# CMPE 132 Information Security

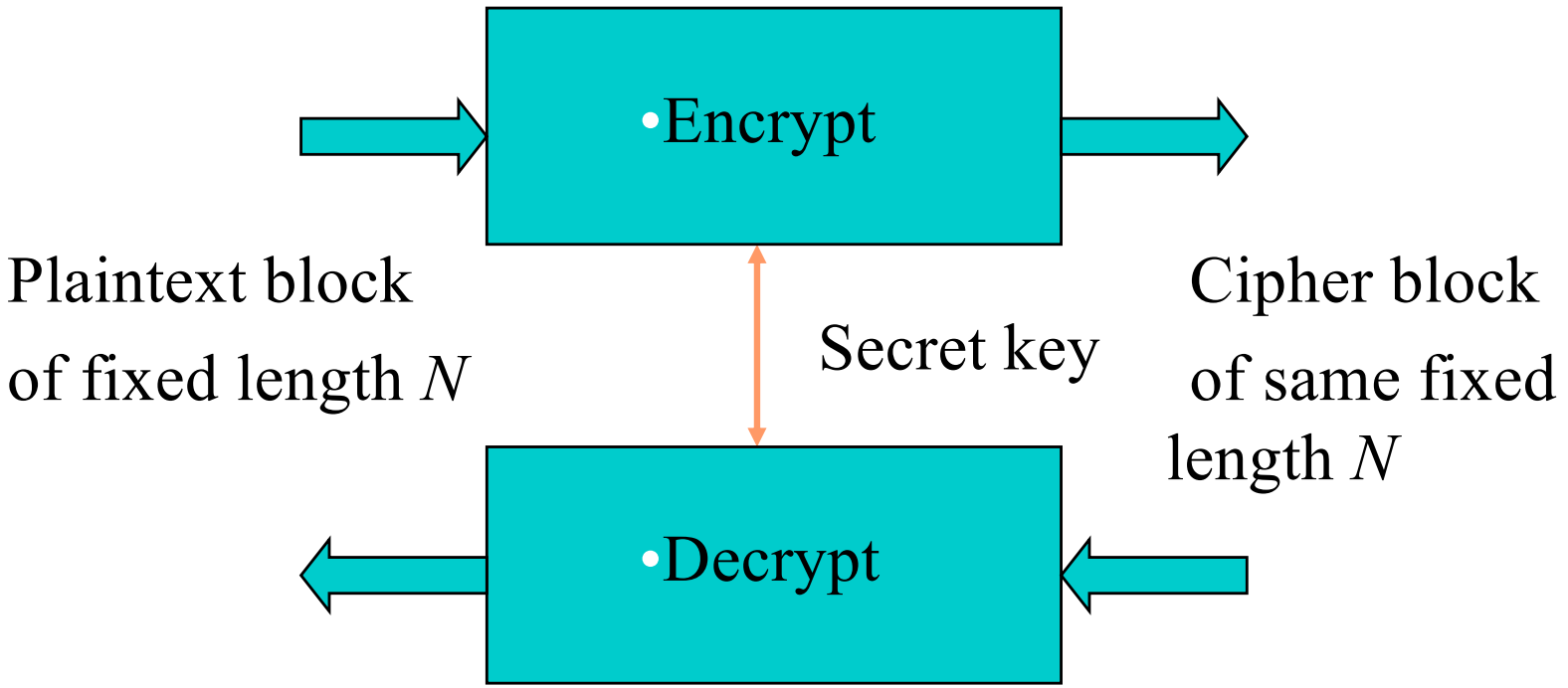
## Secret Key Cryptography

## DES

Dr. Younghee Park

SAN JOSÉ STATE UNIVERSITY

# Generic Block Cipher



# Generic Block Encryption (Cont'd)

- Convert one block to another: one-to-one
- Long enough to avoid known-plaintext attack, but not too long (performance).
  - 64 bits typical
- Naïve: $2^{64}$ input values, 64 bits each
- Output should look random
  - No correlation between plaintext and ciphertext
  - Bit spreading

# Block Cipher Principles

- Most symmetric block ciphers are based on a **Feistel Cipher Structure**
- It must be able to **decrypt** ciphertext to recover messages efficiently
- Block ciphers look like an extremely large substitution
- It needs table of $2^{64}$ entries for a 64-bit block
- Create from smaller building blocks

