University of Colorado **Boulder**

CSCI 3403 INTRO TO CYBERSECURITY

Lecture: 4-1

Topic: Applied Crypto

Presenter: Matt Niemiec

Announcements

- Please remember that the material for the extra credit is NOT optional!
- Please see Piazza for the clarification of the Javascript problem
- You should be in your groups on Moodle now
- Get PHP running on your machine
  - Pro tip: don't use Windows
  - A few options: biz.nf, CU CS VM, PHPStorm

Random Numbers

Random Numbers

- What makes a number random?
- Do random numbers exist?
- Can we generate random numbers?
- What sorts of things are random on a computer?

The Fundamental Problem?

- Computers always do exactly what they're told!

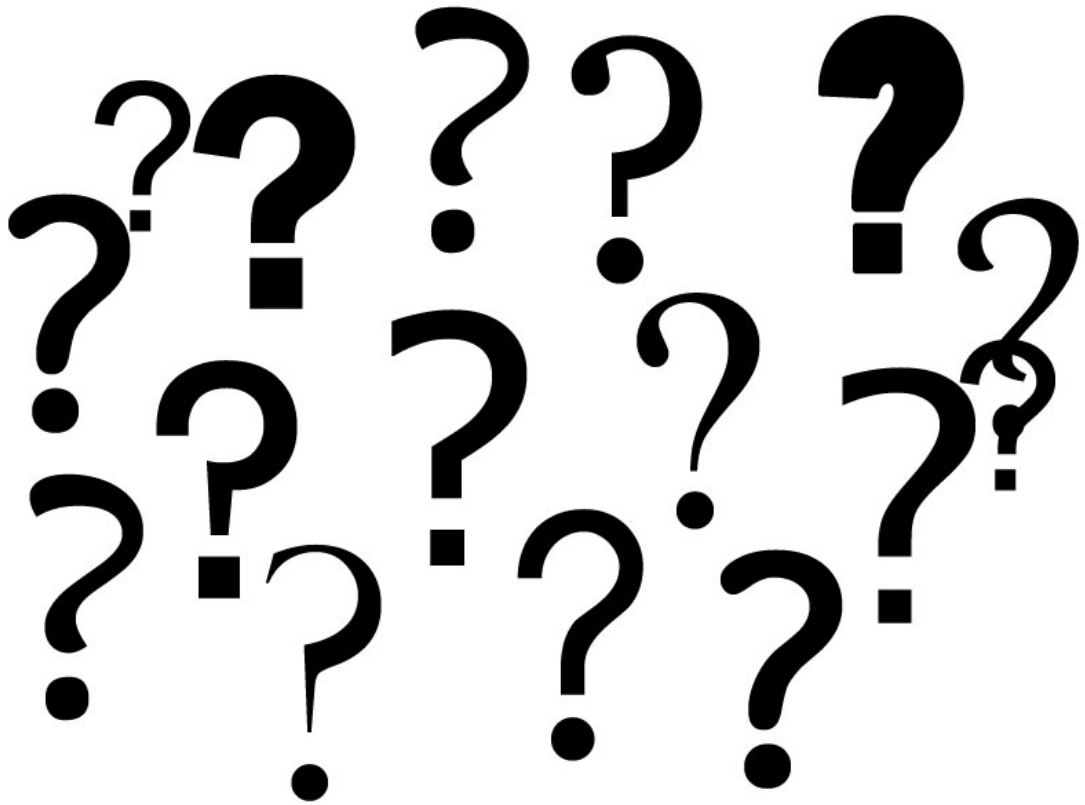
Why Do We Care?

- Generate symmetric secret keys
  - Create digital envelopes
  - Temporary session keys
  - Stream cipher randomness
- Generate prime numbers for RSA/public key
- Handshakes to prevent replay attacks

So We Need...

