NTT

Coupon Collector's Problem for Fault Analysis against AES

High Tolerance for Noisy Fault Injections

Yu Sasaki[1], Yang Li[2], Hikaru Sakamoto[2], and Kazuo Sakiyama[2]

1: NTT Corporation

2: University of Electro-Communications

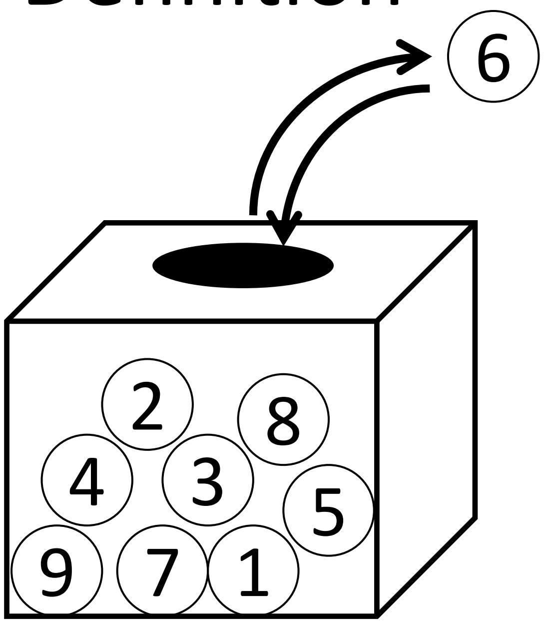
Research Summary

- Improve a side-channel analysis (SQUARE fault analysis) against AES.
- A key is recovered even if undesired fault injection (noise) occur with some probability.
- The attack is evaluated with coupon collector's problem.

Ref.	#desired fault	#noise	complexity
[PY06]	256	**0**	2^{37}
[K10]	44	**0**	2^{34}
Ours	256	**1610**	2^{45}
	128	**49**	2^{41}

Coupon Collector's Problem (CCP)

- Definition



For each coupon drawing event, 1 random coupon is obtained.

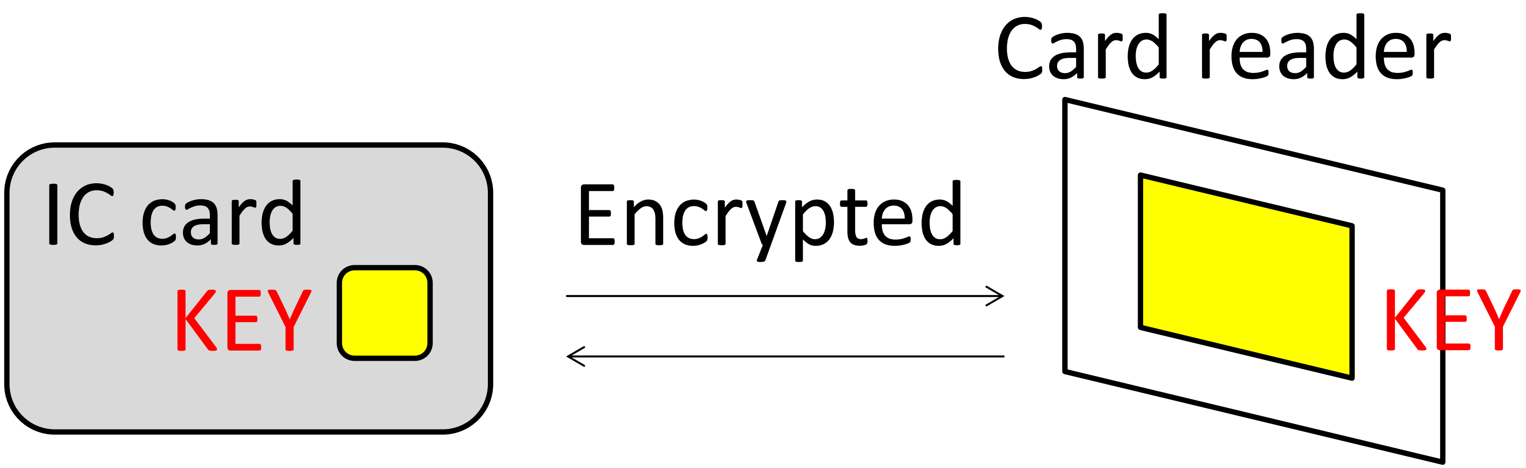
How many events are expected to complete all coupons?

$$n \ln(n)$$

- CCP can be applied to the fault attack.

Symmetric-key Encryption in Practice

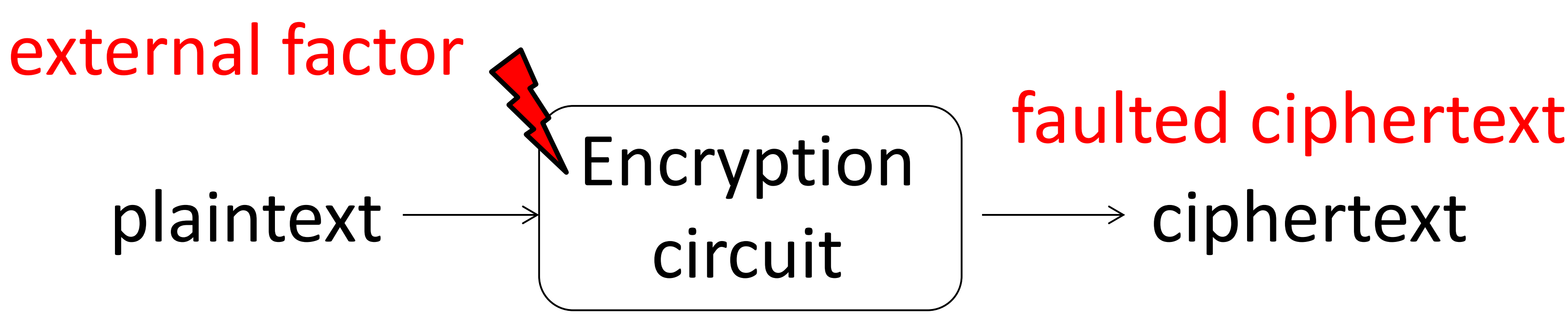
- Symmetric-key encryption is widely used to protect the communication.



