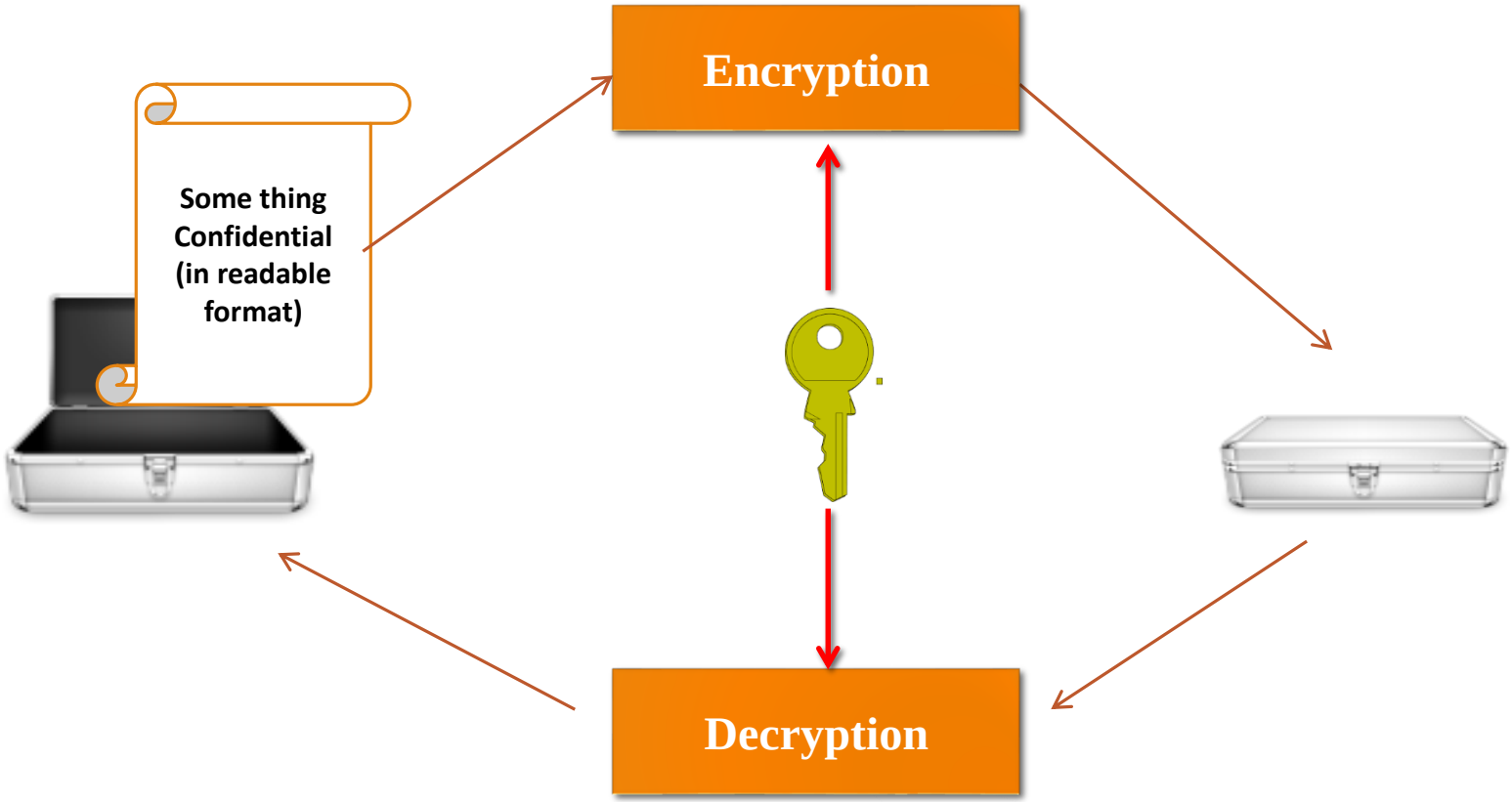
Symmetric Key Encryption

CSS3232-Data & Network Security

K.D.C.G KAPUGAMA – DEPARTMENT OF COMPUTER SCIENCE, UNIVERSITY OF RUHUNA

Symmetric Key Cryptograms



The Classical Cryptography

- The **Encryption Algorithm** and the **related key** are kept secret.
- As there are large number of possible keys, it is hard to break the system.
- Example: For **128 bit** key
  - Using brute force search

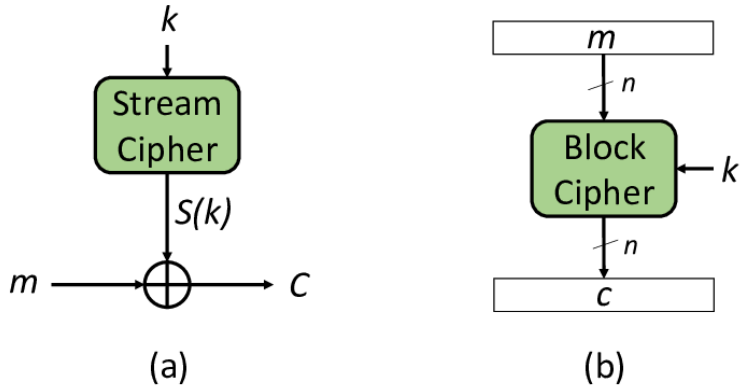
$$2^{128} \approx 10^{38}$$

Keys to search

- The fundamental difficulty is key distribution to parties who want to exchange messages

Stream Vs. Block Ciphers

- **Stream Ciphers:**
  - Encrypt/Decrypt digital data **one bit or one byte** at a time.
- **Block Ciphers:**
  - Process data in **blocks.**
  - Each block is considered as a whole and perform encryption and decryption.

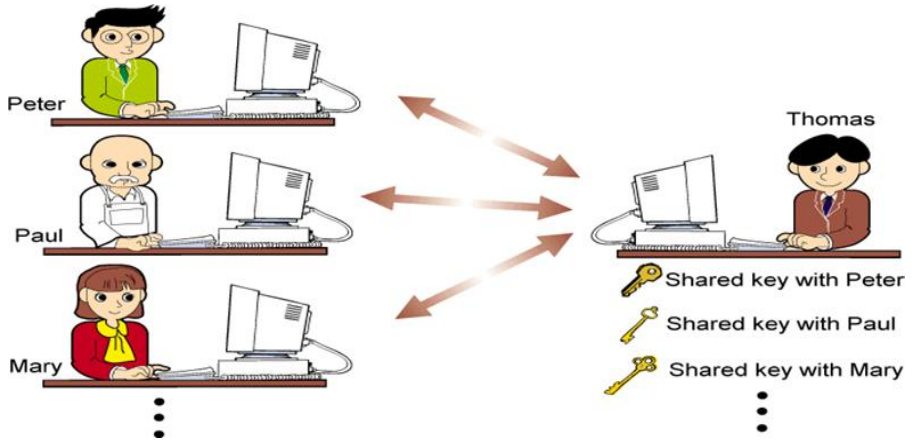


Symmetric Key / Private Key cryptosystem

- Uses a single Private Key shared between users
- **Strengths:**
  - Speed and Efficient Algorithms – Relatively much quicker than Asymmetric Key cryptosystems.
  - When a large Key size is in use, it is hard to break.
  - Ideal for bulk encryption/decryption

Symmetric Key / Private Key cryptosystem

- **Weaknesses:**
  - Poor Key distribution - must be done out of band (different communication medium) e.g. Phone, Email
  - Poor Key Management/ Scalability – Each user needs a unique key



    - For n number of users $n(n-1)/2$
  - Unable to provide authenticity or non – repudiation. Confidentiality only.

Requirements for Symmetric Key Cryptography

- Two requirements for secure use of symmetric encryption.
  1. A strong **encryption algorithm**.
  2. The sender and receiver only know a secret key K.

$$Y = E(X, \boldsymbol{K})$$
$$X = D(Y, \boldsymbol{K})$$

- Assume Encryption algorithm is known
- Implies a secure channel to distribute key.

Data Encryption Standard (DES)

- Most widely used block cipher in the world.
- Adopted in 1977 by NBS (now NIST) as FIPS PUB 46.
- Has been the subject of considerable controversy over its security

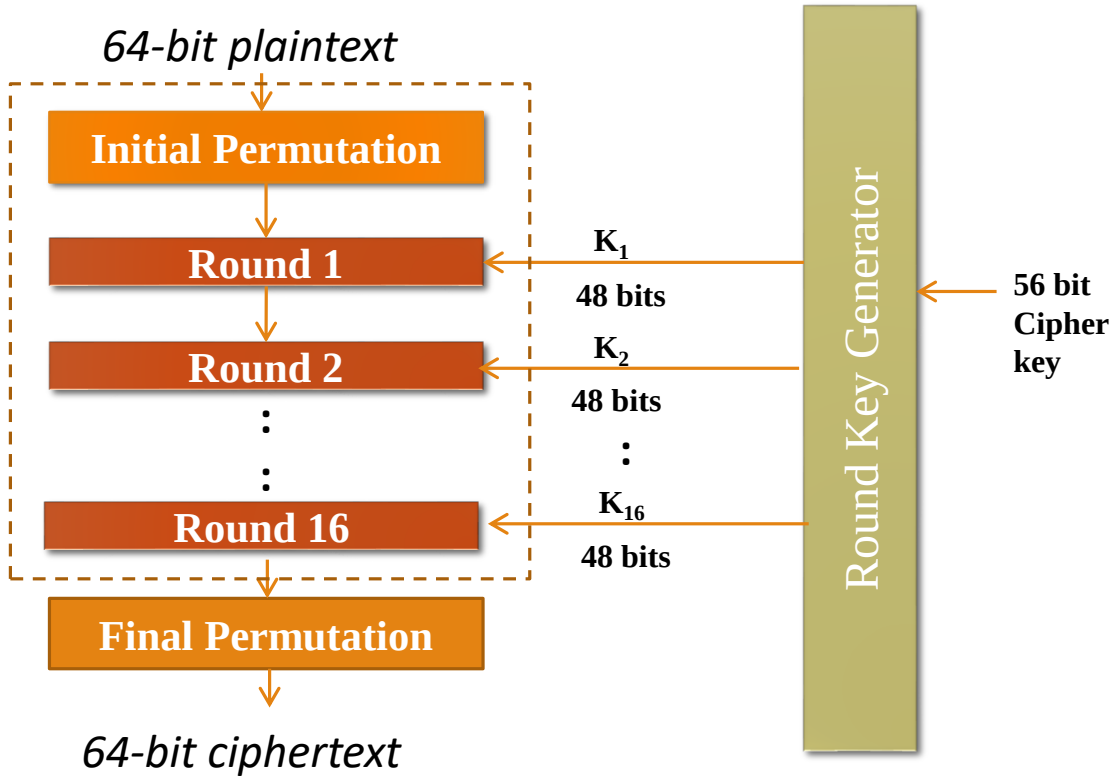
Data Encryption Standard (DES)

- **Overview**
  - Block Cipher: Plain Text is encrypted in **64-bit** blocks.
    - Message is padded to have total message size that is a multiple of **64 bits**
  - Normal key length : **56-bit**.
    - Though the key size is 64 bits, 8 of the 64 bits are not used in the encryption algorithm. (function as check bits)

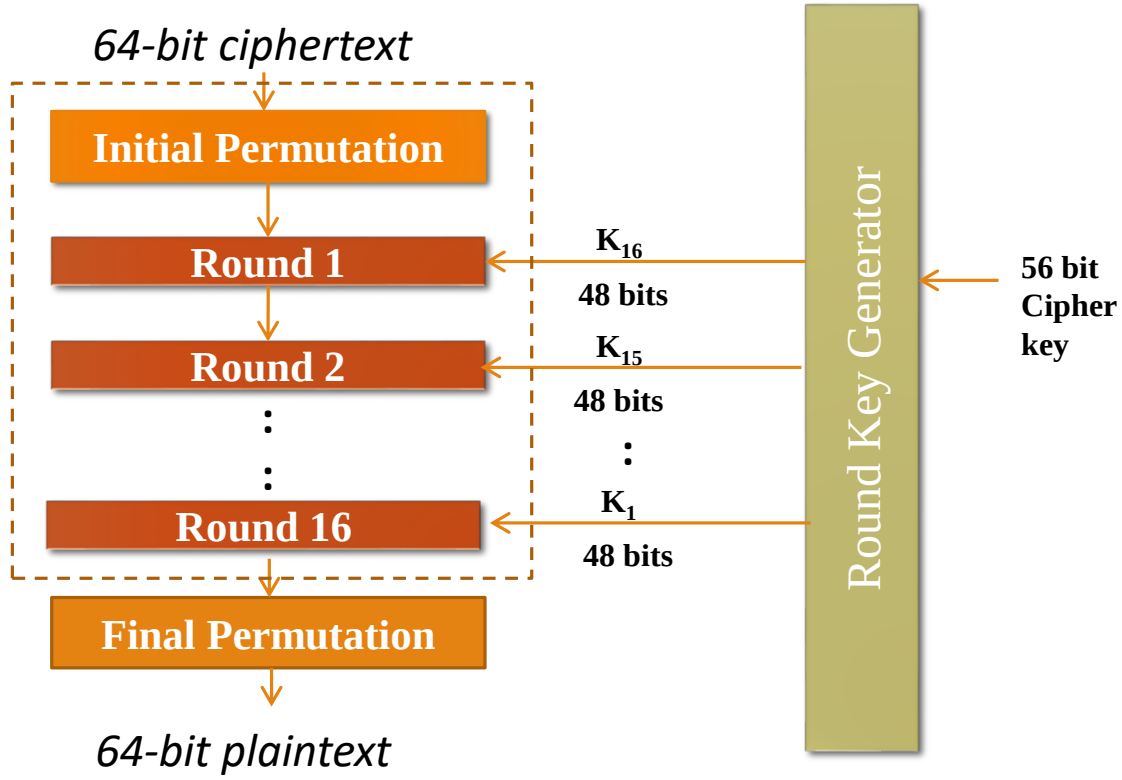
- Algorithm derived from two concepts of **Shannon's theory**.
  - **Confusion :** Piece of information changed, so that the output bits have no relationship to input bits.
  - **Diffusion :** Attempts to spread the effect of one plaintext bit to other bits in cipher text bit.

