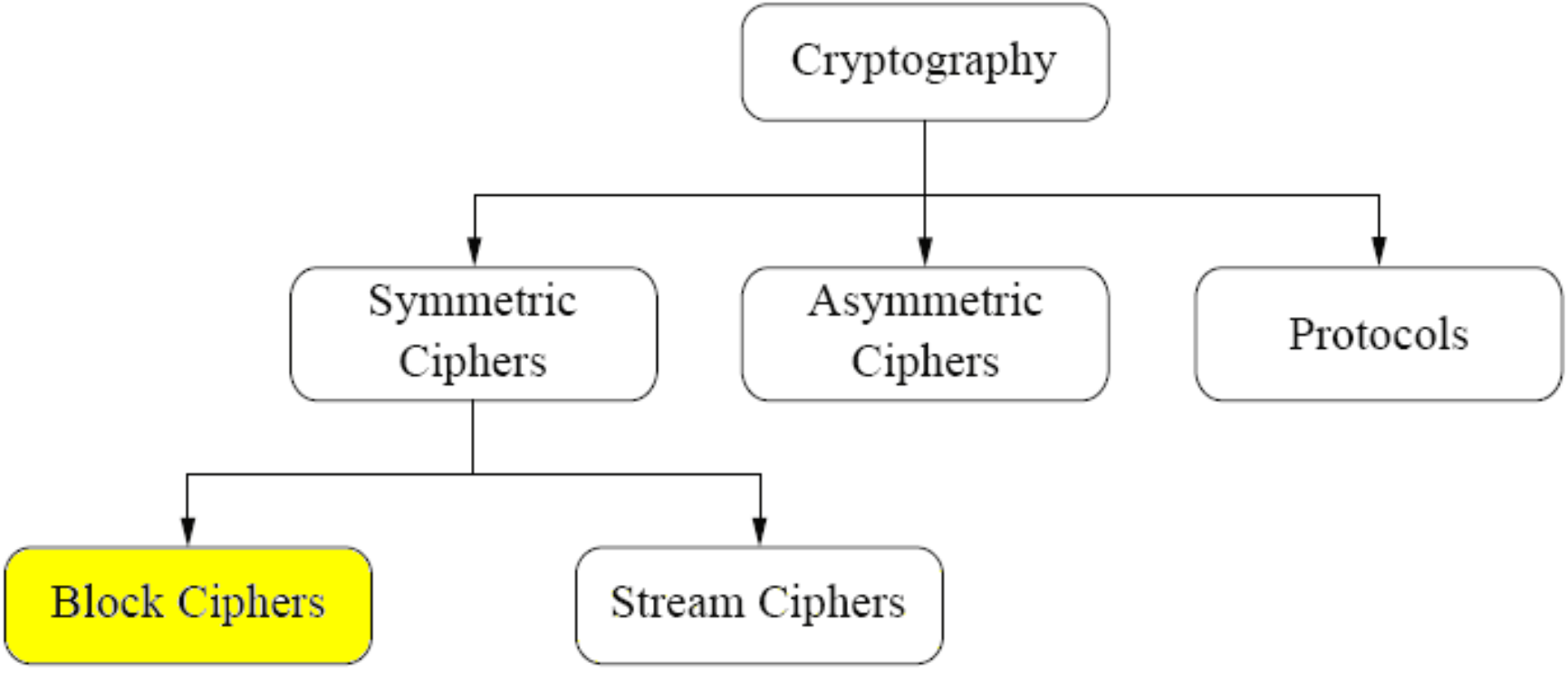
Introdução à Criptografia

Cifras de Bloco
DES

Prof. Rodrigo Minetto
rminetto@dainf.ct.utfpr.edu.br
Universidade Tecnológica Federal do Paraná
Material compilado de: Understanding Cryptography by Christof Paar e Jan Pelzl

Sumário

Algoritmos para criptografia



Introdução

Em 1960, devido a invenção do circuito integrado, os computadores se tornaram mais poderosos e baratos (em 1953 a IBM lançou o seu primeiro computador e em 1957 o Fortran). Contudo, à medida que mais e mais empresas compravam computadores e as cifragens entre elas se difundiam surgiram preocupações com relação à padronização. Em 1973, o National Bureau of Standards americano planejou resolver o problema e formalmente solicitou propostas para um sistema padrão de cifragem que permitisse conversas secretas entre empresas.

Introdução

Horst Feistel, um entusiasta alemão de criptografia, emigrou para os EUA em 1934 e devido a segunda guerra mundial chegou a ficar em prisão domiciliar. Feistel começou suas pesquisas em cifras no Centro de Pesquisas Cambridge da Força Aérea e logo encontrou problemas com a NSA. A NSA é a organização que mais emprega matemáticos e intercepta mensagens no mundo. A NSA não fazia objeções quanto ao passado de Feistel, meramente queria manter o monopólio da pesquisa criptográfica e assim arranjou para que o trabalho de pesquisa dele fosse cancelado várias vezes.

Introdução

Após alguns anos, Feistel se mudou para o laboratório Thomas J. Watson da IBM, perto de Nova York, onde finalmente conseguiu realizar sua pesquisa sem ser importunado. No início da década de 1970 ele desenvolveu o sistema Lucifer. Lucifer logo se tornou um dos mais poderosos sistemas de cifragem disponíveis comercialmente, e consequentemente foi usado por uma grande variedade de organizações.

Introdução

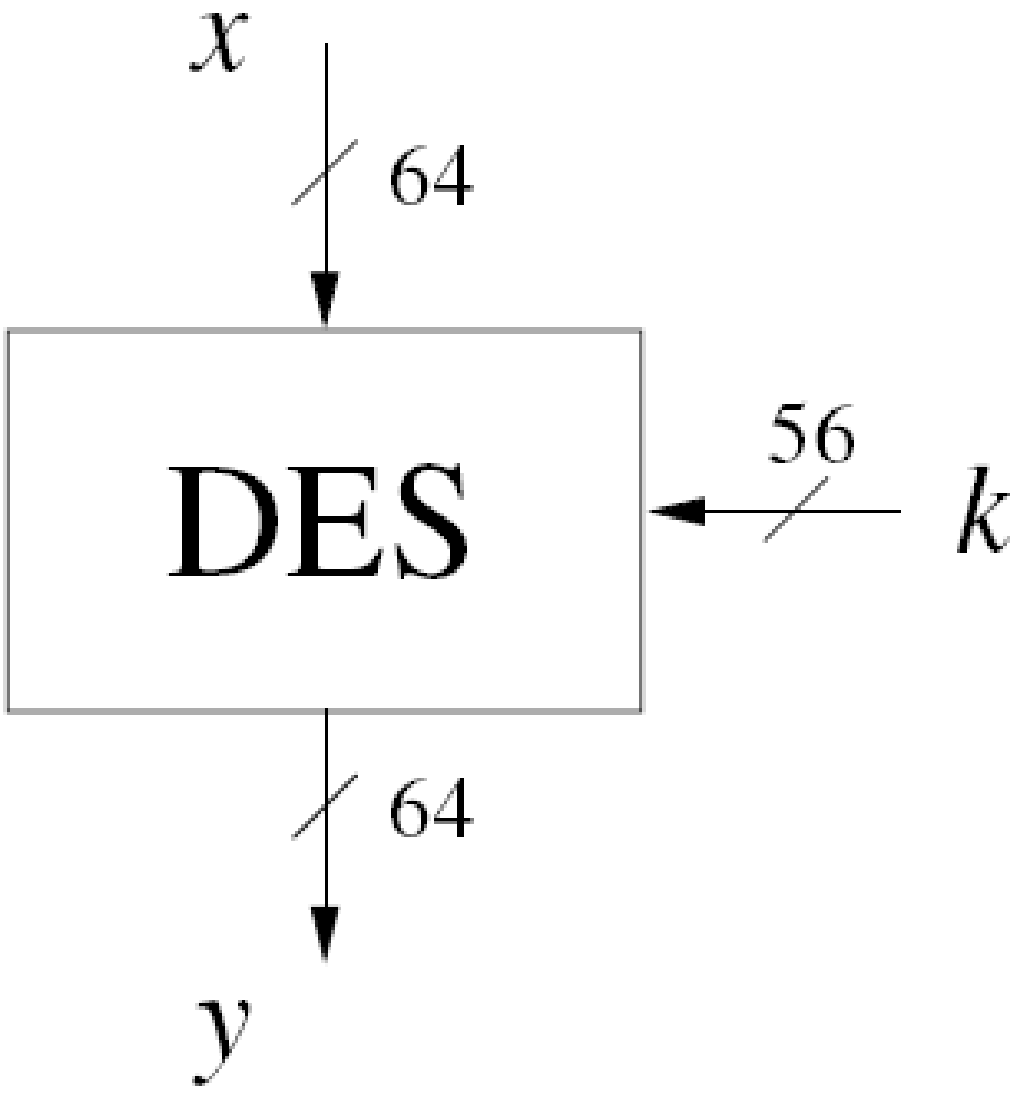
Era inevitável que o sistema Lucifer fosse adotado como padrão. O problema era que Lucifer era tão poderoso que oferecia a possibilidade de um padrão de cifragem além das capacidades de quebra de códigos da NSA. O rumor é que a NSA pressionou para enfraquecer um aspecto de Lucifer, o número de chaves possíveis, antes de permitir que ele fosse adotado como padrão. A NSA argumentou em limitar o número de chaves a aproximadamente 100.000.000.000.000.000 (56 bits). O sistema original utilizava 128 bits para a chave.

Introdução

A NSA acreditava que esse número de bits fornecia segurança dentro da comunidade civil. Contudo a NSA tinha acesso aos maiores sistemas de computação do mundo e seria capaz de decifrar as mensagens. A versão de 56 bits da cifra Lucifer foi adotada em 1976 e batizada como DES - Data Encryption Standard (Padrão de Cifragem de Dados). O algoritmo DES só deixou de ser utilizado devido à ataques por força-bruta (tempo de decifragem em até 22 horas), ou seja, nenhuma vulnerabilidade séria na cifra foi encontrada até hoje.

Sumário

Data Encryption Standard (DES)



Data Encryption Standard (DES)

Claude Shannon definiu duas operações primitivas básicas que algoritmos para criptografia forte deveriam se basear:

- **Confusão**: operação para ocultar a relação entre o texto em claro e o texto cifrado. A **substituição**, presente no DES e AES, é um elemento comum para alcançar confusão.

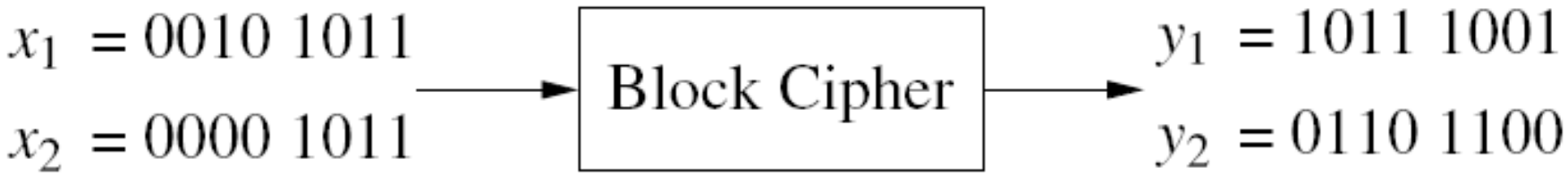
Data Encryption Standard (DES)

Claude Shannon definiu duas operações primitivas básicas que algoritmos para criptografia forte deveriam se basear:

- **Difusão**: operação para dissipar a redundância do texto em claro, pulverizando-a no texto cifrado. A **permutação**, presente no DES e AES, é um elemento comum para alcançar difusão.

Data Encryption Standard (DES)

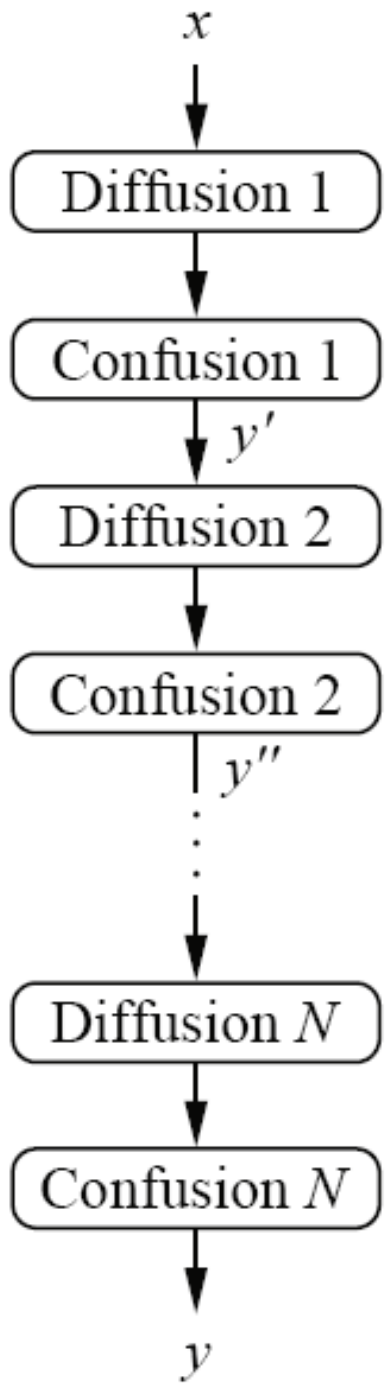
Cifras de bloco modernas têm excelente difusão: modificando um bit do texto em claro resulta em média na mudança de metade dos bits da saída (independência estatística).



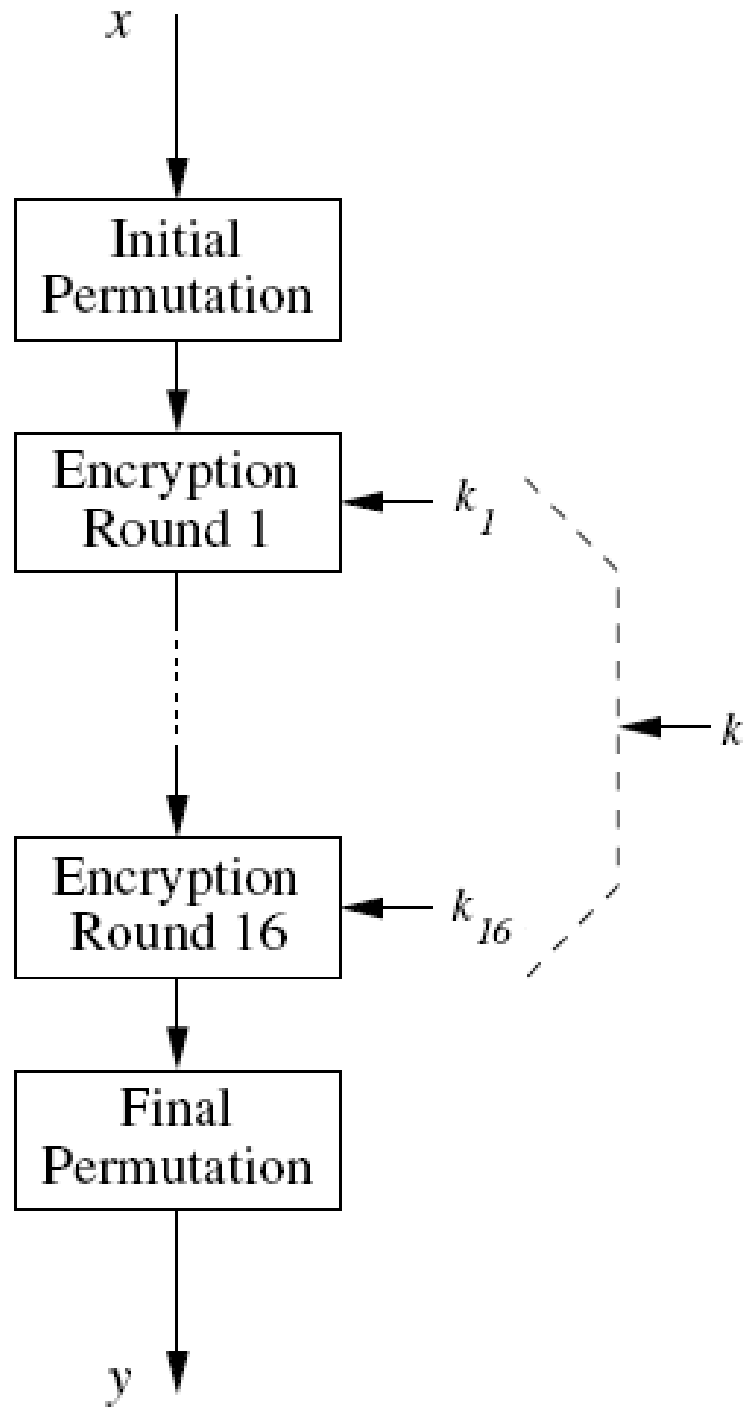
Introdução

Difusão e confusão: não produzem segurança por si próprias. A ideia é concatenar elementos de confusão e difusão para construir as chamadas cifras de produto. O projeto de cifras modernas de bloco baseia-se no conceito de uma cifra de produto iterada (várias rodadas).

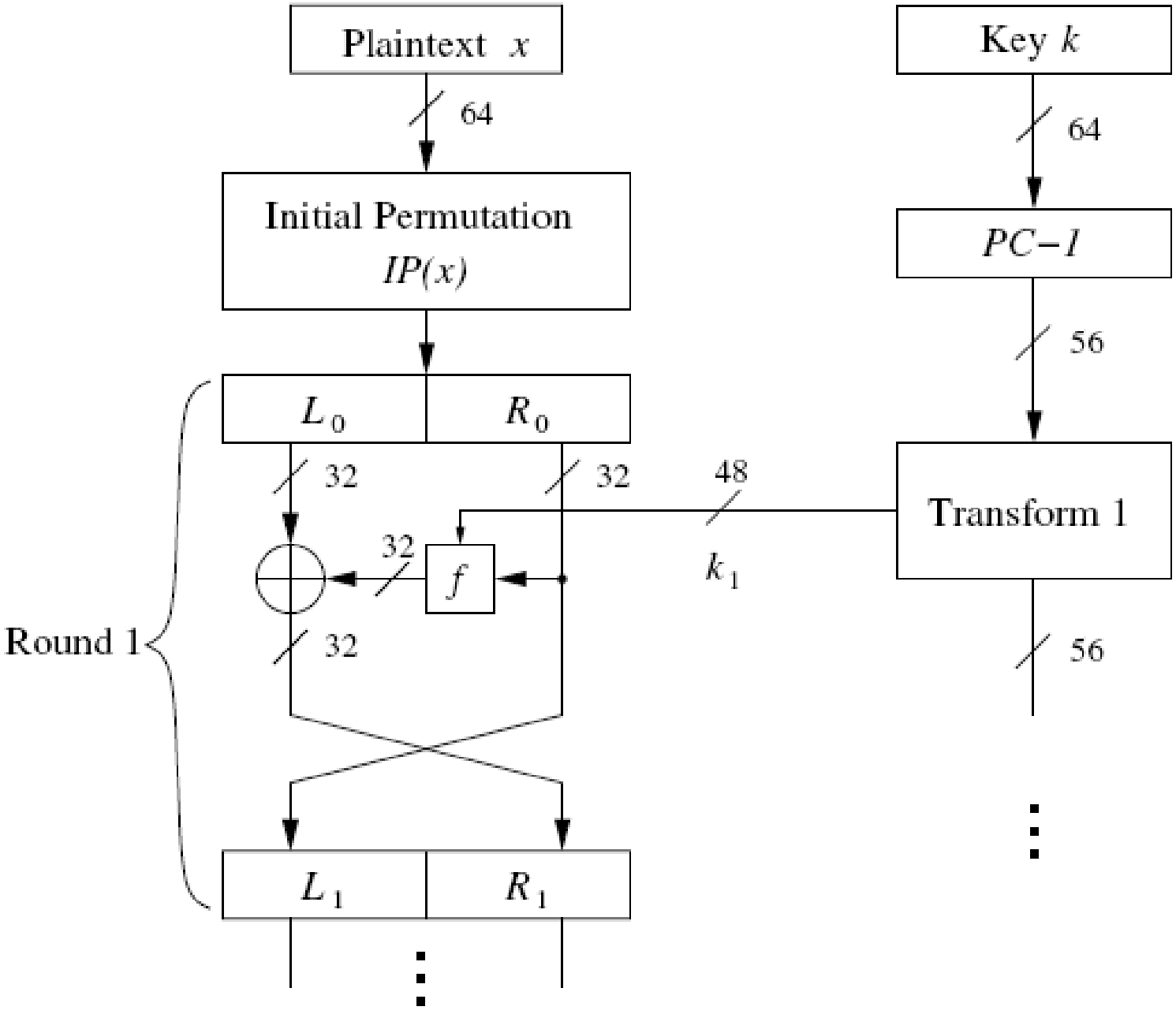
Cifra de **Produto**



Estrutura do **DES**



Data Encryption Standard (DES)

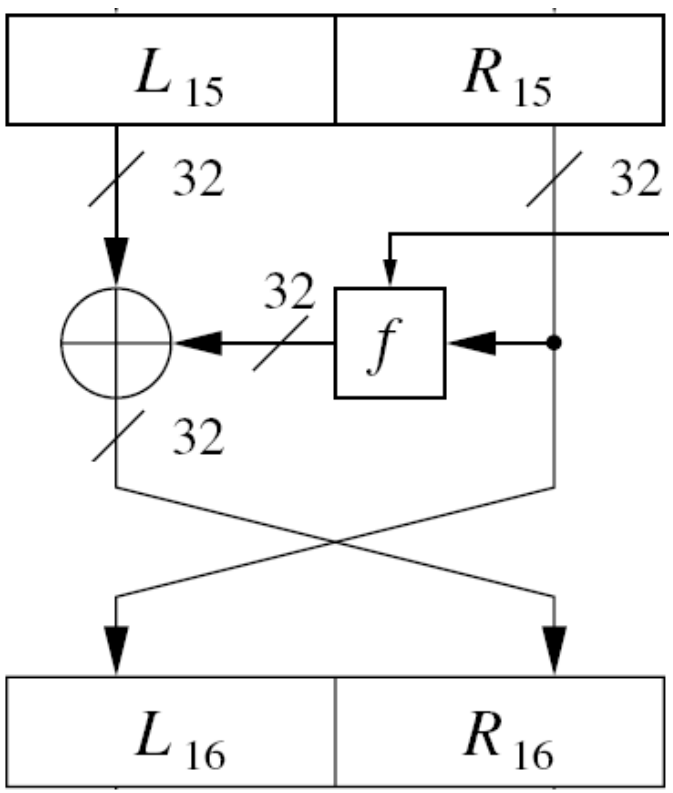


Data Encryption Standard (DES)

Redes de Feistel (Feistel network):

$$L_{i+1} = R_i$$
$$R_{i+1} = L_i \oplus f(R_i, k_i)$$

para $i = 1, \ldots, 16$.



