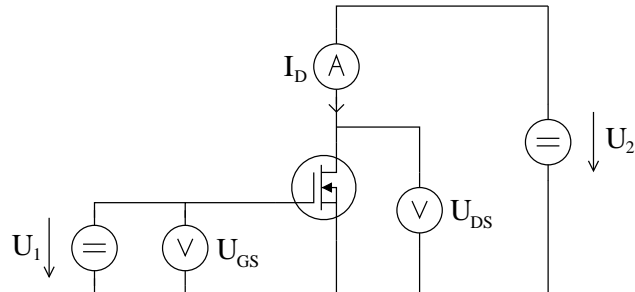


Versuch 1

Versuch 1.2

Messung des Kennlinienfeldes $I_D = f(U_{DS})$ des N-Kanal-MOS-FET mit Parameter $U_{GS} = 3V$ und $4,5V$.

Messschaltung:

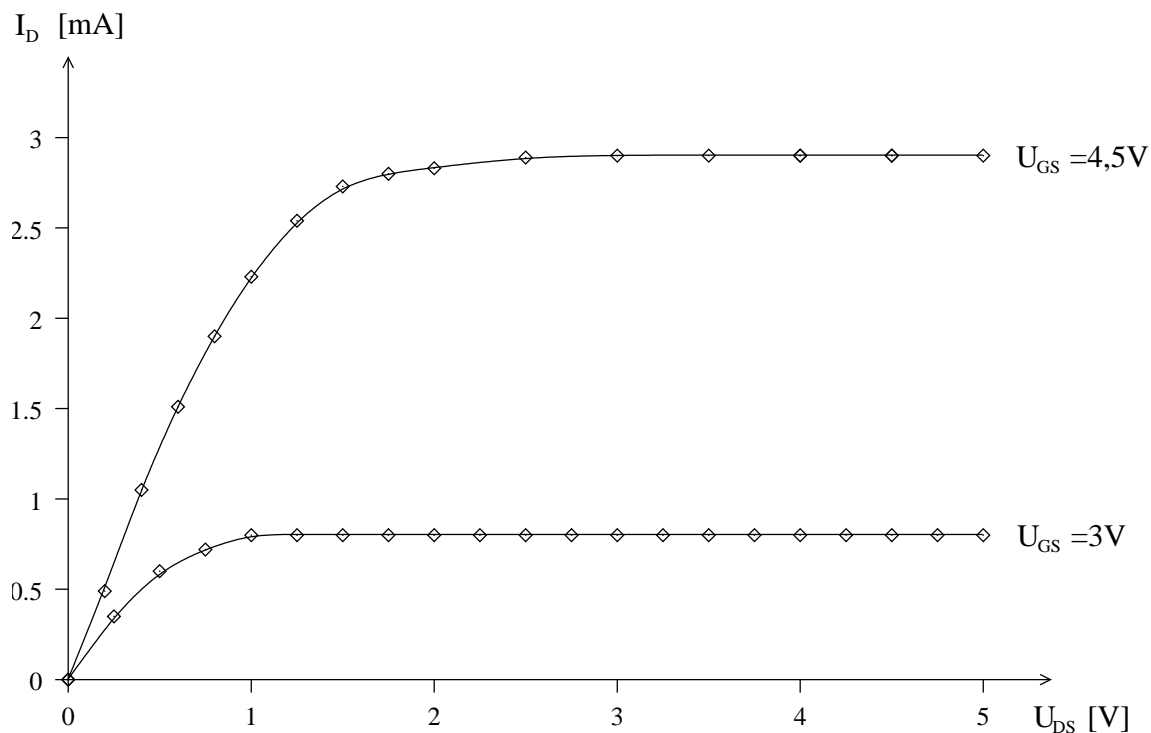


Gemessene Werte:

$U_{GS} = 3V$	
U_{DS} [V]	I_D [mA]
0	0.0
0.25	0.35
0.5	0.6
0.75	0.72
1	0.8
1.25	0.8
1.5	0.8
1.75	0.8
2	0.8
2.25	0.8
2.5	0.8
2.75	0.8
3	0.8
3.25	0.8
3.5	0.8
3.75	0.8
4	0.8
4.25	0.8
4.5	0.8
4.75	0.8
5	0.8

$U_{GS} = 4.5V$	
U_{DS} [V]	I_D [mA]
0	0
0.2	0.49
0.4	1.05
0.6	1.51
0.8	1.9
1	2.23
1.25	2.54
1.5	2.73
1.75	2.8
2	2.83
2.5	2.89
3	2.9
3.5	2.9
4	2.9
4.5	2.9
4	2.9
4.5	2.9
5	2.9

Graphische Darstellung der gemessenen Werte:



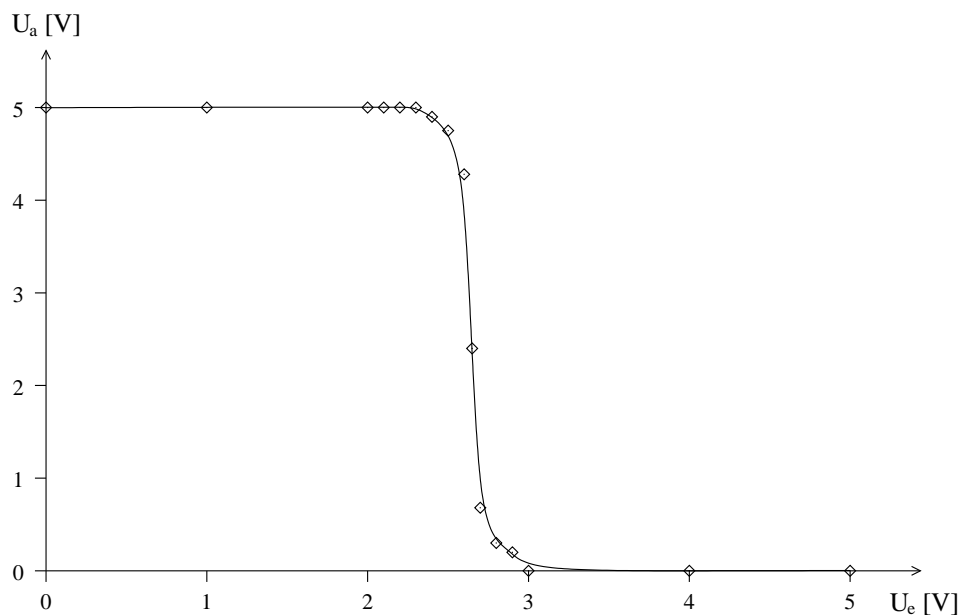
Versuch 1.4

Messung der Transfercharakteristik des CMOS-Inverters.

Gemessene Werte:

Graphische Darstellung der gemessenen Werte:

$U_e [V]$	$U_a [V]$
0	5
1	5
2	5
2.1	5
2.2	5
2.3	5
2.4	4.9
2.5	4.75
2.6	4.28
2.65	2.4
2.7	0.68
2.8	0.3
2.9	0.2
3	0
4	0
5	0



Versuch 1.5

Messung des dynamischen Verhaltens des CMOS-Inverters. Gemessene Verzögerung (mit Kapazität $C = 100\text{pF}$): $10\mu\text{s}$.

Versuch 1.7

Realisierung eines NAND-Gatters mit zwei Eingängen.

Signalnamen	Verbindungen		
E1 = S0	FALA'	1A3	
E2 = S1	FALB'	1A6	
A = D0	FALA	1A1	1A13
	1A4	1A8	
	1A2	1A14	

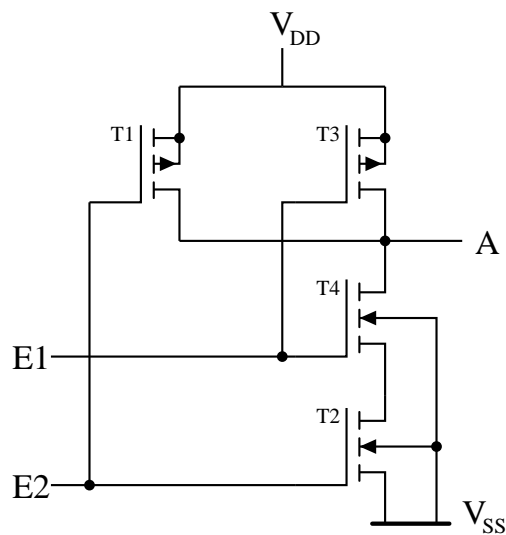


Abbildung 1: Schaltung Versuch 1.7

Versuch 1.8

Realisierung eines NOR-Gatters mit zwei Eingängen.

