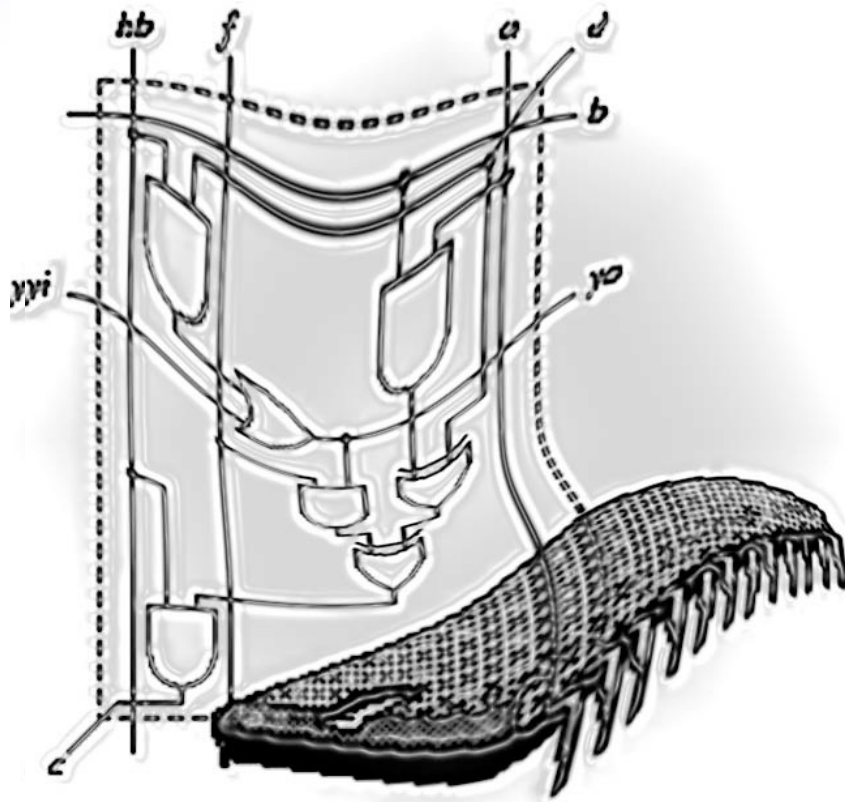


Technische Informatik I



Kapitel 5

Standardschaltnetze

Prof. Dr. Dirk W. Hoffmann



Inhalt und Lernziele

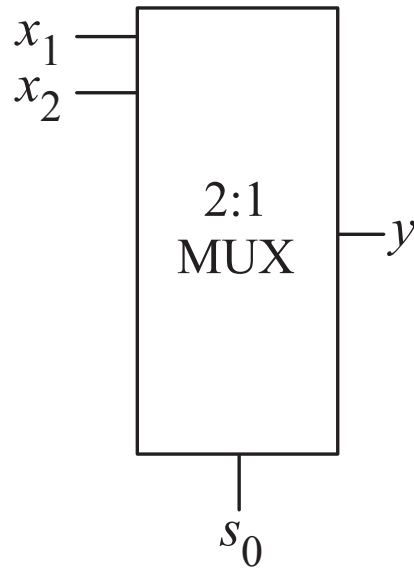
■ Inhalt

- Vorstellung der wichtigsten Standardelemente
- Nur kombinatorische Logik, kein Gedächtnis
 - Multiplexer
 - Demultiplexer
 - PALs, PLAs
 - Halbaddierer, Volladdierer
 - Carry-Ripple-Addierer
 - Carry-Look-Ahead-Addierer
 - ALU

