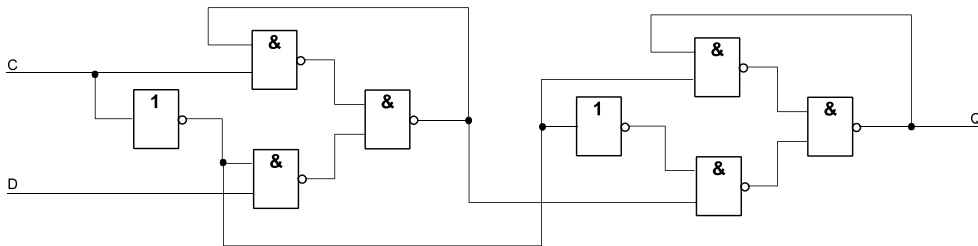


Musterlösungen

zu den Übungsaufgaben vom 2. 2. 07

1. Geben Sie an (Skizze, ggf. Funktionserläuterung), wie ein D-Flipflop auf Grundlage von 2-zu-1-Multiplexern aufgebaut werden kann.

SO:



Es sind zwei aus 2-zu-1-Multiplexern aufgebaute D-Latches hintereinanderschalten. Das erste Latch (der Master) übernimmt die Daten, wenn der Takt Low ist. Geht der Takt auf High, wird das erste Latch gesperrt (Datenbelegung bleibt stehen) und die Datenbelegung in das zweite Latch (den Slave) übernommen. Ob's funktioniert? – Wir werden sehen (im Praktikum...).

2. Minimieren Sie folgende Schaltfunktion mittels Karnaugh-Plan:

$$\bar{a} \bar{b} \bar{c} \bar{d} \vee \bar{a} \bar{b} c \bar{d} \vee \bar{a} b \bar{c} d \vee \bar{a} b c d \vee a \bar{b} \bar{c} \bar{d} \vee a \bar{b} c d \vee a b \bar{c} d \vee a b c \bar{d}$$

		CD:					
		0 0	0 1	1 1	1 0		
AB:	0 0	1			1	0 0	
		<small>0 = $\bar{a} \bar{b} \bar{c} \bar{d}$</small>	<small>1 = $\bar{a} \bar{b} c \bar{d}$</small>	<small>3 = $\bar{a} \bar{b} c d$</small>	<small>2 = $\bar{a} \bar{b} c \bar{d}$</small>		
0 1		1	1			0 1	
		<small>4 = $\bar{a} b \bar{c} \bar{d}$</small>	<small>5 = $\bar{a} b c \bar{d}$</small>	<small>7 = $\bar{a} b c d$</small>	<small>6 = $\bar{a} b c \bar{d}$</small>		
1 1						1 1	
		<small>12 = $a b \bar{c} \bar{d}$</small>	<small>13 = $a b c \bar{d}$</small>	<small>15 = $a b c d$</small>	<small>14 = $a b c \bar{d}$</small>		
1 0		1			1	1 0	
		<small>8 = $a \bar{b} \bar{c} \bar{d}$</small>	<small>9 = $a \bar{b} c \bar{d}$</small>	<small>11 = $a \bar{b} c d$</small>	<small>10 = $a \bar{b} c \bar{d}$</small>		
		0 0	0 1	1 1	1 0		
		CD:					
		$\bar{a} b d \vee \bar{b} \bar{d}$					

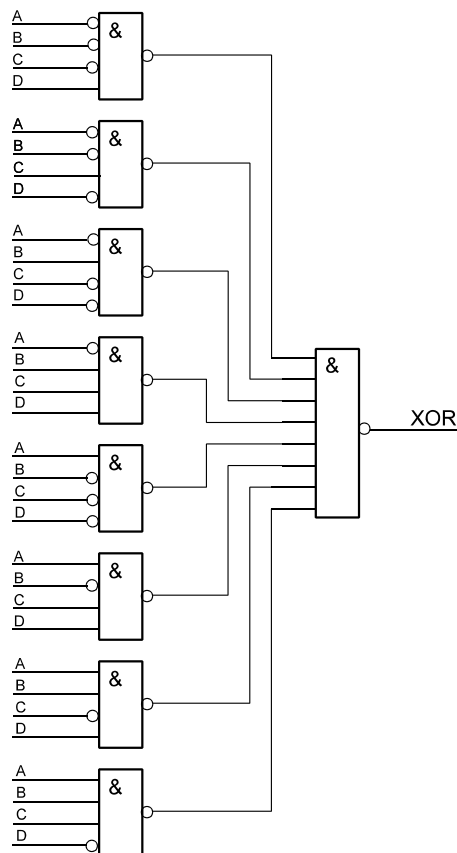
3. Eine State Machine soll mit ROMs (als Zuordner) aufgebaut werden. Welches Prinzip der Zustandscodierung wählen Sie?

