Cryptography and Network Security Chapter 6

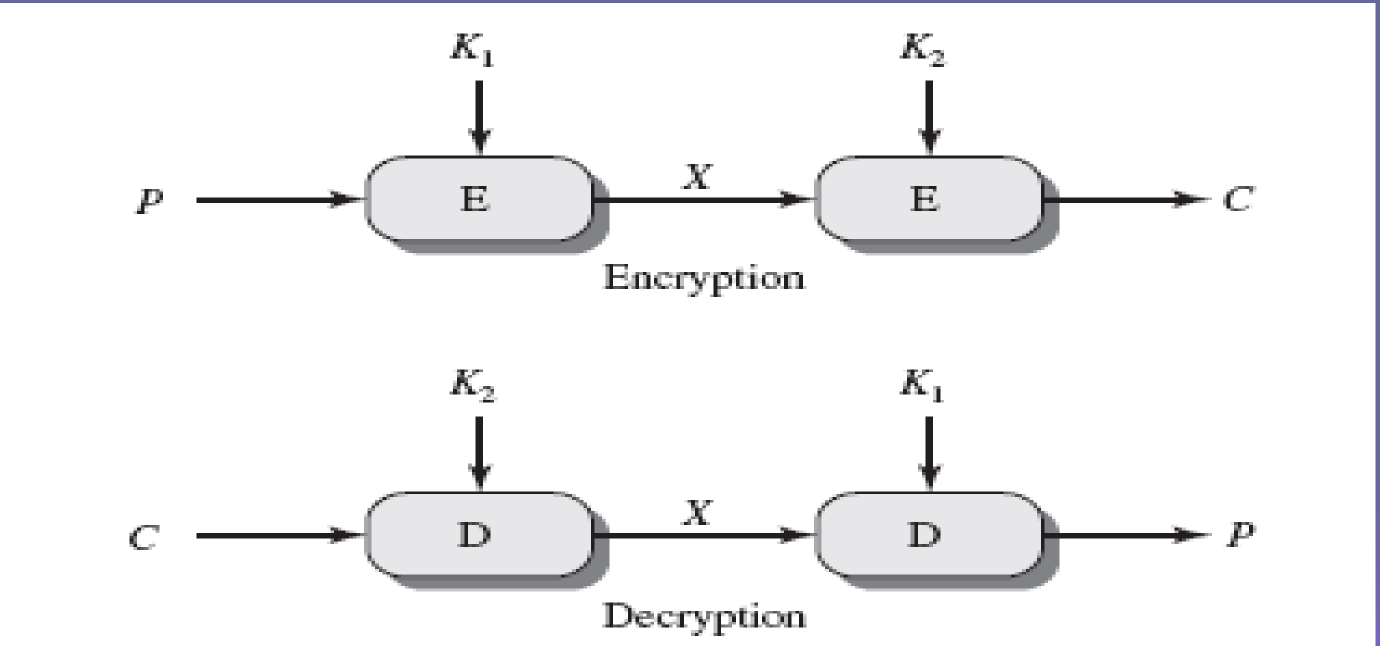
Fifth Edition

by William Stallings

Multiple Encryption & DES

- clear a replacement for DES was needed
  - theoretical attacks that can break it
  - demonstrated exhaustive key search attacks
- AES is a new cipher alternative
- prior to this alternative was to use multiple encryption with DES implementations
- Triple-DES is the chosen form

