OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

INF FAKULTÄT FÜR INFORMATIK

Rechnerarithmetik

Martin Glauer, Marco Dankel, Stephan Möhring, Sarah Puschmann

ZAHLENSYSTEME & HORNER

Zahlensysteme

- Die bekannten Zahlensysteme beruhen auf Basen
  - Jede Zahl lässt sich als Polynom schreiben (ist die Basis): $$f(x) = a_n x^n + \ldots + a_1 x + a_0$$
- Bsp.: $672_8 = 6*8^2 + 2*8 + 7 = 407_{10}$

  Basis

- 2n-1 Multiplikationen
- Wie kann man das verbessern?

Horner-Schema

- Lösung: Einzelne x ausklammern
- $f(x) = a_0 + x(a_1 + \ \dots + x(a_{n-1} + x * a_n) \dots)$
- Nur noch n Multiplikationen
  - So kann jede Zahl schnell(er) in das Dezimalsystem ‚übersetzt' werden
- Intuitivere Berechnung durch graphische Darstellung
- Problem: Noch immer ‚große' Multiplikationen

Kaskadiertes Horner-Schema

- Lösung: Aufspaltung auf Potenzen der Basis (Einer = 10^0, Zehner = 10^1, …)

5	6	2
0	0	7

50 = 6*8 +2 (Basis 8)

10^2		**10^1**		**10^0**			
						6	**8^2**
				5	6	2	**8^1**
		4	5	0	0	7	**8^0**
4		0		7			

OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG INF FAKULTÄT FÜR INFORMATIK

Der Rückweg

Überführe n vom Dekadischen in ein beliebiges Zahlensystem p mit DIV und MOD:

1. n = X*p + R

   X = (n DIV p)

   R = (n MOD p)   = 1. Stelle der neuen Zahl

2. Wende Schritt 1 rekursiv an.
   Verwende X anstelle von N.

OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG INF FAKULTÄT FÜR INFORMATIK

Formal

n = (n DIV p)*p + n MOD p

= (((n DIV p) DIV p) * p + (n DIV p) MOD p) * p + n MOD p

= ...

Am Beispiel: (p = 7)

n = 241

= 34 *7 + 3

= ($\underline{4}$ *7 + 6) *7 + 3

= ($\underline{(0*7 + 4)}$ *7 + 6) *7 + 3

⇈ ⇈ ⇈

Anwendung

Inverses Kaskadiertes Horner-Schema

RECHNEN MIT BINÄREN ZAHLEN

OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG

INF

FAKULTÄT FÜR INFORMATIK

Schriftliche Addition

Mit binären- und Dezimalzahlen

		15
	+	18
Übertrag		1
	=	33

		100101
	+	101010
Übertrag		1
	=	1101111

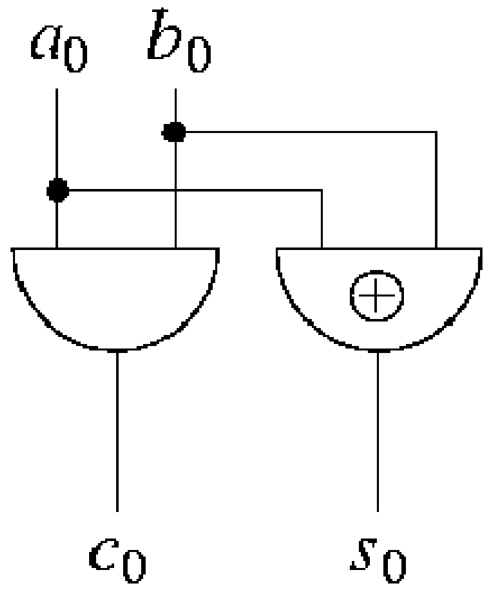
- Überträge werden auch als „Carrys“ bezeichnet
- Bei Dualzahlen entsteht ein Übertrag bei „2“, Dezimalzahlen haben den Übertrag bei „10“ (vgl. Stellenwertsystem)
- Wenn durch einen Übertrag die Dualzahl zu lang wird, kann es zu einem „Overflow“ kommen

Umsetzung in Hardware

Vierteladdierer	Besteht aus einem XOR-Element	Addiert zwei Zahlen ohne Überträge zu beachten
Halbaddierer	Besteht aus einem XOR- und einem UND-Element	Erzeugt auch den Übertrag, rechnet aber nicht damit
Volladdierer	Besteht aus 2 Halbaddierern und einem UND-Element	Erzeugt den Übertrag und rechnet auch damit

