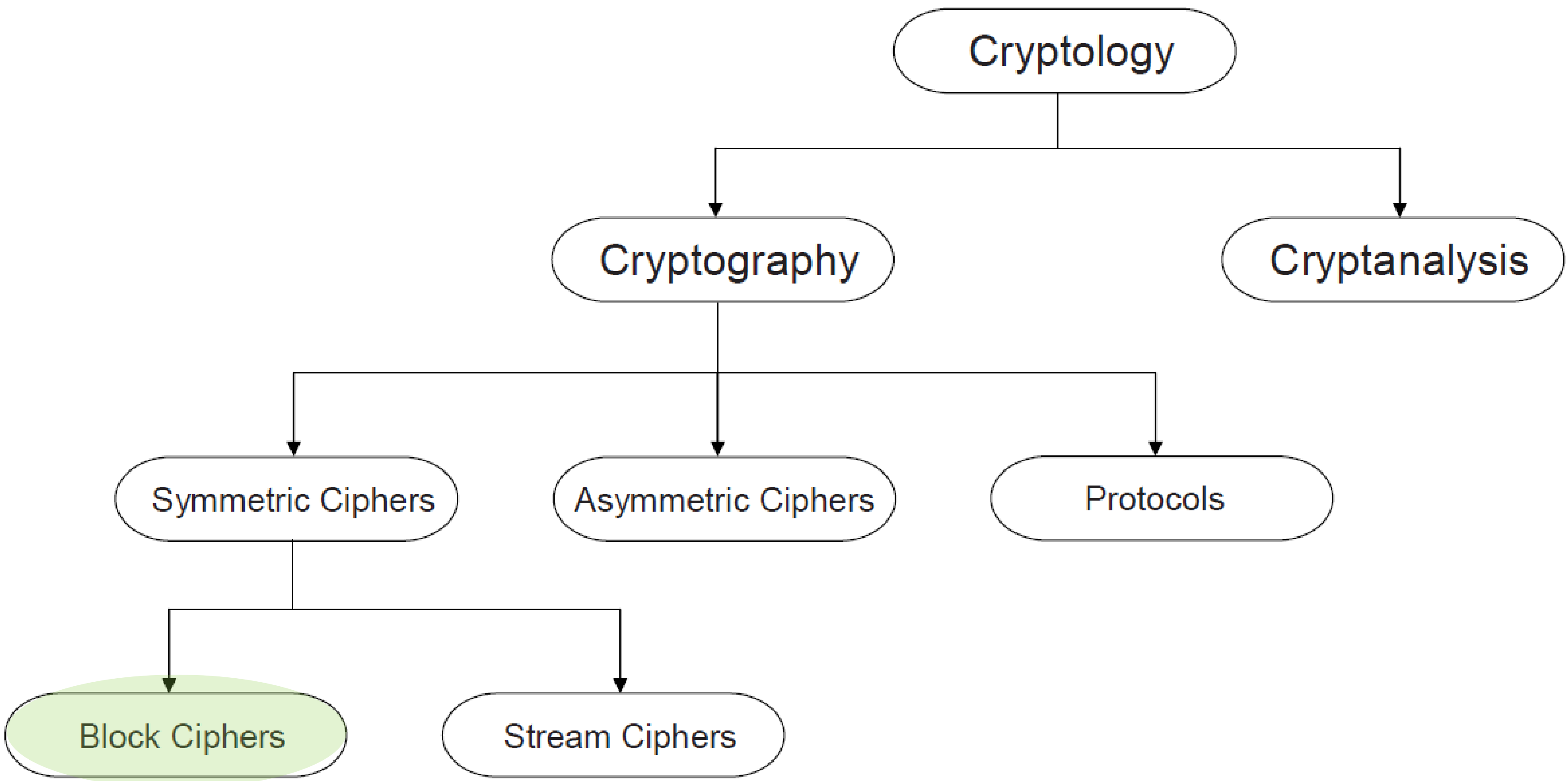
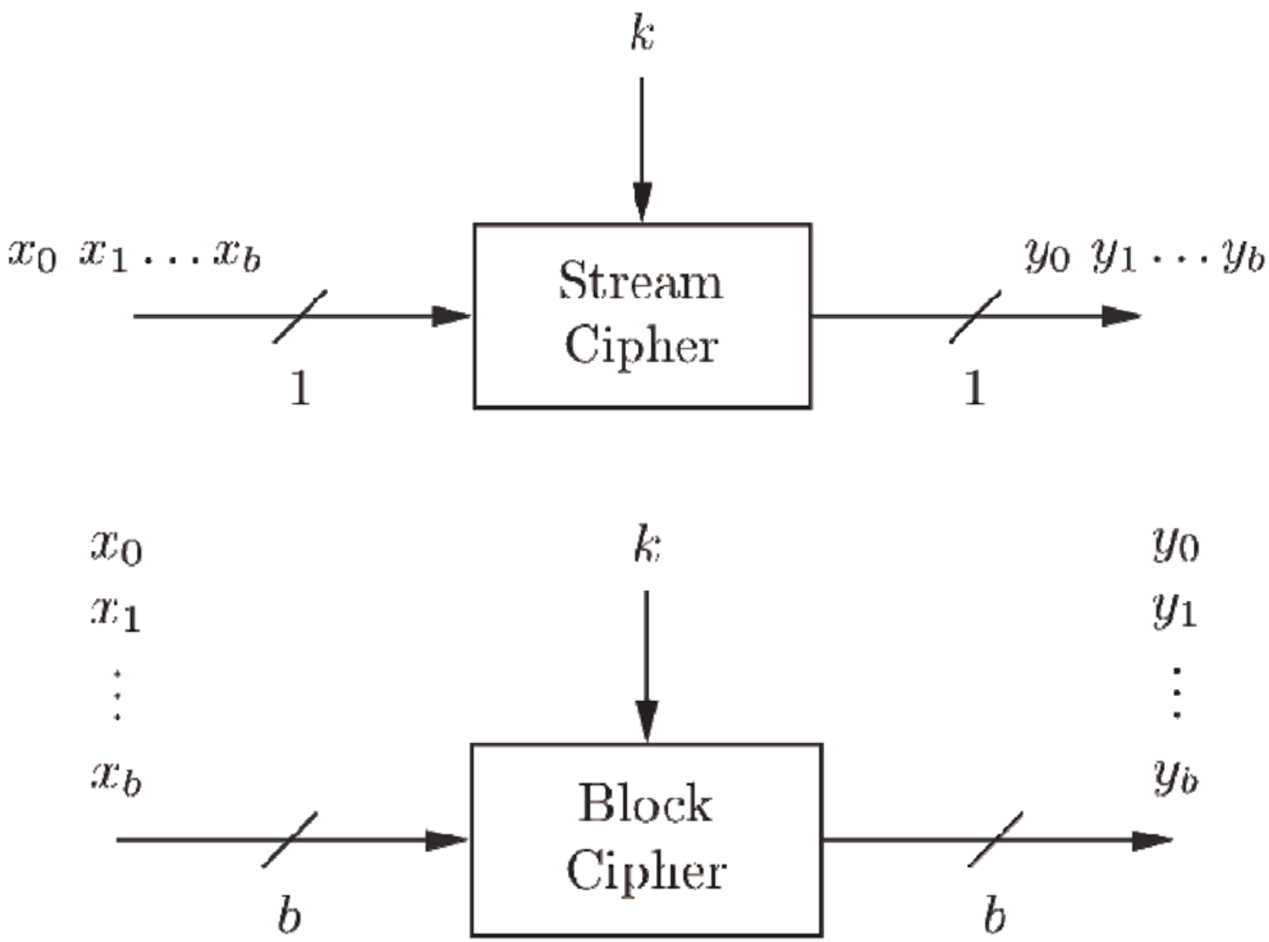
# Block Ciphers