Data Encryption Standard (DES)

❖ General Structure – Encryption

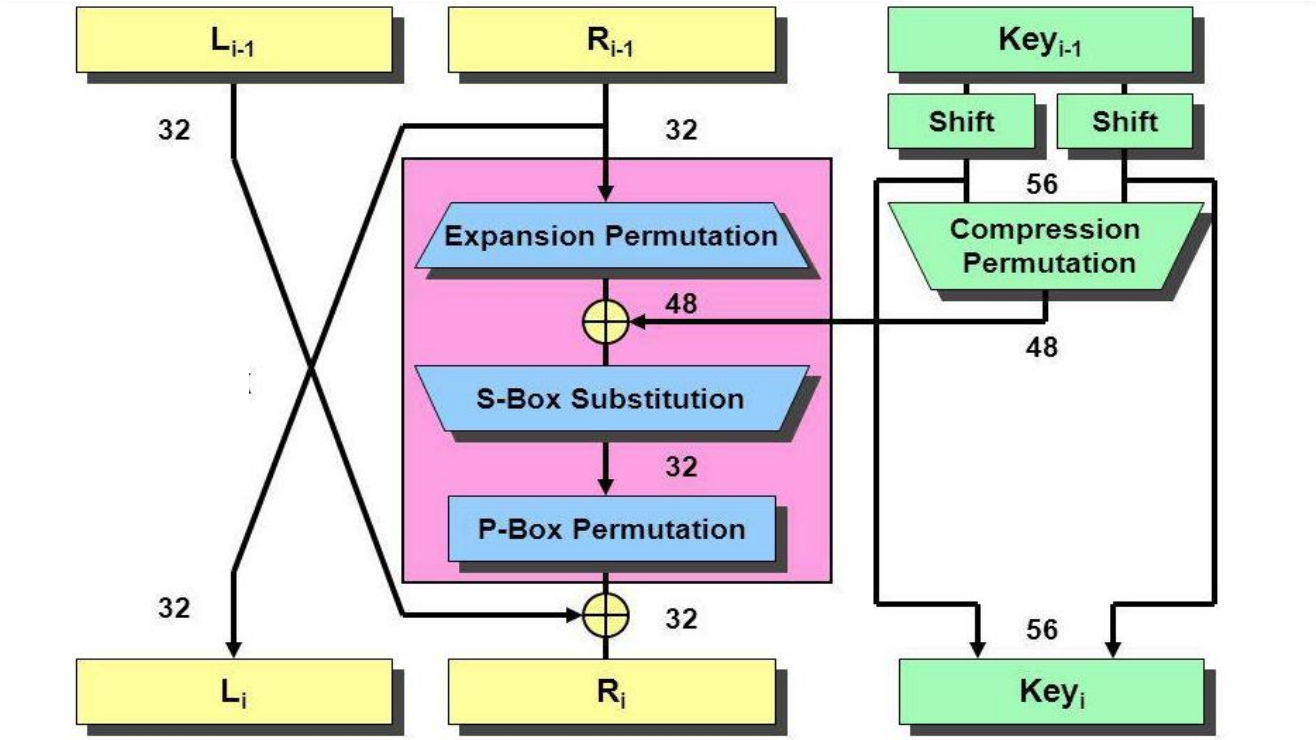


❖ General Structure - Decryption



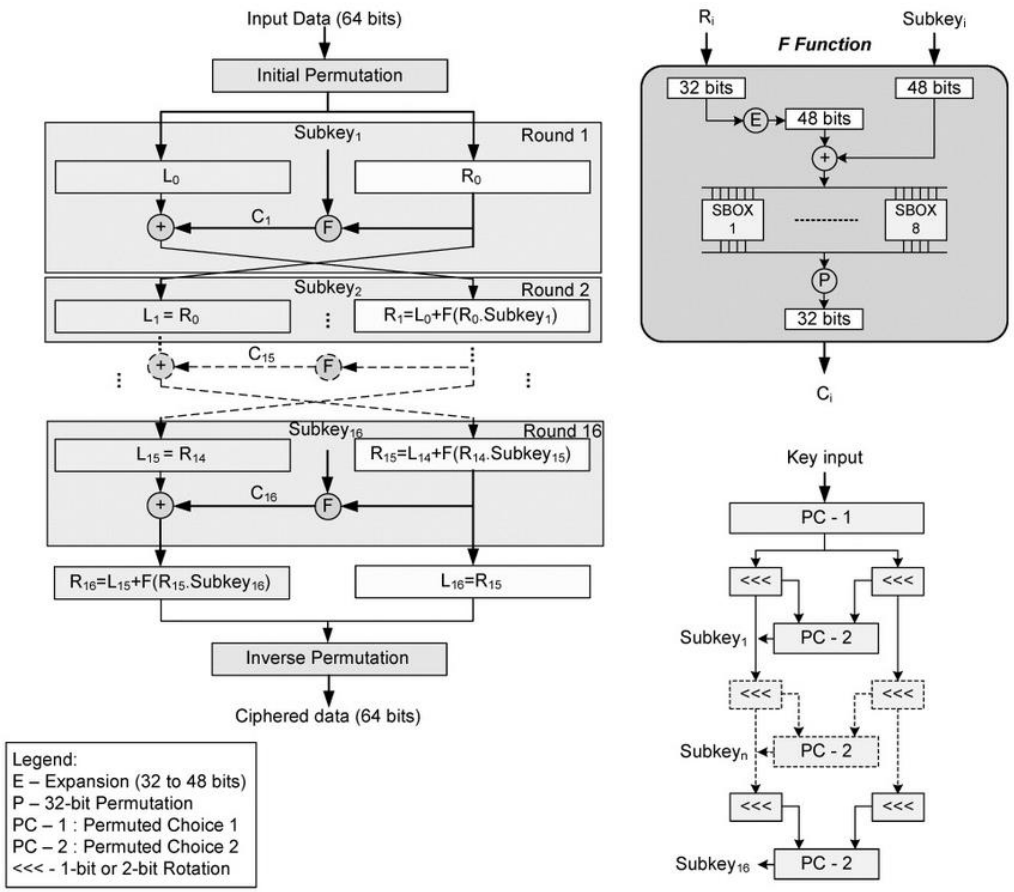
Data Encryption Standard (DES)

- One Round of DES



Data Encryption Standard (DES)

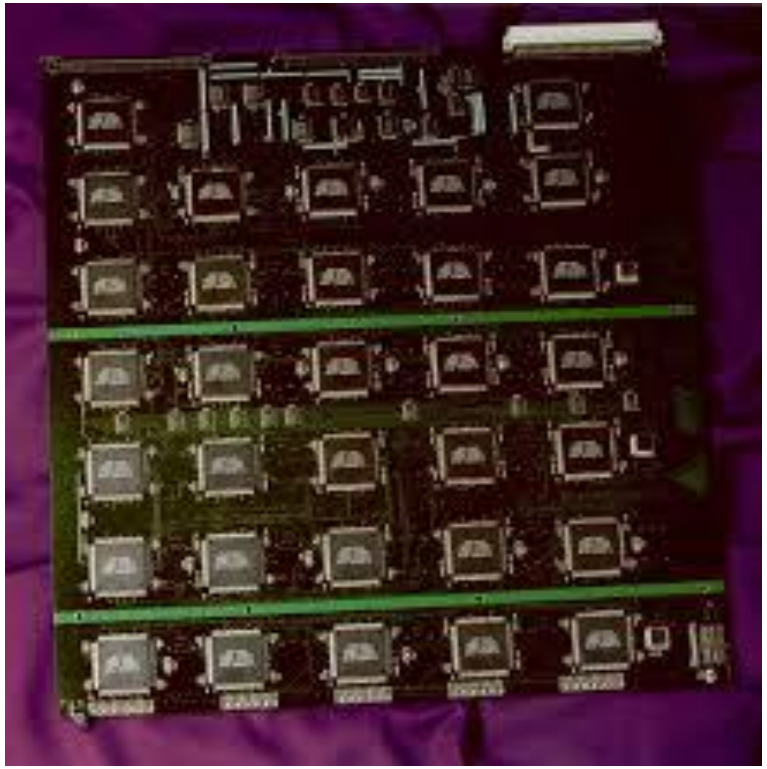
- DES Encryption rounds



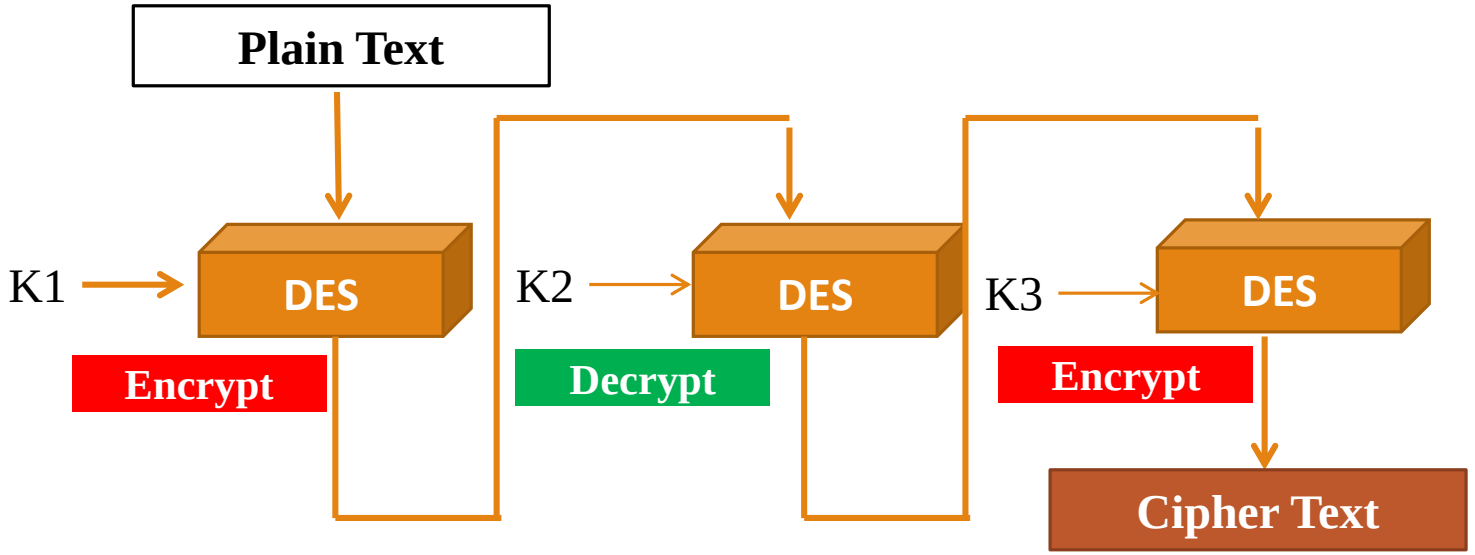
DES - Key Size

- 56 bits keys have $2^{56} = 7.2 \times 10^{16}$ values.
- Brute force search seems hard.
- Recent advances have shown that this is possible.
  1. In 1997 on Internet in a few months.
  2. In 1998 on DES Cracker, dedicated h/w (EFF), in a less than 3 days (cost: $250,000)
  3. In 1999 on Internet in a few hours
  4. in 2010 above on Internet in a few minutes

DES Cracker



Triple DES



$$CipherText = E_{K3}\left(D_{K2}\left(E_{K1}(PlainText)\right)\right)$$

$$PlainText = D_{K1}\left(E_{K2}\left(D_{K3}(CipherText)\right)\right)$$

Triple DES with Two Keys

- Without using 3 different keys, 2 keys can be used E-D-E sequence.

$$C = E_{K_1}\left[D_{K_2}\left[E_{K_1}[P]\right]\right]$$

  - encrypt and decrypt is equivalent in security.
  - If $K_1 = K_2$ then can work with single DES. (Provide backward compatibility with common DES algorithm.)

