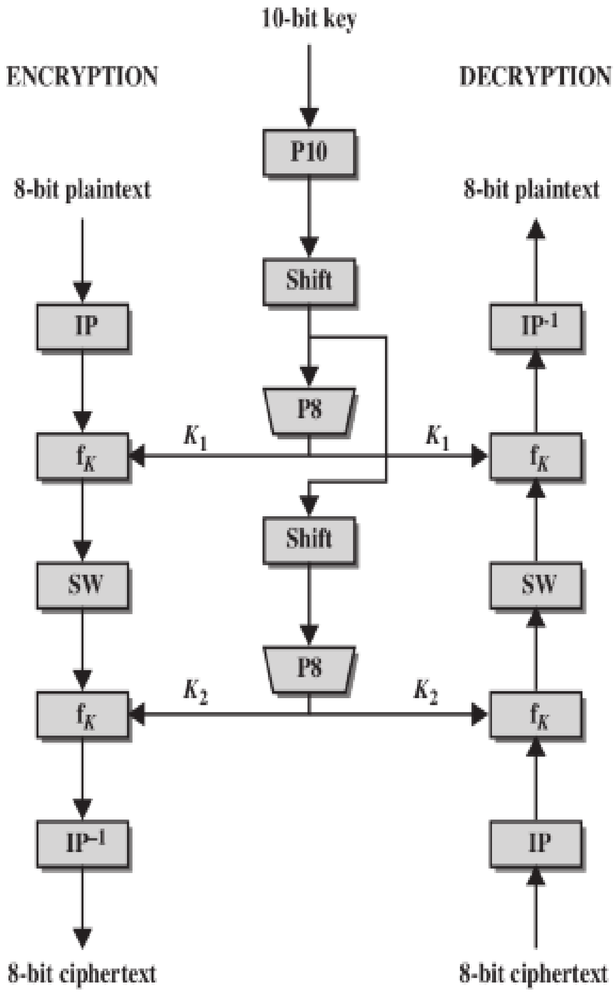
Block Ciphers and DES Examples

Simplified DES

- Input (plaintext) block: 8-bits
- Output (ciphertext) block: 8-bits
- Key: 10-bits
- Rounds: 2
- Round keys generated using permutations and left shifts
- Encryption: initial permutation, round function, switch halves
- Decryption: Same as encryption, except round keys used in opposite order

S-DES Algorithm



S-DES Operations

- P10 (permutate)

```
Input : 1 2 3 4 5 6 7 8 9 10
Output: 3 5 2 7 4 10 1 9 8 6
```

- P8 (select and permutate)

```
Input : 1 2 3 4 5 6 7 8 9 10
Output: 6 3 7 4 8 5 10 9
```

- P4 (permutate)

```
Input : 1 2 3 4
Output: 2 4 3 1
```

S-DES Operations

- EP (expand and permutate)

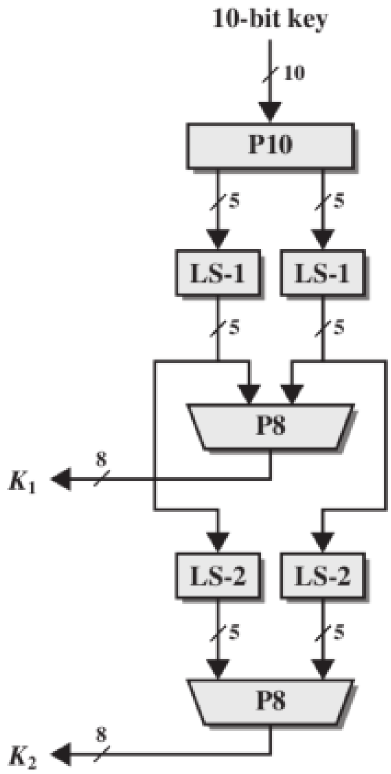
```
Input : 1 2 3 4
Output: 4 1 2 3 2 3 4 1
```

- IP (initial permutation)

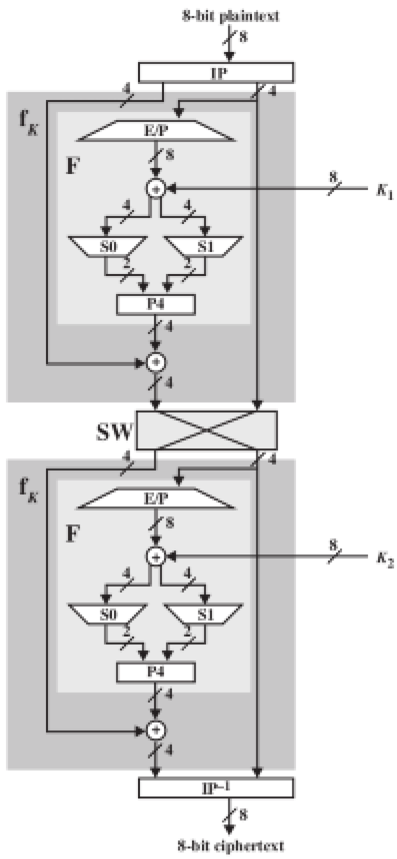
```
Input : 1 2 3 4 5 6 7 8
Output: 2 6 3 1 4 8 5 7
```

