



## 《现代密码学》第四讲

# 分组密码 (二)





## 《现代密码学》第四讲

# 数据加密标准（DES）算法介绍





# 上讲内容回顾

● 分组密码定义

● 分组密码的发展历史

● 保密系统的安全性分析及分组密码的攻  
击





# 本节主要内容

- ❖ **DES 算法的整体结构**
- ❖ **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

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





# DES 算法的整体结构

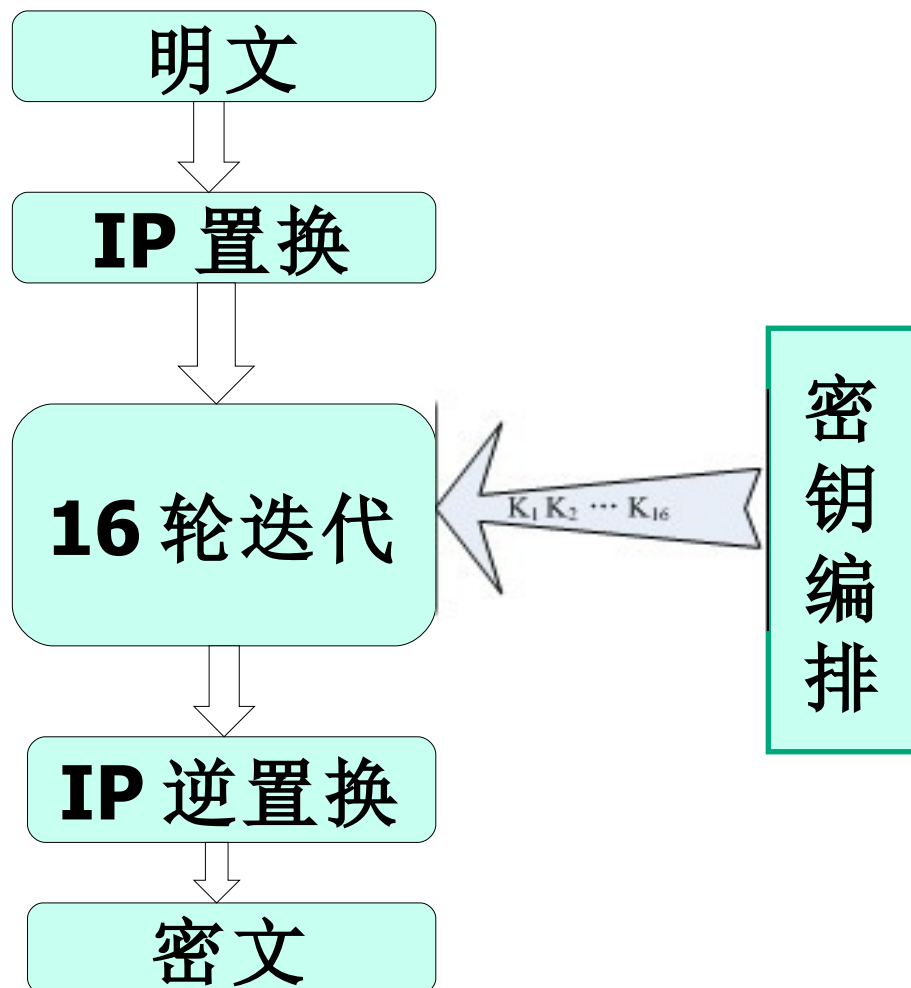
DES 是从 1975 年被美国联邦政府确定为非敏感信息的加密标准，它利用 56 比特长度的密钥  $K$  来加密长度为 64 比特的明文，得到 64 比特长的密文。

1997 年，由于计算机技术迅速发展，DES 的密钥长度已经太短，NIST 建议停止使用 DES 算法作为标准。目前，二重 DES 和三重 DES 仍然广泛使用。



