密 码 学

第五讲　中国商用分组密码SM4与分组密码应用技术

王后珍

武汉大学国家网络安全学院

空天信息安全与可信计算教育部重点实验室

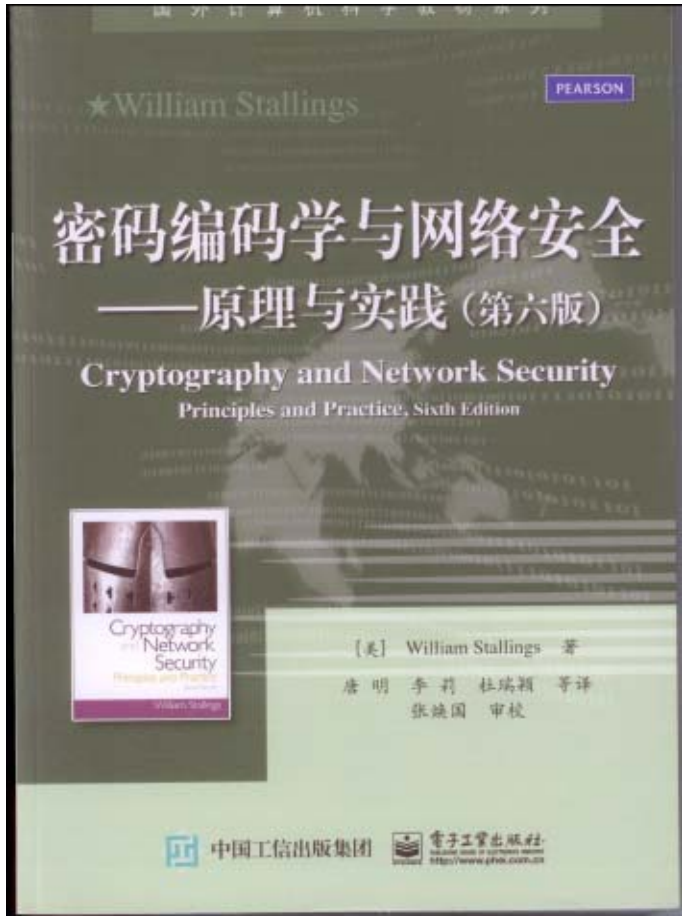
目 录

目 录

教材与主要参考书

教材

参考书



本讲内容

一、中国商用密码SM4

㈠ 我国商用密码的概况

㈡ 中国商用密码SM4的概况

㈢ SM4密码算法

㈣ SM4的可逆性与对合性

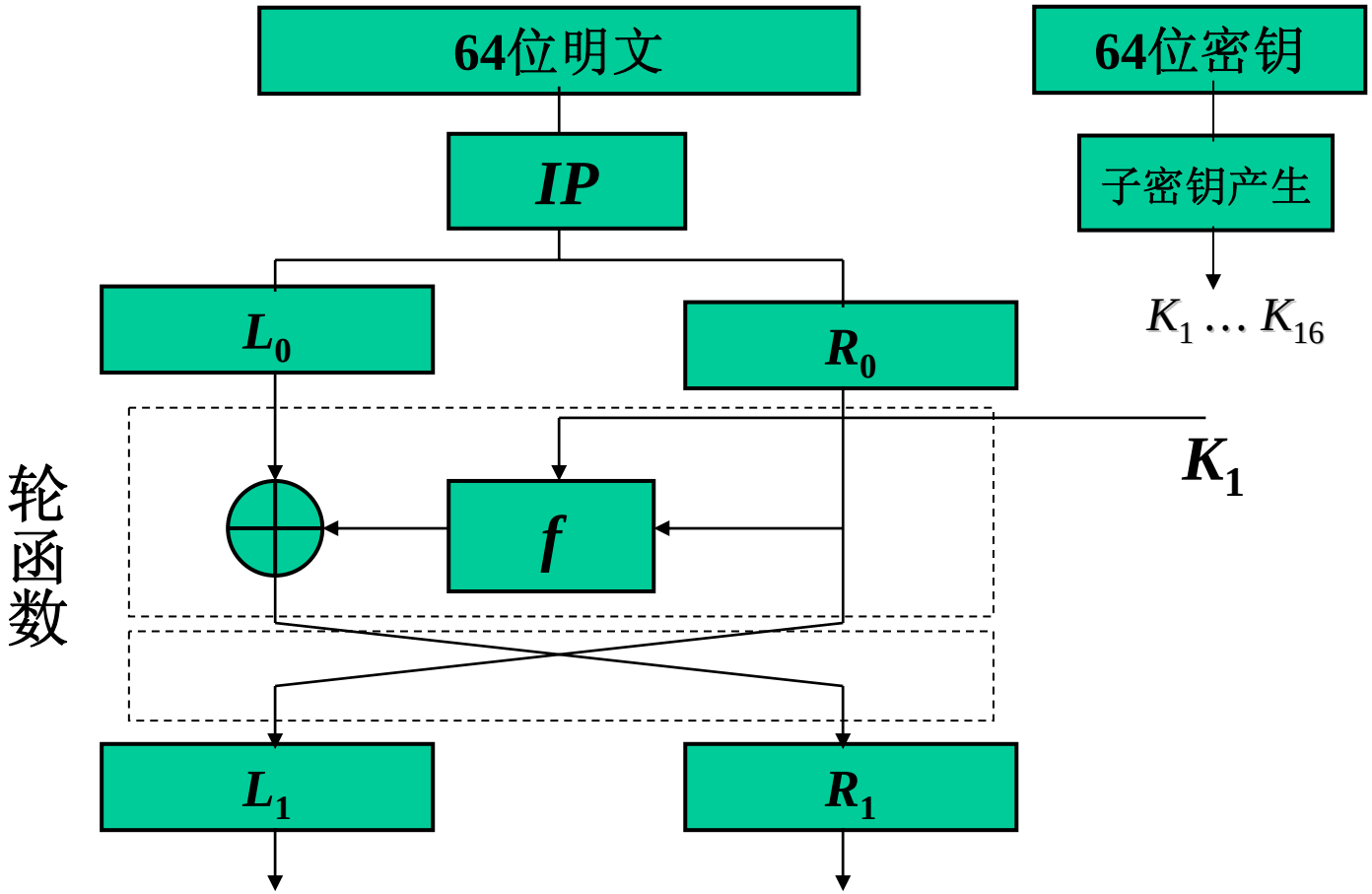
二、分组密码应用技术

㈠ 计算机数据的特殊性

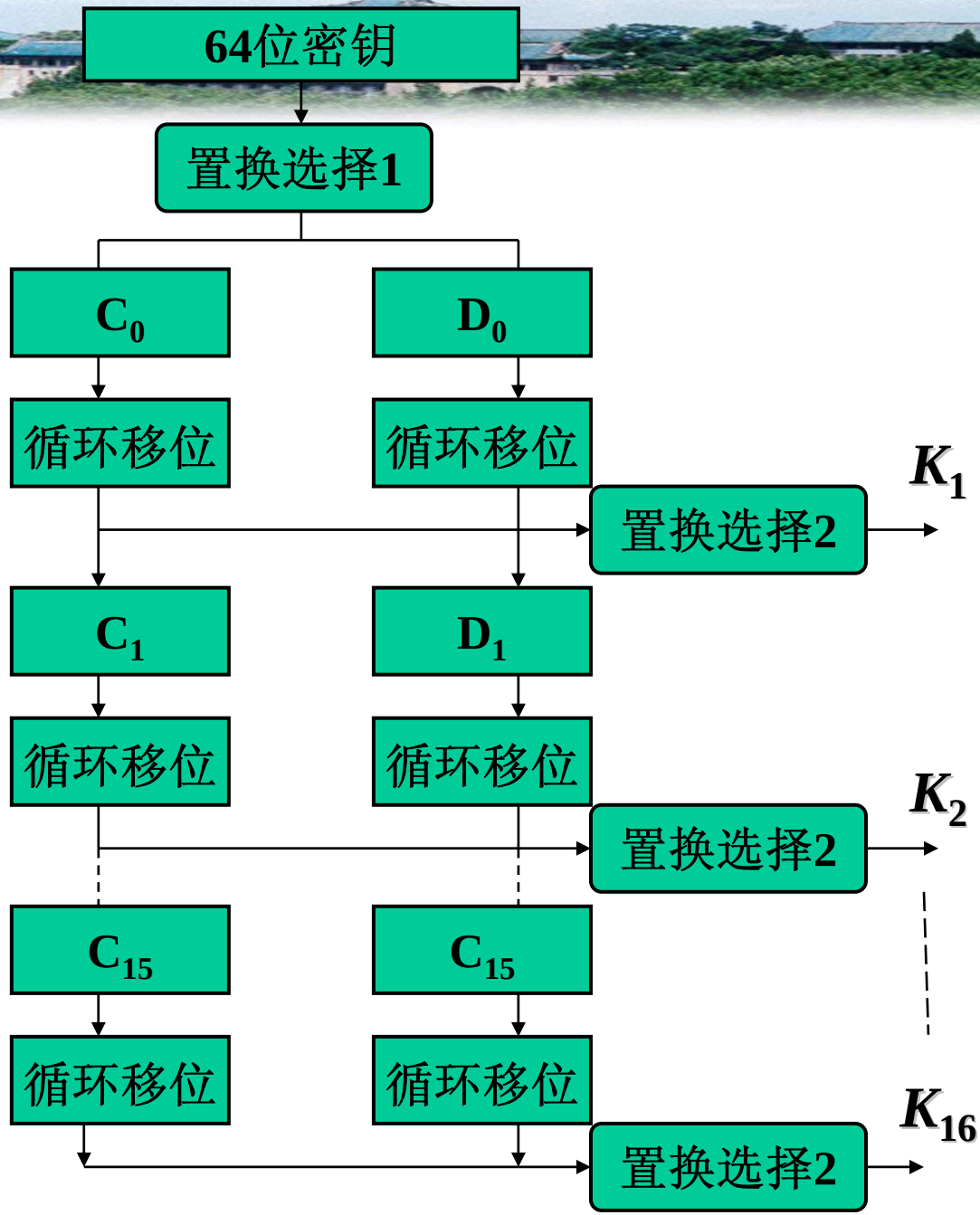
㈡ 分组密码的工作模式

㈢ 短快加密

DES加密算法回顾

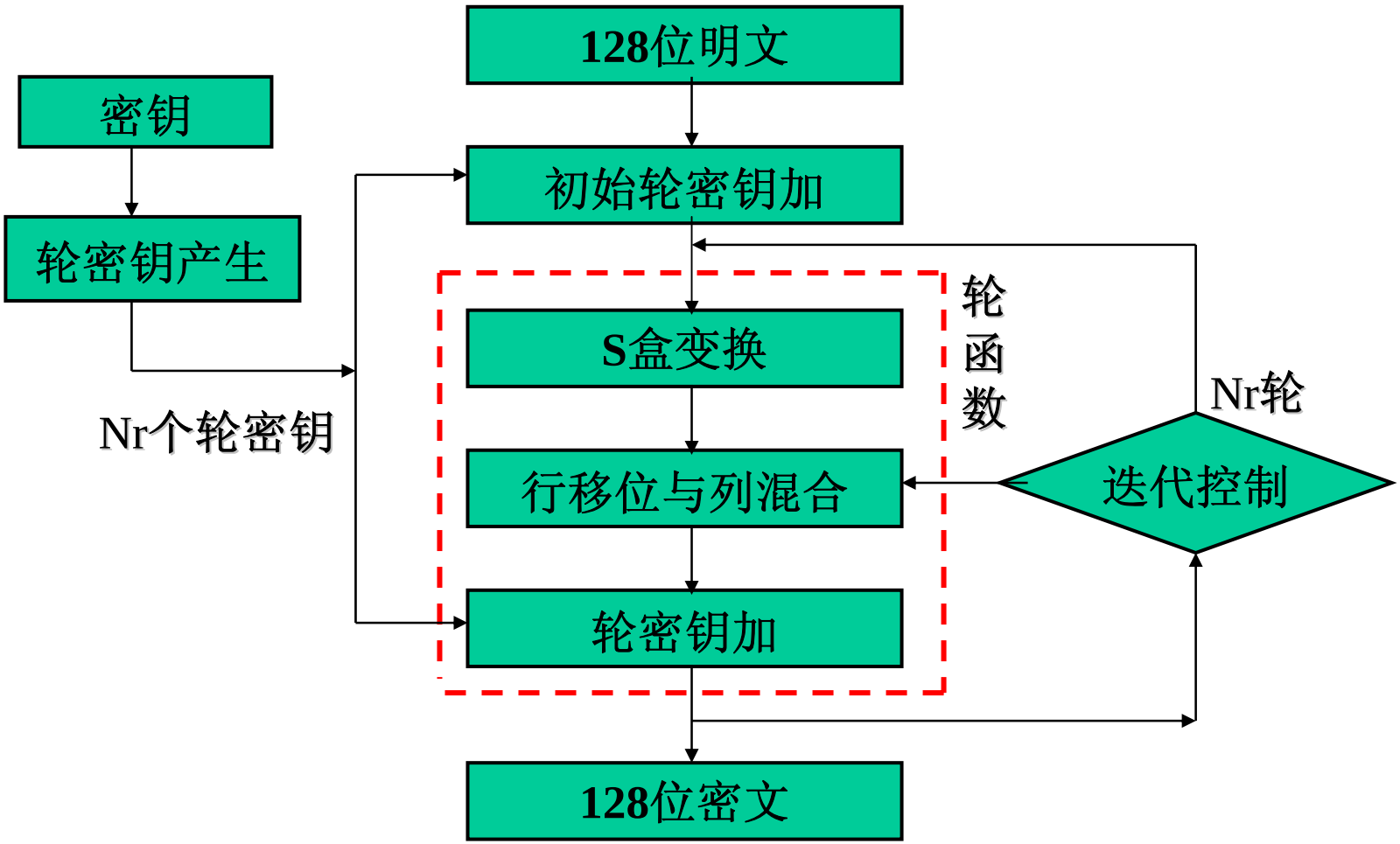


DES子密钥产生框图



武漢大学

AES加密算法回顾


128位明文
密钥
初始轮密钥加
轮密钥产生
轮函数
S盒变换
Nr轮
Nr个轮密钥
行移位与列混合
迭代控制
轮密钥加
128位密文

一、中国商用分组密码SM4

(一) 我国商用密码的概况

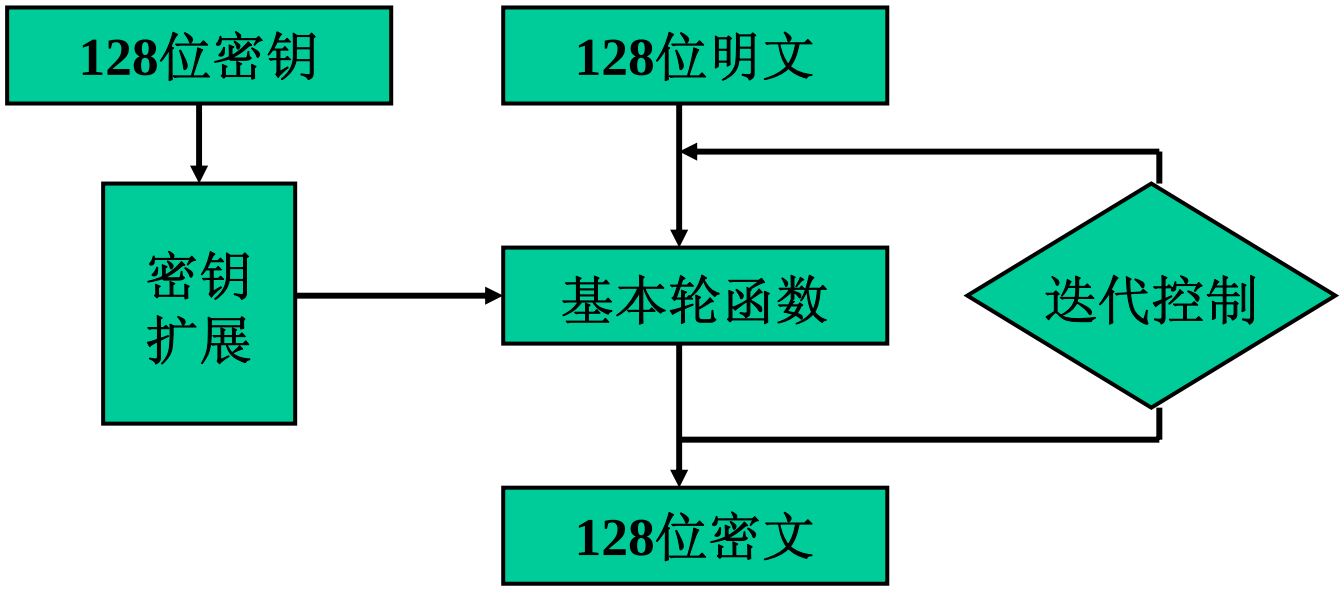
- 我国在密码技术方面具有优势
  - 密码理论
  - 密码分析
- 长期以来不公开密码算法，只提供密码芯片
  - 少数专家设计，难免有疏漏
  - 难于标准化，应用成本高，不利于推广应用
- 近年来我国陆续公布了商用密码算法
  - **2006年2月公布了分组密码SM4**
  - **2011年2月公布了椭圆曲线密码SM2和杂凑算法SM3**
  - 我国商用密码管理更加科学化、与国际接轨
  - 促进了我国商用密码的发展和应用

(二) 中国商用密码SM4的概况

- 分组密码
  - 数据分组（明文，密文）长度=128位、密钥长度=128位
  - 数据处理单位：字节（ 8位），字（32位）
- 密码算法特点
  - 对合运算：解密算法与加密算法相同
  - 子密钥生成算法与加密算法结构类似
- 密码结构
  - 不是SP结构，也不是Feistel结构
  - 是非对称Feistel结构

(三) SM4密码算法

1、SM4 密码算法结构



(三) SM4密码算法

2、SM4 密码算法

(1)基本运算：

① 模2加： ⊕，32 比特异或运算

