HERAKLIT-Fallstudie: Paralleladdierer

Peter Fettke[1,2][0000−0002−0624−4431] und Wolfgang Reisig[3][0000−0002−7026−2810]

[1] Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI), Saarbrücken, Deutschland
peter.fettke@dfki.de
[2] Universität des Saarlandes, Saarbrücken, Deutschland
[3] Humboldt-Universität zu Berlin, Berlin, Deutschland
reisig@informatik.hu-berlin.de

Abstract. This small case study extends the example of modelling the written addition of two three-digit natural numbers in decimal notation from the HERAKLIT short case study [2]. There, a half adder and a full adder are constructed as HERAKLIT modules and composed appropriately for the task mentioned. Here we show a scheme for the parallel addition of 10 numbers of any size.

Zusammenfassung. Diese kleine Fallstudie erweitert das Beispiel der Modellierung des schriftlichen Addierens zweier dreistelliger natürlicher Zahlen in Dezimaldarstellung aus der HERAKLIT-Fallstudie [2]. Dort werden ein Halbaddierer und ein Volladdierer als HERAKLIT-Module konstruiert und für die genannte Aufgabe geeignet komponiert. Hier zeigen wir ein Schema zur parallelen Addition von 10 Zahlen beliebiger Stelligkeit.

1 Klassische Addition

Die HERAKLIT-Fallstudie [2] beschreibt das schriftliche Addieren zweier dreistelliger natürlicher Zahlen in Dezimaldarstellung. Die zu addierenden Zahlen des Halbaddierers liegen zwischen 0 und 9, sind also dezimal als Ziffern darstellbar. Ihre Addition ergibt einen Übertrag, 0 oder 1. Das Modul *Halbaddierer* in Abbildung 1 modelliert diese Addition. Für mehrstellige Zahlen muss die Addition der $i+1$-ten Stelle noch den Übertrag der i-ten Stelle berücksichtigen. Das macht das Modul *Volladdierer* in Abbildung 2. Das komponierte Modul $Halbaddierer \bullet Volladdierer$ in Abbildung 3 addiert zweistellige Zahlen. Seine rechte Schnittstelle enthält zwei Gates mit dem Label X und zwei Gates mit dem Label Y. Seine linke Schnittstelle enthält zwei Gates mit dem Label

zu zitieren als: FETTKE, P.; REISIG, W.: HERAKLIT-*Fallstudie: Paralleladdierer*. 2020. – HERAKLIT–Arbeitspapier, v1, 8. November 2020, http://www.heraklit.org

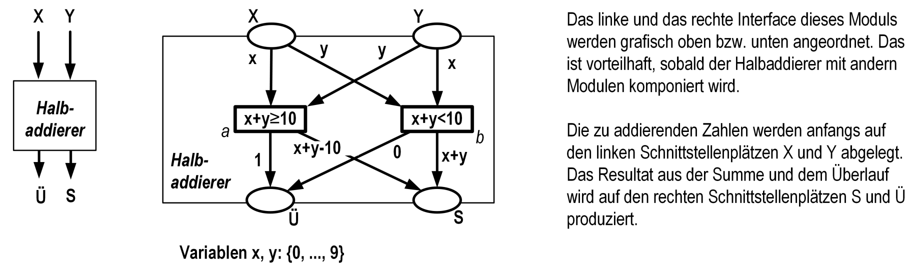


Abb. 1: Der Halbaddierer

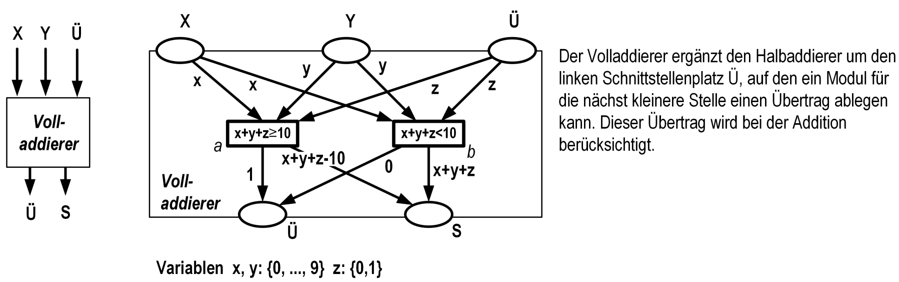


Abb. 2: Der Volladdierer

S. Für praktische Zwecke wären allerdings intuitivere, paarweise disjunkte Labels zweckmäßiger. Das ist erreichbar mit sehr speziellen Modulen, die lediglich Labels an Gates umbenennen. Solche Umbenenner-Module notieren wir oft in verschiedenen Kurzschreibweisen, die Abbildung 4 zeigt.

