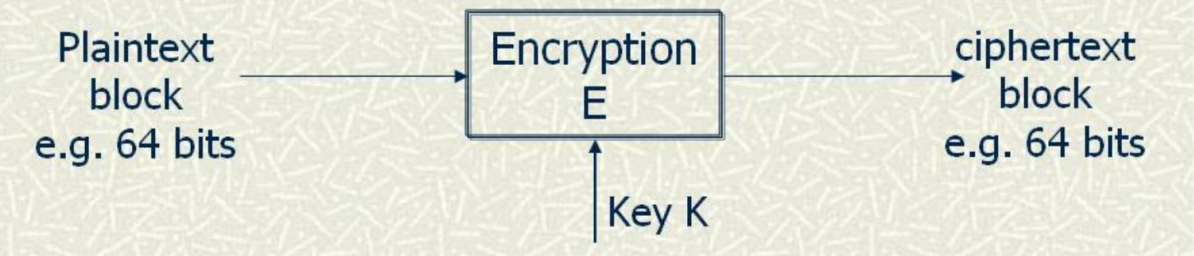
Block Ciphers

Block Ciphers definition

- Encrypt a block of input to a block of output
- Typically, the two blocks are of the same length
- Most symmetric key systems block size is 64.
- Many block ciphers have a Feistel structure.
- Such a structure consists of a number of identical rounds of processing.
- In each round, a substitution is performed on one half of the data being processed,
- followed by a permutation that interchanges the two halves.
- The original key is expanded so that a different key is used for each round.



Iterated Block Cipher - A block cipher that "iterates a fixed number of times of another block cipher, called round function, with a different key, called round key, for each iteration".

Diffusion – dissipates statistical structure of plaintext over bulk of ciphertext.

Confusion – makes relationship between ciphertext and key as complex as possible.

Feistel Cipher:

Feistel ciphers are a special class of iterated block ciphers , where the ciphertext is calculated from the plaintext by repeated application of the same transformation or round function. In a Feistel cipher, the text being encrypted is split into two halves. The round function f is applied to one half using a subkey and the output of f is exclusive-ored with the other half. The two halves are then swapped. Each round follows the same pattern except for the last round where there is no swap.

A nice feature of a Feistel cipher is that encryption and decryption are structurally identical, though the subkeys used during encryption at each round are taken in reverse order during decryption.

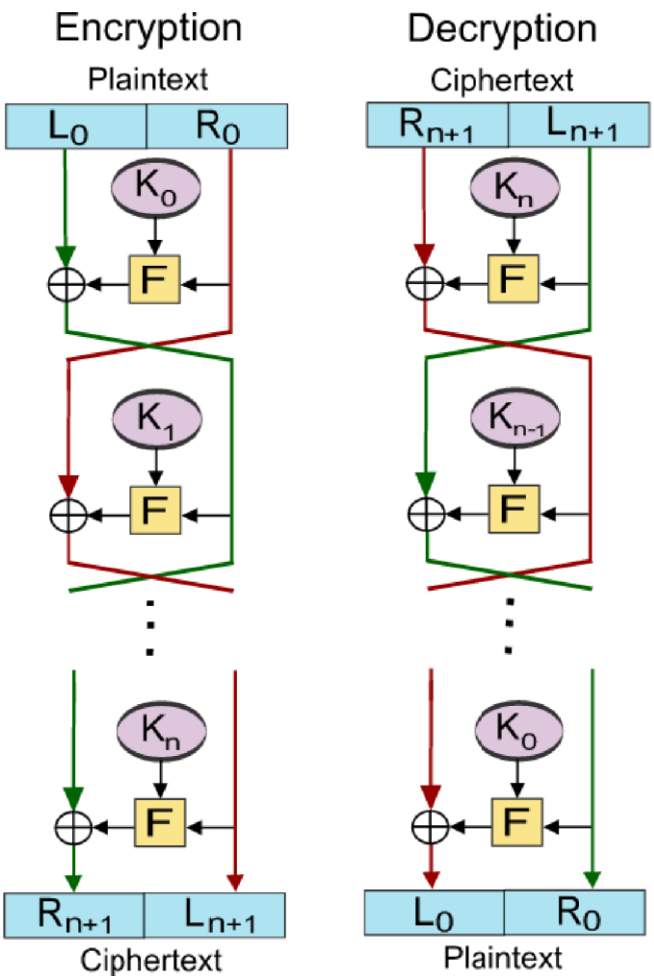
- Several block ciphers are based on the structure proposed byFeistel in 1973.
- A Feistel Network is fully specified given

  - the block size: n = 2w

  - number of rounds: d

  - d round functions $f_1, \ldots, f_d: \{0,1\}^w \rightarrow \{0,1\}^w$

- Used in DES and many other block ciphers.



