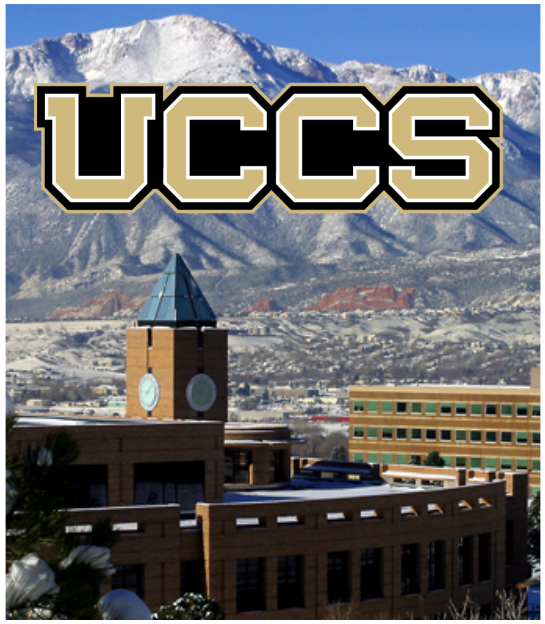


Symmetric Cryptography

Block Cipher and DES

Sang-Yoon Chang, Ph.D.

Modern Cipher (vs. Classical Cipher)

Digital computer communications
based on bits

Product cipher

More sophisticated techniques

Module: Block Cipher and DES

Block Cipher vs. Stream Cipher

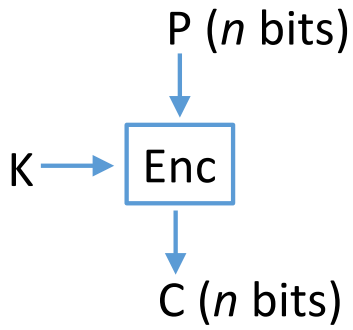
Ideal Block Cipher

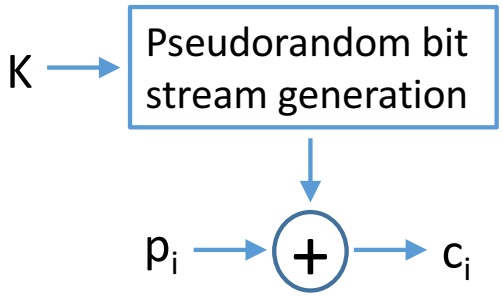
Feistel Cipher

Data Encryption Standard (DES)

Block Cipher vs. Stream Cipher

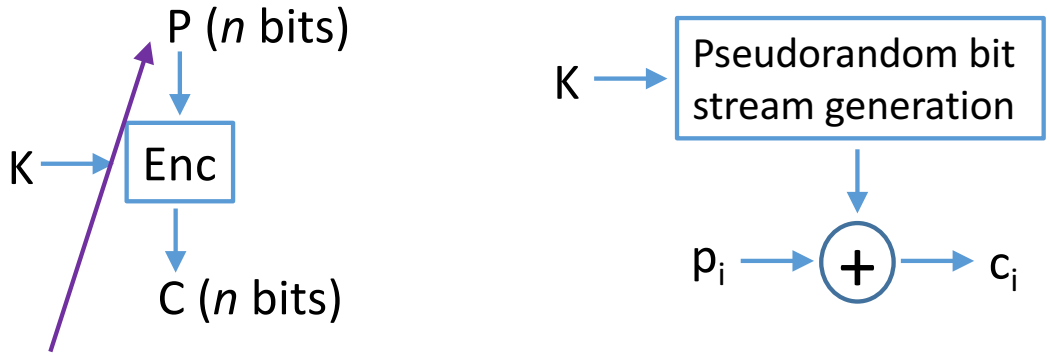
Block cipher (left) processes in blocks (multiple bits) while stream cipher (right) processes them a bit/byte at a time





Block Cipher vs. Stream Cipher

Block cipher (left) processes in blocks (multiple bits) while stream cipher (right) processes them a bit/byte at a time



Pad bits if the last block is incomplete

Block Cipher Function Requirements

Block cipher function: n bits → n bits
2^n possible block options

Reversible function
Dec(K,(Enc(K,X)))=X, for all X

Given the key K, the computation of the function is deterministic and easy

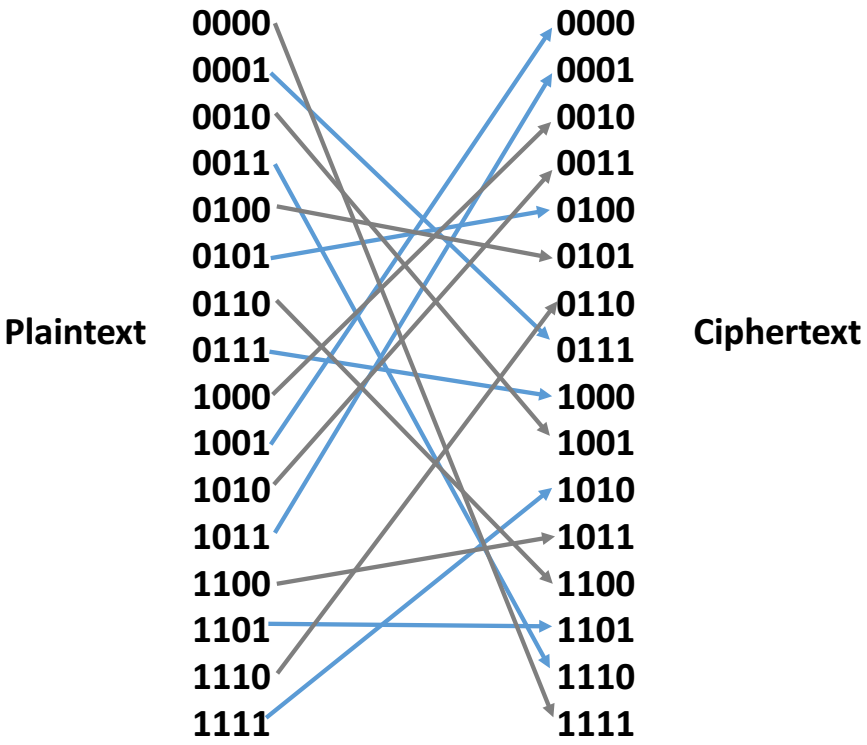
Ideal Block Cipher

Block cipher function: n bits → n bits
2^n possible block options

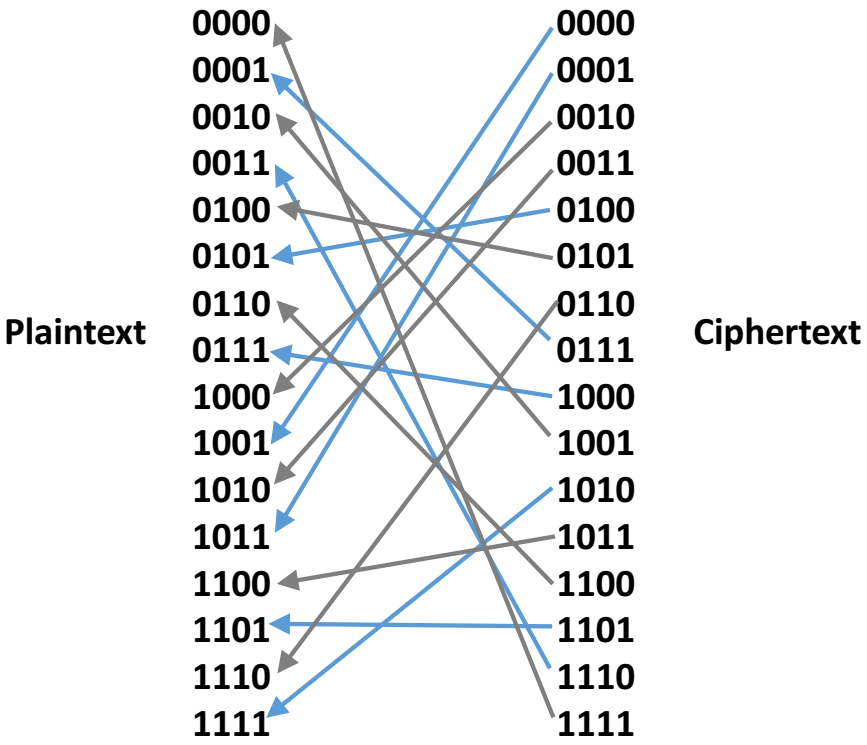
Ideal block cipher supports the maximum number of encryption mappings

Reversible transformation
$2^n!$ Possible transformations or keys

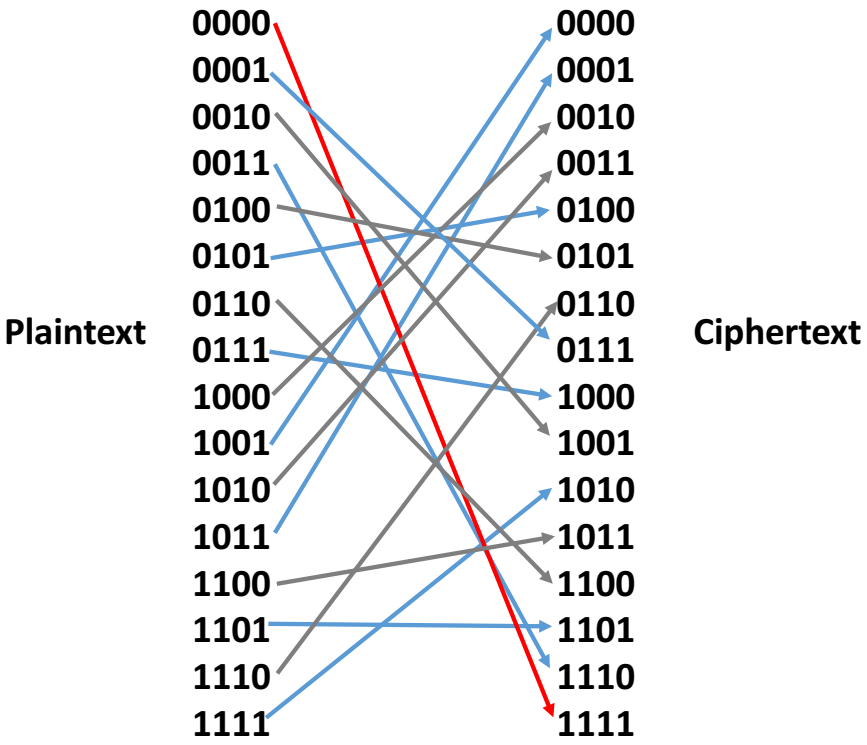
Ideal Block Cipher Example (n=4)



