

Übungen zur Grundlagen der Technischen Informatik

Übung 10 – Master-Slave, Multiplexer, Shifter und Register

Florian Frank

Friedrich-Alexander-Universität Erlangen-Nürnberg

Wintersemester 2018/19



Was machen wir heute?

Organisatorisches: Vorlesungsevaluation

Organisatorisches – Miniklausur

Blatt 9, Aufgabe 4 – Latches und Flipflops *revisited*

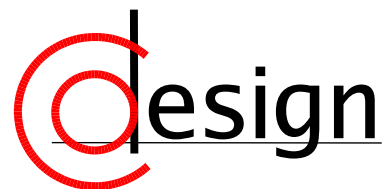
Aufgabe 1 – Master-Slave-Flipflops

Aufgabe 2 – Multiplexer und Demultiplexer

Aufgabe 3 – Barrel-Shifter

Aufgabe 4 – Schieberegister

Organisatorisches: Vorlesungsevaluation



Organisatorisches – Vorlesungsevaluation

Bitte evaluiert die Veranstaltung, nur so können Dinge verbessert werden!
Auch bei keinen Verbesserungsvorschlägen freuen wir uns immer über positives Feedback, damit wir sehen, dass alles so gepasst hat!

- Bei Kommentaren in Freitextfeldern, die sich auf einen **bestimmten** Übungsleiter beziehen, gebt bitte **dessen Name bei diesen Kommentaren** mit an.
 - ↪ Kommentare werden durcheinandergewürfelt ...
 - Das heißt nicht nur einmal den Namen angeben, sondern immer!
 - ↪ Ihr evaluiert die Gesamtveranstaltung „Übungen zu den Grundlagen der Technischen Informatik“ und nicht – wie in AuD oder GRa bspw. – die einzelnen Übungen, deswegen gebt bitte die Namen mit an, wir wären euch sehr verbunden.

Ihr habt noch bis zum **26.01 um 12⁰⁰ Uhr** Zeit zu evaluieren!

Web-basierte Evaluation an der Technischen Fakultät

Evaluation der Vorlesungen, Übungen, Seminare u. Praktika

- Fragen zur Lehrveranstaltung, zur Dozentin/zum Dozent
+ optionale Zusatzfragen (von der Dozentin/vom Dozenten gestellt)
- Die **TAN-Zettel für die Lehrveranstaltungen** erhalten Sie in der LV jeweils von der Dozentin/vom Dozenten. Bei Nichtbenutzung bitte zerstören!

Informationen und Evaluation → <http://eva.tf.fau.de>

Die Auswertung der LV-Umfragen erfolgt automatisiert und kurzfristig, um die Erkenntnisse noch vor Semesterende mit den Studierenden diskutieren zu können.

Tragen Sie zu einem repräsentativen Ergebnis bei!

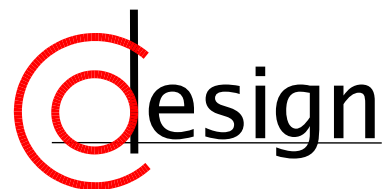
Evaluieren Sie unter:

<http://eva.tf.fau.de>

Frist: Sa., 27. Januar 2018, 12⁰⁰ Uhr (!)



Organisatorisches – Miniklausur



Achtung – Miniklausur

Achtung – Miniklausur

Diesen Donnerstag – am 17. Januar 2019 – findet die **zweite** Miniklausur statt.

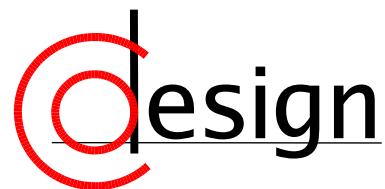
Wo? im H7

Wann? Wir starten um 16:15 Uhr!

*Seid bitte deswegen schon **ungefähr 3 – 5 Minuten** vor Beginn da.*

Worum gehts? Um den Übungsstoff bis Blatt 10 – sprich heute –
und den Vorlesungsstoff bis einschließlich zum 15.01.2019

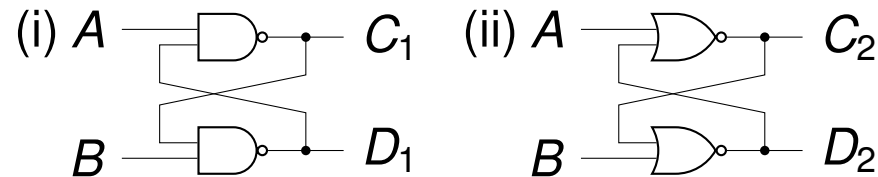
Blatt 9, Aufgabe 4 – Latches und Flipflops *revisited*



Aufgabe 4 – Latches und Flipflops

Die Verzögerungszeit jedes Logikgatters in dieser Aufgabe betrage $\tau = 1$ ns.

- a) An die Eingänge (A , B) der unten stehenden Schaltungen (i) und (ii) werden nacheinander folgende Werte angelegt: (0, 1), (0, 0), (1, 1), (1, 0), (1, 1) und (0, 0). Geben Sie jeweils die Ausgangswerte von (i) und (ii) an und benennen Sie die Signale A bis D_2 sinnvoll.



Aufgabe 4 – Latches und Flipflops: Begriffklärung

Speicherelement

Ein Speicherelement ist ein Gerät/Modul/Element, das einen vorher angelegten Wert speichert.

Aufgabe 4 – Latches und Flipflops: Begriffklärung

Speicherelement

Ein Speicherelement ist ein Gerät/Modul/Element, das einen vorher angelegten Wert speichert.

Eigenschaften von Speicherelementen

Speicherelemente können ...

Operation	Q^t	Q^{t+1}
S(etze)	—	1
R(ücksetze)	—	0
N(ix)	0	0
	1	1

Operationen eines Speicherelements

Aufgabe 4 – Latches und Flipflops: Begriffklärung

Speicherelement

Ein Speicherelement ist ein Gerät/Modul/Element, das einen vorher angelegten Wert speichert.

Eigenschaften von Speicherelementen

Speicherelemente können ...

asynchron sein, wenn Änderungen jederzeit möglich sind.

Operation	Q^t	Q^{t+1}
S(etze)	—	1
R(ücksetze)	—	0
N(ix)	0	0
	1	1

Operationen eines Speicherelements

Aufgabe 4 – Latches und Flipflops: Begriffklärung

Speicherelement

Ein Speicherelement ist ein Gerät/Modul/Element, das einen vorher angelegten Wert speichert.

Eigenschaften von Speicherelementen

Speicherelemente können ...

asynchron sein, wenn Änderungen jederzeit möglich sind.

synchron sein, wenn Änderungen nur zu vorher festgelegten Momenten möglich sind. Diese Momente können ...

Operation	Q^t	Q^{t+1}
S(etze)	—	1
R(ücksetze)	—	0
N(ix)	0	0
	1	1

Operationen eines Speicherelements

Aufgabe 4 – Latches und Flipflops: Begriffklärung

Speicherelement

Ein Speicherelement ist ein Gerät/Modul/Element, das einen vorher angelegten Wert speichert.

Eigenschaften von Speicherelementen

Speicherelemente können ...

asynchron sein, wenn Änderungen jederzeit möglich sind.

synchron sein, wenn Änderungen nur zu vorher festgelegten Momenten möglich sind. Diese Momente können ...

pegelabhängig sein. Man nennt diese Elemente dann auch **pegelgesteuert**.

Operation	Q^t	Q^{t+1}
S(etze)	—	1
R(ücksetze)	—	0
N(ix)	0	0
	1	1

Operationen eines Speicherelements

Aufgabe 4 – Latches und Flipflops: Begriffklärung

Speicherelement

Ein Speicherelement ist ein Gerät/Modul/Element, das einen vorher angelegten Wert speichert.

Eigenschaften von Speicherelementen

Speicherelemente können ...

asynchron sein, wenn Änderungen jederzeit möglich sind.

synchron sein, wenn Änderungen nur zu vorher festgelegten Momenten möglich sind. Diese Momente können ...

pegelabhängig sein. Mann nennt diese Elemente dann auch **pegelgesteuert**.

taktflankenabhängig sein. Mann nennt diese Elemente dann auch **taktflankengesteuert**.

Es können zu **steigender**, **fallender** oder **zu beiden** Flanken Änderungen möglich sein.

Operation	Q^t	Q^{t+1}
S(etze)	—	1
R(ücksetze)	—	0
N(ix)	0	0
	1	1

Operationen eines Speicherelements

Aufgabe 4 – Latches und Flipflops: Begriffklärung

Speicherelement

Ein Speicherelement ist ein Gerät/Modul/Element, das einen vorher angelegten Wert speichert.

Flipflops

Flipflops sind Elemente zur Speicherung eines Bits.

Sie sind – bei uns – **rein taktflankengesteuert**.

Aufgabe 4 – Latches und Flipflops: Begriffklärung

Speicherelement

Ein Speicherelement ist ein Gerät/Modul/Element, das einen vorher angelegten Wert speichert.

Latches

Latches sind ebenfalls Elemente zur Speicherung eines Bits, ähnlich zu den Flipflops.

Im Gegensatz zu Flipflops sind sie aber **rein pegelgesteuert..**

Aufgabe 4 – Latches und Flipflops: Begriffklärung

Speicherelement

Ein Speicherelement ist ein Gerät/Modul/Element, das einen vorher angelegten Wert speichert.

Flipflops

Flipflops sind Elemente zur Speicherung eines Bits.

Sie sind – bei uns – **rein taktflankengesteuert**.

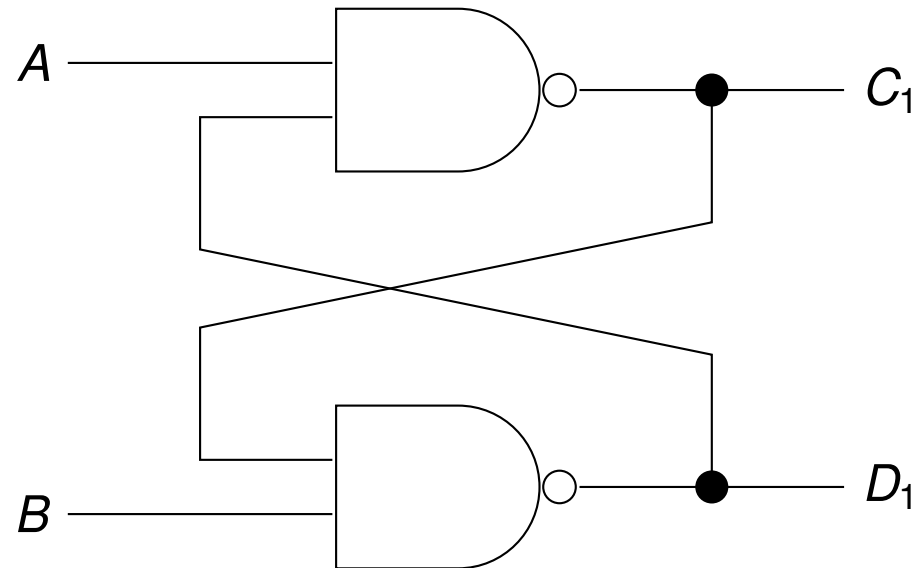
Latches

Latches sind ebenfalls Elemente zur Speicherung eines Bits, ähnlich zu den Flipflops.

Im Gegensatz zu Flipflops sind sie aber **rein pegelgesteuert**.

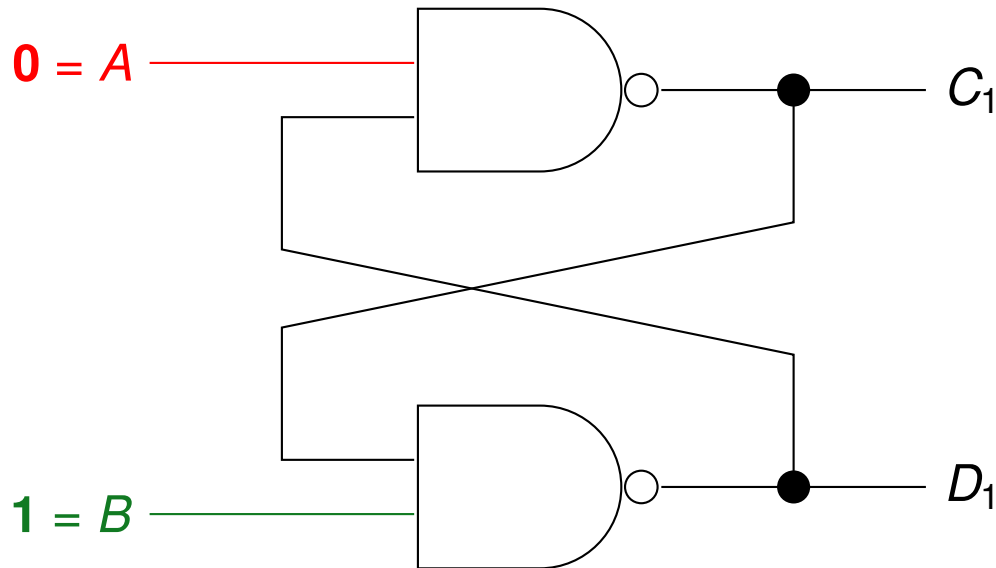
Aufgabe 4 – Latches und Flipflops

Anliegender Wert (**fett** hervorgehoben):



Aufgabe 4 – Latches und Flipflops

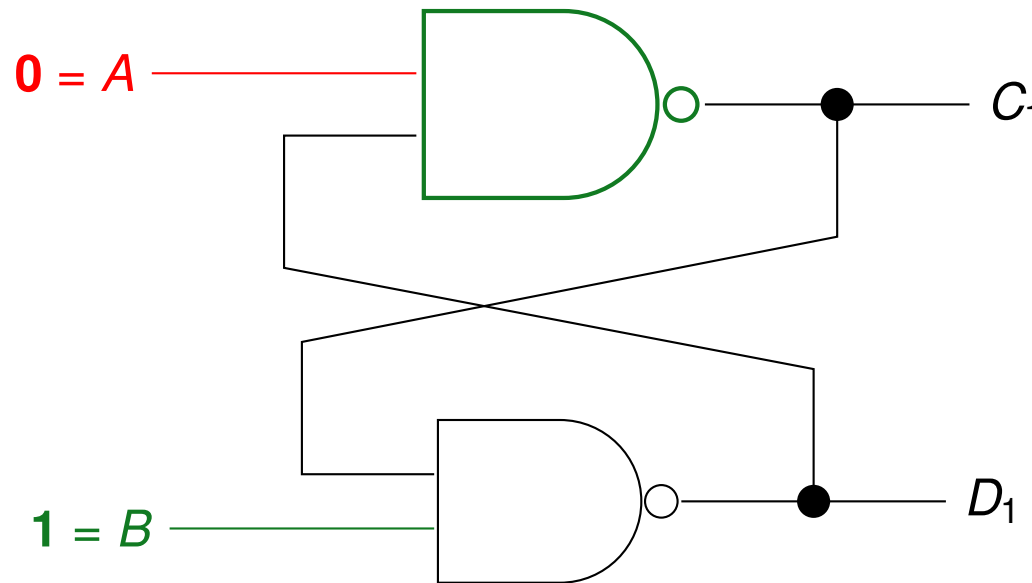
Anliegender Wert (**fett** hervorgehoben): **(0,1)**



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

Anliegender Wert (**fett** hervorgehoben): **(0,1)**

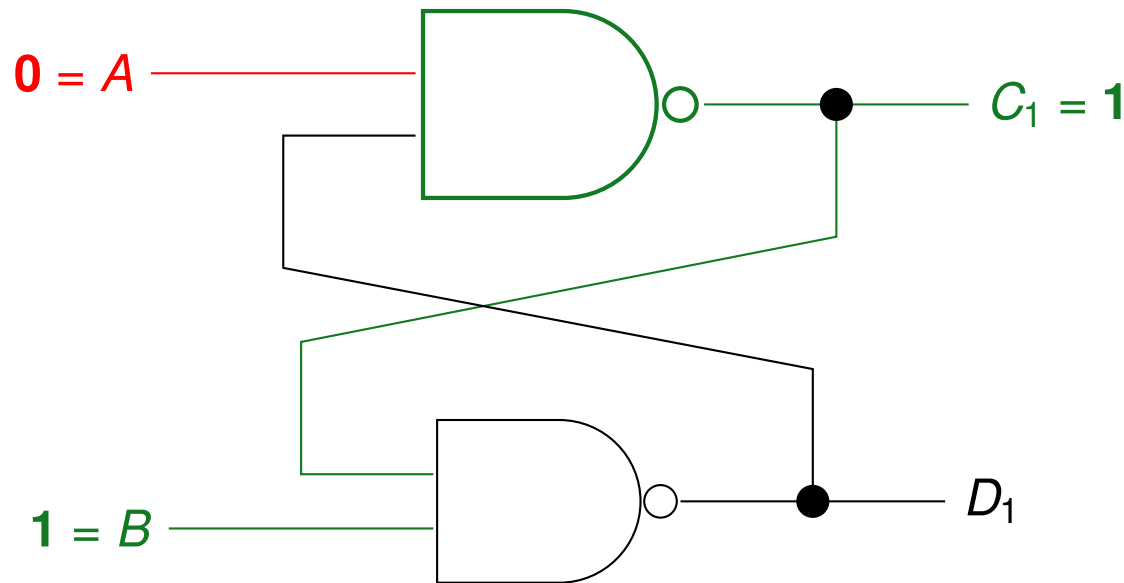


Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.

Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

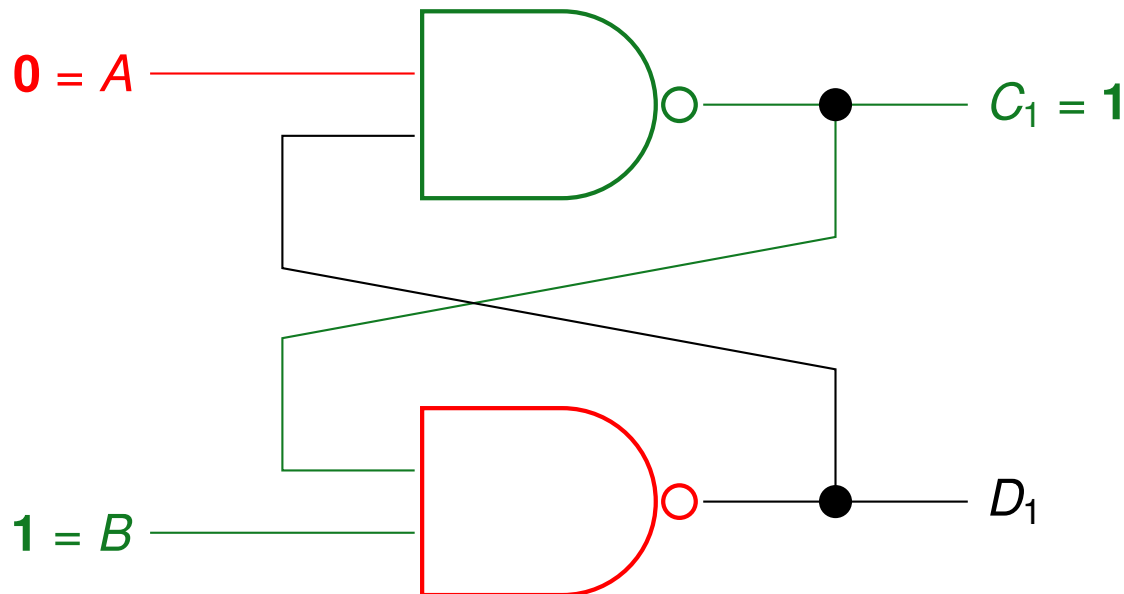
Anliegender Wert (**fett** hervorgehoben): **(0,1)**



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

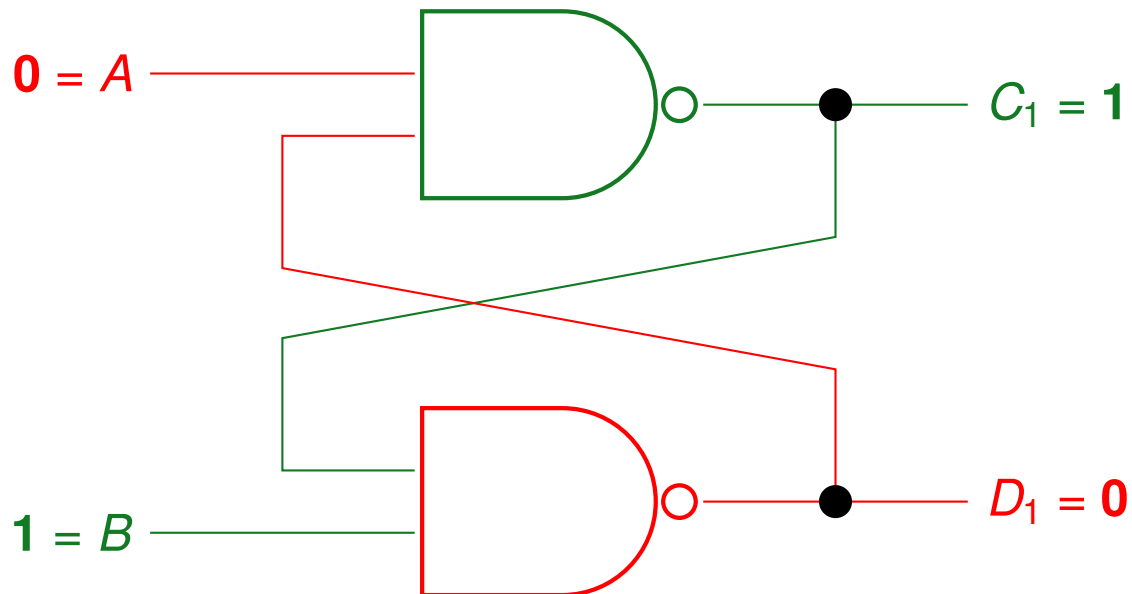
Anliegender Wert (**fett** hervorgehoben): **(0,1)**



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

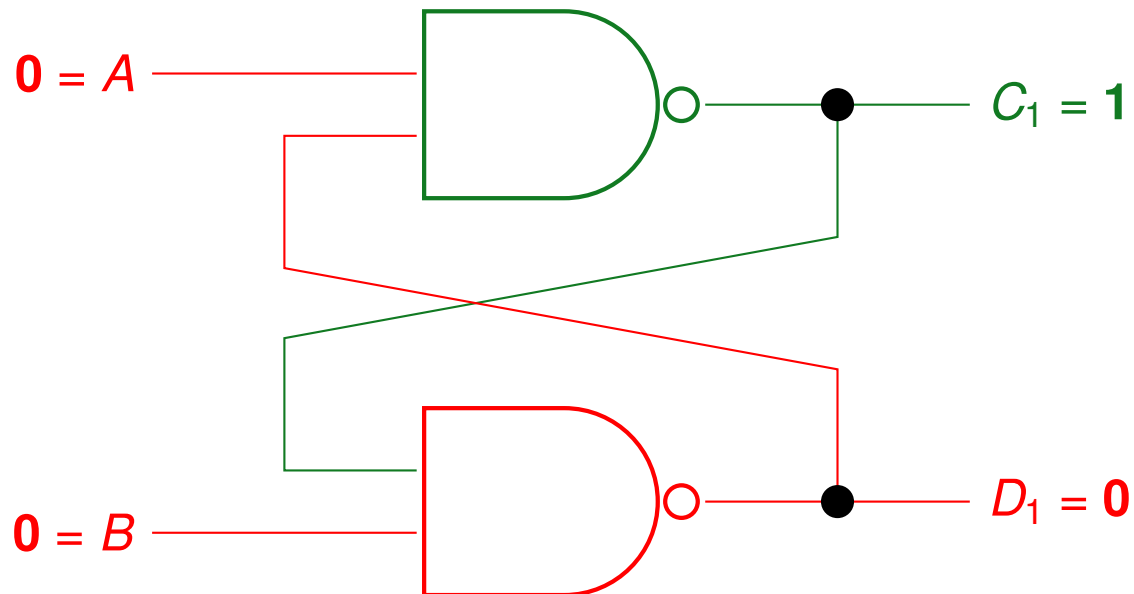
Anliegender Wert (**fett** hervorgehoben): **(0,1)**



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

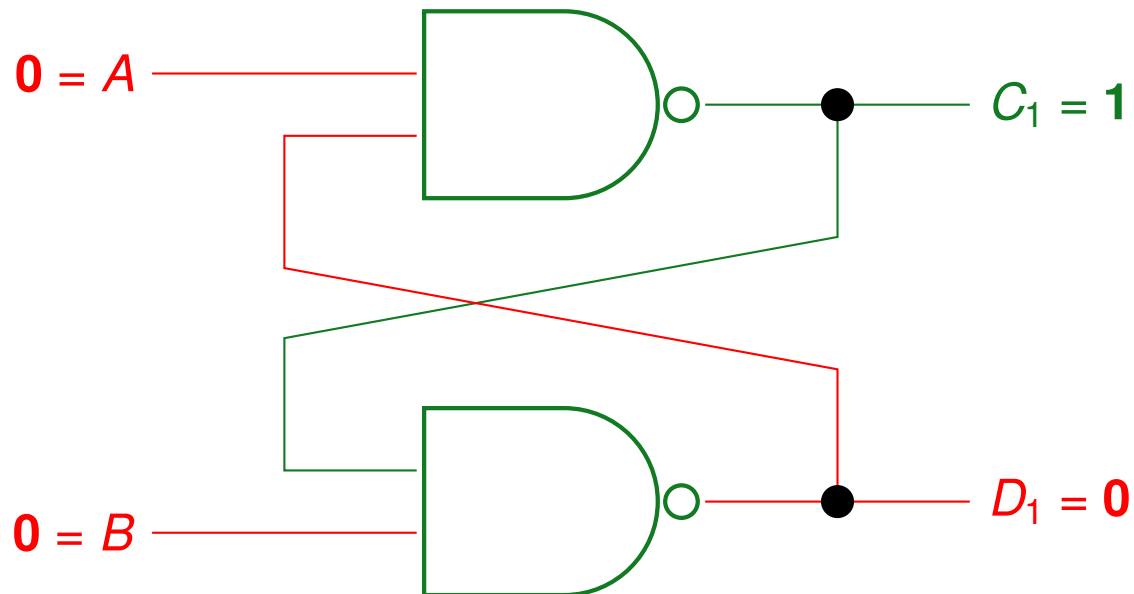
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

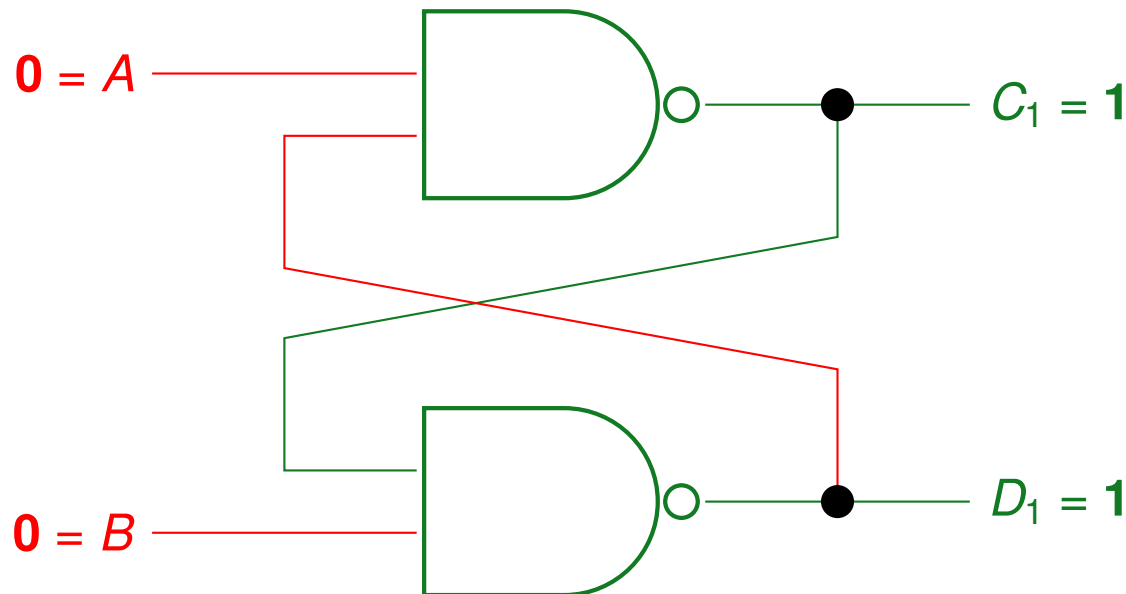
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

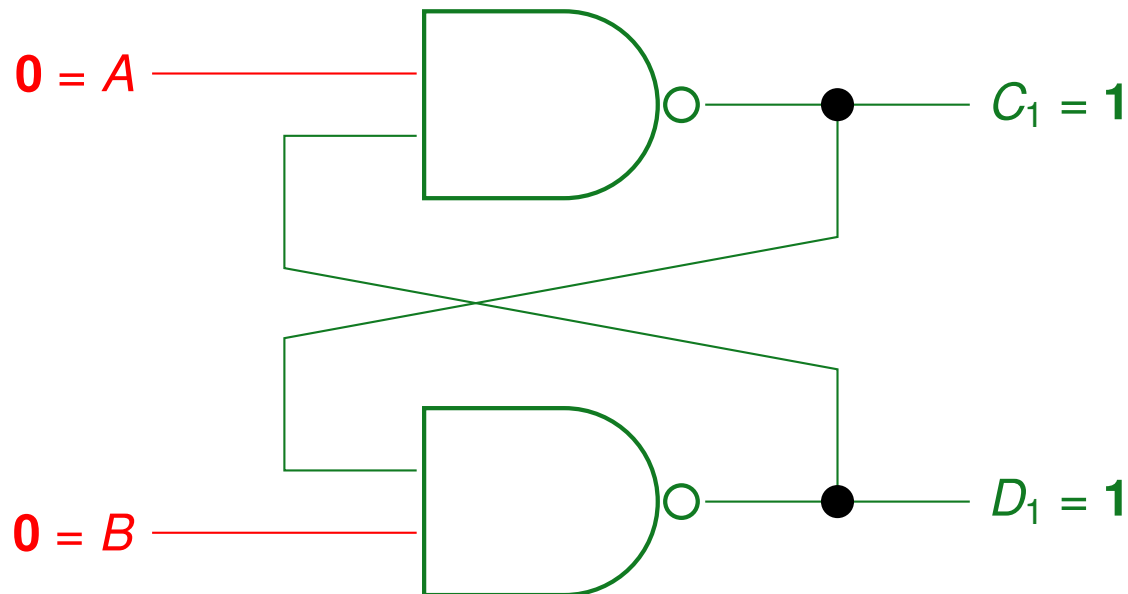
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

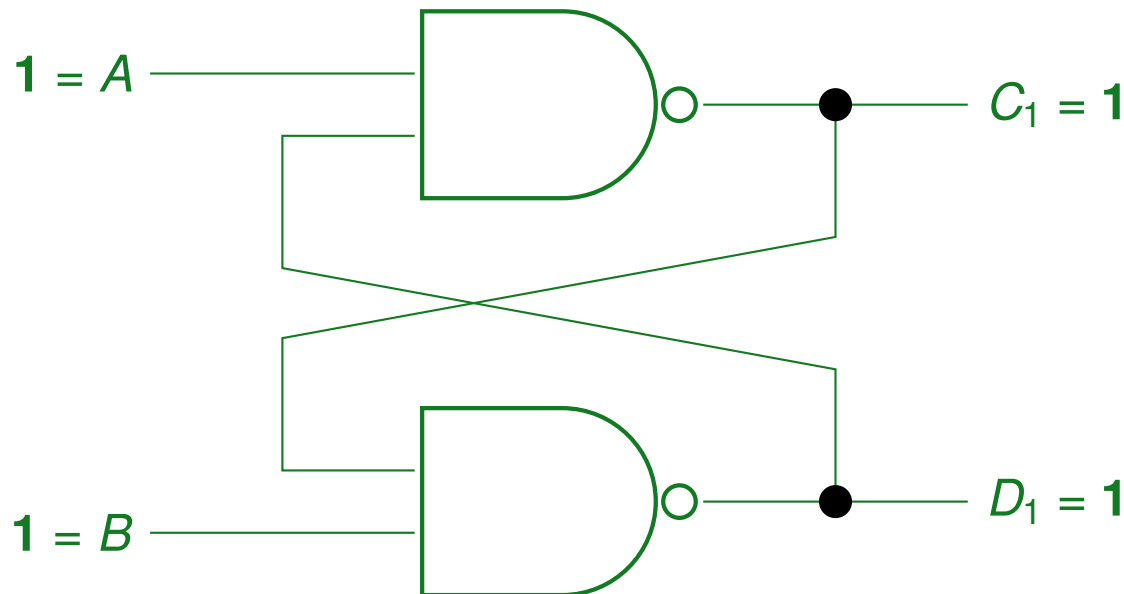
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

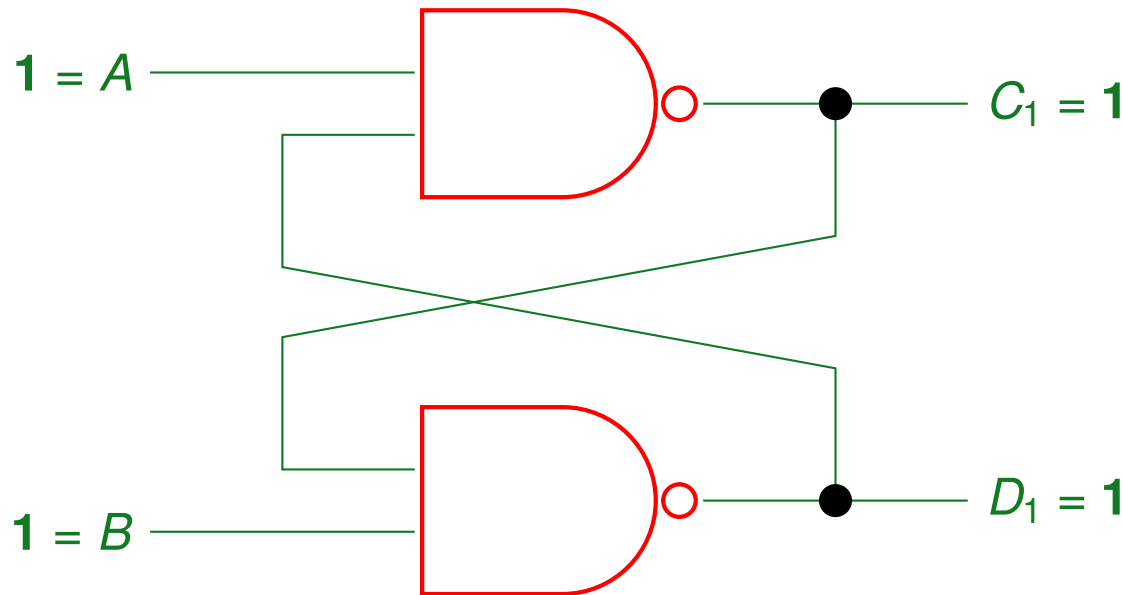
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0) \mapsto (1, 1)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

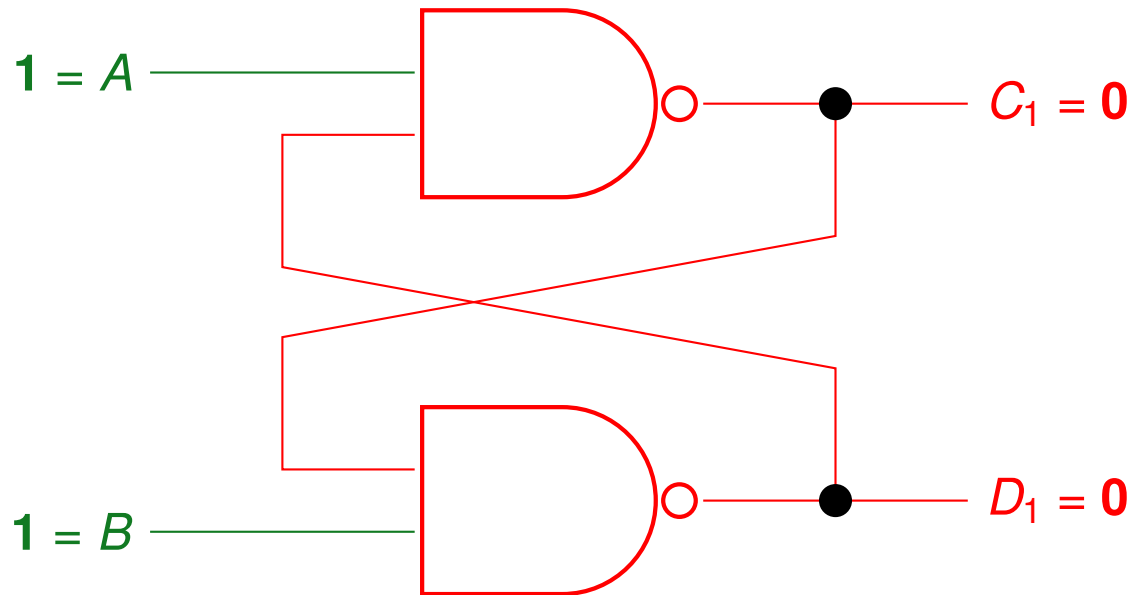
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0) \mapsto (1, 1)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

