

V003: Digitale Bauelemente

Ziel: Es soll die Arbeitsweise von logischen Bauelementen kennengelernt werden.

Stichworte: BCD-Kode, Binärsystem, Binäre Verknüpfungen, Boolesche Algebra, Flip-Flop, Gatter, Impulsdiagramm, Kippstufe, Logik-Pegel, Negation, positive Logik, Quarzoszillator, Wahrheitstabelle, Zähler

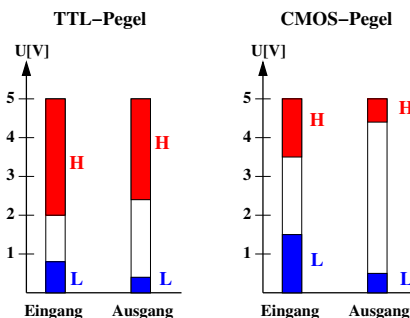
Theoretische Grundlagen

In *logischen Schaltungen* werden Eingangs- und Ausgangssignale logisch verknüpft. Die Verknüpfungen werden durch verschiedene Gatter realisiert, die auf den Aussagen 'wahr' oder 'falsch' beruhen. So wandelt z.B. die *Logische Verknüpfung* 'NICHT' eine richtige

Aussage in eine falsche Aussage um und umgekehrt. Bei der technischen Realisierung werden Signale verwendet, die auf zwei Zuständen beruhen, der '1' (wahr) und der '0' (nicht wahr), sodaß für jede Verknüpfung eine *Wahrheitstabelle* für die **E**ingangs- und **A**usgangssignale erstellt werden kann. Die nebenstehenden Wahrheitstabelle gibt für die Negation (NICHT) die möglichen Zustände an.

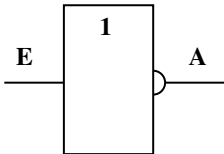
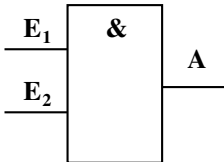
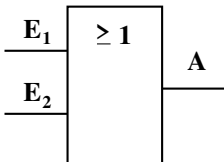
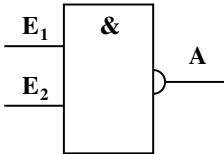
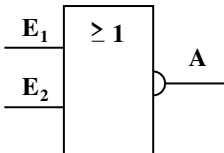
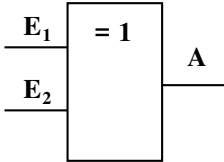
E	A
0	1
1	0

Alle *digitalen Schaltungen* basieren auf Transistorschaltungen, die beim Schalten auf Spannungen und Ströme reagieren. Je nachdem mit welcher Halbleitertechnologie der Transistor aufgebaut ist (z.B. TTL oder CMOS) reagiert das Gatter auf unterschiedliche Spannungswerte. In der nebenstehenden Graphik werden die verschiedenen Signalpegel der TTL- und CMOS-Technik veranschaulicht. Es wird zwischen hohen (High-Level) und niedrigen (Low-Level) Spannungspegel unterschieden. (Die binären Pegel L und H dürfen dabei nicht mit den logischen Zuständen 0 und 1 verwechselt werden. So wird eine *positive Logik* definiert, bei der L einer 0 und H einer 1 entspricht. Bei einer *negativen Logik* entspricht dagegen L einer 1 und H einer 0.)

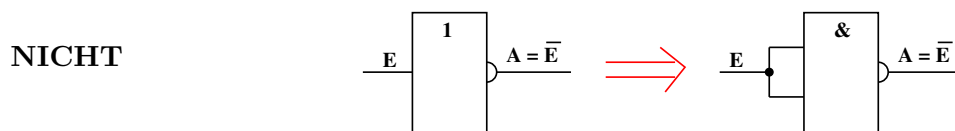


Logische Aussagen können durch Operationen wie zum Beispiel der *Konjunktion* (UND), der *Disjunktion* (ODER) und der *Negation* (NICHT) realisiert werden. Um eine Schaltung mit logischen Bauteilen aufzubauen, müssen zuerst die Variablen und Funktion der Schaltung aufgestellt werden. Mit Hilfe der Wahrheitstabelle kann eine logische Verknüpfungsschaltung beschrieben werden. Aus den Wahrheitstabellen lassen sich dann mit Hilfe der *Booleschen Algebra* digitale Schaltungen berechnen und vereinfachen. Auf eine Einführung in die Boolesche Algebra wird an dieser Stelle verzichtet, da in diesem Praktikumsversuch nur die Funktionsweisen der wichtigsten digitalen Bauelemente kennengelernt werden sollen. Für Interessierte wird z.B. das Buch *Schenk, Kremer Hrsg: Das Physikalische Praktikum* empfohlen.

