现代密码学

Modern Cryptography

张方国

中山大学计算机学院

Office: Room 305, IM School Building

E-mail: isszhfg@mail.sysu.edu.cn

HomePage: https://cse.sysu.edu.cn/content/2460

第十讲 分组密码：AES

- AES
- 分组密码工作模式

AES

- 1997年1月，美国NIST向全世界密码学界发出征集21世纪高级加密标准（AES——Advanced Encryption Standard）算法的公告，并成立了AES标准工作研究室，1997年4月15日的例会制定了对AES的评估标准。
- **AES的标准提纲：**（1）AES是公开的；（2）AES为单钥体制分组密码；（3）AES的密钥长度可变，可按需要增大；（4）AES适于用软件和硬件实现；（5）AES可以自由地使用，或按符合美国国家标准（ANST）策略的条件使用；（6）满足以上要求的AES算法，需按下述条件判断优劣：a. 安全性，b. 计算效率， c. 内存要求， d. 使用简便性，e. 灵活性。

AES

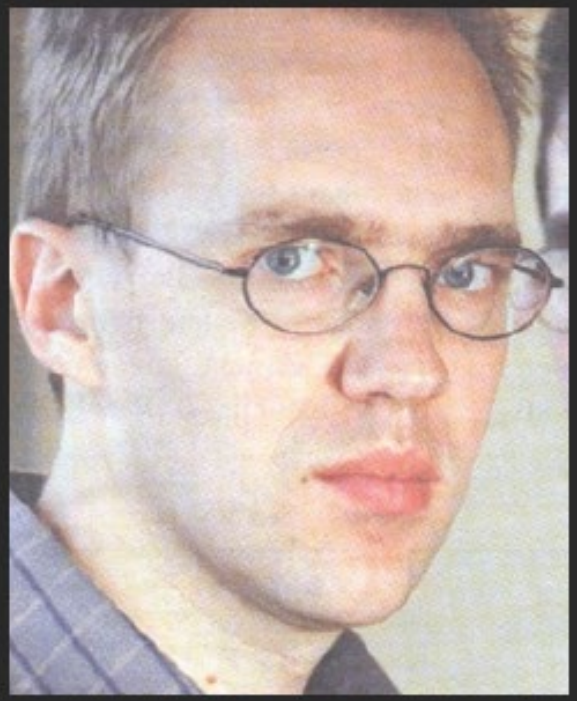
- 1998年4月15日全面征集AES算法的工作结束。1998年8月20日举行了首届AES讨论会，对涉及14个国家的密码学家所提出的候选AES算法进行了评估和测试，初选并公布了15个被选方案，供大家公开讨论。

15个候选算法有：CAST-256，RC-6，CRYPTON-128，DEAL-128，FROG，简易布丁密码，LOKI-97，MAGENTA，MARS，Vaudenay的抗相关快速密码，RIJNDAEL，SAFER+，SERPENT，E-2，TWOFISH。这些算法设计思想新颖，技术水平先进，算法的强度都超过3-DES，实现速度快于3-DES。

- 1999年8月9日NIST宣布第二轮筛选出的5个候选算法为：MARS(C.Burwick等,IBM），RC6™（R. Rivest等,RSA Lab.)，RIJNDEAL(J. Daemen,比)，SERPENT(R. Anderson等，英、以、挪威)，TWOFISH(B. Schiener)。

- 2000年4月13日，第三次AES会议上，对这5个候选算法的各种分析结果进行了讨论。
- 2000年10月，由比利时的Joan Daemen和Vincent Rijmen提出的算法最终胜出。
- 2001年11月，NIST完成了评估并发布了最终标准(FIPS PUB 197)，选择Rijndeal作为AES算法。

