2 Zahlendarstellung

2.1 Stellenwertsysteme (B-adische Systeme)

2.1.1 Darstellung natürlicher Zahlen

Jede natürliche Zahl ist durch die folgende Summenformel darstellbar:
$N = a_{n-1} \cdot B^{n-1} + a_{n-2} \cdot B^{n-2} + \cdots + a_1 \cdot B^1 + a_0 \cdot B^0 = \sum_{i=0}^{n-1} a_i \cdot B^i$

N = Zahl im entsprechenden Stellenwertsystem mit der Stellenanzahl n
B = Basis (Radix), z.B. 10 (dezimal), 2 (dual)
B^i = Wertigkeit (Gewicht) der i-ten Stelle
a_i = Ziffer (Koeffizient) der i-ten Stelle aus der Menge der Ziffern $\{0, 1, 2, \ldots, B\text{-}1\}$

2.1.2 Zerlegung nach dem Horner-Schema

$N = \sum_{i=0}^{n-1} a_i \cdot B^i = (\ldots(a_{n-1} \cdot B + a_{n-2}) \cdot B + \cdots + a_1) \cdot B + a_0$
Man erhält die Ziffern $a_0, a_1, \ldots, a_{n-1}$ dadurch, dass man die Zahl N durch B dividiert und man nach der Division den Rest abtrennt und ihn notiert, d.h. man erhält zuerst a_0. Mit dem Quotienten wird dann in derselben Weise verfahren, bis er nach wiederholter Durchführung verschwindet. Die notierten Ziffern a_i stellen dann die Zahl im Stellenwertsystem zur Basis B dar.

Beispiel 21. *Konvertierung ins Dualsystem*
Ausgehend vom Dezimalsystem und unter Verwendung der Dezimalarithmetik ist es möglich mit dem beschriebenen Verfahren Dualzahlen zu erzeugen:

$167_{10} \rightarrow$	$167 / 2 = 83$	*Rest 1*	*Least Significant Bit, LSB*
	$83 / 2 = 41$	*Rest 1*	
	$41 / 2 = 20$	*Rest 1*	
	$20 / 2 = 10$	*Rest 0*	
	$10 / 2 = 5$	*Rest 0*	
	$5 / 2 = 2$	*Rest 1*	
	$2 / 2 = 1$	*Rest 0*	
	$1 / 2 = 0$	*Rest 1*	*Most Significant Bit, MSB*

$\rightarrow 10100111_2$

Beispiel 22. *Rückkonvertierung ins Dezimalsystem*
Einsetzen der Ziffern (Koeffizienten) in die allgemeine Gleichung zur Zahlendarstellung mit B=2 (Summenformel oder Horner-Schema). Im folgenden Beispiel in Form der Summenformel:
$10100111_2 = 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 167_{10}$

2.2 Vorzeichenlose Zahlen

Vorzeichenlose Dualzahlen (unsigned binary numbers, unsigned integers) werden als natürliche Zahlen im Stellenwertsystem 2 mit n Stellen $\mathsf{a_i}$ folgendermaßen dargestellt:
$\mathsf{D} = \sum_{i=0}^{n-1} a_i \cdot 2^i = a_{n-1} \cdot 2^{n-1} + a_{n-2} \cdot 2^{n-2} + \cdots + a_1 \cdot 2^1 + a_0 \cdot 2^0$

2.2.1 Addition

Die Addition von Dualzahlen wird durchgeführt indem paarweise die Stellen der Summanden addiert werden (mit Übertrag). Angewendet werden folgende Rechenregeln:

$$\begin{aligned} 0+0 &= 0 \\ 0+1 &= 1 \\ 1+0 &= 1 \\ 1+1 &= \mathbf{1}0 \end{aligned}$$

Bei 1+1=**1**0 entsteht eine “1” als Übertrag (Carry c), die eine zusätzliche Stelle benötigt.

Beispiel 23. *Addition:*

$$\begin{array}{lr} 5 + 6 = 11 \;\; \rightarrow & 0101 \\ & +0110 \\ \hline & \mathring{1}011 \end{array}$$

$$\begin{array}{lr} 5 + 12 = 17 \;\; \rightarrow & 0101 \\ & +1100 \\ \hline & \mathring{1}\mathring{0}001 \end{array}$$

2.2.2 Subtraktion

Die Subtraktion von Dualzahlen wird durchgeführt indem paarweise die Stellen des Subtrahenden von denen des Minuenden subtrahiert werden (mit Übertrag). Angewendet werden folgende Rechenregeln:

$$\begin{aligned} 0\text{-}0 &= 0 \\ 0\text{-}1 &= 1 \\ 1\text{-}0 &= 1 \\ 1\text{-}1 &= 0 \end{aligned}$$

Bei 0-1=1 wird von der nächsthöheren Stelle eine “1” als Übertrag geborgt (Borrow b).