- Randomness
  - Statistically uniform distribution
  - Can’t infer any value in sequence from others
  - Good enough for machine learning, etc.
- Unpredictablility
  - Can’t predict future numbers
  - Can’t predict numbers based on environment variables

A Demo In C



Random Number Generators

- A random number generator is a pseudo-random number generator with a random seed
- Examples of bad seeds
  - Time
  - What else?
- Examples of good seeds
  - Radiation
  - Lava lamps
  - /dev/urandom
  - What else?

Encryption Modes

Block vs Stream Cipher

- What's the difference?
- Why does it matter?

Block vs Stream Cipher

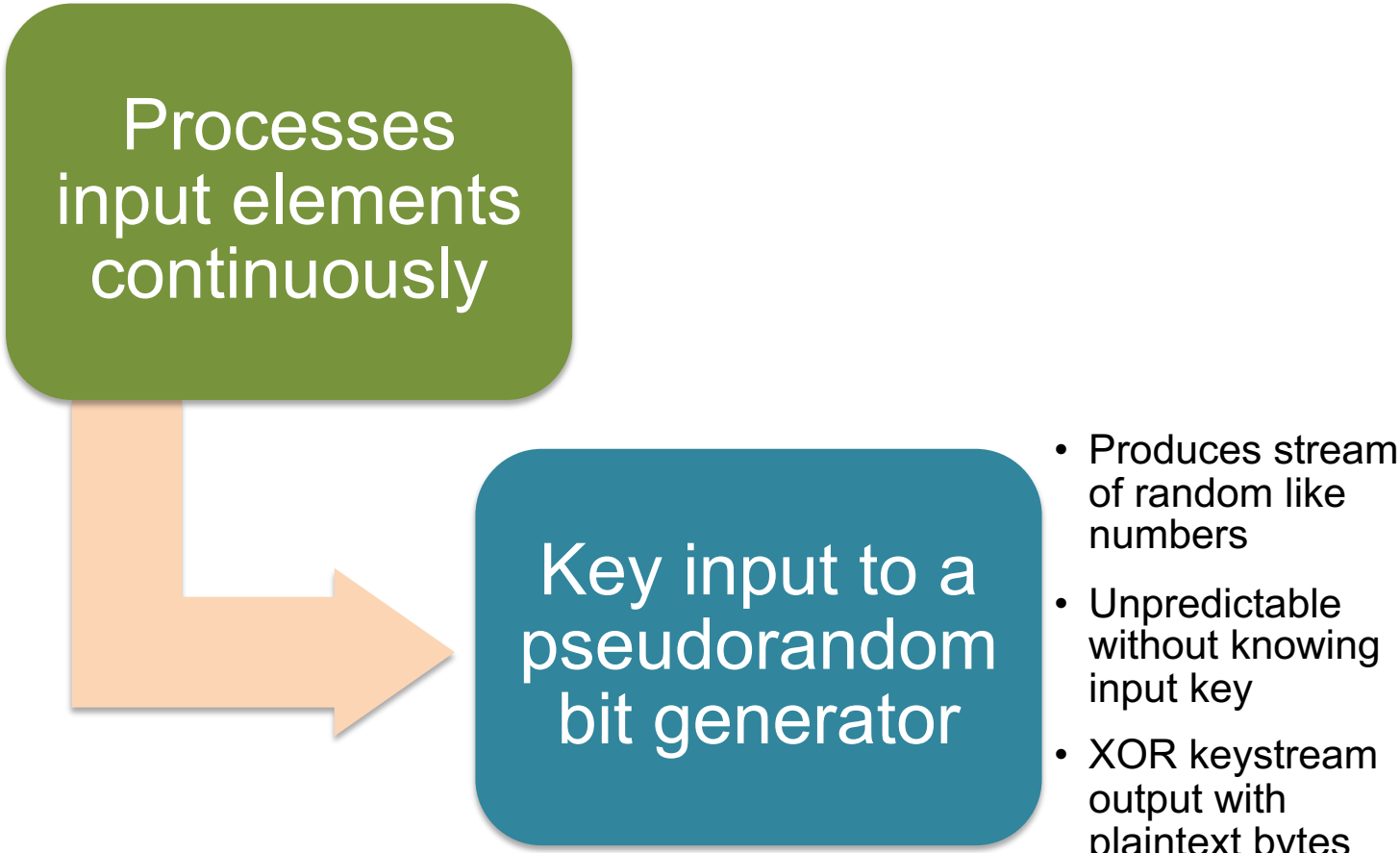
Block Cipher

- Processes the input one block of elements at a time
- Produces an output block for each input block
- Can reuse keys
- More common

