INTRODUCTION TO THE SYMMETRICAL CRYPTOGRAPHY

Assoc. Prof. Ahmet Koltuksuz, Ph.D.
<ahmet.koltuksuz@yasar.edu.tr>

Yasar University

College of Engineering
Department of Computer Engineering
İzmir, Turkey

" It may be roundly asserted that human ingenuity cannot concoct a cipher which human ingenuity cannot resolve. "

POE, Edgar Allan
The Gold Bug
1840

Was POE right ?

CRYPTOLOGY

In Greek
kryptos = hidden,
graphein = to write

- CRYPTOGRAPHY

  encryption, decryption

- CRYPTANALYSIS

A CRYPTOGRAPHIC SYSTEM

is a family ***T*** of transformations on **P**laintext and **C**iphertext.

family member index

key (k)

encryption	decryption
$E(P) = C$	$D(C) = P$
$E_k(P) = C$	$D_k(C) = P$

Now; $E_k = D_k$ ⇒ **Symmetrical**

$E_k \neq D_k$ ⇒ **Asymmetrical**

"Public Key Cryptosystems"

Symmetrical

- Substitution
- Transposition
- Block - Product
- Stream

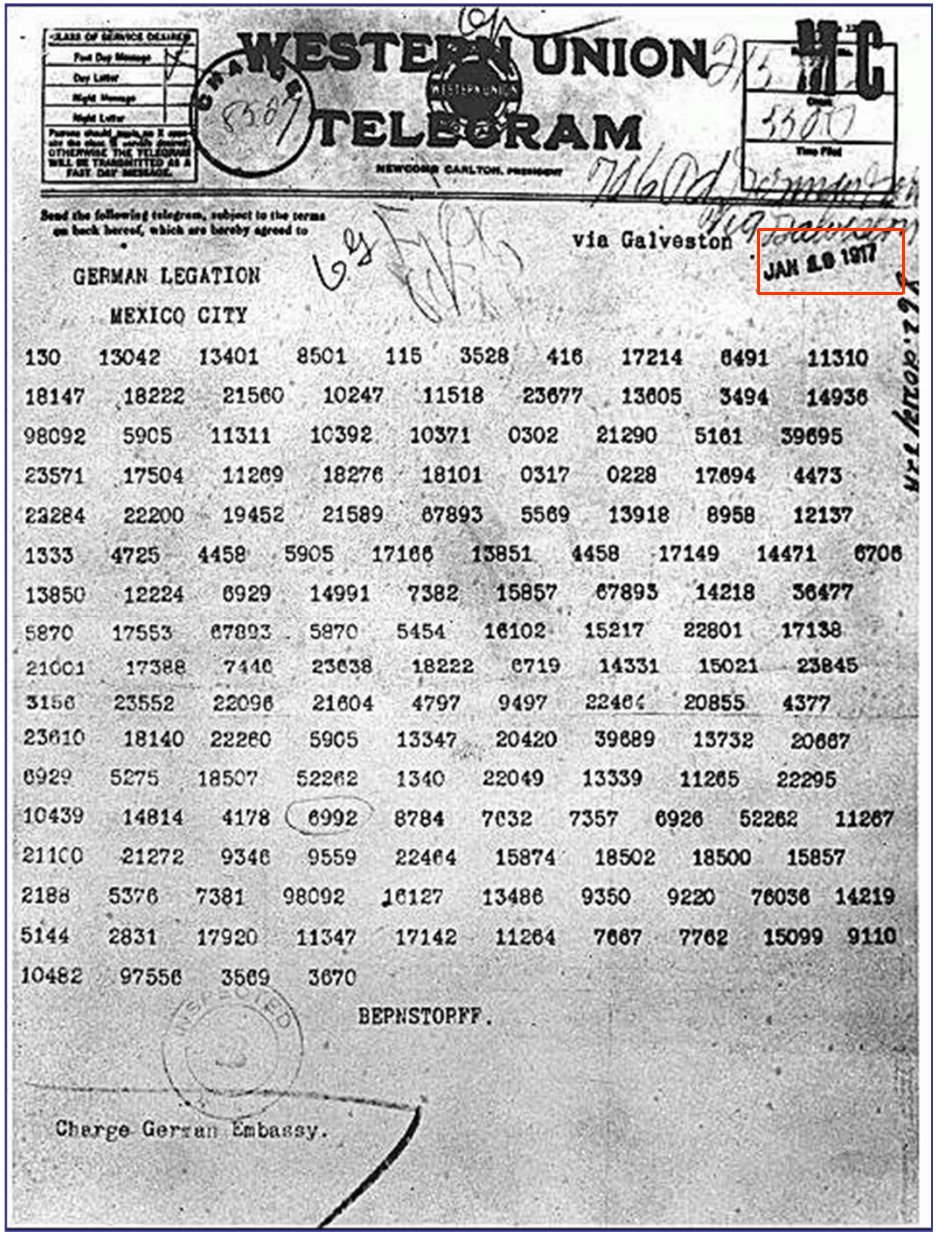
Asymmetrical

- **DLP**
  - **Diffie-Hellman**
  - **ElGamal-DSA**
- **Integer Factorization**
  - **RSA**
- **Elliptic Curves**
  - **ECC**
- **Lattice**
  - **NTRU**

- Monoalphabetical
- Polyalphabetical
- One-Time-Pad
- Simple
- Nihilist
- DES
- AES
- DES-AES variants

- Ancient Greeks and Romans
  - 1900 B.C. Tomb inscriptions.
  - 475 B.C. Spartans - Scytale Cipher.
  - 60 B.C. Julius Caesar - Substitution Cipher.
- Middle Ages
  - 1378 - 1417 Gabriele de Lavinde of Parma, First manual on ciphers !
- Renaissance
  - 1518 Johannes Trithemius: " Polygraphia ", first printed work!
- 20th. Century
  - 1917 Zimmermann Telegramme.

The Zimmermann Telegramme, 1917.



WESTERN UNION TELEGRAM

NEWCOMB CARLTON, PRESIDENT

Send the following telegram, subject to the terms on back hereof, which are hereby agreed to

via Galveston

JAN 19 1917

GERMAN LEGATION

MEXICO CITY

130 13042 13401 8501 115 3528 416 17214 6491 11310
18147 18222 21560 10247 11518 23677 13605 3494 14936
98092 5905 11311 10392 10371 0302 21290 5161 39695
23571 17504 11269 18276 18101 0317 0228 17694 4473
23284 22200 19452 21589 67893 5569 13918 8958 12137
1333 4725 4458 5905 17166 13851 4458 17149 14471 6706
13850 12224 6929 14991 7382 15857 67893 14218 36477
5870 17553 67893 5870 5454 16102 15217 22801 17138
21001 17388 7446 23638 18222 6719 14331 15021 23845
3156 23552 22096 21604 4797 9497 22464 20855 4377
23610 18140 22260 5905 13347 20420 39689 13732 20667
6929 5275 18507 52262 1340 22049 13339 11265 22295
10439 14814 4178 6992 8784 7632 7357 6926 52262 11267
21100 21272 9346 9559 22464 15874 18502 18500 15857
2188 5376 7381 98092 16127 13486 9350 9220 76036 14219
5144 2831 17920 11347 17142 11264 7667 7762 15099 9110
10482 97556 3569 3670

BEPNSTORFF.

Charge German Embassy.

TELEGRAM RECEIVED.

FROM 2nd from London # 5747.

"We intend to begin on the first of February unrestricted submarine warfare. We shall endeavor in spite of this to keep the United States of America neutral. In the event of this not succeeding, we make Mexico a proposal of alliance on the following basis: make war together, make peace together, generous financial support and an understanding on our part that Mexico is to reconquer the lost territory in Texas, New Mexico, and Arizona. The settlement in detail is left to you. You will inform the President of the above most secretly as soon as the outbreak of war with the United States of America is certain and add the suggestion that he should, on his own initiative, invite Japan to immediate adherence and at the same time mediate between Japan and ourselves. Please call the President's attention to the fact that the ruthless employment of our submarines now offers the prospect of compelling England in a few months to make peace." Signed, ZIMMERMANN.

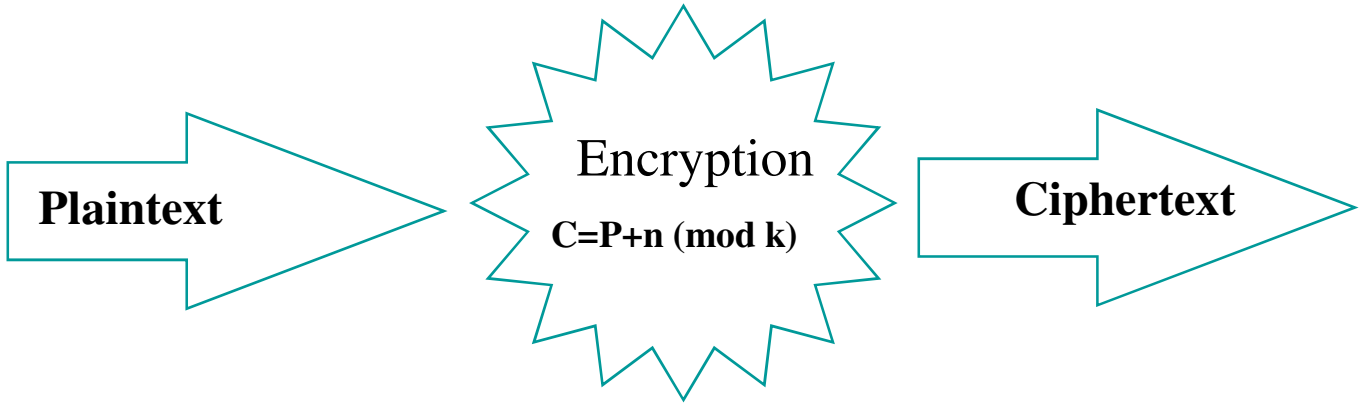
- 20th. Century
  - 1926 Vernam, " one - time - pad ".
  - 1939-1945 2nd. World War : " Enigma " - " Purple "
  - 1945-1995 Cold War Years, NSA: "VENONA"

- Contemporary Ciphers: Early Years
  - 1971 IBM announces Lucifer, A Block cipher.
  - 1975 IBM offers Lucifer as a standard.
  - 1976 Diffie & Hellman, Public Key concept.
  - 1977 Lucifer gets approved by NIST (formerly NBS) as Data Encryption Standard (DES), a Block Cipher.
  - 1978 Rivest-Shamir-Adleman (RSA), Public Key Cryptosystem.
- Contemporary Ciphers: AES
  - DES is no longer a standard.
  - DoC, National Institute of Standards and Technology (NIST) announced the need for a new standard in January 2, 1997.
  - New standard is the Advanced Encryption Standard – AES,2001.

Symmetrical

Substitution

Ceasar Cipher, 60 B.C.



n=5, k=26 (English)

S-M-I-T-H ⇨ ⇨ **X-R-N-Y-M**

Symmetrical

Substitution

Monoalphabetical Substitution

- MCRKHAT LNBDEFGI J ZPOYSÖUŞÇİĞÜV

- ABCÇDE FGĞHI İ JKLMNOÖPRSŞTUÜVYZ

- Plaintext : TÜRKİYE

- Ciphertext : ŞİSGEÜA

Symmetrical

Substitution

Polylphabetical Substitution: Vigenere Table

	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	
A	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	**00**
B	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	**01**
C	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	**02**
Ç	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	**03**
D	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	**04**
E	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	**05**
F	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	**06**
G	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	**07**
Ğ	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	**08**
H	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	**09**
I	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	**10**
İ	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	**11**
J	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	**12**
K	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	**13**
L	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	**14**
M	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	**15**
N	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	**16**
O	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	**17**
Ö	Ö	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	**18**
P	P	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	**19**
R	R	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	**20**
S	S	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	I	J	K	L	M	N	O	Ö	P	R	**21**
Ş	Ş	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	**22**
T	T	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	**23**
U	U	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	**24**
Ü	Ü	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	**25**
V	V	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	**26**
Y	Y	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	**27**
Z	Z	A	B	C	Ç	D	E	F	G	Ğ	H	I	İ	J	K	L	M	N	O	Ö	P	R	S	Ş	T	U	Ü	V	Y	**28**

Plaintext: TÜRKİYE
T=00
Ü=01
R=02
K=03
İ=04
Y=05
E=06

Ciphertext: TVŞNM
Çİ

Symmetrical

Substitution

One Time Pad, Vernam Cipher, 1926

Step #1: Digitize the Alphabet

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26

Step #2: Encryption

Plaintext	:	**H**	**E**	**L**	**L**	**O**
(1) Digitalization	:	**8**	**5**	**12**	**12**	**15**
(2) Random Numbers	:	**12**	**48**	**28**	**32**	**80**
Summation of 1 & 2	:	**20**	**53**	**40**	**44**	**95**
Modulus 26	:	**20**	**1**	**14**	**18**	**17**
Ciphertext	:	**T**	**A**	**N**	**R**	**Q**

Symmetrical

Substitution

One Time Pad, Vernam Cipher, 1926

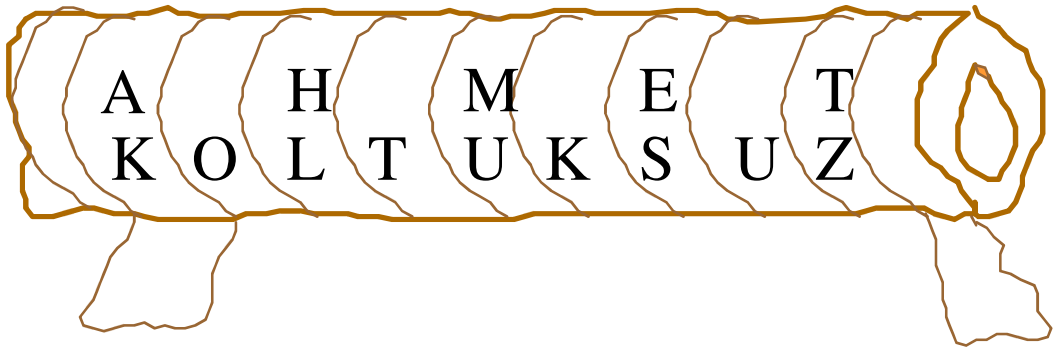
- The only mathematically proven unbreakable cipher. Proven by **Shannon**.
- With the conditions of :The length of the key should be equal to that of the length of the message.
- Each key should only be used once.
- Thus very popular with the intelligence agencies.

Symmetrical

Transposition:
Permutation

475 B.C. Spartans - Scytale Cipher

Plaintext : AHMET KOLTUKSUZ



Ciphertext : AKOHLTMUEKSTUZ

Symmetrical

Transposition: Permutation

Simple Permutation

Plaintext : KRİPTOGRAFİ

Key : G A L O İ S

Index : 1 2 3 4 5 6

Permutation : 4 3 6 2 5 1

Encryption : K R İ P T O G R A F İ

1 2 3 4 5 6 1 2 3 4 5

4 3 6 2 5 1 4 3 6 2 5

Ciphertext : P İ O R T K F A İ R İ

AN OLD BUT VERY GOOD STANDARD
Data Encryption Standard (DES)

Assoc. Prof. Ahmet Koltuksuz, Ph.D.

<ahmet.koltuksuz@yasar.edu.tr>

Yasar University

College of Engineering

Department of Computer Engineering

İzmir, Turkey

Agenda

- History
- Design Criteria & Block Cipher Principles
- Feistel Cipher Structure
- General Structure of DES
- DES Algorithm
  - Initial & Final Permutation, Key Generation, DES Rounds
  - DES Function – XOR, S-BOXes
- DES Decryption, Avalanche Effect & DES
- DES Modes, Weaknesses - Strength & Security of DES
- Cryptanalysis of DES: Differential & Linear Cryptanalysis
- Multiple Encryption – 3DES

History

- IBM developed Lucifer cipher for Llyods, Lloyds & Llyods
  - by a team led by Horst Feistel in late 60's.
  - used 64-bit data blocks with 128-bit key.
- then redeveloped as a commercial cipher with input from NSA and others.
- in 1973 NBS issued request for proposals for a national cipher standard.
- IBM submitted their revised Lucifer which was eventually accepted as the DES.

History

- Used in most EFT and EFTPOS from banking industry
  - It was reconfirmed as a standard for 5 years twice
  - Currently 3DES is recommended
- DES became a federal standard in November 76
  - adopted in 1977 by National Bureau of Standards - NBS (now NIST – National Institutes of Standards and Technologies) As Federal Information Processing Standard FIPS-PUB 46
- Outdated by AES (Advanced Encryption Standard) in 2000

Design Criteria

- The standard is public, the design criteria was classified
- One of the biggest controversies had been the key size (56 bits)
  - W. Diffie, M Hellman "Exhaustive Cryptanalysis of the NBS Data Encryption Standard" IEEE Computer 10(6), June 1977, pp74-84
  - M. Hellman "DES will be totally insecure within ten years" IEEE Spectrum 16(7), Jul 1979, pp 31-41

Block Cipher Principles

- A block cipher operates on a plaintext block of n bits to produce a ciphertext block of n bits,
- most symmetric block ciphers are based on a **Feistel Cipher Structure.**
- needed since must be able to **decrypt** ciphertext to recover messages efficiently
- block ciphers look like an extremely large substitution

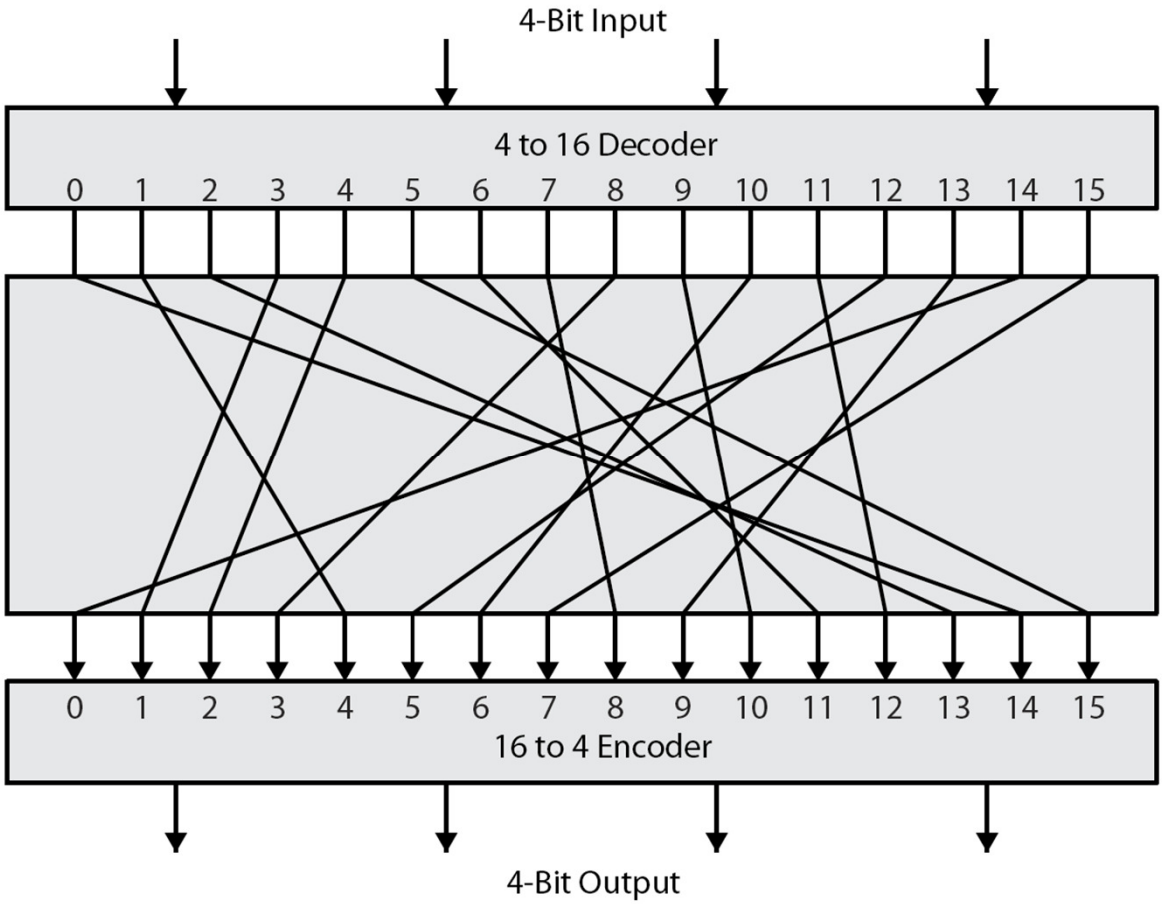
Block Cipher Principles

- One would need a table of 2^{64} entries for a 64-bit block
  - In general, for an *n*-bit general substitution block cipher, the size of the key is $n \times 2^n$.
  - For a 64-bit block, which is a desirable length to thwart statistical attacks, the key size is $64 \times 2^{64} = 2^{70} = 10^{21}$ bits.
- instead one creates from smaller building blocks by using an idea of a product cipher.

