Technische Grundlagen der Informatik – Kapitel 3

Prof. Dr. Jan Peters
Fachgebiet Intelligente Autonome Systeme (IAS)
Fachbereich Informatik

TECHNISCHE UNIVERSITÄT DARMSTADT

WS 14/15

Meisenantworten

TECHNISCHE UNIVERSITÄT DARMSTADT

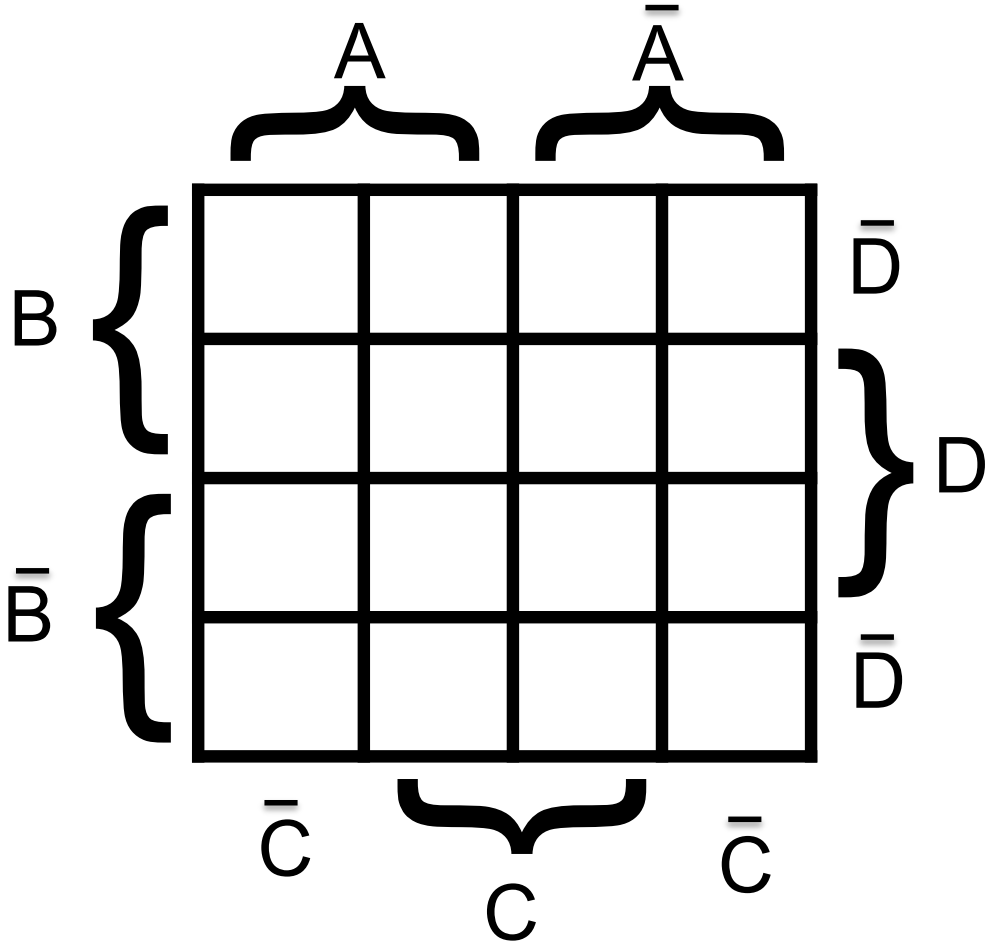
- Tolle Tips zu Karnaugh Diagramme, Multiplexer. *Danke*!
- Skript-Runterladen. *Auf Print drücken*. View | Print | Student Logout (jp14pete) | Tutor Login
- Komplimente! *Ich freue mich*! *Danke*!
- Kapitel 2: Aufhebung der beiden NOTs auf Folien 51/52 sollte passieren. *Schau mal auf Folie 53*!
- Lehre Selbsttests:
  - Antwortsverteilung? *Vielleicht später*…
  - Mehr Zeit für die erste Frage! *OK*…
  - Größerer “Neu Laden” Button für Handys. *Später, OK*!
  - “document.window.reload” problematisch mit Android. *Vorschläge*?
  - “document.window.reload” besser implementieren u. verhindern dass alte Antworten übertragen wird. *Vorschläge*?
- Bitte für Ruhe Sorgen. *Soll ich Chris & Herke mit Baseballschläger rumschicken*? *Ich werde mich bemühen und habe ein langes Gedächtnis*!