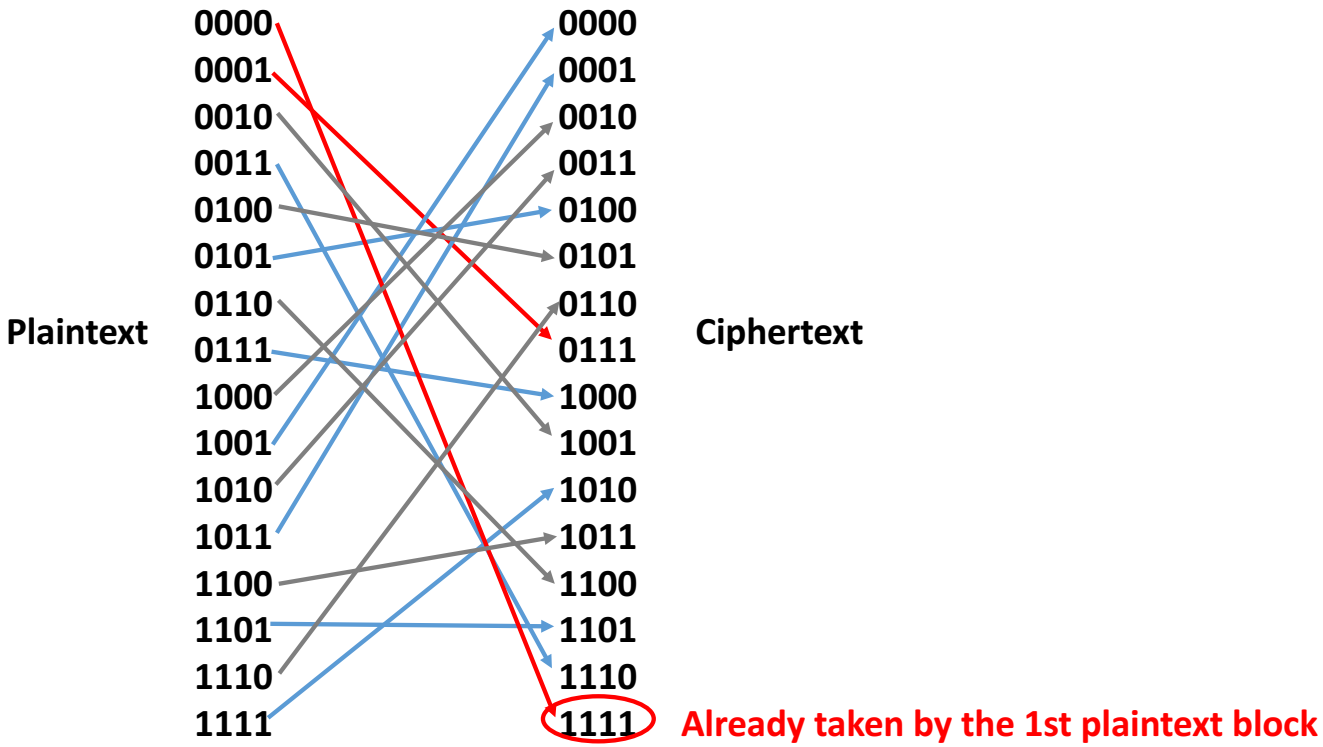
Ideal Block Cipher Example (n=4)



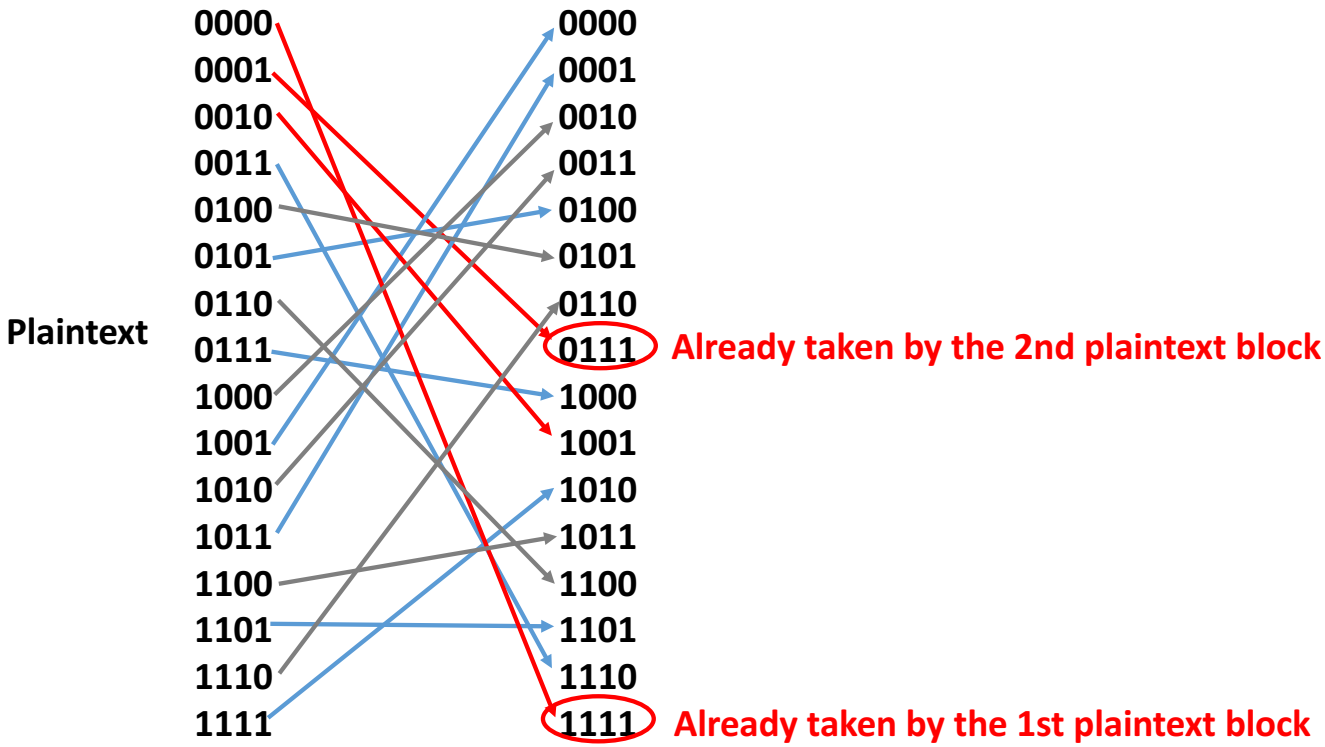
Ideal Block Cipher Example (n=4)



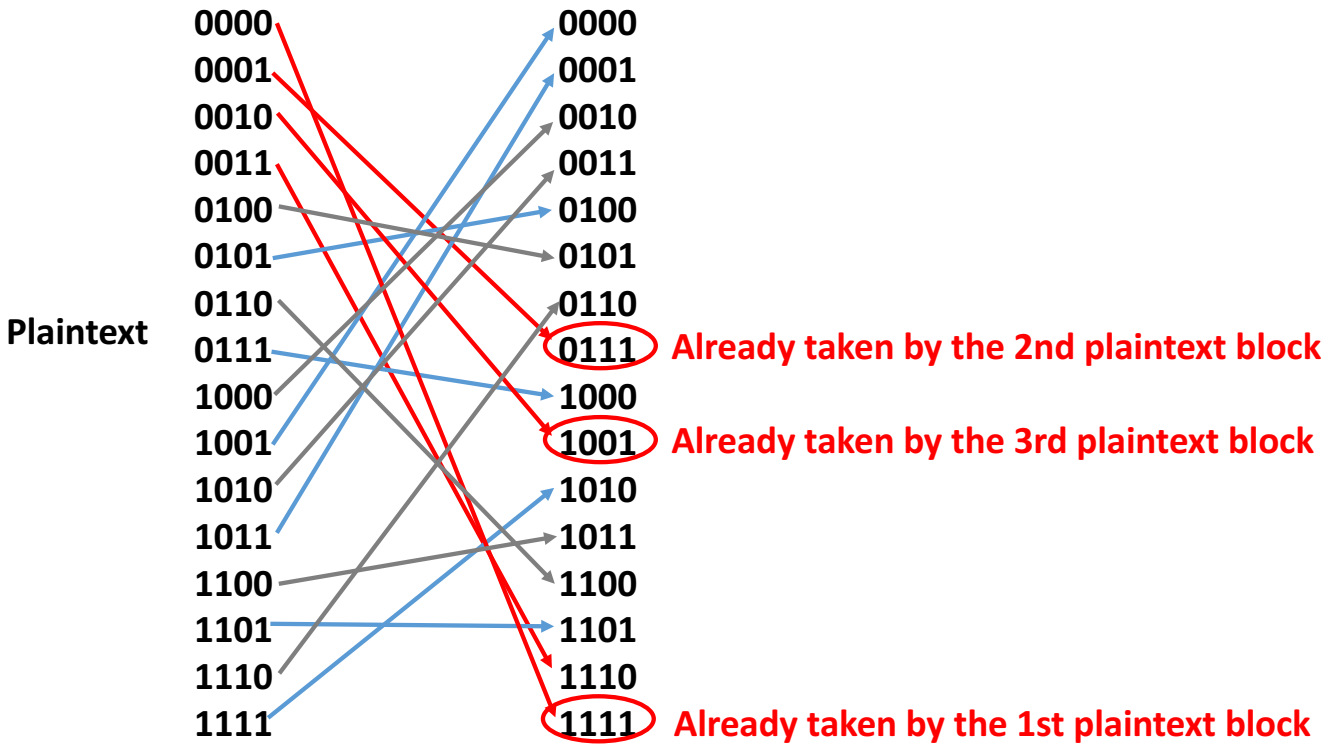
Ideal Block Cipher Example (n=4)