- IP^{-1} (inverse of IP)
- LS-1 (left shift 1 position)
- LS-2 (left shift 2 positions)

S-DES Key Generation



S-DES Encryption Details



S-DES S-Boxes

- S-DES (and DES) perform substitutions using S-Boxes
- S-Box considered as a matrix: input used to select row/column; selected element is output
- 4-bit input: $bit_1, bit_2, bit_3, bit_4$
- $bit_1 bit_4$ specifies row (0, 1, 2 or 3 in decimal)
- $bit_2 bit_3$ specifies column
- 2-bit output

$$S0 = \begin{bmatrix} 01 & 00 & 11 & 10 \\ 11 & 10 & 01 & 00 \\ 00 & 10 & 01 & 11 \\ 11 & 01 & 11 & 10 \end{bmatrix} \quad S1 = \begin{bmatrix} 00 & 01 & 10 & 11 \\ 10 & 00 & 01 & 11 \\ 11 & 00 & 01 & 00 \\ 10 & 01 & 00 & 11 \end{bmatrix}$$

Comparing DES and S-DES

S-DES

- 8-bit blocks
- 10-bit key: 2 x 8-bit round keys
- IP: 8-bits
- F operates on 4 bits
- 2 S-Boxes
- 2 rounds

DES

- 64-bit blocks
- 56-bit key: 16 x 48-bit round keys
- IP: 64 bits
- F operates on 32 bits
- 8 S-Boxes
- 16 rounds

S-DES encryption:

$$ciphertext = IP^{-1}\left(f_{K_2}\left(SW\left(f_{K_1}\left(IP\left(plaintext\right)\right)\right)\right)\right)$$

DES encryption:

$$ciphertext = IP^{-1}\left(f_{K_{16}}\left(SW\left(f_{K_{15}}\left(SW\left(\ldots\left(f_{K_1}\left(IP\left(plaintext\right)\right)\right)\right.\right.\right.\right.\right.$$

Simplified DES Example

Assume input 10-bit key, K, is: 1010000010

Then the steps for generating the two 8-bit round keys, K1 and K2, are:

1. Rearrange K using P10: 1000001100
2. Left shift by 1 position both the left and right halves: 00001 11000
3. Rearrange the halves with P8 to produce K1: 10100100
4. Left shift by 2 positions the left and right halves: 00100 00011
5. Rearrange the halves with P8 to produce K2: 01000011

K1 and K2 are used as inputs in the encryption and decryption stages.

Assume a 8-bit plaintext, P: 01110010

Then the steps for encryption are:

1. Apply the initial permutation, IP, on P: 10101001

2. Assume the input from step 1 is in two halves, L and R: L=1010, R=1001

3. Expand and permutate R using E/P: 11000011

4. XOR input from step 3 with K1: 10100100 XOR 11000011 = 01100111

5. Input left halve of step 4 into S-Box S0 and right halve into S-Box S1:

    a. For S0: 0110 as input: b1,b4 for row, b2,b3 for column
    b. Row 00, column 11 -> output is 10
    c. For S1: 0111 as input:
    d. Row 01, column 11 -> output is 11

6. Rearrange outputs from step 5 (1011) using P4: 0111

7. XOR output from step 6 with L from step 2: 0111 XOR 1010 = 1101

8. Now we have the output of step 7 as the left half and the original R as the right half. Switch the halves and move to round 2: 1001 1101

9. E/P with right half: E/P(1101) = 11101011

10. XOR output of step 9 with K2: 11101011 XOR 01000011 = 10101000

11. Input to s-boxes:

    a. For S0, 1010
    b. Row 10, column 01 -> output is 10
    c. For S1, 1000
    d. Row 10, column 00 -> output is 11

12. Rearrange output from step 11 (1011) using P4: 0111

13. XOR output of step 12 with left halve from step 8: 0111 XOR 1001 = 1110

14. Input output from step 13 and right halve from step 8 into inverse IP

  a. Input us 1110 1101
  b. Output is: 01110111

So our encrypted result of plaintext 01110010 with key 1010000010 is: 01110111

Other examples (encrypt or decrypt) could be:

- Plaintext: 11010101; Key: 0111010001; Ciphertext: 01110011
- Plaintext: 01001100; Key: 1111111111; Ciphertext: 00100010
- Plaintext: 00000000; Key: 0000000000; Ciphertext: 11110000
- Plaintext: 11111111; Key: 1111111111; Ciphertext: 00001111