Unbedingt die binäre Codierung. Jedes gesparte Bit zählt – es vermindert die erforderliche Speicherkapazität um die Hälfte. Die Kompliziertheit der Booleschen Gleichungen spielt keine Rolle, weil der ROM alle 2^n Belegungen der Wahrheitstabelle speichert.

4. Bauen Sie ein XOR-Gatter mit vier Eingängen aus NAND-Gattern mit beliebiger Eingangszahl (Schaltbild).

Es muß immer dann eine Eins herauskommen, wenn die Anzahl der Einsen an den Eingängen ungerade ist.

A	B	C	D	Einsen	XOR
0	0	0	0	0	0
0	0	0	1	1	1
0	0	1	0	1	1
0	0	1	1	2	0
0	1	0	0	1	1
0	1	0	1	2	0
0	1	1	0	2	0
0	1	1	1	3	1
1	0	0	0	1	1
1	0	0	1	2	0
1	0	1	0	2	0
1	0	1	1	3	1
1	1	0	0	2	0
1	1	0	1	3	1
1	1	1	0	3	1
1	1	1	1	4	0



5. Ein 4-Bit-Binärzähler (Abb. 1) soll modulo 11 zählen, also mit jedem 12. Zähltakt wieder den anfänglichen Zählerstand einnehmen. Geben Sie eine Beschaltung an, die bewirkt, daß der Zähler nacheinander die Stellungen 0 – 1 – 2 ... – 10 – 0 – 1 usw. durchläuft.
6. Wir beziehen uns nochmals auf Aufgabe 5 und Abb. 1. Geben Sie eine weitere Schaltung an, in der der Zähler modulo 11 zählt, die aber – im Gegensatz zur Lösung von Aufgabe 5 – mit ersichtlich weniger Aufwand auskommt. Welche Stellungen der Zähler durchläuft, ist gleichgültig.
7. Welche maximale Zykluslänge hat ein linear rückgekoppeltes Schieberegister mit 12 Flipflops?

$$\text{Zykluslänge} = 2^n - 1 = 2^{12} - 1 = 4095.$$

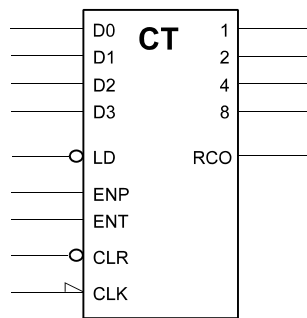
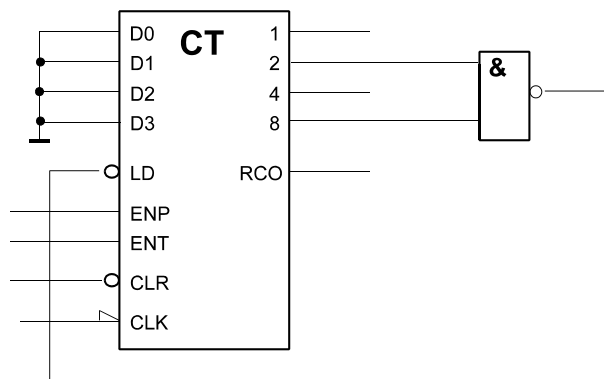
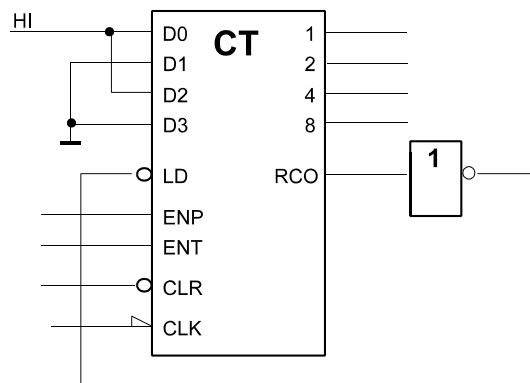


