DES (Data Encryption Algorithm):

Developed by NIST (National Institute of standard technology)

- Symmetric key Block Cipher (64 bit)
- It is implementation of Feistel cipher Model (16 Rounds)
- DES has 16 Round.
- Block size is 64 bit. (plaintext & ciphertext)
- Key length is 56 bit. (original key is 64 bit but 8 bit are not used)

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23

7 x 8 = 56 bit

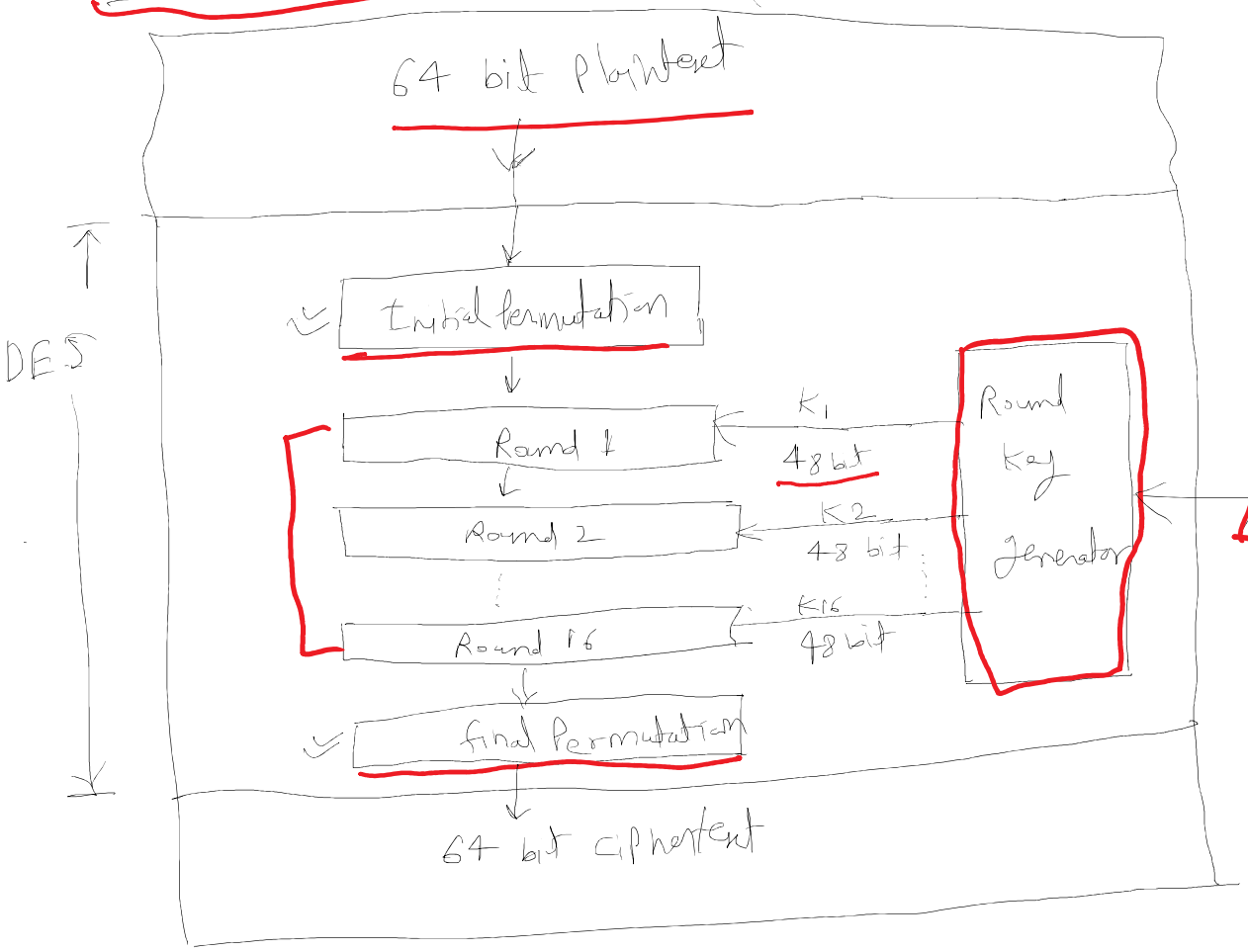
7 bit — lost — 7 bit — lost

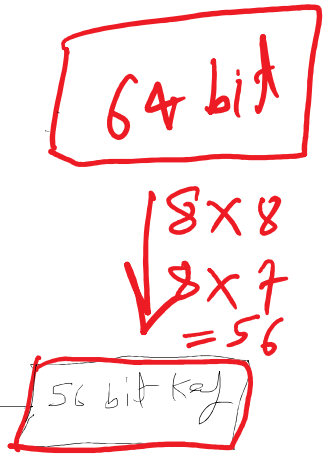
