Zahlensysteme

Inhaltsverzeichnis

Dezimalsystem....................1
Binärsystem....................1
Umrechnen Bin → Dez....................2
Umrechnung Dez → Bin....................2
Rechnen im Binärsystem – Addition....................3
Die negativen ganzen Zahlen im Binärsystem....................4
Rechnen im Binärsystem - Subtraktion....................4
Hexadezimalsystem....................5
Umrechnung Hex → Dez....................5
Umrechnung Dez → Hex....................6
Umrechnung Bin → Hex → Bin....................6

Dezimalsystem

Basis: 10
Stellen: 0,1,2,3,4,5,6,7,8,9

Jede Dezimalzahl kann wie folgt zerlegt, dargestellt werden

$$3268_{dez}=3\cdot10^3+2\cdot10^2+6\cdot10^1+8\cdot10^0$$ (die Basis ist gut zu erkennen)

$$12{,}46_{dez}=1\cdot10^1+2\cdot10^0+4\cdot10^{-1}+6\cdot10^{-2}$$

Binärsystem

Basis: 2
Stellen: 0,1

In der Technik ist das Dezimalsystem unbrauchbar. In den meisten Fällen gibt es nur 2 unterscheidbare Zustände:
Stromleitung → Strom Ja / Nein; Schalter → Ein / Aus; Lichtwellenleiter → Licht Ein / Aus; u.s.w.
Jeder Computer, Microcontroller arbeitet mit dem Binärsystem, das nur 2 Zustände kennt (0,1)

$$1011_{bin} = 1\cdot 2^3 + 0\cdot 2^2 + 1\cdot 2^1 + 1\cdot 2^0$$
$$10{,}11_{bin} = 1\cdot 2^1 + 0\cdot 2^0 + 1\cdot 2^{-1} + 1\cdot 2^{-2}$$

Man nennt in diesem Zusammenhang

1 binäre Zahl = **Bit**
4 binäre Zahlen = **Nibble**
8 binäre Zahlen = **Byte**

Eine Gruppierung des Binärcodes in Nibbles erleichtert das Lesen und das spätere Arbeiten mit Hexadezimalzahlen.

Umrechnen Bin → Dez

Durch die Zerlegung in obige Schreibweise bereits geklärt. Die 0er kann man gleich weglassen

$$1\ 1011_{bin} = 1\cdot 2^4 + 1\cdot 2^3 + 1\cdot 2^1 + 1\cdot 2^0 = 16 + 8 + 2 + 1 = 27_{dez}$$
$$1011\ 1100_{bin} = 1\cdot 2^7 + 1\cdot 2^5 + 1\cdot 2^4 + 1\cdot 2^3 + 1\cdot 2^2 = 128 + 64 + 32 + 8 + 4 = 236_{dez}$$

Eine Kenntnis des Binärsystems bis 2^{10} ist dabei recht hilfreich, und beim Programmieren sowieso ein (stillschweigende) Voraussetzung

2^0	1
2^1	2
2^2	4
2^3	8
2^4	16
2^5	32
2^6	64
2^7	128
2^8	256

2^9	512
2^{10}	1024
2^{11}	2048
2^{12}	4096
2^{13}	8192
2^{14}	16384
2^{15}	32768
2^{16}	65536

Umrechnung Dez → Bin

Entweder wird obige Tabelle verwendet, und die Zahl zusammen gestückelt

$$67_{dez} = 64 + 2 + 1 = 1\cdot 2^6 + 1\cdot 2^1 + 1\cdot 2^0 = 100\ 0011_{bin}$$

Oder man verwendet folgendes System

	Rest
67:2=33	1
33:2=16	1
16:2=8	0
8:2=4	0
4:2=2	0
2:2=1	0
1:2=0	1

↑ Leserichtung ↑

$$67_{dez}=100\ 0011_{bin}$$

In der Praxis schreibt man

$$\begin{array}{r|l} 67 & 1 \\ 33 & 1 \\ 16 & 0 \\ 8 & 0 \\ 4 & 0 \\ 2 & 0 \\ 1 & 0 \\ 0 & 1 \end{array}$$

↑ Leserichtung ↑

Zahlen $0 \leq x < 1$

- Multiplikation mal 2
- die Einerstelle ist die gesuchte Binärstelle
- falls das Produkt größer als 1 ist, wird 1 subtrahiert.

↓ Leserichtung ↓

$$\begin{array}{rl} 0{,}375 \cdot 2 = 0{,}75 & 0 \\ 0{,}75 \cdot 2 = 1{,}5 - 1 = 0{,}5 & 1 \\ 0{,}5 \cdot 2 = 1{,}0 - 1 = 0 & 1 \end{array}$$

$$0{,}375_{dez} = 0{,}011_{bin}$$

Rechnen im Binärsystem – Addition

Für die Addition im Binärsystem verwendet man grundsätzlich dieselbe Prozedur wie im Dezimalsystem.

