

Digitaltechnik & Rechnersysteme

Automatenentwurf

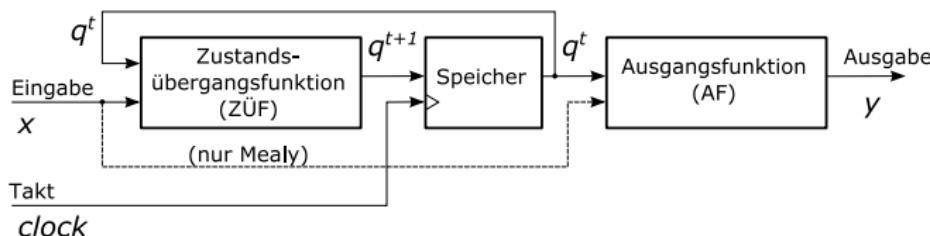
Martin Kumm



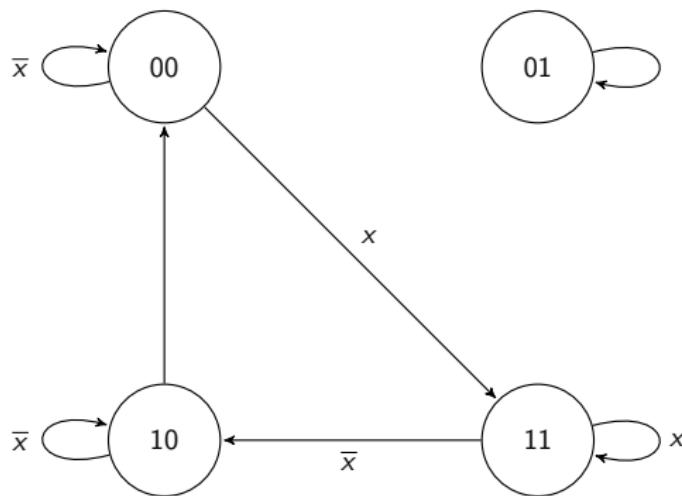
WiSe 2022/2023

Was bisher geschah...

- ① Taktpiegelgesteuertes D-Latch
- ② Taktflankengesteuertes D-Flipflop
- ③ JK-Flipflop und T-Flipflop
- ④ Synchrone Zustandsautomaten
 - Technische Realisierung komplexer Endlicher Zustandsautomaten / *finite state machines*
 - Synchrone Speicher in Rückkopplung ermöglichen beliebige Zustandswechsel
 - In der Regel D-Flipflops
 - Automatensynthese



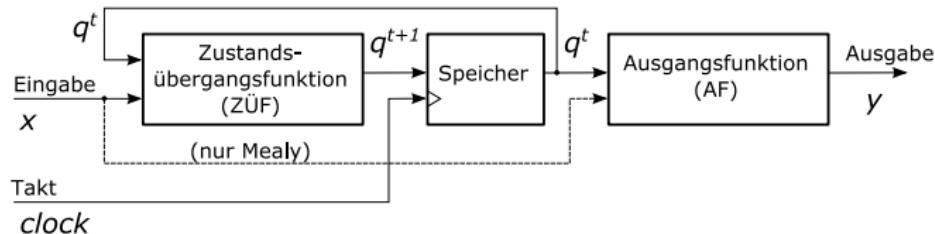
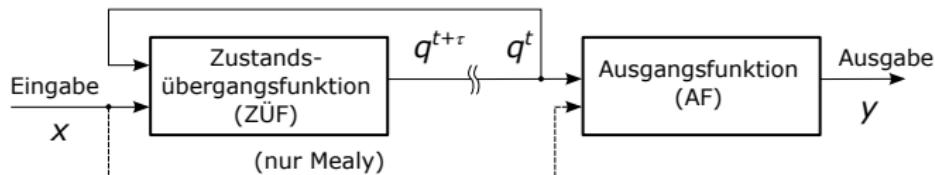
Beispiel



Geht der Automat von $00 \rightarrow 11$ über 01 oder 10?

