Data Encryption Standard (DES)

Dr. Asem Kitana

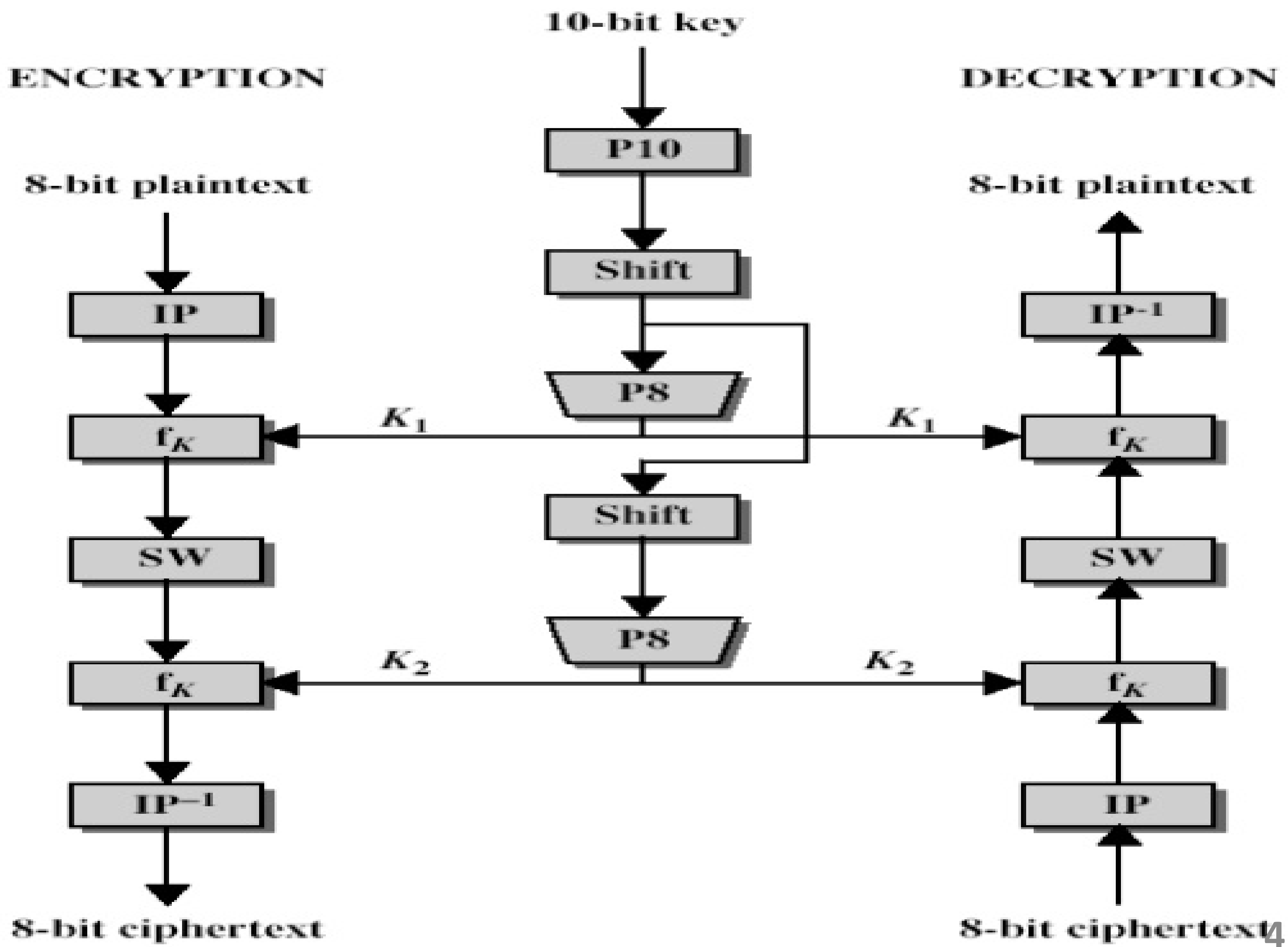
Overview of DES

- Symmetric block cipher.
- 56-bit key.
- 64-bit input block, 64-bit output block.
- Developed in 1977 by National Institute of Standards and Technology (NIST); and designed by IBM.

