping) is a one-to-one function and gives \( a = \text{ENC}(m) \). The matrix \( P_a \) is chosen deterministically based on \( a \), the matrix \( P_b \) is chosen randomly,
only depending on the frequencies of the magnitudes, and the matrix $\mathbf{P}_t$ is chosen deterministically based on $r_{\text{ref}}$.

1) DECODABILITY

We first show that knowing the PUF readout $r_{\text{ref}}$ (ideal, noise-free case) and the helper data $\mathcal{H} = \mathbf{P}$ the message is known. To that end, we study the mutual information $I(\cdot; \cdot)$ [38] and take $x = Pr_{\text{ref}}$ and the one-to-one relation between $a$ and $m$ into account, leading to

$$I(m; r_{\text{ref}}, P) = I(a; x) = k. \quad (4)$$

The last equation holds since $a$ can be decoded from $x$ and since $m$ has an entropy of $k$ bit.

2) NO LEAKAGE WHEN KNOWING THE PUF READOUT ONLY

Second, since the message $m$ is drawn independently from the readout $r_{\text{ref}}$, by definition, we have

$$I(m; r_{\text{ref}}) = 0; \quad (5)$$

no information about $m$ can be extracted when only $r_{\text{ref}}$ is known.

3) NO LEAKAGE WHEN KNOWING THE HELPER DATA ONLY

Finally, we consider the case when only the (public) helper data $\mathcal{H} = \mathbf{P}$ is known. Since $m$ determines $a$, by definition of the mutual information (H(\cdot): entropy) we have

$$I(m; P) = I(a; P) = H(P) - H(P | a). \quad (6)$$

The entropy of a signed permutation matrix is $H(P) = \log_2(2^n - n!) \overset{\text{def}}{=} H_0$. Since $P$ contains the factor $P_0$ which depends on $r_{\text{ref}}$ and the random part $P_b$, knowing $a$ does not reduce the entropy of $P$. Formally,

$$H(P | a) = H(P_0^T P_1^T P_1 | a) = \overset{(s)}{=} H_0 + H(P_0^T P_1 | P_a)$$

$$= H(P_0^T P_1) = H_0, \quad (7)$$

where $(s)$ holds since $P_a$ is a deterministic function of $a$. Hence, we finally have

$$I(m; P) = 0. \quad (8)$$

III. CHANNEL CODING AND SHAPING

In this section, we design coded modulation/shaping schemes suited for the use in MVPUFs. On the one hand, we enlighten the similarities and differences to classical coded modulation, on the other hand, we discuss the similarities and differences to well-established binary hard-decision decoding for PUFs.

A. SIGNAL AND CHANNEL MODEL

As we have seen in the last section, when applying the proposed helper data scheme based on permutations and sign flips, the processed readout is given by (3)—a word $x$ with zero-mean, unit-variance Gaussian elements, plus an superimposed zero-mean Gaussian noise (error) $e$ with variance $\sigma_e^2$. Thereby, the processed readout $x$ at the reproduction phase should mimic the codeword $a$ chosen at the initialization phase.

In classical PUFs, the analog readout would be quantized; all $x \geq 0$ are said to represent a binary 1 and all $x < 0$ a binary 0. A hard-decision decoder for the employed binary channel code is run to produce the estimated message $\hat{m}$.

Using soft-decision, i.e., directly the analog readout, the situation changes. The straightforward approach, also taken in [5], is to consider the PUF readout as a noisy version of a binary phase-shift keying (BPSK) signal, calculating log-likelihood ratios (LLRs) as for BPSK over the AWGN channel, and using these as the input to a soft-input decoder. Even though the performance is very good, this is not the optimum procedure (see [5, Footnote 6]). Going to higher-order constellations and, consequently, to coded modulation, the mismatch between the PUF statistics and that of employed code becomes even more severe.

In order to derive the optimum decoding metric and a suited design of the coded modulation/shaping scheme, a close look at the signal model has to be taken. To that end, in Fig. 3, the situation in PUFs (Gaussian readout, left) and conventional $M$-ary ASK signaling with a scaled (factor $s$) version of $A = \{\pm 1, \pm 3, \ldots, \pm (M - 1)\}$ (right) are compared. In BPSK (2-ary ASK) signaling, the binary 1 may be represented by the real number $+1$; the binary 0 by $-1$. The probability density function (pdf) is discrete; visualized with Dirac deltas. Considering the Gaussian PUF readout, a binary 1 is represented by any real number larger or equal to zero, i.e., any number from the region $R_1 = \{x \mid x \geq 0\}$, and a binary 0 is represented by any real number smaller than zero, i.e., from the region $R_0 = \{x \mid x < 0\}$. The binary information is thus not represented by a single real number, but can be seen as being randomly (following the Gaussian distribution within the region) drawn from the region, which corresponds to the binary information which should be communicated. Hence, similar as in schemes for physical-layer security [42], randomness at the transmitter is present.

This principle generalizes to higher-order ($M$-ary) modulation schemes. In 4-ary ASK signaling, bit pairs are represented by 4 amplitude levels, i.e., the constellation $\{\pm s, \pm 3s\}$ is used. Thereby, $s$ is chosen to adjust the transmit power. Considering the PUF readout, the bit pairs are represented by the regions $R_{\infty} = \{x \mid x < -2s\}, R_{s1} = \{x \mid -2s \leq x < 0\}, R_{10} = \{x \mid 0 \leq x < 2s\}$, and $R_{11} = \{x \mid x \geq 2s\}$, respectively. Given the bit pair to be communicated, the actual number which is used for transmission is drawn according to the Gaussian pdf within the respective region.

Employing 8-ary signaling, eight regions ($R_{\infty}$ up to $R_{111}$) are defined in the same way (cf. Fig. 3). In contrast, 8-ary ASK with signal shaping, i.e., with an imposed non-uniform probability distribution (visualized by the heights of the Dirac pulses), has eight discrete amplitude levels.

Note that in each case we employ natural labeling where the signal points/regions are numbered in order. For the
present setting of one-dimensional constellations this is identical to labeling according to set partitioning [43].

B. MULTILEVEL CODING AND SHAPING

In the literature, various approaches to coded modulation, i.e., the combination of channel coding and modulation/mapping to the signal space, are present. The most prominent ones are multilevel coding (MLC) [32], [33], trellis-coded modulation (TCM) [43], and bit-interleaved coded modulation (BICM) [44]. Additionally, various approaches to signal shaping are available, in particular trellis shaping [31], [34], shell mapping [45], [46], [47], and shaping via a source decoder [48], where a constant composition distribution matcher [49] is of particular interest. For an overview on coded modulation see, e.g., [50], and for signal shaping [31].

Using signal constellations of large size, the combination of coded modulation and shaping is straightforward as they operate on different bits labeling the signal points [33], [34]. This leads to a great flexibility; any kind of (component) codes can be used together with any kind of shaping technique. Alternatively, the approach presented in [51] can be used. There, shaping and coding are more tightly linked.

We now present a suited approach to coded modulation/shaping. Due to its flexibility in adjusting the code rate and the possibility of straightforward combination with shaping to match the non-uniform probabilities over the regions, we employ MLC (cf. also [5]) in combination with trellis shaping. The block diagram of the transmitter is depicted in Fig. 4 (top) for an 8-ary ASK constellation which we will consider subsequently.

In MLC, the code in signal space is composed of so-called component codes at individual coding levels. We expect binary component codes with codeword length $n$. In slight abuse of notation, the index $i = 1, \ldots, n$ enumerating the code symbols/vector elements is denoted as time step; since the code length $n$ is identical to the number of PUF nodes which are treated jointly, the index equivalently enumerates the PUF nodes.

The message word $m \in \{0, 1\}^k$ of $k$ information symbols is partitioned into $3 = \log_2(8)$ words $u_0, u_1, u_2$ of length $k_0, k_1, k_2$, and $n/2$, respectively. To that end $k_0 + k_1 + n/2 = k$ has to hold; the rate of the coded modulation scheme is $R = k/n$. The words $u_0$ and $u_1$ are encoded individually (into the words $e_0$ and $e_1$) using binary error-correcting codes with codeword length $n$ and rates $R_0 = k_0/n$ and $R_1 = k_1/n$, respectively.

Trellis shaping, here in the form of sign-bit shaping [34], is preferably based on two-dimensional (QAM) constellations. Consequently, for the operation of shaping two consecutive time steps are combined. Coding works on words of length $n$ and one-dimensional symbols, shaping on words of length $n/2$ and two-dimensional symbols; a QAM constellation is generated by combining two ASK constellations.