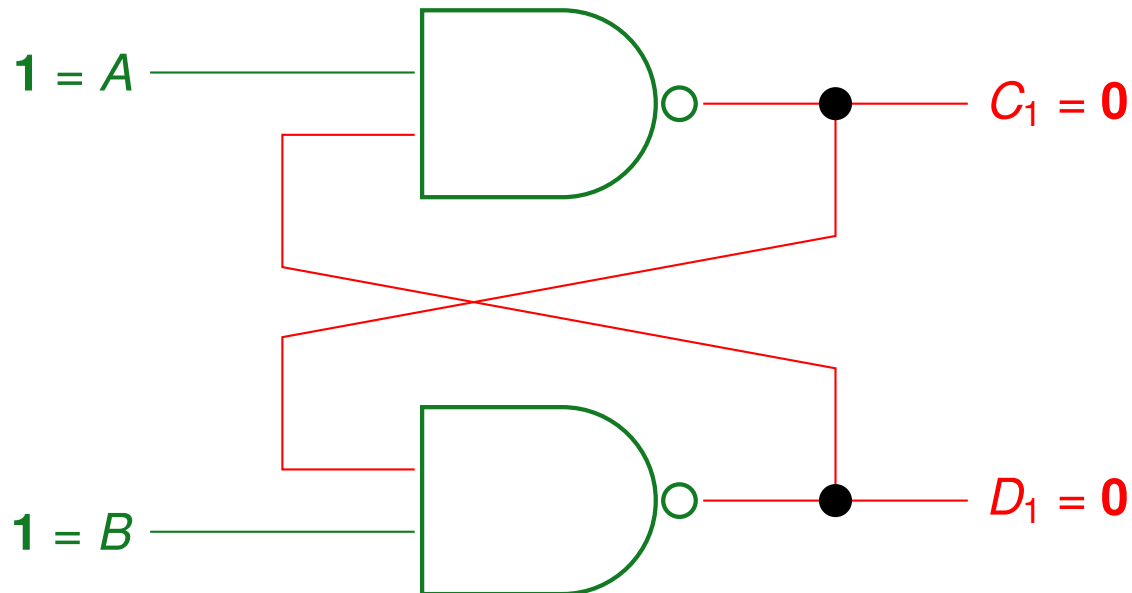
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0) \mapsto (1, 1)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

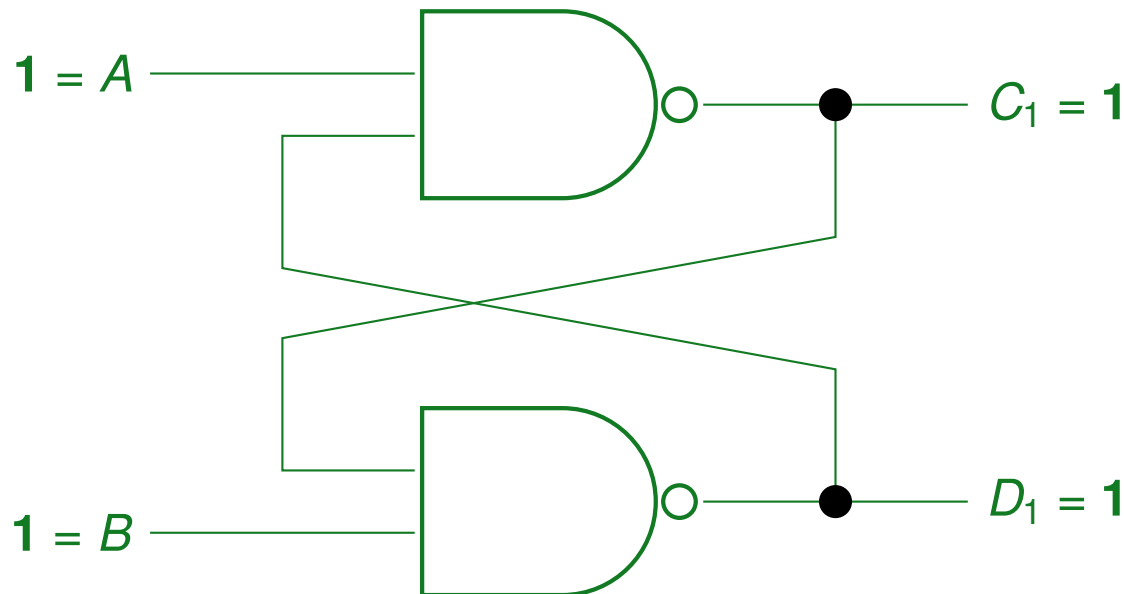
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0) \mapsto (1, 1)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

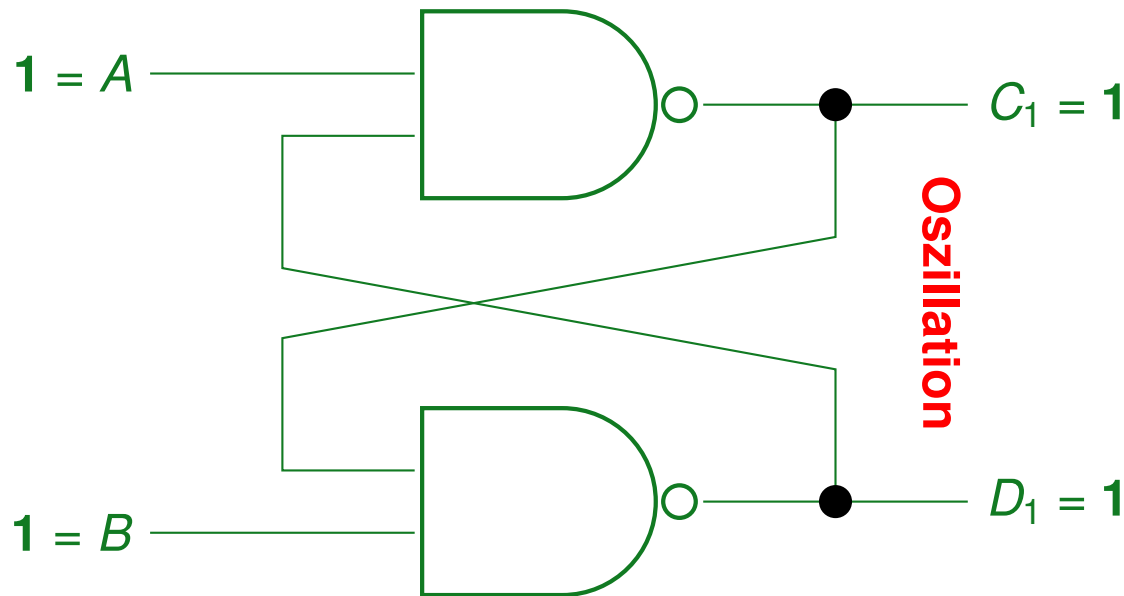
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0) \mapsto \mathbf{(1, 1)}$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

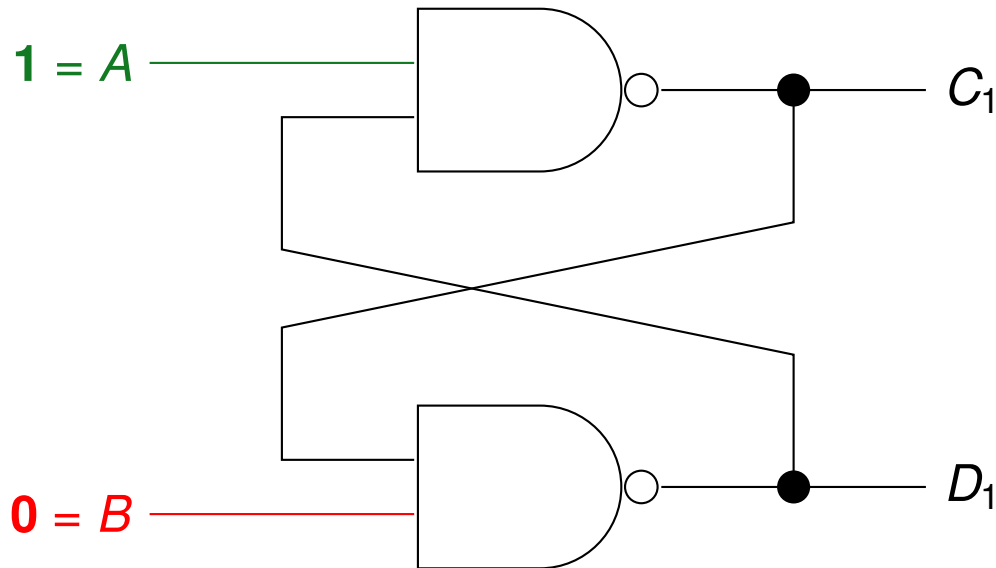
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0) \mapsto (1, 1)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

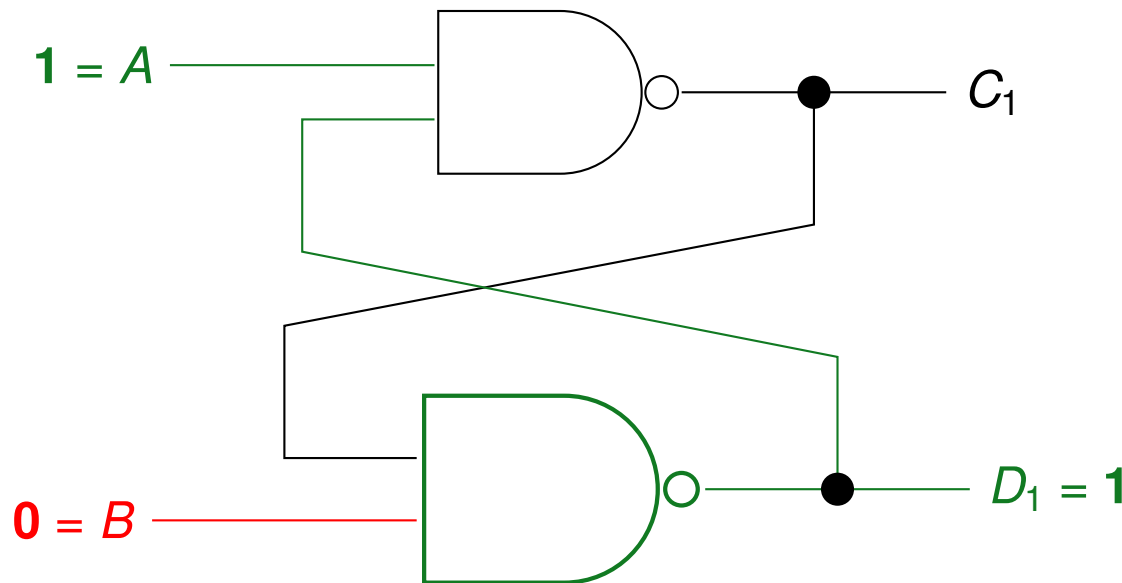
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (0, 0) \mapsto (1, 1) \mapsto (1, 0)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (0, 0) \mapsto (1, 1) \mapsto (1, 0)$

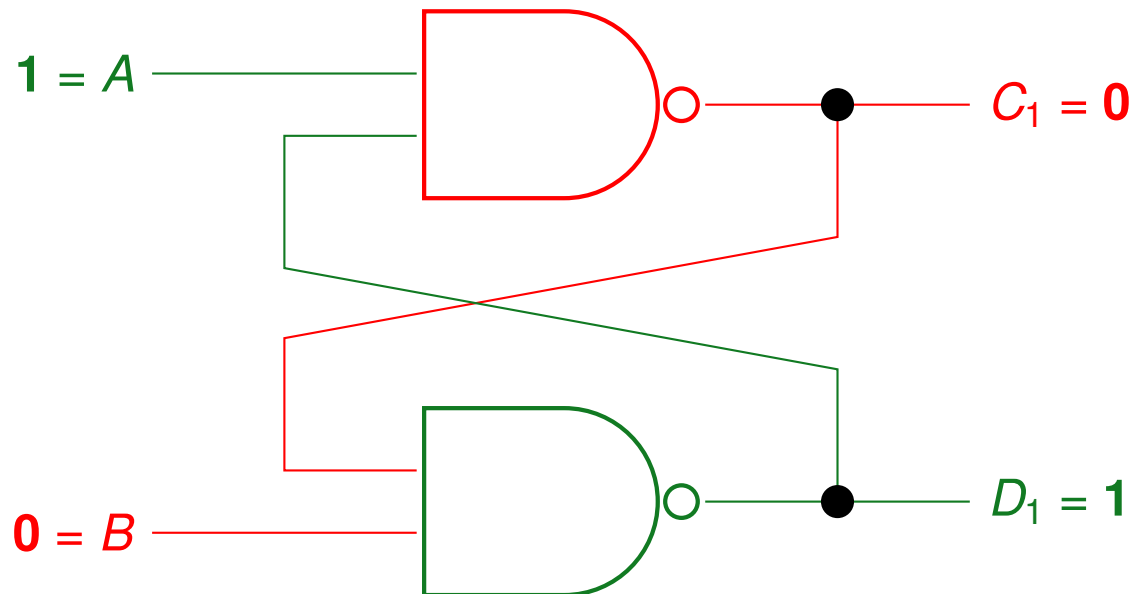


Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.

Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

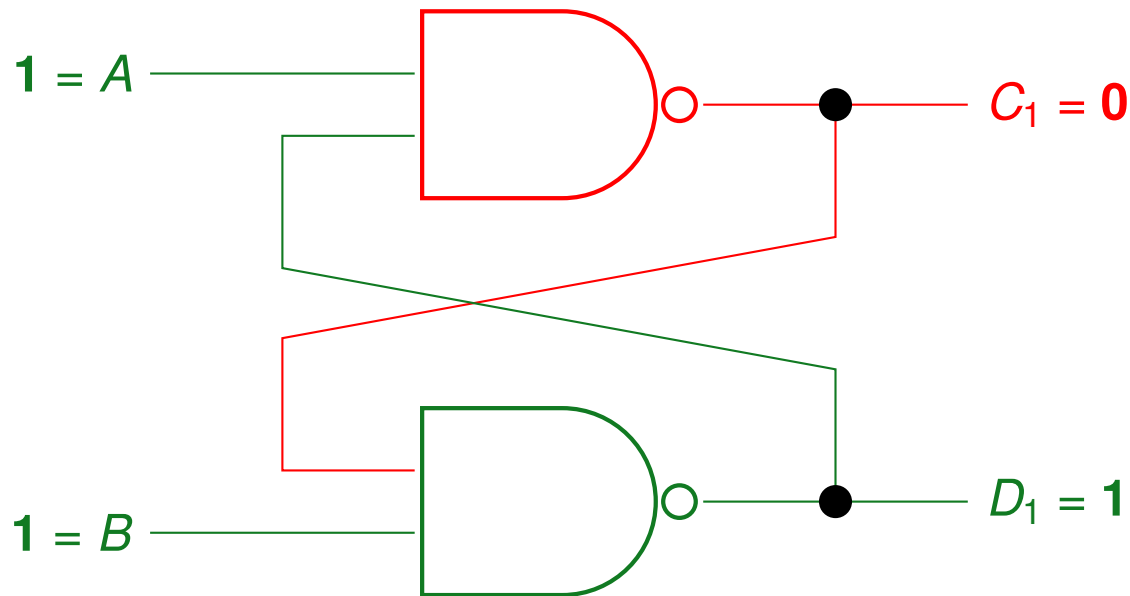
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (0, 0) \mapsto (1, 1) \mapsto (1, 0)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

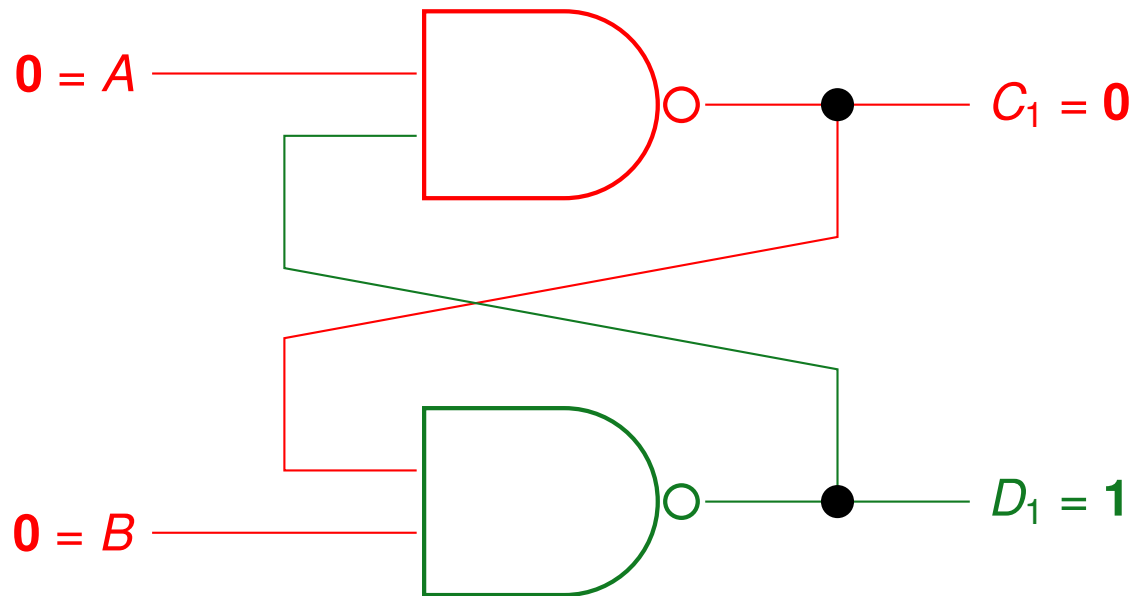
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 1) \mapsto (1, 0) \mapsto (1, 1)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

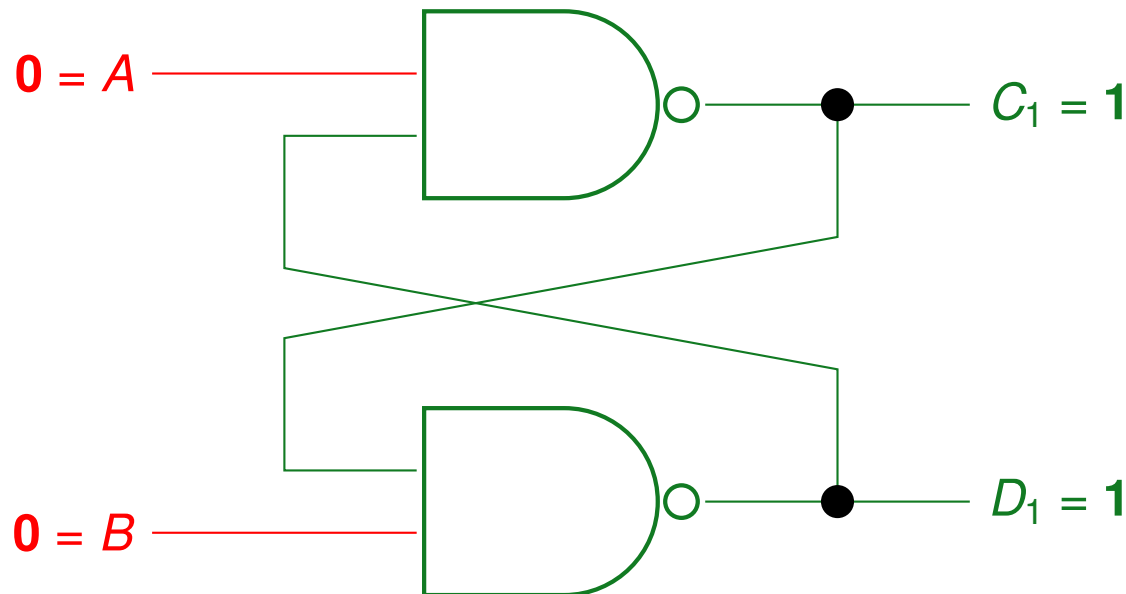
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 0) \mapsto (1, 1) \mapsto (0, 0)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

Aufgabe 4 – Latches und Flipflops

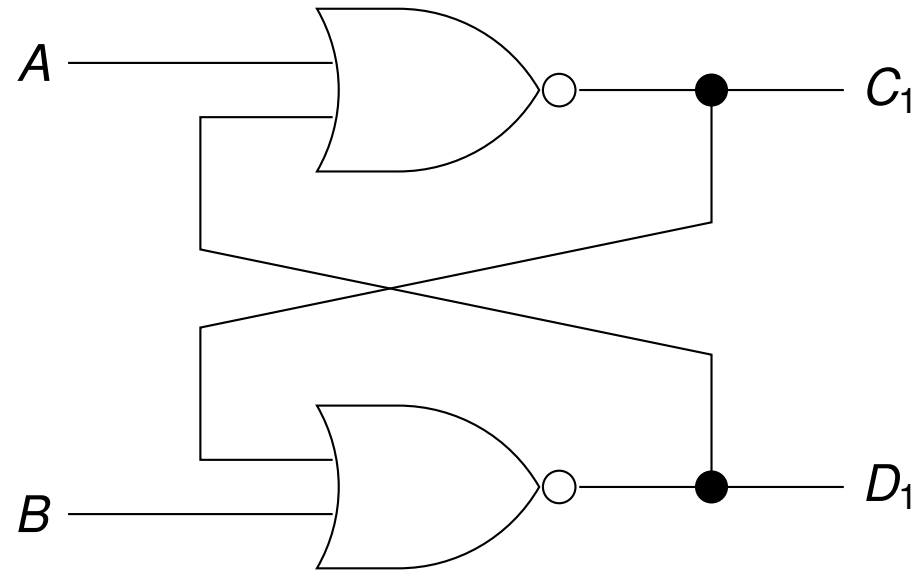
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 0) \mapsto (1, 1) \mapsto (0, 0)$



Hat ein NAND mindestens **eine** 0 im Eingang, so ist der Ausgang 1.
 Hat ein NAND **keine** 0 im Eingang (sprich: nur Einsen), so ist der Ausgang 0.

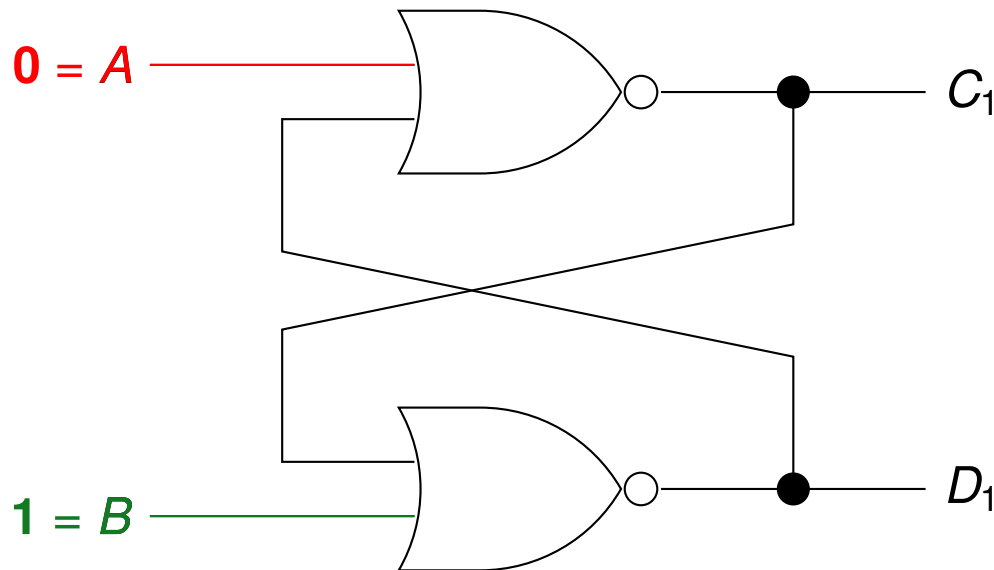
Aufgabe 4 – Latches und Flipflops

Anliegender Wert (**fett** hervorgehoben):



Aufgabe 4 – Latches und Flipflops

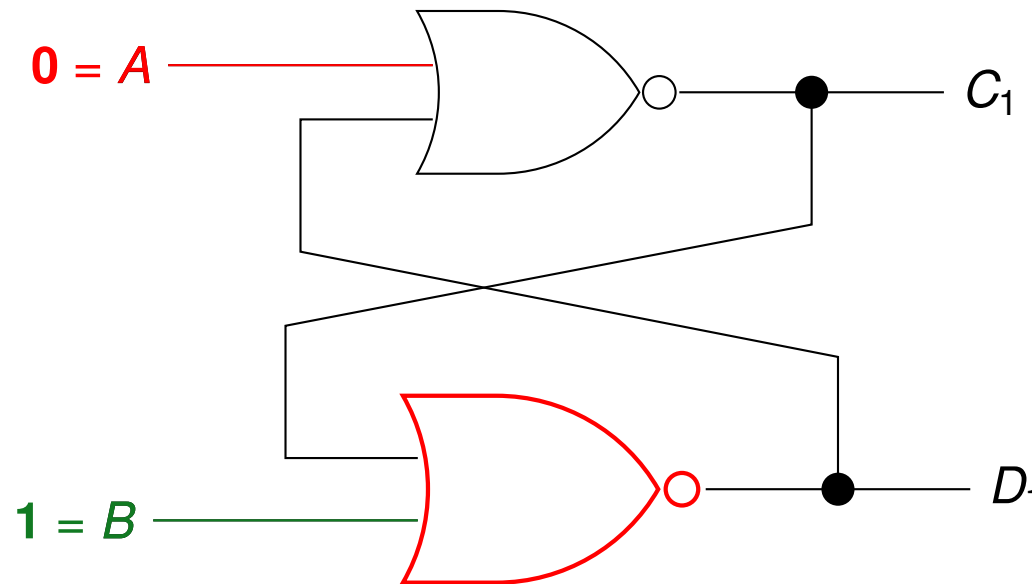
Anliegender Wert (**fett** hervorgehoben): **(0,1)**



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

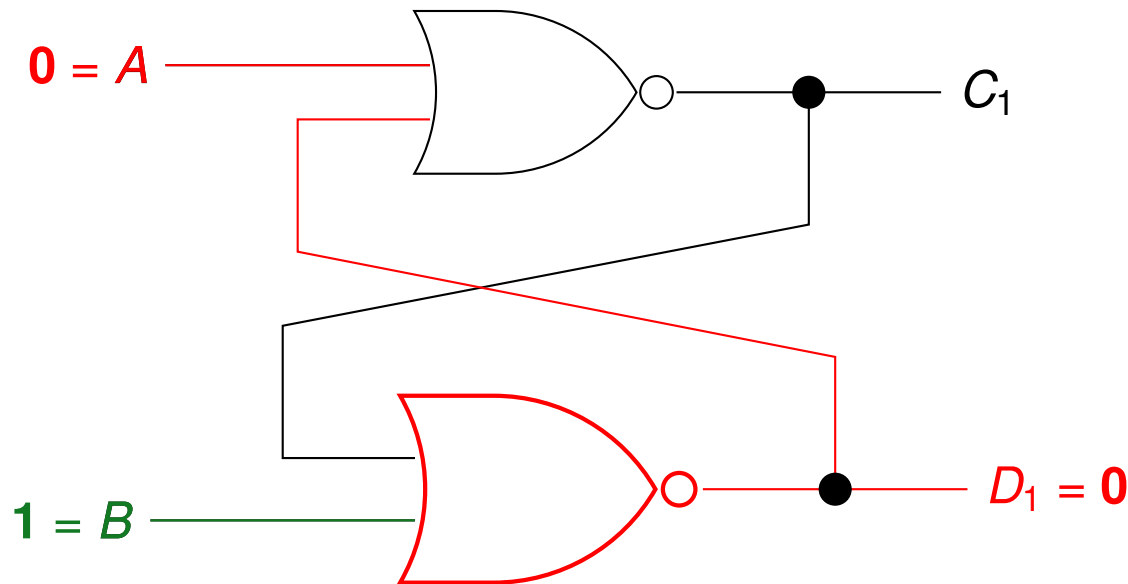
Anliegender Wert (**fett** hervorgehoben): **(0,1)**



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

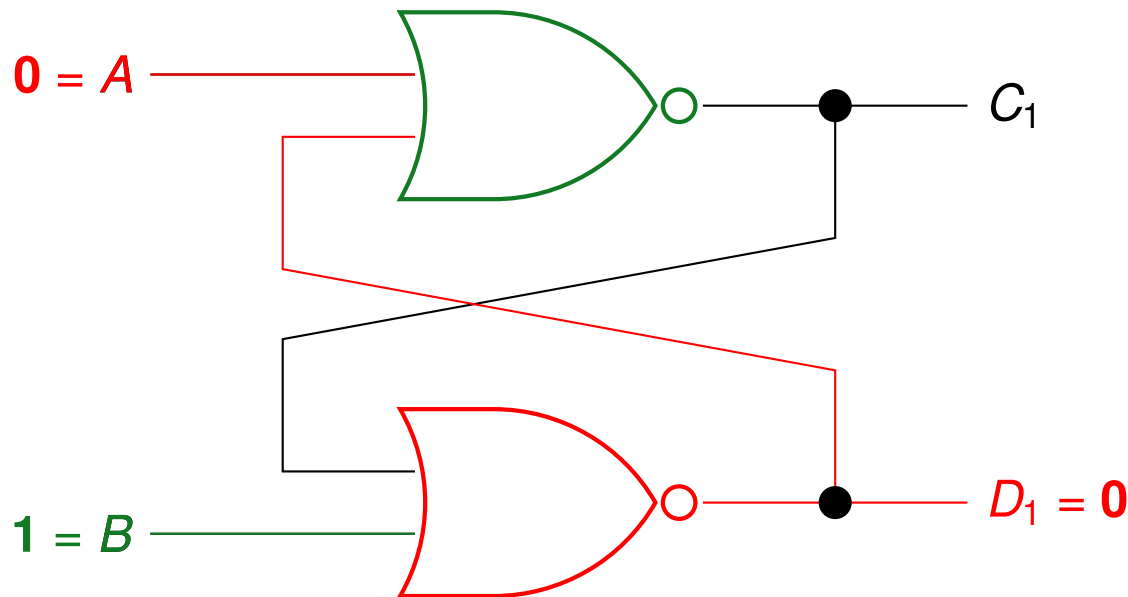
Anliegender Wert (**fett** hervorgehoben): **(0,1)**



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

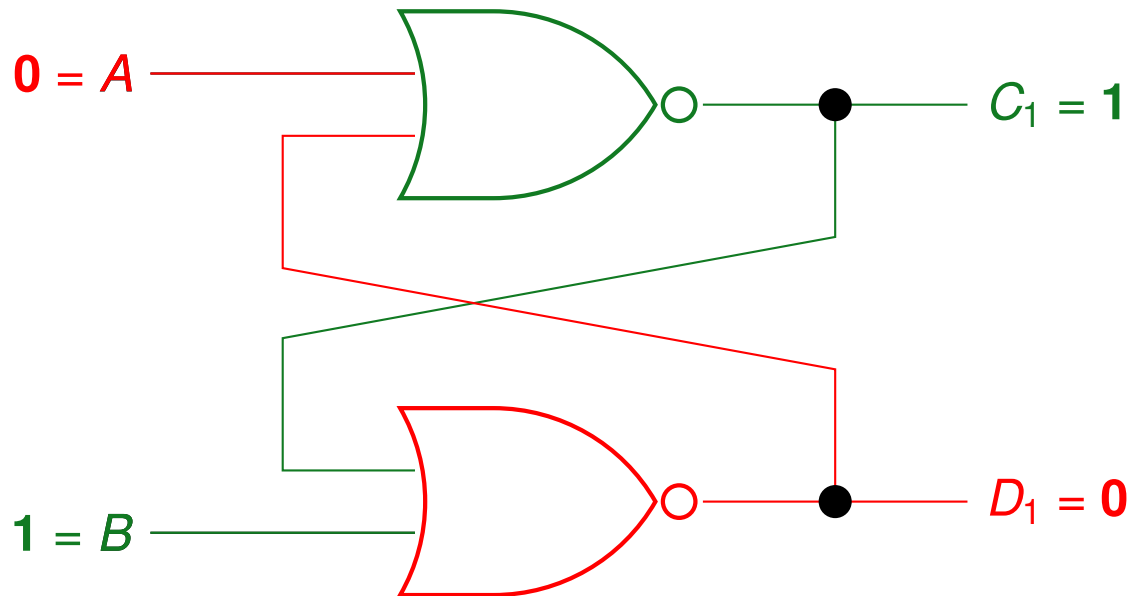
Anliegender Wert (**fett** hervorgehoben): **(0,1)**



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

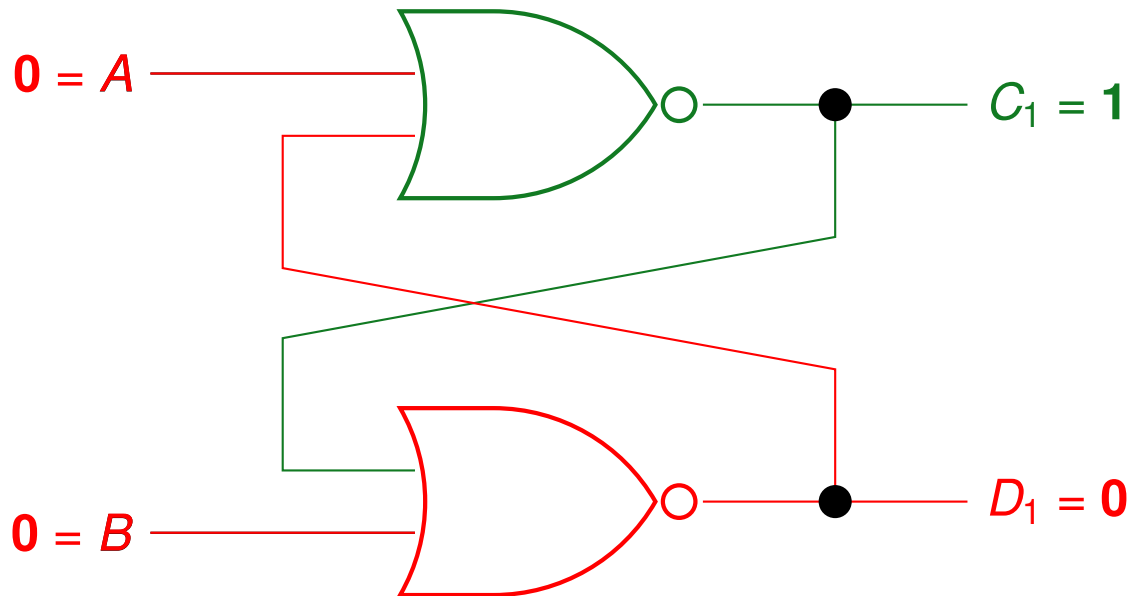
Anliegender Wert (**fett** hervorgehoben): **(0,1)**