Im Folgenden werden die wichtigsten Schaltsymbole mit ihren Wahrheitstabellen aufgeführt. In der rechten Spalte stehen für Interessierte die jeweiligen Regeln der Schaltalgebra.

NICHT		<table><tr><th>E</th><th>A</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	E	A	0	1	1	0	$A = \overline{E}$									
E	A																	
0	1																	
1	0																	
UND		<table><tr><th>E1</th><th>E2</th><th>A</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	E1	E2	A	0	0	0	1	0	0	0	1	0	1	1	1	$A = E_1 \wedge E_2$
E1	E2	A																
0	0	0																
1	0	0																
0	1	0																
1	1	1																
ODER		<table><tr><th>E1</th><th>E2</th><th>A</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	E1	E2	A	0	0	0	1	0	1	0	1	1	1	1	1	$A = E_1 \vee E_2$
E1	E2	A																
0	0	0																
1	0	1																
0	1	1																
1	1	1																
NAND		<table><tr><th>E1</th><th>E2</th><th>A</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	E1	E2	A	0	0	1	1	0	1	0	1	1	1	1	0	$A = \overline{E_1 \wedge E_2}$
E1	E2	A																
0	0	1																
1	0	1																
0	1	1																
1	1	0																
NOR		<table><tr><th>E1</th><th>E2</th><th>A</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	E1	E2	A	0	0	1	1	0	0	0	1	0	1	1	0	$A = \overline{E_1 \vee E_2}$
E1	E2	A																
0	0	1																
1	0	0																
0	1	0																
1	1	0																
XOR		<table><tr><th>E1</th><th>E2</th><th>A</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	E1	E2	A	0	0	0	1	0	1	0	1	1	1	1	0	$A = (E_1 \wedge \overline{E_2}) \vee (\overline{E_1} \wedge E_2)$
E1	E2	A																
0	0	0																
1	0	1																
0	1	1																
1	1	0																

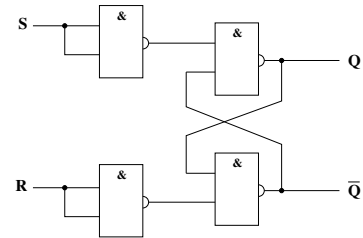
Alle logischen Verknüpfungen lassen sich auf die Grundverknüpfungen UND-, ODER, und NICHT zurückführen. Diese lassen sich durch eine geeignete Kombination von NAND- oder NOR-Gattern realisiert. So zeigt die Abbildung unten eine NICHT-Schaltung, die mit einem NAND-Gatter aufgebaut wurde.



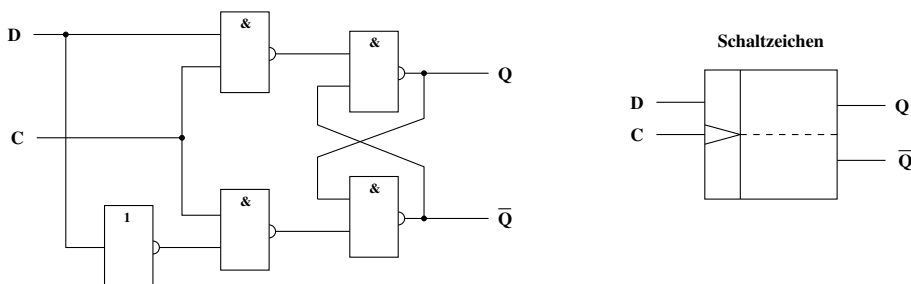
Bei *digitalen Schaltungen* wird zwischen *statischen Schaltungen* und *zeitabhängigen Schaltungen* unterschieden. Bei den statischen Schaltungen definieren die Eingangszustände der logi-

schen Verknüpfungen eindeutig die Ausgangszustände. Die Zeit, wann ein Zustand gesetzt wird, spielt in diesen Schaltungen keine Rolle. Dies ist anders bei den *zeitabhängigen Schaltungen*. Bei diesen Schaltungen reagiert der Ausgangszustand auf eine *Zustandsänderung am Eingang*. Der Signalzustand hängt in diesem Fall nicht nur von den Eingangszuständen sondern auch von der 'Vorgeschichte' und damit von der Zeit ab. Eine der einfachsten *zeitabhängigen Schaltungen* ist das *RS-Flip-Flop*.

