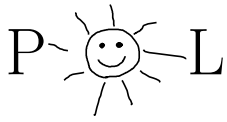
Rechnergestützter Schaltungsentwurf



Dr.-Ing. Peter Leibner

Tel.: +49 (0)3834 43 99 579
e-mail: pel@kedaj.org

Inhaltsverzeichnis

I.
Digitale Schaltungen

1 Logischer Schaltungsentwurf

1.1 Einleitung

Der Entwurf hochintegrierter Schaltungen ist ohne Rechnerunterstützung nicht mehr denkbar. Dies ist eine Folge der Komplexität der zu lösenden Aufgaben und der zu verwaltenden Datenmengen.

Von automatischen Entwurfswerkzeugen wird dabei erwartet, daß sie qualitativ hochwertige Lösungen zu vertretbaren Kosten erzeugen.

Immer mehr Entwurfsaufgaben werden als Semicustombausteine realisiert. Da diese Methode schnell und größtenteils automatisiert ist, werden damit hauptsächlich anwendungsspezifische Schaltungen, sogenannte **ASIC**s (Application-Specific Integrated Circuits), hergestellt.

Ebene	Beispiele für Beschreibungselemente
Verhaltensebene	Schaltfunktionen, Übergangsgraphen und -tabellen
Register-Transfer-Ebene	Speicher, ALUs, Kontrollblöcke
Gatterebene	Gatter, Flip-Flops, Zähler
Schalterebene	Schalter
Schaltkreisebene	R, L, C, Dioden, Transistoren
Device-Ebene	pn-Übergänge, MOS-Strukturen
Prozeß-Ebene	Diffusionszeiten, -temperaturen, Belegungsdichten

Tab. 1.1: Beschreibungsebenen

Die Entwurfsebene (→ Tab.1.1) ist zur Zeit noch vorwiegend die **Gatterebene**. In dieser führt der Schaltungsentwickler den funktionellen Entwurf mit Hilfe von vordefinierten bzw. anpaßbaren Funktionsbausteinen durch. Diese zellenbasierten Schaltungskonzepte bieten dem Entwickler die Möglichkeit, auf bereits ausgetestete Bausteine zurückzugreifen und damit den Entwurf schneller und sicherer zu machen.

Während beim **Full-Custom Design** alle Schritte vom Kunden bestimmt werden können (z.B., wie eine Leitung verlegt werden soll oder wie ein Transistor zu dimensionieren ist), so schränken beim **Semicustom Design** die vordefinierten Zellen bzw. Transistoren die Entscheidungsfreiheit des Kunden ein (*Semi*-Custom). Gleichzeitig werden durch diese Reduktion der Freiheitsgrade die zellenbasierten Konzepte einer automatischen Layoutsynthese wesentlich besser zugänglich.

Der Entwurf der Schaltungen erfolgt oft iterativ. Dabei wiederholen sich Entwurfs- und Verifikationsschritte so lange, bis die gewünschte Schaltfunktion realisiert ist. Diese

pragmatische Vorgehensweise wird in letzter Zeit immer mehr durch eine systematische ersetzt. Dabei ist man bestrebt, den Abstraktionsgrad zu erhöhen und damit die mathematische Beschreibbarkeit der Aufgabe zu verbessern. Dieser Weg führt von der Gatterebene zunächst zur **Register-Transfer**- und anschließend zur **Verhaltensebene** (behavioral level). Auf der Register-Transfer-Ebene wird die Schaltung z.B. durch Speicher, arithmetische Einheiten und Kontrollblöcke, beschrieben. Auf der Verhaltensebene geschieht dies durch Schaltfunktionen bzw. Übergangsgraphen und -tabellen.

Programme, die eine Schaltungsbeschreibung auf Verhaltensebene gestatten, sind z.B. Verilog der Firma CADENCE bzw. VHDL (Very High Speed Integrated Circuits Hardware Description Language) des DOD (U.S. Department of Defense). VHDL wurde bereits in den 80er Jahren entwickelt und inzwischen als IEEE (Institute of Electrical and Electronics Engineers)-Standard 1076 genormt.

Da für die Schaltungsentwicklung wieder zunehmend mathematische Verfahren gefragt sind, wird im folgenden zunächst die Boolesche Algebra aus ihrer Verwandschaft mit der Mengenalgebra abgeleitet und das Rechnen mit Schaltfunktionen geübt.

Nachfolgende Kapitel befassen sich dann mit der Verifikation einer Schaltung durch die Simulation ihres logischen Verhaltens und mit der Erzeugung von Testmustern zur Überprüfung auf Herstellungsfehler.

1.2 Stufen des Schaltungsentwurfs

Der elektrische Schaltungsentwurf beginnt mit der Spezifikation der gewünschten Schaltungsfunktion. Verknüpft die Schaltung binäre Variable mittels logischer Operationen zu logischen Funktionen, so wird sie digital genannt.

Digitale Schaltungen werden mit Hilfe einer zweiwertigen Schaltalgebra berechnet. Die Grundlagen dieser Algebra wurden von GEORGE BOOLE (1854) gelegt. Er systematisierte die Aussagenlogik und schuf damit ein algebraisches System zur Behandlung von logischen Aussagen.

Auf der Aussagen- und Prädikatenlogik basieren die heutigen Verfahren der „künstlichen Inteligenz“ (Artificial Intelligence), die ihre verstärkte Anwendung in Expertensystemen finden.

HUNTINGTON formalisierte (1904) die Boolesche Algebra weiter und bot damit CLAUDE SHANNON die Grundlage zur Definition der Schaltalgebra (1938).

Nach dem **logischen Schaltungsentwurf**, der darin besteht, die Spezifikation der Schaltung in eine formale Schaltungsbeschreibung umzusetzen, wird die Funktion der Schaltung mittels **Logiksimulation** verifiziert. In der Logiksimulation wird anstelle des realen physikalischen Elements (Gatter, Flip-Flop, Speicher, Leitung, usw.) ein Modell mit gleichem zeitlichen und logischen Verhalten verwendet. Durch die geeignete Modellierung aller Schaltungselemente kann somit das reale Verhalten der Schaltung durch einen Rechner sehr genau nachgebildet werden.

Durch die Integration digitaler Schaltungen wird der direkte Zugriff auf die einzelnen Schaltungselemente eingeschränkt. Der Zugang ist nur noch über die externen Schaltungsanschlüsse (Pins) möglich. Da dennoch jeder Chip, um Herstellungsfehler auszuschließen, getestet werden muß, ist die Testvorbereitung und -durchführung zu einem wichtigen Aspekt der Chip-Entwicklung geworden.

Um den Aufwand für die **Testmustergenerierung** in wirtschaftlich vertretbaren Grenzen zu halten, werden testfreundliche Entwurfsmethoden (DFT, Design for Testability), wie z.B. das Scan-Path-Prinzip, eingesetzt.

Beim Entwurf von kundenspezifischen Bausteinen wird die Plazierung und Verdrahtung der Schaltungselemente automatisch vorgenommen. Es wird dabei minimaler Flächenbedarf angestrebt. Durch die Minimierung der Fläche bei der **Layoutsynthese** können die Kosten für den Chip gesenkt (weniger Silizium, kleinere Gehäuse) und die Schaltgeschwindigkeit erhöht (kürzere Signalwege) werden.

1.3 Mathematische Grundlagen

Die Boolesche Algebra hängt eng mit der Mengenalgebra zusammen. Alle Rechenregeln der Mengenalgebra gelten ebenso für die Boolesche Algebra.

Operationen auf Mengen lassen sich symbolisch durch Mengendiagramme (Venn-Diagramm) veranschaulichen. Mit diesen kann die Richtigkeit der Rechenregeln leichter überprüft werden, als dies in der abstrakteren Schaltalgebra möglich ist.

Im nachfolgenden Abschnitt werden daher zunächst der Begriff Menge und die Rechenregeln auf Mengen definiert. Auf Mengen wird dabei nur soweit eingegangen, als dies für das Verständnis der Schaltalgebra notwendig ist.

Anschließend wird über die Isomorphie der Bezug zur Booleschen Algebra und der Schaltalgebra hergestellt.

1.3.1 Aussagen, Prädikate und Mengen

Eine Eigenschaft (**Prädikat**) eines Objektes x findet ihren sprachlichen Ausdruck durch eine **Aussage** über x (z.B., „x ist eine natürliche Zahl“) und wird mit $\mathcal{P}(x)$ bezeichnet. $\mathcal{P}(x)$ ist entweder wahr (z.B. für $x = 3$) oder falsch (z.B. für $x = 1.25$).

Faßt man alle Objekte, für die ein Prädikat $\mathcal{P}_M(x)$ eine wahre Aussage ist, zusammen, so erhält man eine **Menge** M, die Menge aller x, für die $\mathcal{P}_M(x)$ gilt:

$$\boxed{M = \{x \,|\, \mathcal{P}_M(x)\}} \tag{1.1}$$

Die Aussage, für die $\mathcal{P}_M(x)$ wahr ist, ist mit der Aussage „ x ist ein Element von M“ ($x \in M$) gleichbedeutend:

$$x \in M \Leftrightarrow \mathcal{P}_M(x) \qquad (\Leftrightarrow \text{ bedeutet logisch äquivalent})$$

Einige spezielle Mengen sind z.B.:

$$\begin{aligned}
\mathbb{N} &= \{x \,|\, x \text{ ist natürliche Zahl}\} = \{1, 2, 3, \ldots\} \\
\mathbb{N}_0 &= \{x \,|\, x \text{ ist natürliche Zahl einschließlich } 0\} = \{0, 1, 2, 3, \ldots\} \\
\mathbb{Z} &= \{x \,|\, x \text{ ist ganze Zahl}\} = \{\ldots, -2, -1, 0, 1, 2, \ldots\} \\
\mathbb{Q} &= \{x \,|\, x \text{ ist rationale Zahl}\} = \{x \,|\, x = \tfrac{p}{q},\ p \in \mathbb{Z} \wedge q \in \mathbb{N}\} \\
\mathbb{R} &= \{x \,|\, x \text{ ist reelle Zahl}\} \\
\mathbb{C} &= \{x \,|\, x \text{ ist komplexe Zahl}\}
\end{aligned}$$

Ist $\mathcal{P}_M(x)$ immer falsch, so ist M eine **leere Menge**. Sie wird mit $\emptyset$ bezeichnet. Wählt man aus M einen Teil der Elemente aus, so erhält man eine **Teilmenge** N von M,

$$(N \subset M) \Leftrightarrow (x \in N \Rightarrow x \in M) \Leftrightarrow (\mathcal{P}_N(x) \Rightarrow \mathcal{P}_M(x)).$$

Das Zeichen $\subset$ soll hier so aufgefaßt werden, daß auch

$$M \subset M \quad \text{und} \quad \emptyset \subset M \quad \text{gilt.}$$

Für den Fall, daß zwei Mengen M und N gleich sind, folgt somit

$$(M = N) \Leftrightarrow (M \subset N \wedge N \subset M).$$

Durch $\Rightarrow$ wird eine **Implikation** angegeben. Die Implikation gilt nur dann, wenn für alle x die **Subjunktion** ($\rightarrow$) wahr ist ($\rightarrow$ Tab.1.2, erste, zweite und vierte Zeile. Die dritte Zeile ist keine Implikation). Wird für „alle x“ der **Allquantor** gesetzt, so folgt

$$\forall_x[\mathcal{P}_N(x) \rightarrow \mathcal{P}_M(x)] \Leftrightarrow \text{w}.$$

In der gleichen Schreibweise wird die **Äquivalenz** durch

$$\forall_x[\mathcal{P}_N(x) \leftrightarrow \mathcal{P}_M(x)] \Leftrightarrow \text{w}$$

definiert. Zwei Aussagen sind folglich äquivalent, wenn für alle x die **Bijunktion** ($\leftrightarrow$) wahr ist ($\rightarrow$ Tab.1.2, erste und vierte Zeile).

Weitere Symbole in Tabelle 1.2 sind $\wedge$ (**Konjunktion** bzw. **und**-Verknüpfung) und $\vee$ (**Disjunktion** bzw. **oder**-Verknüpfung). Für eine wahre Aussage steht „w“, für eine falsche „f“. Durch die Negation einer Aussage ($\neg\mathcal{P}(x)$) wird ihr Wahrheitswert umgekehrt: $\neg\text{f} = \text{w}$ bzw. $\neg\text{w} = \text{f}$.

(Der Allquantor wird in Tabelle 1.2 und nachfolgend zur Vereinfachung der Schreibweise weggelassen.)

$\mathcal{P}_N(x)$	$\mathcal{P}_M(x)$	$\mathcal{P}_N(x) \wedge \mathcal{P}_M(x)$	$\mathcal{P}_N(x) \vee \mathcal{P}_M(x)$	$\mathcal{P}_N(x) \rightarrow \mathcal{P}_M(x)$	$\mathcal{P}_N(x) \leftrightarrow \mathcal{P}_M(x)$
f	f	f	f	w	w
f	w	f	w	w	f
w	f	f	w	f	f
w	w	w	w	w	w

Tab. 1.2: Wahrheitstabelle logischer Aussagen

Die Wahrheitstafeln für die Konjunktion, Disjunktion und Bijunktion sind unmittelbar verständlich und benötigen daher keiner zusätzlichen Erläuterung. Dies trifft für die Subjunktion nicht zu. Für sie gilt zeilenweise:

1.Zeile: Wenn es kein x mit den Eigenschaften $\mathcal{P}_N(x)$ und $\mathcal{P}_M(x)$ gibt, so folgt $N = M = \emptyset$. Die Implikation ist dann wegen $\emptyset \subset \emptyset$ immer erfüllt.

2.Zeile: Abbildung 1.1 zeigt, daß $N \subset M$ ist, da es ein x_1 mit $x_1 \notin N$ jedoch $x_1 \in M$ gibt.

3.Zeile: Es gibt ein $x_1 \in N$ jedoch $x_1 \notin M$. Dies ist ein Widerspruch zur Annahme, daß $N \subset M$ ist. Die Subjunktion ist daher nicht erfüllt, die Implikation existiert für diesen Fall nicht.

4.Zeile: Hier gilt $M = N$ ($\rightarrow$ Abb.1.2).

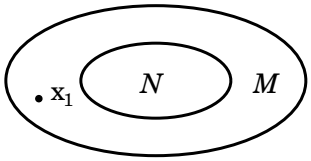


Abb. 1.1: $N \subset M$

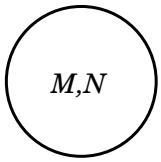
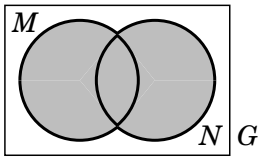


Abb. 1.2: $N = M$

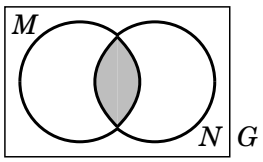
Zwei Mengen können zu einer neuen Menge verknüpft werden. Die Grundverknüpfungen zweier Mengen sind nachfolgend gegeben. Die Veranschaulichung dieser Verknüpfungen erfolgt durch **Venn-Diagramme**.

a) **Vereinigung** $M \cup N$ von M und N:



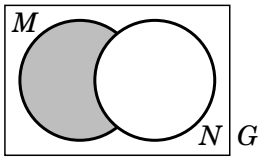
$$\boxed{M \cup N = \{x \,|\, x \in M \vee x \in N\}}$$

b) **Durchschnitt** $M \cap N$ von M und N:



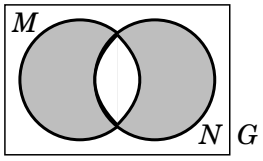
$$\boxed{M \cap N = \{x \,|\, x \in M \wedge x \in N\}}$$

c) **Differenz** $M \setminus N$ von M und N:



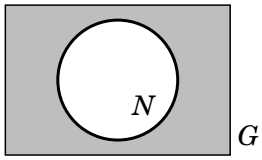
$$\boxed{M \setminus N = \{x \,|\, x \in M \wedge x \notin N\}}$$

d) **Symmetrische Differenz**:



$$\boxed{M \triangle N = (M \setminus N) \cup (N \setminus M)}$$

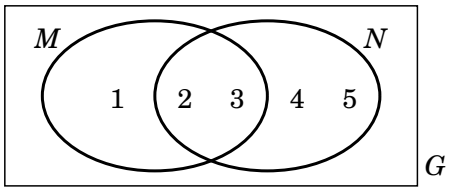
e) Ist N Teilmenge einer fest gegebenen Grundmenge G, $N \subset G$, so heißt $G \setminus N$ das **Komplement** $\bar{N}$ von N (bezüglich G):



$$\boxed{G \setminus N = \bar{N}}$$

Beispiel 1.1 *Gegeben sind die Mengen* $M = \{1,2,3\}$, $N = \{2,3,4,5\}$ *und* $G = \{1,2,3,4,5\}$. *Daraus lassen sich durch Verknüpfungen folgende neue Mengen bilden:*

$$\begin{aligned} M \cup N &= \{1,2,3,4,5\}, \\ M \setminus N &= \{1\}, \\ M \cap N &= \{2,3\}, \\ M \triangle N &= \{1,4,5\} \end{aligned}$$



Für das Komplement von N *bezüglich* G *folgt:*

$$G \setminus N = \{1,2,3,4,5\} \setminus \{2,3,4,5\} = \{1\} = \overline{\{2,3,4,5\}} = \bar{N}$$

Ist der Durchschnitt von M und N leer ($M \cap N = \emptyset$), so werden M und N **disjunkt** (durchschnittsfremd) genannt.

Für Mehrfachverknüpfungen von Mengen gelten zahlreiche Rechenregeln. Wichtige Grundregeln sind:

(a)	$M \cup N = N \cup M$	($\cup$ ist kommutativ)
(b)	$M \cup (N \cup P) = (M \cup N) \cup P$	($\cup$ ist assoziativ)
(c)	$M \cup M = M$	($\cup$ ist idempotent)
(d)	$M \cup (N \cap P) = (M \cup N) \cap (M \cup P)$	($\cup$ ist distributiv bezüglich $\cap$)
(e)	$M \cup (M \cap N) = M$	($\cup$ ist adjunktiv bezüglich $\cap$)

(1.2)

Wegen der Assoziativität können in (1.2b) die Klammern wegfallen. Außerdem kann in (1.2a-c) $\cup$ durch $\cap$ ersetzt werden und in (1.2d,e) $\cup$ und $\cap$ ihre Plätze tauschen. Die Gültigkeit der Regeln (1.2) folgt aus der jeweiligen Definition, bzw. aus der Konstruktion entsprechender Venn-Diagramme.

Für $M, N \subset G$ gelten zusätzlich die Regeln von **de Morgan**

$$\overline{M \cup N} = \bar{M} \cap \bar{N} \quad \text{und} \quad \overline{M \cap N} = \bar{M} \cup \bar{N}, \tag{1.3}$$

sowie

$$M \setminus N = M \cap \bar{N}. \tag{1.4}$$

Mit Hilfe von Venn-Diagrammen kann man sich leicht klarmachen, daß

$$M \subset M \cup N \quad \text{und} \quad M \cap N \subset M \tag{1.5}$$

gilt. Außerdem ist für $N \subset M$

$$M \cup N = M \quad \text{und} \quad M \cap N = N. \tag{1.6}$$

Für $M \subset G$ folgt

$$M \cup \bar{M} = G,\ M \cap \bar{M} = \emptyset \quad \text{und} \quad M \cup \emptyset = M,\ M \cap G = M. \tag{1.7}$$

Beispiel 1.2 *Es sei* $G = \{1,2,3,4,5\}$, $M = \{1,2,3\}$ *und* $N = \{3,4\}$. *Dann gilt mit* (1.3)

$$\overline{M \cup N} = \overline{\{1,2,3,4\}} = \{5\},$$
$$\bar{M} \cap \bar{N} = \{4,5\} \cap \{1,2,5\} = \{5\},$$

und mit (1.4)

$$M \setminus N = \{1,2\},$$
$$M \cap \bar{N} = \{1,2,3\} \cap \{1,2,5\} = \{1,2\}.$$

1.3.1.1 Mengensysteme

Mengen, deren Elemente selbst wieder Mengen sind, werden **Mengensysteme** genannt. In den wichtigsten Fällen sind alle Elemente eines Mengensystems (gekennzeichnet durch Unterstreichung, z.B. $\underline{M}$) Teilmengen einer gegebenen Menge M, z.B. (endliches Mengensystem):

$$\underline{M} = \{M_1, M_2, \ldots, M_n\}, \quad M_i \subset M, \quad i = 1,2,\ldots,n$$

Faßt man alle Teilmengen einer Menge M zu einem Mengensystem zusammen, so wird dieses System **Potenzmenge** der Menge M genannt.

$$\underline{P}(M) = \{X \,|\, X \subset M\}$$

Beispiel 1.3 *Zu* $M = \{1,2,3\}$ *gehört die Potenzmenge*

$$\underline{P}(M) = \{\emptyset, \{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}, M\}.$$

Die Potenzmenge $\underline{P}(M)$ enthält 2^n Mengen, wenn M eine endliche Menge mit n Elementen ist. Dies kann mit Hilfe der Kombinatorik nachgewiesen werden.

Bekanntlich gibt es $\binom{n}{k}$ Möglichkeiten, aus einer Menge mit n Elementen Teilmengen mit k Elementen zu bilden. Die Potenzmenge besitzt somit

$$\binom{n}{0} \text{ leere}, \binom{n}{1} \text{ einelementige}, \binom{n}{2} \text{ zweielementige}, \cdots, \binom{n}{n} \text{ n-elementige}$$

Mengen, insgesamt also

$$m = \sum_{k=0}^{n} \binom{n}{k} \quad \text{Elemente.}$$

Mit Hilfe der binomischen Formel,

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^k b^{n-k} \quad \text{und} \quad a = b = 1 \quad \text{folgt} \quad m = 2^n.$$

1.3.2 Algebraische Strukturen

Eine algebraische Struktur wird definiert durch eine Menge A, **Trägermenge** genannt und eine oder mehrere Operationen (z.B. $*, \circ, \oplus, +, \cdot$) auf dieser Menge.

Dabei muß für

$$x_1 \in A \quad \text{und} \quad x_2 \in A \quad \text{z.B.} \quad x_1 \oplus x_2 = x_3 \in A \quad \text{erfüllt sein.}$$

Jede algebraische Operation auf A, wie z.B. $\oplus$, ist eine Abbildung $\varphi = \oplus$ aus $A \times A$ nach A:

$$\oplus : A \times A \to A \quad \text{bzw.} \quad \varphi : A \times A \to A$$

So gilt z.B.

$$\oplus(x_1, x_2) = \varphi(x_1, x_2) = x_1 \oplus x_2.$$

Beispiel 1.4 *Aus zwei Mengen M und N wird eine neue Menge gebildet, wenn jedes Element $x \in M$ mit jedem $y \in N$ zu einem* **geordneten Paar** (x, y) *kombiniert wird. Die Menge aller möglicher Paare heißt* **kartesisches Produkt** *von M und N:*

$$M \times N = \{(x, y) \,|\, x \in M \wedge y \in N\}$$

Für $M = \{1, 2\}$ und $N = \{2, 3, 4\}$ gilt dann z.B.

$$M \times N = \{(1, 2), (1, 3), (1, 4), (2, 2), (2, 3), (2, 4)\}.$$

Das kartesische Produkt kann leicht auf mehrere Faktoren erweitert werden:

$$M_1 \times M_2 \times \cdots M_n = \{(x_1, x_2, \ldots, x_n) \,|\, x_1 \in M_1 \wedge \ldots \wedge x_n \in M_n\}$$

Die Elemente $(x_1, x_2, \ldots, x_n)$ werden **geordnete n-Tupel** *genannt.*

Wird zwei Elementen einer Menge A ein weiteres Element x_3 durch $x_1 \oplus x_2$ zugeordnet, so spricht man von einer zweistelligen Operation.

Eine einstellige Operation ist z.B. die Negation einer reellen Zahl:

$$f(x) = -x; \quad x \in \mathbb{R}$$

Als nullstellige Operation wird z.B. die Auswahl einer bestimmten Zahl $x \in \mathbb{R}$ (z.B. $x = 1$) oder einer Teilmenge von $\mathbb{R}$ (z.B. „x ist eine gerade Zahl“) bezeichnet.

Sind n Operationen $\varphi_1, \ldots, \varphi_n$ auf einer Menge A mit den jeweiligen Stellenzahlen $k_1, \ldots, k_n$ definiert, so spricht man von einer **universellen algebraischen Struktur** (Algebra) des **Typs** $(k_1, \ldots, k_n)$ und bezeichnet sie mit dem Struktursymbol

$$(A, \varphi_1, \varphi_2, \ldots, \varphi_n).$$

Eine universelle algebraische Struktur vom Typ (2,2,1,0,0) ist z.B.

$$\boxed{(\underline{P}(M), \cup, \cap, \bar{}, \emptyset, M)}. \tag{1.8}$$

M ist dabei eine beliebige (endliche) Menge. Auf der Trägermenge $\underline{P}(M)$ sind fünf Operationen $\varphi_1, \ldots, \varphi_5$ definiert:

$$\begin{array}{llll} \varphi_1 = \cup & : & \text{Vereinigung,} & k_1 = 2; \\ \varphi_2 = \cap & : & \text{Durchschnitt,} & k_2 = 2; \\ \varphi_3 = \bar{} & : & \text{Komplementbildung,} & k_3 = 1; \\ \varphi_4 = \emptyset & : & \text{Auswahl von } \emptyset \subset M, & k_4 = 0; \\ \varphi_5 = M & : & \text{Auswahl von } M \subset M, & k_5 = 0; \end{array}$$

Die so definierte Algebra heißt (endliche) Mengenalgebra.

Sind zwei Algebren vom gleichen Typ, so werden sie **isomorph** (gleichgestaltig) genannt.

1.3.2.1 Boolesche Algebra

Eine Boolesche Algebra ist vom gleichen Typ wie die Mengenalgebra. Sie wird mit

$$\boxed{(\mathbb{B}, \vee, \wedge, \bar{}, 0, 1)} \tag{1.9}$$

angegeben. Die Operationen $\vee, \wedge, \bar{}, 0, 1$ haben folgende Eigenschaften:

a) Die zweistelligen Operationen $\vee$ und $\wedge$ (Disjunktion und Konjunktion) sind assoziativ und kommutativ. Außerdem ist jede der Operationen distributiv und adjunktiv bezüglich der anderen ($\rightarrow$ (1.2)).

b) Für die einstellige Operation $\bar{}$ (Negation) gilt für alle $x \in \mathbb{B}$

$$x \vee \bar{x} = 1 \quad \text{und} \quad x \wedge \bar{x} = 0 \quad (\rightarrow (1.7)).$$

c) Durch die beiden nullstelligen Operationen werden die neutralen Elemente festgelegt. Für das neutrale Element der Disjunktion wird 0, für das der Konjunktion 1 gewählt. Es gilt also

$$x \vee 0 = x \quad \text{und} \quad x \wedge 1 = x \quad (\rightarrow (1.7)).$$

Die Mengenalgebra $(\underline{P}(M), \cup, \cap, \bar{}, \emptyset, M)$ ist somit eine Boolesche Algebra. Man kann leicht nachprüfen, daß sie alle oben notierten Eigenschaften hat, wenn man $\mathbb{B}$ mit $\underline{P}(M)$, $\vee$ mit $\cup$, $\wedge$ mit $\cap$ usw. identifiziert. Beide Algebren sind isomorph.

Wegen der Isomorphie zur Mengenalgebra kann es keine endliche Boolesche Algebra mit einer beliebigen Anzahl von Trägerelementen $x \in \mathbb{B}$ geben, da die Anzahl der Elemente beider Trägermengen gleich sein muß. Für die Potenzmenge einer Menge M mit n Elementen ist sie 2^n. Demnach kann die Menge $\mathbb{B}$ nur $2, 4, 8, \ldots$ Elemente enthalten, nie jedoch $3, 5, \ldots$.

Beispiel 1.5 *Es ist eine Struktur* $(\mathbb{B}, \vee, \wedge, \bar{}, 0, 1)$ *mit* $\mathbb{B} = \{0, a, b, 1\}$ *gegeben. Da* $\mathbb{B}$ *vier Elemente hat, kann nur die Potenzmenge einer zweielementigen Menge* $M = \{\alpha, \beta\}$ *Träger der isomorphen Mengenalgebra* $(\underline{\mathcal{P}}(M), \cup, \cap, \bar{}, \emptyset, M)$ *sein. Für die Mengenalgebra ergeben sich die Operationstafeln in* Tabelle 1.3.

$\cup$	$\emptyset$	$\{\alpha\}$	$\{\beta\}$	M
$\emptyset$	$\emptyset$	$\{\alpha\}$	$\{\beta\}$	M
$\{\alpha\}$	$\{\alpha\}$	$\{\alpha\}$	M	M
$\{\beta\}$	$\{\beta\}$	M	$\{\beta\}$	M
M	M	M	M	M

$\cap$	$\emptyset$	$\{\alpha\}$	$\{\beta\}$	M
$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
$\{\alpha\}$	$\emptyset$	$\{\alpha\}$	$\emptyset$	$\{\alpha\}$
$\{\beta\}$	$\emptyset$	$\emptyset$	$\{\beta\}$	$\{\beta\}$
M	$\emptyset$	$\{\alpha\}$	$\{\beta\}$	M

$^-$	$\emptyset$	$\{\alpha\}$	$\{\beta\}$	M
	M	$\{\beta\}$	$\{\alpha\}$	$\emptyset$

Tab. 1.3: Operationstafeln der Mengenalgebra

Zur Bestimmung der Tafeln der Booleschen Algebra ($\rightarrow$ Tab.1.4) *wird die Isomorphie ausgenutzt. Es muß dann lediglich* $\emptyset$ *durch* 0, $\{\alpha\}$ *durch* a, $\{\beta\}$ *durch* b *sowie* $\cup$ *durch* $\vee$, $\cap$ *durch* $\wedge$, *das Komplement durch die Negation und* M *durch* 1 *ersetzt werden.*

$\vee$	0	a	b	1
0	0	a	b	1
a	a	a	1	1
b	b	1	b	1
1	1	1	1	1

$\wedge$	0	a	b	1
0	0	0	0	0
a	0	a	0	a
b	0	0	b	b
1	0	a	b	1

$^-$	0	a	b	1
	1	b	a	0

Tab. 1.4: Operationstafeln der Booleschen Algebra

1.3.2.2 Schaltalgebra

Die wichtigste Boolesche Algebra ist die Schaltalgebra. Ihre Trägermenge enthält nur die Elemente 0 und 1.

$$(\mathbb{B}, \vee, \wedge, ^-, 0, 1) = (\{0,1\}, \vee, \wedge, ^-, 0, 1)$$

Die Schaltalgebra ist dann zu einer Mengenalgebra mit einer einelementigen Menge $M = \{m\}$ und der zweielementigen Potenzmenge $\underline{\mathcal{P}}(M) = \{\emptyset, M\}$ isomorph. Die Operationstafeln der Schaltalgebra und der zugehörigen Mengenalgebra sind in den Tabellen 1.6 und 1.5 gegeben.

$\cup$	$\emptyset$	M
$\emptyset$	$\emptyset$	M
M	M	M

$\cap$	$\emptyset$	M
$\emptyset$	$\emptyset$	$\emptyset$
M	$\emptyset$	M

$^-$	$\emptyset$	M
	M	$\emptyset$

Tab. 1.5: Operationstafeln der Mengenalgebra für $M = \{m\}$

$\vee$	0	1
0	0	1
1	1	1

$\wedge$	0	1
0	0	0
1	0	1

$^-$	0	1
	1	0

Tab. 1.6: Operationstafeln der Schaltalgebra

Die wichtigsten Rechenregeln (1.10) der Schaltalgebra folgen unmittelbar aus (1.2).

(a)	$(x \vee y) \vee z = x \vee (y \vee z);$	$(xy)z = x(yz)$
(b)	$x \vee y = y \vee x;$	$xy = yx$
(c)	$x \vee xy = x;$	$x(x \vee y) = x$
(d)	$x \vee yz = (x \vee y)(x \vee z);$	$x(y \vee z) = xy \vee xz$
(e)	$x \vee \bar{x} = 1;$	$x\bar{x} = 0$
(f)	$x \vee 0 = x;$	$x \wedge 1 = x$
(g)	$\bar{0} = 1;$	$\bar{1} = 0$
(h)	$\overline{x \vee y} = \bar{x}\bar{y};$	$\overline{xy} = \bar{x} \vee \bar{y}$
(i)	$x \vee x = x;$	$x \wedge x = x$
(j)	$x \vee 1 = 1;$	$x \wedge 0 = 0$
(k)	$\bar{\bar{x}} = x$	

(1.10)

Da die Konjunktion (→ Tab.1.6) der Multiplikation natürlicher Zahlen formal gleich ist, kann das Symbol $\wedge$ weggelassen werden. Es wird dann anstatt $x_3 \wedge x_2 \wedge x_1$ verkürzt $x_3x_2x_1$ geschrieben.

Genauso könnte man $\vee$ durch $+$ ersetzen. Da sich die Addition von der Disjunktion jedoch für $1 \vee 1 = 1$ unterscheidet, wird im folgenden $\vee$ beibehalten.

Werden in (1.10) die Operationen und neutralen Elemente ausgetauscht,

$$\wedge \leftrightarrows \vee \quad \text{und} \quad 0 \leftrightarrows 1,$$

so gehen die Regeln für die Disjunktion in die der Konjunktion über und umgekehrt. Jede gültige Beziehung hat folglich immer auch einen dualen Partner. Dieses Gesetz wird **Dualitätsprinzip** der Booleschen Algebra genannt.

1.3.3 Schaltfunktionen

Eine reelle Funktion f von n Variablen wird durch eine Abbildung von $\mathbb{R}^n$ nach $\mathbb{R}$ definiert:

$$f : \mathbb{R}^n \to \mathbb{R}, \quad f(\boldsymbol{x}) = y \quad \text{mit} \quad \boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T \in \mathbb{R}^n \quad \text{und} \quad y \in \mathbb{R}$$

Für eine n-stellige Schaltfunktion gilt analog

$$f : \mathbb{B}^n \to \mathbb{B}, \quad f(\boldsymbol{x}) = y \quad \text{mit} \quad \boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T \in \mathbb{B}^n \quad \text{und} \quad y \in \mathbb{B} = \{0, 1\}.$$

Eigenschaften einer Schaltfunktion:

a) Der Definitionsbereich $\mathbb{B}^n = \{0, 1\}^n$ umfaßt 2^n Werte, da jede der Variablen in $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T$ die Werte 0 und 1 annehmen kann.

b) Der Wertebereich wird durch die Menge $\mathbb{B} = \{0, 1\}$ gebildet und enthält 2 Elemente.

c) Die Menge aller n-elementigen Schaltfunktionen entspricht damit der Menge aller Abbildungen aus einer 2^n-elementigen Menge in eine 2-elementige. Dafür gibt es $2^{(2^n)}$ Möglichkeiten (→ Tab.1.7, für $n = 2$).

d) Die Schaltfunktionen sind selbst wieder Trägermenge einer Schaltalgebra. Für die Variablen x, y, z in (1.10) können folglich auch Schaltfunktionen eingesetzt werden.

	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
00	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
01	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
10	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
11	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○

Tab. 1.7: $2^{(2^2)} = 16$ mögliche Abbildungen einer 4- in eine 2-elementige Menge.

Beispiel 1.6 *Von einer Variablen x gibt es $2^{(2^1)} = 4$ Funktionen* ($\rightarrow$ Tab.1.8).

x	f_0	f_1	f_2	f_3
0	0	1	0	1
1	0	0	1	1

Tab. 1.8: Menge aller 1-stelligen Schaltfunktionen

Dies sind $f_0(x) = 0, \quad f_1(x) = \bar{x}, \quad f_2(x) = x, \quad$ *und* $\quad f_3(x) = 1.$

Beispiel 1.7 *Zwei Variable x_1, x_2 ergeben 16 mögliche Funktionen* ($\rightarrow$ Tab.1.9, vgl. auch Tab.1.7). *Die wichtigsten Verknüpfungen der Schaltalgebra,* **UND, ODER, EXOR** (Antivalenz)**, Negation, NAND** und **NOR***, sind in der Tabelle durch Fettdruck hervorgehoben. Die restlichen Beziehungen kommen vorwiegend in der Aussagenlogik vor. Sie werden hier nur der Vollständigkeit halber angeführt.*

x_1 x_2	1 1 0 0 1 0 1 0	i	$f_i(\boldsymbol{x}) = f_i(x_1, x_2)$	Bezeichnung
	0 0 0 0	0	0	Kontradiktion
	0 0 0 1	1	$\overline{x_1 \vee x_2}$	**NOR**
	0 0 1 0	2	$\bar{x}_1 x_2$	Inhibition
	0 0 1 1	3	$\bar{x}_1$	**Negation**
	0 1 0 0	4	$x_1 \bar{x}_2$	Inhibition
	0 1 0 1	5	$\bar{x}_2$	**Negation**
	0 1 1 0	6	$x_1 \oplus x_2$	**EXOR** (Antivalenz)
	0 1 1 1	7	$\overline{x_1 x_2}$	**NAND**
	1 0 0 0	8	$x_1 x_2$	**UND** (Konjunktion)
	1 0 0 1	9	$x_1 \leftrightarrow x_2$	Bijunktion
	1 0 1 0	10	x_2	Projektion
	1 0 1 1	11	$x_1 \rightarrow x_2$	Subjunktion
	1 1 0 0	12	x_1	Projektion
	1 1 0 1	13	$x_2 \rightarrow x_1$	Subjunktion
	1 1 1 0	14	$x_1 \vee x_2$	**ODER** (Disjunktion)
	1 1 1 1	15	1	Tautologie

Tab. 1.9: Menge aller 2-stelliger Schaltfunktionen

1.3.4 Darstellungsalternativen logischer Funktionen

1.3.4.1 Wertetabelle

Da der Definitionsbereich einer Schaltfunktion nur endlich viele Elemente hat, kann die Funktion durch eine endliche Tabelle vollständig beschrieben werden.

Die binären Werte

$$i_n i_{n-1} \dots i_2 i_1 \in \mathbb{B}^n, \quad i_j \in \mathbb{B} = \{0,1\} \quad \text{für} \quad j \in \{1,2,3,\dots,n\}$$

der 2^n Elemente des Definitionsbereichs numerieren dabei in ihrer dezimalen Schreibweise

$$i_n i_{n-1} \dots i_2 i_1 = \text{Bin}(i), \quad i \in \mathbb{N}_0$$

die Zeilen der Wertetabelle durch. Sie erhält damit folgendes Aussehen:

i	$x_n x_{n-1} \cdots x_2 x_1$	$y = f(x_1, x_2, \dots, x_n)$
0	$00 \cdots 00$	$f(0,0,\dots,0,0)$
1	$00 \cdots 01$	$f(1,0,\dots,0,0)$
2	$00 \cdots 10$	$f(0,1,\dots,0,0)$
3	$00 \cdots 11$	$f(1,1,\dots,0,0)$
$\vdots$	$\vdots$	$\vdots$
i	$i_n i_{n-1} \cdots i_2 i_1 = \text{Bin}(i)$	$f(i_1, i_2, \dots, i_{n-1}, i_n)$
$\vdots$	$\vdots$	$\vdots$
$2^n - 1$	$11 \cdots 11$	$f(1,1,\dots,1,1)$

Tab. 1.10: Wertetabelle einer logischen Funktion

Zur vollständigen Charakterisierung einer n-stelligen Schaltfunktion reicht es, die Teilmenge $I_f \subset I$ derjenigen Indizes anzugeben, für welche $f(i_1, i_2, \dots, i_{n-1}, i_n) = 1$ ist.

Beispiel 1.8 *Durch die in der rechten Spalte der* Wertetabelle 1.11 *enthaltenen Funktionswerte ist eine zweistellige Schaltfunktion* $f : \mathbb{B}^2 \to \mathbb{B}$, $y = f(x_1, x_2)$ *gegeben. Der Definitionsbereich ist durch die Indexmenge* $I = \{0,1,2,3\}$ *gegeben, die Funktion selbst wird durch die Teilmenge* $I_f = \{0,2\} \subset I$ *bestimmt.*

i	$x_2 x_1$	$f(x_1, x_2)$
0	00	1
1	01	0
2	10	1
3	11	0

Tab. 1.11: Wertetabelle

1.3.4.2 Algebraische Darstellung durch Normalpolynome

Unter einem Polynom wird die Verknüpfung der Variablen $x_1, x_2, \dots, x_n$ sowie der Konstanten 0 und 1 durch die Operationssymbole $\vee, \wedge$ und $^-$ verstanden, z.B.,

$$f(\boldsymbol{x}) = (x_4 x_3 \vee \bar{x}_1) x_1 \wedge 1 \vee 0 = (x_4 x_3 \vee \bar{x}_1) x_1 = x_4 x_3 x_1. \tag{1.11}$$

Das Polynom wird von $n = 4$ Variablen erzeugt. Es müssen dabei nicht alle Unbekannten vorkommen (x_2 fehlt zum Beispiel). Das von 4 Variablen erzeugte Polynom ist eine Schaltfunktion. Dies läßt sich durch das Aufstellen ihrer Wertetabelle ($\rightarrow$ Tab.1.12) leicht überprüfen.

i	$x_4x_3x_2x_1$	$f(\boldsymbol{x})$	i	$x_4x_3x_2x_1$	$f(\boldsymbol{x})$
0	0000	0	8	1000	0
1	0001	0	9	1001	0
2	0010	0	10	1010	0
3	0011	0	11	1011	0
4	0100	0	12	1100	0
5	0101	0	13	1101	1
6	0110	0	14	1110	0
7	0111	0	15	1111	1

Tab. 1.12: Wertetabelle des Polynoms $f(\boldsymbol{x}) = (x_4x_3 \vee \bar{x}_1)x_1 \wedge 1 \vee 0$

Jedem Polynom ist eine Schaltfunktion eindeutig zugeordnet. Dies gilt jedoch nicht für die Umkehrung, da jede Schaltfunktion auf verschiedene Art und Weise ($\rightarrow$ (1.11)) durch ein Polynom dargestellt werden kann.

Eine eindeutige Darstellung einer Schaltfunktion ist mit Hilfe spezieller Polynome, sogenannter **Minterme**, möglich.

Ein Minterm m_i wird wie folgt definiert:

$$m_i = m_i(x_1, x_2, \ldots, x_n) = x_n^{i_n} x_{n-1}^{i_{n-1}} \cdots x_1^{i_1} = \boldsymbol{x}_i \tag{1.12}$$

Dabei gilt wieder

$$i \in I = \{0, 1, 2, \ldots, 2^n - 1\} \quad \text{und} \quad i_n i_{n-1} \ldots i_1 = \text{Bin}(i).$$

Weiter wird vereinbart, daß für beliebige $j \in \{1, 2, \ldots, n\}$

$$x_j^0 = \bar{x}_j \quad \text{und} \quad x_j^1 = x_j \quad \text{gelten soll (positive Logik).}$$

Beispiel 1.9 *In der nebenstehenden Tabelle ($\rightarrow$ Tab.1.13) sind alle Minterme einer Funktion von zwei Variablen angeführt.*

i	x_2x_1	$x_2^{i_2}x_1^{i_1} = m_i$
0	00	$x_2^0x_1^0 = \bar{x}_2\bar{x}_1 = m_0$
1	01	$x_2^0x_1^1 = \bar{x}_2x_1 = m_1$
2	10	$x_2^1x_1^0 = x_2\bar{x}_1 = m_2$
3	11	$x_2^1x_1^1 = x_2x_1 = m_3$

Tab. 1.13: Minterme

Ein Minterm ist folglich ein Polynom, das aus der Konjunktion aller n Unbekannten gebildet wird, wobei diese auch negiert sein dürfen.

Beispiel 1.10 *Es gilt z.B.*

a) $m_5(x_1, x_2, x_3) = x_3^1x_2^0x_1^1 = x_3\bar{x}_2x_1$ *mit* $\text{Bin}(5) = 101$ *und*

b) $m_3(x_1, x_2, x_3, x_4) = x_4^0 x_3^0 x_2^1 x_1^1 = \bar{x}_4 \bar{x}_3 x_2 x_1$ *mit* $\mathrm{Bin}(3) = 0011$.

Ein Minterm nimmt nur für eine bestimmte Belegung der Variablen den Wert 1 an, für alle übrigen Belegungen hat er den Wert 0:

$$m_i(x_1, x_2, \ldots, x_n) = \begin{cases} 1 & \text{für} \quad i_n i_{n-1} \ldots i_2 i_1 \;=\; \mathrm{Bin}(i) \\ 0 & \text{sonst} \end{cases}$$

Mit Hilfe der Minterme kann nun die Schaltfunktion $f(\boldsymbol{x})$ eindeutig dargestellt werden. Diese Darstellung wird **Kanonische Disjunktive Normalform (KDNF)** genannt.

$$\boxed{f(x_1, \ldots, x_n) = \bigvee_{i \in I_f} m_i(x_1, \ldots, x_n)} \tag{1.13}$$

Die KDNF ist eine *disjunktive* Verknüpfung von Konjunktionstermen. Das Wort *kanonisch* wird hier im Sinne von eindeutig verwendet.

Beispiel 1.11 *Durch die* Tabelle 1.14 *ist eine Schaltfunktion gegeben. Es gilt* $I_f = \{1, 5, 7\}$. *Damit folgt für die KDNF:*

$$\begin{aligned} f(\boldsymbol{x}) &= \bigvee_{i \in \{1,5,7\}} m_i(x_1, x_2, x_3) \\ &= m_1 \vee m_5 \vee m_7 \\ &= x_3^0 x_2^0 x_1^1 \vee x_3^1 x_2^0 x_1^1 \vee x_3^1 x_2^1 x_1^1 \\ &= \bar{x}_3 \bar{x}_2 x_1 \vee x_3 \bar{x}_2 x_1 \vee x_3 x_2 x_1 \end{aligned}$$

i	$x_3 x_2 x_1$	f	$\bar{f}$
0	000	0	1
1	001	1	0
2	010	0	1
3	011	0	1
4	100	0	1
5	101	1	0
6	110	0	1
7	111	1	0

Tab. 1.14: Wertetabelle

Für zwei n-stellige Minterme gilt weiter

$$m_i \wedge m_j = \begin{cases} m_i & \text{falls} \quad i = j \\ 0 & \text{sonst,} \end{cases} \tag{1.14}$$

z.B. für $n = 3$,

$$m_1 \wedge m_5 = \bar{x}_3 \bar{x}_2 x_1 x_3 \bar{x}_2 x_1 = \underbrace{\bar{x}_3 x_3}_{0} \bar{x}_2 x_1 = 0$$

Eine weitere zweistufige Normalform zur Darstellung von Schaltfunktionen ist die **Kanonische Konjunktive Normalform (KKNF)**. Die KKNF ist eine *konjunktive* Verknüpfung von Disjunktionstermen. Sie folgt unmittelbar aus dem Dualitätsprinzip der Booleschen Algebra.

$$\boxed{f(x_1, \ldots, x_n) = \bigwedge_{i \in \bar{I}_f} \bar{m}_i(x_1, \ldots, x_n)} \tag{1.15}$$

Beispiel 1.12 *Für die Schaltfunktion aus dem vorherigen Beispiel wird die KKNF aufgestellt. Es ist* $\bar{I}_f = \{0, 2, 3, 4, 6\}$. *Damit folgt*

$$f(\boldsymbol{x}) \;=\; \bigwedge_{i \in \{0,2,3,4,6\}} \bar{m}_i(x_1, x_2, x_3)$$

$$
\begin{aligned}
&= \bar{m}_0 \wedge \bar{m}_2 \wedge \bar{m}_3 \wedge \bar{m}_4 \wedge \bar{m}_6 \\
&= \overline{x_3^0 x_2^0 x_1^0} \wedge \overline{x_3^0 x_2^1 x_1^0} \wedge \overline{x_3^0 x_2^1 x_1^1} \wedge \overline{x_3^1 x_2^0 x_1^0} \wedge \overline{x_3^1 x_2^1 x_1^0} \\
&= \overline{\bar{x}_3 \bar{x}_2 \bar{x}_1} \wedge \overline{\bar{x}_3 x_2 \bar{x}_1} \wedge \overline{\bar{x}_3 x_2 x_1} \wedge \overline{x_3 \bar{x}_2 \bar{x}_1} \wedge \overline{x_3 x_2 \bar{x}_1} \\
&= (x_3 \vee x_2 \vee x_1)(x_3 \vee \bar{x}_2 \vee x_1)(x_3 \vee \bar{x}_2 \vee \bar{x}_1)(\bar{x}_3 \vee x_2 \vee x_1)(\bar{x}_3 \vee \bar{x}_2 \vee x_1)
\end{aligned}
$$

Das gleiche Ergebnis erhält man durch das Aufstellen der KDNF von $\bar{f}$ (*siehe* Tabelle 1.14) *und anschließendem Negieren der Funktion* $\bar{f}$*:*

$$
\begin{aligned}
\bar{f} &= \bar{x}_3 \bar{x}_2 \bar{x}_1 \vee \bar{x}_3 x_2 \bar{x}_1 \vee \bar{x}_3 x_2 x_1 \vee x_3 \bar{x}_2 \bar{x}_1 \vee x_3 x_2 \bar{x}_1 \\
\bar{\bar{f}} &= \overline{\bar{x}_3 \bar{x}_2 \bar{x}_1 \vee \bar{x}_3 x_2 \bar{x}_1 \vee \bar{x}_3 x_2 x_1 \vee x_3 \bar{x}_2 \bar{x}_1 \vee x_3 x_2 \bar{x}_1} \\
f &= (x_3 \vee x_2 \vee x_1)(x_3 \vee \bar{x}_2 \vee x_1)(x_3 \vee \bar{x}_2 \vee \bar{x}_1)(\bar{x}_3 \vee x_2 \vee x_1)(\bar{x}_3 \vee \bar{x}_2 \vee x_1)
\end{aligned}
$$

Durch das erste Negieren wird $\bar{I}_f$ erzeugt, durch das zweite die KDNF mit (1.10h) in eine KKNF umgerechnet. Die negierten Minterme $\bar{m}_i$ werden **Maxterme** genannt.

1.3.4.3 Karnaugh–Diagramm

Karnaugh-Diagramme stellen einen graphischen Ersatz für die Wertetabelle dar. Sie werden zur Vereinfachung von Schaltfunktionen verwendet. Durch die Vereinfachung erhält man Schaltfunktionen mit weniger Unbekannten und weniger Operationen. Dies verringert den Realisierungsaufwand und damit die Kosten für die Herstellung der Schaltung. Da die Diagramme mit zunehmender Anzahl der Variablen schnell sehr groß werden, ist ihre Verwendung nur bis zu $n = 5$ sinnvoll. Die Tafeln der Karnaugh-Diagramme werden induktiv durch Spiegelung gebildet. In jedem Schritt wird das Karnaugh-Diagramm durch eine Spiegelung (→ Abb.1.3, 1.4, 1.5, 1.6 und 1.7) verdoppelt. Die jeweils neu hinzukommende Variable hat in der alten Tafel den Wert 0, in der neuen den Wert 1. Gespiegelte Kästchen werden in der Reihenfolge der ursprünglichen weitergezählt.

Die Darstellung und Minimierung einer Funktion durch Karnaugh-Diagramme wird nachfolgend anhand einiger Beispiele erläutert.

Beispiel 1.13 *Für eine Variable* x_1 *hat die Wertetabelle zwei Zeilen. Jede dieser Zeilen wird im Karnaugh-Diagramm durch ein Kästchen repräsentiert. Die Zeilennummerierung erscheint wieder in der rechten unteren Ecke der Kästchen. Durch* x_1 *wird der Bereich im Karnaugh-Diagramm markiert, in dem* x_1 *den Wert* 1 *hat (im nicht markierten Feld ist* x_1 *somit* 0*).*

i	x_1	$f(x_1)$
0	0	0
1	1	1

$f(x_1) = x_1$

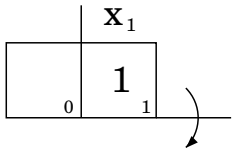


Abb. 1.3: K-Diagramm einer Variablen

Beispiel 1.14 *Durch die Hinzunahme einer weiteren Variablen* x_2 *wird die Anzahl der Zeilen und Kästchen verdoppelt.*

i	x_2	x_1	$f(x_1, x_2)$
0	0	0	0
1	0	1	1
2	1	0	0
3	1	1	1

$$
\begin{aligned}
f(x_1, x_2) &= m_1 \vee m_3 = \bar{x}_2 x_1 \vee x_2 x_1 \\
f(x_1, x_2) &= (\bar{x}_2 \vee x_2) x_1 \\
f(x_1, x_2) &= x_1 \quad \text{(DNF)}
\end{aligned}
$$

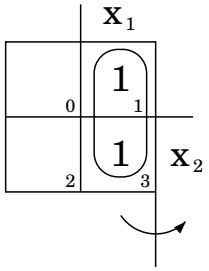


Abb. 1.4: K-Diagramm zweier Variablen

Das Zusammenfassen benachbarter Felder entspricht dem Verschmelzen der beiden Minterme m_1 und m_3.

Dabei fällt die Variable x_2 heraus. Die minimierte Funktion wird als **DNF** *bezeichnet.*

Beispiel 1.15 *In den Mintermen m_0, m_2, m_4 und m_6 gilt $\bar{x}_1$, in den Mintermen m_0, m_1, m_5 und m_4 entsprechend $\bar{x}_2$. Der Durchschnitt der durch die Minterme gebildeten Flächen enthält m_0 und m_4. Diese bilden die Funktion, da sie den Wert 1 haben. Es können folglich auch Terme über den Rand hinweg zusammengefaßt werden.*

i	x_3	x_2	x_1	$f(x_1, x_2, x_3)$
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

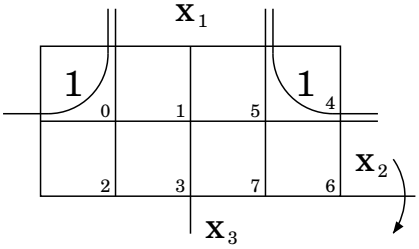


Abb. 1.5: K-Diagramm dreier Variablen

$$
\begin{aligned}
f(\boldsymbol{x}) &= \bar{x}_3 \bar{x}_2 \bar{x}_1 \vee x_3 \bar{x}_2 \bar{x}_1 \\
f(\boldsymbol{x}) &= \bar{x}_2 \bar{x}_1 \quad \text{(DNF)}
\end{aligned}
$$

Durch die Spiegelung des Karnaugh-Diagramms ($\rightarrow$ Abb.1.4) *um die angegebene Achse entsteht ein neues Diagramm* ($\rightarrow$ Abb.1.5) *mit der doppelten Anzahl von Kästchen.*

Das Kästchen mit der Nummer 0 *wird von links oben nach rechts oben gespiegelt. Es erhält dann die Nummer* 4. *Das mit der* 1 *kommt von Position zwei von links nach zwei von rechts und erhält die Nummer* 5. *Auf die gleiche Art und Weise wird die Nummerierung der restlichen Kästchen fortgesetzt.*

Beispiel 1.16 *Es können* $2, 4, 8$ *usw. Felder zusammengefaßt werden. Die resultierenden Konjunktionsterme werden dabei um* $1, 2, 3$ *usw. Variable kürzer.*

i	x_4	x_3	x_2	x_1	$f(\boldsymbol{x})$
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	0
12	1	1	0	0	0
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	0

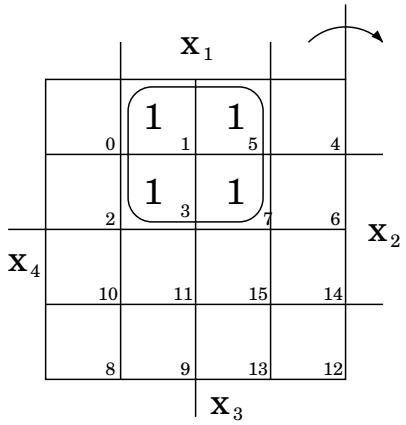


Abb. 1.6: K-Diagramm von vier Variablen

$$
\begin{aligned}
f(\boldsymbol{x}) &= \bar{x}_4\bar{x}_3\bar{x}_2x_1 \vee \bar{x}_4\bar{x}_3x_2x_1 \vee \bar{x}_4x_3\bar{x}_2x_1 \\
&\quad \vee \bar{x}_4x_3x_2x_1 \\
f(\boldsymbol{x}) &= \bar{x}_4x_1 \quad \text{(DNF)}
\end{aligned}
$$

Beispiel 1.17 *Die Schaltfunktion aus dem Karnaugh-Diagramm von fünf Variablen* (→ Abb.1.7) *ist nachfolgend gegeben. Die zusammengefaßten Minterme stehen jeweils unter den Konjunktionstermen.*

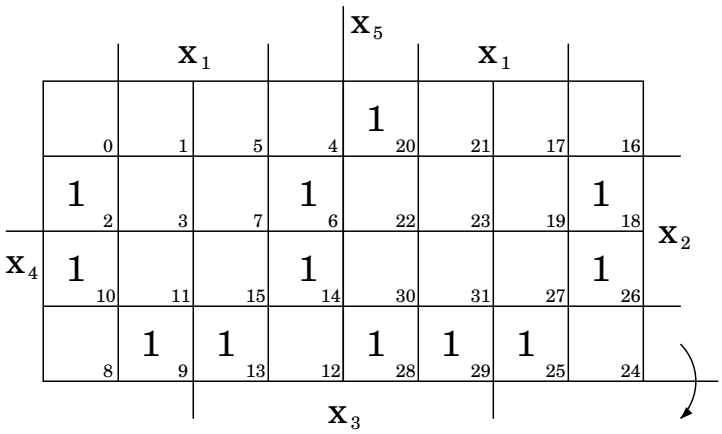


Abb. 1.7: K-Diagramm von fünf Variablen

$$
f(\boldsymbol{x}) = \underbrace{\bar{x}_5x_2\bar{x}_1}_{2,6,10,14} \vee \underbrace{\bar{x}_3x_2\bar{x}_1}_{2,10,18,26} \vee \underbrace{x_4\bar{x}_2x_1}_{9,13,25,29} \vee \underbrace{x_5x_3\bar{x}_2\bar{x}_1}_{20,28} \tag{1.16}
$$

Wird eine durch eine Wertetabelle oder als KDNF gegebene Schaltfunktion optimiert, so erhält man die **Disjunktive Normalform (DNF)** der Funktion. Diese ist im Gegensatz zur KDNF nicht mehr eindeutig.

Um zur **Konjunktiven Normalform (KNF)** zu gelangen, werden die leeren Felder im Karnaugh–Diagramm (Minterme der negierten Funktion $\bar{f}$) zur DNF zusammengefaßt und diese anschließend nochmals negiert.

Beispiel 1.18 *Die DNF der Funktion* (→ Abb.1.8) *ergibt sich zu*

$$f(\boldsymbol{x}) = x_3x_1 \vee x_3x_2 \vee x_2x_1.$$

Das Zusammenfassen der leeren Kästchen (→ Abb.1.9) *führt zur DNF der negierten Funktion*

$$\bar{f}(\boldsymbol{x}) = \bar{x}_3\bar{x}_1 \vee \bar{x}_3\bar{x}_2 \vee \bar{x}_2\bar{x}_1.$$

Nochmaliges Negieren ergibt die KNF:

$$f(\boldsymbol{x}) = \overline{\bar{f}} = (x_3 \vee x_1)(x_3 \vee x_2)(x_2 \vee x_1)$$

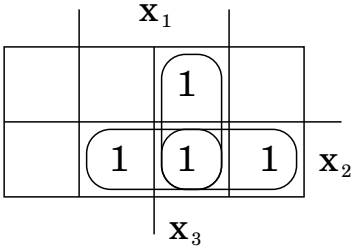


Abb. 1.8: K-Diagramm von $f(\boldsymbol{x})$

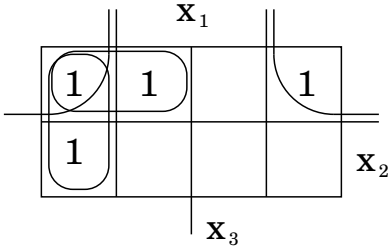


Abb. 1.9: K-Diagramm von $\bar{f}(\boldsymbol{x})$

Um von einer DNF wieder zur KDNF zu gelangen, werden die Konjunktionsterme um die fehlenden Variablen erweitert.