---

main functionality of DES:

1. Round function
2. key schedule
3. Additional Processing (Initial & final permutation)

Structure of DES:

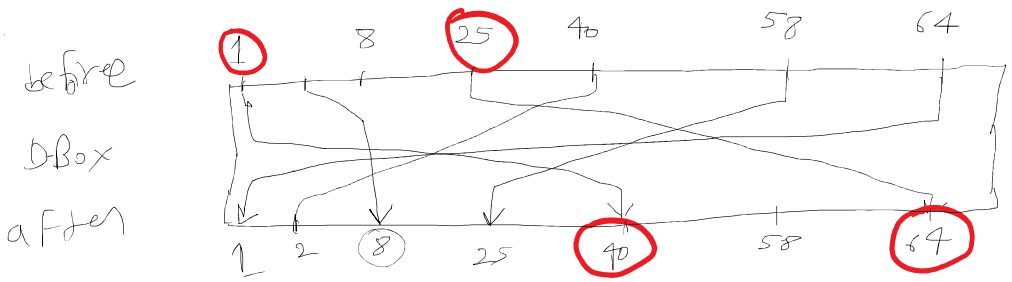




① Initial permutation:

Initial & final permutation are straight permutation Box used to apply substitution operation on plaintext/intermediate text. (D-Box) (straight)

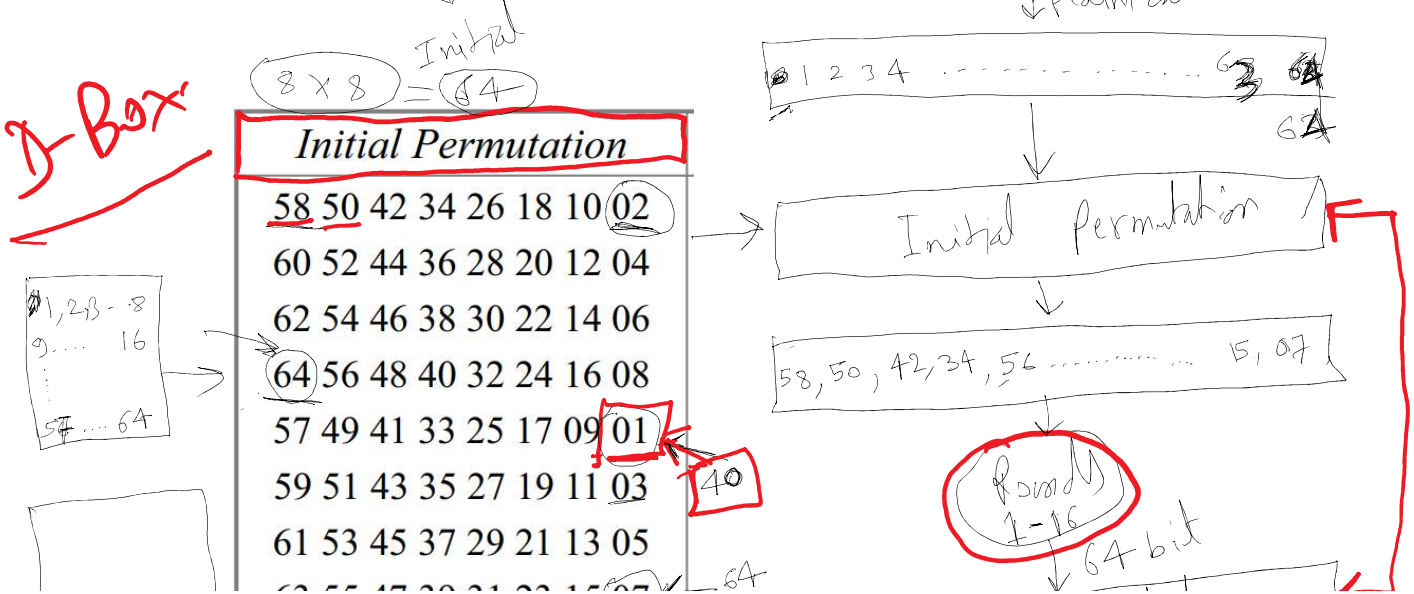
apply substitution operation on plaintext / intermediate text. [D-Box] (straight)



