

Digitale Bauelemente in Schaltungen anwenden

Da digitale Bauelemente elementare Bestandteile jeder EDV-Anlage sind, müssen Informatikerinnen und Informatiker die grundsätzliche Funktionsweise dieser Bauelemente verstehen.

Digitale Bauelemente sind hochintegrierte Schaltungen, in denen logische Verknüpfungen nach den Gesetzen der Booleschen Algebra realisiert sind. Sie ermöglichen die Manipulation, Berechnung und Speicherung von Daten in digitaler Form.

Üblicherweise liegen Informationen in analoger Form vor, z.B. Temperatur, Musik. Daher ist in EDV-Anlagen für die Ein- und Ausgabe eine Analog-/Digital-Wandlung bzw. Digital-/Analog-Wandlung notwendig.

Im Lernbereich 1 werden die logischen Grundfunktionen und deren Realisierung erläutert. Anschließend wird die Anwendung der Booleschen Algebra, zur Vereinfachung und Minimierung des Bauteileaufwandes bei Digitalschaltungen aufgezeigt.

Ausgehend von den logischen Grundfunktionen wird im Lernbereich 2 und 3 die Wirkungsweise höherintegrierter Logikschaltungen, wie Zähler, Schieberegister, Decoder und Multiplexer, dargestellt.

Den Abschluss dieses Lernmoduls bildet der Lernbereich 4 mit einer Beschreibung der Merkmale und Einsatzmöglichkeiten von D/A- A/D-Wandlern.

Voraussetzung für dieses Lernmodul ist eine erfolgreiche Bearbeitung des Lernmoduls 2 dieses Faches

- Elektronische Bauteile kennen und elektronische Schaltungen berechnen

Alle weiteren notwendigen Informationen und Arbeitsunterlagen sind in diesem Lernmodul und in dem Modul „Formeln und Datenblätter“ enthalten.

Dieses Lernmodul ist im häuslichen Studium zu erarbeiten.

Der benötigte Zeitaufwand liegt bei ca. 17 Stunden.

Zusätzlich finden im Begleitunterricht 4 Stunden Festigung und Vertiefung fachspezifischer und fächerübergreifender Zusammenhänge sowie die Beschreibung von Lösungsverfahren zur Bearbeitung typischer Aufgaben und Problemstellungen statt.

In diesen 4 Stunden ist das lernmodulübergreifende Laborprojekt „Analoge und digitale Schaltungen“ im Umfang von ca. 2 Stunden integriert.

LERNMODUL 3

Ziele

Ausgangssituation

Planung

Inhaltsverzeichnis

1 Logische Verknüpfungen	3
1.1 Grundfunktionen und deren Realisierung	3
1.2 Boolesche Algebra	16
2 Schaltungen mit Speicherverhalten	27
2.1 Flip-Flop	27
2.2 Zähler	37
2.3 Schieberegister	42
3 Codier- und Auswahlschaltungen	50
3.1 Decoder	52
3.2 Multiplexer	58
4 D/A- und A/D-Wandler	64
4.1 D/A-Wandler	64
4.2 A/D-Wandler	69
Lösungsanhang	84

1 Logische Verknüpfungen

Lernbereich

1.1 Grundfunktionen und deren Realisierung

In allen Bereichen der Technik werden zunehmend digitale Signale verwendet, da sie gegenüber analogen Signalen wesentliche Vorteile aufweisen:

- Die Verarbeitung einer digitalen Größe kann sowohl mittels Hardware als auch mittels Software erfolgen.
- Die Verarbeitung mittels Software bietet ein hohes Maß an Flexibilität, da eine Anpassung an eine vorgegebene Aufgabenstellung durch eine Programmanpassung möglich ist.
- Digitale Größen können durch Erhöhung der Bit-Anzahl mit beliebiger Genauigkeit dargestellt werden.
- Digitale Signale sind auf Grund der verwendeten Spannungspegel relativ störunanfällig.

Ein einfaches Beispiel für eine digitale Signalverarbeitung zeigt das unten abgebildete Beispiel eines Förderbandes:

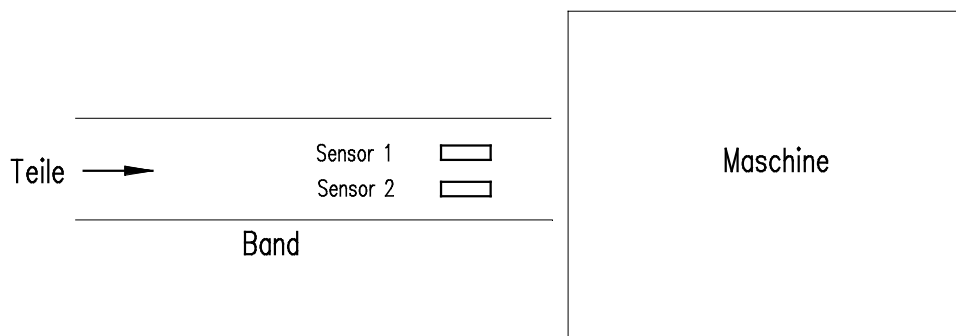


Abbildung 1 Beispiel Förderband

Mithilfe eines Förderbandes werden der Maschine zu bearbeitende Teile zugeführt. Um alle ankommenden Teile zu erfassen, werden zwei Sensoren eingesetzt. Erfassen die Sensoren 1 und/oder 2 ein Teil, so wird das Bearbeitungsprogramm der Maschine für das Teil gestartet. Die abgebildete Tabelle gibt die möglichen Signalkombinationen der Sensoren wieder.

Sensor 1	Sensor 2	Bemerkung
Kein Signal	Kein Signal	Kein Teil erfasst
Signal	Kein Signal	Teil erfasst
Kein Signal	Signal	Teil erfasst
Signal	Signal	Teil erfasst

Tabelle 1 Sensorsignale

Die Ausgänge der Sensoren liefern zwei Zustände, Signal und kein Signal. Diese beiden Zustände, auch Binärwerte genannt, werden mit „0“ und „1“ bezeichnet, unabhängig vom Typ der Sensoren und der verwendeten Logik zur Auswertung der Signale. Bezogen auf die Schaltungsrealisierung werden die Bezeichnungen High-Pegel „H“ und Low-Pegel „L“ verwendet.

Man unterscheidet die positive und die negative Logik mit folgender Zuordnung:

Positive Logik	Negative Logik
$H \triangleq 1$	$H \triangleq 0$
$L \triangleq 0$	$L \triangleq 1$

Tabelle 2 Positive und negative Logik

Im Folgendem wird die **positive Logik** verwendet.

„L“ Low-Pegel beschreibt einen niedrigen Spannungspegel und „H“ High-Pegel einen hohen Spannungspegel. In der schaltungstechnischen Realisierung ist nur bei einer Kontaktlogik ein Spannungspegel den Signalen „L“ und „H“ zugeordnet, z.B.

„L“ $\triangleq 0\text{ V}$ und „H“ \triangleq Betriebsspannung (z.B. 5 V)

In allen anderen Fällen ist den beiden logischen Pegeln ein Spannungsbereich zugeordnet, z.B.

„L“ $\triangleq 0\text{ V} \dots 0,8\text{ V}$ und „H“ $\triangleq 2\text{ V} \dots 5\text{ V}$

Die Spannungsbereichswerte werden durch die unterschiedlichen technischen Ausführungen der verwendeten Logikschaltkreise bestimmt, z.B. bipolare Transistoren oder CMOS-Transistoren. Zweckmäßigerweise wird daher eine digitale Logik-Schaltung mit Bausteinen, auch Gatter genannt, der gleichen Logikfamilie realisiert, z.B. in TTL- oder CMOS-Logik.

Die Schaltung zur Auswertung der Sensorsignale beruht, unabhängig von der technischen Ausführung, auf den logischen Grundverknüpfungen ODER (OR), UND (AND), und NICHT (NOT). Neben diesen Grundverknüpfungen werden bei der Realisierung derartiger Schaltungen auch die Verknüpfungen NICHT-UND (NAND), NICHT-ODER (NOR) und EXKLUSIV-ODER (EXOR) eingesetzt.

ODER-Verknüpfung, OR-Glied, OR-Gatter

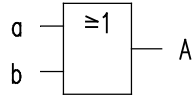
Im oben aufgeführten Beispiel erzeugt die Schaltung zur Auswertung der Sensorsignale mithilfe ihrer Eingangssignale a (Signal Sensor 1) und b (Signal Sensor 2) das Ausgangssignal A. Meldet der Sensor 1 ($a = 1$) oder der Sensor 2 ($b = 1$) oder beide ein Teil, so erscheint am Ausgang der Verknüpfung $A = 1$, „Teil erfasst“. Geben beide Sensoren das logische Signal „0“ ab, so bedeutet dies: „Kein Teil erfasst“. Beschrieben wird diese Verknüpfung mithilfe einer Wahrheitstabelle, in der die möglichen Eingangskombinationen und die zugehörigen Zustände der Ausgangsgröße aufgelistet sind.

Anzahl der möglichen Eingangskombinationen: 2^N

$N = 2 \Rightarrow 2^2 = 4$ Eingangskombinationen

a (Sensor 1)	b (Sensor 2)	A	Bemerkung
0	0	0	Kein Teil erfasst
0	1	1	Teil erfasst
1	0	1	Teil erfasst
1	1	1	Teil erfasst

Tabelle 3 Wahrheitstabelle der ODER-Verknüpfung

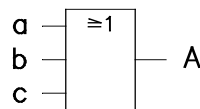

Abbildung 2 Schaltsymbol
ODER-Verknüpfung

Für die ODER-Verknüpfung wird das abgebildete Schaltzeichen verwendet.

Das Zeichen ≥ 1 bedeutet, dass der Ausgang $A = 1$ ist, wenn wenigstens ein Eingang 1 ist.

Die folgende Wahrheitstabelle beschreibt also ebenfalls eine ODER-Verknüpfung, allerdings für 3 Eingänge.

a	b	c	A
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Tabelle 4 Wahrheitstabelle für
3 Eingangsgrößen

Abbildung 3 ODER-Gatter
mit 3 Eingängen

Die ODER-Verknüpfung mit zwei Eingängen wird mithilfe der Schaltalgebra durch die Funktionsgleichung

$$A = a \vee b \quad (\vee \text{ Zeichen für die ODER-Verknüpfung von Eingangsgrößen})$$

beschrieben. Häufig wird auch das Zeichen $+$ verwendet.

$$A = a + b$$

UND-Verknüpfung, AND-Glied, AND-Gatter

Abbildung 4 zeigt ein Beispiel einer UND-Verknüpfung in Kontakttechnik.

Die Lampe leuchtet nur auf, wenn der Schalter S1 **UND** der Schalter S2 **UND** der Schalter S3 geschlossen sind.

Für die abgebildete Schaltung gilt somit folgende Zuordnung:

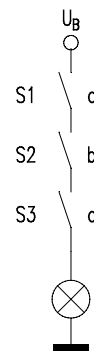


Abbildung 4 Reihenschaltung von Schaltern

S1	S2	S3	Lampe
Offen	Offen	Offen	Aus
Offen	Offen	Zu	Aus
Offen	Zu	Offen	Aus
Offen	Zu	Zu	Aus
Zu	Offen	Offen	Aus
Zu	Offen	Zu	Aus
Zu	Zu	Offen	Aus
Zu	Zu	Zu	Ein

Tabelle 5 Zustände der Reihenschaltung

Somit arbeitet diese Schaltung bei positiver Logik als UND-Verknüpfung:

a	b	c	A
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Tabelle 6 Wahrheitstabelle der UND-Verknüpfung

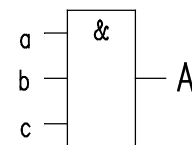


Abbildung 5 Schaltzeichen UND-Verknüpfung

Das Zeichen & bedeutet, dass der Ausgang $A = 1$ ist, wenn an allen Eingängen 1 anliegt.

Die Formel

$$A = a \wedge b \wedge c \quad (\wedge \text{ Zeichen für die UND-Verknüpfung von Eingangsgrößen})$$

beschreibt die UND-Verknüpfung der drei Eingängen a, b, c. Häufig wird auch das Zeichen „·“ verwendet oder die Punkte werden ganz weggelassen.

$$A = a \cdot b \cdot c = a b c$$

NICHT-Glied, NOT-Gatter, Inverter

Das Nicht-Glied wird auch als Inverter, Negator oder NOT-Gatter bezeichnet. Dieses Glied liefert am Ausgang A den umgekehrten logischen Pegel des Eingangs.

a	A
0	1
1	0

Tabelle 7 Inverter

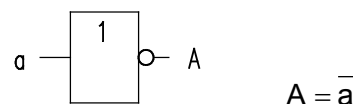


Abbildung 6 Schaltzeichen und Gleichung der NICHT-Stufe

Durch den Negationsstrich über der Eingangsvariablen wird verdeutlicht, dass die Ausgangsgröße A den invertierten Zustand der Eingangsgröße annimmt.

UND-NICHT-Verknüpfung, NAND-Glied, NAND-Gatter

Die UND-NICHT-Verknüpfung wird auch als NAND-Verknüpfung bezeichnet. Bei einer NAND-Verknüpfung werden mehrere Eingangssignale zuerst UND verknüpft. Das Ergebnis dieser Verknüpfung wird invertiert (Abbildung 7a). Beim NAND-Schaltzeichen, wird der Inverter durch den Negationspunkt am Ausgang des zum Symbols ersetzt (Abbildung 7b).

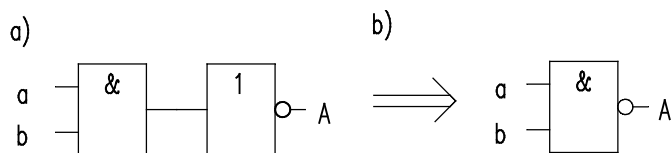


Abbildung 7 Schaltsymbole der UND-NICHT-Verknüpfung

Es ergibt sich folgende Wahrheitstabelle und Funktionsgleichung:

a	b	A
L	L	H
L	H	H
H	L	H
H	H	L

$$A = \overline{a \wedge b}$$

Tabelle 8 Wahrheitstabelle NAND-Verknüpfung mit logischen Pegeln in positiver Logik

NICHT-ODER-Verknüpfung, NOR-Glied, NOR-Gatter

Die NICHT-ODER-Verknüpfung wird auch als NOR-Verknüpfung bezeichnet. Bei einem NOR-Gatter werden mehrere Eingangssignale zunächst von einem ODER-Gatter ausgewertet. Das Ergebnis dieser Verknüpfung wird invertiert (Abbildung 8a). In Abbildung 8b ist das Symbol für die NOR-Verknüpfung wiedergegeben.

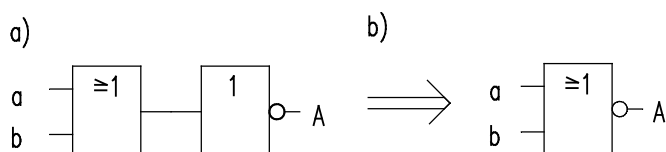


Abbildung 8 Schaltsymbole der NICHT-ODER-Verknüpfung

Es ergibt sich folgende Wahrheitstabelle und Funktionsgleichung:

a	b	A
L	L	H
L	H	L
H	L	L
H	H	L

$$A = \overline{a \vee b}$$

Tabelle 9 Wahrheitstabelle NOR-Verknüpfung mit logischen Pegeln in positiver Logik

EXKLUSIV-ODER-Verknüpfung, EXOR-Gatter

In einigen Fällen müssen zwei digitale Signale miteinander verglichen werden.

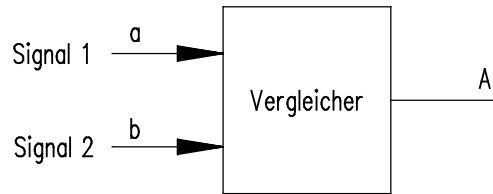


Abbildung 9 Digitaler Vergleicher

Ein Oszillator soll ein Frequenz von 100 MHz (Sollwert = a) erzeugen. Auf Grund von Temperaturschwankungen kann die erzeugte Frequenz von der Vorgabe 100 MHz abweichen. Der Oszillator muss also nachgestimmt werden, damit er in jedem Fall eine Frequenz von 100 MHz liefert. Der Istwert, Signal 1 (Eingangsvariable a), die vom Oszillator erzeugte Frequenz, und der Sollwert Signal 2 (Eingangsvariable b) der Frequenz werden daher mithilfe einer logischen Verknüpfung verglichen. Diese Verknüpfung liefert am Ausgang A immer dann eine 1 bzw. HIGH-Pegel, wenn die beiden digitalen Signale unterschiedlich sind. Dies führt zur abgebildeten Wahrheitstabelle:

A	b	A	Bemerkung
L	L	L	Istwert = Sollwert
L	H	H	Istwert \neq Sollwert, Oszillator nachstimmen
H	L	H	Istwert \neq Sollwert, Oszillator nachstimmen
H	H	L	Istwert = Sollwert

Tabelle 10 Wahrheitstabelle EXOR-Verknüpfung mit logischen Pegeln in positiver Logik

Die Gleichung lautet: $A = (\bar{a} \wedge b) \vee (a \wedge \bar{b})$.

Das Schaltzeichen ist in Abbildung 10a wiedergegeben. In Abbildung 10b ist die Realisierung der logischen Verknüpfung mit den drei Grund-Verknüpfungen abgebildet.

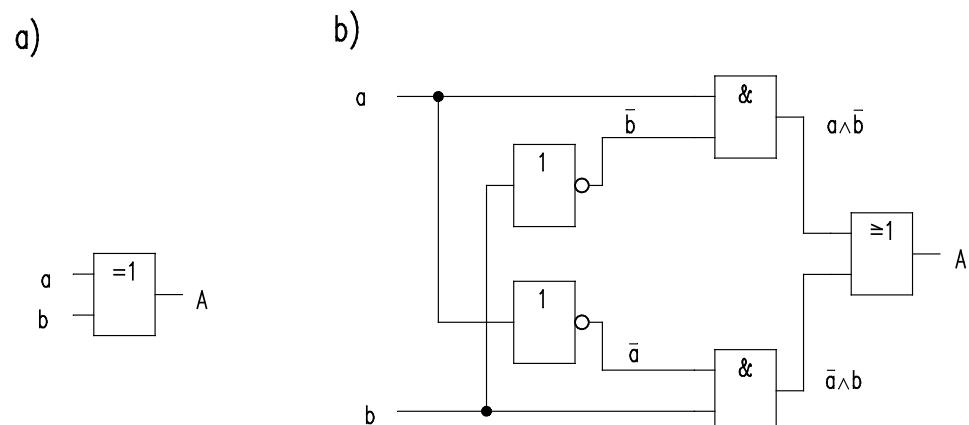


Abbildung 10 Exklusiv-Oder-Verknüpfung

= 1 bedeutet, dass der Ausgang dieser Verknüpfung immer dann 1 ist, wenn nur an **einem** Eingang eine logische 1 anliegt.

TTL-Logik

Logik-Gatter in TTL-Technik (Transistor Transistor Logic) werden als integrierte Schaltungen unter hauptsächlichlicher Verwendung von bipolaren Transistoren hergestellt. Der Grundbaustein der TTL-Technik ist das NAND-Gatter (Abbildung 11).

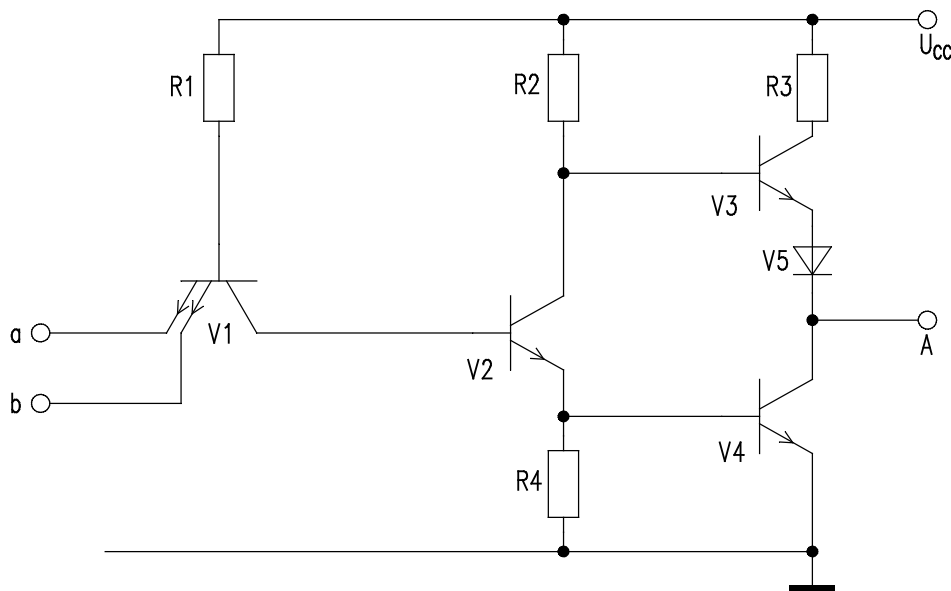


Abbildung 11 Standard TTL-NAND-Gatter vom Typ 7400

Durch die beiden Transistoren V3 und V4 wird der Ausgang praktisch lastunabhängig auf 0 V (L-Pegel) bzw. U_{CC} (H-Pegel) gelegt. Die niederohmige Gegentakt-Endstufe (V_3 , V_4) liefert hohe Ausgangsströme zum schnellen Umladen von Lastkapazitäten und ermöglicht Ausgangssignale mit großer Flankensteilheit.

Der Aufbau der Gegentakt-Endstufe bedingt, dass Ausgänge zweier Gatter nicht parallel geschaltet werden dürfen, da sonst Ausgleichsströme fließen und der Logikpegel nicht mehr eindeutig ist.

Wie verhalten sich offene Eingänge (a oder b an) bei der TTL-Technik? Ein offener Eingang ist nicht in der Lage, den Emitter von V_1 nach Masse zu ziehen. Daher kann der Multi-Emitter-Transistor V_1 (Transistor mit 2 oder mehr Emitter-Anschlüssen) nicht durchsteuern. Die Basis-Emitter-Strecke bleibt gesperrt. Stattdessen fließt ein Strom von U_{CC} über die Basis-Kollektor-Strecke des Transistors V_1 zur Basis von V_2 . V_2 leitet. Über V_2 wird die Basis von V_4 angesteuert und der Ausgang A wird von V_4 auf Masse gezogen. Offene Eingänge bei TTL-Schaltungen haben daher die gleiche Wirkung, wie ein Eingang, der mit H-Pegel beschaltet ist.

TTL-Gatter: **offener Eingang** $\hat{=}$ **H-Pegel**

Der Einsatz von Logik-Gattern in Bussystemen erfordert aber die Verschaltung mehrerer Ausgänge auf eine Sammelleitung. Hierfür stehen spezielle TTL-Gatter mit einem sog. **Tristate-Ausgang** zur Verfügung. Neben den üblichen Pegelzuständen HIGH und LOW existiert noch eine dritte Möglichkeit: der hochohmige Z-Zustand.

Für die dritte Möglichkeit besitzen diese speziellen Gatter einen Steuereingang EN (enable = Freigabe). Mit einem entsprechenden logischen Pegel am zusätzlichen Eingang kann der Ausgang aktiv (niederohmiger H- bzw. L-Pegel) oder passiv (hochohmig) geschaltet werden. Liegt der Steuereingang auf H-Pegel ($EN = 1$), so verhält sich die Schaltung wie ein normales NAND-Gatter. Wird dagegen L-Pegel an den Steuereingang gelegt ($EN = 0$), so sperren die Endstufentransistoren V_3 und V_4 . Der Ausgangswiderstand ist hochohmig. Der Ausgang des NAND-Gatters ist damit abgeschaltet.

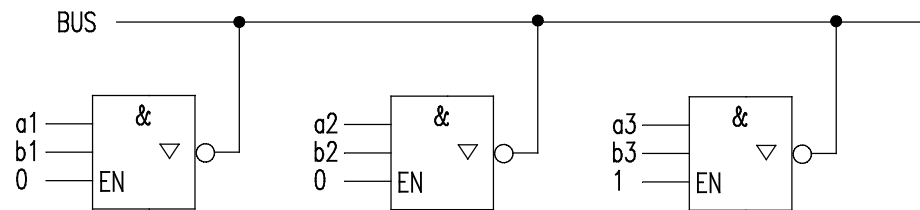


Abbildung 12 Tristate-Gatter an einer Busleitung

Tristate-Gatter dürfen parallel geschaltet (Abbildung 12) werden und damit auf eine gemeinsame Busleitung arbeiten. Es muss aber sichergestellt sein, dass zu einem Zeitpunkt immer nur ein Gatter über den Steuereingang EN aktiviert werden kann.

Tristate-Ausgänge sind durch ein dreieckförmiges Symbol ∇ gekennzeichnet.

CMOS-Logik

Im Gegensatz zur bipolaren TTL-Technik werden in der CMOS-Technik (**C**omplementary **M**etal **O**xide **S**emiconductor) unipolare Feldeffekt-Transistoren mit isoliertem Gate (MOS-FETs) zur Realisierung von Logikfunktionen eingesetzt. Dazu werden selbstsperrende komplementäre MOS-FETs in P-Kanal und N-Kanal Ausführung zusammengeschaltet. Beide Transistortypen sind isoliert auf einem Substrat untergebracht und haben geringe Abmessungen, sodass sich integrierte Schaltungen mit großer Packungsdichte herstellen lassen.

Die folgende Abbildung zeigt den typischen symmetrischen Aufbau von CMOS-Gattern am Beispiel eines NICHT-Gatters.

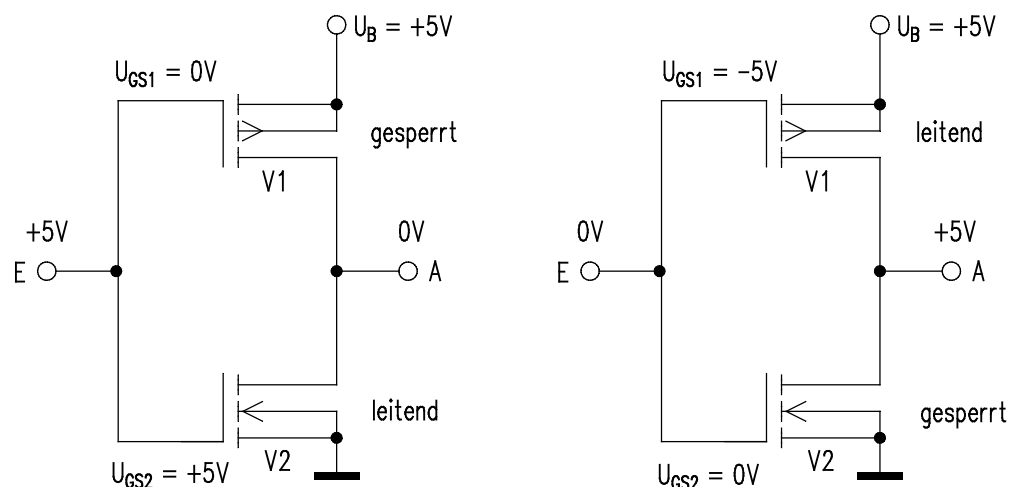


Abbildung 13 NICHT-Gatter

Da beim CMOS-NICHT-Gatter immer einer der beiden Transistoren gesperrt ist, fließt kein Ruhestrom. Wegen der hochohmigen Eingänge (10^{12} bis $10^{14} \Omega$) wird auch kein Steuerstrom benötigt. Nur während des Umschaltens von einem Logikzustand in den anderen fließt ein kleiner Querstrom, da beide Transistoren kurze Zeit schwach leiten und Transistorkapazitäten umgeladen werden. Der Leistungsbedarf ist daher besonders niedrig, nimmt aber mit wachsender Umschalthäufigkeit stetig zu. So liegt die Leistungsaufnahme bei einer Schaltfrequenz von 1 kHz unter 1 μ W und steigt auf ca. 100 μ W bei 100 kHz.

CMOS-Schaltungen sind sehr empfindlich gegenüber statischen Aufladungen. Um die Zerstörung der Transistoren zu vermeiden, schützen zusätzliche Diodennetzwerke die hochohmigen Eingänge der Logik-Gatter und begrenzen Überspannungen auf zulässige Werte.

Offene CMOS-Eingänge besitzen kein definiertes Potenzial. Auf Grund der hochohmigen Eingangsimpedanz neigen Gatter mit offenen Eingängen zu Schwingungen. Dadurch erhöht sich zum einen die Verlustleistung erheblich (ständiger Querstrom durch beide Ausgangstransistoren) und zum anderen stellen sich unerwünschte Logikzustände ein. Deshalb **müssen** alle Eingänge von CMOS-Gattern beschaltet werden. Unbenutzte Eingänge sind an Masse bzw. $+U_B$ anzuschließen. Dies gilt auch für vollständig unbenutzte Gatter, da sich sonst die Schwingungen über die internen Verbindungsleitungen auf benutzte Gatter übertragen können.

Die CMOS-Schaltungsfamilie wird als sog. **4000'er-Serie** von vielen Herstellern vertrieben. Die Typenbezeichnung unterscheidet sich meist nur im Präfix, z.B. CD4011, HBF4011 oder MC14011 für ein vierfach NAND-Gatter.

Technologische Fortschritte im Bereich der Chipgeometrien und der Ersatz der Metal-Gate-Technik durch Silicon-Gate-Technik ermöglichten eine Erhöhung der Arbeitsfrequenzen um den Faktor 10 bei sonst gleichen Eigenschaften. Die Silicon-Gate- oder **High Speed-CMOS-Gatter** (HCMOS) vereinen daher die stromsparenden Vorteile der CMOS-Technik mit der hohen Geschwindigkeit und Ausgangsbelastbarkeit der Low-Power-Schottky-TTL-Gatter. Sie sind die bevorzugte Logikfamilie bei der Neuentwicklung von Digitalschaltungen.

Vergleich der CMOS- und TTL-Logikfamilien

Bevor eine Logikfamilie für den Entwurf einer digitalen Schaltung ausgewählt werden kann, müssen deren Leistungsmerkmale bekannt sein. Dabei sind insbesondere

- Lastfaktoren
- Störsicherheit
- Schaltzeiten

der einzelnen Logikfamilie zu beachten. Dazu kommen die in den Datenblättern der Hersteller angegebenen Grenzwerte für Betriebsspannungen, Ein- und Ausgangsströme sowie Verlustleistungen.

Der **Lastfaktor** beschreibt die Belastung, die ein Gatterausgang durch die Beschaltung mit nachfolgenden Gattereingängen erfährt. Es werden zwei Lastfaktoren verwendet: der Eingangslastfaktor (**Fan In**) und der Ausgangslastfaktor (**Fan Out**).

Für jede Schaltungsfamilie wird eine normale Eingangsbelastung, die sog. Einheitslast festgelegt. Dies entspricht der Stromaufnahme eines Gattereinganges bei H- bzw. L-Pegel. Für TTL-Gatter gilt z.B.:

L-Pegel am Eingang	$I_{IL} = -1,6 \text{ mA}$	(Eingangsdiode in Flussrichtung)
H-Pegel am Eingang	$I_{IH} = 40 \mu\text{A}$	(Eingangsdiode in Sperrichtung)

Das Fan In eines Einganges gibt dann an, um welchen Faktor die Stromaufnahme größer ist als die Einheitslast der Schaltungsfamilie. Normale Gattereingänge besitzen ein Fan In von 1. Falls das Fan In größer als 1 ist (z.B. Kippstufen und höher integrierte Logikschaltungen), wird der Ausgang des treibenden Gatters entsprechend stärker belastet.

Der Ausgangslastfaktor (Fan Out) gibt an, wie viele Eingänge (Fan In = 1) derselben Schaltungsfamilie maximal an den Gatterausgang angeschlossen werden dürfen.

$$\text{Anzahl der Eingänge } n = \frac{\text{Fan Out}}{\text{Fan In}}$$

Durch eingekoppelte Störspannungen können Gatterausgänge ihren Logikzustand unerwünscht wechseln. Damit das nicht geschieht, ist ein Sicherheitsabstand zwischen den zulässigen Logikpegeln erforderlich. Die zulässigen Pegelbereiche und Schaltschwellen sind für die einzelnen Logikfamilien in Datenblättern und in entsprechenden Übertragungskennlinien festgelegt. Die folgenden Abbildungen zeigen die typischen Übertragungskennlinien für TTL-Gatter und CMOS-Schaltkreise.

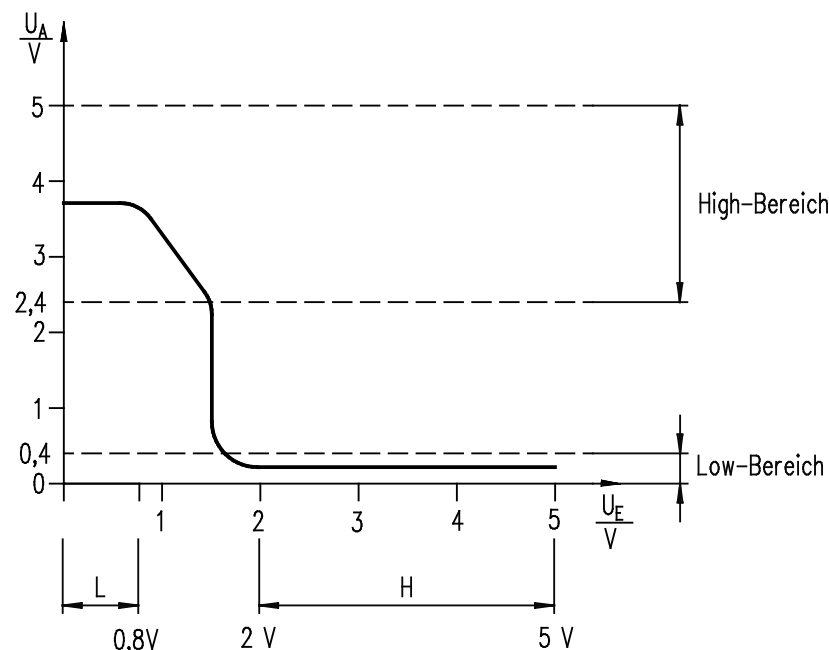


Abbildung 14 Übertragungskennlinie eines TTL-Inverters

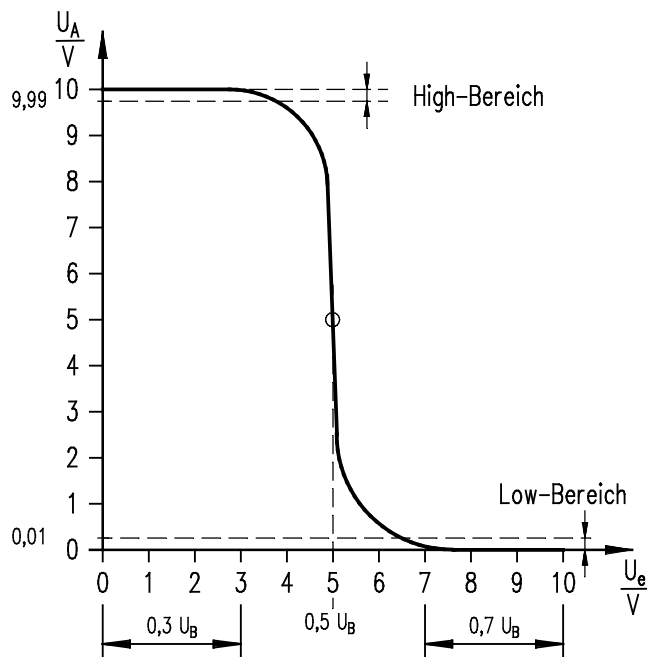


Abbildung 15 Übertragungskennlinie eines CMOS-Inverters

Je nach Breite des Störimpulses wird zwischen statischer **Störsicherheit** (Störimpulsbreite > Signallaufzeit) und dynamischer Störsicherheit (Störimpulsbreite < Signallaufzeit) unterschieden.