In der nebenstehenden Abbildung ist ein *Flip-Flop*, bestehend aus vier NAND-Gattern dargestellt. Ein *Flip-Flop* besitzt zwei stabile Ausgangszustände (bistabile Kippstufe). Der Schaltzustand eines Flip-Flops richtet sich also nach der Abfolge der Signale. Die Eingänge übernehmen dabei die Aufgabe des **S**etzen und **R**ücksetzen (Löschen) der Einheit. An den Ausgängen werden komplementäre Zustände gesetzt. Wenn zum Beispiel der Ausgang $Q = 1$ gesetzt ist, dann ist $\bar{Q} = 0$. Dieser Zustand bleibt erhalten bis am *R-Eingang* eine 1 gesetzt wird. Diese einfachen Speichereinheiten werden *RS-Flip-Flop* genannt.

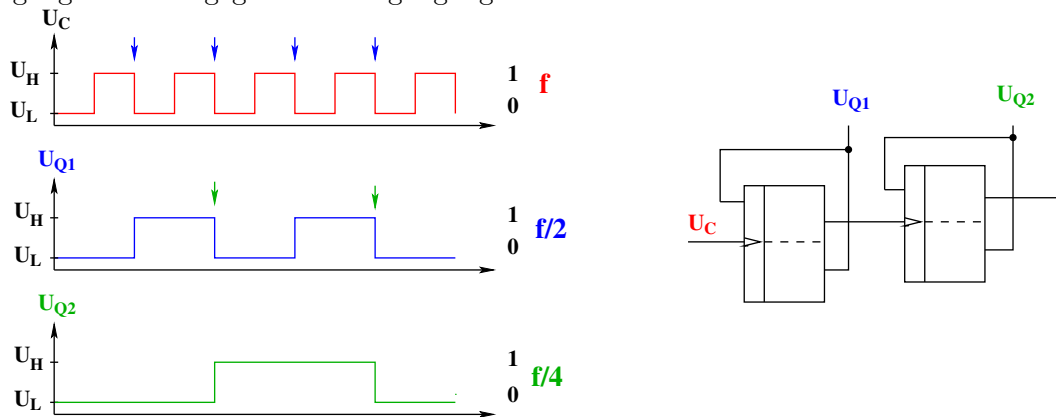


Die Kombination $S = R = 1$ ist 'verboten', da die Kippstufe nicht gleichzeitig gesetzt und rückgesetzt werden kann. Das Flip-Flop befindet sich dann in einem nicht definierten Zustand, da das Verhalten des Flip-Flops nach einem Wechsel des Eingangssignals nicht vorhersehbar ist und z.B. von 'Unsymmetrien in der Halbleiterstruktur' der beiden *NAND-Gatter* abhängt. *RS-Flip-Flops* werden häufig als Datenspeicher verwendet, bei dem die Dateneingabe und Datenausgabe durch ein Taktsignal (**C**lock) gesteuert wird. Liegt am *C-Eingang* ein H-Pegel an, dann übernimmt das Flip-Flop die an *D* angelegte Information. Liegt dagegen ein L-Pegel am *C-Eingang*, dann behält das Flip-Flop seinen Zustand bei. Aber auch bei diesem *taktzustands-gesteuerten* Flip-Flop gibt es einen verbotenen Zustand bei dem das Flip-Flop in eine nicht definierte Lage kippt. Um einen undefinierten ($R = S = 1$) Zustand zu vermeiden, kann das Signal des R-Eingang mit einem NICHT-Gatter aus dem S-Eingang erzeugt werden. Ein solches Flip-Flop ist das *D-Flip-Flop*, das auf dem Experimentierboard verwendet wird. Das Schaltbild und das Schaltzeichen eines *D-Flip-Flops* sind in der unten stehenden Abbildung zu sehen.



