

Alles Wichtige findest Du hier:

Website: Slides, Zusatzmaterial, Links



n.ethz.ch/~ncantieni/digitech

Polybox: Slides, Zusatzmaterial



u.ethz.ch/NXMB8

ncantieni@ethz.ch

Digitaltechnik 9: Zähler

Nic Cantieni

ncantieni@ethz.ch



Persönliches

- Hallo! Ich bin Nic Cantieni, der rotierende TA.
- Tim ist nicht verloren gegangen, er kann heute nicht hier sein.
- Ihr habt ihn nächste Woche wieder zurück :)

Organisatorisches

- Normale Übungsstunde (Theorie, Tipps, Fragen)
- Folien auf n.ethz.ch/~ncantieni/digitech
- Serienabgabe: auf Moodle oder auf Papier, ich korrigiere sie dann
- Serienrückgabe: auf Moodle, oder nächste Woche durch Tim
- Study Center: heute Abend im ETF E 1, mit NuS1 zusammen

1. Recap: Automaten

2. Asynchrone Zähler

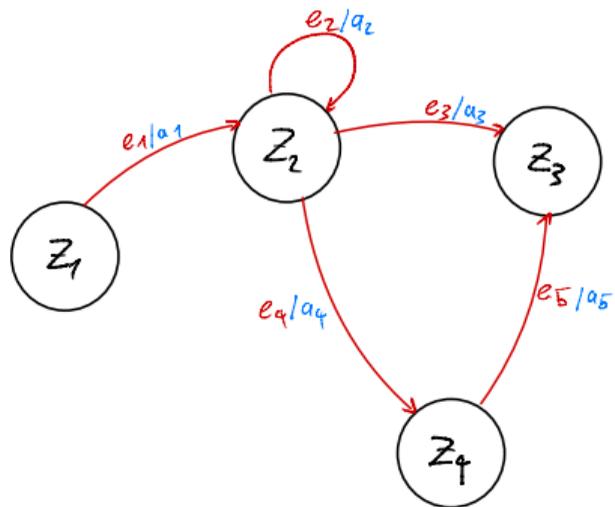
3. Set-/ Rücksetzeingang

4. Synchronzähler

Zustandsdiagramm

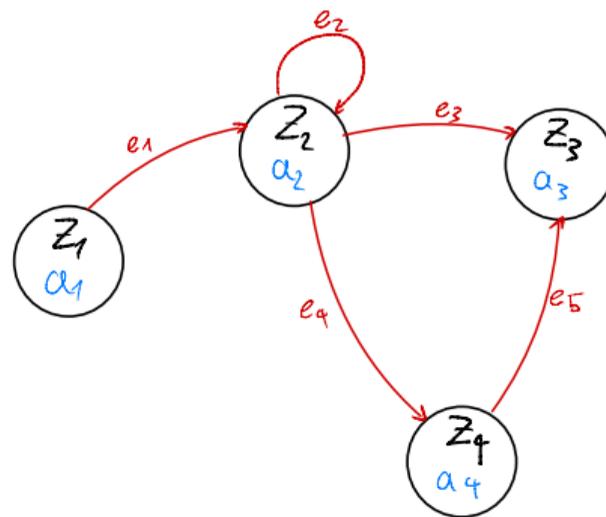
Mealy-Automat

Ausgänge: Kanten

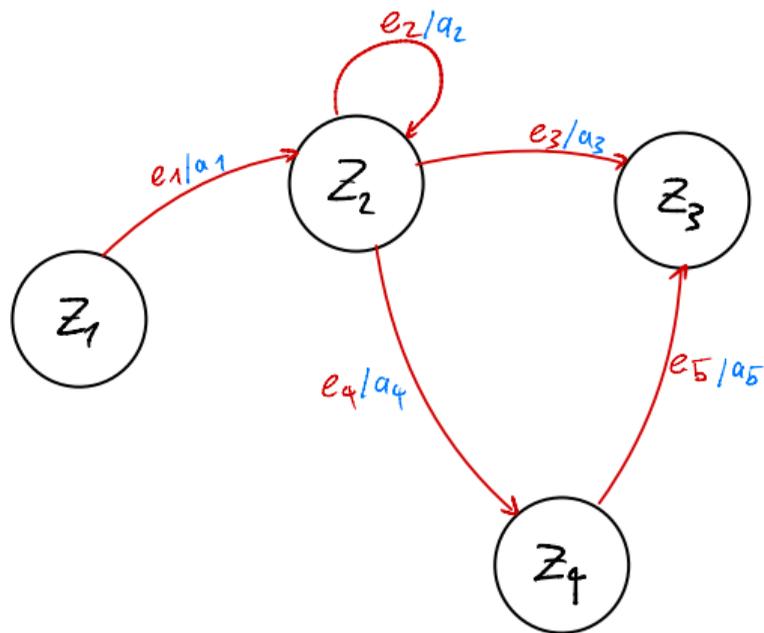