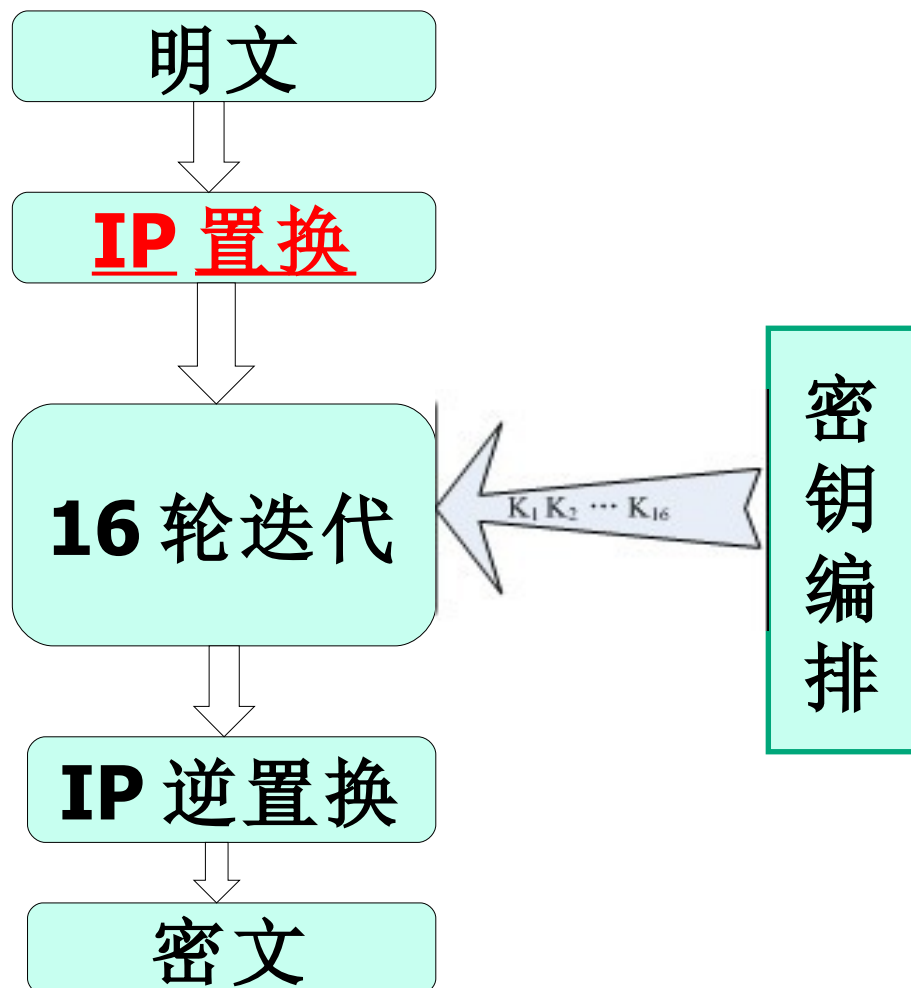


# DES 算法的整体结构





# DES 算法的整体结构





# DES 算法的整体结构

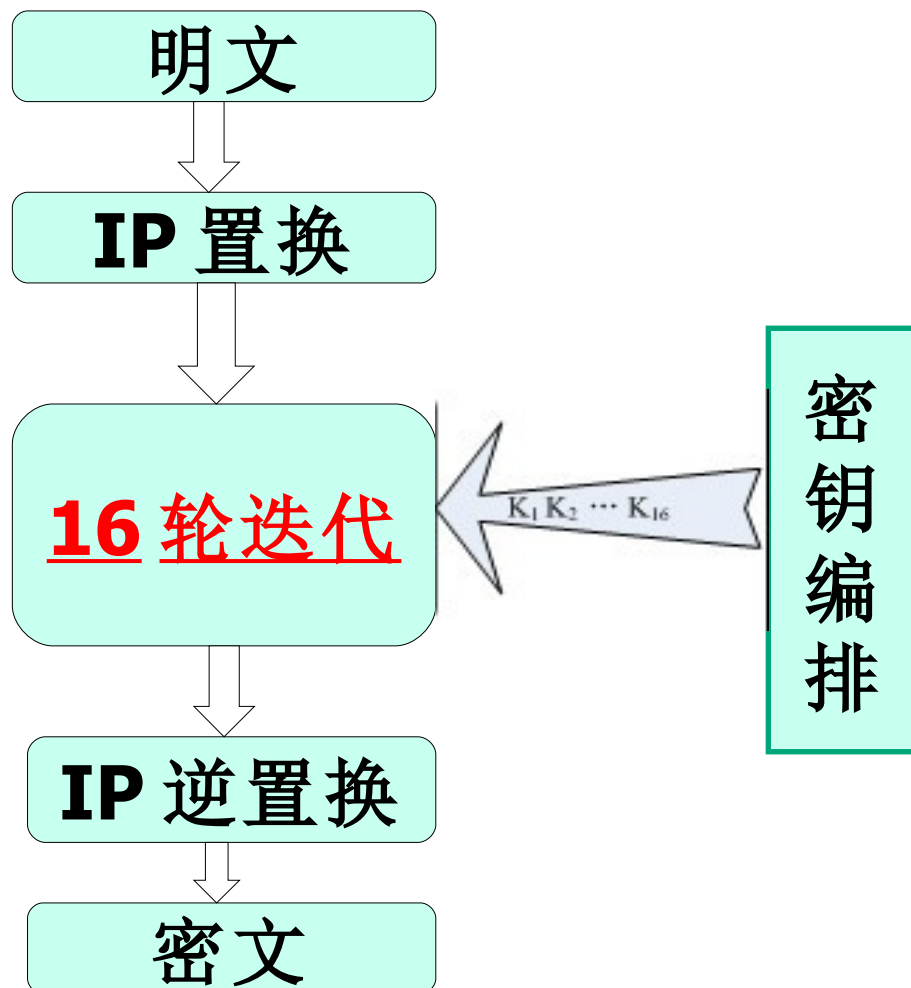
- 1. 给定明文，通过一个固定的初始置换 IP 来重排输入明文块 P 中的比特，得到比特串  $P_0 = IP(P) = L_0 R_0$ ，这里  $L_0$  和  $R_0$  分别是  $P_0$  的前 32 比特和后 32 比特

