Workshop on Cryptographic Hardware and Embedded Systems (CHES 2020)

Lightweight Authenticated Encryption Mode of Operation for Tweakable Block Ciphers

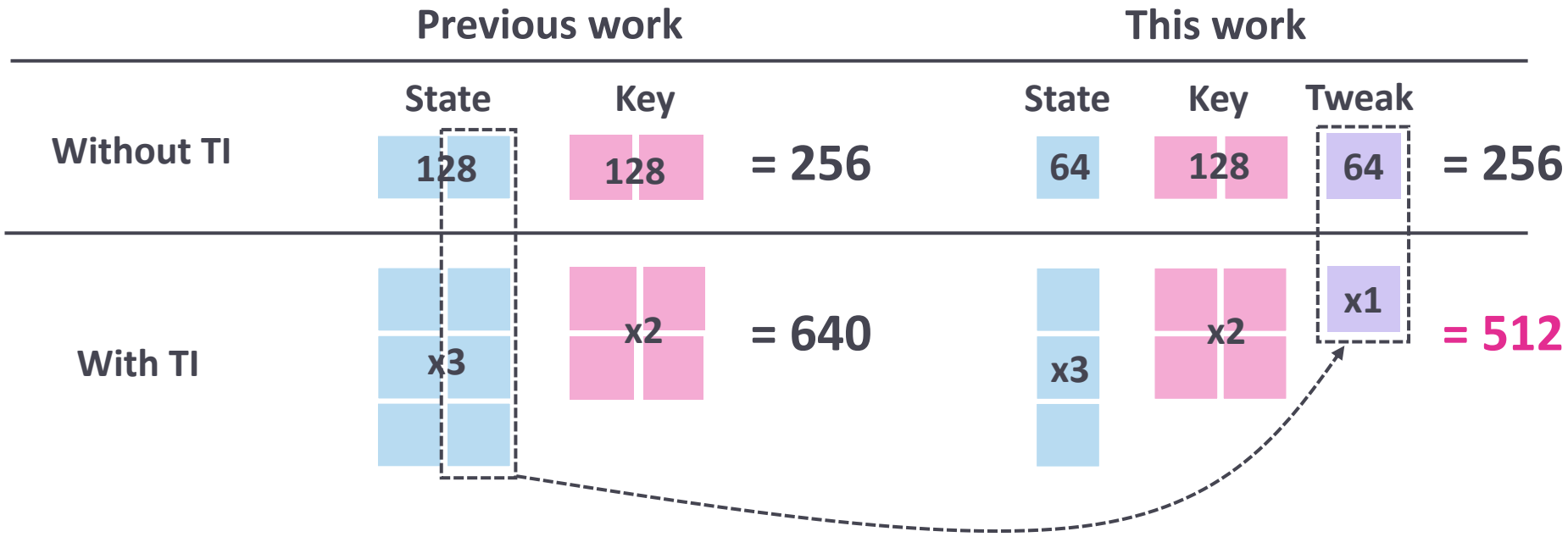
Yusuke Naito* and Takeshi Sugawara**

*Mitsubishi Electric Corporation

**The University of Electro-Communications

Our New Design: PFB (Plaintext Feedback) Mode

- **Key features**
  - 64-bit security with a 64-bit tweakable block cipher (the beyond-the-birthday-bound security)
  - **Low memory usage with threshold implementation (TI)**
    - By replacing a non-linearly updated 64-bit state into a public tweak



Lightweight Cryptography

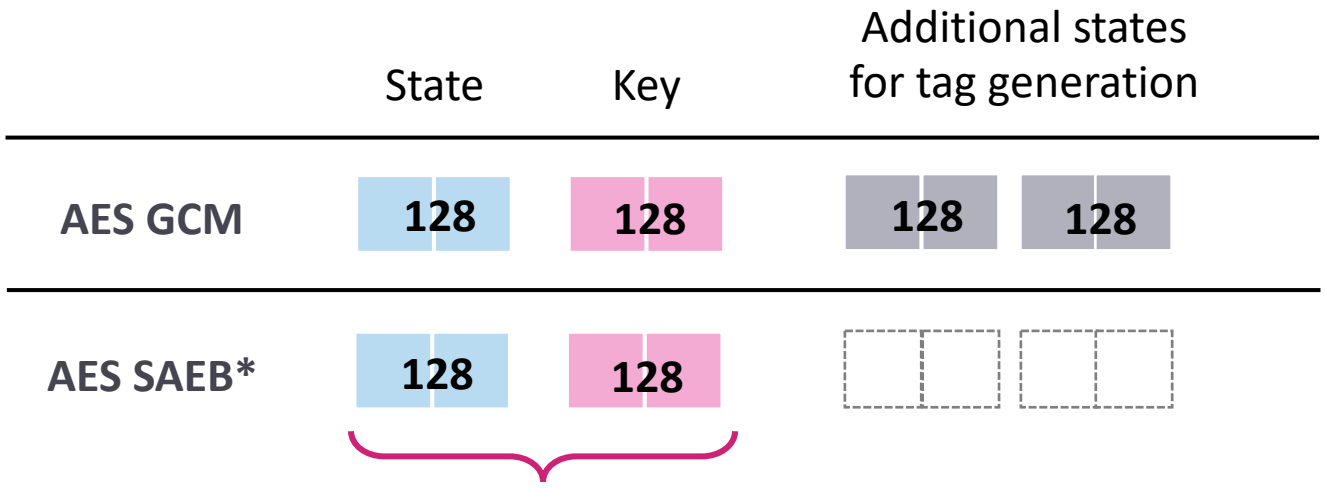
- Security for resource-constrained IoT devices

- Lightweight block ciphers
  - Standardization
  - **64-bit primitives** are popular

- Memory (register) is a bottleneck in hardware implementation
  - 4-bit S-box: 20--40 gates
  - 128-bit register: 600--900 gates

Lightweight Authenticated Encryption (AE)

- NIST is running a competition (LWC) for choosing a lightweight AE
- Optimizing the mode of operation for lightweight implementation
  - **Only 32-bit security** when combined with a mode of operation with the birthday-bound security, **which is subject to a practical attack****



*Y. Naito, M. Matsui, T. Sugawara, and D. Suzuki, “SAEB: A Lightweight Blockcipher-Based AEAD Mode of Operation,” CHES 2018.

