Assignment No ICS-1

TITLE	Implementation of S-DES
PROBLEM STATEMENT/ DEFINITION	Implementation of S-DES
OBJECTIVE	To understand how S-DES works Implementation of S-DES
OUTCOME	Students will learn and implement S- DES
S/W PACKAGES AND HARDWARE APPARATUS USED	Core 2 DUO/i3/i5/i7 64-bit processor OS-LINUX 64 bit OS Editor-gedit/Eclipse S/w- Jupyter Notebook/ Weka/ Python

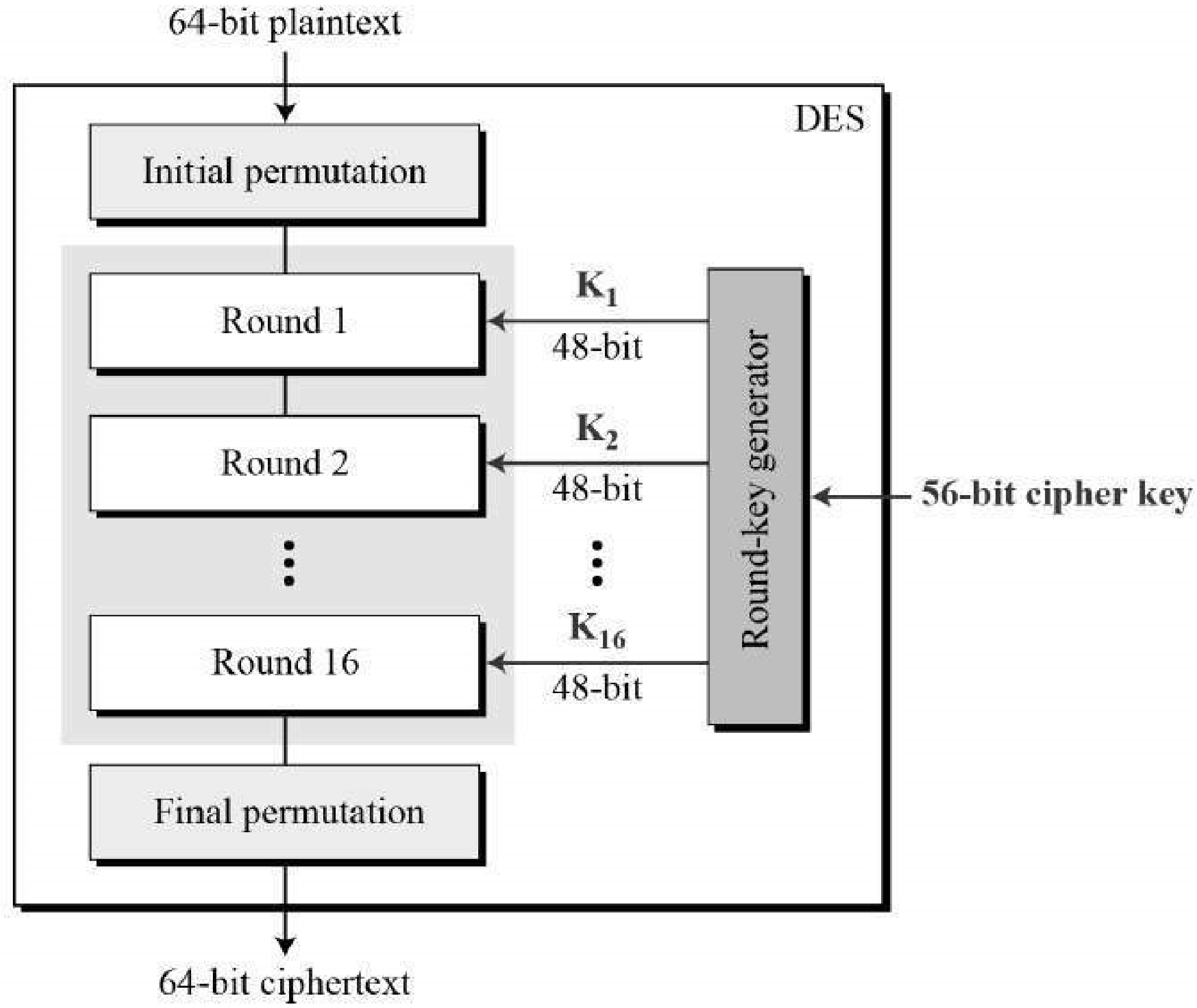
CONCEPTS RELATED THEORY:

Simplified-DES is an algorithm that inhibits many features of DES, but much simpler than DES. Like DES, this algorithm is also a block cipher.

