密 码 学

第十章 密码学的新方向

网络空间安全学院
胡 伟　朱丹
weihu/zhudan@nwpu.edu.cn

章节安排

Outline

AES能量侧信道分析

AES能量侧信道防护

AES故障注入攻击

章节安排

Outline

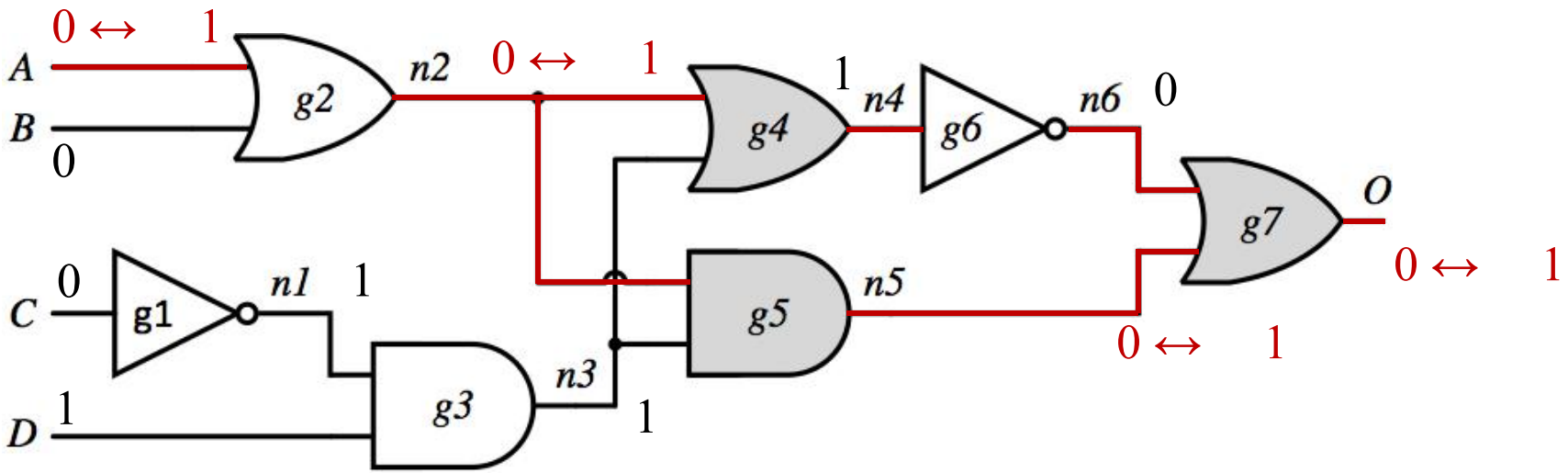
AES能量侧信道分析

AES能量侧信道防护

AES故障注入攻击

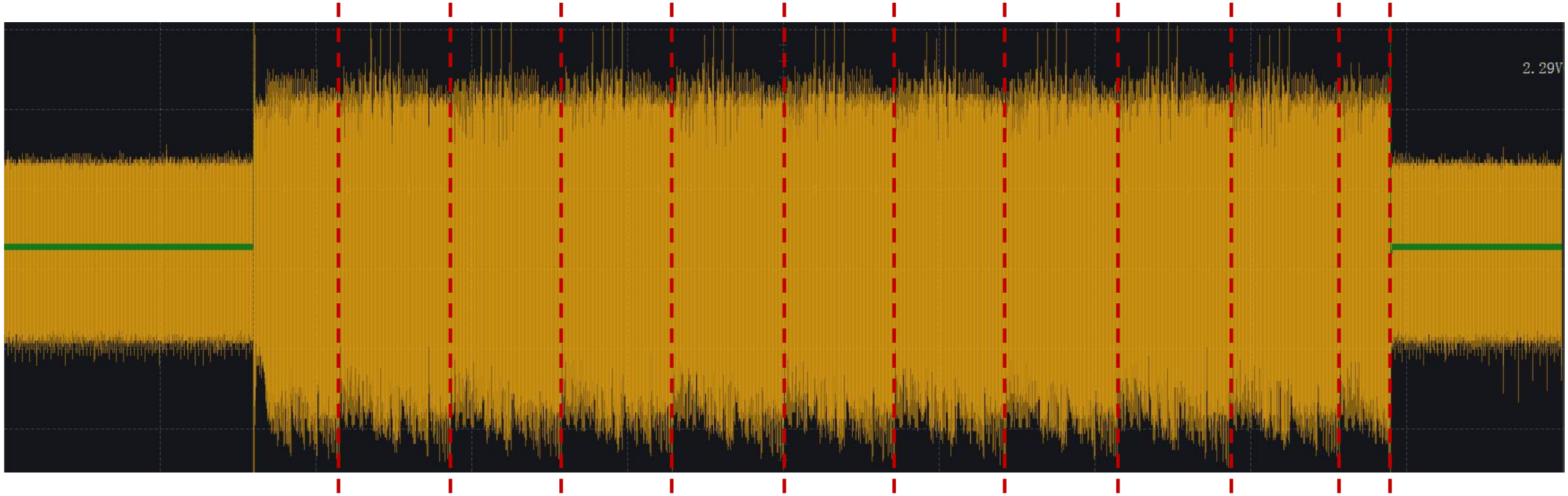
10.1(1) 能量侧信道产生原理

- 电路的翻转行为产生能耗



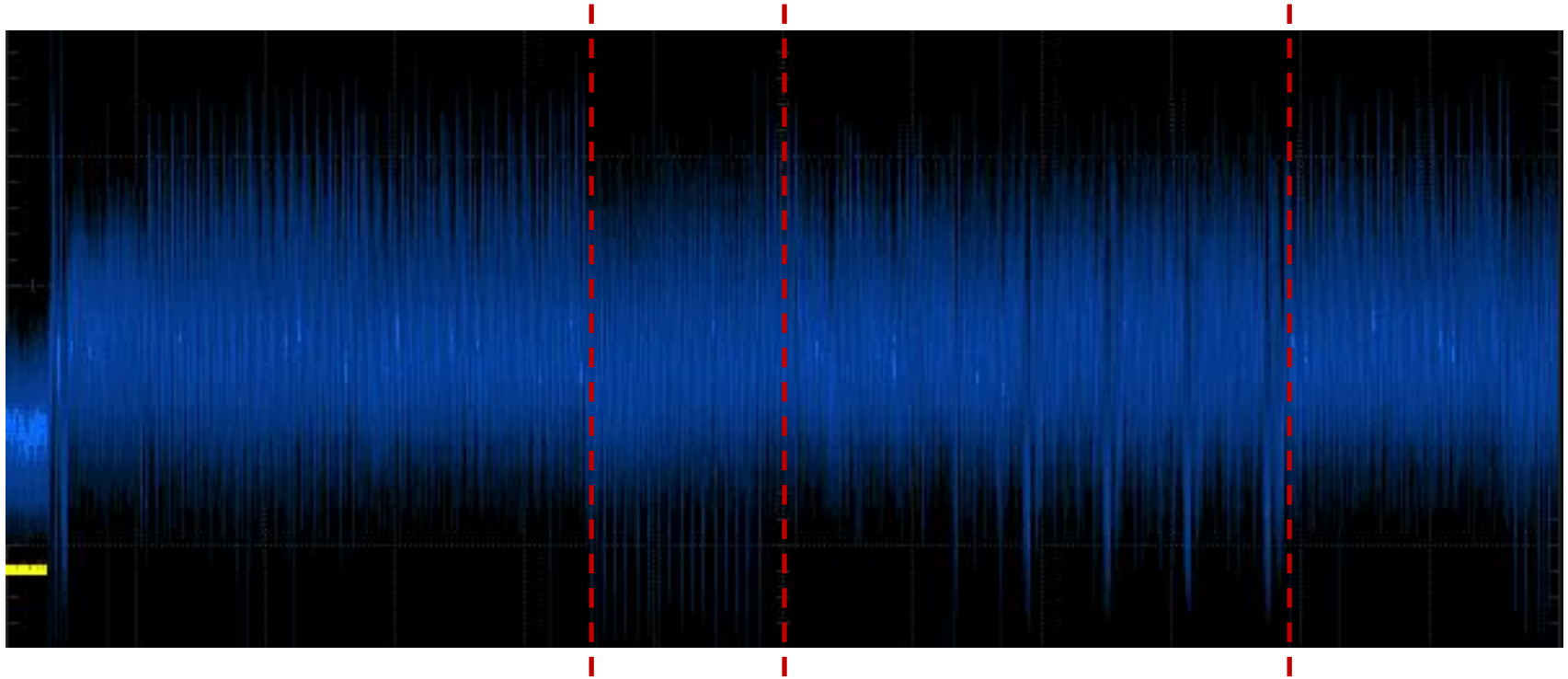
10.1(1) 能量侧信道产生原理

- AES-128加密能量迹



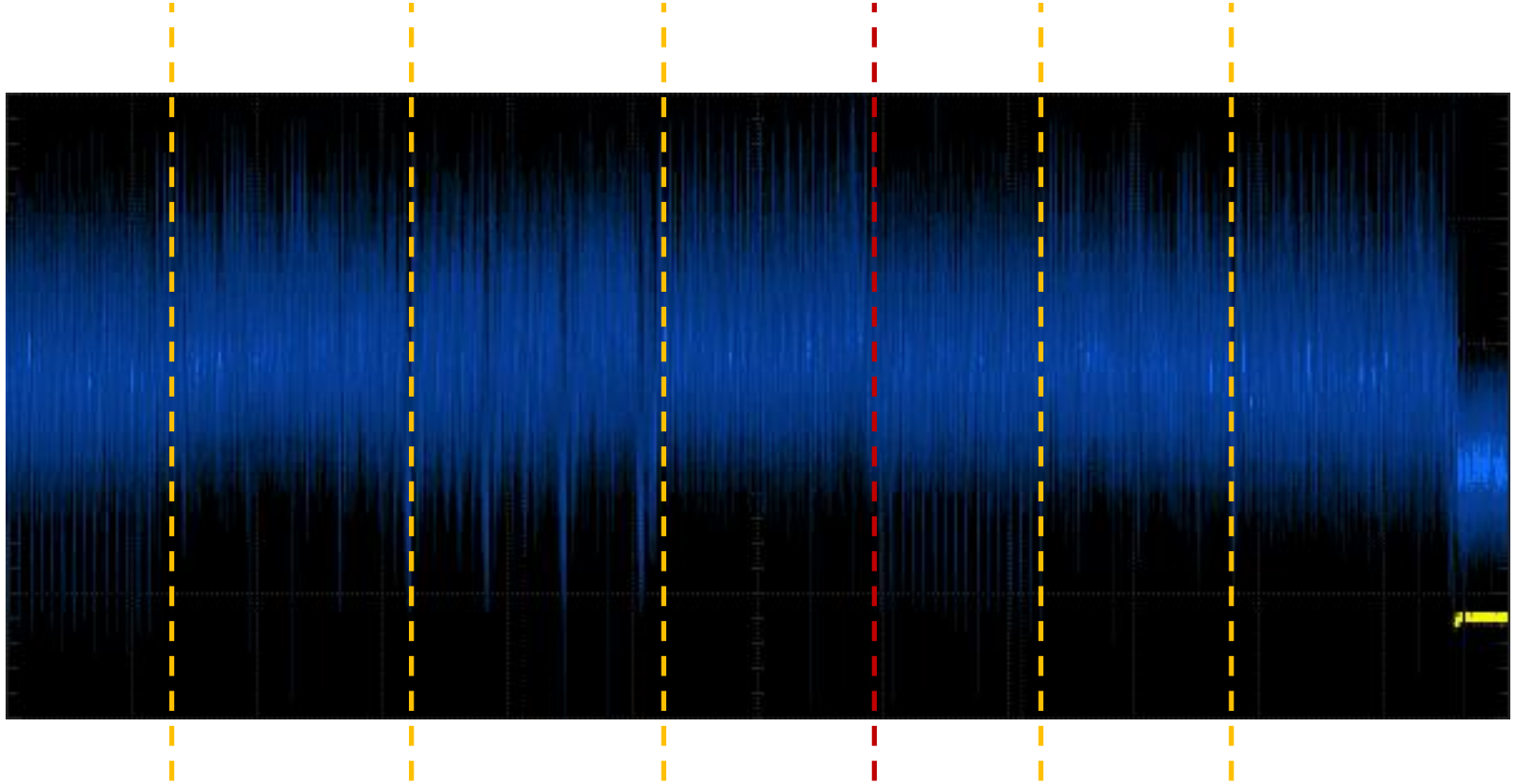
10.1(1) 能量侧信道产生原理

- AES加密能量迹（第一轮）



10.1(1) 能量侧信道产生原理

- AES加密能量迹（最后两轮）



10.1(2) 能量模型

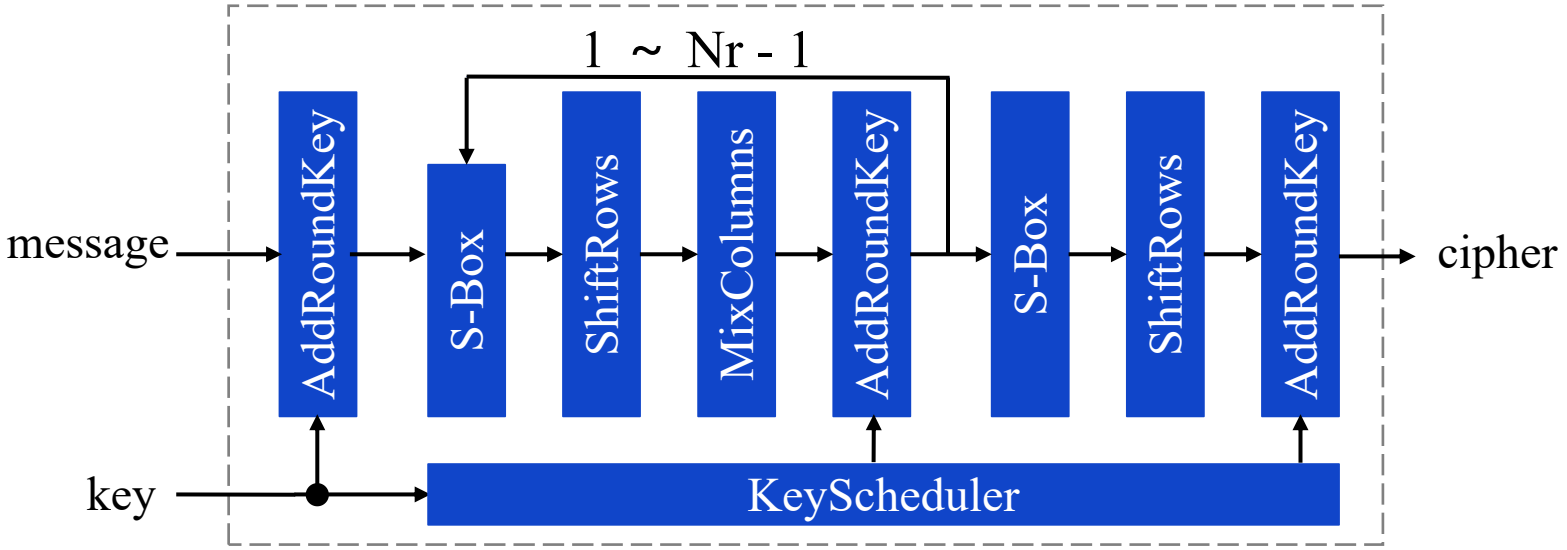
- 汉明重量(Hamming Weight)模型
  - 假设X是n比特的变量
  - X的汉明重量定义为$w(X) = \sum x[i],\ 1 \le i \le n$
- 汉明距离(Hamming Distance)模型
  - 假设X和Y都是n比特的变量
  - X和Y的汉明距离定义为$d(X, Y) = \sum x[i] \oplus y[i],\ 1 \le i \le n$

如何选用模型?

例，X = 1001，Y= 1100。分别计算X和Y的汉明重量以及X和Y的汉明距离

10.1(3) AES能量侧信道产生原理

- 1: 密钥流向了泄漏负载函数
- 2: 泄露负载函数通常为非线性环节