Construction details

Let F be the round function and let $K_0, K_1, \ldots, K_n$ be the sub-keys for the rounds $0, 1, \ldots, n$ respectively.

Then the basic operation is as follows:

Split the plaintext block into two equal pieces, (L_0, R_0)

For each round , $i = 0, 1, \ldots, n$ compute

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

Then the ciphertext is (R_{n+1}, L_{n+1}).

Decryption of a ciphertext (R_{n+1}, L_{n+1}) is accomplished by computing for $i = n, n-1, \ldots, 0$

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

Then (L_0, R_0) is the plaintext again.

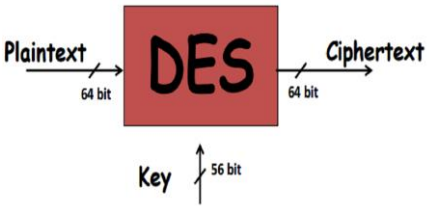
The diagram illustrates both encryption and decryption. Note the reversal of the subkey order for decryption; this is the only difference between encryption and decryption.

Data Encryption Standard (DES)

The Data Encryption Standard (DES), known as the Data Encryption Algorithm (DEA) by ANSI and the DEA-1 by the ISO, has been most widely used block cipher in world, especially in financial industry.

DES Features:

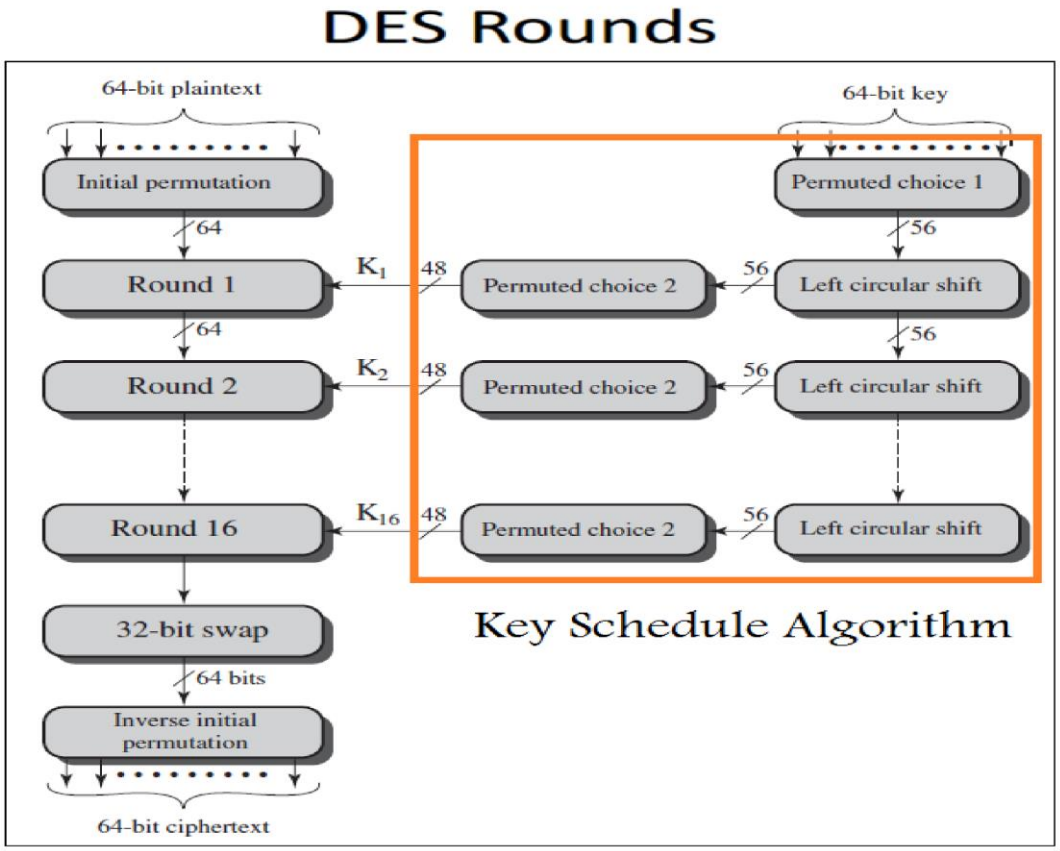
- It encrypts 64-bit data (Block size = 64 bits).
- Key size = 56 bits (in reality, 64 bits, but 8 are used as parity-check bits for error control slide).
- Number of rounds = 16. (a 16-round Feistel cipher with block size of 64 bits.)
- 16 intermediary keys, each 48 bits.
- The algorithm is a combination of the two basic techniques of encryption: confusion and diffusion.



