Multiple Encryption & DES

E. Suresh Babu M.Tech(CSE).,(PhD)

Associate Professor

Department of CSE

Goals:

- To present Double-DES and its vulnerability to the meet-in-the-middle attack
- To present two-key Triple-DES and Triple-DES
- To present the five different modes in which a block cipher can be used in practical systems for secure communications

Why Multiple Encryptions are Needed

Why Multiple Encryptions are required

- As you already know,
  - ✓ The DES cryptographic system was **not very secure** about 10 years ago.
- AES cryptography was designed which is **extremely secure**
  - ✓ But the world of commerce and finance does not want to **give up on DES** that quickly (because of all the investment that has already been in DES-related **software and hardware**)

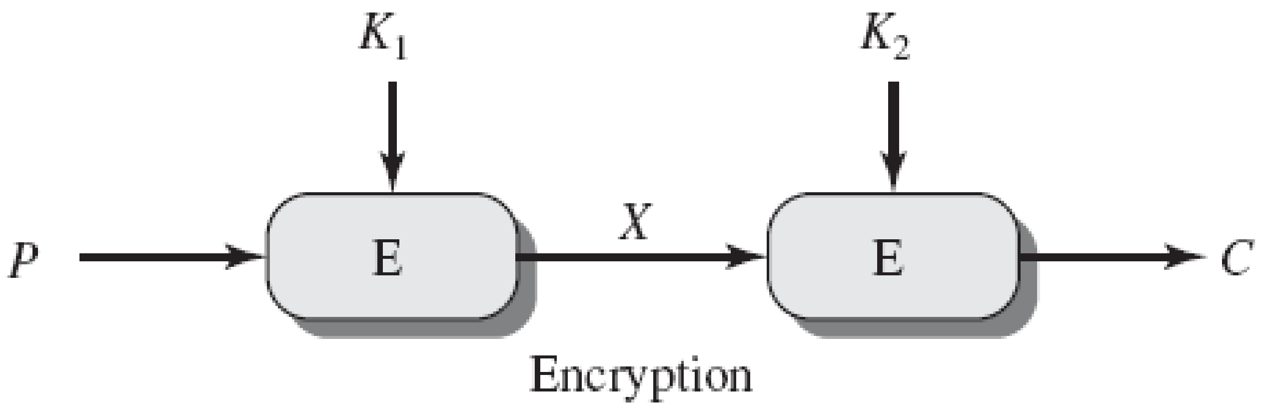
Next Questions Raises

- How about a **cryptographic system** that carries out **repeated encryptions with DES?**
  - ✓ Would that be **more secure?**

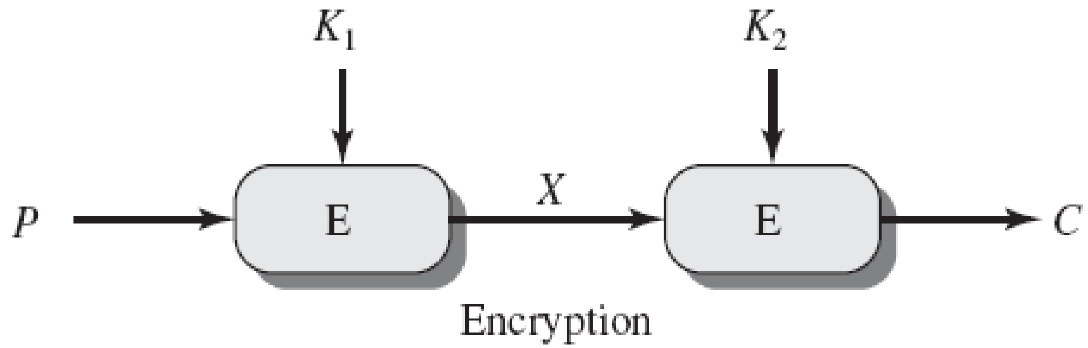
DOUBLE DES

DOUBLE DES

- The simplest form of **multiple encryptions** with DES
  - ✓ The double DES that has **two** DES-based encryption **stages** using two different keys.



Working Model of Double DES



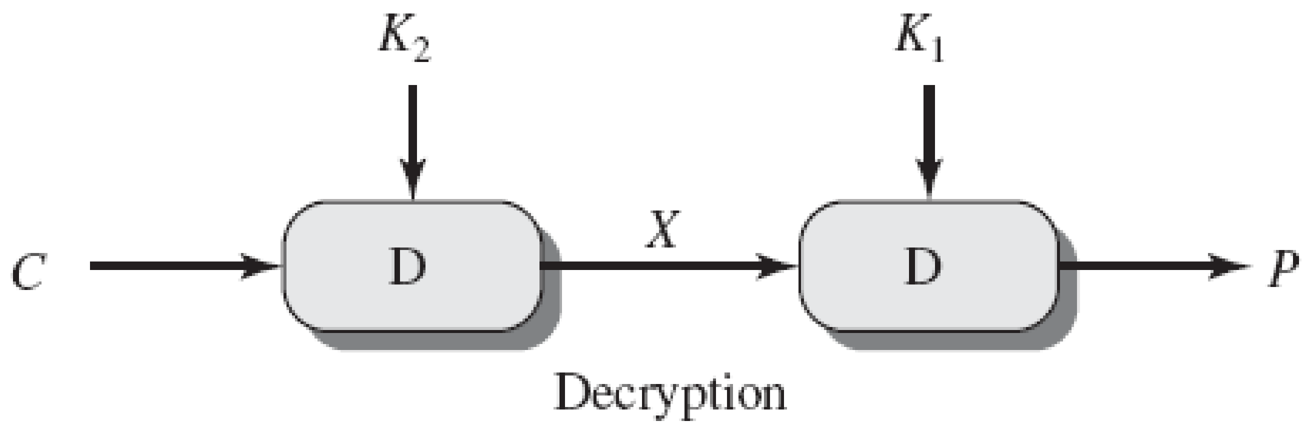
- Let's say that P represents a **64-byte block of plaintext**.
- Let E represent the **process of encryption** that transforms a plaintext block into a cipher text block.
- Let's use **two 56-byte** encryption keys **K_1 and K_2** for a double application of DES to the plaintext.

Working Model of Double DES

❖ Let C represent the **resulting block of cipher text**. We have

$$C = E(K_2, E(K_1, P))$$

❖ **Decryption** requires that the keys be applied in **reverse order:**



$$P = D(K_1, D(K_2, C))$$

Observation

- With **Two keys**, each of length **56 bits**,
  - ✓ Double DES in effect uses a 112 bit key.
  - ✓ Double DES would result in a **dramatic increase** in the cryptographic strength of the cipher (Against the brute-force attacks to which the regular DES is so vulnerable).

Can a Double-DES (2DES) is Equivalent to a Single-DES

Can a Double-DES (2DES) is Equivalent to a Single-DES

- Suppose If we said that 2DES with the two keys (K_1,K_2) is equivalent to a single application of DES with some key K_3.

$$\mathbf{E(K_2,\ E(K_1,\ P)) = E(K_3,\ P)}$$

Plaintext-to-Cipher text Mapping

- The plaintext-to-cipher text **mapping must be one-one**, the mapping created by a single application of DES encryption can be thought of as a specific **permutation of the 2^{64} different possible integer values** for a plaintext block.
- The **total number** of all possible plaintext-to-cipher text mappings is the **very large number 2^{64}!** for the 64-bit block encryption

Plaintext-to-Cipher text Mapping

- Each mapping can be thought of as a **permutation of the 2^{64}** possible words at the input, we have a **maximum of 2^{64}!** Possible mappings between the input words and the output words.

$$
\begin{aligned}
\left(2^{64}\right)! \quad &= \quad 10^{347380000000000000000} \\
&> \quad \left(10^{10^{20}}\right)
\end{aligned}
$$

Plaintext-to-Cipher text Mapping

- Now with a key size of 56 bits, we have a total of **2^{56} different keys**. Each **key** corresponds to **one of the 2^{64}!** different possible mappings. The number 2^{56} is upperbounded by **10^{17}**.
- **Therefore, it is reasonable to assume that if DES is used twice with different keys, it**
- **will produce one of the many mappings that are not defined by a single application**

Conclusion

❖ It is reasonable assume that if DES is used **twice with different keys**, it will produce one of the many mappings that are not defined by a **single application of DES.**

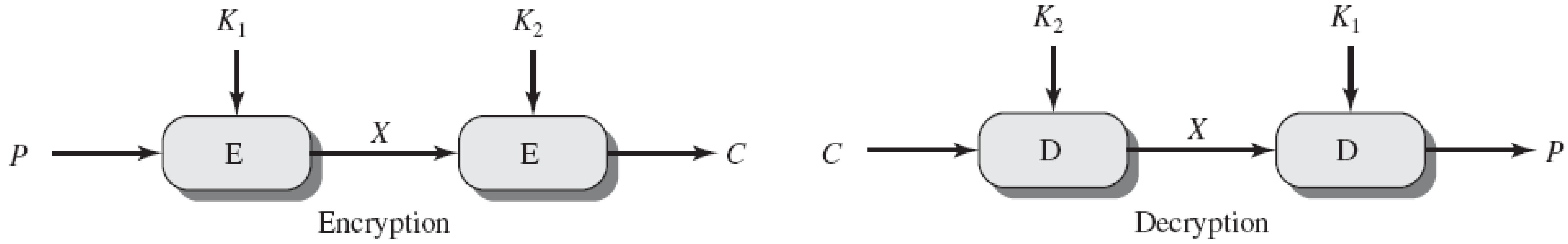
Vulnerability of Double DES to the Meet-in-the-Middle Attack

Vulnerability of Double DES

- Any double block cipher, that carries out double encryption of the plaintext using **two different keys** in order to increase the cryptographic strength of the cipher.
  - ✓ Cipher is **open** to what is known as the **Meet-in-the-Middle Attack.**

What do Meant by Meet-in-the-Middle Attack.

- Let's the see the relationship between the **plaintext P** and the **ciphertext C** for double DES:



$$C = E(K_2, E(K_1, P)) \qquad P = D(K_1, D(K_2, C))$$

- where K_1 and K_2 are the two 56-bit keys used in the two stages of encryption.

How Meet-in-the-Middle Attack occurs 2DES.