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

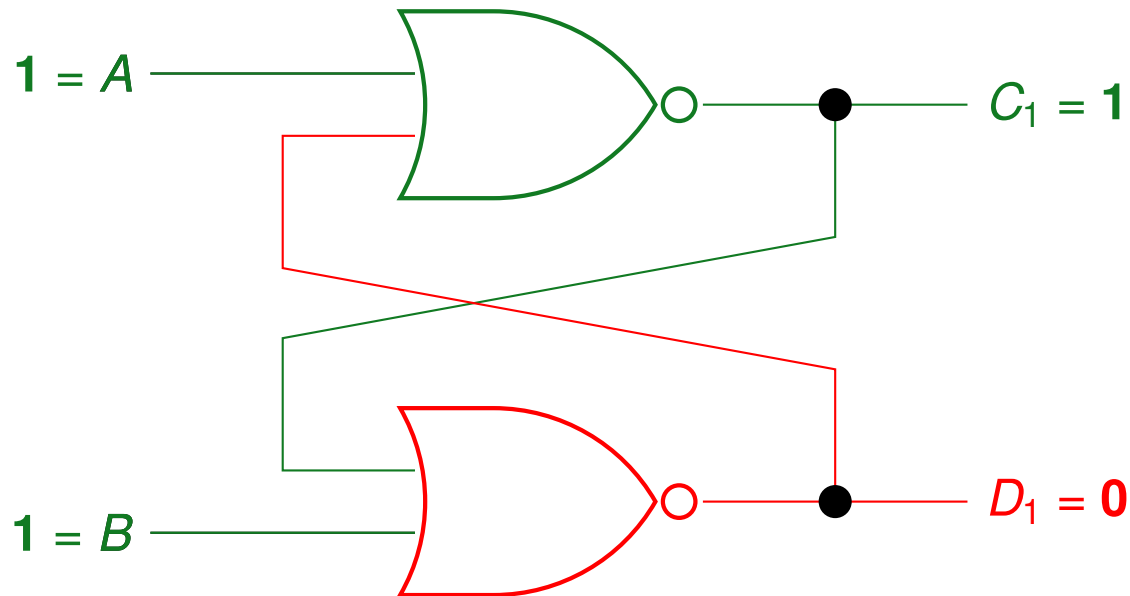
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

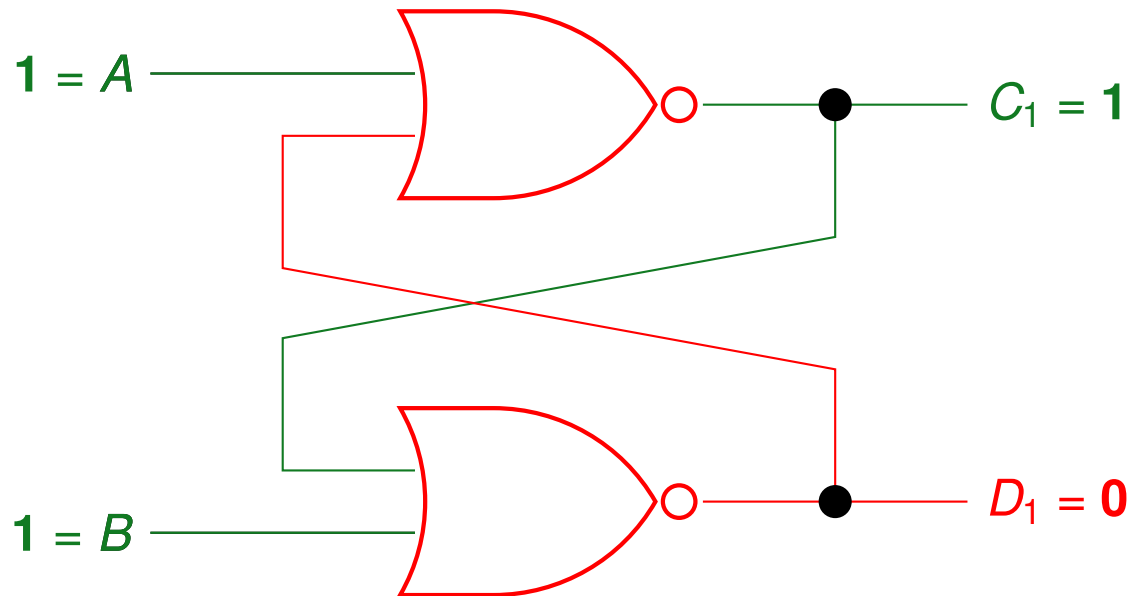
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0) \mapsto (1, 1)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

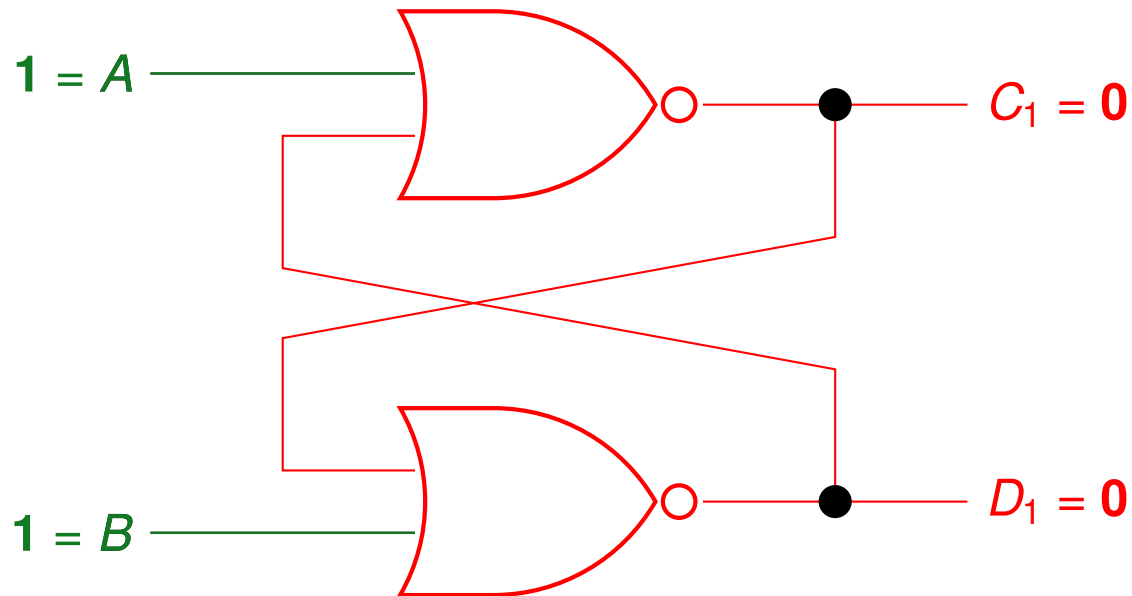
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0) \mapsto (1, 1)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

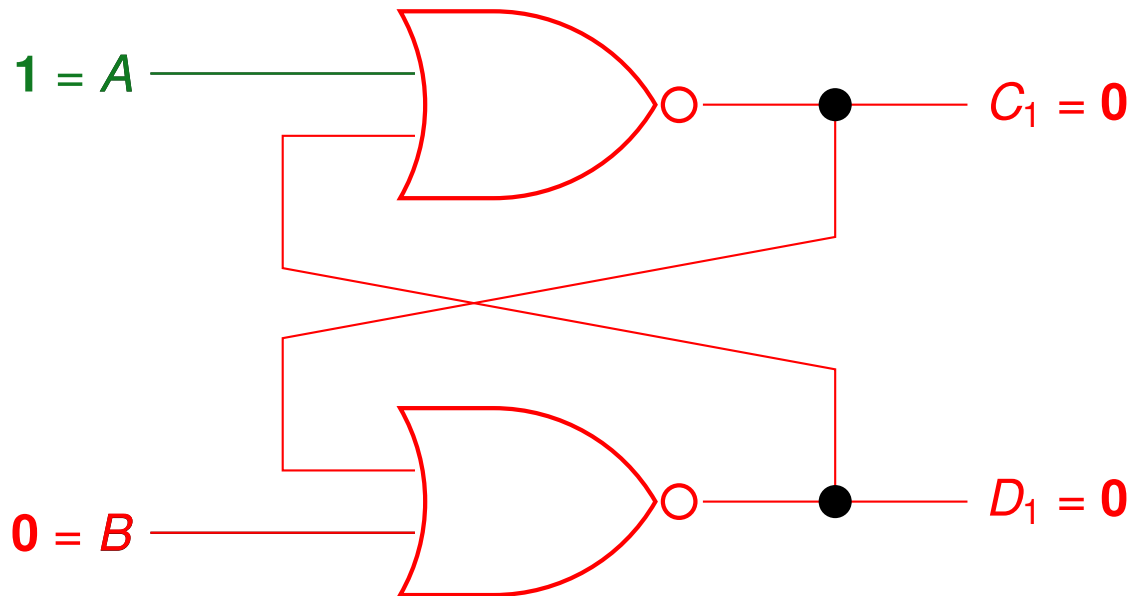
Anliegender Wert (**fett** hervorgehoben): $(0,1) \mapsto (0, 0) \mapsto (1, 1)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

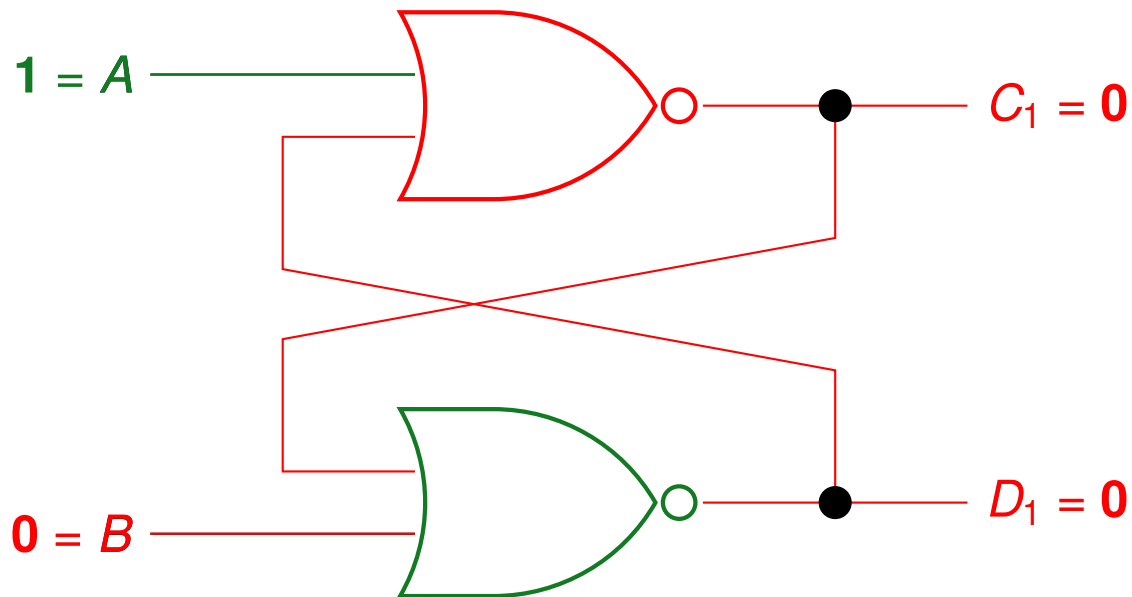
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (0, 0) \mapsto (1, 1) \mapsto (1, 0)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

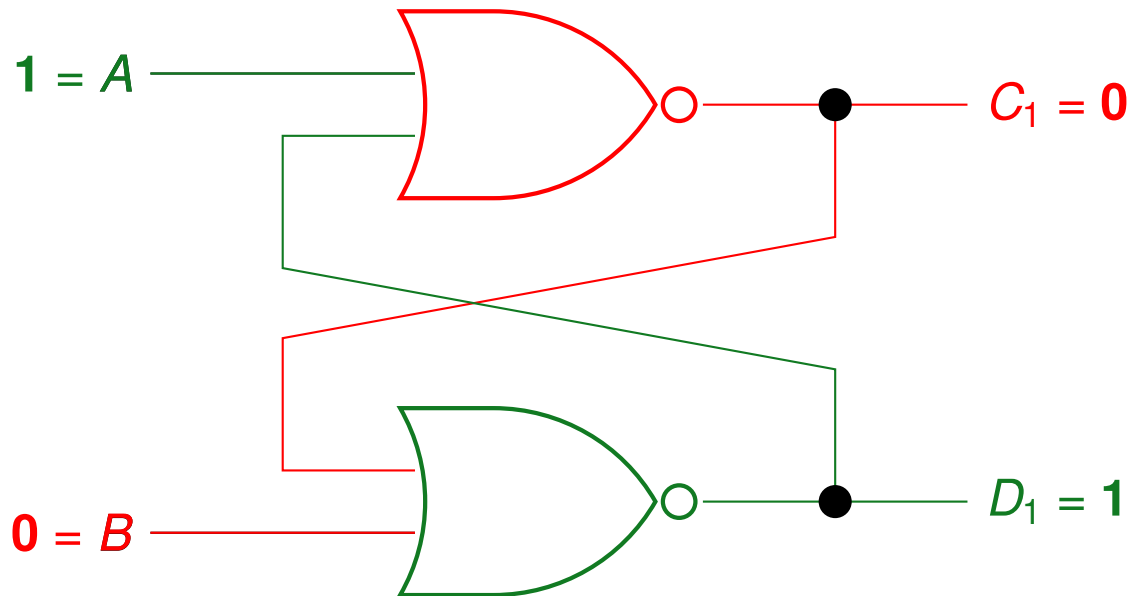
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (0, 0) \mapsto (1, 1) \mapsto (1, 0)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

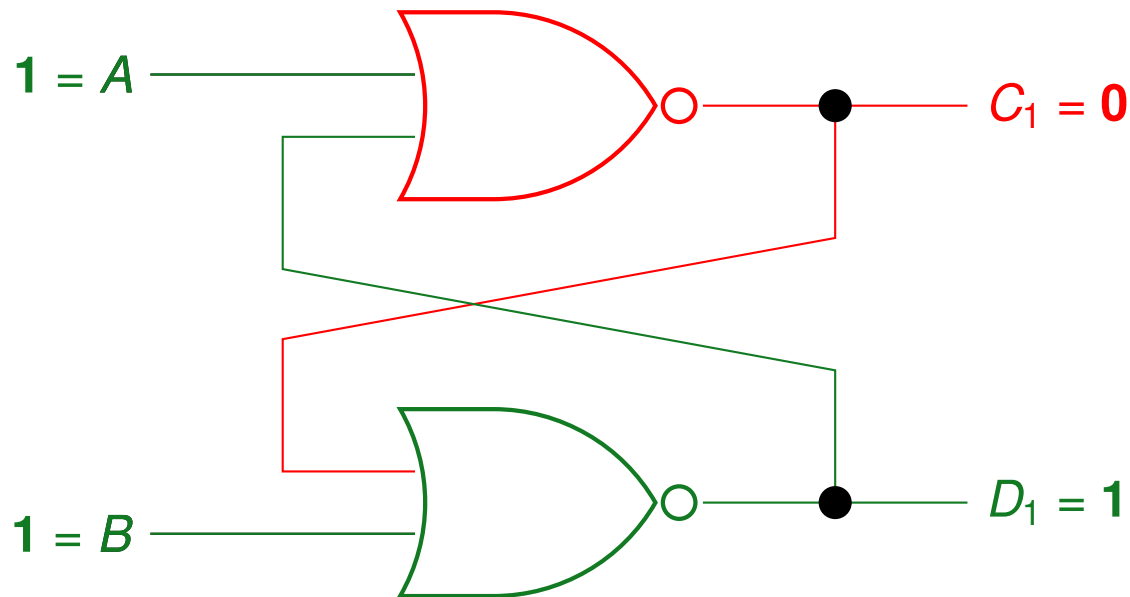
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (0, 0) \mapsto (1, 1) \mapsto (1, 0)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

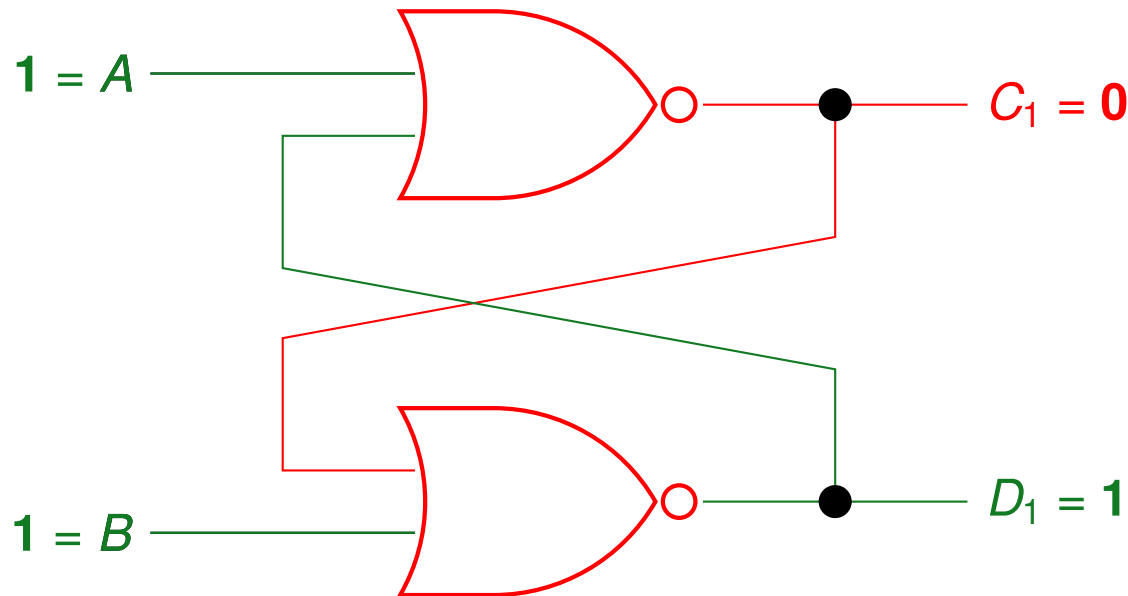
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 1) \mapsto (1, 0) \mapsto (1, 1)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

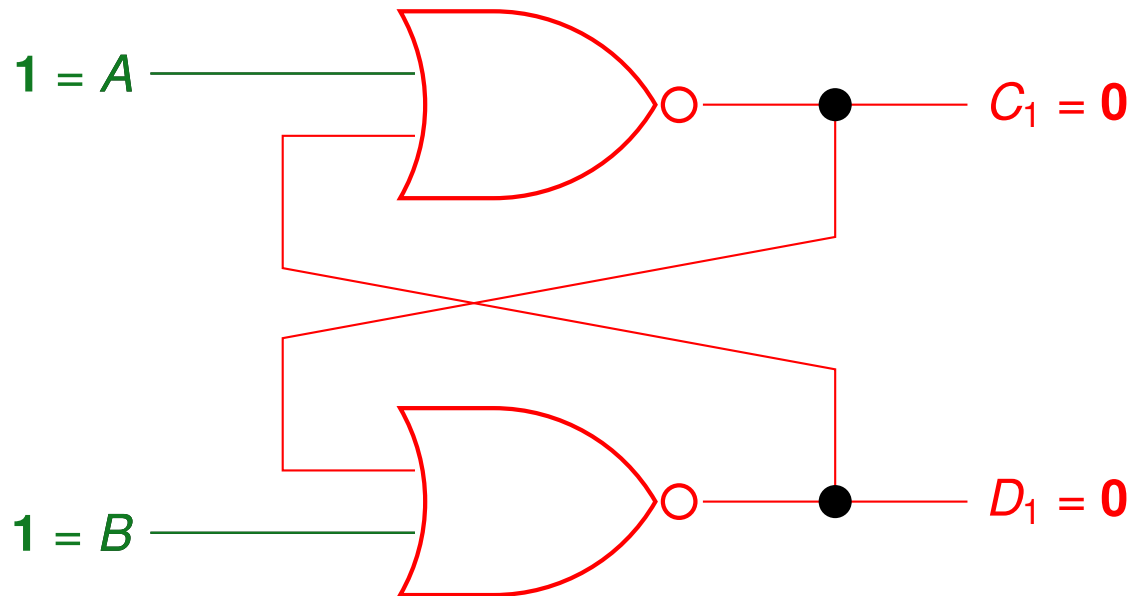
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 1) \mapsto (1, 0) \mapsto (1, 1)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

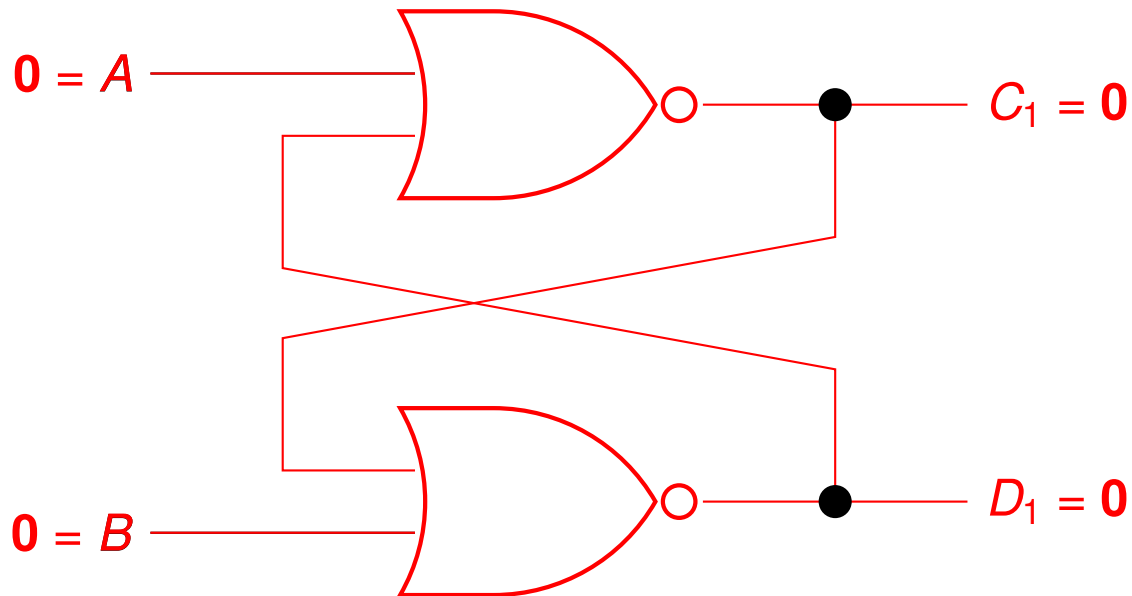
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 1) \mapsto (1, 0) \mapsto (1, 1)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

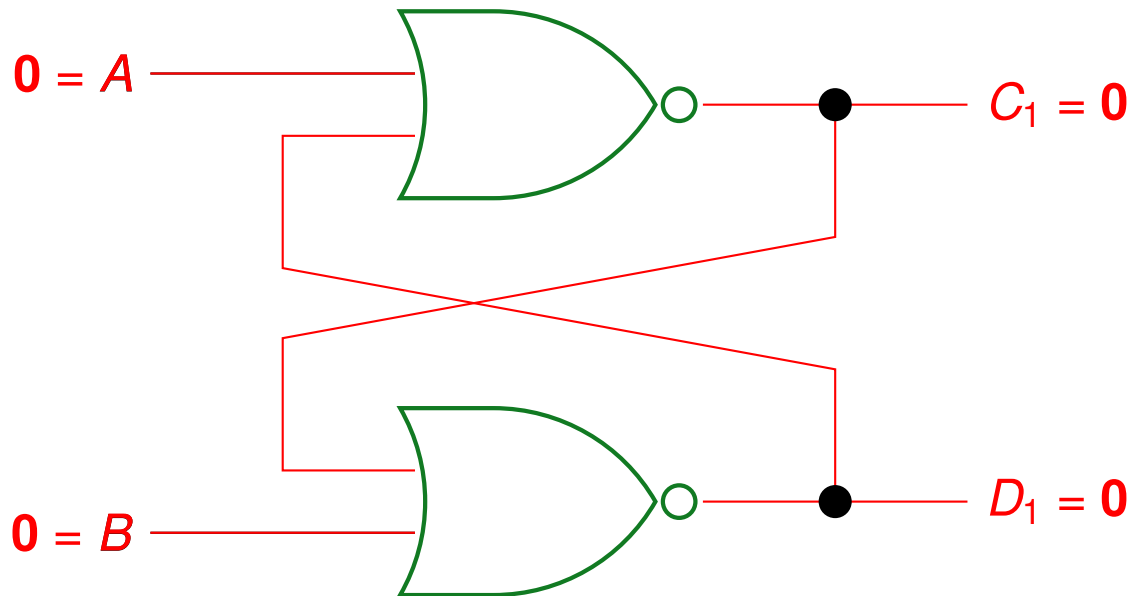
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 0) \mapsto (1, 1) \mapsto (0, 0)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

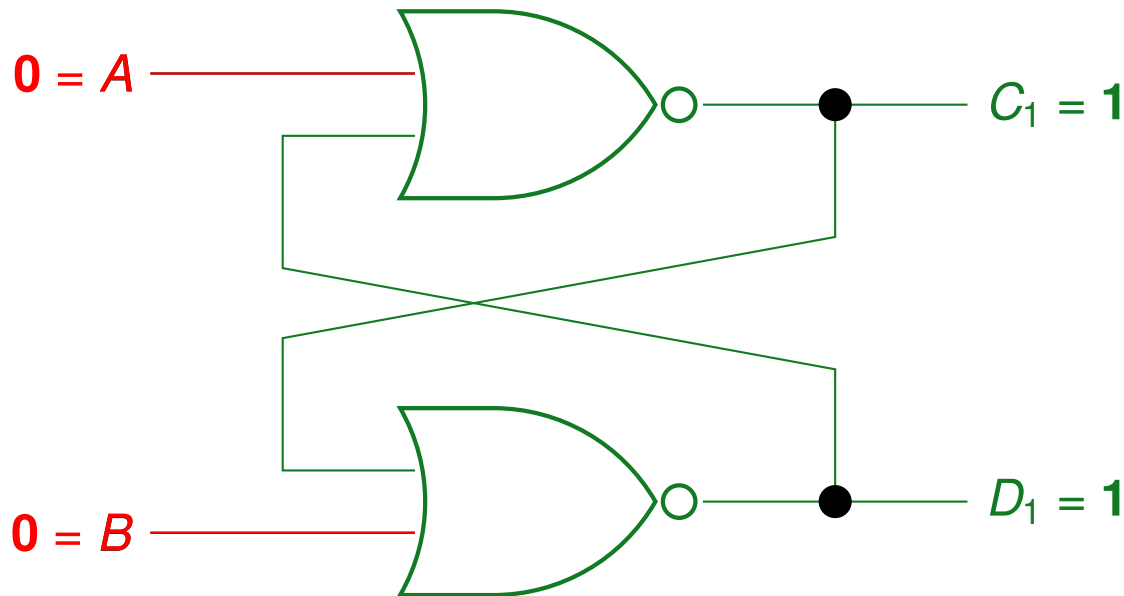
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 0) \mapsto (1, 1) \mapsto (0, 0)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

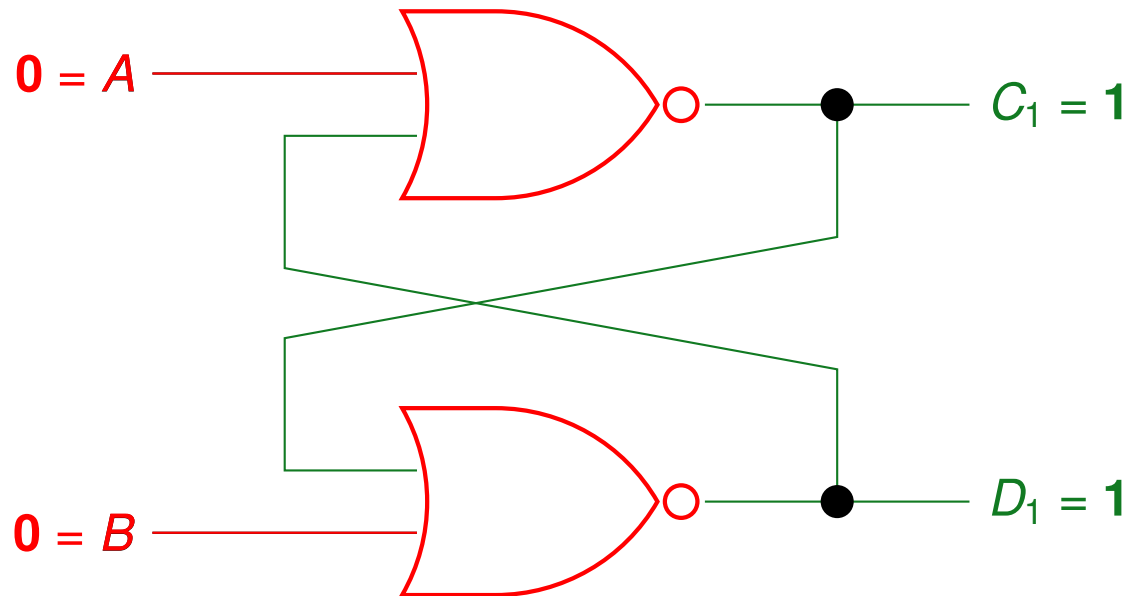
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 0) \mapsto (1, 1) \mapsto (0, 0)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

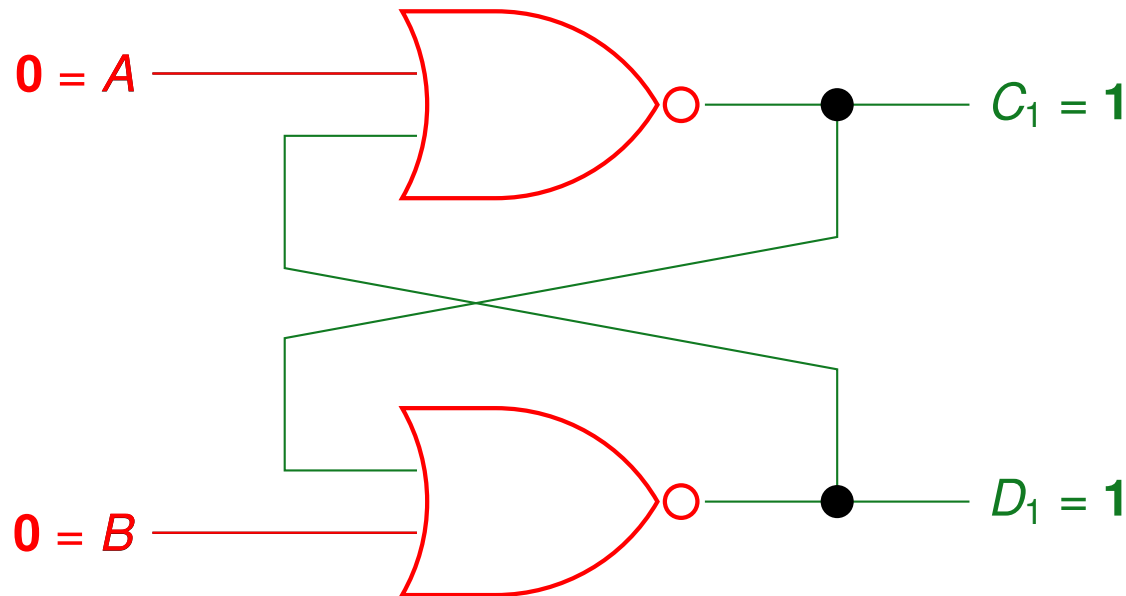
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 0) \mapsto (1, 1) \mapsto (0, 0)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

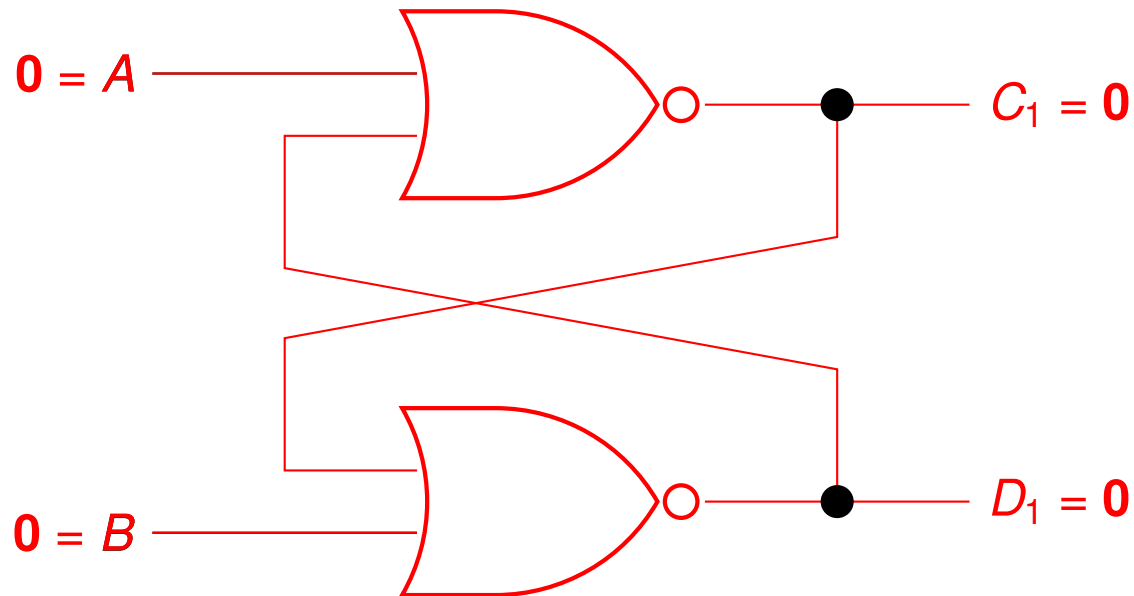
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 0) \mapsto (1, 1) \mapsto (0, 0)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

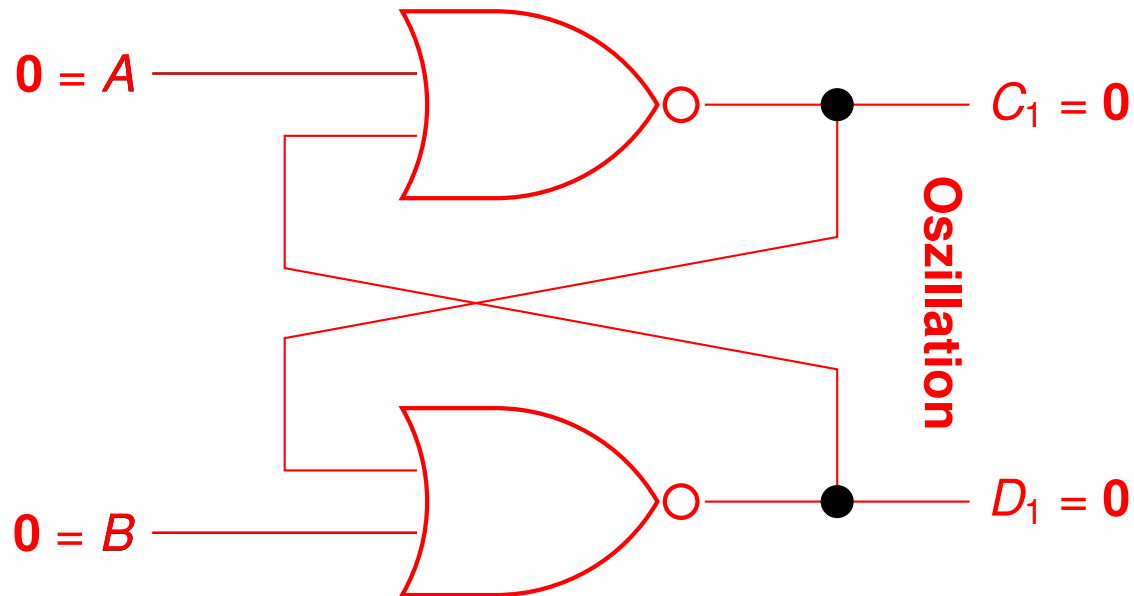
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 0) \mapsto (1, 1) \mapsto (0, 0)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

Aufgabe 4 – Latches und Flipflops

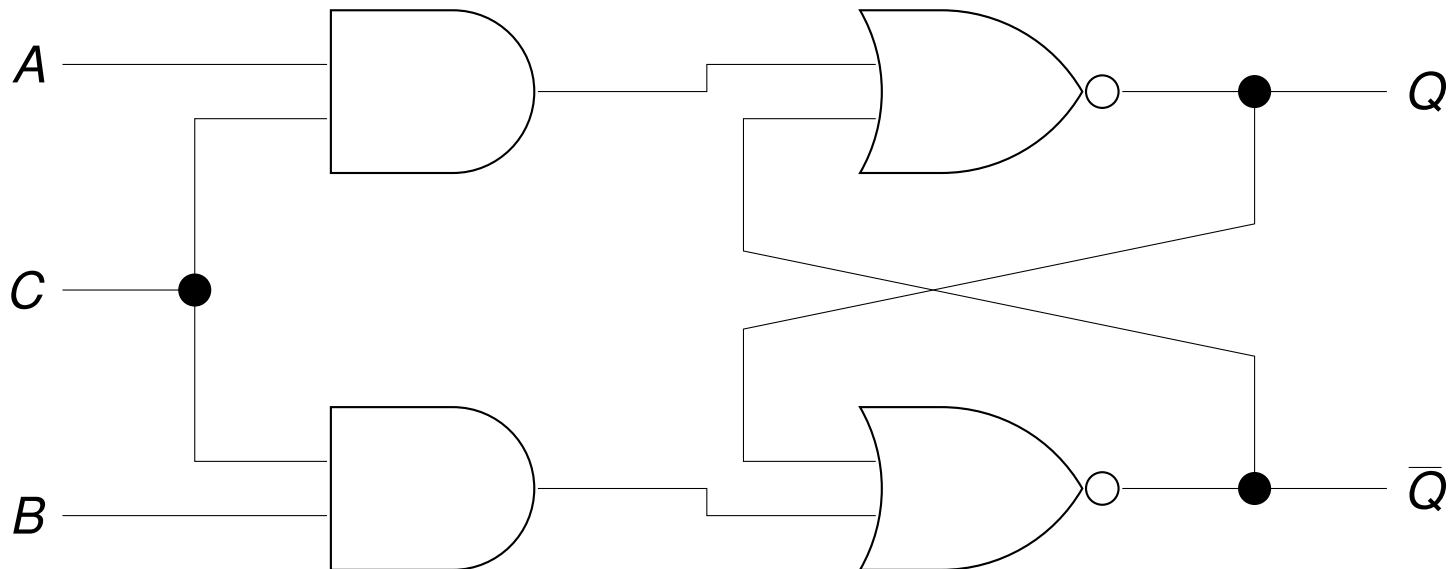
Anliegender Wert (**fett** hervorgehoben): ... $\mapsto (1, 0) \mapsto (1, 1) \mapsto (0, 0)$



Hat ein NOR mindestens **eine** 1 im Eingang, so ist der Ausgang 0.
 Hat ein NOR **keine** 1 im Eingang (sprich: nur Nullen), so ist der Ausgang 1.

