

Prof. Dr.-Ing. Dr. h. c. J. Becker

becker@kit.edu

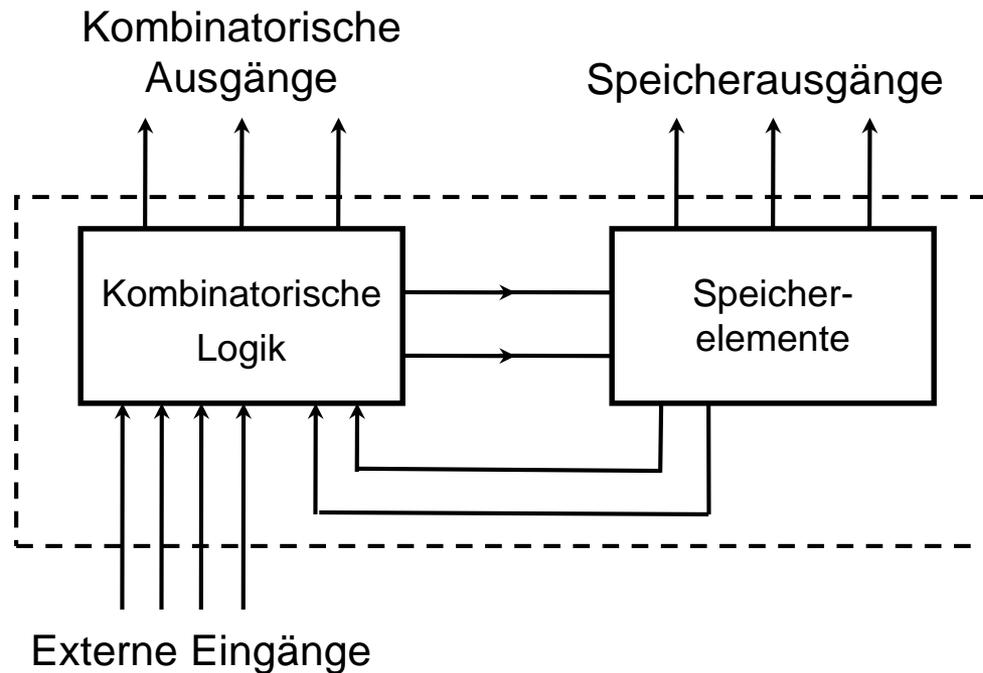
Karlsruher Institut für Technologie (KIT)

Institut für Technik der Informationsverarbeitung (ITIV)

Digitaltechnik

Digitale Speicherbausteine (FlipFlop-Schaltungen)

- Sequentielle Schaltkreise bestehen meist aus einem oder mehreren **Rückkopplungspfaden** in Kombination mit **Speicherelementen**



- **Sequentieller Schaltkreis** = Kombinatorische Logik + Speicherelemente

- Es existieren **zwei Typen** von **sequentiellen Schaltkreisen**:
 - **synchron**: Ausgänge ändern sich zu **festgelegten** Zeitpunkten
→ **Takt**
 - **asynchron**: Ausgänge ändern sich zu **beliebigen** Zeitpunkten
- Man unterscheidet zwei bistabile logische Speicherkomponenten:
 - pegel- und
 - flankengesteuerte **FlipFlops**
- **FlipFlops**: unterscheiden sich in der Art und Weise wie ihre **Zustände** **geändert** werden
→ in Abhängigkeit der **Eingangssignale** sowie des **Taktsignals**

- **Speicherelement:** ein Gerät oder Modul, welches einen zuvor-angelegten Wert unbegrenzt Speichern und auf Wunsch wieder ändern kann



Charakteristische Tabelle:

Befehl (zum Zeitpunkt t)	$Q(t)$	$Q(t+1)$
Set	-	1
Reset	-	0
Speichern / keine Änderung	0	0
	1	1