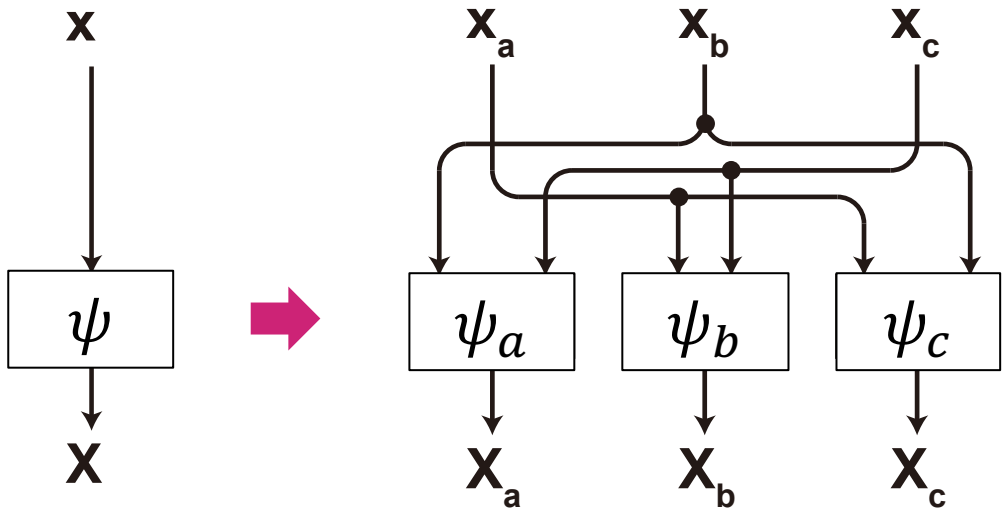
** K. Bhargavan, G. Leurent "On the Practical (In-)Security of 64-bit Block Ciphers: Collision Attacks on HTTP over TLS and OpenVPN", CCS2016.

Lightweight + SCA Resistance

- Resource-constrained devices are used in a hostile environment in which side-channel attack (SCA) is a serious threat
- **SCA protection in resource-constrained devices** is even more challenging
- Lightweight cryptography that enable efficient SCA countermeasure is a new frontier of research, e.g., TI-friendly S-box and SCREAM

(1st order) Threshold Implementation

- Encode a sensitive value as a share, and implement crypto while preserving the shared representation
- Efficiency provides security in the presence of glitches
- **Multiplies the memory cost!**