Moore-Automat

Ausgänge: Knoten



Beispiel Zustandsdiagramm \leftrightarrow Zustandsfolgetabelle



Idx	X_n	Z_n	Z_{n+1}	Y_n
1	e_1	Z_1	Z_2	a_1
2	e_2	Z_2	Z_2	a_2
3	e_3	Z_2	Z_3	a_3
4	e_4	Z_2	Z_4	a_4
5	e_5	Z_4	Z_3	a_5

Umwandlung Mealy \leftrightarrow Moore

- Mealy und Moore lassen sich immer ineinander umwandeln
- Moore ist ein Untertyp von Mealy: keine Umwandlung nötig
- Mealy zu Moore:
 - Einfach falls gleicher Zustand immer gleichen Ausgang produziert, unabhängig vom Eingang
 - Sonst: zusätzliche Zustände definieren

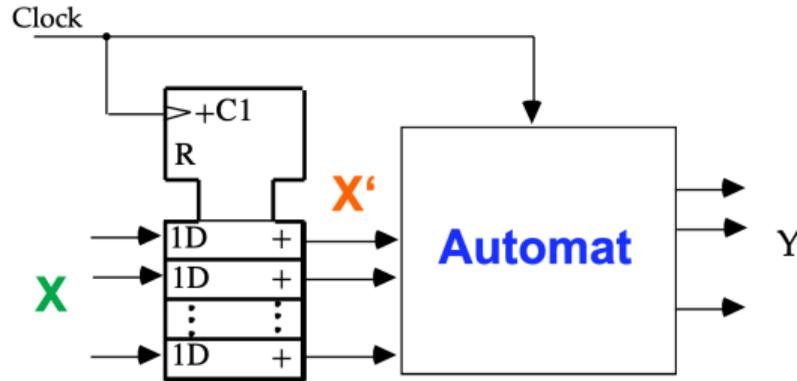
Fragen?

Automaten: Zeitliches Verhalten

- Mealy: Änderung des Eingangs wird sofort am Ausgang sichtbar
- Moore: Änderung des Eingangs wird bei nächster aktiver Taktflanke sichtbar

Synchronisierung Zustände

Eingänge werden in D-FF gespeichert und bei nächster aktiver Taktflanke eingelesen



1. Recap: Automaten

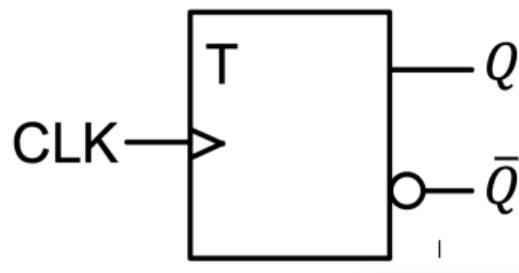
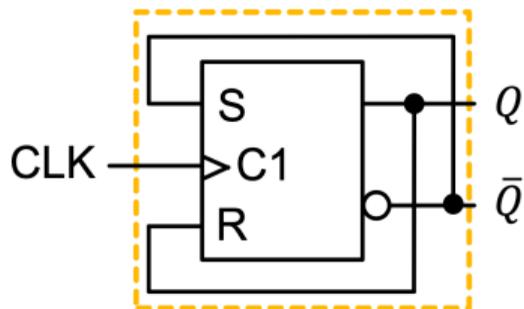
2. Asynchrnzähler