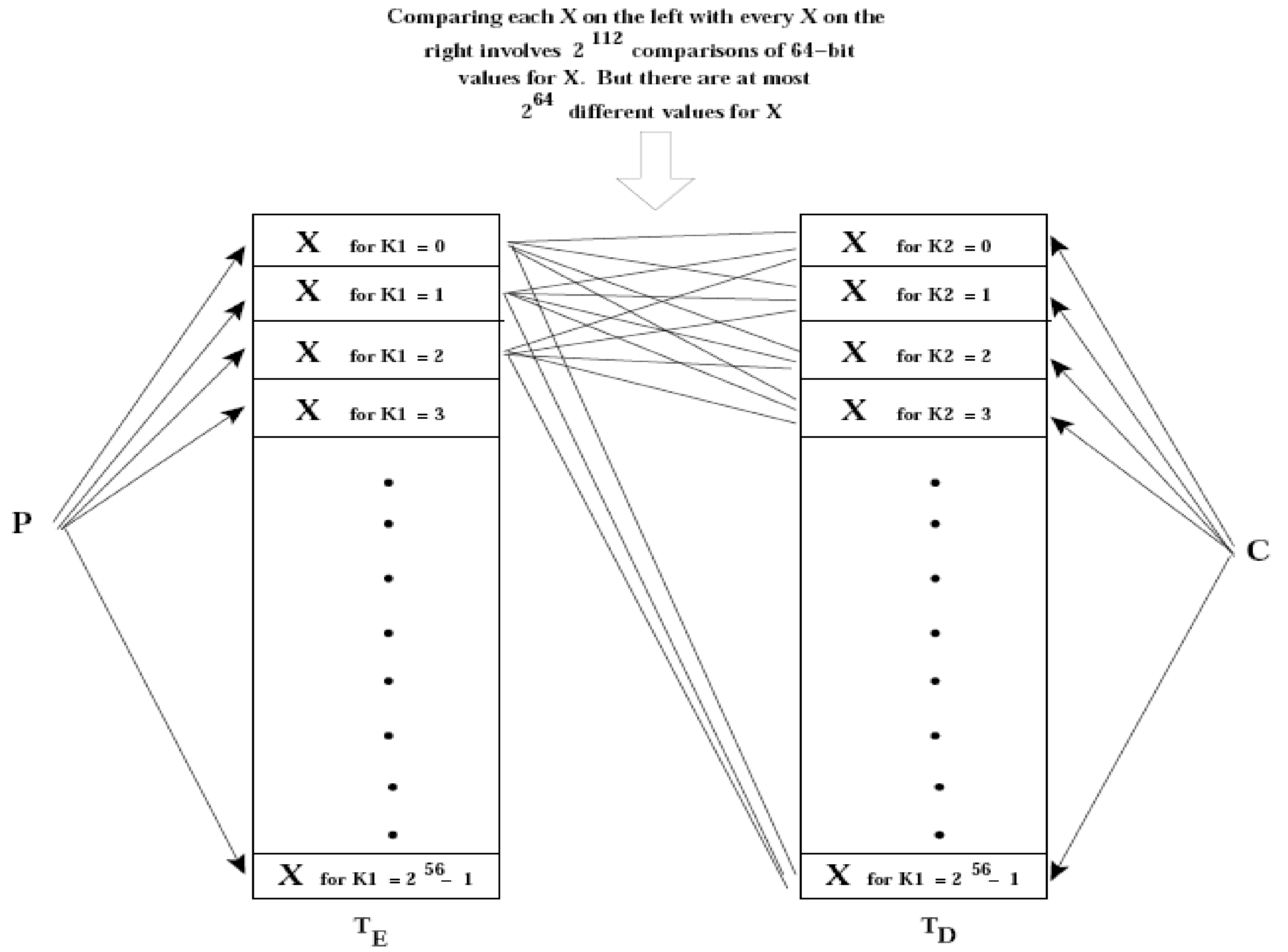
- Let's say that an attacker has available to him/her a **plaintext-ciphertext pair (P,C).**
- From the perspective of the attacker, there exists an X such that

$$\mathbf{X = E(K_1, P) = D(K_2, C)}$$

How Meet-in-the-Middle Attack occurs 2DES.

- In order to mount the attack, the attacker creates a **sorted table** of all possible value for X for a given P by trying **all possible 2^{56} keys**. This **table** will have **2^{56} entries**. We will refer to this table as **T_E.**
- The attacker also creates **another sorted table** of all possible X by decrypting C using every **one of the 2^{56} keys**. This **table** also has 2^{56} entries. Let's call this table **T_D.**

Attacker Comparison Table



How Meet-in-the-Middle Attack occurs 2DES.

- We need to make a total of 2^{112} comparisons in order to figure out which entries in the tables are the same.
- The comparisons involve only 2^{64} different possible values for X. (Recall that X is a 64-bit word.) Then it must be case that

$$\frac{2^{112}}{2^{64}} = 2^{48}$$

- we can expect **2^{48} entries** in the T_E table to be the **same** as the entries in the T_D entries in the T_D table

How Meet-in-the-Middle Attack occurs 2DES.

- Therefore, when we compare the **2^{56} entries of X in T_E** with the **2^{56} entries of X′ in T_D**, on the average we are likely to run into **2^{48} false alarms.**
- Therefore, the matching entry in comparing T'_E with T'_D is practically guaranteed to yield the **encryption keys K_1 and K_2.**

Modes of Operation

Use Of Modern Block Ciphers.

- Just because a block cipher has been demonstrated to be **strong** does **not imply** that it will be **sufficiently secure** if you are using it to transmit **long messages.**
  - ✓ By “long”, we mean many times **longer than** the block length.

Use Of Modern Block Ciphers.

- The interaction between the **block-size of ciphers** and **any repetitive structures in the plaintext** may still leave **too many clues** in the ciphertext that compromise its **security.**

Modes of Operation

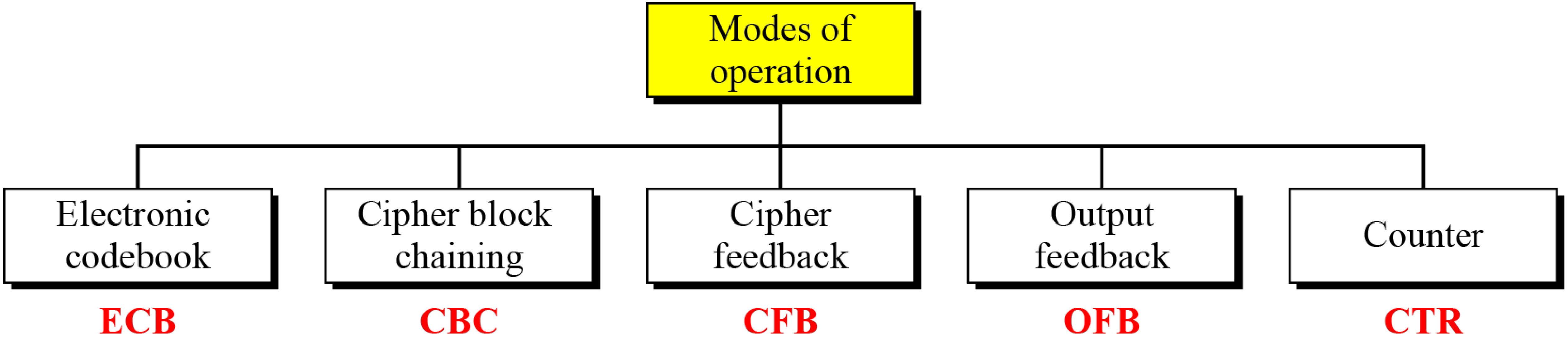
- Symmetric-key Encipherment can be done using modern block ciphers.
  - ✓ **Modes of Operation** have been devised to encipher text of **any size(Long Message)** employing either DES or AES.

Modes of Operation

❖ The goal of this section is to present the **five different modes** in which **any block cipher** can be used.

1. **Electronic Codebook (ECB) Mode**
2. **Cipher Block Chaining (CBC) Mode**
3. **Cipher Feedback (CFB) Mode**
4. **Output Feedback (OFB) Mode**
5. **Counter (CTR) Mode**

Five Modes of Operation



Electronic Codebook (ECB) Mode

Electronic Codebook (ECB) Mode

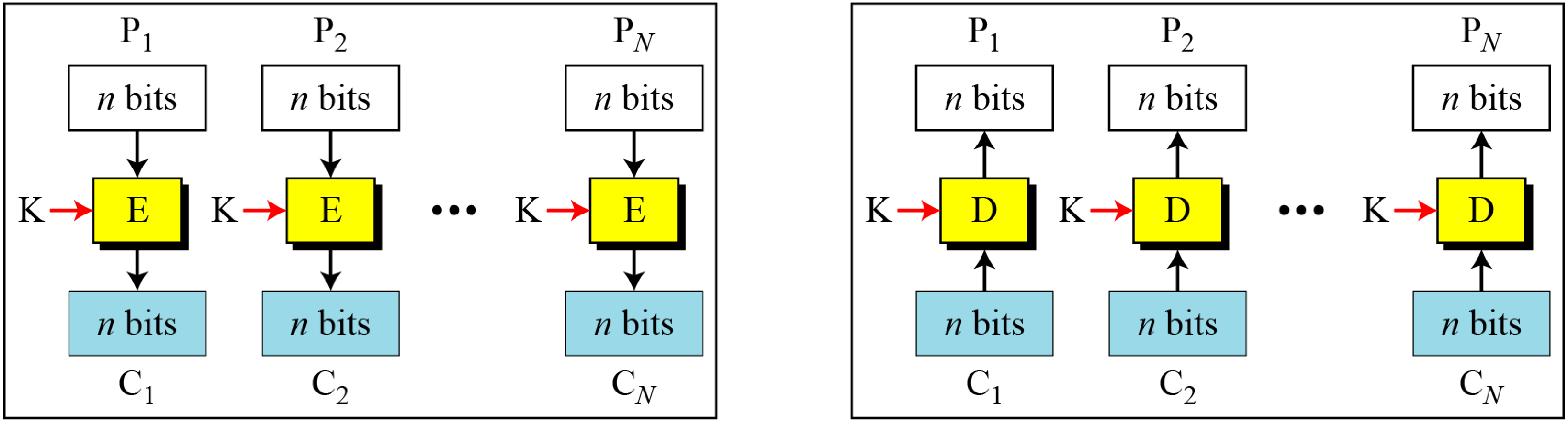
- The simplest mode of operation is called the **Electronic Codebook (ECB) Mode.**
- In this method, the encryption process is represented with **fixed mapping** between the **input blocks of plaintext** and the **output blocks of cipher text.**
  - ✓ Similar to the **Code Book Approach**

Electronic Codebook (ECB) Mode

- The Code Book Approach would **list the ciphertext mapping** for **each plaintext word.**

Electronic Codebook (ECB) Mode

E: Encryption
D: Decryption
P_i: Plaintext block i
C_i: Ciphertext block i
K: Secret key



Encryption

Decryption

Electronic Codebook (ECB) Mode

- In this ECB mode,
  - Each block of plaintext is **coded independently** which is **not secure** for **long segments** of plaintext.
    - Especially plaintext containing **repetitive information**
- This ECB mode used primarily for **secure transmission of short pieces of information**, such as an **encryption key.**