John Daemen
*1965



Vincent Rijmen
*1970

- Belgians
- KU Leuven

- 1997: Rijndael

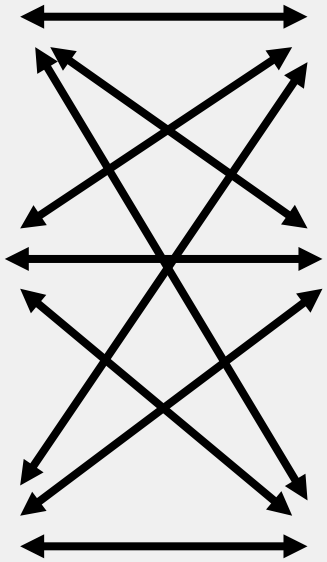
AES的设计原则

- 能抵抗所有已知的攻击；
- 在各种平台上易于实现，速度快；
- 设计简单。

Rijndael是一个分组密码算法，其分组长度和密钥长度相互独立，都可以改变。

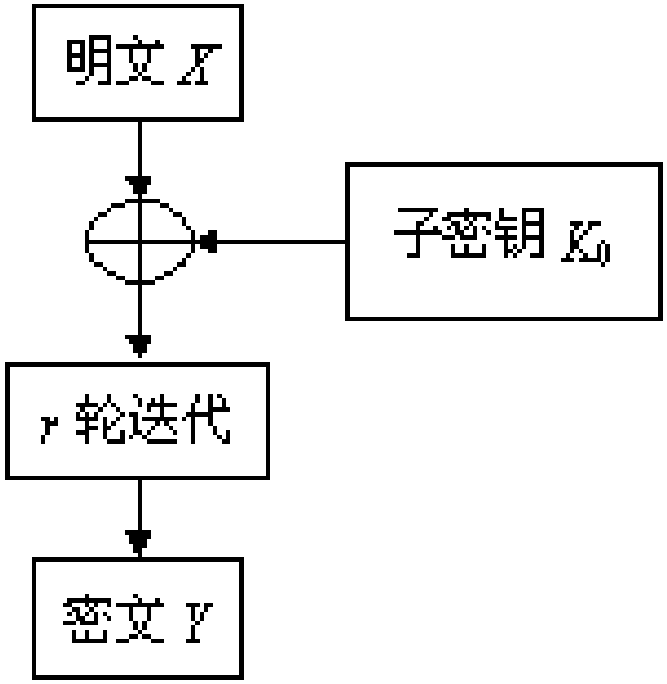
表 1. 分组长度和密钥长度的不同取值

分组长度（bit）
128
192
256

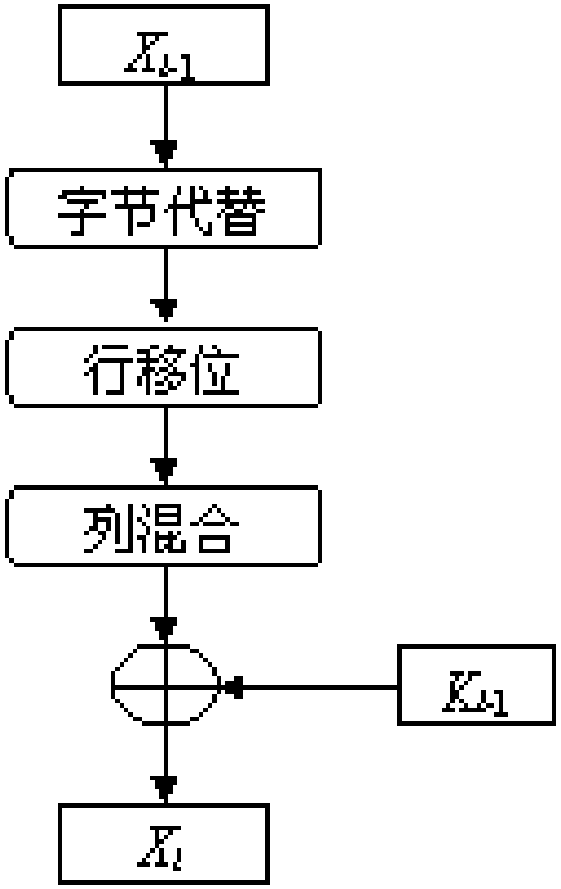


密钥长度（bit）
128
192
256

AES算法结构



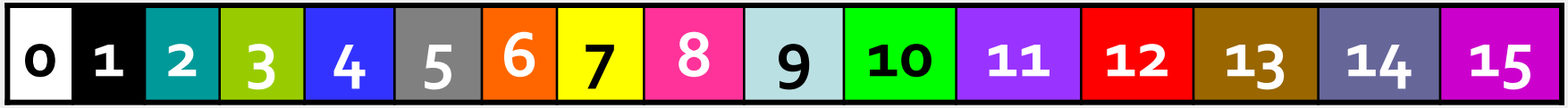
（a）AES 算法框图



（b） 一轮 AES 结构

AES 算法加密部分的实现

1. 明文分组和密钥的组织排列方式



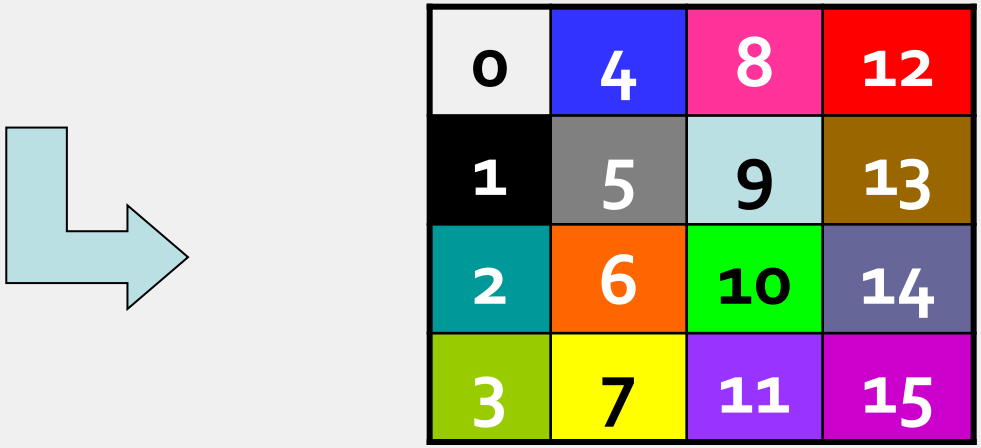


Fig 1.1. 以明文分组为128bits为例组成的阵列

Fig 1.2. 以明文分组（或密钥）为128bits、192bits 、256bits为例组成的阵列

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

0	4	8	12	16	20
1	5	9	13	17	21
2	6	10	14	18	22
3	7	11	15	19	23

0	4	8	12	16	20	24	28
1	5	9	13	17	21	25	29
2	6	10	14	18	22	26	30
3	7	11	15	19	23	27	31

一些相关的的术语定义和表示

- 状态（State）：密码运算的中间结果称为状态。
- State的表示：状态用以字节为基本构成元素的矩阵阵列来表示，该阵列有4行，列数记为Nb。Nb=分组长度（bits）÷ 32 Nb可以取的值为4，6，8，对应的分组长度为128， 192， 256 bits。
- 密码密钥（Cipher Key）的表示： Cipher Key类似地用一个4行的矩阵阵列来表示，列数记为Nk。Nk=密钥长度（bits）÷32 Nk可以取的值为4，6，8，对应的密钥长度为128， 192，256 bits。

Fig 1.3. 当Nb=6时的状态和Nk=4时的密钥布局

$a_{0,0}$	$a_{0,1}$	$a_{0,2}$	$a_{0,3}$	$a_{0,4}$	$a_{0,5}$
$a_{1,0}$	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	$a_{1,4}$	$a_{1,5}$
$a_{2,0}$	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	$a_{2,4}$	$a_{2,5}$
$a_{3,0}$	$a_{3,1}$	$a_{3,2}$	$a_{3,3}$	$a_{3,4}$	$a_{3,5}$

$K_{0,0}$	$K_{0,1}$	$K_{0,2}$	$K_{0,3}$
$K_{1,0}$	$K_{1,1}$	$K_{1,2}$	$K_{1,3}$
$K_{2,0}$	$K_{2,1}$	$K_{2,2}$	$K_{2,3}$
$K_{3,0}$	$K_{3,1}$	$K_{3,2}$	$K_{3,3}$

Nb = 6

Block Length = 192 bits

Nk = 4

Key Length = 128 bits

Fig 1.4. 分组长度和密钥长度均为128 bits时的Rijndael加密算法框图

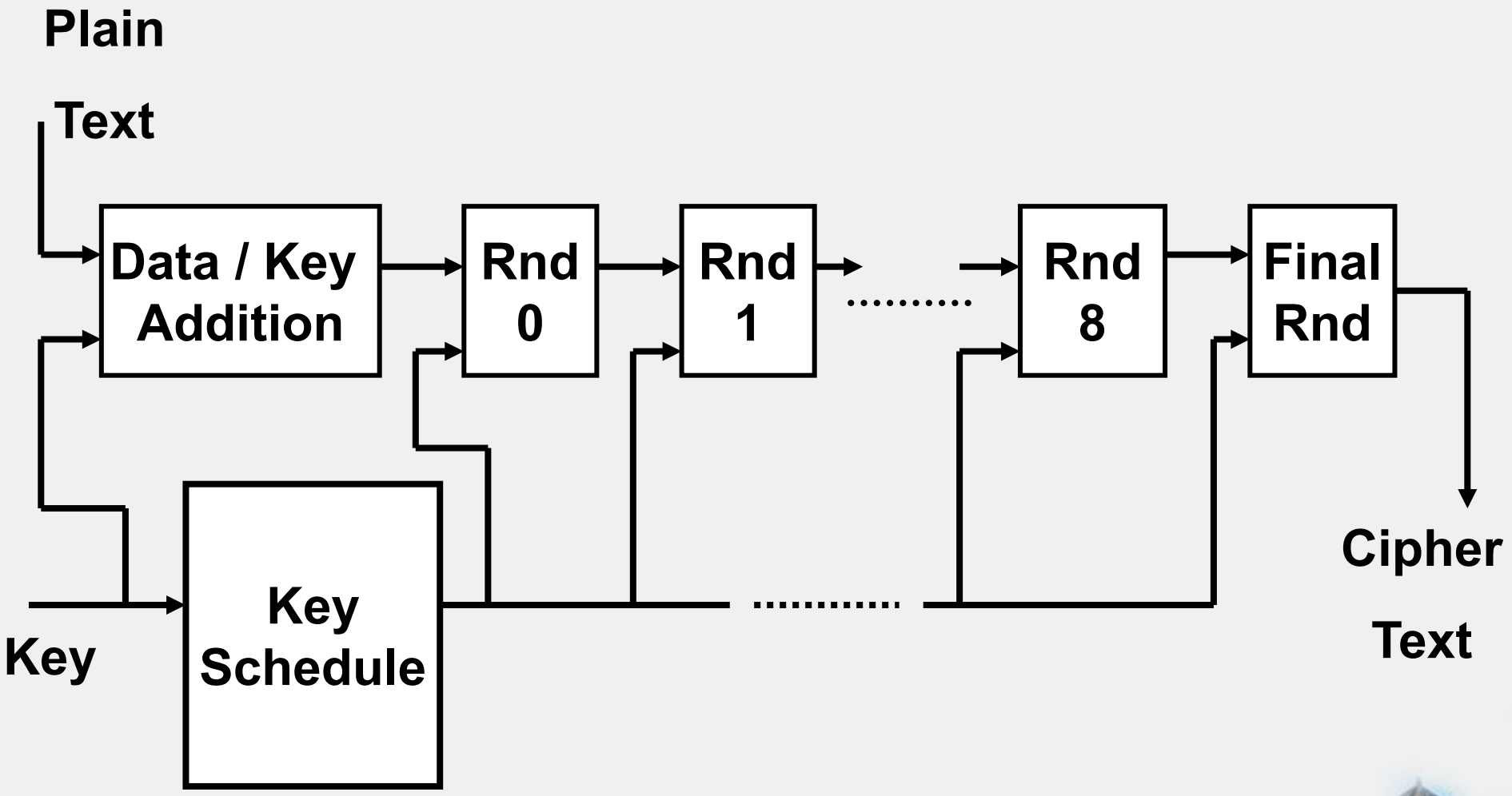
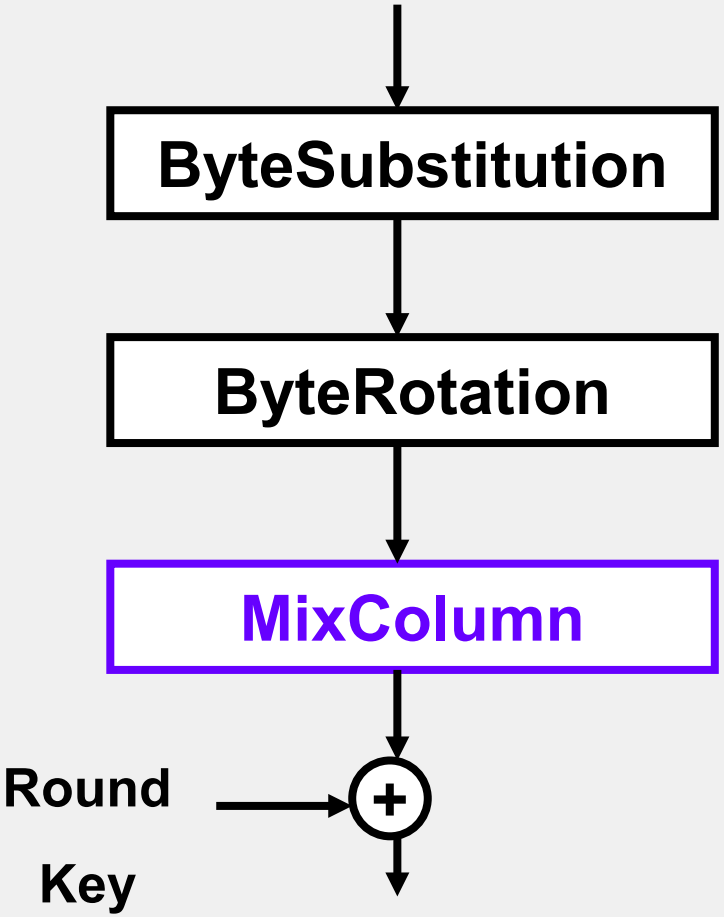


