

5. **SPEICHERELEMENTE UND REGISTERSCHALTUNGEN**

5.1 **Vorbemerkungen**

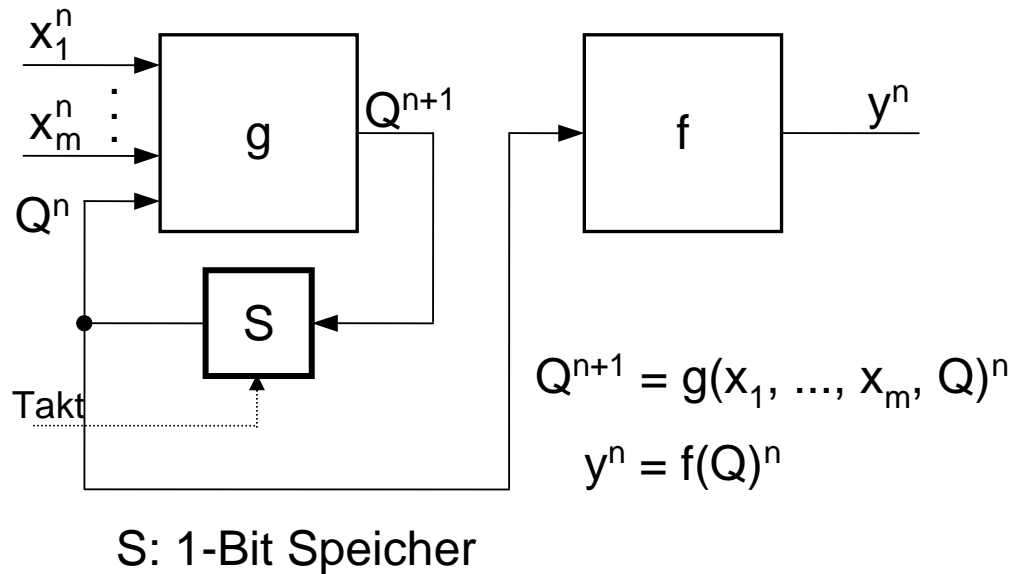
Bei den Schaltnetzen zeigten schon die Multiplizierer und andere Rechenschaltungen, dass Schaltnetzen (= „kombinatorische Logik“) Grenzen in komplexeren Anwendungen gesetzt sind und diese besser iterativ, also in Form eines mehrschrittigen Ablaufs, realisiert werden. Außerdem ist es schwer, sequentielle Aufgaben wie z.B. eine Aufzugssteuerung nur mit einem reinen Schaltnetz zu realisieren. Auf der einen Seite wird der Realisierungsaufwand reduziert, wenn Hardware mehrfach mit unterschiedlichen Daten oder Teilen von Operanden verwendet werden kann (z.B. wiederholte stellengewichtete Addition statt eines Parallel-Multiplizierers). Auf der anderen Seite ergibt sich ein flexibleres, mächtigeres Verhalten einer Digitalschaltung.

Dafür müssen in beiden Fällen Zwischenergebnisse (wie die Partialprodukte) bzw. innere Abarbeitungszustände gespeichert werden. Das ist mit reinen Schaltnetzen nicht zu machen, weil sie abgesehen von Laufzeitverzögerungen „zeitlos“ arbeiten. Es ist also ein anderer Typus von Digitalschaltungen gefragt, der ein „zeitabhängiges“ Verhalten entwickeln kann, das von der Vorgeschichte abhängt, die zusammenfassend in Form eines inneren Zustands(vektors) repräsentiert wird. Dieser neue Schaltungstypus wird Schaltwerk („sequentielle Logik“) genannt.

Für das Speichern von Zuständen sind in jedem Schaltwerk Flipflops als Speicherelemente enthalten. Nach der Vorstellung der verschiedenen Typen von Flipflops werden daraus typische Anwendungen abgeleitet. Der systematische Entwurf von Schaltwerken folgt dann im nächsten Kapitel.

Allgemeines Speicherelement

Speicherelemente basieren auf einer zustandserhaltenden Rückkopplung, die von äußeren Signalen beeinflusst werden kann.



Die **Übergangsfunktion** g beschreibt den nächsten Zustand der Zustandsvariablen Q^{n+1} als Funktion des aktuellen Zustands Q^n und der aktuellen Eingangsvariablen zum Zeitpunkt n :

$$Q^{n+1} = g(x_1, \dots, x_m, Q)^n$$

Die **Ausgangsfunktion** f gibt an, wie der Ausgang y eines Speicherelementes von seinem (aktuellen) Zustand abhängt.

Bei einfachen Flipflops ist das meist die Identitätsfunktion.

Die **charakteristische Funktion** beschreibt das Verhalten des Flipflops in knapper Form:

$$Q^{n+1} = g(x_1, \dots, x_m, Q)^n \quad \text{in DMF.}$$

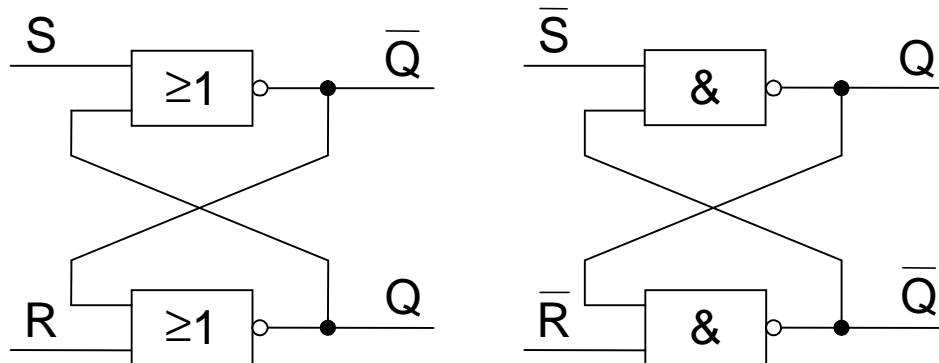
Sie kann algebraisch oder aus der Wahrheitstafel z. B. mittels KV-Diagramm abgeleitet werden.

5.2 Speicherelemente aus Flipflops (Bistabile Kippstufen)

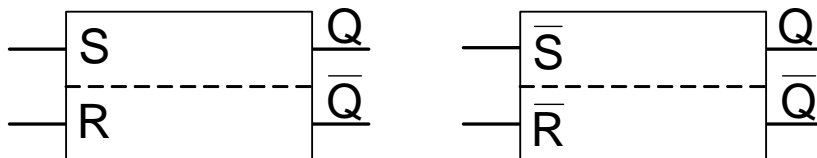
5.2.1 RS-Basis-Flipflop

Ein RS-Flipflop kann einen von zwei möglichen stabilen Zuständen annehmen (bistabil), die von den beiden Eingängen zum **R**ücksetzen und **S**etzen (Reset, Set) abhängen; deshalb die Bezeichnung „RS-Flipflop“ (oder SR-Flipflop).