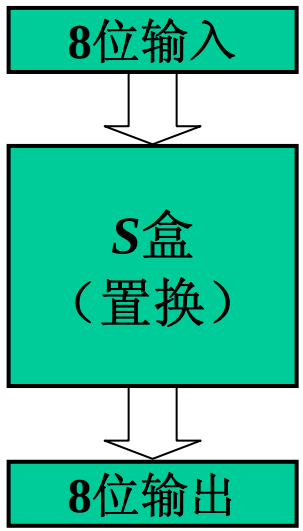
② 循环移位：<<< *i* ，把32位字循环左移*i* 位

(2)基本密码部件：

① 非线性字节变换*S*盒：起混淆作用

- 8位输入，8位输出。
- 本质上是 8位的非线性置换。
- 设输入为*a*，输出为*b*，*S*盒运算可表示为：

$$b=S\_Box(a)$$



(三) SM4密码算法

S盒数据表：

S 盒中数据均采用 16 进制表示。

高位 \ 低位	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	d6	90	e9	fe	cc	e1	3d	b7	16	b6	14	c2	28	fb	2c	05
1	2b	67	9a	76	2a	be	04	c3	aa	44	13	26	49	86	06	99
2	9c	42	50	f4	91	ef	98	7a	33	54	0b	43	ed	cf	ac	62
3	e4	b3	1c	a9	c9	08	e8	95	80	df	94	fa	75	8f	3f	a6
4	47	07	a7	fc	f3	73	17	ba	83	59	3c	19	e6	85	4f	a8
5	68	6b	81	b2	71	64	da	8b	f8	eb	0f	4b	70	56	9d	35
6	1e	24	0e	5e	63	58	d1	a2	25	22	7c	3b	01	21	78	87
7	d4	00	46	57	9f	d3	27	52	4c	36	02	e7	a0	c4	c8	9e
8	ea	bf	8a	d2	40	c7	38	b5	a3	f7	f2	ce	f9	61	15	a1
9	e0	ae	5d	a4	9b	34	1a	55	ad	93	32	30	f5	8c	b1	e3
a	1d	f6	e2	2e	82	66	ca	60	c0	29	23	ab	0d	53	4e	6f
b	d5	db	37	45	de	fd	8e	2f	03	ff	6a	72	6d	6c	5b	51
c	8d	1b	af	92	bb	dd	bc	7f	11	d9	5c	41	1f	10	5a	d8
d	0a	c1	31	88	a5	cd	7b	bd	2d	74	d0	12	b8	e5	b4	b0
e	89	69	97	4a	0c	96	77	7e	65	b9	f1	09	c5	6e	c6	84
f	18	f0	7d	ec	3a	dc	4d	20	79	ee	5f	3e	d7	cb	39	48