3. Set-/ Rücksetzeingang

4. Synchronzähler

T-Flipflop

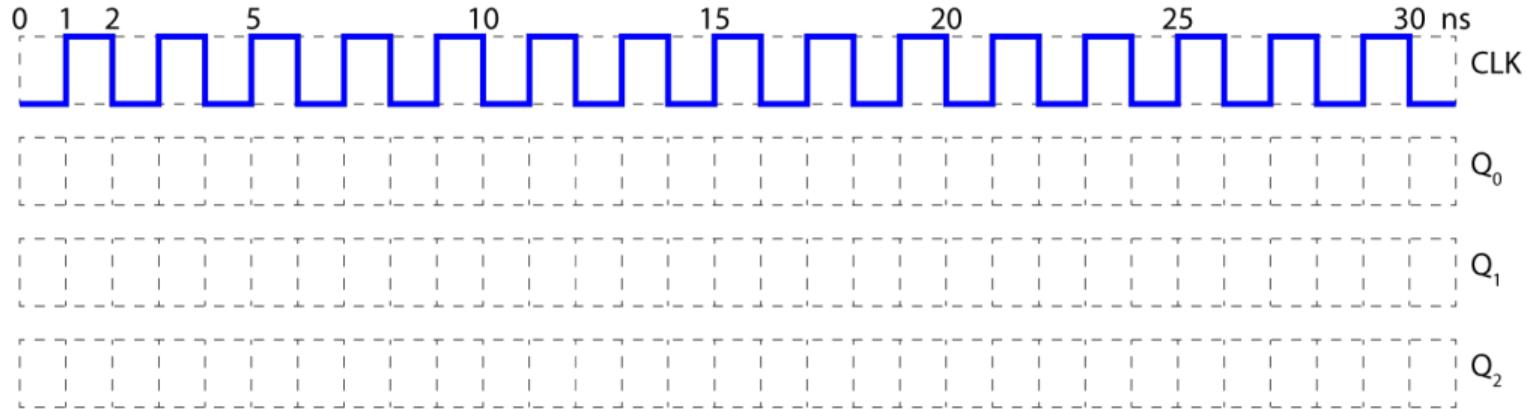
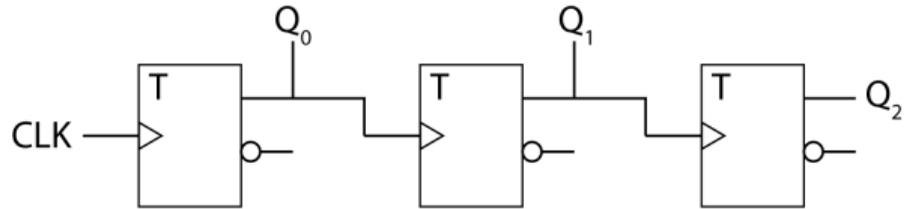
Toggle



