Lecturer: Dr. Michael J. May
Kinneret College

Block Cipher Modes

3 April 2025
Lecture 3

Slide Credits: Steve Zdancewic (UPenn)

Topics for Today

- Block Cipher Modes:
  - ECB
  - CBC
  - OFB
  - CTR
  - GCM
- Other ciphers and modes

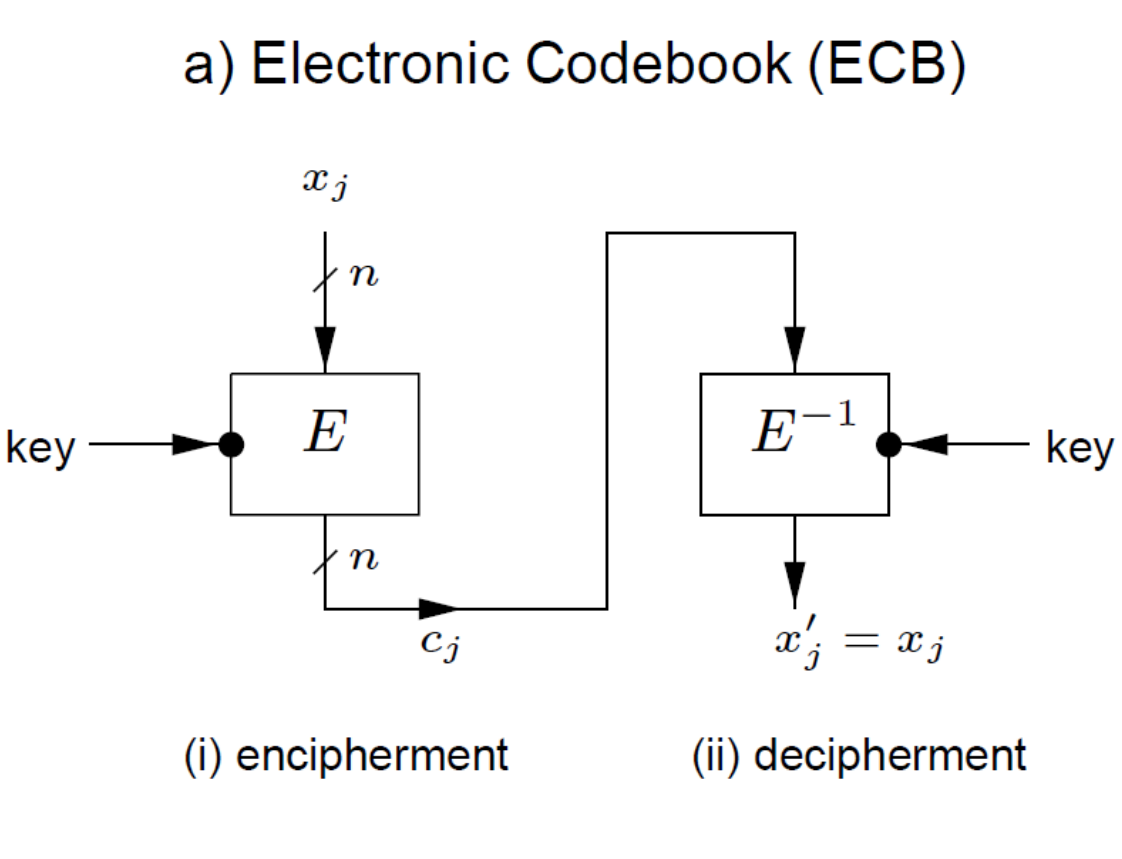
- Sources: HAC 7.2.2, 9.1-9.4, 12.6.1

Block Cipher Modes

What do we do with a block cipher of size *n* if the message size is greater than *n*?