(三) SM4密码算法

- *S*盒的置换规则：
  - 以输入的高半字节为行号，低半字节为列号，行列交叉点处的数据即为输出。
  - 举例：设输入为“*ef*”，则行号为*e*，列号为*f*，于是*S*盒的输出值为表中第 *e* 行和第 *f* 列交叉点的值，即

$$Sbox(\text{‘}ef\text{’}) = \text{‘}84\text{’}$$

  - 说明：在主要密码学指标上达到最佳，与AES的*S*盒相当

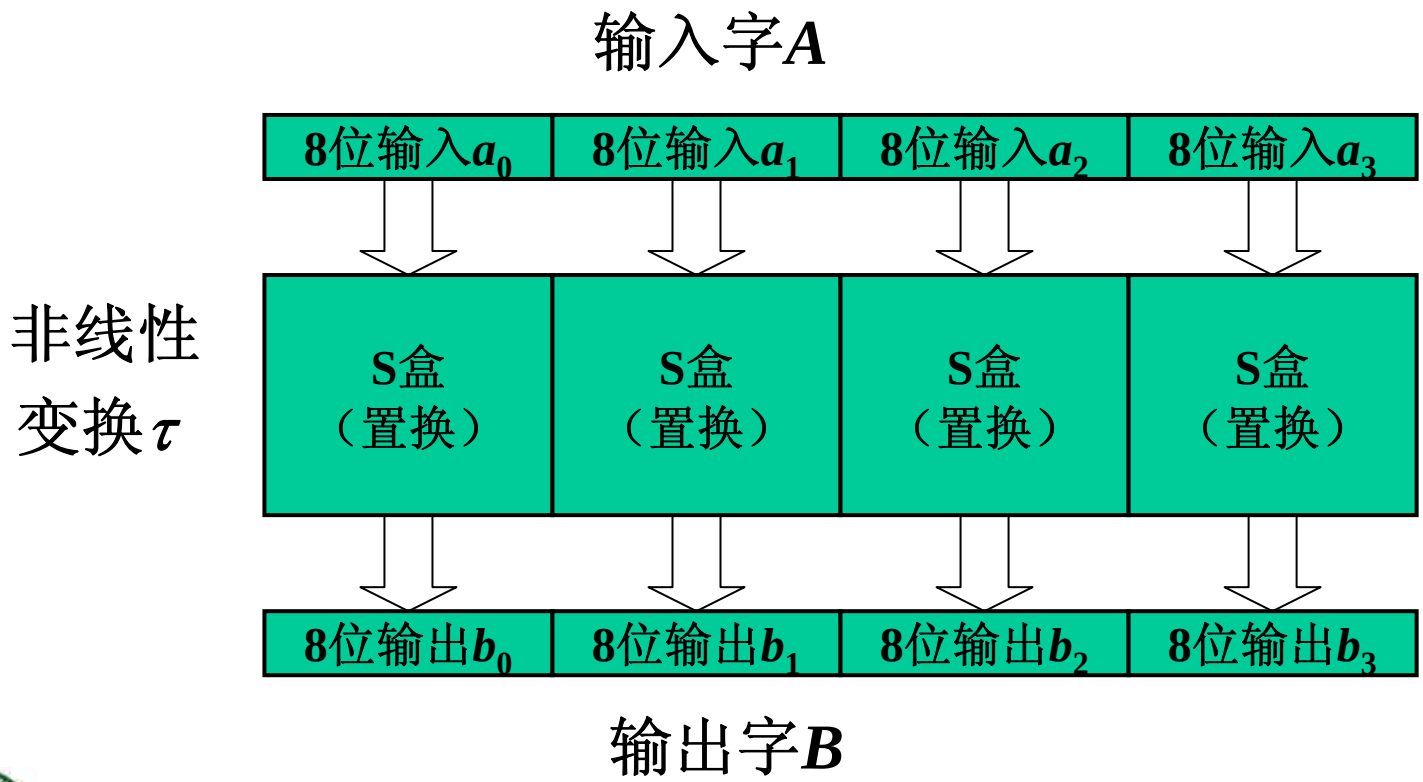
②非线性字变换τ：起混淆作用

- 4个S盒并行置换
- 设输入字$A=(a_0,a_1,a_2,a_3)$,输出字$B=(b_0,b_1,b_2,b_3)$,

$$B = \tau(A)=(S\_box(a_0), S\_box(a_1), S\_box(a_2), S\_box(a_3))$$

(三) SM4密码算法

②非线性变换τ: 32位字的非线性变换



(三) SM4密码算法

③字线性部件L变换：起扩散作用

- 32位输入，32位输出。
- 设输入为B，输出为C，表为：

$$C=L(B)$$

- 运算规则：

$$C=L(B)$$

$$=B\oplus(B<<<2)\oplus(B<<<10)\oplus(B<<<18)\oplus(B<<<24)$$

④字合成变换T：

- 由非线性变换 τ 和线性变换L 复合而成；

$$T(X)=L(\tau(X))。$$

先S盒变换，后L变换。

(三) SM4密码算法

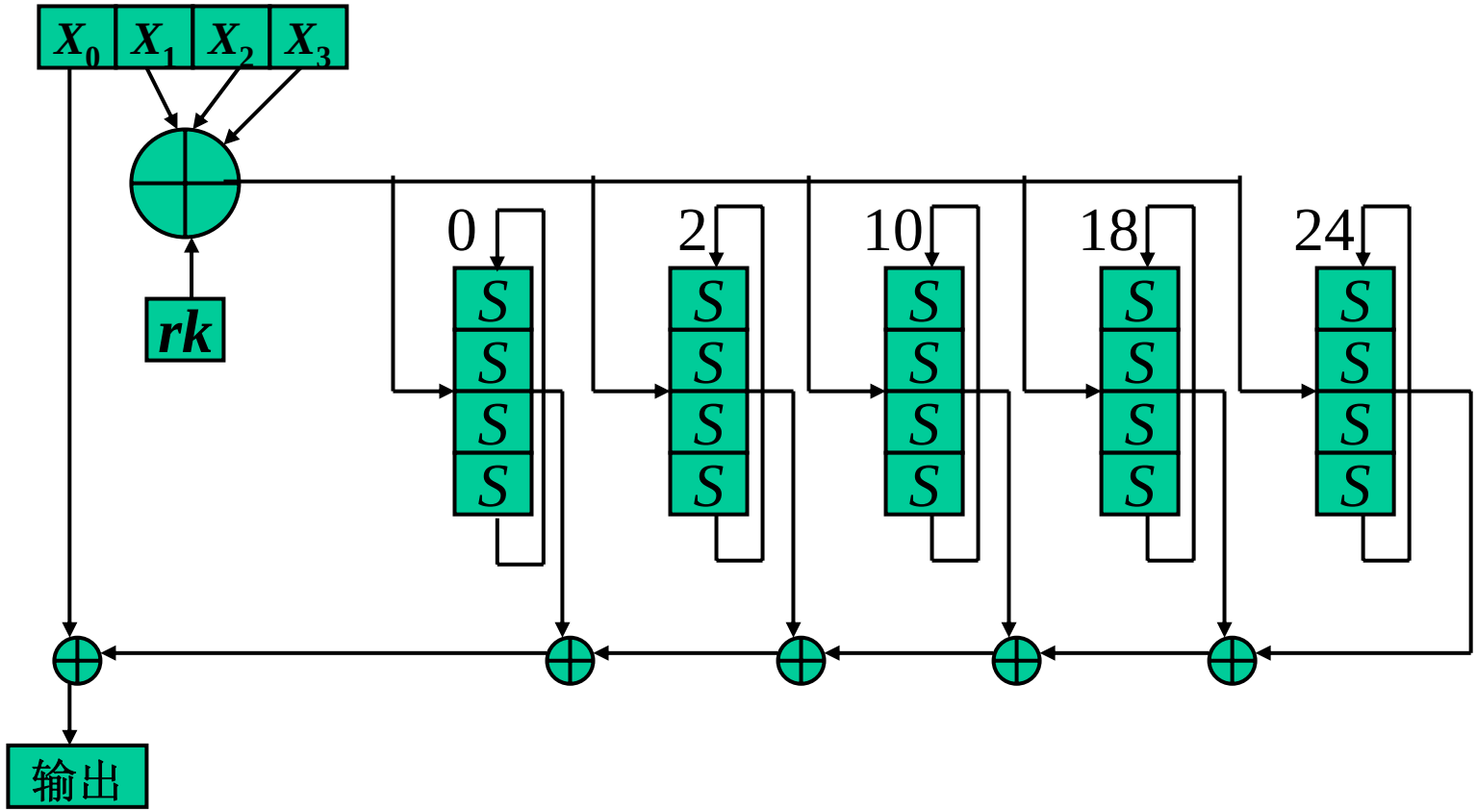
(3)轮函数F：

- 输入数据：（X_0，X_1，X_2，X_3），128位，四个32位字。
- 输入轮密钥：rk，32位字。
- 输出数据：32位字。
- 轮函数F：

$$F（X_0，X_1，X_2，X_3，rk）$$
$$= X_0 \oplus T（X_1 \oplus X_2 \oplus X_3 \oplus rk)$$

(三) SM4密码算法

(3)轮函数F：



(三) SM4密码算法

(4)加密算法：

- 输入明文:$(M_0, M_1, M_2, M_3)=(X_0, X_1, X_2, X_3)$, 128位，四个字。
- 输入轮密钥：rk_i，i=0,1,…,31，共32个轮密钥。
- 输出密文：（Y_0，Y_1，Y_2，Y_3），128位，四个字。
- 算法结构：轮函数32轮迭代，每轮使用一个轮密钥。
- 加密算法：

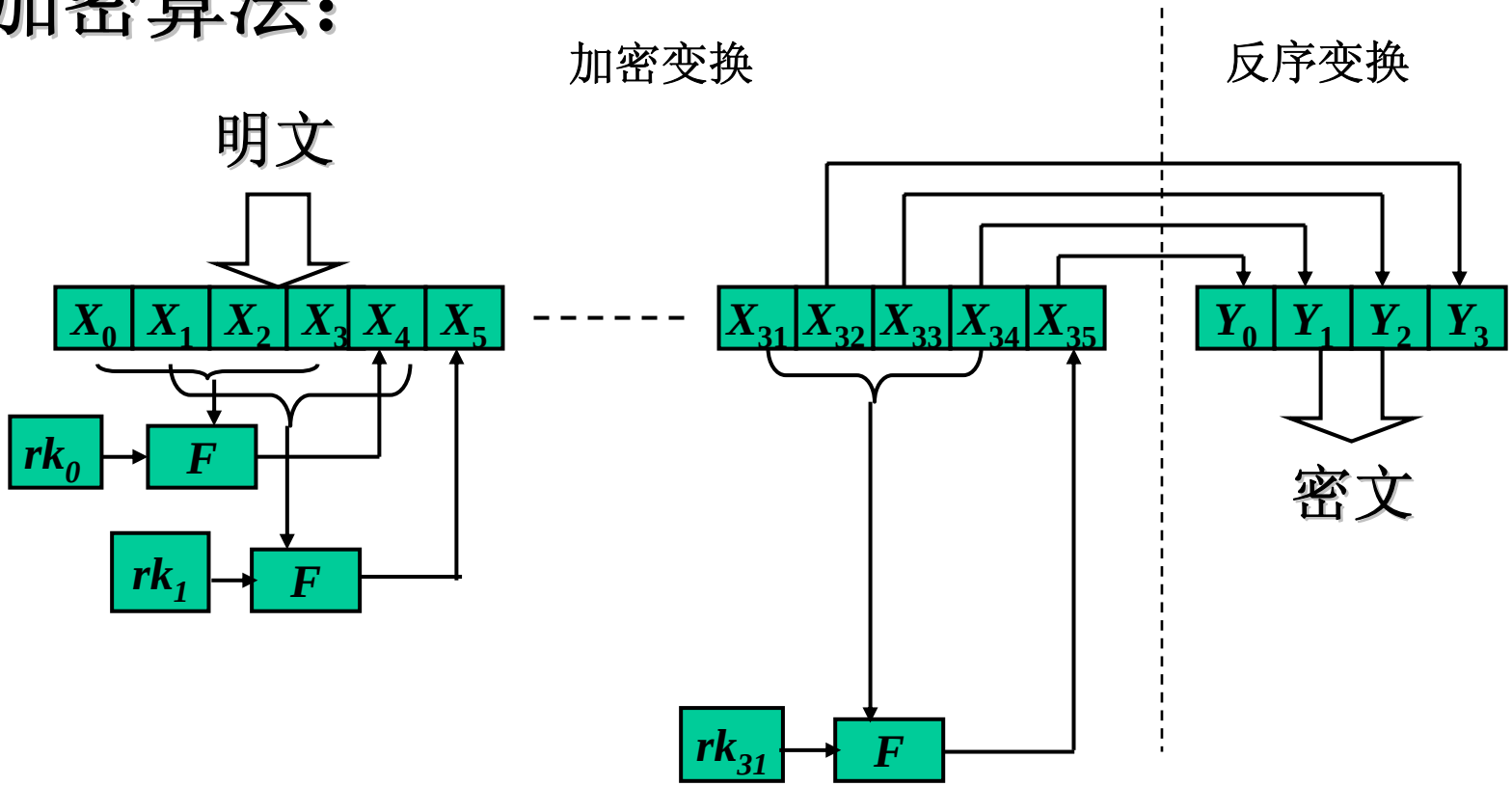
① 加密变换：$X_{i+4}=F(X_i, X_{i+1}, X_{i+2}, X_{i+3}, rk_i)$

