Multiple-Valued Debiasing for Phisically Unclonable Functions and Its Application to Fuzzy Extractors

Manami Suzuki, Rei Ueno, Naofumi Homma, and Takafumi Aoki

Tohoku University, Japan

Outline

- Background
- Preliminary and related works
- Proposed Multiple-valued debiasing
- Performance evaluation
- Concluding remarks

Background

- High demand for secure LSI authentication
- Physically unclonable function (PUF) is expected to prevent counterfeiting LSIs
  - Major features for authentication: Stability and Uniformity

Stability

Challenge		Response
$x^{(1)}$		$R^{(1)}$
$x^{(2)}$	$\longrightarrow$ PUF y $\longrightarrow$	$R^{(2)}$
$\vdots$		$\vdots$
$x^{(t)}$		$R^{(t)}$

If $x^{(1)} = x^{(2)} = \ldots = x^{(t)}$, then

$$R^{(1)} = R^{(2)} = \ldots = R^{(t)}$$

Uniformity

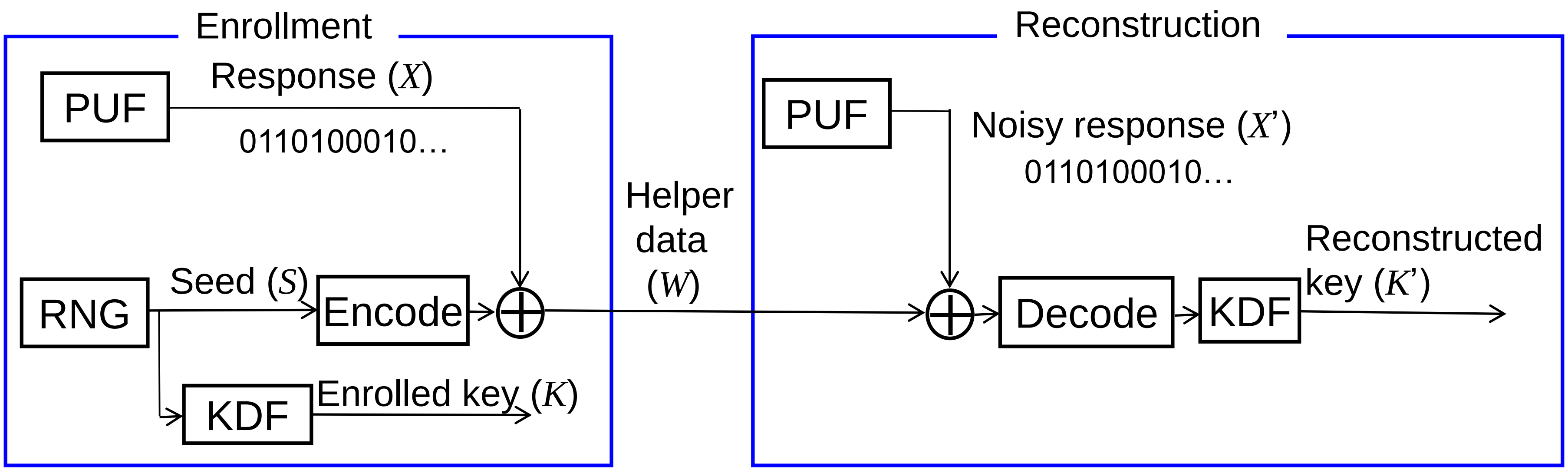
A response $\underbrace{10\cdots0110}_{p_0 \fallingdotseq p_1}$

p_0: Occurrence probability of 0
p_1: Occurrence probability of 1

- What if PUF response is unstable and biased?

Unstable and biased PUF response

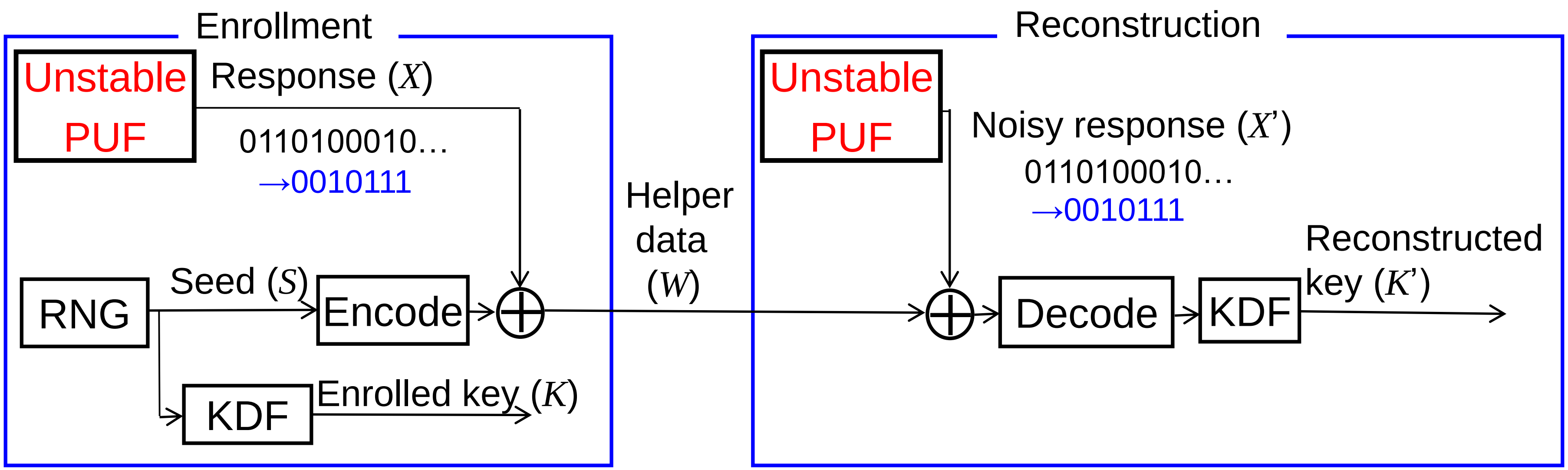
PUF-based key generation with Fuzzy extractor (FE)



- Problems on unstable and biased PUF response
  - Helper data leaks information about seed (entropy loss)
  - Difficult to extract entropy from unstable response

Unstable and biased PUF response

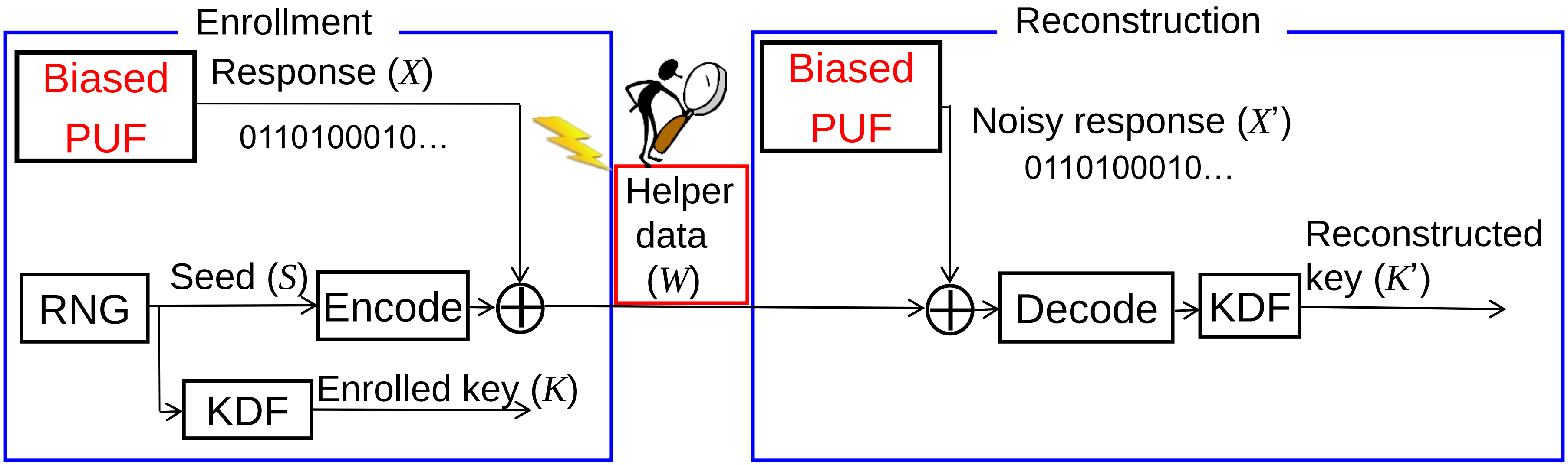
PUF-based key generation with Fuzzy extractor (FE)



- Problems on unstable and biased PUF response
  - Helper data leaks information about seed (entropy loss)
  - Difficult to extract entropy from unstable response