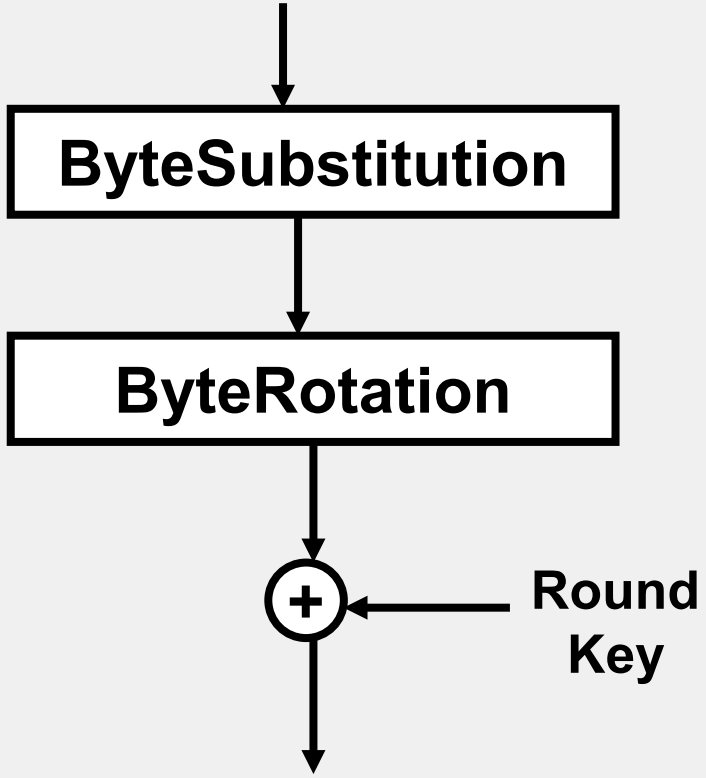
表 1.2. 圈数（Round）的不同取值

圈数（Round）	Block Length=128	Block Length=192	Block Length=256
Key Length=128	10	12	14
Key Length=192	12	12	14
Key Length=256	14	14	14

Fig 1.5. Rijndael Round的构成



一般的圈变换



最后一圈的圈变换

用伪代码表示的Rijndael圈变换

一般的圈变换

```
Round(State, RoundKey)
{
    ByteSubstitution;
    ByteRotation;
    MixColumn;
    AddRounKey;
    }
```

结尾圈的变换

```
FinalRound(State,
RoundKey)
{
    ByteSubstituion;
    ByteRotation;
    AddRoundKey;
    }
```

ByteSubstitution(字节替代)

ByteSubstitution是一个非线性的字节替代，独立地在每个状态字节上进行运算。它包括两个变换。

1. 在有限域**GF(2^8)**上求乘法逆，‘00’映射到它自身。
2. 在**GF(2)**上进行下面的仿射变换：

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

ByteSubstitution(字节替代)

Algorithm 4.4: SUBBYTES($a_7a_6a_5a_4a_3a_2a_1a_0$)

external FIELDINV, BINARYTOFIELD, FIELDTOBINARY
$z \leftarrow$ BINARYTOFIELD$(a_7a_6a_5a_4a_3a_2a_1a_0)$
if $z \neq 0$
  then $z \leftarrow$ FIELDINV(z)
$(a_7a_6a_5a_4a_3a_2a_1a_0) \leftarrow$ FIELDTOBINARY(z)
$(c_7c_6c_5c_4c_3c_2c_1c_0) \leftarrow (01100011)$
comment: In the following loop, all subscripts are to be reduced modulo 8
for $i \leftarrow 0$ **to** 7
  do $b_i \leftarrow (a_i + a_{i+4} + a_{i+5} + a_{i+6} + a_{i+7} + c_i) \bmod 2$
return $(b_7b_6b_5b_4b_3b_2b_1b_0)$

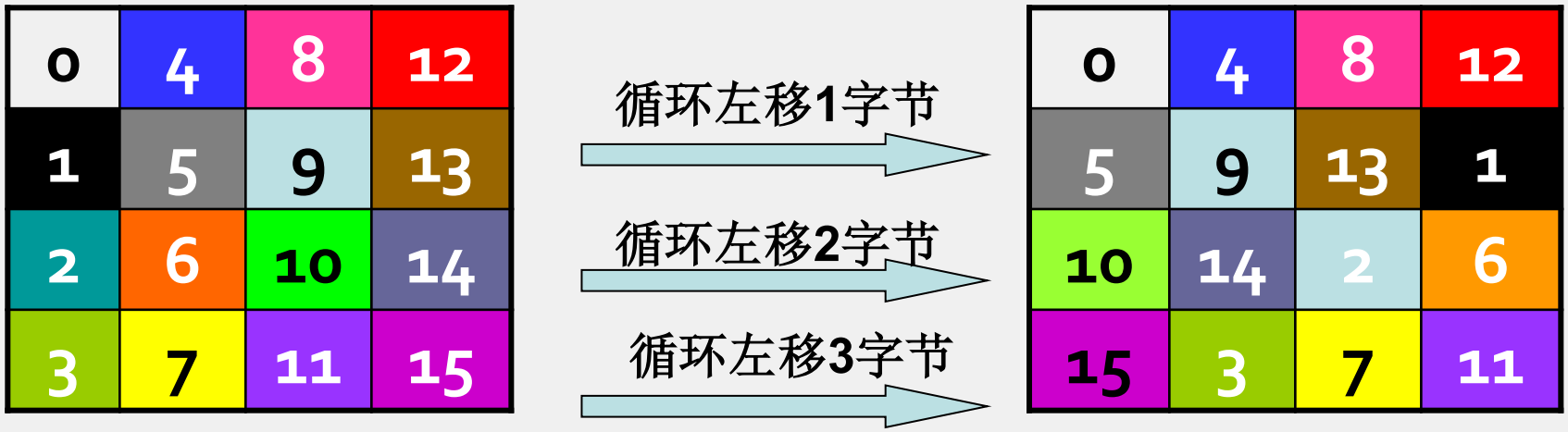
ByteRotation(字节移位)