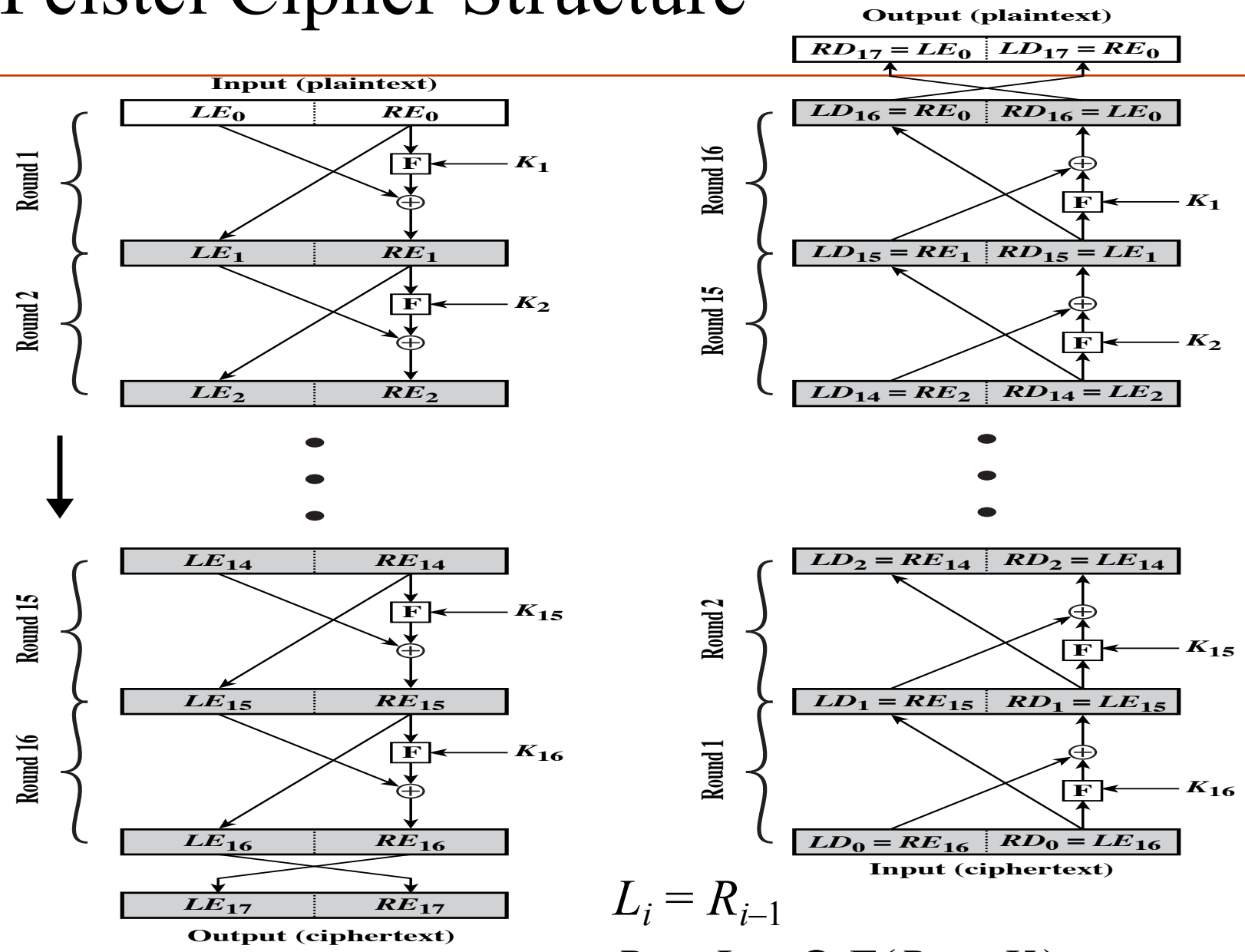
# Feistel Cipher

- Confusion
  - Make the relationship between the plaintext/key and the ciphertext as complex as possible
  - Achieved by complex substitution algorithm.
- Diffusion
  - Dissipate the statistical structure of the plaintext
  - Achieved by having each plaintext digit affect many ciphertext digits
  - Equivalently, having each ciphertext digit affected by many plaintext digits.

# Feistel Cipher (cont'd)

- Is a practical application of a 1949 proposal by Claude Shannon to develop a product cipher that alternates confusion and diffusion functions
  - Alternate diffusion and confusion
  - Equivalently, alternate substitution and permutation
- Horst Feistel devised the **feistel cipher**
  - based on concept of invertible product cipher
- Is the structure used by many significant symmetric block ciphers currently in use

