GTI – ÜBUNG 9

CMOS, NAND, PAL, FLIPFLOPS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

CMOS

Die Elektrotechnik lässt grüßen

Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Beschreibung:

Sei die Schaltfunktion $f(x_3, x_2, x_1, x_0) = x_0\,\overline{x_1}\,\overline{x_2} + x_0\,\overline{x_1}\,\overline{x_3}$ gegeben.

Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Begriffsklärung:

CMOS:

- Akronym für Complementary MOS
- Komposition aus komplementären NMOS-Transistor-Feld (Pull-Down) und PMOS-Transistorfeld (Pull-Up)
- Eine Schaltfunktion wird also zwei Mal umgesetzt
- Vorteil: keine Energieverbrauch im festen Schaltzustand, da Stromfluss nur während der Schaltvorgänge

Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Transistor:

- Transfer Resistor (steuerbarer Widerstand)
- Nutzung als steuerbarer Schalter (hier!) bzw. Verstärker
- Elementares Element nahezu jeglicher Digitalschaltung

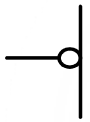
MOS:

- Spezieller Typ eines Transistors
- Elemente:
  - Source: Quelle bzw. Eingang des Transistors
  - Gate: positiv angelegte Spannung erzeugt Stromfluss zwischen Source und Drain
  - Drain: Ziel bzw. Ausgang des Transistors

Aufgabe 1 – CMOS

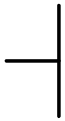
PMOS

- Unterart des MOS-Transistor (bestimmte Dotierung: p-dotiert (Löcher))
- Genutzt im Pull-Up Netz eines CMOS
- Sperrt bei logischer 1 (d.h. 5 V) (deshalb Darstellung mit Inverter)



NMOS

- Unterart des MOS-Transistor (bestimmte Dotierung: n-dotiert)
- Genutzt im Pull-Down Netz eines CMOS
- Öffnet bei logischer 1 (d.h. 5 V)



Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Beispiel: CMOS – Darstellung eines Inverters

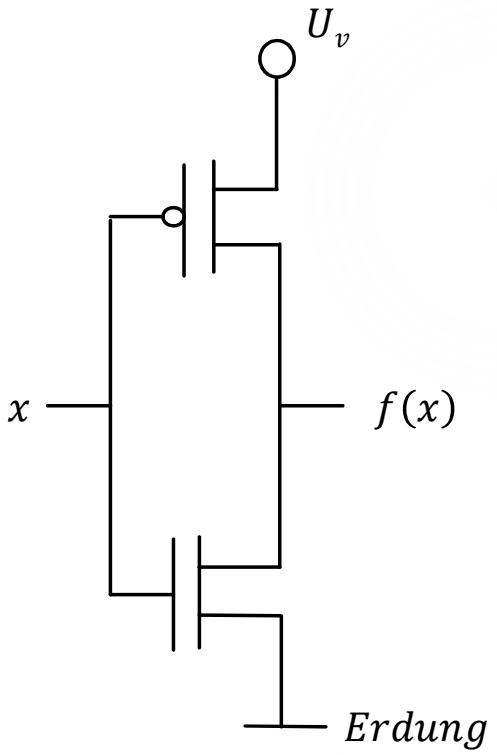
- $f(x) = \bar{x}$

Pull-Up:

- Realisiert normale Schaltfunktion $f(x) = \bar{x}$
- Besteht nur aus negativen Eingängen (PMOS)

Pull-Down:

- Realisiert negierte Schaltfunktion $f(x) = \bar{\bar{x}} = x$
- Besteht nur aus positiven Eingängen (NMOS)
- Falls negatives Literal im Ausdruck; Inverter vorschalten



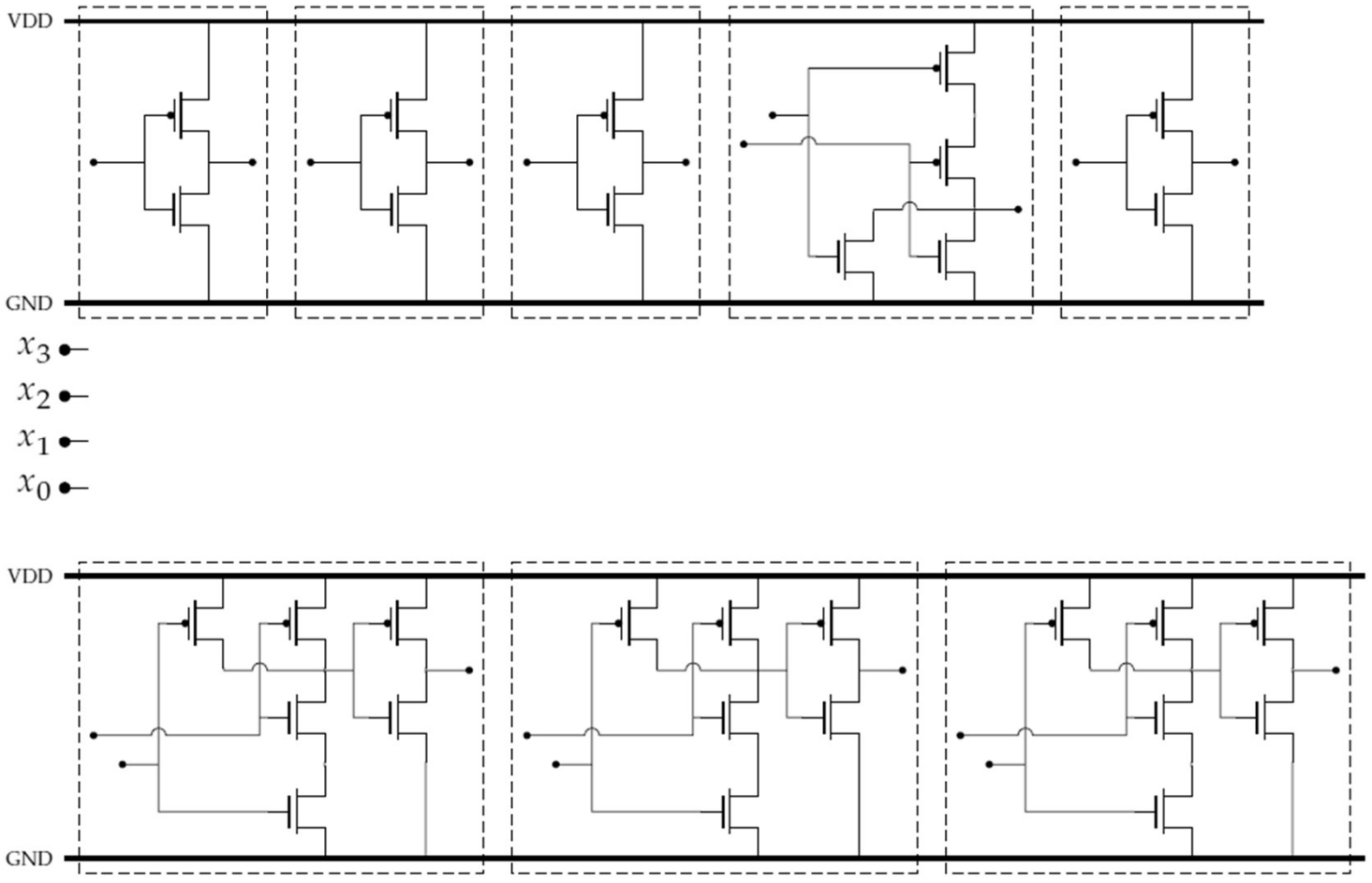
Aufgabe 1 – CMOS

- Standardzellen sind vorgefertigte CMOS-Realisierungen einfacher Schaltfunktionen, wie zum Beispiel Und, Oder oder Nicht, die im Baukastenprinzip zusammengesetzt werden können.
- Schalten Sie die folgenden Standardzellen so zusammen, dass sie f realisieren

Hinweis: die einzelnen Standardzellen sind gestrichelt umrahmt.

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 1 – CMOS



Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 1 – CMOS

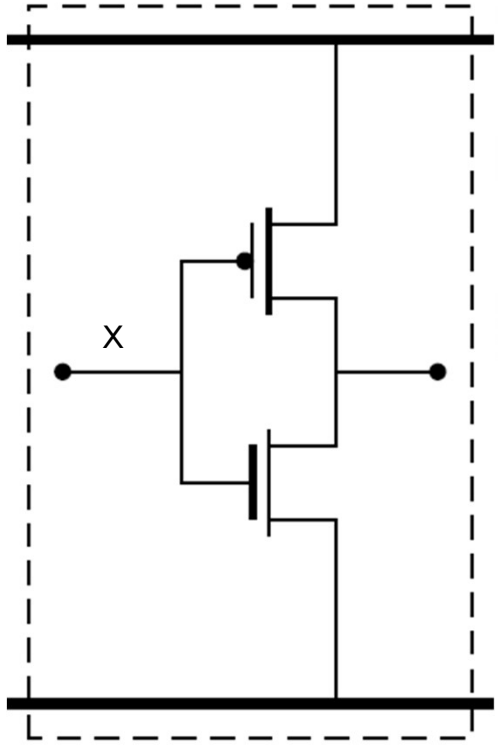
- Gehen wir die einzelnen Gatter durch

Pull-Up: $\bar{x}$

Pull-Down: $x \rightarrow Negation$: $\bar{x}$

Hatte ich ja schon einmal als Beispiel:

Der Inverter!



Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

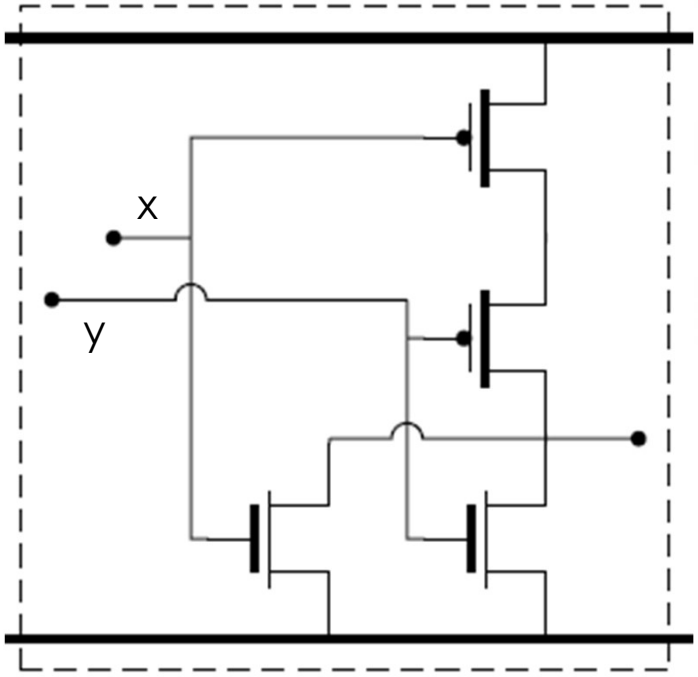
- Gehen wir die einzelnen Gatter durch

Pull-Up: $\bar{x} \cdot \bar{y} = \overline{x + y}$

Pull-Down: $x + y$

$\rightarrow Negation$: $\overline{x + y}$

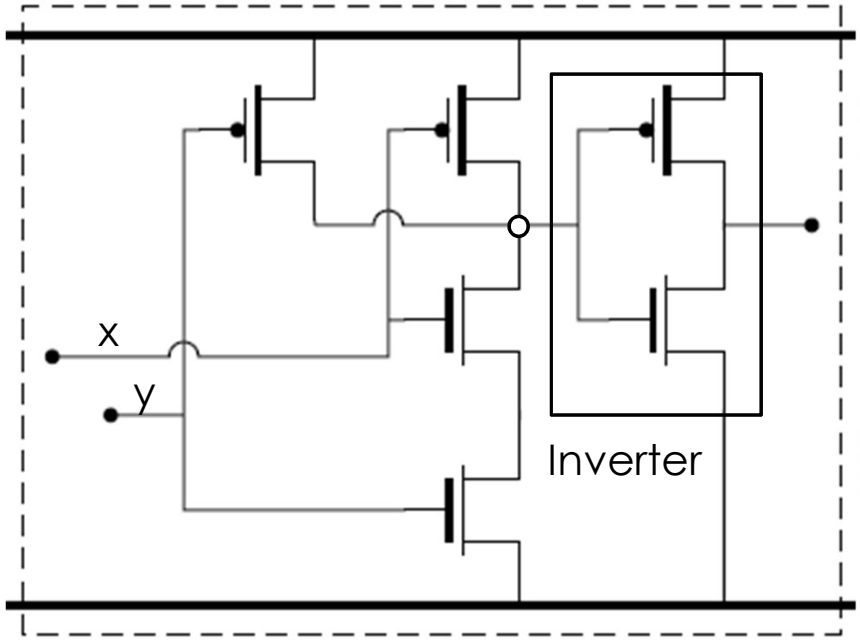
Das NOR-Gatter.



Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Gehen wir die einzelnen Gatter durch



Pull-Up: $\overline{\bar{x} + \bar{y}} = x \cdot y$

Pull-Down: $\overline{x \cdot y}$

$\rightarrow Negation$: $\overline{\overline{x \cdot y}} = x \cdot y$

NAND-Gatter links wird invertiert!

