Prof. Dr. Timm Grams
Hochschule Fulda
Fachbereich Elektrotechnik und Informationstechnik
Marquardstraße 35
36039 Fulda

Tel (0661) 78157
(0661) 9640-550
Fax (0661) 9640-559
mailto:Timm.Grams@et.hs-fulda.de
www.hs-fulda.de/~grams

Samstag, 27. März 2010 (aktualisiert 07.06.2012)

Das Z1-Addierermodell im Konrad-Zuse-Museum Hünfeld – Eine kleine Entdeckungsreise

Inhalt

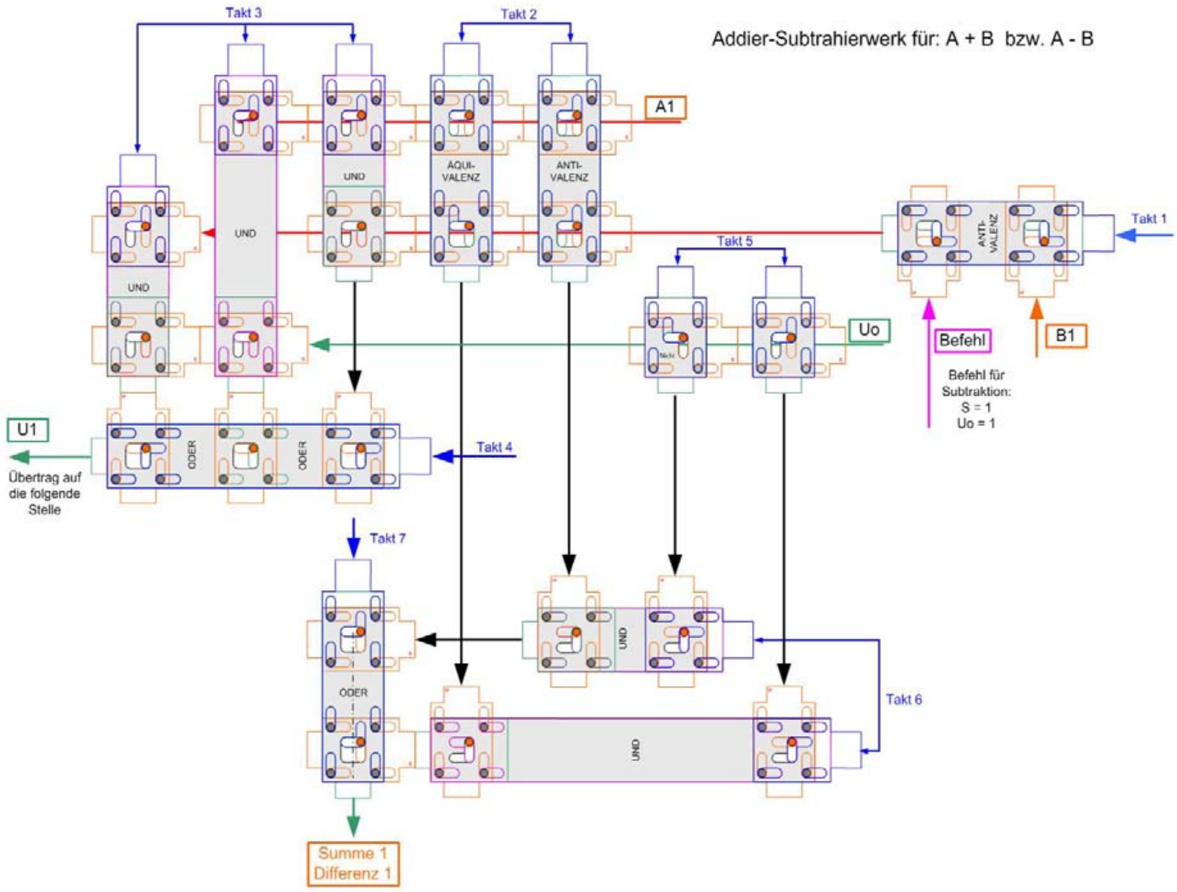
Anmerkungen zum Demonstrationsmodell „Mechanischer Addierer“ im Zuse-Museum Hünfeld (27.03.2010)

Das im Museum ausgestellte Modell eines Addierers ist als Demonstrationsmodell für die in mechanischer Schaltgliedtechnik gedacht und dient in erster Linie didaktischen Zwecken. Wünschenswert ist ein Modell, das darüber hinaus auch die Funktionsweise des von Zuse tatsächlich realisierten Z1-Addierers zeigt.

In dieser kleinen Studie wird der Frage nachgegangen, inwieweit das derzeitige Modell diesem Anspruch genügt und wie man es verbessern könnte.

Beschreibung des aktuell ausgestellten Modells

Das Modell für einen Addierer in mechanischer Schaltgliedtechnik – Exemplare befinden sich im Museum Hünfeld und im Zuseum Bautzen – geht auf Vorarbeiten von Herrn Helmuth Fies, Bad Krozingen, zurück. Sein Entwurf sieht so aus:



Ich bezeichne einmal die beiden zu addierenden Bits mit *a* und *b*. Das Carry-Bit von der niedrigerwertigen Stelle bezeichne ich mit *c* und den Übertrag auf die nächsthöhere mit *t*. Das Summenbit ist *s*. In der folgenden Darstellung lasse ich die Komplementbildung zunächst einmal weg (eingangsseitige Invertierung der Variablen *b*).

Nach dem Standard-Entwurfsverfahren ergeben sich die folgenden Darstellungen in kanonischer disjunktiver Normalform (KDNF):

$$s = a\neg b\neg c \vee \neg ab\neg c \vee \neg a\neg bc \vee abc$$

$$t = ab\neg c \vee a\neg bc \vee \neg abc \vee abc$$

Das Konjunktionszeichen $\wedge$ wird nicht hingeschrieben: $a\neg b\neg c$ steht für $a\wedge\neg b\wedge\neg c$ usw. Äquivalenztransformationen liefern für das Übertragsbit eine einfachere Darstellung:

$$t = ab \vee ac \vee bc$$

Außerdem ergibt sich eine Vereinfachung der Formel für das Summenbit durch Nutzung der Äquivalenz und Antivalenz:

$$s = (a{\neq}b)\neg c \;\vee\; (a{=}b)c.$$

Diese Formeln liegen dem obigen Layout zu Grunde. Auf Basis dieses Layouts wurde das Modell von Andreas Samuel (Zuseum Bautzen) in farbigem Plexiglas gefertigt.

Zuses Beschreibung des Addierers mit einschrittigem Übertrag

Zuse hat – da sich die Antivalenz in mechanischer und in elektromechanischer Schaltgliedtechnik besonders einfach und kompakt realisieren lässt – noch weitergehende Vereinfachungen unternommen. Eine betrifft das Summenbit, denn hier lassen sich $\wedge$, $\vee$ und $\neg$ komplett durch Einführung eines weiteren Antivalenzgliedes eliminieren:

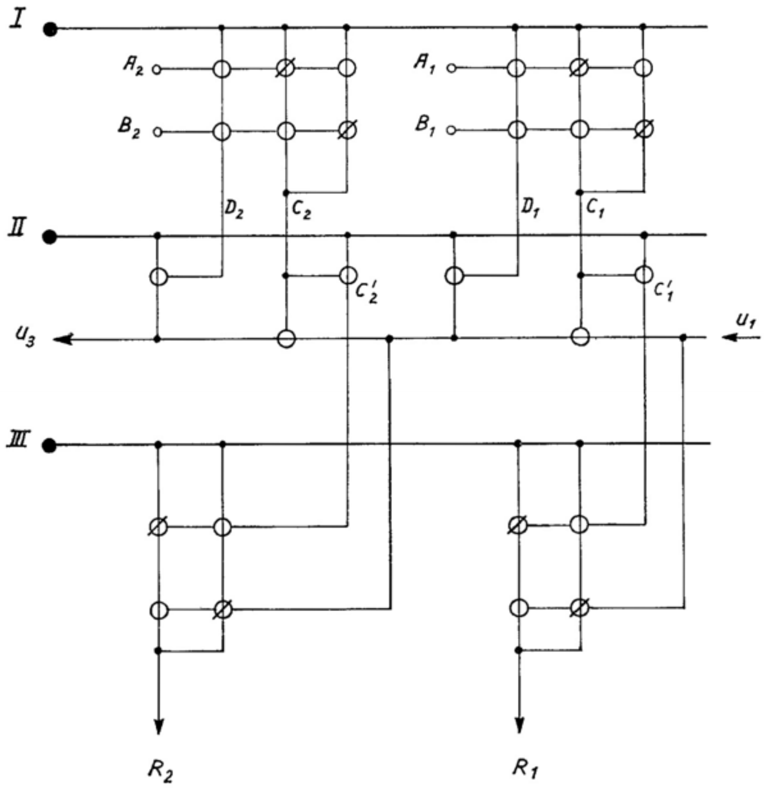
$$s = (a \neq b) \neq c.$$

Auch für das Carry-Bit gibt es eine günstigere Darstellung. Um das zu sehen, geht man am besten auf die ursprüngliche KDNF zurück. Ein paar Umformungen liefern die kompakte Darstellung mittels Antivalenz:

$$t = (a \neq b)c \vee ab.$$

Es lohnt sich, diesen Ausdruck genauer zu betrachten: Der Teilausdruck *ab* ist vom letzten Übertrag *c* unabhängig und steht sofort zur Verfügung. Das gilt auch für den Teilausdruck $a \neq b$. Er macht das Übertragsbit der vorhergehenden Stelle wirksam.