Stream Cipher

- Processes the input elements continuously
- Produces output one element at a time
- Primary advantage is that they are almost always faster and use far less code
- Encrypts plaintext one byte at a time
- Pseudorandom stream is one that is unpredictable without knowledge of the input key

Stream Cipher



- Produces stream of random like numbers
- Unpredictable without knowing input key
- XOR keystream output with plaintext bytes

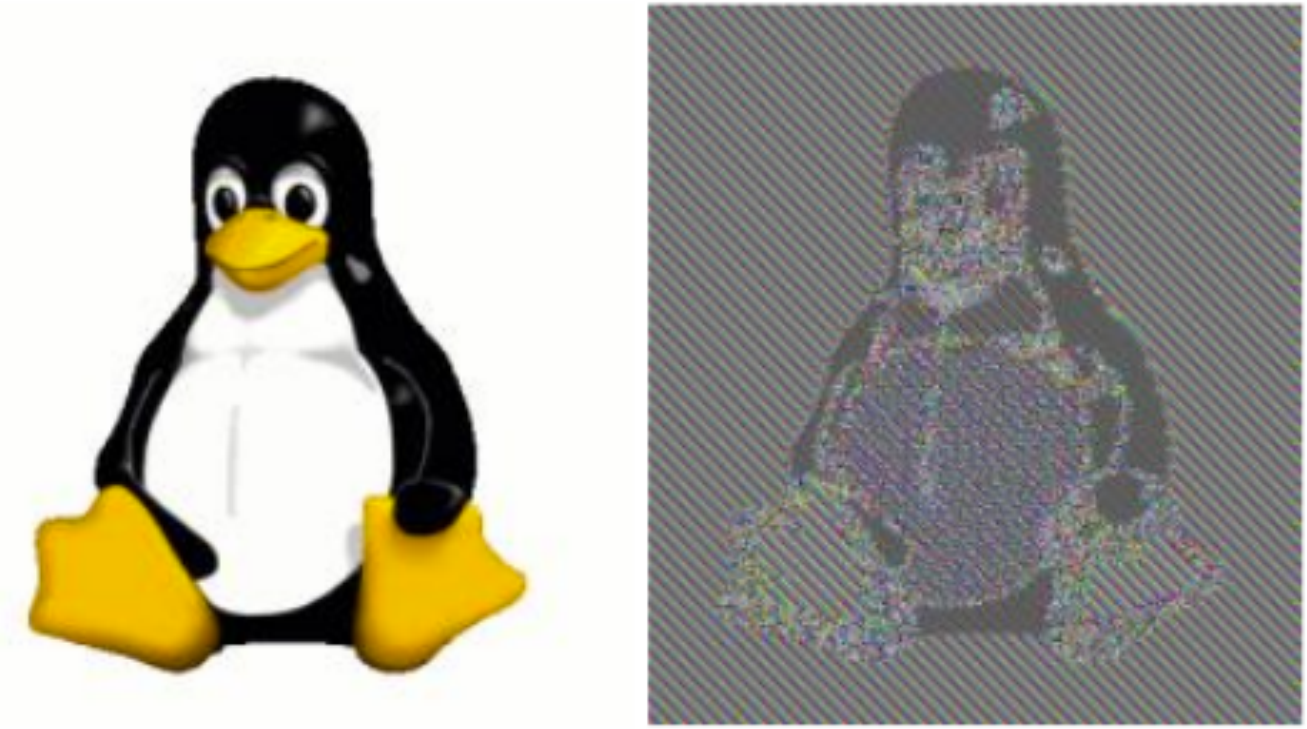
Can We Make a Block Cipher Look More Like a Stream Cipher?