Abb. 1

Zu Aufg. 6: Stellung 10 = 1010B decodieren und auf den Ladeeingang zurückführen, so daß Zähler mit Null geladen wird. Vereinfachte Decodierung möglich, weil in Stellung 1010B Bitposition 2 erstmals dann auf 1 geht, wenn Bitposition 8 = 1 ist. Als weitere Übungsgelegenheit: unter welcher Bedingung darf man den CLR-Eingang zum Löschen des Zählers ausnutzen?



Zu Aufg. 7: Wir nutzen den Ausgangsübertrag aus. Ausgangsübertrag wird in Stellung 16 wirksam. Um bis 15 modulo 11 zu zählen, muß mit Stellung 5 begonnen werden (16 - 11 (Zweierkomplement)). Rückführung des invertierten Ausgangsübertrags auf den Ladeeingang. Festbeschaltung der Dateneingänge mit Wert 5 = 0101B.



8. Abb. 2 zeigt die Innenschaltung eines Decoderschaltkreises. Versuchen Sie, sich in die Funktionsweise der Schaltung hineinzudenken. Was stellen die zwischen den beiden Fragezeichen (?...?) angeordneten Schaltmittel dar und wozu sind sie vorgesehen?

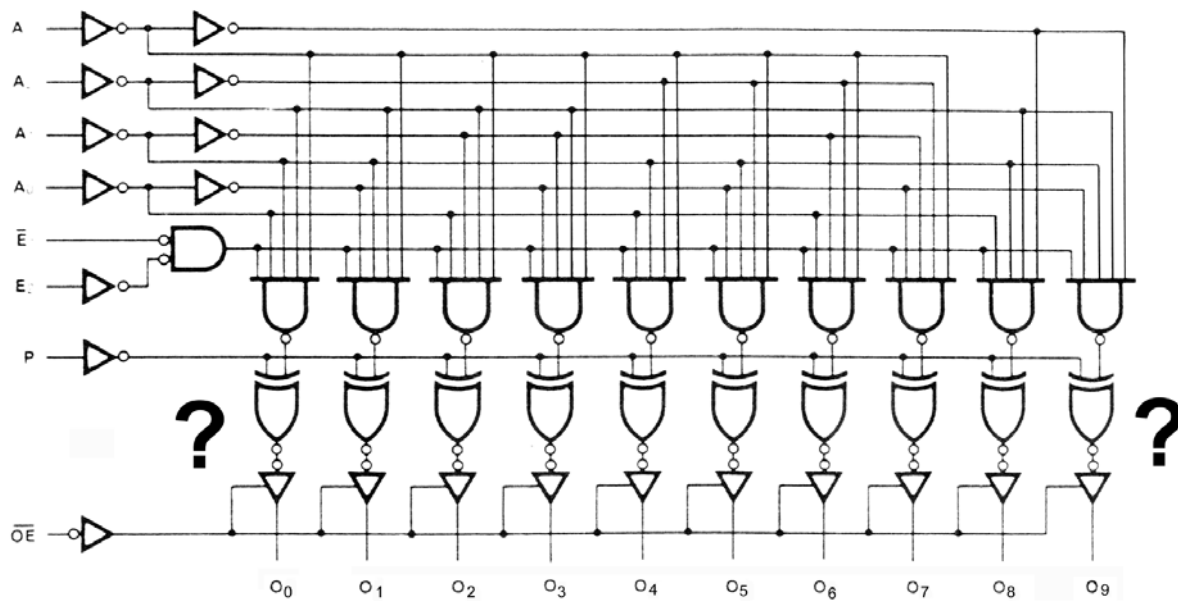
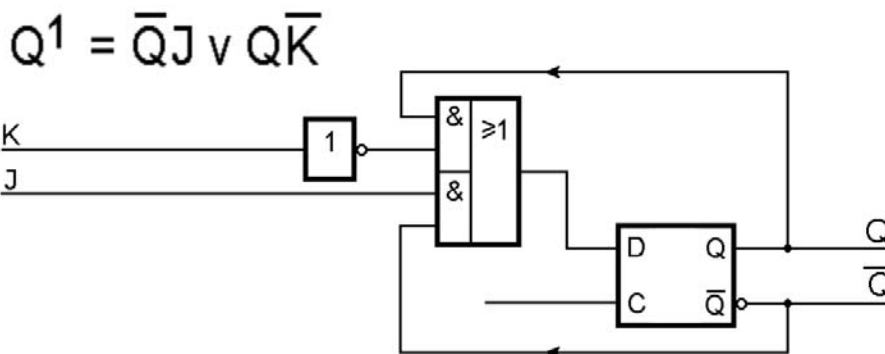