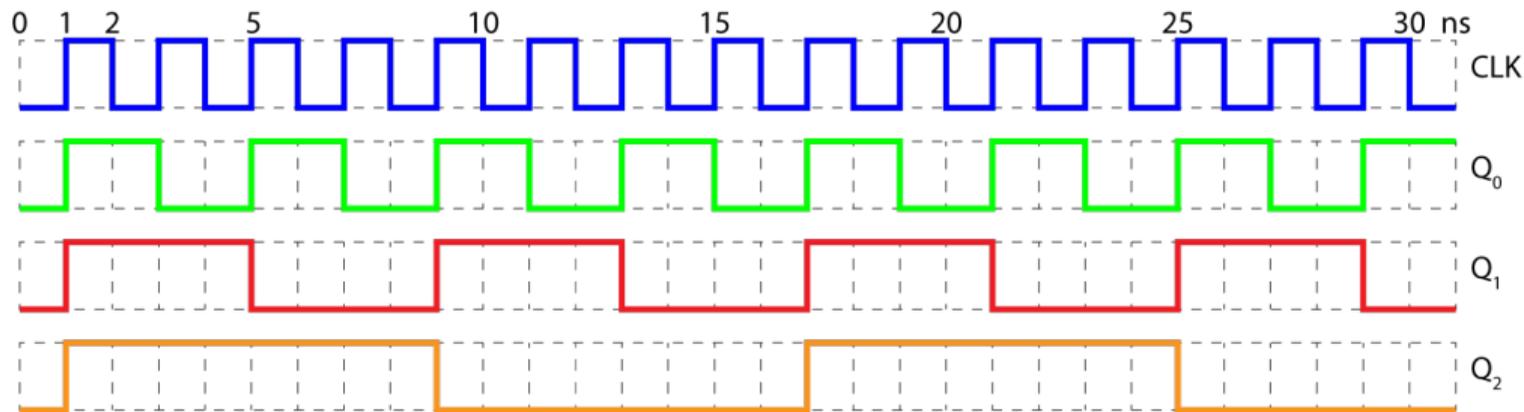
T	$Q_{1,n+1}$	$Q_{2,n+1}$	
X	$\overline{Q_{1,n}}$	$\overline{Q_{2,n}}$	wechseln

$$Q_{1,n+1} = \overline{Q_n} \text{ wenn CLK } 0 \rightarrow 1$$

Frequenzteiler / Zähler



Frequenzteiler Lösung



$$f_{CLK} = \frac{1}{2 \cdot 10^{-9} s} = 5 \cdot 10^8 \text{ Hz} = 500 \text{ MHz}$$

$$f_0 = \frac{1}{4 \cdot 10^{-9} s} = 2.5 \cdot 10^8 \text{ Hz} = 250 \text{ MHz}$$

$$f_1 = \frac{1}{8 \cdot 10^{-9} s} = 1.25 \cdot 10^8 \text{ Hz} = 125 \text{ MHz}$$

$$f_2 = \frac{1}{16 \cdot 10^{-9} s} = 6.25 \cdot 10^7 \text{ Hz} = 62.5 \text{ MHz}$$

Asynchrnzähler

- Implementierung oben: Asynchrnzähler
 - Nachteile:
 - Verzögerung rippelt
 - Verzögerung rippelt nicht gleichmässig
 - Zustände können nicht eindeutig abgelesen werden
- Lösung: Synchronzähler

