



## 6. Übungsblatt zur Vorlesung

### Technische Informatik I

#### Aufgabe 1

Um ein  $m$ -Bit Datenwort in den Hamming-Code zu überführen, werden  $r \geq 1 + \lfloor \log_2 m \rfloor$  Prüfbits hinzugenommen. Der genaue Wert von  $r$  ergibt sich durch Finden des minimalen Wertes, der die Bedingung  $(m + r + 1) \leq 2^r$  erfüllt. Diese  $r$  Prüfbits stehen in dem  $(m + r)$ -Bit Codewort an den Positionen  $2^0, 2^1, \dots, 2^{r-1}$ . Die ursprünglichen  $m$  Datenbits werden unverändert übernommen, nur ihre Position ist entsprechend verschoben. Man beachte, dass hier die Positionen von 1 ab gezählt werden und dass die Position 1 (das LSB) *ganz rechts* steht!

O.b.d.A. wird das Prüfbit so gesetzt, dass die Anzahl der Einsen an den zu überprüfenden Positionen gerade ist. Die Position  $b$  im  $(m + r)$ -Bit Wort wird durch die Bits an den Positionen  $b_1, \dots, b_j$  überprüft, mit  $b = b_1 + \dots + b_j$  und  $b_i$  ist eine 2er Potenz. (Darstellung einer Zahl als Summe von 2er Potenzen). Beispiel: Das Bit an Position 6 wird von den Bits an den Positionen 2 und 4 überprüft, da  $6 = 4 + 2$  ist.

#### Aufgabenstellung:

a) Überführe die Wörter  $10111101_2$ ,  $10010010_2$  und  $10101110_2$  in einen 1-*Bit* fehlerkorrigierenden Hamming-Code.  
Hinweis zur Vorgehensweise:

- Füge an den Positionen der Prüfbits Leerstellen in das Datenwort ein.
- Finde im Datenwort die Positionen, für welche das Prüfbit an Stelle  $2^0$  mitverantwortlich ist. Verfahre mit den restlichen Prüfbits analog.
- Füge die zuvor berechneten Prüfbits an den richtigen Stellen ein.
- Zum Test: Für  $10111101_2$  ergibt sich  $101111101101_2$ .

b) Bei den codierten Wörtern  $011110000010_2$  und  $101101101011_2$  wurde jeweils ein Bit falsch übertragen. Finde heraus, um welches Bit es sich handelt.

#### Aufgabe 2

Zur Vermarktung des von Bill, Steve und Linus entwickelten *kaWumm* Rechners, haben sie die Firma *intelAct* gegründet.

- a) Eine beauftragte Werbeagentur liefert ihnen folgenden, einprägsamen Werbespruch:

DIE DISKREPANZ ZWISCHEN DER REALITÄT UND IHRER BEHAUPTUNG  
UNTERSTREICHT DIE KUEHNHEIT IHRER AUSSAGE.

Generiere für diesen Text den Huffman-Code, wobei das Leerzeichen „ “ und der Punkt „.“ ignoriert werden sollen. Bestimme dazu die Häufigkeiten der einzelnen Zeichen. Sortiere die Zeichen absteigend nach ihren Häufigkeiten und bei gleicher Häufigkeit in lexikographischer Reihenfolge (z.B.  $D, S$  anstatt  $S, D$ ). Bei der Generierung neuer „Knoten“ sollen die Kombinationen nahe der Blätter bevorzugt werden (z.B.  $\{Z\} \cup \{B\}$  anstatt  $\{B\} \cup \{L, W\}$ ).

b) Dekodiere mit Hilfe des in Teilaufgabe a) generierten Code folgende Nachricht:

001110010010110011101101101101111000110

#### Aufgabe 3

Bei der **Einer-Komplement-Darstellung** ( $[\cdot]_1$ ) gibt die höchstwertige Stelle das Vorzeichen an. Ist dieses Bit gleich 0, so repräsentieren die nachfolgenden Stellen in der üblichen Weise direkt die Zahl. Ist das Vorzeichenbit aber gesetzt, so geben die nachfolgenden Stellen das (bitweise) Komplement des Betrags der Zahl an. Beispiel: Die Folge  $01101$  stellt die Zahl 13 dar, da das höchstwertige Bit nicht gesetzt und  $1101_2 = 13$  ist.  $-13$  hingegen wird durch die Folge  $10010$  dargestellt. Allgemein:

$$[d_n d_{n-1} \dots d_0 d_{-1} \dots d_{-k}]_1 = \left( \sum_{i=-k}^{n-1} 2^i d_i \right) - d_n (2^n - 2^{-k})$$

Beweise:

- a) Der Zahlenbereich ist symmetrisch.
- b) Es gibt zwei Darstellungen für die Null.
- c) Aus der Darstellung von  $a$  erhält man die Darstellung von  $-a$  durch Komplementieren aller Bits.

Hinweis: Fallunterscheidung bezüglich  $d_n$ .

#### Aufgabe 4

Gegeben sei das IEEE-Format für *single precision* Zahlen. Bestimme für die angegebenen Zahlenpaare  $(a, b)$  jeweils, ob  $a > b$ ,  $a < b$  oder  $a = b$  gilt. Die Zahlen sind dabei im Format Vorzeichen|Exponent|Mantisse dargestellt.

- a)  $a = 000110100000010011110100100111$   
 $b = 0001011110111011010011101000110$
- a)  $a = 1001001111111011110001001011001$   
 $b = 00000110111000001001110100111011$
- a)  $a = 10100101100110100100111000101001$   
 $b = 10100101100110110100001000100001$

Abgabetermin: Dienstag, 12.12.0000, 14:00Uhr