- AES is the most popular algorithm.
- Its implementation needs to be protected.

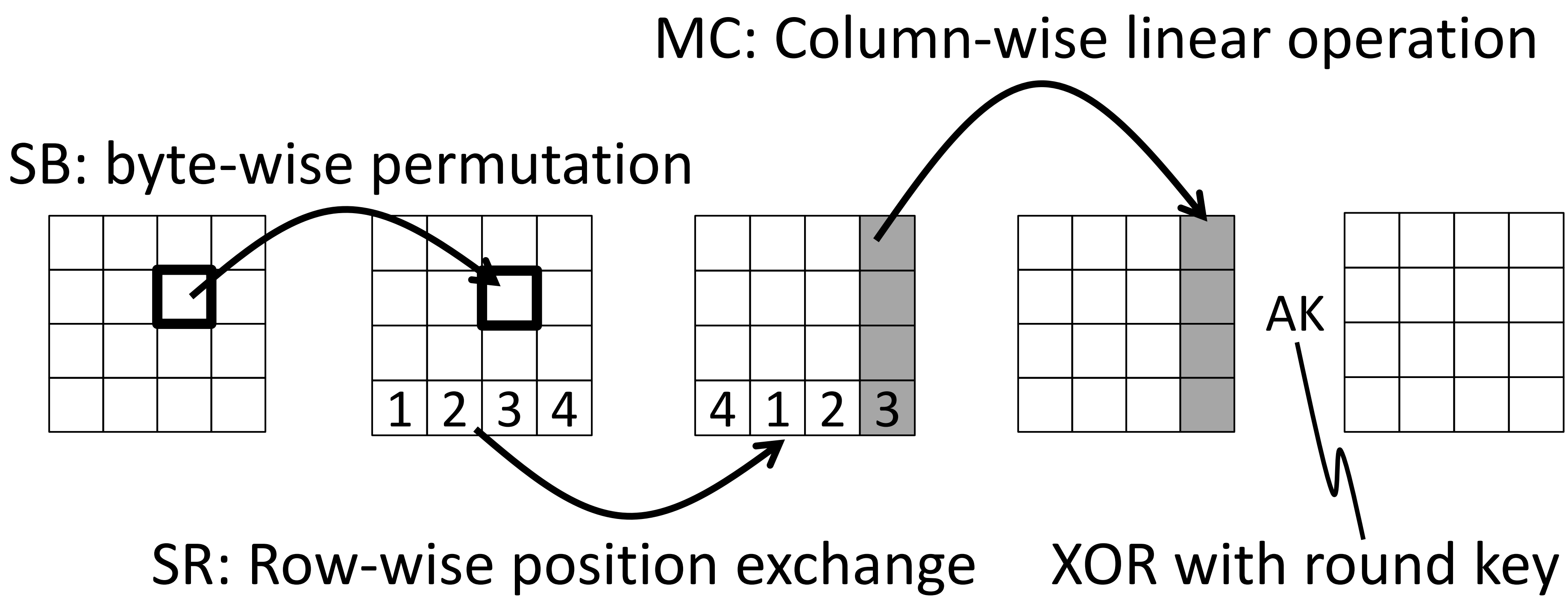
Fault Attack

- A kind of side-channel analysis.
- Give some external factor during the encryption computation to make some error.
  - Laser irradiation: give extra energy to flip internal state bits.
  - Clock glitch: force to start the next computation before the previous computation is finished.

