

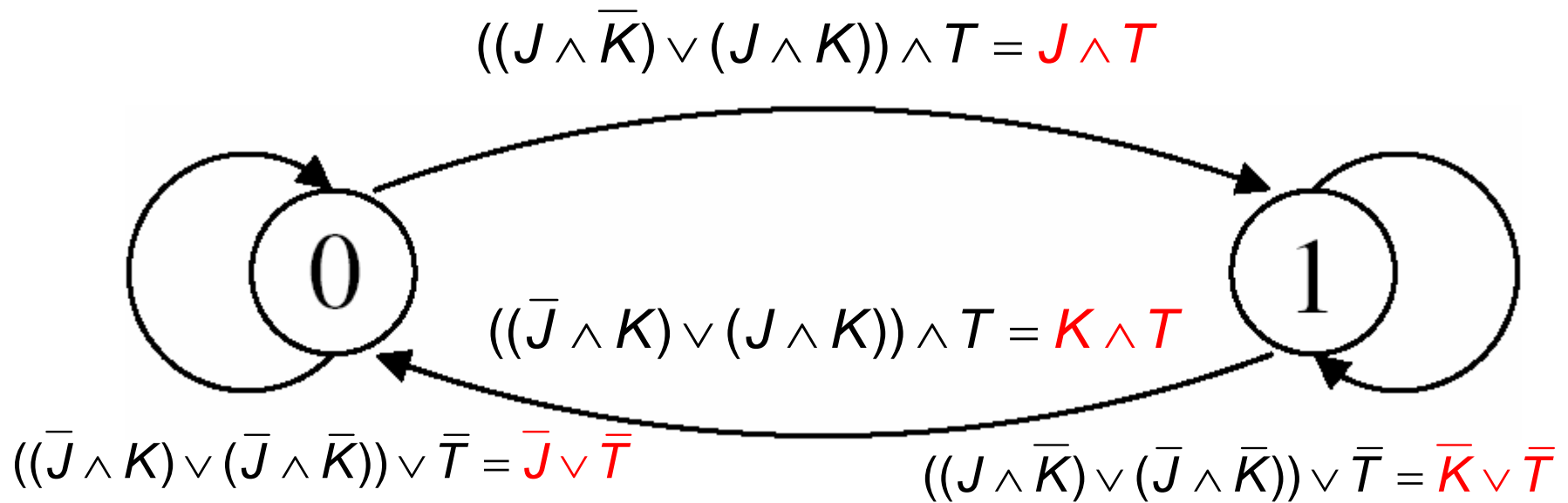
# Digitaltechnik II

## SS 2006

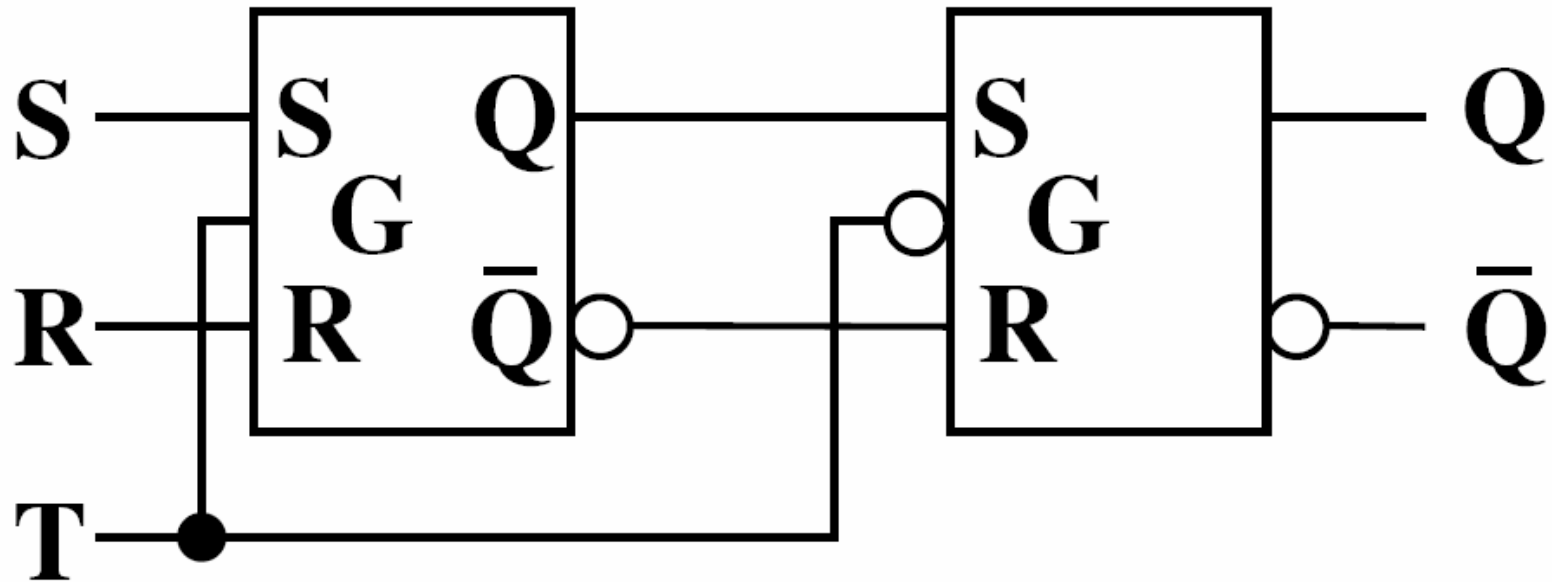
### 6. Vorlesung

Klaus Kasper

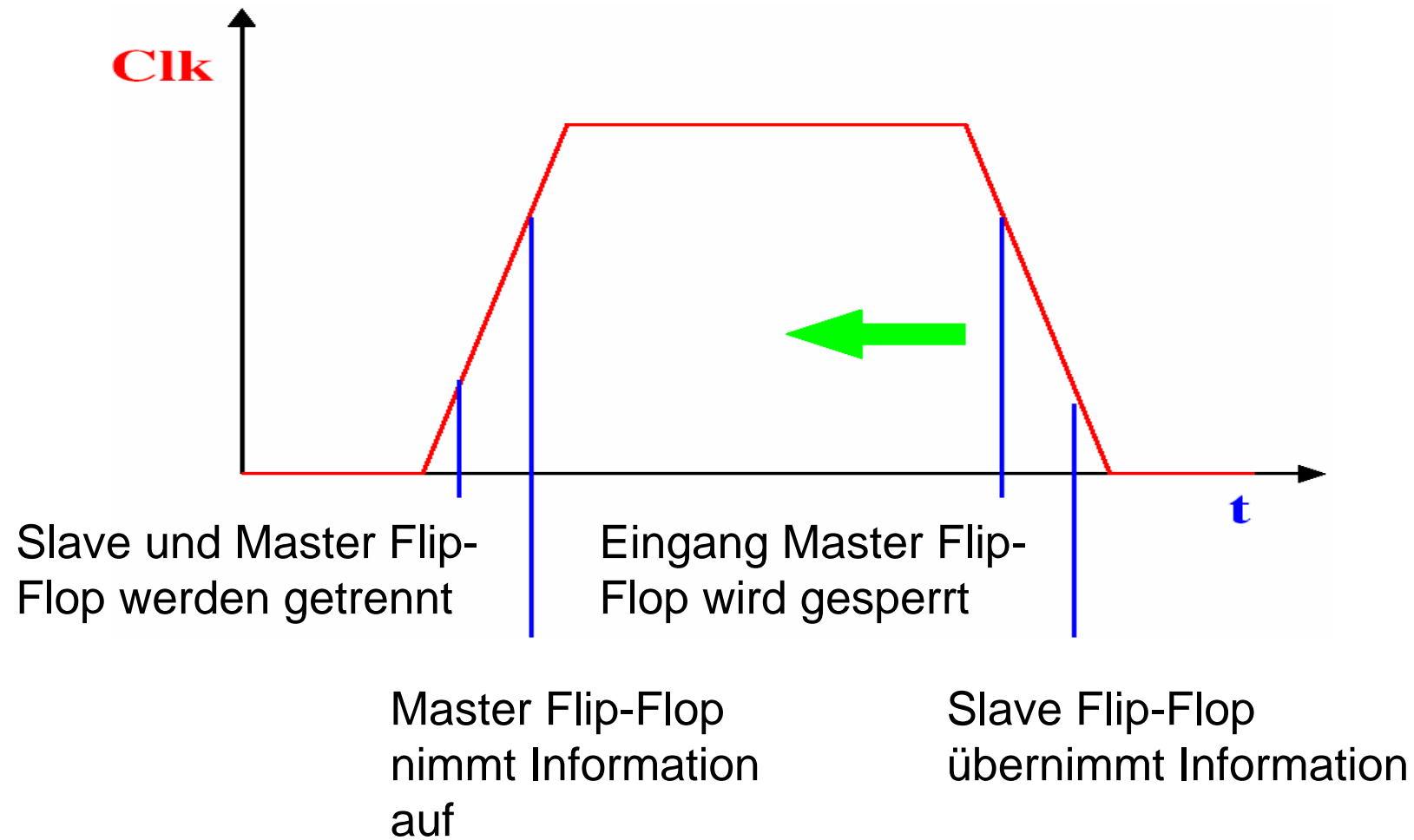
# Übung: Zustandsdiagramm JK Flip-Flop



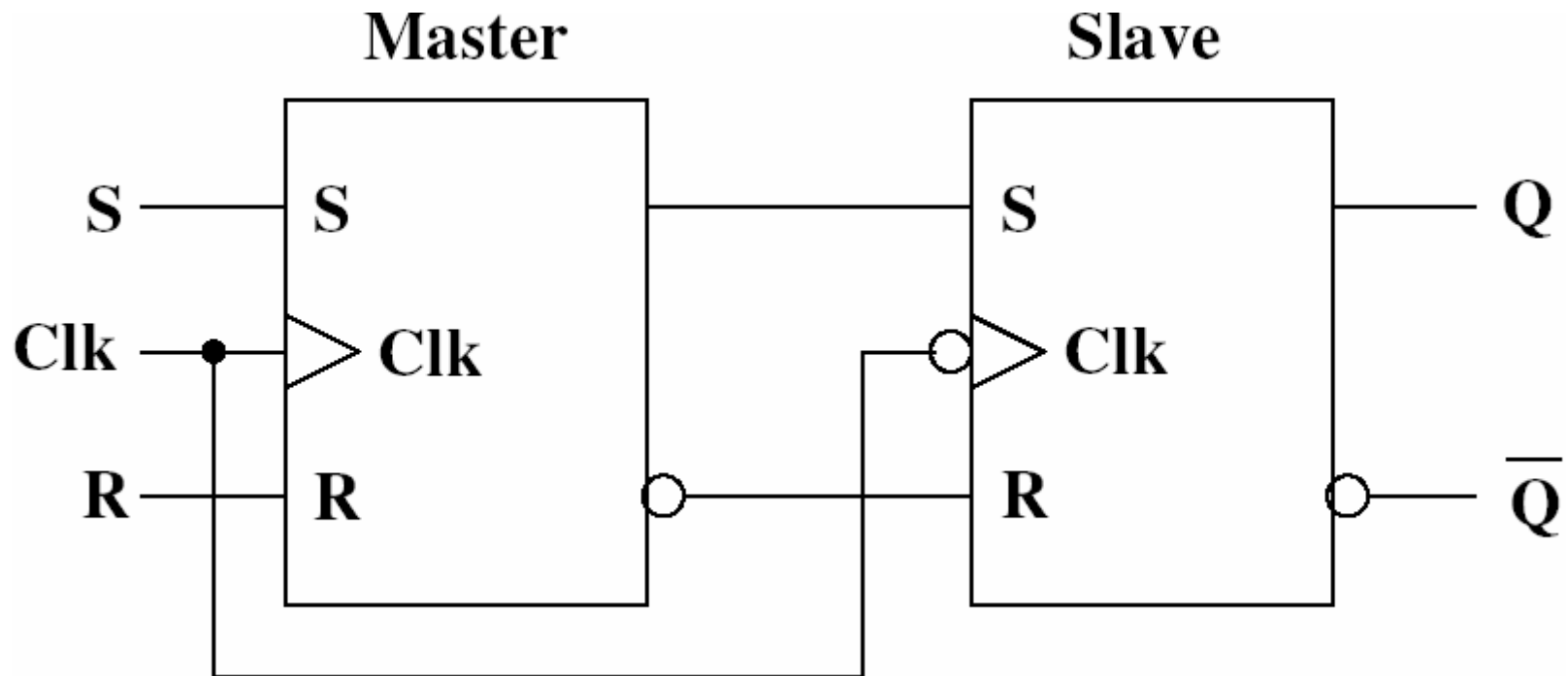
# Master-Slave Flip-Flop



# Master-Slave Flip-Flop



# Master-Slave Flip-Flop



# Master-Slave Flip-Flop

- Zwei Flanken gesteuertes Flip-Flop
- zwei Speicher
- Master-Slave Flip-Flop kann in gleicher Weise mit einem JK Flip-Flop als Master realisiert werden
- besonders sicher
- Information steht am Slave Flip-Flop erst verzögert zur Verfügung