$= X_i \oplus T(X_{i+1} \oplus X_{i+2} \oplus X_{i+3} \oplus rk_i)$，$i$ = 0,1…31

② 反序变换：$(Y_0, Y_1, Y_2, Y_3)=(X_{35}, X_{34}, X_{33}, X_{32})$

(三) SM4密码算法

(4)加密算法：



武漢大學

(三) SM4密码算法

(5)解密算法：

- SM4密码算法是对合的，因此解密与加密算法相同，只是轮密钥的使用顺序相反。
- 输入密文：$(Y_0, Y_1, Y_2, Y_3) = (X_{35}, X_{34}, X_{33}, X_{32})$
- 输入轮密钥：rk_i，i =31,30，…,1，0
- 输出明文：(X_0, X_1, X_2, X_3)
- 算法：轮函数的32轮迭代，每轮使用一个轮密钥。
- 解密算法：用符号X描述
  - 解密变换：$X_i = F(X_{i+4}, X_{i+3}, X_{i+2}, X_{i+1}, rk_i)$
    $= X_{i+4} \oplus T(X_{i+3} \oplus X_{i+2} \oplus X_{i+1} \oplus rk_i)$, i=31,…1,0
  - 反序变换：$(X_3, X_2, X_1, X_0) = (M_0, M_1, M_2, M_3)$

(三) SM4密码算法

(6)密钥扩展算法：

①常数FK

- 在密钥扩展中使用一些常数（32位字）：

FK_0=(A3B1BAC6)

FK_1=(56AA3350)

FK_2=(677D9197)

FK_3=(B27022DC)

(三) SM4密码算法

(6)密钥扩展算法：

②固定参数CK，32位字：

- 32 个固定参数Ck_i，i=0,1,2…31

  00070e15, 1c232a31, 383f464d, 545b6269,
  70777e85, 8c939aa1, a8afb6bd, c4cbd2d9,
  e0e7eef5, fc030a11, 181f262d, 343b4249,
  50575e65, 6c737a81, 888f969d, a4abb2b9,
  c0c7ced5, dce3eaf1, f8ff060d, 141b2229,
  30373e45, 4c535a61, 686f767d, 848b9299,
  a0a7aeb5, bcc3cad1, d8dfe6ed, f4fb0209,
  10171e25, 2c333a41, 484f565d, 646b7279

  产生规则：$Ck_{ij}= (4i+j)\times 7$（mod 256），i=0,1,2…31, j=0,1,…3 。

(三) SM4密码算法

(6)密钥扩展算法：

- 输入加密密钥：$MK=(MK_0, MK_1, MK_2, MK_3)$
- 输出轮密钥：rk_i，$i = 0, 1\ldots,30, 31$
- 中间数据：K_i，$i = 0, 1, \ldots, 34, 35$
- 密钥扩展算法：

① $(K_0,K_1,K_2,K_3) = (MK_0 \oplus FK_0, MK_1 \oplus FK_1, MK_2 \oplus FK_2, MK_3 \oplus FK_3)$

② For $i=0, 1\ldots,30, 31$ Do

$$ik_i = K_{i+4} = K_i \oplus T'(K_{i+1} \oplus K_{i+2} \oplus K_{i+3} \oplus CK_i)$$

说明：T'变换与加密算法的T基本相同，只将其中的线性变换L改为L'：$L'(B)=B \oplus (B <<< 13) \oplus (B <<< 23)$

(三) SM4密码算法

3、实例：

- 明文: 01 23 45 67 89 ab cd ef fe dc ba 98 76 54 32 10
- 密钥: 01 23 45 67 89 ab cd ef fe dc ba 98 76 54 32 10
- 密文: 68 1e df 34 d2 06 96 5e 86 b3 e9 4f 53 6e 42 46

4、安全性

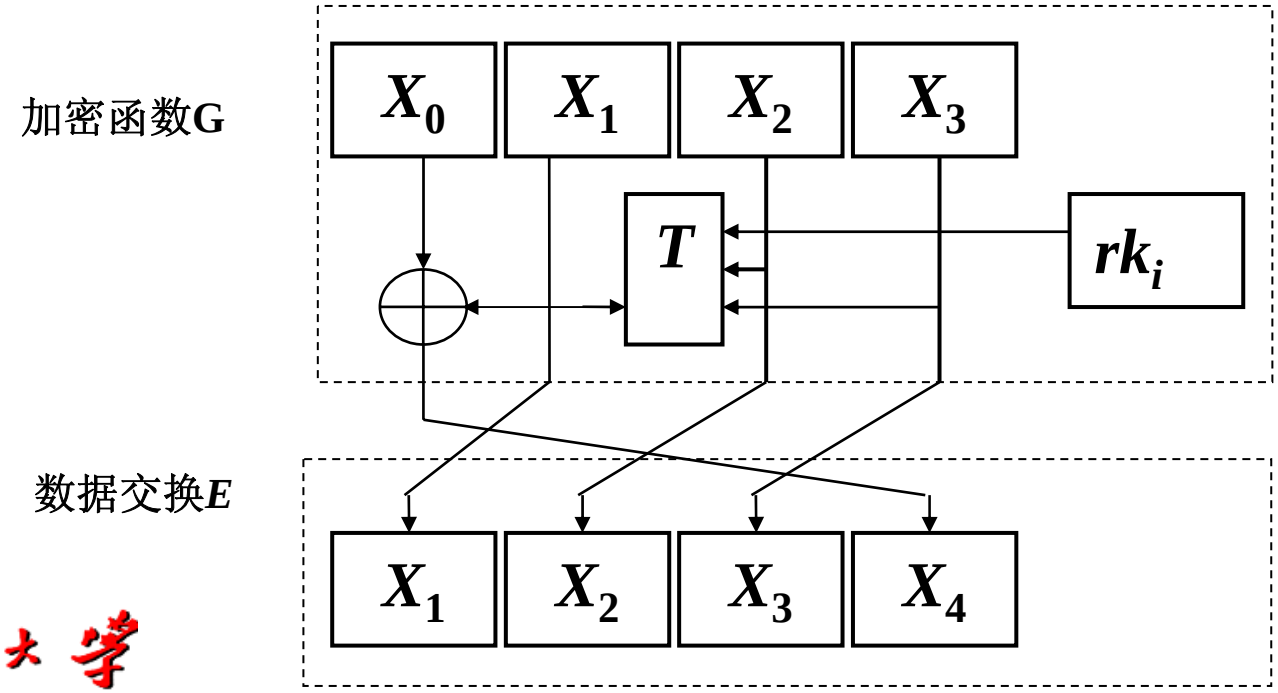
- 国家专业机构设计：算法简洁，以字和字节为处理单位，对合运算，符合当今分组密码主流。进行了充分的密码分析，因此是安全的。有弱点也是正常的。
- 有文献对23轮SM4进行了差分密码分析。
- 有文献报导，SM4对抗差分故障攻击能力较弱。

(四) SM4的可逆性与对合性

- 对合性
  - SM4的加密轮函数

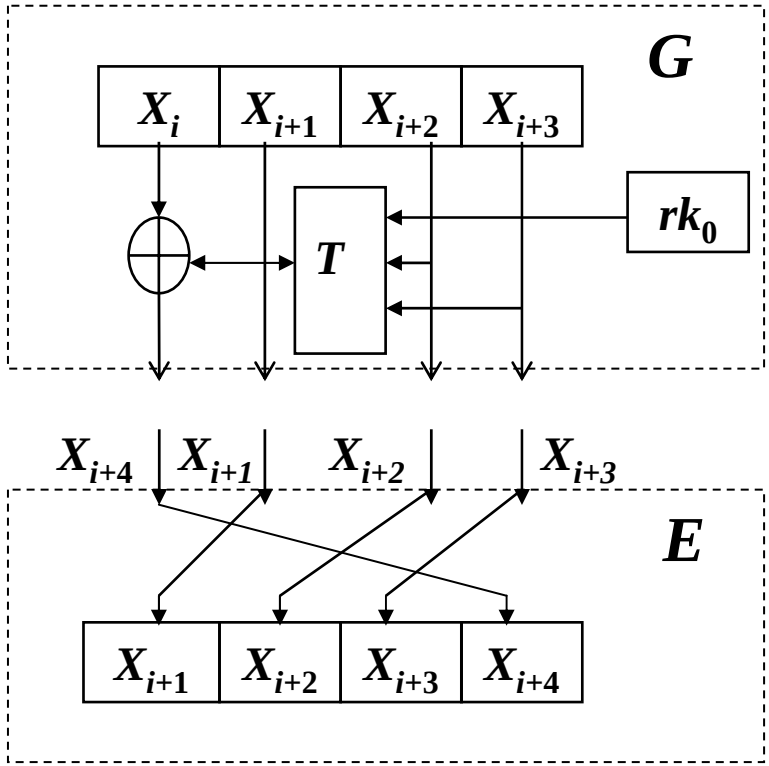
加密变换：$X_{i+4}=F(X_i, X_{i+1}, X_{i+2}, X_{i+3}, rk_i)$

$= X_i \oplus T(X_{i+1} \oplus X_{i+2} \oplus X_{i+3} \oplus rk_i)$，　$i = 0,1\ldots31$



(四) SM4的可逆性与对合性

- 对合性
  - SM4的加密轮函数
    - 分成加密函数*G*和数据交换*E*
    - 加密函数*G*进行加密处理
    - 数据交换*E*进行数据顺序交换
    - 轮函数$F_i=G_iE$



- $G_i=G_i(X_i, X_{i+1}, X_{i+2}, X_{i+3}, rki)$

  $=(X_i \oplus T(X_{i+1}, X_{i+2}, X_{i+3}, rki), X_{i+1}, X_{i+2}, X_{i+3})$

- $E(X_{i+4}, (X_{i+1}, X_{i+2}, X_{i+3}))=((X_{i+1}, X_{i+2}, X_{i+3}), X_{i+4})$