Electronic Code Book (ECB)

- Simplest idea: Break the message into *n* bit blocks and encipher each one independently



ECB Properties

Identical plaintext blocks

- Under the same key result in identical ciphertext
- Preserves patterns in messages

No Chaining Dependencies

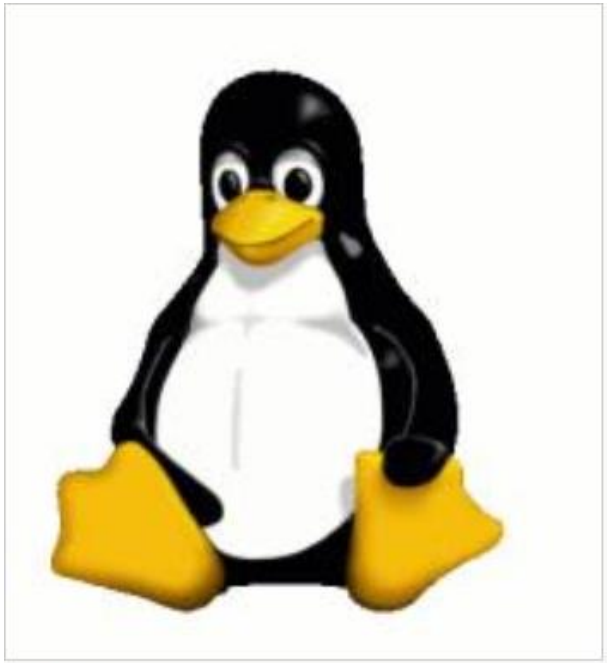
- Blocks are enciphered independently of all other blocks.
- Re-ordering ciphertext blocks results in corresponding re-ordered plaintext blocks

Error propagation

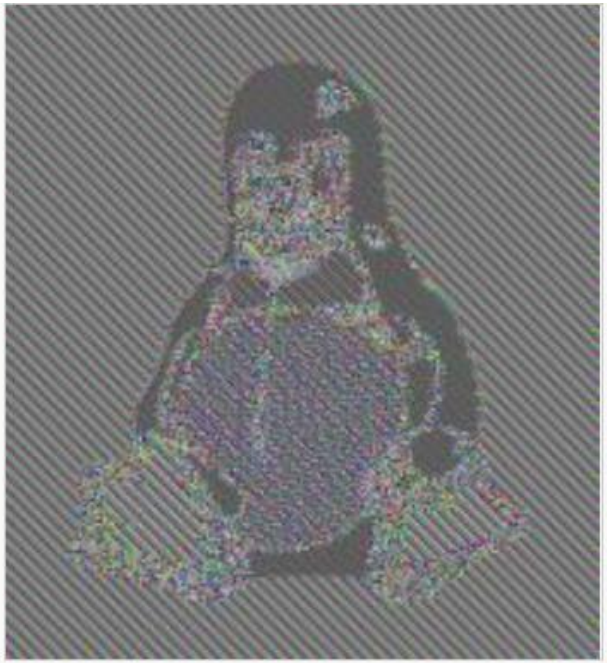
- One or more bit errors in a single ciphertext block affect decipherment of that block only

