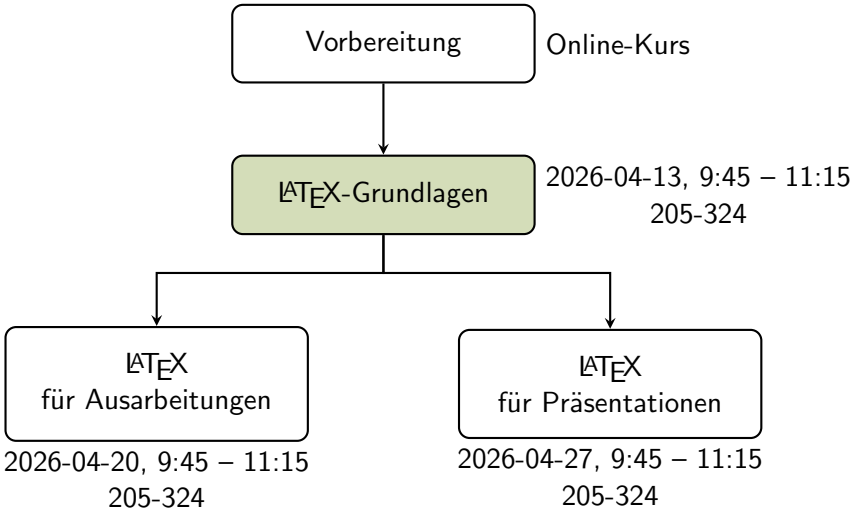
Hochschule
Albstadt-Sigmaringen
Albstadt-Sigmaringen University

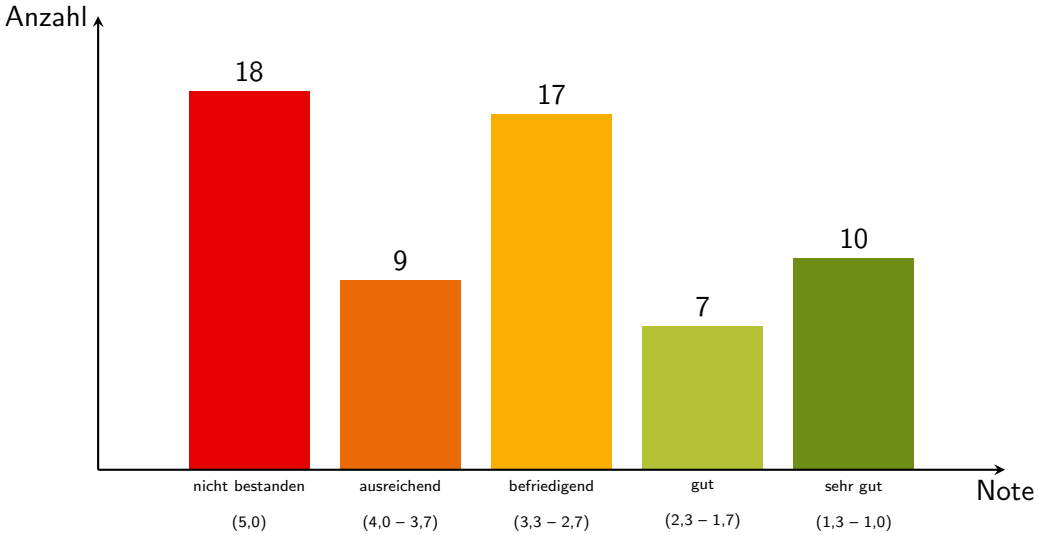
# Einführung IT Security

LaTeX


Vorbereitung
Online-Kurs
LaTeX-Grundlagen
2026-04-13, 9:45 – 11:15
205-324
LaTeX
für Ausarbeitungen
2026-04-20, 9:45 – 11:15
205-324
LaTeX
für Präsentationen
2026-04-27, 9:45 – 11:15
205-324

