《现代密码学》第四讲

分组密码（三）

《现代密码学》第四讲

高级加密标准（AES）算法介绍

上讲内容回顾

- DES 算法的整体结构——Feistel 结构
- DES 算法的轮函数
- DES 算法的密钥编排算法
- DES 的解密变换

本节主要内容

- AES 算法的整体结构
- AES 算法的轮函数
- AES 算法的密钥编排算法
- AES 的解密变换
- DES 的扩散和 AES 的扩散

本节主要内容

- AES 算法的整体结构
- AES 算法的轮函数
- AES 算法的密钥编排算法
- AES 的解密变换
- DES 的扩散和 AES 的扩散

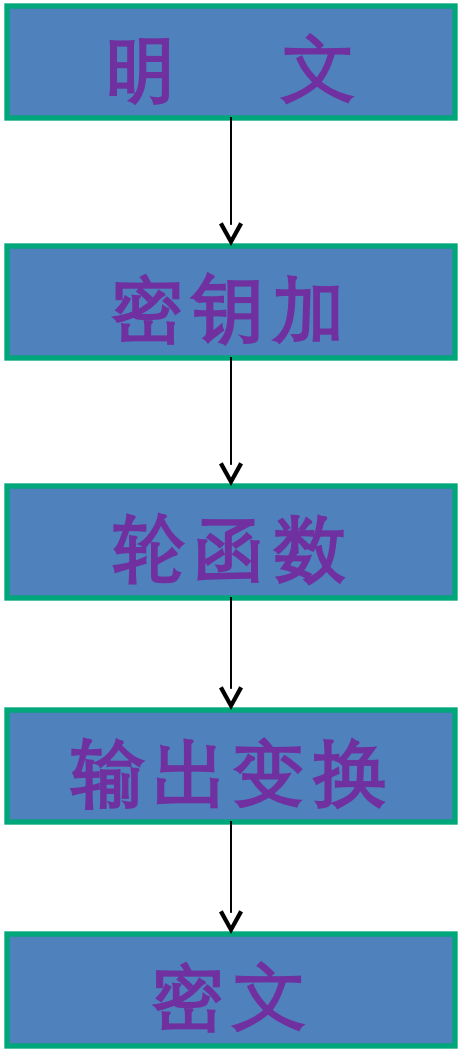
AES 算法的整体结构

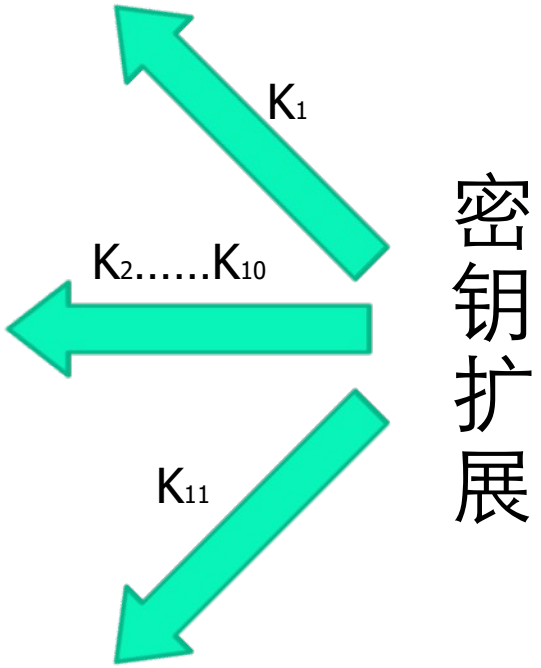
- Rijndael 由比利时的 Joan Daemen 和 Vincent Rijmen 设计，算法的原型是 Square 算法，经过修改后确定为高级数据加密标准 AES.
- 典型的 SPN 结构
- 有较好的数学理论作为基础；结构简单、速度快

	Key Length (Nk words)	Block Size (Nb words)	Number of Rounds (Nr)
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

AES 算法的整体结构

下面以AES－128版本为例





本节主要内容

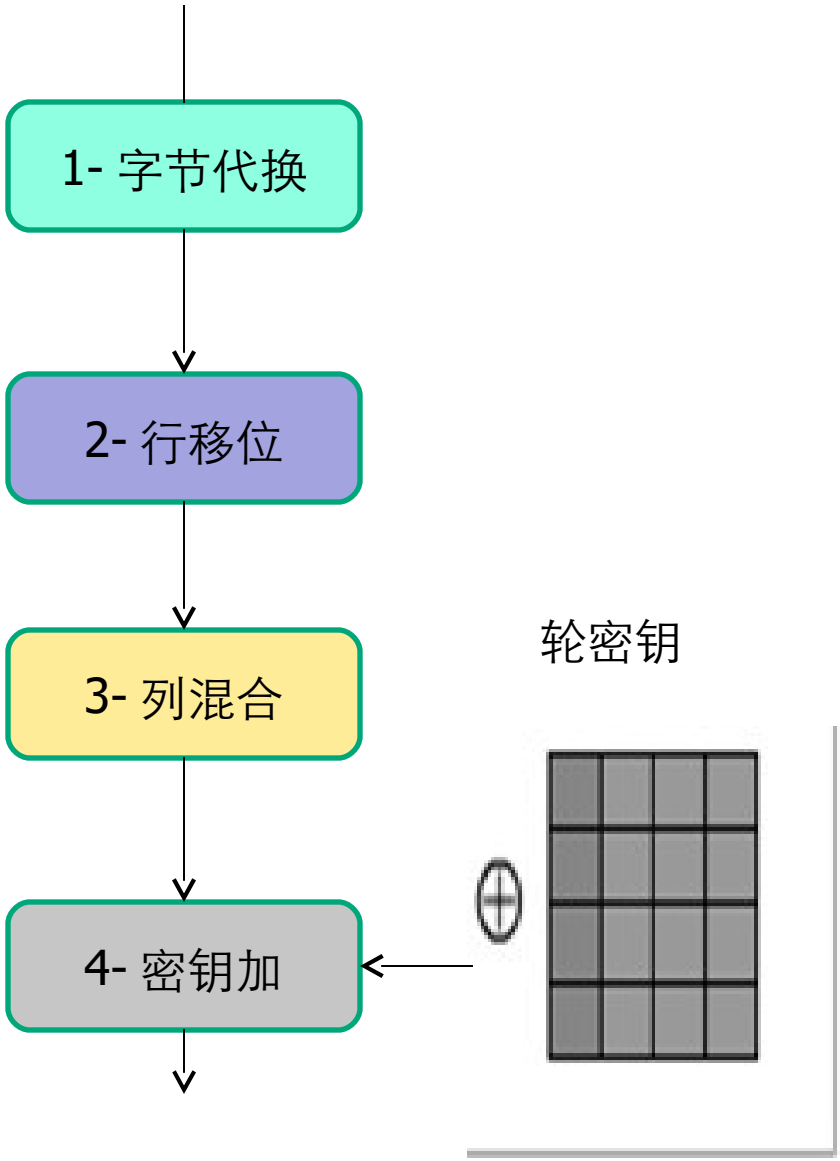
- AES 算法的整体结构
- **AES 算法的轮函数**
- AES 算法的密钥编排算法
- AES 的解密变换
- DES 的扩散和 AES 的扩散

