Lecture 4:

Block Ciphers and DES

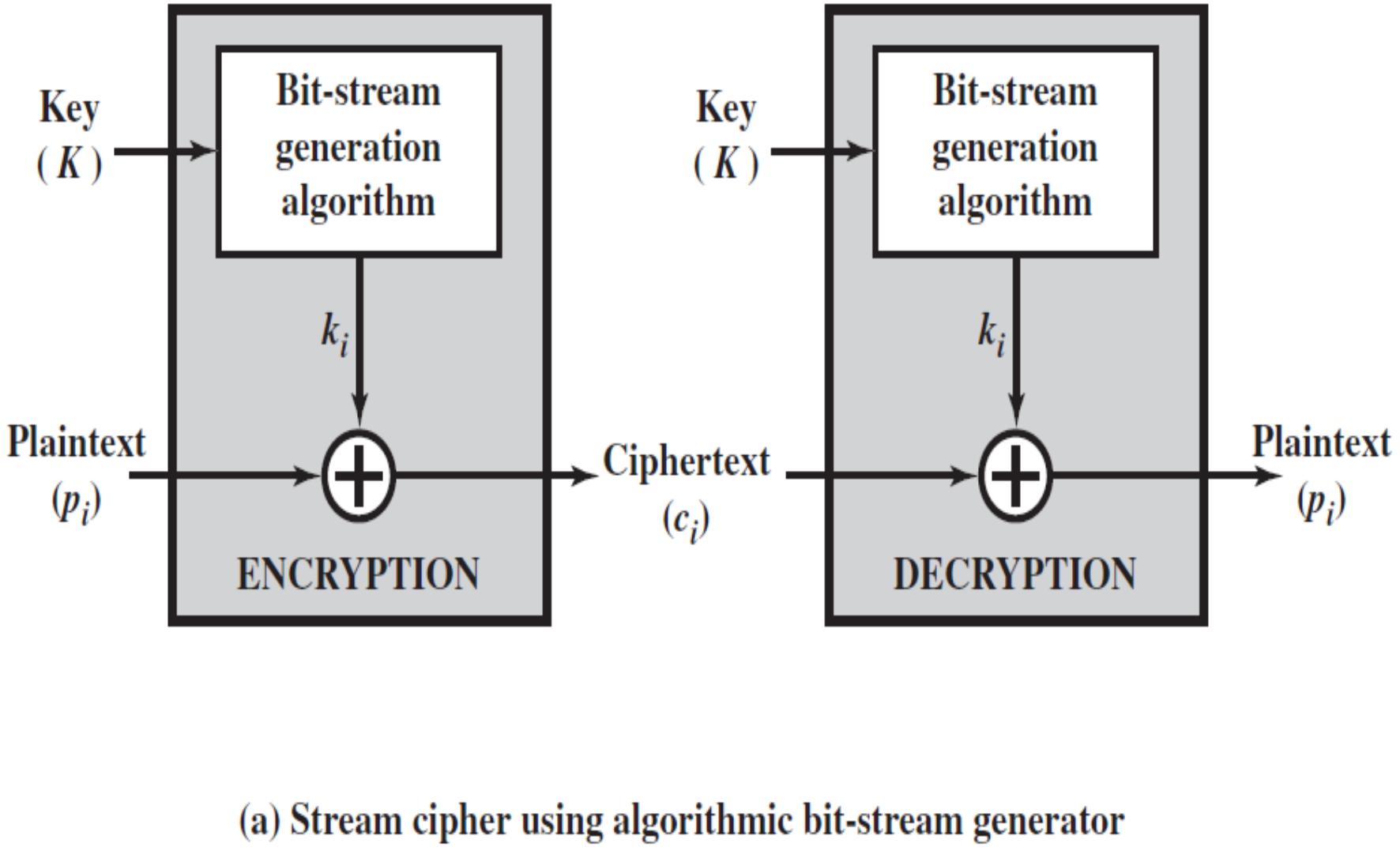
4th Year- Course, CCSIT, UoA

Lecture Goals

1. To introduce the notion of a block cipher in the modern context.
2. To introduce the notion of the Feistel Cipher Structure
3. To go over DES, the Data Encryption Standard