Mit Störspannungsabstand wird dabei der Spannungshub bezeichnet, um den die Ausgangsspannung eines Gatters variieren darf, ohne dass ein angeschlossener Gattereingang derselben Logikfamilie in den verbotenen Pegelbereich gelangt. Der statische Störspannungsabstand S lässt sich aus der Übertragungskennlinie bestimmen. Im ungünstigsten Fall (worst-case) gilt:

$$S_H = U_{AH \min} - U_{EH \min} \quad \text{und} \quad S_L = U_{EL \max} - U_{AL \max}$$

Für TTL-Gatter ergibt sich mit

$$S_H = 2,4 \text{ V} - 2 \text{ V} = 0,4 \text{ V} \quad S_L = 0,8 \text{ V} - 0,4 \text{ V} = 0,4 \text{ V}$$

jeweils ein Störabstand von 0,4 V. Einen größeren Störabstand von jeweils ca. 3 V (bei $U_B = 10 \text{ V}$) weisen dagegen CMOS-Gatter auf.

Der dynamische Störabstand kennzeichnet das Verhalten von Logik-Gattern gegenüber kurzen Störimpulsen und wird von den Herstellern in Datenblättern durch Diagramme mit Grenzkurven angegeben.

Für den Einsatz einer Logikfamilie ist die charakteristische **Schaltzeit** bzw. Signallaufzeit von besonderem Interesse. Wegen der vorhandenen Kapazitäten reagiert der Ausgang eines Gatters erst nach einer gewissen Verzögerungszeit auf einen Pegelwechsel am Eingang. In den Datenblättern werden meist idealisierte Ein- und Ausgangssignalverläufe angegeben, wie sie in der folgenden Abbildung für ein NICHT-Gatter zu sehen sind.

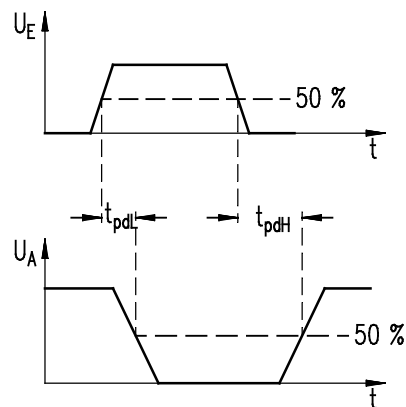


Abbildung 16 Impulsverzögerungszeiten an einem NICHT-Gatter

Die typische **Signallaufzeit** t_{pd} (propagation delay time) einer Logikfamilie wird als mittlere Signallaufzeit der Impulsverzögerungszeiten t_{pHL} (High-Low-Wechsel) und t_{pLH} (Low-High-Wechsel) festgelegt:

$$t_{pd} = \frac{1}{2} \cdot (t_{pHL} + t_{pLH})$$

Die Signallaufzeiten, Leistungsaufnahme und Betriebsspannungsbereiche unterscheiden sich bei den verschiedenen Logikfamilien sehr stark. Zur besseren Übersicht sind sie und weitere Kenndaten der einzelnen Logikfamilien in der folgenden Tabelle zusammengefasst.

Technologie	Standard TTL	Schottky TTL	Low-Power Schottky TTL	Advanced Schottky TTL	Advanced Low-Power Schottky TTL	Metal Gate CMOS	Silicon Gate CMOS (High Speed CMOS)	
	74	74 S	74 LS	74 AS	74 ALS	4000	74 HC	74 HCT
Serienbezeichnung	V	4,75	4,75	4,75	4,75	3,0	2,0	4,75
	min	5,0	5,0	5,0	5,0	5,0	5,0	5,0
	max	5,5	5,5	5,5	5,5	15,0	6,0	5,25
Verlustleistung je Gatter statistisch bei 100 kHz	mW	19	2	8,5	1	10 ⁻³	2,5 · 10 ⁻⁶	2,5 · 10 ⁻⁶
	mW	19	2	8,5	1	0,1	0,17	0,17
t _{pd}	ns	3	10	1,7	4	60	10	10
f _{Takmax} (bei C _L = 15 pF)	MHz	125	40	200	70	8	40	40
Fan Out (LS-Lasten)								
	Standard Ausgänge	40	20	50	20	4	10	10
	High Current Ausgänge	120	60	120/160	60/120	-	15	15
V _{OH}	V	2,0	2,0	2,0	2,0	4,0	3,15	2,0
V _{IL}	V	0,8	0,8	0,8	0,8	1,0	0,8	0,8
V _{OH}	V	2,7	2,7	2,7	2,7	4,5	4,5	2,7
V _{OL}	V	0,5	0,5	0,5	0,5	0,05	0,1	0,5
I _{OH}	µA	50	20	50	20	1	±1	±1
I _{IL}	µA	-2000	-400	-500	-100	-1	±1	±1
I _{OS} (V _O = 0,4 V) min	µA	16	8	20	8	1,6	4	4
Standard Ausgänge	µA	48	24	48/64	24/48	-	6	6
High Current Ausgänge	µA							

Tabelle 11 Kenndaten wichtiger Logikfamilien

1.2 Boolesche Algebra

Die Schaltalgebra, auch Boolesche Algebra genannt, Algebra der logischen Funktionen, hat sich zu einem bedeutenden Hilfsmittel in der digitalen Signalverarbeitung entwickelt. Sie hat die Aufgabe den funktionellen Zusammenhang zwischen den logischen Werten der Ein- und Ausgangsgrößen zu beschreiben. Logische Schaltungen, die auf den Grundfunktionen, UND, ODER und NICHT basieren, sind immer durch die Schaltalgebra beschreibbar. Die Anwendung der Schaltalgebra bietet folgende Vorteile:

- Beim Entwurf einer Schaltung wird mit ihr die Schaltbedingung festgelegt. Die Gleichung beschreibt eindeutig, wie die Ausgangsgrößen von den Eingangsgrößen abhängen.
- Unter Verwendung der Rechenregeln der Booleschen Algebra ist häufig eine Vereinfachung der Ausgangsgleichung möglich. Die Anzahl der Logik-Gatter kann eventuell verringert werden.
- Anhand der Gleichung kann die Logik-Schaltung aufgebaut werden. Es ist sogar möglich unter Verwendung der Regeln die Gleichung für bestimmte Gatter-Typen umzuformen. Dies bezeichnet man als Typisierung.

Das folgende Kapitel ist eine Einführung in die Grundlagen der Schaltalgebra.

Rechenregeln für eine Variable

Jede Größe, die den Wert 0 oder 1 annehmen kann, ist eine Variable.

Jede Größe, die einen vorgegebenen nicht änderbaren Wert hat, ist eine Konstante.

Auf Basis der UND-, ODER- und NICHT-Verknüpfung kann man bei der Verknüpfung einer Konstanten und einer Variablen Vereinfachungsregeln ableiten:

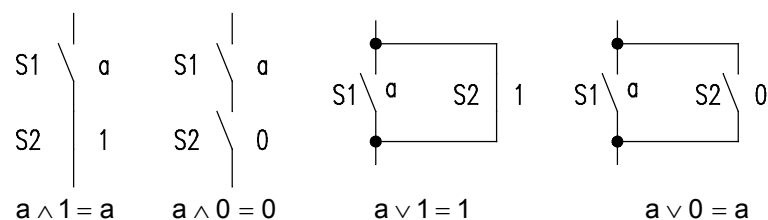


Abbildung 17 Vereinfachungsregeln am Beispiel einer Kontaktschaltung

Weitere Vereinfachungsregeln betreffen die Verknüpfung einer Variablen mit sich selbst:

$$a \wedge a \wedge a \wedge \dots = a \quad a \vee a \vee a \vee \dots = a$$

Wird eine Variable mit ihrer NICHT-Form verknüpft gilt:

$$a \wedge \bar{a} = 0 \quad a \vee \bar{a} = 1$$

Erstellung einer Funktionsgleichung und eines Schaltplans

Anhand eines Beispiels werden die wichtigsten Regeln und die Vorgehensweise von der Wahrheitstabelle bis zur Schaltung erläutert.

In einem Raum mit gefährlichen Chemikalien sind drei Gassensoren angebracht. Melden mindestens zwei der drei Sensoren gefährliche Gase, so soll eine Sirene ausgelöst werden und eine Meldung an die Überwachungszentrale geschickt werden. Die Sensoren geben bei einer erhöhten Gaskonzentration eine 1 an ihren Ausgängen ab. Die Sirene soll ausgelöst werden, wenn am Ausgang der Auswertungsschaltung eine logische 1 erscheint.

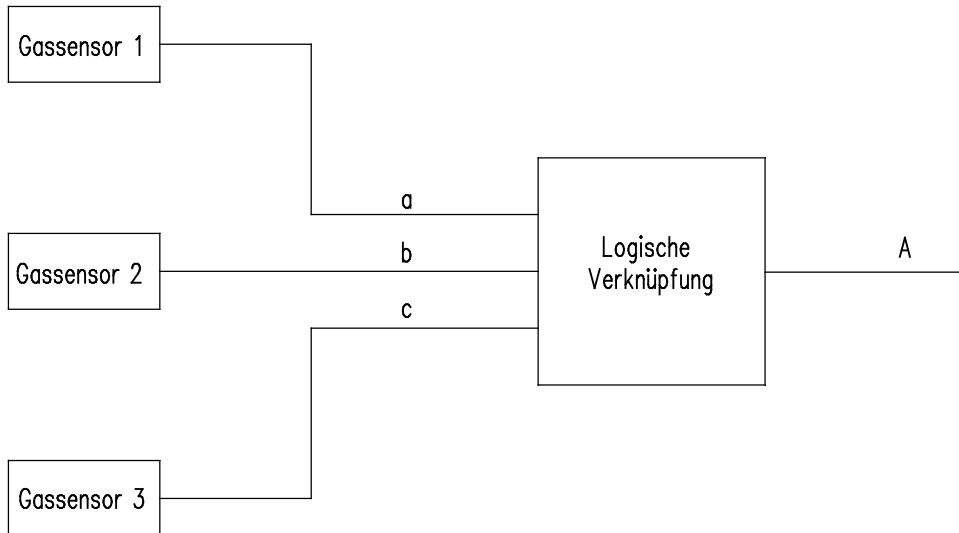


Abbildung 18 Meldeanlage

1. Aufstellen der Wahrheitstabelle

Aus der Beschreibung der Wirkungsweise ergibt sich folgende Tabelle:

Bei 3 Eingangsgrößen (Eingangsvariablen) gibt es $2^3 = 8$ Kombinationen. Immer, wenn mindestens 2 Eingangsgrößen den Wert 1 haben ist die Ausgangsgröße A = 1.

a	b	c	A
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

← 4. Zeile

← 6. Zeile

← 7. Zeile

← 8. Zeile

2. Aufstellen der Funktionsgleichung in ODER-Normalform

Alle Zeilen, bei denen die Ausgröße den Wert 1 hat, müssen miteinander verknüpft werden.

Die Ausgangsgröße A ist 1, wenn die Kombination der 4. Zeile ODER die Kombination der 6. Zeile ODER die Kombination der 7. Zeile ODER die Kombination der 8. Zeile erfüllt ist.

Die Kombination der 4. Zeile ergibt A = 1, wenn a = 0 UND b = 1 UND c = 1 ist. Diese UND-Verknüpfung wird als Konjunktion bezeichnet. Entsprechend kann man bei den anderen Zeilen verfahren.

Daraus kann man eine allgemeine Regel zur Aufstellung einer Funktionsgleichung ableiten:

Jeder 1-Zustand der Ausgangsgröße wird durch eine UND-Verknüpfung (Konjunktion) der Eingangsgrößen hervorgerufen. Hat eine Variable den Wert 0, so tritt sie negiert in der UND-Verknüpfung auf. Hat sie den Wert 1, so geht sie **nicht** negiert in die UND-Verknüpfung ein. Bei mehreren Konjunktionen in einer Wahrheitstabelle ergibt sich die ODER-Normalform durch die ODER-Verknüpfung der Konjunktionen.

Konjunktionen:

$$4. \text{ Zeile : } a = 0 \quad b = 1 \quad c = 1 \Rightarrow \bar{a} \wedge b \wedge c$$

$$6. \text{ Zeile : } a = 1 \quad b = 0 \quad c = 1 \Rightarrow a \wedge \bar{b} \wedge c$$

$$7. \text{ Zeile : } a = 1 \quad b = 1 \quad c = 0 \Rightarrow a \wedge b \wedge \bar{c}$$

$$8. \text{ Zeile : } a = 1 \quad b = 1 \quad c = 1 \Rightarrow a \wedge b \wedge c$$

Alle Konjunktionen (UND-Verknüpfungen) der Zeilen, für die die Ausgangsgröße 1 ist, müssen ODER verknüpft werden, um die Funktionsgleichung in ODER-Normalform zu erhalten.

Somit lautet die Funktionsgleichung:

$$A = (\bar{a} \wedge b \wedge c) \vee (a \wedge \bar{b} \wedge c) \vee (a \wedge b \wedge \bar{c}) \vee (a \wedge b \wedge c)$$

Da die UND-Verknüpfung eine höhere Priorität als die ODER-Verknüpfung hat, kann die Gleichung auch ohne Klammern angegeben werden.

$$A = \bar{a} \wedge b \wedge c \vee a \wedge \bar{b} \wedge c \vee a \wedge b \wedge \bar{c} \vee a \wedge b \wedge c$$

Eine bessere Übersicht bietet die folgende Schreibweise, die nachfolgend im Lernbereich angewandt wird:

$$A = \bar{a}bc \vee a\bar{b}c \vee abc \vee abc$$

In diesem Beispiel wurden die Zeilen in aufsteigender Reihenfolge ODER-verknüpft. Dies ist nicht zwingend vorgeschrieben. Die Reihenfolge kann innerhalb einer gleichartigen Verknüpfung vertauscht werden:

$$\bar{a}bc = cba$$

oder

$$\bar{a}bc \vee a\bar{b}c \vee abc \vee abc = abc \vee \bar{a}bc \vee a\bar{b}c \vee abc.$$

Dieses ist das so genannte Kommutativgesetz (Vertauschungsgesetz).

Mit der Gleichung $A = \bar{a}bc \vee a\bar{b}c \vee ab\bar{c} \vee abc$ kann jetzt eine Schaltung mit den Grund-Verknüpfungen realisiert werden (Abbildung 19):

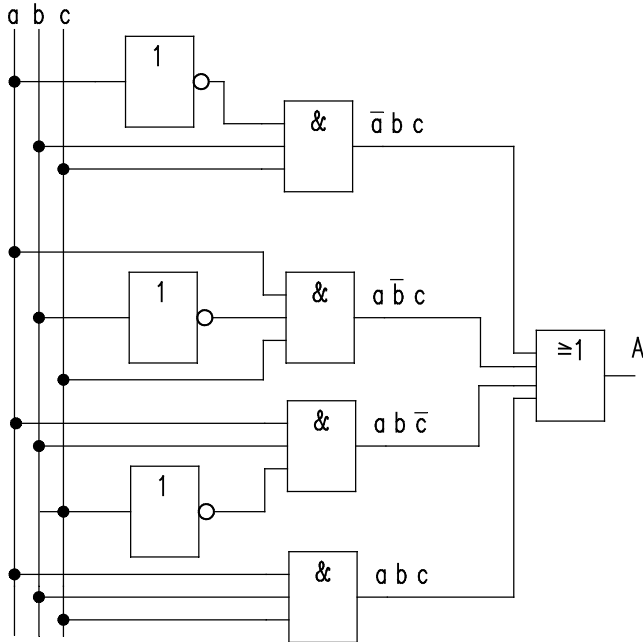


Abbildung 19 Logische Verknüpfung

Zunächst werden die Konjunktionen mit UND-Gattern erstellt. Negierte Eingangsgrößen bzw. Variablen werden durch Inverter realisiert. Anschließend werden die Ausgänge der UND-Gatter ODER-verknüpft. Die Schaltung ist also das direkte Abbild der Gleichung.

Bei dieser Lösung werden UND-Gatter mit 3 Eingängen und ein ODER-Gatter mit 4 Eingängen verwendet. Sollen nur Gatter mit 2 Eingängen eingesetzt werden, so kann man unter Verwendung des Assoziativgesetzes (Verbindungsgesetz) z.B. ein UND-Gatter mit drei Eingängen durch zwei UND-Gatter mit zwei Eingängen ersetzen, siehe Abbildung 20.

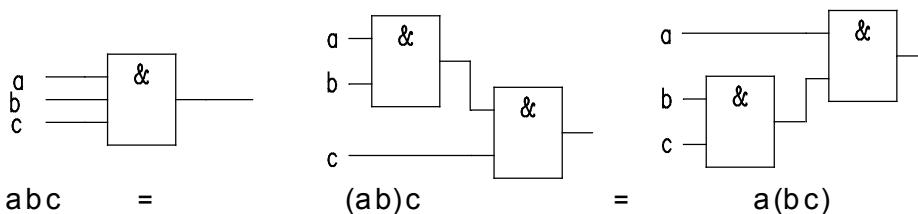


Abbildung 20 Anwendung des Assoziativgesetzes

Setzt man dieses auf die Schaltung von Abbildung 20 um, ergibt sich folgende Gleichung und die dazugehörige Schaltung:

$$A = \{(\bar{a}b)c \vee (a\bar{b})c\} \vee \{a(b\bar{c}) \vee a(bc)\}$$

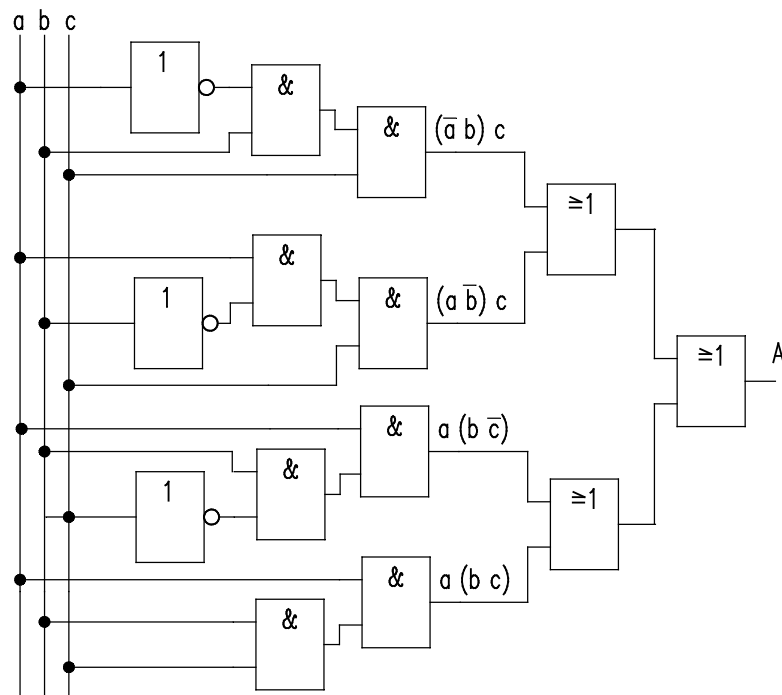


Abbildung 21 Realisierung einer Logikschaltung unter Verwendung von Gattern mit zwei Eingängen

3. Vereinfachung der Gleichung

Die ermittelte Funktionsgleichung muss so umgeformt werden, wenn möglich, dass eine der zuvor aufgeführten Vereinfachungsregeln angewandt werden kann.

$$A = \underbrace{(\bar{a}\bar{b}c)}_{1.} \vee \underbrace{(\bar{a}\bar{b}\bar{c})}_{2.} \vee \underbrace{(\bar{a}b\bar{c})}_{3.} \vee \underbrace{(abc)}_{4.} \quad \text{Term}$$

Bei dieser Gleichung kann das Distributivgesetz angewandt werden:

$$a(b \vee c) = (ab) \vee (ac) \quad a \vee (bc) = (a \vee b)(a \vee c)$$

Betrachtet man $b \vee c$ als eine Variable kann man $b \vee c$ beim 1. und 4. Term ausklammern:

$$(\bar{a}b\bar{c}) \vee (abc) = \bar{a}b(\bar{c} \vee c)$$

Laut Vereinfachungsregel gilt $\bar{c} \vee c = 1$ und somit:

$$(\bar{a}b\bar{c}) \vee (abc) = \bar{a}b(\bar{c} \vee c) = \bar{a}b1 = \bar{a}b$$

Diese Betrachtung kann aber nicht nur auf den 1. und 4. Term angewandt werden, sondern auch auf den 2. und 4. und ebenfalls auf den 3. und 4. Term. Die 4. UND-Verknüpfung bzw. der 4. Term muss 3-mal für eine Ausklammerung herangezogen werden. Da laut Regel

$$a \wedge a \wedge a = a \quad \text{bzw.} \quad a \vee a \vee a = a$$

gilt, kann man einen Term beliebig oft ergänzen. Die Gleichung wird dadurch nicht verändert.

Zur Vereinfachung wird die Gleichung des Beispiels um 2 Terme $a b c$ erweitert:

$$A = (\bar{a}bc) \vee (a\bar{b}c) \vee (ab\bar{c}) \vee (abc) \vee (abc) \vee (abc)$$

und umgestellt

$$A = (\bar{a}bc) \vee (abc) \vee (a\bar{b}c) \vee (abc) \vee (ab\bar{c}) \vee (abc)$$

Ausgeklammert wird

$$\quad \quad \quad bc \quad \quad \quad ac \quad \quad \quad ab$$

und man erhält

$$A = bc(\underbrace{\bar{a} \vee a}) \vee ac(\underbrace{\bar{b} \vee b}) \vee ab(\underbrace{\bar{c} \vee c})$$

$$\quad \quad \quad = 1 \quad \quad \quad = 1 \quad \quad \quad = 1$$

$$A = bc1 \vee ac1 \vee ab1$$

$$\underline{\underline{A = bc \vee ac \vee ab}}$$

In Abbildung 22 ist die Realisierung mit logischen Gattern abgebildet.

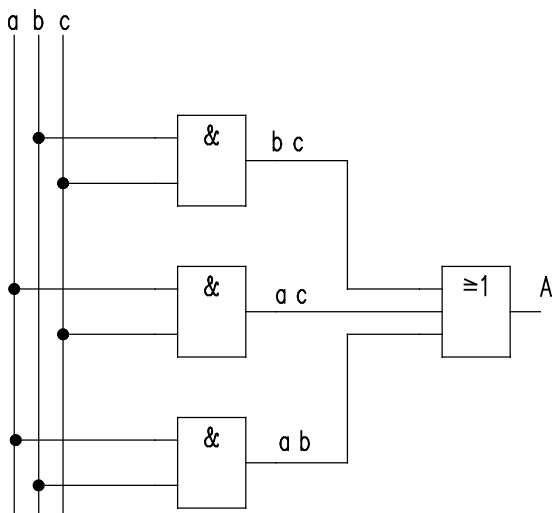


Abbildung 22 Vereinfachte Schaltung

4. Typisierung

Die abgebildete Schaltung ist mit NAND-Gattern zu realisieren.

$$A = bc \vee ac \vee ab$$

Eine Umformung der Gleichung ist unter Berücksichtigung zweier Regeln möglich:

- Eine doppelte Negation hat keine Wirkung

$$\overline{\overline{a}} = a \quad \text{oder} \quad \overline{\overline{a \vee b}} = a \vee b$$

- Inversionsgesetz bzw. De Morgansche Theoreme

Eine durchgehende Negation kann aufgelöst werden, indem sie auf die einzelnen Variablen verteilt wird und die betroffene Verknüpfungsart umgekehrt wird, d.h. aus ODER wird UND und umgekehrt.

$$\overline{a \vee b} = \overline{a} \wedge \overline{b}$$

$$\overline{ab \vee c} = \overline{a} \vee \overline{bc}$$

Bei der zweiten Gleichung hat sich ein Fehler eingeschlichen. Da die UND-Verknüpfung Vorrang vor der ODER-Verknüpfung hat, kann man die Ausgangsgleichung so formulieren

a UND b vor c

Diese Zuordnung darf auch beim Auflösen von Negationen nicht verloren gehen.

Oben steht aber nach Auflösung der Negation

$$\overline{b} \text{ UND } \overline{c} \text{ vor } \overline{a}.$$

Die Zuordnung hat sich geändert! Durch Setzen von Klammern kann man diese Fehlerquelle vermeiden. Die richtige Gleichung lautet:

$$\overline{(ab) \vee c} = (\overline{a} \vee \overline{b}) \overline{c}$$

Da die Gleichung für ein NAND-Gatter \overline{ab} lautet, muss die Ausgangsgleichung in diese Form gebracht werden.

Zunächst wird die rechte Seite doppelt negiert.

$$A = \overline{\overline{bc \vee ac \vee ab}}$$

Als Nächstes wird die untere Negation unter Berücksichtigung der NAND-Form aufgelöst.

$$A = \overline{\overline{bc} \overline{ac} \overline{ab}}$$

Nach diesem Schritt hat die Funktionsgleichung die richtige Form:

\overline{bc} und \overline{ac} und \overline{ab} sind NAND-Gatter mit zwei Eingängen. Die Ausgangsgröße A wird dann mit einem NAND-Gatter mit drei Eingängen gebildet.

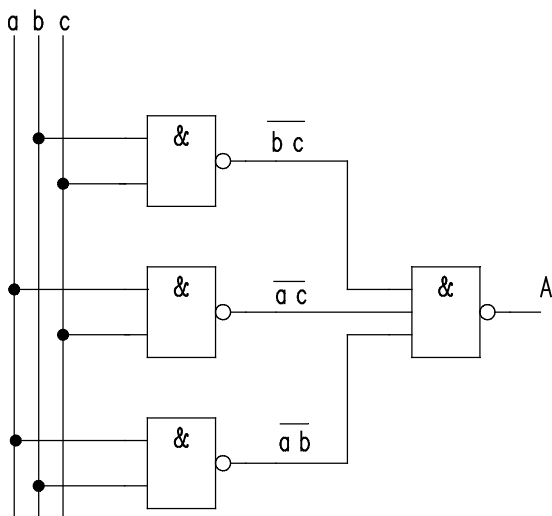


Abbildung 23 Schaltung mit NAND-Gattern

Eine andere Möglichkeit wäre die Umformung in die NOR-Gatter-Form.

$$A = bc \vee ac \vee ab$$

Im ersten Schritt sind die UND-Verknüpfungen umzuformen.

$$A = \overline{\overline{bc}} \vee \overline{\overline{ac}} \vee \overline{\overline{ab}}$$

Die untere Negation wird aufgelöst.

$$A = \overline{\overline{b} \vee \overline{c}} \vee \overline{\overline{a} \vee \overline{c}} \vee \overline{\overline{a} \vee \overline{b}}$$

Die UND-Verknüpfungen sind in NOR-Verknüpfungen umgewandelt worden. Die NOR-Gatter werden durch ODER verknüpft. Daher muss hier auch noch einmal doppelt negiert werden.

$$A = \overline{\overline{\overline{\overline{b} \vee \overline{c}} \vee \overline{\overline{a} \vee \overline{c}} \vee \overline{\overline{a} \vee \overline{b}}}}$$

Diese Gleichung ergibt die unten abgebildete Schaltung.

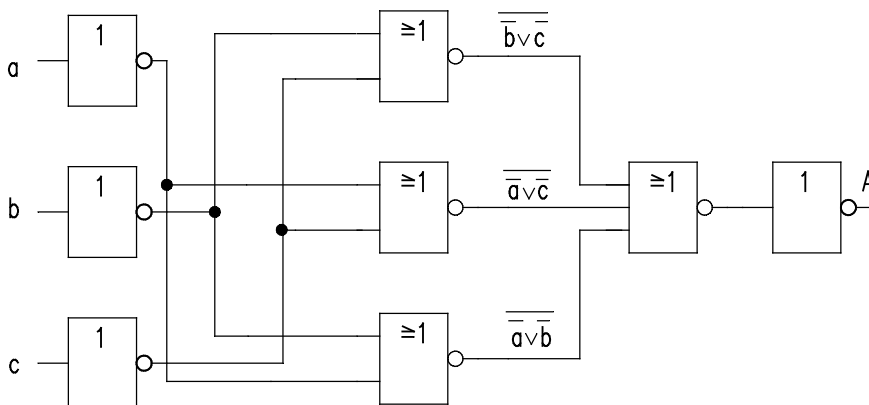


Abbildung 24 Logik-Schaltung mit NOR-Gattern

Da Inverter auch durch NOR-Gatter und NAND-Gatter realisierbar sind, können die eingezeichneten Inverter ebenfalls durch NOR-Gatter ersetzt werden.

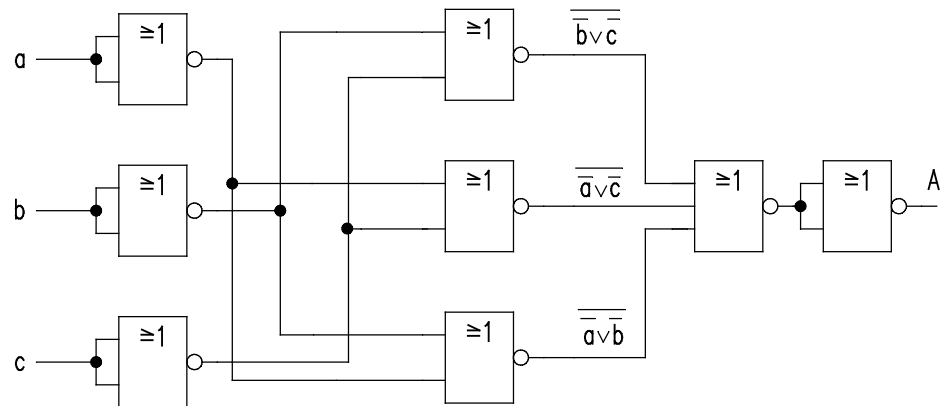


Abbildung 25 Logik-Schaltung nur mit NOR-Gattern realisiert

Aufgaben

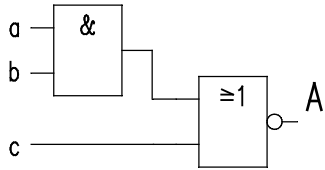
Aufgabe 1

Im Zusammenhang mit einem Fahrstuhl ergibt sich folgende Wahrheitstabelle für die Sicherheitsschaltung des Fahrstuhls:

a	b	c	A
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

- 1.1 Ermitteln Sie die Funktionsgleichung!
- 1.2 Geben Sie den Schaltplan unter Verwendung der Grundglieder UND, ODER und NICHT an!
- 1.3 Realisieren Sie den Schaltplan nur mit NAND-Gattern!
- 1.4 Realisieren Sie den Schaltplan nur mit NOR-Gattern!

Aufgabe 2

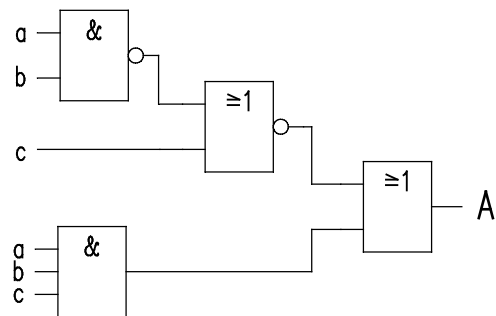


2.1 Stellen Sie die Funktionsgleichung auf!

2.2 Stellen Sie die Schaltung nur mit NOR-Gattern dar!

2.3 Stellen Sie die Schaltung nur mit NAND-Gattern dar!

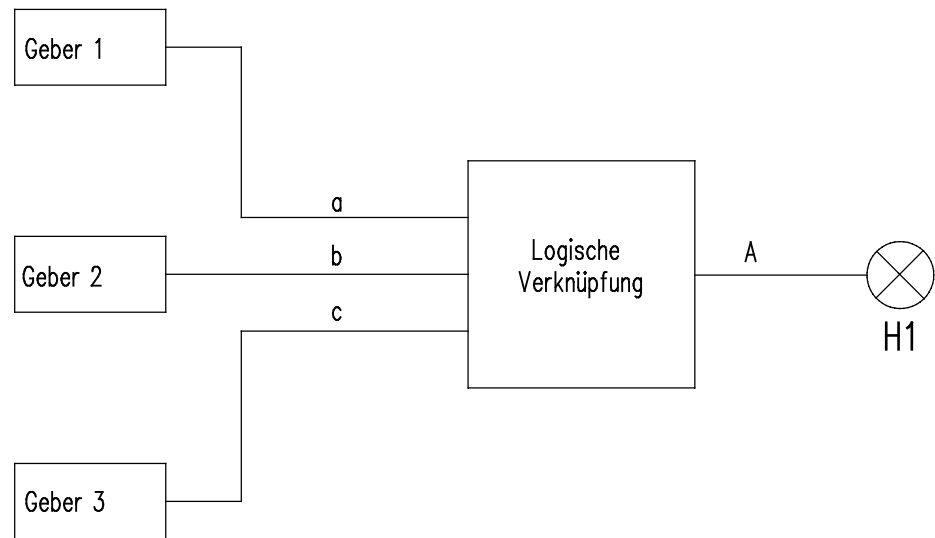
Aufgabe 3



3.1 Stellen Sie die Funktionsgleichung der abgebildeten Schaltung auf!

3.2 Vereinfachen Sie die Schaltung!
(Lösen Sie zuerst die Negationen auf!)

Aufgabe 4



Die Signallampe H1 soll nur dann leuchten, wenn kein Geber oder nur ein Geber 1-Signal führt.

- 4.1 Stellen Sie die Wahrheitstabelle auf!
- 4.2 Ermitteln Sie die Funktionsgleichung!
- 4.3 Vereinfachen Sie die Gleichung!
- 4.4 Zeichnen Sie die Schaltung, wenn nur NOR-Gatter verwendet werden!