# Taxonomy of Cryptology



# Concept: Stream vs Block Cipher



# Stream Cipher vs Block Cipher

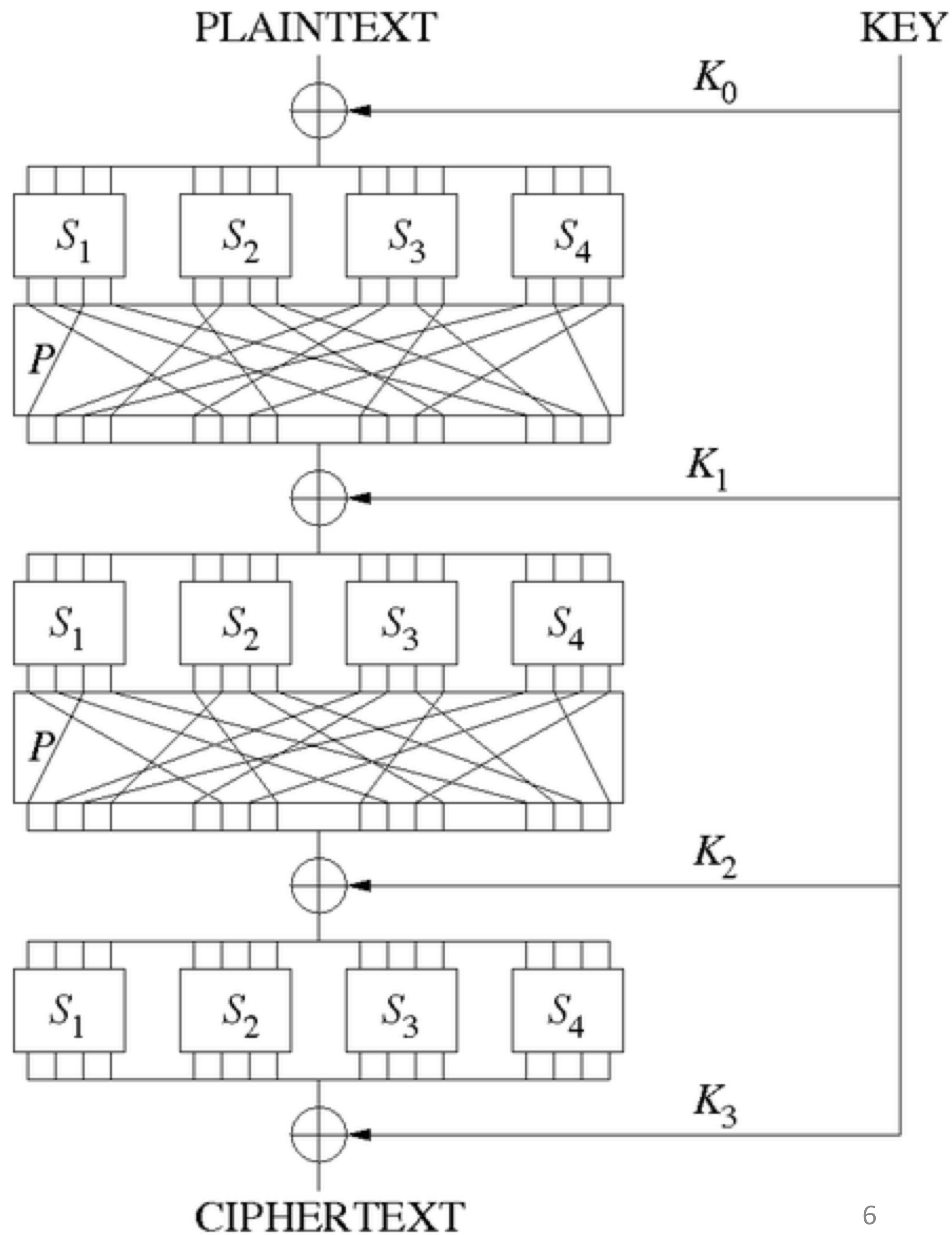
- Stream ciphers
  - Encrypt bits individually
  - Usually small and inline with I/O and transmission
  - Examples: OTP, A5/1 (GSM), RC4 (WEP, TLS, PPTP)
  - Advantages
    - Encryption/Decryption errors do not propagate
    - Small and Fast
  - Disadvantages
    - Low diffusion
    - Susceptible to bit flips, insertions, modification

# Stream Cipher vs Block Cipher

- Block ciphers
  - Encrypt bits as a block of bits (64, 128)
  - Generally slower, larger programs
  - Examples: DES, 3DES, AES
  - Advantages
    - High diffusion
    - Robust to bit flips, insertions, modification
  - Disadvantages
    - Encryption/Decryption errors do propagate
    - Generally Slower

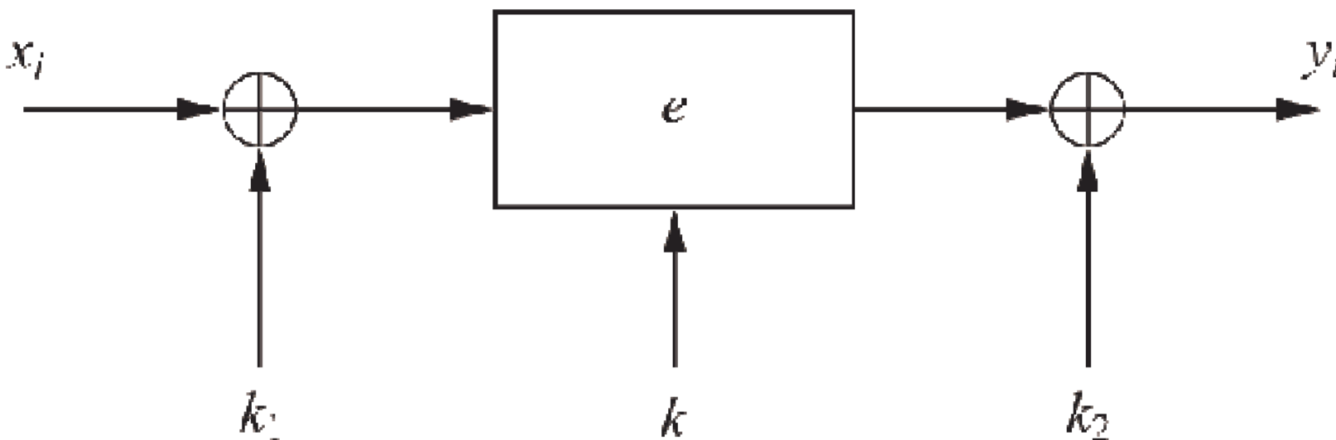
# Block Ciphers

- Almost all are product ciphers
  - Feistel
  - Generalized Feistel
  - Sub-Perm Network (shown)
- Key schedule
  - 1 key per round
  - Sometimes key whitening step (shown)