Realisierung mit NOR- bzw. NAND-Gattern



Schaltsymbole



Zustandstabelle (Zustandsübergangstabelle)

Wahrheitstafel; die den Folgezustand für alle Kombinationen aus Zuständen und Eingangsvariablen angibt

R^n	S^n	Q^{n+1}	\bar{Q}^{n+1}
0	0	Q^n	\bar{Q}^n
0	1	1	0
1	0	0	1
1	1	unzulässig	

Verhalten eines RS-Flipflops

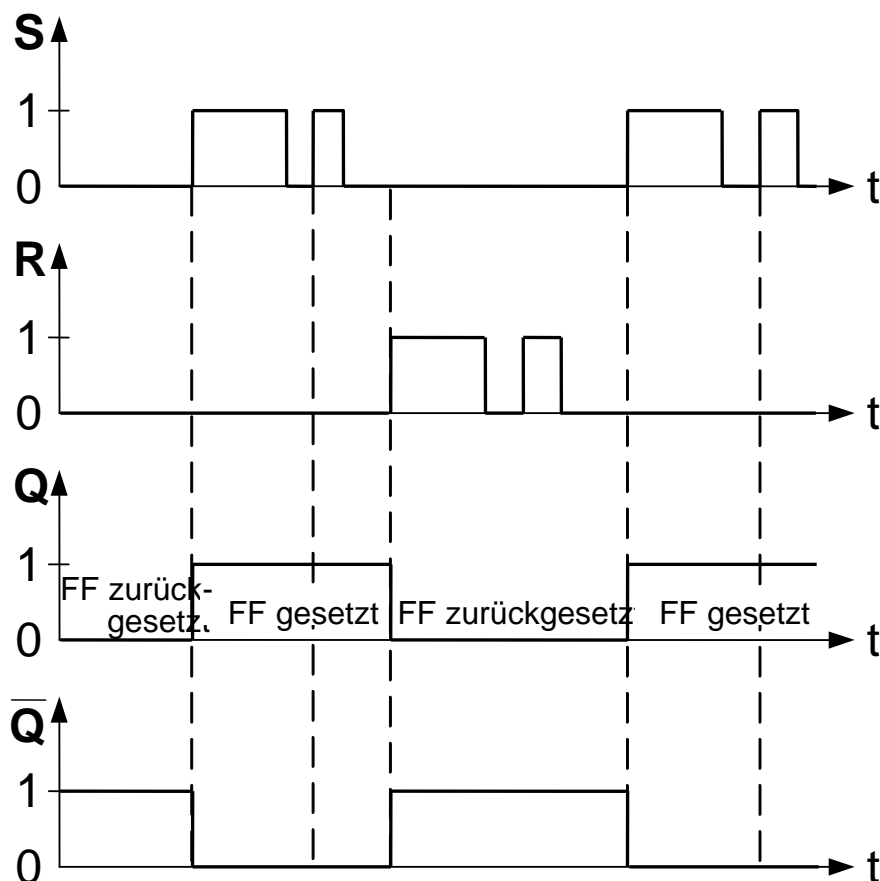
$$Q^{n+1} = \overline{\overline{R^n} + \overline{Q^n}} = \overline{\overline{R^n} + (\overline{S^n} + Q^n)} = \overline{R^n}(\overline{S^n} + Q^n)$$

Charakteristische Funktion

$$Q^{n+1} = \overline{R^n}Q^n + S^n \quad \text{mit } RS = 0 \text{ (Nebenbedingung)}$$

Impulsdiagramme (Signal-Zeit-Diagramm) stellen das zeitliche Verhalten schematisch (bei kleineren Funktionalitäten meist vollständig, d.h. für alle möglichen Signalkombinationen) dar.

Impulsdiagramm des RS-Flipflops



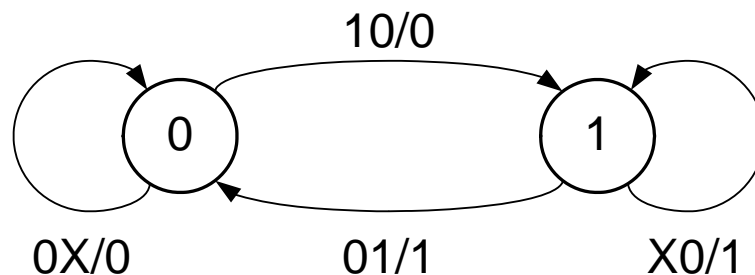
Die Zustandsübergänge können bei einem RS-Flipflop zu jedem beliebigen Zeitpunkt (d.h. „asynchron“) erfolgen.

Zustandsgraphen (Zustandsübergangsdiagramme) beschreiben das funktionale Verhalten (analog zu endlichen Automaten) vollständig, aber ohne Zeitbezug.

Zustände werden als Knoten und Zustandsübergänge als Kanten dargestellt.

Die Bedingungen für die Zustandsübergänge werden an den Kanten angegeben, die Ausgaben je nach zugrunde liegendem Modell (s. u.) an den Knoten oder den Kanten.

Zustandsgraph (Zustandsübergangsdiagramm) des RS-Flipflops



Zustände: $Q = 0$ und $Q = 1$

Zustandsübergänge: SR/Q zum Zeitpunkt n
(Notation) (X = beliebig, don't care)

Erweitertes RS-Flipflop: JK-Flipflop (asynchron)

Eine einfache Lösung zur Vermeidung des unerlaubten Zustands beim RS-Flipflop mit $R = S = 1$ ist die Verriegelung der Eingänge mittels UND-Gattern in Abhängigkeit vom Ausgangszustand.

Auf diese Weise wird ein sog. JK-Flipflop gebildet, benannt nach den Eingängen, die nun J (Jump) und K (Kill) heißen. (siehe unten)

5.2.2 Taktsteuerung

Damit ein asynchrones Speicherelement nicht mit einem Zustandswechsel auf kurze transiente Übergangs- oder Störimpulse wie Glitches reagiert, müssen die Eingänge verriegelt bzw. nur für wohl definierte Zeitintervalle oder Zeitpunkte freigeschaltet werden.

Flipflops erhalten dazu einen zusätzlichen „Enable“- bzw. Takteingang, der bestimmt, wann das Umschalten des Flipflops freigeschaltet ist („Wirkintervall“). Es gibt darüber hinaus verschiedene Realisierungen, wann das Flipflop in den gespeicherten neuen Zustand wechselt („Kippintervall“).

Eine Zustandsänderung ist also nur zu vordefinierten Zeiten möglich ("synchrone Flipflops").

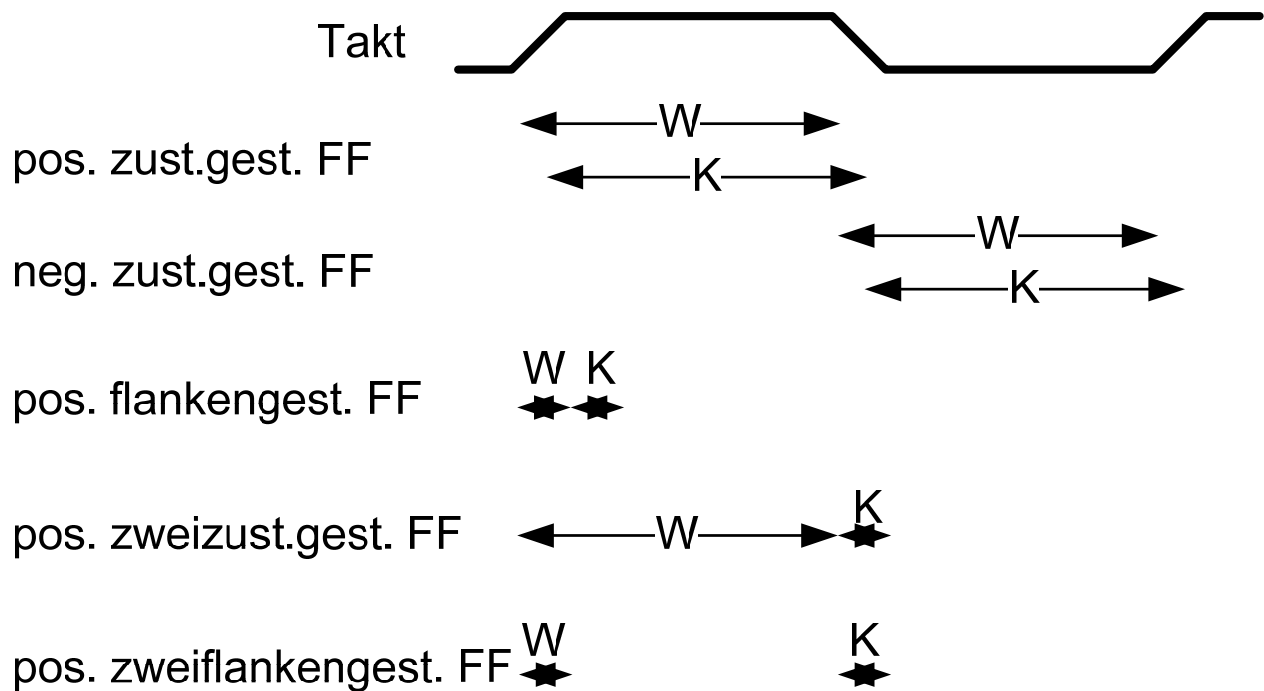
- Zustandssteuerung (Pulstriggerung)
Am Eingang anliegende Information wird während der gesamten Dauer des "aktiven" Taktimpulszustandes übernommen (und wirkt für diese Zeit meist auch am Ausgang).
- Einflankensteuerung
Eingangsinformation wird mit einer aktiven (der positiven/steigenden oder negativen/fallenden) Taktflanke (Vorder- bzw. Rückflanke) übernommen. Anschließend Änderungen während der Impulsdauer des Taktes wirken nicht mehr auf den Ausgang.
- Zweizustandssteuerung (Master-Slave-Prinzip)
Die Eingangsinformation wird während der gesamten aktiven Taktphase in einem Zwischenspeicher übernommen (1. Zustand), aber erst mit der nachfolgenden Taktphase am Ausgang wirksam (2. Zustand).