Damit konstruieren wir nun eine Variante des $Ripple - Carry\ Adders$ ([1], umgangssprachlich: handschriftliches Addieren wie Kinder es lernen) für n-stellige Dezimalzahlen induktiv als HERAKLIT-Modul, wie in Abbildung 5 dargestellt: für $n = 1$ addiert das Modul Addierer $Z10$ einstellige Zahlen, algebraisch geschrieben als:

$$Z10 := (E1 \rhd X) \bullet (E2 \rhd Y) \bullet Halbaddierer \bullet (S \rhd E).$$

Dabei steht $E1$, $E2$ und E für $Einer$ und Z für $Zehner$. Entsprechend sind die Module Addierer $Z100$ und Addierer $Z1000$ zur Addition zwei- und dreistelliger Zahlen in Abbildung 5 definiert. Ihre algebraische Darstellung ist:

$$Z100 := Z10 \bullet (Z1 \rhd X) \bullet (Z2 \rhd Y) \bullet Volladdierer \bullet (S \rhd Z)$$
$$Z1000 := Z100 \bullet (H1 \rhd X) \bullet (H2 \rhd Y) \bullet Volladdierer \bullet (S \rhd H).$$

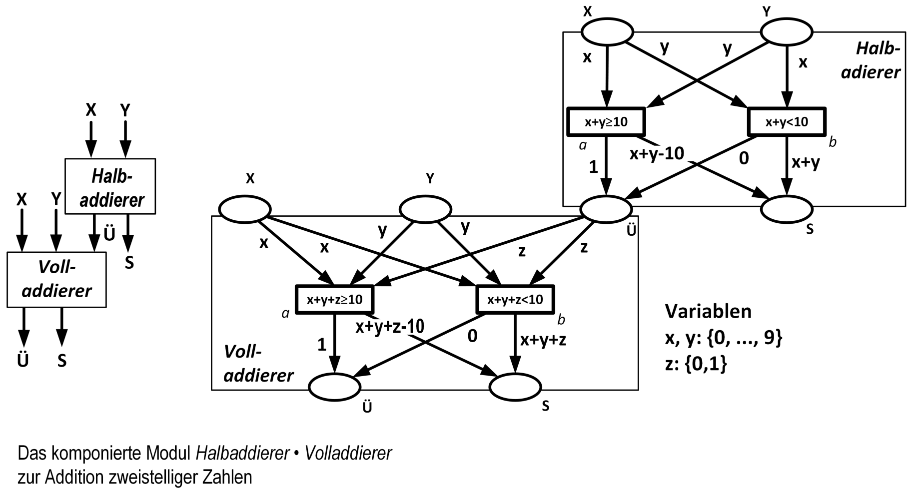


Abb. 3: Die Komposition

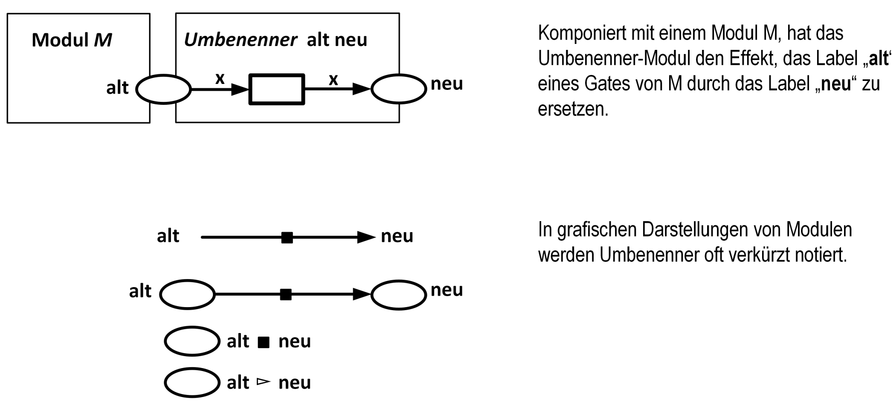


Abb. 4: Umbenennung

Der Weg zur Addition zweier Zahlen beliebiger Stelligkeit n ist damit offensichtlich: Der Addierer für $n-1$– stellige Zahlen wird links um einen weiteren Volladdierer ergänzt.