1. Recap: Automaten

2. Asynchrone Zähler

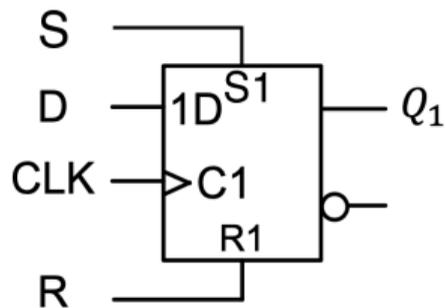
3. Set-/ Rücksetzeingang

4. Synchronzähler

Set-/ Rücksetzeingang

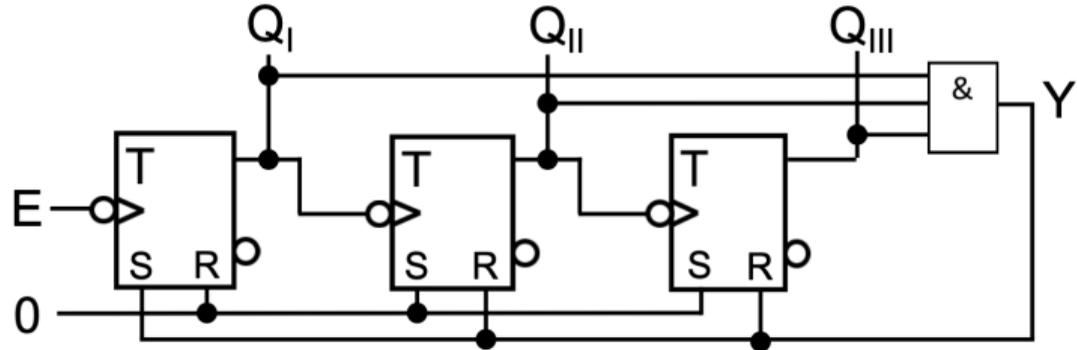
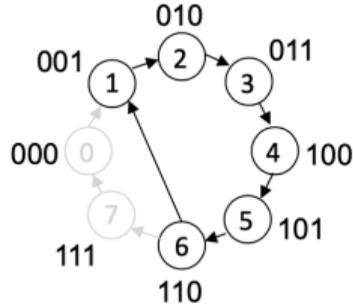
Taktunabhängiger Eingriff in Flipflop-Zustand

- Asynchroner (taktunabhängiger) Eingang
- Bestimmt Zustand sofort
- R-Eingang \rightarrow Zustand = 0
- S-Eingang \rightarrow Zustand = 1



Anwendung SR-Eingang

- Modulo-Zähler (zählt von n_1 bis n_2 und springt nach Erreichen von n_2 zurück zu n_1)
- Umsetzung: Erkennen des Zustandes $n_2 + 1$ mit einem Schaltnetz, dann setzen der SR-Eingänge wie benötigt.



1. Recap: Automaten

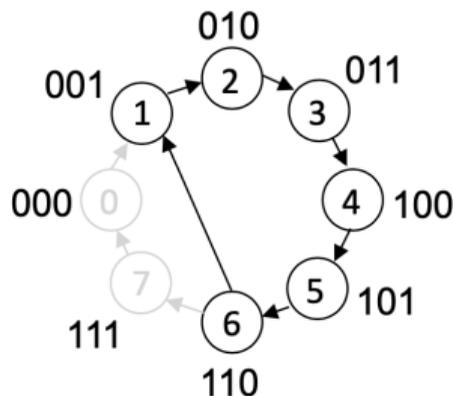
2. Asynchrone Zähler

3. Set-/ Rücksetzeingang

4. Synchronzähler

Synchronzähler

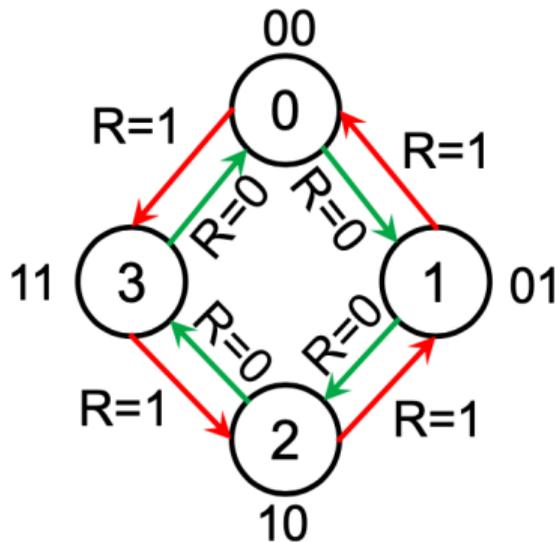
- Synchronzähler sind im Prinzip Medwedjew-Automaten
- Konstruktion wie alle Automaten: Zustandsdiagramm \rightarrow Zustandsfolgetabelle \rightarrow Schaltnetz