⇒ Der Weg über 01 führt in eine Sackgasse!

Asynchroner vs. synchroner Automat



Inhalte

1 Wrap-Up

2 Automatensynthese

3 Beispielautomat: Zähler

Automatensynthese



Ausgehend von einer Problembeschreibung soll ein synchroner Automat entworfen werden, das diese Beschreibung realisiert. Ausgegangen wird meist von einer verbalen Aufgabenstellung.

Ablauf:

- ① Festlegung der Ein- und Ausgangsvariablen
- ② Festlegung ob Moore/Mealy-Automat
- ③ Bestimmung der Anzahl der Zustände
- ④ Ermittlung des Zustandsgraphen
- ⑤ Zustandskodierung
- ⑥ Bestimmung Zustandsübergangs- und Ausgangstabelle
- ⑦ Ermittlung (minimaler) Übergangs- und Ausgangsfunktionen
- ⑧ Erstellung des Schaltbildes aus Gattern und FFs

Beispiel: Eins-Detektor



Entwickeln Sie einen Automaten, der in einer Sequenz aus Binärziffern den Wechsel von einer Null auf eine Eins erkennt und daraufhin für einen Takt eine Eins ausgibt. Bei mehreren Einsen, die aufeinander folgen, soll der Automat lediglich für die erste Eins eine Eins ausgeben.

Beispiel:

Eingabe	0	0	0	1	0	0	1	1	0	1	0	1	1	1	1	0	0	0	0	1	0	0	1	1
Ausgabe	0	0	1	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0

Zustandsvariablen und -kodierung

Der Zustand lässt sich durch die Zustandsvariablen q_i beschreiben:

$$q^t = (q_0, q_1, \dots, q_i, \dots, q_{n-1}), \quad q^t \in B^n$$

Mindestanzahl erforderlicher Zustandsvariablen bei k Zuständen ist $n \geq \log_2 k$, n ganzzahlig.

Die Zuordnung der Zustände zu den Werten der Zustandsvariablen bezeichnet man als **Zustandskodierung**.

Die Zustandskodierung kann prinzipiell eine **beliebige** aber **eindeutige** Kodierung sein

Die Zustandskodierung hat Einfluss auf Komplexität und Laufzeit.

Zustandskodierung

Binärcodierung

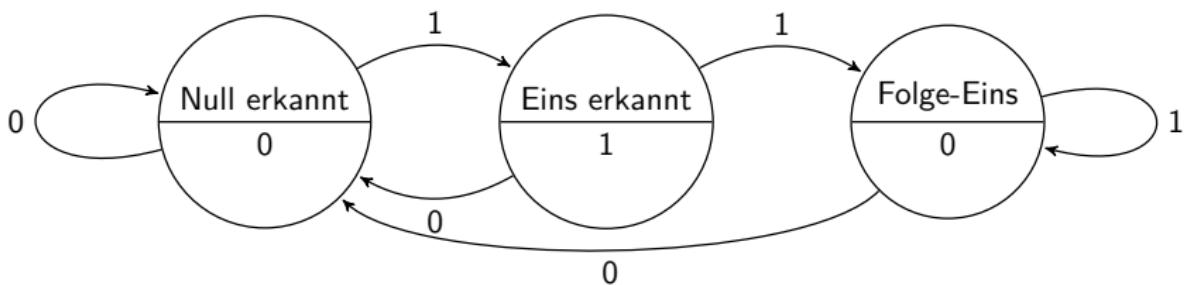
- Jedem Zustand wird eine **Binärzahl** zugeordnet
- Kompakte Kodierung, erfordert wenige Flipflops
- Beispiel für 4 Zustände: 00, 01, 10, 11