Electronic Codebook (ECB) Mode

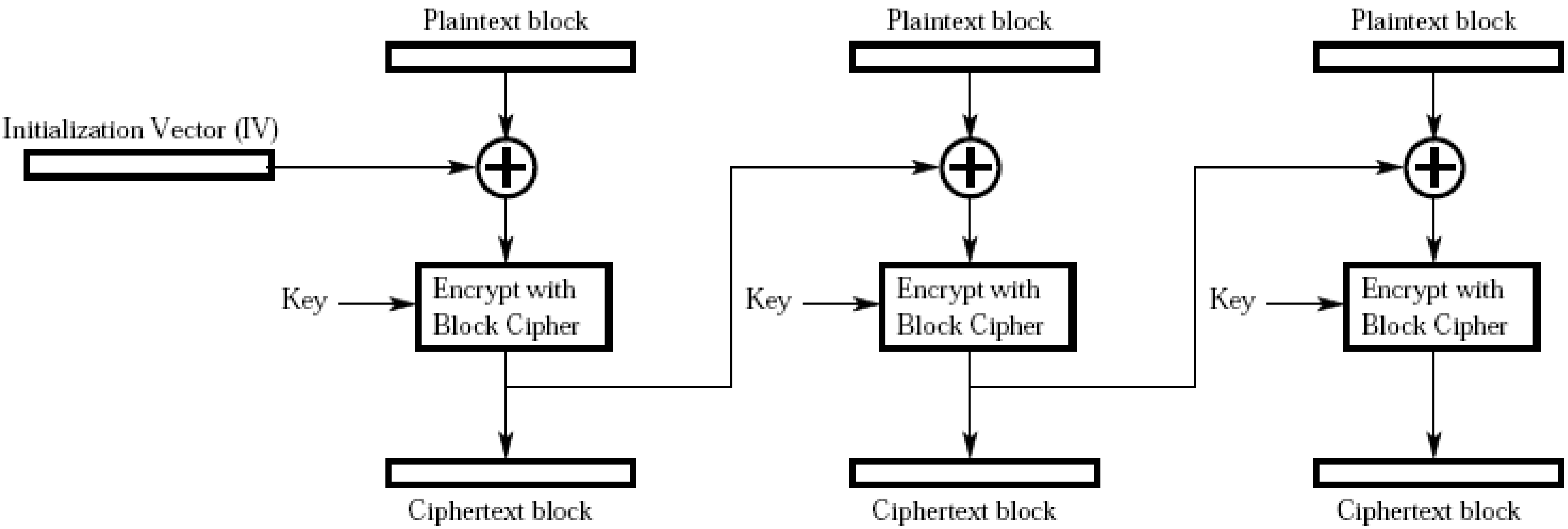
- Another **shortcoming of ECB** is that the **length** of the plaintext message must be **integral multiple of the block size**. When that condition is not met, the plaintext message must be **padded** appropriately.

The Cipher Block Chaining Mode (CBC)

The Cipher Block Chaining Mode (CBC)

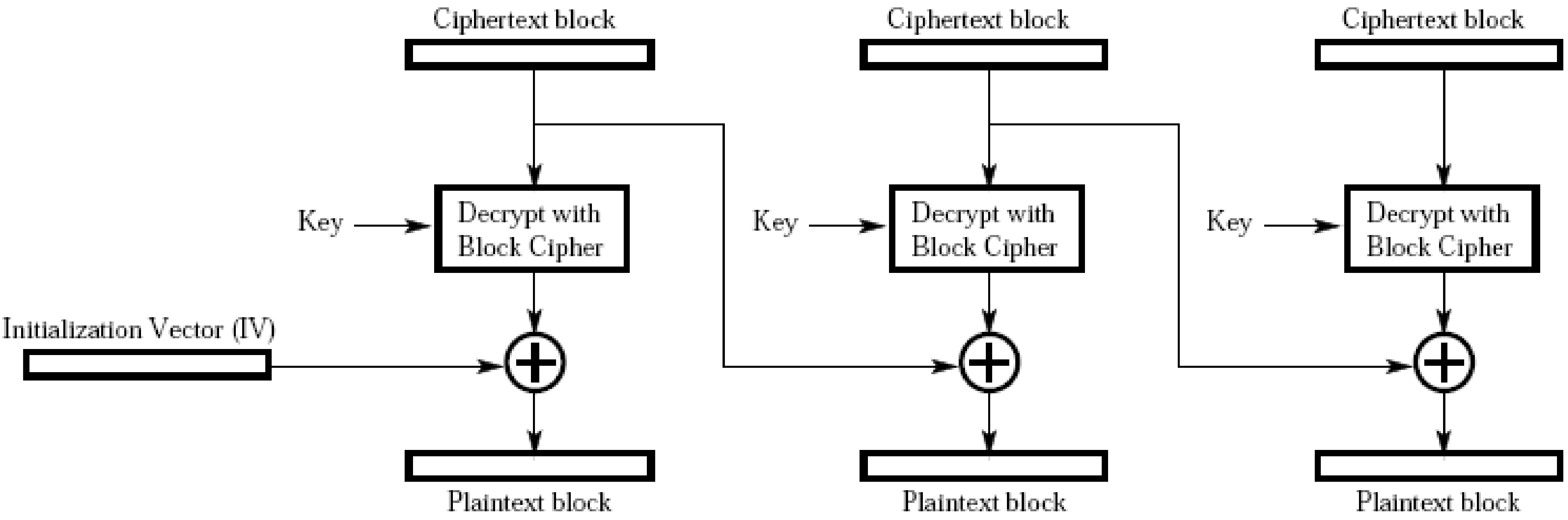
- This CBC Method will **overcome** the **security deficiency** of the ECB mode.
- In this Method, the **input to the encryption** algorithm consists of the **XOR of the plaintext block** and the **ciphertext produced** from the **previous plaintext block.**

The Cipher Block Chaining Mode (CBC): Encryption



CBC Encryption

The Cipher Block Chaining Mode (CBC): Decryption



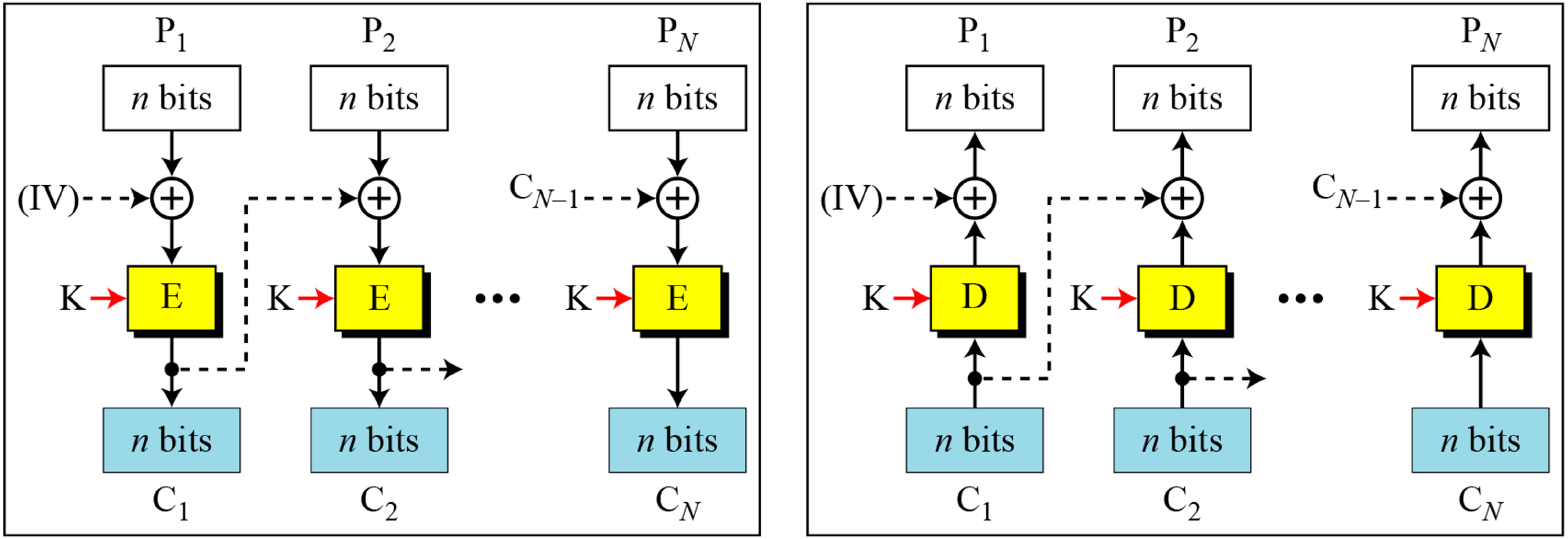
CBC Decryption

The Cipher Block Chaining Mode (CBC)

- This chaining scheme makes use of **initialization vector** for the first invocation of the encryption algorithm.
  - ✓ The **initialization vector** is sent separately as a **short message** using **the ECB mode.**
- In this Method, the ciphertext block for any given plaintext block becomes a **function of all the previous ciphertext blocks.**

The Cipher Block Chaining Mode (CBC)

E: Encryption
D : Decryption
P_i: Plaintext block i
C_i : Ciphertext block i
K: Secret key
IV: Initial vector (C_0)