在ByteRotation变换中，状态阵列的后3行循环移位不同的偏移量。第1行循环移位C1字节，第2行循环移位C2字节，第3行循环移位C3字节。

偏移量C1、C2、C3与分组长度Nb有关，如下表所示：

Nb	C_1	C_2	C_3
4	1	2	3
6	1	2	3
8	1	3	4

Fig 1.7. ByteRotation



MixColumn(列混合)

将状态的列看作是有限域GF(2^8)上的多项式a(x)，与多项式c(x) = 03 x^3 + 01 x^2 + 01 x +02相乘(模x^4+1)。

令b(x) = c(x) × a(x)，写成矩阵形式为：

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

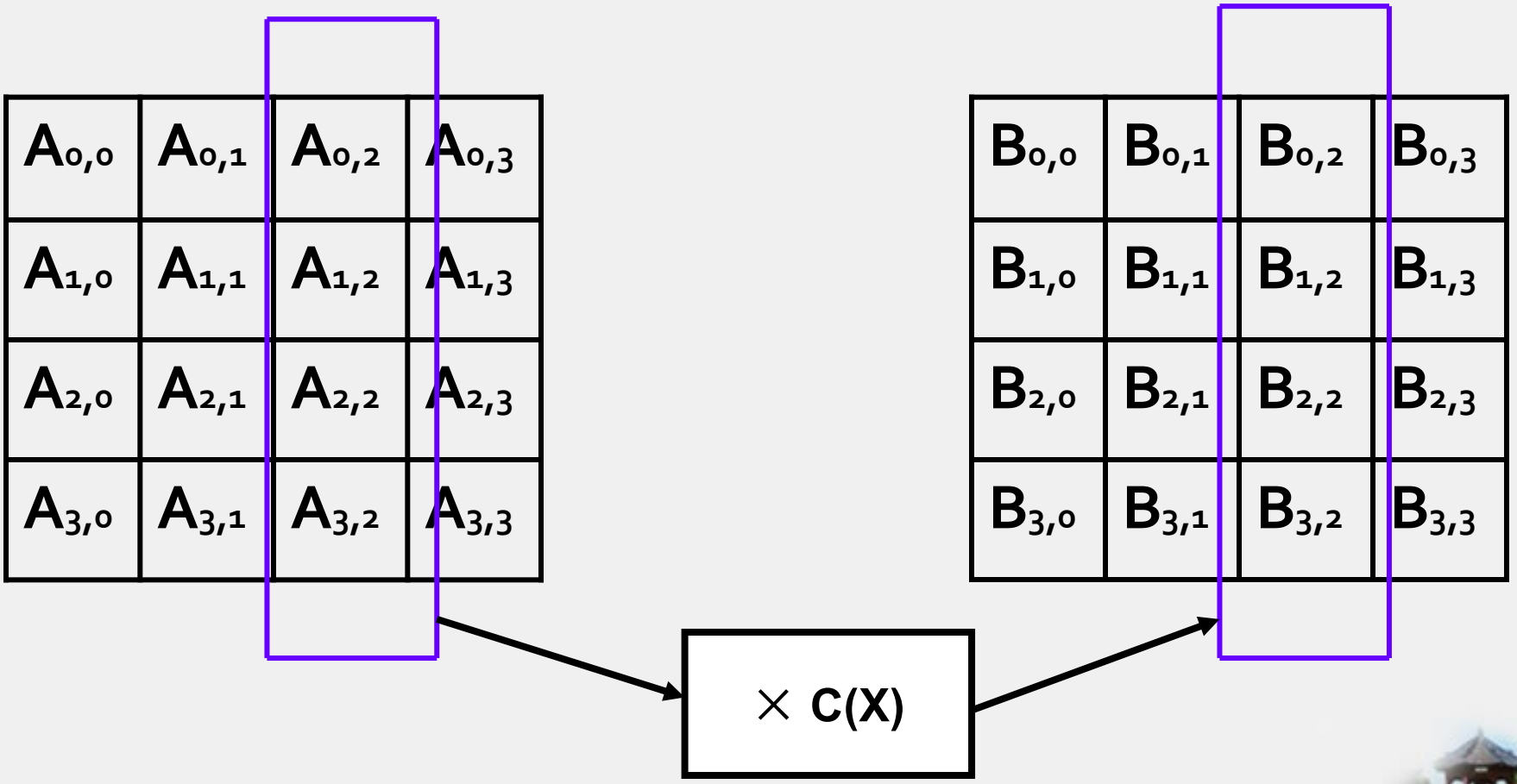
列混合

Algorithm 4.5: MIXCOLUMN(c)

external FIELDMULT, BINARYTOFIELD, FIELDTOBINARY
for $i \leftarrow 0$ **to** 3
  do $t_i \leftarrow$ BINARYTOFIELD($s_{i,c}$)
$u_0 \leftarrow$ FIELDMULT$(x, t_0) \oplus$ FIELDMULT$(x+1, t_1) \oplus t_2 \oplus t_3$
$u_1 \leftarrow$ FIELDMULT$(x, t_1) \oplus$ FIELDMULT$(x+1, t_2) \oplus t_3 \oplus t_0$
$u_2 \leftarrow$ FIELDMULT$(x, t_2) \oplus$ FIELDMULT$(x+1, t_3) \oplus t_0 \oplus t_1$
$u_3 \leftarrow$ FIELDMULT$(x, t_3) \oplus$ FIELDMULT$(x+1, t_0) \oplus t_1 \oplus t_2$
for $i \leftarrow 0$ **to** 3
  do $s_{i,c} \leftarrow$ FIELDTOBINARY(u_i)

Fig 1.8. MixColumn

这一运算作用在每一列上



AddRoundKey(圈密钥加)

$A_{0,0}$	$A_{0,1}$	$A_{0,2}$	$A_{0,3}$
$A_{1,0}$	$A_{1,1}$	$A_{1,2}$	$A_{1,3}$
$A_{2,0}$	$A_{2,1}$	$A_{2,2}$	$A_{2,3}$
$A_{3,0}$	$A_{3,1}$	$A_{3,2}$	$A_{3,3}$

$\oplus$

$K_{0,0}$	$K_{0,1}$	$K_{0,2}$	$K_{0,3}$
$K_{1,0}$	$K_{1,1}$	$K_{1,2}$	$K_{1,3}$
$K_{2,0}$	$K_{2,1}$	$K_{2,2}$	$K_{2,3}$
$K_{3,0}$	$K_{3,1}$	$K_{3,2}$	$K_{3,3}$