- Standardized in ANSI X9.17 & ISO8732
- No current reported practical attacks.

DES to AES

- Clearly replacement for DES was needed.
  - Have theoretical attacks that can break it.
  - Have demonstrated exhaustive key search attacks.
- Can use triple DES, but slow with small blocks.
- NIST issued a call for ciphers in 1997
- 15 candidates accepted in June 1998
- 5 were short listed in August 1999
- **Rijndael** was selected as the AES in October 2000
- Issued as FIPS PUB 197 standard in November 2001

Advanced Encryption Standard (AES)

- In 2001, National Institute of Standards and Technology (NIST) issued AES known as FIPS.
- AES is based on **Rijndael** proposed by Joan Daeman, Vincent Rijmen from Belgium.

Advanced Encryption Standard (AES)

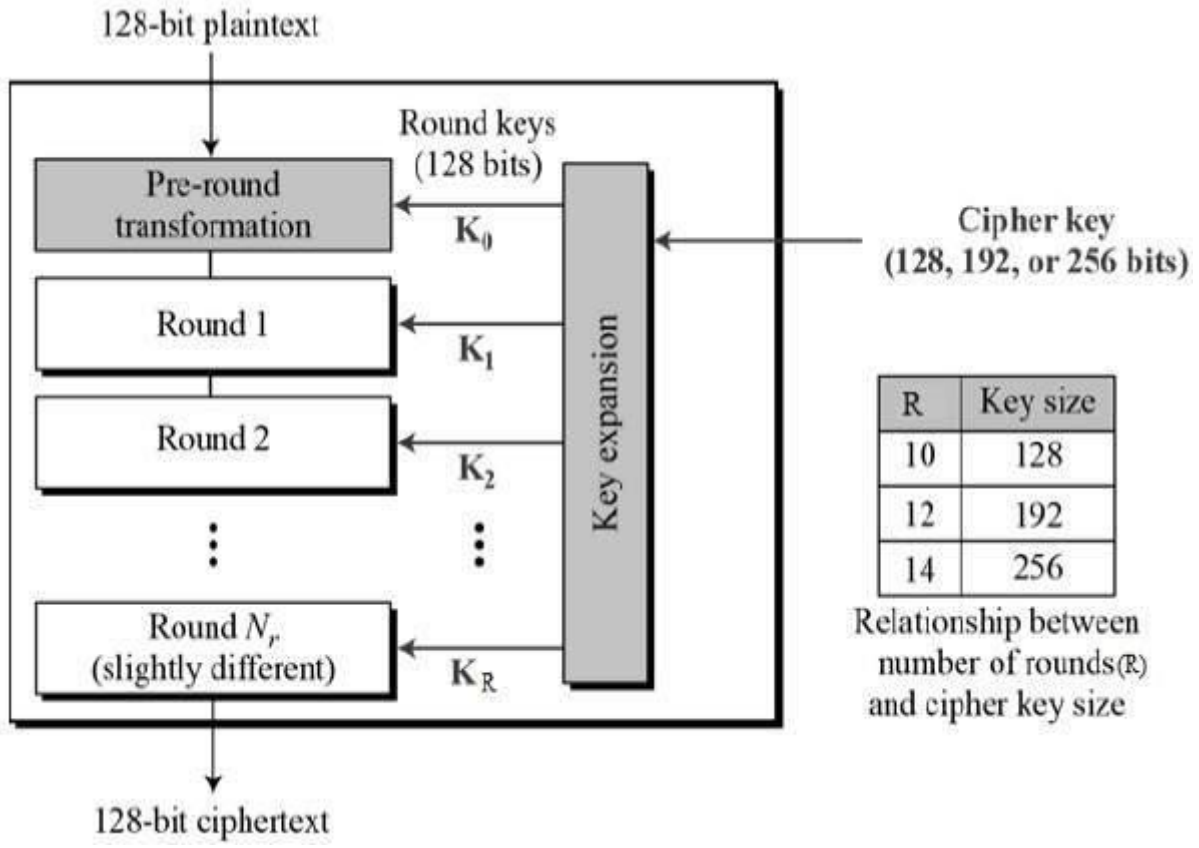
- Block Length **128bits**.
- Supported key lengths are 128, 192 and 256.
- Unlike DES, number of rounds in AES depends on the length of the key.

Key Size	Number of Rounds
128	10
192	12
256	14

- Each of these rounds uses a different **128-bit** round key, which is calculated from the original AES key.

Advanced Encryption Standard (AES)

- **Encryption**

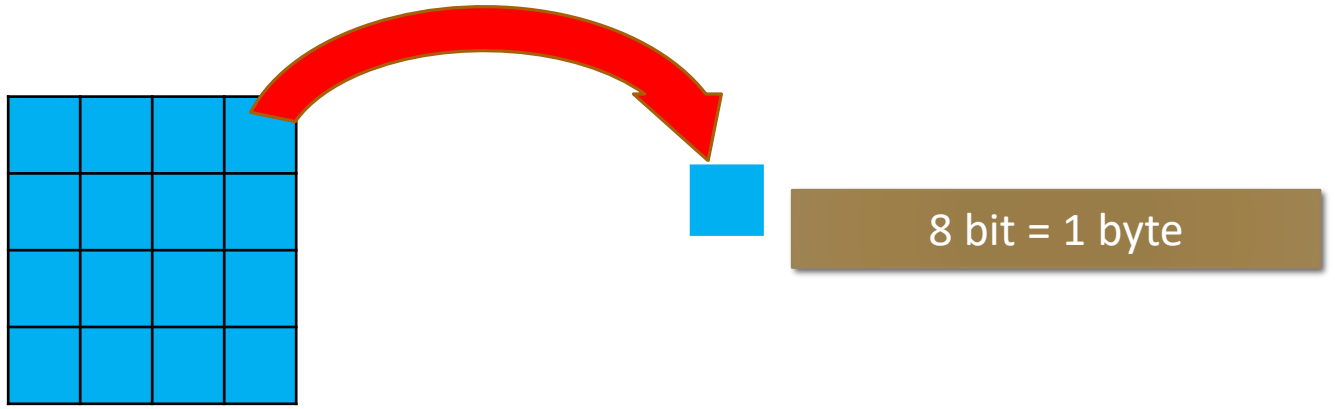


R	Key size
10	128
12	192
14	256

Relationship between number of rounds (R) and cipher key size

Advanced Encryption Standard (AES)

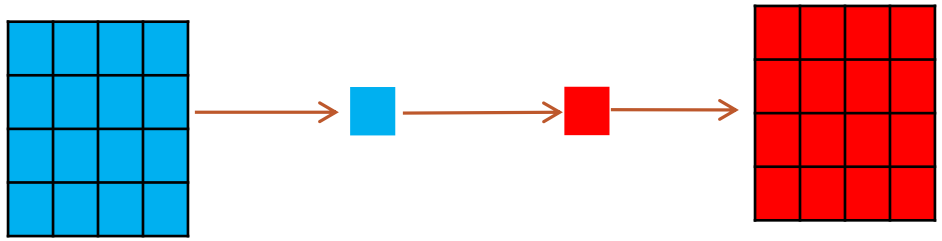
- **Encryption – Typical Round**



1. 4×4 array of 128 bit plain text is the input to each round

Advanced Encryption Standard (AES)

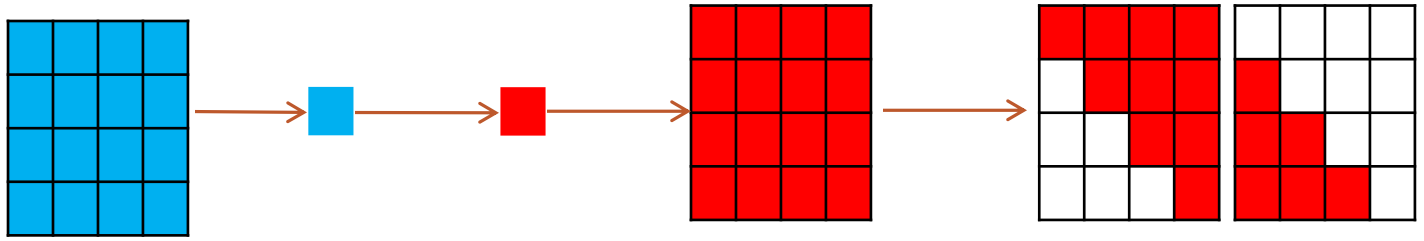
- **Encryption – Typical Round**



2. Each byte of the array is mapped into a new byte

Advanced Encryption Standard (AES)