$Q(t)$: aktueller Zustand

$Q(t+1)$: nächster Zustand

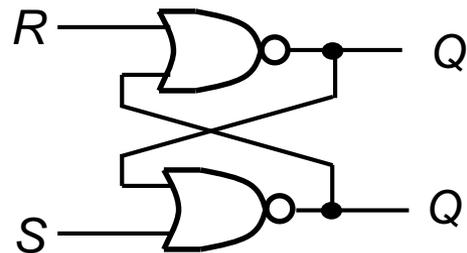
- Zwei **Typen** der **Trigger-/Aktivierungsmechanismen**
 - **pegelgesteuert**
 - **flankengesteuert**
- **Typ 1:** **pegelgesteuerte Speicherelemente** (FlipFlops) werden auch als **Latches** bezeichnet -> mit **Pegeln** von Signalen angesteuert
 - **Pegel auf high** -> das Speicherelement ist **aktiviert**
 - **Pegel auf low** -> das Speicherelement ist **deaktiviert**
- **Typ 2:** **flankengesteuerte Speicherelemente** (FlipFlops) werden mit steigender oder fallender **Flanke** angesteuert
 - **positive Flankensteuerung:**
 - aktiviert wenn Übergang von **0 nach 1**
 - deaktiviert sonst
 - **negative Flankensteuerung:**
 - aktiviert wenn Übergang von **1 nach 0**
 - deaktiviert sonst

RS-FlipFlop (Latch):

- Komplementäre Ausgänge Q und Q,
- 2 Steuereingänge R, S:
 - Latch im Zustand SET → Q auf HIGH
 - Latch im Zustand RESET → Q auf LOW
- Für ein **active-HIGH RS-Latch** (realisiert mit **NOR-Gattern**) gilt:
 - R = HIGH (und S = LOW) ⇒ **RESET-Zustand**
 - S = HIGH (und R = LOW) ⇒ **SET-Zustand**
 - S = LOW und R = LOW ⇒ **keine Zustandsänderung**
 - S = HIGH und R = HIGH ⇒ Q und Q', sind auf LOW (**nicht zulässig!!!**)
- Für ein **active-LOW RS-Latch** (realisiert mit **NAND-Gattern**) gilt:
 - R = LOW (und S = HIGH) ⇒ **RESET-Zustand**
 - S = LOW (und R = HIGH) ⇒ **SET-Zustand**
 - S = HIGH und R = HIGH ⇒ **keine Zustandsänderung**
 - S = LOW und R = LOW ⇒ Q und Q', sind auf LOW (nicht zulässig!!!)
- **Nachteil der RS-Latches: ungültige Eingangskombination existiert!!!**
 - diese muss in der Praxis verhindert werden

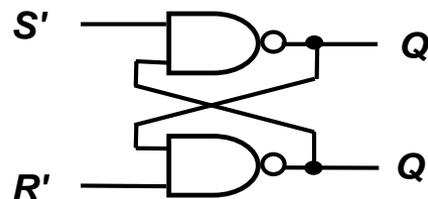
RS-FlipFlop (RS-Latch):

■ *Active-HIGH RS-Latch:*



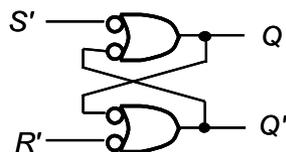
S	R	Q	Q'	
1	0	1	0	Initialzustand
0	0	1	0	(nach S=1, R=0)
0	1	0	1	
0	0	0	1	(nach S=0, R=1)
1	1	0	0	ungültig!