ECB Visualized

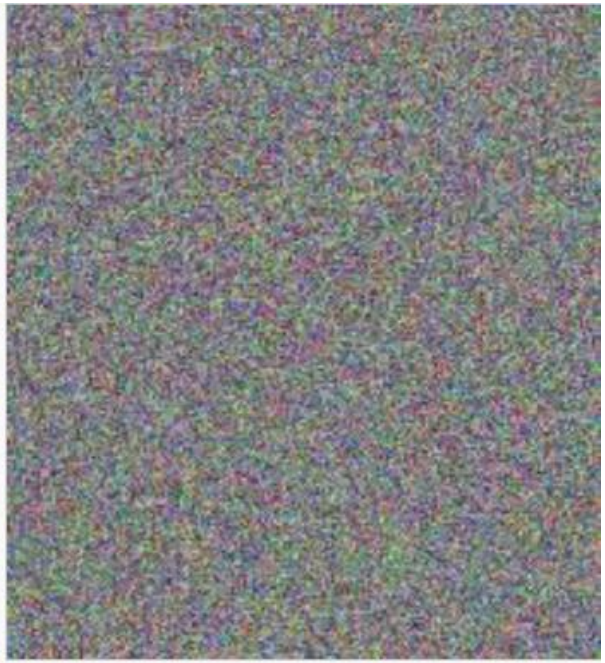
Image source: Wikipedia



Original image



Encrypted using ECB mode



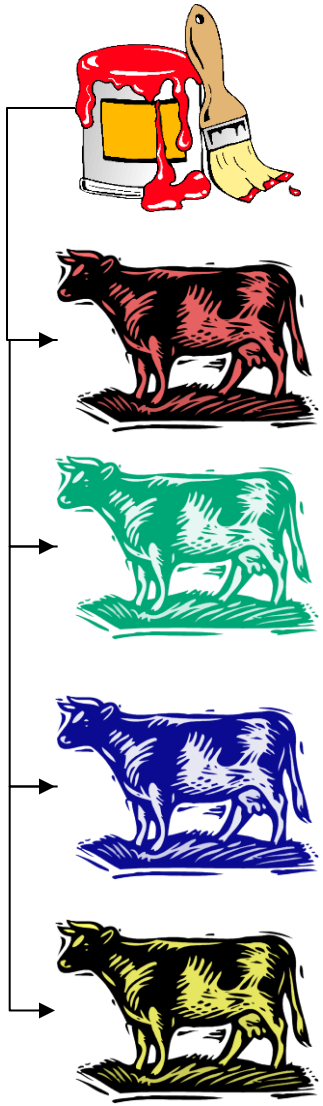
Modes other than ECB result in pseudo-randomness

So Far

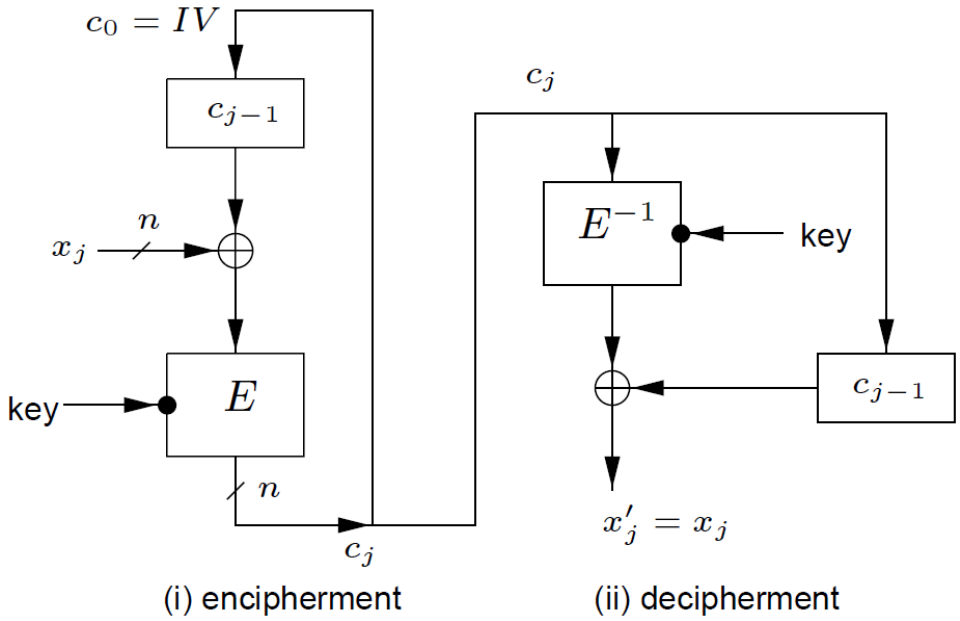
- Block Cipher Modes:
  - ECB
  - CBC
  - OFB
  - CTR
  - GCM
- Other ciphers and modes

Cipher Block Chaining (CBC)

- Chain each block based on the previous one
  - Introduce randomness to each block
  - Introduce dependencies in the message
- Use an Initialization Vector (IV) for the first block



b) Cipher-block Chaining (CBC)



(i) encipherment (ii) decipherment

CBC Properties

Identical plaintexts

- Identical ciphertext blocks result when the same plaintext is enciphered under the same key and *IV*

Chaining dependencies

- Chaining mechanism causes ciphertext c_j to depend on x_j and all preceding plaintext blocks
- Entire dependency on preceding blocks is, however, contained in the value of the previous ciphertext block.

