

Die Prozessorarchitektur der Rechenmaschine Z1

Raúl Rojas, Julian Röder, Hai Nguyen

FB Mathematik und Informatik

Freie Universität Berlin

Arnimallee 7

14195 Berlin

Email: raul.rojas@fu-berlin.de

Zusammenfassung

Dieser Aufsatz beschreibt zum ersten Mal die Datenflussarchitektur des Prozessors der Z1, der ersten von Konrad Zuse gebauten Rechenmaschine (1936-1938). Die Struktur wurde aus den Designunterlagen der im Jahr 1989 fertiggestellten Rekonstruktion der Z1 herausdestilliert. Die Z1 war ein Fließkommarechner. Der Prozessor enthält zwei Einheiten: eine arithmetisch-logische Einheit für die Behandlung der Exponenten und eine zweite für die Mantissen. Der Prozessor ähnelt stark dem der Z3, ist aber einfacher, da an einem großen Shifter und andere Datenflussoptionen verzichtet wurde. Der Prozessor der Z1 brauchte mehr Zyklen für die Grundrechenarten als die Z3, konnte aber mit wenigen logischen Komponenten auskommen, ein wichtiges Designkriterium für eine mechanische Rechenmaschine.

Abstract

This paper describes the dataflow architecture of the processor of the Z1, the first computer designed and built by Konrad Zuse (between 1936 and 1938). The structure was recovered from an analysis of the design blueprints drawn for a modern reconstruction of the Z1 which was finished in 1989. The Z1 was a floating-point machine: the processor consists of two sections, one containing the ALU for the exponent, a second with an ALU for the mantissas. The processor is fairly similar to the processor of the Z3, but is simpler since a large shifter and some other dataflow options were sidestepped. The processor of the Z1 needed more cycles than the Z3 for the basic arithmetic operations but could be built with less logical components, an important design feature for a mechanical machine.

Stichworte: Geschichte des Computers, Konrad Zuse, Rechenmaschine Z1, Mikroprogramm.

1 Einleitung

Vor mittlerweile mehr als 20 Jahren habe ich die Rekonstruktion der Rechenmaschine Z1 im Museum für Verkehr und Technik in Berlin nachdenklich bewundert. Das Ungetüm hat mich damals beeindruckt. Sollte dies wirklich der erste Computer der Welt sein? Eine detaillierte Beschreibung der Maschine war allerdings nirgends zu finden. Wer vor der Maschine steht, kapituliert auch heute noch beim Versuch, die Funktionsweise von mehreren Hundert Blechen, Stangen und Rädern allein durch Anschauung verstehen zu wollen. Die Z1 wirkt wie ein versteinertes unergründliches Uhrwerk.

In diesem Aufsatz wird zum ersten Mal die Architektur des Prozessors der Rechenmaschine Z1 beschrieben. Die grobe modulare Architektur der Z1 ist seit den achtziger Jahren bekannt, nicht aber die Detailstruktur des Prozessors. Bei der Z1 handelt es sich um eine von-Neumann-Architektur mit getrennten Prozessor- und Speichereinheiten, wobei das Programm in einem Lochstreifen untergebracht wird. Die Blockarchitektur der Z1 ist ähnlich der der Z3, eine Maschine von 1941, die anhand ihrer Patentanmeldung bereits ausführlich beschrieben worden ist [1,2].

Die Z1 wurde in den Jahren 1936 bis 1938 von Konrad Zuse in Berlin entwickelt [3]. Sie war die erste von vier Rechenmaschinen, die Zuse bis zum Ende des Zweiten Weltkrieges bauen würde. Der Z1 gebührt die Ehre, zusammen mit der Rechenmaschine von Charles Babbage, einer der beiden einzigen vollmechanischen Computer der Welt zu sein. Die Z1 wurde allerdings zu Ende gebaut, etwas das mit dem anspruchsvollen Design von Charles Babbage nicht gelingen konnte. Babbage war seiner Zeit einfach zu weit voraus. Die Z1 wurde im Krieg zerstört, aber eine neue Z1 wurde von 1986 bis 1989 von Konrad Zuse für das Berliner Museum für Verkehr und Technik gebaut. Von dieser Rekonstruktion gibt es zahlreiche Blaupausen, die allerdings unvollständig und ungeordnet vorliegen und kaum Lesehinweise enthalten. Für das Verständnis des Prozessors der Z1 kann man allerdings nur auf diese sehr schwer interpretierbaren Unterlagen zugreifen. Es ist fast das Einzige, was Zuse über die Z1 hinterlassen hat. Es ist kaum zu glauben: Beim ersten Prototyp dokumentierte Zuse seine Z1 sehr spärlich, beim zweiten, fünfzig Jahre später, auch nicht wirklich. Es ist, als ob ihm das vollendete Werk als Selbstdokumentation ausgereicht hätte. Nur für den Speicher der Z1 liegt eine gute Beschreibung von Schweier und Saupe [4], den beiden Helfern von Zuse bei der Rekonstruktion der Z1, vor.