Stream Ciphers

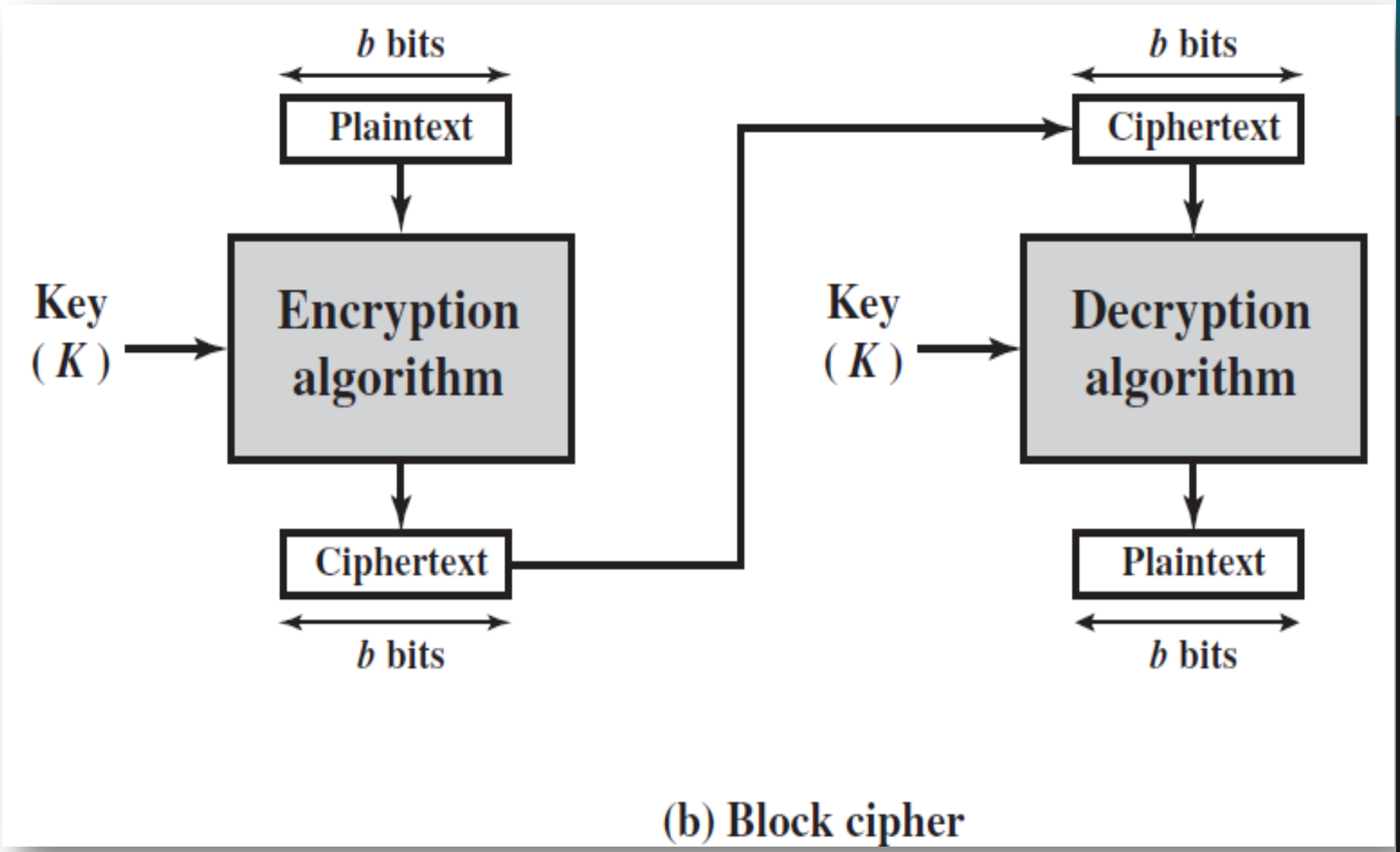
- A ***stream cipher*** is one that encrypts a digital data stream one bit or one byte at a time. Examples of classical stream ciphers are the autokeyed Vigenère cipher and the Vernam cipher.
- For practical reasons, the bit-stream generator must be implemented as an algorithmic procedure, so that the cryptographic bit stream can be produced by both users. In this approach (see next Figure), the bit-stream generator is a key-controlled algorithm and must produce a bit stream that is cryptographically strong.
- The two users need only share the generating key, and each can produce the keystream.



(a) Stream cipher using algorithmic bit-stream generator

Block Ciphers

- A ***block cipher*** is one in which a block of plaintext is treated as a whole and used to produce a ciphertext block of equal length. Typically, a block size of 64 or 128 bits is used. As with a stream cipher, the two users share a symmetric encryption key (see next Figure).
- Using some of the modes of operation (explained later), a block cipher can be used to achieve the same effect as a stream cipher.
- Far more effort has gone into analyzing block ciphers. In general, they seem applicable to a broader range of applications than stream ciphers.



