Zahlencodierung III

In unseren Betrachtungen der Zahlensysteme fehlen uns noch wichtige Darstellungsarten, nämlich die negativen und die reellen Zahlen. Diese sind für wissenschaftliche Berechnungen, Simulationen usw. unabdingbar.

Negative Zahlen

Alle bisherigen Zahlen die wir betrachtet haben waren positiv. Nun gibt es aber unendlich viele Anwendungen für negative Zahlen, wie stellen wir diese in der Informatik dar?
Die am nahe liegendste Möglichkeit wäre es, die bestehenden Zahlen mittels eines Bits die Information "Positiv" oder "Negativ" mitzugeben. Bsp:

Vorzeichen				Zahlenteil				Dezimal
0	0	0	0	0	0	0	1	+1
0	0	0	0	0	1	1	1	+7
1	0	0	0	1	0	0	1	-9

Leider ergibt sich bei dieser fiktiven Darstellung ein Problem bei der Berechnung. Wir nehmen ein Beispiel aus dem Dezimalsystem, dabei wird durch die Addition von negativen Zahlen eine Subtraktion erreicht. $55 - 13 \triangleq 55 + (-13) = 42$

Mit dieser Methode kann bei einem Prozessor eine Operation (Minusrechnen) eingespart werden, indem vor dem Addieren die zu subtrahierende Zahl negiert wird. Damit das funktioniert, wird die **Einerkomplementdarstellung** eingeführt:

Einerkomplement

Wie funktioniert die Negierung einer Zahl?

Die Zahl als Binärzahl darstellen, dabei eine genügend grosse Länge auswählen	$13_{10} = 00001101_2$
Das Vorzeichen wird gewechselt indem alle Bits umgekehrt (invertiert) werden	$13_{10} = 00001101_2$ $-13_{10} = 11110010_2$

Nun machen wir eine Berechnung: $55_{10} - 13_{10}$

```
  00110111₂
+ 11110010₂
  111 11
-----------
  00101001₂
```

Rechnen Sie alle Beispiele eigenhändig nach. Sie müssen die Addition in allen Zahlensystemen von AB114-01 beherrschen!

Einschub Addition von Binärzahlen:
Die Summenbildung erfolgt wie bei den Dezimalzahlen von rechts nach links. Es wird nach folgenden Regeln addiert:

	Bsp 1	Bsp 2	Bsp 3
Zahl 1	0	01	011
Zahl 2	1	01	011
Übertrag	*0*	*10*	*110*
Summe	1	10	110

Was fällt Ihnen bei der obigen Berechnung im Einerkomplement auf?

Minimale und Maximale Zahlen im Einerkomplement. Rechnen Sie ins Dezimalsystem um! ➔***1***

Wie Sie aus der obigen Übung entnehmen, können Zahlen von -127 bis +127 dargestellt werden. Zudem gibt es zwei Darstellungsarten von 0. Auch die Addition rechnet nicht ganz korrekt. Das Einerkomplement muss also verbessert werden.

Zweierkomplement

Wie funktioniert die Negierung einer Zahl?

Die Zahl als Binärzahl darstellen, dabei eine genügend grosse Länge auswählen	$13_{10} \triangleq 00001101_2$
Das Vorzeichen wird gewechselt indem alle Bits umgekehrt (invertiert) werden	$13_{10} \triangleq 00001101_2$ $-13_{10} \triangleq 11110010_2$
Es wird zur Zahl 1 dazugerechnet	$-13_{10} \triangleq 11110011_2$

Nun machen wir eine Berechnung: 55_{10} - 13_{10}

$$
\begin{array}{lll}
 & 00110111_2 & \text{(Zahl 1)} \\
+ & 11110011_2 & \text{(Zahl 2)} \\
 & 111\ 111 & \text{(Übertrag)} \\
\hline
 & 00101010_2 \triangleq & 42_{10}
\end{array}
$$

Wir sehen, dass die Subtraktion bzw. die Addition der Zahl im 2er Komplement das richtige Resultat ergibt. Auch die Problematik mit den zwei Darstellungen von 0 fällt weg. ➔***2***

Exzessdarstellung (engl. excess oder offset binary)

Eine in der Praxis oft vorkommende Variante der Zahlendarstellung basiert auf der Verschiebung des Zahlenbereichs:

	Zahlenbereich							
Positive Zahlen	0	1	2	3	4	5	6	7
Binäre Zahlen	000	001	010	011	100	101	110	111
Exzessdarstellung	-4	-3	-2	-1	0	1	2	3

Wie einfach zu erkennen ist, wird der Zahlenbereich der positiven Zahlen bei der Exzessdarstellung verschoben. Dabei wird eine negative Zahl mehr dargestellt als positive. (-4 bis +3).
Wir können nun einfach vorhersehen welche Zahlen mit 8 Bits dargestellt würden:

2^8 = 256 -> es würden die Zahlen -128 bis +127 dargestellt.

Die Umrechnung von Dezimalzahlen in die Exzessdarstellung und umgekehrt ist denkbar einfach; Es wird immer 2^{n-1} addiert oder subtrahiert. ➔***3***

Fliesskommazahlen (engl. floating point numbers)

Die Königsklasse der Zahlendarstellungen sind die Fliesskommazahlen. In der realen Welt sind ja schliesslich auch fast nur solche zu finden:

- Oberfläche einer Kugel
- Grösse eines Baumes
- Fallzeit eines Blattes
- Zinsen pro Monat

Eine "normale" Fliesskommazahl könnte so aussehen: 5,437712
Auch sehr kleine Zahlen müssen berücksichtigt werden: 0,0000000000000089
(Das Komma ist nach rechts "geflossen")
Grosse Zahlen sind auch wichtig: 14'445'713'809'098'224'451

Wie können nun all diese Anforderungen unter einen Hut gebracht werden? Die Lösung heisst Exponentialdarstellung.