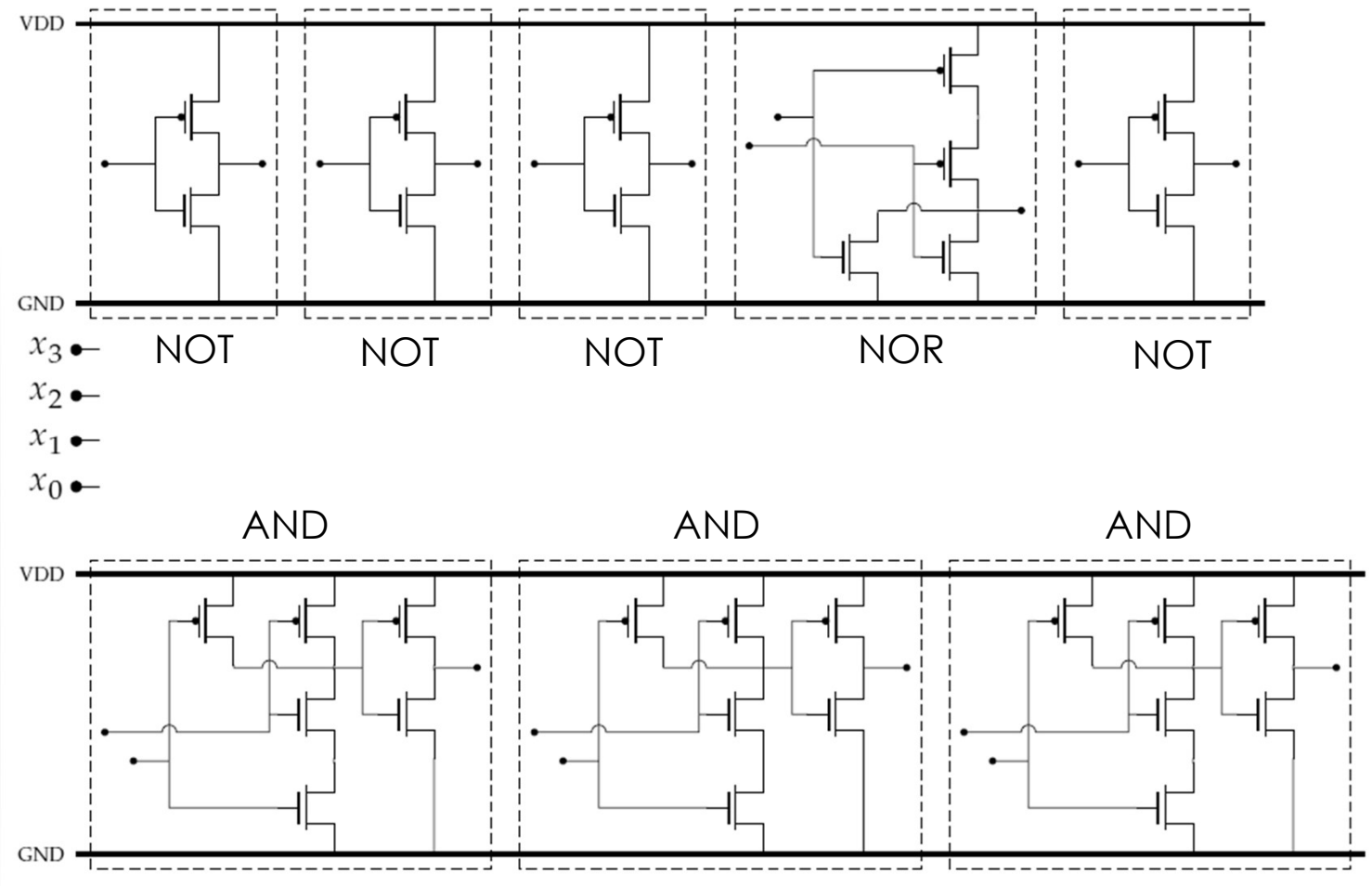
Das AND-Gatter.

Hinweis:

Das Schaltnetz kann auf Grund des Verknüpfungspunkts (o), an dem Pull-Up und Pull-Down zusammengeführt werden, in zwei Bereiche aufgeteilt werden.

Diese kann man am besten separat voneinander auswerten!

Aufgabe 1 – CMOS



Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Umformen des Ausdrucks, so dass unsere Gatter ausreichen

Wir haben: 4 NOT, 1 NOR, 3 ANDs

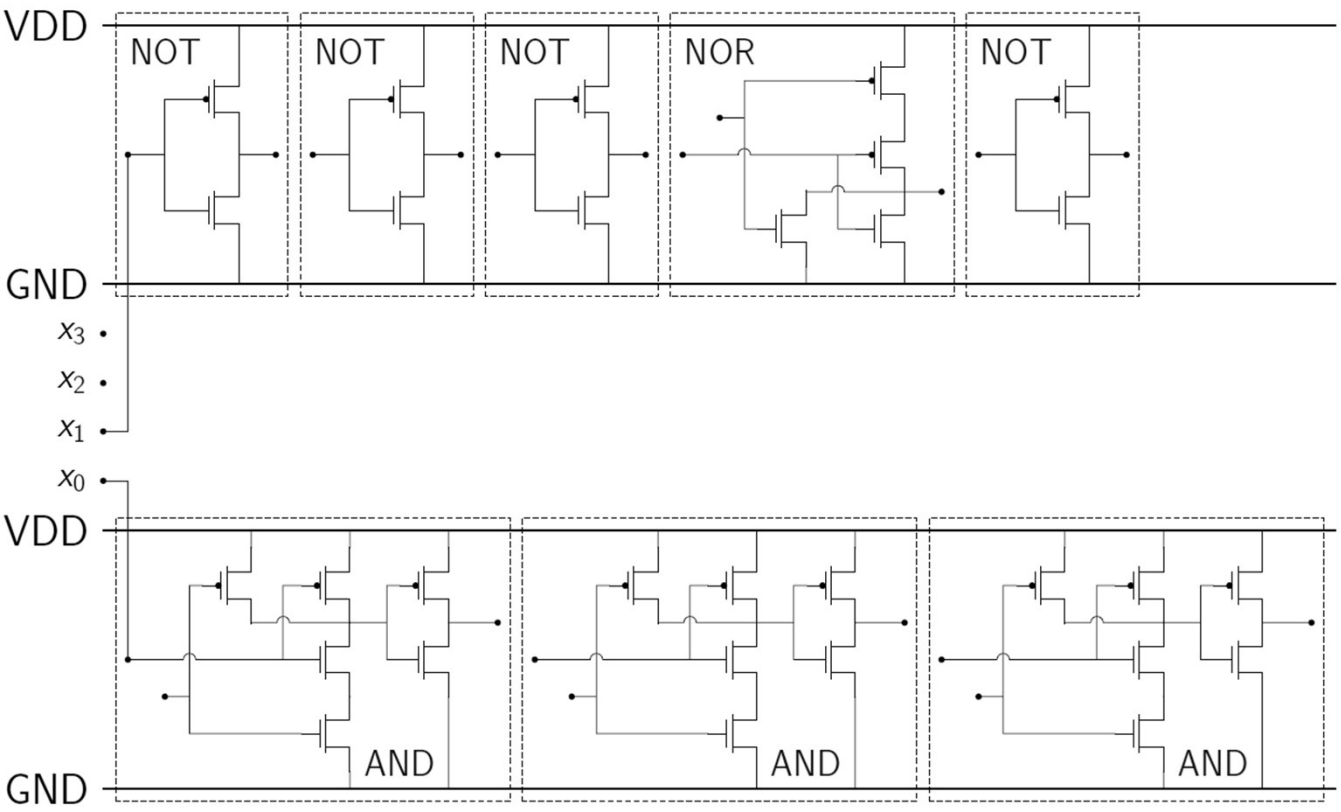
$$
\begin{aligned}
f(x_3, x_2, x_1, x_0) &= x_0\,\overline{x_1}\,\overline{x_2} + x_0\,\overline{x_1}\,\overline{x_3} && \text{| kein ODER (Doppelnegation)} \\
&= \overline{\overline{(x_0\,\overline{x_1}\,\overline{x_2}) + (x_0\,\overline{x_1}\,\overline{x_3})}} && \text{| großes NOR unter Negation} \\
&= \overline{\overline{(A\,\overline{x_2}) + (A\,\overline{x_3})}} && \text{| Wiederverwenden von A} = x_0\,\overline{x_1}
\end{aligned}
$$

Geht genau auf ☺ Was für ein Zufall …

Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

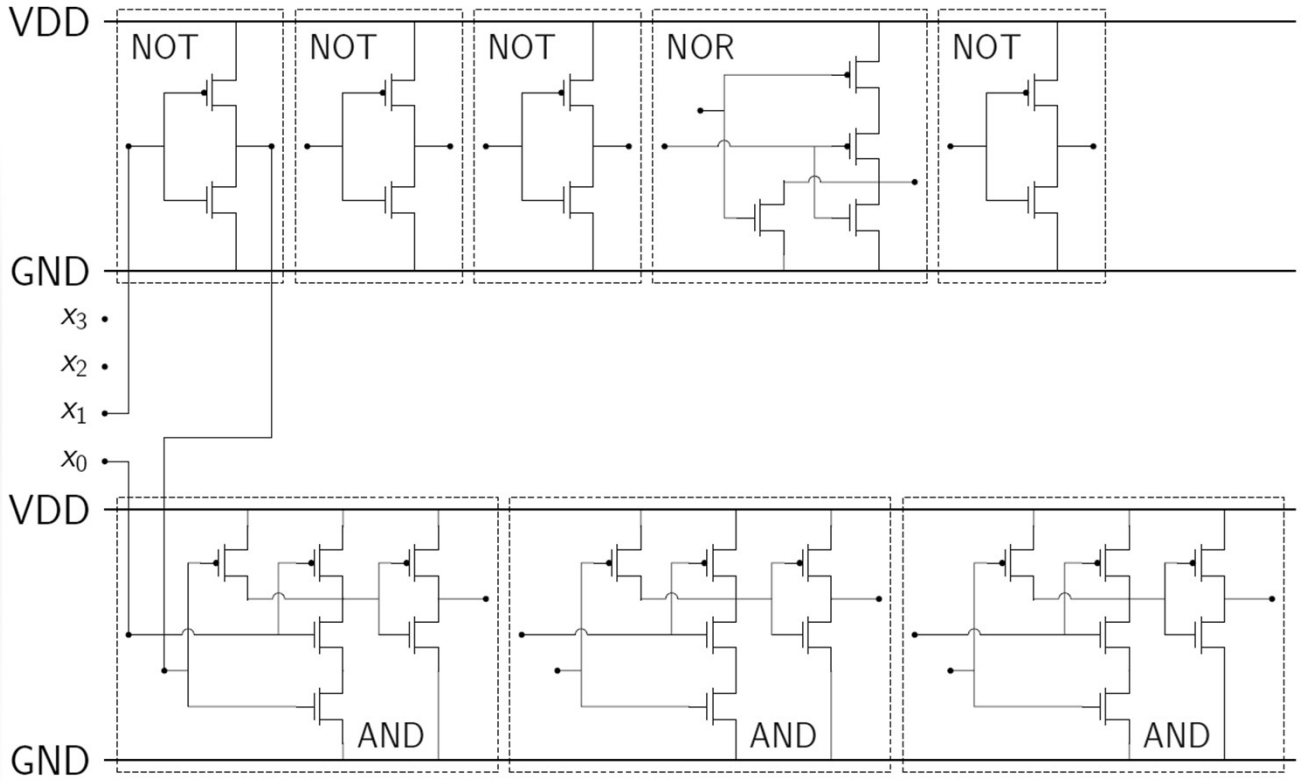
Zuerst das A: $f(x_3, x_2, x_1, x_0) = \overline{\overline{(A\ \overline{x_2}) + (A\ \overline{x_3})}}$ | $A = x_0\ \overline{x_1}$



Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

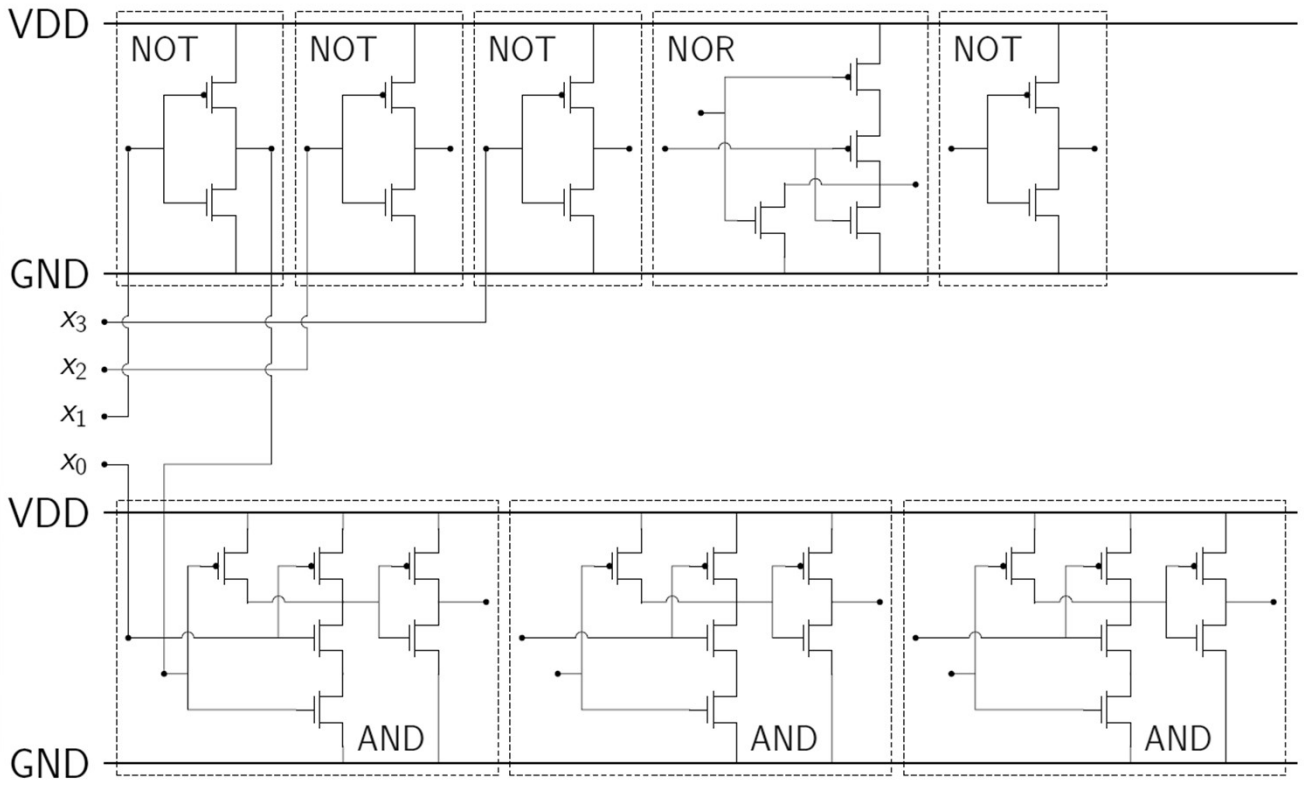
Zuerst das A: $f(x_3, x_2, x_1, x_0) = \overline{\overline{(A\ \overline{x_2}) + (A\ \overline{x_3})}}$ | $A = x_0\ \overline{x_1}$



Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

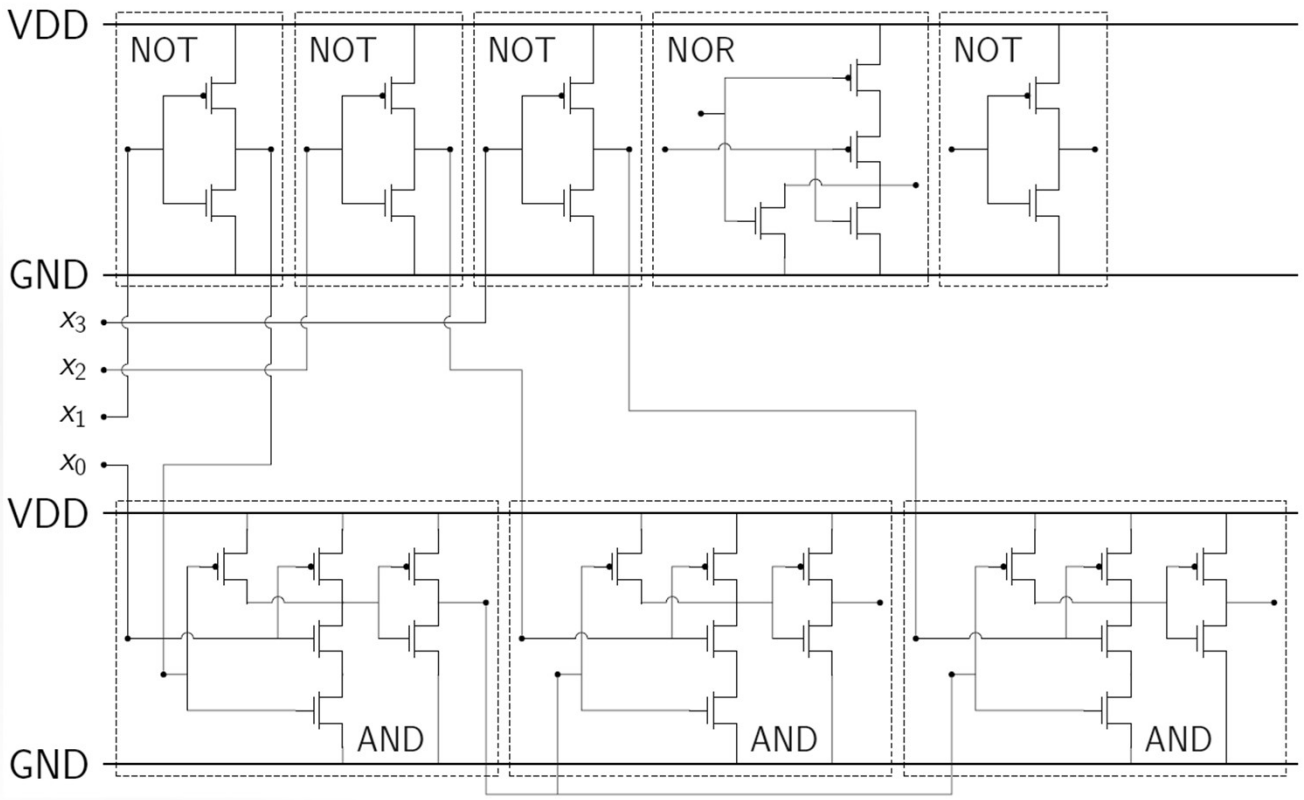
Dann $\overline{x_2}$, $\overline{x_3}$: $f(x_3, x_2, x_1, x_0) = \overline{\overline{(A\ \overline{x_2})} + \overline{(A\ \overline{x_3})}}$ | $A = x_0\ \overline{x_1}$



Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

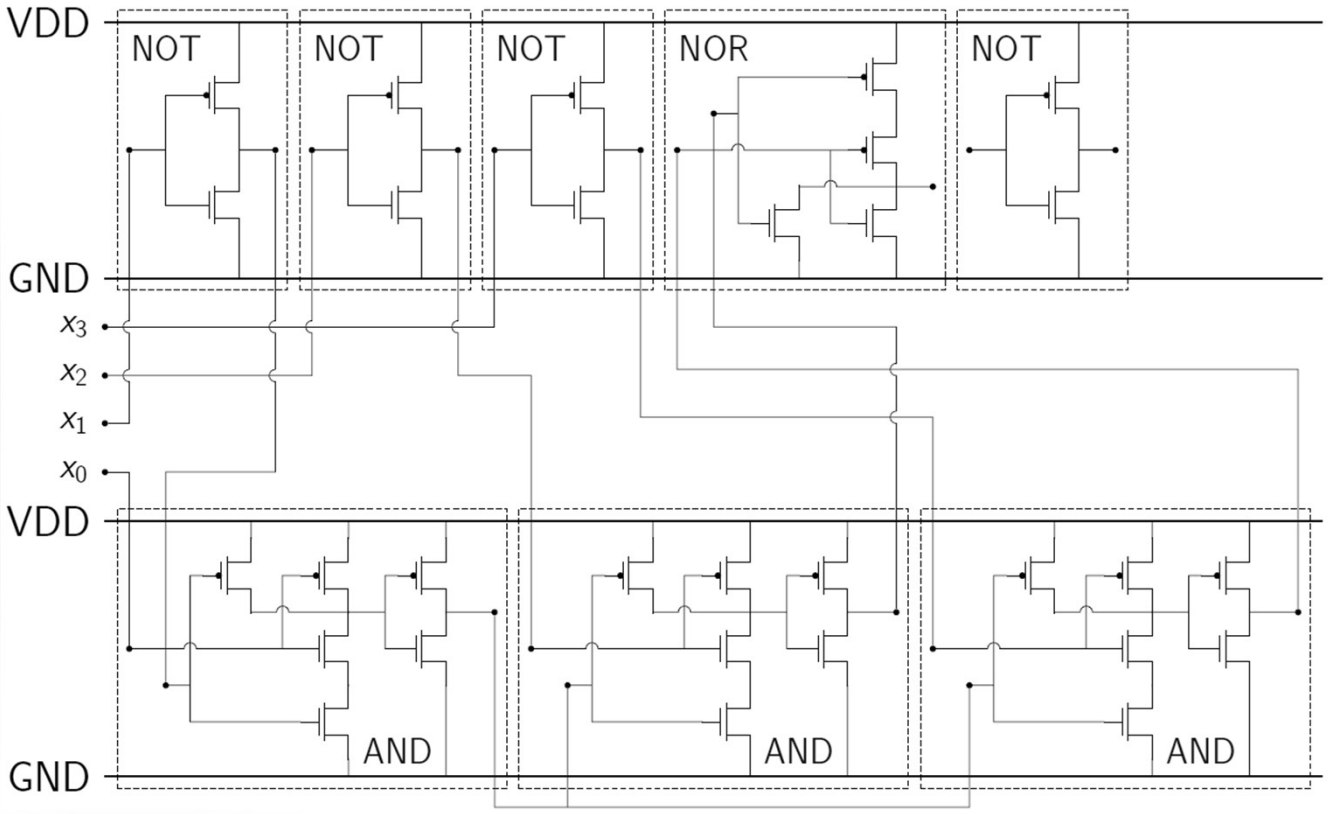
Dann die ANDs: $f(x_3, x_2, x_1, x_0) = \overline{\overline{(A\ \overline{x_2}) + (A\ \overline{x_3})}}$ | $A = x_0\ \overline{x_1}$



Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

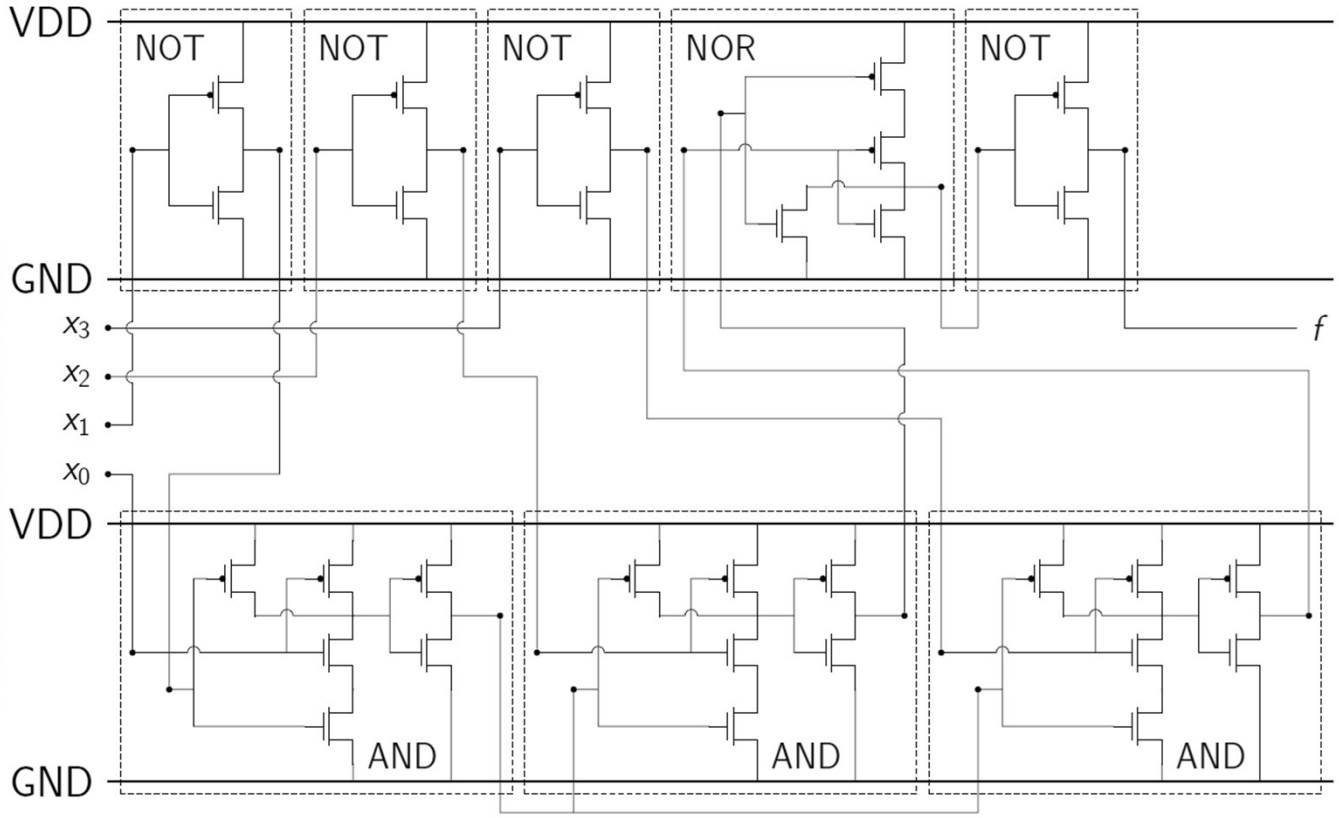
Dann das NOR: $f(x_3, x_2, x_1, x_0) = \overline{\overline{(A\ \overline{x_2}) + (A\ \overline{x_3})}}$ | $A = x_0\ \overline{x_1}$



Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Dann negieren. Fertig: $f(x_3, x_2, x_1, x_0) = \overline{\overline{(A\ \overline{x_2}) + (A\ \overline{x_3})}}$ | $A = x_0\ \overline{x_1}$



Aufgabe 1 – CMOS

- Realisieren Sie die Schaltfunktion f ($x3$, $x2$, $x1$, $x0$) als CMOS-Schaltung mit möglichst wenig Transistoren, wobei alle Eingänge nur in der nicht invertierten Form zur Verfügung stehen.

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 1 – CMOS

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- CMOS-Schaltung mit möglichst wenig Transistoren, wobei alle Eingänge nur in der nicht invertierten Form zur Verfügung stehen.

Minimieren des Ausdrucks:

$$f(x_3, x_2, x_1, x_0) = x_0\,\overline{x_1}\,\overline{x_2} + x_0\,\overline{x_1}\,\overline{x_3} \quad \text{| Distributivgesetz}$$

$$= x_0\,\overline{x_1}(\,\overline{x_2} + \overline{x_3})$$

PMOS-Ausdruck (alle Literale sind negiert):

$$\mathrm{f_{PMOS}} = x_0\,\overline{x_1}(\,\overline{x_2} + \overline{x_3}) \quad (x_0 \text{ wird durch einen Inverter negiert})$$

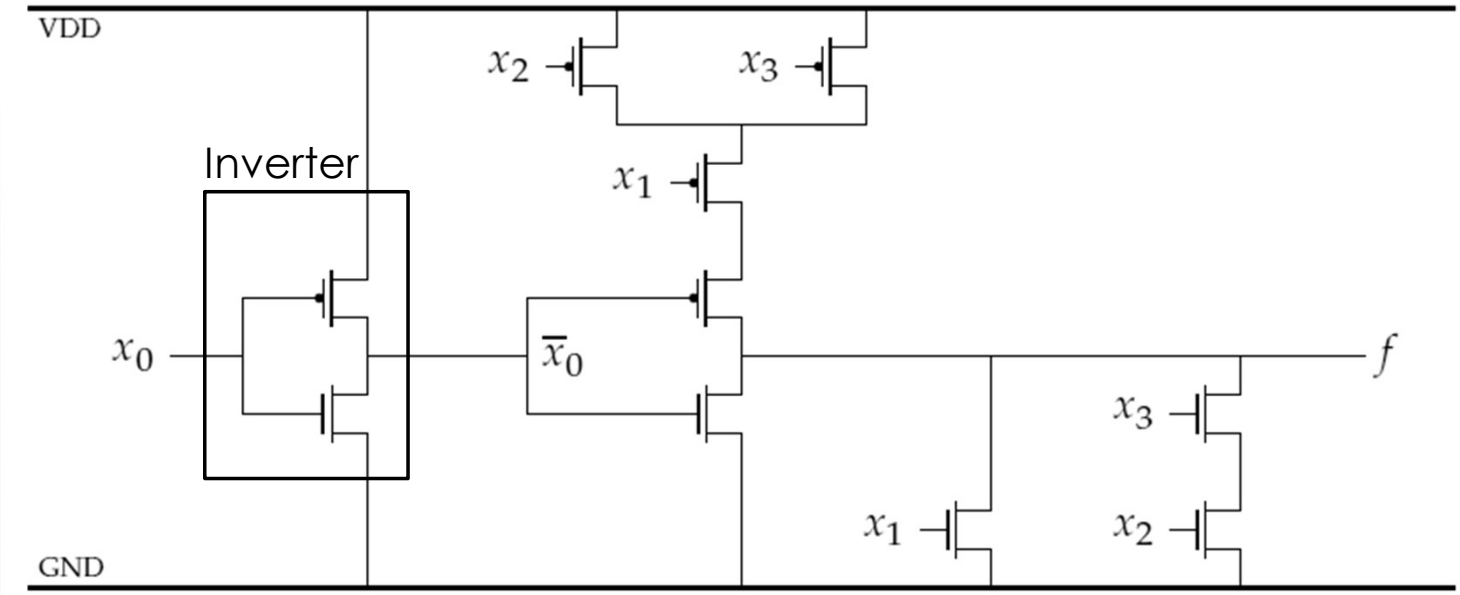
NMOS-Ausdruck (alle Literale sind nicht negiert):

$$\mathrm{f_{NMOS}} = \overline{\mathrm{f_{PMOS}}} = \overline{x_0\,\overline{x_1}(\,\overline{x_2} + \overline{x_3})} \quad \text{|De Morgan}$$

$$= \overline{x_0} + \mathrm{x_1} + \mathrm{x_2 x_3}$$

Aufgabe 1 – CMOS

$f_{PMOS} = x_0\,\overline{x_1}(\overline{x_2} + \overline{x_3})$ $\qquad$ $f_{NMOS} = \overline{x_0} + x_1 + x_2x_3$



Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 1 – CMOS

- Vergleichen Sie die Anzahl benötigter Transistoren in a) und b). Für welche Anwendungsfälle eignen sich die beiden Entwurfsmethoden jeweils?