Encryption

Decryption

Security of CBC

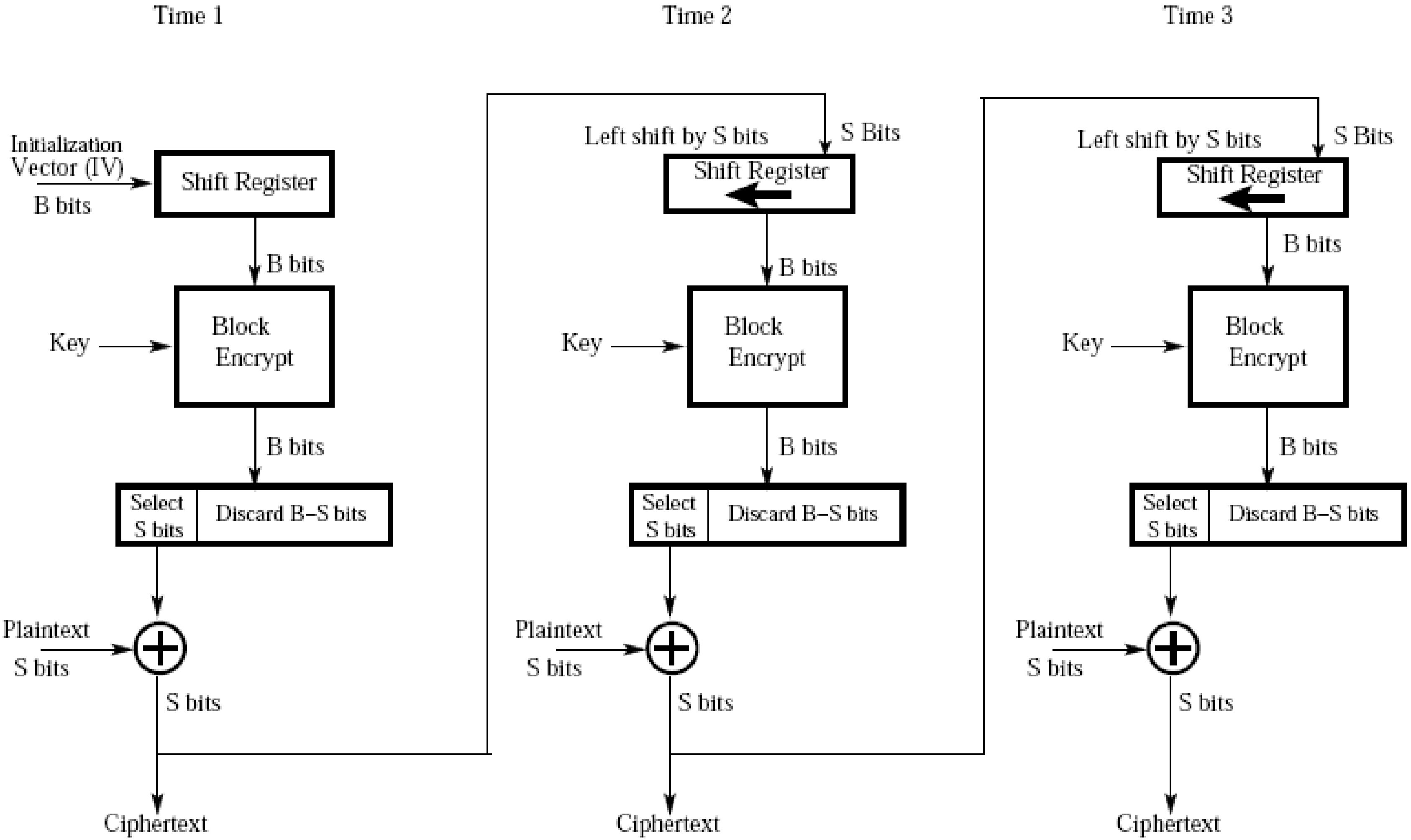
- This Method makes more difficult for a **cryptanalyst to break the code**
  - ✓ It look for patterns (known structure of the plaintext) in the ciphertext.

The Cipher Feedback Mode (CFB)

The Cipher Feedback Mode (CFB)

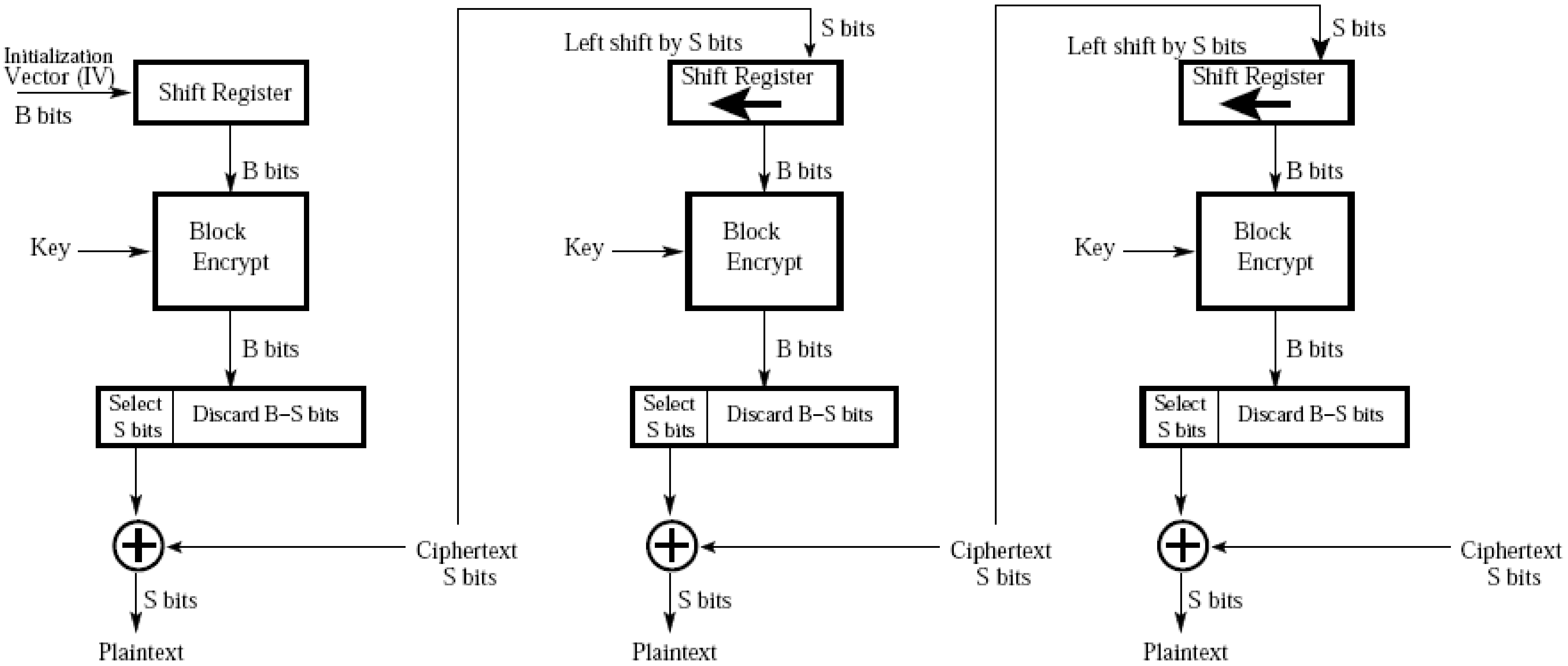
- In **Block Cipher**, if the **length of the message** is **not** an **integral number of blocks** then you must **pad the message.**
- In some situations, we need to use **DES or AES** as secure ciphers, but the plaintext or ciphertext **block sizes** are to be **smaller.**
- This approach allows a block cipher to be used as a **stream cipher**

Working Model of CFB



CFB Encryption

Working Model of CFB



CFB Decryption

Working Model of CFB

1. Start with an **Initialization Vector(IV)** of the same size as the block size expected by the block cipher.
   - ✓ The Initialization Vector(IV) is stored in **shift register**
2. **Encrypt** the Initialization Vector(IV) with the block cipher encryption algorithm.
3. **Retain only one byte** from the output of the encryption algorithm.
   - ✓ **Keep** the most significant byte. **Discard** the rest of the output.

Working Model of CFB

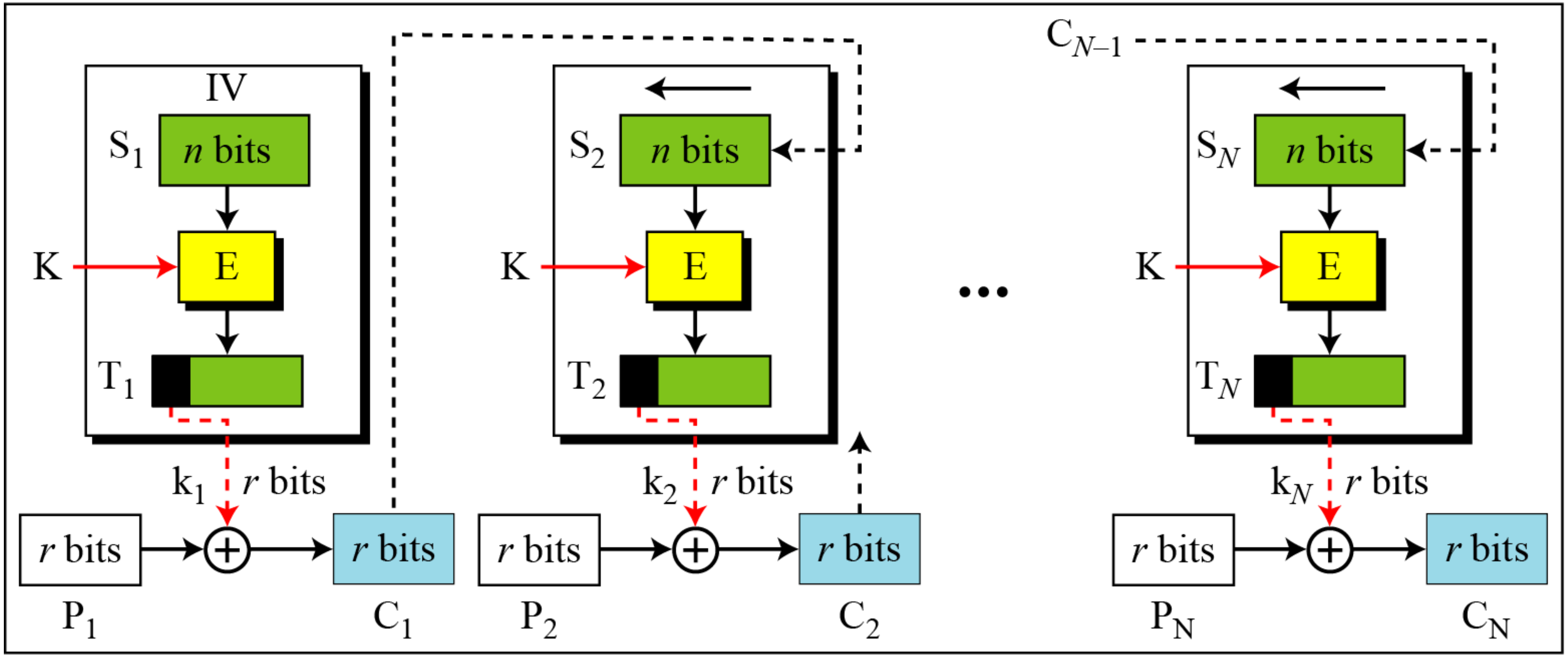
5. **XOR** the **byte retained** with the **byte of the plaintext** that needs to be transmitted.
   - ✓ Transmit the output byte produced.
6. **Shift the IV one byte** to the **left** (**discarding the leftmost byte**) and **insert the ciphertext byte** produced by the **previous step** as the rightmost byte. So the **new IV is still of the same length** as the block size expected by the encryption algorithm.