Beispiel 24. *Subtraktion:*

$$\begin{array}{lr} 12 - 5 = 7 \;\; \rightarrow & 1100 \\ & -0101 \\ \hline & \mathring{0}\mathring{1}\mathring{1}1 \end{array}$$

$$\begin{array}{lr} 4 - 5 = -1 \;\; \rightarrow & 0100 \\ & -0101 \\ \hline & \mathring{1}\mathring{1}\mathring{1}\mathring{1}1 \end{array}$$

2.2.3 Zahlenbereich

Der Zahlenbereich von Dualzahlen kann als Zahlenring dargestellt werden. In der Abbildung 2.1 ist die Zahlenbereichsgrenze eingezeichnet. Wird der Zahlenbereich überschritten, so ist das an dem Übertragsbit des höchstwertigen Bits zu erkennen (Carry am Most Significant Bit).

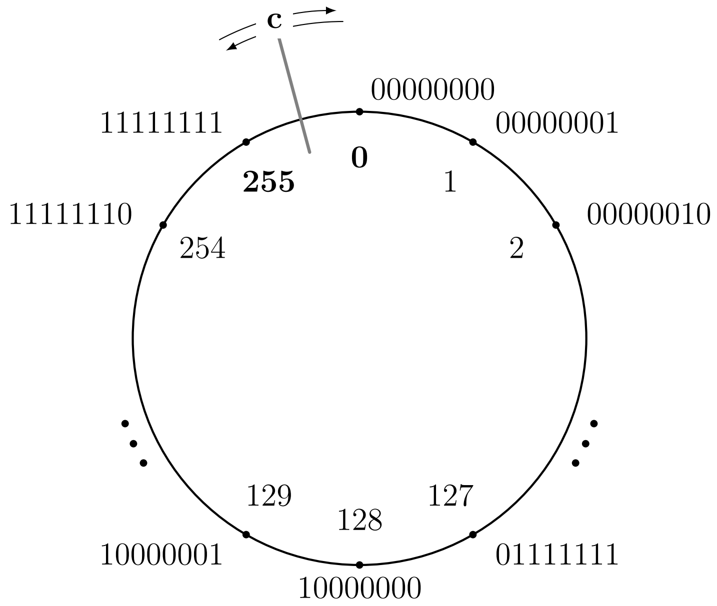

Abbildung 2.1: Zahlenring für vorzeichenlose Dualzahlen

2.3 Vorzeichenbehaftete Zahlen

Es gibt unterschiedliche Möglichkeiten negative Dualzahlen darzustellen. Ausgangspunkt bei der Codierung negativer Zahlen ist die Codierung positiver Dualzahlen. Im Folgenden werden die verbreitetsten Varianten aufgelistet:

- Vorzeichen-/Betrags-Zahlen (VB-Zahlen):
  Höchstwertiges (`n`-tes) Bit bestimmt das Vorzeichen: "0" zeigt eine positive , "1" eine negative Zahl an. Die restlichen `n`-1 Bits sind codiert wie vorzeichenlose Dualzahlen und bestimmen den Betrag der Zahl.

- 1-Komplement-Zahlen (1K-Zahlen):
  Negative Zahlen erhält man durch das bitweise Invertieren einer vorzeichenlosen Dualzahl mit selben Betrag. Höchstwertiges (`n`-tes) Bit impliziert das Vorzeichen: "0" zeigt eine positive , "1" eine negative Zahl an.

- 2-Komplement-Zahlen (2K-Zahlen):
  Negative Zahlen werden erzeugt indem die entsprechenden positiven Dualzahl bitweise invertiert wird und anschließend eine “1” addiert wird. Höchstwertiges (n-tes) Bit (Negative n) impliziert das Vorzeichen: “0” zeigt eine positive , “1” eine negative Zahl an.