Data Encryption Standard (DES)

Redes de Feistel: ciframento

$$\begin{aligned} L_{i+1} &= R_i \\ R_{i+1} &= L_i \oplus f(R_i, k_i) \end{aligned}$$

para $i = 1, \ldots, 16$.

Redes de Feistel: deciframento

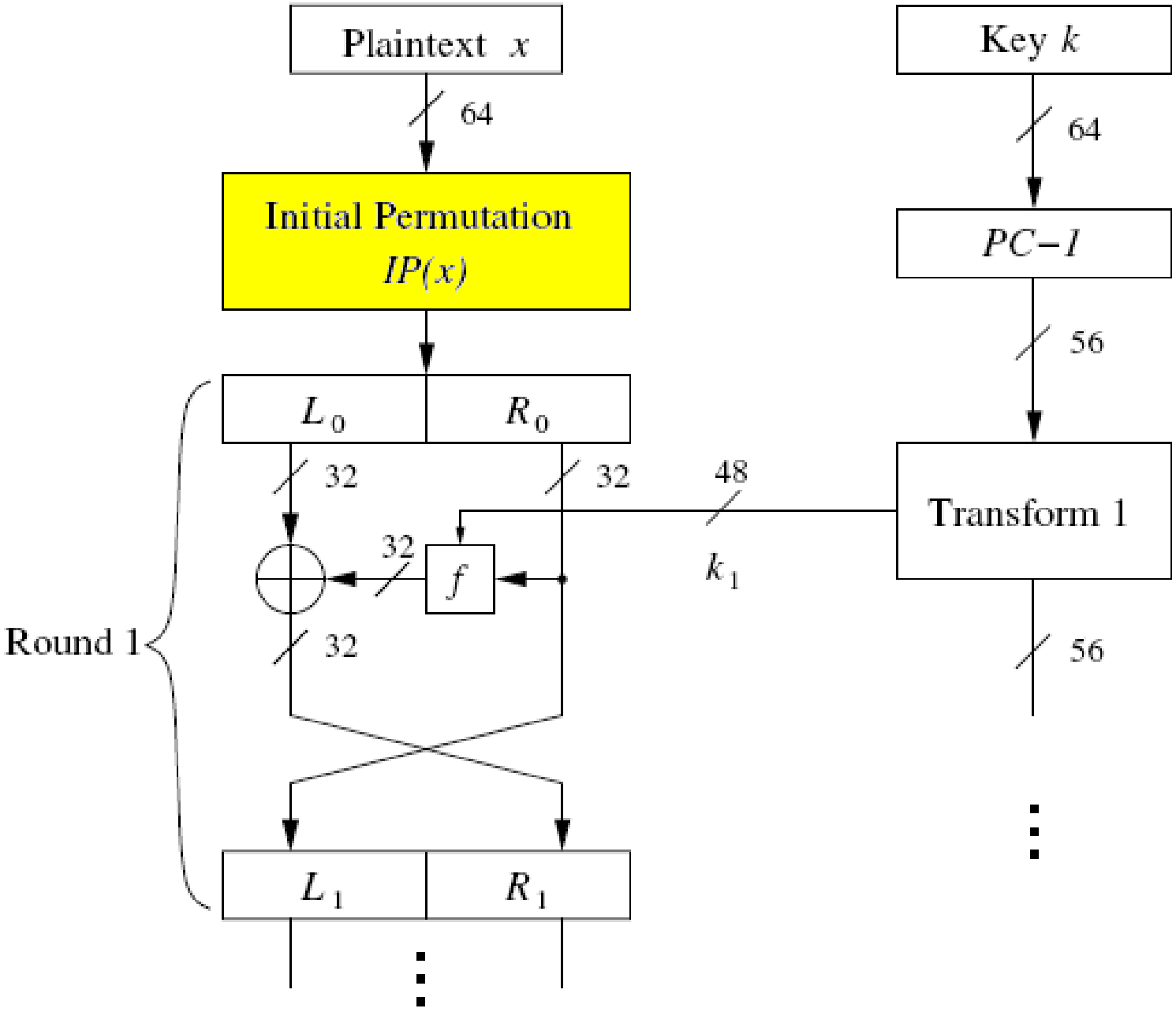
$$\begin{aligned} R_i &= L_{i+1} \\ L_i &= R_{i+1} \oplus f(L_{i+1}, k_i) \end{aligned}$$

para $i = 1, \ldots, 16$.

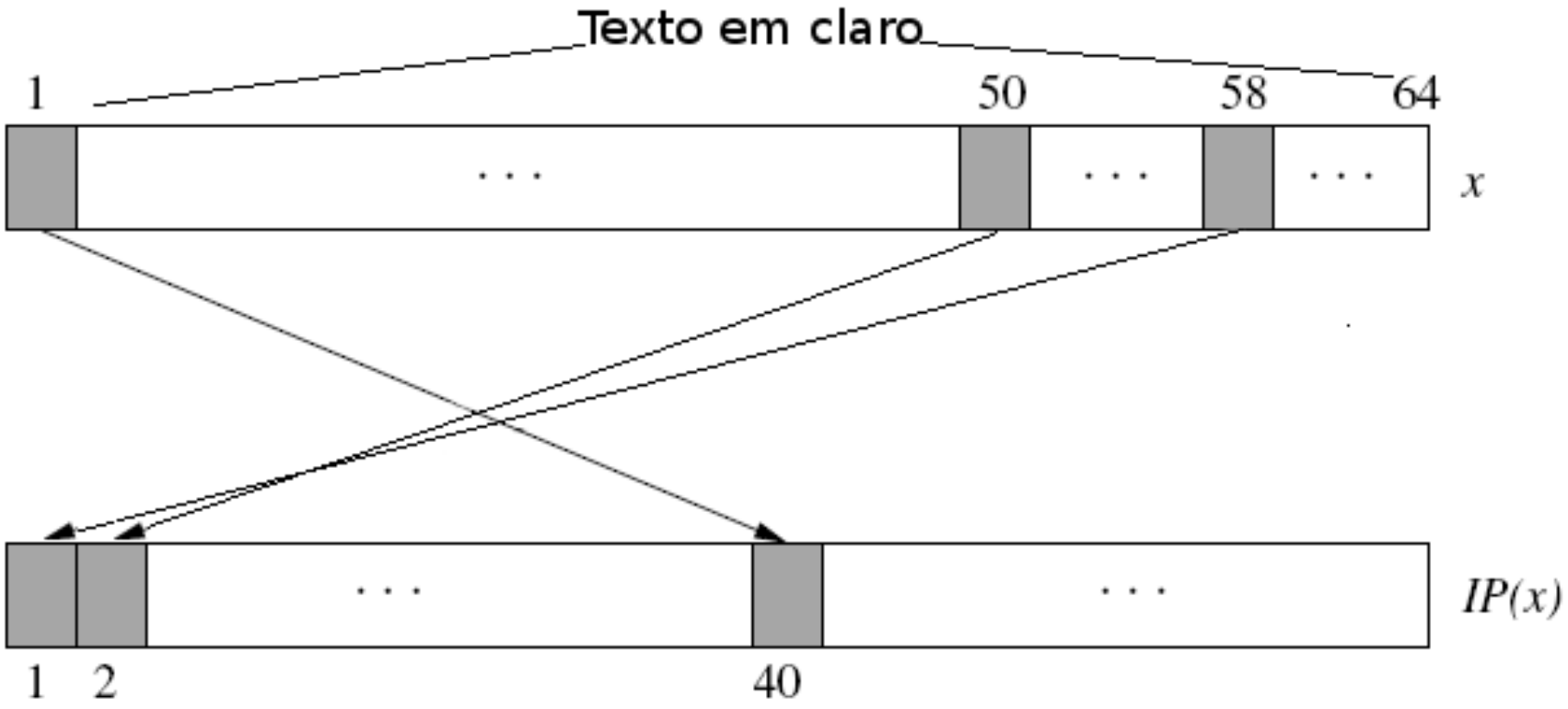
Data Encryption Standard (DES)

Redes de Feistel (Feistel network): a estrutura de Feistel tem a vantagem de que as operações de cifragem e decifragem são muito semelhantes, sendo idênticas em alguns casos, necessitando apenas da utilização das chaves na ordem inversa.

Data Encryption Standard (DES)



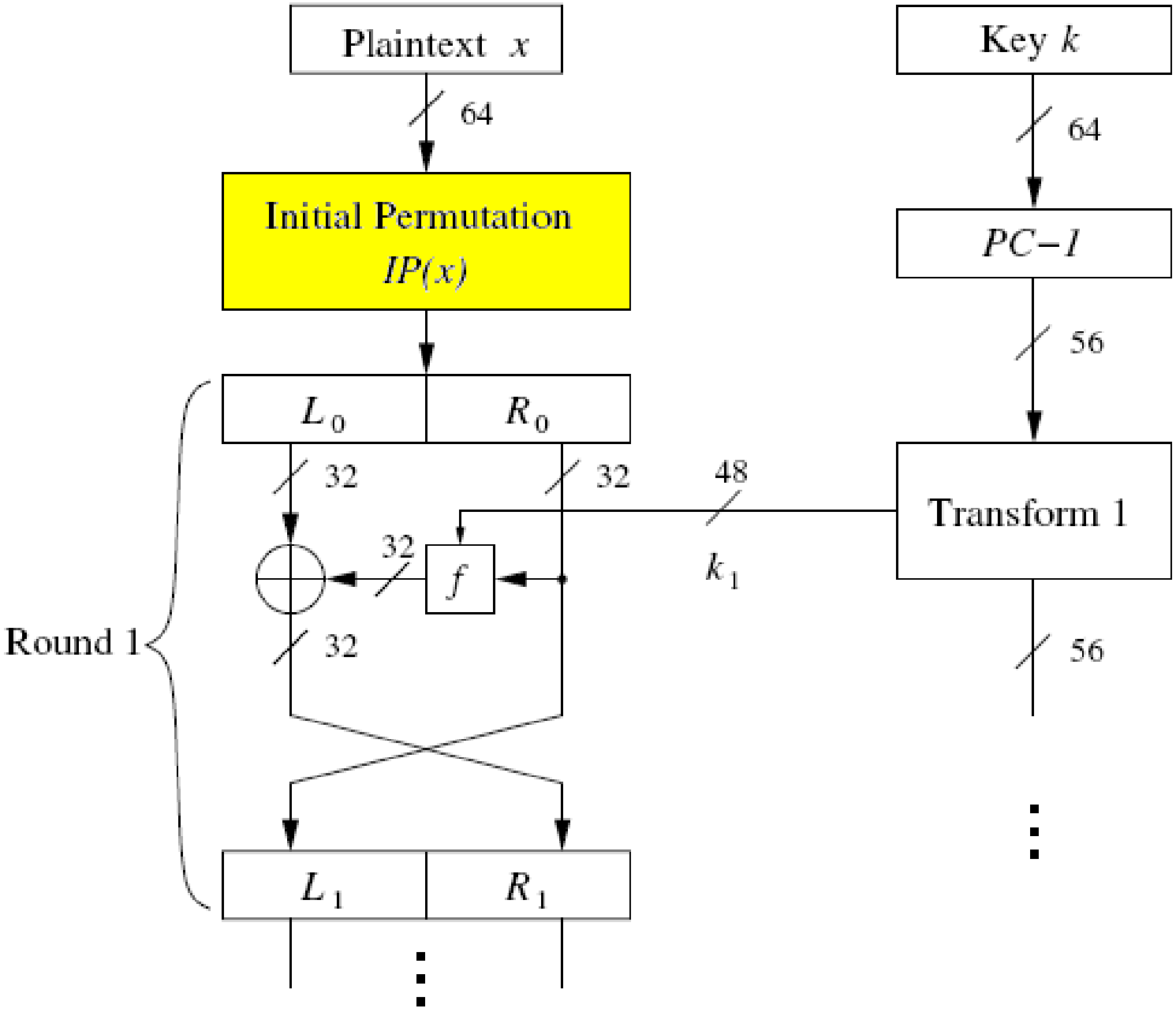
Permutação inicial (IP)



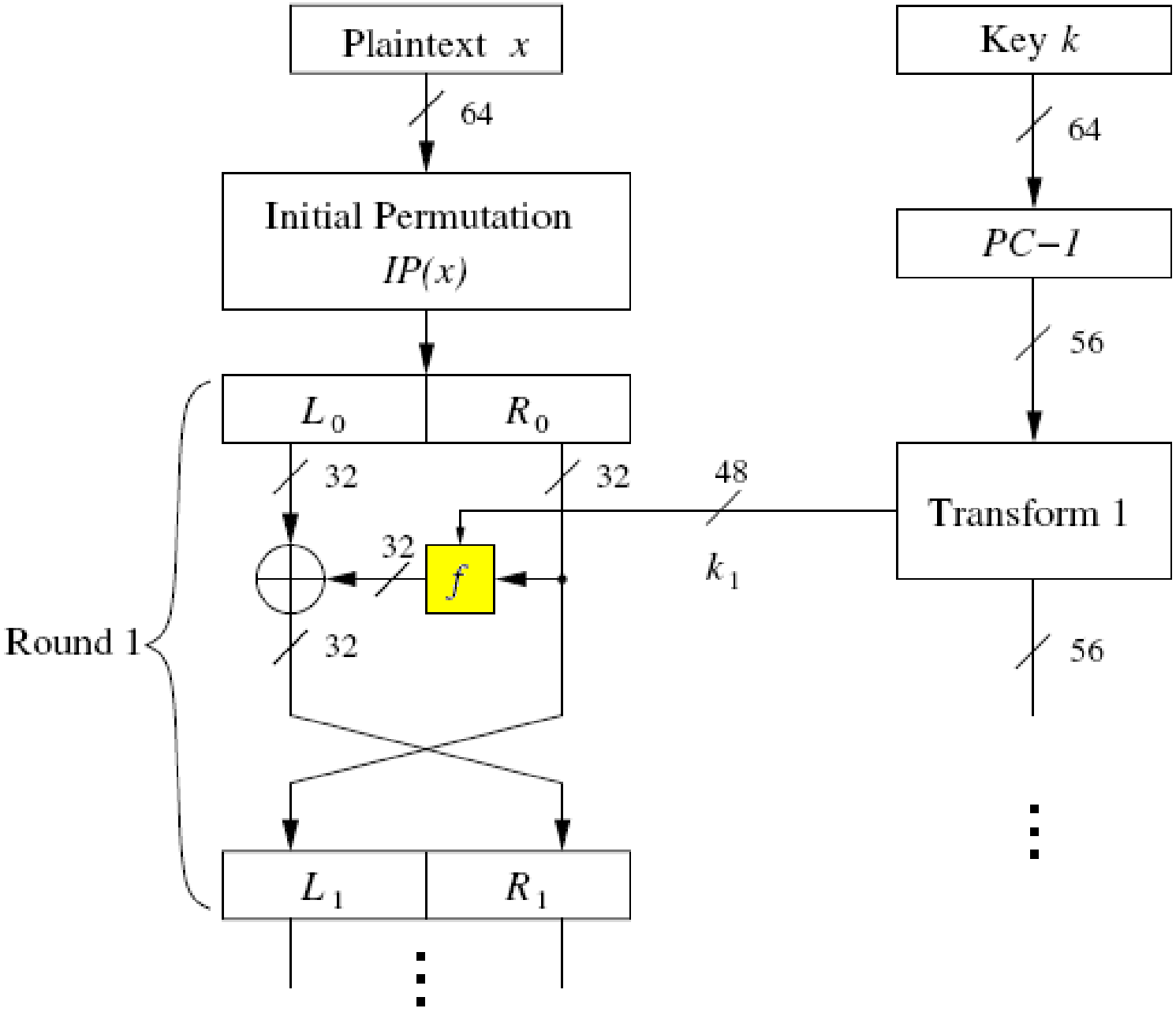
Permutação inicial (IP)

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

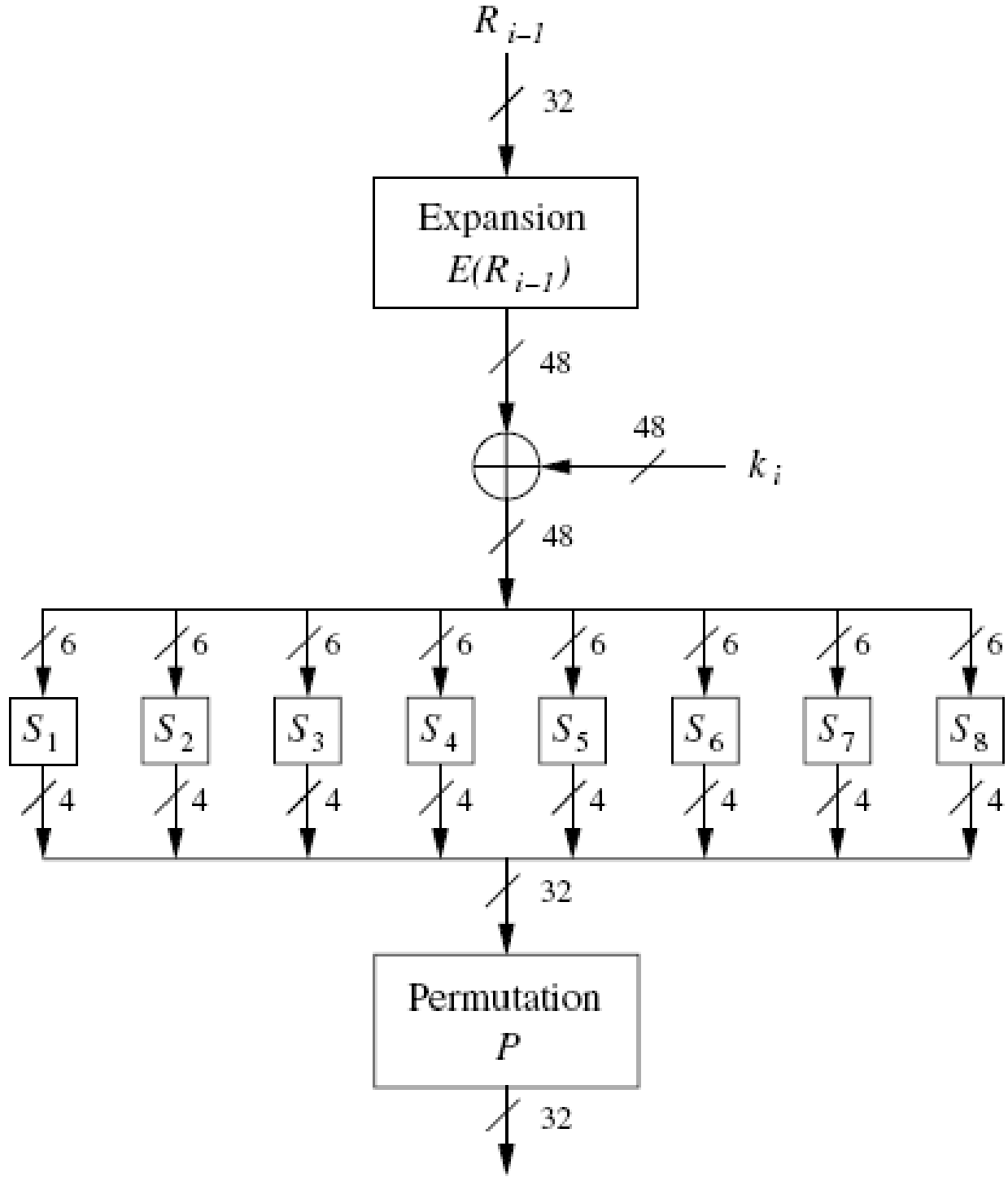
Data Encryption Standard (DES)



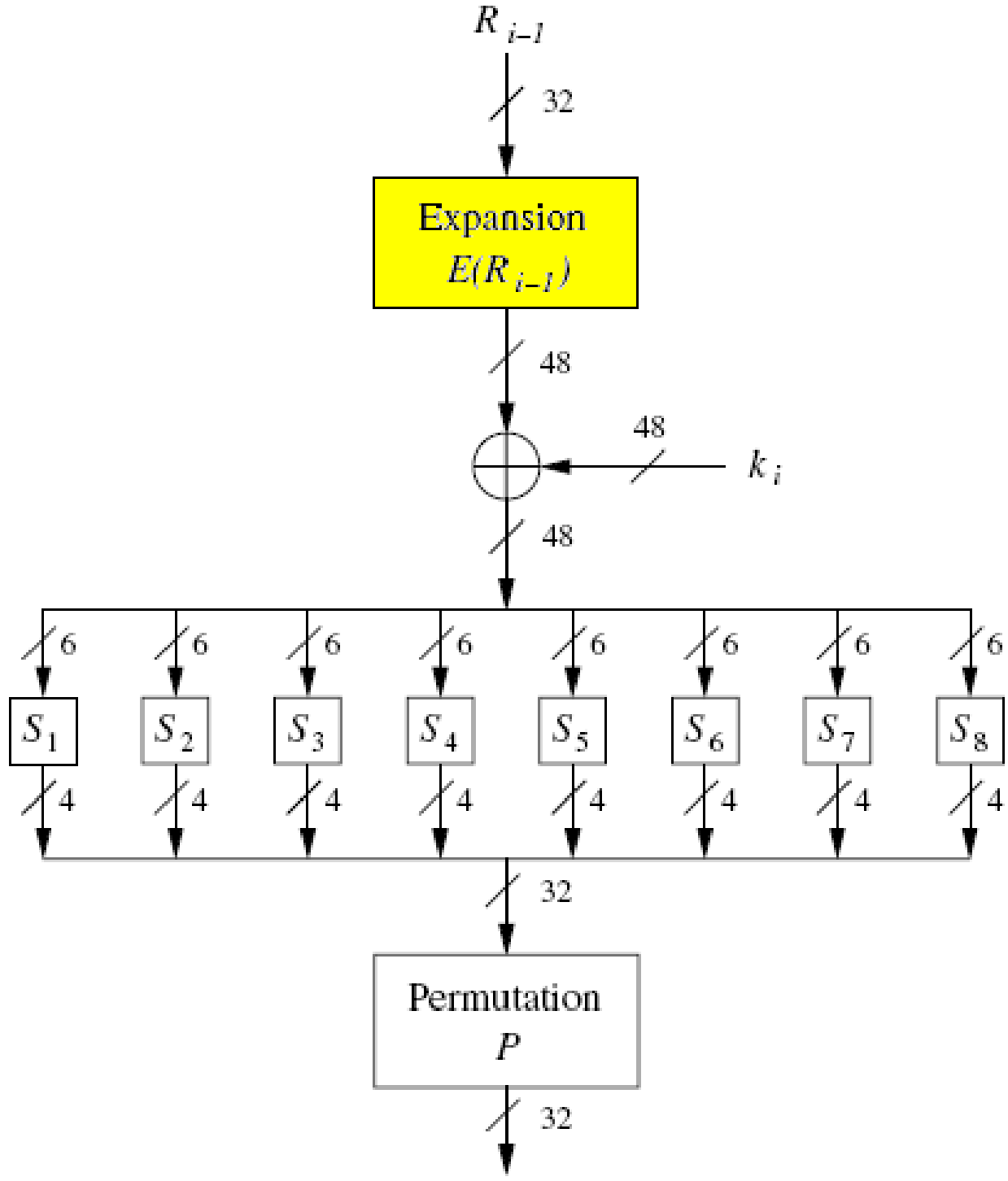
Data Encryption Standard (DES)