Signalnamen	Verbindungen			
E1 = S0	FALA'	1A6		
E2 = S1	FALB'	1A3		
A = D0	FALA	1A1	1A8	1A5
	1A2	1A13		
	1A4	1A7		

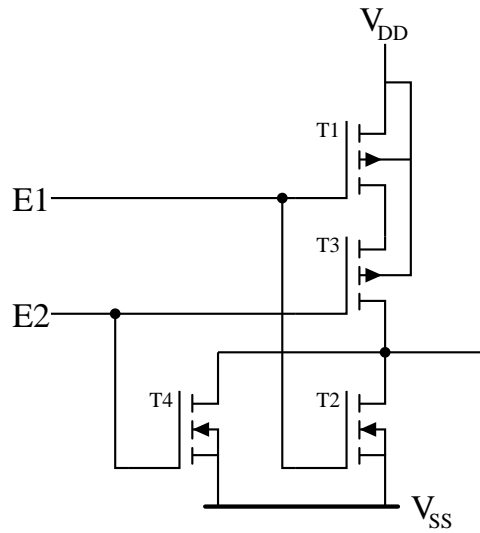


Abbildung 2: Schaltung Versuch 1.8

Versuch 1.9

Realisierung eines Exklusiv-OR-Gatters mit zwei Eingängen.

Signalnamen	Verbindungen			
E1 = S0	FALA'	1A6	1B3	
E2 = S1	FALB'	1A3	1B6	
	1A13	1A2		
	1A1	1A8	1A5	1A10
	1A4	1A9	1A7	
A = D0	FALA	1A12	1B5	
	1A11	1B13	1B1	
	1B4	1B8		
	1B2	1B14		

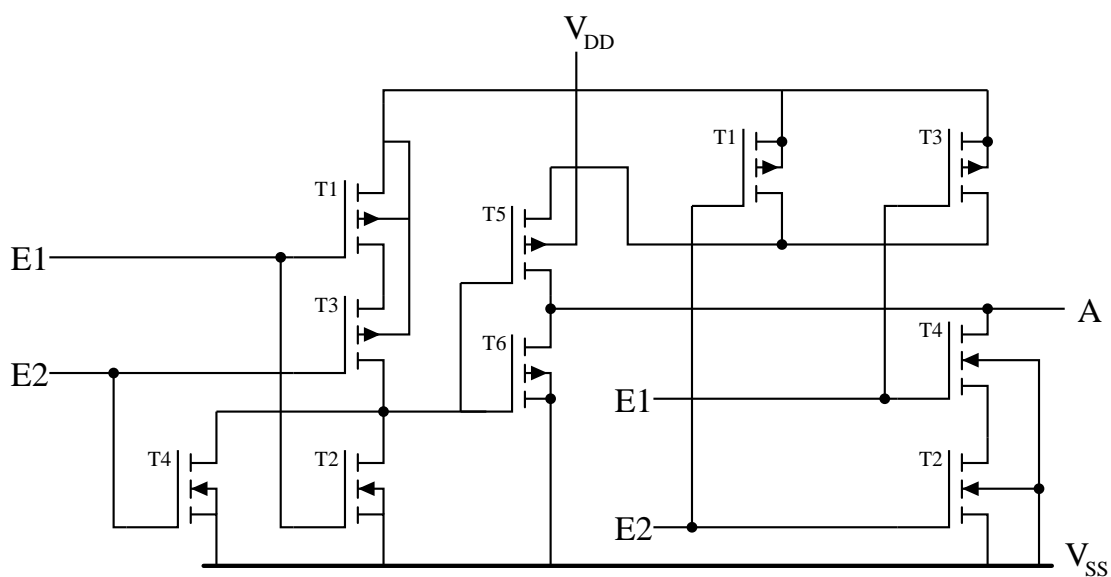


Abbildung 3: Schaltung Versuch 1.9

Versuch 1.10

Realisierung eines Exklusiv-NOR-Gatters mit zwei Eingängen.

Signalnamen	Verbindungen			
E1 = S0	FALA'	1A3	1B6	
E2 = S1	FALB'	1A6	1B3	
A = D0	FALA	1A12	1B1	
	1A2	1A14	1A11	
	1A8	1A4		
	1A13	1A1	1A5	1A10
	1B13	1B2		
	1A9	1B5	1B8	
	1B7	1B4		

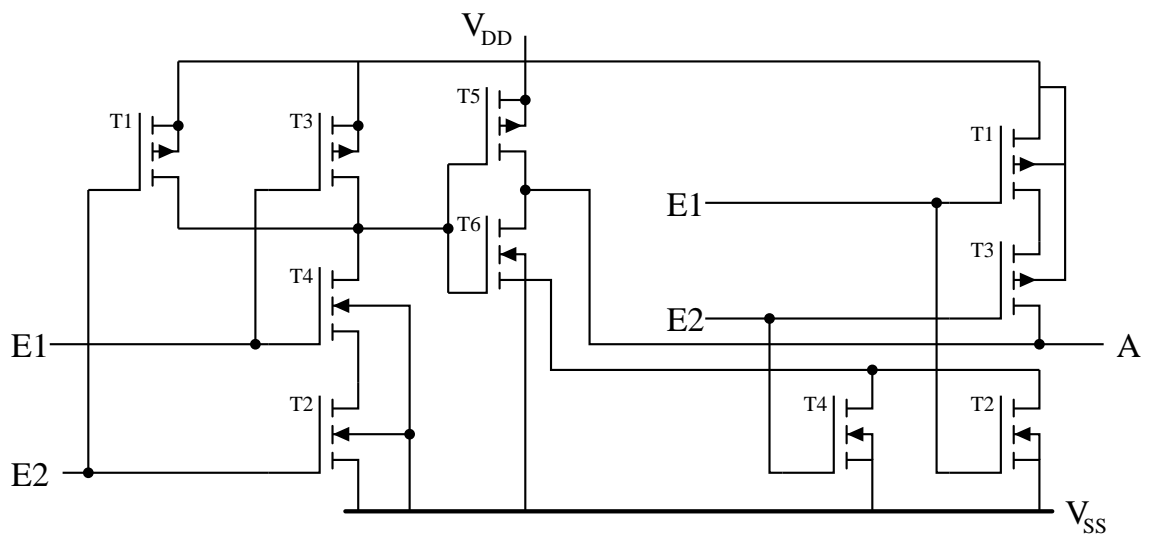


Abbildung 4: Schaltung Versuch 1.10

Versuch 2

Versuch 2.2

Realisierung eines RS-Flipflop aus NAND-Gattern mit Setz- und Rücksetzeingang.

Signalnamen	Verbindungen		
$\bar{R} = S0$	FALA'	1C1	
$\bar{S} = S1$	FALB'	1C5	
$Q = D0$	FALA	1C3	1C4
$\bar{Q} = D1$	FALB	1C6	1C2

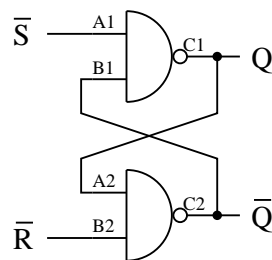


Abbildung 5: Schaltung Versuch 2.2

Versuch 2.3

Realisierung eines statisch getakteten RS-Flipflop aus NAND-Gattern.

Signalnamen	Verbindungen		
$S = S0$	FALA'	1C9	
$C = S1$	FALB'	1C10	1C13
$R = S2$	FALC'	1C12	
$Q = D0$	FALA	1C3	1C4
$\bar{Q} = D1$	FALB	1C6	1C2
	1C1	1C8	
	1C5	1C11	

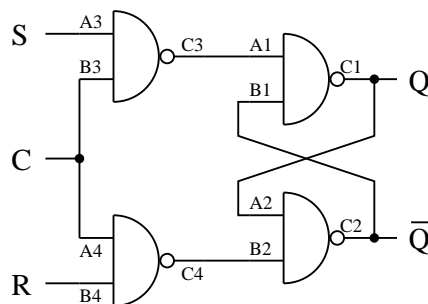


Abbildung 6: Schaltung Versuch 2.3

Versuch 2.4

Realisierung eines statisch getakteten D-Flipflops aus NAND-Gattern.

Signalnamen	Verbindungen		
C = S0	FALA'	1C10	1C13
D = S1	FALB'	1A1	1C12
	1A2	1C9	
\bar{Q} = D0	FALA	1C3	1C4
Q = D1	FALB	1C6	1C2
	1C1	1C8	
	1C5	1C11	

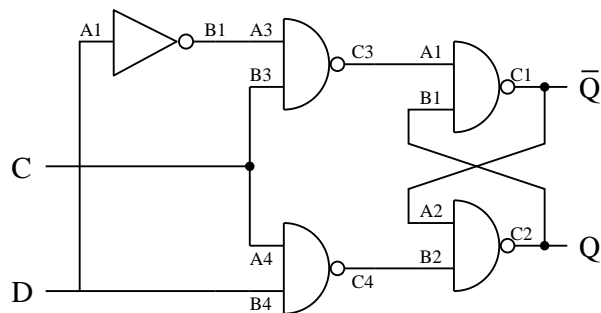


Abbildung 7: Schaltung Versuch 2.4

Versuch 2.5

Realisierung eines statisch getakteten RS-Flipflops mit statischen Setz- und Rücksetzeingängen.