# EITS Klausur

# Klausurergebnisse WiSe 2025/26



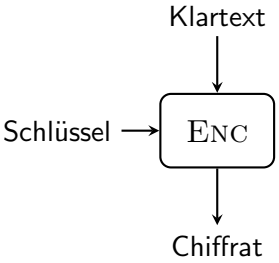
# Verschlüsselungsverfahren

# Moderne Kryptologie

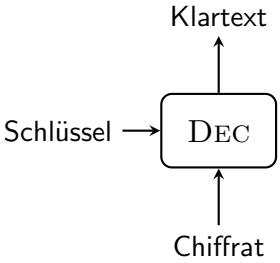
Hochschule Albstadt-Sigmaringen
Albstadt-Sigmaringen University

| | **symmetrische Kryptologie** | **asymmetrische Kryptologie** |
|---|---|---|
| **Vertraulichkeit** | symmetrische Verschlüsselung | asymmetrische Verschlüsselung |
| **Integrität** | MAC | Signaturen |
| **Authentizität** | | PKI |
| **Schlüsselaustausch** | symmetrischer Schlüsselaustausch | asymmetrischer Schlüsselaustausch |
| **Kryptoanalyse** | symmetrische Kryptoanalyse | Mathematik |
| | Implementierungsangriffe | |
| | Seitenkanalanalyse | |
| **Bausteine** | Hashfunktionen | |
| | ⋮ | ⋮ |

## Verschlüsselung

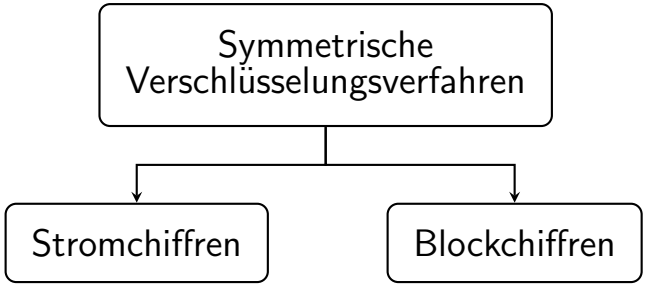



## Entschlüsselung

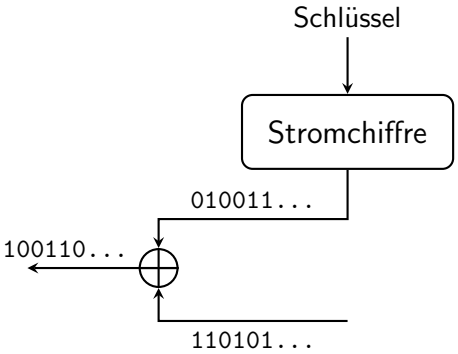



**Korrektheit:** $\mathrm{DEC}(\mathrm{ENC}(m)) = m$

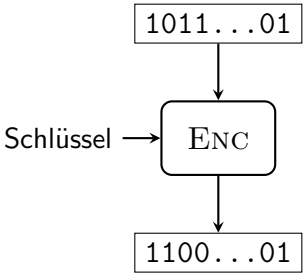
# Symmetrische Verschlüsselungsverfahren



# Symmetrische Verschlüsselungsverfahren

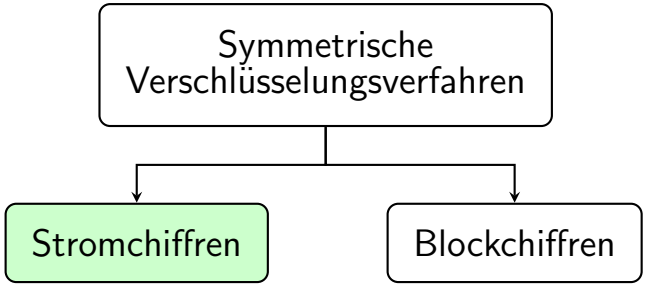



- Pseudozufallszahlengenerator
- erzeugt Schlüsselstrom
- Verknüpfung mit Klartextstrom mit XOR
- Ergebnis ist Chiffratstrom




## Blockchiffre

- zufällige Permutation
  (*random permutation*)
- Eingabe ist Klartextblock fester Länge
- Ausgabe ist Chiffratblock gleicher Länge