One-Hot Codierung

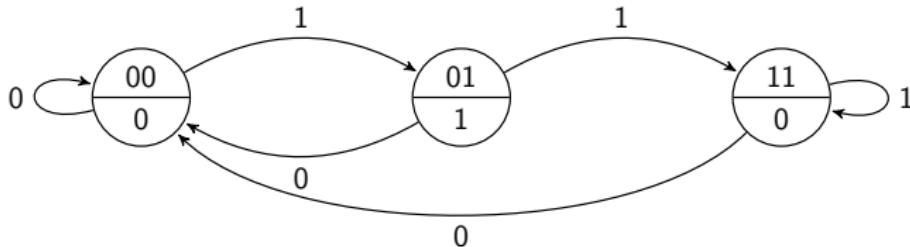
- Jedem Zustand wird ein **Zustandsbit** zugeordnet
- Zu jedem Zeitpunkt ist immer genau ein Zustandsbit »1«.
- Beispiel für 4 Zustände: 0001, 0010, 0100, 1000
- Benötigt mehr Flipflops
- Zustandsübergangs- und Ausgangsfunktionen vereinfachen sich oft

Zustandscodierung

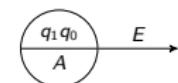
Zustandsdiagramm:



Zustandsdiagramm mit Zustandscodierung:

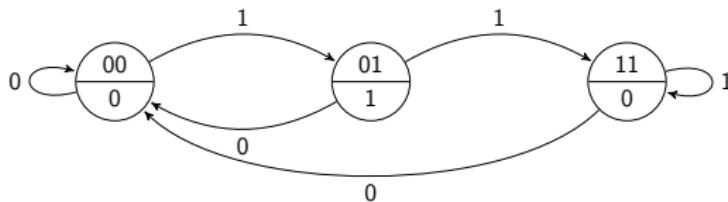


Notation:



Eins-Detektor als Moore-Automat

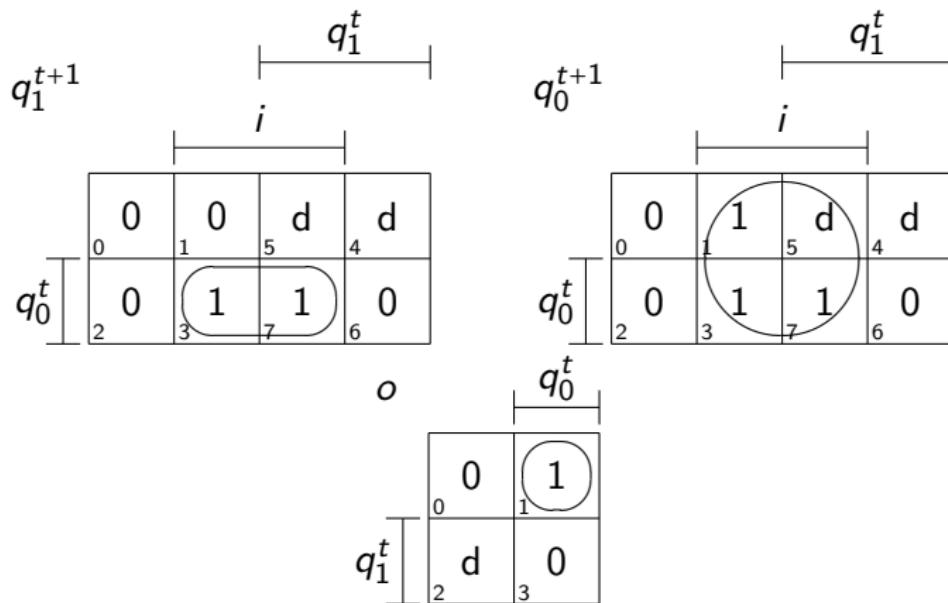
Vorlesungsaufgabe: Bestimmen Sie die Zustandsübergangs- und Ausgangstabelle



q_1^t	q_0^t	i	q_1^{t+1}	q_0^{t+1}	q_1^t	q_0^t	o
0	0	0	0	0	0	0	
0	0	1	0	1	0	1	
0	1	0	1	0	1	0	
0	1	1	1	1	1	1	
1	0	0	1	0	1	0	
1	0	1	0	1	1	1	
1	1	0	0	1	0	1	
1	1	1	1	1	1	1	