■ *Active-Low RS-Latch:*



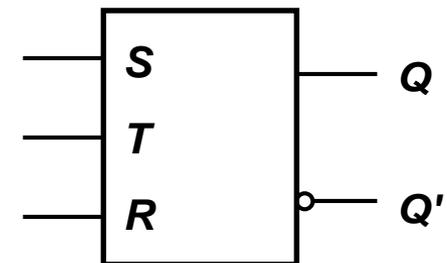
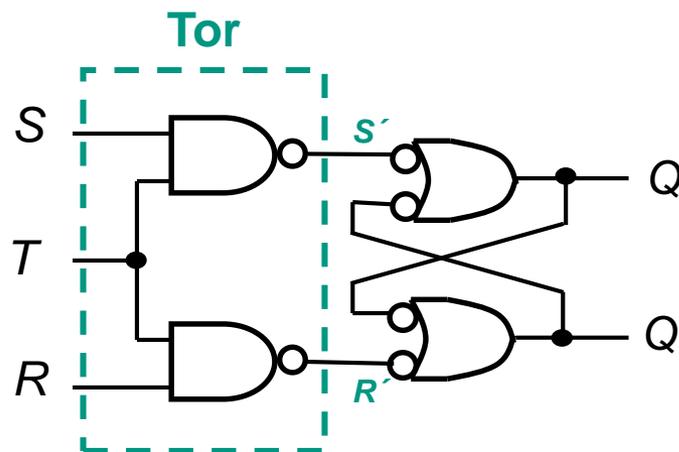
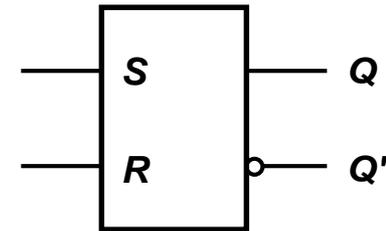
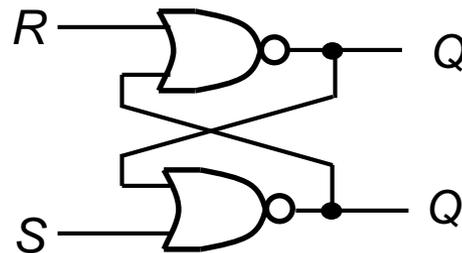
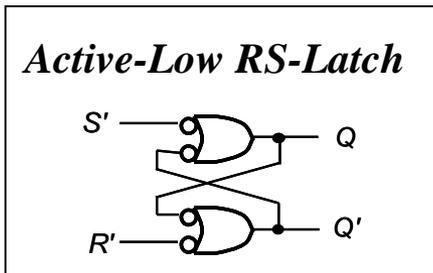
S'	R'	Q	Q'	
1	0	0	1	Initialzustand
1	1	0	1	(nach S'=1, R'=0)
0	1	1	0	
1	1	1	0	(nach S'=0, R'=1)
0	0	1	1	ungültig!

Alternative:



Getaktetes RS-Latch (active-high):

- RS-Latch + Takt-Eingang T und 2 NAND-Gatter -> getaktetes RS-Latch:



Getaktetes *RS-Latch*:

- Ausgang verändert sich nur wenn $T = HIGH$ ist
- Unter welchen Bedingungen wechseln die Zustände?

Charakteristische Tabelle:

$T=1$:

$Q(t)$	S	R	$Q(t+1)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	undefiniert
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	undefiniert

S	R	$Q(t+1)$
0	0	$Q(t)$ Kein Wechsel
0	1	0 Reset
1	0	1 Set
1	1	undefiniert

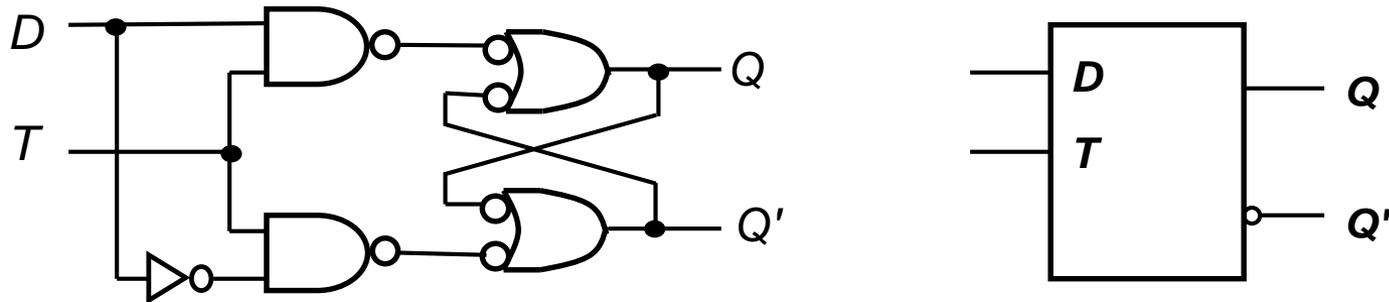
$$Q(t+1) = S \text{ or } (\bar{R} \text{ and } Q)$$

$$S \text{ and } R = 0 !$$

Getaktetes D-FlipFlop (D-Latch):