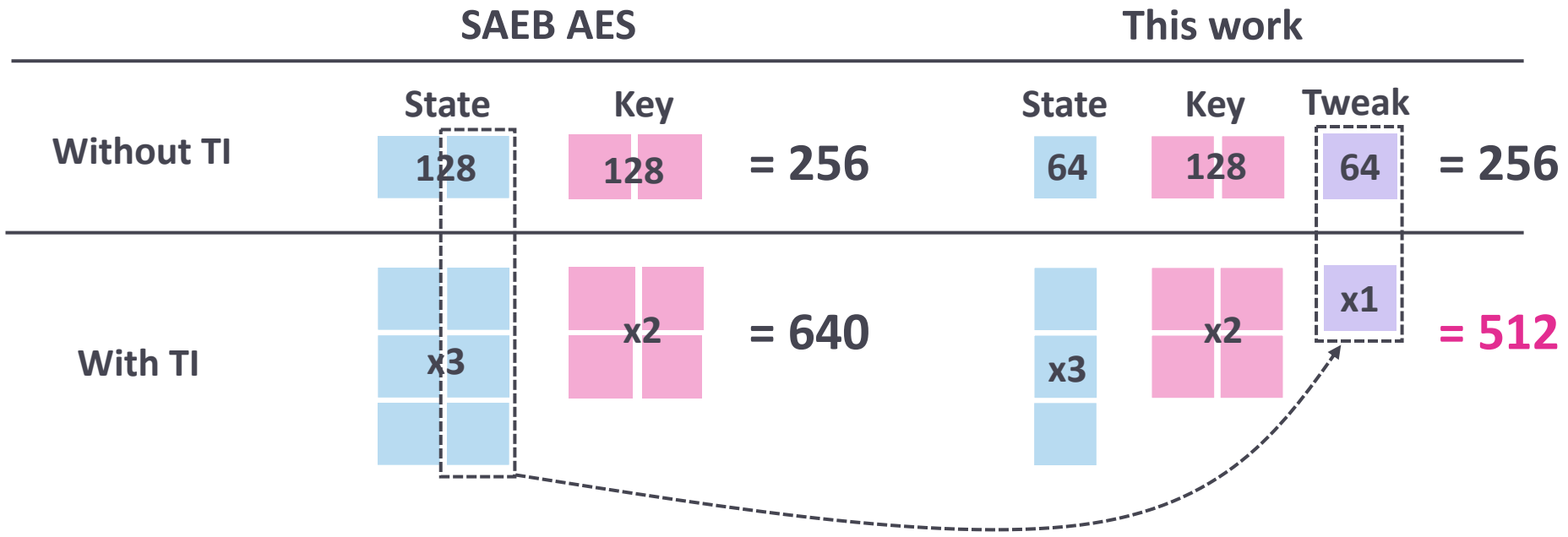


Input share (x_a, x_b, x_c)
satisfying $x_a \oplus x_b \oplus x_c = x$

Output share (X_a, X_b, X_c)
satisfying $X_a \oplus X_b \oplus X_c = X$

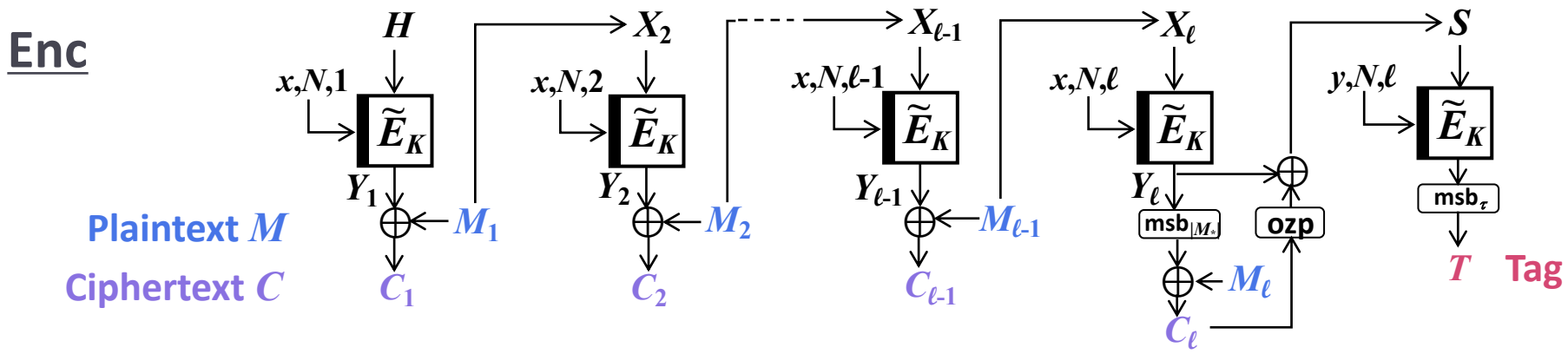
Reduce the Size of Non-Linearly Updated State

- **Low memory usage with threshold implementation (TI)**
- Challenge: birthday-bound security
  - We use a tweakable block cipher (TBC) to efficiently achieve the beyond-the-birthday-bound security, i.e., 64-bit security with a 64-bit primitive



New Mode of Operation PFB (Plaintext Feedback)

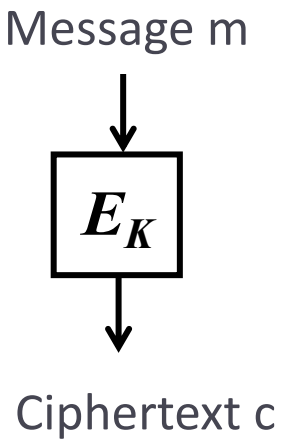
- A nonce-based authenticated encryption with associated data using TBC
  - Provides the beyond-the-birthday-bound security: security level = block length
- Based on iCOFB (Chakraborti et al. CHES2017) with several improvements:
  - Adding associate-data processing
  - Supporting arbitrary-length message
  - Giving a new proof for a tighter security bound
- Hardware performance evaluation with TI



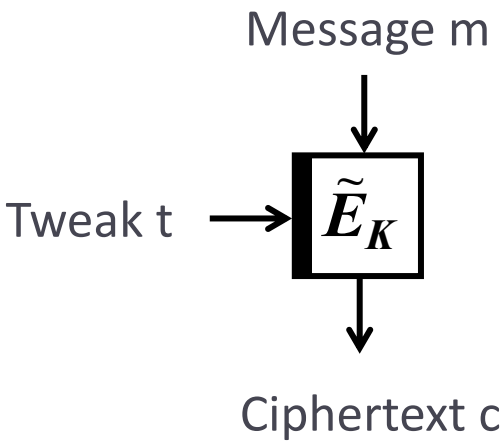
Tweakable block cipher

- An extension of a block cipher with the third input called tweak
- We get an independent random permutation for each tweak, i.e., efficient rekeying