Unstable and biased PUF response

PUF-based key generation with Fuzzy extractor (FE)



- Problems on unstable and biased PUF response
  - Helper data leaks information about seed (entropy loss)
  - Difficult to extract entropy from unstable response

Extraction of PUF response

- Conventional methods for extracting stable and uniform response from unstable and biased PUFs
  - Multiple-valued response
    - Consider random (unstable) cell as stable cell to output third value
    - Higher entropy than binary
  - Debiasing
    - Debiased response would have full-entropy
    - Applied to PUF response prior to FE
- Multiple-valued response cannot work with FE ☹
  - Conventional FEs can accept only binary inputs
  - Limitation of application scenarios

Challenge Response

$x \rightarrow \boxed{\text{PUF}} \rightarrow R$

$R = 01?011??1??0$

?: Random cell

This work

Efficient extraction of stable and uniform response from unstable and biased PUFs

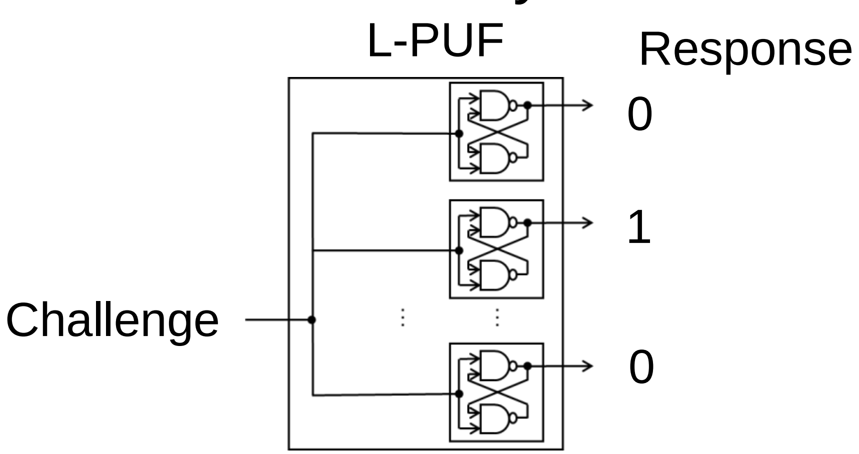
- Key trick
  - Multiple-valued debiasing
    - Input: multiple-valued response
    - Output: binary response that can be applied to FE
- Results
  - Proposed method can extract 36% longer full-entropy response than conventional one
  - Application to authentication with FE
    - 100% successful authentication even in some cases where conventional method fails

Outline

- Background
- Preliminary and related works
  - Unstable/Biased PUF and conventional debiasing
- Proposed multiple-valued debiasing
- Performance evaluation
- Concluding remarks

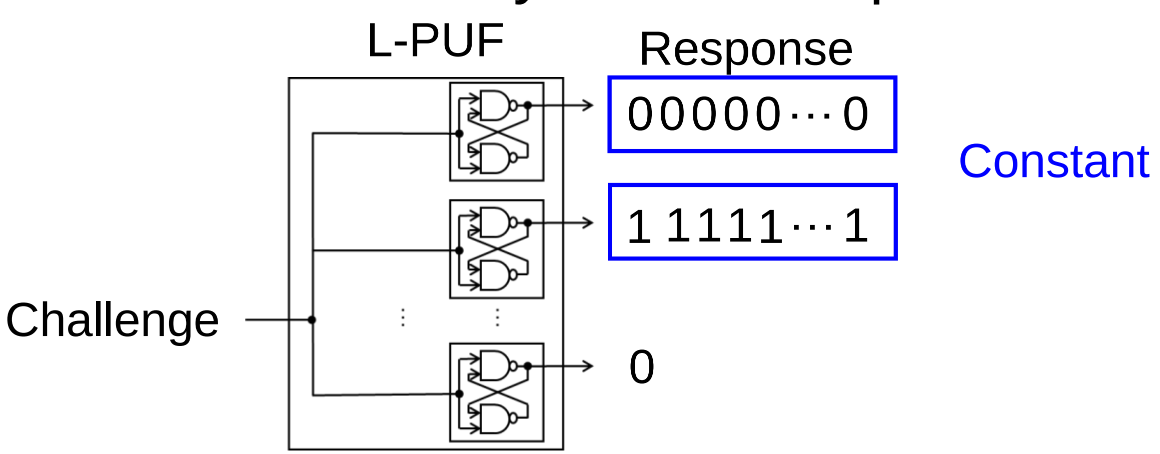
Unstable PUF

- n-bit PUF consists of n cells
  - Each cell outputs one-bit response at a measurement
- Two types of cells if same challenge is repeated
  - Constant cell: always 0 or always 1
  - Random cell: 0 or 1 at random
- Random cell is not preferable, because…
  - Cannot be used as response
  - reduce the stability of PUF response



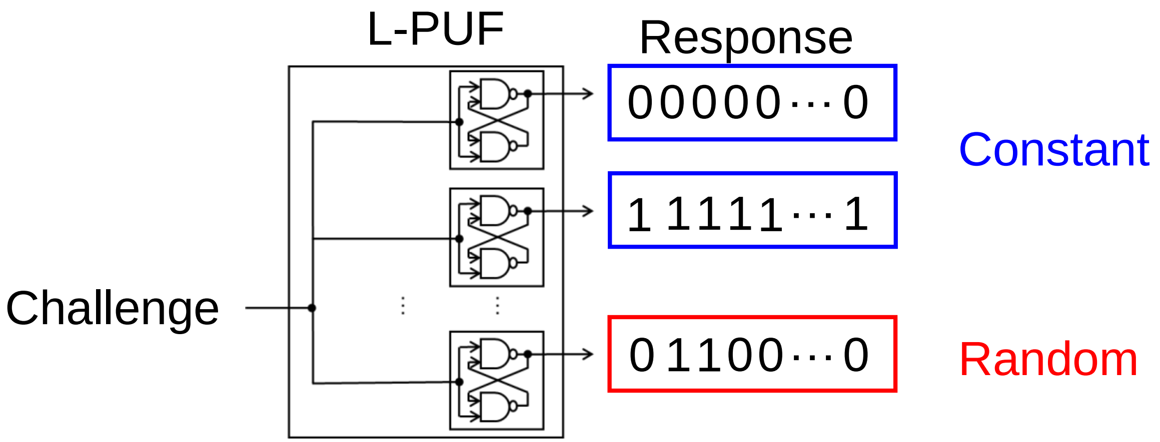
Unstable PUF

- n-bit PUF consists of n cells
  - Each cell outputs one-bit response at a measurement
- Two types of cells if same challenge is repeated
  - Constant cell: always 0 or always 1
  - Random cell: 0 or 1 at random
- Random cell is not preferable, because…
  - Cannot be used as response
  - reduce the stability of PUF response



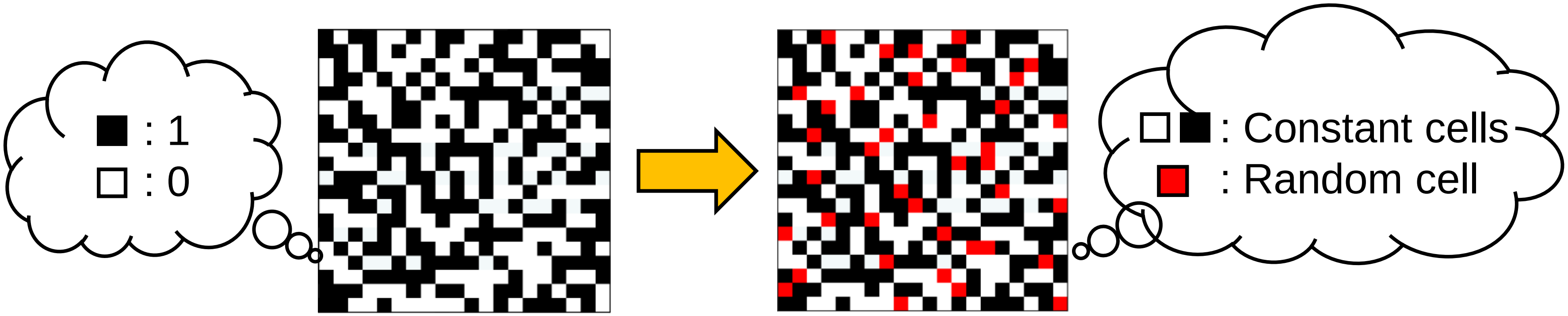
Unstable PUF

- n-bit PUF consists of n cells
  - Each cell outputs one-bit response at a measurement
- Two types of cells if same challenge is repeated
  - Constant cell: always 0 or always 1
  - Random cell: 0 or 1 at random
- Random cell is not preferable, because…
  - Cannot be used as response
  - reduce the stability of PUF response