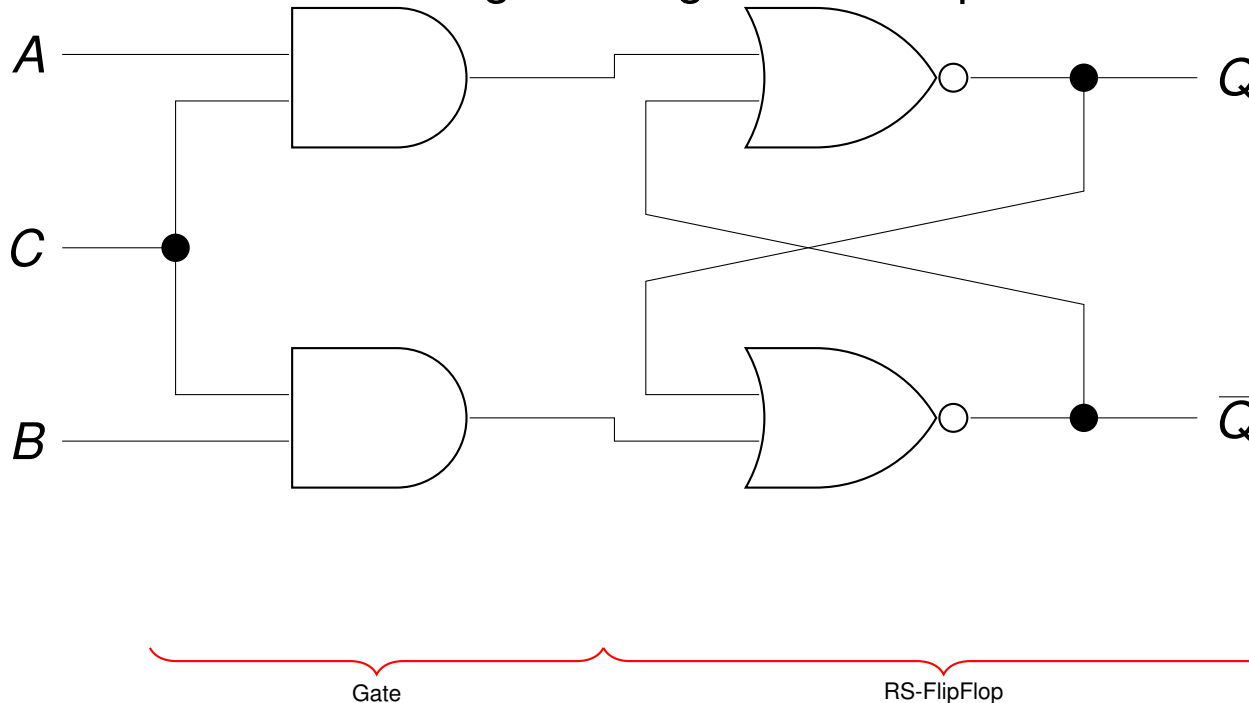
Aufgabe 4 – Latches und Flipflops

b) Sei C ein Taktsignal. Wie bezeichnet man dann das hier abgebildete Speicherelement?



Bestimmung

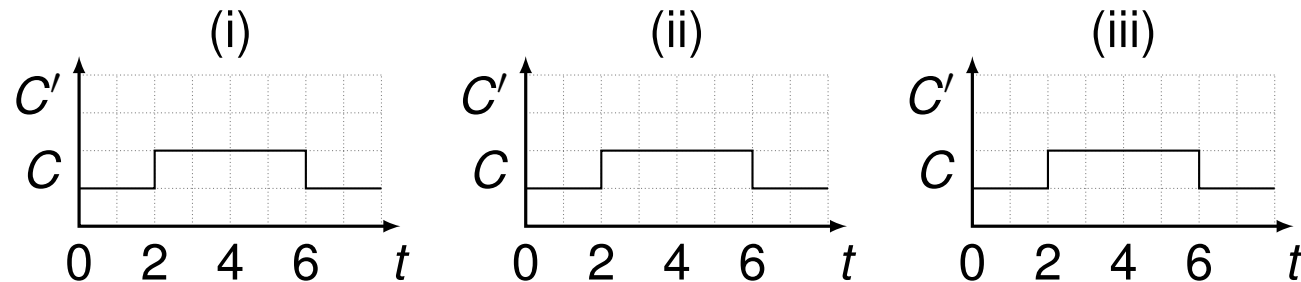
Gesucht: Bezeichnung des abgebildeten Speicherelements.



- $C = 0$: Gate deaktiviert, da an den Ausgängen immer (0,0) anliegt
 - $C = 1$: A und B werden weitergeleitet (UND wegdenken)
- Das Flipflop ist also ein pegelgesteuertes RS-Flipflop.
Außerdem ist es wieder Active-high.

Aufgabe 4 – Latches und Flipflops

b) Dieses soll so erweitert werden, dass es (i) nur bei der steigenden, (ii) der fallenden und (iii) bei jeder Flanke von C auf die Eingänge A und B reagiert. Geben Sie jeweils das Schaltnetz der Flankenerkennung $C \mapsto C'$ an und vervollständigen Sie die folgenden Wellenformdiagramme:



Verzögerungsstrategien

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

Inverter:

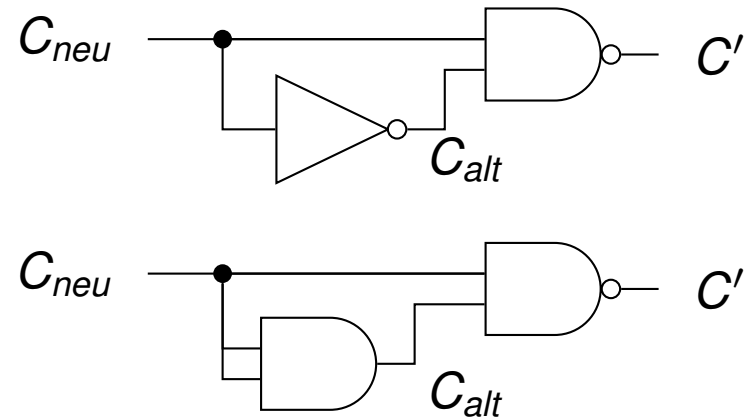
Man erhält ein negiertes C_{alt}

UND/ODER:

Man erhält ein nicht negiertes C_{alt}

Sonstige Gatter:

- NAND/NOR genutzt, da geringere Latenz als UND/ODER
- Andere Gatter möglich



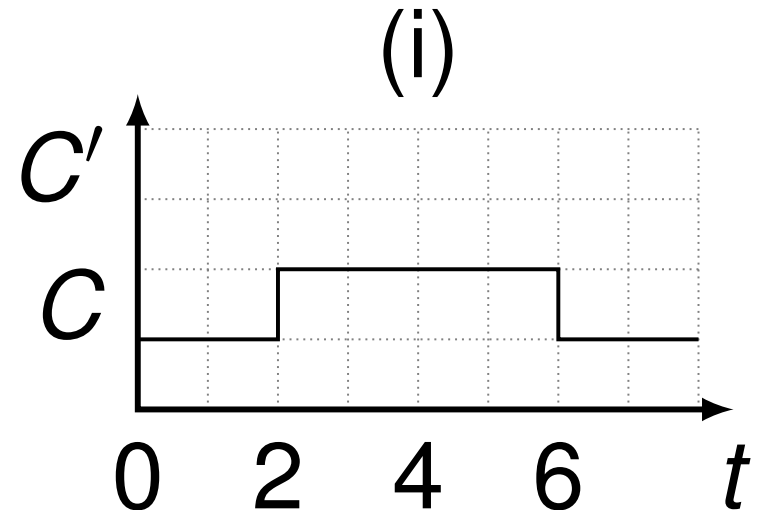
Steigende Flanke I

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

Wir wollen folgendes Verhalten für jeden Zeitschritt umsetzen:

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

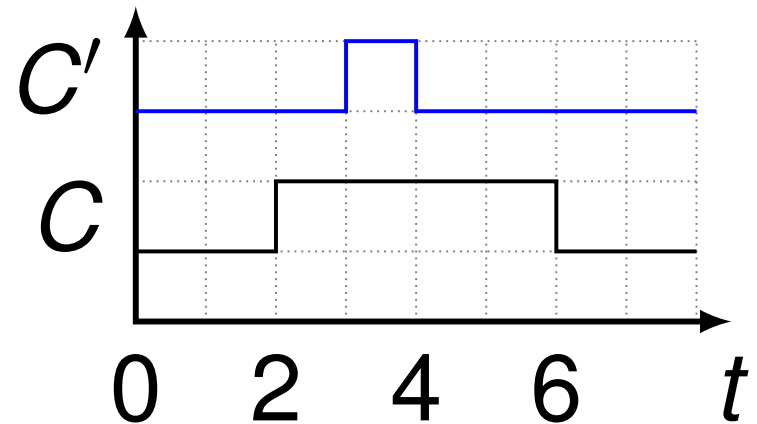
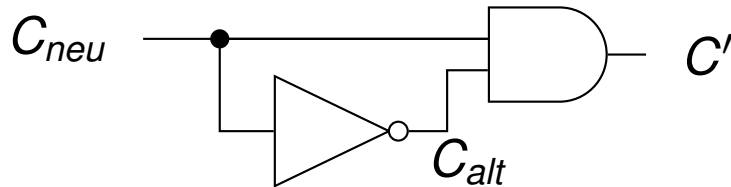
$$f(C_{alt}, C_{neu}) = \overline{C_{alt}} \cdot C_{neu}$$



Steigende Flanke II

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.

Fallende Flanke I

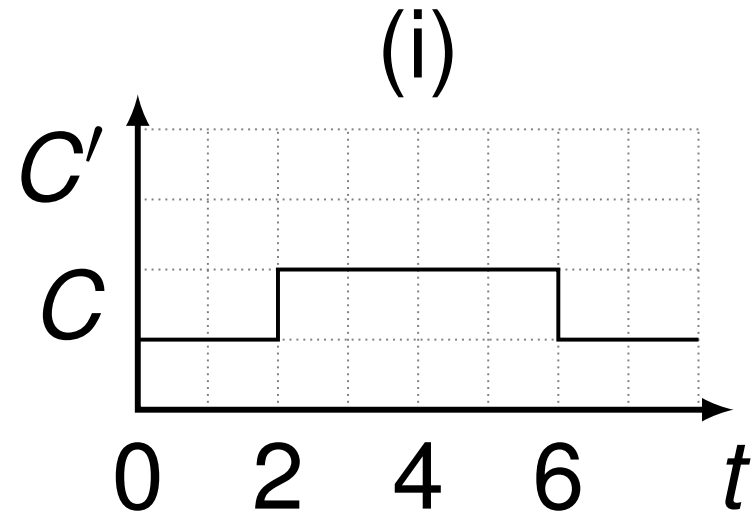
Wir wissen, dass die Verzögerungszeit $\tau = 1$ beträgt.

Wir wollen folgendes Verhalten für jeden Zeitschritt umsetzen:

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

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

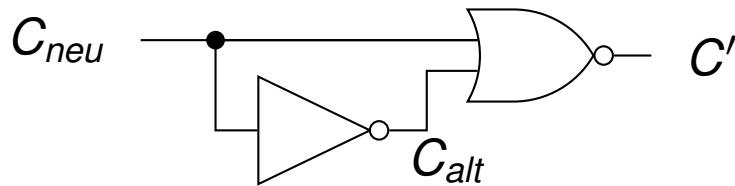
$$f(C_{alt}, C_{neu}) = \overline{C_{neu}} \cdot C_{alt} = \overline{\overline{C_{alt}}} \cdot C_{neu}$$



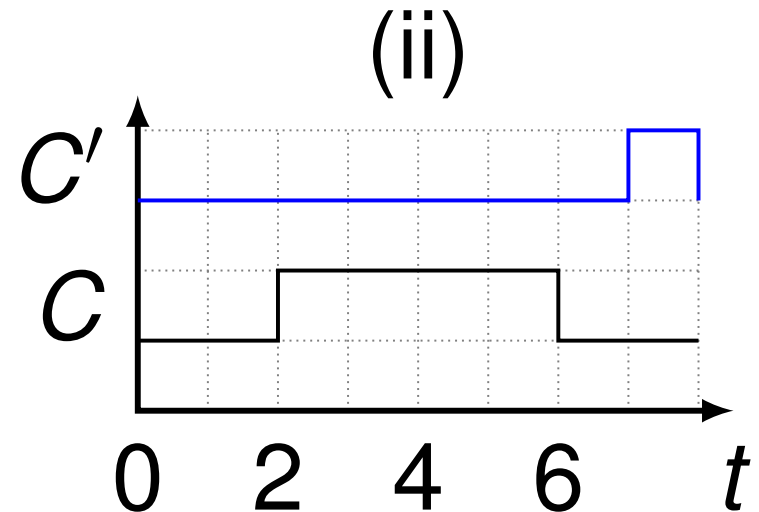
Fallende Flanke II

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

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



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



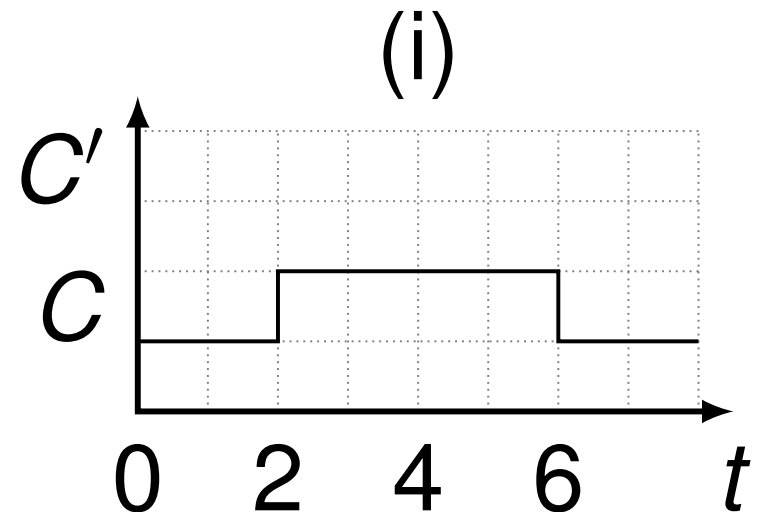
Steigende und Fallende Flanke I

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

Wir wollen folgendes Verhalten für jeden Zeitschritt umsetzen:

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

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



Steigende und Fallende Flanke I

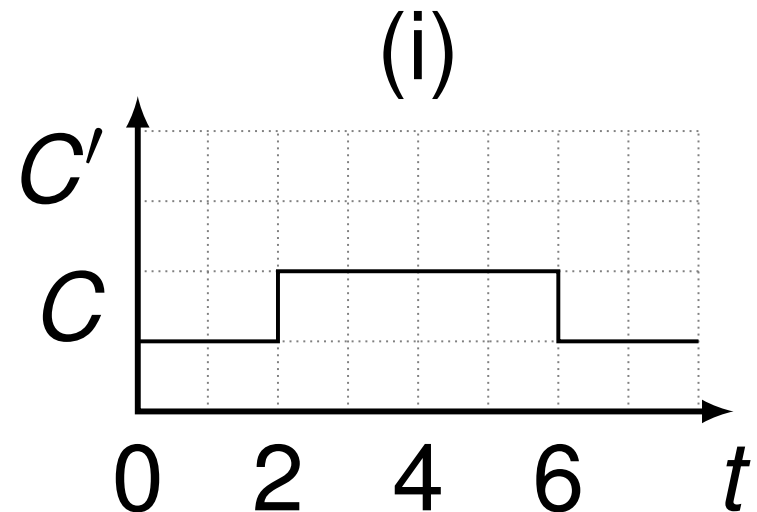
Wir wissen, dass die Verzögerungszeit $\tau = 1$ beträgt.

Wir wollen folgendes Verhalten für jeden Zeitschritt umsetzen:

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

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

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

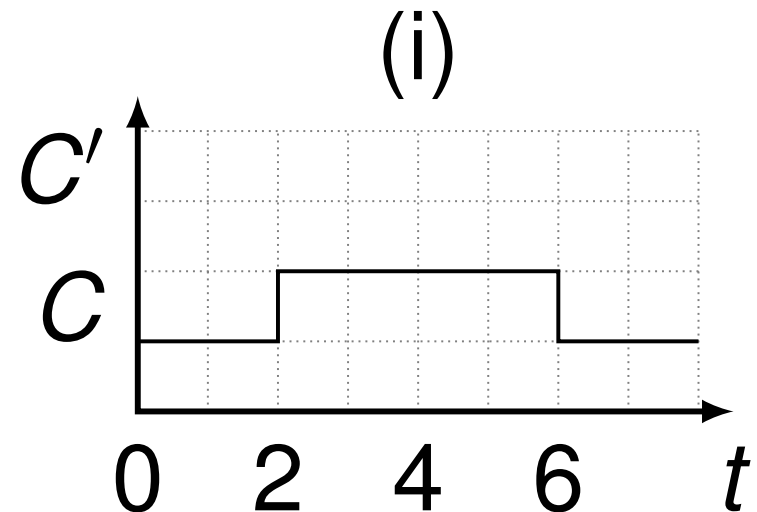


Steigende und Fallende Flanke I

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

Wir wollen folgendes Verhalten für jeden Zeitschritt umsetzen:

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

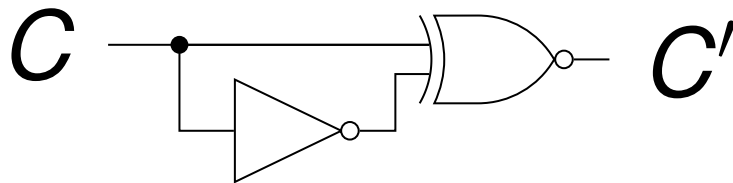


$$\begin{aligned}
 f(C_{alt}, C_{neu}) &= \overline{C_{neu}} \cdot C_{alt} + \overline{C_{alt}} \cdot C_{neu} \\
 &= (\overline{C_{neu}} + \overline{C_{alt}}) \cdot (C_{alt} + C_{neu}) \\
 &= \overline{C_{alt} C_{neu}} + C_{alt} C_{neu} = \overline{C_{alt} \oplus C_{neu}}
 \end{aligned}$$

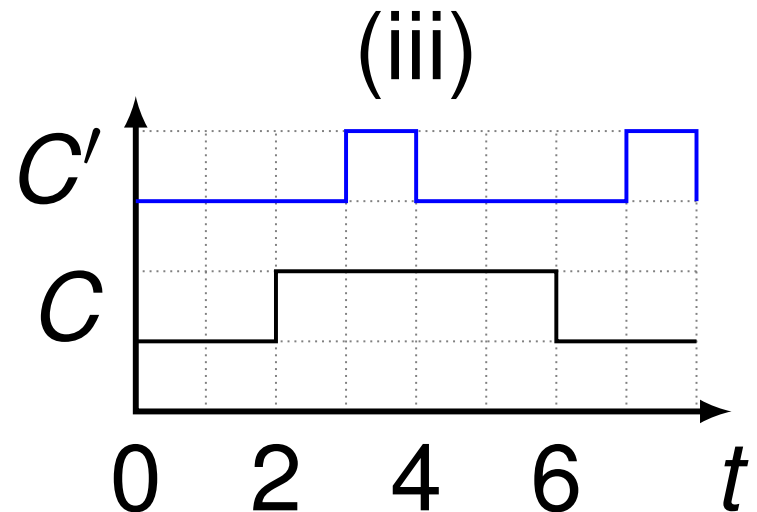
Steigende und Fallende Flanke II

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

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

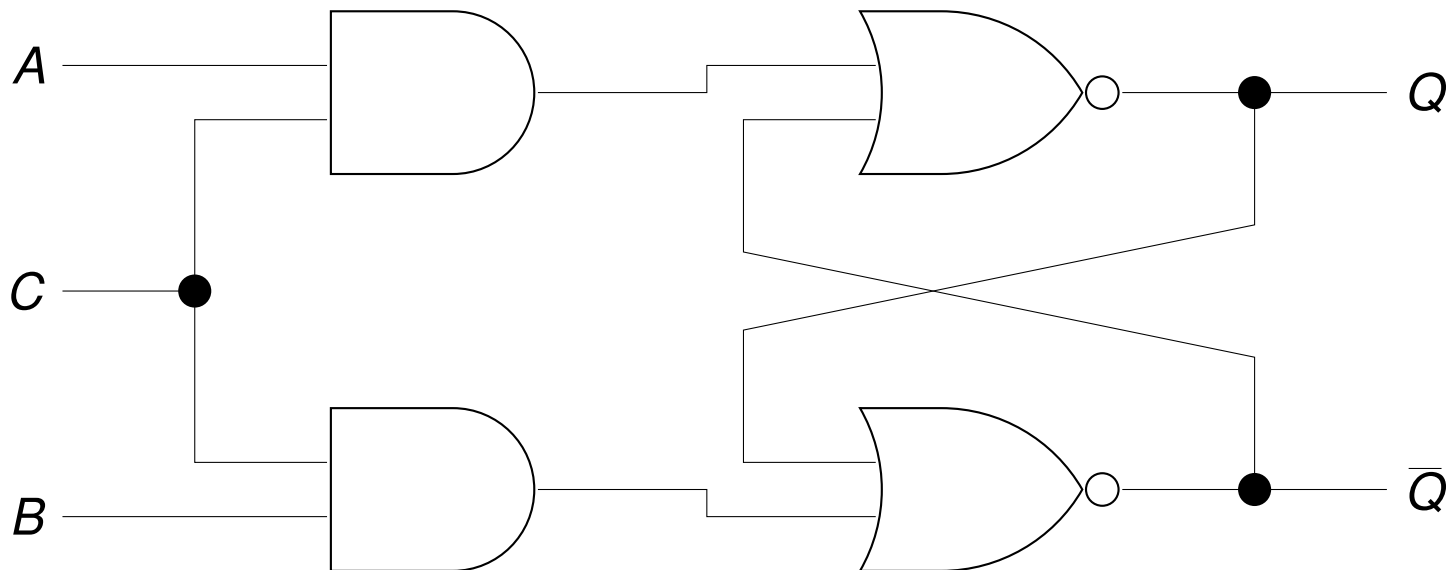


Das NOR-Gatter verzögert das C' um eine Zeiteinheit.



Aufgabe 4 – Latches und Flipflops

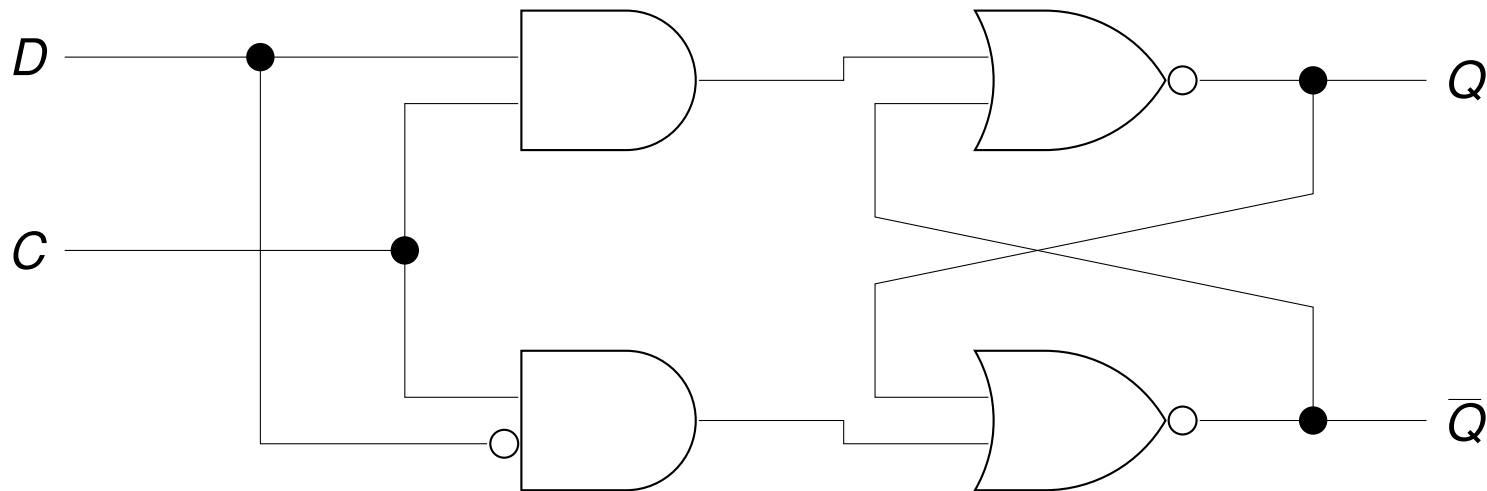
c) Erweitern Sie nun die Schaltung aus Teilaufgabe b) dahingehend, dass keine undefinierten Zustände, wie sie in Aufgabe a) der Fall waren, mehr auftreten.



Zum Beispiel: D-Flipflop

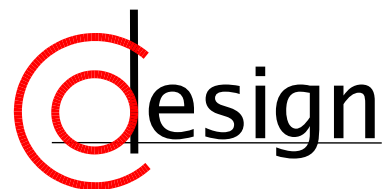
Gesucht: Elimination der ungültigen Zustände

Lösung: Nur noch eine Datenleitung (D-Flipflop)



Der zweite Eingang ist stets die Negation des ersten, weshalb der ungültige Zustand nicht auftreten kann.

Aufgabe 1 – Master-Slave-Flipflops

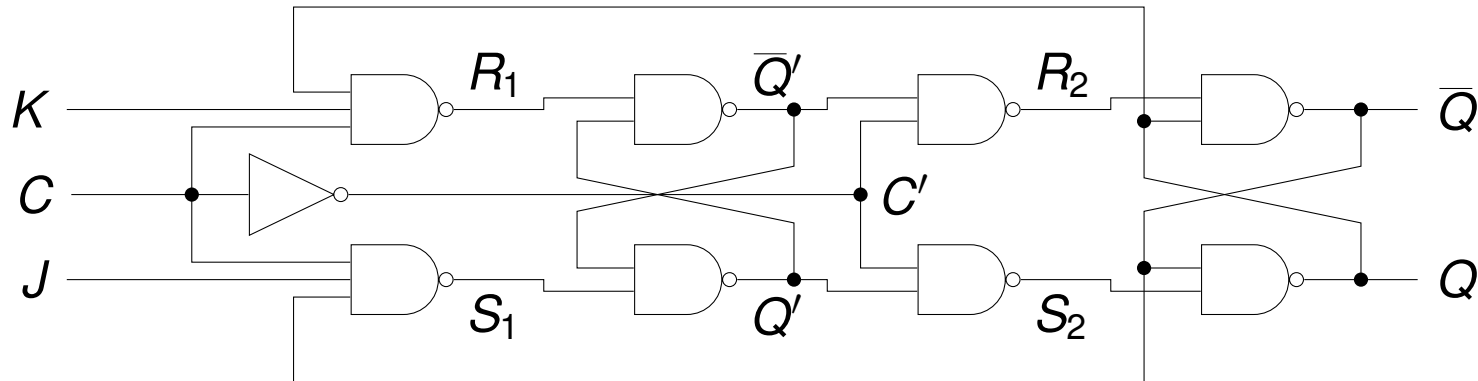


Aufgabe 1 – Master-Slave-Flipflops

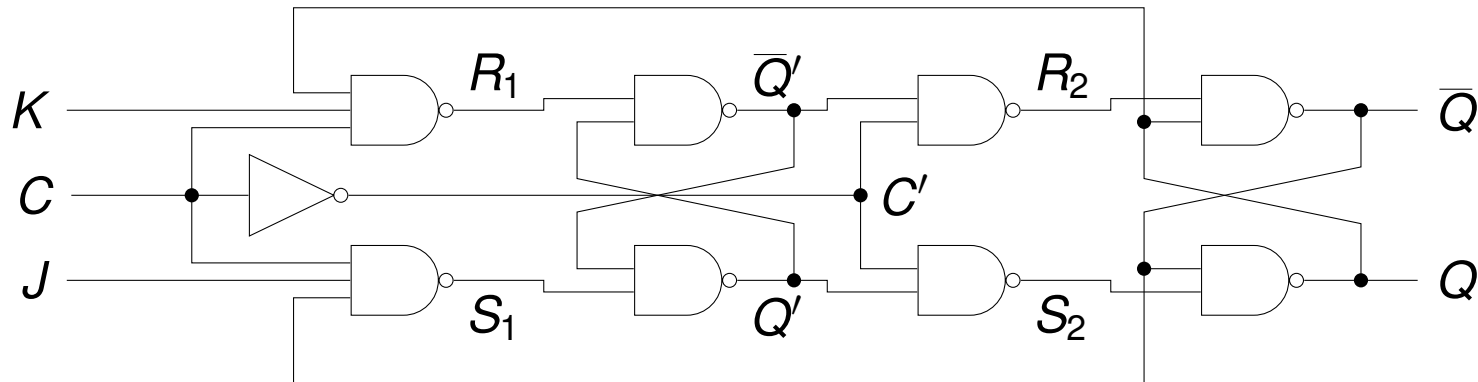
- a) Welches Problem pegelgesteuerter JK-Flipflops löst das gegebene JK-Master-Slave-Flipflop?

Aufgabe 1 – Master-Slave-Flipflops

a) Welches Problem pegelgesteuerter JK-Flipflops löst das gegebene JK-Master-Slave-Flipflop?



Aufgabe 1 – Master-Slave-Flipflops

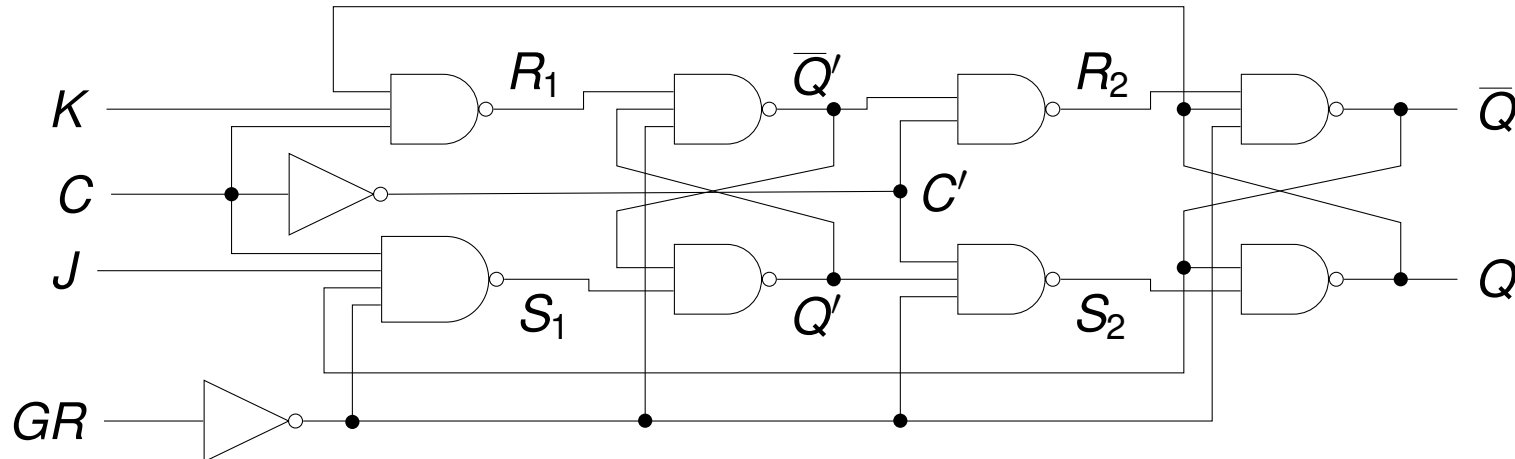


- b) Erweitern Sie das gegebene JK-Master-Slave-Flipflop um ein Reset-Signal GR , das unabhängig von allen anderen Signalen das Flipflop asynchron zurücksetzt.

Asynchrones Reset I

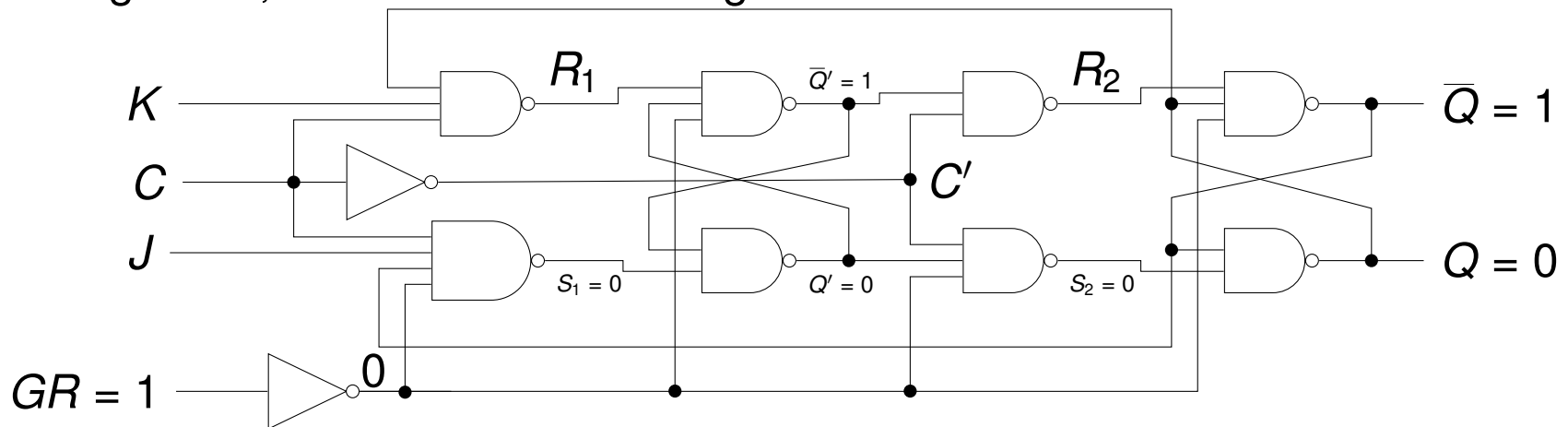
Zwei Aufgaben:

- Zurücksetzen des Master- und des Slave-Flipflop unabhängig der Werte K , R_1 und R_2 .
- Setzen von S_1 und S_2 auf inaktiv, um den Toggle-Zustand zu verhindern.



Asynchrones Reset II

Ist $GR = 1$, wird deshalb durch den Inverter jeweils eine 0 an S_1 und S_2 angelegt und die beiden Gatter auf 1 gesetzt. Analog werden auch \bar{Q}' und \bar{Q} auf 1 gesetzt, also Q und Q' zurückgesetzt.



