



Kapitel 12: Realisierung digitaler Bausteine

Transistoren

Logische Gatter

Flip Flops

1-Bit Speicherzelle

Schaltpläne und ihre Belegungen

Definition (Schaltplan):

Ein Schaltplan SP wird definiert wie ein Schaltkreis: $SP = (X_n, Y_n, G, typ, in, out)$, wobei X_n, Y_n, typ, in, out wie in Schaltkreisen und G ein nicht notwendigerweise azyklischer Graph.

Definition (Belegung von Schaltplänen):

Sei $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n) \in \mathbb{B}^n$ eine Belegung und $\Psi_\alpha: S \rightarrow \{0,1\}$ eine Belegung, die den Signalen stabile Belegungen zuordnet über:

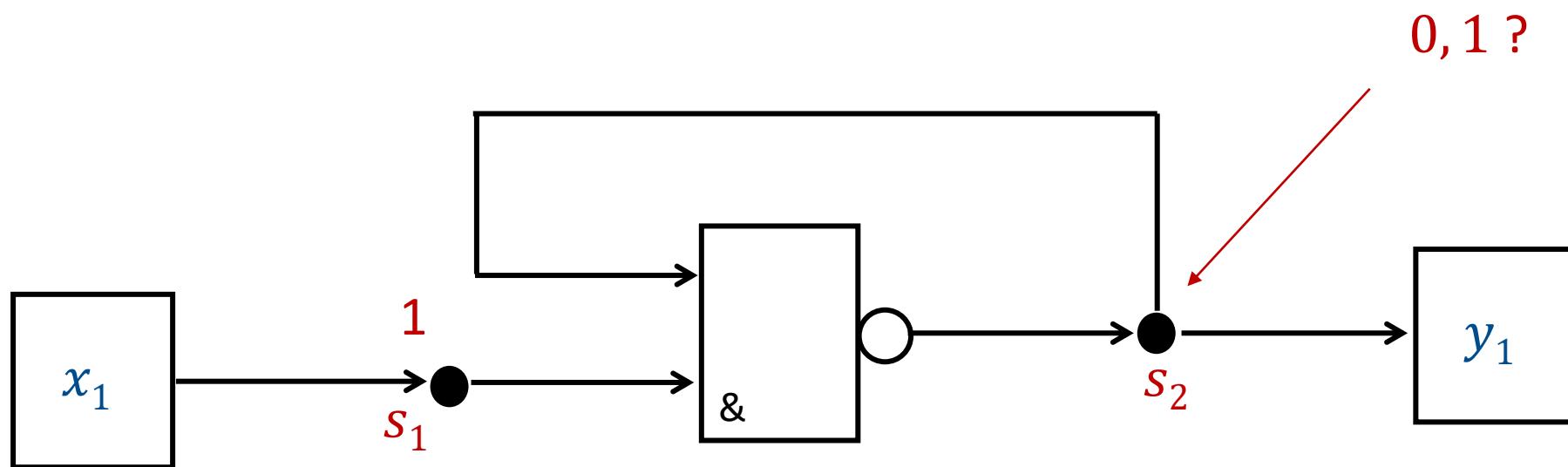
- $\Psi_\alpha(s) = 0$ falls $(0, s) \in E$
- $\Psi_\alpha(s) = 1$ falls $(1, s) \in E$
- $\Psi_\alpha(s) = \alpha_i$ falls $(x_i, s) \in E$
- $(\Psi_\alpha(t_1), \dots, \Psi_\alpha(t_m)) = g_{typ(m)}(\Psi_\alpha(s_1), \dots, \Psi_\alpha(s_k))$ für $m \in M, in(m) = \{s_1, \dots, s_k\}, out(m) = \{t_1, \dots, t_m\}$

Beobachtung:

Es ist möglich, dass es zu einer Eingangsbelegung α

- keine stabile Signalbelegung gibt
- mehrere stabile Signalbelegungen gibt

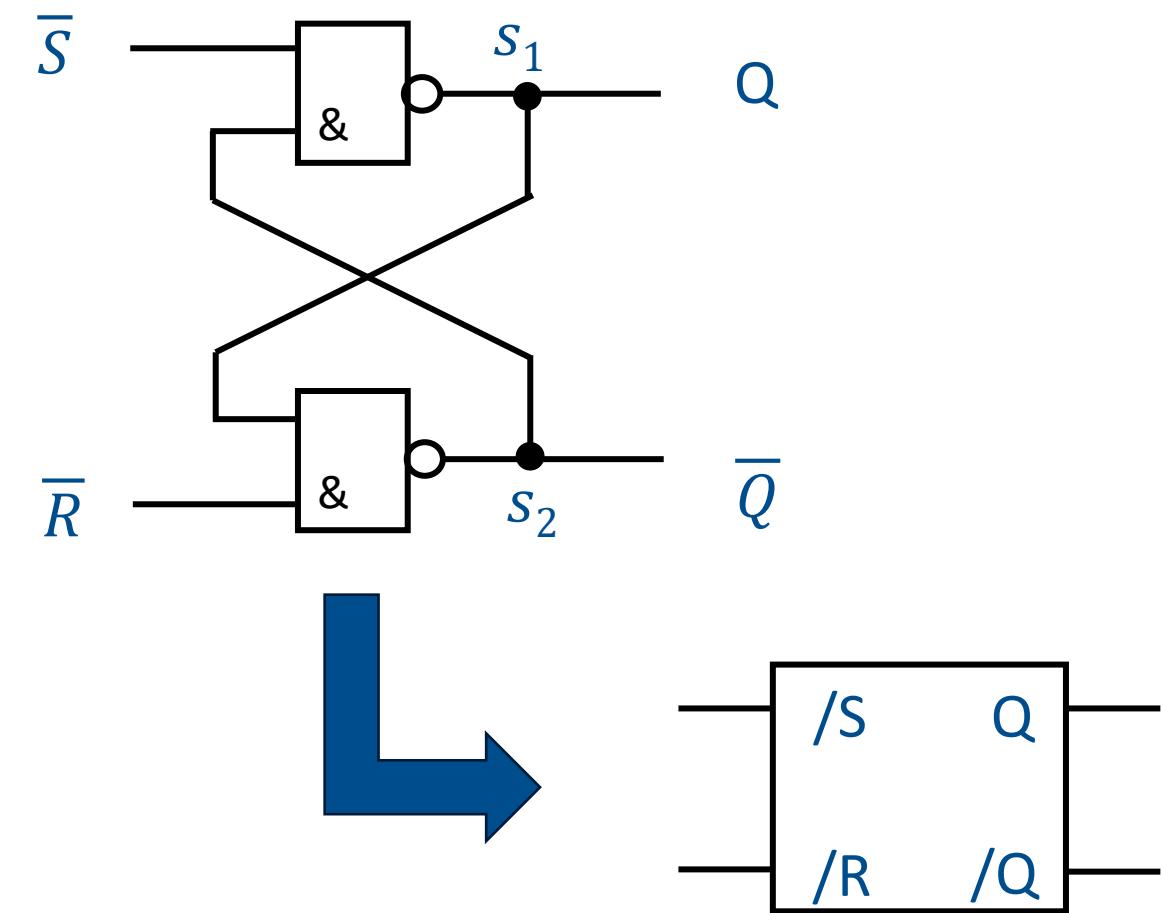
Beispiel a)