Ob mehrfache Änderungen während der aktiven Taktphase in den Zwischenspeicher übernommen werden, hängt vom konkreten Flipflop-Typ ab.

- Zweiflankensteuerung (Master-Slave-Prinzip)

Die Eingangsinformation wird während der ersten Flanke des Taktes in den (internen) Zwischenspeicher übernommen, aber erst mit der zweiten Taktflanke am Ausgang wirksam. Eingangsänderungen nach der ersten Flanke, insbesondere während der Impulsdauer des Taktes, sind unwirksam.

Wirk- und Kippintervalle von Flipflops



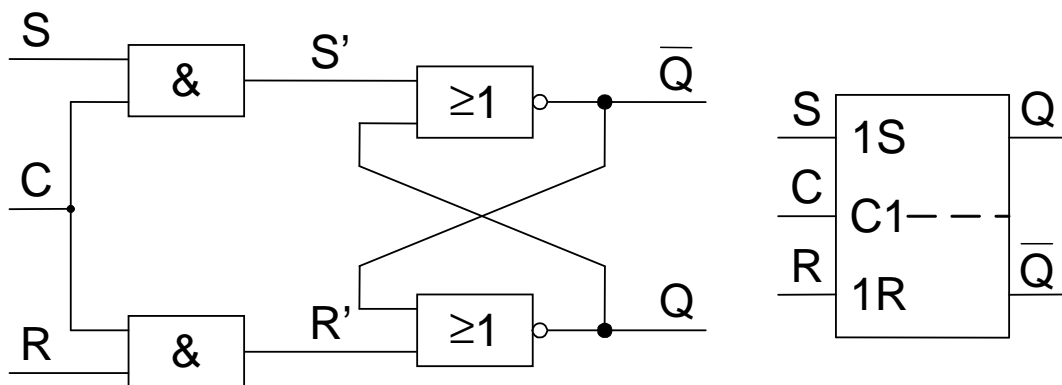
Das Wirkintervall gibt an, wann der Flipflop-Eingang dessen Zustand (abhängig von der Taktsteuerung) beeinflussen kann.

Das Kippintervall zeigt an, wann der aktuelle Zustand am Ausgang eingenommen (und gehalten) wird.

Vom konkreten Einsatz hängt ab, ob die positive oder negative Taktflanke oder –zustand verwendet wird.

(Takt)Zustandsgesteuertes RS-Flipflop

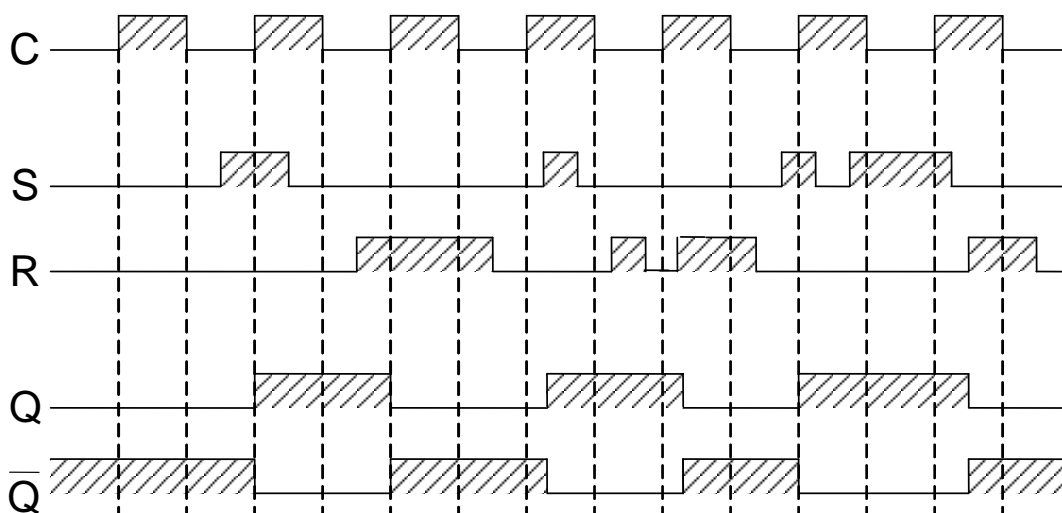
(hier: positive Taktzustandssteuerung)



Eingangsbeschaltung des asynchronen RS-Flipflops mit zwei UND-Gattern zur Verriegelung bzw. Freigabe der Eingänge durch das Taktsignal C. Nur für den Taktzustand $C = 1$ kann das Flipflop hier Eingangsinformation übernehmen. Bei $C = 0$ bleibt der Zustand des Flipflops unverändert.

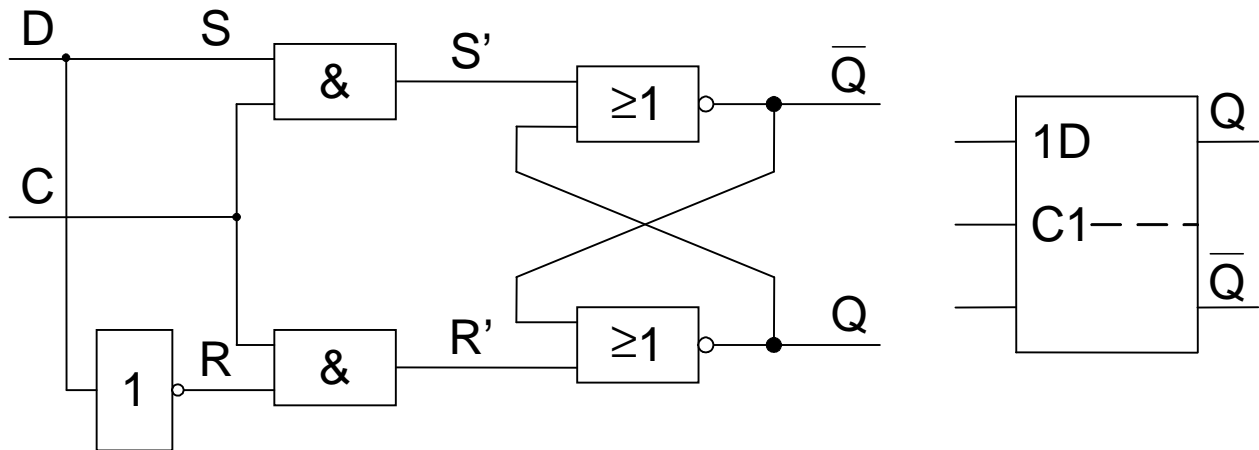
C^n	R^n	S^n	Q^{n+1}	\overline{Q}^{n+1}
1	0	0	Q^n	\overline{Q}^n
1	0	1	1	0
1	1	0	0	1
1	1	1	unzulässig*	
0	X	X	Q^n	\overline{Q}^n

Signal-Zeit-Diagramm für taktzustandsgesteuertes RS-FF



Taktzustandsgesteuertes D-Flipflop (Data-Flipflop)

Soll nur ein Eingangssignal (z.B. ein Datenbit) verarbeitet, z.B. gespeichert werden, werden D-Flipflops verwendet. Sie ergeben sich z.B. aus einem RS-FF durch einen zusätzlichen Inverter am Eingang.