Double DES



(a) Double encryption

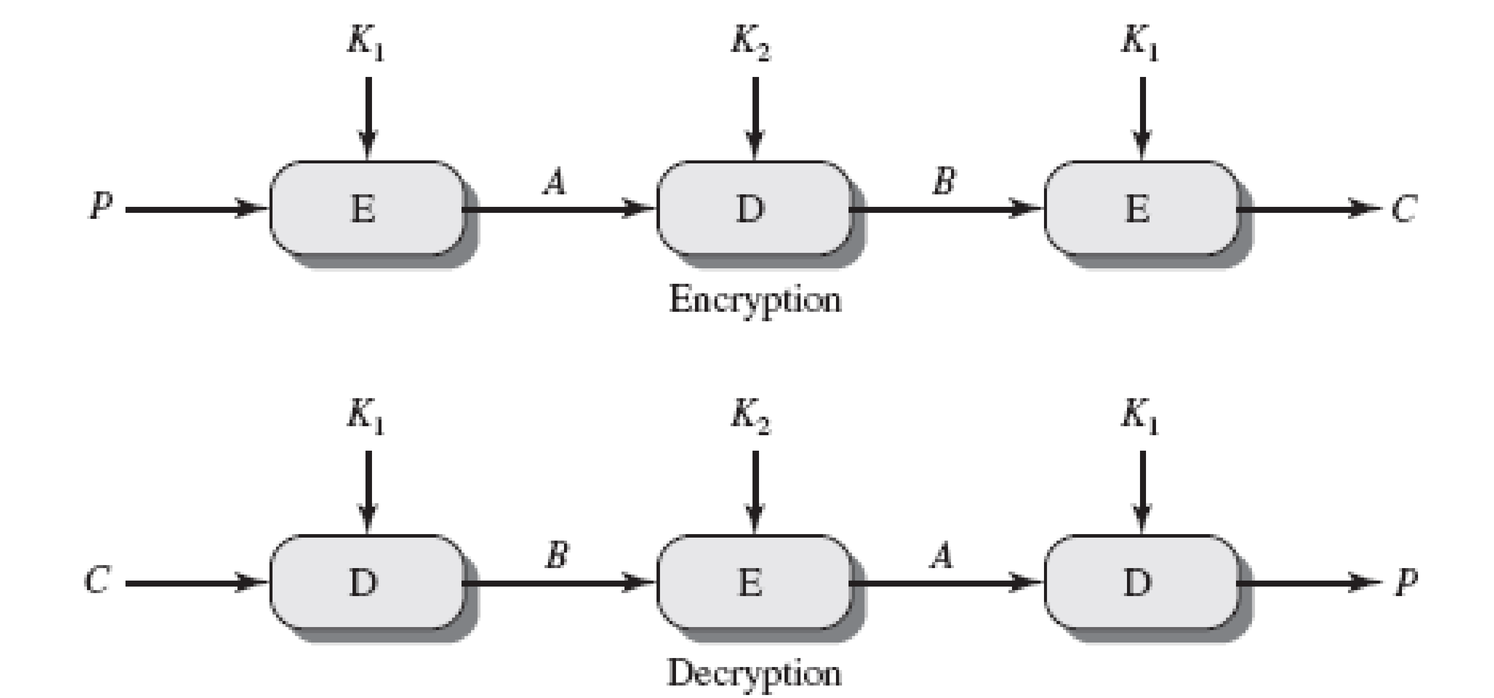
Double-DES?

- could use 2 DES encrypts on each block
  - $C = E_{K2}(E_{K1}(P)) = E_{K3}(P)$
- issue of reduction to single stage
- and have "meet-in-the-middle" attack
  - works whenever use a cipher twice
  - since $X = E_{K1}(P) = D_{K2}(C)$
  - attack by encrypting P with all keys and store
  - then decrypt C with keys and match X value
  - can show takes $O(2^{56})$ steps

Triple-DES with Two-Keys

- hence must use 3 encryptions
  - would seem to need 3 distinct keys
- but can use 2 keys with E-D-E sequence
  - $C = E_{K1}(D_{K2}(E_{K1}(P)))$
  - nb encrypt & decrypt equivalent in security
  - if `K1=K2` then can work with single DES
- standardized in ANSI X9.17 & ISO8732
- no current known practical attacks
  - several proposed impractical attacks might become basis of future attacks

Triple-DES



(b) Triple encryption

Triple-DES with Three-Keys

- although are no practical attacks on two-key Triple-DES have some indications
- can use Triple-DES with Three-Keys to avoid even these
  - $C = E_{K3}(D_{K2}(E_{K1}(P)))$
- has been adopted by some Internet applications, eg PGP, S/MIME