$Q_{3,n}$	$Q_{2,n}$	$Q_{1,n}$	$Q_{3,n+1}$	$Q_{2,n+1}$	$Q_{1,n+1}$
0	0	0	X	X	X
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	0	0	1
1	1	1	X	X	X

Reversibler Zähler

- Basiert auf Synchronzähler, mit Zustatzeingang für Richtungswahl



R=0: Vorwärts
R=1: Rückwärts

R	Q_{2n}	Q_{1n}	Q_{2n+1}	Q_{1n+1}
0	0	0	0	1
0	0	1	1	0
0	1	0	1	1
0	1	1	0	0
1	0	0	1	1
1	0	1	0	0
1	1	0	0	1
1	1	1	1	0

Vorwärts
Rückwärts

Hints Serie 9

- Überall: Sauberes Zustandsdiagramm und/oder Zustandsfolgetabelle

Prüfungsaufgabe: Asynchrone Zähler

Teilaufgabe 3: Asynchrone Zähler mit T-Flipflops

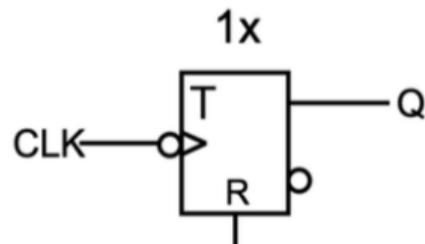
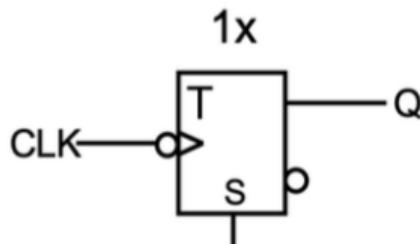
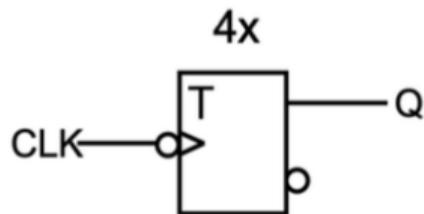
Das Ziel dieser Teilaufgabe ist, einen asynchronen Binärzähler zu entwerfen, der in der Regel von 0 bis 15 zählt ausser wenn ein Eingangssignal $IN=1$ ist. In diesem Fall muss er bei 5 neu starten, sobald er 12 erreicht. Alle Zahlen sind als Dualzahlen kodiert. Ihnen stehen 1 T-Flipflop mit einem asynchronen Setz (S) Eingang, 1 T-Flipflop mit einem asynchronen Rücksetz (R) Eingang sowie 4 normale T-Flipflops ohne zusätzliche asynchrone Eingänge zur Verfügung. Diese Bauelemente sind in Abbildung D3 gezeigt.

Beim T-Flipflop mit dem asynchronen Setz Eingang wird der Ausgang Q sofort 1 wenn $S=1$, unabhängig vom Taktsignal. Beim T-Flipflop mit dem asynchronen Rücksetz Eingang wird der Ausgang Q sofort 0 wenn $R=1$, auch unabhängig vom Taktsignal. Wenn der zu entwerfende Zähler 12 erreicht und $IN=1$, dann muss er sofort bei 5 neu anfangen. Sonst ($IN=0$) zählt er von 0 bis 15 und fängt dann wieder bei 0 an.

Wenn Sie N Flipflops benötigen, um diesen Zähler aufzubauen, dann sollten die Flipflop Ausgänge Q_0 bis Q_{N-1} benannt werden, wobei Q_0 dem 'least significant bit' entspricht und Q_{N-1} dem 'most significant bit'.