- What are benefits/drawbacks that are at stake?
  - Parallelizability
  - Repeated block segments
  - Any extra sensitive data to protect
  - Pad messages to fit block length
  - Extra computations
  - Resilience
  - Subject to mathematical weaknesses/manipulation
  - An elegant solution

Electronic Codebook Mode (ECB)

- Encrypt one block at a time
- Advantages/drawbacks?

Electronic Codebook Mode (ECB)



(a) Plaintext

(b) ECB ciphertext

Cipher Block Chaining (CBC) Mode

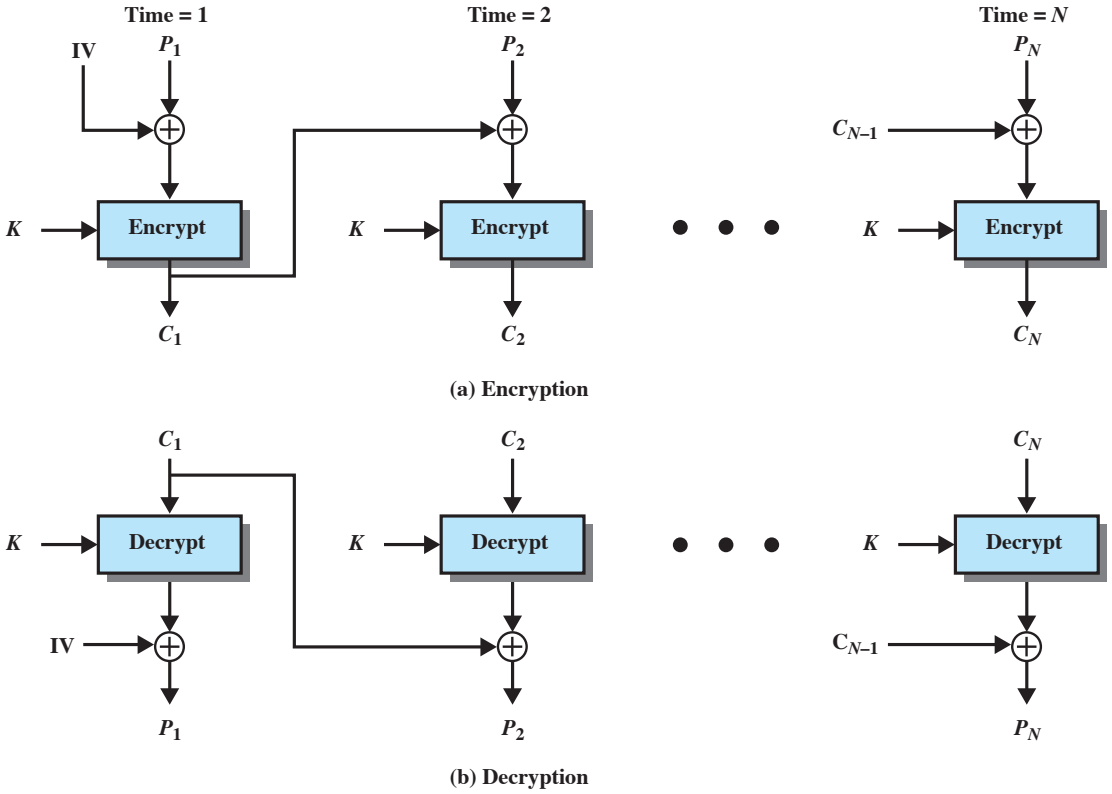
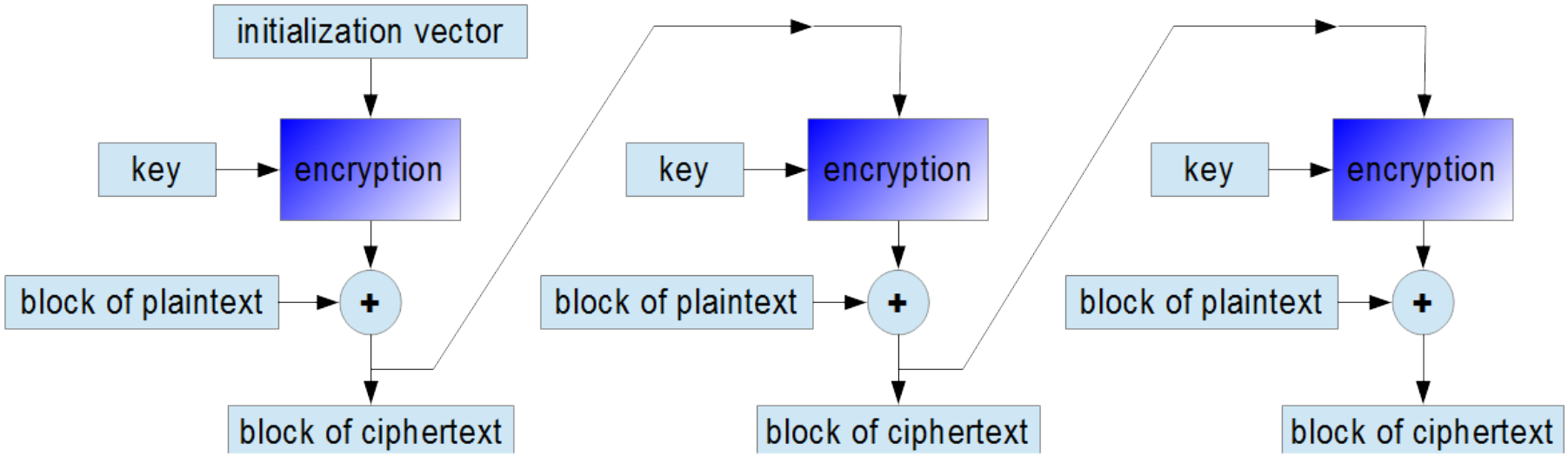