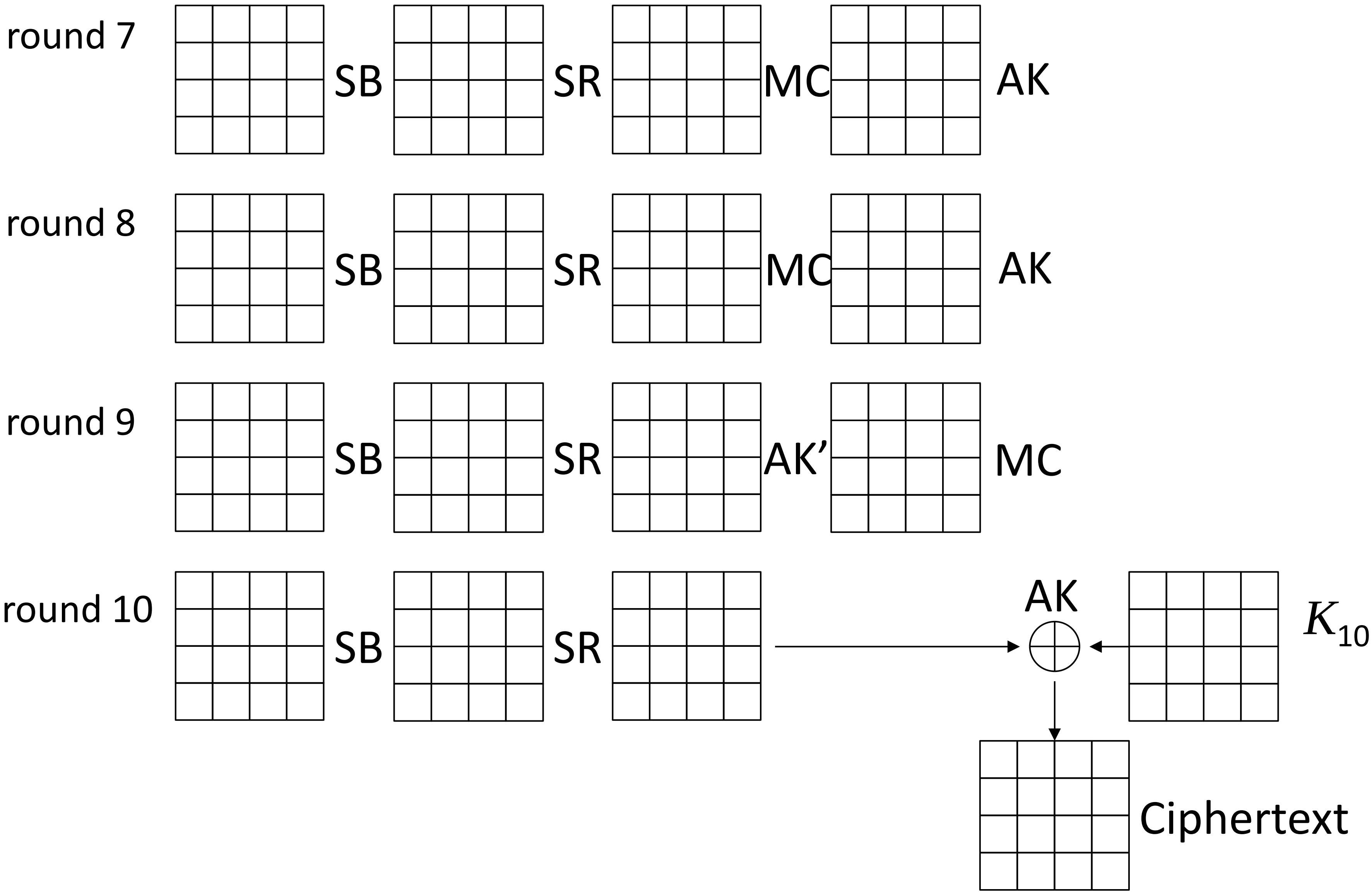


AES

- 128-bit block-cipher
- Standardized and used all over the world
- Mix 16-byte data with 10 rounds
- Computations in each round is as follows.