Meisenantworten

TECHNISCHE UNIVERSITÄT DARMSTADT

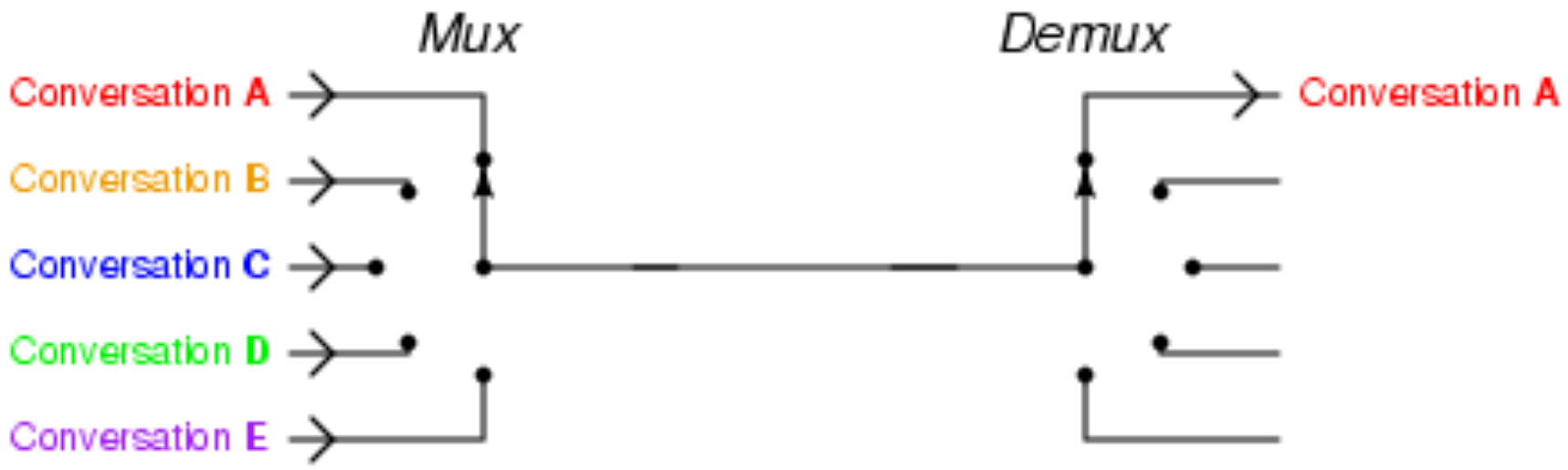
- Sniggers-Statistik-Analyse. *Danke für die detailierten Informationen*.
- Wurfanalyse. *Dankeschön*! *Ich weiss, mein Wurfstil ist nicht gut – aber unterhaltsam*?
- Sniggerverteilung:
  - Stratified sampling (weibl. Sniggers ~ weibl. Stud/ges. Stud.). *OK*
  - Programm schreiben. *Hmm*…
  - Roboter dafür? Als Mastarbeit! *Au ja*! *Bitte komm zu mir wenn Du soweit bist*!
- "Wann ein Baum im Wald umfällt, aber keiner hörts, macht er trotzdem ein Geräusch?"
- "Wann bekommst Du Dein Tablet wieder?" Wurde leider gerade mit einem noch sehr schwachen Linux verunsichert…

Tolles Verständnis für Karnaugh Diagramme (*ein Meisentip*!)



Vorteile:
Einfacher
Anschaulicher
Lesbarer

Multiplexer:



http://de.wikipedia.org/wiki/Multiplexer

Kapitel 3: Themen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Einleitung
- Latches und Flip-Flops
- Entwurf synchroner Logik
- Endliche Zustandsautomaten
- Zeitverhalten sequentieller Logik
- Parallelismus

Einleitung

TECHNISCHE UNIVERSITÄT DARMSTADT

- Ausgänge sequentieller Logik hängen ab von
  - aktuellen Eingabewerten
  - vorherigen Eingabewerten
- Schaltung speichert einen internen Zustand

- Definitionen
  - Zustand: interne Informationen, aus denen weiteres Schaltungsverhalten hergeleitet werden kann
  - Latches und Flip-Flops: Speicherelemente für jeweils 1 Bit Zustand
  - Synchrone sequentielle Schaltung: Kombinatorische Logik gefolgt von Flip-Flops

Sequentielle Schaltungen

TECHNISCHE UNIVERSITÄT DARMSTADT

- Können Folgen von Ereignissen bearbeiten
- Haben “Gedächtnis” (in der Regel nur Kurzzeit-)

- Benutzen Rückkopplungen von Logikausgängen zu Logikeingänge, um Informationen zu speichern
  - Rückkopplungen: Keine kombinatorischen Schaltungen mehr!

Zustandselemente

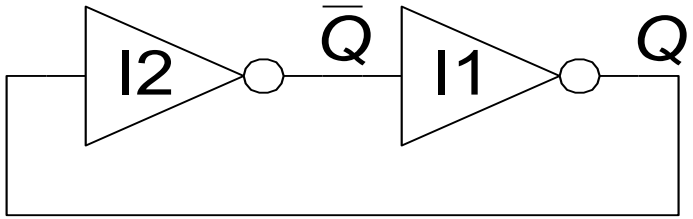
TECHNISCHE UNIVERSITÄT DARMSTADT

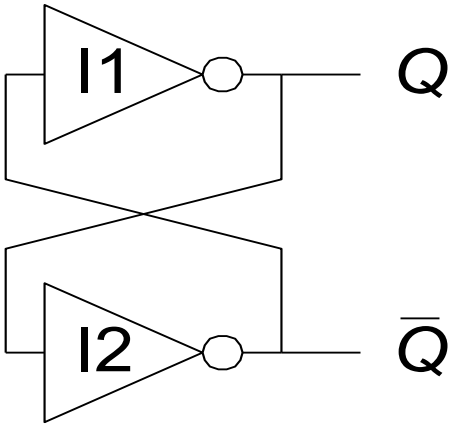
- Der Zustand einer Schaltung beeinflusst das zukünftige Verhalten

- Speicherelemente speichern Zustand
  - Bistabile Schaltungen
  - SR Latch
  - D Latch
  - D Flip-Flop
  - Manchmal auch Zustandselemente genannt

Bistabile Grundschaltung

- Fundamentaler Baustein der anderen Speicherelemente
- Zwei Ausgänge: Q, $\overline{Q}$
- Keine Eingänge

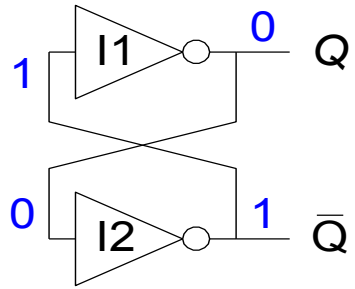




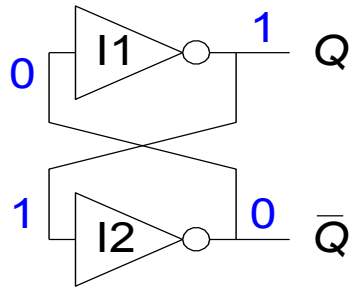
Analyse der bistabilen Grundschaltung

TECHNISCHE UNIVERSITÄT DARMSTADT

- Betrachte zwei Möglichkeiten:
  - $Q = 0$: dann $\overline{Q} = 1$ und $Q = 0$
  - Konsistent und stabil



