Network and Information Security
Lecture 15

B.Tech. Computer Engineering
Sem. VI.

Prof. Mrudang T. Mehta
Associate Professor
Computer Engineering Department
Faculty of Technology,
Dharmsinh Desai University, Nadiad

DES round contains 3 types of elements

- Self invertible like $\oplus$

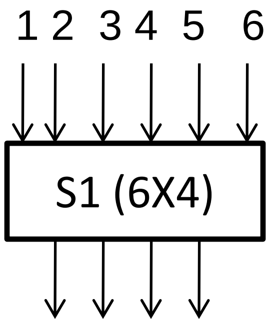
$$z = x \oplus y$$

$$x = z \oplus y$$

$$y = z \oplus x$$

- Invertible like P-Box
- Non-invertible like S-box (S1 to s8)

S-boxes are non-invertible



	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	10	03	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-box1

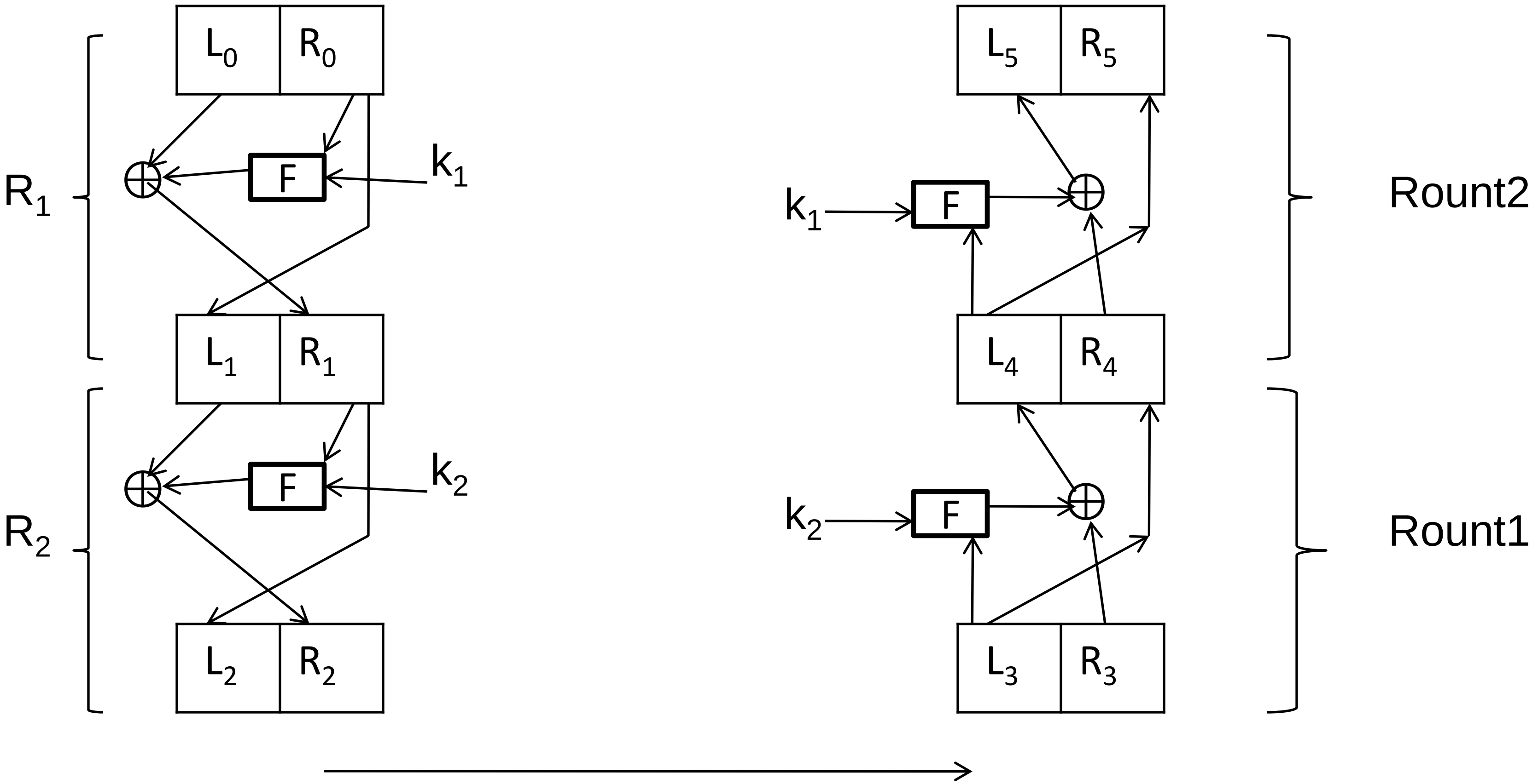
$0\underline{1011}0 \longrightarrow (12)_d$ [For 12 we can get multiple input values]