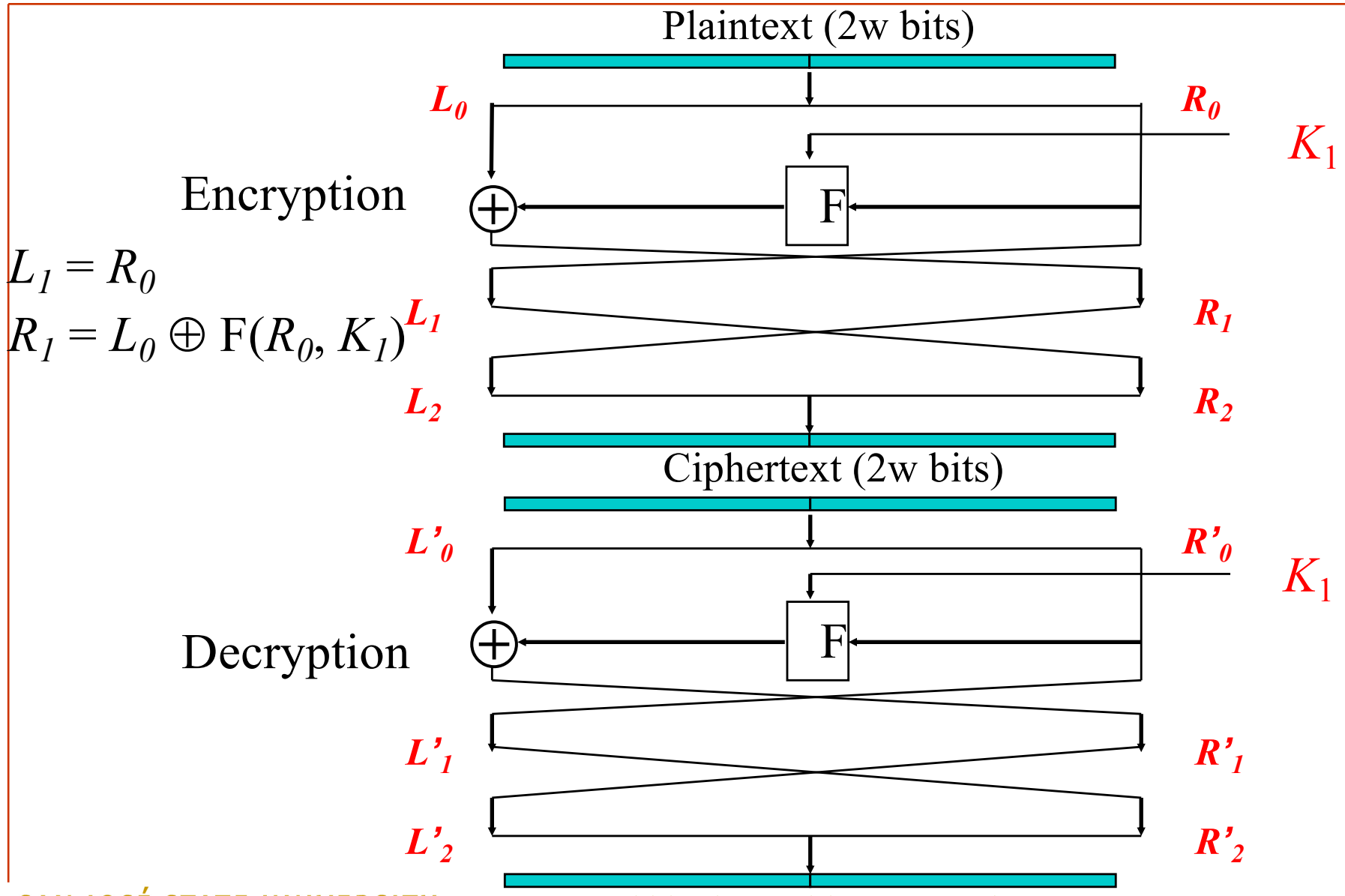
# Feistel Cipher Structure




$$L_i = R_{i-1}$$

$$R_i = L_{i-1} \oplus \mathrm{F}(R_{i-1}, K_i)$$

# One Round Feistel Cipher



# Feistel Example

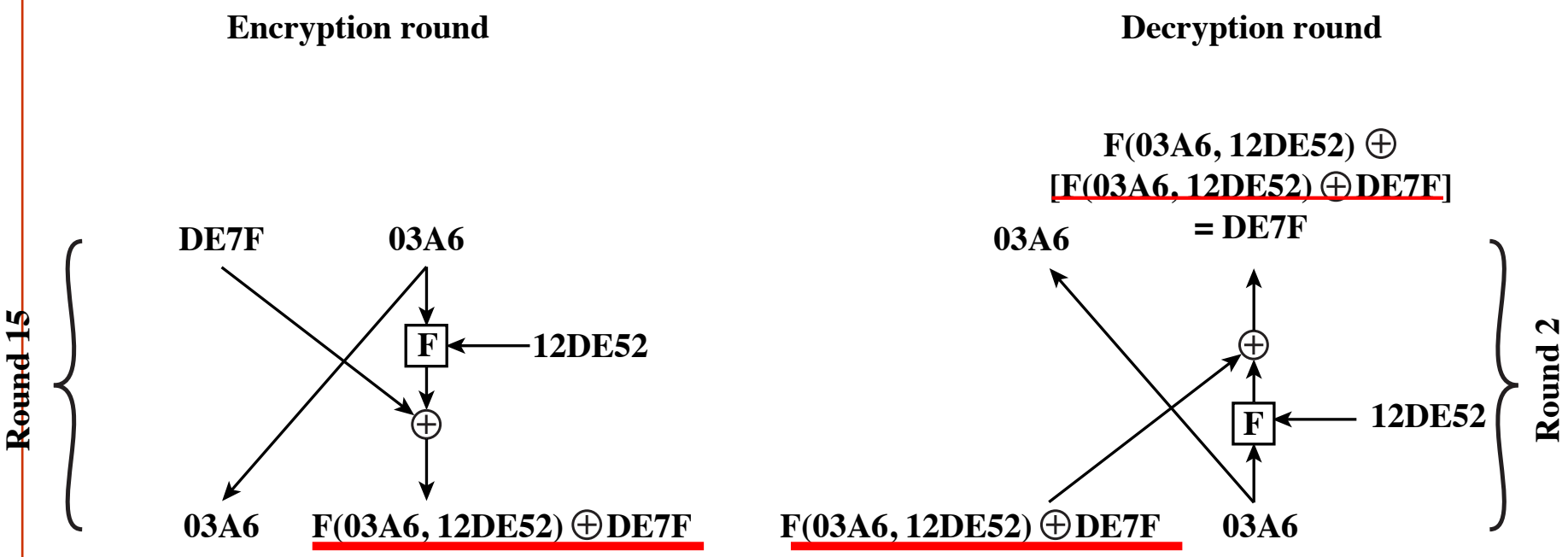



**Figure 3.4 Feistel Example**

# Realization of Feistel Cipher

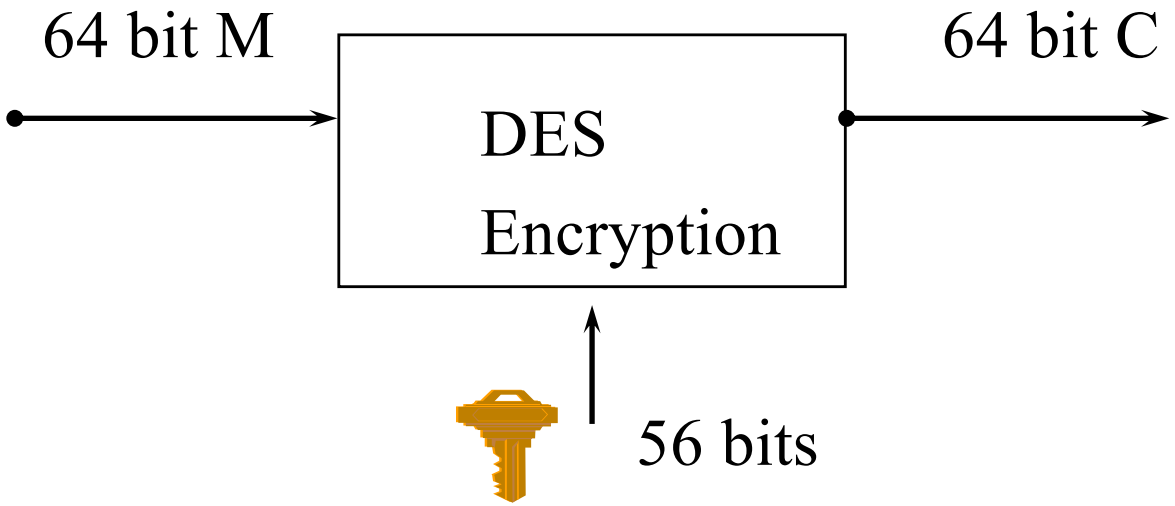
- Parameters
  - Block Size: typically 64 bits
  - Key Size: commonly 128 bits
  - Number of Rounds: 16
  - Subkey Generation algorithm
  - Round Function