Use of random cell: multiple-valued response

- Detect random cell and consider it as third value



Binary response
(Contain 10% erroneous cells)

Multiple-valued response
(No longer erroneous)

- How to assign “third value” to random cells

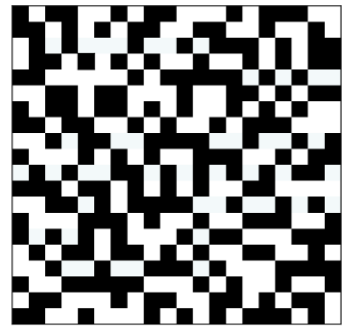
Type of cell		Assigned value
Constant	0	00
	1	11
Random		10

Ternary assignment by two bits [CHES11]

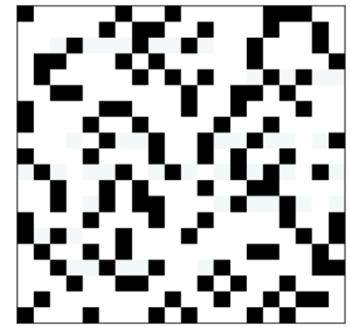
- Ternary response cannot work with conventional FEs

Biased PUF

- Bias has influence on secure key generation
  - p-biased PUF: $|Pr(X_i = 0) - 0.5| = p$
  - If bias is high, then entropy decreases
  - Typical FEs require $p < 0.082$

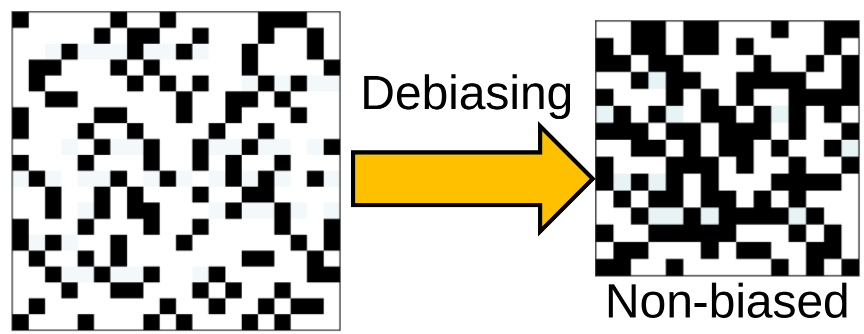


Non-biased (p = 0)



high-biased (p = 0.2)

- Debiasing
  - Extract low-biased response from high-biased one
  - Debiased response is shorter than original

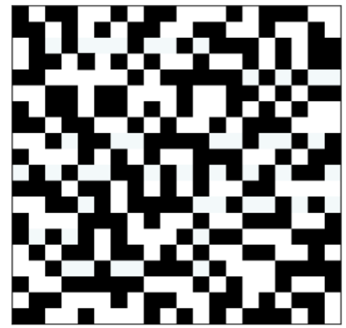


Biased response

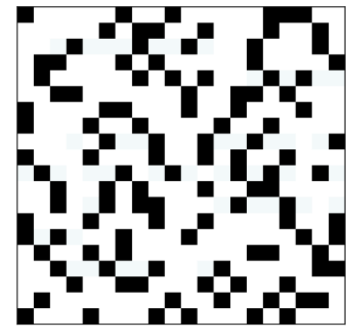
Non-biased response

Biased PUF

- Bias has influence on secure key generation
  - p-biased PUF: $|Pr(X_i = 0) - 0.5| = p$
  - If bias is high, then entropy decreases
  - Typical FEs require $p < 0.082$

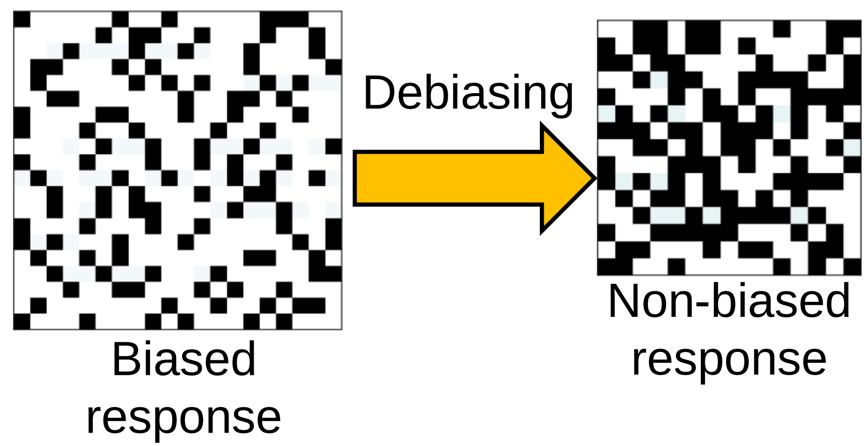


Non-biased (p = 0)



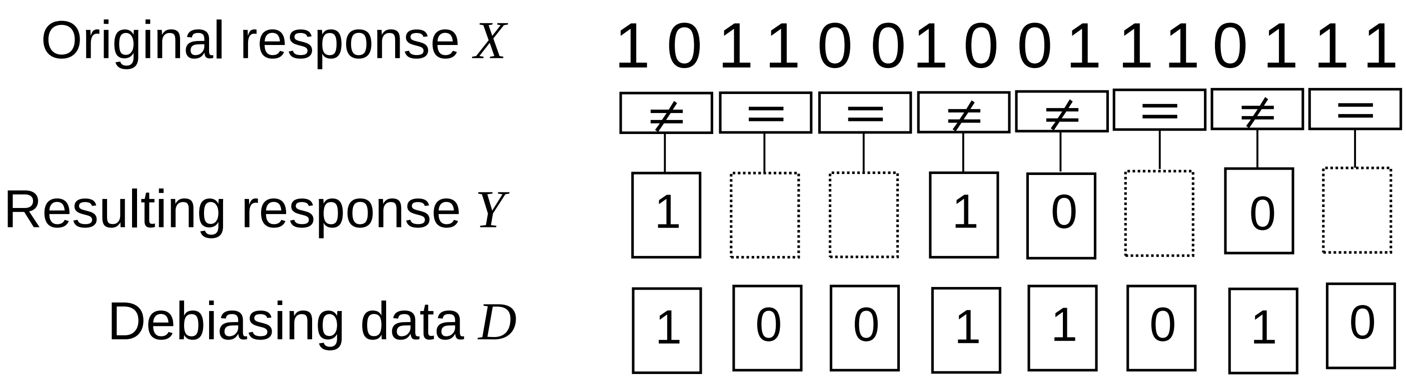
high-biased (p = 0.2)

- Debiasing
  - Extract low-biased response from high-biased one
  - Debiased response is shorter than original



Conventional debiasing method

- Classic deterministic randomness extractor (CDRE) proposed by von Neumann
  - Handle input bit string with a pair of two consecutive bits
  - (1, 0) and (0, 1) are assigned to 1 and 0, respectively
  - (0, 0) and (1, 1) are discarded
- Debiasing based on CDRE [CHES15]



Enrollment

Debiasing based on CDRE

- Enrollment
  - Generate debiased response Y and debiasing data D
- Reconstruction
  - Reconstructs noisy debiased response Y' based on D

Enrollment		
input	output	
$x_{2i}x_{2i+1}$	y_i	d_i
0 0	discard	0
0 1	0	1
1 0	1	1
1 1	discard	0