- 3: 泄漏负载函数通常实现混淆功能



10.1(4) 能量侧信道分析类型

Types of Power Side Channel Analysis

01

SPA

简单能量分析

02

DPA

差分能量分析

03

CPA

相关能量分析

04

TPA

模板能量分析

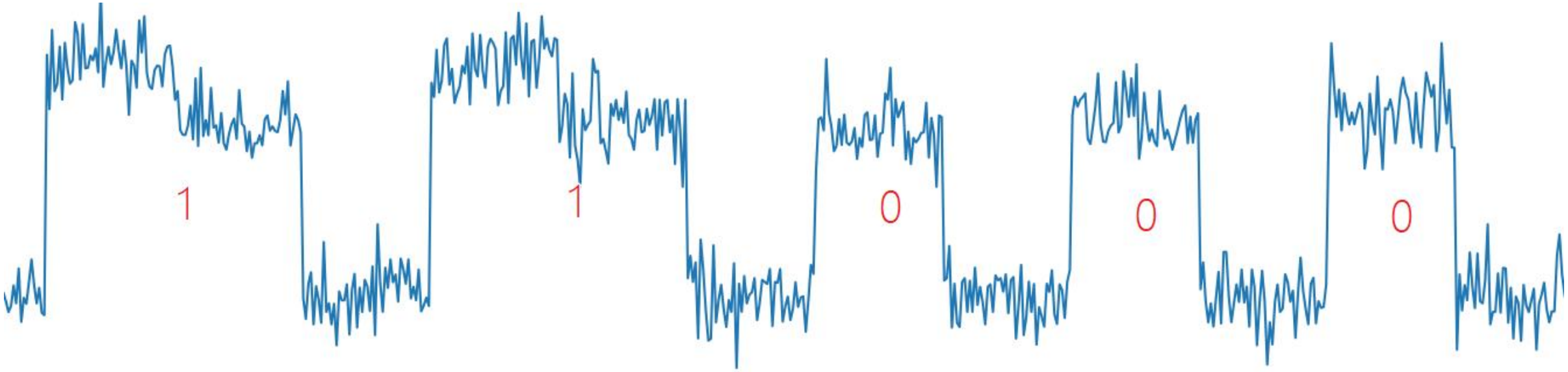
05

AI-based PA

基于人工智能的能量分析

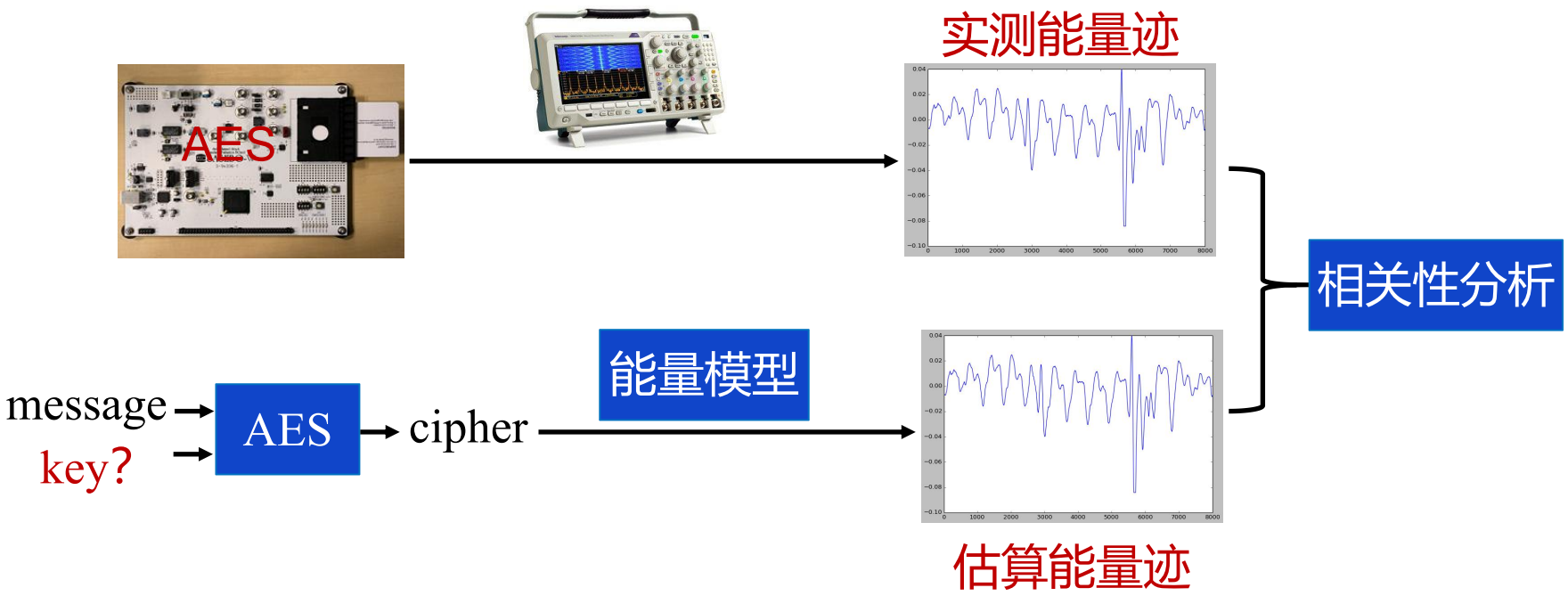
10.1(4) 简单能量分析

- 直接从能量迹能够分析出敏感信息
  - RSA是典型例子
  - 密钥位为0和1能耗存在差别
  - 直接反映在能量迹波形的形态上



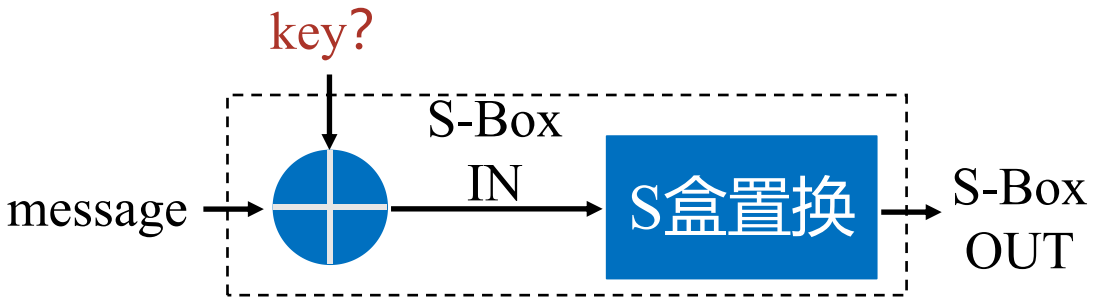
10.1(4) 相关能量分析

- AES相关能量侧信道分析
  - 能量迹的采集
  - 理论能量迹的估算
  - 相关性分析
  - 密钥恢复

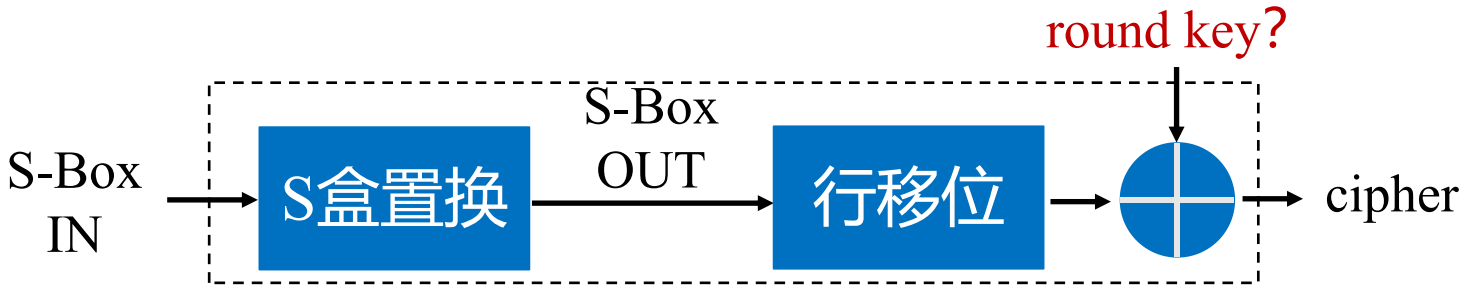


10.1(4) AES相关能量分析

- 攻击第一轮



- 攻击最后一轮



10.1(4) AES相关能量分析

- 攻击步骤（以攻击第一轮为例）
  - S1: 加密N（约10000）条明文并用示波器采集能量迹（Power trace）
  - S2: 对这N条明文，对于每个密钥字节的可能取值key[i] ∈ [0，255]，1 ≤ i ≤ 16，分别计算得到N个S-Box IN和S-Box OUT的值

$$T = \begin{bmatrix} t_{11} & \cdots & t_{1k} \\ \vdots & \ddots & \vdots \\ t_{N1} & \cdots & t_{Nk} \end{bmatrix}$$

