

Bits, Nibbles, Bytes und Zahlensysteme

Kaufmännisches Rechnen

WS-LAT

28. Februar 2025

1 Bits und Bytes

- Das kleinste Speicherelement eines Computers ist das **Bit** (binary digit).
- Ein Bit kann zwei Zustände haben: **0** oder **1** (binäres System).
- **Ein Byte besteht aus 8 Bits** und ist die kleinste adressierbare Speichereinheit in vielen Computersystemen.

| Einheit | Größe in Bits |
|----------|---------------|
| 1 Bit | 1 Bit |
| 1 Nibble | 4 Bits |
| 1 Byte | 8 Bits |

- Ein **Nibble** besteht aus 4 Bits und entspricht einer hexadezimalen Ziffer (0–F).

2 Binäres und Hexadezimals Zahlensystem

- Das **dezimale System** (Basis 10) verwendet die 10 Ziffern **0 bis 9**.

– Jede Stelle in einer Binärzahl entspricht einer Zehnerpotenz, z. B.:

$$* 1010 = 1 * 10^3 + 0 * 10^2 + 1 * 10^1 + 0 * 10^0 = 1000 + 0 + 10 + 1$$

| Dezimal (Basis 10) | | | | |
|--------------------|--------|--------|--------|--------|
| 10^4 | 10^3 | 10^2 | 10^1 | 10^0 |
| 4 | 5 | 6 | 3 | 2 |
| 10.000 | 1.000 | 100 | 10 | 1 |
| Summe | | | | |
| 45632 | | | | |

- Das **binäre System** (Basis 2) verwendet nur die Ziffern **0 und 1**.
- Jede Stelle in einer Binärzahl entspricht einer Zweierpotenz, z. B.:

$$\begin{aligned}
 * 1010 &= 1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0 = 8 + 0 + 2 + 0 = 10 \\
 * 1100 &= 1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 0 * 2^0 = 8 + 4 + 0 + 0 = 12 \\
 * 11 &= 1 * 2^1 + 1 * 2^0 = 2 + 1 = 3
 \end{aligned}$$

| Binär (Basis 2) | | | | |
|-----------------|-------|-------|-------|-------|
| 2^4 | 2^3 | 2^2 | 2^1 | 2^0 |
| 1 | 0 | 1 | 1 | 1 |
| 16 | 8 | 4 | 2 | 1 |
| Summe | | | | |
| 23 | | | | |

- Das **hexadezimale System** (Basis 16) verwendet die Ziffern **0-9 und A-F**:

- 0-9 entspricht 0-9 im Dezimalsystem.
- A-F entspricht 10-15 im Dezimalsystem.
- Hexadezimalzahlen werden oft mit **0x** oder **h** gekennzeichnet (z. B. 0x2F oder 2Fh).
- Jede Stelle einer Hexadezimalzahl, entspricht einer Potenz von 16, z. B.:

$$\begin{aligned}
 * A3F1 &= 10 * 16^3 + 3 * 16^2 + 15 * 16^1 + 1 * 16^0 \\
 &= 40.960 + 768 + 240 + 1 = 41.969
 \end{aligned}$$

| Hexadezimal (Basis 16) | | | | | |
|------------------------|--------|--------|--------|--------|--------|
| 16^5 | 16^4 | 16^3 | 16^2 | 16^1 | 16^0 |
| C | B | 3 | E | 6 | A |
| 1.048.576 | 65.536 | 4.096 | 256 | 16 | 1 |
| Summe | | | | | |
| 13319786 | | | | | |

Beispiel für eine binäre zu hexadezimale Umwandlung:

- Binär: 1010 1100 → 8+0+2+0=10 ; 8+4+0+0=12 → Hexadezimal: AC (A=10, C=12)

3 Speichergrößen: KB, MB, TB, PB vs. KiB, MiB, TiB, PiB

- In der Informatik werden Speichergrößen oft in **Kilobyte (KB)**, **Megabyte (MB)**, **Terabyte (TB)**, **Petabyte (PB)** angegeben.
- Diese Begriffe basieren ursprünglich auf Zweierpotenzen (1 KB = 1024 Byte), werden aber im Marketing oft als Zehnerpotenzen (1 KB = 1000 Byte) verwendet.

- Zur eindeutigen Unterscheidung gibt es die **binären Präfixe** der IEC-Norm:

| Dezimale Einheit | | Größe in Bytes | Binäre Einheit | | Größe in Bytes |
|------------------|-----------|-------------------------|------------------|----------|-------------------------|
| 1 KB (Kilobyte) | 10^3 | 1.000 B | 1 KiB (Kibibyte) | 2^{10} | 1.024 B |
| 1 MB (Megabyte) | 10^6 | 1.000.000 B | 1 MiB (Mebibyte) | 2^{20} | 1.048.576 B |
| 1 GB (Gigabyte) | 10^9 | 1.000.000.000 B | 1 GiB (Gibibyte) | 2^{30} | 1.073.741.824 B |
| 1 TB (Terabyte) | 10^{12} | 1.000.000.000.000 B | 1 TiB (Tebibyte) | 2^{40} | 1.099.511.627.776 B |
| 1 PB (Petabyte) | 10^{15} | 1.000.000.000.000.000 B | 1 PiB (Pebibyte) | 2^{50} | 1.125.899.906.842.624 B |