Signalnamen	Verbindungen		
$\overline{\text{SET}}$ = S0	FALA'	2A11	2A4
S = S1	FALB'	2A1	
C = S2	FALC'	2A13	2A3
R = S3	FALD'	2A5	
$\overline{\text{RESET}}$ = S4	FALE'	2B13	2A2
	2A12	2A10	
	2A6	2B2	
Q = D0	FALA	2A8	2B1
\bar{Q} = D1	FALB	2B12	2A9

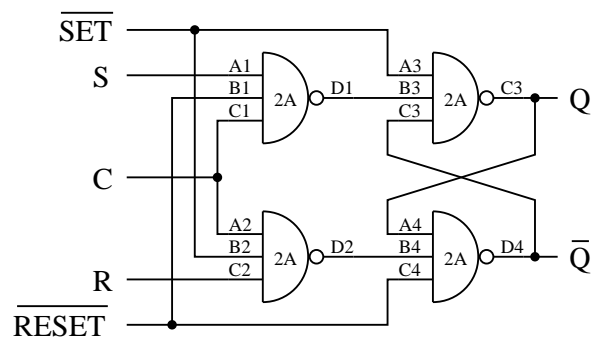


Abbildung 8: Schaltung Versuch 2.5

Versuch 2.6

Übernahme von unterschiedlichen Bytes in das Latch C573 mit der Clock C. Das Signal \overline{OE} (Output Enable) hat die Funktion, alle Ausgänge auf einen hochohmigen Widerstand zu setzen und damit die gesamte Schaltung von z.B. einem Bus zu trennen.

Signalnamen	Verbindungen		
D0 = S0	FALA'	2C2	
D1 = S1	FALB'	2C3	
⋮	⋮	⋮	⋮
D7 = S7	FALK'	2C9	
Q0 = D0	FALA	2C19	
Q1 = D1	FALB	2C18	
⋮	⋮	⋮	⋮
Q7 = D7	FALK	2C12	
$\overline{OE} = S8$	FARA'	2C1	
C = S9	FARB'	2C11	

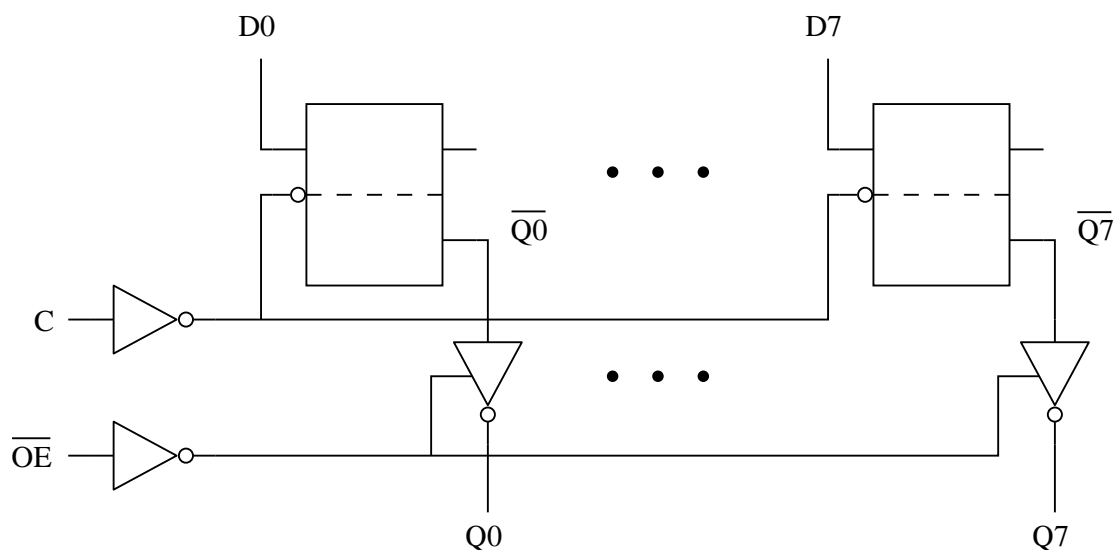


Abbildung 9: Schaltung Versuch 2.6

Versuch 2.7

Realisierung eines statisch getakteten RS-Master-Slave-Flipflops und Darstellung des Zwischenausgangs Q_M .

Signalnamen	Verbindungen			
$S = S0$	FALA'	1C1		
$C = S1$	FALB'	1C2	1C4	1A1
$R = S2$	FALC'	1C5		
	1C3	1C9		
	1C6	1C12		
$Q_M = D0$	FALA	1C8	1C13	3A1
$\overline{Q_M} = D1$	FALB	1C11	1C10	3A5
	1A2	3A2	3A4	
	3A3	3A9		
	3A6	3A12		
$Q = D2$	FALC	3A8	3A13	
$\overline{Q} = D3$	FALD	3A11	3A10	

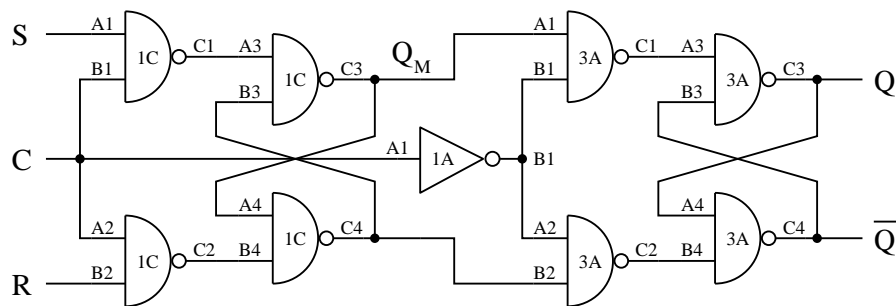


Abbildung 10: Schaltung Versuch 2.7

Versuch 2.8

Realisierung eines statisch getakteten JK-Master-Slave-Flipflops.

Signalnamen	Verbindungen			
$J = S0$	FALA'	2A2		
$C = S1$	FALB'	2A13	2A3	1A1
$K = S2$	FALC'	2A4		
	2A12	1C1		
	2A6	1C5		
$Q_M = D0$	FALA	1C3	3A1	1B4
$\overline{Q_M} = D1$	FALB	1C6	3A5	1C2
	1A2	3A2	3A4	
	3A3	3A9		
	3A6	3A12		
$Q = D2$	FALC	3A8	3A13	2A5
$\overline{Q} = D3$	FALD	3A11	3A10	2A1

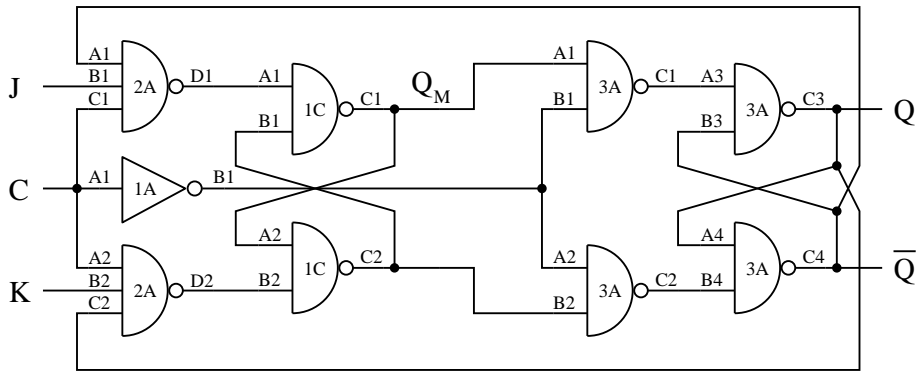


Abbildung 11: Schaltung Versuch 2.8

Versuch 2.9

Realisierung eines flankengesteuerten D-Flipflops und Darstellung der Ausgänge Q und \bar{Q} . Parallel dazu wird ein integriertes D-Flipflop des Bausteins C74 betrachtet.