Symmetrische
Verschlüsselungsverfahren
Stromchiffren
Blockchiffren

XOR - exklusives Oder

| $\oplus$ | 0 | 1 |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 1 | 0 |

reversible Verknüpfung:

$b \oplus k = c$

$c \oplus k = b$

XOR - exklusives Oder

| $\oplus$ | 0 | 1 |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 1 | 0 |

reversible Verknüpfung:

$b \oplus k = c$

$c \oplus k = b$

$b \oplus (k \oplus k) = b$

$0$

$(b \oplus k) \oplus k$

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | = 0 |
| 1 | 0 | 0 | = 1 |
| 0 | 1 | 1 | = 0 |
| 1 | 1 | 1 | = 1 |

XOR - exklusives Oder

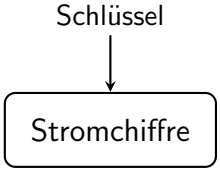
| $\oplus$ | 0 | 1 |
|---|---|---|
| 0 | 0 | 1 |
| 1 | 1 | 0 |

reversible Verknüpfung:

$b \oplus k = c$

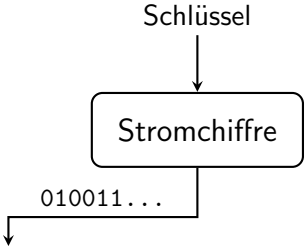
$c \oplus k = b$

Korrektheit: $(b \oplus k) \oplus k = b$



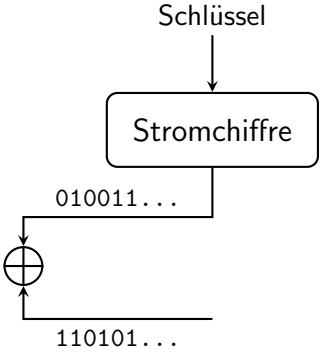

## Stromchiffre

- Pseudozufallszahlengenerator




## Stromchiffre

- Pseudozufallszahlengenerator
- erzeugt Schlüsselstrom



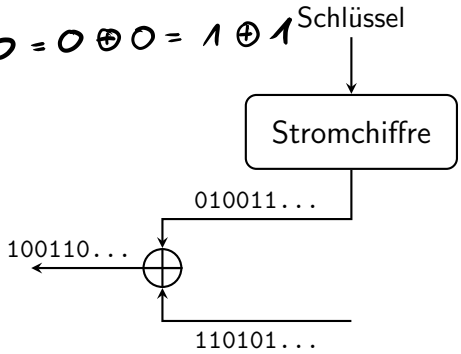

## Stromchiffre

- Pseudozufallszahlengenerator
- erzeugt Schlüsselstrom
- Verknüpfung mit Klartextstrom mit XOR

c b k b k

1 = 1 ⊕ 0 = 0 ⊕ 1

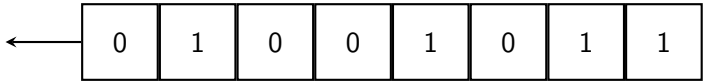
0 = 0 ⊕ 0 = 1 ⊕ 1



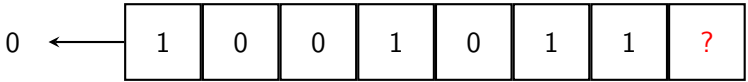

## Stromchiffre

- Pseudozufallszahlengenerator
- erzeugt Schlüsselstrom
- Verknüpfung mit Klartextstrom mit XOR
- Ergebnis ist Chiffratstrom