Während der aktiven Taktphase (hier $C = 1$) ist das D-Flipflop transparent, d. h. der Zustand von D ist (abgesehen von Laufzeiten) unmittelbar am Ausgang sichtbar.

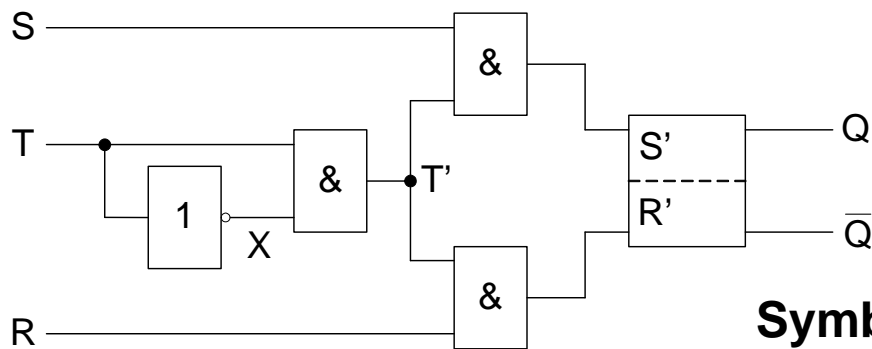
Für $C = 0$ wird der letzte Zustand von D gehalten (gespeichert), bis C wieder auf 1 geht.

Solche takt-phasenweise transparenten Flipflops werden zur Unterscheidung auch Latch genannt.

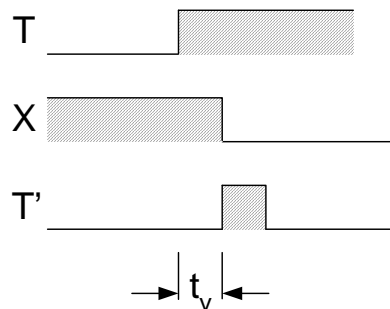
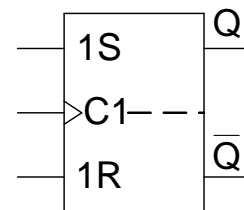
Charakteristische Funktion:

$$Q^{n+1} = D^n$$

Einflankengesteuertes RS-Flipflop (hier: positive Taktflankensteuerung)

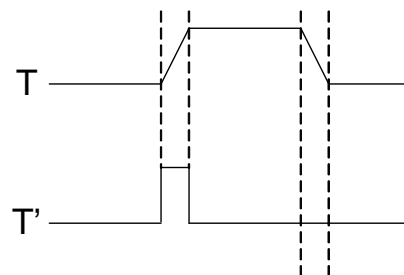
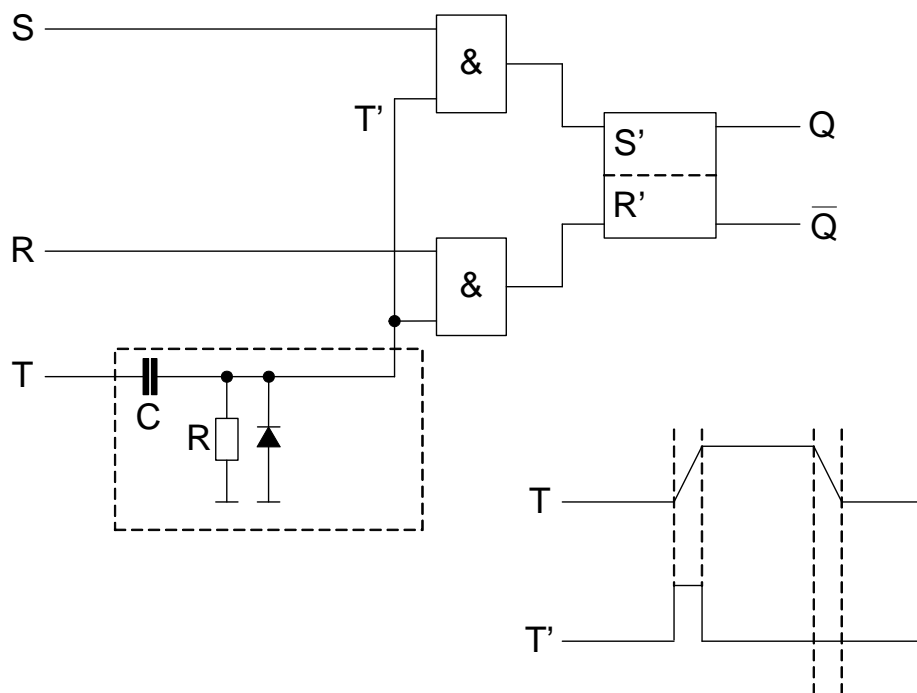


Symbol:



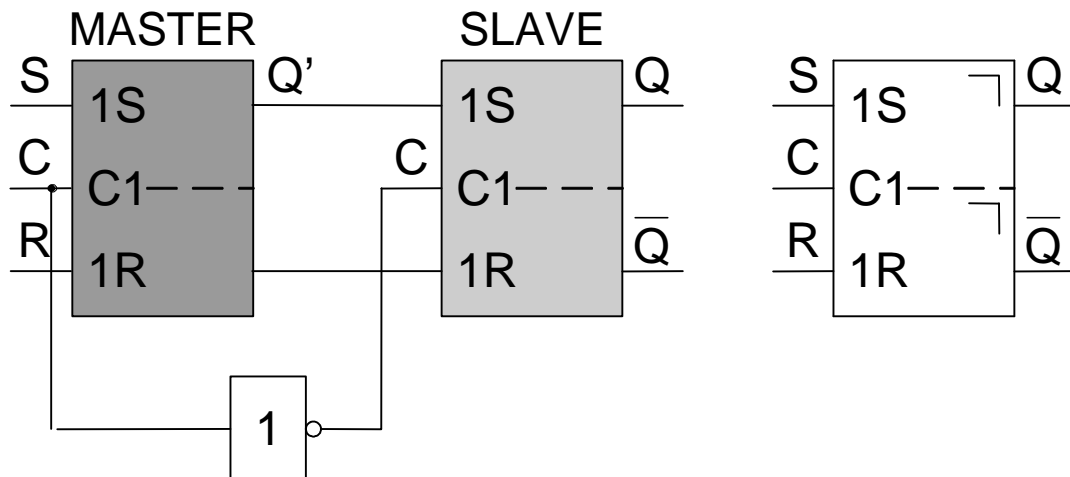
t_v = Gatterlaufzeit

Die Flankensteuerung geschieht durch die Erzeugung eines kurzen Impulses (oben bspw. durch die Verzögerungszeit eines Inverters als „Laufzeitdifferenzierer“) zur kurzzeitigen Freigabe der Informationsübernahme.