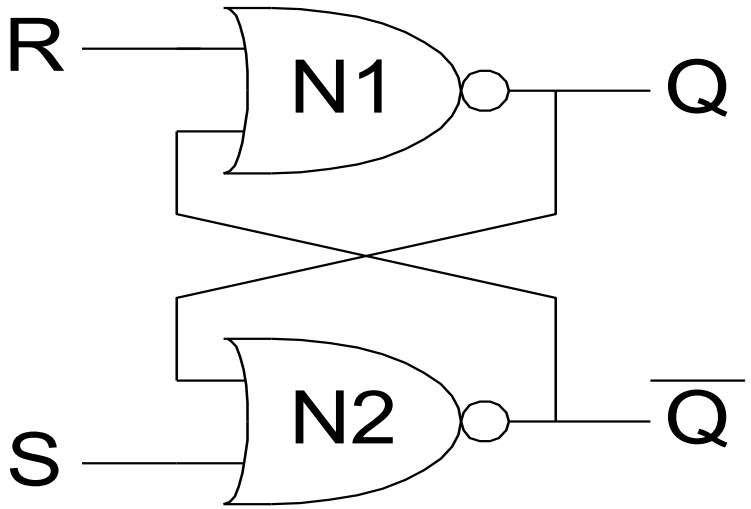
  - $Q = 1$: dann $\overline{Q} = 0$ und $Q = 1$
  - Konsistent und stabil



- Bistabile Schaltung speichert 1 Zustandsbit in Zustandsvariable Q (oder $\overline{Q}$)
- Es gibt aber bisher **keine Eingänge, um diesen Zustand zu beeinflussen**

SR (Setzen/Rücksetzen) Latch

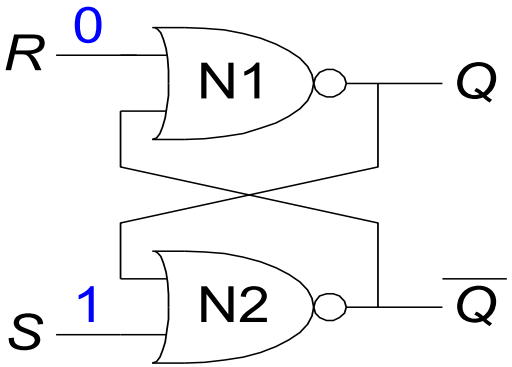
- SR Latch



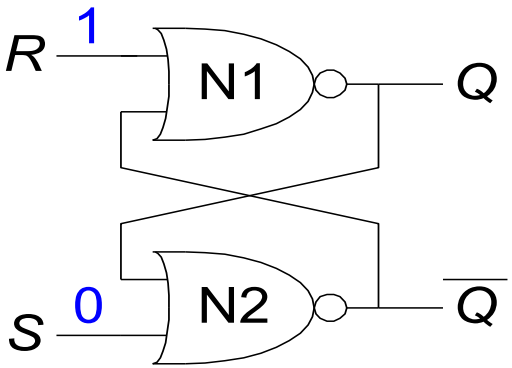
- Betrachte Fälle:
  - $S = 1, R = 0$
  - $S = 0, R = 1$
  - $S = 0, R = 0$
  - $S = 1, R = 1$

Analyse des SR Latches

- $S = 1$, $R = 0$: dann $Q = 1$ und $\overline{Q} = 0$

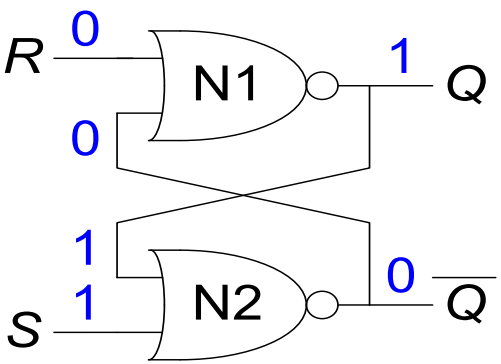


- $S = 0$, $R = 1$: dann $Q = 0$ und $\overline{Q} = 1$

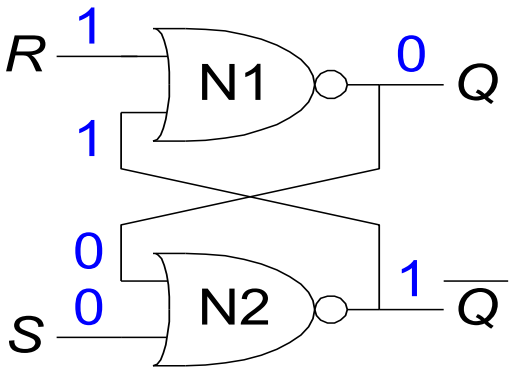


Analyse des SR Latches

- $S = 1$, $R = 0$: dann $Q = 1$ und $\overline{Q} = 0$



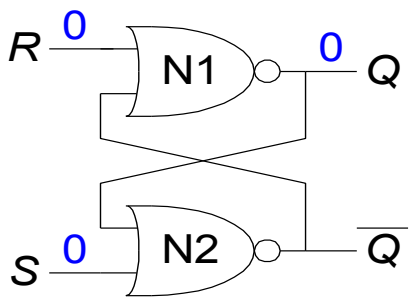
- $S = 0$, $R = 1$: dann $Q = 0$ und $\overline{Q} = 1$