Signalnamen	Verbindungen			
$\bar{S} = S0$	FALA'	3B4	1A5	3A1
D = S1	FALB'	1A1	3B2	
C = S2	FALC'	3B3	3A10	3A13
$\bar{R} = S3$	FALD'	3B1	1A11	3A5
	1A2	1A3	1C2	
	1A4	1C4		
	1C3	1C9		
	1C6	1C12		
	1C8	3A2		
	1C11	3A4		
	3A3	3A9		
	3A6	3A12		
	3A8	2A2	1C1	1C13
	3A11	2A4	1C5	1C10
	1A6	1A9		
	1A10	1A13		
	1A8	2A1		
	1A12	2A5		
Q = D0	FALA	2A12	2A3	
$\bar{Q} = D1$	FALB	2A6	2A13	
$Q_{C74} = D2$	FALC	3B5		
$\bar{Q}_{C74} = D3$	FALD	3B6		

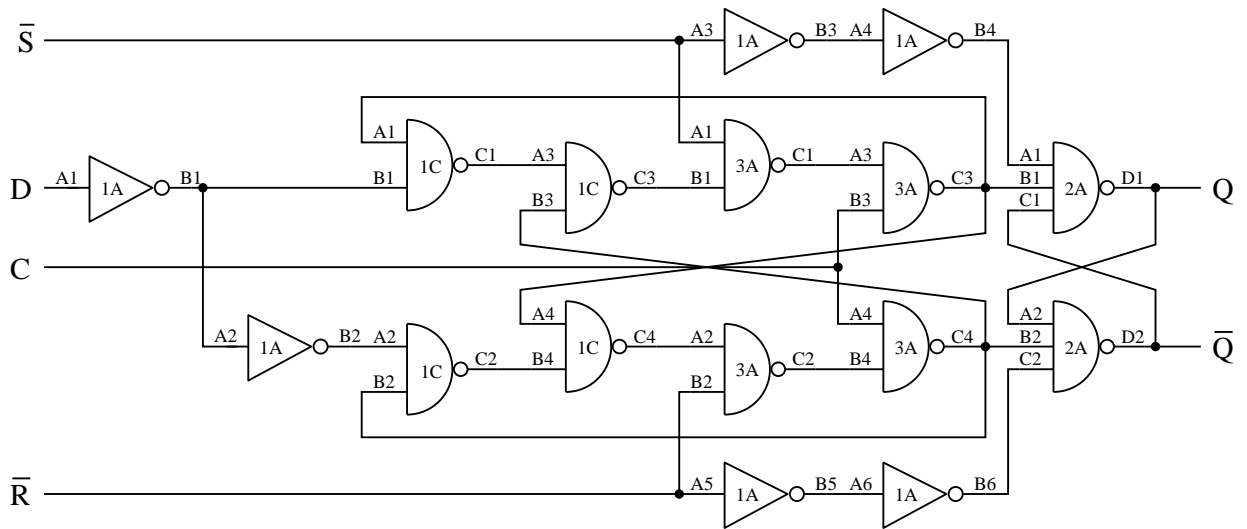


Abbildung 12: Schaltung Versuch 2.9

Versuch 2.10

Realisierung eines flankengesteuerten JK-Flipflops.

Signalnamen	Verbindungen			
$\bar{S} = S0$	FALA'	1A5	3A1	
$\bar{K} = S1$	FALB'	2B13		
$C = S2$	FALC'	3A9	3A13	
$J = S3$	FALD'	2B3		
$\bar{R} = S4$	FALE'	1A11	3A5	
	2B12	2B11		
	2B6	2B5		
	2B8	1A1	1C4	
	1A2	1C2		
	1C3	1C9		
	1C6	1C12		
	1C8	3A2		
	1C11	3A4		
	3A3	3A10		
	3A6	3A12		
	3A8	2A2	1C1	1C13
	3A11	2A4	1C5	1C9
	1A6	1A9		
	1A10	1A13		
	1A8	2A1		
	1A12	2A9		
$Q = D0$	FALA	2A12	2A3	2B1
$\bar{Q} = D1$	FALB	2A6	2A13	2B5
V_{cc}	2B14	2B2	2B4	2B10

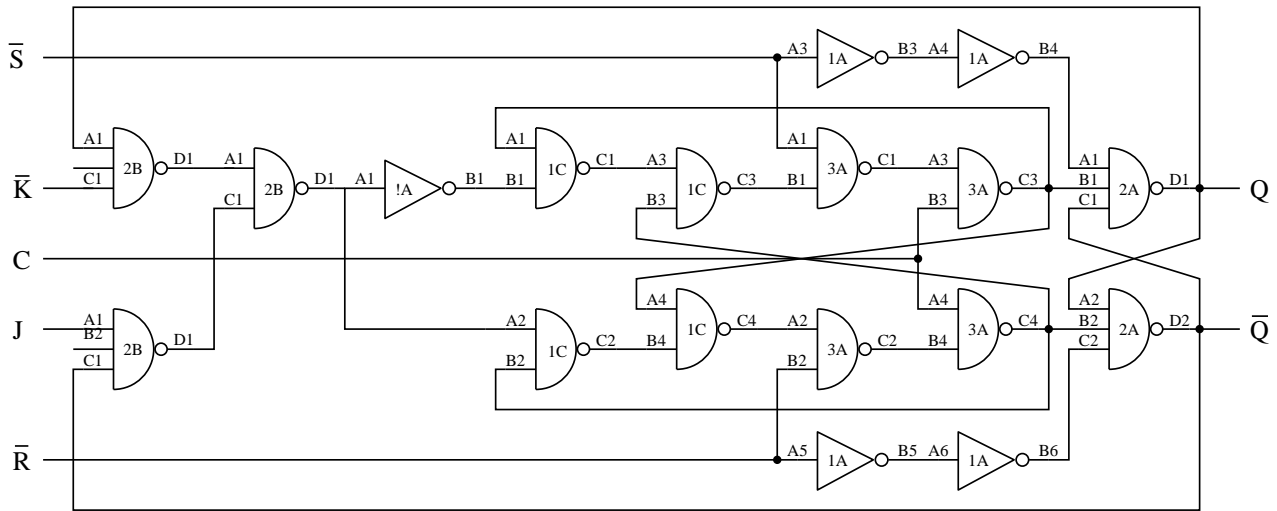


Abbildung 13: Schaltung Versuch 2.10

Versuch 2.11

Vergleich des JK-Flipflops aus 2.10 mit dem integrierten JK-Flipflop des Bausteins C109. Zusätzlich zu der Netzliste aus Versuch 2.10 müssen die folgenden Verbindungen gesteckt werden:

Signalnamen	Verbindungen		
$\bar{S} = S0$	FALA'	3C5	
$\bar{K} = S1$	FALB'	3C3	
$C = S2$	FALC'	3C4	
$J = S3$	FALD'	3C2	
$\bar{R} = S4$	FALE'	3C1	
$Q = D2$	FALC	3C6	
$\bar{Q} = D3$	FALD	3C7	

Versuch 3

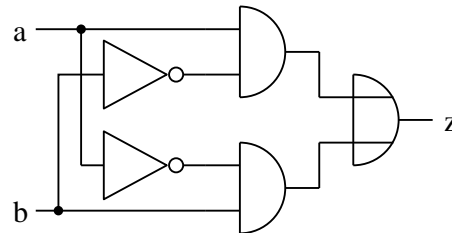
Versuch 3.1

Antivalenzschaltung:

a	b	z
0	0	0
0	1	1
1	0	1
1	1	0

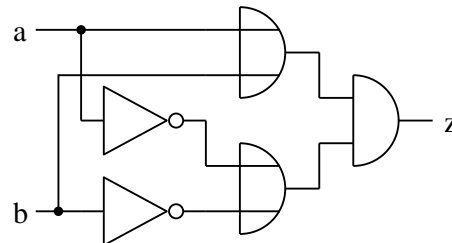
a) AND - OR

$$z = a\bar{b} + \bar{a}b$$



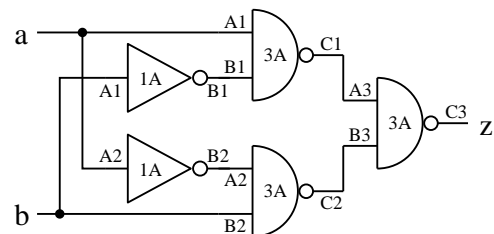
b) OR - AND

$$\begin{aligned} z &= a\bar{b} + \bar{a}b \\ &= a\bar{b} + a\bar{a} + b\bar{b} + \bar{a}b \\ &= (a+b)(\bar{a}+\bar{b}) \end{aligned}$$



c) NAND - NAND

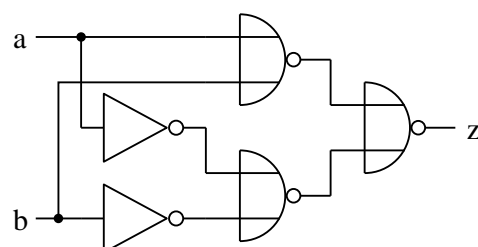
$$\begin{aligned} z &= \overline{\overline{a\bar{b}} + \overline{\bar{a}b}} \\ &= \overline{\overline{a\bar{b}} + \overline{\bar{a}b}} \\ &= \overline{\overline{a\bar{b}}} \overline{\overline{\bar{a}b}} \end{aligned}$$