Das *D-Flip-Flop* besteht also aus einem taktgesteuertem *RS-Flip-Flop* in Verbindung mit einem NICHT-Gatter, das so geschaltet ist, dass immer $S = \bar{R}$ gilt. Der *R-Eingang* wird dadurch nicht mehr von außen angesteuert und undefinierte Zustände werden vermieden. Die Ansteuerung kann auf verschiedene Arten erfolgen. Mit der *statischen Ansteuerung* erfolgt sie über einen Spannungsimpuls, der einige Zeit andauert. Mit der *dynamischen Ansteuerung* ändert sich der Ausgangszustand durch eine steile Spannungsflanke. Liegt zum Beispiel am *Takt-Eingang C* eine periodische Rechteckspannung an, dann ist die Ausgangsfrequenz an

Qhalb so groß wie die Taktfrequenz. Das Flip-Flop arbeitet als *Frequenzteiler*. Durch jedes Flip-Flop wird die Eingangsfrequenz durch zwei geteilt, bei n Flip-Flops kann die Eingangsfrequenz maximal durch 2^n geteilt werden. Werden die Ausgänge mit 2^0 , 2^1 und 2^2 bezeichnet, dann wird klar, dass die Flip-Flops als *Binärzähler* geschaltet sind. In der unten stehenden Abbildung ist links der Takteingang C und die Ausgangszustände Q_i zweier negativ-Flanken-gesteuerten Flip-Flops zu sehen. Um die Funktionsweise zu verdeutlichen wird ein *Impulsdiagramm* verwendet. Diese 'Zeitablauf'-Diagramme veranschaulichen die Pulsabfolge der Ausgänge nach vorgegebenen Eingangssignalen.



So durchläuft ein 3-Bit-Binärzähler nacheinander die Binärzahlen von 000 bis 111 . Zählt also von 0 bis 7 im Dezimalsystem. Um von 0 bis 9 (binär 1001) zählen zu lassen werden also vier Flip-Flops benötigt, womit sogar bis 15 (1111) gezählt werden kann. Häufig ist es sinnvoll den Zähler nur bis zu einem vorgegebenen Wert zählen zu lassen. In diesen Fällen wird der Zähler beim Erreichen dieses Wertes durch zusätzliche Gatter auf Null zurückgesetzt. Ein Zähler, der von Dezimal 0 bis 9 zählt und dann zurückgesetzt wird, wird als **Binär Codierter Dezimalzähler** bezeichnet.

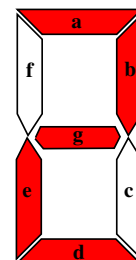
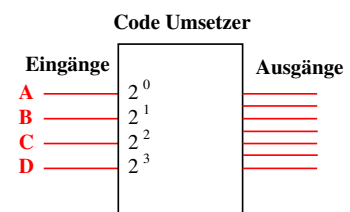
Um in einem physikalischen Experiment die Frequenz eines Signals zu messen, werden die Signalimpulse gezählt, die innerhalb einer bestimmten *Torzeit* T ankommen. Zur Erzeugung der Torzeit wird häufig ein Quarzoszillator verwendet, dessen Frequenz mit Teilern herabgeregelt wird. Durch eine logische Verknüpfung von Signal und Torzeit, die bei Zählerbausteinen häufig als *ENABLE* bereits integriert ist, werden nur Signalimpulse gezählt, die während der Torzeit eintreffen. Um so länger die Torzeit gewählt wird, um so genauer wird das Ergebnis.

Vorbereitung

- Es stehen nur NAND-Elemente mit je zwei Eingängen zur Verfügung. Bauen Sie mit den NAND-Gattern eine *UND-Verknüpfung* und eine *Oder-Verknüpfung* auf.

