

Digitaltechnik & Rechnersysteme

Latches und Flipflops

Martin Kumm



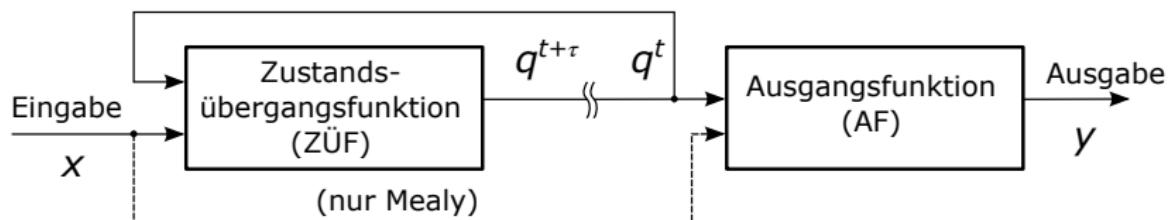
WiSe 2022/2023

Was bisher geschah...

- Arithmetisch-logische Einheit / *Arithmetic Logic Unit* (ALU)
 - Fasst arithmetische und logische Operationen in einer Einheit zusammen
 - Das Rechenwerk eines Prozessors
- Schaltwerke (asynchrone Automaten)
 - Schaltwerke = Schaltnetze mit Rückkopplungen
 - Dadurch interner Zustand
 - Ermöglicht Speicherverhalten
 - Beschreibung zur Zustandsübergangs- und Ausgangsfunktion
 - Beschreibung zur Zustandsübergangs- und Ausgangstabelle
 - Darstellung durch Zustandsübergangsdiagramm
 - Unterscheidung Moore und Mealy Automaten
- Schaltwerkanalyse
 - Analyse durch Auftrennen der Rückkopplungen

Schaltwerke

Allgemeines Blockschaltbild eines Schaltwerks:



MEALY-Automat: Allgemeine Form

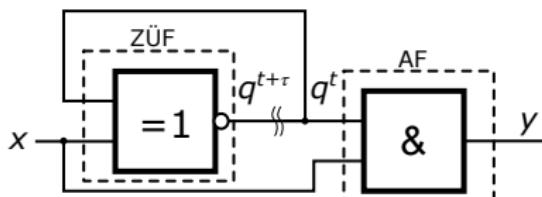
MOORE-Automat: AF hängt nur von aktuellem Zustand ab.

Analyse: **Rückkopplungen auftrennen**

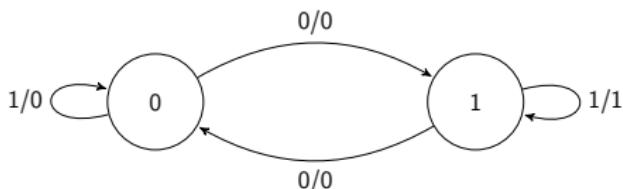
Beschreibung über:

- Zustandsübergangs- und **Ausgangstabelle/-funktion**
- **Zustandsübergangsgraph**

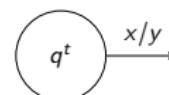
Beispiel (aus Vorlesungsaufgabe)



q^t	x	ZÜF $q^{t+\tau}$	AF y
0	0	1	0
0	1	0	0
1	0	0	0
1	1	1	1



Notation:



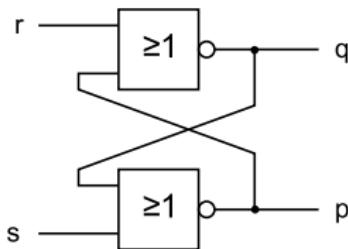
Inhalte



- 1 Wrap-Up
- 2 RS-Latch
- 3 Synchrone Speicher
- 4 Taktpegelgesteuerte Speicher
- 5 Taktflankengesteuerte Speicher
- 6 Weitere FF-Typen
 - JK-Flipflop
 - T-Flipflop
- 7 Synchrone Zustandsautomaten

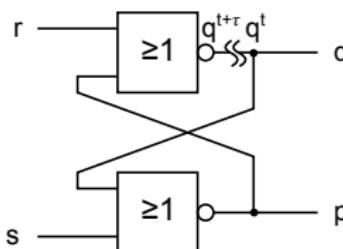
Speichern einer binären Variablen

Ein sog. RS-Latch lässt sich aus zwei rückgekoppelten NOR-Gliedern realisieren:

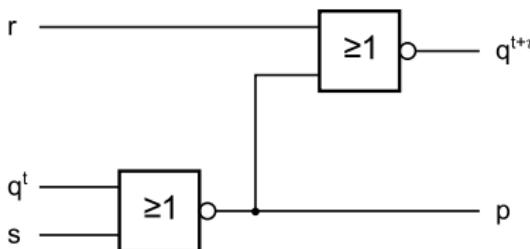


⇒ Zur Analyse werden so lange Rückkopplungen aufgetrennt bis ein Schaltnetz entsteht.

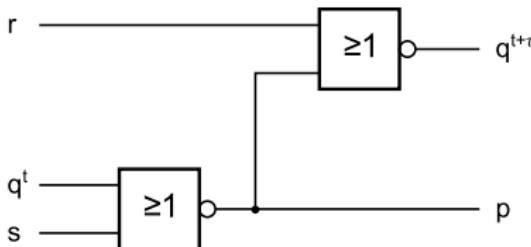
Analyse der Funktionsweise



Daraus ergibt sich ein gewöhnliches Schaltnetz:



Analyse der Funktionsweise



Funktion der Schaltung:

$$p = \overline{s + q^t} = \overline{s} \overline{q^t}$$

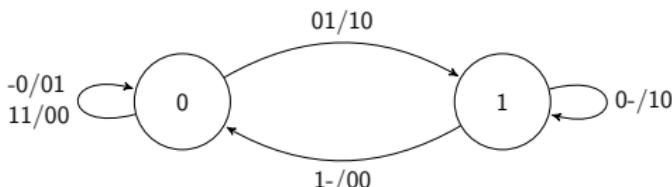
$$\begin{aligned} q^{t+\tau} &= \overline{r + \overline{(s + q^t)}} \\ &= \overline{r} (s + q^t) \\ &= \overline{r} s + \overline{r} q^t \end{aligned}$$

Zustandsübergangstabelle:

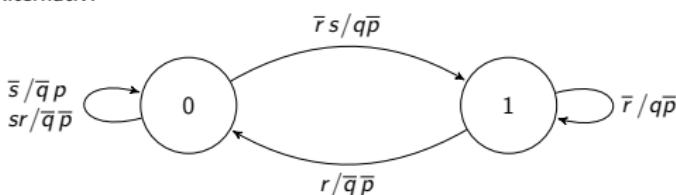
q^t	r	s	$q^{t+\tau}$	p
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	0	0
<hr/>				
1	0	0	1	0
1	0	1	1	0
1	1	0	0	0
1	1	1	0	0

Zustandsdiagramm des RS-Latch

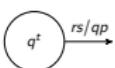
q^t	r	s	$q^{t+\tau}$	p
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	0	0
<hr/>				
1	0	0	1	0
1	0	1	1	0
1	1	0	0	0
1	1	1	0	0



Alternativ:



Notation:



Hinweis: Hier steht >>-<< für *don't care*.

Beispiel: Eine Transition mit $rs = 0$ wird aktiv sobald $s = 0$ ist, unabhängig von r . Sie reagiert also auf $rs = 00$ und $rs = 10$.

Verhalten des RS-Latch

	q^t	r	s	$q^{t+\tau}$	p	Bemerkung
0	0	0	0	0	1	speichern (0)
1	0	0	1	1	0	setzen
2	0	1	0	0	1	rücksetzen
3	0	1	1	0	0	(rücksetzen)
4	1	0	0	1	0	speichern (1)
5	1	0	1	1	0	setzen
6	1	1	0	0	0	rücksetzen
7	1	1	1	0	0	instabil

- Speicherung: stabile Zustände, mit $q^{t+\tau} = q^t \Rightarrow$ Zeilen 0, 4
- Setzen: mit $s = 1 \rightarrow q = 1 \Rightarrow$ Zeilen 1 und 5
- Rücksetzen: mit $r = 1 \rightarrow q = 0 \Rightarrow$ Zeilen 2 und 6