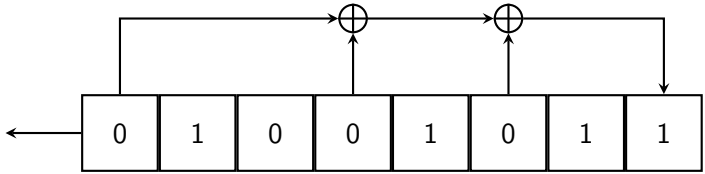
# Linear Rückgekoppelte Schieberegister



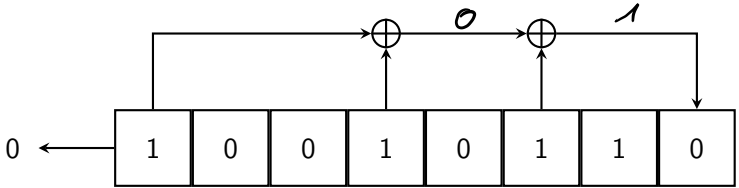
# Linear Rückgekoppelte Schieberegister



# Linear Rückgekoppelte Schieberegister

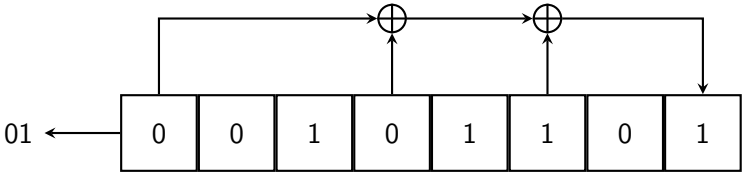


# Linear Rückgekoppelte Schieberegister



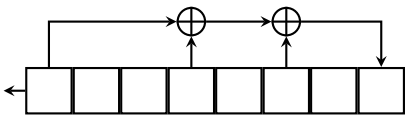

LFSR: *linear-feedback shift register*

# Linear Rückgekoppelte Schieberegister




LFSR: *linear-feedback shift register*

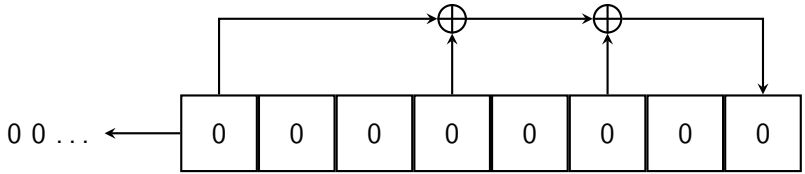
# Periode von LFSR



Wie viele verschiedene Zustände kann ein LFSR der Länge $n$ haben?

$2^n$ interne Zustände sind möglich, aber...

