1. Zahlensysteme:

(2)

131072	65536	32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1
2^{17}	2^{16}	2^{15}	2^{14}	2^{13}	2^{12}	2^{11}	2^{10}	2^{9}	2^{8}	2^{7}	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}

(16)

16^5	16^4	16^3	16^2	16^1	16^0

Größter und Kleinster Wert einer vorzeichenbehafteten Dualzahl mit n-Bits:

$Z_{max} = 2^{n-1} - 1$ MSB für Vorzeichen deshalb n - 1

$Z_{min} = - 2^{n-1}$

Addition im positiven Zahlenraum: einfach Rechnen, 0 + 0 = 0, 1 + 0 = 1, 1 + 1 = 0 1 gemerkt, 1 + 1 + 1 = 1 1 gemerkt

Subtraktion: (a – b = ?) → b in ZK umformen → (a + (ZKb)) rechnen → Ergebnis auf Gültigkeit prüfen (ü Regel unten)

Multiplikation: 1) Vorzeichen ermitteln: linkeste Bit → 1 Negativ, 0 Positiv

2) Stellen Multiplikator (n) + Stellen Multiplikator (m) = (n + m) Stellen Produkt

3) Negativ mit 1sen auffüllen auf (n + m) insgesamt, Positiv mit 0len auffüllen auf (n + m) insgesamt

4) Bei negativen Ergebnis ZK bilden um Betrag herauszufinden (für negative Darstellung nicht nötig)

Allgemein: $ü_{n+1} = ü_n$ → Korrektes Ergebnis

$ü_{n+1} \neq ü_n$ → Ergebnis im x-Bit-Rahmen ungültig/ Überlauf, Erweiterung erforderlich, Übertrag weg!!!

Dezimal	Binär	Gray-Code	BCD-Code
0	0000	0000	0000
1	0001	0001	0001
2	0010	0011	0010
3	0011	0010	0011
4	0100	0110	0100
5	0101	0111	0101
6	0110	0101	0110
7	0111	0100	0111
8	1000	1100	1000
9	1001	1101	1001
10	1010	1111	0000
11	1011	1110	0001
12	1100	1010	0010
13	1101	1011	0011
14	1110	1001	0100
15	1111	1000	0101

1.) Boolsche Algebra

Bezeichnung	Funktion	Schaltzeichen
Leistungsstufe, Buffer, Treiber	$Q = A$	A ▷ Q
Inverter, NICHT, NOT	$Q = \overline{A}$	A 1 o Q
Inverter mit Negation am Eingang	$Q = A$	A o 1 Q
UND AND	$Q = A \wedge B$ $Q = A \cdot B = AB$	A, B & Q
NAND	$Q = \overline{A \wedge B}$ $Q = \overline{A \cdot B} = \overline{AB}$	A, B & o Q
ODER OR	$Q = A \vee B$ $Q = A + B$	A, B ≥1 Q
NOR	$Q = \overline{A \vee B}$ $Q = \overline{A + B}$	A, B ≥1 o Q
Exklusives ODER XOR, EXOR	$Q = \bar{A}B \vee A\bar{B}$ $Q = \bar{A}B + A\bar{B}$	A, B =1 Q
EX-NOR XNOR	$Q = \overline{\bar{A}B \vee A\bar{B}}$ $Q = \overline{\bar{A}B + A\bar{B}}$	A, B =1 o Q

#	Name	UND-Operator	ODER-Operator
1	Identität	$a \cdot 1 = a$	$a + 0 = a$
2	Elimination	$a \cdot 0 = 0$	$a + 1 = 1$
3	Idempotenz	$a \cdot a = a$	$a + a = a$
4	Involution	$\overline{\overline{a}} = a$	
5	Komplement	$a \cdot \overline{a} = 0$	$a + \overline{a} = 1$
6	Kommutativität	$a \cdot b = b \cdot a$	$a + b = b + a$
7	Assoziativität	$(a \cdot b) \cdot c = a \cdot (b \cdot c)$	$(a + b) + c = a + (b + c)$
8	Distributivität	$a \cdot (b + c) = (a \cdot b) + (a \cdot c)$	$a + (b \cdot c) = (a + b) \cdot (a + c)$
9	Vereinigung	$(a + b) \cdot (a + \overline{b}) = a$	$(a \cdot b) + (a \cdot \overline{b}) = a$
10	Absorption (1)	$a \cdot (a + b) = a$	$a + (a \cdot b) = a$
11	Absorption (2)	$(a \cdot \overline{b}) + b = a + b$	$(a + \overline{b}) \cdot b = a \cdot b$
11	Absorption (2.1)	$(a \cdot b) + \overline{b} = a + \overline{b}$	$(a + b) \cdot \overline{b} = a \cdot \overline{b}$
12	Faktorisierung	$(a + b) \cdot (\overline{a} + c) = (a \cdot c) + (\overline{a} \cdot b)$	$(a \cdot b) + (\overline{a} \cdot c) = (a + c) \cdot (\overline{a} + b)$
13	Konsens	$(a + b) \cdot (b + c) \cdot (\overline{a} + c) = (a + b) \cdot (\overline{a} + c)$	$(a \cdot b) + (b \cdot c) + (\overline{a} \cdot c) = (a \cdot b) + (\overline{a} \cdot c)$
14	De Morgans Gestz	$\overline{a \cdot b \cdot \ldots} = \overline{a} + \overline{b} + \cdots$	$\overline{a + b + \cdots} = \overline{a} \cdot \overline{b} \cdot \ldots$
15	Negations-theorem	$\overline{f(x_1, x_2, \ldots, x_n, 0, 1, \cdot, +)} = f(\overline{x_1}, \overline{x_2}, \ldots, 1, 0, +, \cdot)$	