Diese Beobachtungen lassen sich in der Relaistechnik vorteilhaft nutzen. Die Schaltskizze zeigt die Addierschaltung in der von Zuse entwickelten „abstrakten Schaltgliedtechnik“[1]. (In der Skizze bezeichnet *c* die Zwischengröße $a \neq b$; und für den Übertrag steht *u* anstelle von *c*.)



Ein mit $a \neq b$ angesteuertes Relais kann das vorhergehende Übertragsbit *c* verzögerungsfrei durchschalten bzw. sperren. Leitungsabschnitte (in der Skizze $u_3 \leftarrow u_1$) sorgen für eine Wired-Or-Verknüpfung der durch die Produkte *ab* erzeugten Signale. Eine Rückwirkung auf den vorhergehenden Volladdierer ist dennoch ausgeschlossen: Wenn *ab* gleich 1 ist, ist der Leitungsabschnitt zum vorhergehenden Volladdierer aufgetrennt, da in diesem Fall ja $a \neq b$ gleich 0 ist. Ein Carry-Bit mit dem Wert 1 kann sich also nur von rechts nach links, und nie von links nach rechts fortpflanzen. Damit ist das Lösungsprinzip des Zuseschen Addierers mit einschrittigem Übertrag erklärt.

Übertragung auf die mechanische Schaltgliedtechnik

Zuses raffinierte Lösung des Addierers mit einschrittigem Übertrag in Relaistechnik lässt die Frage aufkommen, wie sein Addierer in rein mechanischer Schaltgliedtechnik wohl aussieht.

Die verlässlichste Quelle ist der Z1-Nachbau im Technischen Museum in Berlin, den Konrad Zuse noch selbst vorgenommen hat. Allerdings bekennt Horst Zuse[2]: „Die Funktionsweise der arithmetischen Einheit (Binäres Gleitkommarechenwerk) ist Konrad Zuses Geheimnis geblieben und er hat es 1995 mit ins Grab genommen. Die existierenden Aufzeichnungen entsprechen in vielen Teilen nicht der tatsächlichen Struktur bzw. Bauweise der Teile der Maschine Z1. Es ist eine Herausforderung an historisch interessierte Personen, die Funktion der gesamten Maschine Z1 zu verstehen und graphisch darzustellen.“

---

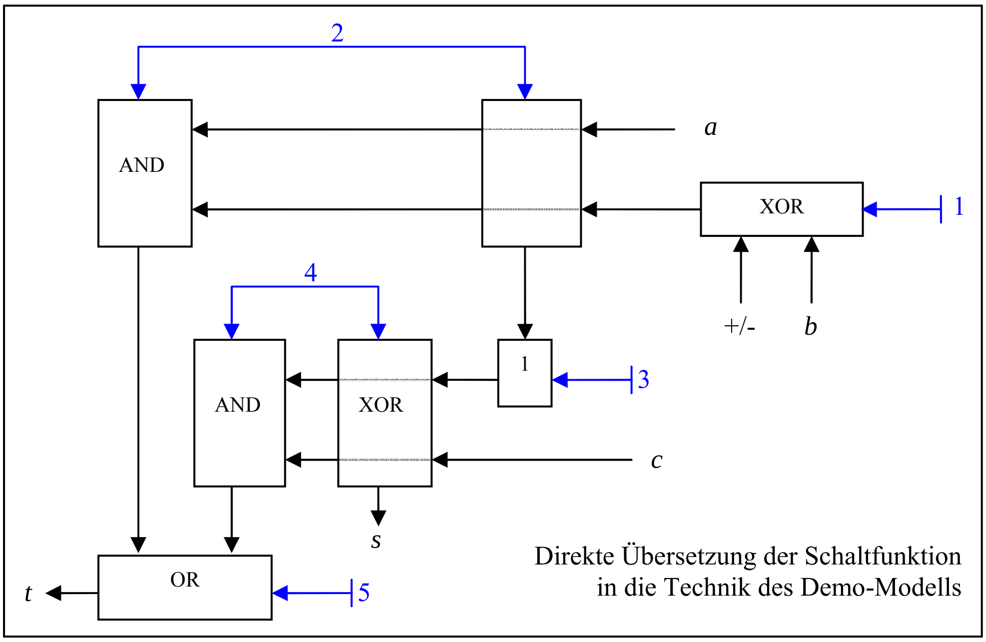
[1] Zuses, Konrad: Der Computer – Mein Lebenswerk. Springer-Verlag, Berlin, Heidelberg 1990

[2] Alex, Jürgen; Flessner, Hermann; Mons, Wilhelm; Pauli, Kurt; Zuse, Horst: Konrad Zuse. Der Vater des Computers. Verlag Parzeller, Fulda 2000

Ich will eine *erste Annäherung an den Addierer der Z1* versuchen. Ich nutze zwei Quellen:

1. Die Demonstrationsmodelle im Zuse-Museum Hünfeld und im Zuseum Bautzen (s. oben). Diese Modelle dienen jedoch in erster Linie didaktischen Zwecken und sind nicht mit der Absicht entstanden, den Addierer der Z1 zu rekonstruieren – so der I-deengeber für diese Schaltungen, Helmuth Fies.
2. Ein Hinweis, den Konrad Zuse in seiner Autobiographie im Abschnitt über „die abs-trakte Schaltgliedtechnik“ gibt (S. 180 ff.): „Das mechanische Gerät Z1 und das späte-re elektromechanische Gerät haben fast die gleichen abstrakten Grundschaltungen.“

Nahe liegt der Versuch, die Formeln für den Addierer mit einschrittigem Übertrag in die Technik der Demonstrationsmdelle zu übersetzen. Die nebenstehende Skizze zeigt eine solche Lösung.

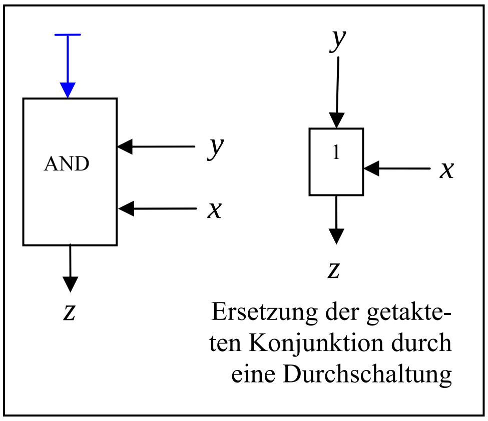


Direkte Übersetzung der Schaltfunktion in die Technik des Demo-Modells

Dieser Entwurf genügt den folgenden Vorgaben:

1. Bei den Ein- und Ausgangsgrößen sind die Bewegungen jeweils genau so orientiert wie im Demonstrationsmodell.
2. Es werden nur getaktete Schaltglieder für die Verknüpfungen verwendet, wie sie auch im Demonstrationsmodell vorkommen.
3. Alle Verknüpfungen, die den Übertrag vom vorhergehenden Addierer benötigen, sind auf möglichst wenige Taktstufen konzentriert und an das Ende der Verarbeitungskette gelegt.

Die Takte 1, 2 und 3 können in allen Addierern der Addie-rerkette parallel und gleichzeitig erfolgen: Erst kommt der Takt 1 in allen Addieren, dann Takt 2 und schließlich Takt 3. Der Zeitaufwand für die ersten drei Takte beträgt also genau 3 Takteinheiten für sämtliche Addierer. Die Takte 4 und 5 allerdings müssen nacheinander erfolgen, und zwar fortschreitend vom Addierer ganz rechts bis zum Addierer ganz links. Bei der Addition zweier *n*-stelliger Dualzahlen sind also 3+2*n* Takte erforderlich.



Ersetzung der getakte-ten Konjunktion durch eine Durchschaltung

Bei Lockerung der Entwurfsvorgaben sind weitere Vereinfachungen denkbar, beispielsweise die Ersetzung einer getakteten Konjunktion durch eine einfache Durchschaltung, bei der die Taktung durch eine der Eingangsgrößen ersetzt wird. Dabei entstehen allerdings Reihenfolgeprobleme beim Setzen und Rücksetzen der Variablenwerte. Außerdem sind die Eingangsgrößen verschieden orientiert.

Eine zweistufige Schaltungsvariante

Wenn es einzig darum geht, die Anzahl der Takte gering zu halten, bietet sich auch eine direkte Realisierung auf Basis einer minimierten disjunktiven Normalform mit den Formeln

$$s = a\neg b\neg c \vee \neg ab\neg c \vee \neg a\neg bc \vee abc$$

$$t = ab \vee ac \vee bc$$

an.