Block cipher

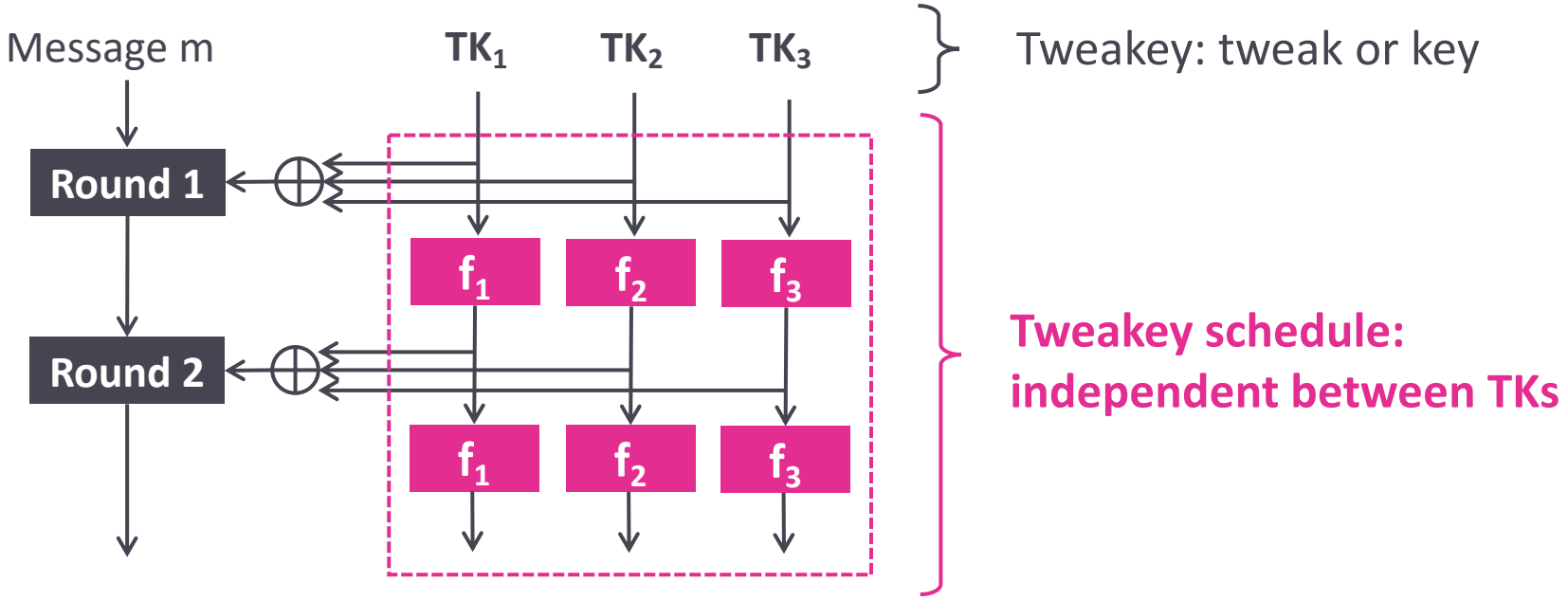


Tweakable block cipher



Tweakable block cipher SKINNY

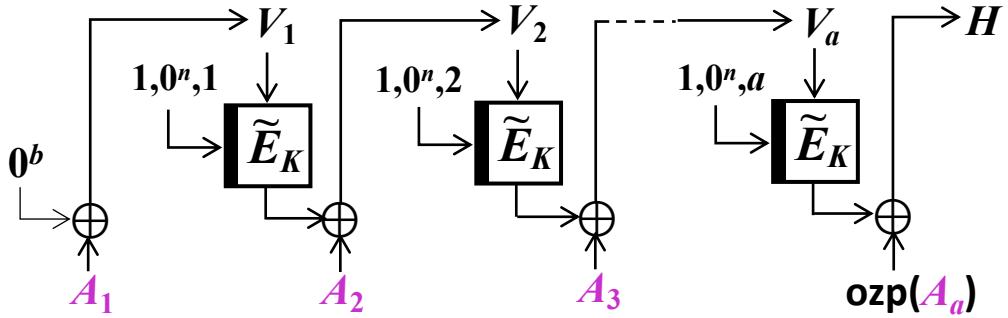
- A popular lightweight TBC
- Tweakey framework: no discrimination between the key and tweak



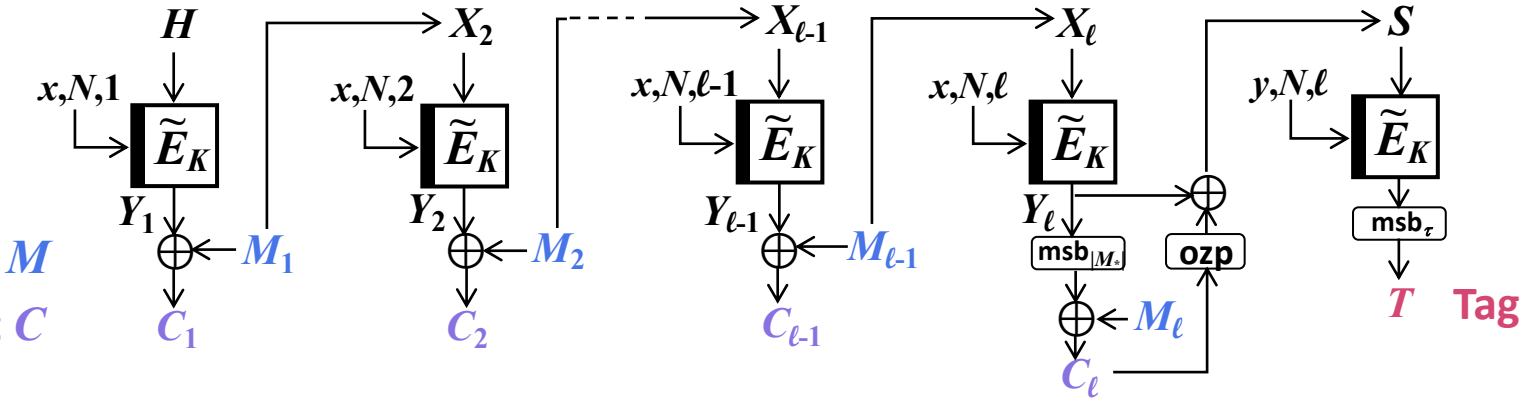
Beierle et al., "The SKINNY Family of Block Ciphers and Its Low-Latency Variant MANTIS," In *CRYPTO 2016*.

