

Wiederholung - Zahlendarstellung

- Alle Zahlen durch andere Basen darstellbar
 - Computer benutzt Basis 2
- Stelle der Ziffer bestimmt Wert
- Konvertierung: Wie funktioniert das Horner-Schema? (Dezimal -> Binär)
- Addition, Subtraktion, Multi., Division
- Vorzeichen: VZB, 1K, 2K
- Umrechnung von VZL -> 2K?
- Rechnen mit 2K: Wann gibt es einen Overflow?



Gleitkommazahlen

IEEE 754, Arithmetik

Gleitkommazahlen

- Zahlen mit Stellen vor und nach dem Komma

$$\begin{aligned} & a_{n-1}a_{n-2} \dots a_1a_0 \text{ , } a_{-1}a_{-2} \dots a_{-m} \\ &= a_{n-1}B^{n-1} + \dots + a_1B^1 + a_0B^0 + a_{-1}B^{-1} + a_{-2}B^{-2} + \dots + a_{-m}B^{-m} = \\ &= \sum_{i=0}^{n-1} a_i B^i + \sum_{i=-m}^{-1} a_i B^i \end{aligned}$$


- Nicht alle rationale Zahlen darstellbar auf Computer
 - Begrenzte Ressourcen

Umrechnung

- Dezimal -> Binär (Horner-Schema):

Beispiel: $0,24_D$

$0,24_D \rightarrow$

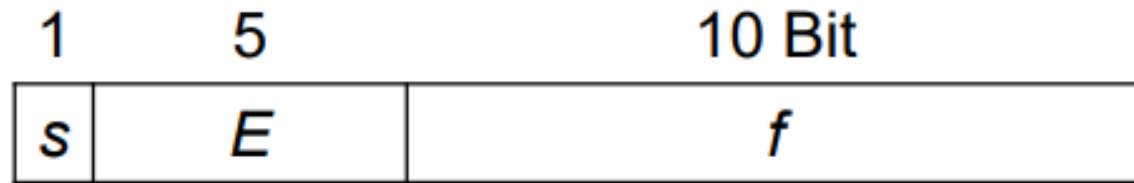
$0,24 \cdot 2 = 0,48 + 0$		MSB!
$0,48 \cdot 2 = 0,96 + 0$		
$0,96 \cdot 2 = 0,92 + 1$		
$0,92 \cdot 2 = 0,84 + 1$		
$0,84 \cdot 2 = 0,68 + 1$		
$0,68 \cdot 2 = 0,36 + 1$		
$0,36 \cdot 2 = 0,72 + 0$		
$0,72 \cdot 2 = 0,44 + 1$		

$\rightarrow 0,00111101_B$

LSB!

Minifloat-Format

- Es gibt verschiedene Minifloat-Formate



- half precision (16Bit)
- s: sign bit
- E: transformierter Exp. $E = e + \text{bias}$ (bias = 15 für 16Bit)
- f: Fraction (Bruch nach Normalisierung)
 - 1 vorm Komma wird weggelassen

Runden

- Warum? Nur begrenzte Nachkommastellen
- Wir behandeln Runden nicht

ROrg:

- round towards zero (Rundungsregel)
- Hintere Bits abschneiden
- In Realität meist anders



Addition

1. Normalisieren
 2. Kleineren Exponenten an größeren anpassen
 3. Mantissen addieren
- Nach jedem Schritt an 16Bit-Format anpassen (10 Nachkommastellen)

Multiplikation

1. Exponenten nicht anpassen
2. Mantisse multiplizieren
3. Exponenten addieren
4. Normalisieren