# Zyklische Folgeschaltung

t	A	B	C
0	0	0	0
1	1	1	1
2	1	0	1
3	1	1	0
4	0	0	1
5	0	1	0
6	0	0	0
7	1	1	1

Das System wird vom Takt getrieben

6 Zustände, zyklische Wiederholung

3 Flip-Flops können 8 Zustände realisieren

Realisierung mit JK-Flip-Flops

Ausnutzung der Toggle Eigenschaft der JK-Flip-Flops

# Zyklische Folgeschaltung

A	B	C	AJ	AK	BJ	BK	CJ	CK
0	0	0	1	=				
1	1	1	-	0				
1	0	1	-	0				
1	1	0	=	1				
0	0	1						
0	1	0						
0	0	0						

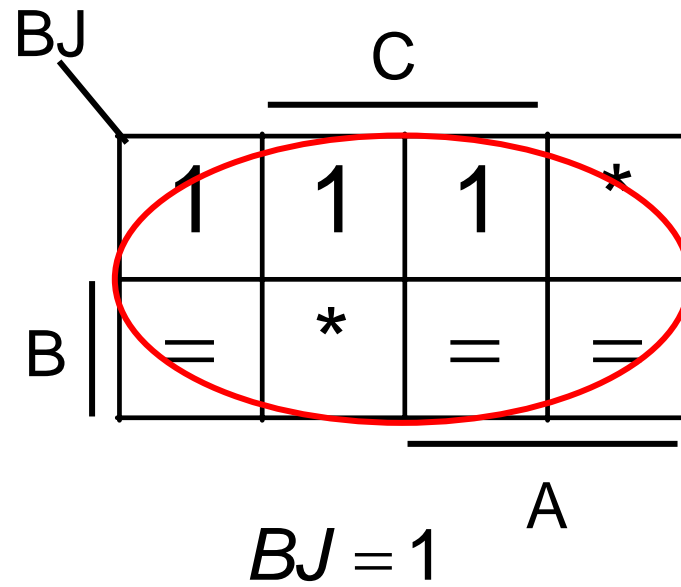
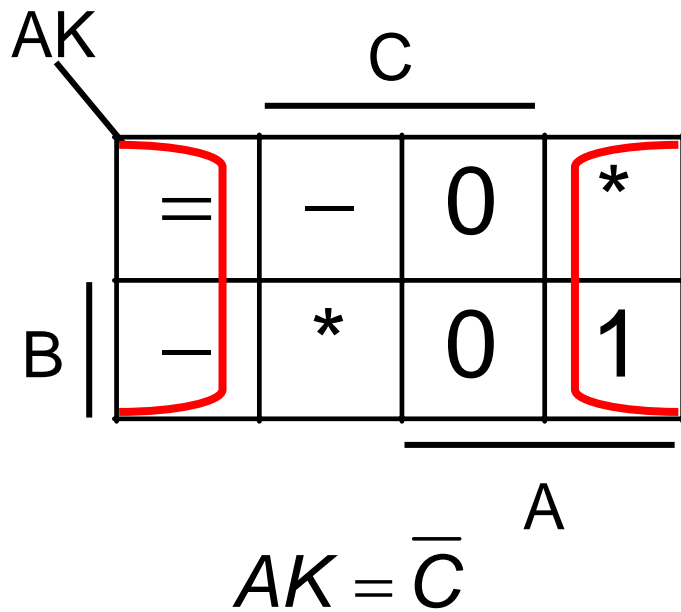
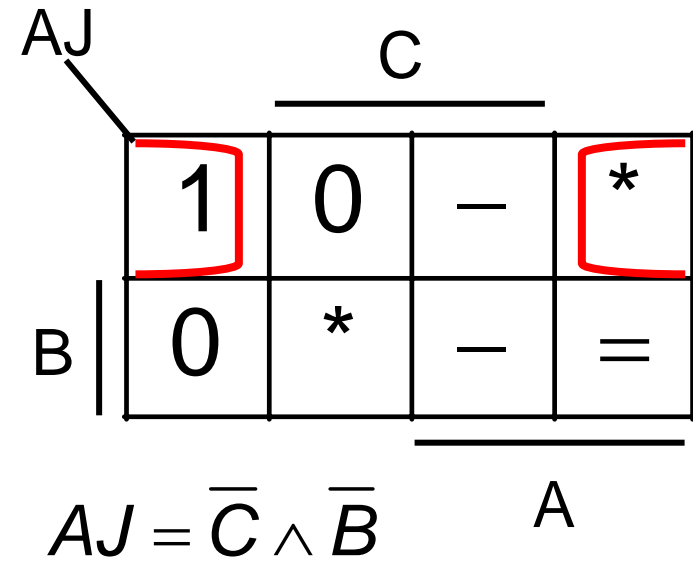
Zustandstabelle