2 Schaltungen mit Speicherverhalten

Lernbereich

2.1 Flip-Flop

Mit Flip-Flop oder bistabile Kippstufe wird eine elektronische Schaltung bezeichnet, die zwei stabile Zustände besitzt. Sie kann durch entsprechende Eingangssignale von einem Zustand (gesetzt) in den anderen (rückgesetzt) geschaltet werden. Auf diese Weise speichern Flip-Flops binäre Informationen (1-Bit).

Flip-Flops lassen sich durch rückgekoppelte Logik-Gatter realisieren. Die Rückkopplung bewirkt, dass der Logikzustand der Ausgangsvariablen nicht nur vom aktuellen Logikzustand der Eingangsvariablen, sondern auch vom Zustand des Flip-Flops vor der Änderung der Eingangsvariablen bestimmt wird. Diese Zeitabhängigkeit (Vorgeschichte) muss daher zusätzlich in den Verknüpfungstabellen berücksichtigt werden.

Dazu wird die rückgekoppelte Ausgangsvariable zu einem Zeitpunkt t_n betrachtet und mit Vorherzustand Q_n bezeichnet. Der sich zu einem etwas späteren Zeitpunkt t_{n+1} auf Grund der logischen Verknüpfung ergebende Ausgangszustand wird Nachherzustand Q_{n+1} genannt.

Alle Steuerschaltungen, die mit einer zeitlichen Abfolge arbeiten, enthalten Flip-Flops. Zur Beschreibung der Flip-Flop-Funktion werden vielfach Impulsdiagramme verwendet, da sich dann das zeitabhängige Verhalten besonders deutlich darstellen lässt.

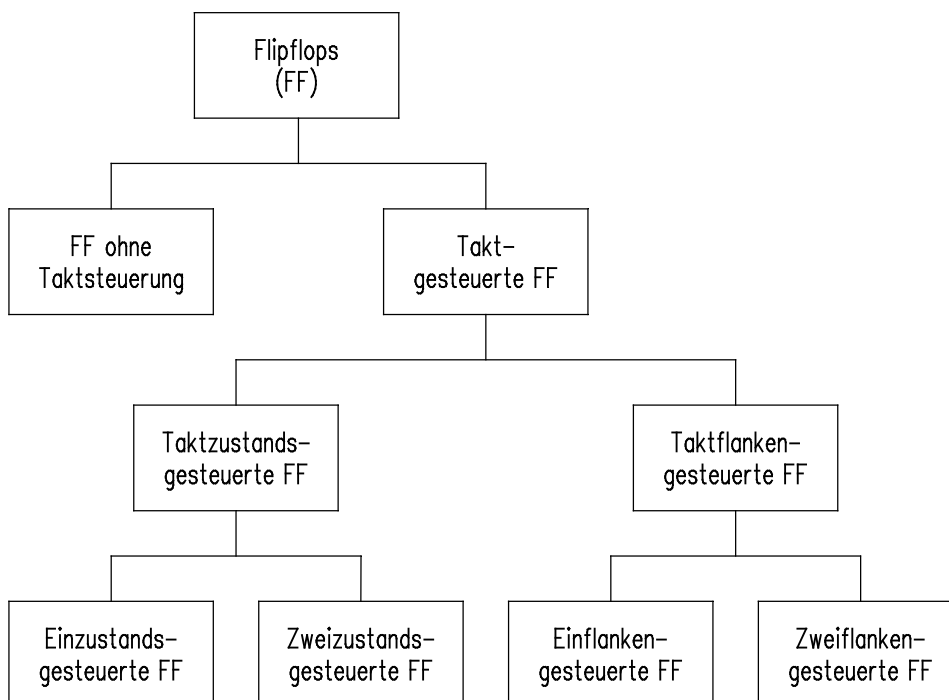
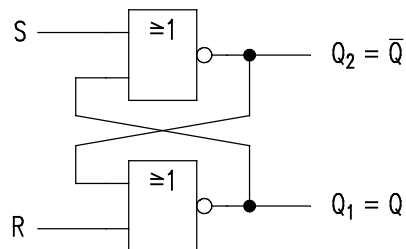


Abbildung 26 Steuerungsarten von Flip-Flops

Je nach Ansteuerungsart und Aufbau lassen sich Flip-Flops mit unterschiedlicher Wirkungsweise realisieren. Bedeutung erlangt und als integrierte Schaltung erhältlich sind RS-, D-, T-, und JK-Flip-Flops in verschiedenen Ansteuerungsvarianten (siehe Abbildung 26).

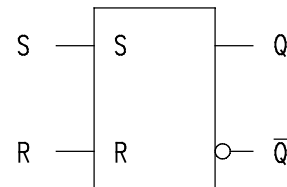
Ungetaktetes Flip-Flop

Der einfachste Flip-Flop-Typ ist das ungetaktete RS-Flip-Flop (Basis-Flip-Flop), das durch Rückkopplung zweier NOR-Gatter entsteht.



ausführliche Schaltung

Abbildung 27 RS-Flip-Flop (NOR-Technik)



Schaltsymbol

Es besitzt die beiden Eingänge S (Set) und R (Reset) sowie den Ausgang Q_1 und den komplementären Ausgang Q_2 . Unter Berücksichtigung des Vorherzustandes ergibt sich folgende Übergangstabelle für den Ausgang Q_1 :

Fall	S	R	Q_{1n}	Q_{1n+1}
1	0	0	0	0
2	0	0	1	1
3	0	1	0	0
4	0	1	1	0
5	1	0	0	1
6	1	0	1	1
7	1	1	0	-
8	1	1	1	-

Tabelle 12 Ausführliche Veknüpftabelle RS-Flip-Flop

Die Übergangstabelle zeigt, dass

- bei $S = 0$ und $R = 0$ der ursprüngliche Ausgangszustand erhalten (**gespeichert**) bleibt, d.h. $Q_{1n} = Q_{1n+1}$;
- bei $S = 0$ und $R = 1$ unabhängig vom Vorherzustand Q_{1n} der Ausgang Q_1 den Zustand „0“ annimmt, d.h. das Flip-Flop **zurückgesetzt** wird;
- bei $S = 1$ und $R = 0$ unabhängig vom Vorherzustand Q_{1n} der Ausgang Q_1 den Zustand „1“ annimmt, d.h. das Flip-Flop **gesetzt** wird;
- bei $S = 1$ und $R = 1$ kein Ausgangszustand definiert ist. Diese Eingangskombination ist **nicht zulässig**, da bei $S = R = 1$ Ausgang Q_1 und der Komplementärgang Q_2 gleichzeitig auf „0“ gehen (Widerspruch). Zudem lässt sich nicht mehr vorhersagen, welche Ausgangszustände sich einstellen, wenn S und R anschließend gleichzeitig „0“ werden (Ausgangszustand abhängig von Unsymmetrien im Gatteraufbau).

Wird die Bedingung S und R nicht gleichzeitig „1“ eingehalten, lässt sich für ein RS-Flip-Flop nach Abbildung 27 eine einfache Übergangstabelle angeben:

S	R	Q_{n+1}	
0	0	Q_n	Vorherzustand bleibt erhalten (speichern)
0	1	0	rücksetzen
1	0	1	setzen
1	1	-	verbotener Fall

Tabelle 13 Übergangstabelle RS-Flip-Flop

Typischer Anwendungsfall ungetakteter RS-Flip-Flops ist das Entprellen mechanischer Tasten. Das mehrmalige Abprallen der Kontaktfläche vom Gegenkontakt lässt dicht aufeinander folgende Störimpulse entstehen. Eine nachgeschaltete Zählstufe würde an Stelle eines gewünschten Impulses die zusätzlichen Störimpulse mitregistrieren.

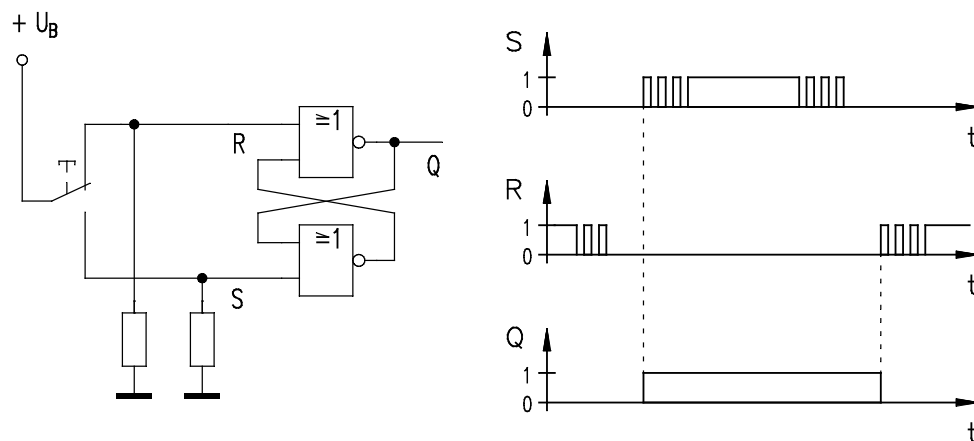


Abbildung 28 Prellfreier Taster

Mit der oben abgebildeten Schaltung wird das Prellen unterdrückt, da schon der erste Impuls das Flip-Flop setzt ($Q = 1$). Erst durch Loslassen des Tasters und dem damit verbundenen Rücksetzen des Flip-Flops geht das Ausgangssignal auf L-Pegel zurück. Es entsteht ein sauberer Zählimpuls, dessen Länge nur von der Betätigungsdauer des Tasters bestimmt wird.

Die Anwendung der De-Morganschen Regeln ermöglicht auch den Aufbau eines RS-Flip-Flops ausschließlich mit NAND-Gattern.

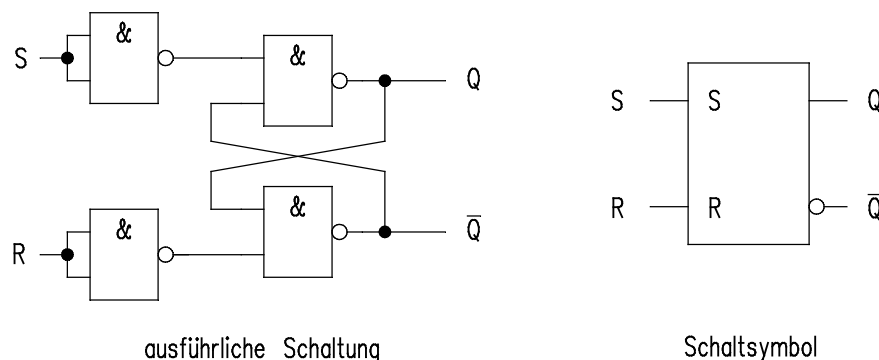
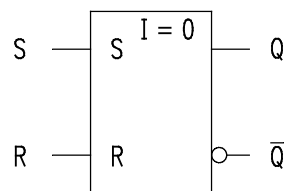


Abbildung 29 RS-Flip-Flop mit NAND-Gattern

In einigen Anwendungsfällen kann mit den komplementären Eingangssignalen \bar{R} und \bar{S} gearbeitet werden, z.B. beim prellfreien Taster (Tasterpol auf Masse legen, Widerstände nach $+U_B$). Dann lassen sich zwei NAND-Gatter einsparen. Entsprechende integrierte Schaltkreise werden auch von der Industrie angeboten, z.B. 74LS279, ein vierfach $\bar{R}\bar{S}$ -NAND-Flip-Flop. Bei diesen Flip-Flops ist zu beachten, dass der verbotene Fall dann vorliegt, wenn an beiden Eingängen L-Pegel ansteht.

Die als integrierte Schaltung angebotenen RS-Flip-Flops besitzen in der Regel eine Vorzugslage (Grundstellung), die sich nach dem Einschalten der Versorgungsspannung einstellt.



Schaltsymbol

Abbildung 30 RS-Flip-Flop mit Grundstellung

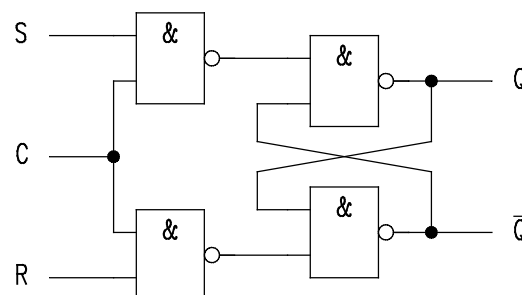
RS-Flip-Flops mit Grundstellung können durch eine Eintragung im Schaltzeichen gekennzeichnet werden.

In der nebenstehenden Abbildung ist das Flip-Flop mit $I = 0$ gekennzeichnet, d.h. nach dem Einschalten (Initialisierung) nimmt der Ausgang Q den Zustand „0“ (\bar{Q} entsprechend den Zustand „1“) an. Das RS-Flip-Flop ist damit nach dem Einschalten zurückgesetzt.

Einzustandsgesteuerte Flip-Flops

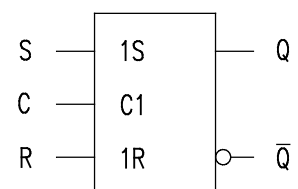
Einzustandsgesteuerte Flip-Flops reagieren nur zu einer bestimmten Zeit auf angelegte Eingangssignale. Sie besitzen einen weiten Eingang, den Steuereingang C. Nur während der Zeit, in der das Taktsignal den Zustand „1“ aufweist, wirkt sich eine Änderung der Eingangsvariablen, entsprechend der zuständigen Übergangstabelle, auf den Ausgang aus. Während der „1“-Phase des Taktsignals kann sich der Zustand der Ausgangsvariablen mehrfach ändern. Erst wenn das Taktsignal den Zustand „0“ annimmt, ist das Flip-Flop verriegelt und der aktuelle Zustand der Ausgangsvariablen bleibt gespeichert.

Einzustandsgesteuerte RS-Flip-Flops sind mit vier NAND-Gattern entsprechend nachstehender Abbildung realisierbar.



ausführliche Schaltung

Abbildung 31 Einzustandsgesteuertes RS-Flip-Flop



Schaltsymbol

S und R sind jetzt sog. Vorbereitungseingänge, da sie sich nicht mehr unmittelbar auswirken können. Für $C = 1$ verhält sich das Flip-Flop wie ein Basis RS-Flip-Flop und der zugehörigen Übergangstabelle (Tabelle 13). Das Impulsdiagramm verdeutlicht noch einmal die Wirkungsweise dieses Flip-Flops.

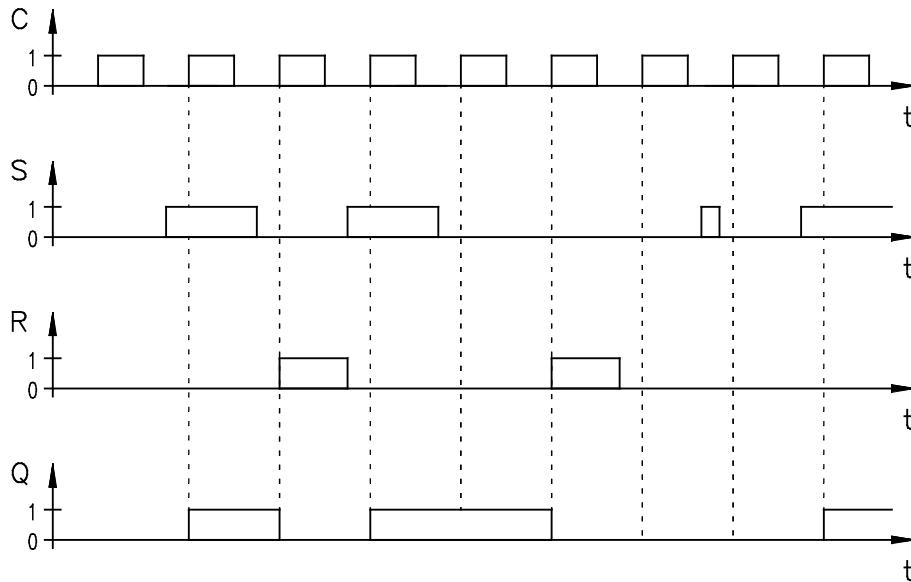


Abbildung 32 Impulsdiagramm eines einzustandsgesteuerten RS-Flip-Flops

Eine häufig verwendete Variante des einzustandsgesteuerten RS-Flip-Flops ist das **einzustandsgesteuerte D-Flip-Flop**. Es dient zur Speicherung einer Variablen, meist dem Zustand einer Datenleitung D. Dazu wird einfach das am Setzeingang S anliegende Signal D mit einem NICHT-Gatter invertiert und auf den Rücksetzeingang R gegeben (Abbildung 33).

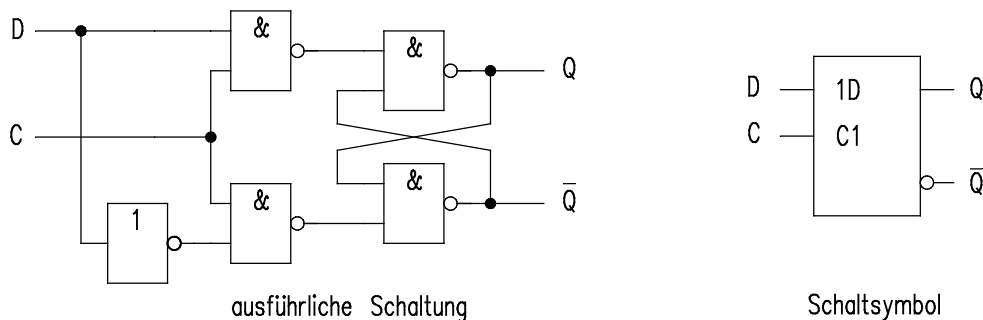


Abbildung 33 Einzustandsgesteuertes D-Flip-Flop

Bei der so entstandenen Speicherzelle (Data-Latch) hat der Ausgang Q den gleichen Zustand wie das Eingangssignal D, solange der Takt $C = 1$ ist. Geht das Taktsignal C auf „0“, wird der aktuelle Ausgangszustand gespeichert.

C	D	Q_{n+1}
0	0	Q_n
0	1	Q_n
1	0	0
1	1	1

Tabelle 14 Übergangstabelle D-Flip-Flop

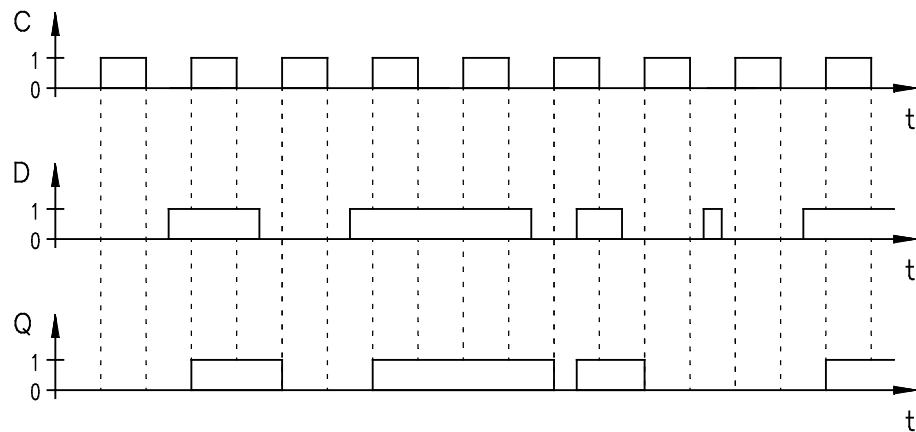


Abbildung 34 Impulssdiagramm eines einzustandsgesteuerten D-Flip-Flops

Typischer Anwendungsfall ist die Adressen- oder Datenspeicherung in Mikrocomputersystemen. Da diese Systeme byte-orientiert sind, werden vorzugsweise integrierte Schaltungen eingesetzt, die acht D-Flip-Flops enthalten. Der TTL-Baustein 74LS373 ist ein solches 8-bit-D-Latch. Diese D-Flip-Flops sind mit Tristate Ausgängen versehen, um auch den Einsatz an Busleitungen zu ermöglichen.

Einflankengesteuerte Flip-Flops

Flip-Flops, bei denen die Vorbereitungseingänge (RS, JK, D, T) nur während einer auslösenden Taktflanke wirksam werden, bezeichnet man als einflankengesteuert. Positiv flankengesteuert sind Flip-Flops, die auf den 0-1 Übergang (\uparrow) des Taktsignals reagieren. Entsprechend negativ flankengesteuert sind Flip-Flops, die beim 1-0 Übergang (\downarrow) des Taktes schalten. Direkt nach der jeweiligen auslösenden Flanke stellt sich, entsprechend den Pegeln an den Vorbereitungseingängen, der Ausgangszustand des Flip-Flops ein. Er bleibt dann bis zur nächsten schaltenden Flanke unverändert.

In den Übergangstabellen ist Q_n der Ausgangszustand vor, und Q_{n+1} der Ausgangszustand nach der steuernden Flanke. Damit gelten auch für das **einflankengesteuerte RS-Flip-Flop** die Übergangsbedingungen (siehe Tabelle 13) des ungetakteten RS-Flip-Flops. Sie werden jedoch erst mit der auslösenden Flanke wirksam. Die Schaltsymbole eines positiv und eines negativ flankengesteuerten RS-Flip-Flops sind in der folgenden Abbildung dargestellt. Die Wirkungsweise des einflankengesteuerten RS-Flip-Flops verdeutlicht das Impulssdiagramm.

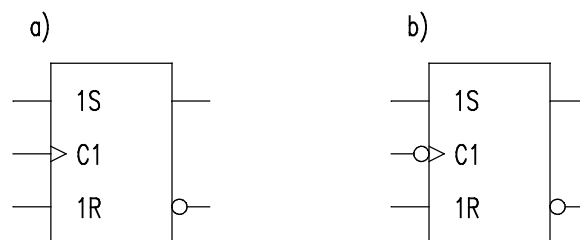


Abbildung 35 Positiv (a) und negativ (b) flankengesteuertes RS-Flip-Flop

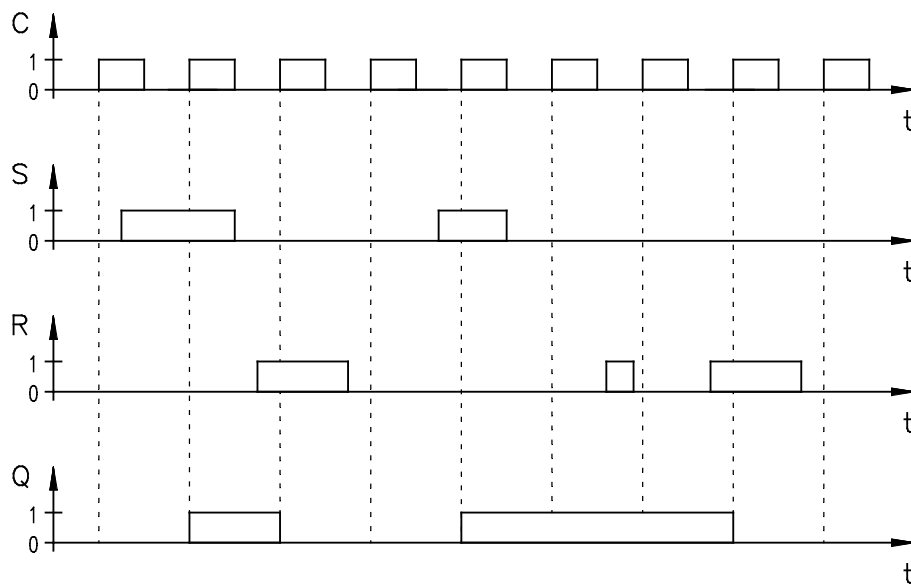


Abbildung 36 Impulsdiagramm eines positiv flankengesteuerten RS-Flip-Flops

Durch UND-Verknüpfung der RS-Eingänge mit den zurückgekoppelten Ausgängen entsteht aus dem RS-Flip-Flop das **einflankengesteuerte JK-Flip-Flop**. Auch für sie gilt die Übergangstabelle der JK-Flip-Flops.

In integrierter Form werden JK-Flip-Flops vorzugsweise mit negativer Flankensteuerung angeboten. Die moderneren negativ einflankengesteuerten JK-Flip-Flops können somit in vielen Schaltungen die Funktion der älteren zweizustandsgesteuerten JK-Flip-Flops (Master-Slave) übernehmen.

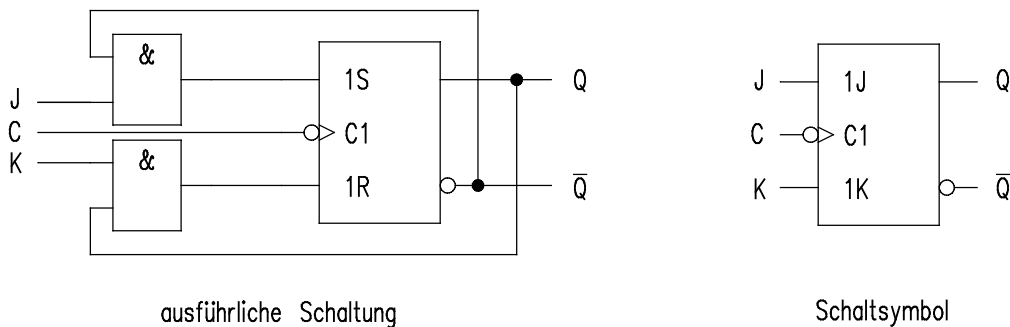


Abbildung 37 Prinzipschaltung und Symbol eines negativ flankengesteuertes JK-Flip-Flops

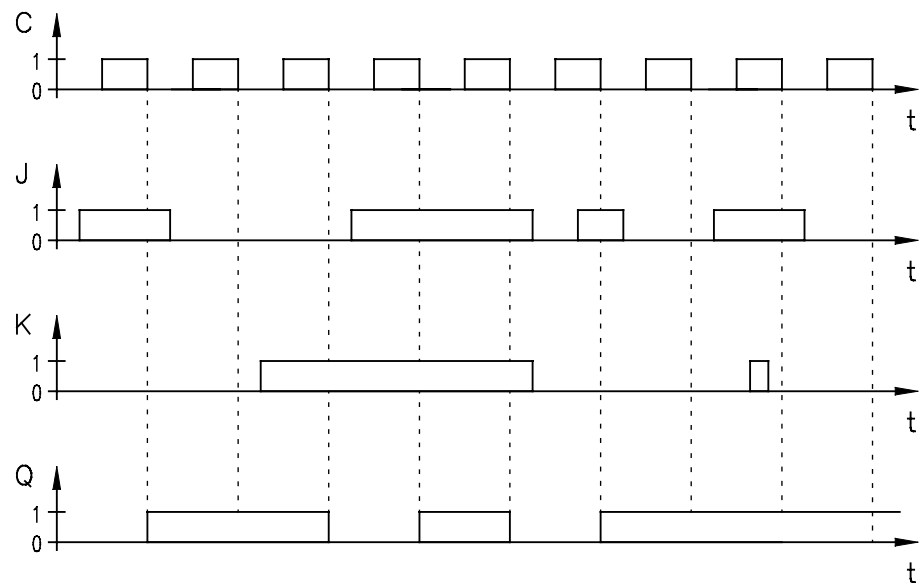


Abbildung 38 Impulsdiagramm eines negativ flankengesteuerten JK-Flip-Flops

Bei integrierten JK-Flip-Flops sind oft zusätzliche Setz (Preset)- und Rücksetz (Clear)-Eingänge realisiert. Sie haben höhere Priorität als die taktabhängigen Eingänge J und K, sodass ein taktunabhängiges Setzen bzw. Rücksetzen des Flip-Flops jeder Zeit möglich ist.

Beispiele für IC-Typen in TTL-Technik:

74LS103 zwei negativ flankengesteuerte JK-FFs mit Clear

74LS112 zwei negativ flankengesteuerte JK-FFs mit Preset und Clear

Das JK-Flip-Flop ist das universellste flankengesteuerte Flip-Flop. Einerseits sind alle vier Eingangskombinationen zulässig (verbotener Fall tritt nicht auf). Andererseits lassen sich aus dem JK-Flip-Flop durch einfache Beschaltung die wichtigen Varianten D-Flip-Flop und T-Flip-Flop erzeugen.

Das **einflankengesteuerte D-Flip-Flop** entsteht durch Inversion des K-Einganges entsprechend Abbildung 39. Damit sind nur zwei Eingangskombinationen möglich ($J = 1, K = 0$ und $J = 0, K = 1$) und es ergibt sich folgende, vereinfachte Übergangstabelle:

C	D	Q_{n+1}
↓	1	1
↓	0	0

Tabelle 15 Übergangstabelle
D-Flip-Flop (negativ
flankengesteuert)

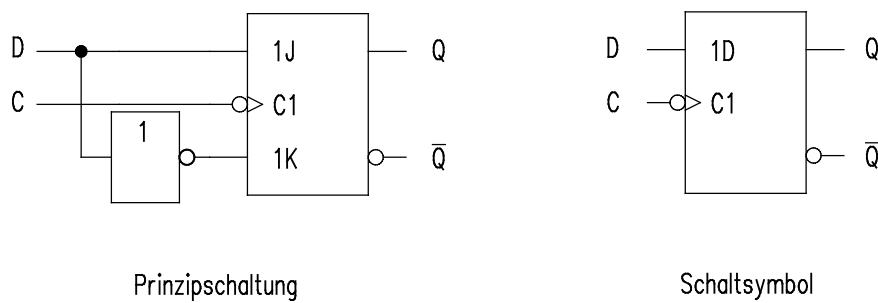


Abbildung 39 Einflankengesteuertes D-Flip-Flop

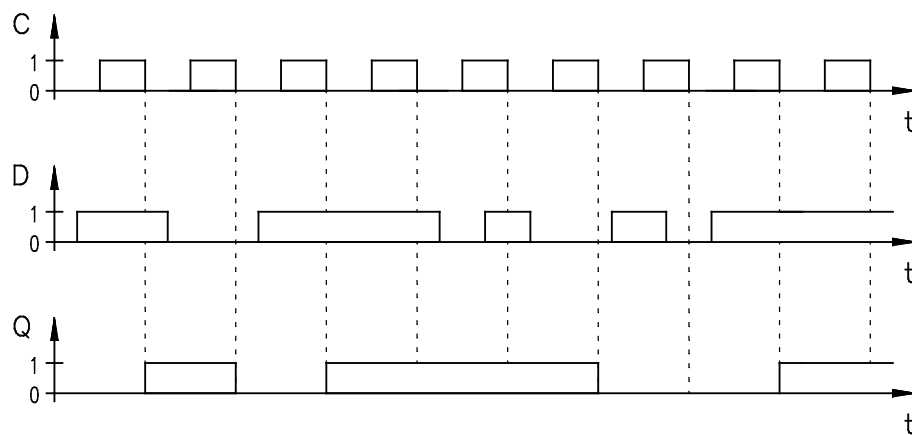


Abbildung 40 Impulsdiagramm eines negativ flankengesteuerten D-Flip-Flops

Einflankengesteuerte D-Flip-Flops werden vielfach in Digitalschaltungen verwendet. Sie kommen als Schieberegister, Datenspeicher und zur Datensynchronisation zum Einsatz.

Ein anderer Flip-Flop Typ, das **einflankengesteuerte T-Flip-Flop**, lässt sich ebenfalls mit einem flankengesteuerten JK-Flip-Flop realisieren. Dazu werden die Eingänge J und K einfach mit einander verbunden und mit T bezeichnet (siehe Abbildung).

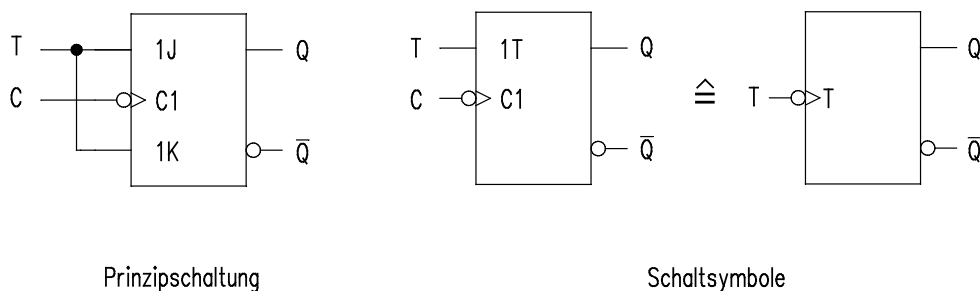


Abbildung 41 Einflankengesteuertes T-Flip-Flop

C	T	Q_{n+1}
↓	1	\bar{Q}_n
↓	0	Q_n

Tabelle 16 Übergangstabelle T-Flip-Flop (negativ flankengesteuert)

Wie die vereinfachte Übergangstabelle erkennen lässt, wird für $T = 1$ bei jeder fallenden Flanke der Vorherzustand invertiert. Ist T statisch mit „1“ beschaltet und wird das T-Flip-Flop mit einem Taktsignal C ansteuert, so entsteht ein Ausgangssignal Q mit der halben Frequenz des Taktsignals (Abbildung 42). Damit eignen sich T-Flip-Flops zur Realisation von Frequenzteilern und Dual-Zählern.

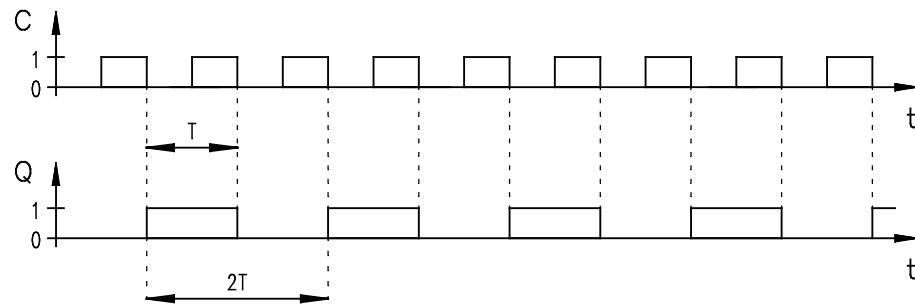


Abbildung 42 Impulsdiagramm eines einflankengesteuerten T-Flip-Flops

2.2 Zähler

Eine typische Aufgabe in der Datenverarbeitung ist das Zählen von Impulsen. Binär arbeitende Zählerschaltungen, sog. Dualzähler, sind dafür besonders geeignet. Sie lassen sich mit einfachen Flip-Flop-Schaltungen realisieren. Zwei Gruppen von Zählern werden grundsätzlich unterschieden: synchrone und asynchrone Zähler.

Bei synchronen Zählern werden alle Flip-Flops parallel an ein gemeinsames Takt-signal (Zählimpulse) angeschlossen. Die Flip-Flop- bzw. Zählerausgänge ändern sich daher gleichzeitig (synchron). Asynchrone Zähler sind durch eine Folgeschaltung von Flip-Flop-Stufen gekennzeichnet. Die Zählimpulse wirken nur auf den Takteingang des ersten Flip-Flops, während die Folgestufen von dem Ausgangssignal der vorhergehenden Stufe getaktet werden. Die Ausgänge ändern sich daher zu unterschiedlichen Zeiten (asynchron).