Aufgabe 1 – Master-Slave-Flipflops

- c) Welches Problem ergibt sich für dieses JK-Master-Slave-Flipflop, wenn sich während der Einsphase des Taktes C die Eingänge J und K ändern?

Aufgabe 2 – Multiplexer und Demultiplexer



Aufgabe 2 – Multiplexer und Demultiplexer

- a) Entwerfen Sie eine digitale Schaltung, die bei einer 0 am Steuereingang s den Wert des Eingangs x_0 , bei einer 1 am Steuereingang s den Wert des Eingangs x_1 am Ausgang y erzeugt:

s	x_0	x_1	y
0	0	-	0
0	1	-	1
1	-	0	0
1	-	1	1

Aufgabe 2 – Multiplexer und Demultiplexer

- b) Die Schaltung soll nun so erweitert werden, dass wahlweise genau einer von vier Eingängen $x_0 \dots x_3$ am Ausgang y erscheint.

Aufgabe 2 – Multiplexer und Demultiplexer

- b) Die Schaltung soll nun so erweitert werden, dass wahlweise genau einer von vier Eingängen $x_0 \dots x_3$ am Ausgang y erscheint.
- Welche Auswirkungen hat dies für den Steuereingang s ?

Aufgabe 2 – Multiplexer und Demultiplexer

b) Die Schaltung soll nun so erweitert werden, dass wahlweise genau einer von vier Eingängen $x_0 \dots x_3$ am Ausgang y erscheint.

- Realisieren Sie die Schaltung mit Und-/Oder-Gattern sowie Invertern.

Aufgabe 2 – Multiplexer und Demultiplexer

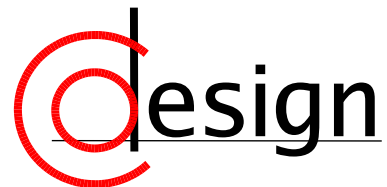
- b) Die Schaltung soll nun so erweitert werden, dass wahlweise genau einer von vier Eingängen $x_0 \dots x_3$ am Ausgang y erscheint.
- Welche Auswirkungen hat dies für den Steuereingang s ?
 - Realisieren Sie die Schaltung mit Und-/Oder-Gattern sowie Invertern.

Aufgabe 2 – Multiplexer und Demultiplexer

c) Schließlich soll eine Demultiplexer-Schaltung entworfen werden. Sie besitzt einen Eingang x , der in Abhängigkeit des Steuereingangs s den Wert von x an einem der vier Ausgänge $y_0 \dots y_3$ erzeugt (siehe folgende Wahrheitstabelle).

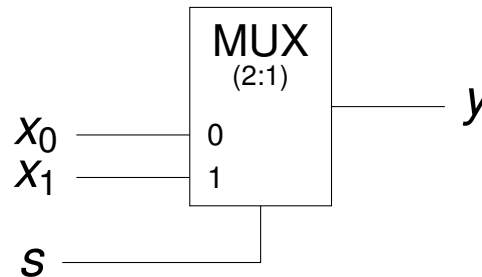
$s_1 s_0$	x	y_0	y_1	y_2	y_3
00	$0/1$	$0/1$	0	0	0
01	$0/1$	0	$0/1$	0	0
10	$0/1$	0	0	$0/1$	0
11	$0/1$	0	0	0	$0/1$

Aufgabe 3 – Barrel-Shifter

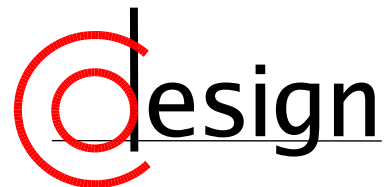


Aufgabe 3 – Barrel-Shifter

Erstellen Sie aus 2:1-Multiplexern, wie im nebenstehenden Blockschaltbild dargestellt, einen Barrel-Shifter, der den Eingang $A = (a_3, a_2, a_1, a_0)$ zyklisch um $N = 0 \dots 3$ Stellen nach links verschiebt und auf dem Ausgang $B = (b_3, b_2, b_1, b_0)$ ausgibt. Ist zum Beispiel $N = 1$, so soll $b_3 = a_2, b_2 = a_1, b_1 = a_0, b_0 = a_3$ gelten.

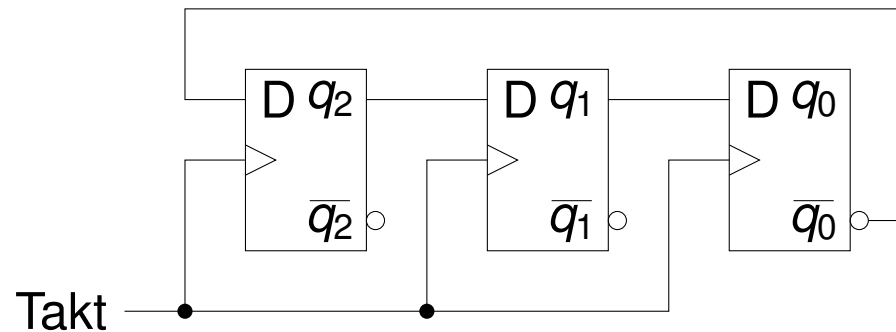


Aufgabe 4 – Schieberegister



Aufgabe 4 – Schieberegister

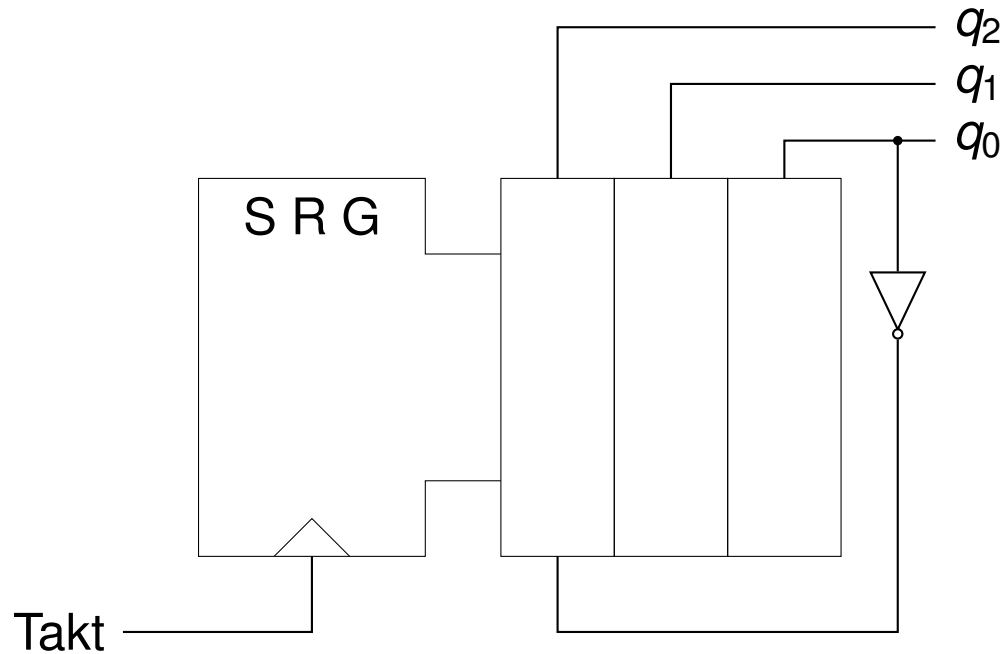
Das folgende Bild zeigt einen sogenannten Johnson-Zähler, bei dem das invertierte Signal q_0 des letzten Flipflops in der Kette an den Eingang D des ersten angeschlossen wird.



- a) Der aktuelle Wert der Speicherzellen sei 000. Stellen Sie die Folge der Speicherinhalte als gerichteten Graphen dar und bestimmen Sie die Anzahl N der durchlaufenen Zustände.