- S invertiert an R zugeführt
→ man erhält das **getaktete D-FlipFlop (D-Latch)**

Vorteil: **D-Latch** eliminiert die ungültige Eingangskombination: $S \text{ and } R = 0!$



Charakteristische Tabelle:

- Wenn $T = \text{HIGH}$:
→ $Q = D$

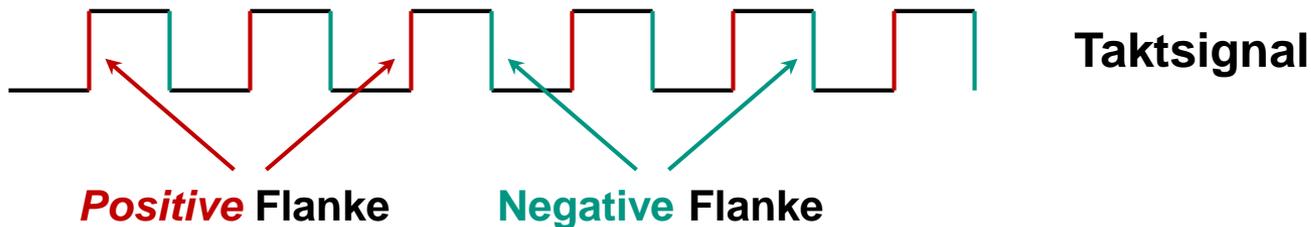
T	D	$Q(t+1)$	
1	0	0	Reset
1	1	1	Set
0	-	$Q(t)$	Kein Wechsel

wenn $T=1$, $Q(t+1) = D$

Flankengesteuerte FlipFlops:

- FlipFlops sind **bistabile Speicherelemente**
- **Ausgang ändert** seinen Zustand zu **synchronen Zeitpunkten**, die durch den **Takt** vorgegeben werden
- Dies geschieht entweder zur **positiven** oder **negativen** Taktflanke
- Anliegender Wert wird in einem **“kleineren Zeitfenster“** vom Eingang auf den Ausgang geschaltet

→ **geringere “Rückkopplungsgefahr“**



Flankengesteuertes RS-FlipFlop:

- **RS-Flipflop** an getriggertem **Flanke** des **Taktsignals**:
 - **S = HIGH** (und R = LOW) \Rightarrow **SET-Zustand**
 - **R = HIGH** (und S = LOW) \Rightarrow **RESET-Zustand**
 - **beide Eingänge auf LOW** \Rightarrow **keine Änderung** der **Ausgänge**
 - **beide Eingänge auf HIGH** \Rightarrow **ungültig**

Charakteristische Tabelle:

für ein auf **positive Flanke** triggerndes **RS-Flipflop**:

S	R	CLK	Q(t+1)	Zustand
0	0	X	Q(t)	Kein Wechsel
0	1	↑	0	Reset
1	0	↑	1	Set
1	1	↑	?	ungültig

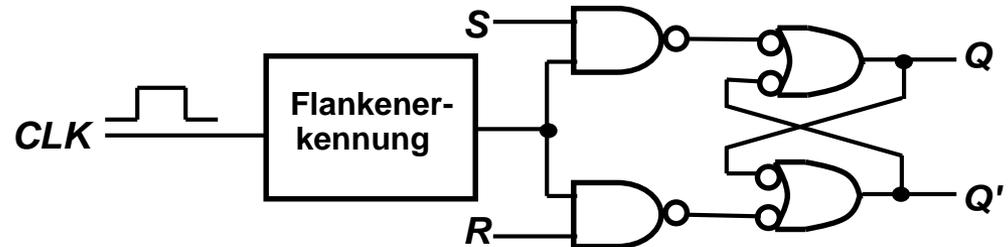
X = irrelevant ("don't care")