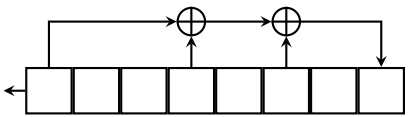
# Linear Rückgekoppelte Schieberegister



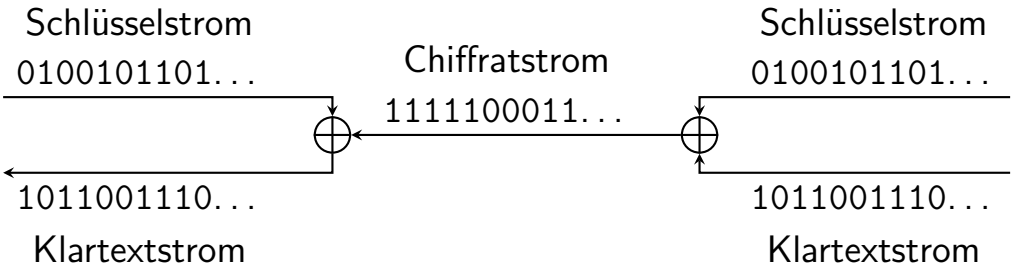

nur Nullen $\rightarrow$ LFSR bleibt im Null-Zustand

Chiffre: Rückkopplung

Schlüssel: Anfangsbelegung



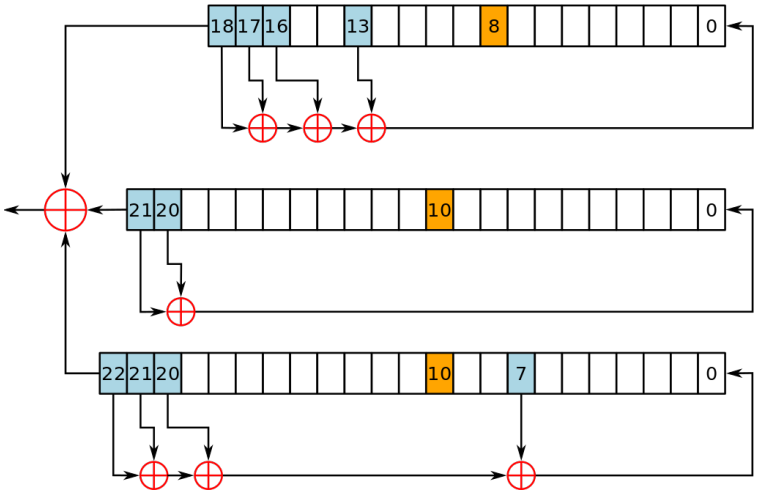
Bei "'guter"' Wahl der Rückkopplung hat ein LFSR der Länge $n$ eine Persiode der Länge $2^n - 1$.


Schlüsselstrom
0100101101...
Chiffratstrom
1111100011...
Schlüsselstrom
0100101101...
1011001110...
Klartextstrom
1011001110...
Klartextstrom

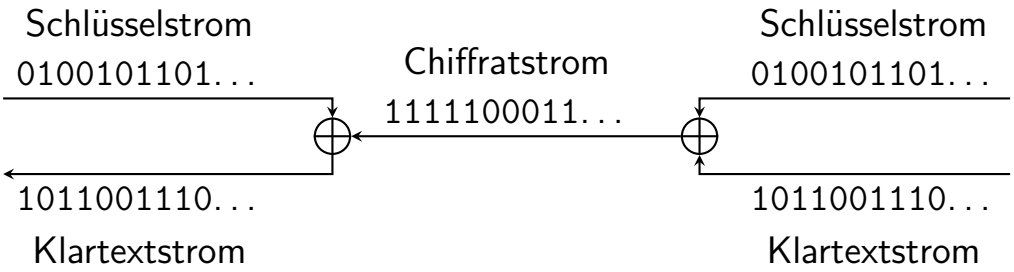

bekannter Klartext $\rightarrow$ Schlüsselstrom

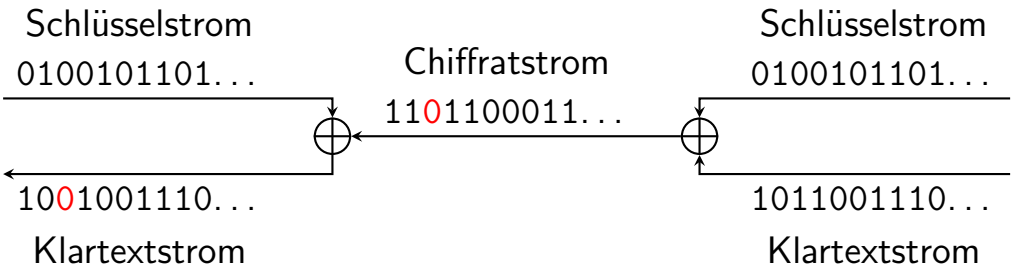
LFSR-Stromchiffren kombinieren mehrere LSFR

# LFSR-Stromchiffre A5/1

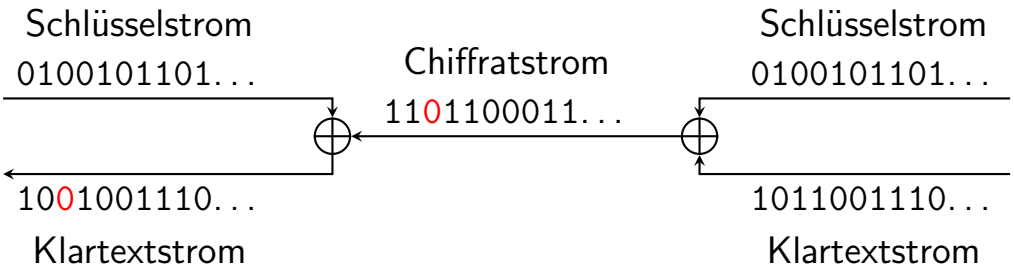



User:Matt Cryptoderivative work: Tsaitgaist (talk) - A5-1.png, Gemeinfrei


Schlüsselstrom
0100101101...
Chiffratstrom
1111100011...
Schlüsselstrom
0100101101...
1011001110...
Klartextstrom
1011001110...
Klartextstrom