Für $x_1 = 1$ existiert keine stabile Signalbelegung (siehe s_2 !)

Beispiel b) RS-Flip-Flop

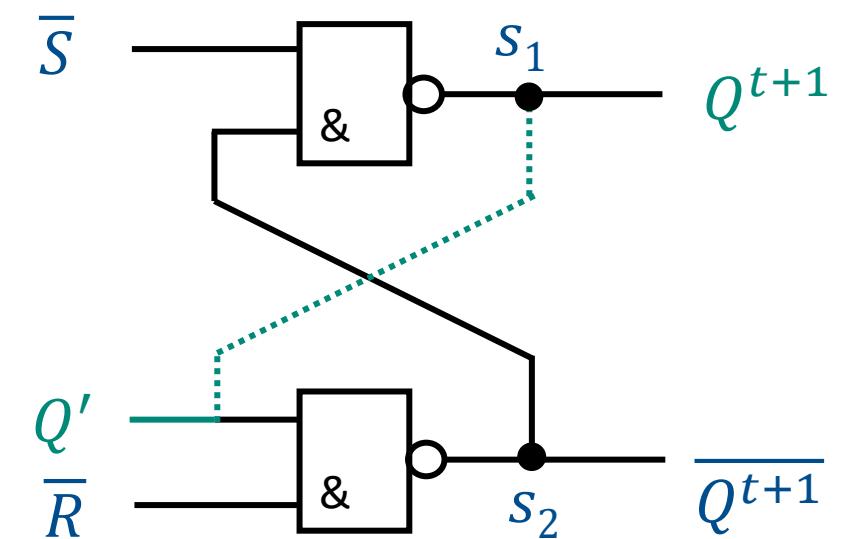
- Nutzung von NAND Gattern zur Existenz mehrerer stabiler Eingangsbelegungen
- Diesmal: Zyklizität „über Kreuz“
- Im nächsten Schritt: Betrachtung nur einer „Rückverknüpfung vom Signal“



Verhalten des RS-Flip-Flop

- Wie verhält sich Q^{t+1} ?
- $Q^{t+1} = s_1 = \overline{\overline{S} \cdot s_2} = \overline{\overline{S} \cdot Q' \cdot \overline{R}}$
- Jetzt setzen wir $Q' = Q^t$:

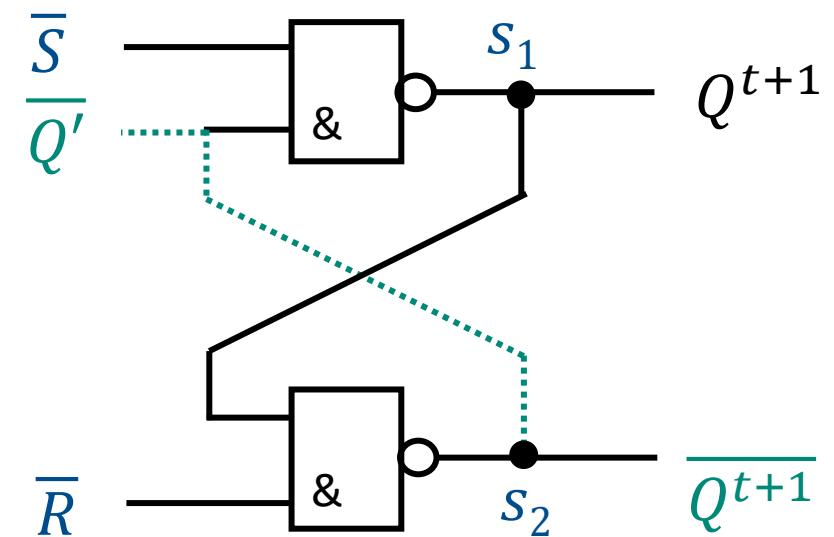
\overline{S}	\overline{R}	Q^{t+1}
0	0	1
0	1	1
1	0	0
1	1	Q^t



Verhalten des RS-Flip-Flop

- Wie verhält sich $\overline{Q^{t+1}}$?
- $\overline{Q^{t+1}} = s_2 = \overline{\overline{R} \cdot s_1} = \overline{\overline{R} \cdot \overline{Q'} \cdot \overline{S}}$
- Jetzt setzen wir $Q' = Q^t$:

\overline{S}	\overline{R}	Q^{t+1}	$\overline{Q^{t+1}}$
0	0	1	1
0	1	1	0
1	0	0	1
1	1	Q^t	$\overline{Q^t}$