Asynchroner Dualzähler

Durch Kaskadierung von flankengesteuerten T-Flip-Flops, bzw. entsprechend verschalteten JK-Flip-Flops, lässt sich ein asynchroner Dualzähler realisieren. Für einen n-bit Zähler werden n Stufen bzw. T-Flip-Flops benötigt. Die folgende Abbildung zeigt einen asynchronen 3-bit Vorwärtszähler.

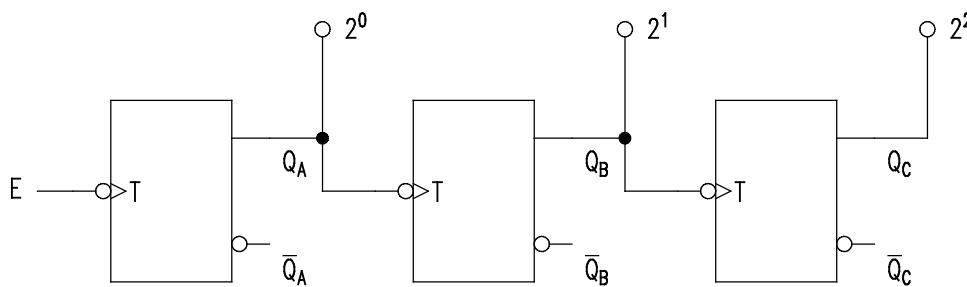
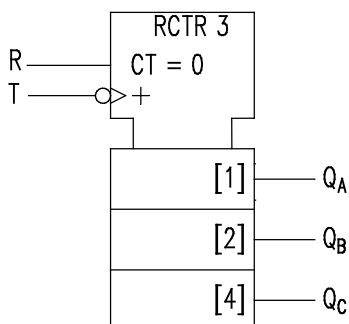


Abbildung 43 Schaltung eines 3-bit asynchronen Vorwärts-Dualzählers



RCTR n : Asynchroner n-Bit Dualzähler
(Ripple-Counter)

CT : Inhalt (Content)

CT = 0 : Zählerstand = 0 (Rücksetzein-gang)

+ : Vorwärtszähler
(- : Rückwärtszähler)

[i] : Wertigkeit des Ausganges

Abbildung 44 Symbol eines 3-bit asynchronen Vorwärts-Dualzählers

Jede Flip-Flop-Stufe arbeitet als Frequenzteiler mit dem Teilungsfaktor 2. Werden die binären Wertigkeiten 2^0 , 2^1 , 2^2 den Ausgängen Q_A , Q_B , Q_C zugeordnet, so lässt sich an den Ausgängen das jeweilige duale Zählergebnis ablesen. Das Impulsdiagramm veranschaulicht diesen Zusammenhang.

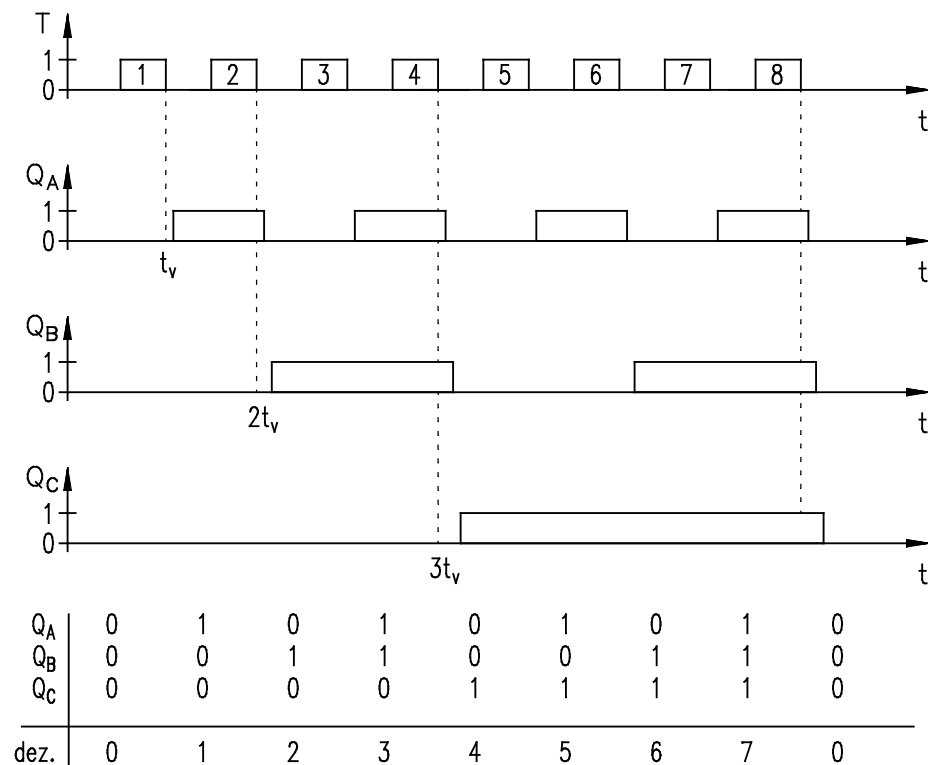


Abbildung 45 Impulssdiagramm eines 3-bit asynchronen Vorwärts-Dualzählers

Erkennbar ist auch die zunehmende Signalverzögerung auf Grund der Signallaufzeit t_v jeder Flip-Flop-Stufe. Sie beträgt bei Flip-Flops in TTL-Technik ca. 30-50 ns. Für einen n-bit asynchronen Zähler ergibt sich eine Gesamtverzögerungszeit:

$$t_{vg} = n \cdot t_v$$

Dieser Nachteil der Asynchronzähler führt zu einer Begrenzung der Zählfrequenz, die mit steigender Stufenzahl immer geringer wird.

Mit dem gleichen Prinzip lassen sich auch asynchrone Rückwärtszähler realisieren. Der eigentliche Unterschied besteht in der Verschaltung der T-Flip-Flops miteinander. Beim Rückwärtszähler ist der **negierte** Flip-Flop-Ausgang mit dem Takteingang der Folgestufe verbunden.

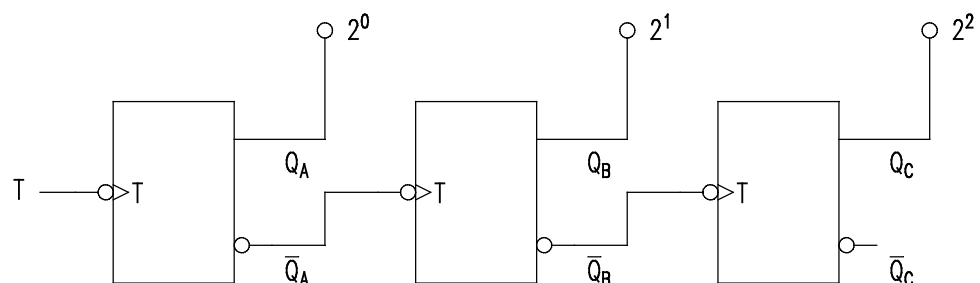


Abbildung 46 Schaltung eines 3-bit asynchronen Rückwärts-Dualzählers

Damit der Rückwärtszähler von seinem Höchststand (alle Ausgänge auf „1“) beginnend herunter zählen kann, ist der Einsatz von JK-Flip-Flops mit zusätzlichen taktunabhängigen Setzeingängen (Preset) sinnvoll. Besitzen die Flip-Flops auch taktunabhängige Rücksetzeingänge (CLR), wie z.B. beim IC 74LS76, lassen sich auch andere Anfangszustände einstellen.

Die Abbildung zeigt das Impulsdigramm eines asynchronen 3-bit Rückwärtszählers bei Voreinstellung auf 111 (Signalverzögerung vernachlässigt).

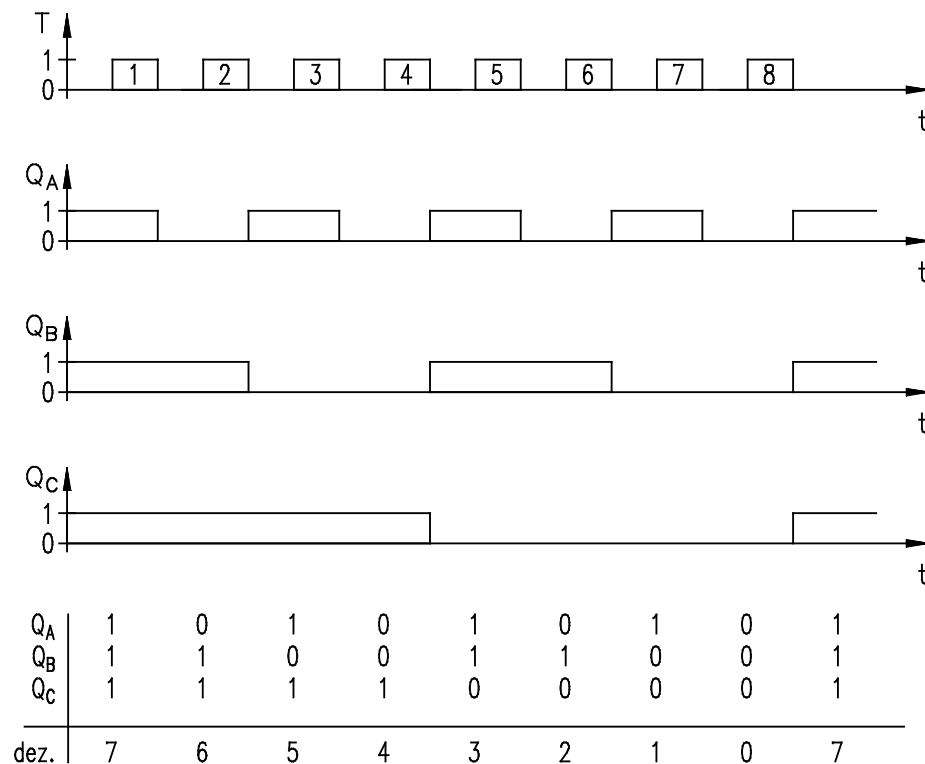


Abbildung 47 Impulsdigramm eines 3-bit asynchronen Rückwärts-Dualzählers

Als integrierte Schaltkreise werden meist 4-bit Zähler angeboten. Sind höhere Zählkapazitäten erforderlich, lassen sich asynchrone Zähler auf einfache Weise kaskadieren. Dazu werden wiederum alle Flip-Flops hintereinander geschaltet, d.h. der höchstwertige Ausgang eines Zählerbausteins wird als Taktsignal des Folgezählers eingesetzt (s. Abbildung).

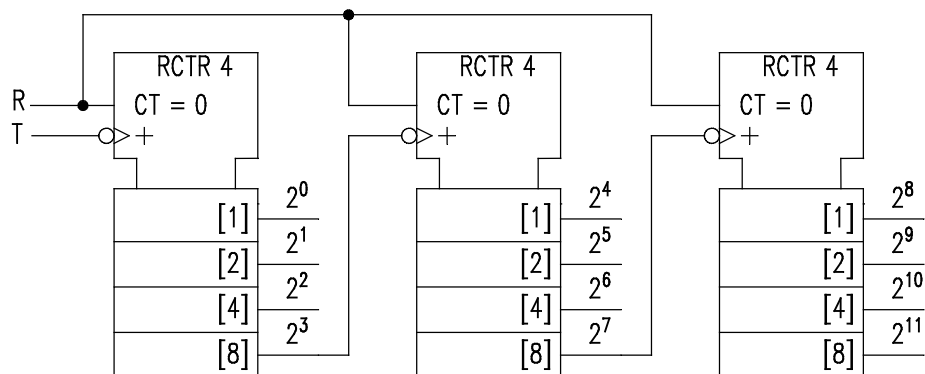


Abbildung 48 Realisierung eines 12-bit asynchronen Zählers

Bei einigen Zählaufgaben werden Zähler benötigt, die die Zählkapazität nicht voll nutzen. Sie sollen nur bis zu einem gewünschten Zahlenwert ($m-1$) zählen und dann zurückgesetzt werden. Diese Zähler werden als **Modulo-m-Zähler** bezeichnet.

Typisches Beispiel ist ein dekadischer Zähler (Modulo-10-Zähler), der von 0 bis 9 zählt und dann wieder bei 0 beginnt. Ein Modulo-m-Zähler lässt sich mithilfe eines Schaltnetzes aus einem Dualzähler realisieren.

Das Schaltnetz hat die Aufgabe, die Ausgänge des Dualzählers zu dekodieren und beim Zählerstand $m = 2^1 + 2^3 = 2 + 8 = 10$ einen Rücksetzimpuls zu erzeugen (s. Abbildung).

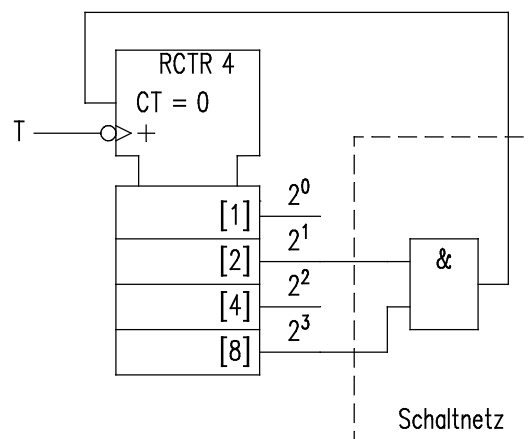


Abbildung 49 Prinzip und Schaltung eines asynchronen Modulo-m-Zählers ($m = 10$)

Synchrone Zähler

Bei hohen Zählfrequenzen sind Asynchronzähler auf Grund ihrer typischen Verzögerungszeit ungeeignet. Synchronzähler sind dadurch charakterisiert, dass alle Zählstufen gleichzeitig getaktet werden. Der Schaltungsaufwand vergrößert sich jedoch gegenüber den asynchronen Zählern, da eine zusätzlich Übertragslogik notwendig wird. Die folgende Abbildung zeigt den prinzipiellen Aufbau eines **synchronen Dualzählers**.

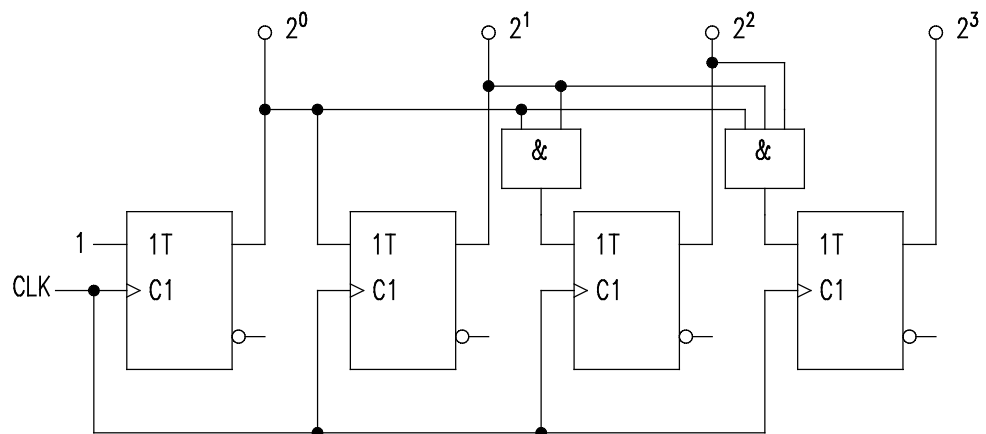


Abbildung 50 Prinzip eines synchronen Dualzählers

Integrierte synchrone Dualzähler haben meist nur eine Zählkapazität von 4-bit. Sie besitzen allerdings weitere Steuereingänge bzw. -ausgänge

- zum Laden des Zählers mit einer beliebigen Zahl ($\overline{\text{LOAD}}$)
- zum Löschen (Rücksetzen) des Zählers ($\overline{\text{CLR}}$)
- und zur einfachen Kaskadierung (ENT , ENP , RCO)

Damit lassen sich mehrere IC's für die Erhöhung der Zählerkapazität ohne zusätzliche externe Gatter zusammenschalten.

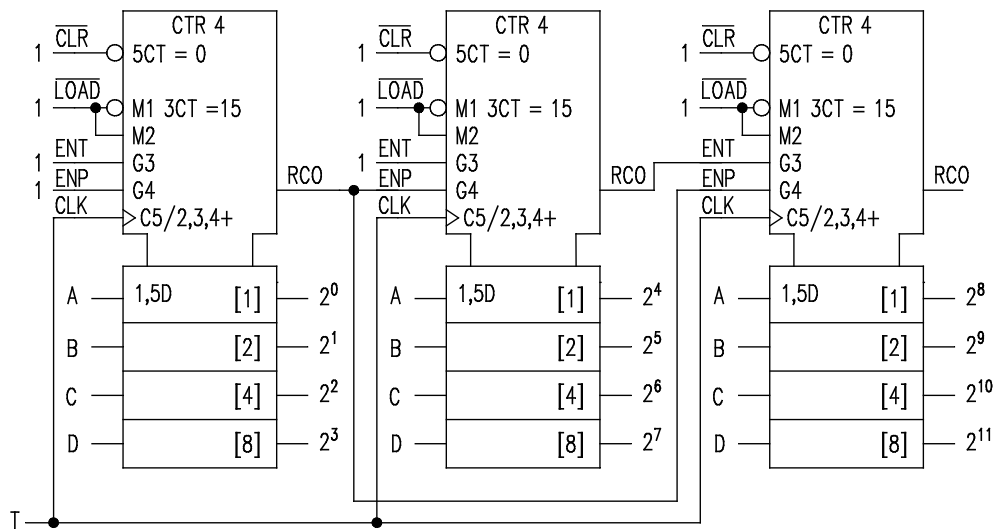


Abbildung 51 12-bit synchroner Dualzähler mit drei IC's 74HC163

Die gängigen Zählertypen werden als integrierte Schaltung sowohl in TTL- als auch in CMOS-Technik angeboten. In der folgenden Tabelle sind einige Beispiele zusammengefasst.

Zählertyp	TTL-Technik	CMOS	Kapazität
async. Dual	74LS293		4-bit
async. Modulo 10 (BCD)	74LS390	74HC390	2 Dekaden
async. Dual	74LS393	74HC393	2 x 4-bit
async. Dual		4521	24-bit
sync. Modulo 10 (BCD)	74LS162	74HC162, 4160	1 Dekade
sync. Dual	74LS163	74HC163, 4163	4-bit
sync. Modulo 10 (BCD) vorw./rückw.	74LS190	74HC190, 4510	1 Dekade
sync. Dual vorw./rückw.	74LS191	74HC191, 4516	4-bit

Tabelle 17 Zählertypen

2.3 Schieberegister

Einige Aufgaben in der Digitaltechnik, z.B. die serielle Datenübertragung oder die Multiplikation/Division von Dualzahlen, erfordert die Erzeugung von Bit-Folgen bzw. ein Verschieben von binären Informationen.

Hierfür werden Schieberegister, hintereinander geschaltete 1-bit-Speicher, eingesetzt. Sie ermöglichen es, Informationen taktgesteuert von einem Speicher zum anderen weiterzugeben. Die einzelnen Speicher bestehen aus Flip-Flops. Vorzugsweise werden flankengesteuerte D-Flip-Flops, bzw. aus RS- oder JK-Flip-Flops realisierte D-Flip-Flops, verwendet.

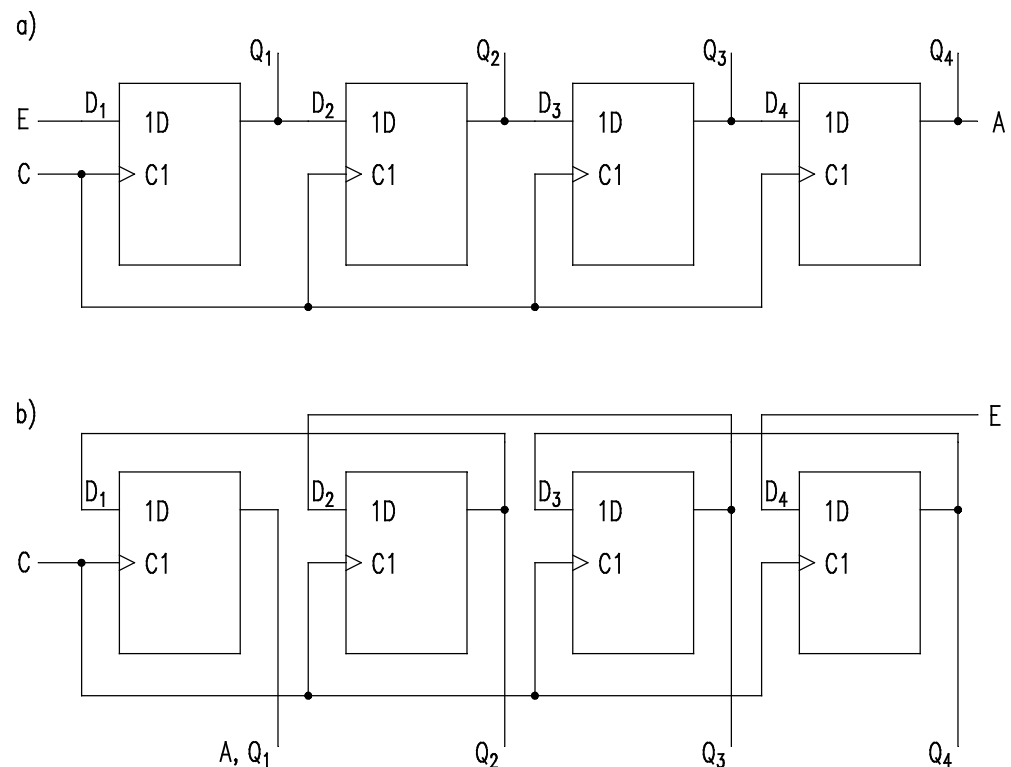


Abbildung 52 Prinzipieller Aufbau eines 4-bit Schieberegisters mit D-Flip-Flops und Schieberichtung rechts (a) bzw. Schieberichtung links (b)

Die obige Abbildung zeigt ein einfaches 4-bit Schieberegister mit serieller Eingabe. Die Schieberichtung wird durch die Verbindung der einzelnen Stufen bestimmt. Wird der Ausgang Q_n einer Stufe mit dem Eingang D_{n+1} der Folgestufe verbunden, ergibt sich die Schieberichtung rechts. Die Verschaltung des Ausgangs Q_n mit dem Eingang D_{n-1} der vorhergehenden Stufe ermöglicht die Schieberichtung links.

Alle Flip-Flop-Stufen werden parallel durch das Taktsignal C gesteuert. Daher ändern sich alle Ausgänge gleichzeitig. Der Eingang des ersten D-Flip-Flops ist der serielle Eingang E des gesamten Schieberegisters und der Ausgang Q des letzten Flip-Flops entspricht dem seriellen Schieberegisterausgang A.

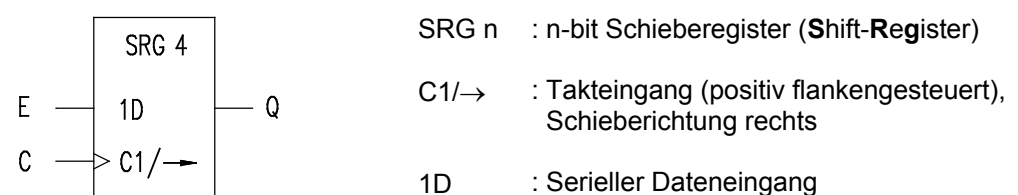


Abbildung 53 Symbol eines 4-bit Schieberegisters

Die Wirkungsweise eines Schieberegisters lässt sich gut an einem Impulsdiagramm erkennen.

Als Beispiel wird in ein zurückgesetztes Schieberegister das Bitmuster 1101 taktsynchron eingegeben. Erfolgt die Eingabe zuerst mit dem am weitesten rechts stehenden Bit, ergibt sich folgendes Impulsdiagramm:

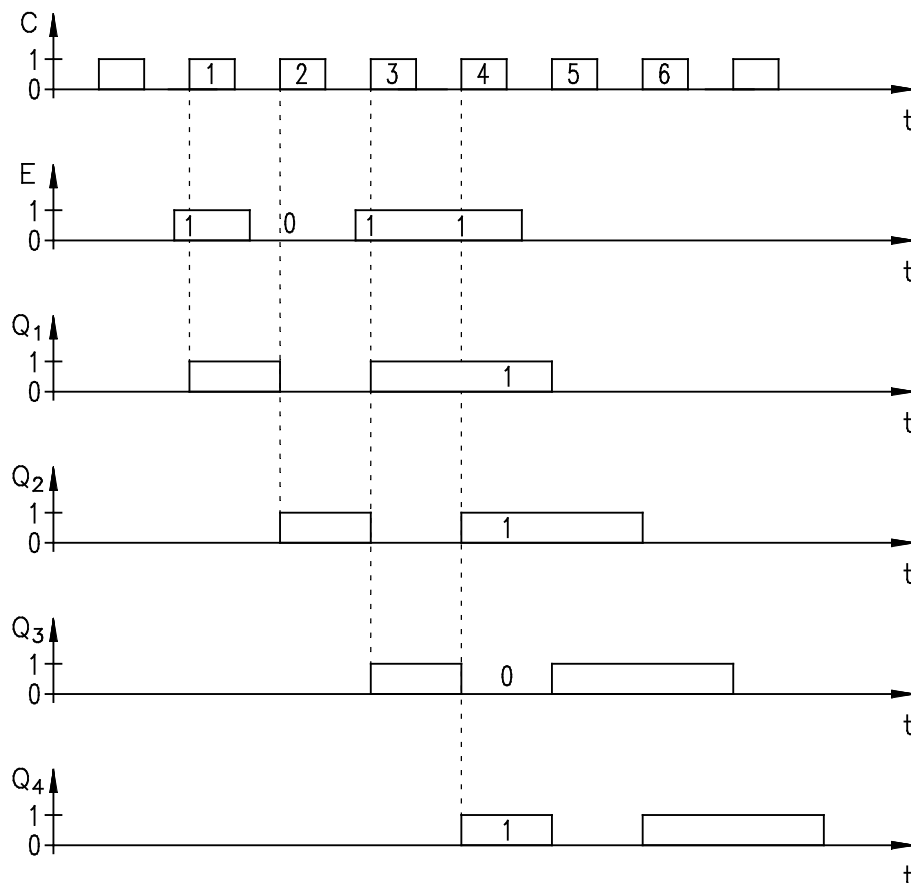


Abbildung 54 Impulsdiagramm eines 4-bit Schieberegisters

Erkennbar ist, dass die Information **seriell** von links nach rechts durch das Register wandert und nach der positiven Flanke des 4. Taktes **parallel** an den Ausgängen zur Verfügung steht. Es hat eine **Seriell-Parallel-Wandlung** stattgefunden. Erfolgen weitere Taktimpulse, wird das Bitmuster wieder **seriell** am Ausgang A ausgegeben.

Sofern die D-Flip-Flops vor dem Anlegen des Taktsignals mit einem Bitmuster parallel geladen werden können, ist daher auch eine **Parallel-Seriell-Wandlung** mit einem Schieberegister durchführbar.

Je nach zusätzlichem Logikaufwand, lassen sich verschiedene Schieberegistervarianten aus der Grundschialtung (Abbildung 52) entwickeln, z.B. mit:

- serieller Ein- und Ausgabe
- serieller Eingabe und paralleler Ausgabe
- paralleler Eingabe und serieller Ausgabe
- paralleler Ein- und Ausgabe
- Wechsel der Schieberichtung

Universell einsetzbare integrierte Schieberegister (z.B. 74HC194) besitzen eine Auswahllogik (Mode Control), die paralleles Laden und einen Wechsel der Schieberichtung ermöglicht.

Typische Anwendungsgebiete von Schieberegistern sind die:

- **Serielle Datenübertragung**

Hier übernehmen Schieberegister auf der Sendeseite die Parallel-Seriell-Wandlung und auf der Empfangsseite die Seriell-Parallel-Wandlung der zu übertragenden Informationen

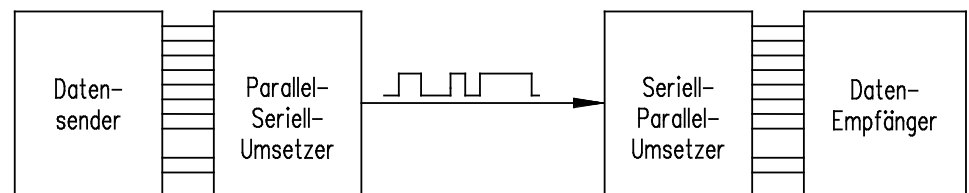


Abbildung 55 Serielle Datenübertragung

- **Multiplikation/Division von Dualzahlen**

In Rechenwerken erfolgt eine Multiplikation/Division durch Verschiebung von Dualzahlen. Ein Schiebescritt nach links bedeutet eine Multiplikation mit 2, ein Schiebescritt nach rechts eine Division durch 2. Die Dauer der Rechenoperation (Mult/Div) ist dabei unabhängig von der Bit-Breite der Dualzahl.

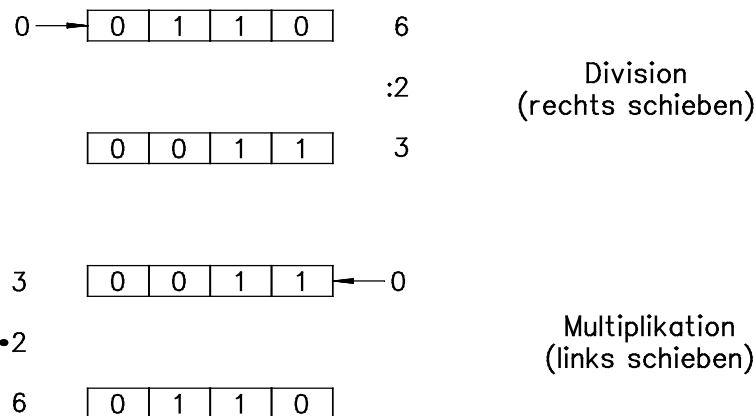


Abbildung 56 Multiplikation und Division mit Faktor 2

- **Signalverzögerung**

Aus dem Impulsdiagramm in Abbildung 54 ist erkennbar, dass das Ausgangssignal eines n-bit Schieberegisters dem um n Takte verzögerten Eingangssignal entspricht. Damit lassen sich Schieberegister zur Signalverzögerung, z.B. bei der Audiotechnik in digitalen Filtern, einsetzen.

• Umlaufregister

Dabei handelt es sich um rückgekoppelte Schieberegister, wie sie die beiden folgenden Abbildungen zeigen.

Bei der oberen Abbildung handelt es sich um einen sog. **Ringzähler**. Hier wird ein Bitmuster (z.B. 1000) im Kreis verschoben. Diese Anordnung kommt z.B. in Schrittschaltwerken zur Steuerung einfacher Maschinen vor. Weitere Anwendungen sind: Lauflichter und Schrittmotorsteuerungen.

Die untere Abbildung zeigt einen **Pseudozufallsgenerator**.

Am Ausgang A des Generators entsteht ein Signal, das scheinbar willkürlich von 0 auf 1 wechselt. Dazu werden bestimmte Ausgänge (siehe Tabelle 18) des Schieberegisters EXOR verknüpft und auf den Eingang zurückgekoppelt.

Damit der Pseudozufallsgenerator sicher startet, dürfen zu Beginn nicht alle Flip-Flops zurückgesetzt sein. Die Zykluslänge dieser Generatoren beträgt bei n Flip-Flops maximal $2^n - 1$ Taktimpulse oder $2^n - 1$ „Zufallszahlen“.

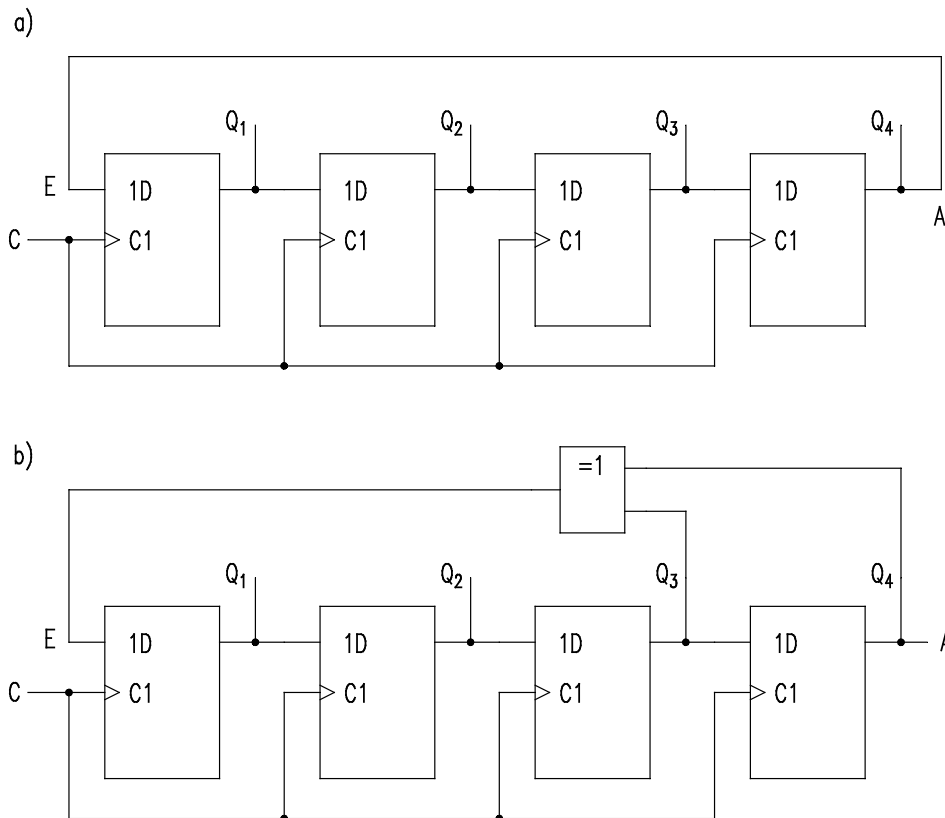


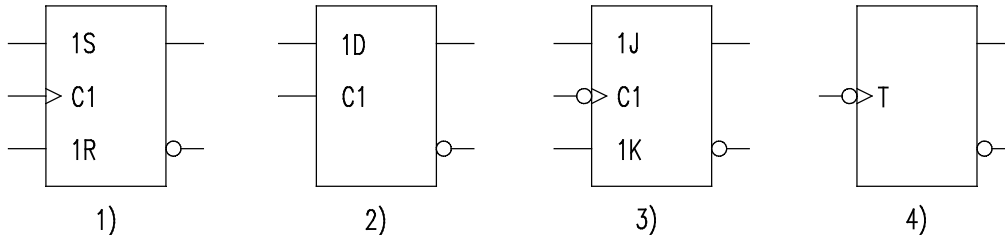
Abbildung 57 Umlaufregister: Ringzähler (a) und Pseudozufallsgenerator (b)

Registerlänge	Zurückgekoppelte Ausgänge	Zykluslänge
2	1, 2	3
3	2, 3	7
4	3, 4	15
5	3, 5	31
6	5, 6	63
7	6, 7	127
8	2, 3, 4, 8	255
9	5, 9	511
10	7, 10	1.023
11	9, 11	2.047
12	2, 10, 11, 12	4.095
13	1, 11, 12, 13	8.191
14	2, 12, 13, 14	16.383
15	14, 15	32.767
16	11, 13, 14, 16	65.535
17	14, 17	131.071
18	11, 18	262.143
19	14, 17, 18, 19	524.287
20	17, 20	1.048.575

Tabelle 18 Rückkopplungsbedingungen für Pseudozufallsgeneratoren

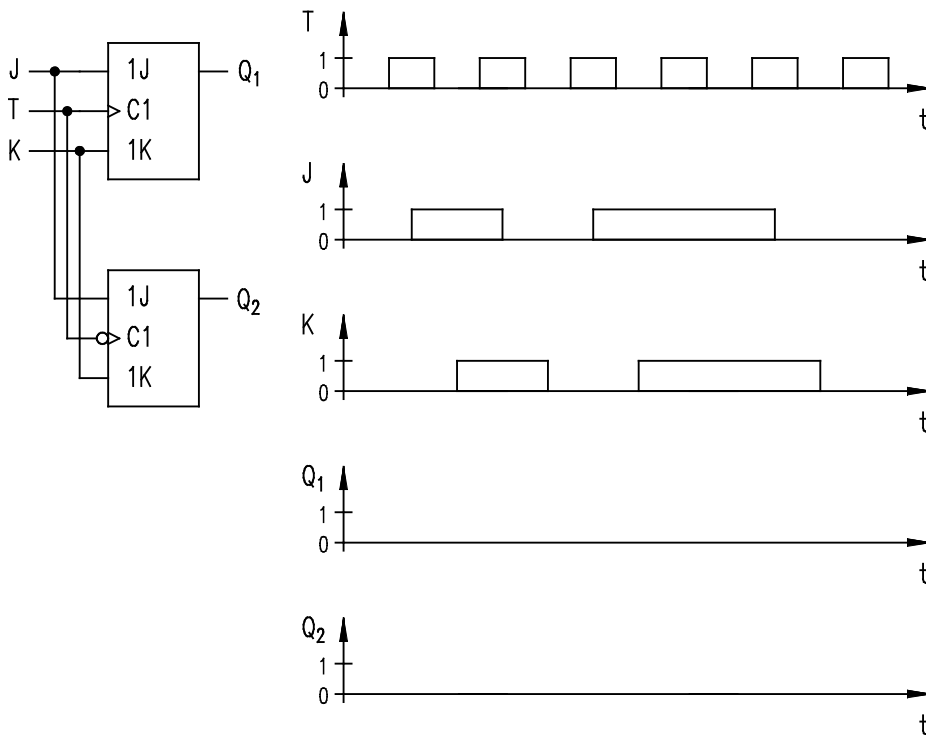
Aufgabe 1

Welche Flip-Flop Typen sind in der Abbildung dargestellt?