In trellis shaping, the words $u_2$ and $w$ are together scrambled in the scrambler $S(D)$, which is a linear dispersive system over the finite field $\mathbb{F}_2$ with two input and output ports. The two generated binary symbols $s_1$ and $s_2$ establish the most significant address bits. Using the 8-ary ASK mapping $a = s, M(c_2 | c_1 | c_0)$ which maps the bit triples to a signal point (as indicated in the right bottom part of Fig. 3) twice, four coded bits and the two shaping bits are mapped to two ASK signal points. This procedure is visualized in Fig. 5.

The task of a shaping encoder (not shown in Fig. 4) is, knowing $u_0$, $u_1$, and $u_2$, to select the $n/2$-bit word $w$.

\[ a = s, M(c_2 | c_1 | c_0) \]

We follow the usual convention, that transfer functions of linear dispersive systems over a finite field are written in $D$-transform notation. The input vector $u_2 = [u_{2,1}, u_{2,2}, \ldots, u_{2,n/2}]$ of finite length $n/2$ can equivalently be written in $D$-notation as $u_2(D) = u_{2,1} + u_{2,2}D + \cdots + u_{2,n/2}D^{n/2-1}$; the same for $w$. Filtering is done over the time steps, or in $D$-domain as $S(D) s_1(D) = [m(D) u_2(D)] S(D)$. The produced sequences $s_1(D)$ and $s_2(D)$ can finally again be written as vectors.

FIGURE 3. Regions of the pdf of the PUF readout (left) together with the natural labeling. Top to bottom: binary, 4-ary, and 8-ary scheme. For comparison: signal constellations (right). Top to bottom: BPSK, 4-ary ASK with uniform signaling, and 8-ary ASK with signal shaping, i.e., non-uniform probabilities.

FIGURE 4. Block diagram of the encoder/decoder of the coded modulation/shaping scheme. Top: Multilevel coding in combination with trellis shaping. Bottom: Multistage decoding and inversion of the scrambling. Depicted for an 8-ary ASK constellation.
FIGURE 5. Mapping in the multilevel coding/trellis shaping scheme.

(shaping redundancy), such that the codeword \( a \) in signal space has least norm, i.e., least average transmit power. This demand immediately imposes a discrete Gaussian (aka Maxwell–Boltzmann distribution) distribution. However, other shaping aims are possible by suitably adapting the metric used in the shaping encoder, providing potential flexibility. The search can efficiently be implemented using the Viterbi algorithm. For an extensive survey on signal shaping we refer to [31, Ch. 4].

Please note that the structure of the modulation/shaping scheme is the same as it would be for the AWGN channel, except that the component codes have different rates.

The corresponding receiver structure is depicted in the bottom of Fig. 4. It performs multistage decoding (MSD) and undoes the scrambling used for shaping. The component codes are successively decoded starting with the code at Level 0. Thereby, \( \mathcal{M}_0^{-1} \) denotes the “demapping”, i.e., the calculation of the decoding metric for this code. Knowing the decoding result (simultaneously an estimate \( \hat{c}_0 \) on the codeword and an estimate \( \hat{u}_0 \) on the information word is produced), the code at Level 1 can be decoded. Thereby, in the metric calculation \( \mathcal{M}_1^{-1} \), the decoding result has to be taken into account. Finally, knowing \( \hat{c}_0 \) and \( \hat{c}_1 \), the highest level can be processed. Combining the results of two consecutive (one-dimensional) time steps, the scrambling can be inverted (scrambling matrix \( \mathbf{S}^{-1}(D) \)). As \( \mathbf{w} \) carries only shaping redundancy, this part can be ignored. After multiplexing the decoding results \( \hat{u}_l \) of all levels, the estimate \( \hat{m} \) on the message is obtained.

Please note that the MLC/MSD scheme, exemplarily shown for an 8-ary ASK constellation, directly translates to all other constellation sizes. For uniform 4-ary transmission only the two coding levels are present; the shaping part is removed. For binary transmission, only the lowest binary coding level is used. For \( (M > 8) \)-ary signaling a respective number of additional coding levels are inserted.

We finally remark that the proposed procedure of matching the probability distribution of the shaped constellation to the PUF readout is in some sense dual to what is usually done in (binary) PUFs. There, usually source coding (e.g., in the form of syndrome coding) is applied after digitization, cf. [52]. Thereby, redundancy is eliminated. In contrast, here the analog data remains as it is and the coded modulation scheme is adapted to the statistics of this readout.

C. UNIFORM SIGNALING

Using the above proposed shaping scheme, the probabilities of the signal points \( a \) reflect the probabilities within the associated region. By this, a good match between the statistics of the PUF readout and that of the transmit symbols in the coded modulation scheme is achieved.

However, a similar match can be attained if the regions are defined such that they are used with equal probabilities, cf., e.g., [4], [20]. Fig. 6 compares the regions with uniform width (except the outer ones) but non-uniform probabilities used up to now (top, shaping) with regions with non-uniform width but uniform probabilities (bottom). The corresponding ASK constellation is non-equidistantly spaced which can be achieved by non-linear distortion, so-called warping, of a regular ASK constellation [31]. For details see the appendix.

The regions of an \( M \)-ary scheme, i.e., the limits \( l_i \), are chosen such that

\[
\int_{\mathcal{R}_c} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \, dx = \frac{1}{M} \quad \forall c.
\]

Hence, no degree of freedom for optimization is present.

The block diagram of the transmitter is depicted in Fig. 7 for an 8-ary scheme. \( \mathcal{M}_w \) denotes the mapping of the binary triples to the warped 8-ary ASK constellation. The binary codes on the coding levels are encoded individually, resulting in signal points that are used with uniform probabilities. Decoding is done with multistage decoding (cf. bottom of Fig. 4) with three component decoders.

FIGURE 6. Regions of the pdf of the PUF readout of equal width but non-uniform probabilities and corresponding non-uniform ASK constellation (top, situation when employing shaping) and regions with non-uniform width but uniform probabilities (bottom). The corresponding ASK constellation is non-equidistantly spaced which can be achieved by non-linear distortion, so-called warping, of a regular ASK constellation [31]. For details see the appendix.

FIGURE 7. Block diagram of the multilevel encoder of the coded modulation scheme with warped constellation. Depicted for an 8-ary ASK constellation.
D. Channel Capacity and Decoding Metric

Since the statistic of the PUF readout (Gaussian) and that of the transmit signal in M-ary ASK signaling (discrete) are fundamentally different, the capacities when using these transmit signals over an AWGN channel will differ. In view of MLC/MSD which is applied here, we are interested in the level capacities

\[ C^{(l)} = I(y; c_l | c_{l-1}, \ldots, c_0), \]

where the label of the signal point/region is \([c_{l-1}, \ldots, c_1, c_0]\), with \(\mu = \log_2(M)\), and \(c_0\) being the least significant bit (LSB).

In Appendix A it is shown that the level capacities are given by

\[ C^{(l)} = \sum_{\forall c_{l-1} \ldots c_0} p_{c_{l-1} \ldots c_0} C(R_{c_{l-1} \ldots c_0}) \]

\[ - \sum_{\forall c_{l} \ldots c_0} p_{c_{l} \ldots c_0} C(R_{c_{l} \ldots c_0}), \]

where \(C(R_{c_{l} \ldots c_0})\) denotes the capacity of the AWGN channel when the channel input is Gaussian with variance 1 but restricted to the region \(R_{c_{l} \ldots c_0}\), which is the union of all regions whose least significant bits have the indicated values. \(p_{c_{l} \ldots c_0}\) is the respective probability, that this region is used (cf. Appendix A). Averaging is done as indicated over all binary \(l\)- and \(l + 1\)-tuples, respectively. This result holds for both the shaping scheme and uniform signaling; the regions and probabilities have to be selected adequately.

Note that since the code length \(n\) coincides with the number of used PUF nodes (RO pairs), we express the rate \(R = \frac{l}{n}\) and the capacity in terms of bit/node. For the case of ROs, the capacity might also be given as bit/RO pair, or, when calling the basic unit as cell, as bit/cell.

The derivation of the optimum decoding metric (LLRS) to be used in the component decoders can be found in Appendix A. We omit a repetition here.