# Key Whitening

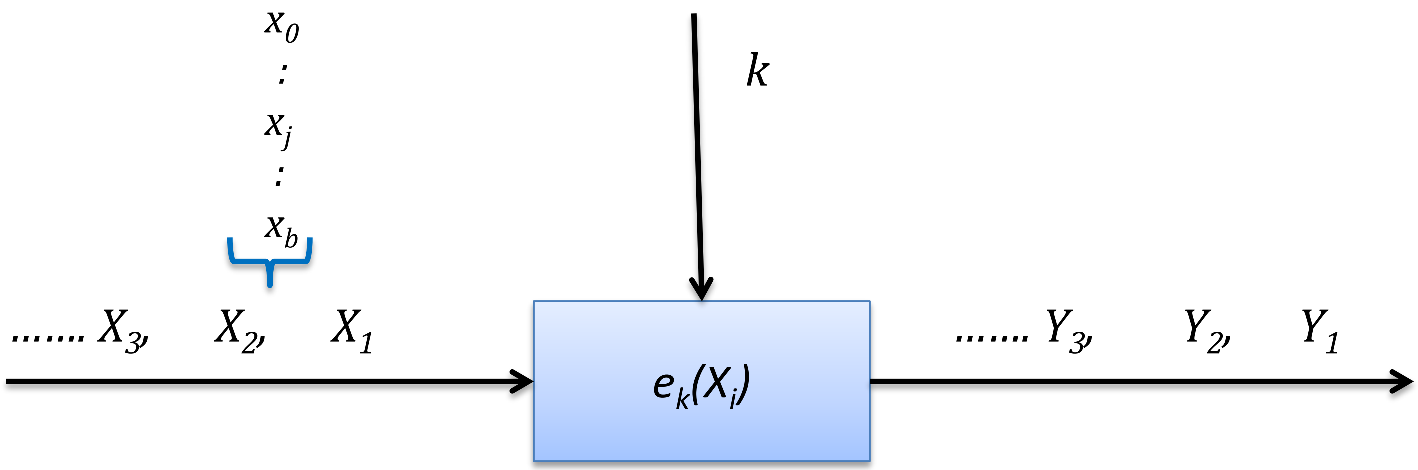
- Most modern ciphers offer a key whitening step




- Pure key addition to hide internal encryption
- Can also strengthen ciphers with short key space

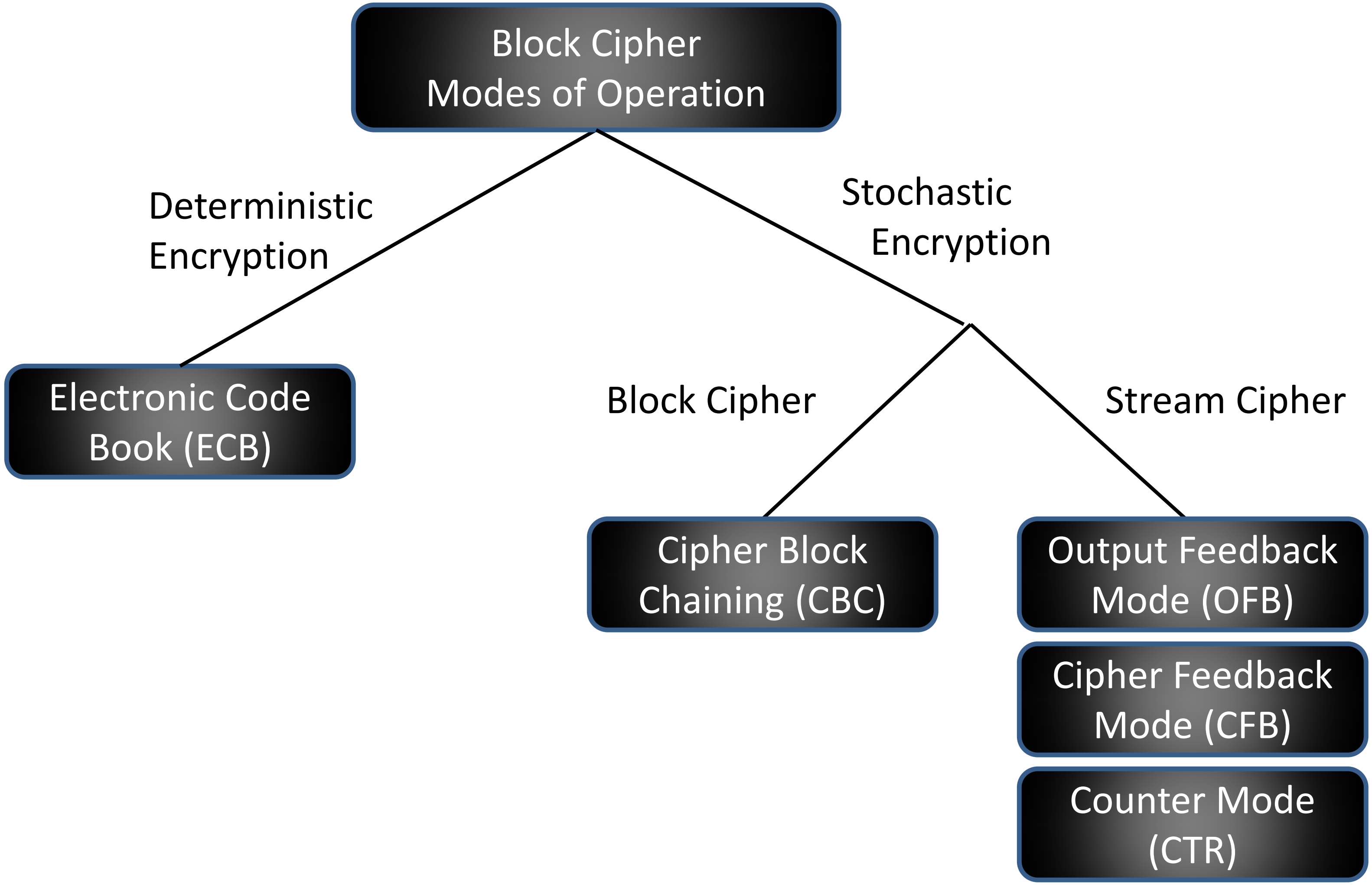
# Block Ciphers

- Let $X_i$ be blocks of data, strings of $x_j$ of length $b$
- $X_i = \{x_0 \ldots x_j \ldots x_b\}$ where $b$ is a block length
- Then $Y_i = \{y_0 \ldots y_j \ldots y_b\}$ is the output of $e_k(X_i)$



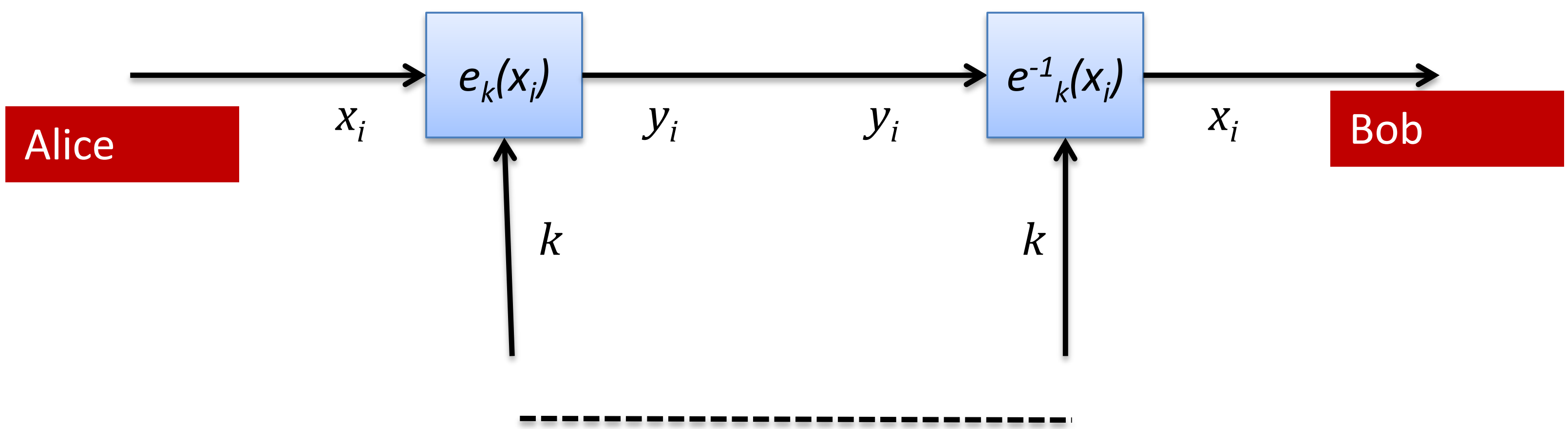

- Can be used in several ways: “modes of operation”