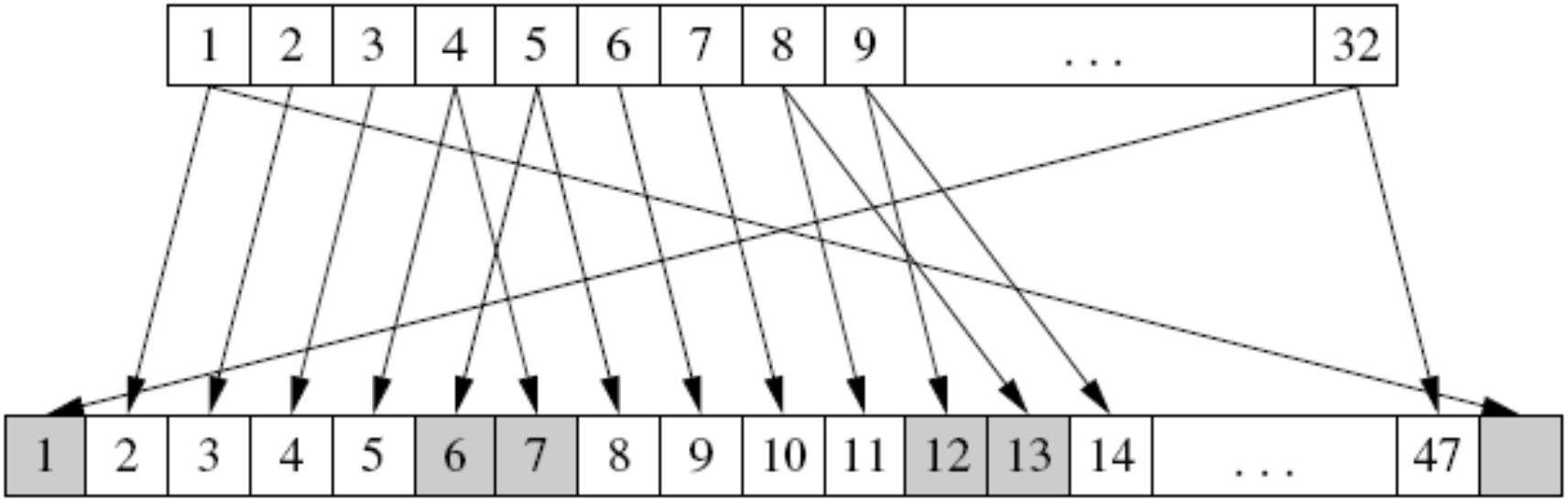
Função f


R_{i-1}
32
Expansion
$E(R_{i-1})$
48
48
k_i
48
6
6
6
6
6
6
6
6
S_1
S_2
S_3
S_4
S_5
S_6
S_7
S_8
4
4
4
4
4
4
4
4
32
Permutation
P
32

Função f


R_{i-1}
32
Expansion
$E(R_{i-1})$
48
48
k_i
48
6
6
6
6
6
6
6
6
S_1
S_2
S_3
S_4
S_5
S_6
S_7
S_8
4
4
4
4
4
4
4
4
32
Permutation
P
32

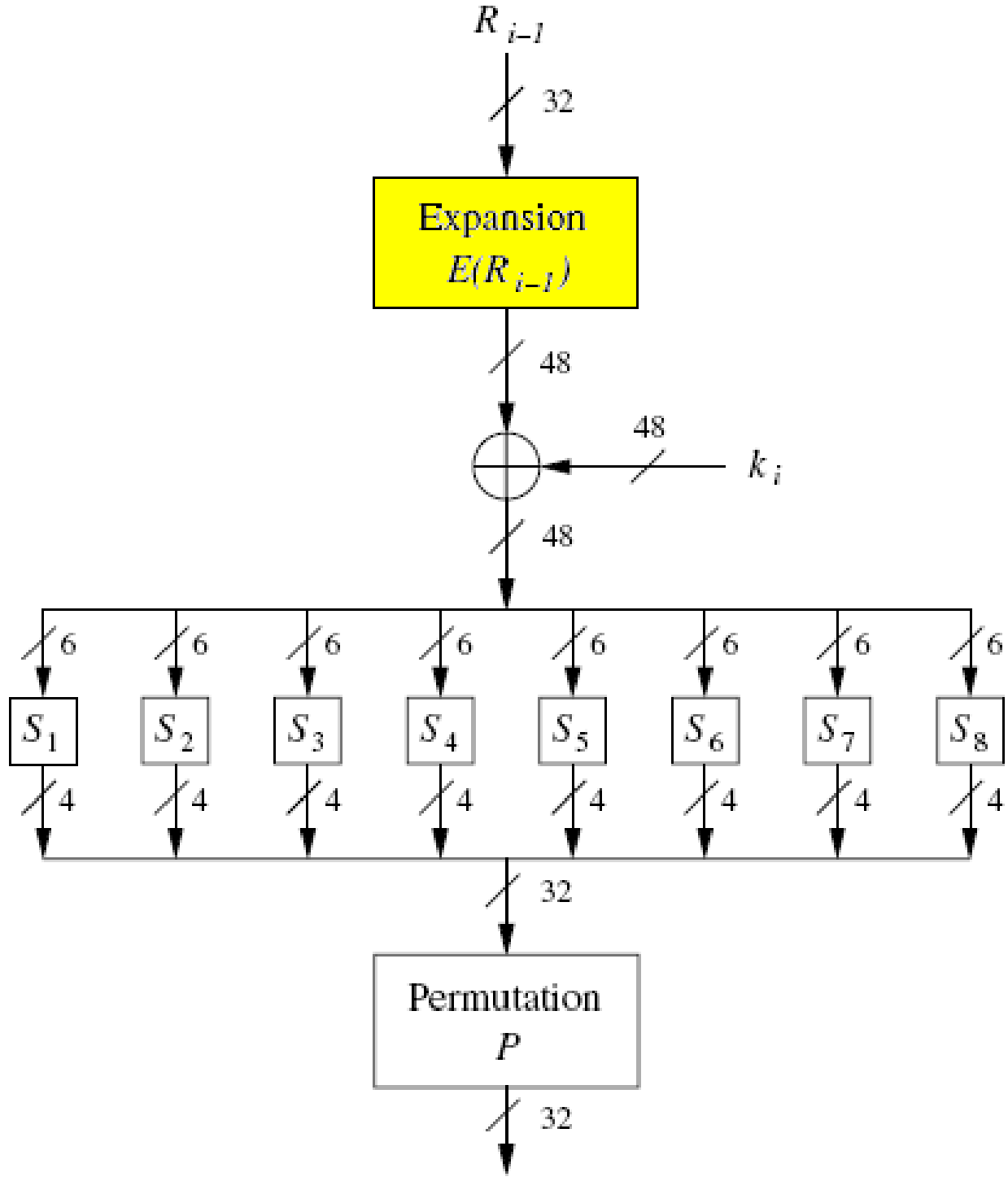
Função de expansão E



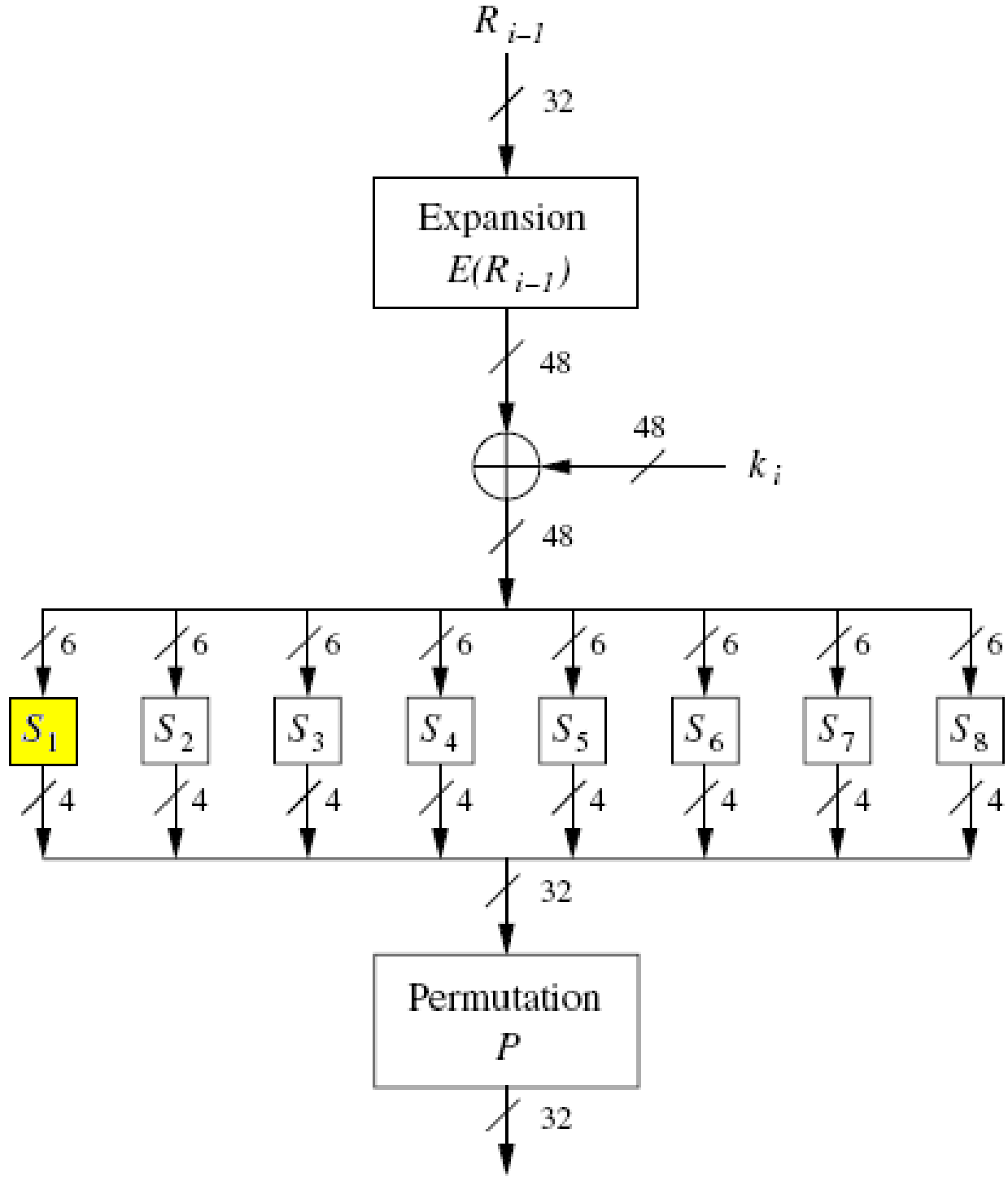
Função de expansão E

E					
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

Função f


R_{i-1}
32
Expansion
$E(R_{i-1})$
48
48
k_i
48
6
6
6
6
6
6
6
6
S_1
S_2
S_3
S_4
S_5
S_6
S_7
S_8
4
4
4
4
4
4
4
4
32
Permutation
P
32

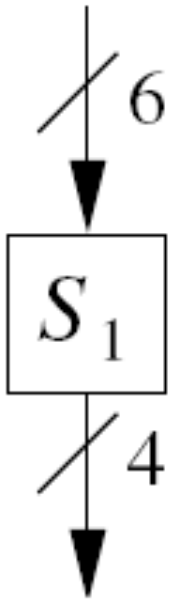
Função f


R_{i-1}
32
Expansion
$E(R_{i-1})$
48
48
k_i
48
6
6
6
6
6
6
6
6
S_1
S_2
S_3
S_4
S_5
S_6
S_7
S_8
4
4
4
4
4
4
4
4
32
Permutation
P
32

Data Encryption Standard (DES)

S-box: S_1

S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13



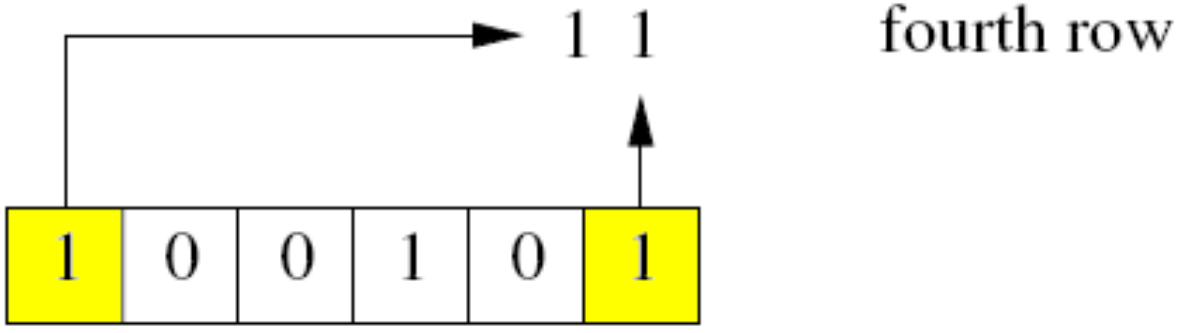
S-box: S_1

S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13

1	0	0	1	0	1

S-box: S_1

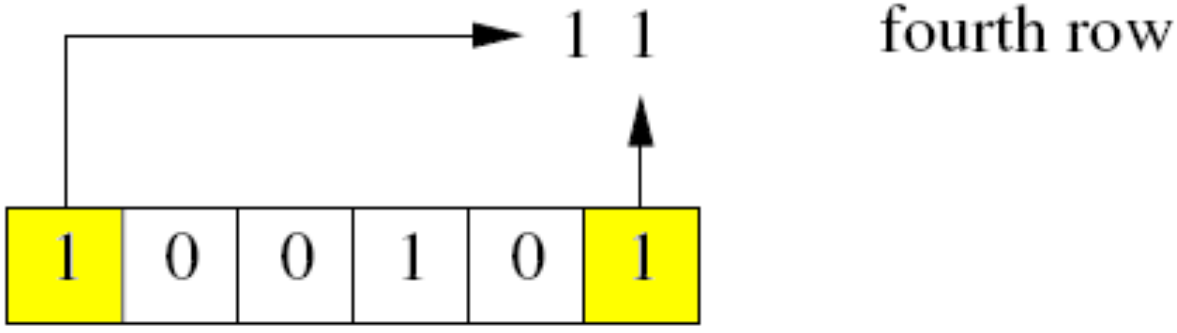
S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13



Data Encryption Standard (DES)