Prüfungsaufgabe: Asynchrone Zähler

- Wie viele T Flipflops sind nötig, um diesen Zähler zu realisieren? (1 Punkt)
- Wie viele nicht verwendete Zustände ergeben sich daraus? Begründen Sie Ihre Aussage. (1 Punkt)
- Was ist die logische Bedingung, damit der Zähler in einen anderen Zustand wechselt, wenn er 12 erreicht? Drücken Sie diese Bedingung als Funktion von den Q -Ausgängen und vom **IN** Signal aus. (1 Punkt)
- Zeichnen Sie diesen Zähler auf dem Lösungsblatt. Benützen Sie die T-Flipflops aus Abbildung D3 sowie NICHT, UND und ODER Gatter mit einer beliebigen Anzahl an Eingängen. (4 Punkte)



Teilaufgabe 3: Asynchrone Zähler mit T-Flipflops

6. Anzahl an T-Flipflops: 4 Flipflops, da 16 Zustände

(Korrektur: richtig oder falsch)

7. Anzahl an nicht verwendeten Zuständen: 0. Alle werden verwendet, wenn $IN=0$

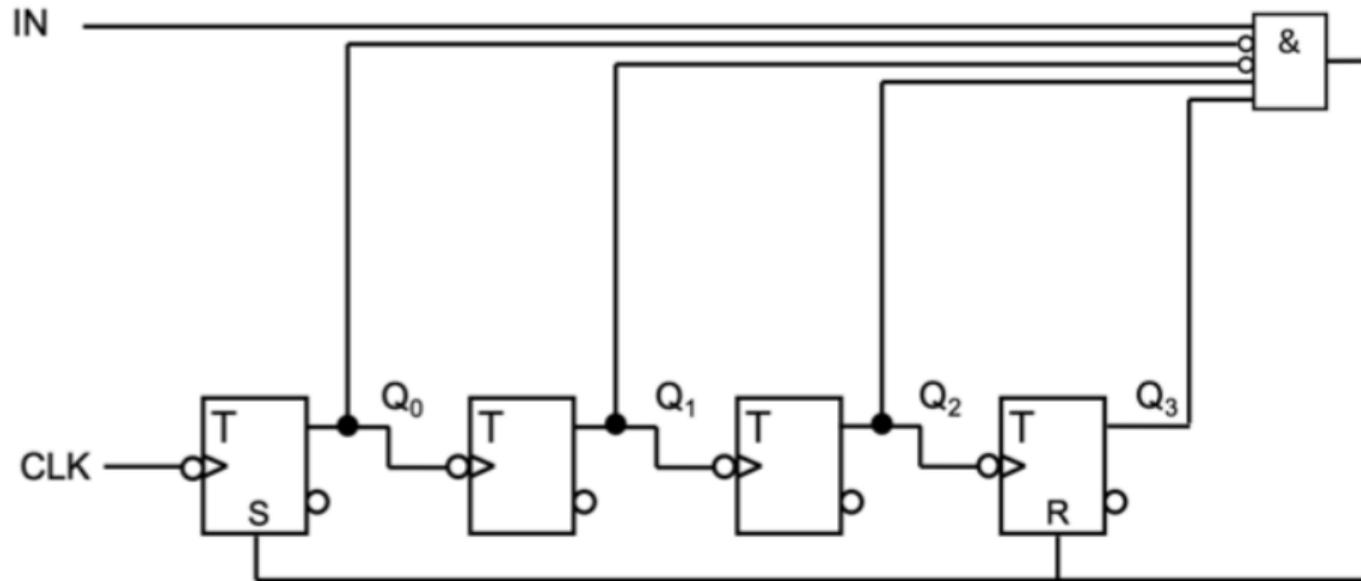
(Korrektur: richtig oder falsch)

8. $IN \wedge \overline{Q_0} \wedge \overline{Q_1} \wedge Q_2 \wedge Q_3 = 1$

(Korrektur: richtig oder falsch)

9. Zähler Schaltung:

Prüfungsaufgabe: Asynchrone Zähler



Alles Wichtige findest Du hier:

Website: Slides, Zusatzmaterial, Links



n.ethz.ch/~ncantieni/digitech

Polybox: Slides, Zusatzmaterial



u.ethz.ch/NXMB8

ncantieni@ethz.ch