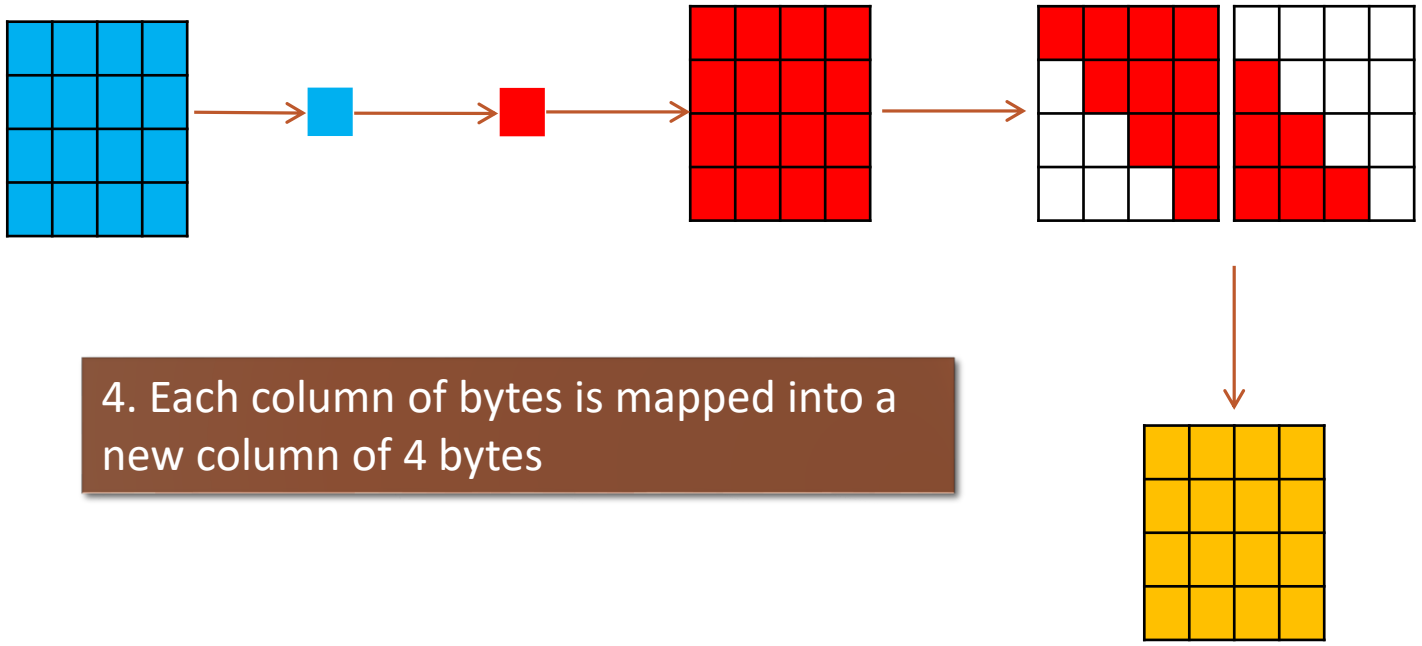
Encryption- Typical Round



3. The second row shifts one byte
   The third row shifts two bytes.
   The fourth row shifts three bytes

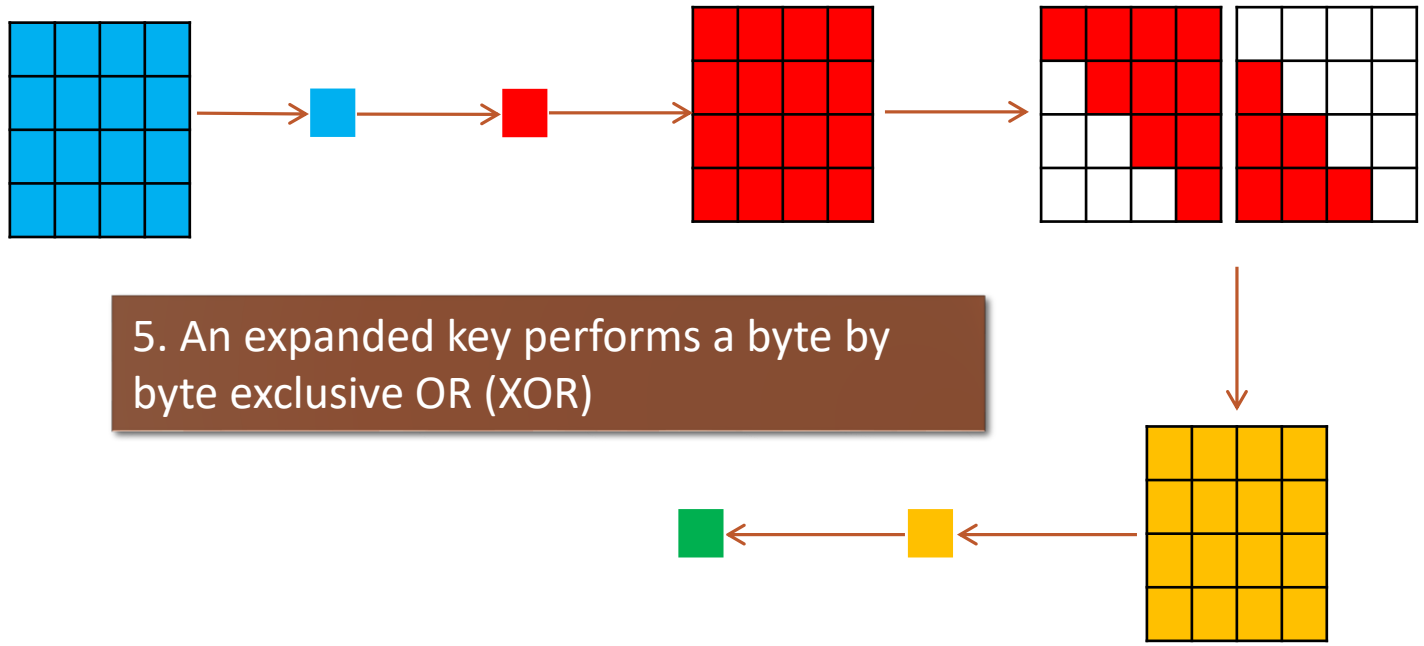
Advanced Encryption Standard (AES)

- **Encryption – Typical Round**



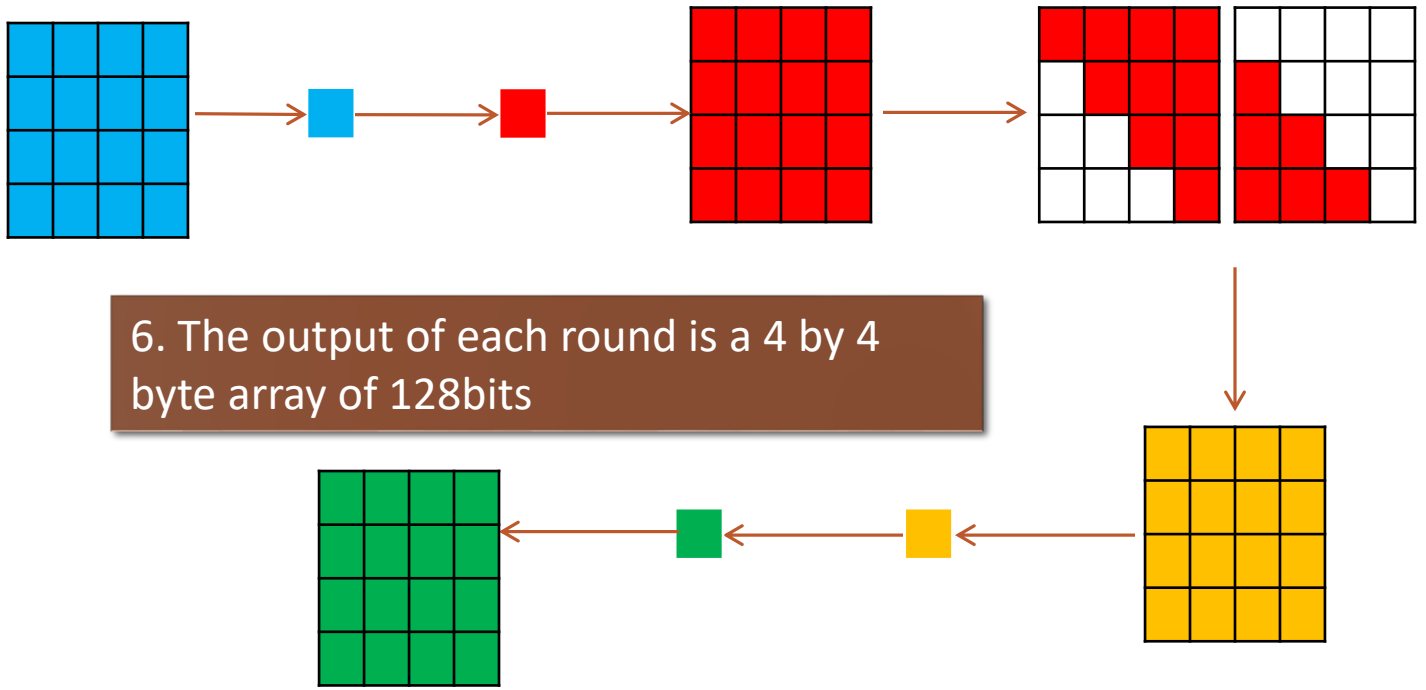
Advanced Encryption Standard (AES)

- **Encryption – Typical Round**



Advanced Encryption Standard (AES)

- **Encryption – Typical Round**



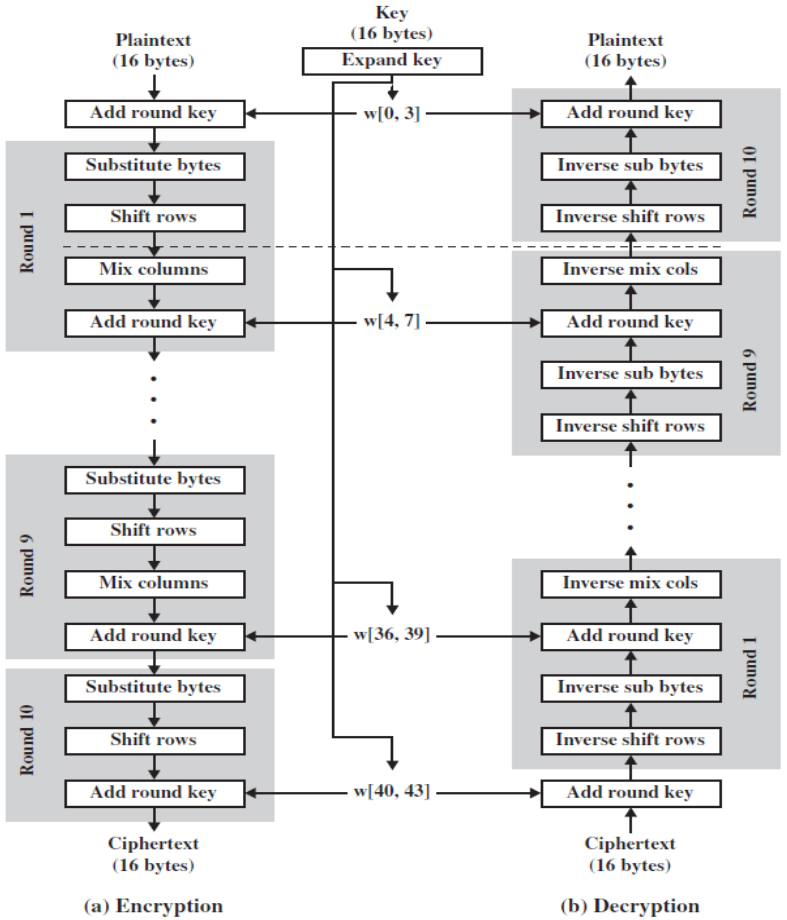
Advanced Encryption Standard (AES)

- **Encryption – Typical Round**

Advanced Encryption Standard (AES)

- **Decryption**
  - Decryption algorithm is not identical to encryption algorithm.
  - Steps in a round in the encryption process are performed in reverse order.

Advanced Encryption Standard (AES)



Block Ciphers – Modes of Operation

- Block Ciphers encrypt fixed size blocks.
- When it is needed to encrypt arbitrary amount of information
  - Four modes were defined for DES in ANSI standard.
  - ANSI X3.106-1983 Modes of Use
  - Subsequently now have 5 for DES and AES
- Basically the long message is divided into a series of sequential message blocks, and the cipher algorithm operates on these blocks.

Electronic Code Book (ECB)

- Message is broken into independent blocks which are encrypted.
- Each block is a value which is substituted, like a codebook
- Each block is encoded independently of other blocks.

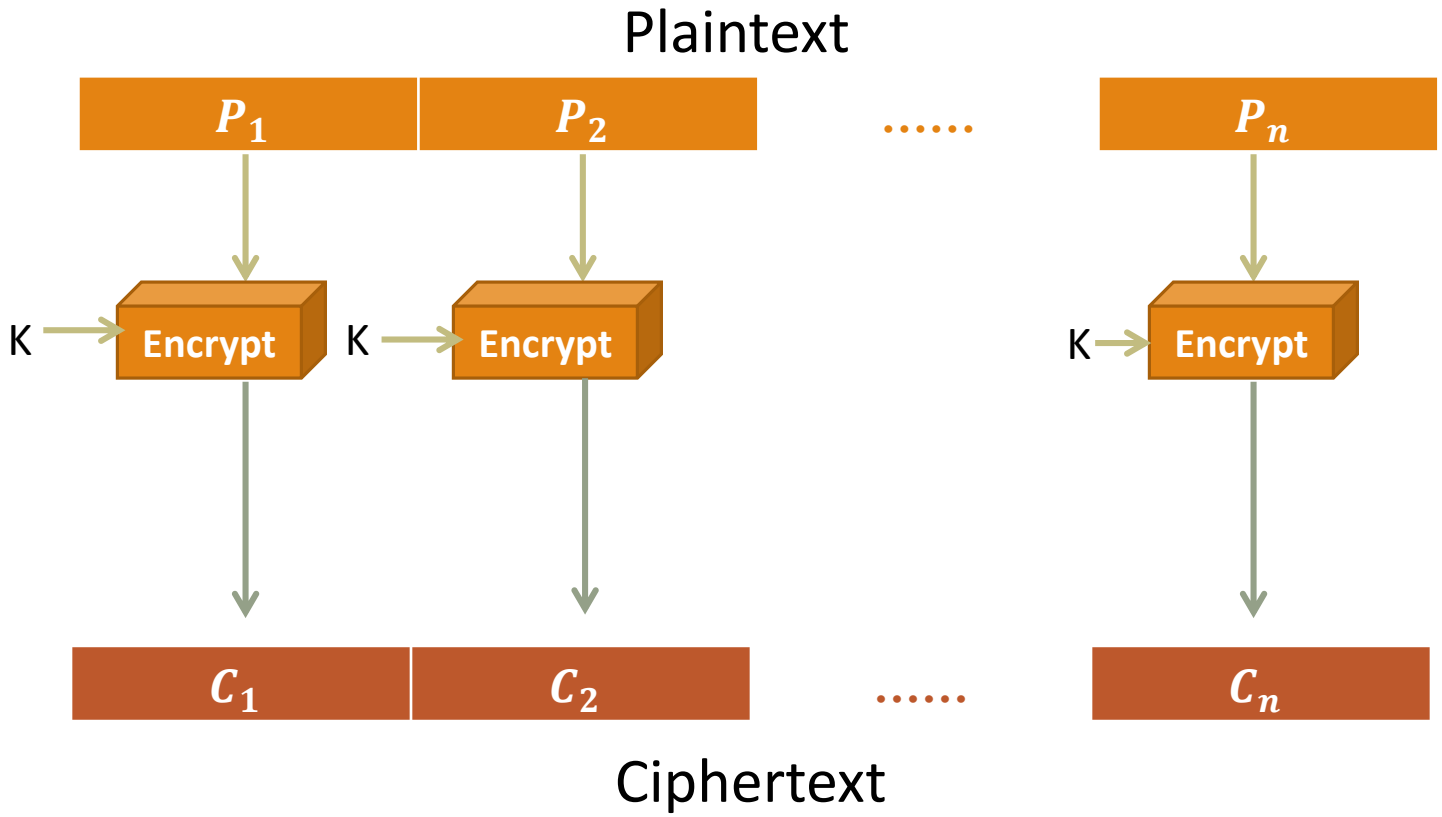
$$C_i = E(P_i, K) \text{ (Encryption)}$$

$$P_i = D(C_i, K) \text{ (Decryption)}$$

- Uses: Secure transmission of single values.