RS-Flip-Flop

Für Eingangsbelegung $S = 1, R = 1$ gibt es zwei stabile Belegungen

- $\Psi_{(1,1)}(s_1) = 0, \Psi_{(1,1)}(s_2) = 1$
- $\Psi'_{(1,1)}(s_1) = 1, \Psi'_{(1,1)}(s_2) = 0$

Bei Ψ spricht man von Zustand $Q = 0$, bei Signalbelegung Ψ' von Zustand $Q = 1$

Bezeichnungen

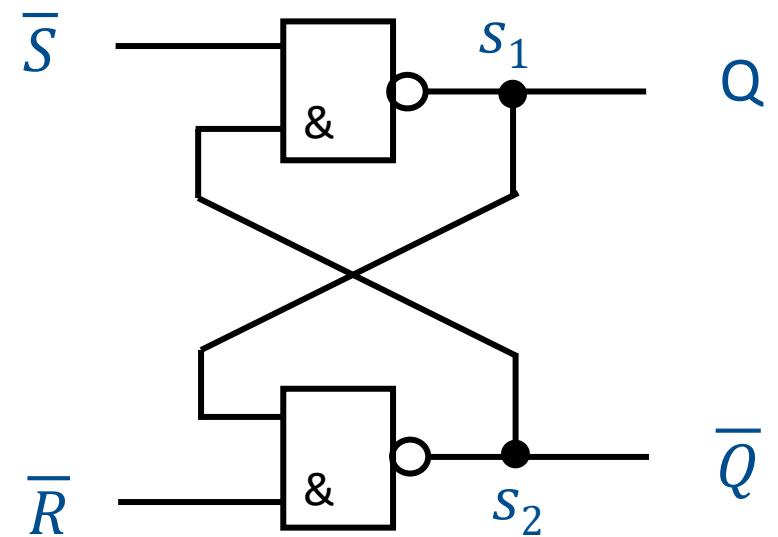
- Im Folgenden wird Flip-Flop durch FF abgekürzt
- Umschalten des FF in Zustand $Q = 1$ heißt Setzen (set)
- Umschalten des FF in Zustand $Q = 0$ heißt Zurücksetzen (reset).
- $/S$ heißt Set-Signal.
- $/R = /C$ heißt Reset- oder Clear-Signal.
- Weil $/R, /S$ durch Absenken aktiviert werden, nennt man sie „active low“
- Signalnamen von Active-Low-Signalen beginnen in der Regel mit $/$

Flackern:

Zu vermeiden ist:

- $/S, /R$ beide aktiv, d.h. $/S = 0, /R = 0$
 $\rightarrow Q = 1, /Q = 1$
- Wenn, dann
 - $/S, /R$ gleichzeitig inaktiv und
 - Gatter gleich schnell schalten

\Rightarrow Flackern = metastabiler Zustand



Nachteil von RS-FF

Beim Speichern eines Wertes 0 oder 1 muss man den Wert kennen:

- 0 → Aktiviere /R
- 1 → Aktiviere /S

Ziel:

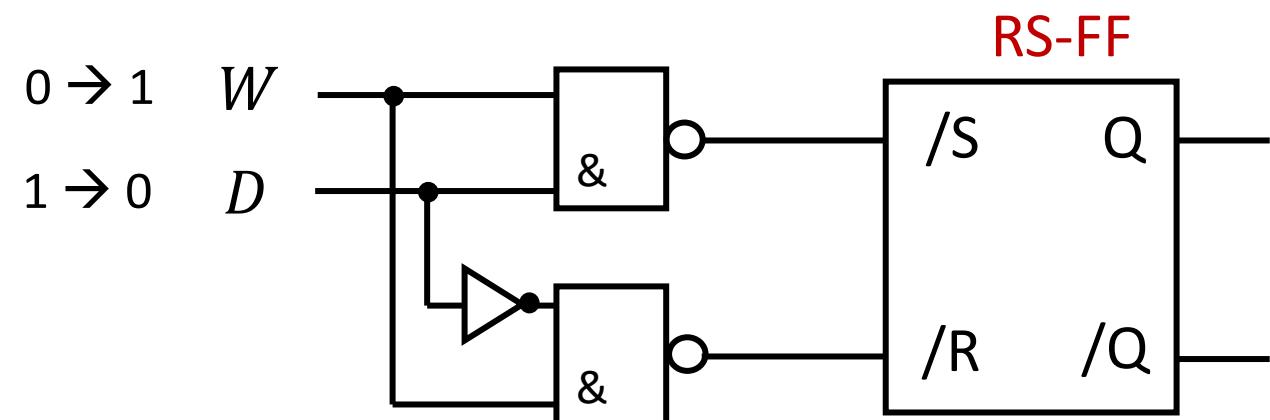
Speichern *unbekannter* Werte

D-Latch - graphisch

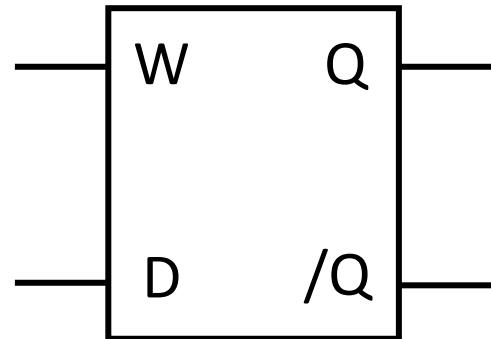
- W ist **active high!**

$W = 0 \Rightarrow /S, /R$ inaktiv

$W = 1 \Rightarrow \begin{cases} /S \text{ aktiv, falls } D = 1 \\ /R \text{ aktiv, falls } D = 0 \end{cases}$