E. Numerical Examples and Rate Design

We now give numerical examples for the capacities and derive the optimum rates of the component codes on the coding levels in the MLC scheme. Subsequently we study the above given 8-ary ASK scheme with shaping and, as a generalization, a 16-ary ASK scheme with shaping (here three coding levels are present). Moreover, a binary (BPSK), a 4-ary and a 8-ary ASK scheme with uniform signaling are considered.

1) Optimization of the Scaling

The choice of the scaling factor \(s\), i.e., the definition of the boundaries of the regions \(R_c\) in the shaping schemes, has a significant impact on the capacity. Hence, for best performance this parameter has to be optimized. For binary signaling, such a degree of freedom obviously does not exist.

Given the scaling factor \(s\), for 4, 8, and 16-ary signaling the signal-to-noise ratio (in dB) \(10 \log_{10}(1/\sigma_s^2)\) which is required to achieve a desired capacity is calculated (in the best case the rate \(R\) of the coding scheme can be equal to the capacity).

In view of the shaping scheme depicted in Fig. 5, where (for the 8 and 16-ary scheme) the highest level is uncoded, uses hard decisions, and carries 1/2 bit/symbol, the sum capacity \(C = C^{(0)} + C^{(1)} + 1/2\) is studied in the 8-ary case; similarly \(C = C^{(0)} + C^{(1)} + C^{(2)} + 1/2\) in the 16-ary case.

Shaping may also be implemented for a 4-ary scheme, however, shaping gains are more pronounced for larger constellations [31]. Hence, in practice, 4-ary ASK with uniform probabilities is of interest. The scaling parameter \(s\) may be chosen such that all regions are used with the same probability 1/4, which, considering (9), is achieved for \(s = 0.337\). In the 4-ary case we consider \(C = C^{(0)} + C^{(1)}\).

Fig. 8 shows the required SNRs for target rates \(R = C = 1.50, 1.25,\) and \(1.00\) bit/node (top to bottom). The respective minima are marked by circles.

![FIGURE 8. Required signal-to-noise ratio (in dB) 10 log_{10}(1/\sigma_s^2) to support a desired capacity in case of Gaussian readout. 4, 8, and 16-ary scheme; top to bottom: R = 1.50, 1.25, and 1.00. Dotted vertical line: s = 0.337; dotted curves: above the line the highest level with hard decision has a capacity larger than 0.5.](image)
a very low rate; the gains of 16-ary signaling compared to 8-ary signaling are not tremendous.

2) SUM CAPACITIES—COMPARISON W.r.t. \( M \)
For the fixed scaling factors \( s_4 = 0.337, s_8 = 0.35, \) and \( s_{16} = 0.175 \), Fig. 9 shows the sum capacities over the signal-to-noise ratio. For large SNRs, the binary (2-ary) scheme reaches 1 bit/node and the 4-ary scheme with uniform probabilities 2 bit/node. Due to the non-uniform probabilities, the 8 and 16-ary schemes do not approach 3 and 4 bit/node, respectively. For capacities up to approximately 1.5 bit/node the 16-ary scheme does not provide significant gain over the 8-ary scheme. For a desired sum capacity of 1.5 bit/node (see the circles in the zoom shown in the right bottom part) the 16-ary scheme approximately gains 1.5 dB. However, compared to the 4-ary scheme the required SNR for this requested capacity is over 4 dB lower for the 8-ary scheme.

![FIGURE 9. Sum capacities over the signal-to-noise ratio (in dB) in case of Gaussian readout. 2-ary, 4-ary (\( s_4 = 0.337 \)), 8-ary (\( s_8 = 0.35 \)), and 16-ary (\( s_{16} = 0.175 \)) scheme; Solid: soft-decision; Dashed: highest level hard decision. Gray: ASK signaling. Bottom: zoom into dashed regions.](image-url)

For comparison, the curves when using hard decisions in the highest level (in the single level for the binary scheme) are shown (dashed curves). In case of the binary scheme, hard decision causes a significant loss. In contrast, hard decision in the highest level of the multilevel scheme does not cause a loss in the regions of interest (cf. [33]).

The capacity curves for the schemes with Gaussian readout may be compared with that for conventional ASK signaling (gray curves; the probability distribution of the ASK schemes is chosen to be a discrete Gaussian (Maxwell–Boltzmann) distribution which has the same entropy (capacity for SNR going to infinity) as the probability distribution of the regions in case of Gaussian readout). A significant difference can be observed. For the same SNR, the Gaussian readout provides (much) less capacity when comparing with conventional digital transmission. As a consequence, the curves do not approach the Shannon limit \( C = \frac{1}{2} \log_2 (1 + \frac{1}{\sigma^2}) \) half a bit below their maximum capacity but only at much lower values. The capacity in case of Gaussian readout more resembles that of coded modulation over a Rayleigh fading channel. There, symbol expansion diversity [53] can be gained, i.e., it is rewarding to more than double the size of the used signal constellation compared to the smallest possible one.

Hence, even for a desired capacity of \( C = 0.5 \) bit/node (see the circles in the zoom shown in the left bottom part of Fig. 9), higher-order modulation is beneficial. Soft-decision decoding of binary codes already provides a gain of approximately 4 dB over conventional hard-decision decoding. Going to a 4-ary scheme, further almost 4 dB can be gained. For such low rates, the practical shaped schemes (with hard decision and \( R_2 = 0.5 \) in the highest level) do not provide further gains (see the dahed lines).

3) SUM CAPACITIES—SHAPING VS. UNIFORM
We now compare the capacities for an 8-ary scheme with shaping and uniform signaling, respectively. Fig. 10 shows the sum capacities over the signal-to-noise ratio. The solid lines correspond to the situation of shaping (uniform regions, non-uniform probabilities), the dashed lines to uniform signaling (non-uniform regions, uniform probabilities). For shaping, two situations are shown: the curve from Fig. 9 is repeated, where the scaling is fixed and the rate in the highest level (due to the shaping scheme) is fixed to 1/2 (blue) and the case when for all signal-to-noise ratios the scaling is optimized individually and the actual capacity of the highest level is taken into account (light blue).

![FIGURE 10. Sum capacities over the signal-to-noise ratio (in dB) in case of Gaussian readout. Solid: 8-ary shaping scheme with per-rate optimized scaling and total capacity with fixed scaling (\( s_8 = 0.35 \)) and highest level limited to \( R_2 = 1/2 \). Dashed: Uniform probabilities of the regions. Gray: ASK signaling.](image-url)

The optimized shaping scheme has a slightly larger capacity than the uniform scheme; when fixing the scaling and restricting the MSB rate, above approximately \( C = 1.3 \) bit/node the capacity of the shaping scheme is inferior. However, up to \( C = 1.5 \) bit/node no significant differences in the capacities are present. Hence, from this point of view, both approaches can be judged to have a similar performance.

For comparison, the curves for conventional ASK signaling over the AWGN channel are given (gray). Here, the
shaping scheme has a slightly more pronounced advantage over the scheme with warped constellation over a wide range of SNRs. Since the signal points are used with uniform probabilities, the warping scheme approaches the maximum rate \( \log_2(M) \).

![FIGURE 11. Level capacities over the signal-to-noise ratio (in dB) in case of Gaussian readout. Left: 4-ary uniform scheme; Middle: 8-ary shaping scheme; Right: 8-ary uniform scheme. Rate design for \( R = 1.5 \) bit/node.](image)

4) LEVEL CAPACITIES AND RATE DESIGN
Fig. 11 shows the capacities of the coding levels together with the sum capacity for the 4-ary uniform scheme (\( s_4 = 0.337 \), left), the 8-ary shaping scheme (\( s_8 = 0.35 \), middle), and the 8-ary uniform scheme (right). A rate design for a target rate of \( R = 1.5 \) bit/node is shown. The 4-ary uniform scheme uses \( R^{(0)} = 0.519 \) and \( R^{(1)} = 0.981 \); the 8-ary shaping scheme uses \( R^{(0)} = 0.145 \), \( R^{(1)} = 0.855 \), and \( R^{(2)} = 0.5 \) (since the shaping level is limited to half a bit per symbol); the 8-ary uniform scheme uses \( R^{(0)} = 0.121 \), \( R^{(1)} = 0.436 \), and \( R^{(2)} = 0.947 \). In comparison to the shaping scheme, a slightly lower rate at the lowest level is present but the highest level carries a significantly higher rate. The difference of about 4 dB in the required SNR between the 4 and 8-ary schemes is visible.