PFB

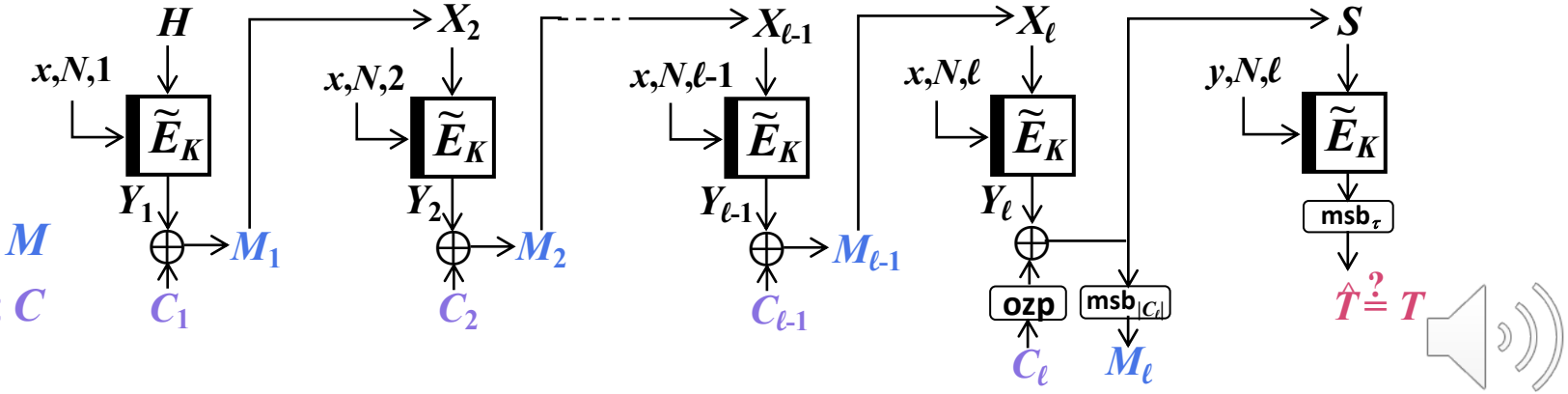
Hash



Enc

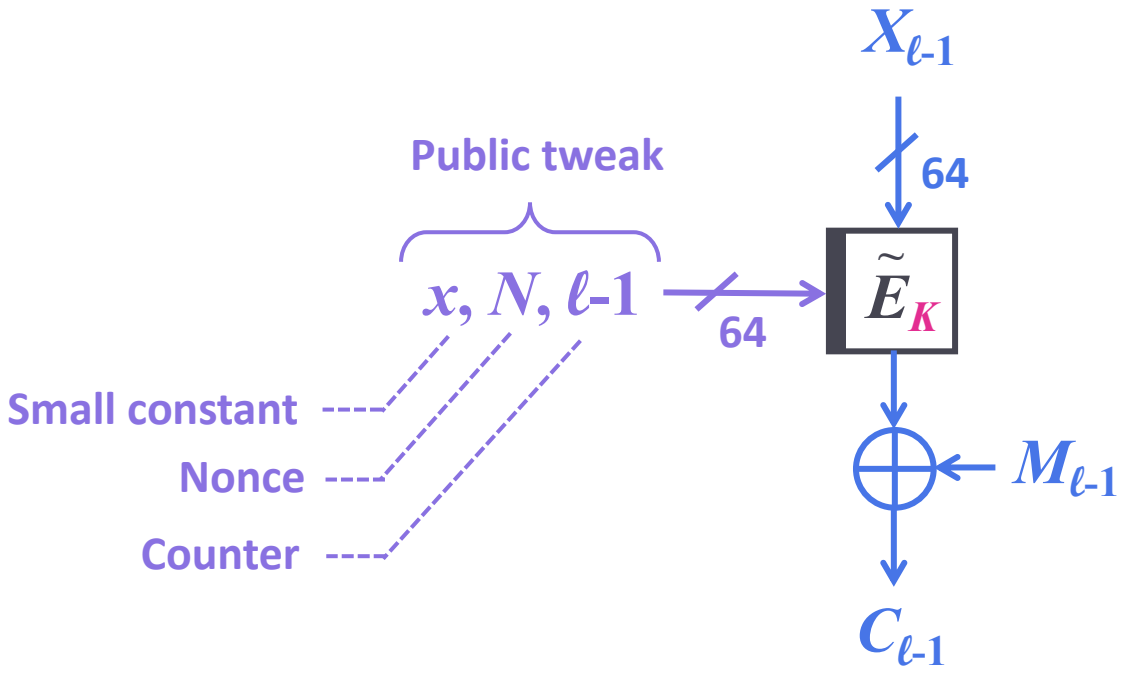


Dec



PFB cont.

- Memory for running a TDC is sufficient for the entire PFB operation.
- Tweak contains public parameters: a constant, nonce, and counter



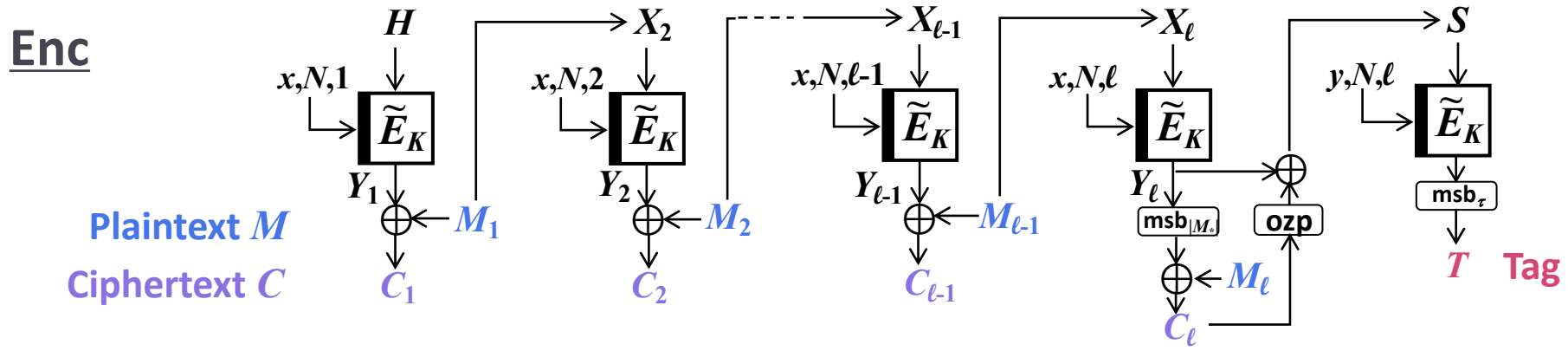
	Memory size			
	State	Key	Tweak	
Without TI	64	128	64	= 256
With TI	x3	x2	x1	= 512