(四) SM4的可逆性与对合性

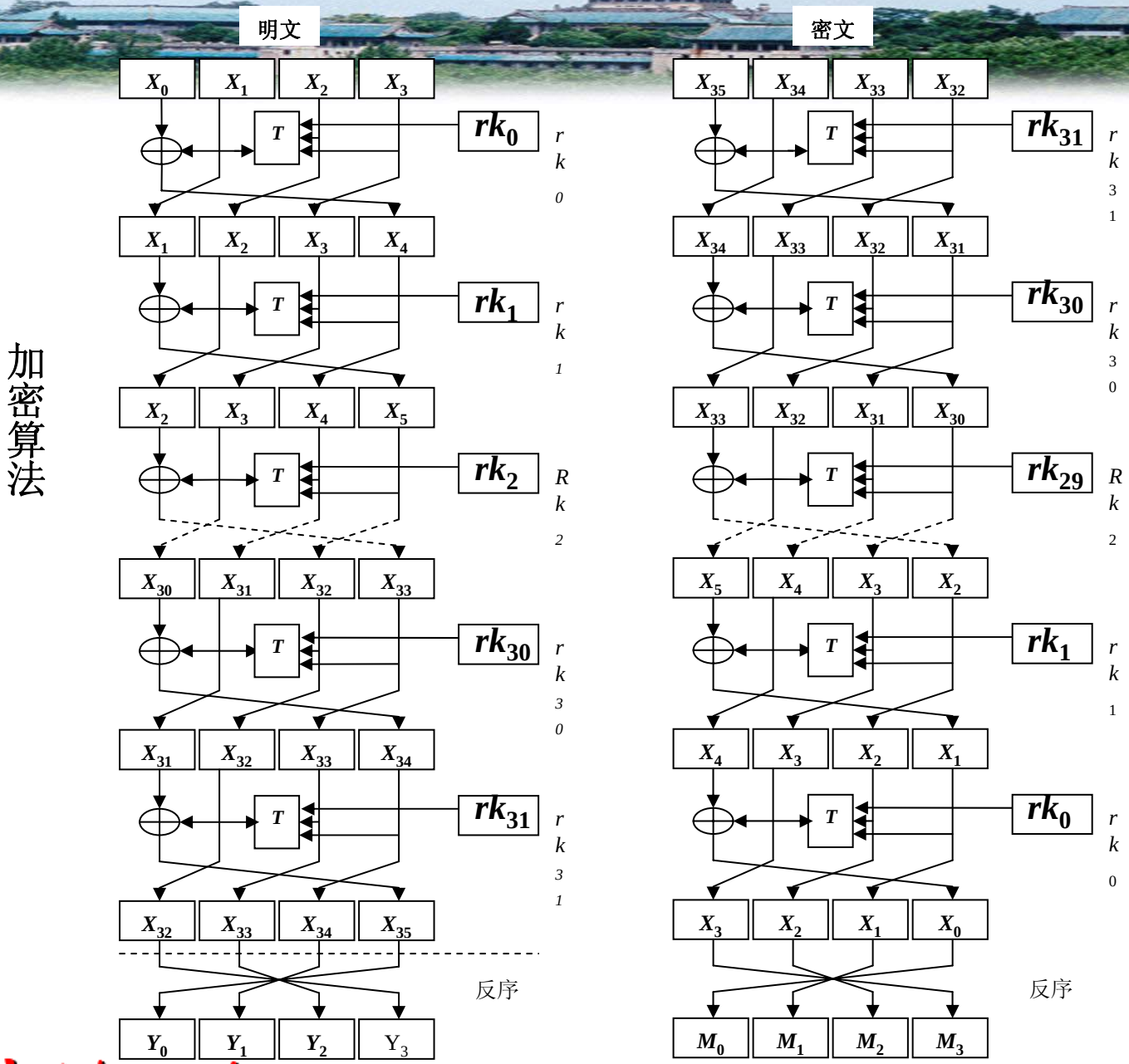
●对合性

$$(G_i)^2 = G_i(X_i \oplus T(X_{i+1}, X_{i+2}, X_{i+3}, rki), X_{i+1}, X_{i+2}, X_{i+3}, rki)$$
$$= (X_i \oplus T(X_{i+1}, X_{i+2}, X_{i+3}, rki) \oplus T(X_{i+1}, X_{i+2}, X_{i+3}, rki), X_{i+1}, X_{i+2}, X_{i+3}, rki)$$
$$= (X_i, X_{i+1}, X_{i+2}, X_{i+3}, rki)$$
$$= I$$

- 这说明加密函数G是对合的。
- 因为，E变换为：

$$E(X_{i+4}, (X_{i+1}, X_{i+2}, X_{i+3})) = ((X_{i+1}, X_{i+2}, X_{i+3}), X_{i+4})$$
$$E^2(X_{i+4}, (X_{i+1}, X_{i+2}, X_{i+3})) = I$$

- 显然，E是对合运算。
- 综上，轮函数是对合的。


明文
密文
加密算法
解密算法
X_0 X_1 X_2 X_3
X_{35} X_{34} X_{33} X_{32}
T
rk_0
rk_{31}
rk_1
rk_{30}
rk_2
rk_{29}
rk_{30}
rk_1
rk_{31}
rk_0
rk0
rk31
rk1
rk30
Rk2
Rk2
rk30
rk1
rk31
rk0
X_1 X_2 X_3 X_4
X_{34} X_{33} X_{32} X_{31}
X_2 X_3 X_4 X_5
X_{33} X_{32} X_{31} X_{30}
X_{30} X_{31} X_{32} X_{33}
X_5 X_4 X_3 X_2
X_{31} X_{32} X_{33} X_{34}
X_4 X_3 X_2 X_1
X_{32} X_{33} X_{34} X_{35}
X_3 X_2 X_1 X_0
反序
反序
Y_0 Y_1 Y_2 Y_3
M_0 M_1 M_2 M_3
密文
明文

(四) SM4的可逆行与对合性

- 对合性
  - 根据加密框图，可把SM4的加密过程写成：

$$SM4 = G_0\ E\ G_1\ E\ \ldots\ G_{30}\ E\ G_{31}\ E\ R$$

  - 根据解密框图，把SM4的解密过程写成：

$$SM4^{-1} = G_{31}\ E\ G_{30}\ E\ \ldots\ G_1\ E\ G_0\ E\ R$$

  - 比较SM4与$SM4^{-1}$ 可知，运算相同，只有密钥的使用顺序不同。
- 所以SM4是对合的。

(四) SM4的可逆行与对合性

- 可逆性
  - 根据加密框图，SM4的加密过程的数据变化：

    $(X_0, X_1, X_2, X_3)\rightarrow(X_1, X_2, X_3, X_4)\rightarrow(X_2, X_3, X_4, X_5)\rightarrow\ldots\rightarrow(X_{32}, X_{33}, X_{34}, X_{35})\rightarrow(X_{35}, X_{34}, X_{33}, X_{32})=(Y_0, Y_1, Y_2, Y_3)$。
    - 其中最后一步变换为反序。
  - 根据解密框图，密文(Y_0, Y_1, Y_2, Y_3)解密过程数据的变化：

    $(X_{35}, X_{34}, X_{33}, X_{32})\rightarrow(X_{34}, X_{33}, X_{32}, X_{31})\rightarrow(X_{33}, X_{32}, X_{31}, X_{30})\rightarrow\ldots\rightarrow(X_3, X_2, X_1, X_0)\rightarrow(X_0, X_1, X_2, X_3)$。
    - 其中最后一步变换为反序。
  - $SM4^{-1}(SM4(X_0, X_1, X_2, X_3))=(X_0, X_1, X_2, X_3)$
- 所以SM4是可逆的。

(四) SM4的可逆性与对合性

- SM4的密码结构
  - SM4密码与DES密码有相似性
  - DES密码采用的是Feistel结构，因此SM4也采用了Feistel结构。
  - 但是，DES中参与交换的两个数据块的长度是相等的，而在SM4中参与交换的两个数据块的长度是不相等的。
  - 因此，密码界称DES密码采用的是对称Feistel结构，**SM4 密码采用的是非对称Feistel结构。**

二、分组密码应用技术

(一) 计算机数据的特殊性

1、存在明显的数据模式

- 许多数据都具有某种固有的模式。这主要是由数据冗余和数据结构引起的。
- 各种计算机语言的语句和指令都十分有限，因而在程序中便表现为少量的语句和指令的大量重复。
- 各种语言程序往往具有某种固定格式。
- 数据库的记录也往往具有某种固定结构。
- 操作系统和网络协议也有同样的问题。

(一) 计算机数据的特殊性

1、存在明显的数据模式：

- 根据明文相同、密钥相同，则密文相同的道理，这些固有的数据模式将在密文中表现出来。
- 掩盖明文数据模式的方法：
  - ■随机掩盖技术：
    - ◆使用一个随机序列掩盖明文数据，从而消除明文中的数据模式。
    - ◆缺点：通信双方必须共享该随机序列，带来许多麻烦。
  - ■链接技术
    - ◆使前后明文块及密文块彼此关联起来，从而消除明文中的数据模式
- 如果不能掩盖数据模式，既使采用安全的密码算法也是徒劳的。

(一) 计算机数据的特殊性

2、分组密码用于数据加密存在短快问题：

- 设明文M长度为n_1，分组密码的明文分组长度为n_2，如果n_1不是n_2的整数倍，则最后一块要加密的数据块的长度必然小于明文分组长度n_2，称此数据块为短块。
- 分组密码不能直接加密短块数据，必须采取特殊的方法处理短块。

(二) 分组密码的工作模式

1977年DES颁布，1981年美国政府针对DES的应用，制定了DES的四种基本工作模式：

- 电码本模式（ECB）
- 密文反馈链接模式（CBC）
- 密码反馈模式（CFB）
- 输出反馈模式（OFB）

(二) 分组密码的工作模式

2000年美国在征集AES的同时又公开征集AES的工作模式，共征集到 15个候选工作模式。

- 经过评审选定了几个新的工作模式。
- 这些新的工作模式将为AES的应用作出贡献。

(二) 分组密码的工作模式

1、电码本模式（ECB）

- 直接利用分组密码对明文的各分组进行加密。
- 设　明文　$M=(M_1, M_2, \ldots, M_n)$，
  密钥为K，
  密文　$C=(C_1, C_2, \ldots, C_n)$，
  其中　$C_i=E(M_i, K),\ i=1,2,\ldots,n$
- 电码本方式是分组密码的基本工作模式。
- 缺点：
  - 可能出现短块，这时需要特殊处理。
  - 密钥K固定，如果$M_i=M_j$，则$C_i=C_j$，从而暴露明文的数据模式。
- 应用：适合加密密钥等短数据

(二) 分组密码的工作模式

2、密文反馈链接模式（CBC）

①明密文链接方式（Plaintext and Ciphertext Block Chaining）

● 设　明文 $M=(M_1, M_2, \ldots, M_n)$，

密钥为K,

密文 $C=(C_1, C_2, \ldots, C_n)$,

其中，

$$C_i=\begin{cases} E(M_i \oplus Z, K), & i=1 \\ E(M_i \oplus M_{i-1} \oplus C_{i-1}, K), & i=2,\ldots,n \end{cases}$$