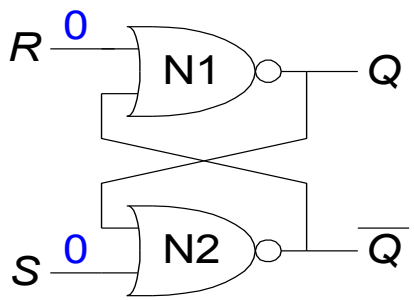
Analyse des SR Latches

- $S = 0$, $R = 0$: dann $Q = Q_{prev}$

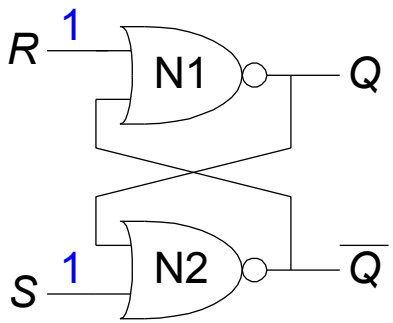
$Q_{prev} = 0$



$Q_{prev} = 1$



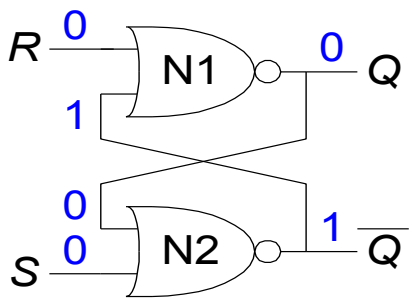
- $S = 1$, $R = 1$: dann $Q = 0$ und $\overline{Q} = 0$



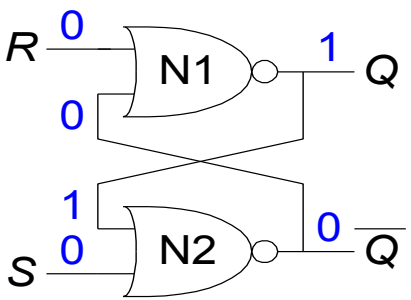
Analyse des SR Latches

- $S = 0$, $R = 0$: dann $Q = Q_{prev}$ und $\overline{Q} = \overline{Q_{prev}}$ (gespeichert!)

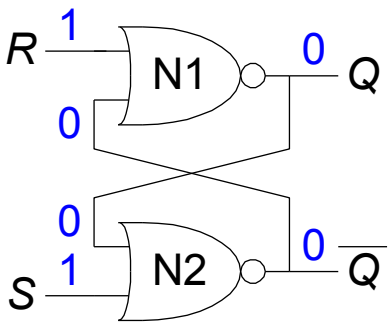
$Q_{prev} = 0$



$Q_{prev} = 1$



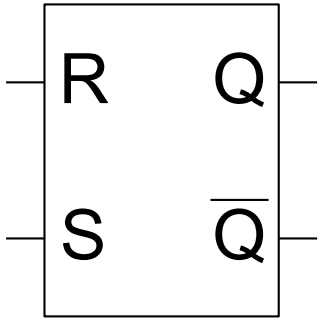
- $S = 1$, $R = 1$: dann $Q = 0$ und $\overline{Q} = 0$ (ungültiger Zustand: $Q \neq \text{NOT } \overline{Q}$)



Schaltplansymbol für SR Latch

- SR steht für Setzen/Rücksetzen Latch (*set/reset*)
  - Speichert ein Bit Zustand (*Q*)
- Festlegen des gespeicherten Wertes mit den *S*, *R* Eingängen
  - Set: Setze Ausgang auf 1 ($S = 1$, $R = 0$, $Q = \mathbf{1}$)
  - Reset: Zurücksetzen des Ausgangs auf 0 ($S = 0$, $R = 1$, $Q = \mathbf{0}$)

SR Latch Symbol

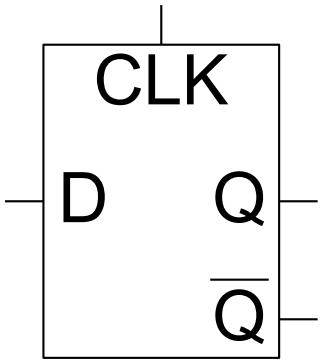


- **Verbotene Zustand vermeiden**
  - **Es darf niemals $S = R = 1$ sein**

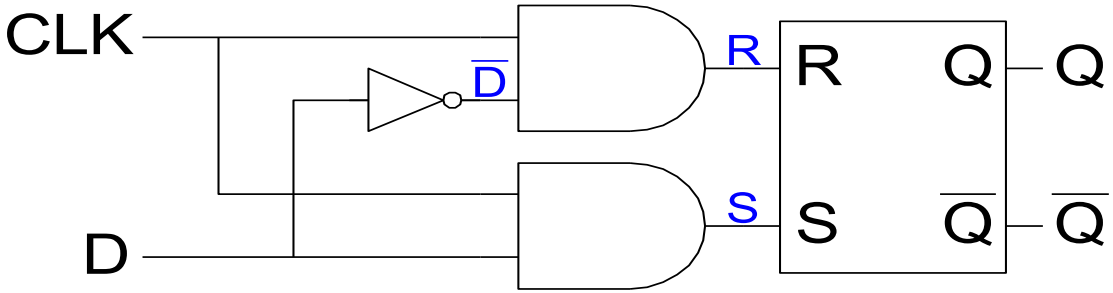
D Latch

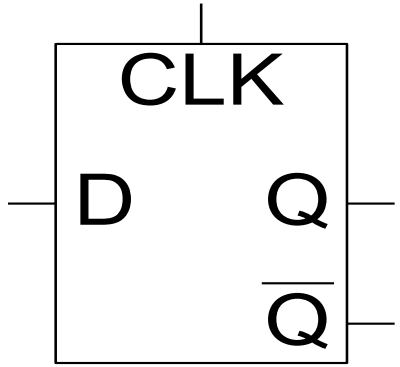
- Zwei Eingänge: *CLK*, *D*
  - *CLK*: steuert, *wann* sich der Ausgang ändert (*clock,* Taktsignal)
  - *D* (der Dateneingang): steuert, auf *was* sich der Eingang ändert
- Funktion
  - Wenn *CLK* = 1 wird *D* weitergereicht an *Q* (das Latch ist transparent)
  - Wenn *CLK* = 0 behält *Q* seinen vorigen Wert (das Latch ist *opak*)
- Illegaler Fall $Q \neq \text{NOT } \overline{Q}$ kann nicht mehr auftreten