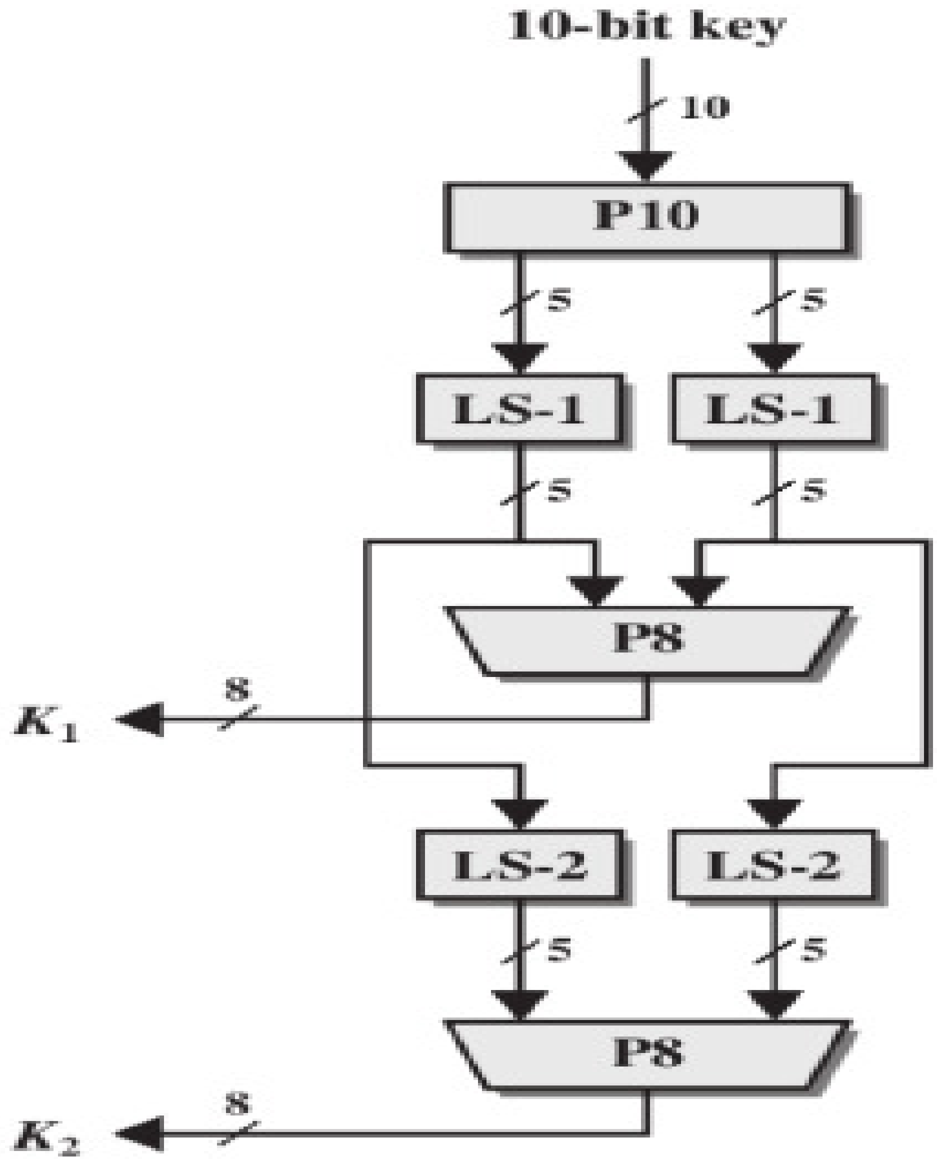
Simplified DES (S-DES)

