DES Algorithm

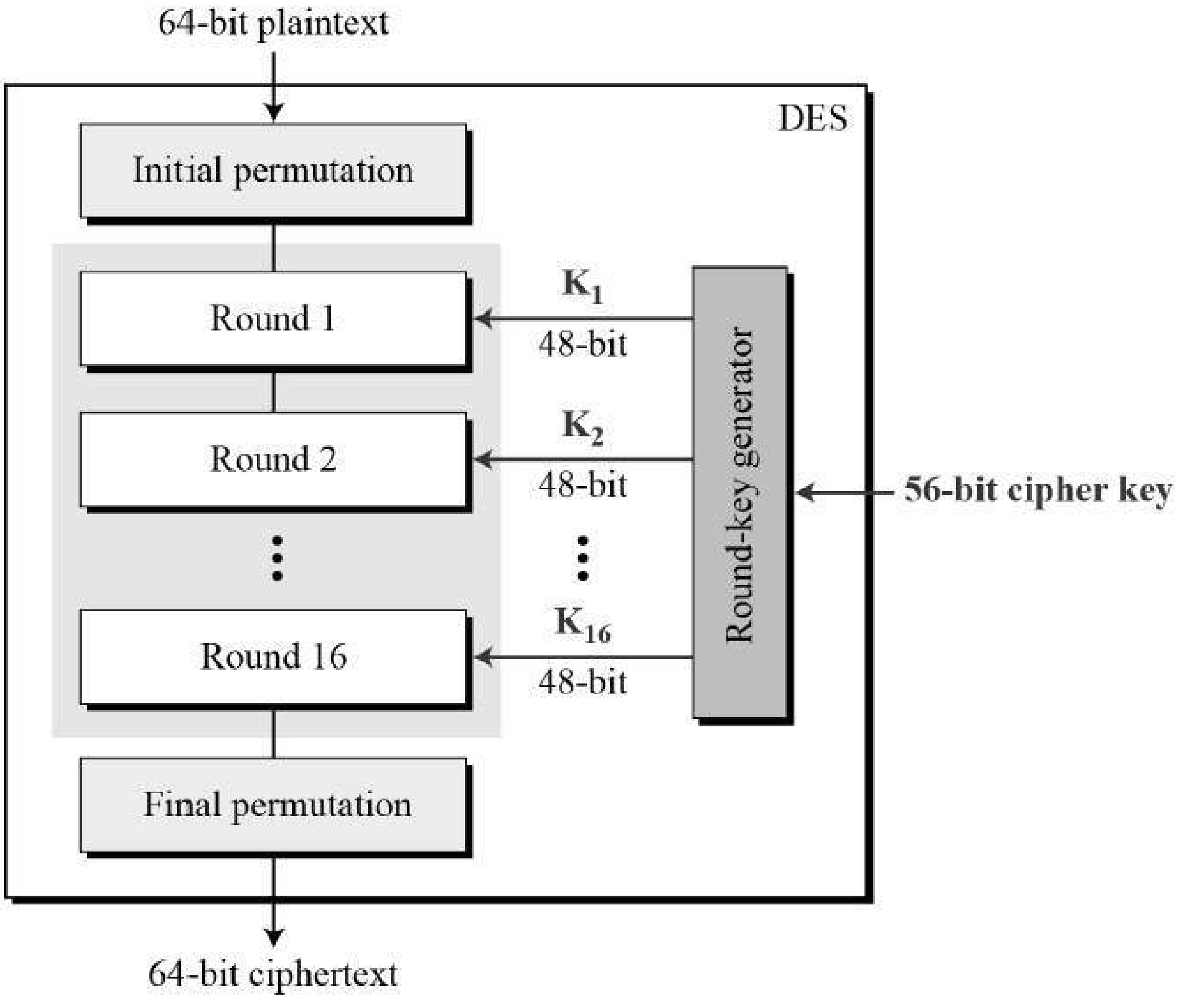
-Zishan Kazi, Keshav Bansal

Introduction

Data Encryption Standard is a symmetric-key algorithm for the encrypting the data. It comes under block cipher algorithm.

In the code, we take plain-text (string) and convert into bitstream on the basis of ascii value of the characters.
Every time we take 64 bits from that and give as input to DES algorithm, which is then processed through 16 rounds and then converted to cipher text.