Beispiel 1.19 *Aus der DNF der Funktion aus dem vorhergehenden Beispiel berechnet sich die KDNF wie folgt:*

$$f(\boldsymbol{x}) = x_3x_1 \vee x_3x_2 \vee x_2x_1 \quad \textit{(DNF)}$$

$$f(\boldsymbol{x}) = x_3(x_2 \vee \bar{x}_2)x_1 \vee x_3x_2(x_1 \vee \bar{x}_1) \vee (x_3 \vee \bar{x}_3)x_2x_1$$

$$f(\boldsymbol{x}) = x_3x_2x_1 \vee x_3\bar{x}_2x_1 \vee x_3x_2\bar{x}_1 \vee \bar{x}_3x_2x_1 \quad \textit{(KDNF)}$$

1.3.4.4 Ringsummen

Rinsummen entsprechen ihrem Erscheinungsbild nach den disjunktiven Normalformen. Es werden lediglich die Operatoren $\vee$ durch $\oplus$ ersetzt. So ist z.B. die rechte Seite von

$$x \vee y = x \oplus y \oplus xy$$

eine Ringsumme. Die Darstellung einer Schaltfunktion durch eine Ringsumme ist oft kürzer als die entsprechende DNF oder KNF. Für die schaltungstechnische Realisierung werden dadurch weniger Gatter benötigt. Dies senkt die Kosten für das Chip, den

Testaufwand und das Gehäuse. Die Chipkosten werden durch den geringeren Halbleiterverbrauch, der Testaufwand durch die geringere Anzahl von Fehlermöglichkeiten und die Gehäusekosten durch kleinere Gehäuse mit weniger Anschlüssen gesenkt. Für die Antivalenz gelten folgende Umrechnungen:

$$\boxed{\begin{array}{rcl} x \oplus y & = & x\bar{y} \vee \bar{x}y = (x \vee y)(\bar{x} \vee \bar{y}) \\ \overline{x \oplus y} & = & xy \vee \bar{x}\bar{y} \end{array}} \tag{1.17}$$

Die Umrechnungen folgen unmittelbar aus Tab.1.15.

xy	$x\bar{y}$	$\bar{x}y$	$\bar{x}y \vee x\bar{y}$	$x \oplus y$
00	0	0	0	0
01	0	1	1	1
10	1	0	1	1
11	0	0	0	0

Tab. 1.15: Wertetabelle der Antivalenz

Jede Schaltfunktion kann mit Hilfe der **Entwicklungssätze von Shannon** in eine DNF oder eine Ringsumme umgerechnet werden.

$$\boxed{\begin{array}{rcl} f(x_1,\ldots,x_i,\ldots,x_n) & = & x_i f(x_1,\ldots,1,\ldots,x_n) \vee \bar{x}_i f(x_1,\ldots,0,\ldots,x_n) \\ f(x_1,\ldots,x_i,\ldots,x_n) & = & x_i f(x_1,\ldots,1,\ldots,x_n) \oplus \bar{x}_i f(x_1,\ldots,0,\ldots,x_n) \end{array}} \tag{1.18}$$

Der Beweis von (1.18) erfolgt durch das Einsetzen einer 0 bzw. 1 für x_i. Für $x_i = 1$ fällt der zweite Teil der rechten Seite weg und es bleibt

$$f(x_1,\ldots,1,\ldots,x_n) = f(x_1,\ldots,1,\ldots,x_n)$$

stehen, für $x_i = 0$ wird der vordere Teil der rechten Seite 0 und damit

$$f(x_1,\ldots,0,\ldots,x_n) = f(x_1,\ldots,0,\ldots,x_n).$$

Durch (1.17) und (1.18) werden die wichtigsten Rechenregeln für Antivalenzen bestimmt.

$$\boxed{\begin{array}{lrcl} \text{(a)} & x \oplus 0 & = & x \\ \text{(b)} & x \oplus 1 & = & \bar{x} \\ \text{(c)} & x \oplus x & = & 0 \\ \text{(d)} & x \oplus \bar{x} & = & 1 \\ \text{(e)} & \bar{x} \oplus \bar{y} & = & x \oplus y \\ \text{(f)} & xy & = & x \oplus y \oplus (x \vee y) \\ \text{(g)} & x(y \oplus z) & = & xy \oplus xz \quad \text{(Distributivgesetz)} \\ \text{(h)} & x \vee y & = & x \oplus y \oplus xy \\ \text{(i)} & (v \oplus w)(x \oplus y) & = & vx \oplus vy \oplus wx \oplus wy \end{array}} \tag{1.19}$$

Die Gleichungen (1.19a-e) können direkt mit (1.17) nachgewiesen werden. So gilt z.B. für (1.19a)

$$x \oplus 0 = (x \wedge \bar{0}) \vee (\bar{x} \wedge 0) = x.$$

Die restlichen Gleichungen ergeben sich aus (1.18). Wird z.B. in (1.19f) die rechte Seite nach x entwickelt, so folgt

$$\begin{aligned} f(x,y) &= x(\underbrace{1 \oplus y}_{\bar{y}} \oplus 1) \vee \bar{x}(\underbrace{0 \oplus y}_{y} \oplus y) \\ &= x(\underbrace{\bar{y} \oplus 1}_{y}) \vee \bar{x}(\underbrace{y \oplus y}_{0}) \\ &= xy. \end{aligned}$$

Das Distributivgesetz (1.19g) kann mit (1.17) nachgewiesen werden. So gilt für die linke Seite

$$x(y\bar{z} \vee \bar{y}z) = xy\bar{z} \vee x\bar{y}z$$

und für die rechte

$$\begin{aligned} xy \oplus xz &= xy\overline{xz} \vee \overline{xy}xz \\ &= xy(\bar{x} \vee \bar{z}) \vee (\bar{x} \vee \bar{y})xz \\ &= xy\bar{z} \vee x\bar{y}z. \end{aligned}$$

Die restlichen Beziehungen folgen auf die gleiche Art und Weise.

Beispiel 1.20 *Für die Funktion*

$$f(\boldsymbol{x}) = x_3x_1 \vee x_3x_2 \vee x_2x_1$$

wird die Ringsummendarstellung mit (1.18) *durch Entwicklung nach* x_1 *berechnet. Es folgt*

$$\begin{aligned} f(\boldsymbol{x}) &= x_1(\underbrace{x_2 \vee x_3 \vee x_2x_3}_{x_2 \vee x_3}) \oplus \bar{x}_1(x_2x_3) \\ &= x_1(x_2 \oplus x_3 \oplus x_2x_3) \oplus (x_1 \oplus 1)x_2x_3 \\ &= x_1x_2 \oplus x_1x_3 \oplus \underbrace{x_1x_2x_3 \oplus x_1x_2x_3}_{0} \oplus x_2x_3 \\ &= x_3x_1 \oplus x_3x_2 \oplus x_2x_1. \end{aligned}$$

Für diese spezielle Funktion mußte lediglich $\vee$ *durch* $\oplus$ *ersetzt werden. Dies gilt jedoch nicht allgemein!*

Liegt die Schaltfunktion als KDNF vor, so erhält man eine eindeutige Ringsummendarstellung, indem man $\vee$ durch $\oplus$ ersetzt.

$$\boxed{f(x_1,\ldots,x_n) = \bigvee_{i \in I_f} m_i(x_1,\ldots,x_n) = \bigoplus_{i \in I_f} m_i(x_1,\ldots,x_n)} \tag{1.20}$$

Dies kann mit Hilfe der vollständigen Induktion leicht nachgewiesen werden.

Wegen (1.14) gilt für $i \neq j$

$$\begin{aligned} m_i \vee m_j &= m_i \oplus m_j \oplus \underbrace{m_im_j}_{0} \\ &= m_i \oplus m_j. \end{aligned}$$

Als Induktionsannahme soll dies auch für k Minterme gelten und m_j nicht in f_k vorkommen. Dann folgt

$$\underbrace{(m_0 \oplus m_1 \oplus \ldots \oplus m_{k-1})}_{k\text{-mal}} m_j = f_k m_j = 0.$$

Der Induktionsschluß nach $k+1$ liefert mit (1.19h)

$$f_{k+1} = f_k \vee m_j = f_k \oplus m_j \oplus \underbrace{f_k m_j}_{0} = f_k \oplus m_j.$$

Beispiel 1.21 *Für eine Funktion als KDNF gilt somit die folgende Substitution der Operationen:*

$$\begin{aligned} f(\boldsymbol{x}) &= x_3x_2x_1 \vee x_3x_2\bar{x}_1 \vee x_3\bar{x}_2x_1 \vee \bar{x}_3x_2x_1 \\ &= x_3x_2x_1 \oplus x_3x_2\bar{x}_1 \oplus x_3\bar{x}_2x_1 \oplus \bar{x}_3x_2x_1 \end{aligned}$$

Die KDNF, KKNF und die aus der KDNF gewonnene Ringsummendarstellung sind jeweils bis auf die Reihenfolge der einzelnen Terme eindeutig.

Beispiel 1.22 *Die Funktion*

$$f(\boldsymbol{x}) = \overline{(\overline{\bar{x}_4x_3 \vee x_2})(\bar{x}_3 \vee x_2)}$$

kann mit Hilfe von (1.18) *in eine DNF bzw. Ringsumme umgerechnet werden. Es folgt für die Entwicklung nach* x_2

$$\begin{aligned} f(\boldsymbol{x}) &= x_2[1] \oplus \bar{x}_2 \left[\overline{\overline{\bar{x}_4x_3}\bar{x}_3}\right] \\ &= x_2 \oplus \bar{x}_2 \left[\overline{(x_4 \vee \bar{x}_3)\bar{x}_3}\right] = x_2 \oplus \bar{x}_2 \left[\bar{x}_4x_3 \vee x_3\right] \\ &= x_2 \oplus x_3\bar{x}_2 = x_2 \oplus (x_2 \oplus 1)x_3 \\ &= \underbrace{x_2 \oplus x_3 \oplus x_3x_2}_{\text{Ringsumme}} = \underbrace{x_3 \vee x_2}_{\text{DNF}}. \end{aligned}$$

1.3.5 Minimale Normalformen

Aus den bereits erwähnten Kostengründen ist man immer bestrebt, möglichst kurze Schaltfunktionen mit wenigen Operationen und Variablen zu erhalten. Eine Verkürzung und damit Optimierung der Schaltfunktion wird durch das Verschmelzen von benachbarten Mintermen erreicht. Dies ist rechnerisch oder mit Hilfe von Karnaugh-Diagrammen möglich. Beide Methoden sind aus Komplexitätsgründen nur für Funktionen von weniger als 6 Variablen sinnvoll. Die praktisch vorkommenden Fälle von 10, 20 oder 30 Eingangsvariablen erfordern den Einsatz eines Rechners und eines passenden Optimierungsprogramms. Ein Algorithmus, der in einem Programm implementiert werden kann, ist das nachfolgend beschriebene Quine-McCluskey Verfahren.

Zur Beschreibung des Verfahrens wird der Begriff **Primterm** benötigt. Ein Primterm (auch **Primimplikant** genannt) ist ein kürzester, formal nicht mehr teilbarer Konjunktionsterm einer Funktion $f(\boldsymbol{x})$. Im Karnaugh-Diagramm entspricht einem Primterm ein größtes zu einem Konjunktionsterm gehörendes Feld.

Beispiel 1.23 *Die im nebenstehenden Karnaugh–Diagramm zusammengefaßten Minterme sind* alle Primterme *der Funktion* $f(\boldsymbol{x}) = x_4x_3\bar{x}_1 \vee x_4x_3\bar{x}_2 \vee x_3x_2\bar{x}_1 \vee \bar{x}_4x_2 \vee \bar{x}_4x_1 \vee \bar{x}_2x_1$. *Die Terme* $\bar{x}_4x_1$ *sowie* $x_4x_3\bar{x}_2$ *und* $x_3x_2\bar{x}_1$ *(gestrichelt) sind dabei überflüssig, da ihre Minterme bereits in den anderen Primtermen enthalten sind. Die Funktion wird daher durch* $f(\boldsymbol{x}) = x_4x_3\bar{x}_1 \vee \bar{x}_4x_2 \vee \bar{x}_2x_1$ *vollständig beschrieben. Ein Verfahren zum Auffinden aller Primterme folgt im nächsten Abschnitt.*

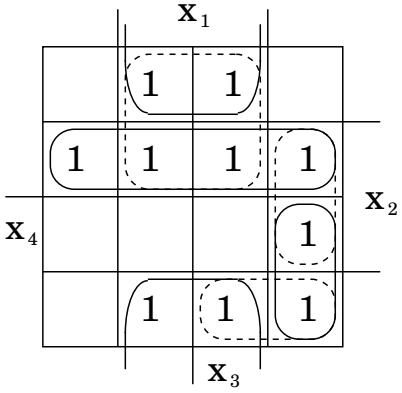


Abb. 1.10: Primterme von $f(\boldsymbol{x})$

1.3.5.1 Quine-McCluskey Verfahren

Primterme entstehen durch das Zusammenfassen einer maximalen Anzahl von Mintermen einer Funktion $f(\boldsymbol{x})$. Dies ist auch der Grundgedanke des Verfahrens von Quine-McCluskey.

Dabei wird zunächst zu jedem Minterm m_i ein Minterm m_j gesucht, der sich von m_i in genau einer Variablen unterscheidet (z.B. unterscheidet sich $x_3x_2\bar{x}_1$ zu $x_3x_2x_1$ in x_1). Dann gilt

$$m_i \vee m_j = m_{i,j}(x_k \vee \bar{x}_k) = m_{i,j},$$

wobei $m_{i,j}$ kein Minterm mehr ist, da er x_k nicht mehr enthält. Anschließend versucht man die verkürzten Terme $m_{i,j}$ mit weiteren $m_{k,l}$ zu verschmelzen

$$m_{i,j} \vee m_{k,l} = m_{i,j,k,l}(x_m \vee \bar{x}_m) = m_{i,j,k,l}\,,$$

so daß immer kürzere Konjunktionsterme entstehen. Die Terme, die dann nicht mehr weiter zusammengefaßt werden können, sind die Primterme der Funktion.

Beispiel 1.24 *Für das vorhergehende Beispiel werden mit Hilfe des Quine-McCluskey Verfahrens alle Primterme der Funktion bestimmt* (→ Tab.1.16). *Dazu werden zunächst alle Minterme der Anzahl der Einsen nach zeilenweise angeordnet. Anschließend wird versucht, alle Terme aus der ersten Gruppe mit allen Termen der zweiten zu verschmelzen. Das Ergebnis wird in die zweite Spalte* ($m_{i,j}$) *eingetragen und die Terme* m_i *und* m_j *in der ersten Spalte abgehakt. Das selbe wird mit der zweiten und dritten, vierten und fünften Gruppe, usw. durchgeführt. Dadurch entstehen in der zweiten Spalte neue Gruppen, für welche der Algorithmus dann wiederholt wird.*

i	m_i		i,j	$m_{i,j}$		i,j,k,l	$m_{i,j,k,l}$
1	0001	$\surd$	1,3	$00-1$	$\surd$	1,3,5,7	$0--1$
2	0010	$\surd$	1,5	$0-01$	$\surd$	1,5,3,7	$0--1$
3	0011	$\surd$	1,9	-001	$\surd$	1,5,9,13	$--01$
5	0101	$\surd$	2,3	$001-$	$\surd$	1,9,5,13	$--01$
6	0110	$\surd$	2,6	$0-10$	$\surd$	2,3,6,7	$0-1-$
9	1001	$\surd$	3,7	$0-11$	$\surd$	2,6,3,7	$0-1-$
12	1100	$\surd$	5,7	$01-1$	$\surd$		
7	0111	$\surd$	5,13	-101	$\surd$		
13	1101	$\surd$	6,7	$011-$	$\surd$		
14	1110	$\surd$	6,14	-110			
			9,13	$1-01$	$\surd$		
			12,13	$110-$			
			12,14	$11-0$			

Tab. 1.16: Ablauf des Quine-McCluskey Verfahrens

Wenn keine weiteren Terme mehr zusammengefaßt werden können, wird das Verfahren beendet. Die nicht gekennzeichneten Terme bilden die Menge der Primterme. Im Beispiel sind dies

$$p_1 = x_3x_2\bar{x}_1,\ p_2 = x_4x_3\bar{x}_2,\ p_3 = x_4x_3\bar{x}_1,\ p_4 = \bar{x}_4x_1,\ p_5 = \bar{x}_2x_1,\ p_6 = \bar{x}_4x_2.$$

Das Quine-McCluskey Verfahren bestimmt in einem ersten Schritt alle Primterme einer Schaltfunktion. Dabei wird zwischen notwendigen und redundanten Termen nicht unterschieden. Um eine minimale DNF zu erhalten, muß daher ein weiterer Berechnungsschritt folgen. Zur Veranschaulichung dieses Schrittes werden die Prim- und Minterme in einer Tabelle ($\rightarrow$ Tab.1.17) angeordnet. Die jeweils in den Primtermen enthaltenen Minterme werden aus Tab.1.16 entnommen und in Tab.1.17 durch ein $\times$ markiert. Da die DNF einer Funktion immer alle Minterme (siehe z.B. (1.16)) enthalten muß, wird nun versucht, aus Tab.1.17 diejenigen Primterme zu bestimmen, die diese Forderung erfüllen und gleichzeitig möglichst kurz sind, d.h. viele Minterme enthalten. Mit den so berechneten Primtermen bekommt man dann eine „Überdeckung“ (alle Minterme sind enthalten) der gesuchten Funktion. Die Tab.1.17 wird daher auch **Mengenüberdeckungstabelle** genannt. Ist in einer Spalte nur ein $\times$ enthalten, so wird die zugehörige Zeile **Kernzeile** und der Primterm **Kernprimimplikant** genannt. Da nur der Kernprimimplikant diesen Minterm enthält, gehört er sicher zur Funktion $f(\boldsymbol{x})$. Der Algorithmus läuft wie folgt ab:

a) *Kernzeilen und die betroffenen Spalten werden gestrichen.* Sie gehören zur Menge der funktionsbildenden Primterme.

b) *Dominierende und gleiche (bis auf eine) Spalten werden gestrichen.* Zu den dominierenden Spalten muß es Spalten geben, deren Primterme ($\times$) eine Teilmenge der Menge der Primterme der dominierenden Spalte sind. Nach dem Streichen der dominierenden Spalte wird der gestrichene Minterm von den in der Teilmenge enthaltenen Primtermen weiterhin überdeckt. Gleiche Spalten können aus dem selben Grund bis auf einen Repräsentanten gestrichen werden.

c) *Dominierte Zeilen werden gestrichen.* Zeilen, die eine Teilmenge der Minterme ($\times$) einer dominierenden Zeile enthalten, können ebenfalls gestrichen werden, da ja die Minterme in der dominierenden Zeile (nicht gestrichenen) erhalten bleiben.

Der Unterschied zwischen der Zeilen- und Spaltendominanz ist zu beachten. Bei der Spaltendominanz werden dominierende Spalten (viele $\times$) gestrichen, bei der Zeilendominanz dominierte (wenige $\times$).

Beispiel 1.25 *In* Tab.1.17 *sind m_2 und m_9 nur in einer Zeile enthalten. Die zugehörigen Zeilen enthalten die Kernprimimplikanten p_6 und p_5. Die Zeile p_6 und die zugehörigen Spalten m_2, m_3, m_6 und m_7 sowie die Zeile p_5 mit den Spalten m_1, m_5, m_9 und m_{13} können gestrichen werden. Der redundante Primterm p_4 fällt dabei heraus. Übrig bleiben die Primterme p_1, p_2 und p_3 mit den Mintermen m_{12} und m_{14}. Zeile p_3 ist zu den beiden anderen Zeilen dominant. Die Zeilen p_1 und p_2 können folglich gestrichen werden. Die minimale DNF lautet somit:*

$$f(\boldsymbol{x}) = p_6 \vee p_5 \vee p_3 = \underbrace{\bar{x}_4 x_2}_{m_2 \vee m_3 \vee m_6 \vee m_7} \vee \underbrace{\bar{x}_2 x_1}_{m_1 \vee m_5 \vee m_9 \vee m_{13}} \vee \underbrace{x_4 x_3 \bar{x}_1}_{m_{12} \vee m_{14}}$$

	m_1	m_2	m_3	m_5	m_6	m_7	m_9	m_{12}	m_{13}	m_{14}
p_1					$\times$					$\times$
p_2								$\times$	$\times$	
p_3								$\times$		$\times$
p_4	$\times$		$\times$	$\times$		$\times$				
p_5	$\times$			$\times$			$\times$		$\times$	
p_6		$\times$	$\times$		$\times$	$\times$				

Tab. 1.17: Primtermtabelle (Mengenüberdeckungstabelle)

Beispiel 1.26 *Für die nachfolgend gegebene* ($\rightarrow$ Tab.1.18) *Mengenüberdeckungstabelle wird eine minimale Zeilenmenge bestimmt.*

	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	s_{14}
z_1					$\times$					$\times$		$\times$		$\times$
z_2						$\times$				$\times$		$\times$		$\times$
z_3		$\times$		$\times$			$\times$		$\times$				$\times$	
z_4	$\times$					$\times$		$\times$		$\times$		$\times$		$\times$
z_5				$\times$	$\times$					$\times$		$\times$		$\times$
z_6	$\times$		$\times$								$\times$		$\times$	
z_7		$\times$					$\times$		$\times$				$\times$	
z_8			$\times$								$\times$		$\times$	

Tab. 1.18: Mengenüberdeckungstabelle

a) Die einzige Kernzeile in Tab.1.18 *ist z_4. Durch ihr Streichen fallen gleichzeitig die Spalten s_1, s_6, s_8, s_{10}, s_{12} und s_{14} weg. Nach dem Streichen bleibt* Tab.1.19 *übrig.*

	s_2	s_3	s_4	s_5	s_7	s_9	s_{11}	s_{13}
z_1				×				
z_3	×		×		×	×		×
z_5			×	×				
z_6		×					×	×
z_7	×				×	×		×
z_8		×					×	×

Tab. 1.19: Mengenüberdeckungstabelle nach Schritt a)

b) Gleiche Spalten sind $\{s_{11}, s_3\}$ und $\{s_9, s_7, s_2\}$. Bis auf die mit dem jeweils niedrigsten Index werden die identischen Spalten gestrichen. Es bleiben die Spalten s_3 und s_2 übrig. Dominant ist s_{13} zu s_2 und s_3. Die dominante Spalte s_{13} wird ebenfalls gestrichen. In Tab.1.20 *sind die restlichen Spalten und Zeilen eingetragen.*

c) Die Zeile z_8 ist mit z_6 identisch und kann damit entfallen. Zeile z_7 wird von z_3 dominiert, sie kann damit ebenfalls gestrichen werden. Das gleiche gilt für z_1 und z_5.

	s_2	s_3	s_4	s_5
z_1				×
z_3	×		×	
z_5			×	×
z_6		×		
z_7	×			
z_8		×		

Tab. 1.20: – nach Schritt b)

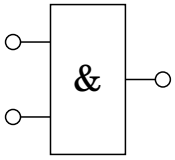
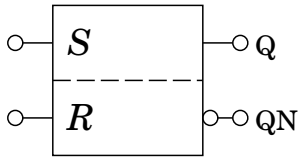
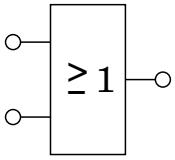
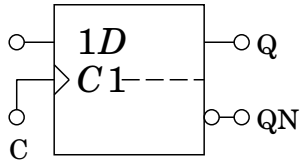
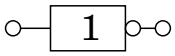
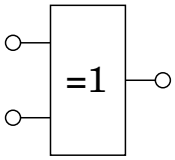
	s_2	s_3	s_4	s_5
z_3	×		×	
z_5			×	×
z_6		×		

Tab. 1.21: – nach Schritt c)

Eine Mengenüberdeckung wird somit mit den Zeilen $\{z_4, z_6, z_5, z_3\}$ erzielt.

1.4 Logische Schaltungen

Die übersichtlichste Darstellungsform für Schaltfunktionen sind Schaltpläne. Zum Zeichnen von Schaltplänen werden die logischen Operationen zuvor in Schaltsymbole umgesetzt (→ Tab.1.22). Diese werden dann entsprechend der zu realisierenden Funktion durch Signale (Leitungen) miteinander verbunden. Der Schaltplan einer logischen Schaltung nimmt dabei die Form eines gerichteten Graphen an. Die Knoten des Graphen sind die Operationen bzw. Schaltsymbole, die Kanten die Signale bzw. Leitungen.

Schaltsymbole nach DIN 40900 Teil 12 (IEC 617-12)			
kombinatorische Elemente		speichernde Elemente	
Typ		Typ	
UND		SR-Flip-Flop	
ODER		D-Flip-Flop	
NICHT			
EXOR			

Tab. 1.22: Beispiele für Schaltsymbole

Logische Schaltungen werden in **Schaltnetze** (kombinatorische Automaten) und in **Schaltwerke** (sequentielle Automaten) eingeteilt. Schaltwerke enthalten speichernde Elemente (z.B. Flip-Flops), Schaltnetze nur kombinatorische (Gatter).

1.4.1 Kombinatorische Schaltungen

Ein Schaltnetz ist eine Anordnung, welche die Werte von logischen Variablen an ihren Eingängen zu logischen Ausganswerten verknüpft. Da die Anordnung rückführungsfrei ist (→ Abb.1.11), kann ihr logisches Verhalten durch Schaltfunktionen eindeutig beschrieben werden.

Abb. 1.11: Schaltnetz

Dem Schaltnetz entspricht somit folgendes System von Gleichungen:

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix} = \begin{bmatrix} f_1(\boldsymbol{x}) \\ \vdots \\ f_m(\boldsymbol{x}) \end{bmatrix} = \boldsymbol{f}(\boldsymbol{x}) \text{ mit } \boldsymbol{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix}$$

Beispiel 1.27 *Ein Halbaddierer* (HA) *ist durch seine Wertetabelle* (→ Tab.1.23) *bzw. die Schaltfunktionen*

$$s_i = \bar{a}_i b_i \vee a_i \bar{b}_i = a_i \oplus b_i \quad \textit{und} \quad u_{i+1} = a_i b_i$$

gegeben.

a_i	b_i	s_i	u_{i+1}
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Tab. 1.23: Wertetabelle des HAs

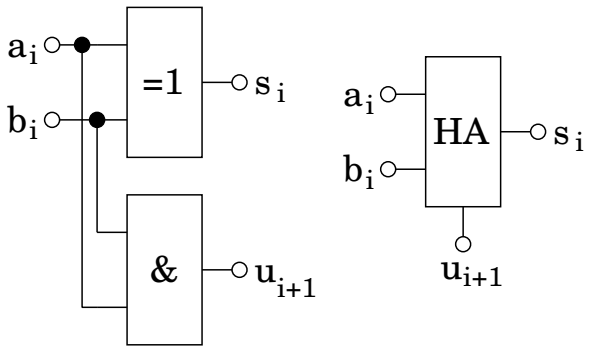


Abb. 1.12: Schaltnetz und Symbol des HAs

Das i-te Summenbit ist s_i, u_{i+1} ist der $i+1$-te Übertrag (→ Abb.1.12).

Beispiel 1.28 *Ein Volladdierer* (→ Tab.1.24) *ist ein weiteres Beispiel für ein einfaches Schaltnetz.*

a_i	b_i	u_i	s'_i	u'_{i+1}	u''_{i+1}	s_i	u_{i+1}
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	1	0	0	1	0
0	1	1	1	0	1	0	1
1	0	0	1	0	0	1	0
1	0	1	1	0	1	0	1
1	1	0	0	1	0	0	1
1	1	1	0	1	0	1	1

Tab. 1.24: Wertetabelle des VAs

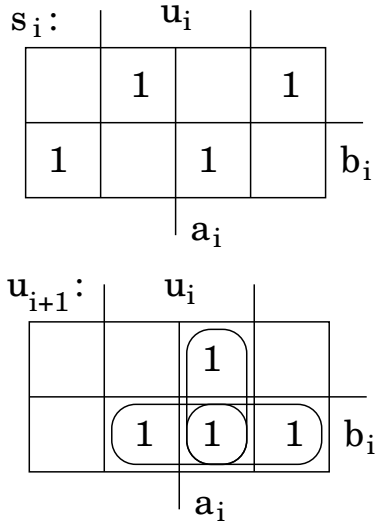


Abb. 1.13: KD für s_i und u_{i+1}

Die Schaltfunktionen s_i und u_{i+1} folgen aus der Wertetabelle ($\rightarrow$ Tab.1.24) *bzw. aus den Karnaugh-Diagrammen* ($\rightarrow$ Abb.1.13). *Für die Summe s_i ist die Ringsumme die kürzeste Darstellungsform:*

$$s_i = a_i(b_i u_i \vee \bar{b}_i \bar{u}_i) \vee \bar{a}_i(b_i \bar{u}_i \vee \bar{b}_i u_i) = \underbrace{a_i(\overline{b_i \oplus u_i})}_{f_1} \vee \underbrace{\bar{a}_i(b_i \oplus u_i)}_{f_2}$$

Da wegen $a_i \bar{a}_i = 0$ auch $f_1 f_2 = 0$ ist, folgt $f_1 \vee f_2 = f_1 \oplus f_2$ und damit

$$\begin{aligned} s_i &= a_i(b_i \oplus u_i \oplus 1) \oplus \\ &\quad (a_i \oplus 1)(b_i \oplus u_i) \\ &= a_i \oplus b_i \oplus u_i. \end{aligned}$$

Für den Übertrag u_{i+1} ergibt sich ebenfalls mit $f_1 f_2 = 0$:

$$u_{i+1} = \underbrace{a_i(\overbrace{u_i \vee b_i}^{u_i \oplus b_i \oplus b_i u_i})}_{f_1} \vee \underbrace{\bar{a}_i b_i u_i}_{f_2}$$
$$= a_i u_i \oplus a_i b_i \oplus b_i u_i$$

Dies entspricht der kürzesten DNF der Funktion:

$$u_{i+1} = a_i u_i \vee a_i b_i \vee b_i u_i$$

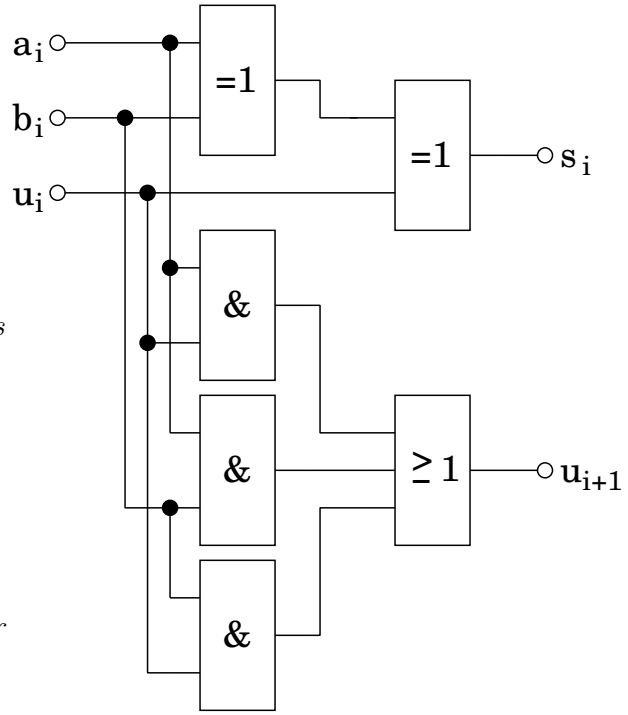


Abb. 1.14: Aufbau des VAs durch Gatter

Eine mögliche Schaltungsrealisierung ist durch Abb.1.14 *gegeben. Werden für die Zwischensummen* ($\rightarrow$ Tab.1.24) *HA verwendet, so folgt der Aufbau des Volladdierers nach* Abb.1.15.

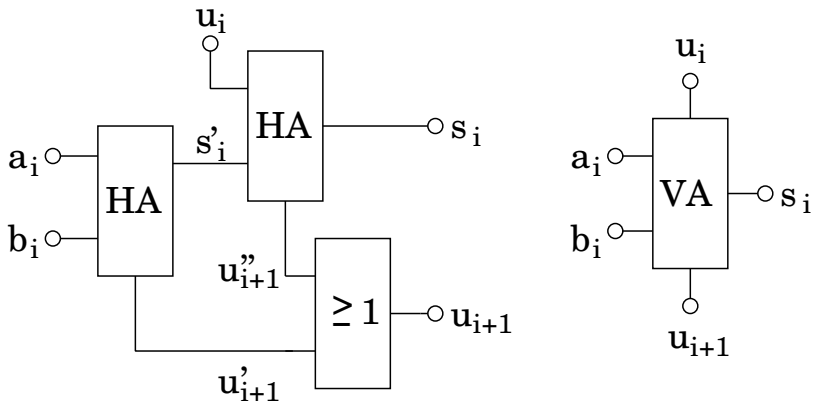


Abb. 1.15: Der Volladdierer und sein Symbol

Aus Halb- und Volladdierern kann man nun z.B. ein Addierwerk für zwei n-Bit lange Dualzahlen

$$\begin{aligned} a &= (a_{n-1}a_{n-2}\cdots a_1a_0)_2 \quad \text{und} \\ b &= (b_{n-1}b_{n-2}\cdots b_1b_0)_2 \end{aligned}$$

aufbauen. Für die niedrigstwertigen Bits (**LSB**, Least Significant Bit) wird ein Halbaddierer, für die restlichen, $(n-1)$ Volladdierer zusammengeschaltet. Die Addition $s = a + b$ liefert ein $(n+1)$-tes Bit s_n, den Übertrag aus der Addition der höchstwertigen Bits (**MSB**, Most Significant Bit) des $(n-1)$-ten Volladdierers. Ein Beispiel für die Addition von 2-Bit Summanden ist in Abb.1.16 gegeben. Alle möglichen Additionen von 2-Bit Summanden mit den jeweiligen Summen stehen in Tab.1.25.

i	b_1	a_1	b_0	a_0	u_1	s_2	s_1	s_0
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	0	1
3	0	0	1	1	1	0	1	0
4	0	1	0	0	0	0	1	0
5	0	1	0	1	0	0	1	1
6	0	1	1	0	0	0	1	1
7	0	1	1	1	1	1	0	0
8	1	0	0	0	0	0	1	0
9	1	0	0	1	0	0	1	1
10	1	0	1	0	0	0	1	1
11	1	0	1	1	1	1	0	0
12	1	1	0	0	0	1	0	0
13	1	1	0	1	0	1	0	1
14	1	1	1	0	0	1	0	1
15	1	1	1	1	1	1	1	0

Tab. 1.25: Wertetabelle eines 2-Bit Addierwerks

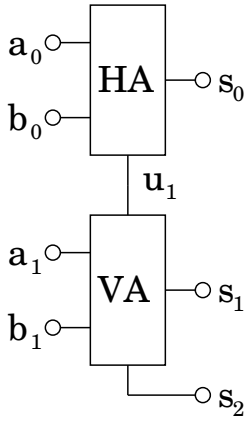


Abb. 1.16: Addierwerk für $n = 2$

Es gibt verschiedene Möglichkeiten eine Schaltfunktion zu realisieren. Sie kann in eine Gatterschaltung, ein **ROM** (Read Only Memory), ein **FPGA** (Field Programmable Gate Array), ein **PLA** (Programmable Logic Array) usw. umgesetzt werden.

Beispiel 1.29

Bei einem ROM bestimmen die Summanden die Adresse der gespeicherten Additionsergebnisse. So werden durch die 2-Bit breiten Summanden a_1a_0 und b_1b_0 durch $b_1a_1b_0a_0$ 16 Ergebniswörter $s_2s_1s_0$ der Breite 3 Bit ausgewählt. Für die Addition wird folglich ein ROM mit einem Speicherplatz von $16 \cdot 3 = 48$ Bit ($\rightarrow$ Abb.1.17) benötigt.

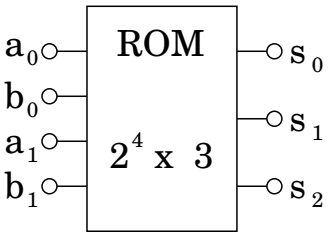


Abb. 1.17

Beispiel 1.30 *Die DNF der Schaltfunktion kann direkt in ein PLA umgesetzt werden. Für die DNF der drei Summanden s_2, s_1 und s_0 folgt* (→ Tab.1.25) *mit Hilfe von Karnaugh-Diagrammen* (→ Abb.1.18)*:*

$$
\begin{aligned}
s_0 &= a_0\bar{b}_0 \vee b_0\bar{a}_0 \\
s_1 &= b_1a_1b_0a_0 \vee \bar{b}_1\bar{a}_1b_0a_0 \vee b_1\bar{a}_1\bar{b}_0 \vee b_1\bar{a}_1\bar{a}_0 \vee \bar{b}_1a_1\bar{a}_0 \vee \bar{b}_1a_1\bar{b}_0 \\
s_2 &= b_1b_0a_0 \vee a_1b_0a_0 \vee b_1a_1
\end{aligned}
$$

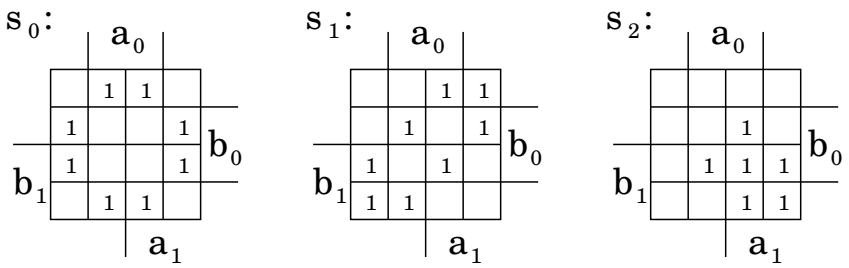


Abb. 1.18: Karnaugh-Diagramme des 2-Bit Addierwerks

Zur direkten Umsetzung der DNF, werden UND-Gatter für die Konjunktionsterme und ODER-Gatter für die Disjunktionsterme benötigt (→ Abb.1.19, der Schrägstrich symbolisiert mehrere Leitungen)*.*

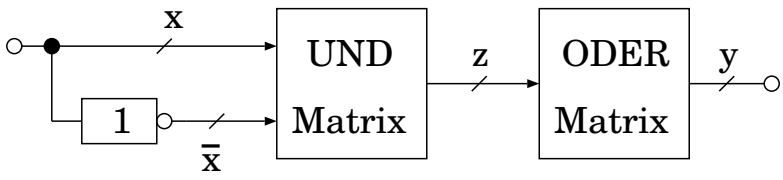


Abb. 1.19: Darstellung der DNF als PLA

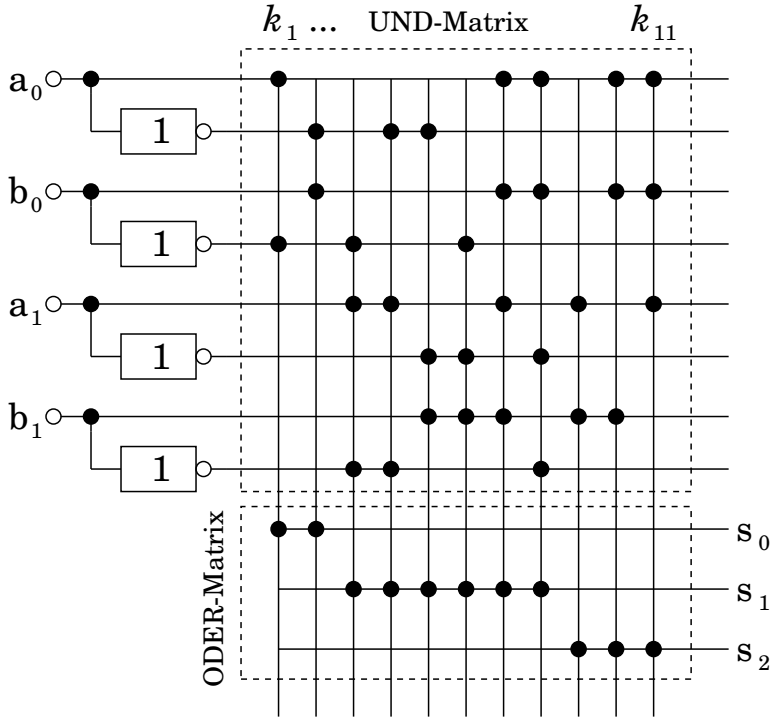


Abb. 1.20: PLA Realisierung des 2-Bit Addierwerks

Die Spalten (→ Abb.1.20) *enthalten die Konjunktionsterme, z.B.* $k_1 = a_0\bar{b}_0$*, die Zeilen die Disjunktionsterme, z.B.* $s_0 = k_1 \vee k_2 = a_0\bar{b}_0 \vee \bar{a}_0 b_0$.

Zur Erhöhung der Übersichtlichkeit der Darstellung, werden die UND- bzw. ODER-Verknüpfungen nur durch einen Punkt wiedergegeben (→ Abb.1.21).

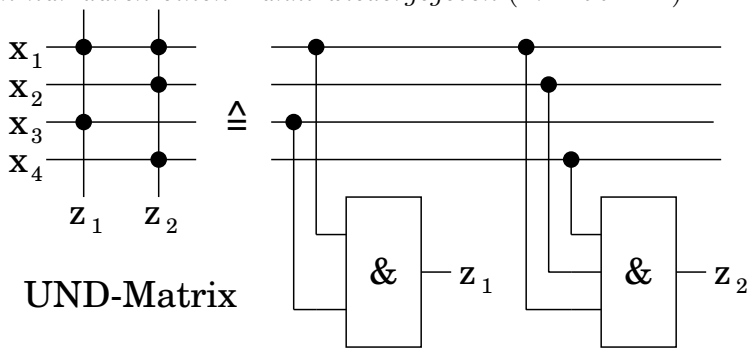


Abb. 1.21: Vereinfachte PLA-Darstellung

1.4.2 Sequentielle Schaltungen

Bei kombinatorischen Schaltungen hängt zu jedem Zeitpunkt t der Ausgangsvektor $\boldsymbol{y}$ direkt von der zu diesem Zeitpunkt gegebenen Eingangsbelegung $\boldsymbol{x}$ ab, und die Anzahl der möglichen Schaltfunktionen ist endlich (→ Seite 13). Diese Einschränkung wird aufgehoben, wenn neben der augenblicklichen Eingangsbelegung $\boldsymbol{x}^t$ beliebige Schaltungszustände aus früheren Zeitpunkten, $\boldsymbol{z}^{t-1}$, $\boldsymbol{z}^{t-2}$, $\boldsymbol{z}^{t-3}$, usw. zur Berechnung der Ausgangswerte $\boldsymbol{y}^t$ hinzukommen.

Um Werte aus der Vergangenheit verwenden zu können, müssen diese zuvor gespeichert werden. Dies erfolgt jeweils zu festen Tanktzeitpunkten. Die Übernahme kann dabei abhängig vom Speicheraufbau durch die steigende oder fallende Taktflanke geschehen. Übernommen wird dabei der am Eingang des Speichers zu diesem Zeitpunkt anliegende Wert. Die Speicherinhalte $\boldsymbol{z}^t$ bilden zusammen mit den aktuellen Werten der Eingangsvariablen $\boldsymbol{x}^t$ die Ausgangswerte $\boldsymbol{y}^t$ und die zukünftigen Speicherwerte $\boldsymbol{z}^{t+1}$ ($\rightarrow$ Abb.1.22). Zum nächsten Taktzeitpunkt $(t+1)$ werden dann die aktuellen Werte an den Speichereingängen $\boldsymbol{z}^{t+1}$ in den Speicher eingelesen.

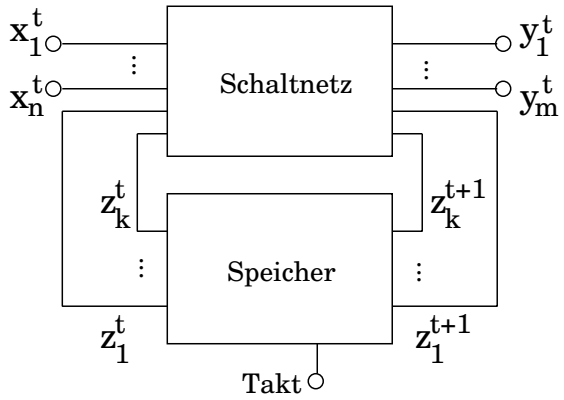


Abb. 1.22: Schaltwerk

Aus Abb.1.22 können folgende Gleichungen abgelesen werden:

$$\boldsymbol{y}^t = \boldsymbol{f}(\boldsymbol{x}^t, \boldsymbol{z}^t) \quad \text{und} \quad \boldsymbol{z}^{t+1} = \boldsymbol{g}(\boldsymbol{x}^t, \boldsymbol{z}^t)$$

Werden die Zeitpunkte für die Übernahme der Signalwerte in den Speicher äquidistant $(t, t+1, t+2, \ldots)$ gewählt, so spricht man von einem **synchronen** Schaltwerk. Erfolgt die Übernahme direkt durch die Änderung der Eingangssignalwerte, so erhält man ein **asynchrones** Schaltwerk. Bei asynchronen Schaltwerken kann es schaltungstechnisch bedingt zu Störungen im Schaltverhalten kommen. Diese Störungen sind meist durch Rückkopplungen verursachte Instabilitäten (Schwingungen), die in synchronen Schaltungen nicht vorkommen.

Durch die Verwendung von Taktsignalen bleiben die zwischen den Tanktzeitpunkten auftretenden Störungen bzw. Signaländerungen vor den Speicherelementen ohne Wirkung auf das Ausgangsverhalten der Schaltung (nur die zum Taktzeitpunkt anliegenden Werte werden übernommen). Setzt man daher Speicher vor die Ausgänge kritischer Schaltungsteile, so können damit interne Störungen durch eine geeignete Wahl der Tanktzeitpunkte „herausgefiltert“ werden.

Auch im Hinblick auf einen testfreundlichen Entwurf (DFT, Design for Testability) sind synchrone Schaltungen den asynchronen vorzuziehen.

Als Speicherelemente werden in der Regel Flip-Flops ($\rightarrow$ Tab.1.22) und Register verwendet.

Beispiel 1.31 *Das dargestellte D-Flip-Flop* ($\rightarrow$ Abb.1.23) *übernimmt den am Eingang* D *anliegenden Signalwert mit der positiven* (0$\rightarrow$1) *Taktflanke* C ($\rightarrow$ Tab.1.26).

Eingänge		*Ausgänge*	
D	*C*	*Q*	*QN*
1	↑	1	0
0	↑	0	1

Tab. 1.26: Wertetabelle des D-FFs

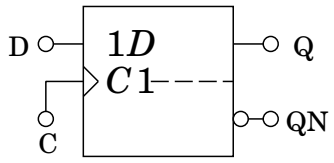


Abb. 1.23: D-FF

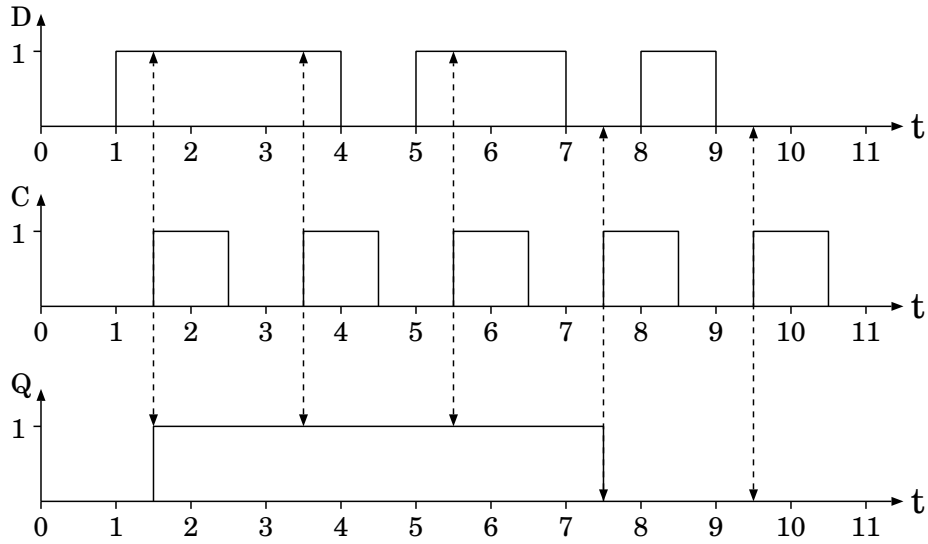


Abb. 1.24: Beispiel für das Verhalten eines D-FFs

Das Impulsdiagramm (→ Abb.1.24) *verdeutlicht die Arbeitsweise des D-Flip-Flops. Die Übernahme der Daten vom Eingang* D *in den Speicher* Q *ist durch Pfeile markiert. Die fallende Flanke hat keine Auswirkungen auf die Werte der Ausgänge* Q *und* QN *des Flip-Flops.*

Der Entwurf synchroner Schaltwerke erfolgt mit Hilfe von **Übergangsgraphen**. Die Zustände (Speicherinhalte) $\boldsymbol{z}$ des Schaltwerks werden hierbei auf die Knoten des Graphen abgebildet, die gerichteten Kanten enthalten die für den Übergang verantwortlichen Eingangsgrößen $\boldsymbol{x}$ und den augenblicklichen Ausgangswert $\boldsymbol{y}$.

Der Entwurf einer sequentiellen Schaltung kann beliebig komplex werden.

Beispiel 1.32 *Es ist eine Schaltung zu entwickeln, die eine bestimmte Folge von Eingangskombinationen erkennt. Die Schaltung hat die Eingänge a und b, die zu erkennende Folge sei ab,* $\bar{a}b$, $\bar{a}\bar{b}$. *Eine erkannte Folge soll am Ausgang y durch eine* 1 *signalisiert werden.*

Zustandstabelle:
Zustand 0: *Anfangszustand*
Zustand 1: 11 *ist aufgetreten*
Zustand 2: 11, 01 *ist aufgetreten*
Zustand 3: 11, 01, 00 *ist aufgetreten*

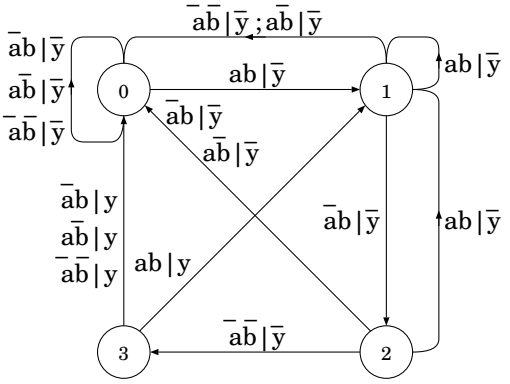


Abb. 1.25: Übergangsgraph

Für 4 Zustände werden zwei binäre Speicherelemente benötigt. Der Übergangsgraph für das zu entwerfende synchrone Schaltwerk ist in Abb.1.25 *dargestellt.*

Aus dem Übergangsgraphen folgt die Übergangstabelle (→ Tab.1.27).

		Zustand			Folgezustand			Ausgang
a^t	b^t	Nr.	z_1^t	z_0^t	Nr.	z_1^{t+1}	z_0^{t+1}	y^t
0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	1	0	0	0	1	0	1	0
0	0	1	0	1	0	0	0	0
0	1	1	0	1	2	1	0	0
1	0	1	0	1	0	0	0	0
1	1	1	0	1	1	0	1	0
0	0	2	1	0	3	1	1	0
0	1	2	1	0	0	0	0	0
1	0	2	1	0	0	0	0	0
1	1	2	1	0	1	0	1	0
0	0	3	1	1	0	0	0	1
0	1	3	1	1	0	0	0	1
1	0	3	1	1	0	0	0	1
1	1	3	1	1	1	0	1	1

Tab. 1.27: Übergangstabelle

Zur Speicherung der angegebenen Zustände werden D-Flip-Flops verwendet.

Für sie gilt:

$$Q_i^t = z_i^t \text{ und } D_i^t = z_i^{t+1} \text{ für } i = 0, 1$$

Aus der Übergangstabelle können für z_0^{t+1}, z_1^{t+1} *und* y^t *mit Hilfe von Karnaugh-Diagrammen die minimierten Schaltfunktionen aufgestellt werden* (→ Abb.1.26).

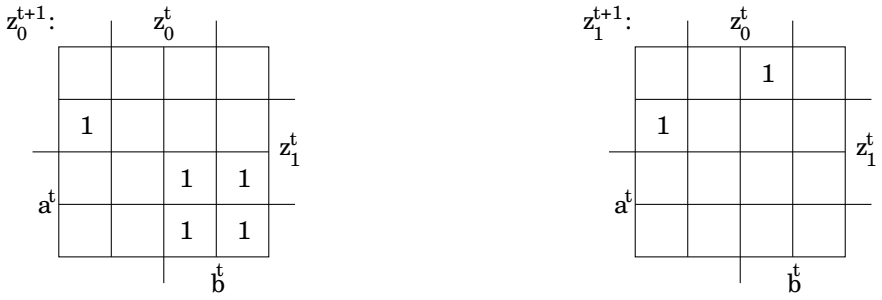
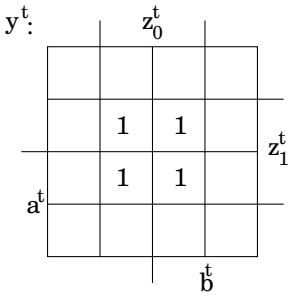


Abb. 1.26: Karnaugh-Diagramme der Schaltfunktionen



Für die Schaltfunktionen folgt:

$$
\begin{aligned}
z_0^{t+1} &= a^t b^t \vee \bar{a}^t \bar{b}^t z_1^t \bar{z}_0^t \\
z_1^{t+1} &= \bar{a}^t \bar{b}^t z_1^t \bar{z}_0^t \vee \bar{a}^t b^t \bar{z}_1^t z_0^t \\
y^t &= z_0^t z_1^t
\end{aligned}
$$

Der Schaltplan des Schaltwerks ist in Abb.1.27 *wiedergegeben.*

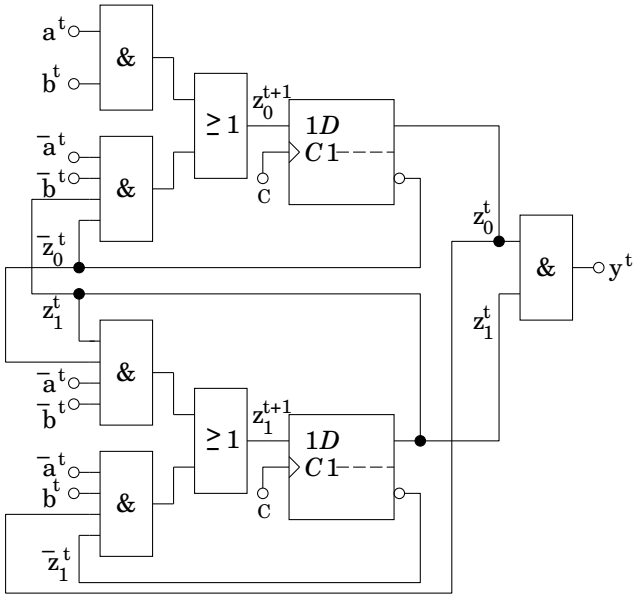


Abb. 1.27: Schaltbild

Beispiel 1.33 *Neben dem D-Flip-Flop werden in synchronen Schaltungen häufig auch JK-Flip-Flops eingesetzt. Ihre Funktion wird durch die in* Tab.1.28 *gegebenen Operationen definiert. Wie aus der Tabelle zu entnehmen ist, kann das Flip-Flop zwei Zustände annehmen. Es kann daher mit einem D-Flip-Flop realisiert werden.*

J^t	K^t	Q^{t+1}
0	0	Q^t
0	1	0
1	0	1
1	1	$\bar{Q}^t$

Tab. 1.28: Funktionstabelle des JK-FFs

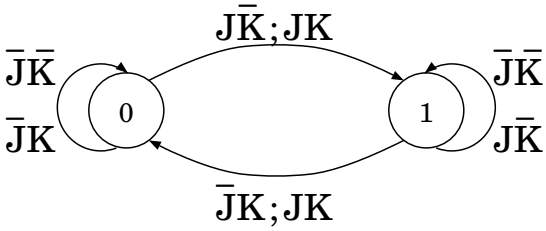


Abb. 1.28: Übergangsgraph des JK-FFs

Aus der Funktionstabelle folgen mit $z = Q$ der Übergangsgraph (→ Abb.1.28) *und die Übergangstabelle* (→ Tab.1.29).

J^t	K^t	z^t	z^{t+1}
0	0	0	0
0	1	0	0
1	0	0	1
1	1	0	1
0	0	1	1
0	1	1	0
1	0	1	1
1	1	1	0

Tab. 1.29: Übergangstabelle des JK-FFs

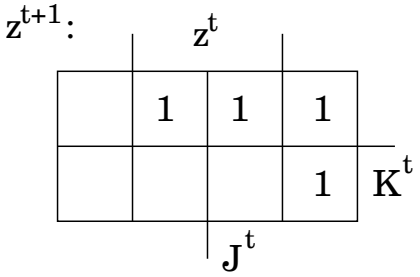


Abb. 1.29: KD des JK-FFs

Aus dem Karnaugh-Diagramm (→ Abb.1.29) *folgt dann*

$$\begin{aligned} z^{t+1} &= z^t\bar{K}^t \vee \bar{z}^t J^t, \quad \textit{bzw. mit} \quad z = Q \\ Q^{t+1} &= Q^t\bar{K}^t \vee \bar{Q}^t J^t. \end{aligned}$$

Das JK-Flip-Flop und dessen Symbol sind in Abb.1.30 *dargestellt.*

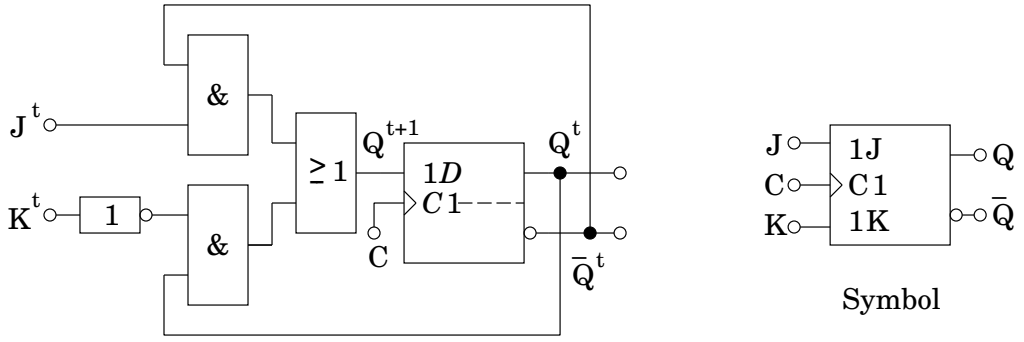


Abb. 1.30: Aufbau des JK-FFs und sein DIN-Symbol

Beispiel 1.34 *Weniger häufig werden RS-Flip-Flops* (→ Tab.1.30) *verwendet. Die Zustände, die sich für* $R = S = 1$ *ergeben, hängen von der Implementierung ab. Sie werden häufig als verbotene Zustände bezeichnet (undefiniert).*

Die Realisierung des synchronen RS-Flip-Flops erfolgt wiederum mit einem D-Flip-Flop.