# DES (Data Encryption Standard)

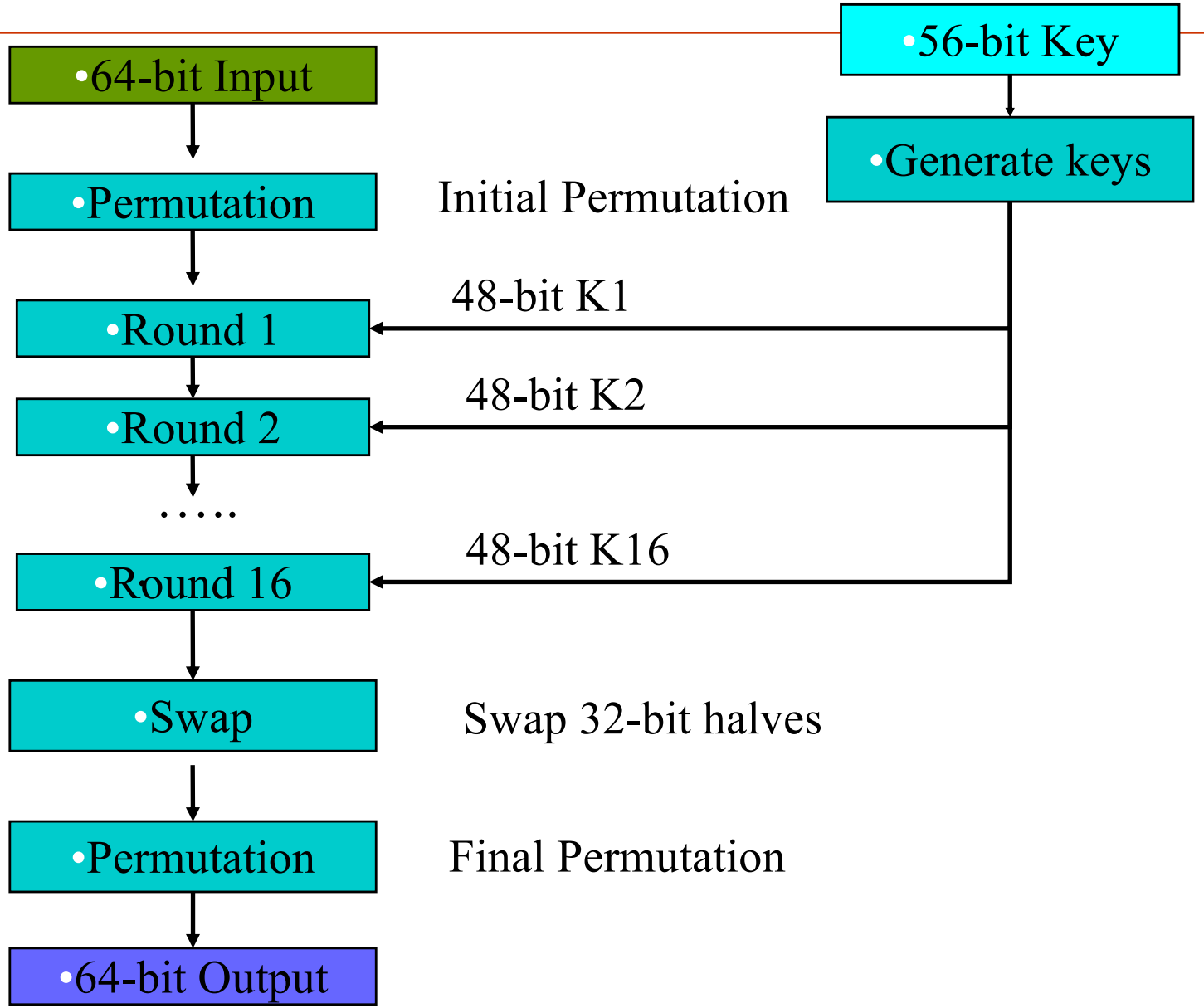
SAN JOSÉ STATE UNIVERSITY

# DES (Data Encryption Standard)

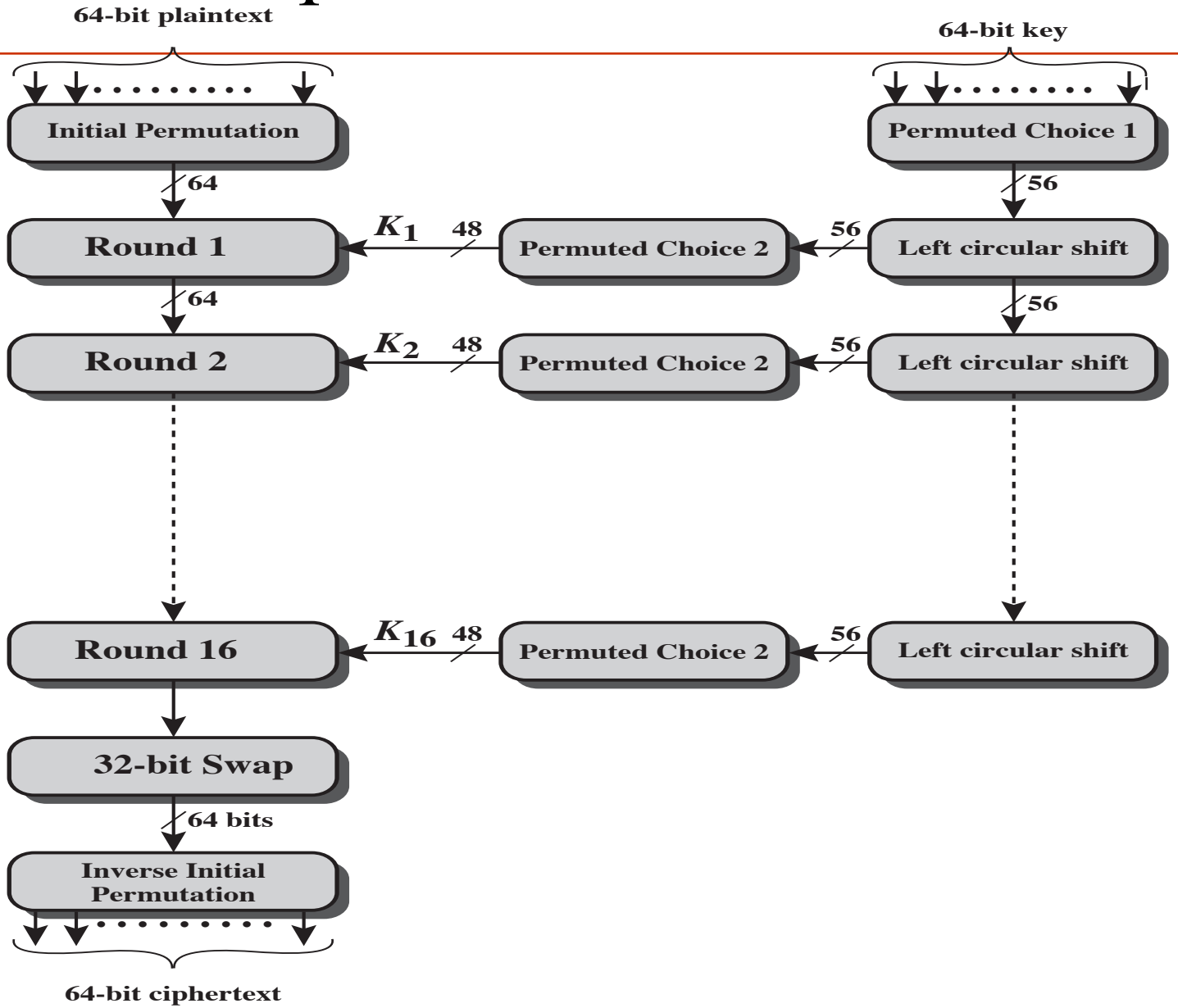
- Published in 1977, standardized in 1979, expired in 1998.
- Similar structure to Feistel cipher
- Key: 64 bit quantity=8-bit parity+56-bit key
  - Every 8th bit is a parity bit.
- 64 bit input, 64 bit output.