Implementation

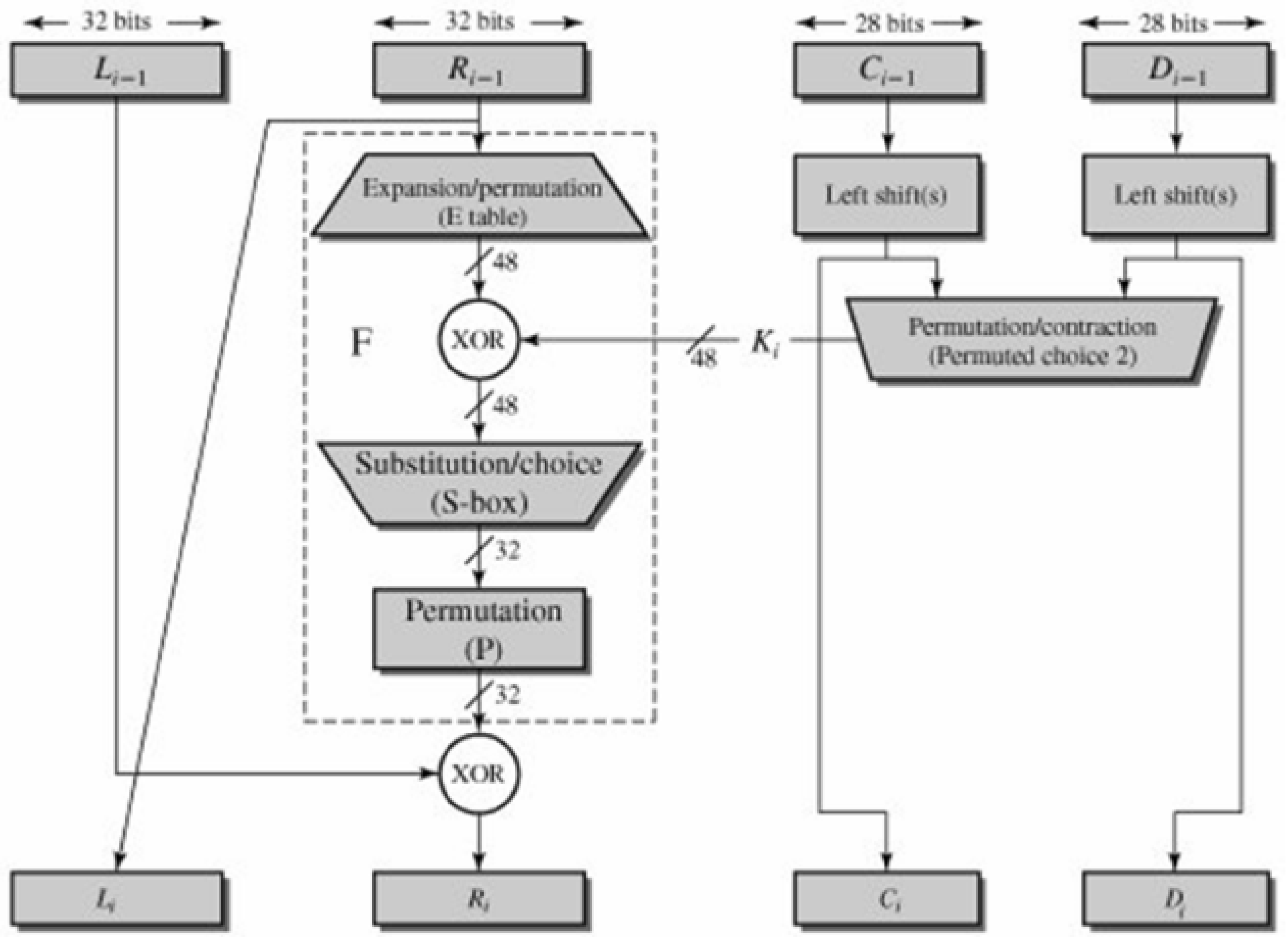


Initial permutation

When the string is converted into 64-bits then it is sent to initial permutation where we change the position of the bits according to the matrix **initial_permutation[64]** which we have defined in the code.

Rounds

After initial permutation, it is sent to round 1 which has its own set of 48-bit key. After which it is simlarly sent to 15 more similar rounds (16 in total) which have their own set of keys. Each round can be shown with the following diagram:



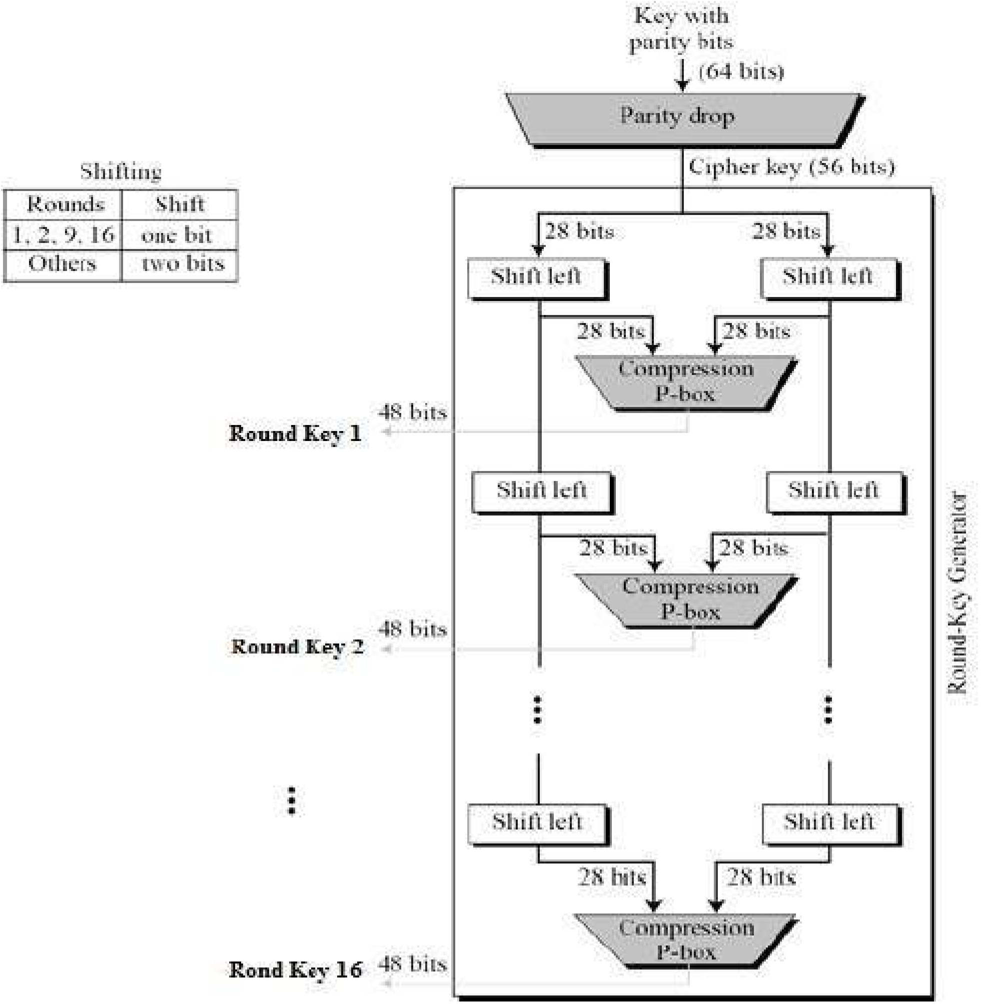
The L[i-1] and R[i-1] are the left half and right half of 64-bit stream sent after initial permutation.

Expansion of R[i-1] from 32 bits to 48 bits takes place so as to XOR with 48-bit key. For every box, the first bit of the next block becomes the last bit of the current block and the last bit of the previous block becomes the first bit of the current block.Thus, every block now contains 6 bits each. Then these 48 bits are XORed with the 48 bit key. It is then sent to 8 S-boxes each with 6 bit each. In an S-box for every 6 bits, 4 bits are returned as an S-box contains 4 rows and 16 columns. The first and the 6th bit determines the row number and the middle 4 bits determine the column number. Thus , a number which is between 0-15 is picked and it is then converted into its 4 bit form . Thus, the 48 bit string is converted into 32 bit string once again and it is permuted again. These 32 bits are then XORed with L[i-1]. These 32 bits form the R[i] and L[i] is nothing but the R[i-1] itself.

32-bit Swap

After completion of 16 rounds, final 64 bits is divided into 32 bit parts and they swap each other.

The right side in the diagram consists of key generation which can be explained in detail in the following diagram:



Key generation

We have 1x64 matrix Original_key[64], which is converted into 56-bit by removing the bits which lie on the positions that are divisible by 8. Thus bits on positions 8,16, 24, 32, 40, 48, 56, 64 are removed and the remaining ones are shuffled. The bits which are on the positions that are multiples of 8 are removed because they are parity bits.

Left Circular Shift

The 56 bit key is divided into 28 bit subkeys and these are then left shifted by a number of shifts which are fixed for a particular round. These are then merged to obtain the 56 bits key again and it is converted into a 48 bit string by sending it to the P-box and these 48 bits are again permuted.

This 48 bit key is XORed with the 48 bit string obtained after expansion permutation.