R^t	S^t	Q^{t+1}
0	0	Q^t
0	1	1
1	0	0
1	1	–

Tab. 1.30: Funktionstabelle des RS-FFs

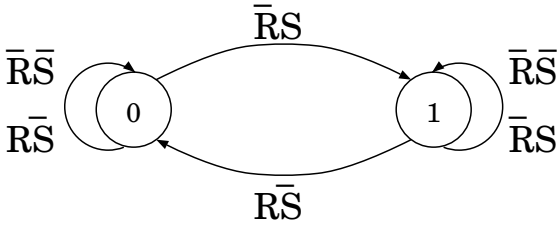


Abb. 1.31: Übergangsgraph des RS-FFs

Aus der Funktionstabelle, dem Übergangsgraphen (→ Abb.1.31)*, der Übergangstabelle* (→ Tab.1.31) *und dem Karnaugh-Diagramm* (→ Abb.1.32) *folgt*

$$\begin{aligned} z^{t+1} &= z^t\bar{R}^t \vee S^t\bar{R}^t = \bar{R}^t(z^t \vee S^t), \quad \textit{bzw. mit} \quad z = Q \\ Q^{t+1} &= \bar{R}^t(Q^t \vee S^t). \end{aligned}$$

R^t	S^t	z^t	z^{t+1}
0	0	0	0
0	1	0	1
1	0	0	0
1	1	0	–
0	0	1	1
0	1	1	1
1	0	1	0
1	1	1	–

Tab. 1.31: Übergangstabelle des RS-FFs

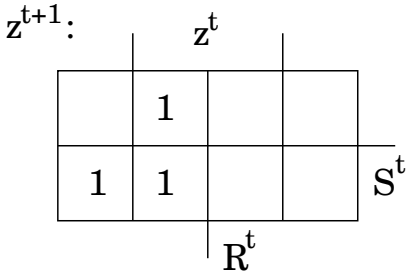


Abb. 1.32: KD des RS-FFs

Das RS-Flip-Flop und dessen Symbol sind in Abb.1.33 *dargestellt.*

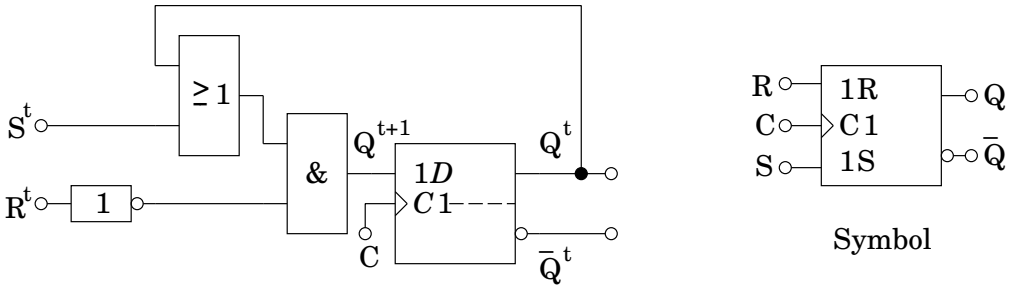


Abb. 1.33: Aufbau des RS-FFs und sein DIN-Symbol

Vergleicht man Tab.1.29 *mit* Tab.1.31 *so fällt auf, daß mit* $K = R$ *und* $J = S$ *die Funktion des RS-Flip-Flops von einem JK-Flip-Flop übernommen werden kann.*

Für $J = K = 1$ kann außerdem mit dem JK-Flip-Flop eine „Toggle“-Funktion,

$$Q^{t+1} = \bar{Q}^t, \tag{1.21}$$

ausgeführt werden (→ Tab.1.32). *Diese Funktion kann auch durch ein T-Flip-Flop realisiert werden.*

Beispiel 1.35 *Die Funktion des T-Flip-Flops ist durch* (1.21) *gegeben* (→ Tab.1.32). *Die Realisierung des synchronen T-Flip-Flops erfolgt ebenfalls mit einem D-Flip-Flop.*

T^t	Q^{t+1}
0	Q^t
1	$\bar{Q}^t$

Tab. 1.32: Funktionstabelle des T-FFs

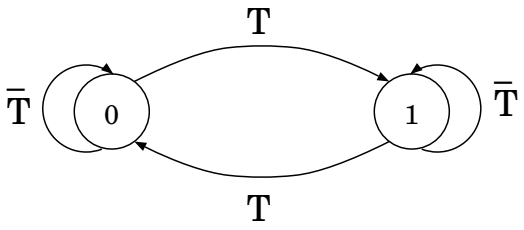


Abb. 1.34: Übergangsgraph des T-FFs

Aus der Funktionstabelle, dem Übergangsgraphen (→ Abb.1.34), *der Übergangstabelle* (→ Tab.1.33) *und dem Karnaugh-Diagramm* (→ Abb.1.35) *folgt dann*

$$\begin{aligned} z^{t+1} &= z^t\bar{T}^t \vee \bar{z}^t T^t = z^t \oplus T^t, \quad \textit{bzw. mit} \quad z = Q \\ Q^{t+1} &= Q^t \oplus T^t. \end{aligned}$$

T^t	z^t	z^{t+1}
0	0	0
1	0	1
0	1	1
1	1	0

Tab. 1.33: Übergangstabelle des T-FFs

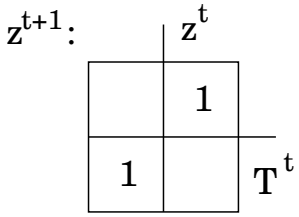


Abb. 1.35: KD des T-FFs

Das T-Flip-Flop und dessen Symbol sind in Abb.1.36 *dargestellt.*

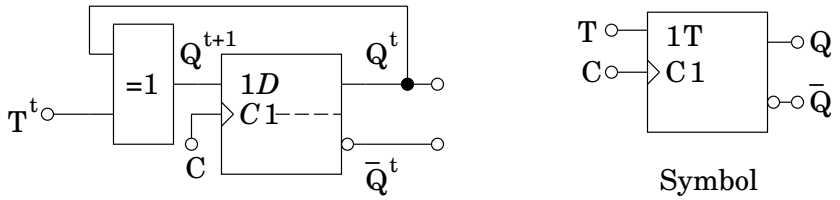


Abb. 1.36: Aufbau des T-FFs und sein DIN-Symbol

Die hier vorgestellten flankengesteuerten Flip-Flops lassen sich alle mit einem D-Flip-Flop und entsprechender Zusatzlogik realisieren. Jede synchrone Schaltung kann daher aus den kombinatorischen Grundgattern sowie dem flankengesteuerten D-Flip-Flop aufgebaut werden.

2 Logiksimulation

Durch die Logiksimulation wird der Schaltungsentwurf auf seine korrekte Funktion hin überprüft. Die Verifikation besteht im allgemeinen aus einer Analyse der Schaltung im jeweiligen Entwurfsstadium und einem Vergleich der Analyseergebnisse mit der vorgegebenen Sollfunktion (z.B. Wertetabelle oder Übergangstabelle). Die Aussagekraft der berechneten Ergebnisse $\boldsymbol{y}(t)$ hängt dabei wesentlich von den gewählten Eingangsbitmustern (**Stimuli**) ab.

Die Modellierung der Schaltung ist auf unterschiedlichen Abstraktionsebenen (→ Kap. 1.1) möglich.

Die höchste Ebene stellt die **Verhaltensebene** (behavioral-level) dar. Auf dieser Ebene wird die Schaltung durch ihre Schaltfunktion bzw. durch Übergangsgraphen und -tabellen beschrieben. Die Schaltungseigenschaften werden dazu durch eine besondere Beschreibungssprache (VHDL, Verilog) formuliert.

Eine grobe strukturelle Beschreibung der Schaltung erfolgt auf der **Register-Transfer-Ebene**. Die Elemente der Struktur sind Speicher, arithmetische Einheiten und Kontrollblöcke, die durch Signalbusse verbunden werden.

Die Elemente der **Gatter-Ebene** sind in DIN 40900 Teil 12 zusammengefaßt (Tabelle 1.22). Die Verbindung der Elemente erfolgt durch unidirektionale Signale.

Eine niedrigere Abstraktionsebene ist die **Schalter-Ebene** (switch-level). Die Elemente sind Schalter, die das Schaltverhalten von Transistoren nachbilden. Die Verknüpfung der Schalter erfolgt durch bidirektionale Signalnetze, eine feste Signalflußrichtung liegt nicht mehr vor.

Die nachfolgend tiefere Ebene ist die **Schaltkreisebene** (circuit-level). Die Elemente werden durch elektrische Bauelemente (Transistoren, Dioden, Kondensatoren, Spulen, Widerstände, usw.) repräsentiert. Die **Analogsimulation** berechnet auf dieser Ebene alle Strom- und Spannungsverläufe an den Schaltungselementen. Der digitale Charakter der Schaltung bleibt dabei unberücksichtigt.

Die Logiksimulation wird auf Gatter-Ebene durchgeführt. Das Schaltverhalten der logischen Elemente wird dazu aus den Ergebnissen einer Netzwerkanalyse auf Schaltkreisebene bestimmt (→ Abb.2.1). Da der Logiksimulator nur eindeutige Zustände kennt, muß das Schaltverhalten der realen physikalischen Schaltung auf diese diskreten Zustände abgebildet werden.

Beispiel 2.1 *Für ein UND-Gatter wird der Umladevorgang an seinem Ausgang in eine Verzögerungszeit umgesetzt. Dazu wird das reale Verhalten des Elements nach einem Signalwechsel an einem Eingang mittels Analogsimulation ermittelt. Anschließend werden zwei Schwellen für die logischen Werte* 0 *und* 1 *festgelegt. Für CMOS (Complementary*

Metal-Oxid-Semiconductor) entspricht ein Spannungswert $\leq 1.5V$ *der logischen* 0*, ein Wert* $\geq 3.5V$ *der logischen* 1 *(die typischen Werte sind* $0V$ (0) *und* $5V$ (1)*). Da nur Spannungswerte jenseits der Schwellen zu eindeutigen Signalwerten führen, kann das reale Schaltverhalten sehr gut durch diskrete Verzögerungszeiten* (→ Abb.2.1) *nachgebildet werden.*

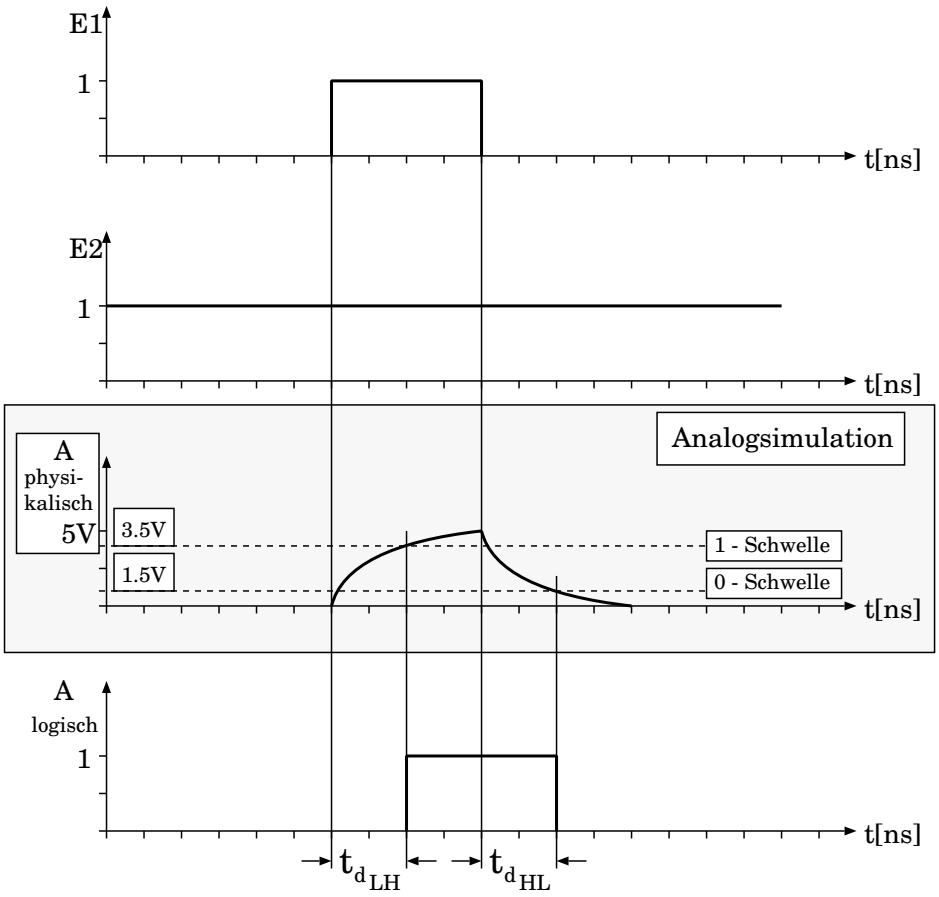


Abb. 2.1: Bestimmung der Verzögerungszeiten

Das logische Element wird durch das Hinzufügen der Verzögerungszeiten zu einem Simulationselement (→ Abb.2.2). Das logische (Wertetabelle) und das zeitliche Verhalten der Elemente wird in der Regel mit Hilfe einer Prozedur oder Funktion in einer passenden (Pascal, C oder eigene Beschreibungssprache) Programmiersprache beschrieben. Alle so modellierten Elemente werden dann in einer Simulationsbibliothek zusammengefaßt. Typische Simulationselemente (**Primitives**) sind z.B. Gatter, Flip-Flops, Zähler und Multiplexer.

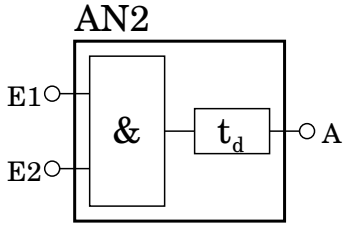


Abb. 2.2: Simulationselement

Die Verzögerungszeit eines Elements wird bei unbelastetem Ausgang (Lastkapazität $CL = 0$) bestimmt. Ist der Ausgang mit einem weiteren Element beschaltet, so wirkt

die Eingangskapazität dieses Elements als Lastkapazität, die bei jedem Signalwechsel umgeladen werden muß. Dies führt zu einer weiteren zeitlichen Verzögerung des Signals. Da auch die Verbindungsleitungen Kapazitäten (der Leitungscharakter wird allein durch die Verlustkapazität beschrieben, Widerstände und Induktivitäten werden vernachlässigt) darstellen, müssen sie ebenfalls bei der Berechnung von Signalverzögerungen berücksichtigt werden.

Beispiel 2.2 *Für den Pfad* E1 → A (→ Abb.2.3)

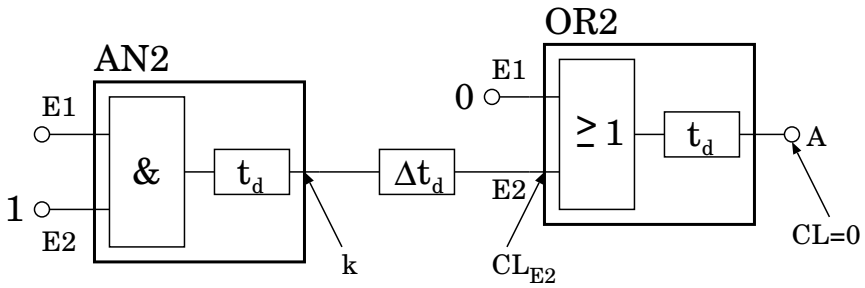


Abb. 2.3: Verzögerungszeit über zwei Elemente

ist die zeitliche Verzögerung für einen Ausgangssignalwechsel *von* 0 *nach* 1 *(LH) bzw. von* 1 *nach* 0 *(HL) zu berechnen. Die Verzögerungszeiten werden dazu aus den Datenblättern* (→ Tab.2.1) *entnommen. Für den Pfad von* E1 *nach* A *folgt dann:*

$$\begin{aligned} t_{D_{HL}} &= \underbrace{t_{d_{HL}} + kHL}_{\text{AN2}} \cdot \underbrace{CL_{E2} + t_{d_{HL}}}_{\text{OR2}} + \underbrace{\Delta t_{d_{HL}}}_{\text{Leitung}} \quad \textit{bzw.} \\ t_{D_{LH}} &= t_{d_{LH}} + kLH \cdot CL_{E2} + t_{d_{LH}} + \Delta t_{d_{LH}} \end{aligned}$$

Dabei ist

t_D	*die Pfadverzögerungszeit* (E1 → A),
t_d	*die Schaltzeit eines Elements,*
k	*die Treiberstärke des Sendeelements,*
CL_{E2}	*die Eingangskapazität am Pin* E2 *des Elements* OR2 *und*
Δt_d	*die Leitungslaufzeit zwischen den Elementen.*

Auszüge aus den Datenblättern							
Element		$t_{d_{HL}}[ns]$	$t_{d_{LH}}[ns]$	$k_{HL}\left[\frac{ns}{LU}\right]$	$k_{LH}\left[\frac{ns}{LU}\right]$	$CL_{E1}[LU]$	$CL_{E2}[LU]$
UND	AN2	0.8	0.7	0.10	0.16	1.03	0.99
ODER	OR2	1.0	0.6	0.11	0.16	0.99	1.02
EXOR	EO	1.2	0.8	0.15	0.28	1.68	1.67
Inverter	IV	0.3	0.4	0.15	0.16	0.89	

Tab. 2.1: Simulationsdaten

Mit den Werten aus Tab.2.1 *und* $\Delta t_{d_{HL}} = \Delta t_{d_{LH}} = 0.3ns$ *folgt:*

$$\begin{aligned} t_{D_{HL}} &= 0.8ns + 0.10\frac{ns}{LU} \cdot 1.02LU + 1.0ns + 0.3ns = 2.202ns \\ t_{D_{LH}} &= 0.7ns + 0.16\frac{ns}{LU} \cdot 1.02LU + 0.6ns + 0.3ns = 1.7632ns \end{aligned}$$

Die Einheit „Loading Unit“ (LU) hängt von der verwendeten Technologie ab. Für einen 1.5μ *CMOS-Prozeß liegt ihr Wert bei* $153fF$ $(1LU = 153fF)$*. Die Treiberstärke „k“ gibt die Umladefähigkeit der Ausgangsstufe des Sendeelements an. Liefert die Ausgangsstufe viel Strom, so ist k klein, liefert sie wenig Strom, so ist k und damit die Verzögerungszeit groß.*

Da die Signallaufzeiten aus den Leitungslängen berechnet werden, stehen sie normalerweise erst nach der Layoutsynthese zur Verfügung. Wegen der oft nicht vernachlässigbaren Zeiten ist es jedoch wünschenswert, Werte bereits zu einem möglichst frühen Zeitpunkt zur Verfügung zu haben. Für Entwurfskonzepte mit fester Chipfläche wird dies durch statistisch ermittelte Werte erreicht. Stehen die geschätzten Laufzeiten zur Verfügung, so ist eine Simulation mit diesen Größen einer Simulation ohne immer vorzuziehen, da sie wesentlich genauere Ergebnisse über das zu erwartende reale Schaltungsverhalten liefern kann.

Bei einer Simulation ohne Laufzeiten werden normalerweise nur die Schaltzeiten der Elemente berücksichtigt ($\Delta t_d = 0$, $CL = 0$).

Die Verzögerungszeiten können von den typischen Werten aus dem Datenblatt durch

a) Parameterstreuungen des Herstellungsprozesses, sowie durch

b) Betriebstemperatur- und

c) Versorgungsspannungsschwankungen

abweichen. Diese Schwankungen werden durch einen Korrekturfaktor bei der Berechnung der Verzögerungszeiten ($k \cdot t_D$) berücksichtigt. Der Faktor wird aus den jeweiligen Abweichungen aus a), b) und c) berechnet:

$$k = k_{VDD} \cdot k_\vartheta \cdot k_P$$

Es ist allgemein üblich, für den Korrekturfaktor den Buchstaben „k“ zu verwenden. Dieses k ist nicht mit der Treiberstärke k_{HL} bzw. k_{LH} zu verwechseln!

Beispiel 2.3 *Für die typischen Werte aus dem Datenblatt gilt (CMOS):*

$$k_{VDD} = 1\,(VDD = 5V),\ k_\vartheta = 1\,(\vartheta = 25^oC),\ k_P = 1 \text{ und damit } k = 1$$

Beispiel 2.4 *Für den „worst-case“ folgt (CMOS):*

$$k_{VDD} = 1.15\,(VDD = 4.5V),\ k_\vartheta = 1.22\,(\vartheta = 85^oC),\ k_P = 1.4 \text{ und damit } k = 1.96$$

Alle Zeiten müssen mit dem Faktor ≈ 2 *multipliziert werden. Die Schaltung ist damit nur noch halb so schnell als im typischen Betriebsfall. Der Prozeßfaktor* $k_P = 1.4$ *tritt häufig für „***Dies***“ (Chips) am Rande des „***Wafers***“ (Siliziumscheibe mit mehreren Dies) auf.*

Beispiel 2.5 *Im „best-case“ gilt (CMOS):*

$$VDD = 5.5V,\ \vartheta = 0^oC,\ k_P = 0.8 \text{ und damit } k = 0.5$$

Die Schaltung ist im „best-case“ doppelt so schnell als im typischen Betriebsfall. Der Prozeßfaktor $k_P = 0.8$ gilt für Schaltungen in der Mitte des Wafers, da dort die Optik zur Belichtung der Scheiben am genauesten (schärfsten) zeichnet.

Die Logiksimulation ist für alle möglichen Betriebsfälle durchzuführen. Falls Fehler in einem der Fälle auftreten, ist die Schaltung gegebenenfalls zu ändern.

Parallel zur Schaltungssimulation werden durch das Simulationsprogramm wichtige Zeitbedingungen überwacht und Verstöße gemeldet.

Beispiel 2.6 *Bei einem D-Flip-Flop* (→ Seite 36) *sind zur korrekten Funktion vier Zeitbedingungen einzuhalten. Es sind dies die „**clock-low-width**“ t_{CWL}, die „**clock-high-width**“ t_{CWH}, die „**setup-time**“ t_s und die „**hold-time**“ t_h. Die Taktdauer bestimmt die maximale Betriebsfrequenz der Schaltung* (→ Abb.2.4)*. Sie hängt von der gegebenen Technologie ab. Die setup-Zeit legt fest, wann ein Signal vor der übernehmenden Taktflanke am D-Eingang anliegen muß und die hold-Zeit, wie lange es nach der Übernahme stabil bleiben muß* (→ Abb.2.4)*.*

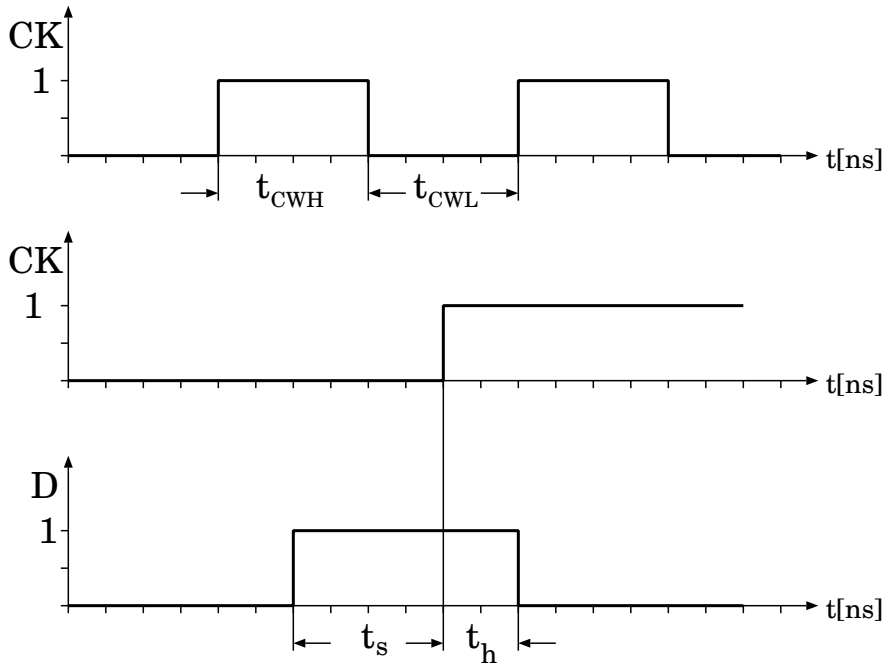


Abb. 2.4: Überwachte Zeiten bei einem D-FF

Der Simulator arbeitet auf einer diskreten Zeitbasis. Alle Zeiten sind folglich immer ein ganzzahliges Vielfaches einer Grundzeiteinheit (z.B. 1 ps). Die Logiksimulation erfolgt **listengesteuert** und **ereignisgetrieben**. Bei einer listengesteuerten Simulation wird die Schaltung durch Netzlisten beschrieben. Die Netzlisten enthalten alle Schaltungselemente sowie deren Verbindungen. Sie werden aus den Zeichnungsdaten (**schematic entry**) mittels „**Netlistcompilation**“ erzeugt.

Die ereignisgetriebene Simulation nutzt die im Durchschnitt niedrige Schaltungsaktivität zur Einsparung von Rechenzeit. Es werden immer nur die Elemente neu berechnet, an denen eine Signalwertänderung am Eingang zu einer Ausgangswertänderung führt. Nach Berechnung der neuen Werte sowie deren Verzögerungszeiten werden diese von einer **Ereignisverwaltung** in eine Warteschlange eingetragen. Die Ereignisverwaltung sorgt dann dafür, daß die gespeicherten Werte nach Ablauf der Verzögerungszeit zur Ausführung gebracht werden. Zeitpunkte, zu denen Signalwertänderungen und Elementeauswertungen stattfinden, werden aktive Zeitpunkte genannt. Beim Weiterschalten der Simulationszeit wird immer der nächste Zeitpunkt in der Ereignisschlange aktiviert.

2.1 Laufzeitbedingte Effekte

Durch die Verzögerungszeiten kommt es zu zeitlichen Verschiebungen im Signalverlauf. Diese Verschiebungen sind die Ursache für unterschiedliche laufzeitabhängige Effekte. Sie führen oft zu Fehlfunktionen, die ohne Verzögerungszeiten ($t_D = 0$) nicht auftreten würden.

2.1.1 Spikes

Spikes sind sehr kurze Impulse an Gatterausgängen, verursacht durch kurz aufeinander folgende Signalwechsel an ihren Eingängen.

Beispiel 2.7 *Durch einen Signalwechsel am Eingang* E1 *eines EXOR-Gatters von* $0 \rightarrow 1$ *bei gleichzeitig stabilem Eingang* E2 (1) ($\rightarrow$ Abb.2.5) *müßte der Ausgang* A *auf den Wert* 0 *wechseln. Bevor sich der Ausgangswert einstellen kann, wechselt jedoch das Signal am Eingang* E2 *von* $1 \rightarrow 0$.

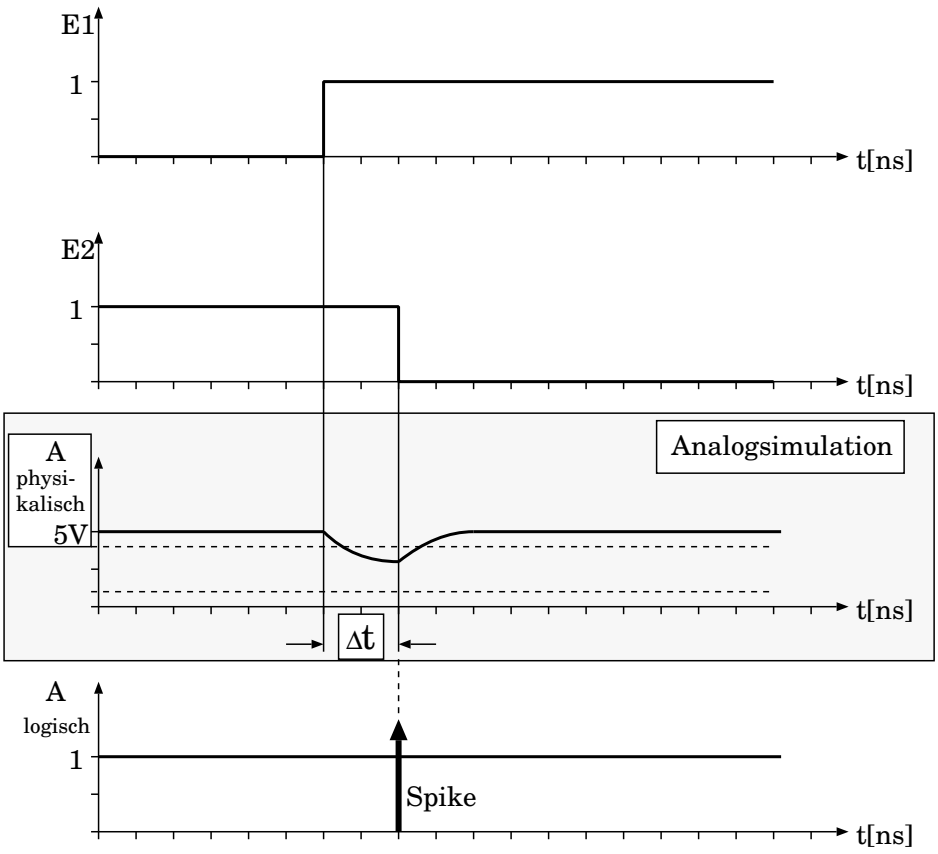


Abb. 2.5: Definition eines Spike

Daraufhin geht der Ausgangswert von einem Zwischenwert wieder auf 1*. Da der Zwischenwert nicht eindeutig ist, kann die Reaktion eines eventuell nachfolgenden Gatters nicht vorhergesagt werden. Der Logiksimulator macht den Schaltungsentwickler auf diese Gefahr durch einen Spike aufmerksam.*

Ein Spike tritt auf, wenn $\Delta t < t_d$ ist.

2.1.2 Races und Hazards

Beispiel 2.8 *Findet z.B. an einem EXOR-Gatter ohne Berücksichtigung von Laufzeiten ein gleichzeitiger Signalwechsel an beiden Eingängen statt* (→ Abb.2.6a)*, so bleibt das Ausgangssignal unverändert.*

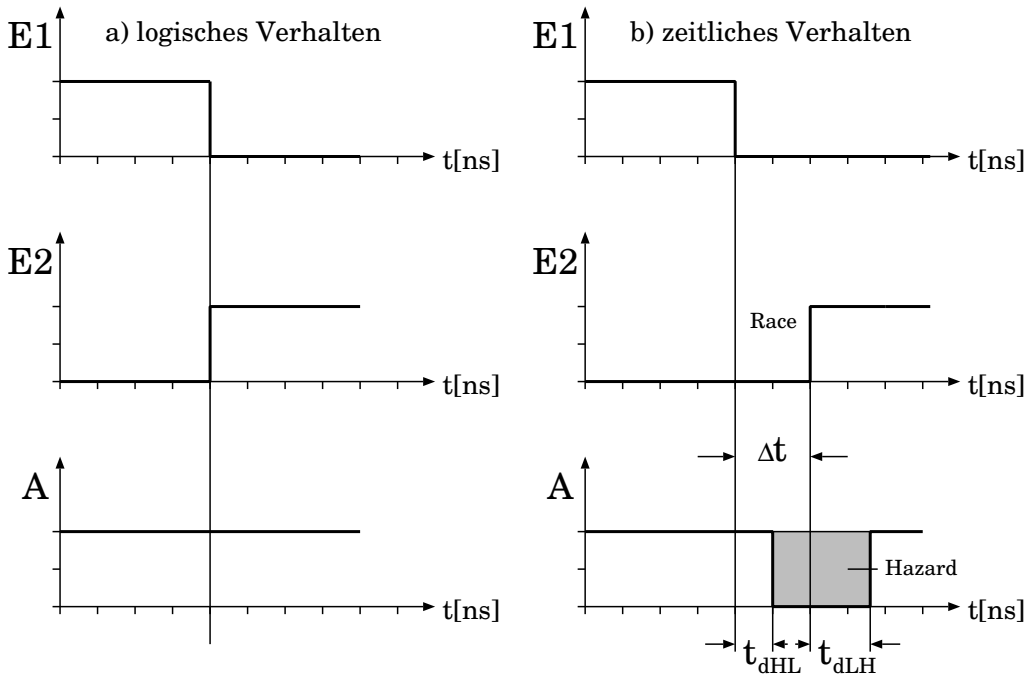


Abb. 2.6: Races und Hazards

Kommt es zu laufzeitbedingten Verzögerungen (**Races**) *der Eingangssignale* (→ Abb. 2.6b) *und ist* $\Delta t > t_d$ *(kein Spike), so reagiert das Gatter durch einen falschen logischen Ausgangszustand, einen sogenannten* **Hazard**. *Der Hazard ist ein Fehler, der schaltungstechnisch behoben werde muß.*

Eine Hilfsmaßnahme gegen laufzeitbedingte Fehler ist die rechtzeitige Synchronisation (Abtaktung) der Schaltung. Durch die Abtaktung wird für den nachfolgenden Schaltungsteil ein neuer eindeutiger und fehlerfreier Ausgangszustand geschaffen. Es kann dann wieder so lange weiter gerechnet werden, bis die aufgelaufenen Verzögerungen zu erneuten kritischen Zeitbedingungen führen. Dann muß eine weitere Synchronisierung, z.B. mit Hilfe von Flip-Flops, vorgenommen werden.

3 Test integrierter Schaltungen

Integrierte Schaltungen nehmen durch eine ansteigende Integrationsdichte an Komplexität immer mehr zu. Der Test dieser Schaltkreise wird daher immer schwieriger. Da ihre Fehlerfreiheit Voraussetzung für die Zuverlässigkeit größerer zusammengesetzter Systeme ist, müssen durch ihren Test möglichst viele potentielle Fehler erkannt werden (es wird ein hoher **Fehlerüberdeckungsgrad** angestrebt). Um die korrekte Funktion mit hoher Sicherheit gewährleisten zu können, werden auf allen Entwurfsstufen wiederholt Tests durchgeführt:

- Bei der **Entwurfsverifikation** wird die Übereinstimmung der Schaltung mit der gegebenen Spezifikation geprüft.
- Der **Herstellungstest** überprüft die fehlerfreie Umsetzung der Logik in eine physikalische Schaltung.
- Beim **Reparaturtest** sollen fehlerhafte Komponenten in einem System erkannt werden.
- Der **Selbsttest** erfolgt während des Betriebs in den jeweiligen Betriebspausen. Dadurch können auch Schäden, die z.B. durch Alterung der Schaltung entstehen, erfaßt werden.

3.1 Fehlerursachen und Fehlerarten

Fehler in integrierten Schaltungen haben unterschiedliche Ursachen, z.B.:

- Fehlstellen im Grundmaterial (statistisch verteilt).
- Fehler bei der Herstellung, z.B.
  - ungleichmäßige Dotierung,
  - ungleichmäßiges Ätzen,
  - Justierungsfehler der Masken,
  - Schäden an den Masken oder
  - Bondfehler.
- Ausfälle nach der Herstellung durch
  - statische Überladung bei CMOS,
  - thermische Überlastung,
  - elektrische Überlastung oder
  - Atommigration.

- Einflüsse der Betriebsumgebung, usw.

Die Fehler lassen sich ihrem Erscheinungsbild nach in zwei Gruppen unterteilen (→ Abb.3.1).

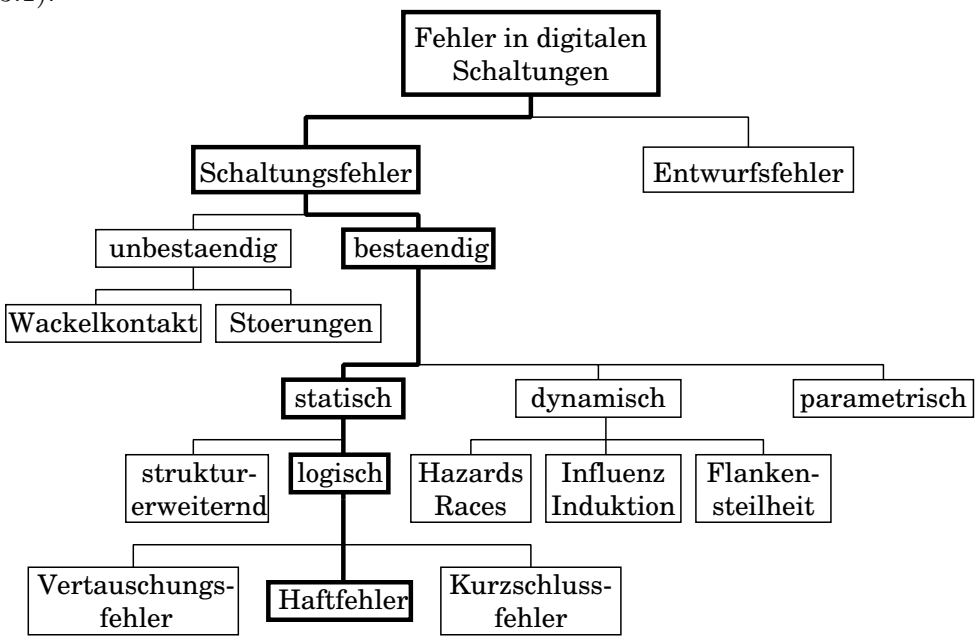


Abb. 3.1: Fehler in digitalen Schaltungen

In die erste Gruppe fallen **Entwurfsfehler**, in die zweite Schaltungs- bzw. **Fertigungsfehler**. Die Erkennung von Entwurfsfehlern ist die Aufgabe der Logiksimulation. Die Schaltungsfehler werden in **ständige** und **unbeständige** unterteilt.

Vorübergehende Fehler entstehen z.B. durch Wackelkontakte, Einbrüche in der Spannungsversorgung oder durch Störungen von außen. Für ihre Erkennung können keine allgemeinen Regeln angegeben werden.

Die ständigen Fehler werden in **statische**, **dynamische** und **parametrische** aufgeteilt.

Die parametrischen Fehler, wie Strom- und Spannungsabweichungen, falsches Tristate-Verhalten, verminderte Treiberstärke (großes kHL bzw. kLH) und überhöhte Leckströme der Ein- und Ausgangstreiber (der Ein- und Ausgangspadzellen) werden durch relativ einfache Strom- und Spannungsmessungen erfaßt.

Dynamische Fehler sind z.B. zu flache Signalflanken, Überschwingen durch Reflexionen, Races, Hazards oder Induktion bzw. Influenz bei paralleler Leitungsführung. Durch zunehmende Betriebsfrequenz und steigende Operationsgeschwindigkeiten treten diese Fehler immer häufiger auf, dynamische Tests auf Einhaltung der Laufzeitbedingungen werden daher immer wichtiger. Um diese Fehler erfassen zu können, wurde z.B. ein **lokales Übergangsmodell** (transition fault model) eingeführt, welches Übergangs- und Verzögerungsfehler an den Ein- und Ausgangssignalen der Schaltungselemente berücksichtigt. Dadurch werden lokale Übergangsfehler, wie z.B. ein gar nicht oder stark verspätet stattfindender Signalwechsel, erkannt. Da auch Verzögerungsfehler, die aus mehreren Signalen bestehen, erfaßt werden, wird das lokale Übergangsmodell auch **Pfadverzögerungsfehlermodell** (path delay fault model) genannt. Mit seiner Hilfe

können diejenigen Fehlerfälle abgedeckt werden, in denen zwar die einzelnen Signale die für einen Signalwechsel vorgegebenen Zeitschranken einhalten, die Summe der Schaltzeiten aller Signale jedoch auf dem zu testenden Pfad größer als die durch die Schaltungsspezifikation und die Betriebsfrequenz festgelegte Gesamtschaltzeit ist.

Zu den statischen zählen die **strukturerweiternden** und die **logischen** Fehler.

Schwer zu überprüfen sind die strukturerweiternden Fehler. Eine Strukturerweiterung entsteht z.B. durch eine kurzschlußbedingte Erhöhung der Anzahl der Ein- und Ausgänge oder durch Kurzschluß entstehende Rückkopplungen. Zur Erkennung dieser Fehler wurde ein **Kurzschlußfehlermodell** (bridging faults) geschaffen. Mit Hilfe dieses Fehlermodells können Überbrückungsfehler nachgebildet werden.

Die logischen Fehler werden nochmals in drei Gruppen unterteilt.

Die ersten beiden Gruppe bilden die Kurzschlußfehler (soweit sie nicht strukturerweiternd sind) und die Vertauschungsfehler (falsches Bauelement).

Die wichtigste Gruppe umfaßt die **Haftfehler** (stuck-at-faults). Bei diesen Fehlern ist der logische Zustand am Fehlerort ständig auf 0 (**stuck-at-0**) bzw. ständig auf 1 (**stuck-at-1**).

Die nachfolgenden Algorithmen zur Testmusterbestimmung sind auf diese Gruppe der ständigen **Einzelfehler** beschränkt. Es hat sich gezeigt, daß viele physikalisch auftretenden Fehler in ihrer Wirkung dieser Fehlergruppe zuzuordnen sind und daher mit den generierten Tests erfaßt werden.

Speziell für die CMOS-Technologie wurden weitere Fehlermodelle, wie z.B. das einfache **stuck-open**- und das **stuck-on**-Fehlermodell, eingeführt. Zusammen mit den oben genannten Fehlermodellen werden die meisten CMOS-spezifischen Fehlermechanismen damit abgedeckt.

3.2 Automatische Testmustergenerierung

Das Hauptproblem bei der Erstellung von Tests für komplexe Schaltungen liegt darin, eine Testmenge zu finden, die kleiner ist als die Menge der möglichen Eingangskombinationen.

Die automatische Testmustergenerierung erstellt Testmuster, die für ein gegebenes Fehlermodell den größtmöglichen Fehlerüberdeckungsgrad (**Fehlersimulation**!),

$$\text{Fehlerüberdeckungsgrad} = \frac{\text{Zahl der erkannten Fehler}}{\text{Zahl der modellierten Fehler}},$$

ergeben. Um dieses Ziel mit wirtschaftlich vertretbarem Aufwand zu erreichen, muß ein optimaler „Tradeoff“ zwischen folgenden konkurrierenden Bedingungen gefunden werden:

- Qualität bzw. Genauigkeit des Fehlermodells,
- Komplexität der zu lösenden Teilprobleme,
- Rechenzeit der für die Lösung der Teilprobleme eingesetzten Algorithmen,

- vollständige Fehlerüberdeckung und
- Anzahl der Tests und damit Kosten der Testdurchführung.

Im folgenden werden Verfahren angegeben, die eine minimale Testmenge für Einzelfehler berechnen. Ein Einzelfehler α,

$$\alpha \in \{\text{stuck-at-0}, \text{stuck-at-1}\},$$

einer Variablen x wird für „stuck-at-0“ mit $x/0$ und für „stuck-at-1“ mit $x/1$ angegeben. Der Fehler α wird nur dann erkannt, wenn das Fehlverhalten an einem Ausgang y der Schaltung beobachtet werden kann (**Beobachtbarkeit** des Fehlers):

$$\boxed{y \oplus y_\alpha = 1} \tag{3.1}$$

Jede Eingangsbelegung $\boldsymbol{x}_i$ (→ Seite 16), die im fehlerhaften Fall einen anderen Ausgangswert als im fehlerfreien liefert (**Einstellbarkeit** des Fehlerortes),

$$\boxed{y(\boldsymbol{x}_i) \oplus y_\alpha(\boldsymbol{x}_i) = 1}, \tag{3.2}$$

ist ein **Testmuster** t_i:

$$t_i := \boldsymbol{x}_i$$

Für alle möglichen Fehler kann eine sogenannte **Fehlerüberdeckungstabelle** aufgestellt werden. Die Anzahl der Zeilen wird durch die Menge der Eingangskombinationen festgelegt, die Anzahl der Spalten entspricht der Menge aller Einzelfehler. Die Minimierung dieser Tabelle erfolgt auf die gleiche Weise, wie die Minimierung der Mengenüberdeckungstabelle (→ Seite 26). Das Ergebnis der Minimierung ist die **minimale Testmenge** $T_{\min}$.

Beispiel 3.1 *Für das UND-Gatter* (→ Abb.3.2) *können 6 verschiedene Einzelfehler* $f_1, \ldots, f_6$ *angegeben werden. Alle Fehler und alle möglichen Testmuster* t_i *sind in der Fehlerüberdeckungstabelle* (→ Tab.3.1) *zusammengefaßt. In der verkürzten Tabelle* (→ Tab.3.2) *sind nur die unterschiedlichen Werte für* y *und* y_α *durch* $\times$ *markiert. Diese Tabelle entspricht einer Mengenüberdeckungstabelle. Aus ihr wird die minimale Testmenge bestimmt.*

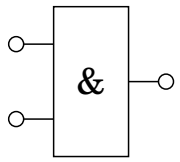


Abb. 3.2: UND-Gatter

t_i	x_2	x_1	y	f_1 $x_1/0$	f_2 $x_1/1$	f_3 $x_2/0$	f_4 $x_2/1$	f_5 $y/0$	f_6 $y/1$
t_0	0	0	0	0	0	0	0	0	1
t_1	0	1	0	0	0	0	1	0	1
t_2	1	0	0	0	1	0	0	0	1
t_3	1	1	1	0	1	0	1	0	1

Tab. 3.1: Fehlerüberdeckungstabelle (FÜT) für das UND-Gatter

Kernzeilen sind t_3, t_2 und t_1. Durch das Streichen von t_3 fallen die Spalten f_1, f_3 und f_5 heraus, durch das Streichen von t_2, f_2 und f_6, durch t_1, f_4. Von den drei Tests werden somit alle Fehler überdeckt. Die minimale Testmenge ist damit $T_{\min} = \{t_3, t_2, t_1\}$.

t_i	f_1	f_2	f_3	f_4	f_5	f_6
t_0						×
t_1				×		×
t_2		×				×
t_3	×		×		×	

Tab. 3.2: Verkürzte (FÜT)

Beispiel 3.2

Zu der Schaltfunktion

$$y(\boldsymbol{x}) = x_3x_2 \vee x_3x_1 \vee x_2x_1$$

gehört das in Abb.3.3 *dargestellte Schaltnetz. Die Fehlerüberdeckungstabelle* (→ Tab.3.3) *enthält alle möglichen Einfachfehler. In der verkürzten Tabelle* (→ Tab.3.4) *sind wiederum nur die Stellen markiert, für die sich y und y_α unterscheiden.*

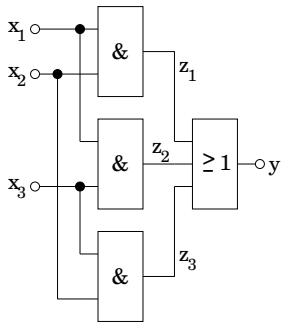


Abb. 3.3: Schaltbild

		f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}
t_i	y	$x_1/0$	$x_2/0$	$x_3/0$	$x_1/1$	$x_2/1$	$x_3/1$	$z_1/0$	$z_2/0$	$z_3/0$	$z_1/1$	$z_2/1$	$z_3/1$	$y/0$	$y/1$
0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	1
1	0	0	0	0	0	1	1	0	0	0	1	1	1	0	1
2	0	0	0	0	1	0	1	0	0	0	1	1	1	0	1
3	1	0	0	1	1	1	1	0	1	1	1	1	1	0	1
4	0	0	0	0	1	1	0	0	0	0	1	1	1	0	1
5	1	0	1	0	1	1	1	1	0	1	1	1	1	0	1
6	1	1	0	0	1	1	1	1	1	0	1	1	1	0	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1

Tab. 3.3: Fehlerüberdeckungstabelle

t_i	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}
0										×	×	×		×
1					×	×				×	×	×		×
2				×		×				×	×	×		×
3	×	×					×						×	
4				×	×					×	×	×		×
5	×		×					×					×	
6		×	×						×				×	
7													×	

Tab. 3.4: Verkürzte Fehlerüberdeckungstabelle

Kernzeilen und somit Kerntests sind t_3, t_5 und t_6. Von ihrem Streichen sind die Fehler $\{f_1, f_2, f_7, f_{13}\}$, $\{f_3, f_8\}$ und $\{f_9\}$ betroffen. Die gleichen Spalten $f_{14}, f_{12}, f_{11}, f_{10}$ werden bis auf f_{10} gestrichen; da f_{10} zu f_4, f_5, f_6 dominant ist, kann die Spalte f_{10} ebenfalls entfallen. Die verbleibenden Fehler f_4, f_5, f_6 werden durch $\{t_1, t_2\}$, $\{t_1, t_4\}$ oder $\{t_2, t_4\}$ getestet. Es sind folglich drei minimale Testmengen möglich: $T_{\min 1} = \{t_1, t_2, t_3, t_5, t_6\}$, $T_{\min 2} = \{t_1, t_4, t_3, t_5, t_6\}$, $T_{\min 3} = \{t_2, t_4, t_3, t_5, t_6\}$.

Enthalten Schaltungen Redundanzen, so werden nicht alle Fehler erkannt und die maximal mögliche Fehlerüberdeckung nicht erreicht. Eine zu geringe Fehlerüberdeckung kann daher nicht nur die Folge zu weniger oder falscher Testmuster sondern auch die Folge von Redundanzen sein. Eine Unterscheidung der Ursachen ist in der Regel nicht möglich.

In Abb.3.4 ist ein Beispiel für ein redundantes Schaltnetz gegeben. Ein angenommener Fehler in x_2 kann durch kein Testmuster erkannt werden, da y nicht von x_2 abhängt. Bei der Fehlersimulation wird dennoch ein Fehler in x_2 angenommen und mit den vorher durch die automatische Testmustergenerierung erzeugten Testmustern geprüft. Da die Anzahl der modellierten Fehler durch $x_2/0$ und $x_2/1$ erhöht wird, sinkt der Fehlerüberdeckungsgrad (→ Seite 53).

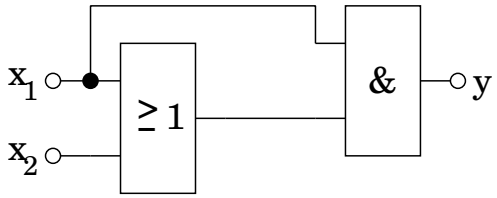


Abb. 3.4: Redundantes Schaltnetz: $y = x_1(x_1 \vee x_2) = x_1$

In der **Testbarkeitsanalyse** werden Maßzahlen für die Einstellbarkeit und Beobachtbarkeit der einzelnen Schaltungssignale einer gegebenen Schaltung ermittelt. Diese Maßzahlen werden als **Testbarkeitsmaße** bezeichnet. Die ermittelten Werte liefern Anhaltspunkte für testbarkeitsverbessernde Maßnahmen. Alle Maßnahmen, die eine Verbesserung der Einstell- und Beobachtbarkeit zur Folge haben, machen einen Entwurf testfreundlicher. Erfolgt ein Entwurf unter dem Aspekt guter Testbarkeit, so wird diese Vorgehensweise **Design for Testability** (DFT) genannt.

3.2.1 Methode der Booleschen Differenzen

Die automatische Testmustergenerierung mit Hilfe des Verfahrens der Booleschen Differenzen basiert auf der zu realisierenden Schaltfunktion. Die Boolesche Differenz wird durch

$$\boxed{\begin{array}{rcl} y_{x_j} & = & y(x_1, \ldots, x_{j-1}, 1, x_{j+1}, \ldots, x_n) \oplus y(x_1, \ldots, x_{j-1}, 0, x_{j+1}, \ldots, x_n) \\ & = & y(x_1, \ldots, x_{j-1}, x_j, x_{j+1}, \ldots, x_n) \oplus y(x_1, \ldots, x_{j-1}, \bar{x}_j, x_{j+1}, \ldots, x_n) \\ & = & y(\boldsymbol{x}) \oplus y(x_1, \ldots, x_{j-1}, \bar{x}_j, x_{j+1}, \ldots, x_n) \end{array}} \quad (3.3)$$

definiert; x_j ist eine beliebige Variable der Schaltfunktion. Alle $\boldsymbol{x}$, die der Bedingung

$$\begin{array}{rcl} y(x_1, \ldots, x_{j-1}, 0, x_{j+1}, \ldots, x_n) \oplus y(x_1, \ldots, x_j, \ldots, x_n) & = & 1, \text{ bzw.} \\ y(x_1, \ldots, x_{j-1}, 0, x_{j+1}, \ldots, x_n) \oplus y(\boldsymbol{x}) & = & 1 \end{array}$$

genügen, sind Tests für $x_j/0$ ($\rightarrow$ (3.2)). Diese Beziehung kann mit (1.18) umgeformt werden. Wendet man auf $y(\boldsymbol{x})$ (1.18) an,

$$y(\boldsymbol{x}) = x_j y(x_1, \ldots, 1, \ldots, x_n) \oplus \bar{x}_j y(x_1, \ldots, 0, \ldots, x_n),$$

so folgt mit (1.19b)

$$\begin{aligned} y(x_1, \ldots, 0, \ldots, x_n) \oplus x_j y(x_1, \ldots, 1, \ldots, x_n) \oplus \bar{x}_j y(x_1, \ldots, 0, \ldots, x_n) &= 1 \\ (1 \oplus \bar{x}_j)\,[y(x_1, \ldots, 0, \ldots, x_n)] \oplus x_j y(x_1, \ldots, 1, \ldots, x_n) &= 1 \\ x_j\,[y(x_1, \ldots, 0, \ldots, x_n) \oplus y(x_1, \ldots, 1, \ldots, x_n)] &= 1 \\ x_j y_{x_j} &= 1. \end{aligned}$$

Die Tests für $x_j/1$ ergeben sich entsprechend aus

$$y(x_1, \ldots, x_{j-1}, 1, x_{j+1}, \ldots, x_n) \oplus y(x_1, \ldots, x_j, \ldots, x_n) = 1$$

zu

$$\bar{x}_j y_{x_j} = 1.$$

Für einen Test t_i auf den Fehler $x_j/0$ muß folglich $x_j = 1$ eingestellt werden (Einstellbarkeit) und die Schaltungsantworten der fehlerbehafteten ($y(x_1, \ldots, 0, \ldots, x_n)$) und der fehlerfreien ($y(x_1, \ldots, 1, \ldots, x_n)$) Schaltung unterschiedlich sein (Beobachtbarkeit: $y_{x_j} = 1$). Für einen Fehler $x_j/1$ gilt entsprechend $x_j = 0$ und $y_{x_j} = 1$.

$$\boxed{\begin{array}{lll} x_j/0: & x_j y_{x_j} & = 1 \\ x_j/1: & \bar{x}_j y_{x_j} & = 1 \end{array}} \tag{3.4}$$

In (3.5) sind wichtigsten Rechenregeln für Boolesche Differenzen zusammengefaßt.

$$\boxed{\begin{array}{ll} \text{(a)} & y_{x_j} = 0 \quad \text{falls } y \neq f(x_j) \\ \text{(b)} & y_y = 1 \\ \text{(c)} & y_{x_j} = \bar{y}_{x_j} \\ \text{(d)} & [z(\boldsymbol{x}) \oplus w(\boldsymbol{x})]_{x_j} = z_{x_j} \oplus w_{x_j} \\ \text{(e)} & [z(\boldsymbol{x}) w(\boldsymbol{x})]_{x_j} = z(\boldsymbol{x}) w_{x_j} \oplus z_{x_j} w(\boldsymbol{x}) \oplus z_{x_j} w_{x_j} \\ \text{(f)} & [z(\boldsymbol{x}) \vee w(\boldsymbol{x})]_{x_j} = \bar{z}(\boldsymbol{x}) w_{x_j} \oplus z_{x_j} \bar{w}(\boldsymbol{x}) \oplus z_{x_j} w_{x_j} \\ \text{(g)} & y_{x_j} = y_z z_{x_j} \quad \text{für } y(\boldsymbol{x}) = f(z(\boldsymbol{x})) \quad \text{(Kettenregel)} \end{array}} \tag{3.5}$$

Hinweise zu den Rechenregeln:

(3.5a) Da y nicht von x_j abhängt, gilt

$$y_{x_j} = y \oplus y = 0.$$

(3.5b) Setzt man y einmal auf 0 und einmal auf 1, folgt

$$y_y = \bar{y} \oplus y = 1.$$

(3.5c) Für $\bar{y}_{x_j}$ folgt mit (1.19b) und (1.19c)

$$\begin{aligned} \bar{y}_{x_j} &= \bar{y}(\boldsymbol{x}) \oplus \bar{y}(x_1, \ldots, \bar{x}_j, \ldots, x_n) \\ &= y(\boldsymbol{x}) \oplus 1 \oplus y(x_1, \ldots, \bar{x}_j, \ldots, x_n) \oplus 1 \\ &= y_{x_j}. \end{aligned}$$

(3.5d) Liegt die Funktion als Ringsumme vor, so ist die Berechnung der Booleschen Differenz mit der partiellen Differentiation nach einer Variablen x_j vergleichbar:

$$\begin{aligned}[z(\boldsymbol{x}) \oplus w(\boldsymbol{x})]_{x_j} &= z(\boldsymbol{x}) \oplus w(\boldsymbol{x}) \oplus z(x_1, \ldots, \bar{x}_j, \ldots, x_n) \oplus w(x_1, \ldots, \bar{x}_j, \ldots, x_n) \\ &= z_{x_j} \oplus w_{x_j}\end{aligned}$$

(3.5e) Für $[z(\boldsymbol{x})w(\boldsymbol{x})]_{x_j}$ folgt

$$[z(\boldsymbol{x})w(\boldsymbol{x})]_{x_j} = [z(\boldsymbol{x})w(\boldsymbol{x})] \oplus z(x_1, \ldots, \bar{x}_j, \ldots, x_n)w(x_1, \ldots, \bar{x}_j, \ldots, x_n).$$

Wegen

$$\begin{aligned}y_{x_j} \oplus y(\boldsymbol{x}) &= y(\boldsymbol{x}) \oplus y(x_1, \ldots, \bar{x}_j, \ldots, x_n) \oplus y(\boldsymbol{x}) \\ &= \underbrace{y(\boldsymbol{x}) \oplus y(\boldsymbol{x})}_{0} \oplus y(x_1, \ldots, \bar{x}_j, \ldots, x_n) \\ &= y(x_1, \ldots, \bar{x}_j, \ldots, x_n)\end{aligned}$$

folgt weiter

$$\begin{aligned}[z(\boldsymbol{x})w(\boldsymbol{x})]_{x_j} &= [z(\boldsymbol{x})w(\boldsymbol{x})] \oplus [z_{x_j} \oplus z(\boldsymbol{x})][w_{x_j} \oplus w(\boldsymbol{x})] \\ &= z_{x_j}w_{x_j} \oplus z_{x_j}w(\boldsymbol{x}) \oplus w_{x_j}z(\boldsymbol{x})\end{aligned}$$

(3.5f) Die disjunktive Darstellung kann mit (1.19h) in eine Ringsumme umgerechnet werden:

$$[z(\boldsymbol{x}) \vee w(\boldsymbol{x})]_{x_j} = [z(\boldsymbol{x}) \oplus w(\boldsymbol{x}) \oplus z(\boldsymbol{x})w(\boldsymbol{x})]_{x_j}$$

Für die Ringsumme folgt mit (3.5d)

$$[z(\boldsymbol{x}) \vee w(\boldsymbol{x})]_{x_j} = z_{x_j} \oplus w_{x_j} \oplus [z(\boldsymbol{x})w(\boldsymbol{x})]_{x_j}$$

und mit (3.5f)

$$\begin{aligned}[z(\boldsymbol{x}) \vee w(\boldsymbol{x})]_{x_j} &= z_{x_j} \oplus w_{x_j} \oplus w_{x_j}z(\boldsymbol{x}) \oplus z_{x_j}w(\boldsymbol{x}) \oplus z_{x_j}w_{x_j} \\ &= z_{x_j}[1 \oplus w(\boldsymbol{x})] \oplus w_{x_j}[1 \oplus z(\boldsymbol{x})] \oplus z_{x_j}w_{x_j} \\ &= z_{x_j}\bar{w}(\boldsymbol{x}) \oplus w_{x_j}\bar{z}(\boldsymbol{x}) \oplus z_{x_j}w_{x_j}\end{aligned}$$

(3.5g) Die Kettenregel erklärt sich aus der Definition der Booleschen Differenz. Die Boolesche Differenz gibt die Empfindlichkeit eines Ausgangs y entlang eines Pfandes zurück zum Eingang wieder. Ist z eine Zwischenvariable entlang dieses Pfades, so muß die Empfindlichkeit von y bezüglich dieser Variablen ($z \to y$) und die Empfindlichkeit von z bezüglich einer davor liegenden Variablen ($x_j \to z$) gegeben sein, wenn der gesamte Pfand $x_j \to z \to y$ sensibilisiert werden soll.