Ideal Block Cipher

A 4-bit input produces one of 16 possible input states, which is mapped by the substitution cipher into a unique one of 16 possible output states, each of which is represented by 4 ciphertext bits.

It illustrates a tiny 4-bit substitution to show that each possible input can be arbitrarily mapped to any output - which is why its complexity grows so rapidly.



The encryption and decryption mappings can be defined by a tabulation, as shown above.

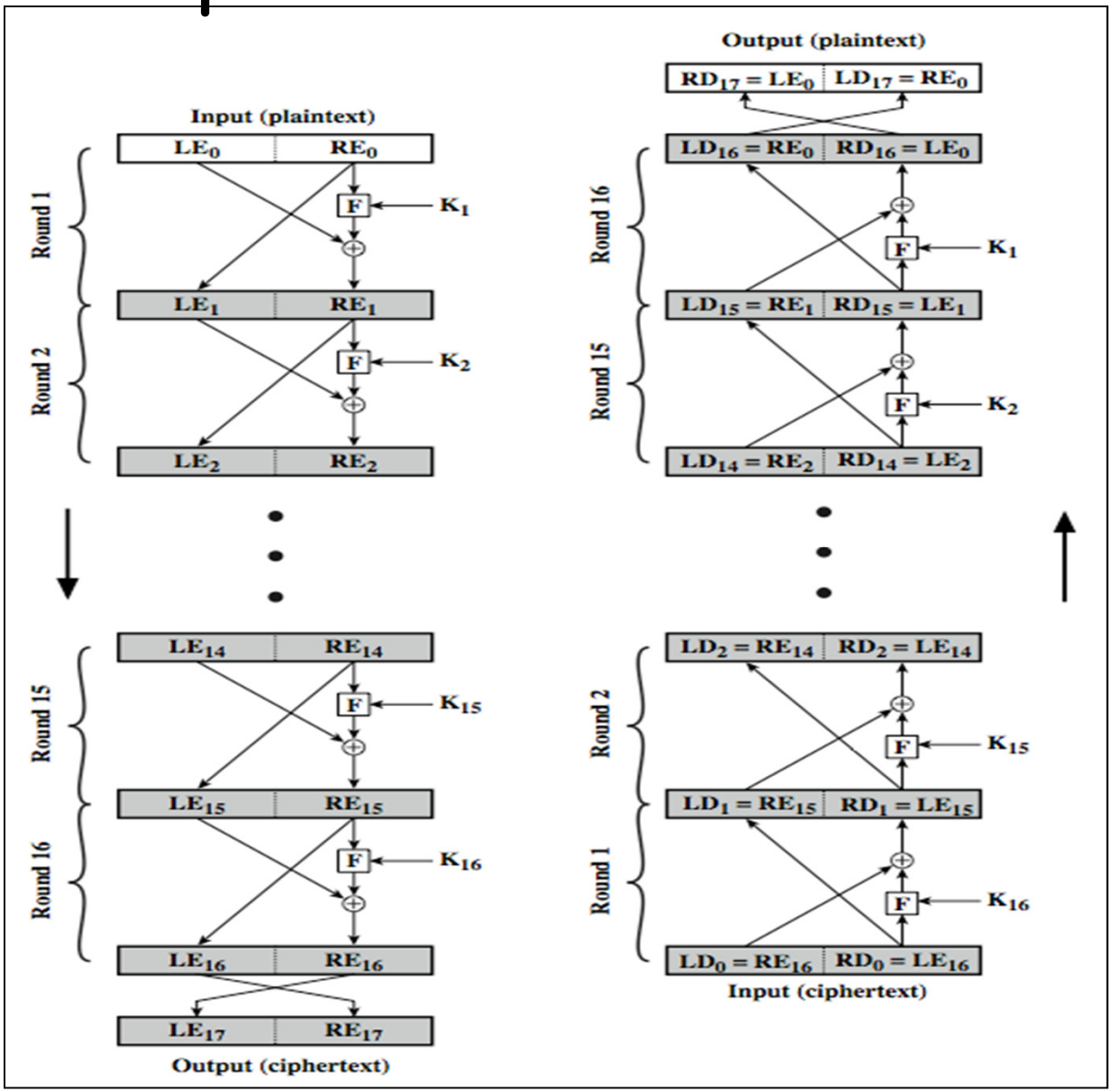
Feistel Cipher Structure

- Horst Feistel; working at IBM Thomas J Watson Research Labs devised the **feistel cipher** (early 70`s)
  - **His main contribution** was the invention of a suitable structure which adapted Shannon's Substitution-Permutation (S-P) network in an easily inverted structure.

Feistel Cipher Structure

- This structure partitions input block into two halves
  - process through multiple rounds which perform a substitution on left data half.
  - And, is based on round function of right half & subkey.
  - then have permutation swapping halves.

Feistel Cipher Structure

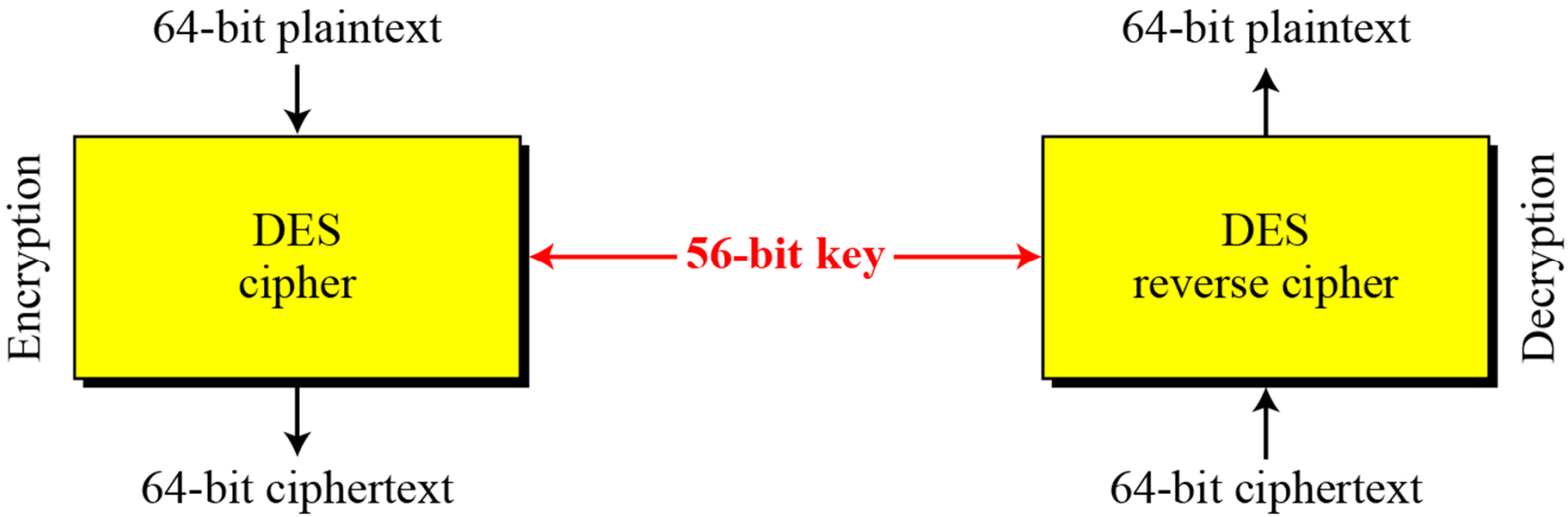


Feistel Cipher Design Elements

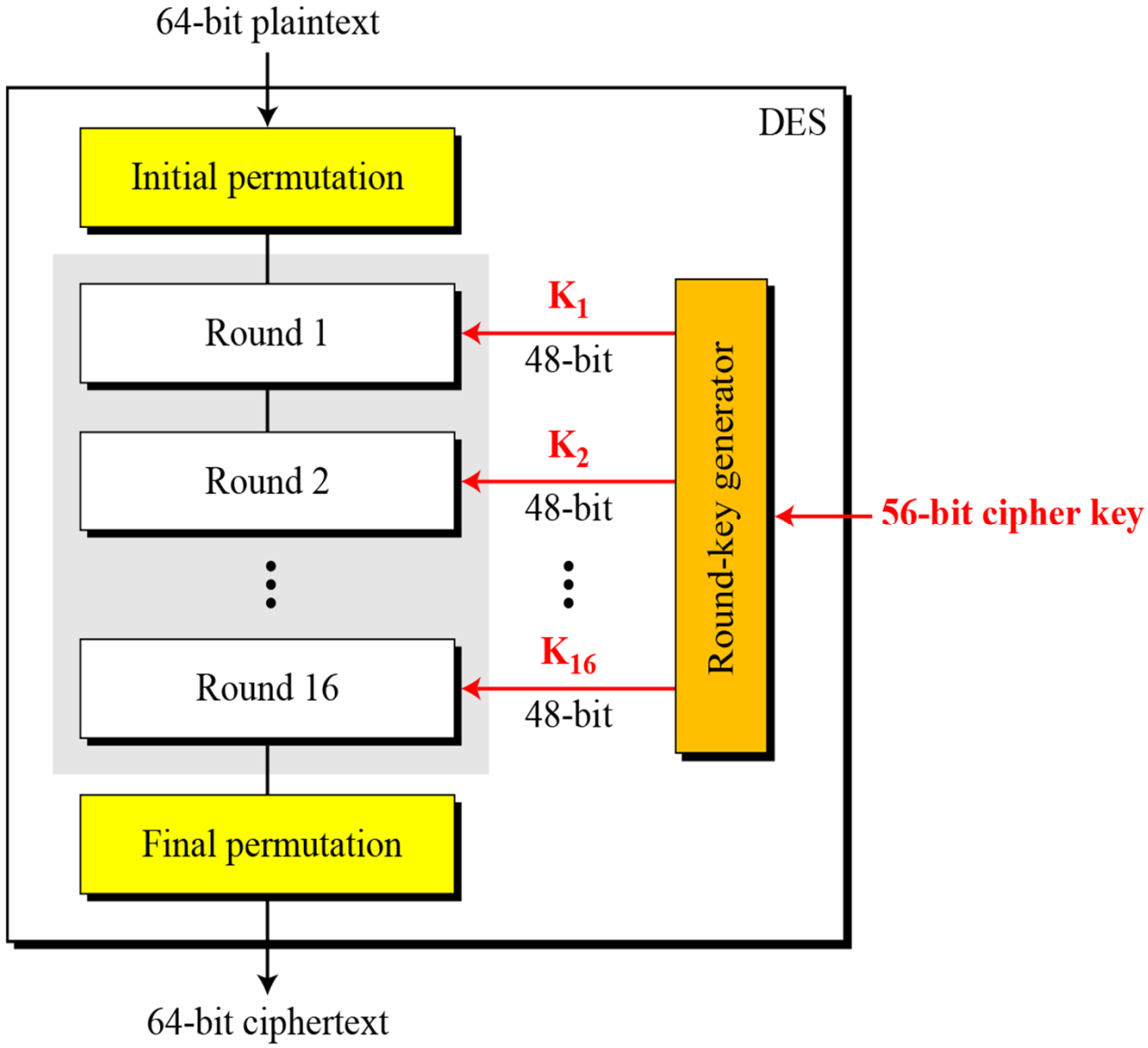
- block size
- key size
- number of rounds
- subkey generation algorithm
- round function
- fast software en/decryption
- ease of analysis

DES is a block cipher

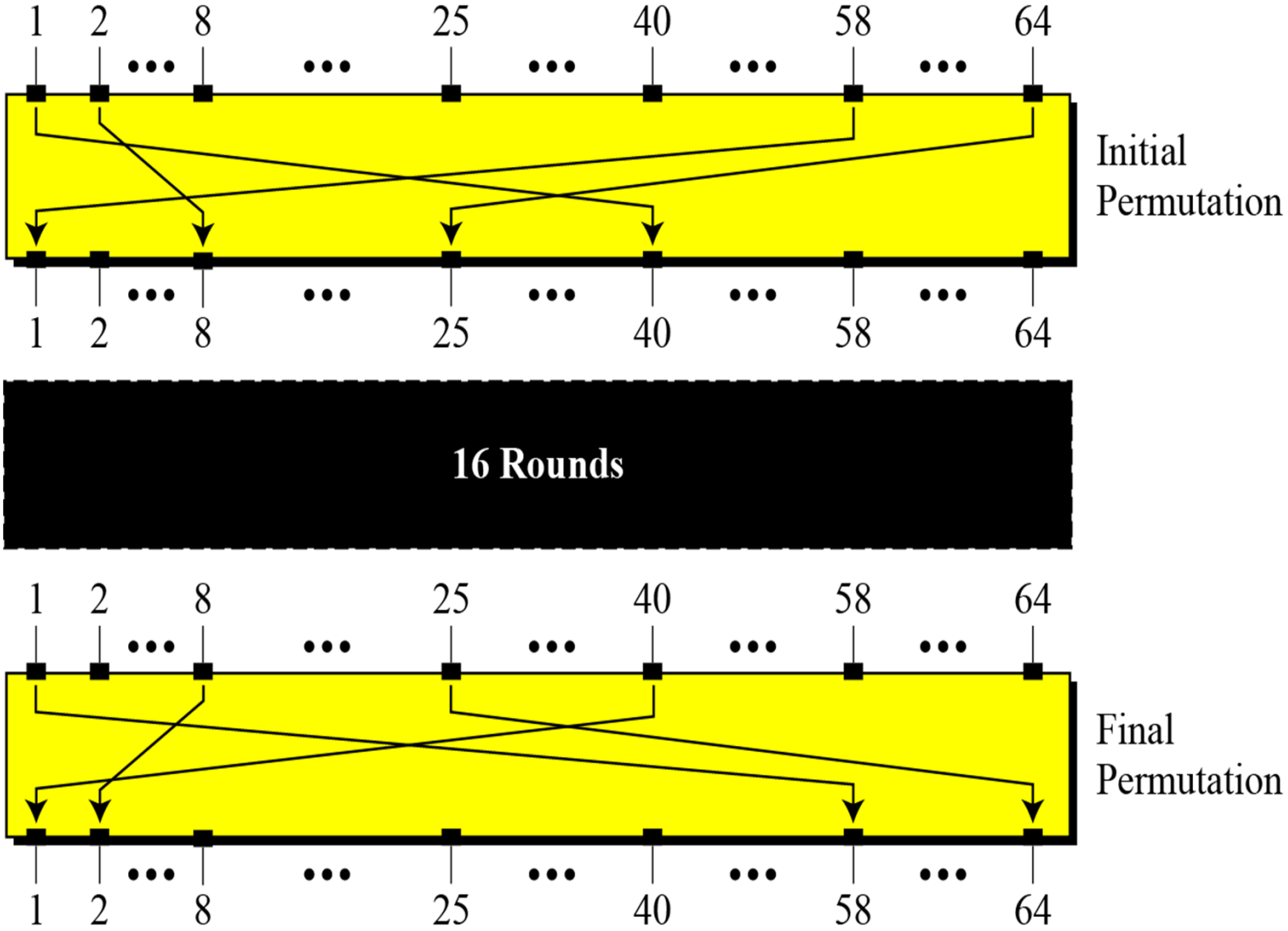
- *Encryption and decryption with DES*



General structure of DES



Initial and final permutation



Initial and final permutation tables

Initial Permutation								*Final Permutation*							
58	50	42	34	26	18	10	02	40	08	48	16	56	24	64	32
60	52	44	36	28	20	12	04	39	07	47	15	55	23	63	31
62	54	46	38	30	22	14	06	38	06	46	14	54	22	62	30
64	56	48	40	32	24	16	08	37	05	45	13	53	21	61	29
57	49	41	33	25	17	09	01	36	04	44	12	52	20	60	28
59	51	43	35	27	19	11	03	35	03	43	11	51	19	59	27
61	53	45	37	29	21	13	05	34	02	42	10	50	18	58	26
63	55	47	39	31	23	15	07	33	01	41	09	49	17	57	25

Initial and final permutation

Example

Find the output of the initial permutation box when the input is given in hexadecimal as: 0x0000 0080 0000 0002

Solution

Only bit 25 and bit 64 are 1s; the other bits are 0s. In the final permutation, bit 25 becomes bit 64 and bit 63 becomes bit 15. Therefore, the result is 0x0002 0000 0000 0001

Initial Permutation	*Final Permutation*
58 50 42 34 26 18 10 02	40 08 48 16 56 24 64 32
60 52 44 36 28 20 12 04	39 07 47 15 55 23 63 31
62 54 46 38 30 22 14 06	38 06 46 14 54 22 62 30
64 56 48 40 32 24 16 08	37 05 45 13 53 21 61 29
57 49 41 33 25 17 09 01	36 04 44 12 52 20 60 28
59 51 43 35 27 19 11 03	35 03 43 11 51 19 59 27
61 53 45 37 29 21 13 05	34 02 42 10 50 18 58 26
63 55 47 39 31 23 15 07	33 01 41 09 49 17 57 25

Initial and final permutation

Example

Prove that the initial and final permutations are the inverse of each other by finding the output of the final permutation if the input is 0x0002 0000 0000 0001

Solution

The input has only two 1s; the output must also have only two 1s. Using tables, we can find the output related to these two bits. Bit 15 in the input becomes bit 63 in the output. Bit 64 in the input becomes bit 25 in the output. So the output has only two 1s, bit 25 and bit 63.

