

Digitaltechnik & Rechnersysteme

Latches und Flipflops

Martin Kumm

Hochschule Fulda
University of Applied Sciences



Angewandte Informatik

WiSe 2023/2024

Evaluation



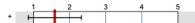
Angewandte Informatik

Globalwerte

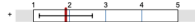
Didaktische Gestaltung

mw=1,5
s=0,7

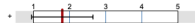
Gestaltung durch die Lehrperson

mw=1,6
s=0,7

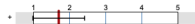
Bedeutsamkeit der Lerninhalte

mw=1,9
s=0,7

Fragen zum eigenen Lernen

mw=1,8
s=0,9

Gesamteinschätzung

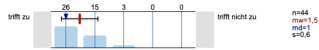
mw=1,7
s=0,7

Evaluation

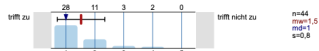


Didaktische Gestaltung

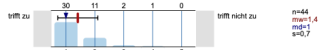
2.1) Die Lehrziele der Lehrveranstaltung sind für mich nachvollziehbar.



2.2) Der inhaltliche Aufbau der Lehrveranstaltung ist für mich verständlich.

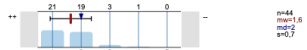


2.3) Die Lehrveranstaltung ist für mich klar strukturiert (roter Faden, Beziehungen/Zusammenhänge zwischen Inhalten, etc.).

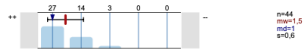


Meine abschließende Gesamteinschätzung...

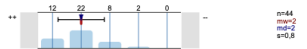
2.17) ... der Lehrveranstaltung insgesamt



2.18) ... des Beitrags der Lehrperson



2.19) ... meines persönlichen Beitrags im Lernprozess



44 Abgaben 😊 (vielen Dank an alle die abgegeben haben!)

Evaluation - Lob



- Vorlesungsaufgaben (10×)
- Struktur (der Folien/Veranstaltung) (9×)
- Wrap-Ups / Rückblick (6×)
- Motivation / Erklärungsfreudig / Enthusiasmus (5×)
- Gruppen und Hausübung (4×)
- Freiwillige Abgabe der Hausübung (Bonuspunkte, Rückmeldung) (3×)
- Erklärungen (2×)
- Vortagsweise (2×)
- Vorstellung der Übungen für Bonuspunkte
- Discord-Server
- Einbindung von Studierenden
- guter Umgang mit Fehlern und Nähe zu Studierenden
- Beispiele an der Tafel
- Tutor in den Übungen wirkt sehr kompetent und freundlich

Evaluation - Kritik an Organisation



- Montags-Übung vor der Vorlesung, Themen manchmal noch nicht behandelt (7×)
- Hausübungen schwieriger als Übung, Übung schwieriger als Vorlesung (6×)
- Folien für die kommende Veranstaltung könnten früher hochgeladen werden (2×)

Evaluation - Kritik zur Vorlesung

- Manchmal wichtige / kompliziertere Inhalte zu schnell (4×)
- Es können nicht immer alle Folien in der Vorlesung besprochen werden
- manchmal hektisch
- kurze Bearbeitungszeit für Vorlesungsaufgaben
- Tempo: Für Anfänger aus meiner Sicht viel zu anspruchsvoll.
- Folien könnten ausführlicher sein, oder weiteres Hilfsmaterial
- Folien sind an manchen Stellen zu kompliziert
- mehr Beispiele in den Folien
- Smileys auf den Folien 😞

Evaluation - Kritik zur Übung



Angewandte Informatik

- Musterlösungen der Gruppenübungen nicht hochgeladen (4×)
- Übungsgruppen zu voll (Zusatzpunkte problematisch) (2×)
- Lösungen zu den Gruppenübungen aber nicht Hausübungen
- Wechseln der Übungsgruppe nicht gestattet
- Auswahlssystem für Vorstellung in Gruppenübung
- Nachträglich Lösung zu den Aufgaben aus der SU hochladen
- Hohe Arbeitslast (3 Tage an einer Übung)
- keine Hilfestellungen oder weitere Hilfsangebote
- Klarere Aufgabenstellung vor allem in den Hausübungen
- Mehr Kulanz in Hausübung
- Menge der Aufgaben ist zu viel (2×)
- praxisnähere Aufgaben
- (Übungen) mehr Klausurbezogen
- hohes Niveau, was man etwas herabsetzen könnte
- nur eine abgabemöglichkeit für übungsaufgaben