Abb. 2

Der Schaltplan zeigt Antivalenzgatter (XORs) mit negierten Ausgängen (also Äquivalenzgatter bzw. XNORs), denen jeweils unmittelbar eingangsseitige Negatoren der Ausgangsstufen nachgeschaltet sind. Ein solcher Negator hebt die vorgeschaltete Negation sofort wieder auf. Somit wirken die Funktionselemente als Antivalenzverknüpfungen (XORs). Deren Zweck: die Wirkungsweise der Ausgänge zwischen aktiv High und aktiv Low umschalten zu können, und zwar über den Steuereingang P. Der Vorteil: man kommt mit einem einzigen Schaltkreistyp aus und kann trotzdem Decoder mit "wahren" und mit invertierten Ausgängen anbieten (im praktischen Einsatz wird P typischerweise mit einem Festwert belegt).

9. In einer Schaltung werden JK-Flipflops benötigt, es sind aber nur D-Flipflops verfügbar. Geben Sie eine kombinatorische Beschaltung an, die ein D-FF in ein JK-FF umwandelt.

Das D-Flipflop hat den Vorteil, daß man die Schaltgleichung des gesuchten Flipfloptyps nur als Gatternetzwerk bauen und dem D-Eingang vorschalten muß – einfacher geht's nich...



10. Am Eingang einer Schaltung messen Sie die dargestellten Signalverläufe (Abb. 3). Was erwarten Sie am Ausgang? (Signalverlauf in idealisierter Form in das Diagramm einzeichnen.)

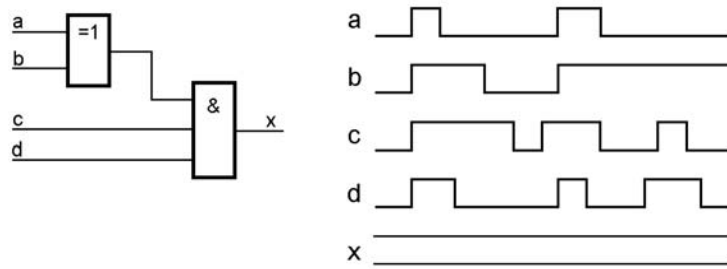
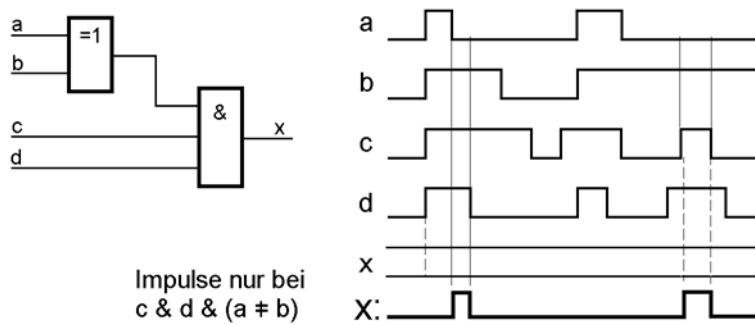


Abb. 3

Es kommt nur dann ein Impuls heraus, wenn c und d beide aktiv sind und wenn a und b ungleich sind.