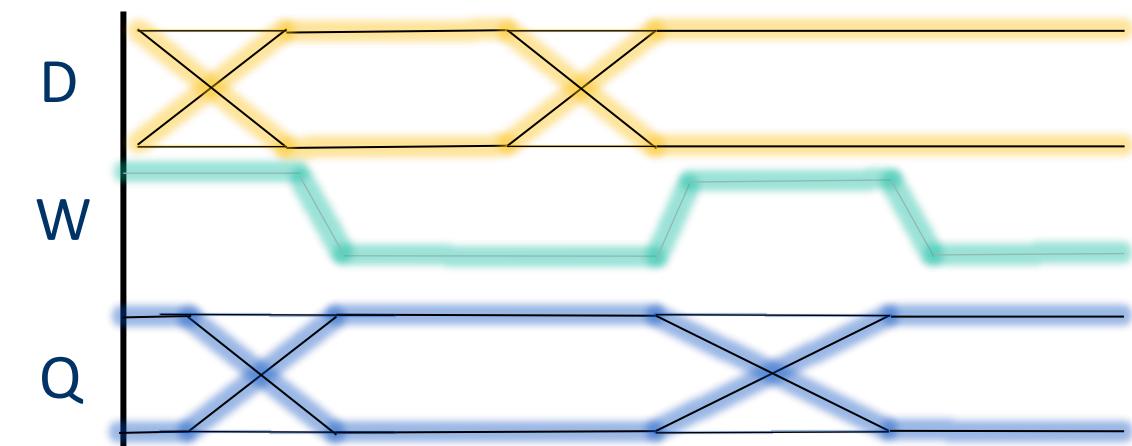


Symbol:



Eigenschaften eines D-Latches

- D-Latch heißt transparent, wenn das Schreibsignal aktiv ist.
- W muss lange genug aktiv sein, damit sich der neue Zustand im RS-FF einstellen kann.
- D-Latch ist pulsgesteuert (Schreibpuls W)



D-Flip-Flop

Taktflankengesteuerte Flip-Flops wie das D-Flip-Flop übernehmen Daten zu einem bestimmten Zeitpunkt (kein transparenter Modus!), nämlich bei der steigenden Flanke des sog. Clocksignals.

Vorteil:

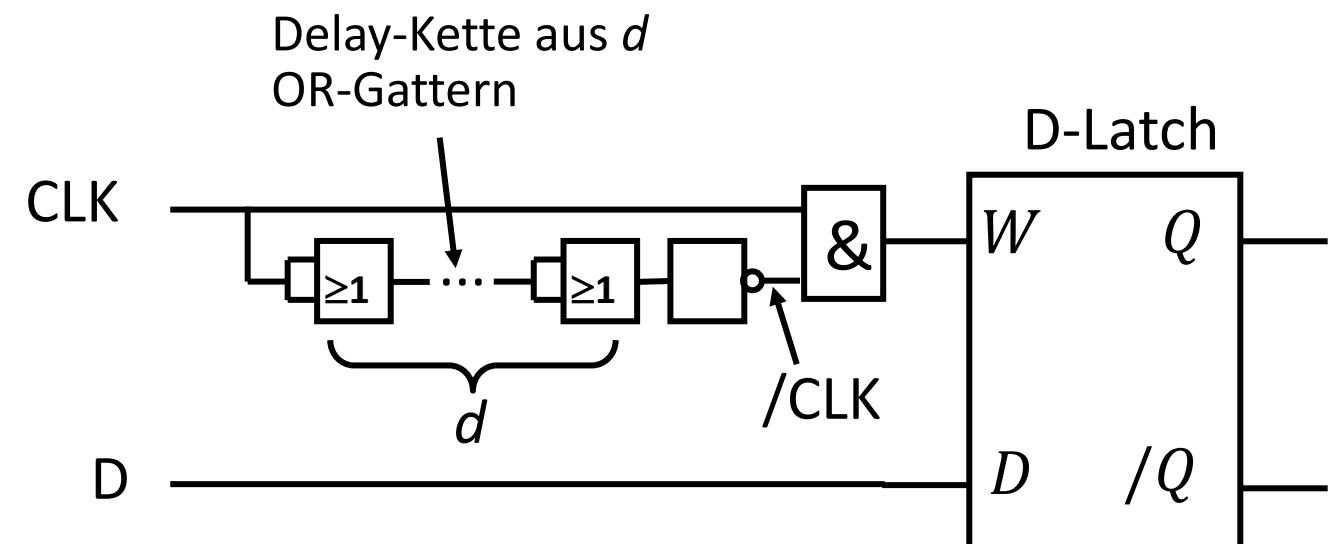
Daten müssen lediglich bei der steigenden Taktflanke stabil sein (zzgl. Setup- und Hold-Zeit)

