

Binäres Zahlensystem (Dualzahlen)

Ein binäres (von lateinisch bina „doppelt, paarweise“) System, auch Dualsystem oder Zweiersystem genannt, kennt nur zwei Zustände und verwendet somit nur zwei Ziffern 0 und 1 zur Darstellung von Zahlen. Jede Ziffer einer Binärzahl oder einer Binärfolge wird als Bit bezeichnet. Im Speicher elektronischer Geräte und bei Datenübertragung werden Bits für kompakte Darstellung in Gruppen von acht Bits eingeteilt. Eine Gruppe aus 8 nacheinander folgenden Bits nennt man Byte oder Oktett.

4096	2048	1024	512	256	128	64	32	16	8	4	2	1
2^{12}	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0

Da es nur zwei Zustände gibt, also 0 und 1 (aus und an) werden beim Umrechnen von Binärzahlen also nur von den „gesetzten Bits“ (alle auf 1) die jeweiligen 2er Potenzen addiert.

z.B:

1	0	1	0 ₂
↓	↓	↓	↓
2 ³	2 ²	2 ¹	2 ⁰
↓	↓	↓	↓
8	4	2	1
↓	↓	↓	↓
8	0	2	0

$8 + 0 + 2 + 0 = \underline{\underline{10_{10}}}$

Die Binäre Zahl 1010₂
Ergibt dezimal 10

Mit 4 Bit können also 16 verschiedene Zustände (Zahlen von 0-15) dargestellt werden.

Mit 8 Bit sind es schon 256 verschiedenen Zustände (Zahlen von 0-255). (**8 Bit** sind **1 Byte** 😊)

Ebenso einfach lassen sich dezimale Zahlen in Binäre Zahlen umwandeln, nur eben in umgekehrter Reihenfolge. Nehmen wir z.B. die Antwort auf Alles also die 42.

Um die 42 binär darzustellen benötigen wir das 32er (2^5), das 8er (2^3) und das 2er (2^1) Bit.

Also schreiben wir es mal auf:

64		32		16		8		4		2		1
		↓				↓				↓		
0		1		0		1		0		1		0

Vorrangestellte Nullen also alle, die wir vor der 32 nicht brauchen (die 64 passt ja nicht in die 42) müssen nicht beachtet werden.

Daraus folgt, dass die binäre Antwort auf Alles „101010“ lautet 😊

„Es gibt 10 Arten von Menschen. Die einen verstehen das binäre Zahlensystem und die anderen nicht.“



Hexadezimals Zählensystem

Große Binärzahlen haben den Nachteil, dass sie sehr unübersichtlich sind. Um dem Abhilfe zu schaffen hat man das Hexadezimalsystem eingeführt. Dabei werden 4 Bit einer Dualzahl durch ein hexadezimales Zeichen ersetzt. Da eine 4-Bit Dualzahl 16 Zustände annehmen kann, wir aber nur 10 dezimale Zahlen kennen, hat man dem hexadezimalen Zahlensystem 6 Buchstaben hinzugefügt.

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

Basis: 16

Größter Nennwert: F

Stellenwerte: $16^0 = 1$, $16^1 = 16$, $16^2 = 256$, usw.

Zum besseren Verständnis der Zählweise im hexadezimalen Zahlensystem dient diese Tabelle. Jeweils 4 Dualstellen bilden eine Hexadezimalstelle.

Dezimal	Binär/Dual	Hexadezimal
0	0 0 0 0	0
1	0 0 0 1	1
2	0 0 1 0	2
3	0 0 1 1	3
4	0 1 0 0	4
5	0 1 0 1	5
6	0 1 1 0	6
7	0 1 1 1	7
8	1 0 0 0	8
9	1 0 0 1	9
10	1 0 1 0	A
11	1 0 1 1	B
12	1 1 0 0	C
13	1 1 0 1	D
14	1 1 1 0	E
15	1 1 1 1	F

Umrechnungen am Beispiel der Zahl:

39917₍₁₀₎

1 0 0 1 1 0 1 1 1 1 0 1 1 0 1₍₂₎

9BED₍₁₆₎

Dezimal- in Binärzahlen

39917

32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1
1	0	0	1	1	0	1	1	1	1	1	0	1	1	0	1
R: 7149			R: 3053	R: 1005		R: 493	R: 237	R: 109	R: 45	R: 13		R: 5	R: 1		R: 0

oder

39917	/ 2	=	19958	R: 1
19958	/ 2	=	9979	R: 0
9979	/ 2	=	4989	R: 1
4989	/ 2	=	2494	R: 1
2494	/ 2	=	1247	R: 0
1247	/ 2	=	623	R: 1
623	/ 2	=	311	R: 1
311	/ 2	=	155	R: 1
155	/ 2	=	77	R: 1
77	/ 2	=	38	R: 1
38	/ 2	=	19	R: 0
19	/ 2	=	9	R: 1
9	/ 2	=	4	R: 1
4	/ 2	=	2	R: 0
2	/ 2	=	1	R: 0
1	/ 2	=	0	R: 1



1 0 0 1 1 0 1 1 1 1 0 1 1 0 1

Dezimal- in Hexadezimal

39917

39917	/ 16	=	2494	R: 13
2494	/ 16	=	155	R: 14
155	/ 16	=	9	R: 11
9	/ 16	=	0	R: 9

9BED

Binär- in Dezimalzahl

1 0 0 1 1 0 1 1 1 1 0 1 1 0 1

32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1
1	0	0	1	1	0	1	1	1	1	1	0	1	1	0	1
↓			↓	↓		↓	↓	↓	↓	↓		↓	↓		↓
32768			4096	2048		512	256	128	64	32		8	4		1

Die Summe aus allen Werten, die auf 1 sind (an)

= 39917

Binär- in Hexadezimalzahl

1 0 0 1 1 0 1 1 1 1 0 1 1 0 1

9 B E D

Hexadezimal- in Binärzahl

1 0 0 1 1 0 1 1 1 1 0 1 1 0 1

9 B E D

Dez	Bin	Hex
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Hexadezimal- in Dezimalzahl

9 B E D

D=	13	$*16^0$	=	13*1	=	13
E=	14	$*16^1$	=	14*16	=	224
B=	11	$*16^2$	=	11*256	=	2816
9=	9	$*16^3$	=	9*4096	=	36864

39917

Rechnen mit binären Zahlen

Addition:

Regeln:

$$\begin{aligned} 0 + 0 &= 0 \\ 0 + 1 &= 1 \\ 1 + 0 &= 1 \\ 1 + 1 &= 0 \quad \text{Übertrag: 1} \\ 1 + 1 + 1 &= 1 \quad \text{Übertrag: 1} \end{aligned}$$

Beispiel:

$$\begin{array}{r} 11010110 \\ + 11100110 \\ \hline 11 \quad 11 \\ 110111100 \end{array}$$

Subtraktion:

Regeln:

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

Beispiel:

$$\begin{array}{r} 11010110 \\ - 1100110 \\ \hline 11 \\ 01110000 \end{array}$$

Multiplikation:

Regeln:

$$\begin{aligned} 0 * 0 &= 0 \\ 0 * 1 &= 0 \\ 1 * 0 &= 0 \\ 1 * 1 &= 1 \quad (\text{klingt verrückt 😊}) \end{aligned}$$

$$\begin{array}{r} 11001100 * 1010 \\ 11001100 \\ \hline 00000000 \\ 11001100 \\ + \quad 00000000 \\ \hline 1111111000 \end{array}$$

Für jede 1 im Multiplikant (die hintere Zahl 😊) wird Multiplikator (die vordere Zahl 😊) aufgeschrieben. Bei JEDER Ziffer (auch der Null) wird um eins nach rechts Verschohen und anschließend nur zusammenaddiert.

Division:

Beispiel:

$$\begin{array}{r}
 101010 / 111 = \cancel{000}110 \\
 \underline{1} \\
 10 \\
 \underline{101} \\
 1010 \\
 \underline{-111} \\
 111 \\
 0011 \\
 \underline{111} \\
 -111 \\
 \underline{-111} \\
 00
 \end{array}$$

Hier ist es wie beim normalen dividieren auch. Man holt sich eine Ziffer nach der anderen von oben runter und schaut, ob der Divisor (die hintere Zahl) reinpasst. Wenn nicht, dann eine Null in die Ergebniszeile schreiben und die nächste Ziffer dazu holen. Wenn sie passt, eine Eins in die Ergebniszeile schreiben und die Differenz zwischen der Zahl und dem Divisor (durch Subtraktion) berechnen und mit dieser dann weitermachen. Am Ende können vorangestellte Nullen weggelassen werden.