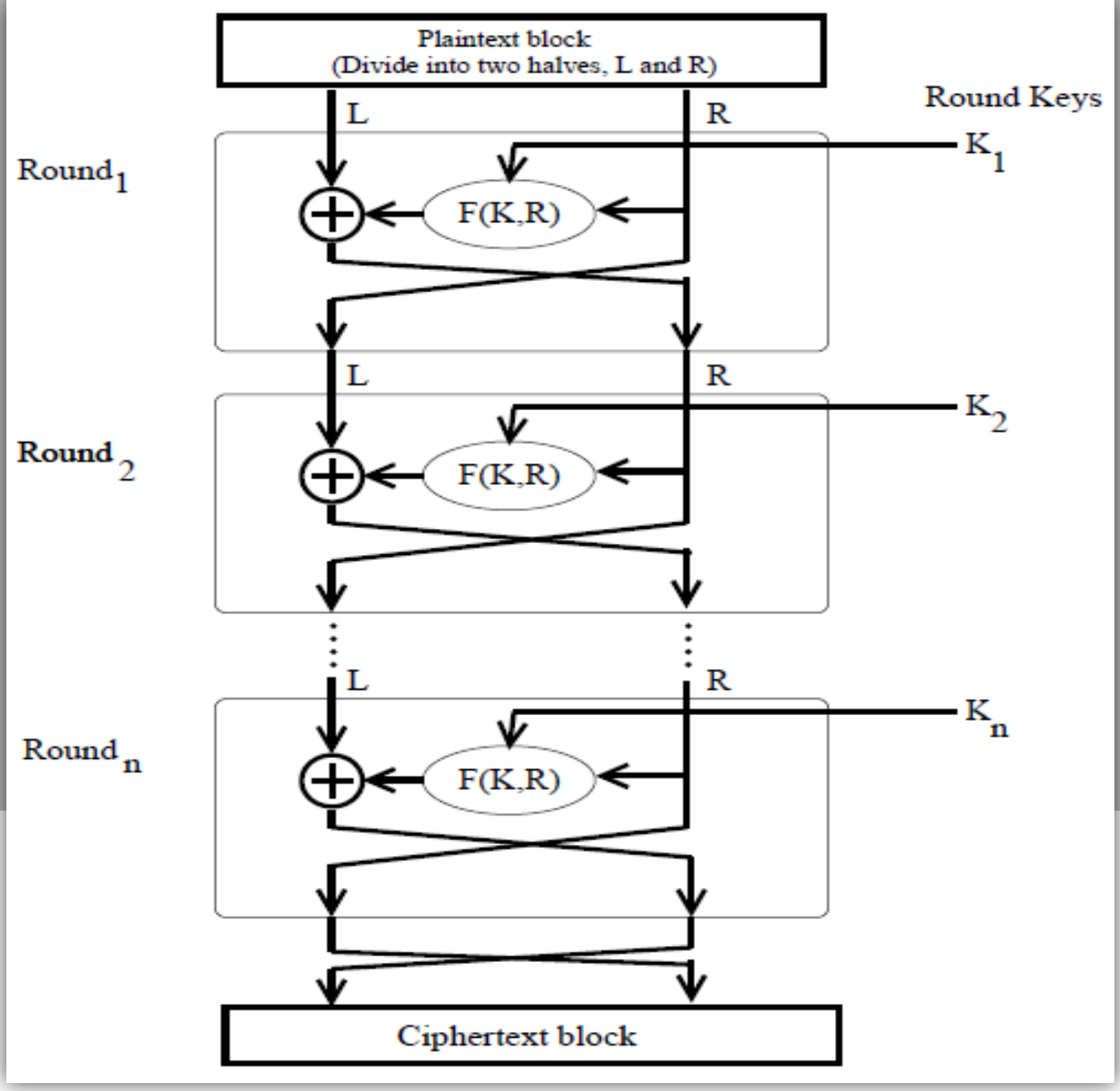
(b) Block cipher

The Feistel Structure for Block Ciphers (1)

- Named after the IBM cryptographer Horst Feistel and first implemented in the Lucifer cipher by Horst Feistel and Don Coppersmith.
- A cryptographic system based on Feistel structure uses the ***same basic algorithm*** for both encryption and decryption.
- As shown in the next Figure, the Feistel structure consists of ***multiple rounds*** of processing of the plaintext, with each round consisting of a ***substitution*** step followed by a ***permutation*** step.