D Latch Symbol



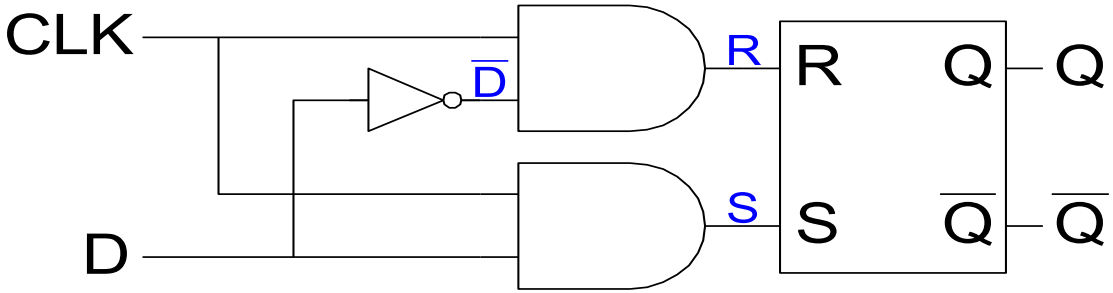
Interner Aufbau eines D Latches

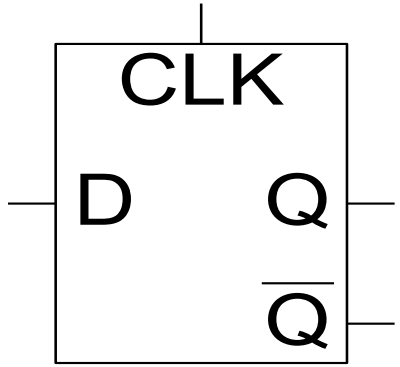




CLK	D	$\overline{D}$	S	R	Q	$\overline{Q}$
0	X					
1	0					
1	1					

Interner Aufbau eines D Latches



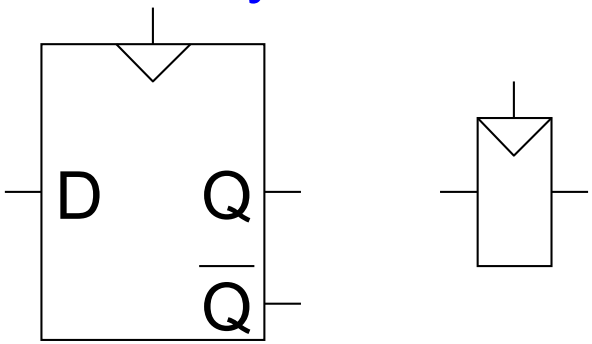


CLK	D	$\overline{D}$	S	R	Q	$\overline{Q}$
0	X	$\overline{X}$	0	0	Q_{prev}	$\overline{Q_{prev}}$
1	0	1	0	1	0	1
1	1	0	1	0	1	0

D Flip-Flop

- Zwei Eingänge: *CLK*, *D*
- *Funktion*
  - Das Flip-Flop liest den aktuellen Wert von *D* bei einer steigenden Flanke von *CLK*
    - Wenn *CLK* von 0 nach 1 steigt, wird *D* weitergegeben zu *Q*
    - Sonst behält *Q* seinen vorigen Wert
  - *Q* ändert sich also nur bei einer steigenden Flanke von *CLK*
- Flip-Flop ist flankengesteuert (*edge-triggered*)
  - Wird bei Flanke des Taktsignals aktiviert

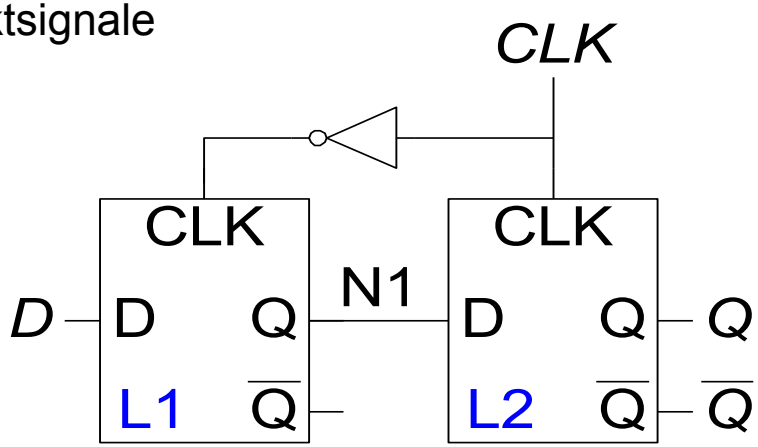
D Flip-Flop
Symbole



Interner Aufbau eines D Flip-Flops

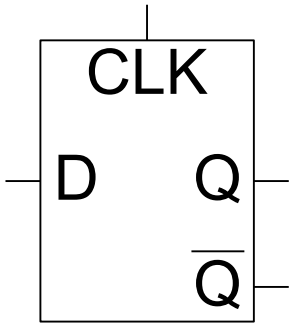
TECHNISCHE UNIVERSITÄT DARMSTADT

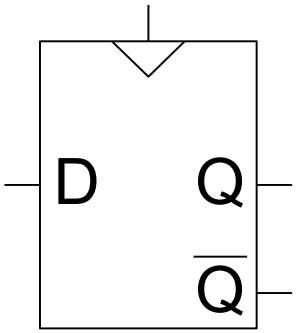
- Zwei Latches in Serie (L1 und L2)
  - … gesteuert durch komplementäre Taktsignale
- Wenn *CLK* = 0
  - … ist L1 transparent
  - … ist L2 opak
  - *D* wird bis N1 weitergegeben
- Wenn *CLK* = 1
  - … ist L2 transparent
  - … ist L1 opak
  - N1 wird an *Q* weitergegeben
- Bei steigender Flanke von CLK (Wechsel von 0 ➔ 1)
  - *D* wird an *Q* weitergegeben