Evaluation - Konsequenzen

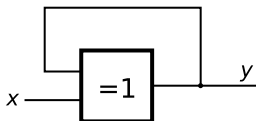
- Musterlösungen Gruppenübungen werden früher freigegeben
- Folien werden früher hochgeladen (spätestens am Abend davor), ggf. kurzfristige Änderungen
- Übungsaufgaben werden in kleineren Teilen vorgestellt, so dass jeder eine Chance auf Bonuspunkte bekommt
- Große Gruppen: Übung am Donnerstag (15:30-17:00) ist stark unterbesetzt (10 Teilnehmer), einfacher Wechsel möglich
- Bearbeitungszeit für Vorlesungsaufgaben wird verlängert
- Weiteres Hilfsmaterial: Siehe Buchempfehlungen aus 1. Vorlesung!
- Klarere Aufgabenstellung in Hausübungen: Bitte konkrete Rückmeldung!
- Manches in Ihrer Eigenverantwortung:
 - Feedback in Vorlesung wenn nicht verstanden, Zeit für Vorlesungsaufgabe nicht ausreicht, etc.!
 - Feedback in Übung wenn Thema noch nicht verstanden!

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

Beispiel

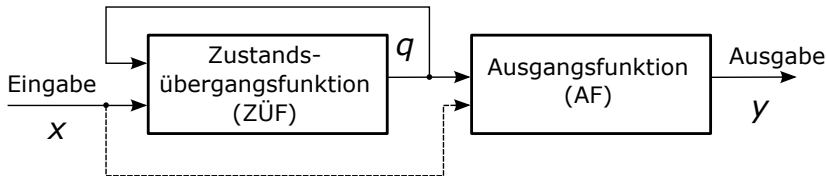


y^t	x	$y^{t+\tau}$
0	0	0
0	1	1
1	0	1
1	1	0

Inhalte

- 1 Wrap-Up
- 2 Schaltwerksanalyse
- 3 Übergangsgraph
- 4 RS-Latch
- 5 Synchrone Speicher
 - Taktpiegelgesteuerte Speicher
 - Taktflankengesteuerte Speicher
- 6 Weitere FF-Typen
 - JK-Flipflop
 - T-Flipflop

Darstellung eines allgemeinen Schaltwerks



Die ZÜF berechnet den Folge-Zustandsvektor $q^{t+\tau}$ in Abhängigkeit des aktuellen Zustandsvektors q^t und der aktuellen Eingabe x .

Die AF berechnet die Ausgabe y in Abhängigkeit des aktuellen Zustandsvektors q^t und der aktuellen Eingabe x .

Moore und Mealy Automaten



Üblicherweise werden zwei Typen von Automaten unterscheiden:

MEALY Automat: ist der allgemeine Fall

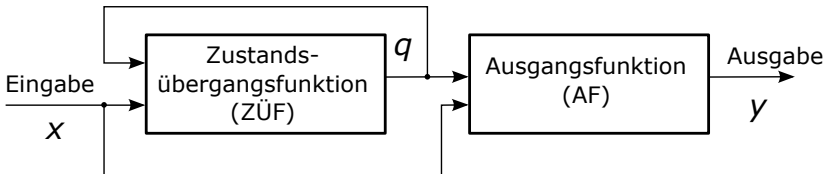
MOORE Automat: Einschränkung, dass die Ausgänge nur von den Zuständen abhängen:

$$y = F(q)$$

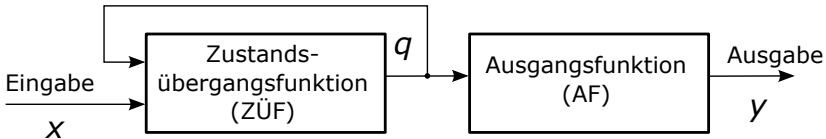
Moore und Mealy Automaten



MEALY Automat:



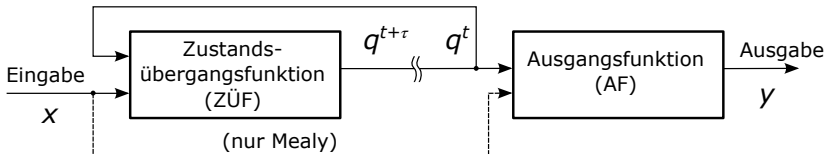
MOORE Automat:



Schaltwerksanalyse

Vorgegeben ist ein Schaltbild.