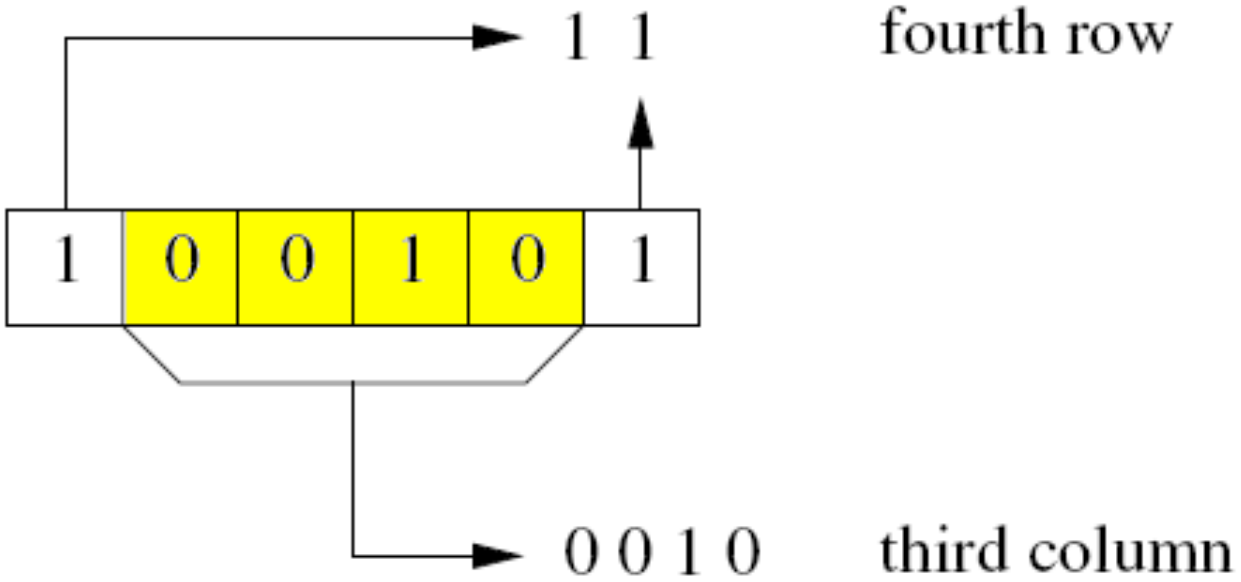
S-box: S_1

S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13



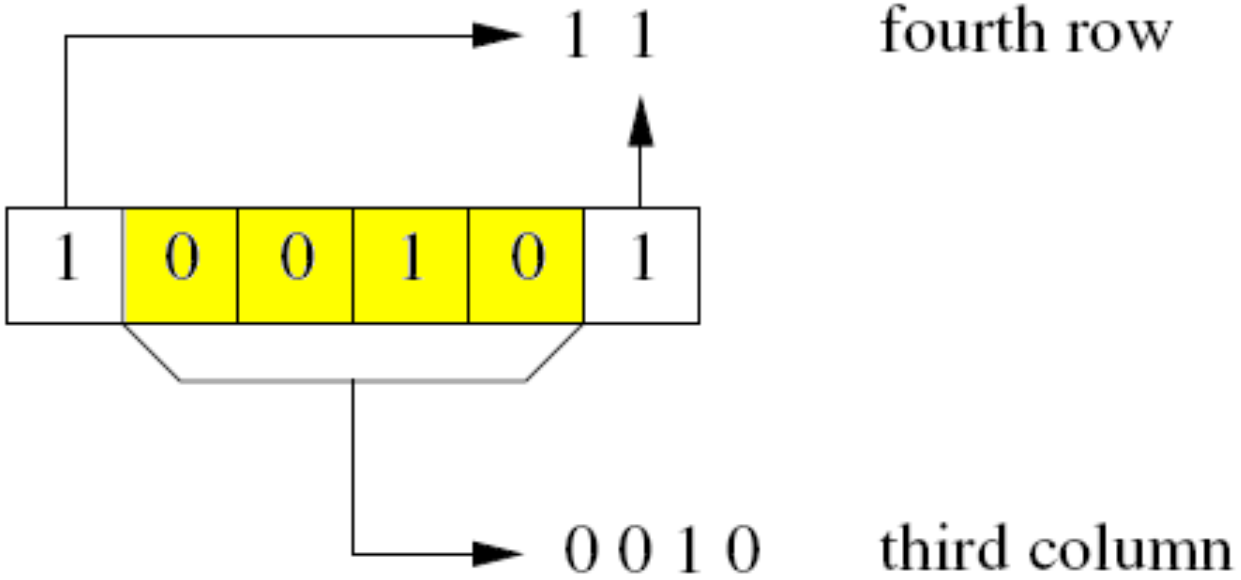
S-box: S_1

S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13



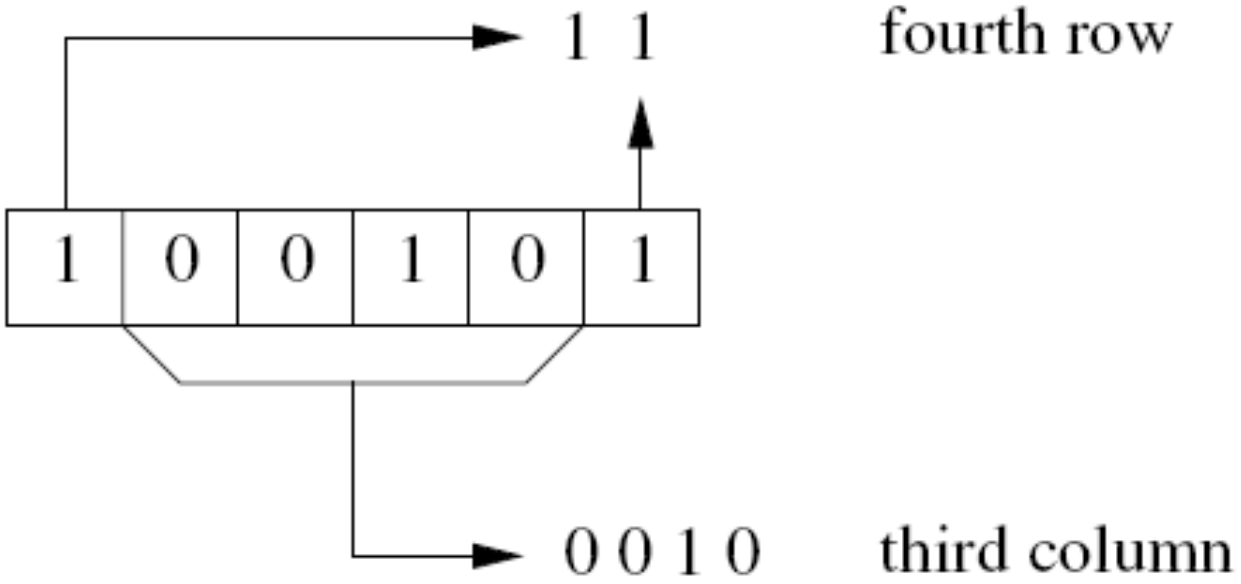
S-box: S_1

S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13

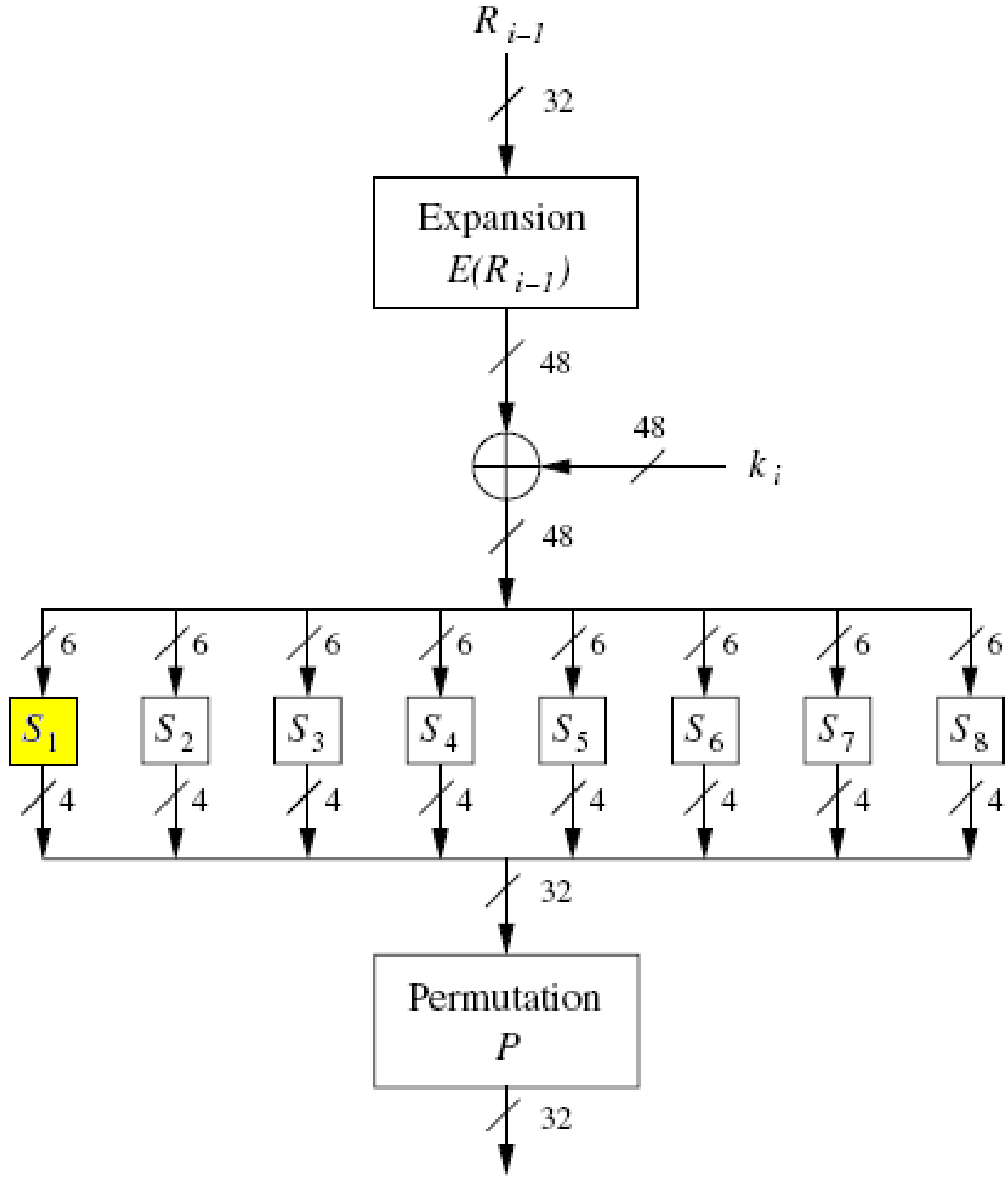


S-box: S_1

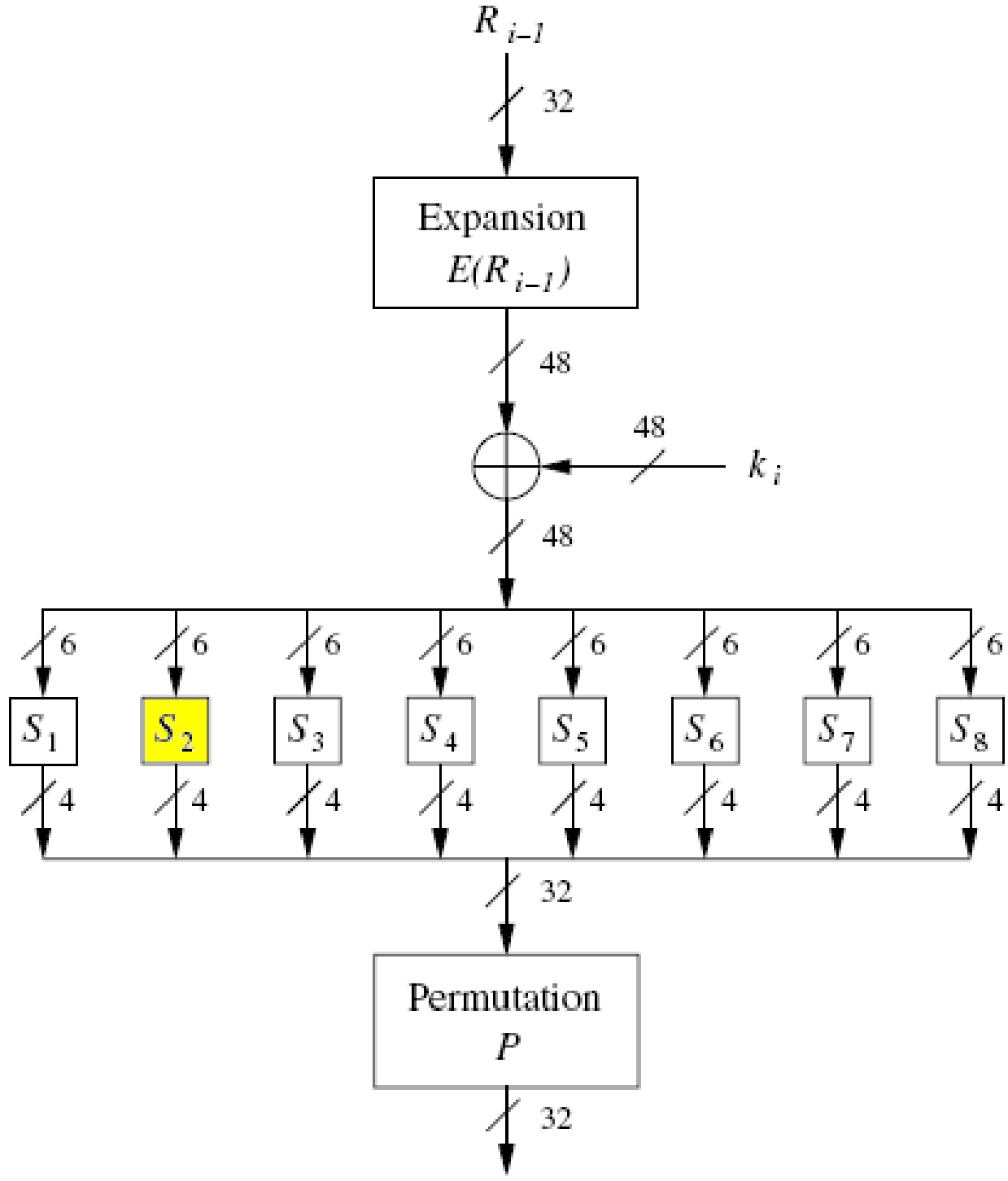
S_1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	04	13	01	02	15	11	08	03	10	06	12	05	09	00	07
1	00	15	07	04	14	02	13	01	10	06	12	11	09	05	03	08
2	04	01	14	08	13	06	02	11	15	12	09	07	03	10	05	00
3	15	12	08	02	04	09	01	07	05	11	03	14	10	00	06	13



Função f


R_{i-1}
32
Expansion
$E(R_{i-1})$
48
48
k_i
48
6
6
6
6
6
6
6
6
S_1
S_2
S_3
S_4
S_5
S_6
S_7
S_8
4
4
4
4
4
4
4
4
32
Permutation
P
32

Função f


R_{i-1}
32
Expansion
$E(R_{i-1})$
48
48
k_i
48
6
6
6
6
6
6
6
6
S_1
S_2
S_3
S_4
S_5
S_6
S_7
S_8
4
4
4
4
4
4
4
4
32
Permutation
P
32

S-box: S_2

S_2	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	15	01	08	14	06	11	03	04	09	07	02	13	12	00	05	10
1	03	13	04	07	15	02	08	14	12	00	01	10	06	09	11	05
2	00	14	07	11	10	04	13	01	05	08	12	06	09	03	02	15
3	13	08	10	01	03	15	04	02	11	06	07	12	00	05	14	09

S-box: S_3

S_3	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	10	00	09	14	06	03	15	05	01	13	12	07	11	04	02	08
1	13	07	00	09	03	04	06	10	02	08	05	14	12	11	15	01
2	13	06	04	09	08	15	03	00	11	01	02	12	05	10	14	07
3	01	10	13	00	06	09	08	07	04	15	14	03	11	05	02	12

S-box: S_4

S_4	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	07	13	14	03	00	06	09	10	01	02	08	05	11	12	04	15
1	13	08	11	05	06	15	00	03	04	07	02	12	01	10	14	09
2	10	06	09	00	12	11	07	13	15	01	03	14	05	02	08	04
3	03	15	00	06	10	01	13	08	09	04	05	11	12	07	02	14

Data Encryption Standard (DES)

S-box: S_5

S_5	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	02	12	04	01	07	10	11	06	08	05	03	15	13	00	14	09
1	14	11	02	12	04	07	13	01	05	00	15	10	03	09	08	06
2	04	02	01	11	10	13	07	08	15	09	12	05	06	03	00	14
3	11	08	12	07	01	14	02	13	06	15	00	09	10	04	05	03

Data Encryption Standard (DES)

S-box: S_6

S_6	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	12	01	10	15	09	02	06	08	00	13	03	04	14	07	05	11
1	10	15	04	02	07	12	09	05	06	01	13	14	00	11	03	08
2	09	14	15	05	02	08	12	03	07	00	04	10	01	13	11	06
3	04	03	02	12	09	05	15	10	11	14	01	07	06	00	08	13

Data Encryption Standard (DES)

S-box: S_7

S_7	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	04	11	02	14	15	00	08	13	03	12	09	07	05	10	06	01
1	13	00	11	07	04	09	01	10	14	03	05	12	02	15	08	06
2	01	04	11	13	12	03	07	14	10	15	06	08	00	05	09	02
3	06	11	13	08	01	04	10	07	09	05	00	15	14	02	03	12

S-box: S_8

S_8	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	13	02	08	04	06	15	11	01	10	09	03	14	05	00	12	07
1	01	15	13	08	10	03	07	04	12	05	06	11	00	14	09	02
2	07	11	04	01	09	12	14	02	00	06	10	13	15	03	05	08
3	02	01	14	07	04	10	08	13	15	12	09	00	03	05	06	11

Data Encryption Standard (DES)

As S-boxes são o coração do DES em termos de segurança. São as únicas operações não lineares do algoritmo

$$S(a) \oplus S(b) \neq S(a \oplus b)$$

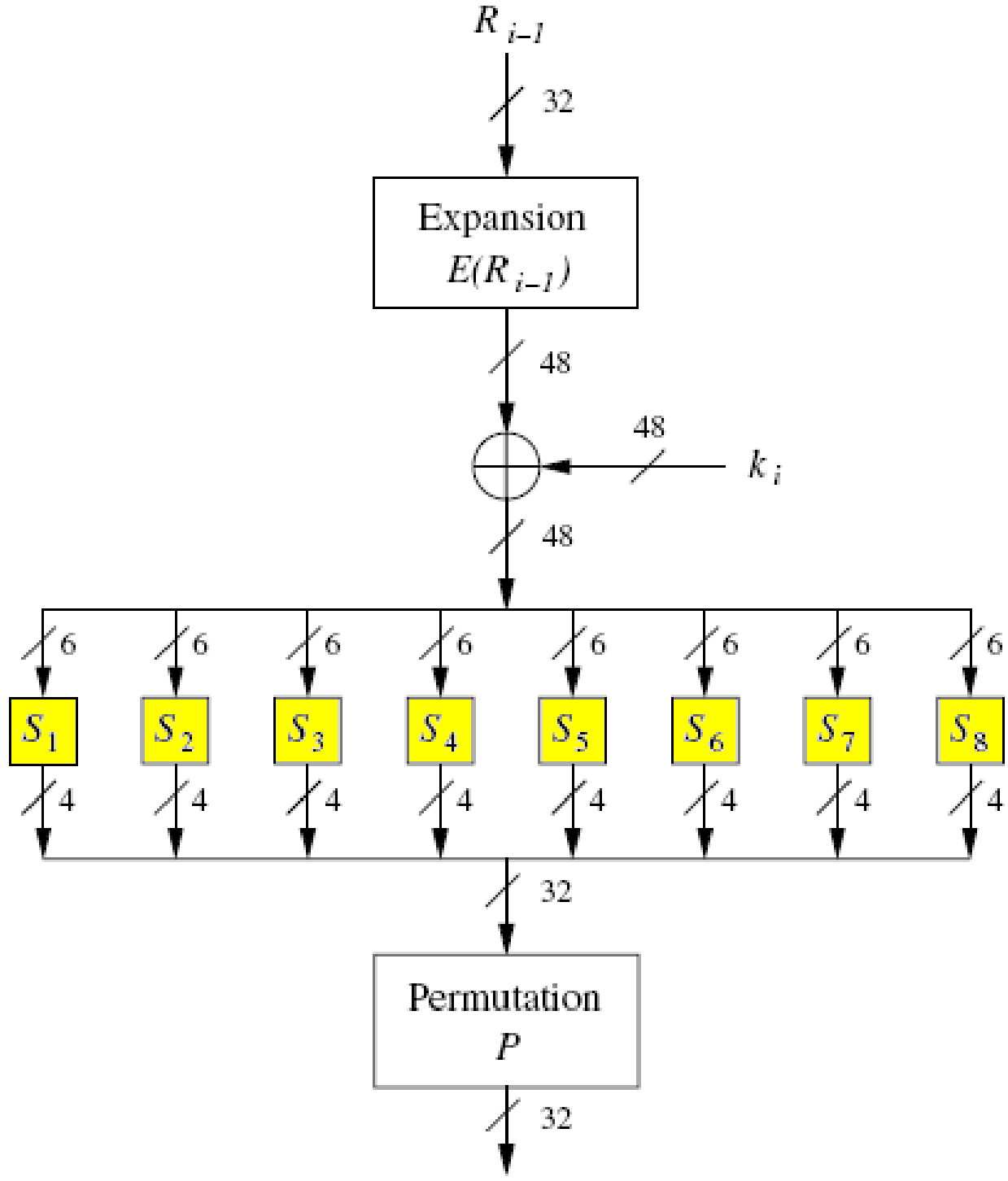
São elementos para prover confusão na cifra. A escolha dos valores dessas caixas só foram revelados em 1990 (13 anos após o padrão ter sido estabelecido).

Data Encryption Standard (DES)

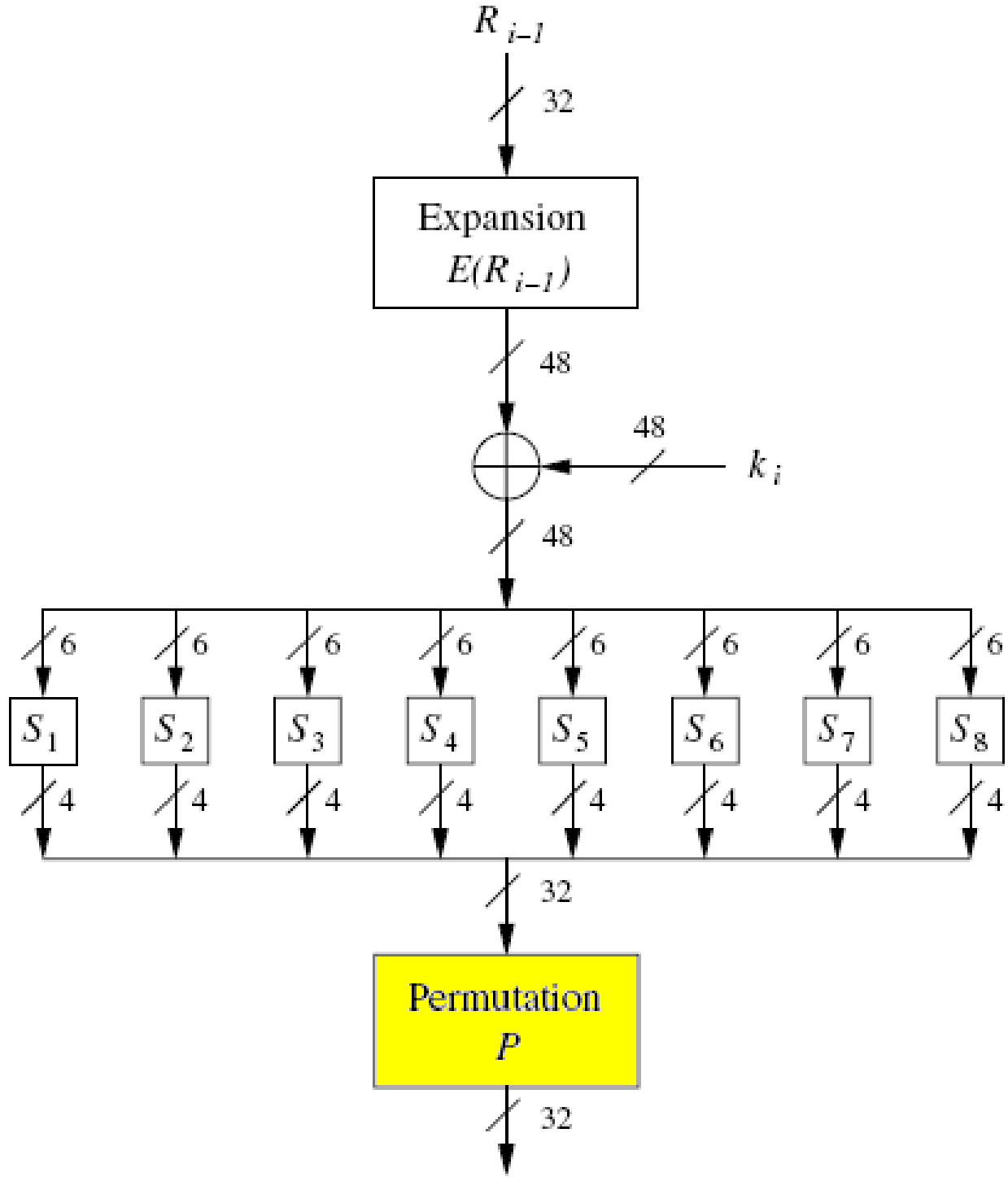
Regras para os valores das S-boxes (8 no total) dentre as quais:

- Cada S-box deve ter 6-bits de entrada e 4 de saída.
- Nenhum bit isolado deve ser próximo de uma combinação linear dos bits de entrada.
- Se duas entradas diferem em exatamente 1 bit, as saídas devem diferir em pelo menos 2 bits.
- Se duas entradas diferem em 2 bits no centro, a saída deve diferir em pelo menos 2 bits.