# Zyklische Folgeschaltung

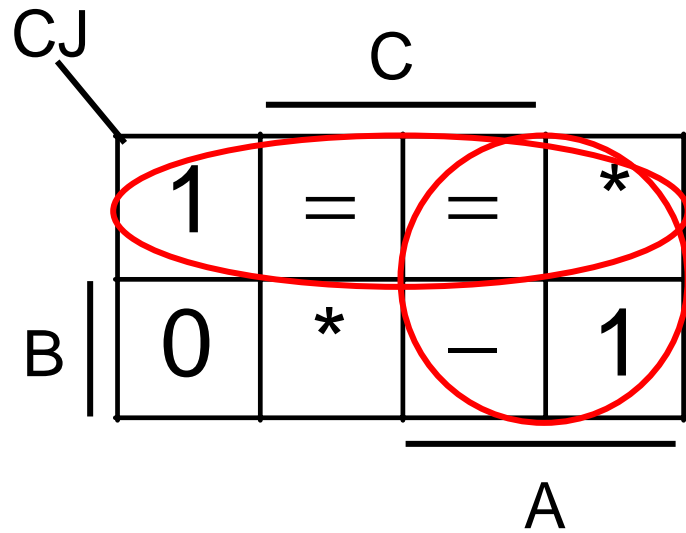
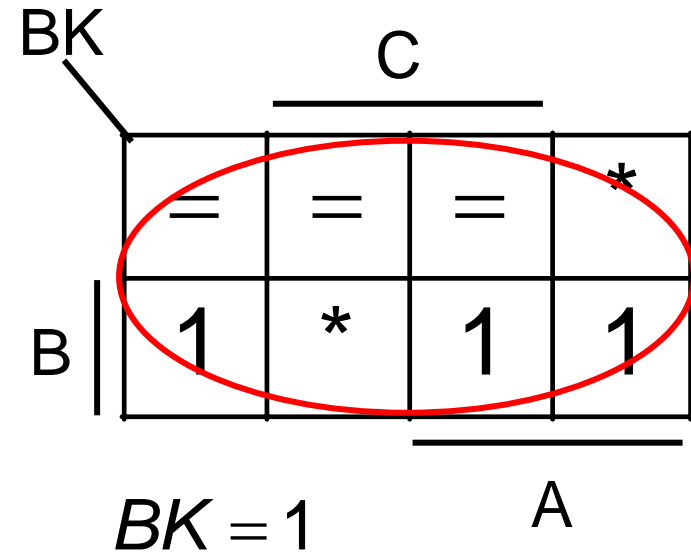
A	B	C	AJ	AK	BJ	BK	CJ	CK
0	0	0	1	=	1	=	1	=
1	1	1	-	0	=	1	-	0
1	0	1	-	0	1	=	=	1
1	1	0	=	1	=	1	1	=
0	0	1	0	-	1	=	=	1
0	1	0	0	-	=	1	0	-
0	0	0						

Zustandstabelle

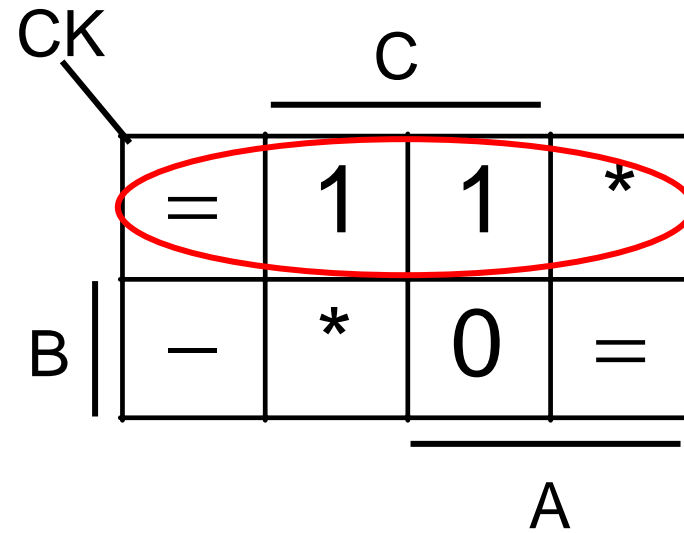
A	B	C	AJ	AK	BJ	BK	CJ	CK
0	0	0	1	=	1	=	1	=
1	1	1	-	0	=	1	-	0
1	0	1	-	0	1	=	=	1
1	1	0	=	1	=	1	1	=
0	0	1	0	-	1	=	=	1
0	1	0	0	-	=	1	0	-
0	0	0						



A	B	C	AJ	AK	BJ	BK	CJ	CK
0	0	0	1	=	1	=	1	=
1	1	1	-	0	=	1	-	0
1	0	1	-	0	1	=	=	1
1	1	0	=	1	=	1	1	=
0	0	1	0	-	1	=	=	1
0	1	0	0	-	=	1	0	-
0	0	0						



$$CJ = \bar{B} \vee A$$



$$CK = \bar{B}$$



# Digitalsimulator kaputt?

„JK-Flip-Flop geht nicht!“

# Zähler

Wie wird binär gezählt?