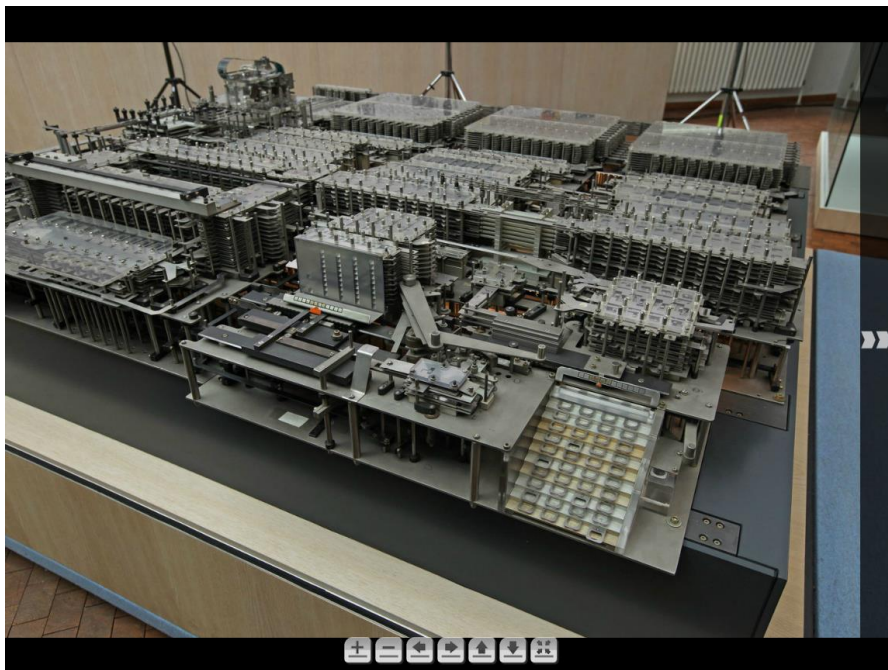


Abb. 1: Ein Screenshot der Dokumentation der Z1 im Zuse Internet Archiv. Der Benutzer kann den Blickwinkel rund um die Z1 ändern und vergrößern. Das System verwaltet Tausende von Bildern, um panoramische Navigation zu ermöglichen.

Diese Dokumentationslücke wird hier für den Prozessor der Z1 geschlossen. Die Blaupausen der Z1 liegen mittlerweile sowohl in Berlin als auch im Archiv des Deutschen Museums in München vor. Gescannte Fotokopien sind von einer Arbeitsgruppe des Deutschen Museums und der FU Berlin ins Internet gestellt worden [5]. Zusammen mit Tausenden von photographischen Ansichten der rekonstruierten Z1 (Abb. 1) enthält die Webseite die Blaupausen, sowie eine Simulation eines mechanischen Addierwerkes, das von Zuse patentiert wurde [6] (die Schaltung des patentierten Addierwerkes entspricht jedoch nicht der Schaltung der Z1 in Berlin). Anhand dieser Unterlagen, insbesondere der Beschreibung der Synchronisation der Befehle der Z1, kann die Struktur des Prozessors entschlüsselt werden. Wie wir hier zeigen, ist der Prozessor der Z1 in der baulichen Ausführung elegant und bündig gehalten. Als Bauteile verwendete Zuse lineare binäre mechanische Komponenten. Die zwei Binärzustände werden durch die Hin- und Herbewegung von Platten bzw. Stangen erzeugt. Statt Zahnräder zu benutzen (die eventuell bis zu zehn oder mehr Zustände anzeigen könnten), entschied Zuse relativ früh, eine echte binäre Maschine zu bauen und dafür seine Konstruktionen mit beweglichen Platten zu gestalten. Solche Binärelemente sind wiederholt, auch von Zuse, beschrieben worden [3]. Die Z1 ist mit bis zu 12 mechanischen Ebenen gebaut worden, d.h. logische Schaltglieder wurden überlagert um Platz zu sparen. Drei Ebenen von Stangen unter der Maschine transportieren die Synchronisationspulse für die vier „Takte“ in einem Zyklus. Hier brauchen wir nicht in die mechanischen Details einzugehen. Es reicht, wenn wir auf einer abstrakten Ebene bleiben und den Datenfluss in der Maschine betrachten. Eine viel ausführlichere Beschreibung der Z1 wird demnächst auf Englisch erscheinen.

