

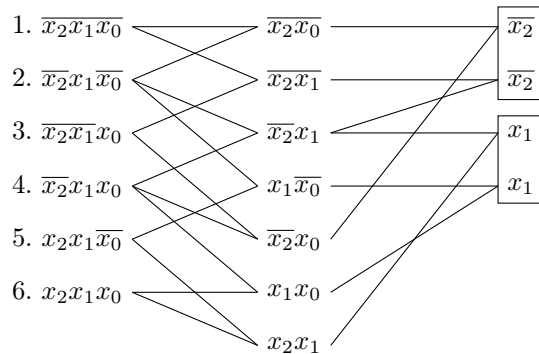
Grundlagen der Rechnerarchitektur: Übungsblatt 7

Alexander Waldenmaier, Maryia Masla

8. Januar 2021

Aufgabe 1: Wiederholung: Quine McCluskey

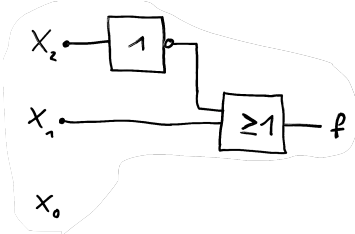
a)



	1	2	3	4	5	6
$\overline{x_2}$	+	+	+	+		
x_1		+		+	+	+

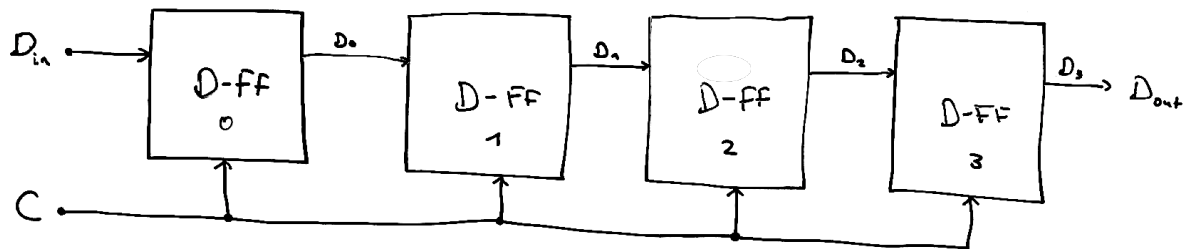
$$\Rightarrow f(x_2, x_1, x_0) = \overline{x_2} + x_1$$

b) Gatterschaltung:



Aufgabe 2: Schieberegister und Asynchronzähler

a) 4-bit FIFO-Schieberegister:



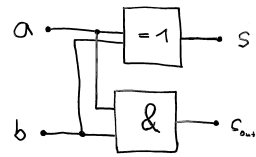
Aufgabe 4: Ripple-Carry-Addierer

a) Halbaddierer:

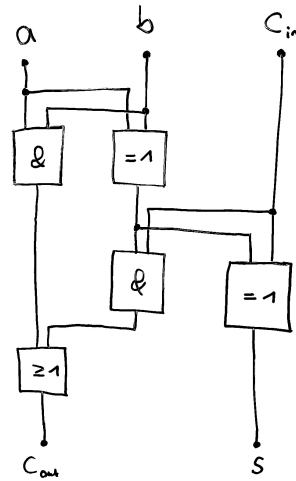
a	b	c_{out}	s
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