Impulse nur bei c & d & (a ≠ b)

11. Welche Schaltfunktion realisiert die in Abb. 4 gezeigte Anordnung? (Angabe als Wahrheitstabelle und Schaltgleichung.)

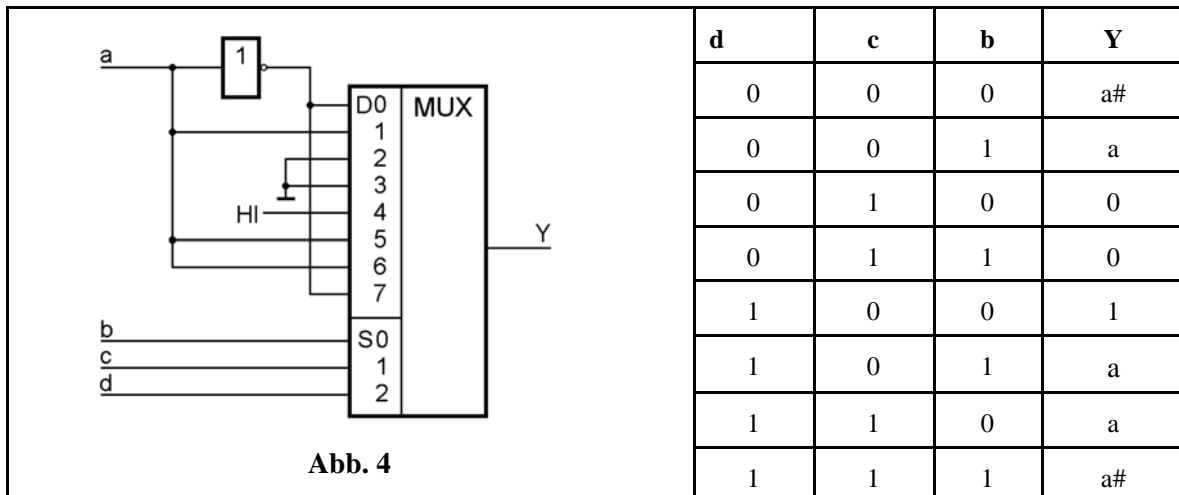
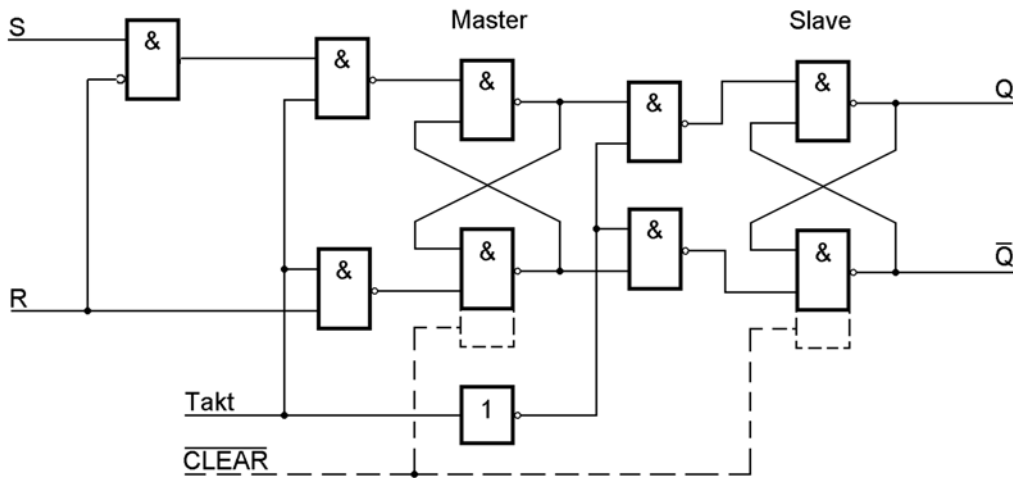


Abb. 4

$$Y = \bar{d}\bar{c}\bar{b}\bar{a} \vee \bar{d}\bar{c}b\bar{a} \vee \bar{d}c\bar{b}\bar{a} \vee \bar{d}c b\bar{a} \vee d\bar{c}\bar{b}\bar{a} \vee d\bar{c}b\bar{a}$$

12. Geben Sie die Gatter-Schaltung eines RS-Master-Slave-Flipflops an, bei der die "verbotene" Belegung R = S = 1 zugelassen ist und die gleiche Wirkung hat wie R = 1, S = 0 (Rücksetzen).