The Last 4 Rounds of AES

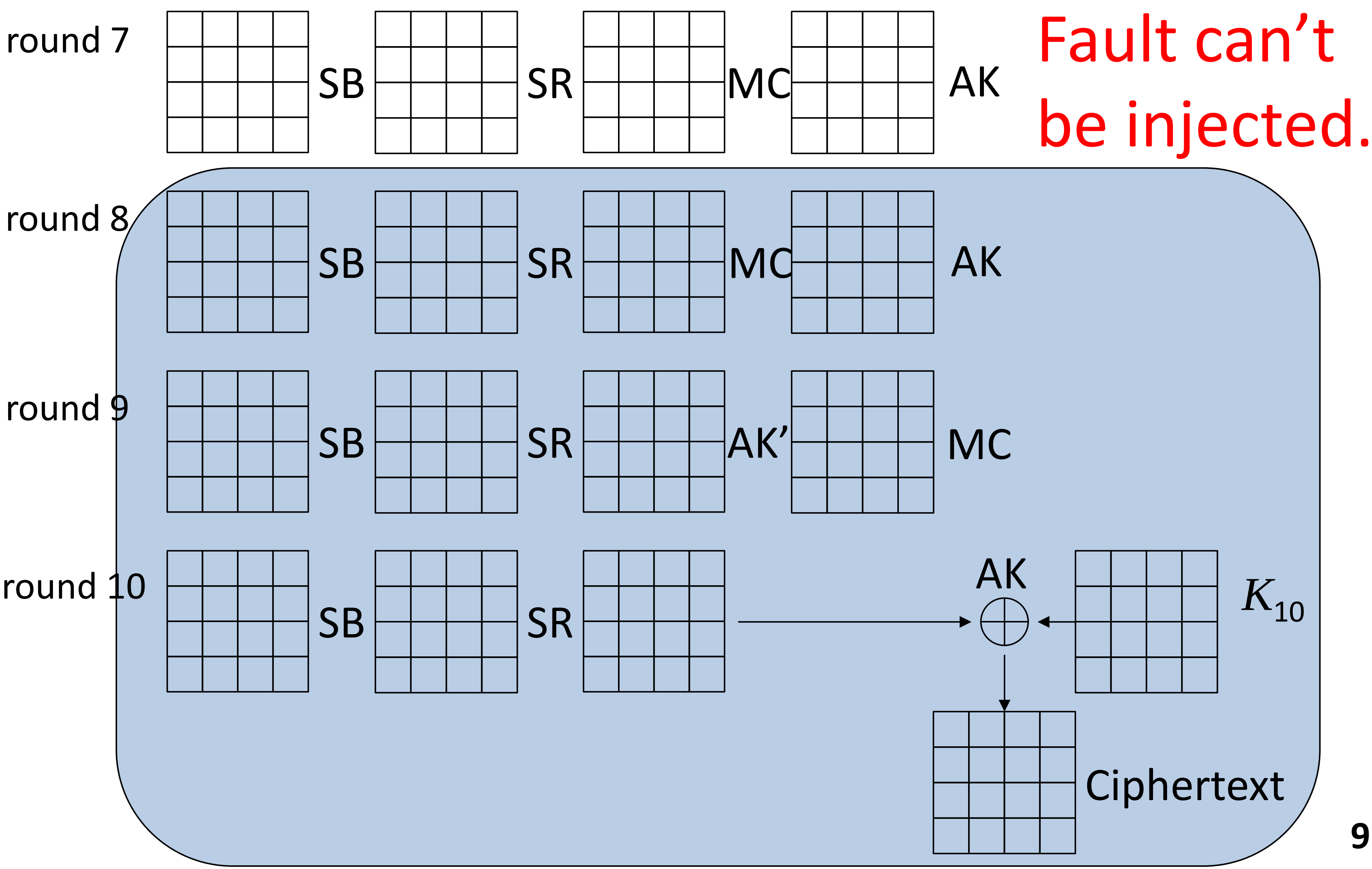


NTT

DFA and Its Countermeasure

- Differential fault analysis (DFA) is famous as a very powerful attack.
- If a fault is injected during the last 3 rounds of AES, the key is recovered easily.
- Countermeasures against fault analysis are expensive (overhead is 200%).
- It’s natural to minimize the location to be protected: only the last 3 rounds.

The Last 4 Rounds of AES



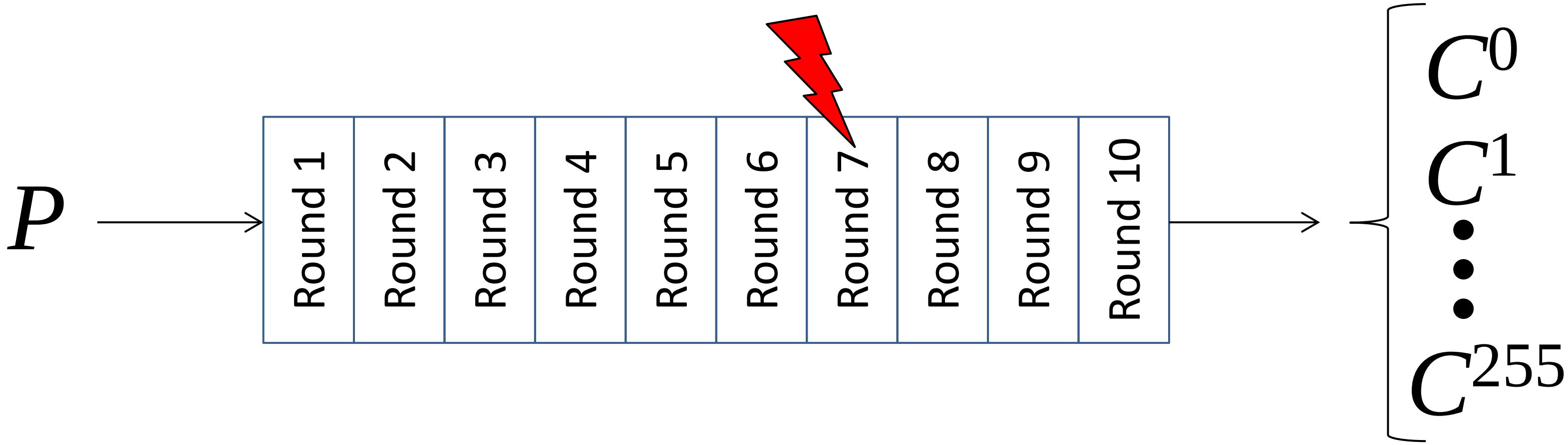
Research Motivation

- Phan and Yin showed that the key is recovered even with the fault in round 7.
- Do we need to protect round 7 as well?
- Unfortunately, their attack assumption (fault model) is very strong.

In this research, we relax the assumption!!

SQUARE DFA [PhanYin06]

- While the same plaintext is encrypted 256 times, a byte in round 7 is forced to take all 256 values by using the fault.