0 1 0 und Übertrag 1

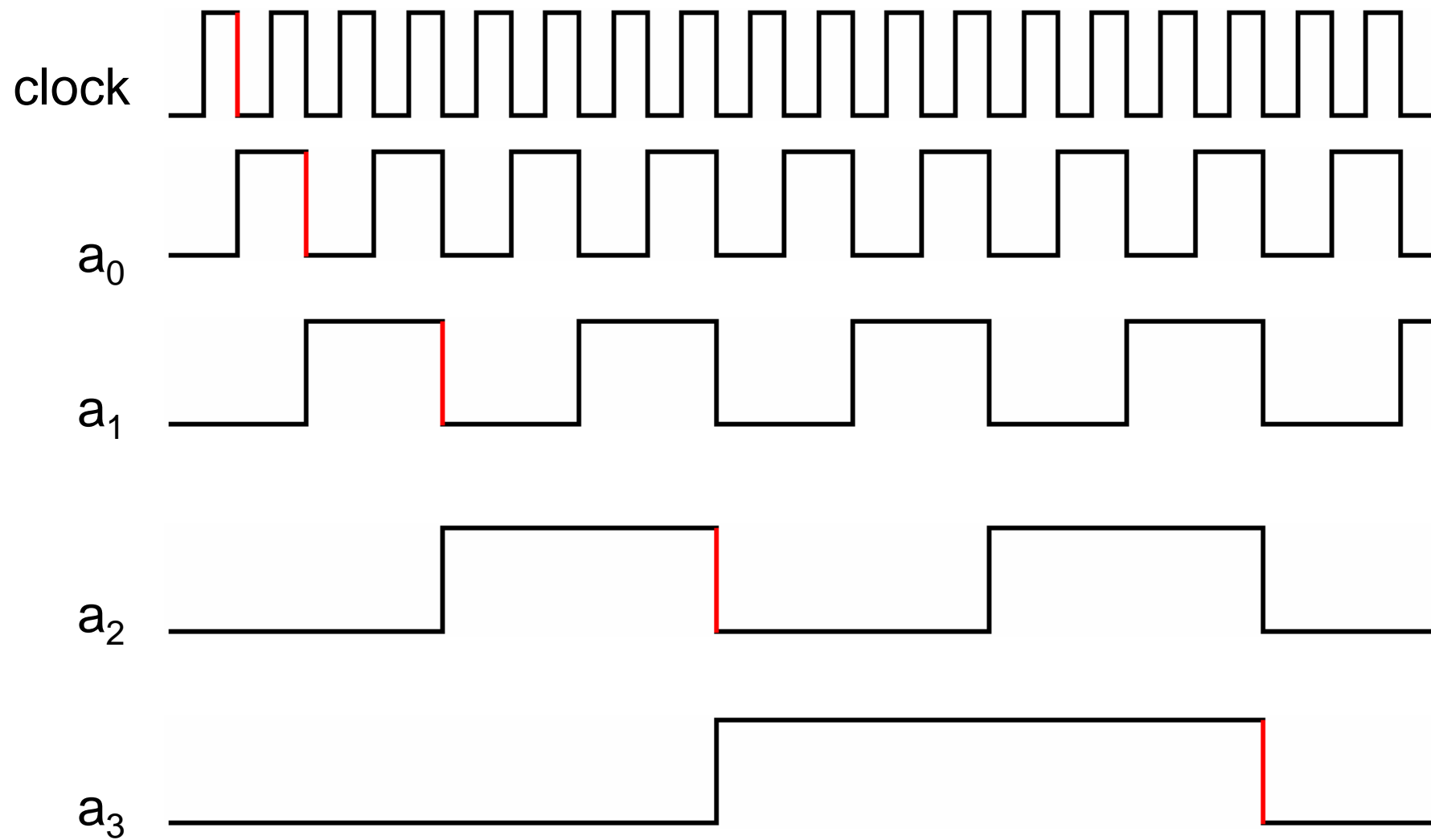
000, 001, 010, 011, 100, 101, 110, 111

000, 001, 010, 011, 100, 101, 110, 111

000, 001, 010, 011, 100, 101, 110, 111

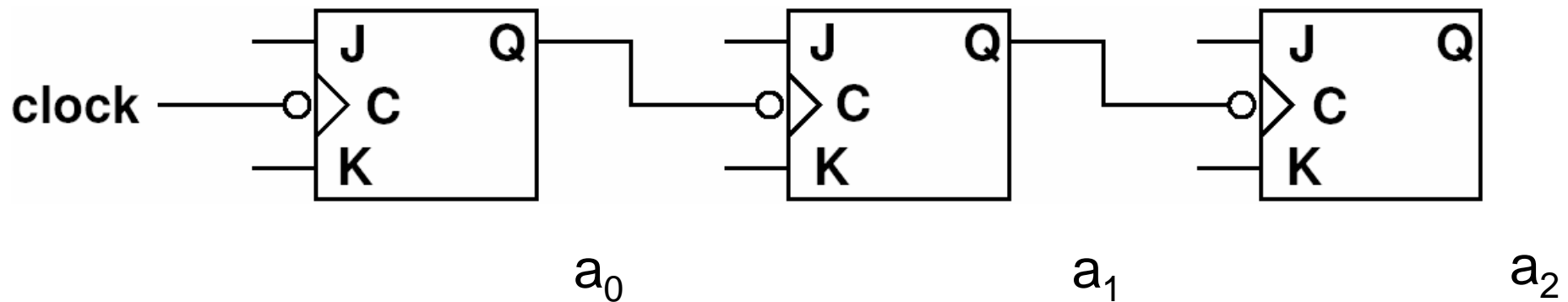
000, 001, 010, 011, 100, 101, 110, 111

# Frequenzteiler



Ideen zur Realisierung eines  
asynchronen Zählers?

# Prinzip asynchroner Zähler



Der Ausgang eines Flip-Flop erzeugt den Takt für das folgende Flip-Flop.

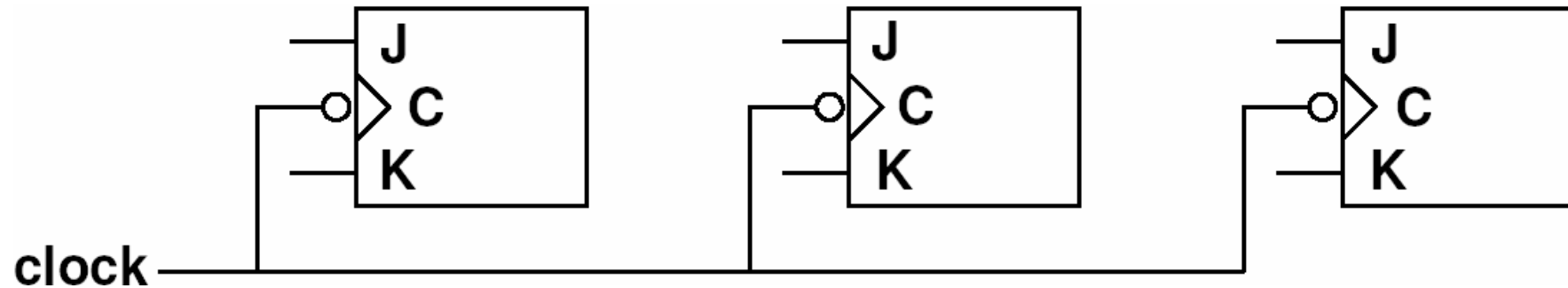
Für alle Flip-Flops gilt:  $J=K=1$

Extrem einfacher Aufbau

Flip-Flops schalten nicht gleichzeitig, so dass falsche Zwischenzustände entstehen.

Ideen zur Realisierung eines  
synchronen Zählers?

# Prinzip synchroner Zähler



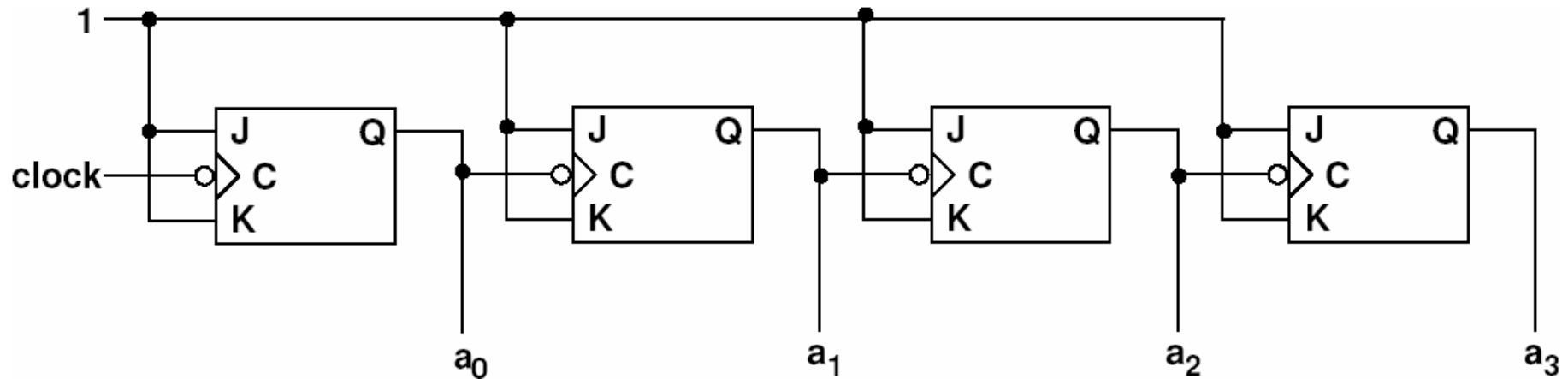
Alle Flip-Flops erhalten den gleichen Takt und schalten daher gleichzeitig.

Mit zusätzlicher Beschaltung muss die adäquate Ansteuerung der Eingänge der Flip-Flops realisiert werden.