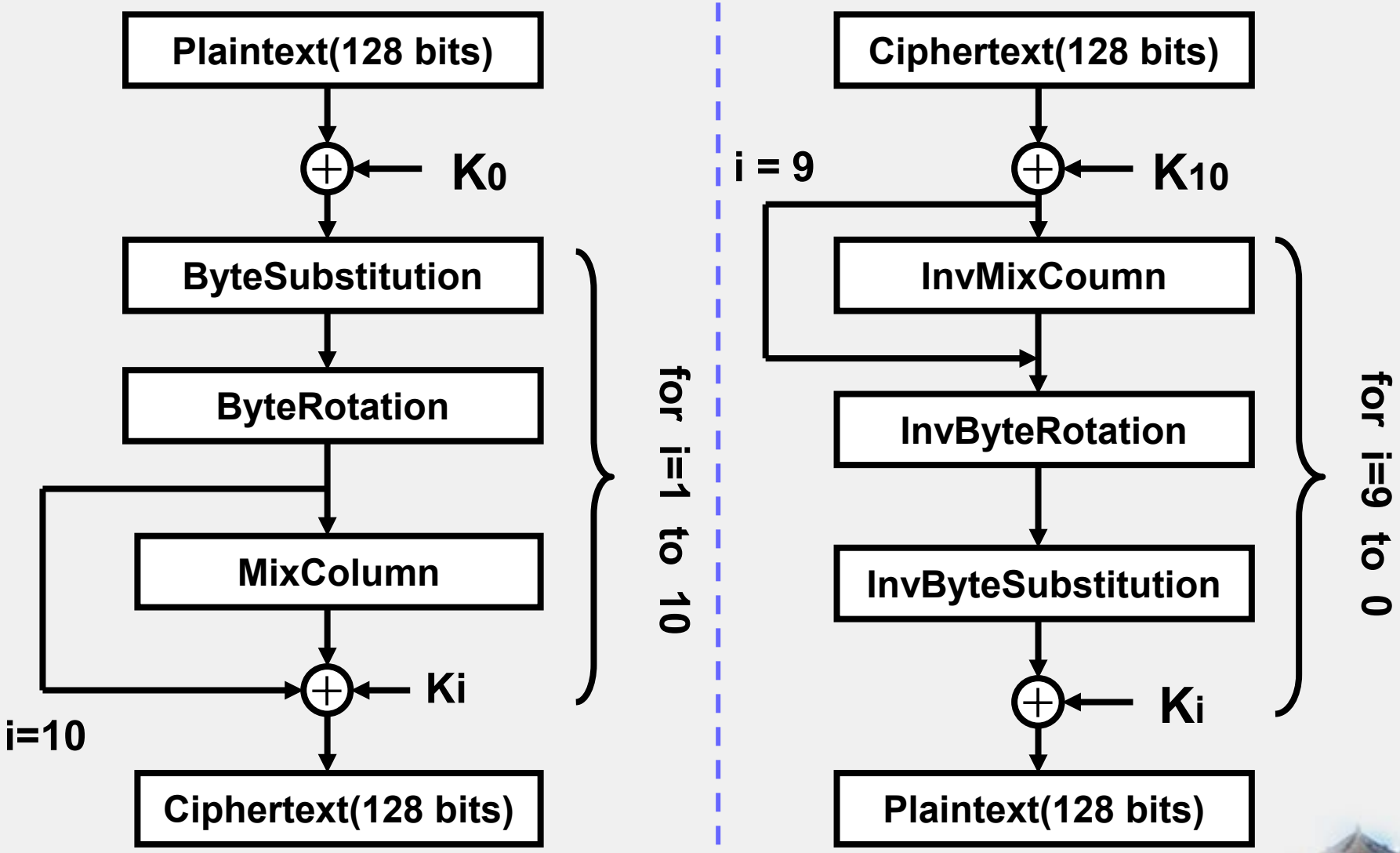
$=$

$B_{0,0}$	$B_{0,1}$	$B_{0,2}$	$B_{0,3}$
$B_{1,0}$	$B_{1,1}$	$B_{1,2}$	$B_{1,3}$
$B_{2,0}$	$B_{2,1}$	$B_{2,2}$	$B_{2,3}$
$B_{3,0}$	$B_{3,1}$	$B_{3,2}$	$B_{3,3}$

$$A_{3,3} + K_{3,3} = B_{3,3} \pmod 2$$

Fig 1.7. Rijndael加密及解密的标准结构

Block , Key Length = 128 bits



加密　　解密

Rijndael密码的构成

Rijndael密码由以下三个部分组成：

- 一个初始圈密钥相加；
- Rnd－1圈；
- 一个结尾圈。

用伪代码表示的Rijndael加密算法

```
Rijndael ( State,  CipherKey )
{
  KeyExpansion ( CipherKey,  ExpandedKey );
  AddRoundKey ( State,  ExpandedKey );
  For ( i=1;  i<Rnd;  i++ )
        Round ( State,  ExpandedKey + Nb*i );
  FinalRound ( State,  ExpandedKey + Nb*Rnd );
  }
```

提前进行密钥扩展后的Rijndael加密算法描述

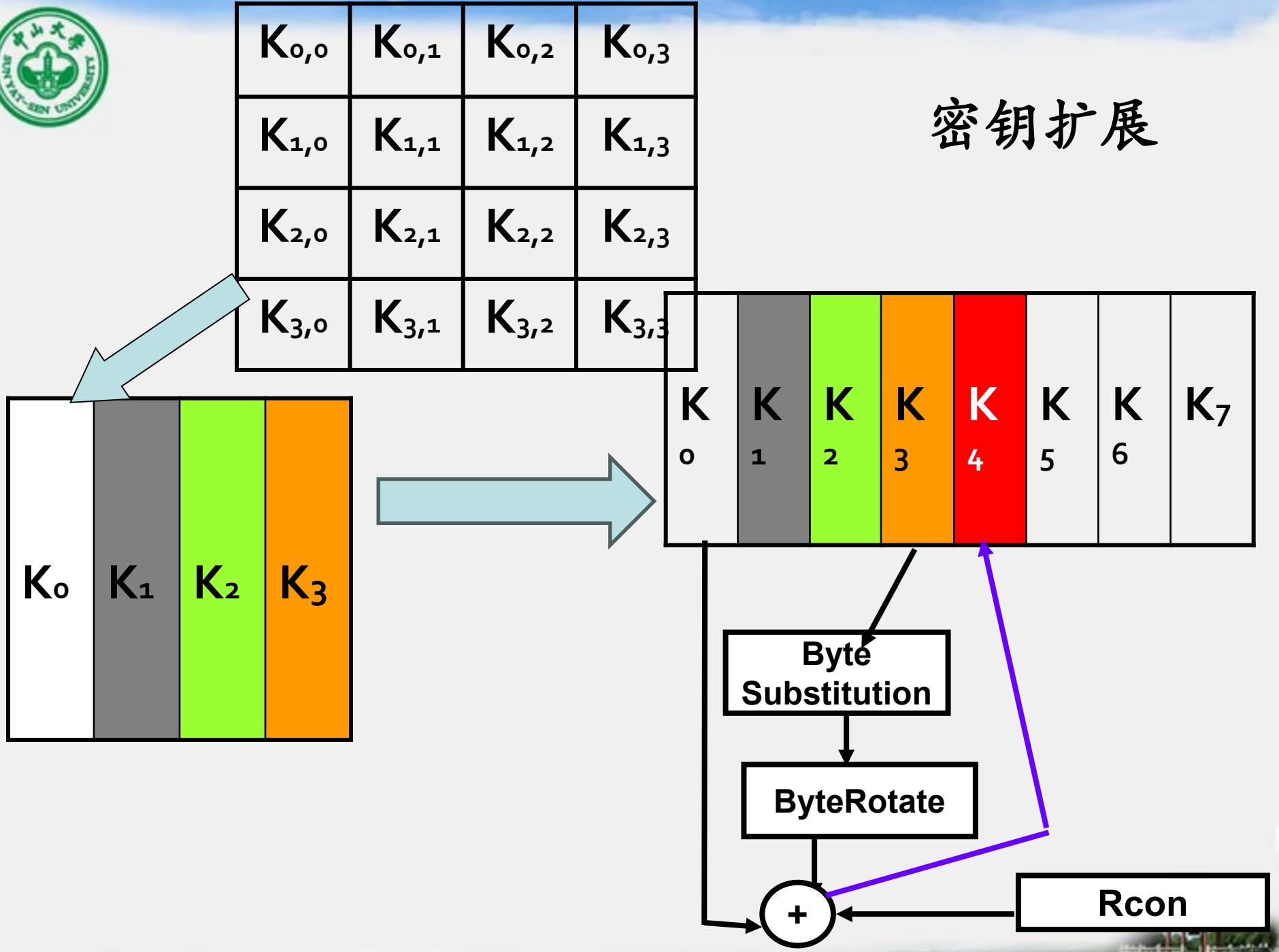
```
Rijndael ( State,  ExpandedKey )
{
  AddRoundKey ( State,  ExpandedKey );
  For ( i=1;  i<Rnd;  i++ )
        Round ( State,  ExpandedKey + Nb*i );
  FinalRound ( State,  ExpandedKey + Nb*Rnd );
  }
```

AES 算法的密钥调度

密钥调度包括两个部分：密钥扩展和圈密钥选取。

- 密钥bit的总数＝分组长度×（圈数Round＋1）例如当分组长度为128bits和圈数Round为10时，圈密钥长度为128×(10＋1)＝1408bits。
- 将密码密钥扩展成一个扩展密钥。
- 从扩展密钥中取出圈密钥：第一个圈密钥由扩展密钥的第一个Nb个4字节字，第二个圈密钥由接下来的Nb个4字节字组成，以此类推。