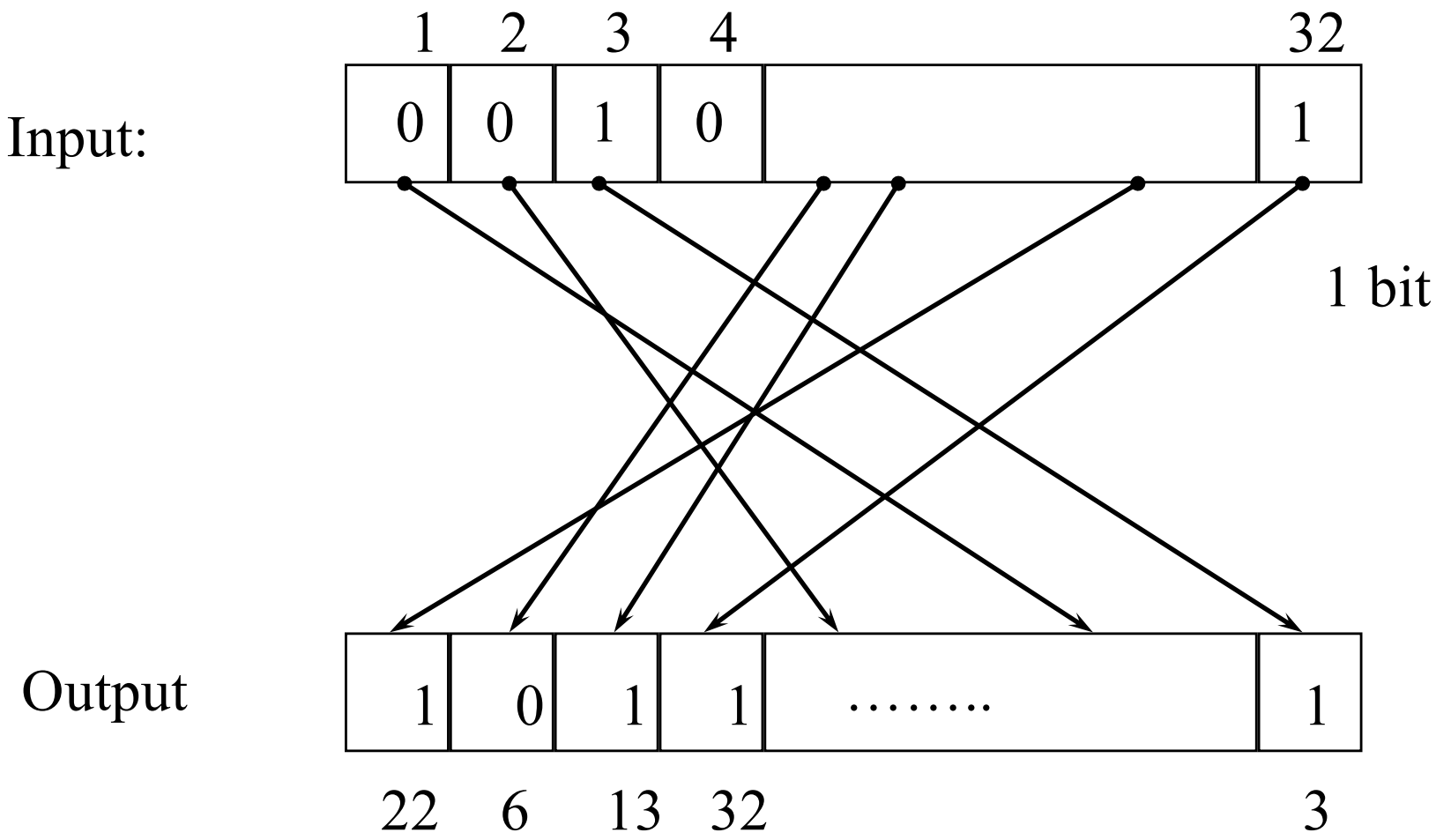
# DES Top View


•56-bit Key
•Generate keys
•64-bit Input
•Permutation
Initial Permutation
48-bit K1
•Round 1
48-bit K2
•Round 2
.....
48-bit K16
•Round 16
•Swap
Swap 32-bit halves
•Permutation
Final Permutation
•64-bit Output

# General Depiction of DES



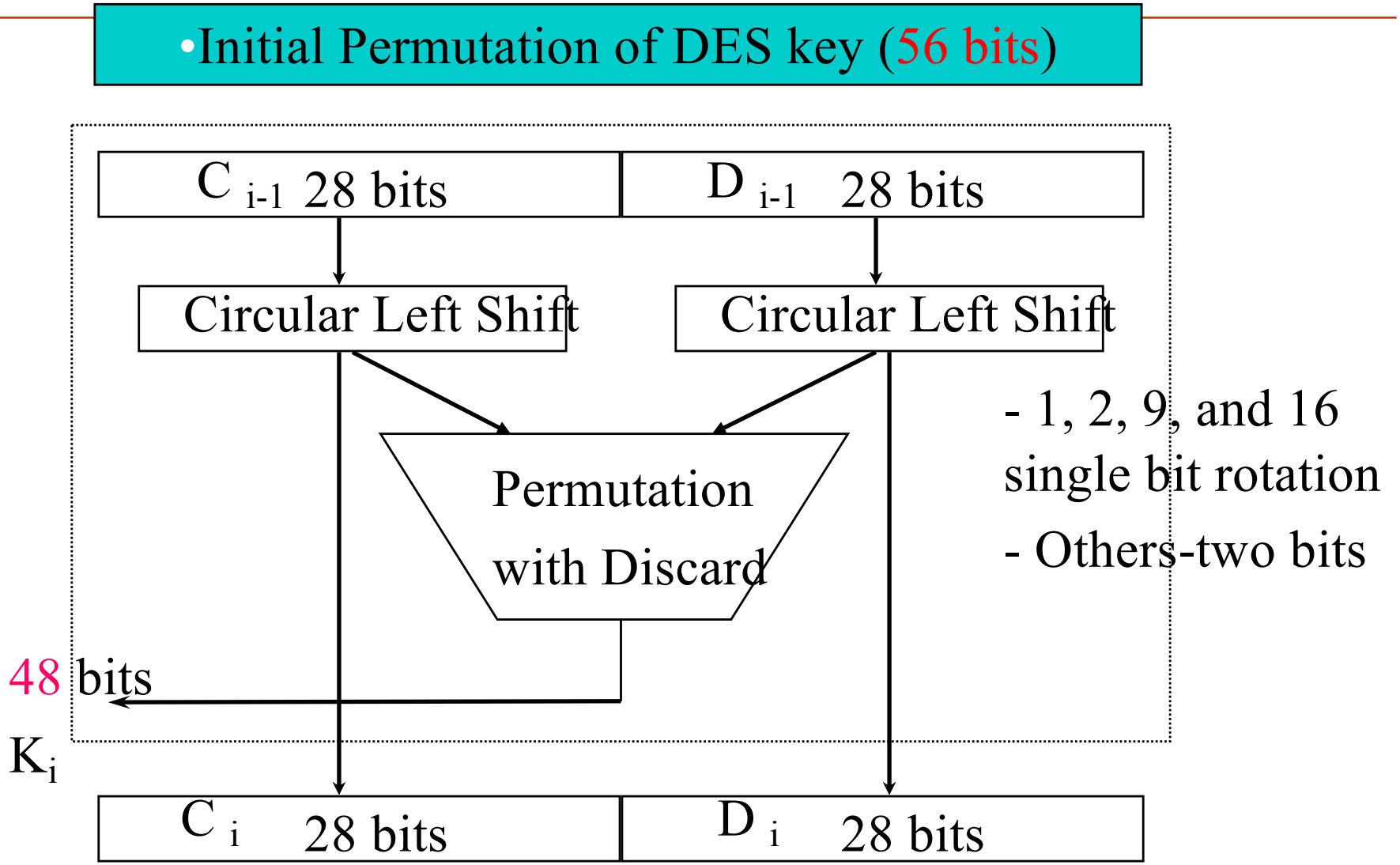
# Bit Permutation (1-to-1)