AES 算法的轮函数

Rijndael 的轮函数将 128 比特中间状态映射为 128 比特输出。

由 4 个变换组成，依次为：

- 1）字节代换（SubByte）
- 2）行移位（ShiftRow）
- 3）列混合（MixColumn）
- 4）密钥加（AddRoundKey）

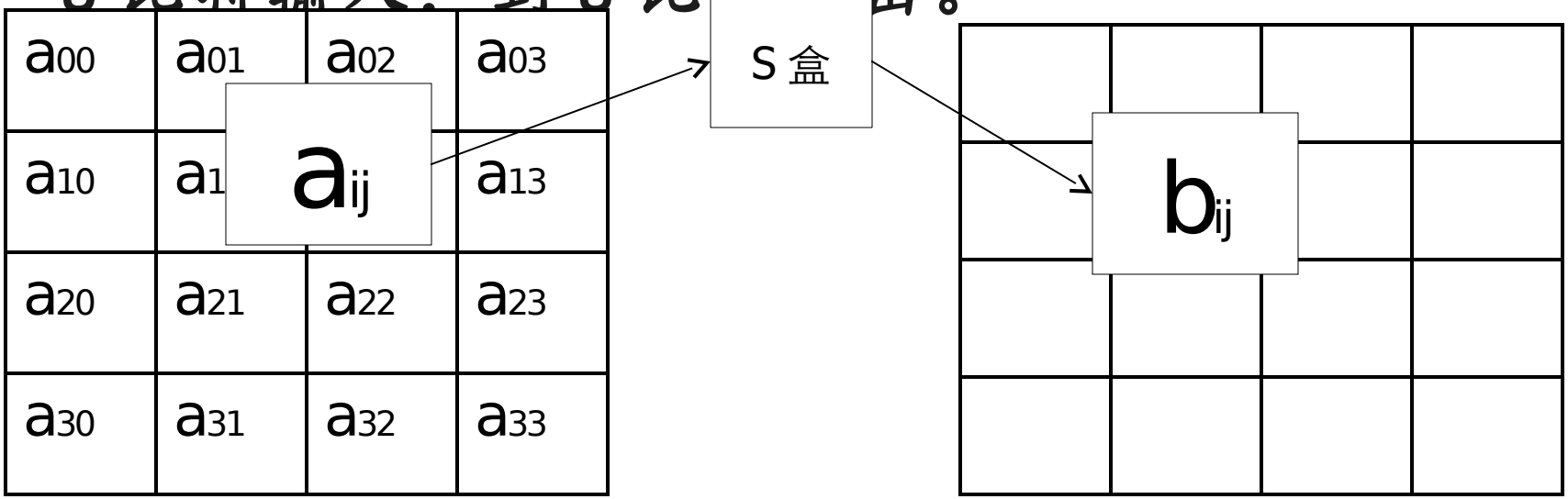


AES 算法的轮函数

1） 字节代换（ByteSub）

字节代换是非线形变换，独立地对状态的每个字节进行，代换表（即 S- 盒）是可逆的 .

8 比特输入，到 8 比特输出。

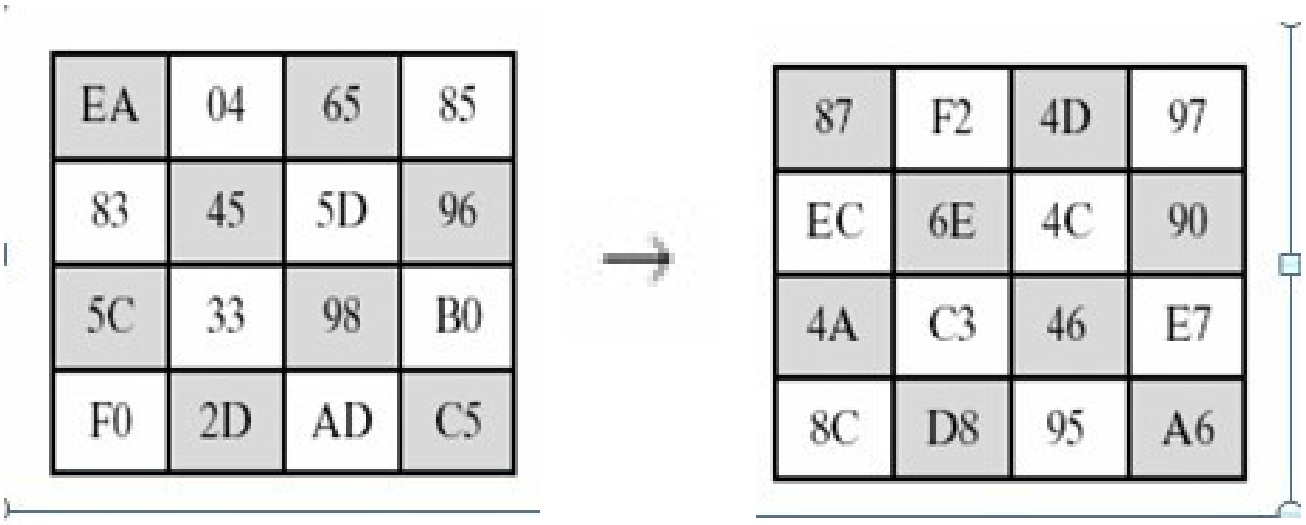


AES 算法的轮函数

x \ y	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

AES 算法的轮函数

例：字节代换（128 比特分组）



AES 算法的轮函数

上述 S- 盒由以下两个变换的合成得到：

首先，将字节看作 GF(2⁸) （约化多项式：$m(x)=x^8+x^4+x^3+x+1$ ）上的元素：

$$s_7s_6s_5s_4s_3s_2s_1s_0 \mapsto s(x),$$

$$s(x)=s_7x^7+s_6x^6+s_5x^5+s_4x^4+s_3x^3+s_2x^2+s_1x+s_0$$

然后，映射到自己的乘法逆元，‘00’映射到‘00’ .