初始置换  
IP

IP							
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7



# DES 算法的整体结构





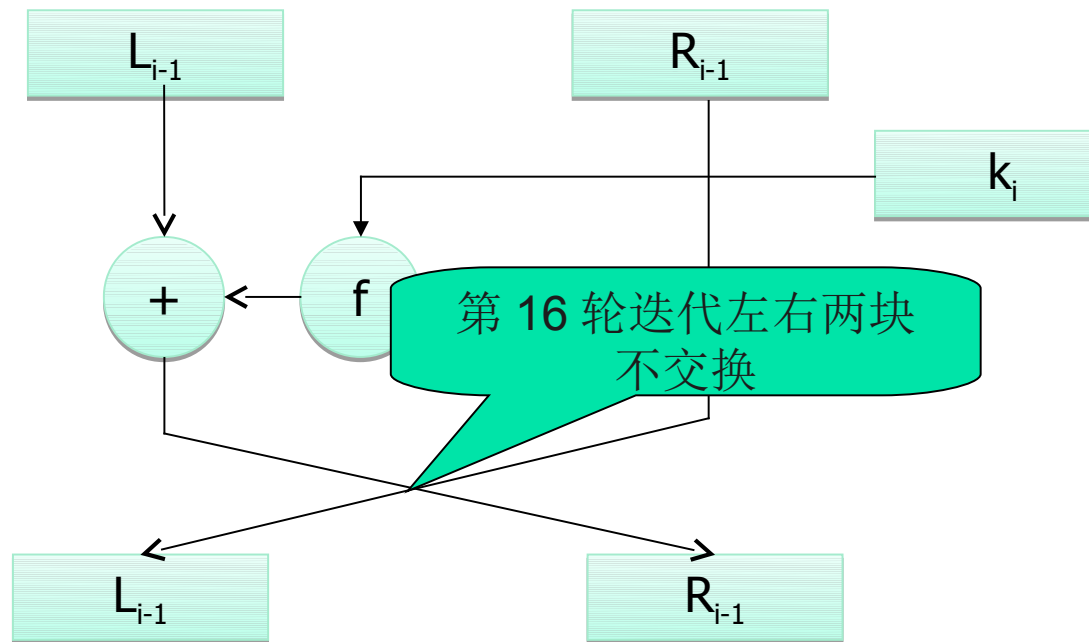
# DES 算法的整体结构

- Feistel 结构
- 2. 按下述规则进行 16 次迭代，即

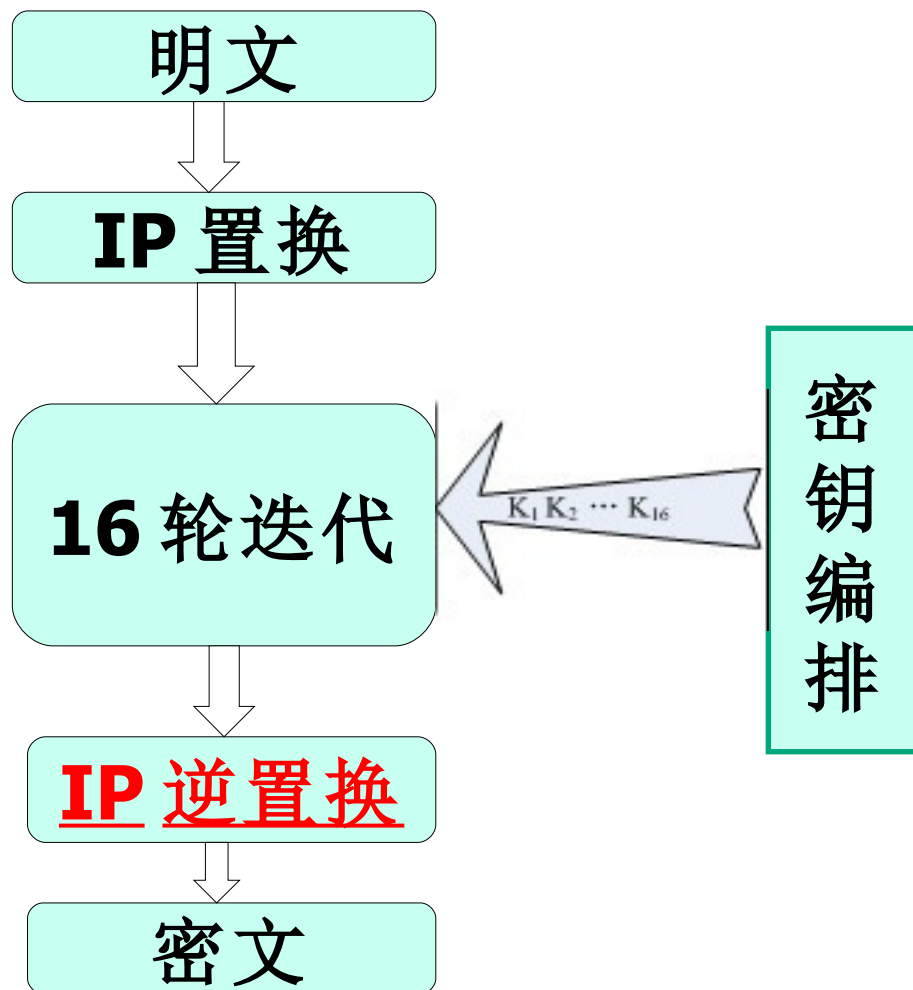
$$L_i = R_{i-1} \oplus R_i = L_i \oplus f(R_{i-1}, K_i)$$

这里 是对应比特的模 2 加， $f$  是一个函数（称为轮函数）；

16 个长度为 48 比特的子密钥  $K_i$  ( $1 \leq i \leq 16$ ) 是由密钥  $k$  经密钥编排函数计算出来的。



# DES 算法的整体结构——Feistel 结构



# DES 算法的整体结构——Feistel 结构


- 3. 对比特串  $R_{16}L_{16}$  使用逆置换  $IP^{-1}$  得到密文  $C$ ，即  $C=IP^{-1}(R_{16}L_{16})$ 。（注意  $L_{16}$  和  $R_{16}$  的相反顺序）