Hinweis: Gates zählen

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 1 – CMOS

- Vor und Nachteile von a) und b) und Anwendungsfälle

Gatteranzahl:

a) 30 Transistoren b) 10 Transistoren

Sieger b): Implementierung ist schneller und platzsparender (auf Chip).

Automatisierbarkeit:

a) Sehr einfach b) Schwer und komplizierter

Sieger a): Billigere Produktionskosten, da nur automatische „Verkabelung“ nötig. Massenproduktion von Basiselementtafeln möglich.

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

NAND

Der Traum jedes Studenten

Aufgabe 2 – NAND

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Beschreibung:

Mit Hilfe von NAND-Gattern, kann jede beliebige Schaltfunktion realisiert werden (Gleiches gilt für die Verwendung von NOR-Gattern.). Realisieren Sie nun die Schaltfunktion:

$$f = AC\overline{D} + \bar{A}\bar{C} + BC + B\overline{D}$$

Unter ausschließlicher Verwendung von NAND-Gattern, die zwei Eingänge besitzen. Wie viele NAND-Gatter sind erforderlich?

Aufgabe 2 – NAND

- Basissysteme:

Ein Basissystem ist erzeugend (und minimal?), d.h. jede beliebige Schaltfunktion kann durch alleinigen Einsatz der Komponenten des Basissystems realisiert werden.

Korollar: Jedes Basissystem kann in jedes andere Basissystem umgeformt werden.

Bedeutende Basissysteme:

- NAND
- NOR
- AND, OR, Negation (Es reicht auch: AND, Negation bzw. OR, Negation)

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 2 – NAND

- Basissysteme:

Zahl der Operatoren	Namen	Zeichen	Darstellung mit UND, ODER,NICHT	Darstellung von a′	Darstellung von a&b	Darstellung von a+b
3	NICHT UND ODER	y= a′ y=a&b y=a+b	-	a′ - -	- y=a&b -	- - y=a+b
2	NICHT UND	y= a′ y=a&b	-	a′ -	- y=a&b	$a \vee b = \overline{\overline{a \vee b}}$ $= \overline{\overline{a} \wedge \overline{b}}$
2	NICHT ODER	y= a′ y=a+b	-	a′ -	$a \wedge b = \overline{\overline{a \wedge b}}$ $= \overline{\overline{a} \vee \overline{b}}$	- y=a+b
2	UND Antivalenz	y=a&b y=a ≢ b	y=a&b $y = \overline{a} \wedge b \vee a \wedge \overline{b}$	a ≢1	y=a&b -	$a \oplus b \oplus (a \wedge b)$
1	NAND	y=a$\overline{\&}$b	$y = \overline{a \wedge b}$ $= \overline{a} \vee \overline{b}$	a$\overline{\&}$a 1$\overline{\&}$a	$\overline{a \barwedge b} =$ $(a \barwedge b) \barwedge (a \barwedge b)$	$\overline{a} \barwedge \overline{b} =$ $(a \barwedge a) \barwedge (b \barwedge b)$
1	NOR	y=a$\overline{+}$b	$y = \overline{a \vee b}$ $= \overline{a} \wedge \overline{b}$	a$\overline{+}$a 0$\overline{+}$a	$\overline{\overline{a} \vee \overline{b}} = \overline{a \barvee b} =$ $(a \barvee a) \barvee (b \barvee b)$	$\overline{\overline{a \vee b}} = \overline{a \barvee b} =$ $(a \barvee b) \barvee (a \barvee b)$

https://www-user.tu-chemnitz.de/~knmat/V/Alt/Dr%20K%94nig/Digtech3_2.pdf

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 2 – NAND

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Nandisierung/Norisierung:

Um einen beliebigen aussagenlogischen Ausdruck in eine Nand-Form oder Nor-Form mit zwei Eingängen zu bringen wenden wir folgende Gesetze an:

Doppelnegation, De Morgan, 1 bzw. 0 Erweiterung

NAND:

$A \rightarrow \bar{\bar{A}} \rightarrow \overline{\overline{A}1}1$

$AB \rightarrow \overline{\overline{AB}} \rightarrow \overline{\overline{AB}1}$

$A + B \rightarrow \overline{\overline{A + B}} \rightarrow \overline{\bar{A}\bar{B}} \rightarrow \overline{\overline{A}1\,\overline{B}1}$

NOR:

$A \rightarrow \bar{\bar{A}} \rightarrow \overline{\overline{A + 0} + 0}$

$AB \rightarrow \overline{\overline{AB}} \rightarrow \overline{\bar{A} + \bar{B}} \rightarrow \overline{\overline{A + 0} + \overline{B + 0}}$

$A + B \rightarrow \overline{\overline{A + B}} \rightarrow \overline{\overline{A + B} + 0}$

Wir können so jeden Ausdruck analog schrittweise umformen.

Aufgabe 2 – NAND

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Nandisierung/Norisierung:

Falls kein 1 oder 0 Eingang zur Verfügung steht, verdoppelt man einfach den Eingang und hat somit aus einem, zwei Eingänge gemacht!

NAND:

$A \rightarrow \bar{\bar{A}} \rightarrow \overline{\overline{A1}1} \rightarrow \overline{\overline{AA}\,\overline{AA}}$

$AB \rightarrow \overline{\overline{AB}} \rightarrow \overline{\overline{AB}1} \rightarrow \overline{\overline{AB}\,\overline{AB}}$

NOR:

$A \rightarrow \bar{\bar{A}} \rightarrow \overline{\overline{A+0}+0} \rightarrow \overline{\overline{A+A}+\overline{A+A}}$

$AB \rightarrow \overline{\overline{AB}} \rightarrow \overline{\bar{A}+\bar{B}} \rightarrow \overline{\overline{A+A}+\overline{B+B}}$

Sowas kommt manchmal in der Miniklausur vor …

… erhöht den Schreibaufwand aber nochmal deutlich.

Aufgabe 2 – NAND

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

$$f = AC\overline{D} + \bar{A}\bar{C} + BC + B\overline{D}$$

Das Vorgehen: erst gesamt doppelt negieren und dann schrittweise auf die kleineren Terme zurückführen bis alles durch NAND mit zwei Eingängen substituiert ist

$$f = \overline{\overline{AC\overline{D} + \bar{A}\bar{C} + BC + B\overline{D}}} \quad \text{| Doppelnegation (Gesamt)}$$

$$= \overline{\overline{AC\overline{D}} \cdot \overline{\bar{A}\bar{C}} \cdot \overline{BC} \cdot \overline{B\overline{D}}} \quad \text{| De Morgan}$$

$$= \overline{\overline{AC\overline{D}1} \cdot \overline{\overline{A}1\,\overline{C}1} \cdot \overline{BC} \cdot \overline{B\overline{D}1}} \quad \text{| Erweitern mit 1}$$

Wir wollen überall genau zwei Eingänge haben. Um mehrere Eingänge eines NAND auf zwei Eingänge zu reduzieren, doppelnegieren wir partiell und nutzen die Erweiterung mit 1.

Aufgabe 2 – NAND

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

$$f = AC\overline{D} + \bar{A}\bar{C} + BC + B\overline{D}$$

$$\text{f} = \overline{\overline{AC\overline{D1}} \cdot \overline{\overline{A1}\,\overline{C1}} \cdot \overline{BC} \cdot \overline{B\overline{D1}}} \qquad \text{| Partielle Doppelnegation AC}$$

$$= \overline{\overline{\overline{AC}\,\overline{D1}} \cdot \overline{\overline{A1}\,\overline{C1}} \cdot \overline{BC} \cdot \overline{B\overline{D1}}} \qquad \text{| 1 Erweiterung}$$

$$= \overline{\overline{\overline{AC}1\,\overline{D1}} \cdot \overline{\overline{A1}\,\overline{C1}} \cdot \overline{BC} \cdot \overline{B\overline{D1}}}$$

Den gleichen Trick nutzt man für den Gesamtterm. Man zergliedert diesen in zwei Hälften und jene ggf. dann wiederum in zwei Hälften, bis wir bei genau einem oder zwei verknüpften Teiltermen angekommen sind.

$$= \overline{\overline{\overline{\overline{AC}1\,\overline{D1}} \cdot \overline{\overline{A1}\,\overline{C1}}1} \cdot 1\overline{BC} \cdot \overline{B\overline{D1}}}$$

Man benötigt insgesamt also 15 NAND-Gatter mit zwei Eingängen.

Man sieht: Diese Aufgabe macht immer besonders viel Spaß ☺

Aufgabe 2 – NAND

Anmerkung, um diese schöne Aufgabe zu rechtfertigen:

„Bei einer Implementierung von Logikfunktionen in CMOS-Technologie stehen dem Designer oft nur NAND-Gatterzellen mit fester Anzahl der Eingänge zur Verfügung (meistens nur zwei). Die Gründe dafür sind vor allem eine sehr starke Verschlechterung der Gatterlaufzeit in Abhängigkeit von der Anzahl (sog. fan in) und der aktuellen Belegung der Eingänge, die im schlimmsten Fall sogar quadratisch mit der Anzahl der Eingänge ansteigt. Zusätzlich erhöht sich auch der Energieverbrauch bei einer größeren Anzahl der Eingänge beträchtlich (mehr Transistoren, in CMOS immer 2*Anzahl der Eingänge, verbrauchen mehr Energie).“

Na ja ☺ Warum man das per Hand machen muss …

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

PAL

Ein Herz für gelbe Weihnachtskugeln

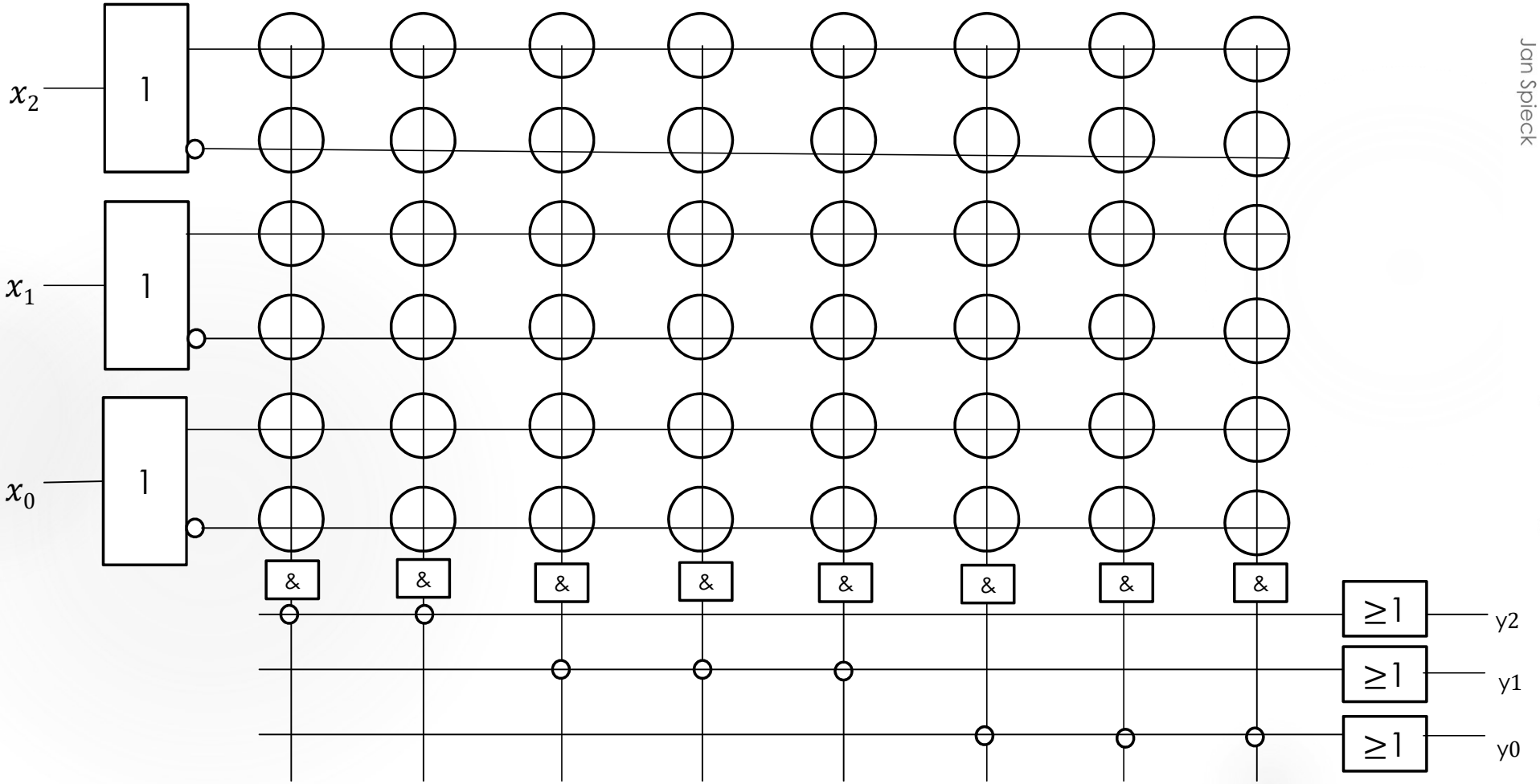
Aufgabe 3 – PAL

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Beschreibung:

Realisieren Sie einen Codeumsetzer, der eine 3 Bit Binärzahl in einen zyklischen Graycode umwandelt. Verwenden Sie dazu die gegebene PAL (Programmable Array Logic) Struktur.