Signalnamen	Verbindungen		
a = S0	FALA'	3A1	3B3
b = S1	FALB'	3A5	3B1
z = D0	FALA	3A8	
	3B2	3A2	
	3B4	3A4	
	3A3	3A9	
	3A6	3A10	

d) NOR - NOR

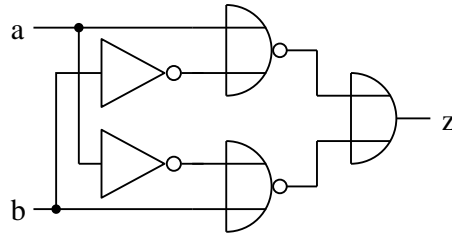
$$\begin{aligned} z &= \overline{\overline{a\bar{b}} + \overline{\bar{a}b}} \\ &= \overline{\overline{(a+b)} + \overline{(\bar{a}+\bar{b})}} \end{aligned}$$



e) NOR - OR

$$z = \overline{ab} + \overline{\overline{a}b}$$

$$= \overline{(a+b)} + \overline{(a+\overline{b})}$$



f) NAND - AND

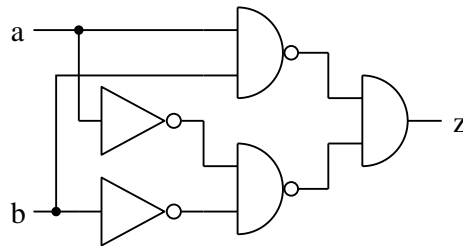
$$z = \overline{ab} + \overline{\overline{a}b}$$

$$= \overline{ab} + \overline{a\overline{b}} + \overline{b\overline{a}} + \overline{\overline{a}\overline{b}}$$

$$= \overline{(a+b)} + \overline{(a+\overline{b})}$$

$$= \overline{\overline{\overline{(a+b)}}} + \overline{\overline{\overline{(a+\overline{b})}}}$$

$$= \overline{\overline{ab}} + \overline{\overline{a\overline{b}}}$$



Versuch 3.2

Interne Logik des Decoders C138.

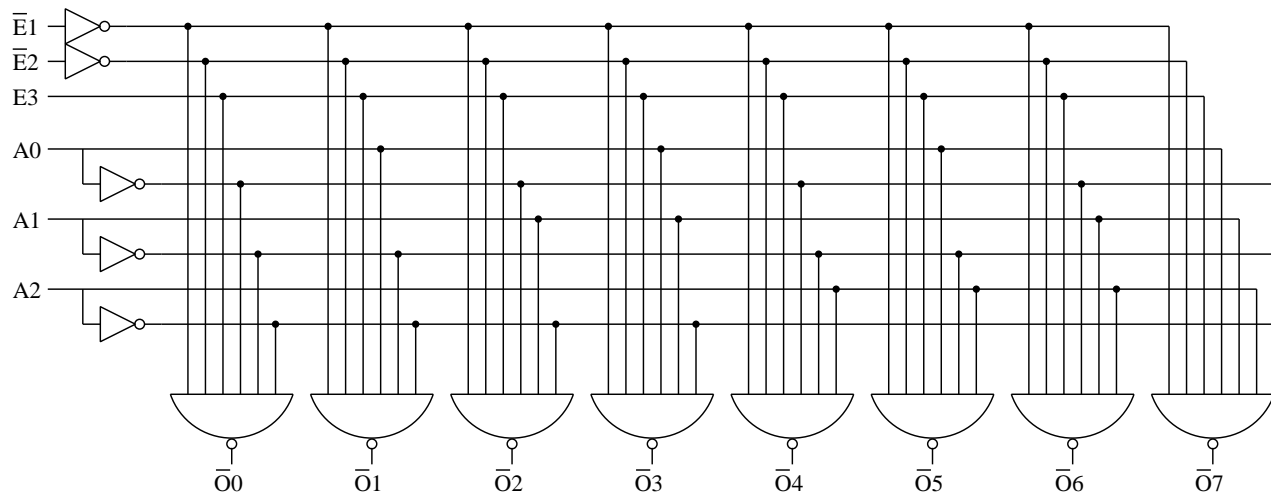


Abbildung 14: Schaltung Versuch 3.2

Versuch 3.3

Signalnamen	Verbindungen			
a = S0	FALA'	1C1	3B1	2A3
b = S1	FALB'	1C2	3B3	
c = S2	FALC'	1C3	3B5	
d = S3	FALD'	1C4	3A2	3A5
y0 = D0	FALA	1C15		
y1 = D1	FALB	1C14		
y2 = D2	FALC	1C13		
y3 = D3	FALD	1C12		
y4 = D4	FALG	1C11		
y5 = D5	FALH	1C10		
y6 = D6	FALJ	1C9		
y7 = D7	FALK	1C7		
y8 = D8	FARA	3A3		
y9 = D9	FARB	3A6		
	1C5	1C8		
	1C6	1C16		
	3B2	2A1		
	3B4	2A2	2A4	
	3B6	2A13	2A5	
	2A12	3B9		
	3B8	3A1		
	2A6	3B11		
	3B10	3A4		

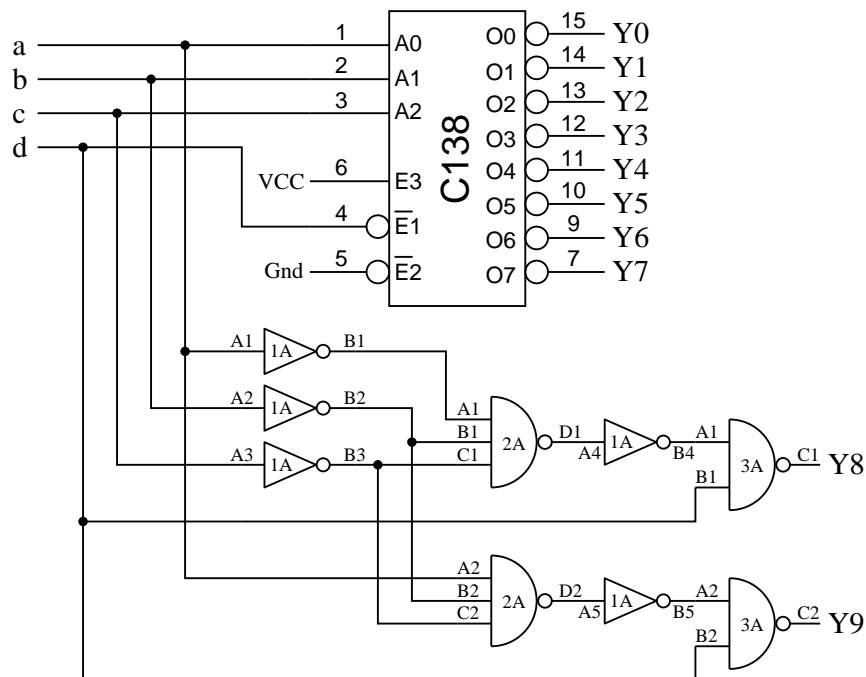


Abbildung 15: Schaltung Versuch 3.3

Versuch 3.4

Netzliste von Versuch 3.4 und 3.6. (Die durch doppelstriche abgetrennten Zeilen / Spalten gehören nur zu Versuch 3.6)

Signalnamen	Verbindungen		
X0 = S0	FALA'	2B11	
X1 = S1	FALB'	2B10	
X2 = S2	FALC'	2B9	
	2B8	2B7	
A0 = S8	FARA'	2B4	2C2
A1 = S9	FARB'	2B3	2C4
A2 = S10	FARC'	2B2	2C6
A3 = S11	FARD'	2B1	2C8
A4 = S12	FARG'	2B15	2C11
A5 = S13	FARH'	2B14	2C13
A6 = S14	FARJ'	2B13	2C15
A7 = S15	FARK'	2B12	2C17
	2B5	1C4	
	1C5	1C8	
	1C6	1C16	
Y0 = S3	FALD'	1C1	
Y1 = S4	FALG'	1C2	
Y2 = S5	FALH'	1C3	
B0 = $\overline{O0}$	FALA	1C15	2C3
B1 = $\overline{O1}$	FALB	1C14	2C5
B2 = $\overline{O2}$	FALC	1C13	2C7
B3 = $\overline{O3}$	FALD	1C12	2C9
B4 = $\overline{O4}$	FALG	1C11	2C12
B5 = $\overline{O5}$	FALH	1C10	2C14
B6 = $\overline{O6}$	FALJ	1C9	2C16
B7 = $\overline{O7}$	FALK	1C7	2C18
$\overline{(A = B)}$	FARA	2C19	
	2C1	2C10	

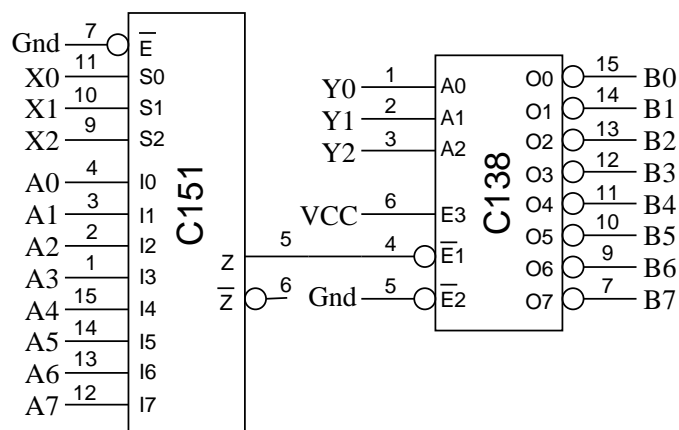


Abbildung 16: Schaltung Versuch 3.4

Versuch 3.5

Interne Logik des Komparators C520.