Beispiel 25. *Zahlen mit 4-Bit-Wortlänge:*

positive Zahlen		*negative Zahlen*			
dez.	*dual*	*dez.*	*VB-Zahl*	*1K-Zahl*	*2K-Zahl*
+0	0 000	-0	1 000	1 111	—
+1	0 001	-1	1 001	1 110	1 111
+2	0 010	-2	1 010	1 101	1 110
+3	0 011	-3	1 011	1 100	1 101
+4	0 100	-4	1 100	1 011	1 100
+5	0 101	-5	1 101	1 010	1 011
+6	0 110	-6	1 110	1 001	1 010
+7	0 111	-7	1 111	1 000	1 001
		-8	—	—	1 000

2.4 2-Komplement-Zahlen

In der Praxis werden (nahezu) ausschließlich vorzeichenbehaftete Zahlen (signed binary numbers, signed integers) in 2K-Codierung benutzt. Im Stellenwertsystem 2 mit n Stellen a_i werden sie als ganze Zahlen folgendermaßen dargestellt:
$Z = -a_{n-1} \cdot 2^{n-1} + \sum_{i=0}^{n-2} a_i \cdot 2^i$ (d.h. mit negativem Gewicht für die Vorzeichenstelle n-1).

2.4.1 Bildung

Das Darstellen einer 2-Komplement-Zahl, d.h. die 2-Komplement-Bildung, folgt der Vorschrift:

$$
\begin{aligned}
-Z &= -(-a_{n-1} \cdot 2^{n-1} + \sum_{i=0}^{n-2} a_i \cdot 2^i) = a_{n-1} \cdot 2^{n-1} - \sum_{i=0}^{n-2} a_i \cdot 2^i \\
&= a_{n-1} \cdot 2^{n-1} - \sum_{i=0}^{n-2} a_i \cdot 2^i + (2^{n-1} - 1) - (2^{n-1} - 1) \\
&= a_{n-1} \cdot 2^{n-1} - 2^{n-1} - \sum_{i=0}^{n-2} a_i \cdot 2^i + \sum_{i=0}^{n-2} 2^i + 1 \\
&= -(1 - a_{n-1}) \cdot 2^{n-1} + \sum_{i=0}^{n-2} (1 - a_i) \cdot 2^i + 1
\end{aligned}
$$

Beispiel 26. *2-Komplement-Bildung:*

$$\begin{array}{rll}
 & 01101001_2 & = +105_{10} \\[4pt]
\textit{Invertierung}: & 10010110_2 & \\
\textit{Addition von Eins}: & +\qquad\;\; 1 & \\ \hline
\textit{2-Komplement}: & 10010111_{2K} & = -105_{10}
\end{array}$$

und zurück:

$$\begin{array}{rll}
 & 10010111_{2K} & = -105_{10} \\[4pt]
\textit{Invertierung}: & 01101000_2 & \\
\textit{Addition von Eins}: & +\qquad\;\; 1 & \\ \hline
\textit{2-Komplement}: & 01101001_{2K} & = +105_{10}
\end{array}$$

2.4.2 Addition

Genau wie bei den vorzeichenlosen Dualzahlen, findet die Addition der 2K-Zahlen stellenpaarweise statt und es gibt einen Übertrag. Auch bei 2K-Zahlen kann es zu einer Bereichsüberschreitung kommen, diese wird aber zur Unterscheidung von der bei den vorzeichenlosen Dualzahlen Überlauf (Overflow v) genannt.

Beispiel 27. *Addition:*

$$\begin{array}{rr}
5 + (-4) = 1 \;\rightarrow & 0101 \\
 & +1100 \\ \hline
 & \mathring{1}\mathring{0}001
\end{array}$$

Es gibt keinen Überlauf, da die beiden 'obersten' Überträge 1 sind ($\mathsf{v} = 1 \oplus 1 = 0$)*. Es handelt sich um das korrekte positive Ergebnis:* $0001_{2K} \equiv +1_{10}$.

2.4.3 Subtraktion

Äquivalent zur Addition erfolgt die Subtraktion der 2K-Zahlen genau so wie bei den vorzeichenlosen Zahlen. Auch hier findet eine Bereichsüberschreitung bei einem Überlauf statt.

Beispiel 28. *Subtraktion:*

$$\begin{array}{rr}
(-128) - 127 = -255 \;\rightarrow & 1000\,0000 \\
 & -0111\,1111 \\ \hline
 & \mathring{0}\mathring{0}\mathring{0}\mathring{0}\,\mathring{0}\mathring{0}\mathring{0}1
\end{array}$$

Es gibt einen Überlauf, da nur einer der beiden 'obersten' Überträge 1 ist ($\mathsf{v} = 0 \oplus 1 = 1$)*. Es ensteht ein falsches positives Ergebnis:* $0000\,0001_{2K} \equiv +1_{10}$.