D-Box (output)

	1	3	-	8	...	25	-	40	...	64
	40			2		58		1		25

Example of permutation & find permutation table



61 53 45 37 29 21 13 05
63 55 47 39 31 23 15 07

64

Final permutation

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

1-16

64 bit

final permutation

ciphertext

64

1

functionality of Round in DES based on feistal structure

64 bit

32 bits

L_{i-1}

32 bits

R_{i-1}

L_0 (32 bit)

R_0 (32 bit)

48 bit

Round 1

Mixer

$f(R_{i-1}, K_i)$

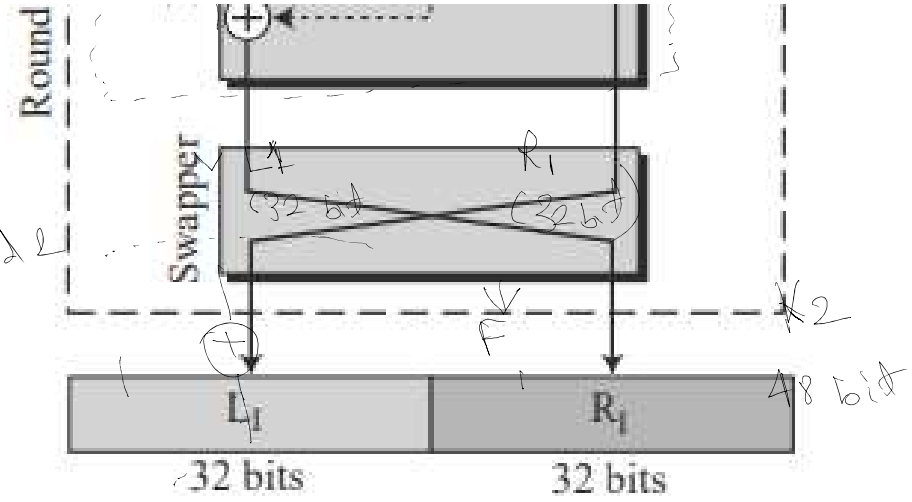
K_1

Round

Block features

1. Block size (n)
2. No. of Round (4)
3. Encryption size method (48)

Round 1: Input (L_0, R_0, K_1)
output (L_1, R_1)



A round in DES (encryption site)

$R_1 = f(R_0, K_1) \oplus L_0$

$L_1 = R_0$

Round 2 : Input (L_1, R_1, K_2)
output (L_2, R_2)