初始置换  
的逆置换  
 $IP$

IP <sup>-1</sup>							
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25



# 本节主要内容

- 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



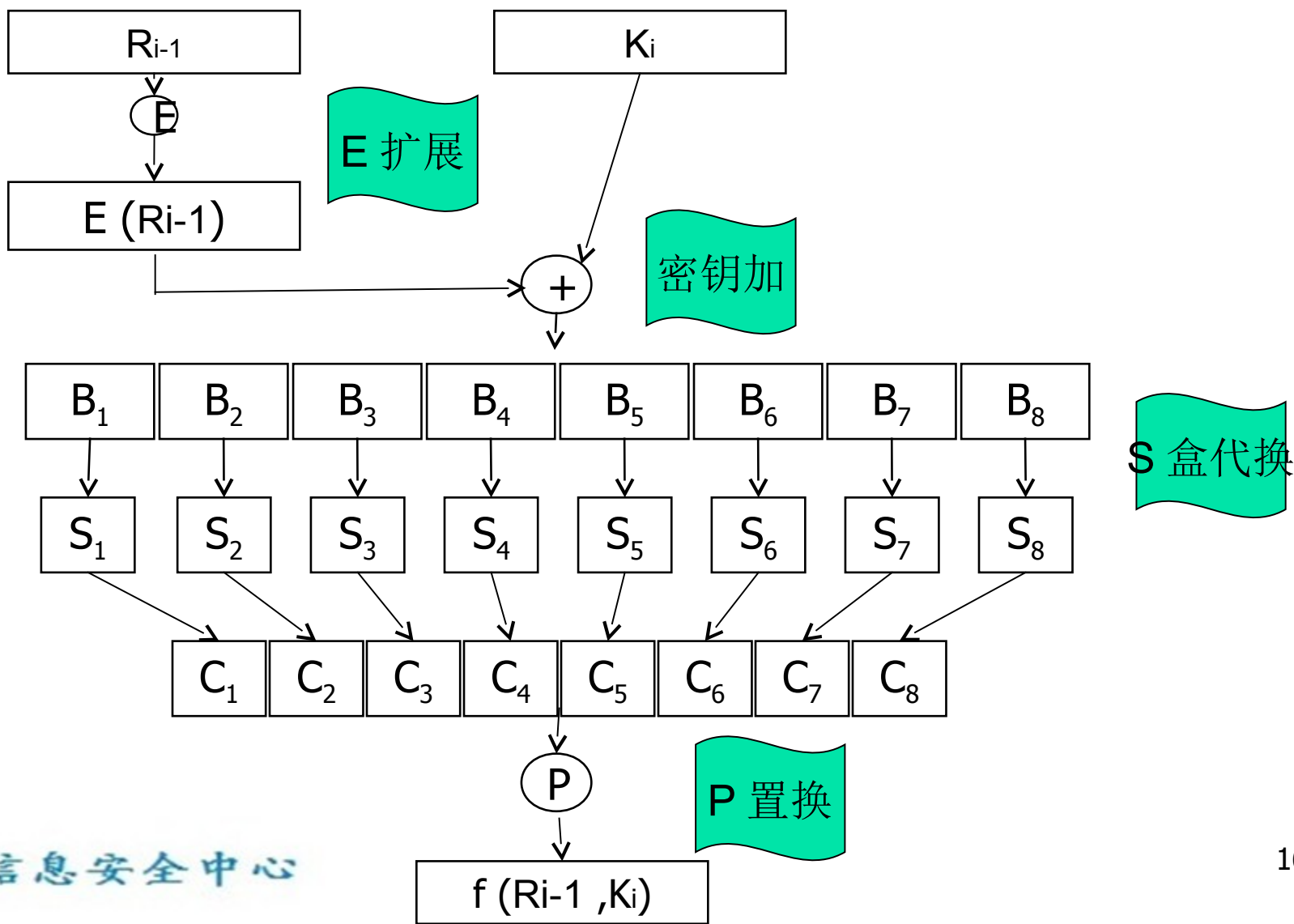
# 分组密码的轮函数

函数  $f$  以长度为 32 比特串  $R_{i-1}$  作为第一输入，以长度为 48 比特串  $K_i$  作为第二个输入，产生长度为 32 比特的输出：