2 Blockarchitektur der Z1

Abb. 2 zeigt die globale Architektur der Z1. Die erste Haupttrennung ist die von Speicher (obere Hälfte) und Prozessor (untere Hälfte). Die Z1 wurde für Rechnungen mit Fließkommazahlen der Form $m \times 2^a$ ausgelegt. Die Zahl m ist die Mantisse -- sie ist der Form „1.bb..bb“, wobei jedes b eine Null oder eine 1 ist. Die Zahl a ist der Exponent der binären „wissenschaftlichen Notation“. Eine Mantisse, die mit einer 1 vor dem Punkt anfängt, nennt man „normalisiert“. Im Speicher der Z1 wird der Exponent einer Fließkommazahl mit 7 Bits in Zweierkomplement-Darstellung festgehalten (womit der binäre Exponent zwischen -64 und $+63$ laufen kann). Die Mantisse wird mit 16 Bits gespeichert, wobei die führende „1“ nicht mitgespeichert zu werden braucht. Sowohl sehr große als auch sehr kleine Zahlen können mit dieser von Zuse genannten „halblogarithmischen Notation“ bearbeitet werden. Das Vorzeichen der Fließkommazahl wird getrennt neben dem Exponenten für jede Zahl im Speicherwerk festgehalten.

Zuse hat für die Rekonstruktion der Z1 diese Anzahl von Bits für Vorzeichen+Exponent und für die Mantisse gewählt, da auf diese Weise drei Module á 8 Bits über acht Ebenen gebaut werden könnten. Dies verringerte die Anzahl der unterschiedliche geformten Bauteile. Insgesamt konnten damit 64 Fließkommazahlen gespeichert werden (im Original von 1938 gab es nur 16 Zahlen im Speicher). Für den Fall, dass eine Zahl „Null“ oder „Unendlich“ ist, verwendete Zuse in der Z3 spezielle Werte des Exponenten, die dann von der Logik als Sonderfälle bearbeitet wurden. Der Leser sei auf [1] verwiesen, wo Zuses Fließkommaformat für die Z3 ausführlich beschrieben wird. Zuses Format ist der modernen IEEE-Fließkomma-Notation sehr ähnlich. Allerdings enthält die rekonstruierte Z1 in Berlin die notwendigen Ausnahmeschaltungen nicht, und diese waren beim Original auch nicht eingebaut worden. Die Z1 konnte nicht mit Null rechnen! Dieses Manko war Zuse natürlich bewusst, der in der Z1 nur ein Prototyp und „proof of concept“ sah.

Abb. 2 zeigt, dass der binäre Befehlscode und die Speicheradresse dem Lochstreifen entnommen werden. Die Befehle Laden/Speichern werden mit 2-Bit-Befehlscode und 6-Bit-Adressen kodiert. Die vier arithmetischen Befehle Addition, Subtraktion, Multiplikation und Division werden mit mehr als zwei Bits kodiert. Die Vorzeichen der beiden Operanden im Prozessor werden vor der Addition bzw. Subtraktion inspiziert und Anpassungen der Befehle werden vor der Ausführung eingeplant: Die

Subtraktion einer negativen Zahl z.B. ist eigentlich eine Addition und so wird der Befehl gehandhabt. Die Vorzeichen der Resultate bei Multiplikation bzw. Division werden im Voraus bestimmt und für das Endergebnis festgehalten.