Beispiel 3.3 *Für die in* Abb.3.3 *dargestellte Schaltfunktion* (→ Seite 23)

$$y(\boldsymbol{x}) = \underbrace{x_3x_2}_{z_3} \vee \underbrace{x_3x_1}_{z_2} \vee \underbrace{x_2x_1}_{z_1} = z_3 \vee z_2 \vee z_1 = x_3x_2 \oplus x_3x_1 \oplus x_2x_1$$

können 14 Einfachfehler angegeben werden.

Für die Fehler $x_1/0$, $x_1/1$, $x_2/0$, $y/0$ *und* $z_1/0$ *werden alle Testmuster mit Hilfe der Booleschen Differenzen ermittelt. Für einen Zwischenwert*

$$z = x_j x_k$$

gilt allgemein

$$z_{x_j} = (0 \wedge x_k) \oplus (1 \wedge x_k) = 0 \oplus x_k = x_k, \quad \textit{bzw.} \quad z_{x_k} = x_j.$$

Da die Schaltfunktion als Ringsumme vorliegt, kann (3.5d) *und anschließend* (1.17) *angewendet werden:*

$$\begin{aligned}
y_{x_1} &= x_2 \oplus x_3 = \bar{x}_2 x_3 \vee x_2 \bar{x}_3 \\
y_{x_2} &= x_1 \oplus x_3 = \bar{x}_1 x_3 \vee x_1 \bar{x}_3 \\
y_y &= 1 \\
y_{z_1} &= (z_2 \vee z_3) \oplus 1 \quad \textit{mit } (1.19\text{b}) \textit{ folgt} \\
&= (\overline{z_2 \vee z_3}) = (\overline{x_3 x_1 \vee x_3 x_2}) \quad \textit{und mit } (1.10\text{h}) \\
&= (\bar{x}_3 \vee \bar{x}_1)(\bar{x}_3 \vee \bar{x}_2) = \bar{x}_3 \vee \bar{x}_2 \bar{x}_1
\end{aligned}$$

Das gleiche Ergebnis erhält man durch wiederholte Anwendung von (3.3)*:*

$$\begin{aligned}
y_{x_1} &= (x_3 x_2 \vee x_3 \vee x_2) \oplus (x_3 x_2) \quad \textit{mit } (1.10\text{c}) \textit{ folgt} \\
&= (x_3 \vee x_2) \oplus (x_3 x_2) \quad \textit{und mit } (1.19\text{h}) \\
&= x_3 \oplus x_2 \oplus x_3 x_2 \oplus x_3 x_2 \quad \textit{bzw. mit } (1.19\text{c}) \\
&= x_3 \oplus x_2 \quad \textit{und } (1.17) \\
&= \bar{x}_2 x_3 \vee x_2 \bar{x}_3 \\
y_{x_2} &= (x_3 \vee x_3 x_1 \vee x_1) \oplus (x_3 x_1) \\
&= (x_3 \vee x_1) \oplus x_3 x_1 \\
&= x_3 \oplus x_1 = \bar{x}_1 x_3 \vee x_1 \bar{x}_3 \\
y_y &= 1 \\
y_{z_1} &= (z_3 \vee z_2 \vee 1) \oplus (z_3 \vee z_2) \\
&= 1 \oplus (z_3 \vee z_2) \\
&= (\overline{z_3 \vee z_2}) = \bar{z}_3 \bar{z}_2 = \bar{x}_3 \vee \bar{x}_2 \bar{x}_1
\end{aligned}$$

(Vorsicht: $z_3 \vee z_2 \vee z_1 \neq z_3 \oplus z_2 \oplus z_1$*, dies gilt nur für die KDNF!) Aus den Booleschen Differenzen folgen die Testmuster zu*

$$\begin{aligned}
x_1/0\colon \quad x_1 y_{x_1} &= 1 = \underbrace{\bar{x}_3 x_2 x_1}_{t_3} \vee \underbrace{x_3 \bar{x}_2 x_1}_{t_5} \\
x_1/1\colon \quad \bar{x}_1 y_{x_1} &= 1 = \underbrace{\bar{x}_3 x_2 \bar{x}_1}_{t_2} \vee \underbrace{x_3 \bar{x}_2 \bar{x}_1}_{t_4} \\
x_2/0\colon \quad x_2 y_{x_2} &= 1 = \underbrace{\bar{x}_3 x_2 x_1}_{t_3} \vee \underbrace{x_3 x_2 \bar{x}_1}_{t_6} \\
y/0\colon \quad y y_y &= 1 = \underbrace{x_2 x_1}_{t_3, t_7} \vee \underbrace{x_3 x_1}_{t_5, t_7} \vee \underbrace{x_3 x_2}_{t_6, t_7} \\
z_1/0\colon \quad z_1 y_{z_1} &= 1 = \underbrace{\bar{x}_3 x_2 x_1}_{t_3} \vee \underbrace{\bar{x}_2 \bar{x}_1 x_2 x_1}_{0}.
\end{aligned}$$

Die ermittelten Testmuster stimmen mit den Einträgen in Tab.3.4 *überein. Nach Berechnung aller Tests kann mit Hilfe der Fehlerüberdeckungstabelle die minimale Testmenge bestimmt werden.*

Bei Schaltnetzen mit **Baumstruktur** (→ Abb.3.5) werden durch das Testen aller Eingangsvariablen alle Zwischenvariable mitgetestet, da nur ein Pfad von einer Eingangs- zu einer Ausgangsvariablen existiert.

Bei einem **vermaschten Schaltnetz** (→ Abb.3.6) sind mehrere Pfade von einer Eingangs- zu einer Ausgangsvariablen möglich; Zwischenvariable werden daher durch das Testen von Eingangsvariablen nicht immer mitgetestet.

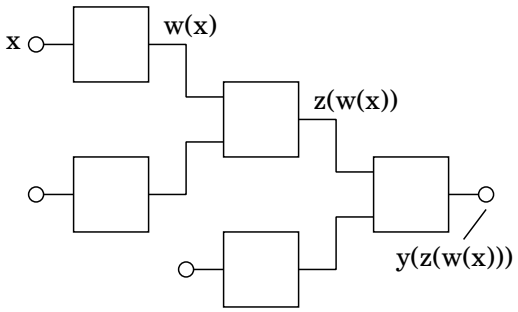


Abb. 3.5: Schaltnetz mit Baumstruktur

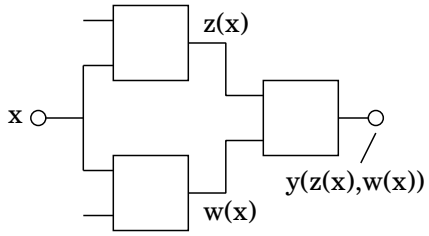


Abb. 3.6: Vermaschtes Schaltnetz

Ein Problem, das durch die Vermaschung entstehen kann, ist die **Selbstmaskierung** (→ Abb. 3.7).

Beispiel 3.4 *In* Abb.3.7 *ist ein Schaltnetz mit* **Rekonvergenzmasche** *dargestellt.*

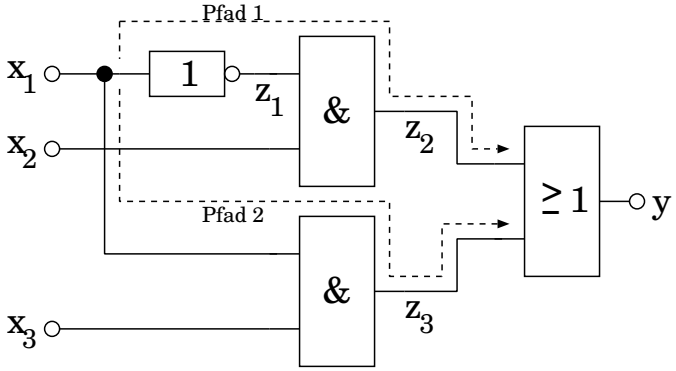


Abb. 3.7: Selbstmaskierung nach einer Rekonvergenzmasche

Es gilt

$$z_2 = \bar{x}_1 x_2 \quad \textit{und}$$
$$z_3 = x_1 x_3.$$

Für $x_3 = x_2 = 1$ *sind beide Pfade fehlerleitend. Dies führt zu einer Selbstmaskierung des Signals* x_1 *nach der Rekonvergenz am ODER-Gatter. Es folgt*

$$z_2 = \bar{x}_1, \quad z_3 = x_1 \quad \textit{und damit} \quad y = \underbrace{\bar{x}_1 \vee x_1}_{\textit{Selbstmaskierung}} = 1.$$

Ein Fehler in x_1 kann so nicht getestet werden, da y unabhängig von x_1 den Wert 1 *hat. Für $x_2 = 0$ wird die Fehlerleitung über den Pfad 1 und damit die Selbstmaskierung aufgehoben. Es gilt dann:*

$$z_2 = 0, \quad z_3 = x_3 x_1, \quad y = x_3 x_1$$

Die Variable x_1 ist jetzt über den Pfad 2 testbar.

Die Selbstmaskierung erfolgt immer dann, wenn ein Pfad der Rekonvergenzmasche (hier ist es der Pfad 1, $x_1 \to z_1 \to z_2$) eine ungerade Anzahl von Negationen enthält.

3.2.2 Der D-Algorithmus

Der D-Algorithmus ist das zur Zeit am häufigsten verwendete Verfahren zur rechnergestützten Testmustergenerierung. Mit dem D-Algorithmus können Tests für alle testbaren Fehler in einer kombinatorischen Schaltung erzeugt werden. Wird D als Übergang von 0 nach 1 und $\bar{D}$ dual als Übergang von 1 nach 0 definiert,

$$D \stackrel{\text{def}}{=} 0 \to 1; \quad \bar{D} \stackrel{\text{def}}{=} 1 \to 0, \tag{3.6}$$

so können für ein gegebenes Gatter die D-**Implikanten** (fehlerleitende D-Belegungen) durch Intersektion von Zeilen der Wahrheitstabelle erzeugt werden. In (3.6) liegt die Betonung auf dem Übergang, nicht auf den Werten. D ist eine quasi dynamische Belegung einer Variablen x. Wird D an einem Fehlerort x angenommen und existiert ein D-Pfad (**Fehlerleitung**) vom Fehlerort zu einem Ausgang y, so ist eine Änderung (ein Fehler) in x am Ausgang y beobachtbar.

Beispiel 3.5

Aus der Wahrheitstabelle des UND-Gatters (→ Tab.3.5) *können mit* (3.6) *alle fehlerleitenden Belegungen durch Intersektion der Zeilen erzeugt werden. Wird z.B. der Übergang von Zeile* 0 *nach* 1 *betrachtet, so folgt mit* $(i_2 i_1, y(i_1, i_2))$ (→ Seite 15)

$$(00, 0) \to (01, 0) = (0D, 0).$$

i	x_2	x_1	y
0	0	0	0
1	0	1	0
2	1	0	0
3	1	1	1

Tab. 3.5

Durch die Intersektion aller Zeilen (→ Tab.3.6) *erhält man alle fehlerleitenden D-Belegungen* (→ Tab.3.7).

	x_2	x_1	y
0→1	0	D	0
0→2	D	0	0
0→3	D	D	D
1→0	0	$\bar{D}$	0
1→2	D	$\bar{D}$	0
1→3	D	1	D

	x_2	x_1	y
2→1	$\bar{D}$	D	0
2→0	$\bar{D}$	0	0
2→3	1	D	D
3→2	1	$\bar{D}$	$\bar{D}$
3→1	$\bar{D}$	1	$\bar{D}$
3→0	$\bar{D}$	$\bar{D}$	$\bar{D}$

Tab. 3.6: D-Wertetab. UND-Gatter

x_2	x_1	y
D	D	D
D	1	D
1	D	D
$\bar{D}$	$\bar{D}$	$\bar{D}$
$\bar{D}$	1	$\bar{D}$
1	$\bar{D}$	$\bar{D}$

Tab. 3.7: Fehlerleit. D-Beleg.

Tabelle 3.8 faßt die erweiterten Wertetabellen aller wichtiger Gatter zusammen. Neben 1,0,D und $\bar{D}$ kommt in Tab.3.8 als weiterer Wert X vor. X steht für einen undefinierten bzw. nicht relevanten Wert. X erhöht damit den Freiheitsgrad einer Variablen (sie kann 0 oder 1 sein). Da der D-Algorithmus mit einer 5-wertigen Logik arbeitet, stellt er keine Boolesche Algebra dar (→ Seite 11)!

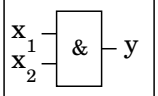
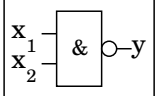
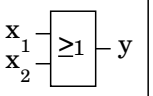
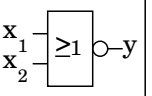
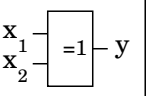
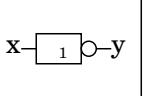
Bestimmung von Tests:

a) Beginnend an der Testvariablen mit D werden alle D-Ketten, die zum Testpunkt y führen, in eine Tabelle (**Sensibilisierungsbelegung** und **Fehlerpfad**) eingetragen.

b) Für jede D-Kette werden die **Belegungsimplikationen** bestimmt und die Konsistenz zu a) geprüft. Bei der Konsistenzprüfung darf nur X auf einen anderen Wert als X treffen.

c) Es erfolgt die Fehlerbelegung an der Testvariablen sowie die **Fehlerbelegungsfortpflanzung**.

d) Aus der Belegung der Eingangsvariablen wird der Test für die gegebene Fehlerbelegung abgelesen. Für X wird dazu 0 und 1 eingesetzt.

AND		NAND		OR		NOR		EXOR		NOT	
											
x_2 x_1	y	x_2 x_1	y	x_2 x_1	y	x_2 x_1	y	x_2 x_1	y	x	y
0 0	0	0 0	1	0 0	0	0 0	1	0 0	0	0	1
0 1	0	0 1	1	0 1	1	0 1	0	0 1	1	1	0
0 $\overline{D}$	0	0 $\overline{D}$	1	0 $\overline{D}$	$\overline{D}$	0 $\overline{D}$	D	0 $\overline{D}$	$\overline{D}$	$\overline{D}$	D
0 D	0	0 D	1	0 D	D	0 D	$\overline{D}$	0 D	D	D	$\overline{D}$
0 X	0	0 X	1	0 X	X	0 X	X	0 X	X	X	X
1 0	0	1 0	1	1 0	1	1 0	0	1 0	1		
1 1	1	1 1	0	1 1	1	1 1	0	1 1	0		
1 $\overline{D}$	$\overline{D}$	1 $\overline{D}$	D	1 $\overline{D}$	1	1 $\overline{D}$	0	1 $\overline{D}$	D		
1 D	D	1 D	$\overline{D}$	1 D	1	1 D	0	1 D	$\overline{D}$		
1 X	X	1 X	X	1 X	1	1 X	0	1 X	X		
$\overline{D}$ 0	0	$\overline{D}$ 0	1	$\overline{D}$ 0	$\overline{D}$	$\overline{D}$ 0	D	$\overline{D}$ 0	$\overline{D}$		
$\overline{D}$ 1	$\overline{D}$	$\overline{D}$ 1	D	$\overline{D}$ 1	1	$\overline{D}$ 1	0	$\overline{D}$ 1	D		
$\overline{D}$ $\overline{D}$	$\overline{D}$	$\overline{D}$ $\overline{D}$	D	$\overline{D}$ $\overline{D}$	$\overline{D}$	$\overline{D}$ $\overline{D}$	D	$\overline{D}$ $\overline{D}$	0		
$\overline{D}$ D	0	$\overline{D}$ D	1	$\overline{D}$ D	1	$\overline{D}$ D	0	$\overline{D}$ D	1		
$\overline{D}$ X	X	$\overline{D}$ X	X	$\overline{D}$ X	X	$\overline{D}$ X	X	$\overline{D}$ X	X		
D 0	0	D 0	1	D 0	D	D 0	$\overline{D}$	D 0	D		
D 1	D	D 1	$\overline{D}$	D 1	1	D 1	0	D 1	$\overline{D}$		
D $\overline{D}$	0	D $\overline{D}$	1	D $\overline{D}$	1	D $\overline{D}$	0	D $\overline{D}$	1		
D D	D	D D	$\overline{D}$	D D	D	D D	$\overline{D}$	D D	0		
D X	X	D X	X	D X	X	D X	X	D X	X		
X 0	0	X 0	1	X 0	X	X 0	X	X 0	X		
X 1	X	X 1	X	X 1	1	X 1	0	X 1	X		
X $\overline{D}$	X	X $\overline{D}$	X	X $\overline{D}$	X	X $\overline{D}$	X	X $\overline{D}$	X		
X D	X	X D	X	X D	X	X D	X	X D	X		
X X	X	X X	X	X X	X	X X	X	X X	X		

Tab. 3.8: Tab. zum D-Algorithmus

Beispiel 3.6 *Für die Variablen x_1 und z_3 der in* Abb.3.3 *dargestellten Schaltung werden alle Testmuster mit Hilfe des D-Algorithmus berechnet. Eine Inkonsistenz wird durch* $\mathcal{I}$(Zeile1,Zeile2,Variable1,...) *gekennzeichnet. Inkonsistenz tritt immer dann auf, wenn sich die Sensibilisierungsbelegung und die Belegungsimplikation widersprechen.*

Für die Variable x_1 sind drei Fehlerpfade zum Ausgang y möglich. Für jeden dieser Pfade werden die Testmuster bestimmt.

Der erste Pfad ist $x_1 \rightarrow z_1 \rightarrow y$ ($\rightarrow$ Tab.3.9). *Die Variable x_1 wird im ersten Schritt auf D gesetzt. Die Fehlerleitung über das UND-Gatter erfolgt mit $x_2 = 1$* ($\rightarrow$ Tab.3.8). *Die Zwischenvariable z_1 erhält damit den Wert D. D wird weitergeleitet, wenn z_2 und z_3 auf Null* ($\rightarrow$ Tab.3.8) *gesetzt werden (2 Zeile).*

Mit den Belegungen aus den Zeilen 1 und 2 ist der Fehlerpfad vom Fehlerort x_1 zum Ausgang y sensibilisiert.

Um $z_2 = 0$ zu erzeugen, muß eine der Variablen x_1 oder x_3 den Wert Null haben, der Wert der anderen ist dabei gleichgültig (Zeilen 3 und 4). Für $z_3 = 0$ gilt das gleiche bezüglich der Variablen x_3 und x_2 (Zeilen 5 und 6). Die Zeilen 3-6 geben die Belegungen der Variablen an, die durch die Werte von z_2 und z_3 in den Zeilen 1 und 2 impliziert (Belegungsimplikationen) werden.

In Zeile 4 wird der Wert von x_1 auf Null festgelegt. Dies widerspricht der Sensibilisierungsbelegung in Zeile 1, Zeile 4 wird daher gestrichen.

Das gleiche gilt für Zeile 6 und die Variable x_2. Für die Sensibilisierungsbelegung des Fehlerpfades muß x_2 den Wert 1 haben, dies widerspricht der Belegung von x_2 in Zeile 6. Zeile 6 wird daher ebenfalls gestrichen.

In Zeile 7 sind die Zeilen 1-6 zusammengefaßt. Die Werte aus den Zeilen 1-2 legen dabei gegebenenfalls die X in den Zeilen 3-6 fest.

Um $x_1/0$ zu testen, wird $x_1 = 1$ gesetzt (Zeile 8). Aus den Zeilen 7 und 8 folgt Zeile 9 und damit der Test t_3 (011).

Für $x_1/1$ wird genauso verfahren.

Testvariable x_1, D-Kette 1: $x_1 \rightarrow z_1 \rightarrow y$								
Schritt	x_3	x_2	x_1	z_3	z_2	z_1	y	Hinweise
1 a		1	D			D		
2 a				0	0	D	D	
3 b	0		X		0			
4 b	X		0		0			$\mathcal{I}(1,4,x_1)$
5 b	0	X		0				
6 b	X	0		0				$\mathcal{I}(1,6,x_2)$
7	0	1	D					
8 c			1					$x_1/0$
9 d	0	1	1					t_3
10 c			0					$x_1/1$
11 d	0	1	0					t_2

Tab. 3.9: Tests: $T_{x_1/0} = \{t_3\}$, $T_{x_1/1} = \{t_2\}$

Der zweite Pfad ist $x_1 \rightarrow z_2 \rightarrow y$ (→ Tab.3.10). *Die Testbestimmung erfolgt wie bei dem Pfad* $x_1 \rightarrow z_1 \rightarrow y$.

Testvariable x_1, D-Kette 2: $x_1 \rightarrow z_2 \rightarrow y$								
Schritt	x_3	x_2	x_1	z_3	z_2	z_1	y	Hinweise
1 a	1		D		D			
2 a				0	D	0	D	
3 b		0	X			0		
4 b		X	0			0		$\mathcal{I}(1,4,x_1)$
5 b	0	X		0				$\mathcal{I}(1,5,x_3)$
6 b	X	0		0				
7	1	0	D					
8 c			1					$x_1/0$
9 d	1	0	1					t_5
10 c			0					$x_1/1$
11 d	1	0	0					t_4

Tab. 3.10: Tests: $T_{x_1/0} = \{t_5\}$, $T_{x_1/1} = \{t_4\}$

Durch die gleichzeitige Fehlerleitung über die Pfade 1 und 2 entsteht der dritte Pfad (→ Tab.3.11). *Hier führen alle Belegungsimplikationen zu einem Widerspruch zu den notwendigen Sensibilisierungsbelegungen. Über diesen Pfad können daher keine Tests generiert werden.*

Testvariable x_1, D-Kette 3: D-Kette 1 $\wedge$ D-Kette 2								
Schritt	x_3	x_2	x_1	z_3	z_2	z_1	y	Hinweise
1 a	1	1	D		D	D		
2 a				0	D	D	D	
3 b	0	X		0				$\mathcal{I}(1,3,x_3)$
4 b	X	0		0				$\mathcal{I}(1,4,x_2)$
Kein Test möglich								

Tab. 3.11: Keine Tests generierbar

Für die Zwischenvariable z_3 *existiert nur ein Pfad zum Ausgang* (→ Tab.3.12). *Hier führt die Zusammenfassung der Sensibilisierungsbelegungen und der Belegungsimplikationen zu Sensibilisierungsalternativen (Zeilen 6 und 7).*

Um $z_3/0$ *zu testen, wird* z_3 *durch* $x_3 = 1$ *und* $x_2 = 1$ *auf* 1 *gesetzt (Zeile 8). Der Wert von* x_1 *ist für* $z_3 = 1$ *irrelevant (Zeile 9).*

Die Zeilen 6 und 9 führen auf den einzigen Test t_6, *da die Zeilen 7 und 9 in* x_3 *und* x_2 *zu einem Widerspruch führen. Um die restlichen Tests für* $z_3/1$ *zu erhalten, muß* z_3 *auf Null gesetzt (Zeilen 12 und 13) werden. Da* z_3 *nicht von* x_1 *abhängt, erhält man die Fehlerbelegungsalternativen in den Zeilen 14 und 15. Anschließend sind wiederum alle Kombinationen der Zeilen 6, 7 und 14, 15 zu erzeugen (Zeilen 16-19). Eventuell widersprüchliche Kombinationen werden dabei gestrichen (hier kommen keine vor).*

Für x_1 *folgt*

$$T_{x_1/0} = \{t_3, t_5\}, \quad T_{x_1/1} = \{t_2, t_4\}, \quad \textit{und für} \quad z_3$$

$$T_{z_3/0} = \{t_6\}, \quad T_{z_3/1} = \{t_0, t_1, t_2, t_4\}.$$

Dies entspricht den Tests in Tab.3.4.

Testvariable z_3, D-Kette: $z_3 \to y$								
Schritt	x_3	x_2	x_1	z_3	z_2	z_1	y	Hinweise
1 a				D	0	0	D	
2 b		X	0			0		
3 b		0	X			0		
4 b	0		X		0			
5 b	X		0		0			
6	X	X	0					Sensibilisierungs-
7	0	0	X					alternativen
8 c	1	1		1				$z_3/0$
9	1	1	X	1				Fehlerbelegungsalternativen
6,9:10 d	1	1	0	1				t_6
7,9:11 d								$\mathcal{I}$(7,9,x_3, x_2)
12 c	X	0		0				$z_3/1$
13 c	0	X		0				
14	X	0	X	0				Fehlerbelegungs-
15	0	X	X	0				alternativen
6,14:16 d	X	0	0	0				t_0, t_4
7,14:17 d	0	0	X	0				t_0, t_1
6,15:18 d	0	X	0	0				t_0, t_2
7,15:19 d	0	0	X	0				t_0, t_1

Tab. 3.12: Testmustergenerierung für z_3: $T_{z_3/0} = \{t_6\}$, $T_{z_3/1} = \{t_0, t_1, t_2, t_4\}$

3.3 Test synchroner Schaltwerke

Bei einem Schaltwerk sind die Speicherelemente im allgemeinen nicht direkt über die Testeingänge zugänglich. Beim Testvorgang muß die Testvariable auf einen bestimmten Wert einstellbar und die Auswirkung an einem Ausgang beobachtbar sein. Die Einstellung und die Fehlerleitung erfolgen in einer sequentiellen Schaltung durch ein Taktsignal. Der Testvorgang nimmt in der Regel mehrere Taktzeitpunkte in Anspruch. Sind viele Speicherelemente (große **sequentielle Tiefe**) in einem Schaltwerk vorhanden, so steigt die Anzahl der notwendigen Takte.

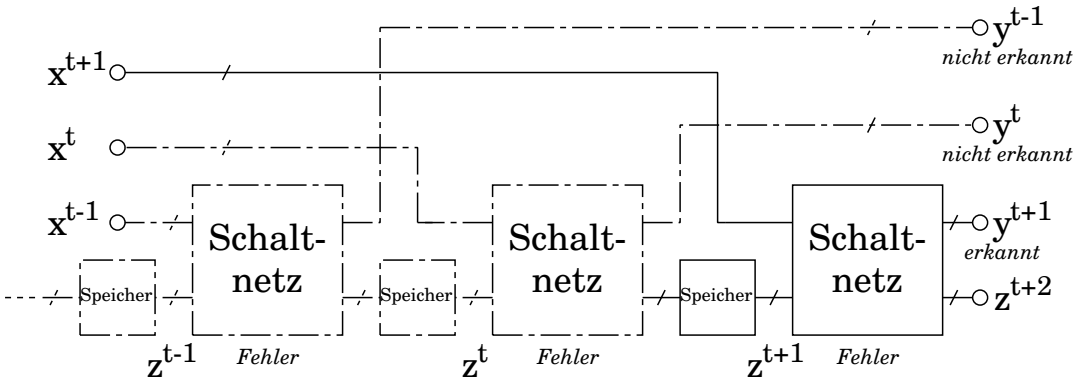


Abb. 3.8: Virtuelles Schaltwerk

Mit der Anzahl der Takte nehmen die Testdauer und damit die Testkosten zu. Sind mehrere Takte für einen Test notwendig, so wird nicht mehr das ursprüngliche sondern ein virtuelles Schaltwerk getestet. In Abb.3.8 ist solch ein virtualisiertes Schaltwerk dargestellt. In diesem Beispiel werden zum Test z.B. drei Taktzeitpunkte benötigt.

$$\text{mögliches Testmuster z.B.:} \quad \boldsymbol{t} = i_n^{t+1} \cdots i_1^{t+1} i_n^t \cdots i_1^t i_n^{t-1} \cdots i_1^{t-1}$$

Das ursprüngliche Schaltwerk wird dadurch scheinbar verdreifacht.

Da der einzelne Ständigfehler zu allen Zeiten besteht, ist dieser in dem virtuellen Schaltwerk dreimal vorhanden. Der Test auf Einzelfehler wird damit zu einem Test auf Mehrfachfehler.

Beispiel 3.7 *In* Abb.3.9 *ist eine sequentielle Schaltung mit einem speichernden Element gegeben. Für die Zwischenvariable z_1 wird mit Hilfe Boolescher Differenzen ein Test für $z_1/0$ bestimmt. Die Testverifikation $(y \oplus y_\alpha = 1)$ erfolgt durch Fehlersimulation.*

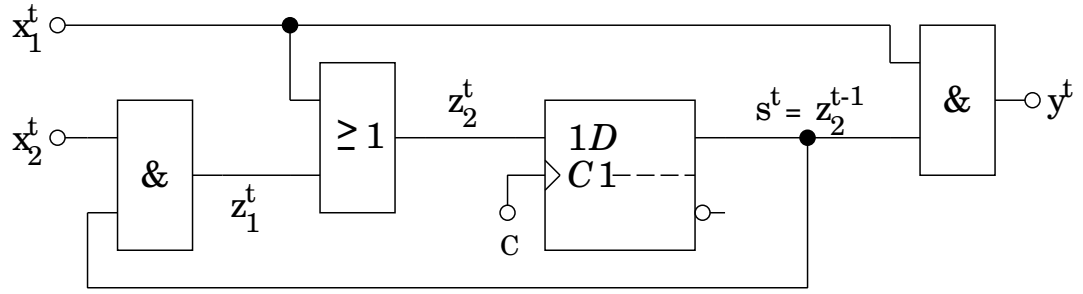


Abb. 3.9: Schaltwerk

Berechnung der Booleschen Differenz $y_{z_1^t}^{t+1}$:

$$\begin{aligned}
y^t &= x_1^t s^t \\
&= x_1^t z_2^{t-1} \\
&= x_1^t \left(x_1^{t-1} \vee z_1^{t-1}\right) \\
y^{t+1} &= x_1^{t+1} x_1^t \vee x_1^{t+1} z_1^t \quad (t+1,\ \textit{weil nur}\ y^{t+1} = f(z_1^t)\ \textit{ist}) \\
y_{z_1^t}^{t+1} &= x_1^{t+1} x_1^t \oplus x_1^{t+1} = x_1^{t+1}\left(x_1^t \oplus 1\right) = x_1^{t+1} \bar{x}_1^t
\end{aligned}$$

Für den Test von $z_1/0$ *folgt:*

$$\begin{aligned} z_1^t \left[y_{z_1^t}^{t+1} \right] &= 1 \\ x_2^t s^t \left[x_1^{t+1} \bar{x}_1^t \right] &= 1 \\ x_2^t z_2^{t-1} \left[x_1^{t+1} \bar{x}_1^t \right] &= 1 \\ x_2^t \left(x_1^{t-1} \vee x_2^{t-1} s^{t-1} \right) \left[x_1^{t+1} \bar{x}_1^t \right] &= 1 \end{aligned}$$

Der in eckigen Klammern eingeschlossene Teil ist die Sensibilisierungsbelegung, der davor stehende die Fehlerbelegung. Die Testmusterfolge ist $x_2^t x_1^{t-1} x_1^{t+1} \bar{x}_1^t = 1$.

Die Sensibilisierung ist von der Testvariablen z_1 *unabhängig, andernfalls wäre es nicht sicher, ob der ermittelte Test gelingt.*

In Tab.3.13 *sind drei Schritte der Fehlersimulation angegeben. Zum Zeitpunkt* $t-1$ *muß* x_1 *auf den Wert* 1 *gesetzt werden* $(x_2^t x_1^{t+1} \bar{x}_1^t = 1)$*, der Wert von* x_2 *ist beliebig. Zur Zeit* t *folgt* $x_2 = 1$ *und* $x_1 = 0$ $(x_1^{t+1} = 1)$*. Dadurch geht der Wert von* y *im fehlerfreien und fehlerbehafteten Fall auf* 0 *(der Fehler wird folglich noch nicht erkannt). Mit* $x_1 = 1$ *und* x_2 *beliebig haben zum nächsten Zeitpunkt* $(t+1)$ y *und* y_α *unterschiedliche Werte* $(y \oplus y_\alpha = 1)$*, der Fehler wird erkannt.*

Zeit	x_2	x_1	y	y_α
$t-1$	X	1	X	X
t	1	0	0	0
$t+1$	X	1	1	0

Tab. 3.13: Fehlersimulation

3.4 Maßnahmen zur Verbesserung der Testbarkeit

Unter dem Begriff **Testbarkeitsverbessernde Maßnahmen** (DFT, s.S. 56) werden Verfahren zusammengefaßt, die eine Verbesserung der Testbarkeit digitaler Schaltungen bewirken. Da durch die begrenzte Anzahl von Ausgangspins bei integrierten Schaltungen auch die Anzahl der Testpunkte beschränkt wird, ist bereits beim Entwurf der Schaltungen auf ihre Testbarkeit zu achten. Diese kann durch zusätzliche Ausgänge für schlecht beobachtbare Signale oder durch Schaltungspartitionierungen (Makrozellen) und die Verwendung von Multiplexern zum Test dieser Makrozellen, erreicht werden. Neben diesen unsystematischen Methoden werden systematische Verfahren, wie z.B. die Schaltungsaufteilung in Speicherelemente und Kombinatorik, Scan-Path-Verfahren und aktive Testhilfen zur Verbesserung der Testfreundlichkeit, angewendet.

3.4.1 Passive Testhilfen

Im Gegensatz zu den aktiven sind passive Testhilfen Verfahren, durch die eine Verbesserung der Testbarkeit erzielt wird, ohne daß gleichzeitig geeignete Testmuster erzeugt werden. Die einfachsten passiven Testhilfen sind zusätzliche Ausgänge für schlecht

beobachtbare Signale. Bei integrierten Schaltungen ist dies wegen der notwendigen Beschränkung auf sehr wenige Ausgänge (Gehäusepins) jedoch nicht sinnvoll. Durch die Entwicklung des **Prüfbusses** (**Scan-Path**) wurde der Zugriff auf interne Knoten möglich, ohne dabei die Anzahl der externen Anschlüsse wesentlich erhöhen zu müssen. Die Verbindung des Prüfbusses mit zustandsgesteuerten (level sensitive) Speicherelementen führte zum **Level Sensitive Scan Design** (LSSD), die Verbindung mit taktflankengesteuerten (edge triggered) zum **Edge Triggered Scan Design** (ETSD). Beide Verfahren setzen synchrone Schaltungsentwürfe voraus.

Mit Hilfe des Scan-Path wird somit der Test sequentieller auf den Test kombinatorischer Schaltungen zurückgeführt. Um den Scan-Path in eine gegebene sequentielle Bausteinstruktur zu integrieren (→ Abb.3.11a), werden alle speichernden Zellen so erweitert, daß sie in einem speziellen Betriebsmodus zu einem Schieberegister (→ Abb.3.11b) zusammengeschaltet werden können. Die notwendigen Erweiterungen z.B. eines D-Flip-Flops zu einer Scan-Path-Zelle sind ein Multiplexer (MUX) und zwei zusätzliche Eingänge (→ Abb.3.10).

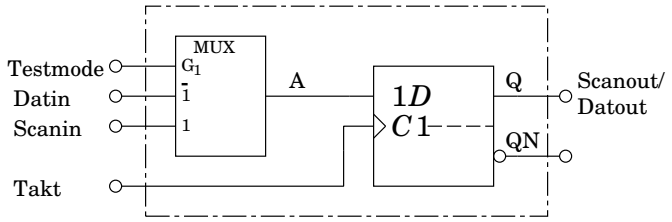


Abb. 3.10: Scan-Path-Grundzelle für ETSP

Der Scanin-Eingang wird zur Zusammenschaltung aller speichernder Elemente zu einem Schieberegister benötigt. Die Zusammenschaltung des Scan-Path erfolgt durch die Verbindung der Scanout- und Scanin-Anschlüsse aufeinanderfolgender speichernder Elemente (→ Abb.3.11b).

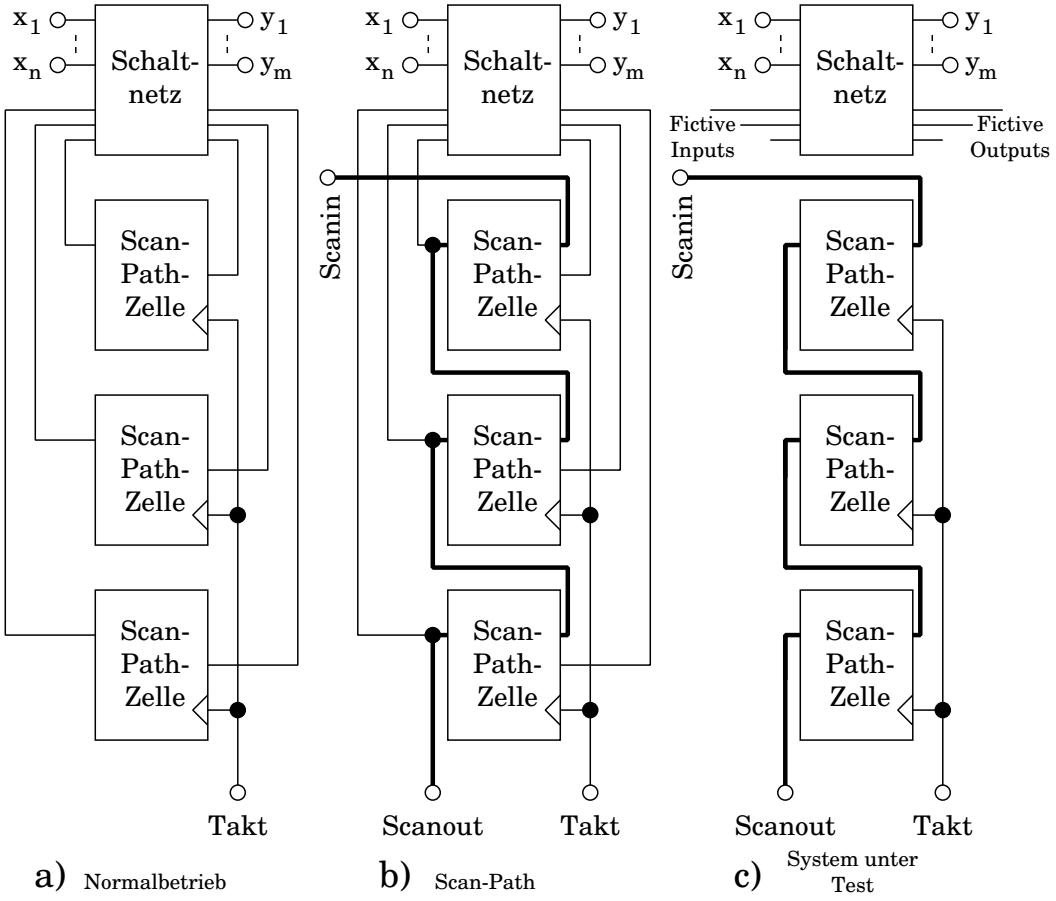


Abb. 3.11: Funktionsweise des Scan-Path: Beispiel mit 3 Scan-Path-Grundzellen

Der Scanin-Anschluß des ersten und der Scanout-Anschluß des letzten Elements werden aus der Schaltung nach außen geführt. Dies erhöht die Anzahl der externen Anschlüsse (**Pins**) um zwei. Mit dem Testmode-Eingang wird einer der Multiplexereingänge ausgewählt.

Im normalen Betriebsmodus (Datin-Eingang) ist Testmode = 0, im Betrieb als Schieberegister (Scanin-Eingang) Testmode = 1.

Da die Betriebsart ebenfalls von außen einstellbar sein muß, wird durch den Testmode-Eingang die Anzahl zusätzlicher externer Anschlüsse auf insgesamt drei erhöht.

Der Ausgang des Flip-Flops kann aufgrund der zeitlichen Trennung zwischen Normalbetrieb und Betrieb als Schieberegister gleichzeitig für Scanout und Datout verwendet werden. Durch den Scan-Path wird jeder Flip-Flop-Ausgang zu einem einstellbaren Schaltungseingang (Fictive Input (→ Abb.3.11c)) und jeder Flip-Flop-Eingang zu einem beobachtbaren Schaltungsausgang (Fictive Output (→ Abb.3.11c)).

Die Durchführung des Tests einer mit einem Scan-Path ausgerüsteten Schaltung auf einem Testautomaten erfolgt in zwei Schritten. In einem ersten Schritt wird durch das Ein- und Ausschieben verschiedener Bitfolgen die speichernde Wirkung und die korrekte Funktion des Scan-Path überprüft. Hierbei ist Testmode = 1 zu setzen.

Im zweiten Schritt wird das Schaltnetz (reine Kombinatorik) geprüft. Der Prüfzyklus hat folgenden Ablauf:

a) Ein Testmuster wird in die durch Testmode = 1 zu eine Schieberegister zusammengeschalteten Flip-Flops eingeschoben und die externen Schaltungseingänge $\boldsymbol{x}$ belegt.

b) Im Betriebsmodus (Testmode = 0) wird die Antwort des Schaltnetzes durch den Takt in den Scan-Path übernommen.

c) Die im Scan-Path gespeicherte Antwort wird mit Testmode = 1 ausgelesen und mit den Sollwerten verglichen.

Die hierfür notwendigen Testmuster können durch eine automatische Testmustergenerierung erzeugt werden.

3.4.2 Aktive Testhilfen

Da jeder Test bei der Scan-Path-Technik das serielle Ein- und Auslesen der Speicher erfordert, nimmt die Testdauer mit der Anzahl der speichernden Elemente zu. Für hochkomplexe Schaltungen kann dies zu unrentabel langen Testzeiten führen. Eine Senkung der Testdauer versprechen aktive Testhilfen. Das Ziel ihrer Einführung besteht vor allem darin, den Test möglichst vollständig innerhalb der Schaltung ablaufen zu lassen, um kurze Testzeiten und einfache Testprogramme zu erhalten.

3.4.2.1 Signaturanalyse

Die Signaturanalyse wurde zuerst in der Nachrichtentechnik zur Bildung von Prüfworten im Rahmen der Sicherung serieller Datenströme eingesetzt. Die Grundlagen der Signaturanalyse stammen aus dem Gebiet der Kodierungstheorie. Dort wird aus einer Nachricht vor einer Übertragung ein Kodewort berechnet, welches mit der Nachricht übertragen wird. Im Empfänger wird dann aus der Nachricht dieses Kodewort nach derselben Vorschrift berechnet und mit dem übertragenen verglichen.

Diese Vorgehensweise kann vorteilhaft auch für den Schaltungstest genutzt werden.

Um mit $(0,1)$-Datenfolgen (3.7) rechnen zu können, müssen diese zuvor in eine äquivalente Polynomdarstellung (3.8) gebracht werden:

$$\mathcal{A}_{m+1} = (A_m, A_{m-1}, \ldots, A_2, A_1, A_0), \quad A_i \in \{0,1\},\ i = 0, \ldots, m \tag{3.7}$$

$$a(x) = A_m x^m + A_{m-1} x^{m-1} + \cdots + A_2 x^2 + A_1 x + A_0 \tag{3.8}$$

Beispiel 3.8 *Für die Folge* $\mathcal{A}_6 = (1,1,0,0,1,1)$ *gilt z.B.* ($\rightarrow$ Abb.3.12):

$$\begin{array}{rl} \textit{Folge}: & (1,1,0,0,1,1) \\ \textit{Polynom}: & x^5 + x^4 + 0 + 0 + x + 1 \\ \textit{Reihenfolge}: & \textit{erstes} \quad \cdots \quad \textit{letztes Element} \end{array}$$

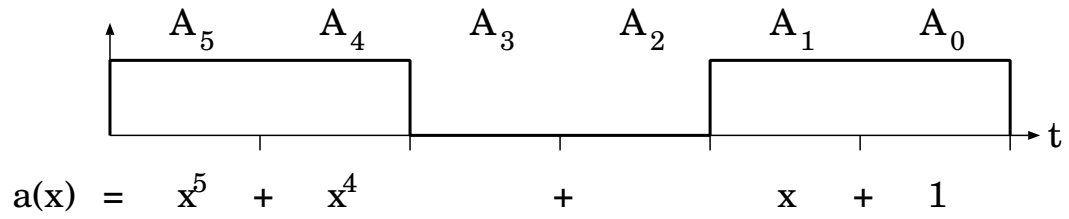


Abb. 3.12: Zuordnung der Datenfolge zu einem äquivalenten Polynom

Die Rechnungen mit den Koeffizienten der Polynome erfolgen modulo 2 (Restklassenalgebra). Die Polynomdivision mit modulo 2 Koeffizientenrechnung wird schaltungstechnisch durch ein über Exor-Gatter rückgekoppeltes Schieberegister (→ Abb.3.13) realisiert.

Zum besseren Verständnis der Funktion der nachfolgenden Schaltungen wird zunächst soweit wie nötig auf die Restklassenalgebra eingegangen.

Zwei Zahlen $a, b \in \mathbb{Z}$ heißen **kongruent modulo m** ($m \in \mathbb{N}$), wenn sie zu derselben **Restklasse modulo m** gehören.

In einer Restklasse modulo m werden alle Zahlen zusammengefaßt, die bei einer Division durch m denselben Rest r haben. Die möglichen Reste r sind $r = 0, 1, \cdots, m-1$. Für eine Restklasse modulo m gilt somit:

$$[r] = \{z | z = m\,q + r;\ q \in \mathbb{Z}\}$$

Beispiel 3.9 *Für $m = 2$ gibt es zwei ($r = 0, 1$) Restklassen:*

$$[0] = \{\ldots, -6, -4, -2, 0, 2, 4, 6, \ldots\} \tag{3.9}$$

$$[1] = \{\ldots, -5, -3, -1, 1, 3, 5, \ldots\} \tag{3.10}$$

($z.B.\ 2 = 2 \cdot 1 + \underbrace{0}_{Rest}$, *oder* $-1 = 2 \cdot (-1) + \underbrace{1}_{Rest}$)

Für die Koeffizientenrechnung gilt (→ Tab.3.14):

$$\boxed{r = (A \pm B) \bmod 2 = A \oplus B \quad \text{mit } A, B \in \{0, 1\}} \tag{3.11}$$

A	B	$A+B$	$A-B$	$(A+B) \bmod 2$	$(A-B) \bmod 2$	$A \oplus B$
0	0	0	0	0	0	0
0	1	1	-1	1	1	1
1	0	1	1	1	1	1
1	1	2	0	0	0	0

Tab. 3.14: Wegen (3.9) ist $\{0, 2\} \in [0]$ und wegen (3.10) $\{-1, 1\} \in [1]$

$A \oplus B$ wird daher auch „Addition modulo 2“ genannt.

Beispiel 3.10 *Die Addition zweier Polynome mit modulo 2 Koeffizienten-Rechnung ergibt z.B.*

$$\begin{aligned}(x^5+x^4+x+1)(+)(x^5+x^3+x) &= \underbrace{(1+1)}_{r=0}x^5+\underbrace{(1+0)}_{r=1}x^4+\underbrace{(0+1)}_{r=1}x^3+\underbrace{(1+1)}_{r=0}x+\underbrace{(1+0)}_{r=1}1\\ &= x^4+x^3+1,\end{aligned}$$

die modulo 2 Subtraktion entsprechend

$$\begin{aligned}(x^5+x^4+x+1)(-)(x^5+x^3+x) &= \underbrace{(1-1)}_{r=0}x^5+\underbrace{(1-0)}_{r=1}x^4+\underbrace{(0-1)}_{r=1}x^3+\underbrace{(1-1)}_{r=0}x+\underbrace{(1-0)}_{r=1}1\\ &= x^4+x^3+1.\end{aligned}$$

Die Division erfolgt ebenfalls mit modulo 2 Koeffizientenrechnung:

$$\begin{array}{l}(\ x^5+x^4+x+1\ )\ (:)\ (\ x^5+x^3+x\ ) = 1+\dfrac{x^4+x^3+1}{x^5+x^3+x}\\ \underline{-\ x^5-x^3-x}\\ x^4+x^3+1\end{array}$$

Wegen (3.11) *gilt ebenso:*

$$\begin{array}{l}(\ x^5+x^4+x+1\ )\ (:)\ (\ x^5+x^3+x\ ) = 1+\dfrac{x^4+x^3+1}{x^5+x^3+x}\\ \underline{\oplus\ x^5\oplus x^3\oplus x}\\ x^4+x^3+1\end{array}$$

Für die Polynomdivision (mit mod-2-Koeffizientenrechnung) gilt allgemein:

$$\frac{a(x)}{b(x)} = c(x) + \frac{s(x)}{b(x)}$$

Dividend:	$a(x)$
Divisor:	$b(x) = B_n x^n + \cdots + B_0, \quad B_i \in \{0,1\},\ i = 0,\ldots,n$
Divisionsrest:	$s(x) = a(x) \bmod b(x) = S_{n-1}x^{n-1} + \cdots + S_0,$ mit $S_i \in \{0,1\}$ und $i = 0,\ldots,n-1$

Beispiel 3.11 *Für den Divisor* $b(x) = x^3+x+1$ *und den Dividenden* $a(x) = x^5+x^4+x+1$ *ergibt die Division z.B.:*

$$\frac{x^5+x^4+x+1}{x^3+x+1} = x^2+x+1+\frac{x}{x^3+x+1}$$

Mit den Koeffizienten $S_1 = 1$ *und* $S_2 = S_0 = 0$ *folgt für den Divisionsrest* $s(x) = x$.

Berechnet man alle Divisionsreste $s_i(x) = x^i \bmod b(x)$ *für* $i = 0,1,2,\ldots,$

$$
\begin{array}{lll}
i=0: & s_0(x)=1 & \mod\ (x^3+x+1)=1,\\
i=1: & s_1(x)=x & \mod\ (x^3+x+1)=x,\\
i=2: & s_2(x)=x^2 & \mod\ (x^3+x+1)=x^2,\\
i=3: & s_3(x)=x^3 & \mod\ (x^3+x+1)=x+1,\\
i=4: & s_4(x)=x^4 & \mod\ (x^3+x+1)=x^2+x,\\
i=5: & s_5(x)=x^5 & \mod\ (x^3+x+1)=x^2+x+1,\\
i=6: & s_6(x)=x^6 & \mod\ (x^3+x+1)=x^2+1,\\
i=7: & s_7(x)=x^7 & \mod\ (x^3+x+1)=1=s_0(x),\\
i=8: & s_8(x)=x^8 & \mod\ (x^3+x+1)=s_1(x),\\
i=9: & s_9(x)=x^9 & \mod\ (x^3+x+1)=s_2(x),\\
 & \vdots, &
\end{array}
$$

so stellt man nach sieben Schritten eine Wiederholung fest:

$$s_{7+i}(x)=s_i(x),\quad i=0,1,2,\ldots$$

Bei dem gewählten Divisor $b(x)=x^3+x+1$ treten alle bei einem Polynom dritter Ordnung ($n=3$) möglichen Divisionsreste (ohne Nullfolge sind es 7) auf.

Die Divisorpolynome $b(x)$ werden in **reduzible** und **irreduzible** unterteilt. Reduzible Polynome lassen sich in irreduzible Teilpolynome zerlegen, wobei die Eigenschaften des gesamten Polynoms von den Eigenschaften der Teilpolynome abhängt. Falls das Polynom irreduzibel (**prim**, **primitiv**, **nicht faktorisierbar**, **maximalperiodisch**) ist, so werden alle Zustände außer dem Null-Zustand (2^n-1) zyklisch durchlaufen (vorheriges Beispiel für $n=3$).

Die Koeffizienten der Divisionsreste, $S_{n-1}, S_{n-2}, \cdots S_0$, bilden mit ihren als Dualzahl interpretierten Belegungen (→ Seite 15)

$$\mathrm{Bin}(i)=i_{S_{n-1}}\cdots i_{S_0},\quad i_{S_j}\in\{0,1\},\ j=0,\ldots,n-1$$

eine quasi zufällige Zahlenfolge (**Pseudozufallszahlenfolge**) der Länge 2^n-1.

Beispiel 3.12

Faßt man die Koeffizientenmuster aus dem vorherigen Beispiel zusammen (→ Tab.3.15), so ergibt dies die Pseudozufallszahlenfolge

$$1,2,4,3,6,7,5,1,2,4,3,6,\ldots.$$

i	S_2	S_1	S_0
1	0	0	1
2	0	1	0
4	1	0	0
3	0	1	1
6	1	1	0
7	1	1	1
5	1	0	1

Tab. 3.15: Divisionsreste $s_i(x)=x^i \mod (x^3+x+1)$, $i=0,\ldots,6$

Die Divisionsrestbildung wird mit Hilfe eines linear rückgekoppelten Schieberegisters (Linear Feedback Shift Register, **LFSR**) realisiert (→ Abb.3.13).

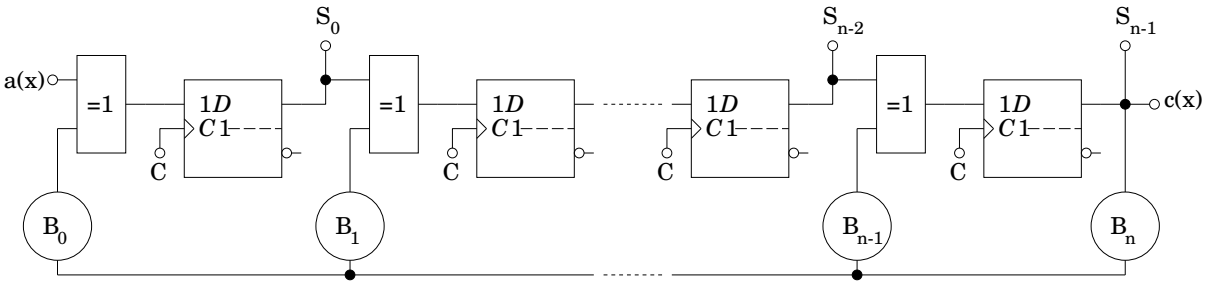


Abb. 3.13: rein rekursives LFSR (Signaturregister)

Durch den Divisor $b(x)$ wird das **Registerpolynom** definiert, der aktuelle **Registerzustand**

$$RZ_n = i_{S_{n-1}} \cdots i_{S_0}$$

kann an den Flip-Flop-Ausgängen abgelesen werden.

Wird eine Eingangsfolge $\mathcal{A}_{m+1}$ in das LFSR eingegeben, so erhält man nach $m+1$ Takten die Ausgangsfolge $\mathcal{C}_{m-n+1}$ und den Registerzustand RZ_n. Die Ausgangsfolge wird verworfen; der Divisionsrest RZ_n wird **Signatur** der Folge $\mathcal{A}_{m+1}$ bezüglich $b(x)$ genannt.

Beispiel 3.13 *Die schaltungstechnische Realisierung des Registerpolynoms* $b(x) = x^3 + x + 1$ *ist in* Abb.3.14 *gegeben.*

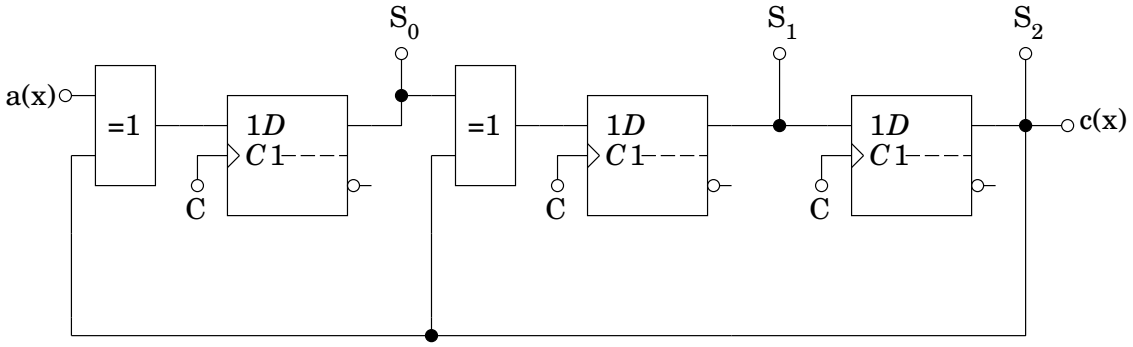


Abb. 3.14: Signaturregister mit $b(x) = x^3 + x + 1$

Wird in das zuvor zurückgesetzte Signaturregister ($RZ_3 = i_{S_2} i_{S_1} i_{S_0} = 000$) *die Eingangsfolge* $\mathcal{A}_6 = (1,1,0,0,1,1)$ *eingegeben* ($a(x) = x^5 + x^4 + x + 1$)*, so stellt sich nach* 6 *Takten der Registerzustand* $RZ_3 = 010$ *ein* ($s(x) = x$*, Berechnung siehe oben).*

Falls die Eingangsfolge $a(x) = 0$ ist und der Anfangszustand $s(x) \neq 0$, so arbeitet das Register als **Zufallszahlengenerator** (Pseudo Noise Generator, **PNG**).

Beispiel 3.14 *Hat z.B. das Signaturregister* (→ Abb.3.14) *den Anfangszustand* $RZ_3 = 011$*, so bildet das getaktete Register für* $a(x) = 0$ *die quasi zufällige Zustandsfolge* $(3,6,7,5,1,2,4)$ *zyklisch mit der Periode 7* (→ Tab.3.15).

3.4.2.2 Testen mit Hilfe der Signaturanalyse

Der Test erfolgt mit einem Zufallsmuster, das von einem maximalperiodischen LFSR erzeugt wird. Die Ausgangsfolge der zu testenden Schaltung wird im Signaturregister zu ihrer Signatur komprimiert ($m \gg n$) und mit der Soll-Signatur verglichen.

$$\text{Zufallszahlengenerator} \implies \text{Schaltung} \implies \text{Signaturregister}$$

Wegen der Kompression der 2^{m+1} möglichen Eingangsfolgen auf 2^n mögliche Signaturen, treten 2^{m+1-n} Folgen mit jeweils gleicher Signatur auf.

Hat eine fehlerhafte Folge die gleiche Signatur wie die fehlerfreie, so wird der Fehler nicht erkannt.

Werden die Nullfolgen nicht berücksichtigt, so ist die Wahrscheinlichkeit dafür, daß ein Fehler *nicht* erkannt wird:

$$p_f = \frac{2^{m+1-n} - 1}{2^{m+1} - 1} \quad \text{bzw. für} \quad m \gg n: \quad p_f \approx \frac{1}{2^n} \tag{3.12}$$

Die Fehlererkennungsrate hängt folglich von der Stellenzahl des Signaturregisters ab. Die Erkennungsrate für Sequenzen, die kürzer sind als die Registerlänge, beträgt 100%, da die Sequenz als Ganzes im Register erhalten bleibt. Längere Sequenzen führen zu einer entsprechend geringeren Fehlererkennungsrate.