IV. NUMERICAL SIMULATIONS
In this section, we study the coded modulation/shaping schemes designed above by means of numerical simulations. We are mainly interested in the word error ratio (WER), i.e., the probability that the decoded message \( \hat{m} \) in the reproduction phase differs from the message \( m \) selected at the initialization phase. A WER below \( 10^{-6} \) is desired to obtain an error probability smaller than the device failure probability \([54]\).

A. SETTING AND PARAMETERS
As binary component code we employ Polar codes \([55]\); they allow a very flexible choice of the code rate and a low-complexity soft-input decoding algorithm is available, which is efficiently implementable in hardware \([56]\). In each case, the codeword length \( n = 1024 \) is used. The codes are designed based on the Bhattacharyya parameter \([55], [57]\); the design SNR is that which is required for transmitting the desired rate over an AWGN channel plus a penalty of 3 dB taking the finite-length performance into account.

Successive cancellation decoding is employed; list decoding \([58]\) is not used. Unless otherwise stated, soft-decision decoding is used; the LLRs are calculated as given in Appendix A and are clipped to a maximum magnitude of 100.

Trellis shaping (in the form of sign-bit shaping) is used for the 8-ary scheme. Here, the \( 2 \times 2 \) scrambler \(^9\) from \([31, \text{Tab. 4.13}]\) for a 4-state shaping scheme is employed. Note that the coded modulation scheme given in Fig. 4 uses (the Cartesian product of) two ASK constellations with natural labeling but shaping is preferably based on a QAM constellation where (at least) the two most significant bits are labeled according to the principles of set partitioning. The matrix on the right-hand side performs the transformation from 2D set-partitioning labeling to \( 2 \times 1D \) natural labeling (cf. \([59]\)).

In each case, we synthetically generate PUF readouts. \(^{10}\) To that end, the elements of the nominal (error-free) readout \( r_{\text{ref}} \) are drawn obeying a zero-mean unit-variance Gaussian distribution and the message word \( m \) is drawn uniformly from the set of length-\( k \) binary words. Encoding by the multilevel structure is performed and the shaping algorithm (Viterbi algorithm selecting the shaping redundancy \( \mathbf{u} \)) is run (for details see \([31]\)). Given the codeword \( a \) in signal space, the helper data is generated and directly applied to produce the processed reference readout \( x \). For each such PUF instance, at least 100 error vectors \( e \) are generated, modeling this number of readouts. As detailed in Sec. II, the components of the error word \( e \) are drawn according to a zero-mean Gaussian distribution with variance \( \sigma^2 \). For non-binary schemes, MSD is performed to recover the message. The word error ratio is averaged over \( 10^6 \) PUF instances.

The designs collected in Tab. 1 are considered.

B. BINARY VS. CODED MODULATION
We first consider the designs for \( R = 0.5 \) bit/node. In Fig. 12, the word error ratio is plotted over the signal-to-noise ratio for different settings.

Starting point is a single binary code with hard- and soft-decision decoding, respectively. As it is well-known, employing soft decisions, a significant gain can be achieved. The finite-length performance of the codes even show a larger gain as anticipated by the capacity results. The optimum decoding metric \((42)\) (solid line) performs slightly better than the conventional for BPSK over the AWGN channel, cf. \((43)\) (dashed line).

Even for this low code rate, going to 4-ary signaling is rewarding. A further gain of \( 4 \ldots 5 \) dB can be realized, which is consistent with the capacity arguments (see Fig. 9).

\(^9\)In contrast to \([31]\), here the MSB is the right-most bit in the bit label, hence the row and column flip of the scrambler matrices compared to the referred table.

\(^{10}\)Note that the numerical results given here cannot be directly compared with that shown in \([5]\) as there a uniform distribution of the PUF readout that matches the used modulo reduction has been considered.
TABLE 1. Designs (coding rates $R_i$ of the component codes and code dimensions $k_i$ for the considered codelength $n = 1024$) used in the numerical simulations.

|          | $R = 0.50$ | $R = 1.00$ | $R = 1.25$ | $R = 1.50$ |
|----------|------------|------------|------------|------------|
| 2-ary scheme | $R_0 = 0.50$, $k_0 = 512$ | — | — | — |
| 4-ary scheme uniform | $R_0 = 0.107$, $k_0 = 110$ | $R_0 = 0.230$, $k_0 = 235$ | $R_0 = 0.343$, $k_0 = 351$ | $R_0 = 0.519$, $k_0 = 532$ |
| 8-ary scheme uniform | — | $R_0 = 0.073$, $k_0 = 73$ | $R_0 = 0.095$, $k_0 = 96$ | $R_0 = 0.121$, $k_0 = 123$ |
| 8-ary scheme shaping | — | $R_0 = 0.016$, $k_0 = 16$ | $R_0 = 0.056$, $k_0 = 57$ | $R_0 = 0.145$, $k_0 = 148$ |

For reference, the performance of the binary rate-0.5 code with BPSK over the AWGN channel is shown. Using two distinct signal levels is clearly superior over the case of a Gaussian pdf for the useful signal, which is the case in PUFs. For comparison, the capacity limits of the respective approaches are indicated with vertical markers. The order of these limits is clearly reflected in the order of the performance of the practical schemes. However, due to the moderate block length, significant gaps to capacity are present.

C. CODED MODULATION AND SIGNAL SHAPING

We now turn to the comparison of the 4 and 8-ary schemes with uniform signaling and the 8-ary scheme with shaping. The results (word error ratio over the signal-to-noise ratio) are displayed in Fig. 13. The designs of Tab. 1 are used.

The capacity curves for the 4-ary and 8-ary schemes differ most for higher capacities and merge as the capacity decreases. Hence, it can be expected that the difference between the performance of the schemes is largest for a rate of $R = 1.50$ bit/node (solid lines). Here, the 8-ary scheme with shaping (blue) gains almost 6 dB over the 4-ary uniform scheme (red). The 8-ary uniform scheme (magenta) performs about 1 dB worse than the shaping scheme. Note that almost no gain can be achieved if the 4-state shaping scrambler is replaced by scrambler with a higher number of states (cf. [31, Tab. 4.13]; not shown).

For a rate of $R = 1.25$ bit/node (dashed) still a gain of almost 4 dB is achieved if the 8-ary shaping scheme is used; the 8-ary uniform scheme gains only little over the 4-ary uniform scheme.

For the low rate of $R = 1.00$ bit/node (dashed-dotted) the 8-ary scheme with shaping performs almost equal to the scheme with $R = 1.25$ bit/node. This is justified by the black dotted curve which gives the WER of Level 2 where WER(2) $\approx 1 - (1 - BER)^n$ and the bit error ratio BER is approximated as given in (29). Thus, the highest level limits performance. However, still 1 dB can be gained over the 4-ary uniform scheme. For this low rate, the 8-ary uniform scheme does not perform well; even worse than the 4-ary uniform scheme. This is due the lowest coding level. For the equal-probable partition, the inner regions are very narrow and hence higher error rates occur when using these regions. The code at level 0 has a very low rate but it cannot compensate for the loss caused by the inner regions. Note that a binary scheme cannot achieve the studied rates.

Again, for comparison, the capacity limits of the respective approaches (for $R = 1.50$ bit/node only) are indicated with vertical markers.

In summary, it can be stated that the 8-ary scheme with shaping has a very good performance for all rates of interest.
For a rate up to $R = 1.50$ bit/node a WER smaller than $10^{-6}$ can be guaranteed if the SNR is larger than 20 dB (cf. Sec. II-A).

### D. EVALUATION WITH ROPUF MEASUREMENT DATA

At the Institute of Microelectronics at Ulm University, 22 instances of ROPUFs have been implemented on FPGAs, each comprising 3800 ROs. Out of these ROs, $n = 1024$ disjoint pairs have been selected randomly and have been measured at various temperatures; for details see [36].

For initialization, for each PUF instance 10 readouts measured at a temperature of 20 °C are averaged and define the reference readout $r_{\text{ref}}$ for this instance. The message $m$ is randomly selected and the helper data (signed permutation matrix $P$) is generated according to the new scheme proposed in Sec. II-B.

In the above description of the scheme with shaping, the highest level was assumed to be uncoded. In practice, some high-rate code with hard-decision decoding may be in place. We assume that such a code is capable to correct up to 3 bit errors per word of length $n/2$.