2.) KV-Diagramm

(1) Boolsche Funktion in DNF überführen
(2) KV-Diagramm konstruieren → 2^n = Felder für n-Variablen → bei Skalierung darf nur immer eine var sich ändern
(3) In allen Feldern die den Mintermen entsprechen eine 1, beim Rest 0
Spezialfall: man trägt Maxterme ein → 0-len eintragen → später verunden → Negationsumkehr → KNF
(4) Möglichst große mit 1 besetzten Rechtecken suchen → über Diagramm hinausdenken, Überlappung erlaubt
(5) Jedes Rechteck ist eine Und-Funktion, der Variablen die konstant bleiben, diese bei DNF verodern
(6) Ab 5 Variablen müssen die Blöcke Faltsymmetrisch liegen

3.) Quine-Mc-Cluskey-Verfahren

(1) Boolsche Funktion in DNF überführen, Alternativ Wahrheitstabelle → Minterme
(2) Sortieren der Minterme nach Gruppen →Kriterium Negationsstriche
(3) Vereinfachungsprinzip benachbarte Gruppen vergleichen
(4) Vereinfacht: Abhacken, Nicht Vereinfacht: Primterm
(5) Primterme entscheiden wenn Spalte fertig ist
(6) Bis man nur noch 2er verknüpfungen hat
(7) Primterm-Minterm-Tabelle → Zeilen: Minterme, Spalten:Primterme
(8) Eintragen der Touchpoints
(9) Zeilenweise schauen: Wo sind die Kreuze allein
(10)Welche Primterme brauchen wir um alle Minterme abzudecken
(11)Die resultierenden Primterme verodern z.B Z = P1 + P2 + P3 + P4

Gruppe	Minterme		1. Vereinfachung		2. Vereinfachung	
0	$\bar{A}\bar{B}\bar{C}\bar{D}$	✓	$\bar{A}\bar{B}\bar{C}$	✓	$\bar{A}\bar{B}$	P2
1	$\bar{A}\bar{B}\bar{C}D$	✓	$\bar{A}\bar{B}\bar{D}$	✓	$\bar{A}\bar{B}$	
	$\bar{A}\bar{B}C\bar{D}$	✓	$\bar{A}\bar{C}\bar{D}$	P1	$\bar{B}\bar{D}$	P3
	$\bar{A}B\bar{C}\bar{D}$	✓	$\bar{B}\bar{C}\bar{D}$	✓	$\bar{B}\bar{D}$	
	$A\bar{B}\bar{C}\bar{D}$	✓	$\bar{A}\bar{B}D$	✓	$\bar{B}C$	P4
			$\bar{A}\bar{B}C$	✓	$\bar{B}C$	
			$\bar{B}C\bar{D}$	✓	AC	P5
			$A\bar{B}\bar{D}$	✓	AC	
2	$\bar{A}\bar{B}CD$	✓	$\bar{B}CD$	✓		
	$A\bar{B}C\bar{D}$	✓	$A\bar{B}C$	✓		
3	$A\bar{B}CD$	✓	$AC\bar{D}$	✓		
	$ABC\bar{D}$	✓	ACD	✓		
4	$ABCD$	✓	ABC	✓		

→ Primterm	P1	P2	P3	P4	P5
→ Minterm	$\bar{A}\bar{C}\bar{D}$	$\bar{A}\bar{B}$	$\bar{B}\bar{D}$	$\bar{B}C$	AC
$\bar{A}\bar{B}\bar{C}\bar{D}$	X	X	X		
$\bar{A}\bar{B}\bar{C}D$		X			
$\bar{A}\bar{B}C\bar{D}$		X	X	X	
$\bar{A}B\bar{C}\bar{D}$	X				
$A\bar{B}\bar{C}\bar{D}$			X		
$\bar{A}\bar{B}CD$		X		X	
$A\bar{B}C\bar{D}$			X	X	X
$A\bar{B}CD$				X	X
$ABC\bar{D}$					X
$ABCD$					X

disjunktive Normalform (DNF): $Z = (Minterm1\ mit\ Wert1) + (Minterm2\ mit\ Wert1) + \dots$

konjunktive Normalform (KNF): $Z = (Maxterm1\ mit\ Wert0) \cdot (Maxterm2\ mit\ Wert0) \cdot \dots$

Vollkonjunktion-Minterm: Und-Verknüpfung (Konjunktion)

Volldisjunktion-Maxterm: Oder-Verknüpfung (Disjunktion)

Sämtliche Variablen nur 1x bejaht, 1x beneint.
Bei n vorkommenden Variablen: Anzahl = 2^n

Disjunktive Minimalform, Konjunktive Minimalform = Ergebnis nach der Vereinfachung mit den Methoden.

Umrechnung DNF/KNF: $F(A,B,C) = \sum m(1,2,3,\dots) = \prod M(1,2,3,\dots)$

Kanonische Normalform – Ringsummennormalform (RNF):

- Zeichen: $\oplus$ →XOR, exklusives ODER →
- Keine negierten Variablen

A	B	Z
0	0	0
1	0	1
0	1	1
1	1	0

Kommutativität	$x \oplus y = y \oplus x$
Assoziativität	$x \oplus (y \oplus z) = (x \oplus y) \oplus z$
Distributivität bzgl. Konjunktion	$x\,(y \oplus z) = x\,y \oplus x\,z$
	$x + y = x \oplus y \oplus x\,y = x \oplus \overline{x}\,y = y \oplus \overline{y}\,x$
	$x \oplus 1 = \overline{x}, \quad x \oplus 0 = x$
	$x \oplus x = 0, \quad x \oplus \overline{x} = 1$
	$0 \oplus 0 \oplus \dots \oplus 0 = 0$
	$1_{(1)} \oplus 1_{(2)} \oplus \dots \oplus 1_{(n)} = \begin{cases} 1\ falls\ n\ ungerade \\ 0\ falls\ n\ gerade \end{cases}$

1.) Methode RNF zu bestimmen:

(1) Voraussetzung: boolscher Ausdruck in DNF
(2) Tabelle aufzeichen →Wahrheitstabelle mit allen Möglichen Variationen → Minterme der DNF → 1
(3) Spaltenbeschriftung nach Schema unten machen:

Anzahl Variablen	n = 0	n = 1	n = 2	n = 3	n = 4
F(x₁..xₙ) =	a_0				
		$\oplus a_1x_1$	$\oplus a_2x_2$	$\oplus a_3x_3$	$\oplus a_4x_4$
			$\oplus a_{12}x_1x_2$	$\oplus a_{13}x_1x_3$ $\oplus a_{23}x_2x_3$	$\oplus a_{14}x_1x_4$ $\oplus a_{24}x_2x_4$ $\oplus a_{34}x_3x_4$
				$\oplus a_{123}x_1x_2x_3$	$\oplus a_{124}x_1x_2x_4$ $\oplus a_{134}x_1x_3x_4$ $\oplus a_{234}x_2x_3x_4$
					$\oplus a_{1234}x_1x_2x_3x_4$

x_3	x_2	x_1	y	1	x_1	x_2	$x_1\,x_2$	x_3	$x_1\,x_3$	$x_2\,x_3$	$x_1\,x_2\,x_3$	Basisterme	
				a_0	a_1	a_2	a_{12}	a_3	a_{13}	a_{23}	a_{123}		
0	0	0	0	* 0								g	0
0	0	1	1	0	1							u	1
0	1	0	1	0		1						u	1
0	1	1	0	0	1	1	0					g	2
1	0	0	1	0				1				u	1
1	0	1	0	0	1			1	0			g	2
1	1	0	0	0		1		1		0		g	2
1	1	1	1	0	1	1	0	1	0	0	0	u	3

(4) Y betrachten: Y = 0 → gerade Anzahl der Basisterme (0,2,4,6,8,…)
Y = 1 → ungerade Anzahl der Basisterme (1,3,5,7,…)
(5) Bei a_0 gilt der eingetragene Wert für gesamte Spalte.
Im Verlauf gilt der Wert nur für Zeilen, wo Spaltenvariable Wert 1 hat
(6) Koeffizienten an Treppe ablesen: $F_{(x1,x2,x3)\,RNF} = 0 \oplus 1\,x_1 \oplus 0\,x_2 \oplus 0\,x_1x_2 \oplus \dots$

2.) Methode RNF zu bestimmen:

(1) Nur wenn es eine kanonische DNF oder eine ortogonale DNF ist →Prüfung ist KV-Diagramm ohne Überlappung
(2) Regel + = $\oplus$
(3) Alle negierten Variablen durch XOR 1 ersetzen (z.B. $\overline{x_1} = x_1 \oplus 1$)
(4) Ausmultiplizieren
(5) $x \oplus x = 0 \quad oder \quad x \oplus 0 = x$ anwenden
(6) RNF ist erzeugt

3.Shannon-Zerlegung mit ROBDD etc.

Hilfestellung:

$\overline{0} = 1$

$\overline{1} = 0$

Bestimmung der Subfunktionen bzw. Kofaktoren

$$F_0 = F(x_1, x_2, x_3)|_{x_1=0} =$$

$$F_1 = F(x_1, x_2, x_3)|_{x_1=1} =$$

$$F_{00} = F_0(x_2, x_3)|_{x_2=0} =$$

$$F_{01} = F_0(x_2, x_3)|_{x_2=1} =$$

$$F_{10} = F_1(x_2, x_3)|_{x_2=0} =$$

$$F_{11} = F_1(x_2, x_3)|_{x_2=1} =$$

$$F_{000} = F_{00}(x_3)|_{x_3=0} =$$

$$F_{001} = F_{00}(x_3)|_{x_3=1} =$$

$$F_{010} = F_{01}(x_3)|_{x_3=0} =$$

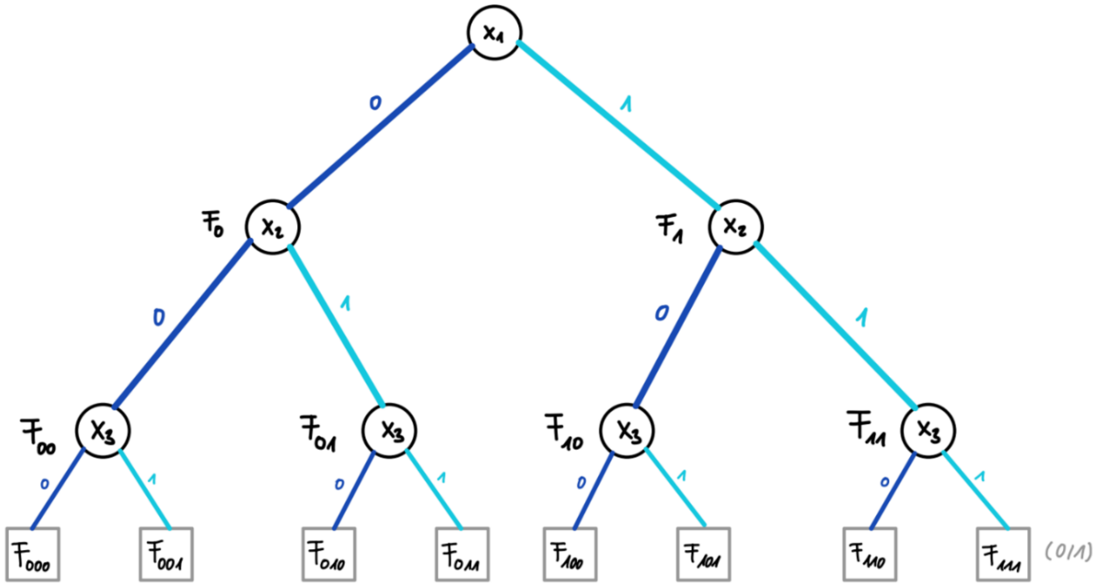
$$F_{011} = F_{01}(x_3)|_{x_3=1} =$$

$$F_{100} = F_{10}(x_3)|_{x_3=0} =$$

$$F_{101} = F_{10}(x_3)|_{x_3=1} =$$

$$F_{110} = F_{11}(x_3)|_{x_3=0} =$$

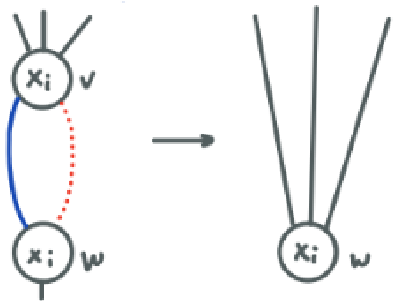
$$F_{111} = F_{11}(x_3)|_{x_3=1} =$$



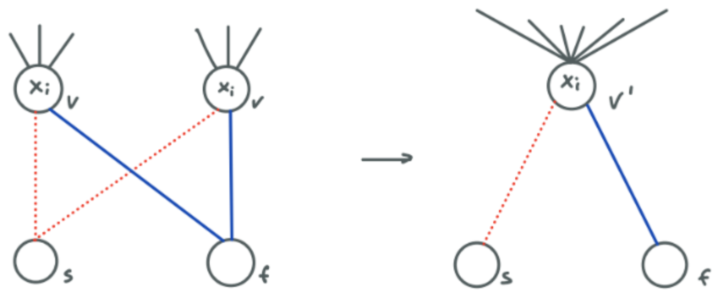
Kanonische Normalform – Binary-Decision-Diagramm (BDD)

- Ordered-Binary-decision-diagramm (OBDD) = auf jedem Pfad alle Variablen höchstens einmal
- Reduced ordered-binary-decision-diagramm (ROBDD)

1. Regel zum kürzen **Deletion Rule**



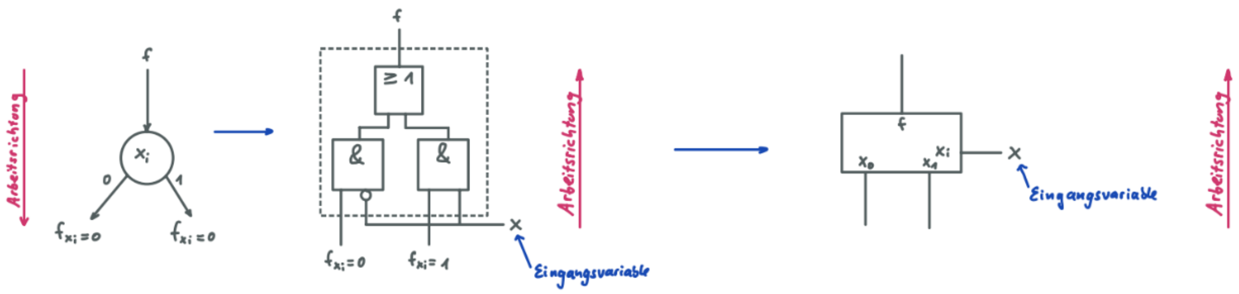
2. Regel zum kürzen **Merging Rule**



Anwendung:

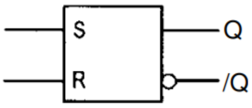
(1) Voraussetzung: DNF
(2) Bestimmung der Sub- und Ko-funktionen → einsetzen von 1 und 0 in ausgewählter Variable
(3) BDD zeichnen
(4) Kürzen dann hat man ROBDD

Überführung in Schaltzeichen:



RS-Flipflop:

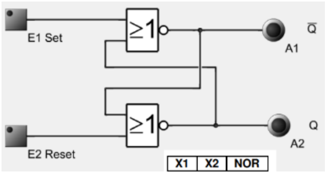
(nicht Taktgesteuert)

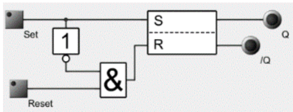


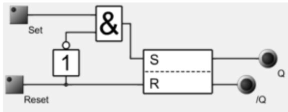
Set	Reset	Q	/Q	
0	0	$Q_{(n-1)}$	$/Q_{(n-1)}$	Speichern
0	1	0	1	Rücksetzen
1	0	1	0	Setzen
1	1	0	0	Irregulär

Dominierend Set			Dominierend Reset		
1	0	**Setzen**	0	1	**Rücksetzen**

Besteht aus:

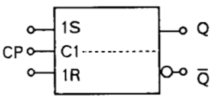






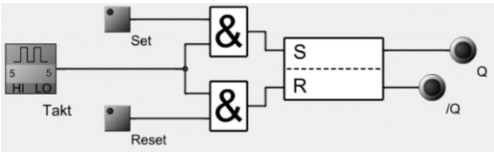
RS-Flipflop:

(**Taktgesteuert**) → Set oder Reset nur möglich bei High-Pegel-Takt



Set	Reset	Q	/Q	
0	0	$Q_{(n-1)}$	$/Q_{(n-1)}$	Speichern
0	1	0	1	Rücksetzen
1	0	1	0	Setzen
1	1	- (0)	- (0)	Irregulär

Besteht aus:



Tastgradformel:

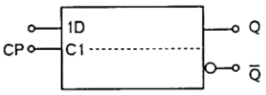
$$D = \frac{Impulsdauer\ T_e}{Periodendauer\ T}$$

charakteristische Gleichung:

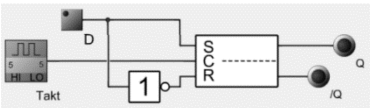
$$Q_{1(tn+1)} = [\overline{R}Q_1 + \ S]_{(tn)}$$

D-Flipflop (Delay-Flipflop)

(D ist Eingang → Info bleibt bis zum nächsten Tackt erhalten)



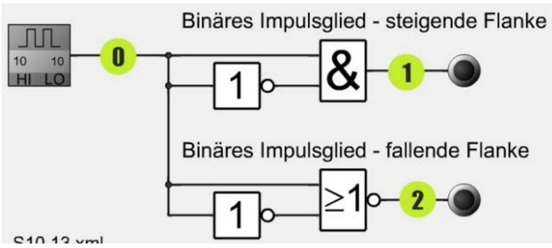
Charakteristische Gleichung: $Q_{1(tn+1)} = J = D$



t_n		t_{n+1}
D		Q
1		1
0		0

Takt	D	Q	/Q	
0	0	$Q_{(n-1)}$	$/Q_{(n-1)}$	Speichern
0	1	$Q_{(n-1)}$	$/Q_{(n-1)}$	
1	0	0	1	Rücksetzen
1	1	1	0	Setzen