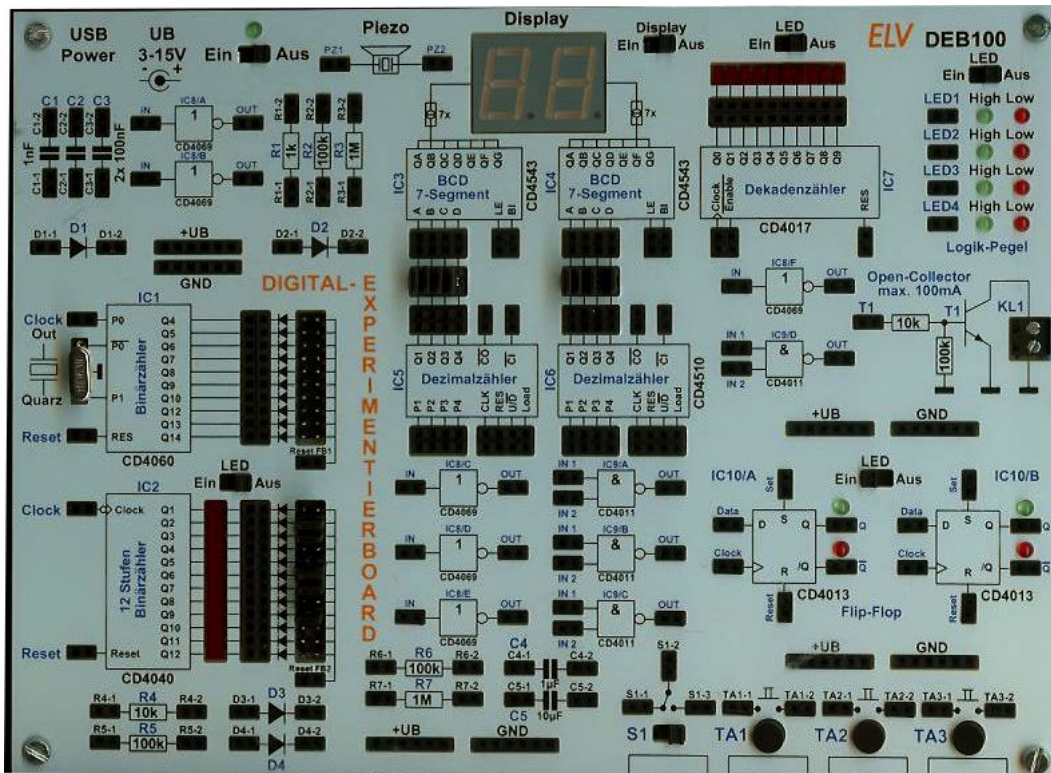
- Für digitale Anzeigegeräte werden häufig *7-Segment-Anzeigeeinheiten* verwendet, wie sie in der untenstehenden Abbildung zu sehen ist. Damit eine Dezimalzahl mit einer 7-Segmentanzeige angezeigt werden kann, muss die Dezimalzahl zuerst in eine Dualzahl umgewandelt werden, um dann anschliessend in einen *7-Segmentcode* umgesetzt zu werden. Vervollständigen Sie die Tabelle. Auf dem Experimentierboard erledigt diese Aufgabe das *CD4543*.

Dezimal- zahl	Eingänge Dualzahl				Ausgänge 7-Segment-Code						
	2^3	2^2	2^1	2^0	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2											
3											
4											
5											
6											
7											
8											
9											



Versuchsaufbau

Kernstück des Experimentes ist eine Platine (DEB100), auf der sich verschiedene digitale Bauelemente befinden. So gibt es einfache Verknüpfungen wie z.B. NAND- und NICHT-Gatter aber auch komplexere Digitale Bauelemente wie z.B. Flip-Flops oder Dezimalzähler. Die Spannungsversorgung der Bauelemente erfolgt durch Batterien auf der Rückseite der Platine. Durch einen Schiebeschalter kann die Spannungsversorgung unterbrochen werden. Auf dem Experimentierboard befinden sich an verschiedenen Stellen LED-s, mit denen der Logikzustand innerhalb der Schaltungen angezeigt werden kann. Die LED's können mit separaten Schaltern aktiviert werden um die Lebensdauer der Batterie zu erhöhen. Für die Verbindung der einzelnen Bauelemente werden *Steckbrücken* verwendet. Um auf die Bauelemente ein *High-Level* zu legen, müssen Sie den Eingang des Bauelementes mit **+UB** verbinden; für ein *Low-Pegel* mit **GND**. Einzelne Signale (z.B. zum Zählen) können mit den Drucktasten (TA1 bis TA3) erzeugt werden. Pulssequenzen können mit dem Quarzoszillator erzeugt werden. Der verwendete Quarzoszillator hat eine Frequenz von 4,194304 MHz und kann mit dem Binärzähler (IC1) bis zu einer Frequenz von 256 Hz heruntergeteilt werden. In diesem Frequenzbereich kann mit Hilfe von Steckbrücken (*Jumper*) verschiedene Frequenzen durch Wahl des geeigneten Teiler- verhältnisses erzeugt werden.