DES :

The Data Encryption Standard (DES) is a symmetric-key block cipher published by the National Institute of Standards and Technology (NIST).DES is an implementation of a Feistel Cipher. It uses a 16 round Feistel structure. The block size is 64-bit.
Though, key length is 64-bit, DES has an effective key length of

56 bits, since 8 of the 64 bits of the key are not used by the encryption algorithm (function as check bits only). General Structure of DES is depicted in the following illustration -

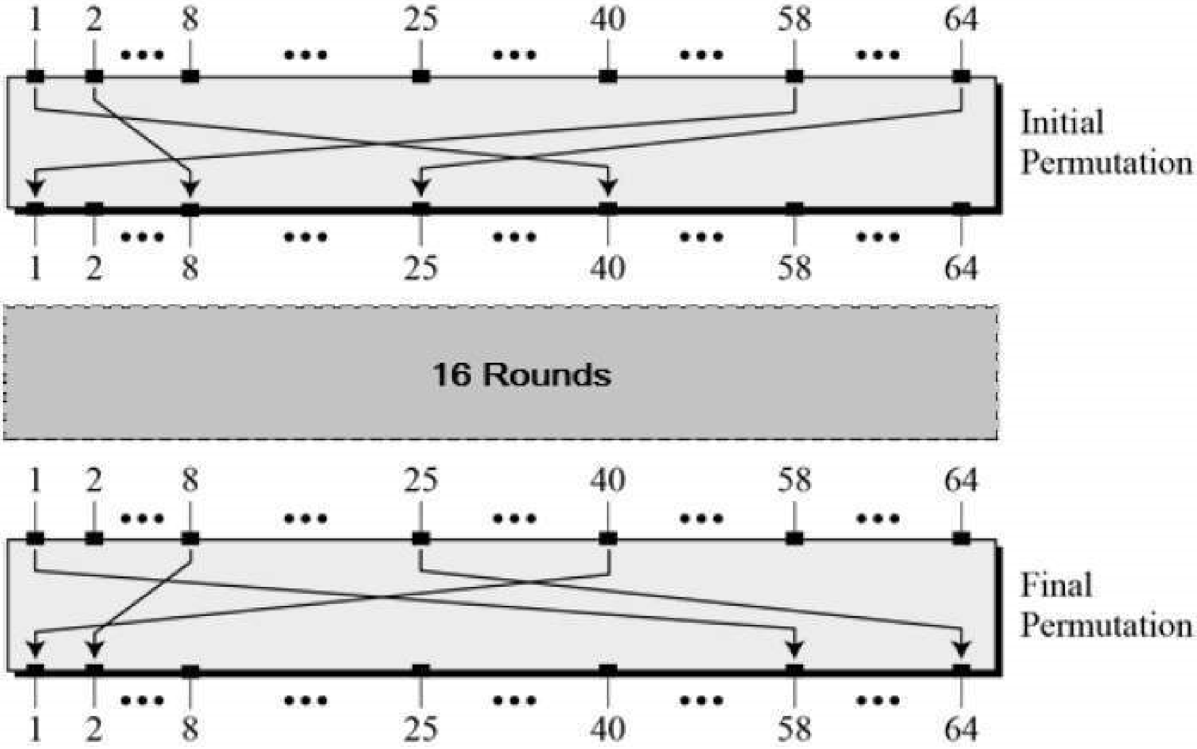


Since DES is based on the Feistel Cipher, all that is required to specify DES is −

- Round function
- Key schedule
- Any additional processing − Initial and final permutation

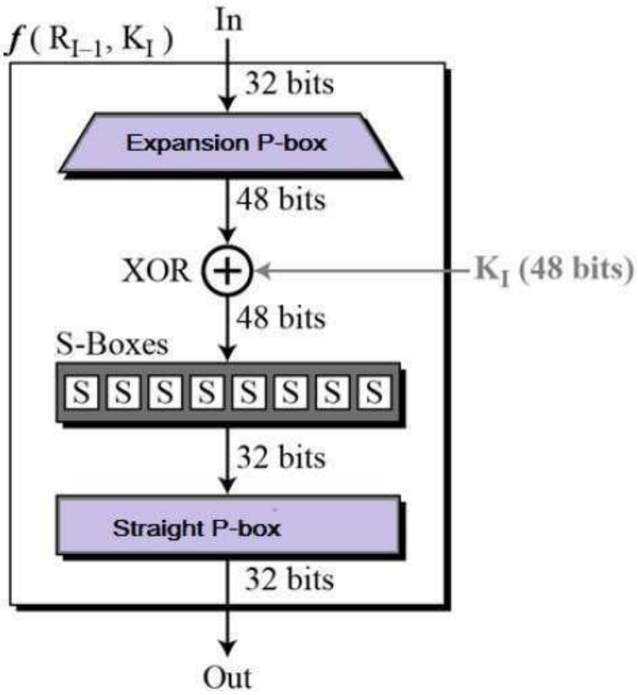
Initial and Final Permutation

The initial and final permutations are straight Permutation boxes (P-boxes) that are inverses of each other. They have no cryptography significance in DES. The initial and final permutations are shown as follows −