Modes of Operation

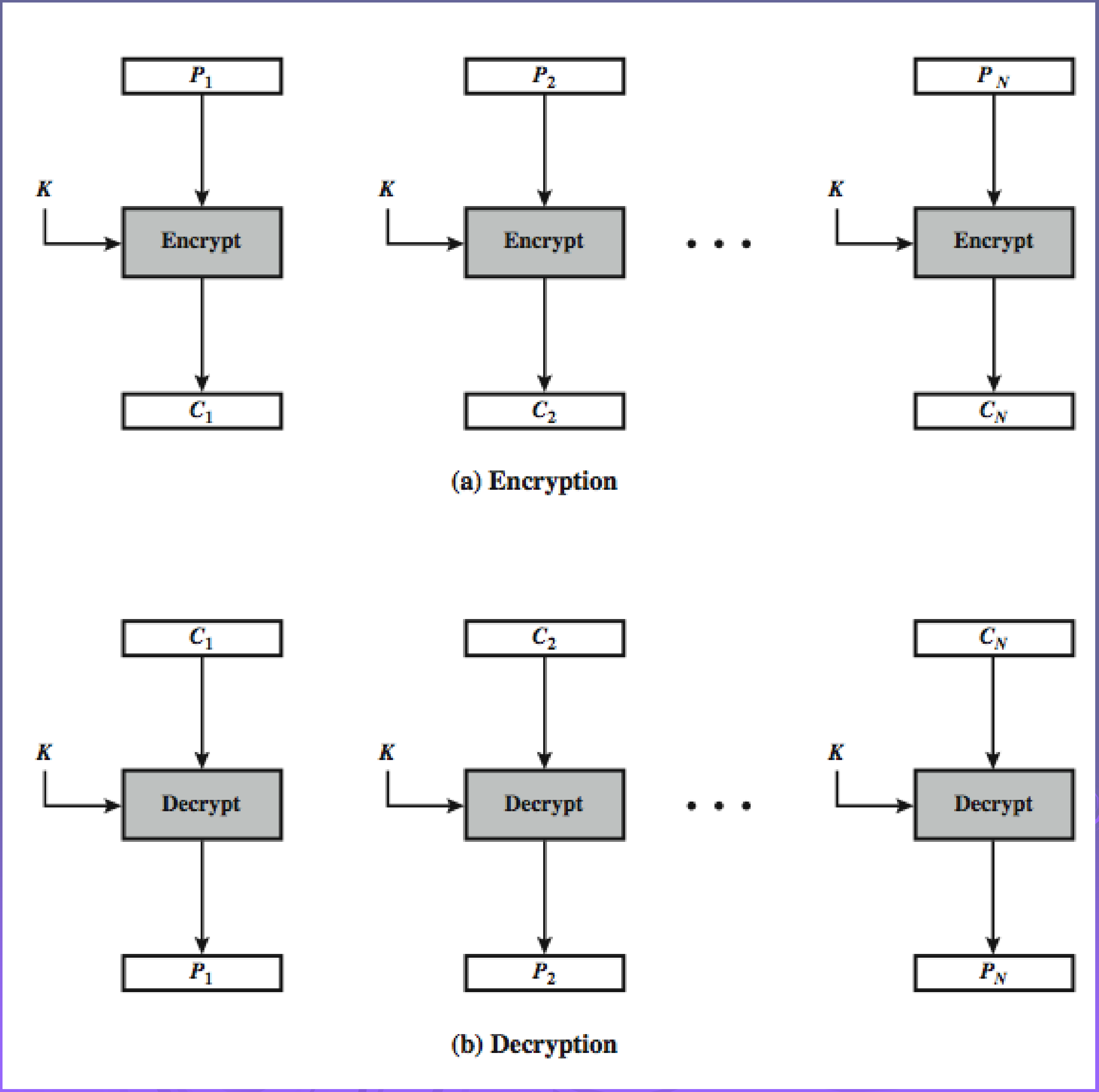
- block ciphers encrypt fixed size blocks
  - eg. DES encrypts 64-bit blocks with 56-bit key
- need some way to en/decrypt arbitrary amounts of data in practise
- NIST SP 800-38A defines 5 modes
- have **block** and **stream** modes
- to cover a wide variety of applications
- can be used with any block cipher

Electronic Codebook Book (ECB)

- message is broken into independent blocks which are encrypted
- each block is a value which is substituted, like a codebook, hence name
- each block is encoded independently of the other blocks

  $C_i = E_K(P_i)$
- uses: secure transmission of single values

Electronic Codebook Book (ECB)



Advantages and Limitations of ECB

- message repetitions may show in ciphertext
  - if aligned with message block
  - particularly with data such graphics
  - or with messages that change very little, which become a code-book analysis problem
- weakness is due to the encrypted message blocks being independent
- main use is sending a few blocks of data

Cipher Block Chaining (CBC)