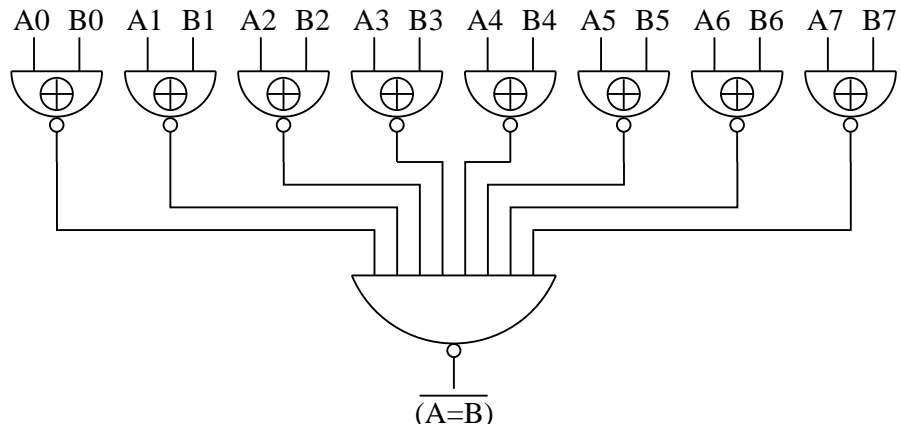


Abbildung 17: Schaltung Versuch 3.5

Versuch 3.6

Netzliste siehe Versuch 3.4.

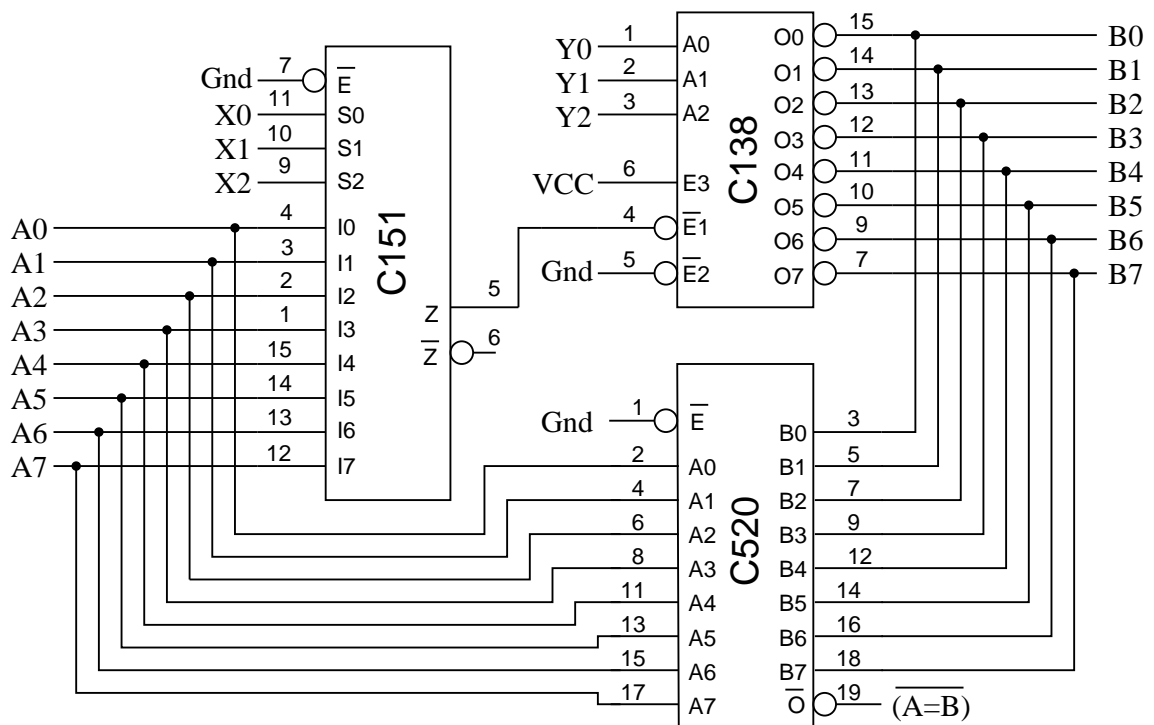


Abbildung 18: Schaltung Versuch 3.6

Versuch 4

Versuch 4.1

Realisierung eines 4-Bit-Seriell-Parallel-Schieberegisters mit Hilfe des Bausteins C74.

Signalnamen	Verbindungen				
$S_{in} = S0$	FALA'	1B2			
$\overline{\text{Reset}} = S1$	FALB'	1B1	1B13	2B1	2B13
$C = S2$	FALC'	1B3	1B11	2B3	2B11
$Q0 = D0$	FALA	1B5	1B12		
$Q1 = D1$	FALB	1B9	2B2		
$Q2 = D2$	FALC	2B5	2B12		
$Q3 = D3$	FALD	2B9			

Versuch 4.2

Realisierung eines 4-Bit-Parallel-Seriell-Schieberegisters mit Hilfe des Bausteins C74.

Signalnamen	Verbindungen				
$P0 = S0$	FALA'	1B4			
$P1 = S1$	FALB'	1B10			
$P2 = S2$	FALC'	2B4			
$P3 = S3$	FALD'	2B10			
$\overline{\text{Reset}} = S4$	FALH'	1B1	1B13	2B1	2B13
$C = S5$	FALH'	1B3	1B11	2B3	2B11
$S_{out} = D0$	FALA	2B9	1B2		
	1B5	1B12			
	1B9	2B2			
	2B5	2B12			

Versuch 4.3

Realisierung eines Zählers mit Hilfe des Bausteins C299. Durch das Steuersignal S ist kann der Zähler als Ringzähler ($S = 0$) oder Johnsonzähler ($S = 1$) arbeiten.

Signalnamen	Verbindungen		
S = S0	FALA'	3A1	2A4
	2C17	3A3	2A2
	3A2	2A1	
	3A4	2A5	
	2A3	1A1	
	2A6	1A2	
	1A3	2C11	
DS7 = S1	FALB'	2C18	
S0 = S2	FALC'	2C1	
S1 = S3	FALD'	2C19	
C = S4	FALG'	2C12	
	2C20	2C2	2C3
$\overline{MR} = S5$	FALH'	2C9	
I/O0 = D0	FALA	2C7	
I/O1 = D1	FALB	2C13	
I/O2 = D2	FALC	2C6	
I/O3 = D3	FALD	2C14	
I/O4 = D4	FALG	2C5	
I/O5 = D5	FALH	2C15	
I/O6 = D6	FALJ	2C4	
I/O7 = D7	FALK	2C16	

Versuch 4.4

Realisierung einer Übertragungsstrecke mit Hilfe des Bausteins C299, auf der Daten seriell in eine Richtung übertragen werden können. Die Datenworte werden parallel vorgegeben und parallel entnommen.

Signalnamen	Verbindungen			
I/O0 _I = S0	FALA'	2C7		
I/O1 _I = S1	FALB'	2C13		
I/O2 _I = S2	FALC'	2C6		
I/O3 _I = S3	FALD'	2C14		
I/O4 _I = S4	FALG'	2C5		
I/O5 _I = S5	FALH'	2C15		
I/O6 _I = S6	FALJ'	2C4		
I/O7 _I = S7	FALK'	2C16		
	2C1	2C10	2C2	2C3
	2C19	2C20		
C = S8	FARA'	2C12	4C12	
\overline{MR} = S9	FARB'	2C9	4C9	
	4C1	4C10		
	4C19	4C20	4C2	4C3
Q7 _I = DS0 _O	2C17	4C11		
I/O0 _O = D0	FALA	4C7		
I/O1 _O = D1	FALB	4C13		
I/O2 _O = D2	FALC	4C6		
I/O3 _O = D3	FALD	4C14		
I/O4 _O = D4	FALG	4C5		
I/O5 _O = D5	FALH	4C15		
I/O6 _O = D6	FALJ	4C4		
I/O7 _O = D7	FALK	4C16		

Versuch 4.5

Realisierung eines 3-Bit asynchronen Binärzählers unter Einsatz von JK-Flipflops (Baustein C109). Jede Zahl wird mit Hilfe des Bausteins C138 dekodiert.