D-Flip-Flop aus D-Latch

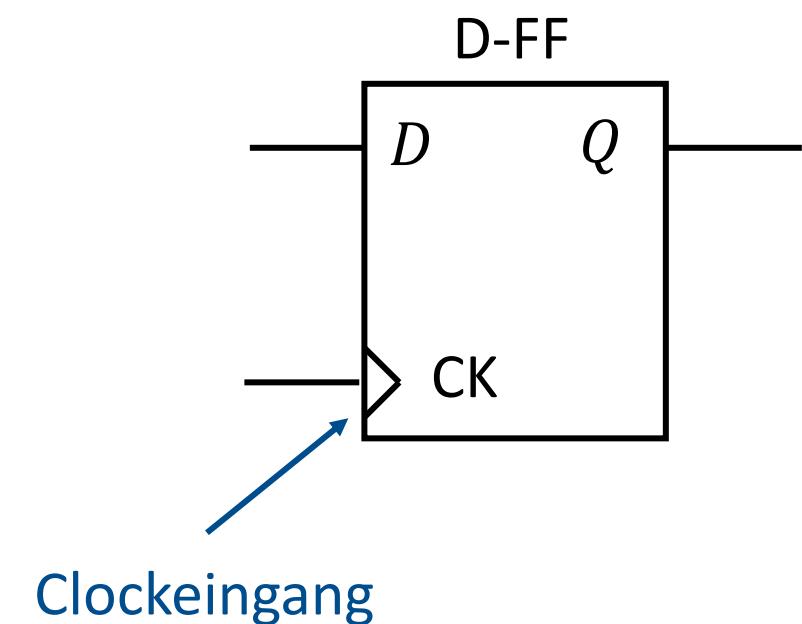
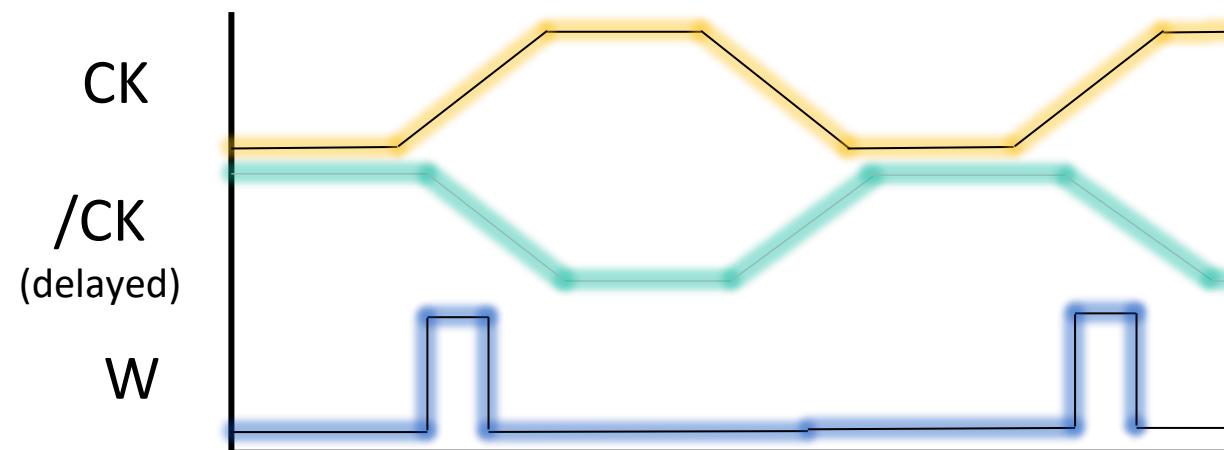
- Prinzipielle Möglichkeit, einen D-FF aus einem D-Latch zu konstruieren:

Idee:

- Erzeuge mit Delay-Kette aus Clocksignal einen Schreibpuls

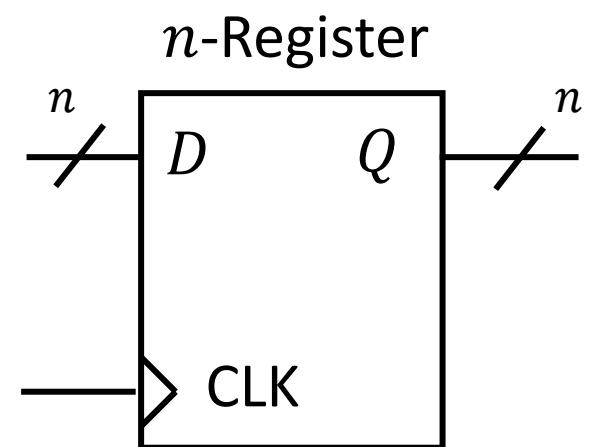


D-Flip-Flop: Symbol

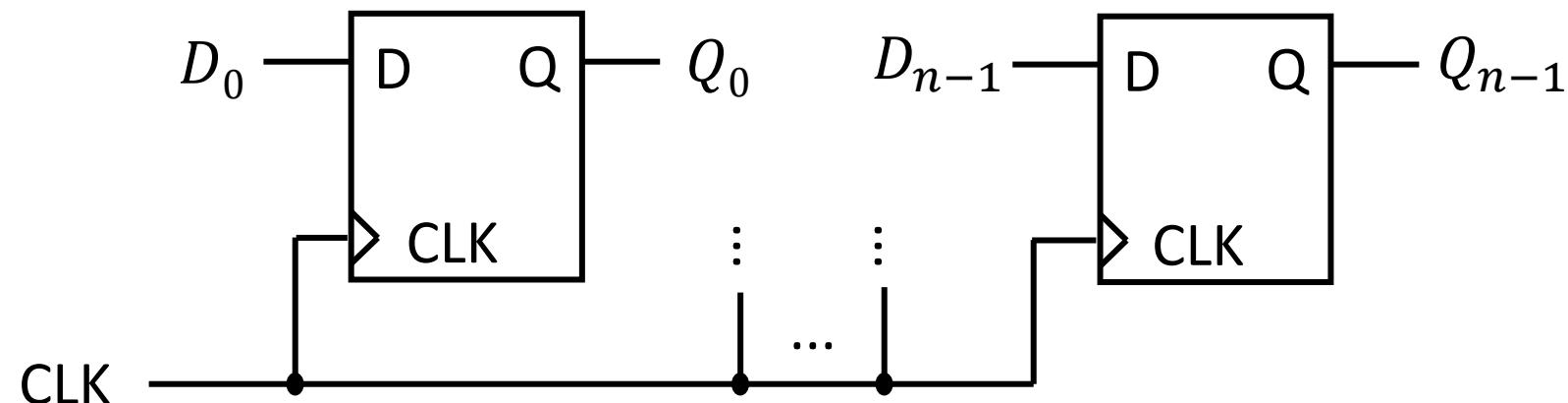


Register

n -Bit-Register = n D-Flip-Flops mit
gemeinsamem Clocksignal



Aufbau eines Registers

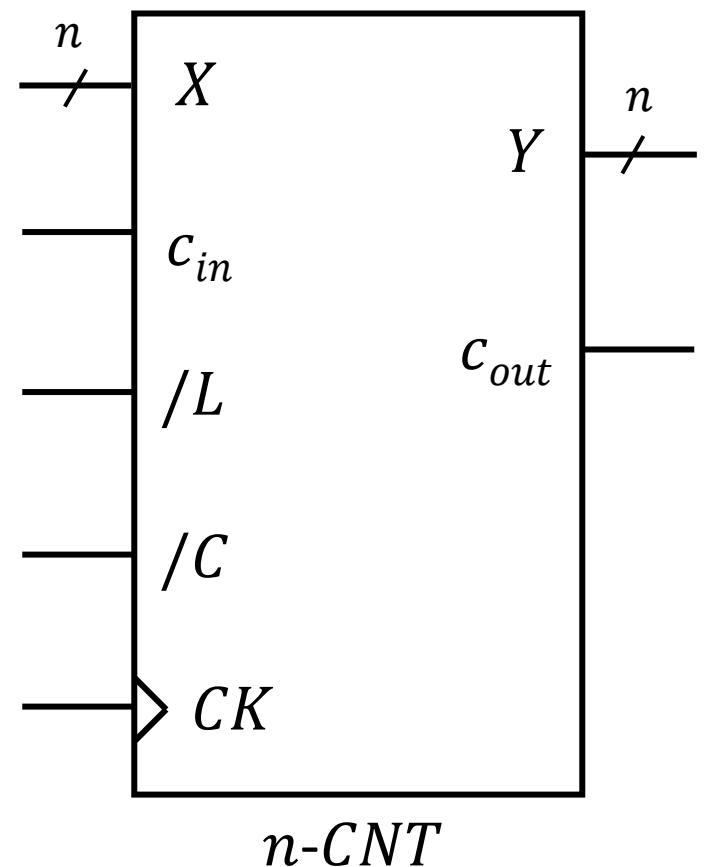


(Entsprechend: n -Bit-Latch = n D-Latches mit gemeinsamem Schreibsignal W)

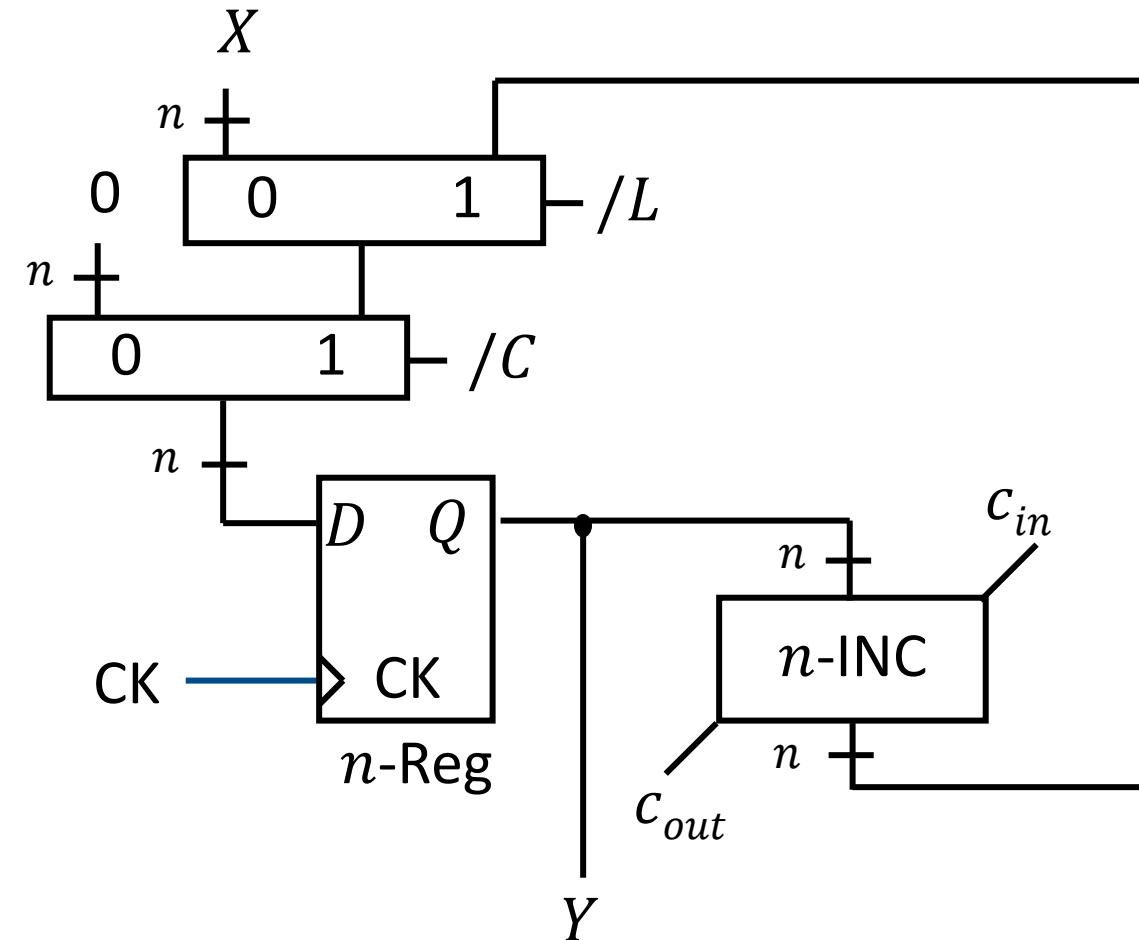
Zähler

Ein **n -Bit-Zähler** ist eine Schaltung mit folgenden Ein- und Ausgängen:

- Dateneingänge $X = (X_{n-1}, \dots, X_0)$
- Datenausgänge $Y = (Y_{n-1}, \dots, Y_0)$
- Dateneingang c_{in} für Eingangsübertrag
- Datenausgang c_{out} für Ausgangsübertrag
- Eingänge für Kontrollsignale:
 - Clear: $/C$
 - Load: $/L$
 - Clock: CK



Aufbau eines Zählers



Funktionsweise eines Zählers (1)

- Ein Zähler speichert ein n -Bit-Wort, das an den Ausgängen Y erscheint. (*Zählerstand*)
- Bei jeder steigenden Flanke von CLK wird ein neuer Zählerstand Y_{neu} gespeichert. Für Y_{neu} gilt:

$$Y_{neu} = \begin{cases} 0 \dots 0 & , \text{falls } /C = 0 \\ X & , \text{falls } /C = 1, /L = 0 \\ \text{bin}_n((Y + C_{in}) \bmod 2^n) & , \text{falls } /C = 1, /L = 1 \end{cases}$$

Funktionsweise eines Zählers (2)

Der Ausgangsübertrag c_{out} ermöglicht es, den Zähler zu *kaskadieren*, z.B. aus k n -Bit-Zählern einen $k \cdot n$ -Bit-Zähler zu bauen.