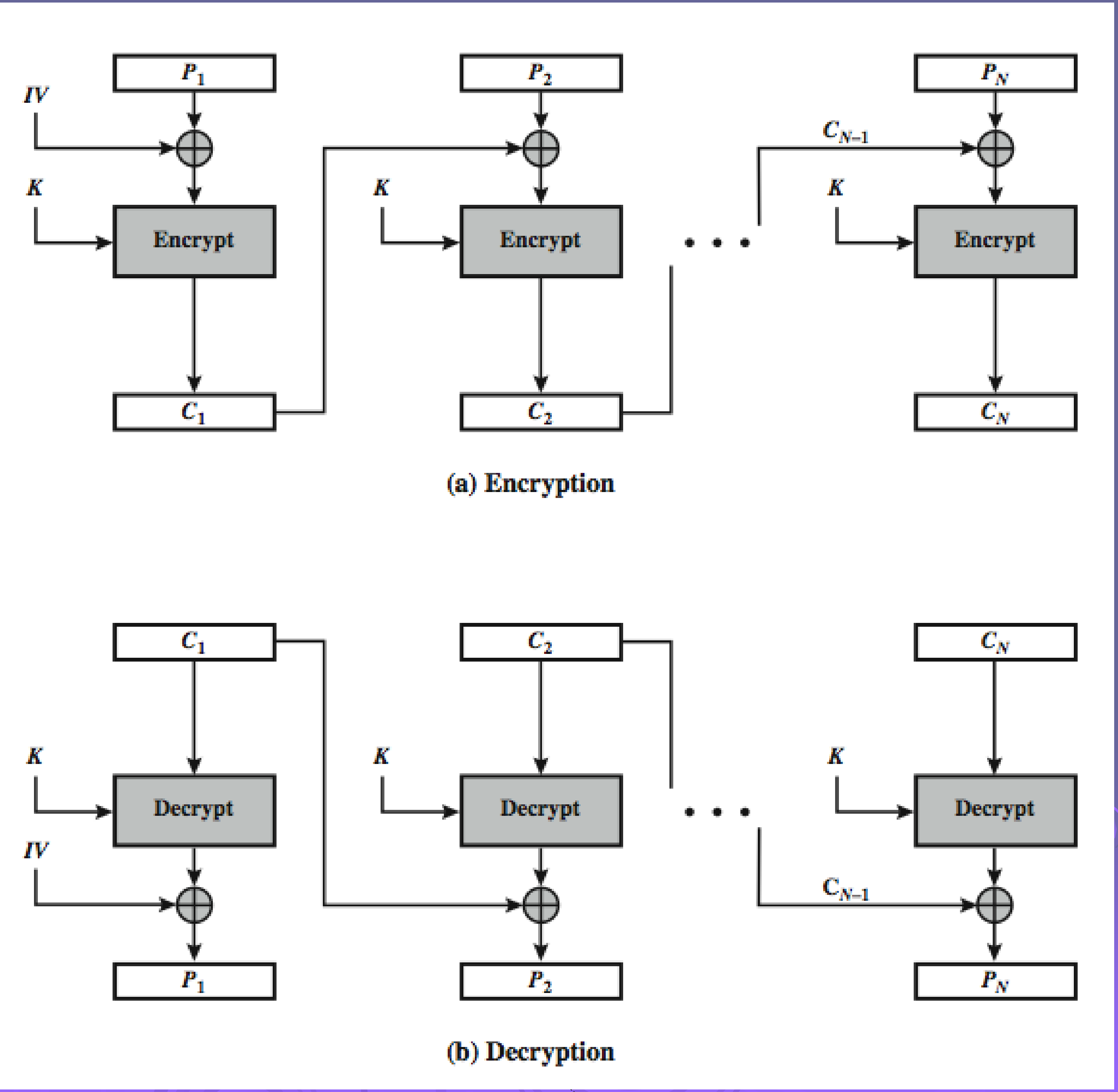
- message is broken into blocks
- linked together in encryption operation
- each previous cipher blocks is chained with current plaintext block, hence name
- use Initial Vector (IV) to start process

  $C_i = E_K(P_i \text{ XOR } C_{i-1})$

  $C_{-1} = IV$

- uses: bulk data encryption, authentication

Cipher Block Chaining (CBC)



Message Padding

- at end of message must handle a possible last short block
  - which is not as large as blocksize of cipher
  - pad either with known non-data value (eg nulls)
  - or pad last block along with count of pad size
    - eg. [b1 b2 b3 0 0 0 0 5]
    - means have 3 data bytes, then 5 bytes pad+count
  - this may require an extra entire block over those in message
- there are other, more esoteric modes, which avoid the need for an extra block

Advantages and Limitations of CBC

- a ciphertext block depends on **all** blocks before it
- any change to a block affects all following ciphertext blocks
- need **Initialization Vector** (IV)
  - which must be known to sender & receiver
  - if sent in clear, attacker can change bits of first block, and change IV to compensate
  - hence IV must either be a fixed value (as in EFTPOS)
  - or must be sent encrypted in ECB mode before rest of message

Stream Modes of Operation

- block modes encrypt entire block
- may need to operate on smaller units
  - real time data
- convert block cipher into stream cipher
  - cipher feedback (CFB) mode
  - output feedback (OFB) mode
  - counter (CTR) mode
- use block cipher as some form of **pseudo-random number** generator

Cipher FeedBack (CFB)