Working Model of CFB

5. Go back to the step "**Encrypt the IV** with the block cipher encryption algorithm".

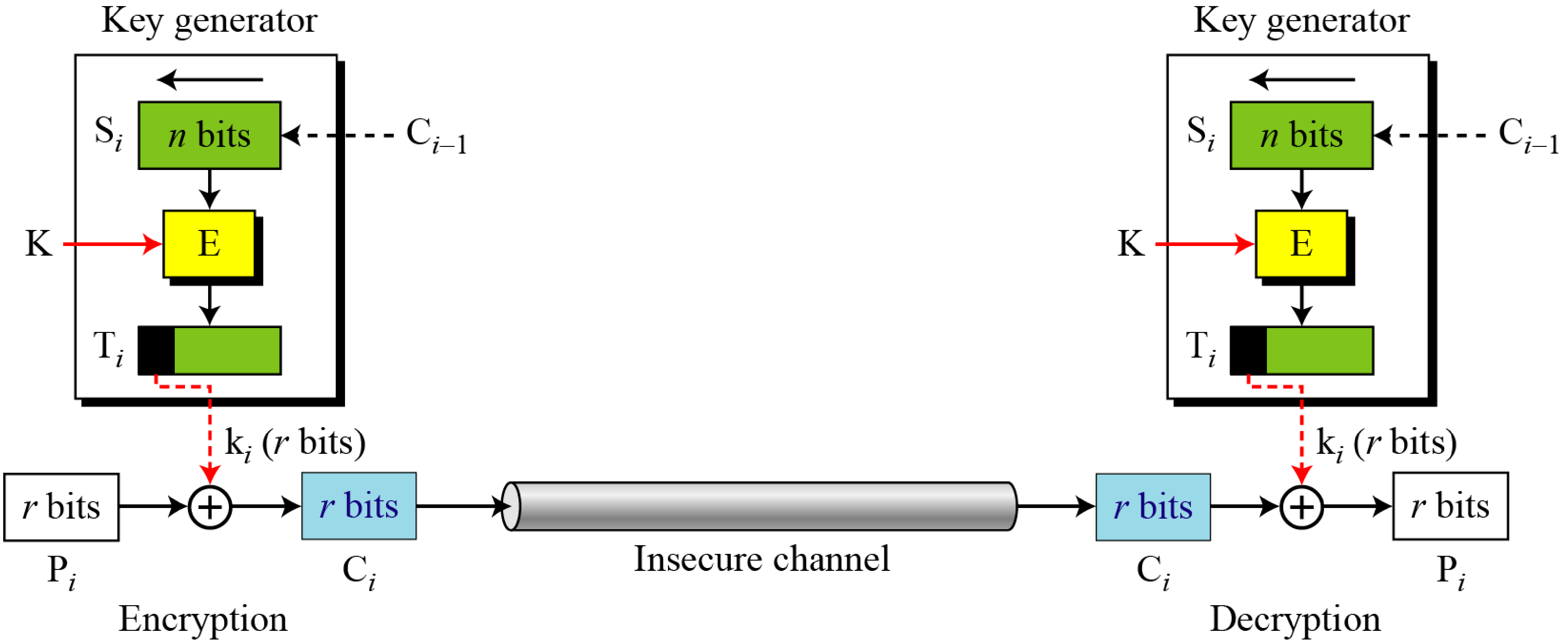
Working Model of CFB

E : Encryption
P_i: Plaintext block *i*
K: Secret key
D : Decryption
C_i: Ciphertext block *i*
IV: Initial vector (S_1)
S_i: Shift register
T_i: Temporary register



Encryption

Cipher feedback (CFB) mode as a Stream Cipher



Observation

- The most important in this method is that **only the encryption algorithm** is used in **both encryption and decryption.**
  - ✓ **Implementation-level** will be easier
- Note that the **ciphertext byte produced** for any plaintext byte **depends** on all the **previous plaintext bytes** in the CFB mode.

Observation

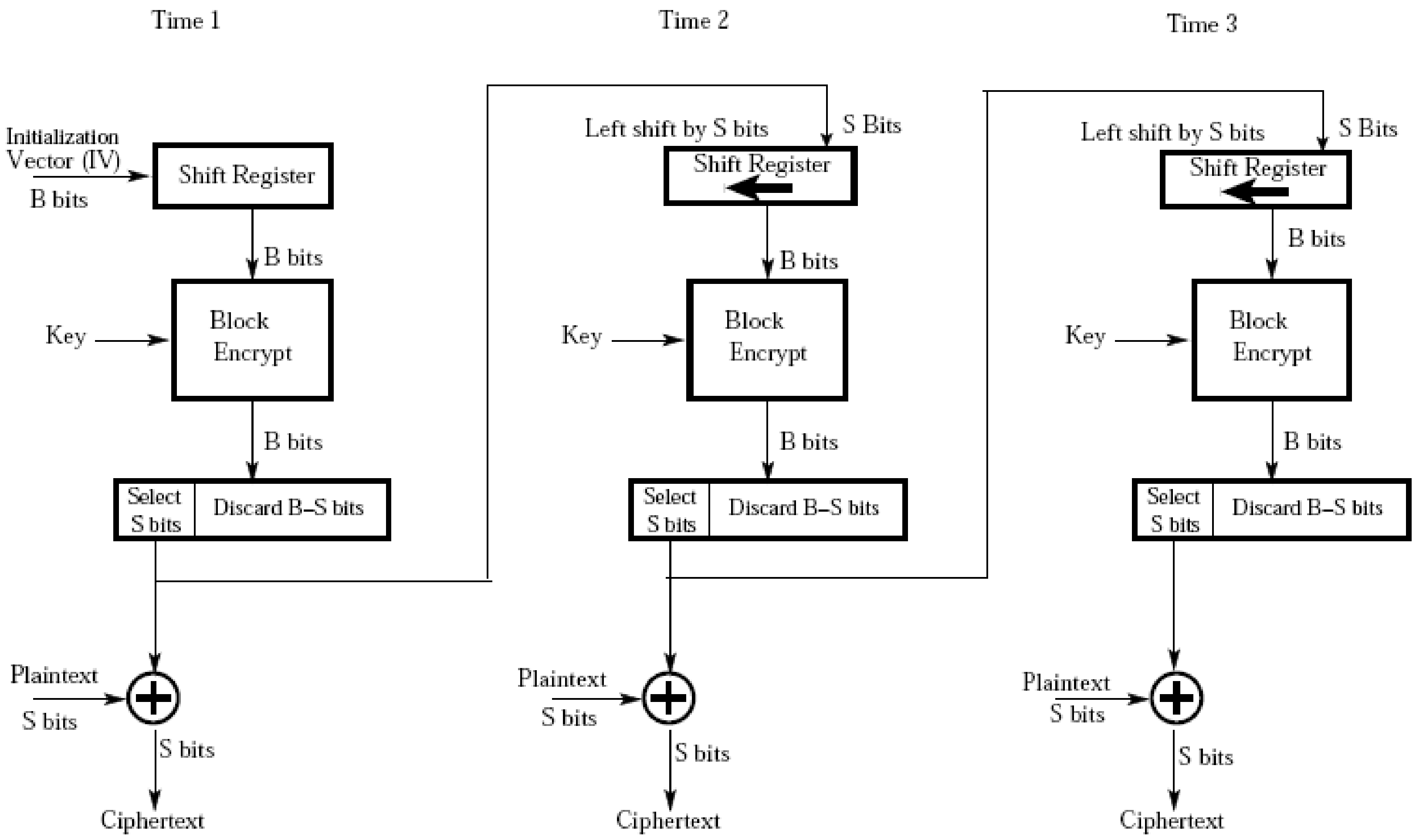
- Suppose we have **encrypted and transmitted** the first byte of plaintext. Assume this **byte** is received with a **one or more bit errors**.
- Hence, Decryption of the **first byte is erroneous**, that error will also **propagate to downstream decryptions** because the received **ciphertext byte is also fed back into the decryption** of the next byte.

The Output Feedback Mode (OFB)

The Output Feedback Mode (OFB)

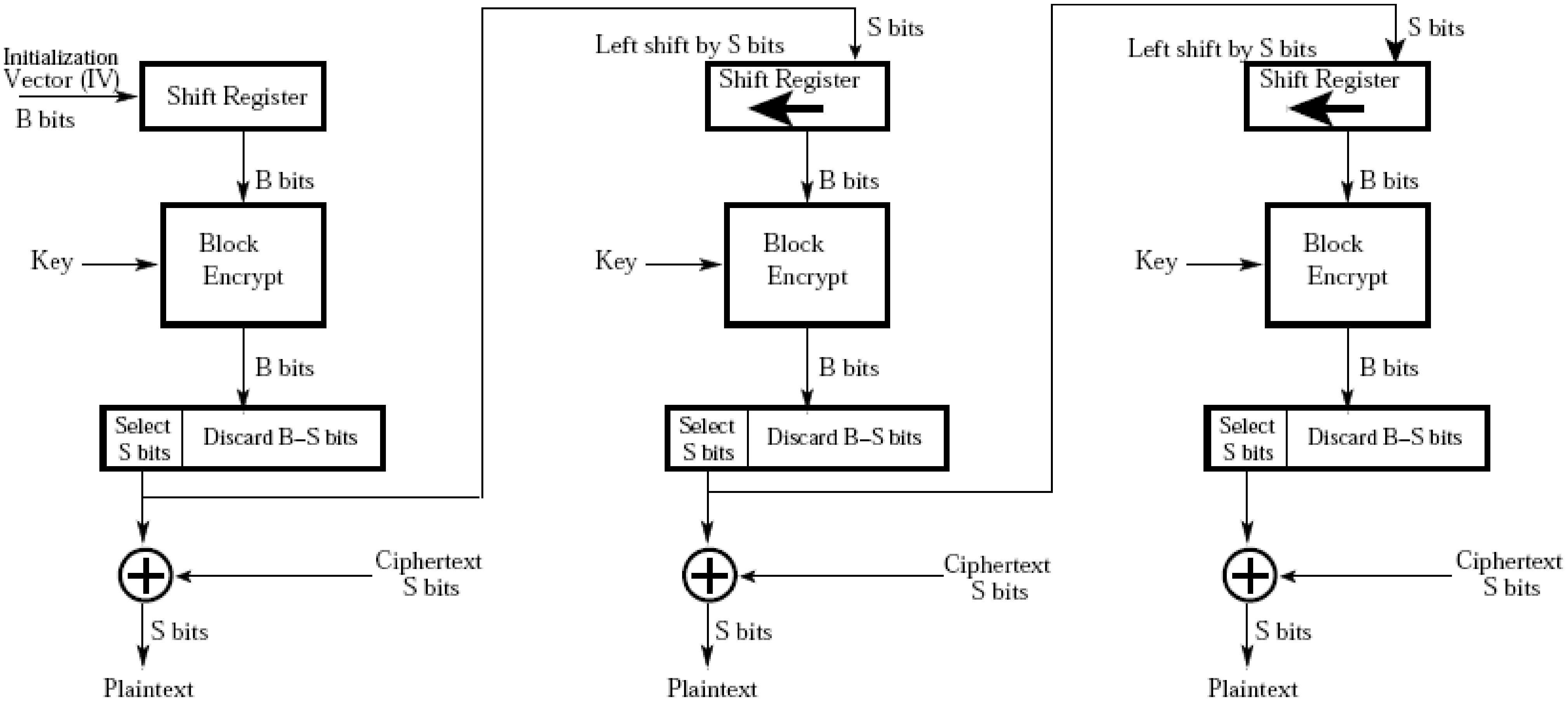
- The Output Feedback Mode (OFB) is **very similar** to the CFB mode. Therefore, this scheme can also be used as a **stream cipher.**
- The only difference between CFB and OFB is that
  - ✓ we **feed back one byte (the most significant byte)** from the **output of the block cipher** encryption algorithm, Instead of feeding back the **actual ciphertext byte.**
  - ✓ OFB makes more resistant to **transmission bit errors.**