能量迹

$$SI = \begin{bmatrix} si_{1,0} & \cdots & si_{1,255} \\ \vdots & \ddots & \vdots \\ si_{N,0} & \cdots & si_{N,255} \end{bmatrix}$$

S-Box IN

$$SO = \begin{bmatrix} so_{1,0} & \cdots & so_{1,255} \\ \vdots & \ddots & \vdots \\ so_{N,0} & \cdots & so_{N,255} \end{bmatrix}$$

S-Box OUT

10.1(4) AES相关能量分析

- 攻击步骤（以攻击第一轮为例）
  - S3: 计算SO的汉明重量矩阵
  - S3: 或者计算SI和SO的汉明距离矩阵

$$w(SO) = \begin{bmatrix} w_{1,0} & \cdots & w_{1,255} \\ \vdots & \ddots & \vdots \\ w_{N,0} & \cdots & w_{N,255} \end{bmatrix}$$

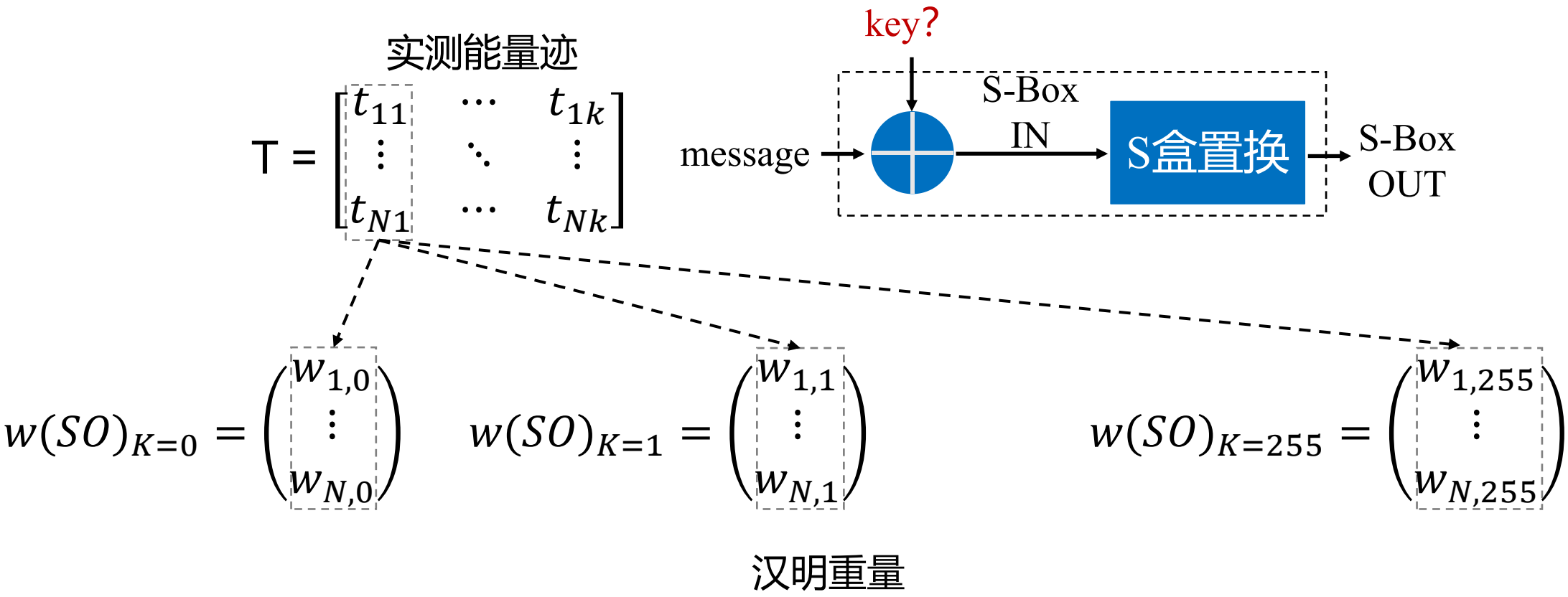
汉明重量

$$d(SI \oplus SO) = \begin{bmatrix} d_{1,0} & \cdots & d_{1,255} \\ \vdots & \ddots & \vdots \\ d_{N,0} & \cdots & d_{N,255} \end{bmatrix}$$

汉明距离

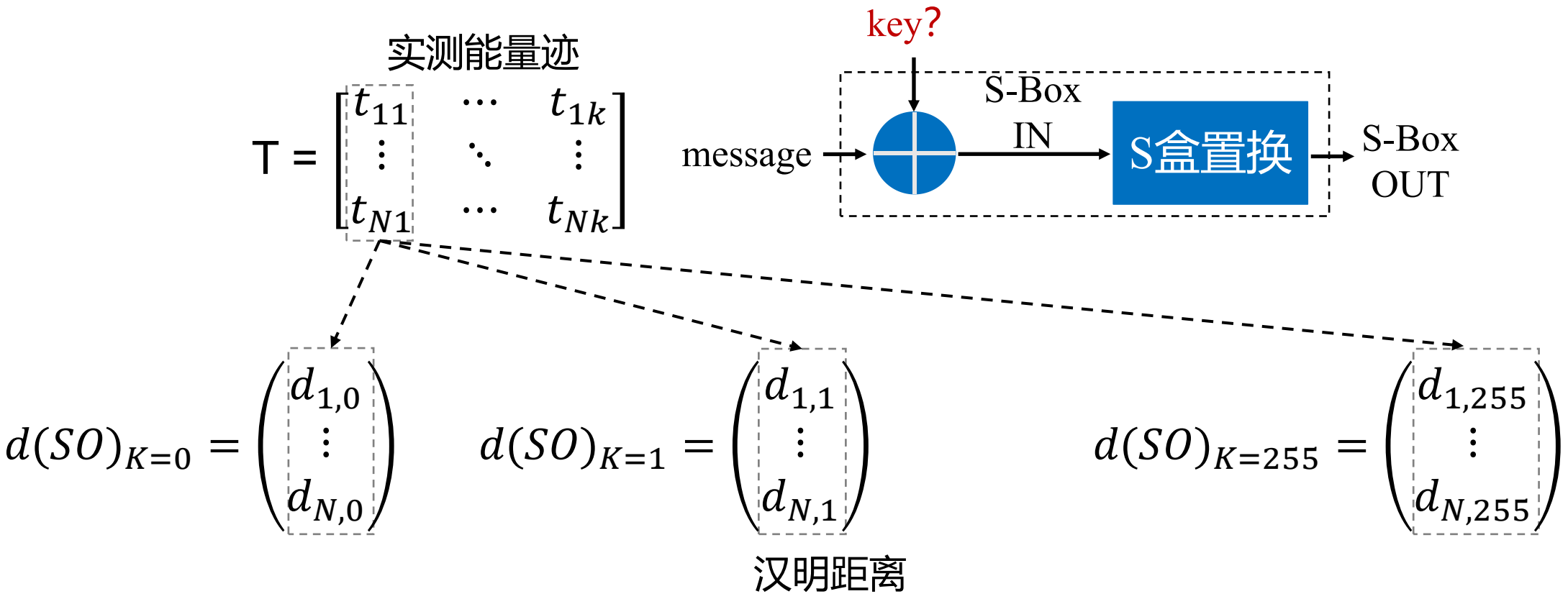
10.1(4) AES相关能量分析

- 攻击步骤（以攻击第一轮为例）
  - S4: 相关性分析(Correlation Analysis)



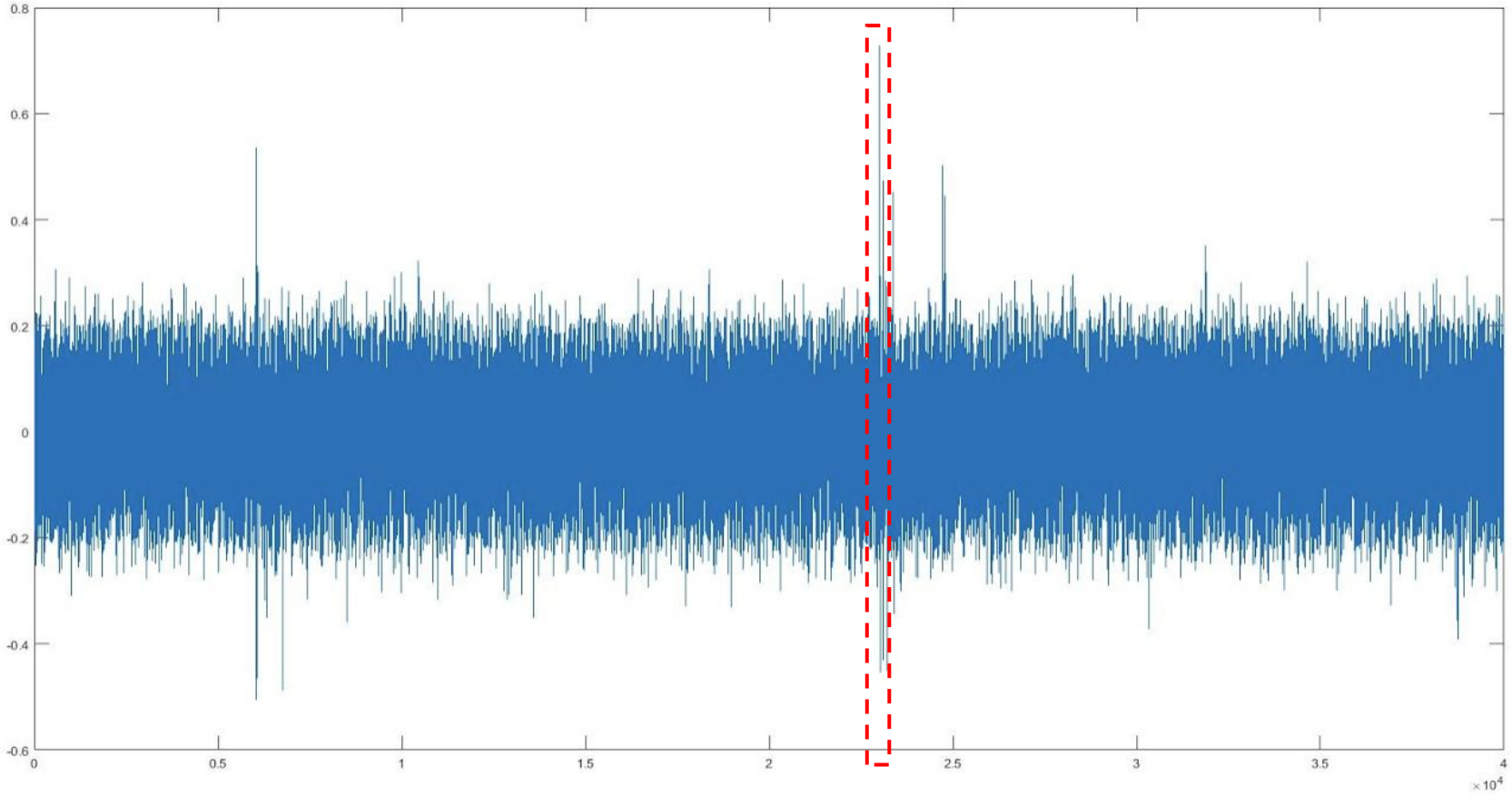
10.1(4) AES相关能量分析

- 攻击步骤（以攻击第一轮为例）
  - S4: 相关性分析(Correlation Analysis)



10.1(4) AES相关能量分析

- 攻击结果示例



章节安排

Outline

AES能量侧信道分析

AES能量侧信道防护

AES故障注入攻击

10.2 能量攻击防护措施

- AES能量侧信道攻击防护
  - 随机化（Randomization）和噪声（Random noise）
  - 掩码算法（Masking）
  - 计算闪烁（Blinking）
  - 定态逻辑（WDDL）

- 随机化（Randomization）
  - 多条明文执行顺序的随机化，交叉执行
  - 可并行环节执行顺序的随机化，如16个S-Box执行顺序
- 噪声（Random noise）
  - 用普通计算的能量噪声掩盖加密运算，如并行计算技术
  - 例如，GPU上的能量侧信道分析比较困难

- 掩码算法（Masking）

Algorithm 1: AES-256 used for the DPA contest v4 [TEL14].