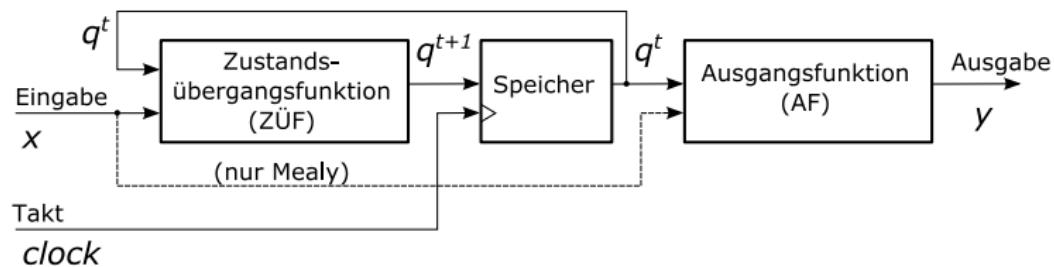
Eins-Detektor als Moore-Automat

Minimierung Zustandsübergangs- und Ausgangsfunktionen:



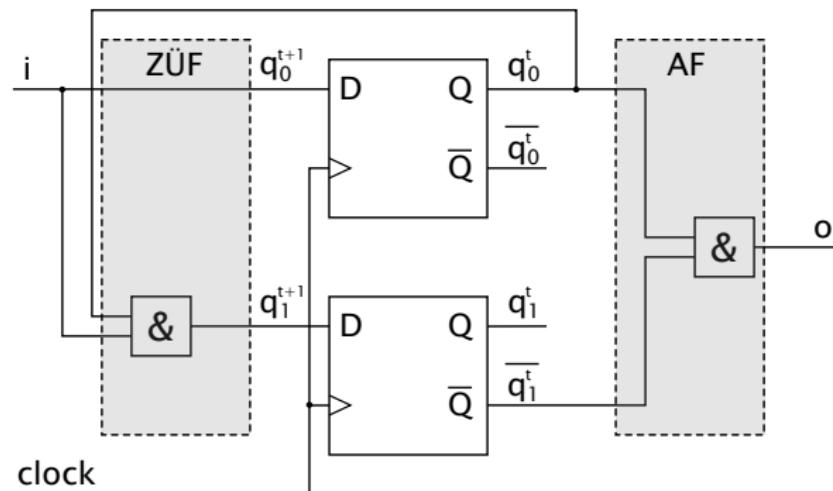
$$q_1^{t+1} = q_0^t i, \quad q_0^{t+1} = i, \quad o = \overline{q_1^t} q_0^t$$

Wdh.: Synchroner Zustandsautomat



Eins-Detektor als Moore-Automat

Schaltbild:

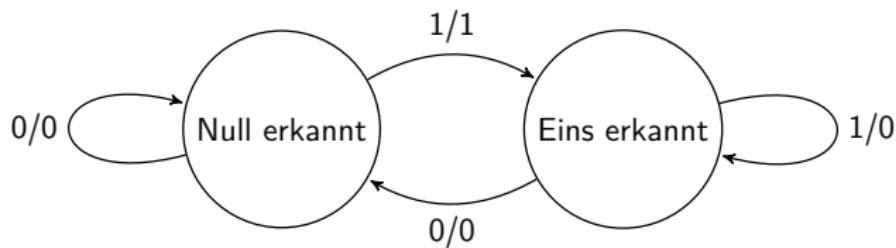


ZÜF: Zustandsübergangsfunktion

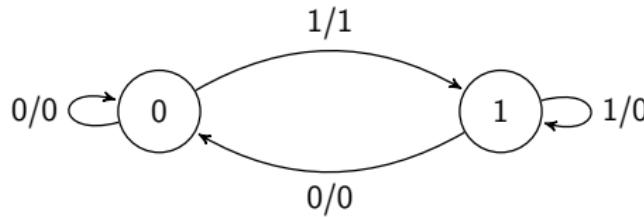
AF: Ausgangsfunktion

Eins-Detektor als Mealy-Automat

Zustandsdiagramm:



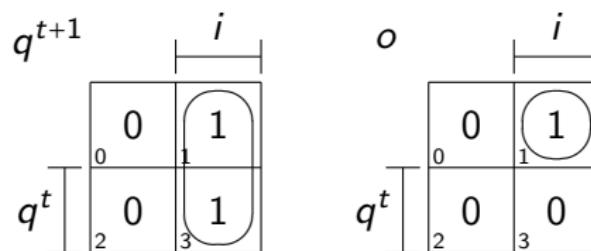
Zustandsdiagramm mit Zustandscodierung:



Eins-Detektor als Mealy-Automat

Zustandsübergangs- und Ausgangstabelle + Minimierung

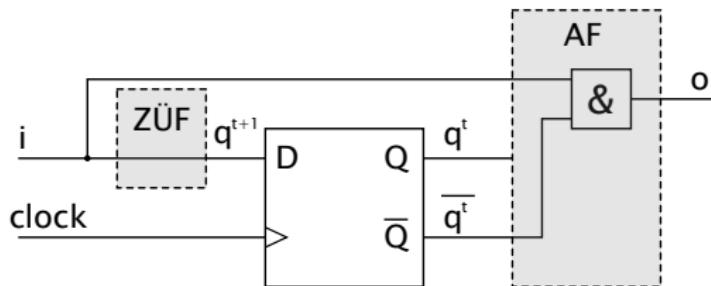
q^t	i	q^{t+1}	o
0	0	0	0
0	1	1	1
1	0	0	0
1	1	1	0



$$q^{t+1} = i, o = \overline{q^t} i$$

Eins-Detektor als Mealy-Automat

Schaltbild:



ZÜF: Zustandsübergangsfunktion

AF: Ausgangsfunktion

Anmerkungen

Die Automatensynthese bietet ein mächtiges Werkzeug um Systeme mit Gedächtnis zu entwerfen

Die Automatensynthese ist ein kreativer Prozess. D.h. oft sind mehrere Iterationen notwendig um zur finalen Lösung zu kommen

Moore- und Mealy-Automaten führen je zu anderen Lösungen

Moore-Automaten benötigen oft mehr Zustände als Mealy-Automaten

Mod-4 Zähler mit enable



Im Folgenden entwickeln wir zusammen einen sog. Modulo-4 (Mod-4) Zähler mit enable.

Dieser soll immer wenn $e = 1$ ist die Ausgabe Y um eins erhöhen.

Nach $Y = 3$ fängt dieser wieder von vorne an ($Y = 4 \bmod 4 = 0$)

D.h. $Y=0,1,2,3,0,1,2,3,\dots$

Mod-4 Zähler mit enable



Ablauf:

- ① Festlegung der Ein- und Ausgangsvariablen ✓
 - Eingabe: e ,
 - Ausgabe: Y mit Werten zwischen 0 und 3
 - Binärcodiert, 2 Bit $\rightarrow y_1, y_0$
- ② Festlegung ob Moore/Mealy-Automat ✓
 - Moore-Automat (aus Aufgabenstellung)
- ③ Bestimmung der Anzahl der Zustände ✓
 - 4 Zustände (\rightarrow min. 2 Zustandsbits)
- ④ Ermittlung des Zustandsgraphen

Mod-4 Zähler mit enable

Vorlesungsaufgabe: Vervollständigen Sie das folgende Zustandsdiagramm zu einem mod-4 Zähler.