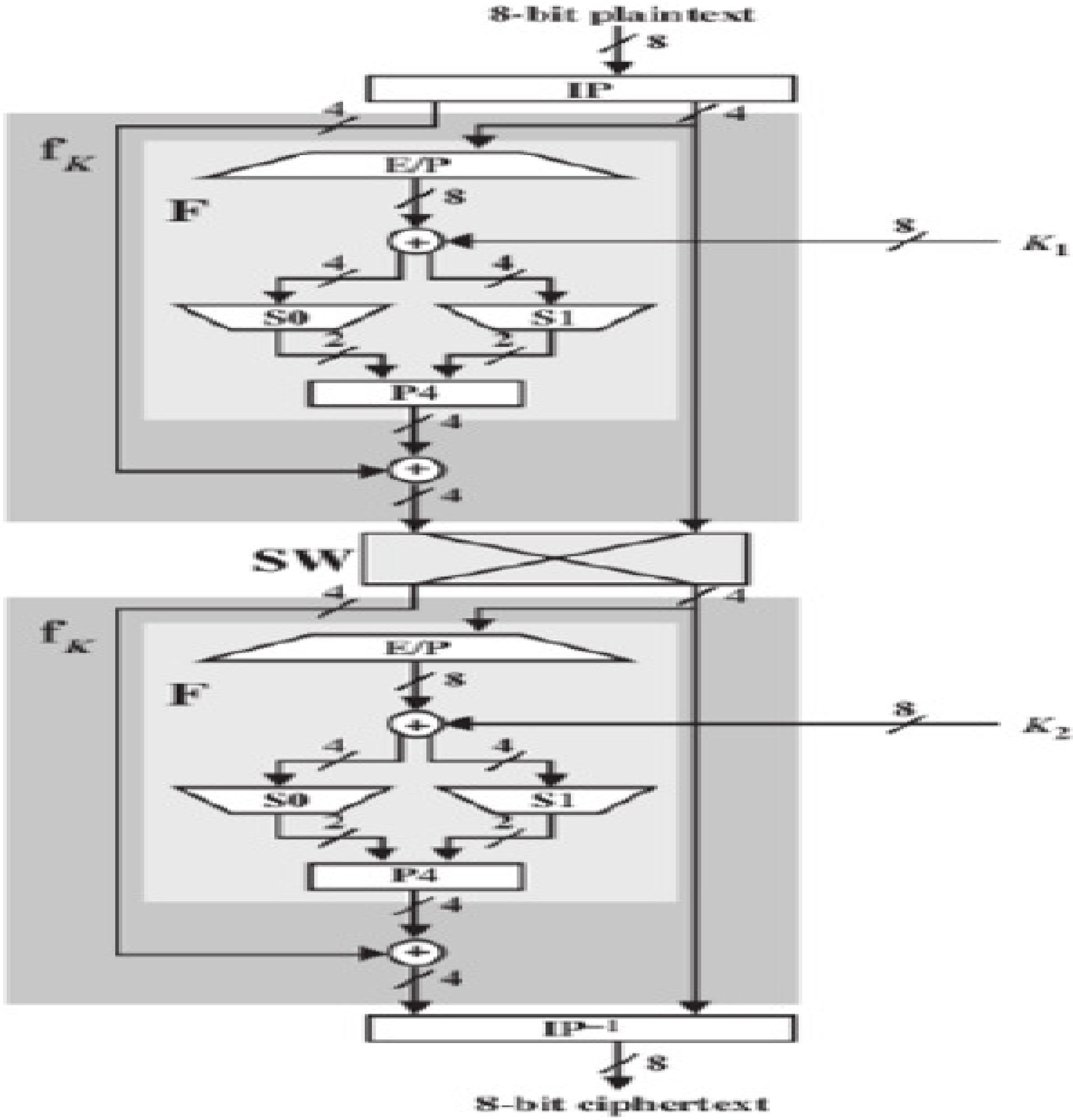
- Input (plaintext) block: 8-bits
- Output (ciphertext) block: 8-bits
- Key: 10-bits
- Rounds: 2
- Round keys generated using permutations and left shifts
- Encryption: initial permutation, round function, switch halves
- Decryption: Same as encryption, except round keys used in opposite order

