technische universität
dortmund

Fachprojekt

Floating Point Unit

Leon Bartmann

28. September 2022

Inhaltsverzeichnis

1 Floating-Point-Standard **3**
1.1 Operationen 3
1.2 Rundungsmodus 3
1.3 Exceptions 4
1.4 Sonderfälle 4

2 Aufbau **5**
2.1 Tests 5

3 Fazit **6**

Literaturverzeichnis **7**

1 Floating-Point-Standard

In diesem Repository wurde eine Floating-Point Arithmetic Unit umgesetzt, wie sie in einer Floating-Point Unit (FPU) verwendet wird. In bestehenden Prozessoren kann diese FPU als Koprozessor eingebunden werden, um die Funktionalität um das Rechnen mit Fließkommazahlen zu erweitern.

Dieses Projekt folgt dem IEEE 754 Standard [1] und ermöglicht Berechnungen mit Single-precision Zahlen. Diese bestehen aus einem Vorzeichenbit s, einem Exponenten e mit 8 bits und einer Mantisse m mit 23 bits.

- der Exponent ist mit einem Bias von 127 dargestellt und besitzt somit einen Wertebereich von -127 bis 128

- die Mantisse besteht nur aus Nachkommastellen und es wird implizit angenommen, dass davor eine weitere 1 steht

In dieser Darstellung existieren einige Sonderfälle, die in der folgenden Tabelle aufgeführt sind.

s	e	m	Wert
0	00000000	00000000000000000000000	+0
1	00000000	00000000000000000000000	-0
0	11111111	00000000000000000000000	$+\infty$
1	11111111	00000000000000000000000	$-\infty$
*	11111111	$\neq$ 00000000000000000000000	NaN

Der repräsentierte Wert einer Fließkommazahl, die nicht einem der Sonderfälle entspricht und deren Exponent in $[-126, 127]$ liegt, ist:

$$(-1)^s \cdot 2^e \cdot (1 + m)$$

1.1 Operationen

Es wurden die Operationen Addition, Subtraktion und Multiplikation umgesetzt. Diese werden durch den *op code* repräsentiert:

op	Operation
00	Addition
01	Subtraktion
10	Multiplikation

1.2 Rundungsmodus

Bei allen Operationen können Ungenauigkeiten auftreten, weil die 23 bits der Mantisse nicht ausreichen, um das Ergebnis exakt darzustellen. In dem Fall existieren

hinter dem darstellbaren Teil der Mantisse noch weitere bits, die zwar nicht darstellbar sind, aber beim Runden des Ergebnisses mit berücksichtigt werden können.

$$
\begin{aligned}
&0\ 01111111\ 00000000000000000000000 \\
+\ &0\ 01111111\ 00000000000000000000001 \\
=\ &0\ 10000000\ 000000000000000000000001 \\
&0\ 10000000\ 00000000000000000000000 \qquad \text{(Abrunden)} \\
&0\ 10000000\ 00000000000000000000001 \qquad \text{(Aufrunden)}
\end{aligned}
$$

Der IEEE 754 Standard sieht fünf Rundungsmodi vor, welche anhand von unterschiedlichen Kriterien die Rundung vorgeben.

rnd	Bedeutung
000	Round to nearest, ties to even
001	Round to nearest, ties away from zero
010	Round toward 0
011	Round toward $+\infty$
011	Round toward $-\infty$

1.3 Exceptions

Bei der Berechnung können fünf Typen von Fehlern auftreten. Jeder Fehlertyp wird dabei durch ein eigenes bit repräsentiert.

bit	Bezeichnung	Bedeutung
1.	Invalid operation	(nicht relevant)
2.	Division by zero	(nicht relevant)
3.	Overflow	das Ergebnis ist größer als die größte darstellbare Zahl
4.	Underflow	das Ergebnis ist kleiner als die kleinste darstellbare Zahl
5.	Inexact	das Ergebnis ist gerundet/nicht exakt darstellbar

1.4 Sonderfälle

Ist mindestens einer der beiden Operanden NaN, dann ist das Ergebnis ebenfalls NaN. Alle weiteren Sonderfälle sind im Folgenden aufgeführt.