$1\underline{1001}0 \longrightarrow (12)_d$

- Round function contains element/components
  - Self invertible
  - Invertible
  - Non invertible
- Feistel structure cipher
  - If there are invertible and non-invertible elements present in the structure (with decryption possible)
  - Example- DES (in Function F, both invertible and non-invertible elements present)

$L_i = R_{i-1},$
$R_i = L_{i-1} \oplus F(R_{i-1},k_i)$
F is Round function

How decryption is possible with non-invertible element present?



Encryption Decryption

Show: If $L_2 = L_3$ and $R_2 = R_3$ then

(1) $L_4=L_1$, $R_4=R_1$

(2) $L_5=L_0$, $R_5=R_0$

If L_2,R_2 is received without error, then using above decryption we can decrypt.

Proof:

From the encryption part, we can write the following equations.

$L_1 = R_0$ -----------------------------(1)

$R_1 = L_0 \oplus F(R_0, k_1)$ -----------------------------(2)

$L_2 = R_1$ -----------------------------(3)

$R_2 = L_1 \oplus F(R_1, k_2)$ -----------------------------(4)

- From the decryption design we can write following:

$R_4 = L_3$ ----------------------------(5)

$L_4 = R_3 \oplus F(L_3, k_2)$ ----------------------------(6)

$R_5 = L_4$ ----------------------------(7)

$L_5 = R_4 \oplus F(L_4, k_1)$ ----------------------------(8)

If $L_2 = L_3$ and $R_2 = R_3$ then $R_4 = R_1$, $L_4 = L_1$

L H S = R_4

$= L_3$ (....from (5))

$= L_2$ (Given)

$= R_1$ (....from (3))

= R H S

$R_4 = R_1$ ....(9)

$L_4 = L_1$ ....(10)

L H S = L_4

$= R_3 \oplus F(L_3,k_2)$ (.. from (6))

$= R_2 \oplus F(L_2,k_2)$ (Given)

$= L_1 \oplus F(R_1,k_2) \oplus F(L_2,k_2)$ (...from(4))

$= L_1 \oplus F(R_1,k_2) \oplus F(R_1,k_2)$ (...from(3))

$= L_1 \oplus 0$ $(a \oplus a = 0)$

$= L_1$ $(a \oplus 0 = a)$

= R.H.S.

If $L_2 = L_3$ and $R_2 = R_3$ then $R_5 = R_0$, $L_5 = L_0$

L H S = R_5

= L_4 (....from (7))

= L_1 (...from 10)

= R_0 (....from (1))

= R H S

L H S = L_5

= $R_4 \oplus F(L_4,k_1)$ (.. from (8))

= $R_1 \oplus F(L_1,k_1)$ (from 9 ,10)

= $L_0 \oplus F(R_0,k_1) \oplus F(L_1,k_1)$

(...from(2))