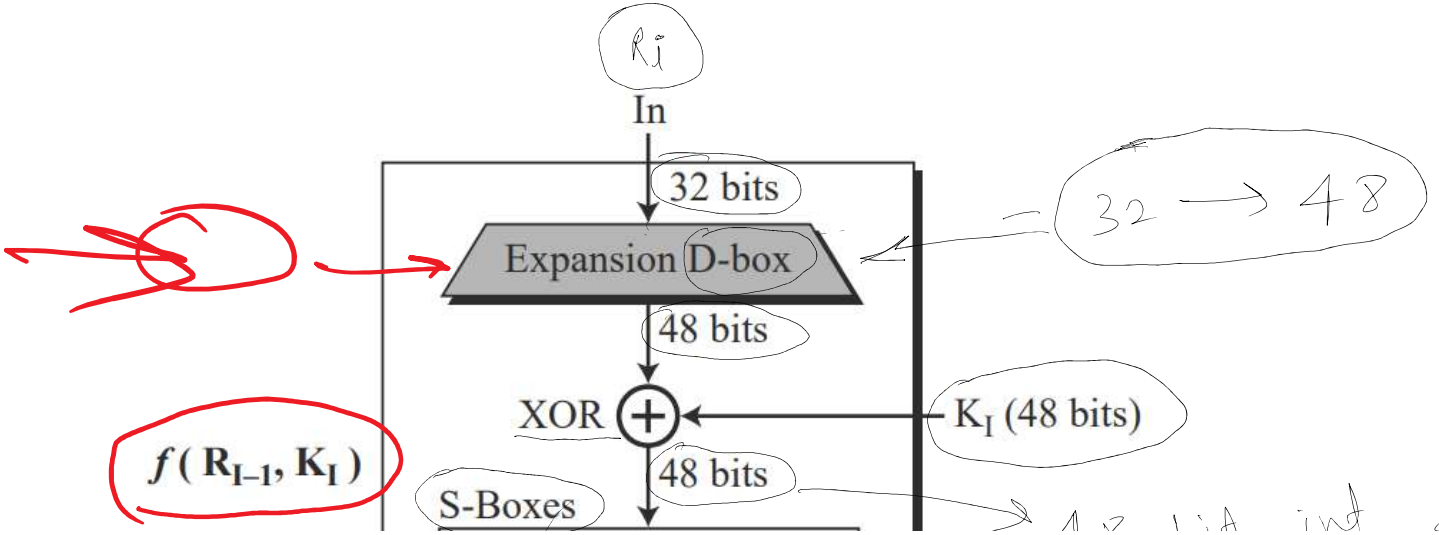
$R_2 = f(R_1, K_2) \oplus L_1$

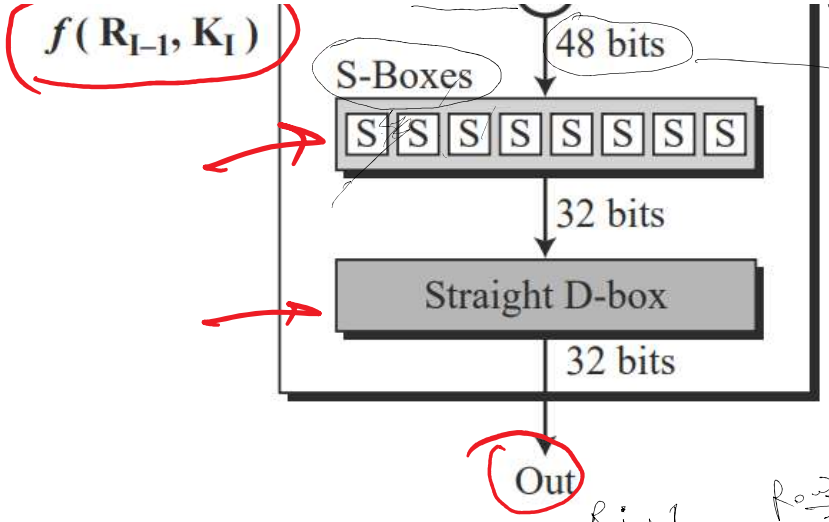
$L_2 = R_1$

Heart of DES →

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

$L_i = R_{i-1}$





A 8 bit int 8 subblock

Row: 01
col: 0100