Notation:



Mod-4 Zähler mit enable

Ablauf:

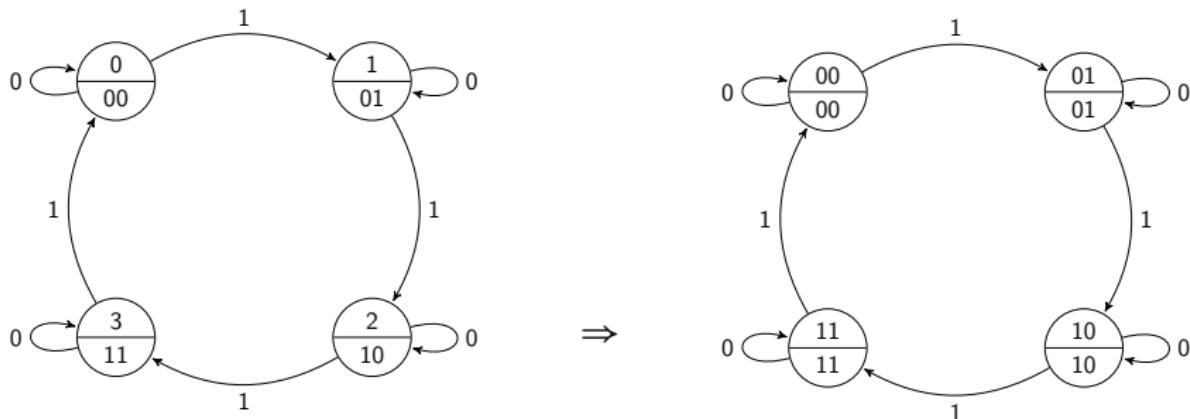
- ① Festlegung der Ein- und Ausgangsvariablen ✓
- ② Festlegung ob Moore/Mealy-Automat ✓
- ③ Bestimmung der Anzahl der Zustände ✓
- ④ Ermittlung des Zustandsgraphen ✓
- ⑤ Zustandskodierung

Prinzipiell beliebig, da hier jede Ausgabe eindeutig ist kann
Zustandskodierung = Ausgangskodierung gewählt werden:

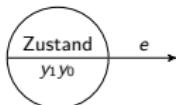
$$\begin{aligned}q_0 &= y_0 \\q_1 &= y_1\end{aligned}$$

Mod-4 Zähler mit enable

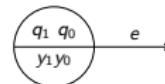
Zustandskodierung:



Notation:



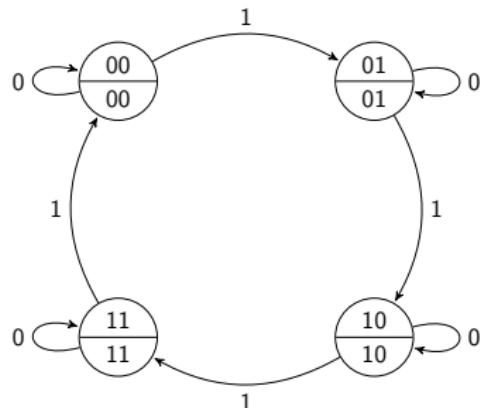
Notation:



Mod-4 Zähler mit enable

- ① Festlegung der Ein- und Ausgangsvariablen ✓
- ② Festlegung ob Moore/Mealy-Automat ✓
- ③ Bestimmung der Anzahl der Zustände ✓
- ④ Ermittlung des Zustandsgraphen ✓
- ⑤ Zustandskodierung ✓
- ⑥ Bestimmung Zustandsübergangs- und Ausgangstabelle

Mod-4 Zähler mit enable



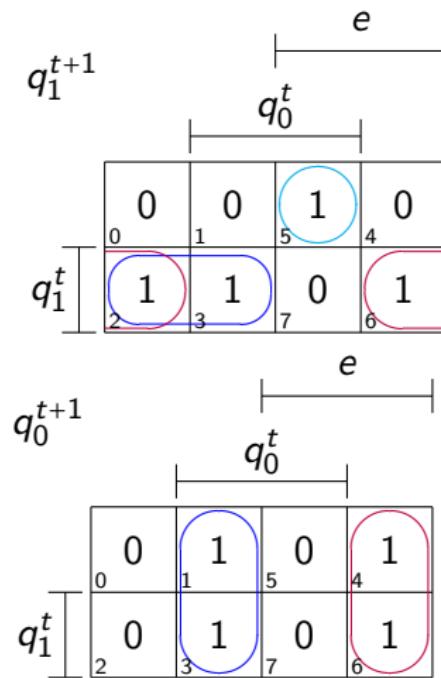
e	q_1^t	q_0^t	q_1^{t+1}	q_0^{t+1}
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	1
1	0	0	0	1
1	0	1	1	0
1	1	0	1	1
1	1	1	0	0