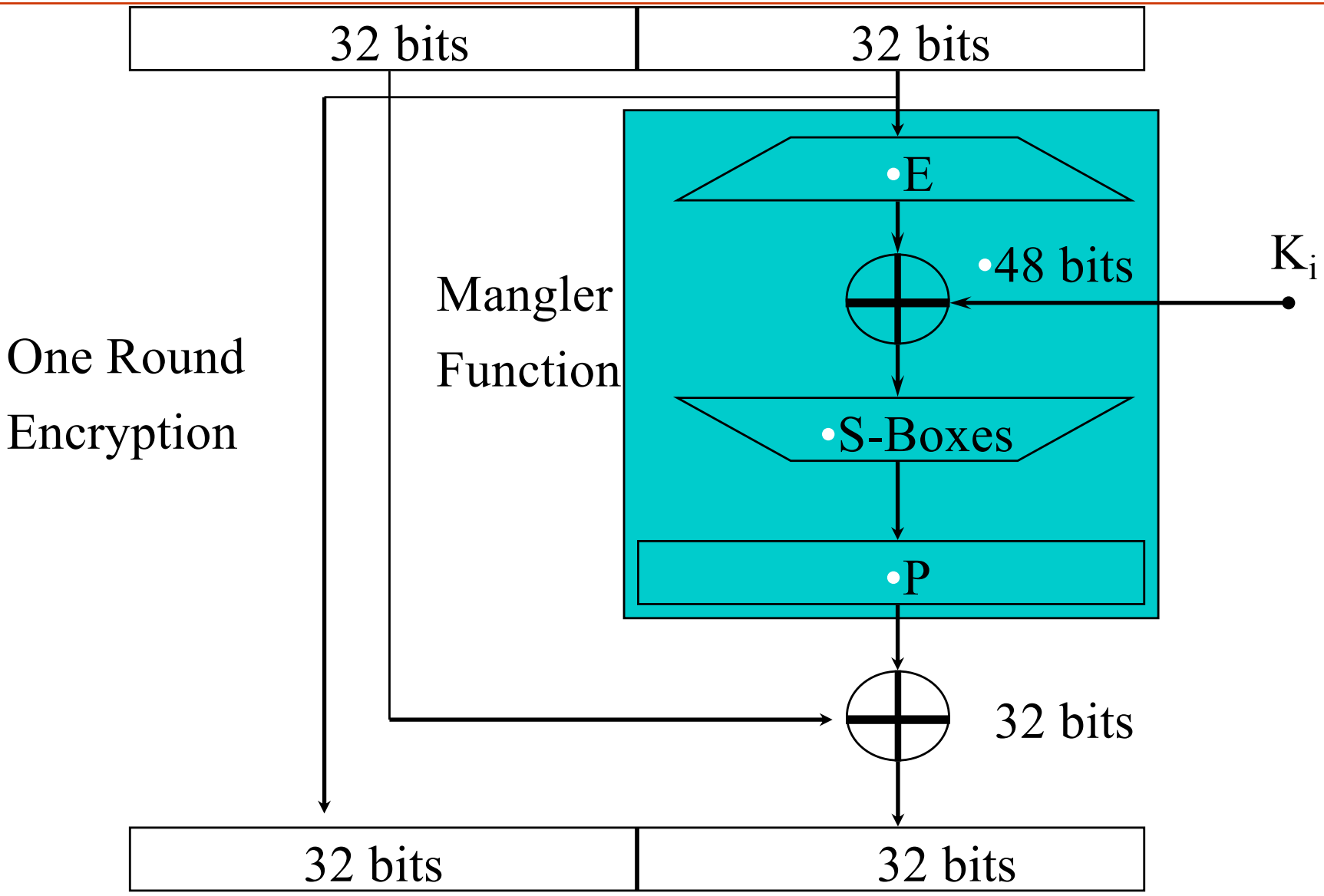
# Initial and Final Permutations

- Initial permutation (IP)
- View the input as M: 8 X 8 bit matrix
- Transform M into M' in two steps
  - Transpose row x into column (9-x), 0<x<9
  - Apply permutation on the rows:
    - For even row y, it becomes row y/2
    - For odd row y, it becomes row (5+y/2)
- Final permutation FP = $IP^{-1}$

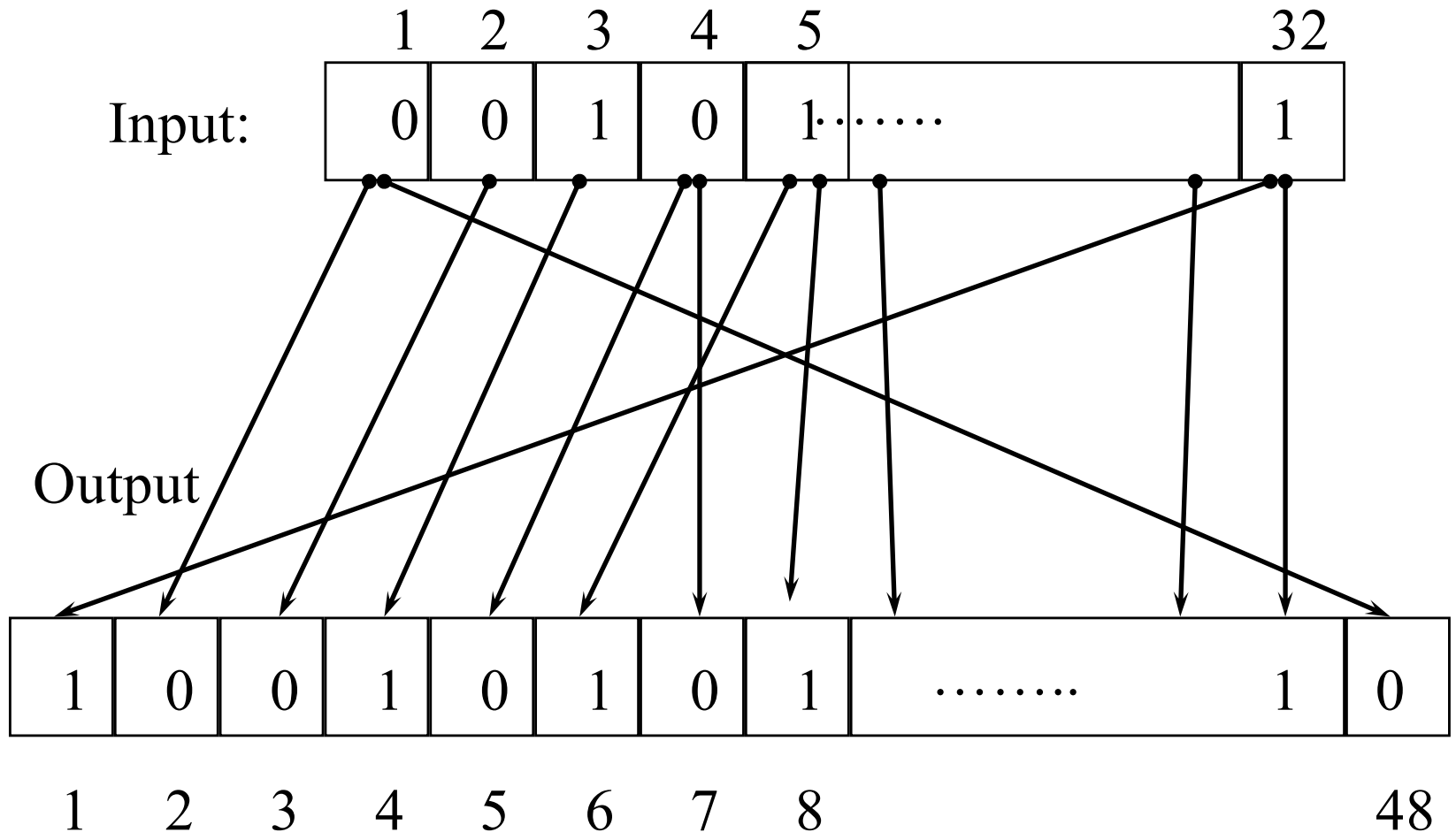
# Per-Round Key Generation



# A DES Round


32 bits
32 bits
E
48 bits
$K_i$
Mangler
Function
One Round
Encryption
S-Boxes
P
32 bits
32 bits
32 bits