1 Block (6 bit)

001001

S-Box

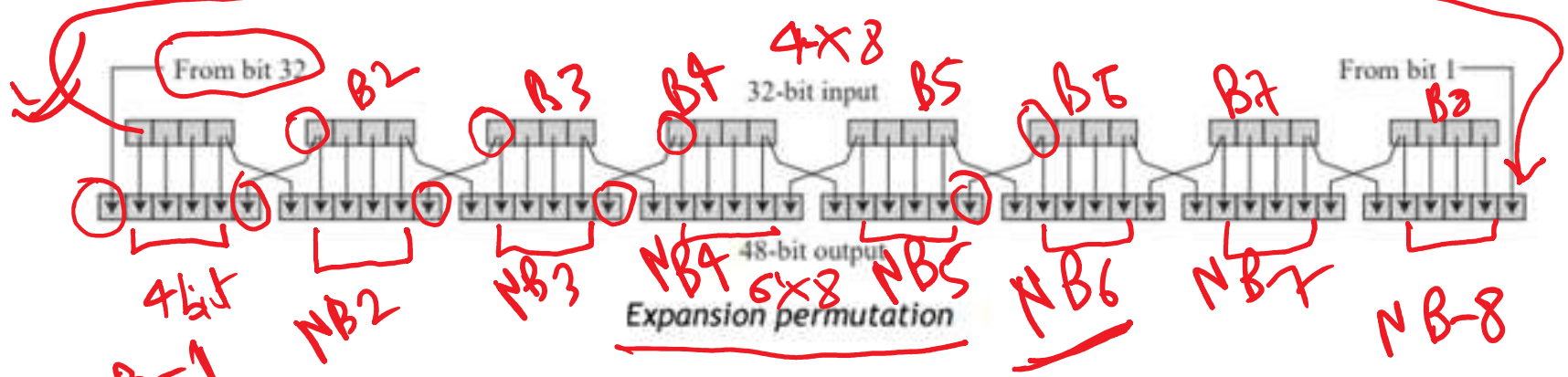
1 Block 4 bit

4X8 = 32

4X16 Col

Row

$f(R_{i-1}, K_i)$ R_{i+1}



Expansion D-box table

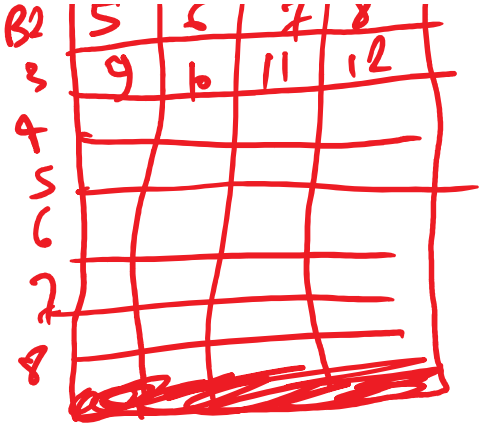
NB1	32	01	02	03	04	05
NB2	04	05	06	07	08	09
	08	09	10	11	12	13