The Feistel Structure for Block Ciphers (2)

- The input block to each round is divided into ***two halves*** that we can denote L and R for the left half and the right half.
- In each round, the right half of the block, R, goes through ***unchanged***. But the left half, L, goes through an ***operation*** that depends on R and the encryption key.
- The permutation step at the end of each round consists of ***swapping*** the modified L and R. Therefore, the L for the next round would be R of the current round. And R for the next round be the output L of the current round.


Plaintext block
(Divide into two halves, L and R)
Round Keys
L
R
Round 1
K 1
F(K,R)
L
R
Round 2
K 2
F(K,R)
L
R
Round n
K n
F(K,R)
Ciphertext block

Mathematical Description of Each Round in the Feistel Structure (1)

- Let LE_i and RE_i denote the output half-blocks at the end of the i^{th} round of processing. The letter 'E' denotes encryption.
- We obviously have

$$LE_i = RE_{i-1}$$
$$RE_i = LE_{i-1} \oplus F(RE_{i-1}, K_i)$$

- where $\oplus$ denotes the bitwise EXCLUSIVE OR operation. The symbol F denotes the operation that "scrambles" RE_{i-1} of the previous round with the current round key K_i.

Mathematical Description of Each Round in the Feistel Structure (2)

- Note that the ***round key*** K_i is derived from the ***main encryption key*** as we will explain later.
- F is referred to as the ***Feistel function***, after Horst Feistel, naturally.
- Assuming 16 rounds of processing (which is typical), the output of the last round of processing is given by

$$LE_{16} = RE_{15}$$
$$RE_{16} = LE_{15} \oplus F(RE_{15}, K_{16})$$

Decryption in Ciphers Based on the Feistel Structure (1)

- As shown in the next Figure, the decryption algorithm is exactly the ***same*** as the encryption algorithm with the ***only difference*** that the round keys are used in the ***reverse order***.
- The output of each round during decryption is the input to the corresponding round during encryption. This property holds true ***regardless of the choice of the Feistel function F*** .
- To prove the above claim, let LD_i and RD_i denote the left half and the right half of the output of the i^{th} round.

Decryption in Ciphers Based on the Feistel Structure (2)

- That means that the output of the first decryption round consists of LD_1 and RD_1. So we can denote the input to the first decryption round by LD_0 and RD_0.
- The relationship between the two halves that are input to the first decryption round and what is output by the encryption algorithm is

$$LD_0 = RE_{16}$$
$$RD_0 = LE_{16}$$

Decryption in Ciphers Based on the Feistel Structure (3)

- We can write the following equations for the output of the first decryption round

$$
\begin{aligned}
LD_1 &= RD_0 \\
&= LE_{16} \\
&= RE_{15} \\
RD_1 &= LD_0 \oplus F(RD_0, K_{16}) \\
&= RE_{16} \oplus F(LE_{16}, K_{16}) \\
&= [LE_{15} \oplus F(RE_{15}, K_{16})] \oplus F(RE_{15}, K_{16}) \\
&= LE_{15}
\end{aligned}
$$

