

# Bits und Bytes

Wie werden Daten gespeichert  
Wie werden sie dargestellt (codiert)

# Es gibt 10 Arten von Leuten

- Jene, die Binärzahlen lesen können
- Und die anderen

Bit (binary digit, Binärziffer)

Anzahl Zustände mit mehreren Bits

# Bytes

- Folge von 8 Bits (normalerweise)

# Zählwerk-Analogie

- Zehnersystem (Dezimalsystem)
- Zweiersystem (Binärsystem)
- Sechzehnersystem (Hexadezimalsystem)

# Zahlen im Binärsystem

- Präfix 0b (“Null be”) für binär
  - Präfix 0x (“Null ix”) für hexadezimal
  - Kein Präfix für dezimal
- Rechnen Sie zwischen binär und dezimal um:
  - a) 0b10 1010    b) 32    c) 31
  - d\*) 0x2c
  - e\*) 0b1111 1010 1100 1110 (nur nach hexadezimal)

# Zahlen im Computer

- 1 Byte:
- 2 Bytes:
- 4 Bytes:
- 8 Bytes:
- Negative Zahlen
- Gebrochene Zahlen → Binärbrüche, z.B.  $0b10.01 = 2.25$



# Buchstaben

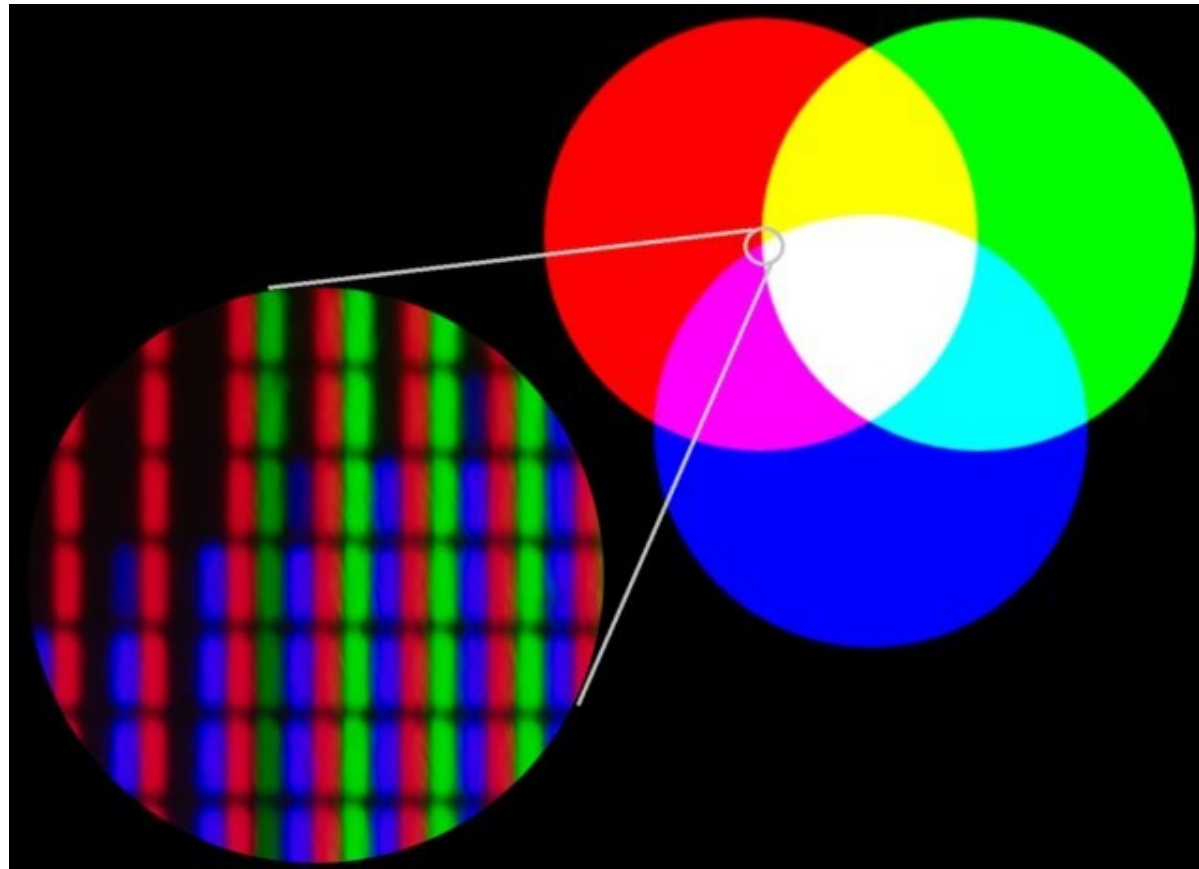
- Idee: Buchstaben durchnummerieren
- ASCII-Code, einfaches lateinisches Alphabet (nur 7 Bits)
  - Hunderte von Codierungen für andere Zeichen (ä, 数, 🙏)
- Unicode: Jedem Buchstaben eine Nummer
  - Zur Zeit über 137'000 Schriftzeichen definiert

# Aufgabe

- Bestimmen Sie den ASCII-Code des ersten Buchstabens Ihres Vornamens, in Gross- und Kleinschreibung
  - Dezimal und binär
- Was ist speziell an der Binärdarstellung der beiden Buchstaben?