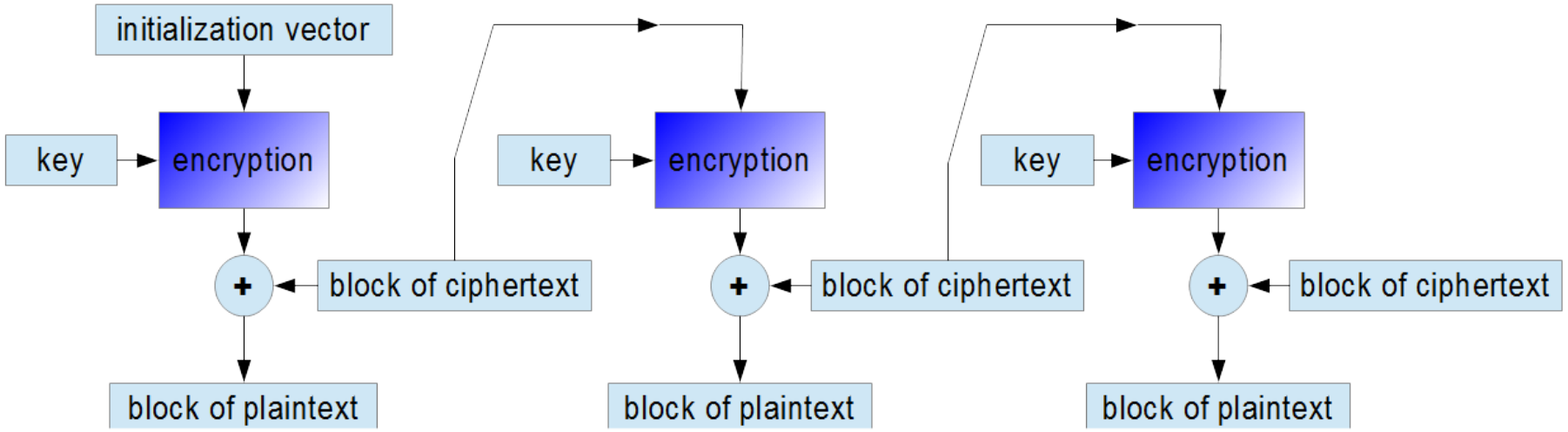
Figure 20.7 Cipher Block Chaining (CBC) Mode