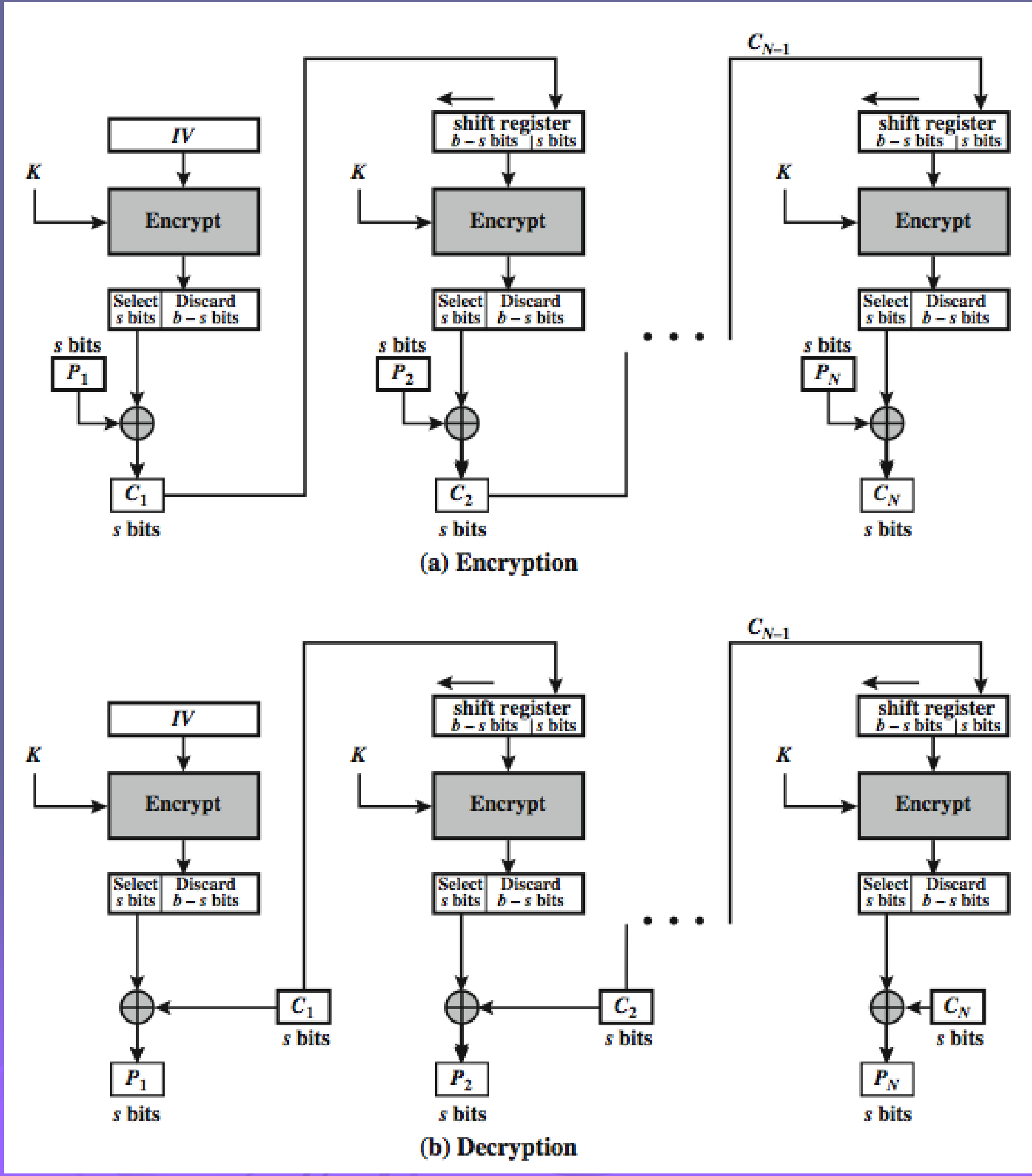
- message is treated as a stream of bits
- added to the output of the block cipher
- result is feed back for next stage (hence name)
- standard allows any number of bit (1,8, 64 or 128 etc) to be feed back
  - denoted CFB-1, CFB-8, CFB-64, CFB-128 etc
- most efficient to use all bits in block (64 or 128)

$$C_i = P_i \text{ XOR } E_K(C_{i-1})$$

$$C_{-1} = IV$$

- uses: stream data encryption, authentication

s-bit Cipher FeedBack (CFB-s)



Advantages and Limitations of CFB

- appropriate when data arrives in bits/bytes
- most common stream mode
- limitation is need to stall while do block encryption after every n-bits
- note that the block cipher is used in **encryption** mode at **both** ends
- errors propogate for several blocks after the error

Output FeedBack (OFB)

- message is treated as a stream of bits
- output of cipher is added to message
- output is then feed back (hence name)
- feedback is independent of message
- can be computed in advance