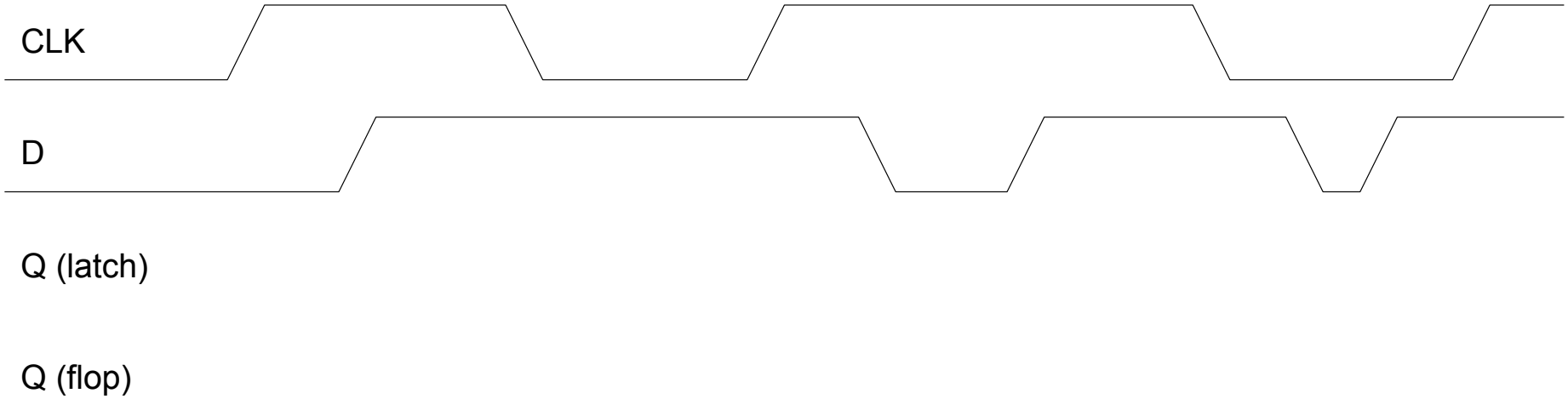


LEHRE WIKI TEST IN ZWEI FOLIEN

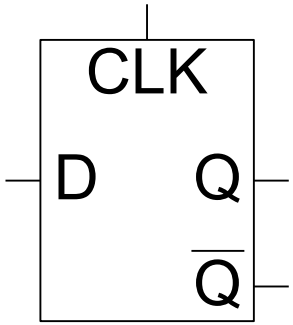
Vergleich D Latch mit D Flip-Flop

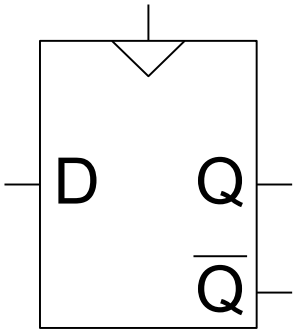


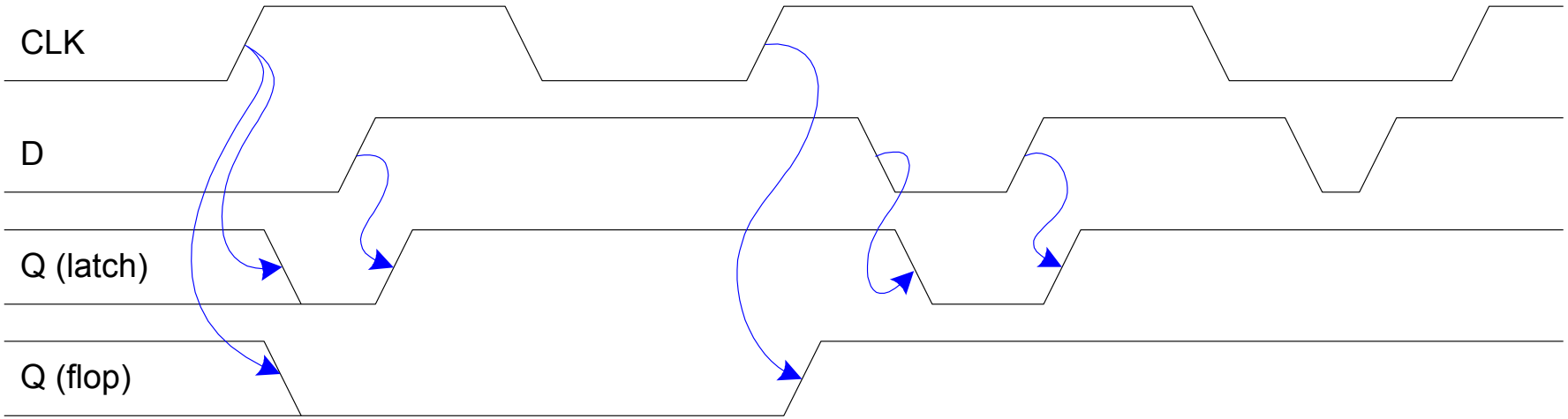




Vergleich D Latch mit D Flip-Flop





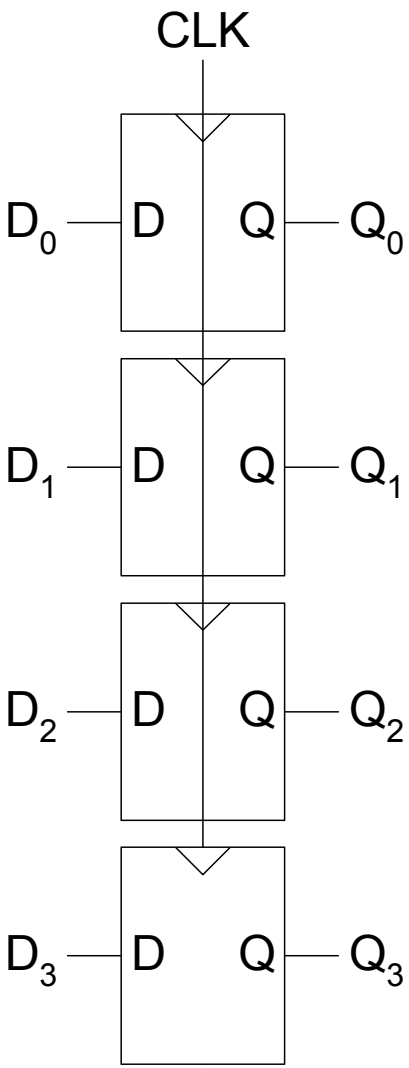


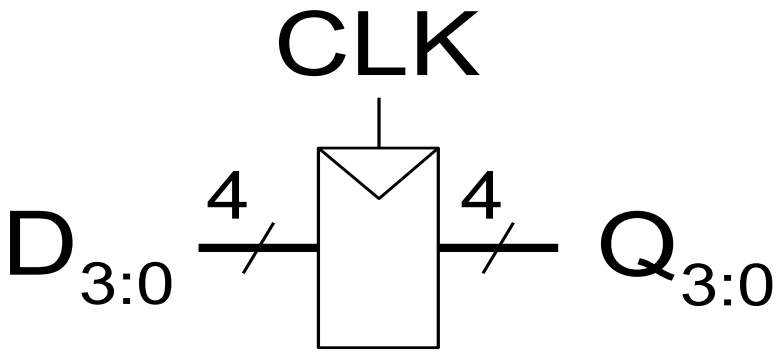
LEHRE WIKI FRAGE

TECHNISCHE UNIVERSITÄT DARMSTADT

- Bitte jetzt auf LEHRE WIKI eine Frage beantworten!

Register

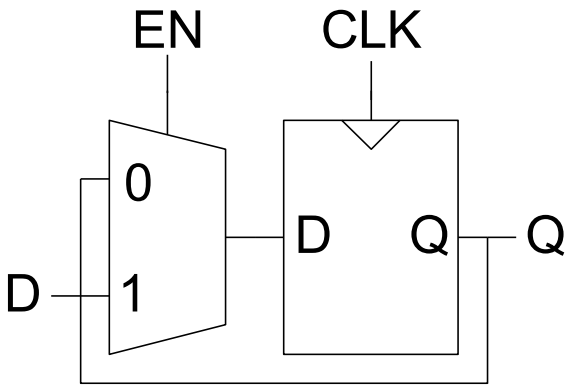




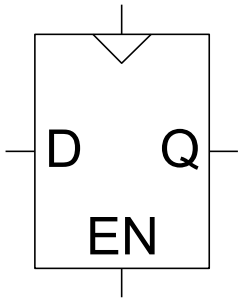
Flip-Flops mit Taktfreigabesignal (*clock enable*)

- Eingänge: *CLK*, *D*, *EN*
  - Freigabeeingang (*EN,* enable) steuert, wann neue Daten (*D*) gespeichert werden
- Funktion