Das auf dem Experimentierboard verwendete D-Flip-Flop (CD 4013) kann auf zwei verschiedene Arten betrieben werden. Im *getakteten Modus* wird bei jeder positiven Flanke am Takteingang der Pegel am D-Eingang auf den Q-Ausgang übertragen. Bei Verwendung der *R*- und *S*-Eingänge setzt ein *High-Level* an *S* das Flip-Flop. Durch ein *High-Level* am Reset Eingang wird es wieder zurückgesetzt.

Auf dem Experimentierboard stehen Ihnen ein *Dekadenzähler* (IC7), *Binärzähler* (IC1 und IC2) sowie *Dezimalzähler* (IC5 und IC6) zur Verfügung. Soll der Zählerstand eines der Zähler zurückgesetzt werden, so muß ein 'High-Pegel' auf den *Reset Eingang* gegeben werden.

Der *Dekadenzähler* hat einen Takteingang (Clock) und 10 Ausgänge (Q_i), deren Pegel mit LED's angezeigt werden können. Beim Dekadenzähler wird immer nur ein Ausgang gesetzt. Eine beliebige Anwendung ist die Realisierung eines 'Laufflichtes'. Sollen weniger (i -LED's) als die 10 zur Verfügung stehenden LED's angesprochen werden, muss der *Reset* mit dem Ausgang Q_i verbunden werden.

Es stehen zwei *Binärzähler* zur Verfügung. Bei Binärzählern können alle Ausgänge aktiv sein. Das Muster der angesprochenen LED's entspricht jeweils dem Zählerstand. Einer der Binärzähler (IC1) ist mit einem Quarzoszillator verschaltet. Von diesem können verschiedene Frequenzen abgeleitet werden. Hierzu muss zuerst das entsprechende Teilverhältnis bestimmt werden, dass sich aus mehreren Ausgängen zusammensetzen kann. Durch die interne Verdrahtung des Experimentierboards ist der Zählerstand $= f/2$. Die entsprechenden Kombinationen können durch Jumper ausgewählt werden. Wird z.B. eine Frequenz von 6kHz benötigt, dann muss die Eingangsfrequenz von 4,194304 MHz durch den Faktor 699 geteilt werden (siehe unten). Jetzt müssen die 'Zählerstandsberechnungen' so kombiniert werden, dass sie zusammen 699 ergeben. Es wird mit dem größt möglichen 'Zählerstand' angefangen (hier 512) und dann sukzessive die nächst größeren Zählerstände ausgewählt. In dem vorliegenden Beispiel muss ein Jumper bei Q_{10} , Q_8 , Q_6 , Q_5 und Q_4 gesetzt werden. Sind nun alle diese Bits gleichzeitig *H*, wird durch das Diodennetzwerk ('wired or') der *Reset* ausgelöst. Da Q_1 bis Q_3 auf diesem Board nicht herausgeführt sind, kann lediglich ein Faktor 696 eingestellt werden und die erzeugte Frequenz ist etwas größer als 6 kHz.

Ausgang	Teilerfaktor	Zählerstand	Jumper
Q1	2	nicht ausgeführt	
Q2	4	nicht ausgeführt	
Q3	8	nicht ausgeführt	
Q4	16	8	x
Q5	32	16	x
Q6	64	32	x
Q7	128	64	
Q8	256	128	x
Q9	512	256	
Q10	1024	512	x
Q11	2048	nicht ausgeführt	
Q12	4096	2048	
Q13	8192	4096	
Q14	16384	8192	
		Summe	696

Beim *Dezimalzähler* ist bereits eine Resetlogik enthalten, die ihn nach Erreichen der 9 wieder auf 0 zurücksetzt. Der Zählerstand kann mittels des *BCD Dekoders* auf der *7-Segmentanzeige* angezeigt werden. Jeder 'High-Pegel' am Takt-Eingang (CLK) erhöht den Zählerstand um Eins. Durch kaskadieren der zwei 7-Segment-Anzeigen kann eine Dezimalzahl mit zwei Ziffern

ausgegeben werden.