Aufgabe 4 – Schieberegister: Johnson-Zähler

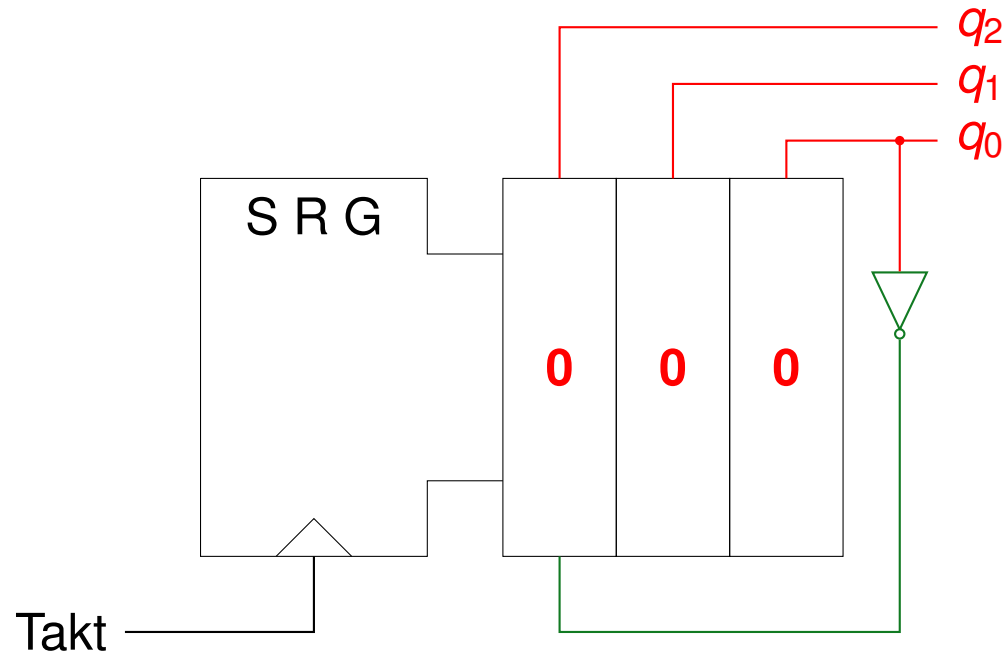
Startwert: 000



000

Aufgabe 4 – Schieberegister: Johnson-Zähler

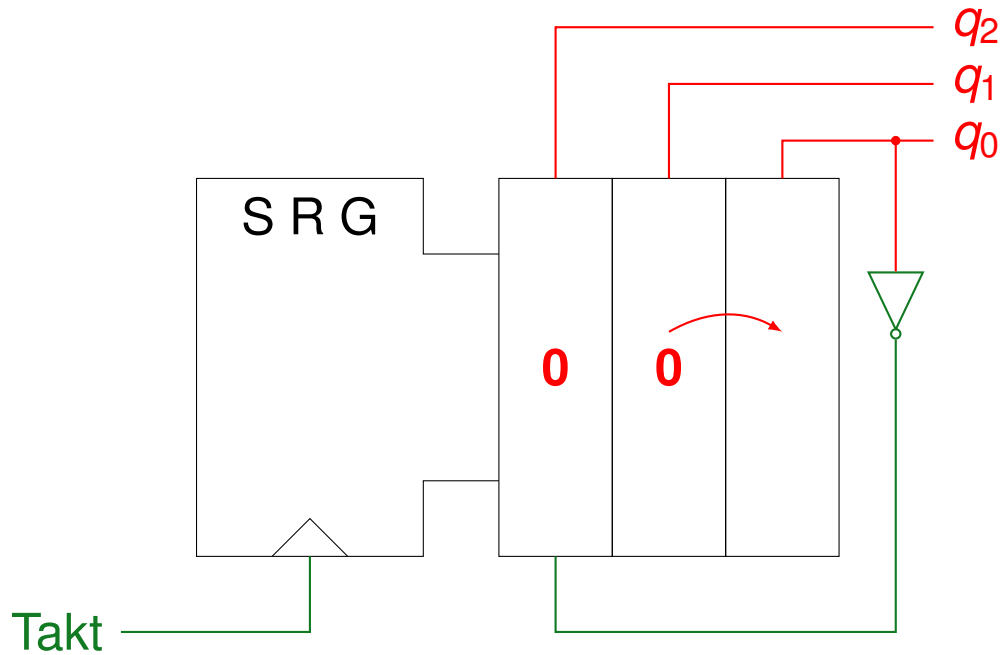
Startwert: 000



000

Aufgabe 4 – Schieberegister: Johnson-Zähler

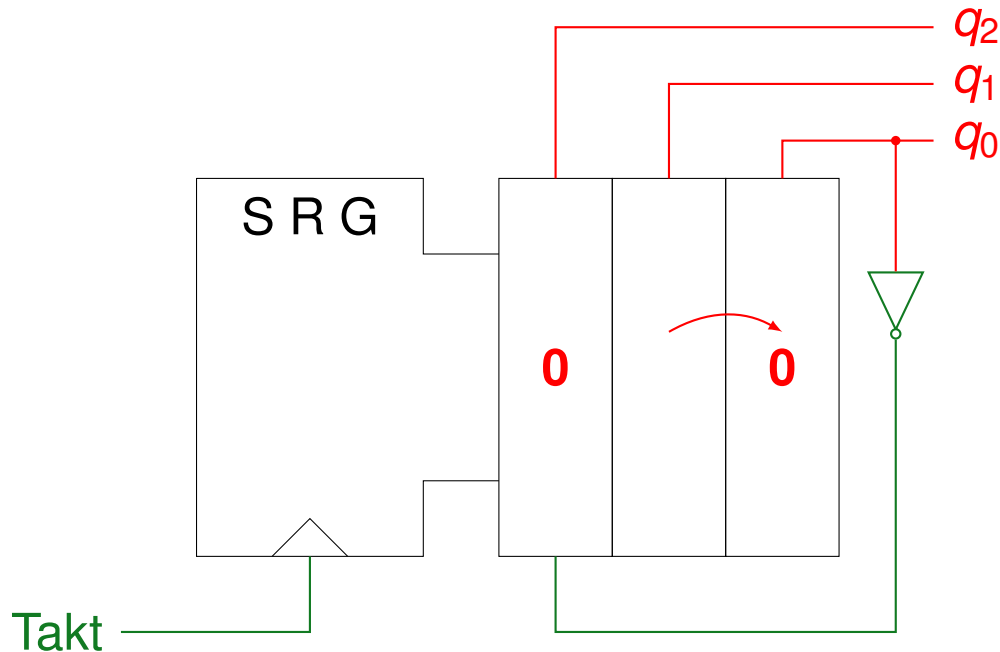
Startwert: 000



000

Aufgabe 4 – Schieberegister: Johnson-Zähler

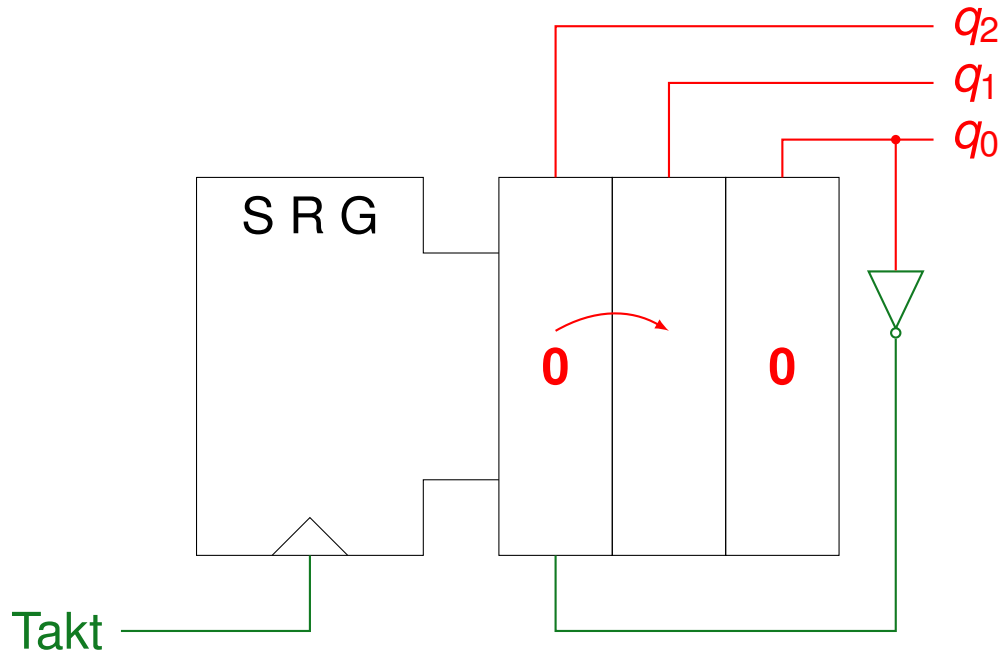
Startwert: 000



000

Aufgabe 4 – Schieberegister: Johnson-Zähler

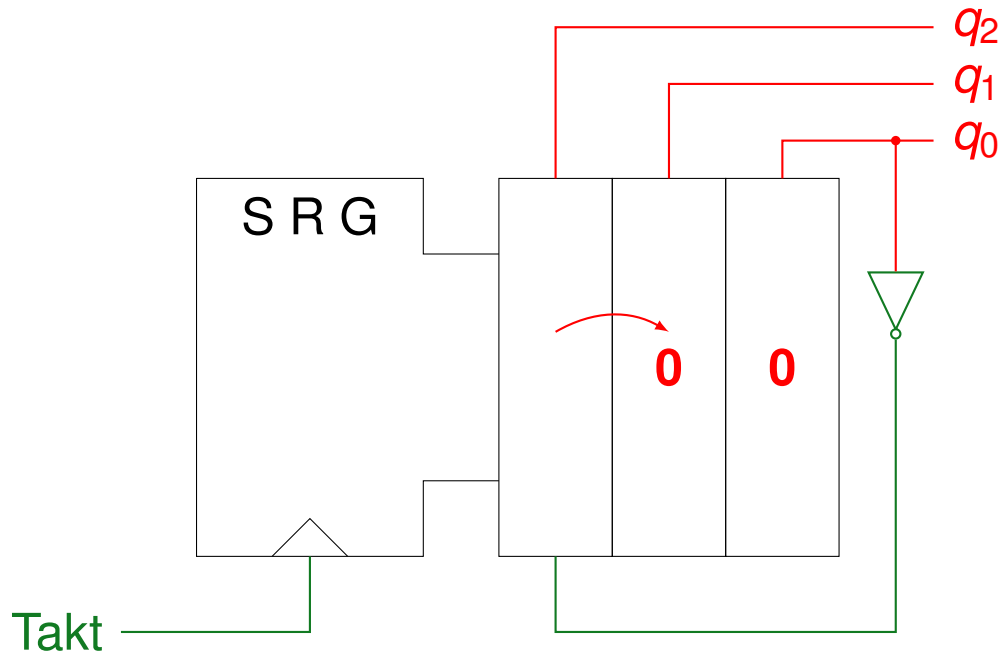
Startwert: 000



000

Aufgabe 4 – Schieberegister: Johnson-Zähler

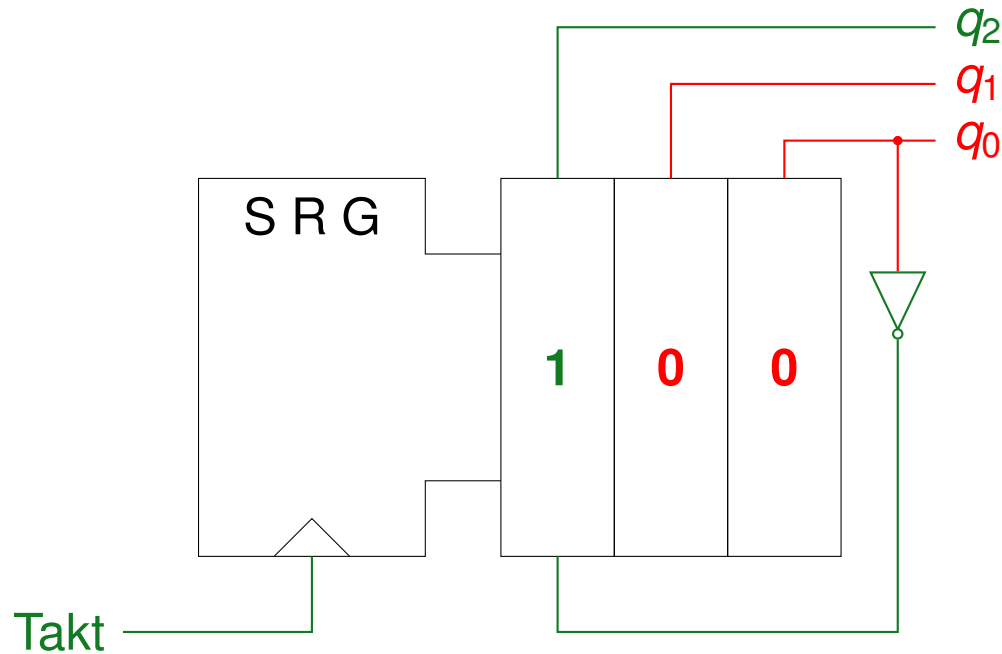
Startwert: 000



000

Aufgabe 4 – Schieberegister: Johnson-Zähler

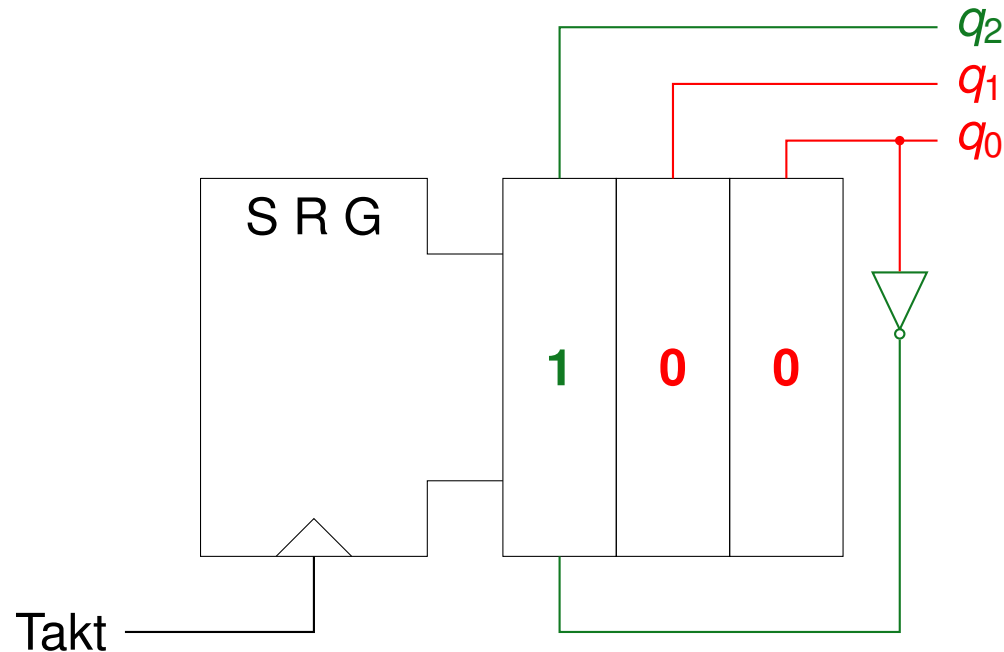
Startwert: 000



000

Aufgabe 4 – Schieberegister: Johnson-Zähler

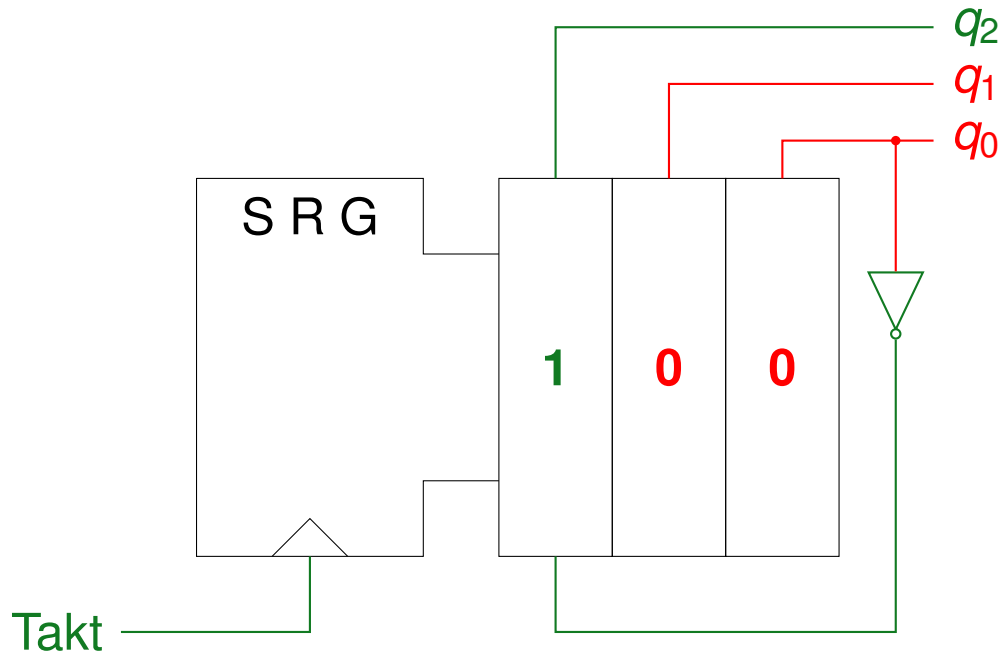
Startwert: 000



000 \mapsto 100

Aufgabe 4 – Schieberegister: Johnson-Zähler

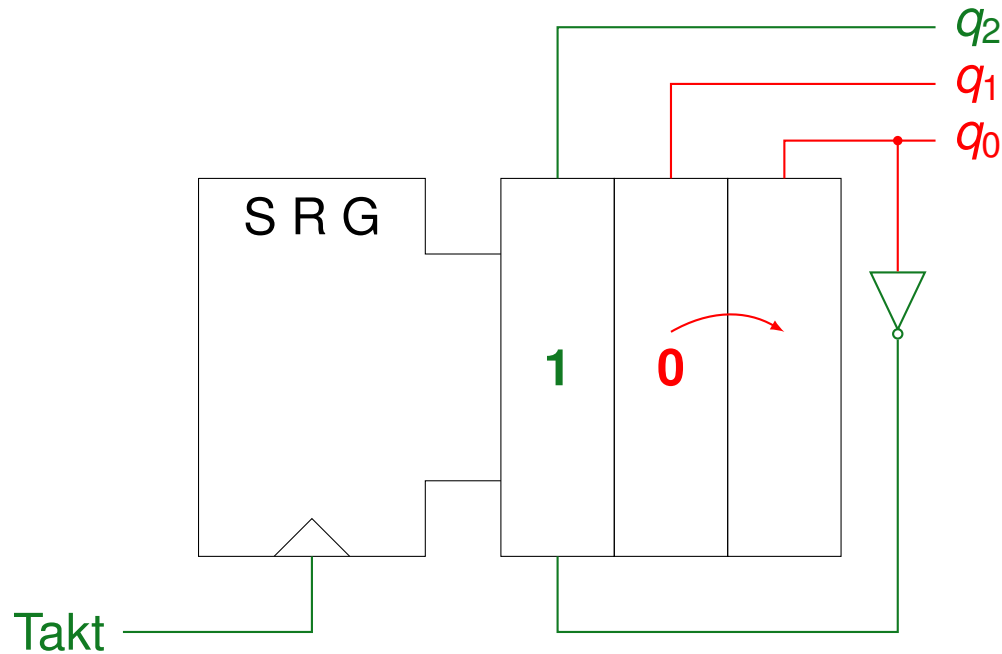
Startwert: 000



000 \mapsto 100

Aufgabe 4 – Schieberegister: Johnson-Zähler

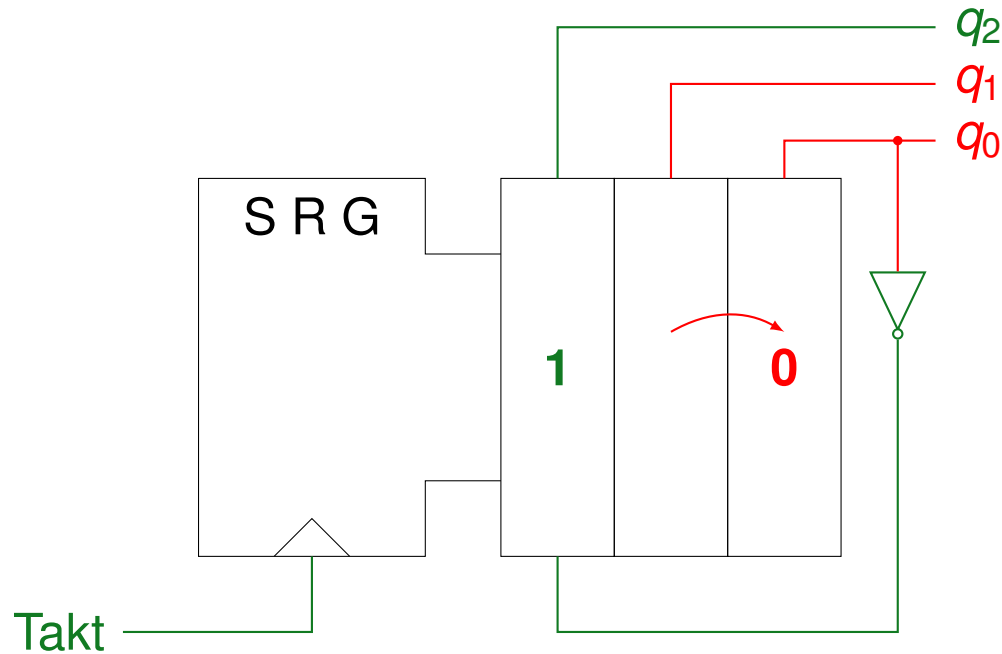
Startwert: 000



000 \mapsto 100

Aufgabe 4 – Schieberegister: Johnson-Zähler

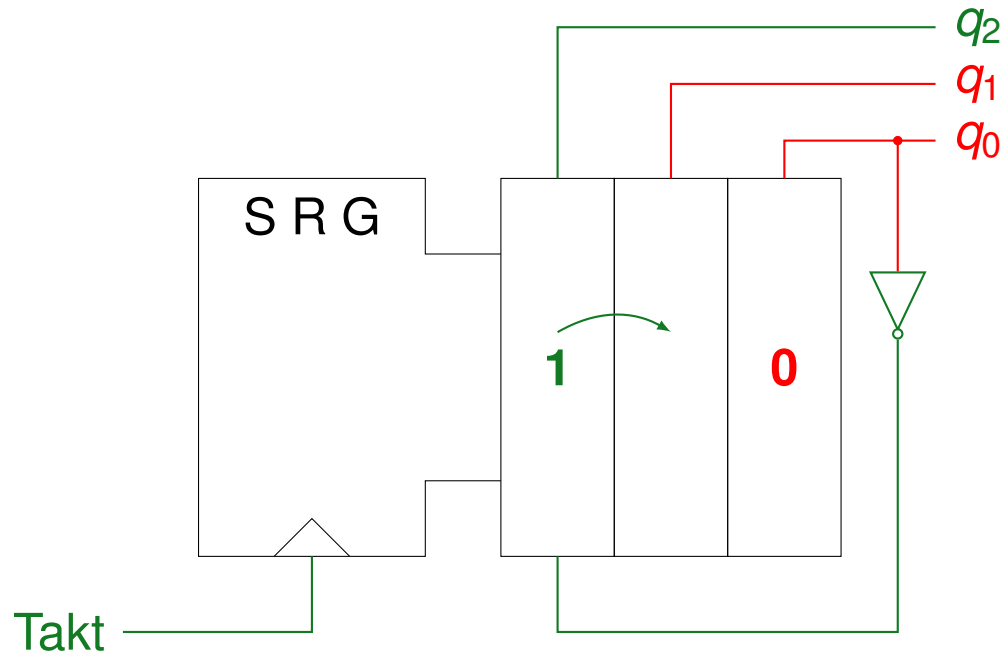
Startwert: 000



000 \mapsto 100

Aufgabe 4 – Schieberegister: Johnson-Zähler

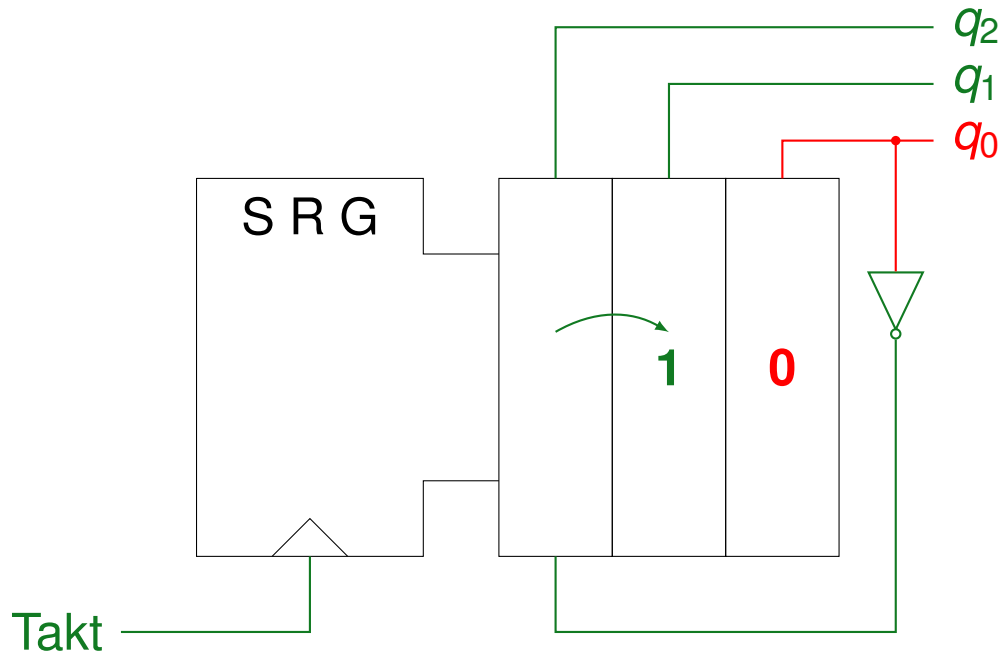
Startwert: 000



000 \mapsto 100

Aufgabe 4 – Schieberegister: Johnson-Zähler

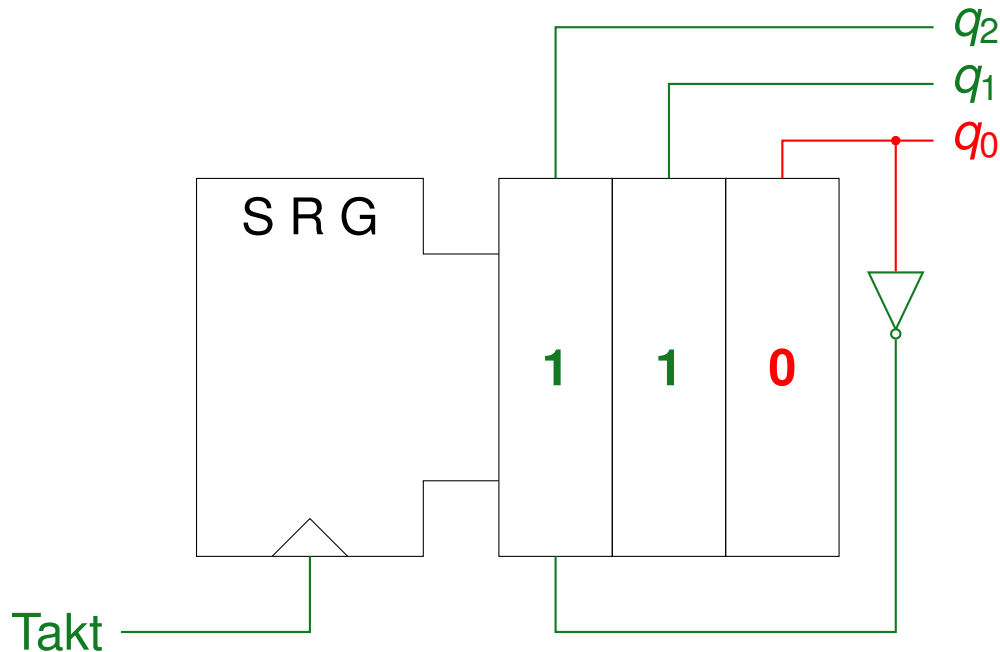
Startwert: 000



000 \mapsto 100

Aufgabe 4 – Schieberegister: Johnson-Zähler

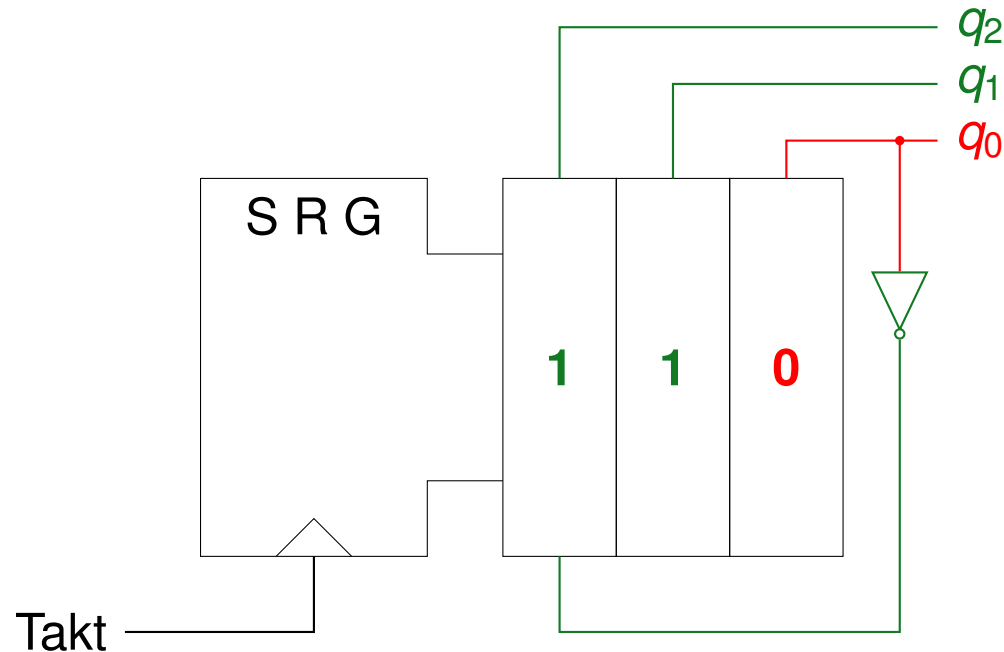
Startwert: 000



000 \mapsto 100

Aufgabe 4 – Schieberegister: Johnson-Zähler

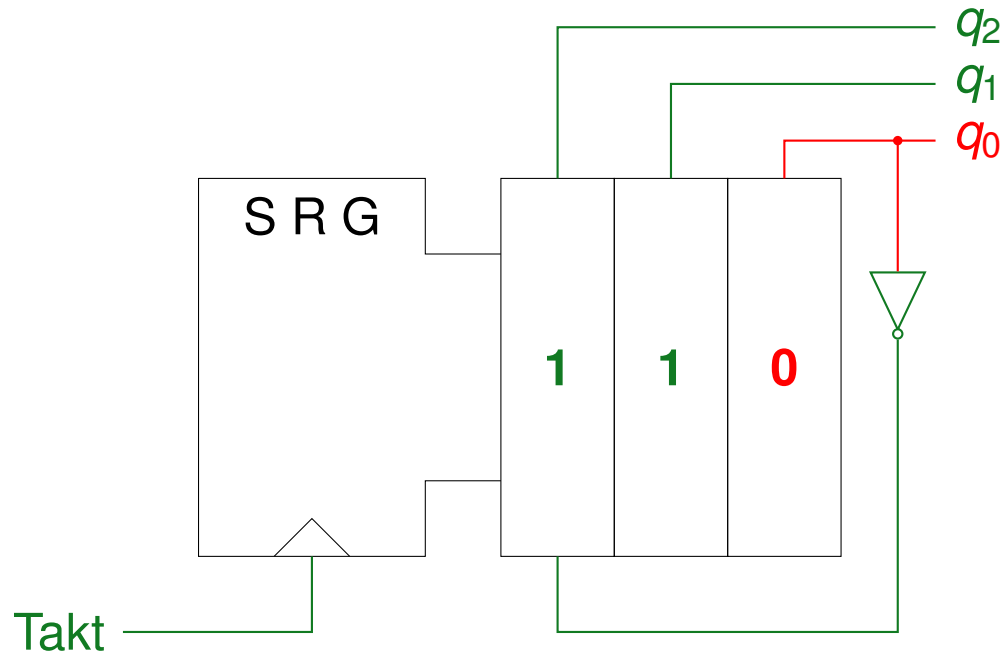
Startwert: 000



000 \mapsto 100 \mapsto 110

Aufgabe 4 – Schieberegister: Johnson-Zähler

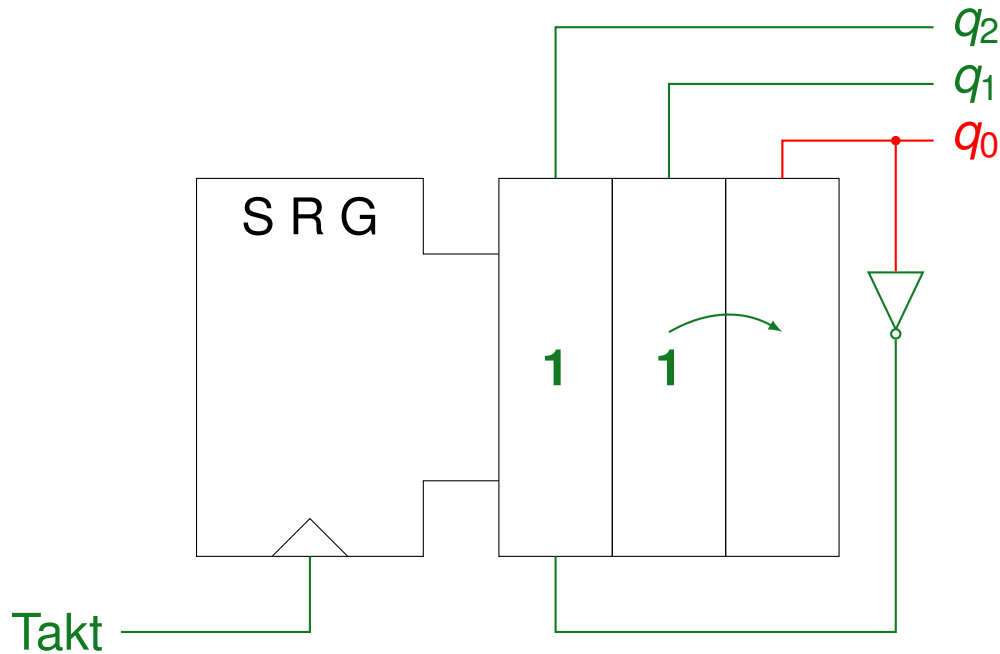
Startwert: 000



000 \mapsto 100 \mapsto 110

Aufgabe 4 – Schieberegister: Johnson-Zähler

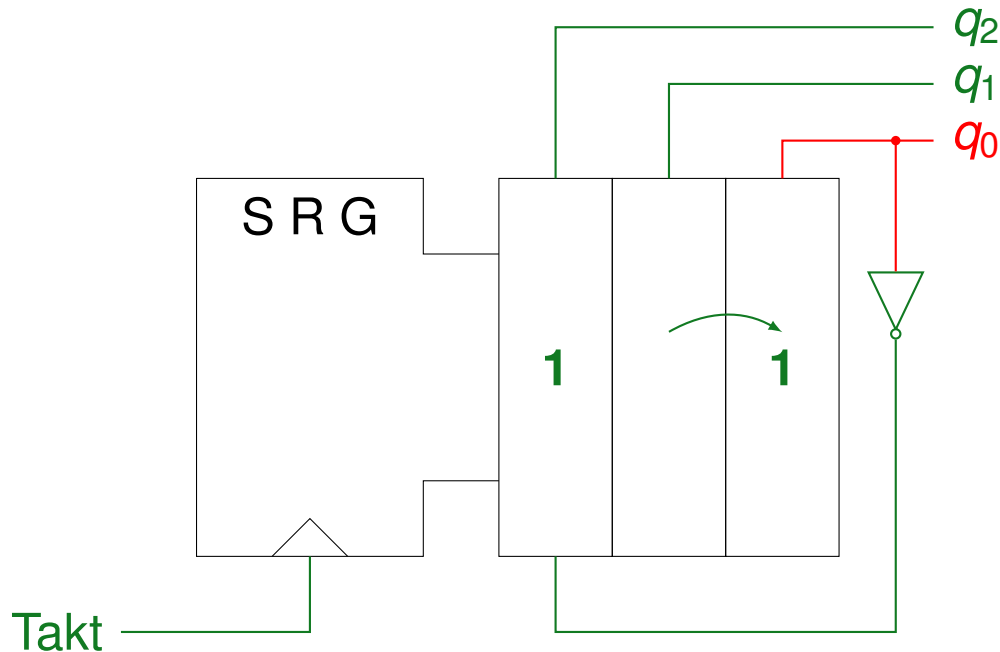
Startwert: 000



000 \mapsto 100 \mapsto 110

Aufgabe 4 – Schieberegister: Johnson-Zähler

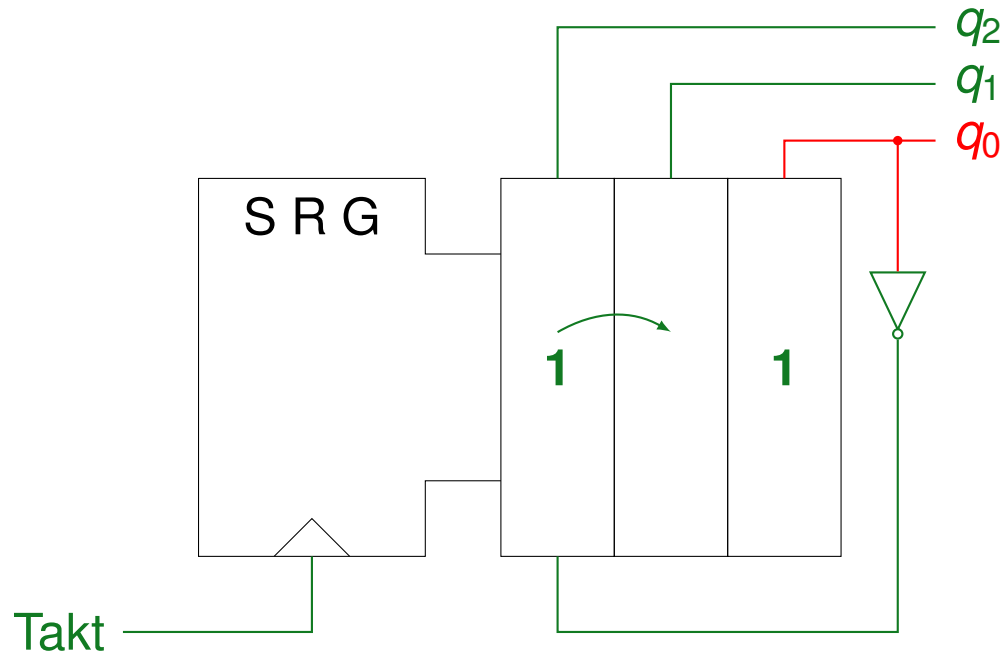
Startwert: 000



000 \mapsto 100 \mapsto 110

Aufgabe 4 – Schieberegister: Johnson-Zähler

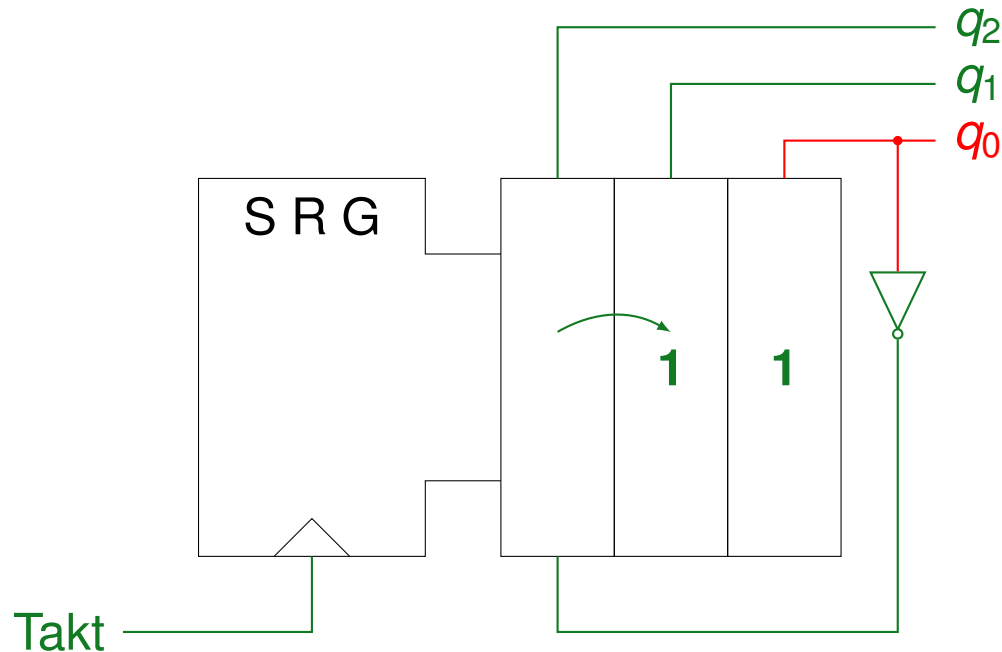
Startwert: 000



000 \mapsto 100 \mapsto 110

Aufgabe 4 – Schieberegister: Johnson-Zähler

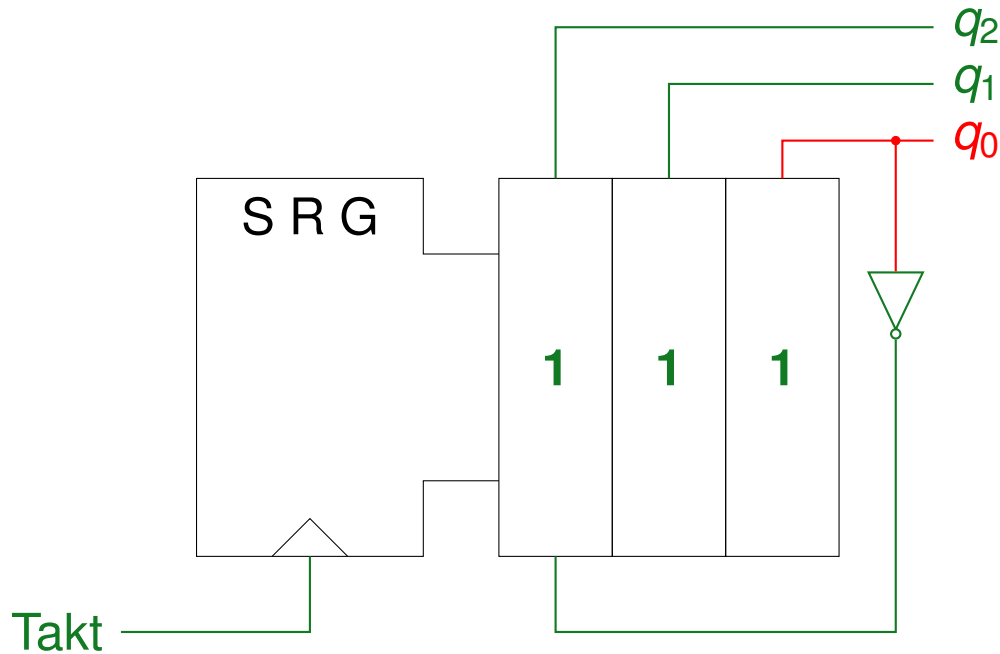
Startwert: 000



000 \mapsto 100 \mapsto 110

Aufgabe 4 – Schieberegister: Johnson-Zähler

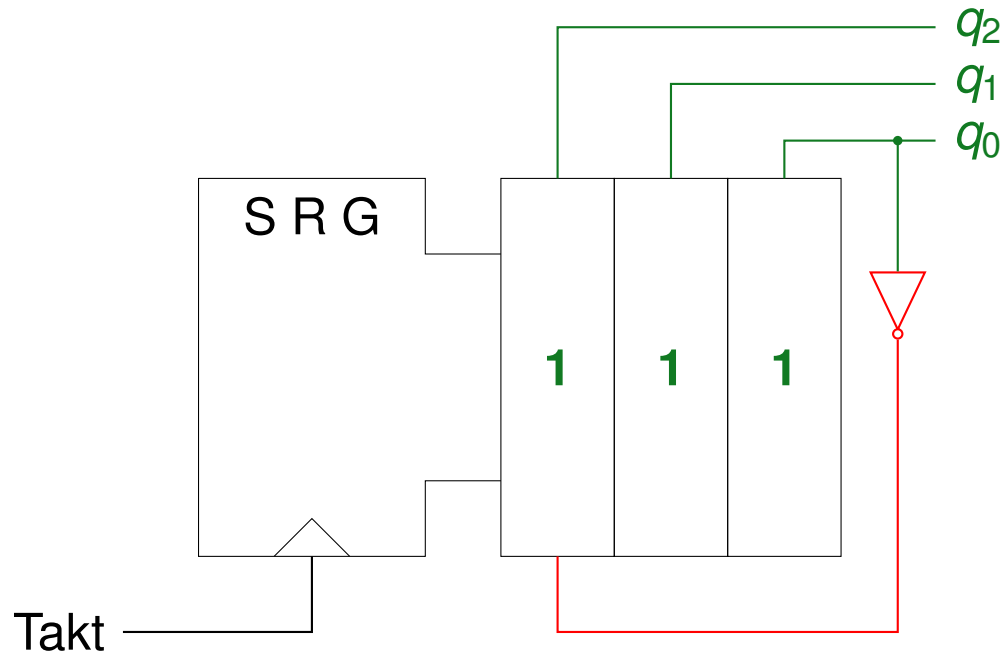
Startwert: 000



000 \mapsto 100 \mapsto 110

Aufgabe 4 – Schieberegister: Johnson-Zähler

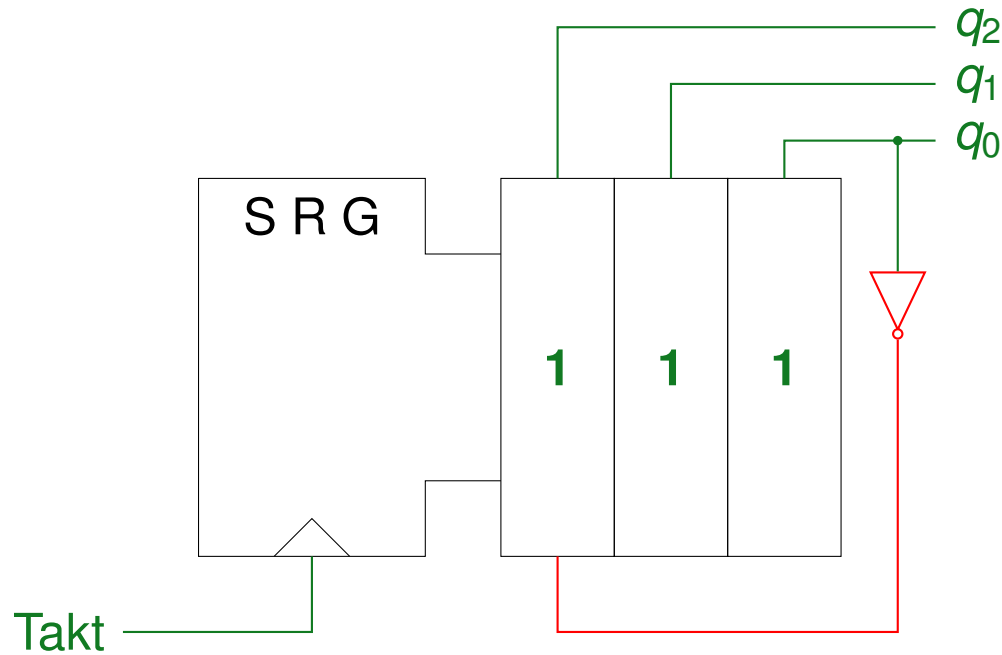
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111

Aufgabe 4 – Schieberegister: Johnson-Zähler

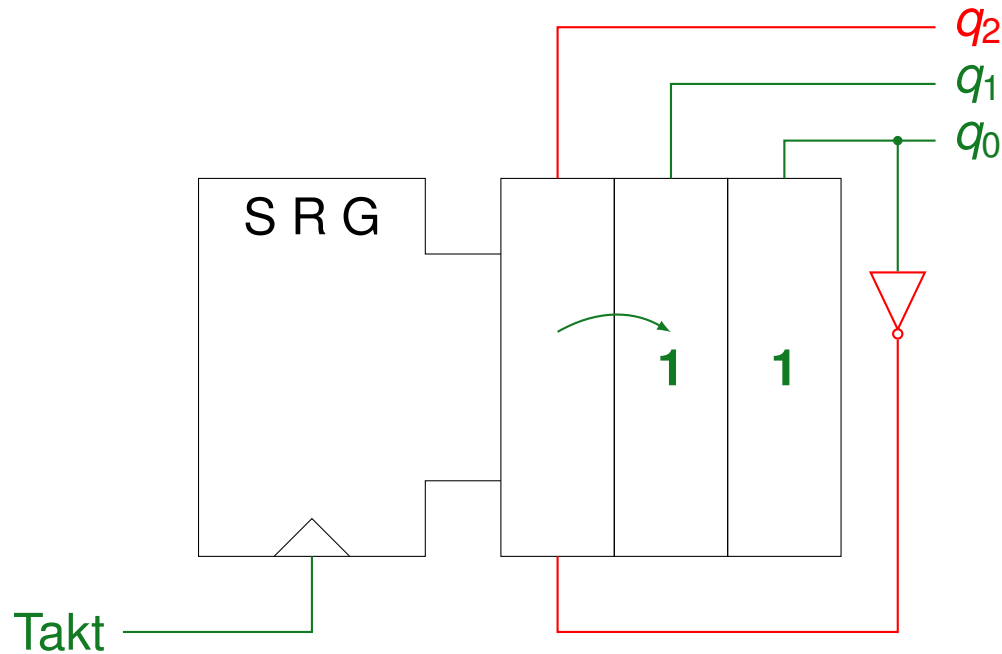
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111

Aufgabe 4 – Schieberegister: Johnson-Zähler

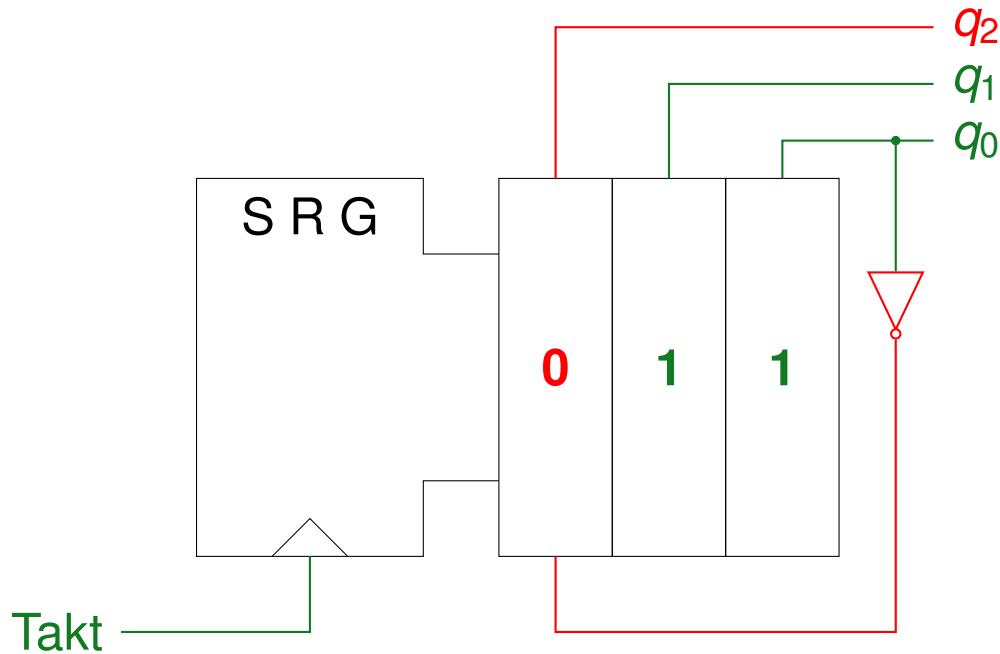
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111

Aufgabe 4 – Schieberegister: Johnson-Zähler

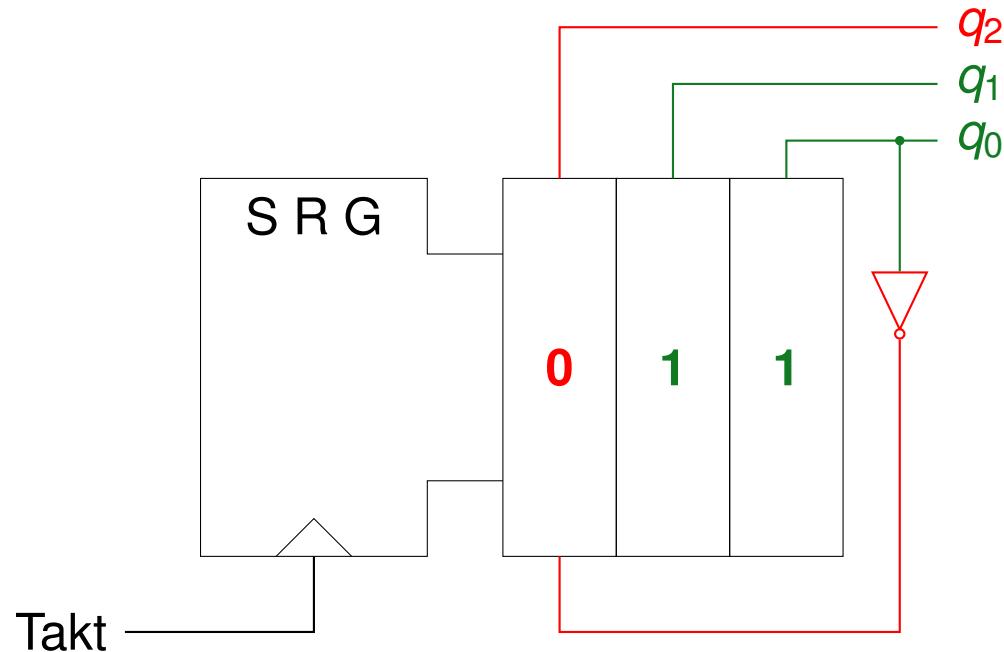
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111

Aufgabe 4 – Schieberegister: Johnson-Zähler

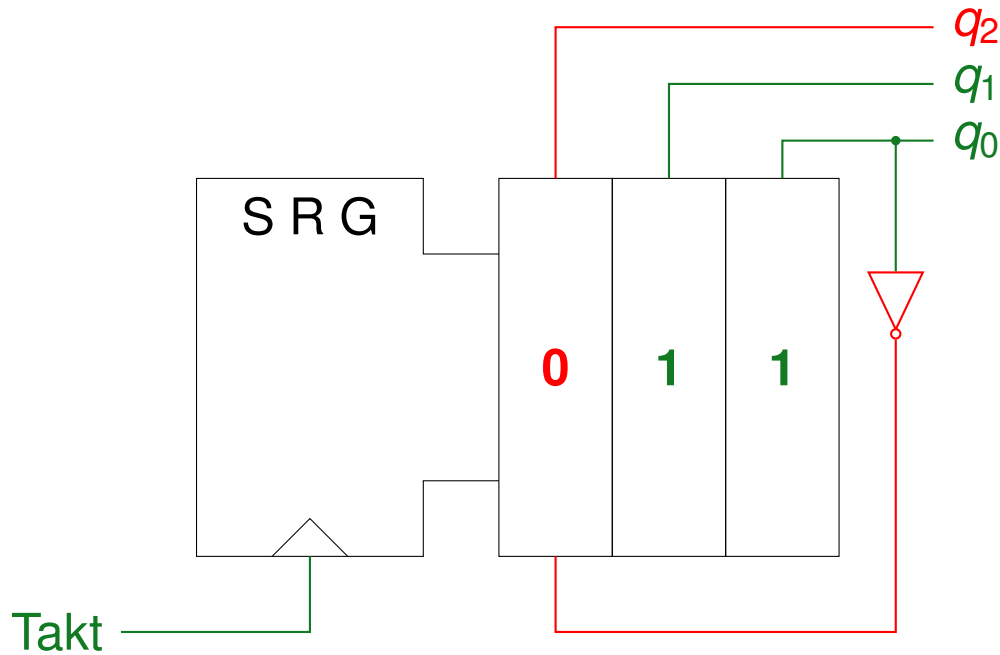
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011

Aufgabe 4 – Schieberegister: Johnson-Zähler

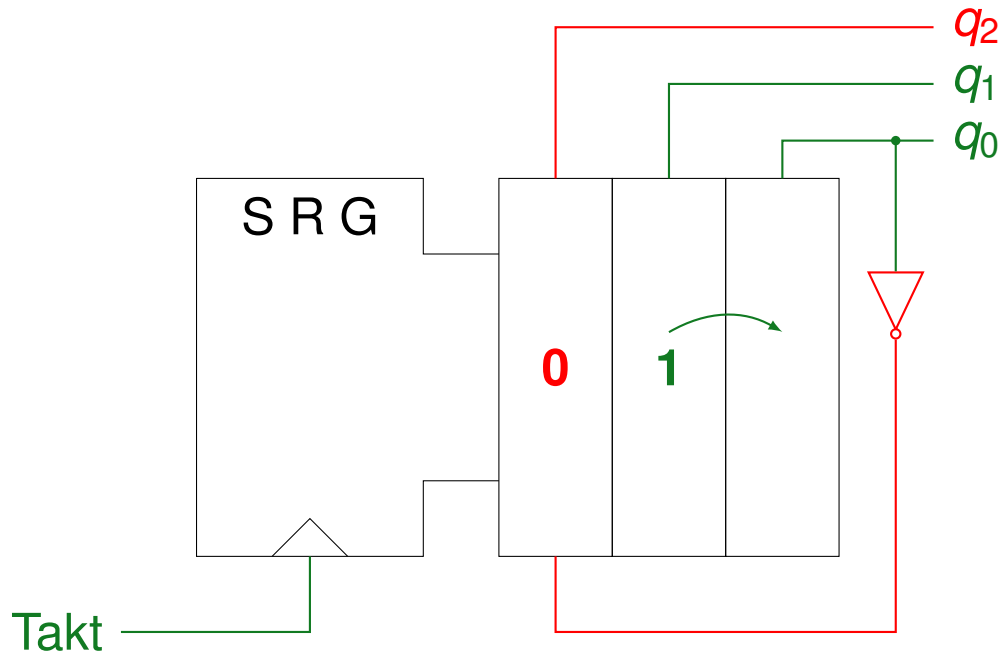
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011

Aufgabe 4 – Schieberegister: Johnson-Zähler

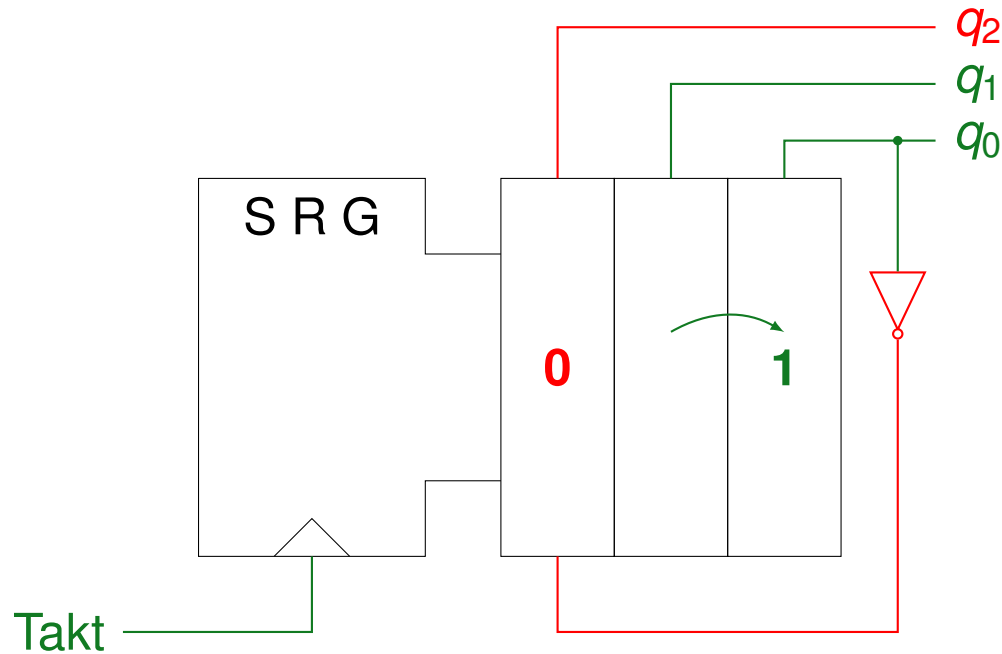
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011

Aufgabe 4 – Schieberegister: Johnson-Zähler

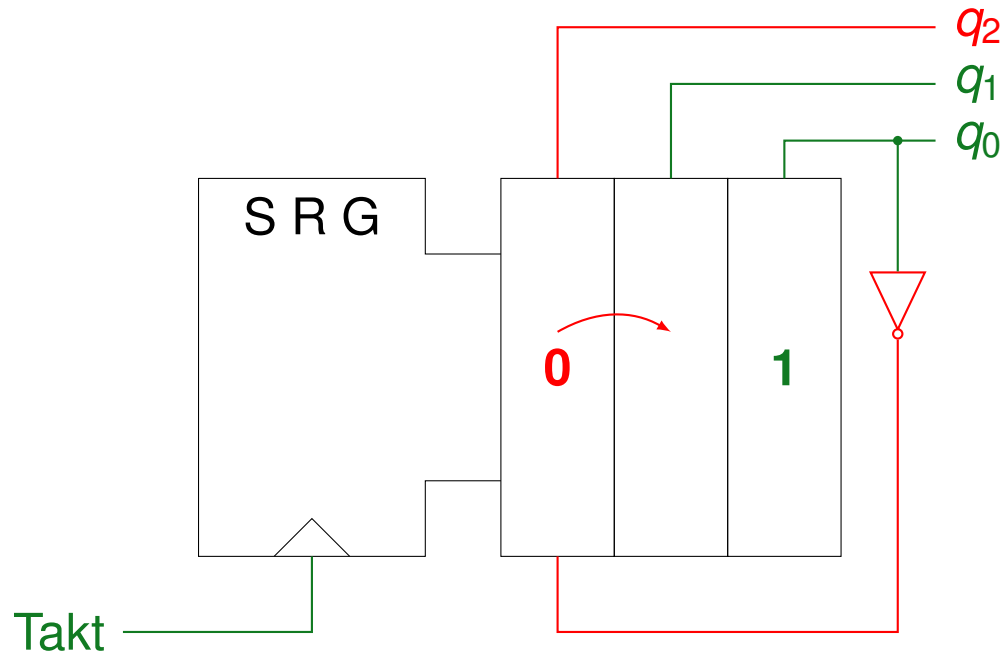
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011

Aufgabe 4 – Schieberegister: Johnson-Zähler

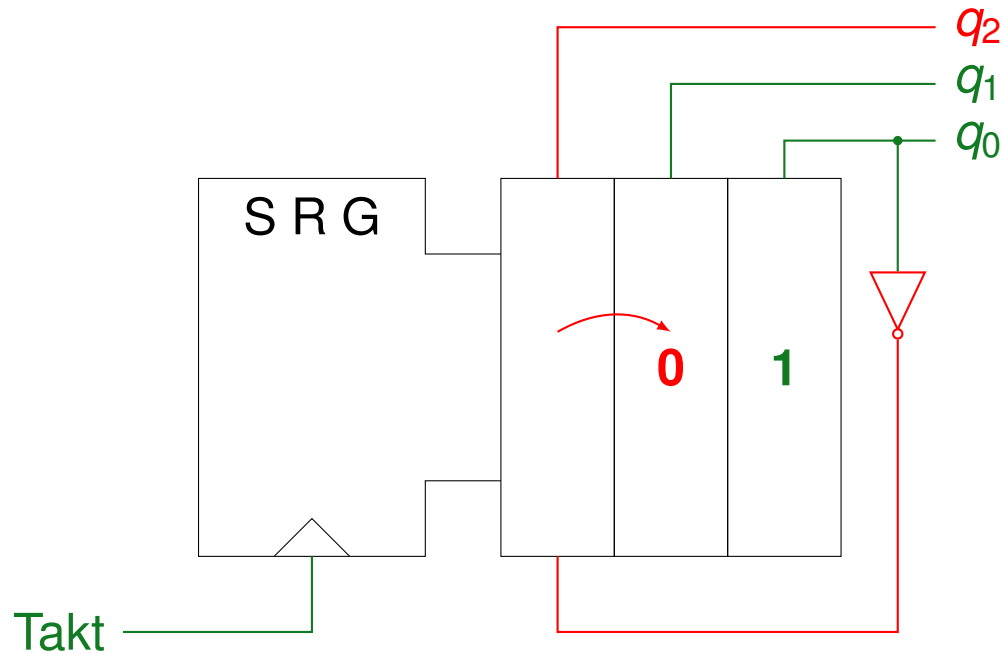
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011