Übergangstabelle RS-Latch

Übergangstabelle (verkürzt):

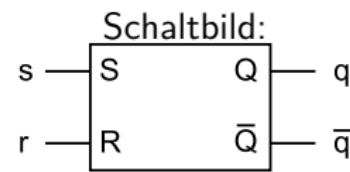
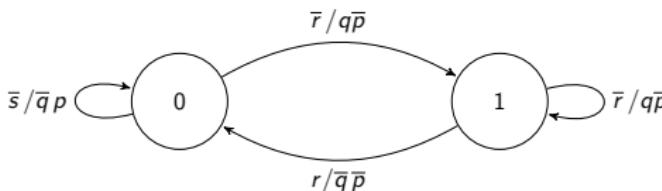
r	s	$q^{t+\tau}$	
0	0	q^t	speichern
0	1	1	setzen
1	0	0	rücksetzen
1	1	-	nicht zulässig

$r = \text{reset}$, $s = \text{set}$

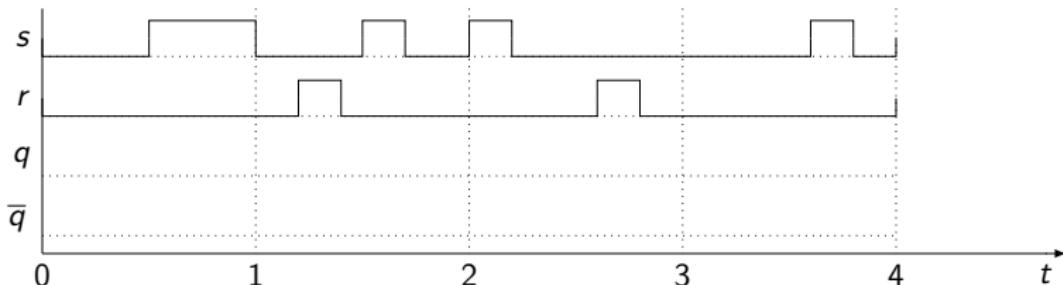
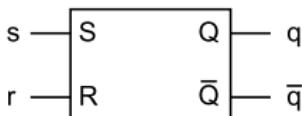
Da $p = \bar{q}$ für stabile Zustände gilt, wird der 2. Ausgang als \bar{q} bezeichnet

Das RS-Latch ist ein Element mit **zwei stabilen** Zuständen, auch **bistabiler** Speicher genannt.

Wegen möglicher Instabilität ist $r = s = 1$ meist verboten!



RS-Latch Timing



Vorlesungsaufgabe: Ermitteln Sie das Timing-Diagramm der Signale q und q' (für $q = 0$ zum Zeitpunkt $t = 0$).

Synchrones Schaltverhalten

Problem: Jede Änderung am Eingang eines Latches wird **sofort** übernommen!

Wenn mehrere Bits gespeichert werden sollen, können Laufzeitunterschiede unterschiedliche (Zwischen-)Zustände bewirken.

Um solche Effekte zu verhindern, wird ein **Takt**signal (engl. *clock*) eingeführt, das durch regelmäßige (periodische) Wechsel zwischen 0 und 1 Zeitpunkte festlegt, zu denen Zustandswechsel erfolgen können.

Man spricht dann von **synchronem** Schaltverhalten (im Gegensatz zum bisherigen **asynchronen** Schaltverhalten).

Taktpiegel- und Taktflankensteuerung



Taktpiegelsteuerung (Taktzustandssteuerung): Eingangsvariablen und Takt werden konjunktiv verknüpft
⇒ Eingänge wirken nur, während Takt 1 ist (relativ lange).
⇒ Speicherelemente werden als **Latch** bezeichnet

