

Aufgabe 3.1

- Dienstleister für die Vermittlungsschicht
- Behandlung von Übertragungsfehlern
- Regulierung des Datenflusses

Aufgabe 3.2

Pakete werden von Rahmen gekapselt.

Aufgabe 3.3

ESC FLAG ESC FLAG ESC ESC X ESC FLAG

Aufgabe 3.4

01111110110111110011111100111111011111010101111110

1. Flag-Bytes (01111110) entfernen:

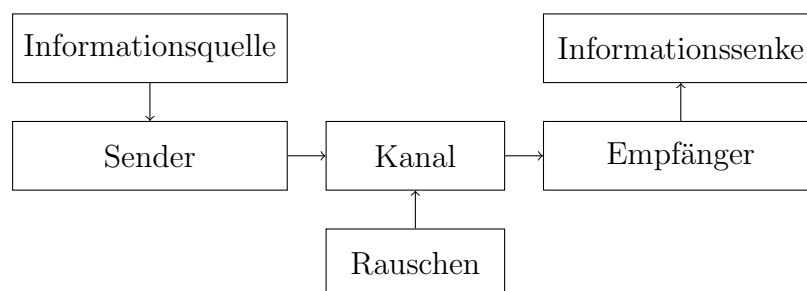
110111110|111110101

2. Da vor dem Einfügen der Flag-Bytes nach jeder Folge von 5 Einsen ein Nullbit eingefügt wurde, muss dieses wieder entfernt werden:

11011111|11111101

Aufgabe 3.5

Modell der Informationsübertragung:

**Aufgabe 3.6**

Es handelt sich um einen Blockcode der Länge 4 mit Codewörter der Länge 6.

Aufgabe 3.7

Die Distanz beträgt $\text{dist}(100001, 001100) = 4$, da sich die Wörter in 4 Positionen unterscheiden.

Bemerkung: Ein Computerprogramm kann dies leicht feststellen, indem es die Codewörter mit der XOR-Funktion (\oplus) verknüpft und dann die Summe der Einsen bestimmt:

$$\begin{array}{r} 100001 \\ \oplus 001100 \\ \hline 101101 \rightarrow 4 \end{array}$$

Aufgabe 3.8

Bestimme das Minimum über alle möglichen Distanzen:

$$\text{dist}(100001, 001100) = 4$$

$$\text{dist}(100001, 000111) = 3$$

$$\text{dist}(100001, 111000) = 3$$

$$\text{dist}(001100, 000111) = 3$$

$$\text{dist}(001100, 111000) = 3$$

$$\text{dist}(000111, 111000) = 6$$

Der Code C hat die Distanz 3.

Aufgabe 3.9

1. Distanz des Codes bestimmen: 3
2. Ein solcher Code kann maximal $3 - 1 = 2$ Fehler erkennen.

Aufgabe 3.10

1. Distanz des Codes bestimmen: 3
2. Der Code kann maximal $\left\lfloor \frac{3-1}{2} \right\rfloor = 1$ Fehler korrigieren.

$\lfloor x \rfloor$ steht für die grösste natürliche Zahl n , die kleiner oder gleich x ist.

Aufgabe 3.11

Fehlererkennung und Fehlerkorrektur wird durch das Hinzufügen *redundanter* Daten ermöglicht. Redundante Daten, sind Daten, die keinen eigentlichen Informationswert besitzen.

Aufgabe 3.12

Da C aus drei Codewörtern besteht, können auch drei Zustände übermittelt werden. Da jedes Codewort aus sechs Bits besteht, beträgt die Informationsrate $3/6 = 0.5 = 50\%$.

Aufgabe 3.13

Der Code muss die Länge $7 + 1 = 8$ haben.

Aufgabe 3.14

In Kommunikationssystemen mit zuverlässigen Kanälen, wo eine Retransmission (erneute Übertragung) keine grosse Verzögerung durch lange Übertragungswege oder erneute Fehler bewirkt.

Aufgabe 3.15

- (a) Das zweite Byte ist falsch übertragen worden, da es eine ungerade Anzahl Einsen enthält.
- (b) Diese Frage lässt sich nicht beantworten, da man mit einem Paritätsbit keine gerade Anzahl Bitfehler erkennen kann.

Aufgabe 3.16

- Ergänze $n = 4$ Nullen rechts: 011000010000
- Polynomdivision 011000010000 : 10011 in $\mathbb{Z}_2 = \{0, 1\}$.
In \mathbb{Z}_2 gibt es keine Überträge (Subtraktion = Addition).

```
011000010000 : 10011 = ...
-10011
-----
  10110
-10011
-----
   10110
-10011
-----
    10100
-10011
-----
     1110 (Rest)
```

- Überschreibe mit dem Rest rechtsbündig die ergänzten 4 Nullen: 011000010111

Aufgabe 3.17

```
111011000001 : 10011 = ...
-10011
-----
  11101
-10011
-----
   11100
-10011
-----
    11110
```

```

-10011
-----
 11010
-10011
-----
 10010
-10011
-----
 0101 (ungleich 0000)

```

Aufgabe 3.18

gesendet: 111111010010

empfangen: 110111110110

Länge eines Burstfehlers (Bündelfehler): Zähle alle Bits vom ersten bis und mit dem letzten Fehler, selbst wenn zwischen diesen Positionen keine weiteren Fehler auftreten.

⇒ Es handelt sich um einen Burstfehler der Länge 8.

Aufgabe 3.19

Die Länge 5 (=Grad des Generatorpolynoms)

Aufgabe 3.20

$$d = 2 \cdot 7 + 1 = 15$$

Aufgabe 3.21

Bei Kanälen mit hohen Latenzzeiten, wo eine Retransmission zu lange dauern würde oder bei Kanälen mit einer hohen Fehlerrate, bei denen eine Retransmission oft wieder zu Fehlern führt.

Aufgabe 3.22

empfangen: 010 110 111 100 001 101 000

korrigiert: 000 111 111 000 000 111 000

Aufgabe 3.23

Die Positionen der Prüfbits sind jeweils unterstrichen.

Positionen	<u>1</u>	<u>2</u>	3	<u>4</u>	5	6	7
Bitfolge	1	0	1	0	1	0	1
Position 1	1		1		1		1
Position 2		0	1		0		1
Position 4				0	1		1

Aufgabe 3.24

Decodierung einer Hamming-codierten Bitfolge:

Pos.	<u>1</u>	<u>2</u>	3	<u>4</u>	5	6	7	<u>8</u>	9	10	11	12	13	14	15
Bits	0	1	0	1	1	1	0	1	0	0	0	0	0	0	0
Pos. 1 f		0		0	1	0		0							
Pos. 2 w		1		0	0	1		0							
Pos. 4 f				1	1	1		0							
Pos. 8 f								1							

Durch Addition aller Fehlerpositionen erkennt man, dass das Bit an der Stelle $1+4+8 = 13$ falsch ist.

Aufgabe 3.25

Das System besteht aus einem Zentralrechner und mehreren angeschlossenen Stationen, die alle durch zwei Kanäle (Up- und Downstream) verbunden sind.

Wenn eine Station senden will, sendet sie ihre Nachricht über den Upstream. Wenn keine andere Station gleichzeitig sendet, so erhält der Zentralrechner die Nachricht und sendet eine Empfangsbestätigung via Downstream.

Wenn hingegen zwei oder mehr Stationen gleichzeitig senden, so empfängt der Zentralrechner aufgrund der Interferenz keine brauchbaren Daten und eine Bestätigung bleibt aus.

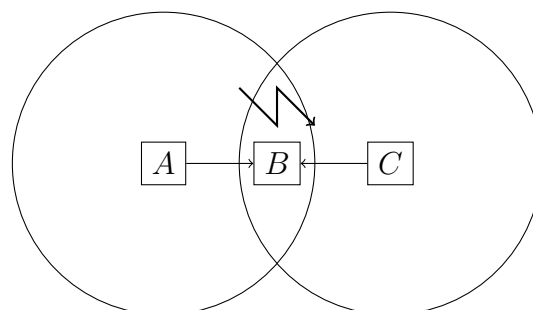
Danach wartet jede der sendewilligen Stationen eine zufällige Zeitspanne und versucht ihr Glück nochmals.

Aufgabe 3.26

- (a) Infrastruktur-Modus
- (b) Ad hoc-Modus

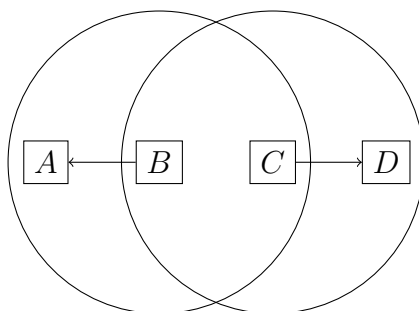
Aufgabe 3.27

Hidden-Station-Problem: Es kommt zu Kollisionen, wenn die Stationen A und C gleichzeitig eine Nachricht an Station B senden wollen. Grund: Die äusseren Stationen A und C wissen aufgrund mangelnder Reichweite nichts von ihrer Existenz.



Aufgabe 3.28

Exposed-Station-Problem: Da die Stationen B und C sich gegenseitig erkennen können, schliessen sie daraus, dass sie nicht gleichzeitig senden dürfen. Dieses Problem besteht jedoch gar nicht, wenn B an A und C an D senden möchte.



Aufgabe 3.29

Der Sender sendet einen RTS-Rahmen (request-to-send), der die Information über die Menge der zu sendenden Daten – und somit über die Kommunikationsdauer enthält.

Erhält der Empfänger diesen Rahmen, bestätigt er diesen mit einem CTS-Rahmen (clear-to-send).

Daraufhin sendet der Sender seine Daten an den Empfänger.

Andere Stationen, die diesen Informationsaustausch mitbekommen, wissen nun, wie lange sie diese Kommunikation nicht stören dürfen.

Hätte ein konkurrierender Sender gleichzeitig einen RTS-Rahmen versendet, würde der Empfänger ein Rauschen empfangen und keinen CTS-Rahmen verschicken. Beide Sender würden dann eine zufällige Zeitspanne warten, bis sie es erneut versuchen.

[