Übertragungsfehler




*bitflip* im Chiffratstrom $\rightarrow$ *bitflip* im entschlüsselten Klartextstrom an der gleichen Position

Vorteile

- beliebige Klartextlänge
- Auswirkungen von Übertragungsfehlern gering
- (schnell & günstig)

## Vorteile

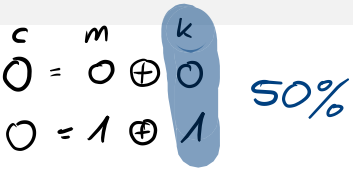
- beliebige Klartextlänge
- Auswirkungen von Übertragungsfehlern gering
- (schnell & günstig)

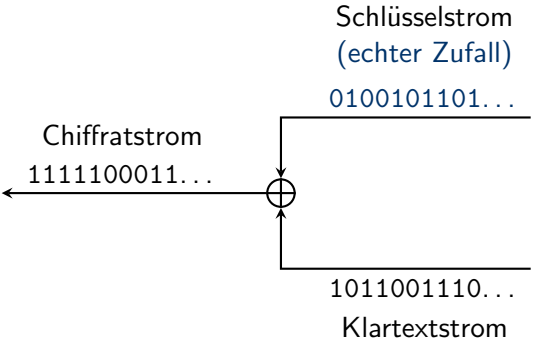
## Nachteile

- bekannter Klartext → Schlüsselstrom
  → Kryptoanalyse
  einfache LFSR sind unsicher
- Manipulation des Chiffrats möglich
- ältere Stromchiffren oft unsicher

LFSR-Chiffren

# One-Time-Pad




- Verschlüsselung mit echtem Zufall
- informationstheoretisch sicher

# One-Time-Pad



Perfekte Sicherheit...

Zu jedem Chiffrat gibt es für jeden möglichen Klartext einen Schlüssel, der das Chiffrat zu diesem Klartext entschlüsselt.

Die Verwendung eines One-Time-Pad ist keine Garantie für perfekte Sicherheit in der Praxis, sondern bestenfalls eine Voraussetzung.

# One-Time-Pad

Hochschule Albstadt-Sigmaringen
Albstadt-Sigmaringen University

Perfekte Sicherheit...

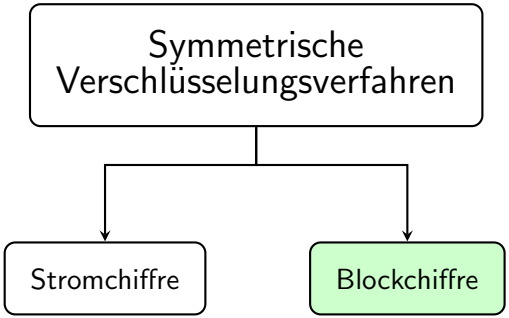
Zu jedem Chiffrat gibt es für jeden möglichen Klartext einen Schlüssel, der das Chiffrat zu diesem Klartext entschlüsselt.

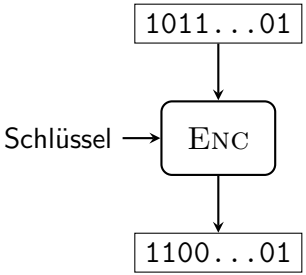
...aber

- Verschlüsselung mit *echtem* Zufall
- Schlüssel so lang wie die Nachricht
- Schlüssel nur einmal nutzbar
- kein Schutz gegen Manipulation (nur Vertraulichkeit, keine Integrität)
- Schlüssel muss sicher ausgetauscht werden
  Quantenschlüsselaustausch (QKD) will dies lösen

Die Verwendung eines One-Time-Pad ist keine Garantie für perfekte Sicherheit in der Praxis, sondern bestenfalls eine Voraussetzung.