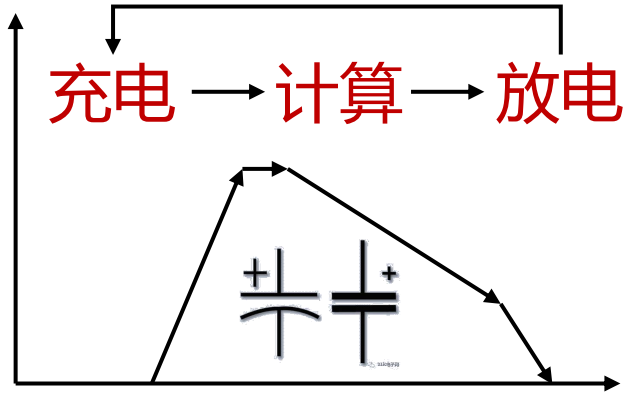
Input : Plaintext X, seen as 16 bytes X_i, $i \in [\![0, 15]\!]$,
Key schedule, 15 128-bit constants RoundKey$[r]$, $r \in [\![0, 14]\!]$
Output: Ciphertext X, seen as 16 bytes X_i, $i \in [\![0, 15]\!]$

Draw a random offset, uniformly in $[\![0, 15]\!]$
$X = X \oplus \text{Mask}_{\text{offset}}$ /* Plaintext blinding */
/* All rounds but the last one */
for $r \in [\![0, 12]\!]$ **do**
  $X = X \oplus \text{RoundKey}[r]$ /* AddRoundKey */
  for $i \in [\![0, 15]\!]$ **do**
    $X_i = \text{MaskedSubBytes}_{\text{offset}+i+r}(X_i)$
  end
  $X = \text{ShiftRows}(X)$
  $X = \text{MixColumns}(X)$
  $X = X \oplus \text{MaskCompensation}_{\text{offset}+1+r}$
end
/* Last round */
$X = X \oplus \text{RoundKey}[13]$
for $i \in [\![0, 15]\!]$ **do**
  $X_i = \text{MaskedSubBytes}_{\text{offset}+13+r}(X_i)$
end
$X = \text{ShiftRows}(X)$
$X = X \oplus \text{RoundKey}[14]$
/* Ciphertext demasking */
$X = X \oplus \text{MaskCompensationLastRound}_{\text{offset}+14}$

TELECOM ParisTech SEN research group. DPA Contest (4th edition), 2013-2014. http://www.DPAcontest.org/v4/.

10.2 能量攻击防护措施

- 计算闪烁（Blinking）
  - 借鉴了人眼的生物特征
  - 利用存储能量执行敏感计算
  - 无法抵御基于IR的能量分析



Ivan Martinovic, Doug Davies, Mario Frank, Daniele Perito, Tomas Ros, and Dawn Song. 2012. On the feasibility of side-channel attacks with brain-computer interfaces. In Proceedings of the 21st USENIX conference on Security symposium (Security'12). USENIX Association, USA, 34.

章节安排

Outline

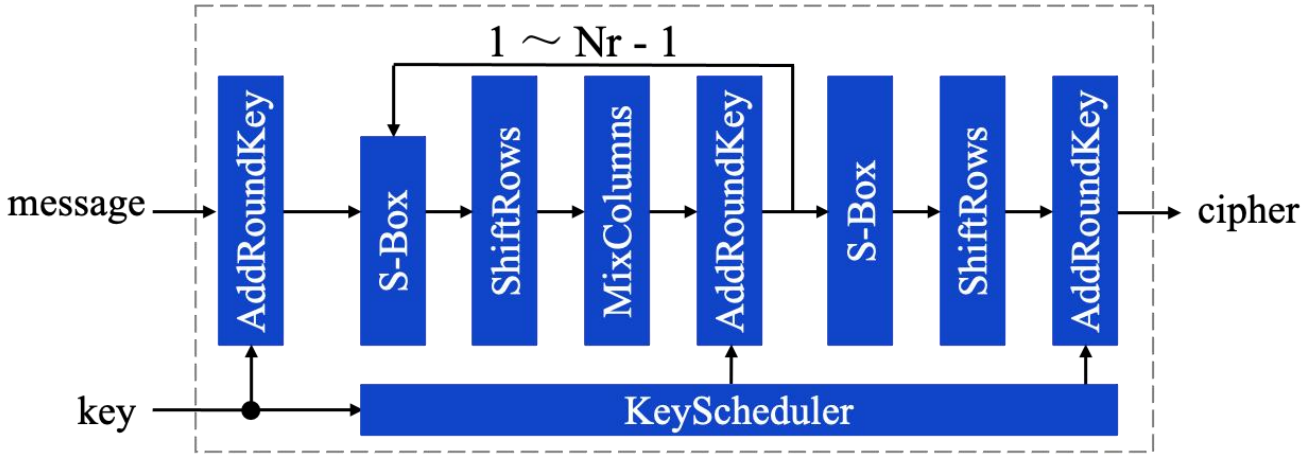
AES能量侧信道分析

AES能量侧信道防护

AES故障注入攻击

10.3(1) AES故障效应分析

- S盒变换 – S-Box （故障效应由S盒定义决定）
- 行移位 – ShiftRows （只是故障位置发生变化）
- 列混合 – MixColumns （故障从1字节扩散到4字节）
- 加轮密钥 – AddRoundKey （结果相应字节发生故障）
- 密钥扩展 – KeyScheduler（轮密钥发生错误）



10.3(1) AES故障效应分析


```
// first row
temp[0] = ptext[0];  temp[4] = ptext[4];  temp[8]  = ptext[8];   temp[12] = ptext[12];
// second row
temp[1] = ptext[5];  temp[5] = ptext[9];  temp[9]  = ptext[13];  temp[13] = ptext[1];
// third row
temp[2] = ptext[10]; temp[6] = ptext[14]; temp[10] = ptext[2];   temp[14] = ptext[6];
// fourth row
temp[3] = ptext[15]; temp[7] = ptext[3];  temp[11] = ptext[7];   temp[15] = ptext[11];
```

$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$	循环左移0位	$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$
$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	循环左移1位	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,0}$
$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	循环左移2位	$a_{2,2}$	$a_{2,3}$	$a_{2,0}$	$a_{2,1}$
$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	循环左移3位	$a_{3,3}$	$a_{3,0}$	$a_{3,1}$	$a_{3,2}$

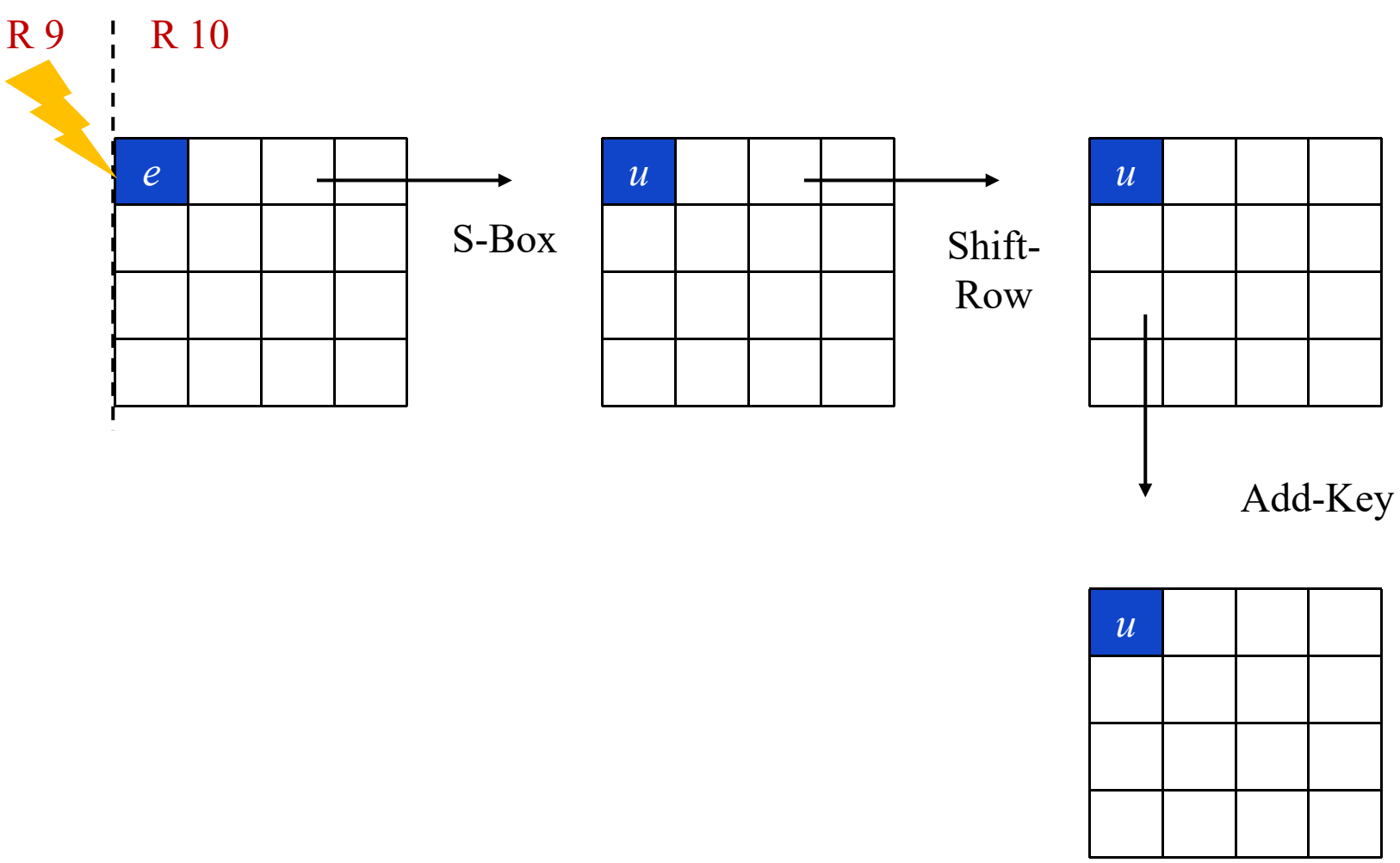
行移位 – ShiftRows （只是故障位置发生变化）

10.3(1) AES故障效应分析


```
temp[0] =  gf_mult2(ptext[0]) ^ (gf_mult2(ptext[1]) ^ ptext[1]) ^ ptext[2] ^ ptext[3];
temp[4] =  gf_mult2(ptext[4]) ^ (gf_mult2(ptext[5]) ^ ptext[5]) ^ ptext[6] ^ ptext[7];
temp[8] =  gf_mult2(ptext[8]) ^ (gf_mult2(ptext[9]) ^ ptext[9]) ^ ptext[10] ^ ptext[11];
temp[12] = gf_mult2(ptext[12]) ^ (gf_mult2(ptext[13]) ^ ptext[13]) ^ ptext[14] ^ ptext[15];

temp[1] = ptext[0] ^ gf_mult2(ptext[1]) ^ (gf_mult2(ptext[2]) ^ ptext[2]) ^ ptext[3];
temp[5] = ptext[4] ^ gf_mult2(ptext[5]) ^ (gf_mult2(ptext[6]) ^ ptext[6]) ^ ptext[7];
temp[9] = ptext[8] ^ gf_mult2(ptext[9]) ^ (gf_mult2(ptext[10]) ^ ptext[10]) ^ ptext[11];
temp[13] = ptext[12] ^ gf_mult2(ptext[13]) ^ (gf_mult2(ptext[14]) ^ ptext[14]) ^ ptext[15];

temp[2] = ptext[0] ^ ptext[1] ^ gf_mult2(ptext[2]) ^ (gf_mult2(ptext[3]) ^ ptext[3]);
temp[6] = ptext[4] ^ ptext[5] ^ gf_mult2(ptext[6]) ^ (gf_mult2(ptext[7]) ^ ptext[7]);
temp[10] = ptext[8] ^ ptext[9] ^ gf_mult2(ptext[10]) ^ (gf_mult2(ptext[11]) ^ ptext[11]);
temp[14] = ptext[12] ^ ptext[13] ^ gf_mult2(ptext[14]) ^ (gf_mult2(ptext[15]) ^ ptext[15]);

temp[3] = (gf_mult2(ptext[0]) ^ ptext[0]) ^ ptext[1] ^ ptext[2] ^ gf_mult2(ptext[3]);
temp[7] = (gf_mult2(ptext[4]) ^ ptext[4]) ^ ptext[5] ^ ptext[6] ^ gf_mult2(ptext[7]);
temp[11] = (gf_mult2(ptext[8]) ^ ptext[8]) ^ ptext[9] ^ ptext[10] ^ gf_mult2(ptext[11]);
temp[15] = (gf_mult2(ptext[12]) ^ ptext[12]) ^ ptext[13] ^ ptext[14] ^ gf_mult2(ptext[15]);
```