The Output Feedback Mode (OFB)



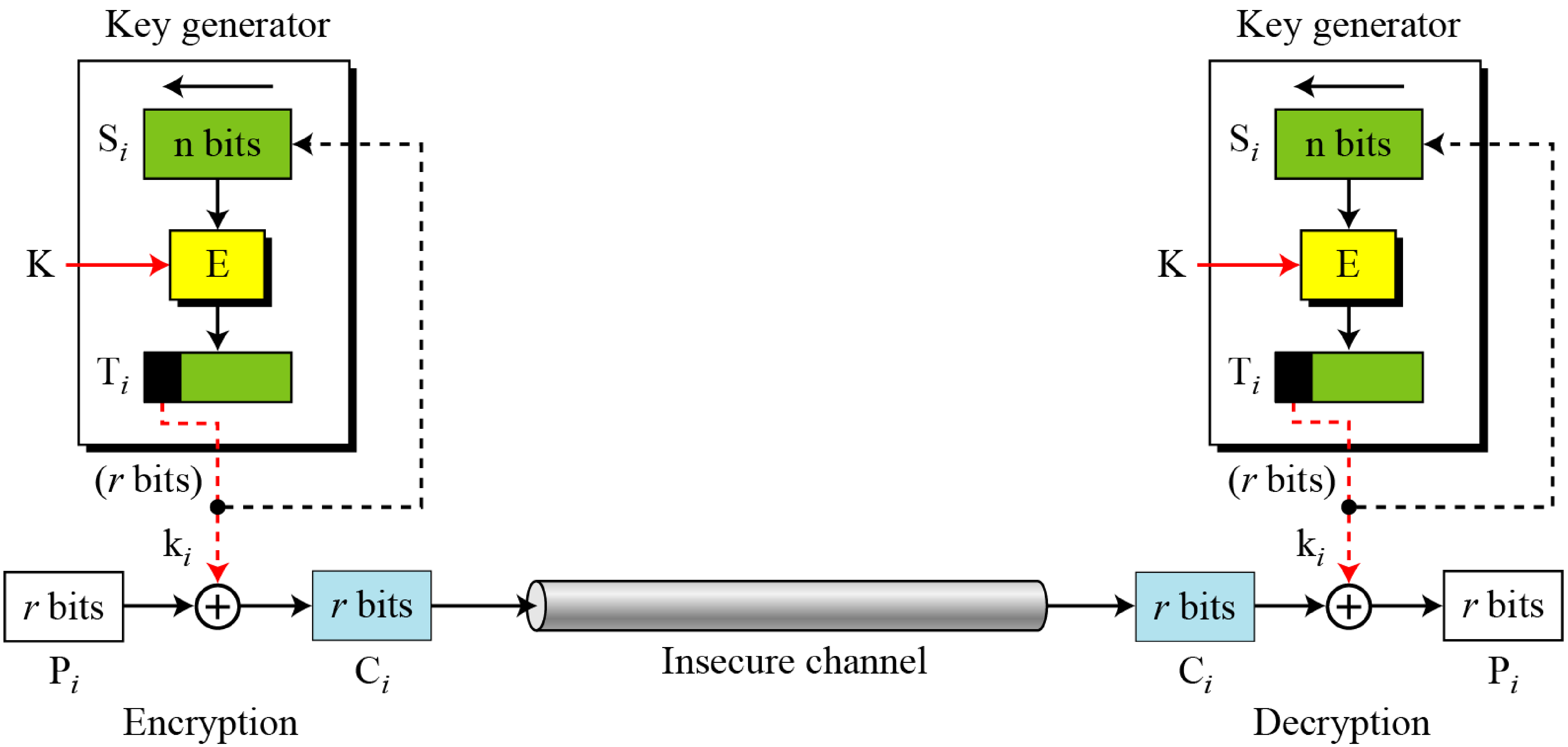
OFB Encryption

The Output Feedback Mode (OFB)



OFB Decryption

The Output Feedback Mode (OFB)



The Counter Mode (CTR)

The Counter Mode (CTR)

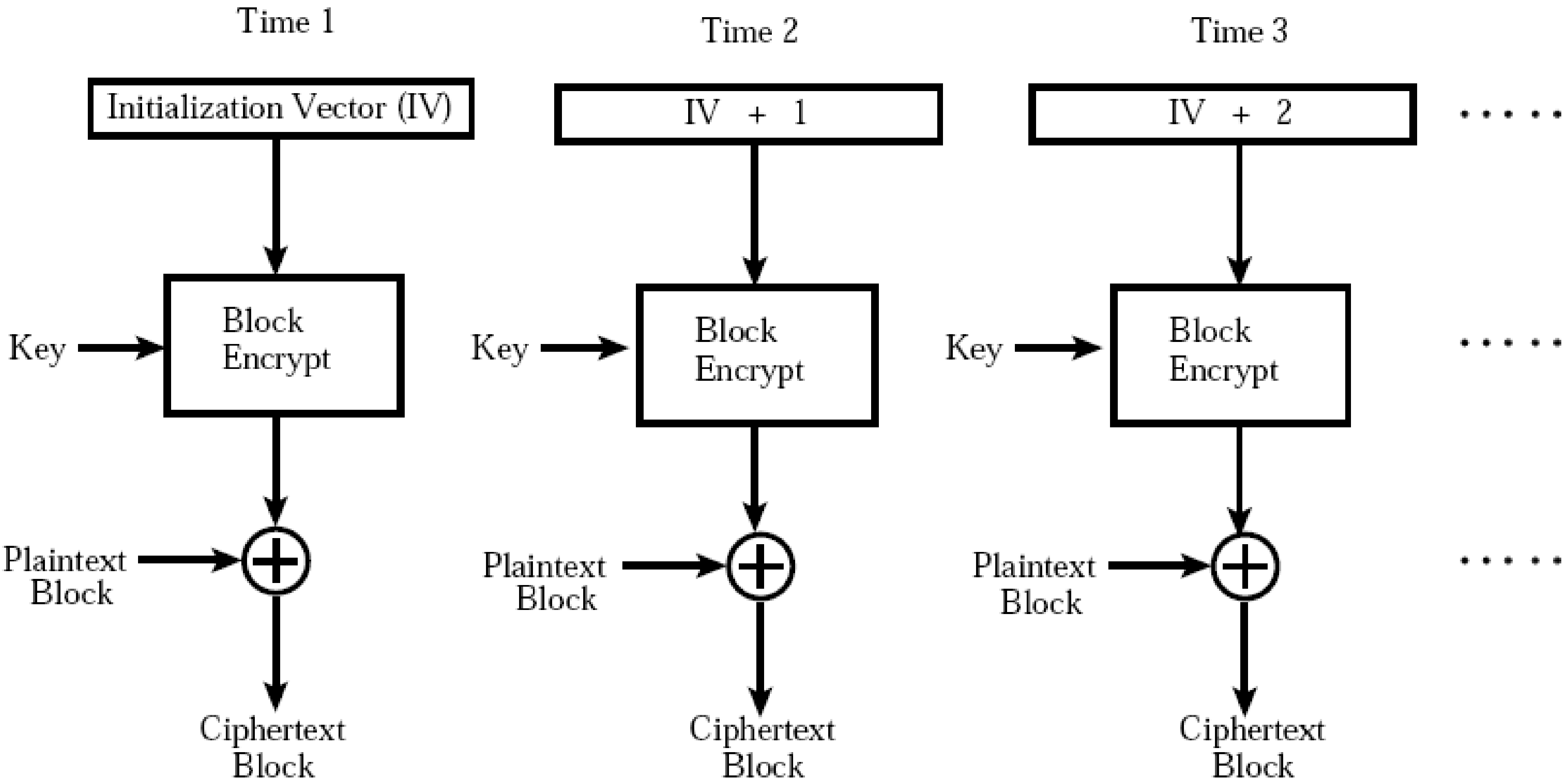
- Whereas the previous two modes, CFB and OFB, are intended to use a block cipher as a **stream cipher.**

- The counter mode (CTR) **retains the pure block structure** relationship between the **plaintext and ciphertext.**

  - ✓ For each **b-bit input plaintext block**, the scheme **produces an b-bit ciphertext block.**