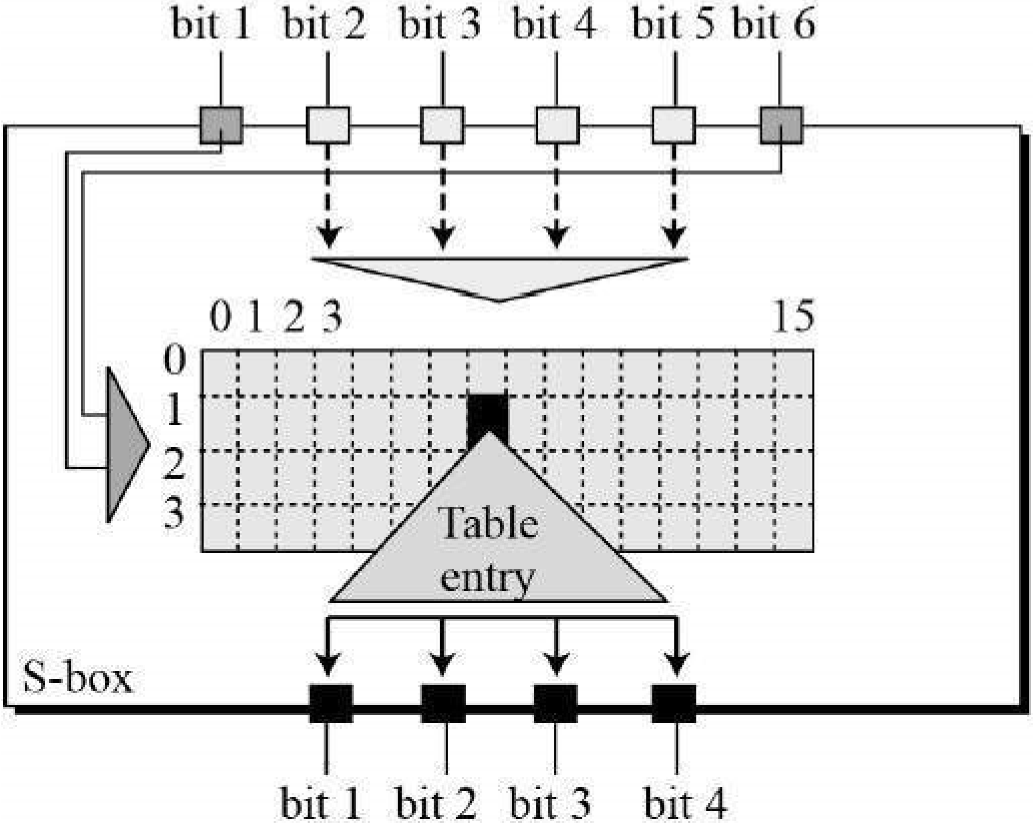


Round Function

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

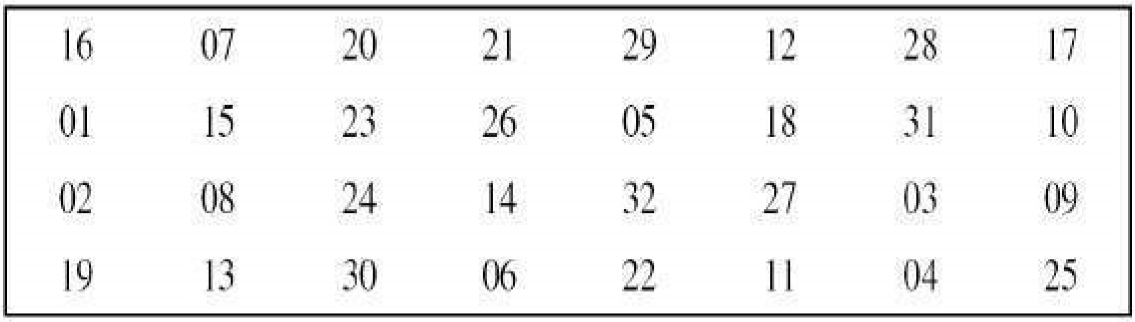


The S-box rule is illustrated below −



There are a total of eight S-box tables. The output of all eight s- boxes is then combined into a 32 bit section.