= $L_0 \oplus F(R_0,k_1) \oplus F(R_0,k_1)$

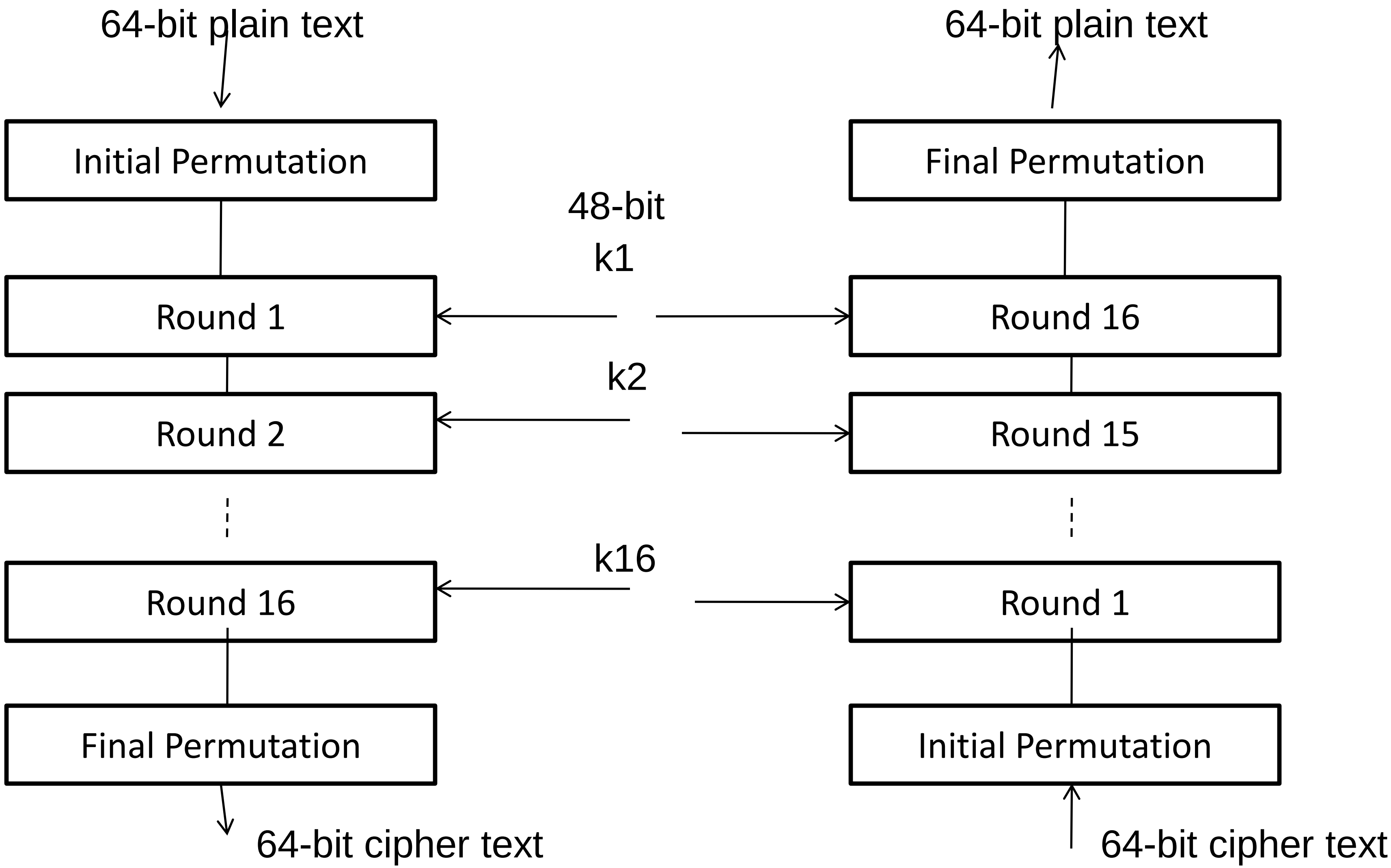
(...from(1))

= $L_0 \oplus 0$ ($a \oplus a = 0$)

= L_0 ($a \oplus 0 = a$)

= R.H.S.