# Symmetrische Verschlüsselungsverfahren






## Blockchiffre

- zufällige Permutation
  (*random permutation*)
- Eingabe ist Klartextblock fester Länge
- Ausgabe ist Chiffratblock gleicher Länge

# Blockchiffren



26 Buchstaben
→ 26! Permutationen

$$
\begin{array}{ccc}
A & \leftrightarrow & a \\
B & \leftrightarrow & b \\
\vdots & & \vdots \\
F & \leftrightarrow & f \\
G & \leftrightarrow & h \\
H & \leftrightarrow & g \\
\vdots & & \vdots \\
Z & \leftrightarrow & z
\end{array}
$$

Blocklänge $n$

# Blockchiffren

26 Buchstaben
→ 26! Permutationen

| | | |
|---|---|---|
| A | ↔ | a |
| B | ↔ | b |
| ⋮ | | ⋮ |
| F | ↔ | f |
| G | ↔ | h |
| H | ↔ | g |
| ⋮ | | ⋮ |
| Z | ↔ | z |

n-Bit-Blöcke

Blocklänge $n$
→ $2^n!$ verschiedene Permutationen

# Blockchiffren

26 Buchstaben
→ 26! Permutationen

| | | |
|---|---|---|
| A | ↔ | a |
| B | ↔ | b |
| ⋮ | | ⋮ |
| F | ↔ | f |
| G | ↔ | h |
| H | ↔ | g |
| ⋮ | | ⋮ |
| Z | ↔ | z |

Blocklänge $n$
→ $2^n$! verschiedene Permutationen

| | | |
|---|---|---|
| 0000 | ↔ | 0000 |
| 0001 | ↔ | 0001 |
| ⋮ | | ⋮ |
| 1001 | ↔ | 1010 |
| 1010 | ↔ | 1001 |
| ⋮ | | ⋮ |
| 1111 | ↔ | 1111 |

# Blockchiffren



gewünschte Eigenschaften

- schnell & effizient
- kurze Schlüssel
- nur ”‘gute”’ Permutationen


m
10111001
Schlüssel
DEC
c
11000101
11000101
m
Schlüssel
ENC
11000101
c


11100101
10111001
Schlüssel
DEC
ENC
Schlüssel
11010101
11000101
Übertragungsfehler

Vorteile

- Rückschlüsse auf Schlüssel schwer

# Eigenschaften Blockchiffren



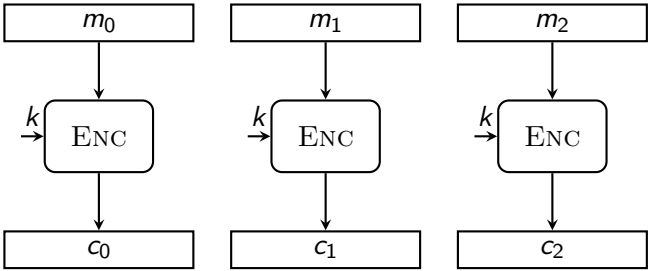
Vorteile

- Rückschlüsse auf Schlüssel schwer

Nachteile

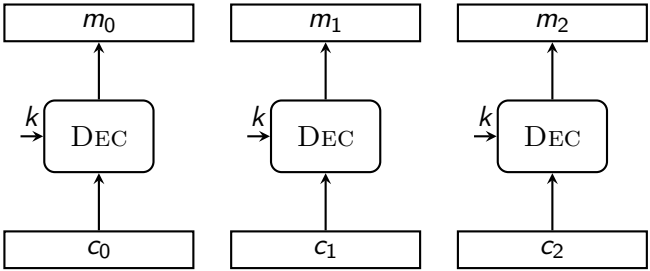
- Ver- und Entschlüsselung:
  nur ganze Blöcke
- Übertragungsfehler "'zerstören"'
  Block

## Electronic Codebook Mode – ECB




Verschlüsselung

## Electronic Codebook Mode – ECB




Entschlüsselung