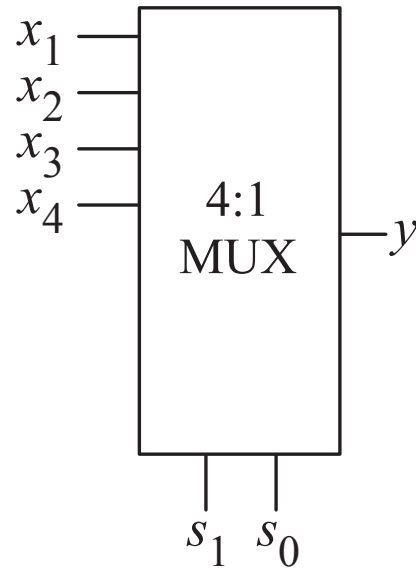
■ Lernziele

- Kenntnis über Aufbau und Funktion der Schaltelemente

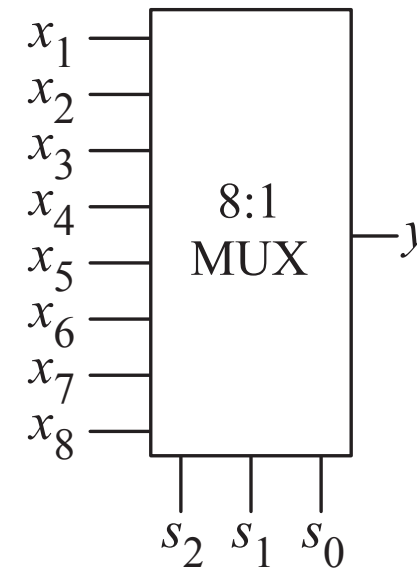
Multiplexer



s_0	y
0	x_1
1	x_2

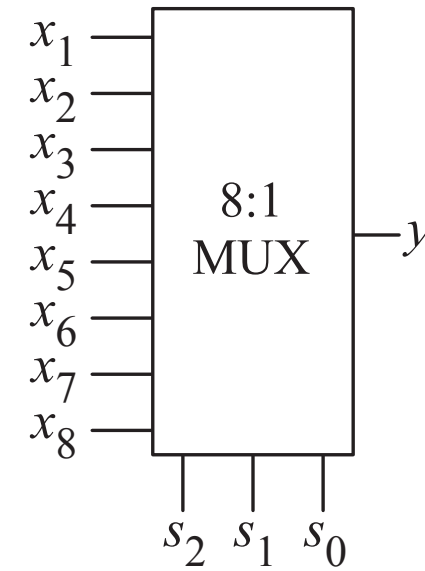
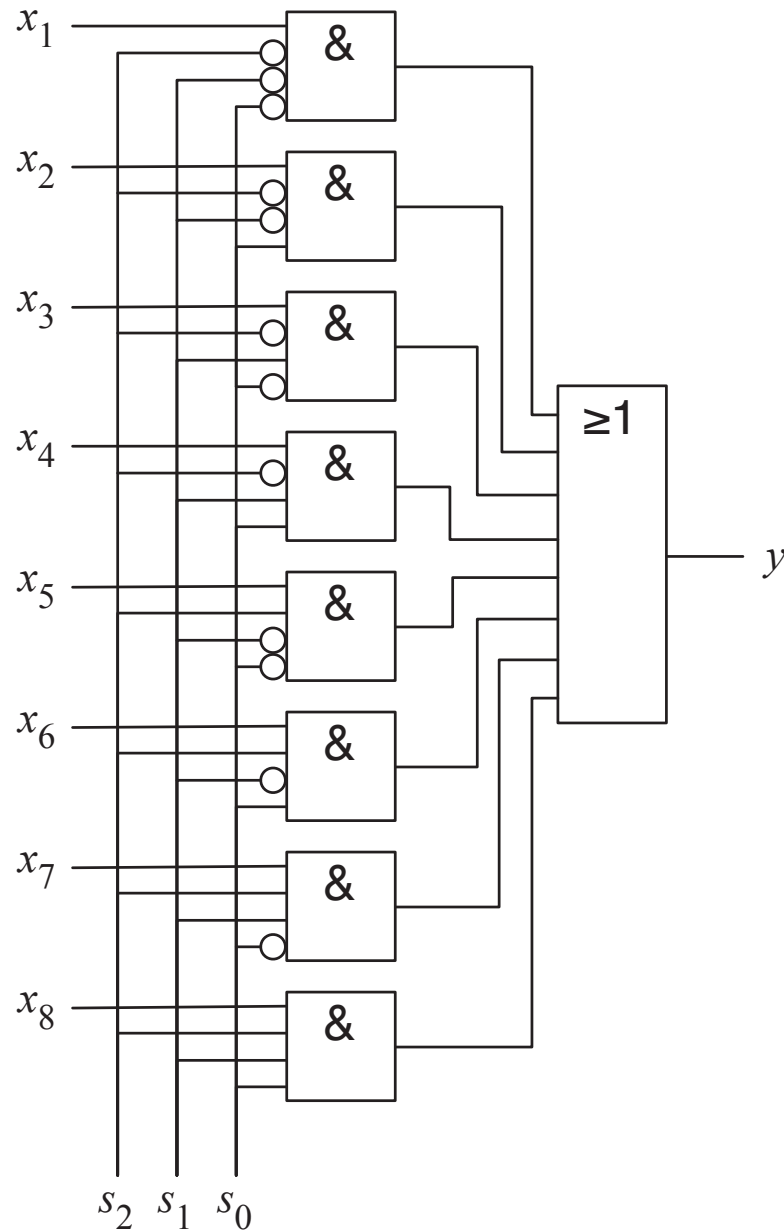


s_1	s_0	y
0	0	x_1
0	1	x_2
1	0	x_3
1	1	x_4



s_2	s_1	s_0	y
0	0	0	x_1
0	0	1	x_2
0	1	0	x_3
0	1	1	x_4
1	0	0	x_5
1	0	1	x_6
1	1	0	x_7
1	1	1	x_8

Multiplexer-Implementierung

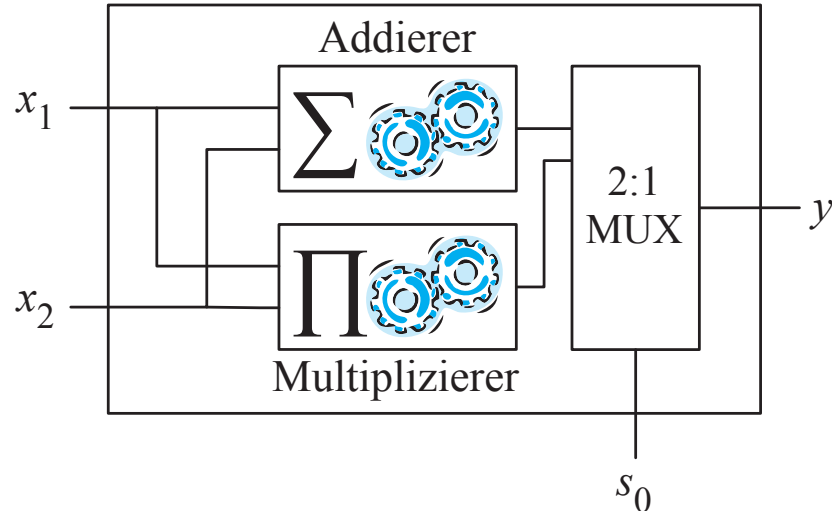


s_2	s_1	s_0	y
0	0	0	x_1
0	0	1	x_2
0	1	0	x_3
0	1	1	x_4
1	0	0	x_5
1	0	1	x_6
1	1	0	x_7
1	1	1	x_8

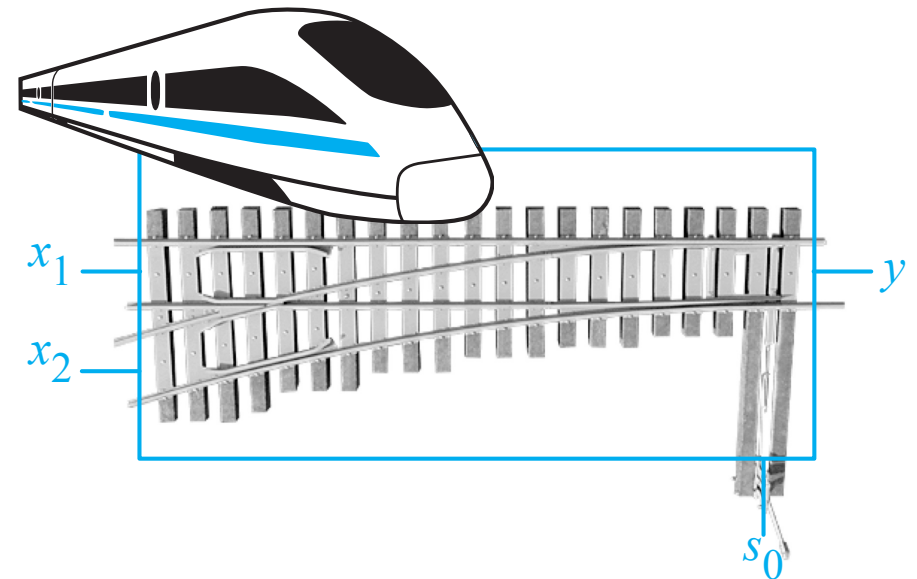
Multiplexer-Anwendungen

Steuerung des Datenflusses

Hier steuert der Multiplexer, ob das Ergebnis des Addierers oder des Multiplizierers weiterverwendet werden soll.