列混合 – MixColumns （故障从1字节扩散到4字节）

10.3(1) AES故障效应分析

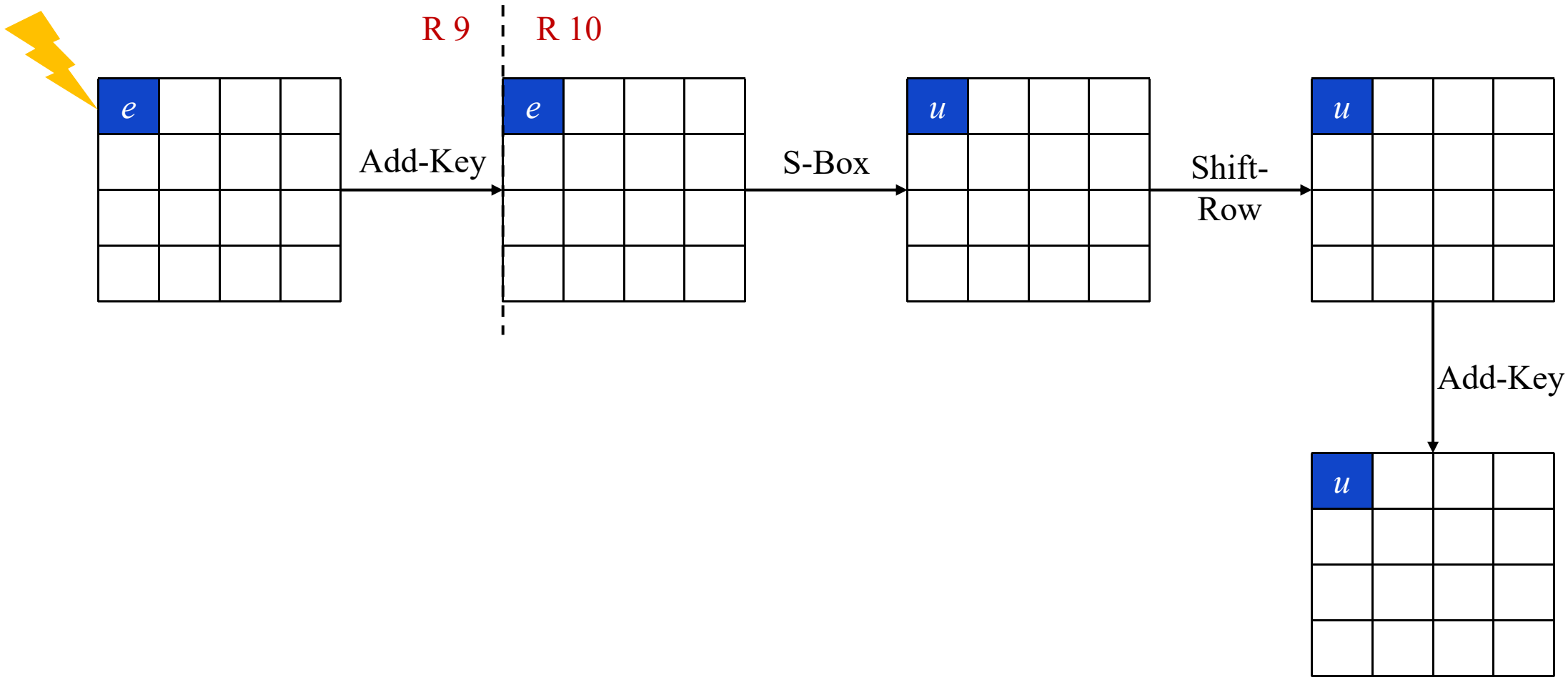


AES State Array

0	4	8	12
1	5	9	13
2	6	10	14
3	7	11	15

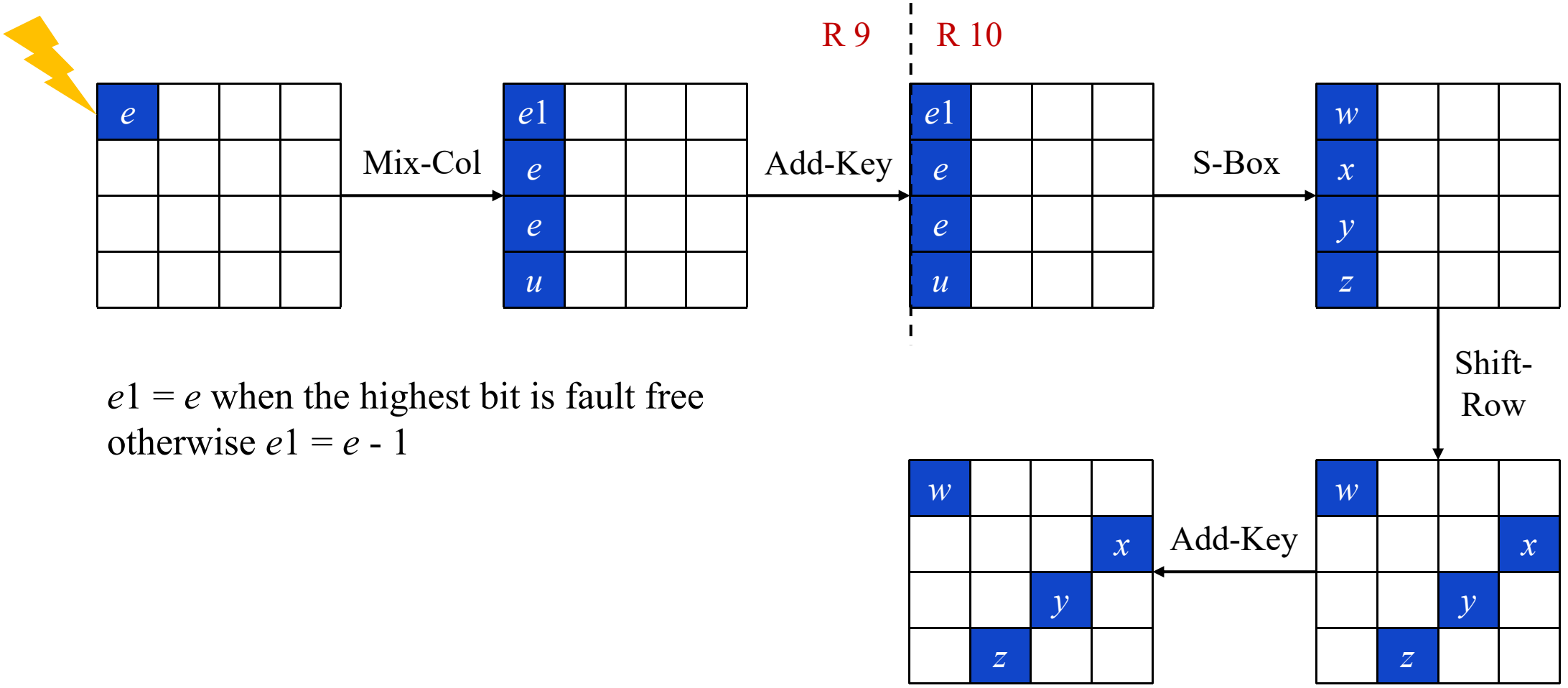
第十轮S盒输入故障

10.3(1) AES故障效应分析



第九轮加轮密钥故障

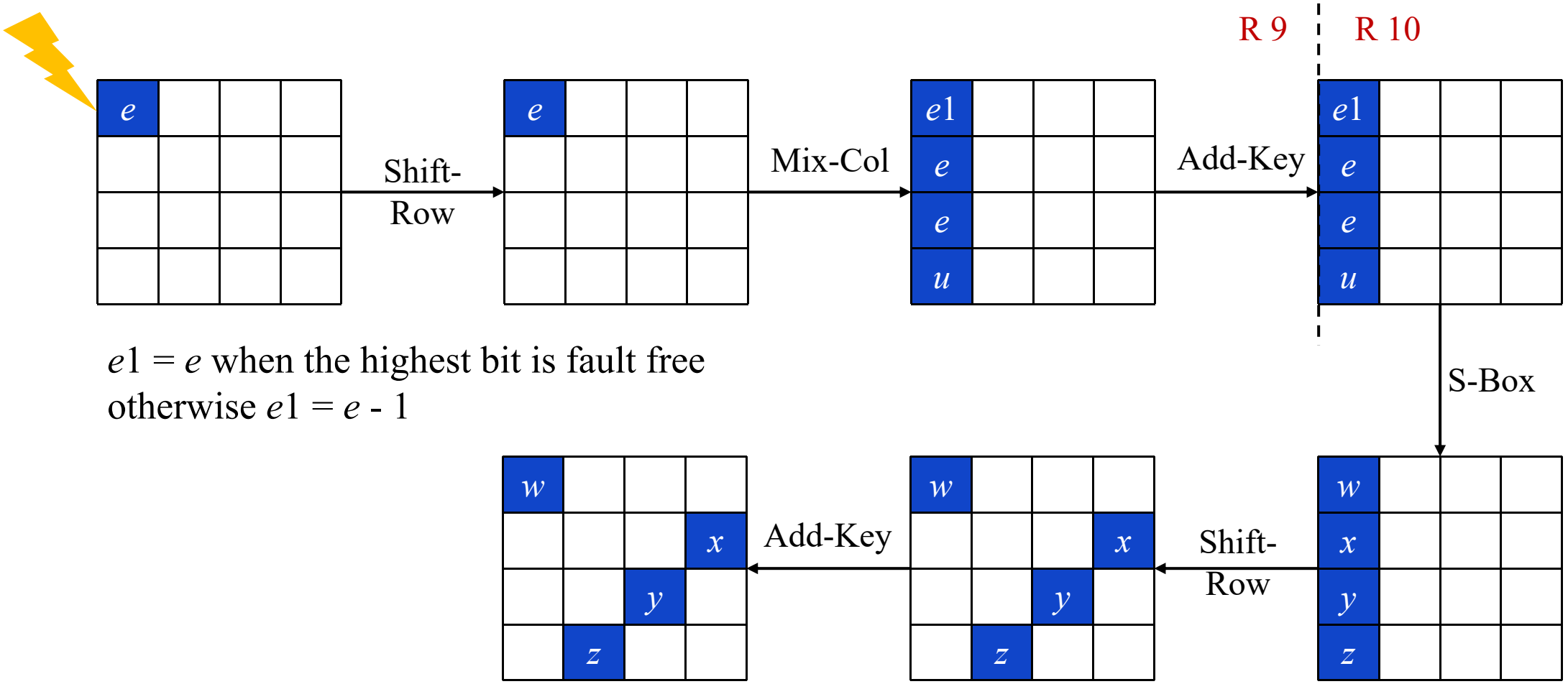
10.3(1) AES故障效应分析



$e1 = e$ when the highest bit is fault free
otherwise $e1 = e - 1$

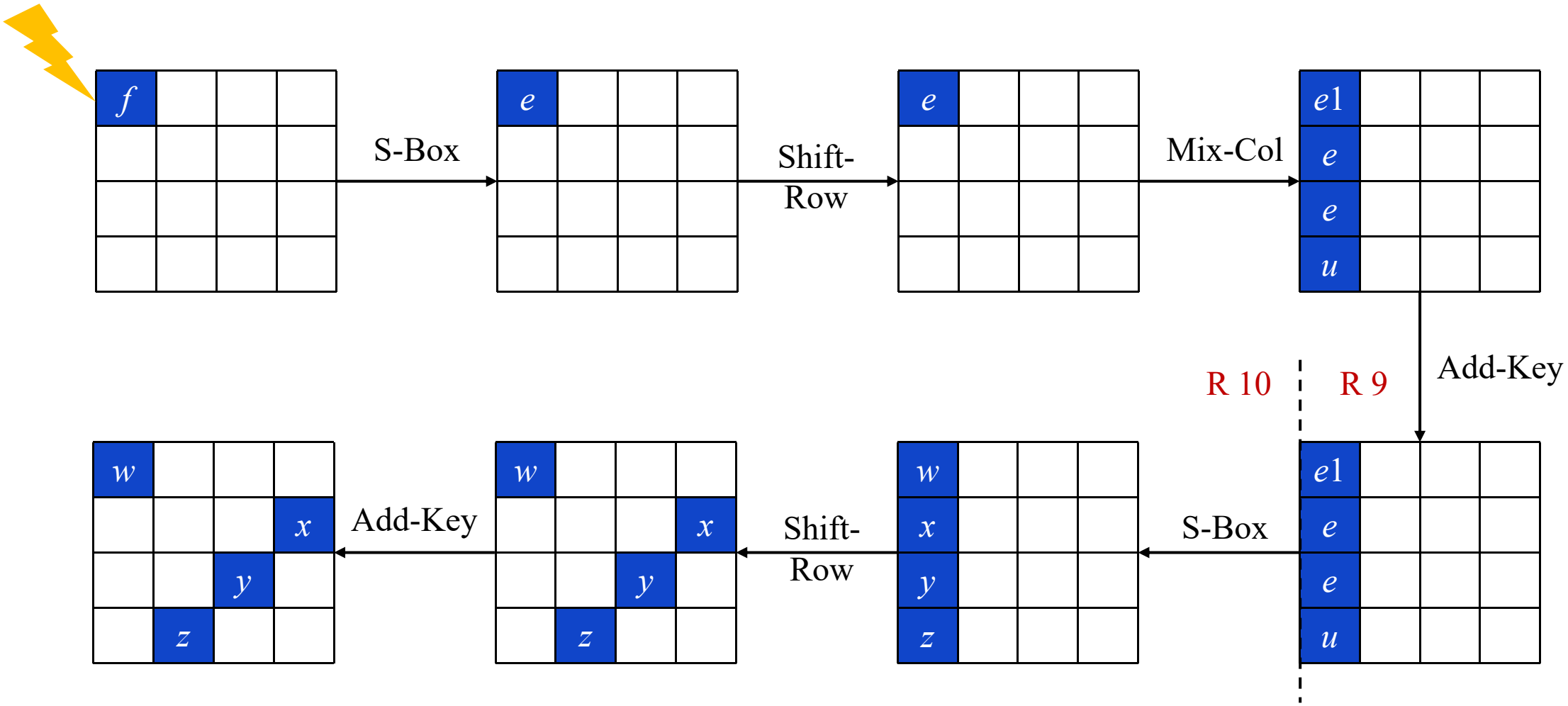
第九轮列混合故障

10.3(1) AES故障效应分析



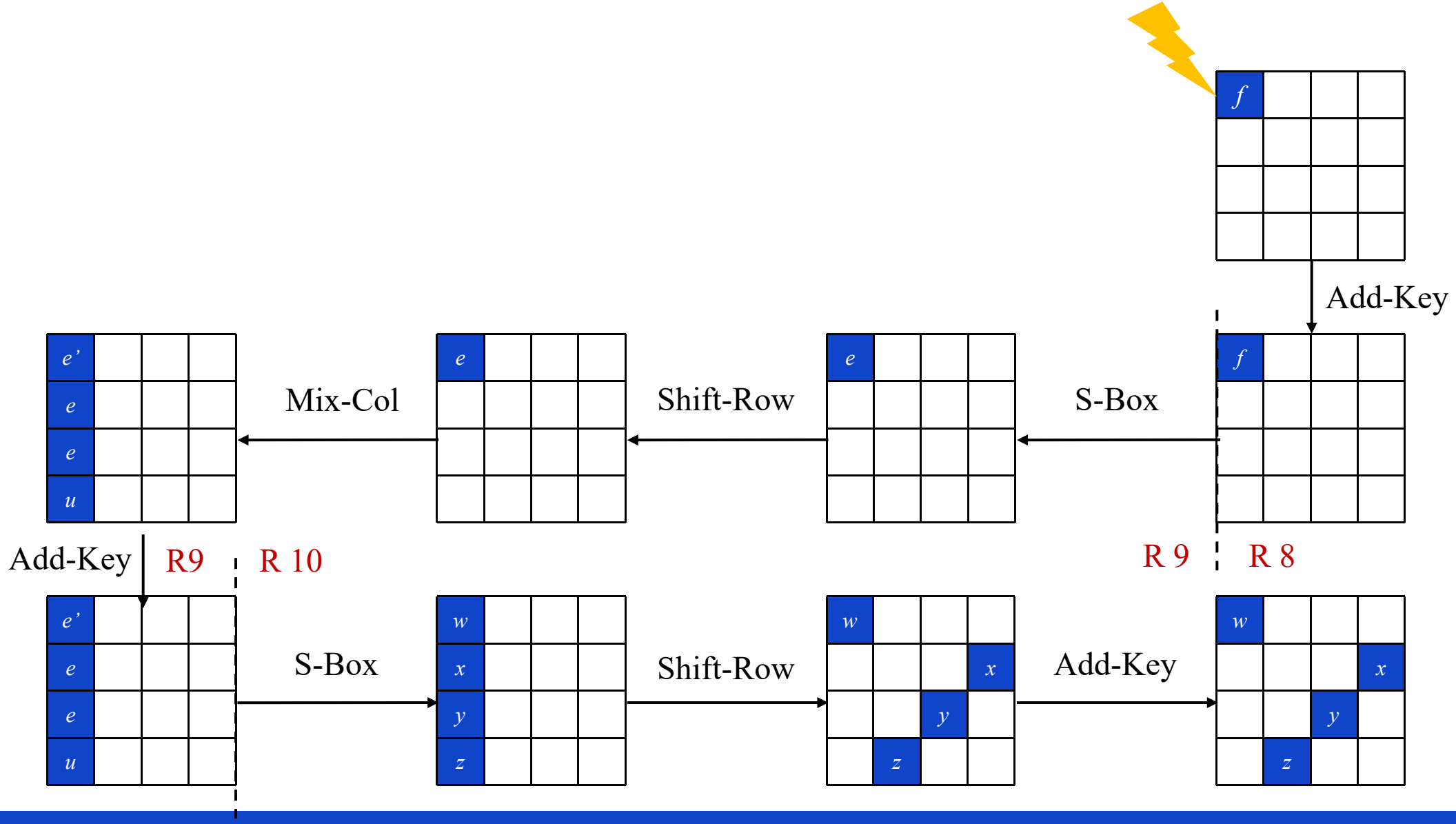
$e1 = e$ when the highest bit is fault free
otherwise $e1 = e - 1$