S-DES Algorithm



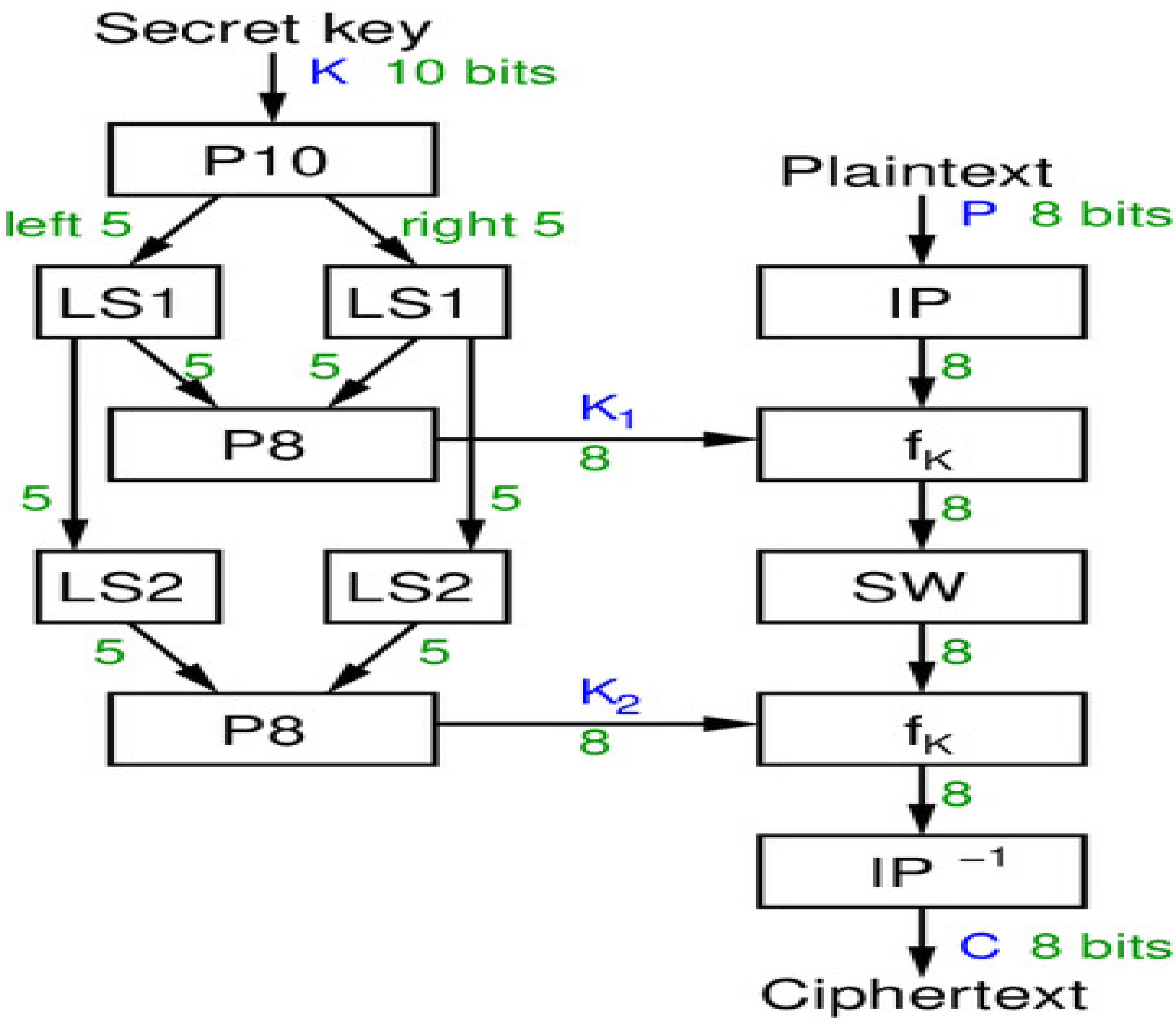
S-DES Round Keys Generation



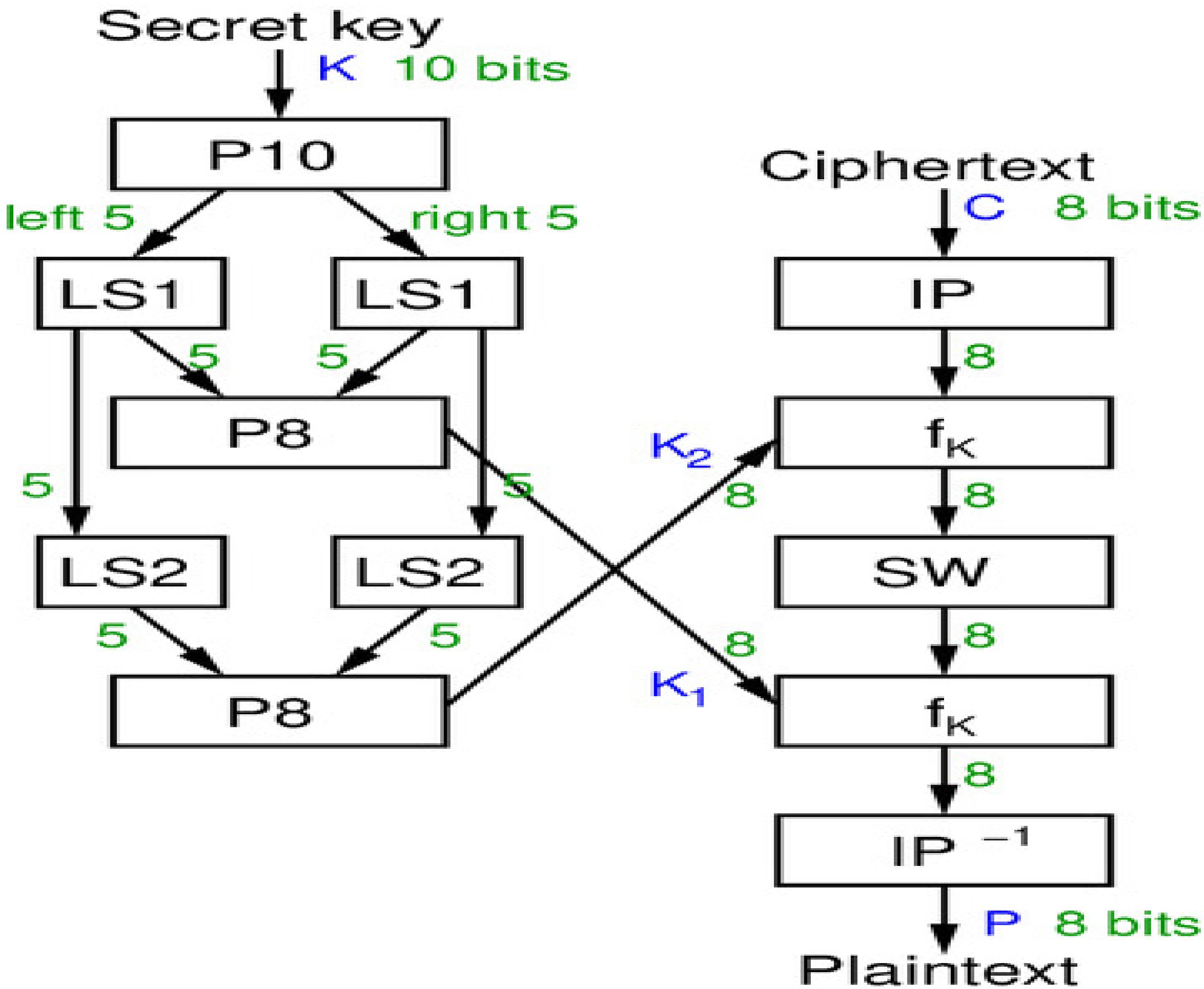


8-bit plaintext
IP
f_K
E/P
F
K_1
S0
S1
P4
SW
f_K
E/P
F
K_2
S0
S1
P4
IP^-1
8-bit ciphertext