For verification, 10,000 readouts per PUF instance and per temperature from $-10$ °C to 50 °C in steps of 10 °C are used (70,000 readouts per PUF instance in total). The generated helper data is applied and decoding is conducted.

In Tab. 2, the number of erroneous PUF instance and the number of word errors per erroneous pin are tabulated. The rate is chosen as $R = 1.00$, $R = 1.25$, and $R = 1.50$ bit/node. Results for the scheme employing 4-ary and 8-ary uniform signaling and 8-ary signaling with shaping are given (employing the parameters of Tab. 1).

| $R$          | $R = 1.00$ | $R = 1.25$ | $R = 1.50$ |
|--------------|------------|------------|------------|
| 4-ary uniform| 1 instance | 9 instances| 12 instances|
|              | 1 error    | 1 to 3 errors| 2 to 3102 errors|
| 8-ary uniform| 1 instance | 1 instance | 4 instances |
|              | 1 error    | 1 error    | 1 to 2 errors |
| 8-ary shaping | 0 instances| 0 instances| 2 instances |
|              | —          | —          | 1 error     |

As can be seen, the 4-ary uniform schemes fails even for the small rates; significant failures occur for $R = 1.50$ bit/node for the 4-ary uniform scheme. The 8-ary uniform scheme performs better. In contrast, the proposed 8-ary coded modulation/shaping scheme is able to error-free recover the message over the entire temperature ranges for rates up to $R = 1.25$ bit/node. Only for the highest rate $R = 1.50$ bit/node among the $22 \times 70,000$ decoding runs, 2 failed. Hence, in summary, the proposed scheme is able to operate reliably over a wide range of temperatures and with high rates.

### V. CONCLUSION

In this paper we have considered multi-valued PUFs, where we potentially extract more than one bit of entropy from each PUF node (or PUF cell, PUF unit). This is achieved by directly accessing the analog readout and resorting to methods from coded modulation. In addition, in order to match the (approximately) Gaussian distribution of the readout, signal shaping has been employed. The coded modulation/shaping scheme has been designed based on capacity arguments. To that end, the respective capacities have been derived. It has been shown via numerical simulations that rates up to 1.50 bit/node are possible using an 8-ary scheme; these results have been complemented by applying the scheme to measurement data which has been provided by the Institute of Microelectronics at Ulm University. In addition, the capacity arguments cover that even for low rates (e.g., 0.50 bit/node) a 4-ary scheme is rewarding.

In the derivations, we have approximated the readout to be Gaussian distributed. As the frequencies of the ROs are measured by counting the oscillations per a given measurement period, the frequencies and, thus, the frequency differences are quantized. For a measurement period of 10 $\mu$s [36], the resolution of the frequency difference is $100$ kHz. As the variance of the unnormalized readout is in the range of $12$ MHz$^2$, after normalization to unit variance, the resolution is $100$ kHz/$\sqrt{12}$ MHz$^2 \approx 0.029$. Since the width of the regions, in which the (normalized) PUF readout is categorized, is in the range of $2\sigma \approx 0.7$, each region comprises $\approx 25$ possible discrete frequency differences. Hence, an approximation by a continuous pdf is reasonable. However, the (densely) discrete nature of the PUF readout can easily be taken into account in the calculations of the capacities. Replacing the pdf (14) suitably, any (known) distribution can be taken into account in the analysis and the design of the coded modulation/shaping scheme. Thus, the approach is not only applicable to Gaussian signals and noise but to a broad field.

### APPENDIX A

CAPACITIES AND DECODING METRICS

In this appendix, we derive the capacities of the coding levels of the multilevel coding scheme and the appropriate decoding metrics. The cases of conventional ASK signaling and that of a Gaussian readout in PUFs are compared.

#### A. SHAPED SIGNAL CONSTELLATION AND CORRESPONDING REGIONS

We consider the situation visualized in Fig. 14, which for the example of an 8-ary scheme is repeated from Fig. 3.

1) ASK SIGNALING

In case of conventional $M$-ary signaling, a signal constellation $sA$, with $A = \{\pm 1, \pm 3, \ldots, \pm (M - 1)\}$ being an $M$-ary ASK signal constellation and $s$ a scaled scaling factor, is used. The mapping $M$ maps $\mu = \log_2(M)$-bit tuples to signal points in $sA$, i.e., to each point corresponds a label
The signal points are used with probabilities $p_c = p_{c_{1}...c_{10}}$. The pdf of the transmit symbols $a$ when a specific bit label is present is given by (δ(·): Dirac delta)

$$f_a(a | c) = \delta(a - M(c)). (13)$$

When transmitting over an AWGN channel, i.e., when zero-mean Gaussian noise with variance $\sigma_e^2$ (pdf $f_e(e)$) is superimposed, the corresponding conditional pdf of the receive signal is given by

$$f_y(y | c) = f_a(y | c) * f_e(y) = \frac{1}{\sqrt{2\pi}\sigma_e} e^{-\frac{(y - M(c))^2}{2\sigma_e^2}}. (14)$$

2) PUF READOUT
In case of PUF readouts, each bit label is represented by a region $R_c = R_{c_{1}...c_{10}}$; the actual number to be communicated is drawn according to a continuous zero-mean unit-variance Gaussian pdf restricted to this region, i.e.,

$$f_x(x | c) = \begin{cases} \frac{1}{\sigma_e} e^{-\frac{x^2}{2\sigma_e^2}}, & x \in R_c \\ 0, & \text{else.} \end{cases} (15)$$

Note that the region is used with probability

$$p_c = \int_{R_c} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma_e^2}} \, dx = \int_{h_{0}}^{h_{1}} \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma_e^2}} \, dx, (16)$$

where $h_0$ and $h_1$ denote the lower and upper limit of the region $R_c$. When transmitting over an AWGN channel, the corresponding conditional pdf of the receive signal can be calculated after some manipulations to be

$$f_y(y | c) = f_x(y | c) * f_e(y) = \frac{1}{\sqrt{2\pi}\sigma_y} e^{-\frac{(y - M(c))^2}{2\sigma_y^2}} (Q(Dl_{1,c} - Fy) - Q(Dl_{0,c} - Fy))$$

$$= \frac{1}{\sqrt{2\pi}\sigma_y} e^{-\frac{y^2}{2\sigma_y^2}} \Delta Q(y, R_c), (17)$$

with the obvious definition of $\Delta Q(\cdot, \cdot)$ and where

$$\sigma_y^2 \overset{\text{def}}{=} 1 + \sigma_e^2, \quad D \overset{\text{def}}{=} \frac{\sigma_y}{\sigma_e}, \quad F \overset{\text{def}}{=} \frac{1}{\sigma_y} \frac{1}{\sigma_e}. (18)$$

and $Q(x)$ is the complementary Gaussian integral function

$$Q(x) \overset{\text{def}}{=} \int_x^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} \, dz. (19)$$

Note that due to the one-to-one correspondence of signal point and region, ASK signaling can also be characterized by the regions—both scenarios can be handled uniformly. The respective pdf over the region is either a weighted Dirac pulse (13) or a continuous Gaussian one (15).

3) UNION OF REGIONS AND PROBABILITIES
It is convenient to define (“∀ $c_{1}...c_{10}$”) indicates that the union is done over all binary tuples of the indicated variables)

$$R_{b} \overset{\text{def}}{=} \bigcup_{\forall c_{1}...c_{10}} R_{c_{1}...c_{10}b}, (20)$$

i.e., the union of regions which represent the least significant bit (LSB) $c_0 = b$. In the same way the regions $R_{b_{1}...b_{0}}$ are defined. In addition, the probabilities $p_{b}$ (and $p_{b_{1}b_{0}}$, etc.) are given for the Gaussian readout by

$$p_{b} = \sum_{\forall c_{1}...c_{10}} p_{c_{1}...c_{10}b}. (22)$$

Finally, we use the denomination “∀ $c_{1} = c_0$” to indicate that the transmit signal is drawn from the region $R_{c_0}$, i.e., the set which represents the LSB with the given value $c_0$. Specifically,

$$f_y(y | c_0) = \sum_{\forall c_{1}...c_{10}} f_x(y | [c_{1}...c_{10}]) * f_e(y) = \sum_{\forall c_{1}...c_{10}} f_y(y | [c_{1}...c_{10}]). (23)$$

In the same ways the condition to $c_1c_0$ has to be understood.