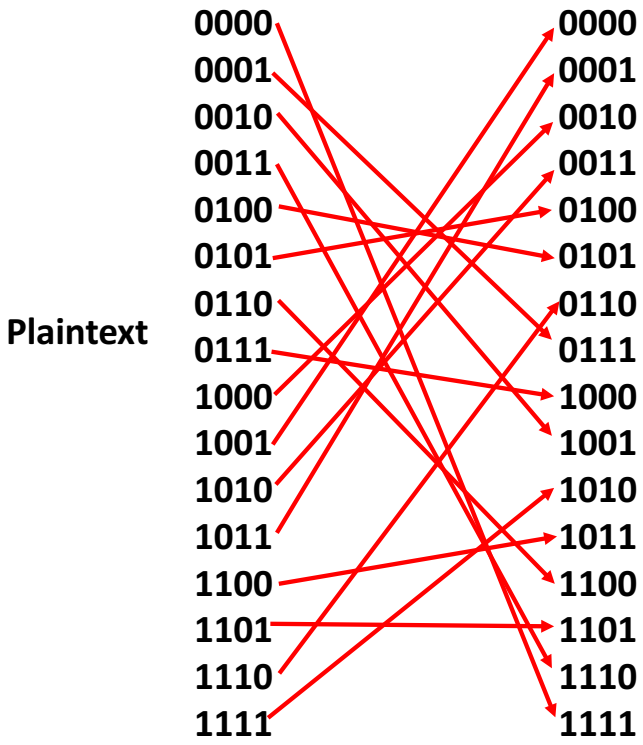
Ideal Block Cipher Example (n=4)



Ideal Block Cipher Example (n=4)

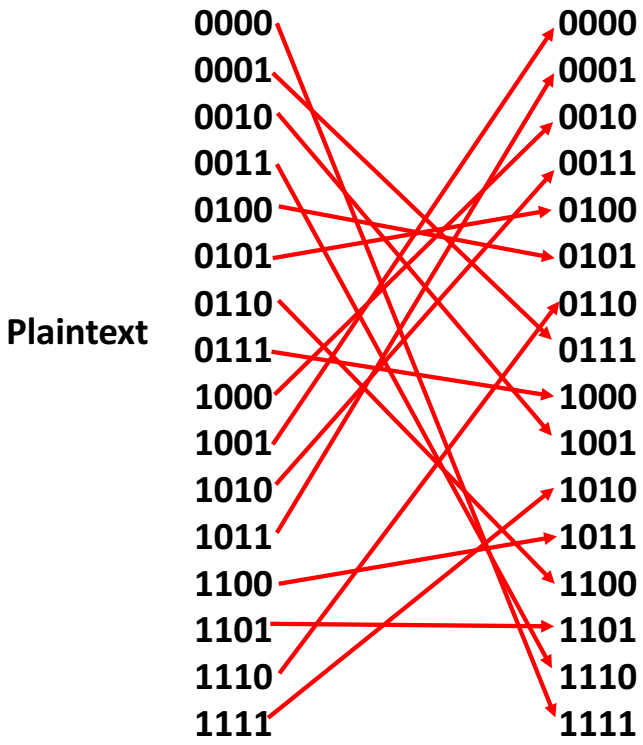


Ideal Block Cipher Example (n=4)



For block i, $0 \leq i \leq 15$,
$(16-i)$ block options

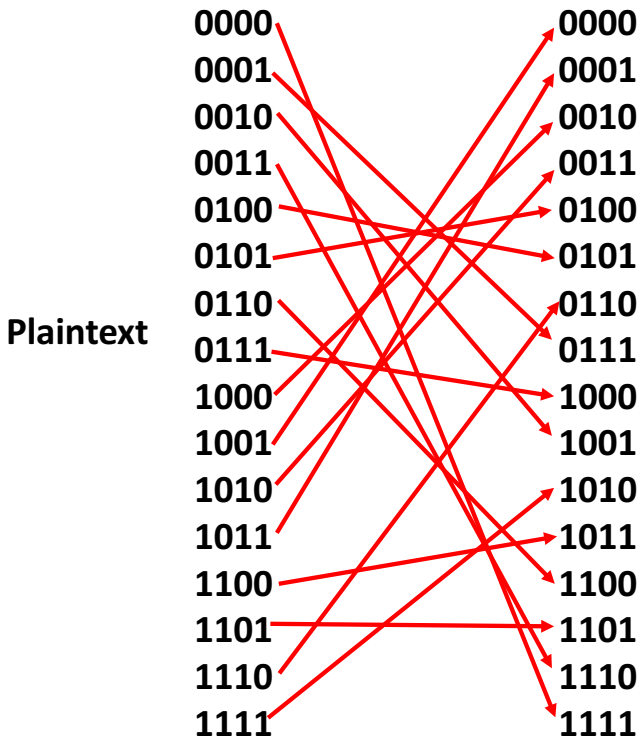
Ideal Block Cipher Example (n=4)



For block i, $0 \leq i \leq 15$,
$(16-i)$ block options

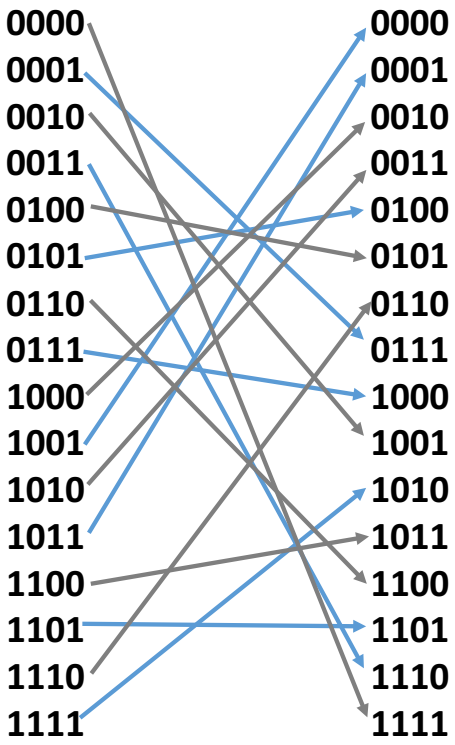
16! mappings/keys

Ideal Block Cipher Example (n=4)



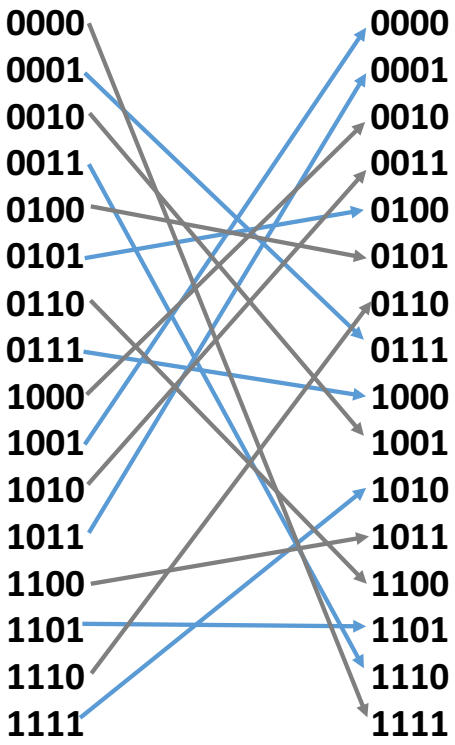
$2^n!$ possible keys

Ideal Block Cipher



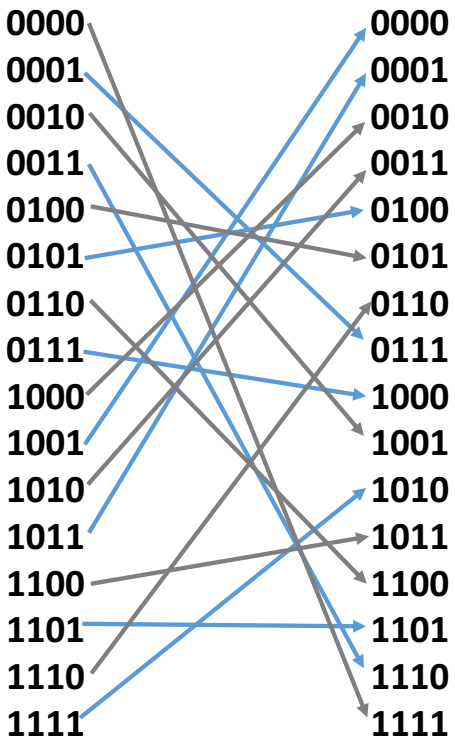
Plaintext	Ciphertext
0000	1111
0001	0111
0010	1001
0011	1110
0100	0101
0101	0100
0110	1100
0111	1000
1000	0010
1001	0000
1010	0011
1011	0001
1100	1011
1101	1101
1110	0110
1111	1010

Ideal Block Cipher



Plaintext	Ciphertext
0000	1111
0001	0111
0010	1001
0011	1110
0100	0101
0101	0100
0110	1100
0111	1000
1000	0010
1001	0000
1010	0011
1011	0001
1100	1011
1101	1101
1110	0110
1111	1010

Ideal Block Cipher



Plaintext	Ciphertext
0000	1111
0001	0111
0010	1001
0011	1110
0100	0101
0101	0100
0110	1100
0111	1000
1000	0010
1001	0000
1010	0011
1011	0001
1100	1011
1101	1101
1110	0110
1111	1010

Need $n \times 2^n$ bits for key

E.g., n=64 ➔ $2^{70}=10^{21}$ bits

Horst Feistel

An IBM researcher

Contributed to DES in 1970s

Wanted an approximation of ideal block cipher, built out of components that are easily realizable

Feistel Cipher (Feistel Network)