Security of PFB

- Target: b-bit security with the b-bit block length
- Assumption
  - TBC as a TRP (Tweakable Random Permutation)
  - Nonce respect setting (i.e., no nonce misuse)
- Privacy
  - Game: distinguishing a ciphertext from a random sequence
  - **PFB achieves perfect security**
- Authenticity
  - Game: forging a valid tag with the query access to the decryption oracle
  - **A successful attack needs 2^b decryption queries, i.e., PFB achieves b-bit security**

Proof sketch for privacy

1. No repeated tweak in encryption
   - ∵ the (non-repeated) nonce and a counter
2. TBC's output $Y_1, Y_2, \ldots, T$ are random and independent by the TRP assumption
3. We cannot distinguish the ciphertexts and tag from a random string, i.e., achieves perfect security



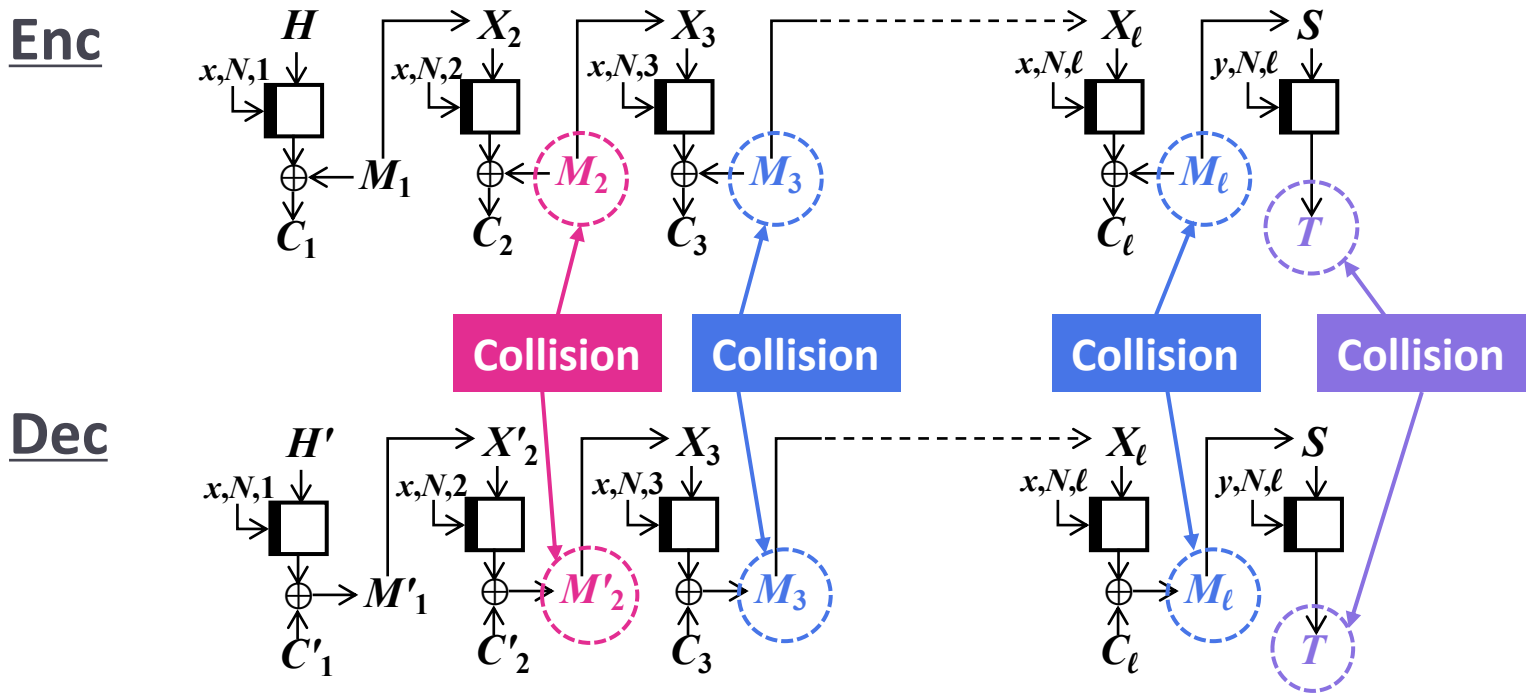
Proof sketch for authenticity

- We consider two attack cases

- Attack case #1: guessing the tag in PFB's decryption
  - The success probability is roughly $1/2^b$ for each query because the tag is almost randomly chosen
  - The probability **$\Pr[\#1] \leqq O(q_D/2^b)$** with q_D queries to the Decryption oracle