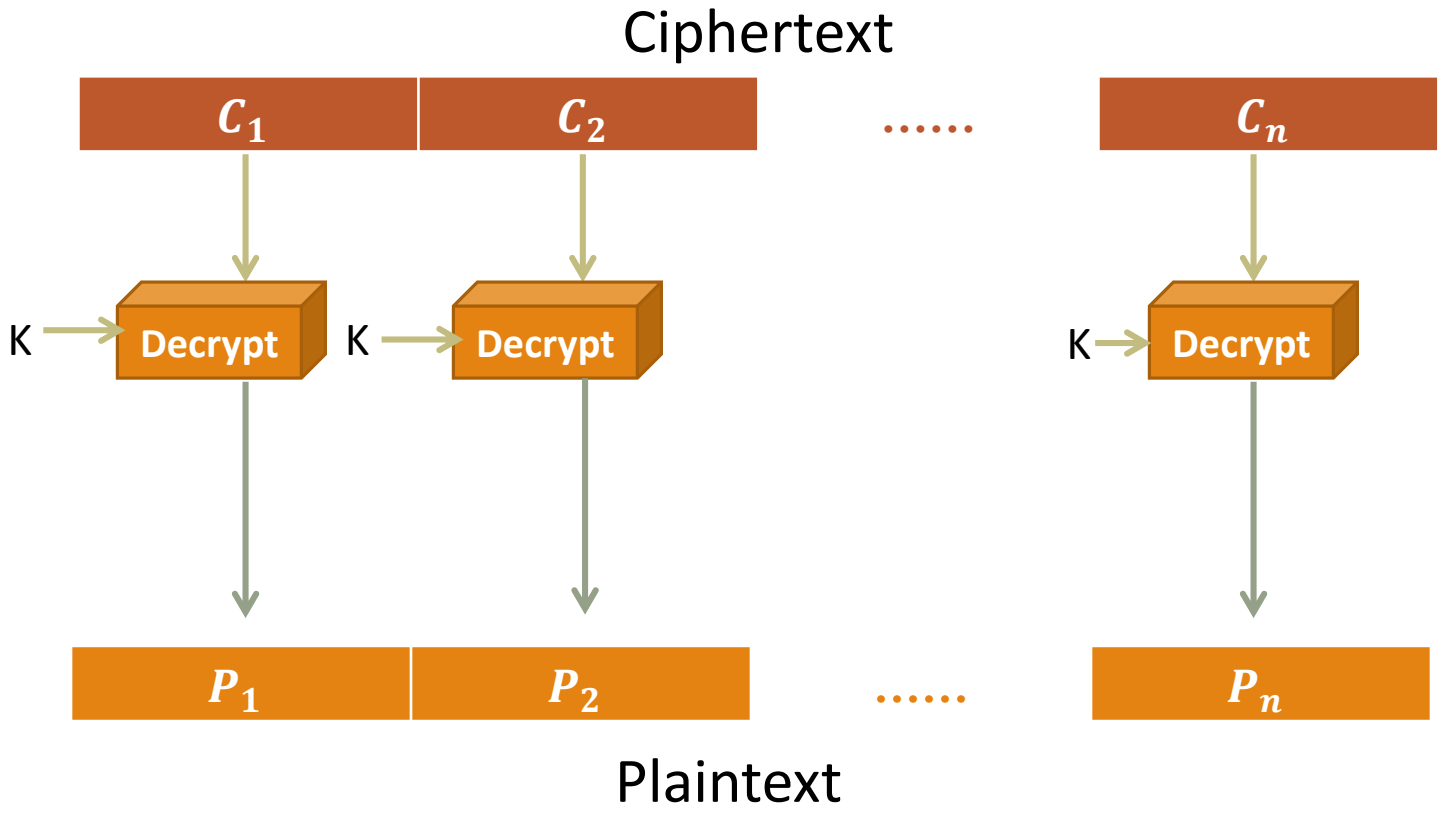
Electronic Code Book (ECB)

- Encryption



Electronic Code Book (ECB)

- Decryption



Advantages and Limitations in ECB

- Message repetitions may show in ciphertext
  - if aligned with message block
  - particularly with data such as graphics.
  - or with messages that change very little.
- Weakness is due to the encrypted message blocks being independent.
- Unable to hide data patterns
- Main use is sending a few blocks of data.

Cipher Block Chaining (CBC)

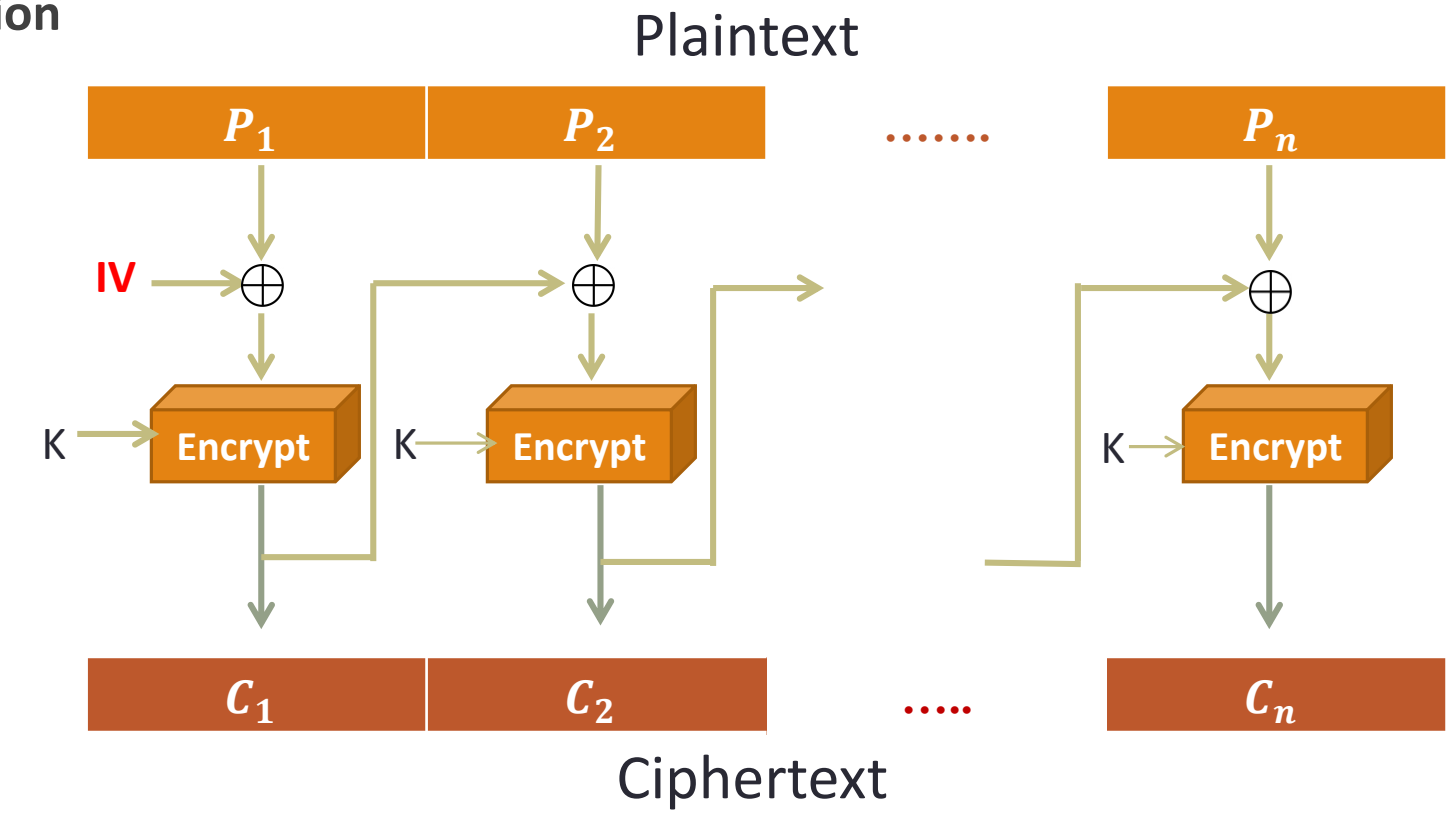
- The Message is divided into blocks.
- Again linked together in encryption operation.
- Each previous ciphertext block is chained with current plaintext block.
- Initial vector (**IV**) is used to start the process.

Encryption	**Decryption**
$C_1 = E([P_1 \oplus IV], K)$ $C_i = E([P_i \oplus C_{i-1}], K)$	$P_1 = D(C_1, K) \oplus IV$ $P_i = D(C_i, K) \oplus C_{i-1}$

- Usages
  i. Bulk data encryption
  ii. Authentication

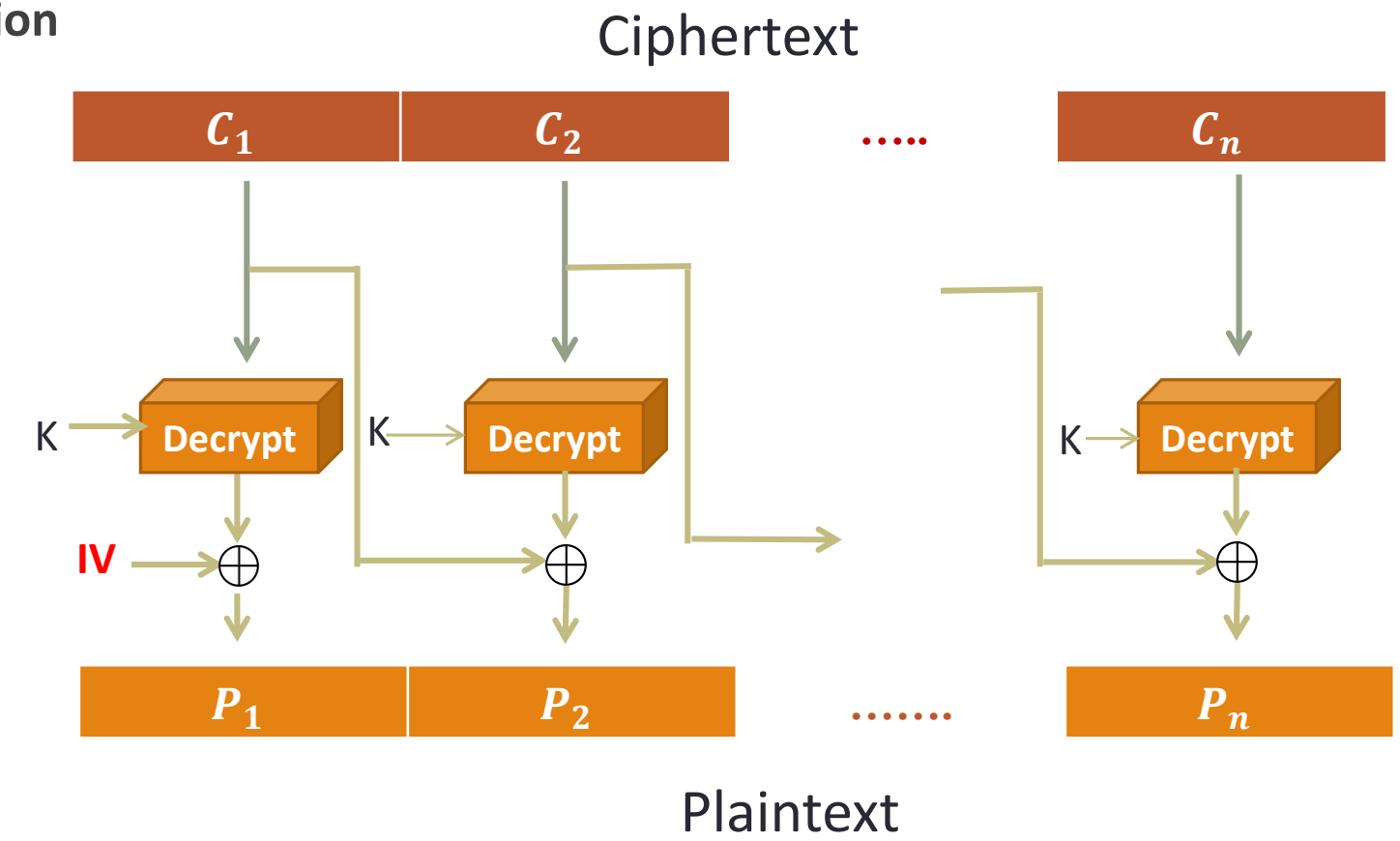
Cipher Block Chaining (CBC)