Key length in DES

- In the DES specification, the key length is 64 bit.
- 8 bytes; in each byte, the 8th bit is a parity-check bit

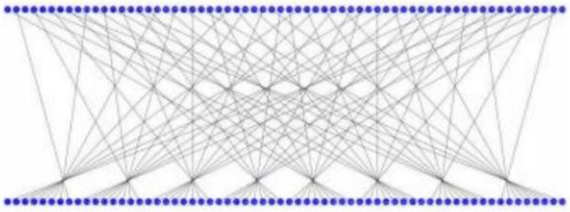
- Each parity-check bit is the XOR of the previous 7 bits



General Depiction of DES Encryption Algorithm

Initial Permutation (IP)

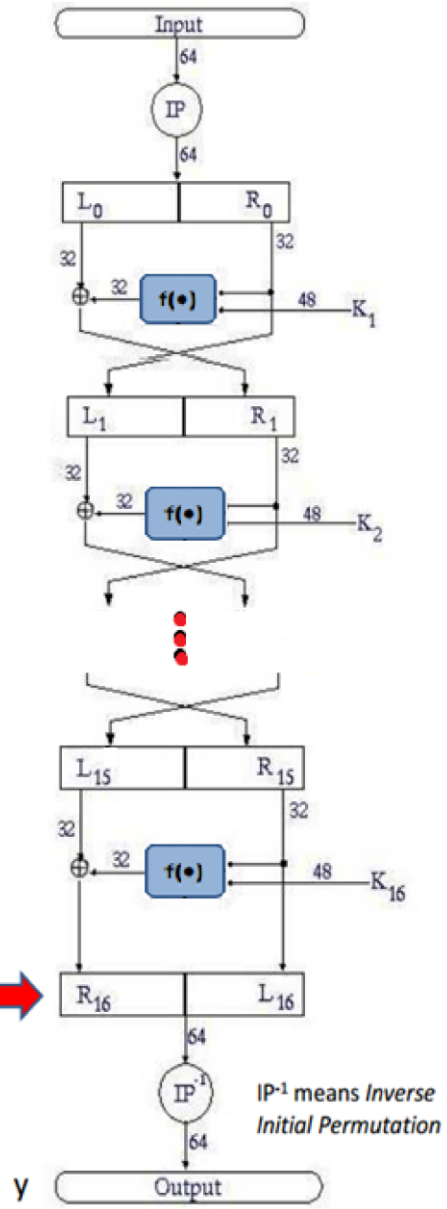
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7



- This table specifies the input permutation on a 64-bit block.
- The meaning is as follows:
  The first bit of the output is taken from the 58th bit of the input; the second bit from the 50th bit, and so on, with the last bit of the output taken from the 7th bit of the input.
- This information is presented as a table for ease of presentation:
  it is a vector, not a matrix.

DES Rounds in Details

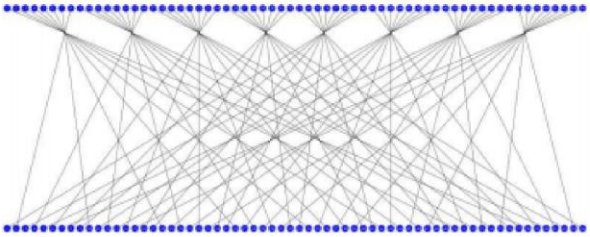
- $IP(x) = L_0R_0$
- $L_i = R_{i-1}$
- $R_i = L_{i-1} \oplus f(R_{i-1}, K_i)$
- $y = IP^{-1}(R_{16}L_{16})$



- Note that, as usual:
  - $R_{16} = L_{15} \oplus f(R_{15}, K_{16})$
  - $L_{16} = R_{15}$
- ... but they are *switched* in the pre-output

Final Permutation (IP^{-1})

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25



The final permutation is the inverse of the initial permutation; the table is interpreted similarly.