Addition

$+$	v_2	0	∞	$-\infty$
v_1	v_3	v_1	∞	$-\infty$
0	v_2	0	∞	$-\infty$
∞	∞	∞	∞	NaN
$-\infty$	$-\infty$	$-\infty$	NaN	$-\infty$

Subtraktion

-	v_2	0	∞	$-\infty$
v_1	v_3	v_1	$-\infty$	∞
0	$-v_2$	0	$-\infty$	∞
∞	∞	∞	NaN	∞
$-\infty$	$-\infty$	$-\infty$	$-\infty$	NaN

Multiplikation

$\times$	v_2	0	∞	$-\infty$
v_1	v_3	0	∞	$-\infty$
0	0	0	NaN	NaN
∞	∞	NaN	∞	$-\infty$
$-\infty$	$-\infty$	NaN	$-\infty$	∞

2 Aufbau

Die Arithmetic Unit hat die folgenden In- und Outputs:

Name	Bedeutung	Bits
num_a	erster Operand	32
num_b	zweiter Operand	32
rnd	Rundungsmodus	3
num_out	Ergebnis	32
exc	Fehler	5

Die oberste Komponente *au* ist ein Demultiplexer und gibt immer das Ergebnis der gewählten Rechenoperation zurück. Die Komponenten *add* und *sub* reichen die Eingaben an die jeweils andere Operation weiter, falls die Vorzeichenbits der Operanden unterschiedlich sind. Zu jeder Operation existiert des weiteren eine übergeordnete Komponente, die Sonderfälle in der Berechnung abfängt. Diese gibt nur dann das Ergebnis der untergeordneten Einheit aus, wenn kein Sonderfall vorliegt. Der gesamte Aufbau der AU ist in Abbildung 1 dargestellt.

2.1 Tests

num_a	1 01111111 00000000000000000000000	= -1.0
num_b	0 01111110 00000000000000000000000	= 0.5
+	1 01111110 00000000000000000000000	= -0.5
−	1 01111111 10000000000000000000000	= -1.5
×	1 01111110 00000000000000000000000	= -0.5

num_a	0 10000001 01000000000000000000000	= 5
num_b	0 10000001 01000000000000000000000	= 5
+	0 10000010 01000000000000000000000	= 10
−	0 00000000 00000000000000000000000	= 0
×	0 10000011 10010000000000000000000	= 25

Neben diesen Testfällen, welche auch in der Datei *src/test.bat* ausgeführt werden, wurden alle Sonderfälle, Overflows und Underflows getestet.

3 Fazit

Die Arithmetic Unit kann Berechnungen mit 32-bit Fließkommazahlen korrekt durchführen. Alle Rundungsmodi und Exceptions wurden umgesetzt und getestet. Mit der AU ist somit der zentrale Bestandteil einer FPU fertig implementiert und er könnte zukünftig in einem Koprozessor eingesetzt werden.

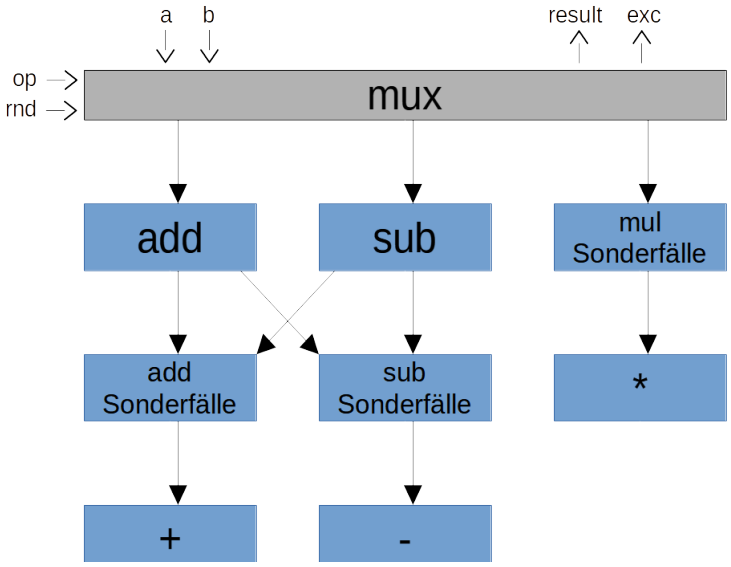

Abbildung 1: Aufbau der AU

Literatur

[1] IEEE 754. `https://codedocs.org/what-is/ieee-754`. Abgerufen am 26.09.2022.