- Encryption

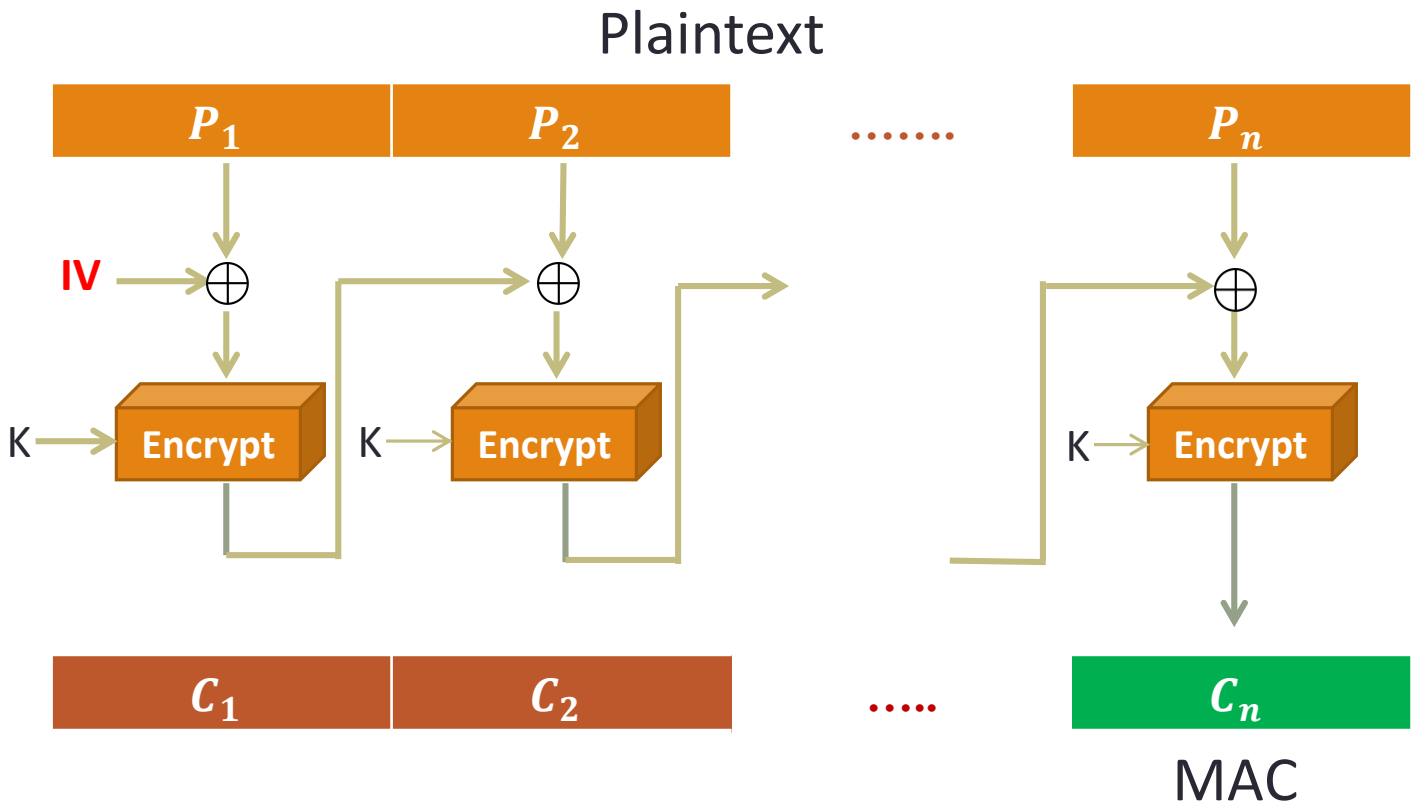


Cipher Block Chaining (CBC)

- Decryption



MAC based on CBC



Advantages and Limitations in (CBC)

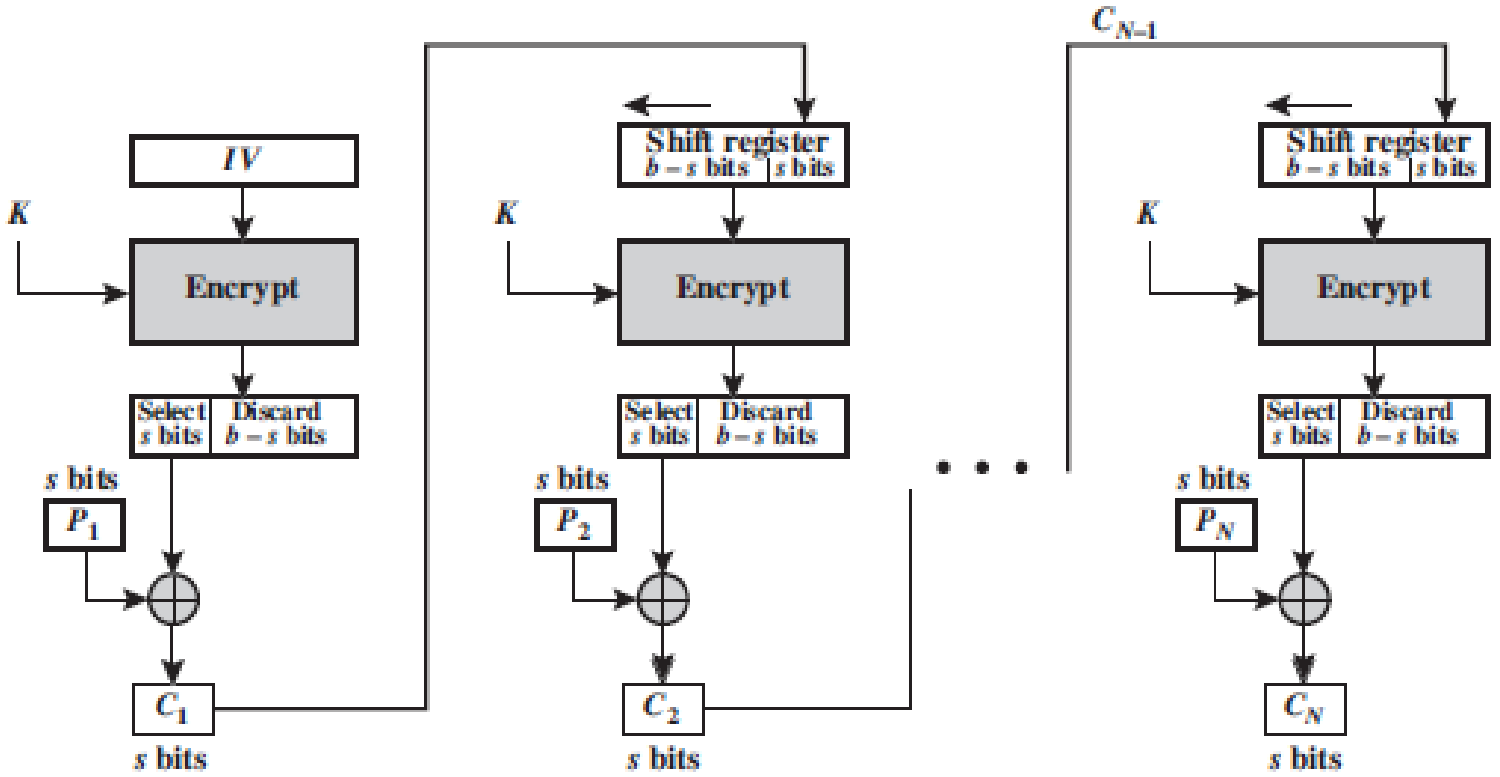
- A cipher text block depends on all preceding message blocks.
- Any change to a block affects all following cipher text blocks.
- Need Initial Vector (IV):
  1. Which must be known to sender and receiver.
  2. If this is sent in plaintext, the attacker can change the bits of first block, and change the IV to compensate.
  3. Hence IV must either be a fixed value
  4. Or must be sent encrypted in ECB mode before rest of message.

Cipher Feed Back (CFB) Mode

- A **Stream Cipher** that uses ciphertext as **feedback** into the Key Generation Source to develop the **next key stream**.
- The Ciphertext is generated by performing an XOR on the Plaintext with the Key stream having same number of bits as the Plaintext.
- Error will propagate in this mode.