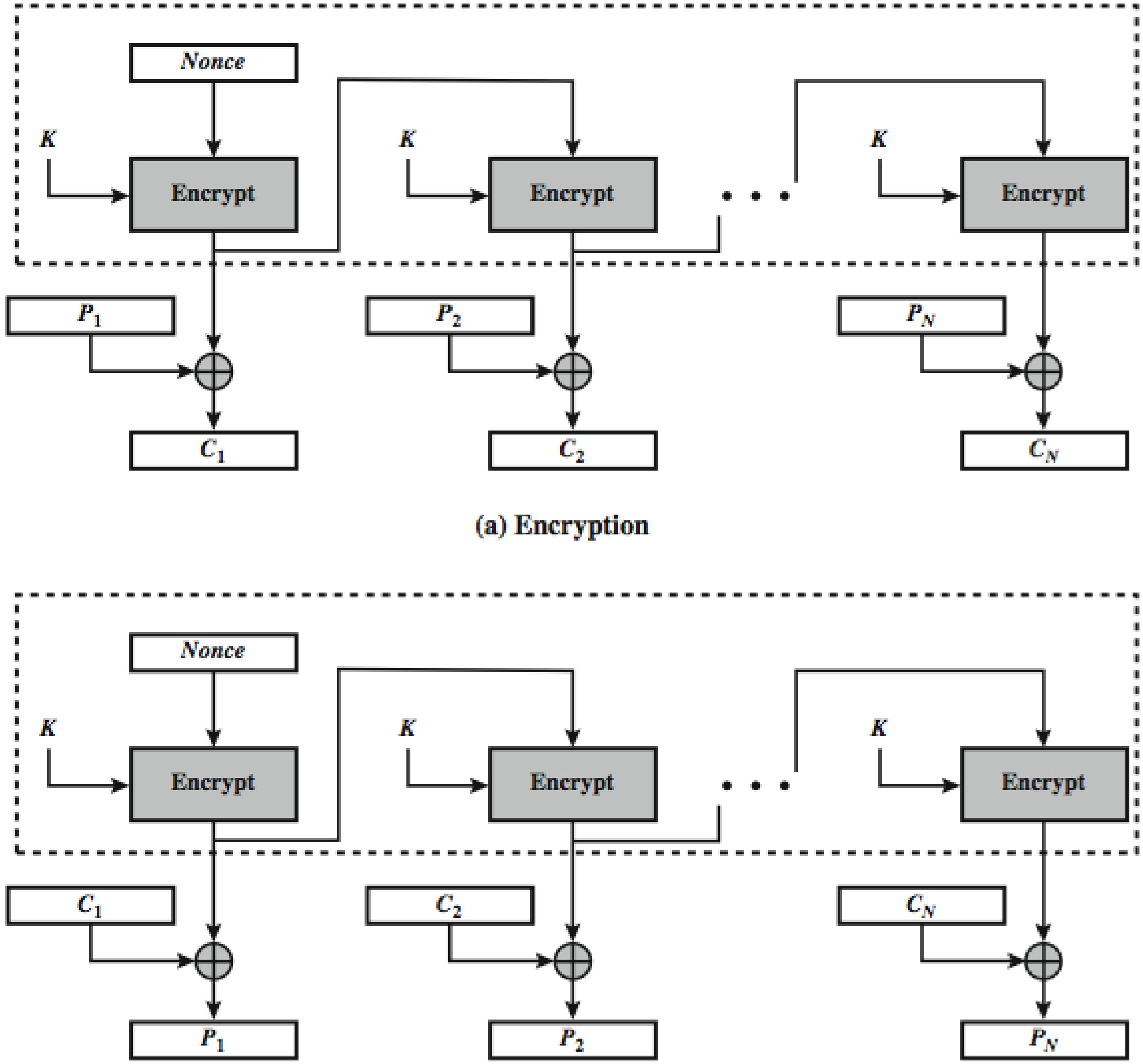
  $O_i = E_K(O_{i-1})$

  $C_i = P_i \text{ XOR } O_i$

  $O_{-1} = IV$

- uses: stream encryption on noisy channels

Output FeedBack (OFB)



(a) Encryption

(b) Decryption

Advantages and Limitations of OFB

- needs an IV which is unique for each use
  - if ever reuse attacker can recover outputs
- bit errors do not propagate
- more vulnerable to message stream modification
- sender & receiver must remain in sync
- only use with full block feedback
  - subsequent research has shown that only **full block feedback** (ie CFB-64 or CFB-128) should ever be used

Counter (CTR)