Kapitel 12: Realisierung digitaler Bausteine

Transistoren

Logische Gatter

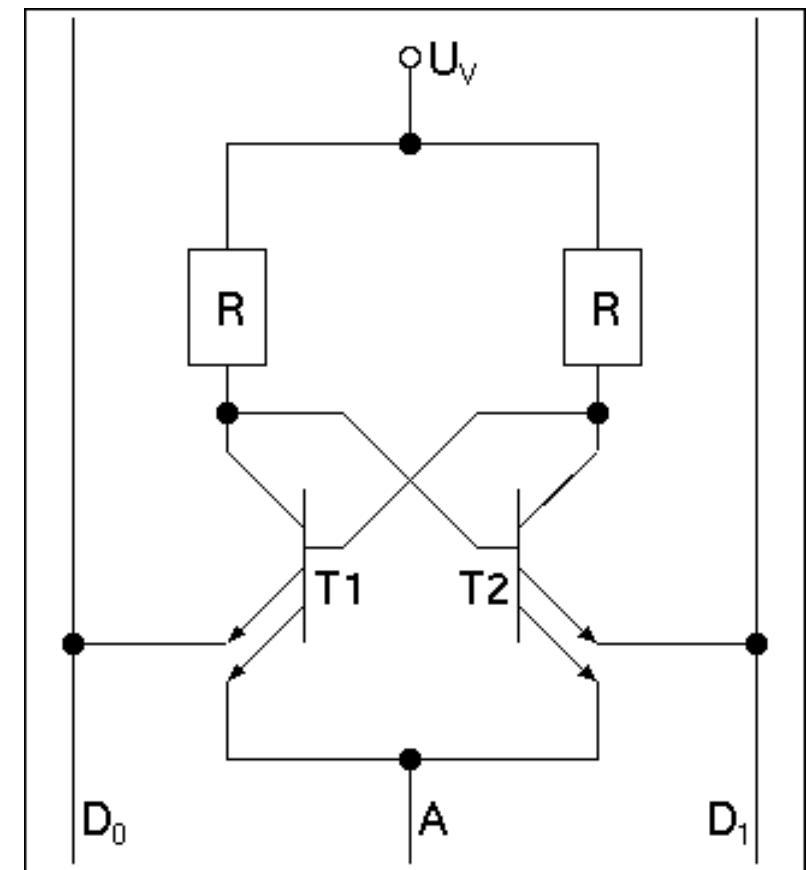
Flip Flops

1-Bit Speicherzelle

1-Bit-Speicherzelle (1)

- D_0 : „0“-Datenleitung
- D_1 : „1“-Datenleitung
- A: Adressleitung

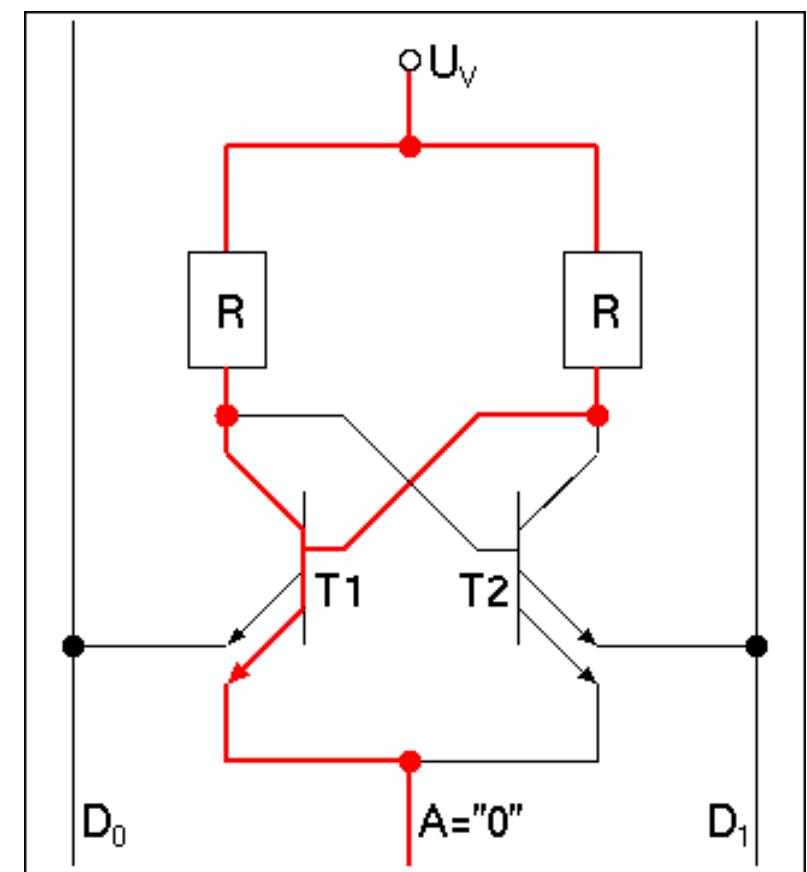
Grundschaltung:



1-Bit-Speicherzelle (2)

- T1 ist *leitend* heißt: die Speicherzelle hat eine „0“ gespeichert.
- Analog dazu ist T2 *leitend*, wenn eine „1“ gespeichert ist.