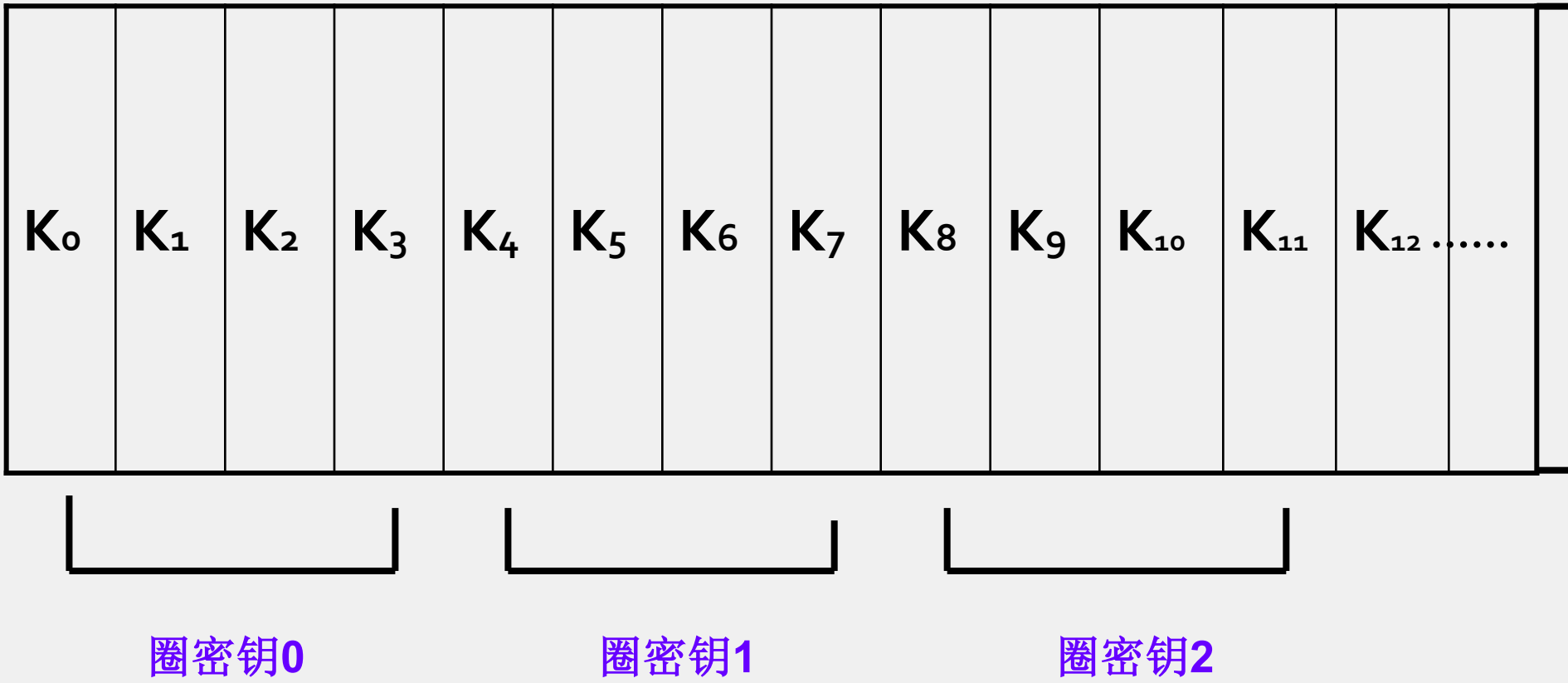
密钥扩展


K0,0 K0,1 K0,2 K0,3
K1,0 K1,1 K1,2 K1,3
K2,0 K2,1 K2,2 K2,3
K3,0 K3,1 K3,2 K3,3
K0 K1 K2 K3
K0 K1 K2 K3 K4 K5 K6 K7
Byte Substitution
ByteRotate
+
Rcon

Algorithm 4.6: KEYEXPANSION(*key*)

external ROTWORD, SUBWORD
$RCon[1] \leftarrow \texttt{01000000}$
$RCon[2] \leftarrow \texttt{02000000}$
$RCon[3] \leftarrow \texttt{04000000}$
$RCon[4] \leftarrow \texttt{08000000}$
$RCon[5] \leftarrow \texttt{10000000}$
$RCon[6] \leftarrow \texttt{20000000}$
$RCon[7] \leftarrow \texttt{40000000}$
$RCon[8] \leftarrow \texttt{80000000}$
$RCon[9] \leftarrow \texttt{1B000000}$
$RCon[10] \leftarrow \texttt{36000000}$
for $i \leftarrow 0$ **to** 3
  do $w[i] \leftarrow (key[4i], key[4i+1], key[4i+2], key[4i+3])$
for $i \leftarrow 4$ **to** 43
  do
$temp \leftarrow w[i-1]$
if $i \equiv 0 \pmod 4$
  then $temp \leftarrow \text{SUBWORD}(\text{ROTWORD}(temp)) \oplus RCon[i/4]$
$w[i] \leftarrow w[i-4] \oplus temp$
return $(w[0], \ldots, w[43])$

圈密钥选取



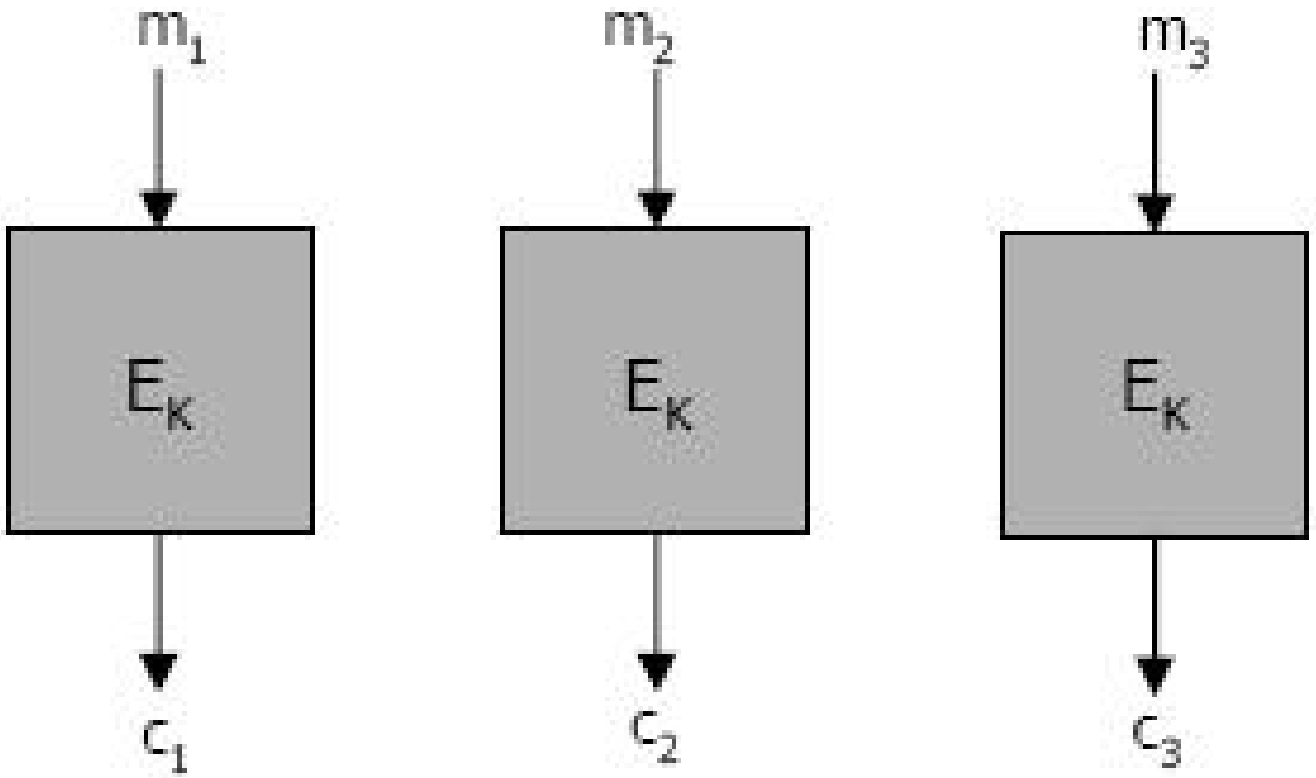
分组密码运行模式

分组密码每次加密的明文数据量是固定的分组长度*n*，而实用中待加密消息的数据量是不定的。另一方面，即使有了安全的分组密码算法，也需要采用适当的工作模式来隐蔽明文的统计特性、数据的格式等，以提高整体的安全性，降低删除、重放、插入和伪造成功的机会。所采用的工作模式应当力求简单、有效和易于实现。