Z为初始化向量。

(二) 分组密码的工作模式

2、密文反馈链接模式（CBC）

①明密文链接方式

- 错误传播
  - ◆加密时，明文或密文发生错误引起对应密文及其后续密文发生错误
  - ◆解密时，密文或明文发生错误引起对应明文及其后续明文发生错误
  - ◆如果密码算法的输入数据错误只引起对应的几个输出数据错误，则成称为错误传播有界
  - ◆如果密码算法的输入数据错误引起对应的输出数据及其后续的输出数据全部错误，则称为错误传播无界

(二) 分组密码的工作模式

2、密文反馈链接模式（CBC）

①明密文链接方式

- 即使$M_i = M_j$，但因一般都有$M_{i-1} \oplus C_{i-1} \neq M_{j-1} \oplus C_{j-1}$，从而使$C_i \neq C_j$，从而掩盖了明文中的数据模式。
- 加密时错误传播无界，当M_i或C_i中发生一位错误时，自此以后的密文全都发生错误。
- 解密时也是错误传播无界。

$$\begin{cases} M_i = E(C_i, K) \oplus Z, & i = 1 \\ M_i = E(C_i, K) \oplus M_{i-1} \oplus C_{i-1}, & i = 2,\ldots,n \end{cases}$$

Z为初始化向量。

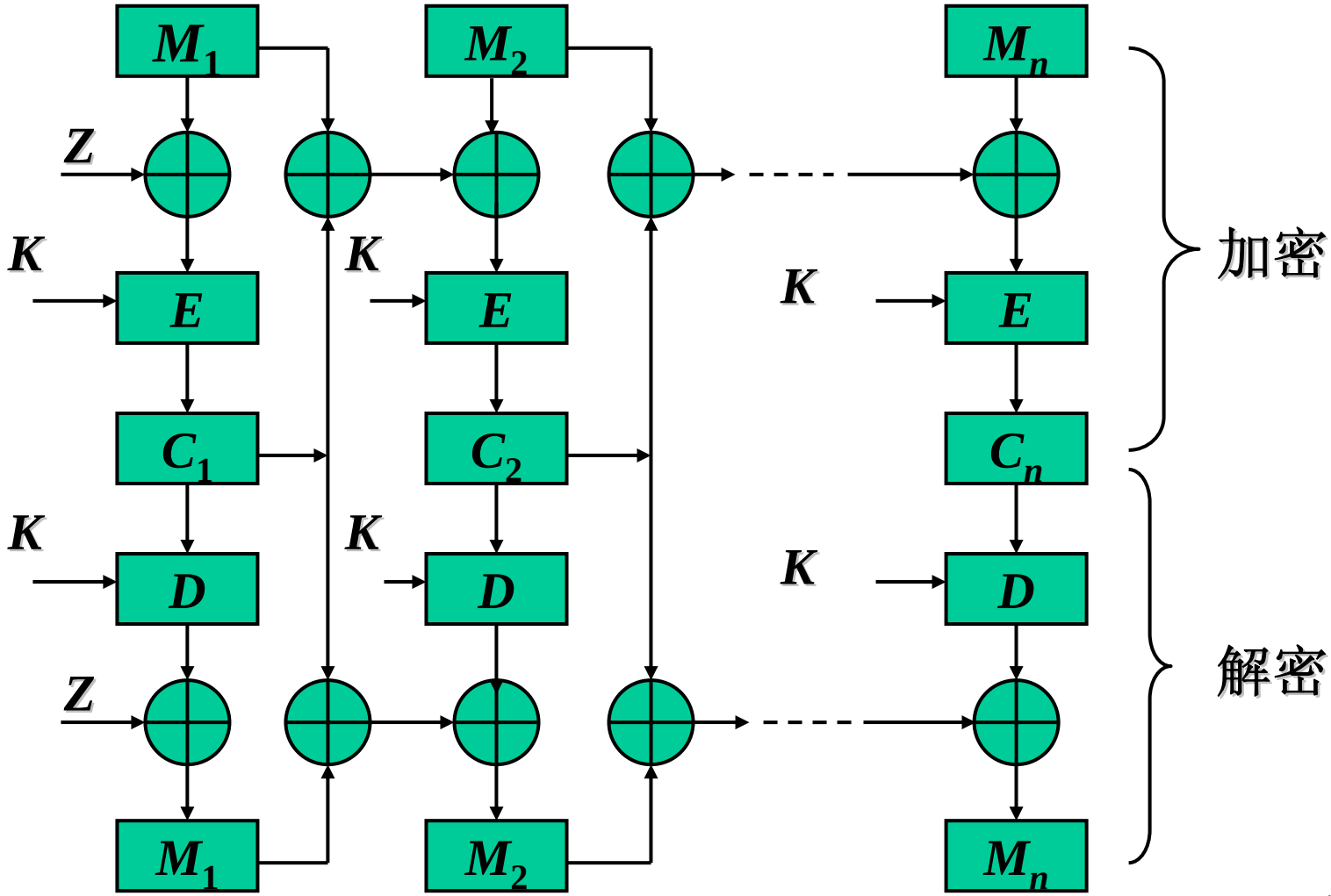
(二) 分组密码的工作模式

2、密文反馈链接模式（CBC）

①明密文链接方式

- 加密是在发送端进行的，加密时明文和密文发生错误的概率都较小。
- 解密是在接收端进行的，解密时明文发生错误的概率较小。但是，密文是经过信道传输的，因此密文在信道中受干扰而发生错误的概率较大。
- 错误传播无界的缺点：当磁盘发生一点损坏时将导致整个密文文件无法解密。
- 错误传播无界的优点：可用于数据完整性、真实性认证。

(二) 分组密码的工作模式



(二) 分组密码的工作模式

2、密文反馈链接模式（CBC）

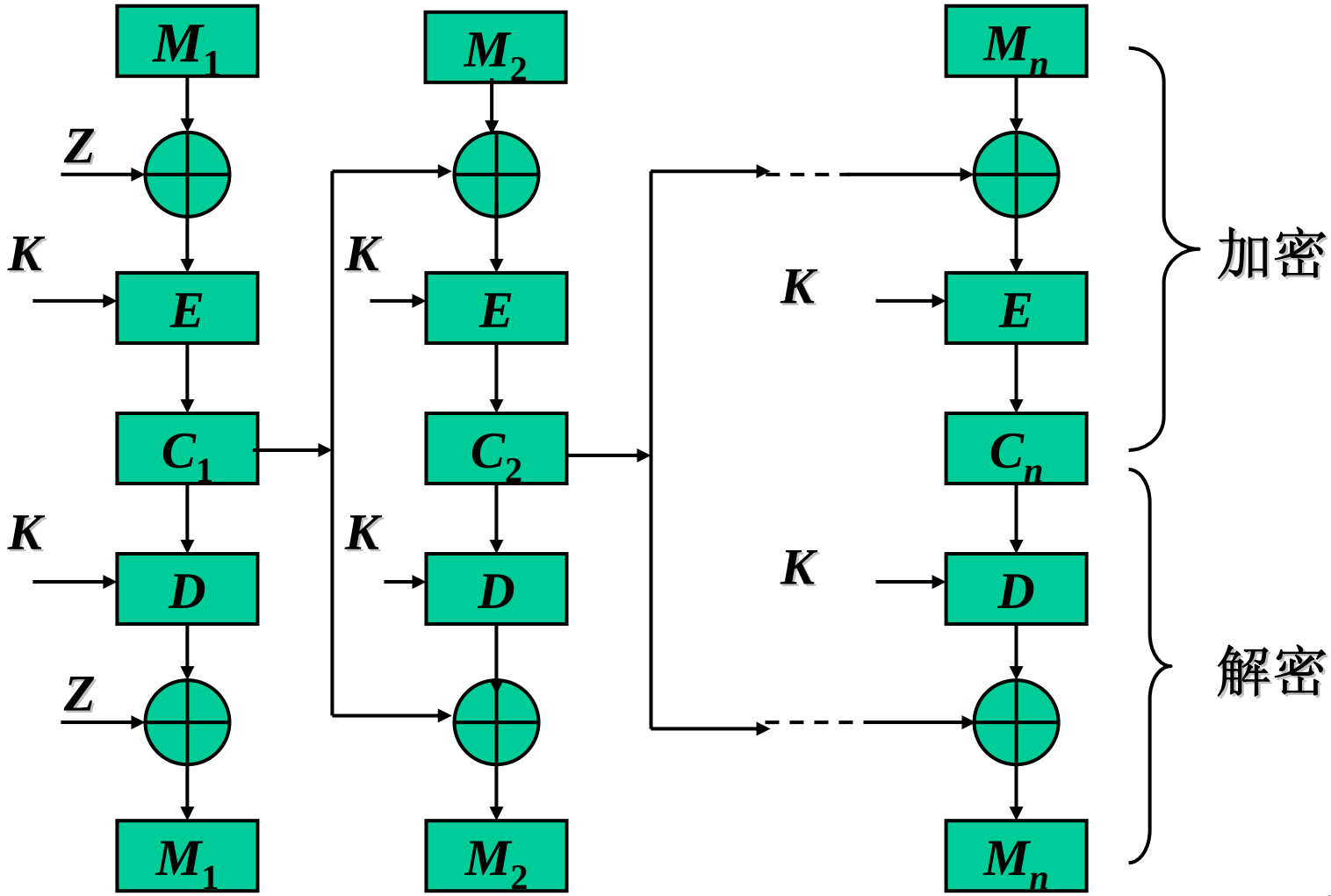
- 明密文链接方式具有加解密错误传播无界的特性，而磁盘文件加密和通信加密通常希望解密错误传播有界，这时可采用密文链接方式。

②密文链接方式（ Ciphertext Block Chaining）

- 设　明文 $M=(M_1, M_2, \ldots, M_n)$，
  密钥为K,
  密文 $C=(C_1, C_2, \ldots, C_n)$,

$$C_i=\begin{cases} E(M_i \oplus Z, K), & i=1，Z\text{为初始化向量。} \\ E(M_i \oplus C_{i-1}, K), & i=2,\ldots,n \end{cases}$$

(二) 分组密码的工作模式



武漢大学

(二) 分组密码的工作模式

2、密文反馈链接模式（CBC）

②密文链接方式

- 加密：错误传播无界
  - M_i 或C_{i-1}错误，将影响C_i及其以后的密文全错。

- 解密时：错误传播有界

$$M_i=\begin{cases} D(C_i, K)\oplus Z, & i=1 \\ D(C_i, K)\oplus C_{i-1}, & i=2,\ldots,n \end{cases}$$