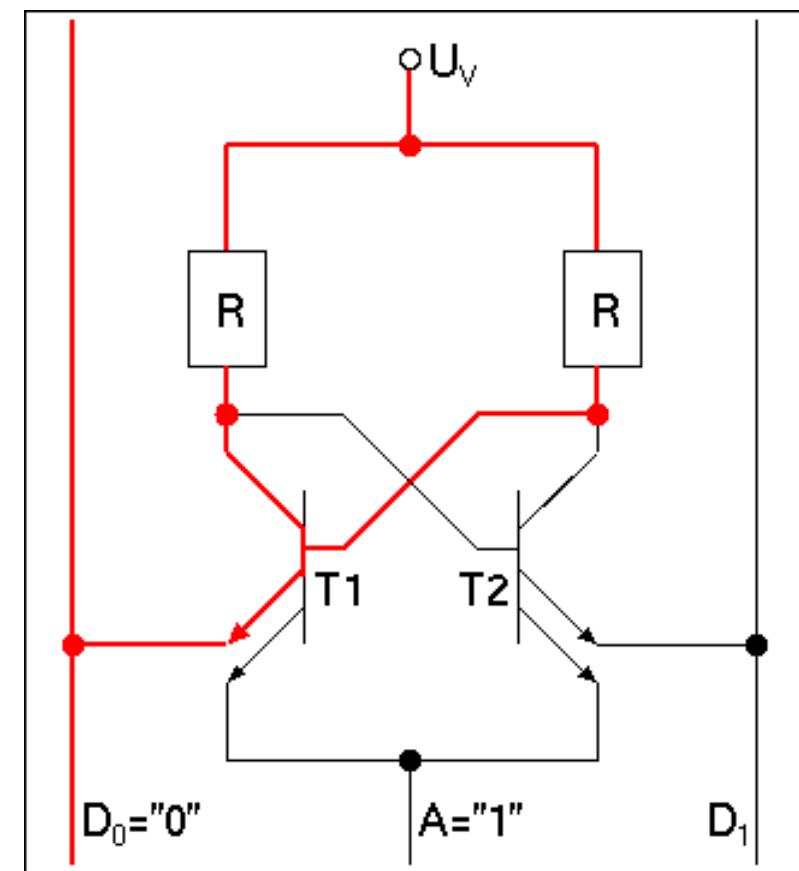
Gespeicherte „0“:



1-Bit-Speicherzelle (3)

- Die Adressleitung wird auf „1“ gesetzt, um die Speicherzelle anzusprechen.
- Beide Datenleitungen werden auf „0“ gesetzt.
- Ist eine „0“ („1“) gespeichert, so fließt ein Strom auf der „0“-Datenleitung („1“-Datenleitung).

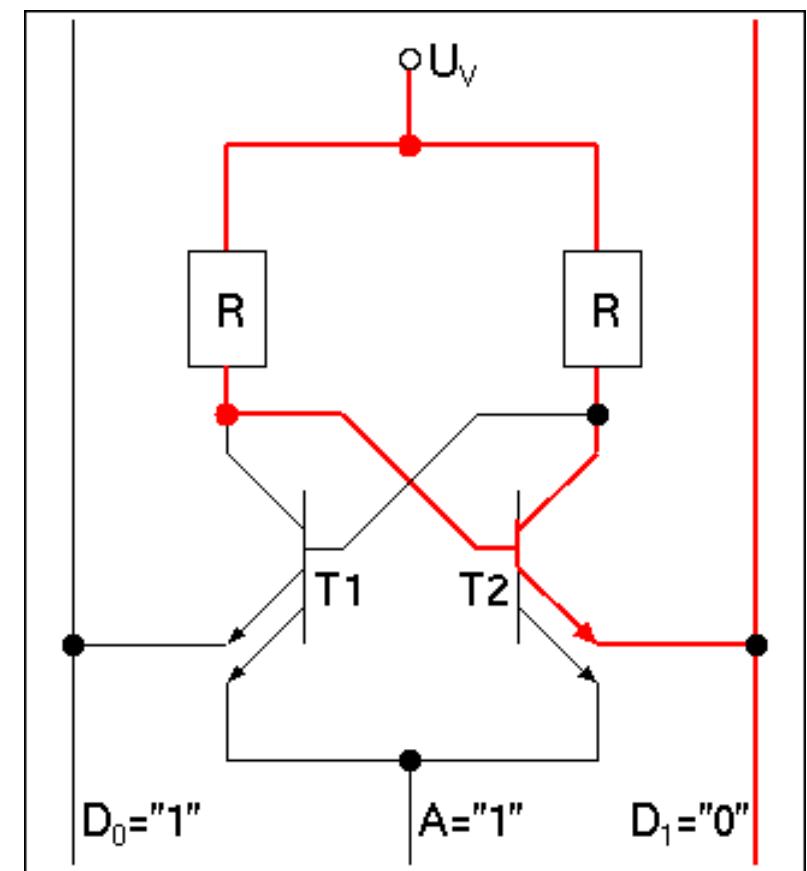
Wert lesen:



1-Bit-Speicherzelle (4)

- Die Adressleitung wird auf „1“ gesetzt.
- Die „1“-Datenleitung wird auf „0“ gesetzt.
- T2 wird leitend, T1 gesperrt.

„1“ speichern:



1-Bit-Speicherzelle (5)

- Die gezeigte Speicherzelle benötigt zusätzliche Logik zum Schreiben und Lesen:
 - Ein Steuersignal muss angeben, ob Daten gelesen oder geschrieben werden.
 - Aus dem Steuersignal ergeben sich die Werte für die Datenleitungen.
 - Beim Lesen muss der Strom auf eine Spannung abgebildet werden (→ Transistor).
- Ein D-Flip-Flop ist **leichter anzusteuern**, benötigt aber **mehr Platz**.

Wortweise adressierbarer Speicher

Speicher für $2^m n$ -Bit-Worte