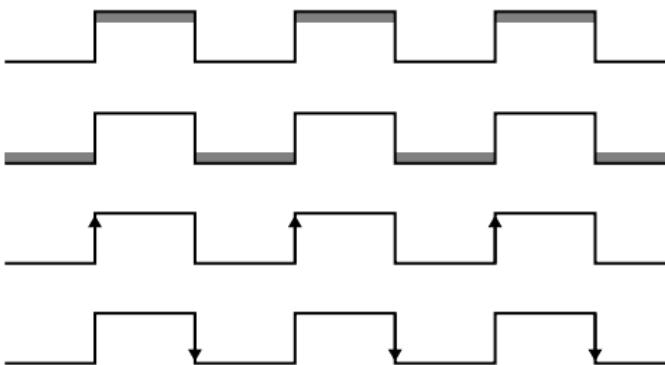
Taktflankensteuerung: Zustandsänderungen (Auswertung) nur bei den Flanken ($0 \rightarrow 1$ Wechsel oder $1 \rightarrow 0$ Wechsel) möglich.
⇒ Speicherelemente werden als **Flipflop** bezeichnet

Pegel- und Flankensteuerung

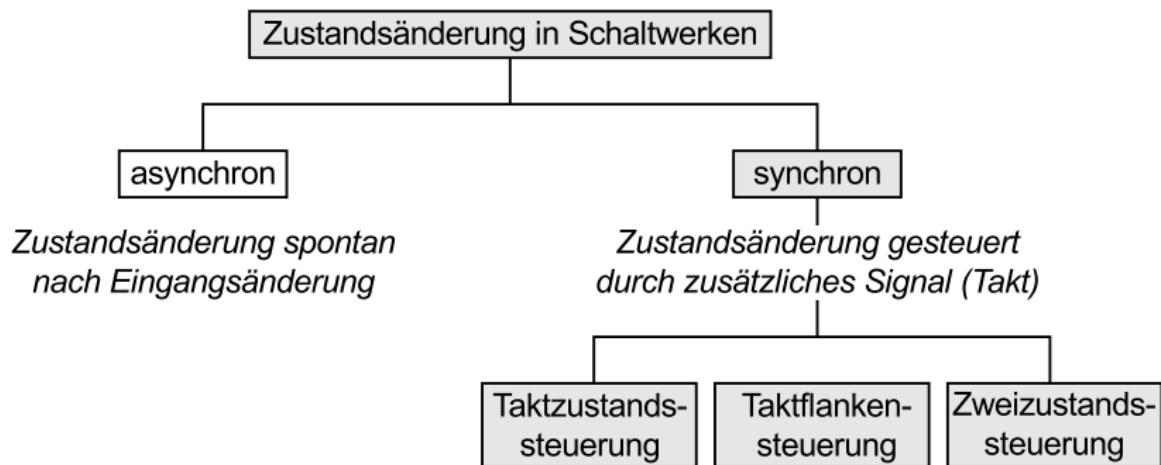
Durch Invertieren des Taktsignals kann auch eine Reaktion auf den Low-Pegel des Taktes bzw. $1 \rightarrow 0$ Taktflanke erreicht werden.

$0 \rightarrow 1$ Übergänge des Takts werden als **Vorderflanke**, **steigende Flanke** oder **positive Taktflanke** bezeichnet.

$1 \rightarrow 0$ Übergänge des Takts werden als **Rückflanke**, **fallende Flanke** oder **negative Taktflanke** bezeichnet.



Taxonomie von Schaltwerken

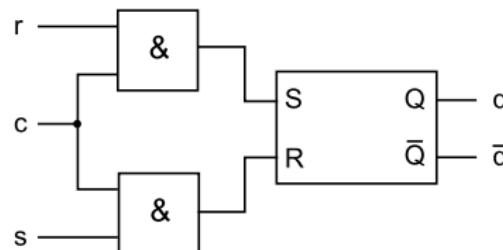


RS-Latch mit Taktpiegelsteuerung I

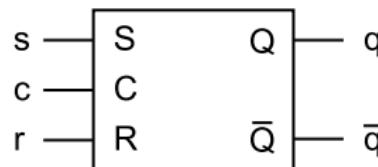
Die Eingänge r und s werden jeweils durch ein UND-Glied mit dem Takt c (für „clock“) verknüpft.

- Zusätzlicher **Takteingang** c

- UND-Verknüpfung von c mit s und r
 $\rightarrow s$ und r werden nur ausgewertet, wenn Taktimpuls vorliegt



Schaltsymbol:



RS-Latch mit Taktpegelsteuerung II

Vereinfachte Übergangstabelle:

c	r	s	q^{t+1}	
0	–	–	q^t	speichern
1	0	0	q^t	speichern
1	0	1	1	setzen
1	1	0	0	rücksetzen
1	1	1	–	nicht zulässig

⇒ Durch die Takt-Steuerung wird es unnötig, zwei Eingänge für das Setzen und Rücksetzen zu verwenden.