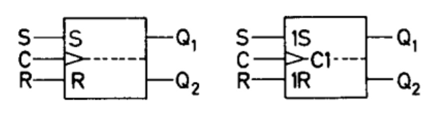
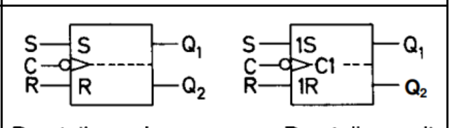
Taktflankengesteuerte Flipflops:



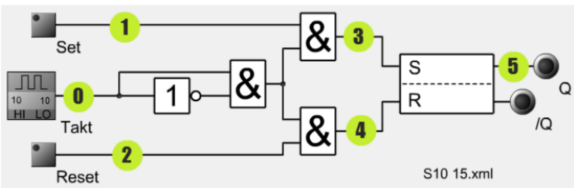
Gibt Signal für eine Signallaufzeit, wenn Takt positive Flanke hat

Gibt Signal für eine Signallaufzeit, wenn Takt fallende Flanke hat

Einflanken gesteuertes RS-Flipflop:

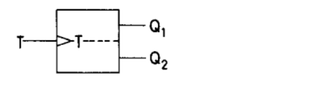
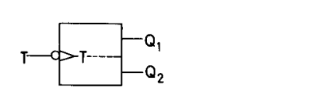
RS-Flipflop einflankengesteuert steigende Flanke	
RS-Flipflop einflankengesteuert fallende Flanke	

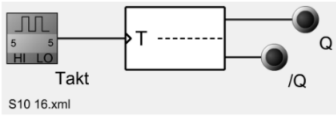
Erweiterung mit Impulsglied:

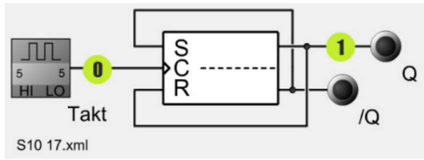


Einflankengesteuertes T-Flipflop (Trigger Flipflop oder Toggle-Flipflop):

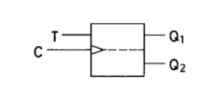
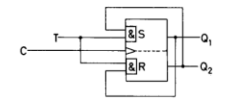
Kippt bei jeder steuernden Flanke in einen anderen Zustand:

T-Flipflop einflankengesteuert steigende Flanke	
T-Flipflop einflankengesteuert fallende Flanke	



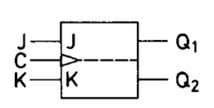
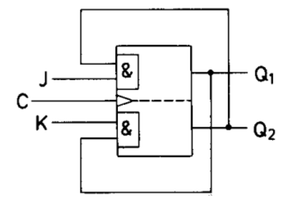


Mit C- und T- Eingang:

Schaltzeichen	Realisierung	Wahrheitstabelle
		see below

t_n	t_{n+1}
T	Q_1
0	Q_{1n}
1	$\overline{Q}_{1n}$

Einflanken gesteuertes JK-Flipflop: (J = Jump also setzen, K = Kill also rücksetzen, C = Taktgeber)

Schaltzeichen	Realisierung
	

Wahrheitstabelle

	t_n		t_{n+1}
Fall	K	J	Q_1
1	0	0	Q_{1n}
2	0	1	1
3	1	0	0
4	1	1	$\overline{Q}_{1n}$

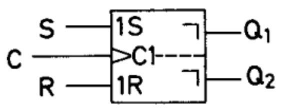
Charakteristische Gleichung:

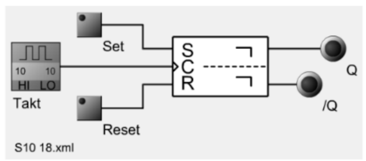
$$Q_{1(tn+1)} = [J\overline{Q}_1 + \overline{K}Q_1]_{(tn)}$$

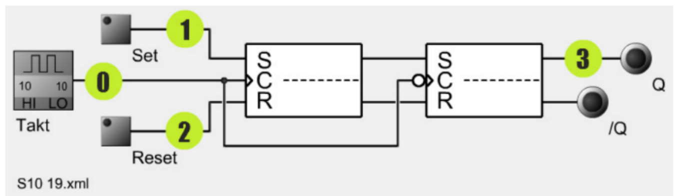
Beide 1 = toggle Fall

Zweiflankengesteuerte Flipflops (Master Slave Flipflop):

Bei steigender Taktflanke → Signalaufnahme am Eingang → Speicherung→Bei fallender Taktflanke→Ausgangsignal

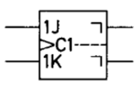
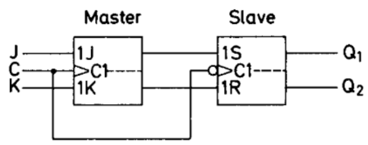




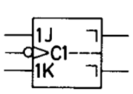
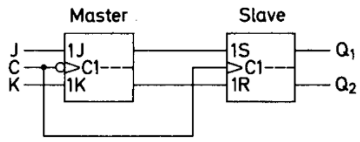


Zweiflankengesteuerte JK-Flipflops, Master-Slave:

JK-Master-Slave-Flipflop, ansteigende Flanke

Schaltzeichen	Realisierung
	

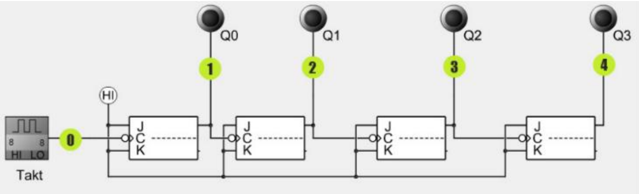
JK-Master-Slave-Flipflop, abfallende Flanke

Schaltzeichen	Realisierung
	

Zähler:

Asynchronzähler = Flipflops werden nicht zu gleichen Zeitpunkt geschaltet (Takt), geschaltet durch Vorglied

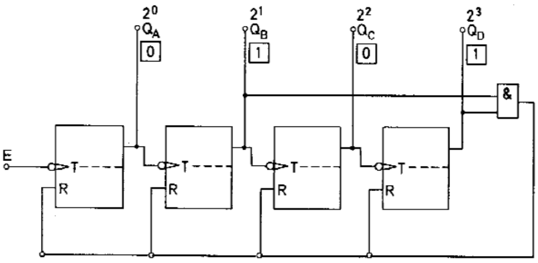
Asynchroner Dual-Vorwärtszähler

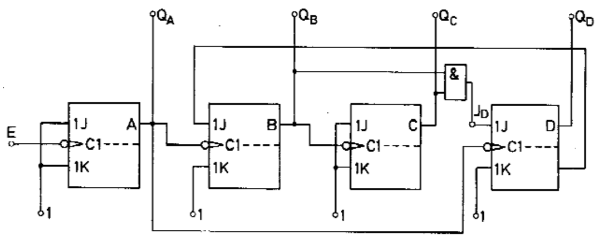


Grenzperiodendauer: $T_G = (n+1) \cdot t_p$ Grenzfrequenz: $f_G = \frac{1}{T_G}$ Nicht schneller Takten da sonst Error

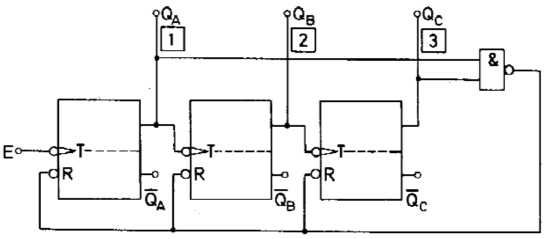
Asynchrone BCD-Zähler: mit T-Flipflop, gibt Reset bei 10 mit JK-Flipflop, schaltet auf 0000 ohne 10

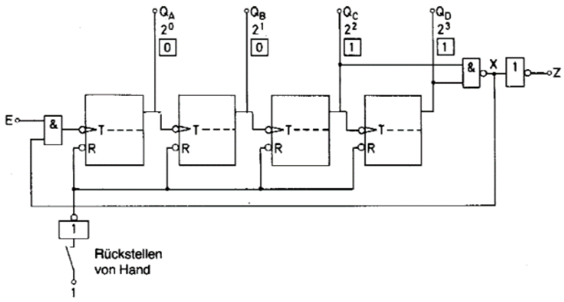
Dezimal-ziffer	2^3 Q_D	2^2 Q_C	2^1 Q_B	2^0 Q_A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1



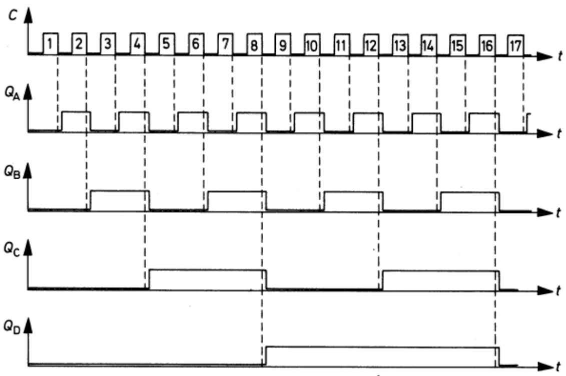


Asynchrone Modulo-n-Zähler: Modulo-5-Zähler, gibt Reset bei 5 Modulo-13-Zähler, bleibt bei 12 stehen,manuel res

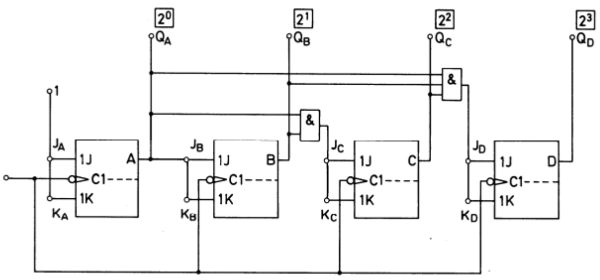




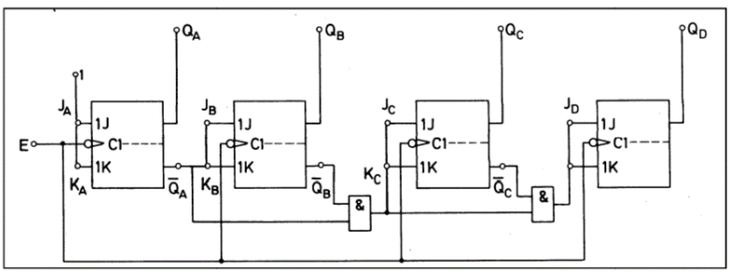
Synchronzähler = Flipflops werden zu gleichem Zeitpunkt geschaltet (Takt)

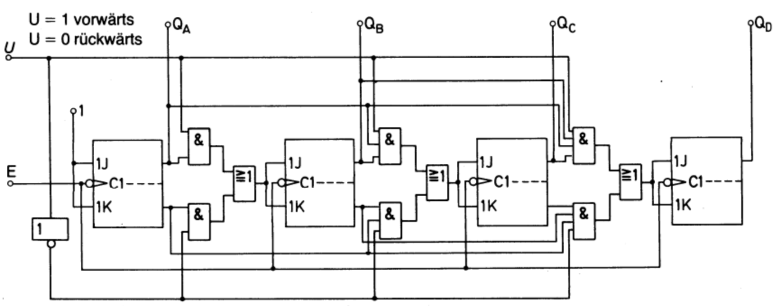


Synchroner 4-Bit-Dual-Vorwärtszähler



Synchroner 4-Bit-Rückwärtszähler **und umschaltbare Zählrichtung**





Entwicklung von zyklischen Synchronzählern „Kochrezept“

0. Überlegen wie viele Glieder man braucht (mit wie vielen Bits kann man eine Zahl darstellen)

1. Aufstellen der Wahrheitstabelle des Zählers:

Zeilen sind Dezimalzahlen, Spalten sind Glieder mit Wertigkeiten 1 2 4 8, Linke Seite t_n und rechte Seite t_{n+1}

2. Aufstellen und Vereinfachen der Anwendungsgleichung des Zählers:

Für jedes Glied der rechten Seite 1sen raussuchen und Minterme der linken Seite mit KV-Diagramm vereinfachen

Achtung auf dont Cares, da nicht alle Variationen links abgebildet sind evtl.