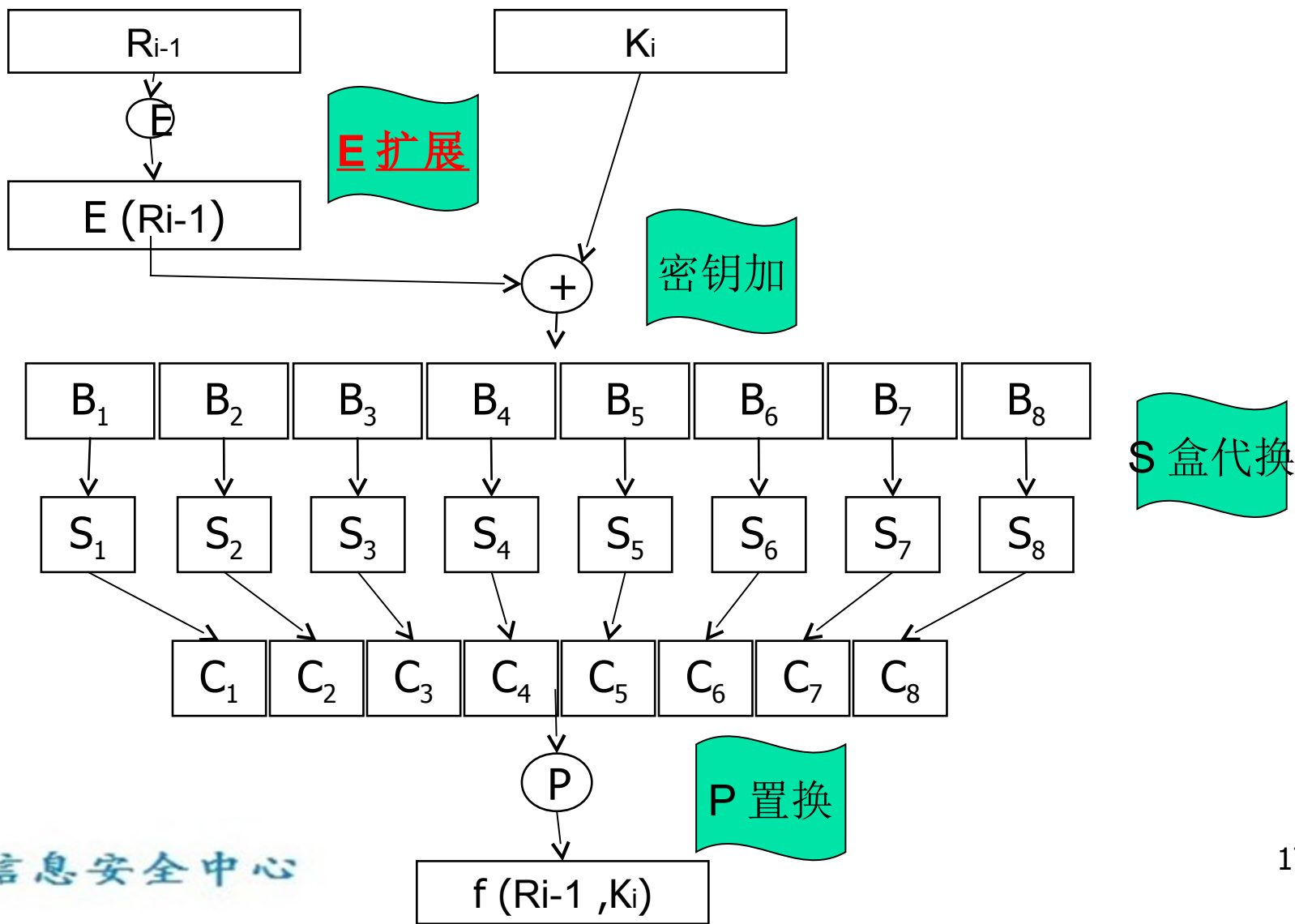
# 分组密码的轮函数







# 分组密码的轮函数



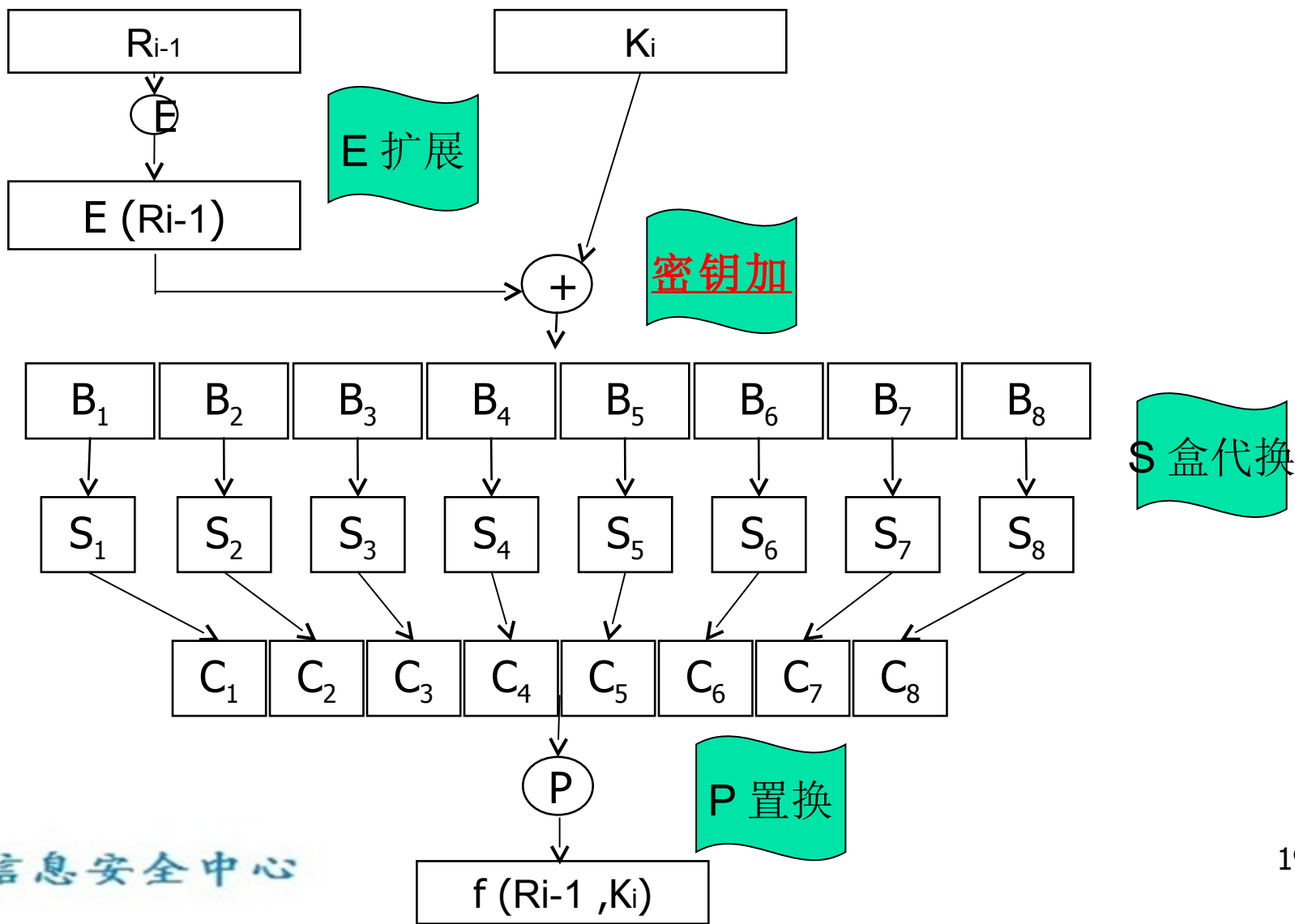
# 分组密码的轮函数

**E 扩展：**  $R_{i-1}$  根据扩展规则扩展为 48 比特长度的串；

E 比特——选择表					
32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1



# 分组密码的轮函数





# 分组密码的轮函数

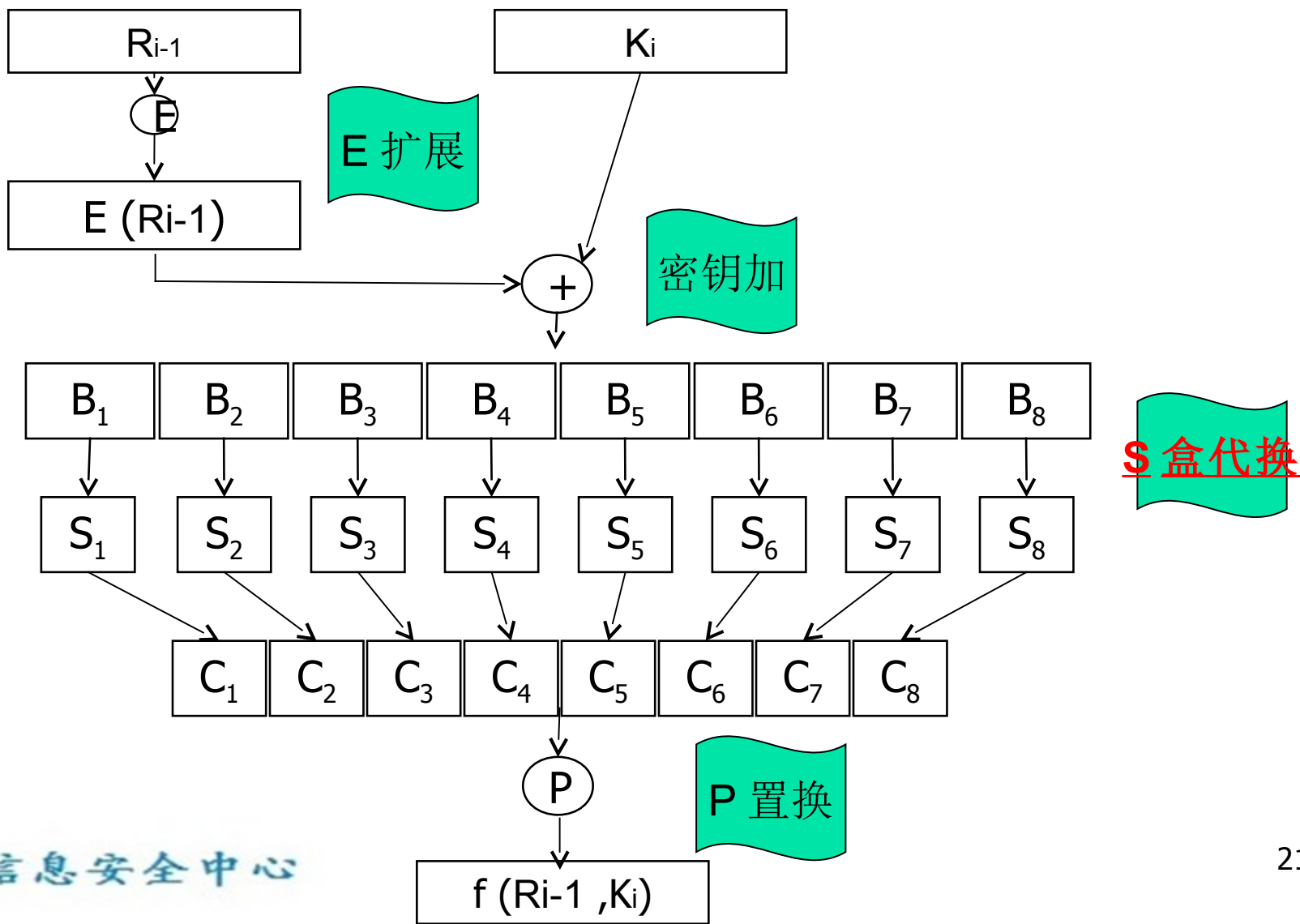
**密钥加**：计算  $E(R_{i-1}) \oplus K_i$  ，并将结果写成 8

个比特串，每个 6 比特，  $B=B_1B_2B_3B_4B_5B_6B_7B_8$ .