Erzeugung eines kurzen Impulses zur Informationsübernahme durch ein Differenzierglied (RC-Glied).

5.2.3 RS-Master-Slave-Flipflop



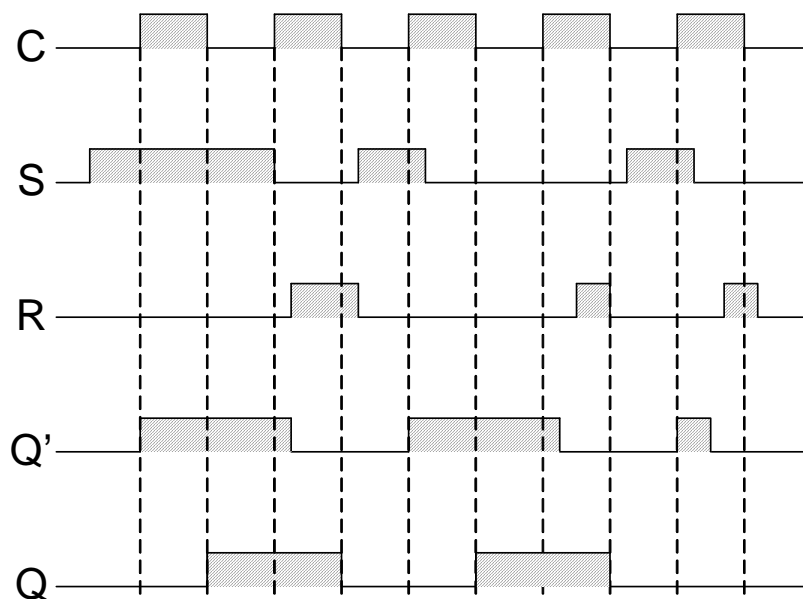
Serielle Schaltung aus zwei RS-Flipflops mit Zweizustandssteuerung durch Ansteuerung mit inversem Takt

$C = 1$: Master wird entsprechend S bzw. R (beliebig häufig) gesetzt oder zurückgesetzt;
Slave hält alten Wert

$C = 0$: Slave übernimmt nur **einmalig** beim Wechsel des Taktsignals den gerade aktuellen Wert vom Master; Master selbst ist nun deaktiviert.

Achtung: verzögerte (retardierte) Reaktion am Ausgang!!!

Impulsdiagramm RS-MS-FF:



5.2.4 JK-(Master-Slave-)Flipflop

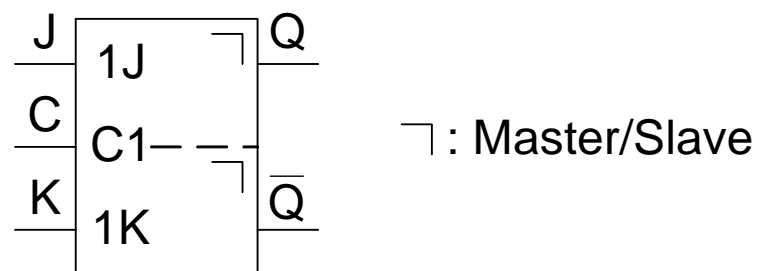
Bei JK-Flipflops (**J**ump, **K**ill) vermeidet eine vorgeschaltete wechselseitige Verriegelung (UND-Gatter) die unerlaubte Signalkombination des RS-Flipflops, nämlich gleichzeitiges $R = 1$ und $S = 1$.

Wahrheitstafel

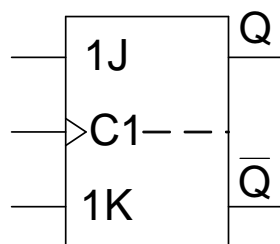
J^n	K^n	Q^{n+1}	\overline{Q}^{n+1}
0	0	Q^n	\overline{Q}^n
0	1	0	1
1	0	1	0
1	1	\overline{Q}^n	Q^n

Charakteristische Funktion $Q^{n+1} = J\overline{Q}^n + \overline{K}Q^n$

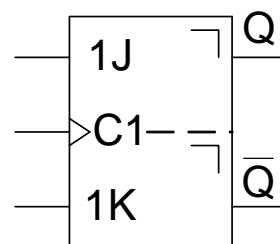
Schaltsymbol



JK-Flipflop mit Flankensteuerung:

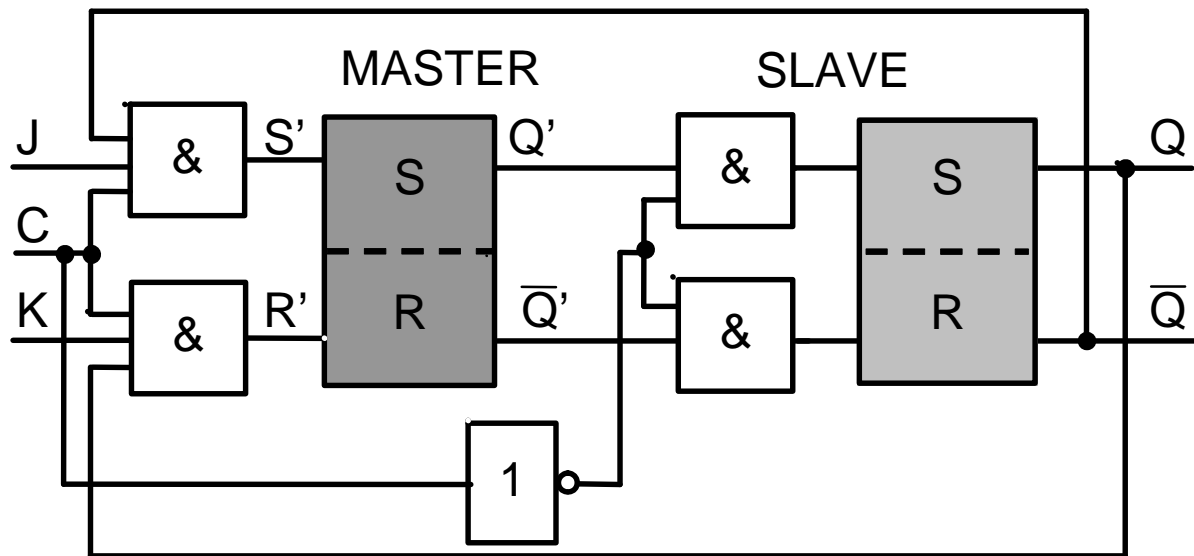


Einflankensteuerung



Zweiflankensteuerung

Beispiel: JK-Master-Slave-Flipflop mit Zweizustandssteuerung:



C = 1: Immer nur ein Eingang ist wirksam, der andere gesperrt:

Für $Q = 0$ ist J freigeschaltet und K gesperrt.