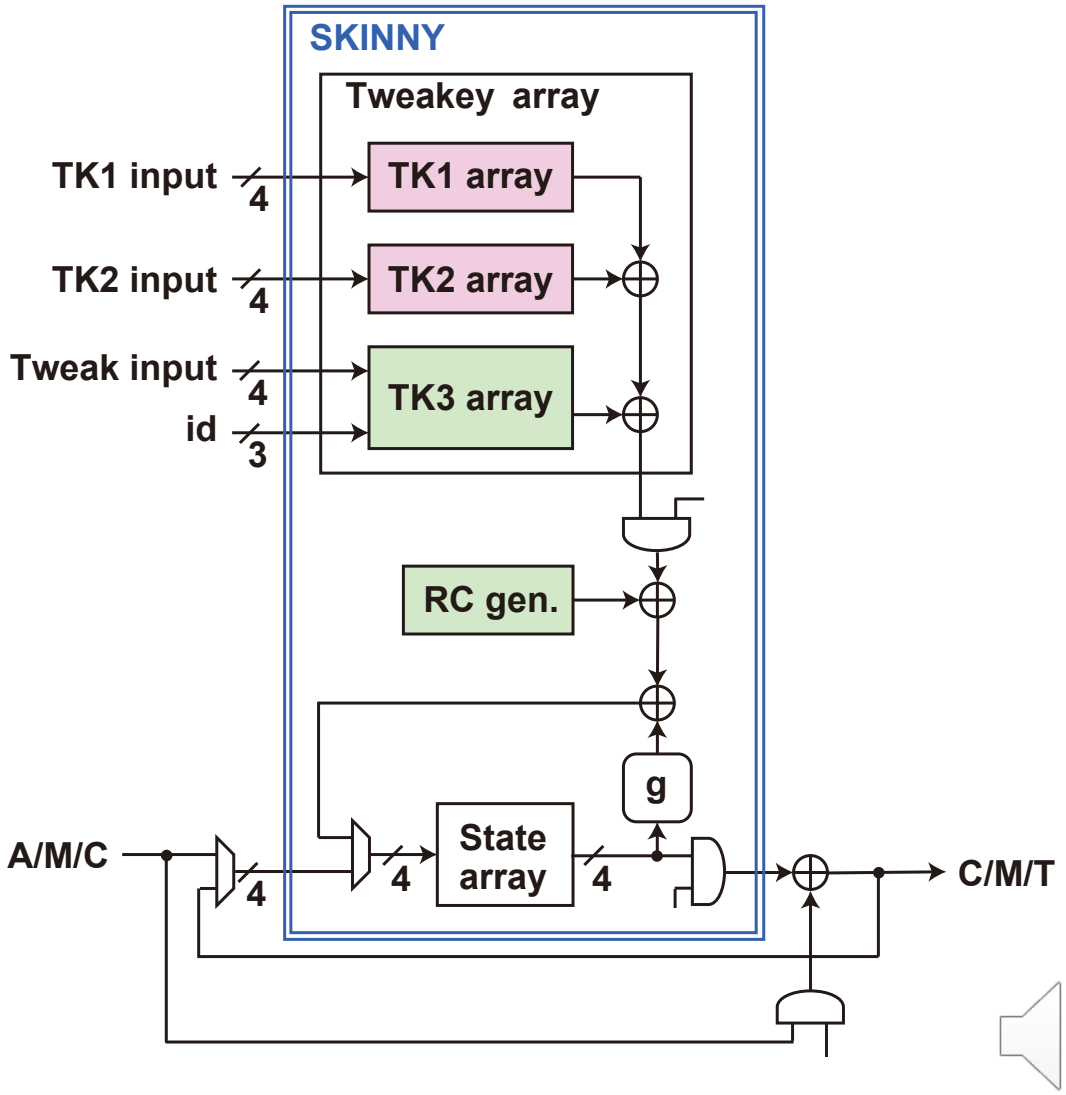
Proof sketch for authenticity cont.

- Attack case #2: exploiting the collision in the PFB states
  - A collision in between the Enc and Dec states with the same nonce results in a collision of the tag, i.e., successful tag forgery
  - The probability to observe a collision is $1/2^b$, so **$\Pr[\#2] \leqq O(q_D/2^b)$** with q_D Decryption queries