Função f


R_{i-1}
32
Expansion
$E(R_{i-1})$
48
48
k_i
48
6
6
6
6
6
6
6
6
S_1
S_2
S_3
S_4
S_5
S_6
S_7
S_8
4
4
4
4
4
4
4
4
32
Permutation
P
32

Função f


R_{i-1}
32
Expansion
$E(R_{i-1})$
48
48
k_i
48
6
6
6
6
6
6
6
6
S_1
S_2
S_3
S_4
S_5
S_6
S_7
S_8
4
4
4
4
4
4
4
4
32
Permutation
P
32

Permutação (P)

P							
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

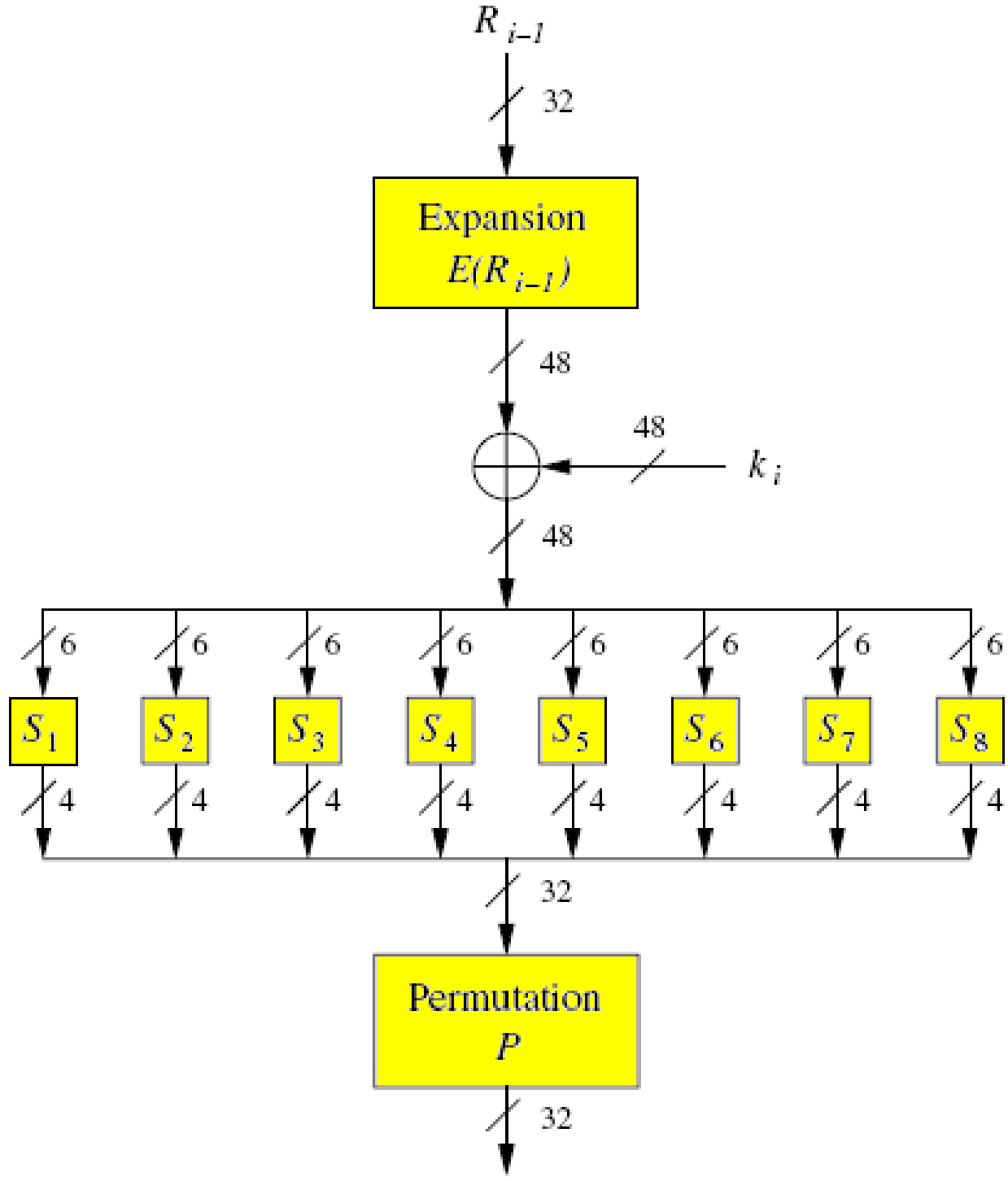
Data Encryption Standard (DES)

Efeito avalanche (confusão e difusão): a função de permutação P provê difusão pois os 4 bits de saída de cada S-box são permutados de tal forma, que afetam diferentes S-boxes nas próximas rodadas. Em resumo, o conjunto de funções Expansão, S-boxes e permutação P garantem que todo bit apartir da quinta rodada está em função de todo o texto em claro (64 bits) e chave (56 bits).

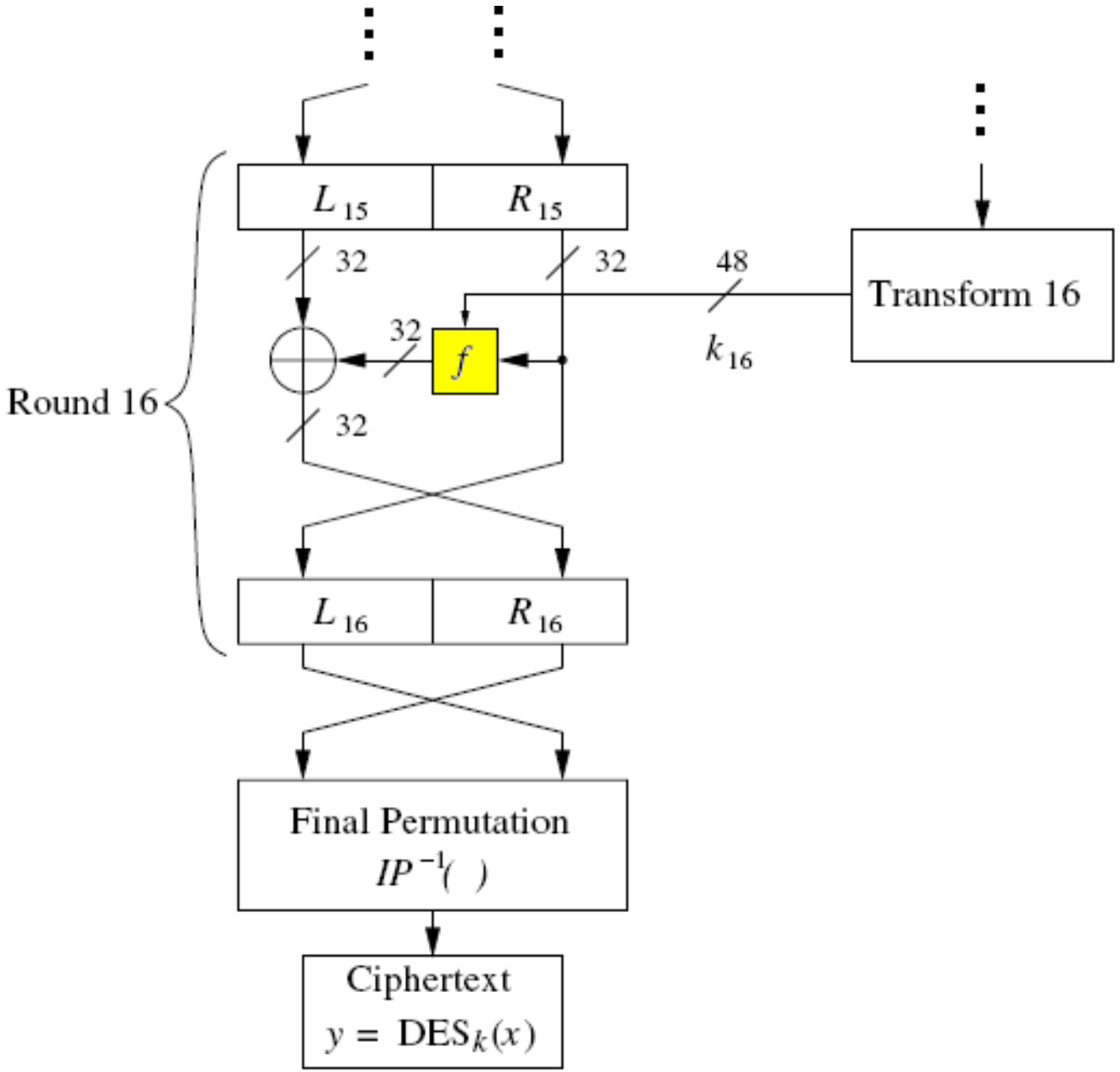
Data Encryption Standard (DES)

Curiosidade: as funções S-box foram cuidadosamente projetadas para resistir ao ataque matemático conhecido como criptoanálise diferencial. A questão é que esse ataque só foi descoberto pela comunidade científica em 1990 (16 anos após o DES ter se tornado padrão de cifragem). A IBM declarou que esse ataque já tinha sido descoberto por eles na época.

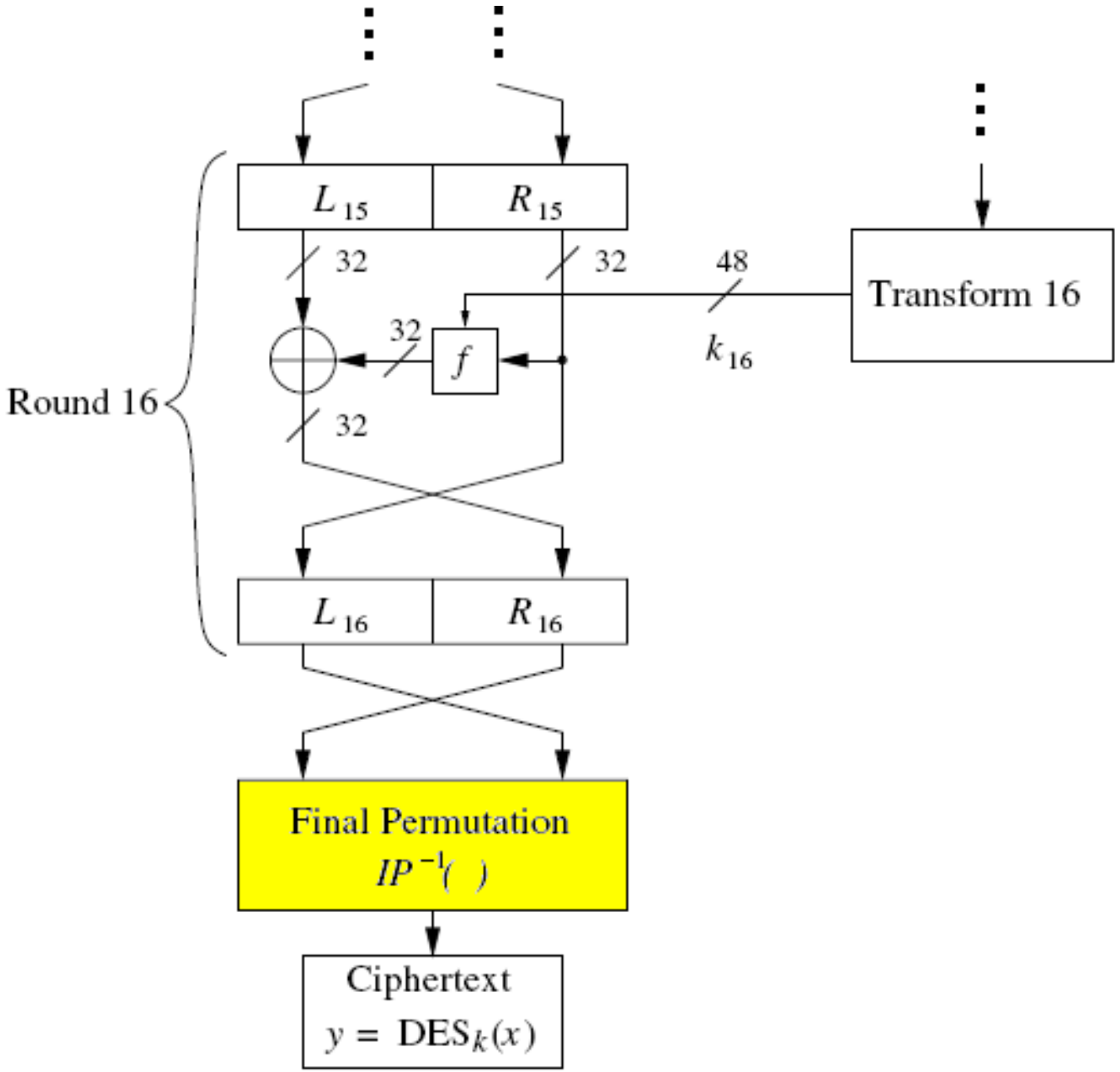
Função f


R_{i-1}
32
Expansion
$E(R_{i-1})$
48
48
k_i
48
6
6
6
6
6
6
6
6
S_1
S_2
S_3
S_4
S_5
S_6
S_7
S_8
4
4
4
4
4
4
4
4
32
Permutation
P
32

Data Encryption Standard (DES)

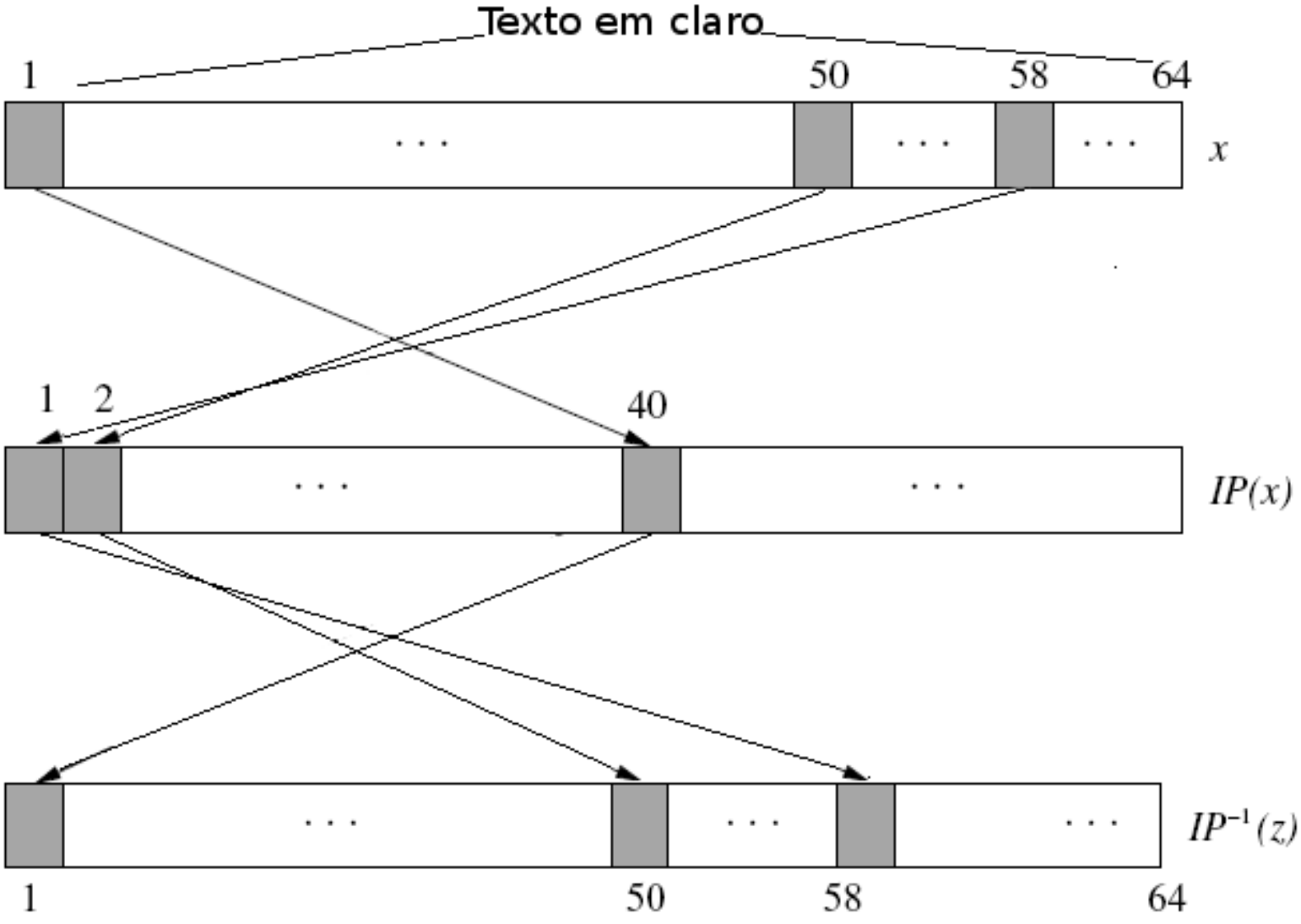


Data Encryption Standard (DES)



Data Encryption Standard (DES)

Permutação final (IP^{-1})



Permutação final (IP^{-1})

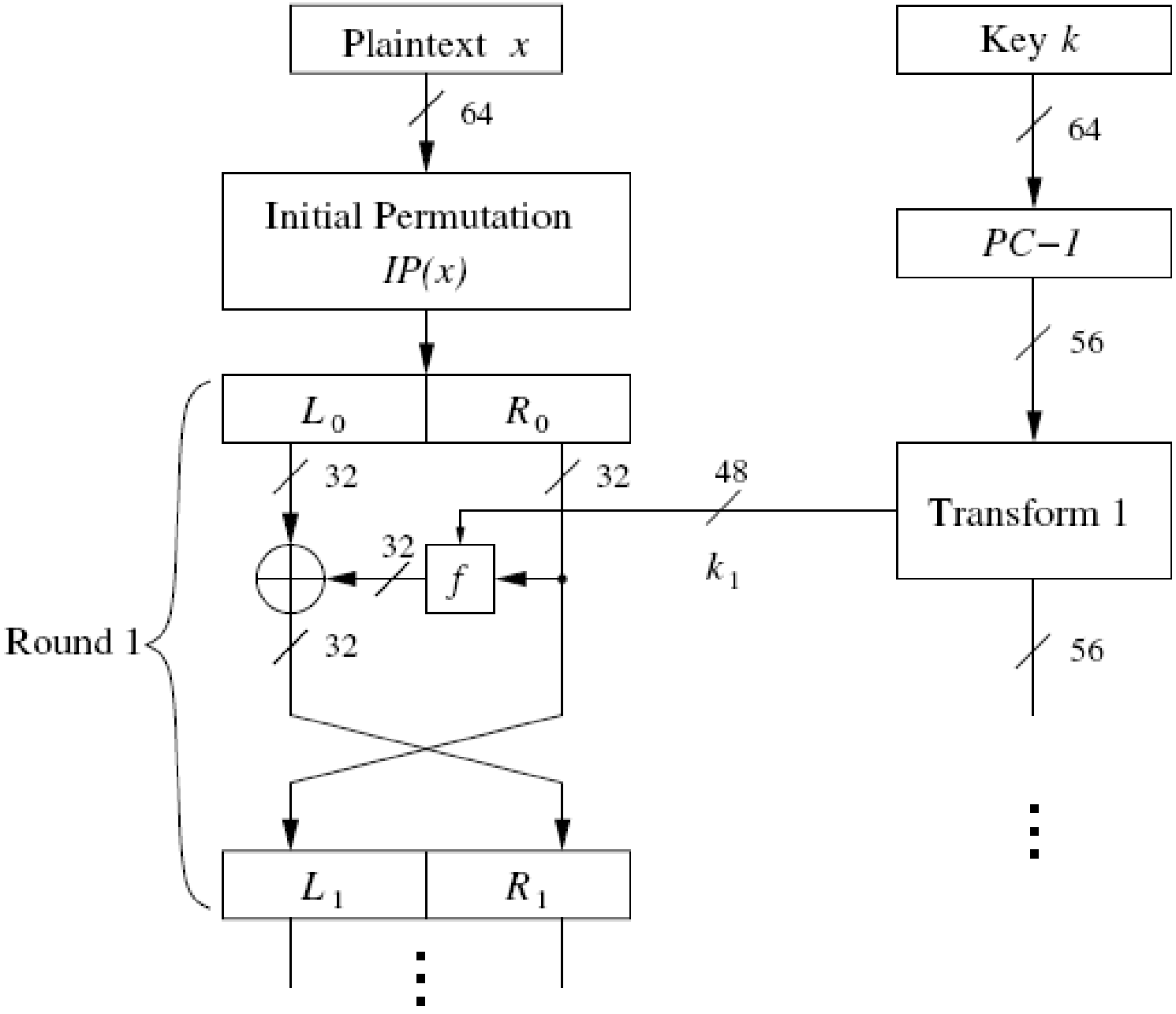
IP^{-1}							
40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

Sumário

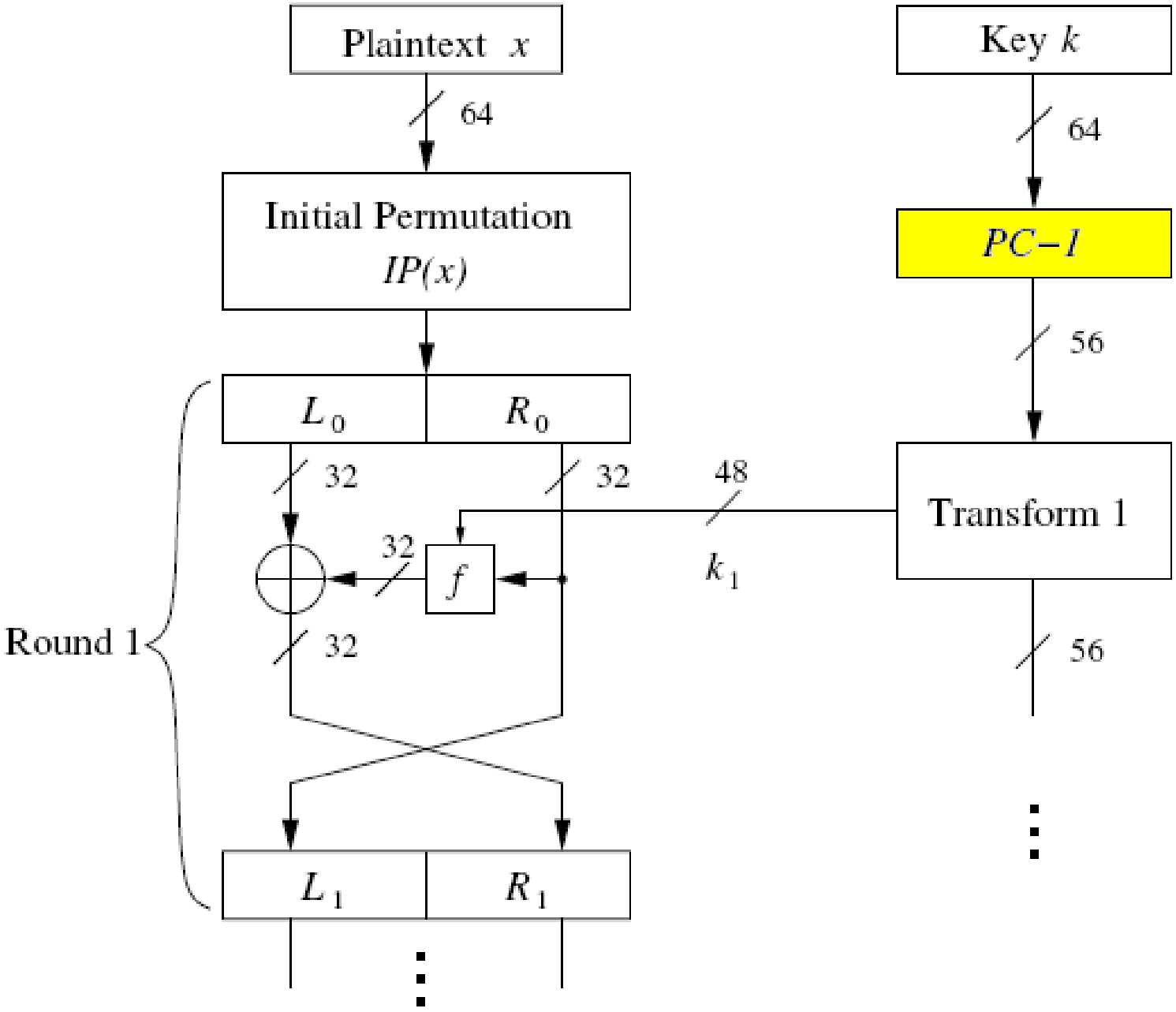
Escalonamento de Chaves - DES

Como calcular as 16 chaves $k_1, k_2, \ldots, k_{16}$ necessárias a cada uma das rodadas que o DES realiza?