$$s(a, b) = a \oplus b$$

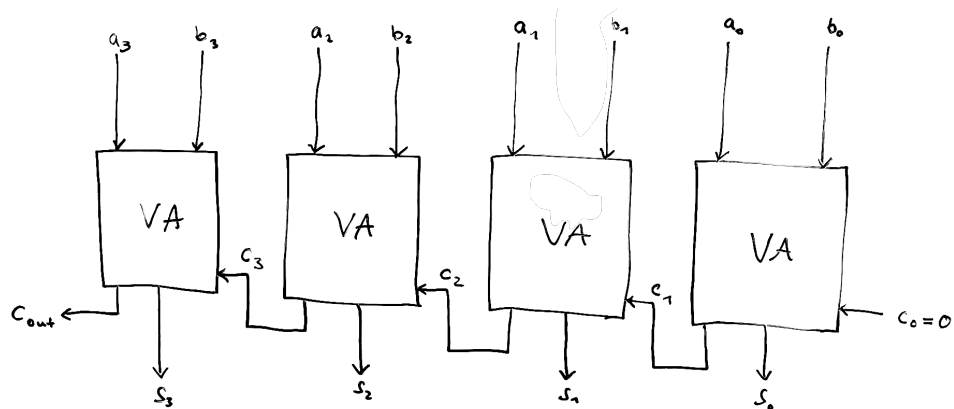
$$c_{out}(a, b) = ab$$



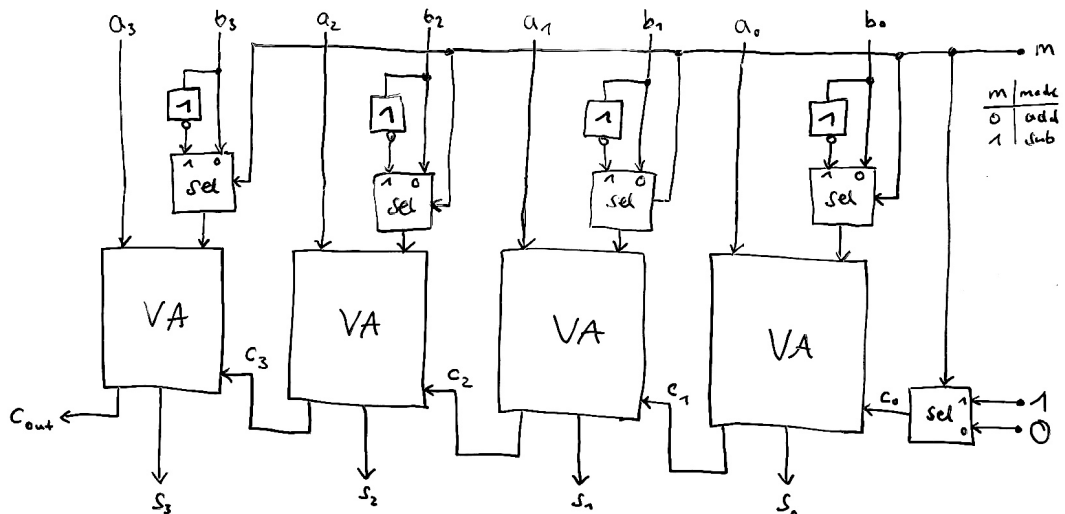
b) Volladdierer:



c) Ripple-Carry-Addierer:



d) Ripple-Carry-Addierer/Subtrahierer:



Noch einfachere Möglichkeit: Alle Multiplexer (mit Invertern) jeweils durch XOR-Gatter ersetzen.

- e) Der Name „Ripple“-Carry suggeriert bereits, dass die Carry-Information wie eine Welle der Reihe nach durch alle Volladdierer hindurchläuft. Dadurch entsteht eine besonders große Schaltungstiefe und folglich eine deutliche Restriktion auf die mögliche Taktfrequenz.

Beim Carry-Select-Addierer werden hingegen für jedes Bit beide Möglichkeiten „durchgespielt“: Einmal mit Carry, einmal ohne. Dahintergeschaltete Multiplexer wählen dann, abhängig des Ergebnisses des vorigen Volladdierers, für das jeweilige Bit den richtigen Wert aus. Zwar muss auch hier darauf gewartet werden, bis sich der Carry durch alle Multiplexer durchpropagiert hat, allerdings geschieht dies deutlich schneller als das Durchlaufen jedes ganzen Volladdierers. Der entscheidende Nachteil: Es werden doppelt so viele Volladdierer benötigt und dazu noch die Multiplexer.