# 分组密码的轮函数





# 分组密码的轮函数

**S 盒代换**：使用 8 个 S 盒  $S_1 \cdots S_8$ 。每个  $S_i$  是一个固定的  $4 \times 16$  阶矩阵，其元素取  $0 \sim 15$  之间的整数。

给定长度为 6 的比特串，如

$B_j = b_1 b_2 b_3 b_4 b_5 b_6$ ， $S_j(B_j)$  计算如下：

1)  $b_1 b_6$  两个比特确定了  $S_j$  的行  $r$  的二进制表示 ( $0 \leq r \leq 3$ )，

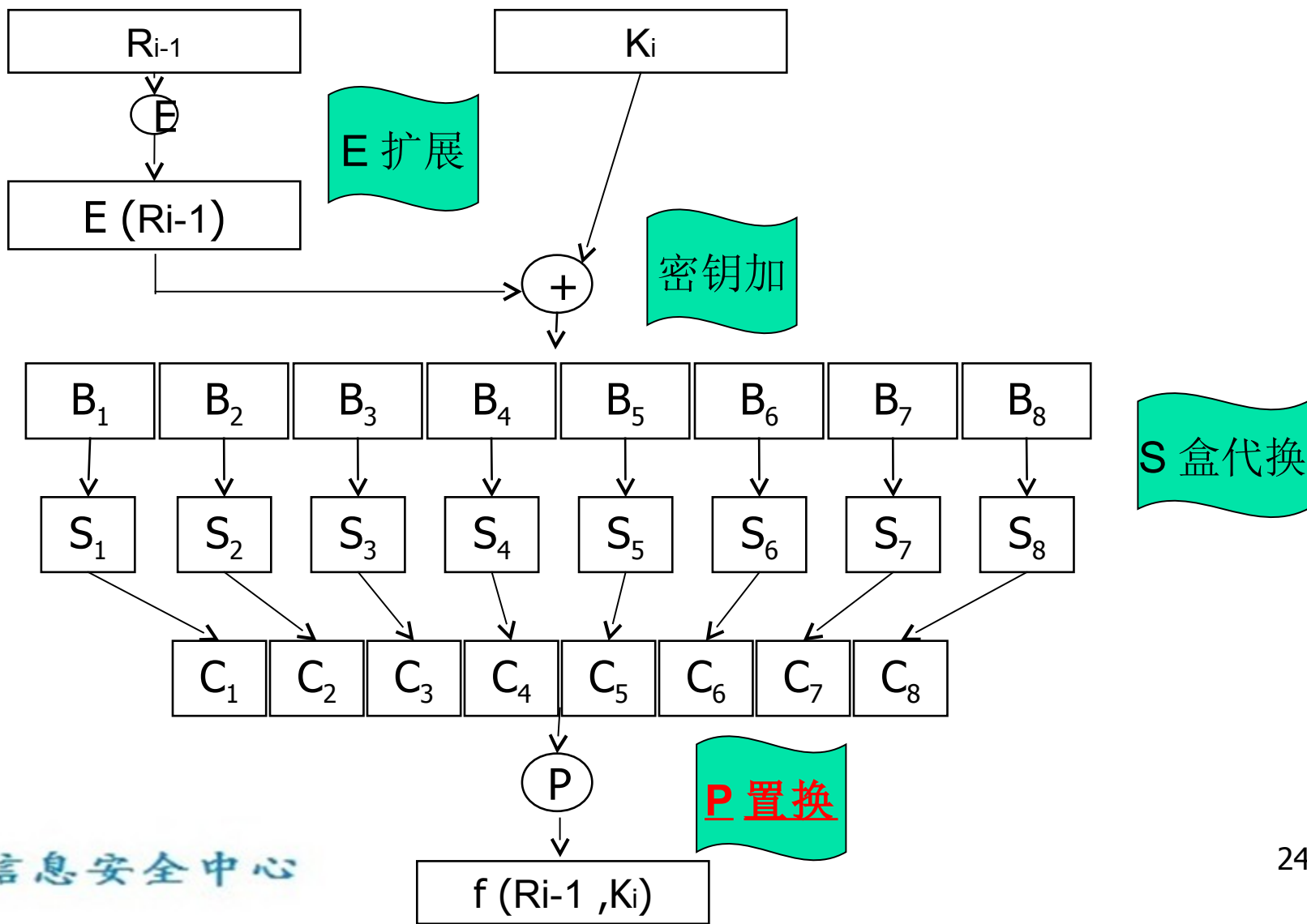
2)  $b_2 b_3 b_4 b_5$  四个比特确定了  $S_j$  的列  $c$  的二进制表示 ( $0 \leq c \leq 15$ )，

3)  $S_j(B_j)$  定义成长度为 4 的比特串的值  $S_j(r, c)$ 。由此可以算出  $C_j = S_j(B_j)$ ,  $1 \leq j$

$S_1$															
14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	15	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13
$S_2$															
15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
$S_3$															
10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
$S_4$															
7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
12	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14
$S_5$															
2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
$S_6$															
12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
$S_7$															
4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
$S_8$															
13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11