# Block Cipher Modes of Operation



# Electronic Codebook (ECB)

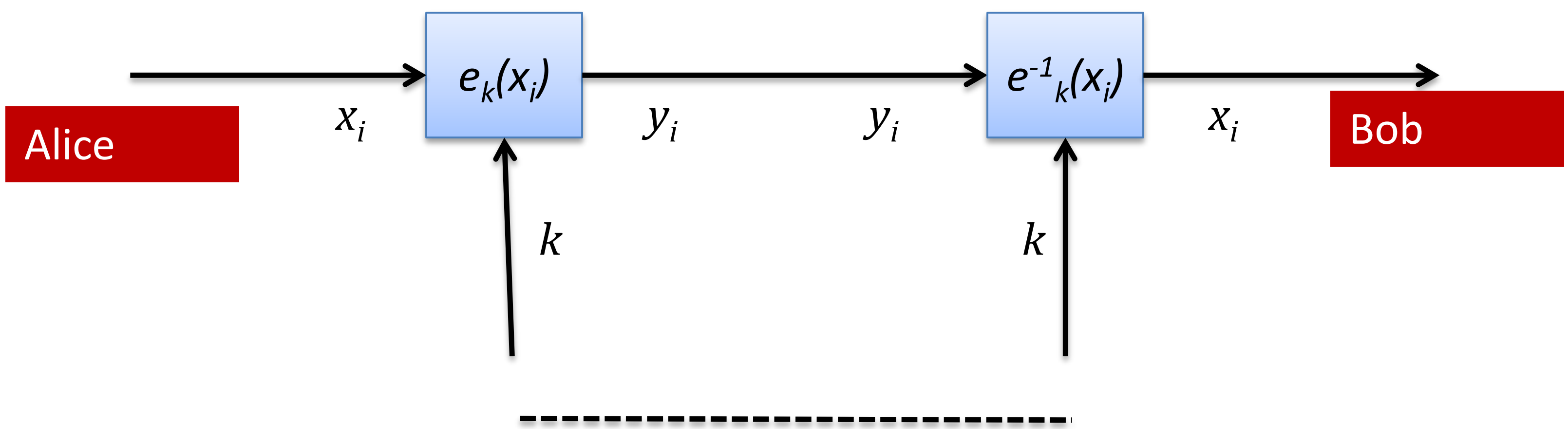
- Let $x_i$ , $y_i$ be blocks and $\mathrm{e_k}()$ be a block cipher of size $b$
- Then
  - Encryption: $y_i = e_k(x_i)$ , $i \geq 1$
  - Decryption: $x_i = e^{-1}{}_k(y_i) = e^{-1}{}_k(e_k(x_i))$ , $i \geq 1$




*Secure channel to share k*

# Electronic Codebook (ECB)

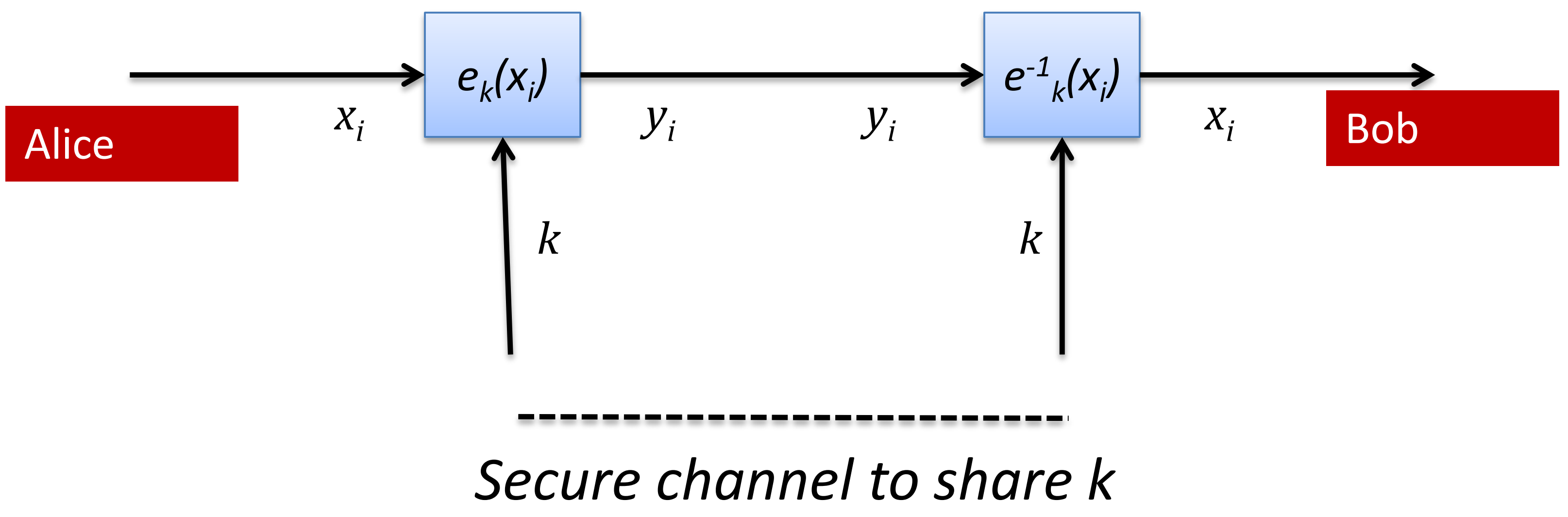
- Advantages
  - *Asynchronous* - Block synchronization is not necessary to decode all received blocks
  - Bit errors in noisy transmission affects *isolated* to a block
  - Encryption and Decryption are *Parallelizable* operations