Keine falschen Zwischenwerte

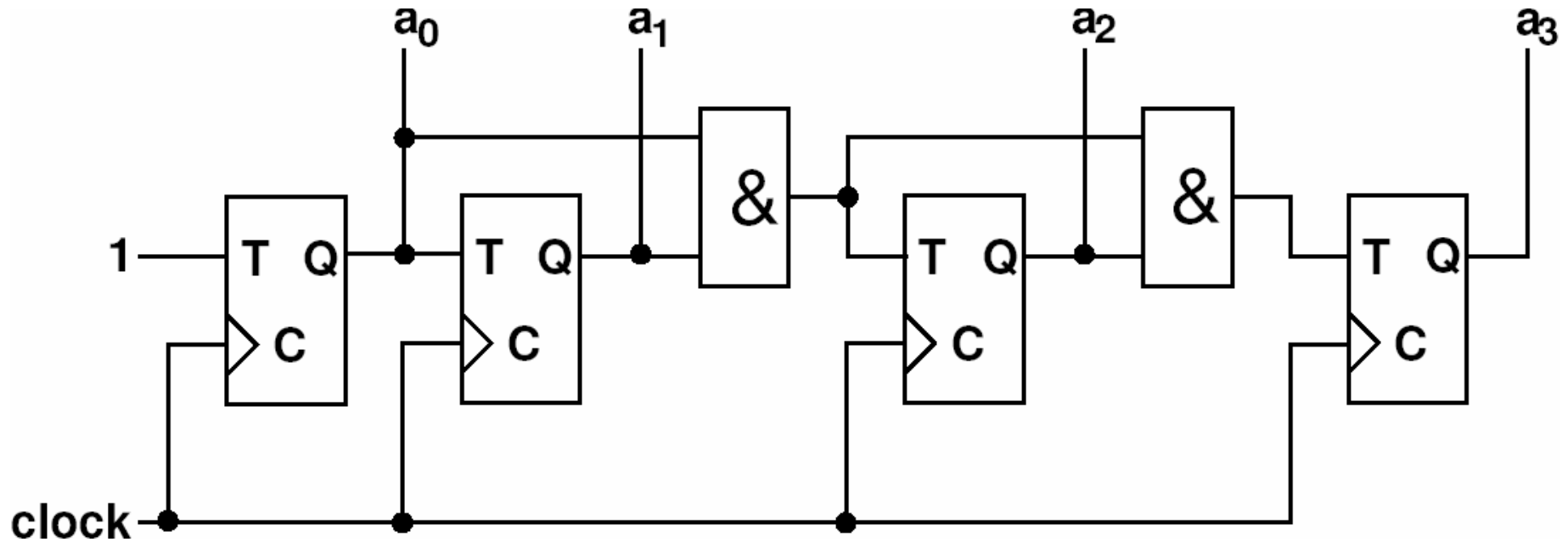
Komplexer Aufbau

# asynchroner Zähler



Flip-Flops müssen eine negative Flankensteuerung haben!