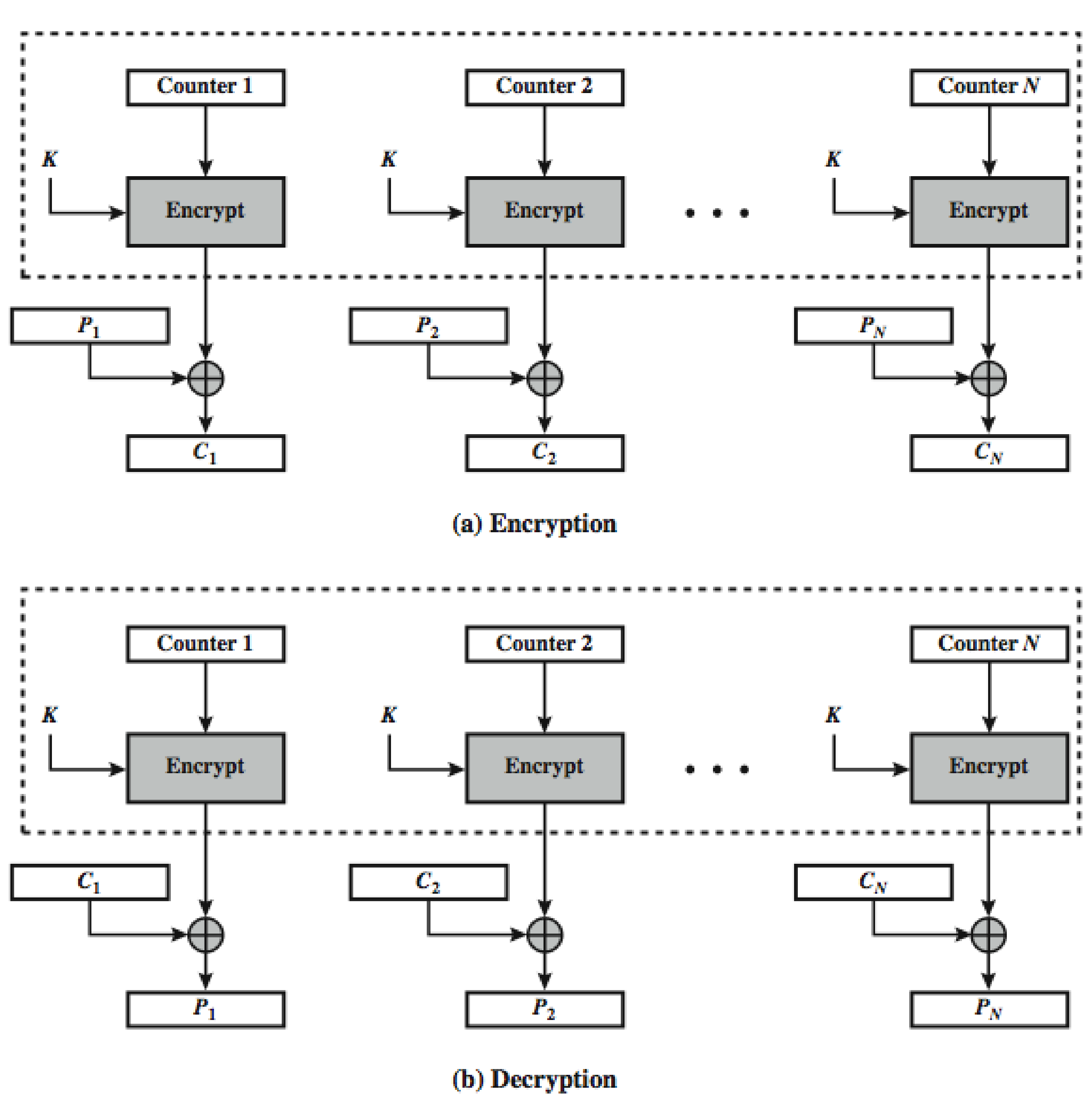
- a "new" mode, though proposed early on
- similar to OFB but encrypts counter value rather than any feedback value
- must have a different key & counter value for every plaintext block (never reused)