3.4.2.3 Selbsttest

Das Testen mit Hilfe der Signaturanalyse ist hervorragend für reguläre Strukturen (z.B. Speicher) geeignet. Für kombinatorische Schaltungen erfolgt der **Selbsttest** (Built-in self-test, **BIST**) in der Regel mit Hilfe eines Prüfbusses. Dazu wird an den Scanin-Eingang ein Pseudozufallszahlengenerator geschaltet und am Scanout-Ausgang die Signatur gebildet. Um zuverlässige Ergebnisse zu erhalten, muß der Selbsttest wiederholbare Signaturen produzieren. Dies ist aufgrund von (3.12) nicht immer möglich.

Eine Schaltung, die für den Selbsttest von Schaltungen mit unregulären Strukturen konzipiert wurde, ist der **Built-In Logic Block Observer** (**BILBO**). Das BILBO-Register kann über drei Steuereingänge in verschiedene Betriebsmodi geschaltet werden.

a) Im Normalbetrieb arbeitet die Schaltung als paralleles Register. Die Flip-Flops werden parallel ein- und ausgelesen, die serielle Verbindung der Flip-Flops zu einem Schieberegister ist unterbrochen.

b) Im Betrieb als Schieberegister werden die parallelen Eingänge abgetrennt und die serielle Verbindung hergestellt.

c) Für die Funktion als Zufallszahlengenerator wird zusätzlich zur seriellen Verbindung der Flip-Flops der Rückkopplungszweig geschlossen (die Eingänge sind abgetrennt).

d) Durch den Anschluß der Eingänge wird aus dem Zufallszahlengenerator ein Signaturregister mit parallelen Eingängen (eine Alternative zu dem oben beschriebenen seriellen Signaturregister, geeignet für Schaltungen mit mehr als nur einem beobachtbaren Ausgang).

Die Anwendung des BILBO-Registers setzt voraus, daß eine Schaltung in Speicherelemente und dazwischenliegende Schaltnetze aufgeteilt werden kann. Die Speicherelemente werden so zusammengefaßt, daß sich unabhängige Schaltnetze ergeben, zwischen denen jeweils ein BILBO-Register angeordnet wird. Ist dies nicht möglich, so müssen Verbindungen innerhalb der Schaltnetze aufgetrennt werden, um diese Bedingung zu erfüllen.

Zum Test wird jeweils ein BILBO-Register als Zufallszahlengenerator betrieben, um das zu testende Schaltnetz mit Testmustern zu versorgen, während das BILBO-Register an den Ausgängen des Schaltnetzes als Signaturregister arbeitet. Der Reihe nach vertauschen nun die BILBO-Register so lange ihre Funktion, bis alle Schaltnetze getestet sind.

Die Testmustergenerierung und die Testauswertung erfolgen beim Selbsttest in der Schaltung, d.h., direkt auf dem Chip.

Die Vorteile des Selbsttests sind:

a) Es ist keine Testumgebung erforderlich,

b) der Test kann jederzeit (z.B. in den Betriebspausen) erfolgen,

c) es sind nur wenige zusätzliche Pins erforderlich,

d) die Testhardware entspricht der Chiptechnologie.

3.4.3 Neuere Entwicklungen

In einer digitalen Schaltung wird ein Fehler erkannt, wenn ein Testmuster den Fehlerort auf einen dualen Wert zum Fehler einstellen kann und dies eine beobachtbare Auswirkung an einem Ausgang hat. Zwei neue Testmethoden konzentrieren sich auf die Lösung des Problems der Beobachtbarkeit.

Die Firma CrossCheck fügt zu diesem Zweck den Gattern ein Netz von Testpunkten (→ Abb.3.15) hinzu.

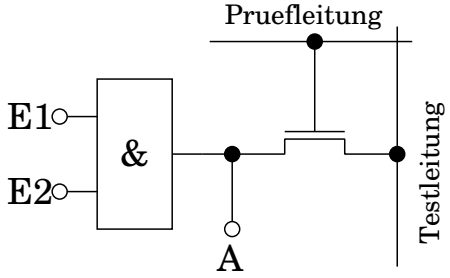


Abb. 3.15: Patentierte Methode der Firma CrossCheck

Ein Testpunkt wird durch die Erweiterung des Gatterausgangs um einen Transistor (CMOS) gebildet. Dieser Transistor leitet zum Testzeitpunkt den Wert des Gatterausgangs weiter. Initiiert wird der Test über eine Prüfleitung. Das Testergebnis wird dann entweder direkt mit einem Sollwert verglichen oder zur Berechnung einer Signatur verwendet.

Diese Methode erlaubt auch den Test asynchroner Schaltungen sowie den Test von anderen als Haftfehlern (**non-stuck-at-faults**).

Ein zweites Verfahren nutzt die statischen Eigenschaften der CMOS-Gatter aus.

Im statischen (eingeschwungenen) Zustand fließt über diese Gatter nur ein sehr geringer Strom ($\approx 1\ pA$ oder weniger). Tritt dagegen ein Fehler auf, so erhöht sich dieser Wert auf 20 - 100 μA (abhängig von der Anzahl der Gatter). Diese Eigenschaft kann sehr einfach zum Testen der Schaltung genutzt werden (**quiescent current testing** bzw. I_{DDQ}-Methode).

Da die Spannungsversorgung an alle Gatter geht, hat sie beim Test die Funktion eines Testnetzwerks. Zum Test wird ein Testmuster an die Schaltungseingänge gelegt und abgewartet, bis alle Umschaltvorgänge in der Schaltung beendet sind. Dann wird der Stromverbrauch gemessen. Liegt er in der oben angegebenen Größenordnung, so muß in der Schaltung ein Fehler angenommen werden. Durch das Erzeugen beider logischer Zustände an allen Schaltungsknoten können mit dieser Methode alle Fehlermöglichkeiten überprüft werden.

Durch die I_{DDQ}-Methode wird eine hohe Fehlerüberdeckung ohne zusätzlichen Schaltungsaufwand erzielt. (Schaltungsteile, die hohen statischen Stromverbrauch haben (z.B. SRAMs), müssen beim Test der Schaltung isoliert werden.)

3.5 Wirtschaftliche Aspekte

Parallel zu steigender Integrationsdichte und Komplexität integrierter Schaltkreise geht die Forderung nach höherer Qualität und schnellerer Marktreife (time to market) der Produkte.

Untersuchungen von McKinsey & Co. (1989) haben gezeigt, daß eine um nur wenige Monate verspätete Marktpräsenz schlimmere Folgen haben kann, als eine 30%-ige Überschreitung des ursprünglichen Entwicklungsetats.

Nimmt man an, daß der Verkauf eines Chips einer trapezförmigen Kurve folgt ($\rightarrow$ Abb.3.16), so geht der Gewinn nach einem 6-monatigen Anstieg, einem Jahr Stagnation und 8-monatiger Abnahme der Verkaufsrate um 36% zurück, wenn das Produkt um nur drei Monate zu spät auf den Markt kommt.

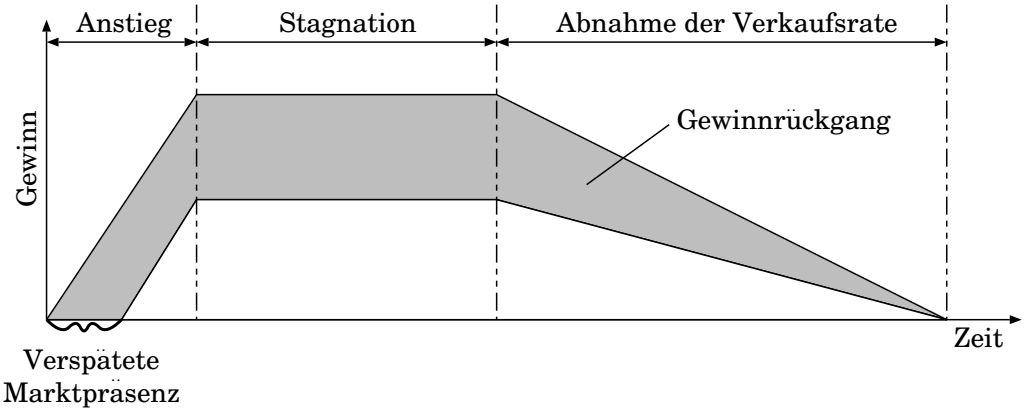


Abb. 3.16: Gewinnrückgang (schattiert) durch verspätete Marktpräsenz

Eine Verspätung um ein Achtel der Lebenszeit eines Produkts bedeutet somit einen Gewinnrückgang um über ein Drittel.

Im gesamten Entwicklunsablauf beansprucht die Testvorbereitung ungefähr ein Drittel der Zeit. Dennoch versagt jeder dritte integrierte Schaltkreis nach seinem Test im späteren Betrieb.

Da die Kosten für das Auffinden fehlerhafter Komponenten mit der Komplexität der Einbaustufe (Chip $\rightarrow$ Leiterplatte $\rightarrow$ Gerät $\rightarrow$ System) jeweils um einen Faktor > 10 zunimmt, muß in erster Linie die Fehlerfreiheit des integrierten Schaltkreises gesichert sein. Dazu ist eine hohe Fehlerüberdeckungsrate beim IC-Test Voraussetzung.

Eine gründliche Testvorbereitung unter Berücksichtigung aller in diesem Kapitel aufgeführter Methoden stellt folglich einen wesentlichen Wirtschaftsfaktor dar. Hohe Qualität und schnelle Marktreife kann nur mit optimalen, für die jeweilige Schaltung angepaßten Methoden (Design for Testability), erreicht werden.

Aufgaben

A 3.1 Stellen Sie für die gegebene Trägermenge $\mathbb{B} = \{0, a, b, c, d, e, f, 1\}$ die Operationstafel für die Disjunktion ($\vee$) auf. Nutzen Sie dabei die Isomorphie zur Mengenalgebra (die Reihenfolge der Elemente von $\mathbb{B}$ ist mit derjenigen der Elemente der Potenzmenge $\underline{P}(M)$ gleichzusetzen) aus. Die Potenzmenge ist

$$\underline{P}(M) = \{\emptyset, \{\alpha\}, \{\beta\}, \{\gamma\}, \{\alpha, \beta\}, \{\alpha, \gamma\}, \{\beta, \gamma\}, M\}.$$

A 3.2 Für eine vierelementige Boolesche Algebra mit der Trägermenge $\mathbb{B} = \{0, 1, 2, 3\}$ und den neutralen Elementen 0 für die Disjunktion und 3 für die Konjunktion ist der Ausdruck

$$\overline{(\overline{2 \vee 1}) \wedge 1 \vee 1}$$

zu vereinfachen.

A 3.3 Geben Sie für

$$y = \overline{x_1 \to x_2} \oplus x_2$$

eine äquivalente DNF an.

A 3.4 Geben Sie für die Erfüllungsmenge

$$I_f = \{0, 2, 4, 6, 8, 9, 10, 14, 16, 20, 24, 25\}$$

die Funktion $f(\boldsymbol{x})$ als minimale DNF an.

A 3.5 Für eine Schaltfunktion mit $n = 5$ Variablen gilt die Erfüllungsmenge

$$I_f = \{2, 6, 9, 10, 13, 14, 18, 20, 25, 26, 28, 29\}.$$

Bestimmen Sie die minimale DNF der Funktion.

A 3.6 Bestimmen Sie die KDNF der Funktion

$$f(\boldsymbol{x}) = [\overline{x_3 \to x_2} \leftrightarrow x_2] \wedge [x_1 \oplus x_3]\,.$$

A 3.7 Für die Schaltfunktion $f(\boldsymbol{x}) = \overline{(\bar{x}_4 \vee x_3)x_2}\overline{(x_3 x_2 x_1 \vee x_1)}$ ist die KNF und die minimale DNF aufzustellen.

A 3.8 Bestimmen Sie die Ringsumme der Funktion $f(\boldsymbol{x}) = x_2(\bar{x}_4 \vee x_3) \vee \bar{x}_1$.

A 3.9 Für die Schaltfunktion

$$f(\boldsymbol{x}) = (\bar{x}_3 \vee \bar{x}_4)(\bar{x}_4 \vee \bar{x}_2)(\bar{x}_3 \vee x_1)(\bar{x}_2 \vee x_1)$$

ist die kürzeste DNF aufzustellen.

A 3.10 Berechnen Sie die KKNF der Funktion

$$f(x_1, x_2, x_3) = \overline{[(\bar{x}_3 \vee x_2)x_3]} \vee x_1 x_3.$$

A 3.11 Berechnen Sie die kürzeste Ringsummendarstellung für

$$f(\boldsymbol{x}) = \overline{\bar{x}_1 x_4 x_3 \vee x_1 \overline{x_4 x_3}}$$

und zeichnen Sie die Schaltung mit EXOR- und UND-Gattern.

A 3.12 Bestimmen Sie aus der Primtermtabelle

	m_2	m_3	m_{10}	m_{11}	m_{18}	m_{20}	m_{22}	m_{26}	m_{28}	m_{30}	m_{21}	m_{29}
p_1	×	×	×	×								
p_2	×		×		×			×				
p_3					×		×	×		×		
p_4						×	×		×	×		
p_5						×			×		×	×

die DNF der zugehörigen Funktion $f(x_1, x_2, x_3, x_4, x_5)$.

A 3.13 Die Erfüllungsmenge einer Funktion mit $n = 3$ lautet

$$I_f = \{0, 1, 3, 4, 6\}.$$

Ermitteln Sie mit Hilfe des Quine-McCluskey-Verfahrens die Primterme dieser Funktion.

A 3.14 Zeichnen Sie für

$$y = \bar{x}_1 x_2 \vee x_1 x_3$$

das Schaltnetz. Verwenden Sie dabei für die Zwischenvariablen die Namen

$$z_1 = \bar{x}_1, \quad z_2 = z_1 \wedge x_2, \quad z_3 = x_1 \wedge x_3 \quad \text{und} \quad y = z_2 \vee z_3.$$

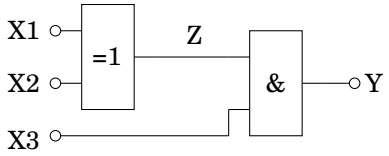
Geben Sie für den Pfad $x_1 \to z_1 \to z_2 \to y$ die Signallaufzeit $t_{D_{HL}}$ bei Vernachlässigung der Leitungslaufzeiten an ($x_2 = 1, z_3 = 0$).

A 3.15 Zeichnen Sie für

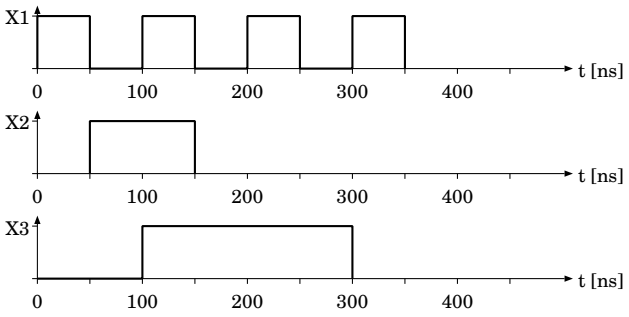
$$y = x_2 x_1 \oplus (x_3 \vee x_1)$$

das Schaltnetz. Verwenden Sie als Zwischenvariable $z_1 = x_2 x_1$ und $z_2 = x_3 \vee x_1$. Zum Zeitpunkt $t = 0$ findet an x_1 ein Signalwechsel von $0 \to 1$ statt, an x_2 liegt konstant 1, an x_3 konstant 0 an. Zeichnen Sie den Signalverlauf von z_1, z_2 und y über t ($CL_{E1,E2}(\text{EXOR}) = \Delta t = 0$).

A 3.16 Für das folgende Schaltnetz



und den gegebenen Verlauf der Eingangssignale



ist der Verlauf des Ausgangssignals Y zu zeichnen. Es sind dabei nur die Verzögerungszeiten der Gatter zu berücksichtigen ($\Delta t_d = CL_{E1}(AN2) = 0$).

A 3.17 Ergänzen Sie für $y = x_2x_1 \oplus (x_3 \vee x_1)$ (→ A 3.15) die nachfolgende Fehlerüberdeckungstabelle um die fehlenden Werte.

			f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}
t_μ	$x_3x_2x_1$	y	$x_1/0$	$x_2/0$	$x_3/0$	$x_1/1$	$x_2/1$	$x_3/1$	$z_1/0$	$z_1/1$	$z_2/0$	$z_2/1$	$y/0$	$y/1$
0	000													
1	001													
2	010													
3	011													
4	100													
5	101													
6	110													
7	111													

A 3.18 Die nachfolgende Fehlerüberdeckungstabelle ist für $y = \bar{x}_1x_2 \vee x_1x_3$ (→ A 3.14) um die fehlenden Werte zu ergänzen.

			f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}
t_μ	$x_3x_2x_1$	y	$x_1/0$	$x_2/0$	$x_3/0$	$x_1/1$	$x_2/1$	$x_3/1$	$z_1/0$	$z_1/1$	$z_2/0$	$z_2/1$	$z_3/0$	$z_3/1$	$y/0$	$y/1$
0	000															
1	001															
2	010															
3	011															
4	100															
5	101															
6	110															
7	111															

Bestimmen Sie aus der Fehlerüberdeckungstabelle eine minimale Testmenge T_{min}.

A 3.19 Bestimmen Sie für $y = \bar{x}_1x_2 \vee x_1x_3$ (→ A 3.14) die Ringsummendarstellung.

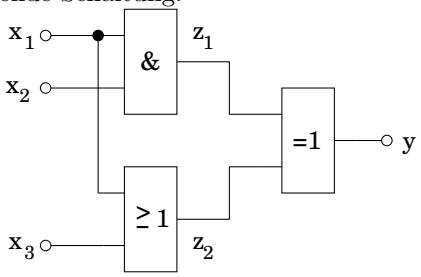
Berechnen Sie daraus die Booleschen Differenzen y_{x_1}, y_{x_2} und y_{z_1} und die Tests für $x_1/0, x_2/1$ und $z_1/0$.

Ermitteln Sie die Tests für $x_3/0$, $x_3/1$ und $z_3/0$ mit Hilfe des D-Algorithmus.

A 3.20 Bestimmen Sie aus der folgenden Fehlerüberdeckungstabelle eine minimale Testmenge.

t_i	f_1 $x_1/0$	f_2 $x_2/0$	f_3 $x_3/0$	f_4 $x_1/1$	f_5 $x_2/1$	f_6 $x_3/1$	f_7 $z_1/0$	f_8 $z_1/1$	f_9 $z_2/0$	f_{10} $z_2/1$	f_{11} $z_3/0$	f_{12} $z_3/1$	f_{13} $y/0$	f_{14} $y/1$
0					×					×		×		×
1						×				×		×		×
2		×	×	×			×		×				×	
3	×					×		×		×		×		×
4				×	×			×		×		×		×
5	×		×								×		×	
6		×	×				×		×				×	
7			×								×		×	

A 3.21 Gegeben ist die folgende Schaltung:



Bestimmen Sie die Ausgangswerte der fehlerfreien sowie der fehlerbehafteten Schaltung und tragen Sie die Unterschiede (×) in die nachfolgende Fehlerüberdeckungstabelle ein.

t_i	$x_3x_2x_1$	y	f_1 $x_1/0$	f_2 $x_2/0$	f_3 $x_3/0$	f_4 $x_1/1$	f_5 $x_2/1$	f_6 $x_3/1$	f_7 $z_1/0$	f_8 $z_1/1$	f_9 $z_2/0$	f_{10} $z_2/1$	f_{11} $y/0$	f_{12} $y/1$
t_0	000													
t_1	001													
t_2	010													
t_3	011													
t_4	100													
t_5	101													
t_6	110													
t_7	111													

A 3.22 Bestimmen Sie aus der gegebenen Fehlerüberdeckungstabelle

	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}
t_0							×		×		×	
t_1		×	×				×		×		×	
t_2	×		×				×		×		×	
t_3				×	×			×		×	×	
t_4	×	×					×		×		×	
t_5				×		×		×		×	×	
t_6					×	×		×	×			×
t_7								×	×			×

die Mindesttestmenge T_{min}.

A 3.23 Am Eingang x_2 der Schaltung aus Aufgabe 3.21 liege eine 1, am Eingang x_3 eine 0 an. Am Eingang x_1 tritt ein Signalwechsel von $0 \rightarrow 1$ auf. Der Ausgang y sei unbeschaltet ($CL = 0$). Die Leitungslaufzeit des Signals z_1 beträgt $\Delta t_{d_{HL}} = \Delta t_{d_{LH}} = 0.3$ns.

Mit welcher zeitlichen Verzögerung tritt der Signalwechsel am Ausgang y auf?

A 3.24 Bestimmen Sie für

$$y = \underbrace{x_2 x_1}_{z_1} \oplus \underbrace{(x_3 \vee x_1)}_{z_2}$$

die Booleschen Differenzen y_{x_1}, y_{x_3} und y_{z_2}. Ermitteln Sie die Tests für $x_1/1$, $x_3/0$ und $z_2/1$.

A 3.25 Geben Sie eine PLA-Realisierung für die Schaltung aus Aufgabe 3.21 an.

A 3.26 Gegeben ist die Schaltfunktion $y = \bar{x}_1 \vee x_2 x_3$. Die Schaltung soll ausschließlich mit NOR-Gattern und Invertern realisiert werden. Daraus ergeben sich folgende Zwischenvariable:

$$z_1 = \bar{x}_1; \quad z_2 = \overline{z_1 \vee x_2}; \quad z_3 = \overline{z_1 \vee x_3}; \quad y = \overline{z_2 \vee z_3}$$

Skizzieren Sie die Schaltung. Bestimmen Sie die Tests für $x_2/0$, $x_2/1$ und $z_2/0$ mit Hilfe des D-Algorithmus.

A 3.27 Bestimmen Sie die Ringsummendarstellung für

$$y = (x_2 \vee x_1) \oplus x_3 x_1 \text{ mit } z_1 = x_2 \vee x_1 \text{ und } z_2 = x_3 x_1.$$

Berechnen Sie die Booleschen Differenzen y_{z_1}, y_{x_1}, y_y und die Tests für $z_1/0$, $z_1/1$, $x_1/0$, $x_1/1$ und $y/0$ sowie $y/1$.

II.
Analoge Schaltungen

4 Einleitung zum zweiten Teil

Parallel zur steigenden Komplexität integrierter Schaltungen (mehr als 100000 aktive Elemente pro Chip) wurden neue Verfahren zur ihrer Überprüfung entwickelt. Dies führte vom zunächst diskreten Aufbau kleinerer Schaltungen und ihrer Messung zur rechnergestützten Simulation der durch Modelle beschriebenen integrierten Schaltung.

In den nachfolgenden Kapiteln wird der Ablauf des rechnergestützten Schaltungsentwurfs auf **Schaltkreisebene** behandelt. Auf dieser Ebene werden die Schaltungen durch Transistoren, Kapazitäten, Widerstände, Leitungen, usw., beschrieben.

Die für diese Beschreibung entwickelten Modelle geben das physikalische Verhalten der realen Bauelemente bei ihrer Simulation auf dem Rechner wieder. Die Genauigkeit der Modelle und der verwendeten Verfahren (Algorithmen) bestimmen dabei sowohl die Zuverlässigkeit des berechneten Schaltungsverhaltens als auch die Zeit, die der Rechner dafür benötigt.

4.1 Analysearten

Die drei wichtigsten Simulationen analoger Schaltungen sind

a) die Gleichstrom-(DC)-Analyse,

b) die Wechselstrom-(AC)-Analyse und

c) die Einschwing- oder Transienten-(TR)-Analyse.

Sie bilden die Grundlage für weitere Verfahren, wie z.B.

- die Rausch-(Noise)-Analyse (AC),
- die Verzerrungs-(Distorsion)-Analyse (AC) oder
- die Empfindlichkeits-(Sensitivity)-Analyse (DC bzw. TR).

4.1.1 Gleichstrom-Analyse

Hierbei wird das Verhalten einer Schaltung bei Anregung mit Gleichstrom oder Gleichspannung untersucht. Da dabei Kapazitäten als Unterbrechungen und Induktivitäten als Kurzschlüsse wirken, besteht die Aufgabe darin, das Verhalten des verbleibenden linearen oder nichtlinearen Widerstandsnetzwerkes zu berechnen.

Für ein nichtlineares Netzwerk erfolgt die Berechnung iterativ mit Hilfe linearisierter Ersatzschaltungen (diese enthalten nur Widerstände und Quellen). Die Lösung wird als **Arbeitspunkt** (**Operating Point**) bezeichnet.

Durch die Berechnung mehrerer Arbeitspunkte für verschiedene Werte einer Quelle an einem definierten Eingang kann eine Kennlinie für diesen Eingang (Driving Point (DP) Characteristic) ermittelt werden.

Ebenso kann für den Zusammenhang zwischen einer Ausgangsgröße und einer Eingangsgröße eine Übertragungskennlinie (Transfer Characteristic (TC)) bestimmt werden.

Durch den Arbeitspunkt sind ferner die Anfangsbedingungen (Initial Conditions (IC)), d.h., die Spannungen an den Kapazitäten und die Ströme durch die Induktivitäten des Netzwerks, gegeben. Diese Werte werden häufig als Ausgangsgrößen für weitere Analysen benötigt.

4.2 Wechselstrom-Analyse

Bei der Wechselstromanalyse wird das Verhalten einer Schaltung bei Anregung mit einem sinusförmigen Signal *kleiner* Amplitude berechnet. Dazu werden die nichtlinearen Kennlinien aller Schaltungselemente im Arbeitspunkt (der durch eine zusätzlich anliegende Gleichspannung oder Gleichstrom festgelegt wird) linearisiert. Durch die Linearisierung erhält man sogenannte Kleinsignalmodelle, die im einfachsten Fall den linearisierten Ersatzschaltungen der Gleichstromanalyse entsprechen. Soll das dynamische Verhalten der Elemente (z.B. der Transistoren) stärker berücksichtigt werden, so werden die Kleinsignalmodelle um (parasitäre) Kapazitäten erweitert.

Durch wiederholte AC-Analysen für verschiedene Frequenzen eines Eingangssignals kann der Frequenzgang einer Schaltung (Bode-Diagramme) ermittelt werden.

4.3 Transient-Analyse

Die Einschwinganalyse berechnet das zeitliche Verhalten einer nichtlinearen dynamischen Schaltung als Antwort auf ein sich änderndes Eingangssignal. Zur Berechnung werden die nichtlinearen Elemente (Dioden, Transistoren, usw.) linearisiert und die dynamischen (C's, L's, usw.) diskretisiert. Mit Hilfe der linearisierten Elemente (siehe Gleichstromanalyse) wird iterativ das Verhalten der nichtlinearen Großsignalmodelle, mit den diskretisierten Elementen (die Diskretisierung führt wieder auf Widerstände und Quellen) durch numerische Integration das Zeitverhalten der Ströme und Spannungen der Schaltung bestimmt.

Alle drei Methoden verwenden zur Berechnung der Schaltungsantworten lineare oder linearisierte Netzwerke. Diese enthalten nur Widerstände und Quellen (bei der AC-Analyse zusätzlich speichernde Elemente, wie z.B. Spulen und Kapazitäten). Die für alle Netzwerke gleiche Knotenanalyse wird im folgenden Kapitel behandelt.

5 Analyse linearer Netzwerke

Die Berechnung linearer Netzwerke ist ein wesentlicher Bestandteil der rechnergestützten Netzwerkanalyse. Einmal können die meisten Netzwerkprobleme als linear angesehen werden und zum anderen erfolgt auch die Analyse nichtlinearer statischer und dynamischer Schaltungen durch wiederholte Berechnung geeignet linearisierter Netzwerke.

Die Analyse erfolgt in der Regel in zwei Schritten:

a) Aufstellen der beschreibenden Netzwerkgleichungen.

b) Numerische Lösung der Gleichungssysteme.

Jede Analyseaufgabe kann unabhängig von der Schaltungsart durch eine Knotenpotentialanalyse gelöst werden.

5.1 Die Inzidenzmatrix eines gerichteten Graphen

Die Aufbaustruktur eines Netzwerks läßt sich vorteilhaft entweder zeichnerisch (in Form eines **gerichteten Graphen**) oder mathematisch (durch **topologische Matrizen**) beschreiben. Solch eine topologische (nur die Aufbaustruktur beschreibende) Matrix ist die **Knoten-** bzw. **Inzidenzmatrix**.

Beispiel 5.1 *Die Aufbaustruktur eines Netzwerks* (→ Abb.5.1) *wird durch einen gerichteten Graphen* (→ Abb.5.2) *wiedergegeben.*

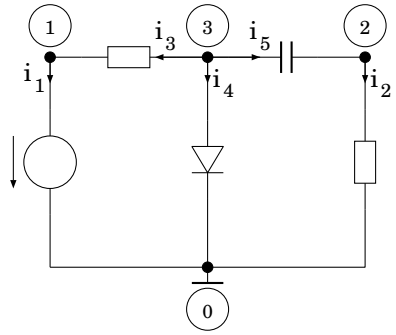


Abb. 5.1: Schaltungsmodell

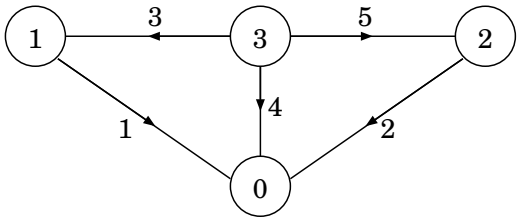


Abb. 5.2: Gerichteter Graph

Jedes Element der Schaltung bildet eine Kante des Graphen, die Richtung der Ströme (willkürlich) legt dabei die Richtung der Kanten fest. Jede Kante beginnt und endet an einem Knoten (Numerierung der Knoten beliebig).

Knotenmatrizen geben an, welche Zweige eines Graphen mit welchen Knoten inzident sind. Entsprechen die Knoten den Zeilen und die Zweige den Spalten der Matrix, so kann für einen gerichteten Graphen die *vollständige* Inzidenzmatrix mit folgender Definition der Matrizenelemente $a_{\mu\nu}$ aufgestellt werden:

$$a_{\mu\nu} = \begin{cases} 1, & \text{wenn der Zweig } \nu \text{ vom Knoten } \mu \text{ wegführt} \\ -1, & \text{wenn der Zweig } \nu \text{ zum Knoten } \mu \text{ hinführt} \\ 0, & \text{wenn der Zweig } \nu \text{ mit Knoten } \mu \text{ nicht inzident ist} \end{cases} \tag{5.1}$$

Beispiel 5.2 *Für das vorhergehende Beispiel 5.1 ergibt sich die* vollständige *Inzidenzmatrix mit* (5.1) *zu:*

$$\boldsymbol{A}_e = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & -1 \\ 1 & 0 & -1 & 0 & 0 \\ -1 & -1 & 0 & -1 & 0 \end{pmatrix}$$

Die Knoten 3, 2, 1, 0 sind den Zeilen 1, 2, 3, 4 zugeordnet, die Reihenfolge der Kanten entspricht derjenigen der Spalten.

Eine genauere Betrachtung der im Beispiel 5.2 aufgestellten Matrix $\boldsymbol{A}_e$ zeigt, daß die Summe ihrer Zeilen stets Null ist. Dies ist so, weil jeder Zweig von einem Knoten ausgeht und an einem Knoten endet. Die Zeilen der Matrix sind folglich voneinander linear abhängig. Es kann daher eine beliebige Zeile der Matrix gestrichen werden, ohne die Information über die Struktur des zugrundeliegenden Graphen zu verlieren (im folgenden wird immer die zum Masseknoten gehörende Zeile bei der Aufstellung der Matrix weggelassen).

Beispiel 5.3 *Durch das Streichen der zum Knoten 0 gehörenden Zeile erhält man für die Inzidenzmatrix aus dem Beispiel 5.2*

$$\boldsymbol{A} = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & -1 \\ 1 & 0 & -1 & 0 & 0 \end{pmatrix}.$$

Für ein Netzwerk mit $|\boldsymbol{N}| = \boldsymbol{n+1}$ **Knoten** und $|\boldsymbol{K}| = \boldsymbol{k}$ **Kanten** hat die Matrix $\boldsymbol{A}$ die Dimension $< n \times k >$. Mit Hilfe der Inzidenzmatrix können sehr einfach Gleichungen für die Ströme und Spannungen eines Netzwerks formuliert werden.

5.2 Das Knoten-Spannungs-System

Für zeitabhängige Größen werden im folgenden kleine Buchstaben (z.B. u), für Gleichstromgrößen große (z.B. U), für Quellen wird der Index 0 verwendet (z.B. U_0) und komplexe Größen werden unterstrichen (z.B. $\underline{i}$ oder $\underline{U}$).

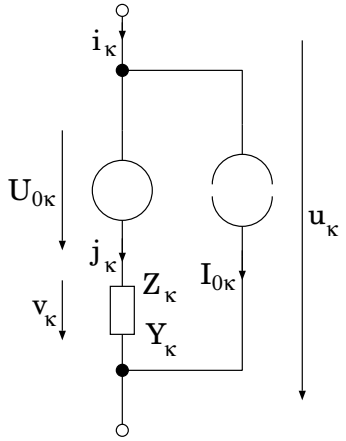
Mit den Kantenstrom- und Kantenspannungsvektoren $\boldsymbol{i} \in \mathbb{R}^r$ und $\boldsymbol{u} \in \mathbb{R}^r$ und dem Knotenspannungsvektor $\boldsymbol{u}_n \in \mathbb{R}^n$ folgt allgemein für das **Kirchhoffsche Stromgesetz**

$$\boxed{\text{KI: } \boldsymbol{A} \cdot \boldsymbol{i} = \boldsymbol{0}} \tag{5.2}$$

und für das **Kirchhoffsche Spannungsgesetz**

$$\boxed{\text{KU: } \boldsymbol{u} = \boldsymbol{A}^T \cdot \boldsymbol{u}_n} \tag{5.3}$$

Zur rechnerorientierten Netzwerkbeschreibung hat sich die Einführung eines sogenannten allgemeinen Zweiges bzw. einer **Standardkante** (→ Abb.5.3) als vorteilhaft erwiesen.



Z_κ	:	Kantenwiderstand (Impedanz)
Y_κ	:	Kantenleitwert (Admittanz)
$Z_\kappa = R_\kappa$	:	Gleichstromfall
$Z_\kappa = R_\kappa + j\omega L_\kappa$	:	Wechselstromfall (stationär)
$Z_\kappa = L_\kappa \frac{d}{dt}$	:	Wechselstromfall (dynamisch)

Problemfälle:

$Y_\kappa \to \infty$	:	bei „Y“-Kanten
$Z_\kappa \to \infty$	:	bei „Z“-Kanten

Abb. 5.3: Standardkante

Mit dem Ohmschen Gesetz (OG) folgt für eine Standardkante κ

$$\begin{array}{rclcrcl} v_\kappa & = & u_\kappa - U_{0\kappa} & = & Z_\kappa(i_\kappa - I_{0\kappa}) & = & Z_\kappa \cdot j_\kappa, \\ j_\kappa & = & i_\kappa - I_{0\kappa} & = & Y_\kappa(u_\kappa - U_{0\kappa}) & = & Y_\kappa \cdot v_\kappa \end{array}$$

und für alle Kanten $1, \ldots, k$ in Vektorschreibweise:

$$\begin{array}{rclcrcl} \boldsymbol{v} & = & \boldsymbol{u} - \boldsymbol{U}_0 & = & \boldsymbol{Z}(\boldsymbol{i} - \boldsymbol{I}_0) & = & \boldsymbol{Z} \cdot \boldsymbol{j}, \\ \boldsymbol{j} & = & \boldsymbol{i} - \boldsymbol{I}_0 & = & \boldsymbol{Y}(\boldsymbol{u} - \boldsymbol{U}_0) & = & \boldsymbol{Y} \cdot \boldsymbol{v} \end{array}$$

Die Leitwertmatrix $\boldsymbol{Y}$ und die Widerstandsmatrix $\boldsymbol{Z}$ sind Diagonalmatrizen:

$$\boldsymbol{Y} = \text{diag}(Y_1, \ldots, Y_\kappa, \ldots, Y_k) \in \mathbb{C}^{k \times k}, \quad \boldsymbol{Z} = \text{diag}(Z_1, \ldots, Z_\kappa, \ldots, Z_k) \in \mathbb{C}^{k \times k}$$

Die drei grundlegenden Gesetze (KU, KI, OG) sind somit:

$$\begin{array}{ll} \text{KI: } \boldsymbol{A} \cdot \boldsymbol{i} = \boldsymbol{0}, & \\ \text{OG:} \quad \boldsymbol{i} - \boldsymbol{I}_0 = \boldsymbol{Y}(\boldsymbol{u} - \boldsymbol{U}_0), & \\ \text{KU:} \quad \boldsymbol{u} = \boldsymbol{A}^T \cdot \boldsymbol{u}_n & \end{array} \tag{5.4}$$

Durch das Einsetzen von KU→OG→KI in (5.4) folgt für das **Knoten-Spannungs-System** allgemein:

$$\boxed{\boldsymbol{A}\boldsymbol{Y}\boldsymbol{A}^T\boldsymbol{u}_n = \boldsymbol{A}(\boldsymbol{Y}\boldsymbol{U}_0 - \boldsymbol{I}_0)} \tag{5.5}$$

Mit den Abkürzungen

$$\boldsymbol{Y}_n = \boldsymbol{A}\boldsymbol{Y}\boldsymbol{A}^T \in \mathbb{C}^{n \times n} \tag{5.6}$$

für die **Knotenadmittanz-Matrix** und

$$\boldsymbol{I}_n = \boldsymbol{A}(\boldsymbol{Y}\boldsymbol{U}_0 - \boldsymbol{I}_0) \in \mathbb{C}^n \tag{5.7}$$

für den **Knotenquellenstrom-Vektor**, folgen dann die Berechnungsvorschriften der Knoten-Spannungs-Analyse ($Z_\kappa \neq 0$ für $\kappa \in \{1, \ldots, k\}$):

$$\boxed{\begin{aligned} \boldsymbol{Y}_n\boldsymbol{u}_n &= \boldsymbol{I}_n, \\ \boldsymbol{u}_n &= \boldsymbol{Y}_n^{-1}\boldsymbol{I}_n, \\ \boldsymbol{u} &= \boldsymbol{A}^T\boldsymbol{u}_n, \\ \boldsymbol{i} &= \boldsymbol{Y}(\boldsymbol{u} - \boldsymbol{U}_0) + \boldsymbol{I}_0 \end{aligned}} \tag{5.8}$$

Beispiel 5.4 *Für das gegebene Netzwerk* ($\rightarrow$ Abb.5.4) *wird mit Hilfe der Gleichungen* (5.6) *und* (5.7) *das Knoten-Spannungs-System aufgestellt.*

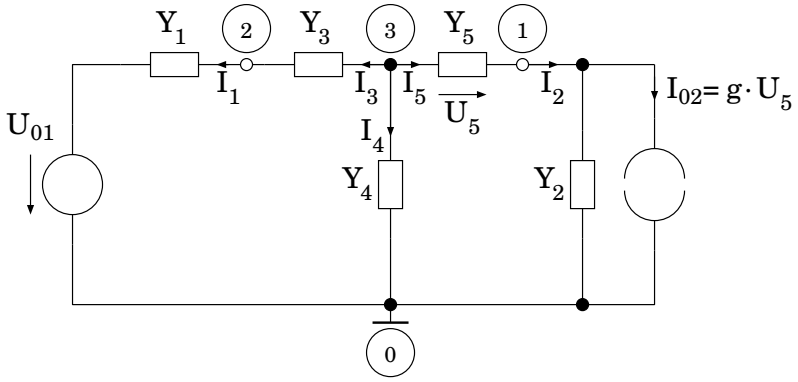


Abb. 5.4: Netzwerk mit spannungsgesteuerter Quelle I_{02}

Die Schaltung enthält eine spannungsgesteuerte Stromquelle I_{02}. Die Behandlung aller Quellen ist identisch. Für gesteuerte Quellen werden lediglich nach dem Aufstellen des Gleichungssystems die steuernden Unbekannten auf die linke Seite gebracht.

Aus dem Netzwerkgraphen ($\rightarrow$ Abb.5.5) *wird zuerst die Inzidenzmatrix* ($\rightarrow$ 5.9) *aufgestellt.*

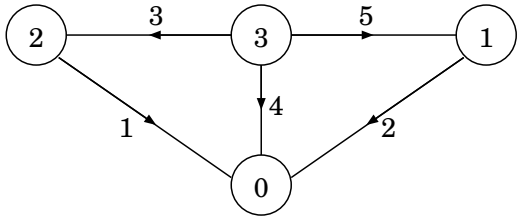


Abb. 5.5: Graph zur Abb.5.4

$$\boldsymbol{A} = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 \end{pmatrix} \tag{5.9}$$

Die Reihenfolge der Knoten bezüglich derjenigen der Zeilen (1, 2, 3) ist wieder 3, 2, 1.

Für die Matrix der Kantenleitwerte gilt

$$\boldsymbol{Y} = \begin{pmatrix} Y_1 & 0 & 0 & 0 & 0 \\ 0 & Y_2 & 0 & 0 & 0 \\ 0 & 0 & Y_3 & 0 & 0 \\ 0 & 0 & 0 & Y_4 & 0 \\ 0 & 0 & 0 & 0 & Y_5 \end{pmatrix}.$$

Die Berechnung von (5.6) *ergibt*

$$\boldsymbol{Y}_n = \boldsymbol{A}\boldsymbol{Y}\boldsymbol{A}^T = \begin{pmatrix} Y_3+Y_4+Y_5 & -Y_3 & -Y_5 \\ -Y_3 & Y_1+Y_3 & 0 \\ -Y_5 & 0 & Y_2+Y_5 \end{pmatrix}. \tag{5.10}$$

Für die Kantenquellenvektoren gilt

$$\boldsymbol{U}_0 = \begin{pmatrix} \boldsymbol{U}_{01} \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad \textit{und} \quad \boldsymbol{I}_0 = \begin{pmatrix} 0 \\ I_{02} \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Damit ergibt sich für (5.7) *mit*

$$\boldsymbol{A}\boldsymbol{Y}\boldsymbol{U}_0 = \begin{pmatrix} 0 \\ Y_1U_{01} \\ 0 \end{pmatrix} \quad \textit{und} \quad \boldsymbol{A}\boldsymbol{I}_0 = \begin{pmatrix} 0 \\ 0 \\ I_{02} \end{pmatrix}$$

$$\boldsymbol{I}_n = \boldsymbol{A}\boldsymbol{Y}\boldsymbol{U}_0 - \boldsymbol{A}\boldsymbol{I}_0 = \begin{pmatrix} 0 \\ Y_1U_{01} \\ -I_{02} \end{pmatrix}. \tag{5.11}$$

Mit

$$I_{02} = g \cdot U_5 = g(U_{n3} - U_{n1}) = g \cdot U_{n3} - g \cdot U_{n1},$$

(5.10) *und* (5.11) *folgt das Knoten-Spannungs-System zu*

$$\begin{pmatrix} Y_3+Y_4+Y_5 & -Y_3 & -Y_5 \\ -Y_3 & Y_1+Y_3 & 0 \\ -Y_5+g & 0 & Y_2+Y_5-g \end{pmatrix} \begin{pmatrix} U_{n3} \\ U_{n2} \\ U_{n1} \end{pmatrix} = \begin{pmatrix} 0 \\ Y_1U_{01} \\ 0 \end{pmatrix}.$$

Kommen keine „Y“-Problemkanten vor, so können aus dem Beispiel die **Bildungsgesetze zum direkten Aufstellen des Knoten-Spannungs-Systems** abgeleitet werden.

a) In der Hauptdiagonalen von $\boldsymbol{Y}_n$ steht in der Zeile i die Summe aller Leitwerte, die am Knoten i anliegen.

b) Das Element $y_{n_{ik}} = y_{n_{ki}}$ ist der Leitwert zwischen den Knoten i und k mit *negativem* Vorzeichen.

c) Das Element i_{n_k} des Knotenquellenstromvektors ist die Summe aller am Knoten k anliegender Stromquellen, wobei einfließende Ströme positiv, herausfließende negativ gezählt werden und aller am Knoten k anliegender Spannungsquellen, multipliziert mit dem Leitwert der Kante, in der die Spannungsquelle liegt, wobei das positive Potential der Spannungsquelle am entsprechenden Knoten positiv gezählt wird.

   Wird die Spannungsquelle mit Innenwiderstand zuvor in eine Stromquelle umgeformt (→ Abb.5.6), so ist die Regel für Spannungsquellen überflüssig.

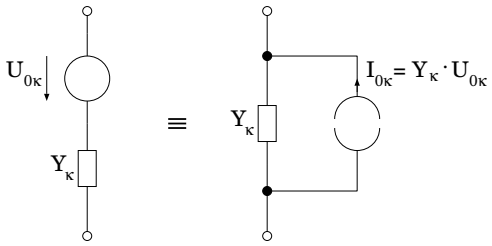
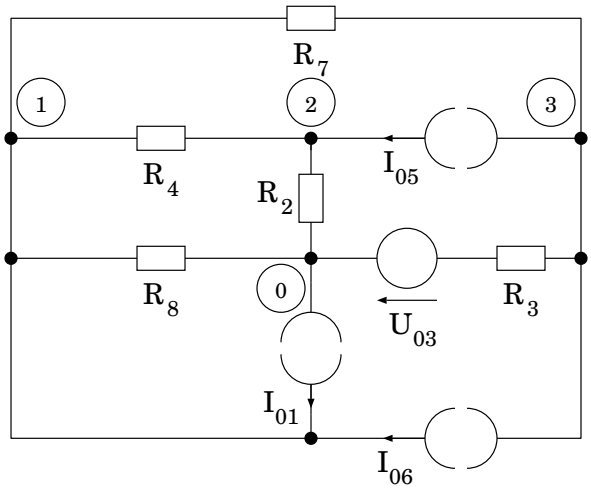


Abb. 5.6: Quellenumwandlung

Beispiel 5.5 *Für das lineare Netzwerk* (→ Abb.5.7) *wird das Knoten-Spannungs-System für die* **Gleichstrom-(DC)-Analyse** *auf direktem Wege aufgestellt.*



$$
\begin{array}{lcl}
R_2 & = & 500\ \Omega \\
R_3 & = & 8\ \mathrm{k}\Omega \\
R_4 & = & 1\ \mathrm{k}\Omega \\
R_7 & = & 2\ \mathrm{k}\Omega \\
R_8 & = & 2\ \mathrm{k}\Omega \\
I_{01} & = & 3\ \mathrm{mA} \\
I_{05} & = & 1\ \mathrm{mA} \\
I_{06} & = & -500\ \mu\mathrm{A} \\
U_{03} & = & 1\ \mathrm{V}
\end{array}
$$

Abb. 5.7: Netzwerk

Um dimensionslos arbeiten zu können, werden zunächst die Werte der Schaltungselemente auf $U_B = 1V$ *und* $R_B = 1k\Omega$ $(I_B = U_B/R_B = 1mA)$ *normiert.*

$$R_N = \frac{R_{phys}}{R_B}; \quad U_N = \frac{U_{phys}}{U_B}; \quad I_N = \frac{I_{phys}}{I_B}$$

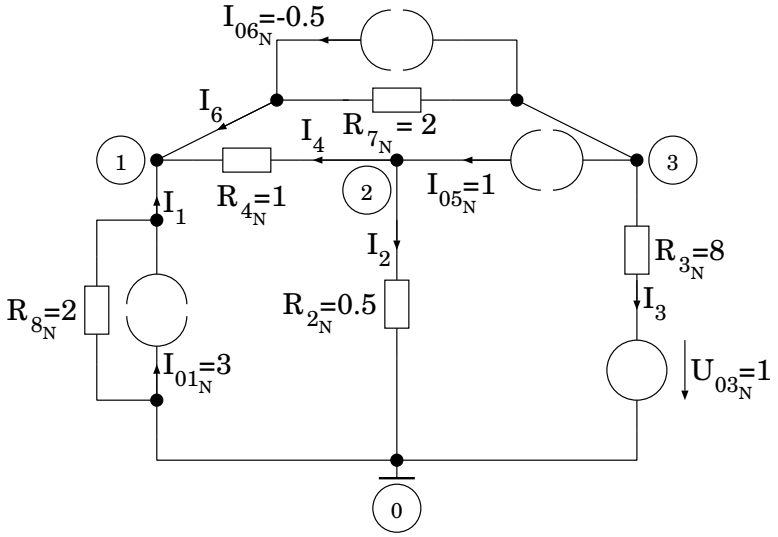


Abb. 5.8: Netzwerk mit normierten Werten

Mit Hilfe der Bildungsgesetze a) *und* b) *folgt für die Knotenadmittanzmatrix* $\boldsymbol{Y}_n$ *(die Knotenreihenfolge entspricht der Zeilenreihenfolge)*

$$\boldsymbol{Y}_n = \begin{pmatrix} 1/R_8 + 1/R_4 + 1/R_7 & -1/R_4 & -1/R_7 \\ -1/R_4 & 1/R_2 + 1/R_4 & 0 \\ -1/R_7 & 0 & 1/R_3 + 1/R_7 \end{pmatrix} = \begin{pmatrix} 2 & -1 & -0.5 \\ -1 & 3 & 0 \\ -0.5 & 0 & 0.625 \end{pmatrix}$$

und mit c) *für den Kantenquellenstromvektor*

$$\boldsymbol{I}_n = \begin{pmatrix} I_{01} + I_{06} \\ I_{05} \\ U_{03}/R_3 - I_{05} - I_{06} \end{pmatrix} = \begin{pmatrix} 2.5 \\ 1 \\ -0.375 \end{pmatrix}.$$

Das Knoten-Spannungs-System ist somit

$$\begin{pmatrix} 1/R_8 + 1/R_4 + 1/R_7 & -1/R_4 & -1/R_7 \\ -1/R_4 & 1/R_2 + 1/R_4 & 0 \\ -1/R_7 & 0 & 1/R_3 + 1/R_7 \end{pmatrix} \begin{pmatrix} U_{n1} \\ U_{n2} \\ U_{n3} \end{pmatrix} = \begin{pmatrix} I_{01} + I_{06} \\ I_{05} \\ U_{03}/R_3 - I_{05} - I_{06} \end{pmatrix}$$

bzw.

$$\begin{pmatrix} 2 & -1 & -0.5 \\ -1 & 3 & 0 \\ -0.5 & 0 & 0.625 \end{pmatrix} \begin{pmatrix} U_{n1} \\ U_{n2} \\ U_{n3} \end{pmatrix} = \begin{pmatrix} 2.5 \\ 1 \\ -0.375 \end{pmatrix}. \tag{5.12}$$

Zur Berechnung der stationären Wechselgrößen eines linearen zeitinvarianten Systems (Linear-Time-Invariant-Sytem: LTI-System) werden die Methoden der Wechselstromrechnung benötigt. Zu ihrer Wiederholung dient das folgende Beispiel.

Beispiel 5.6 *Für eine lineare zeitinvariante Schaltung* (→ Abb.5.9) *soll bei gegebener Eingangserregung*

$$u(t) = \hat{U}\cos(\omega_0 t) \tag{5.13}$$

der Spannungsabfall am Kondensator $u_C(t)$ *berechnet werden. Die Amplitude* $\hat{U}$ *wird dazu sinnvollerweise reell angenommen (keine Phasenverschiebung).*

Für die Schaltung gelten folgende Beziehungen:

$$\begin{aligned} i(t) &= C\,\frac{du_C(t)}{dt} \\ u_R(t) &= i(t)\,R \\ u_L(t) &= L\,\frac{di(t)}{dt} \end{aligned}$$

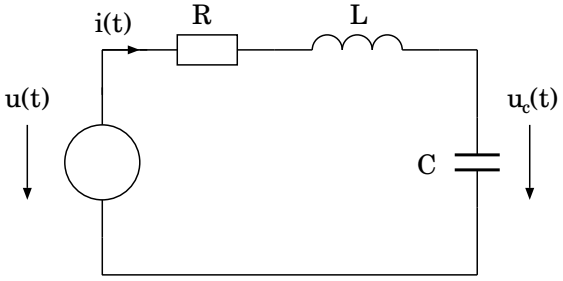


Abbildung 5.9: Serienschwingkreis

Maschengleichung:

$$-u(t) + u_R(t) + u_L(t) + u_C(t) = 0$$

Setzt man in die Maschengleichung die restlichen Beziehungen ein, so folgt

$$-u(t) + i(t)R + L\frac{di(t)}{dt} + u_C(t) = 0,$$

bzw. nach nochmaliger Differentiation

$$-\frac{du(t)}{dt}+\frac{di(t)}{dt}R+L\frac{d^2i(t)}{dt^2}+\frac{du_C(t)}{dt}=0.$$

Der Serienschwingkreis wird somit durch eine gewöhnliche Differentialgleichung (DGL) zweiter Ordnung mit konstanten Koeffizienten beschrieben:

$$L\frac{d^2i(t)}{dt^2}+R\frac{di(t)}{dt}+\frac{1}{C}i(t)=\frac{du(t)}{dt} \tag{5.14}$$

Zur Berechnung der stationären Lösung der Differentialgleichung wird die Eingangsspannung (5.13) *komplex angesetzt:*

$$\underline{u}(t)=\hat{U}e^{j\omega_0 t},\quad \frac{d\underline{u}(t)}{dt}=j\omega_0\hat{U}e^{j\omega_0 t}$$

Der zu berechnende Strom wird ebenfalls komplex geschrieben:

$$\underline{i}(t)=\hat{I}e^{j\varphi_i}e^{j\omega_0 t}=\underline{I}e^{j\omega_0 t} \tag{5.15}$$

Für die Berechnung der beiden Unbekannten, dem Betrag $\hat{I}$ und der Phase φ_i, werden zuerst die Differentialquotienten bestimmt,

$$\frac{d\underline{i}(t)}{dt}=\underline{I}j\omega_0 e^{j\omega_0 t},\quad \frac{d^2\underline{i}(t)}{dt^2}=\underline{I}(j\omega_0)^2e^{j\omega_0 t}$$

und dann in (5.14) *eingesetzt:*

$$L\,\underline{I}(j\omega_0)^2e^{j\omega_0 t}+R\,\underline{I}j\omega_0e^{j\omega_0 t}+\frac{1}{C}\,\underline{I}e^{j\omega_0 t}=j\omega_0\hat{U}e^{j\omega_0 t}$$

Kürzt man nun die Gleichung durch $e^{j\omega_0 t}$ und dividiert durch $j\omega_0$, so folgt

$$j\omega_0 L\,\underline{I}+R\,\underline{I}+\frac{1}{j\omega_0 C}\,\underline{I}=\hat{U}. \tag{5.16}$$

Daraus, daß sich $e^{j\omega_0 t}$ aus (5.16) *herauskürzen läßt, kann man folgern, daß LTI-Systeme zwar die Amplitude, nicht jedoch die Frequenz der Eingangssignale verändern.*

Mit den komplexen Ansätzen der Zeitfunktionen $u(t)$ und $i(t)$ geht die gewöhnliche DGL mit konstanten Koeffizienten (5.14) *in eine* lineare *Gleichung für $\underline{I}$ über. Diese kann sofort nach $\underline{I}$ aufgelöst werden:*

$$\underline{I}=\frac{\hat{U}}{j\omega_0 L+\dfrac{1}{j\omega_0 C}+R}=\frac{\hat{U}}{\underline{Z}} \tag{5.17}$$

komplexer Widerstand		komplexer Leitwert	
Scheinwiderstand (Impedanz):	$\underline{Z}$	Scheinleitwert (Admittanz $\underline{Y}$):	$\frac{1}{\underline{Z}}$
Wirkwiderstand (Resistanz):	R	Wirkleitwert (Konduktanz G):	$\frac{1}{R}$
Blindwiderstand (Reaktanz $\underline{X}_C$):	$\frac{1}{j\omega_0 C}$	Blindleitwert (Suszeptanz $\underline{B}_C$):	$j\omega_0 C$
$\underline{X}_L$:	$j\omega_0 L$	$\underline{B}_L$:	$\frac{1}{j\omega_0 L}$

Zahlenbeispiel: $R = 1\Omega$, $C = 1F$, $L = 1H$, $\hat{U} = 1V$, $\omega_0 = 2s^{-1}$.

Mit diesen Werten folgt für (5.17)

$$\underline{I} = \frac{j2}{-3+j2} = \frac{4-j6}{13} = \frac{2}{\sqrt{13}} \exp\left[j \arctan\frac{(-6)}{4}\right] = \frac{2}{\sqrt{13}} e^{-j0.98}.$$

Durch Realteilbildung erhält man dann aus (5.15)

$$\begin{aligned} i(t) &= \mathrm{Re}\{\underline{i}(t)\} = \mathrm{Re}\{\frac{2}{\sqrt{13}} e^{-j0.98} e^{j2t}\} \\ &= \mathrm{Re}\{\frac{2}{\sqrt{13}}\cos(2t-0.98) + j\frac{2}{\sqrt{13}}\sin(2t-0.98)\} \\ i(t) &= \frac{2}{\sqrt{13}}\cos(2t-0.98). \end{aligned} \tag{5.18}$$

Die Wechselstromrechnung berücksichtigt keine Anfangsbedingungen. Ihre Ergebnisse sind nur für ein stationäres (eingeschwungenes) System richtig. Werden die Anfangsbedingungen berücksichtigt, so folgt für den Strom (5.20)*. Die unterbrochene Linie in* Abb.5.10 *ist die Lösung der Wechseltromrechnung, die durchgezogene stellt die korrekte Lösung dar (dies gilt auch für* Abb.5.11*).*

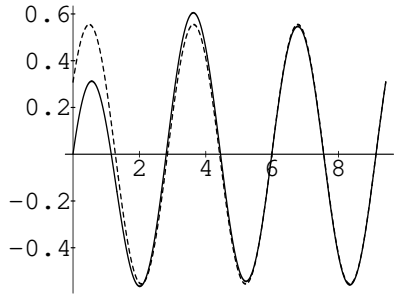


Abb. 5.10: (5.18) und (5.20)

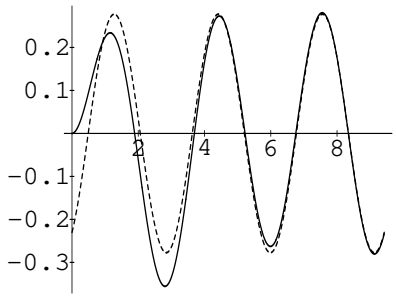


Abb. 5.11: (5.19) und (5.21)

Für die Spannung am Kondensator folgt

$$\underline{U}_C = \underline{I}\frac{1}{j\omega_o C} = \frac{1}{j\sqrt{13}} e^{-j0.98} = \frac{1}{\sqrt{13}} e^{-j(0.98+\frac{\pi}{2})} = \frac{1}{\sqrt{13}} e^{-j2.551} \quad \textit{bzw.}$$

$$\begin{aligned} u_C(t) &= \mathrm{Re}\{\frac{1}{\sqrt{13}} e^{-j2.551} e^{j2t}\} \\ &= \mathrm{Re}\{\frac{1}{\sqrt{13}}\cos(2t-2.551) + j\frac{1}{\sqrt{13}}\sin(2t-2.551)\} \\ u_C(t) &= \frac{1}{\sqrt{13}}\cos(2t-2.551) = \frac{1}{\sqrt{13}}\sin(2t-0.98). \end{aligned} \tag{5.19}$$

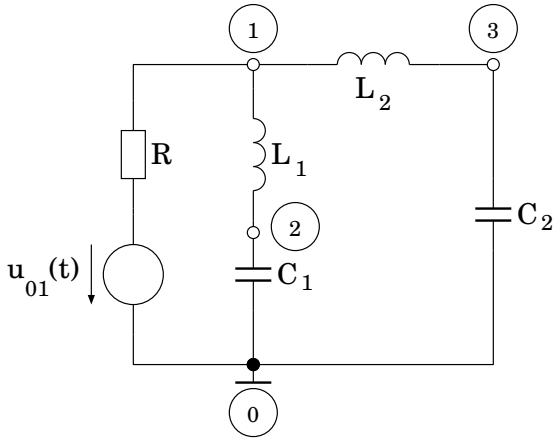
Dies ergibt für den stationären Fall eine Übereinstimmung (→ Abb.5.11) *mit der korrekten Lösung* (5.21).

