



Übersichtsblatt: Hamming-Code

Grundidee:

Um Fehler in einer Übertragung zu erkennen und trotz dem die Redundanz möglichst gering zu halten wurde der Hamming-Code entwickelt.

Bildung des Hamming-Codes:

Hierbei werden jedem Datenwort der Länge m , r Prüfbits zugeordnet so das eine Wortlänge n von $n = m + r$ entsteht. Die Prüfbits werden immer an den Zweierpotenzen der Dualdarstellung der Positionsangabe der Bits eingefügt. Die Datenbits werden dazwischen eingefügt.

Beispiel:

$2^0=1$	$2^1=2$	3	$2^2=4$	5	6	7	$2^3=8$	9	10	11	12
0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100
P	P	D	P	D	D	D	P	D	D	D	D

Somit enthält jedes Prüfbit in seiner dualen Positionsangabe genau eine 1. Es ist damit das Paritätsbit für alle Datenbits, die in ihrer dualen Positionsangabe an der selben Stelle auch eine 1 haben. (es gilt gerade Parität)

Berechnung der Prüfbits:

Um zu entscheiden ob das Prüfbit eine 1 oder eine 0 enthalten soll, werden alle Datenbits die in ihrer dualen Positionsangabe an der selben Stelle eine 1 haben wie das Prüfbit addiert. Das Ergebnis wird dann mod 2 genommen. Jetzt erhält man den Wert des Prüfbits.

Beispiel:

Codierung des Datenwortes: 01001011

0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100
P	P	0	P	1	0	0	P	1	0	1	1

$$P_{1000} = D_{1001} + D_{1010} + D_{1011} + D_{1100} = 1 + 0 + 1 + 1 = 1$$

$$P_{0100} = D_{0101} + D_{0110} + D_{0111} + D_{1100} = 1 + 0 + 0 + 1 = 0$$

$$P_{0010} = D_{0011} + D_{0110} + D_{0111} + D_{1010} + D_{1011} = 0 + 0 + 0 + 0 + 1 = 1$$

$$P_{0001} = D_{0011} + D_{0101} + D_{0111} + D_{1001} + D_{1011} = 0 + 1 + 0 + 1 + 1 = 1$$

1	1	0	0	1	0	0	1	1	0	1	1
----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------	----------

Fehlersyndrom:

Um nun zu überprüfen ob eine Übertragung eines Hamming-Codierten Wortes auch korrekt war wendet man das so genannte Fehlersyndrom an.

Bildung des Fehlersyndroms:

Man bildet hierbei für jedes Prüfbit das Fehlersyndrom indem man jeweils das Prüfbit mit allen Datenbits addiert, welche in ihrer dualen Positionsangabe an der selben Stelle eine 1 haben. Das Ergebnis wird wieder mit mod 2 gewichtet. Kommt 0 raus, so weiß man hier kein Fehler aufgetreten ist, kommt an irgendeiner Stelle 1 raus so weiß man, dass ein Fehler aufgetreten ist. Man kann durch richtiges Ablesen des Fehlersyndroms sogar bestimmen wo der Fehler aufgetreten ist.

Beispiel:

0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100
0	0	1	0	1	0	0	1	1	0	1	1

$$S_{0001} = 0 + 1 + 1 + 0 + 1 + 1 = 0$$

$$S_{0010} = 0 + 1 + 0 + 0 + 0 + 1 = 0$$

$$S_{0100} = 0 + 1 + 0 + 0 + 1 = 0$$

$$S_{1000} = 1 + 1 + 0 + 1 + 1 = 0 \quad \Rightarrow \text{Es ist kein Fehler aufgetreten!}$$

1	1	0	0	0	1	1	1	0	1	1	0
---	---	---	---	---	---	---	---	---	---	---	---

$$S_{0001} = 1 + 1 + 1 + 0 + 1 + 1 = 0$$

$$S_{0010} = 1 + 0 + 1 + 1 + 1 + 1 = 1$$

$$S_{0100} = 0 + 0 + 1 + 1 + 0 = 0$$

$$S_{1000} = 1 + 0 + 1 + 1 + 0 = 1$$

Von unten nach oben ablesen!

\Rightarrow Es ist ein Fehler bei 1 0 1 0 aufgetreten.

Die korrekte Übertragung wäre also: 1 1 0 0 0 1 1 1 0 0 1 0