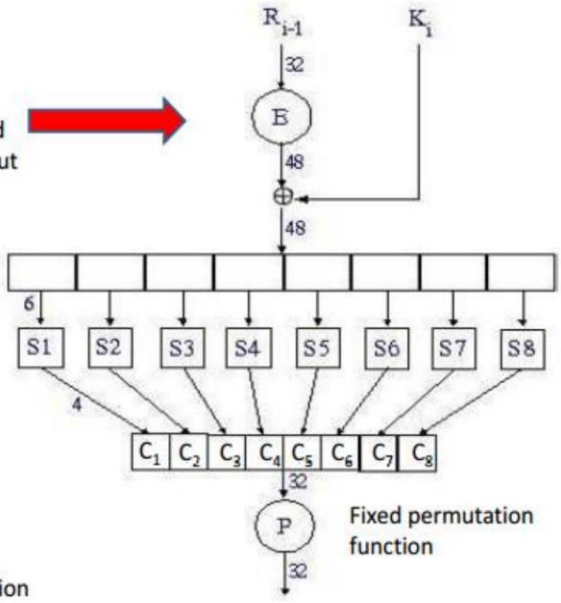
That is, the output of the Final Permutation has bit 40 of the preoutput block as its first bit, bit 8 as its second bit, and so on, until bit 25 of the preoutput block is the last bit of the output.

DES "f(•)" Function

E is an *expansion function* which takes a block of 32 bits as input and produces a block of 48 bits as output

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

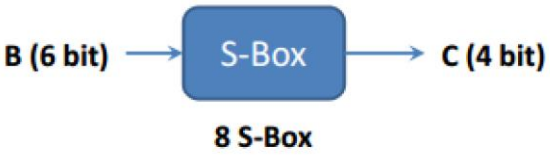
16 bits appear twice, in the expansion



S-boxes

- S-boxes are the only *non-linear* elements in DES design

Each of the unique selection functions $S_1, S_2, \ldots, S_8$, takes a 6-bit block as input and yields a 4-bit block as output



- S = matrix 4x16, values from 0 to 15
- B (6 bit long) = $b_1b_2b_3b_4b_5b_6$
  - b_1b_6 ➔ r = row of the matrix (2 bits: 0,1,2,3)
  - $b_2b_3b_4b_5$ ➔ c = column of the matrix (4 bits:0,1,…15)
- C (4 bit long) = Binary representation of S(r, c)

Example (S1)

Row #	S_1	1	2	3	…	7	…	15	Column #
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	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	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