10.3(1) AES故障效应分析



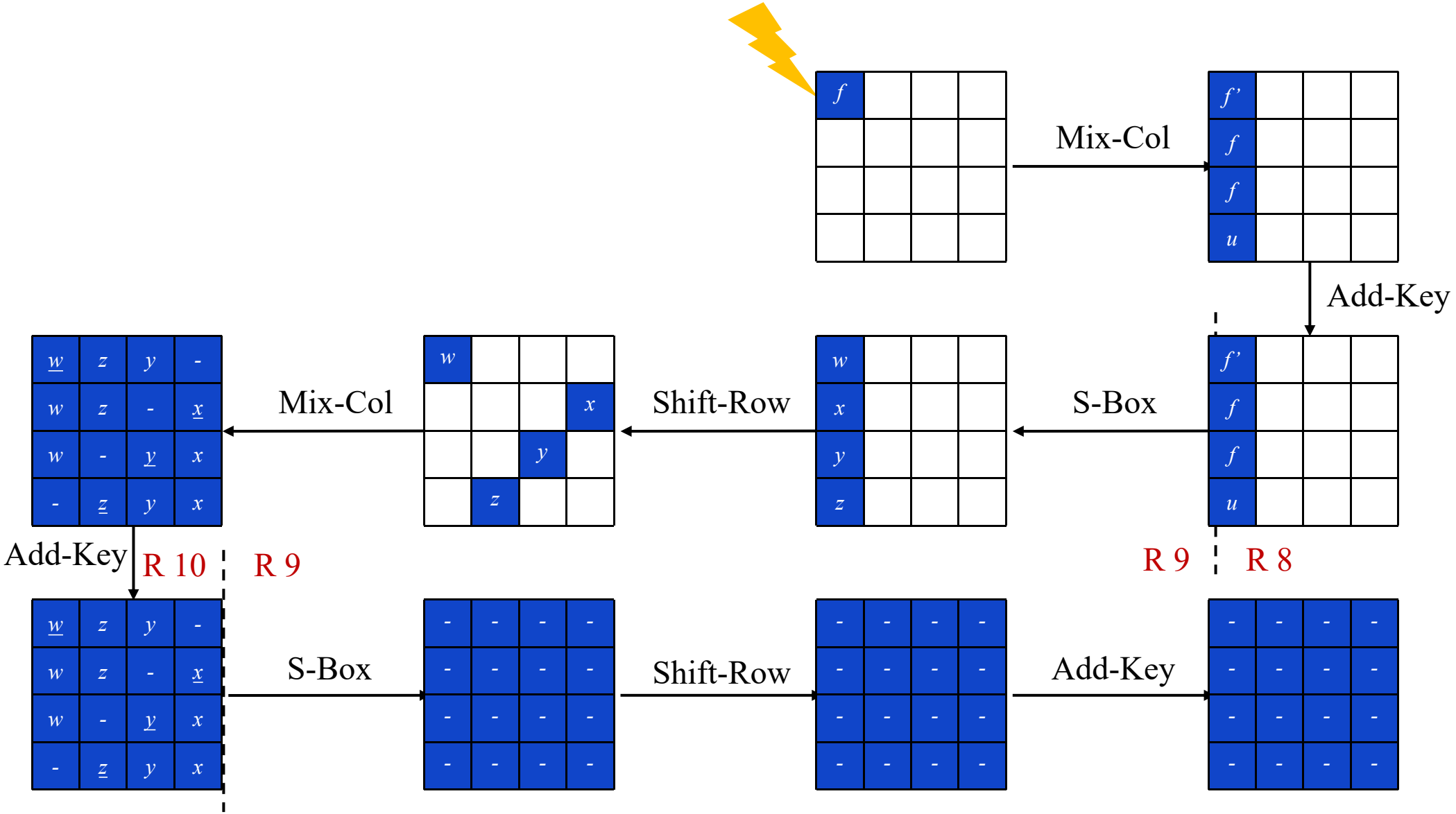
第九轮S盒输入故障

10.3(1) AES故障效应分析



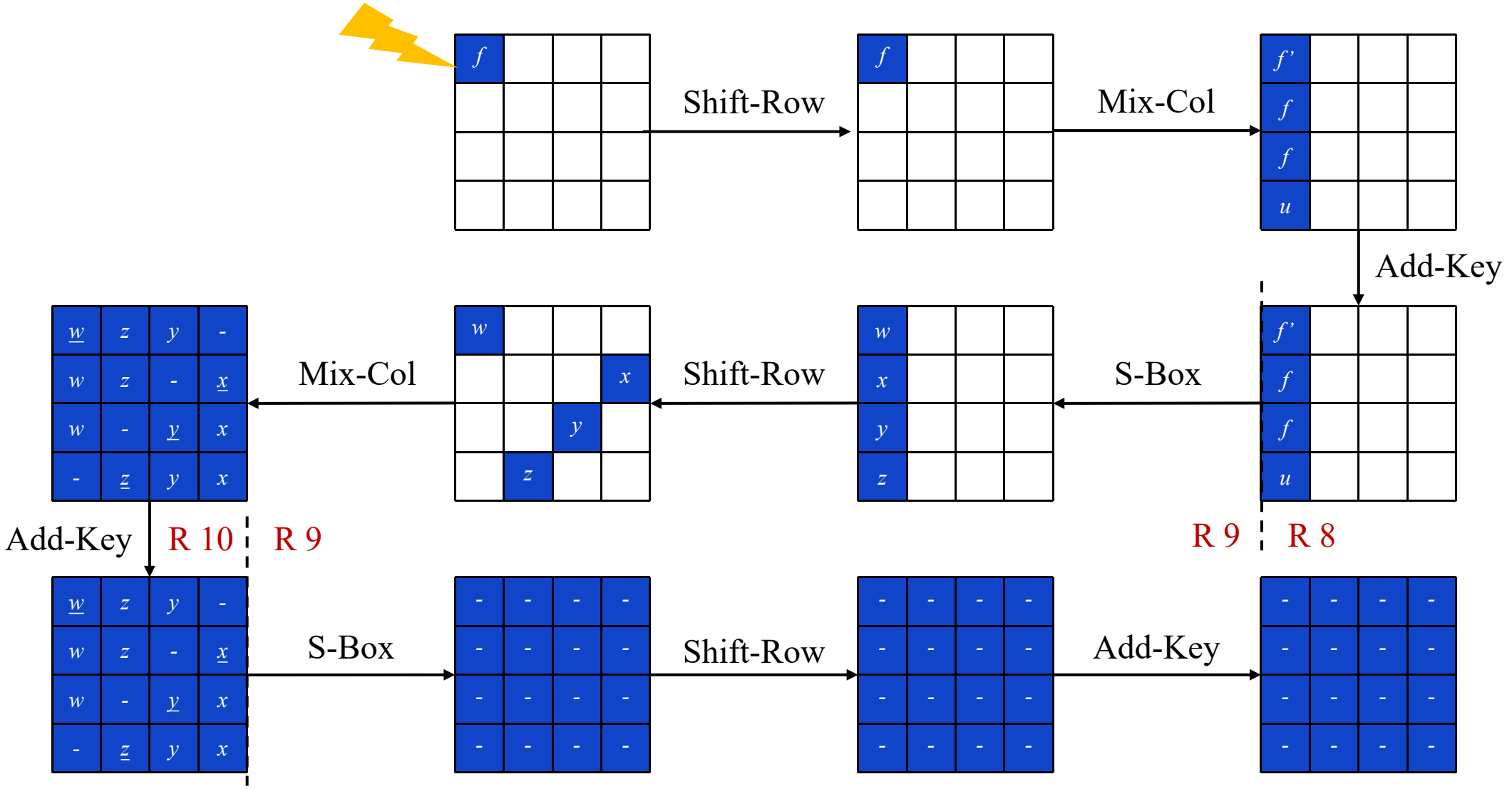
第八轮加轮密钥故障

10.3(1) AES故障效应分析



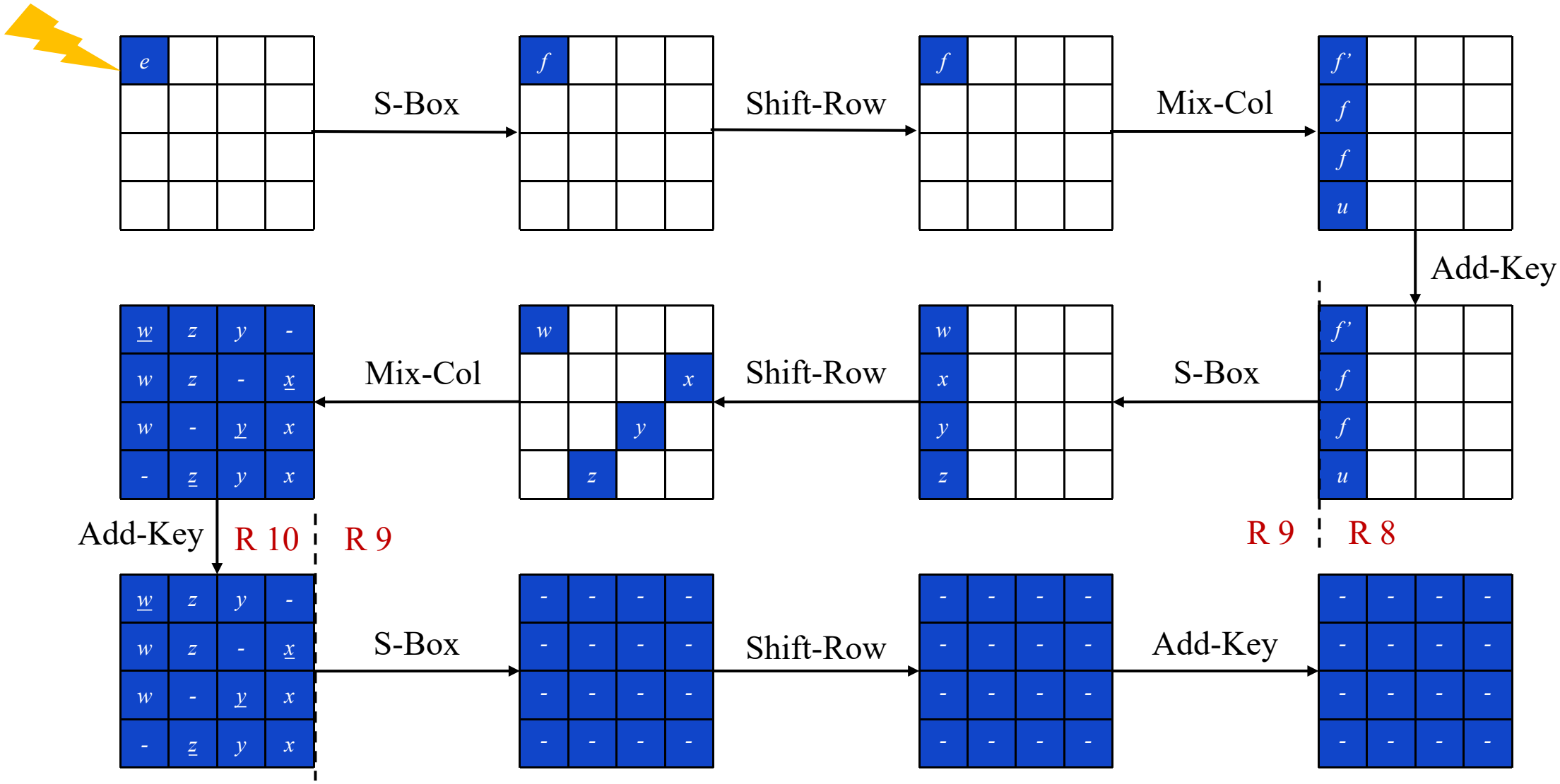
第八轮列混合故障

10.3(1) AES故障效应分析



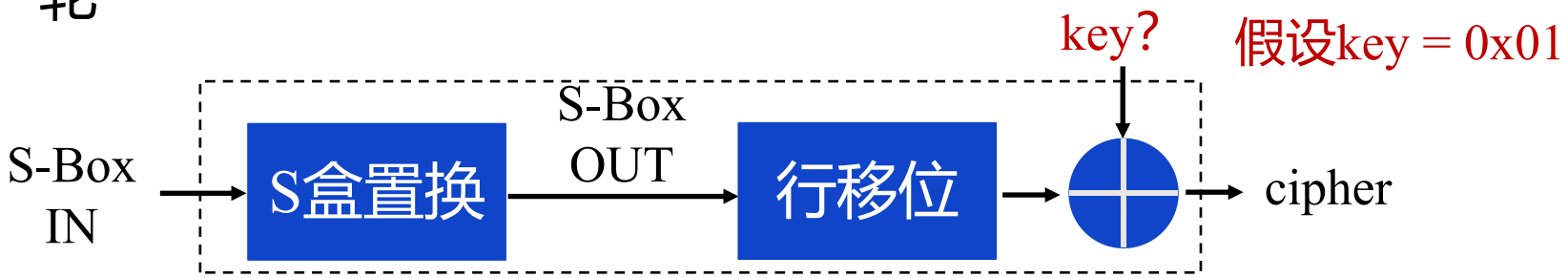
第八轮行移位故障

10.3(1) AES故障效应分析



第八轮S盒输入故障

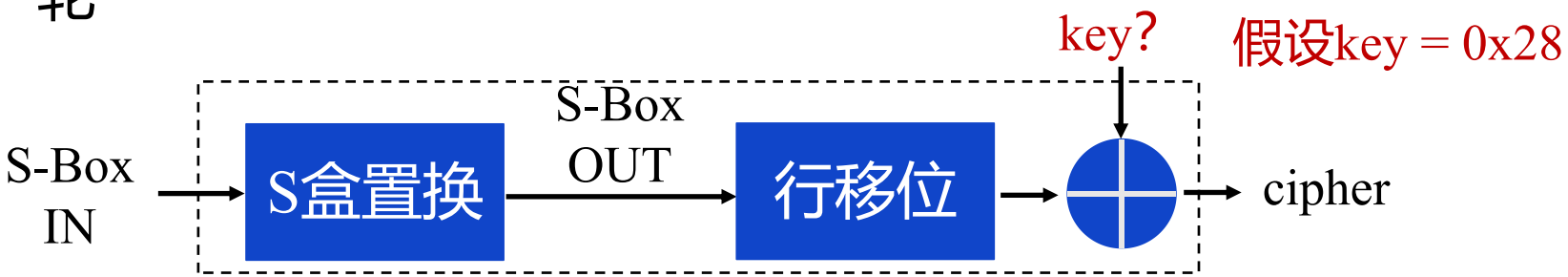
- 攻击最后一轮



correct	6f	e9	1e	1e	1f
faulty		86	44	44	45
correct	30	0a	67	67	66
faulty		3a	80	80	81
correct	e3	78	bc	bc	bd
faulty		9a	b8	b8	b9
…	…	…	…	…	…

10.3(2) AES故障注入攻击

- 攻击最后一轮



correct	01	b2	37	37	1f
faulty		b3	6d	6d	45
correct	01	b6	4e	4e	66
faulty		b7	a9	a9	81