Halbaddierer

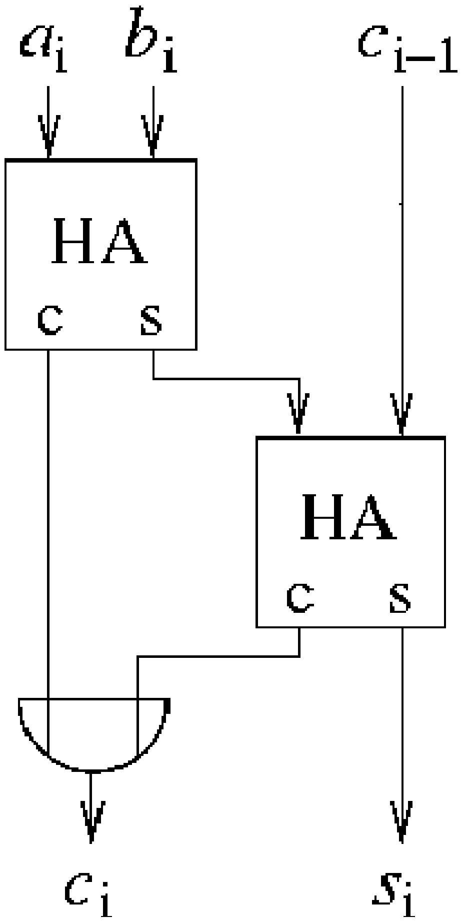
- a und b sind die Eingänge, d.h. die Zahlen an der gleichen Stelle der zu addierenden Zahlen
- c ist der evtl. entstehende Übertrag
- s ist das Ergebnis an der Stelle



OTTO VON GUERICKE UNIVERSITÄT MAGDEBURG INF FAKULTÄT FÜR INFORMATIK

Volladdierer

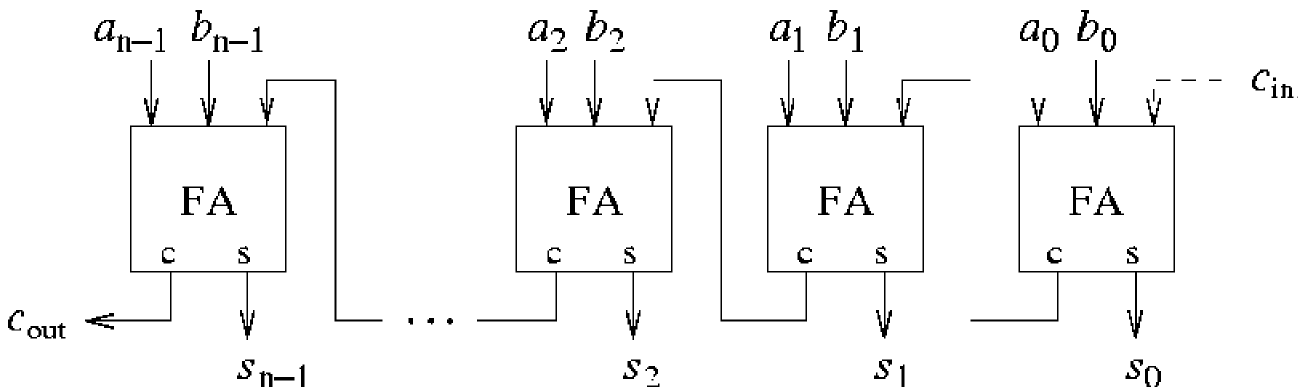
- a und b sind die Eingänge (vgl. Halbaddierer)
- c(i-1) ist der Übertrag von der vorhergehenden Stelle
- c(i) ist der Übertrag für die nächste Stelle
- s ist das Ergebnis an der betreffenden Stelle



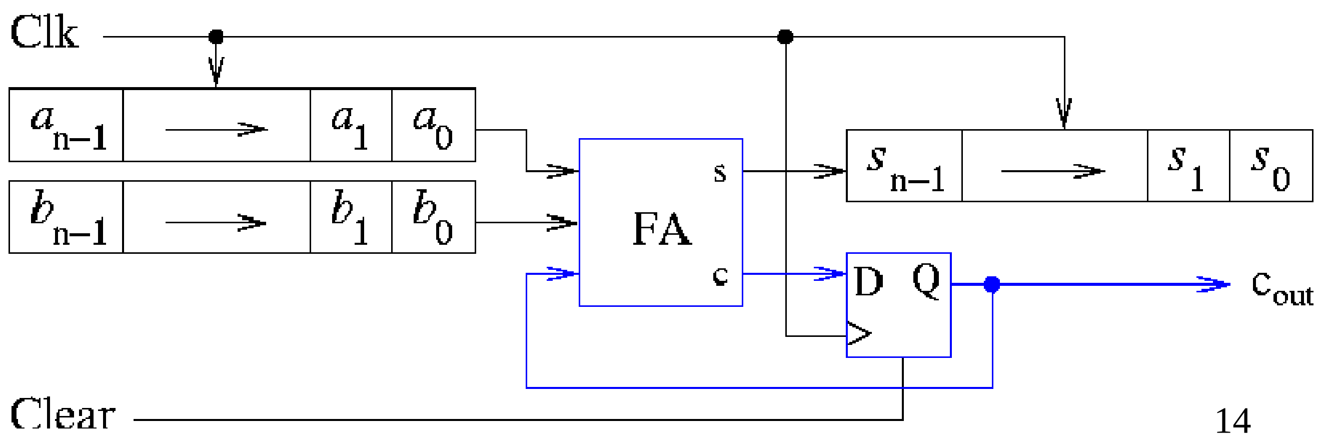
Beispiele

Addierwerke

- Einfache Addition
- Vorher festgelegte Länge der Zahlen
- Immer gleicher Aufwand
- Aufwändiger Aufbau



- Beliebige Länge
- Aufwand entsprechend der Zahlenlänge
- Weniger Bauteile benötigt



ENDE