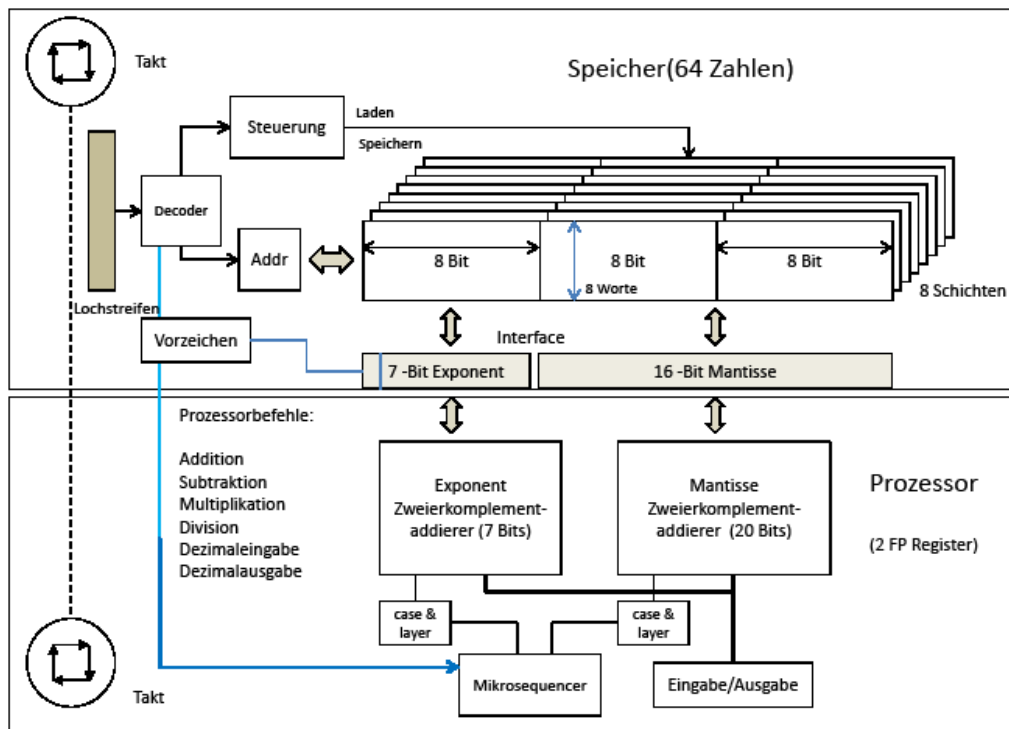


Abb. 2: Die Blockarchitektur der Z1. Speicher und Prozessor sind getrennt. Pro Speicherebene werden acht Fließkommazahlen gespeichert und es gibt acht Speicherebenen. Die Befehle werden aus einem Lochstreifen abgelesen. Der Prozessor enthält je eine ALU für Exponenten- und für Mantissenbehandlung der Fließkommazahlen. Ein Mikrosequencer steuert die ALUs und selektiert die mechanischen Ebenen.

Der Prozessor ist dementsprechend in zwei Teile gegliedert: Es gibt eine ALU (arithmetical logical unit) für den Exponenten und eine für die Mantisse. Sichtbar für den Programmierer waren nur zwei Fließkomma-Register (F und G genannt). Der Programmierer konnte zwei halblogarithmische Zahlen in diese Register laden und eine der vier arithmetischen Grundoperationen starten. Außerdem konnten Dezimalzahlen in die Maschine eingelesen werden (die dann im Register G landeten) und es konnten Dezimalresultate in einem mechanischen Display angezeigt werden. An dieser Stelle ist wiederum die Z3 ein gutes Beispiel für die in der Z1 verwendeten Algorithmen für die Dezimal-Binär-Umwandlung (und umgekehrt) [1].

Während im Speicher der Z1 nur 16 Bits der Mantisse gespeichert wurden, verwendete die Mantissen-ALU 20 Bits. Gespeichert wurden nur die Bits für die Potenzen 2^{-1} bis 2^{-16} . Beim Laden der Mantisse kam ein Bit für die Potenz 2^0 dazu (wegen der normalisierten Fließkommazahlen), noch ein Bit für die Zweierpotenz 2^1 (um einen größeren Zahlenbereich in der ALU abzudecken), und zwei zusätzliche Bits mit der niedrigsten Wertigkeit (2^{-17} und 2^{-18}), um die Genauigkeit von Abrundungen in der ALU zu erhöhen.

Ein arithmetischer Befehl (wie die Multiplikation) verwandelte sich in der ALU in eine Folge von Mikrooperationen. Die Mikrooperationen wurden von der Mikroprogrammsteuerung gestartet, die die einzelnen Mikrobefehle zählte und einen nach dem anderen für die Steuerung der beiden ALUs verwendete. Das Prinzip ist genau das gleiche, das wir heute als Mikroprogrammierung kennen. Während aber die Mikrobefehle in der Z3 mit Wahlrädern implementiert wurden, verwendete Zuse in der Z1 Bleche, die nacheinander aktiviert wurden. Der Mikrosequenzer wählte sowohl den zu bearbeitenden Mikrobefehl als auch die mechanische Ebene, wo er ausgeführt werden sollte. Eine ausführliche Beschreibung des Mikrosequenzer der Z1 ist eingereicht worden [7].