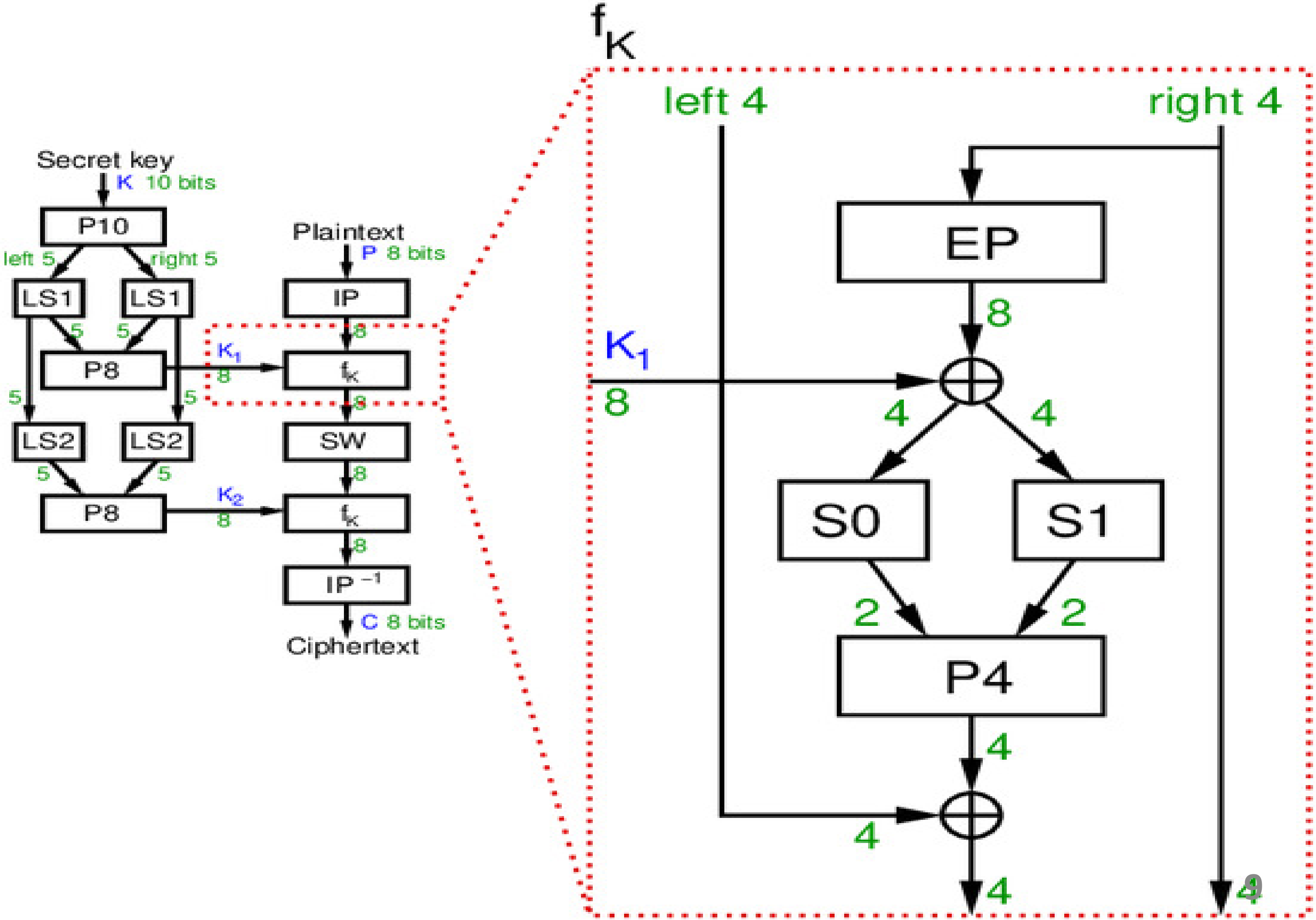
S-DES Key Generation and Encryption



S-DES Key Generation and Decryption



S-DES Round Function



S-DES Permutations

- Permutation means transposition or re-arrangement of bits.

➢ P10 (permutation)

Input	1	2	3	4	5	6	7	8	9	10
Output	3	5	2	7	4	10	1	9	8	6

➢ P8 (selection and permutation)

Input	1	2	3	4	5	6	7	8	9	10
Output	6	3	7	4	8	5	10	9		

➢ P4 (permutation)

Input	1	2	3	4
Output	2	4	3	1

S-DES Operations

- EP (Expansion and Permutation)

Input	1	2	3	4				
Output	4	1	2	3	2	3	4	1

- IP (Initial Permutation)

Input	1	2	3	4	5	6	7	8
Output	2	6	3	1	4	8	5	7

- IP^{-1} (Inverse of Initial Permutation)

Input	1	2	3	4	5	6	7	8
Output	4	1	3	5	7	2	8	6

S-DES Operations

- LS-1: left shift by 1 position
- LS-2: left shift by 2 positions
- IP^{-1}: inverse of IP, such that X = IP^{-1} (IP(X))
- SW: swap the halves (Switching Function)
- f_K: round function using round key K
- F: internal function in each round

XOR Table

- If the bits are similar, the output is 0
- If the bits are different, the output is 1

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

S-Boxes of S-DES

- S-Box considered as a matrix: input used to select row/column; selected element is output
- 4-bit input: $bit_1, bit_2, bit_3, bit_4$
- $bit_1 bit_4$ specifies row (0, 1, 2 or 3 in decimal)
- $bit_2 bit_3$ specifies column
- 2-bit output
- Indexing of S-Boxes starts from 0 to 3 for rows and columns.