S(i, j) <16, can be represented with 4 bits

Example: B =101111

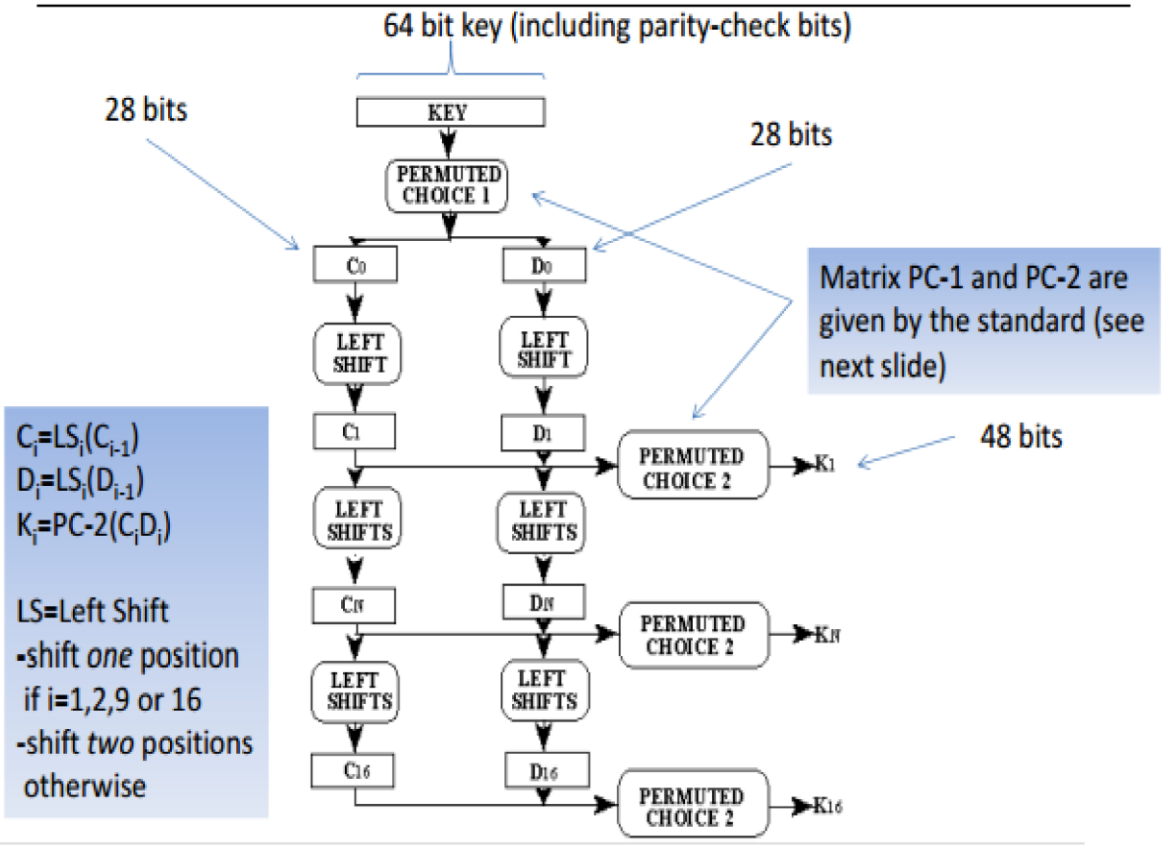
b_1b_6 = 11 = row 3

$b_2b_3b_4b_5$= 0111 = column 7

➔ C=7=0111

Another example: B=011011, C=?

DES Key Generation ($K_1 - K_{16}$)



DES Permuted Choice 1 and 2 (PC-1, PC-2)

Parity-check bits (namely, bits 8,16, 4,32,40,48,56,64) are not chosen, they do not appear in **PC-1**

Left						
57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
Right						
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

14	17	11	24	1	5	3	28
15	6	21	10	23	19	12	4
26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56
34	53	46	42	50	36	29	32

PC-2 selects the 48-bit subkey for each round from the 56-bit key-schedule state

DES Weak Keys

- DES uses 16 48-bits keys generated from a master 56-bit key (64 bits if we consider also parity bits)
- Weak keys: keys make the same sub-key to be generated in more than one round.
- Result: reduce cipher complexity
- Weak keys can be avoided at key generation.
- DES has 4 weak keys
  - 01010101 01010101
  - FEFEFEFE FEFEFEFE
  - E0E0E0E0 F1F1F1F1
  - 1F1F1F1F 0E0E0E0E

DES Decryption

Decryption uses the same algorithm as encryption, except that the subkeys K 1, K 2, …K16 are applied in reversed order