3 Der Prozessor der Z1

Die oben beschriebene Konzeption mutet sehr konventionell an. Diese sollte Informatiker allerdings überraschen, da wir immerhin über einen Rechner von 1938 reden, den ersten, den Konrad Zuse entwarf. Eine solche apodiktische Eleganz vermisst man bei allen anderen Anwärtern auf den Titel des ersten Computers der Welt (wie die amerikanische ENIAC oder die Mark I). Eigentlich fehlte nur der bedingte Sprung, um die Z1 als vollwertigen Universalrechner bezeichnen zu können.

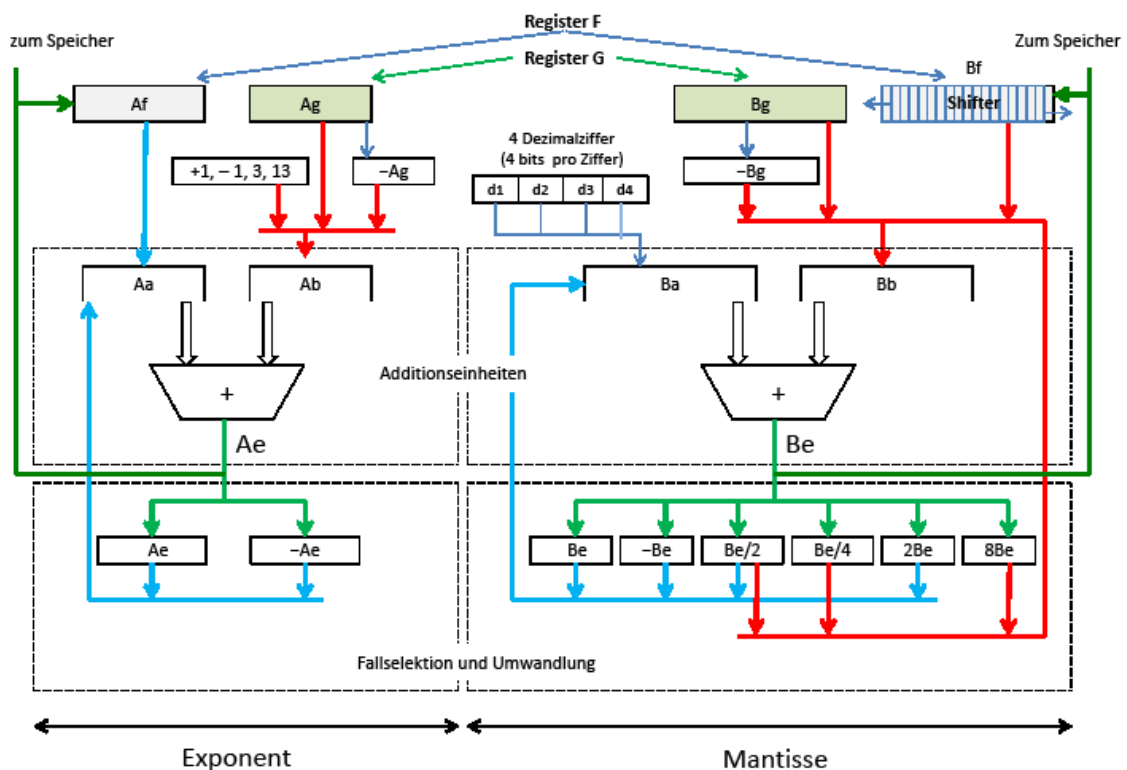


Abb. 3: Die Struktur des Prozessors der Z1. Register F und G sind für den Programmierer sichtbar. Die linke ALU ist für die Exponenten, der rechte für die Mantissen. Die Fließkommaregisterpaare (Aa,Ba) und (Ab,Bb) sind für den Programmierer unsichtbar. Dort werden Zwischenergebnisse der mikroprogrammierten Befehle festgehalten und wieder verwendet.