*Secure channel to share k*

# Electronic Codebook (ECB)

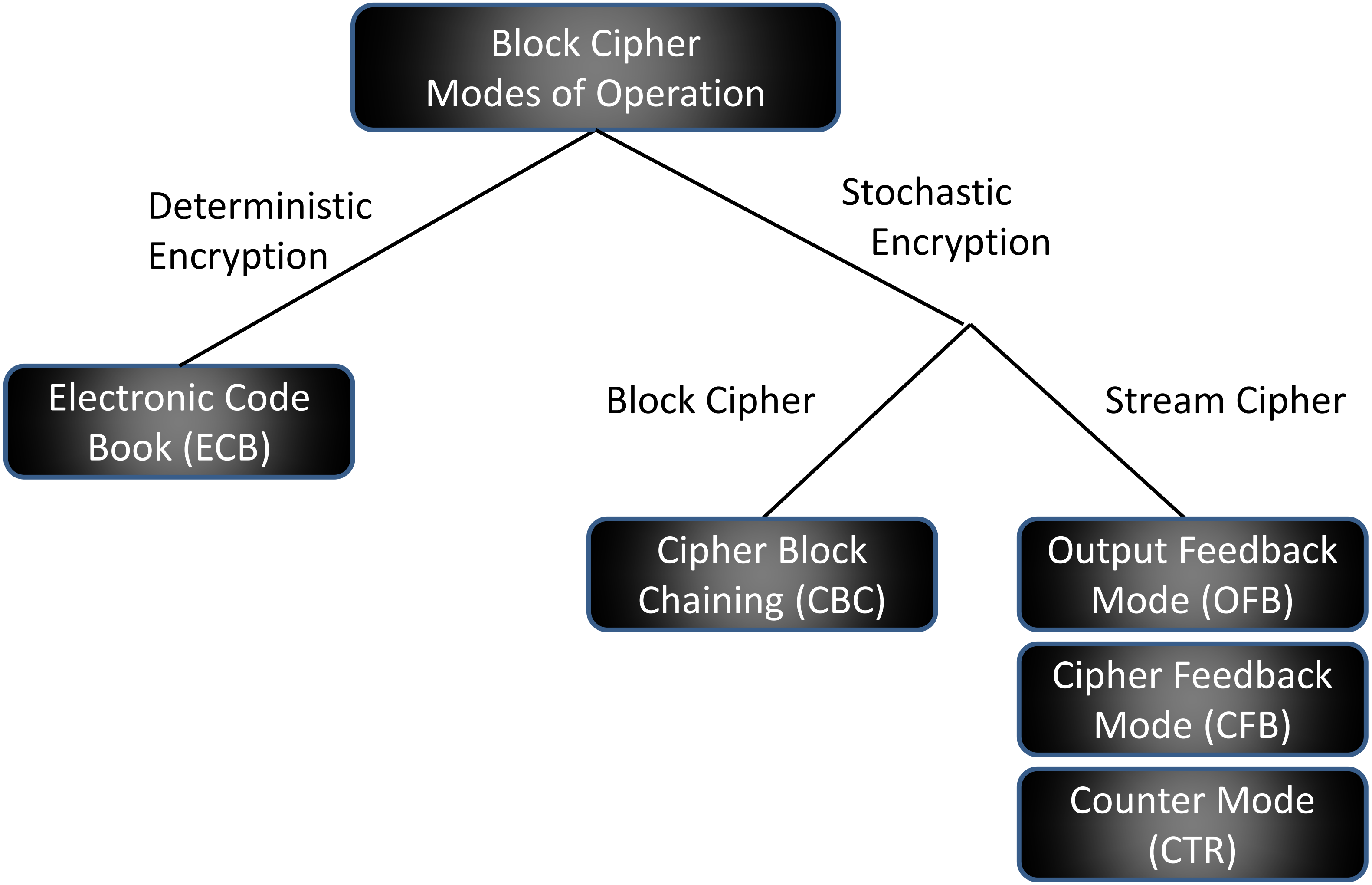
- Disadvantages
  - *Determinism* creates predictable cipher: Identical blocks always encrypt the same with same key
  - Statistical properties in plaintext may be preserved in ciphertext
  - *Attacks* - Subject to reordering, substitution and impersonation



# ECB Workarounds

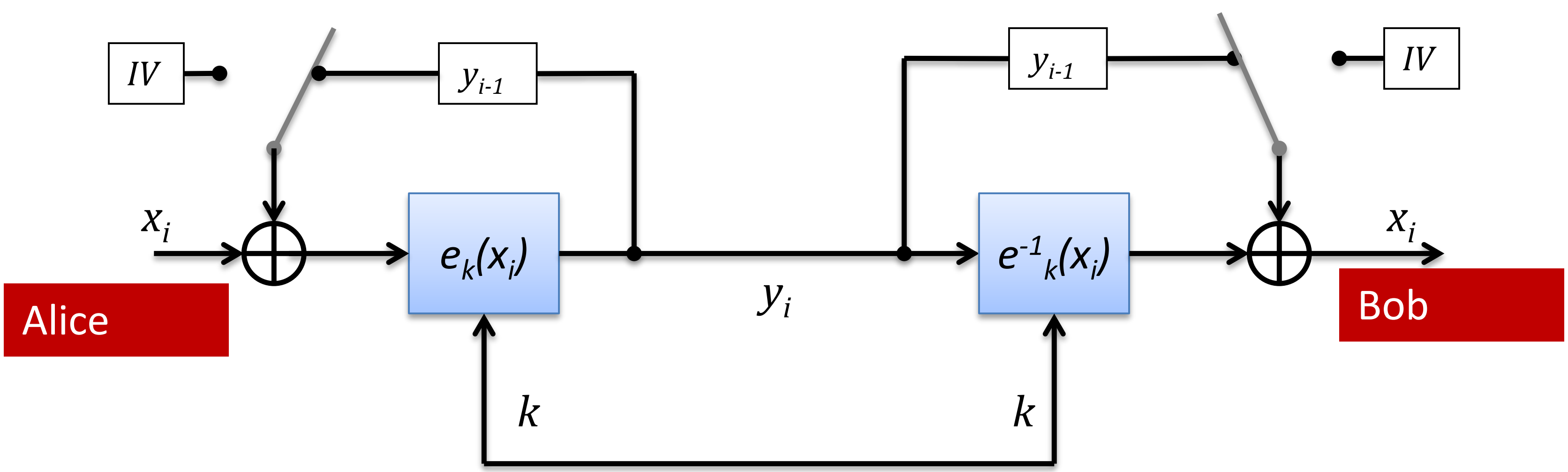
- Add integrity mechanisms
  - Digital signatures
  - Message Authentication Codes
- Randomize or refresh the key frequently
  - Build a ‘key selector’ layer above the crypto system
- Alter the input before encryption
  - Encoding – reduce redundancy
  - Randomize – XOR with a nonce (next mode)

# Block Cipher Modes of Operation



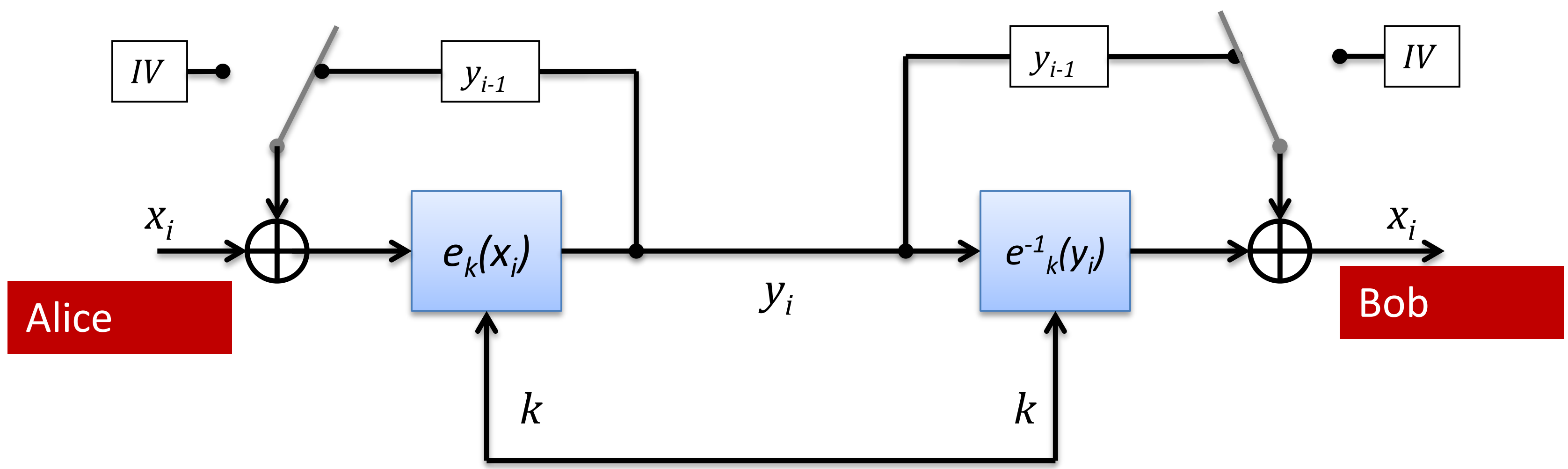
# Cipher Block Chaining (CBC)