Aufgabe 2

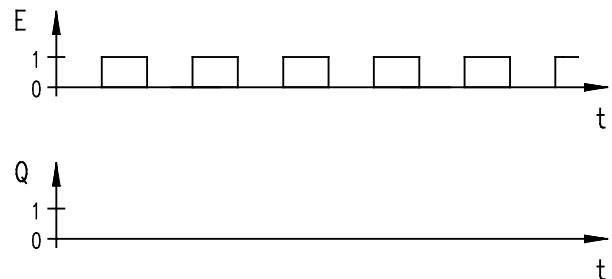
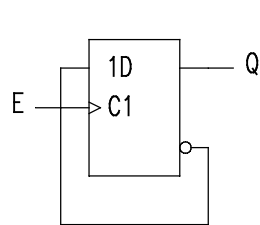
Ergänzen Sie für die abgebildeten Flip-Flops das Impulsdiagramm!



Aufgaben

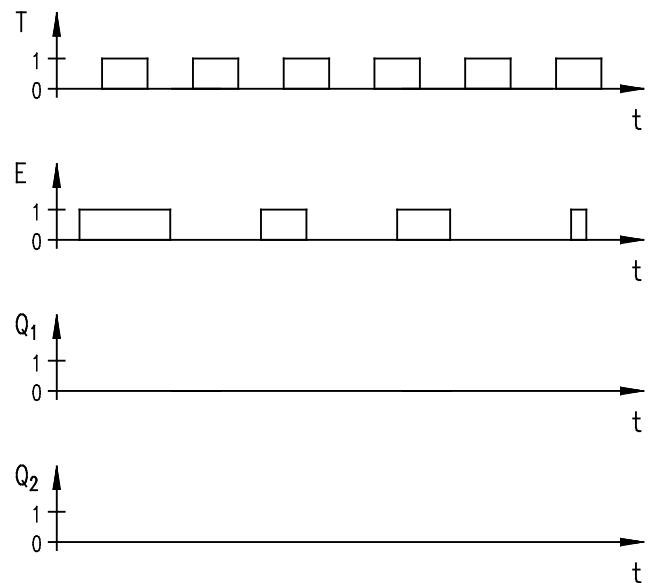
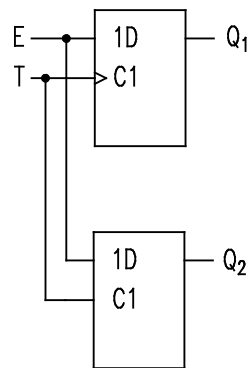
Aufgabe 3

Ergänzen Sie für das abgebildete Flip-Flop das Impulsdiagramm und nennen Sie auf Grund des Ausgangssignals die Schaltungsfunktion!



Aufgabe 4

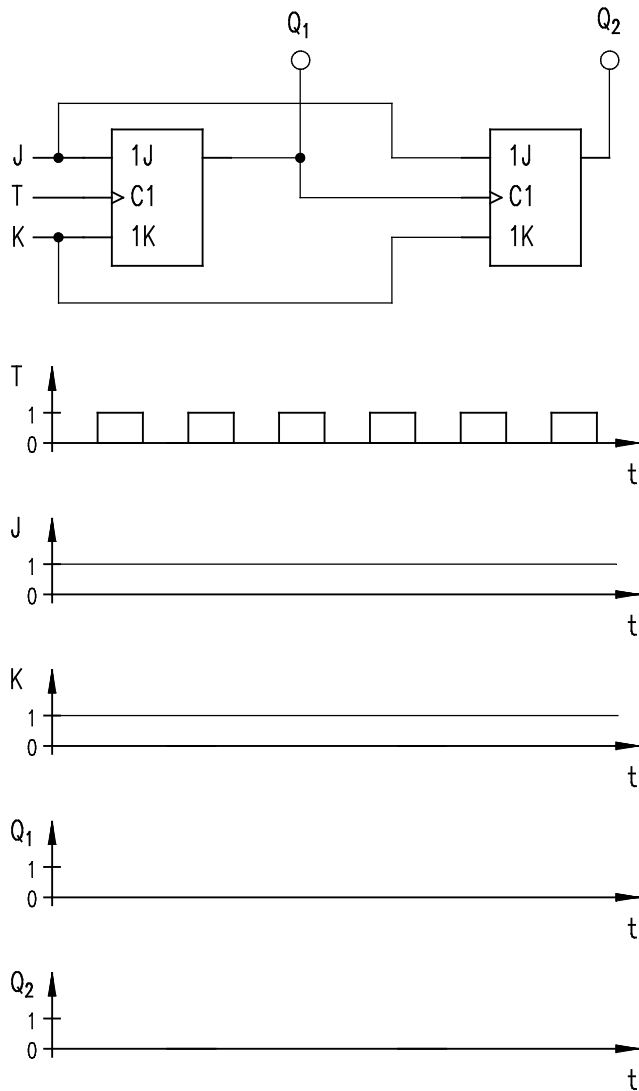
Die abgebildeten Flip-Flops werden mit dem gleichen Eingangssignal E und Taktsignal T angesteuert.



Ergänzen Sie das Impulsdiagramm!

Aufgabe 5

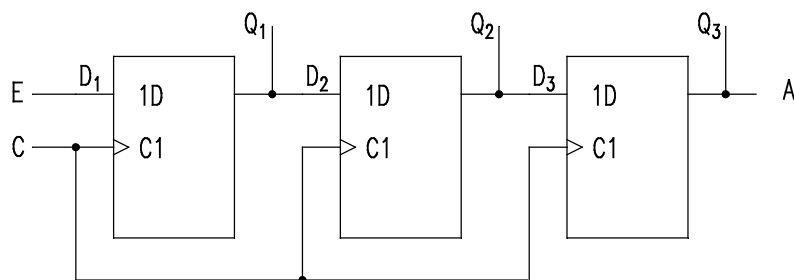
5.1 Ergänzen Sie für die abgebildeten Flip-Flops das Impulsdiagramm!



5.2 Welche Schaltung ist mit den beiden Flip-Flops realisiert worden?

Aufgabe 6

Welche Schaltung ist mit den Flip-Flops realisiert?



Lernbereich

3 Codier- und Auswahlschaltungen

In der digitalen Informationsverarbeitung werden in vielen Bereichen Codier- und Auswahlschaltungen eingesetzt. Typisch ist die Informationszuführung für verschiedene Geräte oder Funktionsgruppen über gemeinsame Datenleitungen, sog. Bussysteme, um den Verdrahtungsaufwand zu reduzieren. Zur Vermeidung von Datenkonflikten auf den Busleitungen darf beim Datenaustausch jeweils nur ein Gerät bzw. eine Funktionsgruppe aktiviert sein. Dies wird durch den Einsatz von speziellen Auswahlschaltungen erreicht.

In der Mikroprozessortechnik müssen z.B. auf Grund der aktuellen Bitkombination auf den Adressleitungen Speicher-IC's angesprochen werden. Zur Aktivierung eines Speichers ist eine **Auswahl-** oder **Decodierschaltung** notwendig (s. folgende Abbildung), da die Datenausgänge aller Speicher parallel auf ein Bussystem geschaltet sind.

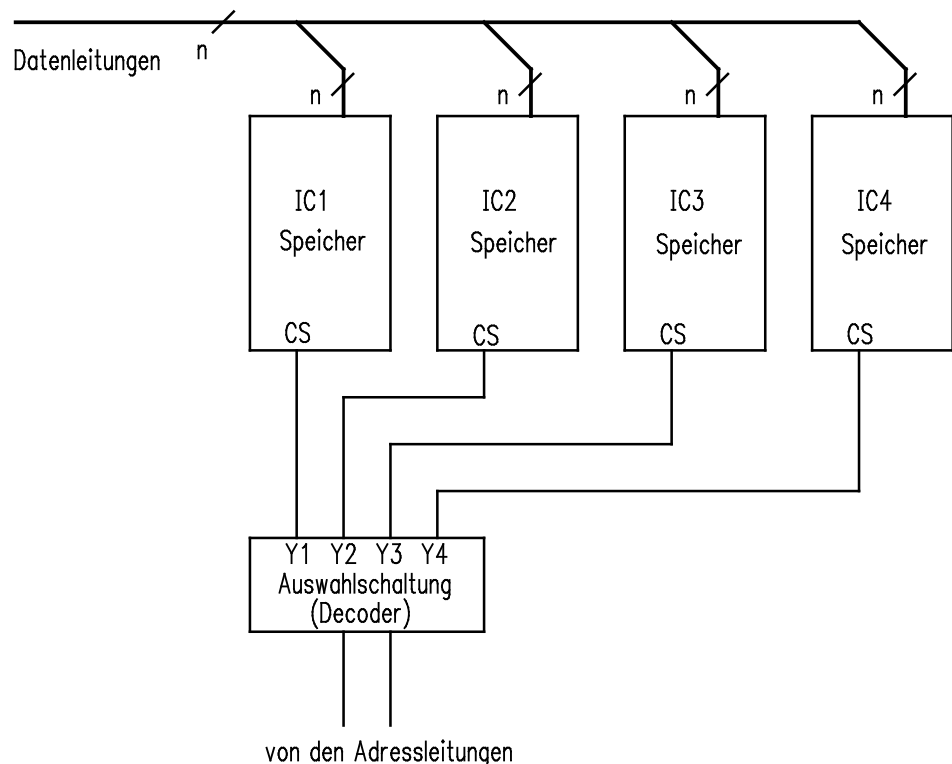


Abbildung 58 Auswahlschaltung zur Ansteuerung von Speicher-IC's

Ein anderes Beispiel für den Einsatz von Auswahlschaltungen ist die Übertragung von Informationen mehrerer Datenquellen über **einen** Übertragungskanal auf unterschiedliche Datenempfänger (s. Abbildung 59). Hier wird auf der Eingangsseite des Übertragungskanals ein **Multiplexer** und auf der Ausgangsseite ein **Demultiplexer** eingesetzt.

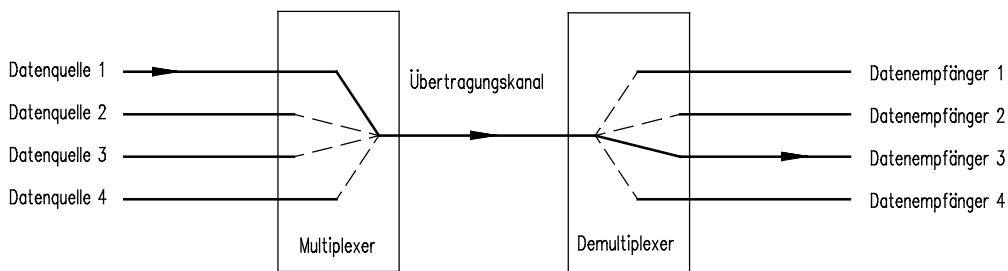


Abbildung 59 Prinzip der Auswahl eines Signalwegs

Ein Multiplexer ist eine Auswahl-schaltung, die durch entsprechende Steuersignale zeitlich nacheinander bestimmte Eingangssignale an ihren Ausgang weiterleitet.

Ein Demultiplexer übernimmt die umgekehrte Aufgabenstellung: das Weiterleiten eines Eingangssignals auf einen bestimmten Ausgang, der je nach Steuerbefehl aus mehreren möglichen Ausgängen gewählt wurde.

Die Steuerung der Auswahl-schaltungen erfolgt mittels codierter Signale. Die Codierung im Dualcode minimiert dabei die Anzahl der notwendigen Steuerleitungen.

Der Dualcode ist die wichtigste Zahlendarstellung in der Informationstechnik. Für bestimmte Anwendungsfälle sind aber andere Zahlendarstellungen günstiger bzw. leichter zu verarbeiten. **Codewandler** übernehmen dabei die Aufgabe, Informationen von einer Zahlendarstellung in eine andere zu decodieren.

Typisch ist die Decodierung binärcodierter in dezimalcodierte Signale oder die Umsetzung BCD-codierter Informationen in Signale zur Ansteuerung einer 7-Segment-Anzeige.

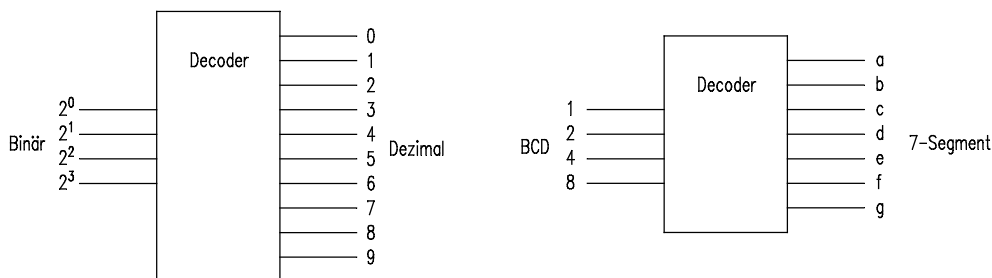


Abbildung 60 Decoder Binär/Dezimal und Decoder BCD zu 7-Segment

3.1 Decoder

Eine für die Datentechnik typische Situation ist die gezielte Ansteuerung von Speicherbausteinen mittels minimaler Anzahl von Steuerleitungen (s. Abbildung 61).

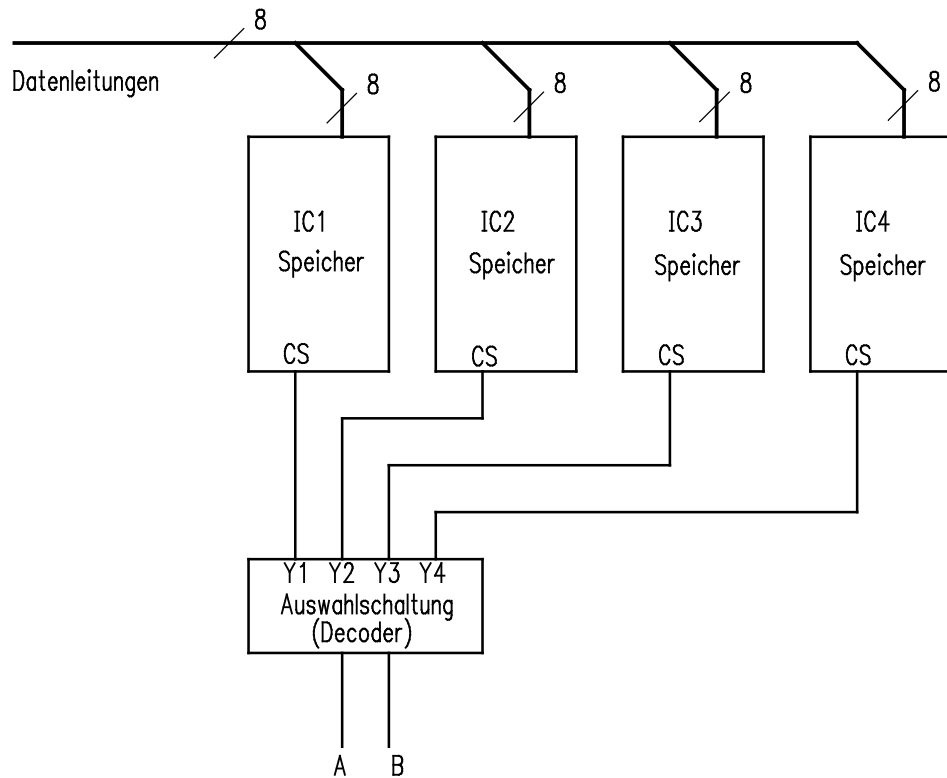


Abbildung 61 Speicherauswahl mit Decoder

Die Logikzustände der Datenleitungen sollen in einem der vier Speicher-IC's gespeichert werden. Die Auswahl, welches IC zur Speicherung der Daten verwendet wird, erfolgt mithilfe der codierten Signale auf den Leitungen A und B (Tabelle 19).

B	A	Anmerkung
0	0	Speichern in IC 1
0	1	Speichern in IC 2
1	0	Speichern in IC 3
1	1	Speichern in IC 4

Tabelle 19 Signalcodierung

Die IC's können die auf den Datenleitungen liegenden binären Informationen speichern, wenn am CS-Eingang (Chip-Select) ein logischer 1-Pegel anliegt. Da die Information, welches IC die Daten speichern soll, in codierter Form vorliegt, wird das Ansteuersignal für die Speicher-IC's mit einer Auswahlschaltung - Decoder genannt - gewonnen. In dem oben beschriebenen Fall wird eine Auswahlschaltung mit zwei Eingängen, A und B, und vier Ausgängen, Y1 bis Y4, verwendet. Diese Auswahlschaltung wird deshalb auch 2-zu-4-Decoder genannt. Da immer nur einer der vier Ausgänge den logischen Pegel 1 führt, spricht man auch von einem 1-aus-4-Decoder. **Ein** Ausgang von **vier** möglichen wird selektiert.

Dieser 1-aus-4-Decoder hat die unten abgebildete Funktionstabelle.

Eingänge		Ausgänge				Anmerkung
B	A	Y1	Y2	Y3	Y4	
0	0	1	0	0	0	Speichern in IC 1
0	1	0	1	0	0	Speichern in IC 2
1	0	0	0	1	0	Speichern in IC 3
1	1	0	0	0	1	Speichern in IC 4

Tabelle 20 Funktionstabelle eines 1-aus-4-Decoders

Die Funktionstabelle kann mit der in Abbildung 62 wiedergegebenen logischen Schaltung realisiert werden.

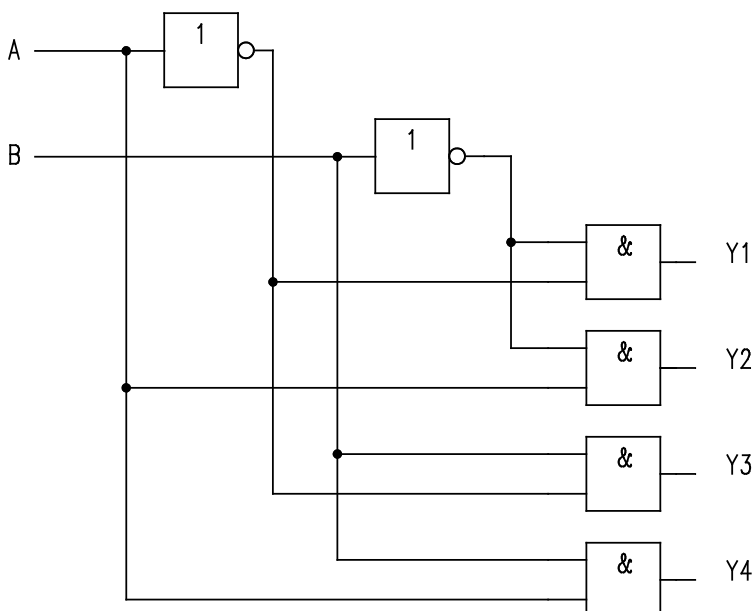


Abbildung 62 Logische Schaltung des 1-aus-4-Decoders

Liegt z.B. am Eingang A eine log. „1“ und an B eine log. „0“, wird der Ausgang Y2 log. „1“ und IC2 wird selektiert.

Binär-zu-Oktal-Decoder

Am Beispiel des IC's 74LS138 wird im Folgenden eine Adressdecodierung mit einem Binär-zu-Oktal-Decoder beschrieben. Dieser Decoder ist ein 1-aus-8-Decoder und arbeitet nach dem selben Funktionsprinzip wie der 1-aus-4-Decoder. Gegenüber dem 1-aus-4-Decoder ist die Anzahl der Ausgänge auf acht und die zur Auswahl notwendigen Eingänge auf drei erweitert ($8 = 2^3$) worden.

In Abbildung 63 ist das Logiksymbol und die Bedeutung der Anschlüsse des IC's wiedergegeben.



Abbildung 63 Logiksymbol 74LS138

An den Dateneingänge A, B und C liegt die Dateninformation binärcodiert vor. Diese Information wird decodiert und einer der acht (oktal) Ausgänge wird aktiv. Bei diesem IC-Typ handelt es sich um einen Baustein mit LOW-aktiven Ausgängen. Neben den drei Dateneingängen hat dieses IC noch drei Freigabe-Eingänge. Nur wenn diese drei Eingänge die entsprechenden logischen Pegel führen (siehe Logiksymbol), kann einer der Ausgänge auf LOW geschaltet werden. Damit ist die Speicherauswahl in einem Mikroprozessorsystem entsprechend der folgenden Abbildung realisierbar.

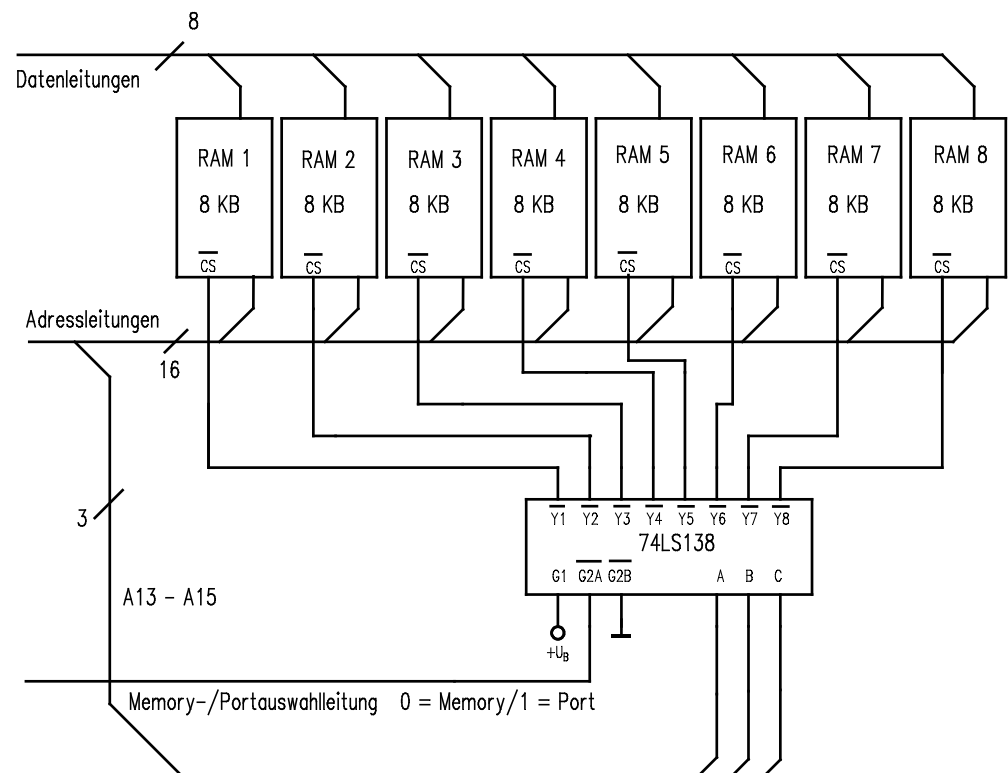


Abbildung 64 Ansteuerung von RAM-Speicherbausteinen mit dem 74LS138 als Adressdecoder

Die acht Speicherbausteine können beschrieben werden, wenn am \overline{CS} -Eingang der acht RAM's ein LOW-Pegel anliegt. Dieses Signal wird mit dem 74LS138 erzeugt. An die Eingänge A, B und C des 74LS138 sind die drei Adressleitungen A13, A14 und A15 angeschlossen.

A15 angeschlossen. Da die Speicher und nicht die Ausgabebausteine (Ports) beschrieben werden sollen, wird zusätzlich mit dem Freigabeeingang $\overline{G2A}$ das LOW-aktive Signal der Memory-/Portauswahlleitung ausgewertet. Ein RAM-Baustein wird daher nur dann zum Schreiben freigegeben, wenn das Memory-/Port-Signal 0-Pegel ausweist und die entsprechende Adresse am IC 74LS138 anliegt.

A15	A14	A13	Adressbereich	IC
0	0	0	0000H - 1FFFH	RAM 1
0	0	1	2000H - 3FFFH	RAM 2
0	1	0	4000H - 5FFFH	RAM 3
0	1	1	6000H - 7FFFH	RAM 4
1	0	0	8000H - 9FFFH	RAM 5
1	0	1	A000H - BFFFH	RAM 6
1	1	0	C000H - DFFFH	RAM 7
1	1	1	E000H - FFFFH	RAM 8

Tabelle 21 Zuordnung der Speicherbereiche

An den Adressleitungen (A15 ... A0) liegt z.B. die Bit-Kombination „1000 0000 0000 0001“ an. Die drei höchstwertigsten Bits, „100“, werden vom IC 74LS138 ausgewertet. Liegt an der Memory-/Portauswahlleitung eine „0“ an, so wird RAM 5 freigegeben, Adressbereich 8000H - 9FFFH. Die übrigen Bits, A12-A0, werden im RAM decodiert. Es wird die Speicherstelle 8001H beschrieben.

BCD-zu-7-Segment-Wandler

Das TTL IC 74LS247 ist ein BCD-zu-7-Segment-Wandler. Es decodiert die in den Datenbits D0 bis D3 enthaltene Information und steuert die Anzeige so an, dass der entsprechende dezimale Zahlenwert erscheint.

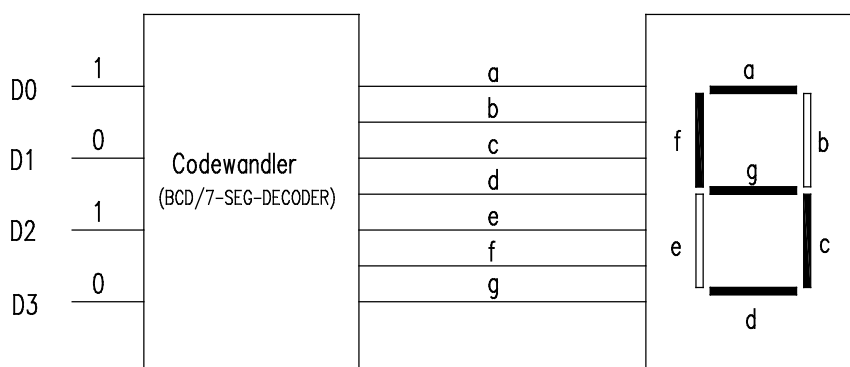


Abbildung 65 Ansteuerung einer 7-Segment-Anzeige durch Codewandler

D3	D2	D1	D0	Ziffer	Segment						
					a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	1	0	1	1	0	0	0	0
0	0	1	0	2	1	1	0	1	1	0	1
0	0	1	1	3	1	1	1	1	0	0	1
0	1	0	0	4	0	1	1	0	0	1	1
0	1	0	1	5	1	0	1	1	0	1	1
0	1	1	0	6	1	0	1	1	1	1	1
0	1	1	1	7	1	1	1	0	0	0	0
1	0	0	0	8	1	1	1	1	1	1	1
1	0	0	1	9	1	1	1	1	0	1	1

Tabelle 22 Zuordnungstabelle BCD-zu-7-Segment-Wandler

Nachfolgend ist das logische Symbol des IC's 74LS247 wiedergegeben.

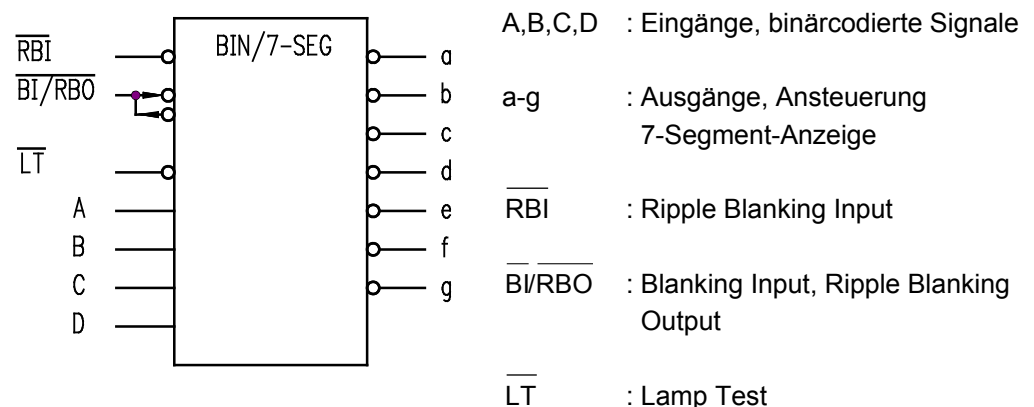


Abbildung 66 Logiksymbol 74LS247

Dieses IC hat OPEN-Kollektor-Ausgänge (a-g). Leuchtdioden können daher mit Strombegrenzungswiderständen direkt an die Ausgänge angeschlossen werden. Neben den Dateneingängen A bis D hat dieses IC noch Steuereingänge: $\overline{\text{LT}}$, $\overline{\text{RBI}}$, $\overline{\text{BI}}$ und $\overline{\text{RBO}}$. Legt man an den Steuereingang $\overline{\text{LT}}$ ein LOW-Signal, so leuchten alle Leuchtdioden der 7-Segment-Anzeige. Mit dem LOW-aktiven $\overline{\text{BI}}$ -Eingang werden alle LED's dunkel gesteuert.

Eine automatische Vornullunterdrückung bei mehrstelligen Anzeigen kann mit dem Eingang $\overline{\text{RBI}}$ und dem Ausgang $\overline{\text{RBO}}$ der Decoder-IC's durchgeführt werden (siehe Abbildung 67).

Wird an den Eingang $\overline{\text{RBI}}$ eine log. „0“ angelegt und sind die Eingänge A, B, C und D ebenfalls log. „0“, dann werden alle LED's dunkel geschaltet. In der Abbildung ist die Beschaltung für eine vierstellige Anzeige wiedergegeben. Die Anschlüsse $\overline{\text{RBI}}$ und $\overline{\text{RBO}}$ sind so verschaltet, dass bei den **ersten drei** Stellen, 1000er, 100er und 10er-Stelle, die Vornullunterdrückung möglich ist.

Solange der Schalter S geöffnet ist, ist die Vornullenunterdrückung ausgeschaltet, da am $\overline{\text{RBI}}$ -Eingang der 1000er-Stelle ein log. „1“-Pegel anliegt. Der Ausgang $\overline{\text{RBO}}$ liefert einen log. „1“-Pegel zum Ausschalten der Vornullenunterdrückung der 100er und 10er-Stelle.

Wird Schalter S geschlossen, liegt am Eingang $\overline{\text{RBI}}$ eine log. „0“ an. Die Vornullenunterdrückung ist eingeschaltet. Da an den Dateneingängen der 1000er-Stelle die Bit-Kombination „0000“ anliegt, werden die LED's dunkel geschaltet und der Ausgang $\overline{\text{RBO}}$ gibt die Freigabe der Vornullenunterdrückung an das IC der 100er-Stelle weiter. Da auch dort die Dateneingänge auf „0“ liegen, sind hier ebenfalls die LED's dunkel geschaltet. Der Ausgang $\overline{\text{RBO}}$ schaltet mit einem log. „0“ die Vornullenunterdrückung der 1er-Stelle ein. Die hier aber die Bit-Kombination der Datenleitungen „0101“ ist, erscheint auf der Anzeige der dezimale Zahlenwert 5. Der Eingang $\overline{\text{RBI}}$ der 1er-Stelle ist fest auf „1“ gelegt. Die Unterdrückung ist nicht aktiviert. Die Anzeige gibt immer den entsprechenden dezimalen Zahlenwert aus, hier ebenfalls eine 5.