# synchroner Zähler



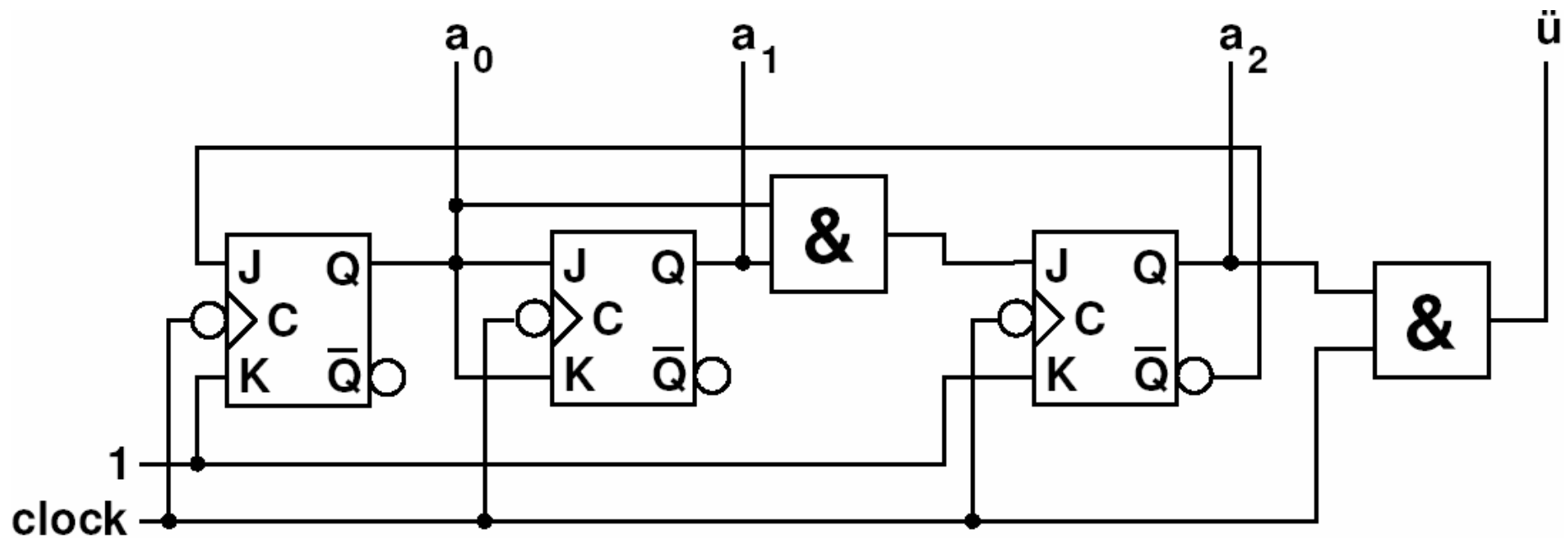
$$T_0 = 1$$

$$T_2 = a_0 \wedge a_1$$

$$T_1 = a_0$$

$$T_3 = (a_0 \wedge a_1) \wedge a_2$$

# Mod-5-Synchronzähler I



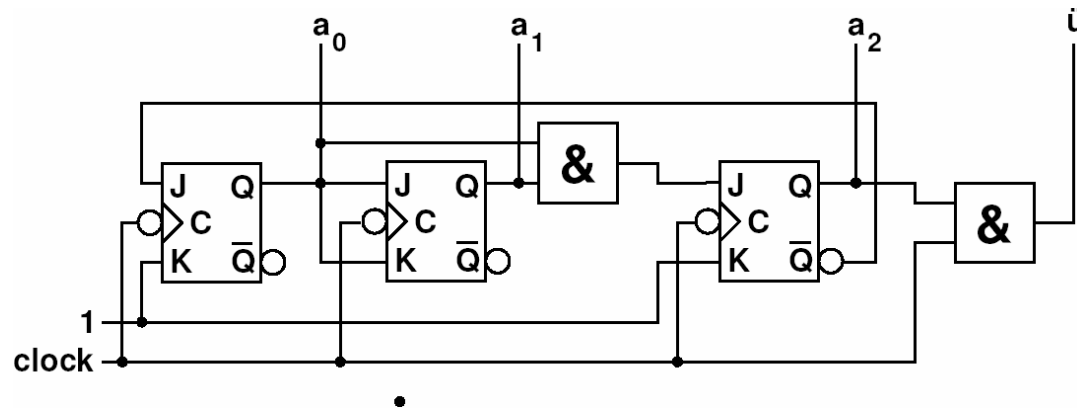
$$J_0 = \bar{a}_2, K_0 = 1$$

$$J_1 = K_1 = a_0$$

$$J_2 = a_0 \wedge a_1, K_2 = 1$$

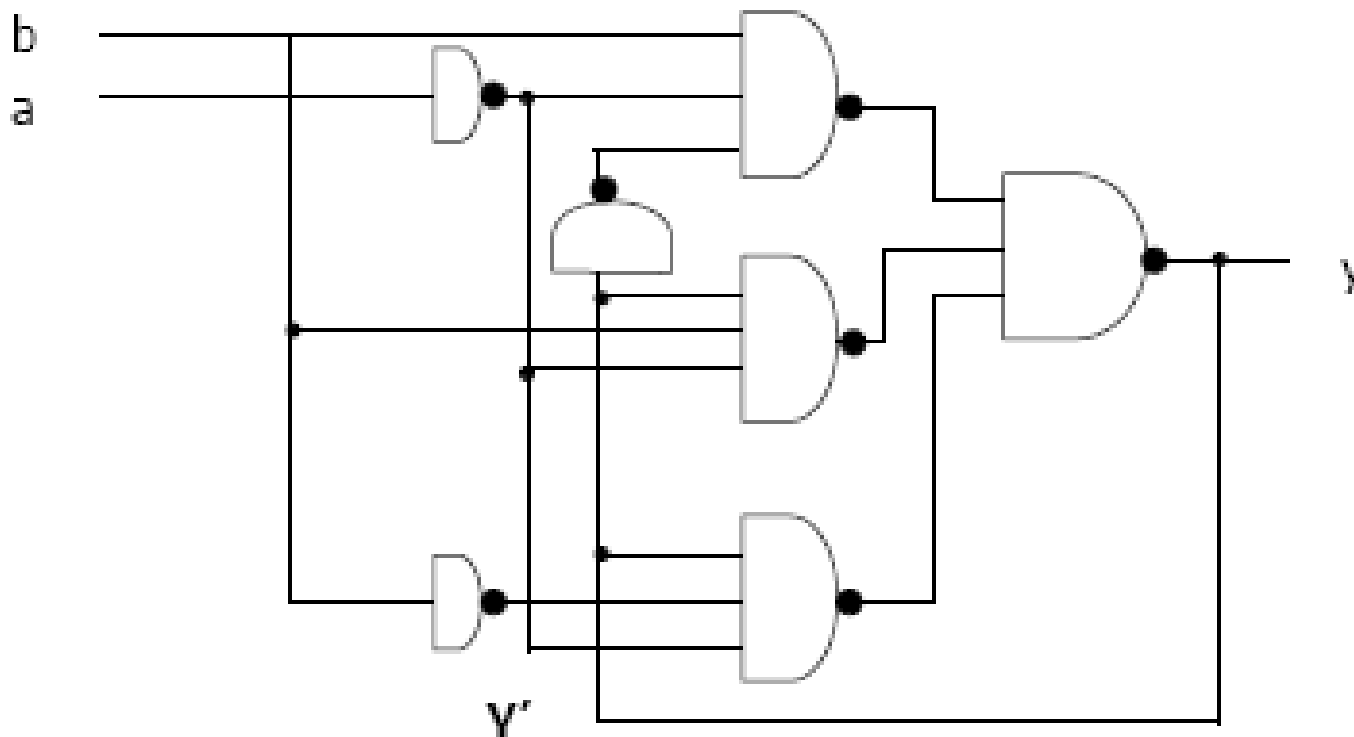
$$\bar{u} = \text{clock} \wedge a_2$$

# Mod-5-Synchronzähler II

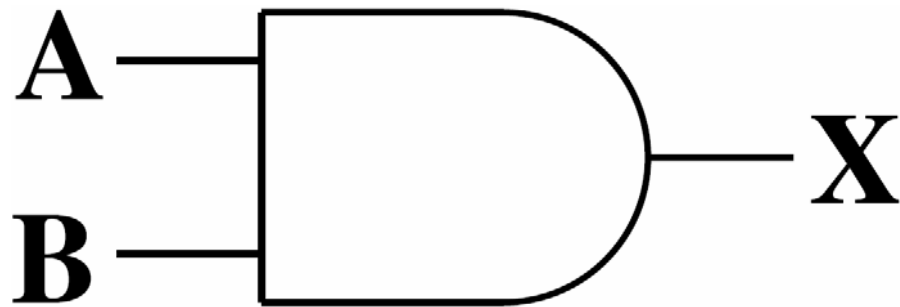
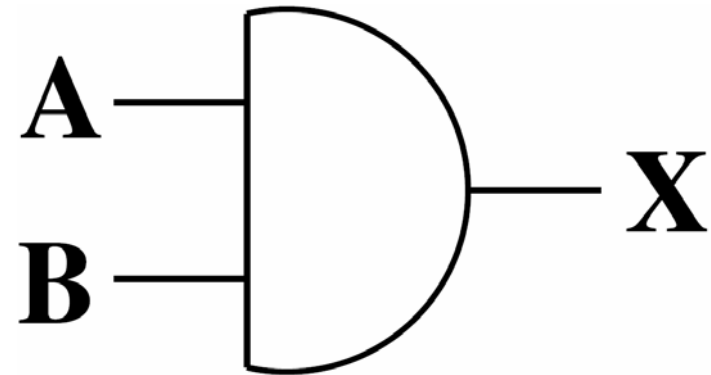
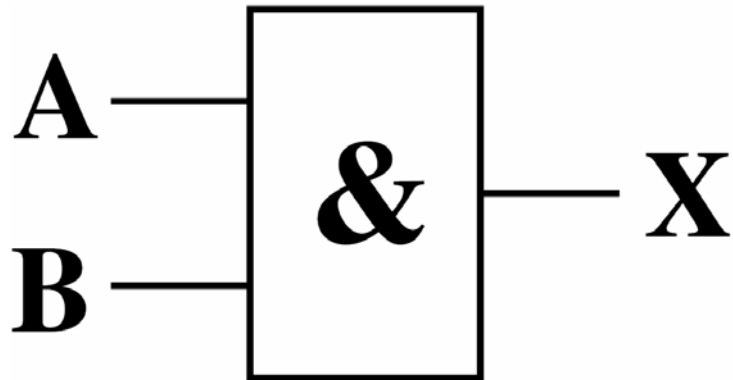


t	$a_0$	$a_1$	$a_2$	$\ddot{u}$
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	1	1	0	0
4	0	0	1	1
5	0	0	0	0
6	1	0	0	0

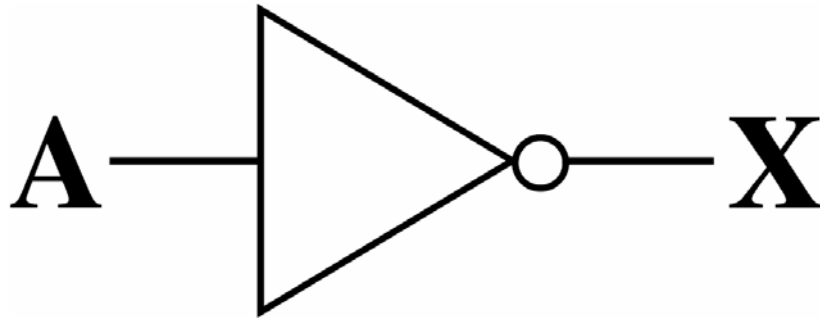
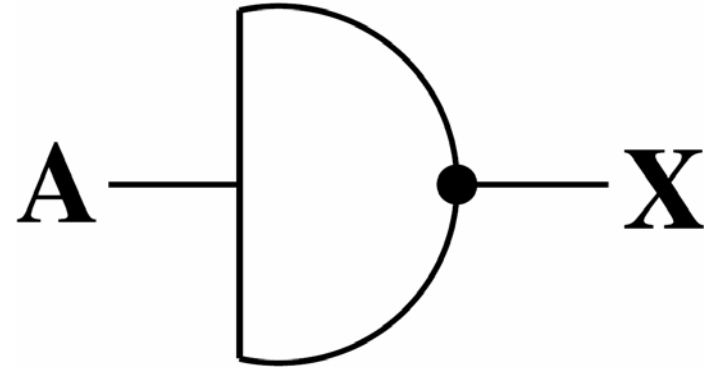
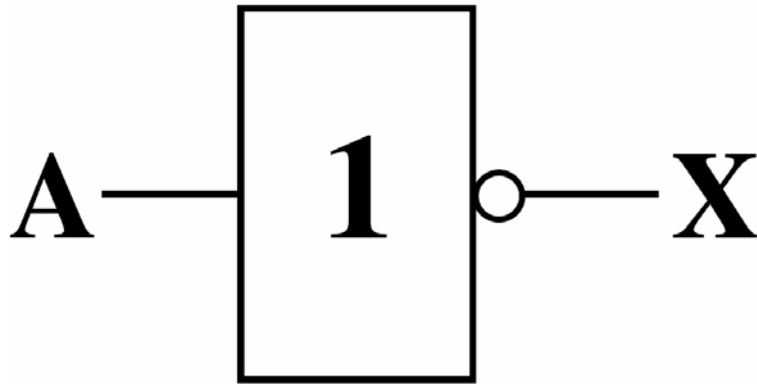
# Schaltungsanalyse