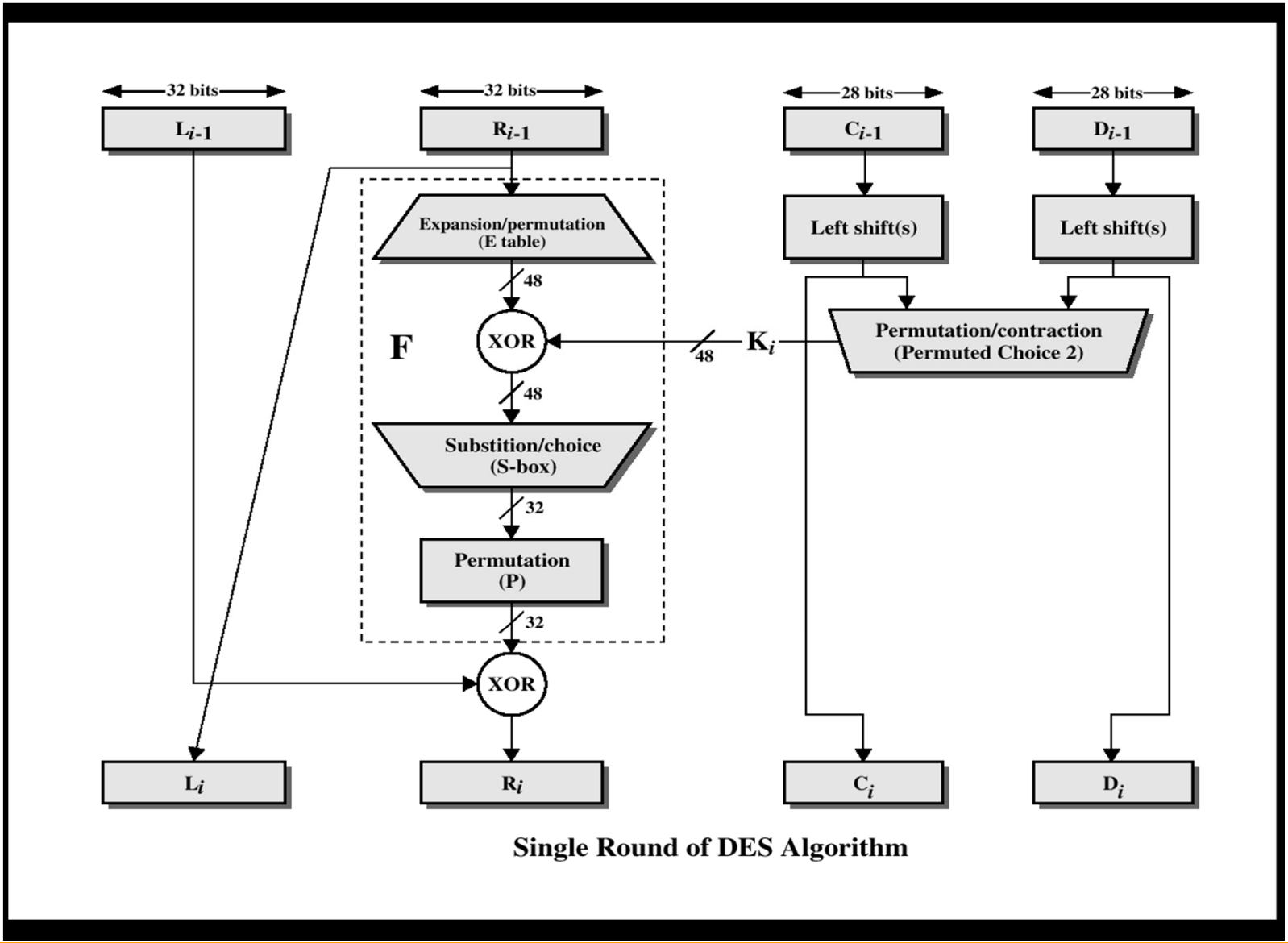
The result in hexadecimal is 0x0000 0080 0000 0002

Note

The initial and final permutations are straight P-boxes that are inverses of each other.
They have no cryptography significance in DES.