Mod-4 Zähler mit enable

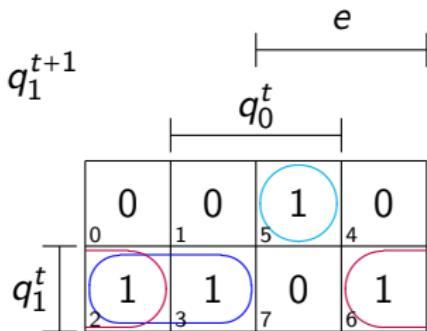
- ① Festlegung der Ein- und Ausgangsvariablen ✓
- ② Festlegung ob Moore/Mealy-Automat ✓
- ③ Bestimmung der Anzahl der Zustände ✓
- ④ Ermittlung des Zustandsgraphen ✓
- ⑤ Zustandskodierung ✓
- ⑥ Bestimmung Zustandsübergangs- und Ausgangstabelle ✓
- ⑦ Ermittlung (minimaler) Übergangs- und Ausgangsfunktionen

Mod-4 Zähler mit enable

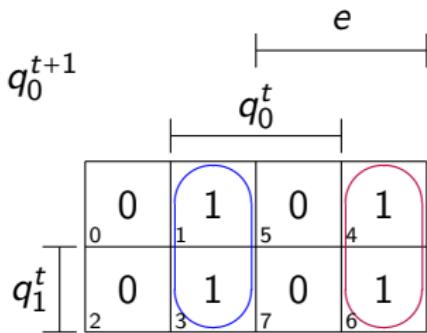
e	q_1^t	q_0^t	q_1^{t+1}	q_0^{t+1}
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	1	1
1	0	0	0	1
1	0	1	1	0
1	1	0	1	1
1	1	1	0	0



Mod-4 Zähler mit enable



$$q_1^{t+1} = \bar{e} q_1^t + e q_0^t \overline{q_1^t} + \overline{q_0^t} q_1^t$$

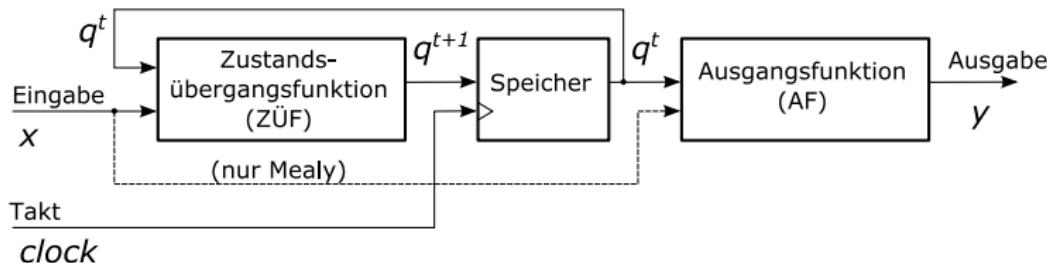


$$q_0^{t+1} = \bar{e} q_0^t + e \overline{q_0^t}$$

Mod-4 Zähler mit enable



- ① Festlegung der Ein- und Ausgangsvariablen ✓
- ② Festlegung ob Moore/Mealy-Automat ✓
- ③ Bestimmung der Anzahl der Zustände ✓
- ④ Ermittlung des Zustandsgraphen ✓
- ⑤ Zustandskodierung ✓
- ⑥ Bestimmung Zustandsübergangs- und Ausgangstabelle ✓
- ⑦ Ermittlung Übergangs- und Ausgangsfunktionen ✓
- ⑧ Erstellung des Schaltbildes aus Gattern und FFs