B. UNIFORM SIGNALING AND CORRESPONDING REGIONS
Shaping/signaling with non-uniform probabilities is used to approximate the continuous Gaussian transmit pdf which is capacity-achieving on the AWGN channel. However, even with equiprobable signaling, the capacity can be approached. To that end, the uniform spacing of the signal points has to be replaced by a non-uniform one. In [60] it is shown that the optimum procedure is to choose $M$ regions, such that the integral over a Gaussian pdf within the regions is always $1/M$. The signal points are then given as the centroids of the regions.
Such non-uniformly spaced constellations can be generated from a regular ASK constellation by non-linear distortion, called warping in the literature [31].

The situation is visualized in Fig. 15 for an 8-ary scheme. The lower \(l_{\min}\) and upper \(l_{\max}\) limits of the regions \(R_{c}\) can be read off from the figure. E.g., \(l_{\min} = -\infty\) and \(l_{\max} = -l_3\). These limits are chosen such that \(p_c\) in (16) equals \(1/M\) for all \(c\).

The above derived conditional pdf and all subsequent derivations are valid for the uniform regions and non-uniform probabilities used in shaping and the non-uniform regions and uniform probabilities in the current situation.

**C. LEVEL CAPACITIES**

In multilevel coding, the capacities of the coding levels are of interest, which, in view of the multistage decoding principle, are given by [33]

\[
C^{(l)} = I(y; c_l | c_{l-1} \ldots c_0),
\]

i.e., the mutual information between the channel output and the input \(c_l\) to the mapper, when the lower levels are already known.

1) LEVEL 0

Following the procedure in [33], the capacity of coding Level 0 can be calculated for the present situation as

\[
C^{(0)} = \int \sum_{y \in \mathcal{Y}} f_y(y | c_0) \log \left( \frac{1}{p_{c_0}} \frac{f_y(y | c_0)}{f_y(y)} \right) dy
\]

\[
= \int \sum_{y \in \mathcal{Y}} f_y(y | c_0) \log \left( \frac{1}{p_{c_0}} \frac{f_y(y | c_0)}{f_y(y)} \right) dy
\]

\[
- \int \sum_{y \in \mathcal{Y}} f_y(y | c_0) \log (f_y(y)) dy
\]

\[
= \sum_{y \in \mathcal{Y}} p_{c_0} \int \frac{1}{p_{c_0}} f_y(y | c_0) \log \left( \frac{1}{p_{c_0}} \frac{f_y(y | c_0)}{f_y(y)} \right) dy
\]

\[
- \int f_y(y) \log (f_y(y)) dy
\]

\[
= h(y) - \sum_{\forall c_0} p_{c_0} h(y | c_0),
\]

where \(h(\cdot)\) denotes differential entropy. This expression can be rewritten as

\[
C^{(0)} = h(y) - h(e) - \sum_{\forall c_0} p_{c_0} h(y | c_0) + h(e)
\]

\[
= C(R) - \sum_{\forall c_0} p_{c_0} C(R_{c_0}),
\]

where \(C(R_{c_0})\) denotes the capacity of the AWGN channel when the channel input is Gaussian with variance 1 but restricted to the region \(R_{c_0}\). \(R = \mathbb{R}\) is the union of all regions, i.e., the entire real axis.

2) LEVEL 1

In the same way we obtain the capacity of coding Level 1 (\(p_{c_1 | c_0}\) is the conditional probability of \(c_1\) given \(c_0\))

\[
C^{(1)} = \sum_{\forall c_0} p_{c_0} \left( C(R_{c_0}) - \sum_{\forall c_1} p_{c_1 | c_0} C(R_{c_1 | c_0}) \right)
\]

\[
= \sum_{\forall c_0} p_{c_0} C(R_{c_0}) - \sum_{\forall c_1} p_{c_1 | c_0} C(R_{c_1 | c_0}).
\]

3) LEVEL 2

Finally, the capacity of coding Level 2 equals

\[
C^{(2)} = \sum_{\forall c_1} p_{c_1 | c_0} C(R_{c_1 | c_0})
\]

\[
- \sum_{\forall c_2 | c_1} p_{c_2 | c_1} C(R_{c_2 | c_1}).
\]

Please note that in contrast to ASK signaling where the capacity of a constellation using a single signal point is zero, in the case of Gaussian readout \(C(R_{c_2 | c_1}) > 0\).

Usually, for the highest level, hard decisions are utilized. Here, a threshold device decides between i) \(R_{000}\) and \(R_{100}\), or between ii) \(R_{001}\) and \(R_{101}\), or between iii) \(R_{010}\) and \(R_{110}\), or between iv) \(R_{011}\) or \(R_{111}\). As for the Gaussian readout the minimum distance of the useful signal to the threshold is 3\(s\), the bit error ratio is upper bounded by

\[
\text{BER} \leq Q\left(\frac{3s}{\sigma_e}\right).
\]

The end-to-end model including mapping and receiver-side quantization is given by the BSC; however, the input does not have uniform probabilities. Assuming that the regions \(R_{c_1 | c_0}\) and \(R_{c_2 | c_1}\) are used with probabilities \(p_{c_1 | c_0}\) and \(p_{c_2 | c_1}\), respectively, the quantized channel output takes on the value “0” (approximately) with probability

\[
(1 - \text{BER}) \cdot p_{c_1 | c_0} + \text{BER} \cdot p_{c_1 | c_0}.
\]

Note that \(C = H(Y) - H(Y | X)\) and \(H(Y | X) = H_2(\text{BER})\), where

\[
H_2(x) \triangleq -x \log_2(x) - (1 - x) \log_2(1 - x)
\]
is the binary entropy function. Hence, the capacity of Level 2 when using hard decision is given by
\[
C_{\text{HD}}^{(2)} = \sum_{c_1 \in \{0,1\}} p_{c_1|0} \left( H_2((1 - \text{BER}) \cdot p_{c_0|c_1=0} + \text{BER} \cdot p_{c_1|c_1=0}) \right)
\]
\[
- H_2(\text{BER}) \right). \quad (32)
\]

**D. DECODING METRICS**

At the receiver, suited decoding metrics have to be calculated ("inverse mappings" $M_t^{-1}$, see Fig. 4). As usual in practical schemes, log-likelihood ratios (LLR) are used.

1) LEVEL 0

Due to the symmetry of the probability distribution, the LSB $c_0$ takes on the two values "0" and "1" each with probability $p_0 = p_1 = 1/2$. The LLR for $c_0$ is given by
\[
\text{LLR}^{(0)} = \log \frac{\Pr[c_0 = 1 | y]}{\Pr[c_0 = 0 | y]} = \log \frac{f_y(y | c_0 = 0) p_0}{f_y(y | c_0 = 1) p_1} = \log \frac{f_y(y | c_0 = 0)}{f_y(y | c_0 = 1)}. \quad (33)
\]

For ASK signaling, considering (23) and (14), we arrive at
\[
\text{LLR}^{(0)} = \log \left( \frac{\sum_{c_1 \in \{0,1\}} p_{c_2|c_1 \in \{0,1\}} \exp \left( \frac{-y - z M(c_2|c_1=0)}{2\sigma^2} \right)}{\sum_{c_1 \in \{0,1\}} p_{c_2|c_1 \in \{0,1\}} \exp \left( \frac{-y - z M(c_2|c_1=1)}{2\sigma^2} \right)} \right), \quad (34)
\]
and for the Gaussian readout, considering (23) and (17), we have
\[
\text{LLR}^{(0)} = \log \left( \frac{\sum_{c_1 \in \{0,1\}} \Delta Q(y, R_{c_2|c_1 \in \{0,1\}})}{\sum_{c_1 \in \{0,1\}} \Delta Q(y, R_{c_2|c_1 \in \{0,1\}})} \right). \quad (35)
\]

2) LEVEL 1

When $c_0$ is known, obeying the multistage decoding principle, the binary symbol $c_1$ is decoded. When $c_0 = 0$, the region $R_0$ is used and the region $R_{10}$ represents the binary "0" and region $R_{11}$ the binary "1". Note that now the probability for $c_1 = 0$ and $c_1 = 1$ are $p_{00}$ and $p_{10}$, respectively, which may be different. Thus, the LLR for $c_1$ is given by (for brevity we state the final result only for the case of Gaussian readout; the ASK case is straightforward)
\[
\text{LLR}^{(1), c_0=0} = \log \frac{f_y(y | c_0 = 0, c_1 = 0) p_{00}}{f_y(y | c_0 = 0, c_1 = 1) p_{10}} = \log \frac{\sum_{c_2} \Delta Q(y, R_{c_2|c_1=0})}{\sum_{c_2} \Delta Q(y, R_{c_2|c_1=1})} + \log \left( \frac{p_{00}}{p_{10}} \right). \quad (36)
\]

In the same way we have
\[
\text{LLR}^{(1), c_0=1} = \log \frac{f_y(y | c_0 = 0, c_1 = 1) p_{01}}{f_y(y | c_0 = 1, c_1 = 1) p_{11}} = \log \frac{\sum_{c_2} \Delta Q(y, R_{c_2|c_1=0})}{\sum_{c_2} \Delta Q(y, R_{c_2|c_1=1})} + \log \left( \frac{p_{01}}{p_{11}} \right). \quad (37)
\]

3) LEVEL 2

When $c_0$ and $c_1$ are known, the binary symbol $c_2$ is decoded. Here, hard decision can be used without significant loss and a decision boundary half way between the regions is almost optimum.