$$s(x) \mapsto s(x)^{-1}$$

最后，对字节做如下的（GF(2) 上的，可逆的）仿射变换：

AES 算法的轮函数

$$
\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \end{bmatrix}
=
\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}
\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{bmatrix}
+
\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}
$$

AES 算法的轮函数

2) 行移位（ShiftRow）

行移位是将状态阵列的各行进行循环移位，不同状态行的位移量不同：

第 0 行不移动，

第 1 行循环左移 C_1 个字节，

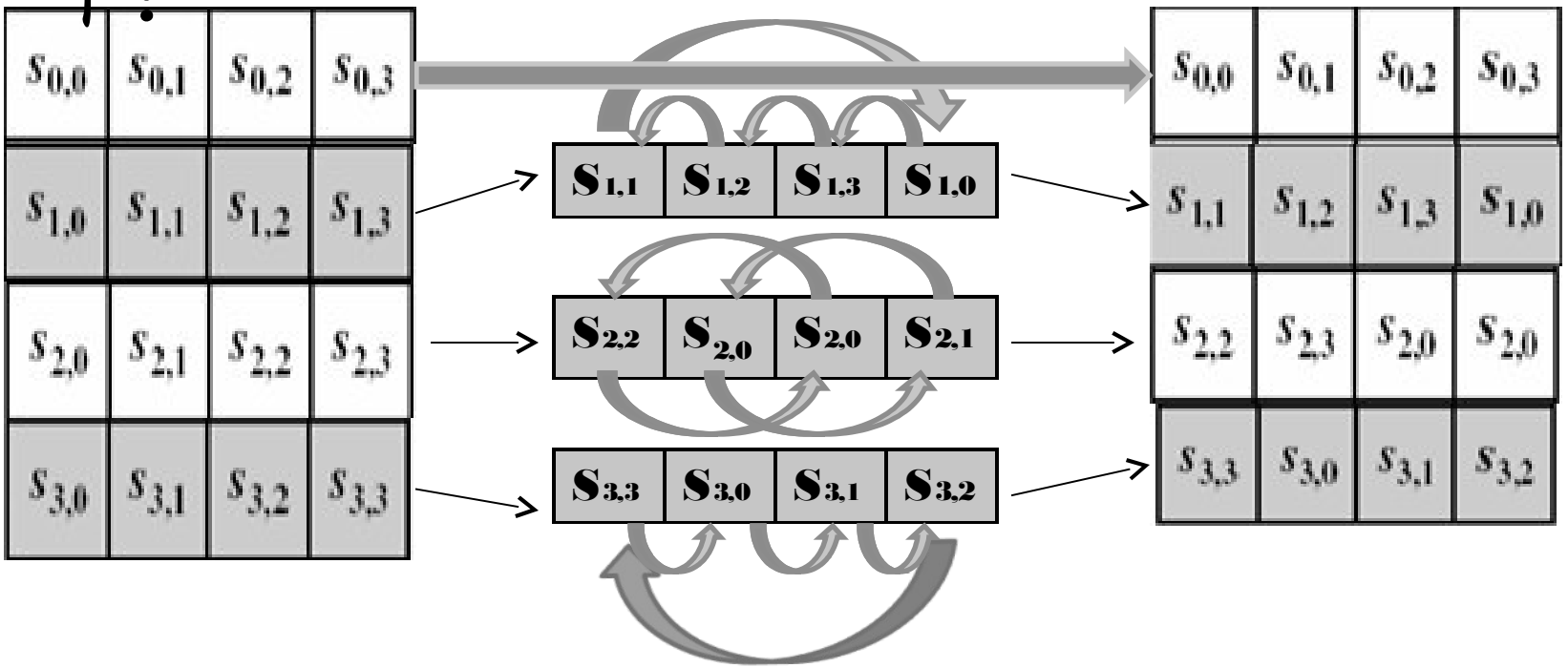
第 2 行循环左移 C_2 个字节，

第 3 行循环左移 C_3 个字节。

位移量 C_1、C_2、C_3 的取值与 N_b 有关，由下表给出：

AES 算法的轮函数

例：当 Nb=4 时，具体的操作如下：



AES 算法的轮函数

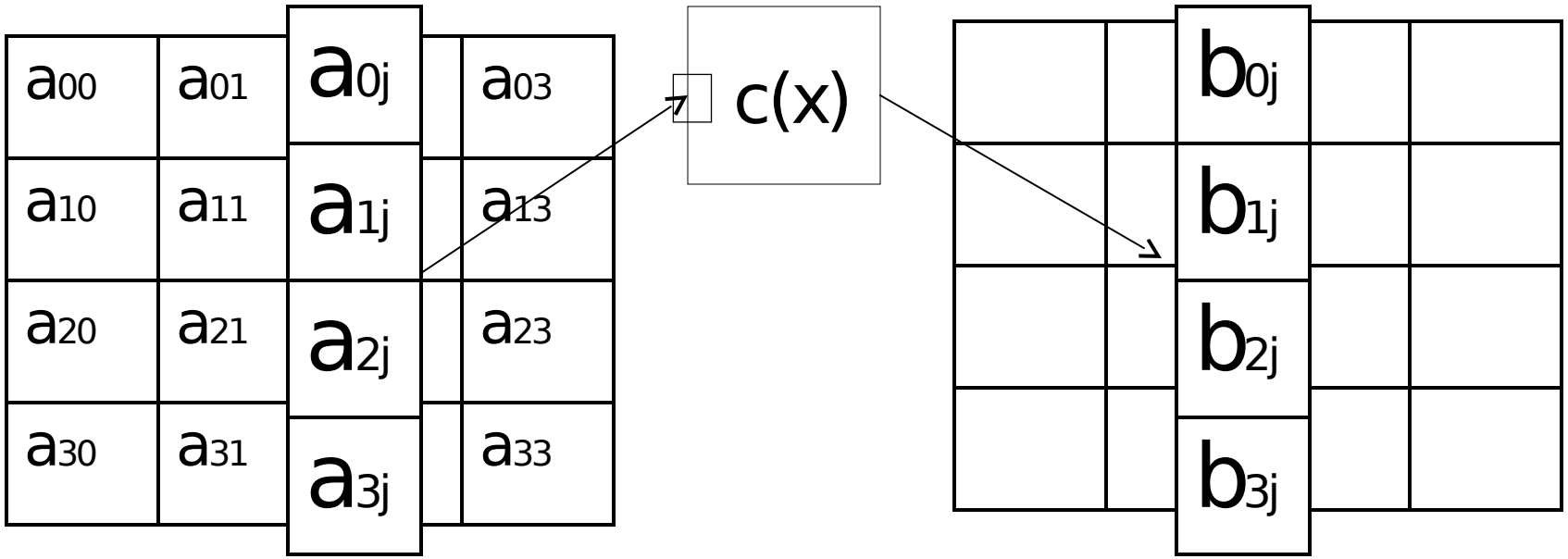
- 3）列混合（MixColumn）

列混合变换中，将状态阵列的每列视为 GF((2^8)^4) 上的多项式，再与一个固定的多项式 c(x) 进行模 x^4+1 乘法．

Rijndael 的设计者给出的 c(x) 为（系数用十六进制数表示）：

c(x)=‘03’ x^3+‘01’ x^2+‘01’ x+‘02’