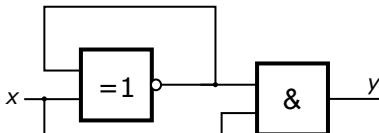
Durch die **Analyse** soll auf die Funktion geschlossen werden.
Dazu werden Rückkopplungen so aufgetrennt, dass die Struktur eines Schaltnetzes entsteht (rückkopplungsfrei).



Vorlesungsaufgabe



Analysieren Sie das folgende Schaltwerk:



Erstellen Sie die Zustandsübergangs- und Ausgangstabelle

Lösung Vorlesungsaufgabe



Stabile Zustände



Transitionen (Zustandsübergänge), bei denen (alle) q^t und $q^{t+\tau}$ übereinstimmen führen zu **stabilen Zuständen**.

D.h. für die Eingabekombination dieser Transition bleibt der Automat in seinem Zustand.

Vorlesungsaufgabe



Markieren Sie die Transitionen, welche zu stabilen Zuständen führen:

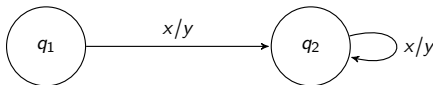
q^t	x	$q^{t+\tau}$	y
0	0	1	0
0	1	0	0
1	0	0	0
1	1	1	1

Beschreibung als Übergangsgraph

Übergangsgraph/Zustandsübergangsgraph: Den Knoten (Ecken) des Graphen werden die Zustände zugeordnet. Zustandsübergänge (**Transitionen**) entsprechen gerichteten Kanten (Pfeile).

Die **Knoten** werden durch Kreise dargestellt und erhalten als Beschriftung die Zustandsbezeichnung.

Die **Kanten** werden mit dem Eingangsvektor beschriftet, der den entsprechenden Übergang auslöst, sowie dem Ausgabevektor:



Beispiel: XOR mit Rückkopplung

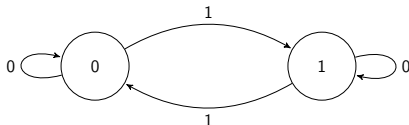
y^t	x	$y^{t+\tau}$
0	0	0
0	1	1
1	0	1
1	1	0

← Ausgang bleibt stabil: $y^{t+\tau} = y^t = 0$

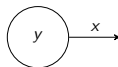
← Ausgang wechselt von 0 auf 1

← Ausgang bleibt stabil: $y^{t+\tau} = y^t = 1$

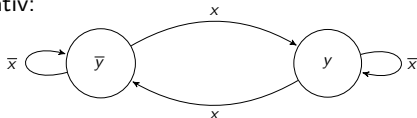
← Ausgang wechselt von 1 auf 0



Notation:



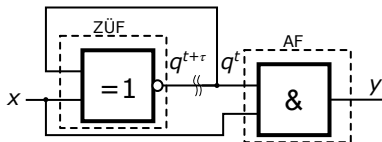
Alternativ:



Vorlesungsaufgabe

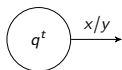


Erstellen Sie für das Schaltwerk den Zustandsübergangsgraph.



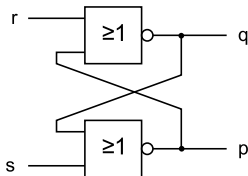
q^t	x	$q^{t+\tau}$	y
0	0	1	0
0	1	0	0
1	0	0	0
1	1	1	1

Notation:



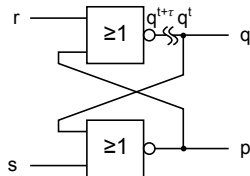
Speichern einer binären Variablen

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

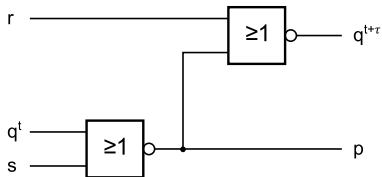


⇒ Zur Analyse werden so lange Rückkopplungen aufgetrennt bis ein Schaltnetz entsteht.