3.1 Umrechnung durch wiederholtes Multiplizieren / Dividieren mit 1000/1024

Sie können zwischen den Einheiten durch wiederholtes Multiplizieren / Dividieren mit 1000/1024 umrechnen.
Beispiel:

- $42 \text{ TB} * 1000 * 1000 * 1000 * 1000 = 42.000.000.000.000 \text{ B} / 1024 / 1024 / 1024 / 1024 = 38,1987774743 \text{ TiB} \approx 39 \text{ TiB}$
- $42 \text{ TB} * 1000 * 1000 * 1000 * 1000 = 42.000.000.000.000 \text{ B} / (1024 * 1024 * 1024 * 1024) = 38,1987774743 \text{ TiB} \approx 39 \text{ TiB}$
- $42 \text{ TB} * 10$

12

 $= 42.000.000.000.000 \text{ B} / 2$

40

 $\approx 39 \text{ TiB}$

Warum ist das wichtig?

- Betriebssysteme wie Windows nutzen oft die **dezimale Darstellung** (1 KB = 1000 Byte).
- Linux und andere Unix-Systeme nutzen häufig die **binäre Darstellung** (1 KiB = 1024 Byte).
- Dies kann zu Verwirrung führen, z. B. zeigt eine **1 TB Festplatte** unter Windows **1.000 GB**, unter Linux aber nur **931 GiB**.

3.2 Aufgaben

- Ein Unternehmen betreibt ein Rechenzentrum, in dem pro Sekunde 1 Gb/s an Datenverkehr verarbeitet wird.
 - Wie viele GByte werden pro Stunde verarbeitet?
 - Wie viele GiB entspricht das in einer Stunde auf volle GiB gerundet?
- Ein Server speichert täglich 850 GByte an Logdateien. Die Logdateien müssen für 30 Tage gespeichert bleiben, bevor sie gelöscht werden.
 - Wie viele Terrabyte (TB) Speicher ist nötig?
 - Wie groß wäre der Speicherbedarf in Tebibyte (TiB)?
 - Wie viele Mebibyte (MiB) entsprechen der gesamten Speichermenge?
- Ein Unternehmen speichert seine täglichen Produktionsdaten mit einem Volumen von 15 TiB in einem Cloud-Backup-System. Das Backup-System zeigt die Datenkapazität jedoch in TB.
 - Wie viele TB entsprechen den 15 TiB?
 - Das Unternehmen plant, die Speichermenge auf zwei Monate auszudehnen. Wie viel Speicherplatz in TB wird dann benötigt?
 - Wievielen PiB entspricht das rund?

4 Fazit

- Bits und Bytes sind die Grundlage der digitalen Speicherung.
- Das binäre und hexadezimale Zahlensystem wird in der Informatik intensiv genutzt.
- Speichergrößen haben unterschiedliche Konventionen (dezimal vs. binär).
- Die richtige Interpretation ist wichtig für Speicherberechnungen und technische Vergleiche.

5 Lösungen

1. Lösungswege: $1 \text{ Gb/s} \cdot 60 \text{ s/min} \cdot 60 \text{ min/h} = 3.600 \text{ Gb/s}$

$$3.600 \text{ Gbs/s} \cdot 1.000 \cdot 1.000 \cdot 1.000 / 8 \text{ bit/Byte} = 450.000.000.000 \text{ B(yte)}$$

a) $450.000.000.000 \text{ B(yte)} / (1.000 \cdot 1.000 \cdot 1.000) = 450 \text{ GB}$

b) $450.000.000.000 \text{ B(yte)} / (1.024 \cdot 1.024 \cdot 1.024) \approx 420 \text{ GiB}$

2. Lösungswege: $850 \text{ GB} \cdot 30\text{d} = 25.500 \text{ GB}$

a) $25.500 \text{ GB} / 1.000 = 25,5 \text{ TB}$

b) $25.500 \text{ GB} \cdot 10^9 / 2^{40} = 23,1921148951 \text{ TiB} \approx 24 \text{ TiB}$

c) $23,1921148951 \text{ TiB} \cdot (1024 \cdot 1024) = 24.318.695,0682 \text{ MiB} \approx 24.318.695$

3. Lösungswege:

a) $15 \text{ TiB} \cdot (1.024 \cdot 1.024 \cdot 1.024 \cdot 1.024) / (1.000 \cdot 1.000 \cdot 1.000 \cdot 1.000) = 16,4926744166 \text{ TB} \approx 17 \text{ TB}$

b) $16,4926744166 \text{ TB} \cdot 31\text{d} \cdot 2 \text{ Monate} = 1.022,54581383 \text{ TB} \approx 1.023 \text{ TB}^1$

c) $1.022,54581383 \text{ TB} \cdot 10^{12} / 2^{50} = 0,908203125001 \text{ PiB} \approx 1 \text{ PiB}$

¹Sie müssen mit der maximalen Anzahl an Tagen pro Monat rechnen: Beim Übergang von Juli auf August und Dezember auf Januar folgen jeweils 2 Monate mit 31 Tagen aufeinander.