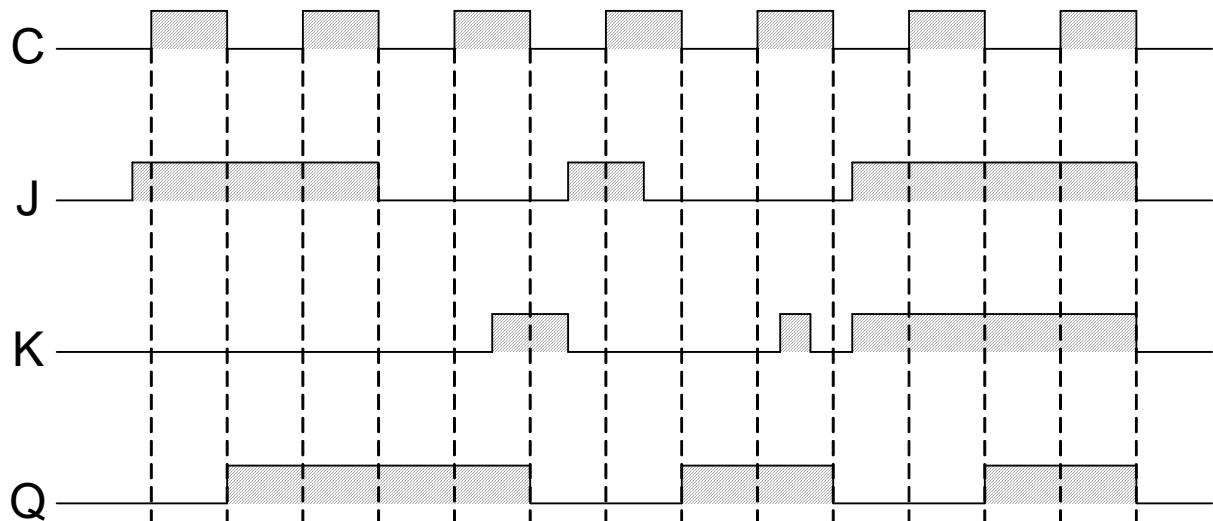
Für $Q = 1$ ist K freigeschaltet und J gesperrt.

C = 0: Beide Eingänge gesperrt; Zustand wird vom Master in Slave übernommen.

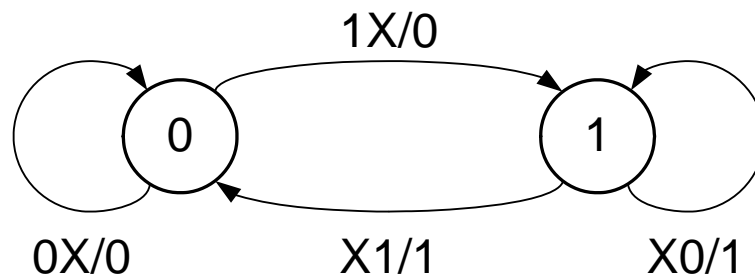
Achtung: Der Master kann während der Halbperiode $C = 1$ seinen Zustand im Gegensatz zu einem RS-Flop aufgrund der Verriegelung nur **einmal** ändern!!!

Regel: Die JK-Signale sollten sich nur während der Halbperiode $C = 0$ ändern und während $C = 1$ stabil bleiben!

Impulsdiagramm JK-MS-FF (Zweizustandssteuerung)



Zustandsübergangsdiagramm



Zustände: $Q = 0$ und $Q = 1$

Zustandsübergänge: JK/Q (X = beliebig)

VHDL-Codebeispiel für ein JK-Flipflop

```
LIBRARY ieee;                                -----
USE ieee.std_logic_1164.ALL;                --    J-K FF    --
                                           -----

ENTITY JK_FFLOP IS
    PORT(clk, j, k      : IN      std_logic;
          q              : BUFFER std_logic);
END JK_FFLOP;
```

```
ARCHITECTURE arc OF JK_FFLOP IS
-- positive edge-triggered JK-flipflop
SIGNAL jk : std_logic_vector (1 DOWNT0 0);
BEGIN
    jk<=j&k;
    PROCESS (clk, j, k)
    BEGIN
        IF (clk'EVENT AND clk = '1') THEN
            -- positive edge trigger
            CASE jk IS
                WHEN "00" => q <= q;          --Hold
                WHEN "01" => q <= '0';        --Reset
                WHEN "10" => q <= '1';        --Set
                WHEN "11" => q <= NOT q;      --Toggle
                WHEN OTHERS => q <= q;
            END CASE;
        END IF;
    END PROCESS;
END arc;
```

5.2.5 D-Flipflop (D = Data)

Um Daten (flankengesteuert) zu speichern, braucht ein D-Flipflop nur einen Dateneingang und einen (Übernahme-) Takteingang.

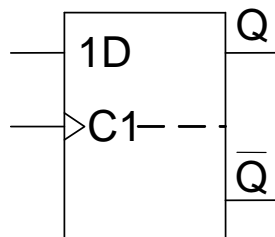
D^n	Q^{n+1}	\overline{Q}^{n+1}
0	0	1
1	1	0

Charakteristische Funktion:

$$Q^{n+1} = D^n$$

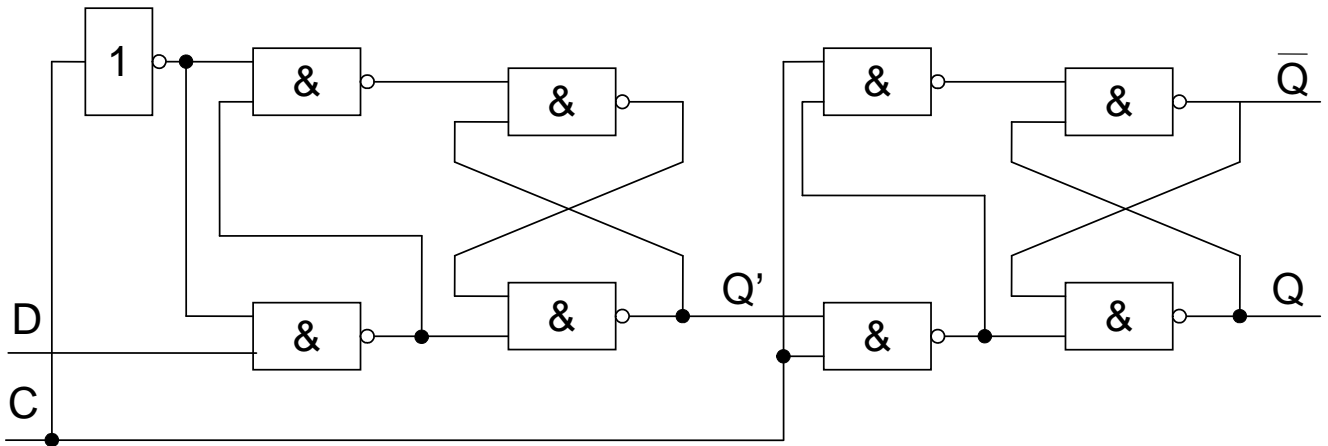
D.h., ein D-Flipflop übernimmt mit dem Taktsignal den am D-Eingang anliegenden Zustand.

Schaltsymbol



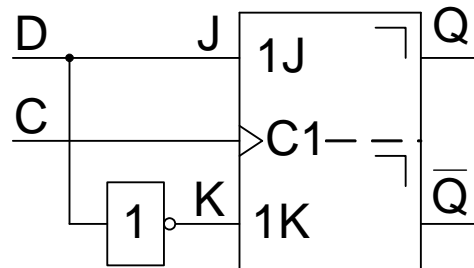
Weitere Ausführungen als taktzustandsgesteuertes und einflankengetriggertes D-FF.

Positiv einflankengetriggertes D-Flipflop



Ein D-Flipflop ist auch aus einem JK-MS-Flipflop mit $D = J = \bar{K}$ herstellbar.

Zweiflankensteuerung:



VHDL-Code für ein taktzustandsgest. D-Flipflop („Latch“)

```
LIBRARY ieee;                                -----
USE ieee.std_logic_1164.ALL;                -- D Latch    --
ENTITY D_LATCH IS                            -----
    PORT (clk,d0      : IN std_logic;
          q0          : OUT std_logic);
END D_LATCH ;
```

```
ARCHITECTURE arc OF D_LATCH IS
BEGIN
    PROCESS (d0,e0)
    BEGIN
        IF clk = '1' THEN -- transparent
            q0<=d0;
        END IF;
    END PROCESS;
END arc;
```

VHDL-Code für ein flankengetr. D-Flipflop mit Reset

```
LIBRARY ieee;                                -----
USE ieee.std_logic_1164.ALL;                -- D-FF with Reset --
ENTITY D_FFLOP IS                            -----
    PORT (clk, n_reset, d      : IN      std_logic;
          q                    : OUT     std_logic);
END D_FFLOP;
```

```
ARCHITECTURE arc OF D_FFLOP IS
BEGIN
    PROCESS (clk, n_reset)
    BEGIN
        IF (n_reset = '0') THEN -- negative active
            q<='0';
        ELSE -- positive edge-triggered
            IF (clk'EVENT AND clk = '1') THEN
                q<=d;
            END IF;
        END IF;
    END PROCESS;
END arc;
```

5.2.6 T-Flipflop oder W-Flipflop (T = Trigger, Toggle)

Falls beim JK-(MS)-Flipflop $E = J = K = 1$ ist und ein Takt angelegt wird, ergibt sich ein Wechselflipflop.

