Dr.-Ing. Wilfried Dankmeier
HS Aschaffenburg

Codierung

Fehlerbeseitigung	Verschlüsselung	Kompression
Speicher (RAM, Platten, CD) Mobilfunk, Datenübertragung	Geheimhaltung von Dokumenten, elektr. Unterschrift, Datenintegrität	ZIP, JPEG, MP3

Beispiel mit ASCII - Infobyte „**B**“ ≡ **0100 0010** (binäre Darstellung) ≡ **42** (hexadezimale Darstellung)

„B“ wurde gesendet, ein Bit aber während der Übertragung gekippt:

„B“ 01010010 ⇒ Empfänger sieht „**R**“

Fehlererkennung durch Anhängen eines Prüfbits auf gerade Anzahl von „1Bits“:

01000010 **0**

bei Störung erhält Empfänger:

01010010 **0**

und erkennt durch Vergleich mit Prüfbit, dass ein Fehler auftrat (ungerade Zahl von 1Bits)

Preis: Prüfbits vergrößern Datenvolumen

⇒ Redundanzerhöhung

Ziel:
mit wenig Aufwand viele Fehler erkennen, oder sogar korrigieren

(Hamming-, BCH-, Reed-Solomon-, Turboprodukt Code)

Verschlüsselung:

Klartext „B“	**0 1 0 0 0 0 1 0**
Schlüssel „*“	+ 0 0 1 0 1 0 1 0
Zwischenergebnis	0 1 1 0 1 0 2 0

MOD 2

⇒ **Geheimtext** **0 1 1 0 1 0 0 0**

Entschlüsselung:

Geheimtext	**0 1 1 0 1 0 0 0**
Schlüssel „*“	- 0 0 1 0 1 0 1 0
Zwischenergebnis	0 1 0 0 0 0-1 0

MOD 2

⇒ **Klartext** **0 1 0 0 0 0 1 0**

Ziel:
mit wenig Aufwand einen für Unbefugte unkenntlichen Geheimtext erzeugen

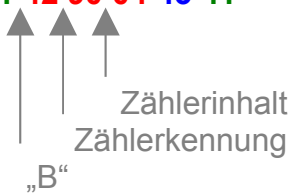
(DES, RSA, Fiat-Shamir, Diffie-Hellman)

Zu übertragen ist die Zeichenfolge:

ABBBBCA ≡ 41 42 42 42 42 43 41
(7 Zeichen)

Kompression ohne Informationsverlust durch:

41 42 90 04 43 41
(6 Zeichen)



komprimiert um 1/7 auf 6/7 ≈ 0,86
Sonderfall Zählerkennung „90“ tritt selbst als Zeichen auf: 90 00 ⇒ Zählerinhalt ist 0

Ziel:
möglichst viele Bits einsparen, ohne dass Information verloren geht oder verändert wird

(Variante für Grafik- und Akustikdaten: „kleine“ Verluste zulassen, solange Bild oder Musik erkennbar bleiben)