Die Struktur des Prozessors der Z1 ist ähnlich modern. Sie ist in Abb. 3 wiedergegeben. Beide ALUs, für Exponenten und Mantisse sind in der Abbildung zu sehen. Register F besteht aus Exponenten und Mantisse, d.h. (Af,Bf), und Register G ebenfalls, d.h. (Ag,Bg). Jede ALU kann nur zwei Binärzahlen

addieren (unter Ausnutzung der Zweierkomplementdarstellung). Die Registerpaare (Aa,Ba) und (Ab,Bb) sind Hilfsregister, die für den Programmierer unsichtbar sind. Dort werden die Zwischenresultate festgehalten, die durch die Mikrobefehle erzeugt werden.

Vier Dezimalziffern aus der Eingabetastatur werden in Gruppen von je vier Bits verwandelt und über dem Hilfsregister Ba in die ALU eingebracht, exakt wie in der Z3 [1].

Die einfachste zu beschreibende Operation ist die Addition: Die Registerpaare (Aa,Ba) und (Ab,Bb) werden mit den Daten aus dem Fließkommaregister F und G geladen und das Gesamtergebnis liegt nach mehreren Zyklen fertig bereit. Bei einer Addition zweier positiver Fließkommazahlen $m_1 \times 2^{a_1}$ und $m_2 \times 2^{a_2}$ wird zuerst ermittelt welche der beiden Exponenten a_1 bzw. a_2 grösser ist. Beide Zahlen werden dann auf diesen Exponenten reduziert, so dass man anschließend die Mantissen addieren kann. Um den größten Exponenten zu bestimmen, werden zuerst die Exponenten in der linken ALU subtrahiert. Der Ausgang der ALUs kann entweder über Ae unverändert im Register Aa, oder negiert über die Schaltung $-Ae$, geladen werden. Die ALU-Logik überprüft vorher das Vorzeichen des Resultats Ae, so dass falls Ae negativ ist, der Weg über $-Ae$ eingeschaltet wird. So oder so, hat man am Ende der Ausführung den absoluten Wert der Differenz im Hilfsregister Aa stehen.

Ein schöner Nebeneffekt der Trennung der ALUs ist, dass die linke ALU als Zähler für die rechte ALU verwendet werden kann. Man kann den absoluten Wert der Differenz der Exponenten jedes Mal um Eins verringern, das Ergebnis wieder in Aa laden, bis Ae Null wird, und so eine Schleife steuern. Deswegen gibt es in der linken ALU die Möglichkeit, die Registerwerte direkt einzugeben (oder negiert, für eine Subtraktion von Ag), es können aber auch gewisse Konstanten direkt eingegeben werden. Diese Konstanten sind bei bestimmten Operationen hilfreich, wie etwa um ein Zwischenresultat um Eins zu erhöhen oder zu verringern (die Konstanten +1 und -1). Die Konstanten 3 und 13 haben bei anderen Operationen eine spezielle Anwendung.

Die ALUs sind immer aktiv: In jedem Zyklus werden die Hilfsregister addiert. Hilfsregister die nicht gleich nachgeladen werden, werden auf Null gesetzt. So kann man ein Ergebnis „in Bewegung“ festhalten, indem z.B. Ae nach Aa transferiert wird und immer wieder mit Null in Ab addiert wird, bis man das Ergebnis für etwas braucht.

Bei dem Mantissenteil kann man zwei Register addieren und das Ergebnis anschließend nach rechts oder links verschieben, um es wieder zu verwenden. Man kann eine oder zwei Stellen nach rechts shiften (mit den Schaltungen $Be/2$ bzw. $Be/4$) oder eine bzw. drei Stellen nach links ($2Be$, $8Be$). Man kann das Zwischenergebnis unverändert wieder verwenden (Be) oder negiert in die Zwischenregister laden ($-Be$). Damit sind alle am meisten verwendeten Bit-Verschiebungen direkt im Prozessor implementiert.

Solche speziellen Shift-Schaltungen bilden den größten Unterschied zwischen den Prozessoren der Z3 und der Z1. Die Rechenmaschine Z3 hatte einen aufwendigen Shifter, der die Mantisse bis zu 16 Stellen nach rechts und 15 nach links verschieben konnte. Damit war es möglich, Mantissen in weniger als einem Zyklus korrekt zueinander für eine Addition zu verschieben. In der Z1 muss man dagegen die Shift-Operationen iterieren. Will man fünfmal nach rechts verschieben, muss man den Einzelshift $Be/2$ fünfmal verwenden (mit Hilfe z.B. der Exponent-ALU, der als Zähler arbeitet). Man kann aber auch zweimal $Be/4$ und einmal $Be/2$ verwenden.