Mod-4 Zähler mit enable

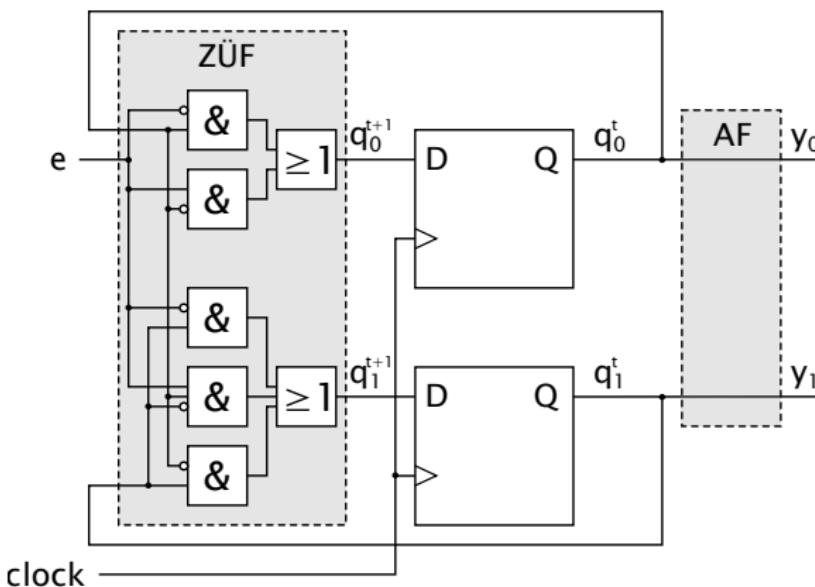
$$\text{ZÜF: } q_0^{t+1} = \bar{e} q_0^t + e \overline{q_0^t}$$

$$q_1^{t+1} = \bar{e} q_1^t + e q_0^t \overline{q_1^t} + \overline{q_0^t} q_1^t$$

AF:

$$y_0 = q_0^t$$

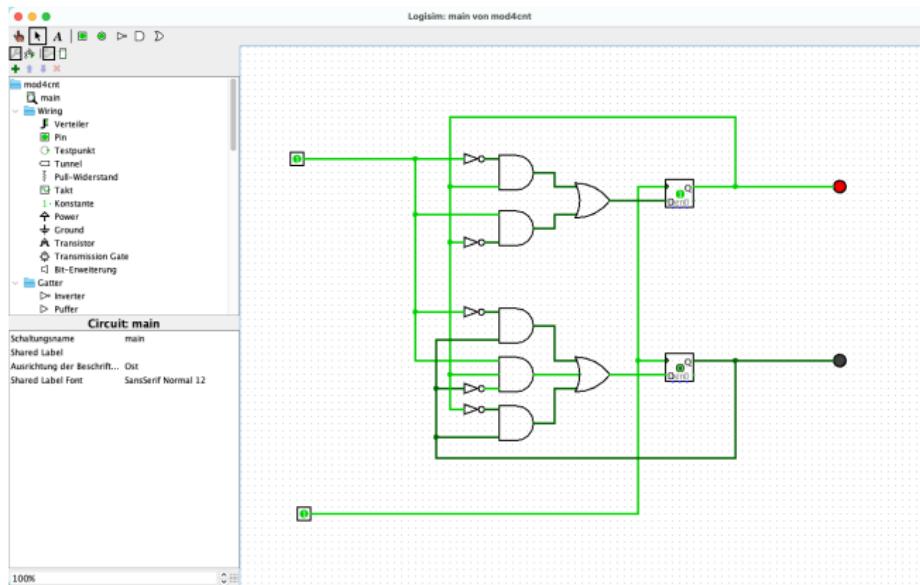
$$y_1 = q_1^t$$



Simulation: Mod-4 Zähler mit enable

AI

Bei Automaten bietet sich die Simulation an um den Ablauf besser zu verstehen:





Frohe
Weihnachten