Simplified Cipher Feedback Mode (CFB) Encryption



*Taken from http://www.crypto-it.net/eng/theory/modes-of-block-ciphers.html

Simplified Cipher Feedback Mode (CFB) Decryption



*Taken from http://www.crypto-it.net/eng/theory/modes-of-block-ciphers.html

Actual CFB

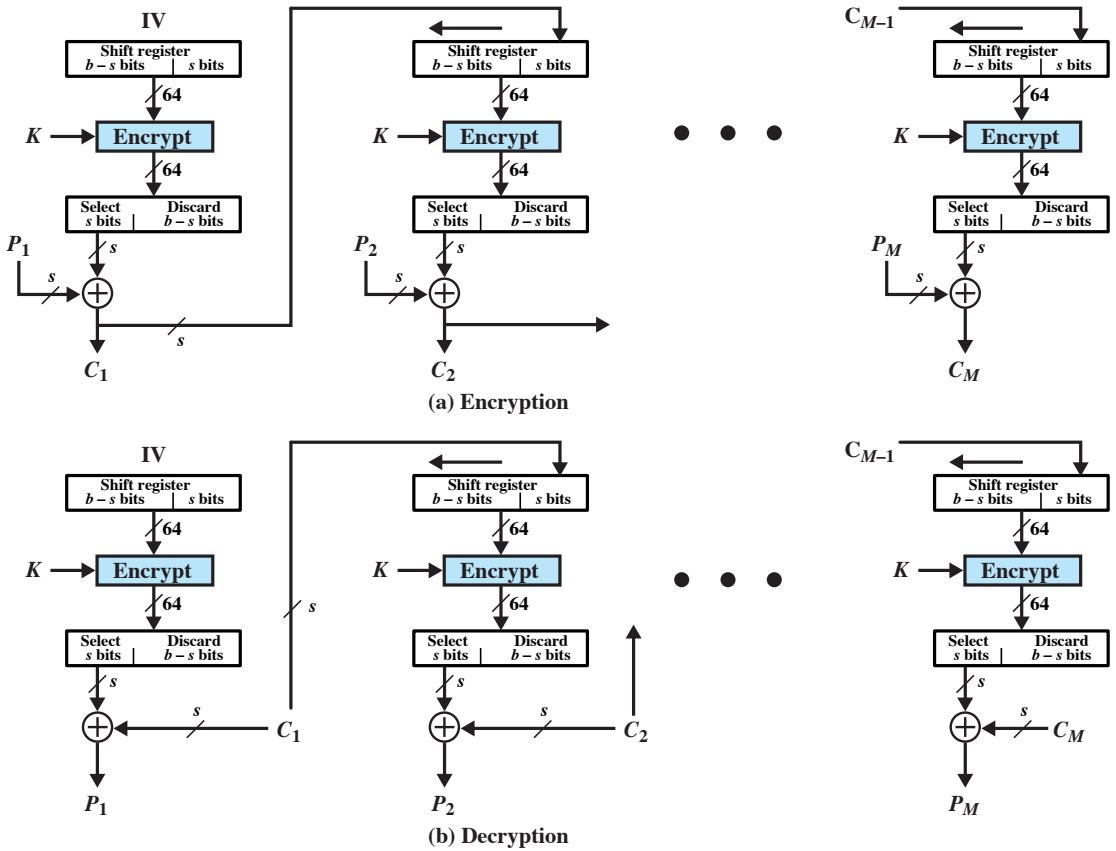


Figure 20.8 *s*-bit Cipher Feedback (CFB) Mode

Output Feedback Mode (OFB)

- Almost the same as CFB
- And that’s about all I got to say about that...