# Bits Expansion

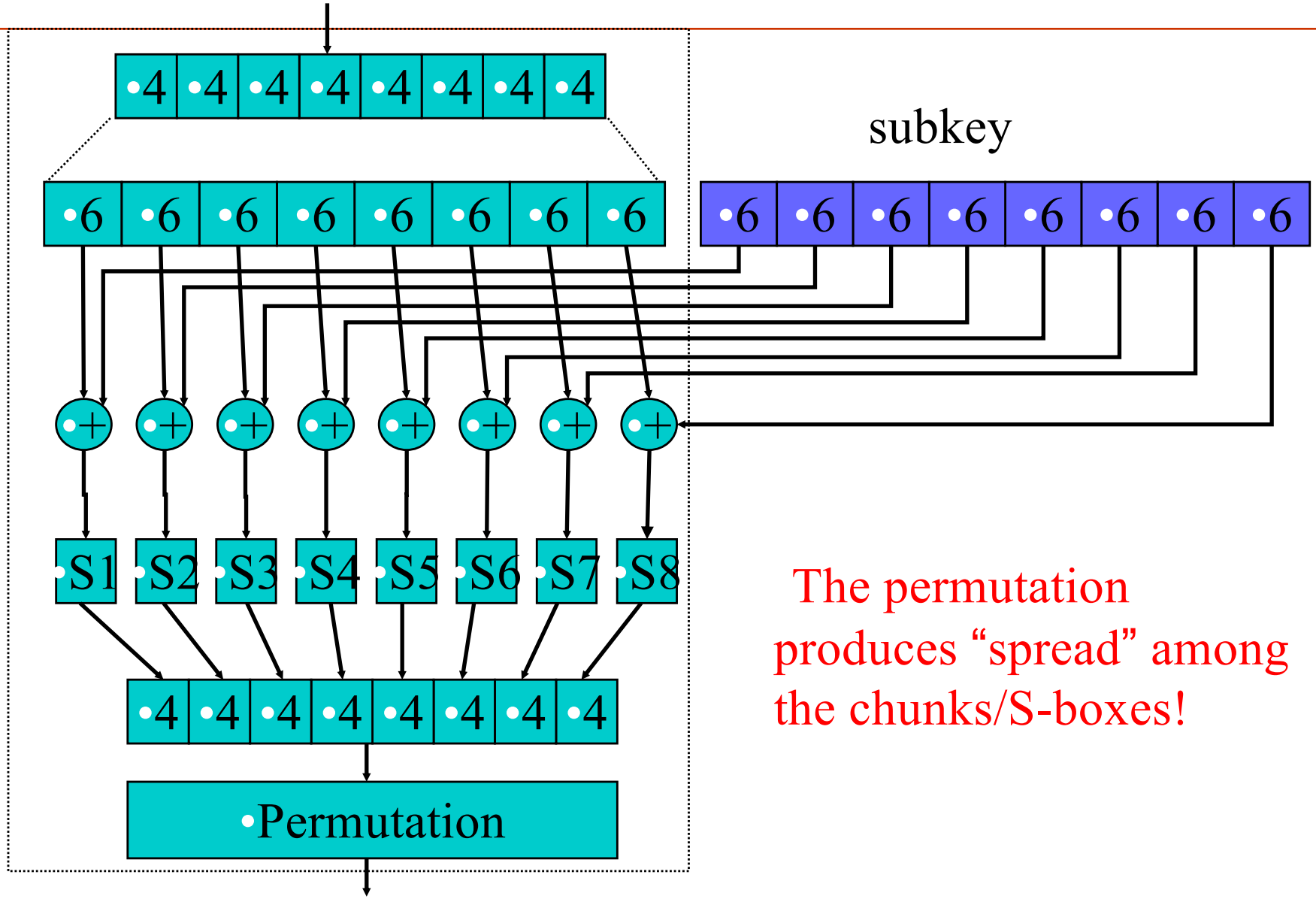


# E Box of DES

How is the E Box defined?

| | | | | | |
|---|---|---|---|---|---|
| 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 |