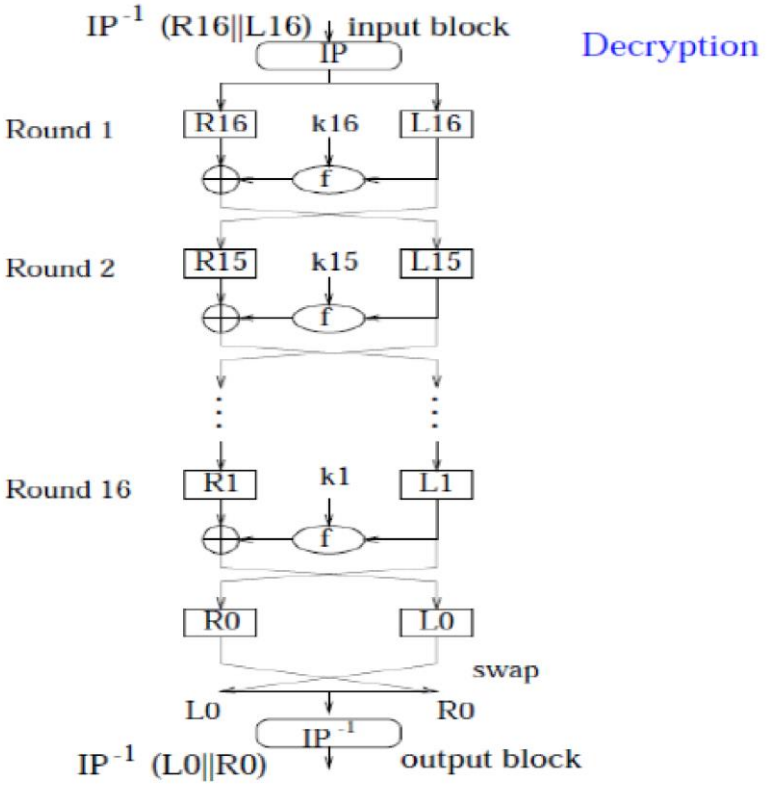
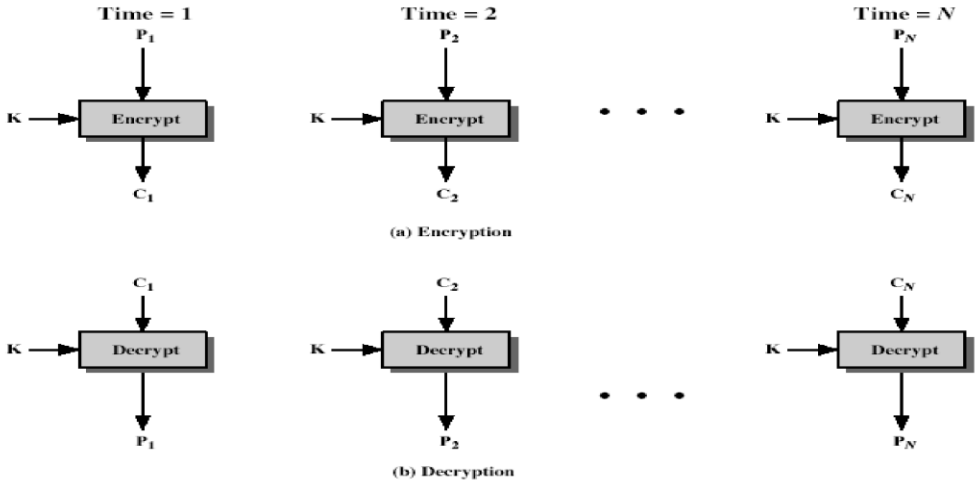