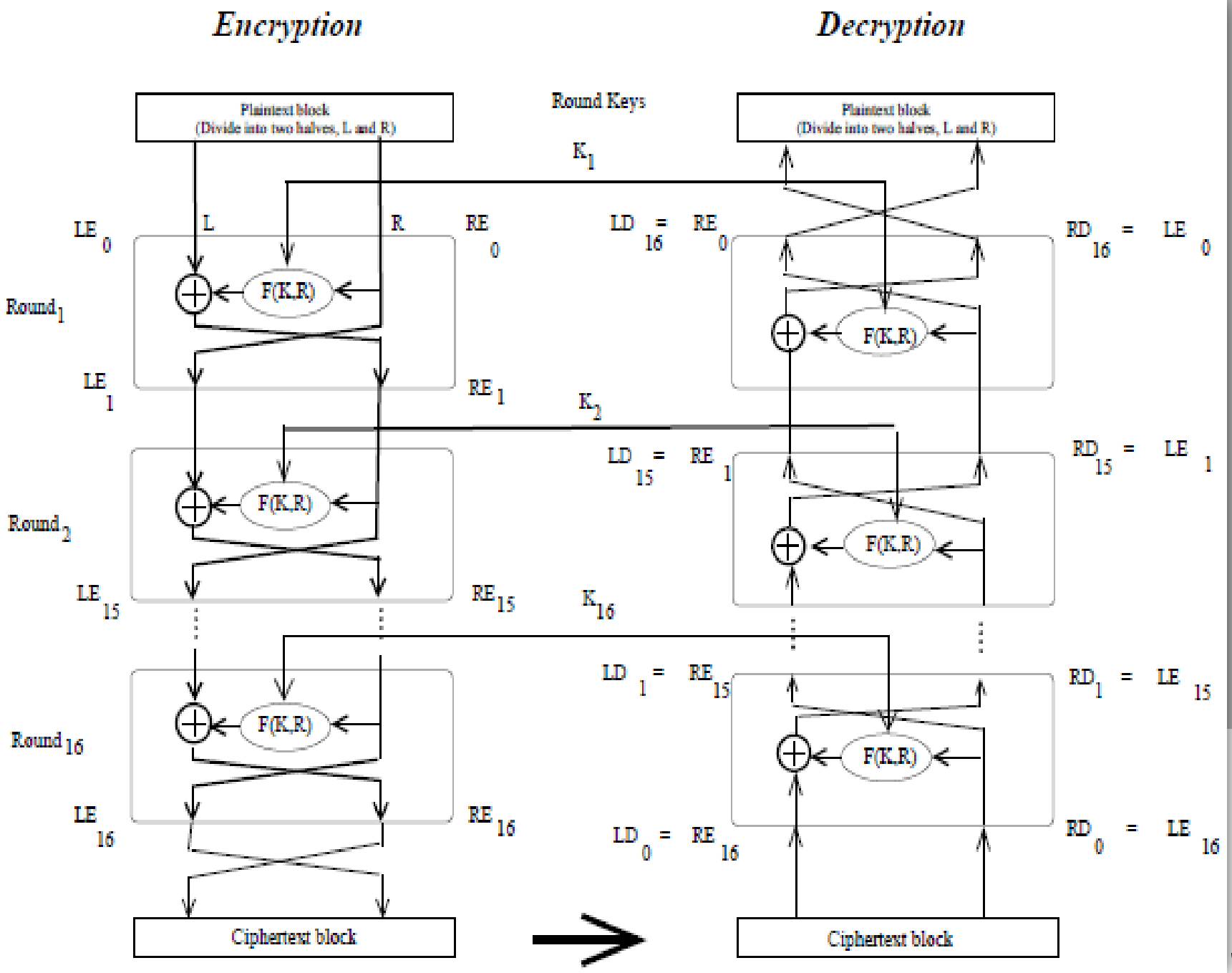
- This shows that the output of the first round of decryption is the same as the input to the last stage of the encryption round since we have $LD_1 = RE_{15}$ and $RD_1 = LE_{15}$

Decryption in Ciphers Based on the Feistel Structure (4)

- The following equalities are used in the above derivation. Assume that A, B, and C are bit arrays.

$$[A \oplus B] \oplus C = A \oplus [B \oplus C]$$
$$A \oplus A = 0$$
$$A \oplus 0 = A$$

- ***The above result is independent of the precise nature of F .***


Encryption
Decryption
Plaintext block
(Divide into two halves, L and R)
Round Keys
K1
LE0
L
R
RE0
LD16 = RE0
RD16 = LE0
Round1
F(K,R)
LE1
RE1
K2
LD15 = RE1
RD15 = LE1
Round2
LE15
RE15
K16
LD1 = RE15
RD1 = LE15
Round16
LE16
RE16
LD0 = RE16
RD0 = LE16
Ciphertext block
Ciphertext block

Feistel Network Parameters

1. ***Block size:*** Larger block sizes mean greater security but reduced encryption/decryption speed for a given algorithm.
2. ***Key size:*** Larger key size means greater security but may decrease encryption/ decryption speed.
3. ***Number of rounds:*** The essence of the Feistel cipher is that a single round offers inadequate security but that multiple rounds offer increasing security.
4. ***Subkey generation algorithm:*** Greater complexity in this algorithm should lead to greater difficulty of cryptanalysis.
5. ***Round function F :*** Again, greater complexity generally means greater resistance to cryptanalysis.

The F Function

1. The heart of a Feistel block cipher is the function F, which provides the element of ***confusion*** (will be explained latter) in a Feistel cipher.
2. Thus, it must be difficult to "unscramble" the substitution performed by F.
3. One obvious criterion is that F be ***nonlinear***. The more nonlinear F, the more difficult any type of cryptanalysis will be.