Aufgabe 4 – Schieberegister: Johnson-Zähler

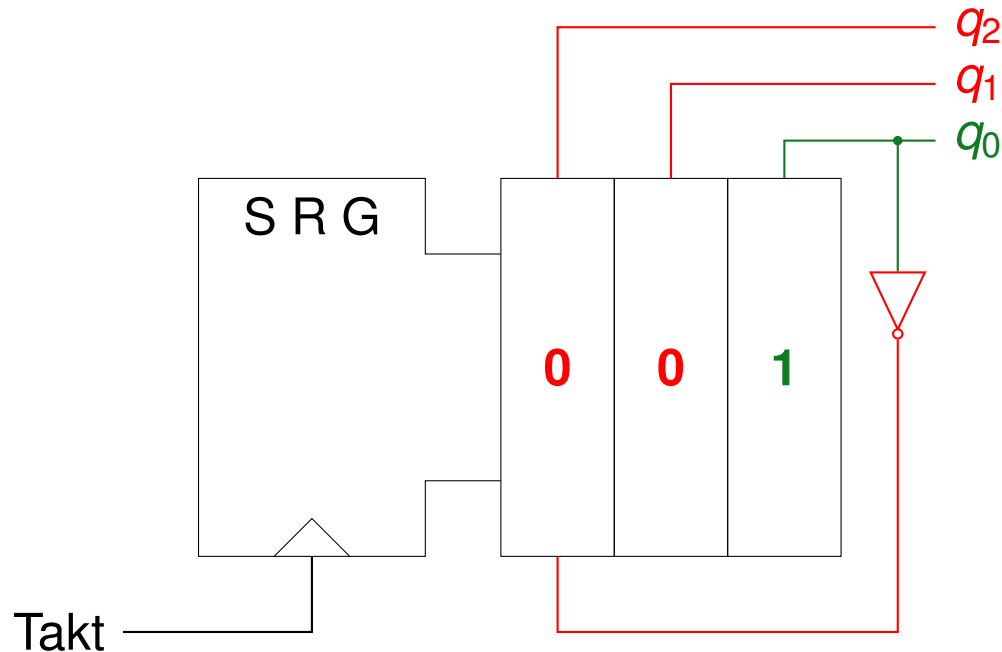
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011

Aufgabe 4 – Schieberegister: Johnson-Zähler

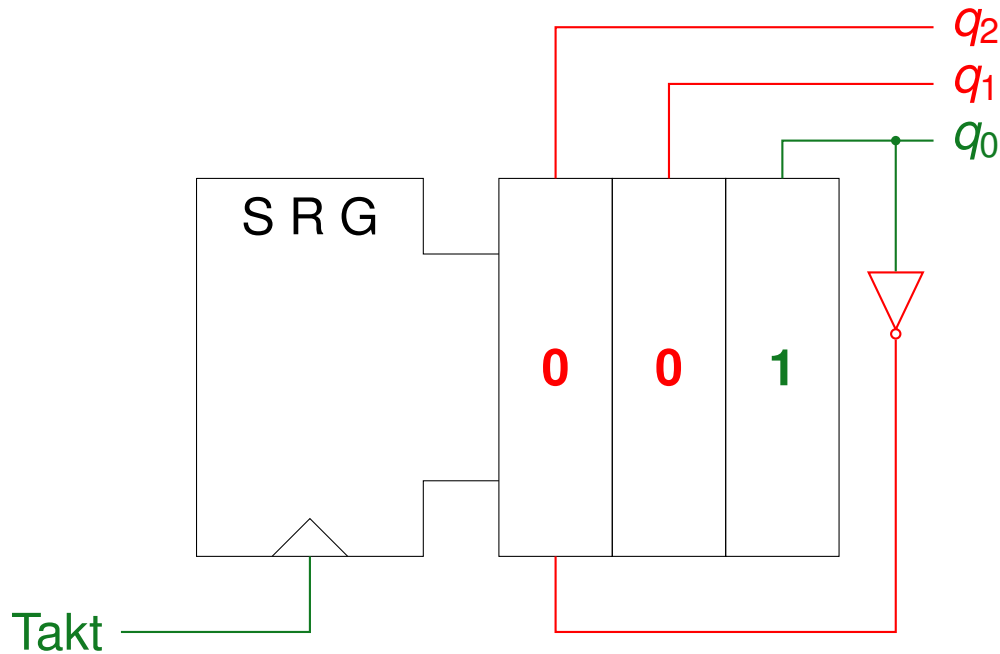
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001

Aufgabe 4 – Schieberegister: Johnson-Zähler

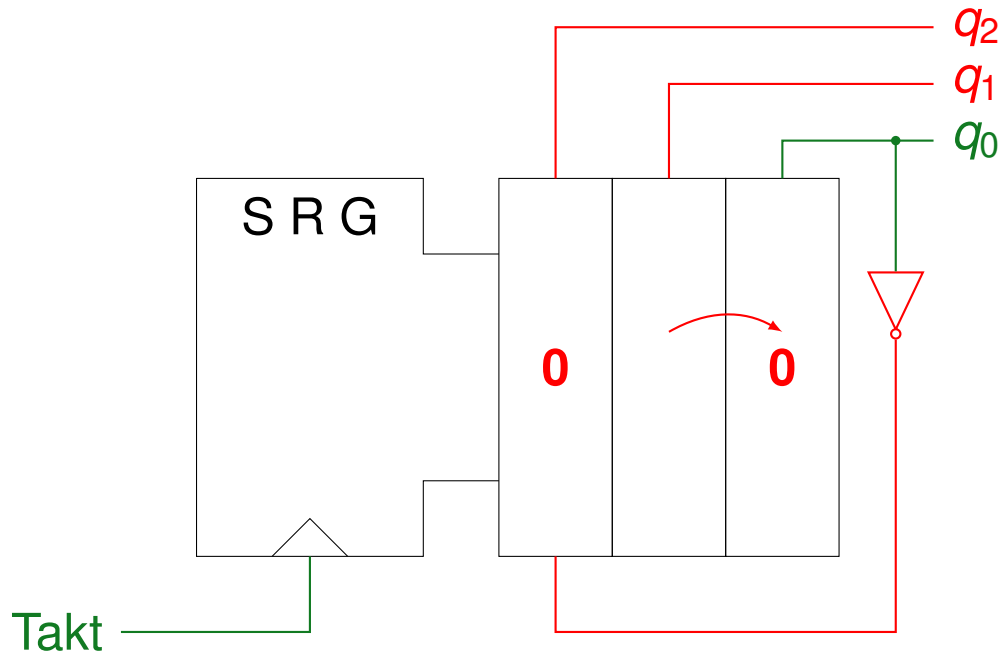
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001

Aufgabe 4 – Schieberegister: Johnson-Zähler

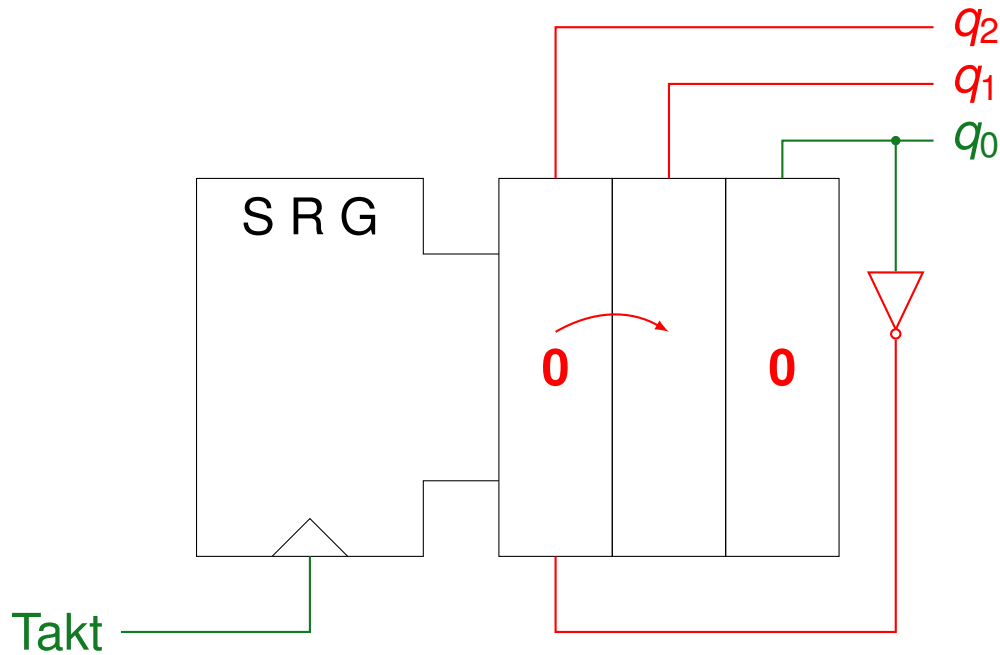
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001

Aufgabe 4 – Schieberegister: Johnson-Zähler

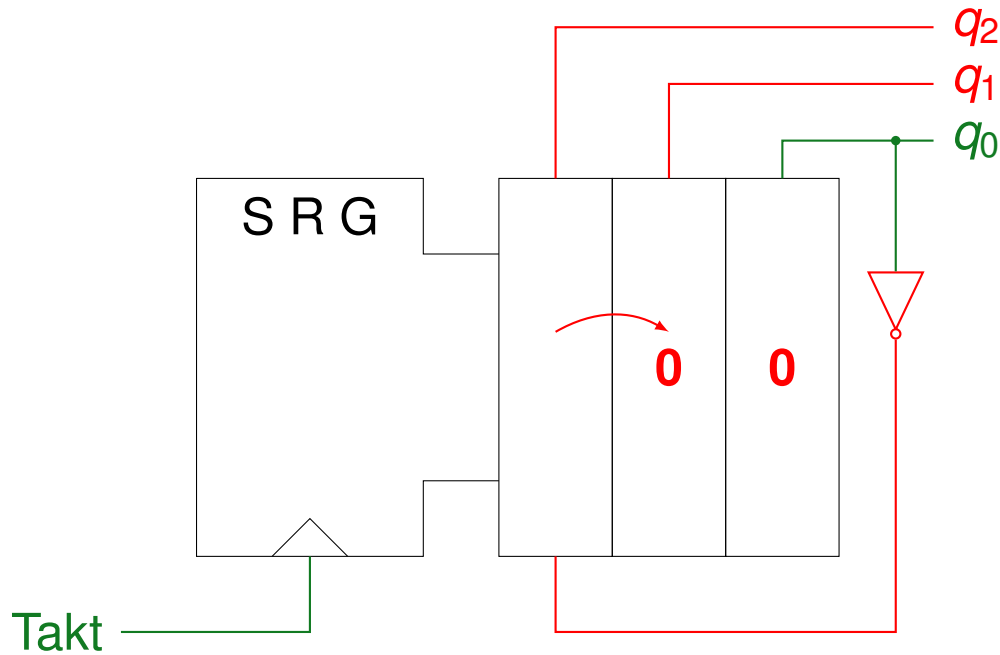
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001

Aufgabe 4 – Schieberegister: Johnson-Zähler

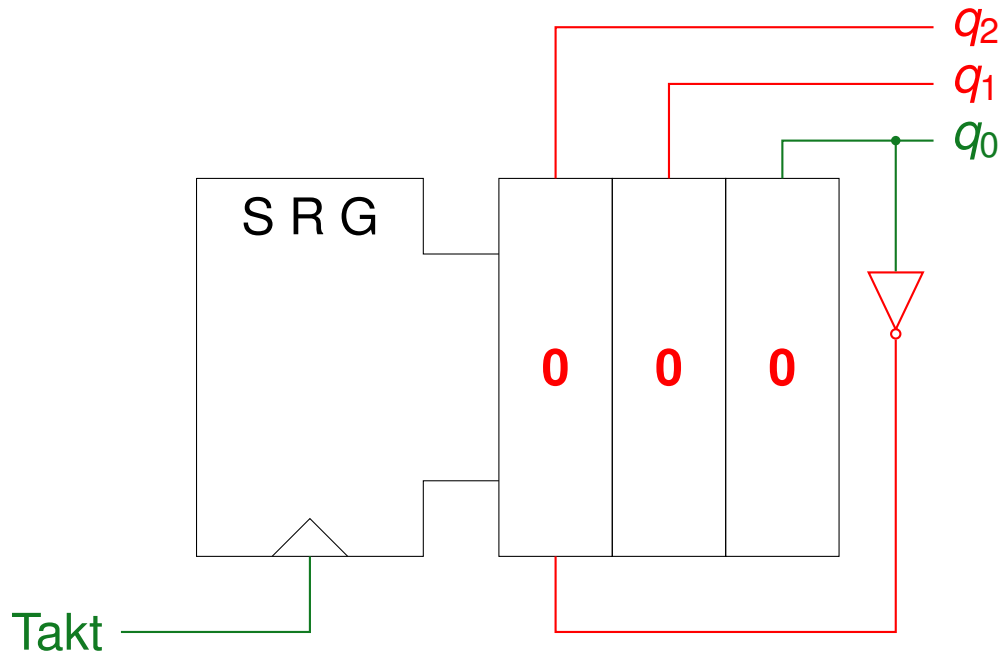
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001

Aufgabe 4 – Schieberegister: Johnson-Zähler

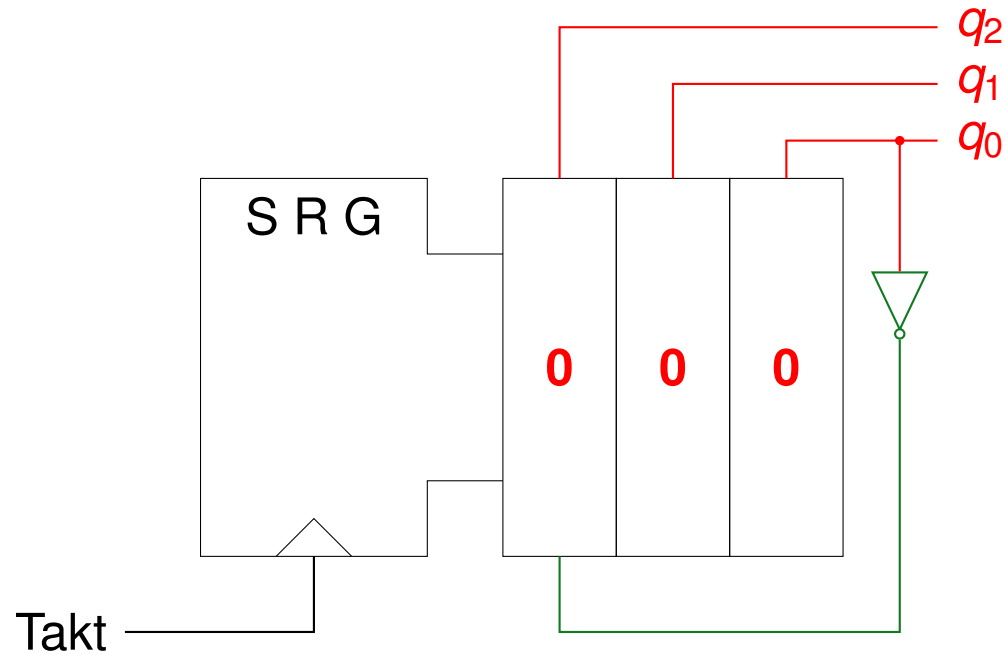
Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001

Aufgabe 4 – Schieberegister: Johnson-Zähler

Startwert: 000



000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

a)

Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

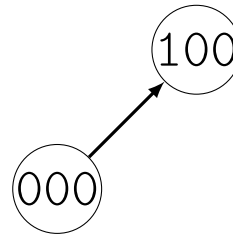
a)

000

Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

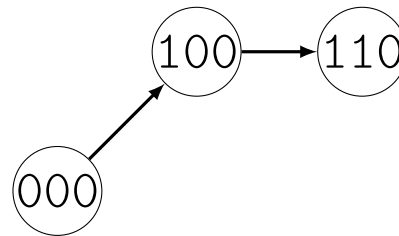
a)



Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

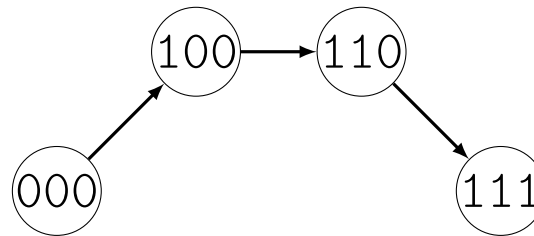
a)



Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

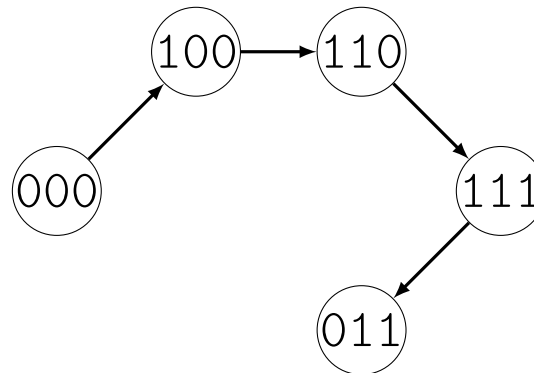
a)



Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

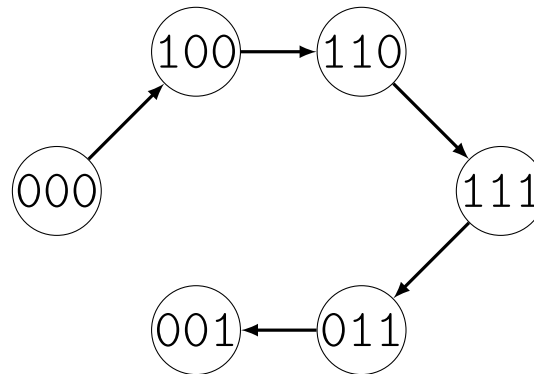
a)



Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

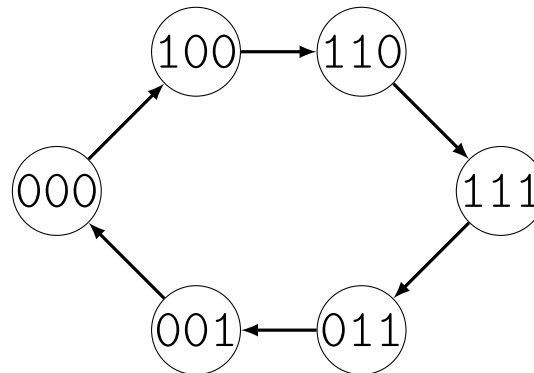
a)



Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

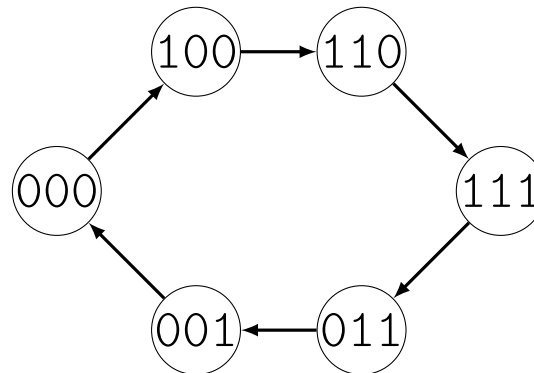
a)



Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

a)

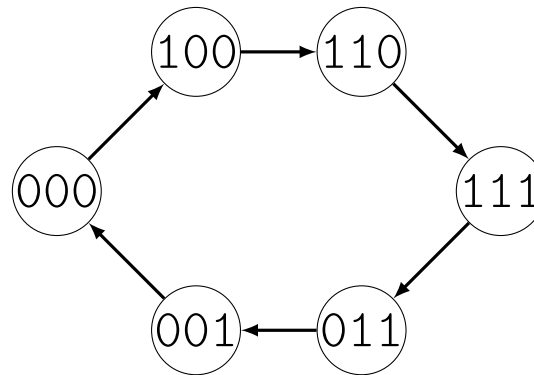


b) Welche Speicherbelegungen kommen in der Folge nicht vor?
Vervollständigen Sie den Graphen aus a) entsprechend.

Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

b) Welche Speicherbelegungen kommen in der Folge nicht vor?

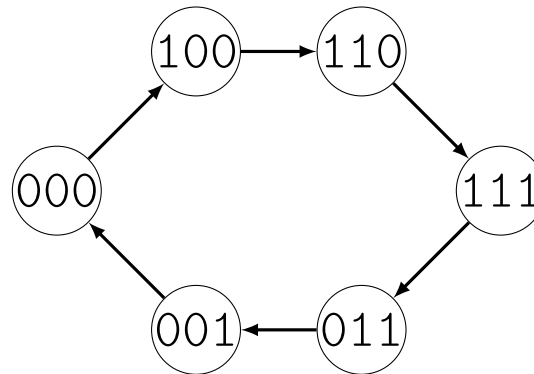


Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

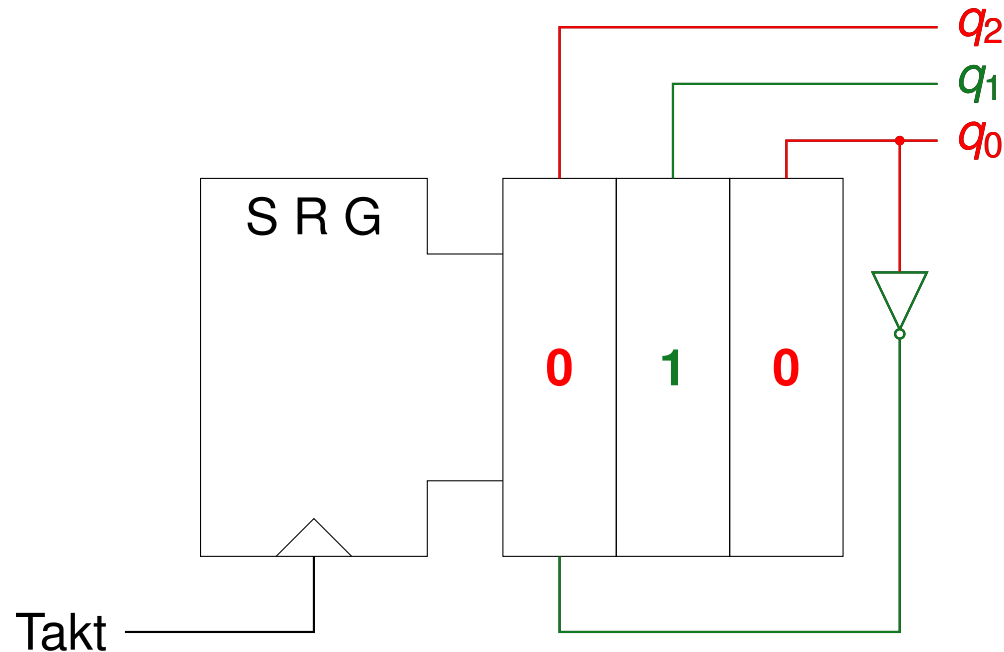
b) Welche Speicherbelegungen kommen in der Folge nicht vor?