Straight Permutation − The 32 bit output of S-boxes is then subjected to the straight permutation with rule shown in the following illustration:

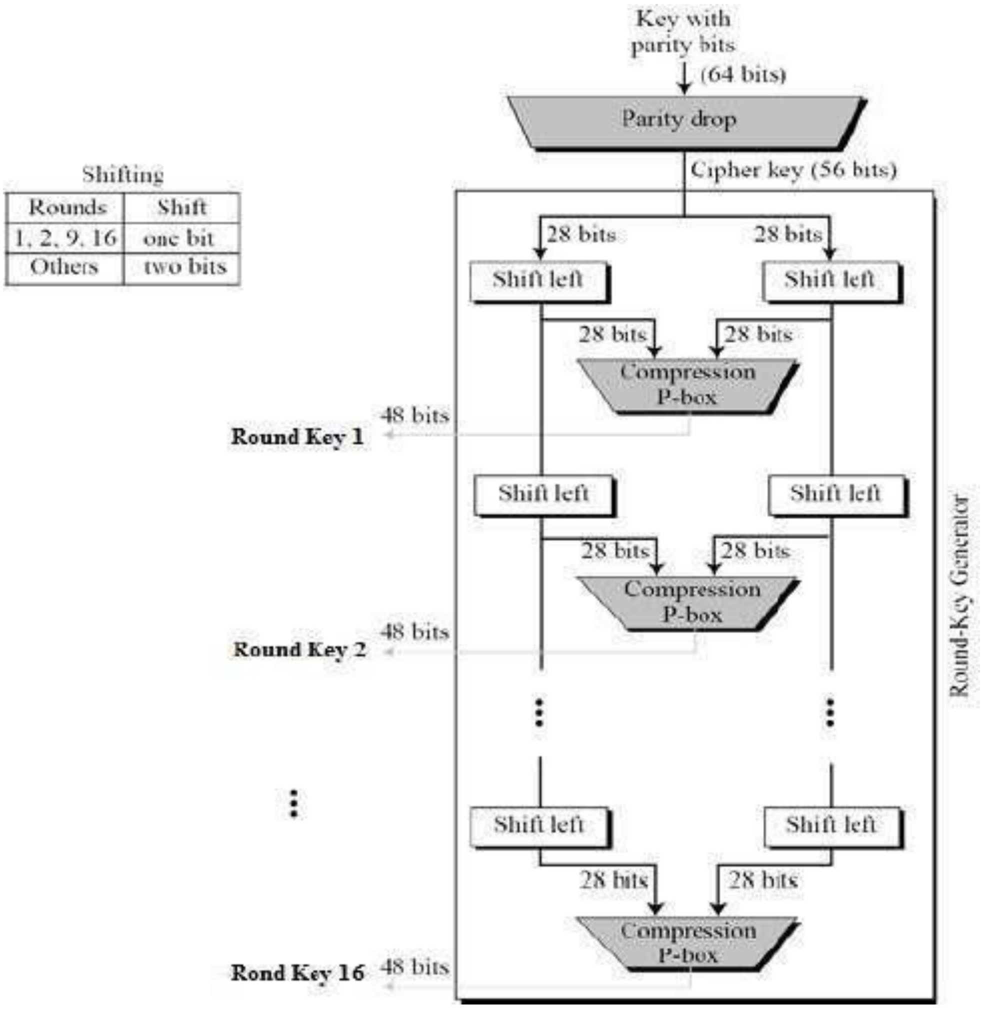


16	07	20	21	29	12	28	17
01	15	23	26	05	18	31	10
02	08	24	14	32	27	03	09
19	13	30	06	22	11	04	25

Key Generation

The round-key generator creates sixteen 48-bit keys out of a 56-bit cipher key.

The process of key generation is depicted in the following illustration −



DES Analysis

The DES satisfies both the desired properties of block cipher. These two properties make cipher very strong.

- Avalanche effect − A small change in plaintext results in the very great change in the ciphertext.
- Completeness − Each bit of ciphertext depends on many bits of plaintext.

During the last few years, cryptanalysis have found some weaknesses in DES when key selected are weak keys. These keys shall be avoided. DES has proved to be a very well designed block cipher. There have been no significant cryptanalytic attacks on DES other than exhaustive key search.