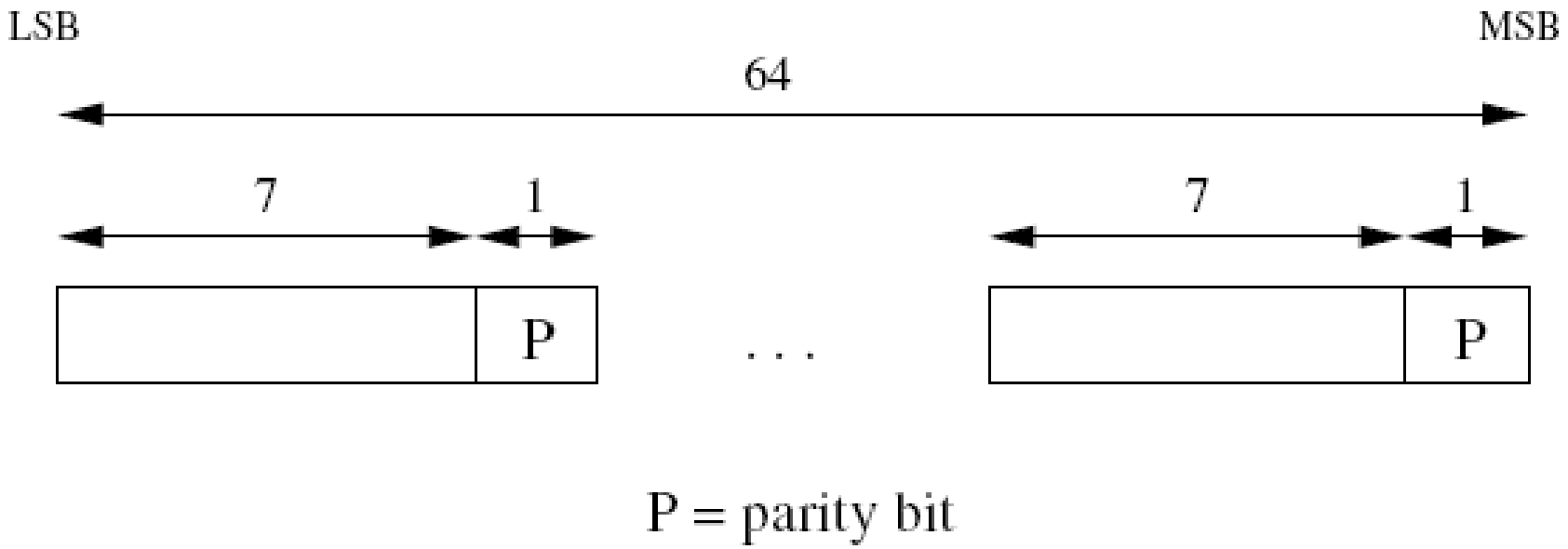
Data Encryption Standard (DES)



Data Encryption Standard (DES)

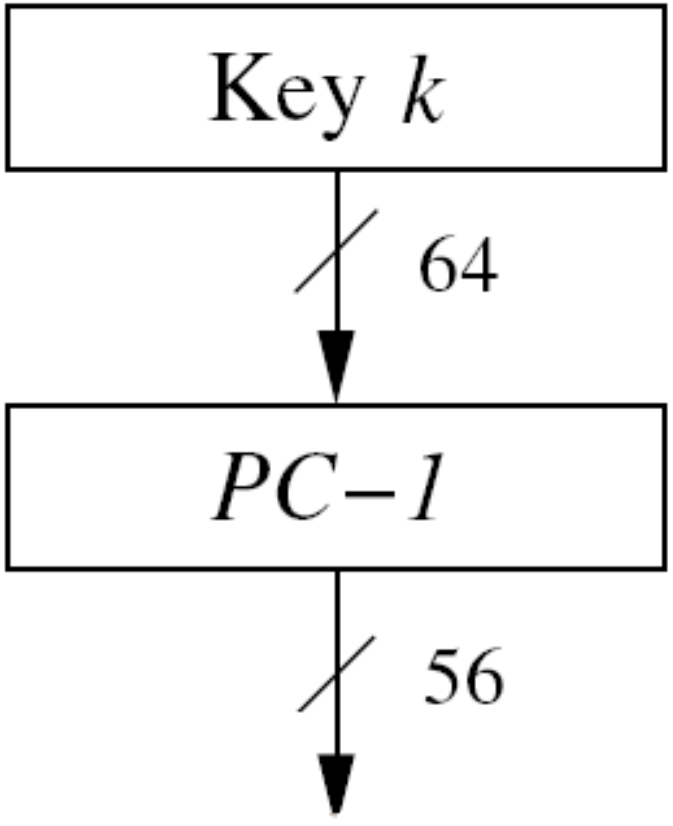


Permutação inicial da chave PC - 1

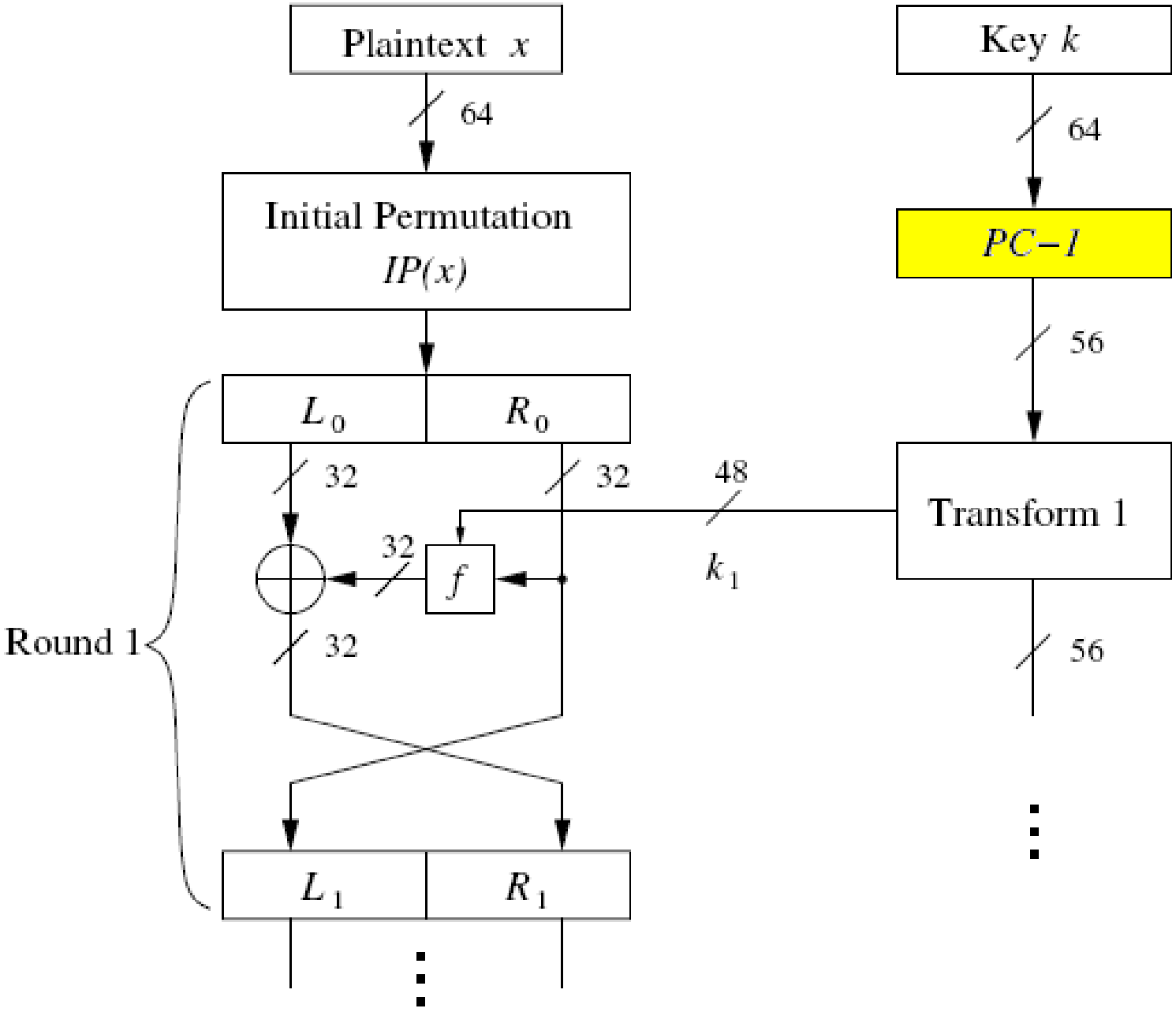


Permutação inicial da chave PC - 1

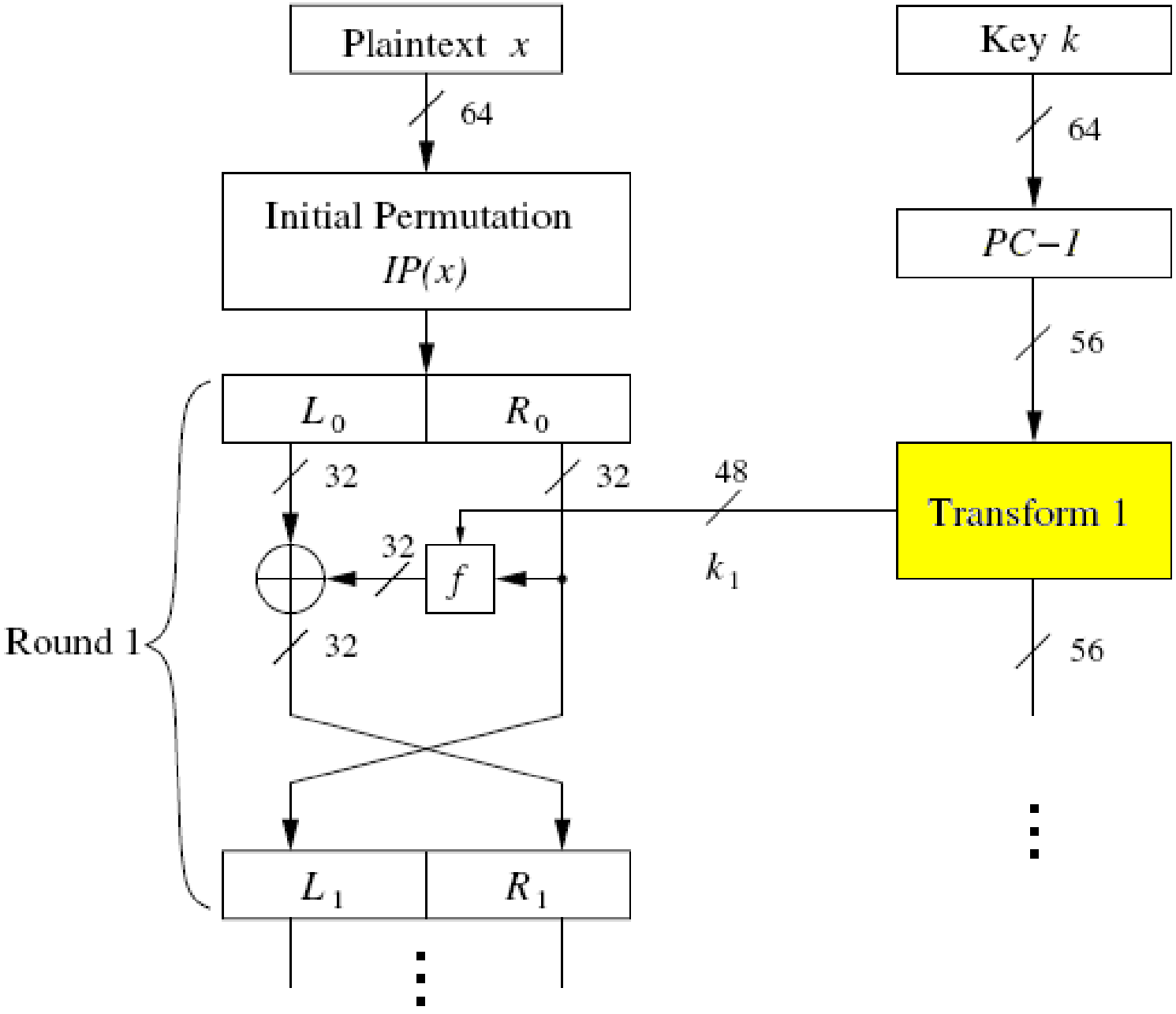
$PC-1$							
57	49	41	33	25	17	9	1
58	50	42	34	26	18	10	2
59	51	43	35	27	19	11	3
60	52	44	36	63	55	47	39
31	23	15	7	62	54	46	38
30	22	14	6	61	53	45	37
29	21	13	5	28	20	12	4



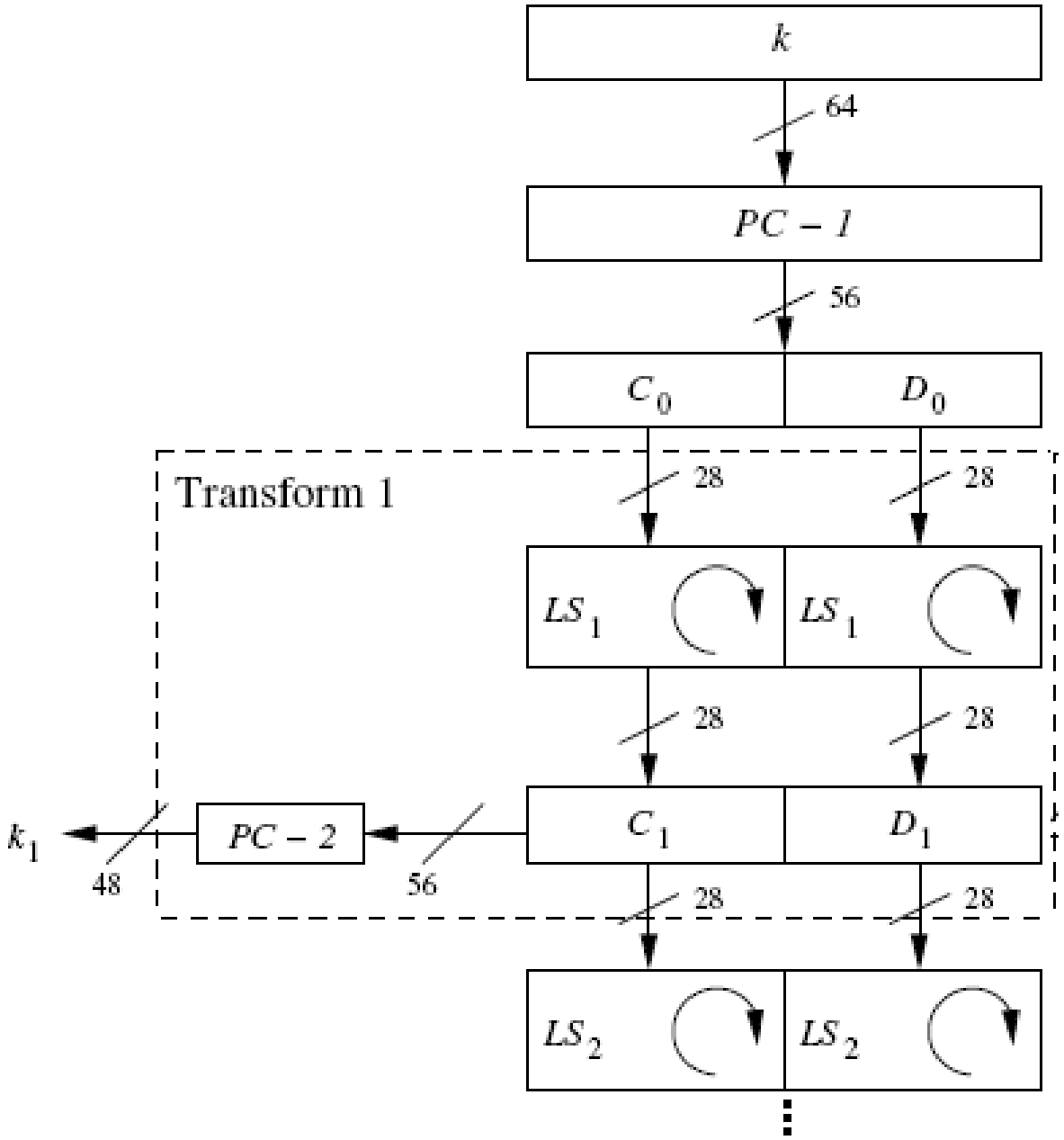
Data Encryption Standard (DES)



Data Encryption Standard (DES)



Data Encryption Standard (DES)



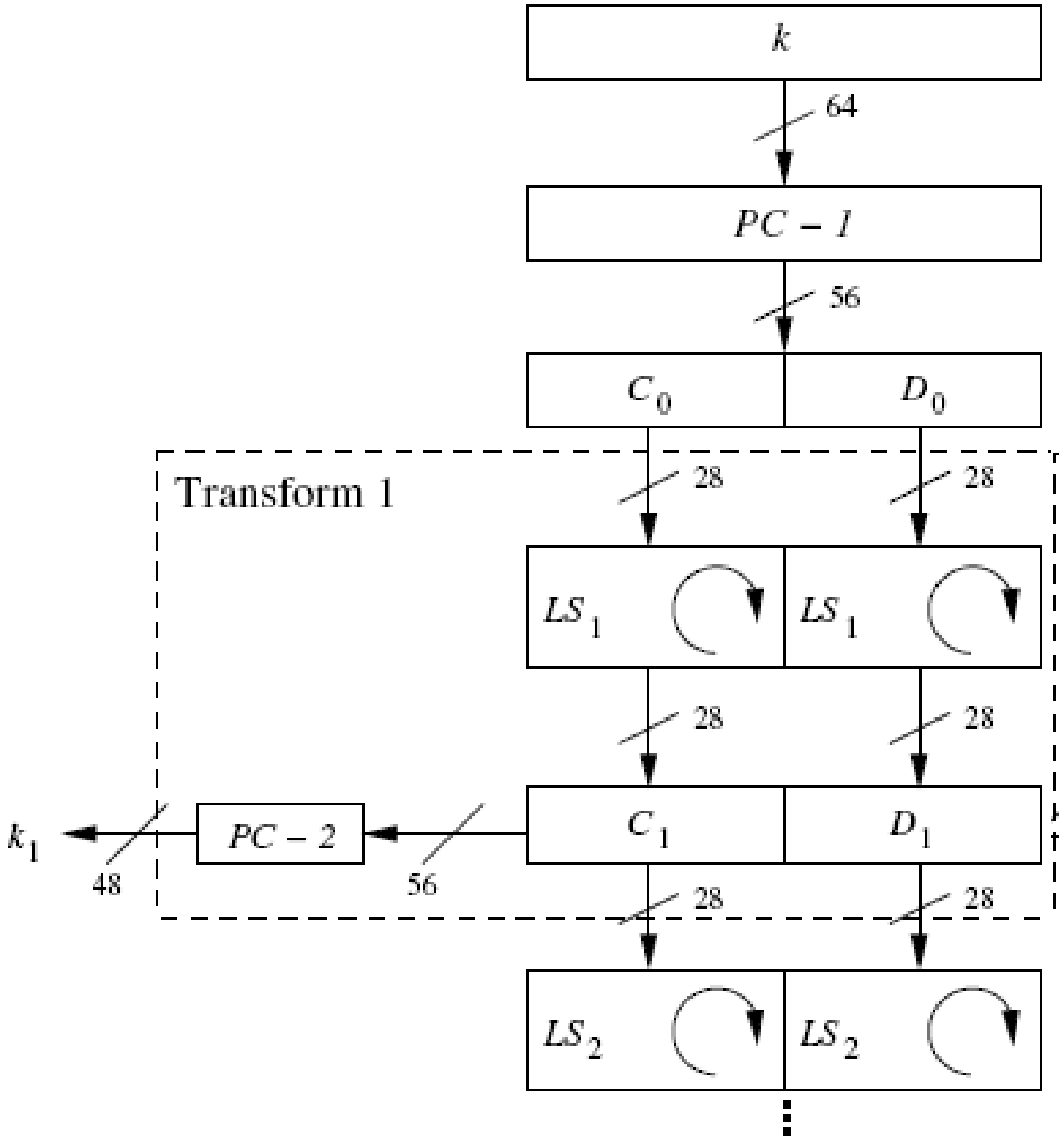
Data Encryption Standard (DES)

Divida a chave em duas metades. Seja C os primeiros 28 bits e D os últimos 28 bits. Para $i = 1, \ldots, 16$ realize um deslocamento circular para a esquerda, tanto em C quanto em D, de acordo com a seguinte regra:

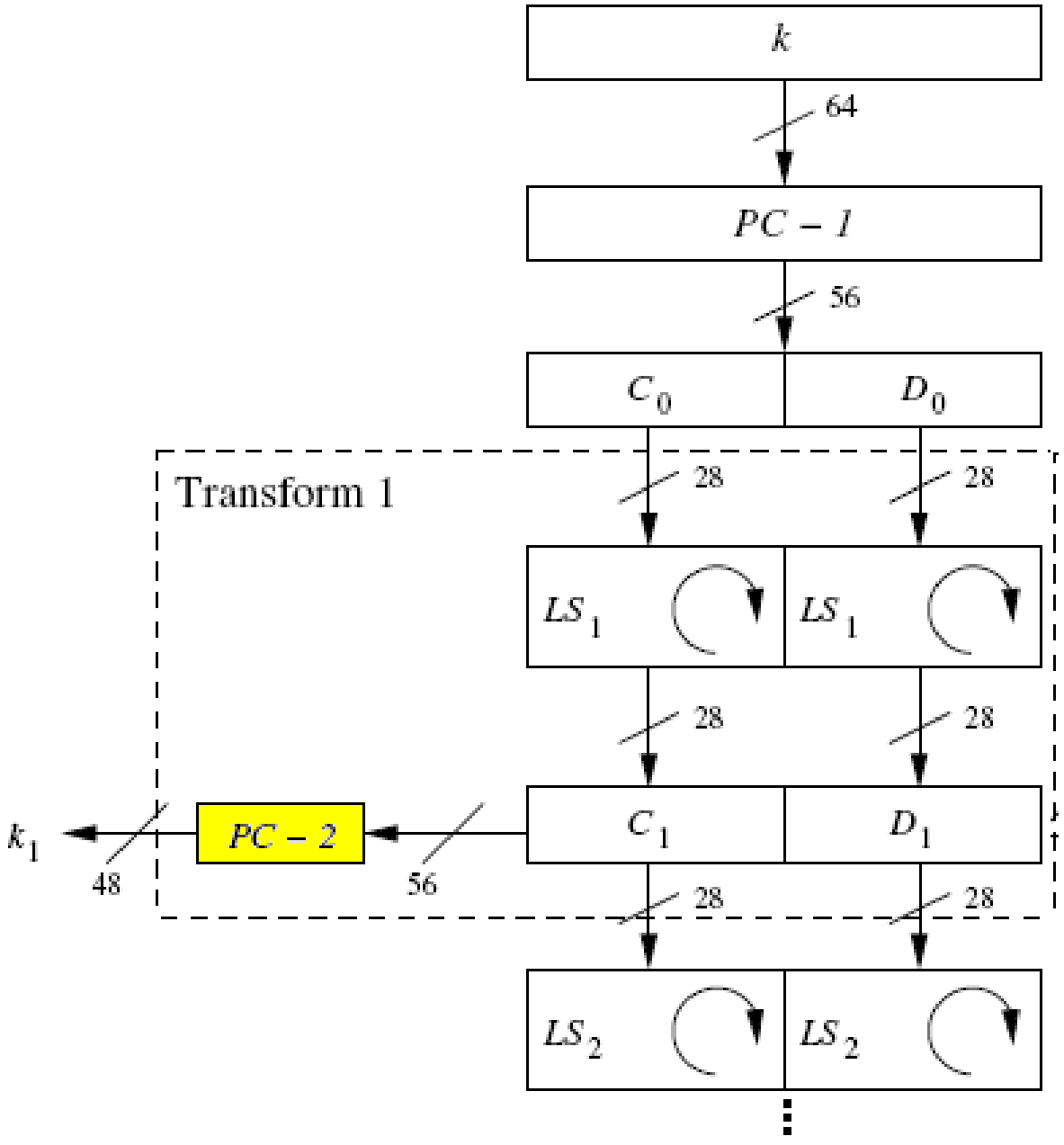
i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
shift	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Note que 4 * 1 + 12 * 2 = 28, assim temos $C_0 = C_{16}$ e que $D_0 = D_{16}$ (para decrifar é só reverter o processo).