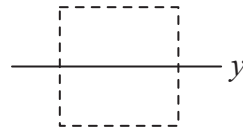
Analogie: Schienenverkehr
Multiplexer entspricht
der Weichensteuerung



Schaltnetzsynthese mit Multiplexern

Erster Schritt

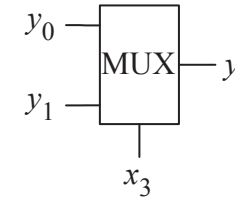
x_3	x_2	x_1	y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Zweiter Schritt

x_2	x_1	y_0
0	0	0
0	1	0
1	0	1
1	1	0

x_2	x_1	y_1
0	0	0
0	1	1
1	0	1
1	1	1



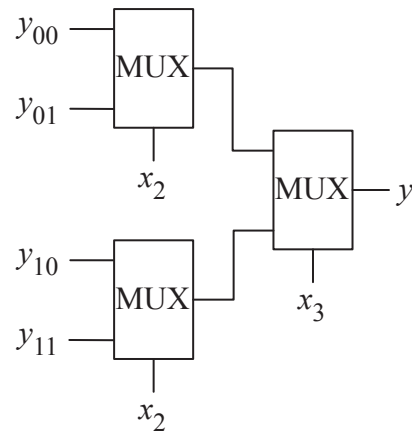
Dritter Schritt

x_1	y_{00}
0	0
1	0

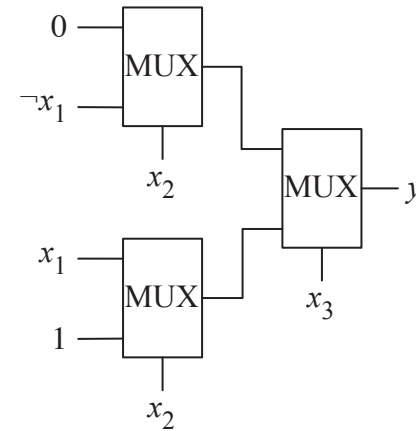
x_1	y_{01}
0	1
1	0

x_1	y_{10}
0	0
1	1

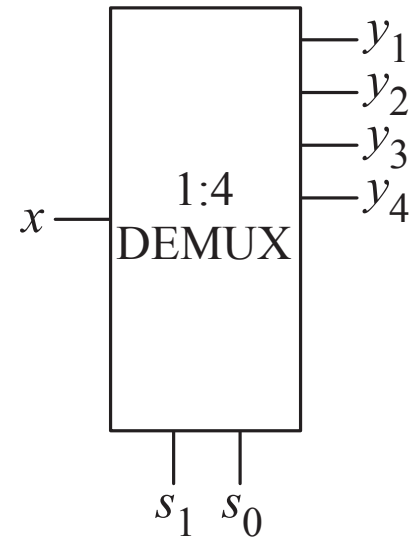
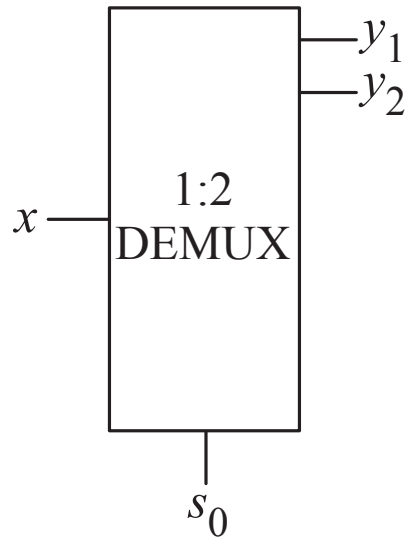
x_1	y_{11}
0	1
1	1



Vierter Schritt



Demultiplexer



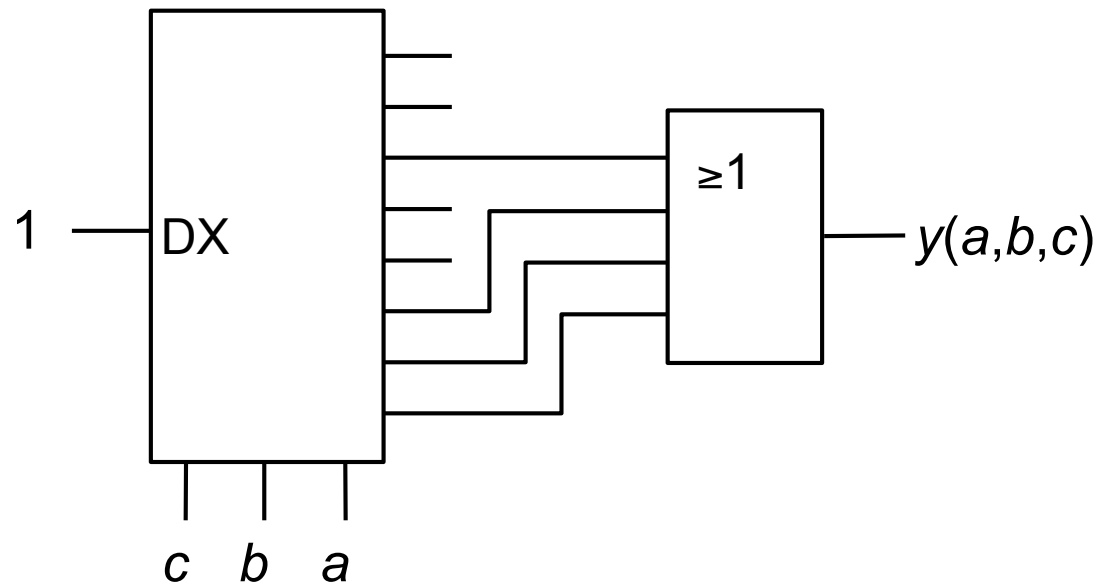
s_0	y_1	y_2
0	x	0
1	0	x

s_1	s_0	y_1	y_2	y_3	y_4
0	0	x	0	0	0
0	1	0	x	0	0
1	0	0	0	x	0
1	1	0	0	0	x

Demultiplexer-Anwendungen (2)

