

Grundlagen

Zahlensysteme

- Dual (Basis: 2, Wertebereich: [0,1]) $101 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$
- Dezimal (Basis: 10, WB: [0,9]) $852 = 8 \cdot 10^2 + 5 \cdot 10^1 + 2 \cdot 10^0$
- Hexadezimal (Basis: 16, WB: [0,9]&[A,F]) $1FF = 1 \cdot 16^2 + 15 \cdot 16^1 + 15 \cdot 16^0$

Zahlen umrechnen

- von Dual nach Dezimal $101001_2 \rightarrow 1 \cdot 2^5 + 1 \cdot 2^3 + 1 \cdot 2^0 \rightarrow 32 + 8 + 1 = 41_{10}$

Dual	2^6	2^5	2^4	2^3	2^2	2^1	2^0
Dualzahl eintragen	1	1	0	1	0	1	0
Wert in Dezimal	64	32		8		2	
Dezimalwerte addieren				106			

- von Hexadezimal nach Dezimal: Honorschema

Zahl eintragen	1	F	F
Addition	+ 0	16	496
Ergebnis	1	31	511

*(Arrows indicate multiplication by 16: 1 * 16 = 16, 16 * 16 = 256, 256 + 16 = 272, 272 + 16 = 288, 288 + 16 = 304, 304 + 16 = 320, 320 + 16 = 336, 336 + 16 = 352, 352 + 16 = 368, 368 + 16 = 384, 384 + 16 = 400, 400 + 16 = 416, 416 + 16 = 432, 432 + 16 = 448, 448 + 16 = 464, 464 + 16 = 480, 480 + 16 = 496)*

- von Dezimal nach Dual

$$\begin{aligned} 24 / 2 &= 12 \text{ Rest } 0 \\ 12 / 2 &= 6 \text{ Rest } 0 \\ 6 / 2 &= 3 \text{ Rest } 0 \\ 3 / 2 &= 1 \text{ Rest } 1 \\ 1 / 2 &= 0 \text{ Rest } 1 \rightarrow 11000_2 \end{aligned}$$

- von Dezimal nach Hexadzimal

$$\begin{aligned} 256 / 16 &= 16 \text{ Rest } 0 & 255 / 16 &= 15 \text{ Rest } 15 \\ 16 / 16 &= 1 \text{ Rest } 0 & 15 / 16 &= 0 \text{ Rest } 15 \rightarrow FF_{16} \\ 1 / 16 &= 0 \text{ Rest } 1 \rightarrow 100_{16} \end{aligned}$$

- von Dual nach Hexadzimal

Dualzahl	0	0	1	0	1	0	1	0
Dezimalwerte			2			8		

immer 4 Stellen zusammenzählen $2A_{16}$

- Addition im Dualsystem
Vorgehen: Addition der einzelnen Stellen: $0 + 0 = 0$, $0 + 1 = 1$, $1 + 0 = 1$, $1 + 1 = 0$ mit Übertrag von 1

$\overset{1}{+}$	$\overset{1}{1}$	$\overset{1}{1}$	$\overset{1}{1}$	$\overset{0}{0}$	$\overset{1}{1}$
	1	0	1	0	1

$\overset{1}{+}$	$\overset{1}{1}$	$\overset{1}{0}$	$\overset{1}{1}$	$\overset{0}{0}$	$\overset{1}{0}$	$\overset{1}{0}$	$\overset{1}{1}$
	1	0	0	0	0	1	0

Speicherverwaltung

Speicher eines PCs:

```

1011100111001101011101110110110110111000010010010101010
10100101011110100110111101001101010101010101010101010
1010101011010101010101010101010101010101010101011011111010
10000101010110010110010100100101010101010101010101010101
010101010101010101010101010101011011111010100001010101001
0110010101010101010101010101010101010101010101010101100101100101
010101010101011101001101010101010101010101010101010101010101
011010101010101010101010101010101010101010101010110111110101000010
  
```

Problem: Was sollen Einsen und Nullen bedeuten? Speicher muss verwaltet werden! Für einen bestimmten Teil des Speichers wird also die Bedeutung festgelegt. Bsp: Datentypen: int speichert ganze Zahlen
Beim Speichern der Zahl 22 wird Platz im Speicher reserviert. Die Menge des Platzes ist dabei nur vom gewählten Datentyp abhängig, nicht vom Wert der Zahl. Für int sind 4 Byte üblich:

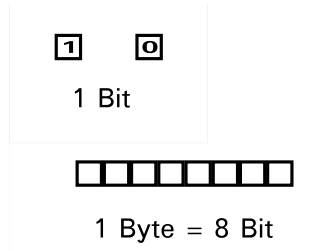


Zahl 22



Ein Experiment

Nehmen wir an, uns wird zum Speichern einer ganzen Zahl 1 Byte Speicherplatz zur Verfügung gestellt:



Fall 1: Speichern der Zahl 101

$$\begin{aligned} 101 / 2 &= 50 \text{ Rest } 1 \\ 50 / 2 &= 25 \text{ Rest } 0 \\ 25 / 2 &= 12 \text{ Rest } 1 \\ 12 / 2 &= 6 \text{ Rest } 0 \\ 6 / 2 &= 3 \text{ Rest } 0 \\ 3 / 2 &= 1 \text{ Rest } 1 \\ 1 / 2 &= 0 \text{ Rest } 1 \rightarrow 01100101_2 \end{aligned}$$

Fall 1: Speichern der Zahl 282

$$\begin{aligned} 282 / 2 &= 141 \text{ Rest } 0 \\ 141 / 2 &= 70 \text{ Rest } 1 \\ 70 / 2 &= 35 \text{ Rest } 0 \\ 35 / 2 &= 17 \text{ Rest } 1 \\ 17 / 2 &= 8 \text{ Rest } 1 \\ 8 / 2 &= 4 \text{ Rest } 0 \\ 4 / 2 &= 2 \text{ Rest } 0 \\ 2 / 2 &= 1 \text{ Rest } 0 \\ 1 / 2 &= 0 \text{ Rest } 1 \rightarrow 100011010_2 \end{aligned}$$