Figure . DES Decryption

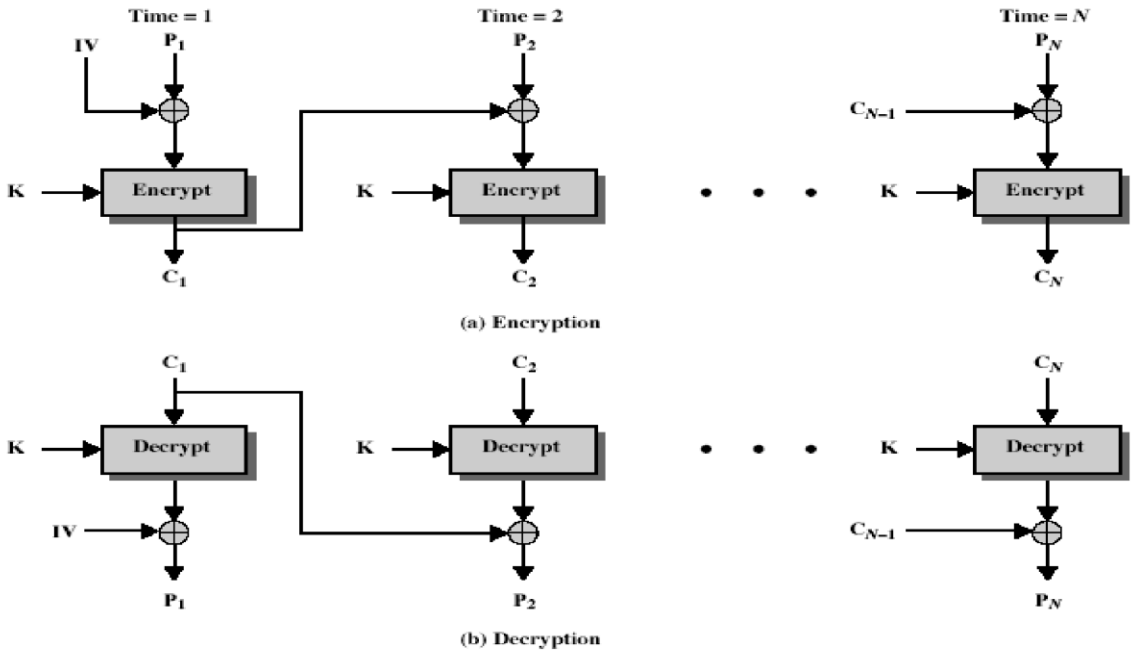
DES Modes

1. ECB (Electronic Code Book)Operation Mode - Blocks of clear text are encrypted independently. Main properties of this mode:
   - Identical clear text blocks are encrypted to identical cipher text blocks.
   - Re-ordering clear text blocks results in re-ordering cipher text blocks.
   - An encryption error affects only the block where it occurs.

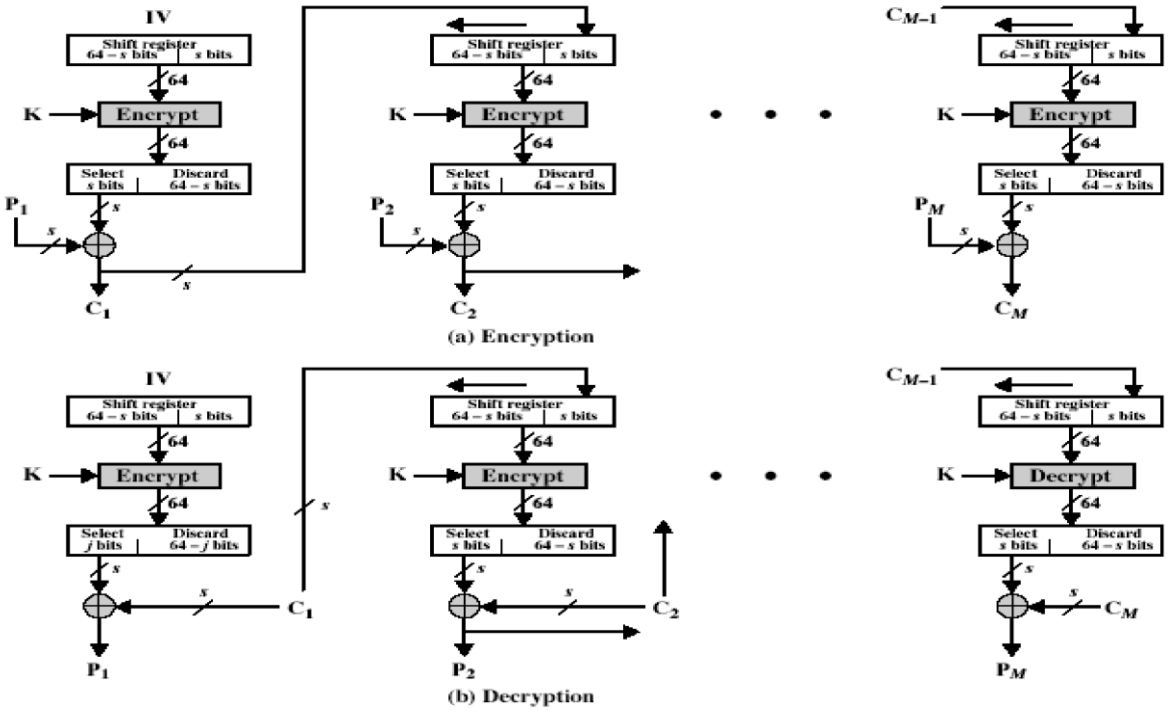


2. **CBC (Cipher Block Chaining) Operation Mode** - The previous cipher text block is XORed with the clear text block before applying the encryption mapping. Main properties of this mode:

An encryption error affects only the block where is occurs and one next block.



3. **CFB (Cipher FeedBack)** - Message is treated as a stream of bits , Bitwise-added to the output of the block cipher , Result is feedback for next stage (hence name) Cipher FeedBack (CFB) Message is treated as a stream of bits , Bitwise-added to the output of the block cipher , Result is feedback for next stage (hence name).



4. **OFM (Output Feedback Mode)**- The block cipher is used as a stream cipher, it produces the random key stream.