- Realisierung von logischen Funktionen
 - Der Decoder erzeugt alle 2^n Minterme seiner n Steuerleitungen
 - Ein ODER-Gatter erzeugt die Einsmenge der Funktion

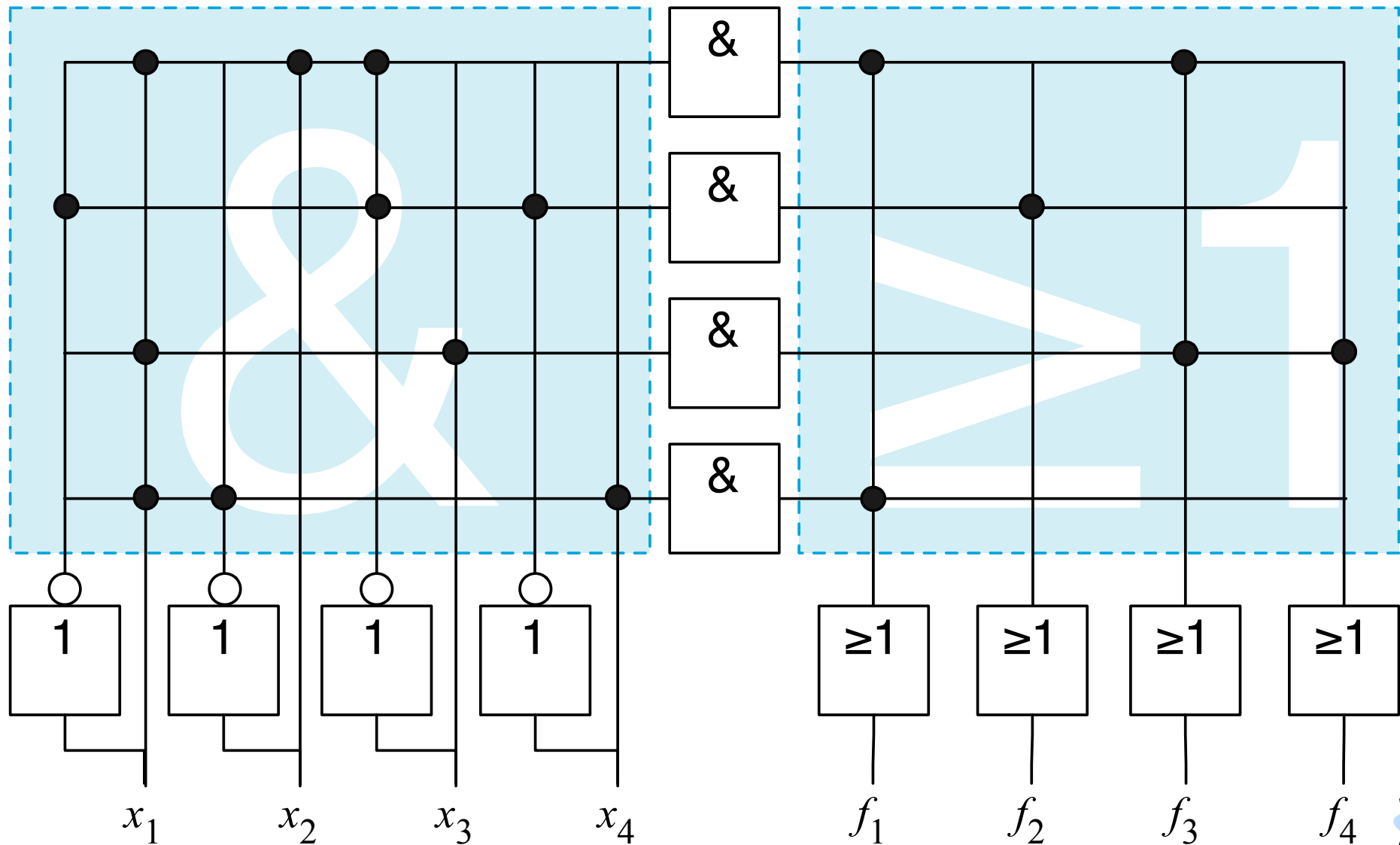
c	b	a	y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Beispiel

UND-Matrix

ODER-Matrix

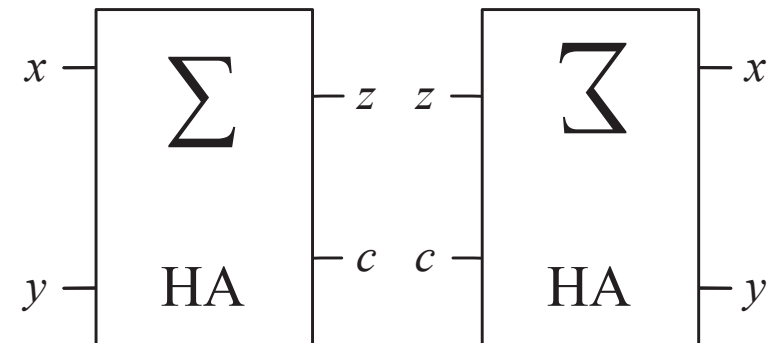


Addierer

▪ Halbaddierer

- Addiert zwei Binärziffern
- Ergebnis ist der Summenwert und ein Übertrag
- Es werden 2 Eingänge und zwei Ausgänge benötigt

Fall 1	Fall 2
$\begin{array}{r} 0 \\ + 0 \\ \hline = 0 \ 0 \end{array}$	$\begin{array}{r} 0 \\ + 1 \\ \hline = 0 \ 1 \end{array}$
Fall 3	Fall 4
$\begin{array}{r} 1 \\ + 0 \\ \hline = 0 \ 1 \end{array}$	$\begin{array}{r} 1 \\ + 1 \\ \hline = 1 \ 0 \end{array}$



