1. Aufgabe: Kodierungsbegriff

a. Erläutern Sie anhand von Abb. 4.1 aus dem KTII-Skript die Begriffe Quellkodierung, Kanalkodierung und Leitungskodierung und nennen Sie jeweils Beispiele dafür.

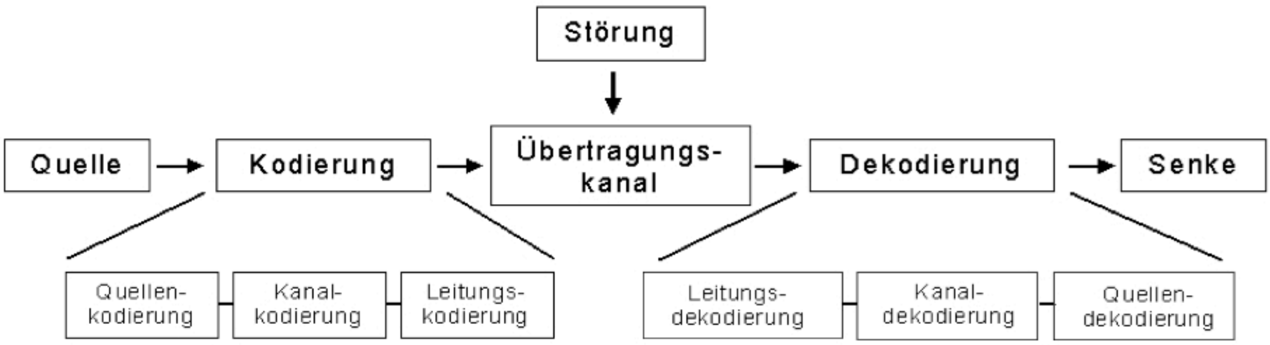

Abb. 1: Technisches Kommunikationsmodell (aus dem KTII-Skript)

- Die **Quellkodierung** hat die Aufgabe, die zu sendende Information zu komprimieren, also die Datenmenge, die zur Übertragung oder Speicherung benötigt wird zu verringern. Die **Quelldekodierung** wandelt das komprimierte Signal wieder in das Ausgangssignal zurück. Beispiele für bitratenreduzierende Verfahren sind die Huffman-Kodierung und FLAC (Redundanzkodierung) oder MPEG-1 Layer 3 (MP3, Irrelevanzkodierung).
- Bei der **Kanalkodierung** werden dem Signal zusätzliche Informationen hinzugefügt - die Datenmenge wird wieder erhöht. Das macht es möglich durch die Übertragung entstandene Fehler ggf. bei der **Kanaldekodierung** zu erkennen und zu korrigieren. Beispiele sind Paritätsbits oder das Interleaving von Datenblöcken.
- Die **Leitungskodierung** bereitet die in digitaler Form vorliegenden Signale so auf, dass sie analog übertragen werden können. Dafür werden die meist binären Daten in z.B. unterschiedliche Spannungs- oder Helligkeitswerte umgewandelt. Die **Leitungsdekodierung** nimmt dann eine A/D-Wandlung der Signale vor. Beispiele für Leitungskodes sind RZ (Return to zero) und NRZ (Non return to zero).

2. Aufgabe: Kanal- und Leitungskodierung

Gegeben sei die Bitfolge von 101100111000

a. Skizzieren Sie den Spannungsverlauf dieses Signals. Sind die Codes selbstaktend?
   i) in NRZ-Kodierung, ii) in NRZI-Kodierung, iii) im Biphase Mark-Kode

i) NRZ (non return to zero): bildet 1 auf hohes, 0 auf niedriges Potential ab
ii) NRZI (NRZ-inverted): Potentialwechsel in der Mitte einer Bitperiode für 1 (in jede Richtung), kein Potentialwechsel für 0
iii) Biphase Mark/FM (Frequency Modulation): Potentialwechsel am Anfang und in der Mitte einer Bitperiode für 1, Potentialwechsel am Anfang einer Bitperiode für 0