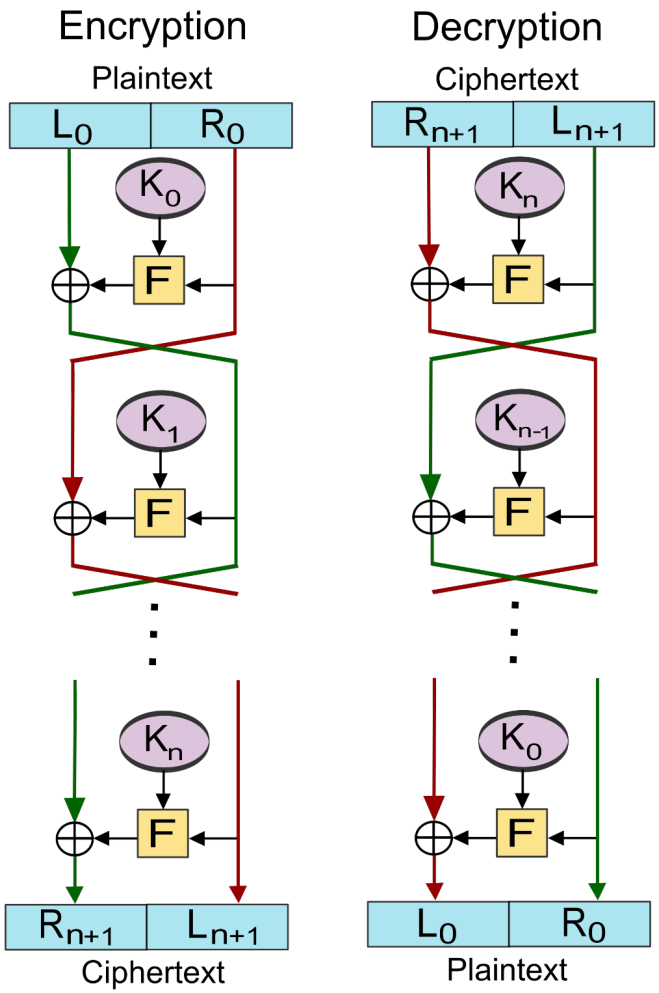
Product cipher

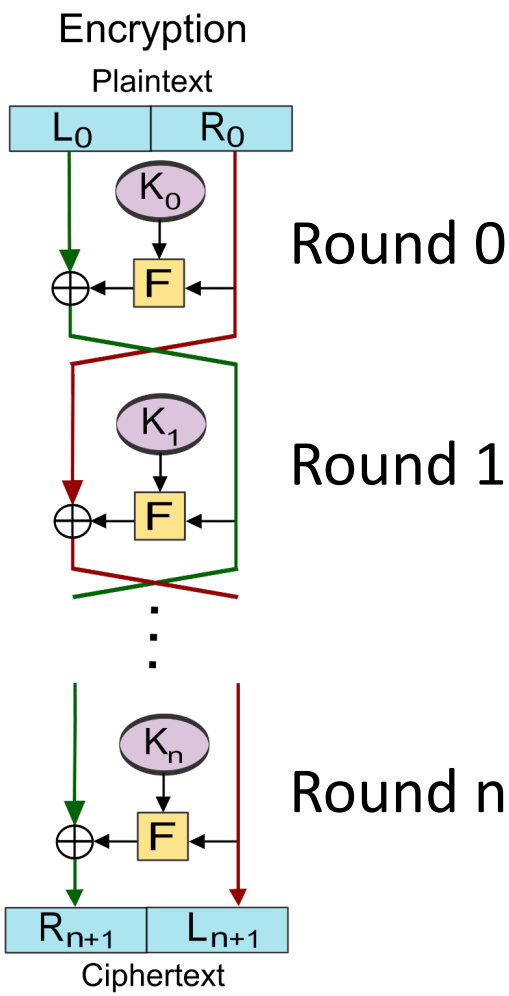
Structure for symmetric block ciphers

Key length k bits < $n \times 2^n$ bits

➔ 2^k possible keys < $2^n!$ mappings

Feistel Cipher

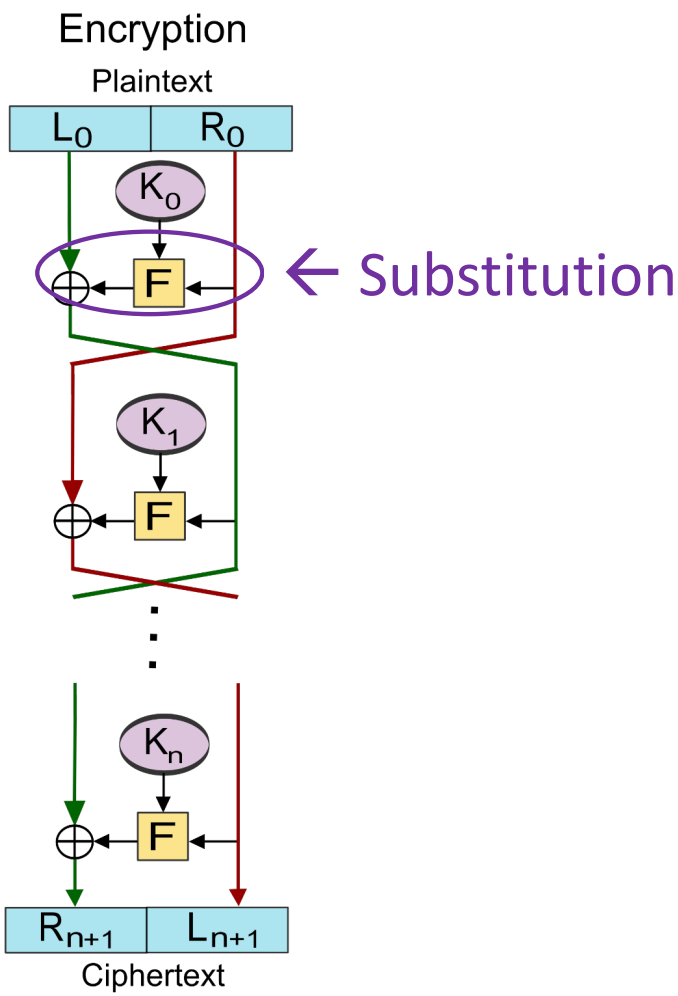




L_i : the left half of data after round i

R_i : the right half of data after round i

K_i : the subkey for round i

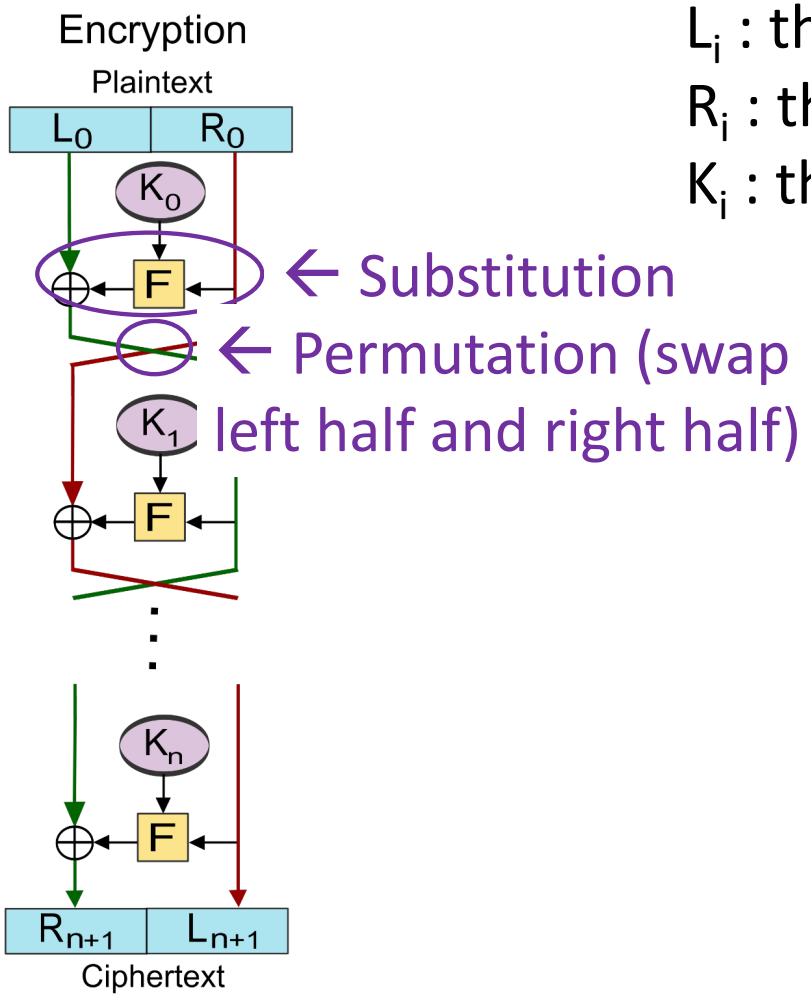


← Substitution

L_i : the left half of data after round i

R_i : the right half of data after round i

K_i : the subkey for round i



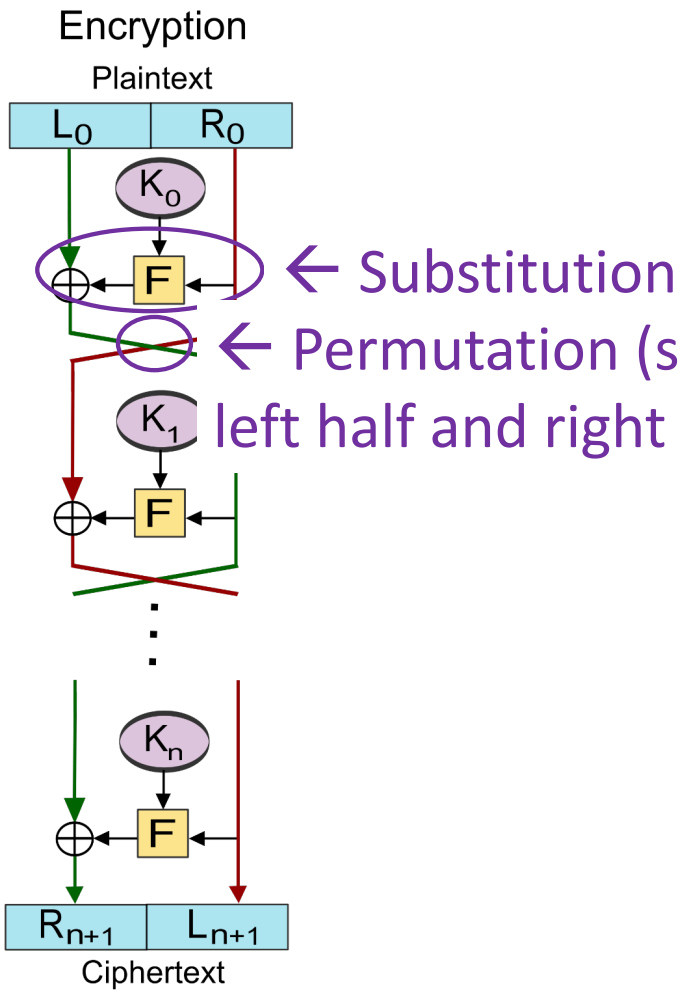
← Substitution

← Permutation (swap left half and right half)

L_i : the left half of data after round i

R_i : the right half of data after round i

K_i : the subkey for round i



← Substitution

← Permutation (swap left half and right half)