Symmetrical

Block:DES-1977

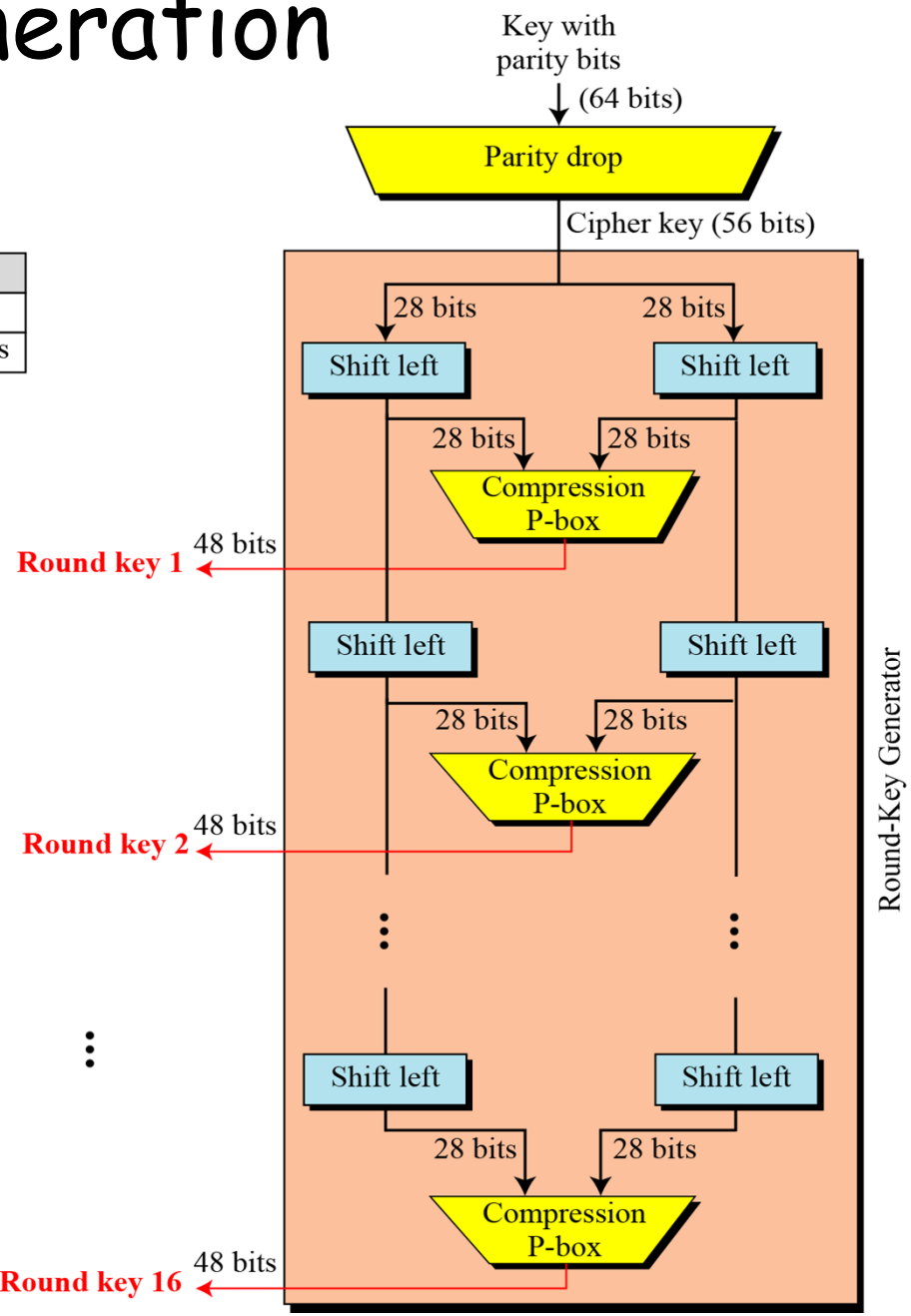


Single Round of DES Algorithm

Key Generation

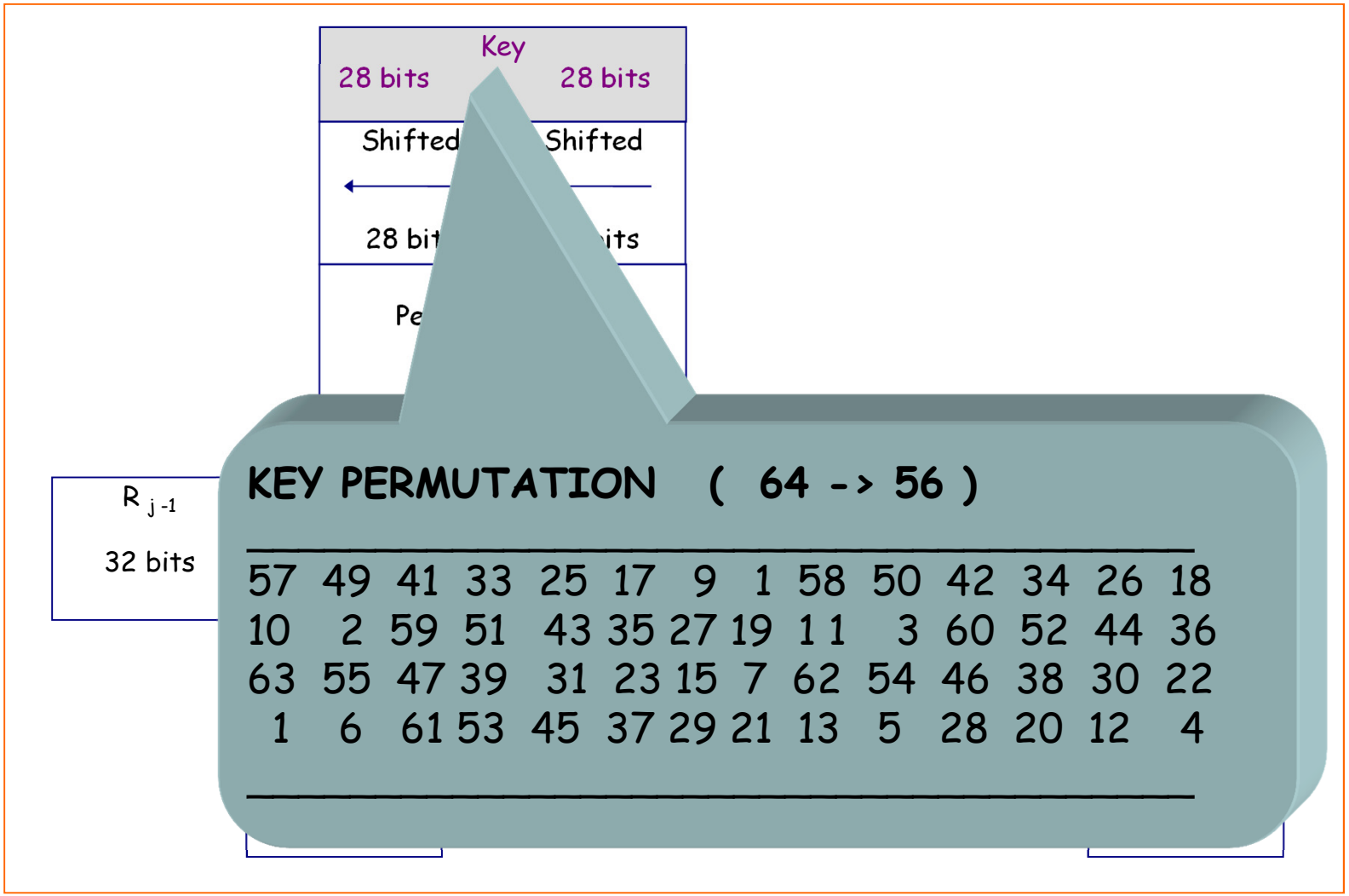
Shifting

Rounds	Shift
1, 2, 9, 16	one bit
Others	two bits



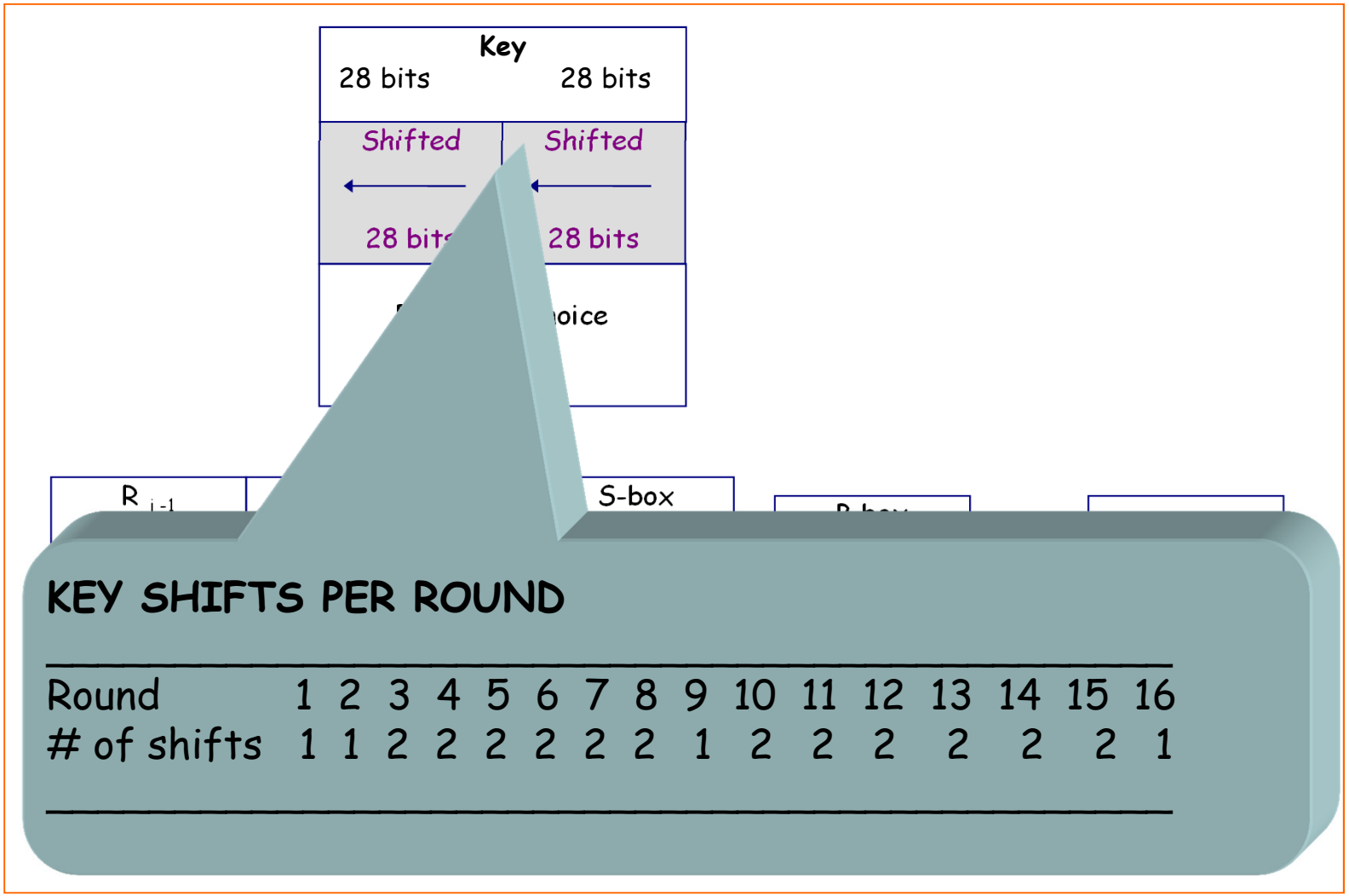
Symmetrical

Block:DES-1977



Symmetrical

Block:DES-1977

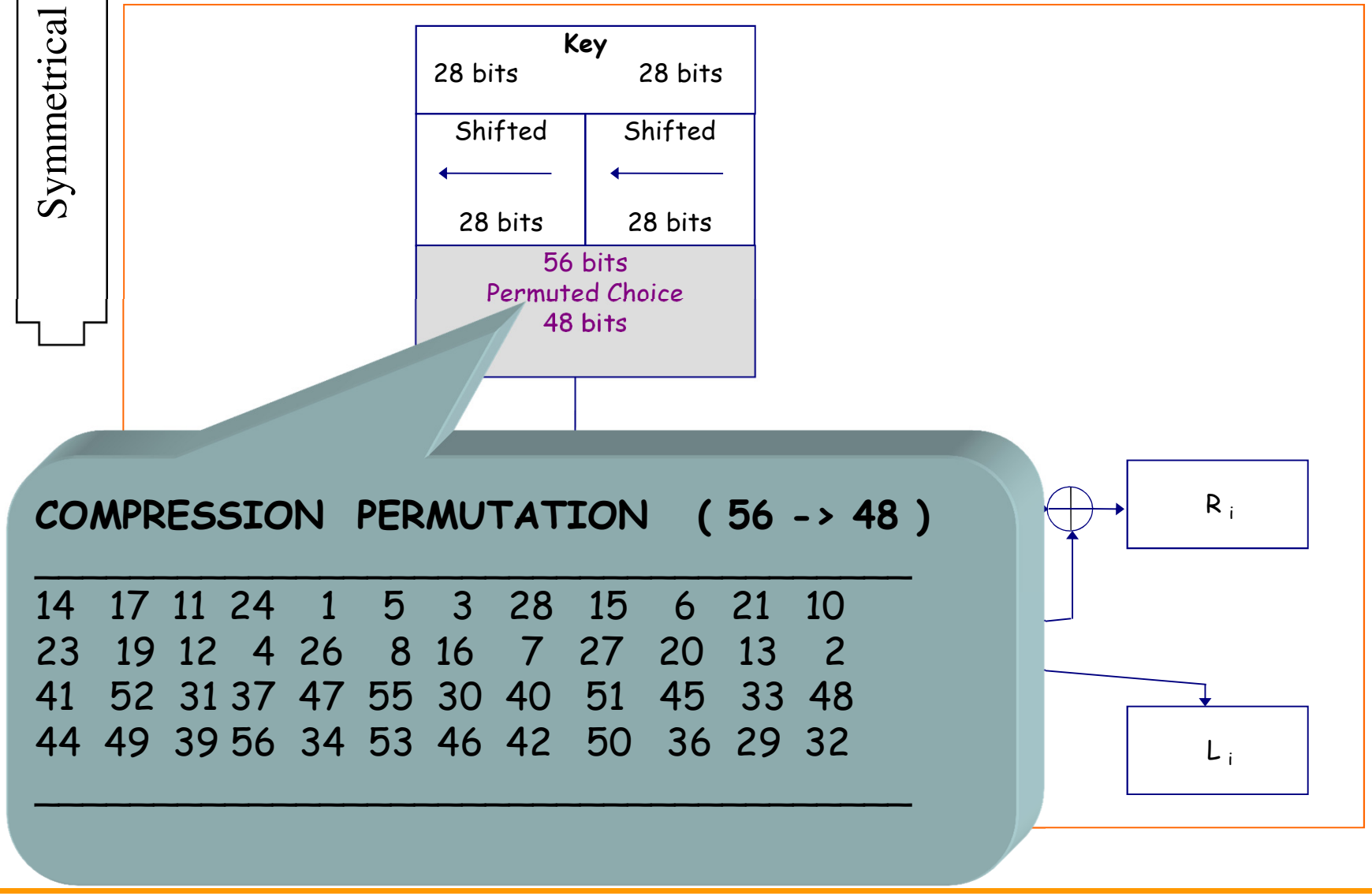


KEY SHIFTS PER ROUND

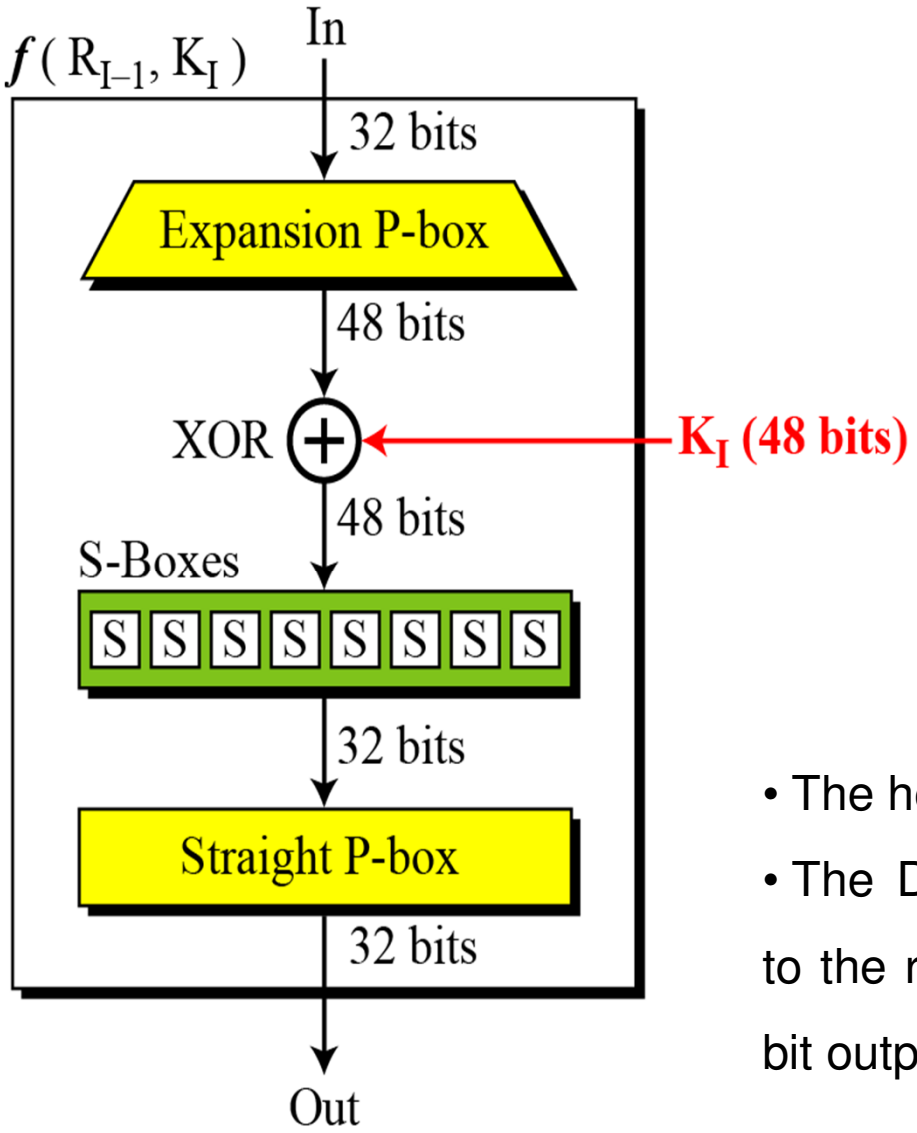
Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
# of shifts	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Symmetrical

Block:DES-1977



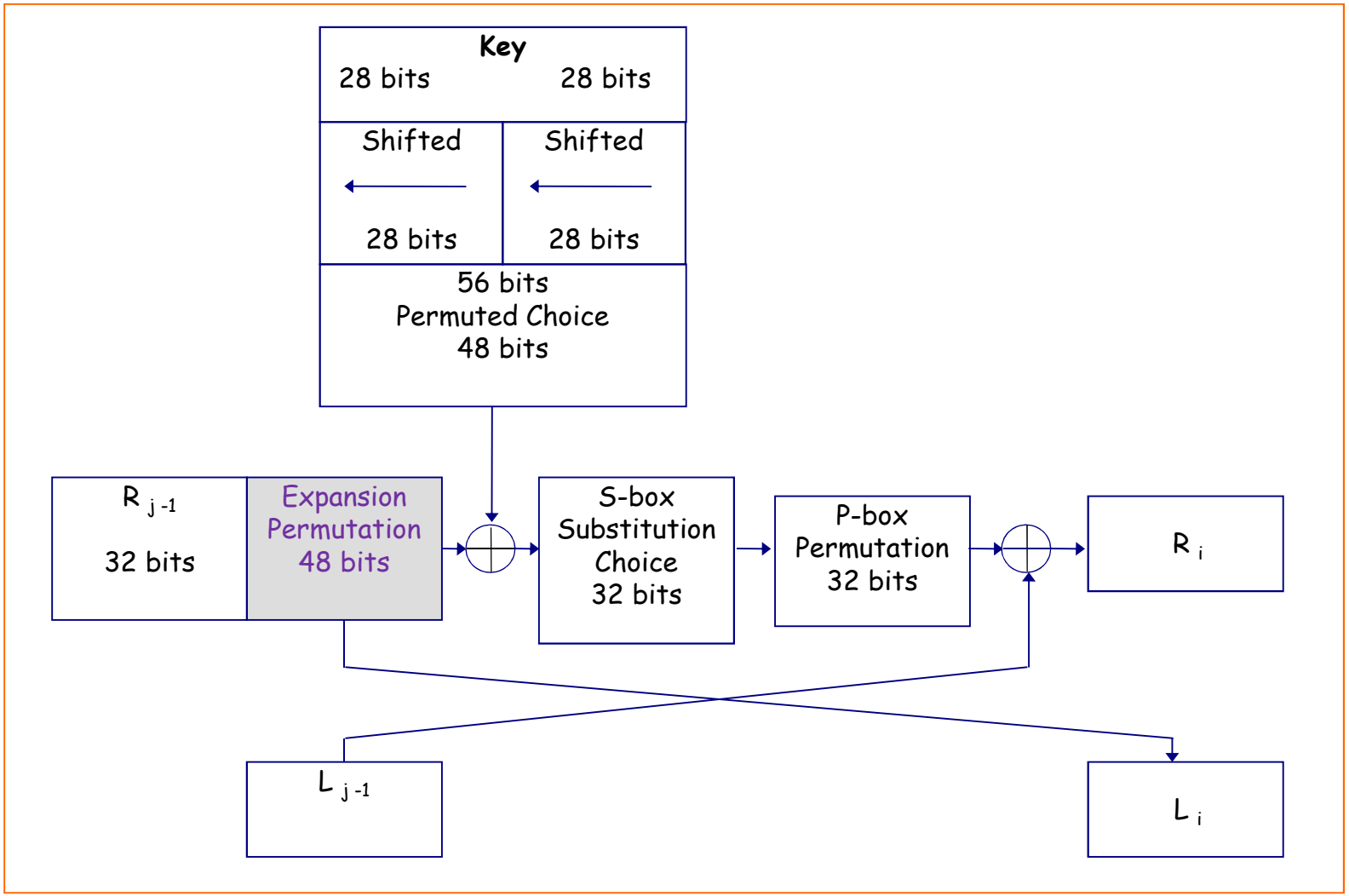
DES Function



- The heart of DES is the DES function.
- The DES function applies a 48-bit key to the rightmost 32 bits to produce a 32-bit output.

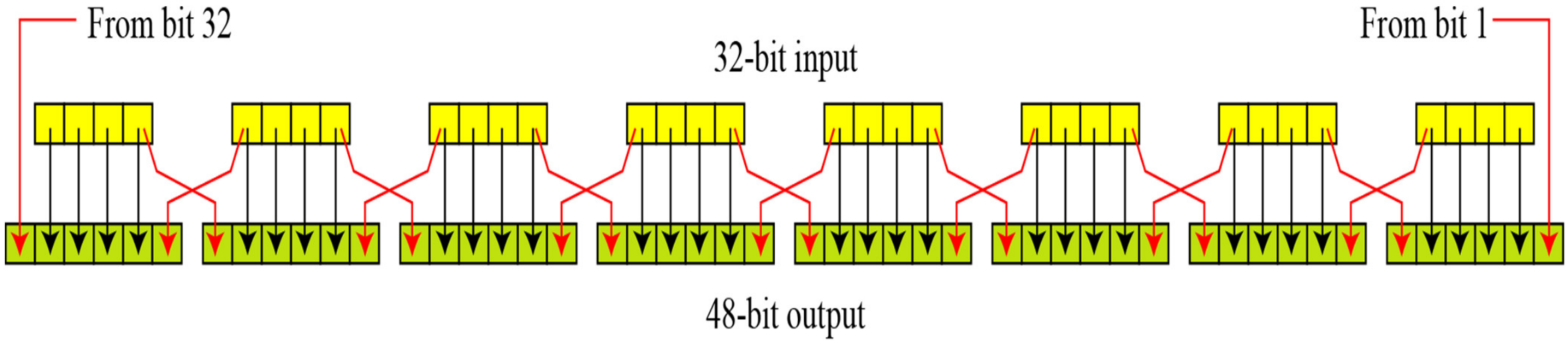
Symmetrical

Block:DES-1977



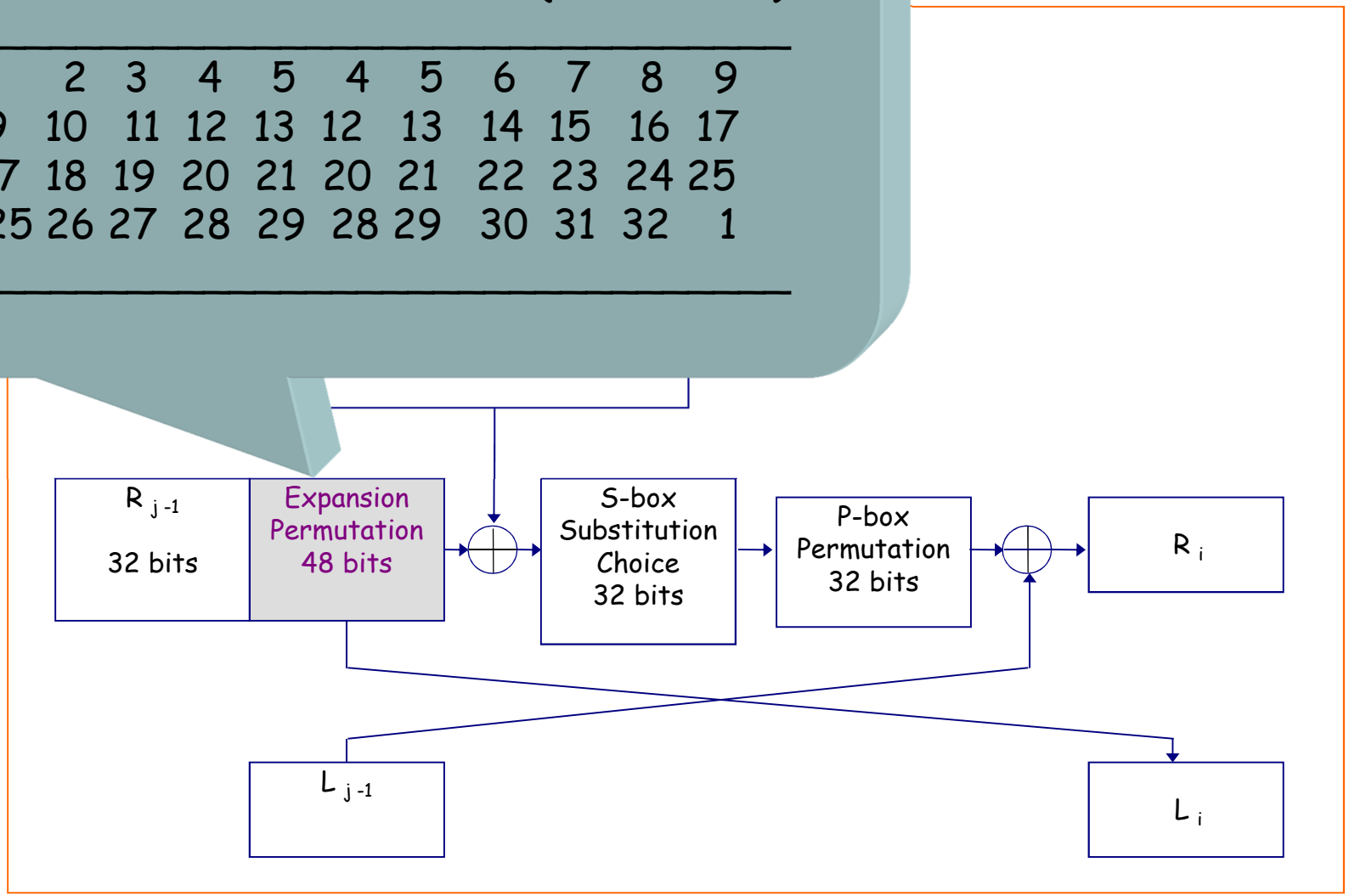
Expansion P-box

Since R_{l-1} is a 32-bit input and K_l is a 48-bit key, we first need to expand R_{l-1} to 48 bits.



EXPANSION PERMUTATION (32 -> 48)

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



XOR

Whitener (XOR)

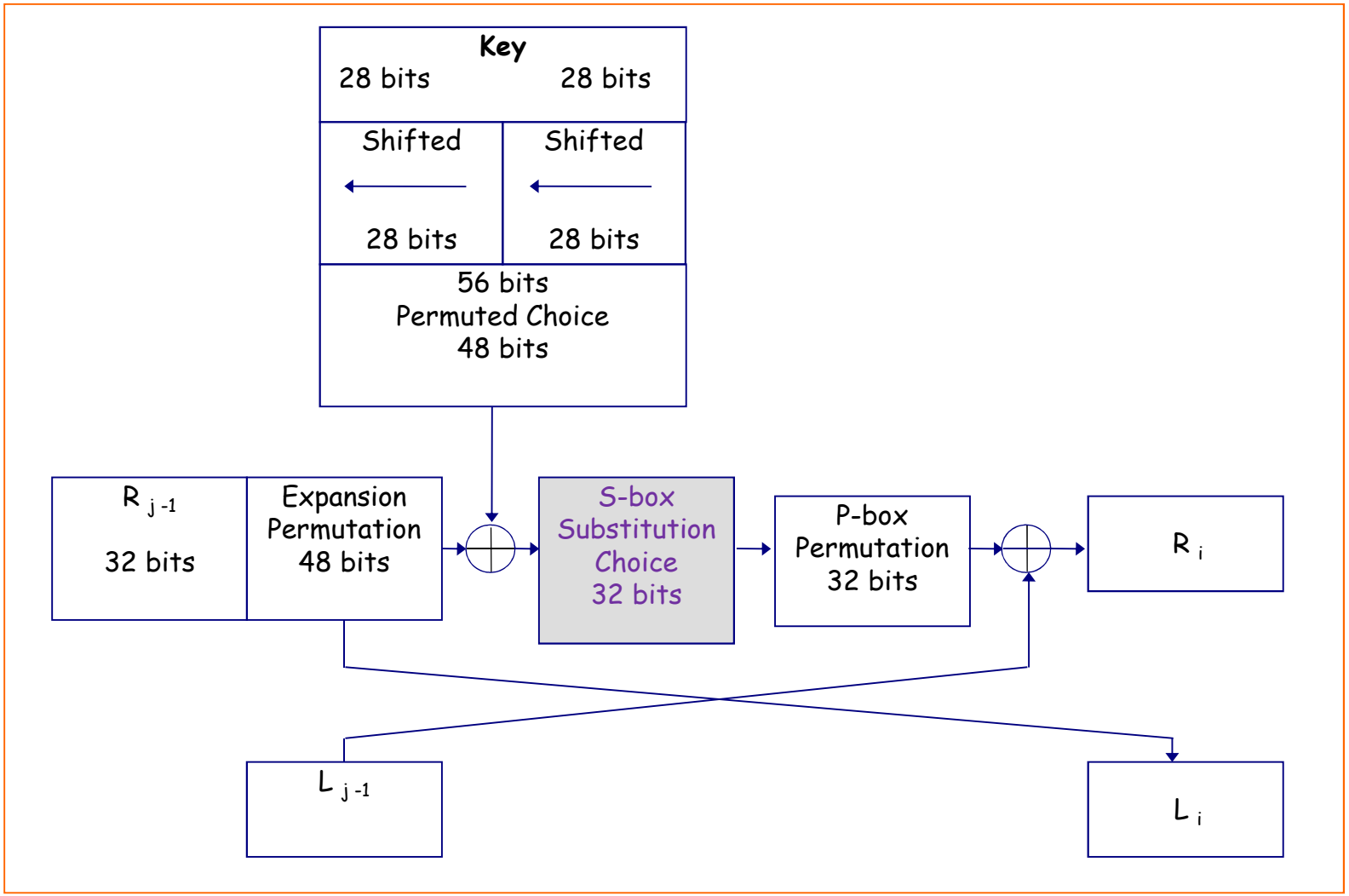
After the expansion permutation, DES uses the XOR operation on the expanded right section and the round key.

Note that both **the right section and the key are 48-bits in length**.

Also note that **the round key is used only in this operation**.