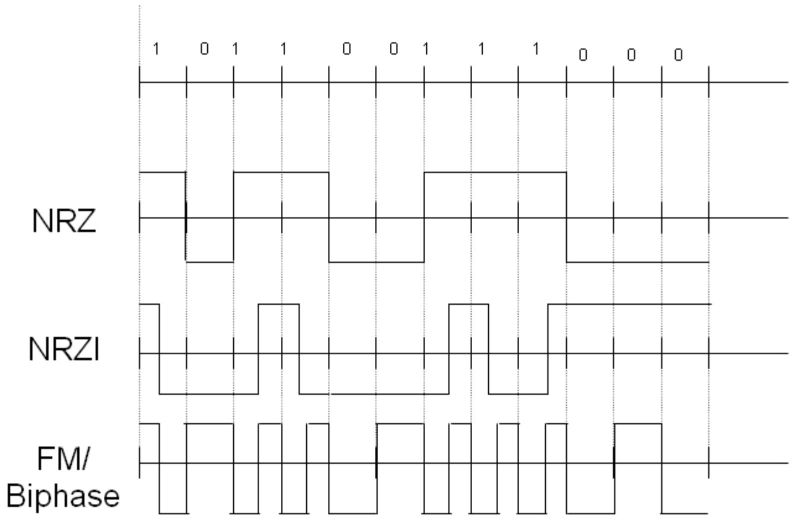

Abb. 2: Leitungskodes

Der NRZ und NRZI Kode sind nicht selbsttaktend, da nicht festgelegt ist, wie viel Zeit maximal zwischen zwei Potentialwechseln vergehen kann. Der Biphase Mark/FM Kode ist selbsttaktend weil es mindestens einen Potentialwechsel je übertragenem Symbol gibt.

b. Konstruieren Sie eine eigene Kodetabelle für einen 3/5-Gruppenkode mit einer (0,2) RLL Lauflängenkodierung (min/max Anzahl der 0en zwischen zwei 1en).

Der zu erstellende Kode muss Datenworte von m = 3 bit Länge auf speziell ausgewählte Kanalkodeworte der Länge n = 5 abbilden.
Die 2^3 = 8 möglichen Datenworte umfassen das binäre Intervall von [000;111]. Die (0,2)-RLL Kodierung verlangt eine minimale Anzahl von d = 0 und eine maximale Anzahl von k = 2 Nullen zwischen zwei aufeinander folgenden „1“-Symbolen (ist auch bei aufeinander folgenden Kodeworten einzuhalten).

Unter Berücksichtigung dieser Bedingungen lassen sich den Datenworten entsprechende Kanalkodeworte zuweisen. Die Tabelle zeigt für alle Datenworte die ihnen zuzuweisenden Kanalkodeworte. Dabei handelt es sich um eine rein willkürliche (!) Auswahl und Zuweisung.

Datenwort	Kanalkodewort
000	01110
001	01101
010	01011

011	10011
100	10101
101	10110
110	11010
111	01111

c. Worin liegt der Vorteil von Run-Length-Limited Gruppenkodes?

Die Kanalbitrate erhöht sich gegenüber dem Quellkode durch die Gruppenkodierung zwar um den Faktor n/m (in diesem Fall 5/3), dies kann aber durch eine geeignete Wahl von (d,k) ausgeglichen werden. Außerdem kann eine Optimierung unter dem Gesichtspunkt höherer Fehlersicherheit erzielt werden, indem für Kanalkodeworte der minimaler Kodewortabstand $d_{\min}$ (Hammingabstand = Anzahl der unterschiedlichen Bitstellen je Kodewort, vgl. Skript) maximiert wird.

3. Aufgabe: Fehlerkorrektur I

Gegeben sei folgender Kode, bestehend aus vier Kodewörtern:

10100   01000   10011   01111

a. Um was für einen Kode handelt es sich?

Es ist ein 2/5 Gruppenkode mit einer (0,4)-RLL Kodierung.

b. Wie viele Bit-Fehler können mit dem Kode erkannt bzw. korrigiert werden?

Die Anzahl der Fehlerhaften bits die korrigiert bzw. erkannt werden können, hängt vom minimalen Hammingabstand ab. Für binäre Kodeworte ist er durch die Modulo 2 Addition der Bits zweier Kodewörter gegeben:

$$d(a_i, a_j) = \sum_{g=1}^{n} (u_{ig} \oplus u_{jg})$$

Der minimale Abstand für den gegebenen Kode ist $d_{\min}$ =3. Es können also

$$f_e = d_{\min} - 1 = 2$$

fehlerhafte Bits erkannt und

$$f_k = \frac{d_{\min} - 1}{2} = 1$$

fehlerhafte Bits korrigiert werden.

4. Aufgabe: Fehlerkorrektur II

Audiosymbole mit einer Länge von 8 bit werden mit einem Paritätsbit zur Fehlererkennung kodiert.

a. Wird bei einer Paritätsprüfung das empfangene Kodewort 100100101 als fehlerfrei klassifiziert?