To justify this, assume that two signal points, denoted as $a_0$ and $a_1$ (with $a_0 < a_1$), are present; they are used with probabilities $p_0$ and $p_1$, respectively. A decision is taken in favor for $a_0$ if
\[
\Pr[a_0 | y] \geq \Pr[a_1 | y], \quad (38)
\]

which is fulfilled for
\[
y \leq \frac{a_1 + a_0}{2} + \frac{\sigma^2}{a_1 - a_0} \cdot \log \left( \frac{p_0}{p_1} \right). \quad (39)
\]

In the present application, $p_{000}$ and $p_{100}$ exhibit the largest difference; for $\sigma^2 = 0.01$ (cf. Sec. II), the shift compared to a decision boundary half way between the signal points is given by
\[
\frac{\sigma^2}{a_1 - a_0} \cdot \log \left( \frac{p_0}{p_1} \right) \approx \frac{0.01}{8} \cdot \log \left( \frac{0.235}{0.013} \right) = 0.0036, \quad (40)
\]
which can be neglected in comparison to the distance 4 to the decision threshold.

**E. BINARY SCHEME**

For completeness, the results for the binary (2-ary) scheme with Gaussian readout are stated.

1) CAPACITY

Following the above derivations, the capacity of the binary scheme is given as
\[
C = C_{\text{Gauss}} - C_{\text{half Gauss}}, \quad (41)
\]
where $C_{\text{Gauss}} = \frac{1}{2} \log_2 (1 + 1/\sigma^2)$ denotes the capacity of the AWGN channel with Gaussian input and $C_{\text{half Gauss}}$ the respective capacity when the input is half-normal distributed.

2) DECODING METRIC

The decoding metric for soft-decision in the binary scheme is given by (cf. (17) and (18))
\[
\text{LLR} = \log \left( \frac{Q(-F y)}{Q(+F y)} \right). \quad (42)
\]

Note that for BPSK over the AWGN channel the decoding metric would be (the minus sign is present since here we assume that the binary "0" is represented by the signal point $-1$, cf. Fig. 3)
\[
\text{LLR} = \frac{2}{\sigma^2} y. \quad (43)
\]

**ACKNOWLEDMENT**

The authors thank Holger Mandry and Maurits Ortmanns, Institute of Microelectronics, Ulm University, for valuable discussions and for providing the measurement data of their implementation of ROPUFs.
REFERENCES

[1] A. Juel and M. Wattenberg, “A fuzzy commitment scheme,” in Proc. 6th ACM Conf. Comput. Commun. Secur. (CCS), 1999, pp. 28–36.

[2] J.-P. Linnartz and P. Tuyls, “New shielding functions to enhance privacy and prevent misuse of biometric templates,” in Proc. Int. Conf. Audio-Video-Biometric Person Authentication, 2003, pp. 393–402.

[3] Y. Dodis, L. Reyzin, and A. Smith, “Fuzzy extractors: How to generate strong keys from biometrics and other noisy data,” in Proc. Int. Conf. Theory Appl. Cryptograph. Techn., 2004, pp. 523–540.

[4] V. Immler and K. Uppud, “New insights to key derivation for tamper-resistant physical unclonable functions,” IACR Trans. Cryptograph. Hardw. Embedded Syst., vol. 3, pp. 30–65, May 2019.

[5] S. Muelich, H. Mandy, M. Ortmanns, and R. F. H. Fischer, “A multi-level coding scheme for multi-valued physical unclonable functions,” IEEE Trans. Inf. Forensics Security, vol. 16, pp. 3814–3827, 2021.

[6] J. Park and J.-Y. Sim, “A physically unclonable function combining a process mismatch amplifier in an oscillator collapse topology,” in IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers, Feb. 2021, pp. 504–506.

[7] Y. Zhang, Z. He, M. Wan, J. Liu, H. Gu, and X. Zou, “A SC PUF standard cell used for key generation and anti-invasive-attack protection,” IEEE Trans. Inf. Forensics Security, vol. 16, pp. 3958–3973, 2021.

[8] R. Maes, A. Van Herreweghe, and I. Verbauwhede, “PUFKY: A fully functional PUF-based cryptographic key generator,” in Proc. Int. Workshop Cryptograph. Hardw. Embedded Syst., 2012, pp. 302–319.

[9] S. Muelich, S. Puchinger, M. Bossert, M. Hiller, and G. Sigl, “Error correction for physical unclonable functions using generalized concatenated codes,” in Proc. Int. Workshop Algeb. Combinat. Coding Theory, 2014, pp. 1–6.

[10] S. Puchinger, S. Muelich, M. Bossert, M. Hiller, and G. Sigl, “On error correction for physical unclonable functions,” in Proc. 10th Int. ITG Conf. Syst. Commun. Coding (SCC), 2015, pp. 1–6.

[11] S. Muelich, S. Puchinger, V. Stukalov, and M. Bossert, “A channel model and soft-decision helper data algorithms for ROPUFs,” in Proc. 12th Int. ITG Conf. Syst. Commun. Coding (SCC), 2019, pp. 1–6.

[12] R. Maes, P. Tuyls, and I. Verbauwhede, “A soft decision helper data algorithm for SRAM PUFs,” in Proc. IEEE Int. Symp. Inf. Theory, Jun. 2009, pp. 2101–2105.

[13] V. Immler, J. Obermaier, M. Konig, M. Hiller, and G. Sigl, “B-TREPID: Batteryless tamper-resistant envelope with a PUF and integrity detection,” in Proc. IEEE Int. Symp. Hardw. Oriented Secur. Trust (HOST), Apr. 2018, pp. 49–56.

[14] V. Immler, J. Obermaier, K. K. Ng, F. X. Ke, J. Lee, Y. P. Lim, W. K. Oh, K. H. Wee, and G. Sigl, “Secure physical enclosures from covers with tamper-resistance,” IACR Trans. Cryptograph. Hardw. Embedded Syst., pp. 51–96, Nov. 2018.

[15] K.-H. Chuang, E. Bury, R. Degraeve, B. Kaczer, G. Groeseneken, I. Verbauwhede, and D. Linten, “Physically unclonable function using CMOS breakdown position,” in Proc. IEEE Int. Rel. Phys. Symp. (IRPS), Apr. 2017, pp. 4C-1.1–4C-1.7.

[16] K.-H. Chuang, E. Bury, R. Degraeve, B. Kaczer, T. Kallstenius, G. Groeseneken, D. Linten, and I. Verbauwhede, “A multi-bit/cell PUF using analog breakdown positions in CMOS,” in Proc. IEEE Int. Rel. Phys. Symp. (IRPS), Mar. 2018, pp. P.CR.2.1–P.CR.2.5.

[17] V. C. Immler, “Higher-order alphabet physical unclonable functions: Constructions, properties, and applications,” Ph.D. dissertation, Dept. Elect. Comput. Eng., TU München, Munich, Germany, 2019.

[18] V. Immler, M. Hiller, Q. Liu, A. Lenz, and A. Wachtler-Zeh, “Variable-length bit mapping and error-correcting codes for higher-order alphabet PUFs,” in Proc. Int. Conf. Secur. Privacy, Appl. Cryptogr. Eng., 2017, pp. 190–209.

[19] V. Immler, M. Hiller, Q. Liu, A. Lenz, and A. Wachtler-Zeh, “Variable-length bit mapping and error-correcting codes for higher-order alphabet PUFs—Extended version,” J. Hardw. Secur. Surv., vol. 3, no. 1, pp. 78–93, Mar. 2019.