Symmetrical

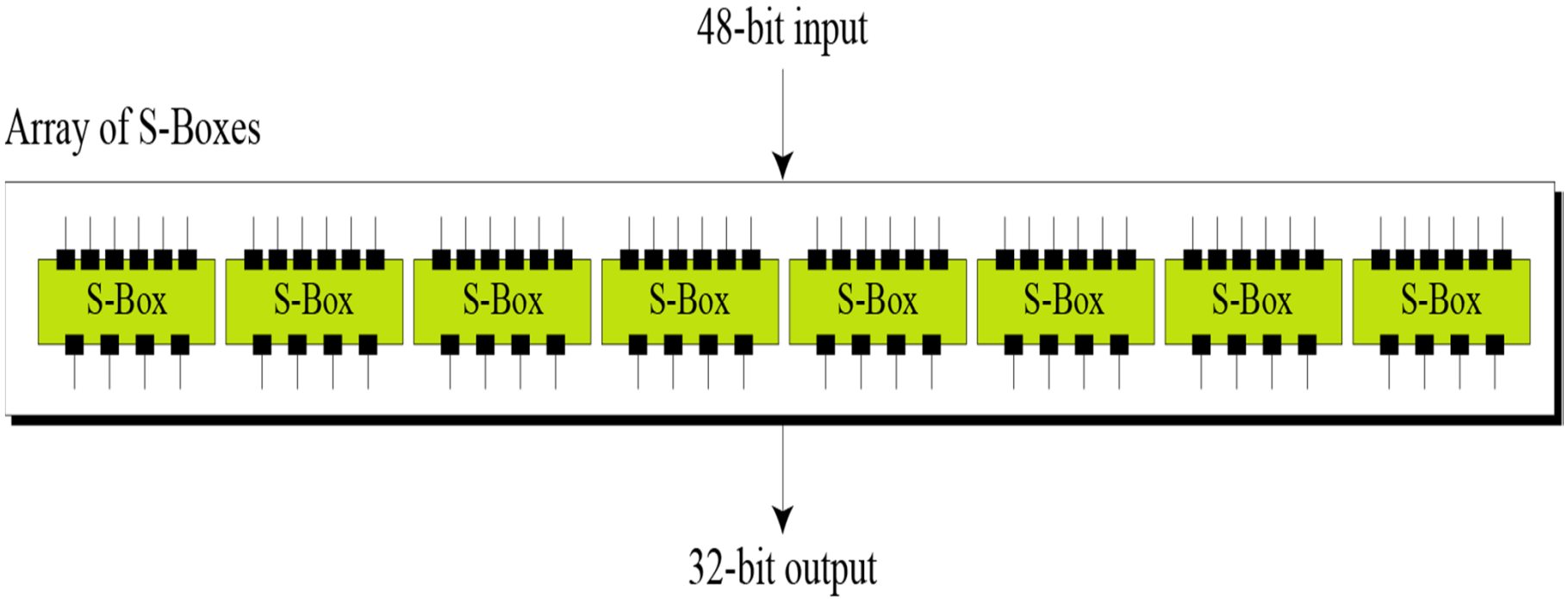
Block:DES-1977



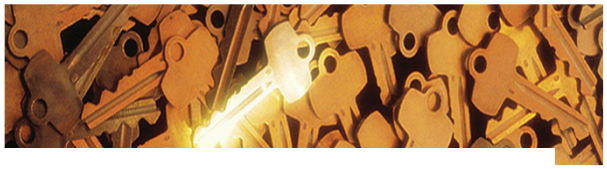
S-Boxes

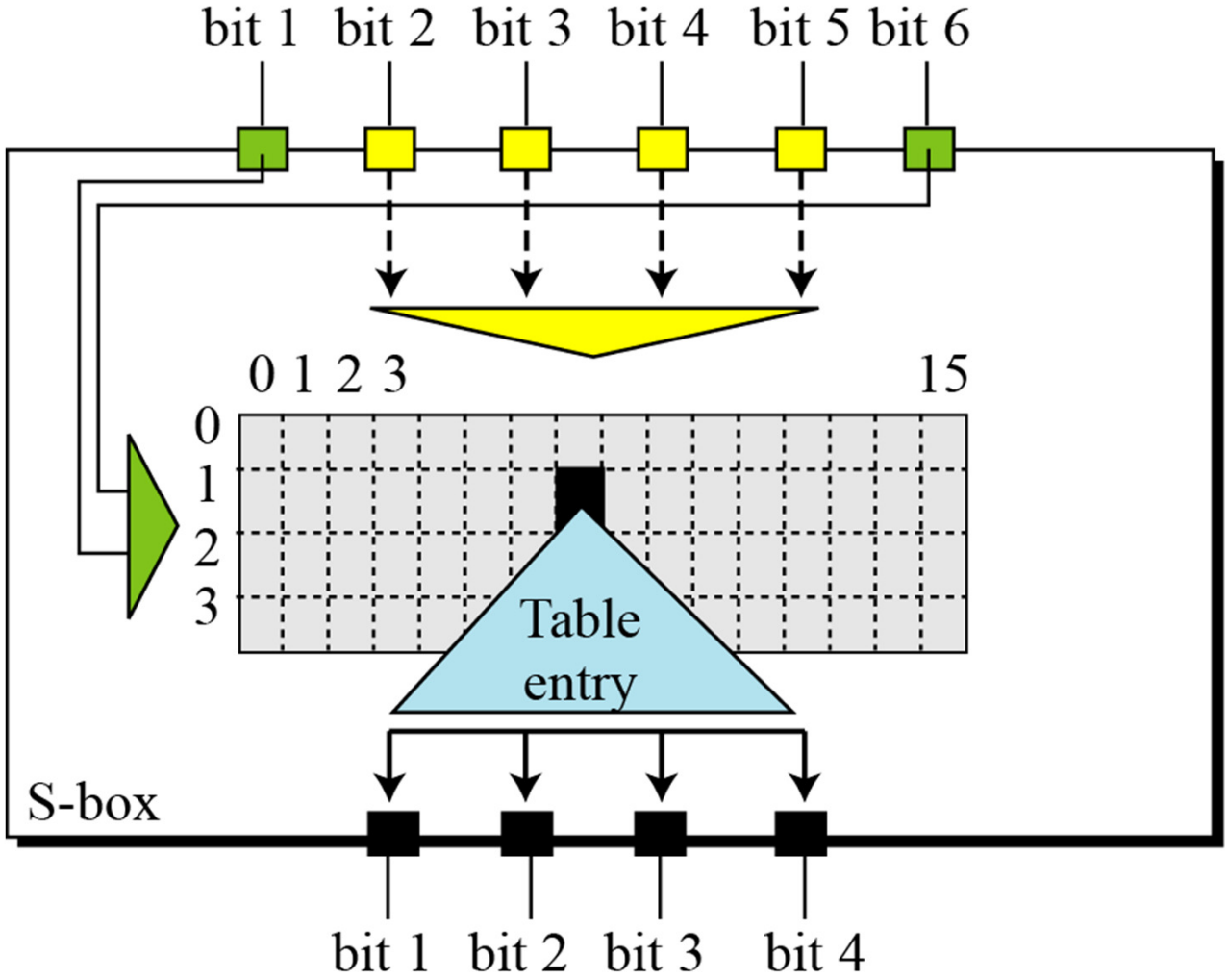
The S-boxes do the **real mixing (confusion).**

DES uses 8 S-boxes, each with a 6-bit input and a 4-bit output.



S-Box Rule




bit 1
bit 2
bit 3
bit 4
bit 5
bit 6
0 1 2 3
15
0
1
2
3
Table entry
S-box
bit 1
bit 2
bit 3
bit 4

*** s1 ***

14	4	13	1	2	15	11	8	3	10	6	12	5	9
0	15	7	4	14	2	13	1	10	6	12	11	9	5
4	1	14	8	13	6	2	11	15	12	9	7	3	10
15	12	8	2	4	9	1	7	5	11	3	14	10	0

*** s2 ***

15	1	8	14	6	11	3	4	9	7	2	13	12	
3	13	4	7	15	2	8	14	12	0	1	10		
0	14	7	11	10	4	13	1	5	8	12			
13	8	10	1	3	15	4	2	11	6				

*** s3 ***

10	0	9	14	6	3	15	5	1	13				
13	7	0	9	3	4	6	10	2					
13	6	4	9	8	15	3	0	11					
1	10	13	0	6	9	8	7	4					

*** s4 ***

7	13	14	3	0	6	9	10	1					
13	8	11	5	6	15	0	3	4	7				
10	6	9	0	12	11	7	13	15	1				
3	15	0	6	10	1	13	8	9	4				

*** s5 ***

2	12	4	1	7	10	11	6	8			15		
14	11	2	12	4	7	13	1	5		15	10	3	9
4	2	1	11	10	13	7	8	15	9	12	5	6	3
11	8	12	7	1	14	2	13	6	15	0	9	10	4

*** s6 ***

12	1	10	15	9	2	6	8	0	13	3	4	14	7
10	15	4	2	7	12	9	5	6	1	13	14	0	11
9	14	15	5	2	8	12	3	7	0	4	10	1	13
4	3	2	12	9	5	15	10	11	14	1	7	6	0

*** s7 ***

4	11	2	14	15	0	8	13	3	12	9	7	5	10
13	0	11	7	4	9	1	10	14	3	5	12	2	15
1	4	11	13	12	3	7	14	10	15	6	8	0	5
6	11	13	8	1	4	10	7	9	5	0	15	14	2

*** s8 ***

13	2	8	4	6	15	11	1	10	9	3	14	5	0
1	15	13	8	10	3	7	4	12	5	6	11	0	14
7	11	4	1	9	12	14	2	0	6	10	13	15	3
2	1	14	7	4	10	8	13	15	12	9	0	3	5

(15)=(1111)
10 2

EXAMPLE S-BOX USAGE:

...T TO THE 5th. S-BOX IS ==> 110011

1 0 0 1 1

... THE LAST BITS
...RM 11
...CH MEANS 3rd. ROW

...LE FOUR BITS
...E TO FORM 1001
(1001) = (9) WHICH MEANS 9th. COLUMN
2 10

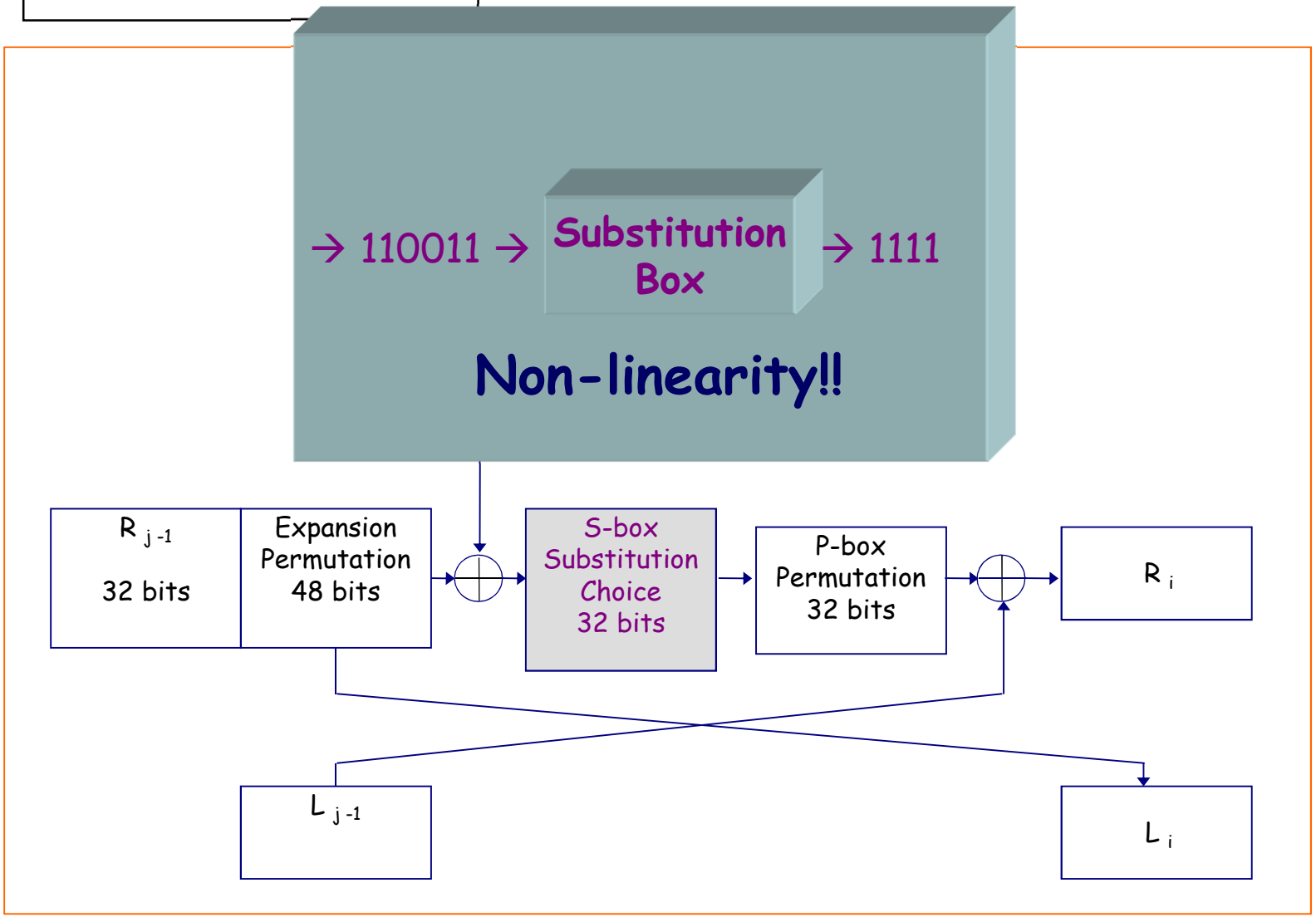
3rd. ROW, 9th. COLUMN OF
THE 5th. S-BOX IS ==> 15, AND SO