Sicher haben Sie schon auf einem Taschenrechner diese Angabe gesehen: 1,3E12
Was bedeutet das? Das Komma der Zahl 1,3 wird um 12 Stellen nach rechts verschoben, es wird eigentlich die Zahl 1'300'000'000'000 dargestellt.

Die Formel dazu lautet: $1{,}3 * 10^{12}$

Begriffe: $$\textbf{Mantisse} * \textbf{Basis}^{\textbf{Exponent}}$$

Ein Mikroprozessor rechnet fast mit dem gleichen System. Der kleine Unterschied ist die Basis, welche 2 statt 10 ist. Damit wird die Umrechnung für den Mikroprozessor einfacher, für den Menschen aber schwieriger. (➔Artikel "AB114-03 Gleitkommazahlen.pdf" als Vertiefung)

Damit alle Mikroprozessoren mit dem gleichen System arbeiten und die Zahlen unter den Rechnern austauschbar sind, wurden Normen festgelegt. (**IEEE 754**) ➔***4***

Es gibt 2 Normen:

Single Precision 32 Bit
Double Precision 64 Bit

Je nach Norm wird eine andere Anzahl an Bits für die Darstellung der Mantisse und des Exponents verwendet.

Um die Details der Berechnung zu verstehen hilft die Excel Datei "AB114-03 Rechnen mit Dualzahlen.xls". ➔***5***

Das IEEE-Gleitkommaformat (IEEE 754)

Genauigkeit	single precision	double precision
Vorzeichen (bits)	1 (0 = positiv, 1 = negativ)	
Exponent (bits)	8	11
Fraction (= Mantisse ohne hidden bit) (bits)	23 (hidden bit)	52 (hidden bit)
Gesamtbreite (bits)	32	64
Excess	127	1023
Exponent	-126 ... +127	-1022 ... +1023

Aufgabenteil

1) Negative Zahlen im 1er Komplement

Rechnen Sie die Zahlen links (1er Komplement) in Dezimalzahlen um!

$00000000_2 \triangleq$ ____________________$_{10}$

$11111111_2 \triangleq$ ____________________$_{10}$

$00000001_2 \triangleq$ ____________________$_{10}$

$11111110_2 \triangleq$ ____________________$_{10}$

2) Negative Zahlen im 2er Komplement

Lösen Sie folgende Aufgaben mit 8 Bit Dualzahlen im 2-Komplement:

$+32_{10}$ $+15_{10}$	$+32_{10}$ -15_{10}
-32_{10} $+15_{10}$	-32_{10} -15_{10}

Nennen Sie die grösste und kleinste Zahl (Dezimal) welche mit dem 2-Komplement mit 8 Bits dargestellt werden kann:

Kleinste Zahl: ________________$_{10}$ ________________$_2$

Grösste Zahl: ________________$_{10}$ ________________$_2$

Was passiert wenn Sie die Zahl 0_{10} mittels 8 Bit ins 2-Komplement umrechnen?

3) Dezimalzahlen <-> Exzessdarstellung

Berechnen Sie im Kopf die jeweils fehlenden Werte in der Tabelle!
Es wird immer in einem 8 Bit System gerechnet.

DEZIMAL	EXZESSDARSTELLUNG
Bsp: -34	01011110
100	
-63	
	11001100
	00101101
1	
-1	

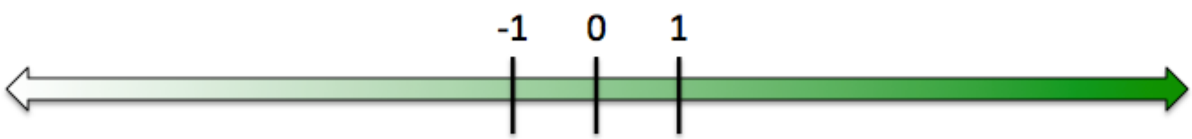
Zur Kontrolle Ihrer Berechnungen können Sie Formeln in Excel bauen.

Wie können Sie in der Exzessdarstellung zwischen positiven und negativen Zahlen unterscheiden? Begründen Sie!

4) Fliesskommazahlen

Ordnen Sie die Vorzeichenvarianten der entsprechenden Stelle auf dem Zahlenstrahl zu!

Vorzeichen Mantisse	+	+	−	−
Vorzeichen Exponent	+	−	+	−
Zuordnen:	A	B	C	D



5) Fliesskommazahlen

Stellen Sie mit Hilfe der Exceldatei die folgenden Zahlen dar. Notieren Sie dazu die Bits des Exponents (gelbe Zellen) und der Mantisse (blaue Zellen).
Das "hidden Bit" wird dabei nicht notiert. Alle Bits in einer Reihe werden weiter zu einer Hexadezimalen Zahl umgerechnet, um diese einfacher darzustellen.

Beispiel:

```
376917.25    ≙ 010010001 01110000000101010101000
             ≙ 48B80AA8
```

```
5.75    ≙ ____________________________________________

        ≙ ____________________________________________

-5.75   ≙ ______________________________________

        ≙ ____________________________________________

0.0089  ≙ ____________________________________________

        ≙ ____________________________________________
```

Welche Fliesskommazahl wird aus der folgenden Hexadezimalzahl gewonnen?

```
493C8C74    ≙ ____________________________________________
```