Data Encryption Standard (DES)

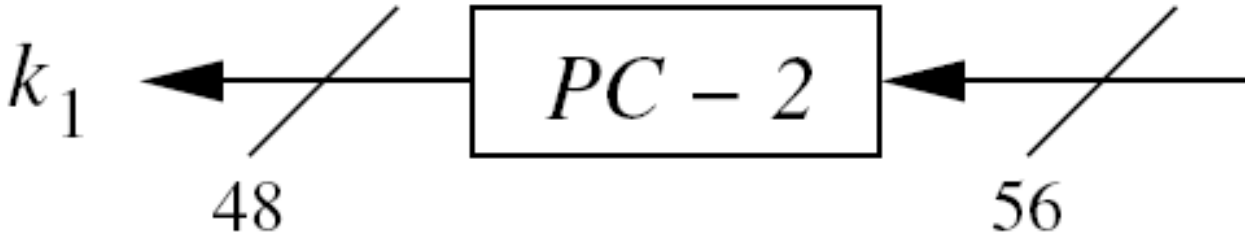


Data Encryption Standard (DES)



Permutação da chave PC-2

$PC-2$							
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



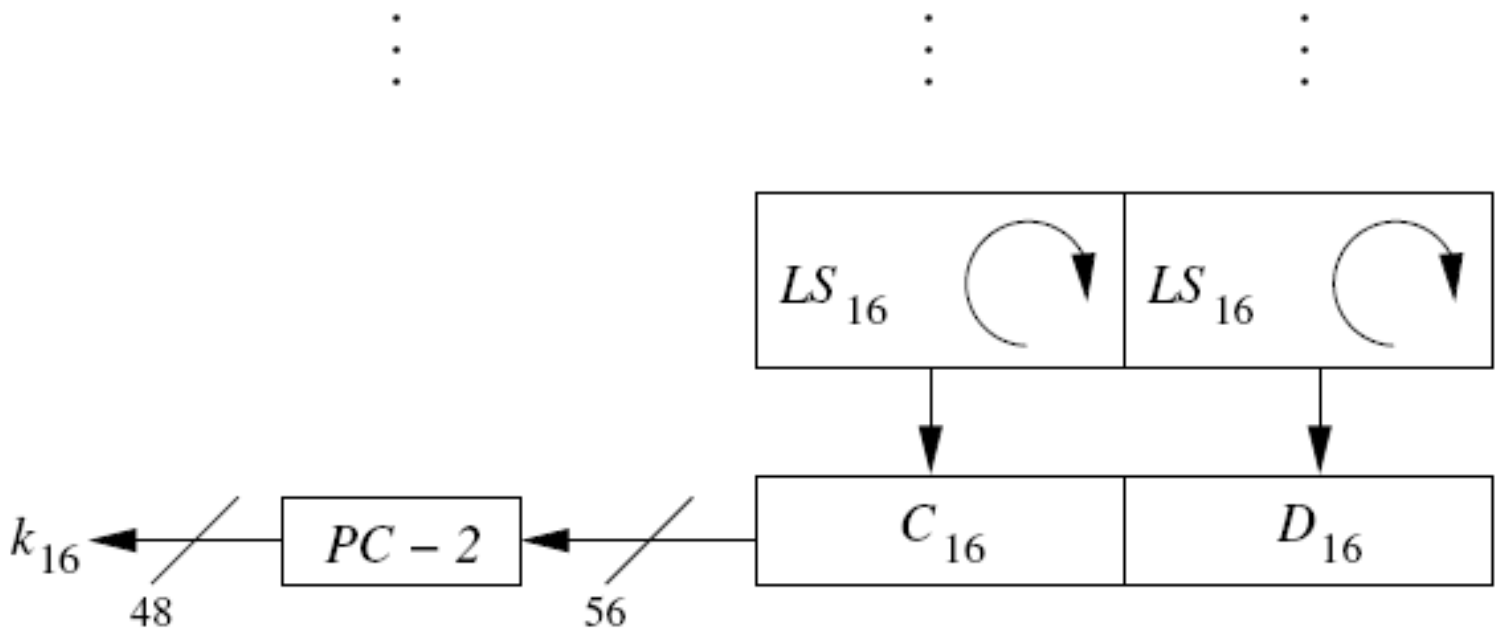
Sumário

Data Encryption Standard (DES)

Decifrando: a vantagem do DES é que a função de decifrar é essencialmente a mesma função de cifrar (rede de Feistel). A única diferença é que o escalonador de chaves é realizado de forma inversa. Assim, para decifrar o round 1, a sub-chave 16 é necessária, para o round 2 a sub-chave 15, ... Assim, a ordem das chaves para decifragem é $k_{16}, k_{15}, \ldots, k_1$.

Data Encryption Standard (DES)

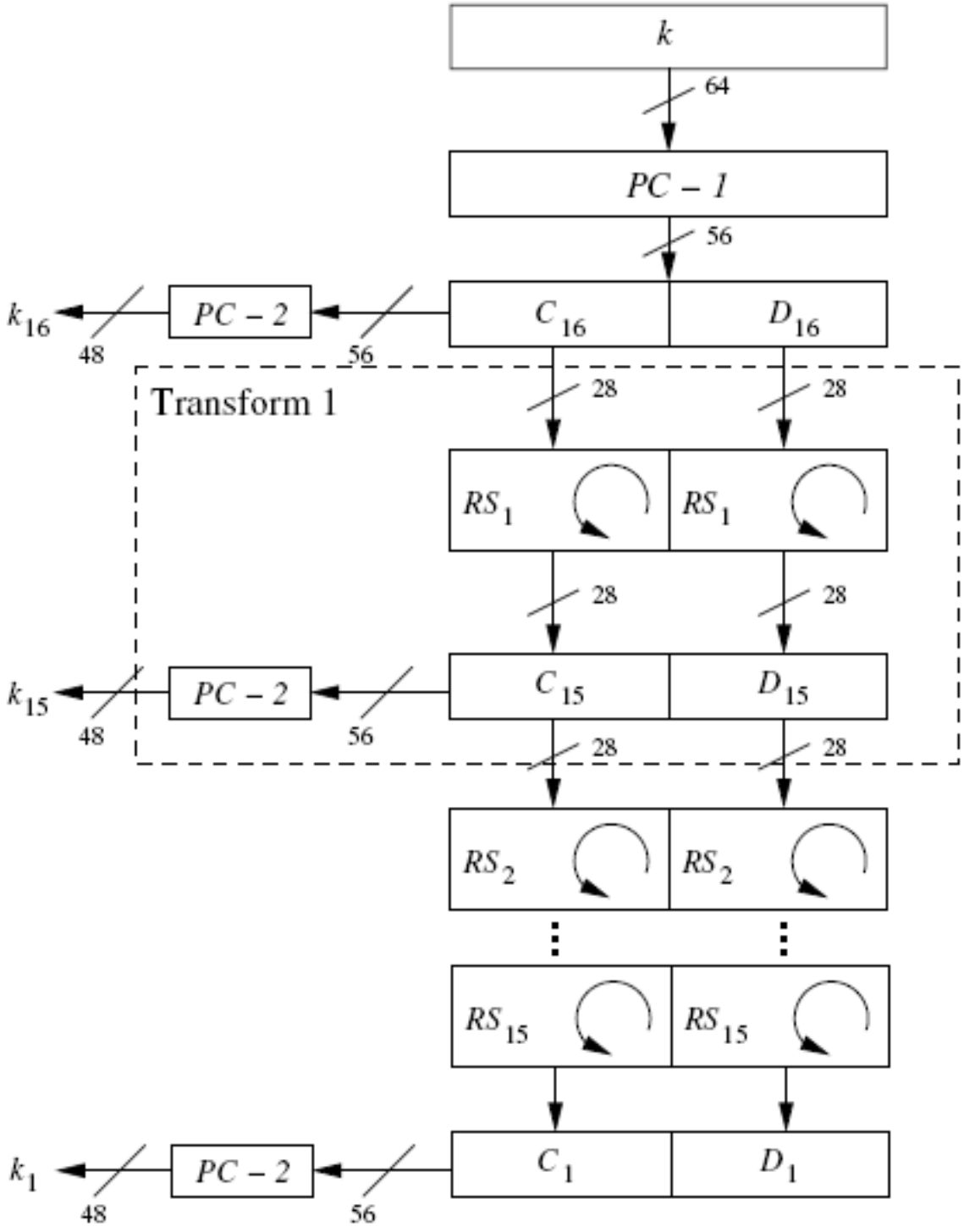
Questão: dado a chave simétrica inicial K (de posse do usuário) como produzir a sub-chave k_{16} para a decifragem?



Data Encryption Standard (DES)

Lembre-se que $C_0 = C_{16}$ e que $D_0 = D_{16}$, por isso k_{16} pode ser diretamente derivada de $PC - 1$

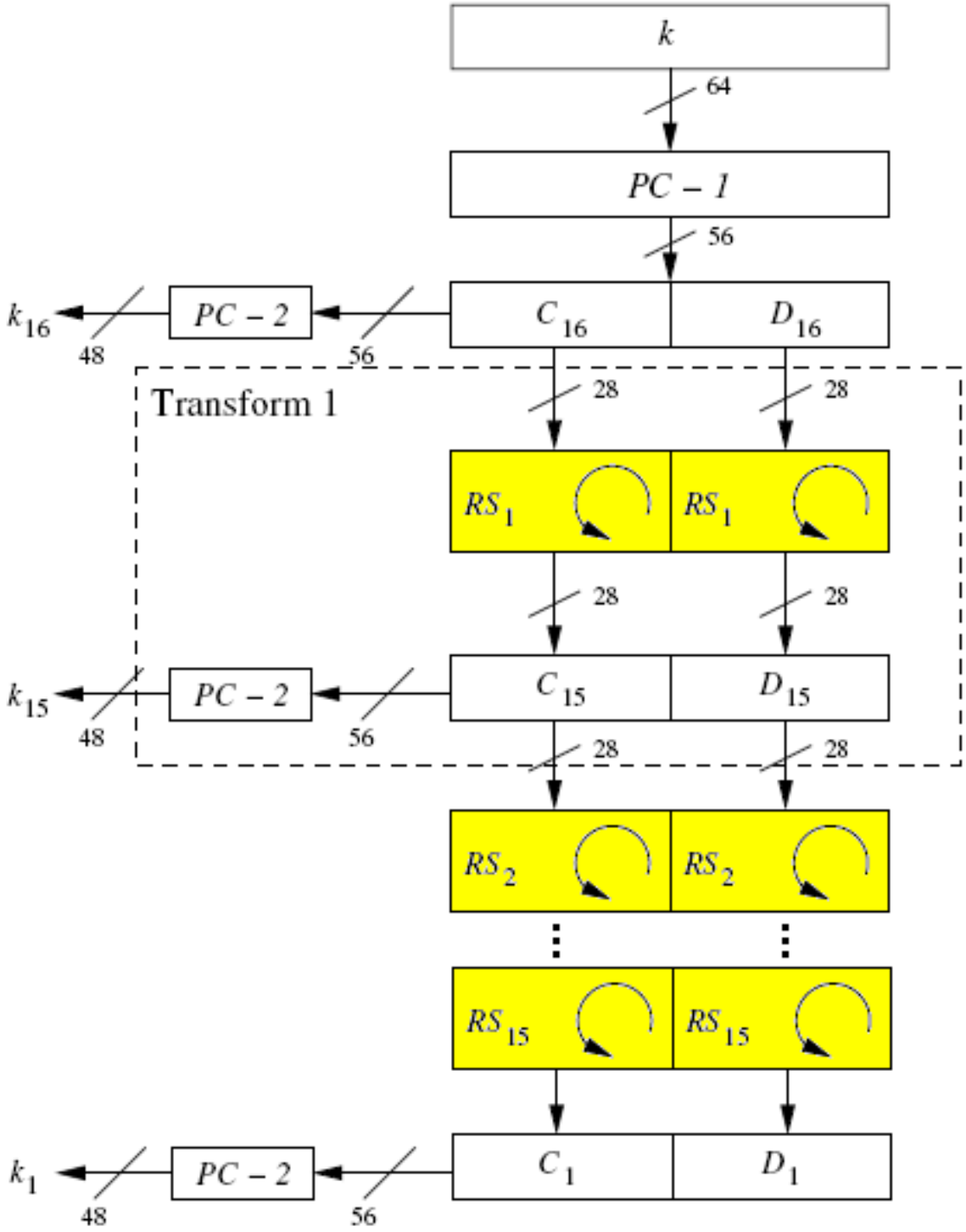
$$\begin{aligned} k_{16} &= PC - 2(C_{16}, D_{16}) \\ &= PC - 2(C_0, D_0) \\ &= PC - 2(PC - 1(K)) \end{aligned}$$

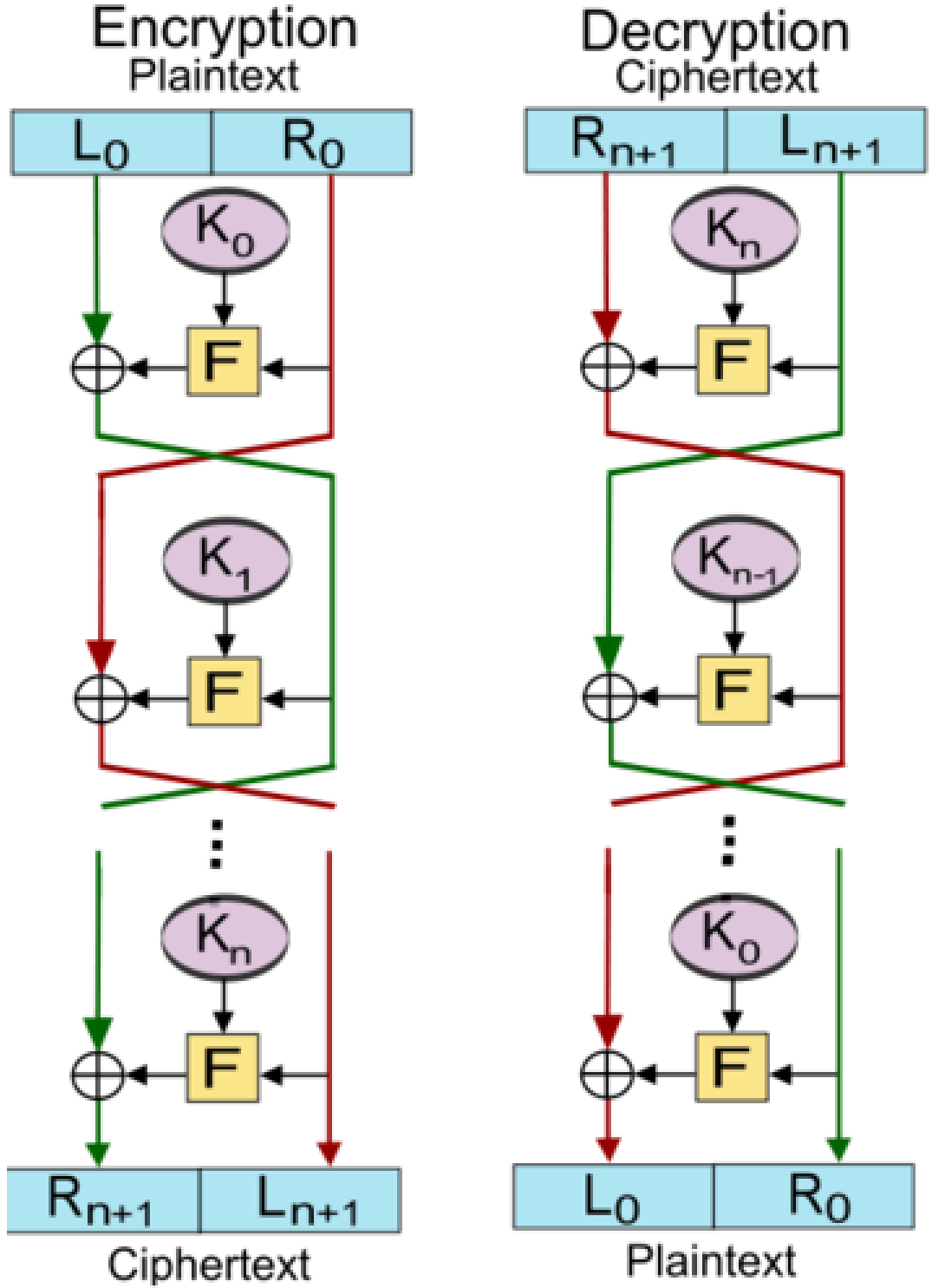

k
64
PC – 1
56
k_{16}
PC – 2
48
56
C_{16}
D_{16}
Transform 1
28
28
RS_1
RS_1
28
28
k_{15}
PC – 2
48
56
C_{15}
D_{15}
28
28
RS_2
RS_2
RS_{15}
RS_{15}
k_1
PC – 2
48
56
C_1
D_1

Data Encryption Standard (DES)

As chaves restantes são deslocadas para a direita (operação $RS_{1,2,\ldots,16}$) de 1 ou 2 bits, revertendo o processo da cifragem.

i	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
shift		1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

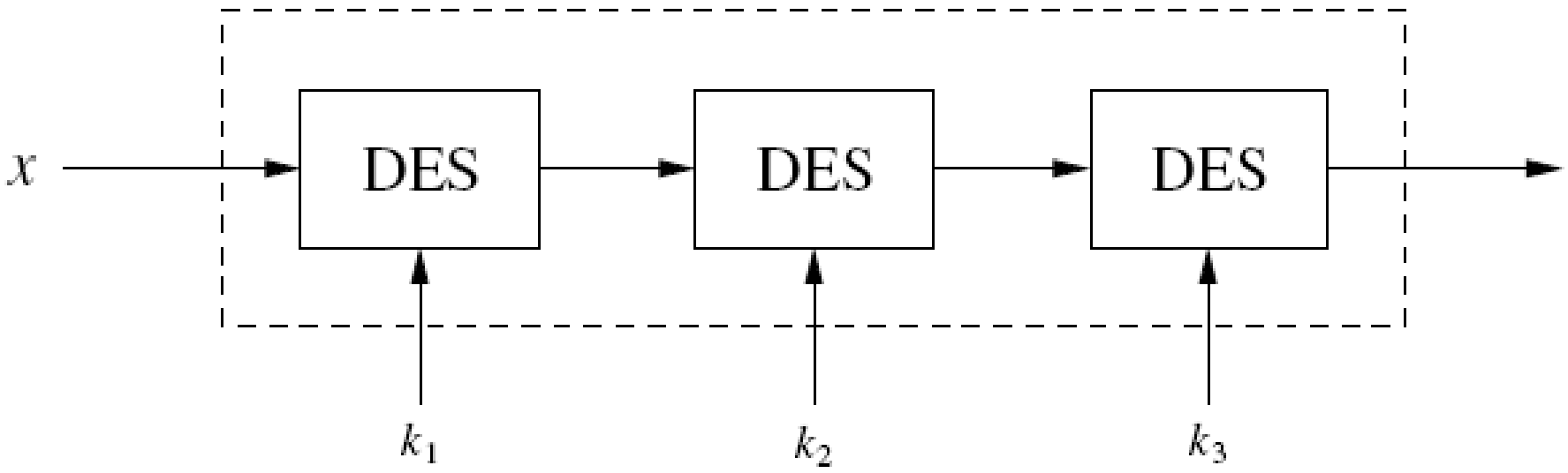

k
64
PC – 1
56
C 16
D 16
PC – 2
k 16
48
56
Transform 1
28
28
RS 1
RS 1
28
28
C 15
D 15
PC – 2
k 15
48
56
28
28
RS 2
RS 2
RS 15
RS 15
C 1
D 1
PC – 2
k 1
48
56


Encryption
Plaintext
L_0
R_0
K_0
F
K_1
F
K_n
F
R_{n+1}
L_{n+1}
Ciphertext
Decryption
Ciphertext
R_{n+1}
L_{n+1}
K_n
F
K_{n-1}
F
K_0
F
L_0
R_0
Plaintext

Data Encryption Standard (DES)

3-DES (resistente a ataques por força-bruta)

$$y = DES_{k_3}(DES_{k_2}(DES_{k_1}(x)))$$



Introdução

- O DES foi substituído pelo AES em 2000.
- O DES atualmente é inseguro devido a chave pequena de 56 bits.
- O 3-DES é considerado muito seguro e ainda amplamente utilizado nos dias de hoje.
- O DES cifra blocos de tamanho de 64 bits.
- É o algoritmo mais popular e estudado para criptografia simétrica.