32 = 4X8

B1	1	2	3	4
B2	5	6	7	8
B3	9	10	11	12

NB2

04	05	06	07	08	09
08	09	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	01

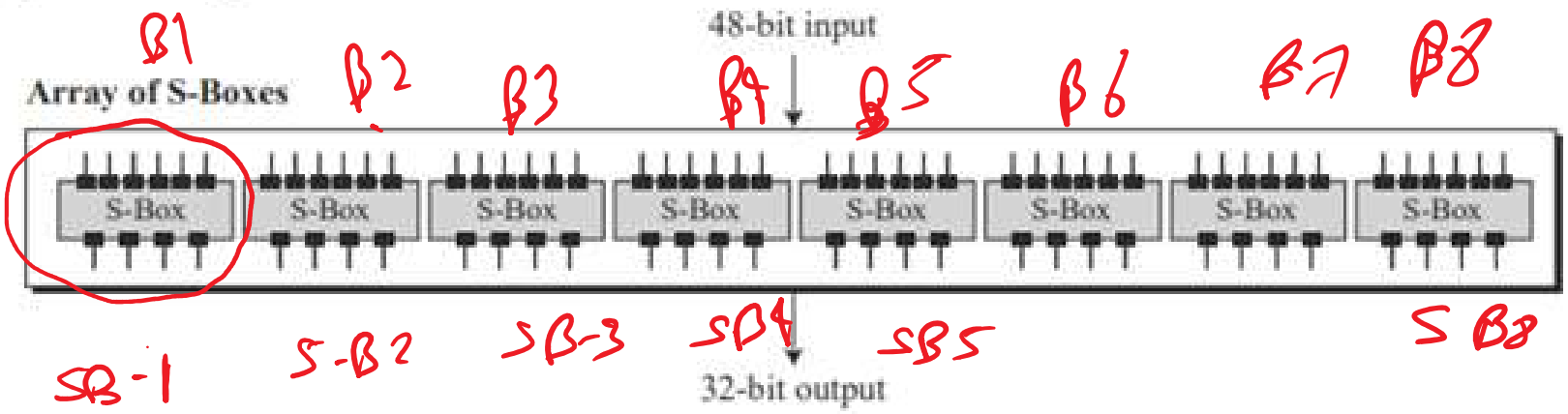


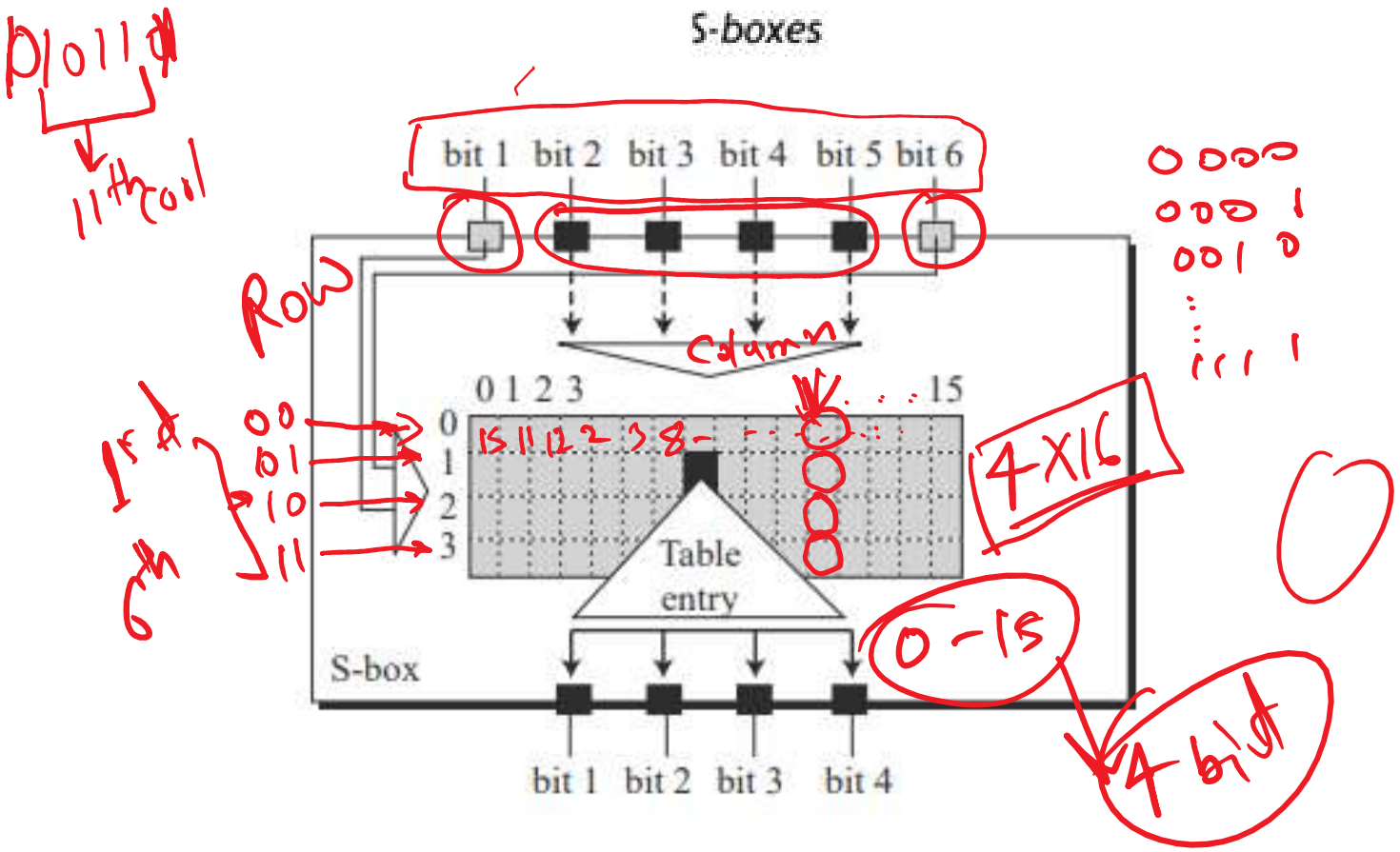
D-Box

Expansion D-Box

48 bit

6X8


48-bit input
Array of S-Boxes
S-Box
S-Box
S-Box
S-Box
S-Box
S-Box
S-Box
S-Box
32-bit output