AES 算法的轮函数



列混合运算示意图

AES 算法的轮函数

列混合运算也可写为矩阵乘法．

设 b(x)= c(x)(a(x)，则

$$\begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{pmatrix} = \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{pmatrix}$$

AES 算法的轮函数

GF(2⁸) 的多项式乘法，约化多项式为

$$m(x)=x^8+x^4+x^3+x+1$$

例： 57_x 乘以 83_x

$$(x^6+x^4+x^2+x+1) \times (x^7+x+1)$$

$$= (x^{13}+x^{11}+x^9+x^8+x^7) \hat{} (x^7+x^5+x^3+x^2+x) \hat{} (x^6+x^4+x^2+x+1)$$

$$= x^{13}+x^{11}+x^9+x^8+x^6+x^5+x^4+x^3+1$$

$$= x^7+x^6+1 \mod m(x)$$

AES 算法的轮函数

课堂练习：列混合运算（128 比特分组）

$$\begin{bmatrix} 87 \\ 6E \\ 46 \\ A6 \end{bmatrix} \bullet \begin{pmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{pmatrix} = \begin{bmatrix} 47 \\ 37 \\ 94 \\ ED \end{bmatrix}$$

AES 算法的轮函数

4) 密钥加（AddRoundKey）

密钥加是将轮密钥简单地与状态进行逐比特异或． 轮密钥由种子密钥通过密钥编排算法得到。

a_{00}	a_{01}	a_{02}	a_{03}
a_{10}	a_{11}	a_{12}	a_{13}
a_{20}	a_{21}	a_{22}	a_{23}
a_{30}	a_{31}	a_{32}	a_{33}

$\oplus$

k_{00}	k_{01}	k_{02}	k_{03}
k_{10}	k_{11}	k_{12}	k_{13}
k_{20}	k_{21}	k_{22}	k_{23}
k_{30}	k_{31}	k_{32}	k_{33}

$=$

b_{00}	b_{01}	b_{02}	b_{03}
b_{10}	b_{11}	b_{12}	b_{13}
b_{20}	b_{21}	b_{22}	b_{23}
b_{30}	b_{31}	b_{32}	b_{33}

本节主要内容

- AES 算法的整体结构
- AES 算法的轮函数
- **AES 算法的密钥编排算法**
- AES 的解密变换
- DES 的扩散和 AES 的扩散

AES 算法的密钥编排算法

密钥编排指从种子密钥得到轮密钥的过程，AES 的密钥编排由密钥扩展和轮密钥选取两部分组成，其基本原则如下：

- 1）轮密钥的总比特数等于轮数加 1 再乘以分组长度；如 128 比特的明文经过 10 轮的加密，则总共需要（10+1）*128=1408 比特的密钥.
- 2）种子密钥被扩展成为扩展密钥；
- 3）轮密钥从扩展密钥中取，其中第 1 轮轮密钥取扩展密钥的前 N_b 个字，第 2 轮轮密钥取接下来的 N_b 个字，依次类推.

AES 算法的密钥编排算法

- 1） 密钥扩展

扩展密钥是以 4 字节字为元素的一维阵列，表示为 W[Nb* (N_r+1)]，其中前 N_k 个字取为种子密钥，以后每个字按递归方式定义．扩展算法根据 $N_k \leqslant 6$ 和 $N_k > 6$ 有所不同。