美国NSB在[FIPS PUB 74和81]中规定了DES的四种基本工作模式：**电子码本**(ECB)，**密码分组链接**(CBC)，**密码反馈**(CFB)，**输出反馈**(OFB)。

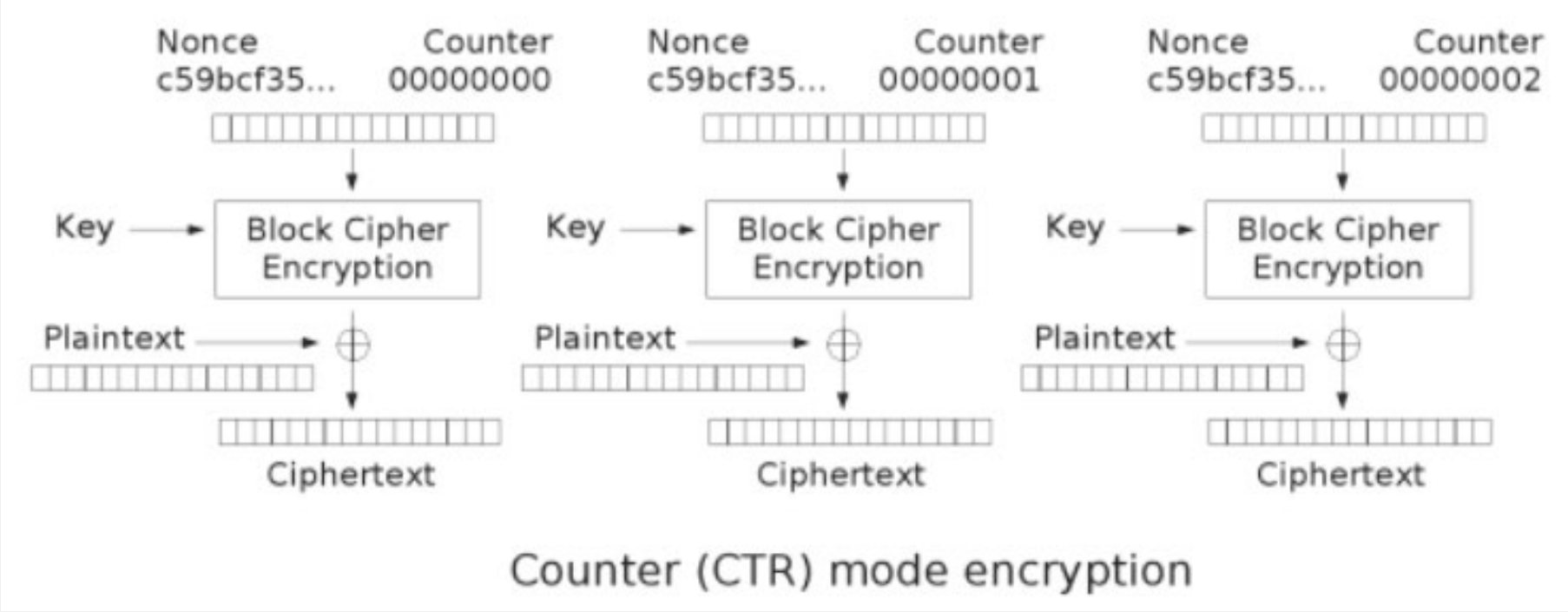
800-38A中NIST将推荐模式扩展为5个，**记数器模式**(CTR).

Electronic Code Book Mode (ECB)



- 缺点：在给定的密钥下同一明文组总产生同样的密文组，这会暴露明文数据的格式和统计特征。在ECB模式，所有这些特征都将被反映到密文中，使密码分析者可以对其进行统计分析、重传和代换攻击。