Die vollständigen Lösungen werden mit Hilfe der Laplace-Transformation berechnet (→ Kap.7.1). *Diese berücksichtigt die Anfangsbedingungen bei der Berechnung der Differentialgleichung* (5.14). *Die berechneten Lösungen enthalten dann die flüchtigen und die stationären Anteile der Signale $i(t)$ und $u_C(t)$. Mit $t \to \infty$ gehen die flüchtigen Anteile gegen Null und es bleiben die stationären Schwingungen* (5.18) *und* (5.19) *übrig.*

$$i(t) = \underbrace{\frac{2}{\sqrt{13}}\cos(2t-0.98)}_{\textit{stationär}} + \underbrace{\frac{2}{\sqrt{39}} e^{-t/2}\cos(\frac{\sqrt{3}}{2}t+2.86)}_{\textit{flüchtig}} \tag{5.20}$$

$$u_C(t) = \underbrace{\frac{1}{\sqrt{13}}\sin(2t-0.98)}_{\textit{stationär}} + \underbrace{\frac{e^{-t/2}}{\sqrt{13}}\left[\sin(\frac{\sqrt{3}}{2}t+2.86) - \frac{1}{\sqrt{3}}\cos(\frac{\sqrt{3}}{2}t+2.86)\right]}_{\textit{flüchtig}} \tag{5.21}$$

Beispiel 5.7 *Für das lineare Netzwerk* (→ Abb.5.12) *wird das Knoten-Spannungs-System für eine* **stationäre Wechselstrom-(AC)-Analyse** *aufgestellt.*



R	=	1 kΩ
L_1	=	1 mH
L_2	=	500 μH
C_1	=	1 nF
C_2	=	2 nF
$u_{01}(t)$	=	4 $\cos(\omega t)$
($\underline{U}_{01}$	=	4 V)
ω	=	$10^6 s^{-1}$

Abb. 5.12: RLC-Netzwerk

Zuerst werden die Werte der Schaltungselemente auf die folgenden Bezugsgrößen normiert.

$$U_B = 1V, \quad R_B = 1k\Omega \quad \textit{und} \quad \omega_B = 10^6 s^{-1}$$

Für den normierten Widerstandswert ergibt sich dann mit

$$R_N = \frac{R_{phys}}{R_B}$$

$R_N = 1$, *für die Spannung mit*

$$\underline{U}_N = \frac{\underline{U}_{phys}}{U_B}$$

$\underline{U}_{01_N} = \hat{U}_{01_N} = 4$, *für die Induktivitäten mit*

$$L_B = \frac{R_B}{\omega_B} \quad \textit{und} \quad L_N = \frac{L_{phys}}{L_B}$$

$L_{1_N} = 1$ *und* $L_{2_N} = 0.5$, *für die Kapazitäten mit*

$$C_B = \frac{1}{R_B \omega_B} \quad \textit{und} \quad C_N = \frac{C_{phys}}{C_B}$$

$C_{1_N} = 1$, $C_{2_N} = 2$ *und für die Kreisfrequenz mit*

$$\omega_N = \frac{\omega_{phys}}{\omega_B}$$

$\omega_N = 1$. *Unter Verwendung der Bildungsgesetze* a)–c) *erhält man schließlich das Knoten-Spannungs-System*

$$\begin{pmatrix} \frac{1}{R} + \frac{1}{j\omega L_1} + \frac{1}{j\omega L_2} & -\frac{1}{j\omega L_1} & -\frac{1}{j\omega L_2} \\ -\frac{1}{j\omega L_1} & \frac{1}{j\omega L_1} + j\omega C_1 & 0 \\ -\frac{1}{j\omega L_2} & 0 & \frac{1}{j\omega L_2} + j\omega C_2 \end{pmatrix} \begin{pmatrix} \underline{U}_{n1} \\ \underline{U}_{n2} \\ \underline{U}_{n3} \end{pmatrix} = \begin{pmatrix} \underline{U}_{01}/R \\ 0 \\ 0 \end{pmatrix}$$

bzw. mit normierten Werten das Gleichungssystem

$$\begin{pmatrix} 1-3j & j & 2j \\ j & 0 & 0 \\ 2j & 0 & 0 \end{pmatrix} \begin{pmatrix} \underline{U}_{n1} \\ \underline{U}_{n2} \\ \underline{U}_{n3} \end{pmatrix} = \begin{pmatrix} 4 \\ 0 \\ 0 \end{pmatrix}. \tag{5.22}$$

Alle bisher behandelten Schaltungen waren frei von „Y“-Problemkanten. Eine „Y“-Problemkante entsteht z.B., wenn eine ideale Spannungsquelle allein in einem Zweig liegt. Neben diesen Kanten mit $Z_\kappa = 0$ kommen häufig solche mit $Z_\kappa \approx 0$ vor.

Bei der Berechnung des Knoten-Spannungs-Systems kann es dann wegen der sehr unterschiedlichen Werte der Matrixelemente zu numerischen Schwierigkeiten und Fehlern kommen. Um dies zu vermeiden, werden Kanten mit $Z_\kappa \cong 0$ getrennt behandelt. Diese Vorgehensweise führt auf das sogenannte erweiterte Knoten-Spannungs-System.

5.3 Das erweiterte Knoten-Spannungs-System

Zur Aufstellung des erweiterten Knoten-Spannungs-Systems werden die Kanten in „Y“-Kanten (Normalfall) und „Z“-Kanten (Problemkanten) aufgeteilt. Die Inzidenzmatrix $\boldsymbol{A}$ und der Kantenstromvektor $\boldsymbol{i}$ werden dazu nach „Y“- und „Z“-Kantenzugehörigkeit sortiert:

$$\boldsymbol{A} = (\, \boldsymbol{A}_Y \boldsymbol{A}_Z \,), \quad \boldsymbol{i} = \begin{pmatrix} \boldsymbol{i}_Y \\ \boldsymbol{i}_Z \end{pmatrix}$$

Für (5.4) folgt damit

$$\begin{aligned} \text{KI: } & \boldsymbol{A}_Y \cdot \boldsymbol{i}_Y & + \boldsymbol{A}_Z \cdot \boldsymbol{i}_Z = \boldsymbol{0}, \\ \text{OG: } & \boldsymbol{i}_Y - \boldsymbol{I}_{0Y} = & \boldsymbol{Y}(\boldsymbol{u}_Y - \boldsymbol{U}_{0Y}), \\ \text{KU: } & & \boldsymbol{u}_Y = \boldsymbol{A}_Y^T \cdot \boldsymbol{u}_n. \end{aligned} \tag{5.23}$$

Setzt man nun KU→OG→KI ein, so ergibt dies

$$\boldsymbol{A}_Y \boldsymbol{Y} \boldsymbol{A}_Y^T \boldsymbol{u}_n + \boldsymbol{A}_Z \cdot \boldsymbol{i}_Z = \boldsymbol{A}_Y(\boldsymbol{Y}\boldsymbol{U}_{0Y} - \boldsymbol{I}_{0Y}). \tag{5.24}$$

Für die „Z“-Kanten gilt

$$\begin{aligned} &\text{OG: } \boldsymbol{u}_Z - \boldsymbol{U}_{0Z} = \boldsymbol{Z}(\boldsymbol{i}_Z - \boldsymbol{I}_{0Z}), \\ &\text{KU: } \boldsymbol{u}_Z = \boldsymbol{A}_Z^T \cdot \boldsymbol{u}_n. \end{aligned} \tag{5.25}$$

Mit KU→OG folgt

$$\boldsymbol{A}_Z^T \cdot \boldsymbol{u}_n - \boldsymbol{Z} \cdot \boldsymbol{i}_Z = \boldsymbol{U}_{0Z} - \boldsymbol{Z} \cdot \boldsymbol{I}_{0Z}. \tag{5.26}$$

Die Gleichungen (5.24) und (5.26) können nun zum erweiterten Knoten-Spannungs-System zusammengefaßt werden:

$$\begin{pmatrix} \boldsymbol{A}_Y \boldsymbol{Y} \boldsymbol{A}_Y^T & \boldsymbol{A}_Z \\ \boldsymbol{A}_Z^T & -\boldsymbol{Z} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_n \\ \boldsymbol{i}_Z \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}_Y(\boldsymbol{Y}\boldsymbol{U}_{0Y} - \boldsymbol{I}_{0Y}) \\ \boldsymbol{U}_{0Z} - \boldsymbol{Z} \cdot \boldsymbol{I}_{0Z} \end{pmatrix} \tag{5.27}$$

Mit den Definitionen (5.6) und (5.7) folgt einfacher

$$\boxed{\begin{pmatrix} \boldsymbol{Y}_n & \boldsymbol{A}_Z \\ \boldsymbol{A}_Z^T & -\boldsymbol{Z} \end{pmatrix} \begin{pmatrix} \boldsymbol{u}_n \\ \boldsymbol{i}_Z \end{pmatrix} = \begin{pmatrix} \boldsymbol{I}_n \\ \boldsymbol{U}_{0Z} - \boldsymbol{Z} \cdot \boldsymbol{I}_{0Z} \end{pmatrix}} \tag{5.28}$$

Beispiel 5.8 *Die gegebene Schaltung* (→ Abb.5.13) *enthält eine Spannungsquelle ohne Innenwiderstand.*

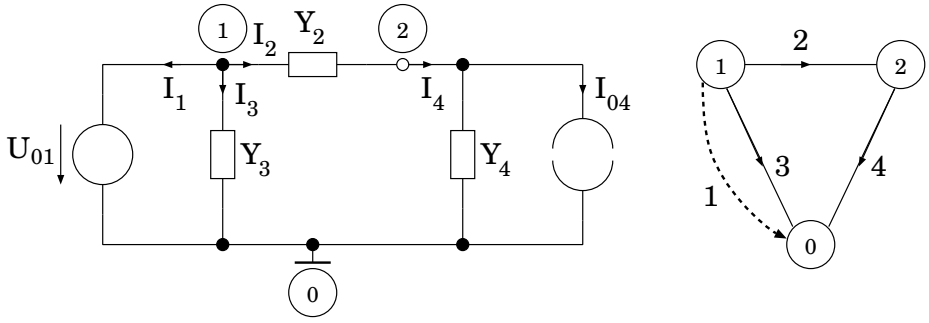


Abb. 5.13: Schaltnetz mit Problemkante (Kante 1)

Für die Problemkante 1 ist die Inzidenzmatrix (Knoten- gleich Zeilenreihenfolge)

$$\boldsymbol{A}_Z = \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \tag{5.29}$$

Ohne Berücksichtigung der Problemkanten folgt für die Knotenadmittanzmatrix

$$\boldsymbol{Y}_n = \begin{pmatrix} Y_2+Y_3 & -Y_2 \\ -Y_2 & Y_2+Y_4 \end{pmatrix}, \textit{ für } \boldsymbol{Z}=0. \tag{5.30}$$

Weiter gilt

$$\boldsymbol{I}_n = \begin{pmatrix} 0 \\ -I_{04} \end{pmatrix} \textit{ und } \boldsymbol{U}_{0Z} - \boldsymbol{Z}\cdot I_{0Z} = U_{01}. \tag{5.31}$$

Mit (5.29), (5.30) *und* (5.31) *folgt für das erweiterte Knoten-Spannungs-System:*

$$\begin{pmatrix} Y_2+Y_3 & -Y_2 & 1 \\ -Y_2 & Y_2+Y_4 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} U_{n1} \\ U_{n2} \\ I_1 \end{pmatrix} = \begin{pmatrix} 0 \\ -I_{04} \\ U_{01} \end{pmatrix} \tag{5.32}$$

Beispiel 5.9 *Für das gegebene Netzwerk* ($\rightarrow$ Abb.5.13) *wird das erweiterte Knotenspannungssytem aufgestellt. Die Problemkante mit* $Z_5 = 0$ *ist Kante 5.*

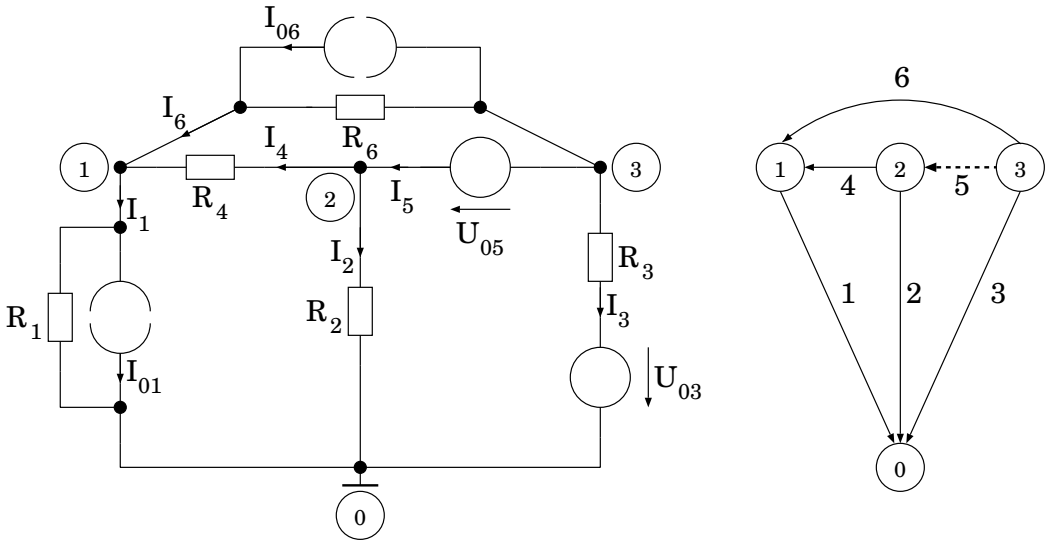


Abb. 5.14: Schaltnetz und Graph mit Problemkante (Kante 5)

Für die Inzidenzmatrix (Knoten- gleich Zeilenreihenfolge) der Problemkanten folgt damit

$$\boldsymbol{A}_Z = \begin{pmatrix} 0 \\ -1 \\ 1 \end{pmatrix}. \tag{5.33}$$

Die Knotenadmittanzmatrix der Leitwerte ergibt

$$\boldsymbol{Y}_n = \begin{pmatrix} Y_1+Y_4+Y_6 & -Y_4 & -Y_6 \\ -Y_4 & Y_2+Y_4 & 0 \\ -Y_6 & 0 & Y_3+Y_6 \end{pmatrix}, \textit{ die Widerstandsmatrix } \boldsymbol{Z}=0. \tag{5.34}$$

Mit

$$\boldsymbol{I}_n = \begin{pmatrix} -I_{01}+I_{06} \\ 0 \\ U_{03}\cdot Y_3 - I_{06} \end{pmatrix} \textit{ bzw. } \boldsymbol{U}_{0Z} - \boldsymbol{Z}\cdot I_{0Z} = U_{05},$$

(5.33) *und* (5.34) *folgt*

$$\begin{pmatrix} Y_1+Y_4+Y_6 & -Y_4 & -Y_6 & 0 \\ -Y_4 & Y_2+Y_4 & 0 & -1 \\ -Y_6 & 0 & Y_3+Y_6 & 1 \\ 0 & -1 & 1 & 0 \end{pmatrix} \begin{pmatrix} U_{n1} \\ U_{n2} \\ U_{n3} \\ I_5 \end{pmatrix} = \begin{pmatrix} -I_{01}+I_{06} \\ 0 \\ U_{03}\cdot Y_3 - I_{06} \\ U_{05} \end{pmatrix}. \tag{5.35}$$

Nimmt man weiter an, daß die Spannung U_{05} z.B. eine stromgesteuerte Spannungsquelle ist,

$$U_{05} = r \cdot I_4,$$

so folgt wegen

$$U_{05} = r\frac{U_{n2}-U_{n1}}{R_4}$$

für das erweiterte Knoten-Spannungs-System:

$$\begin{pmatrix} Y_1+Y_4+Y_6 & -Y_4 & -Y_6 & 0 \\ -Y_4 & Y_2+Y_4 & 0 & -1 \\ -Y_6 & 0 & Y_3+Y_6 & 1 \\ r\cdot Y_4 & -1-r\cdot Y_4 & 1 & 0 \end{pmatrix} \begin{pmatrix} U_{n1} \\ U_{n2} \\ U_{n3} \\ I_5 \end{pmatrix} = \begin{pmatrix} -I_{01}+I_{06} \\ 0 \\ U_{03}\cdot Y_3 - I_{06} \\ 0 \end{pmatrix} \tag{5.36}$$

Beispiel 5.10 *Im Schaltnetz aus* Abb.5.14 *werden drei Kanten zu Z-Kanten gemacht. Neben der Problemkante 5 (U_{05}) sollen dies die Kanten 1 und 4 sein. Dies ist z.B. dann sinnvoll, wenn R_1 und R_4 sehr klein sind. Für die Inzidenzmatrix folgt dann (aufsteigende Kantenreihenfolge)*

$$\boldsymbol{A}_Z = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix}.$$

Die Widerstandsmatrix ist

$$\boldsymbol{Z} = \begin{pmatrix} R_1 & 0 & 0 \\ 0 & R_4 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

und für die Strom- und Spannungsquellen folgt

$$\boldsymbol{U}_{0Z} = \begin{pmatrix} 0 \\ 0 \\ U_{05} \end{pmatrix} \quad \textit{und} \quad \boldsymbol{I}_{0Z} = \begin{pmatrix} I_{01} \\ 0 \\ 0 \end{pmatrix}.$$

Mit

$$\boldsymbol{U}_{0Z} - \boldsymbol{Z}\cdot\boldsymbol{I}_{0Z} = \begin{pmatrix} -R_1\cdot I_{01} \\ 0 \\ U_{05} \end{pmatrix}$$

ergibt sich für das erweiterte Knoten-Spannungs-System:

$$\begin{pmatrix} Y_6 & 0 & -Y_6 & 1 & -1 & 0 \\ 0 & Y_2 & 0 & 0 & 1 & -1 \\ -Y_6 & 0 & Y_3+Y_6 & 0 & 0 & 1 \\ 1 & 0 & 0 & -R_1 & 0 & 0 \\ -1 & 1 & 0 & 0 & -R_4 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} U_{n1} \\ U_{n2} \\ U_{n3} \\ I_1 \\ I_4 \\ I_5 \end{pmatrix} = \begin{pmatrix} I_{06} \\ 0 \\ U_{03}\cdot Y_3 - I_{06} \\ -R_1\cdot I_{01} \\ 0 \\ U_{05} \end{pmatrix}$$

5.4 Der Gauß-Algorithmus

Für ein inhomogenes Gleichungssystem der Form

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}, \tag{5.37}$$

bzw.,

$$\boldsymbol{A} \cdot \boldsymbol{x} = \boldsymbol{b}, \tag{5.38}$$

wird die Lösung $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ gesucht ($\boldsymbol{A}$ ist hier nicht die Inzidenzmatrix, n nicht die Anzahl der Knoten!).

Ein nach Carl Friedrich Gauss benanntes Lösungsverfahren geht davon aus, daß

a) bei einer Vertauschung zweier Zeilen,

b) einer Multiplikation einer Gleichung mit einer Zahl $\neq 0$ und

c) einer Addition (bzw. Subtraktion) eines Vielfachen einer Gleichung zu (bzw. von) einer anderen

das ursprüngliche Gleichungssystem in ein dazu äquivalentes übergeht. Da diese Umformungen auf dieselbe Art und Weise wieder rückgängig gemacht werden können, ist die Lösungsmenge des umgeformten mit derjenigen von (5.37) identisch.

Alle Umformungen erfolgen zeilenweise, es ist daher sinnvoll, $\boldsymbol{A}$ und $\boldsymbol{b}$ zuvor zu einer *erweiterten Koeffizientenmatrix* $(\boldsymbol{A}\,\boldsymbol{b})$ zusammenzufassen:

$$(\boldsymbol{A}\,\boldsymbol{b})^{(1)} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ a_{21}^{(1)} & a_{22}^{(1)} & \cdots & a_{2n}^{(1)} & b_2^{(1)} \\ \vdots & \vdots & & \vdots & \vdots \\ a_{n1}^{(1)} & a_{n2}^{(1)} & \cdots & a_{nn}^{(1)} & b_n^{(1)} \end{pmatrix} \tag{5.39}$$

Der Index (1) kennzeichnet das ursprüngliche Gleichungssystem vor dem ersten Eliminationsschritt.

Die *Gauß'sche Elimination* besteht aus zwei Schritten,

a) einer **Vorwärtselimination** und

b) einer **Rückwärtssubstitution**.

Zur **Vorwärtselimination** bringt man durch eventuelle Zeilenvertauschung eine Zahl $\neq 0$ an die erste Stelle der ersten Spalte und macht anschließend die darunterstehenden Zahlen durch die Subtraktion des $(a_{21}^{(1)}/a_{11}^{(1)})$-fachen der ersten Zeile von der zweiten, des $(a_{31}^{(1)}/a_{11}^{(1)})$-fachen der ersten Zeile von der dritten, usw., zu Null. Das resultierende Gleichungssystem ist dann

$$\begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} & b_n^{(2)} \end{pmatrix} = \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & & & & \\ \vdots & & (\boldsymbol{A}_1\,\boldsymbol{b}_1)^{(2)} & & \\ 0 & & & & \end{pmatrix}$$

Die gleichen Umformungen werden nun in $(\boldsymbol{A}_1\,\boldsymbol{b}_1)^{(2)}$ durchgeführt.

Für $\det \boldsymbol{A} \neq 0$ ist die Vorwärtselimination nach $n-1$ Schritten beendet.

$$(\boldsymbol{A}\,\boldsymbol{b})^{(1)} \to (\boldsymbol{A}_1\,\boldsymbol{b}_1)^{(2)} \to \cdots \to (\boldsymbol{A}_{n-2}\,\boldsymbol{b}_{n-2})^{(n-1)} \to (\boldsymbol{A}_{n-1}\,\boldsymbol{b}_{n-1})^{(n)} = (a_{nn}^{(n)}\,b_n^{(n)})^{(n)}$$

Zur **Rücksubstitution** wird zuerst aus

$$x_n \cdot a_{nn}^{(n)} = b_n^{(n)}$$

x_n, dann aus

$$x_{n-1} \cdot a_{n-1,n-1}^{(n-1)} + x_n \cdot a_{n-1,n}^{(n-1)} = b_{n-1}^{(n-1)}$$

x_{n-1}, usw., bestimmt. Die Lösung $\boldsymbol{x}$ ist nach n Schritten berechnet.

Beispiel 5.11 *Für das Knoten-Spannungs-System* (5.12) *von Seite 97 wird mit Hilfe des Gauß-Algorithmus die Lösung* $\boldsymbol{U}_n$ *berechnet. Die erweiterte Koeffizientenmatrix ist*

$$\begin{pmatrix} 2 & -1 & -0.5 & 2.5 \\ -1 & 3 & 0 & 1 \\ -0.5 & 0 & 0.625 & -0.375 \end{pmatrix}.$$

Durch die Subtraktion des $(-1/2)$*-fachen der ersten Zeile* ($a_{11}^{(1)} = 2$ *und* $a_{21}^{(1)} = -1$) *von der zweiten und des* $(-0.5/2)$*-fachen der ersten von der dritten wird die Größe* U_{n1} *eliminiert:*

$$\begin{pmatrix} 2 & -1 & -0.5 & 2.5 \\ -1-\left(\frac{-1}{2}\right)\cdot(2) & 3-\left(\frac{-1}{2}\right)\cdot(-1) & 0-\left(\frac{-1}{2}\right)\cdot(-0.5) & 1-\left(\frac{-1}{2}\right)\cdot(2.5) \\ -0.5-\left(\frac{-0.5}{2}\right)\cdot(2) & 0-\left(\frac{-0.5}{2}\right)\cdot(-1) & 0.625-\left(\frac{-0.5}{2}\right)\cdot(-0.5) & -0.375-\left(\frac{-0.5}{2}\right)\cdot(2.5) \end{pmatrix}$$

$$= \begin{pmatrix} 2 & -1 & -0.5 & 2.5 \\ 0 & 2.5 & -0.25 & 2.25 \\ 0 & -0.25 & 0.5 & 0.25 \end{pmatrix}$$

Die gleiche Vorgehensweise angewendet auf

$$(\boldsymbol{A}_1\,\boldsymbol{b}_1)^{(2)} = \begin{pmatrix} 2.5 & -0.25 & 2.25 \\ -0.25 & 0.5 & 0.25 \end{pmatrix}$$

eliminiert U_{n2}*:*

$$\begin{pmatrix} 2 & -1 & -0.5 & 2.5 \\ 0 & 2.5 & -0.25 & 2.25 \\ 0 & -0.25-\left(\frac{-0.25}{2.5}\right)\cdot(2.5) & 0.5-\left(\frac{-0.25}{2.5}\right)\cdot(-0.25) & 0.25-\left(\frac{-0.25}{2.5}\right)\cdot(2.25) \end{pmatrix}$$

$$= \begin{pmatrix} 2 & -1 & -0.5 & 2.5 \\ 0 & 2.5 & -0.25 & 2.25 \\ 0 & 0 & 0.475 & 0.475 \end{pmatrix} \tag{5.40}$$

Durch Rücksubstitution erhält man

$$0.475 \cdot U_{n3} = 0.475 \quad \textit{bzw.} \quad U_{n3} = 1,$$

$$2.5 \cdot U_{n2} - 0.25 \cdot 1 = 2.25 \quad \textit{bzw.} \quad U_{n2} = 1 \quad \textit{und}$$
$$2 \cdot U_{n1} - 1 \cdot 1 - 0.5 \cdot 1 = 2.5 \quad \textit{bzw.} \quad U_{n1} = 2.$$

Die Lösung des Gleichungssystems lautet somit:

$$\boldsymbol{U}_n = \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix}$$

Der Gauß-Algorithmus erzeugt durch eine Reihe von Zeilenumformungen aus $\boldsymbol{A} \in \mathbb{C}^{n\times n}$ eine Matrix in rechter oberer Dreiecksform (z.B. (5.40)). Entstehen durch diese elementaren Zeilenumformungen keine Nullzeilen, so sind die Zeilen von $\boldsymbol{A}$ linear unabhängig und der **Rang** der Matrix ist

$$\operatorname{rang}(\boldsymbol{A}) = n.$$

Treten durch die Umformungen in $\boldsymbol{A}$ eine oder mehrere Nullzeilen auf, so gilt

$$\operatorname{rang}(\boldsymbol{A}) = r < n.$$

Ist gleichzeitig

$$\operatorname{rang}(\boldsymbol{A}\,\boldsymbol{b}) \neq \operatorname{rang}(\boldsymbol{A}),$$

so hat das Gleichungssystem $\boldsymbol{A} \cdot \boldsymbol{x} = \boldsymbol{b}$ keine Lösung, es ist **inkonsistent** (es gibt z.B. kein x_n, für das die Gleichung $0 \cdot x_n = b_n^{(n)}$ erfüllt wird).

Ist

$$\operatorname{rang}(\boldsymbol{A}\,\boldsymbol{b}) = \operatorname{rang}(\boldsymbol{A}) = r < n,$$

so gibt es $n - r$ frei wählbare Variable (sogenannte **freie Variable**), bzw. eine $(n-r)$-dimensionale **Lösungsmannigfaltigkeit**.

Die Gauß-Elimination kann folglich ein eindeutiges, ein mehrdeutiges oder gar kein Ergebnis liefern. Die Lösungsstrukturen in (5.41) bzw. (5.42) enthalten alle diese Möglichkeiten.

$$\begin{pmatrix} a_{11}^{(1)} & \cdots & \cdots & a_{1r}^{(1)} & a_{1,r+1}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2r}^{(2)} & a_{2,r+1}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & & \ddots & \vdots & \vdots & & \vdots \\ 0 & \cdots & 0 & a_{rr}^{(r)} & a_{r,r+1}^{(r)} & \cdots & a_{rn}^{(r)} \\ 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & & & \vdots & \vdots & & \vdots \\ 0 & \cdots & \cdots & 0 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_r \\ x_{r+1} \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \vdots \\ b_r^{(r)} \\ b_{r+1}^{(r)} \\ \vdots \\ b_n^{(r)} \end{pmatrix} \tag{5.41}$$

$$\begin{pmatrix} \boldsymbol{R}_{rr} & \boldsymbol{A}_{r,n-r} \\ \boldsymbol{0}_{n-r,r} & \boldsymbol{0}_{n-r,n-r} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_r \\ \boldsymbol{x}_{n-r} \end{pmatrix} = \begin{pmatrix} \boldsymbol{b}_r \\ \boldsymbol{b}_{n-r} \end{pmatrix} \tag{5.42}$$

Die tiefergestellten Indizes in (5.42) geben jeweils die Anzahl der Zeilen (erster) und Spalten (zweiter) an. Ist die **Lösbarkeitsbedingung**

$$\boldsymbol{b}_{n-r} = \boldsymbol{0} \quad \text{bzw.} \quad \operatorname{rang}(\boldsymbol{A}\,\boldsymbol{b}) = \operatorname{rang}(\boldsymbol{A})$$

erfüllt, so kann $\boldsymbol{A} \cdot \boldsymbol{x} = \boldsymbol{b}$ vier unterschiedliche Lösungen haben:

a) $r = n$; $\boldsymbol{b} \neq \boldsymbol{0}$: inhomogenes Gleichungssystem, $\text{rang}(\boldsymbol{A}) = n$, $\det(\boldsymbol{A}) \neq 0$.

$$\boldsymbol{R}_{nn} \cdot \boldsymbol{x}_n = \boldsymbol{b}_n$$

Die Lösung $\boldsymbol{x}_n = \boldsymbol{R}^{-1} \cdot \boldsymbol{b}$ wird durch Rücksubstitution (Beispiel 5.11) bestimmt.

b) $r = n$; $\boldsymbol{b} = \boldsymbol{0}$: homogenes Gleichungssystem, $\text{rang}(\boldsymbol{A}) = n$, $\det(\boldsymbol{A}) \neq 0$.

$$\boldsymbol{R}_{nn} \cdot \boldsymbol{x}_n = \boldsymbol{0}$$

Die einzige Lösung ist $\boldsymbol{x}_n = \boldsymbol{0}$.

c) $r < n$; $\boldsymbol{b} \neq \boldsymbol{0}$: inhomogenes Gleichungssystem, $\text{rang}(\boldsymbol{A}) < n$, $\det(\boldsymbol{A}) = 0$.

$$\boldsymbol{R}_{rr} \cdot \boldsymbol{x}_r = -\boldsymbol{A}_{r,n-r} \cdot \boldsymbol{x}_{n-r} + \boldsymbol{b}_r, \ \det(\boldsymbol{R}_{rr}) \neq 0$$

Die allgemeine Lösung berechnet sich zu

$$\boldsymbol{x}_r = -\boldsymbol{R}_{rr}^{-1} \cdot \boldsymbol{A}_{r,n-r} \cdot \boldsymbol{x}_{n-r} + \boldsymbol{R}_{rr}^{-1} \cdot \boldsymbol{b}_r.$$

Die $n-r$ freien Variablen in $\boldsymbol{x}_{n-r}$ können beliebig gewählt werden. Die Festlegung von $\boldsymbol{x}_{n-r}$ führt auf eine **spezielle** (**partikuläre**) Lösung, z.B.

$$\boldsymbol{x}_r = \boldsymbol{R}_{rr}^{-1} \cdot \boldsymbol{b}_r \quad \text{für} \quad \boldsymbol{x}_{n-r} = \boldsymbol{0}.$$

d) $r < n$; $\boldsymbol{b} = \boldsymbol{0}$: homogenes Gleichungssystem, $\text{rang}(\boldsymbol{A}) < n$, $\det(\boldsymbol{A}) = 0$.

$$\boldsymbol{R}_{rr} \cdot \boldsymbol{x}_r = -\boldsymbol{A}_{r,n-r} \cdot \boldsymbol{x}_{n-r}, \ \det(\boldsymbol{R}_{rr}) \neq 0$$

Für die allgemeine Lösung folgt

$$\boldsymbol{x}_r = -\boldsymbol{R}_{rr}^{-1} \cdot \boldsymbol{A}_{r,n-r} \cdot \boldsymbol{x}_{n-r},$$

die $n - r$ freien Variablen in $\boldsymbol{x}_{n-r}$ können wiederum beliebig gewählt werden.

Beispiel 5.12 *Das Knoten-Spannungs-System* (5.22) *von Seite 101 wird mit dem Gauß-Verfahren gelöst. Die erweiterte Koeffizientenmatrix ist*

$$\begin{pmatrix} 1-3j & j & 2j & 4 \\ j & 0 & 0 & 0 \\ 2j & 0 & 0 & 0 \end{pmatrix}.$$

Der erste Eliminationsschritt ergibt

$$\begin{pmatrix} 1-3j & j & 2j & 4 \\ j - \left(\frac{j}{1-3j}\right)\cdot(1-3j) & 0 - \left(\frac{j}{1-3j}\right)\cdot(j) & 0 - \left(\frac{j}{1-3j}\right)\cdot(2j) & 0 - \left(\frac{j}{1-3j}\right)\cdot(4) \\ 2j - \left(\frac{2j}{1-3j}\right)\cdot(1-3j) & 0 - \left(\frac{2j}{1-3j}\right)\cdot(j) & 0 - \left(\frac{2j}{1-3j}\right)\cdot(2j) & 0 - \left(\frac{2j}{1-3j}\right)\cdot(4) \end{pmatrix}$$

$$= \begin{pmatrix} 1-3j & j & 2j & 4 \\ 0 & 0.1+0.3j & 0.2+0.6j & 1.2-0.4j \\ 0 & 0.2+0.6j & 0.4+1.2j & 2.4-0.8j \end{pmatrix},$$

der zweite

$$\begin{pmatrix} 1-3j & j & 2j & 4 \\ 0 & 0.1+0.3j & 0.2+0.6j & 1.2-0.4j \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Durch den Rangabfall von 1 *entsteht eine* 1-dimensionale Lösungsmannigfaltigkeit*, die Variable* $\underline{U}_{n3}$ *ist daher* frei wählbar*:*

$$\underline{U}_{n3} = \lambda$$

Für $\underline{U}_{n2}$ *in Abhängigkeit von* λ *erhält man dann*

$$\begin{aligned} \underline{U}_{n2} &= \frac{1}{0.1+0.3j}\left[1.2-0.4j-\lambda(0.2+0.6j)\right] \\ &= -4j-2\lambda \end{aligned}$$

und für $\underline{U}_{n1}$

$$\begin{aligned} \underline{U}_{n1} &= \frac{1}{1-3j}\left[(4j+2\lambda)j-2j\lambda+4\right] \\ &= 0. \end{aligned}$$

Die Lösung des Gleichungssystems lautet somit

$$\underline{\boldsymbol{U}}_n = j\begin{pmatrix} 0 \\ -4 \\ 0 \end{pmatrix} + \lambda\begin{pmatrix} 0 \\ -2 \\ 1 \end{pmatrix}.$$

Da $\underline{U}_{n1} = 0$ *ist, sind beide Serienschwingkreise in Resonanz. Der Strom durch* R *teilt sich dabei auf beide Schwingkreise auf. Über die Werte der Teilströme ist keine Aussage möglich.*

6 Analyse nichtlinearer resistiver Netzwerke

Ein großer Teil der Elemente elektrischer Schaltungen ist nichtlinear. Ein nichtlineares Gleichungssystem ist in der Regel nicht geschlossen sondern nur iterativ lösbar. Ein Verfahren zur iterativen Berechnung nichtlinearer Gleichungssysteme ist das **Newton-Verfahren**. Die Anwendung des Verfahrens auf nichtlineare Schaltungen wird im folgenden beschrieben.

6.1 Das Newton-Verfahren

Gesucht ist die Nullstelle x^* einer nichtlinearen Funktion $f(x)$,

$$f(x^*) = 0. \tag{6.1}$$

Dazu legt man in einem Punkt $x^{(0)}$ die Tangente

$$\bar{f}(x) = f(x^{(0)}) + f'(x^{(0)})(x - x^{(0)})$$

an $f(x)$ und berechnet deren Nullstelle ($\rightarrow$ Abb.6.1):

$$f(x^{(0)}) + f'(x^{(0)})(x^{(1)} - x^{(0)}) = 0 \Rightarrow x^{(1)} = x^{(0)} - \frac{f(x^{(0)})}{f'(x^{(0)})}$$

Im nächsten Schritt wird der gleiche Vorgang im Punkt $x^{(1)}$ wiederholt ($\rightarrow$ Abb.6.1):

$$f(x^{(1)}) + f'(x^{(1)})(x^{(2)} - x^{(1)}) = 0 \Rightarrow x^{(2)} = x^{(1)} - \frac{f(x^{(1)})}{f'(x^{(1)})}$$

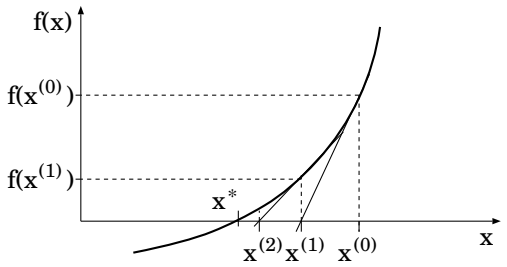


Abb. 6.1: Newton-Iteration

Die Iteration wird so lange fortgeführt, bis ein zu Beginn festgelegtes Abbruchkriterium ε erreicht ist:

$$|x^{(\mu)} - x^{(\mu-1)}| < \varepsilon$$

Der zuletzt berechnete Wert $x^{(\mu)}$ stimmt dann mit einer durch ε festgelegten Anzahl von Stellen mit der gesuchten Lösung x^* überein.

Das Newton-Verfahren hat folgende Iterationsvorschrift:

$$\boxed{x^{(\mu+1)} = x^{(\mu)} - \frac{f(x^{(\mu)})}{f'(x^{(\mu)})};\ \mu = 0, 1, \ldots} \tag{6.2}$$

Die durch (6.2) erzeugte Folge $(x^{(\mu)})_{\mu \geq 0}$ konvergiert sicher für jeden Startwert $x^{(0)} \in I = [a, b]$ monoton gegen x^*, wenn die nachfolgenden Bedingungen erfüllt sind. Diese Bedingungen sind hinreichend, jedoch nicht notwendig. Das Verfahren kann auch dann konvergieren, wenn eine oder mehrere Forderungen verletzt werden.

Konvergenzbedingungen:

a) $f(a) \cdot f(b) < 0$.

Die Nullstelle muß von a und b eingeschlossen werden ($\rightarrow$ Abb.6.1).

b) $f'(x) \neq 0$.

Im Intervall darf die Funktion weder einen Wendepunkt noch ein Maximum oder Minimum besitzen, da sonst die Tangente in diesem Punkt nur im Unendlichen einen Schnittpunkt mit der Abszisse hat (Division durch Null in (6.2)). Die Funktion muß im Intervall monoton verlaufen. Monotonie allein reicht jedoch nicht ($\rightarrow$ Abb.6.4b). Es wird zusätzlich Konvexität oder Konkavität gefordert.

c) α) $f''(x) \geq 0$ für $x \in I$ (Konvexität $\rightarrow$ Abb.6.2) oder

β) $f''(x) \leq 0$ für $x \in I$ (Konkavität $\rightarrow$ Abb.6.3).

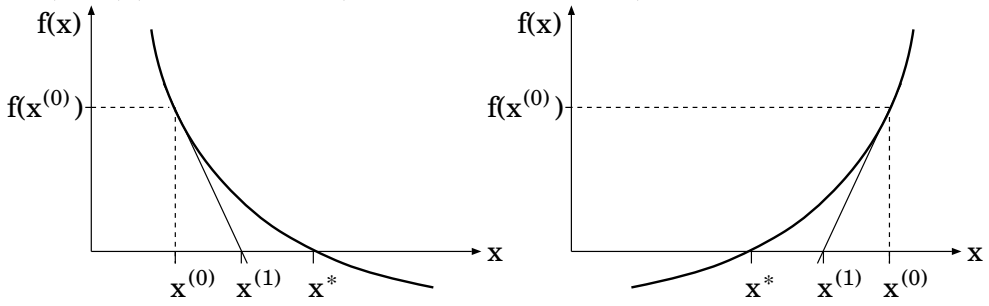


Abb. 6.2: Konvexer Verlauf der Funktion in I

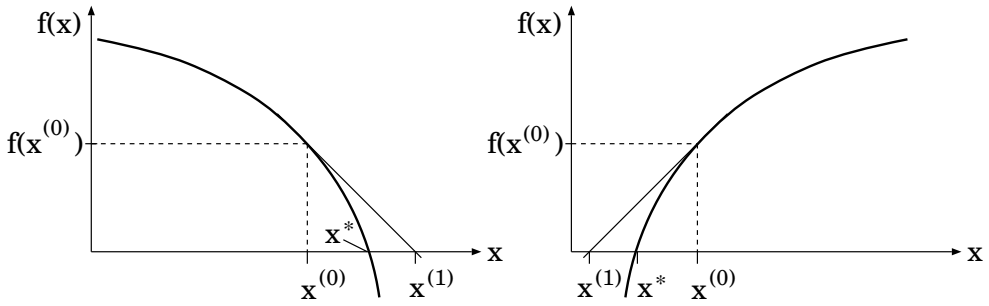


Abb. 6.3: Konkaver Verlauf der Funktion in I

Wird für eine konvexe Funktion der Startwert so gewählt, daß $f(x^{(0)}) > 0$ ist ($\rightarrow$ Abb.6.2), so konvergiert die Folge $(x^{(\mu)})_{\mu \geq 0}$ monoton gegen x^*.

Für eine konkave Funktion gilt entsprechend $f(x^{(0)}) < 0$.

Sind die Vorzeichen von $f''(x)$ und $f(x^{(0)})$ verschieden ($\rightarrow$ Abb.6.3), so „springt“ das Verfahren über x^* hinweg. (Ist die Konvexität bzw. Konkavität nicht für das gesamte Intervall I gesichert, so ist nach solch einem Sprung (z.B. über eine Polstelle hinweg) die Konvergenz fraglich.)

d) $\left|\dfrac{f(x)}{f'(x)}\right| \leq b-a.$

Diese Forderung kann immer erfüllt werden, es ist $b-a$ nur entsprechend klein zu wählen.

Die **Konvergenzzuordnung** des Verfahrens wird mit Hilfe der Taylorreihe der Funktion in $x^{(\mu)}$,

$$f(x^*) = f(x^{(\mu)}) + f'(x^{(\mu)})(x^* - x^{(\mu)}) + \frac{1}{2}f''(\xi)(x^* - x^{(\mu)})^2 = 0;\ \xi \in [x^{(\mu)}, x^*], \quad (6.3)$$

und der Nullstelle der Tangente in $x^{(\mu)}$,

$$f(x^{(\mu)}) + f'(x^{(\mu)})(x^{(\mu+1)} - x^{(\mu)}) = 0, \quad (6.4)$$

bestimmt. Mit dem Lösungsfehler

$$\varepsilon^{(\mu+1)} = (x^{(\mu+1)} - x^*) \quad (6.5)$$

und der Subtraktion der Gleichung (6.4) von (6.3),

$$f(x^{(\mu)}) + f'(x^{(\mu)})(\varepsilon^{(\mu+1)} - \varepsilon^{(\mu)}) - [f(x^{(\mu)}) - f'(x^{(\mu)}) \cdot \varepsilon^{(\mu)} + \frac{1}{2}f''(\xi)(\varepsilon^{(\mu)})^2] = 0,$$

folgt

$$f'(x^{(\mu)}) \cdot \varepsilon^{(\mu+1)} - \frac{1}{2}f''(\xi)(\varepsilon^{(\mu)})^2 = 0.$$

Liegen ξ und $x^{(\mu)}$ nahe genug bei x^*, so ist der Lösungsfehler

$$\varepsilon^{(\mu+1)} \approx \frac{1}{2}(\varepsilon^{(\mu)})^2 \frac{f''(x^*)}{f'(x^*)}. \quad (6.6)$$

Das Verfahren konvergiert also für einfache Nullstellen ($f'(x^*) \neq 0$) **quadratisch**. Für Funktionen mit Nullstellen n-ter Ordnung,

$$f(x) = (x - x^*)^n \cdot h(x), \quad h(x^*) \neq 0,$$

folgt mit (6.2) und (6.5)

$$\varepsilon^{(\mu+1)} = \varepsilon^{(\mu)} - \frac{f(x^{(\mu)})}{f'(x^{(\mu)})} = \varepsilon^{(\mu)} \cdot \frac{n-1}{n} \cdot \frac{\dfrac{\varepsilon^{(\mu)}h'(x^{(\mu)})}{n-1} + h(x^{(\mu)})}{\dfrac{\varepsilon^{(\mu)}h'(x^{(\mu)})}{n} + h(x^{(\mu)})}.$$

Für $x^{(\mu)}$ nahe x^* ($\varepsilon^{(\mu)} \ll 1$) gilt dann nur noch **lineare** Konvergenz:

$$\varepsilon^{(\mu+1)} \approx \varepsilon^{(\mu)} \cdot \frac{n-1}{n}$$

Bei der praktischen Anwendung des Newton-Verfahrens wird in der Regel nicht überprüft, ob alle Voraussetzungen für die Konvergenz erfüllt sind. Aus diesem Grund kommt es häufig zu Konvergenzproblemen (→ Abb.6.4b,c). Zur Vermeidung dieser Probleme ist eine günstige Wahl des Startwertes (in der Nähe der Nullstelle) von entscheidender Bedeutung.

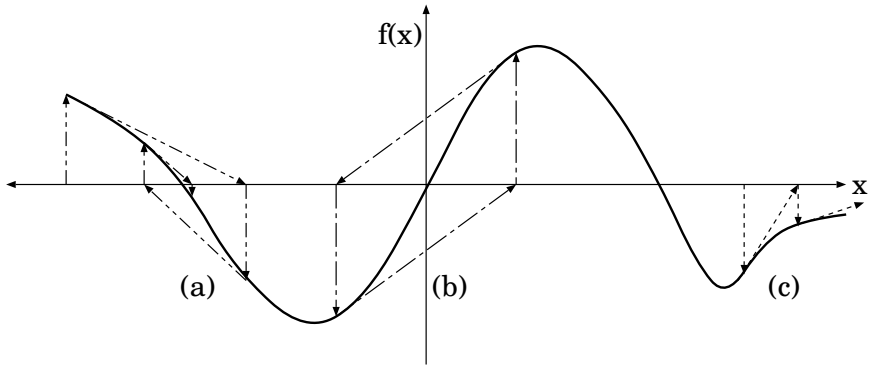


Abb. 6.4: (a) Konvergenz, (b) Oszillation und (c) Divergenz des Newton-Verfahrens

Beispiel 6.1 *Für die gegebene Schaltung* (→ Abb.6.5) *wird der Arbeitspunkt mit Hilfe des Newton-Verfahrens bestimmt.*

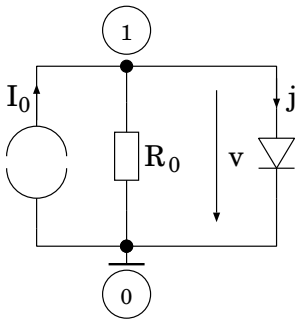


Abb. 6.5

$$
\begin{aligned}
j &= I_s(\exp\left(\frac{v}{U_T}\right) - 1) \\
I_s &= 10^{-10}\ A\ \textit{(Sperrstrom)} \\
U_T &= \frac{kT}{q} = 0.0259\ \textit{(Temp.-Spannung)} \\
q &= 1.6 \cdot 10^{-19}\ As\ \textit{(Elementarladung)} \\
k &= 1.39 \cdot 10^{-23}\ \frac{Ws}{K}\ \textit{(Boltzmannkonst.)} \\
T &= 298\ K\ \textit{(abs. Temperatur)} \\
R_0 &= 100\ \Omega;\ I_0 = 0.1\ A
\end{aligned}
$$

Die Summe der Ströme am Knoten 1 ist

$$f(v) = I_0 - \frac{v}{R_0} - I_s(\exp\left(\frac{v}{U_T}\right) - 1). \tag{6.7}$$

Gesucht ist die Spannung v^*, die das Kirchhoffsche Stromgesetz erfüllt:

$$f(v^*) = 0 \tag{6.8}$$

Neben der numerischen gibt es auch eine graphische Methode zur Arbeitspunktbestimmung. Der Arbeitspunkt (→ Abb.6.6) *ist dabei der Schnittpunkt der Arbeitsgeraden*

$$y = -0.01 \cdot v + 0.1$$

mit der Diodenkennlinie

$$j = I_s(\exp\left(\frac{v}{U_T}\right) - 1). \tag{6.9}$$

Der Schnittpunkt der Arbeitsgeraden mit der Abszisse ergibt sich aus der Klemmenspannung zwischen den Knoten 1 und 0 bei Leerlauf ($j = 0$)*, der Schnittpunkt mit der Ordinate aus dem Kurzschlußstrom* ($v = 0$)*.*

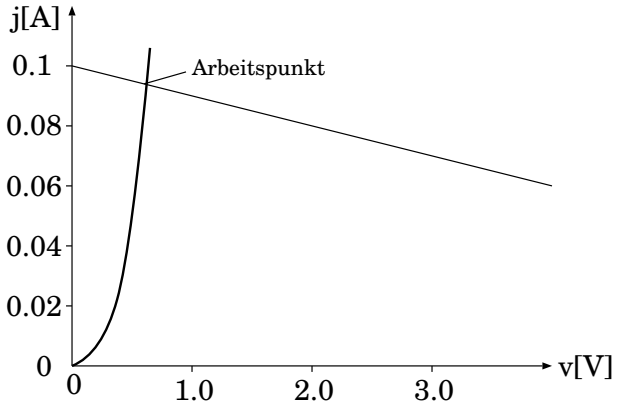


Abb. 6.6: Graphische Arbeitspunktbestimmung

Für die numerische Nullstellenbestimmung folgt mit

$$f'(v) = -\frac{1}{R_0} - \frac{I_s}{U_T} \cdot \exp\left(\frac{v}{U_T}\right)$$

die Iterationsvorschrift zu:

$$v^{(\mu+1)} = \varphi(v^{(\mu)}) = v^{(\mu)} - \frac{v^{(\mu)} - I_0 R_0 + I_s R_0 (\exp\left(\frac{v^{(\mu)}}{U_T}\right) - 1)}{1 + \frac{I_s R_0}{U_T} \exp\left(\frac{v^{(\mu)}}{U_T}\right)},\ \mu = 0, 1, 2, \ldots \qquad (6.10)$$

Aus Abb.6.6 *kann man ablesen, daß der Arbeitspunkt bei ungefähr* $0.5\,V$ *liegen muß. Das Intervall für die Wahl des Startwertes wird daher zu* $I = [0, 0.6]$ *gewählt. In* I *gilt dann*

a) $f(0)f(0.6) = 0.1 \cdot (-1.0681) = -0.1068 < 0$,

b) $f'(v) \neq 0$ *und*

c) $f''(v) = -\frac{I_s}{(U_T)^2} \cdot \exp\left(\frac{v}{U_T}\right) < 0$ *(konkav).*

Wegen a) und c) wird $v^{(0)} = 0.6$ *als Startwert gewählt (gleiches Vorzeichen von* $f(v^{(0)})$ *und der zweiten Ableitung). Die sich mit* (6.10) *ergebende Folge* $(v^{(\mu)})_{\mu \geq 0}$ *ist:*

μ	0	1	2	3	4
$\varphi(v^{(\mu)})$	0.57621	0.55559	0.54141	0.53579	0.53508

Nach üblichem Runden folgt dann für den Arbeitspunkt:

$$v^* = 0.5351$$

Eine schaltungstechnisch näherliegende Methode zur Arbeitspunktbestimmung ist die iterative Berechnung der **linearisierten Schaltung**.

Dazu wird zuerst die Diodengleichung (6.9) *an der Stelle* $v^{(\mu)}$ *linearisiert:*

$$\begin{aligned}
\bar{j}^{(\mu+1)} &= j^{(\mu)} + \underbrace{\frac{\partial j(v^{(\mu)})}{\partial v}}_{G^{(\mu)}} \left(v^{(\mu+1)} - v^{(\mu)}\right) \\
&= j^{(\mu)} + G^{(\mu)} \cdot v^{(\mu+1)} - G^{(\mu)} \cdot v^{(\mu)} \quad mit \quad G^{(\mu)} = \frac{I_s}{U_T} \exp\left(\frac{v^{(\mu)}}{U_T}\right) \\
\bar{j}^{(\mu+1)} &= G^{(\mu)} \cdot v^{(\mu+1)} + \underbrace{j^{(\mu)} - G^{(\mu)} \cdot v^{(\mu)}}_{Quellenkorrektur}
\end{aligned} \tag{6.11}$$

Aus (6.11) *kann dann das linearisierte Ersatzschaltbild* (→ Abb.6.7) *direkt angegeben werden.*

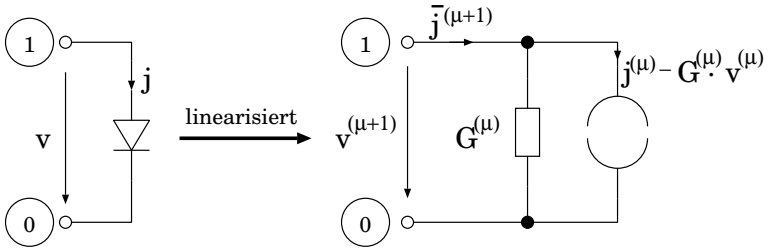


Abb. 6.7: Linearisiertes Ersatzschaltbild der Diode im $(\mu+1)$-ten Iterationsschritt

Durch das Einsetzen der Ersatzschaltung der Diode in das Netzwerk (→ Abb.6.8)*, erhält man die Schaltung im* $(\mu+1)$*-ten Iterationsschritt.*

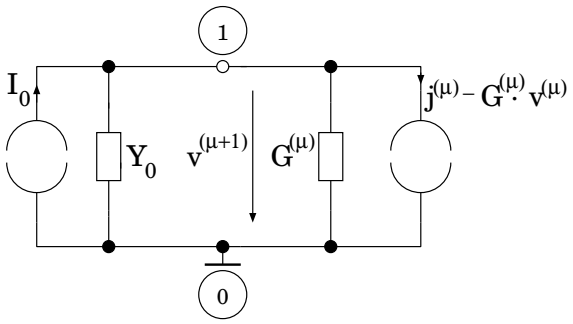


Abb. 6.8: Linearisierte Schaltung aus Abb.6.5

Für (6.8) *folgt dann mit* (6.11)

$$\begin{aligned}
I_0 - \frac{v^{(\mu+1)}}{R_0} - \bar{j}^{(\mu+1)} &= 0 \quad bzw. \\
I_0 - \frac{v^{(\mu+1)}}{R_0} - G^{(\mu)} v^{(\mu+1)} - j^{(\mu)} + G^{(\mu)} v^{(\mu)} &= 0
\end{aligned}$$

und durch Umstellung nach $v^{(\mu+1)}$ *wieder* (6.10)*:*

$$v^{(\mu+1)} = \frac{G^{(\mu)} v^{(\mu)} - j^{(\mu)} + I_0}{\frac{1}{R_0} + G^{(\mu)}}$$

$$
\begin{aligned}
v^{(\mu+1)} &= \frac{\frac{I_s}{U_T}\exp\left(\frac{v^{(\mu)}}{U_T}\right)v^{(\mu)} - I_s(\exp\left(\frac{v^{(\mu)}}{U_T}\right)-1) + I_0}{\frac{1}{R_0}+\frac{I_s}{U_T}\exp\left(\frac{v^{(\mu)}}{U_T}\right)} \\
&= \frac{I_0R_0 + \frac{I_sR_0}{U_T}\exp\left(\frac{v^{(\mu)}}{U_T}\right)v^{(\mu)} - I_sR_0(\exp\left(\frac{v^{(\mu)}}{U_T}\right)-1)}{1+\frac{I_sR_0}{U_T}\exp\left(\frac{v^{(\mu)}}{U_T}\right)} \\
&= \frac{v^{(\mu)}\left(1+\frac{I_sR_0}{U_T}\exp\left(\frac{v^{(\mu)}}{U_T}\right)\right) - \left(v^{(\mu)} - I_0R_0 + I_sR_0(\exp\left(\frac{v^{(\mu)}}{U_T}\right)-1)\right)}{1+\frac{I_sR_0}{U_T}\exp\left(\frac{v^{(\mu)}}{U_T}\right)} \\
v^{(\mu+1)} &= v^{(\mu)} - \frac{v^{(\mu)} - I_0R_0 + I_sR_0(\exp\left(\frac{v^{(\mu)}}{U_T}\right)-1)}{1+\frac{I_sR_0}{U_T}\cdot\exp\left(\frac{v^{(\mu)}}{U_T}\right)}
\end{aligned}
$$

Die iterative Berechnung der linearisierten Schaltung entspricht somit der direkten Berechnung der nichtlinearen Ausgangsgleichung (6.8) *durch das Newton-Verfahren.*

Die Nullstellenberechnung bei einem System von n Gleichungen,

$$\boldsymbol{f}(\boldsymbol{x}^*) = \boldsymbol{0},$$

erfolgt nach zeilenweiser Linearisierung der Funktionen

$$\bar{f}_i(\boldsymbol{x}) = f_i(\boldsymbol{x}^{(\mu)}) + \frac{\partial f_i(\boldsymbol{x}^{(\mu)})}{\partial x_1}\cdot(x_1 - x_1^{(\mu)}) + \frac{\partial f_i(\boldsymbol{x}^{(\mu)})}{\partial x_2}\cdot(x_2 - x_2^{(\mu)}) + \cdots$$

$$+\frac{\partial f_i(\boldsymbol{x}^{(\mu)})}{\partial x_n}\cdot(x_n - x_n^{(\mu)}) \quad \text{für} \quad i = 1,\ldots,n \quad \text{und} \quad \frac{\partial f_i(\boldsymbol{x}^{(\mu)})}{\partial x_j} := \left.\frac{\partial f_i(\boldsymbol{x})}{\partial x_j}\right|_{\boldsymbol{x}=\boldsymbol{x}^{(\mu)}},$$

sowie ihrer Zusammenfassung zu einem Gleichungssystem,

$$\bar{\boldsymbol{f}}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x}^{(\mu)}) + \boldsymbol{J}(\boldsymbol{x}^{(\mu)})\cdot(\boldsymbol{x} - \boldsymbol{x}^{(\mu)}),$$

durch die iterative Lösung der Tangentengleichungen

$$\bar{\boldsymbol{f}}(\boldsymbol{x}^{(\mu+1)}) = \boldsymbol{f}(\boldsymbol{x}^{(\mu)}) + \boldsymbol{J}(\boldsymbol{x}^{(\mu)})\cdot(\boldsymbol{x}^{(\mu+1)} - \boldsymbol{x}^{(\mu)}) = \boldsymbol{0}; \quad \mu = 0,1,\ldots \tag{6.12}$$

Umgestellt nach $\boldsymbol{x}^{(\mu+1)}$ führt (6.12) auf die Iterationsvorschrift für das n-dimensionale Newton-Verfahren:

$$\boxed{\boldsymbol{x}^{(\mu+1)} = \boldsymbol{x}^{(\mu)} - \boldsymbol{J}^{-1}(\boldsymbol{x}^{(\mu)})\cdot\boldsymbol{f}(\boldsymbol{x}^{(\mu)});\ \mu = 0,1,\ldots} \tag{6.13}$$

Die Matrix $\boldsymbol{J}$ wird **Funktional**- bzw. **Jacobische** Matrix genannt. Sie enthält alle Ableitungen aller Funktionen nach allen Variablen:

$$\boldsymbol{J}(\boldsymbol{x}) = \frac{\partial \boldsymbol{f}(\boldsymbol{x})}{\partial \boldsymbol{x}} = \begin{pmatrix} \frac{\partial f_1(\boldsymbol{x})}{\partial x_1} & \cdots & \frac{\partial f_1(\boldsymbol{x})}{\partial x_n} \\ \vdots & \left(\frac{\partial f_\sigma(\boldsymbol{x})}{\partial x_\rho}\right)_{\sigma,\rho} & \vdots \\ \frac{\partial f_n(\boldsymbol{x})}{\partial x_1} & \cdots & \frac{\partial f_n(\boldsymbol{x})}{\partial x_n} \end{pmatrix}$$

Soll die Folge $(\boldsymbol{x}^{(\mu)})_{\mu\geq 0}$ gegen $\boldsymbol{x}^*$ konvergieren, so muß $\boldsymbol{J}(\boldsymbol{x})$ nicht singulär ($\det \boldsymbol{J}(\boldsymbol{x}) \neq 0$) und Lipschitzstetig ($\boldsymbol{J}(\boldsymbol{x})$ ist stetig und $\partial \boldsymbol{J}(\boldsymbol{x})/\partial \boldsymbol{x}$ beschränkt) sein und der Anfangswert $\boldsymbol{x}^{(0)}$ nahe genug bei $\boldsymbol{x}^*$ gewählt werden.