2 Parallele Addition

Wir verallgemeinern das Prinzip des $Ripple - Carry\ Adders$ und konstruieren einen Addierer, der bis zu 10 n-stellige Zahlen der Form

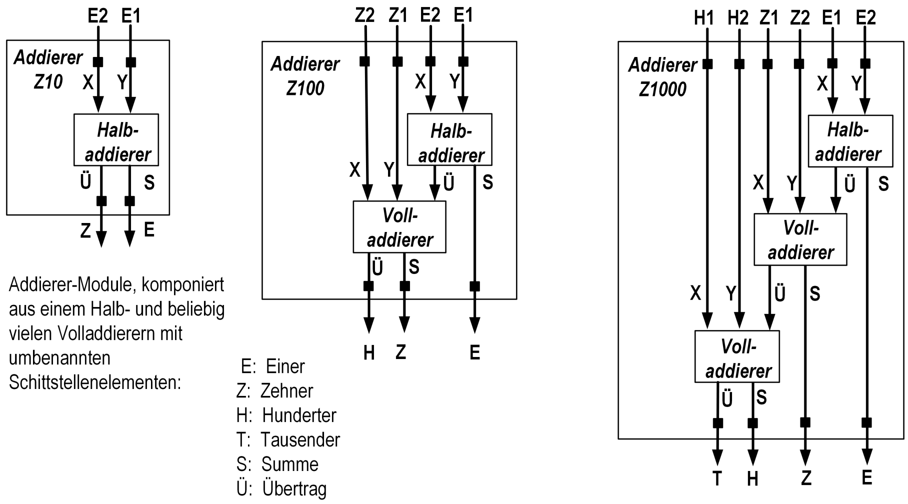


Abb. 5: Addierer-Module

$$
\begin{array}{l}
x_{1,n}, \ldots, x_{1,2}, x_{1,1} \\
x_{2,n}, \ldots, x_{2,2}, x_{2,1} \\
\quad \ldots \\
x_{10,n}, \ldots, x_{10,2}, x_{10,1} \ mit\ 0 < x_{i,j} < 9
\end{array}
$$

parallel addiert. Die Beschränkung auf 10 Zahlen garantiert, dass Überträge einstellig bleiben. Abbildung 6 zeigt den Addierer als Heraklit-Modul mit seinen Ein- und Ausgaben in der abstraktesten Form.

Eine ausführbare Form des Addierer-Moduls kann aus Instanzen eines einzigen ausführbaren Moduls, des inkrementellen Addierers, komponiert werden, wie ihn die Abbildung 7 zeigt.

Die grundlegende Idee des Addierers: Zunächst werden die Ziffern jeder einzelnen Spalte j, also jeder einzelne Stelligkeit, für sich addiert. Dafür werden 10 inkrementelle Addierer so angeordnet, wie Abbildung 8 zeigt: Dabei entsteht natürlich für jedes j ein Überlauf. Diese Überläufe werden dann – wie üblich von rechts nach links – (in Dezimalschreibweise) verrechnet. Abbildung 9 zeigt das gesamte Modul mit seinen Besonderheiten für die äußeren beiden Spalten. Verfeinert man jedes abstrakte Exemplar des inkrementellen Addierers durch seine ausführbare Fassung, entsteht das ausführbare Modul aus Abbildung10.

3 Nebenläufigkeit und verteilter Ablauf

Die vorherigen Abbildungen zeigen deutlich, wie die 10 Ziffern jeder einzelnen Stelligkeit addiert werden, unabhängig von den andern Stelligkeiten. Erst die Re-