CBC Properties

Error propagation

- A single bit error in ciphertext block c_j affects decipherment of blocks c_j and c_{j+1} (since x_j depends on c_j and c_{j-1}).
- Block x'_j recovered from c_j is typically totally random (50% in error)
- The recovered plaintext x'_{j+1} has bit errors precisely where c_j did.
- Thus an adversary may cause predictable bit changes in x_{j+1} by altering corresponding bits of c_j

Error recovery

- CBC mode is *self-synchronizing* in the sense that if an error (including loss of one or more entire blocks) occurs in block c_j but not c_{j+1}, c_{j+2} is correctly decrypted to x_{j+2}

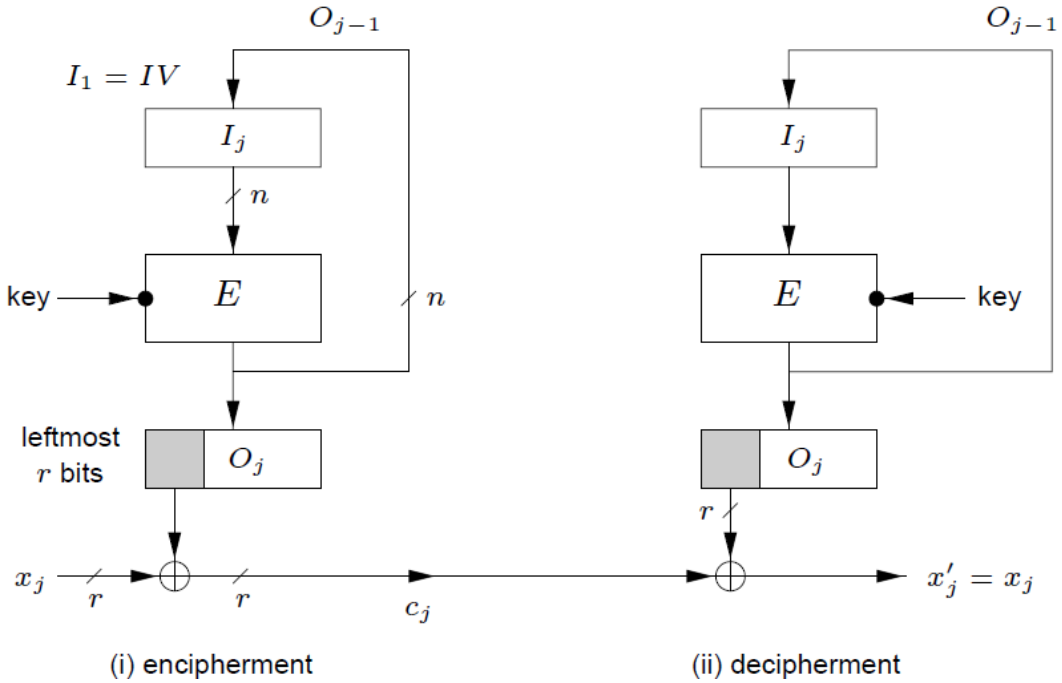
So Far

- Block Cipher Modes:
  - ECB
  - CBC
  - OFB
  - CTR
  - GCM
- Other ciphers and modes