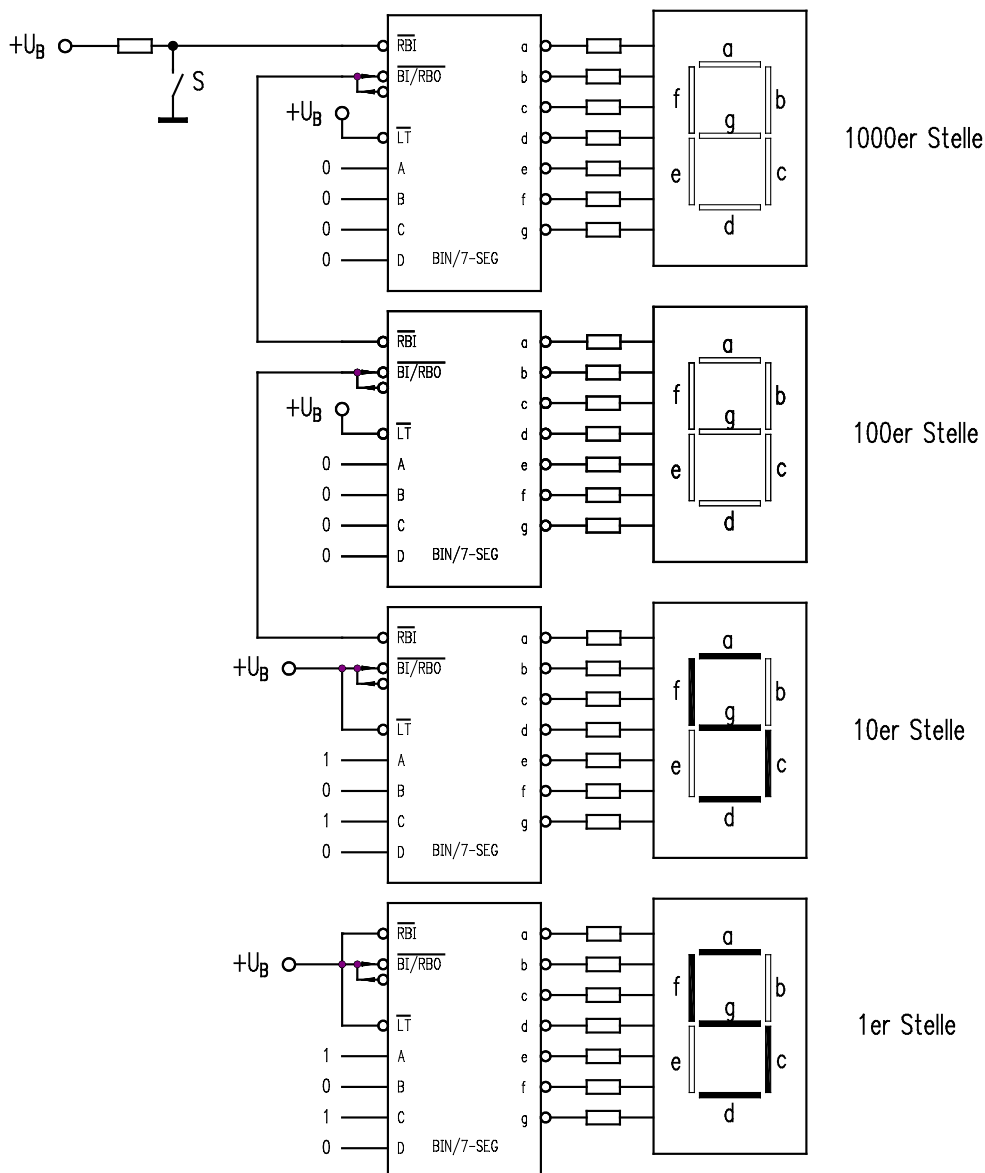


Abbildung 67 Prinzip einer Vornullenunterdrückung mit IC 74LS247

3.2 Multiplexer

In der Informationsverarbeitung ist es oft erforderlich, eine Datenquelle (Leitungen) von mehreren auf einen Übertragungskanal zu schalten. (s. Abbildung 68).

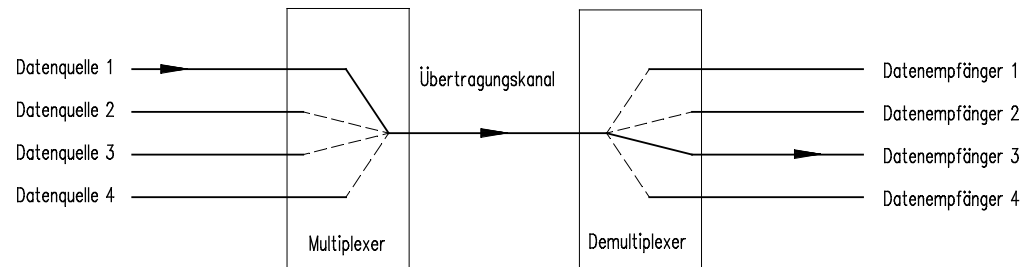


Abbildung 68 Prinzip eines Datenselektors

Eingangsseitig wird dazu ein Multiplexer, ausgangsseitig ein Demultiplexer eingesetzt. Wenn die Umschaltung zwischen den Leitungen sehr schnell erfolgen muss, werden elektronische Schalter verwendet.

Im oben abgebildeten Fall des Multiplexers wird eine Leitung aus vier Leitungen ausgewählt. Der Multiplexer enthält daher einen 1-aus-4-Decoder mit 2 Steuerleitungen (Abbildung 69, Tabelle 23).

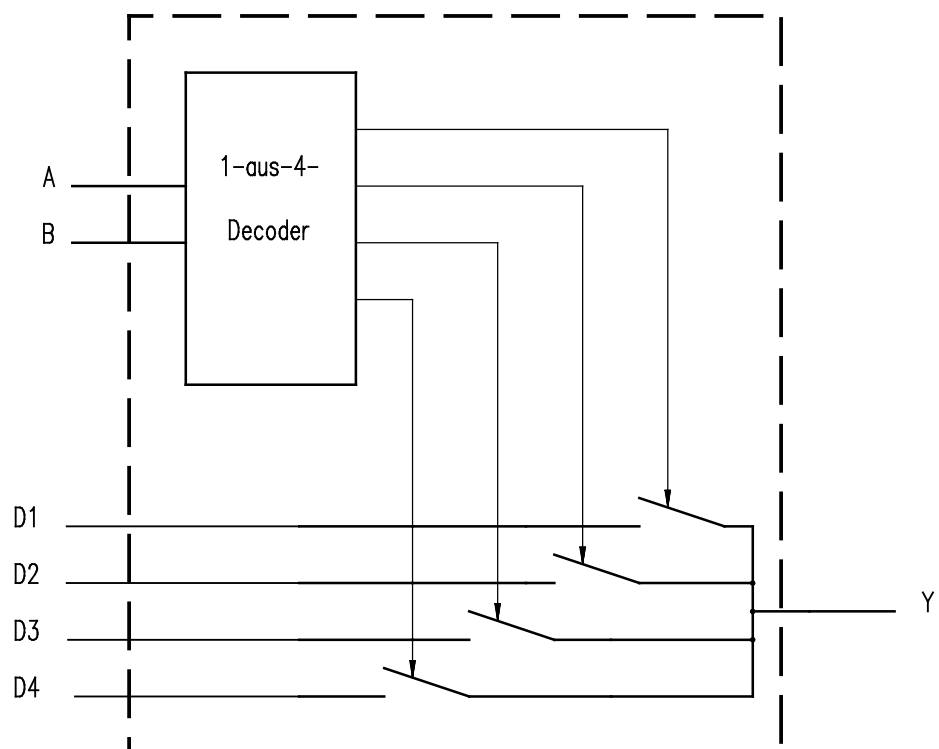


Abbildung 69 Prinzipieller Aufbau eines Multiplexers

B	A	Anmerkung
0	0	D1 auf Y schalten
0	1	D2 auf Y schalten
1	0	D3 auf Y schalten
1	1	D4 auf Y schalten

Tabelle 23 Funktionstabelle Multiplexer

Diese Funktionstabelle kann mit folgender logischer Schaltung realisiert werden:

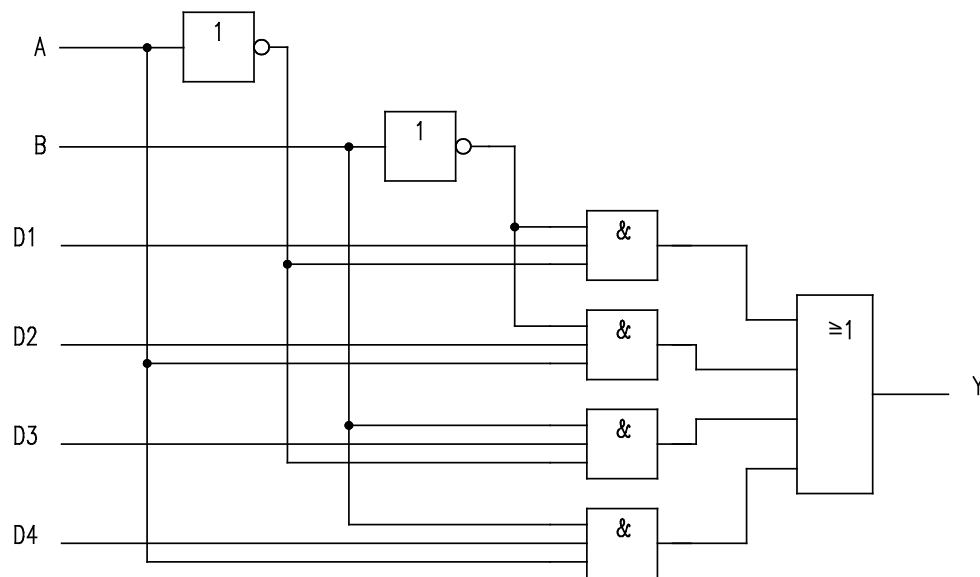


Abbildung 70 Logikplan 4-zu-1-Multiplexer

Soll z.B. die Datenleitung D4 auf den Ausgang Y geschaltet werden, müssen die Steuereingänge A und B entsprechend beschaltet werden. Der Funktionstabelle kann man entnehmen, dass für diesen Fall an die Steuereingänge A und B der log.-Pegel „1“ angelegt werden muss.

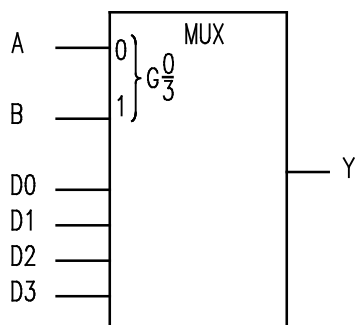


Abbildung 71 Schaltsymbol eines 4-zu-1-Multiplexers

- MUX : Multiplexer
- A,B : Steuereingänge zur Auswahl eines Ausgangs (0...3)
A, Eingang mit der niedrigsten Wertigkeit (2^0)
B, Eingang mit der höchsten Wertigkeit (2^1)
- Y : Ausgang
- D0 ... D3 : Datenleitungen

Demultiplexer

Mit einem Demultiplexer wird das Signal des Übertragungskanal auf eine der vier Leitungen verteilt (vgl. Abbildung 68). Der Demultiplexer enthält einen 1-aus-4-Decoder mit 2 Steuerleitungen (Abbildung 72, Tabelle 24).

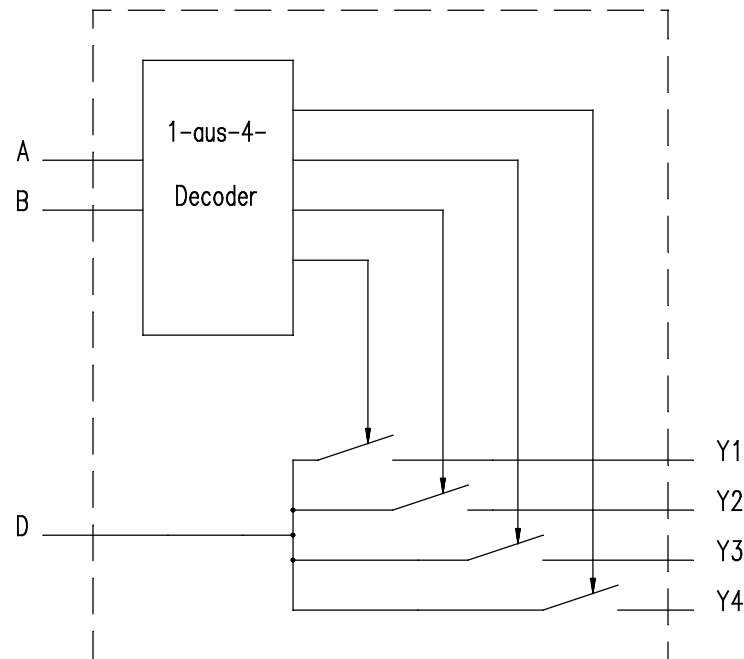


Abbildung 72 prinzipielle Funktionsweise eines Demultiplexers

B	A	Anmerkung
0	0	D nach Y1 durchschalten
0	1	D nach Y2 durchschalten
1	0	D nach Y3 durchschalten
1	1	D nach Y4 durchschalten

Tabelle 24 Funktionstabelle Demultiplexer

Soll z.B. der Datenkanal D auf den Ausgang Y2 geschaltet werden, muss an den Steuereingang A eine log. „1“ und an den Steuereingang B eine log. „0“ angelegt werden.

Demultiplexer lassen sich mit Decodern realisieren. Verschaltet man das IC 74LS138 entsprechend der Abbildung 73, so arbeitet es als 8 Kanal-Demultiplexer mit **invertierenden** Ausgängen. Die Eingänge $\overline{G2A}$ und $\overline{G2B}$ sind auf LOW gelegt, sodass der Eingang G1 den logischen Pegel am Ausgang der UND-Verknüpfung und somit des selektierten Ausgangs bestimmt. G1 ist also der Übertragungskanal bzw. die Datenleitung (vgl. Abbildung 68).

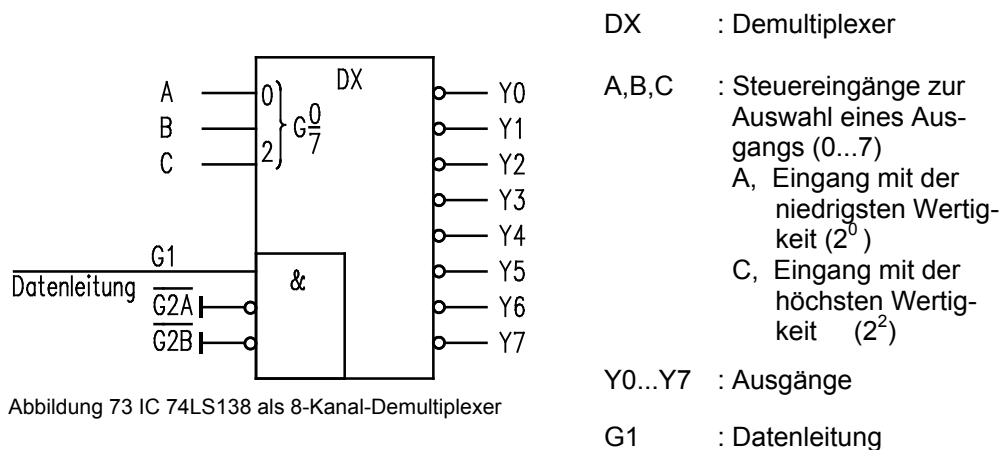


Abbildung 73 IC 74LS138 als 8-Kanal-Demultiplexer

Häufig sind in der Informationstechnik analoge Signale über einen Übertragungskanal zu senden. Die digitalen Multiplexer/Demultiplexer sind in diesem Fall nicht geeignet. In der CMOS-Schaltkreisfamilie gibt es IC's, die auch Analogsignale verarbeiten können. Diese IC's arbeiten auf Grund der verwendeten Transmission-Gates bidirektional, d.h. sie können als Multiplexer oder Demultiplexer eingesetzt werden. Das verwendete Transmission-Gate verhält sich dabei wie ein gesteuerter Schalter (Abbildung 74). In Abbildung 75 ist der innere Aufbau des Gates dargestellt.

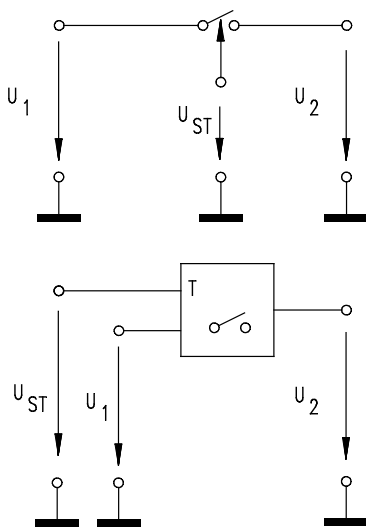


Abbildung 74 Funktionsweise des Transmission-Gates

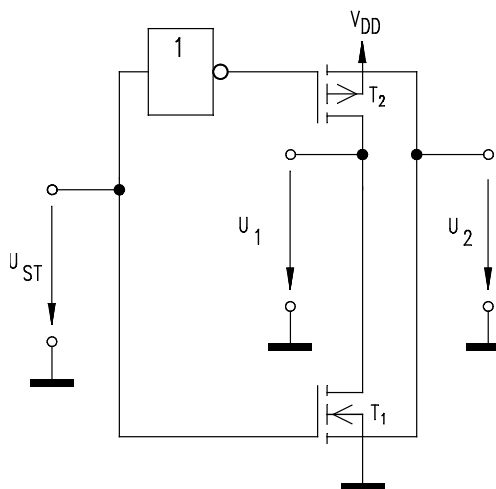


Abbildung 75 Innerer Aufbau des Transmission-Gates

Die folgende Tabelle enthält einige der gebräuchlichsten Multiplexer/Demultiplexer.

IC-Typ	TTL-Technik	CMOS
vier 2-zu-1-Multiplexer	74LS157	4066 analog
zwei 4-zu-1-Multiplexer	74LS153	4052 analog
einen 16-zu-1-Multiplexer	74LS150	4067 analog
drei 1-aus-2-Demultiplexer		4053 analog
zwei 1-aus-4-Demultiplexer	74LS239	4052 analog
einen 1-aus-8-Demultiplexer	74LS138	4051 analog
einen 1-aus-16-Demultiplexer	74LS154	4067 analog

Tabelle 25 Multiplexer/Demultiplexer-Typen

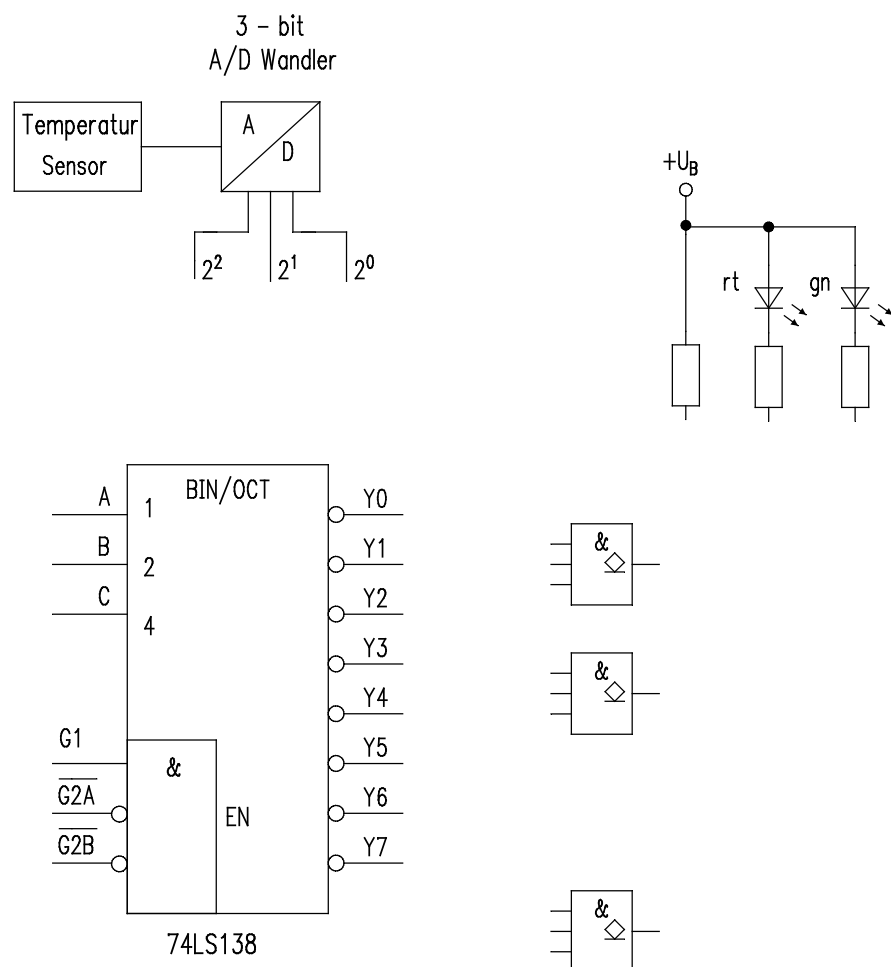
Aufgaben

Aufgabe 1

In einer Fabrik werden Kunststoffteile zum Aushärten in einen Heizofen gegeben. Zur Überwachung der Ofentemperatur ist ein Temperatursensor mit einem A/D-Wandler an einen Decoder gekoppelt. Der A/D-Wandler ist so dimensioniert, dass den 8 Stufen (0 ... 7) folgende Temperaturbereiche zugeordnet sind:

	$v < 10\text{ °C}$	Stufe 0
10 °C	$\leq v < 20\text{ °C}$	Stufe 1
20 °C	$\leq v < 30\text{ °C}$	Stufe 2
.	.	.
60 °C	$\leq v < 70\text{ °C}$	Stufe 6
70 °C	$\leq v$	Stufe 7

Der Decoder soll eine rote Kontrollleuchte (LED) ansteuern, wenn die Ofentemperatur unter 30 °C oder 60 °C und mehr beträgt. Eine grüne Kontrollleuchte (LED) soll aufleuchten, wenn die Heiztemperatur in Ordnung ist, d.h. über 30 °C und unter 60 °C liegt!



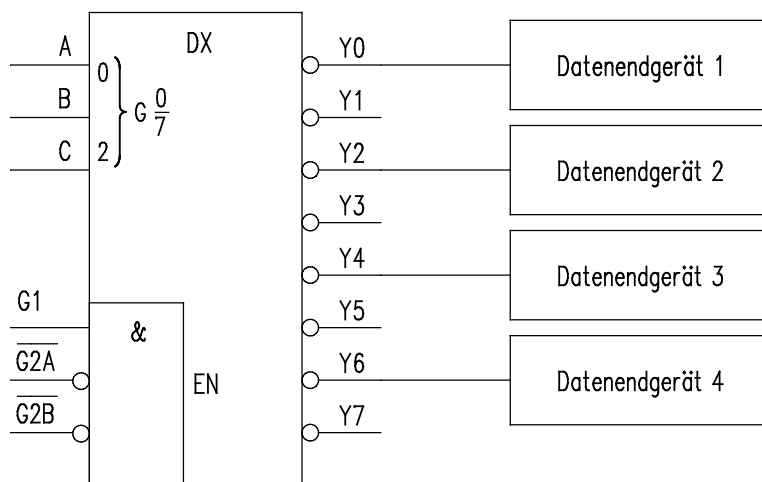
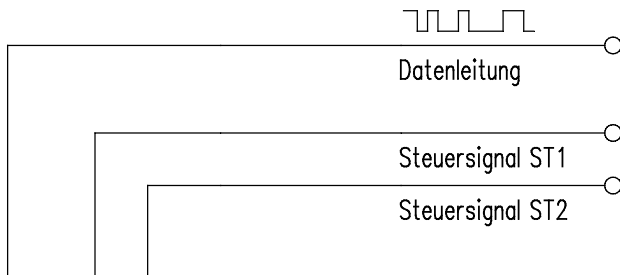
Beschalten Sie den Decoder und vervollständigen Sie den Schaltplan so, dass die oben beschriebene Funktion erreicht wird!

Hinweis: Nutzen Sie das Datenblatt aus dem Modul Formeln und Datenblätter.

Aufgabe 2

Ein serielles Datensignal soll durch einen Demultiplexer auf vier Datenendgeräte verteilt werden.

Die Signale ST1 (Wertigkeit 2^0) und ST2 (Wertigkeit 2^1) steuern die Verteilung des Datensignals auf die Datenendgeräte. Als Demultiplexer steht das TTL-IC 74LS138 zur Verfügung.



Beschalten Sie den Demultiplexer und vervollständigen Sie den Schaltplan so, dass eine **nichtinvertierende** Datenübertragung realisiert wird!

Hinweis: Nutzen Sie das Datenblatt aus dem Modul Formeln und Datenblätter.

Lernbereich

4 D/A- und A/D-Wandler

4.1 D/A-Wandler

Um aus einem digitalen Signal ein analoges Signal zu erzeugen, werden Digital-Analog-Wandler, kurz: D/A-Wandler, eingesetzt. Ein typisches Beispiel ist in der folgenden Abbildung wiedergegeben:

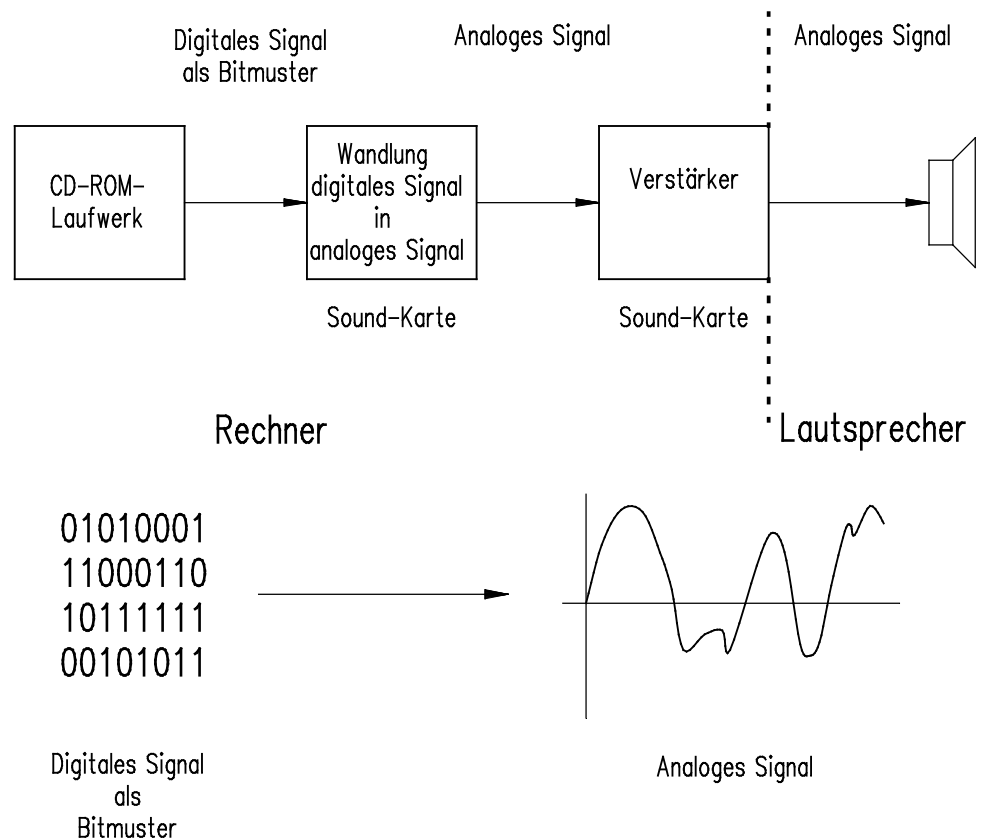


Abbildung 76 Beispiel eines D/A-Wandlers

Musik wird fast nur noch in digitaler Form gespeichert, z.B. auf einer Musik-CD oder einer Spiele-CD (WAV-Dateien). Diese Form weist einige Vorteile gegenüber der analogen auf: zum einen ist sie störunanfällig (kein Knistern, keine Aussetzer), zum anderen sind dadurch erst Spiele in der heute üblichen Form möglich (bewegte Bilder mit Musik).

Auf der CD ist die Musik in digitaler Form als Bit-Muster gespeichert (Abbildung 77a). Das CD-Rom-Laufwerk liest ca. alle 22,6 μ s ein neues Bitmuster von einer Audio-CD.

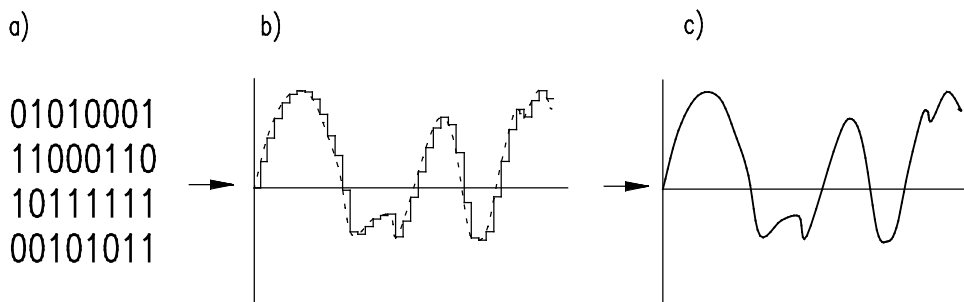


Abbildung 77 Signalformen

Dieses Bitmuster wird einem D/A-Wandler zugeführt, der alle 22,6 µs aus dem Bitmuster einen analogen Wert ermittelt, Abbildung 77b. Dies ist noch nicht das ursprüngliche analoge Signal, sondern ein treppenförmiges analoges Signal. Es hat genauso viele Werte wie das Signal in digitaler Form. Liegt beispielsweise ein digitales 3-Bit-Signal vor, so gibt es 8 Bitmuster, von 000 bis 111. Mit dezimalen Zahlen ausgedrückt: 0 bis 7. Dies sind die Werte der treppenförmigen analogen Spannung. Das ursprüngliche kontinuierliche analoge Signal, Abbildung 77c, wird durch ein Filter aus der treppenförmigen Spannung ermittelt. Das Filter ermittelt Zwischenwerte, z.B.: 2,35, 4,4726 usw. Ein **kontinuierliches** Signal wird erzeugt.

Kenngrößen

Unabhängig vom Arbeitsprinzip eines D/A-Wandlers weist jeder Wandler charakteristische Kennwerte auf:

- Anzahl der Bits
- Maximale Ausgangsspannung
- Maximale Schrittweite
- Auflösung

Die **Anzahl der Bits** bestimmt, wie viele verschiedene Spannungswerte einschließlich 0 V von dem Wandler erzeugt werden können:

N Bit's: 2^N verschiedene Spannungswerte

Die **maximale Ausgangsspannung** wird durch die interne Schaltung und eine Referenzspannung festgelegt. Mögliche Werte sind z.B. 0 ... 10 V und -5 V ... +5 V.

Die **maximale Schrittweite** eines Wandlers wird durch die maximale Ausgangsspannung und die Anzahl der Bits bestimmt:

$$\Delta U_A = \frac{U_{A \max}}{2^N - 1}$$

Die Anzahl der Binärstellen wird als **Auflösung** eines Digital-Analog-Wandlers bezeichnet.

Digital-Analog-Wandler gibt es in verschiedenen Ausführungsformen:

- D/A-Wandler mit Spannungsausgang
- D/A-Wandler mit Stromausgang
- Multiplizierende D/A-Wandler

D/A-Wandler mit Spannungsausgang

Bei dem so genannten Parallelverfahren werden mit einem schaltbaren Spannungsteiler Teilspannungen erzeugt.

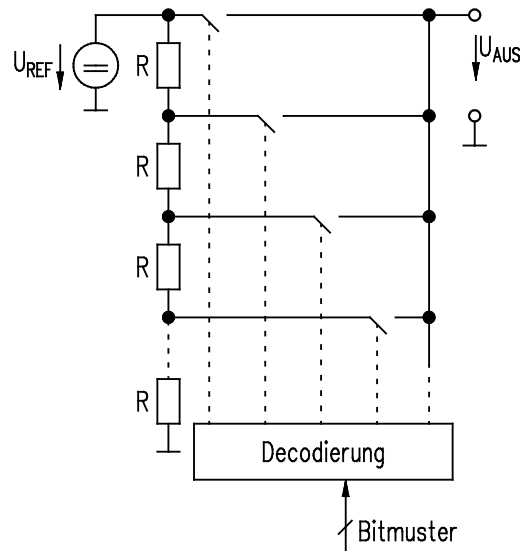


Abbildung 78 Parallelverfahren

Bei einem 3-Bit-D/A-Wandler entspricht die Referenzspannung dem maximalen Spannungswert: $111 \hat{=} 7 \text{ V}$. Die Anzahl der Widerstände und Schalter wird durch die Anzahl der Bitkombinationen bestimmt: 8 Bit \rightarrow 256 Spannungswerte = 256 Schalter. Mithilfe der Auswertelogik steuert das angelegte Bit-Muster die Schalter. Bei der Bit-Kombination 011 wird der Schalter geschlossen, der die Teilspannung von 3 V auf den Ausgang gibt.

Auf Grund des geringen Aufwandes wird die Umsetzung eines Bit-Musters mithilfe eines Addierers häufig durchgeführt.

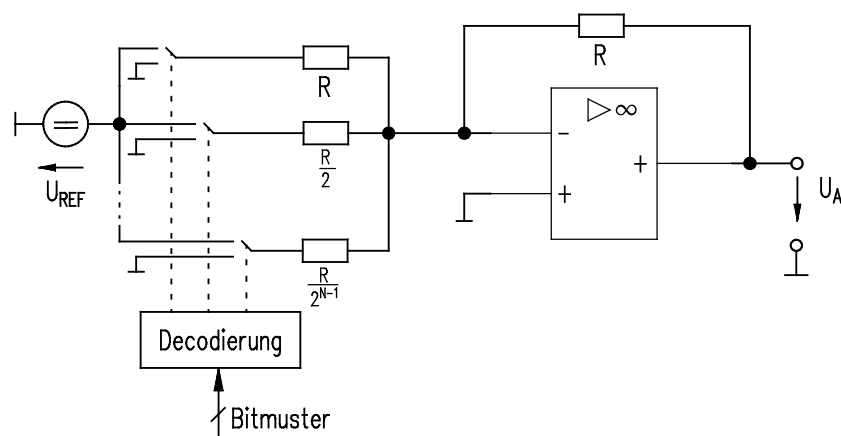


Abbildung 79 D/A-Wandler mit Additionsverstärker

Die Vorwiderstände sind dem Stellenwert des digitalen Bit-Musters entsprechend gewichtet:

Bei 3-Bit und einer Referenzspannung von 1 V gilt beispielsweise:

3-Bit → 3 Vorwiderstände mit den Werten R , $R/2$ und $R/4$;

Ausgangsspannungsbereich 0 V ... 7 V in 1 V-Schritten.

Neben der Stabilität und der Genauigkeit der Referenzspannung sind die Widerstände das schwächste Glied dieser Schaltung. Einen Ausweg bietet die Verwendung eines R-2R-Widerstandsnetzwerkes (Abbildung 80).