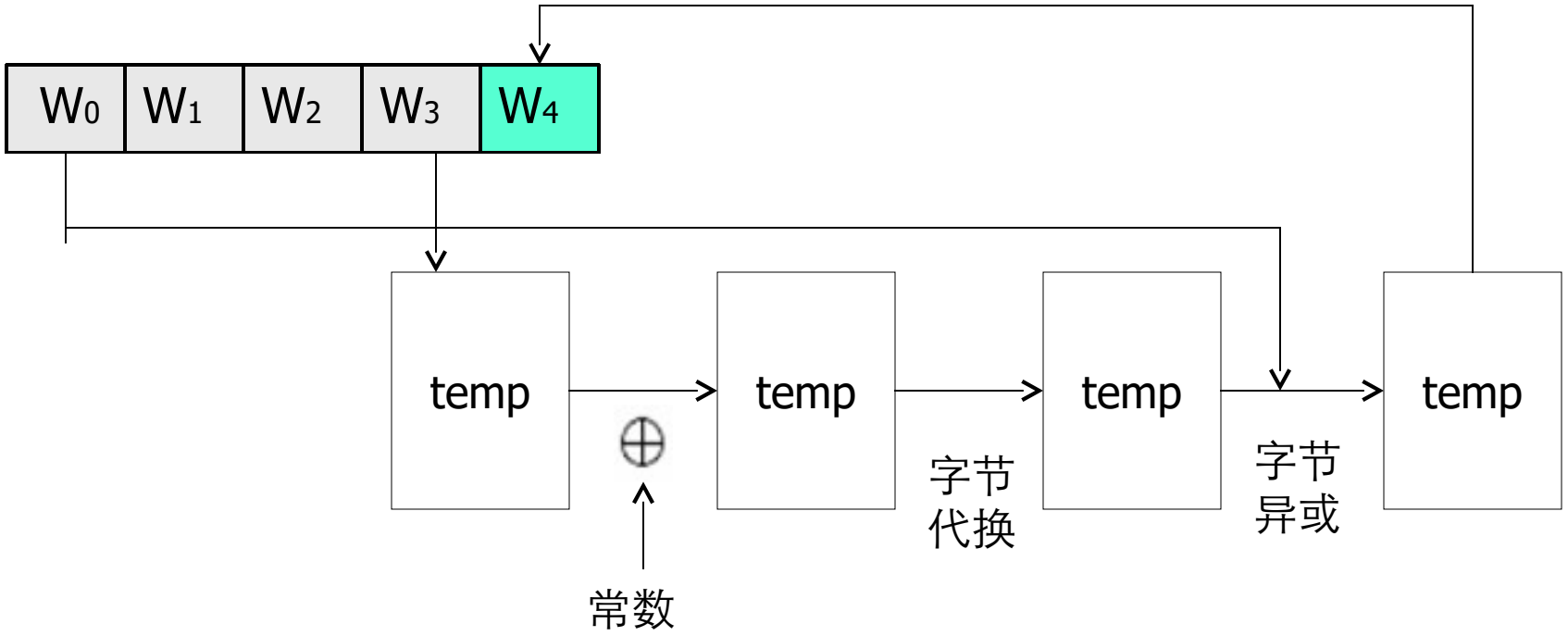
AES 算法的密钥编排算法

当 Nk≤6 时，扩展算法如下：

```
KeyExpansion (byteKey[4*Nk] , W[Nb*(Nr+1)])
{
    for (i =0; i < Nk; i ++)
    W[i]=(Key[4* i],Key[4* i +1],Key[4* i +2],Key[4*
i +3] );
    for (i =Nk; i <Nb*(Nr+1); i ++)
    {
    temp=W[i-1];
    if (i % Nk= =0)
          temp=SubByte (RotByte (temp))^Rcon[i /Nk];
    W[i]=W[i-Nk]^ temp;
    }
}
```