# 分组密码的轮函数





# 分组密码的轮函数

**P 置换**：长度为 32 比特串  $C=C_1C_2C_3C_4C_5C_6C_7C_8$ ，  
根据固定置换  $P(*)$  进行置换，得到比特串  $P(C)$ 。

P 置换			
16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25



# 本节主要内容

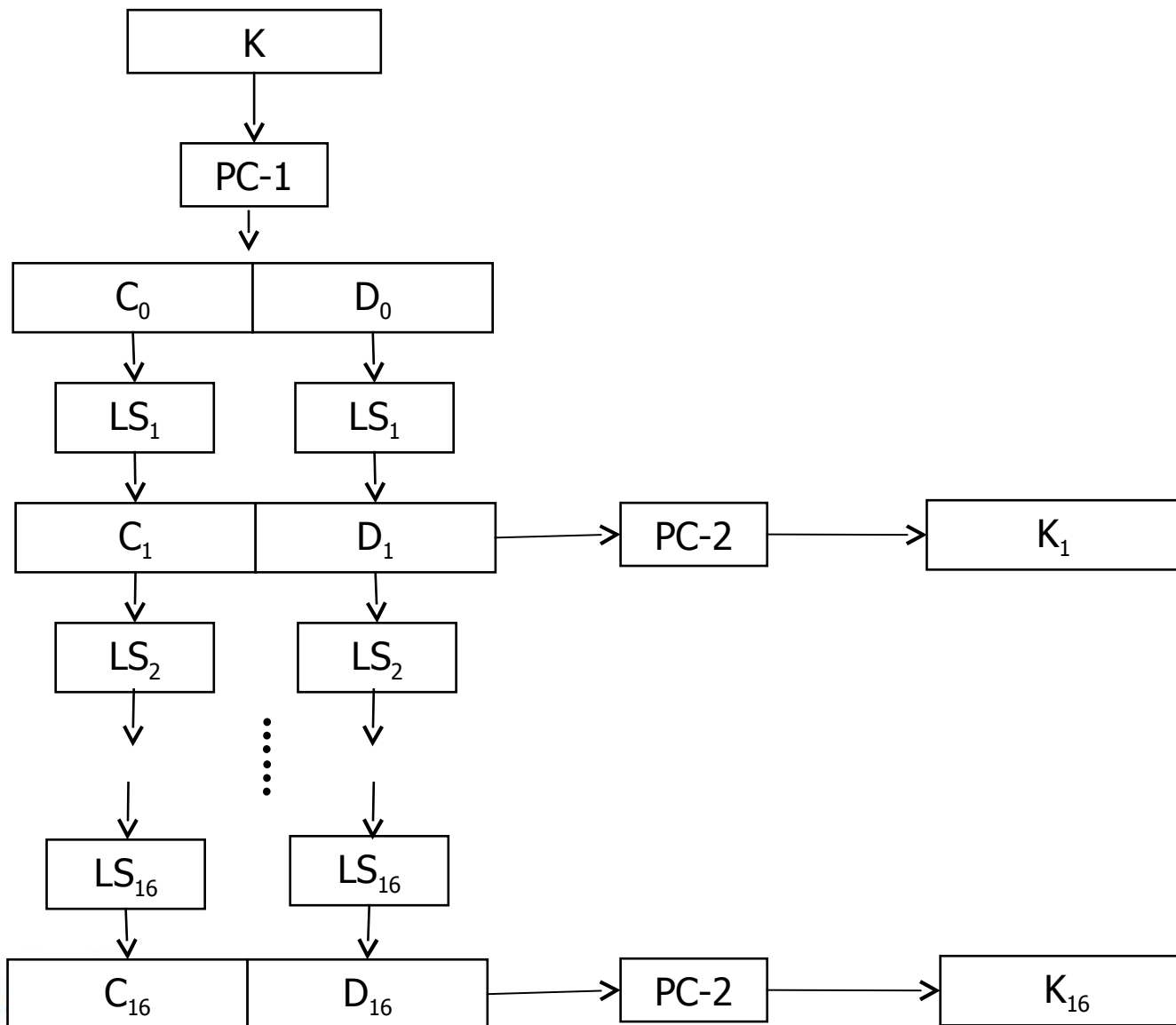
- DES 算法的整体结构—— Feistel 结构
- DES 算法的轮函数
- **DES 算法的密钥编排算法**
- DES 的解密变换





# DES 算法的密钥编排算法

每轮中所使用的子密钥  
根据密钥来获得的



# DES 算法的密钥编排算法

- 给定 64 比特密钥  $K$ ，根据固定的置换  $PC-1$  来处理  $K$  得到  $PC-1(K) = C_0D_0$ ，其中  $C_0$  和  $D_0$  分别由最前和最后 28 比特组成

PC-1						
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4



# DES 算法的密钥编排算法

- 计算  $C_i = LS_i(C_{i-1})$  和  $D_i = LS_i(D_{i-1})$ ，且  $K_i = PC-2(C_i D_i)$ ， $LS_i$  表示循环左移两个或一个位置，具体地，如果  $i=1, 2, 9, 16$  就移一个位置，否则就移两个位置， $PC-2$  是另一个固定的置换。





# DES 算法的密钥编排算法

PC-2					
14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32



# DES 算法的密钥编排算法

- 对  $1 \leq i \leq 16$  , DES 的每一轮中使用 K 的 56 比特中的 48 个比特, 具体选取位置由下表确定





轮 1

10	51	34	60	49	17	33	57	2	9	19	42
3	35	26	25	44	58	59	1	36	27	18	41
22	28	39	54	37	4	47	30	5	53	23	29
61	21	38	63	15	20	45	14	13	62	55	31

轮 2

2	43	26	52	41	9	25	49	59	1	11	34
60	27	18	17	36	50	51	58	57	19	10	33
14	20	31	46	29	63	39	22	28	45	15	21
53	13	30	55	7	12	37	6	5	54	47	23

轮 3

51	27	10	36	25	58	9	33	43	50	60	18
44	11	2	1	49	34	35	42	41	3	59	17
61	4	15	30	13	47	23	6	12	29	62	5
37	28	14	39	54	63	21	53	20	38	31	7

轮 4

35	11	59	49	9	42	58	17	27	34	44	2
57	60	51	50	33	18	19	26	25	52	43	1
45	55	62	14	28	31	7	53	63	13	46	20
21	12	61	23	38	47	5	37	4	22	15	54

轮 5;

19	60	43	33	58	26	42	1	11	18	57	51
41	44	35	34	17	2	3	10	9	36	27	50
29	39	46	61	12	15	54	37	47	28	30	4
5	63	45	7	22	31	20	21	55	6	62	38

轮 6

3	44	27	17	42	10	26	50	60	2	41	35
25	57	19	18	1	51	52	59	58	49	11	34
13	23	30	45	63	62	38	21	31	12	14	55
20	47	29	54	6	15	4	5	39	53	46	22

轮 7

52	57	11	1	26	59	10	34	44	51	25	19
9	41	3	2	50	35	36	43	42	33	60	18
28	7	14	29	47	46	22	5	15	63	61	39
4	31	13	38	53	62	55	20	23	37	30	6