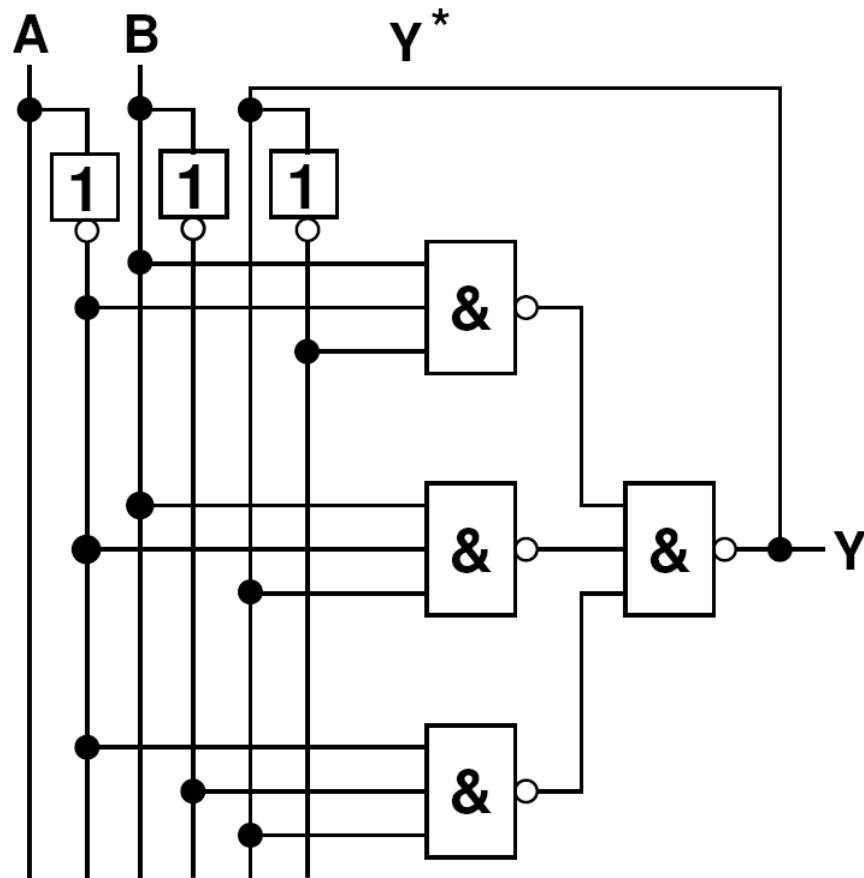
# Schaltzeichen: UND



# Schaltzeichen: Negation



# Schaltungsanalyse



- algebraische Darstellung
- Wahrheitstabelle
- KV-Diagramm
- stabile Zustände markieren
- effiziente Realisierung
- ausschließlich NOR-Gatter

# Schaltungsanalyse

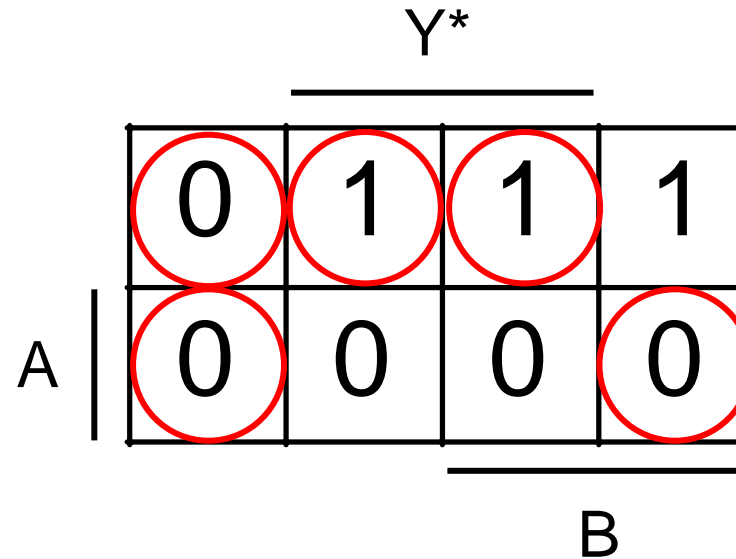
$$\begin{aligned} Y &= \overline{\overline{(B \wedge \bar{A} \wedge \bar{Y}^*)} \wedge \overline{(B \wedge \bar{A} \wedge Y^*)} \wedge \overline{(\bar{B} \wedge \bar{A} \wedge Y^*)}} \\ &= (B \wedge \bar{A} \wedge \bar{Y}^*) \vee (B \wedge \bar{A} \wedge Y^*) \vee (\bar{B} \wedge \bar{A} \wedge Y^*) \\ &= (B \wedge \bar{A}) \vee (\bar{B} \wedge \bar{A} \wedge Y^*) \\ &= \bar{A} \wedge (B \vee (\bar{B} \wedge Y^*)) \\ &= \bar{A} \wedge ((B \vee \bar{B}) \wedge (B \vee Y^*)) \\ &= \bar{A} \wedge (B \vee Y^*) \end{aligned}$$

# Schaltungsanalyse

B	A	$Y^*$	$Y_1 = \neg(B \wedge A \wedge Y^*)$	$Y_2 = \neg(B \wedge A \wedge Y^*)$	$Y_3 = \neg(\neg B \wedge A \wedge Y^*)$	$Y = \neg(Y_1 \wedge Y_2 \wedge Y_3)$
0	0	0	1	1	1	0
0	0	1	1	1	0	1
0	1	0	1	1	1	0
0	1	1	1	1	1	0
1	0	0	0	1	1	1
1	0	1	1	0	1	1
1	1	0	1	1	1	0
1	1	1	1	1	1	0

# Schaltungsanalyse

B	A	Y*	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0



Markierung der stabilen Zustände!

Schaltungsaufbau ausschließlich mit NOR Gattern und Inverter!



KMF

# Schaltungsanalyse

	$Y^*$			
	0	1	1	1
A	0	0	0	0
	$B$			

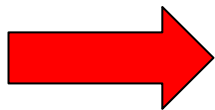
$$Y = \bar{A} \wedge (B \vee Y^*)$$

$$= \overline{\bar{A} \wedge (B \vee Y^*)}$$

$$= \overline{\bar{A}} \vee \overline{(B \vee Y^*)}$$

$$= A \vee \overline{(B \vee Y^*)}$$

Welche Schaltung?

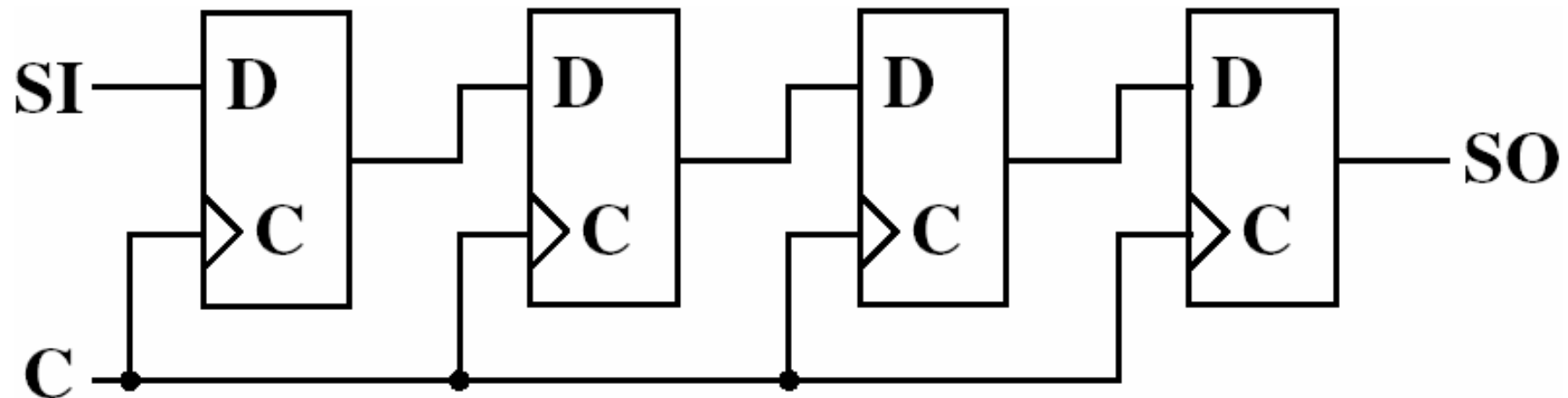


NOR SR Flip-Flop

# Register

- Parallele Anordnung von Flip-Flops mit gemeinsamen Takt.
- Auffang- oder Buffer-Register zur Zwischenspeicherung von Bitfolgen.
- Schiebe- oder Shift-Register zur Parallel-Seriell-Umwandlung oder für binäre Multiplizierer / Dividierer