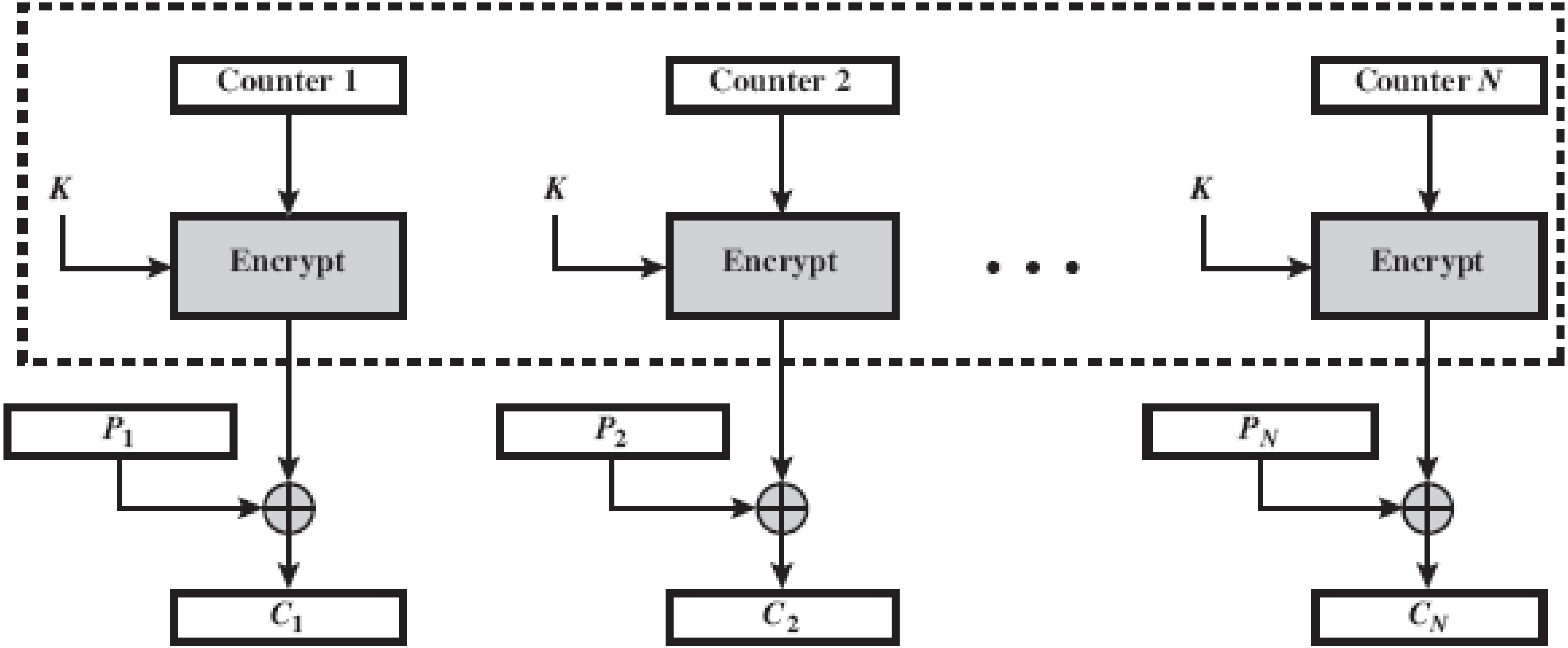
  - ✓ The block cipher encryption algorithm carries out a **b-bits to b-bits transformation.**

The Counter Mode (CTR)



CTR Encryption

The Counter Mode (CTR)



(a) Encryption

The Counter Mode (CTR)

- In this method, **no part of the plaintext** is directly **exposed** to the block encryption algorithm.
- The encryption algorithm **encrypts only a b-bit integer** produced by the **counter.**
- Next, **transmitted the integer to the XOR** of the encryption and the **b bits of the plaintext** which produces the **cipher text.**

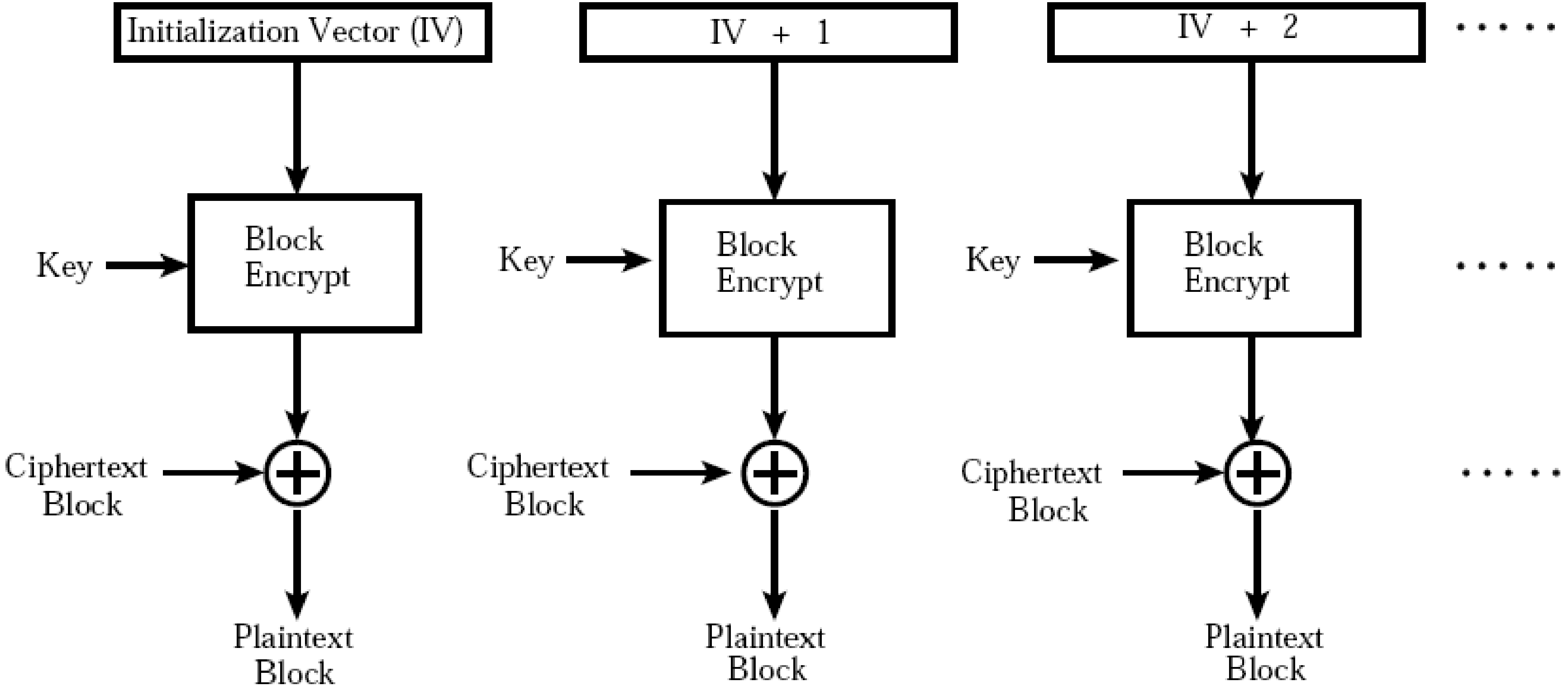
The Counter Mode (CTR)

- The **pseudo randomness** is used in the counter. To generate the counter value,
  - ✓ we start with **some number (pseudo randomness)** for the first plaintext block and then **increment** this value modulo 2^b from block to block

The Output Feedback Mode (OFB)

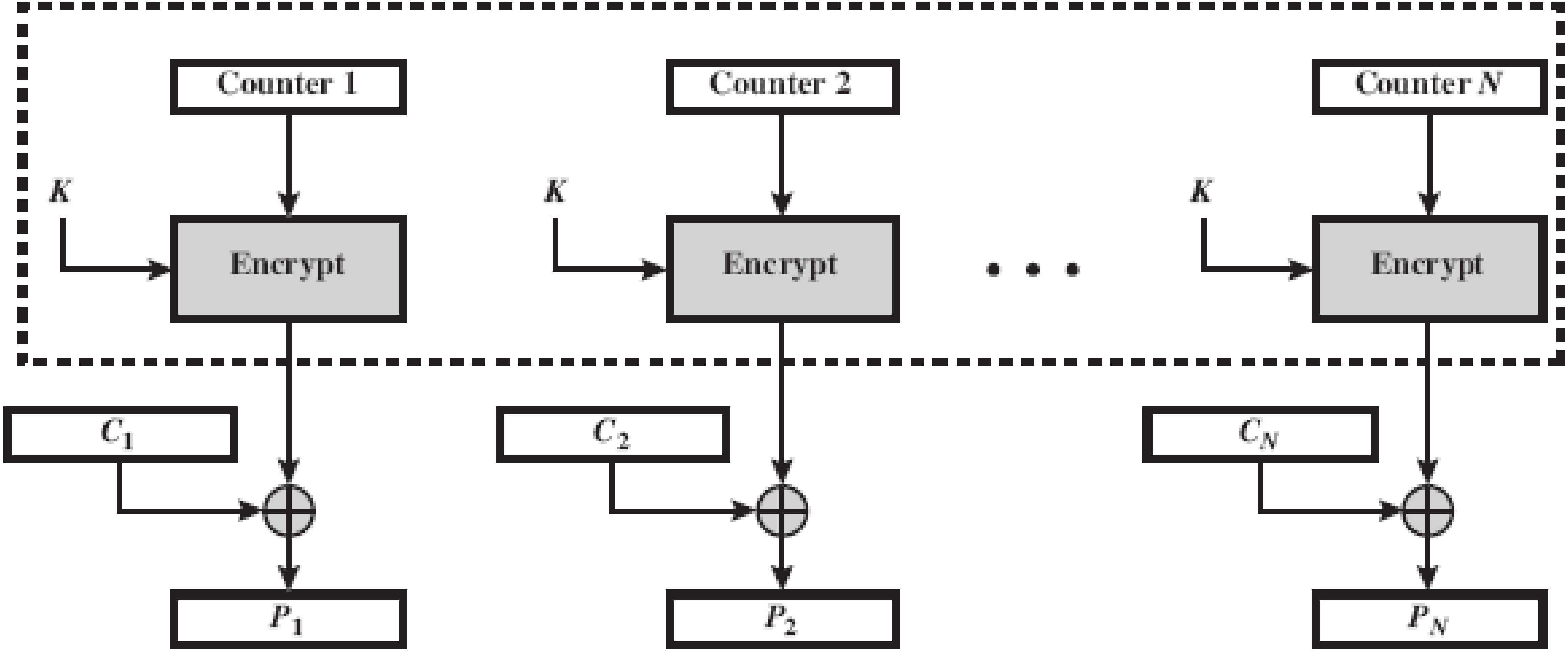
- This Method only uses the **forward encryption algorithm** for both **encryption and decryption.**

The Counter Mode (CTR)