↑ = Taktübergang von LOW auf HIGH

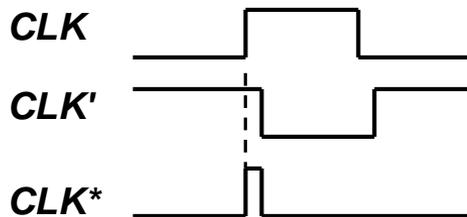
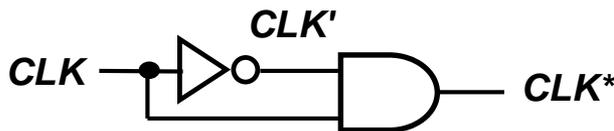
Flankengesteuertes RS-Flipflop:

- Impulsgesteuerter Schaltkreis (NAND-Gatter)
- Flankenerkennungsschaltkreis
- Der **Flankenerkennungsschaltkreis** erkennt positiven (oder negativen) Flankenübergang und generiert einen **Impuls** von **kurzer Dauer**

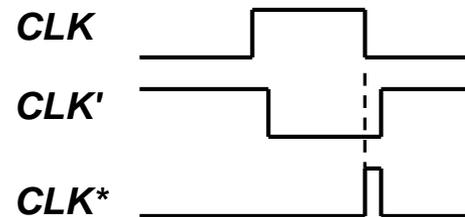
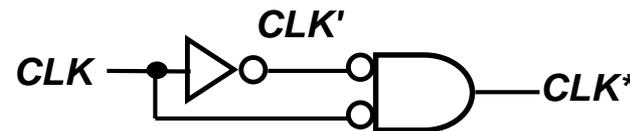
Flankengesteuertes RS-FlipFlop



Flankenerkennungsschaltkreis:



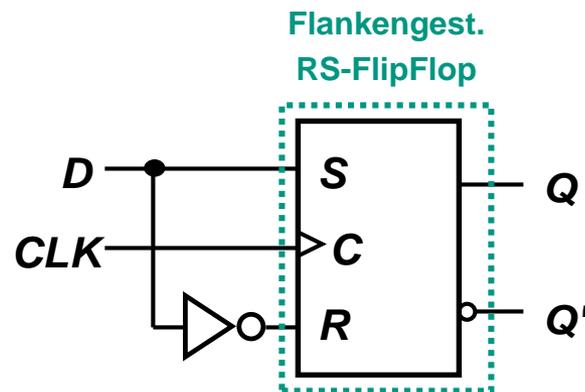
Positive Flanke (rising edge)



Negative Flanke (falling edge)

Flankengesteuertes D-Flipflop:

- Ein Eingang D (Data)
 - $D = HIGH \Rightarrow$ SET-Zustand
 - $D = LOW \Rightarrow$ RESET-Zustand
- Q wird bei **steigender Flanke** des Taktsignals CLK auf D gesetzt

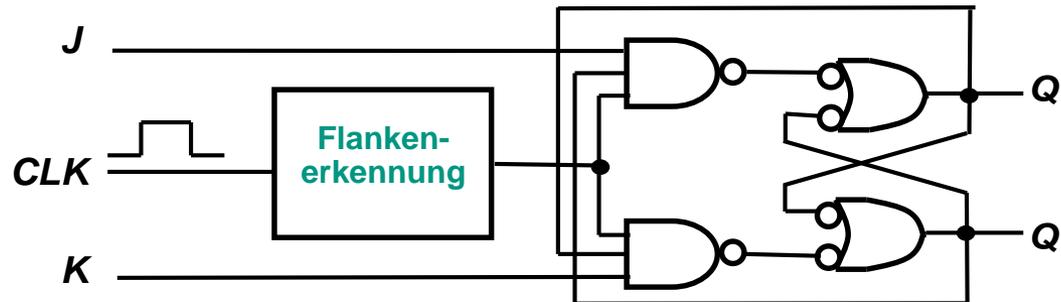


D	CLK	$Q(t+1)$	Zustand
1	\uparrow	1	Set
0	\uparrow	0	Reset

\uparrow = Taktübergang von LOW auf $HIGH$

Ein auf **positive Flanke** triggerndes **D-Flipflop** aus einem **RS-FlipFlop** gebildet