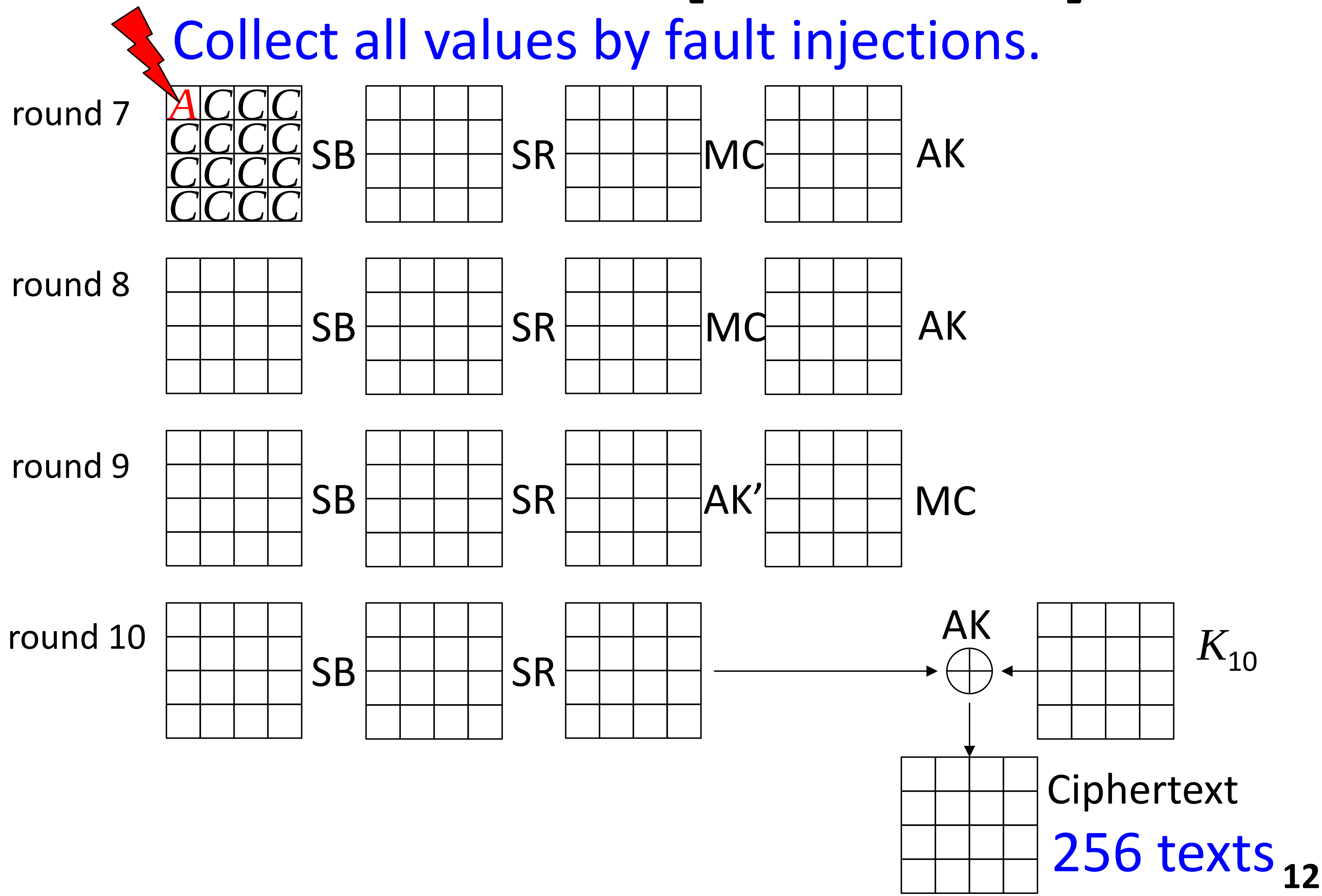


Fault model:

- *The attacker can flip any bit*
- *Undesired fault (noise) never occurs*

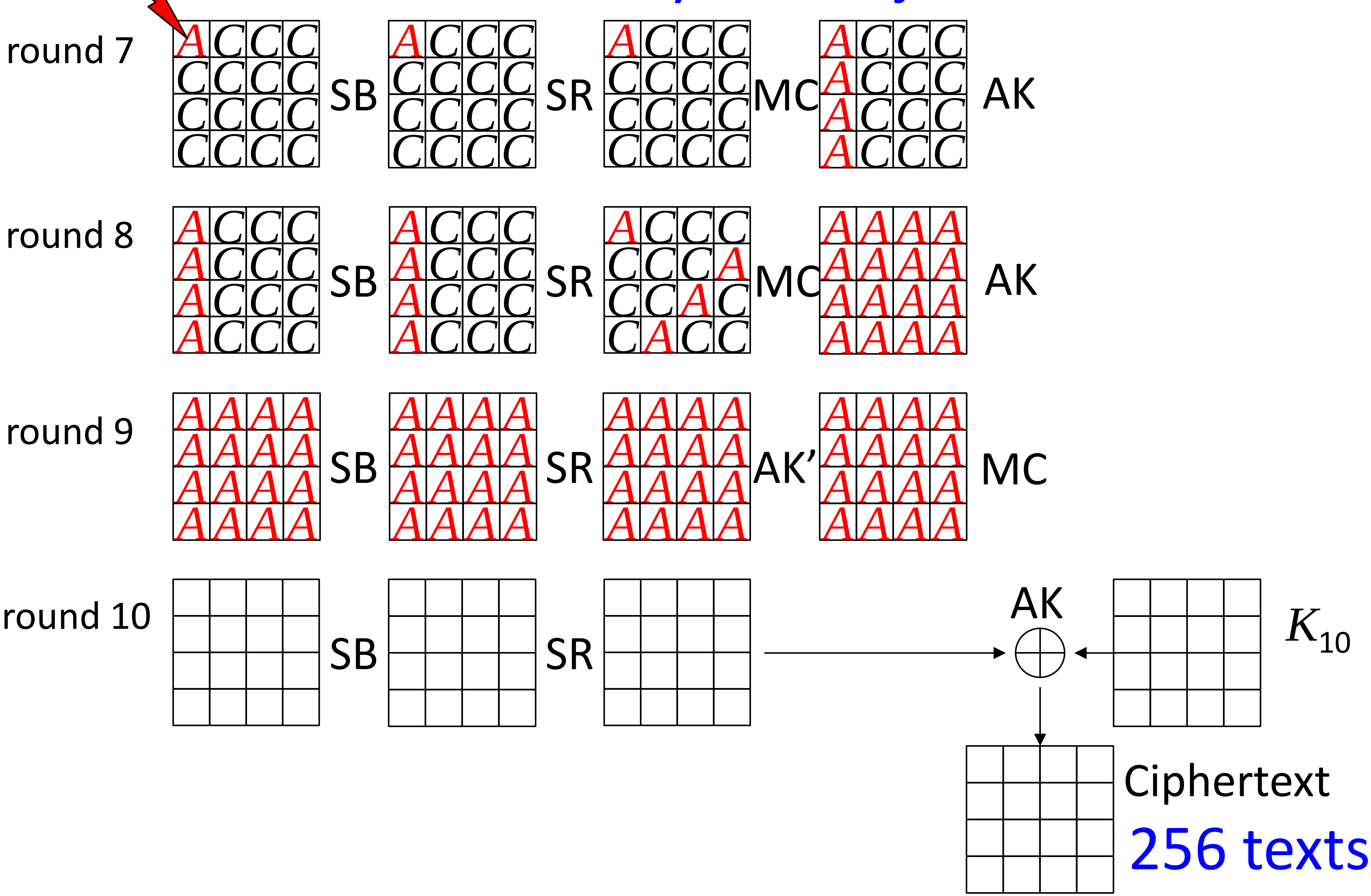
SQUARE DFA [PhanYin06]

Collect all values by fault injections.