  - C_{i-1} 发生了错误，则只影响M_{i-1} 和M_i 发生错误，其余不错，因此错误传播有界。
  - 解密错误传播有界有利于提高密码处理的可用性。

(二) 分组密码的工作模式

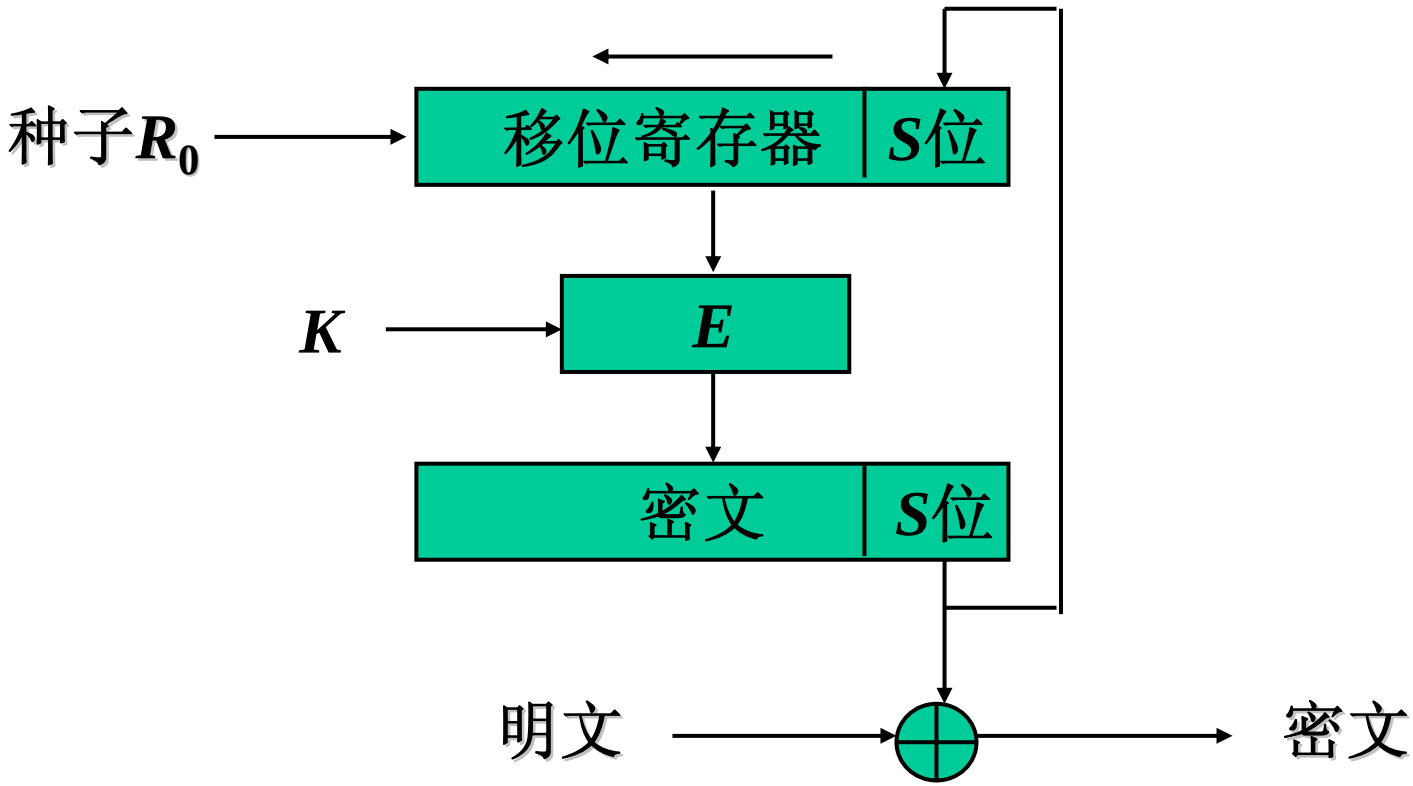
3、输出反馈模式 (OFB)

- 将一个分组密码转换为一个密钥序列产生器。从而可以实现用分组密码按流密码的方式进行加解密。

  - 采用一个移位寄存器，R_0是初始内容，称为种子。
  - E 是DES、AES、SM4 等强密码，加密移位寄存器的内容，输出密文最右边的s（ $s \geq 1$）位作密钥，与明文摸2加加密。
  - 移位寄存器左移s位，密文最右边的s位 反馈到移位寄存器的右s位。E再加密，再输出密钥。
  - 如此继续，把分组密码转变成了序列密码。

(二) 分组密码的工作模式

3、输出反馈模式 (OFB)



(二) 分组密码的工作模式

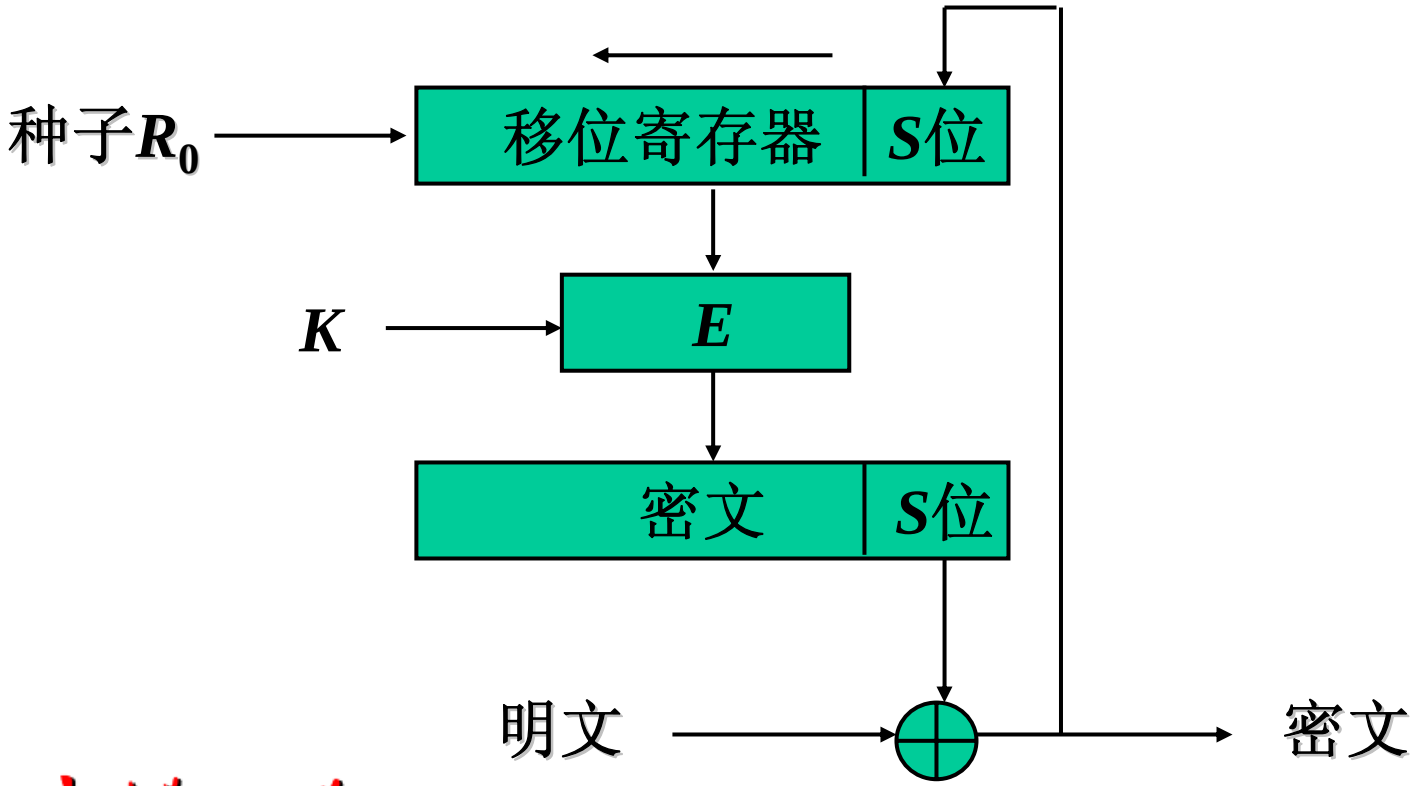
3、输出反馈模式 (OFB)

- 如果分组密码是安全的，则产生的密钥序列也是安全的。
- 加解密都没有错误传播。
- 适于加密冗余度较大的数据，如语音和图象数据。
- 为了提高速度可输出最右边的 8位。
- 因无错误传播，所以对密文的篡改难以检测。

(二) 分组密码的工作模式

4、密码反馈模式 CFB（Cipher Feedback）

- CFB模式也是用分组密码产生密钥序列。



武漢大学

(二) 分组密码的工作模式

4、密码反馈模式 (CFB)

- 与OFB的不同是，把密文反馈到移位寄存器。
- 加密时若明文错了一位，则影响相应的密文错，这一错误反馈到移位寄存器后将影响到后续的密钥序列错，导致后续的密文都错。
- 解密时若密文错了一位，不仅影响相应的明文错，而且密文的这一错误反馈到移位寄存器后将影响到后续的密钥序列错，导致后续的明文都错。
- 因错误传播无界，可用于检查发现对明密文的篡改。

(二) 分组密码的工作模式

5、X CBC (Extended Cipher Block Chaining Encryption)模式

- 2000年美国学者John Black和Phllip Rogaway提出X CBC模式，作为CBC模式的扩展，推荐为AES的工作模式，被美国政府采纳作为标准。
- X CBC主要是解决了CBC要求明文数据的长度是密码分组长度的整数倍的限制，可以处理任意长的数据。如果用分组密码是安全的，则密钥序列就是安全的。

(二) 分组密码的工作模式

5、X CBC (Extended Cipher Block Chaining Encryption)模式

- 设明文$M=(M_1, M_2, \ldots, M_{n-1}, M_n)$，相应的密文$C=(C_1, C_2, \ldots, C_{n-1}, C_n)$，而$M_n$可能是短块。
- 使用3个密钥K_1，K_2，K_3进行加密。
- 使用填充函数$Pad(X)$对短块数据进行填充。

(二) 分组密码的工作模式

5、X CBC (Extended Cipher Block Chaining Encryption)模式

- 填充函数*Pad*（*X*）定义如下：

$$\text{Pad}(X)=\begin{cases} X, & \text{当}X\text{不是短块;} \\ X10\ldots0, & \text{当}X\text{是短块。} \end{cases}$$

- 经填充函数 *Pad*（*X*）填充后的数据块一定是标准块。

(二) 分组密码的工作模式

5、X CBC (Extended Cipher Block Chaining Encryption)模式

- 令$Z=0$，以Z作为初始化向量。加密过程如下：

$$C_i = \begin{cases} E(M_i \oplus Z, K_1), & i=1，Z为初始化向量。 \\ E(M_i \oplus C_{i-1}, K_1), & i=2,\ldots,n-1 \end{cases}$$

$$C_n = \begin{cases} E(M_n \oplus C_{n-1} \oplus K_2, K_1), & 当M_n不是短块； \\ E(\mathrm{PAD}(M_n) \oplus C_{n-1} \oplus K_3, K_1), & 当M_n是短块。 \end{cases}$$

(二) 分组密码的工作模式

5、X CBC (Extended Cipher Block Chaining Encryption)模式

- X CBC与CBC的区别：
  - CBC要求最后一个数据块是标准块，不是短块。
  - X CBC既允许最后一个数据块是标准块，也允许是短块。
  - 最后一个数据块的加密方法与 CBC不同。
  - 因为有填充，需要传输填充长度信息。