Algorithm :

Block Size :

In S-DES encryption and decryption is done on blocks of 12 bits. The plaintext / ciphertext is divided into blocks of 12 bits and the algorithm is applied on each block.

Key :

The key has 9 bits. The key k_i for the i^{th} round of encryption is obtained by using 8 bits of K, sorting with j bit.

The encryption algorithm involves five functions:

1) an initial permutation (IP).
2) a complex function labeled f_k, which involves both permutation and substitution operations and depends on a key input.
3) a simple permutation function that switches (SW) the two halves of the data.
4) the function f_k again.
5) a permutation function that is the inverse of the initial permutation.

The function f_k takes as input not only the data passing through the encryption algorithm, but also an 8-bit key. Here a 10-bit key is used from which two 8-bit subkeys are generated. The key is first subjected to a permutation (P10). Then a shift operation is performed. The output of the shift operation then passes through a permutation function that produces an 8-bit output (P8) for the first subkey (K1). The output of the shift operation also feeds into another shift and another instance of P8 to produce the second subkey (K2).

The encryption algorithm can be expressed as a composition of functions: $IP^{-1}\ f_{K2}\ SW\ f_{k1}\ IP$

Which can also be written as

Ciphertext = IP^{-1} (f_{K2} (SW (f_{k1} (IP (plaintext))))) Where

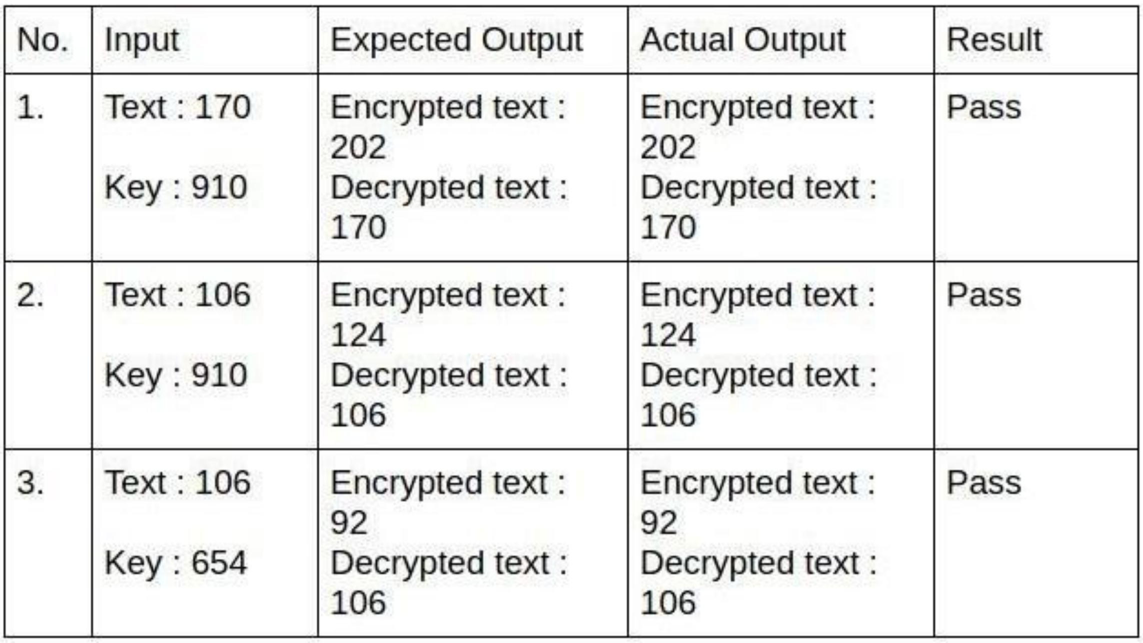
K1 = P8 (Shift (P10 (Key)))

K2 = P8 (Shift (shift (P10 (Key))))

Decryption can be shown as

Plaintext = IP^{-1} (f_{K1} (SW (f_{k2} (IP (ciphertext)))))

TEST CASES:



No.	Input	Expected Output	Actual Output	Result
1.	Text : 170 Key : 910	Encrypted text : 202 Decrypted text : 170	Encrypted text : 202 Decrypted text : 170	Pass
2.	Text : 106 Key : 910	Encrypted text : 124 Decrypted text : 106	Encrypted text : 124 Decrypted text : 106	Pass
3.	Text : 106 Key : 654	Encrypted text : 92 Decrypted text : 106	Encrypted text : 92 Decrypted text : 106	Pass

CONCLUSION:

Hence we have algorithm have successfully studied and implemented the concepts of the S-DES algorithm.