Grundlagen der Rechnerarchitektur

Tim Luchterhand, Paul Nykiel (Abgabegruppe 117)

1. Februar 2019

Hinweis: Jede Zahl, bei der keine Basis spezifiziert ist, ist im 10er System zu interpretieren, sofern nicht anders angegeben.

1 Zahlendarstellung

1.1

1.1.1 Gleitkommazahlen

Gleitkommazahlen sind flexibel im Bezug auf die Größenordnung der Zahl. Das heißt man kann sehr große und sehr kleine Zahlen gut darstellen und ohne zu wissen, in welcher Größenordnung sich die Zahlen befinden.

Mathematische Operationen auf Gleitkommazahlen sind vergleichsweise aufwendig und erfordern häufig ein spezielles Rechenwerk.

1.1.2 Festkommazahlen

Mathematische Operationen auf Festkommazahlen lassen sich bereits mit einem einfachen Ganzzahlrechenwerk durchführen.

Bei Festkommazahlen muss im Vorraus bekannt sein, in welcher Größenordnung sich die Zahl befinden wird, da jeweils eine feste Anzahl Bits für Vor- und Nachkommastellen reserviert sind.

1.2

$$\frac{43}{30} = 1 + \frac{13}{30} = 1 + \frac{1}{2 + \frac{4}{13}} = 1 + \frac{1}{2 + \frac{1}{3+\frac{1}{4}}}$$

$$\frac{55}{19} = 2 + \frac{17}{19} = 2 + \frac{1}{1 + \frac{2}{17}} = 2 + \frac{1}{1 + \frac{1}{8+\frac{1}{2}}}$$

2 Gleitkommaarithmetik

2.3

(a) Größte Zahl: Mantisse Maximal ($22222_3 = 242$), Exponent Maximal ($222_3 - 13 = 13$) und Vorzeichen positiv.

$$z_{\max} = +242 \cdot 3^{13} = 385826166$$

Kleinste Zahl: wie größte Zahl nur mit negativem Vorzeichen

$$z_{\min} = -z_{\max} = -385826166$$

(b) i.

$$2.25 = 1.125 \cdot 2^1 = (-1)^0 \cdot 2^{(4-3)} \cdot (1 + 2^{-3}) = 0\ 100\ 00100_{\text{Gleitkomma}}$$

$$5.625 = 1.40625 \cdot 2^2 = (-1)^0 \cdot 2^{(5-3)} \cdot (1+2^{-2}+2^{-3}+2^{-5}) = 0\ 101\ 01101_{\text{Gleitkomma}}$$

Folgende Rechnung im Gleitkommaformat:

$$0\,100\,00100 + 0\,101\,01101 \overset{\text{Exponent angleichen}}{=} 0\,101\,10010 + 0\,100\,01101 = 0\,101\,01111$$

Da die Mantisse nur 5 Bit lang ist kann die Zahl nicht ohne Fehler dargestellt werden.

ii.

$$2.5 = 1.25 \cdot 2^1 = (-1)^0 \cdot 2^{(4-3)} \cdot (1 + 2^{-2}) = 0\ 100\ 01000_{\text{Gleitkomma}}$$

$$16.8 = 1.05 \cdot 2^4 \approx (-1)^0 \cdot 2^{(7-3)} \cdot (1+2^{-5}+2^{-6}+2^{-9}+\ldots) \approx 0\ 111\ 00001_{\text{Gleitkomma}}$$

Fünf Bit Mantisse reichen hier nicht aus, um alle Nachkommastellen darzustellen.

Folgende Rechnung im Gleitkommaformat:

$$0\ 100\ 00100 \cdot 0\ 111\ 00001 = 0\ 000\ 10101$$

Die Exponenten werden addiert und dann wieder normiert (Bias abziehen). Dabei kommt es zu einem Overflow, der Exponent ist eigentlich 1000_2, es werden aber nur die unteren drei Bits gespeichert. Bei IEEE-754 gibt es für diesen Fall den Spezialwert infinity.

iii.

$$-6.375 = -1 \cdot 2^2 \cdot 1.59375 = (-1)^1 \cdot 2^{(5-3)} \cdot (1+2^{-1}+2^{-4}+2^{-5}) = 1\ 101\ 10011_{\text{Gleitkomma}}$$
$$4.125 = 1.03125 \cdot 2^2 = (-1)^0 \cdot 2^{(5-3)} \cdot (1+2^{-5}) = 0\ 101\ 00001_{\text{Gleitkomma}}$$

Folgende Rechnung im Gleitkommaformat:

$$1\ 101\ 10011 + 0\ 101\ 00001 = 1\ 100\ 00100$$

3 Festkommarithmetik

3.4

(a) (i)

$$8.875 = 2^3 + 2^{-1} + 2^{-2} + 2^{-3} = 01000,11100_2$$

(ii)

$$31.125 = 2^4 + 2^3 + 2^2 + 2^1 + 2^0 + 2^{-3} = 11111,00100_2$$

(b) (i)

$$-5.625 = -(2^2 + 2^0 + 2^{-1} + 2^{-3}) = -00101,10100_2$$
$$7.65625 = 2^2 + 2^1 + 2^0 + 2^{-1} + 2^{-3} + 2^{-5} = 00111,10101_2$$

Folgende Rechnung im binären Festkommaformat:

$$-00101,10100 + 00111,10101 = 10,00001$$

(ii)

$$7.03125 = 2^2 + 2^1 + 2^0 + 2^{-5} = 00111,00001_2$$
$$-(2.03125) = -(2^1 + 2^{-5}) = -00010,00001_2$$

Folgende Rechnung im binären Festkommaformat:

$$00111,00001 - (-00010,00001) = 00111,00001 + 00010,00001 = 01001,00010$$

(iii)

$$-5.15625 = -(2^2 + 2^1 + 2^{-3} + 2^{-5}) = -00110,00101_2$$
$$0.25 = 2^{-2} = 00000,01000_2$$

Folgende Rechnung im binären Festkommaformat:

$$-00110,00101_2 : 00000,01000_2 = -11000,10100_2$$

(iv)

$$4.0 = 2^2 = 00100,00000_2$$
$$-2.0625 = -(2^1 + 2^{-4}) = -00010,00010_2$$

Folgende Rechung im binären Festkommaformat:

$$00100,00000_2 \cdot -00010,00010_2 = -01000,01000$$