  - *EN* = 1
    - *D* wird weitergegeben an *Q* bei steigender Taktflanke
  - *EN* = 0
    - Q behält alten (gespeicherten) Wert

Interner Aufbau



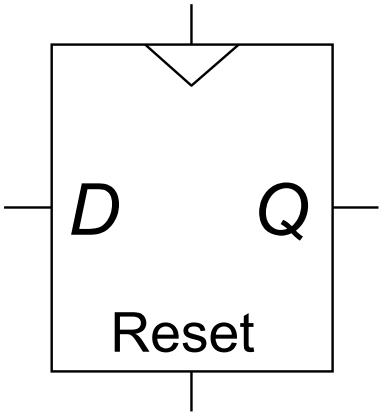
Symbol

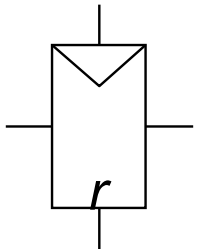


Zurücksetzbare Flip-Flops

- Eingänge: *CLK*, *D*, *Reset*
- Funktion:
  - *Reset* = 1
    - *Q* wird auf 0 gesetzt
  - *Reset* = 0
    - Verhält sich wie normales D Flip-Flop

Symbole





Zurücksetzbare Flip-Flops

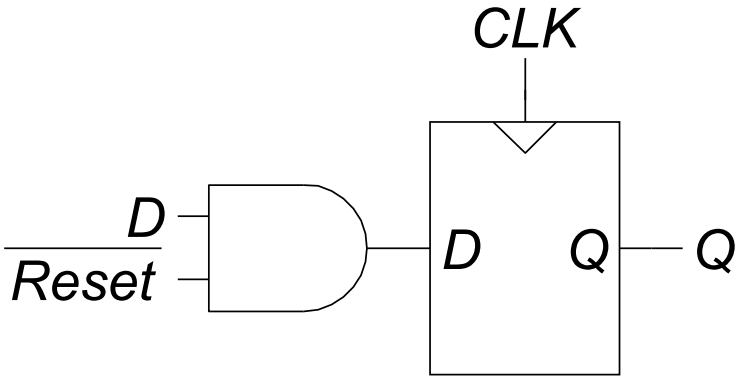
TECHNISCHE UNIVERSITÄT DARMSTADT

- Zwei Arten:
  - Synchron: Rücksetzen geschieht zu steigender Taktflanke
  - Asynchron: Rücksetzen geschieht sofort bei *Reset* = 1
- Interner Aufbau
  - Asynchron: Übung 3.10 im Buch
  - Synchron?