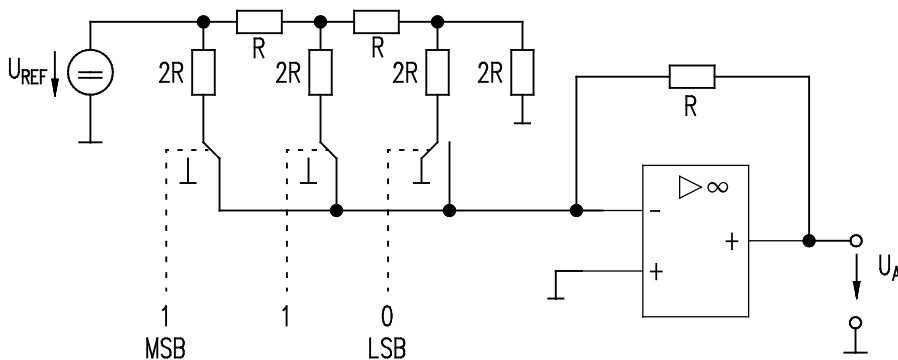


Abbildung 80 3-Bit-D/A-Wandler mit R-2R-Netzwerk

Bei einer Referenzspannung von 8 V und der abgebildeten Schalterstellung (110) geht das MSB auf Grund des Netzwerkes mit $U_{\text{Ref}}/2 = 4 \text{ V}$ in die Ausgangsspannung ein. Das 2. Bit ist ebenfalls gesetzt. Durch das R-2R-Netzwerk wird eine Spannung von $U_{\text{Ref}}/4 = 2 \text{ V}$ zur Ausgangsspannung addiert. Eine Ausgangsspannung von 6 V entspricht also der Bit-Kombination 110.

D/A-Wandler mit einem R-2R-Netzwerk besitzen eine höhere Genauigkeit als Addierer mit gewichteten Widerständen, wenn Widerstände mit gleichen Toleranzen verwendet werden. Das liegt daran, dass unabhängig vom Bit-Muster stets alle Widerstände in Betrieb sind und sich die Fehler bei entsprechender Widerstandswahl ganz bzw. teilweise kompensieren.

D/A-Wandler mit Stromausgang

Neben den D/A-Wandlern mit Spannungsausgang gibt es auch Wandler mit Stromausgang. Ihr wesentlicher Vorteil ist die geringe Umsetzzeit. Abbildung 81 zeigt einen 3-Bit-D/A-Wandler mit gewichteten Stromquellen.

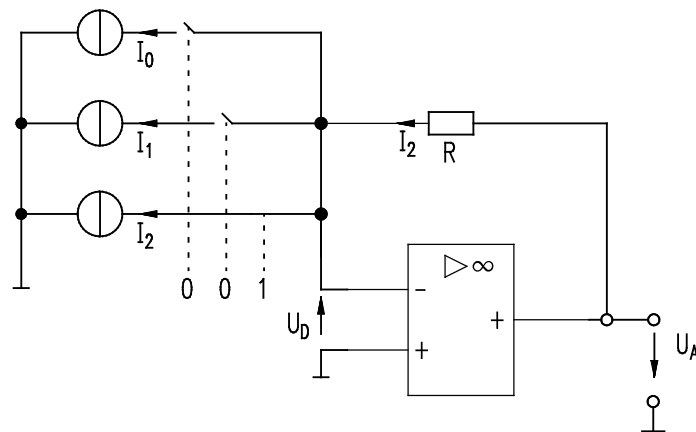


Abbildung 81 D/A-Wandler mit gewichteten Stromquellen

Wesentlicher Bestandteil der Schaltung ist ein Operationsverstärker. Da kein Strom in den Zuleitungen seiner Eingänge fließt, müssen die Ströme der Quellen über R fließen. Der Anschluss des OPs liegt praktisch auf Masse ($U_D = 0 \text{ V}$). Die Spannung an R, hervorgerufen durch die Ströme, entspricht der Ausgangsspannung U_A . In Abbildung 81 ist die Bit-Kombination 001 eingezeichnet. Die Ausgangsspannung beträgt daher $U_A = R \cdot I_2$. Bei den beiden anderen Stromquellen richtet sich der Wert des Stromes nach der Wertigkeit der zugeordneten Bitstelle. Beim abgebildeten Beispiel gilt:

$$I_1 = 2 \cdot I_2 \quad \text{und}$$

$$I_0 = 4 \cdot I_2$$

Multiplizierende D/A-Wandler

Sowohl die D/A-Wandler mit einem Spannungsausgang als auch die mit einem Stromausgang bieten die Möglichkeit die Referenzspannung zu ändern. In Abbildung 82 ist die Referenzspannung durch die Eingangsspannung U_E ersetzt.

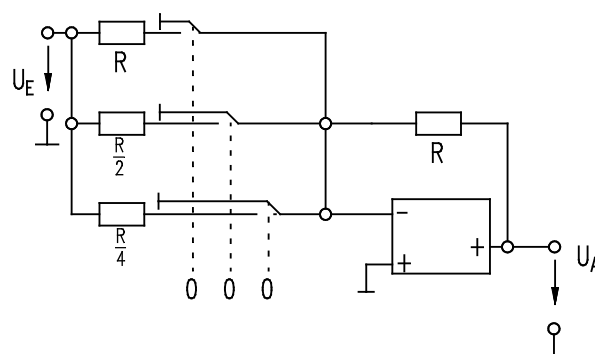


Abbildung 82 Multiplizierender D/A-Wandler

Am Ausgang dieser Schaltung erscheint die Spannung

$$U_A \approx U_E \cdot U_{\text{Bitmuster}}$$

Da die eine Eingangsgröße analog ist, die andere digital, ist ein derartiger Wandler eine hybride (gemischte analog-digitale) Komponente.

4.2 A/D-Wandler

In vielen Bereichen werden Rechner zur Erfassung und Verarbeitung von Messsignalen eingesetzt. Wenn große Genauigkeit, hohe Geschwindigkeiten und ein gutes Preis-Leistungsverhältnis angestrebt wird, werden analoge Größen digital verarbeitet. Durch den Einsatz eines Computers erreicht man ein hohes Maß an Flexibilität für die Erfassung und Auswertung physikalischer Größen. In der nachfolgenden Abbildung ist das Beispiel einer Temperaturregelung dargestellt.

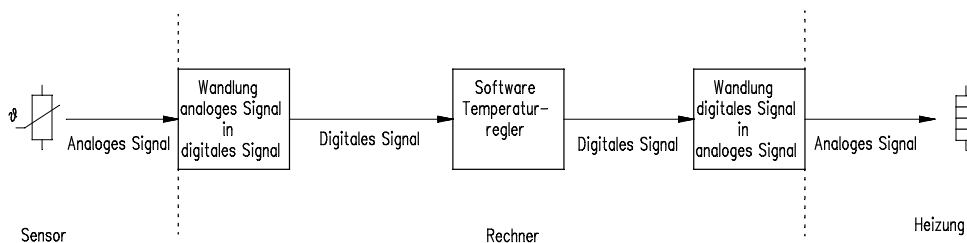


Abbildung 83 Beispiel Temperaturregelung

Der Sensor liefert ein analoges Signal, eine Spannung, die proportional zur Temperatur ist. Da der Computer nur digitale Signale verarbeiten kann, wird das analoge Signal mit einem Analog-Digital-Wandler (A/D-Wandler) in ein digitales Signal umgesetzt.

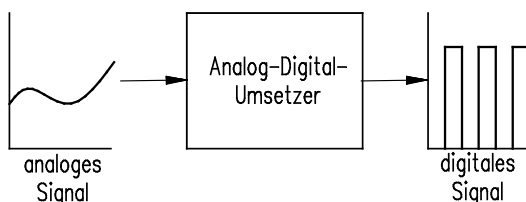


Abbildung 84 Prinzip einer Analog-Digital-Umsetzung

In vielen Fällen benötigt der A/D-Wandler eine zusätzliche Peripherie, um richtig arbeiten zu können, einen Abtast/Halterverstärker. Das digitale Signal wird dann mittels Software aufbereitet und über einen D/A-Wandler wird die Heizung entsprechend angesteuert.

Abbildung 85 gibt den schematischen Aufbau eines A/D-Wandlers wieder.

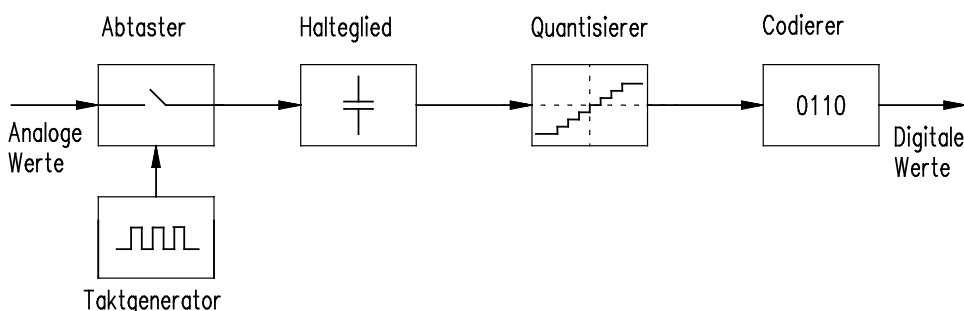


Abbildung 85 Schematischer Aufbau eines A/D-Wandlers

Hauptbestandteil des Wandlers ist der **Quantisierer**. Er wandelt die Eingangsspannung in eine dazu proportionale digitale Zahl um. In Abbildung 86 ist die Umsetzung für einen 3-Bit-Wandler mit 1 V Schritten dargestellt. Eine Eingangsspannung von 3 V wird in das 3-Bit-Signal 011 umgesetzt.

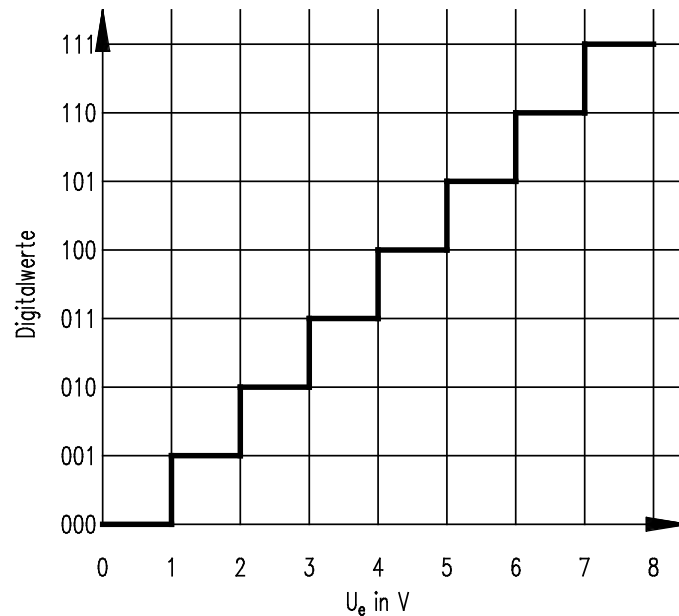


Abbildung 86 Eingangsspannung und Ausgangssignal eines Analog-Digital-Umsetzers

Beträgt die Eingangsspannung 3,5 V wird diese ebenfalls in den Digitalwert 011 umgesetzt. Es treten also Ungenauigkeiten auf Grund der Digitalisierung auf. Im ungünstigsten Fall steht die Bit-Kombination 011 für eine Spannung von 3,99999 V. Diese Abweichungen wirken sich auf die Signalwerte wie ein überlagertes Störsignal aus, das als **Quantisierungsrauschen** bezeichnet wird. Die Anzahl der Bit's bestimmt die Größe des Quantisierungsrauschens.

Bei schnellen Signalen kann der Quantisierer das analoge Signal nicht direkt verarbeiten, da er für die Umsetzung eine bestimmte Zeit benötigt. Durch eine zusätzliche Schaltung muss sichergestellt werden, dass während der Wandlungszeit des Quantisierers, das Eingangssignal konstant ist. Diese Aufgabe übernimmt das **Abtast-Halte-Glied** auch Sample and Hold-Schaltung genannt. Abbildung 87 zeigt das Prinzip einer derartigen Schaltung.

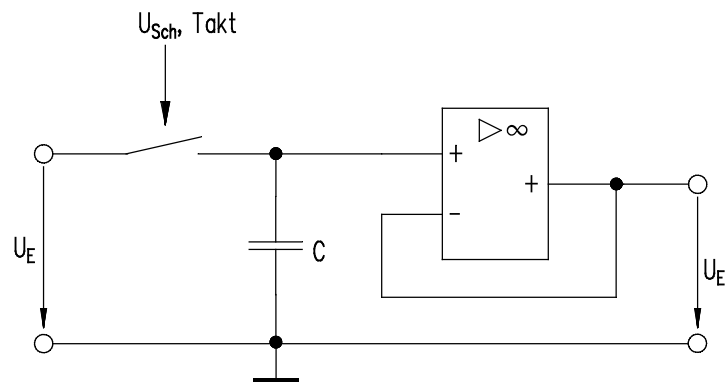


Abbildung 87 Prinzip einer Sample and Hold-Schaltung

Der Schalter wird durch die Spannung U_{Sch} gesteuert. Das Eingangssignal U_E wird in einem Zeitraster abgetastet. Der zu diesem Zeitpunkt erfasste Spannungswert wird im Kondensator bis zur nächsten Abtastung gespeichert, graue Markierung. Der Eingangsspannungsverlauf der Schaltung ist in Abbildung 88 wiedergegeben.

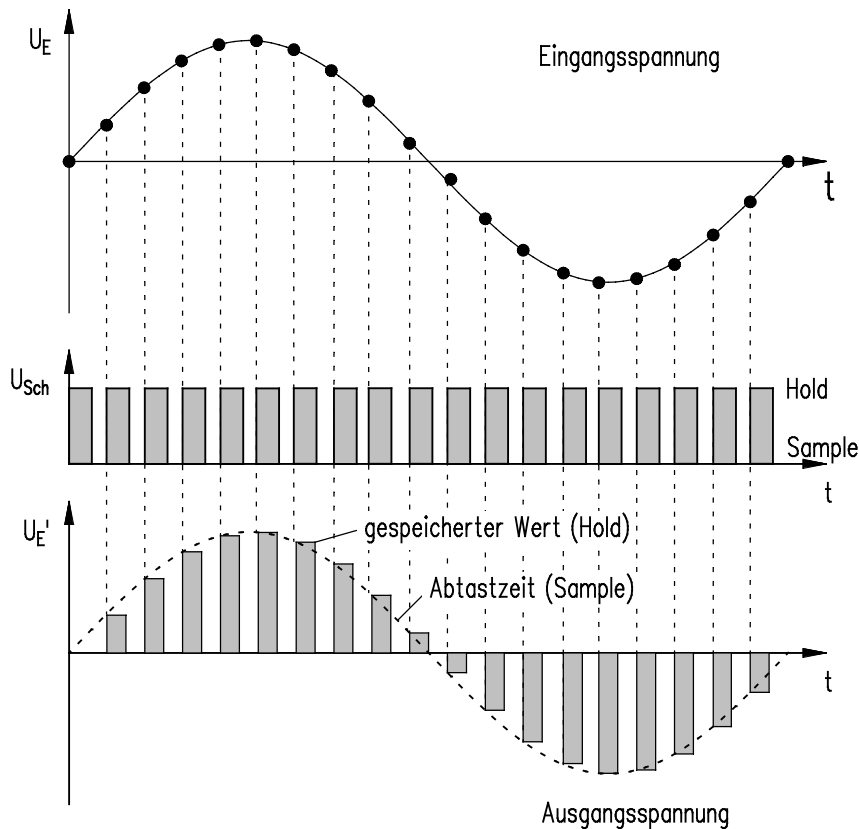


Abbildung 88 Ein- und Ausgangssignal einer Sample and Hold Schaltung

Da der Quantisierer nach jeder Abtastung ein digitales Signal erzeugt, sind die Datenmengen unter Umständen sehr groß. Es stellt sich die Frage: wie weit darf die Abtastrate verringert werden. Beantworten lässt sich diese Frage mit dem Abtasttheorem von Shannon.

Dazu betrachtet man das bei einer Abtastung entstehende Frequenzspektrum, die Verteilung der Frequenzanteile. Ein bekanntes Beispiel ist die Audio-CD: Höchste Frequenz $f_E = 20 \text{ kHz}$. Abgetastet wird mit einer Frequenz von $f_A = 44,1 \text{ kHz}$. Abbildung 89 zeigt das Frequenzspektrum der Abtastung. Es wird ein idealer Tiefpass mit der Grenzfrequenz $f_E = 20 \text{ kHz}$ benutzt.

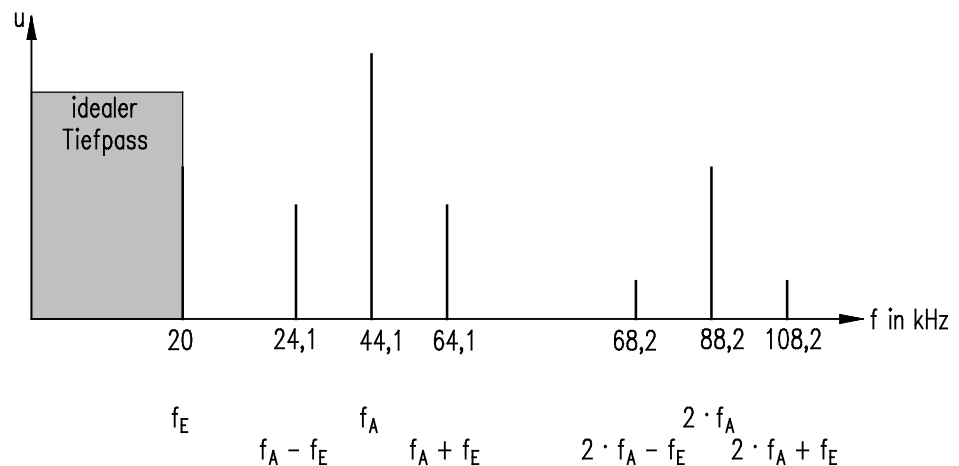


Abbildung 89 Frequenzspektrum einer abgetasteten Sinusschwingung

Neben der Frequenz f_E der abzutastenden Sinusschwingung und der Abtastfrequenz f_A , existieren weitere Frequenzanteile.

Bei der Rekonstruktion des ursprünglichen Analogsignals, werden die höheren Frequenzanteile der Ausgangsspannung mit einem Tiefpass herausgefiltert, damit nur noch das eigentliche Nutzsignal (grauer Balken) übrig bleibt. Er enthält nur Frequenzen des Nutzsignals. Bei Verkleinerung der Abtastfrequenz muss sichergestellt sein, dass die untere Seitenbandfrequenz $f_A - f_E$ nicht kleiner wird als die Grenzfrequenz des Tiefpasses. Beim Einsatz eines idealen Tiefpasses darf die Seitenbandfrequenz $f_A - f_E$ minimal der Grenzfrequenz des Tiefpasses entsprechen. Die kleinstmögliche Abtastfrequenz ist daher 40 kHz, Abbildung 90.

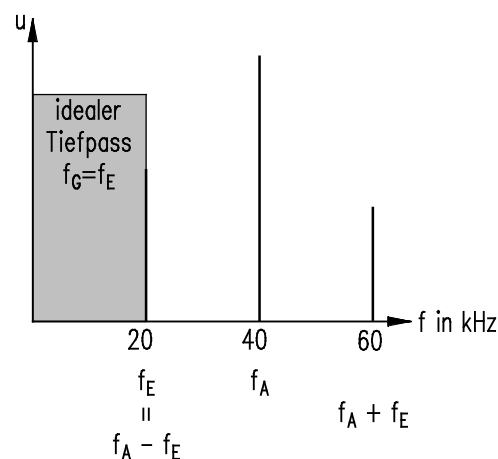


Abbildung 90 Frequenzspektrum bei $f_A = 2 \cdot f_E$

Daraus ergibt sich:

$$f_A \geq 2 \cdot f_E$$

Die Abtastfrequenz f_A muss mindestens doppelt so groß sein wie die abzutastende Frequenz f_E .

Entsprechend dem Abtasttheorem nach SHANNON muss eine Sinusspannung nur einmal pro Halbwelle abgetastet werden, um sie nach der Digital-Analog-Umsetzung wieder zurückzugewinnen. Voraussetzung ist eine kontinuierlich anliegende Sinusspannung und die Verwendung eines idealen Tiefpasses.

Wird die Abtastfrequenz z.B. mit 34 kHz gewählt, so liegt die bei der Abtastung entstehende Frequenz $f_A - f_E = 14$ kHz im Nutzbereich von 0 ... 20 kHz. Dieses Signal wird nicht heraus gefiltert. Es liegt eine Störung vor. Sie wird als Aliasing-Rauschen bezeichnet.

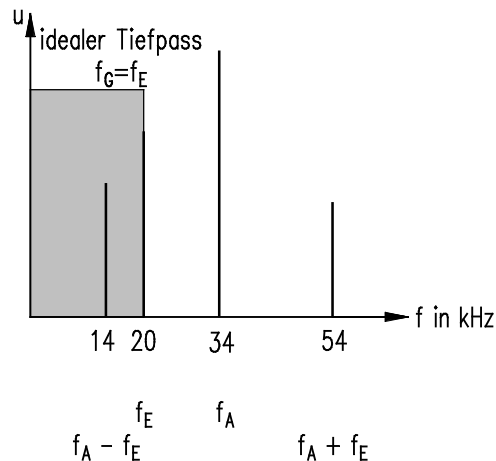


Abbildung 91 Frequenzspektrum bei $f_A < 2 \cdot f_E$

Um diese Störungen zu verhindern, wird vor der eigentlichen Abtastung die Bandbreite des Eingangssignals mit einem so genannten Aliasingfilter begrenzt. Das entsprechende Blockschaltbild sieht folgendermaßen aus:

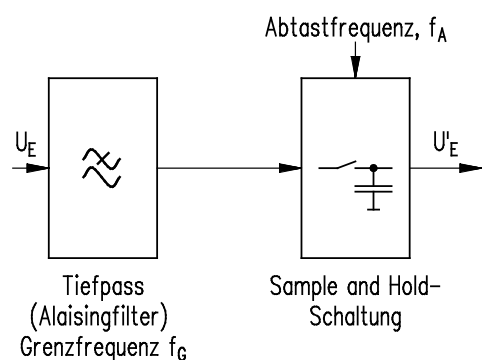


Abbildung 92 Sample and Hold-Schaltung mit Filter

Da für die Grenzfrequenz des Tiefpasses immer $f_G < \frac{f_A}{2}$ gewählt, ist das Abtasttheorem immer erfüllt.

Das Shannon-Theorem geht von einem idealen Tiefpass aus, den es in der Praxis nicht gibt. Üblicherweise wählt man daher die Abtastfrequenz mit $f_A = 2,5$ bis $5 \cdot f_E$.

Die letzte Baugruppe des A/D-Wandlers, der Codierer, passt das digitale Signal dem digitalen System des Rechners an. Bei analogen Signalen, die sowohl positive als auch negative Werte aufweisen können, können die Werte unterschiedlich dargestellt werden. In der folgenden Tabelle sind zwei Möglichkeiten dargestellt:

	Zweierkomplement-Code	Offsetbinärkode
3	011	111
2	010	110
1	001	101
0	000	100
-1	111	011
-2	110	010
-3	101	001
-4	100	000

Tabelle 26 Kennwerte von A/D-Umsetzern

Einige charakteristische Kennwerte von Analog-Digital-Umsetzern sind:

- Auflösung
- Quantisierungsfehler
- Umsetzzeit

Die Wortbreite des A/D-Wandlers legt die Anzahl der möglichen Amplitudenintervalle des Quantisierers fest.

3 Bit → 8 Amplitudenintervalle

8 Bit → 256 Amplitudenintervalle

Je größer die Wortbreite ist, desto besser ist die **Auflösung** des Wandlers:

$$\Delta U = \frac{U_{\max}}{2^N}$$

ΔU - Spannungsauflösung
 U_{\max} - maximale Eingangsspannung

Der Begriff **Quantisierungsfehler** wurde bereits an anderer Stelle erläutert. Die kontinuierliche Signalform des Eingangssignals wird durch den Quantisierer in ein nicht kontinuierliches Signal umgesetzt. Der kleinste Amplitudenschritt, die geringste Auflösung, ist durch das niederwertigste Bit (LSB) gekennzeichnet. Bei einem Digitalmultimeter ist dies die kleinste angezeigte Stelle. Dieser Fehler kann niemals kleiner als $\pm \frac{1}{2}$ LSB sein. In dem Beispiel aus Abbildung 86 beträgt der Fehler 1 V bzw. ist gleich +1 LSB. D.h. bei der Bit-Kombination 101 kann der analoge Wert 5 V sein, aber auch 5,9999 V, 1 LSB mehr.

Eine weitere wichtige Kenngröße ist die **Umsetzzeit** (conversion time), auch Wandlungszeit genannt. Sie beschreibt, wie lange der A/D-Wandler benötigt, um das anliegende analoge Signal in digitaler Form bereitzustellen. Diese Zeit bestimmt maßgeblich das Einsatzgebiet des Wandlers.

Die A/D-Wandler können in zwei Gruppen eingeteilt werden:

- Momentanwertumsetzer
- Integrierende Umsetzer

Momentanwertumsetzer

Es gibt drei Verfahren:

- Zählverfahren
- Sukzessive Approximation
- Flash-Wandler (Parallelverfahren)

Ein Wandler nach dem Zählverfahren besteht im Wesentlichen aus folgenden Komponenten:

- Komparator
- Taktgenerator
- Zähler
- Linearer Sägezahn-generator

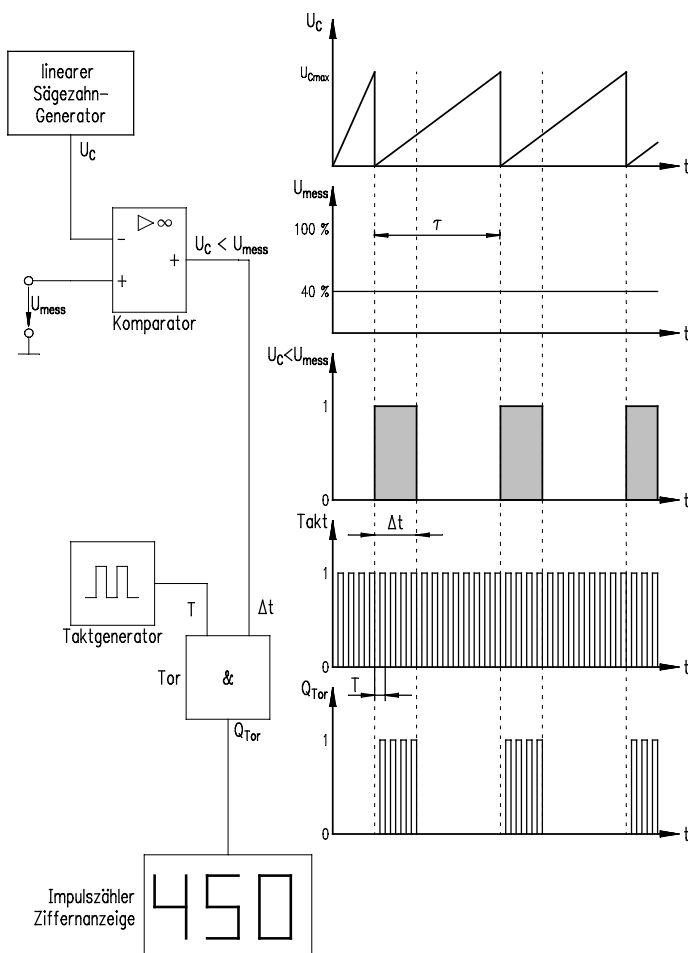


Abbildung 93 Prinzipschaltbild eines Wandlers nach dem Zählverfahren

Abbildung 93 gibt den Aufbau des Wandlers wieder. Die Umsetzung wird durch Rücksetzen des Zählers auf Null gestartet. Der Sägezahngenerator stellt eine linear ansteigende mit dem Takt gekoppelte Spannung U_C zur Verfügung. Mit jedem Taktimpuls erhöht sich die Sägezahnspannung um den gleichen Betrag. Der Spannungswert der Sägezahnspannung wird durch den Komparator mit dem Wert des anliegenden Eingangssignals U_E verglichen. Ist die Spannung U_E größer als die Spannung U_C , so erscheint am Ausgang des Komparators eine 1. Das Tor wird geöffnet (UND-Gatter) und die Impulse des Taktgenerators gelangen zum Zähler. Jeder Impuls des Takt-Generators erhöht den Zählerstand um eins, siehe Abbildung 94.

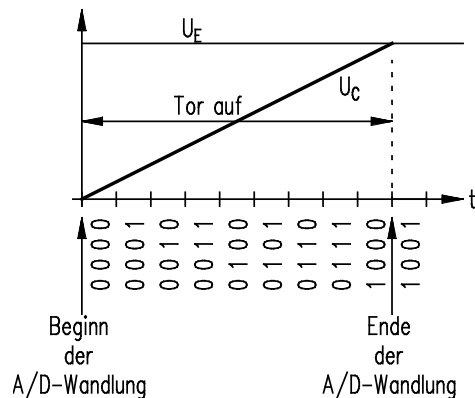


Abbildung 94 Zeitlicher Verlauf der A/D-Wandlung

Ist die Spannung U_C größer als U_E , so erscheint am Ausgang des Komparators eine 0. Das Tor wird geschlossen. Die Impulse des Taktgenerators gelangen nicht mehr zum Zähler. Der Zähler hat den Endstand 1001 erreicht. Der Zählerstand wird gespeichert und angezeigt. Anschließend setzt die Steuerelektronik den Zähler auf Null und der Wandlungsvorgang beginnt erneut.

Die Umsetzungsgeschwindigkeit bzw. die Umsetzzeit wird durch zwei Faktoren bestimmt, die interne Taktrate des Generators (f_T) und die Anzahl der benötigten Quantisierungsschritte. Der ungünstigste Fall liegt vor, wenn der maximal mögliche Eingangsspannungswert anliegt. Der A/D-Wandler benötigt $2^N - 1$ Schritte und somit die Zeit:

$$T_U = \frac{(2^N - 1)}{f_T}$$

Da die Umsetzzeit direkt durch die Wortbreite beeinflusst wird, ist dieses Verfahren nur für langsame Signalvorgänge geeignet.

Ein wesentlicher Nachteil dieses Wandlers ist, dass der Zähler immer bei Null beginnend aufwärts zählt. In den meisten Fällen weicht der Folgewert aber nur geringfügig vom vorherigen Wert ab. Durch eine Abwandlung der Schaltung kann dieser Nachteil vermieden werden.

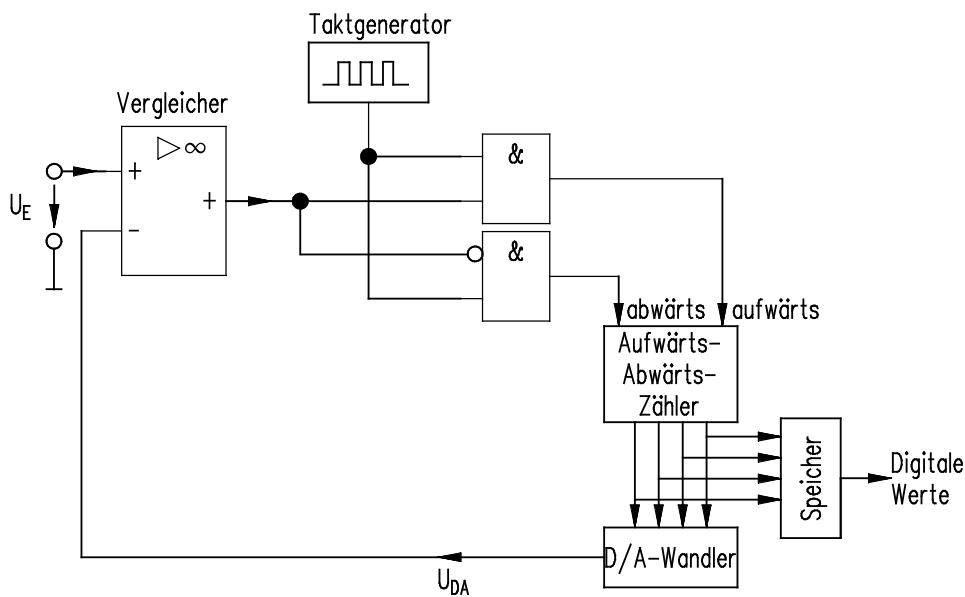


Abbildung 95 A/D-Wandler mit Aufwärts-Abwärtszähler

Statt eines Sägezahngenerators wird ein D/A-Wandler verwendet. Er setzt den Zählerstand in die analoge Spannung U_{DA} um. Die Vergleichsspannung ist dadurch immer mit dem Zähler synchronisiert. Nach jedem Taktimpuls entscheidet die interne Steuer elektronik, ob der nächste Taktimpuls den vorhandenen Zählerstand erhöht oder verringert.

Bei sehr großen Eingangsspannungssprüngen ergeben sich allerdings keine wesentlichen Verbesserungen.

Erheblich schneller ist ein A/D-Wandler nach dem Verfahren der **sukzessiven Approximation**.

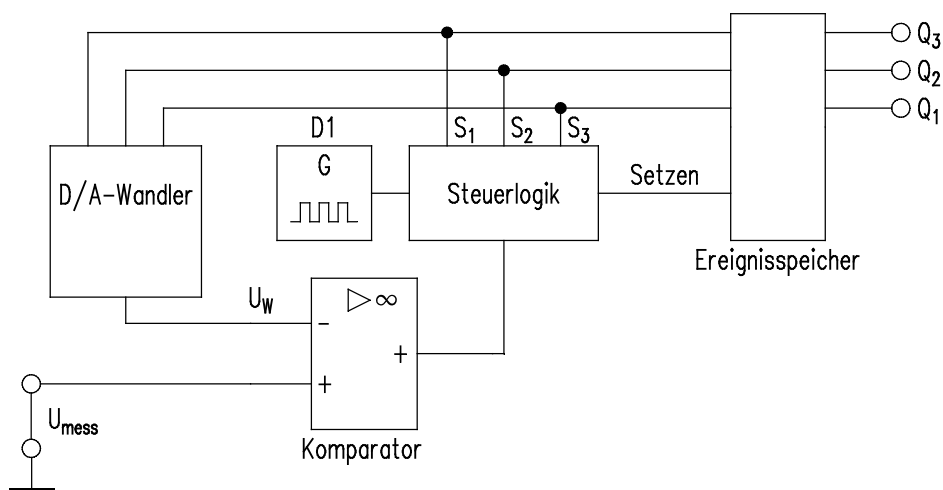


Abbildung 96 Prinzipschaltung eines 3-Bit Analog-Digital-Umsetzers nach dem Verfahren der sukzessiven Approximation

Zur Vereinfachung der Funktionsbeschreibung wird ein 3-Bit Analog-Digital-Umsetzer betrachtet, der eine Eingangsspannung von 0 bis 8 V hat.