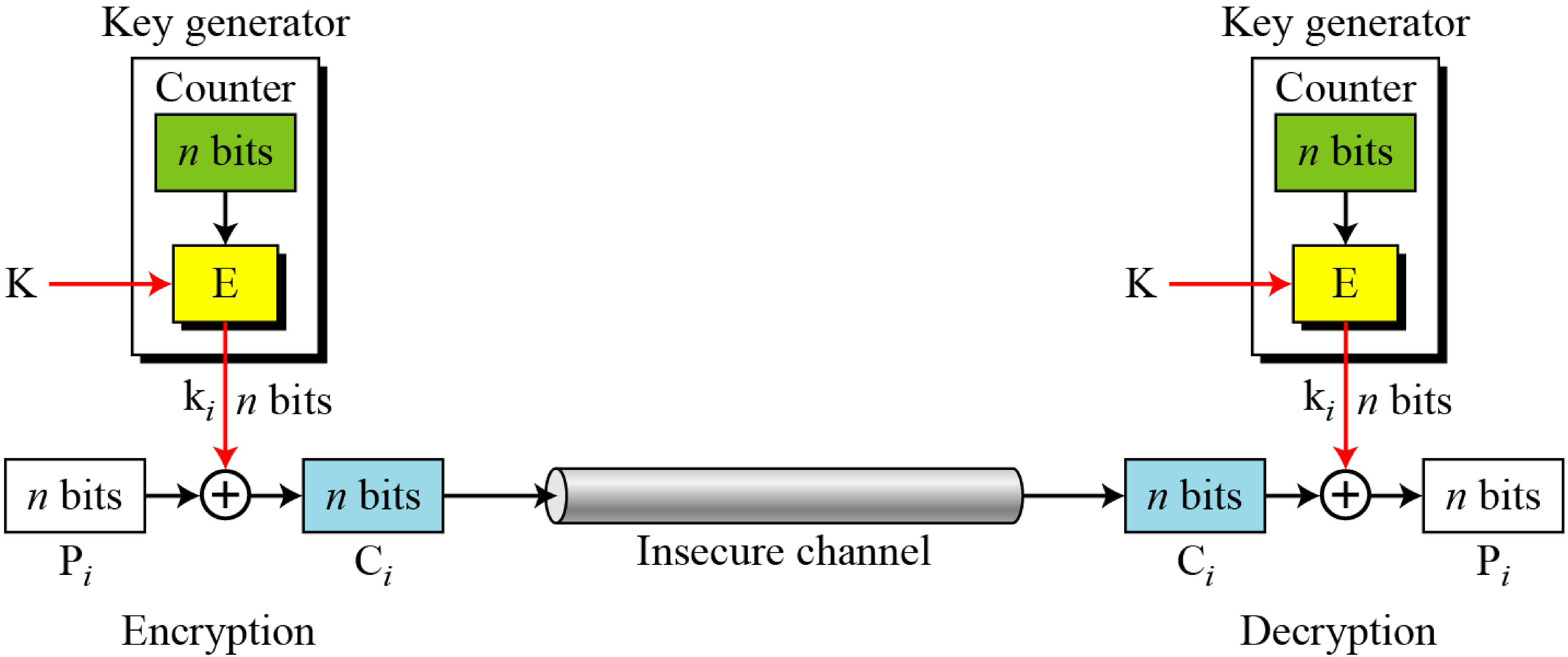
CTR Decryption

The Counter Mode (CTR)



(b) Decryption

The Counter Mode (CTR)

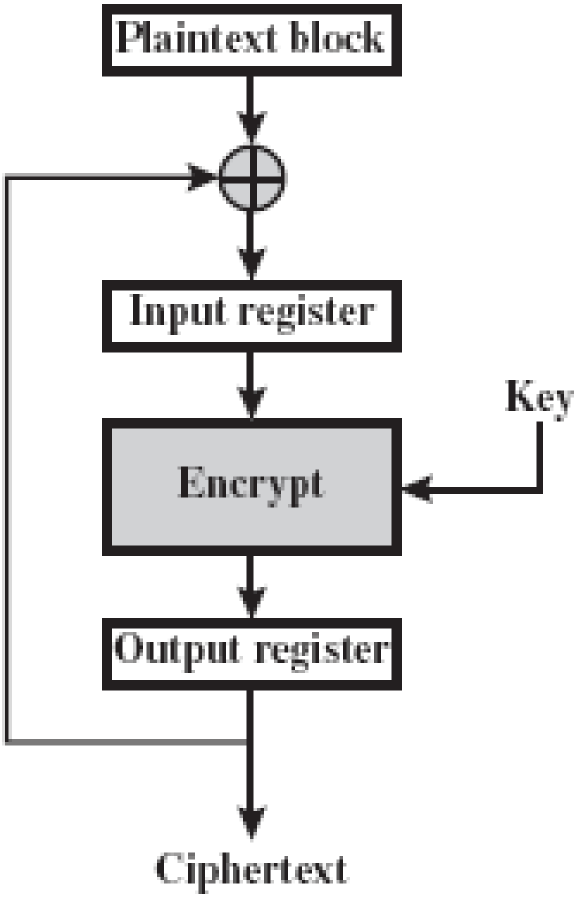


Advantages of the Counter Mode (CTR)

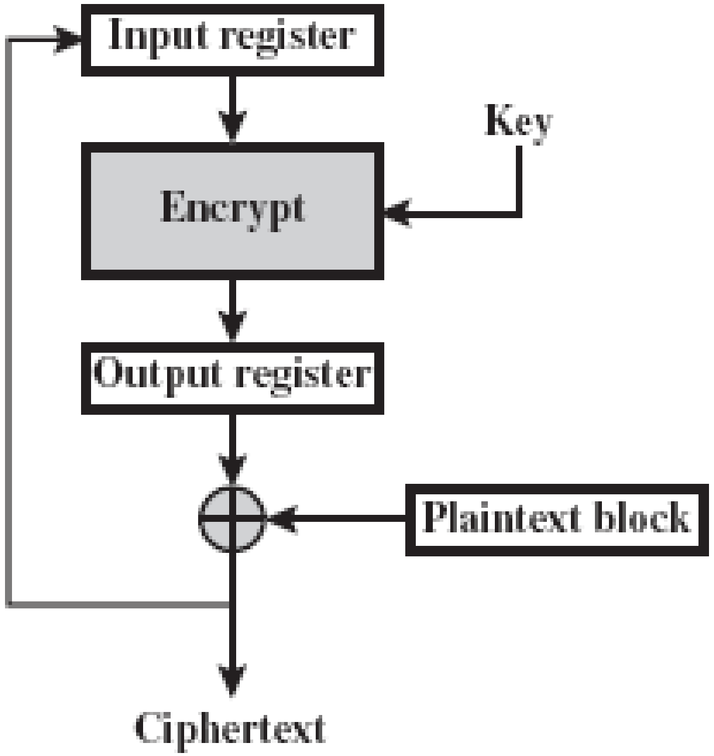
- **Fast** Encryption and Decryption.
- **CTR is as secure** as the **other four modes** for using block ciphers.
- In this method, there is **no block-to-block feedback**, the algorithm is highly amenable to **implementation on parallel machines.**

Summary

Summary

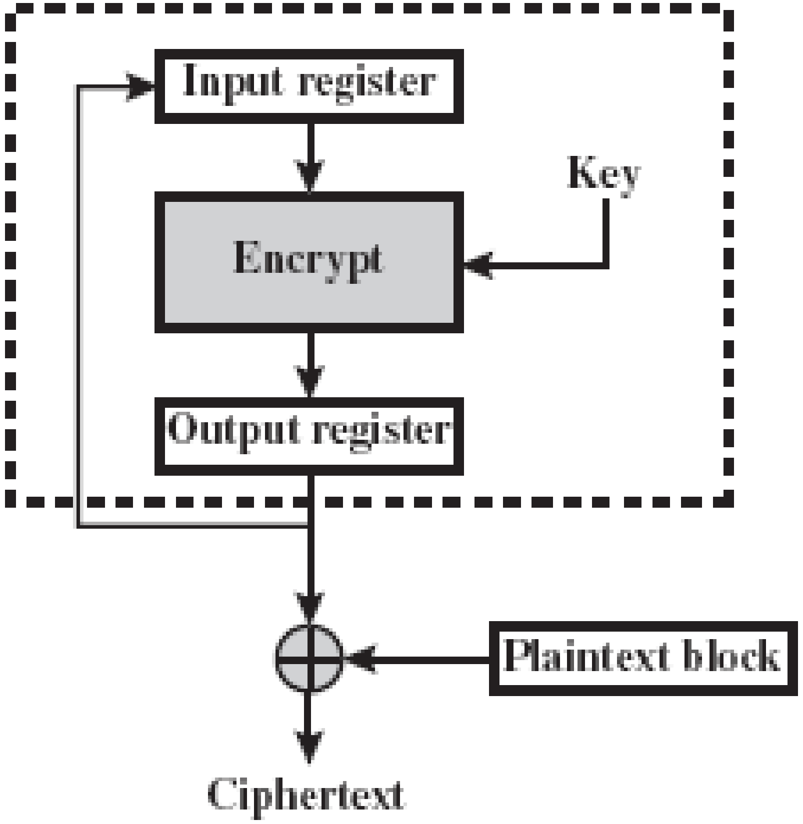


(a) Cipher block chaining (CBC) mode

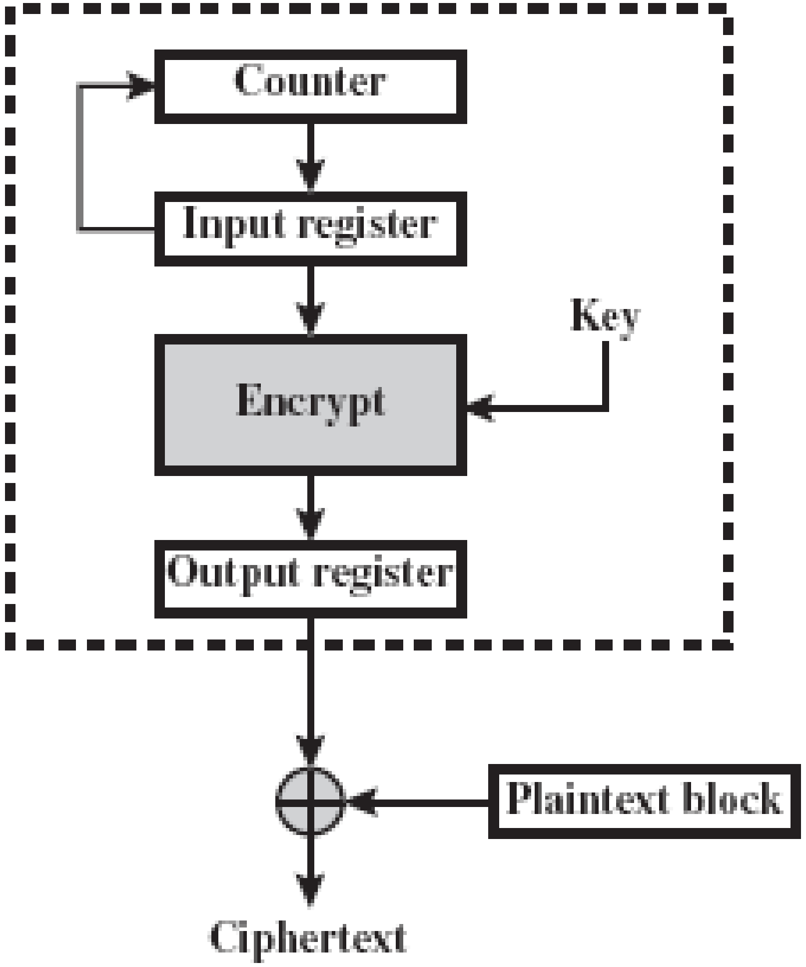


(b) Cipher feedback (CFB) mode

Summary



(c) Output feedback (OFB) mode



(d) Counter (CTR) mode

Goals:

- To present Double-DES and its vulnerability to the meet-in-the-middle attack
- To present two-key Triple-DES and Triple-DES
- To present the five different modes in which a block cipher can be used in practical systems for secure communications

Thank U