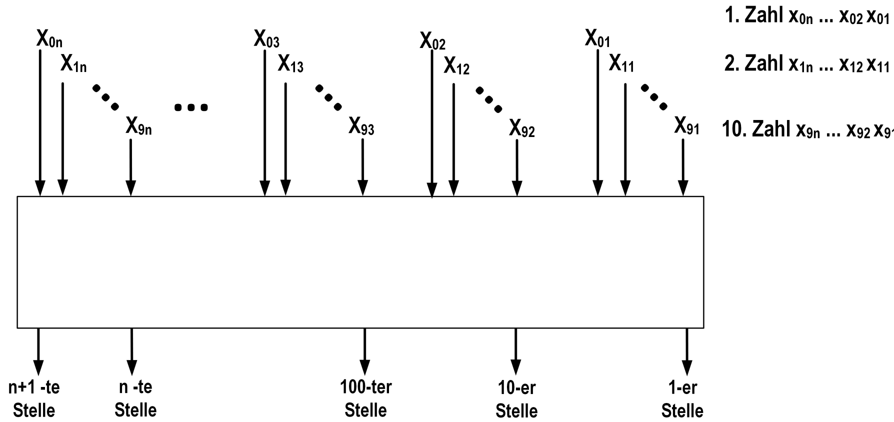
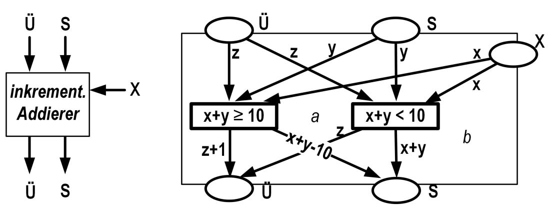


Abb. 6: Abstraktes Modul des Paralleladdierers



Die linke Schnittstelle enthält die Eingabe einer bislang berechneten Zahl, also die aktuellen Resultat-Summe S mit dem aktuellen Überhang Ü, sowie die zu addierende Zahl X. Die rechte Schnittstelle ist die Ausgabe des Resultats der Addition, bestehend aus der neuen Resultat-Summe S, und dem neu entstandenen Überhang Ü.

Um die übliche Darstellung von Dezimalzahlen und ihrer handschriftlichen Addition beizubehalten, sind die Gates der linken Schnittstelle deshalb oben (und sogar rechts) angeordnet, die Gates der rechten Schnittstell liegt unten.

Abb. 7: Der inkrementelle Addierer

sultate werden miteinander verbunden, indem die Überträge der Additionen von rechts nach links verrechnet werden. Besonders deutlich wird das nebenläufige Rechnen am Beispiel der konkreten Rechnung aus Abbildung 11.

Wenn tatsächlich für jede $x_{i,j}$ ein eigener Prozessor P_{ij} verfügbar ist, kann das Verfahren auch wiederholt angewendet werden. Beispielsweise kann man 100 $n-1$-stellige Zahlen addieren, indem man zunächst 10 Gruppen von je 10 Zahlen addiert und auf die 10 n–stelligen Resultate dann noch einmal den Paralleladdierer anwendet. Dabei kann P_{ij} mit der Bearbeitung der ersten Zahl einer neuen Gruppe beginnen, unmittelbar nachdem P_{ij} seinen Beitrag zur letzten Zahl der aktuellen Gruppe geliefert hat.