- Chaining: Ciphertext ($y_{i-1}$) XOR'd with plaintext ($x_i$)
- Encryption randomized with initialization vector (*IV*)



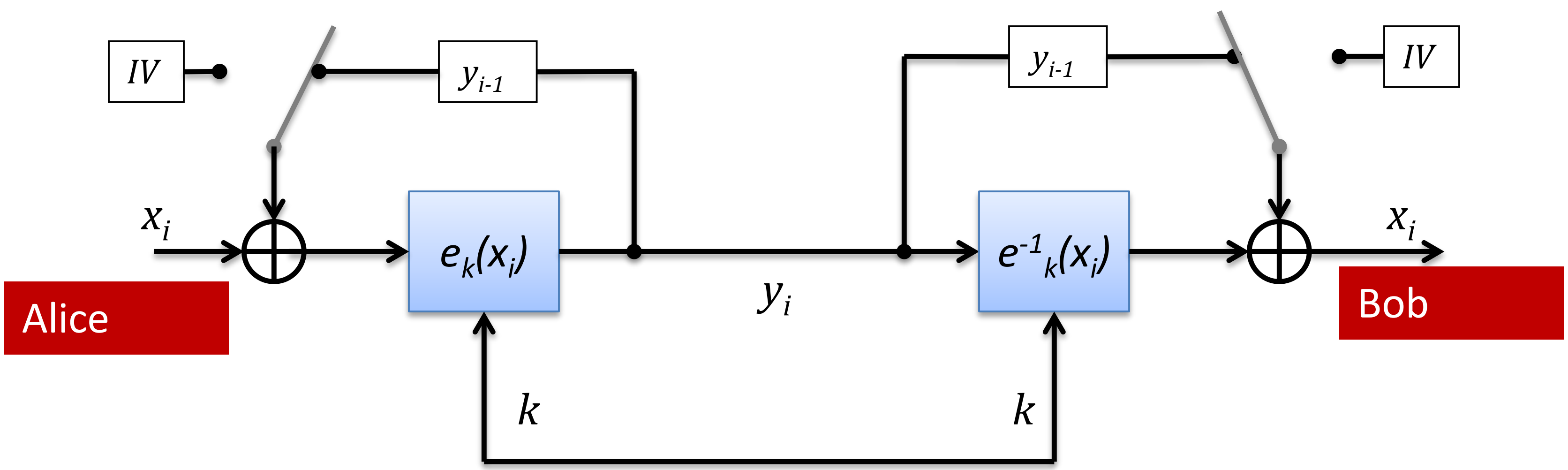
# Cipher Block Chaining (CBC)

- Let $x_i$ , $y_i$ be blocks, $e_k()$, a block cipher & $IV$, a *nonce* of size $b$
  - (first) Encryption: $y_1 = e_k(x_1 \oplus IV)$
  - (general) Encryption: $y_i = e_k(x_i \oplus y_{i-1})$ , $i \geq 2$
  - (first) Decryption: $x_1 = e^{-1}_k(y_1) \oplus IV$
  - (general) Decryption: $x_i = e^{-1}_k(y_i) \oplus y_{i-1}$, $i \geq 2$



# Cipher Block Chaining (CBC)

- Consider the sending of two identical files, $f_1$ and $f_2$
- What happens if $f_1$ and $f_2$ are sent with $k_1 = k_2$ and $IV_1 = IV_2$?
- What happens if $f_1$ and $f_2$ are sent with $k_1 = k_2$ and $IV_1 = IV_2+1$?



# Cipher Block Chaining (CBC)

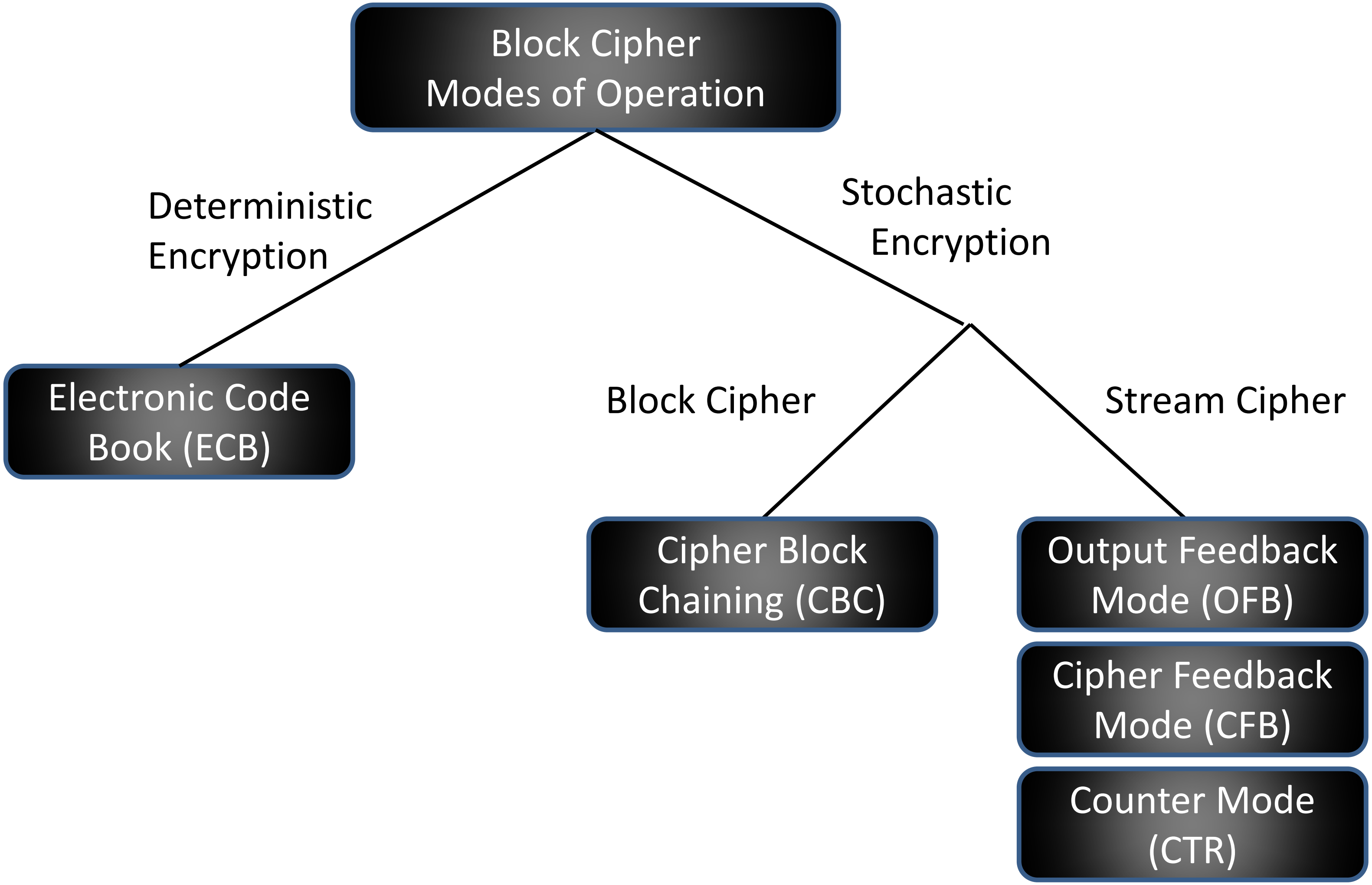
- Advantages
  - Non deterministic encryption if *IV* is new
  - Robust to reordering, better with substitution attacks
  - Helps prevent impersonation (replay) attacks
  - Reduced statistical relationship in plaintext-ciphertext