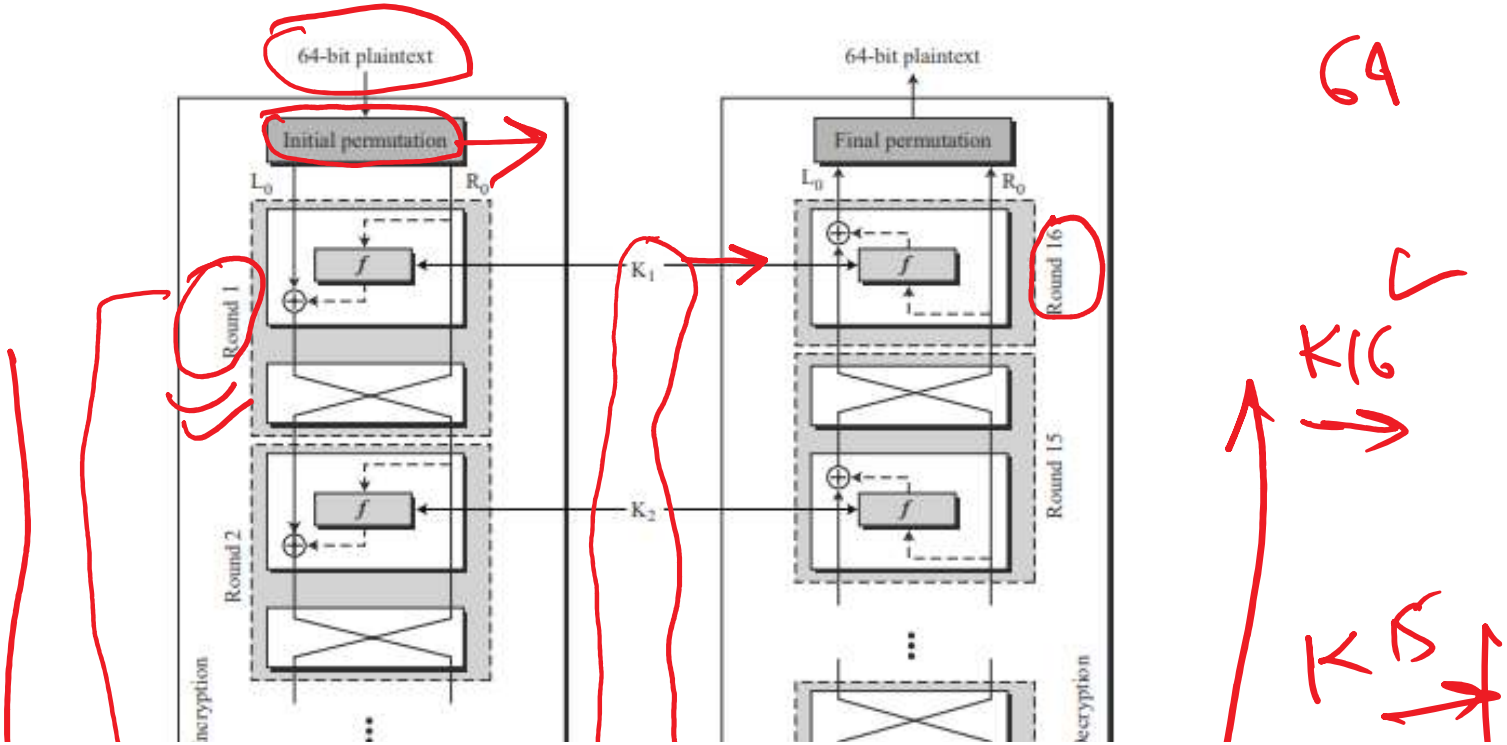

S-boxes
bit 1 bit 2 bit 3 bit 4 bit 5 bit 6
Row
Column
0 1 2 3
. . . 15
00
01
10
11
0
1
2
3
Table entry
S-box
bit 1 bit 2 bit 3 bit 4

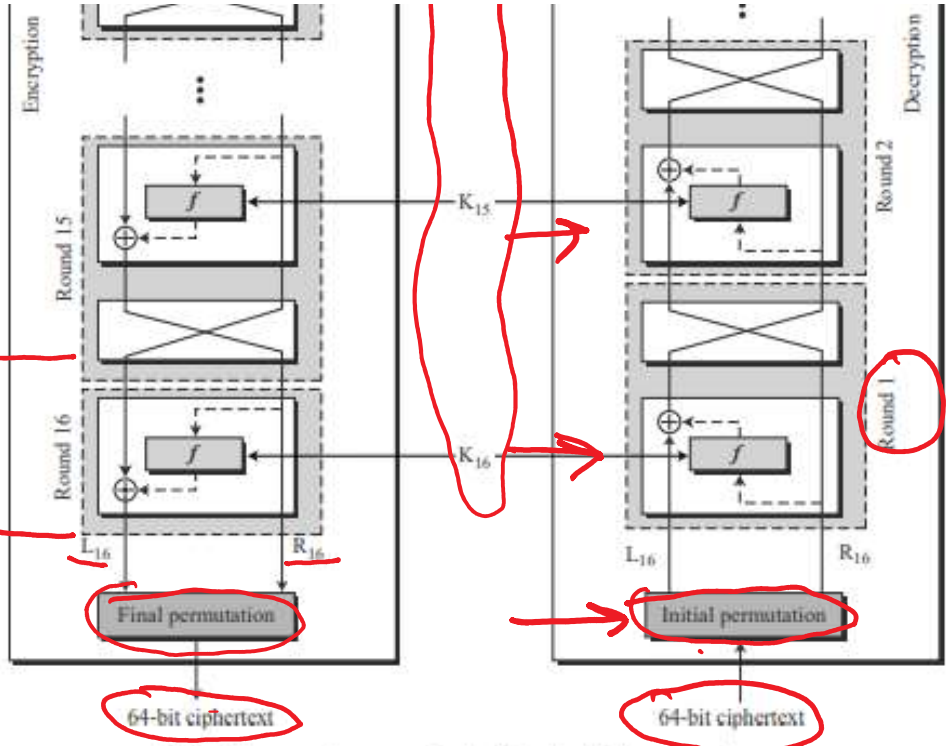
S-box 1

	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-box 2

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	15	01	08	14	06	11	03	04	09	07	02	13	12	00	05	10
1	03	13	04	07	15	02	08	14	12	00	01	10	06	09	11	05
2	00	14	07	11	10	04	13	01	05	08	12	06	09	03	02	15
3	13	08	10	01	03	15	04	02	11	06	07	12	00	05	14	09





DES cipher and reverse cipher for the first approach