# Farben

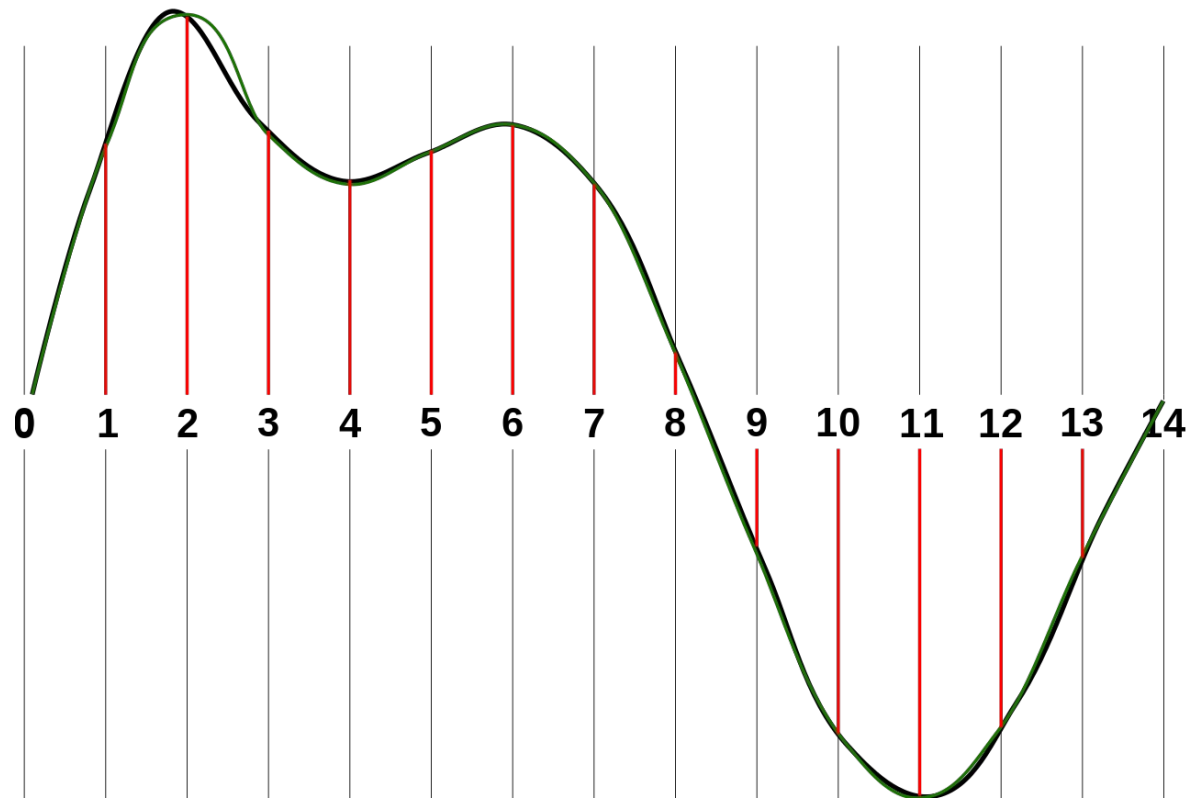
- Lichtmischung aus
  - rot, grün, blau: RGB
- Intensität jeder Farbe
  - in einem Byte codiert
  - 3 Bytes →



# Bilder

- Sammlung von Farbpunkten (Pixel)
  - Folge von 3-Byte Packeten
- Video: Abfolge von Bildern

# Ton



# Dokumente

- Text-Dateien
- Evtl. komprimiert oder zip-Archive von Text-Dateien

# Umrechnungsalgorithmen

- Algorithmus
  - Unmissverständliche Verfahrensvorschrift mit definierter Ein- und Ausgabe
  - Formulierung und Detailtiefe hängen vom Zielpublikum ab.
  - Kann i.d.R. als Computerprogramm umgesetzt werden

# Umrechnung dezimal → binär

- **Eingabe:** Dezimalzahl  $d > 0$
- **Ausgabe:** Binärzahl  $b$
- Beginne mit  $b$  leer (noch keine Ziffern)
- Wiederhole folgende Punkte so lange  $d > 0$  ist:
  - Wenn  $d$  gerade ist, füge vorne bei  $b$  eine 0 an
  - Sonst:
    - Füge vorne bei  $b$  eine 1 an und vermindere  $d$  um 1
  - Teile  $d$  durch 2



# Aufgabe (auf Papier)

- Je einen Algorithmus zur Umrechnung
  - Von binär zu dezimal
  - Von dezimal zu binär
- Algorithmen austauschen und an einem Beispiel prüfen
  - Korrektes Resultat?
  - Unmissverständlichkeit?
  - Allgemein korrekt?