Flankengesteuertes JK-FlipFlop:



- Q und Q' -> rückgekoppelt auf impulsgesteuerte **NAND-Gatter**
- **Keine ungültigen Zustände!**
- Beinhaltet einen **Wechselzustand**:
 - **J = HIGH** (und K = LOW) ⇒ **SET-Zustand**
 - **K = HIGH** (und J = LOW) ⇒ **RESET-Zustand**
 - **beide Eingänge sind auf LOW** ⇒ **keine Änderung**
 - **beide Eingänge sind auf HIGH** ⇒ **Q = Q'**

Charakteristische Tabelle:

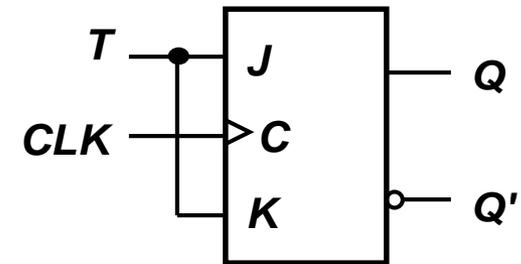
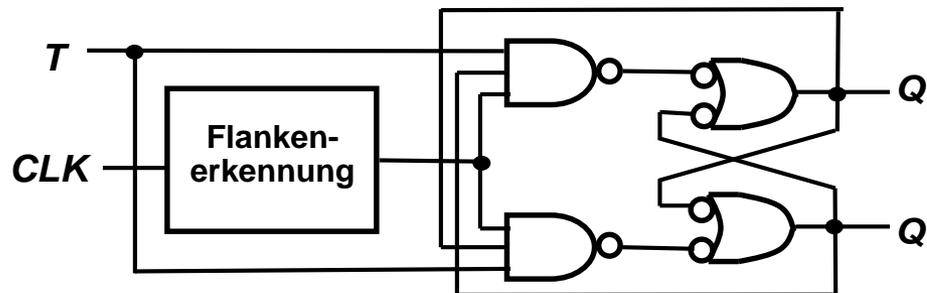
J	K	Takt	Q(t+1)	Zustand
0	0	↑	Q(t)	halten
0	1	↑	0	Reset
1	0	↑	1	Set
1	1	↑	Q(t)'	wechseln

$$Q(t+1) = (J \cdot Q) + (K' \cdot Q')$$

Toggle-Flipflop (T-FlipFlop):

■ 1-Eingang-Version der JK-Flipflops

-> gebildet durch **zusammenschalten** der **J-** und **K-Eingänge**



Charakteristische Tabelle:

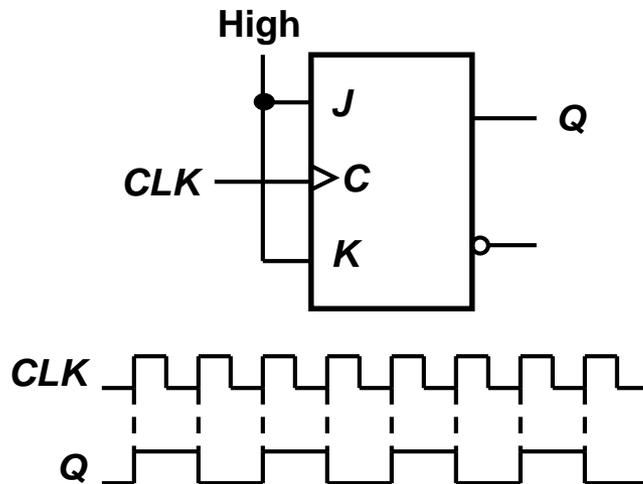
T	CLK	$Q(t+1)$	Zustand
0	↑	$Q(t)$	halten
1	↑	$Q(t)'$	wechseln