L_i : the left half of data after round i

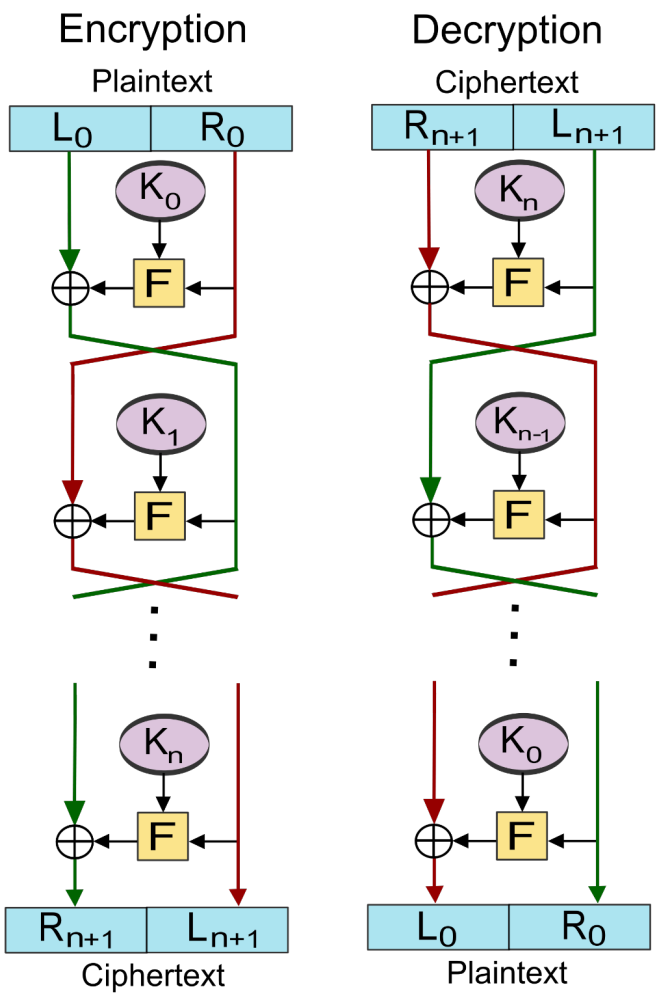
R_i : the right half of data after round i

K_i : the subkey for round i

In the i-th round:

$L_i = R_{i-1}$

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



L_i : the left half of data after round i

R_i : the right half of data after round i

K_i : the subkey for round i

In the i-th round:

$L_i = R_{i-1}$

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

F function does not need to be reversible (Decryption also uses F)

Feistel Cipher Design Parameters

Block size

Key size

Number of rounds

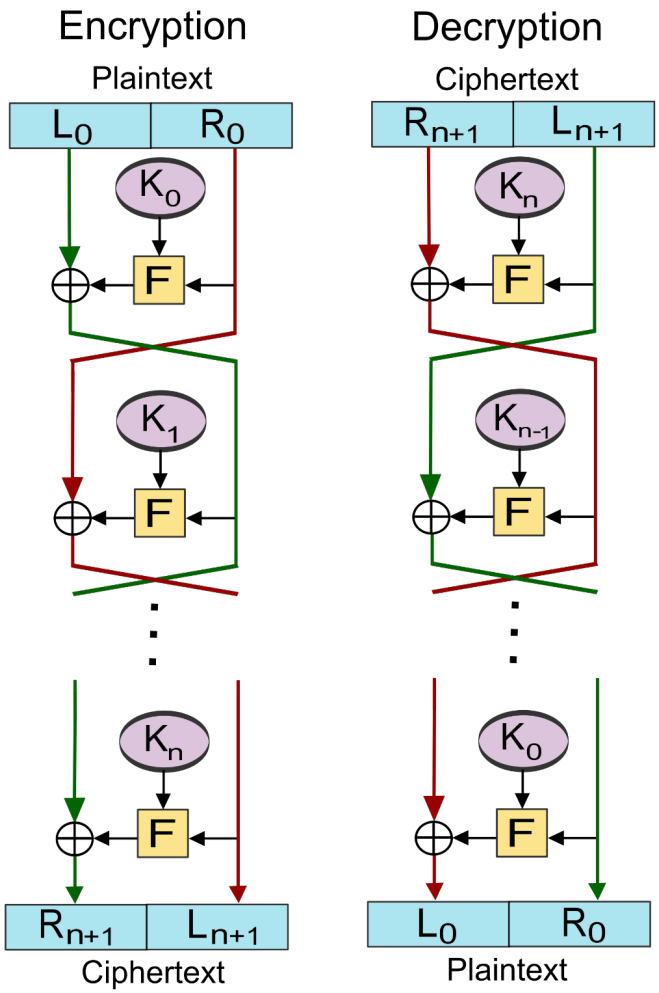
Subkey generation

Round function

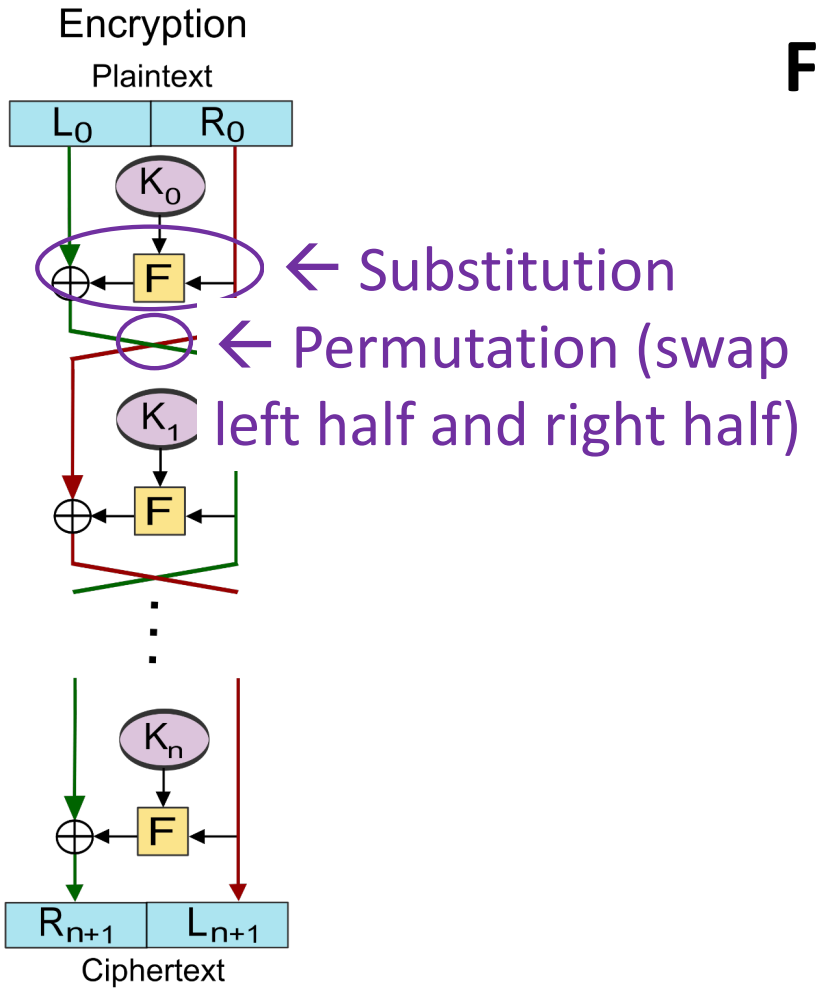
Data Encryption Standard (DES)

- Most widely used block cipher
- Based on Feistel Cipher
- In 1973, NBS (NIST) issued request for proposal for national cipher standard
- In 1977, adopted/published as DES
- Developed by IBM (Feistel) + NSA
- Considered broken but still widely used, e.g., legacy application

Feistel Cipher



Feistel Cipher



← Substitution

← Permutation (swap left half and right half)

Feistel Cipher Design Parameters

Block size

Key size

Number of rounds

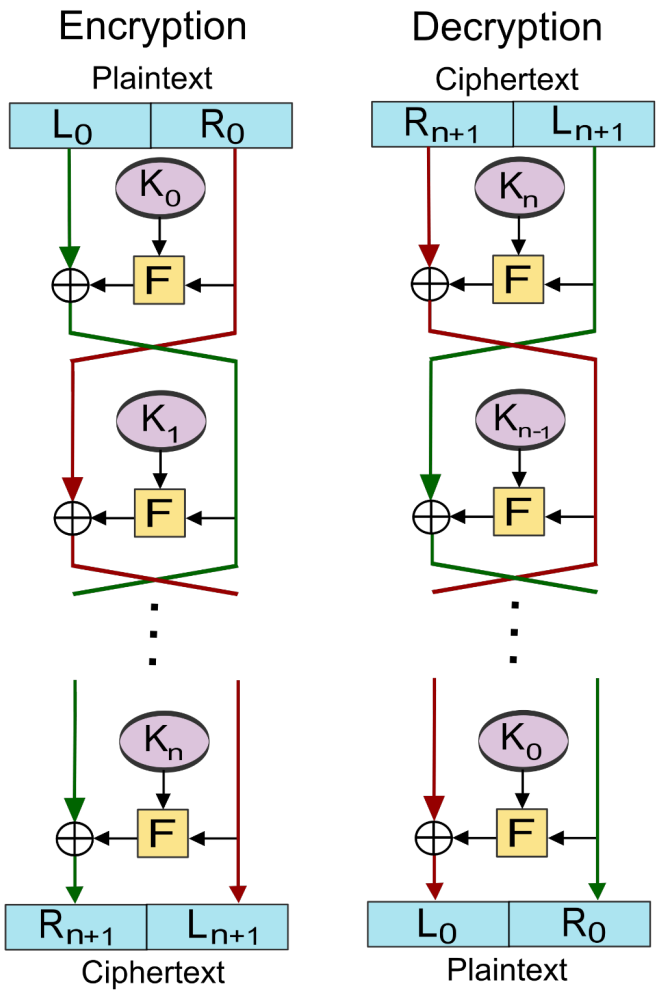
Subkey generation

Round function

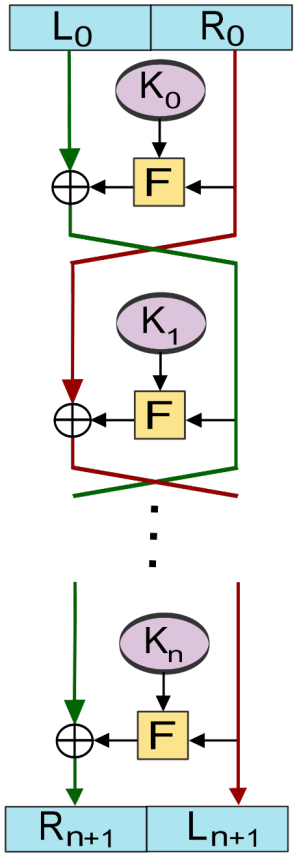
Feistel Cipher Design Parameters

	DES
Block size	64 bits
Key size	56 bits
Number of rounds	16 rounds
Subkey generation	(Later)
Round function	(Later)