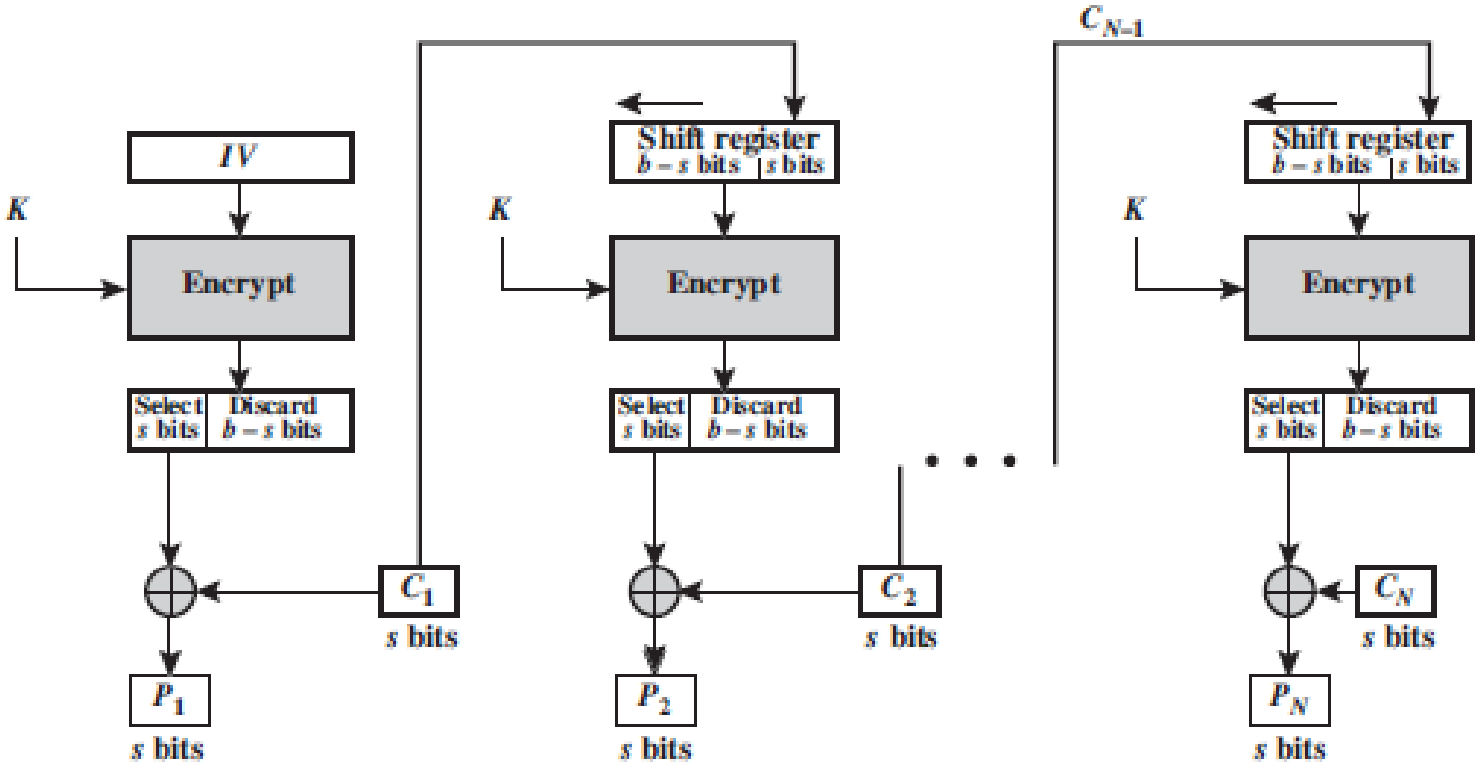
Cipher Feed Back (CFB) Mode

- Encryption



Cipher Feed Back (CFB) Mode

- Decryption

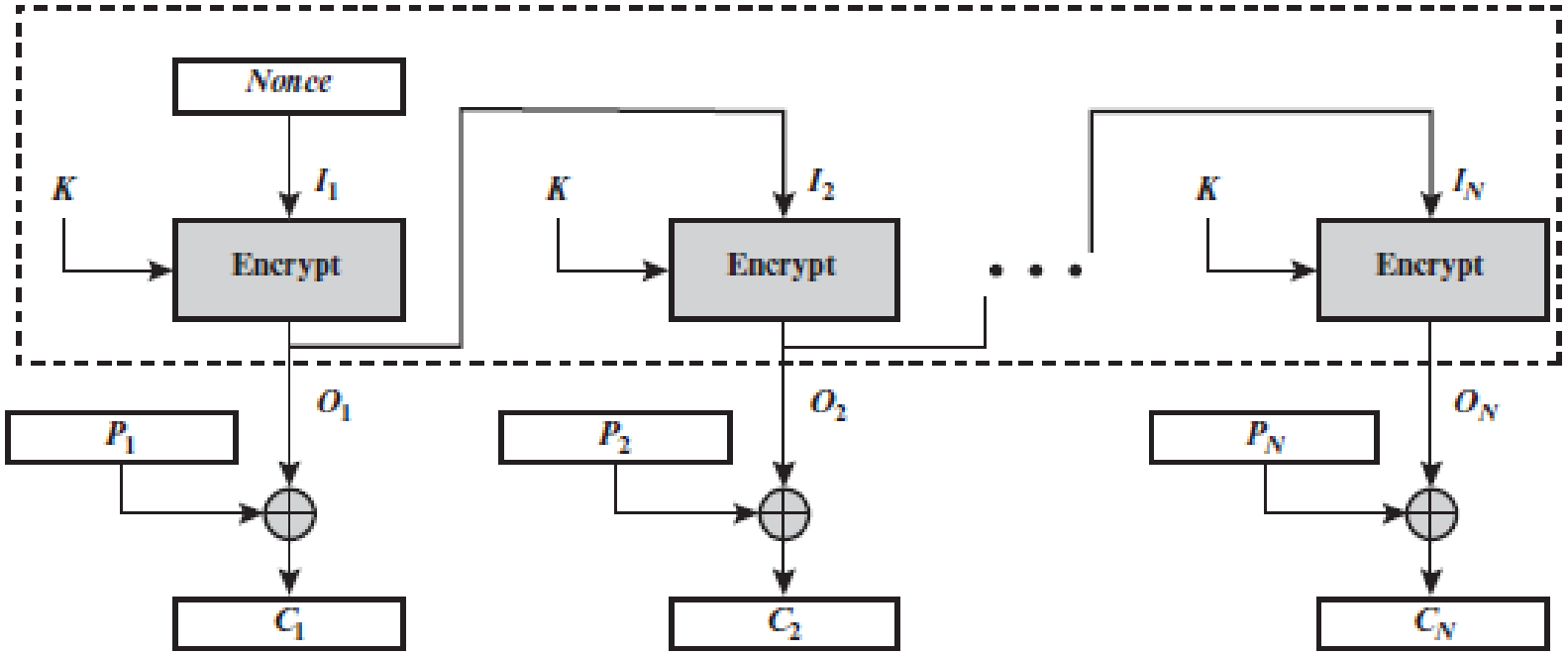


Output Feed Back (OFB) Mode

- A stream cipher that generates the ciphertext key by performing XOR operation on the Plaintext with a Key stream.
  - Operates on full blocks of plaintext and ciphertext
- Requires an initialization vector.
  - Must be unique to each execution of the encryption operation (nonce).
- Feedback is used to generate the Key Stream
  - Therefore key stream will vary.
- Errors will not propagate in this mode.

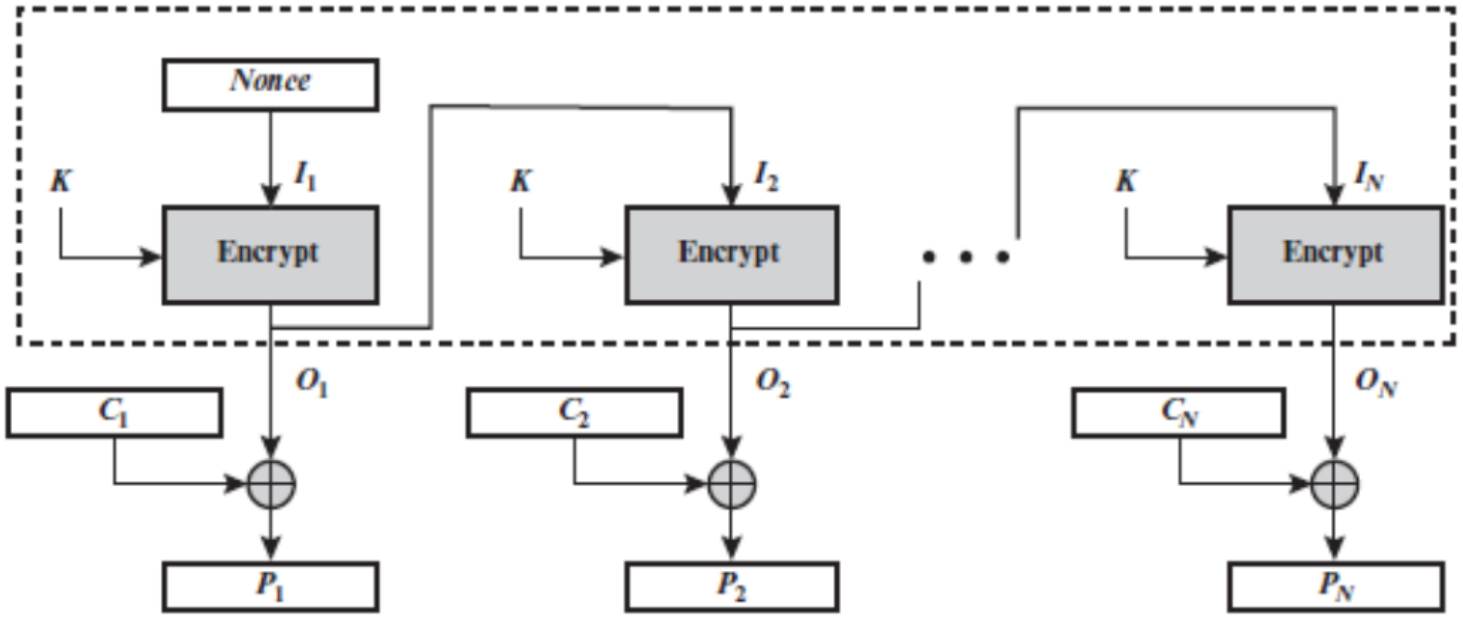
Output Feed Back (OFB) Mode

- Encryption



Output Feed Back (OFB) Mode

- Decryption

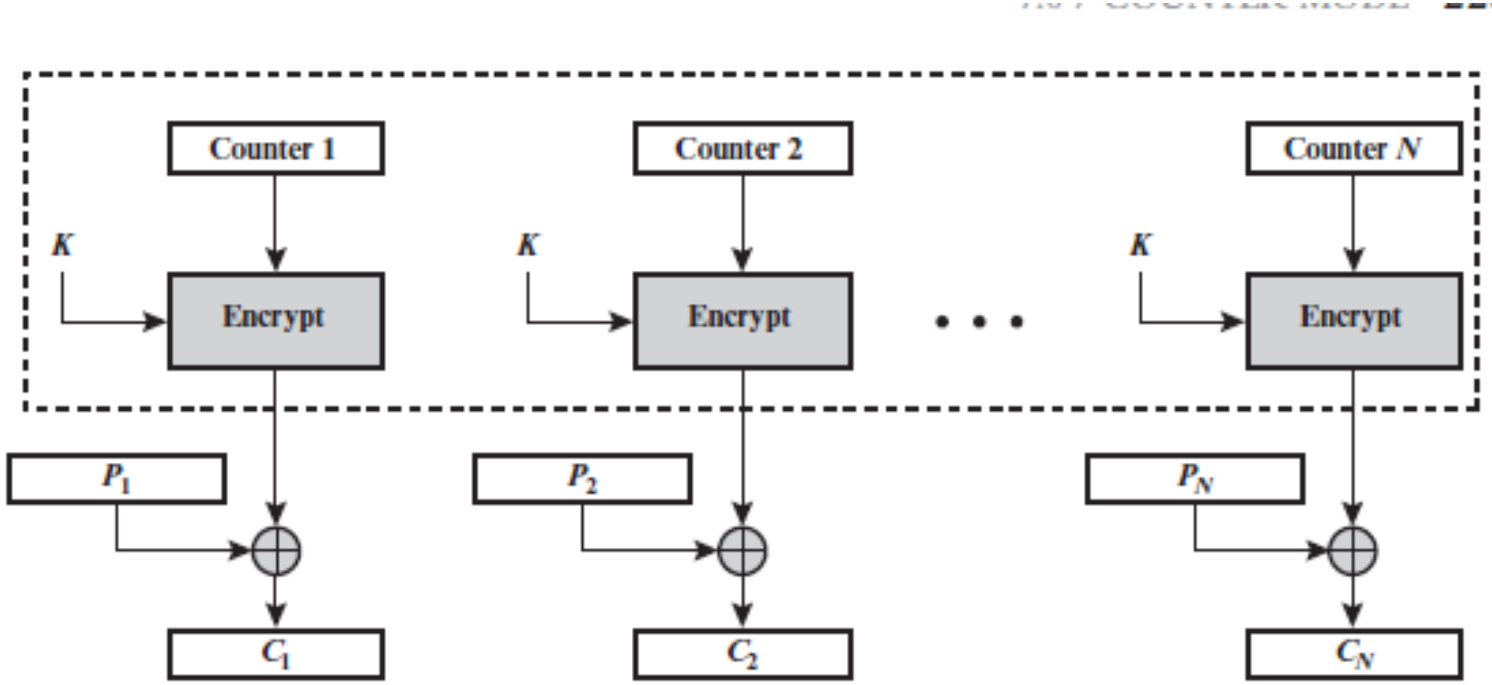


Counter Mode (CTR)

- **Counter:** Equal to the plaintext block size
- Ciphertext is generated by performing XOR with the plaintext block and counter value
- The counter must have a different value for each plaintext block.
  - Initialized to some value and incremented by one
- Uses: High speed network encryptions

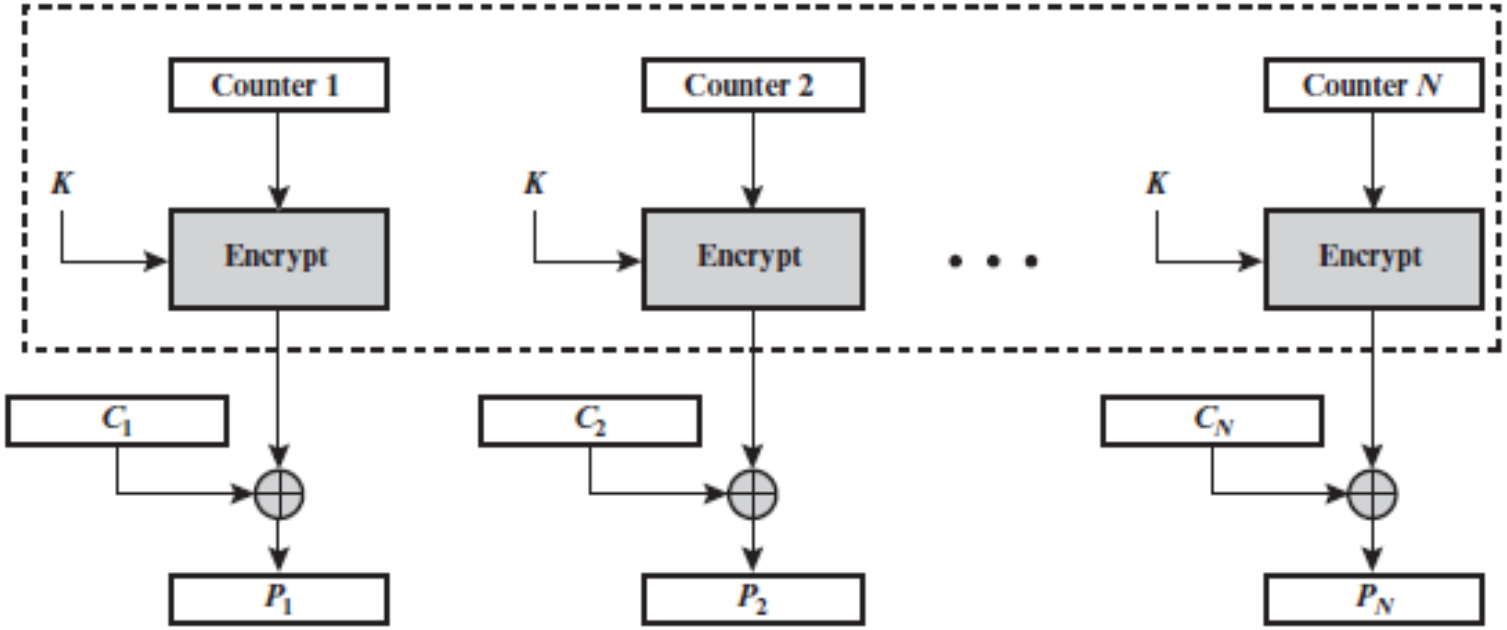
Counter Mode (CTR)

- Encryption



Counter Mode (CTR)

- Decryption



Counter Mode (CTR)