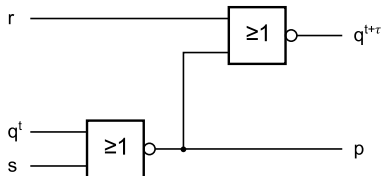
Analyse der Funktionsweise



Daraus ergibt sich ein gewöhnliches Schaltnetz:



Analyse der Funktionsweise



Funktion der Schaltung:

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

Zustandsübergangstabelle:

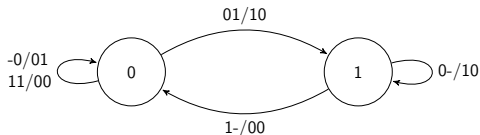
q^t	r	s	$q^{t+\tau}$	p
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	0	0
1	0	0	1	0
1	0	1	1	0
1	1	0	0	0
1	1	1	0	0

Zustandsdiagramm des RS-Latch

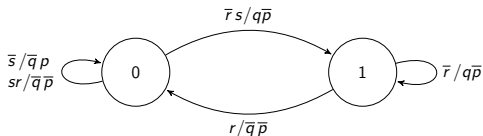


Angewandte Informatik

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
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 $\gg\ll$ 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):

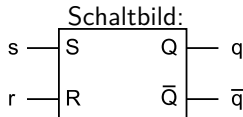
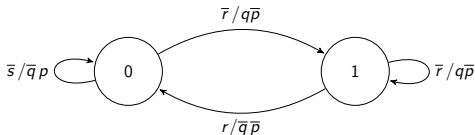
r	s	$q^{t+\tau}$	
0	0	q^t	speichern
0	1	1	setzen
1	0	0	rücksetzen
1	1	–	nicht zulässig

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

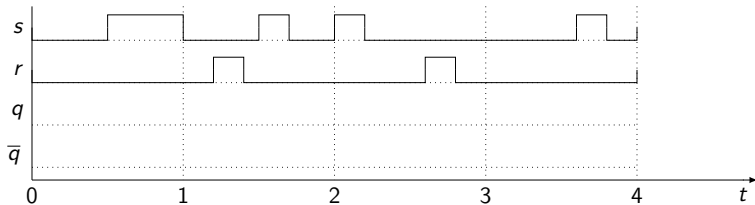
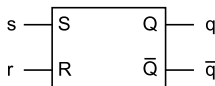
r = reset, s = set

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.



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).

Taktpegel- und Taktflankensteuerung



Taktpegelsteuerung (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→1 Wechsel oder 1→0 Wechsel) möglich.

⇒ Speicherelemente werden als **Flipflop** bezeichnet

Pegel- und Flankensteuerung

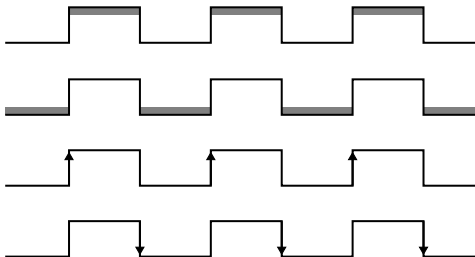


Angewandte Informatik

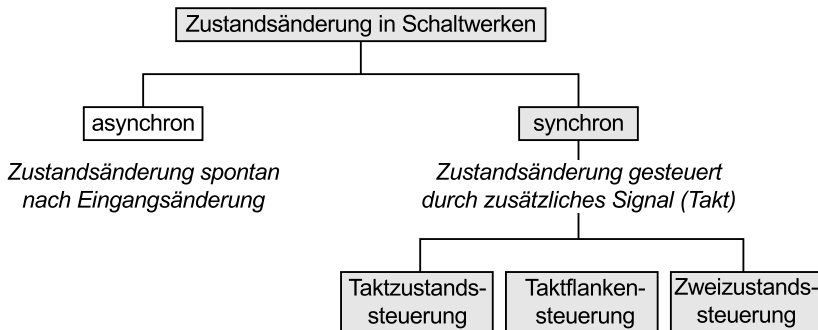
Durch Invertieren des Taktsignals kann auch eine Reaktion auf den Low-Pegel des Taktes bzw. $1 \rightarrow 0$ Taktflanke erreicht werden.

$0 \rightarrow 1$ 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.