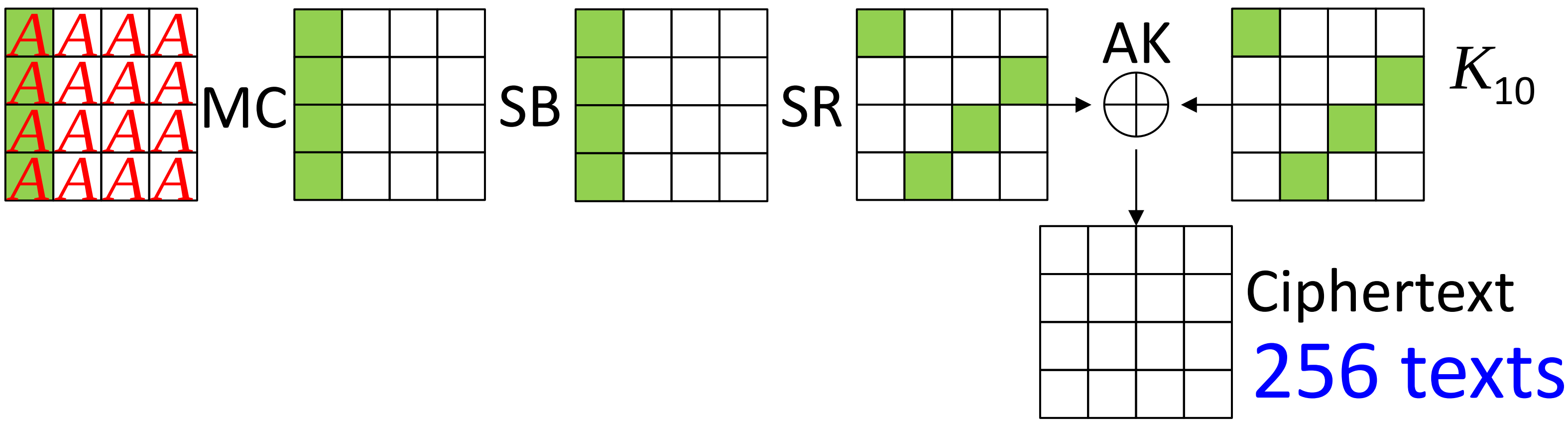


SQUARE DFA [PhanYin06]

Collect all values by fault injections.



SQUARE DFA [PhanYin06]

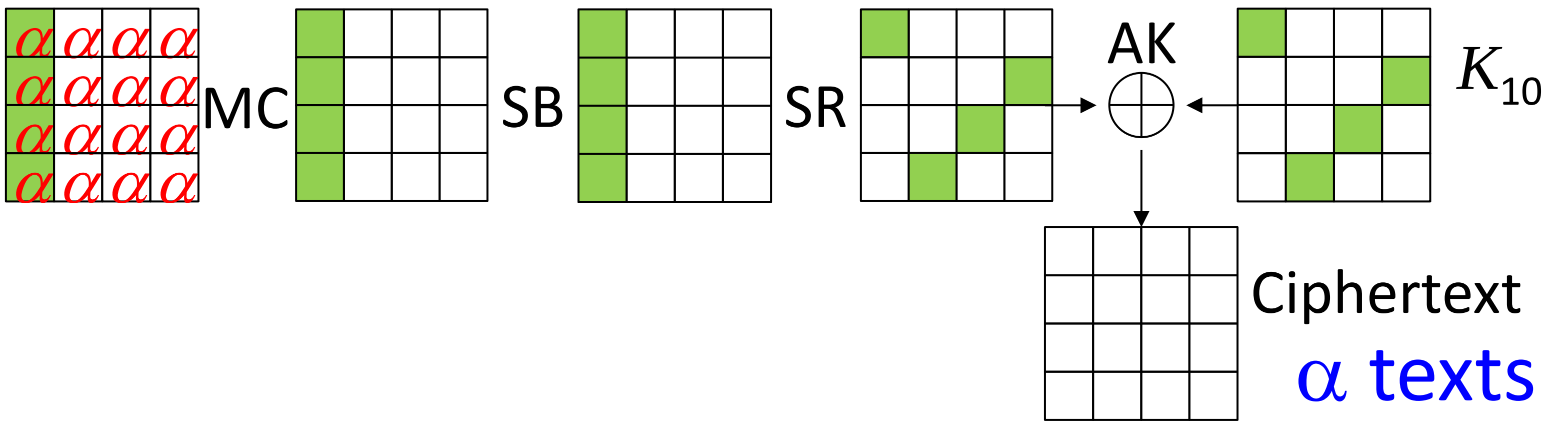


The key K_{10} is guessed column by column.

If the guess is correct, each byte takes all 256 distinct values after the 1 round decryption.

Probability: $$\left(\prod_{i=0}^{255} \frac{(256-i)}{256}\right)^4$$

Improved SQUARE DFA [Kim11]



256 values are not necessary. α values are enough.

For the correct guess, each byte takes α values.