Zurücksetzbare Flip-Flops

- Zwei Arten:
  - Synchron: Rücksetzen geschieht zu steigender Taktflanke
  - Asynchron: Rücksetzen geschieht sofort bei *Reset* = 1
- Interner Aufbau
  - Asynchron: Übung 3.10 im Buch
  - Synchron?

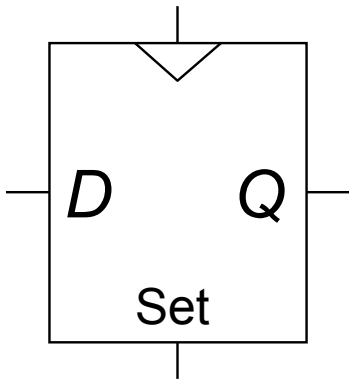
Interner Aufbau

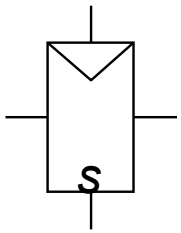


Setzbare Flip-Flops

- Eingänge: *CLK*, *D*, *Set*
- Funktion:
  - *Set* = 1
    - *Q* wird auf 1 gesetzt
  - *Set* = 0
    - Verhält sich wie normales D Flip-Flop

Symbole

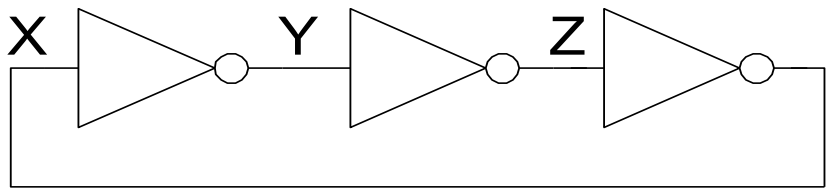


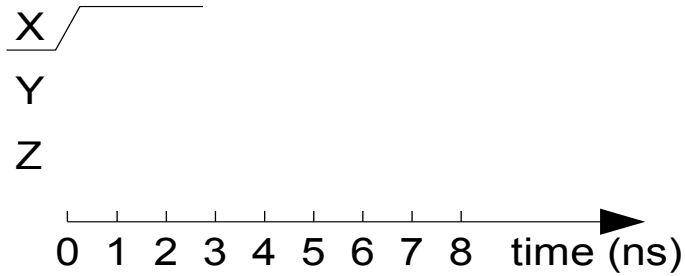


Sequentielle Logik

TECHNISCHE UNIVERSITÄT DARMSTADT

- Sequentielle Schaltungen: Alle nicht-kombinatorischen Schaltungen
- Merkwürdige Schaltung:



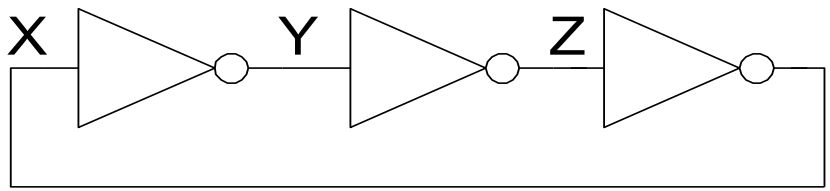


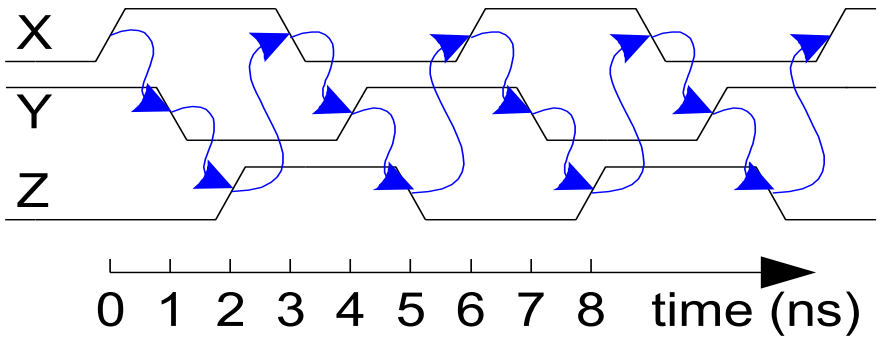
- Keine Eingänge
- 1…3 Ausgänge (Knoten X, Y, Z)

LEHRE WIKI TEST IN ZWEI FOLIEN

Sequentielle Logik

- Sequentielle Schaltungen: Alle nicht-kombinatorischen Schaltungen
- Merkwürdige Schaltung:





- Keine Eingänge
- 1…3 Ausgänge (Knoten *X*, *Y*, *Z*)
- Instabile Schaltung, oszilliert
- Periode hängt von Inverterverzögerung ab
  - Variiert mit Herstellungsprozess, Temperatur, …
- Schaltung hat einen Zyklus: Ausgang rückgekoppelt auf Eingang

Entwurf synchroner sequentieller Logik

TECHNISCHE UNIVERSITÄT DARMSTADT

- Rückkopplungen durch Einfügen von Registern aufbrechen
- Diese Register halten den Zustand der Schaltung
- Register ändern Zustand nur zur Taktflanke
  - Schaltung wird synchronisiert mit der Taktflanke

- Regeln für den Aufbau von synchronen sequentiellen Schaltungen
  - Jedes Schaltungselement ist entweder ein Register oder eine kombinatorische Schaltung
  - Mindestens ein Schaltungselement ist ein Register
  - Alle Register werden durch das gleiche Taktsignal gesteuert
  - Jeder Zyklus enthält mindestens ein Register

- Zwei weit verbreitete synchrone sequentielle Schaltungen
  - Endliche Zustandsautomaten (*Finite State Machines,* FSMs)
  - Pipelines (manchmal Fließbandverarbeitung genannt)

LEHRE WIKI FRAGE

TECHNISCHE UNIVERSITÄT DARMSTADT

- Bitte jetzt auf LEHRE WIKI eine Frage beantworten!