Back to TOC

3.2 THE FEISTEL STRUCTURE FOR BLOCK CIPHERS

The DES (Data Encryption Standard) algorithm for encryption and decryption, which is the main theme of this lecture, is based on what is known as the **Feistel Structure**. This section and the next two subsections introduce this structure:

- 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 Figure 2, 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 input block to each round is divided into two halves that I have denoted **L** and **R** for the left half and the right half.

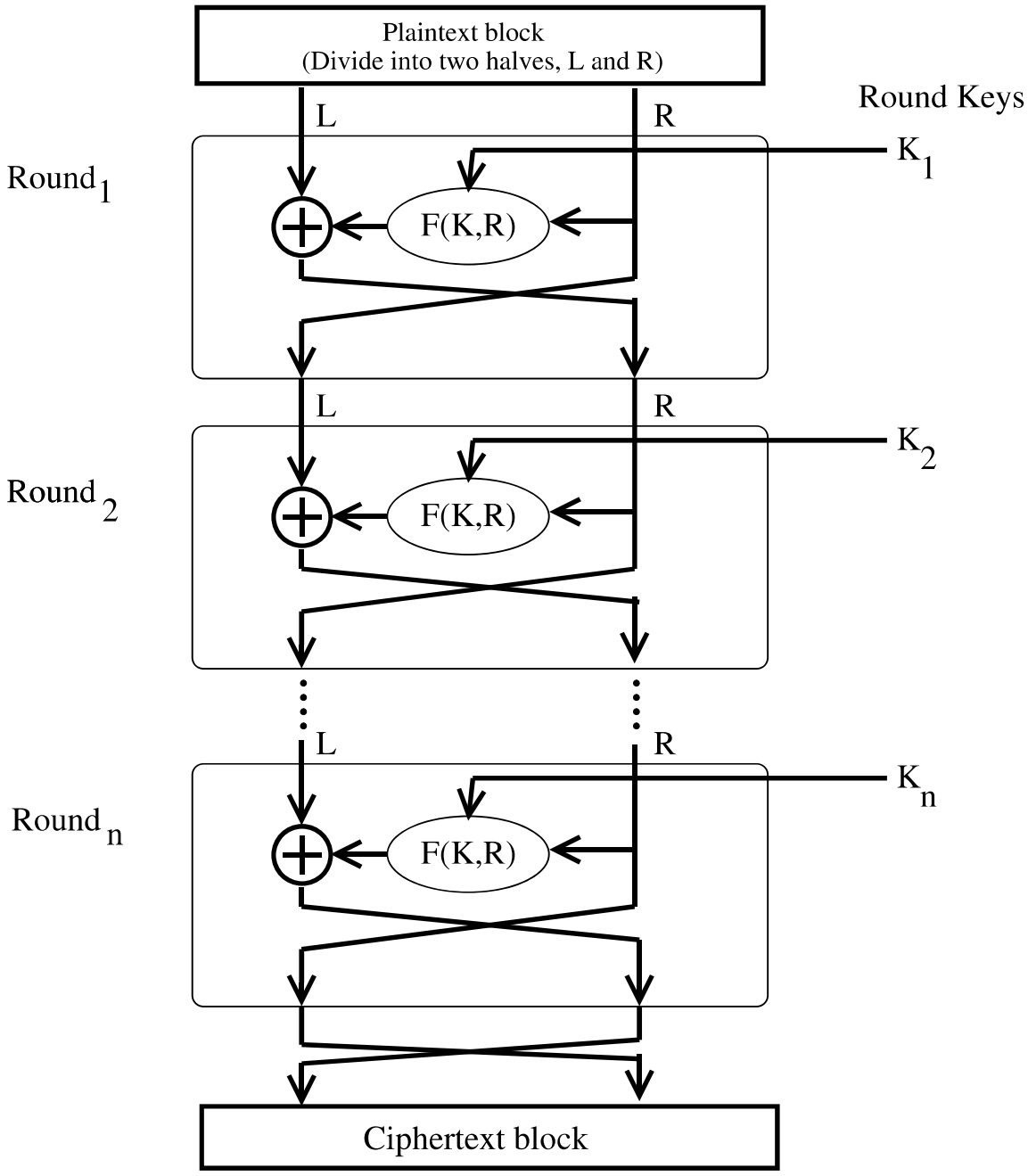

Figure 2: *The Feistel Structure for symmetric key cryptography* (*This figure is from Lecture 3 of "Lecture Notes on Computer and Network Security" by Avi Kak*)

- 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 operation carried out on the left half **L** is referred to as the **Feistel Function**.

- 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.*

- The next two subsection present important properties of the Feistel structure. As you will see, these properties are invariant to our choice for the Feistel Function.

- Besides DES, there exist several block ciphers today — the most popular of these being **Blowfish**, **CAST-128**, and **KASUMI** — that are also based on the Feistel structure.

Back to TOC

3.2.1 Mathematical Description of Each Round in the Feistel Structure

- 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.