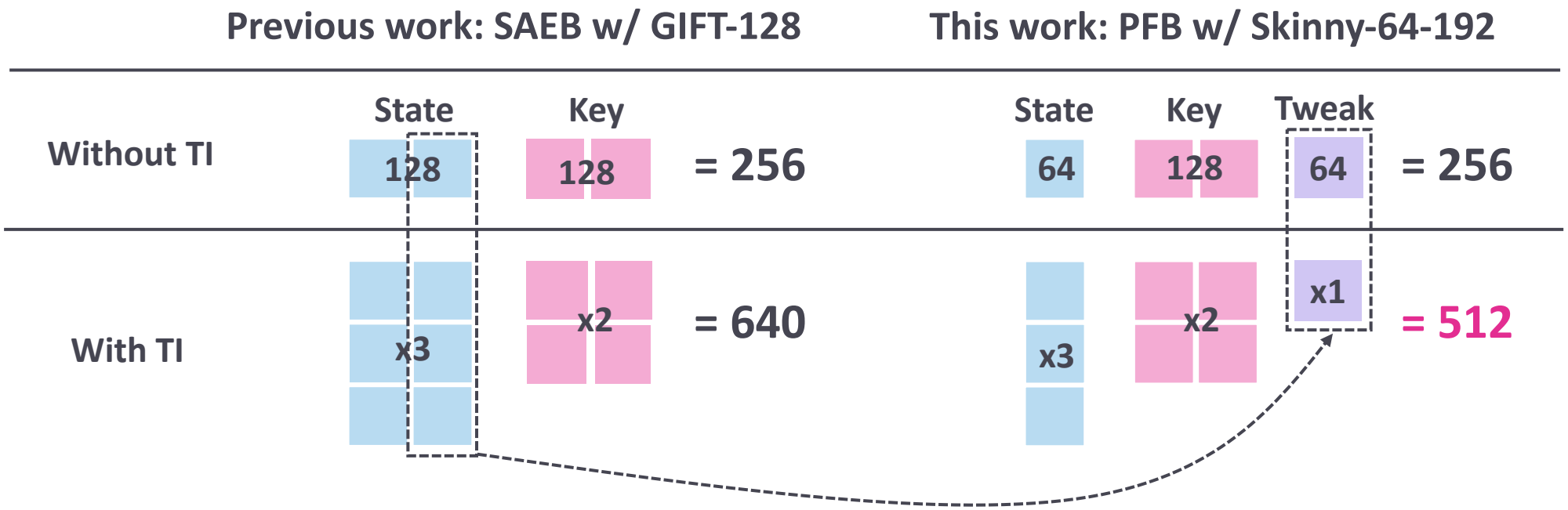
Hardware architecture

- PFB with SKINNY-64-192 (a variant with 64-bit block and 192-bit tweakey)
- A serial SKINNY architecture with 4-bit datapath
- The mode of operation is a thin wrapper: with the MUX, XOR, AND gates
- **Heterogeneous number of shares**
  - **Green: 1-share (public)**
  - **Red: 2-share (linear secret)**
  - **Others: 3-share (nonlinear secret)**



Comparing memory sizes

- We traded a 64-bit non-linear state with a 64-bit public tweak
- The proposed method **saves 128 bits with TI**



Hardware performance comparison w/ 3-share TI

- Smaller circuit area compared with the state-of-the-art: SAEB with GIFT-128
- Advantage over sponge-based schemes
  - Key/tweak use the smaller number of shares

Ref.	Scheme	Circuit Area /GE
This work	PFB/Skinny-64	**5,858**
This work	SAEB/GIFT-128	**6,229**
Groß et al.*	Ascon w/o IF	7,970
Groß et al.*	Ascon w IF	9,190
Arribas et al.**	Ketje-JR	18,335

Proposed method

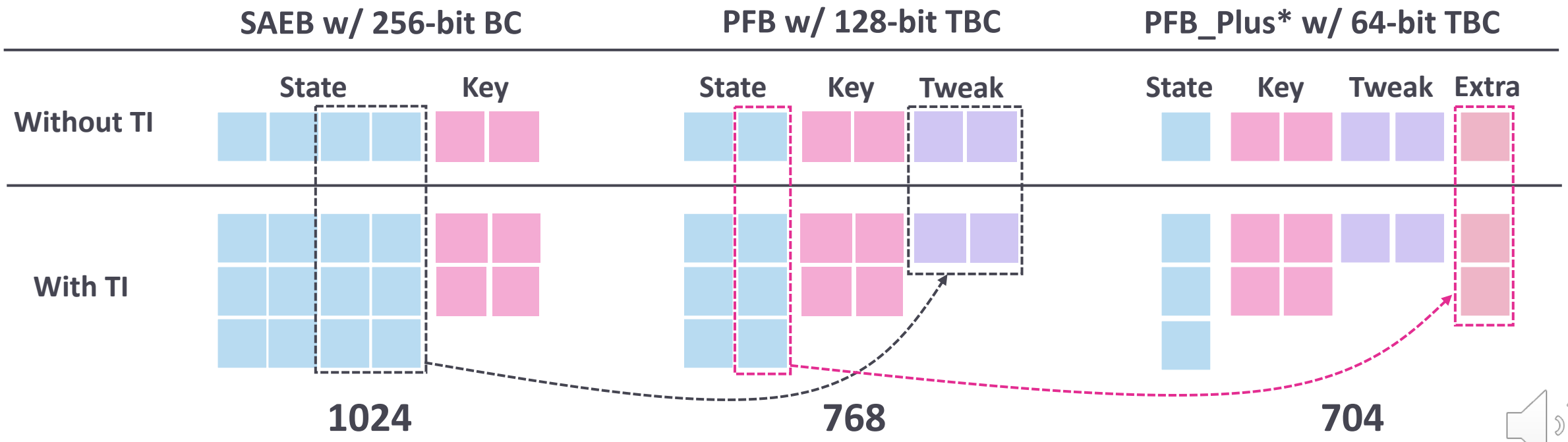
A 128-bit block cipher-based scheme implemented with the same design policy

Previous AE implementations with TI

[1] Groß et al., "Suit up! - Made-to-Measure Hardware Implementations of ASCON," *DSD 2015*.
[2] Arribas et al., "Guards in Action: First- Order SCA Secure Implementations of Ketje Without Additional Randomness," *DSD 2018*.

Further improvement for 128-bit security*

- Further reducing the non-linearly updated state
- PFB_Plus that satisfy 2b-bit security for the b-bit blockcipher
  - 128-bit security with a 64-bit TBC; even more efficient with TI



*Y. Naito, Y. Sasaki, and T. Sugawara, "Lightweight Authenticated Encryption Mode Suitable for Threshold Implementation," EUROCRYPT 2020

Conclusion

- PFB: plaintext feedback mode
  - Provides the beyond-the-birthday bound security, i.e., 64-bit security with a 64-bit primitive
  - **Low memory usage with threshold implementation (TI)**
  - Achieves the smallest circuit area in hardware implementation

- TI-friendly mode of operation
  - Further improvement: PFB_Plus
- The heterogeneity between state/key/tweak (cf. homogeneity in permutation-based schemes) leads to a better performance with TI

Thank you for watching!

Questions?