Reconstruction		
input		output
$x'_{2i}x'_{2i+1}$	d_i	y'_i
0 -	1	0
1 -	1	1
- -	0	discard

p_0, p_1 : Occurrence probability of 0 and 1 in X

Zeros and ones appear in Y with same probability $p_0 p_1$

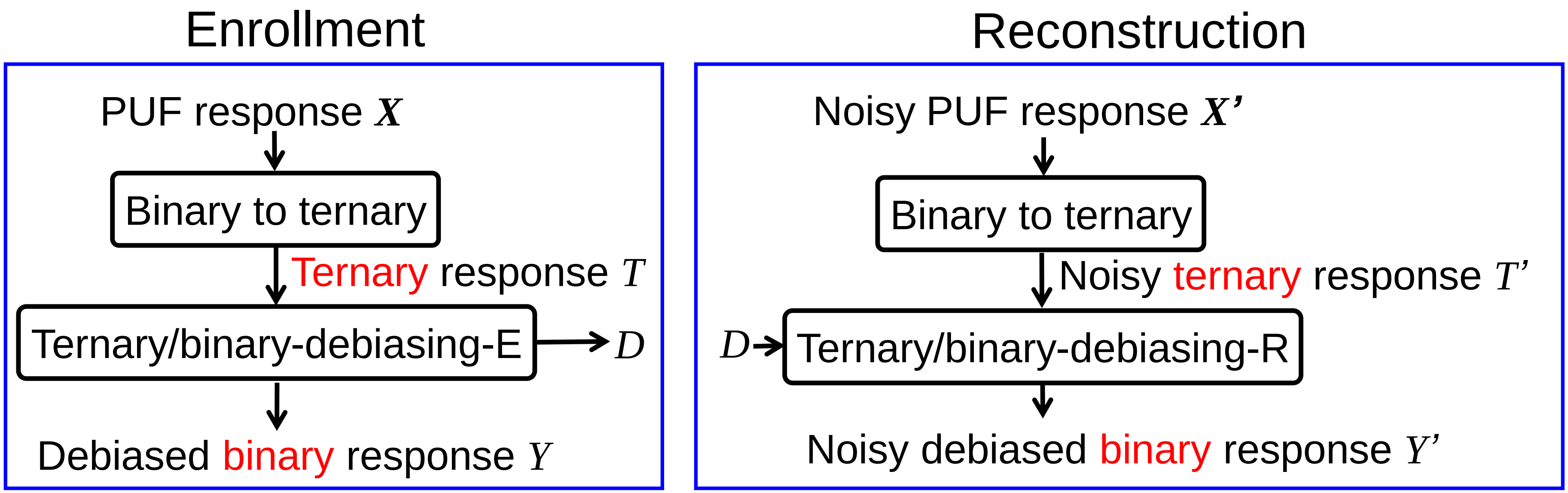
x_i : ith bit of X y_i : ith bit of Y
d_i : ith bit of D - : Don't care

Outline

- Background
- Preliminary and related works
- Proposed multiple-valued debiasing
- Performance evaluations
- Concluding remarks

Proposed debiasing method

- Input： ternary response
  - Ternary digit string with 0, 1, and r (random value)
- Output: debiased binary response



- Conventional FEs can be used together with proposed debiasing method

Proposed debiasing method

- Handle input with a pair of consecutive digits
- Perform error correction in reconstruction

Enrollment		
input	output	
$t_{2i}\, t_{2i+1}$	y_i	d_i
0 0	discard	0
1 1	discard	0
r r	discard	0
0 1	0	1
r 1	0	1
0 r	0	1
1 0	1	1
r 0	1	1
1 r	1	1

Reconstruction		
input		output
$t'_{2i}\, t'_{2i+1}$	d_i	y'_i
0 -	1	0
1 -	1	1
r r	1	1
r 0	1	1
r 1	1	0
- -	0	discard

Both 0s and 1s appear by probability $p_0\, p_1 + p_0\, p_r + p_1\, p_r$ in resulting response

p_0, p_1, p_r :
Occurrence probability of constant cell (0 or 1) and random cell (r) in X

t_i: ith bit of T, y_i: ith bit of Y
d_i: ith bit of D, - : Don't care

Error bits in reconstruction

- Error patterns of response bits in reconstruction

Binary response

1	0	1	1	$\cdots$	0	1
2	0	1	0	$\cdots$	0	1
3	0	1	1	$\cdots$	1	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	0	1	0	$\cdots$	0	1

Ternary response

Enrollment	0	1	r	$\cdots$	r	1
Reconstruction						

Error bits in reconstruction

- Error patterns of response bits in reconstruction

Binary response

1	0	1	1	$\cdots$	0	1
2	0	1	0	$\cdots$	0	1
3	0	1	1	$\cdots$	0	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
n	0	1	0	$\cdots$	0	1

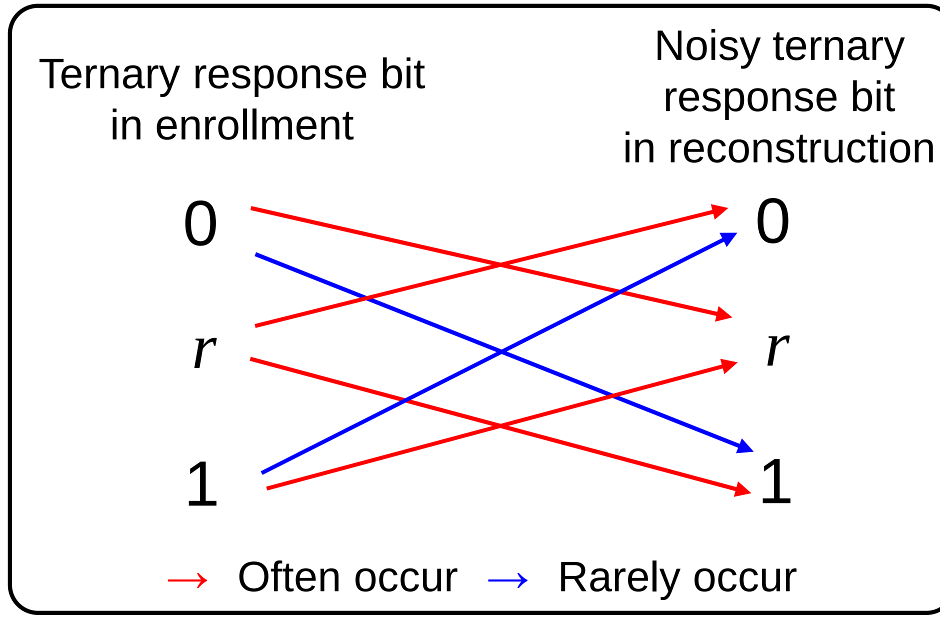
Ternary response

	⬇	⬇	⬇		⬇	⬇
Enrollment	0	1	r	$\cdots$	r	1
Reconstruction	0	1	r	$\cdots$	0	1

Error bits in reconstruction

- Error patterns of response bits in reconstruction

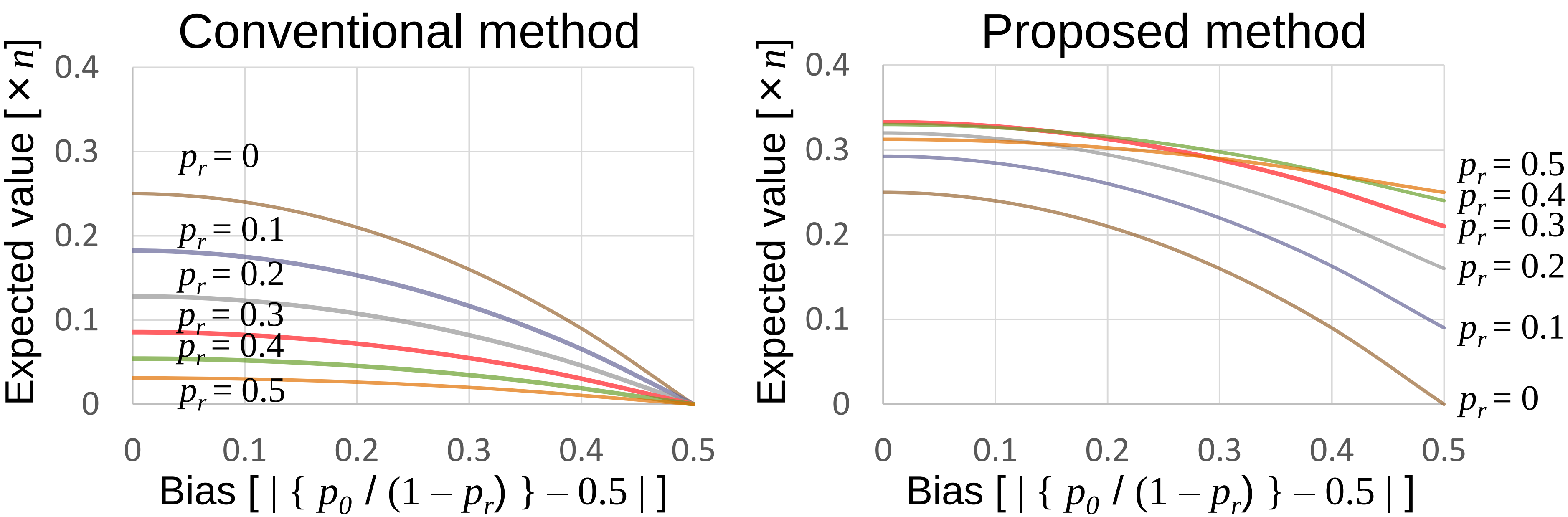
Binary response						
1	0	1	1	⋯	0	1
2	0	1	0	⋯	0	1
3	0	1	1	⋯	0	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮
n	0	1	0	⋯	0	1
Ternary response	⬇	⬇	⬇		⬇	⬇
Enrollment	0	1	r	⋯	r	1
Reconstruction	0	1	r	⋯	0	1



Proposed method is considered as error correction using a code {(0, 1), (1, 0)} with erasure symbol r