S-Boxes of S-DES

$$S0 = \begin{bmatrix} 01 & 00 & 11 & 10 \\ 11 & 10 & 01 & 00 \\ 00 & 10 & 01 & 11 \\ 11 & 01 & 11 & 10 \end{bmatrix} \quad S1 = \begin{bmatrix} 00 & 01 & 10 & 11 \\ 10 & 00 & 01 & 11 \\ 11 & 00 & 01 & 00 \\ 10 & 01 & 00 & 11 \end{bmatrix}$$

S-Boxes of S-DES

$$S0 = \begin{bmatrix} 1 & 0 & 3 & 2 \\ 3 & 2 & 1 & 0 \\ 0 & 2 & 1 & 3 \\ 3 & 1 & 3 & 2 \end{bmatrix} \qquad S1 = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 2 & 0 & 1 & 3 \\ 3 & 0 & 1 & 0 \\ 2 & 1 & 0 & 3 \end{bmatrix}$$

S-DES vs. DES

	S-DES	**DES**
Block size	8 bits	64 bits
Key size	10 bits	56 bits
Rounds	2	16
IP	8 bits	64 bits
S-Boxes	2	8
Round keys	2	16
Round key size	8 bits	48 bits

S-DES summary

- Educational encryption algorithm
- S-DES expressed as functions:

$$\text{ciphertext} = \mathrm{IP}^{-1}\left(f_{K_2}\left(\mathrm{SW}\left(f_{K_1}(\mathrm{IP}(\text{plaintext}))\right)\right)\right)$$

$$\text{plaintext} = \mathrm{IP}^{-1}\left(f_{K_1}\left(\mathrm{SW}\left(f_{K_2}(\mathrm{IP}(\text{ciphertext}))\right)\right)\right)$$

- Brute force attack on S-DES is easy since only 10-bit key
- If we know plaintext and corresponding ciphertext, can we determine key? Very hard

Example1

Deploying S-DES cipher, encrypt the plaintext (01110010) using the key (1010000010).

❖ Round keys generation (k_1 and k_2):

K: 1010000010 (10-bit key)
P10: 1000001100
LS-1: 0000111000 (deployed on both halves of P10)
P8: 10100100 (represents k_1)
LS-2: 0010000011 (deployed on both halves of LS-1)
P8: 01000011 (represents k_2)

k_1 and k_2 (each 8-bit) are used as inputs in the encryption and decryption stages.

Example1, cont.

❖ Encryption:

➢ Round1:

Plaintext: 01110010

IP: 10101001

R-half: 1001

L-half: 1010

EP: 11000011 (deployed on R-half)

XOR: 01100111 (EP XOR k_1, which represents substitution)

S0: 0110 (left half of XOR deployed on S-Box 0)

row = 00 (decimal 0)

column = 11 (decimal 3)

output = 10 (row 0 and column 3 of S0)

Example1, cont.

S1: 0111 (right half of XOR deployed on S-Box 1)

row = 01 (decimal 1)

column = 11 (decimal 3)

output = 11 (row 1 and column 3 of S1)

S0S1: 1011

P4: 0111 (deployed on S0S1)

XOR: 1101 (P4 XOR L-half)

Result: 11011001 (XOR + R-half)

End of round1

SW: 10011101 (swapping the two halves of Result)

The output of SW function (10011101) is used as input in round2.

Example1, cont.

➢ <u>Round2:</u>

SW: 10011101

R-half: 1101

L-half: 1001

EP: 11101011 (deployed on R-half)

XOR: 10101000 (EP XOR k_2)

S0: 1010 (left half of XOR deployed on S-Box 0)

row = 10 (decimal 2)

column = 01 (decimal 1)

output = 10 (row 2 and column 1 of S0)

Example1, cont.

S1: 1000 (right half of XOR deployed on S-Box 1)

row = 10 (decimal 2)

column = 00 (decimal 0)

output = 11 (row 2 and column 0 of S1)

S0S1: 1011

P4: 0111 (deployed on S0S1)

XOR: 1110 (P4 XOR L-half)

Result: 11101101 (XOR + R-half)

IP^{-1}: 01110111

The ciphertext is (01110111)

Example2

- Deploying S-DES algorithm, decrypt the ciphertext (00111000) using the key (1010000010).

DES Algorithm