AES 算法的密钥编排算法

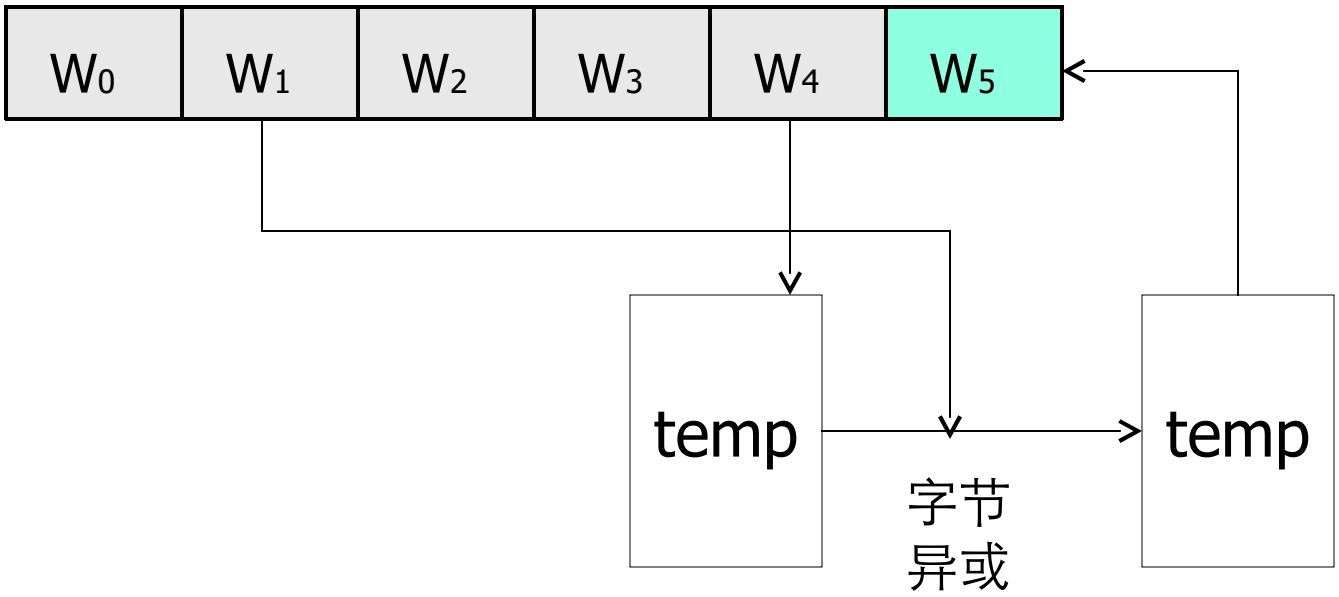
i=4　　i/4==0



Nk=4

AES 算法的密钥编排算法

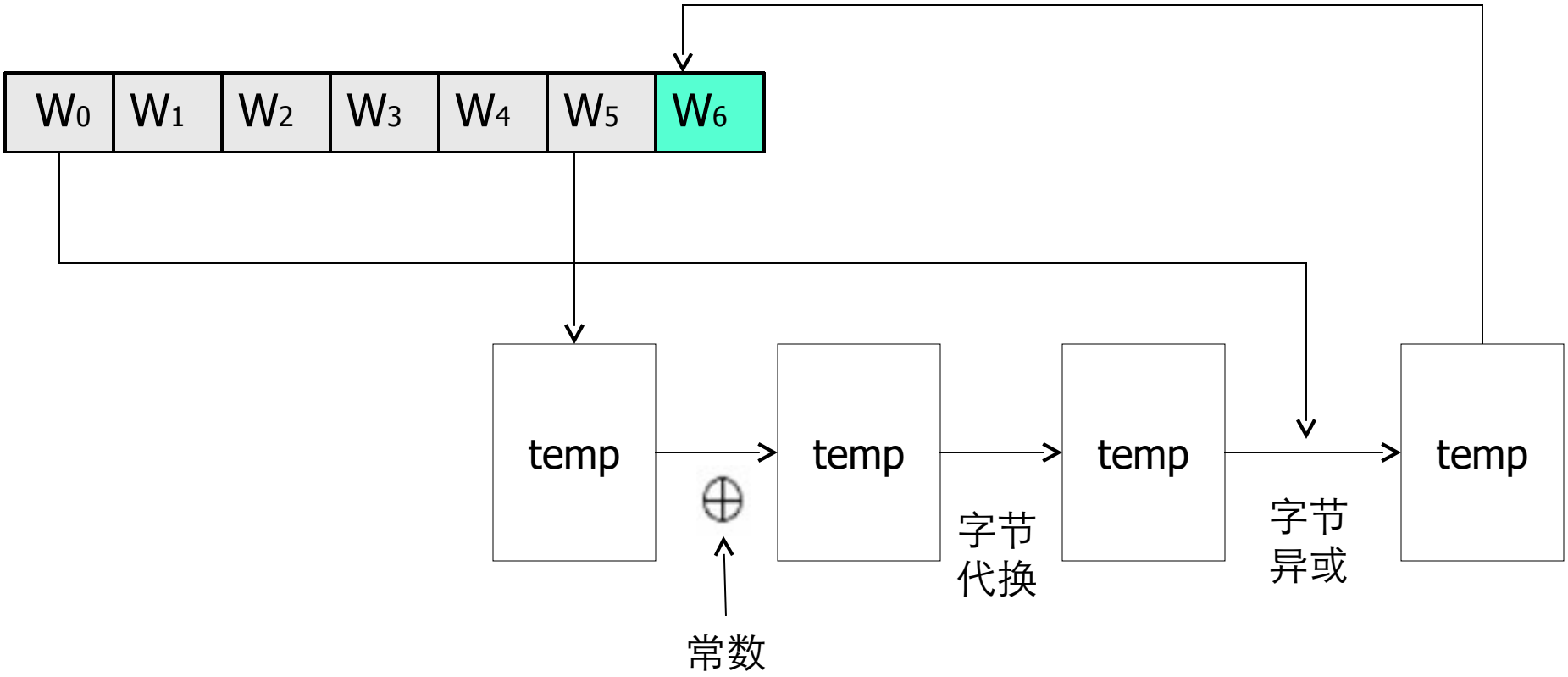
i=5 5/4 ≠0



Nk=4

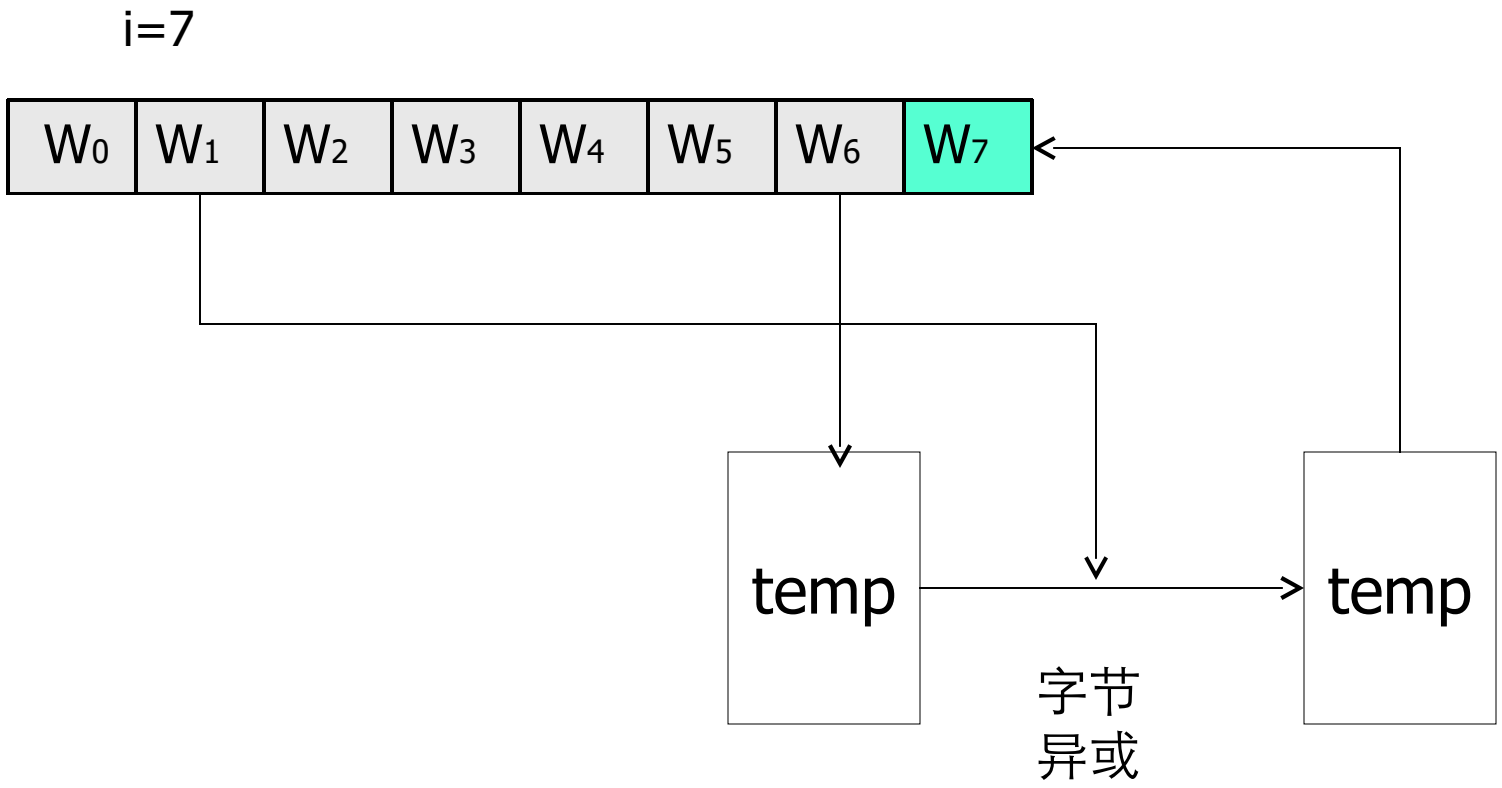
AES 算法的密钥编排算法

i=6　i/4==2



Nk=6

AES 算法的密钥编排算法



Nk=6

AES 算法的密钥编排算法

- Key[4*Nk] 为种子密钥，看作以字为元素的一维阵列；

- 函数 SubByte（）返回 4 字节字，其中每一个字节都是用 Rijndael 的 S 盒作用到输入字对应的字节得到；

- 函数 RotByte（） 也返回 4 字节字，该字由输入的字循环移位得到，即当输入字为（a，b，c，d）时，输出字为 （b，c，d，a）.