Expected entropy after debiasing

- $E_{Conv} = np_0 p_1(1\text{-}p_r)$
- $E_{Proposal} = n\,(p_0 p_1 + p_0 p_r + p_1 p_r)$



- Random cells contribute to entropy in proposed method

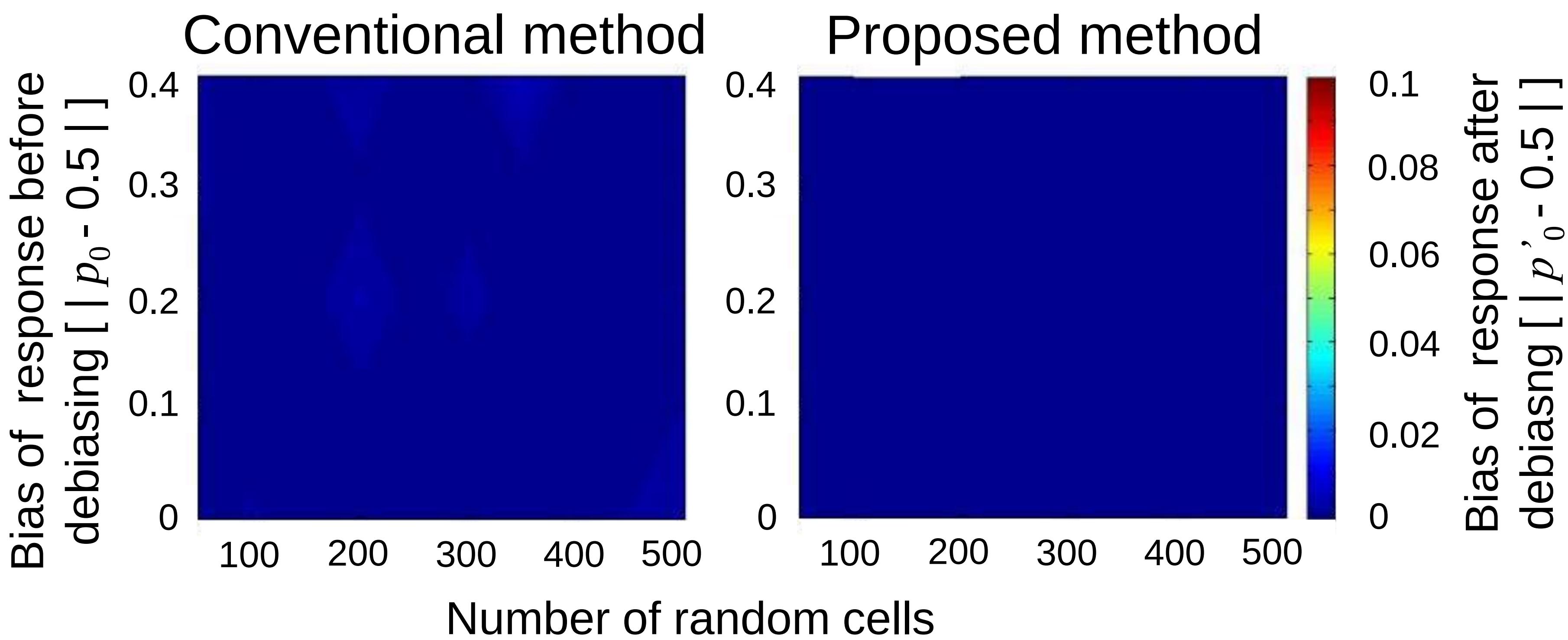
Outline

- Background
- Related works
- Proposed multiple-valued debiasing
- Performance evaluations
- Concluding remarks

Experimental simulation

- Evaluate resulting bias and response length
- Generate ternary responses by simulation
  - Length of ternary response: 1,024
  - With different bias and number of random cells
    - Bias range from 0 to 0.5
    - Number of random cells from 50 to 500
  - Number of responses for each parameter: 1,000