3. Bestimmen der charakteristischen Gleichung der Flipflops: Siehe FS für spezielles Glied oder Angabe

4. Bestimmen der Verknüpfungsgleichung (J und K Anschlüsse herausfinden) durch Koeffizientenvergleich

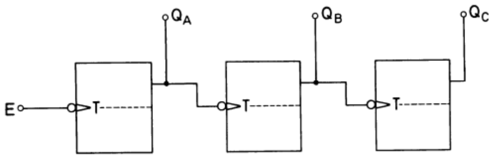
5. Zeichnen des Schaltbildes

<u>Frequenzteiler:</u>

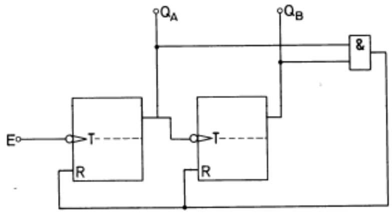
Asynchron Frequenzteiler mit festem Teilerverhältnis: (pro Bauteil geteilt durch 2)

$f_T = \frac{f_E}{2^n}$ mit f_T = geteilte Frequenz, f_E = Eingangsfrequenz, n = Anzahl der Flipflops

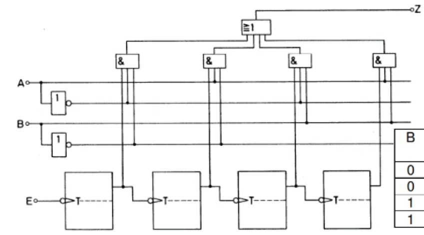
Gerades Teiler Verhältnis: (8:1)



Ungerades Teiler Verhältnis: (3:1)

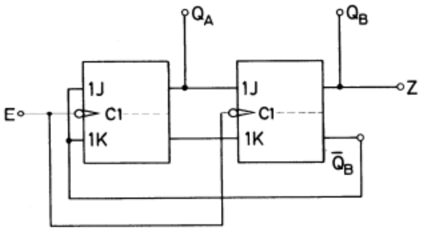


einstellbarem Teiler Verhältnis:



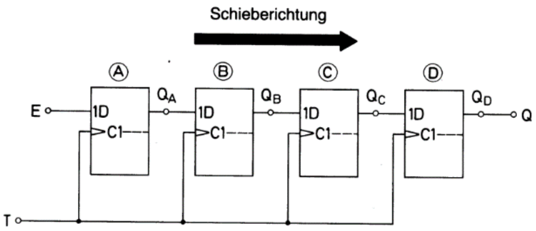
B	A	Frequenz-teilerverhältnis
0	0	2:1
0	1	4:1
1	0	8:1
1	1	16:1

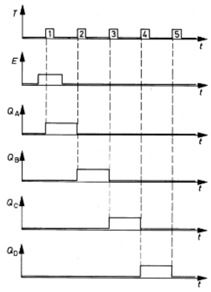
Synchrone Frequenzteiler: Beispiel (3:1) Regeln wie bei Asynchron Frequenzteiler



Schieberegister: Ketten von Flipflops, die am Eingang angelegte Info jeden Takt um ein Flipflop weiterschieben

Serieller Ein- und Ausgabe:





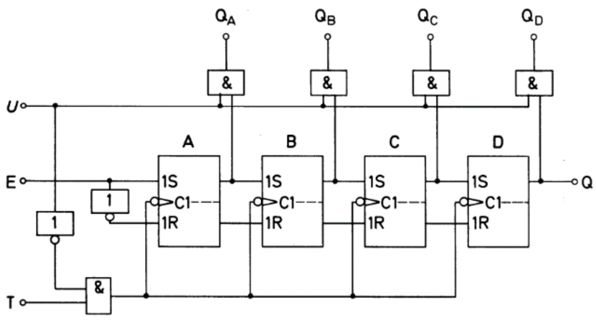
Takt Nr. n	Zustände nach Takt Nr. n E	Q_A	Q_B	Q_C	$Q_D = Q$
	1	0	0	0	0
1	0	1	0	0	0
2	0	0	1	0	0
3	0	0	0	1	0
4	0	0	0	0	1
5	0	0	0	0	0

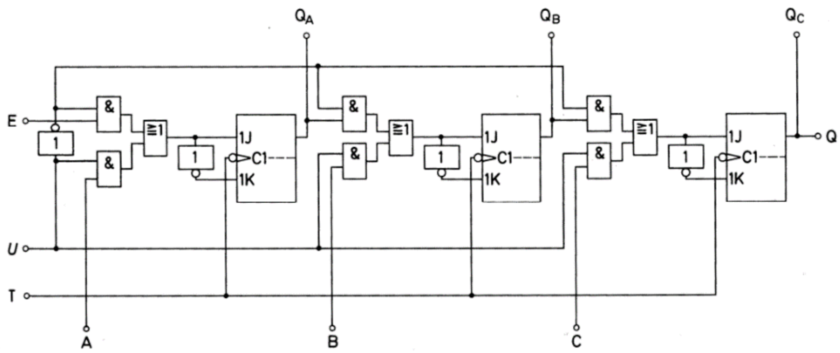
Funktionstabelle

Paralleler Ein- und Ausgabe: (kann zusätzlich die gespeicherten Daten parallel ausgeben)