轮 8

36	41	60	50	10	43	59	18	57	35	9	3
58	25	52	51	34	19	49	27	26	17	44	2
12	54	61	13	31	30	6	20	62	47	45	23
55	15	28	22	37	46	39	4	7	21	14	53





轮 9

57	33	52	42	2	35	51	10	49	27	1	60
50	17	44	43	26	11	41	19	18	9	36	59
4	46	53	55	23	22	61	12	54	39	37	15
47	7	20	14	29	38	31	63	62	13	6	45

轮 13

58	34	17	43	3	36	52	11	50	57	2	35
51	18	9	44	27	41	42	49	19	10	1	60
7	45	20	39	22	21	28	15	53	38	4	14
46	6	23	13	63	37	30	62	61	47	5	12

轮 10

41	17	36	26	51	19	35	59	33	11	50	44
34	1	57	27	10	60	25	3	2	58	49	43
55	30	37	20	7	6	45	63	38	23	21	62
31	34	4	61	13	22	15	47	46	28	53	29

轮 14

42	18	1	27	52	49	36	60	34	41	51	9
35	2	58	57	11	25	26	33	3	59	50	44
54	29	4	23	6	5	12	62	37	22	55	61
30	53	7	28	47	21	14	46	45	31	20	63

轮 11

25	1	49	10	35	3	19	43	17	60	34	57
18	50	41	11	59	44	9	52	51	42	33	27
39	14	21	4	54	53	29	47	22	7	5	46
15	38	55	45	28	6	62	31	30	12	37	13

轮 15

26	2	50	11	36	33	49	44	18	25	35	58
19	51	42	41	60	9	10	17	52	43	34	57
38	13	55	7	53	20	63	46	21	6	39	45
14	37	54	12	31	5	61	30	29	15	4	47

轮 12

9	50	33	59	19	52	3	27	1	44	18	41
2	34	25	60	43	57	58	36	35	26	17	11
23	61	5	55	38	37	13	31	6	54	20	30
62	22	39	29	12	53	46	15	14	63	21	28

轮 16

18	59	42	3	57	25	41	36	10	17	27	50
11	43	34	33	52	1	2	9	44	35	26	49
30	5	47	62	45	12	55	38	13	61	31	37
6	29	46	4	23	28	53	22	21	7	63	39



# 本节主要内容

- 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**

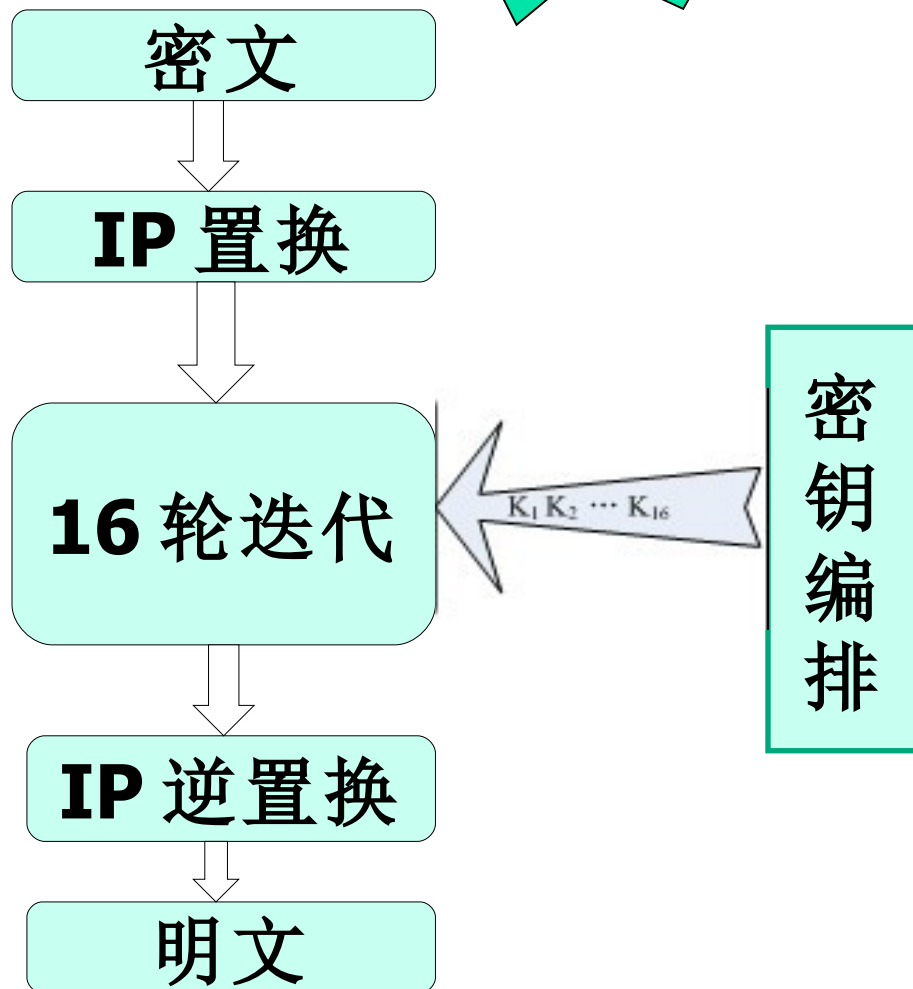




# DES 的解密变换

DES 的解密与加密一样使用相同的算法，它以密文  $y$  作为输入，但以相反的顺序使用密钥编排  $K_{16}, K_{15}, \dots, K_1$ ，输出的是明文  $x$

为什么？





# 课堂练习

1. 设明文为 0123456789ABCDEF (16 进制)，密钥为 133457799BBCDFF1，求第一轮加密后的数据流。

2. 应用 IP，得到二进制的  $L_0R_0$  为：

$L_0=11001100000000001100110011111111$

$L_1=R_0=11110000101010101111000010101010$



# 课堂练习

## 第 1 轮

$$E(R_0) = 011110100001010101010101011110100001010101010101$$

$$K_1 = 000110110000001011101111111111000111000001110010$$

$$E(R_0) \oplus K_1 = 011000010001011110111010100001100110010100100111$$

$$S\text{-boxoutputs} = 01011100100000101011010110010111$$

$$f(R_0, K_1) = 00100011010010101010100110111011$$

$$L_2 = R_1 = 11101111010010100110010101000100$$



# 主要知识点小结

- 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





**THE END !**