In mechanischer Schaltgliedtechnik sind prinzipiell Verknüpfungsglieder mit mehr als zwei Eingangsgrößen möglich, und die Negation einer Eingangsgröße bedeutet keinen Mehraufwand. Damit wäre ein mechanischer Addierer möglich, der die heute übliche zweistufige elektronische Schaltungstechnik getreulich nachbildet.

Die Addition erfordert jetzt je Volladdierer nur zwei Takte. Allerdings hat man es je Takt mit vielen bewegten Teilen auf einmal zu tun. Ein weiteres Problem bieten die Schaltglieder mit mehr als zwei Eingangsgrößen: Man benötigt mehrstufig gekoppelte Bleche mit entsprechend reduzierter Fertigungstoleranz.

Wie weiter?

An der Z1 im Technischen Museum Berlin könnte unter Beisein von kundigem Personal und möglichst auch unter Anwesenheit der an der Rekonstruktion der Z1 beteiligten Personen (Ursula Schweier und Dietmar Saupe) erkundet werden, wie der Addierer dort aufgebaut ist.

Angestrebtes Ergebnis ist eine Schaltskizze des Volladdierers, wie ihn Konrad Zuse tatsächlich realisiert hat.

Dann könnte man Modelle für die Addition zweier zweistelliger Dualzahlen aus einem Halbaddierer und einem Volladdierer in der bewährten Demo-Technik aufbauen und diese den interessierten Museen und Ausstellern zum Kauf anbieten.

Mechanischer Addierer mit einschrittigem Übertrag (Ergänzung: 09.04.2010)

Bei der Internetrecherche traf ich nach Jahren wieder auf ein Java-Applet von Alexander Thurm, das den Volladdierer der Z1 simuliert:

http://www.zib.de/zuse/Inhalt/Programme/Simulationen/Z1/Z1Adder/adder.html.

Das obige Schaltbild einer *ersten Annäherung an den Addierer der Z1* hat mir diesmal geholfen, diese Simulation zu verstehen, denn die Java-Simulation ist im Wesentlichen eine Realisierung dieses Schaltbilds.

Aber es gibt auch Unterschiede. Tatsächlich wird hier für die Und-Verknüpfung von $a \neq b$ und c – wie bereits vermutet – eine einfache Durchschaltung verwendet. Außerdem wird die Oder-Verknüpfung ohne ein Taktsignal bewerkstelligt, und zwar derart, dass das Ausgangsblech entweder von dem einen Eingang oder vom anderen Eingang in die 1-Position verschoben werden kann. Aus diesem Wirkungsprinzip ergibt sich die Notwendigkeit, nach getaner Arbeit alle Bleche jeweils wieder in die 0-Position zurückzusetzen.

Das folgende Schaltbild zeigt meine *zweite Annäherung an den Addierer der Z1*.

Achtung beim Lesen des Schaltbildes: Ist eine Linie (leicht abgeschwächt) durch ein Schaltgliedblock durchgezogen, dann handelt es sich um eine Eingangsgröße, die auch an den folgenden Block weitergegeben wird, und nicht etwa um Ein- und Ausgangsgröße!

Da die Carry-Bits jetzt selbst für die Bewegung sorgen, also die „bewegenden Bleche“ sind, und die Bewegung sich über eine ganze Kette von Volladdierern fortsetzen kann, spielen die Fertigungstoleranzen eine ausschlaggebende Rolle: Die Bewegung des letzten Bleches einer solchen Kette ist um die Summe der Spielräume für die beteiligten Schaltstifte vermindert.

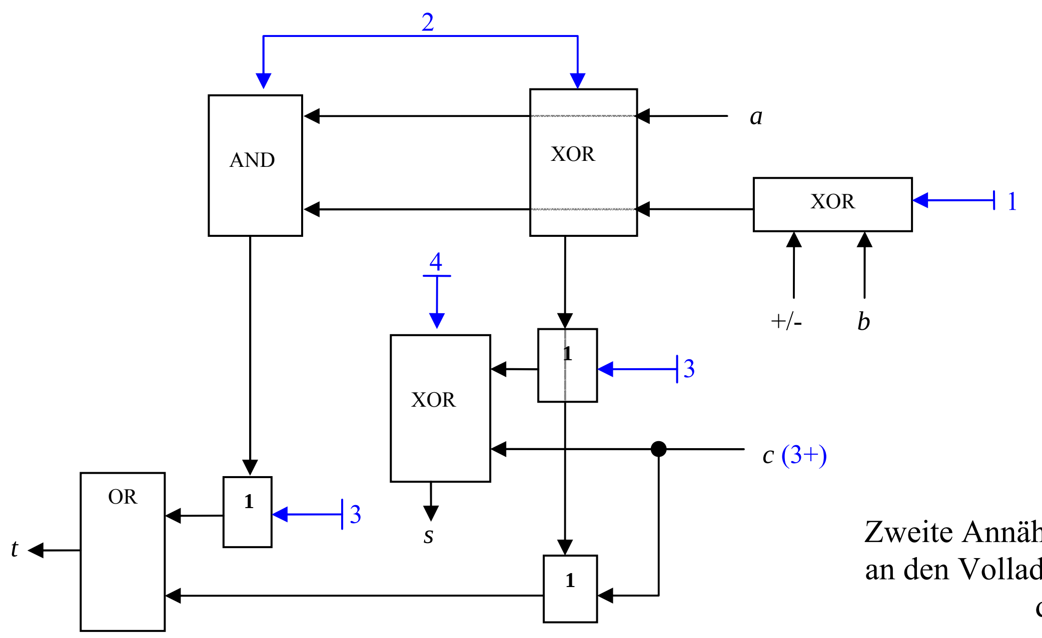
Die ersten beiden Takte betreffen nur die Variablen *a* und *b* und können nach wie vor in sämtlichen Volladdierern gleichzeitig durchlaufen werden. Eine wechselseitige Beeinflussung der Volladdierer ist ausgeschlossen.

Der Takt 3 betrifft jetzt zwei Durchschaltglieder, deren Aufgabe in erster Linie darin besteht, die Signalrichtung um 90° zu drehen. Takt 3 bewirkt die Bildung des Übertrags und ist der entscheidende „antreibende“ Schritt für Übertragsketten (Schritt 3+). Dieser Übertrag geschieht in allen Volladdierern gleichzeitig. Der Mechanismus dieses dritten Schrittes ist trickreich und – wegen der Fertigungstoleranzen – auch problematisch.

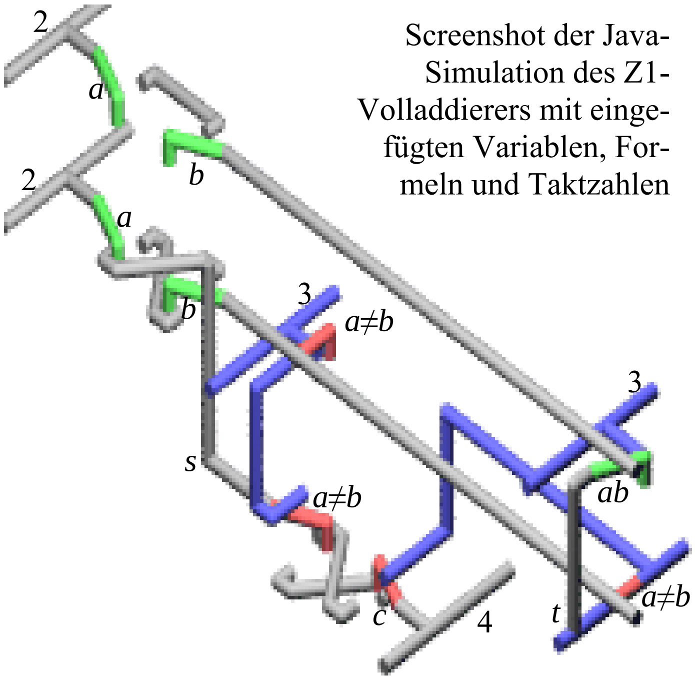
Schließlich erfolgt die Darstellung des Ergebnisbits *s* mit Schritt 4. Auch dieser Schritt kann in sämtlichen Volladdieren parallel ablaufen. Die Zahl der Takte für die Addition von Zahlen mit beliebiger Stellenzahl benötigt jetzt also insgesamt nur noch vier Taktschritte. Es handelt sich hier tatsächlich um einen *Addierer mit einschrittigem Übertrag.*

Angehängt habe ich einen Screenshot der Java-Simulation des Volladdierers. Die Bleche für die Eingangsvariablen der Schaltglieder sind rot und grün dargestellt. Die von mir eingefügten Variablenbezeichner und Formeln sollen den Vergleich der Bilder erleichtern. Außerdem habe ich die Taktbleche mit den zugehörigen Taktnummern bezeichnet. Die genaue geometrische Entsprechung zwischen den Bildern erhält man mithilfe einiger elementarer Transformationen: Spiegelung, Vertauschen der Werte 0 und 1, Vertauschen von Eingangsvariablen, Translationen usw.

Im Java-Simulationsmodell fehlt die Möglichkeit der Invertierung von Bit *b*. Dadurch entfallen auch das entsprechende XOR-Schaltglied und der zugehörige Takt 1.



Zweite Annäherung
an den Volladdierer
der Z1



Screenshot der Java-Simulation des Z1-Volladdierers mit eingefügten Variablen, Formeln und Taktzahlen

Wie weiter? – zum Zweiten

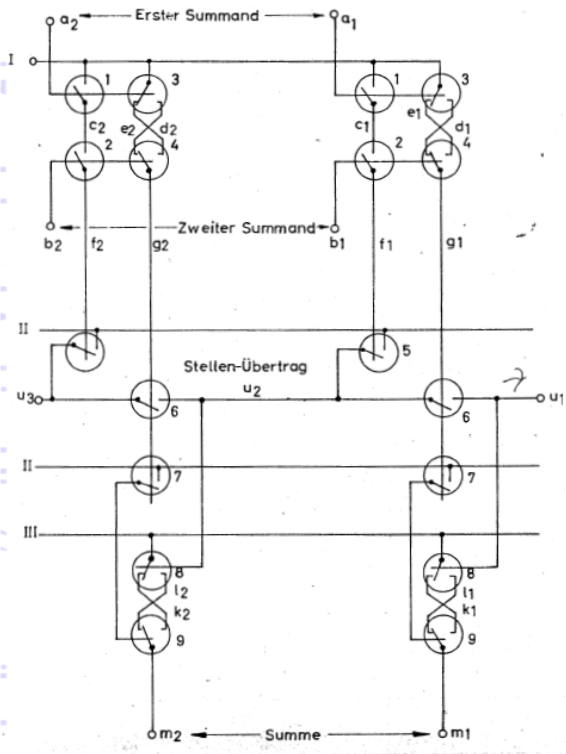
Auf Anraten von Winfried Görke, Karlsruhe, sollte man sich zunächst einmal über die Pläne und Unterlagen im Nachlass Konrad Zuses im Deutschen Museum in München informieren und dann mithilfe dieser Unterlagen die *tatsächliche Geometrie des Addierers* erkunden.

Winfried Görke und Helmuth Fies, Bad Krozingen, regen an, den Aufbau eines Addierermodells für vier anstelle von nur zwei Dualstellen ins Auge zu fassen, weil ja erst dann „die Einschrittigkeit überhaupt erkennbar wird“ (Fies).

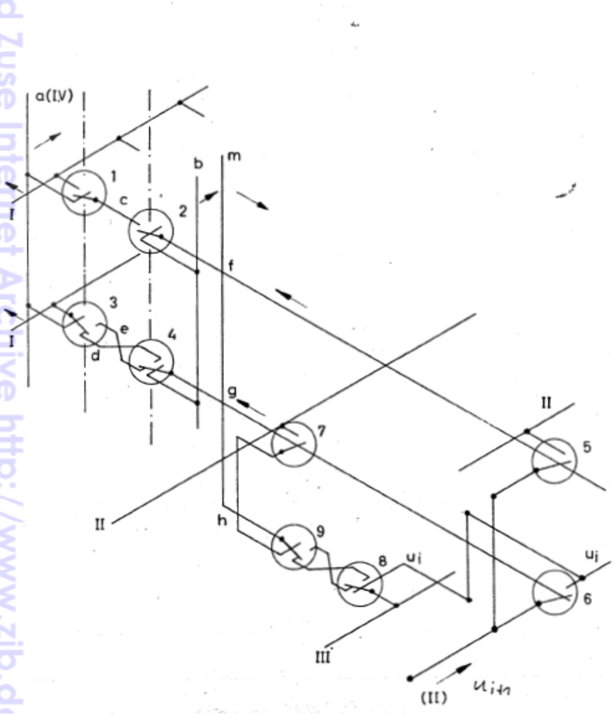
Auf der Zielgeraden (Ergänzung: 1.11.2010)

Ein Gespräch mit Horst Zuse am 24. Oktober 2010 in Berlin hat mich darauf gebracht, doch noch einmal im Zuse-Archiv (http://www.zib.de/zuse/) nach den Vorlagen für die Realisierung des Addierers zu fahnden. Und tatsächlich bin ich fündig geworden. Die Bilder 7 und 8 aus dem Dokument der Zuse K.-G. „Die historischen Modelle der Zuse-Rechengeräte-Entwicklung“ vom 23. April 1960 zeigen in voller Klarheit, wie ein Modell aussehen könnte, das die anfangs aufgestellten Forderungen nach 1. *didaktischer Zweckmäßigkeit* und 2. weitgehender *Originaltreue* möglichst gut erfüllt. Hier die Bilder:

Ersatzschaltbild für das mechanische Addierwerk (Gerät Z1)

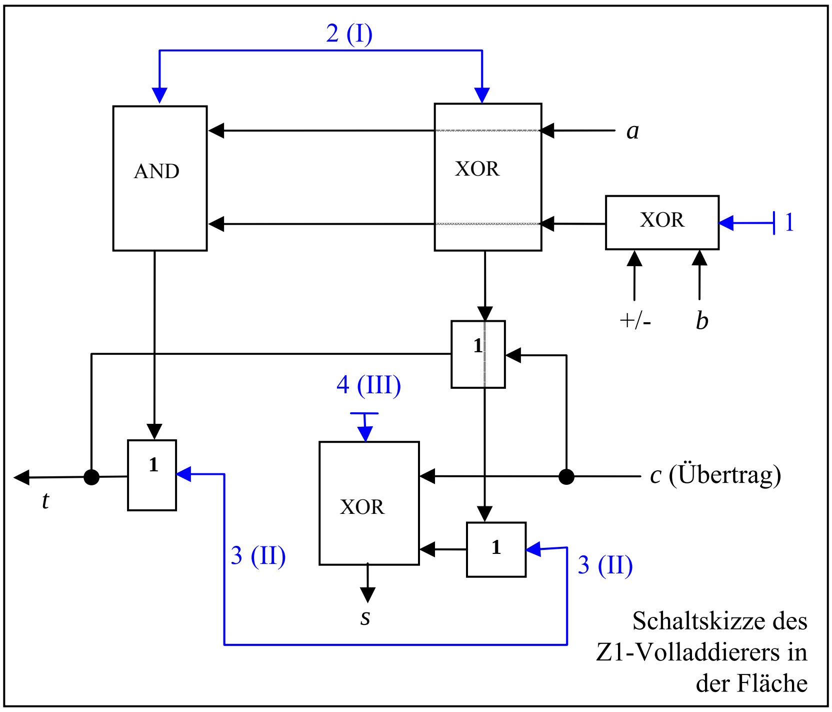


Räumliche Anordnung der mechanischen Schaltglieder des Addierwerkes entspr. Fig.7 (Gerät Z1)



Zuses Anordnung ist räumlich. Alexander Thurms Modell bildet diese Anordnung getreulich nach. Aus didaktischen Gründen wollen wir bei der flächigen Anordnung und bei den Bewegungsrichtungen des ursprünglichen Demonstrationsmodells bleiben. Das nebenstehende Bild zeigt eine mögliche flächige Anordnung.

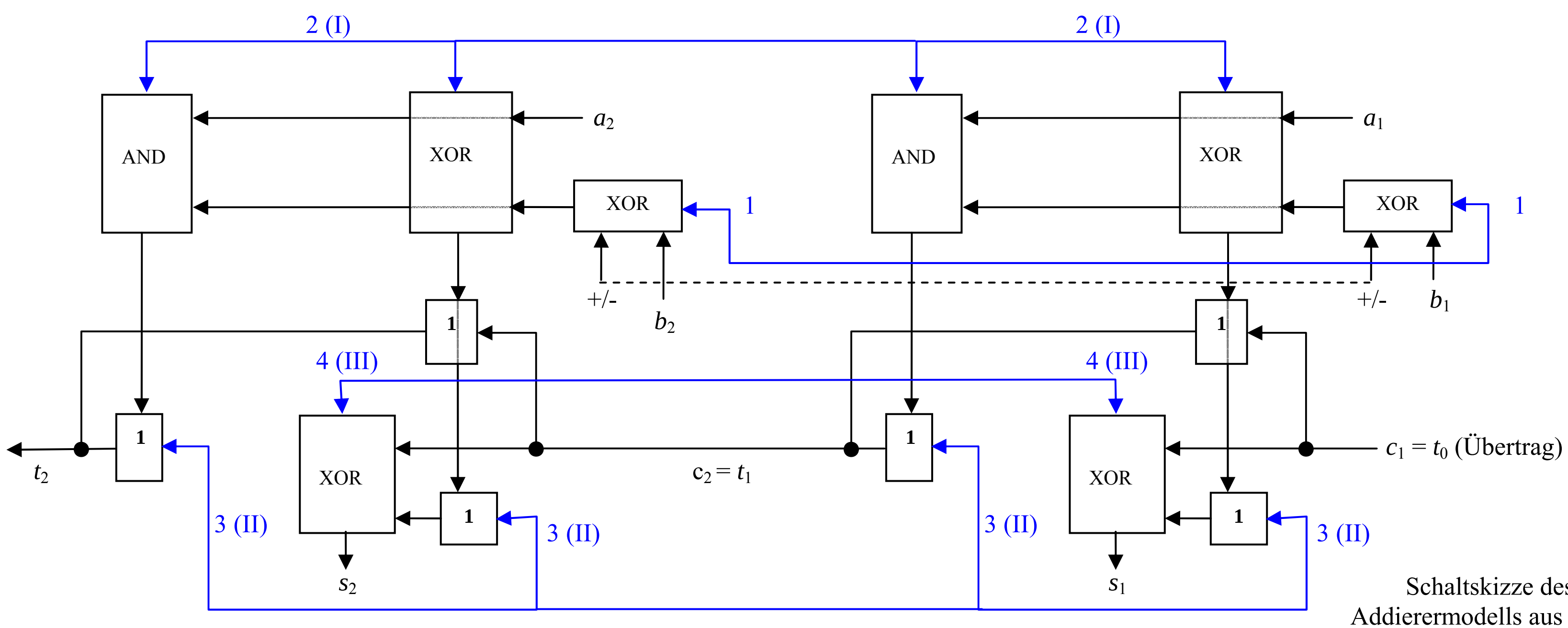
Die Pfeile an den Linien für die Takt- und Variablensignale sagen nur etwas über die Wirkungsrichtung der Signale aus. Sie legen nicht fest, zu welcher Blechposition die 0 und zu welcher die 1 gehört.



Schaltskizze des Z1-Volladdierers in der Fläche

Demonstrationsmodell aus zwei Volladdierern

Zu guter Letzt soll noch eine Schaltung mit zwei Volladdierern dargestellt werden.



Schaltskizze des Z1-Addierermodells aus zwei Volladdierern

Schlussbemerkung

Noch um 1945 schreibt Konrad Zuse in einem „Bericht über meine Rechengeräte“: „Die mechanische Relaistechnik ist betriebssicher auch ohne sorgfältige Wartung. Sie ist besonders geeignet für die Serienherstellung. Damit werden Geräte zu billigem Preis hergestellt werden können.“

In einem Bericht über die „Entwicklungslinien einer Rechengeräte-Entwicklung von der Mechanik zur Elektronik“ aus dem Jahr 1960 schreibt Konrad Zuse dann über die „natürlichen Grenzen der Mechanik“: „Man kann nicht im beliebigen Maße ‚um die Ecke herum’ übertragen.“

In seinen Lebenserinnerungen „Der Computer – mein Lebenswerk“ (1984, 1990) stellt Konrad Zuse das Scheitern des rein mechanischen Lösungsversuchs fest: „Nach einigen Anfangserfolgen mußte ich allerdings feststellen, daß die Mechanik für solche Aufgaben nicht flexibel genug ist.“ (S. 34)

Neben der Richtungsabhängigkeit des Schaltens gibt es noch zwei weitere Gründe für die Schwierigkeiten mit der Z1: Die Schalterketten, wie sie beim Addierer im Zusammenhang mit dem Übertrag vorkommen, sorgen dafür, dass je Schaltvorgang (in Abhängigkeit von den zu addierenden Zahlen) ziemlich große Massen durch eine antreibende Kraft zu bewegen sind. Außerdem dürften die Toleranzprobleme aufgrund der Verkettung von mehreren Schaltblechen mittels Schaltstiften den gleichzeitigen Übertrag über mehrere Stellen hinweg ziemlich unzuverlässig machen.

„Postskriptum“ (19.11.2010)

Inzwischen hat mich eine E-Mail von Winfried Görke erreicht, in der er eine Reihe von Vorschlägen macht, wie man aus meinen Notizen eine vorzeigbare Ausarbeitung machen könnte. Ich will diese Vorschläge hier anhängen, dass sie nicht in Vergessenheit geraten.

Wenn tatsächlich Modelle auf der Basis dieser Notizen entstehen – und Herr Samuel hat wissen lassen, dass er gern daran gehen würde –, dann sollten wir bei der Anwenderdokumentation die Korrekturvorschläge berücksichtigen. Hier der Text:

„Inzwischen habe ich Ihre Darstellung AnmerkungAddierer.pdf vom 1.11.10. genauer durchgesehen. Ich finde sie sehr gut, weil sie tatsächlich die Idee von Konrad Zuse genauer als das bisherige Modell in Bautzen oder Hünfeld wiedergibt. Man sollte also versuchen, es zu implementieren. Es stellt übrigens den Paralleladdierer mit seriellem Übertrag (Eisenbahnzug) dar, wie wir ihn in mancher Vorlesung 30 Jahre nach Zuse und später behandelt haben, wenn auch nicht als mechanische Implementierung. Die beiden vorgeschlagenen Stellen machen die Sache kompliziert genug, so daß sich auf weitere Stellen vorerst verzichten läßt. Ich schlage allerdings einige Punkte zur Verbesserung Ihrer Darstellung vor, die Sie einbauen müßten, weil ich ja nur eine pdf-Version habe, natürlich sofern Sie davon überzeugt sind:

1. Ich würde statt t lieber durchweg c oder ü für den Übertrag benutzen, um Verwechslungen mit den Taktsteuerungen zu vermeiden.
2. Die Bilder sollten mit 7 bis 10 numeriert werden, damit der Bezug erleichtert wird.
3. Dann würde ich in Bild 9 rechts c0, links c1 eintragen (oder cin bzw cout), außerdem f und g nach Bild 7 unterhalb der oberen UND- und ANTIV-Elemente.
4. Die gleichen Änderungen betreffen Bild 10, in dem Indizes für die Stellen 1 und 2 links auch für f, g und c verwendet werden sollten, also z.B. c0 rechts, c1 in der Mitte, c2 links.

Die Funktion müßte natürlich für die Benutzer beschrieben werden, z.B. in der folgenden Form:

1. Voreinstellungen: alle Eingänge +/- müssen gleich eingestellt sein, durch sie wird entweder die Addition oder die Subtraktion ausgewählt. Die Phase 1 erzeugt dann aus den bi deren Stellenkomplement.
2. Nach der Einstellung von allen ai, bi und c0 erzeugt Phase 2(I) die Variablen fi und gi.
3. Phase 3(II) erzeugt anschließend alle Eingänge für die si und alle ci,
4. Phase 4(III) erzeugt schließlich alle si aus den gi und ci.
5. c2m2m1 bilden nun das Resultat der Operation (d.h. 0 bis 7 je nach Eingängen). Nach dem Ablesen sind alle Schaltelemente zurückzusetzen, damit die nächste Operation durchgeführt werden kann.

Während die Addition am Modell leicht nachvollzogen werden kann, erfordert die Subtraktion einige Zusatzerläuterungen. Sie erfolgt hier durch Addition des Stellenkomplements, d.h. statt a - b wird a + Kompl(b) berechnet. Das erfordert je nach Ergebnis evtl. eine Korrektur,

also:

1. c2 wird 1 (Überlauf), Ergebnis ist positiv, erfordert eine Addition von 1 (oder die Überlaufstelle wird zum Ergebnis addiert).
2. c2 bleibt 0 (kein Überlauf, Ergebnis ist negativ, erfordert also eine Rückkomplentbildung,

um das (negative) Ergebnis zu verstehen.

Dies gilt nur für c0 = 0. Setzt man c0 = 1, wird analog zur Addition a + 1 + Kompl(b) berechnet, alle Ergebnisse erhöhen sich also um 1.

Wahrscheinlich wäre es nützlich eine Schautafel anzufertigen, die die 32 Eingangskombinationen für Addition und Subtraktion veranschaulicht.

Ohne sie wird man sich für das Verständnis gut an die Technische Informatik erinnern müssen, womit die meisten Museumsbesucher überfordert sein dürften."

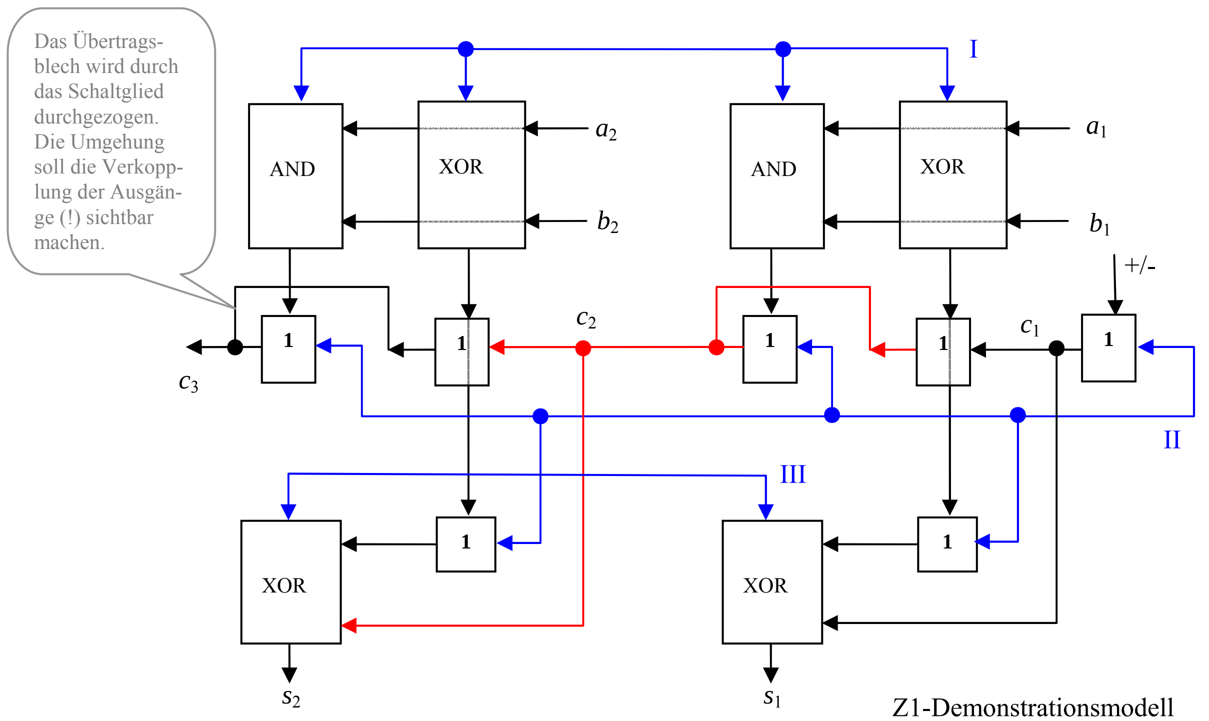
Demonstrationsmodell „Einschrittiger Übertrag“

Erstellung eines Prototyps (09.06.2011)

Vom 7. bis 9. Juni 2011 haben Andreas Samuel, Micha Schönberner und ich im Zuseum Bautzen einen Prototyp für den einschrittigen Übertrag so weit erarbeitet, dass die Funktionsfähigkeit des Modells ersichtlich wurde. Die folgenden Funktions- und Layout-Festlegungen wurden getroffen.

1. Die Invertierung der Eingänge für die Subtraktion wird weggelassen. Da bei Zuse die Eingabe der Werte und auch die Invertierung mit dem letzten Taktschritt (IV) des vorherigen Taktzyklus geschehen, wird durch den zusätzlichen Eingabeschritt (bei mir Schritt 1) die Klarheit des „Zuseschen Dreischritts“ vernebelt. Im Modell werden die Takte nur noch mit I, II und III bezeichnet.
2. Auch die Subtraktion lässt sich an dem Modell demonstrieren: Der Anwender muss die *b*-Bits invertiert eingeben und für einen Übertrag von rechts („aus dem Nichts“) sorgen: +/- wird mit dem Takt II zum Übertragsbit c_1. Für die taktgenaue Realisierung dieses Übertrags wird ein zusätzliches Durchschaltglied (1-Glied) benötigt.
3. Die folgende Schaltskizze gibt in etwa die Anordnung der Schaltelemente des Prototyps wieder.

In der Schaltskizze ist eins der Übertragsbleche rot hervorgehoben.



Z1-Demonstrationsmodell „Einschrittiger Übertrag“

Modell des mechanischen Addierers mit einschrittigem Übertrag (Z1)

Das Modell – „ein weiterer Prototyp“, wie Andreas Samuel meint – wurde am 25.8.2011 in Bautzen fertig gestellt, nach Fulda gebracht und am 26. und 27. August 2011 auf der fibit-Messe in Fulda erstmals einer Öffentlichkeit vorgestellt.

Skript für die Präsentation des Modells

Herzstück eines jeden Computers ist – neben den Möglichkeiten, Werte zu speichern und im Speicher nach rechts oder links zu verschieben – der *Addierer.* Alle anderen Rechenoperationen lassen sich auf die Addition zurückführen. Zum Beispiel die Subtraktion: Will man beispielsweise von der Zahl 2010 die Zahl 1642 abziehen, dann ersetze man im Subtrahenden jede Ziffer durch ihr Neunerkomplement, in unserem Fall liefert das die Zahl 8357. Diese Zahl erhöhe man um eins und Addiere sie zum Minuenden. Außerdem unterdrücke man den über vier Stellen hinausgehenden Übertrag: 2010+8357+1= 0368. Fertig.

Die Multiplikation ist in der Binärwelt des Computers nichts anderes als eine wiederholte Addition und die Division ist Ergebnis wiederholter Subtraktionen, wobei die Zahlen jeweils geeignet zu verschieben sind.

Additionen laufen im Computer ständig ab. Es kommt also darauf an, sie möglicht schnell durchzuführen. Und Konrad Zuse hat genau diesen Aspekt gründlich bedacht. Denn eins ist klar: Die schriftliche Addition, wie wir sie in der Schule gelernt haben, ist für den Computer zu langsam. Wenn wir beispielsweise zur Zahl 2011 die Zahl 989 addieren wollen, fangen wir bei der letzten Stelle an: 1+9 liefert 0 und den Übertrag 1, der auf der Zehnerstelle zu berücksichtigen ist. Dort addieren wir 1+8, berücksichtigen den Übertrag und erhalten wieder 0 mit dem Übertrag 1 für die Hunderterstelle. Hier ergibt sich dann der Wert 0+9 plus Übertrag 1, also wiederum 0 und der Übertrag 1 auf die Tausenderstelle. Hier steht dann 2+0 plus Übertrag, also 3. Ergebnis: 3000.

Wichtige Erkenntnis: Bevor man die Addition für eine beliebige Stelle durchführen kann, muss man wissen, ob es einen Übertrag von der vorhergehenden (rechts davon stehenden) Stelle gibt oder nicht. Die Rechnung läuft rein sequentiell ab, Stelle für Stelle von rechts nach links. Und das ist für einen Automaten, der im Grunde ständig diese Additionen durchzuführen hat, zu zeitaufwendig. Am liebsten würde man den Rechner beauftragen, die Summe für alle Stellen gleichzeitig – also parallel – zu berechnen.

Für diese Parallelisierung organisieren wir die Addition etwas anders, und zwar in drei Schritten

I Wir addieren unabhängig voneinander für jede Stelle. Wir berücksichtigen dabei keine Überträge, denn die kennen wir ja noch nicht. Aber wir merken uns zweierlei: Erstens, ob bei der stellenweisen Addition ein originärer Übertrag entsteht. Das ist immer dann der Fall, wenn die Addition einen Wert größer neun liefert. Und zweitens merken wir uns, wenn der Wert neun herausgekommen ist. In diesem Fall ist nämlich ein möglicher Übertrag von der vorhergehenden Stelle zu berücksichtigen, der dann sozusagen durchgekoppelt werden muss. Den *originären Übertrag* markieren wir mit einem liegenden Kreuz ×, die *Kopplung* mit einem Kreis o. Für unser Beispiel erhalten wir:

```
2011
0989
----
2990
 oo×
```

II Im zweiten Schritt wird nur die Zeile der Übertragsmarkierungen behandelt: Der Durchkopplungskreis wird durch ein Kreuz markiert, wenn rechts davon ein Kreuz steht (mit oder ohne Kreis). Die Kreuze (mit oder ohne Kreis) entsprechen nun jeweils den Überträgen auf die nächsthöhere Stelle. Nach dieser Regel wird aus der Markierungszeile o o× die Zeile ⊗⊗×. Diese Berechnung der Überträge geschieht in einem Schritt, daher der Name „einschrittiger Übertrag“.

III Im dritten Schritt ergibt sich das Ergebnis aus den stellenweisen Summen unter Berücksichtigung der jeweils rechts davon stehenden Überträge: Steht in der Spalte rechts daneben ein Kreuz (mit oder ohne Kreis), so ist jeweils noch 1 zu addieren:

```
2990
 ⊗⊗×
----
3000
```

Nun übertragen wir diese Prozedur ins Binäre: Wir addieren 3+1, binär: 11+01. Nach dem Schema geht das so:

I

```
11
01
--
10
o×
```

II

```
11
01
--
10
⊗×
```

III

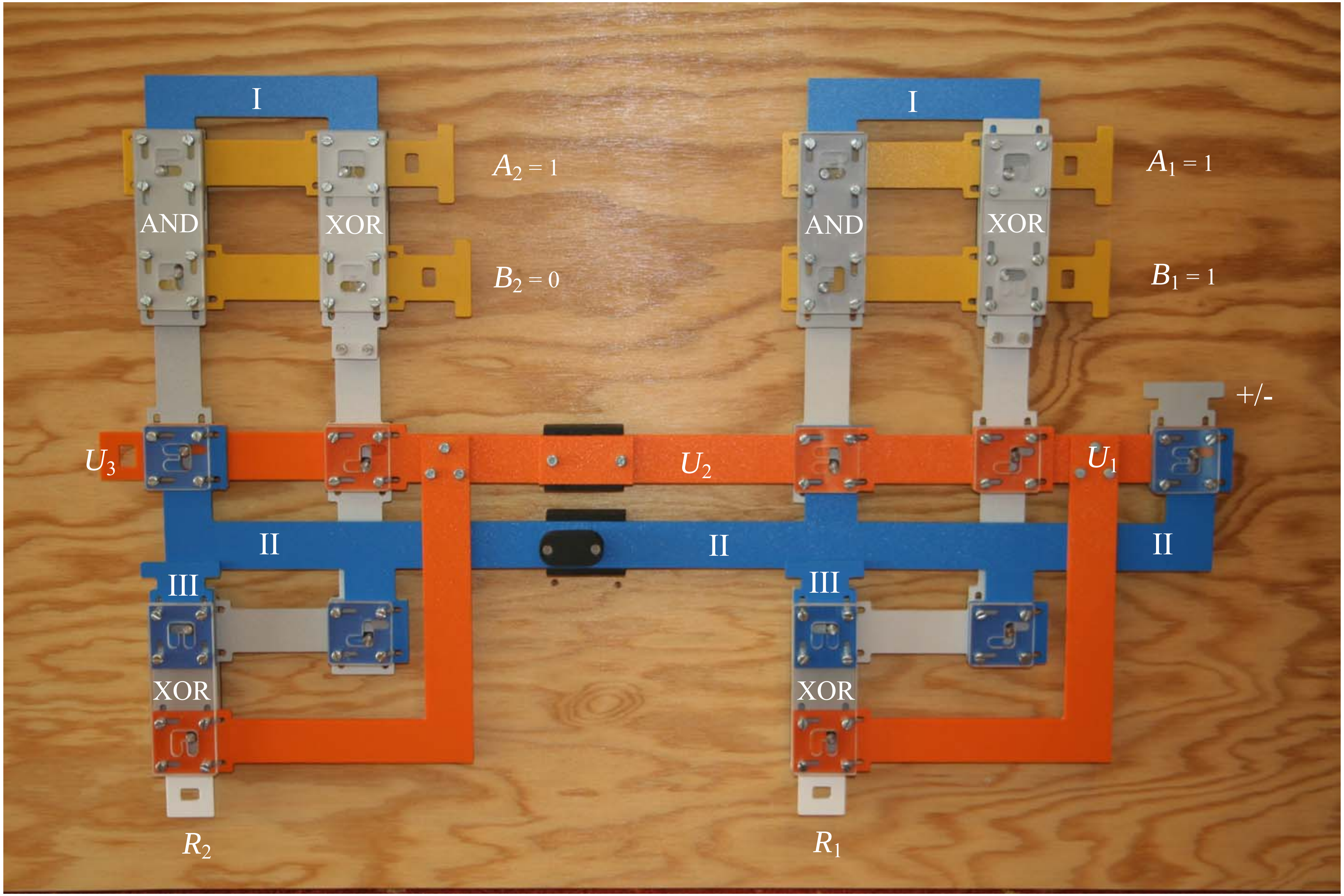
```
 11
 01
 10
 ⊗×
---
100
```

Das Kreuz steht jetzt jeweils für eine 1. In derselben Farbe grundierte Felder gehören zusammen. Nur zusammengehörigen Werte werden im dritten Schritt miteinander verknüpft, und zwar durch Addition ohne Übertrag, denn der wurde ja schon berücksichtigt! Das Resultat ist gleich 4 (binär: 100).

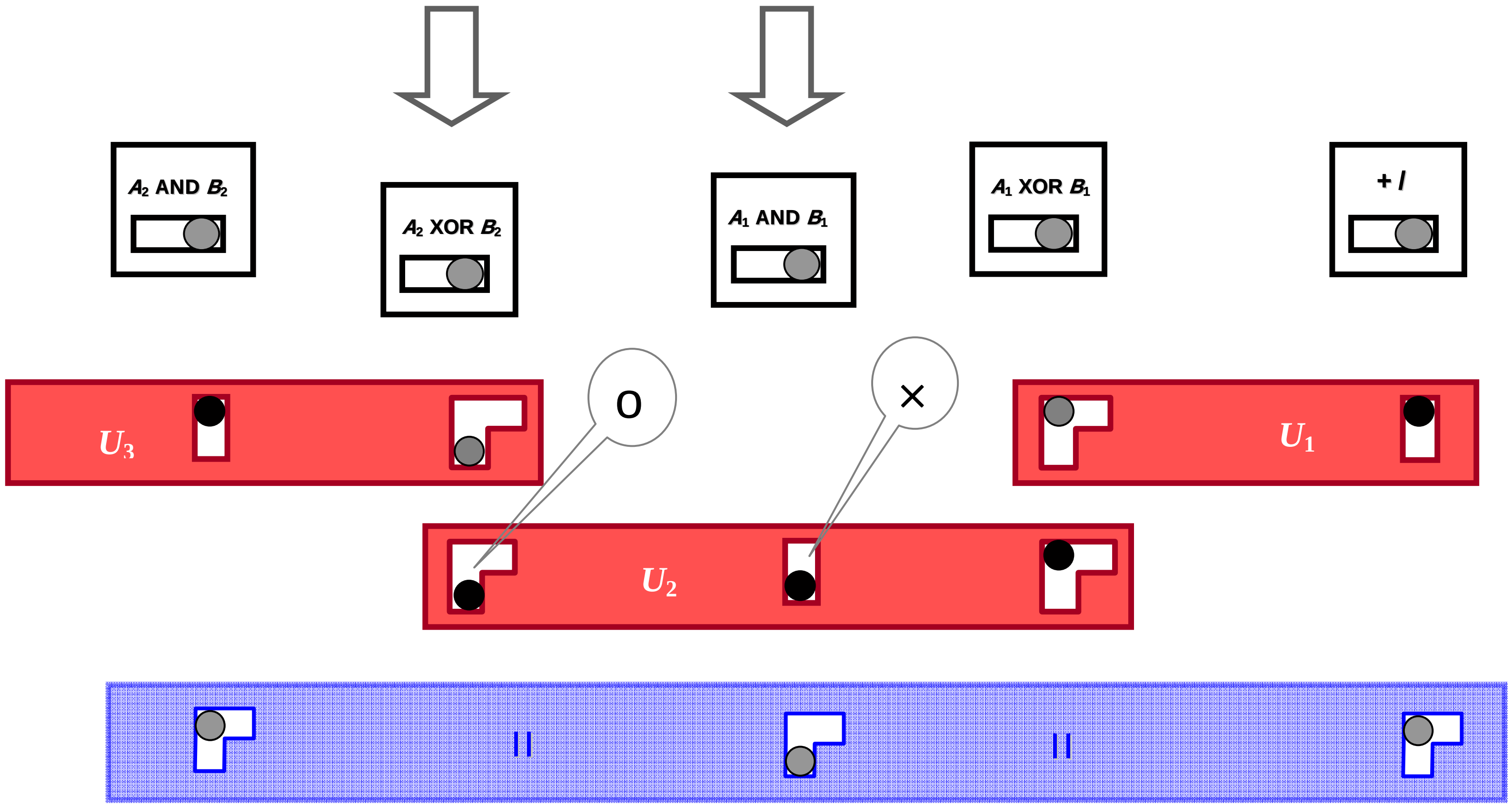
Dieses Beispiel wird nun mit dem mechanischen Addierermodell durchgerechnet, das auf der folgenden Seite abgebildet ist. Die Bezeichnungen der Variablen sind von Zuses Addierschaltung in abstrakter Schaltgliedtechnik übernommen worden.

Zuerst werden die Eingaben gesetzt: A_2=1, A_1=1, B_2=0, B_1=1. Takt I bewirkt die stellenweise Addition und stellt die Informationen für den originären Übertrag und die Übertragskopplung bereit.

Das Foto der folgenden Seite zeigt das Modell in genau diesem Zustand: Die Schieber der Eingänge A_2, A_1 und B_1 sind nach links geschoben, in Position 1. Entsprechend sind die diesen Eingängen zugeordneten Schaltstifte in den Verknüpfungsglieder in linker Position. Der Schieber des Eingangs B_2 ist herausgezogen und damit auf Position 0. Dementsprechend sind die zugeordneten Schaltstifte der Schaltglieder in rechter Position.



I
I
AND
XOR
AND
XOR
$A_2 = 1$
$B_2 = 0$
$A_1 = 1$
$B_1 = 1$
+/-
U_3
U_2
U_1
II
II
II
III
III
XOR
XOR
R_2
R_1



A2 AND B2
A2 XOR B2
A1 AND B1
A1 XOR B1
+ I
o
×
U3
U1
U2

Takt I ist schon erfolgt. Der dadurch erreichte Zustand ist an den Positionen der Schaltstifte in den Verknüpfungsgliedern der Übertragsbleche U_1, U_2 und U_3 abzulesen. Das übernächste Bild zeigt – in einem grafisch vereinfachten Explosionsbild – die dadurch hergestellten Kopplungsbeziehungen.

In Kommentarblasen eingetragen sind die Hinweise auf den „Merker" für den originären Übertrag, das liegende Kreuz ×, und den „Merker" für die Kopplung, der Kreis.

Wegen A_1 AND B_1 = 1 ist der entsprechende Schaltstift nach unten gewandert und hat das Taktblech II mit Übertragsblech U_2 gekoppelt. Das ist der originäre Übertrag: Eine Bewegung des Taktbleches II nach links bewirkt, dass sich auch das Übertragsblech U_2 nach links bewegt und in die 1-Position übergeht.

Die Verkopplung der Übertragsbleche U_2 und U_3 ist wegen A_2 XOR B_2 = 1 gegeben. Dadurch wird mit Takt II auch das Übertragsblech U_3 nach links in die 1-Position bewegt. Damit ist der Übertrag auf die Stelle mit der Wertigkeit 4 gegeben (2^2).

Die Verschiebung des Taktblechs II bewirkt also eine gleichsinnige und gleichzeitige Verschiebung der Übertragsbleche U_2 und U_3. Das ist der einschrittige Übertrag!

Durch den Takt II werden auch die Eingänge der unteren Verknüpfungsglieder gesetzt: Beim rechten XOR-Glied tut sich eingangsseitig nichts, da die stellenweise Addition den Wert 0 ergeben hat und der Übertrag von rechts (Übertragsblech U_1) ebenfalls gleich 0 ist.

Anders bei dem linken XOR-Glied: Sowohl die Addition als auch der Übertrag haben jeweils den Wert 1. Durch Takt II werden also die beiden Schaltstifte des XOR-Gliedes nach links verschoben.

Schließlich erhält man durch Bewegung der Taktbleche III nach unten die Ergebnisse. Da bei beiden Schaltgliedern die Eingänge äquivalent sind, bewirkt der Takt III keine Bewegung der Ausgangsbleche R_1 und R_2; sie verharren in der Position 0.

Damit haben wir das richtige Ergebnis: $U_3R_2R_1$ = 100 (dezimal: 4).

Das Modell ermöglicht auch die Subtraktion zweistelliger Zahlen: Der Subtrahend ist im Einerkomplement darzustellen und außerdem muss eine 1 addiert werden. Das geht mit dem Übertrag von rechts, der hier mit dem Zeichen +/- markiert ist.

Bedienungshinweise

Für jeden neuen Rechendurchgang müssen zunächst die Takte zurückgesetzt werden. Das muss zwingend in der umgekehrten Reihenfolge (III, II, I) geschehen.

Das Modell sollte horizontal aufliegen und nicht stehen.

Bei den Schiebevorgängen keine Gewalt anwenden! Wird ein Schaltstift durch einen Schaltvorgang nicht richtig mitgenommen, genügt es meist, von der Seite etwas gegen das Schaltglied zu drücken. Dann wird die Verklemmung aufgehoben und der Schaltstift kann sich aufgrund seines Eigengewichts aufrichten.

Für die Bewegung des Taktbleches II ist der schwarze Knopf in der Mitte der Modellplatte gedacht.

Notizen zur Historie

Konrad Zuse hatte für seinen geplanten Computer zunächst eine elektromechanische Lösung ins Auge gefasst (Der Computer – mein Lebenswerk, S. 34): „Das Fernmelderelais war mir schon bekannt; überschlägige Überlegungen zeigten aber, dass eine solche Rechenanlage Tausende von Relais benötigt hätte, also ein Zimmer voll Relaisschränke[…] Insbesondere die Speicherung Tausender von Zahlen war problematisch[…] Ich bemühte mich daher doch wieder um mechanische Lösungen. Es gelang mir, für das Speicherwerk eine kompakte mechanische Konstruktion zu entwickeln[…] Zwei Jahre hatte ich mich mit mechanischen Konstruktionen herumgequält, bis ich es doch aufgeben musste[…] Immerhin hatte ich auch an ihnen die Grundgesetze der Schaltungstechnik entwickeln und erproben können.“

Im „Anhang 2. Aufbau der Geräte“, ab Seite 170, kommt Zuse zunächst noch einmal auf seinen mechanischen Speicher zu sprechen. Und dann, auf Seite 172, spricht er über das Addierwerk und erläutert detailliert, wie er Schritt für Schritt zur Lösung für den Addierer mit einschrittigem Übertrag gekommen ist, und zwar in elektromagnetischer Technik, mit Relais also. Dieser Prozess mündet in das in abstrakter Schaltglieddarstellung entworfene Addierwerk.

Auf Seite 180 seiner Lebenserinnerungen schreibt Konrad Zuse: „Zunächst bot sich das elektromagnetische Relais als idealer Baustein an. Die schon erwähnten Erfolge mit der mechanischen Konstruktion des Speichers veranlassten mich aber, ein mechanisches Analogon für die Relais zu suchen. Auf diese Weise war es mögliche, einen neutralen Entwurf in einer ‚abstrakten’ Schaltgliedtechnik zu machen, der sowohl in die elektromagnetische als auch in die mechanische Schaltungstechnik übertragen werden konnte[…] Das mechanische Gerät Z1 und das spätere elektromechanische Gerät Z3 haben fast die gleichen abstrakten Grundschaltungen.“

Aus diesen Notizen schließe ich, dass das Rechenwerk zunächst in Relaistechnik geplant war und dass die Entwicklung des einschrittigen Übertrags auch in dieser Technik geschehen ist. Dafür spricht auch, dass die Relais die von Zuse erarbeitete Form des einschrittigen Übertrags problemlos und über viele Stellen hinweg ermöglichen. In anderen Technologien ist das nicht so ohne weiteres möglich.

Die Probleme mit der Relaislösung für den Speicher veranlassten Konrad Zuse jedoch, zunächst die rein mechanische Lösung anzugehen.

Aber diese Lösung stößt an Grenzen. Zuse war „die Mechanik für solche Aufgaben nicht flexibel genug“ (Seite 34 der Lebenserinnerungen). Anhand unseres mechanischen Addierermodells lässt sich gut sehen, welcher Art die Grenzen sind.

1. Die Richtungsabhängigkeit der Signale: Gehen die Eingänge horizontal in ein Schaltglied hinein, kommen die Ergebnisse vertikal gerichtet heraus und umgekehrt. Das sind lästige Randbedingungen für den Entwurf von Schaltnetzen und Schaltwerken.
2. Die Übertragsketten können sehr lang werden. Dann sind große Massen von nur einer Stelle aus zu bewegen: Eine „Lokomotive“ muss viele „Waggons“ ziehen. Das erfordert große Antriebs- und Federkräfte.
3. Die Verkopplung der Übertragsbleche ist mit Toleranzen behaftet. Bewegt die „Lokomotive“ den ersten und zweiten „Waggon“ noch leidlich gut, kann es am Ende einer Kette zu Bewegungen mit zu geringem Hub kommen: Die Schaltstifte rasten nicht mehr an der richtigen Stelle ein. Es kommt zu Verklemmungen und Verkantungen.

Diese Schwierigkeiten veranlassten Zuse, bei der Z3 dann doch wieder auf einen rein elektromechanischen Aufbau zurückzukommen. Später, in der Z4, kombinierte er die Vorteile beider Techniken, indem er den Speicher rein mechanisch und das Rechenwerk elektromechanisch realisierte. Und dieser Rechner war dann ja ein großer Erfolg, auch in kommerzieller Hinsicht. Außerdem schrieb er Mathematikgeschichte[3].

---

[3] Bruderer, Herbert: Konrad Zuse und die Schweiz. Wer hat den Computer erfunden? Oldenbourg Verlag, München 2012

Die Realisierung eines Addierwerks mit einschrittigem Übertrag stößt nicht nur in der Mechanik an Grenzen, sondern auch in der modernen integrierten Transisitortechnik[4].

Überlegungen zur Mathematik- und Informatik-Didaktik

Der Schüler sieht im Computer ein Gerät, das Ergebnisse aufgrund irgendwelcher okkulter Prozesse liefert – Voodoo sozusagen. Die Grundlagen des Rechnens sind dem geistigen Zugriff des Schülers entzogen. Dadurch entwickelt sich in der Ausbildung ein Trend zur „Oberflächenkompetenz".

Diese Entfremdung von den Grundlagen und vom mathematischen Geschehen erschwert das Verstehen von Problemlösungsprozessen und letztlich kann sie die korrekte Interpretation der Ergebnisse verhindern.

Dem kann eine „Mathematik zum Anfassen" entgegenwirken. Und in einer solchen Mathematik zum Anfassen hat das mechanische Addierermodell seinen Platz. An dem Modell lässt sich zeigen und *begreifen*,

1. wie sich Logik prinzipiell realisieren lässt,
2. wie diese Logik im ersten Computer auch tatsächlich realisiert worden ist,
3. wie Logik den Aufbau komplexer Rechenwerke ermöglicht, und
4. an welche Grenzen man technologiebedingt stoßen kann.

[4] Liebig, H.; Menge, M.: Zuse und der nicht-einschrittige Übertrag. Informatik Spektrum (23. Dezember 2000), S. 398- 402