- Disadvantages
  - Not parallelizable for a single message transmission
  - Block synchronization needed to decode received blocks
  - Noisy transmission results in cascading failures
  - Still susceptible to substitution and integrity attacks

# CBC Workarounds

- Add integrity mechanisms
  - Digital signatures
  - Message Authentication Codes
- Ensure synchronization and reliability
  - Reliable transmission protocol – TCP
  - Message queuing and QoS middleware
- Parallelize groups of messages - break a message into groups to increase parallelization

# Block Cipher Modes of Operation



# Block Modes for Stream Ciphers

- Streaming: Moves plaintext ($x_i$) outside of block encryption function
  - Encrypt: Plaintext ($x_i$) XOR'd with enciphered input ($s_i$)

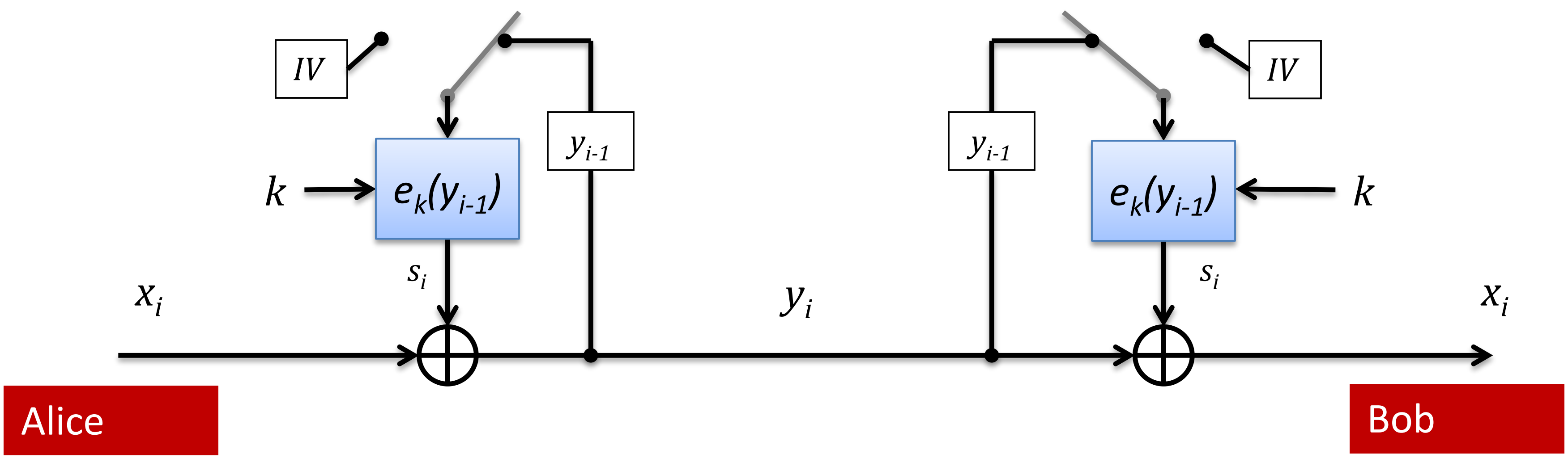
    $y_i = s_i \oplus x_i$
  - Decrypt: Ciphertext ($y_i$) XOR'd with enciphered input ($s_i$)

    $x_i = s_i \oplus y_i$

- 3 (+1) Types
  - Output Feedback Mode (OFB)
  - Cipher Feedback Mode (CFB)
  - Counter Mode (CTR)
    - +1 – Galois Counter Mode (GCM)

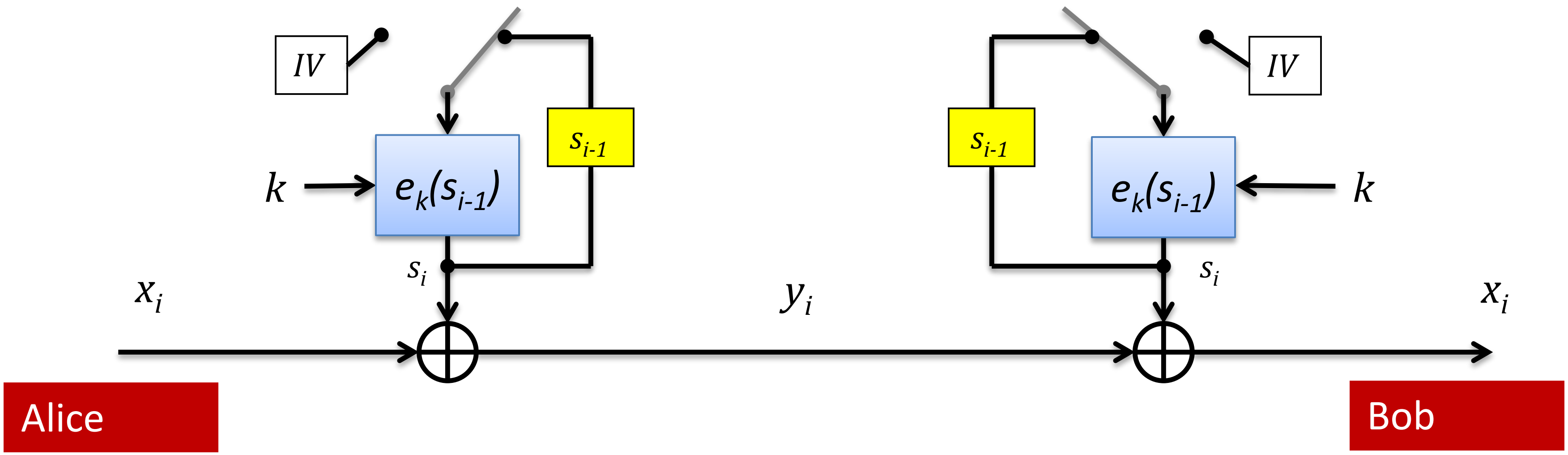
# Block Modes for Stream Ciphers
# *Cipher Feedback Mode (CFB)*

- Let $x_i$ , $y_i$ be blocks, $e_k()$, a block cipher & $IV$, a *nonce* of size $b$
  - (first) Encryption: $y_1 = e_k(IV) \oplus x_1$
  - (general) Encryption: $y_i = e_k(y_{i-1}) \oplus x_i$ , $i \geq 2$
  - (first) Decryption: $x_1 = e_k(IV) \oplus y_1$
  - (general) Decryption: $x_i = e_k(y_{i-1}) \oplus y_i$ , $i \geq 2$