(15) = (1111)
10 2

SO THE VALUE **1111**
IS SUBSTITUED FOR **110011**

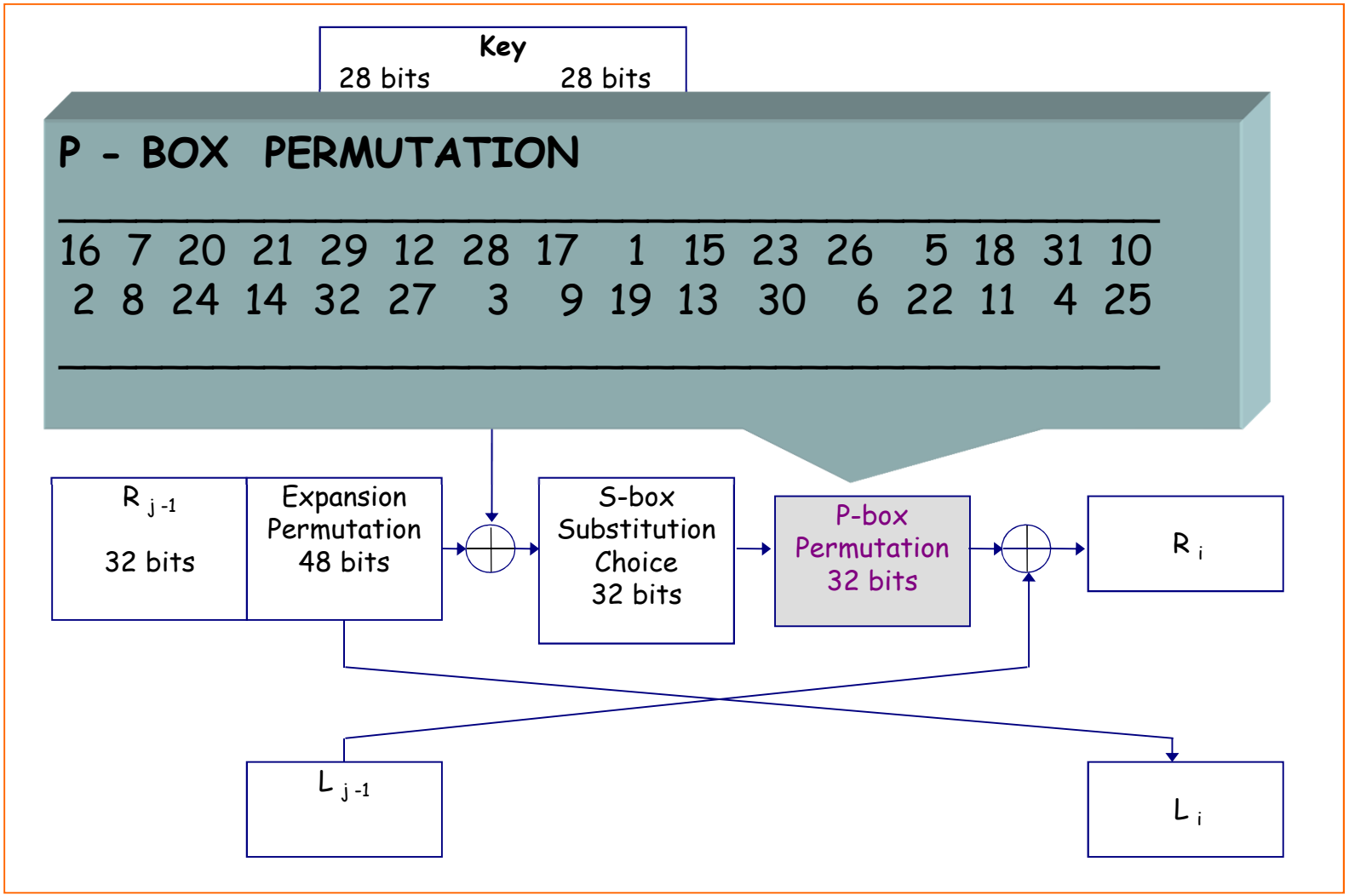
Symmetrical

Block:DES-1977

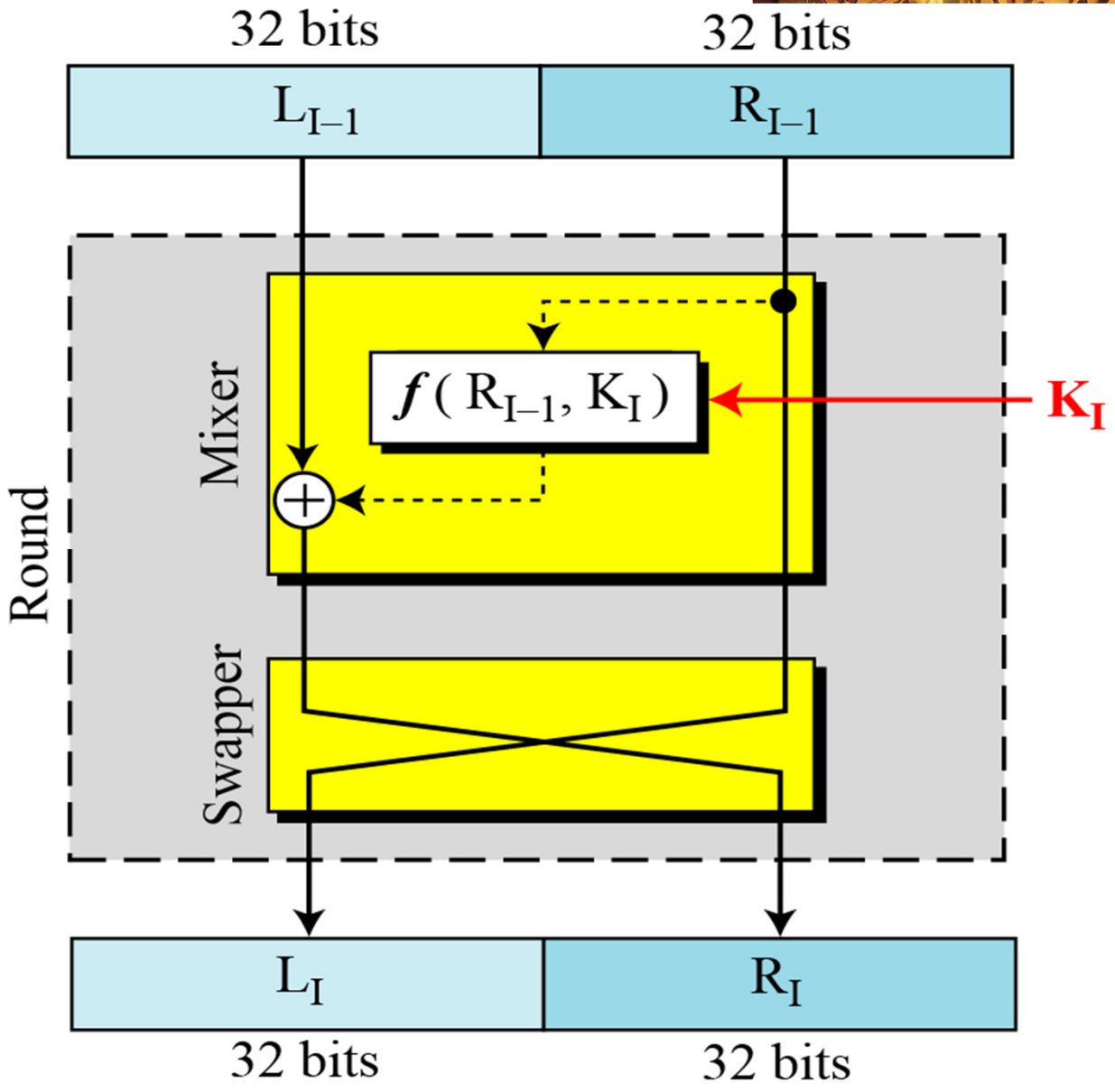


Symmetrical

Block:DES-1977



DES Rounds

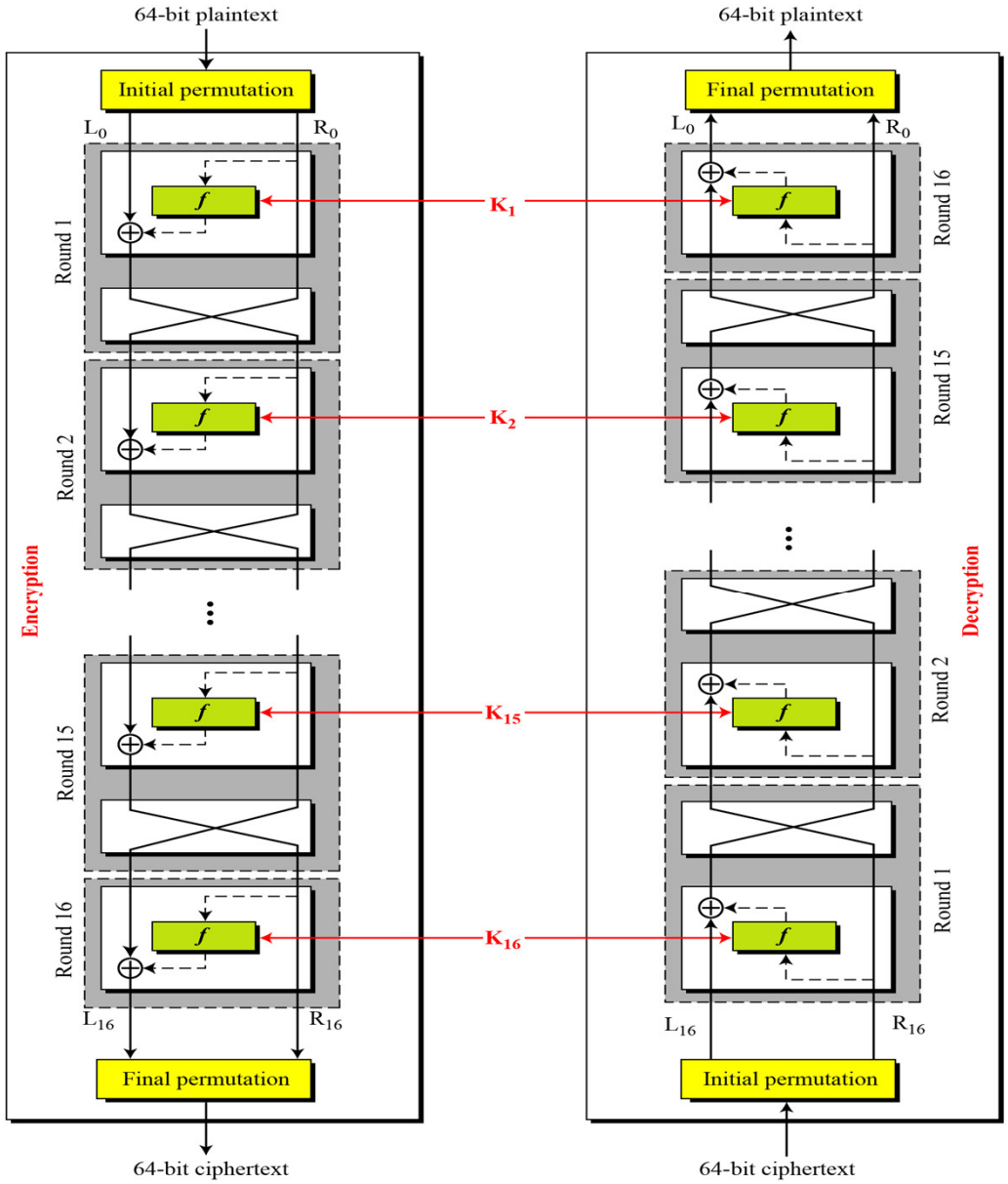


- DES uses 16 rounds.
- Each round of DES is a **Feistel cipher.**

DES Decryption

- Decrypt must unwind steps of data computation
- With Feistel design, do encryption steps again using subkeys in reverse order (SK16 … SK1)
  - IP undoes final FP step of encryption
  - 1st round with SK16 undoes 16th encrypt round
  - ….
  - 16th round with SK1 undoes 1st encrypt round
  - Then final FP undoes initial encryption IP
  - Thus recovering original data value

DES Decryption



The complete reversal
= Decryption

Encryption Example

We choose a random plaintext block and a random key, and determine what the ciphertext block would be (all in hexadecimal):

Plaintext: 123456ABCD132536
Key: AABB09182736CCDD

Plaintext: 123456ABCD132536

*After initial permutation:*14A7D67818CA18AD
After splitting: L_0=14A7D678 R_0=18CA18AD

Round	*Left*	*Right*	*Round Key*
Round 1	18CA18AD	5A78E394	194CD072DE8C
Round 2	5A78E394	4A1210F6	4568581ABCCE
Round 3	4A1210F6	B8089591	06EDA4ACF5B5
Round 4	B8089591	236779C2	DA2D032B6EE3
Round 5	236779C2	A15A4B87	69A629FEC913
Round 6	A15A4B87	2E8F9C65	C1948E87475E
Round 7	2E8F9C65	A9FC20A3	708AD2DDB3C0
Round 8	A9FC20A3	308BEE97	34F822F0C66D
Round 9	308BEE97	10AF9D37	84BB4473DCCC
Round 10	10AF9D37	6CA6CB20	02765708B5BF
Round 11	6CA6CB20	FF3C485F	6D5560AF7CA5
Round 12	FF3C485F	22A5963B	C2C1E96A4BF3
Round 13	22A5963B	387CCDAA	99C31397C91F
Round 14	387CCDAA	BD2DD2AB	251B8BC717D0
Round 15	BD2DD2AB	CF26B472	3330C5D9A36D
Round 16	19BA9212	CF26B472	181C5D75C66D

After combination: 19BA9212CF26B472

Ciphertext: C0B7A8D05F3A829C *(after final permutation)*

Decryption Example

Let us see how Bob, at the destination, can decipher the ciphertext received from Alice using the same key.

Ciphertext: C0B7A8D05F3A829C			
After initial permutation: 19BA9212CF26B472 After splitting: L_0=19BA9212 R_0=CF26B472			
Round	*Left*	*Right*	*Round Key*
Round 1	CF26B472	BD2DD2AB	181C5D75C66D
Round 2	BD2DD2AB	387CCDAA	3330C5D9A36D
. . .	. . .	. . .	. . .
Round 15	5A78E394	18CA18AD	4568581ABCCE
Round 16	14A7D678	18CA18AD	194CD072DE8C
After combination: 14A7D67818CA18AD			
Plaintext:123456ABCD132536 (after final permutation)			

Avalanche Effect

A desirable property of any encryption algorithm is that **a small change in either the plaintext or the key should produce a significant change in the ciphertext.**

- A main desired property of an encryption algorithm.
- Where a change of **one** input or key bit results in changing approx **half** output bits.
- Making attempts to “home-in” by guessing keys impossible.
- And, DES exhibits strong avalanche.

Avalanche in DES

Table shows the result when the fourth bit of the plaintext is changed.
The plaintext is 12468aceeca86420.
The second column shows the intermediate 64-bit values at the end of each round for the two plaintexts.
The third column shows the number of bits that differ between the two intermediate values.
The table shows that after just three rounds, 18 bits differ between the two blocks.

Round		δ
	02468aceeca86420 12468aceeca86420	1
1	3cf03c0fbad22845 3cf03c0fbad32845	1
2	bad2284599e9b723 bad3284539a9b7a3	5
3	99e9b7230bae3b9e 39a9b7a3171cb8b3	18
4	0bae3b9e42415649 171cb8b3ccaca55e	34
5	4241564918b3fa41 ccaca55ed16c3653	37
6	18b3fa419616fe23 d16c3653cf402c68	33
7	9616fe2367117cf2 cf402c682b2cefbc	32
8	67117cf2c11bfc09 2b2cefbc99f91153	33

Round		δ
9	c11bfc09887fbc6c 99f911532eed7d94	32
10	887fbc6c600f7e8b 2eed7d94d0f23094	34
11	600f7e8bf596506e d0f23094455da9c4	37
12	f596506e738538b8 455da9c47f6e3cf3	31
13	738538b8c6a62c4e 7f6e3cf34bc1a8d9	29
14	c6a62c4e56b0bd75 4bc1a8d91e07d409	33
15	56b0bd7575e8fd8f 1e07d4091ce2e6dc	31
16	75e8fd8f25896490 1ce2e6dc365e5f59	32
IP^{-1}	da02ce3a89ecac3b 057cde97d7683f2a	32

Modes of DES

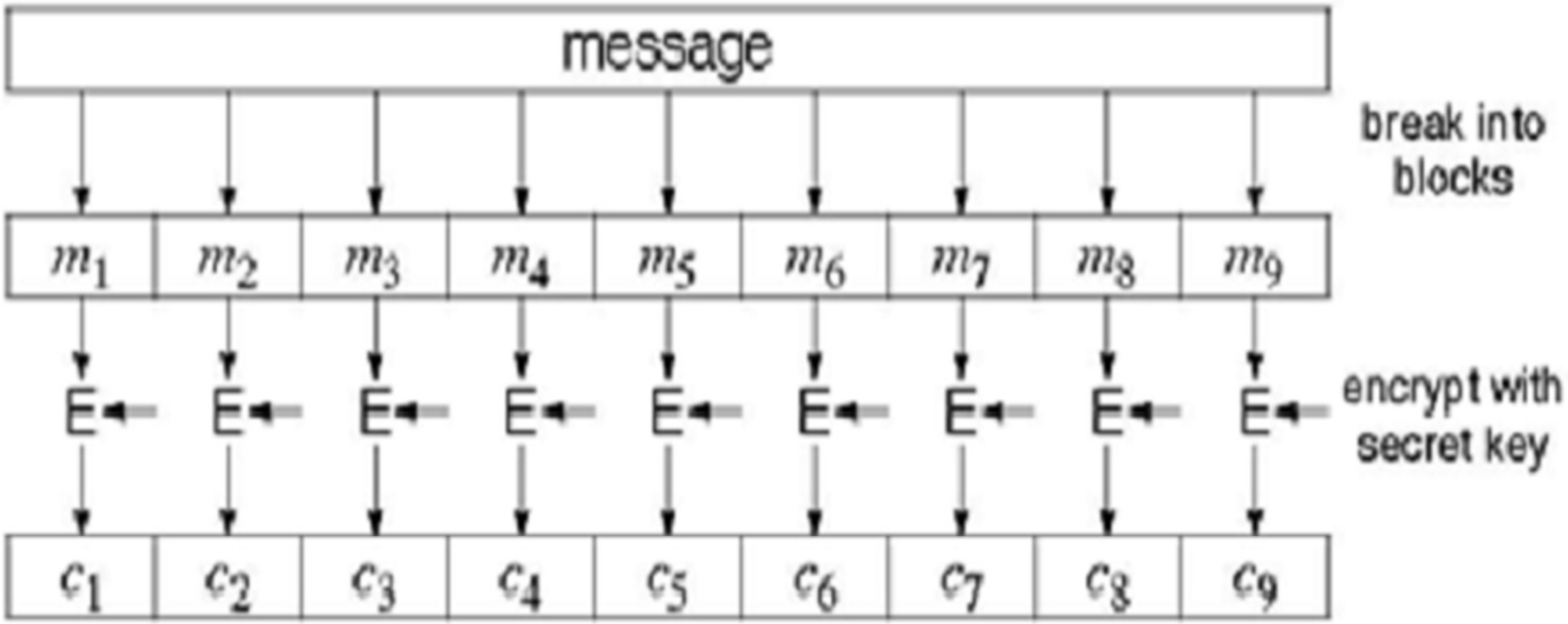
- Refers to the different utilization modes of the same algorithm; which are
  - Electronic Code Book-ECB
  - Cipher Block Chaining – CBC
  - Cipher Feedback – CFB
  - Output Feedback – OFB
  - Counter - CTR

Electronic Code Book-ECB

- Message is broken into independent blocks which are encrypted.
- Each block is a value which is substituted, like a codebook, hence the name.
- Each block is encoded independently of the other blocks.

$$C_i = DES_{K1}(P_i)$$

Electronic Code Book-ECB



Limitations of ECB

- Repetitions in message can be reflected in cipher text:
  - if aligned with message block,
  - particularly with data such graphics,
  - or with messages that change very little, which become a code-book analysis problem.
- Weakness is because enciphered message blocks are independent of each other.

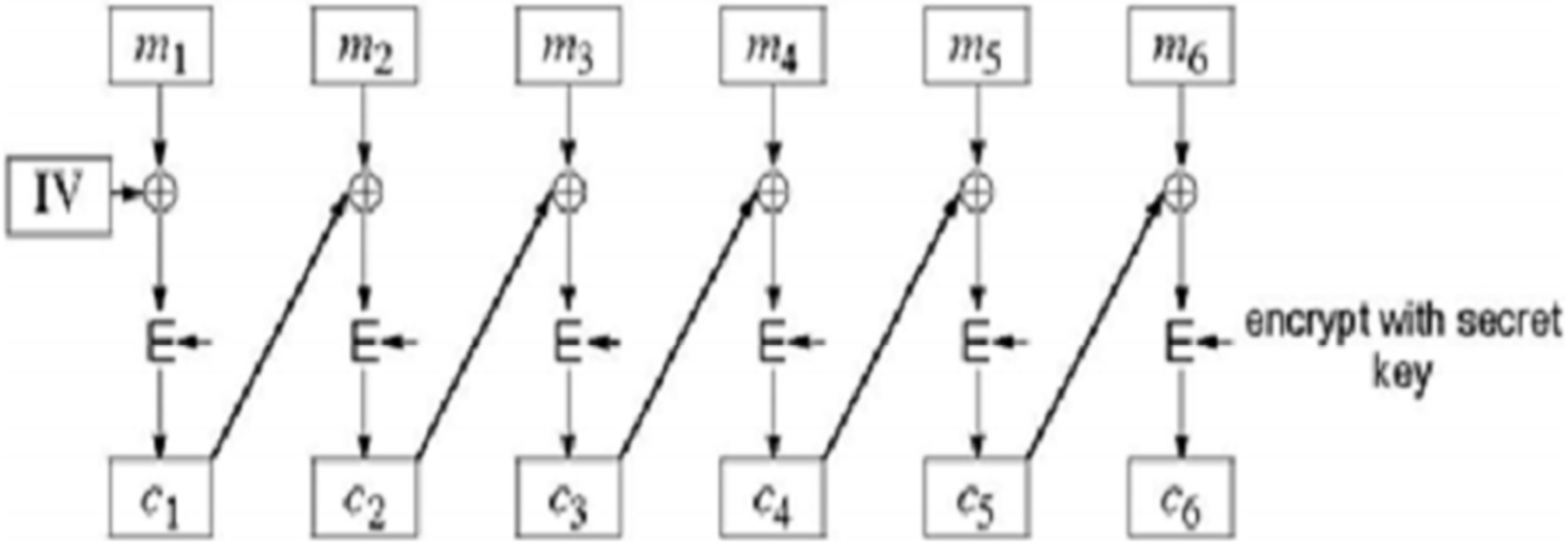
Cipher Block Chaining - CBC

- Is an enhanced mode of ECB.
- Message is broken into blocks.
- Linked together in encryption operation.
- Each previous cipher blocks is chained with current plaintext block, hence name.
- Use Initial Vector (IV) to start process.

$$C_{-1} = IV$$

$$C_i = E_K(P_i \text{ XOR } C_{i-1})$$

Cipher Block Chaining - CBC



Advantages and Limitations of CBC

- A cipher text block depends on all blocks before it.
- Any change to a block affects all following cipher text blocks.

To start need an **Initial Value** (IV) which must be known by both sender and receiver .

  - However; if IV is sent in the clear, an attacker can change bits of the first block, and change IV to compensate.
  - Hence, either IV must be a fixed value (as in EFTPOS) or it must be sent encrypted in ECB mode before the rest of message.

