



WIEDERHOLUNG

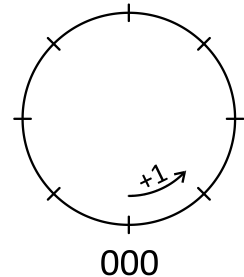
Wandle in Dezimalzahlen bzw. Binärzahlen um:

- | | | | |
|-------------------|-------------------|-------------------|-------------------|
| a) 1110_2 | b) 1011_2 | c) 0111_2 | d) 1001_2 |
| e) $1010\ 0110_2$ | f) $1100\ 1001_2$ | g) $0101\ 1100_2$ | h) $1000\ 0100_2$ |
| i) 9_{10} | k) 15_{10} | l) 4_{10} | m) 13_{10} |
| n) 64_{10} | o) 128_{10} | p) 107_{10} | q) 222_{10} |

MOTIVATION

A) I) Ergänze alle in 3 Bit darstellbaren Binärzahlen im Zahlenkreis rechts. Füge jeweils die entsprechende Dezimalzahl hinzu.

II) Verdeutliche an dem Zahlenkreis die Durchführung und das Ergebnis der Rechenoperation $3+3$.

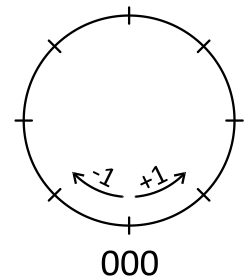


B) Neben den positiven soll der Zahlenkreis zusätzlich negative Zahlen enthalten.

a) Stelle sowohl den positiven als auch den negativen Dezimalzahlen Binärzahlen gegenüber, indem du die Binärzahlen aus dem oberen Kreis übernimmst und die Dezimalzahlen entsprechend der Pfeile einträgst.

b) Nenne den in diesem Fall entstehenden Zahlenbereich in der Mengenschreibweise.

c) Nenne ein sinnvolles Erkennungsmerkmal für negative Zahlen.



ÜBERSICHT

Die hier verwendete Darstellung negativer Binärzahlen wird **Zweierkomplement** genannt. Dieses lässt sich vergleichsweise einfach berechnen, indem alle Bits getauscht werden (dies ergibt das **Einerkomplement**) und anschließend 1 aufaddiert wird. Ein Übertrag über die Stellenzahl hinaus fällt vorne weg (Bei der Null als Spezialfall).

Beispiel:

Binärzahl	1001 1100	0000 0000
Einserkomplement (invertiert)	0110 0011	1111 1111
Zweierkomplement (+1)	0110 0100	±0000 0000

Das höchstwertige Bit wird auch als **MSB** (engl. most significant bit) bezeichnet. Prinzipiell lässt dieses Bit eine negative (bei 1) oder eine positive (bei 0) Zahl erkennen.



TRAINIEREN

Aufgabe 1

Führe mithilfe der in 3.2 durchgeführten Umrechnungsvorschriften zwischen Binär- und Dezimalzahlen folgende Rechnungen über die binäre Darstellung durch und beschreibe deine Beobachtungen.

a)

$$\begin{array}{r} 011 \quad (+3) \\ + \quad 110 \quad (-2) \\ \hline \quad \quad (+1) \end{array}$$

b)

$$\begin{array}{r} \quad \quad (+3) \\ + \quad \quad (-3) \\ \hline \end{array}$$

c)

$$\begin{array}{r} \quad \quad (-2) \\ + \quad \quad (-1) \\ \hline \end{array}$$

d)

$$\begin{array}{r} \quad \quad (-2) \\ + \quad \quad (-2) \\ \hline \end{array}$$

e)

$$\begin{array}{r} \quad \quad (+2) \\ + \quad \quad (+3) \\ \hline \end{array}$$

Aufgabe 2

Stelle alle Zahlenpaare (d.h. Zahl und Gegenzahl) für 4-Bit lange Binärzahlen auf.

Aufgabe 3

Ermittle die Zahlen: - 5, - 6, - 7 als Binärzahlen mithilfe des Zweierkomplements.

Aufgabe 4

Berechne im Binärsystem durch Addition der Gegenzahl:

a) $1110 - 0110$,

b) $1010 - 1100$

c) $-1010 - 1111$.

Aufgabe 5

Gib die Grenzen der Zahlenbereiche von *shortint* (8 Bit) und *integer* (16 Bit) sowohl als Binär- als auch als Dezimalzahl an.

VERNETZEN

Aufgabe 6

Entwirf analog zum 4-Bit-Addierer einen 4-Bit-Subtrahierer.

→ *Tipp: Benutze dazu den 1-Bit-Inkrementierer und den 4-Bit-Addierer*