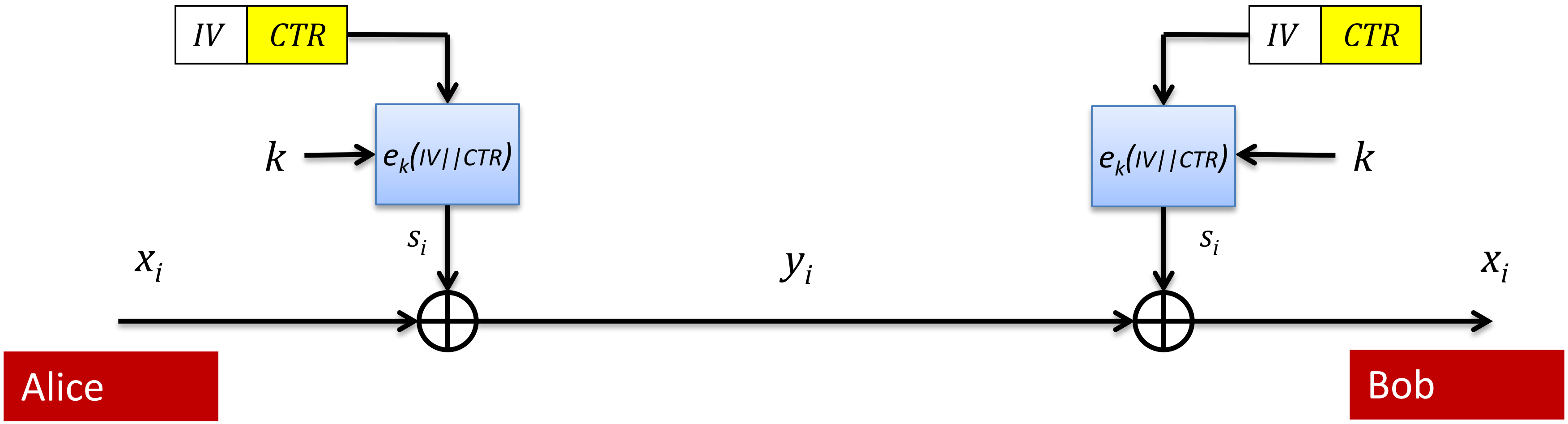
# Block Modes for Stream Ciphers
## ***Output Feedback Mode (OFB)***

- Let $x_i$ , $y_i$ be blocks, $e_k()$, a block cipher & $IV$, a *nonce* of size $b$
  - (first) Encryption: $y_1 = e_k(IV) \oplus x_1$
  - (general) Encryption: $y_i = e_k(s_{i-1}) \oplus x_i$ , $i \geq 2$
  - (first) Decryption: $x_1 = e_k(IV) \oplus y_1$
  - (general) Decryption: $x_i = e_k(s_{i-1}) \oplus y_i$ , $i \geq 2$



# Block Modes for Stream Ciphers
# ***Counter Mode (CTR)***

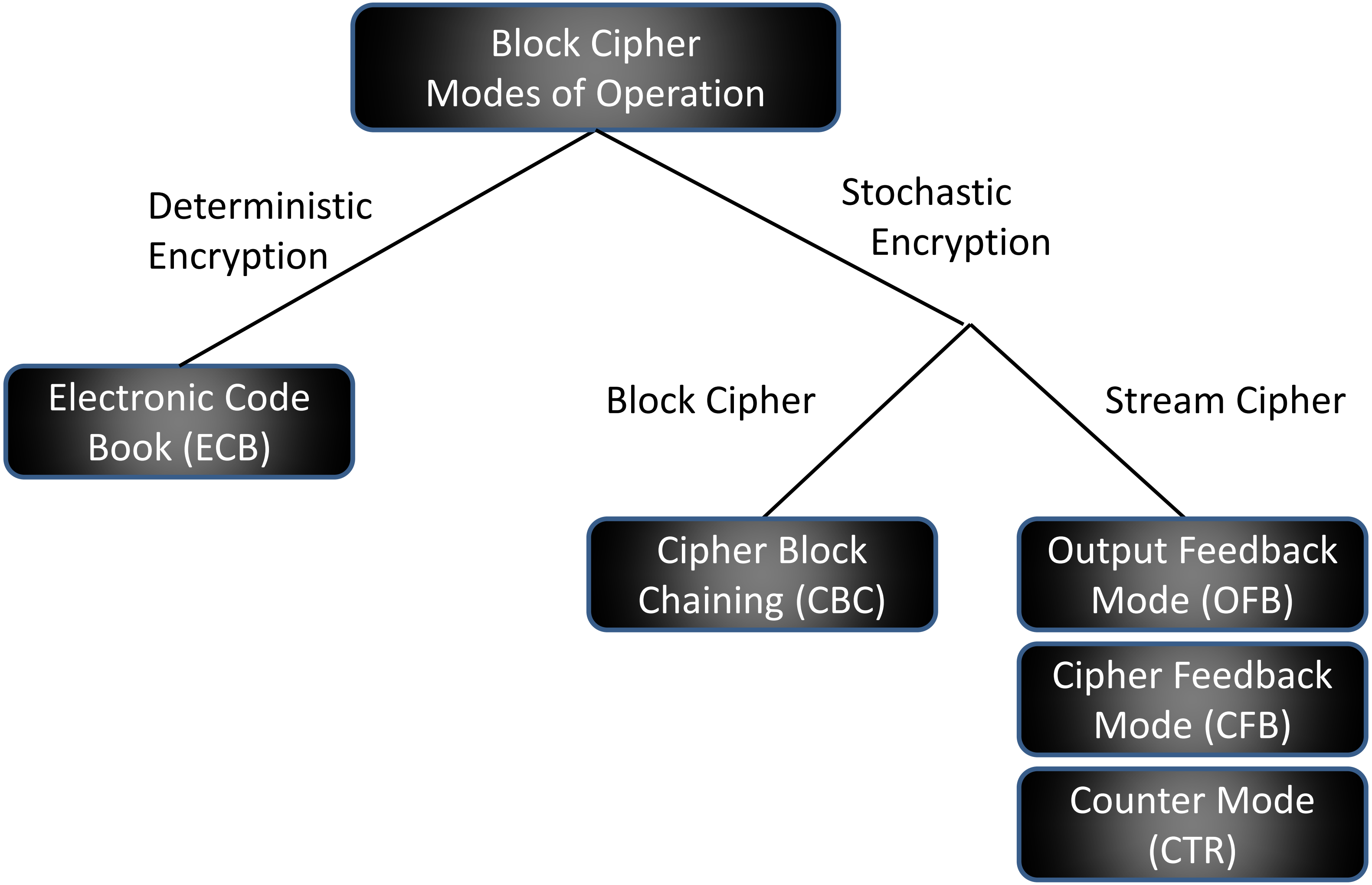
- Let $x_i$ , $y_i$ be blocks, $e_k()$, a block cipher of size $b$
  - $IV$ (*nonce)* and CTR (counter) are concatenated (||) string
  - CTR initialized to 0, $IV||CTR$ of size $b$
  - Encryption: $y_i = e_k(IV||CTR) \oplus x_i$ , $i \geq 1$
  - Decryption: $x_i = e_k(IV||CTR) \oplus y_i$ , $i \geq 1$



# Block Modes for Stream Ciphers

- CFB
  - Robust to reordering attacks
  - Faster than CBC (bitwise XOR)
- OFB
  - Eliminates error propagation (rely on $s_i$ )
  - Increases parallelism with independence from cipher
  - Lacks integrity mechanism
- CTR
  - Can share (*IV* || *CTR*) in the clear
  - Parallelizable (encrypt for next *n* time steps)
  - Can provide message integrity (Galois Counter Mode)

# Block Cipher Modes of Operation



# Block Ciphers

- Most common symmetric encryption
- Most are based on product ciphers
  - Confusion (substitution)
  - Diffusion (permutation)
- 5 modes of operation
  - ECB, CBC, CFB, OFB, CTR
- Many types: (i.e. DES, 3DES, AES)