- In the Feistel structure, the relationship between the output of the i^{th} round and the output of the previous round, that is, the $(i-1)^{th}$ round, is given by

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

  where $\oplus$ denotes the bitwise `EXCLUSIVE-OR` operation. The symbol F denotes the operation that "scrambles" RE_{i-1} of the previous round with what is shown as the **round key** K_i in Figure 2. **The round key K_i is derived from the main encryption key as will be explained 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

$$\begin{aligned} LE_{16} &= RE_{15} \\ RE_{16} &= LE_{15} \oplus F(RE_{15}, K_{16}) \end{aligned}$$

Back to TOC

3.2.2 Decryption in Ciphers Based on the Feistel Structure

- As shown in Figure 3, 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 — except for the left-right switch between the two halves. 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.

- 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:

$$LD_0 \quad = \quad RE_{16}$$

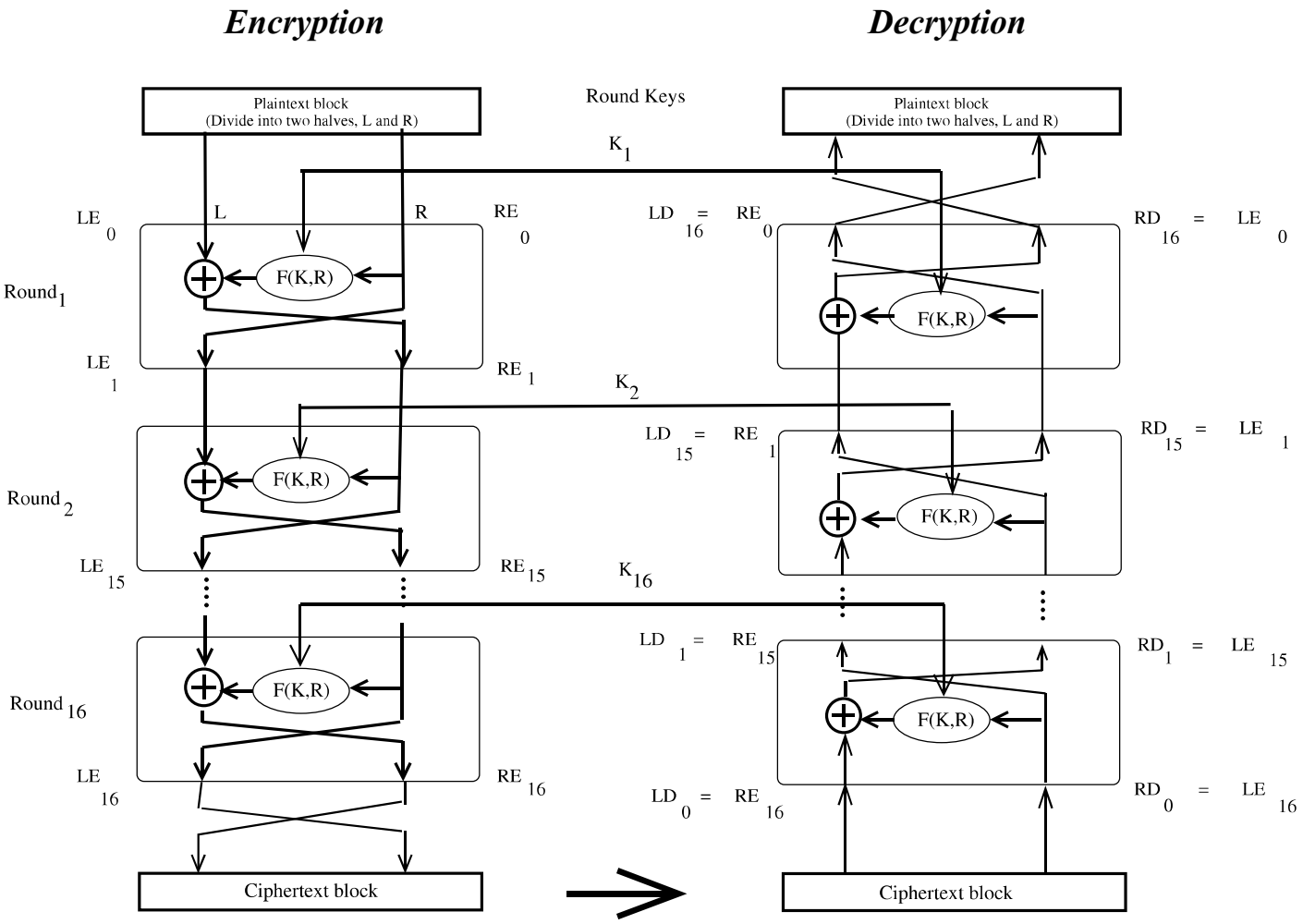

Figure 3: *When a Feistel structure is used, decryption works the same as encryption.* (*This figure is from Lecture 3 of "Lecture Notes on Computer and Network Security" by Avi Kak*)

$$RD_0 \quad = \quad LE_{16}$$

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

$$\begin{aligned} LD_1 &= RD_0 \\ &= LE_{16} \\ &= RE_{15} \end{aligned}$$

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

  This shows that, except for the left-right switch, 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 $RD_1 = LE_{15}$

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

$$\begin{aligned} [A \oplus B] \oplus C &= A \oplus [B \oplus C \ ] \\ A \oplus A &= 0 \\ A \oplus 0 &= A \end{aligned}$$

- **The above result is independent of the precise nature of the Feistel function** F. That is, the output of each round during decryption is the input to the corresponding round during encryption for every choice of the Feistel function F.