Counter Mode (CTR)

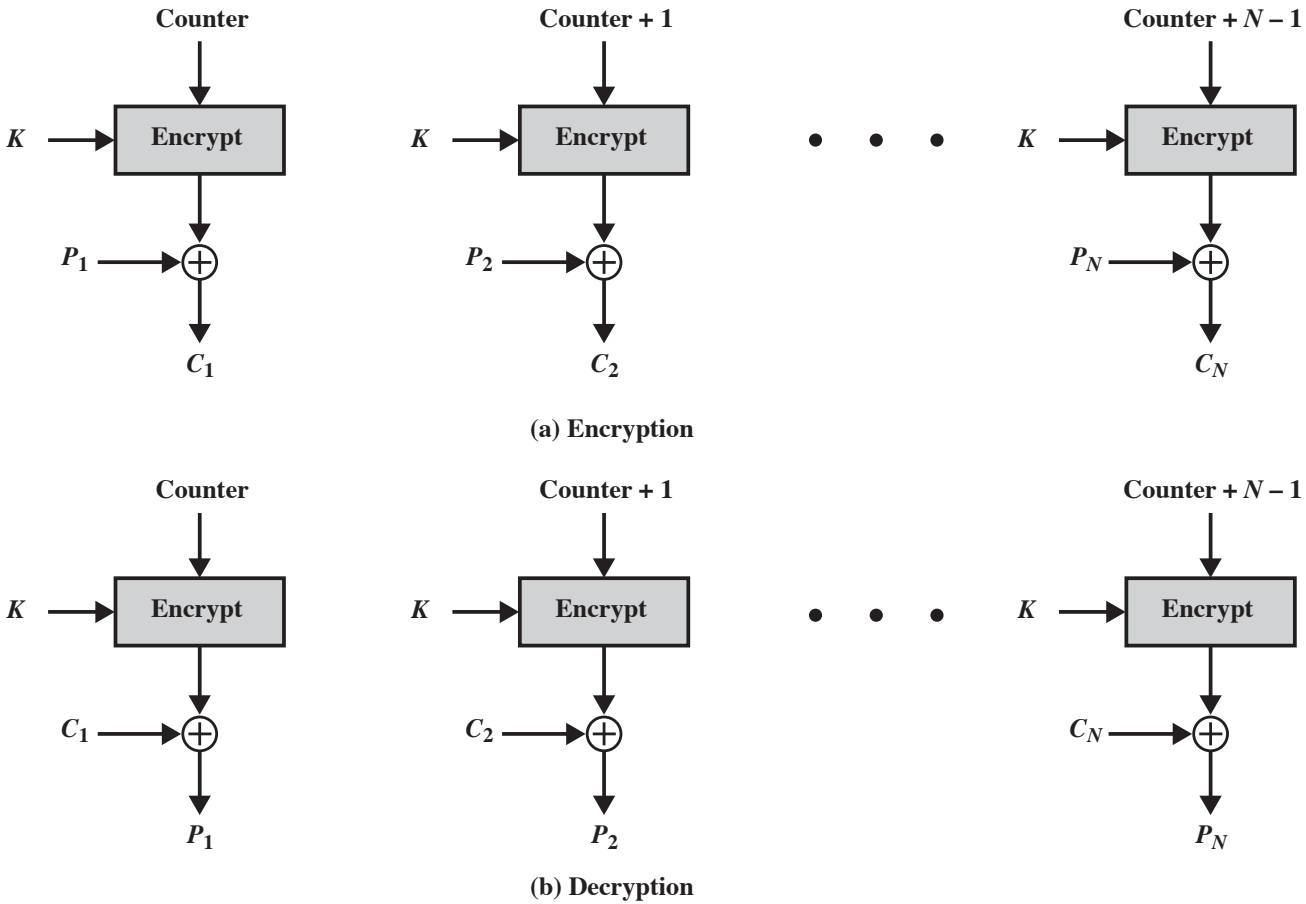


Figure 20.9 Counter (CTR) Mode

So Far...

- How's the CIA doing?
  - Confidentiality is pretty well in check
  - What about integrity?

Galosis/Counter Mode

- Similar to CTR mode
- Provides integrity as it's computed
- Not even going to look at the diagram