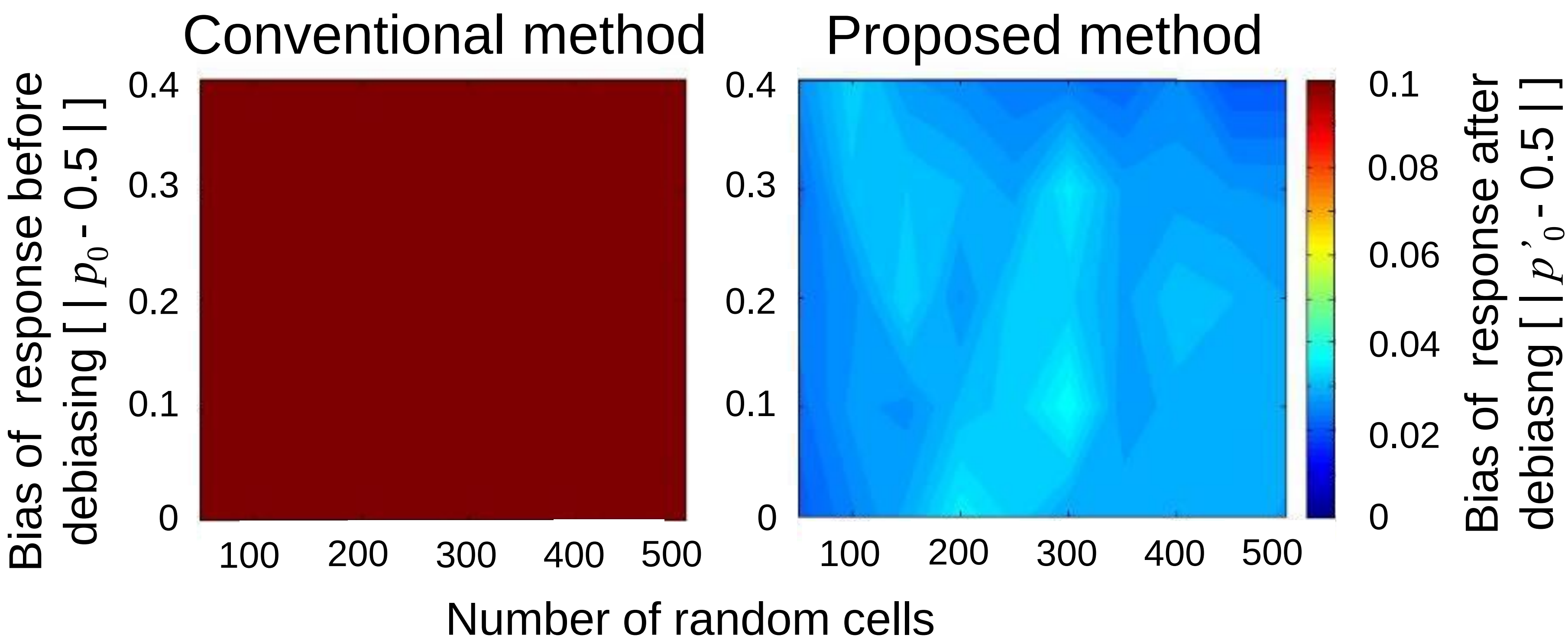
Average bias of resulting response



Condition for secure key generation
with a typical FE: $|p'_0 - 0.5| < 0.082$

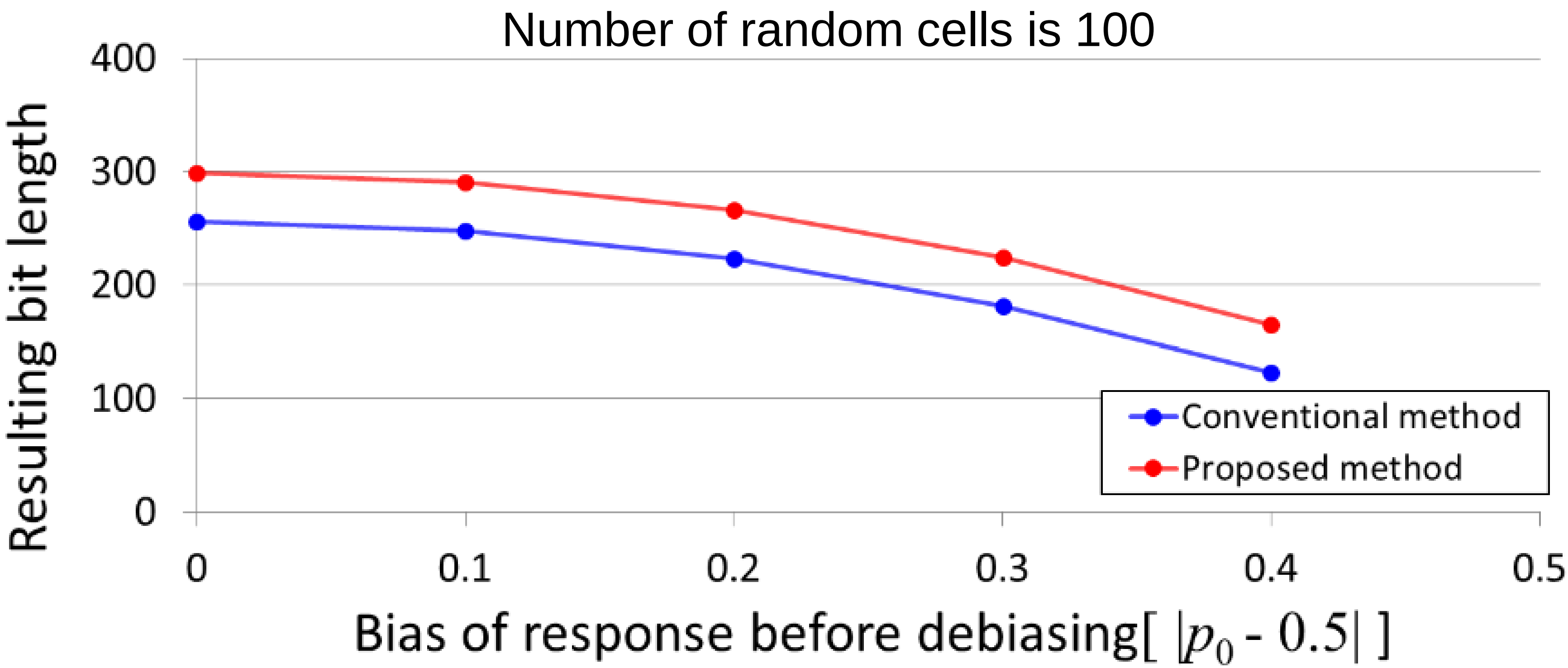
- Both responses on average satisfied the condition

Worst-case bias of resulting response



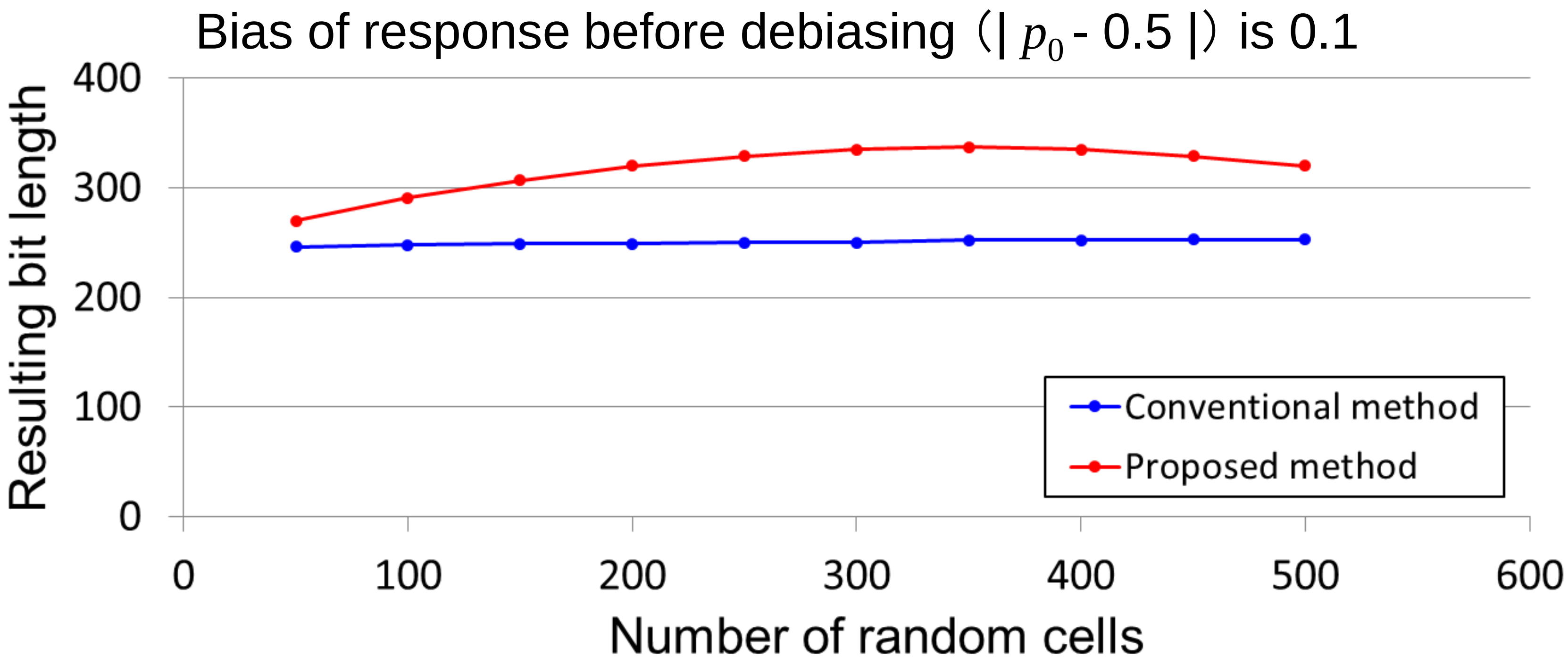
- Responses extracted by proposed method satisfied the condition even in worst-case
  - Use of ternary response increases entropy of response

Resulting bit length for different biases



- High-bias results in short response in both methods
- Proposed method obtained 22% longer bit length than conventional method
  - Use of ternary response can extract high entropy

Resulting bit length for different # of random cells

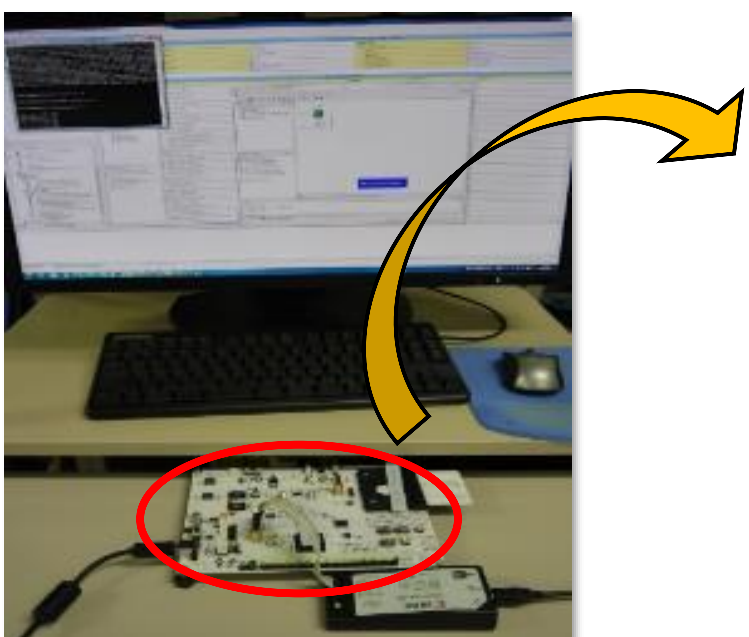


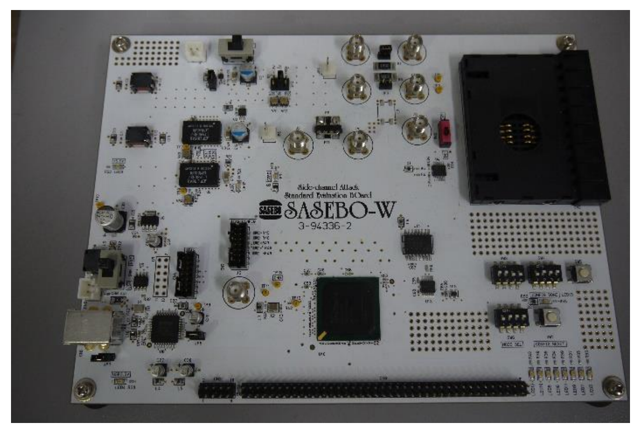
- Proposed method extracted longest bit length when the number of random cells was 300-400
  - Entropy of ternary response is largest when number of random cells is one-third of all cells

Experiment with FPGA implementation

- Implement Latch-PUF on FPGA
  - Using 3 FPGAs (Xilinx Spartan 6)
  - Implemented at 10 different locations