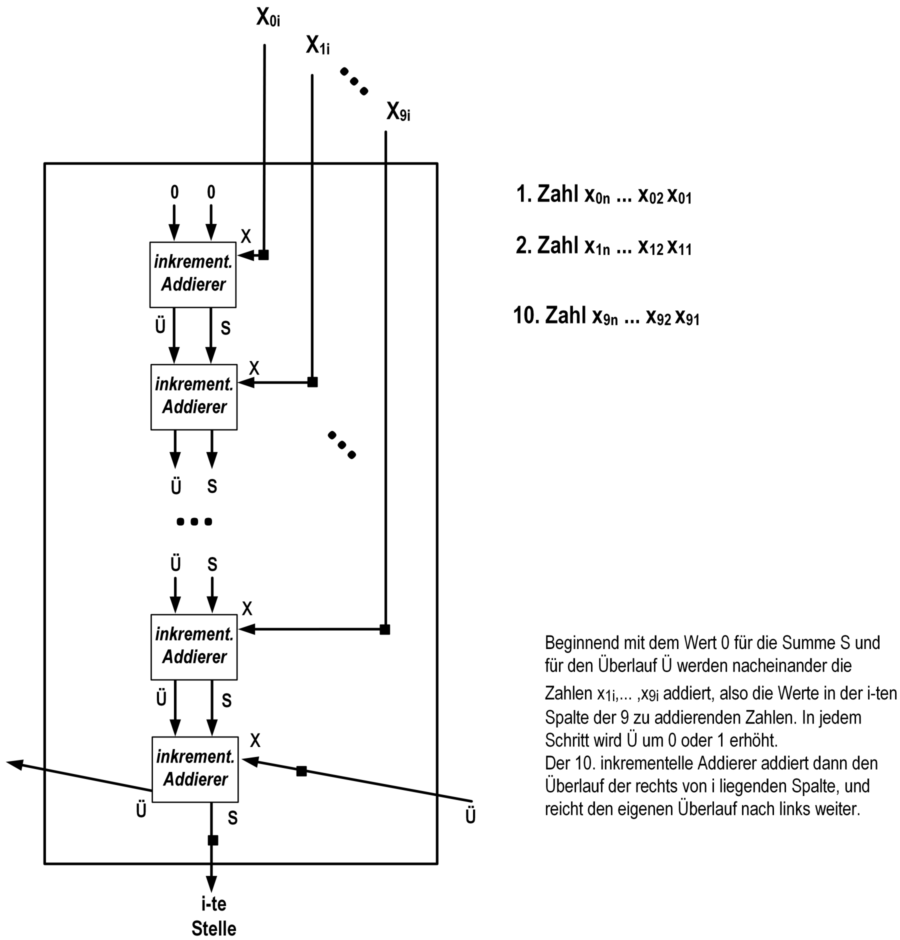


Abb. 8: eine Spalte

4 Abschließende Bemerkung

Dieses Beispiel ist inhaltlich keine typische Anwendung von Heraklit, illustriert jedoch besonders deutlich die Kombinierbarkeit von Abstraktion, Umbenennung und Wiederverwendung einzelner Module.

Literatur

1. Burgess, N. : Fast Ripple-Carry Adders in Standard-Cell CMOS VLSI. In: *20th IEEE Symposium on Computer Arithmetic*, 2020, S. 103–111
2. Fettke, P. ; Reisig, W. : Heraklit-*Fallstudie: Addierer*. 2020. – Heraklit-Arbeitspapier, v1, 8. November 2020, `http://www.heraklit.org`