Mit dem *Piezosummer* können akustische Signaltöne in einem breiten Frequenzbereich erzeugt werden. Die Resonanzfrequenz des verwendeten Piezos liegt zwischen 4.5 kHz und 5.5 kHz. In diesem Frequenzbereich hat der Piezo seine größte Lautstärke. Kleinere Frequenzen z.B. 1 kHz sind weniger störend.

Moderne Gatter und IC's haben in der Regel viele Eingänge und Ausgänge. Damit ungenutzte Eingänge bzw Ausgänge nicht 'floaten' (d.h. es durch unkontrollierte Ströme zu Fehlschaltungen kommt), müssen die ungenutzten Eingänge und Ausgänge mit einer logischen '1' oder '0' verbunden werden. Dies wird in der Regel durch einen geeigneten Widerstand realisiert. Auf dem verwendeten Experimentierboard DEB100 sind alle Eingänge und Ausgänge intern durch Widerstände oder Dioden auf ein definiertes Potential gelegt. Aus diesem Grund müssen Sie die ungenutzten Eingänge bzw Ausgänge nicht verschalten.

Aufgaben und Versuchsdurchführung

Aufgabe 1: Machen Sie sich mit dem *Experimentierboard* vertraut. Identifizieren Sie hierzu die einzelnen Bauelemente und beantworten Sie folgende Fragen.

- Wieviel *NICHT-Gatter* befinden sich auf dem Board? _____
- Was passiert, wenn Sie alle *NICHT-Gatter* hintereinander schalten?

- Welchen Logikpegel erhalten Sie dann? _____

Aufgabe 2: Bauen Sie eine UND-Verknüpfung und eine ODER-Verknüpfung aus NAND-Elementen auf (Vorbereitungsaufgabe) und ermitteln Sie mit Hilfe der Wahrheitstabellen die Richtigkeit ihrer Lösung.

UND-Verknüpfung

E_1	E_2	A	LED	
			High	Low

ODER-Verknüpfung

E_1	E_2	A	LED	
			High	Low

Aufgabe 3: Bauen Sie ein NAND-Gatter mit drei Eingängen auf und ermitteln Sie mit Hilfe der Wahrheitstabelle die Richtigkeit der Lösung.

E_1	E_2	E_3	A

E_1	E_2	E_3	A

Schaltbild:

Aufgabe 4: Bauen Sie ein RS-Flip-Flop aus NAND-Gattern auf und vervollständigen Sie die Wahrheitstabelle. Beschreiben Sie was bei $\overline{S} = \overline{R} = 0$ und $\overline{S} = \overline{R} = 1$ passiert.

\overline{S}	\overline{R}	Q	\overline{Q}	
0	0	1	1	logisch verboten
1	0			Setzen
0	1			Rücksetzen
1	1	X	X	Speichern

Aufgabe 5: Bauen Sie aus dem *Dekadenzähler* (IC7) ein Lauflicht. Verwenden Sie hierzu einmal 10 LED's und einmal 8 LED's. Verwenden Sie eine Frequenz von 16 Hz.

Aufgabe 6: Bauen Sie aus dem *12-Stufen Binärzähler* (IC2) einen Zähler auf, der bis zur Dezimalzahl '9' zählt. Überprüfen Sie die *Binärzahl* mit der *7-Segment-Anzeige*.

Aufgabe 7: Erzeugen Sie mit dem Piezo einen pulsierenden Ton. Verwenden Sie den Binärzähler mit Quarzoszillator (IC1) um die Signalfrequenz zu generieren und den 12-Stufen Binärzähler für das Festsetzen der Pausen. Um eine bestimmte Abfolge von Pulsen und Pausen zu realisieren, müssen die verschiedenen Ausgänge über NICHT-Gatter und NAND-Gatter verknüpft werden.

In der folgenden Schaltung wird ein Doppelpuls (Piep-Piep - Pause - Piep-Piep - usw) erzeugt. Überprüfen Sie die Puls-Pause Sequenz. Durch Variation der Ausgänge können andere Sequenzen erzeugt werden. Erzeugen sie einen Dreifachpuls (Piep-Piep-Piep - Pause - Piep-Piep-Piep - usw.)