⇒ Setzen/Rücksetzen lässt sich aus einer Datenleitung d ableiten
 → D-Latch

D-Latch mit Taktpegelsteuerung

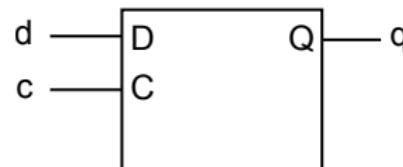
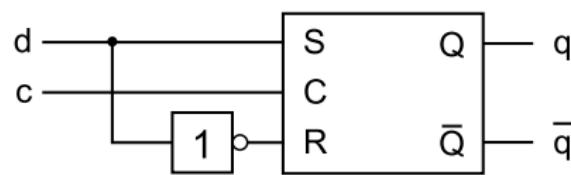
Set und reset lässt sich einem Datensignal d ableiten:

$$s = d$$

$$r = \overline{d}$$

Somit wird bei $d = 1$ das Latch mit jedem Taktpuls gesetzt, für $d = 0$ mit jedem Taktpuls zurückgesetzt.

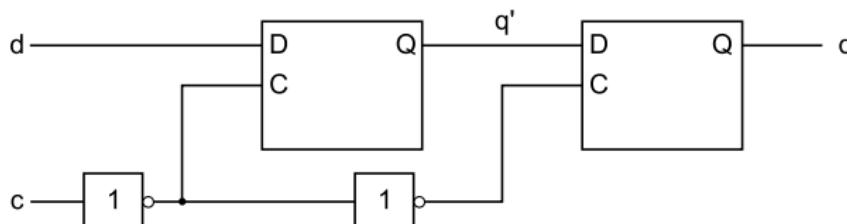
Schaltsymbol:



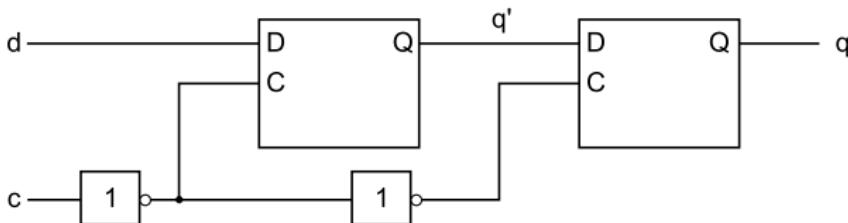
Taktflankengesteuertes D-Flipflop

Die Eigenschaft des getakteten D-Latches, während der aktiven Taktphase den Eingang transparent auf den Ausgang durchzuschalten, ist unerwünscht (sehr langer Zeitraum).

Durch Hintereinanderschalten zweier taktpiegelgesteuerter D-Latches erhält man ein flankengesteuertes D-Flipflop:

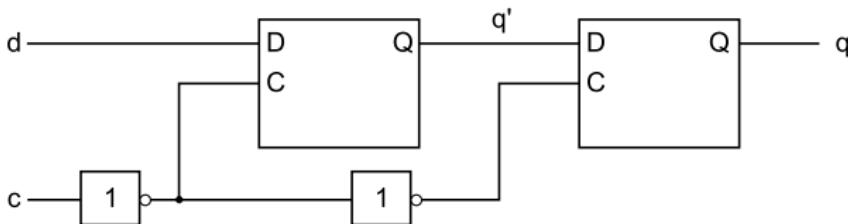


Taktflankengesteuertes D-Flipflop

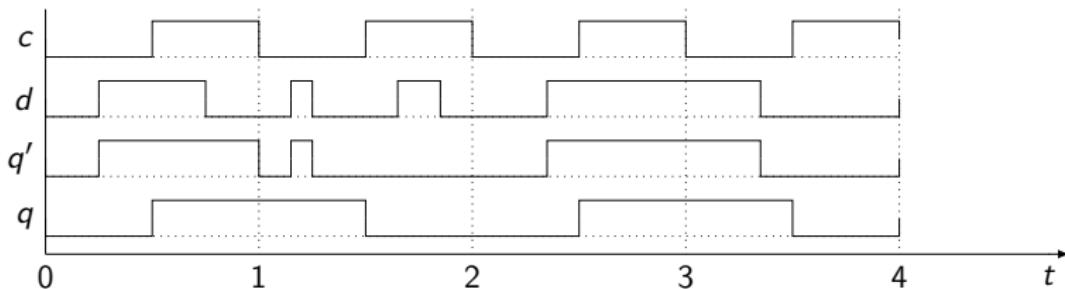


Das erste Flipflop übernimmt ein Eingangssignal während der aktiven Taktpulse, das zweite Flipflop hält zur gleichen Zeit seinen Inhalt unverändert. Beim $0 \rightarrow 1$ -Übergang des Taktsignals wird das erste Flipflop gesperrt, während das zweite dessen Inhalt übernimmt.

Taktflankengesteuertes D-Flipflop

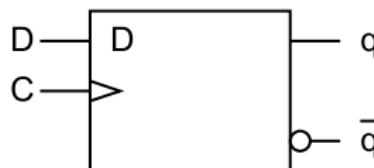


Beispiel-Timing:



Flankengesteuertes D-Flipflop

Das Schaltsymbol eines D-Flipflop sieht folgendermaßen aus:



Dabei zeigt das **unausgefüllte Dreieck** an, dass es sich um ein **positiv** flankengesteuertes Flipflop handelt. Eine Invertierung des Eingangs bzw. alternativ ein **ausgefülltes Dreieck** bezeichnen eine Steuerung mit der **negativen** Flanke.

Funktion des D-Flipflops

Übergangsfunktion: $q^{t+1} = d$ für $c = 0 \rightarrow 1$

Übergangstabelle:

d	q^{t+1}
0	0
1	1

JK-Flipflop

Beim RS-Flipflop war die Eingangskombination $r = s = 1$ nicht erlaubt.

Durch Festlegung eines definierten Verhaltens für $r = s = 1$, nämlich der Wechsel des Zustands, erhält man aus dem RS-Flipflop ein **JK-Flipflop**.

Ein Gerücht besagt, das JK-Flipflop wurde möglicherweise nach **Jack Kilby** benannt (Physik-Nobelpreis 2000 für „ *Beitrag zur Entwicklung des Integrierten Schaltkreises (IC)*“).

In Wirklichkeit beruht der Name auf einer willkürlichen „Durchnummerierung“ der Eingänge verschiedener Flipflops mit Buchstaben durch Dr. Eldred Nelson (Hughes Aircraft) um 1968.

Funktion des JK-Flipflops



Übergangstabelle:

j	k	q^{t+1}	
0	0	q^t	speichern
1	0	1	setzen
0	1	0	rücksetzen
1	1	\bar{q}^t	wechseln / toggeln

$$q^{t+1} = j\overline{q^t} + \overline{k} q^t \text{ für } c = 0 \rightarrow 1$$

Toggle-Flipflop

Ein JK-Flip-Flop mit $j = k = t$ führt zum **Toggle-Flipflop**: Für $t = 1$ ändert sich der Zustand (toggle), für $t = 0$ wird gespeichert.

Übergangstabelle:

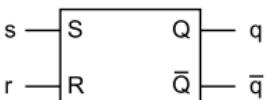
t	q^{t+1}
0	q^t
1	\bar{q}^t

speichern toggle

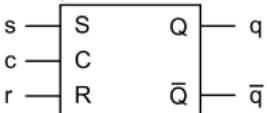
$$q^{t+1} = t\bar{q}^t + \bar{t}q^t \text{ für } c = 0 \rightarrow 1$$

Übersicht Flipflops

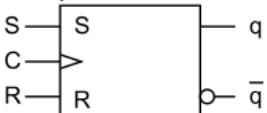
RS-Latch



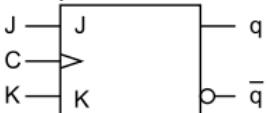
RS-Latch (Taktpiegelsteuerung)



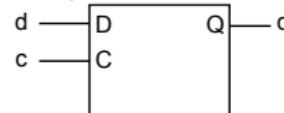
RS-Flipflop (Taktflankensteuerung)