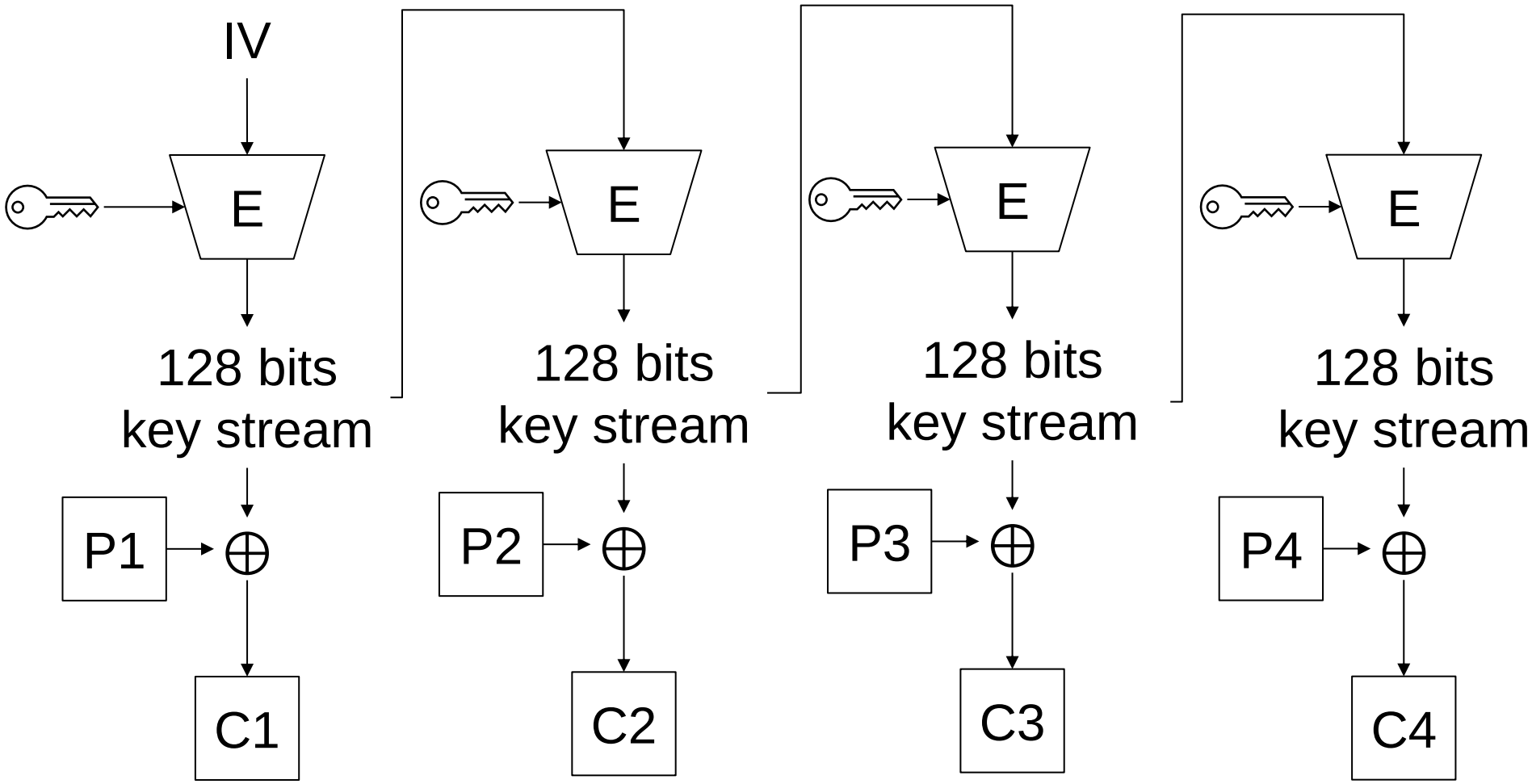
Output Feedback Mode (OFB)

- Make the encryption solely dependent on the IV and key
  - Remove all chaining dependencies
- Creates a stream cipher from a block cipher
  - No need for a separate decryption algorithm

d) Output feedback (OFB), r-bit characters/n-bit feedback



OFB Operations



OFB Properties

- **Identical plaintexts:** identical ciphertext blocks result when the same plaintext is enciphered under the same key and *IV.*
  - The IV must be changed if the key is to be reused.