Eine Addition in der Z1 kann deswegen ziemlich viele Zyklen benötigen. Ein Beispiel kann dies illustrieren.

Nehmen wir an, wir möchten die Zahlen 1.0×2^0 und 1.0×2^{-10} addieren. Vor der Addition der Mantissen muss die Darstellung beider Zahlen denselben Exponenten aufweisen. Wir benutzen das Maximum der Exponenten, so dass die zu addierenden Zahlen folgende sind:

$$1.00000000000000000000 \times 2^0$$

$$0.00000000010000000000 \times 2^0$$

Man muss also die kleinere Mantisse zehn Positionen nach rechts verschieben. Anschließend können die Mantissen addiert werden. Da in diesem Fall das Ergebnis normalisiert ist, muss man den Exponenten der Summe nicht mehr verändern. Ist allerdings das Ergebnis einer Addition größer oder gleich 2, muss die Mantisse nach links verschoben werden und der Exponent dementsprechend justiert werden. Das bedeutet, dass alle Rechenoperationen in der Z1 eine variable Anzahl von Zyklen benötigen, je nachdem welche Daten bearbeitet werden.

Jetzt allgemeiner: Eine Addition würde im Prozessor der Z1 in den Schritten unten ausgeführt. Man beachte den Datenfluss zwischen den Registern (mit einem Pfeil angedeutet). In jedem Zyklus werden Register in den ALUs, die nicht ausdrücklich geladen werden, auf Null gesetzt. Die ALUs berechnen, wie oben angemerkt, immer eine Addition der Hilfsregister in jedem Zyklus. Eine Addition enthält dann folgende Mikrooperationen:

Zyklus	Operationen	Kommentar
(1)	$A_f \rightarrow A_a, -A_g \rightarrow A_b$; berechne Differenz der Exponenten
(2)	$A_e \rightarrow A_a$, falls $A_e > 0$ setze $S_1 = 1$; setze Hilfsbit S_1 auf 1 falls $A_e > 0$ (Null ist default)
(3)	Falls $S_1 = 1$ dann $A_e \rightarrow A_a, B_g \rightarrow B_b$ sonst $-A_e \rightarrow A_a, B_f \rightarrow B_b$; größter Exponent ist A_f , kleinere Zahl in B_b ; größter Exponent ist A_g , kleinere Zahl in B_b ; A_a enthält absoluten Wert der Exponentendifferenz
(4..n)	Wiederhole bis $A_e = 0$ $-1 \rightarrow A_b, A_e \rightarrow A_a, B_e/2 \rightarrow B_b$ Wenn $A_e = 0, B_e \rightarrow B_a$; reduziere Zähler, shifte Mantisse rechts ; Zähler bei Null, verschobene Mantisse in B_a
(n+1)	Falls $S_1 = 1$ $A_f \rightarrow A_a, B_f \rightarrow B_b$ sonst $A_g \rightarrow A_b, B_g \rightarrow B_b$; größter Exponent in A_a , Mantisse größere Zahl in B_b ; größter Exponent in A_b , Mantisse größere Zahl in B_b

(Addition der Mantissen wird ausgeführt, Ergebnis liegt in (A_e, B_e) vor, Normalisierung folgt dann, falls notwendig)

Eine Subtraktion wird in ähnlicher Form ausgeführt, aber zuerst werden die Exponenten der beteiligten Zahlen durchgesehen. Wenn wir die Zahlen a und b als positiv betrachten, werden folgende Transformationen von der Kontrolllogik verwendet:

$$(+a) - (+b) = + (a - b)$$

$$(-a) - (+b) = - (a + b)$$

$$(+a) - (-b) = + (a + b)$$

$$(-a) - (-b) = - (a - b)$$

Die endgültige Festsetzung des Vorzeichens des Resultats wird a posteriori gemacht, nachdem das Vorzeichen einer Subtraktion vorliegt. Bei Addition braucht das vorläufige Vorzeichen nicht korrigiert zu werden.