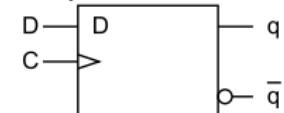
JK-Flipflop (Taktflankensteuerung)



D-Latch (Taktpiegelsteuerung)



D-Flipflop (Taktflankensteuerung)



T-Flipflop (Taktflankensteuerung)



Komplexere Schaltwerke

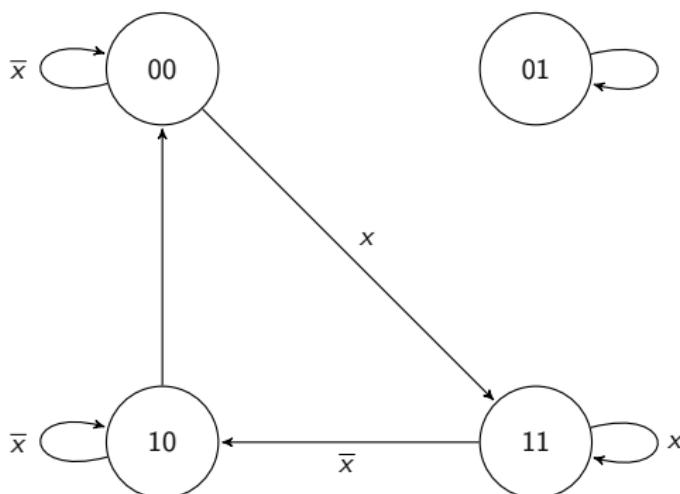
Eine wichtige Schaltungsklasse der Digitaltechnik bilden **endliche Zustandsautomaten**, im Englischen: *finite state machines (FSM)*

Einfache Zustandsautomaten mit wenigen Zuständen lassen sich durch Schaltwerke realisieren.

Bei mehreren Zustandsbits müssen Zustandswechsel verhindert werden bei denen sich mehr als ein Bit verändert. Ansonsten können fehlerhafte (Zwischen-)Zustände erreicht werden.

- ⇒ Diese lassen sich in asynchronen Schaltwerken nicht verhindern!
- ⇒ Für komplexere Automaten wird daher der Zustandsübergang immer mit FFs synchronisiert (⇒ synchroner Automat).

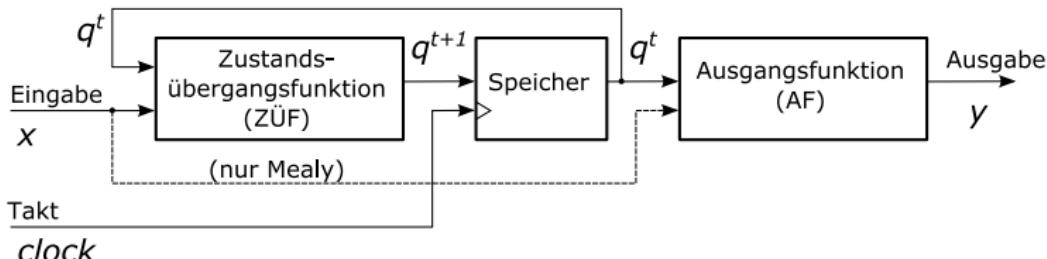
Beispiel



Geht der Automat von 00 → 11 über 01 oder 10?

⇒ Der Weg über 01 führt in eine Sackgasse!

Lösung: Synchrone Zustandsautomaten



Durch das Hinzufügen eines synchronen Speichers in die Rückkopplung wird die taktsynchrone Änderung des Zustandsvektors erzwungen.

Taktperiode muss hierbei größer als die Laufzeiten der Zustandsübergangsfunktionen sein.

Wahl des synchronen Speichers

Der taktsynchrone Speicher kann prinzipiell beliebigen Typs sein

Jedoch müssen i.A. entsprechende **Ansteuergleichungen** ermittelt werden, bei denen bestimmt wird, wie die Eingänge gesetzt werden müssen um ein Zustandsbit zu speichern.

Im Falle von D-Flipflops als Speicher vereinfachen sich die Ansteuergleichungen drastisch (Identität).

Im Weiteren werden ausschließlich synchrone Automaten mit D-Flipflops betrachtet.