correct	01	ac	95	95	bd
faulty		ad	91	91	b9
…	…	…	…	…	…

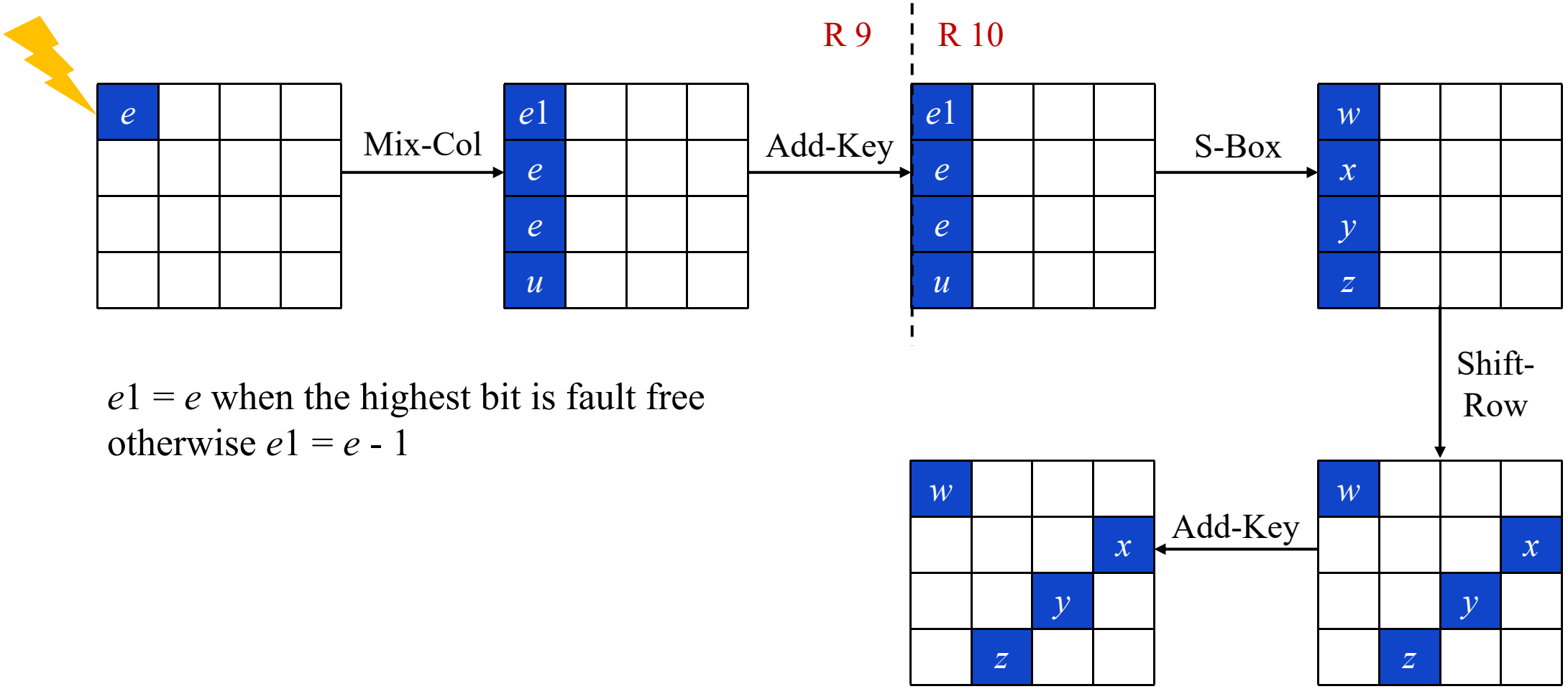
10.3(3) AES相关故障注入攻击

- 攻击倒数第二轮，在列混合之前注入故障

$$M\left(A \oplus \begin{bmatrix} e_1 & 0 & 0 & 0 \\ e_2 & 0 & 0 & 0 \\ e_3 & 0 & 0 & 0 \\ e_4 & 0 & 0 & 0 \end{bmatrix}\right) = M(A) \oplus \begin{bmatrix} 2 \bullet e_1 \oplus 3 \bullet e_2 \oplus e_3 \oplus e_4 = e'_1 & 0 & 0 & 0 \\ e_1 \oplus 2 \bullet e_2 \oplus 3 \bullet e_3 \oplus e_4 = e'_2 & 0 & 0 & 0 \\ e_1 \oplus e_2 \oplus 2 \bullet e_3 \oplus 3 \bullet e_4 = e'_3 & 0 & 0 & 0 \\ 3 \bullet e_1 \oplus e_2 \oplus e_3 \oplus 2 \bullet e_4 = e'_4 & 0 & 0 & 0 \end{bmatrix}$$