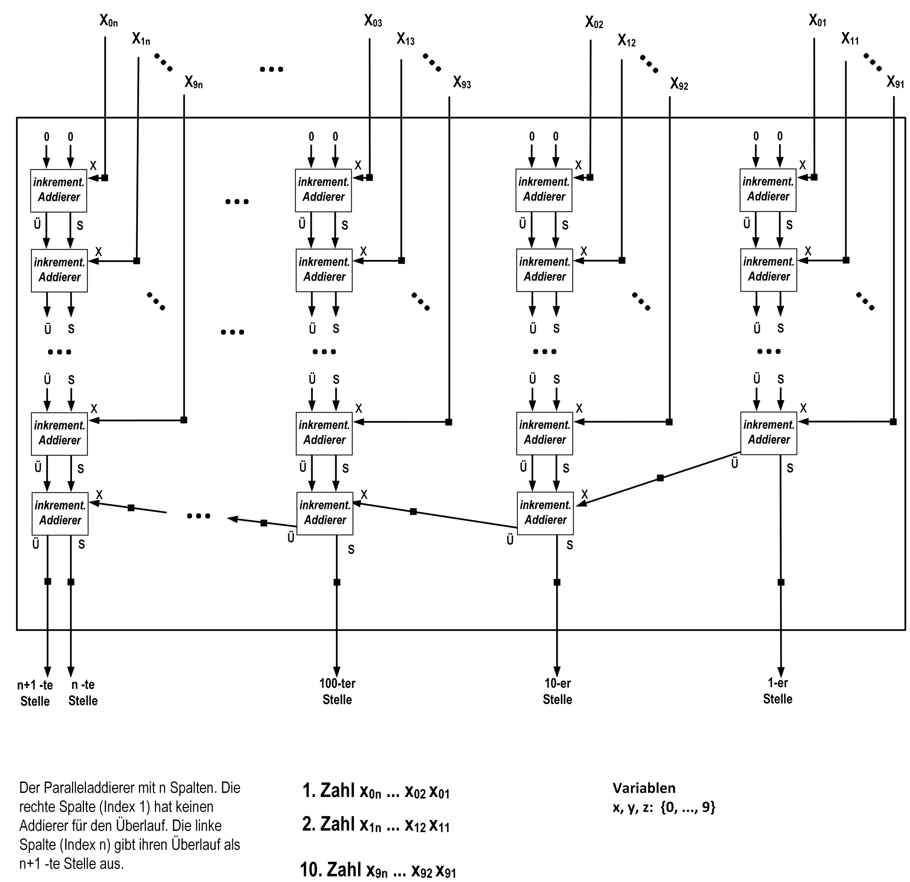


Abb. 9: System mit inkrementellen Addierer

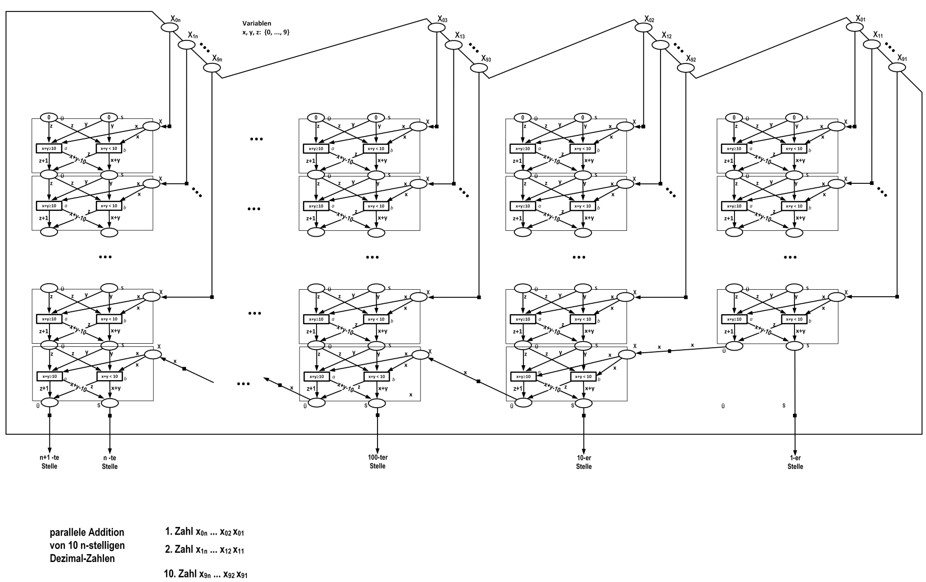


Abb. 10: ausführbarer Paralleladdierer

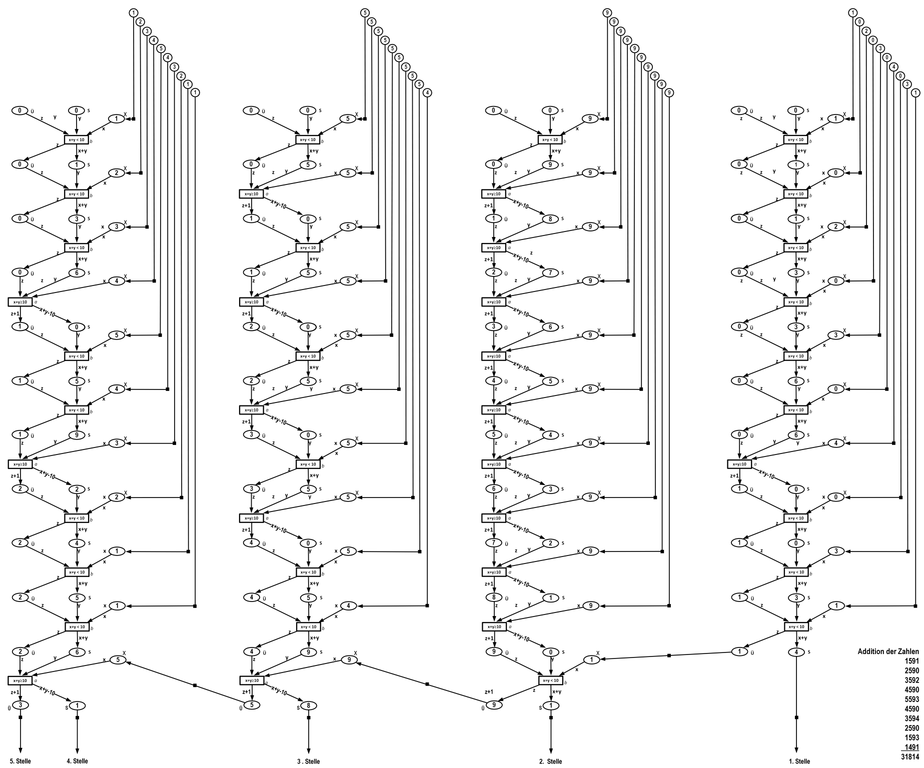


Abb. 11: eine Beispielrechnung als verteilter Ablauf