Aufgabe 3 – PAL



Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 3 – PAL

- Begriffsklärung

PAL:

- Programmable Array Logic, d.h. es können konjunktive Terme über ein Feld frei programmiert werden
- Schaltung zur Repräsentation einer DMF
  - Erste Stufe: Auswahl der Literale für konjunktive Terme (Verundung der selektierten Literale) (programmierbar)
  - Zweite Stufe: Auswahl der programmierten konjunktiven Terme (fest)

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

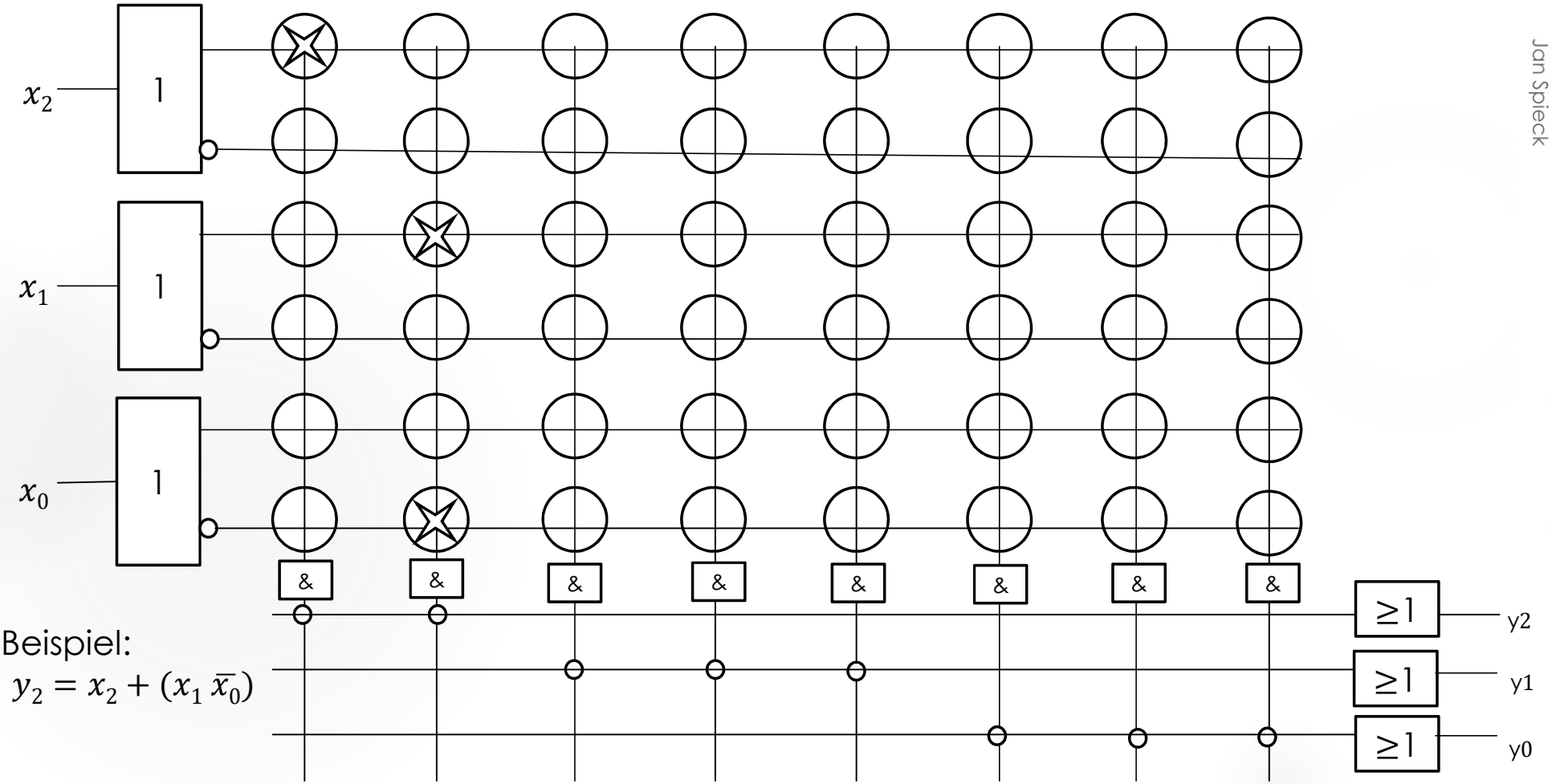
Aufgabe 3 – PAL

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Weitere wichtige Schaltungstypen im Vergleich:

- PAL (Programmable Array Logic):
  - Programmierbare UND-Matrix, feste ODER-Matrix
- PLA (Programmable Logic Array):
  - Programmierbare UND-Matrix, programmierbare ODER-Matrix
- ULA (Universal Logic Array):
  - Feste UND-Matrix, feste ODER-Matrix
  - Auswahl der 2^n Minterme programmierbar
- ROM (Read Only Memory):
  - Feste UND-Matrix, programmierbare ODER-Matrix

Aufgabe 3 – PAL



Beispiel:

$y_2 = x_2 + (x_1\,\overline{x_0})$

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 3 – PAL

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Codeumsetzer: 3 Bit Binärzahl -> zyklischer Graycode

Darstellung der Schaltfunktion in einer Tabelle:

x_2	x_1	X_0	y_2	y_1	y_0
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

Aufgabe 3 – PAL

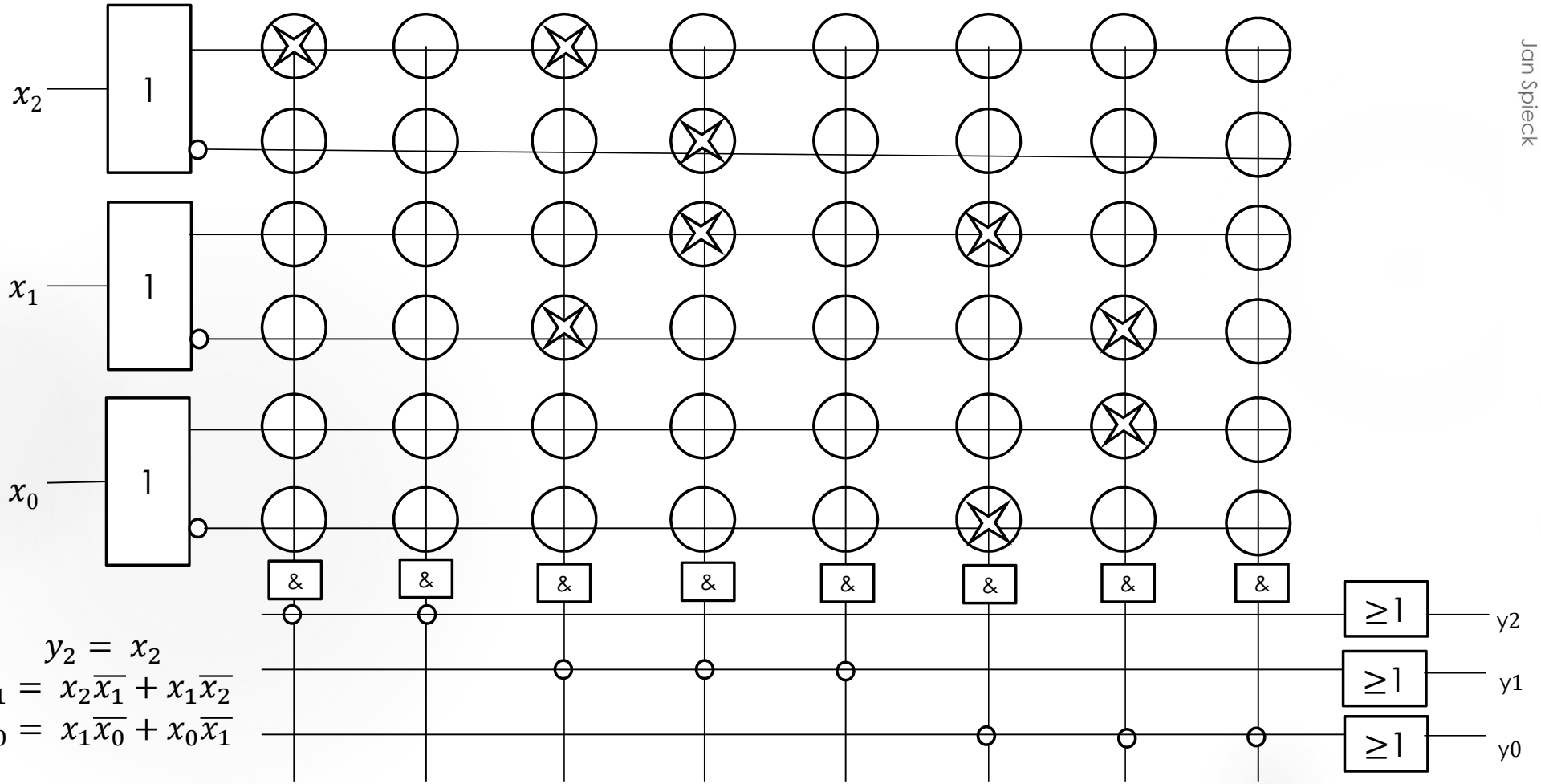
Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Auslesen der Schaltfunktionen (alternativ Schaltfunktionen im Symmetriediagramm)

$$y_2 = x_2$$
$$y_1 = x_2\overline{x_1} + x_1\overline{x_2}$$
$$y_0 = x_1\overline{x_0} + x_0\overline{x_1}$$

x_2	x_1	x_0	y_2	y_1	y_0
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

Aufgabe 3 – PAL



$$y_2 = x_2$$
$$y_1 = x_2\overline{x_1} + x_1\overline{x_2}$$
$$y_0 = x_1\overline{x_0} + x_0\overline{x_1}$$

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Denkpause

Aufgabe: Scherzfragen

1: Wo kommt Silvester vor Weihnachten?

2: Was sagt der Weihnachtsmann, wenn er kopfüber in den Kamin fällt?

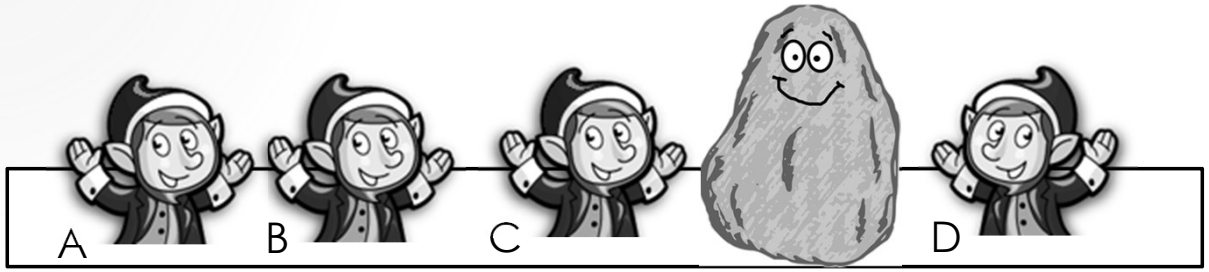
3: Es hängt an der Dachrinne und muss weinen, wenn die Sonne wird wieder scheinen.

Vier Wichtel A, B, C und D, werden eines Tages vom Weihnachtsmann beim Stehlen erwischt, bekommen eine Kappe aufgesetzt und werden in den Schnee eingegraben. Zwischen C und D befindet sich ein Felsen. Beide können also keinen der anderen sehen. B kann nur C sehen. A kann B und C sehen. Keiner kann jedoch seinen eigenen Hut sehen.

Der Weihnachtsmann verkündet: „Kann einer von ihnen seine eigene Kappenfarbe nennen, so kommen alle frei. Schafft es keiner oder rät einer falsch, benutze ich euch als Golfbälle.“

Die Wichtel wissen, dass es genau zwei rote und zwei violette Kappen gibt. Nach fünf Minuten löst einer das Rätsel und sagt seine eigene Kappenfarbe. Wer und wie hat er es gemacht?

.



Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Denkpause

Lösung:

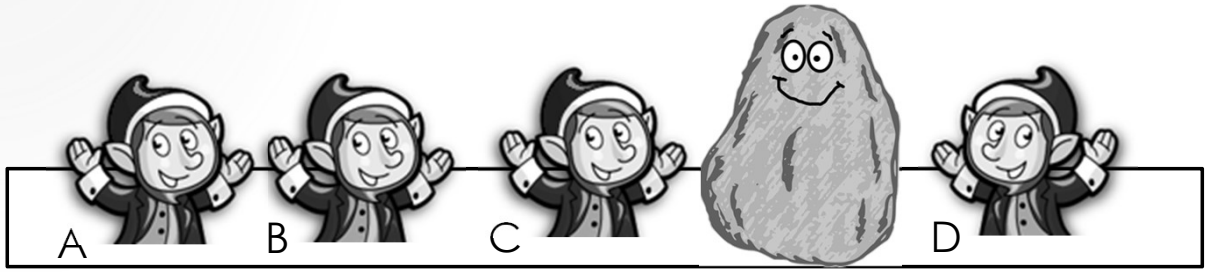
1: Im Wörterbuch

2: Oh Oh Oh

3: Eiszapfen

Wichtel B sagt seine Kappenfarbe.

Er hat wohl folgenden Gedankengang gehabt: „Wenn mein Kappe so wie die meines Vordermannes C violett wäre, dann hätte A sofort gewusst, dass sein eigener Hut rot sein muss. Da A aber schon seit fünf Minuten schweigt muss mein Hut rot sein, denn nur dann kann A nicht wissen, welche Farbe sein Hut hat.“



Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Latches und Flipflops

Endlich mal speichern

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Beschreibung

- In dieser Aufgabe sollen die Eigenschaften ausgesuchter Flipflopschaltungen untersucht werden. Die Verzögerungszeit eines jeden Logikgatters beträgt hierbei $\tau = 1\ \mathrm{ns}$

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Begriffsklärung

Speicherelement:

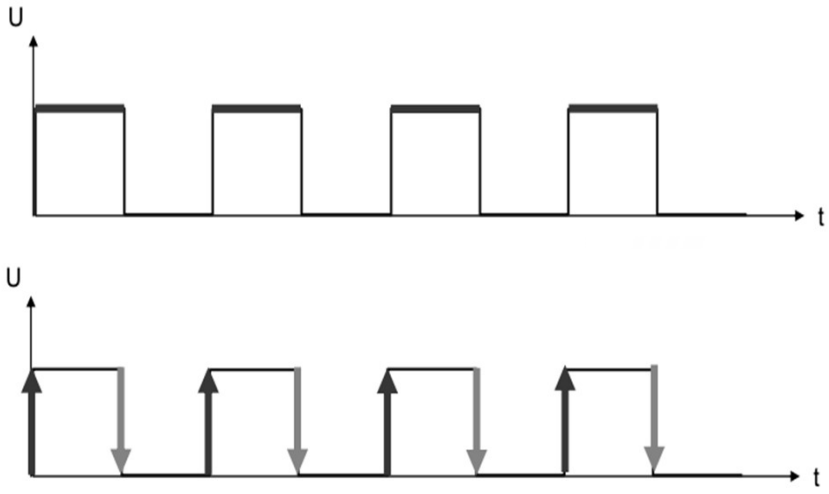
- Gerät/Modul, das einen zuvor angelegten Wert unbegrenzt speichert
- Charakteristische Tabelle:

Befehl	**Q(t): aktueller Zustand**	**Q(t+1): nächster Zustand**
Set (S)	-	1
Reset (R)	-	0
Speichern/ keine Änderung	0	0
	1	1

Aufgabe 4 – Flipflops

Aktivierungsmechanismen:

- Pegelsteuerung
  - Pegel auf High (aktiviert)
  - Pegel auf Low (deaktiviert)
- Flankensteuerung
  - Positive/steigende Flanke (aktiviert)
  - Negative/fallende Flanke (aktiviert)



Änderungszeitpunkte:

- Synchron: zu festen durch den Takt (Clock) vorgegebenen Zeitpunkte
  (1) während des Pegels oder (2) während einer Flanke
- Asynchron: zu beliebigen Zeitpunkten

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 4 – Flipflops

Flipflop:

- Speicherelement zur Speicherung eines Bits
- Verschiedene Arten der Umsetzung (RS, JK, Master-Slave, D, …)
- Zwei konträre gängige Definitionen:
  - Rein flankengesteuerte Speicherelemente
  - Überbegriff für jegliche Speicherelemente

Latch:

- Pegelgesteuertes Flipflop
- Deutsche Bezeichnung: transparentes Flipflop

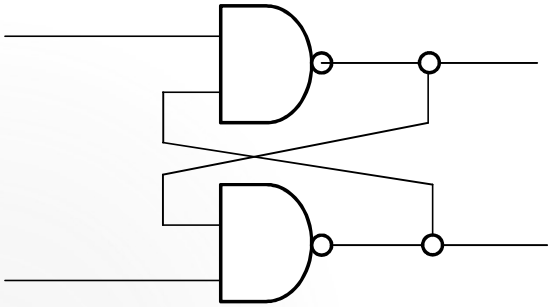
Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 4 – Flipflops

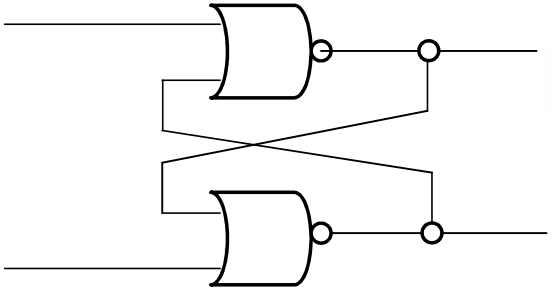
Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- An die Eingänge (A, B) der Schaltungen (a.1) und (a.2) werden nacheinander die Werte aus der Tabelle angelegt. Vervollständigen Sie die Tabelle und geben Sie den Signalen A bis D jeweils sinnvolle Bezeichnungen.

a.1



a.2



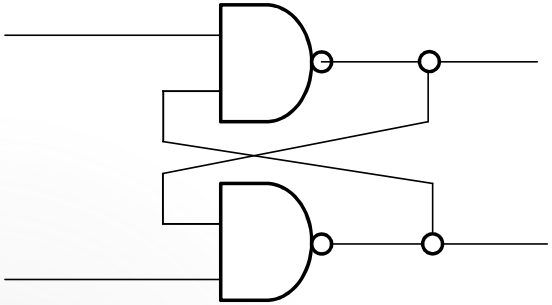
A	B	C1	D1	C2	D2
0	1				
0	0				
1	1				
1	0				
1	1				
0	0				

Aufgabe 4 – Flipflops

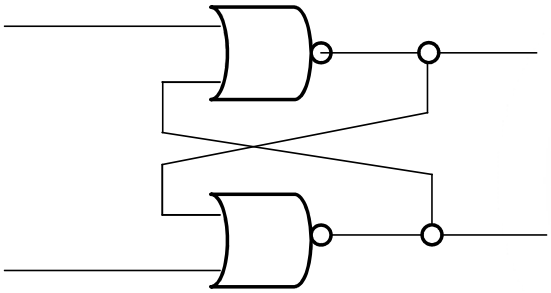
Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Wir betrachten die Schalttabellen von NAND und NOR-Gattern

a.1



a.2



NOR:

Ausgang 0, sobald mind. eine 1

NAND:

Ausgang 1, sobald mind. eine 0

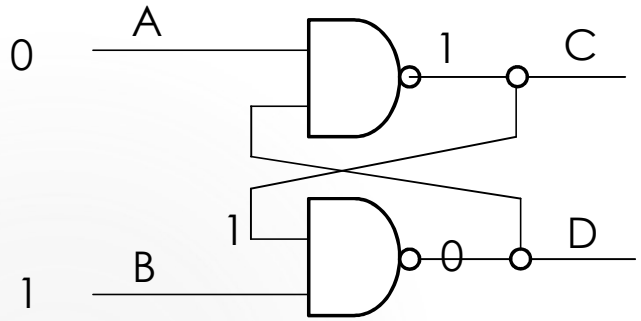
A	B	NOR	NAND
0	0	1	1
0	1	0	1
1	0	0	1
1	1	0	0

Dies gilt es zu beachten, da die Rückkoppelung bei diesen Fällen keine Änderung mehr hervorruft – wir also in einem stabilen Zustand landen (solange A und B konstant).

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Beginnen wir mit dem Flipflop a.1. Um die Einträge der Tabelle zu bekommen, legen wir an die Eingänge A und B jeweils die geforderten Werte (0 bzw. 1) an und geben den stabilen Endzustand an (falls dieser existiert)



A	B	C	D
0	1	1	0
0	0		
1	1		
1	0		
1	1		
0	0		

1: Eine 0 am Eingang eines NAND impliziert eine 1 am Ausgang.

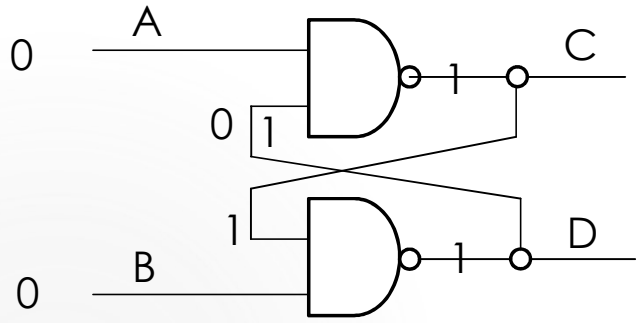
2: Zwei Einsen am Eingang eines NAND implizieren eine 0 am Ausgang

Dadurch ändert sich der Wert von C nicht (stabiler Endzustand)

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Ebenso verfahren wir für die anderen Kombinationen an den Eingängen A und B und beachten dabei die vorherige Belegung



A	B	C	D
0	1	1	0
0	0	1	1
1	1		
1	0		
1	1		
0	0		

1: Durch die vorherige Belegung liegen 0 und 1 an je einem NAND-Eingang

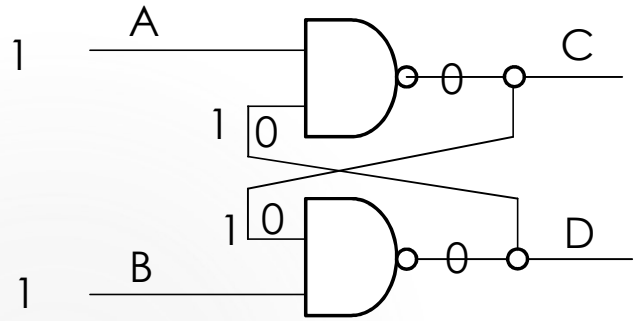
2: Durch die Nullen am zweiten Eingang ergeben sich an den Ausgängen zwei Einsen

Die neue 1 am oberen NAND-Eingang ändert nichts am Ausgang (stabiler Zustand). Dieser Zustand ist unzulässig; dies werden wir bei späteren FFs vermeiden.

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Ebenso verfahren wir für die anderen Kombinationen an den Eingängen A und B und beachten dabei die vorherige Belegung



A	B	C	D
0	1	1	0
0	0	1	1
1	1	?	?
1	0		
1	1		
0	0		

1: Durch die vorherige Belegung liegt jeweils eine 1 an einem NAND-Eingang

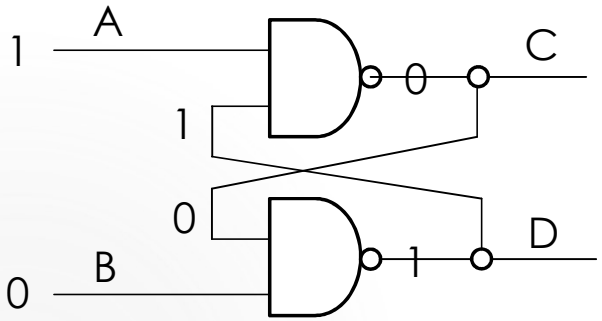
2: Somit ergeben sich am Ausgang zwei Nullen

3: Diese Nullen implizieren am Ausgang wieder zwei Einsen. Gehe zu Schritt 2.

Wir bekommen also eine Oszillation zwischen 0 und 1 am Ausgang (instabil!)

Aufgabe 4 – Flipflops

- Ebenso verfahren wir für die anderen Kombinationen an den Eingängen A und B und beachten dabei die vorherige Belegung



A	B	C	D
0	1	1	0
0	0	1	1
1	1	?	?
1	0	0	1
1	1		
0	0		

1: Die untere 0 impliziert eine 1 am Ausgang

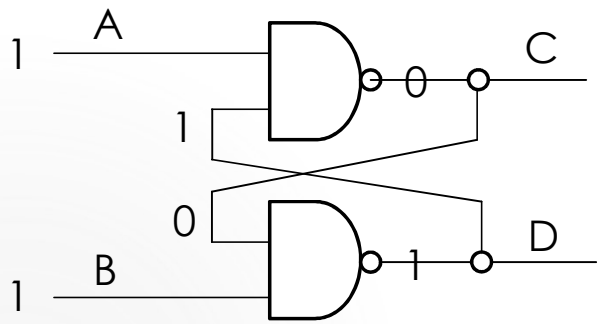
2: Jene 1 impliziert am oberen NAND eine 0 am Ausgang

Dieser Zustand ist stabil, da sich durch die neue 0 nichts mehr ändert.

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 4 – Flipflops

- Ebenso verfahren wir für die anderen Kombinationen an den Eingängen A und B und beachten dabei die vorherige Belegung



A	B	C	D
0	1	1	0
0	0	1	1
1	1	?	?
1	0	0	1
1	1	0	1
0	0		

1: Aus der vorherigen Belegung resultierte eine 0 und eine 1

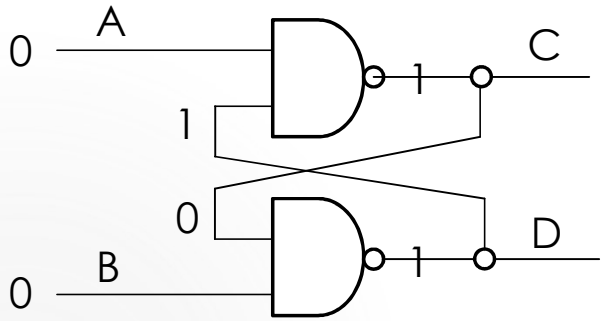
2: Somit ändert sich am Ausgang nichts mehr -> keine Änderung

Dieser Zustand ist stabil.

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 4 – Flipflops

- Ebenso verfahren wir für die anderen Kombinationen an den Eingängen A und B und beachten dabei die vorherige Belegung



A	B	C	D
0	1	1	0
0	0	1	1
1	1	?	?
1	0	0	1
1	1	0	1
0	0	1	1

1: Aus der vorherigen Belegung resultierte eine 0 und eine 1

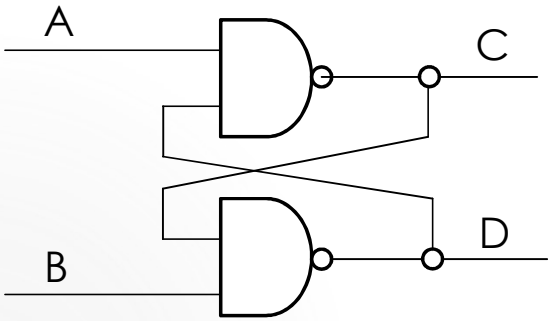
2: Beide Nullen implizieren eine 1 am Ausgang.

Dieser Zustand ist stabil.

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 4 – Flipflops

- Betrachten wir nun die Tabelle zur Benennung der Ein- und Ausgänge



A	B	C	D	Entspricht
0	1	1	0	Reset/Set
0	0	1	1	Ungültig!
1	1	?	?	Oszillation
1	0	0	1	Set/Reset
1	1	0	1	Keine Änderung
0	0	1	1	Ungültig!

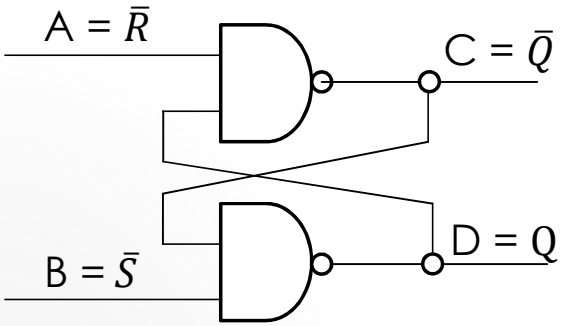
Befehl	Q(t): aktueller Zustand	Q(t+1): nächster Zustand
Set (S)	-	1
Reset (R)	-	0
Speichern/	0	0
keine Änderung	1	1

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Betrachten wir nun die Tabelle zur Benennung der Ein- und Ausgänge



A	B	C	D	Entspricht
0	1	1	0	Reset/Set
0	0	1	1	Ungültig!
1	1	?	?	Oszillation
1	0	0	1	Set/Reset
1	1	0	1	Keine Änderung
0	0	1	1	Ungültig!

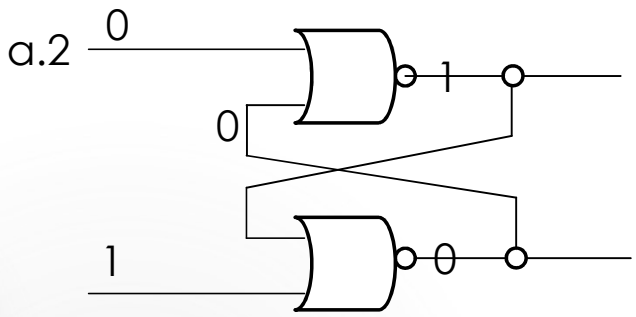
Wir haben ein asynchrones Flipflop (keine Clock) mit einem ungültigen Zustand. Dies ist also eine R/S-Flipflop. Laut Definition erreicht man einen ungültigen Zustand durch das Eingangstupel (1,1) und nicht (0,0), weshalb wir unsere Eingänge als negiert annehmen. Wir definieren uns A als negierten Reset-Eingang und B als negierten Set-Eingang (auch andersherum möglich). Damit ist C = $\bar{Q}$ und D = Q.

Diese Art von Flipflop nennt man Active-low. (da negierter Eingang)

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Nun betrachten wir den zweiten Flipflop a.2:



A	B	C	D
0	1	1	0
0	0		
1	1		
1	0		
1	1		
0	0		

1: Aus der unteren 1 resultiert eine 0 am Ausgang

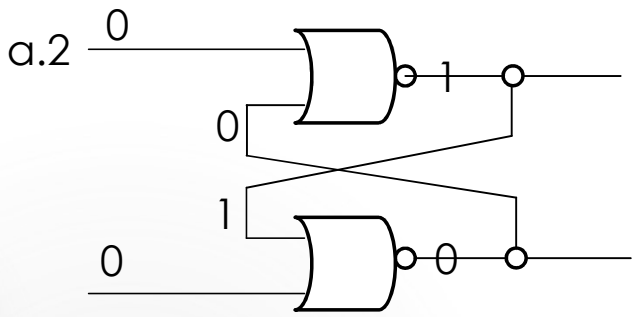
2: Beide Nullen implizieren eine 1 am oberen Ausgang.