E^n	Q^{n+1}	\overline{Q}^{n+1}
0	Q^n	\overline{Q}^n
1	\overline{Q}^n	Q^n

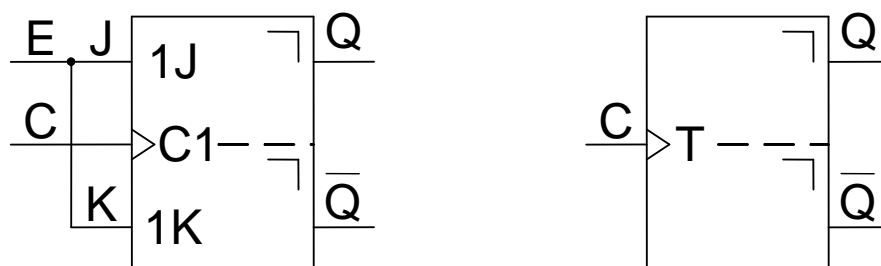
Charakteristische Funktion

$$Q^{n+1} = E\overline{Q}^n + \overline{E}Q^n$$

D.h., damit ein T-Flipflop seinen Zustand ändert, muss der E-Eingang (oft auch T genannt) = 1 sein.

Soll der Zustand unabhängig von einem Steuereingang bei jedem Taktsignal wechseln, wird der Steuereingang i.d.R. nicht gezeichnet.

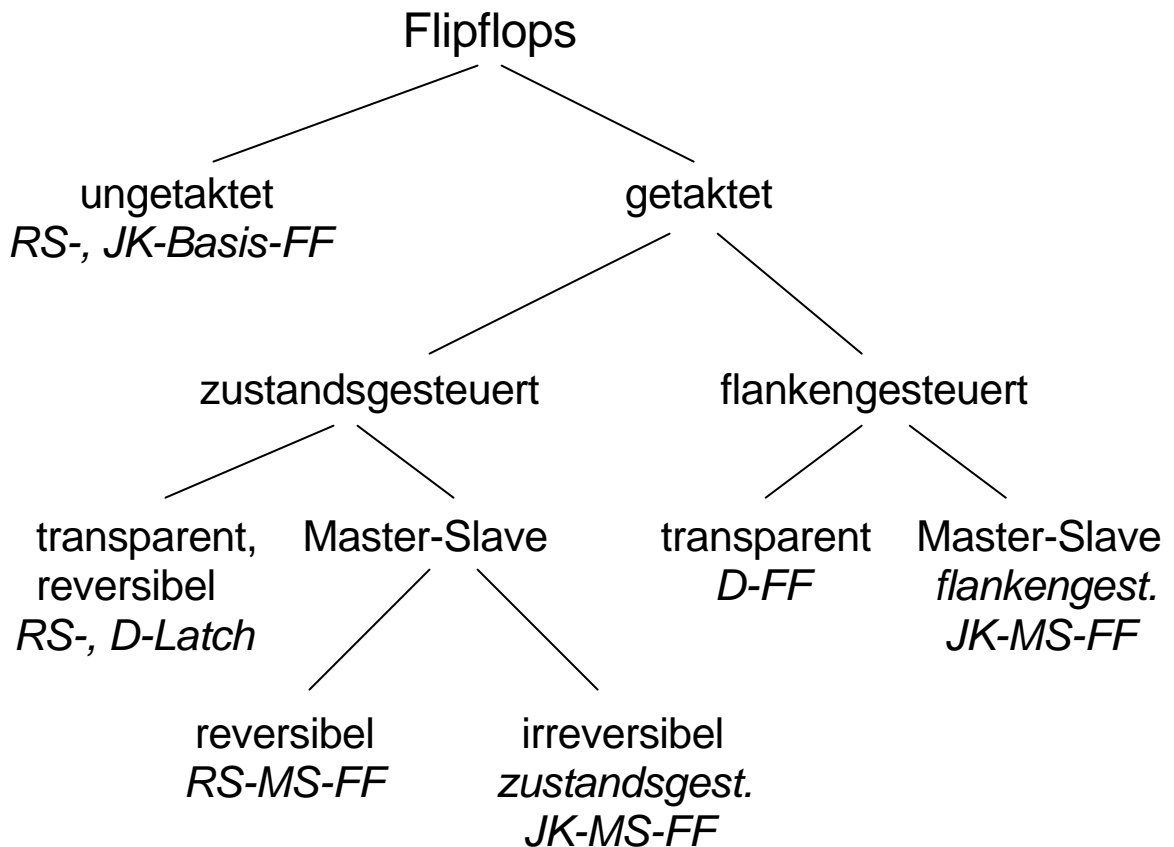
T-Flipflop mit Zweiflankensteuerung:



Varianten mit Einflankensteuerung sind ebenfalls sehr gebräuchlich.

5.2.7 Übersicht über die Flipflop-Typen

Klassifizierung von Flopflops



Transparent reversible Flipflops werden auch **Latch(es)** genannt (Zustandssteuerung). Ihr Eingang wirkt während der aktiven Taktphase bis auf den Ausgang durch.

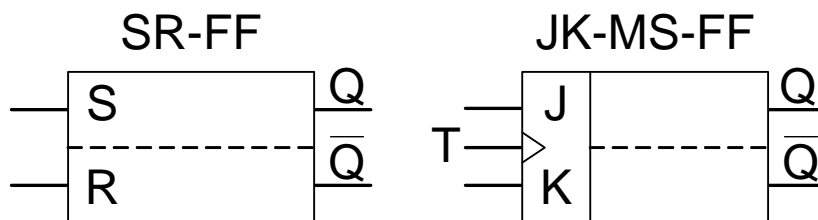
Master-Slave-Flipflops haben einen *retardierten* Ausgang. D.h., ihr Zustand wird zu Beginn der aktiven Taktphase übernommen, ändert sich aber erst verzögert beim Abfall der aktiven Taktphase.

Schaltsymbole

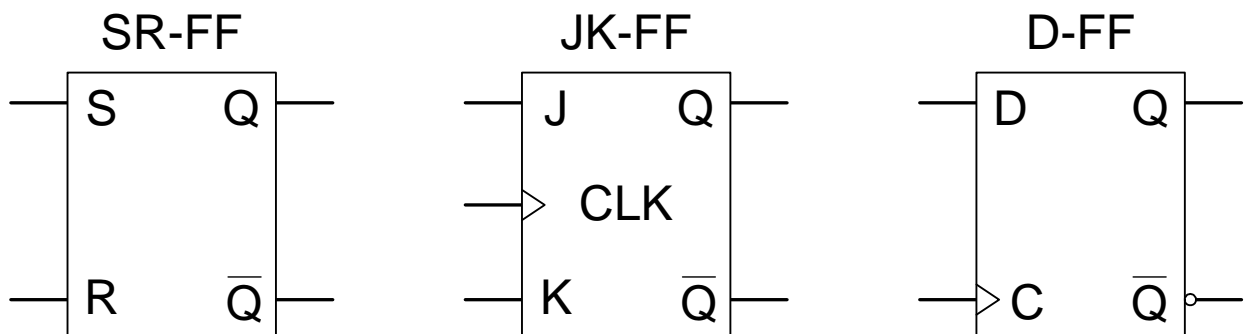
Neue DIN-Norm

	ohne Taktsteuerung	positive				negative	
		Zustandssteuerung	Zwei-Zustandssteuerung	Einflankensteuerung	Zweiflankensteuerung	Zustandssteuerung	Flankensteuerung
RS - FF							
D - FF							
JK - FF							
T - FF							

Alte DIN-Norm, z. B.:



Amerikanische Symbole (weniger normiert), z. B.:



Für MS-Flipflops i. Allg. keine eigenen Symbole.