Sumário

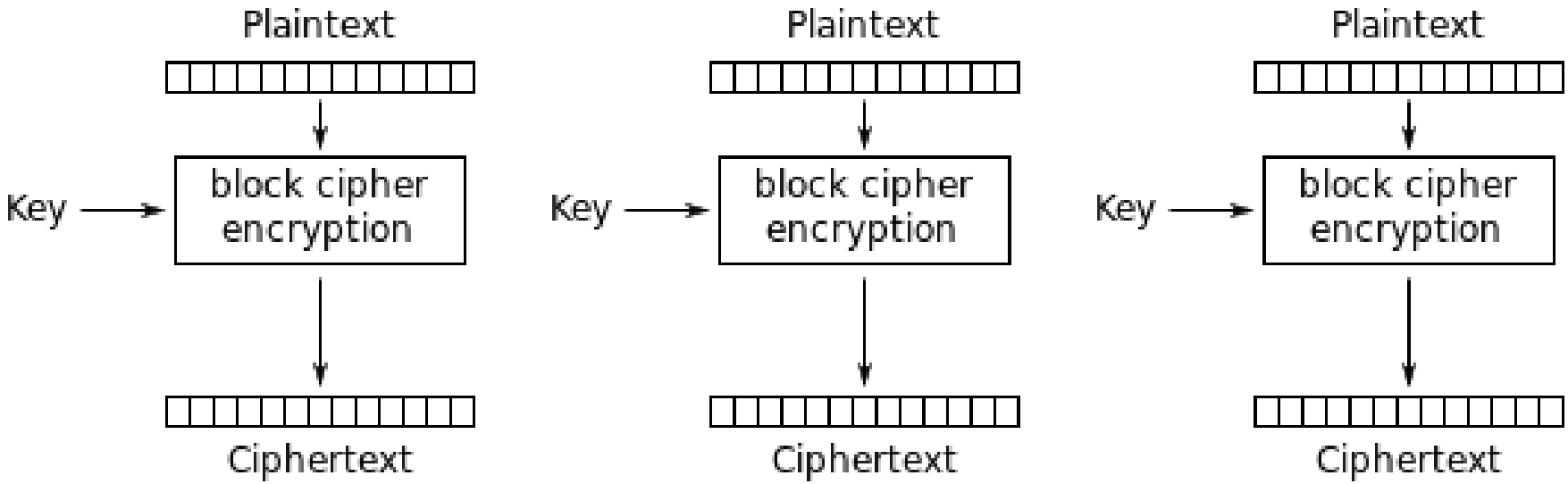
Cifras de Bloco - Modos de Operação

Cifras de bloco como AES e DES processam blocos de comprimento fixo:

- Tamanhos típicos: 64, 128, 256 bits.
- Mensagem é particionada em blocos com o tamanho definido.
- Observe que o último bloco pode não ser totalmente preenchido, e assim uma regra pré-estabelecida é utilizada para o preenchimento (padding).

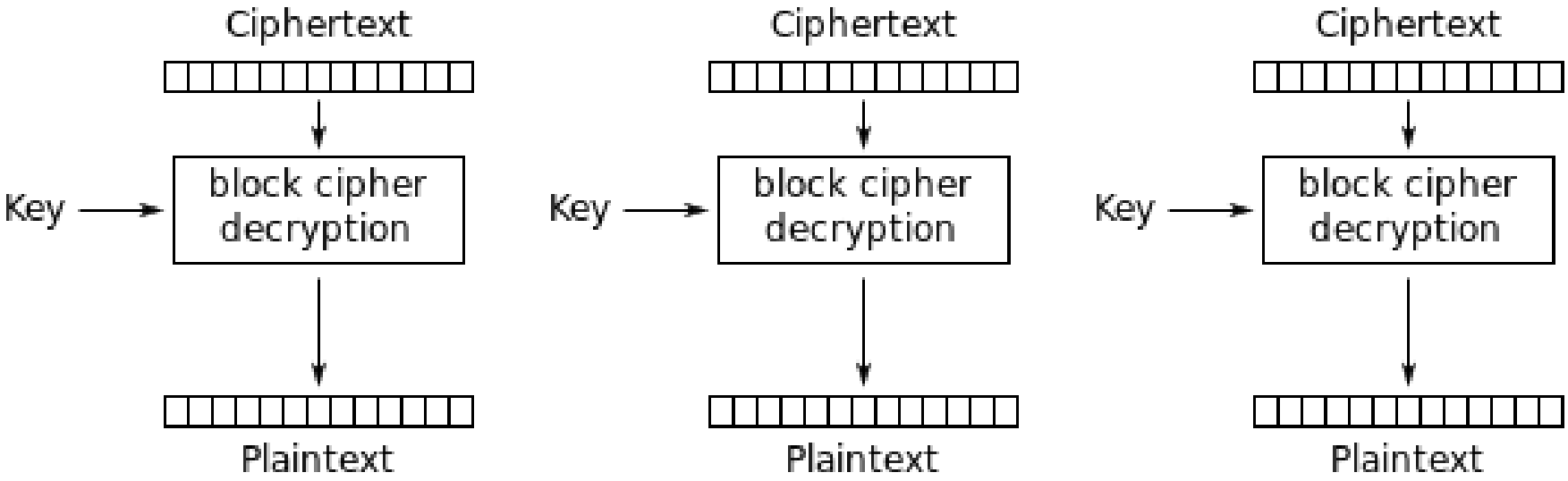
Eletronic CodeBook Model (EBC)

Cifrar



Eletronic CodeBook Model (EBC)

Decifrar



Eletronic CodeBook Model (EBC)

Características:

Assíncrona

Ciframento paralelizável? **Sim**

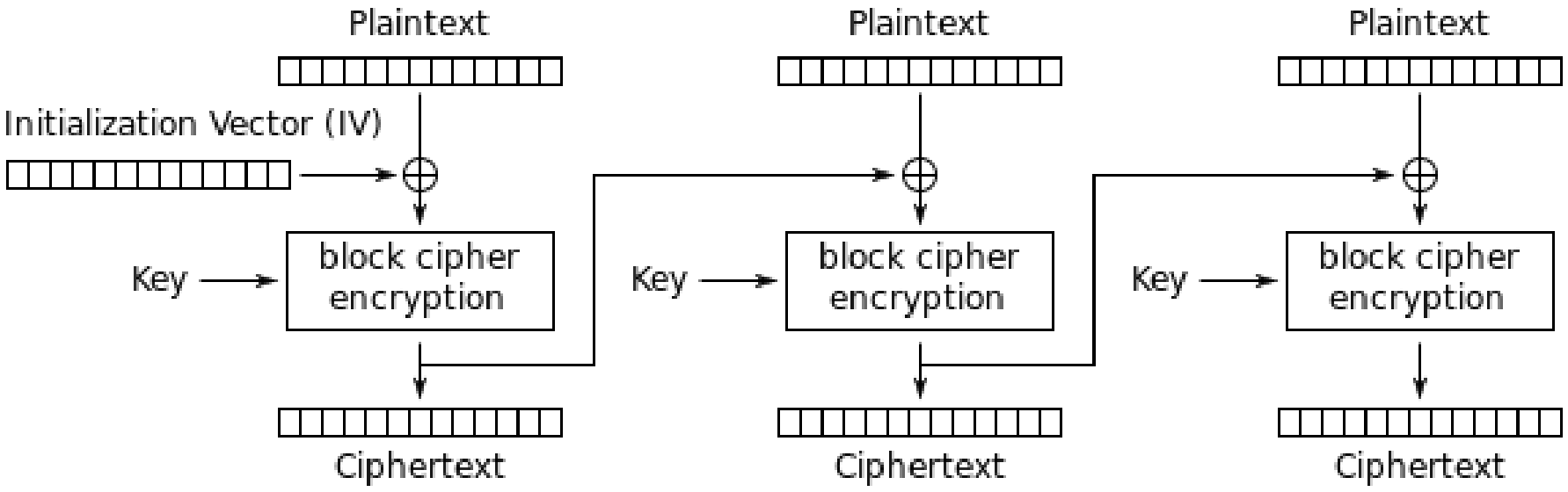
Deciframento paralelizável? **Sim**

Sensível a ataques de substituição.

Criptografia altamente determinística.

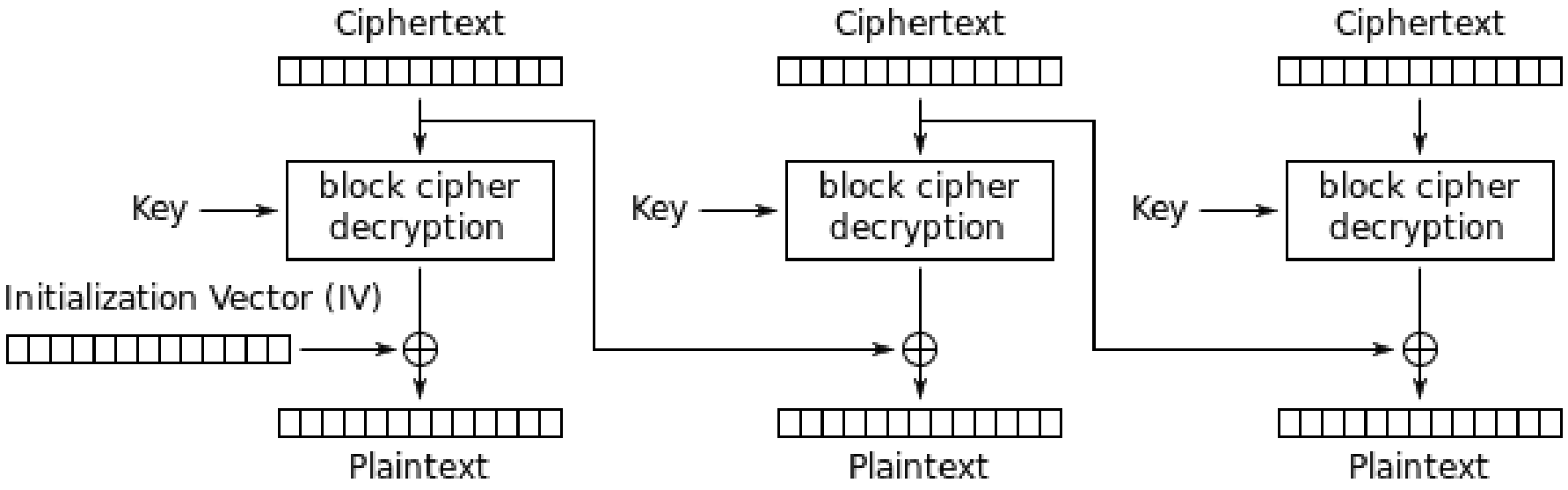
Cipher Block Chaining (CBC)

Cifrar



Cipher Block Chaining (CBC)

Decifrar



Cipher Block Chaining (CBC)

Características:

Auto sincronização

Ciframento paralelizável? **Não**

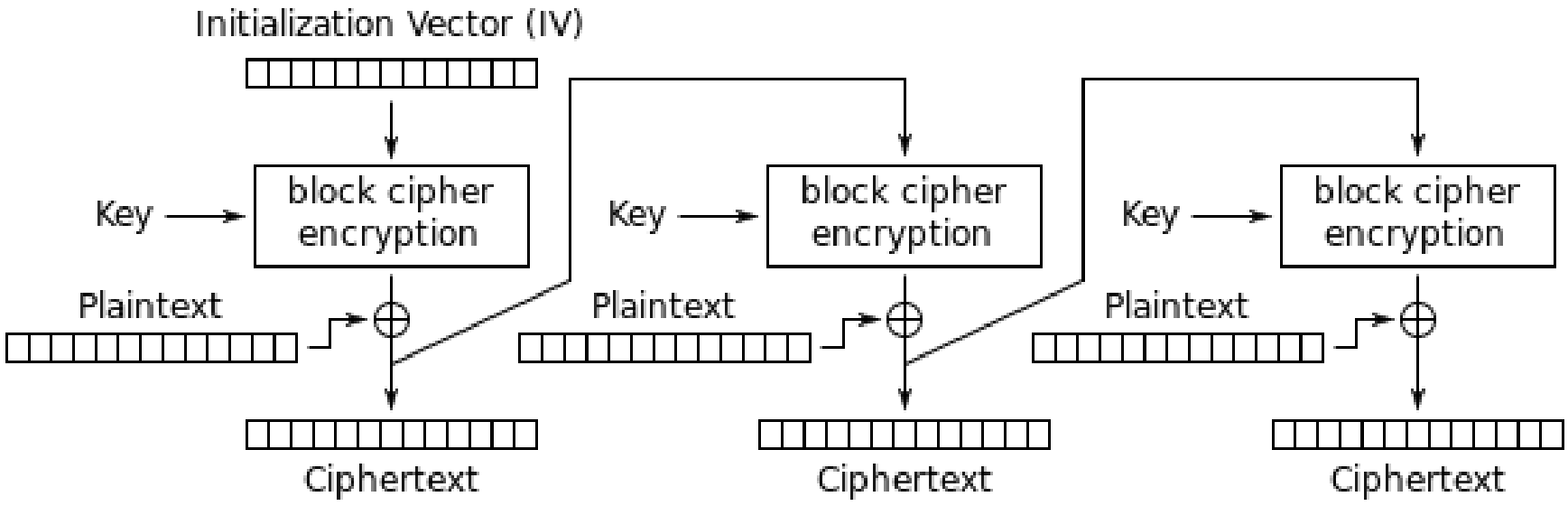
Deciframento paralelizável? **Sim**

Necessidade de IV (vetor de inicialização).

Modo de operação mais utilizado.

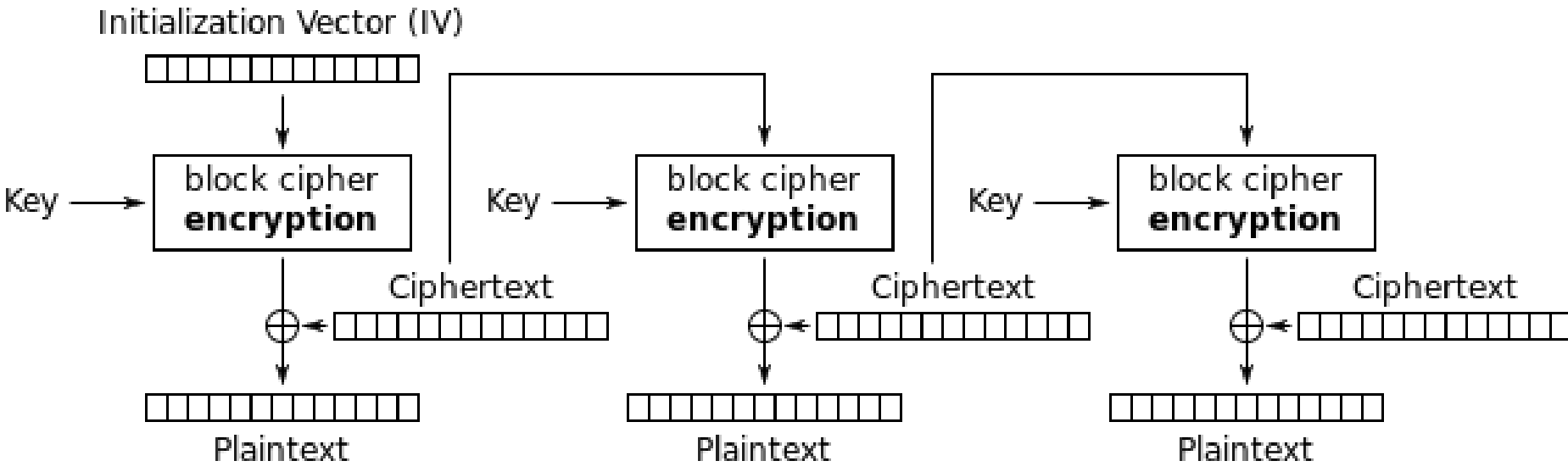
Cipher FeedBack Mode (CFB)

Cifrar



Cipher FeedBack Mode (CFB)

Decifrar



Cipher FeedBack Mode (CFB)

Características:

Auto sincronização

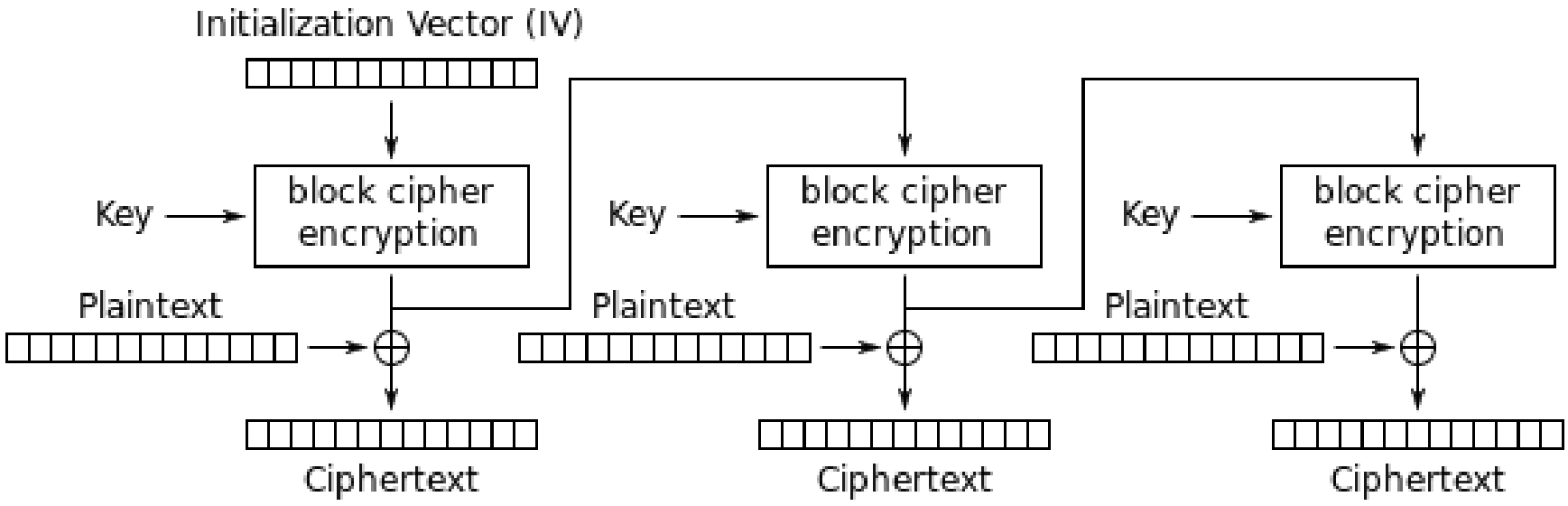
Ciframento paralelizável? **Não**

Deciframento paralelizável? **Sim**

Necessidade de IV (vetor de inicialização).

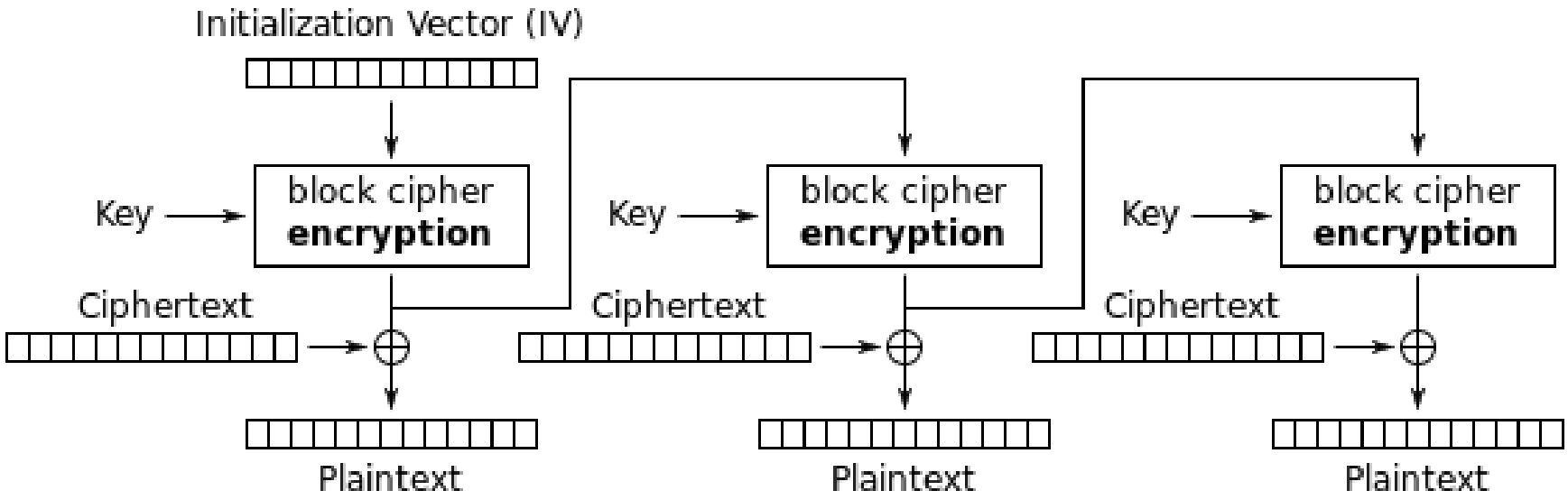
Output FeedBack Mode (OFB)

Cifrar



Output FeedBack Mode (OFB)

Decifrar



Output FeedBack Mode (OFB)

Características:

Assíncrona

Ciframento paralelizável? **Não**

Deciframento paralelizável? **Não**