U = 1 → Takt gesperrt, U = 0 → Takt freigegeben

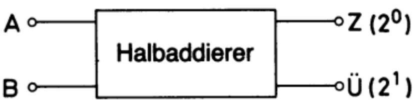
taktabhängiger paralleler Dateneingabe





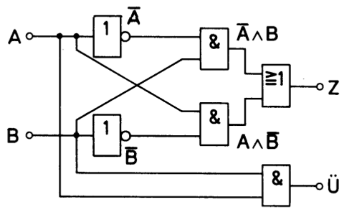
U = 1 ⟹ Parallelbetrieb
U = 0 ⟹ Serieller Betrieb
A, B, C: Eingänge für Paralleleingabe

Halbaddierer: (2 Bits addieren)

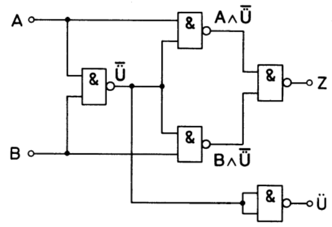


Fall	B	A	Ü	Z	
1	0	0	0	0	
2	0	1	0	1	→ $A \wedge \overline{B}$
3	1	0	0	1	→ $\overline{A} \wedge B$
4	1	1	1	0	

Ü → $A \wedge B$

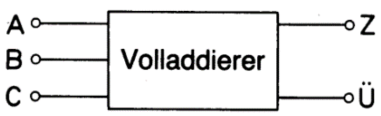


Halbaddiererschaltung mit Grundgliedern

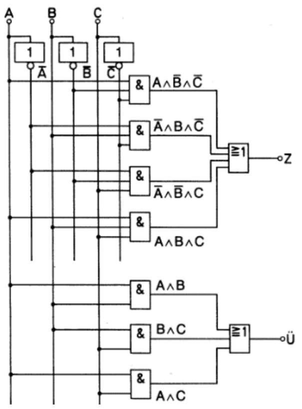


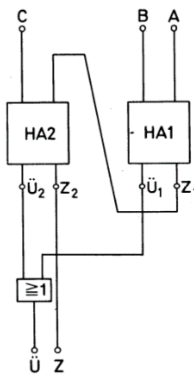
Halbaddiererschaltung mit NAND-Gliedern

Volladdierer: (3 Bits addieren)



Fall	C	B	A	Ü	Z
1	0	0	0	0	0
2	0	0	1	0	1
3	0	1	0	0	1
4	0	1	1	1	0
5	1	0	0	0	1
6	1	0	1	1	0
7	1	1	0	1	0
8	1	1	1	1	1





Übertrag: $Ü = A\,B\,\overline{C} + \overline{A}\,B\,C + A\,\overline{B}\,C + A\,B\,C$

Summe: $Z = \overline{A}\,B\,\overline{C} + A\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}\,C + A\,B\,C$

<u>**Digitale Auswahl- und Verbindungsschaltungen:**</u>

Demultiplexer:

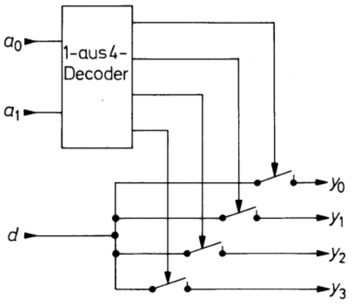


Abb. 9.54 Prinzipielle Wirkungsweise

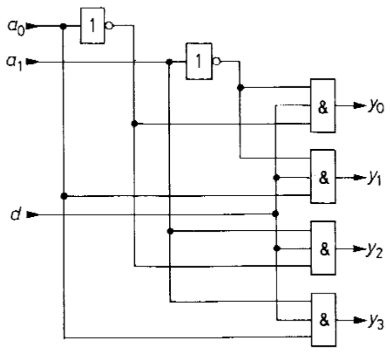
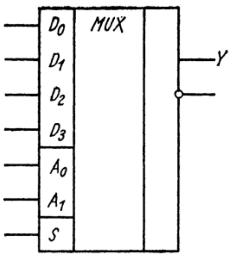


Abb. 9.55 Schaltung eines Demultiplexers

$y_0=\bar{a}_0\bar{a}_1 d,\ y_1=a_0\bar{a}_1 d,\ y_2=\bar{a}_0 a_1 d,\ y_3=a_0 a_1 d$

Multiplexer:



Vereinfachtes Schaltungssymbol

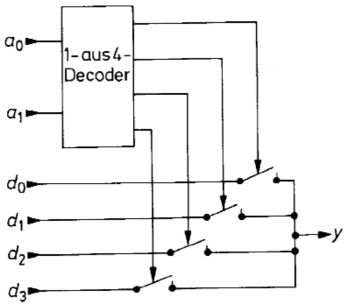


Abb. 9.57 Prinzipielle Wirkungsweise eines Multiplexers

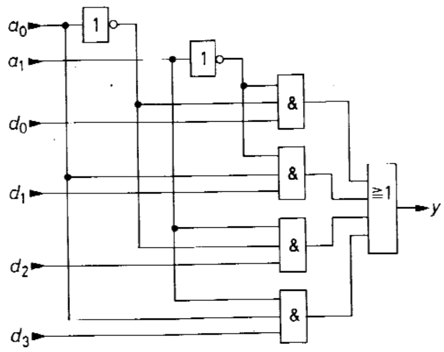


Abb. 9.58 Schaltung eines Multiplexers

$y=\bar{a}_0\bar{a}_1 d_0+a_0\bar{a}_1 d_1+\bar{a}_0 a_1 d_2+a_0 a_1 d_3$

CMOS Schaltung:

n-Kanal-Fets: Bei n-Kanal-Fets wird der Kanalstrom umso grösser, je weiter das Gatepotential steigt.
Leitet bei H-Pegel

p-Kanal-Fets: Bei p-Kanal-Fets wird der Kanalstrom umso kleiner, je weiter das Gatepotential steigt.
Leitet bei L-Pegel

Anreicherungstyp: (enhancement) – selbstsperrend Bei einem Anreicherungstyp ist bei der Gatespannung von 0 V noch kein leitfähiger Kanal vorhanden; er muss erst durch die Gatespannung erzeugt werden.