Dieser Zustand ist stabil!

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Nun betrachten wir den zweiten Flipflop a.2:



A	B	C	D
0	1	1	0
0	0	1	0
1	1		
1	0		
1	1		
0	0		

1: Aus der vorherigen Belegung ergibt sich eine alte 0 und 1

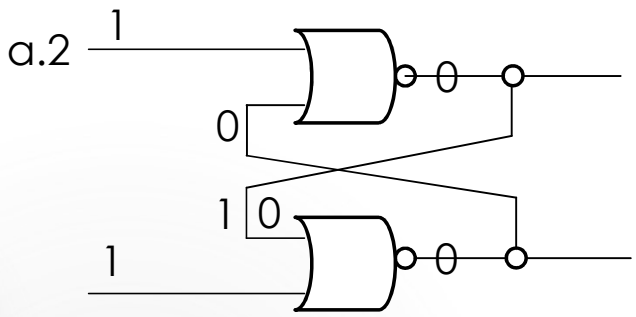
2: Durch die angelegten Nullen ändert sich nichts an der Ausgabe

Dieser Zustand ist stabil!

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Nun betrachten wir den zweiten Flipflop a.2:



A	B	C	D
0	1	1	0
0	0	1	0
1	1	0	0
1	0		
1	1		
0	0		

1: Aus der vorherigen Belegung ergibt sich eine alte 0 und 1

2: Durch die angelegten Einsen ergeben sich oben und unten zwei Nullen

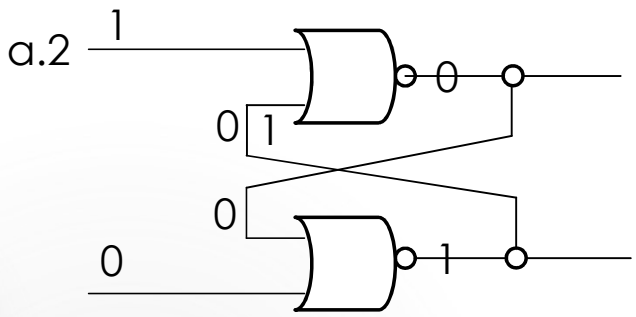
3: Die neue 0 am unteren NOR verändert die Ausgabe nicht

Dieser Zustand ist stabil!

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Nun betrachten wir den zweiten Flipflop a.2:



A	B	C	D
0	1	1	0
0	0	1	0
1	1	0	0
1	0	0	1
1	1		
0	0		

1: Aus der vorherigen Belegung ergibt sich zwei alte Nullen

2: Am unteren NOR ergibt sich am Ausgang so eine 1

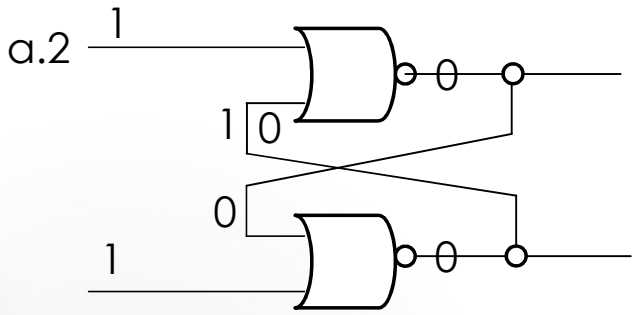
3: Die neue 1 am oberen NOR verändert die Ausgabe nicht

Dieser Zustand ist stabil!

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Nun betrachten wir den zweiten Flipflop a.2:



A	B	C	D
0	1	1	0
0	0	1	0
1	1	0	0
1	0	0	1
1	1	0	0
0	0		

1: Aus der vorherigen Belegung ergibt sich eine alte 1 und eine 0

2: Die beiden neuen Einsen ergeben am Ausgang zwei Nullen

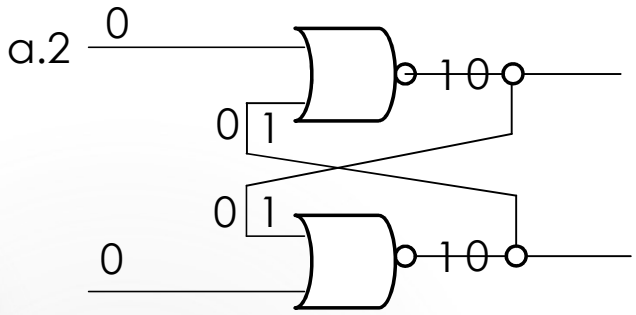
3: Die neue 0 am oberen NOR verändert die Ausgabe nicht

Dieser Zustand ist stabil!

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Nun betrachten wir den zweiten Flipflop a.2:



A	B	C	D
0	1	1	0
0	0	1	0
1	1	0	0
1	0	0	1
1	1	0	0
0	0	?	?

1: Aus der vorherigen Belegung ergeben sich zwei alte Nullen

2: Somit ergibt sich oben wie unten eine 1 am Ausgang

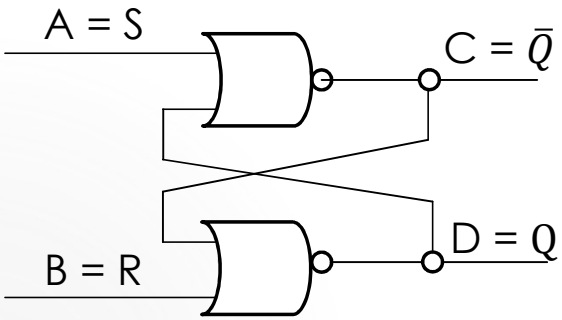
3: Die neuen Einsen ergeben wiederum Nullen am Ausgang (Wdh. Schritt 2)

Dieser Zustand ist instabil! Es kommt zur Oszillation

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Betrachten wir nun die Tabelle zur Benennung der Ein- und Ausgänge



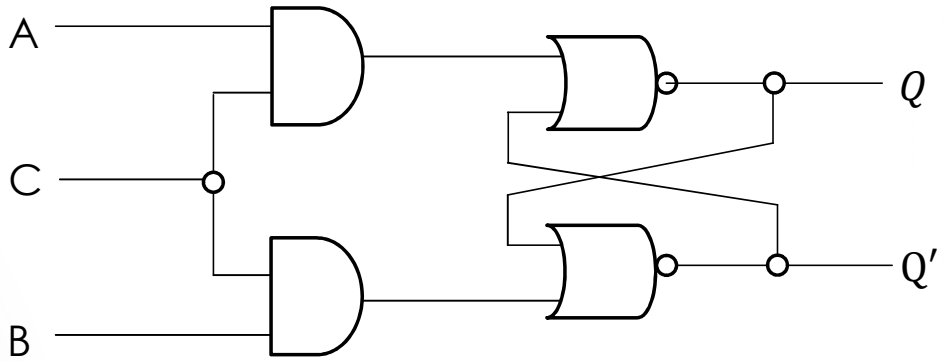
A	B	C	D	Entspricht
0	1	1	0	Reset/Set
0	0	1	0	Keine Änderung
1	1	0	0	Ungültig!
1	0	0	1	Set/Reset
1	1	0	0	Ungültig!
0	0	?	?	Oszillation

Wir haben ein asynchrones Flipflop (keine Clock) mit einem ungültigen Zustand. Dies ist also eine R/S-Flipflop. Laut Definition erreicht man einen ungültigen Zustand durch das Eingangstupel (1,1), weshalb wir unsere Eingänge als nicht negiert annehmen. Wir definieren uns A als Set-Eingang und B als Reset-Eingang (auch andersherum möglich). Damit ist C = $\bar{Q}$ und D = Q

Diese Art von Flipflop nennt man Active-high. (da nicht-negierter Eingang)

Aufgabe 4 – Flipflops

- Wie bezeichnet man folgendes Flipflop?



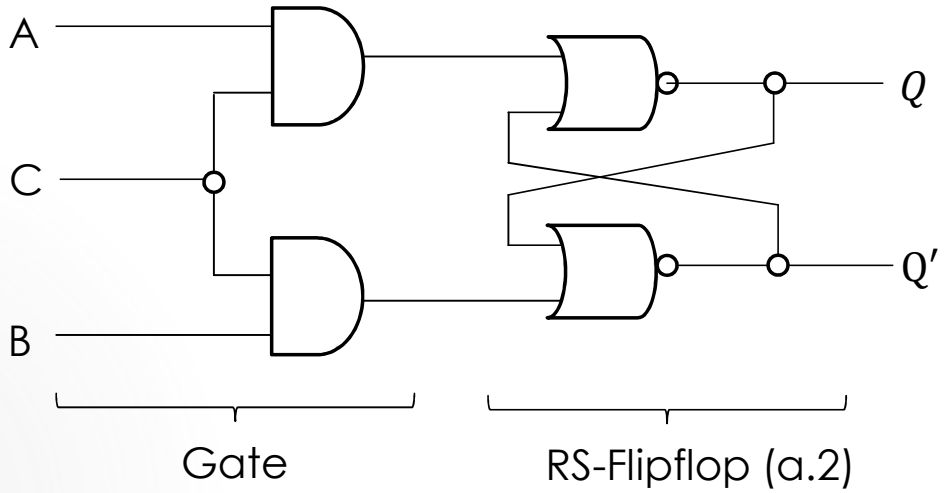
Hinweis: C steht für Clock

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Wie bezeichnet man folgendes Flipflop?



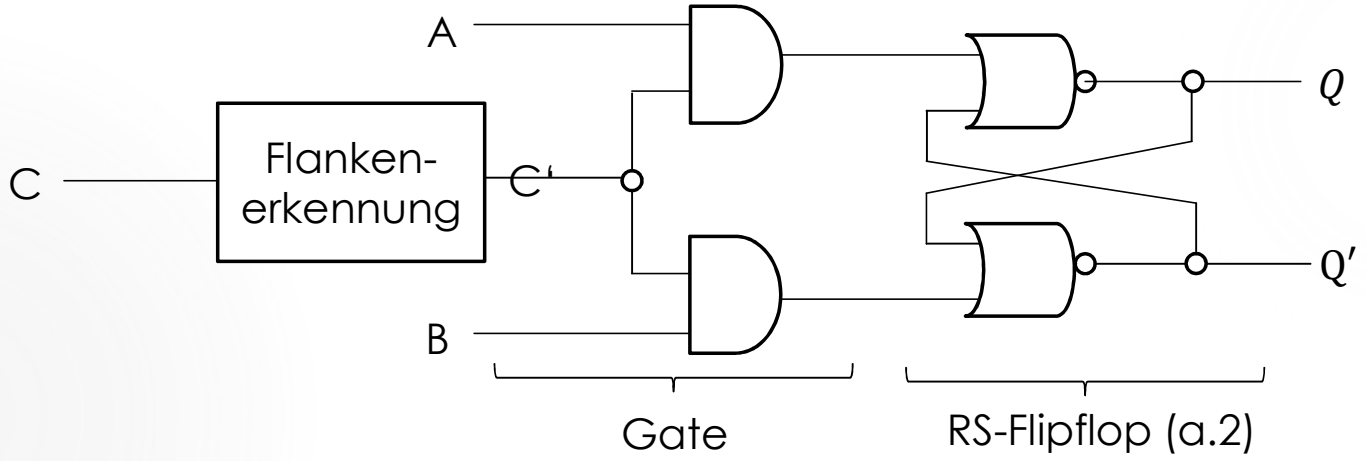
Solange C = 0 ist das Gate deaktiviert, da an den Ausgängen immer (0,0) anliegt.

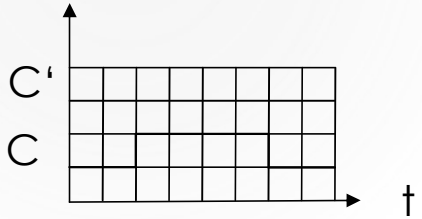
Bei C = 1, kann man sich das UND-Gatter wegdenken und es werden A und B weitergeleitet (aktiviert). Das Flipflop ist also ein pegelgesteuertes RS-Flipflop.

Außerdem ist es wieder Active-high.

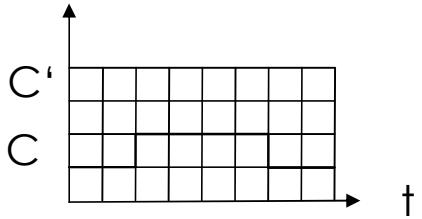
Aufgabe 4 – Flipflops

- Das Flipflop soll so erweitert werden, dass es bei der steigenden (b.1), der fallenden (b.2) und bei jeder Flanke (b.3) die Daten übernimmt. Geben Sie jeweils das Schaltnetz und das zugehörige Zeitverhalten der Flankenerkennung an.

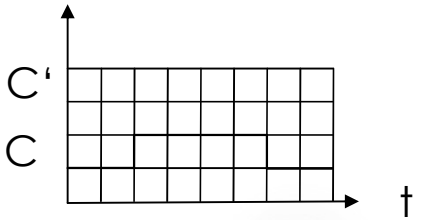




b.1



b.2



b.3

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

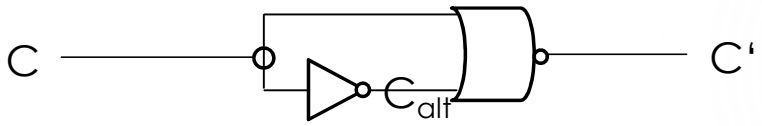
Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

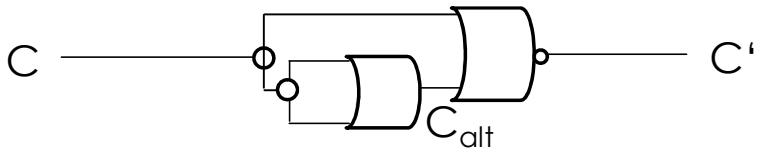
- Verzögerungsstrategien

Es gibt verschiedene Verzögerungsarten, um ein C_{alt} zu bekommen.

- Inverter:
  - geringe Verzögerung
  - Man erhält ein negiertes C_{alt}



- UND/ODER:
  - mittlere Verzögerung
  - Man erhält ein nicht neg. C_{alt}



- Sonstige Gatter:
  - NAND/NOR genutzt, da geringere Verzögerung als AND/ODER (Latenz)
  - Andere Gatter werden auch hin und wieder verwendet

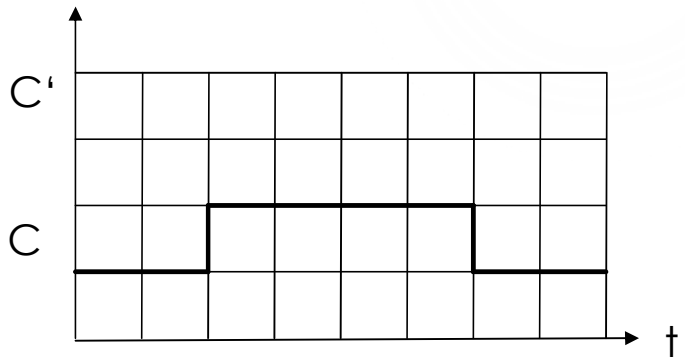
Aufgabe 4 – Flipflops

- Das Flipflop soll so erweitert werden, dass es bei der steigenden Flanke die Daten übernimmt. Geben Sie jeweils das Schaltnetz und das zugehörige Zeitverhalten der Flankenerkennung an.