Dezimalsystem				**Binärsystem**								
	5	7		0	0	1	1	1	0	0	1	
+	4	4		0	0	1	0	1	1	0	0	+
1	1		***Übertrag***		1	1	1					***Übertrag***
1	0	1	***Ergebnis***	0	1	1	0	0	1	0	1	***Ergebnis***

Die negativen ganzen Zahlen im Binärsystem

Man verwendet zur Darstellung der negativen Zahlen sogenannte „Flags“ (Vorzeichenbits). So wird also z.B. bei einer 8-bit Zahl das erste Bit für positive Zahlen auf 0 gesetzt, für negative Zahlen auf 1.

29	0	0	0	1	1	1	0	1
-29	1	0	0	1	1	1	0	1

Der Zahlenbereich für eine 8-bit Binärzahl wird also von 0..255 auf -128...127 verändert. Im allgemeinen kann man also mit *n* Bits Zahlen von -2^{n-1} bis $+2^{n-1}-1$ darstellen.

Rechnen im Binärsystem - Subtraktion

Für die Subtraktion gibt es mehrere Möglichkeiten. Man kann genau wie im Dezimalsystem die Zahlen untereinander schreiben und mit Übertrag von einander abziehen:

17 – 4 = 13

0	0	0	1	0	0	0	1	17
0	0	0	0	0	1	0	0	- 4
			1	1				*Übertrag*
0	**0**	**0**	**0**	**1**	**1**	**0**	**1**	***13***

Die weitaus wichtigere Methode stellt die Addition der **Zweierkomplemente** dar. Dabei invertiert man den Subtrahenden und addiert dann die inverse Zahl. Um das Zweierkomplement bilden zu können, benötigt man zuerst das **Einerkomplement**. Das wird einfach gebildet, indem man jedes Bit invertiert, dh. jede 0 zu 1 macht und umgekehrt:

0	0	0	0	0	1	0	0	4
1	1	1	1	1	0	1	1	Einerkomplement von 4

Nachteil des Einerkomplements ist, dass für die Null zwei Darstellungen existieren: 00000000 und 11111111 sind identisch.

0	0	0	0	0	1	0	0	4
1	1	1	1	1	0	1	1	- 4
								Übertrag
1	1	1	1	1	1	1	1	0

Man kann also jetzt nicht mehr davon ausgehen, dass zwei Zahlen genau dann gleich sind, wenn ihre Binärcodes übereinstimmen. Diesen Fehler behebt man mit dem Zweierkomplement. Dabei wird zum Einerkomplement +1 addiert.

1	1	1	1	1	0	1	1	Einerkomplement von 4
							1	
1	**1**	**1**	**1**	**1**	**1**	**0**	**0**	***Zweierkomplement von 4***

Das Zweierkomplement stellt jetzt eine eindeutige Darstellung einer negativen Zahl dar. Der Vorteil für die Recheneinheit besteht nun vor allem darin, dass man nicht zwischen positiven und negativen Zahlen unterscheiden muss, sondern einfach nur bitweise addiert und so zum gewünschten Ergebnis kommt.

Zurück zum Ausgangsbeispiel: 17 – 4 = 13

0	0	0	1	0	0	0	1	17
1	1	1	1	1	1	0	0	Zweierkomplement von 4
1	1	1						Übertrag
0	**0**	**0**	**0**	**1**	**1**	**0**	**1**	**Ergebnis**

Der 9. Stelle, die in diesem Falle mit 1 belegt wäre, wird keinerlei Beachtung geschenkt und somit erhält man das richtige Ergebnis.

Hexadezimalsystem

Basis: 16
Stellen: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

Da Binärzahlen leicht recht groß werden, verwendet man gerne das Hexadezimalsystem, das recht eng mit dem Binärsystem verbunden ist.

Umrechnung Hex → Dez

$$AF41_{hex}=A\cdot 16^3+F\cdot 16^2+4\cdot 16^1+1\cdot 16^0=10\cdot 4096+15\cdot 256+4\cdot 16+1=44865_{dez}$$

Umrechnung Dez → Hex

	Rest
1998:16=124	14 = E
124:16=7	12 = C
7:16=0	7

↑ Leserichtung ↑

$$1998_{dez}=7CE_{hex}$$

Zahlen $0\leq x<1$

- Multiplikation mal 16
- die Einerstelle ist die gesuchte Hex-Stelle
- falls das Produkt größer als 1 ist, wird die Einerstelle subtrahiert

$0{,}575\cdot 16=9{,}2-9=0{,}2$	9
$0{,}2\cdot 16=3{,}2-3=0{,}2$	3
$0{,}2\cdot 16=3{,}2-3=0{,}2$	3

↓ Leserichtung ↓

$$0{,}575_{dez}=0{,}93\dot{3}_{hex}$$ (periodisch)

Umrechnung Bin → Hex → Bin

Je ein Nibble entspricht einer Hex-Zahl

$$\underbrace{1011}_{11=B}\ \underbrace{0110}_{6}\ \underbrace{0101}_{5}\ \underbrace{1111}_{15=F}=B65F_{hex}$$

$$A2F5_{hex}=\underbrace{1010}_{A}\ \underbrace{0010}_{2}\ \underbrace{1111}_{F}\ \underbrace{0101}_{5}$$