Wenn R aktiv ist, wird S blockiert, so daß bei R S = 1 nur R zur Wirkung kommt.



13. Entwerfen Sie ein vollsynchrones 4-Bit-Register (Abb. 5) mit den Funktionen gemäß der folgenden Tabelle. Grundlage: D-Flipflops sowie beliebige Gatter.

Vorrangregeln (wenn zwei oder mehr Steuersignale gleichzeitig aktiv sind):

- CLR dominiert über alle anderen Funktionen.
- CLR und PRE dominieren über die Ladefunktionen.
- LD dominiert über LDN.

Signal	Funktion
LD	Laden
LDN	Laden invertiert
PRE	Setzen (alle Stellen = 1)
CLR	Löschen (alle Stellen = 0)
-	Datenbelegung halten

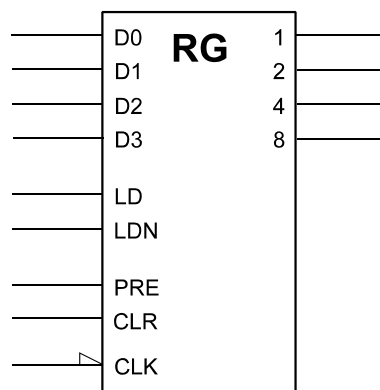
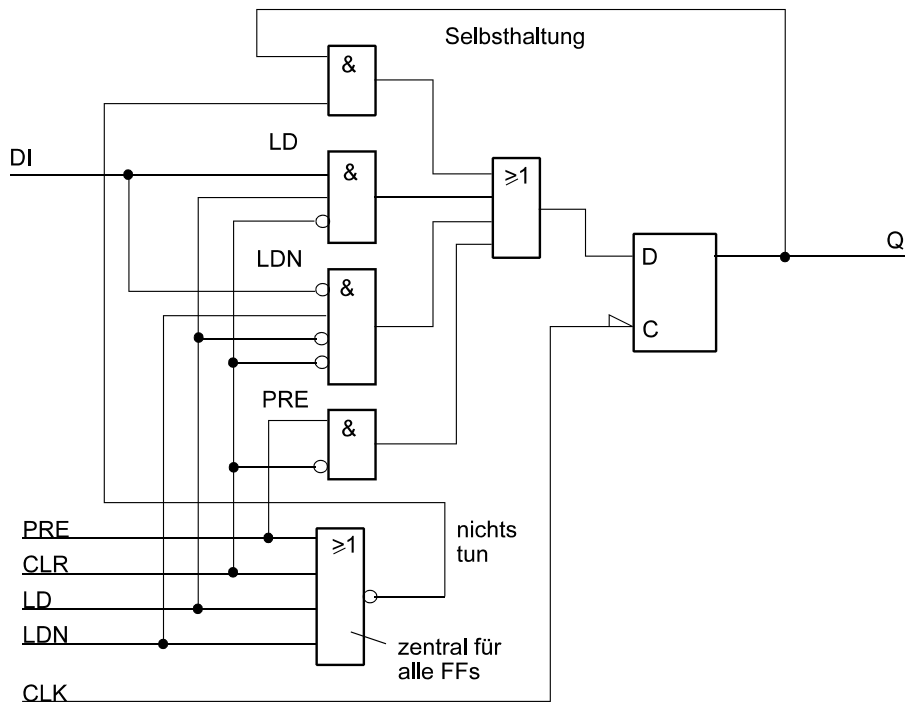


Abb. 5



14. Führen Sie folgende Umrechnungen aus (Ergebnisse in Tabelle eintragen):

binär in hexadezimal		hexadezimal in binär		dezimal in hexadezimal	
1100 1101B	CDH	C69H	1100 0110 1001B	99	63H
11 1101 0100B	3D4H	8BFH	1000 1011 1111	210	D2H

Alle Zahlen sind vorzeichenlos.