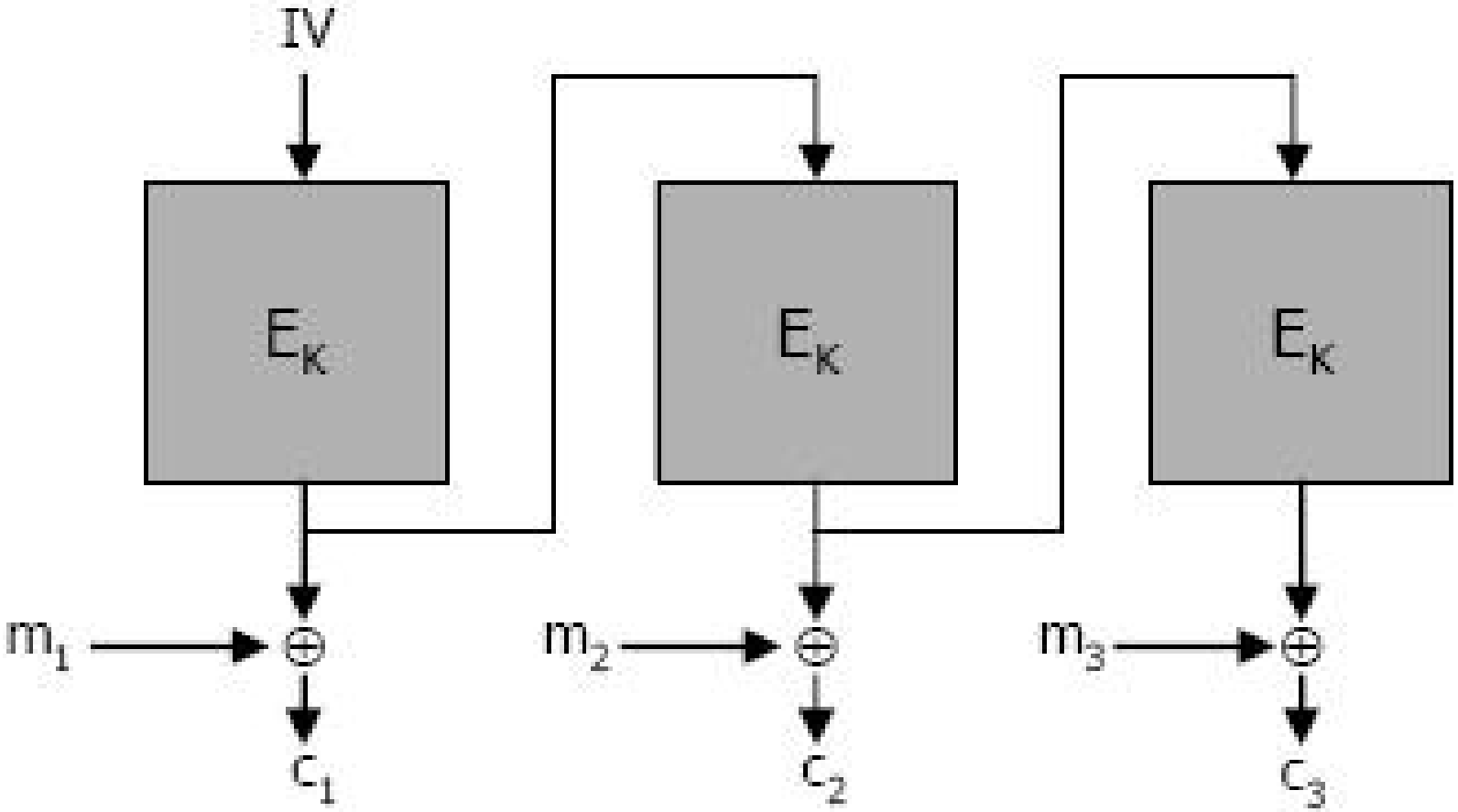
Counter (CTR) mode



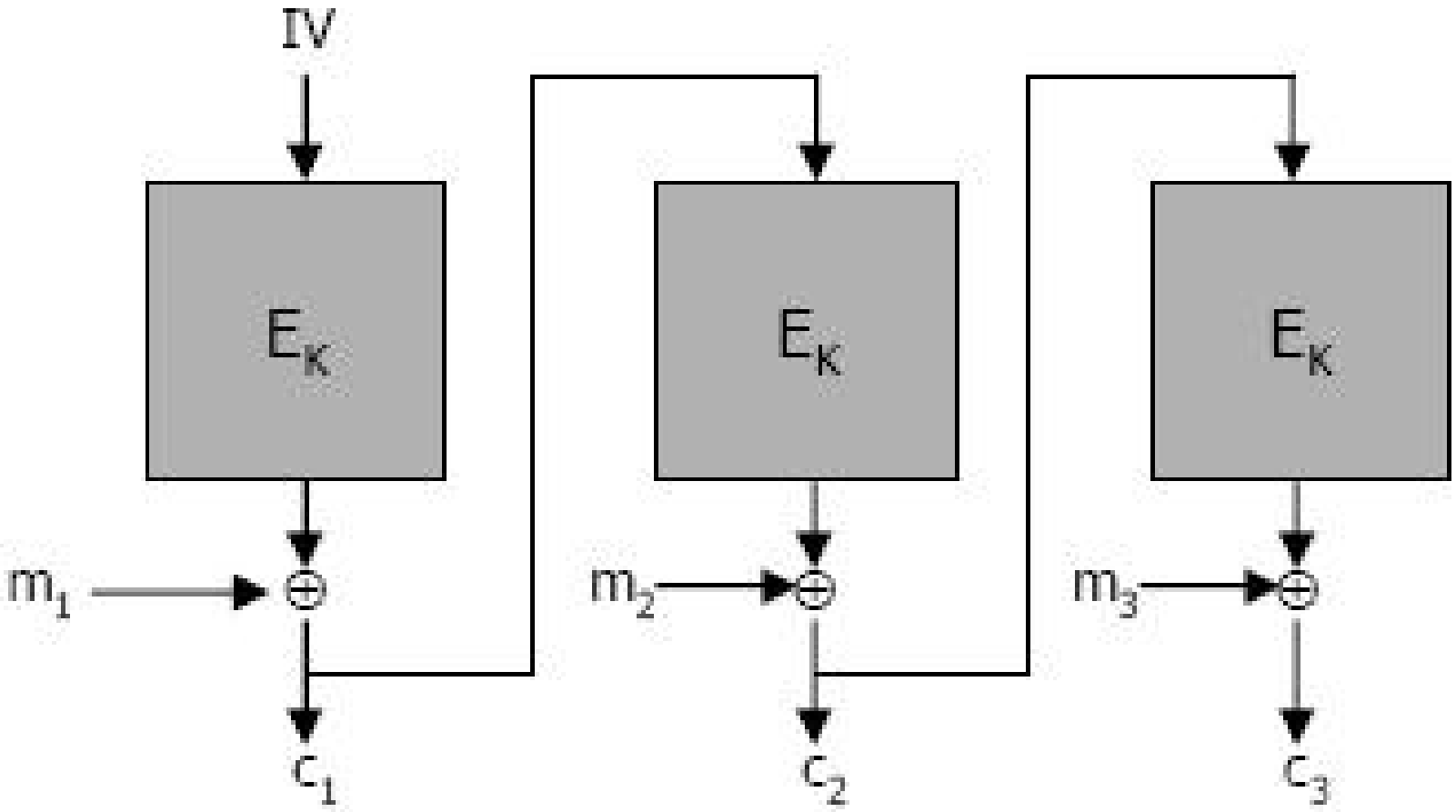
Counter (CTR) mode encryption

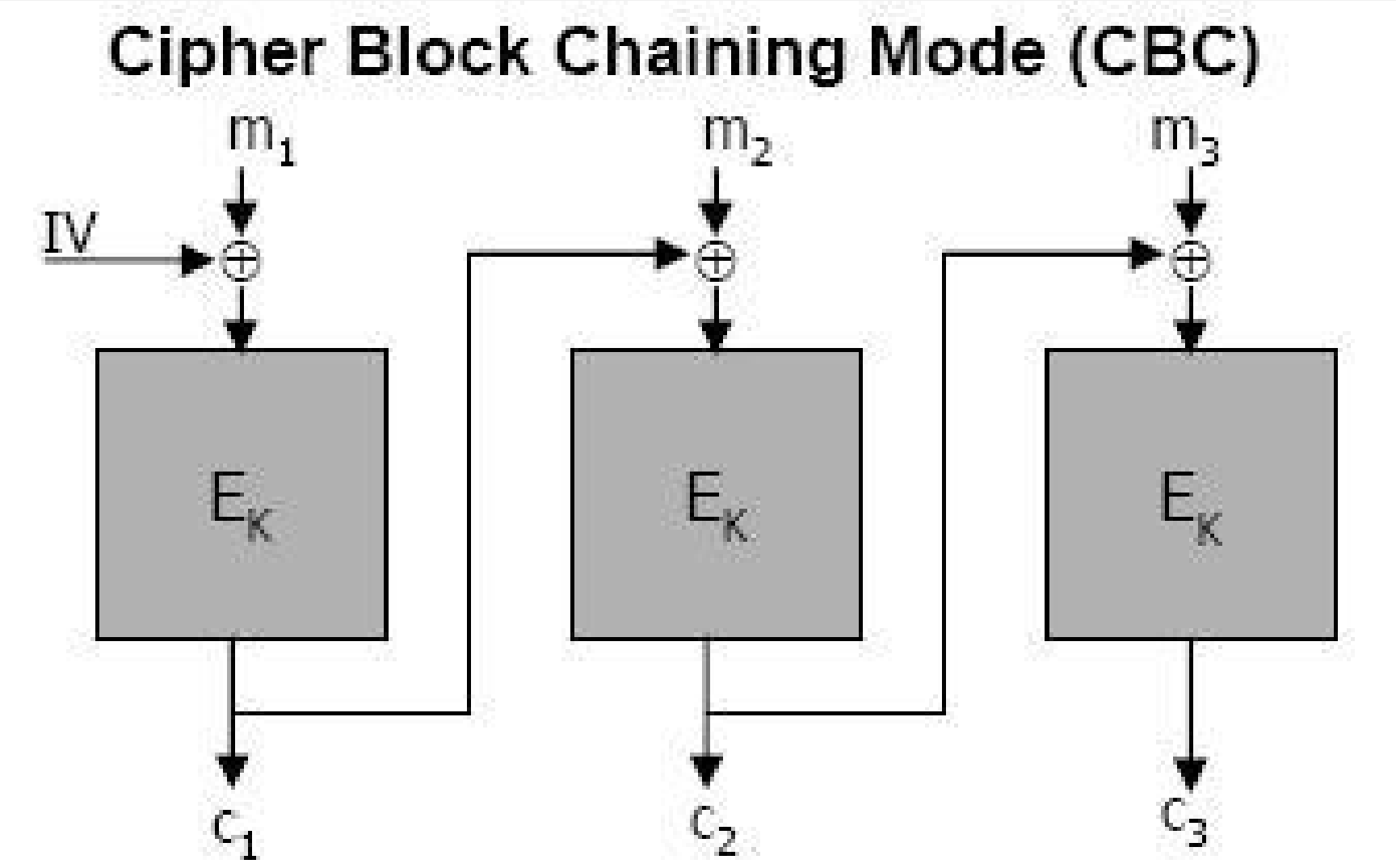
- can be precomputed and fully parallelized
- allows random access

Output Feedback Mode (OFB)



Cipher Feedback Mode (CFB)




Cipher Block Chaining Mode (CBC)
m1
m2
m3
IV
EK
EK
EK
c1
c2
c3

计数密码分组链接模式(CCM)

- 基本上就是计数模式和CBC模式的组合使用