DES cipher and reverse cipher



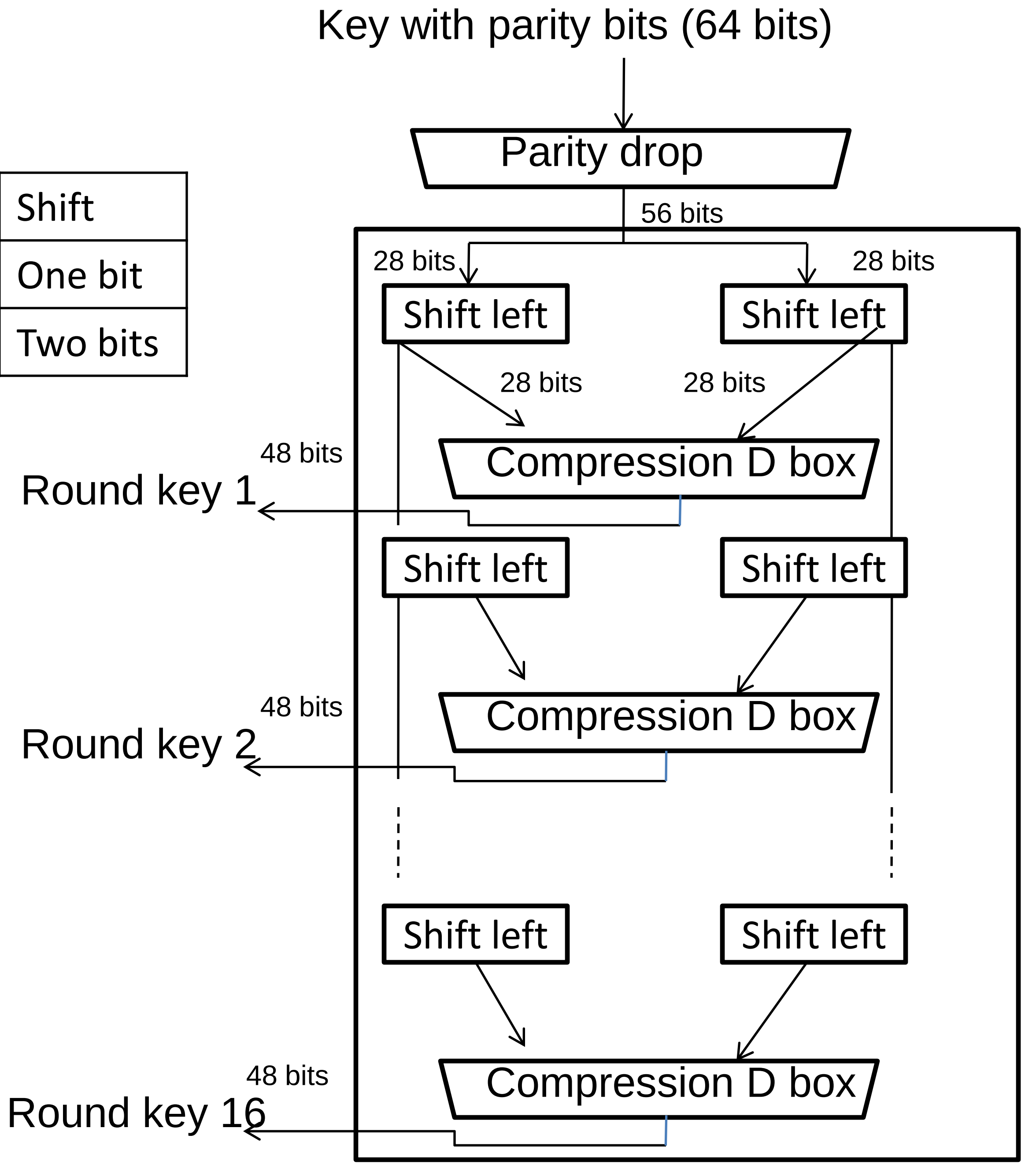
Alternative approach

- In the first approach round 16 is different from other rounds, there is no swapper in this round.
- This is needed to make the last mixer in the cipher and the first mixer in the reverse cipher aligned.
- We can make all 16 rounds the same by one swapper to the 16^{th} round and add an extra swapper after that (two swapper cancels the effect of each other).

Key Generation

- The round-key generator creates sixteen 48-bit keys out of a 56-bit cipher key.
- However, the cipher key is normally given as 64-bit in which 8 extra bits are the parity bits, which are dropped before the actual key-generation process.

Rounds	Shift
1,2,9,16	One bit
Others	Two bits



Key
Generation

- Parity drop
- The preprocess before key expansion is a compression transposition step that we call parity bit drop.
- It drops the parity bits (8,16,24,32,40,48,56,64) from the 64-bit key and permutes the test of the bits according to table.
- The remaining 56-bit value the actual cipher key which is used to generate round keys.

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

Parity bit drop table

- Shift left
  - After the straight permutation, the key is divided into two 28-bit parts.
  - Each part is shifted left (circular shift) one or two bits.
  - In round 1,2,9 and 16, shifting is one bit; in the other rounds it is two bits.
  - The two parts are then combined to form a 56-bit part.

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

- Compression Permutation (Compression D-box)

The compression D-box changes the 58-bits to 48-bits, which are used as a key for a round.

14	17	11	24	01	05	03	28
15	06	21	10	23	19	12	04
26	08	16	07	27	20	13	02
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32

Note: (In a book, P-box is named as D-box, Both are same)

- Example 1 We choose a random plaintext block and a random key, and determine what the ciphertext block would be (all in hexadecimal):

Plain text: 123456ABCD132536
Cipher text: C0B7A8D05F3A829C

Key: AABB09182736CCDD

Plain text: 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
Round9	308BEE97	10AF9037	84BB4473DCCC
Round10	10AF9037	6CA6CB20	02765708B5BF
Round11	6CA6CB20	FF3C485F	6D5560AF7CA5
Round12	FF3C485F	22A5963B	C2C1E96A4BF3
Round13	22A5963B	387CCDAA	99C31397C91F
Round14	387CCDAA	BD2DD2AB	251B8BC717D0
Round15	BD2DD2AB	CF26B472	3330C5D9A36D
Round16	19BA9212	CF26B472	181C5D75C66D
After combination: 19BA9212CF26B472			
Cipher text: C0B7A8D05F3A829C (after final permutation)			