$$O_i = E_K(i)$$

$$C_i = P_i \text{ XOR } O_i$$

- uses: high-speed network encryptions

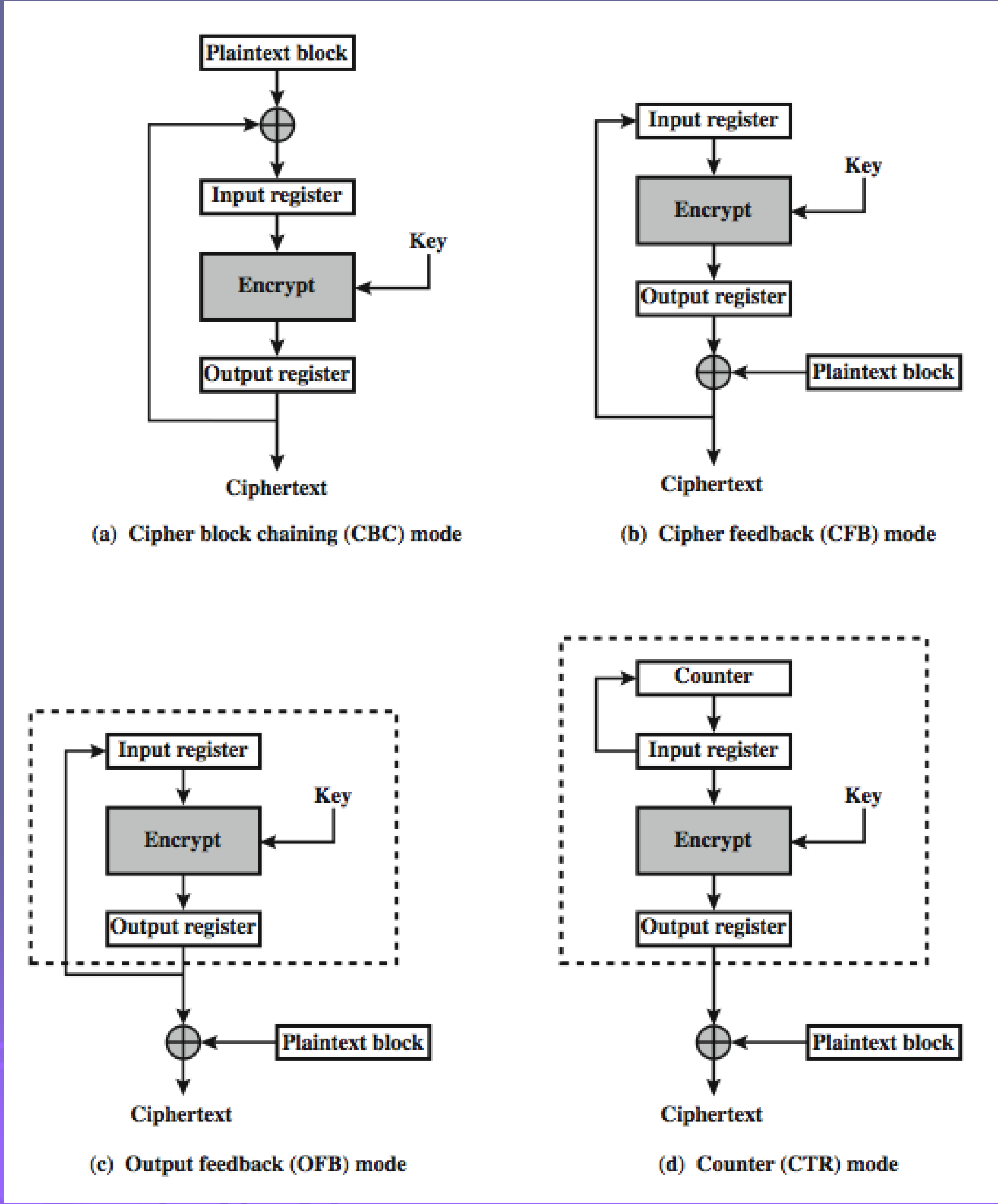
Counter (CTR)


Counter 1
Counter 2
Counter N
K
Encrypt
P_1
P_2
P_N
C_1
C_2
C_N
(a) Encryption
Counter 1
Counter 2
Counter N
K
Encrypt
C_1
C_2
C_N
P_1
P_2
P_N
(b) Decryption

Advantages and Limitations of CTR

- efficiency
  - can do parallel encryptions in h/w or s/w
  - can preprocess in advance of need
  - good for bursty high speed links
- random access to encrypted data blocks
- provable security (good as other modes)
- but must ensure never reuse key/counter values, otherwise could break (cf OFB)

Feedback Characteristics



Block Cipher Modes of Operation

Table 6.1 Block Cipher Modes of Operation

Mode	Description	Typical Application
Electronic Codebook (ECB)	Each block of plaintext bits is encoded independently using the same key.	• Secure transmission of single values (e.g., an encryption key)
Cipher Block Chaining (CBC)	The input to the encryption algorithm is the XOR of the next block of plaintext and the preceding block of ciphertext.	• General-purpose block-oriented transmission • Authentication
Cipher Feedback (CFB)	Input is processed *s* bits at a time. Preceding ciphertext is used as input to the encryption algorithm to produce pseudorandom output, which is XORed with plaintext to produce next unit of ciphertext.	• General-purpose stream-oriented transmission • Authentication
Output Feedback (OFB)	Similar to CFB, except that the input to the encryption algorithm is the preceding encryption output, and full blocks are used.	• Stream-oriented transmission over noisy channel (e.g., satellite communication)
Counter (CTR)	Each block of plaintext is XORed with an encrypted counter. The counter is incremented for each subsequent block.	• General-purpose block-oriented transmission • Useful for high-speed requirements

Summary

- Multiple Encryption & Triple-DES
- Modes of Operation
  - ECB, CBC, CFB, OFB, CTR, XTS-AES