[20] H. Mandy, A. Herkle, S. Muelich, J. Becker, R. F. H. Fischer, and M. Ortmanns, “Normalisation and multi-valued symbol extraction from RO-PUFs for enhanced uniform probability distributions,” IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 67, no. 12, pp. 3372–3376, Dec. 2020.

[21] M. Pehl, T. Tretschk, D. Becker, and V. Immler, “Spatial context tree weighting for physical unclonable functions,” in Proc. Eur. Conf. Circuit Theory Design (ECCDT), Sep. 2020, pp. 1–4.

[22] A. Herkle, H. Mandry, J. Becker, and M. Ortmanns, “In-depth analysis and length bit mapping and error-correcting codes for higher-order alphabet PUFs,” Elements of Information Theory, New York, NY, USA: Wiley, 1991.

[23] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, “Siemens physical random functions,” in Proc. 9th ACM Conf. Comput. Commun. Secur. (CCS), 2002, pp. 148–160.

[24] A. Herkle, H. Mandy, J. Becker, and M. Ortmanns, “In-depth analysis and enhancements of RO-PUFs with a partial reconfiguration framework on Xilinx Zynq-7000 SoC FPGA,” in Proc. IEEE Int. Symp. Hardw. Oriented Secur. Trust (HOST), May 2019, pp. 238–247.

[25] R. Maes, “An accurate probabilistic reliability model for silicon PUFs,” in Cryptographic Hardware and Embedded Systems—CHES 2013, G. Bertoni and J.-S. Coron, Eds. Berlin, Heidelberg: Springer, 2013, pp. 73–89.

[26] M. Hiller and J. A. Thomas, Elements of Information Theory. New York, NY, USA: Wiley, 1991.

[27] M. Hiller, D. Merli, F. Stumpf, and G. Sigl, “Complementary IBS: Application specific error correction for PUFs,” in Proc. IEEE Int. Symp. Hardw.-Oriented Secur. Trust, Jun. 2012, pp. 1–6.

[28] H. Mandy, M. Werner, L. R. Lima, M. Birkner, and G. Sigl, “Breaking through fixed PUF block limitations with differential sequence coding and convolutional codes,” in Proc. 3rd Int. Workshop Trustworthy Embedded Devices, 2013, pp. 49–54.

[29] M. Yu and S. Devadas, “Secure and robust error correction for physical unclonable functions,” IEEE Des. Test Comput., vol. 27, no. 1, pp. 48–65, Jan. 2010.

[30] M. Bloch and J. Barros, Physical-Layer Security: From Information Theory to Security Engineering. Cambridge, U.K.: Cambridge Univ. Press, 2011.

[31] G. Ungerboeck, “Channel coding with multilevel/phase signals,” IEEE Trans. Inf. Theory, vol. IT-28, no. 1, pp. 51–57, Jan. 1982.
[44] G. Caire, G. Taricco, and E. Biglieri, “Bit-interleaved coded modulation,” *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 927–946, May 1998.

[45] A. Khandani and P. Kabal, “Shaping multidimensional signal spaces—Part I: Optimum shaping, shell mapping, part II: Shell-addressed constellations,” *IEEE Trans. Inf. Theory*, vol. 39, no. 6, pp. 1799–1819, Nov. 1993.

[46] F. R. Kschischang and S. Pasupathy, “Optimal shaping properties of the truncated polydisc,” *IEEE Trans. Inf. Theory*, vol. 40, no. 3, pp. 892–903, May 1994.

[47] R. Laroia, N. Farvardin, and S. A. Tretter, “On optimal shaping of multidimensional constellations,” *IEEE Trans. Inf. Theory*, vol. 40, no. 4, pp. 1044–1056, Jul. 1994.

[48] J. G. Forney, R. G. Gallager, G. Lang, F. M. Longstaff, and S. U. Qureshi, “Efficient modulation for band-limited channels,” *IEEE J. Sel. Areas Commun.*, vol. SAC-2, no. 5, pp. 632–647, Sep. 1984.

[49] P. Schulte and G. Böcherer, “Constant composition distribution matching,” *IEEE Trans. Inf. Theory*, vol. 62, no. 1, pp. 430–434, Jan. 2016.

[50] G. D. Forney, Jr., and G. Ungerboeck, “Modulation and coding for linear Gaussian channels,” *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2384–2415, Oct. 1998.

[51] G. Böcherer, P. Schulte, and F. Steiner, “Bandwidth efficient and rate-matched low-density parity-check coded modulation,” *IEEE Trans. Commun.*, vol. 63, no. 12, pp. 4651–4665, Dec. 2015.

[52] R. Maes, Physically Unclonable Functions: Constructions, Properties and Applications. Berlin, Germany: Springer, 2013.

[53] U. Hansson and T. Aulin, “Channel symbol expansion diversity,” *Electron. Lett.*, vol. 31, no. 18, pp. 1545–1546, Aug. 1995.

[54] Xilinx. (2019). Xilinx Device Reliability Report First Half 2018. Accessed: Nov. 19, 2021. [Online]. Available: https://www.xilinx.com/support/documentation/user_guides/ug116.pdf

[55] E. Arikan, “Channel polarization: A method for constructing capacity-achieving codes for symmetric binary-input memoryless channels,” *IEEE Trans. Inf. Theory*, vol. 55, no. 7, pp. 3051–3073, Jul. 2019.

[56] A. A. Andi and O. Gazi, “Fast decoding of polar codes using tree structure,” *IET Commun.*, vol. 13, no. 14, pp. 2063–2068, Aug. 2019.

[57] H. Vangala, E. Viterbo, and Y. Hong, “A comparative study of polar code constructions for the AWGN channel,” 2015, arXiv:1501.02473.

[58] I. Tal and A. Vardy, “List decoding of polar codes,” *IEEE Trans. Inf. Theory*, vol. 61, no. 5, pp. 2213–2226, May 2015.

[59] A. Alvarado, A. G. I. Amat, F. Brannstrom, and E. Agrell, “On the equivalence of TCM encoders,” in *Proc. IEEE Int. Symp. Inf. Theory Proc.*, Jul. 2012, pp. 2401–2405.

[60] F.-W. Sun and H. C. A. van Tilborg, “Approaching capacity by equiprobable signaling on the Gaussian channel,” *IEEE Trans. Inf. Theory*, vol. 39, no. 5, pp. 1714–1716, Sep. 1993.

---

**SVEN MÜELICH**

received the B.Sc. degree in computer science from Aalen University, Germany, in 2011, and the M.Sc. degree in computer science from Ulm University, in 2013. He defended his Ph.D. thesis with the title *Channel Coding for Hardware-Intrinsic Security* in 2019, and subsequent to his Ph.D. graduation, he has been engaged as a Postdoctoral Researcher at the Institute of Communications Engineering, Ulm University. In 2013, he joined the Institute of Communications Engineering, as a Research Assistant, aiming to gain the Ph.D. degree. His research focus was on physical unclonable functions, while he also worked on rank metric codes and code-based cryptography. His main research interest includes coding for physical unclonable functions.

---

**ROBERT F. H. FISCHER** (Senior Member, IEEE) received the Dr.-Ing. and Habilitation degrees from the University of Erlangen–Nürnberg, Erlangen, Germany, in 1996 and 2001, respectively.

From 1992 to 1996, he was a Research Assistant with the Telecommunications Institute, University of Erlangen–Nürnberg. In 1997, he was with the IBM Research Laboratory, Zürich, Switzerland. In 1998, he returned to the University of Erlangen–Nürnberg, and in 2005, he spent a sabbatical with ETH Zürich, Zürich. Since 2011, he has been a Full Professor with the University of Ulm, Ulm, Germany. He is currently teaching the undergraduate and graduate courses on signals and systems and digital communications. He authored the textbook *Precoding and Signal Shaping for Digital Transmission* (Wiley, 2002). His current research interests include fast, reliable, and secure digital transmission, including single-carrier and multicarrier modulation techniques, information theory, coded modulation, digital communications, signal processing, and especially precoding and shaping techniques.

Dr. Fischer was a recipient of the Dissertation Award from the Technische Fakultät, University of Erlangen–Nürnberg, in 1997, the Publication Award of the German Society of Information Techniques in 2000, the Wolfgang Finkelnburg Habilitation Award in 2002, and the Philipp–Reis–Preis in 2005.