Die Umsetzung beginnt damit, dass die Steuerlogik das höchstwertige Bit S_3 im Datenwort S_1 bis S_3 zunächst auf logisch „1“ setzt, die beiden übrigen auf logisch „0“, 100.

Die folgende Wägung wird zeigen, ob die Vorgabe richtig war.

Der schnelle Digital-Analog-Umsetzer liefert gemäß dem angelegten Digitalwort -100- eine Ausgangsspannung von $U_W = 4 \text{ V}$ an den Komparator (Vergleicher), der beide Spannungen U_W und U_{mess} miteinander vergleicht.

Ist U_{mess} größer als U_W , liefert der Komparator am Ausgang eine logische „1“ und die Steuerlogik setzt das Bit S_3 endgültig auf „1“.

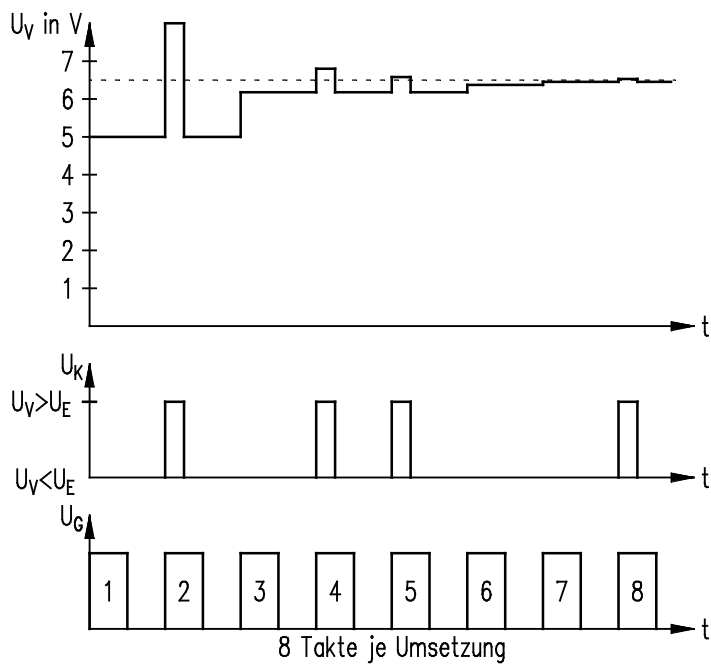
Ist U_{mess} aber kleiner als U_W , bewirkt das „0“-Signal am Komparatorausgang, dass das Bit S_3 auf „0“ zurückgesetzt wird.

Damit ist der erste Schritt der Umsetzung vollzogen und der Wert des Bit S_3 steht fest.

Der zweite Schritt der Umsetzung verläuft sinngemäß und beginnt damit, dass die Steuerlogik zunächst das zweithöchste Bit S_2 mit „1“-Signal vorgibt, 110 und damit eine erneute Wägung einleitet.

Nach drei Annäherungsschritten liegt das richtige Datenwort S_1 bis S_3 an den Ausgängen der Steuerlogik. Diese gibt abschließend einen Setzimpuls an den Ergebnisspeicher, sodass die Datenausgänge Q_1 bis Q_3 aktualisiert werden. Somit benötigt ein N-Bit Analog-Digital-Umsetzer, der nach dem Verfahren der sukzessiven Approximation arbeitet, N Schritte bzw. Wägungen um die Eingangsspannung zu wandeln.

Abbildung 97 veranschaulicht noch einmal den Abgleichvorgang für einen 8-Bit Analog-Digital-Umsetzer. Bei diesem Beispiel beträgt die Referenzspannung $U_{\text{ref}} = 10 \text{ V}$ und die Eingangsspannung $6,5 \text{ V}$.



Takt	1	2	3	4	5	6	7	8
D ₇	1	1	1	1	1	1	1	1
D ₆	0	0	0	0	0	0	0	0
D ₅	0	0	1	1	1	1	1	1
D ₄	0	0	0	0	0	0	0	0
D ₃	0	0	0	0	0	0	0	0
D ₂	0	0	0	0	0	1	1	1
D ₁	0	0	0	0	0	0	1	1
D ₀	0	0	0	0	0	0	0	0

= D_A

Digitalausgang

Abbildung 97 Abgleichvorgang des Analog-Digital-Umsetzers nach dem Verfahren der sukzessiven Approximation

Takt 1: $U_{\text{mess}} > U_W$ D₇ bleibt auf „1“ gesetzt.
 Takt 2: $U_{\text{mess}} < U_W$ D₆ wird zurückgesetzt, „0“.
 Takt 3: $U_{\text{mess}} > U_W$ D₅ bleibt auf „1“ gesetzt.
 Takt 4: $U_{\text{mess}} < U_W$ D₄ wird zurückgesetzt, „0“.
 Takt 5: $U_{\text{mess}} < U_W$ D₃ wird zurückgesetzt, „0“.
 Takt 6: $U_{\text{mess}} > U_W$ D₂ bleibt auf „1“ gesetzt.
 Takt 7: $U_{\text{mess}} > U_W$ D₁ bleibt auf „1“ gesetzt.
 Takt 8: $U_{\text{mess}} < U_W$ D₀ wird zurückgesetzt, „0“.

Die kürzesten Wandlungszeiten ermöglicht das **Parallelverfahren**, der Flash-Wandler. Der Parallelumsetzer wandelt die angelegte Messspannung in einem Schritt in eine äquivalente Dualzahl um. Die folgende Abbildung zeigt einen solchen Umsetzer für 3 Bit.

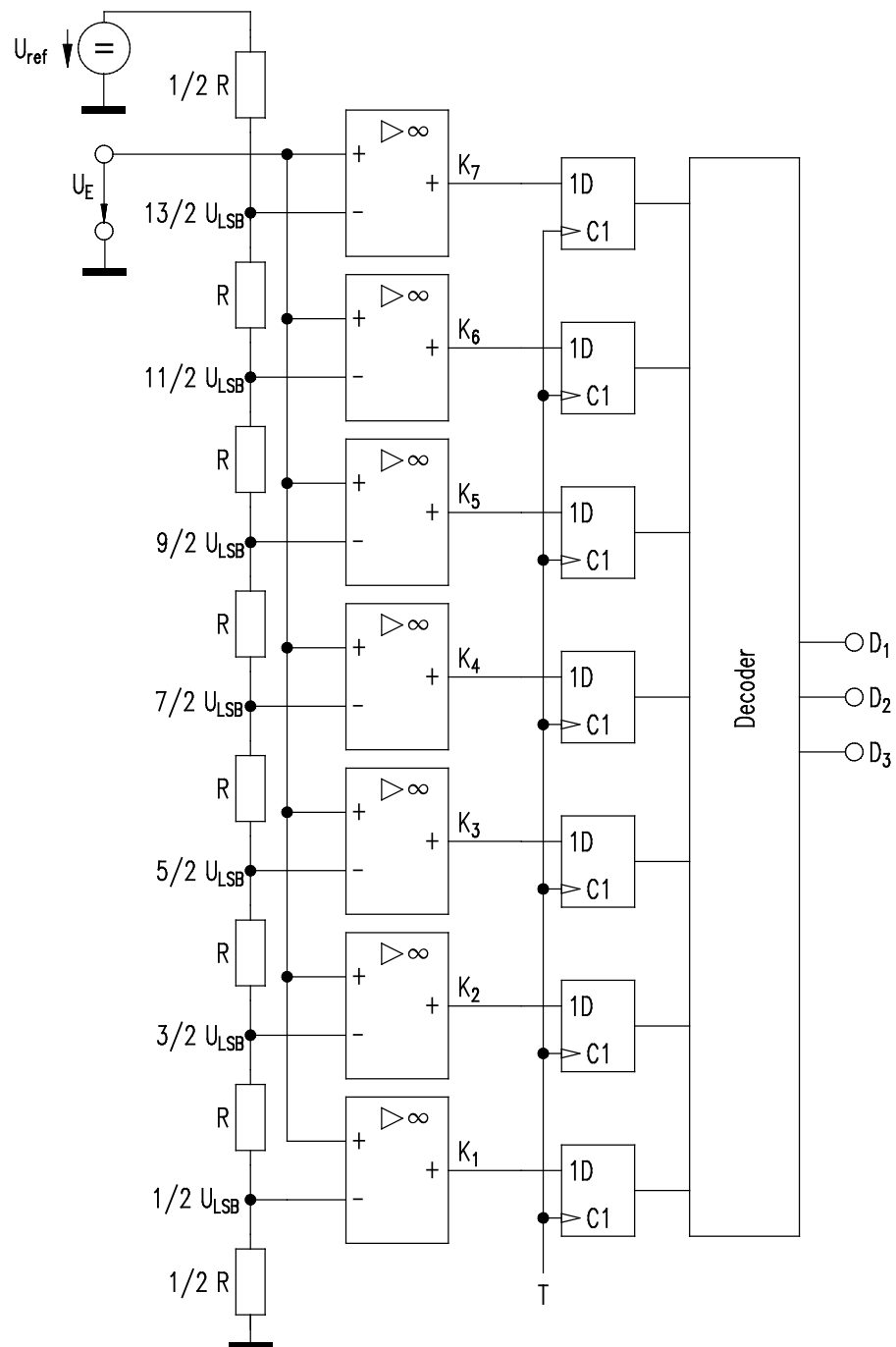


Abbildung 98 3-Bit Analog-Digital-Umsetzer nach dem Parallelverfahren

Man benötigt für einen 3-Bit Analog-Digital-Wandler sieben Komparatoren. Die zugehörigen sieben Referenzspannungen werden mithilfe eines Spannungsteilers erzeugt.

Bei diesem Wandler ist der Spannungsteiler so aufgebaut, dass bei einer Referenzspannung von 7 V die Teilspannungen 0,5 V, 1,5 V, 2,5 V usw. erzeugt werden.

Legt man nun eine Eingangsspannung von 3 V an, so setzen die drei untersten Komparatoren ein „1“-Signal am Ausgang ($U_E > U_{TEIL}$), die anderen Komparatoren ein „0“-Signal ($U_E < U_{TEIL}$). Die Signale werden zwischengespeichert. Der nachfolgende Decoder geniert dann das digitale Signal: 011. In der folgenden Wahrheitstabelle ist der Zusammenhang zwischen den Komparatorzuständen und der zugehörigen Dualzahl dargestellt.

Eingangsspannung	Komparatorzustände							Dualzahl			Dezimal-äquivalent
U_E / U_{LSB}	k_7	k_6	k_5	k_4	k_3	k_2	k_1	D_2	D_1	D_0	Z
0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	1	0	0	1	1
2	0	0	0	0	0	1	1	0	1	0	2
3	0	0	0	0	1	1	1	0	1	1	3
4	0	0	0	1	1	1	1	1	0	0	4
5	0	0	1	1	1	1	1	1	0	1	5
6	0	1	1	1	1	1	1	1	1	0	6
7	1	1	1	1	1	1	1	1	1	1	7

Tabelle 27 Zusammenhang zwischen den Komparatorzuständen und dem Ausgang des Analog-Digital-Umsetzers

Eine vor allem in der Messtechnik eingesetzte Gruppe von Wandlern sind die **integrierenden Wandler**. Aufgrund ihrer Arbeitsweise benötigen sie keine Sample and Hold Schaltung. Mit relativ geringem Aufwand, kann man sehr gute Genauigkeiten erzielen. Nachteilig ist die langsame Umsetzung. Dies schränkt das Einsatzgebiet erheblich ein.

Beim Dual Slope-Verfahren wird nicht nur die Referenzspannung, sondern auch die Eingangsspannung integriert. Die Abbildung 99 gibt die Prinzipschaltung eines Analog-Digital-Wandlers nach dem Dual Slope-Verfahren wieder.

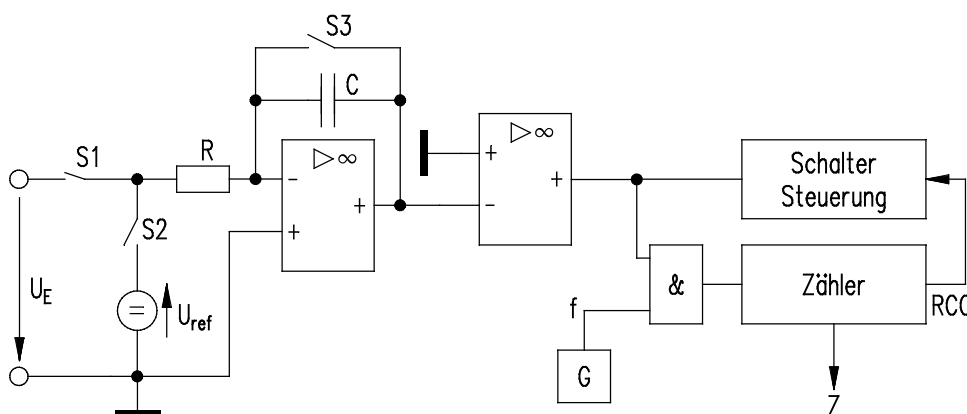


Abbildung 99 Prinzipschaltung eines dual slope Analog-Digital-Wandlers

In der Abbildung sind im Ruhezustand die Schalter S_1 und S_2 offen und S_3 geschlossen. Der Kondensator ist entladen. In dieser Stellung wird die Ausgangsspannung des Integrators auf null gesetzt.

Bei Messbeginn wird der Zähler gelöscht, der Schalter S_3 geöffnet und S_1 geschlossen, wodurch die Eingangsspannung U_E den Kondensator auflädt. Der Komparator gibt den Taktgenerator frei. Das Ende der Messdauer ist erreicht, wenn der Zähler überläuft und damit wieder auf null steht.

Die Ausgangsspannung des Integrators ist proportional zum Mittelwert der Eingangsspannung U_E .

Anschließend erfolgt die Umschaltung auf die Referenzspannung, S_1 offen, S_2 geschlossen. Die Wahl des Vorzeichens der Referenzspannung erfolgt so, dass der Kondensator entladen wird. Gleichzeitig wird der Zähler wieder gestartet. Erreicht die Spannung am Ausgang des Integrators den Wert Null, wird der Zählvorgang gestoppt. Der Zählerstand entspricht dem Digitalwert des analogen Eingangssignals U_E .

Ein Messzyklus setzt sich somit aus zwei Zeitabschnitten zusammen:

Zeitabschnitt t_1 : In dieser Zeitspanne wird der Kondensator durch die Messspannung aufgeladen, die Messspannung wird integriert.

Zeitabschnitt t_2 : Nach Betätigen der Schalter entlädt die Referenzspannung U_{ref} den Kondensator, und es wird der digitale Wert ermittelt.

Aus den zwei zuvor vorgestellten Integrationsphasen ergibt sich die Bezeichnung: Doppelintegrations-Verfahren oder Dual Slope-Verfahren. Die Spannungsverläufe des Integrators werden in der Abbildung 100 an zwei unterschiedlichen Eingangsspannungen dargestellt.

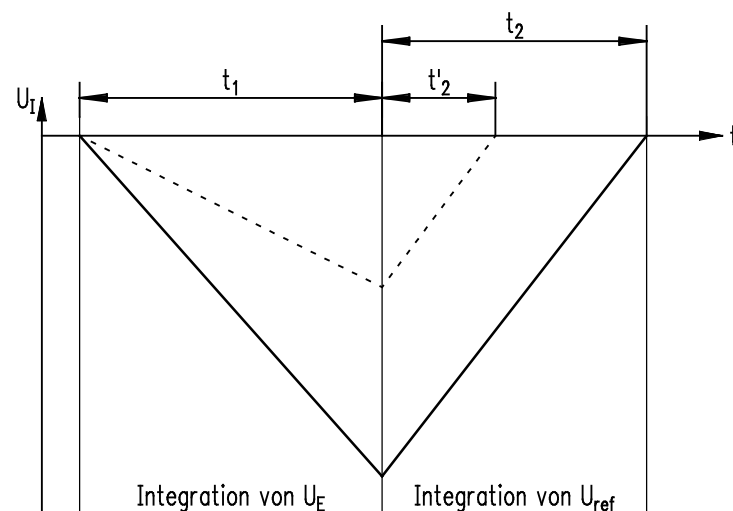


Abbildung 100 Zeitlicher Verlauf der Integratorspannung für zwei verschiedene Eingangsspannungen

Wie aus der Funktionsweise ersichtlich ist, fließt der Momentanwert der Messspannung nicht in das Ergebnis mit ein, sondern nur ihr Mittelwert über die Messzeit t_1 . Störende Wechselspannungsanteile werden bei richtiger Auslegung vollständig unterdrückt.

Aufgabe 1

Kreuzen Sie an, welcher A/D-Wandler zur Umsetzung der aufgeführten Signale einzusetzen ist!

	Parallel-Wandler Flash-Wandler	Dual-Slope-Wandler
Gleichspannungsmessung mit großer Genauigkeit		
Videosignale		

Aufgabe 2

Ein analoges Messsignal mit einem Frequenzumfang von 0 Hz bis 10 kHz, soll ohne Abtastfehler, Alasingrauschen, abgetastet werden. Sie haben die Möglichkeit mittels Software folgende Abtastraten einzustellen:

15 kHz, 20 kHz, 21,5 kHz, 24 kHz

Geben Sie die Abtastfrequenzen an, die unter realen Bedingungen möglich sind!

Aufgabe 3

Wie viele Schritte benötigen die aufgeführten 8-Bit-A/D-Wandler maximal, um einen Spannungswert in einen digitalen Wert umzuwandeln?

A/D-Wandlerverfahren	Anzahl der Schritte für die Wandlung
Zählverfahren	
Sukzessive Approximation	
Parallelverfahren	

Aufgaben

Lösungen

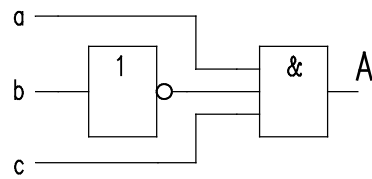
Lösungsanhang

1 Logische Verknüpfungen

Aufgabe 1.1

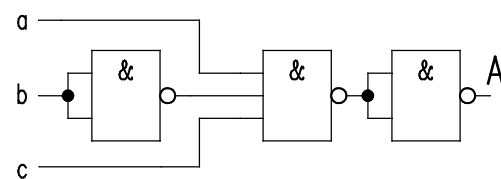
$$A = a \bar{b} c$$

Aufgabe 1.2



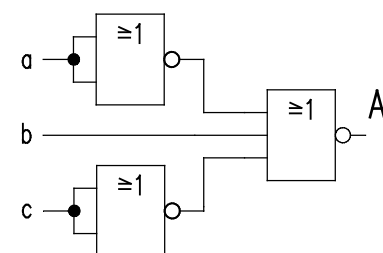
Aufgabe 1.3

$$A = \overline{\overline{a} \bar{b} c}$$



Aufgabe 1.4

$$A = \overline{\overline{a} \vee b \vee \overline{c}}$$

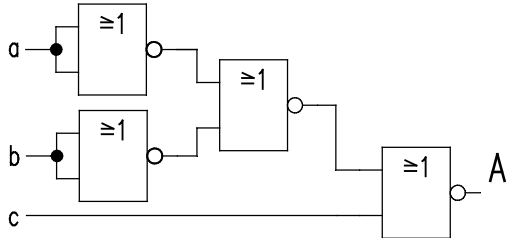


Aufgabe 2.1

$$A = \overline{a b \vee c}$$

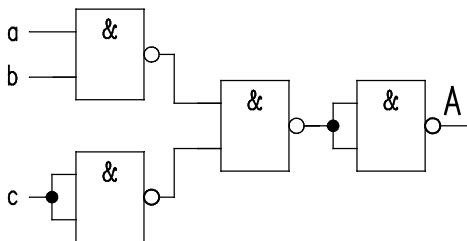
Aufgabe 2.2

$$A = \overline{\overline{a} \vee \overline{b} \vee c}$$



Aufgabe 2.3

$$A = \overline{\overline{a} \overline{b} \overline{c}}$$



Aufgabe 3.1

$$A = \overline{ab \vee c \vee abc}$$

Aufgabe 3.2

$$A = \overline{(\overline{a} \vee \overline{b}) \vee c \vee a b c}$$

$$A = a b \overline{c} \vee a b c$$

$$A = a b (\overline{c} \vee c)$$

$$A = ab$$

Aufgabe 4.1

a	b	c	A
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

Aufgabe 4.2

$$A = \overline{a}\overline{b}\overline{c} \vee \overline{a}b\overline{c} \vee \overline{a}bc \vee a\overline{b}\overline{c}$$

Aufgabe 4.3

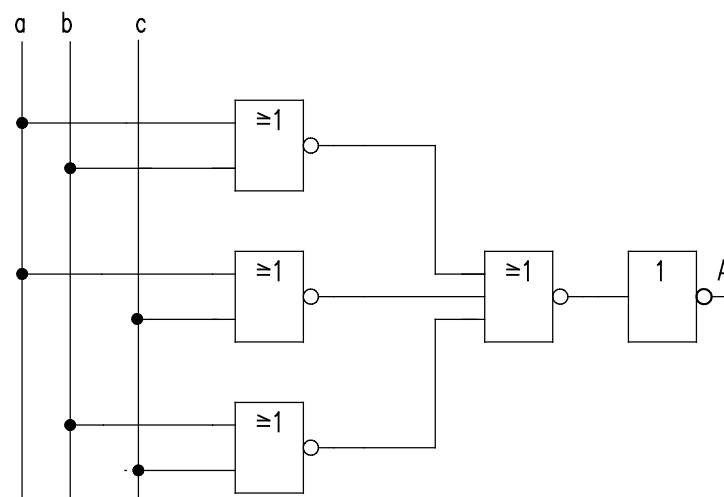
$$A = \overline{a}b \vee \overline{a}c \vee bc$$

Aufgabe 4.4

$$A = \overline{\overline{a}\overline{b}} \vee \overline{\overline{a}\overline{c}} \vee \overline{\overline{b}\overline{c}}$$

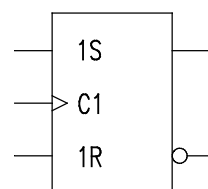
$$A = \overline{a \vee b} \vee \overline{a \vee c} \vee \overline{b \vee c}$$

$$A = \overline{\overline{a \vee b} \vee \overline{a \vee c} \vee \overline{b \vee c}}$$



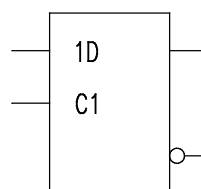
2 Schaltungen mit Speicherverhalten

Aufgabe 1



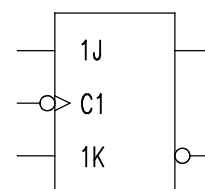
1)

Positiv
flankengesteuertes
SR-Flip-Flop



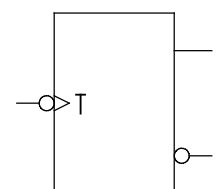
2)

Zustands-
gesteuertes
D-Flip-Flop



3)

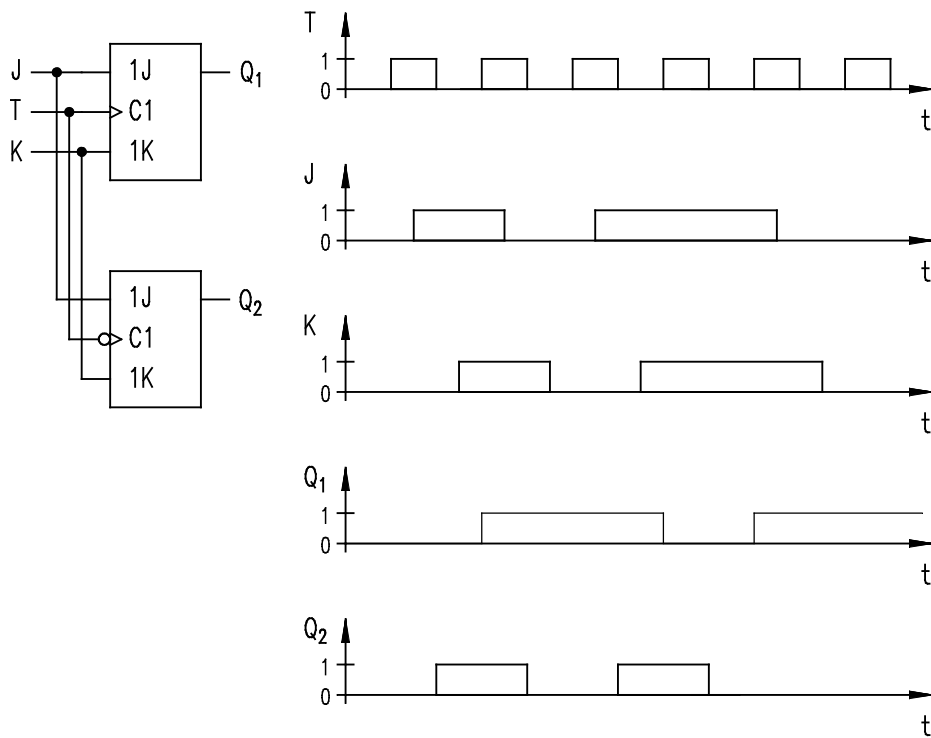
Negativ
flankengesteuertes
JK-Flip-Flop



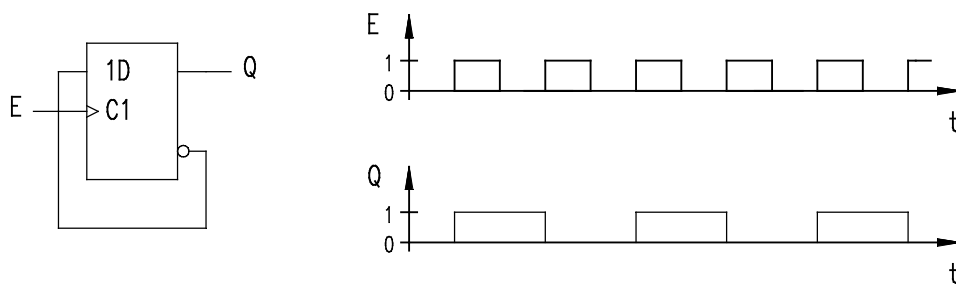
4)

Negativ
flankengesteuertes
T-Flip-Flop

Aufgabe 2

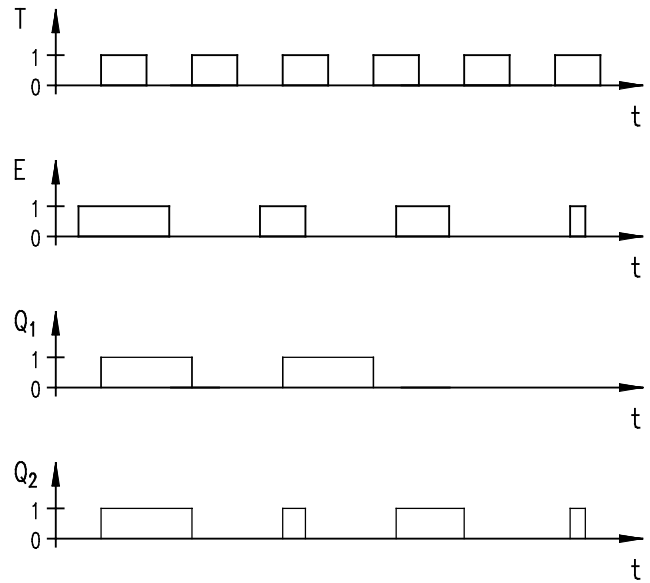
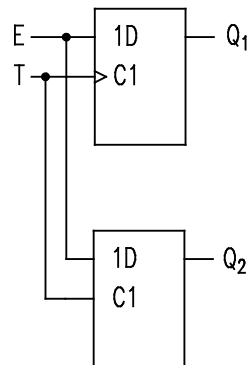


Aufgabe 3

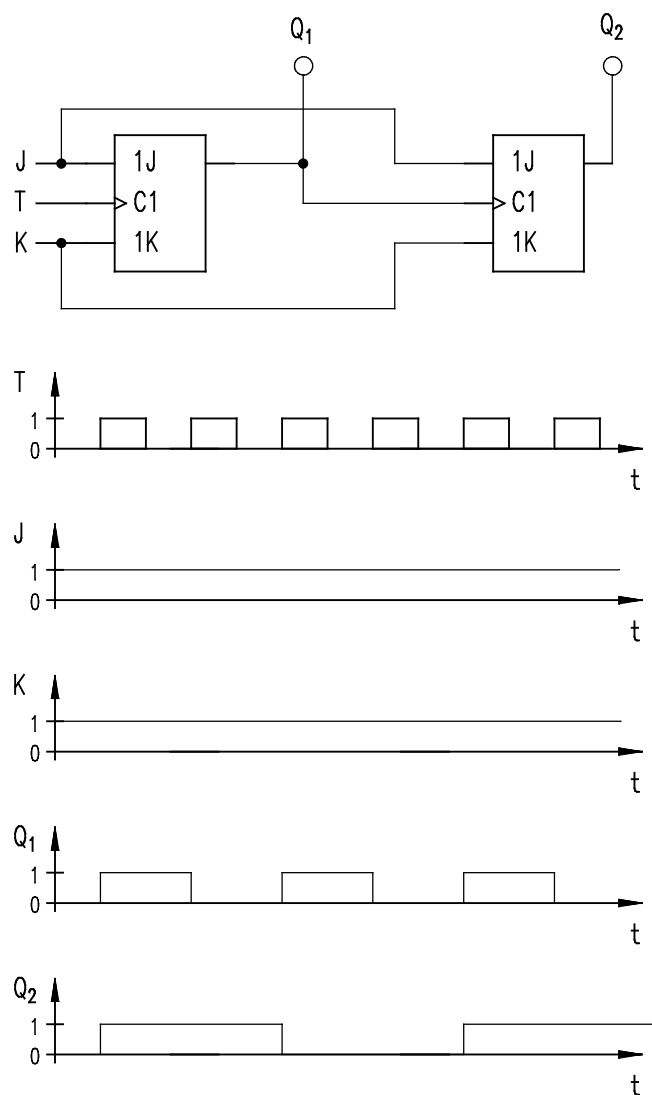


Funktion: T-Flip-Flop

Aufgabe 4



Aufgabe 5.1



Aufgabe 5.2

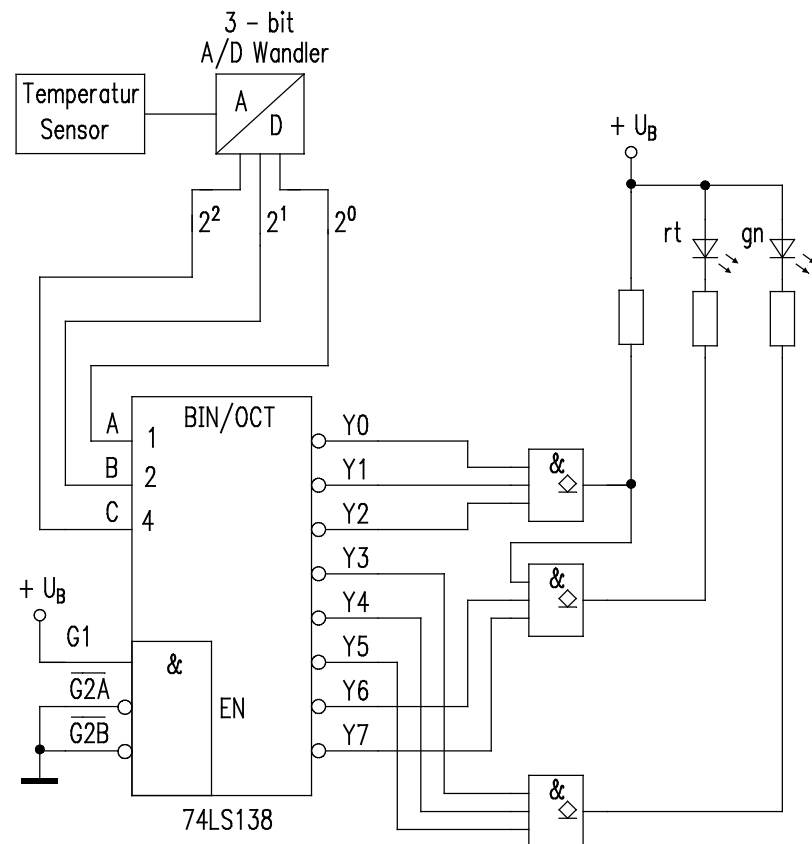
Asynchroner 2-Bit-Zähler

Aufgabe 6

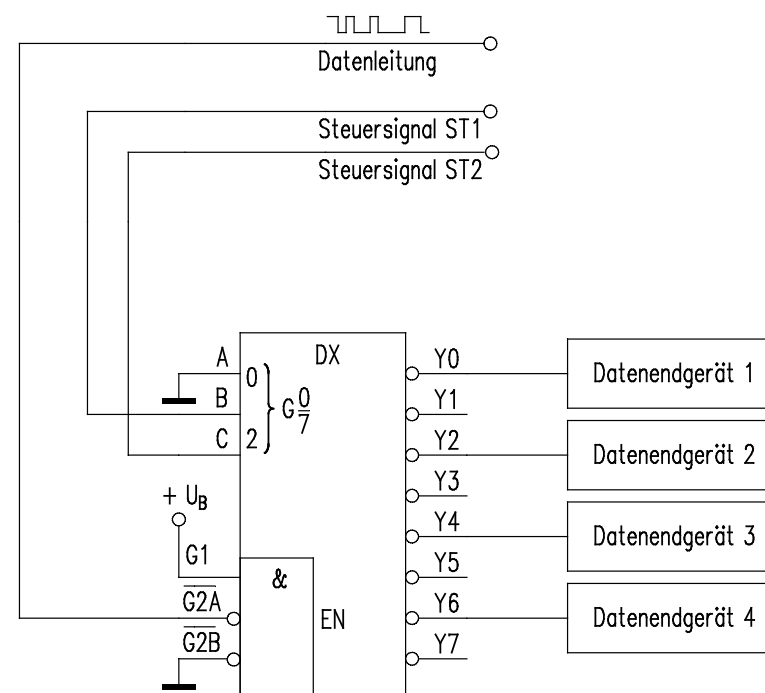
3-Bit-Schieberegister

3 Codier- und Auswahlschaltungen

Aufgabe 1



Aufgabe 2



4 D/A- und A/D-Wandler

Aufgabe 1

	Parallel-Wandler Flash-Wandler	Dual-Slope-Wandler
Gleichspannungsmessung mit großer Genauigkeit		X
Videosignale	X	

Aufgabe 2

$$f_A > 2 \cdot f_E \Rightarrow f_A > 20 \text{ kHz} \Rightarrow 21,5 \text{ kHz} \text{ und } 24 \text{ kHz}$$

Aufgabe 3

A/D-Wandlerverfahren	Anzahl der Schritte für die Wandlung
Zählverfahren	$2^N - 1 \Rightarrow 255$
Sukzessive Approximation	$N \Rightarrow 8$
Parallelverfahren	1