6.2 Das linearisierte Knoten-Spannungs-System

Ein nichtlineares Netzwerk besitzt Standardkanten mit nichtlinearen Elementen

$$j_\kappa = g(v_\kappa).$$

Faßt man alle „Y“-Kanten zusammen, so gilt

$$\boldsymbol{j} = \boldsymbol{g}(\boldsymbol{v}). \tag{6.14}$$

Für (5.4) folgt dann

$$\begin{aligned} &\text{a: } \boldsymbol{A}\cdot\boldsymbol{i} = \boldsymbol{0}, \\ &\text{b: } \quad \boldsymbol{i} = \boldsymbol{j} + \boldsymbol{I}_0 = \boldsymbol{g}(\boldsymbol{v}) + \boldsymbol{I}_0, \\ &\text{c: } \quad \boldsymbol{v} \;\; = \boldsymbol{u} - \boldsymbol{U}_0 = \boldsymbol{A}^T\cdot\boldsymbol{u}_n - \boldsymbol{U}_0. \end{aligned} \tag{6.15}$$

Durch das Einsetzen von c→b→a erhält man die nichtlineare Funktion

$$\boldsymbol{f}(\boldsymbol{u}_n) = \boldsymbol{A}\cdot\boldsymbol{g}(\underbrace{\boldsymbol{A}^T\cdot\boldsymbol{u}_n - \boldsymbol{U}_0}_{\boldsymbol{v}}) + \boldsymbol{A}\cdot\boldsymbol{I}_0, \tag{6.16}$$

deren Nullstelle gesucht wird.

Zur Aufstellung der Iterationsvorschrift (6.13) wird (6.16) in $\boldsymbol{u}_n^{(\mu)}$ linearisiert,

$$\bar{\boldsymbol{f}}(\boldsymbol{u}_n) = \boldsymbol{f}(\boldsymbol{u}_n^{(\mu)}) + \frac{\partial \boldsymbol{f}(\boldsymbol{u}_n^{(\mu)})}{\partial \boldsymbol{u}_n}(\boldsymbol{u}_n - \boldsymbol{u}_n^{(\mu)})$$

und die Nullstellen der Tangenten berechnet:

$$\bar{\boldsymbol{f}}(\boldsymbol{u}_n^{(\mu+1)}) = \boldsymbol{f}(\boldsymbol{u}_n^{(\mu)}) + \frac{\partial \boldsymbol{f}(\boldsymbol{u}_n^{(\mu)})}{\partial \boldsymbol{u}_n}(\boldsymbol{u}_n^{(\mu+1)} - \boldsymbol{u}_n^{(\mu)}) = \boldsymbol{0} \tag{6.17}$$

Die Ableitung von (6.16) an der Stelle $\boldsymbol{u}_n^{(\mu)}$ ist

$$\begin{aligned} \frac{\partial \boldsymbol{f}(\boldsymbol{u}_n^{(\mu)})}{\partial \boldsymbol{u}_n} &= \frac{\partial \boldsymbol{f}(\boldsymbol{g}^{(\mu)})}{\partial \boldsymbol{g}} \cdot \frac{\partial \boldsymbol{g}(\boldsymbol{v}^{(\mu)})}{\partial \boldsymbol{v}} \cdot \frac{\partial \boldsymbol{v}(\boldsymbol{u}_n^{(\mu)})}{\partial \boldsymbol{u}_n} \\ &= \boldsymbol{A}\cdot\frac{\partial \boldsymbol{g}(\boldsymbol{v}^{(\mu)})}{\partial \boldsymbol{v}}\cdot\boldsymbol{A}^T \\ &= \boldsymbol{A}\cdot\underbrace{\begin{pmatrix} \frac{\partial g_1(\boldsymbol{v}^{(\mu)})}{\partial v_1} & \cdots & \frac{\partial g_1(\boldsymbol{v}^{(\mu)})}{\partial v_k} \\ \vdots & & \vdots \\ \frac{\partial g_k(\boldsymbol{v}^{(\mu)})}{\partial v_1} & \cdots & \frac{\partial g_k(\boldsymbol{v}^{(\mu)})}{\partial v_k} \end{pmatrix}}_{\boldsymbol{Y}^{(\mu)}}\cdot\boldsymbol{A}^T \end{aligned}$$

$$\frac{\partial \boldsymbol{f}(\boldsymbol{u}_n^{(\mu)})}{\partial \boldsymbol{u}_n} = \boldsymbol{A} \cdot \boldsymbol{Y}^{(\mu)} \cdot \boldsymbol{A}^T$$

Für die differentielle Knotenadmittanzmatrix folgt damit

$$\boldsymbol{Y}_n^{(\mu)} = \boldsymbol{A} \cdot \boldsymbol{Y}^{(\mu)} \cdot \boldsymbol{A}^T. \tag{6.18}$$

Mit

$$\begin{aligned} \boldsymbol{j}^{(\mu)} &= \boldsymbol{g}(\boldsymbol{v}^{(\mu)}), \\ \boldsymbol{A}^T \cdot \boldsymbol{u}_n^{(\mu)} &= \boldsymbol{v}^{(\mu)} + \boldsymbol{U}_0, \end{aligned}$$

(6.16) und (6.18) ergibt sich für (6.17)

$$\begin{aligned} \boldsymbol{A} \cdot \boldsymbol{j}^{(\mu)} + \boldsymbol{A} \cdot \boldsymbol{I}_0 + \boldsymbol{Y}_n^{(\mu)}(\boldsymbol{u}_n^{(\mu+1)} - \boldsymbol{u}_n^{(\mu)}) &= \boldsymbol{0}, \\ \boldsymbol{A} \cdot \boldsymbol{j}^{(\mu)} + \boldsymbol{A} \cdot \boldsymbol{I}_0 + \boldsymbol{Y}_n^{(\mu)} \cdot \boldsymbol{u}_n^{(\mu+1)} - \boldsymbol{A} \cdot \boldsymbol{Y}^{(\mu)} \cdot \boldsymbol{A}^T \cdot \boldsymbol{u}_n^{(\mu)} &= \boldsymbol{0}, \\ \boldsymbol{A} \cdot \boldsymbol{j}^{(\mu)} + \boldsymbol{A} \cdot \boldsymbol{I}_0 + \boldsymbol{Y}_n^{(\mu)} \cdot \boldsymbol{u}_n^{(\mu+1)} - \boldsymbol{A} \cdot \boldsymbol{Y}^{(\mu)}(\boldsymbol{v}^{(\mu)} + \boldsymbol{U}_0) &= \boldsymbol{0}, \quad \text{bzw.,} \end{aligned}$$

$$\begin{aligned} \boldsymbol{Y}_n^{(\mu)} \cdot \boldsymbol{u}_n^{(\mu+1)} &= \boldsymbol{A}[\boldsymbol{Y}^{(\mu)} \cdot \boldsymbol{U}_0 - \underbrace{(\boldsymbol{I}_0 + \boldsymbol{j}^{(\mu)} - \boldsymbol{Y}^{(\mu)} \cdot \boldsymbol{v}^{(\mu)})}_{\boldsymbol{I}_0^{(\mu)}}] \quad \text{und} \\ \boldsymbol{Y}_n^{(\mu)} \cdot \boldsymbol{u}_n^{(\mu+1)} &= \underbrace{\boldsymbol{A}[\boldsymbol{Y}^{(\mu)} \cdot \boldsymbol{U}_0 - \boldsymbol{I}_0^{(\mu)}]}_{\boldsymbol{I}_n^{(\mu)}}. \end{aligned}$$

Das Knoten-Spannungs-System im $(\mu+1)$-ten Iterationsschritt ist dann

$$\boxed{\boldsymbol{Y}_n^{(\mu)} \cdot \boldsymbol{u}_n^{(\mu+1)} = \boldsymbol{I}_n^{(\mu)}} \tag{6.19}$$

Bei der eben beschriebenen Vorgehensweise wird für die nichtlineare Schaltung zuerst das Gesamtsystem (6.16) aufgestellt und anschließend linearisiert. Dies ist sehr aufwendig und wie bereits im Beispiel 6.1 gezeigt, nicht notwendig.

Einfacher ist es, zuerst für die nichtlinearen Elemente kantenweise die linearen Ersatzschaltbilder zu bestimmen und in das Netzwerk einzusetzen. Für das so linearisierte Netzwerk kann dann mit den Bildungsgesetzen von Seite 95 das Knoten-Spannungs-System wieder direkt aufgestellt werden.

Die Linearisierung der Kantenelemente ergibt

$$\begin{aligned} \bar{\boldsymbol{j}}^{(\mu+1)} &= \boldsymbol{g}(\boldsymbol{v}^{(\mu)}) + \frac{\partial \boldsymbol{g}(\boldsymbol{v}^{(\mu)})}{\partial \boldsymbol{v}}(\boldsymbol{v}^{(\mu+1)} - \boldsymbol{v}^{(\mu)}), \\ \bar{\boldsymbol{i}}^{(\mu+1)} = \bar{\boldsymbol{j}}^{(\mu+1)} + \boldsymbol{I}_0 &= \boldsymbol{j}^{(\mu)} + \boldsymbol{Y}^{(\mu)}(\boldsymbol{v}^{(\mu+1)} - \boldsymbol{v}^{(\mu)}) + \boldsymbol{I}_0, \\ &= \boldsymbol{Y}^{(\mu)} \cdot \boldsymbol{v}^{(\mu+1)} + \underbrace{\boldsymbol{I}_0 + \boldsymbol{j}^{(\mu)} - \boldsymbol{Y}^{(\mu)} \cdot \boldsymbol{v}^{(\mu)}}_{\boldsymbol{I}_0^{(\mu)}}, \\ &= \boldsymbol{Y}^{(\mu)} \cdot \boldsymbol{v}^{(\mu+1)} + \boldsymbol{I}_0^{(\mu)} = \boldsymbol{Y}^{(\mu)}(\boldsymbol{u}^{(\mu+1)} - \boldsymbol{U}_0) + \boldsymbol{I}_0^{(\mu)}. \end{aligned}$$

Für (6.15) folgt dann

$$\begin{aligned} &\text{a: } \boldsymbol{A}\cdot\bar{\boldsymbol{i}}^{(\mu+1)} = \boldsymbol{0}, \\ &\text{b: } \quad \bar{\boldsymbol{i}}^{(\mu+1)} = \boldsymbol{Y}^{(\mu)}\,(\boldsymbol{u}^{(\mu+1)} - \boldsymbol{U}_0) + \boldsymbol{I}_0^{(\mu)}, \\ &\text{c: } \qquad\qquad\qquad \boldsymbol{u}^{(\mu+1)} = \boldsymbol{A}^T\cdot\boldsymbol{u}_n^{(\mu+1)}. \end{aligned} \tag{6.20}$$

Durch das Einsetzen von c→b→a und (6.18) erhält man wieder das Knoten-Spannungs-System im $(\mu+1)$-ten Iterationsschritt:

$$\boldsymbol{Y}_n^{(\mu)}\cdot\boldsymbol{u}_n^{(\mu+1)} = \boldsymbol{I}_n^{(\mu)}$$

Die Linearisierung der Kantenelemente ($j_\kappa = g_\kappa(v_\kappa)$) erfolgt wie im Beispiel 6.1:

$$\begin{aligned} \bar{j}_\kappa^{(\mu+1)} &= g(v_\kappa^{(\mu)}) + \frac{\partial g(v_\kappa^{(\mu)})}{\partial v_\kappa}\left(v_\kappa^{(\mu+1)} - v_\kappa^{(\mu)}\right), \quad \frac{\partial g(v_\kappa^{(\mu)})}{\partial v_\kappa} := G^{(\mu)}, \\ &= j_\kappa^{(\mu)} + G^{(\mu)}\cdot v_\kappa^{(\mu+1)} - G^{(\mu)}\cdot v_\kappa^{(\mu)} \end{aligned}$$

Der linearisierte Kantenstrom ist allgemein

$$\boxed{\bar{j}_\kappa^{(\mu+1)} = G^{(\mu)}\cdot v_\kappa^{(\mu+1)} + j^{(\mu)} - G^{(\mu)}\cdot v_\kappa^{(\mu)}} \tag{6.21}$$

In Abb.6.9 ist die Modifikation der Standardkante durch die Linearisierung dargestellt.

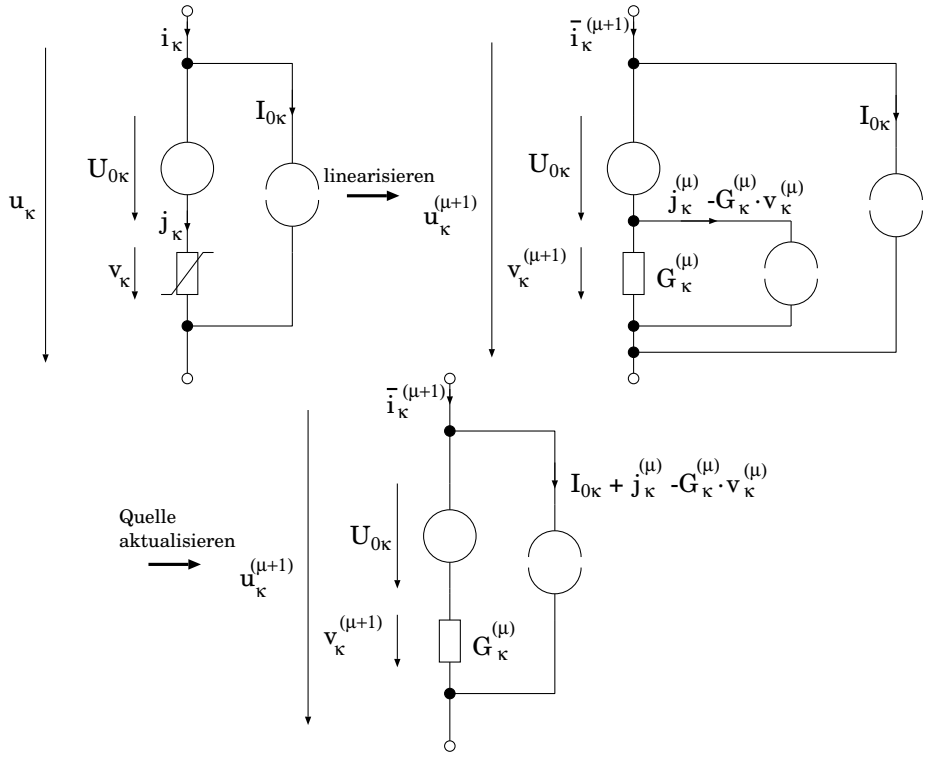


Abb. 6.9

Beispiel 6.2 *Für den gegebenen* (→ Abb.6.10) *Transistorverstärker ist das linearisierte Knoten-Spannungs-System aufzustellen.*

Der Transistor arbeitet im aktiven Bereich (Basis-Emitter-Diode in Flußrichtung, Basis--Kollektor-Diode in Sperrichtung). Eine gute Annäherung an das Verhalten des Transistors in diesem Bereich wird mit dem vereinfachten Großsignalersatzschaltbild (→ Abb.6.11) *nach Ebers und Moll erzielt.*

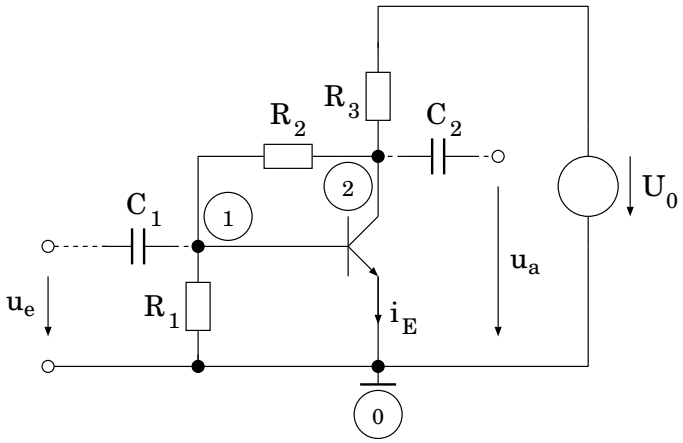


Abb. 6.10: Transistorverstärker

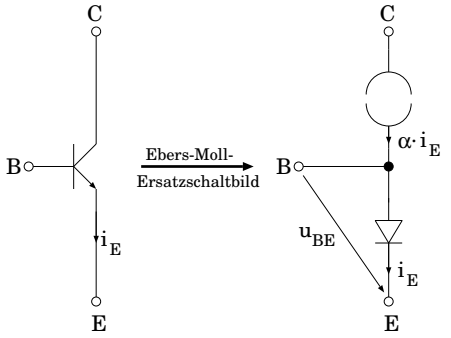


Abb. 6.11: Großsignalersatzschaltbild des Bipolartransistors

Der Emitterstrom ist $i_E = I_S(\exp\left(\frac{u_{BE}}{U_T}\right) - 1)$, *$\alpha$ ist ein Verstärkungsfaktor.*

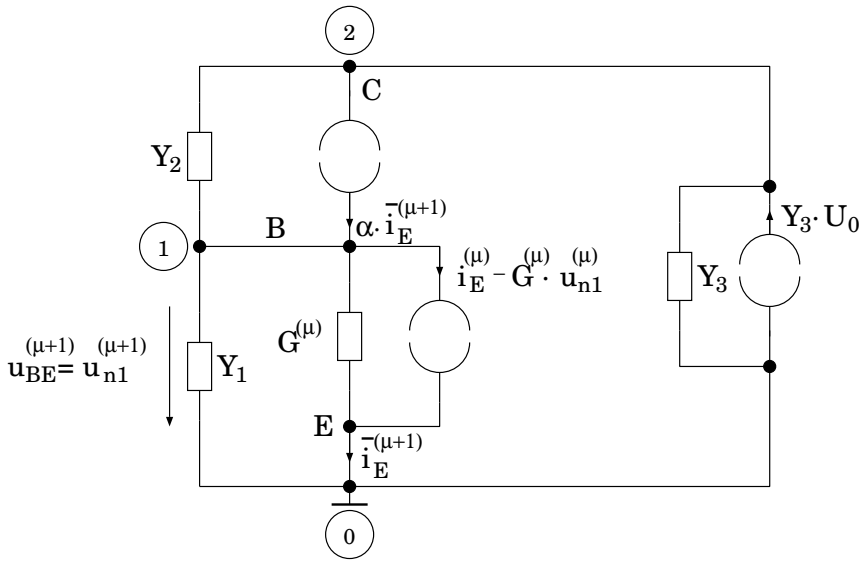


Abb. 6.12: Linearisierte Schaltung aus Abb.6.10

Setzt man für die Diode das linearisierte Ersatzschaltbild aus Abb.6.7 *in die Schaltung ein, so erhält man die lineare Schaltung* (→ Abb.6.12) *für das Netzwerk im* $(\mu+1)$*-ten Iterationsschritt. Für das Knoten-Spannungs-System folgt unter Verwendung der Bildungsgesetze von Seite 95*

$$\begin{pmatrix} Y_1+Y_2+G^{(\mu)} & -Y_2 \\ -Y_2 & Y_2+Y_3 \end{pmatrix} \begin{pmatrix} u_{n1}^{(\mu+1)} \\ u_{n2}^{(\mu+1)} \end{pmatrix} = \begin{pmatrix} \alpha\bar{i}_E^{(\mu+1)} + G^{(\mu)}u_{n1}^{(\mu)} - i_E^{(\mu)} \\ Y_3U_0 - \alpha\bar{i}_E^{(\mu+1)} \end{pmatrix}. \tag{6.22}$$

Im Knotenquellenstromvektor steht noch $\bar{i}_E^{(\mu+1)}$*,*

$$\bar{i}_E^{(\mu+1)} = G^{(\mu)} \cdot u_{n1}^{(\mu+1)} + i_E^{(\mu)} - G^{(\mu)} \cdot u_{n1}^{(\mu)}. \tag{6.23}$$

Setzt man (6.23) *in* (6.22) *ein,*

$$\begin{pmatrix} Y_1+Y_2+G^{(\mu)} & -Y_2 \\ -Y_2 & Y_2+Y_3 \end{pmatrix} \begin{pmatrix} u_{n1}^{(\mu+1)} \\ u_{n2}^{(\mu+1)} \end{pmatrix} = \begin{pmatrix} \alpha G^{(\mu)}u_{n1}^{(\mu+1)} + (\alpha-1)(i_E^{(\mu)} - G^{(\mu)}u_{n1}^{(\mu)}) \\ Y_3U_0 - \alpha G^{(\mu)}u_{n1}^{(\mu+1)} - \alpha(i_E^{(\mu)} - G^{(\mu)}u_{n1}^{(\mu)}) \end{pmatrix}$$

und bringt die Unbekannte $u_{n1}^{(\mu+1)}$ *auf die linke Seite, so folgt*

$$\begin{pmatrix} Y_1+Y_2+G^{(\mu)}(1-\alpha) & -Y_2 \\ -Y_2+\alpha G^{(\mu)} & Y_2+Y_3 \end{pmatrix} \begin{pmatrix} u_{n1}^{(\mu+1)} \\ u_{n2}^{(\mu+1)} \end{pmatrix} = \begin{pmatrix} (\alpha-1)(i_E^{(\mu)} - G^{(\mu)}u_{n1}^{(\mu)}) \\ Y_3U_0 - \alpha(i_E^{(\mu)} - G^{(\mu)}u_{n1}^{(\mu)}) \end{pmatrix}. \tag{6.24}$$

Dabei sind

$$\begin{aligned} i_E^{(\mu)} &= I_S[\exp\left(\frac{u_{n1}^{(\mu)}}{U_T}\right) - 1] \quad \textit{und} \\ G^{(\mu)} &= \frac{I_S}{U_T} \cdot \exp\left(\frac{u_{n1}^{(\mu)}}{U_T}\right). \end{aligned}$$

Wie bereits erwähnt, konvergiert das Newton-Verfahren nicht in jedem Fall. Existieren mehrere Lösungen, so kann oft nicht vorausgesagt werden, gegen welche das Verfahren konvergieren wird.

Speziell bei Nichtlinearitäten mit exponentiellem Verlauf kann man das Konvergenzverhalten wesentlich verbessern, indem man die spezielle Form der Kennlinien ausnutzt. Im Falle von

$$u_{n1}^{(\mu+1)} < 0$$

wird der normale *Verlauf und für*

$$u_{n1}^{(\mu+1)} > 0$$

ein modifizierter *Verlauf zur Berechnung des Ausgangswertes für den nächsten Iterationsschritt gewählt.*

Beim normalen Verlauf wird der mit (6.24) *berechnete Wert für* $u_{n1}^{(\mu+1)}$ *zur Aktualisierung des Emitterstroms verwendet:*

$$\begin{aligned} i_E^{(\mu+1)} &= I_S[\exp\left(\frac{u_{n1}^{(\mu+1)}}{U_T}\right) - 1], \\ G^{(\mu+1)} &= \frac{I_S}{U_T} \cdot \exp\left(\frac{u_{n1}^{(\mu+1)}}{U_T}\right) \end{aligned}$$

In Abb.6.13a *ist der normale Verlauf skizziert:*

$$(u_{n1}^{(\mu)}, i_E^{(\mu)}) \rightarrow (u_{n1}^{(\mu+1)}, \bar{i}_E^{(\mu+1)}) \rightarrow (u_{n1}^{(\mu+1)}, i_E^{(\mu+1)})$$

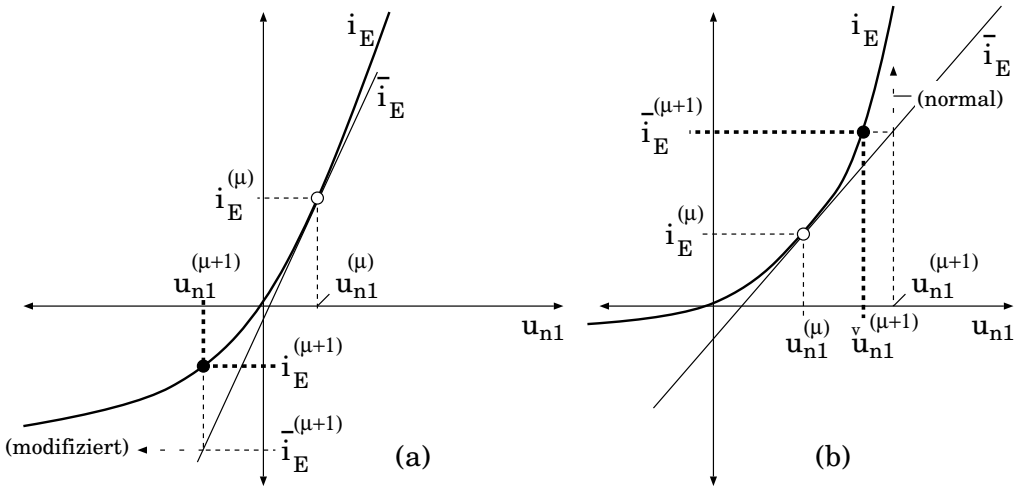


Abb. 6.13: (a) Normaler Verlauf, (b) modifizierter Verlauf des Newton-Verfahrens

Beim modifizierten Verlauf wird der linearisierte Emitterstrom zur Berechnung einer günstigeren Knotenspannung $\check{u}_{n1}^{(\mu+1)}$ *verwendet:*

$$\begin{aligned} \check{u}_{n1}^{(\mu+1)} &= U_T \cdot \ln\left[1 + \frac{\bar{i}_E^{(\mu+1)}}{I_S}\right], \\ \check{G}^{(\mu+1)} &= \frac{I_S}{U_T} \cdot \exp\left[\frac{\check{u}_{n1}^{(\mu+1)}}{U_T}\right] = \frac{1}{U_T}\left(I_S + \bar{i}_E^{(\mu+1)}\right) \end{aligned}$$

Abb.6.13b *zeigt die modifizierte Vorgehensweise:*

$$(u_{n1}^{(\mu)}, i_E^{(\mu)}) \rightarrow (u_{n1}^{(\mu+1)}, \bar{i}_E^{(\mu+1)}) \rightarrow (\breve{u}_{n1}^{(\mu+1)}, \bar{i}_E^{(\mu+1)})$$

Die Stabilität des Verfahrens wird durch die entsprechende Wahl des normalen oder modifizierten Verfahrens wesentlich verbessert.

Beispiel 6.3 *Für einen Schmitt-Trigger* (→ Abb.6.17) *soll der Arbeitspunkt der beiden Transistoren berechnet werden. Die Schaltung wird mit zwei Sperrschicht-n-Kanal-Feldeffektransistoren des Typs 2N4222 realisiert* (→ Abbn.6.14 und 6.15).

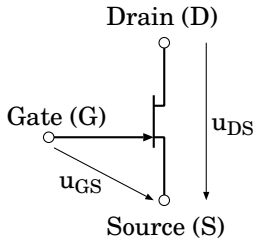


Abb. 6.14: Symbol des n-Kanal-Sperrschicht-FETs

Die Schaltung soll im aktiven Bereich betrieben werden. Für den Drainstrom im aktiven Bereich gilt:

$$i_D = \begin{cases} \dfrac{I_{DS}}{(U_P)^2}(u_{GS} - U_P)^2 & \text{für } u_{GS} > U_P \text{ und} \\ & |u_{DS}| > |u_{GS} - U_P| \text{ (aktiver Bereich)} \\ 0 & \text{für } u_{GS} \leq U_P \end{cases} \tag{6.25}$$

Dabei sind $U_P = -6\,V$ *die konstante Schwellenspannung (pinch-off-voltage) und* $I_{DS} = 9\,mA$ *der Sättigungsstrom.*

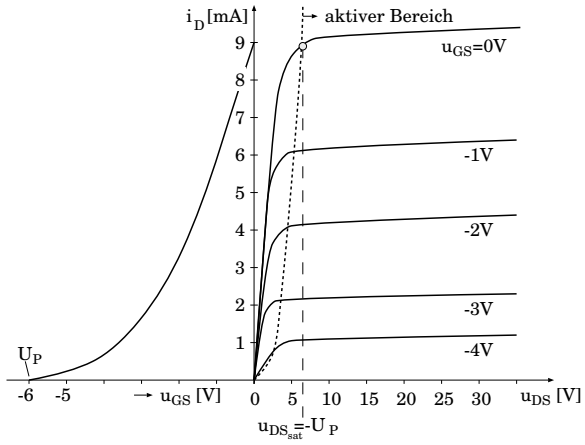


Abb. 6.15: Kennlinien des n-Kanal-Sperrschicht-FETs

Für den Drainstrom gilt im $(\mu+1)$*-ten Iterationsschritt:*

$$\bar{i}_D^{(\mu+1)} = i_D^{(\mu)} + \frac{\partial i_D(u_{GS}^{(\mu)})}{\partial u_{GS}} \left(u_{GS}^{(\mu+1)} - u_{GS}^{(\mu)} \right), \quad S^{(\mu)} := \frac{\partial i_D(u_{GS}^{(\mu)})}{\partial u_{GS}},$$

$$\bar{i}_D^{(\mu+1)} = \underbrace{S^{(\mu)} \cdot u_{GS}^{(\mu+1)}}_{\text{gesteuerte Quelle}} + \underbrace{i_D^{(\mu)} - S^{(\mu)} \cdot u_{GS}^{(\mu)}}_{\text{Urquelle}}$$

Die Urquelle bewirkt die Modifikation des Knotenquellenvektors (→ Abb.6.16), $S^{(\mu)}$ *ist die „Steilheit“ des Transistors.*

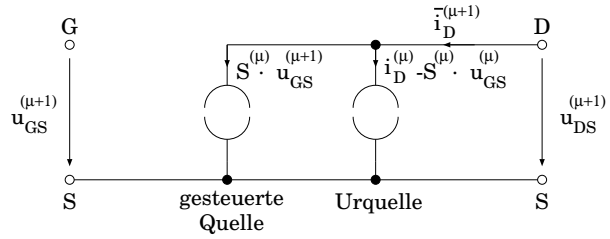
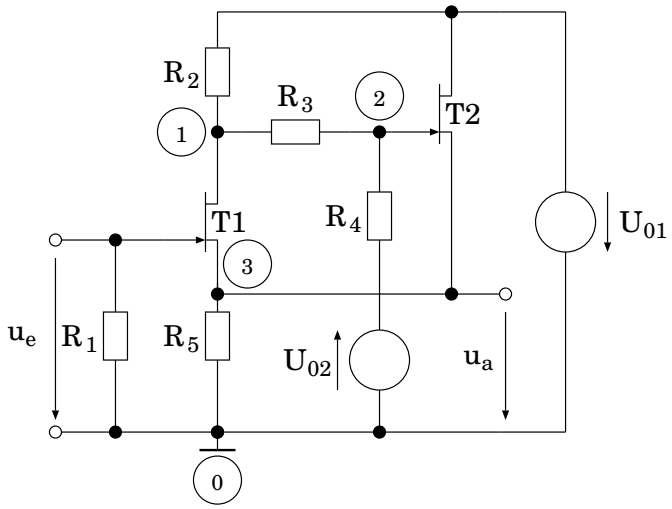


Abb. 6.16: Linearisiertes Ersatzschaltbild des n-Kanal-Sperrschicht-FETs

Setzt man die Werte für den Sättigungsstrom und die Schwellenspannung in (6.25) *ein, so erhält man:*

$$i_D = \begin{cases} 0.25\,\frac{mA}{V^2}\,(u_{GS}[V] + 6\ V)^2 & \text{für } u_{GS} > -6\ V,\ |u_{DS}| > |u_{GS} + 6\ V| \\ 0 & \text{für } u_{GS} \le -6\ V \end{cases}$$

$$S = \begin{cases} 0.5\frac{mA}{V^2}\,(u_{GS}[V] + 6\ V) & \text{für } u_{GS} > -6\ V,\ |u_{DS}| > |u_{GS} + 6\ V| \\ 0 & \text{für } u_{GS} \le -6\ V \end{cases}$$



$$\begin{aligned} R_1 &= 1\ M\Omega \\ R_2 &= 6.8\ k\Omega\ (6.8) \\ R_3 &= 51\ k\Omega\ (51) \\ R_4 &= 82\ k\Omega\ (82) \\ R_5 &= 2.2\ k\Omega\ (2.2) \\ U_{01} &= 20\ V\ (20) \\ U_{02} &= 20\ V\ (20) \\ U_P &= -6\ V\ (-6) \\ I_{DS} &= 9\ mA\ (9) \end{aligned}$$

Abb. 6.17: Schmitt-Trigger

Die Schaltung des Schmitt-Triggers (→ Abb.6.17) *wird nun linearisiert* (→ Abb.6.18). *Dazu werden statt der beiden Transistoren ihre Ersatzschaltbilder eingesetzt und der Drainanschluß von* T2 *(über die ideale Spannungsquelle* U_{01}*) an Masse* (→ Abb.6.9) *gelegt.* U_{01} *und* R_2 *liegen dann in Reihe und können wie* U_{02} *und* R_4 *zu einer äquivalenten Stromquelle umgeformt werden. Für die Gate-Source-Spannungen gilt außerdem* $u_{GS1} = u_e - u_{n3}$ *und* $u_{GS2} = u_{n2} - u_{n3}$.

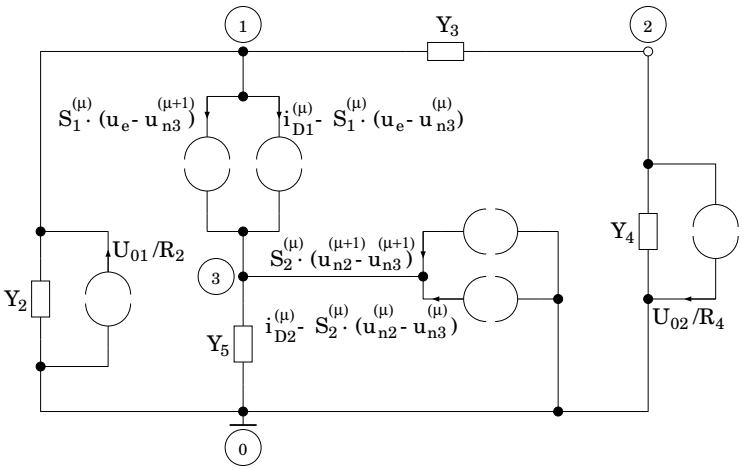


Abb. 6.18: Linearisiertes Schaltbild des Schmitt-Triggers

Vor dem ersten Iterationsschritt werden die Netzwerkgrößen auf die Bezugsgrößen

$$U_B = 1\ V \quad und \quad I_B = 1\ mA \quad (R_B = U_B/I_B = 1\ k\Omega)$$

normiert. (Die dimensionslosen Werte stehen in Klammern hinter den physikalischen.) Für das Knoten-Spannungs-System im $(\mu+1)$-ten Interationsschritt und $u_e = 0$ folgt mit Hilfe der Bildungsgesetze von Seite 95:

$$\begin{pmatrix} Y_2{+}Y_3 & -Y_3 & -S_1^{(\mu)} \\ -Y_3 & Y_3{+}Y_4 & 0 \\ 0 & -S_2^{(\mu)} & Y_5{+}S_1^{(\mu)}{+}S_2^{(\mu)} \end{pmatrix} \begin{pmatrix} u_{n1}^{(\mu+1)} \\ u_{n2}^{(\mu+1)} \\ u_{n3}^{(\mu+1)} \end{pmatrix} =$$

$$\begin{pmatrix} Y_2U_{01} - i_{D1}^{(\mu)} - S_1^{(\mu)} \cdot u_{n3}^{(\mu)} \\ -Y_4U_{02} \\ i_{D1}^{(\mu)} + i_{D2}^{(\mu)} + S_1^{(\mu)} \cdot u_{n3}^{(\mu)} - S_2^{(\mu)}(u_{n2}^{(\mu)} - u_{n3}^{(\mu)}) \end{pmatrix}$$

Setzt man die Werte für die konstanten Elemente ein, so folgt:

$$\begin{pmatrix} 0.1667 & -0.01961 & -S_1^{(\mu)} \\ -0.01961 & 0.03180 & 0 \\ 0 & -S_2^{(\mu)} & 0.4545{+}S_1^{(\mu)}{+}S_2^{(\mu)} \end{pmatrix} \begin{pmatrix} u_{n1}^{(\mu+1)} \\ u_{n2}^{(\mu+1)} \\ u_{n3}^{(\mu+1)} \end{pmatrix} =$$

$$\begin{pmatrix} 2.941 - i_{D1}^{(\mu)} - S_1^{(\mu)} \cdot u_{n3}^{(\mu)} \\ -0.2439 \\ i_{D1}^{(\mu)} + i_{D2}^{(\mu)} + S_1^{(\mu)} \cdot u_{n3}^{(\mu)} - S_2^{(\mu)}(u_{n2}^{(\mu)} - u_{n3}^{(\mu)}) \end{pmatrix}$$

Die zu iterierenden Knotenpotentiale sind u_{n2} und u_{n3}. Die Iteration soll beendet werden, wenn

$$|u_{ni}^{(\mu+1)} - u_{ni}^{(\mu)}| < 0.05;\ i = 2, 3,$$

erfüllt ist (50 mV Genauigkeit). Da nur u_{n2} und u_{n3} berechnet werden müssen, wird zuerst durch einen Gauß-Eliminationsschritt das Untersystem $(\boldsymbol{Y}_{n_1}^{(\mu)} \boldsymbol{I}_{n_1}^{(\mu)})^{(2)}$ bestimmt:

$$\begin{pmatrix} 0.1667 & -0.01961 & -S_1^{(\mu)} \\ 0 & 0.02949 & -0.1176 \cdot S_1^{(\mu)} \\ 0 & -S_2^{(\mu)} & 0.4545{+}S_1^{(\mu)}{+}S_2^{(\mu)} \end{pmatrix} \begin{pmatrix} u_{n1}^{(\mu+1)} \\ u_{n2}^{(\mu+1)} \\ u_{n3}^{(\mu+1)} \end{pmatrix} =$$

$$\begin{pmatrix} 2.941 - i_{D1}^{(\mu)} - S_1^{(\mu)} \cdot u_{n3}^{(\mu)} \\ 0.1021 - 0.1176\,(i_{D1}^{(\mu)} + S_1^{(\mu)} \cdot u_{n3}^{(\mu)}) \\ i_{D1}^{(\mu)} + i_{D2}^{(\mu)} + S_1^{(\mu)} \cdot u_{n3}^{(\mu)} - S_2^{(\mu)}(u_{n2}^{(\mu)} - u_{n3}^{(\mu)}) \end{pmatrix}$$

Das Untersystem ist:

$$\begin{pmatrix} 0.02949 & -0.1176 \cdot S_1^{(\mu)} \\ -S_2^{(\mu)} & 0.4545 + S_1^{(\mu)} + S_2^{(\mu)} \end{pmatrix} \begin{pmatrix} u_{n2}^{(\mu+1)} \\ u_{n3}^{(\mu+1)} \end{pmatrix} = \begin{pmatrix} 0.1021 - 0.1176\,(i_{D1}^{(\mu)} + S_1^{(\mu)} \cdot u_{n3}^{(\mu)}) \\ i_{D1}^{(\mu)} + i_{D2}^{(\mu)} + S_1^{(\mu)} \cdot u_{n3}^{(\mu)} - S_2^{(\mu)}(u_{n2}^{(\mu)} - u_{n3}^{(\mu)}) \end{pmatrix}$$

Eine erste Iteration soll mit den Startwerten

$$u_{n2}^{(0)} = 2 \quad \textit{und} \quad u_{n3}^{(0)} = 5.5$$

$$(u_{GS1}^{(0)} = -5.5 > U_P \quad \textit{und} \quad u_{GS2}^{(0)} = -3.5 > U_P)$$

durchgeführt werden. Die Ströme und Steigungen sind dann

$$\begin{aligned} i_{D1}^{(0)} &= 0.25(6 - u_{n3}^{(0)})^2 = 0.0625, \\ i_{D2}^{(0)} &= 0.25[(6 + (u_{n2}^{(0)} - u_{n3}^{(0)})]^2 = 1.563, \\ S_1^{(0)} &= 0.5(6 - u_{n3}^{(0)}) = 0.25, \\ S_2^{(0)} &= 0.5[(6 + (u_{n2}^{(0)} - u_{n3}^{(0)})] = 1.25 \end{aligned}$$

und für das Untersystem folgt

$$\begin{pmatrix} 0.02949 & -0.02940 \\ -1.25 & 1.9545 \end{pmatrix} \begin{pmatrix} u_{n2}^{(1)} \\ u_{n3}^{(1)} \end{pmatrix} = \begin{pmatrix} -0.06695 \\ 7.376 \end{pmatrix}.$$

Die Lösung des Gleichungssystems ist

$$\begin{pmatrix} u_{n2}^{(1)} \\ u_{n3}^{(1)} \end{pmatrix} = \begin{pmatrix} 4.120 \\ 6.409 \end{pmatrix}$$

und die Aktualisierung der Ströme und Steigungen ergibt

$$\begin{aligned} u_{GS1}^{(1)} &= -6.409 < U_P \quad \Rightarrow \quad i_{D1}^{(1)} = 0;\ S_1^{(1)} = 0, \\ u_{GS2}^{(1)} &= -2.29 > U_P \quad \Rightarrow \quad i_{D2}^{(1)} = 3.443;\ S_2^{(1)} = 1.856. \end{aligned}$$

Mit diesen Werten ist das Gleichungssystem im 2-ten Iterationsschritt

$$\begin{pmatrix} 0.02949 & 0 \\ -1.856 & 2.311 \end{pmatrix} \begin{pmatrix} u_{n2}^{(2)} \\ u_{n3}^{(2)} \end{pmatrix} = \begin{pmatrix} 0.1021 \\ 7.691 \end{pmatrix}.$$

Es hat die Lösung

$$\begin{pmatrix} u_{n2}^{(2)} \\ u_{n3}^{(2)} \end{pmatrix} = \begin{pmatrix} 3.462 \\ 6.108 \end{pmatrix}.$$

Die Ergebnisse des nächsten Iterationsschrittes,

$$\begin{pmatrix} u_{n2}^{(3)} \\ u_{n3}^{(3)} \end{pmatrix} = \begin{pmatrix} 3.462 \\ 6.123 \end{pmatrix},$$

erfüllen dann das geforderte Abbruchkriterium.

Die Berechnung des Arbeitspunkts wird nun mit neuen Startwerten,

$$u_{n2}^{(0)} = -4 \quad \textit{und} \quad u_{n3}^{(0)} = 3,$$

wiederholt. Es ergeben sich dabei folgende Werte:

$$\begin{array}{rclcrcl}
u_{n2}^{(0)} & = & -4 & ; & u_{n3}^{(0)} & = & 3 \\
u_{n2}^{(1)} & = & -2.800 & ; & u_{n3}^{(1)} & = & 3.454 \\
u_{n2}^{(2)} & = & -2.853 & ; & u_{n3}^{(2)} & = & 3.483 \\
u_{n2}^{(3)} & = & -2.853 & ; & u_{n3}^{(3)} & = & 3.483
\end{array}$$

Der Schmitt-Trigger hat folglich zwei stabile Zustände:

1. $u_{n2}^{(3)} = 3.462 \quad ; \quad u_{n3}^{(3)} = 6.123$

   (Transistor T1 *sperrt)*

2. $u_{n2}^{(3)} = -2.853 \quad ; \quad u_{n3}^{(3)} = 3.483$

   (Transistor T2 *sperrt)*

7 Einschwinganalyse dynamischer Netzwerke

Linearisierte Schaltungen, die zusätzlich speichernde Elemente (L's, C's) enthalten, werden durch **gewöhnliche Differentialgleichungen** (DGL) **mit konstanten Koeffizienten** beschrieben (→ Beispiel 5.6, Gl. 5.14). Ihre Lösung läßt sich sehr leicht mit Hilfe der **Laplace-Transformation** bestimmen. Dies geschieht in der Regel von Hand mit Hilfe von Tabellen oder durch Analogrechner. Die Laplace-Transformation eignet sich daher nur für kleinere Schaltungen.

Nimmt die Komplexität der Schaltungen zu, so erfolgt ihre Berechnung auf Digitalrechnern numerisch. Die dazu notwendige **Diskretisierung** macht aus dynamischen Schaltungen rein resistive (siehe vorheriges Kapitel). Durch die Diskretisierung der Differentialgleichungen erhält man **Differenzengleichungen**, die mittels **numerischer Integrationsverfahren** gelöst werden können. Der dabei gegenüber der analytischen Lösung gemachte Fehler wird **Diskretisierungsfehler** $\epsilon(\nu\Delta t)$ genannt. Die Genauigkeit der Lösungen hängt von der gewählten Schrittweite Δt ab. Eine kleine Schrittweite führt auf genauere Lösungen, beansprucht aber mehr Rechenzeit. Es ist daher sinnvoll, die Schrittweite den Stabilitätsbedingungen des Integrationsverfahrens gemäß vor jedem Integrationsschritt anzupassen (**Schrittweitenanpassung**).

Da alle vorkommenden Differentialgleichungen die gleiche Form haben, wird zur Berechnung des Diskretisierungsfehlers eine verallgemeinerte **Testdifferentialgleichung** verwendet:

$$\dot{x}(t) = p_\infty \cdot x(t) \tag{7.1}$$

Die analytische Lösung von (7.1) erfolgt mittels Laplace-Transformation. Die analytische Lösung der korrespondierenden Differenzengleichung,

$$\dot{x}(\nu\Delta t) = p_\infty \cdot x(\nu\Delta t), \tag{7.2}$$

erhält man mit Hilfe der **Z-Transformation**. Zur Abschätzung des Diskretisierungsfehlers werden beide Lösungen benötigt.

Im folgenden werden daher zunächst die wichtigsten Eigenschaften und Regeln der Laplace- und Z-Transformation wiederholt.

Es wird dabei nur die Transformation in den Bildbereich behandelt. Die Rücktransformation erfolgt dann mit den dabei bestimmten Korrespondenzen auf die zuvor partialbruchzerlegten Bildfunktionen. Die für die allgemeine Rücktranformation erforderlichen Regeln und Sätze der Funktionentheorie (Holomorphie, Residuensatz) werden dazu nicht benötigt.

7.1 Laplace-Transformation

Die „Laplace-Transformation“ wurde 1887 von Oliver Heaviside vorgestellt. Er zeigte, daß die symbolische Berechnung nichtstationärer Vorgänge als Verallgemeinerung der Wechselstromrechnung aufgefaßt werden kann. Die Grundidee seiner Methode besteht darin, den Differentialoperator $\frac{d}{dt}$ durch p zu ersetzen. Durch diese Substitution wird die Differentiation zu einer algebraischen Operation. Für die zur Differentiation „inverse“ Operation, die Integration, wird der „inverse“ Operator $1/p$ gesetzt. Die von Heaviside vorgestellte Methode war mathematisch unbegründet und führte mitunter zu falschen Resultaten. Von dem Historiker E.T. Bell wurde die Geschichte des Heaviside-Kalküls als eine „Symbolische Komödie in drei Akten“ bezeichnet:

Erster Akt, „völliger Unsinn“, zweiter Akt, „trivial“, dritter Akt, „lange vor Heaviside bekannt“.

Das Problem der Mehrdeutigkeit der von Heaviside eingeführten Operatoren wurde durch ein Integral, das Pierre Simon Laplace in seinem Buch über die Wahscheinlichkeitsrechnung (1812) verwendete, gelöst.

$$\boxed{X(p) := \mathfrak{L}\{x(t)\} := \int_0^\infty x(t)e^{-pt}\,dt} \tag{7.3}$$

Eine Funktion $x{:}[0,\infty) \to \mathbb{R}$ heißt Laplace-transformierbar, wenn das uneigentliche Integral (7.3) für

$$p = \alpha + j2\pi f = \alpha + j\omega \in \mathbb{C} \quad \text{und} \quad \alpha, \omega \in \mathbb{R},$$

konvergiert. In diesem Fall nennt man $X(p)$ die **Laplace-Transformierte** von $x(t)$. Man schreibt hierfür auch

$$x(t) \circ\!\!-\!\!\!\stackrel{L}{\bullet} X(p)$$

und nennt dies eine **Laplace-Korrespondenz** mit $X(p)$ als **Bildfunktion** von $x(t)$ und $x(t)$ als **Urbildfunktion** von $X(p)$. Die Urbildfunktion erhält man durch **Rücktransformation** der Bildfunktion aus dem Bildbereich:

$$x(t) = \mathfrak{L}^{-1}\{X(p)\}$$

Das Integral (7.3) konvergiert für stückweise stetige Funktionen $x(t)$ mit höchstens exponentiellem Wachstum,

$$|x(t)| \le M \cdot e^{kt}; \quad M, k \in \mathbb{R}.$$

Für sie gilt

$$\left|\int_0^\infty x(t)e^{-pt}\,dt\right| \le \int_0^\infty |x(t)|\underbrace{|e^{-pt}|}_{e^{-\alpha t}}\,dt \le M\cdot\int_0^\infty e^{(k-\alpha)t}\,dt = \frac{M}{k-\alpha}\cdot e^{(k-\alpha)t}\Big|_0^\infty = \frac{M}{\alpha-k} < \infty$$

für $\alpha > k$. Die Funktionen t^n, $\cosh(t)$, e^{kt} sind demnach Laplace-transformierbar, t^{-1}, e^{t^2} dagegen nicht. Die Voraussetzungen für die Konvergenz sind jedoch nur hinreichend, nicht notwendig. Es gibt durchaus L-transformierbare Funktionen, die nicht die Voraussetzungen erfüllen.

Beispiel 7.1 *Für die Sprung-(Heaviside)-Funktion* ($\rightarrow$ Abb.7.1),

$$\sigma(t) := \begin{cases} 0; & t < 0 \\ 1; & t \geq 0 \end{cases}, \qquad (7.4)$$

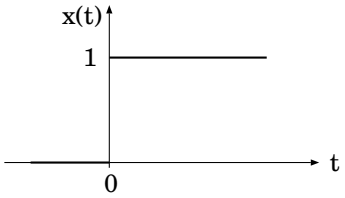


Abb. 7.1: $x(t) = \sigma(t)$

gilt:

$$X(p) = \int_0^\infty e^{-pt}\,dt = \lim_{\tau\to\infty} \frac{e^{-p\tau}-1}{-p} = \frac{1}{p} \quad \textit{für} \quad \mathrm{Re}\{p\} = \alpha > 0$$

Die L-Korrespondenz ist somit

$$\boxed{\sigma(t) \circ\!\!-\!\!\bullet\, \frac{1}{p}; \quad t \geq 0} \qquad (7.5)$$

Beispiel 7.2 *Es ist die L-Transformierte von*

$$x(t) = \frac{t^{\nu-1}}{(\nu-1)!} e^{p_\mu t}; \quad t \geq 0,\ \nu \in \mathbb{N},\ p_\mu \in \mathbb{C}$$

zu berechnen. Die Berechnung erfolgt durch partielle Integration von

$$\frac{1}{(\nu-1)!} \int_0^\infty t^{\nu-1} \cdot e^{(p_\mu - p)t}\,dt.$$

Es gilt

$$X(p) = \frac{1}{(\nu-1)!} \int_0^\infty \underbrace{t^{\nu-1}}_{u} \cdot \underbrace{e^{(p_\mu-p)t}}_{v'}\,dt = \frac{1}{(\nu-1)!} \cdot I_1.$$

Die Integration von I_1 ergibt

$$I_1 = \underbrace{\left[t^{(\nu-1)} \frac{1}{p_\mu - p} e^{(p_\mu-p)t}\right]_0^\infty}_{0} - \int_0^\infty (\nu-1)t^{\nu-2} \frac{1}{p_\mu - p} e^{(p_\mu-p)t}\,dt$$

für $\mathrm{Re}\{p_\mu - p\} < 0$ *bzw.* $\alpha > \alpha_\mu$. *Ein weiterer Integrationsschritt ergibt*

$$\begin{aligned} I_1 &= \frac{\nu-1}{p-p_\mu} \int_0^\infty \underbrace{t^{\nu-2}}_{u} \cdot \underbrace{e^{(p_\mu-p)t}}_{v'}\,dt = \frac{\nu-1}{p-p_\mu} \cdot I_2, \\ I_2 &= \underbrace{\left[t^{(\nu-2)} \frac{1}{p_\mu - p} e^{(p_\mu-p)t}\right]_0^\infty}_{0} - \int_0^\infty (\nu-2)t^{\nu-3} \frac{1}{p_\mu - p} e^{(p_\mu-p)t}\,dt, \\ I_2 &= \frac{\nu-2}{p-p_\mu} \int_0^\infty t^{\nu-3} e^{(p_\mu-p)t}\,dt. \end{aligned}$$

Nach $\nu-1$ Integrationen folgt ($t^0=1$)

$$X(p)=\frac{1}{(\nu-1)!}\left[\frac{(\nu-1)(\nu-2)\cdots 1}{(p-p_\mu)^{\nu-1}}\int_0^\infty e^{(p_\mu-p)t}\,dt\right]=\frac{1}{(p-p_\mu)^\nu}.$$

Die L-Korrespondenz ist damit

$$\boxed{\sigma(t)\cdot\frac{t^{\nu-1}}{(\nu-1)!}e^{p_\mu t}\circ\!\!-\!\!\bullet\ \frac{1}{(p-p_\mu)^\nu}} \tag{7.6}$$

Speziell für $\nu=1$ gilt

$$\boxed{\sigma(t)\cdot e^{p_\mu t}\circ\!\!-\!\!\bullet\ \frac{1}{(p-p_\mu)}} \tag{7.7}$$

Ist $p_\mu=0$, so folgt für (7.7) *wieder* (7.5).

Beispiel 7.3 *Mit* (7.7) *folgt*

$$\cos(\omega t)=\frac{e^{j\omega t}+e^{-j\omega t}}{2}\circ\!\!-\!\!\bullet\ \frac{1}{2}\left[\frac{1}{p-j\omega}+\frac{1}{p+j\omega}\right]=\frac{p}{p^2+\omega^2}$$

und

$$\sin(\omega t)=\frac{e^{j\omega t}-e^{-j\omega t}}{2j}\circ\!\!-\!\!\bullet\ \frac{1}{2j}\left[\frac{1}{p-j\omega}-\frac{1}{p+j\omega}\right]=\frac{\omega}{p^2+\omega^2}.$$

Die L-Korrespondenzen sind demnach

$$\boxed{\sigma(t)\cdot\cos(\omega t)\circ\!\!-\!\!\bullet\ \frac{p}{p^2+\omega^2}} \tag{7.8}$$

und

$$\boxed{\sigma(t)\cdot\sin(\omega t)\circ\!\!-\!\!\bullet\ \frac{\omega}{p^2+\omega^2}} \tag{7.9}$$

7.1.1 Die Dirac-Deltafunktion

In manchen Anwendungen hat man es mit inhomogenen linearen DGLn zu tun, in denen von der nur kurzzeitig wirkenden Störfunktion $x(t)$ (z.B. bei einem Hammerschlag oder einem Stromstoß) nur der „Gesamtimpuls“

$$\int_{t_0}^{t_1} x(t)\,dt,\quad t_1 \text{ nahe bei } t_0,$$

bekannt ist. Man spricht dann von einer **Impulsfunktion** (→ Abb.7.2)

$$\delta_\epsilon(t)=\begin{cases}0; & -\infty<t<0\\ \epsilon^{-1}; & 0\le t<\epsilon\\ 0; & \epsilon\le t<\infty\end{cases}\qquad \text{mit } \int_{-\infty}^{\infty}\delta_\epsilon(t)dt=1 \text{ und } \epsilon>0.$$

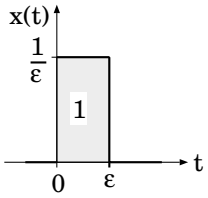


Abb. 7.2: Impulsfunktion

Kommt es bei einer Impulsfunktion, die in einem Intervall $(t_0, t_0+\epsilon)$ wirkt, nur auf den Gesamtimpuls I_0 an, so ersetzt man sie durch $I_0\delta_\epsilon(t-t_0)$.

In der Praxis möchte man den Gesamtimpuls gern auf einen Zeitpunkt $t = t_0$ konzentriert haben:

$$\delta(t) := \lim_{\epsilon\to 0} \delta_\epsilon(t) = \begin{cases} 0; & t \neq 0 \\ \infty; & t = 0 \end{cases} \tag{7.10}$$

$\delta(t)$ ist jedoch keine Funktion mehr, da ∞ kein Funktionswert und

$$\int_{-\infty}^{\infty} \delta(t)\,dt \tag{7.11}$$

im strengen Sinne nicht (Riemann-) integrierbar ist.

Zur Beschreibung eines derartigen stoßförmigen Impulses mit einer Zeitfunktion, verwendet man $\delta(t)$ als **verallgemeinerte Funktion** und nennt sie nach dem englischen Physiker Paul Dirac die **Dirac-Deltafunktion**. Man benötigt zur strengen Begründung von $\delta(t)$ die Theorie der Distributionen. Der praktische Umgang mit dem „Dirac“ ist jedoch immer unkritisch, wenn $\delta(t)$ als Faktor vor stetigen Funktionen unter bestimmten Integralen auftritt. In der Gleichungskette

$$\begin{aligned} \int_{-\infty}^{\infty} g(t)\delta(t-t_0)\,dt \quad &= \quad \int_{-\infty}^{\infty} g(t) \lim_{\epsilon\to 0} \delta_\epsilon(t-t_0)\,dt \\ &\text{„}=\text{“} \quad \lim_{\epsilon\to 0} \int_{-\infty}^{\infty} g(t)\delta_\epsilon(t-t_0)\,dt = g(t_0) \end{aligned} \tag{7.12}$$

ist zwar die Grenzwertvertauschung nicht erlaubt, die Beziehung ist aber dennoch für jede in t_0 stetige Funktion $g(t)$ ($\rightarrow$ Abb.7.3) richtig.