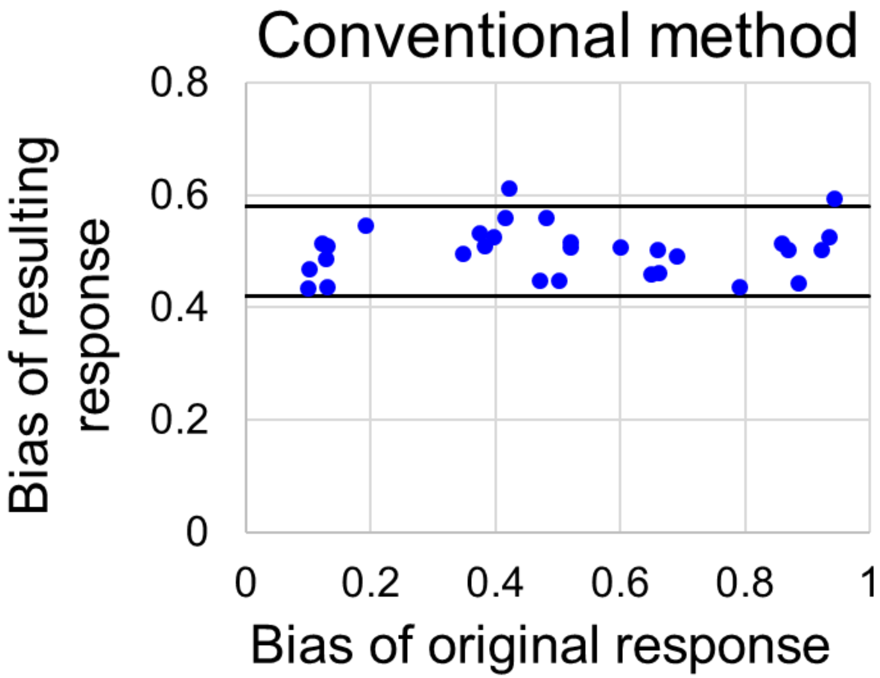
  30 L-PUFs
  - Response bit length：1,024
  - Number of challenges to detect random cells: 256

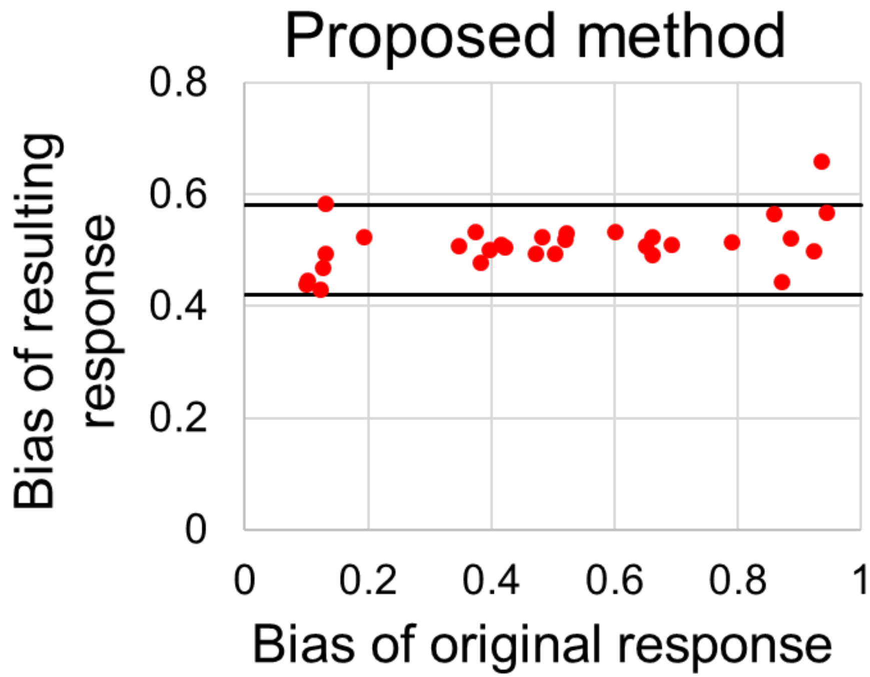




Xilinx Spartan 6

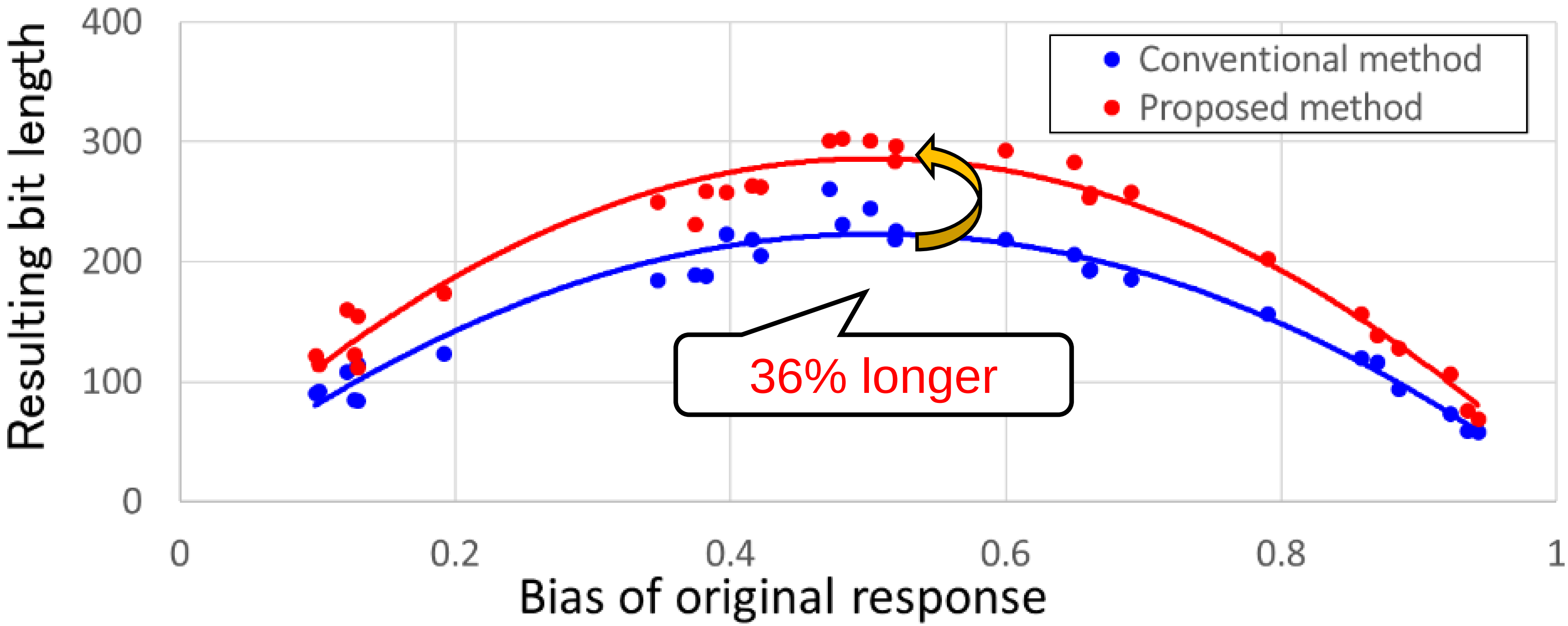
Biases of resulting responses





- Both methods reduced biases significantly
  - Percentage of random cells was ~10% in the experiment

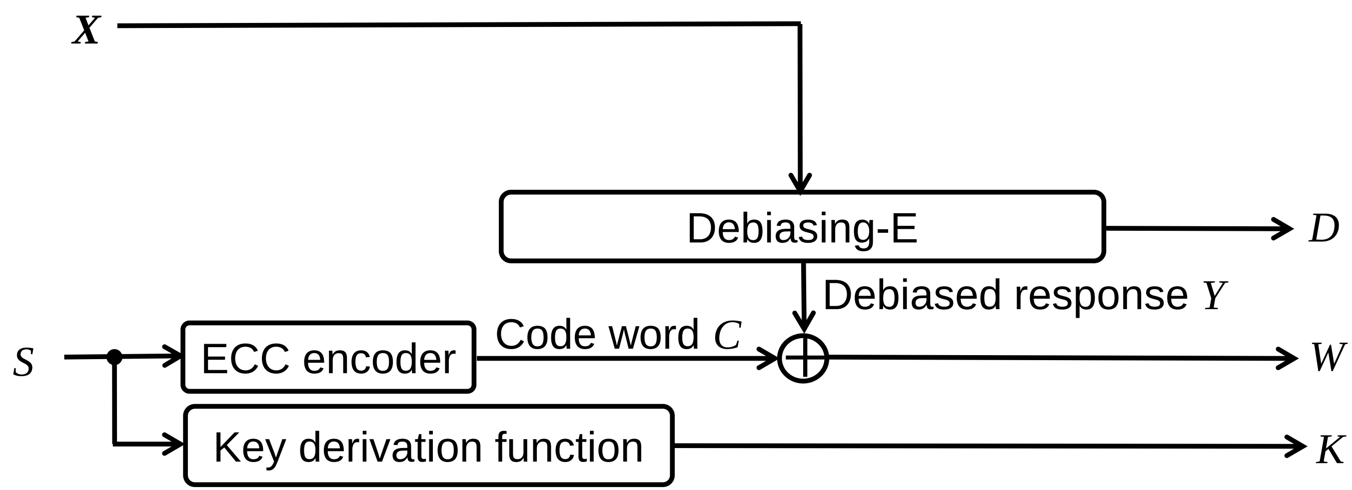
Resulting bit length for original biases