现假设$e_2 = e_3 = e_4 = 0$，考虑只有一个字节出错的情况

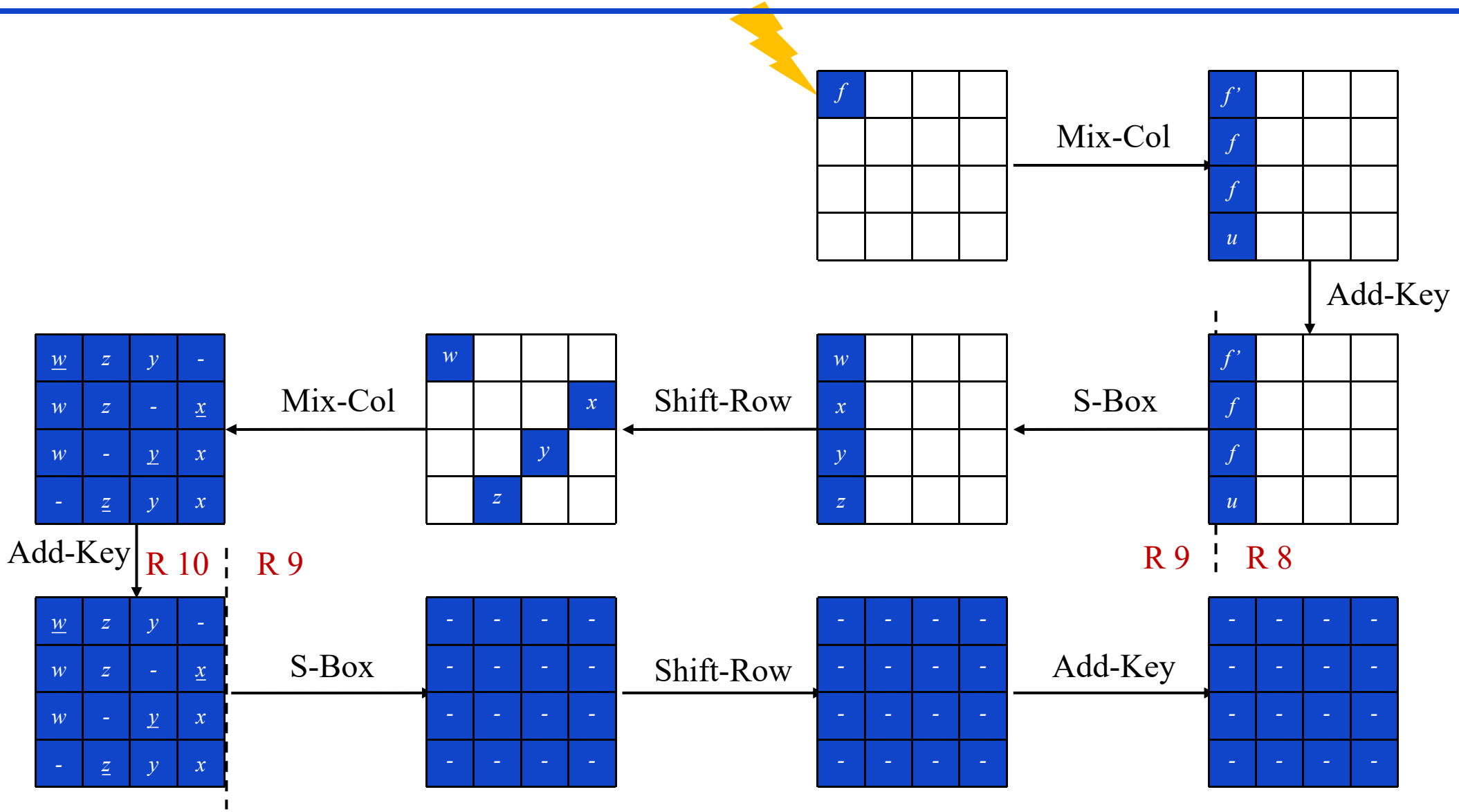
10.3(3) AES相关故障注入攻击



$e1 = e$ when the highest bit is fault free
otherwise $e1 = e - 1$

第九轮列混合故障

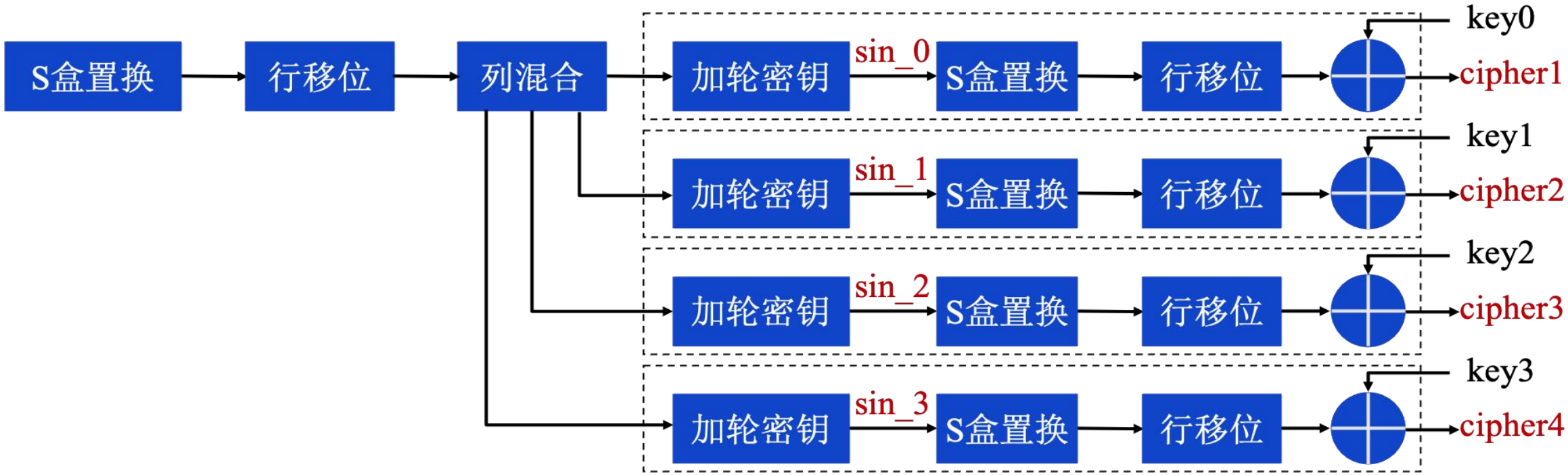
10.3(3) AES相关故障注入攻击



第八轮列混合故障

10.3(3) AES相关故障注入攻击

列混合前错误状态 ε_0 → 列混合后错误状态 $\begin{bmatrix} 2*\varepsilon_0 & 0 & 0 & 0 \\ \varepsilon_0 & 0 & 0 & 0 \\ \varepsilon_0 & 0 & 0 & 0 \\ 3*\varepsilon_0 & 0 & 0 & 0 \end{bmatrix}$ => $\begin{bmatrix} e_0 & 0 & 0 & 0 \\ e_1 & 0 & 0 & 0 \\ e_2 & 0 & 0 & 0 \\ e_3 & 0 & 0 & 0 \end{bmatrix}$



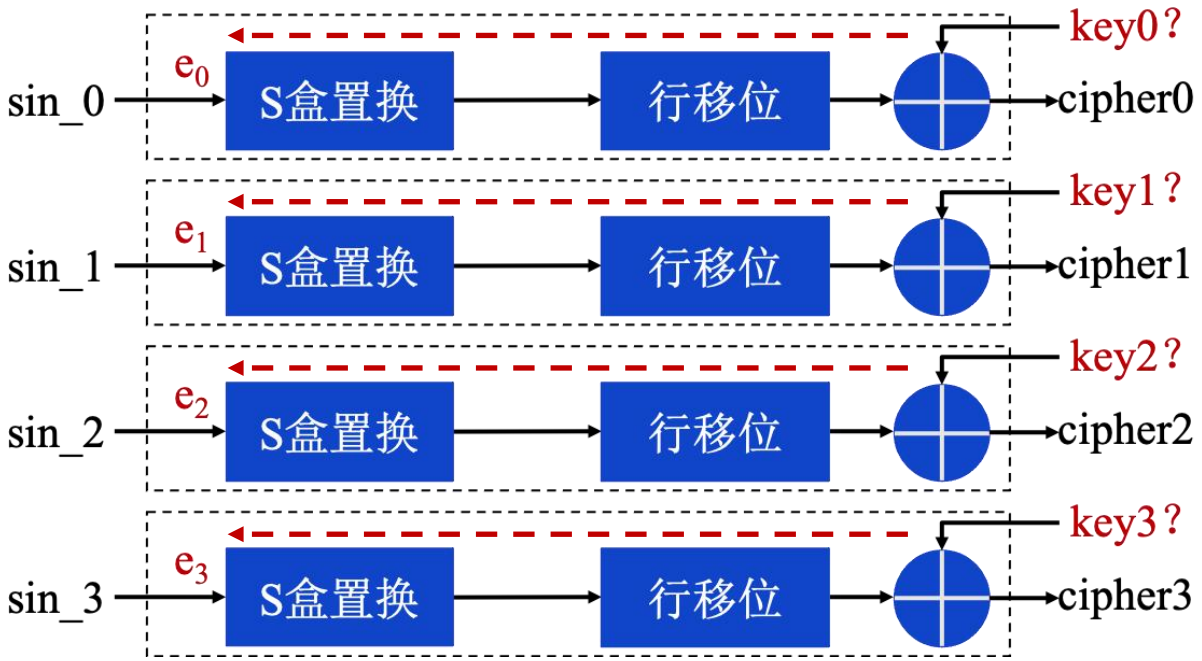
10.3(3) AES相关故障注入攻击

列混合前错误状态 ε_0 ➡ 列混合后错误状态 $\begin{bmatrix} 2*\varepsilon_0 & 0 & 0 & 0 \\ \varepsilon_0 & 0 & 0 & 0 \\ \varepsilon_0 & 0 & 0 & 0 \\ 3*\varepsilon_0 & 0 & 0 & 0 \end{bmatrix} \Rightarrow \begin{bmatrix} e_0 & 0 & 0 & 0 \\ e_1 & 0 & 0 & 0 \\ e_2 & 0 & 0 & 0 \\ e_3 & 0 & 0 & 0 \end{bmatrix}$

$$e_1 = e_2$$

$$e_0 = 2 * e_1$$

$$e_3 = e_0 + e_1$$



课 后 阅 读

- Paul C. Kocher, Joshua Jaffe, and Benjamin Jun. 1999. Differential Power Analysis. In Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology (CRYPTO '99). Springer-Verlag, Berlin, Heidelberg, 388–397.
- P. C Kocher, J. M. Jaffe , and B. C. Jun . "Differential power analysis," Springer-Verlag 2009.


感谢聆听!
THANK YOU FOR YOUR ATTENTION!