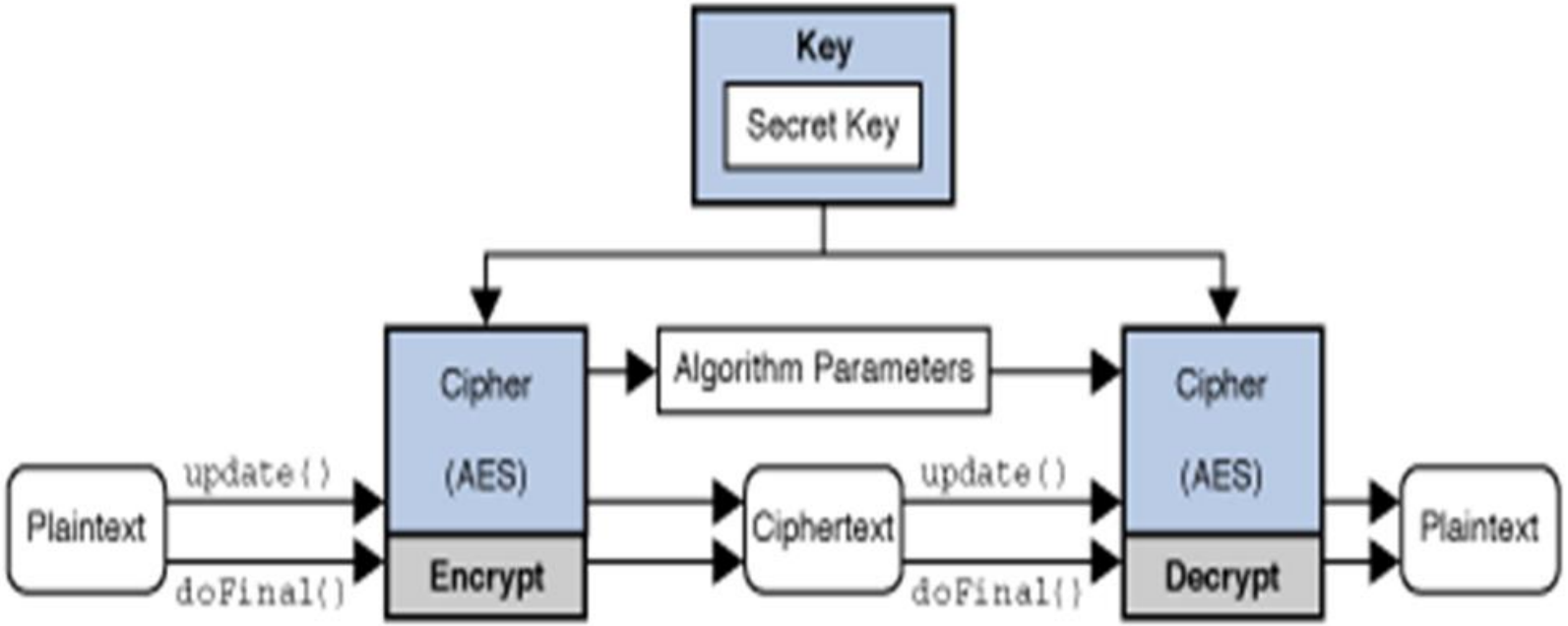
❖ **Advantages**

1. Able to perform parallel encryption in H/W or S/W
2. Can be pre-processed in advance
3. Good for high speed links
4. Random access to data blocks
5. Provable security

❖ **Limitation**

➢ Must ensure never reuse key/counter values; otherwise, it could be broken

AES-ECB Process



AES-ECB Encryption

1. **Key generation:**

```
KeyGenerator generator = KeyGenerator.getInstance("AES");
generator.init(128);
Key key = generator.generateKey();
```

2. **Obtain Cipher Engine:**

```
Cipher c =Cipher.getInstance(“AES/ECB/PKCS5Padding”)
```

3. **Initializing the cipher engine for encryption:**

```
c.int(Cipher.ENCRYPT_MODE, key)
```

4. **Do the padding and finish the encryption:**

```
byte[] cipherText = c.doFinal(input)
```

AES-ECB Decryption

1. **Use the same key used in encryption**
2. **Obtain Cipher Engine:**

```
Cipher c =Cipher.getInstance("AES/ECB/PKCS5Padding")
```

3. **Initializing the cipher engine for encryption:**

```
c.int(Cipher.DECRYPT_MODE, key)
```

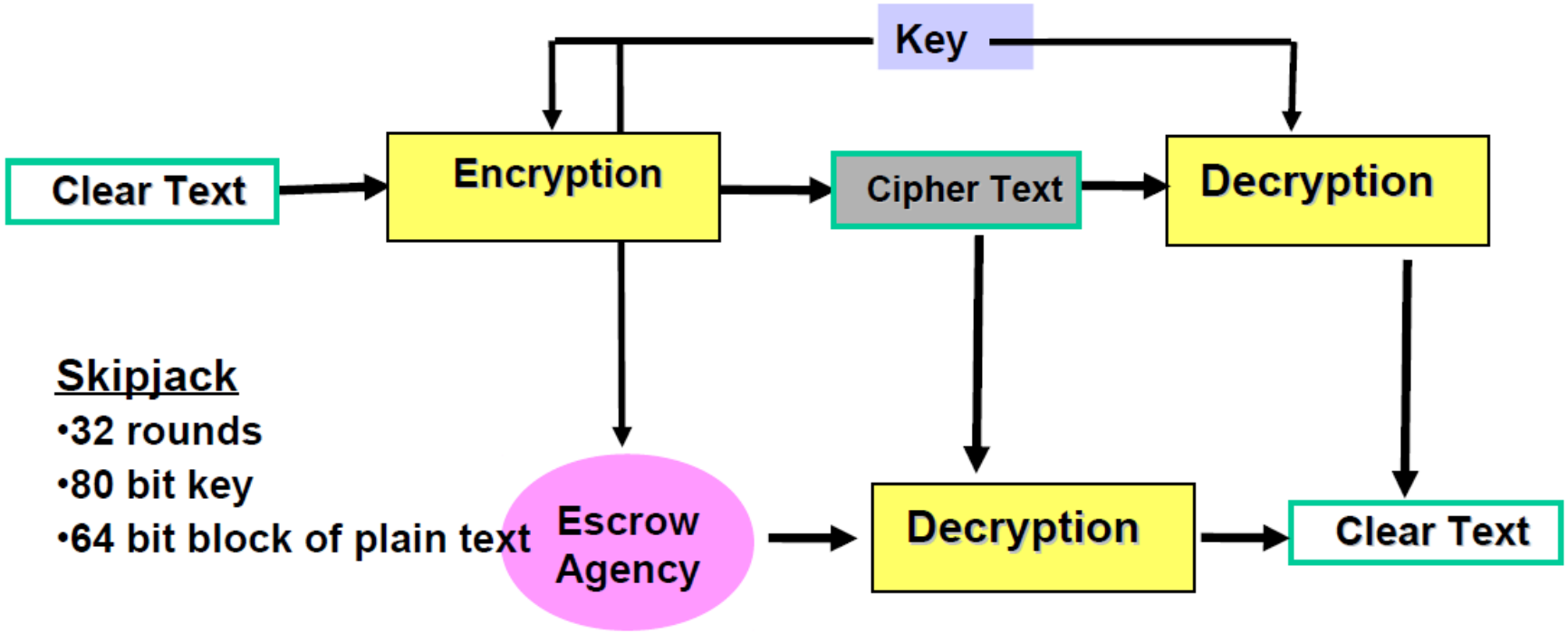
4. **Do the padding and finish the encryption:**

```
byte[] cipherText = c.doFinal(input);
```

Key Escrow

- Separate agencies maintain components of private key, which, when combined, can be used to decrypt ciphertext.
- Stated reason is to decrypt drug related communications.
- Clipper Chip is an example
  - Secret algorithm
  - Unpopular, unused.
- Issues include key storage.

Key Escrow Standard



Other Symmetric Block Cipher

1. **International Data Encryption Algorithm (IDEA)**
   - 128 bit key
   - Used in PGP
2. **Blowfish**
   - Easy to implement
   - High execution speed
   - Run in less than 5K of memory.

Other Symmetric Block Cipher

3. **RC5**
   - Suitable for hardware and software
   - Fast, simple
   - Adaptable to processors of different word lengths
   - Variable number of rounds.
   - Variable length key
   - Low memory requirement
   - High security.
   - Data – dependent rotation.

4. **Cast – 128**
   - Key size from 40 to 128 bits.
   - The round function differs from round to round.

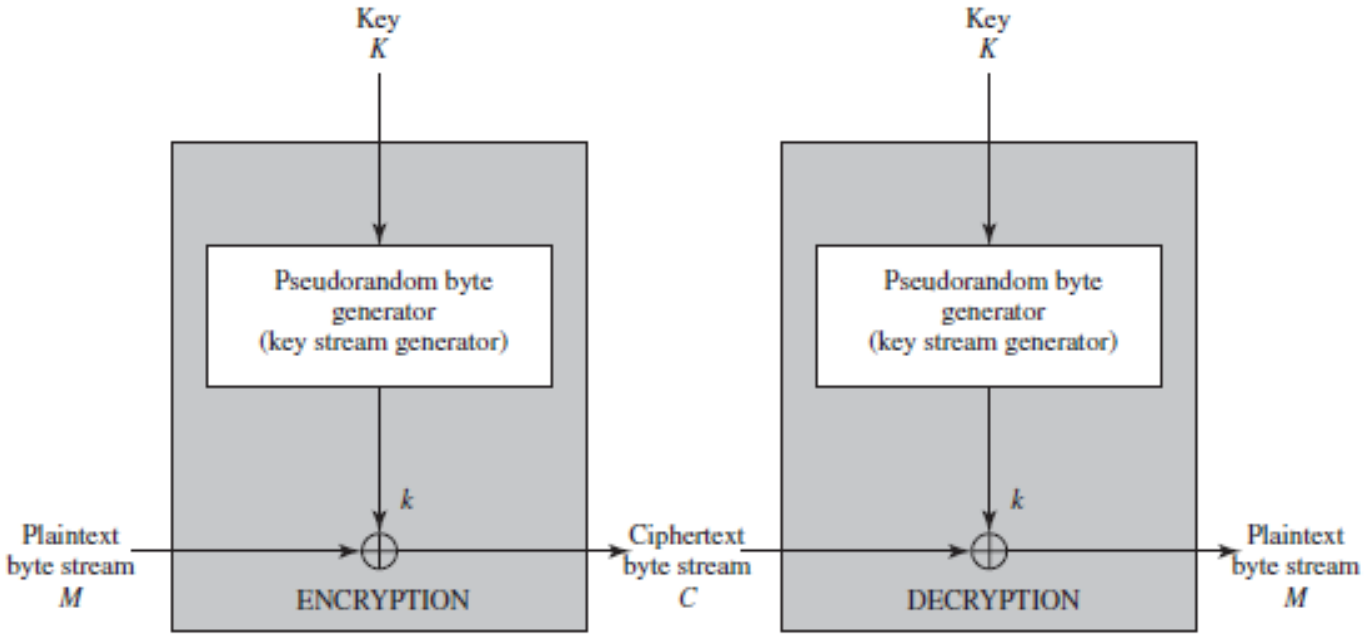
Stream Ciphers

- Process the message bit by bit.
- Typically have a (pseudo) random stream key.
  - Given a key, a pseudo random bit generator generates a keystream.
- Combined (XOR) with plaintext bit by bit.
- Randomness of stream key completely destroys any statistically properties in the message

$$C_i = M_i \oplus StreamKey_i$$

- Must avoid reuse stream key.
  - Otherwise can remove effect and recover messages.

Stream Ciphers



Stream Cipher Properties

- **Design Considerations:**
  1. Long period no repetitions
  2. Statistically random
  3. Depends on large enough key
  4. Large linear complexity
  5. Correlation immunity
  6. Confusion
  7. Diffusion
  8. Use of high non-linear Boolean functions

RC4

- A proprietary cipher owned by RSA DSI.
- Another Ron Rivest design, simple but effective.
- Variable key size, byte oriented stream cipher.
- Widely used (web SSL/TLS, wireless WEP)
- Key forms random permutation of all 8-bit values.
- Uses that permutations to scramble input information processed a byte at a time.

RC4 Security

- Claimed that secure against known attacks
  - Have some analysis, none practical
- Result is very non-linear
- Avoid reuse of a key, as this is a stream cipher.

Advantages and Disadvantages of Symmetric Ciphers

Advantages

1. Algorithms are very fast.
2. The same key is used to Encryption and Decryption
3. As long as the key remains secret, the system also provides authentication

Disadvantages

1. When the key is revealed the interceptors can encrypt and decrypt all information.
2. Problem of key distribution
3. Number of keys increases with the square of the number of people exchanging the information.