(二) 分组密码的工作模式

5、X CBC (Extended Cipher Block Chaining Encryption)模式

- X CBC模式的主要优点：
  - 可以处理任意长度的数据。
  - 适于计算产生检测数据完整性的消息认证码MAC。
- X CBC模式的主要缺点：
  - 有填充，不适合文件和数据库加密。
  - 使用3个密钥，需要传输填充长度，控制复杂。

(二) 分组密码的工作模式

6、CTR（Counter Mode Encryption）模式

- CTR模式是Diffie和Hellman于1979年提出的，在征集AES工作模式的活动中由California大学的Phillip Rogaway等人的推荐。
- 设T_1，T_2，…，T_{n-1}，T_n 是一给定的计数序列，M_1，M_2，…，M_{n-1}，M_n 是明文，其中M_1，M_2，…，M_{n-1}是标准块，M_n 的可能是标准块，也可能是短块。设M_n 的长度等于u，$u \leqslant$ 分组长度。

(二) 分组密码的工作模式

6、CTR（Counter Mode Encryption）模式

- CTR的工作模式的加密过程如下：

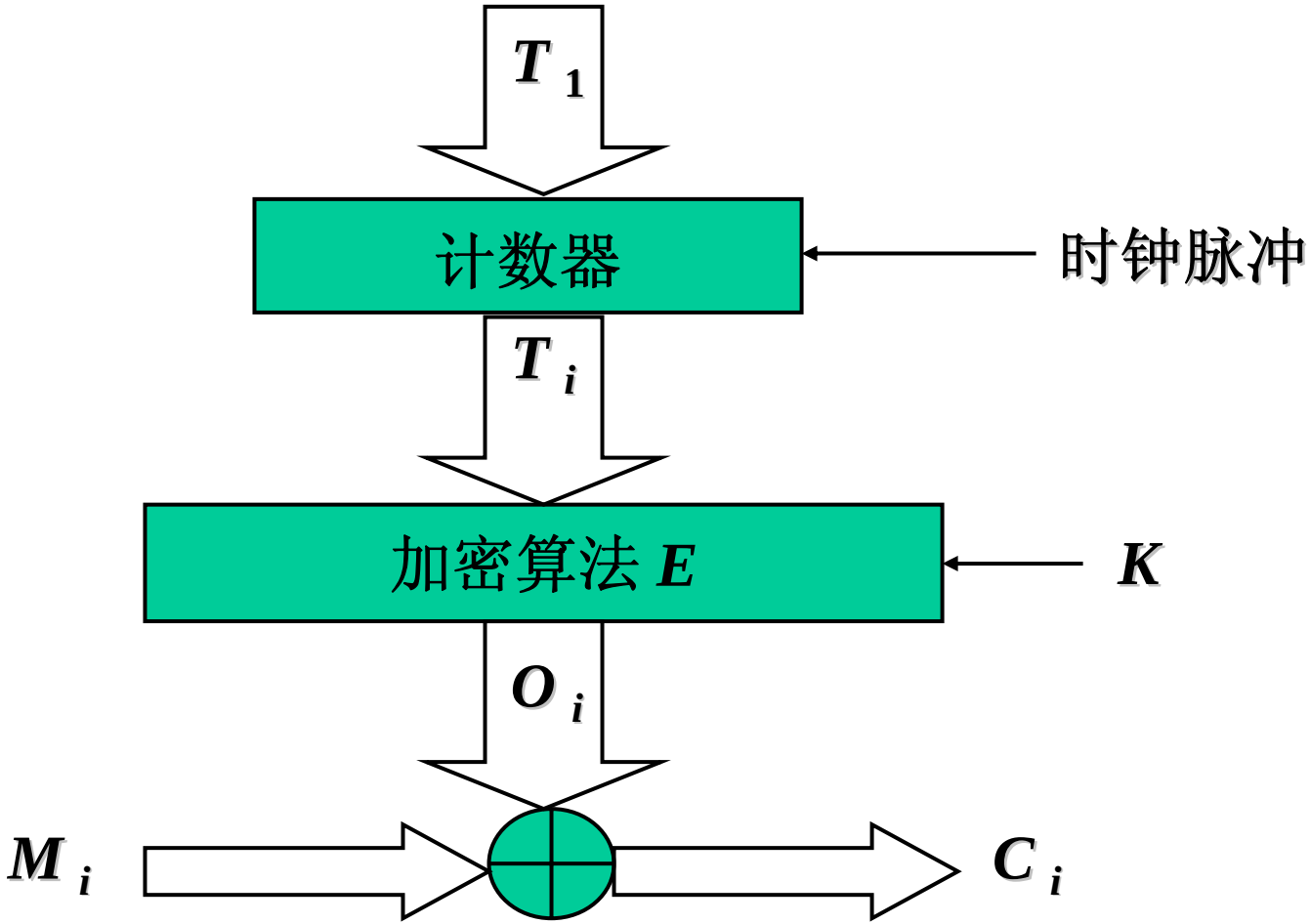
$$O_i = E(T_i, K),\ i=1,2,\ldots,n.$$

$$C_i = M_i \oplus O_i,\ i=1,2,\ldots,n-1.$$

$$C_n = M_n \oplus MSB_u(O_n).$$

其中$MSB_u(O_n)$表示O_n中的高u位。

(二) 分组密码的工作模式



(二) 分组密码的工作模式

6、CTR（Counter Mode Encryption）模式

- CTR的工作模式的解密过程如下：

$$O_i = E(T_i, K), \quad i=1,2,\ldots,n.$$

$$M_i = C_i \oplus O_i, \quad i=1,2,\ldots,n-1.$$

$$M_n = C_n \oplus MSB_u(O_n).$$

其中$MSB_u(O_n)$表示O_n中的高u位。

(二) 分组密码的工作模式

6、CTR（Counter Mode Encryption）模式

- CTR的工作模式的优点：
  - CTR模式的优点是安全、高效、可并行、适合任意长度的数据；
  - O_i的计算可预处理高速进行；
  - 由于采用了摸2加实现加密，是对合运算，解密运算与加密运算相同。
  - 适合随机存储数据的解密。
- CTR模式的缺点：
  - 没有错误传播，因此不易确保数据完整性。

(三) 短块加密

- 分组密码一次只能对一个固定长度的明文（密文）块进行加（解）密。
- 称长度小于分组长度的数据块为短块。
- 必须采用合适的技术解决短块加密问题。
- 短块处理技术：
  - 填充技术
  - 密文挪用技术
  - 序列加密

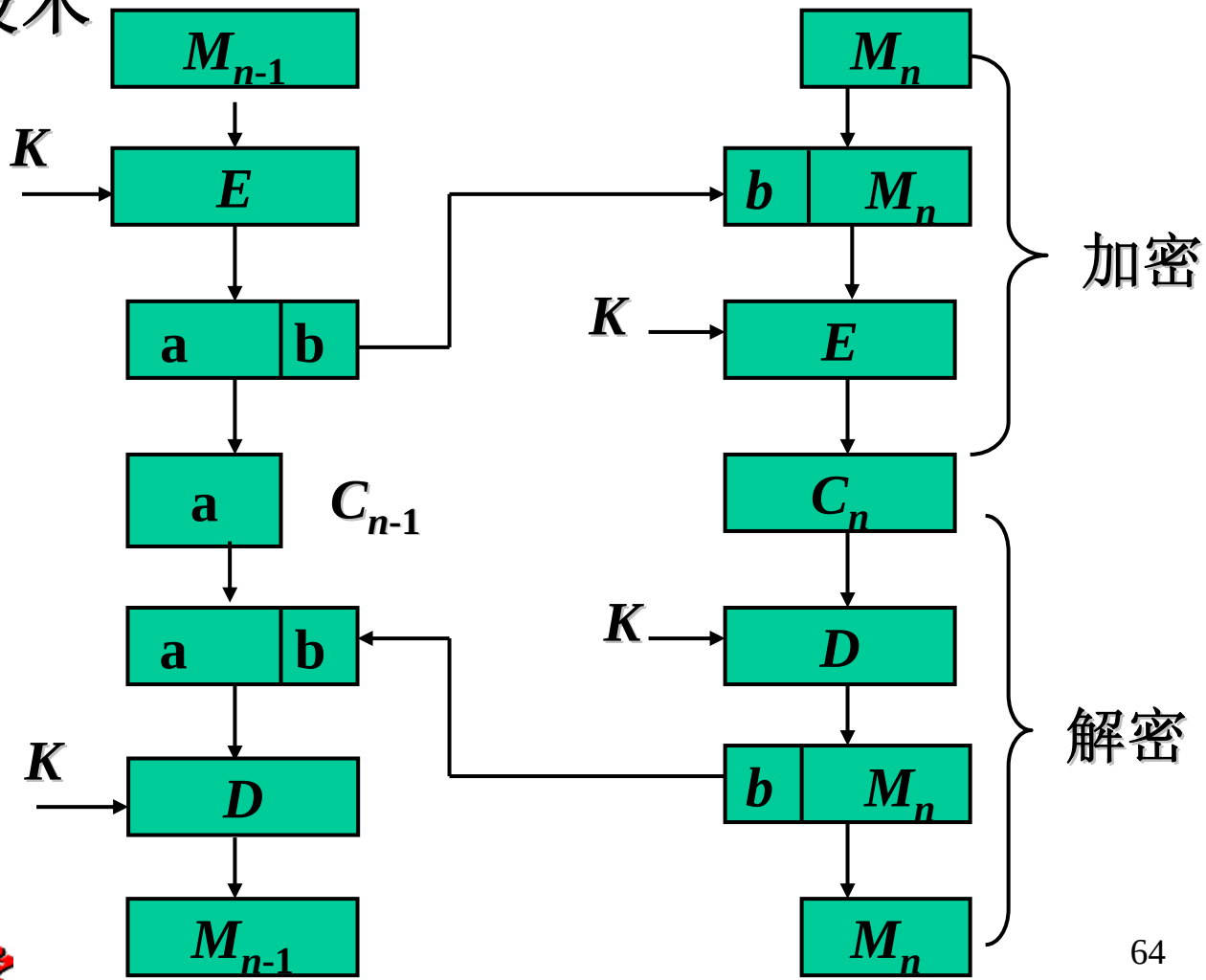
(三) 短块加密

1、填充技术

- 用无用的数据填充短块，使之成为标准块。
- 为了确保加密强度，填充数据应是随机的。
- 但是收信者如何知道哪些数字是填充的呢？这就需要增加指示信息，通常用最后8位作为填充指示符。
- 填充可能引起存储器溢出，因而可能不适于文件和数据库加密。

(三) 短块加密

2、密文挪用技术



(三) 短块加密

- 密文挪用法也需要指示挪用位数的指示符，否则收信者不知道挪用了多少位，从而不能正确解密。

- 密文挪用加密短块的优点是不引起数据扩展。

- 缺点是解密时要先解密C_n 、还原挪用后再解密C_{n-1}，从而使控制复杂。

(三) 短块加密

3、序列加密

- 对于最后一块短块数据，直接使用密钥K与短块数据模2相加。
- 序列加密方法的优点是简单。
- 但是如果短块太短，则加密强度不高。

谢　谢！