Signalnamen	Verbindungen			
C = S0	FALA'	3A1		
	3A2	3B4		
Reset = S1	FALB'	3B1	3B15	4B1
V _{CC}	3B16	3B5	3B11	4B5
V _{CC}	3B16	3B2	3B14	4B2
Gnd	–	3B3	3B13	4B3
Q0	3B6	4A1	3A3	
	3A4	3B12		
Q1	3B10	4A2	3A5	
	3A6	4B4		
Q2	4B6	4A3		
$\bar{E}1 = \bar{E}2$	4A8	4A4	4A5	
E3	4A16	4A6		
$\bar{O}0 = D0$	FALA	4A15		
$\bar{O}1 = D1$	FALB	4A14		
$\bar{O}2 = D2$	FALC	4A13		
$\bar{O}3 = D3$	FALD	4A12		
$\bar{O}4 = D4$	FALG	4A11		
$\bar{O}5 = D5$	FALH	4A10		
$\bar{O}6 = D6$	FALJ	4A9		
$\bar{O}7 = D7$	FALK	4A7		

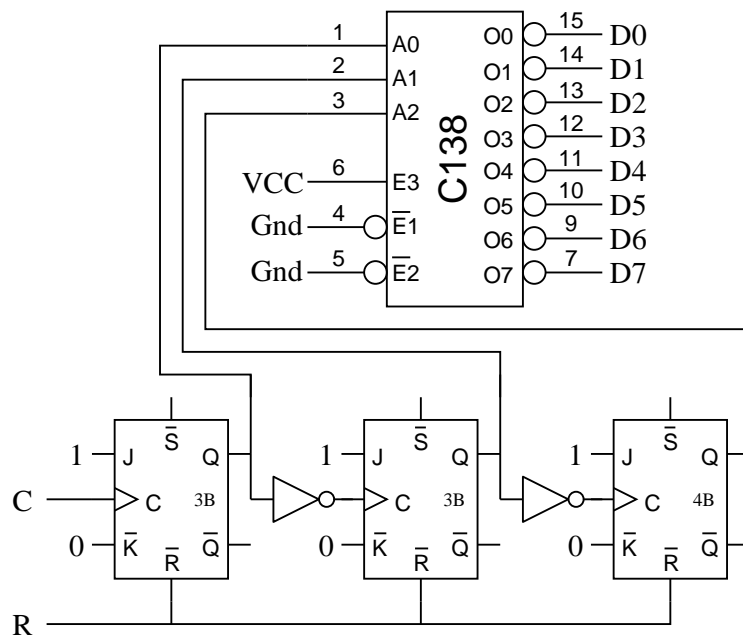


Abbildung 19: Schaltung Versuch 4.5

Versuch 4.6

Entwicklung eines synchronen Modulo-5-Zählers mit JK-Flipflops.

Signalnamen	Verbindungen			
$\bar{S} = S0$	FALA'	3B5	3B11	4B5
$\bar{R} = S1$	FALB'	3B1	3B15	4B1
$C = S2$	FALC'	3B4	3B12	4B4
$Q0 = D0$	FALA	3B6	2A4	2A12
$\bar{O}0$	3B7	2A9	2A2	
$Q1 = D1$	FALB	3B10	2A13	2A10
$\bar{O}1$	3B9	2A5		
$Q2 = D2$	FALC	4B6		
$\bar{O}2$	4B7	2A1		
$J0 = \bar{K}0$	2A3	3B2	3B3	
	2A6	1A1		
	2A8	1A2		
$J1 = \bar{K}1$	1A3	3B14	3B13	
$J2 = \bar{K}2$	2A11	4B2	4B3	

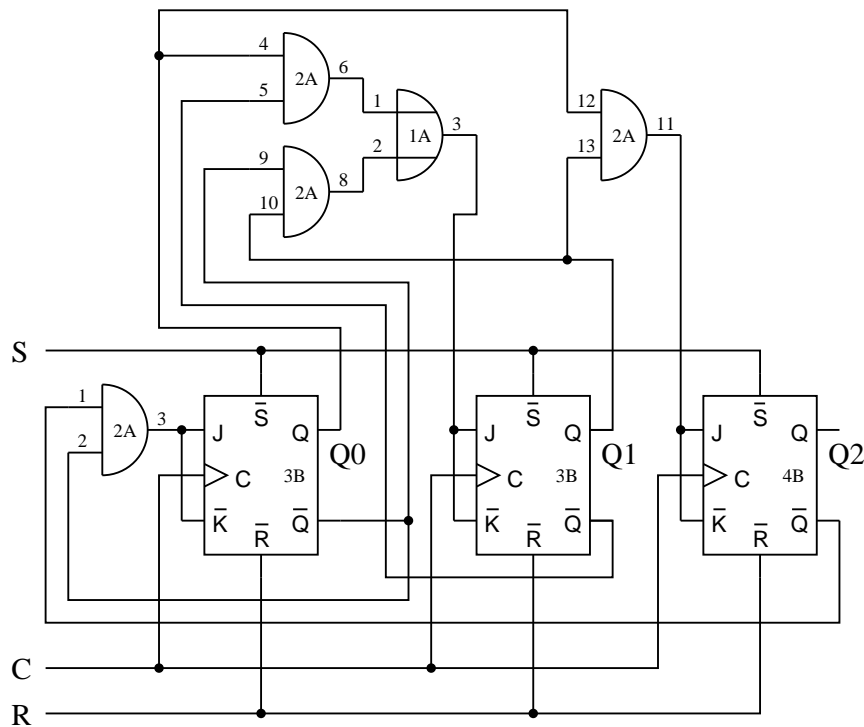


Abbildung 20: Schaltung Versuch 4.6

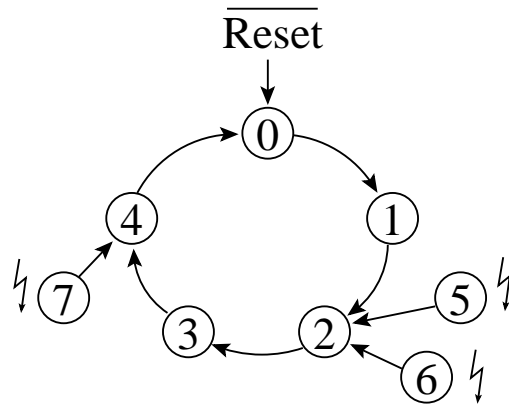


Abbildung 21: Schaltwerk-Graph des Modulu-5-Zählers

Versuch 5

Teil a)

Realisierung eines 7-Segment-Decoders als PAL.

Dez	DA	DB	DC	DD	A	B	C	D	E	F	G
0	0	0	0	0	0	0	0	0	0	0	1
1	1	0	0	0	1	0	0	1	1	1	1
2	0	1	0	0	0	0	1	0	0	1	0
3	1	1	0	0	0	0	0	0	1	1	0
4	0	0	1	0	1	0	0	1	1	0	0
5	1	0	1	0	0	1	0	0	1	0	0
6	0	1	1	0	0	1	0	0	0	0	0
7	1	1	1	0	0	0	0	1	1	1	1
8	0	0	0	1	0	0	0	0	0	0	0
9	1	0	0	1	0	0	0	0	1	0	0

$SG0 = 1, SG1 = 0, SL0 = 0$

Daraus abgeleitete Bedingungsgleichungen für die Segmentvariablen A - G:

$$A = (DA * \overline{DB} * \overline{DC} * \overline{DD}) + (\overline{DA} * \overline{DB} * DC * \overline{DD})$$

$$B = (DA * \overline{DB} * DC * \overline{DD}) + (\overline{DA} * DB * DC * \overline{DD})$$

$$C = (\overline{DA} * DB * \overline{DC} * \overline{DD})$$

$$D = (DA * \overline{DB} * \overline{DC} * \overline{DD}) + (\overline{DA} * \overline{DB} * DC * \overline{DD}) \\ = + (DA * DB * DC * \overline{DD})$$

$$E = (DA * \overline{DB} * \overline{DC} * \overline{DD}) + (DA * DB * \overline{DC} * \overline{DD}) \\ + (\overline{DA} * \overline{DB} * DC * \overline{DD}) + (DA * \overline{DB} * DC * \overline{DD}) \\ + (DA * \overline{DB} * \overline{DC} * \overline{DD}) + (DA * DB * DC * \overline{DD})$$

$$F = (DA * \overline{DB} * \overline{DC} * \overline{DD}) + (\overline{DA} * DB * \overline{DC} * \overline{DD}) \\ + (DA * DB * \overline{DC} * \overline{DD}) + (DA * DB * DC * \overline{DD})$$

$$G = (\overline{DA} * \overline{DB} * \overline{DC} * \overline{DD}) + (DA * \overline{DB} * \overline{DC} * \overline{DD}) \\ = + (DA * DB * DC * \overline{DD})$$

Teil b)

Die Aufnahme des Kurvenverlaufs erfolgt folgendermaßen: Die eingestellte Spannung U_p wird gemessen und abgelesen. Der Zählerwert wird abgelesen. Die beiden Werte werden gegeneinander aufgetragen, Zählerwert auf der X-Achse, U_p auf der Y-Achse. Das so entstehende Diagramm gibt den Kurvenverlauf der erzeugten Spannung wieder.