Wir wissen, dass die Verzögerungszeit $\tau = 1$ beträgt.

Wir wollen folgendes Verhalten (pro t = 1) umsetzen:

C_{alt}	C_{neu}	C‘
0	0	0
0	1	1 (steigend)
1	0	0 (fallend)
1	1	0



b.1

$$f(C_{alt}, C_{neu}) = \overline{C_{alt}} \cdot C_{neu}$$

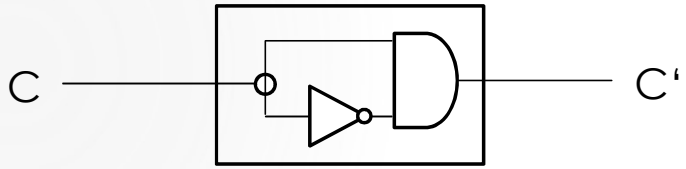
Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 4 – Flipflops

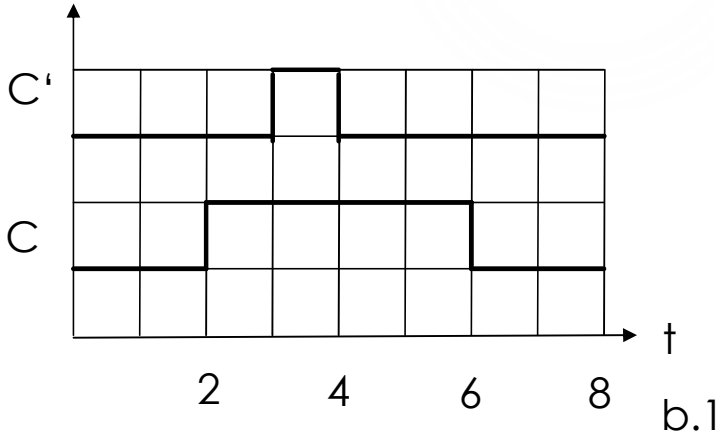
- Das Flipflop soll so erweitert werden, dass es bei der steigenden Flanke die Daten übernimmt. Geben Sie jeweils das Schaltnetz und das zugehörige Zeitverhalten der Flankenerkennung an.

Wir wissen, dass die Verzögerungszeit $\tau = 1$ beträgt. Dadurch entsteht ein C_{alt} nach dem Inverter, da die obere Leitung keine Verzögerung besitzt.

$$f(C_{alt}, C_{neu}) = \overline{C_{alt}} \cdot C_{neu}$$



Das UND-Gatter verzögert das C‘ um eine Zeiteinheit.



b.1

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

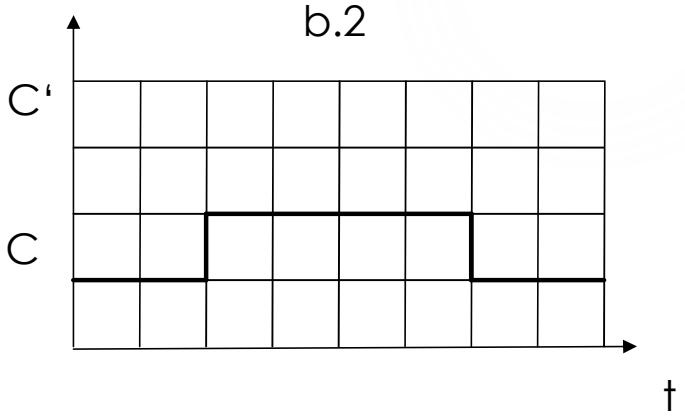
Aufgabe 4 – Flipflops

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

- Das Flipflop soll so erweitert werden, dass es bei der fallenden Flanke die Daten übernimmt. Geben Sie jeweils das Schaltnetz und das zugehörige Zeitverhalten der Flankenerkennung an.

Wir wissen, dass die Verzögerungszeit $\tau = 1$ beträgt. Wir wollen folgendes Verhalten (pro t=1) umsetzen:

C_{alt}	C_{neu}	C‘
0	0	0
0	1	0 (steigend)
1	0	1 (fallend)
1	1	0



Problem: Wir bekommen durch einen Inverter nur $\overline{C_{alt}}$, deshalb formen wir um

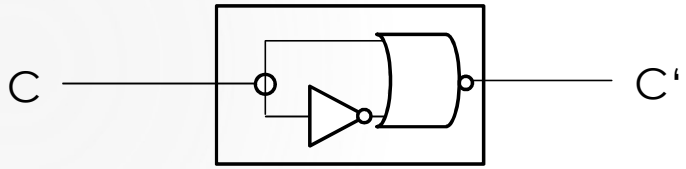
$f(C_{alt}, C_{neu}) = \overline{C_{neu}} \cdot C_{alt} = \overline{\overline{C_{alt}} + C_{neu}}$

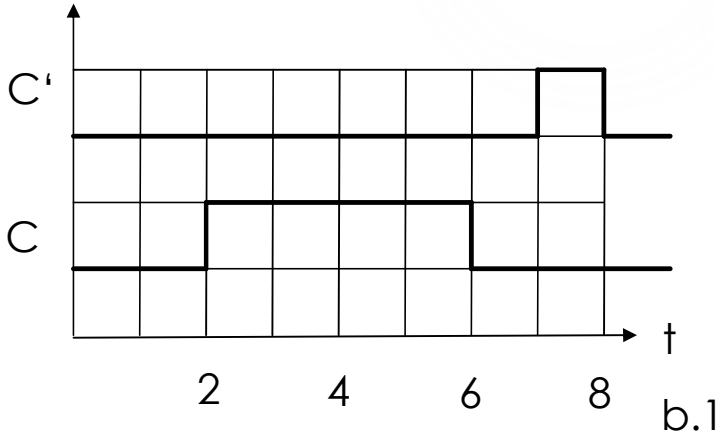
Aufgabe 4 – Flipflops

- Das Flipflop soll so erweitert werden, dass es bei der steigenden Flanke die Daten übernimmt. Geben Sie jeweils das Schaltnetz und das zugehörige Zeitverhalten der Flankenerkennung an.

Wir wissen, dass die Verzögerungszeit $\tau = 1$ beträgt. Dadurch entsteht ein C_{alt} nach dem Inverter, da die obere Leitung keine Verzögerung besitzt. Wir nutzen ein NOR.

$$f(C_{alt}, C_{neu}) = \overline{\overline{C_{alt}} + C_{neu}}$$





b.1

Das NOR-Gatter verzögert das C‘ um eine Zeiteinheit.

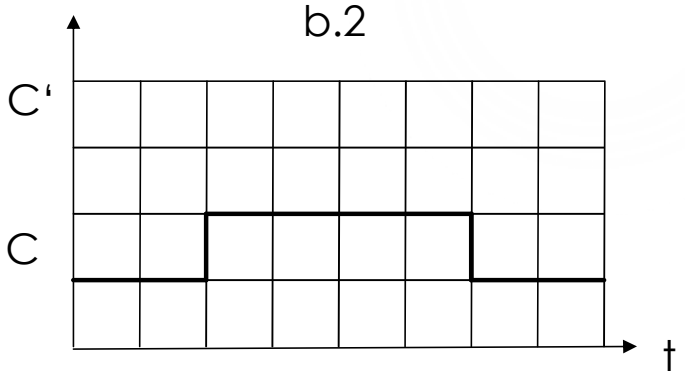
Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 4 – Flipflops

- Das Flipflop soll so erweitert werden, dass es bei beiden Flanken die Daten übernimmt. Geben Sie jeweils das Schaltnetz und das zugehörige Zeitverhalten der Flankenerkennung an.

Wir wissen, dass die Verzögerungszeit $\tau = 1$ beträgt. Wir wollen folgendes Verhalten (pro t=1) umsetzen:

C_{alt}	C_{neu}	C‘
0	0	0
0	1	1 (steigend)
1	0	1 (fallend)
1	1	0



$$f(C_{alt}, C_{neu}) = \overline{C_{neu}} \cdot C_{alt} + \overline{C_{alt}} \cdot C_{neu} = (\overline{C_{neu}} + \overline{C_{alt}}) \cdot (C_{alt} + C_{neu})$$

$$= \overline{\overline{C_{alt}}\ \overline{C_{neu}} + C_{alt}\ C_{neu}} = \overline{\overline{C_{alt}} \oplus C_{neu}}$$

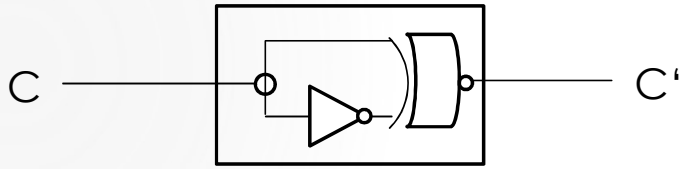
Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

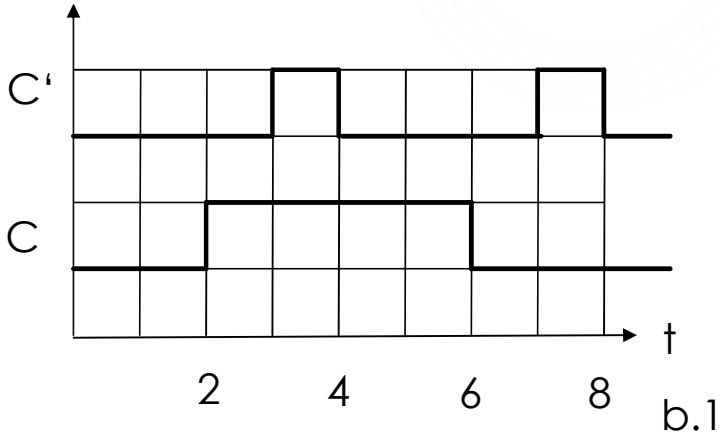
Aufgabe 4 – Flipflops

- Das Flipflop soll so erweitert werden, dass es bei der steigenden Flanke die Daten übernimmt. Geben Sie jeweils das Schaltnetz und das zugehörige Zeitverhalten der Flankenerkennung an.

Wir wissen, dass die Verzögerungszeit $\tau = 1$ beträgt. Dadurch entsteht ein C_{alt} nach dem Inverter, da die obere Leitung keine Verzögerung besitzt. Wir nutzen ein XNOR.

$$f(C_{alt}, C_{neu}) = \overline{\overline{C_{alt}} \oplus C_{neu}}$$





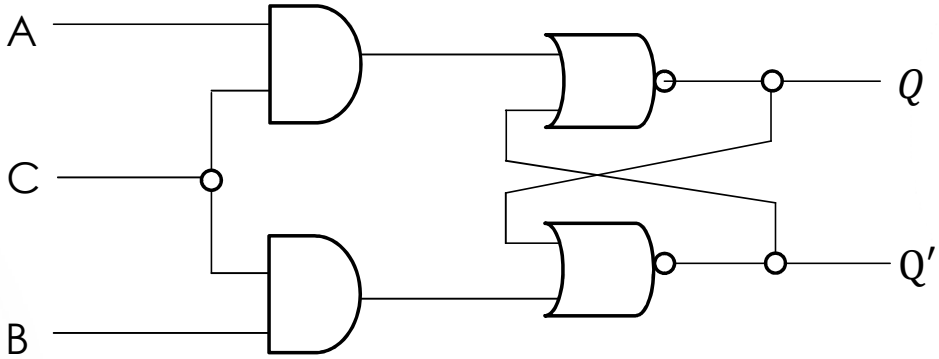
b.1

Das XNOR-Gatter verzögert das C‘ um eine Zeiteinheit.

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 4 – Flipflops

- Erweitern Sie nun die Schaltung dahingehend, dass keine undefinierten Zustände, wie sie in Aufgabe a) der Fall waren, mehr auftreten.

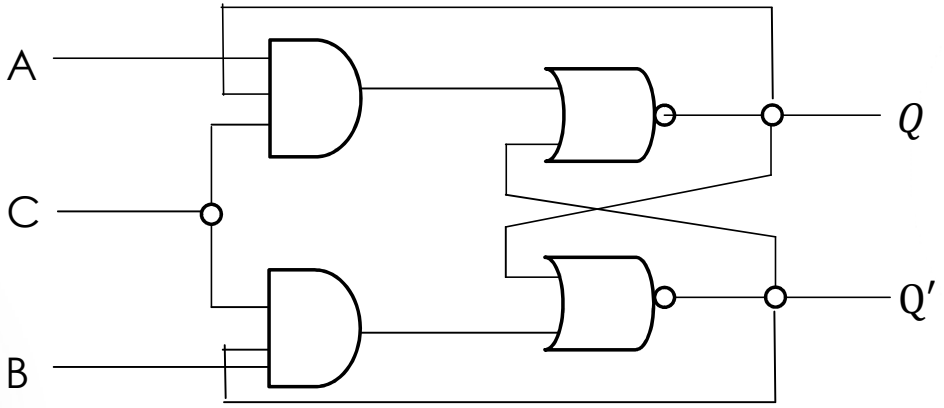


Hinweis: der ungültige Zustand trat bei A = 1 und B = 1 auf. Wir bauen uns jetzt ein JK-Flipflop

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Aufgabe 4 – Flipflops

Lösung: Rückleiten der Ausgabe Q/Q‘ in die Flankensteuerung



Der ungültige Zustand trat bei A = 1 und B = 1 auf.

Wir sperren nun einen der Eingänge A und B, wenn Q bzw. Q‘ vorher 0 ist. Dadurch haben wir eine Differenzierung zwischen oben und unten.

Wir umgehen somit gleiche Werte.

A	B	C	D	Entspricht
0	1	1	0	Reset/Set
0	0	1	0	Keine Änderung
1	1	0	0	Ungültig!
1	0	0	1	Set/Reset
1	1	0	0	Ungültig
0	0	?	?	Oszillation

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck

Vielen Dank für eure großzügige Aufmerksamkeit

‘Ci sono soltanto due possibili conclusioni: se il risultato conferma le ipotesi, allora hai appena fatto una misura; se il risultato è contrario alle ipotesi, allora hai fatto una scoperta.’

Enrico Fermi

Dizionario Biografico degli Italiani, vol. 46.

Friedrich-Alexander Universität Erlangen-Nürnberg
Jan Spieck