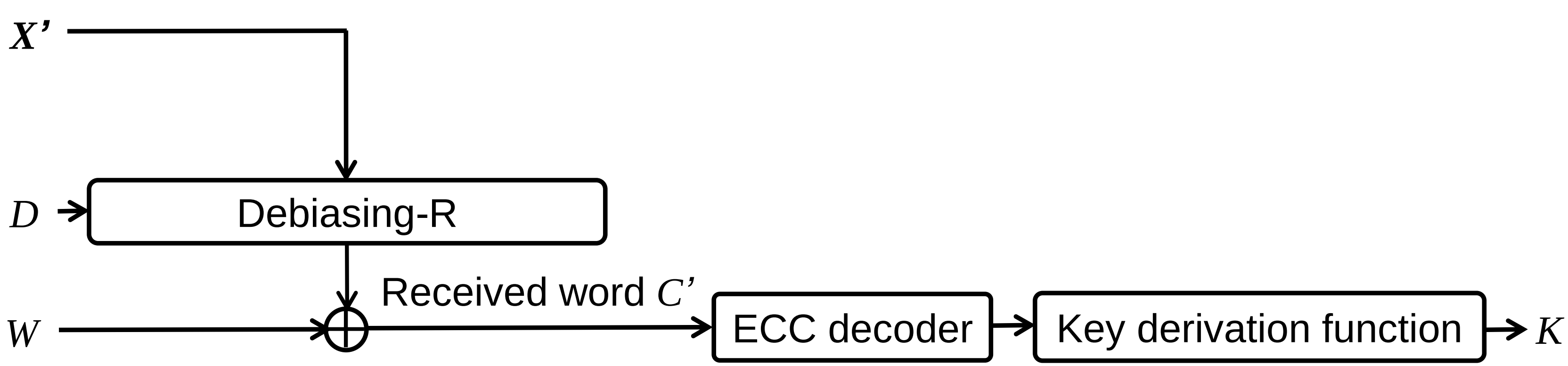
- High bias decreases resulting bit lengths for both methods as the same as in simulation
- Proposed method could extract larger bit length

FE using proposed debiasing method

- Enrollment

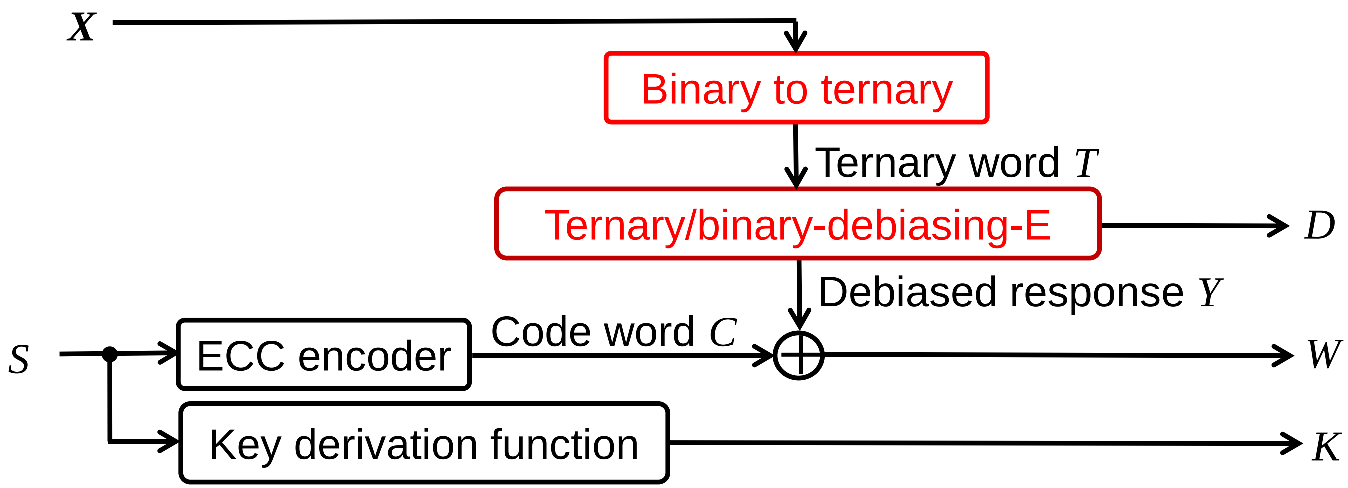


- Reconstruction

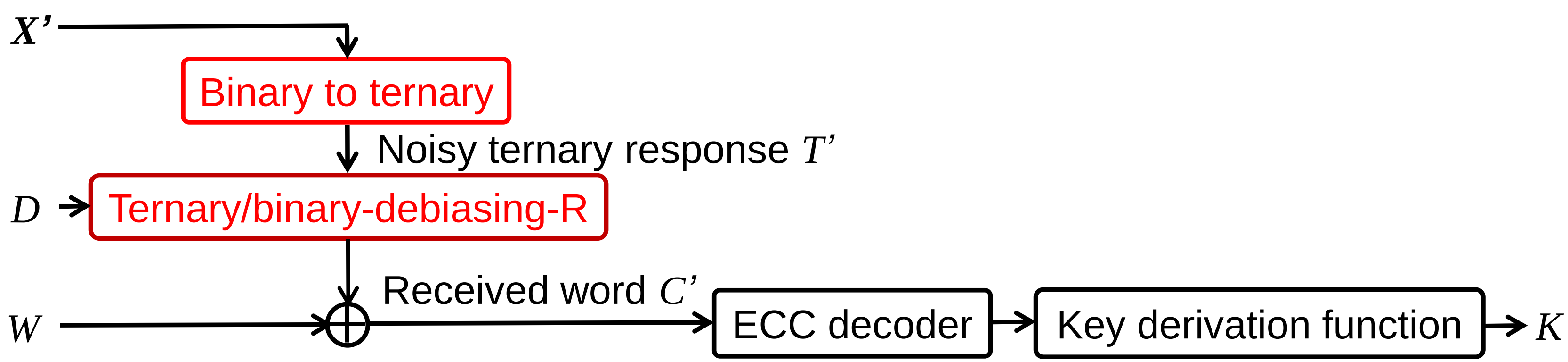


FE using proposed debiasing method

- Enrollment



- Reconstruction



Performance evaluation for FEs

- Evaluation of FEs with simulated PUF responses
  - Comparison of authentication failure rate and efficiency

Efficiency = debiased bit length / original PUF response length

- Simulated response based on L-PUF implemented on FPGA
- ECC in FE: connected code
  - (24,12) Golay code and (8,1) repetition code

Comparison of debiasing results by FEs

Bias	Random cell	Conventional method		Proposed method	
		P_{fail}	Efficiency	P_{fail}	Efficiency
0.1	0.1	0	0.236	0	0.286
	0.2	0	0.237	0	0.312
	0.3	0.013	0.243	0	0.328
0.3	0.1	0	0.172	0	0.220
	0.2	0.002	0.184	0	0.264
	0.3	0.240	0.195	0	0.287

10,000 challenges

- $P_{fail} = 0$ under experimental conditions
  - Thanks to high stability of multiple-valued response
  - Proposed method does not require strong ECC in FE

Comparison of debiasing results by FEs

Bias	Random cell	Conventional method		Proposed method	
		P_{fail}	Efficiency	P_{fail}	Efficiency
0.1	0.1	0	0.236	0	0.286
	0.2	0	0.237	0	0.312
	0.3	0.013	0.243	0	0.328
0.3	0.1	0	0.172	0	0.220
	0.2	0.002	0.184	0	0.264
	0.3	0.240	0.195	0	0.287

10,000 challenges

- $P_{fail} = 0$ under experimental conditions
  - Thanks to high stability of multiple-valued response
  - Proposed method does not require strong ECC in FE
- Our method achieved 21-47% higher efficiency
  - Efficiency is high when more random cells appear

Concluding remarks

- Multiple-valued response extraction can be used with key generation based on FE
  - Improved stability and longer full-entropy response
    - Even in worst-case bias, our method satisfied the condition to generate secret information securely
    - 36% longer full-entropy than conventional binary debiasing in an experiment

- Future works
  - ECC design taking advantage of proposed method
  - Further evaluation using other types of PUFs