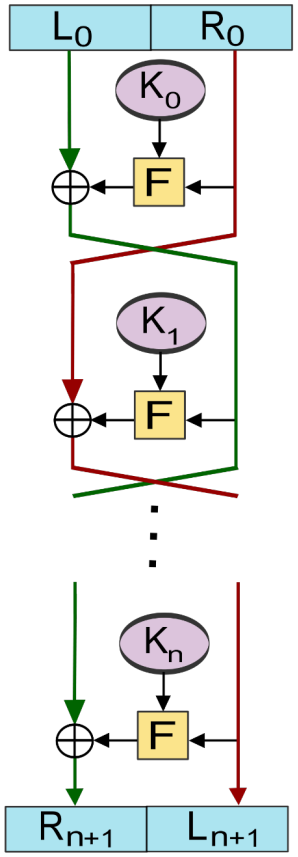
DES Overview



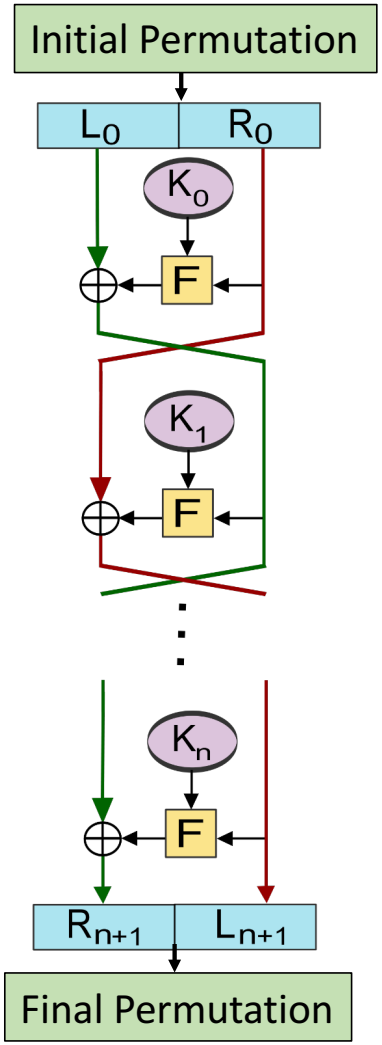
DES Overview



DES Overview



n=16

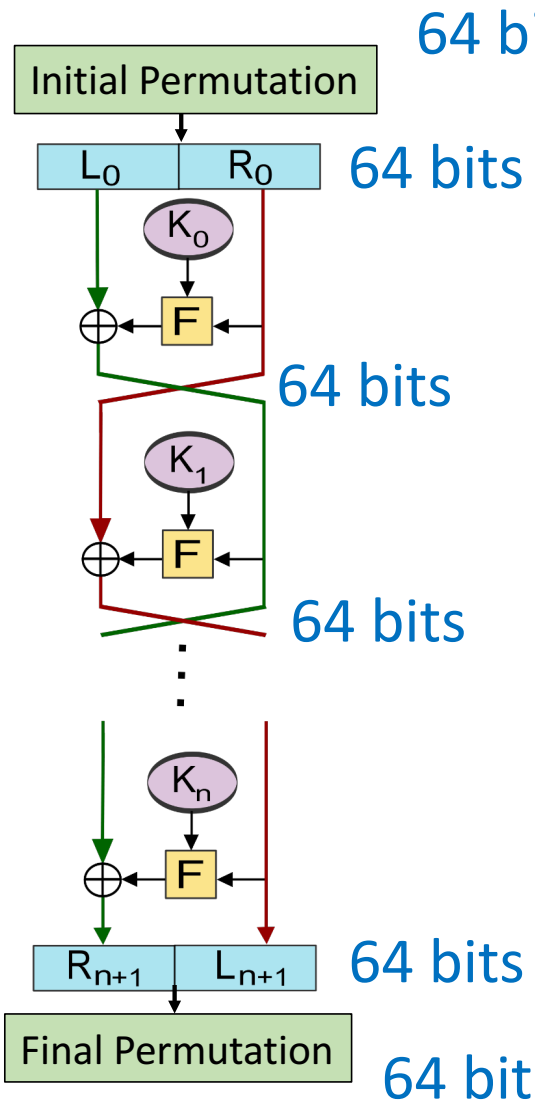


DES Overview

n=16

Add permutation blocks at the beginning and the end

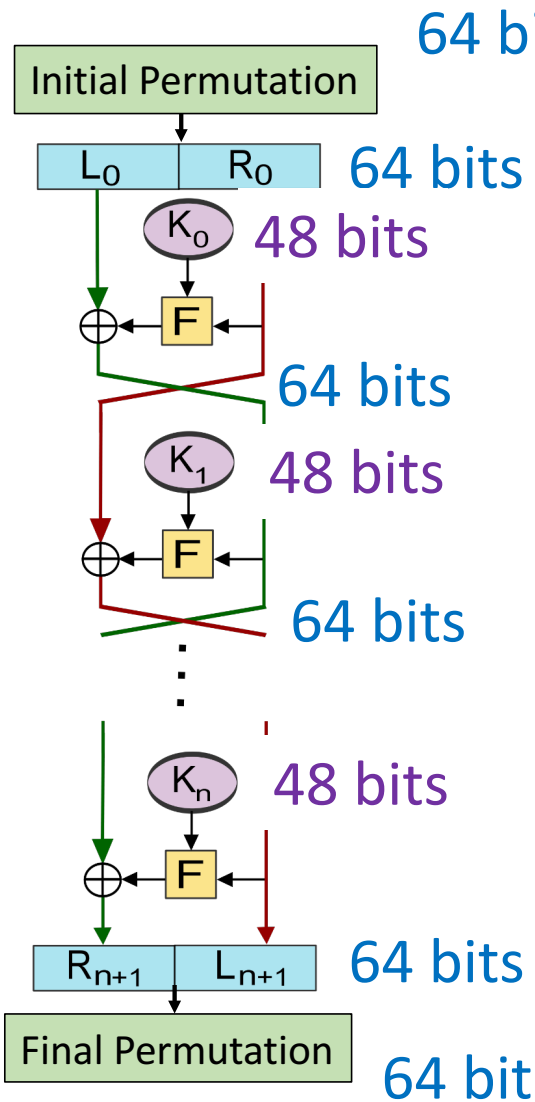
DES Overview



n=16

Add permutation blocks at the beginning and the end

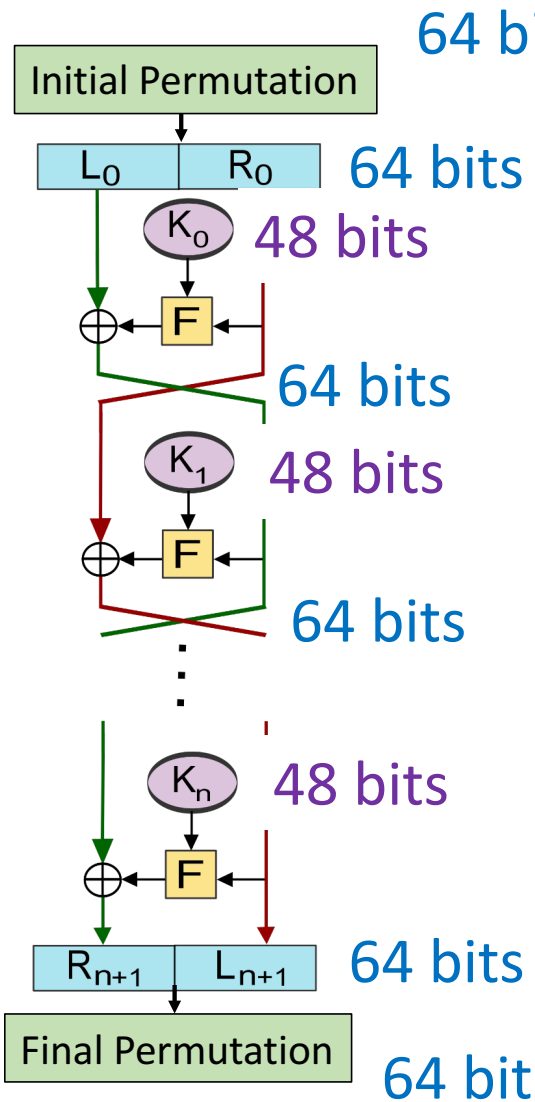
DES Overview



n=16

Add permutation blocks at the beginning and the end

DES Overview



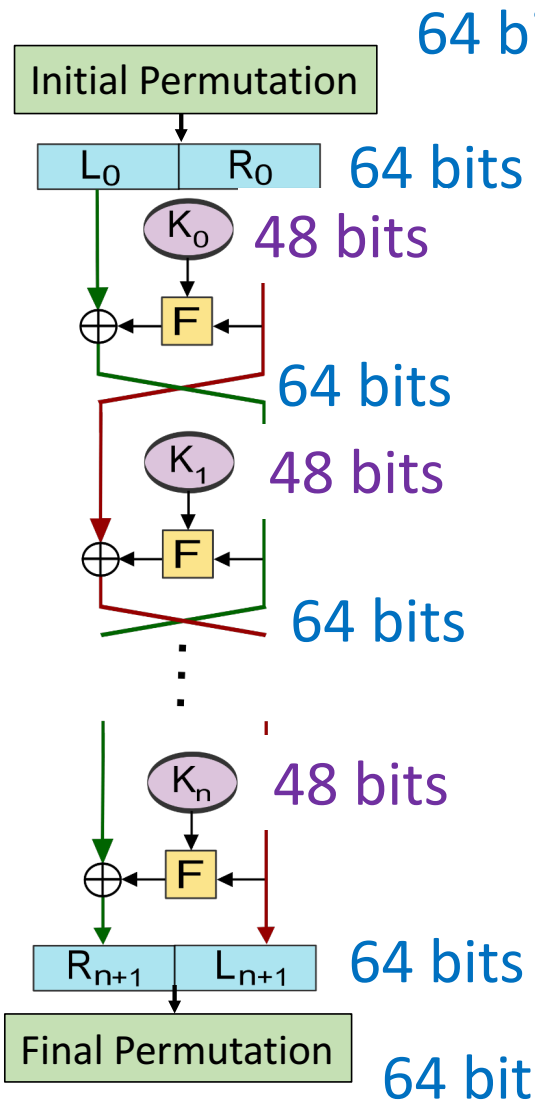
Subkey (K_i) generation:

56 bits → 16 · 48 bits

Round function F:

32 bits → 32 bits

DES Overview



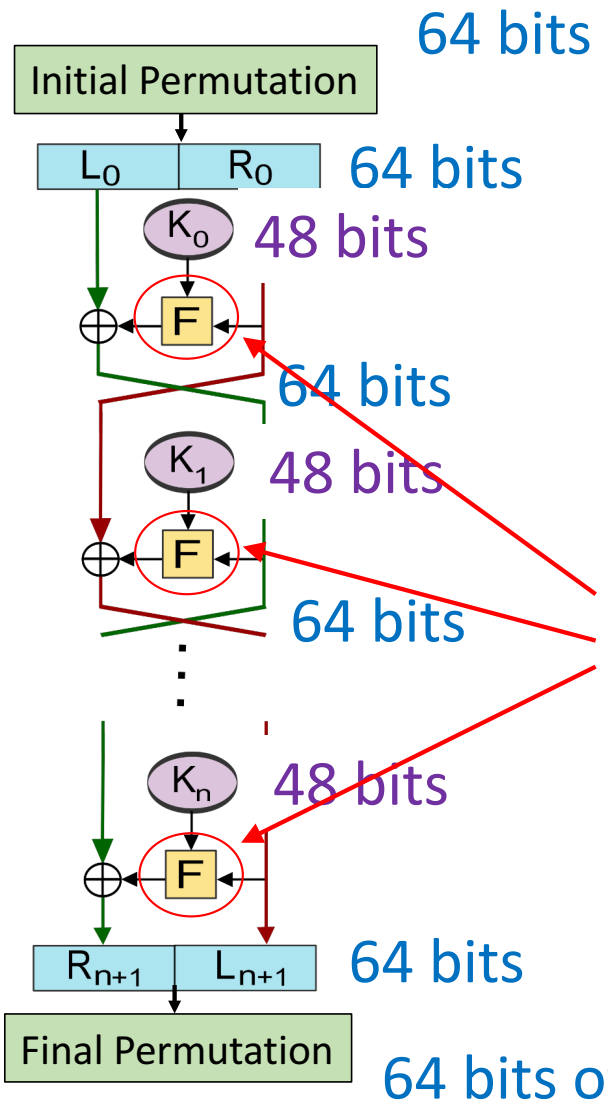
Subkey (K_i) generation:

56 bits → 16 · 48 bits

Round function F:

32 bits → 32 bits

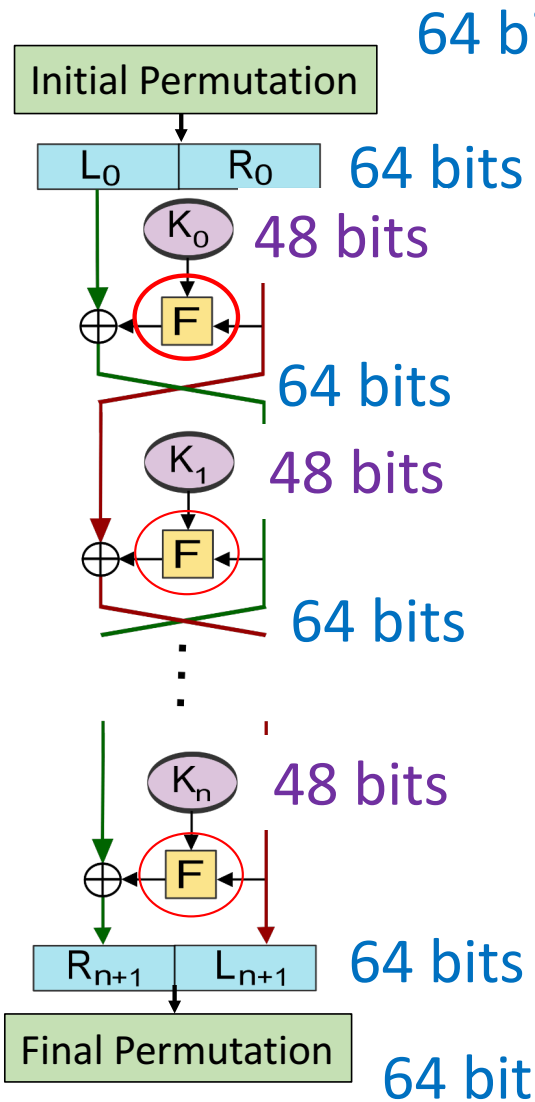
DES Overview



Round function F:

32 bits → 32 bits

DES Overview



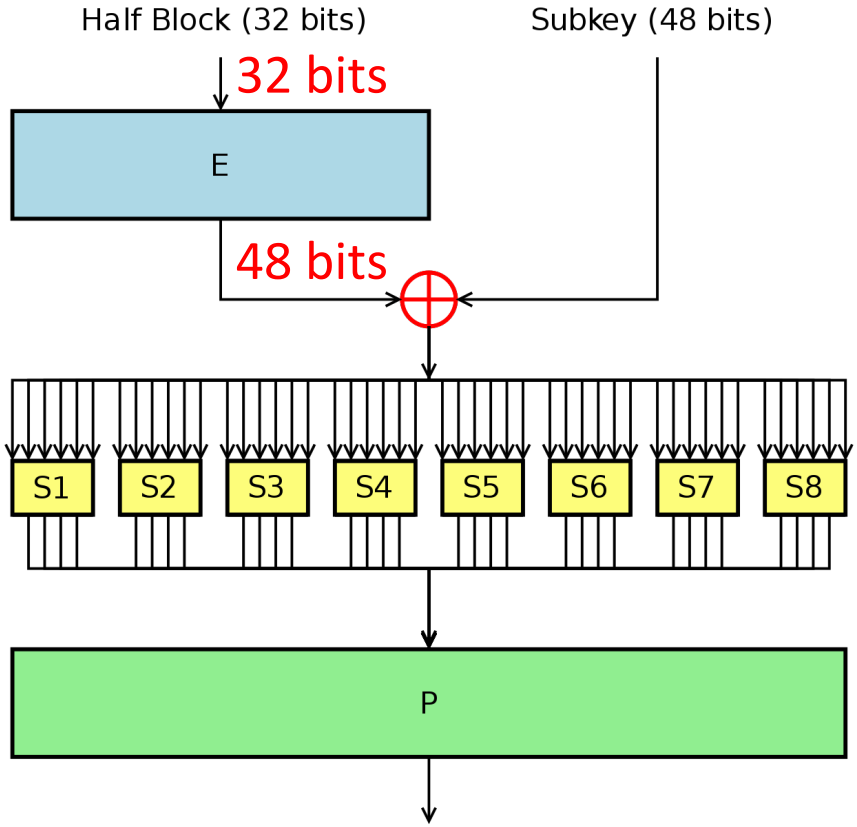
F actually has two inputs:

R_i (32 bits), K_i (48 bits)

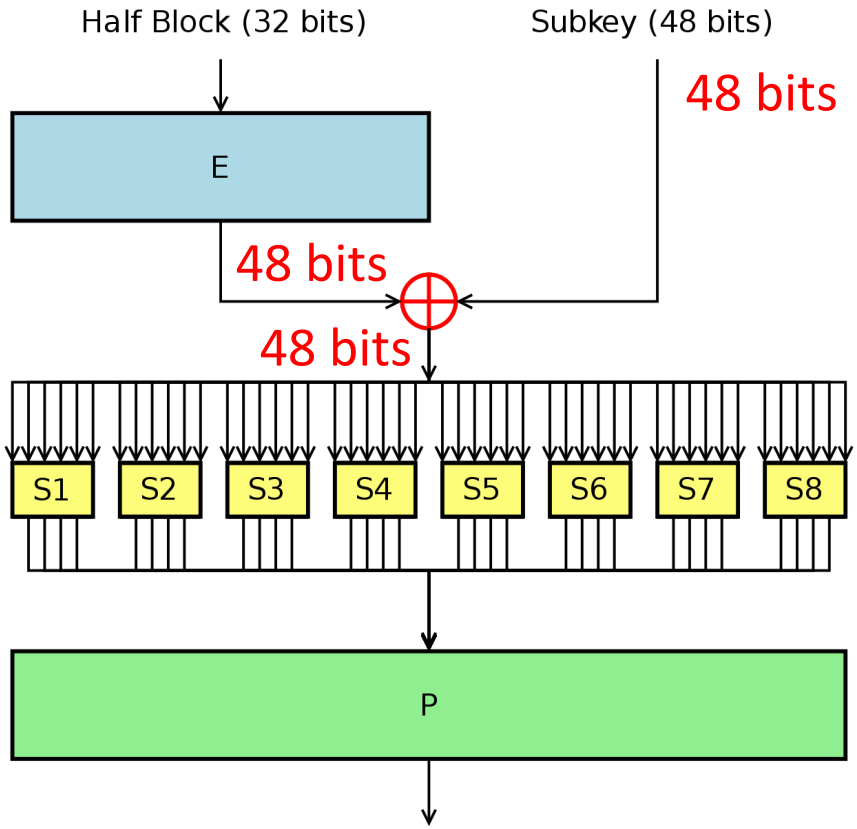
Round function F:

32 bits → 32 bits

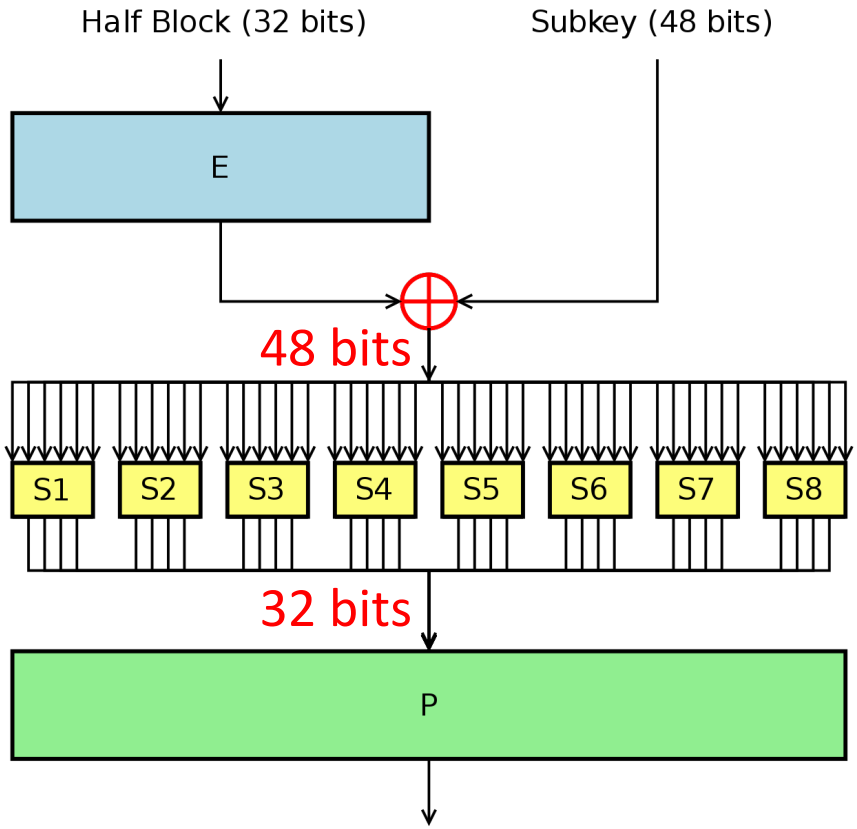
DES Round Function (F)



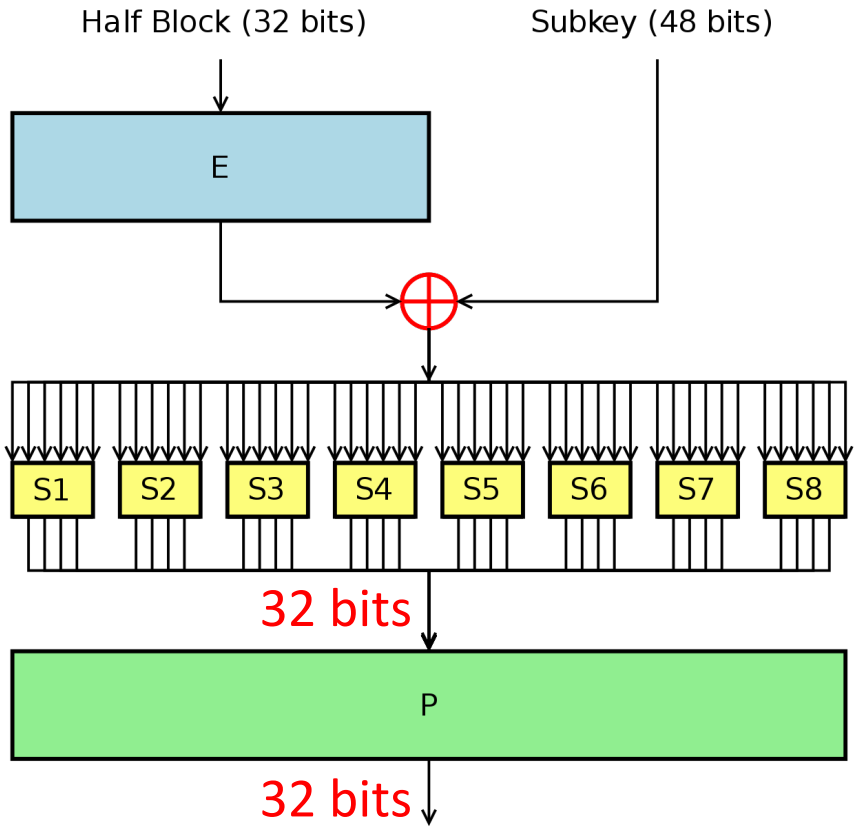
DES Round Function (F)



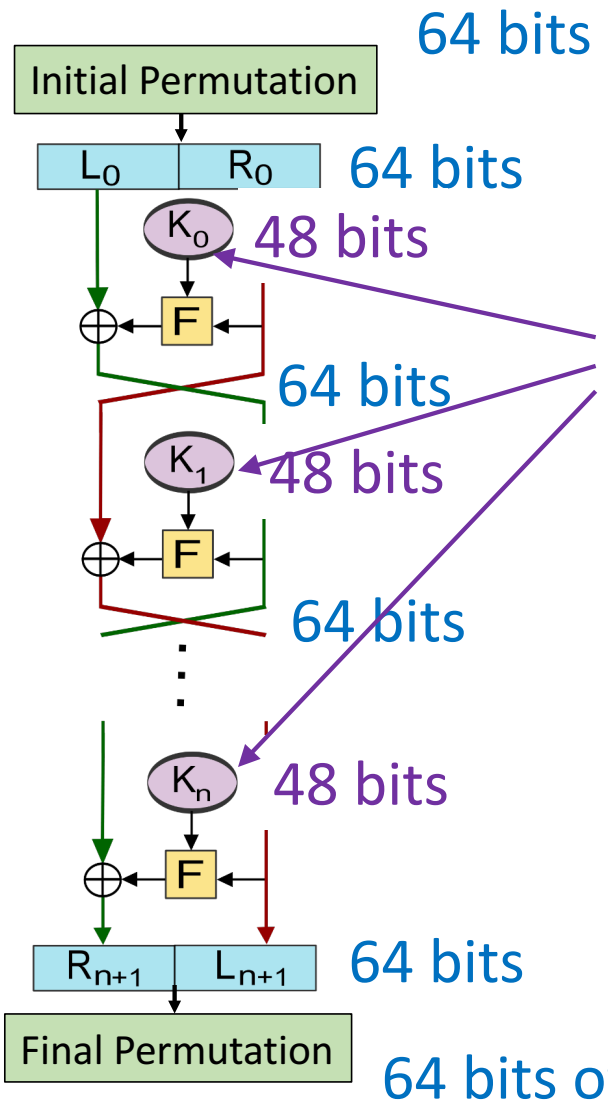
DES Round Function (F)



DES Round Function (F)



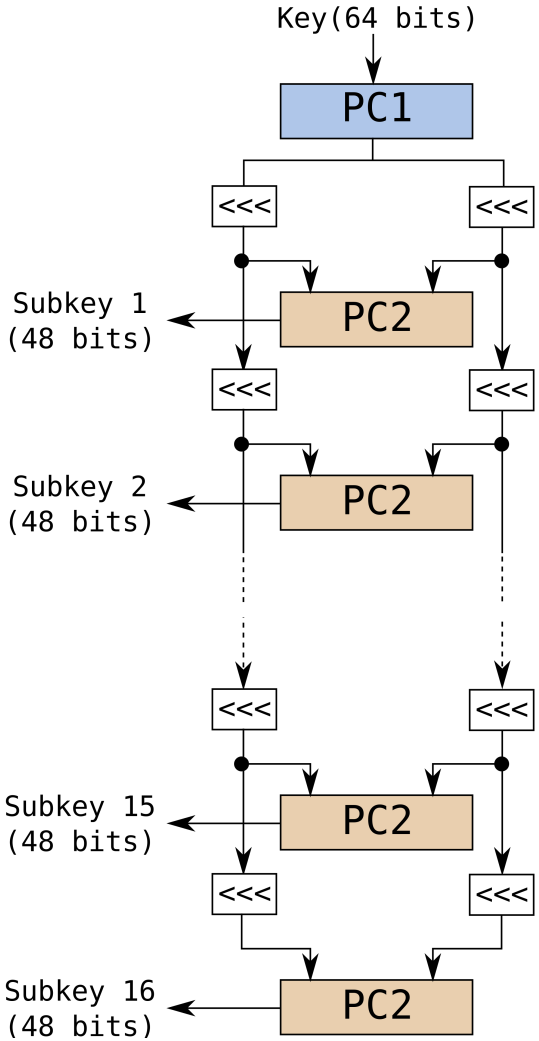
DES Overview

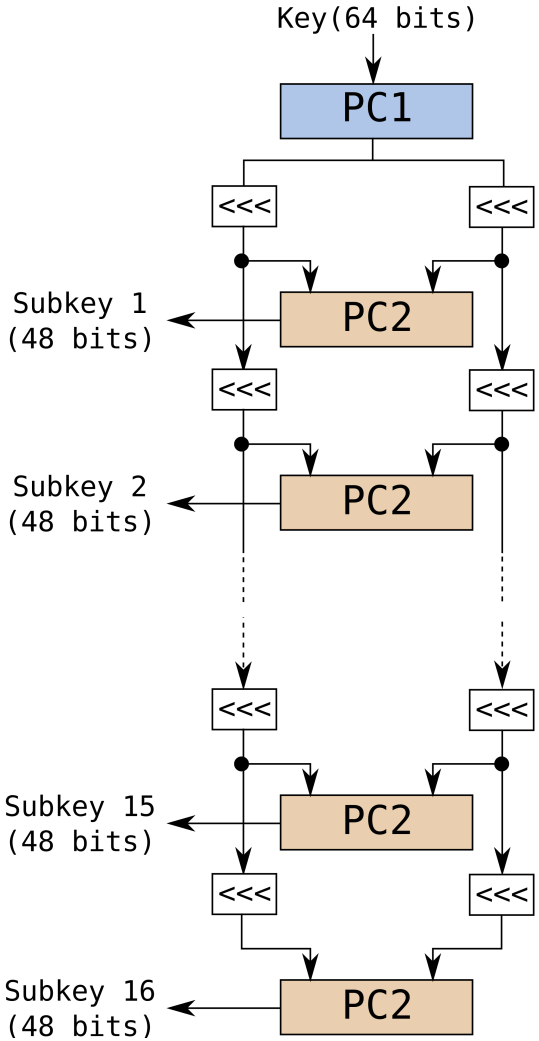


Subkey (K_i) generation:

56 bits → 16 · 48 bits

DES Subkey Generation (K_i)





DES Subkey Generation (K_i)

PC are Permuted Choice:

- **PC1**: 64 bits → 56 bits
- **PC2**: 56 bits → 48 bits

<<< is left-circular shift (LCS)

Every round, LCS by 1 or 2 bits, depending on the round

DES Strength

Avalanche Effect

Change of one plaintext bit or
one key bit changes about half
the ciphertext bits

DES Brute Force

56-bit key ➔ Attacker effort $O(2^{55})$

Require recognizing the correct plaintext

Demonstration of Brute Force attacks:
In 1997: a few months to find the key
In 1998: a few days
In 1999: 22 hours

DES Security

Brute Force attacks in practice

Cryptanalytic attacks that can
further reduce the complexity

Timing attacks on computation