Cipher Feedback - CFB

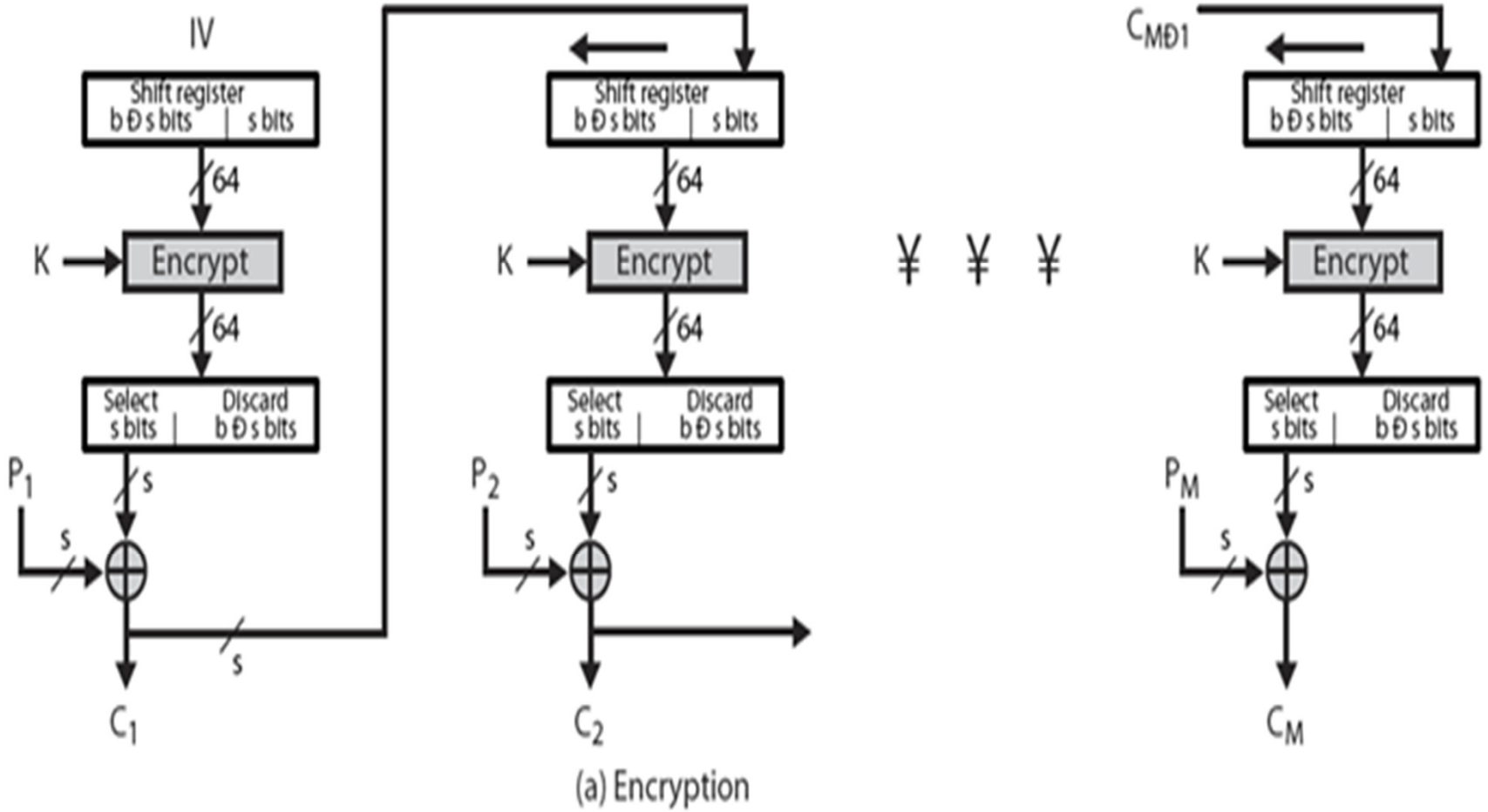
- Message is treated as a stream of bits or bytes.
- Result is feed back for next stage (hence the name).
- Standard allows any number of bit (1,8, 64 or 128 etc) to be feed back.
  - denoted CFB-1, CFB-8, CFB-64, CFB-128 etc
- Most efficient to use all bits in block (64 or 128).

$$C_{-1} = IV$$

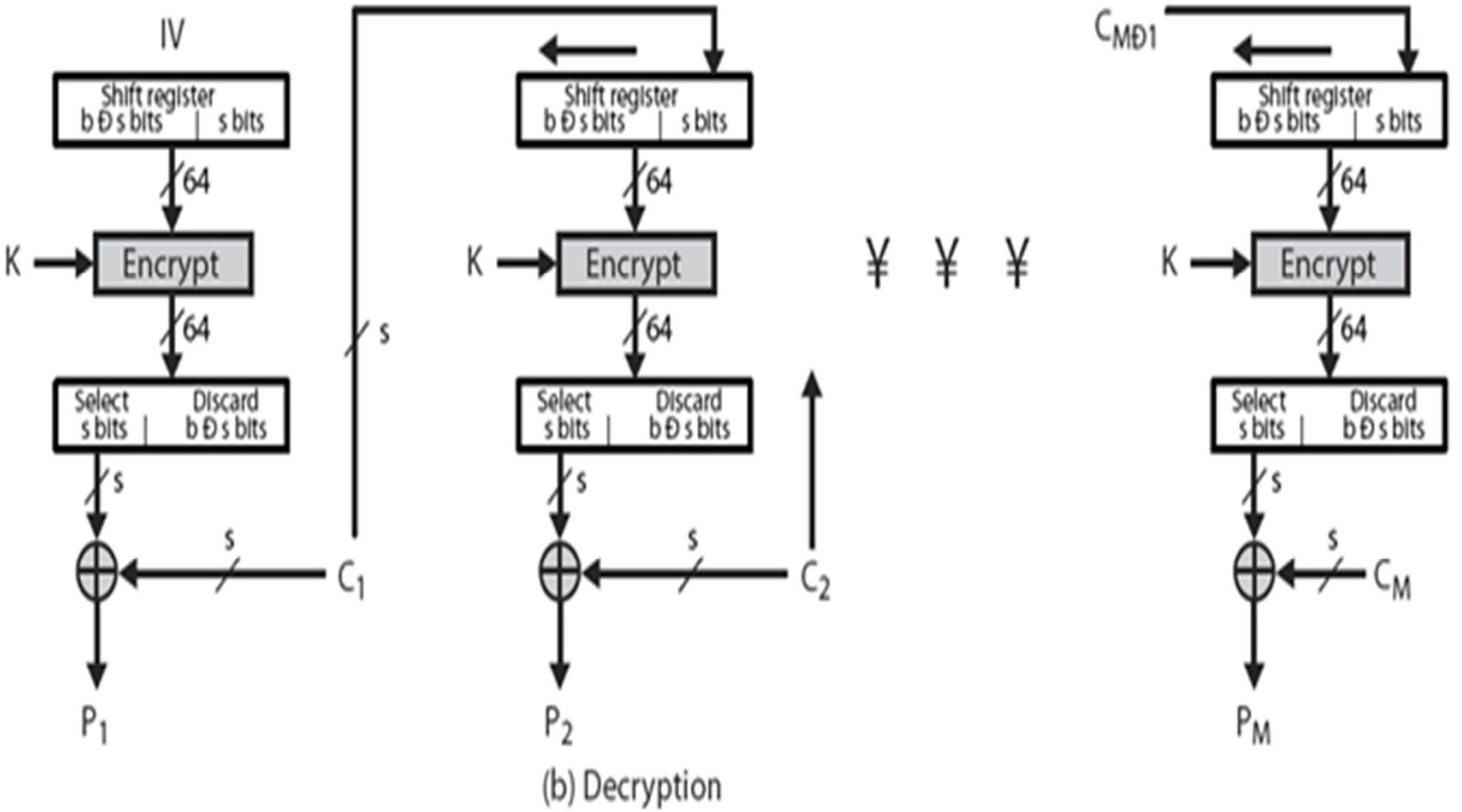
$$C_i = P_i \text{ XOR } E_K(C_{i-1})$$

- Used for stream data encryption.

Cipher Feedback - CFB



Cipher Feedback - CFB



(b) Decryption

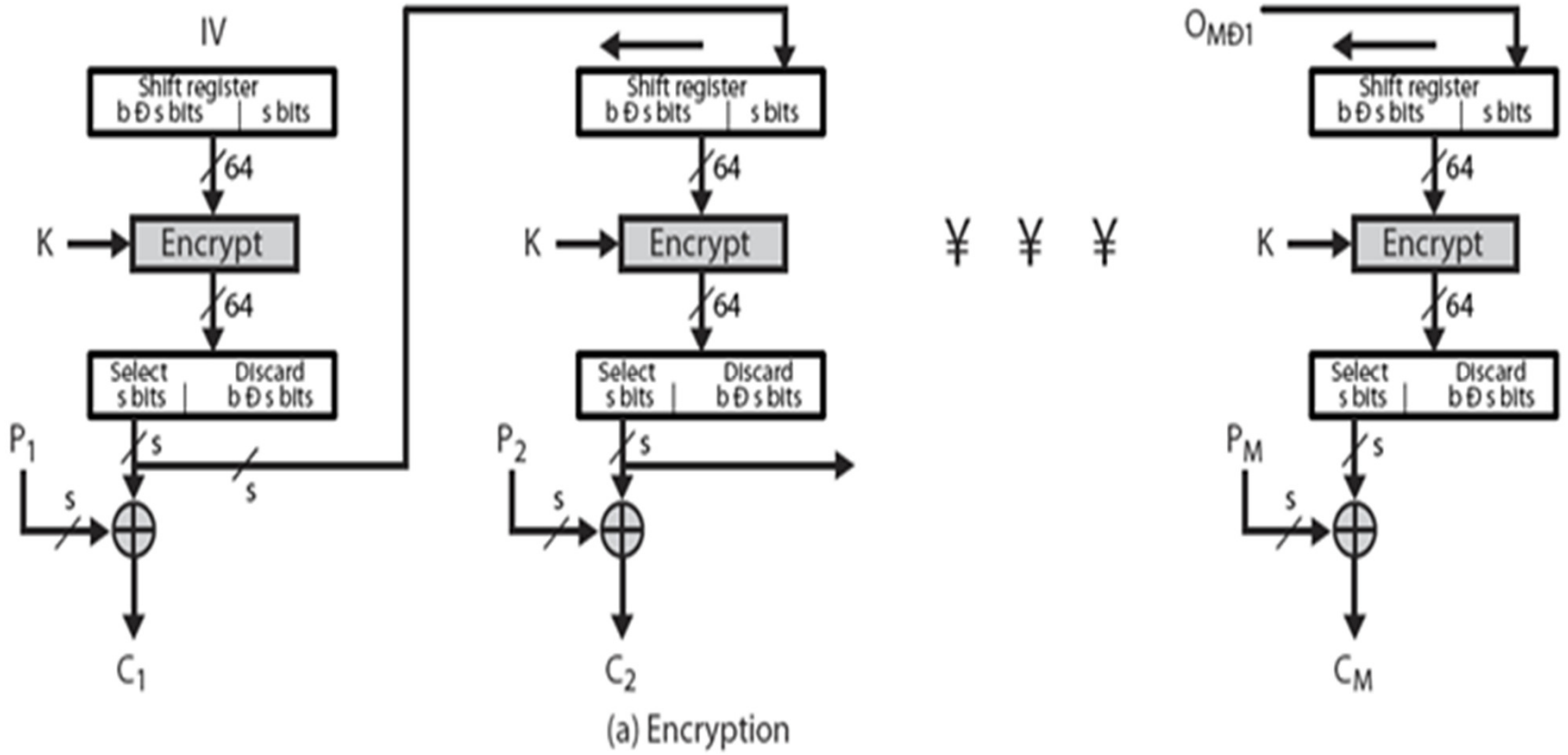
Advantages and Limitations of CFB

- Appropriate when data arrives in bits/bytes
- Most common stream mode.
- Note that the block cipher is used in encryption mode at both ends.
- Errors during transmission propagate for several blocks only (till the “dirty” part is eliminated from the shift register).

Output Feedback - OFB

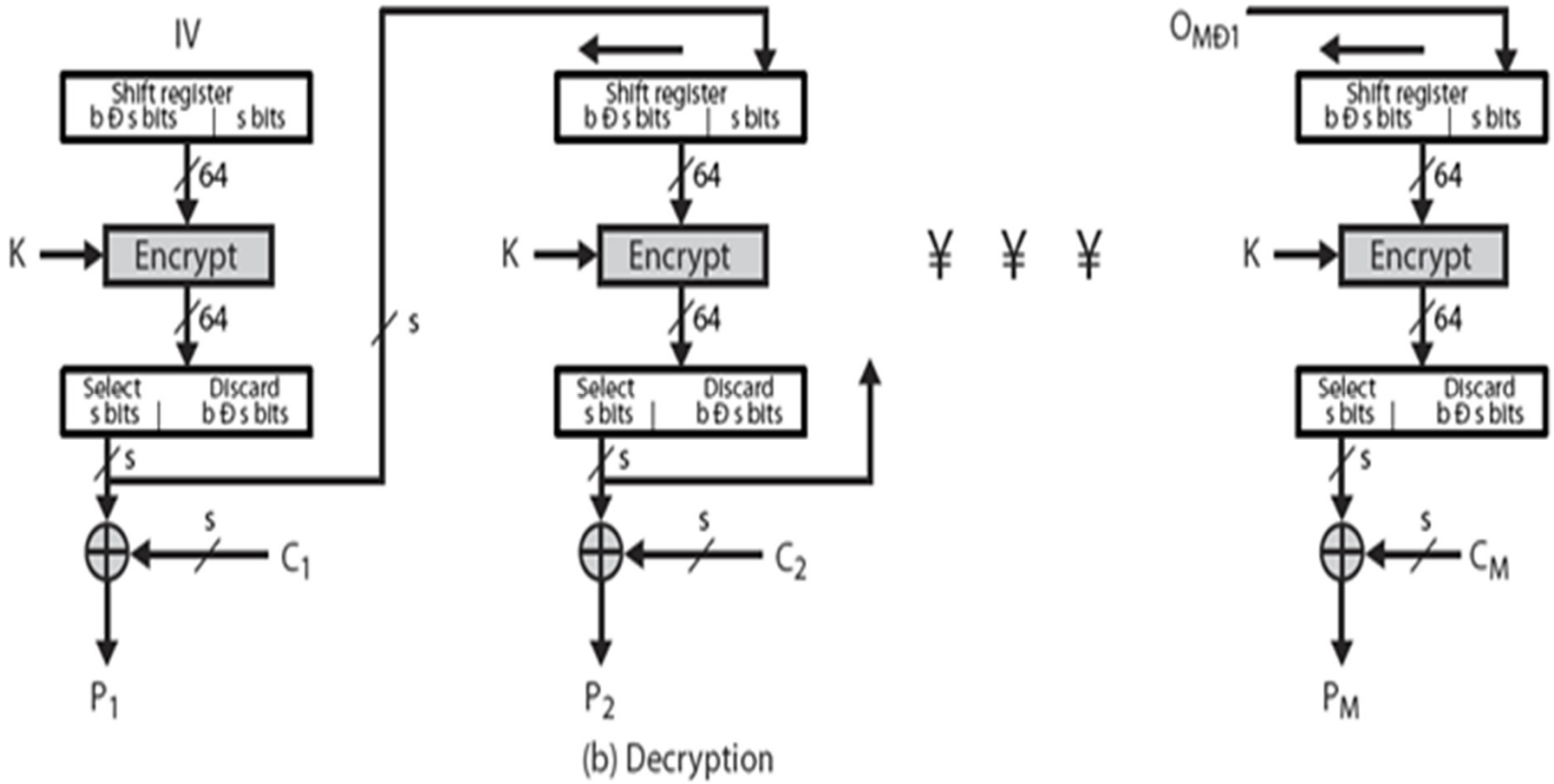
- Message is treated as a stream of bits.
- Output of cipher is added to message.
- Output is then feed back (hence the name).
- Feedback is independent of message
  - $C_i = P_i \oplus E_k(O_{i-1})$, with $O_{-1} = IV$
- So feedback can be computed in advance.

Output Feedback - OFB



(a) Encryption

Output Feedback - OFB



(b) Decryption

Advantages and Limitations of OFB

- Bit errors do not propagate
- Is superficially similar to CFB, but the feedback is from the output of the block cipher and is independent of the message.
- Encryption and decryption of blocks can be done in parallel.