Weil der Zähler höchstens bis 99 zählen kann, maximal also 99 Takte erfolgen dürfen, darf das Signal ENP nicht länger als $99 \cdot T = 99 \cdot 44\mu s = 4,356ms$ auf 1 liegen. Der Wert von U_p muss so eingestellt werden, dass ENP diese Bedingung erfüllt.

Beim Entwurf des Dezimalzählers ist zu beachten, dass dieser abhängig vom Signal ENP entweder zählen soll ($ENP = 1$), oder den bis dahin erreichten Wert halten soll ($ENP = 0$).

Teil c)

Realisierung eines 4-Bit-Dezimalzählers als PAL.

Für $\text{/LOAD} = 0$ ergibt sich die folgende Tabelle, für $\text{/LOAD} = 1$ müssen alle Ausgänge auf 0 gelegt werden.

ENP	DA	DB	DC	DD	RC0	QA	QB	QC	QD
1	0	0	0	0	0	1	0	0	0
1	1	0	0	0	0	0	1	0	0
1	0	1	0	0	0	1	1	0	0
1	1	1	0	0	0	0	0	1	0
1	0	0	1	0	0	1	0	1	0
1	1	0	1	0	0	0	1	1	0
1	0	1	1	0	0	1	1	1	0
1	1	1	1	0	0	0	0	0	1
1	0	0	0	1	0	1	0	0	1
1	1	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	0	0	0
0	0	1	0	0	0	0	1	0	0
0	1	1	0	0	0	1	1	0	0
0	0	0	1	0	0	0	0	1	0
0	1	0	1	0	0	1	0	1	0
0	0	1	1	0	0	0	1	1	0
0	1	1	1	0	0	1	1	1	0
0	0	0	0	1	0	0	0	0	1
0	1	0	0	1	0	1	0	0	1

$\text{SG0} = 0, \text{SG1} = 1, \text{SL0} = 0$

$$\begin{aligned}
 \text{RC0} &= (\text{ENP} * \text{DA} * \text{/DB} * \text{/DC} * \text{DD} * \text{LOAD}) \\
 \text{QA} &= (\text{ENP} * \text{/DA} * \text{LOAD}) + (\text{/ENP} * \text{DA} * \text{LOAD}) \\
 \text{QB} &= (\text{ENP} * \text{DA} * \text{/DB} * \text{/DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{ENP} * \text{/DA} * \text{DB} * \text{/DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{ENP} * \text{DA} * \text{/DB} * \text{DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{ENP} * \text{/DA} * \text{DB} * \text{DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{/ENP} * \text{/DA} * \text{DB} * \text{/DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{/ENP} * \text{DA} * \text{DB} * \text{/DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{/ENP} * \text{/DA} * \text{DB} * \text{DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{/ENP} * \text{DA} * \text{DB} * \text{DC} * \text{/DD} * \text{LOAD}) \\
 \text{QC} &= (\text{ENP} * \text{DA} * \text{DB} * \text{/DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{ENP} * \text{/DA} * \text{/DB} * \text{DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{ENP} * \text{DA} * \text{/DB} * \text{DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{ENP} * \text{/DA} * \text{DB} * \text{DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{/ENP} * \text{/DA} * \text{/DB} * \text{DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{/ENP} * \text{DA} * \text{/DB} * \text{DC} * \text{/DD} * \text{LOAD}) \\
 &\quad + (\text{/ENP} * \text{/DA} * \text{DB} * \text{DC} * \text{/DD} * \text{LOAD})
 \end{aligned}$$

$$\begin{aligned} & + (/ENP * DA * DB * DC * /DD * LOAD) \\ QD = & (ENP * DA * DB * DC * /DD * LOAD) \\ & + (ENP * /DA * /DB * /DC * DD * LOAD) \\ & + (/ENP * /DA * /DB * /DC * DD * LOAD) \\ & + (/ENP * DA * /DB * /DC * DD * LOAD) \end{aligned}$$

Versuch 6

1) Zustände der Aufzugssteuerung

Zustand	Zi	Q1	Q2	Q3
Auf	Z0	0	0	0
Etage0	Z1	0	0	1
Etage1	Z2	0	1	0
Etage2	Z3	0	1	1
Ab	Z4	1	0	0
WaitAuf	Z5	1	0	1
WaitAb	Z6	1	1	0

2) Zustandsgraph

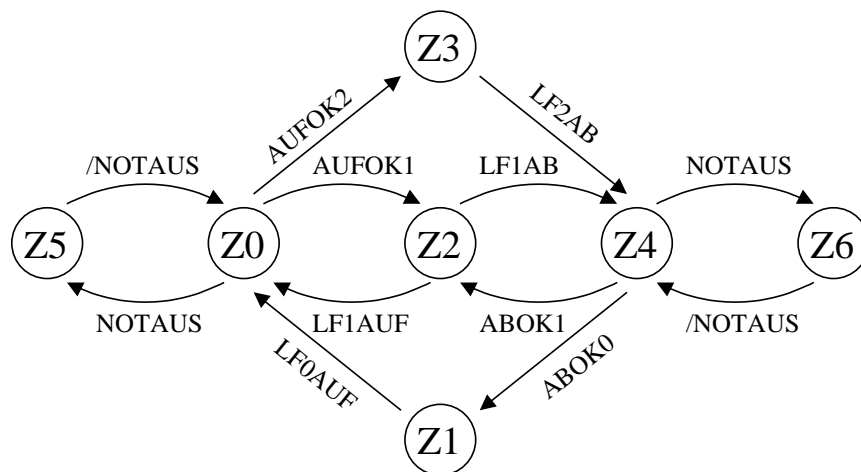


Abbildung 22: Zustandsgraph der Aufzugssteuerung

3) Programmierung

Pin 17 - 19 REGISTERED

```
; State Machine Segment:  
STATE  
MOORE_MACHINE  
DEFAULT_BRANCH HOLD_STATE  
START_UP := POWER_UP -> Auf
```

```
;Zustaende:
```

```
Auf =      /Q0 * /Q1 * /Q2  
Etage0 =   Q0 * /Q1 * /Q2  
Etage1 =  /Q0 * Q1 * /Q2  
Etage2 =   Q0 * Q1 * /Q2  
Ab =       /Q0 * /Q1 * Q2  
WaitAuf =  Q0 * /Q1 * Q2  
WaitAb =   /Q0 * Q1 * Q2
```

```
;Zustandswechsel
```

```
;(Aktueller Zustand Bedingung nächster Zustand)
```

```

Etag0 := LF0AUF -> Auf +-> Etag0;
Etag1 := LF1AUF -> Auf + LF1AB -> Ab +-> Etag1;
Etag2 := LF2AB -> AB +-> Etag2;
Auf := NOTAUS -> WaitAuf + AUFOK1->Etag1 + AUFOK2-> Etag2 +-> Auf;
Ab := NOTAUS -> WaitAb + ABOK1-> Etag1 +ABOK0->Etag0;
WaitAuf:= /NOTAUS -> Auf +-> WaitAuf;
WaitAb := /NOTAUS -> Ab +-> WaitAb;

```

```

; (Ausgabefunktion)

```

```

Etag0.OUTF = /MOTAUF * /MOTAB;
Etag1.OUTF = /MOTAUF * /MOTAB;
Etag2.OUTF = /MOTAUF * /MOTAB;
Auf.OUTF = MOTAUF * /MOTAB;
Ab.OUTF = /MOTAUF * MOTAB;
WaitAuf.OUTF = /MOTAUF * /MOTAB;
WaitAb.OUTF = /MOTAUF * /MOTAB;

```

```

;---- Condition Segment

```

```

CONDITIONS
LF0AUF = (ANFOR1 + ANFOR2) * /NOTAUS * /TIME;
LF1AUF = ANFOR2 * /NOTAUS * /TIME;
LF1AB = /ANFOR2 * ANFOR0 * /NOTAUS * /TIME;
LF2AB = (ANFOR0 + ANFOR1) * /NOTAUS * /TIME;
AUFOK1 = /ETAG0 * ETAG1 * /ETAG2 * /NOTAUS * ANFOR1;
AUFOK2 = /ETAG0 * /ETAG1 * ETAG2 * /NOTAUS;
ABOK1 = /ETAG0 * ETAG1 * /ETAG2 * /NOTAUS * ANFOR1;
ABOK0 = ETAG0 * /ETAG1 * /ETAG2 * /NOTAUS;

```

```

; (bei Etag1 also nur bei Anforderung stoppen)

```

```

;---- Boolean Equation Segment

```

```

EQUATIONS
STOP0 = Etag0;
STOP1 = Etag1 * ANFOR1;
STOP2 = Etag2;

```

```

;---- Simulation

```

```

SIMULATION
TRACE_ON Q0 Q1 Q2 MOTAUF MOTAB
Setf ANFOR2
Check Q0
Check Q1
Check Q2
TRACE_OFF

```