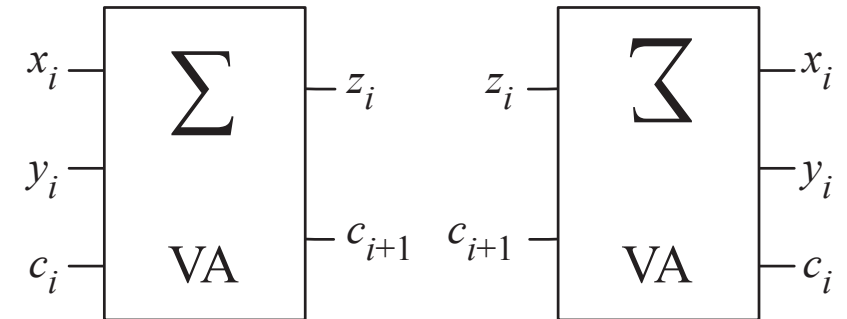
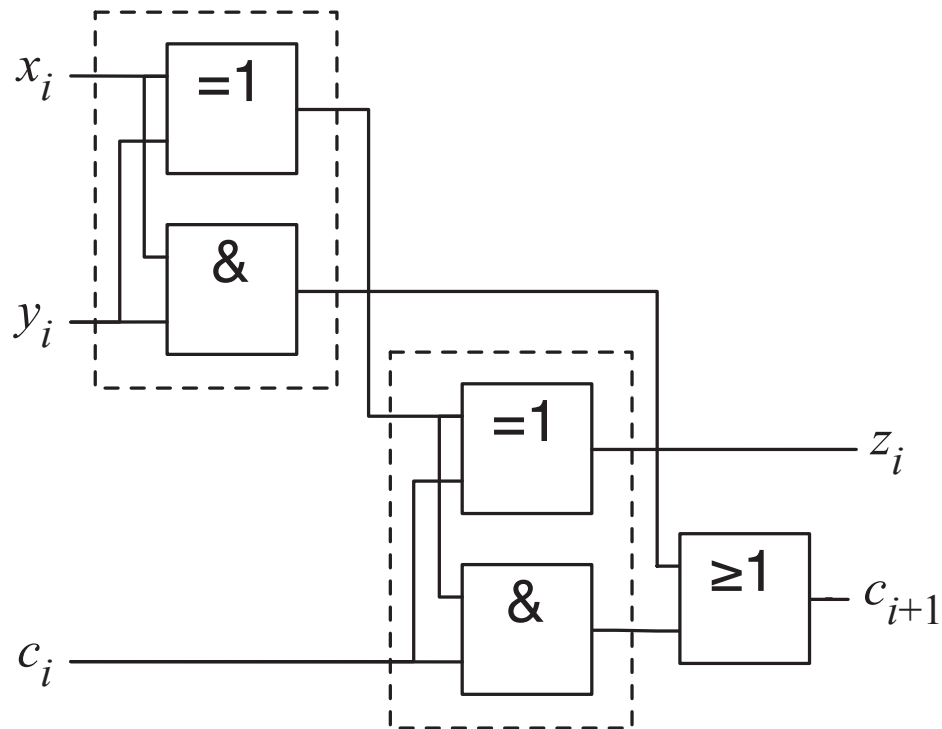
x	y	z	c
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



Volladdierer

■ Volladdierer

- Addiert zwei Binärziffern unter Berücksichtigung eines Übertrags
- Ziel: Addition mehrstelliger Binärzahlen

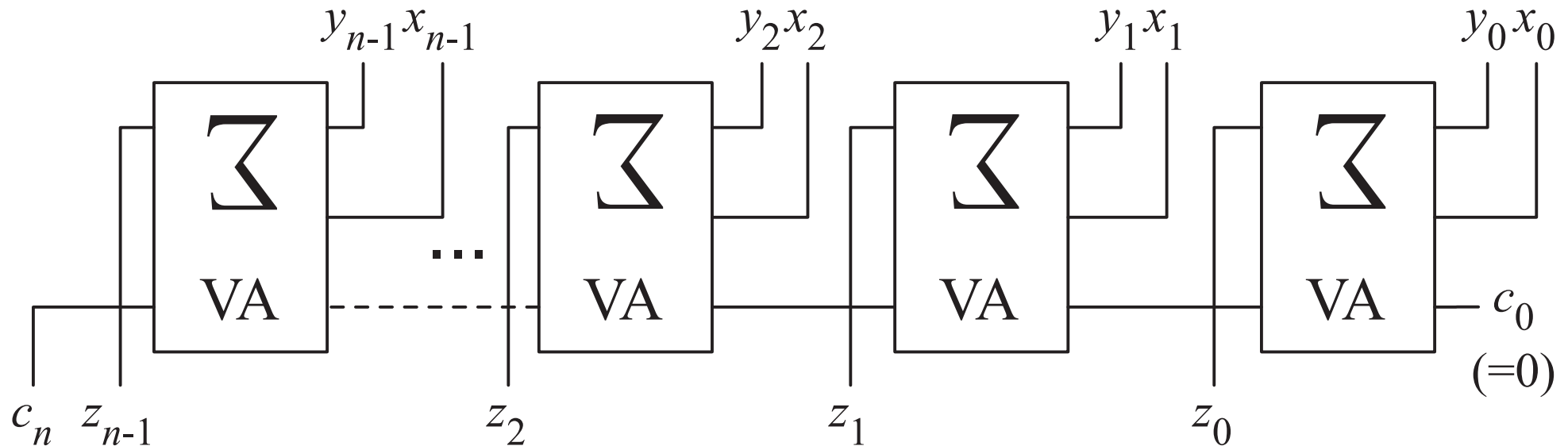


x_i	y_i	c_i	z_i	c_{i+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Addierer

▪ Carry-Ripple-Addierer

- Idee: Sequenzielle Aneinanderreihung von Volladdierern



▪ Vorteile

- Ökonomisch: Pro Bit genau ein Volladdierer
- Anzahl der Gatter steigt linear mit der Bitbreite