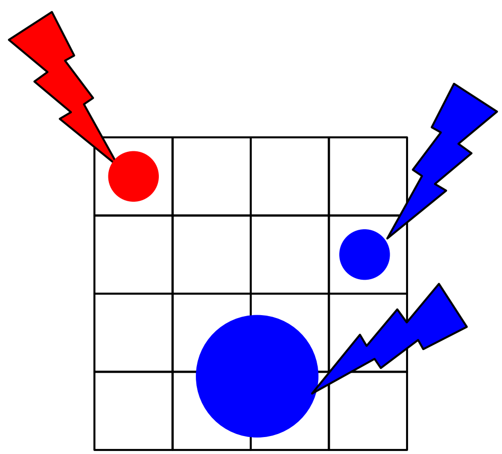
Probability: $$\left(\prod_{i=0}^{\alpha-1}\frac{(256-i)}{256}\right)^4$$

The probability is smaller than 2^{-32} for α = 44.

Noisy Fault Model

- Previous SQUARE DFAs assume that unintended fault never occurs.
- But, in practice, noise is obtained.

intended fault

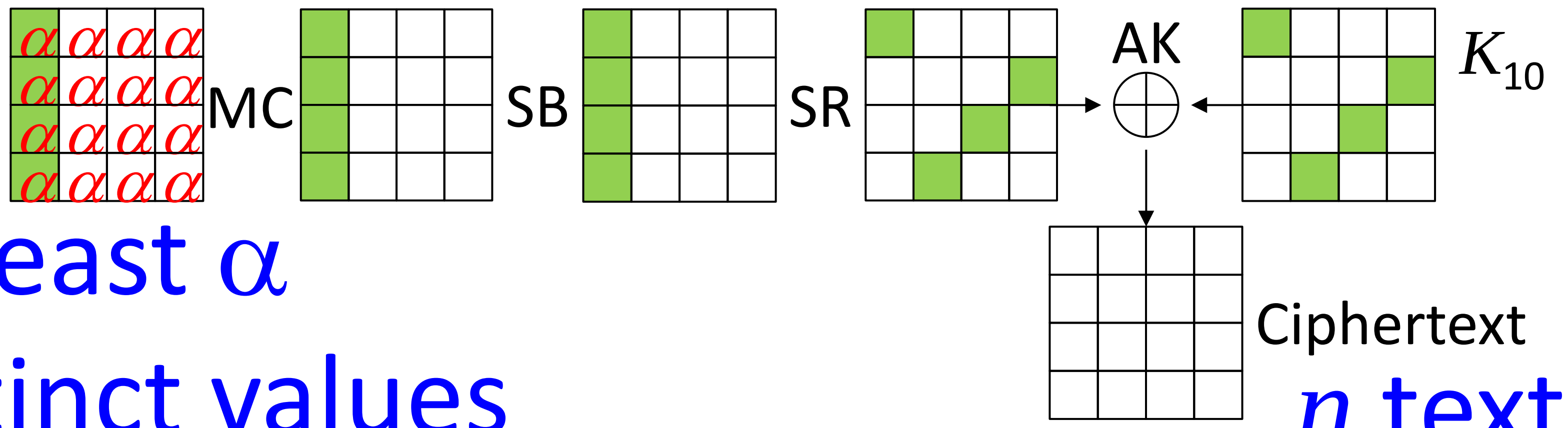


different byte

multi-bytes

➡ We can still recover the key !!

Our Attacks



At least α distinct values

n texts

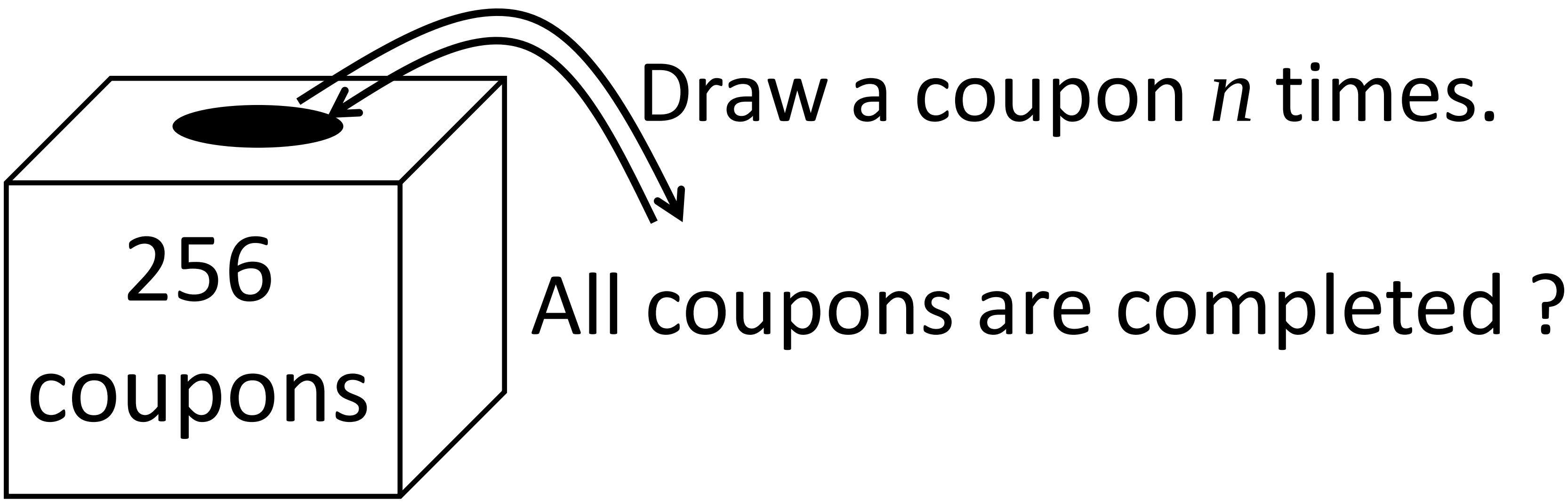
- α: the number of distinct fault values
- n: the total number of texts to be analyzed

For the correct guess at least α distinct values appear, otherwise, the guess is wrong.