Charakteristische Gleichungen

Flipflop-Typ	Charakteristische Gleichung
RS-Flipflop (Latch)	$Q^{n+1} = \overline{R}^n Q^n + S^n$
RS-MS-Flipflop	$Q^{n+1} = \overline{R}^n Q^n + S^n$
D-Latch (pulsgetriggert)	$Q^{n+1} = D^n$
D-Flipflop (flankengetr.)	$Q^{n+1} = D^n$
D-Flipflop mit (pos.) Enable	$Q^{n+1} = E^n \cdot D^n + \overline{E}^n \cdot Q^n$
JK-MS-Flipflop	$Q^{n+1} = J\overline{Q}^n + \overline{K}Q^n$
T-Flipflop	$Q^{n+1} = \overline{Q}^n$
T-Flipflop mit Enable	$Q^{n+1} = E\overline{Q}^n + \overline{E}Q^n$

Weitere Flipflop-Varianten:

Praktisch zu allen Flipfloptypen gibt es weitere Varianten z. B. mit:

- zusätzlichen Reset(Clear)/Preset-Eingängen oder
- zusätzlichen Gattern an den Setz-/Rücksetz-Eingängen, z.B. zum (asynchronen) Freischalten der Datenübernahme durch ein weiteres Eingangssignal (Enable) oder
- einer universellen Nutzungsmöglichkeit durch entsprechende externe Beschaltung (z.B. TTL-IC SN7474 als „Universal-D-Flipflop“)

SN5474, SN54LS74A, SN54S74 SN7474, SN74LS74A, SN74S74

DUAL D-TYPE POSITIVE-EDGE-TRIGGERED FLIP-FLOPS WITH PRESET AND CLEAR

SDLS119 – DECEMBER 1983 – REVISED MARCH 1988

- Package Options Include Plastic "Small Outline" Packages, Ceramic Chip Carriers and Flat Packages, and Plastic and Ceramic DIPs
- Dependable Texas Instruments Quality and Reliability

description

These devices contain two independent D-type positive-edge-triggered flip-flops. A low level at the preset or clear inputs sets or resets the outputs regardless of the levels of the other inputs. When preset and clear are inactive (high), data at the D input meeting the setup time requirements are transferred to the outputs on the positive-going edge of the clock pulse. Clock triggering occurs at a voltage level and is not directly related to the rise time of the clock pulse. Following the hold time interval, data at the D input may be changed without affecting the levels at the outputs.

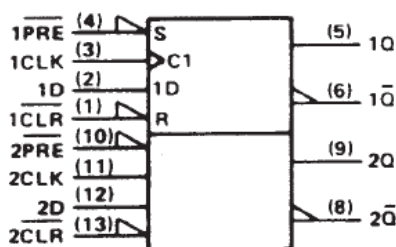
The SN54' family is characterized for operation over the full military temperature range of -55°C to 125°C . The SN74' family is characterized for operation from 0°C to 70°C .

FUNCTION TABLE

INPUTS				OUTPUTS	
$\overline{\text{PRE}}$	$\overline{\text{CLR}}$	CLK	D	Q	$\overline{\text{Q}}$
L	H	X	X	H	L
H	L	X	X	L	H
L	L	X	X	H [†]	H [†]
H	H	↑	H	H	L
H	H	↑	L	L	H
H	H	L	X	Q ₀	$\overline{\text{Q}}_0$

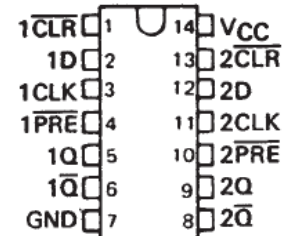
[†] The output levels in this configuration are not guaranteed to meet the minimum levels in V_{OH} if the lows at preset and clear are near V_{IL} maximum. Furthermore, this configuration is nonstable; that is, it will not persist when either preset or clear returns to its inactive (high) level.

logic symbol[†]

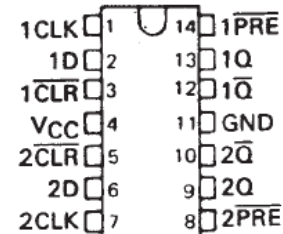


[†]This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12. Pin numbers shown are for D, J, N, and W packages.

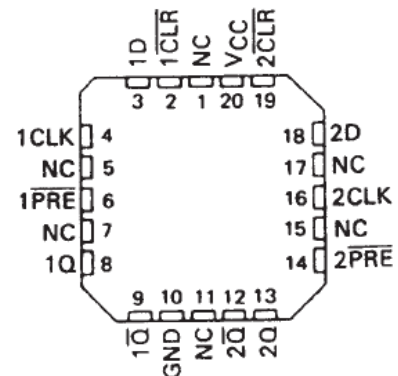
SN5474 . . . J PACKAGE
SN54LS74A, SN54S74 . . . J OR W PACKAGE
SN7474 . . . N PACKAGE
SN74LS74A, SN74S74 . . . D OR N PACKAGE
(TOP VIEW)



SN5474 . . . W PACKAGE
(TOP VIEW)

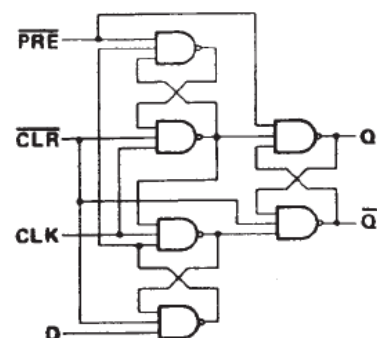


SN54LS74A, SN54S74 . . . FK PACKAGE
(TOP VIEW)



NC - No internal connection

logic diagram (positive logic)



5.3 Weitere zeitabhängige Schaltglieder

Neben den *Flipflops* („*bistabile Kippstufen*“) gibt es:

- *Monoflops*, auch *monostabile Kippstufen* genannt

Monoflops nehmen nach einem Triggerimpuls für eine vordefinierte Zeit den aktiven Zustand ein und fallen danach wieder zurück in den inaktiven Zustand.

Man unterscheidet *retriggerbare* und *nicht retriggerbare* Monoflops.

Retriggerbare Monoflops bleiben im aktiven Zustand, falls sie während der aktiven Phase einen neuen Triggerimpuls bekommen. Die Ablaufzeit startet dann jeweils neu.

Nicht retriggerbare Monoflops können erst wieder erneut getriggert werden, wenn die vorgegebene Zeit abgelaufen ist.

Eine typische Anwendung von Monoflops sind *Zeitgeber*, vorzugsweise *Timer* genannt.

- *Oszillatoren*, auch *Multivibratoren* oder *astabile Kippstufen* genannt

Oszillatoren wechseln ihren Zustand periodisch in vorgegebenen Zeitabständen.

Eine typische Anwendung von Oszillatoren sind *Taktgeneratoren*.

Wie bei den Flipflops gibt es auch bei den anderen Formen von Kippstufen weitere Steuereingänge zum Freischalten, (asynchronen) Setzen oder Löschen etc.