# Mangler Function

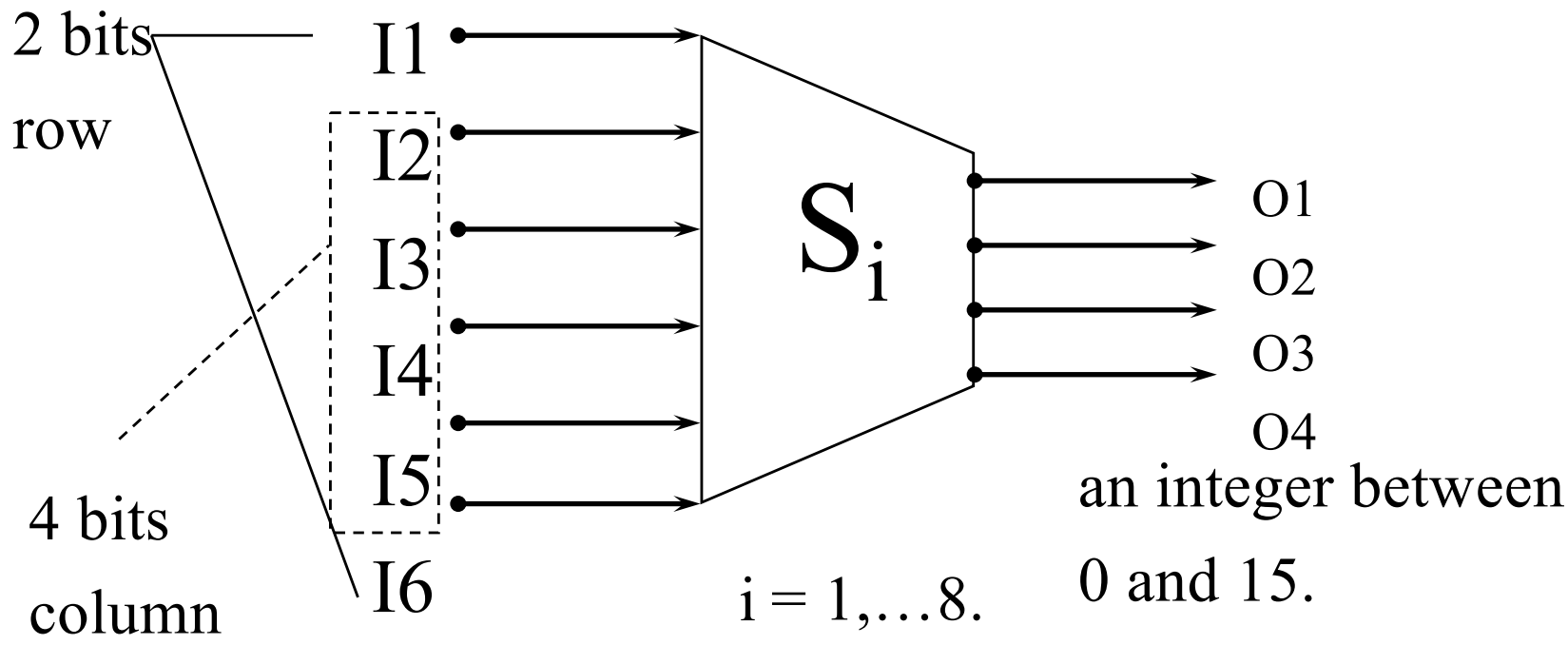



The permutation produces "spread" among the chunks/S-boxes!

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

# S-Box (Substitute and Shrink)

- 48 bits ==> 32 bits. (8*6 ==> 8 *4)
- 2 bits used to select amongst 4 permutations for the rest of the 4-bit quantity



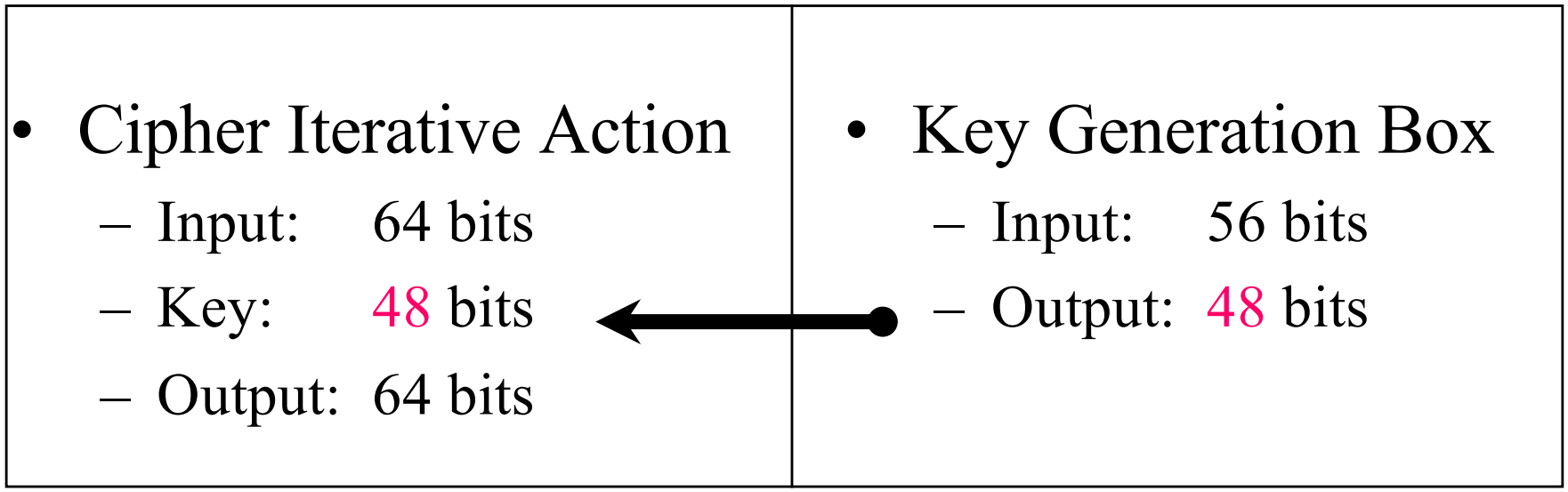
# S1: (p. 71)

Each row and column contain different numbers.

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | … | 15 |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 14 | 4 | 13 | 1 | 2 | 15 | 11 | | |
| 1 | 0 | 15 | 7 | 4 | 14 | 2 | 13 | | |
| 2 | 4 | 1 | 14 | **8** | 13 | 6 | 2 | | |
| 3 | 15 | 12 | 8 | 2 | 4 | 9 | 1 | | |

Example: input: 100110 output: ???

# DES Standard




One round (Total 16 rounds)

# Summary: DES Round Structure

- Uses two 32-bit L & R halves
- As for any Feistel cipher can describe as:

  $L_i = R_{i-1}$

  $R_i = L_{i-1} \oplus \mathrm{F}(R_{i-1}, K_i)$

- F takes 32-bit R half and 48-bit subkey:
  - expands R to 48-bits using permute E
  - adds to subkey using XOR
  - passes through 8 S-boxes to get 32-bit result
  - finally permutes using 32-bit permute P

# Avalanche Effect

- A small change in either the plaintext or the key should produce a significant change in the ciphertext.
- DES has a strong avalanche effect.
- Example
  - Plaintexts: 0X0000000000000000 and 0X8000000000000000
  - Same key: 0X016B24621C181C32
  - 34 bits difference in cipher-texts
  - Similar result with same plaintext and slightly different keys

# Avalanche Effect

- Key desirable property of encryption algorithm
- Where a change of **one** input or key bit results in changing approximate **half** output bits
- Making attempts to "home-in" by guessing keys impossible
- DES exhibits strong avalanche

# DES Summary

- Simple, easy to implement:
  - Hardware/gigabits/second, software/megabits/second
- 56-bit key DES maybe acceptable for non-critical applications but triple DES (DES3) should be secure for most applications today
- Supports several operation modes: ECB CBC, OFB, CFB

# Strength of DES – Key Size

- 56-bit keys have $2^{56} = 7.2 \times 10^{16}$ values
- Brute force search looks hard
- But, recent advances have shown is possible
  - in 1997 on Internet in a few months
  - in 1998 on dedicated h/w (EFF) in a few days
    - http://en.wikipedia.org/wiki/EFF_DES_cracker
  - in 1999 above combined in 22hrs!
- Still must be able to recognize plaintext
- Must now consider alternatives to DES

# Difference between DES and AES

- DES is really old while AES is relatively new
- DES is breakable while AES is still unbreakable
- DES uses a much smaller key size compared to AES
- DES uses a smaller block size compared to AES
- DES uses a balanced Feistel structure while AES uses substitution-permutation

# Summary

- Traditional Block Cipher Structure
  - Stream ciphers
  - Block ciphers
  - Feistel cipher
  - Example) Data Encryption Standard (DES)
    - Encryption/Decryption/Avalanche effect/Timing Attacks
- Block cipher design principles
  - DES design criteria (Strict avalanche criterion/Bit Independent Criterion)
  - Number of rounds
  - Design of function F
  - Key schedule algorithm

# Study Tips

- S-box
- DES features