2.4.4 Zahlenbereich

Der Zahlenkreis verdeutlicht wie sich der Zahlenbereich, und somit die Grenze an der eine Bereichsüberschreitung stattfinden kann, verschoben hat.

Die ‘0’ markiert im Gegensatz zu den vorzeichenlosen Zahlen nicht mehr das untere Ende des Zahlenbereiches, sondern die Mitte, und die Grenze ist zu der anderen Stelle des Zahlenkreises gewandert, an der das höchstwertigste Bit (MSB, negative Bit) seinen Wert ändert. Die Zahlenbereichsüberschreitung ist am Überlauf (Overflow v) zu erkennen.

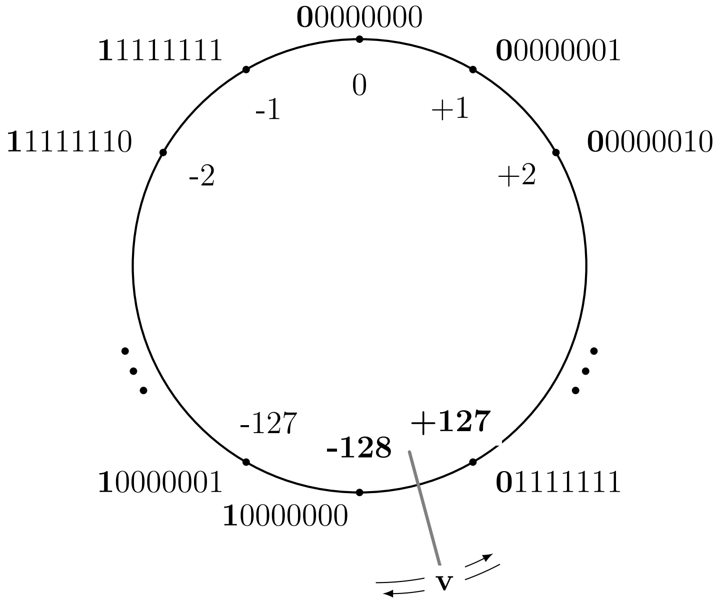

Abbildung 2.2: Zahlenring für 2-Komplement-Zahlen

2.5 Gray-Code

Neben der Codierung zur Informationsdarstellung und -verarbeitung gibt es weitere Codierungsmöglichkeiten bei deren Verwendung andere Randbedingungen im Vordergrund stehen. Bei dem Gray-Code sind diese Bedingungen Sicherheit und Zuverlässigkeit, die darin Berücksichtigung finden, dass benachbarte Codes sich jeweils durch genau ein Bit unterscheiden. Der Übergang von einem Code zum nächsten findet somit durch eine wohldefinierte Veränderung statt. Für die Erzeugung des Gray-Code gibt es eine rekursive Bildungsvorschrift:

$$[\mathsf{G}_{\mathsf{i+1}}]_{\mathsf{2n\times(i+1)}} = \begin{array}{l} \begin{bmatrix}0\\ \vdots\\ 0\end{bmatrix}_{\mathsf{n}} \cup \begin{bmatrix} \mathsf{G}_{\mathsf{i}} \end{bmatrix}_{\mathsf{n\times i}} \\ \begin{bmatrix}1\\ \vdots \\ 1\end{bmatrix}_{\mathsf{n}} \cup \begin{bmatrix} \mathsf{G}_{\mathsf{i}}^{\mathsf{R}}\end{bmatrix}_{\mathsf{n\times i}}\end{array}, \quad \mathsf{G}_1 = \begin{bmatrix}0\\1\end{bmatrix}_{\mathsf{2\times 1}}, \quad \begin{array}{lll} \mathsf{i} & \geqslant & \mathsf{1} \\ \mathsf{n} & \dots & \text{Anzahl der Codes in Stufe i} \\ \mathsf{G}_{\mathsf{i}}^{\mathsf{R}} & \dots & \mathsf{G}_{\mathsf{i}} \text{ gespiegelt}\end{array}$$

Den reflektierten Code erhält man durch Spiegelung in der Zeilenmitte.

Beispiel 29. *Gray-Code für 3 Bit:*

Dez.	*Gray-Code*
0_{10}	*000*
1_{10}	*001*
2_{10}	*011*
3_{10}	*010*
4_{10}	*110*
5_{10}	*111*
6_{10}	*101*
7_{10}	*100*