Counter - CTR

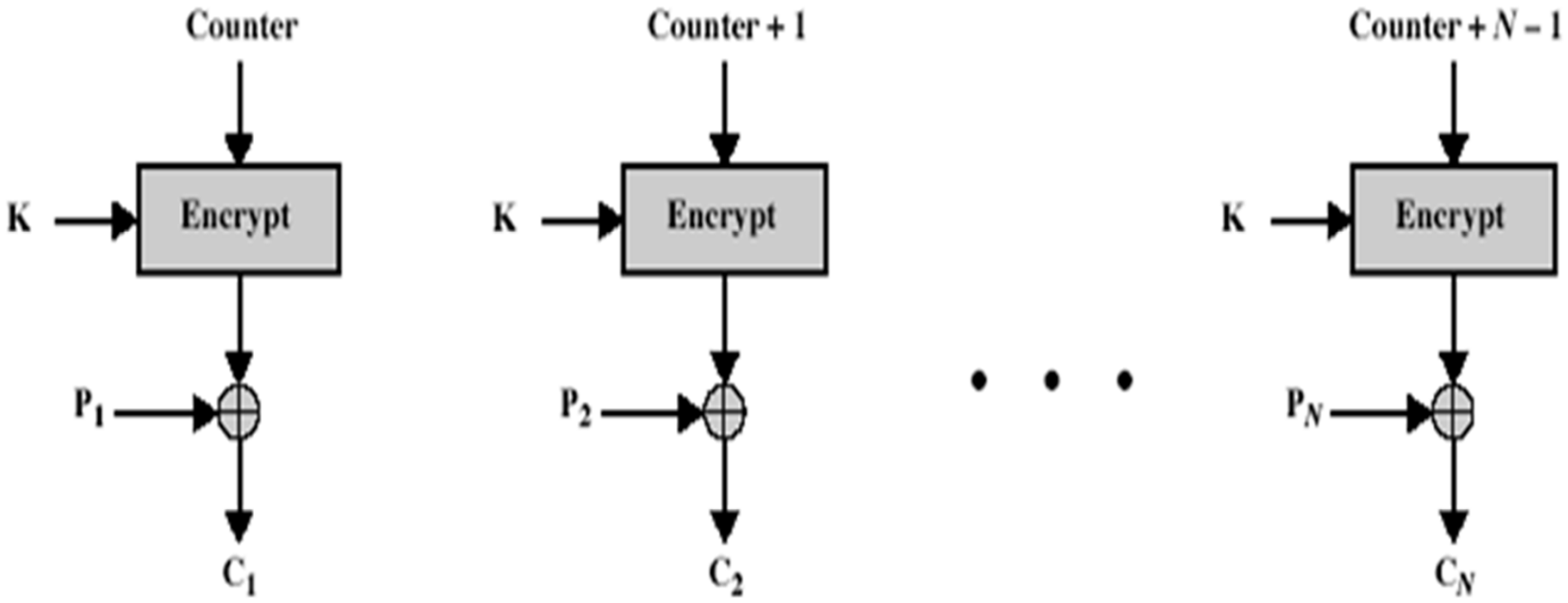
- Similar to OFB but encrypts counter value rather than any feedback value.
- Must have a different counter value for every plaintext block (never reused).

$$C_i = P_i \text{ XOR } O_i$$

$$O_i = DES_{K1}(i)$$

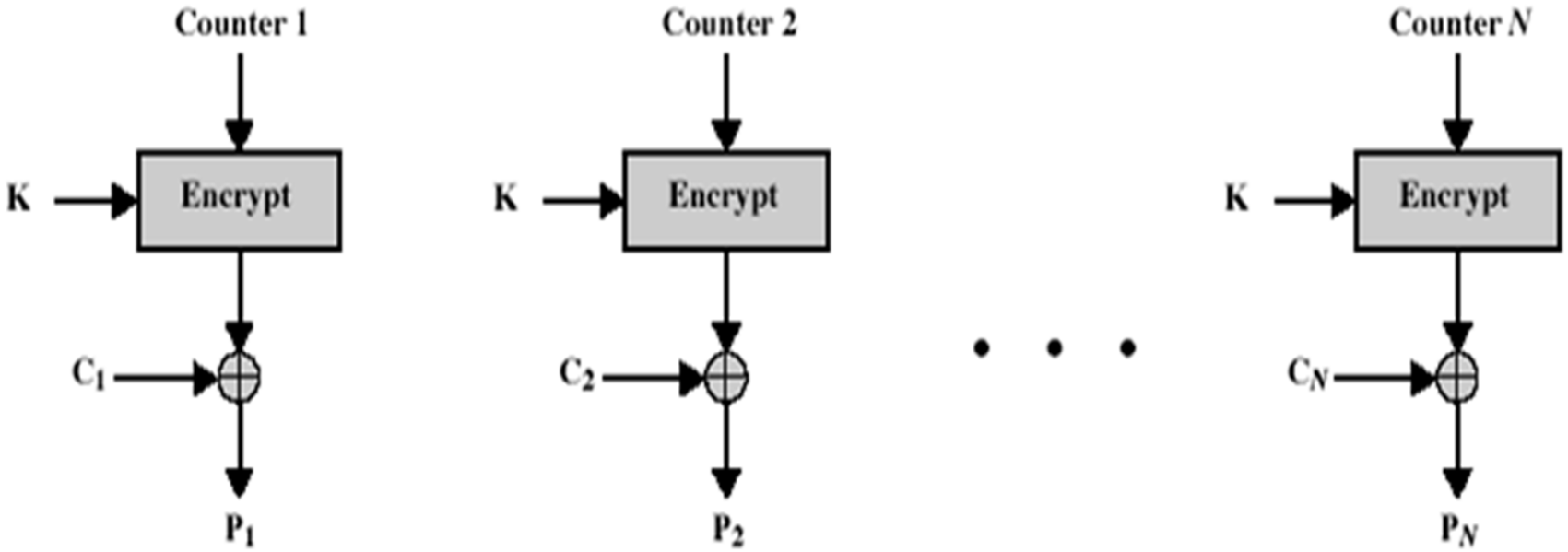
- Uses: high-speed network encryptions.

Counter - CTR



(a) Encryption

Counter - CTR



(b) Decryption

Advantages and Limitations of CTR

- Efficiency
  - can do parallel encryptions in h/w or s/w.
  - can preprocess in advance of need.
- Random access to encrypted data blocks.
- Provable security (good as other modes)
- But must ensure never reuse key/counter values, otherwise could break.

Completeness Effect

Completeness effect means that each bit of the ciphertext needs to depend on many bits on the plaintext.

Weaknesses

Weaknesses in Cipher Design

1. Weaknesses in S-boxes
2. Weaknesses in P-boxes
3. Weaknesses in Key

Weaknesses in Key

Table 6.18 *Weak keys*

Keys before parities drop (64 bits)	*Actual key (56 bits)*
0101 0101 0101 0101	0000000 0000000
1F1F 1F1F 0E0E 0E0E	0000000 FFFFFFF
E0E0 E0E0 F1F1 F1F1	FFFFFFF 0000000
FEFE FEFE FEFE FEFE	FFFFFFF FFFFFFF

Weaknesses in Key

Table 6.19 *Semi-weak keys*

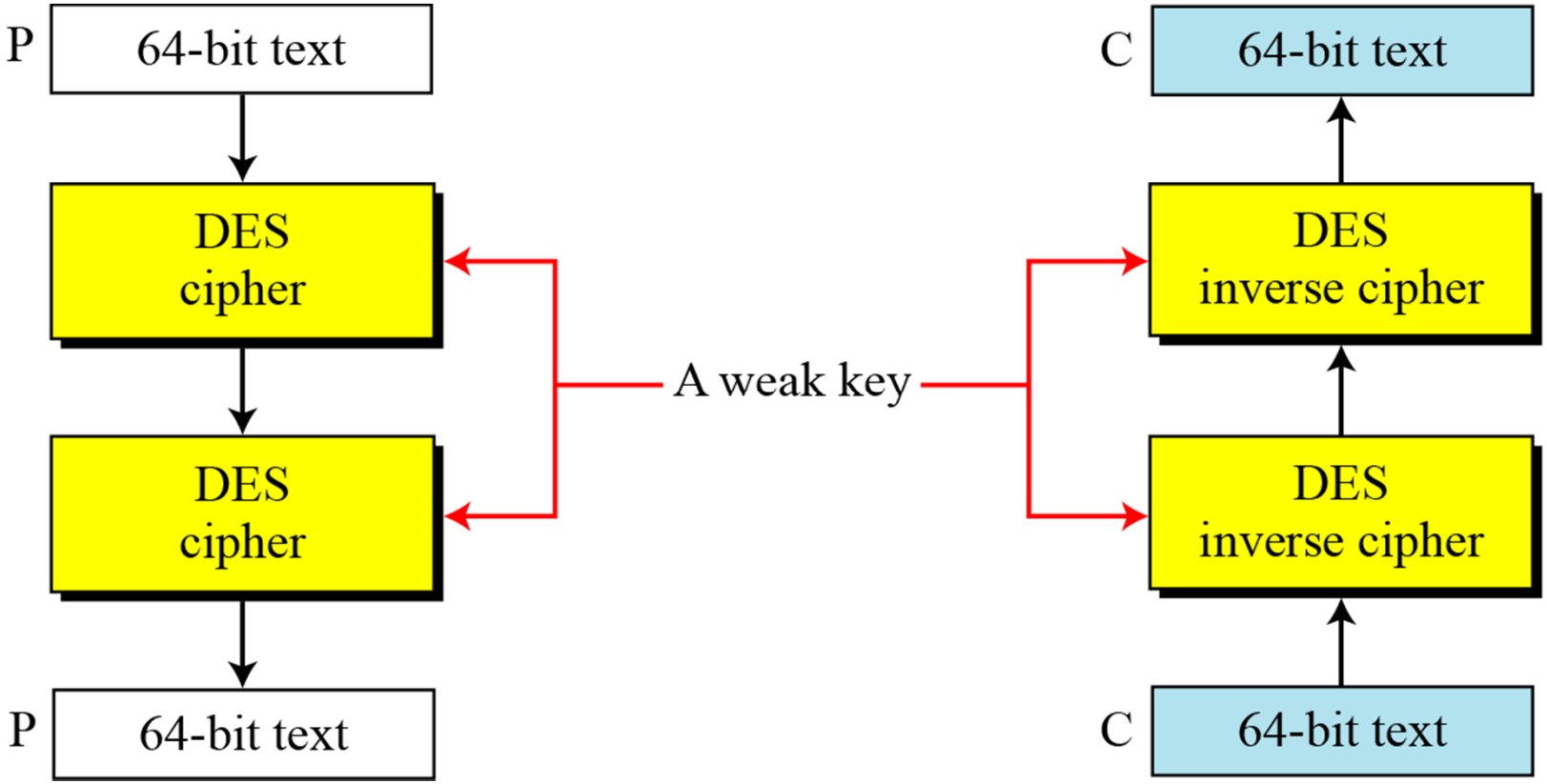
First key in the pair	*Second key in the pair*
01FE 01FE 01FE 01FE	FE01 FE01 FE01 FE01
1FE0 1FE0 0EF1 0EF1	E01F E01F F10E F10E
01E0 01E1 01F1 01F1	E001 E001 F101 F101
1FFE 1FFE 0EFE 0EFE	FE1F FE1F FE0E FE0E
011F 011F 010E 010E	1F01 1F01 0E01 0E01
E0FE E0FE F1FE F1FE	FEE0 FEE0 FEF1 FEF1

Weaknesses in Key

Round key 1	9153E54319BD	6EAC1ABCE642
Round key 2	6EAC1ABCE642	9153E54319BD
Round key 3	6EAC1ABCE642	9153E54319BD
Round key 4	6EAC1ABCE642	9153E54319BD
Round key 5	6EAC1ABCE642	9153E54319BD
Round key 6	6EAC1ABCE642	9153E54319BD
Round key 7	6EAC1ABCE642	9153E54319BD
Round key 8	6EAC1ABCE642	9153E54319BD
Round key 9	9153E54319BD	6EAC1ABCE642
Round key 10	9153E54319BD	6EAC1ABCE642
Round key 11	9153E54319BD	6EAC1ABCE642
Round key 12	9153E54319BD	6EAC1ABCE642
Round key 13	9153E54319BD	6EAC1ABCE642
Round key 14	9153E54319BD	6EAC1ABCE642
Round key 15	9153E54319BD	6EAC1ABCE642
Round key 16	6EAC1ABCE642	9153E54319BD

Weaknesses: Example

Double encryption and decryption with a weak key



Weaknesses: Example

Double encryption and decryption with a weak key

Let us try the first weak key in the Table to encrypt a block two times. After two encryptions with the same key the original plaintext block is created. Note that we have used the encryption algorithm two times, not one encryption followed by another decryption.

Key: 0x0101010101010101
Plaintext: *0x123456788765432 1* Ciphertext: 0x814FE938589154F7

Key: 0x0101010101010101
Plaintext: 0x814FE938589154F7 Ciphertext: *0x123456788765432 1*

Strength:Key Size

- 56-bit keys have $2^{56} = 7.2 \times 10^{16}$ values
- brute force search looks hard
- recent advances have shown that it is possible that:
  - in 1997 on Internet in a few months
  - in 1998 on dedicated h/w (Electronic Frontier Foundation EFF, "DES cracker" machine, $250,000) in a few days
  - in 1999, above combined in 22hrs!

Security

- Now have several analytic attacks on DES
- These utilise some deep structure of the cipher
  - by gathering information about encryptions
  - can eventually recover some/all of the sub-key bits
  - if necessary then exhaustively search for the rest
- Generally these are statistical attacks
  - differential cryptanalysis
  - linear cryptanalysis
  - related key attacks

Brute-Force Attack

- We have discussed the weakness of short cipher key in DES.
- Combining this weakness with the key complement weakness, it is clear that DES can be broken using 2^{55} encryptions.

Differential Cryptanalysis

- One of the most significant advances in cryptanalysis.
- Known by NSA in 70's DES design.
- Murphy, Biham & Shamir published in 90’s.
- Powerful method to analyse block ciphers.
- Used to analyse most current block ciphers with varying degrees of success.

Differential Cryptanalysis

It has been revealed that the designers of DES already knew about this type of attack and designed S-boxes and chose 16 as the number of rounds to make DES specifically resistant to this type of attack.

Differential Cryptanalysis

- Biham & Shamir show Differential Cryptanalysis can be successfull, and requiring 2^{47} chosen plaintexts.
- Although 2^{47} is certainly significantly less than 2^{55}, the need for the adversary to find 2^{47} chosen plaintexts makes this attack of only theoretical interest.
- They also demonstrated this form of attack on a variety of encryption algorithms and hash functions.

Differential Cryptanalysis

- Differential cryptanalysis was known to the IBM DES design team as early as 1974 (as a T attack), and influenced the design of the S-boxes and the permutation P to improve its resistance to it.
- Compare DES's security with the cryptanalysis of an eight-round LUCIFER algorithm which requires only 256 chosen plaintexts, verses an attack on an eight-round version of DES requires 2^{14} chosen plaintexts.

Differential Cryptanalysis

- Compairing the pairs of encryptions.
- Actually, with a known difference in the input, searching for a known difference in output when same subkeys are used.

$$\begin{aligned}\Delta m_{i+1} &= m_{i+1} \oplus m'_{i+1} \\ &= [m_{i-1} \oplus \mathrm{f}(m_i, K_i)] \oplus [m'_{i-1} \oplus \mathrm{f}(m'_i, K_i)] \\ &= \Delta m_{i-1} \oplus [\mathrm{f}(m_i, K_i) \oplus \mathrm{f}(m'_i, K_i)]\end{aligned}$$

Differential Cryptanalysis

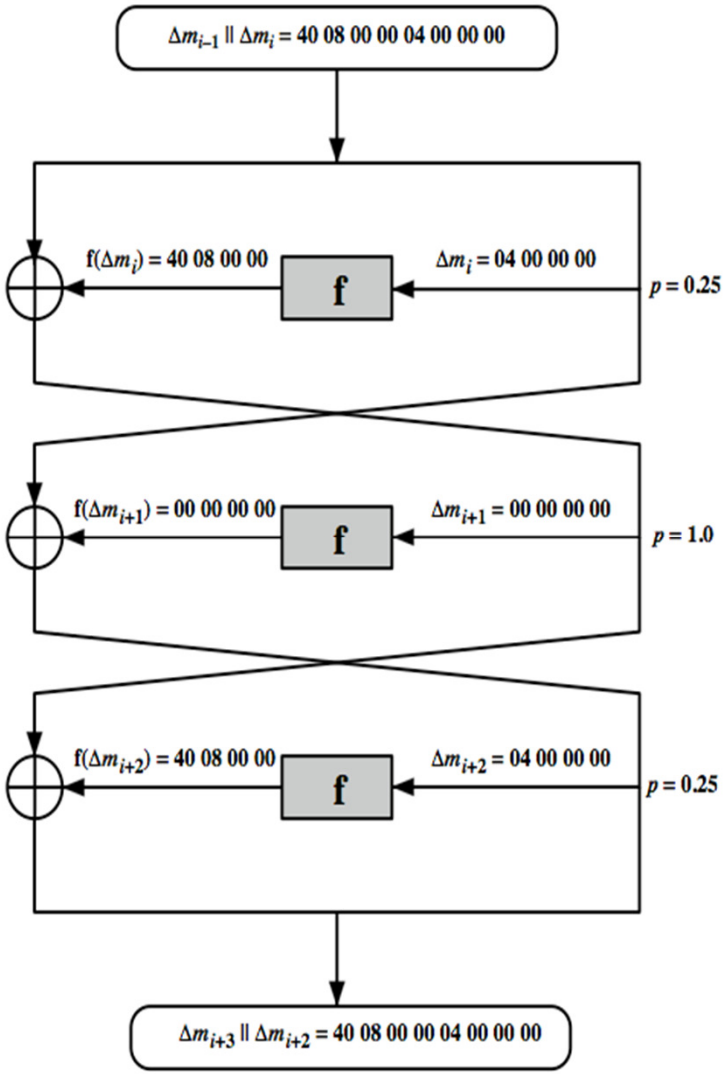
- Have some input difference giving some output difference with probability p.
- If find instances of some higher probability input / output difference pairs occurring.
- Can infer subkey that was used in round.
- Then must iterate process over many rounds (with decreasing probabilities).

Differential Cryptanalysis

Overall, after three rounds the probability that the output difference is as shown is equal to 0.25*1*0.25=0.0625.

Since the output difference is the same as the input, this 3 round pattern can be iterated over a larger number of rounds, with probabilities multiplying to be successively smaller.

Attack on full DES requires an effort on the order of 2^{47} encryptions, requiring 2^{47} chosen plaintexts to be encrypted, with a considerable amount of analysis – in practise exhaustive search is still easier, even though up to 2^{55} encryptions are required for this.



Linear Cryptanalysis

- Linear cryptanalysis is newer than differential cryptanalysis.
- DES is more vulnerable to linear cryptanalysis than to differential cryptanalysis.
- S-boxes are not very resistant to linear cryptanalysis.
- Also a statistical method

Linear Cryptanalysis

- Must be iterated over rounds, with decreasing probabilities.
- Developed by Matsui *et. al.* in early 90's.
- Based on finding linear approximations.
- Can attack DES with 2^{43} known plaintexts, easier but still infeasible in practise.

DES Design Criteria: Revisited

S-Boxes

The design provides confusion and diffusion of bits from each round to the next.

P-Boxes

They provide diffusion of bits.

Number of Rounds

DES uses sixteen rounds of Feistel ciphers. The ciphertext is thoroughly a random function of plaintext and ciphertext.

DES Design Criteria: Revisited

- as reported by Dan Coppersmith
- 7 criteria for S-boxes provide for
  - non-linearity,
  - resistance to differential cryptanalysis,
  - good confusion.
- 3 criteria for permutation P provide for
  - increased diffusion.

DES Design Criteria: Revisited

- Basic principles still like Feistel's in 1970's
- Number of rounds
  - more is better, exhaustive search best attack
- Function f:
  - provides "confusion", is nonlinear, avalanche
  - have issues of how S-boxes are selected
- Key schedule
  - complex subkey creation, key avalanche

Multiple Encryption

- In 2001, NIST published the Advanced Encryption Standard (AES) to replace DES.
- But users in commerce and finance are not ready to give up on DES.
- As a temporary solution to DES's security problem, one may encrypt a message (with DES) multiple times using multiple keys:
  - 2DES is not much securer than the regular DES
  - So, 3DES with either 2 or 3 keys is used

3DES-2

3DES with 2 keys

- A straightforward implementation would be :

$$c := E_{k_1}\left(E_{k_2}\left(E_{k_1}(m)\right)\right)$$

- In practice : $c := E_{k_1}\left(D_{k_2}\left(E_{k_1}(m)\right)\right)$
  - Also referred to as EDE encryption
- Reason : if $k_1 = k_2$, then 3DES = 1DES.
  Thus, a 3DES software can be used as a single-DES.
- Standardized in ANSI X9.17 & ISO 8732.
- No practical attacks are known.

3DES-3

3DES with 3 keys

- Encryption: $c := E_{k_3}\left(D_{k_2}\left(E_{k_1}(m)\right)\right)$.
- If $k_1 = k_3$, it becomes 3DES with 2 keys.
- If $k_1 = k_2 = k_3$, it becomes the regular DES.
- So, it is backward compatible with both 3DES with 2 keys and the regular DES.
- Some internet applications adopt 3DES with three keys; e.g. PGP and S / MIME.

Thank you so much for
your attention.