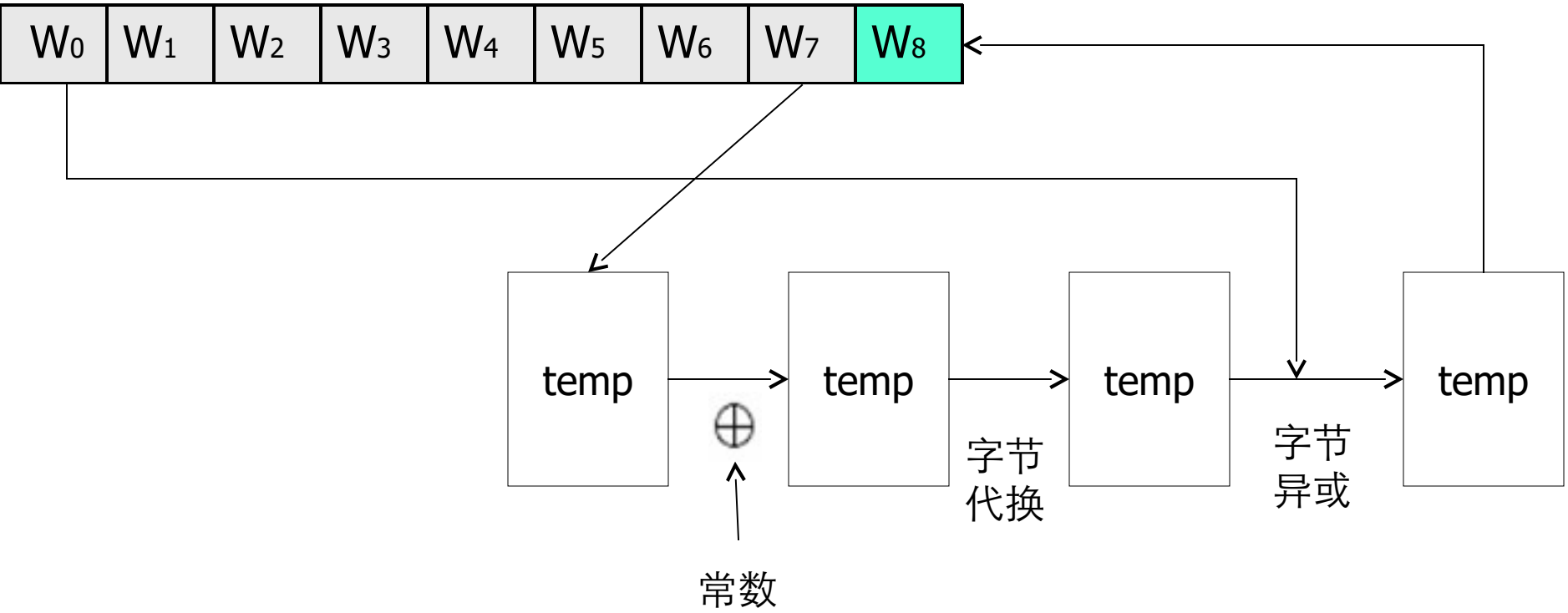
AES 算法的密钥编排算法

当 Nk>6 时，扩展算法如下：

```
KeyExpansion (byte  Key[4*Nk] , W[Nb*(Nr+1)])
{
     for (i=0; i < Nk; i ++)
     W[i]=(Key[4* i], Key[4* i +1], Key[4* i +2],
Key[4* i +3] );
for (i =Nk; i <Nb*(Nr+1); i ++)
     {
     temp=W[i -1];
     if (i % Nk= =0)
          temp=SubByte (RotByte (temp))^Rcon[i /Nk];
        else if (i % Nk==4)
            temp=SubByte (temp);
W[i]=W[i - Nk]^ temp;
     }
}
```