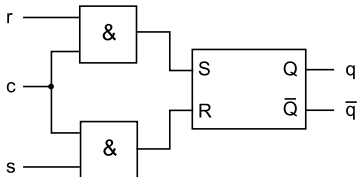
Taxonomie von Schaltwerken



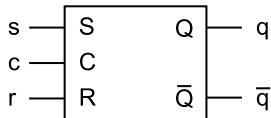
RS-Latch mit Taktpiegelsteuerung I

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
- 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 Taktpiegelsteuerung

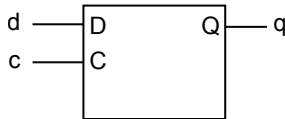
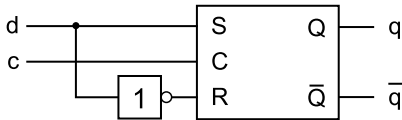
Set und reset lässt sich einem Datensignal d ableiten:

$$s = d$$

$$r = \overline{d}$$

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

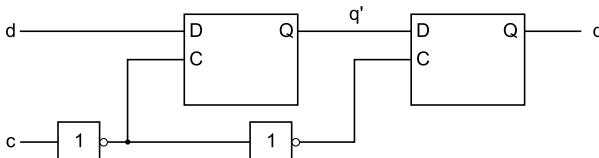
Schaltsymbol:



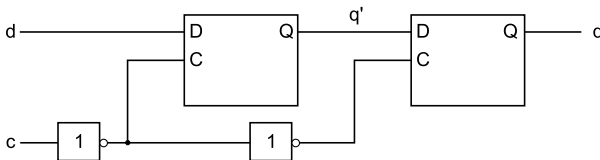
Taktflankengesteuertes D-Flipflop

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 taktpegelgesteuerter D-Latches erhält man ein flankengesteuertes D-Flipflop:

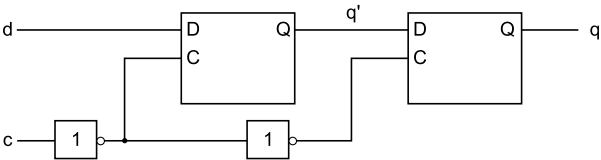


Taktflankengesteuertes D-Flipflop

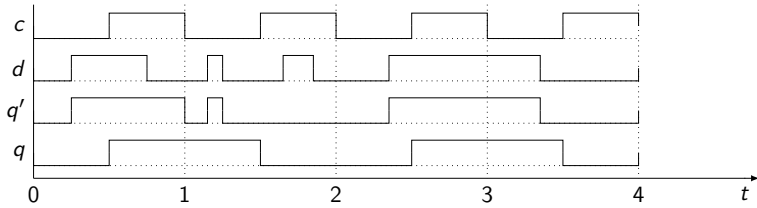


Das erste Flipflop übernimmt ein Eingangssignal während der aktiven Taktphase, das zweite Flipflop hält zur gleichen Zeit seinen Inhalt unverändert. Beim 0→1-Übergang des Taktsignals wird das erste Flipflop gesperrt, während das zweite dessen Inhalt übernimmt.

Taktflankengesteuertes D-Flipflop

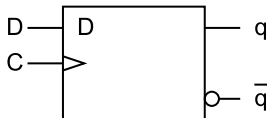


Beispiel-Timing:



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 definierte 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

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

Toggle-Flipflop

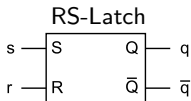
Ein JK-Flip-Flop mit $j = k = t$ führt zum **Toggle-Flipflop**: Für $t = 1$ ändert sich der Zustand (toggle), für $t = 0$ wird gespeichert.

Übergangstabelle:

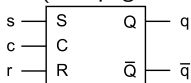
t	q^{t+1}	
0	q^t	speichern
1	$\overline{q^t}$	togglen

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

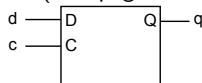
Übersicht Flipflops



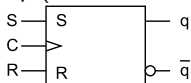
RS-Latch (Taktpegelsteuerung)



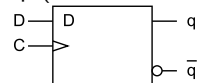
D-Latch (Taktpegelsteuerung)



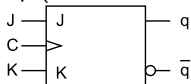
RS-Flipflop (Taktflankensteuerung)



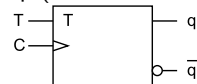
D-Flipflop (Taktflankensteuerung)



JK-Flipflop (Taktflankensteuerung)



T-Flipflop (Taktflankensteuerung)



**FROHE
WEIHNACHTEN**