Die Multiplikation, die Division und die Dezimal-Binär-Umwandlung werden in der Z1 ähnlich wie in der Z3 behandelt. Für das Abtasten und Setzen der einzelnen Bits des Multiplikators wurde ein Shifter verwendet, der direkt neben dem Register B_f angebracht wurde. Das ist die einzige Hardware, die zusätzlich benötigt wird um die Mikrosequenzen für die Multiplikation und Division zu steuern. Aus Platzgründen gehe ich hier nicht auf diese Algorithmen ein und verweise den Leser wiederum auf die Algorithmen der Z3.

4 Fazit

Es ist heute unmöglich, abschließend zu klären, ob die Rekonstruktion der Z1 in Berlin wirklich dem Original von 1938 entspricht. Man erkennt an den damaligen Bildern sofort, dass die neue Z1 aufpolierter und schlanker als die alte Maschine ist. Ob im Inneren alles identisch war, können wir nicht mehr bestimmen und an der Stelle müssen wir Konrad Zuse einfach glauben. Er hatte eigentlich keinen gewichtigen Grund, die Maschine nachträglich zu verschönern. Wir kennen allerdings die Tücken des menschlichen Erinnerungsvermögens – manchmal lässt es uns im Stich oder wir glauben Ideen früher als in der Wirklichkeit gehabt zu haben.

Sicher ist jedoch, dass die Berliner Z1 ein Muster an architektonischer Eleganz ist. Sie enthält Hunderte von Blechen und Stangen, aber keine zu viel. Man könnte vielleicht denken, dass die verschiedenen Shifts am Ausgang der Mantissen-ALUs durch nur zwei Shifts ersetzt werden könnten (einmal nach links und einmal nach rechts). Die zusätzlichen Shifts vermeiden jedoch unnötige Zwischenspeicherungen und beschleunigen die gebräuchlichsten Operationen. Deswegen ist der Prozessor der Z1 eigentlich sogar eleganter als der der Z3, da es sich um eine kompaktere Maschine handelt. Es ist, als ob Zuse mit den Telefonrelais der Z3 weniger bei den Komponenten sparen musste und deswegen eine aufwendigere Konstruktion wagte. Dasselbe wiederholte sich Jahre später mit der Z4, die eigentlich nur eine größere und sperrigere Z3 war.

Es ist für mich heute noch erstaunlich, wie dem jungen Diplomingenieur Konrad Zuse 1936 eine solch moderne Konstruktion gelang. Die Erbauer der ENIAC waren eine Schar von Ingenieuren, darunter gestandene Professoren. Was sie erschufen ist aber im Vergleich zur Z1 ein Dinosaurier. Die Z1 ist, trotz der mechanischen Ausführung, eine ausgereifere Konstruktion als vieles, was bis 1945 gebaut wurde. Noch schlankere Rechnerarchitekturen wurden erst beim Nachfolger der ENIAC, der EDVAC, konzipiert, aber da hatte schon ein John von Neumann seine Finger im Spiel. Er lebte von 1926 bis 1929 in Berlin und war der jüngste Privatdozent an der Berliner Universität. Vielleicht sind der Erstsemestler Zuse und der Privatdozent sich irgendwann über den Weg gelaufen. Wie viel hätte in Berlin sein können -- bevor eine ausgedehnte Nacht über Deutschland hereinbrach.

Referenzen

- [1] Rojas, R. (Hrsg.): *Die Rechenmaschinen von Konrad Zuse*, Springer-Verlag, Berlin, 1998.
- [2] Rojas, R.: "Konrad Zuse's legacy: the architecture of the Z1 and Z3", *Annals of the History of Computing*, Vol. 19, N. 2, 1997, S. 5–16.
- [3] Zuse, K.: *Der Computer – Mein Lebenswerk*, Springer-Verlag, Berlin, 3. Auflage, 1993.
- [4] Ursula Schweier, Dietmar Saupe, „Funktions- und Konstruktionsprinzipien der programmgesteuerten mechanischen Rechenmaschine Z1“, Arbeitspapiere der GMD 321, GMD, Sankt Augustin, August 1998.
- [5] Webseite: Architecture and Simulation of the Z1 Computer, <http://zuse-z1.zib.de/>, letzter Zugriff: July 21st, 2013.
- [6] Zuse, K.: "Rechenvorrichtung aus mechanischen Schaltglieder", Zuse Papers, GMD 019/003 (undatiert), <http://zuse.zib.de/>, letzter Zugriff July 21st, 2013.
- [7] Rojas, R.: „The Z1: Architecture and Algorithms of Konrad Zuse's First Computer“, unveröffentlichtes Manuskript, eingereicht beim *Annals of the History of Computing*.