Q	T	$Q(t+1)$
0	0	0
0	1	1
1	0	1
1	1	0

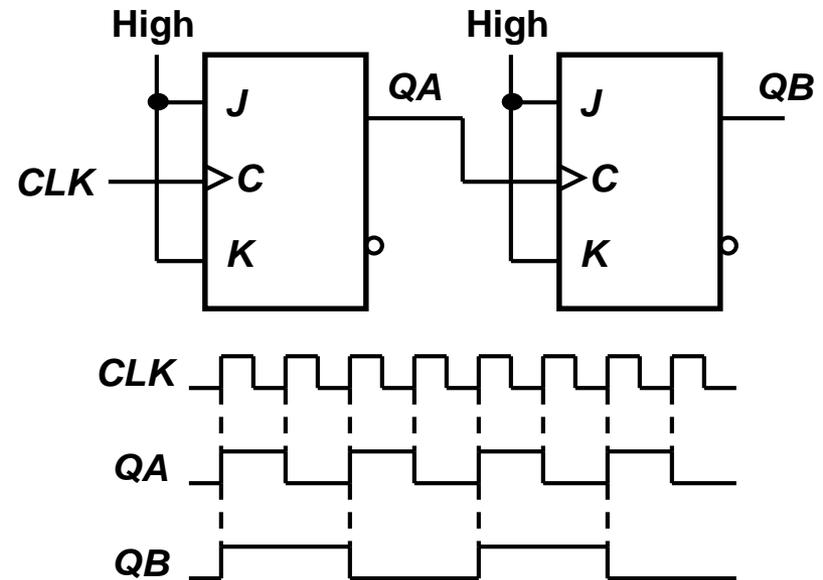
$$Q(t+1) = (T \cdot Q) + (T' \cdot Q')$$

T-Flipflop:

- Applikation: Frequenzteiler



Taktfrequenz geteilt durch 2.

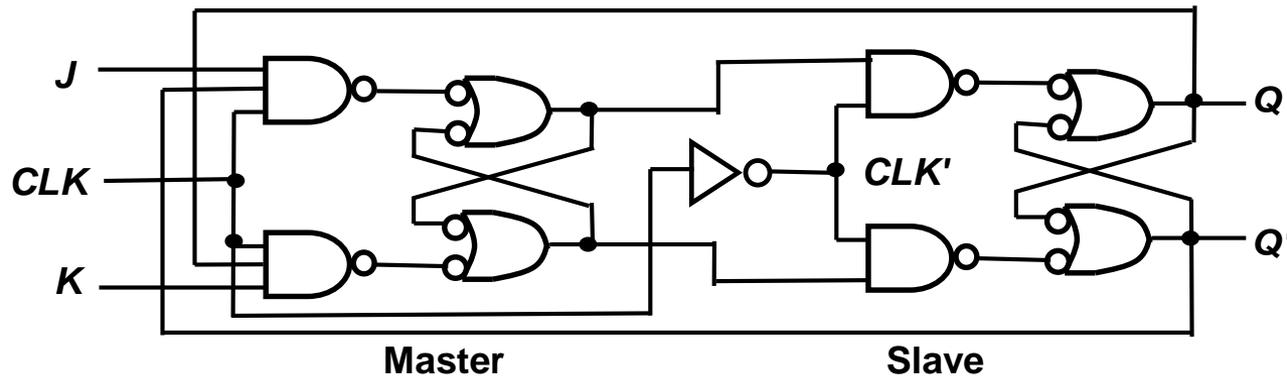


Taktfrequenz geteilt durch 4.

Flankengesteuertes Master-Slave-FlipFlop:

- *Master* ist aktiviert bei steigender Flanke, *Slave* ist aktiviert bei fallender Flanke
-> vollständige Entkopplung des Ausgangs vom Eingang

Master-Slave-JK-FlipFlop:



J	K	CLK	$Q(t+1)$	Zustand
0	0		$Q(t)$	Halten
0	1		0	Reset
1	0		1	Set
1	1		$Q(t)'$	invertieren