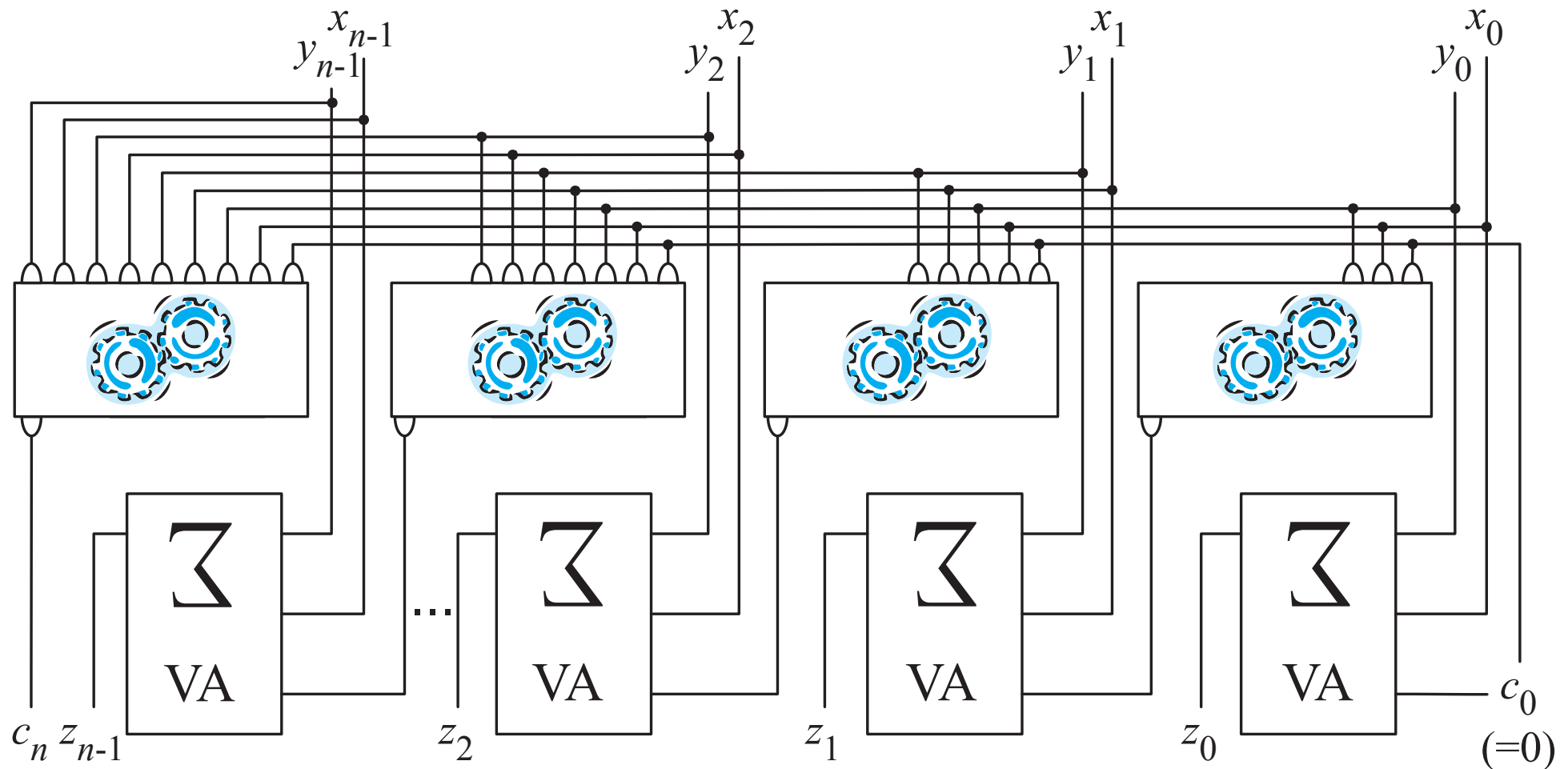
▪ Nachteile

- Hohe Laufzeit (Carry-Bit wird von rechts nach links durchgereicht)

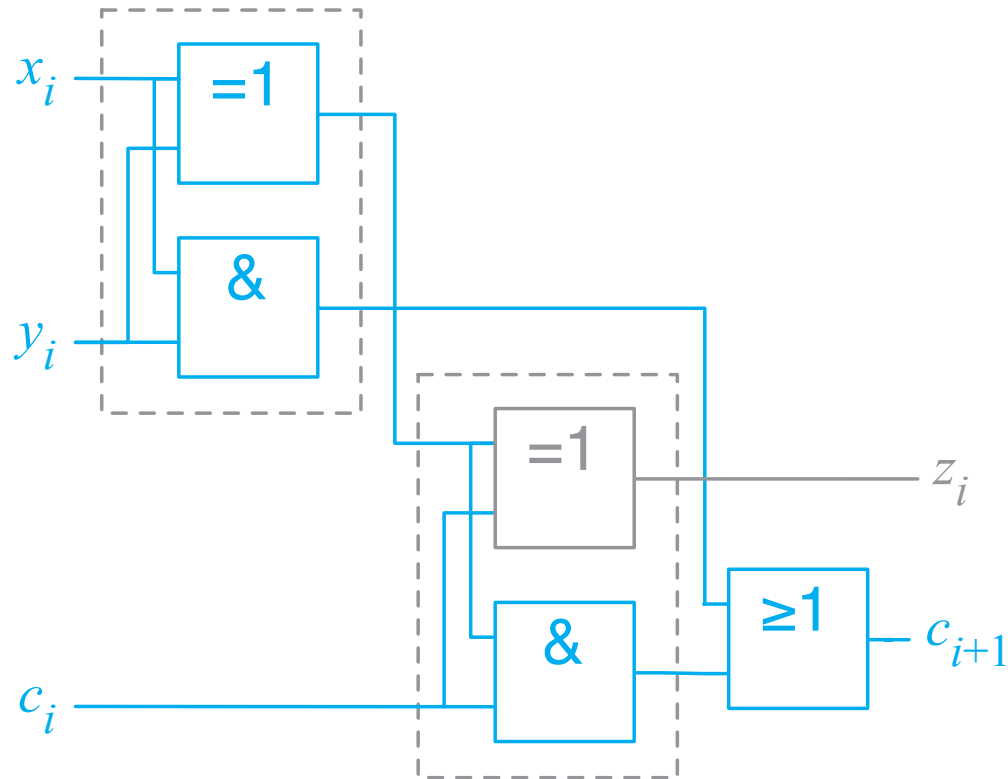
Addierer

- Verbesserungsidee

- Parallele Berechnung aller Übertrage mit zweistufigen Schaltnetzen



Vorausberechnung des Carry-Bits



- Rekursionsschema:

$$c_{i+1} = (x_i \wedge y_i) \vee (c_i \wedge (x_i \leftrightarrow y_i))$$

- Abkürzende Schreibweise:

$$g_i := x_i \wedge y_i$$

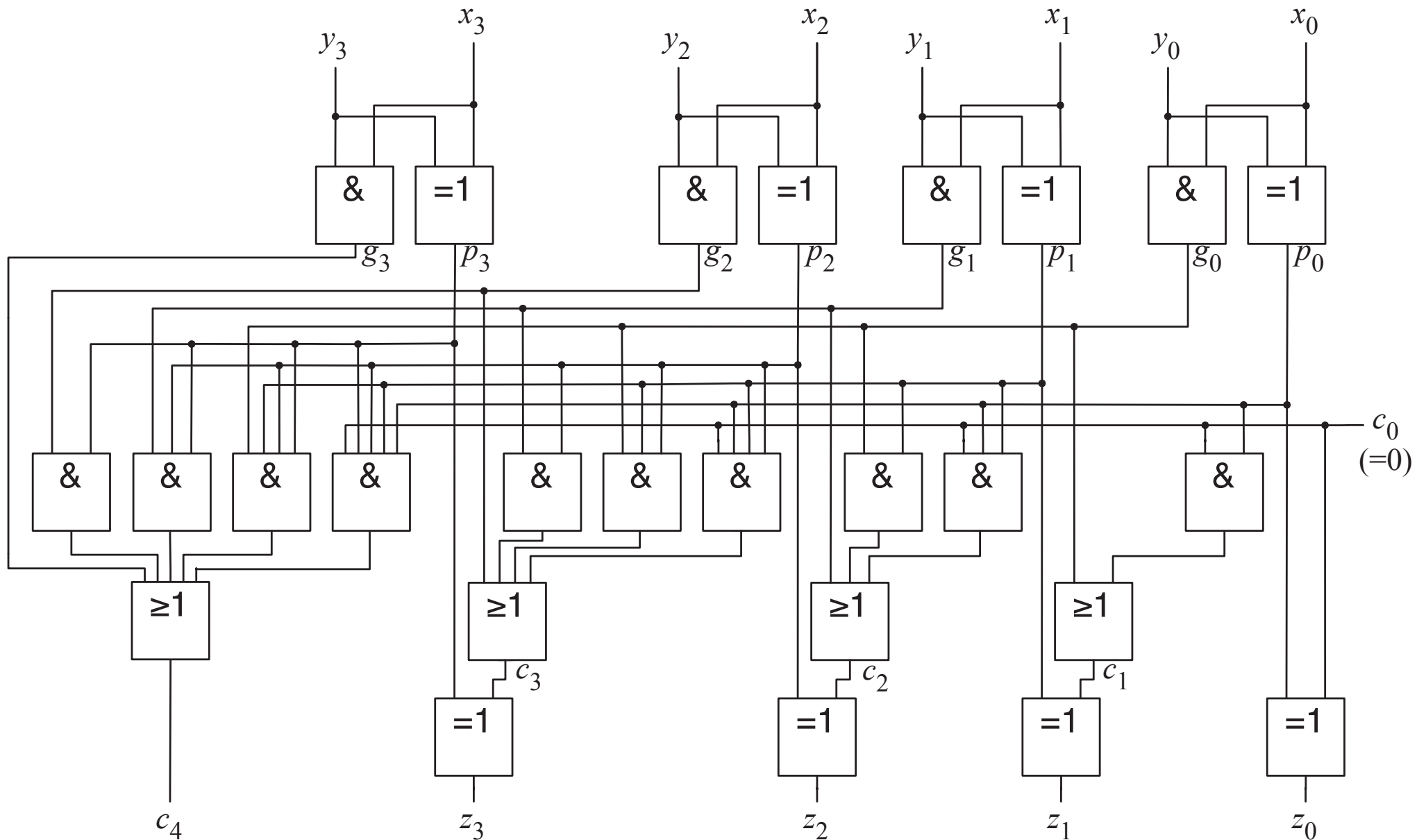
$$p_i := x_i \leftrightarrow y_i$$

- Dann gilt:

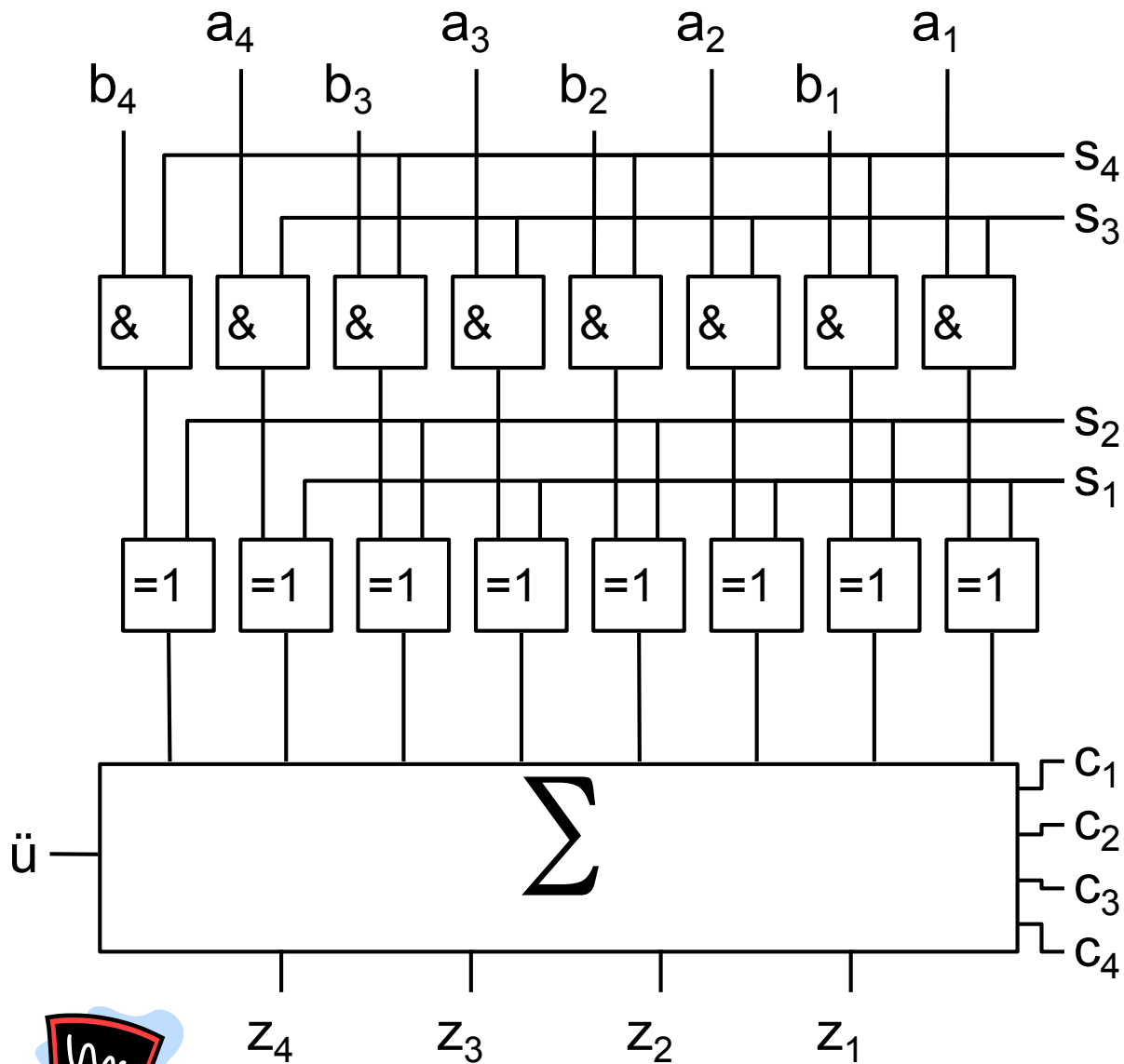
$$c_{i+1} = g_i \vee (c_i \wedge p_i)$$



Carry-look-ahead-Addierer



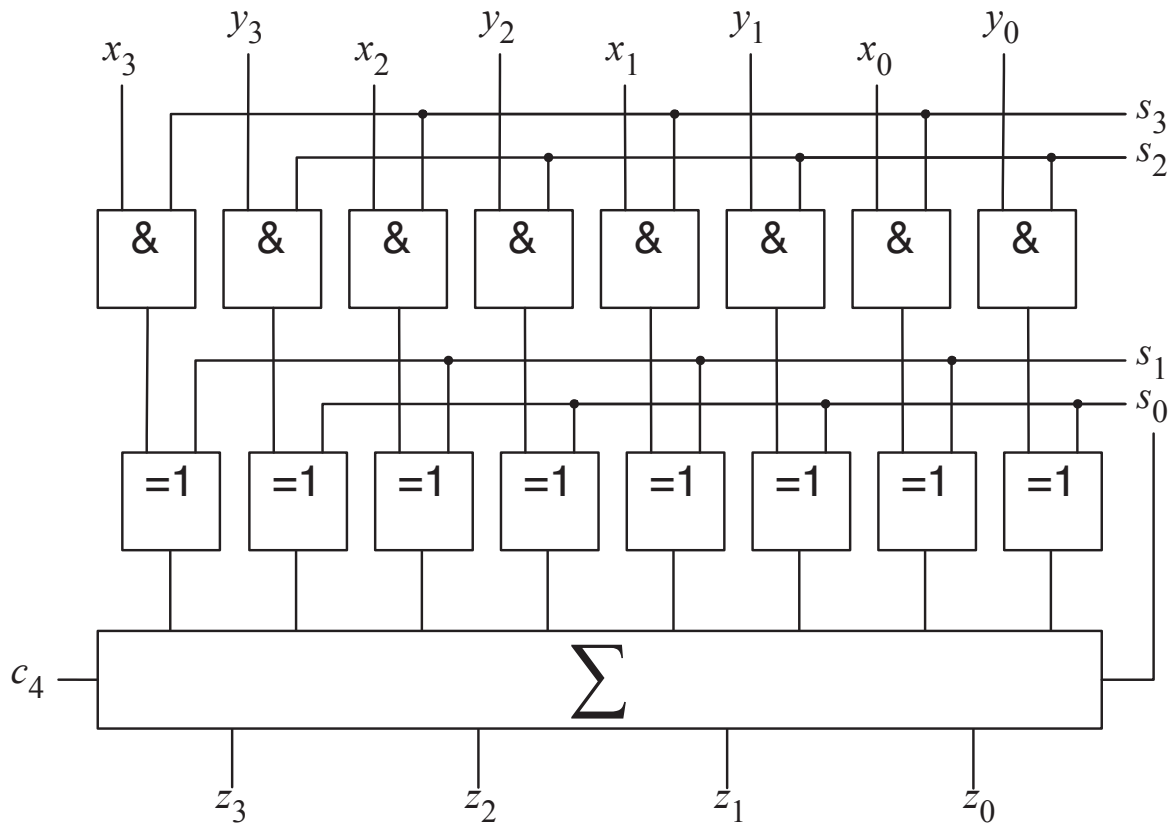
Einfache Arithmetikeinheit



S ₄	S ₃	S ₂	S ₁	Z
0	0	0	0	C
0	0	0	1	C-1
0	0	1	0	C-1
0	0	1	1	C-2
0	1	0	0	C+A
0	1	0	1	C-A-1
0	1	1	0	C+A-1
0	1	1	1	C-A-2
1	0	0	0	C+B
1	0	0	1	C+B-1
1	0	1	0	C-B-1
1	0	1	1	C-B-2
1	1	0	0	C+A+B
1	1	0	1	C+B-A-1
1	1	1	0	C+A-B-1
1	1	1	1	C-A-B-2



Eine einfache arithmetisch-logische Einheit



	s_3	s_2	s_1	s_0	z
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	-1
3	0	0	1	1	-1
4	0	1	0	0	y
5	0	1	0	1	$-y$
6	0	1	1	0	$y-1$
7	0	1	1	1	$-y-1$
8	1	0	0	0	x
9	1	0	0	1	x
10	1	0	1	0	$-x-1$
11	1	0	1	1	$-x-1$
12	1	1	0	0	$x+y$
13	1	1	0	1	$x-y$
14	1	1	1	0	$-x+y-1$
15	1	1	1	1	$-x-y-1$