Je nach Paritäts-Typ würde dieses Wort als gültig oder ungültig erkannt werden. Bei der häufiger anzutreffenden „even parity“ wird das angefügte Prüfbit so gewählt, dass die Anzahl der Einsen im Gesamtwort (9-bit) gerade ist. Für diesen Fall wäre unser Kodewort gültig, auch wenn wir damit die (geringe) Restfehlerwahrscheinlichkeit, dass genau 2, 4, 6 oder gar 8 Bit fehlerhaft sind, außer Acht lassen. Bei „odd parity“ würde das Kodewort als fehlerhaft erkannt werden.

JA für „even parity“
NEIN für „odd parity“

Ein Nachteil einfacher Paritätskodes ist, dass nur eine ungeradzahlige Anzahl von fehlerhaften Bits erkannt wird. Zudem können diese nicht korrigiert werden, da durch den Paritätscheck nur erkannt wird, dass ein Fehler existiert, nicht aber wo er sich befindet.

b. Wie groß ist die Wahrscheinlichkeit einer falschen Klassifizierung, wenn in dem so kodierten Kanal Bitfehler (random bit errors) mit einer Bit Error Rate (BER) von 10^{-3} auftreten?

Die Wahrscheinlichkeit einer falschen Klassifizierung, d.h. der fälschlichen Akzeptanz eines fehlerhaften Kodewortes durch die Paritätsprüfung kann man mithilfe der Bernoullischen Formel zur Binominalverteilung berechnen.

$$p_k = \binom{n}{k} p^k \cdot (1-p)^{n-k} \qquad \text{mit} \qquad \binom{n}{k} = \frac{n!}{k!(n-k)!}$$

Sie gibt die Wahrscheinlichkeit an, dass bei einem n mal wiederholten Zufallsexperiment (n voneinander unabhängigen Einzelversuche) das Ereignis A mit der Wahrscheinlichkeit p genau k mal auftritt.

Im Kontext der Bitfehlerbetrachtung ist dabei

A: Auftreten eines fehlerhaften Bits
n = 9 Bit bzw. 9 Versuche je Kodewort
k = Anzahl fehlerhafter Bits (Ereignisse) pro Kodewort (Versuchsreihe n)
p= Auftretenswahrscheinlichkeit eines Bitfehlers

Der Term p^k beschreibt die Wahrscheinlichkeit, dass genau k Bits fehlerhaft sind (multiplikative Verknüpfung der Wahrscheinlichkeiten konjunkter Ereignisse: $p_{tot} = p(A) \cdot p(B)$). Der Term $(1 - p)^{n-k}$ beschreibt die Wahrscheinlichkeit, dass die restlichen n-k Bits fehlerfrei sind. Die Konjunktivität (Gleichzeitigkeit) der zwei beschriebenen Zustände verlangt die multiplikative Zusammenfassung der zwei Terme. Der Term $\binom{n}{k}$ beschreibt die Gesamtanzahl aller verschiedenen Anordnungen von *k* fehlerhaften Bits in einem *n*-stelligen Kodewort (Fehlermuster) und muss mit den beiden anderen Termen multipliziert werden.

Um nun die Wahrscheinlichkeit für Falschklassifikation bei Paritätskanalcodierung zu bestimmen, muss man sich zuerst klarmachen, welche Fehler zur Falschakzeptanz eines Kodeworts führen:
Die Paritätsprüfung erkennt dann Fehler nicht, wenn sie geradzahlig d.h. 2-, 4-, 6-, 8-fach usw. auftreten. Die Kodewortfehlerwahrscheinlichkeit ergibt sich nach dem Additionstheorem für disjunkte Ereignisse ($p_{tot} = p(A)+p(B)$) als Summe der Einzelwahrscheinlichkeiten:

$$p_{Kodewortfehler} = p_{2Fehler} + p_{4Fehler} + p_{6Fehler} + p_{8Fehler}$$

Man berechnet also die Auftretenswahrscheinlichkeiten P aller Bitfehlermuster, die von der Paritätsprüfung nicht erkannt werden und summiert diese zur Gesamtfehlerwahrscheinlichkeit $P_{Kodewort}$ auf.

$$p_{Kodewort} = \sum_k p_k \rightarrow [k = 2,4,6,8]$$

$$p_{Kodewort} = \binom{9}{2} 10^{-3\cdot 2}(1-10^{-3})^7 + \binom{9}{4} 10^{-3\cdot 4}(1-10^{-3})^5 + \binom{9}{6} 10^{-3\cdot 6}(1-10^{-3})^3 + \binom{9}{8} 10^{-3\cdot 8}(1-10^{-3})^1$$

$$p_{Kodewort} = 3{,}57\cdot 10^{-5} + 1{,}25\cdot 10^{-10} + 8{,}37\cdot 10^{-17} + 8{,}99\cdot 10^{-24}$$

$$p_{Kodewort} = 3{,}57\cdot 10^{-5}$$