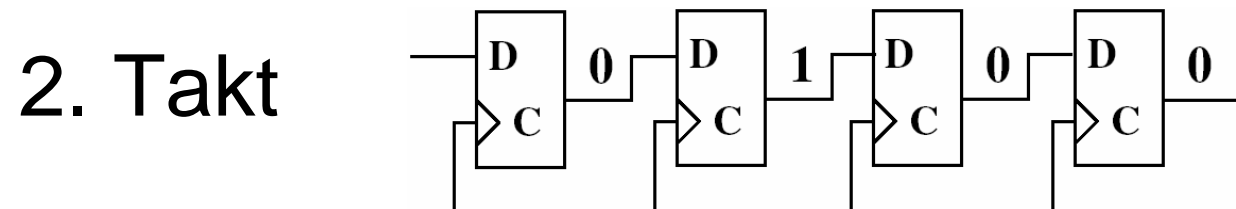
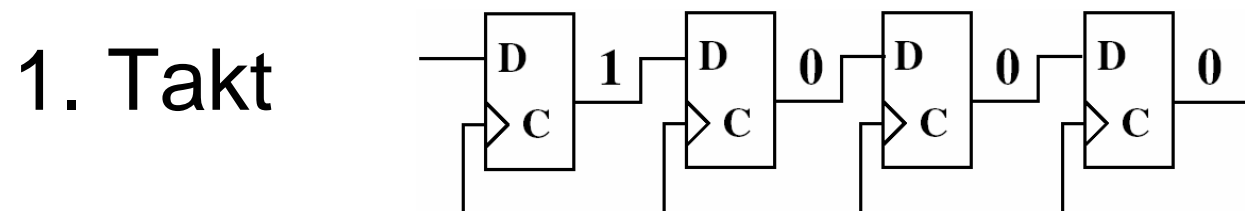
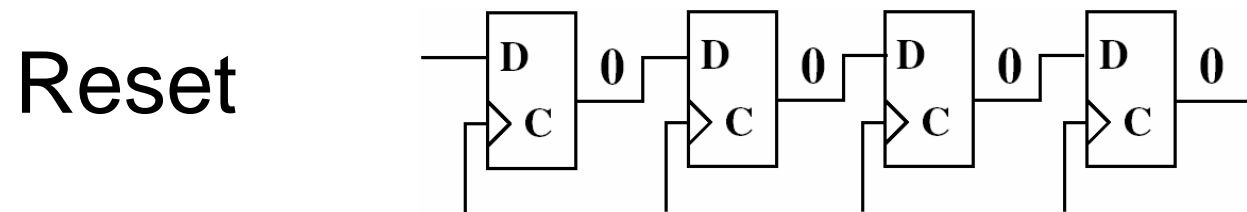
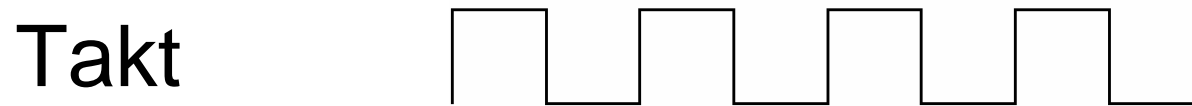
# Schieberegister (SISO)



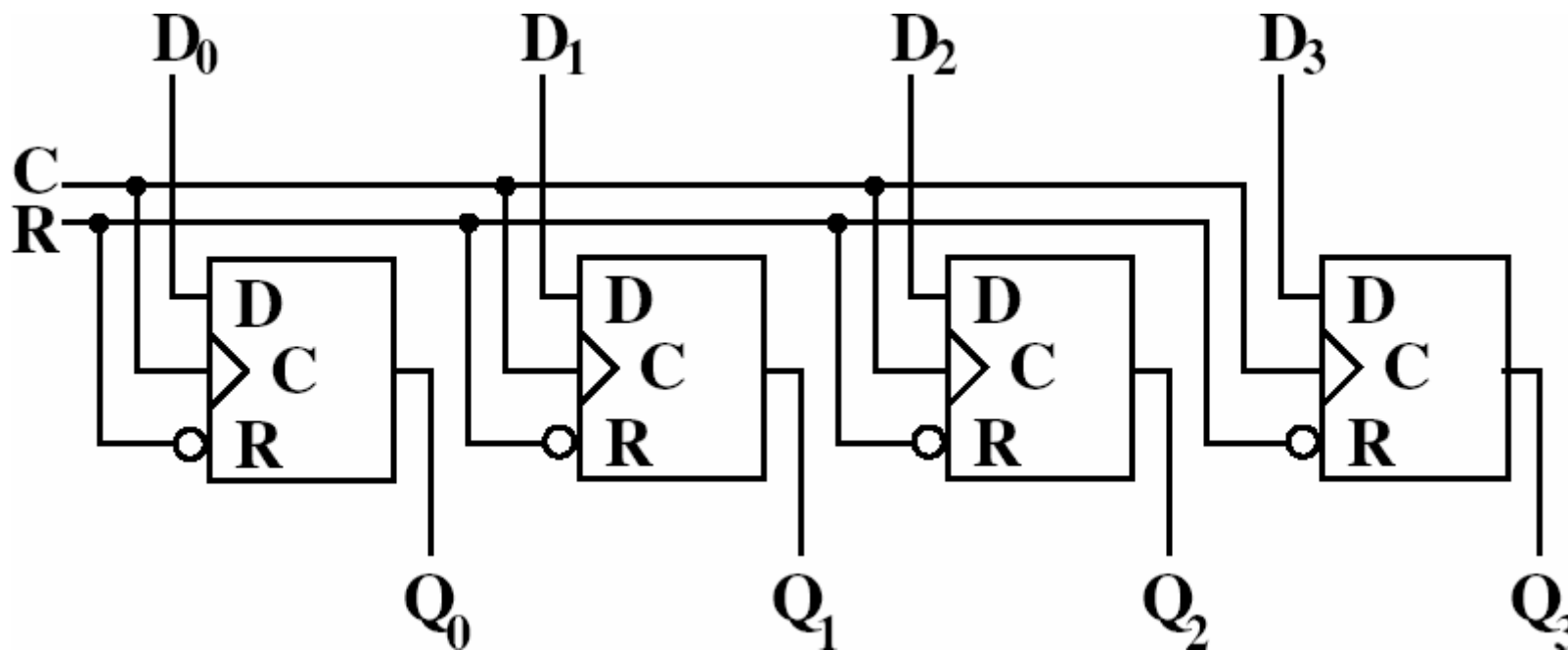
4-bit-Schieberegister (serieller Eingang, serieller Ausgang)

Mit jeder positiven Flanke wird jedes bit ein Flip-Flop nach rechts verschoben.

# Zeitlicher Verlauf (SISO)



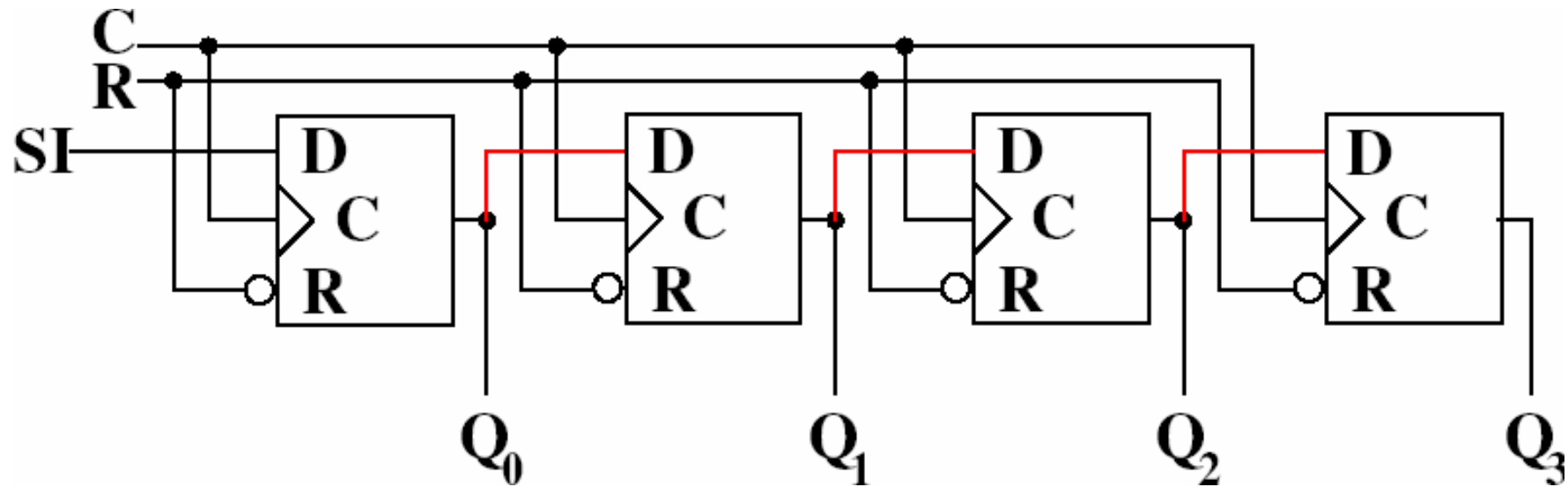
# Auffangregister (PIPO)



4 bit werden parallel gespeichert und können parallel gelesen werden

Mit dem Reset Eingang (R) können alle Ausgänge auf 0 gesetzt werden

# Schieberegister (SIPO)



Daten werden seriell eingelesen und parallel ausgelesen.

Mit dem Reset (R) Eingang können alle Ausgabewerte auf 0 gesetzt.