\rightarrow 101 und 010



Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

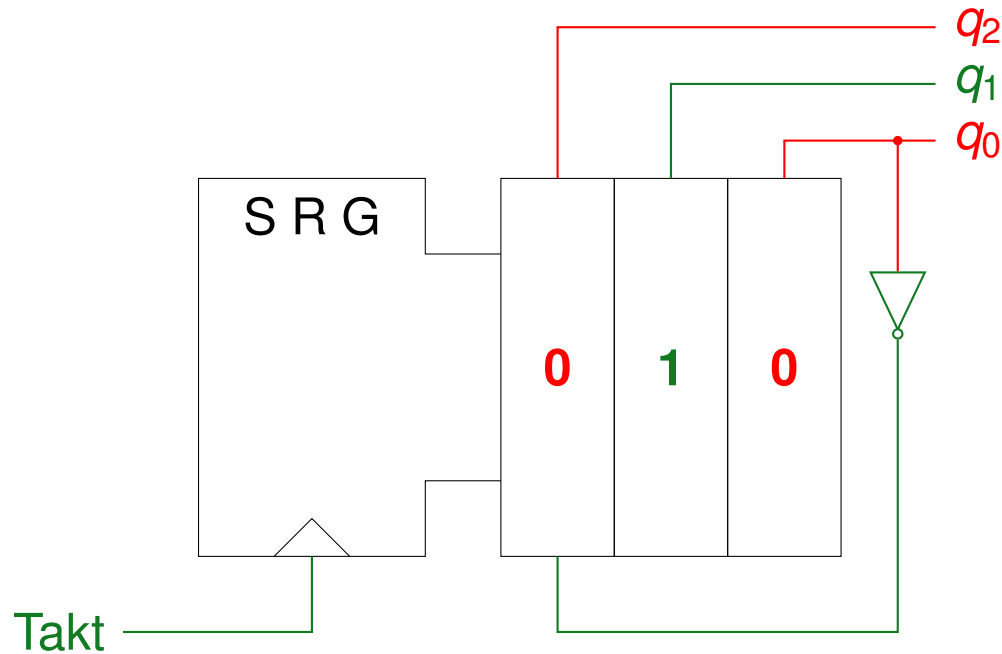
Startwert: 010



010

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

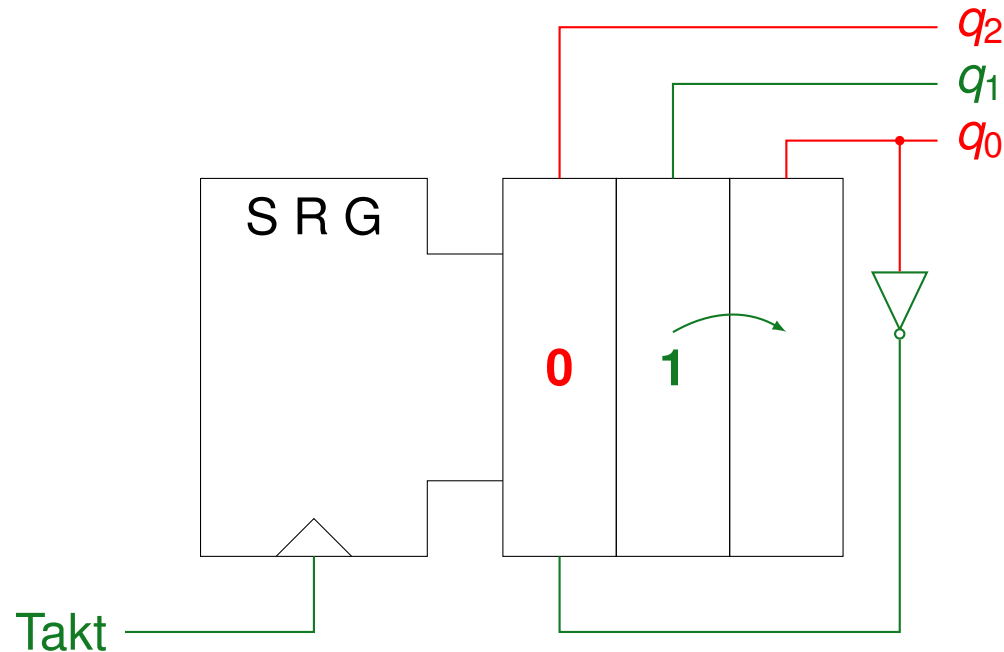
Startwert: 010



010

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

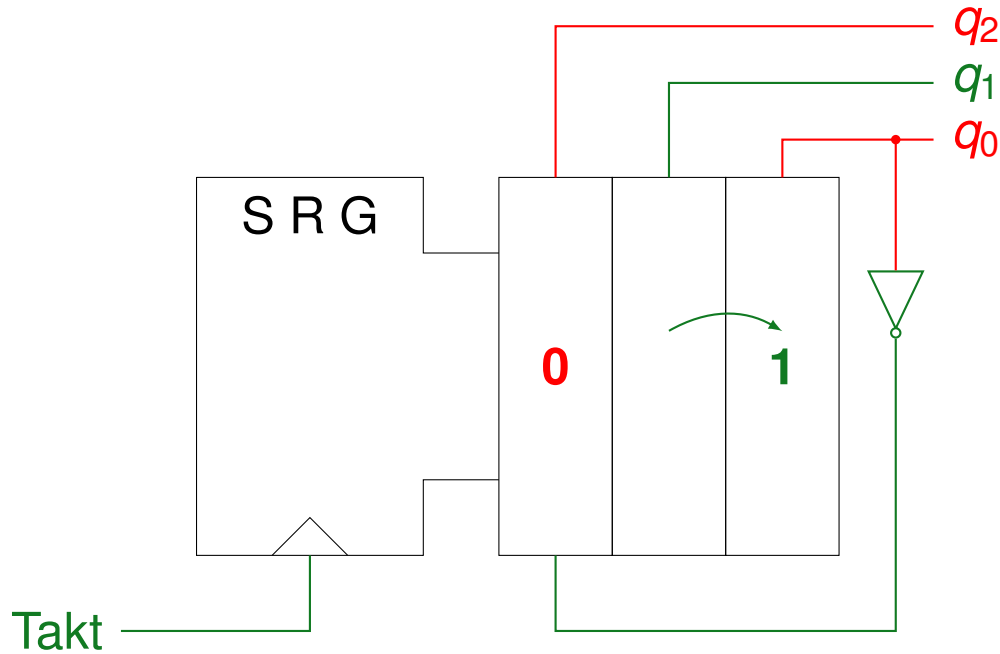
Startwert: 010



010

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

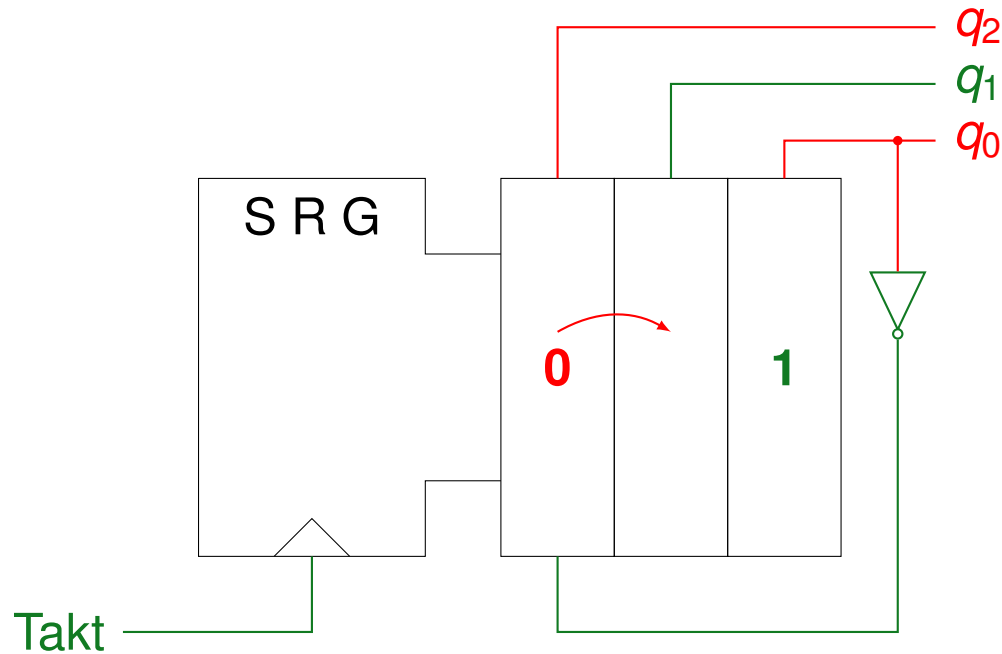
Startwert: 010



010

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

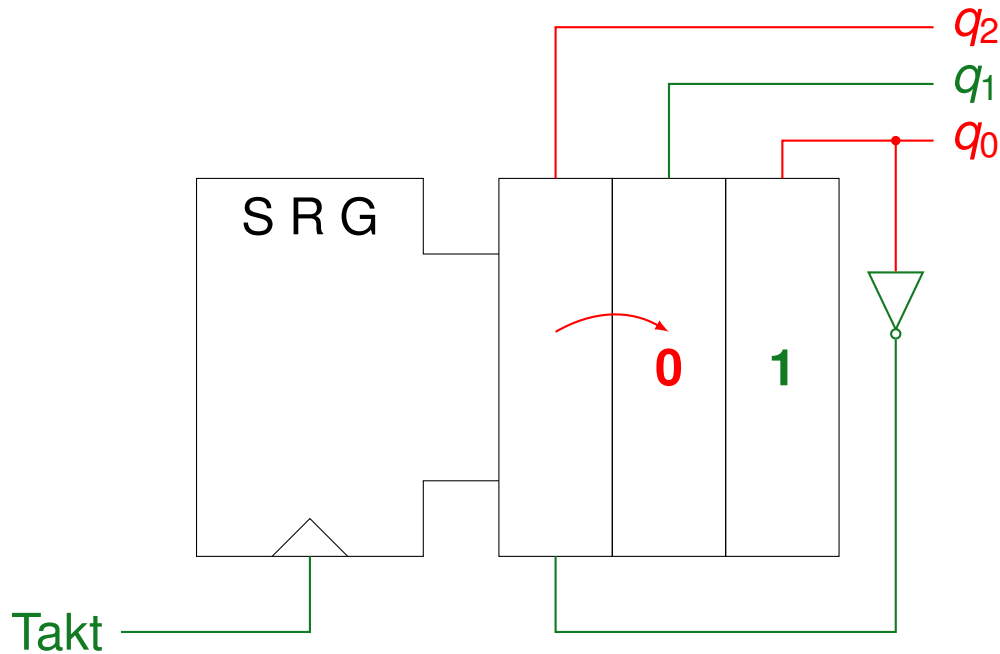
Startwert: 010



010

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

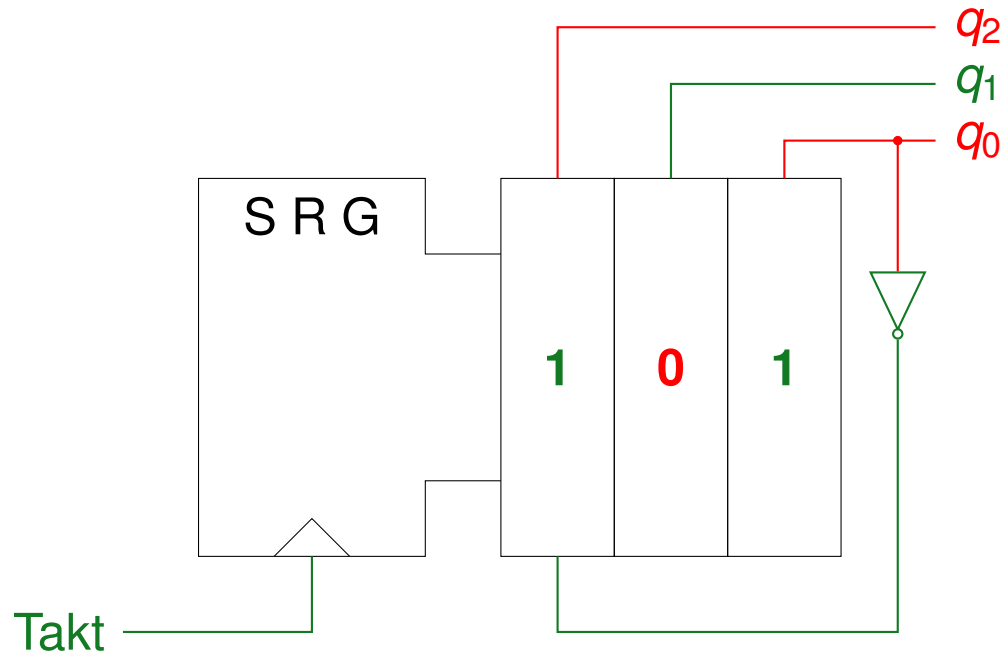
Startwert: 010



010

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

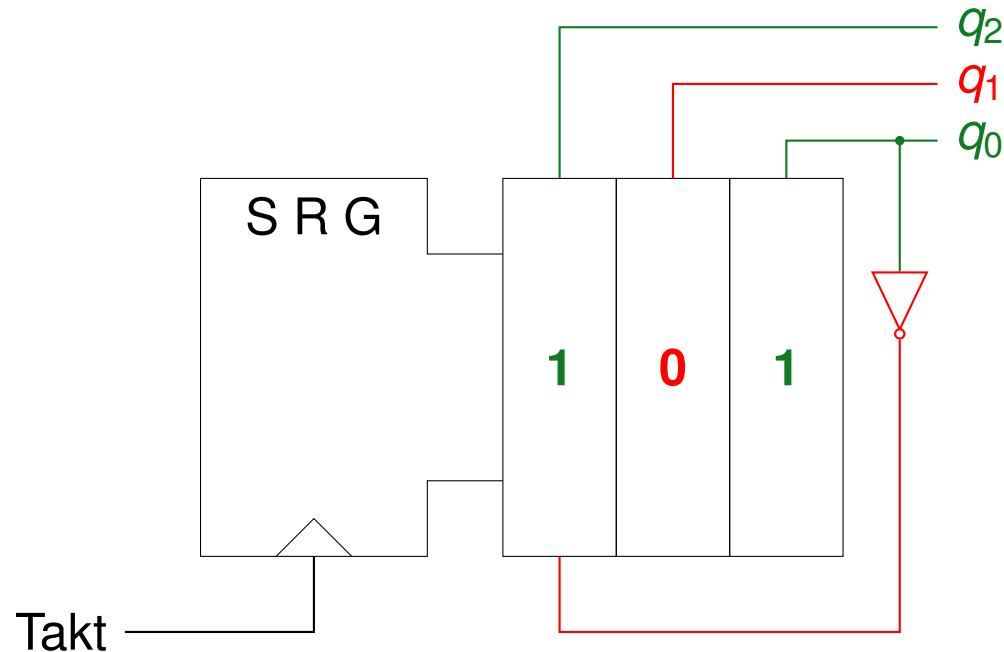
Startwert: 010



010

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

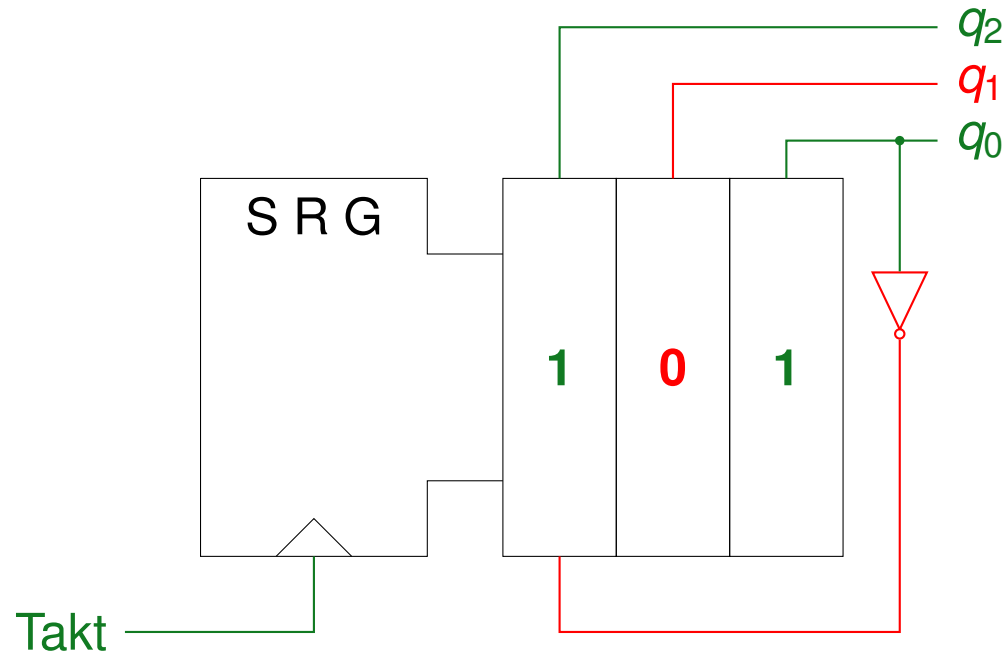
Startwert: 010



010 \mapsto 101

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

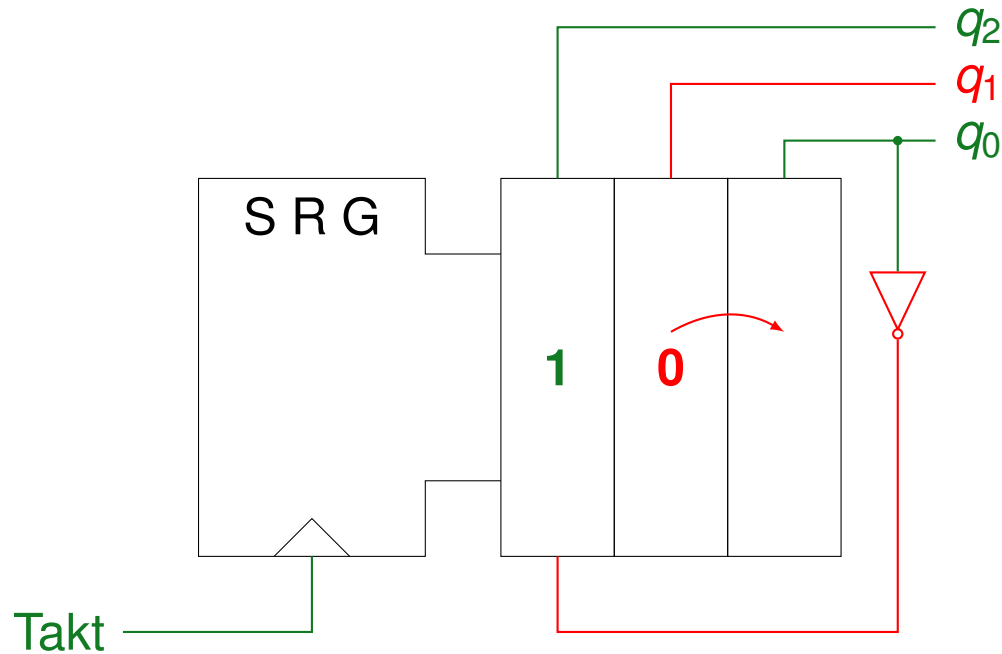
Startwert: 010



010 \mapsto 101

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

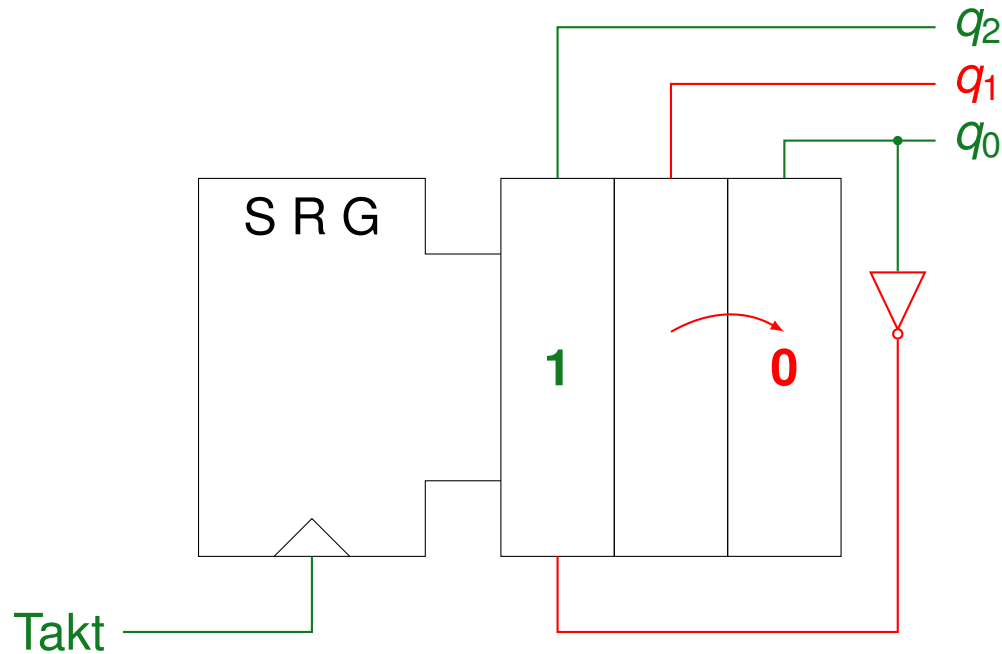
Startwert: 010



010 \mapsto 101

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

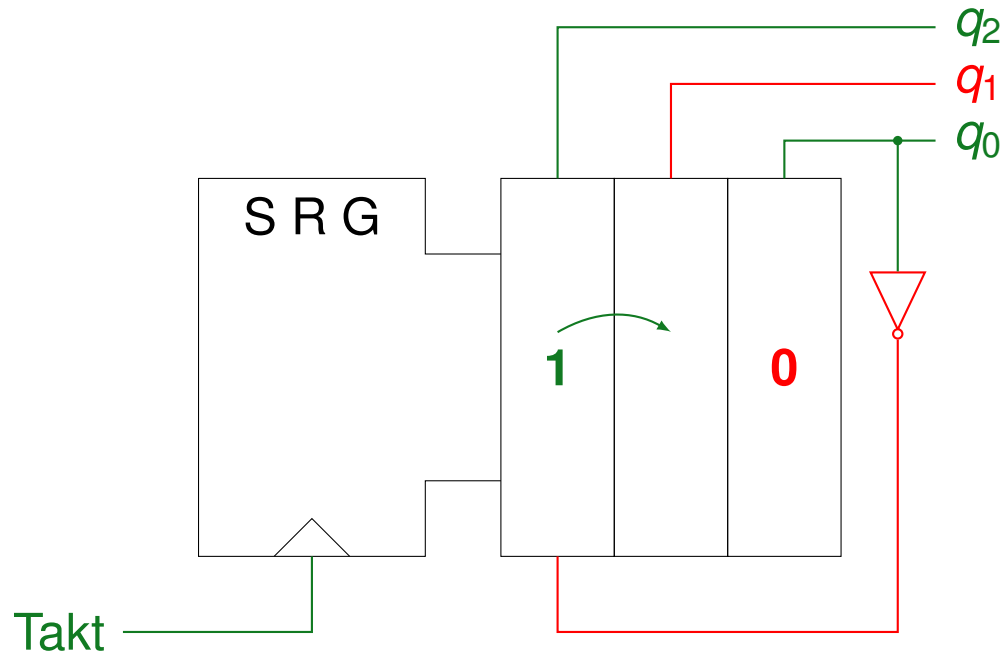
Startwert: 010



010 \mapsto 101

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

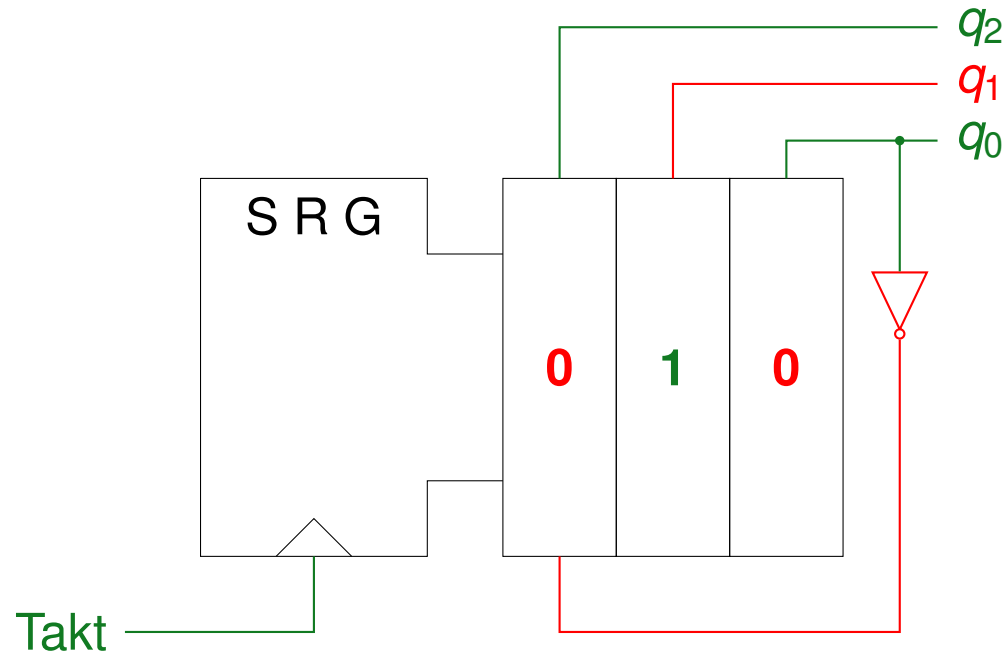
Startwert: 010



010 \mapsto 101

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

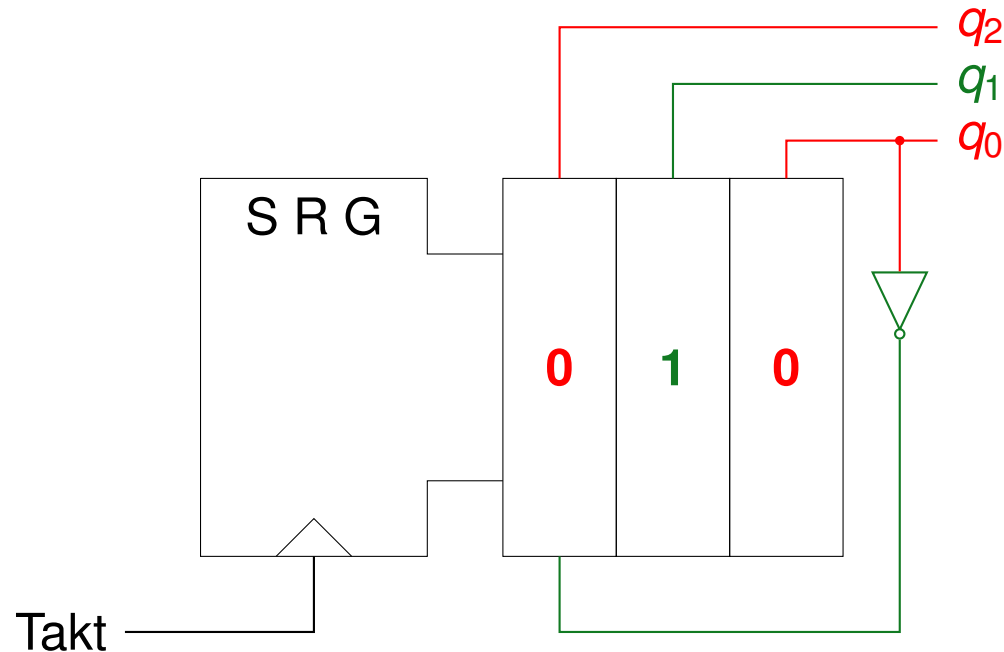
Startwert: 010



010 \mapsto 101

Aufgabe 4 – Schieberegister: Johnson-Zähler *revisited*

Startwert: 010



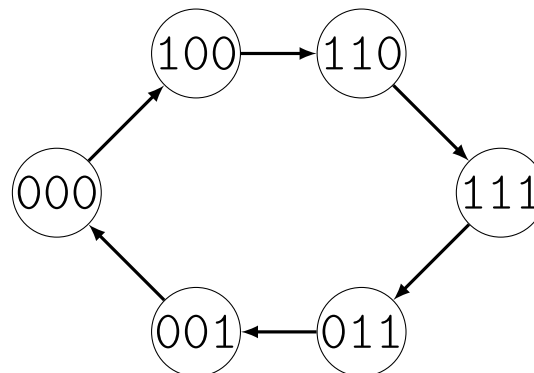
010 \mapsto 101 \mapsto 010

Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000
 010 \mapsto 101 \mapsto 010

b) Welche Speicherbelegungen kommen in der Folge nicht vor?

\rightarrow 101 und 010



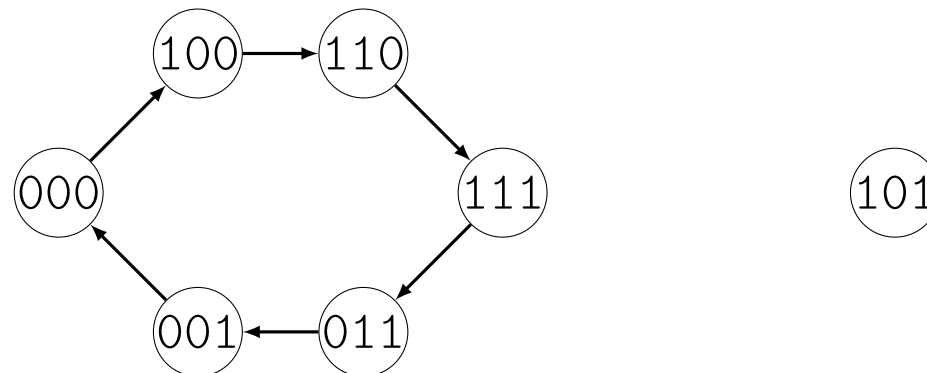
Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

010 \mapsto 101 \mapsto 010

b) Welche Speicherbelegungen kommen in der Folge nicht vor?

\rightarrow 101 und 010



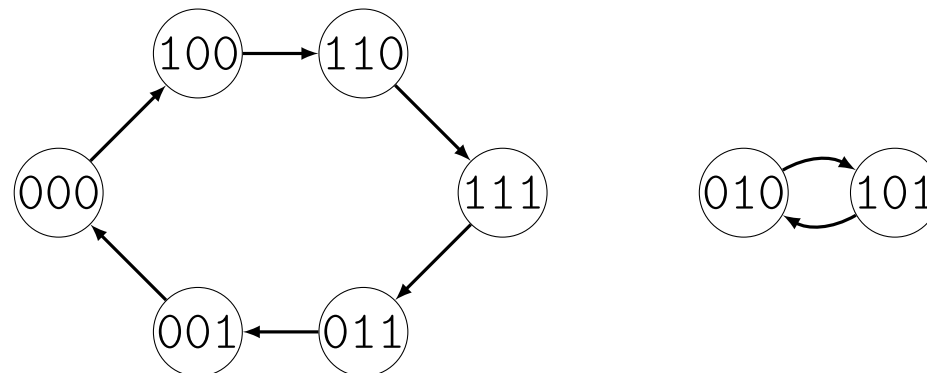
Aufgabe 4 – Schieberegister

000 \mapsto 100 \mapsto 110 \mapsto 111 \mapsto 011 \mapsto 001 \mapsto 000

010 \mapsto 101 \mapsto 010

b) Welche Speicherbelegungen kommen in der Folge nicht vor?

\rightarrow 101 und 010



Aufgabe 4 – Schieberegister

- c) Realisieren Sie ein Schaltnetz, welches aus allen N Zuständen des Johnson-Zählers aus der Abbildung ein Lauflicht für 6 einzelne Lampen $L_1 \dots L_6$ ansteuern kann. Sie können dazu auf die Ausgangswerte q_0, q_1, q_2 sowie deren negierte Signale $\overline{q_0}, \overline{q_1}, \overline{q_2}$ zurückgreifen. Beginnen Sie mit einer Wahrheitstabelle.

Aufgabe 4 – Schieberegister

- c) Lauflicht → Leuchtende Lampe „läuft“ von links nach rechts. Sprich: Nach L_1 leuchtet L_2 , dann L_3 ... und nach L_6 leuchtet wieder L_1 .

Aufgabe 4 – Schieberegister

- c) Lauflicht → Leuchtende Lampe „läuft“ von links nach rechts. Sprich: Nach L_1 leuchtet L_2 , dann L_3 ... und nach L_6 leuchtet wieder L_1 .
 Unser Johnson-Zähler hat **genau** 6 zyklische Zustände. Wir verwenden diese also und stellen unsere Wahrheitstabelle auf:

Aufgabe 4 – Schieberegister

c) Lauflicht → Leuchtende Lampe „läuft“ von links nach rechts. Sprich: Nach L_1 leuchtet L_2 , dann L_3 ... und nach L_6 leuchtet wieder L_1 .
 Unser Johnson-Zähler hat **genau** 6 zyklische Zustände. Wir verwenden diese also und stellen unsere Wahrheitstabelle auf:

q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	1	0	0	0	0	0
1	0	0	0	1	0	0	0	0
1	1	0	0	0	1	0	0	0
1	1	1	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0
0	0	1	0	0	0	0	0	1
0	1	0	-	-	-	-	-	-
1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister

- c) Lauflicht → Leuchtende Lampe „läuft“ von links nach rechts. Sprich: Nach L_1 leuchtet L_2 , dann L_3 ... und nach L_6 leuchtet wieder L_1 . Unser Johnson-Zähler hat **genau** 6 zyklische Zustände. Wir verwenden diese also und stellen unsere Wahrheitstabelle auf:

q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	1	0	0	0	0	0
1	0	0	0	1	0	0	0	0
1	1	0	0	0	1	0	0	0
1	1	1	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0
0	0	1	0	0	0	0	0	1
0	1	0	-	-	-	-	-	-
1	0	1	-	-	-	-	-	-

Da die Zustände 010 und 101 nie auftreten (siehe Teilaufgabe b), können sie als Freistellen angenommen werden → Damit ergeben sich „minimalere“ Funktionen.

Aufgabe 4 – Schieberegister: Teilaufgabe c

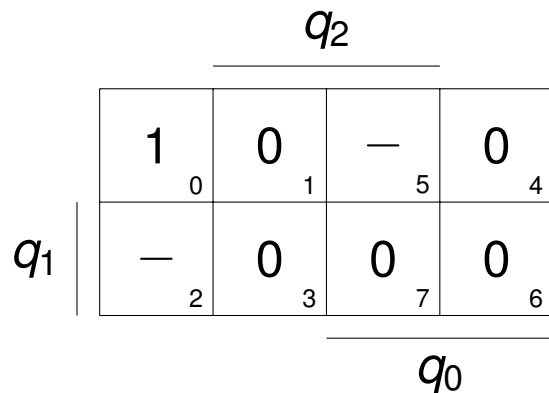
Wir minimieren also per Symmetriediagramm:

Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_1 :

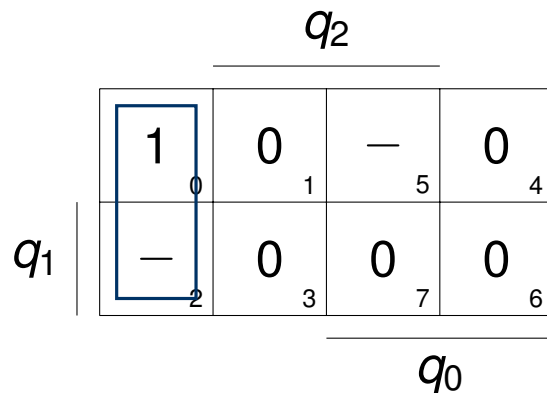


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_1 :

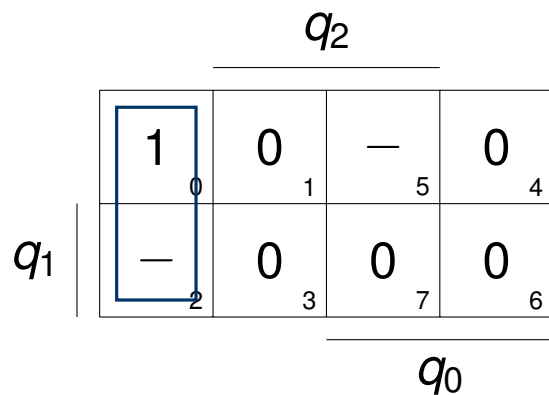


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_1 :



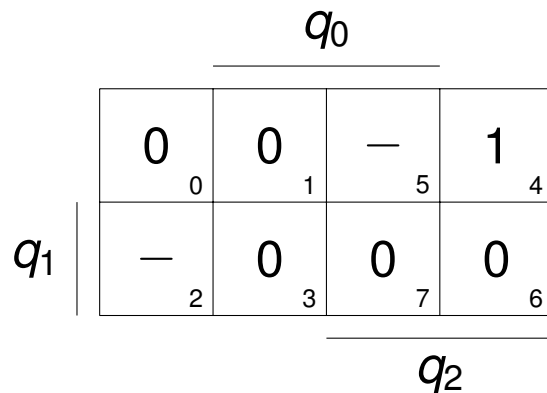
$$\rightarrow L_1 = \overline{q_2} \overline{q_0}$$

Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_2 :

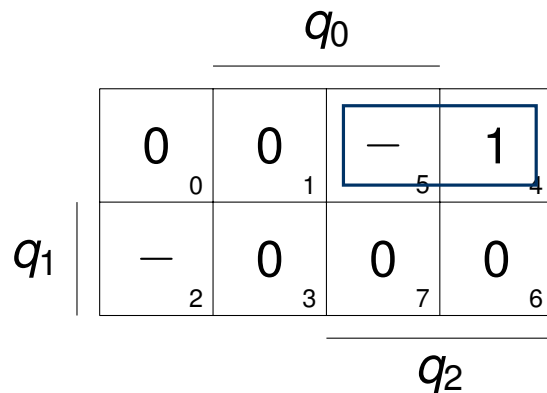


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_2 :

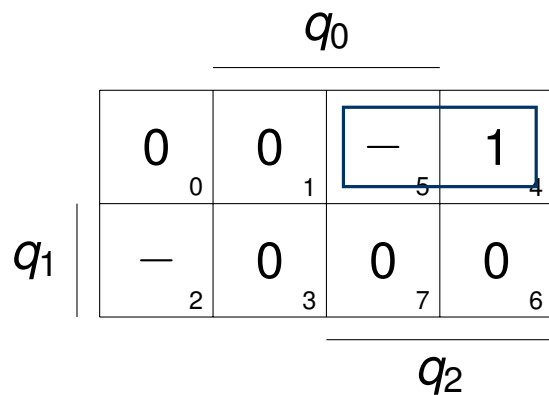


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_2 :



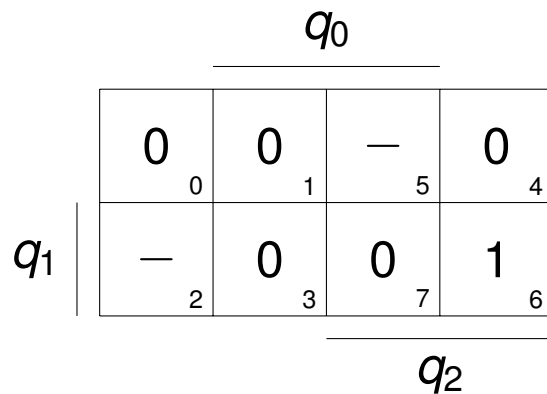
$$\rightarrow L_2 = q_2 \overline{q_1}$$

Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_3 :

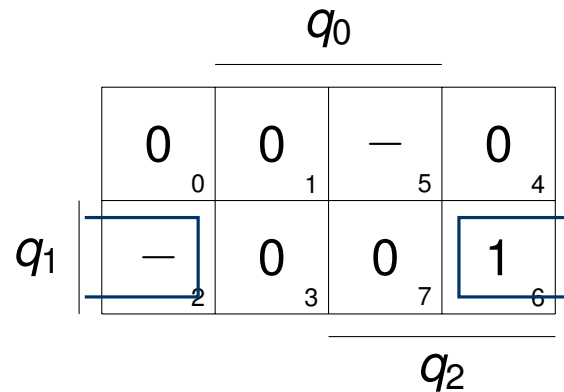


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_3 :

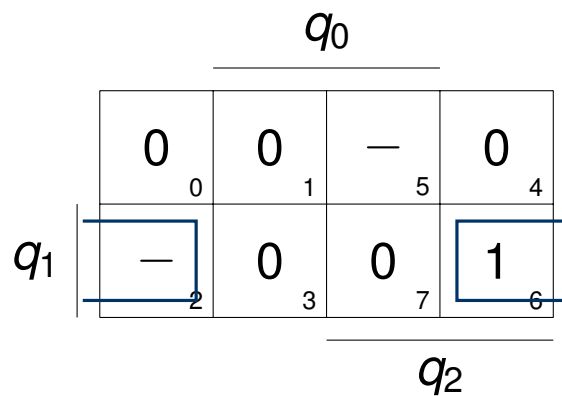


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_3 :



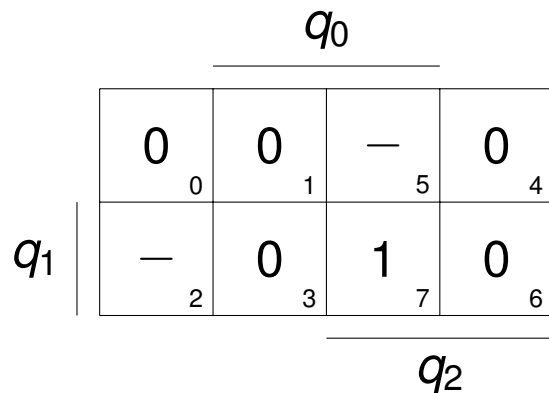
$$\rightarrow L_3 = q_1 \overline{q_0}$$

Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_4 :

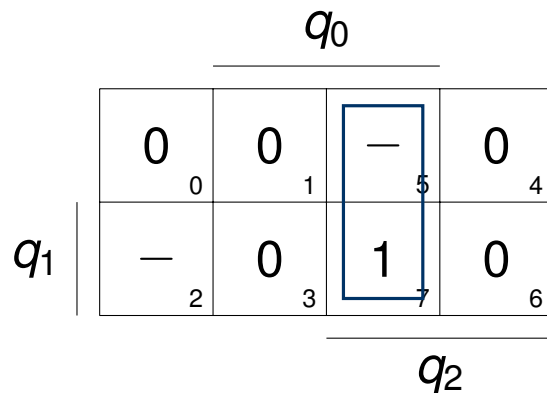


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_4 :

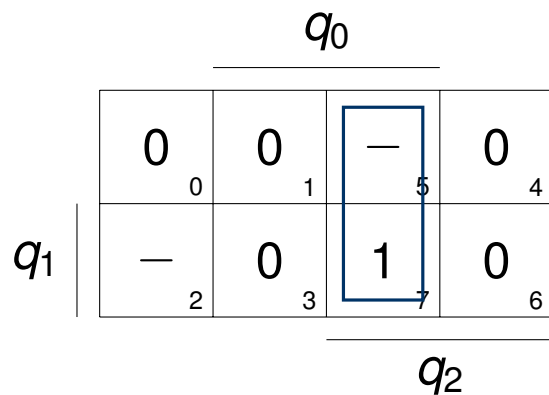


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_4 :



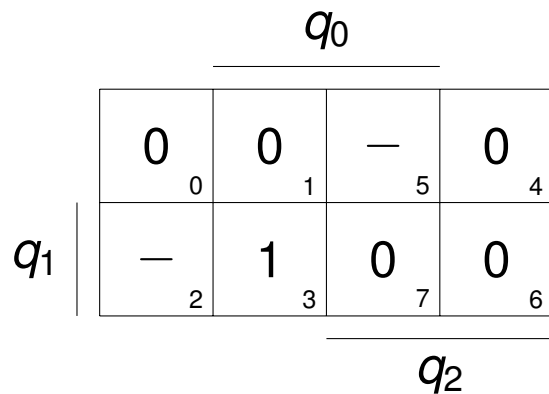
$\rightarrow L_4 = q_2 q_0$

Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_5 :

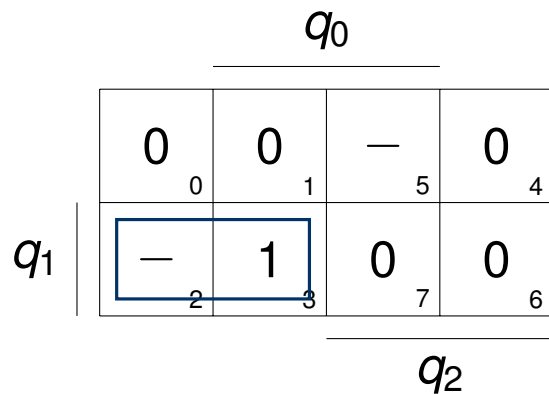


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_5 :

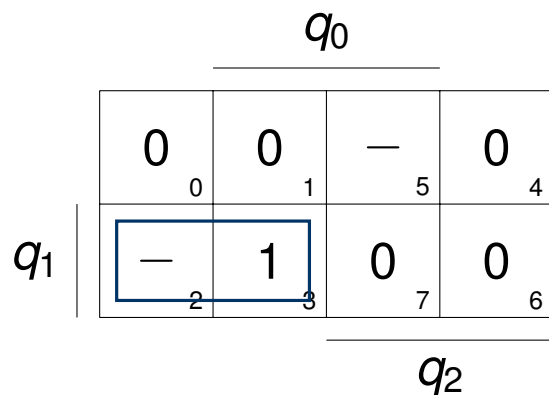


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_5 :



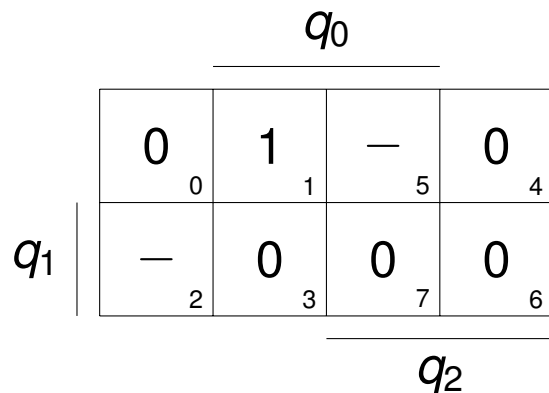
$$\rightarrow L_5 = \overline{q_2} q_1$$

Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_6 :

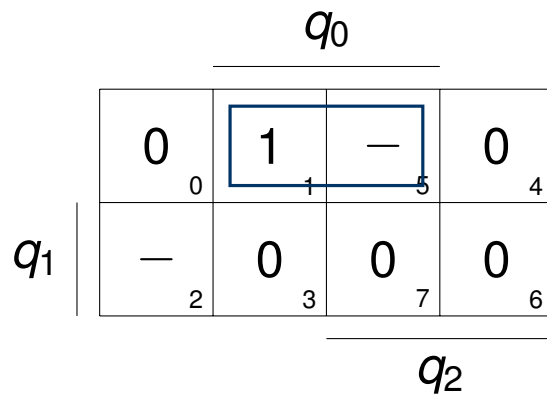


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_6 :

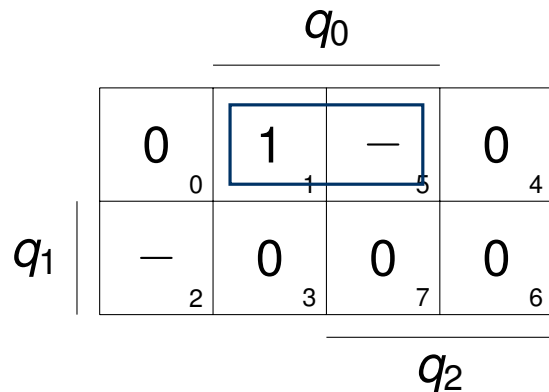


Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Wir minimieren also per Symmetriediagramm:

L_6 :



$$\rightarrow L_6 = \overline{q_1} q_0$$

Dec	q_2	q_1	q_0	L_1	L_2	L_3	L_4	L_5	L_6
0	0	0	0	1	0	0	0	0	0
4	1	0	0	0	1	0	0	0	0
6	1	1	0	0	0	1	0	0	0
7	1	1	1	0	0	0	1	0	0
3	0	1	1	0	0	0	0	1	0
1	0	0	1	0	0	0	0	0	1
2	0	1	0	-	-	-	-	-	-
5	1	0	1	-	-	-	-	-	-

Aufgabe 4 – Schieberegister: Teilaufgabe c

Durch Minimierung erhalten wir:

$$L_1 = \overline{q_2} \cdot \overline{q_0}, \quad L_2 = q_2 \cdot \overline{q_1}, \quad L_3 = q_1 \cdot \overline{q_0}, \quad L_4 = q_2 \cdot q_0, \quad L_5 = \overline{q_2} \cdot q_1, \quad L_6 = \overline{q_1} \cdot q_0.$$

Damit können wir alles in einem Schaltwerk zusammenschalten:

