Das Innenleben eines Rechners: Schaltkreise

Ideen und Konzepte der Informatik

Kurt Mehlhorn

Schaltkreise

In dieser Einheit lernen wir das Innenleben von Rechnern genauer kennen. Insbesonder werden wir uns ansehen, wie ein Schaltkreis für die Addition im Detail aussieht. In der Einheit über Binärzahlen lernten wir den Volladdierer kennen. Ein Volladdierer addiert drei Bits und stellt das Ergebnis als 2-stellige Binärzahl dar. Das hintere Bit ist das Summenbit und das vordere Bit ist der Übertrag in die nächste Stelle. Wir sahen dann weiter, wie man aus n Volladdierern einen Schaltkreis für die Addition von n-stelligen Binärzahlen bauen kann. Heute sehen wir an, wie man einen Volladdierer aus Und-, Oder- und Nicht-Gattern zusammenbauen kann. Und-, Oder- und Nichtgatter operieren auf Paaren von Bits bzw. auf einem einzelnen Bit in ganz einfacher Weise.

Am Schluss der Einheit sehen wir uns dann die anderen Komponenten eines Rechners an: Speicher, Rechenwerk, und Befehlszyklus.

Aufbau der Einheit Innenleben

- Gatter: Und, Oder, Negation
- Ein Schaltkreis, der entscheidet, ob zwei Bits gleich sind.
- Halbaddierer und Volladierer
- Ein Schaltkreis für die Addition
- Speicher, Rechenwerk, Befehlszyklus

Und-, Oder- und Nicht-Gatter

Ein Schaltnetz besteht aus Gattern. Die einfachsten sind Und-Gatter ($\wedge$), Oder-Gatter ($\vee$), und Nicht-Gatter ($\neg$). Sie operieren auf den booleschen Werten (auch Bits genannt) 0 und 1 gemäß folgenden Regeln.

x	y	$x \wedge y$	$x \vee y$	$\neg x$
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

Der Ausgang eines Und-Gatters ist also genau dann 1, wenn beide Eingänge 1 sind,
der Ausgang eines Oder-Gatters ist 1, wenn mindestens ein Eingang 1 ist,
das Nicht-Gatter dreht den Eingang um.

Ein Schaltkreis für das Exklusive-Oder

Mit Hilfe der Gatter können wir kompliziertere Funktionen bilden, z. B. $x \oplus y = (x \vee y) \wedge (\neg(x \wedge y))$. Hier ist die Funktionstafel, schrittweise aufgebaut.

x	y	$x \vee y$	$x \wedge y$	$\neg(x \wedge y)$	$(x \vee y) \wedge \neg(x \wedge y)$
0	0	0	0	1	0
0	1	1	0	1	1
1	0	1	0	1	1
1	1	1	1	0	0

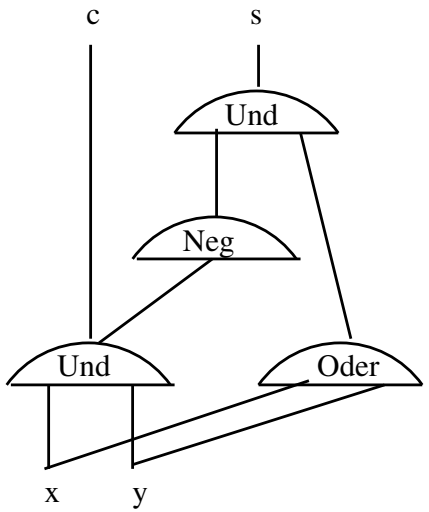
Die ersten beiden Spalten enthalten die vier möglichen Kombinationen für x und y. Die dritte und vierte Spalten zeigen die Werte von $x \vee y$ und $x \wedge y$. Die fünfte Spalte ist die Negation der vierten und die letzte Spalte ist das Und der dritten und fünften.

Die Funktion $x \oplus y$ ist unter den Namen Ungleich, Exklusives Oder, Summe modulo 2, Entweder-Oder bekannt.

Das Innenleben eines Halbaddierer (HA)

Ein Halbaddierer hat zwei Eingabebits x und y und berechnet die Binärdarstellung der Summe $x + y$,
also 00, wenn beide Bits null sind,
01, wenn genau eines der Bits eins ist, und
10, wenn beide Bits eins sind.

Das vordere Ausgabebit nennen wir c (Carry) und das hintere s (Sum).



Das Bit c ist eins, wenn beide Eingaben eins sind, also $c = x \wedge y$.

$$x + y = 2c + s$$

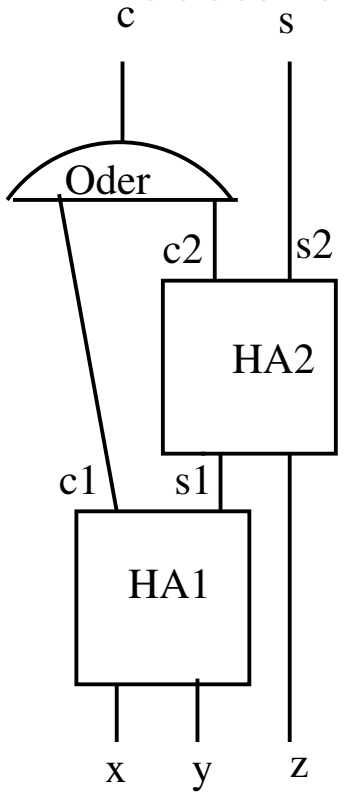
Das Bit s ist eins, wenn genau eine Eingabe eins ist, d.h. $s = x \oplus y = (x \vee y) \wedge (\neg(x \wedge y))$.

Ein Volladdierer (VA)

Ein Volladdierer hat drei Eingabebits x, y und z und berechnet ihre Summe als zweistellige Binärzahl mit den Ziffern c und s.

$$x + y + z = 2c + s$$

Er besteht aus zwei HAn und einem Oder-Gatter.



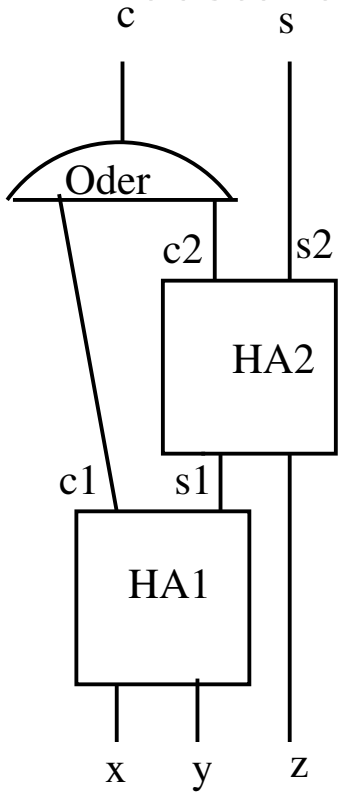
- Alle Eingaben Null: Dann $c_1 = s_1 = 0$ und daher $c_2 = s_2 = 0$ und daher $c = s = 0$.
- Drei Einsen: Dann $(c_1, s_1) = (1, 0)$ und daher $(c_2, s_2) = (0, 1)$. Dann $(c, s) = (1, 1)$.

Ein Volladdierer (VA)

Ein Volladdierer hat drei Eingabebits x, y und z und berechnet ihre Summe als zweistellige Binärzahl mit den Ziffern c und s.

$$x + y + z = 2c + s$$

Er besteht aus zwei HAn und einem Oder-Gatter.



- Genau eine Eins:

  Wenn $z = 1$ und $x = y = 0$, $(c_1, s_1) = (0, 0)$ und $(c_2, s_2) = (0, 1)$. Also $(c, s) = (0, 1)$.

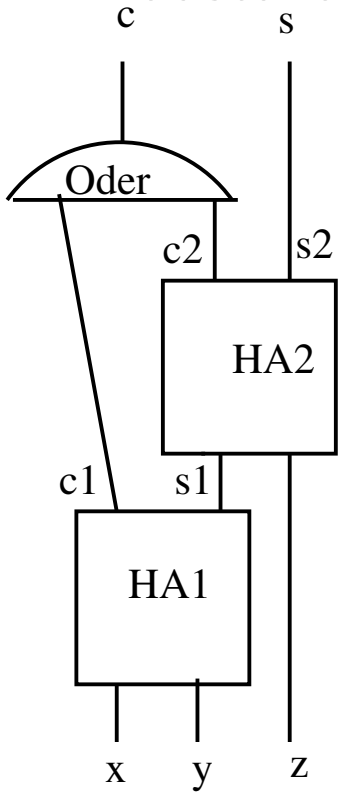
  Falls $z = 0$ und $x + y = 1$, dann $(c_1, s_1) = (0, 1)$ und daher $(c_2, s_2) = (0, 1)$. Also $(c, s) = (0, 1)$.

Ein Volladdierer (VA)

Ein Volladdierer hat drei Eingabebits x, y und z und berechnet ihre Summe als zweistellige Binärzahl mit den Ziffern c und s.

$$x + y + z = 2c + s$$

Er besteht aus zwei HAn und einem Oder-Gatter.



- Genau zwei Einsen:

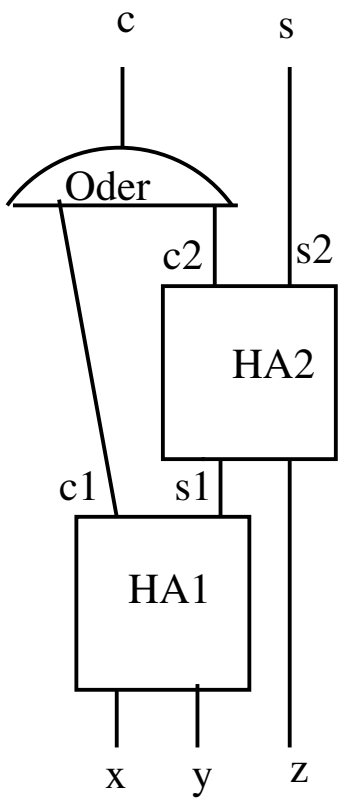
  Falls $z = 1$ und $x + y = 1$, dann $(c_1, s_1) = (0, 1)$ und daher $(c_2, s_2) = (1, 0)$. Also $(c, s) = (1, 0)$.

  Falls z Null ist und $x = y = 1$, dann $(c_1, s_1) = (1, 0)$ und daher $(c_2, s_2) = (0, 0)$. Damit $(c, s) = (1, 0)$.

Elegantere Argumentation für die Korrektheit des VA

Ein Volladdierer hat drei Eingabebits x, y und z und berechnet ihre Summe als zweistellige Binärzahl mit den Ziffern c und s.

$$x + y + z = 2c + s$$



Wir haben

$$x + y = 2c_1 + s_1 \quad \text{und} \quad s_1 + z = 2c_2 + s_2$$

und daher

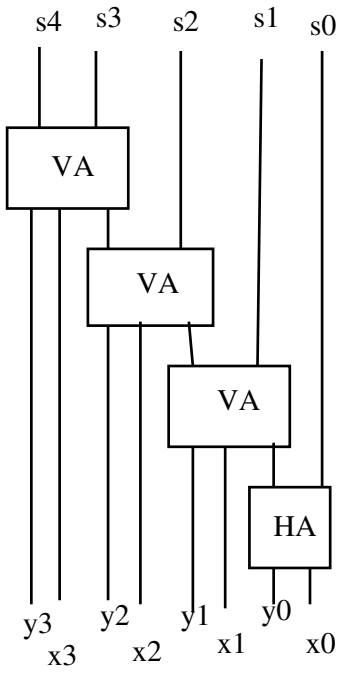
$$\begin{aligned} x + y + z &= 2c_1 + s_1 + z = 2c_1 + 2c_2 + s_2 \\ &= 2(c_1 + c_2) + s_2. \end{aligned}$$

Die Ausgabe des Volladierers ist $(c_1 \vee c_2, s_2)$. Nun sind c_1 und c_2 nie gleichzeitig 1, denn wenn $c_1 = 1$ dann $s_1 = 0$ und daher $c_2 = 0$.

Also $c_1 + c_2 = c_1 \vee c_2$.

Addition von Binärzahlen

Ein Schaltkreis mit 8 Eingängen und 5 Ausgängen, der die Binärzahlen $x_3x_2x_1x_0$ und $y_3y_2y_1y_0$ addiert.



Erinnern Sie sich, wie man zwei Zahlen addiert. Man schreibt die Zahlen übereinander.

x_3	x_2	x_1	x_0
y_3	y_2	y_1	y_0

Dann geht man die beiden Zahlen von rechts nach links durch. Man addiert x_0 und y_0 und erhält eine Summe s_0 und einen Übertrag c_0. Den Übertrag schreibt man in die Spalte mit den Ziffern x_1 und y_1. Also

x_3	x_2	x_1	x_0
y_3	y_2	y_1	y_0
		c_0	
			s_0

Genau das macht auch der HA in dem Schaltkreis. Nun addiert man die 3 Bits x_1, y_1 und c_0 und erhält eine Summe s_1 und einen Übertrag c_1. Genau das macht auch der erste VA. Wir erhalten:

x_3	x_2	x_1	x_0
y_3	y_2	y_1	y_0
	c_1	c_0	
		s_1	s_0

Und so weiter.

Die anderen Komponenten eines Rechners I

Der Speicher besteht aus Speicherzellen, die jeweils ein Bit speichern können. Jeweils 32 oder 64 Speicherzellen sind zu einem Wort zusammengefasst. Die Worte sind (implizit) von 0 an durchnummeriert: $M[0]$, $M[1]$, $M[2]$, $M[3]$, ...

Der Speicher ist mit einer Ansteuerung ausgestattet. Gegeben eine Adresse i (als Binärzahl) verbindet die Ansteuerung das Wort $M[i]$ mit dem Ausgang des Speichers. Dann kann man $M[i]$ lesen oder schreiben.

Ansteuerung hat die Form eines Baums.

Die anderen Komponenten eines Rechners II

- Rechenwerk: besteht aus Schaltkreisen für die Grundoperationen Addition, Multiplikation, Vergleich von zwei Worten auf $=$, $\leq$, ... Ähnlich zur Addition.
- Befehlszyklus:
  - Eine Uhr (Quartz) gibt den Takt vor. Schlägt ca. eine Milliarde mal pro Sekunde. Schaltet Wege frei und schließt sie wieder (Ampel).
  - Der Befehlszähler gibt vor, welche Programmzeile auszuführen ist.
  - Man sieht nach, welcher Befehl in der Zeile steht und auf welchen Speicherzellen der Befehl auszuführen ist.
  - Man schaltet die entsprechenden Verbindungen zum richtigen Schaltkreis im Rechenwerk frei und führt den Befehl aus.
  - Beispiel $R1 \leftarrow R1 + M[5]$.