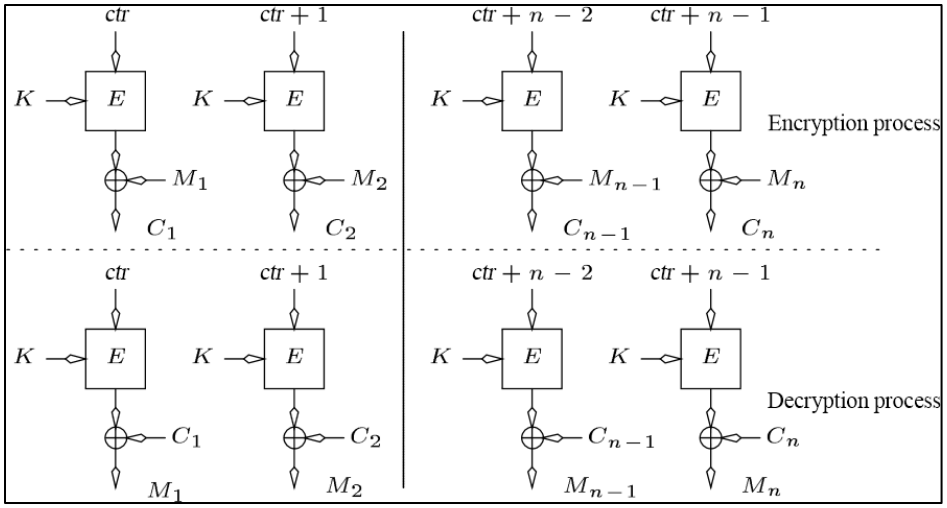
- **Error Propagation:** One or more bit errors in any ciphertext character c_j affects the decipherment of only that character in the precise bit position(s) c_j is in error, causing the corresponding plaintext bit(s) to be complemented.

- **Error recovery:** OFB recovers from ciphertext bit errors, but can't self-synchronize after loss of ciphertext bits, which destroys alignment

- **Throughput:** Since keystream is independent of plaintext or ciphertext, it may be pre-computed.

Counter Mode (CTR)

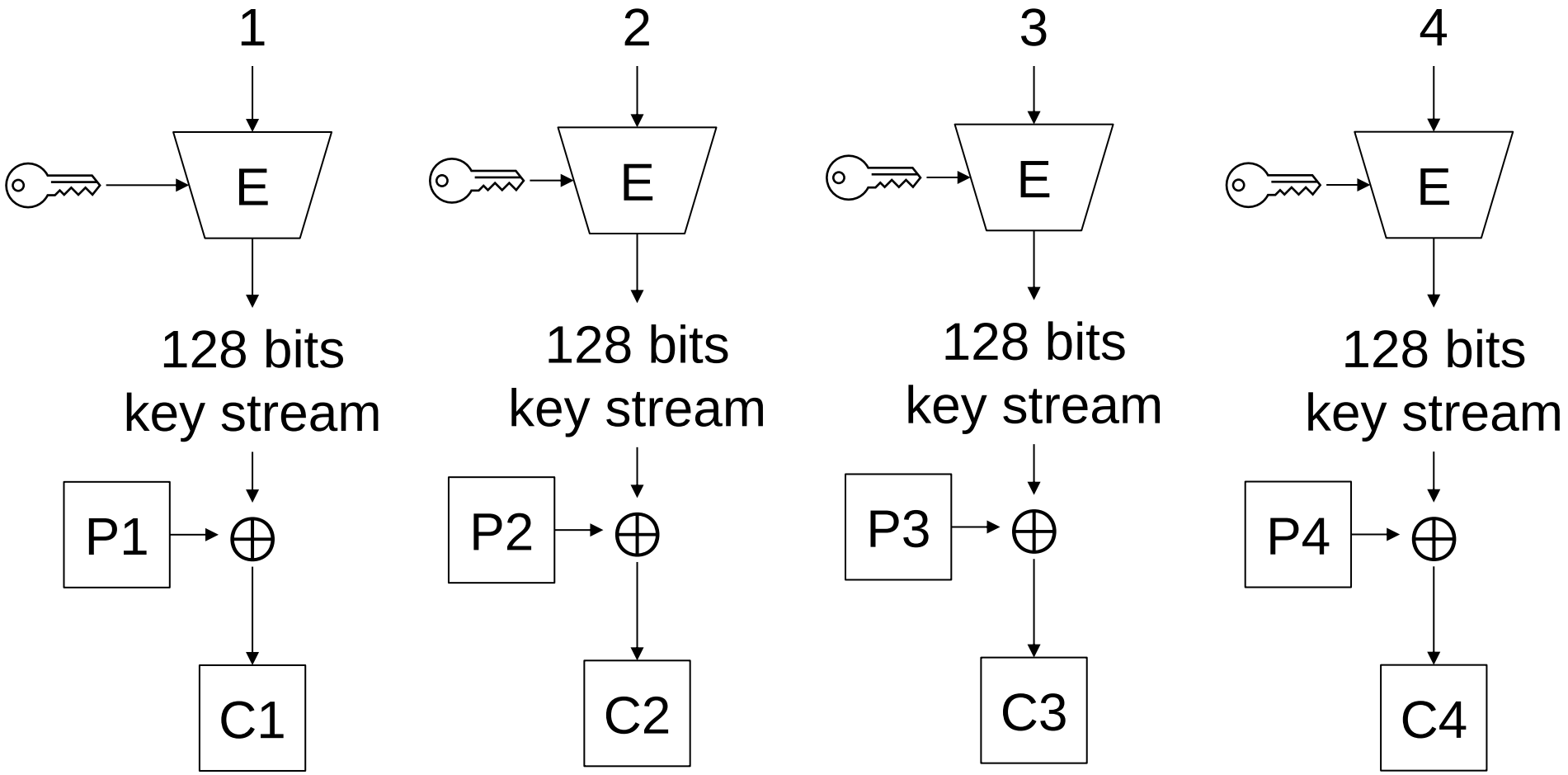
- A simplification of OFB in which $IV = 0$ and the input blocks $I_{j+1} = I_j + 1$ rather than using feedback



Properties:

- Avoids problem of repeating IV (if encrypting IV many times eventually leads to it recurring)
- Allows random access decryption
  - Ciphertext block i need not be decrypted to decrypt block $i + 1$

CTR Operations

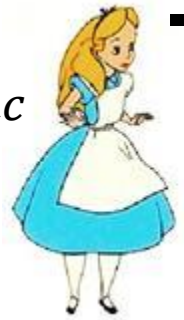


A look ahead

- Modern cipher modes combine encryption with authentication:
  - *Authenticated Encryption with Associated Data (AEAD)*

- Later we'll talk about Message Authentication Codes (MAC) in general
  - *HMAC*
  - *Encrypt and then MAC*
  - *Galois Counter Mode (GCM)*

Using GCM



K_{Enc}

c, h, t, IV

K_{Enc}

GMAC mode: msg is empty and h isn't empty → computes a MAC on h

1. Wants to send msg with secrecy and integrity
2. Wants to send h with integrity only
3. Sets Additional Authentication Data $AAD = h$
4. Chooses tag length len (128 bits)
5. Chooses **unique** IV (96 bits)
6. GCM Encrypts $(c, t) = E_{K_{Enc},AAD,IV}\{msg\}$

   GCM gives two outputs (t is len bits)

7. GCM Decrypts

   $p = D_{K_{Enc},h,t,IV}\{c\}$

8. If D doesn't return `FAIL`:

   $p == msg$ and unchanged

   h is unchanged

GCM Notes (from NIST)

Uniqueness of IV is critical
Using same IV twice with same key leads to compromise

- Using 96 bit IV is recommended, longer or shorter ones are hashed

Short tags are bad: 128 bits is recommended.

- 32 bit tags can only be used for tens of bytes per key
- 64 bit tags can only be used for millions of bytes per key (few MBs)

Don't use a key more than 2^{32} times no matter what

GCM can encrypt up to 64 GB per message securely

Other Ciphers and Modes

Block Ciphers

- Speck (IoT)
- Simon (IoT)
- CAST-256
- Camellia

- Other Modes:
  - Cipher Feedback (CFB)
  - CCM (Counter with CBC-MAC)
  - Ciphertext Stealing (CTS)

Stream Ciphers

- ChaCha20 / Salsa20
- HC-256

Conclusion

- Block Cipher Modes:
  - ECB
  - CBC
  - OFB
  - CTR
  - GCM
- Other ciphers and modes