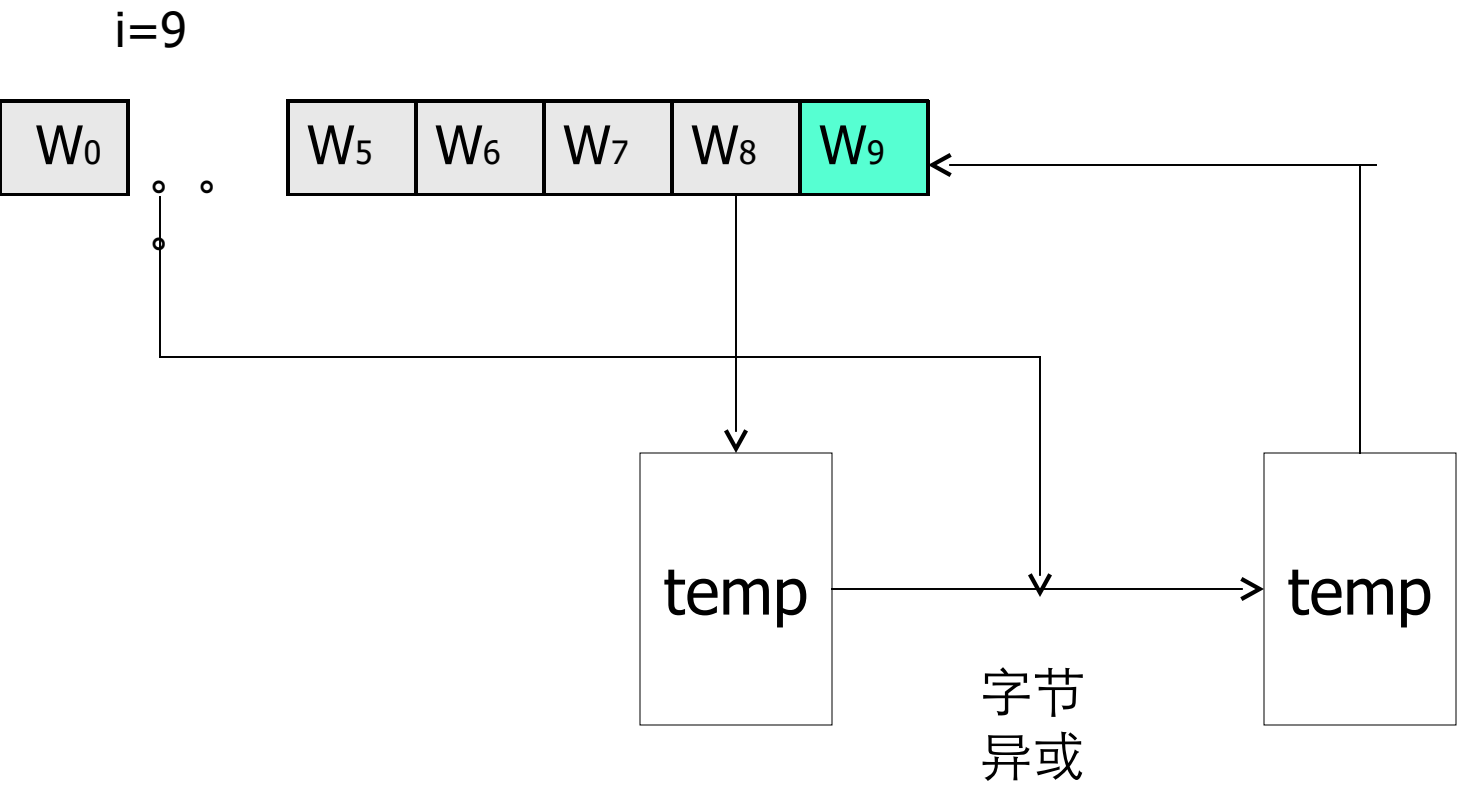
AES 算法的密钥编排算法

i=8　i/4==0



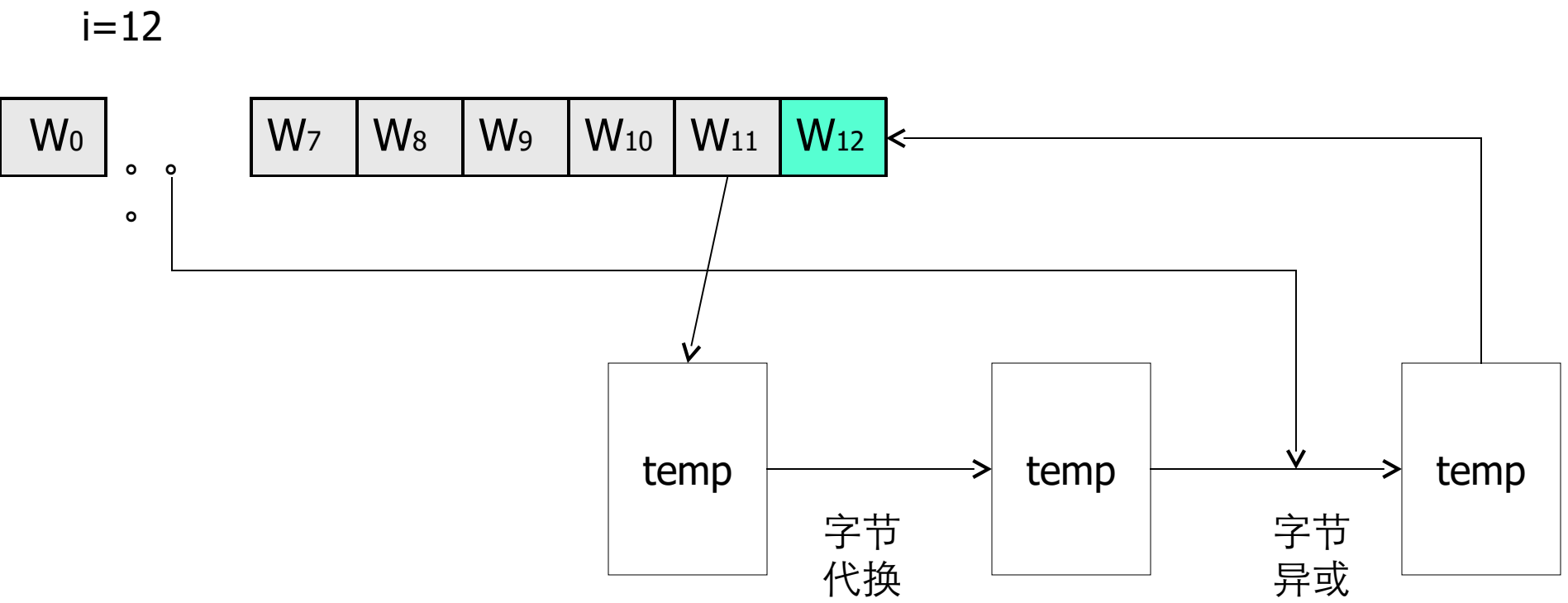
Nk=8

AES 算法的密钥编排算法



Nk=8

AES 算法的密钥编排算法



Nk=8

AES 算法的密钥编排算法

Rcon[i/Nk] 为轮常数，其值与 Nk 无关，定义为（字节用十六进制表示，同时理解为 GF(2⁸) 上的元素）：

Rcon [i]=(RC[i], ‘00’, ‘00’, ‘00’)

其中 RC[i] 是 GF(2⁸) 中值为 x^{i-1} 的元素，因此

RC[1] =1（即‘01’）

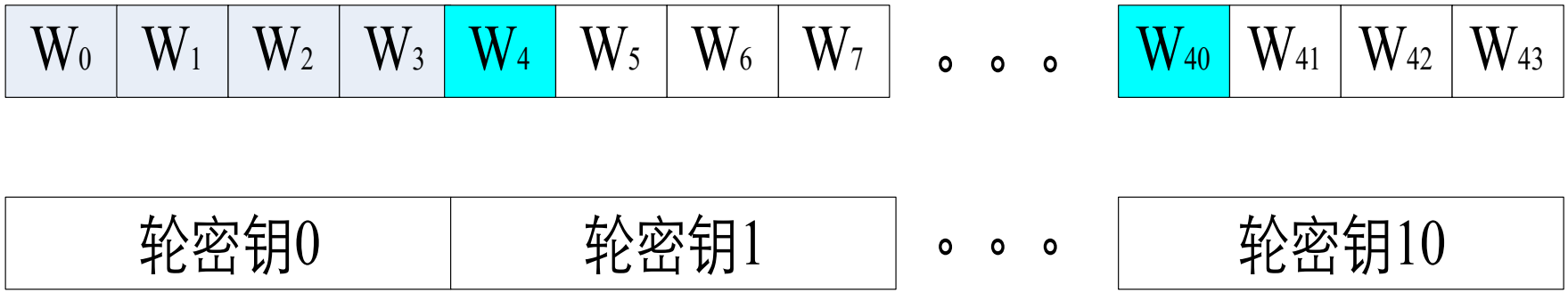
RC[2] = x（即‘02’）

RC[i]=x·RC[i-1]= x^{i-1}

AES 算法的密钥编排算法

2） 轮密钥选取

轮密钥 i （即第 i 个轮密钥）由轮密钥缓冲字 W[Nb* i] 到 W[Nb*(i+1)] 给出：



Nb=4 及 Nk=4 时的密钥扩展与轮密钥选取

AES 算法的密钥编排算法

W_0	W_1	W_2	W_3	W_4	W_5	W_6	W_7	。。。	W_{48}	W_{49}	W_{50}	W_{51}

轮密钥0	轮密钥1	。。。	轮密钥12

Nb=4 及 Nk=6 时的密钥扩展与轮密钥选取

W_0	W_1	W_2	W_3	W_4	W_5	W_6	W_7	W_8	。。。	W_{56}	W_{57}	W_{58}	W_{59}

轮密钥0	轮密钥1	。。。	轮密钥14

Nb=4 及 Nk=8 时的密钥扩展与轮密钥选取

本节主要内容

- AES 算法的整体结构
- AES 算法的轮函数
- AES 算法的密钥编排算法
- **AES 的解密变换**
- DES 的扩散和 AES 的扩散

AES 的解密变换

AES 解密运算是加密运算的逆运算，其中轮函数的逆为：

- 1）ByteSub 的逆变换由代换表的逆表做字节代换，也可通过如下两步实现： 首先进行仿射变换的逆变换，再求每一字节在 GF(2^8) 上逆元 .

- 2） 行移位运算的逆变换是循环右移，位移量与左移时相同 .

AES 的解密变换

- 3) 列混合运算的逆运算是类似的，即每列都用一个特定的多项式 d(x) 相乘， d(x) 满足

$$(03x^3+01x^2+01x+02)(d(x)=01$$

由此可得

$$d(x)=0Bx^3+0Dx^2+09x+0E$$

- 4) 密钥加运算的逆运算是其自身。

主要知识点小结

- AES 算法的整体结构

- AES 算法的轮函数

THE END !