What's the probability?

Probability Estimation with CCP

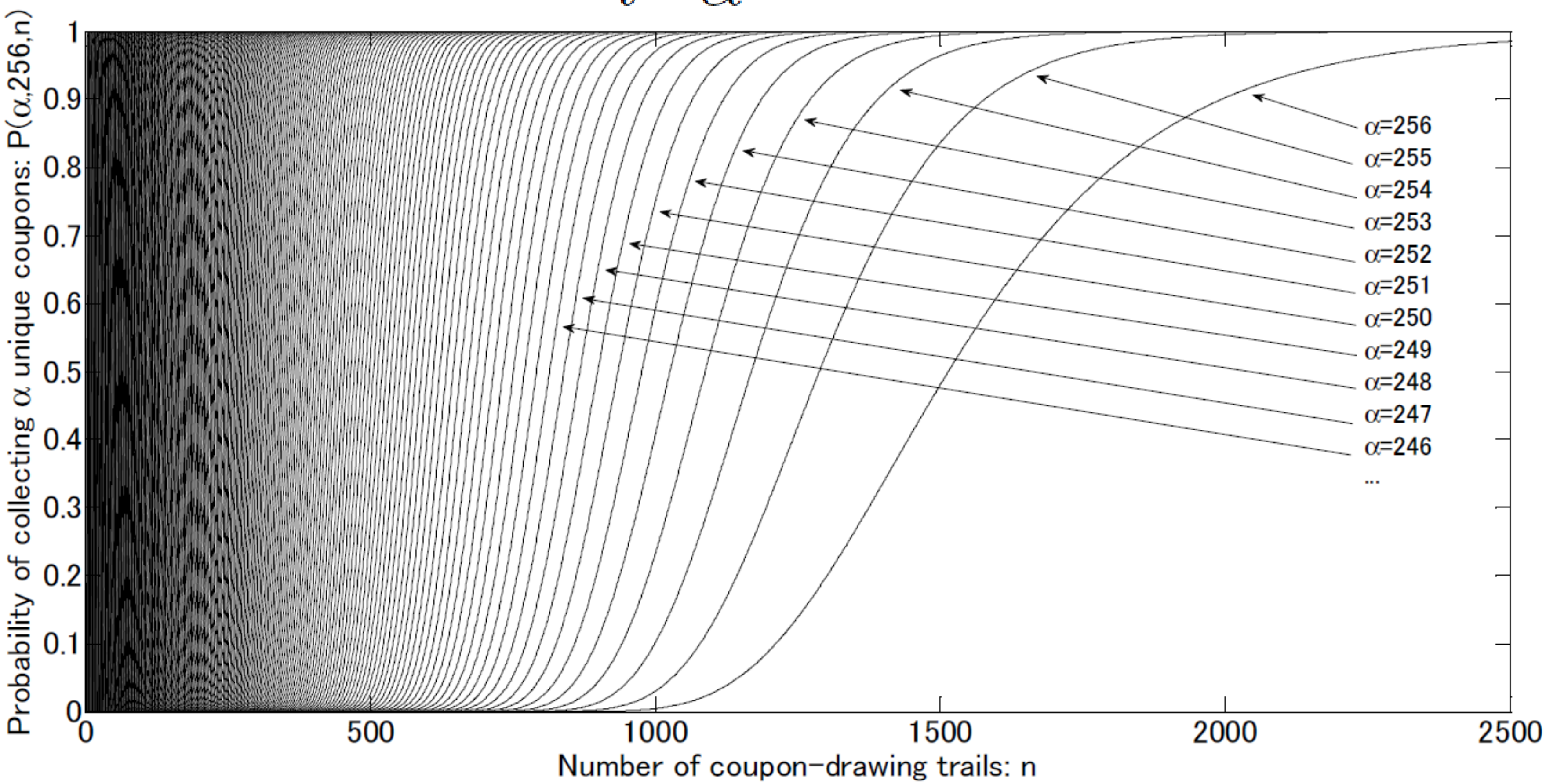
- Suppose that α = 256. Each guess is a right key candidate if all 256 values are completed after *n* trials.



- equivalent to the CCP. $Pr=2^{-1}$ even if n=1553.
- For α < 256, it becomes a variant of the CCP.

Probability Estimation with CCP

$$\binom{\beta}{\alpha}\binom{\alpha}{1}\sum_{i=\alpha}^{n}\frac{Q(\alpha-1,i-1)}{\beta^{i}}$$



Example Parameters

Value of n

	$\alpha = 64$	$\alpha = 128$	$\alpha = 256$
$P(\alpha, 256, n)^4 = 2^{-1}$	77	186	1866
$P(\alpha, 256, n)^4 = 2^{-4}$	73	177	1553
$P(\alpha, 256, n)^4 = 2^{-32}$	66	156	933

Conclusion

- We generalized the SQUARE DFA so that the noisy fault injection can be accepted.
- We did the probability estimation with the coupon collector's problem.
- Possible future direction
  - Detect a suitable fault injection method.
  - Evaluate other ciphers.

Thank you for your attention !!