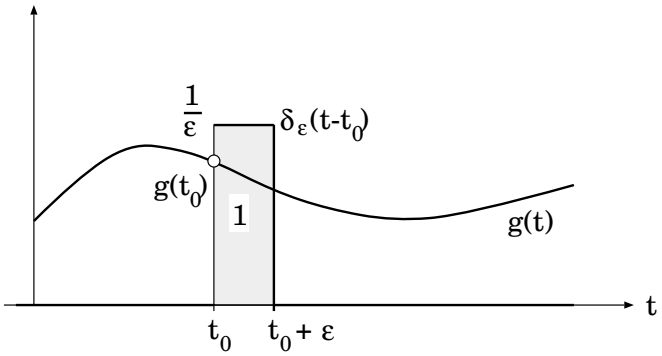


Abb. 7.3: $\int_{-\infty}^{\infty} g(t)\delta_\epsilon(t-t_0)\,dt$

Aus (7.12) folgt die sogenannte **Ausblendeigenschaft** der Deltafunktion:

$$\boxed{\int_{-\infty}^{\infty} g(t)\delta(t-t_0)\,dt = g(t_0), \quad \text{falls } g \text{ stetig in } t_0} \tag{7.13}$$

Beispiel 7.4 *Mit* (7.13) *folgt*

$$\mathfrak{L}\{\delta(t-t_0)\} = \int_0^\infty e^{-pt}\delta(t-t_0)\,dt = e^{-pt_0}$$

und für $t_0 = 0$

$$\mathfrak{L}\{\delta(t)\} = \int_0^\infty e^{-pt}\delta(t)\,dt = 1.$$

Die L-Korrespondenzen sind

$$\boxed{\delta(t-t_0) \circ\!\!-\!\!\overset{L}{\bullet}\!\!- e^{-pt_0}} \tag{7.14}$$

und

$$\boxed{\delta(t) \circ\!\!-\!\!\overset{L}{\bullet}\!\!- 1} \tag{7.15}$$

7.1.2 Partialbruchzerlegung

Zerlegt man eine rationale Funktion in Partialbrüche, so kann die korrespondierende Zeitfunktion durch die Rücktransformation der einzelnen Terme mit Hilfe von Tabellen ($\rightarrow$ Tab.7.1) bestimmt werden.

Hat die Funktion $X(p)$ m-Pole, wobei der κ-te Pol die Ordnung ν hat, so folgt für ihre Partialbruchzerlegung

$$X(p) = \frac{A_1}{(p-p_\kappa)} + \frac{A_2}{(p-p_\kappa)^2} + \cdots + \frac{A_\nu}{(p-p_\kappa)^\nu} + \sum_{\mu=\nu+1}^{m} \frac{A_\mu}{(p-p_\mu)}.$$

Multipliziert man beide Seiten mit $(p-p_\kappa)^\nu$, so erhält man

$$f(p) := (p-p_\kappa)^\nu X(p) = \underbrace{A_\nu + A_{\nu-1}(p-p_\kappa) + \cdots + A_1(p-p_\kappa)^{\nu-1}}_{\text{Taylorpolynom}} + (p-p_\kappa)^\nu \sum_{\mu=\nu+1}^{m} \frac{A_\mu}{(p-p_\mu)}.$$

Das $(\nu-1)$-te Taylorpolynom $T_{\nu-1}(p,p_\kappa)$ (nach B. Taylor) einer $(\nu-1)$-mal komplex differenzierbaren Funktion $f{:}I \rightarrow \mathbb{C}$ lautet für einen **Entwicklungspunkt** $p_\kappa \in I$

$$T_{\nu-1}(p,p_\kappa) = \underbrace{f(p_\kappa)}_{A_\nu} + \underbrace{\frac{f'(p_\kappa)}{1!}}_{A_{\nu-1}}(p-p_\kappa) + \underbrace{\frac{f''(p_\kappa)}{2!}}_{A_{\nu-2}}(p-p_\kappa)^2 + \cdots + \underbrace{\frac{f^{(\nu-1)}(p_\kappa)}{(\nu-1)!}}_{A_1}(p-p_\kappa)^{\nu-1}.$$

Dieses Polynom approximiert die Funktion $f(p)$ in der Umgebung des Punktes p_κ.

Aus dem Koeffizientenvergleich folgen die Berechnungsvorschriften für die Koeffizienten der Partialbruchzerlegung:

$$\boxed{\begin{array}{lcl} A_\nu & = & \lim\limits_{p\to p_\kappa}\left[(p-p_\kappa)^\nu X(p)\right] \\ A_{\nu-1} & = & \dfrac{1}{1!}\lim\limits_{p\to p_\kappa}\left[\dfrac{d}{dp}\left((p-p_\kappa)^\nu X(p)\right)\right] \\ & \vdots & \\ A_1 & = & \dfrac{1}{(\nu-1)!}\lim\limits_{p\to p_\kappa}\left[\dfrac{d^{\nu-1}}{dp^{\nu-1}}\left((p-p_\kappa)^\nu X(p)\right)\right] \end{array}} \tag{7.16}$$

Entsprechend folgt für die Koeffizienten A_μ der restlichen $\mu = \nu+1,\ldots,m$ einfachen Pole

$$\boxed{A_\mu = \lim_{p\to p_\mu}\left[(p-p_\mu)X(p)\right]} \tag{7.17}$$

7.1.3 Rechenregeln

Der entscheidende Vorteil der L-Transformation liegt in einer Reihe von einfachen Regeln. Hat man eine Tabelle der wichtigsten L-Korrespondenzen (→ Tab.7.1), so muß in keinem der in der Praxis vorkommenden Fälle (7.3) explizit berechnet werden.

a) Linearität

$$\boxed{ax(t)+by(t) \circ\!\!-\!\!\!-\!\!\bullet\ aX(p)+bY(p)} \tag{7.18}$$

b) Rechtsverschiebung

$$\begin{aligned} \mathfrak{L}\{x(t-t_0)\} &= \int_0^\infty x(\overbrace{t-t_0}^{\tau})e^{-pt}\,dt = \int_{-t_0}^\infty x(\tau)e^{-p(\tau+t_0)}\,d\tau; \quad x(\tau)=0 \text{ für } \tau<0 \\ &= e^{-pt_0}\underbrace{\int_0^\infty x(\tau)e^{-p\tau}\,d\tau}_{X(p)} \end{aligned}$$

$$\boxed{x(t-t_0) \circ\!\!-\!\!\!-\!\!\bullet\ e^{-pt_0}X(p)} \tag{7.19}$$

c) Modulation

$$\mathfrak{L}\{e^{p_0 t}x(t)\} = \int_0^\infty e^{p_0 t}x(t)e^{-pt}\,dt = \int_0^\infty x(t)e^{-(p-p_0)t}\,dt = X(p-p_0)$$

$$\boxed{e^{p_0 t}x(t) \circ\!\!-\!\!\!-\!\!\bullet\ X(p-p_0)} \tag{7.20}$$

Für die Dämpfung ($p_0 = -\alpha_0$) gilt:

$$\boxed{e^{-\alpha_0 t}x(t) \circ\!\!-\!\!\!-\!\!\bullet\ X(p+\alpha_0)} \tag{7.21}$$

d) Differentiation

$$\begin{aligned}\mathfrak{L}\{\dot{x}(t)\} &= \int_0^\infty \underbrace{\dot{x}(t)}_{u'}\underbrace{e^{-pt}}_{v}\,dt = [uv]_0^\infty - \int_0^\infty uv'\,dt = \left[x(t)e^{-pt}\right]_0^\infty - (-p)\int_0^\infty x(t)e^{-pt}\,dt \\ &= -x(+0) + pX(p); \quad \text{mit} \quad x^{(0)}(+0) := x(+0)\end{aligned}$$

$$\begin{aligned}\mathfrak{L}\{\ddot{x}(t)\} &= \int_0^\infty \ddot{x}(t)e^{-pt}\,dt = -\dot{x}(+0) + p\int_0^\infty \dot{x}(t)e^{-pt}\,dt \\ &= -\dot{x}(+0) + p[-x(+0) + pX(p)] \\ &= p^2X(p) - px(+0) - \dot{x}(+0); \quad \text{und} \quad x^{(1)}(+0) := \dot{x}(+0)\end{aligned}$$

Damit ergibt sich folgende Rekursion:

$$\boxed{\frac{d^n x(t)}{dt^n} \circ\!\!-\!\!\bullet^{\,L}\; p^nX(p) - \sum_{\nu=0}^{n-1} x^{(\nu)}(+0)p^{n-\nu-1}} \tag{7.22}$$

Für die Differentiation im Spektralbereich gilt:

$$\frac{d^nX(p)}{dp^n} = \int_0^\infty \frac{d^n}{dp^n}\left(x(t)e^{-pt}\right)\,dt = \int_0^\infty \left[(-1)^n t^n x(t)\right] e^{-pt}\,dt$$

$$\boxed{\frac{d^nX(p)}{dp^n} \bullet\!\!-\!\!\circ^{\,L}\; (-1)^n t^n x(t)} \tag{7.23}$$

e) Integration

$$\begin{aligned}\mathfrak{L}\{\int_{\tau=0}^{t} x(\tau)\,d\tau\} &= \int_{t=0}^\infty \underbrace{\int_{\tau=0}^{t} x(\tau)\,d\tau}_{u}\underbrace{e^{-pt}}_{v'}\,dt = [uv]_0^\infty - \int_0^\infty vu'\,dt \\ &= \left[\int_0^t x(\tau)\,d\tau\left(-\frac{1}{p}\right)e^{-pt}\right]_0^\infty - \left(-\frac{1}{p}\right)\int_0^\infty x(t)e^{-pt}\,dt \\ &= \underbrace{\left[\int_0^\infty x(\tau)\,d\tau\left(-\frac{1}{p}\right)e^{-p\infty} - \int_0^0 \cdots\right]}_{0} + \frac{1}{p}\int_0^\infty x(t)e^{-pt}\,dt \\ &= \frac{1}{p}X(p)\end{aligned}$$

$$\boxed{\int_{\tau=0}^{t} x(\tau)\,d\tau \circ\!\!-\!\!\bullet^{\,L}\; \frac{1}{p}X(p)} \tag{7.24}$$

f) Endwertsatz (Pole dürfen nur in der linken Halbebene und auf der imaginären Achse auftreten)

$$\lim_{p\to 0}\left[\int_0^\infty \dot{x}(t)\underbrace{e^{-pt}}_{1}\,dt\right] = x(+\infty)-x(+0) = \lim_{p\to 0}\underbrace{[pX(p)-x(+0)]}_{\text{d)}} = \lim_{p\to 0} pX(p)-x(+0)$$

$$\boxed{x(+\infty) = \lim_{p\to 0} pX(p)} \tag{7.25}$$

g) Anfangswertsatz ($x(t)$ darf in $t=0$ höchstens einen endlichen Sprung haben)

$$\lim_{\substack{p\to\infty\\(\mathrm{Re}\{p\}\to\infty)}}\left[\int_0^\infty \dot{x}(t)\underbrace{e^{-pt}}_{0}\,dt\right] = 0 = \lim_{\substack{p\to\infty\\(\mathrm{Re}\{p\}\to\infty)}}\underbrace{[pX(p)-x(+0)]}_{\text{d)}}$$

$$\boxed{x(+0) = \lim_{p\to\infty} pX(p)} \tag{7.26}$$

Die Grenzwertsätze gelten nur unter der Bedingung, daß tatsächlich ein Grenzwert im Zeitbereich existiert.

h) Faltung

$$\begin{aligned} X_1(p)X_2(p) &= \int_0^\infty x_1(v)e^{-pv}\,dv\int_0^\infty x_2(\tau)e^{-p\tau}\,d\tau = \int_0^\infty\int_0^\infty x_1(v)x_2(\tau)e^{-p\overbrace{(v+\tau)}^{t}}\,dv\,d\tau \\ &= \int_{\tau=0}^\infty\left[\int_{t=\tau}^\infty x_1(t-\tau)x_2(\tau)e^{-pt}\,dt\right]d\tau = \int_{t=0}^\infty\left[\int_{\tau=0}^t x_1(t-\tau)x_2(\tau)\,d\tau\right]e^{-pt}\,dt \end{aligned}$$

Für das Integral $\int_{\tau=0}^{t} x_1(t-\tau)x_2(\tau)\,d\tau$ wird symbolisch $x_1(t)*x_2(t)$ geschrieben. Die L-Korrespondenz der Faltung ist damit:

$$\boxed{x_1(t)*x_2(t)\circ\!\!-\!\!\overset{L}{\bullet}\!\!- X_1(p)X_2(p)} \tag{7.27}$$

Die Integrationsreihenfolge kann (→ Abb.7.4) vertauscht werden, da beide Variablen bei der Integration die gleiche Fläche oberhalb der Winkelhalbierenden überstreichen. Wird zuerst das Integral von $t=\tau$ bis unendlich ausgewertet, so erfolgt die Integration über den senkrechten Weg und durch $\tau=0$ bis unendlich über die gesamte Fläche oberhalb der Winkelhalbierenden. Durch die Vertauschung der Reihenfolge wird zuerst von $\tau=0$ bis t waagrecht und anschließend von $t=0$ bis unendlich über dieselbe Fläche wie vorher integriert.

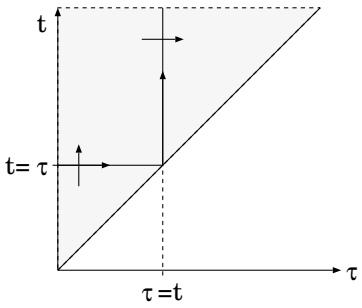


Abb. 7.4: Faltung

Beispiel 7.5 *Die Deltafunktion ist das Einselement der Faltung:*

$$x(t) * \delta(t) \circ\!\!-\!\!\overset{L}{\bullet} X(p) \bullet\!\!-\!\!\overset{L}{\circ} x(t)$$

Beispiel 7.6 *Für das gegebene Netzwerk* ($\rightarrow$ Abb.7.5)

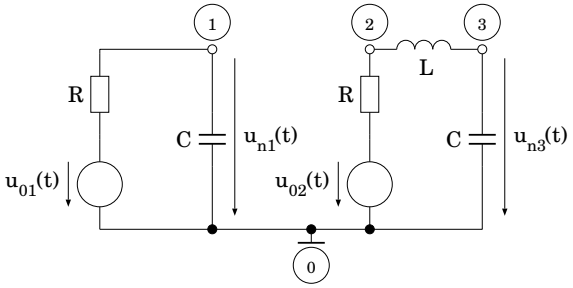


Abb. 7.5: LTI-Schaltung

soll die Ausgangsspannung $u_{n3}(t)$ als Antwort auf die Deltafunktion (**Impulsantwort**) *bestimmt werden. Die Quelle $u_{02}(t)$ sei eine spannungsgesteuerte Spannungsquelle:*

$$u_{02}(t) = -v \cdot u_{n1}(t); \quad v \in \mathbb{R}$$

Mit (7.22) *und $i(+0) = u(+0) = 0$ folgt*

$$i(t) = C\frac{du(t)}{dt} \circ\!\!-\!\!\overset{L}{\bullet} I(p) = pC \cdot U(p) \quad \textit{und} \quad u(t) = L\frac{di(t)}{dt} \circ\!\!-\!\!\overset{L}{\bullet} U(p) = pL \cdot I(p).$$

Für das Knoten-Spannungs-System gilt dann:

$$\begin{pmatrix} \frac{1}{R} + pC & 0 & 0 \\ 0 & \frac{1}{R} + \frac{1}{pL} & -\frac{1}{pL} \\ 0 & -\frac{1}{pL} & \frac{1}{pL} + pC \end{pmatrix} \begin{pmatrix} U_{n1}(p) \\ U_{n2}(p) \\ U_{n3}(p) \end{pmatrix} = \begin{pmatrix} U_{01}(p)/R \\ U_{02}(p)/R \\ 0 \end{pmatrix}$$

Verwendet man die Abkürzungen

$$\tau = RC \quad \textit{und} \quad \alpha = \frac{R}{L},$$

so folgt für das Gleichungssystem

$$\begin{pmatrix} \frac{1+p\tau}{R} & 0 & 0 \\ 0 & \frac{p+\alpha}{pR} & -\frac{1}{pL} \\ 0 & -\frac{1}{pL} & \frac{1+p^2LC}{pL} \end{pmatrix} \begin{pmatrix} U_{n1}(p) \\ U_{n2}(p) \\ U_{n3}(p) \end{pmatrix} = \begin{pmatrix} U_{01}(p)/R \\ U_{02}(p)/R \\ 0 \end{pmatrix}.$$

Für $U_{n1}(p)$ ergibt sich unmittelbar

$$U_{n1}(p) = \frac{U_{01}(p)}{1+p\tau}. \tag{7.28}$$

Durch einen Gauß-Eliminationsschritt erhält man:

$$\begin{pmatrix} \frac{1+p\tau}{R} & 0 & 0 \\ 0 & \frac{p+\alpha}{pR} & -\frac{1}{pL} \\ 0 & 0 & \frac{1}{pL}+pC-\frac{\alpha}{pL(p+\alpha)} \end{pmatrix} \begin{pmatrix} U_{n1}(p) \\ U_{n2}(p) \\ U_{n3}(p) \end{pmatrix} = \begin{pmatrix} U_{01}(p)/R \\ U_{02}(p)/R \\ \frac{\alpha}{p+\alpha}\cdot\frac{U_{02}(p)}{R} \end{pmatrix}$$

Die Rücksubstitution liefert mit (7.28) *und der spannungsgesteuerten Spannungsquelle,*

$$U_{02}(p) = -v \cdot U_{n1}(p),$$

$$U_{n3}(p) = \frac{-v\cdot\alpha\cdot U_{01}(p)}{(p^2\tau+p\alpha\tau+\alpha)(p\tau+1)}.$$

Es soll $v=\sqrt{2}$ *und* $\alpha=\tau=1$ *gelten. Damit folgt für die Ausgangsspannung*

$$U_{n3}(p) = \frac{-\sqrt{2}\cdot U_{01}(p)}{(p^2+p+1)(p+1)} \tag{7.29}$$

Die Übertragungsfunktion $G(p)$ *des Systems ist*

$$G(p) = \frac{U_{n3}(p)}{U_{01}(p)}.$$

Aus (7.29) *folgt*

$$\begin{aligned} G(p) &= \frac{1}{(p+1)}\,\frac{-\sqrt{2}}{\left[p-(-\frac{1}{2}+j\frac{1}{2}\sqrt{3})\right]\left[p-(-\frac{1}{2}-j\frac{1}{2}\sqrt{3})\right]} \\ &= \frac{A_1}{p+1}+\frac{A_2}{p-p_\infty}+\frac{A_3}{p-p_\infty^*}, \quad p_\infty = -\frac{1}{2}+j\frac{1}{2}\sqrt{3} \end{aligned}$$

Die Impulsantwort ist wegen (7.15)

$$U_{n3}(p) = G(p)\cdot\underbrace{U_{01}(p)}_{1} = G(p).$$

Die Koeffizienten der Partialbruchzerlegung berechnen sich zu

$$A_1 = \lim_{p\to-1}\left[(p+1)G(p)\right] = \lim_{p\to-1}\left[\frac{-\sqrt{2}}{p^2+p+1}\right] = -\sqrt{2} \quad \textit{und}$$

$$\begin{aligned} A_2 &= \lim_{p\to p_\infty}\left[(p-p_\infty)G(p)\right] = \lim_{p\to p_\infty}\left[\frac{-\sqrt{2}}{(p+1)\left[p-\left(-\frac{1}{2}-j\frac{1}{2}\sqrt{3}\right)\right]}\right] \\ &= \frac{-\sqrt{2}}{\left(-\frac{1}{2}+j\frac{1}{2}\sqrt{3}+1\right)\left[-\frac{1}{2}+j\frac{1}{2}\sqrt{3}+\frac{1}{2}+j\frac{1}{2}\sqrt{3}\right]} = \frac{-\sqrt{2}}{\left(\frac{1}{2}+j\frac{1}{2}\sqrt{3}\right)j\sqrt{3}} \\ &= \frac{-\sqrt{2}}{\frac{1}{2}j\sqrt{3}-\frac{1}{2}3}\,. \end{aligned}$$

Für den komplexen Nenner folgt

$$\frac{1}{2}j\sqrt{3}-\frac{1}{2}3=\sqrt{\frac{9}{4}+\frac{3}{4}}\cdot\exp\left[j\arctan\left(\frac{\sqrt{3}}{-3}\right)\right]=\sqrt{3}\cdot\exp\left(j\frac{5\pi}{6}\right).$$

Damit ergibt sich für A_2 $(-1=\exp(j\pi))$

$$A_2=-\sqrt{\frac{2}{3}}\cdot\exp\left(-j\frac{5\pi}{6}\right)=\sqrt{\frac{2}{3}}\cdot\exp\left(j\pi-j\frac{5\pi}{6}\right)=\sqrt{\frac{2}{3}}\cdot\exp\left(j\frac{\pi}{6}\right)$$

und für A_3

$$A_3=\lim_{p\to p_\infty^*}\left[(p-p_\infty^*)G(p)\right]=\sqrt{\frac{2}{3}}\cdot\exp\left(-j\frac{\pi}{6}\right)=A_2^*.$$

Die gliedweise Rücktransformation der Partialbruchzerlegung (7.30) *liefert:*

$$U_{n3}(p)\;\bullet\!\!-\!\!\overset{L}{\circ}\;u_{n3}(t)=-\sqrt{2}e^{-t}+A_2e^{p_\infty t}+A_2^*e^{p_\infty^* t}$$

$$\begin{aligned}u_{n3}(t) &= -\sqrt{2}e^{-t}+\sqrt{\frac{2}{3}}\cdot\exp\left(-\frac{t}{2}\right)\left[\exp\left(j\frac{\sqrt{3}}{2}t+\frac{\pi}{6}\right)+\exp\left(-j\frac{\sqrt{3}}{2}t+\frac{\pi}{6}\right)\right]\\ &= -\sqrt{2}e^{-t}+2\underbrace{\sqrt{\frac{2}{3}}}_{|A_2|}\cdot\exp[\overbrace{\left(-\frac{1}{2}\right)}^{\alpha_\infty}t]\cos[\underbrace{\frac{\sqrt{3}}{2}}_{\omega_\infty}t+\underbrace{\frac{\pi}{6}}_{\mathrm{arc}A_2}]\end{aligned}$$

Für konjugiert komplexe Polstellen gilt allgemein:

$$\boxed{\frac{A_2}{p-\underbrace{(\alpha_\infty+j\omega_\infty)}_{p_\infty}}+\frac{A_2^*}{p-\underbrace{(\alpha_\infty-j\omega_\infty)}_{p_\infty^*}}\;\bullet\!\!-\!\!\overset{L}{\circ}\;\sigma(t)\cdot 2|A_2|e^{\alpha_\infty t}\cos(\omega_\infty t+\mathrm{arc}A_2)}\qquad(7.30)$$

Die wichtigsten L-Korrespondenzen sind in der nachfolgenden Tabelle (→ Tab.7.1) zusammengefaßt.

Laplace-Transformation	
$x(t),\ x(t) = 0$ für $t < 0$	$X(p)$
$\delta(t)$	1
$\sigma(t)$	$\dfrac{1}{p}$
$\sigma(t) \cdot \dfrac{t^{\nu-1}}{(\nu-1)!}\, e^{p_0 t},$ $\nu = 1, 2, \ldots; \quad p_0 \in \mathbb{C}$	$\dfrac{1}{(p-p_0)^\nu}$
$\sigma(t) \cdot \sin(\omega_0 t)$	$\dfrac{\omega_0}{p^2+\omega_0^2}$
$\sigma(t) \cdot \cos(\omega_0 t)$	$\dfrac{p}{p^2+\omega_0^2}$
$\sigma(t) \cdot 2\lvert A\rvert e^{\alpha_\infty t} \cos(\omega_\infty t + \mathrm{arc}A)$ $\mathrm{arc}A = \arctan \dfrac{\mathrm{Im}\{A\}}{\mathrm{Re}\{A\}}$	$\dfrac{A}{p-(\alpha_\infty + j\omega_\infty)} + \dfrac{A^*}{p-(\alpha_\infty - j\omega_\infty)}$
Für die Impuls- bzw. Diracfunktion gilt $\int_{-\infty}^{\infty} \delta(t)\, dt = 1,$ wobei $\delta(t) = 0$ für $t \neq 0$ ist.	

Tabelle 7.1: Zusammenfassung der L-Korrespondenzen

7.2 Z-Transformation

Die Z-Transformation ordnet einem im Abstand Δt abgetasteten Signal,

$$x(\nu) := x(\nu \Delta t), \quad \nu \in \mathbb{N}_0,$$

durch die Vorschrift

$$\boxed{(x(\nu))_{\nu \geq 0} \circ\!\!-\!\!\!\overset{z}{\bullet}\!\!- X(z) := \sum_{\nu=0}^{\infty} x(\nu) z^{-\nu}} \tag{7.31}$$

eine Laurent-Reihe zu. Der Konvergenzradius R der Potenzreihe (7.31) wird durch die Pole z_∞ (Singularitäten, Nullstellen des Nenners) von $X(z)$ bestimmt. Die Reihe

konvergiert für alle z mit

$$|z| > R > \max(|z_\infty|) \quad (\rightarrow \text{Abb.7.6})$$

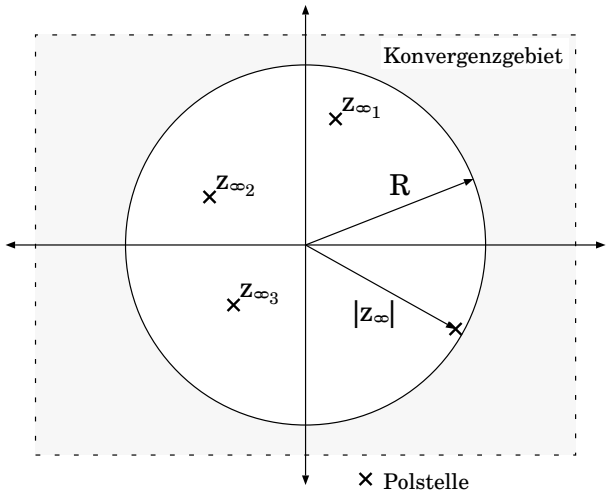


Abb. 7.6: Konvergenzgebiet der Z-Transformation

Aus den gleichen Gründen wie bei der Laplace-Transformation wird auch hier nur die Transformation in den Bildbereich betrachtet. Die Rücktransformation erfolgt dann wieder mit Hilfe von Tabellen (→ Tab.7.2). Dabei ist zu beachten, daß das Signal vor der Abtastung **bandbegrenzt** und das **Abtasttheorem** eingehalten wird (beides wird bei den nachfolgenden Betrachtungen vorausgesetzt).

Beispiel 7.7 *Für einen Impuls in $t = 0$,*

$$\delta(\nu) := \begin{cases} 0; & \nu = 1, 2, \dots \\ 1; & \nu = 0 \end{cases} , \qquad (7.32)$$

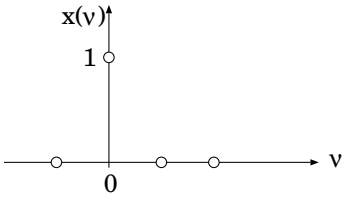


Abb. 7.7: $x(\nu) = \delta(\nu)$

gilt:

$$X(z) = \sum_{\nu=0}^{\infty} \delta(\nu) z^{-\nu} = 1$$

Die Z-Korrespondenz ist somit

$$\boxed{\delta(\nu) \circ\!\!-\!\!\bullet^{\,Z}\; 1} \qquad (7.33)$$

Beispiel 7.8 *Für*

$$x(\nu) = \begin{cases} a^\nu & ; \quad \nu \geq 0 \\ 0 & ; \quad sonst \end{cases} \qquad (7.34)$$

wird die Z-Transformierte

$$X(z) = \sum_{\nu=0}^{\infty} a^\nu z^{-\nu} = \sum_{\nu=0}^{\infty} (az^{-1})^\nu$$

berechnet. Für $|az^{-1}| < 1$ *bzw.* $|z| > |a|$ *folgt mit Hilfe der* **geometrischen Reihe**

$$X(z) = \frac{1}{1 - az^{-1}} = \frac{z}{z - a}.$$

Die Z-Korrespondenz ist damit

$$\boxed{a^\nu \circ\!\!-\!\!\bullet^{\,z}\; \frac{z}{z-a}; \quad \nu = 0, 1, 2, \ldots, \quad |z| > |a|} \tag{7.35}$$

Für $a = 1$ *folgt die Z-Korrespondenz der diskreten Sprungfunktion:*

$$\boxed{\sigma(\nu) \circ\!\!-\!\!\bullet^{\,z}\; \frac{z}{z-1}; \quad \nu = 0, 1, 2, \ldots, \quad |z| > 1} \tag{7.36}$$

7.2.1 Rechenregeln

a) Linearität

$$\boxed{ax(\nu) + by(\nu) \circ\!\!-\!\!\bullet^{\,z}\; aX(z) + bY(z)} \tag{7.37}$$

b) Rechtsverschiebung

$$x(\nu - k) \circ\!\!-\!\!\bullet^{\,z}\; \sum_{\nu=0}^{\infty} x\overbrace{(\nu - k)}^{\lambda} z^{-\nu} = \sum_{\lambda=-k}^{\infty} x(\lambda) z^{-k} z^{-\lambda} = z^{-k}[\underbrace{\sum_{\lambda=0}^{\infty} x(\lambda) z^{-\lambda}}_{X(z)} + \underbrace{\sum_{\lambda=-k}^{\lambda=-1} x(\lambda) z^{-\lambda}}_{0}]$$

Die zweite Summe wird zu Null, da $x(\lambda) = 0$ für $\lambda < 0$ ist (Kausalität).

$$\boxed{\left.\begin{array}{lcl} x(\nu - k) & ; & \nu = k, k+1, \ldots \\ 0 & ; & \text{sonst} \end{array}\right\} \circ\!\!-\!\!\bullet^{\,z}\; z^{-k} X(z)} \tag{7.38}$$

Beispiel 7.9 *Für*

$$y(\nu) = x(\nu - 1) = \left\{ \begin{array}{lcl} a^{\nu-1} & ; & \nu = 1, 2, \ldots \\ 0 & ; & \textit{sonst} \end{array} \right. \tag{7.39}$$

folgt mit (7.34) *(*$k = 1$ *in* (7.38)*),*

$$Y(z) = z^{-1} \cdot X(z) = z^{-1} \cdot \frac{z}{z-a} = \frac{1}{z-a}.$$

Die Z-Korrespondenz für (7.39) *ist somit*

$$\boxed{\left.\begin{array}{lcl} a^{\nu-1} & ; & \nu = 1, 2, \ldots \\ 0 & ; & \textit{sonst} \end{array}\right\} \circ\!\!-\!\!\bullet^{\,z}\; \frac{1}{z-a}, \quad |z| > |a|} \tag{7.40}$$

c) Linksverschiebung

$$x(\nu+k) \circ\!\!-\!\!\bullet^{\,z} \sum_{\nu=0}^{\infty} x\overbrace{(\nu+k)}^{\lambda} z^{-\nu} = \sum_{\lambda=k}^{\infty} x(\lambda) z^{k} z^{-\lambda} = z^{k}[\underbrace{\sum_{\lambda=0}^{\infty} x(\lambda) z^{-\lambda}}_{X(z)} - \sum_{\lambda=0}^{k-1} x(\lambda) z^{-\lambda}]$$

In der zweiten Summe werden diejenigen Terme wieder subtrahiert, die in der ersten hinzugefügt wurden.

$$\boxed{\left.\begin{array}{ll} x(\nu+k) & ;\ \nu=0,1,\ldots \\ 0 & ;\ \text{sonst} \end{array}\right\} \circ\!\!-\!\!\bullet^{\,z}\ z^{k}X(z) - z^{k}x(0) - z^{k-1}x(1) - \cdots - zx(k-1)} \tag{7.41}$$

Beispiel 7.10 *Mit Hilfe der Linksverschiebung soll $x(\nu)$ von*

$$X(z) = \frac{1}{(z-a)^{s}}, \quad s = 1, 2, \ldots, \tag{7.42}$$

bestimmt werden.

$s = 1$: *Es gilt*

$$X(z) = \frac{1}{z} \cdot \underbrace{\frac{1}{1-b}}_{\text{geometr. Reihe}} = \frac{1}{z} \cdot \sum_{\kappa=0}^{\infty} b^{\kappa} \quad \text{mit} \quad b = \frac{a}{z}.$$

Der Vergleich mit (7.41) *ergibt $k = 1,\ x(0) = 0$ und*

$$zX(z) = \sum_{\kappa=0}^{\infty} \underbrace{a^{\kappa}}_{x(\kappa+1)}\ z^{-\kappa}.$$

Mit $\kappa + 1 = \nu$ bzw. $\kappa = \nu - 1$ folgt für $x(\nu)$,

$$x(\nu) = a^{\nu-1}, \quad \text{für} \quad \nu = 1, 2, \ldots.$$

Das Ergebnis stimmt mit (7.40) *überein.*

$s = 2$: *Es gilt wieder*

$$X(z) = \frac{1}{z^{2}} \cdot \left(\frac{1}{1-b}\right)^{2} = \frac{1}{z^{2}} \cdot \left(\sum_{\kappa=0}^{\infty} b^{\kappa}\right)^{2} \quad \text{mit} \quad b = \frac{a}{z}.$$

Das **Cauchy-Produkt** *der Reihen liefert:*

$$(1 + b + b^{2} + b^{3} + \cdots) \cdot (1 + b + b^{2} + b^{3} + \cdots) =$$

$$1 + \underbrace{b + b}_{2b} + \underbrace{b^{2} + b \cdot b + b^{2}}_{3b^{2}} + \underbrace{b^{3} + b^{2} \cdot b + b \cdot b^{2} + b^{3}}_{4b^{3}} + \cdots$$

Die Koeffizienten der durch die Multiplikation gebildeten Potenzreihe sind demnach $(\kappa+1),\ \kappa=0,1,\ldots.$ *Ein Vergleich mit* (7.41) *ergibt* $k=2,\ x(0)=x(1)=0$ *und*

$$z^2X(z)=\sum_{\kappa=0}^{\infty}\underbrace{(\kappa+1)a^{\kappa}}_{x(\kappa+2)}z^{-\kappa}.$$

Mit $\kappa+2=\nu$ *bzw.* $\kappa=\nu-2$ *folgt für* $x(\nu)$*,*

$$x(\nu)=(\nu-1)a^{\nu-2},\quad \textit{für}\quad \nu=2,3,\ldots.$$

$s=3:$ *Für* $s=3$ *ist* (7.42)

$$X(z)=\frac{1}{z^3}\cdot\left(\frac{1}{1-b}\right)^3=\frac{1}{z^3}\cdot\left(\sum_{\kappa=0}^{\infty}b^{\kappa}\right)^3,\quad b=\frac{a}{z}.$$

Die Koeffizienten des Cauchy-Produkts,

$$(1+2b+3b^2+4b^3+\cdots)\cdot(1+b+b^2+b^3+\cdots),\quad \textit{sind}\quad \frac{(\kappa+2)(\kappa+1)}{2}.$$

Der Vergleich mit (7.41) *ergibt* $k=3,\ x(0)=x(1)=x(2)=0$ *und*

$$z^3X(z)=\sum_{\kappa=0}^{\infty}\underbrace{\frac{(\kappa+2)(\kappa+1)}{2}a^{\kappa}}_{x(\kappa+3)}z^{-\kappa}.$$

Mit $\kappa+3=\nu$ *bzw.* $\kappa=\nu-3$ *folgt dann für* $x(\nu)$

$$x(\nu)=\frac{(\nu-1)(\nu-2)}{2}a^{\nu-3},\quad \textit{für}\quad \nu=3,4,\ldots.$$

Die Koeffizienten der durch die Cauchy-Produkte gebildeten Reihen sind **Binomialkoeffizienten:**

$$\binom{\nu-1}{s-1}:=\frac{(\nu-1)(\nu-2)\cdots(\nu-s+1)}{(s-1)!}\tag{7.43}$$

Mit (7.43) *wird die Z-Korrespondenz für* (7.42) *durch*

$$\boxed{\left.\begin{array}{ll}\binom{\nu-1}{s-1}a^{\nu-s} & ;\ \ \nu=s,s+1,\ldots;\ s>0\\ 0 & ;\ \ \textit{sonst}\end{array}\right\}\ \circ\!\!-\!\!\!\overset{z}{\bullet}\ \frac{1}{(z-a)^s};\quad |z|>|a|}\tag{7.44}$$

bestimmt.

d) Modulation

$$e^{p_0\nu\Delta t}x(\nu)\ \circ\!\!-\!\!\!\overset{z}{\bullet}\ \sum_{\nu=0}^{\infty}x(\nu)(z\underbrace{e^{-p_0\Delta t}}_{z_0^{-1}})^{-\nu}=X(\frac{z}{z_0})$$

$$\boxed{z_0^{\nu} \cdot x(\nu) \circ\!\!-\!\!\overset{z}{\bullet}\!\!- X(\frac{z}{z_0}), \quad z_0 = e^{p_0 \Delta t}} \tag{7.45}$$

e) Multiplikation mit spezieller Folge

$$\frac{dX(z)}{dz} = \frac{d}{dz}\sum_{\nu=0}^{\infty} x(\nu) z^{-\nu} = \sum_{\nu=0}^{\infty}(-\nu x(\nu)) z^{-\nu} z^{-1}$$

$$\boxed{\nu x(\nu) \circ\!\!-\!\!\overset{z}{\bullet}\!\!- \; -z\frac{dX(z)}{dz}} \tag{7.46}$$

Beispiel 7.11 *Es ist die Z-Korrespondenz für $x(\nu) = \nu$ mit Hilfe der bekannten Korrespondenz der Sprungfunktion* (7.36) *zu berechnen. Mit* (7.46) *folgt*

$$(\nu) = -\nu \cdot \underbrace{(-1^{\nu})}_{-\sigma(\nu)} \circ\!\!-\!\!\overset{z}{\bullet}\!\!- \; z \cdot \frac{d}{dt} \underbrace{\left(\frac{-z}{z-1}\right)}_{(7.36)} = \frac{z}{(z-1)^2}, \quad \textit{und damit}$$

$$\boxed{(\nu) \circ\!\!-\!\!\overset{z}{\bullet}\!\!- \; \frac{z}{(z-1)^2}; \quad \nu = 0, 1, 2 \ldots, \quad |z| > 1} \tag{7.47}$$

f) Endwertsatz

$$\lim_{z \to 1+0}[(z-1)X(z)] = \lim_{z \to 1+0}[\underbrace{\sum_{\nu=0}^{\infty} x(\nu) z^{-(\nu-1)}}_{zX(z)} - \sum_{\nu=0}^{\infty} x(\nu) z^{-\nu}]$$

Mit c), $zX(z) - zx(0) \; \bullet\!\!-\!\!\overset{z}{\circ}\!\!- \; x(\nu+1)$, folgt

$$\begin{aligned}
\lim_{z \to 1+0}[(z-1)X(z)] &= \lim_{z \to 1+0}\left[\sum_{\nu=0}^{\infty} x(\nu+1) z^{-\nu} + zx(0) - \sum_{\nu=0}^{\infty} x(\nu) z^{-\nu}\right] \\
&= \sum_{\nu=0}^{\infty}[x(\nu+1) - x(\nu)] + x(0) \\
&= [x(1) - x(0)] + [x(2) - x(1)] + \cdots + x(0) = x(\infty)
\end{aligned}$$

$$\boxed{x(\infty) = \lim_{z \to 1+0}[(z-1)X(z)]} \tag{7.48}$$

g) Anfangswertsatz

$$\lim_{\nu \to 0} x(\nu) = \lim_{z \to \infty} \sum_{\nu=0}^{\infty} x(\nu) z^{-\nu} = \lim_{z \to \infty}\left[x(0) + \frac{x(1)}{z} + \frac{x(2)}{z^2} + \cdots\right] = x(0)$$

$$\boxed{x(0) = \lim_{z \to \infty} X(z)} \tag{7.49}$$

h) Faltung

$$
\begin{aligned}
X_1(z)X_2(z) &= \left(\sum_{\nu=0}^{\infty} x_1(\nu)z^{-\nu}\right)\left(\sum_{\nu=0}^{\infty} x_2(\nu)z^{-\nu}\right) = \sum_{\nu=0}^{\infty} x(\nu)z^{-\nu} = X(z) \\
&= \left(x_1(0) + x_1(1)z^{-1} + x_1(2)z^{-2} + \cdots\right)\left(x_2(0) + x_2(1)z^{-1} + x_2(2)z^{-2} + \cdots\right) \\
&= [x_1(0)x_2(0)] + [x_1(0)x_2(1) + x_1(1)x_2(0)]\, z^{-1} \\
&\quad + [x_1(0)x_2(2) + x_1(1)x_2(1) + x_1(2)x_2(0)]\, z^{-2} + \cdots
\end{aligned}
$$

Aus dem Koeffizientenvergleich folgt

$$
x(\nu) = \sum_{\lambda=0}^{\nu} x_1(\lambda)x_2(\nu-\lambda) = \sum_{\lambda=0}^{\nu} x_1(\nu-\lambda)x_2(\lambda)\,.
$$

$$
\boxed{x(\nu)=\sum_{\lambda=0}^{\nu} x_1(\lambda)x_2(\nu-\lambda)=\sum_{\lambda=0}^{\nu} x_1(\nu-\lambda)x_2(\lambda) \circ\!\!-\!\!\bullet^{\,z}\; X_1(z)X_2(z)=X(z)} \tag{7.50}
$$

Z-Transformation		
$x(\nu), \quad \nu \geq 0$	$X(z)$	Konvergenzbereich
$\delta(\nu)$	1	ganze Z-Ebene
$\sigma(\nu)$	$\dfrac{z}{z-1}$	$\mid z \mid > 1$
a^{ν}	$\dfrac{z}{z-a}$	$\mid z \mid > \mid a \mid$
$\binom{\nu-1}{s-1}a^{\nu-s}$; $\nu = s, s+1, \ldots;\ s > 0$ 0 ; sonst	$\dfrac{1}{(z-a)^s}$	$\mid z \mid > \mid a \mid$
(ν)	$\dfrac{z}{(z-1)^2}$	$\mid z \mid > 1$
$\sin(\omega_0 \nu \Delta t)$	$\dfrac{z\sin(\omega_0\Delta t)}{(z^2-2z\cos(\omega_0\Delta t)+1)}$	$\mid z \mid > 1$
$\cos(\omega_0 \nu \Delta t)$	$\dfrac{z(z-\cos(\omega_0\Delta t))}{(z^2-2z\cos(\omega_0\Delta t)+1)}$	$\mid z \mid > 1$

Tabelle 7.2: Zusammenfassung der Z-Korrespondenzen

7.3 Einfache numerische Integrationsverfahren

Nachfolgend werden zwei bekannte numerische Integrationsmethoden, die **explizite Euler-Methode** (**Forward Euler**) und die **implizite Euler-Methode** (**Forward Euler**) zur Lösung einer diskretisierten Differentialgleichung (Differenzengleichung) angewendet. Es wird dabei der jeweilige Diskretisierungsfehler $\epsilon(\nu)$ bezüglich der analytischen Lösung der Testdifferentialgleichung (7.1) ermittelt und das Stabilitätsverhalten der Methoden untersucht.

Anschließend wird kurz auf die **Trapezregel** und die **Gear-Methoden** eingegangen. Auf eine genauere Untersuchung wird dabei verzichtet.

Beispiel 7.12 *Für die gegebene lineare Schaltung* ($\rightarrow$ Abb.7.8) *wird der* analytische *Verlauf der Systemantwort* $y(t)$ *auf einen Sprung am Eingang,*

$$i_0(t) = I_0 \cdot \sigma(t),$$

mit Hilfe der L-Transformation berechnet.

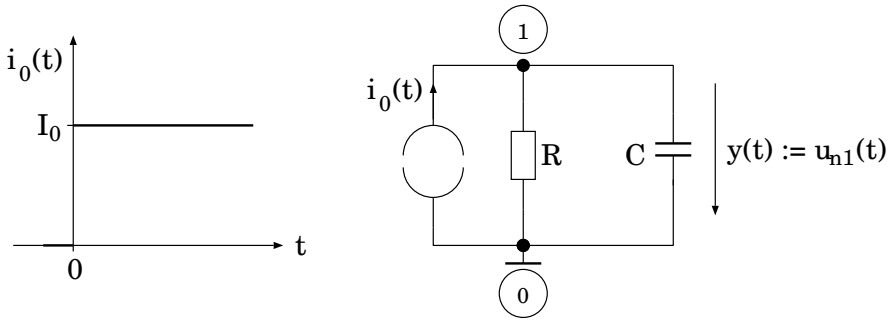


Abb. 7.8: LTI-Netzwerk

Aus dem Kirchhoffschen Stromgesetz (KI) ergibt sich für den Knoten 1 :

$$\begin{aligned} C \cdot \dot{y}(t) + \frac{1}{R} \cdot y(t) &= i_0(t) \\ \dot{y}(t) + \frac{1}{RC} \cdot y(t) &= R \cdot \frac{1}{RC} \cdot i_0(t) \end{aligned} \tag{7.51}$$

Mit (7.22) *und* (7.4) *folgt für die L-Korrespondenz von* (7.51) *(der Kondensator sei zum Zeitpunkt* $t = 0$ *ohne Ladung:* $y(+0) = 0$) :

$$p \cdot Y(p) - \underbrace{y(+0)}_{0} + \underbrace{\frac{1}{RC}}_{-p_\infty} \cdot Y(p) = R \cdot \underbrace{\frac{1}{RC}}_{-p_\infty} \cdot \frac{I_0}{p}$$

Nach der Unbekannten $Y(p)$ *umgestellt erhält man*

$$Y(p) = \frac{-p_\infty R I_0}{p(p - p_\infty)} = \underbrace{\frac{RI_0}{p} - \frac{RI_0}{p - p_\infty}}_{\textit{Partialbruchzerlegung}} .$$

Die Rücktransformation der Partialbrüche mit Hilfe von Tabelle 7.1 *ergibt*

$$y(t) = R\, I_0\, (1 - e^{p_\infty t}), \quad t \geq 0. \tag{7.52}$$

Bei einer numerischen Integration wird die Lösung $y(t)$ im interessierenden Zeitintervall nur zu diskreten Zeitpunkten $\nu \cdot \Delta t$, $\nu \in \mathbb{N}_0$ betrachtet. Jeder numerische Integrationsalgorithmus liefert anstelle der exakten Lösung $y(\nu) := y(\nu \Delta t)$ nur eine Näherungslösung $\hat{y}(\nu) := \hat{y}(\nu \Delta t)$.

Wurde schon eine Reihe von Lösungspunkten $\hat{y}(1), \ldots, \hat{y}(\nu)$ ermittelt, so kann diese Information zusammen mit den Ableitungen in diesen Punkten zur Berechnung einer neuen Lösung $\hat{y}(\nu+1)$ verwendet werden:

$$\boxed{\hat{y}(\nu+1) = \sum_{i=0}^{p} a_i \, \hat{y}(\nu - i) + \Delta t \cdot \sum_{i=-1}^{p} b_i \, \hat{\dot{y}}(\nu - i)} \tag{7.53}$$

Weil bei der Berechnung des neuen Wertes $\hat{y}(\nu+1)$ Lösungen aus vorhergehenden Zeitpunkten verwendet werden, wird das Verfahren (7.53) **Mehrschrittverfahren** genannt.

Je nach Wahl der Koeffizienten a_i, b_i und der Summationsgrenze p erhält man verschiedene Algorithmen. Für den Sonderfall $p = 0$ wird aus dem Mehrschrittverfahren ein **Einschritt-Verfahren**, da nur die Information über die Lösung $\hat{y}(\nu)$ zum aktuellen Zeitpunkt verwendet wird. Weiter unterscheidet man **explizite Verfahren**, die durch $b_{-1} = 0$ gekennzeichnet sind und **implizite Verfahren** mit $b_{-1} \neq 0$.

7.3.1 Die explizite Euler-Methode

Das einfachste explizite Einschritt-Verfahren ($p = 0$) ist die explizite Euler-Integration mit $a_0 = 1$, $b_{-1} = 0$ und $b_0 = 1$. Setzt man diese Werte in (7.53) ein, so erhält man

$$\boxed{\hat{y}(\nu+1) = \hat{y}(\nu) + \Delta t \cdot \hat{\dot{y}}(\nu)} \tag{7.54}$$

Die geometrische Interpretation eines Integrationsschrittes ist in Abb.7.9 gegeben.

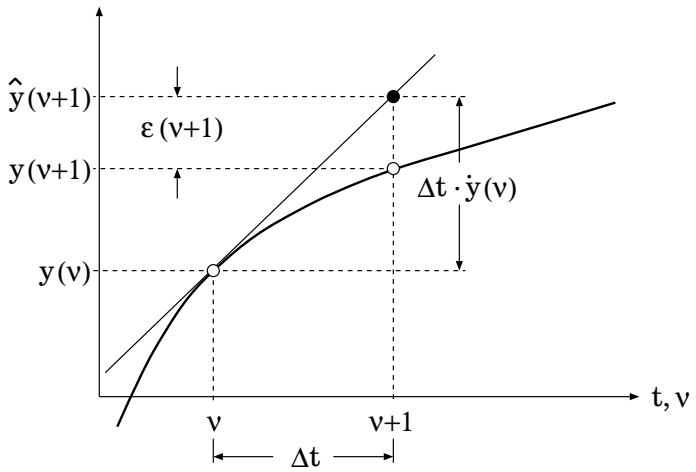


Abb. 7.9: Geometrische Interpretation der expliziten Euler-Methode

Wie aus Abb.7.9 zu ersehen ist, kann bei einem Integrationsschritt unter Umständen ein beträchtlicher Fehler entstehen. Diesen bei einem Schritt verursachten Fehler nennt

man den **lokalen Diskretisierungsfehler** $\epsilon(\nu+1)$ (local truncation error LTE).

$$\boxed{\epsilon(\nu+1) = \hat{y}(\nu+1) - y(\nu+1)} \tag{7.55}$$

Wie in Abb.7.9 wird davon ausgegangen, daß zu Beginn des Integrationsschritts kein Fehler vorhanden ist:

$$y(\nu) = \hat{y}(\nu)$$

Da normalerweise viele Schritte gemacht werden, findet eine Fehlerakkumulation statt. Sie wird durch den **globalen Diskretisierungsfehler** (global truncation error GTE),

$$\boxed{\epsilon_g(\nu+1) = \hat{y}(\nu+1) - y(\nu+1), \quad \text{für} \quad y(0) = \hat{y}(0),} \tag{7.56}$$

beschrieben. Die Fehler können durch eine Verkleinerung der Schrittweite reduziert werden ($\rightarrow$ Abb.7.10). Durch die Verkleinerung von Δt nimmt jedoch wegen der zusätzlich erforderlichen Schritte der Rundungsfehler stark zu. Es gibt deshalb eine optimale Schrittweite, bei welcher der Gesamtfehler ein Minimum wird. Diese Schrittweite kann jedoch in der Praxis kaum gewählt werden, wenn sie sich überhaupt bestimmen läßt. Es wird vielmehr versucht, die Rechenzeit möglichst klein zu halten. Dazu wird die Schrittweite gemäß eines vorgegebenen maximalen Diskretisierungsfehlers möglichst groß gewählt.

Beispiel 7.13 *Für die in* Abb.7.8 *gegebene Schaltung wird die* numerische *Lösung der Differentialgleichung* (7.51) *mit dem expliziten Euler-Verfahren bestimmt. Die Diskretisierung von* (7.51) *ergibt die Differenzengleichung*

$$\dot{y}(\nu) - p_\infty \cdot y(\nu) = -R \cdot p_\infty \cdot i_0(\nu), \quad p_\infty = -\frac{1}{RC}. \tag{7.57}$$

Setzt man (7.57) *in* (7.54) *ein, so folgt für die Näherungslösung*

$$\begin{aligned} \hat{y}(\nu+1) &= \hat{y}(\nu) + \Delta t \cdot [p_\infty \cdot \hat{y}(\nu) - R \cdot p_\infty \cdot \underbrace{i_0(\nu)}_{I_0 \cdot \sigma(\nu)}\ ] \\ &= (1 + p_\infty \Delta t) \cdot \hat{y}(\nu) - \underbrace{p_\infty \Delta t R I_0}_{const.} \cdot \sigma(\nu) \end{aligned}$$

Für lineare Schaltungen erhält man lineare Differenzengleichungen. Diese können mit Hilfe der Z-Transformation geschlossen gelöst werden.

Mit (7.41) *und* (7.36) *sowie der Annahme, daß der Kondensator zum Zeitpunkt* $t = 0$ *keine Ladung enthält, folgt*

$$z\hat{Y}(z) - z\underbrace{\hat{y}(0)}_{0} = (1 + p_\infty \Delta t) \cdot \hat{Y}(z) - p_\infty \Delta t R I_0 \left(\frac{z}{z-1}\right), \quad \textit{bzw.}$$

$$\hat{Y}(z) = \frac{(-p_\infty R \Delta t I_0) \cdot z}{[z - (1 + p_\infty \Delta t)]\,(z-1)} = \underbrace{\frac{zRI_0}{z-1} - \frac{zRI_0}{z - (1 + p_\infty \Delta t)}}_{Partialbruchzerlegung}.$$

Aus der Partialbruchzerlegung kann mit Hilfe von Tabelle 7.2 *die diskrete Lösung bestimmt werden:*

$$\hat{y}(\nu) = [1 - (1 + p_\infty \Delta t)^\nu]\, RI_0, \quad \nu = 0, 1, 2, \ldots \tag{7.58}$$

Beispiel 7.14 *Die Lösungen* (7.58) *und* (7.52) *aus den Beispielen* 7.12 *und* 7.13 *sind nachfolgend für*

$$I_0 R = 1 \quad \textit{und} \quad p_\infty = -1 \quad \textit{skizziert.}$$

In den Bildern ist die exakte Lösung $y(t)$ durch die durchgezogene, die Näherungslösung $\hat{y}(\nu)$ durch die unterbrochene Linie dargestellt.

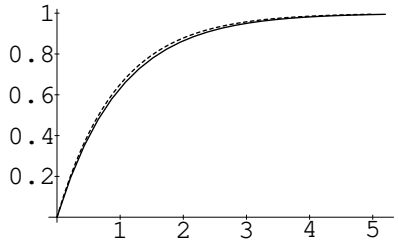


Abb. 7.10: Asympt. stabil: $\Delta t = 0.1$

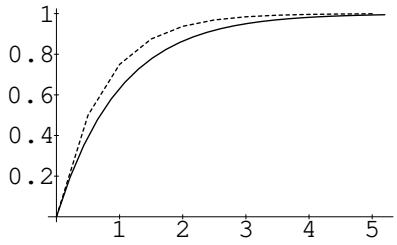


Abb. 7.11: Asympt. stabil: $\Delta t = 0.5$

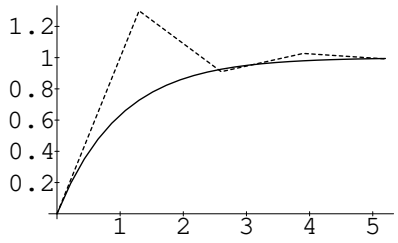


Abb. 7.12: Asympt. stabil: $\Delta t = 1.3$

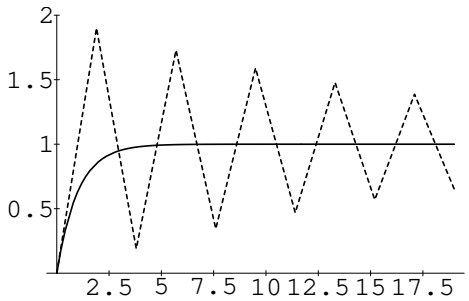


Abb. 7.13: Asympt. stabil: $\Delta t = 1.9$

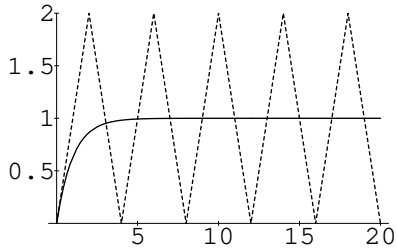


Abb. 7.14: Verfahren schwingt: $\Delta t = 2.0$

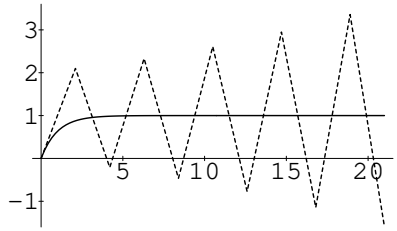


Abb. 7.15: Instabiles Verhalten: $\Delta t = 2.1$

Aus den Bildern kann man entnehmen, daß für die gegebenen Werte mit $\Delta t = 2.0$ die Stabilitätsgrenze erreicht wird. Für Werte $\Delta t > 2.0$ divergiert das Verfahren. Man sieht außerdem, daß für kleine Schrittweiten die Näherung besser mit der exakten Lösung übereinstimmt.

7.3.1.1 Genauigkeit des Verfahrens

Um eine sinnvolle Lösung einer Differentialgleichung zu erhalten, muß ein eventuell schon vorhandener lokaler Diskretisierungsfehler in den nachfolgenden Integrationsschritten abnehmen, das Verfahren also stabil sein. Diese Stabilitätseigenschaft des Lösungsalgorithmus ist nicht mit der Stabilität der Schaltung zu verwechseln, welche durch die Differentialgleichung bestimmt wird.

Zur Berechnung des globalen Diskretisierungsfehlers wird die Testdifferentialgleichung

$$\dot{x}(t) = p_\infty \cdot x(t) \tag{7.59}$$

mit dem Eigenwert (Pol der Übertragungsfunktion → (7.60))

$$p_\infty = \alpha_\infty + j\omega_\infty$$

verwendet. Mit (7.22) folgt für die L-Transformierte der Testdifferentialgleichung

$$p \cdot X(p) - x(0) = p_\infty \cdot X(p) \quad \text{bzw.}$$

$$X(p) = \frac{x(0)}{p - p_\infty}, \quad \text{mit} \quad x(0) = x(+0). \tag{7.60}$$

Die Rücktransformation (→ Tab.7.1) ergibt für die *exakte* Lösung der Differentialgleichung

$$\boxed{x(t) = x(0) \cdot e^{p_\infty t}} \tag{7.61}$$

Mit der diskretisierten Gleichung (7.59),

$$\dot{x}(\nu) = p_\infty \cdot x(\nu), \tag{7.62}$$

eingesetzt in (7.54), folgt

$$\hat{x}(\nu+1) = \hat{x}(\nu) + p_\infty \Delta t \cdot \hat{x}(\nu) = (1 + p_\infty \Delta t) \cdot \hat{x}(\nu). \tag{7.63}$$

Die Berechnung der Näherungslösung $\hat{x}(\nu)$ erfolgt mit Hilfe der Z-Transformation.

Es gilt

$$\underbrace{z\hat{X}(z) - z \cdot x(0)}_{(7.41)} = (1 + p_\infty \Delta t) \cdot \hat{X}(z) \quad \text{bzw.}$$

$$\hat{X}(z) = \frac{z \cdot x(0)}{z - (1 + p_\infty \Delta t)}$$

und damit (→ Tab.7.2)

$$\boxed{\hat{x}(\nu) = x(0)\,(1 + p_\infty \Delta t)^\nu, \quad \nu = 0, 1, 2, \ldots} \tag{7.64}$$

Zur Berechnung des globalen Diskretisierungsfehlers wird $x(0) = 1$ angenommen. Der globale Fehler (7.56) ist

$$\epsilon_g(\nu) = \underbrace{(1+p_\infty \Delta t)^\nu}_{\hat{x}(\nu)} - \underbrace{e^{p_\infty \nu \Delta t}}_{x(\nu)} = e^{p_\infty \nu \Delta t} \left\{ \left[e^{-p_\infty \Delta t}(1+p_\infty \Delta t) \right]^\nu - 1 \right\}. \tag{7.65}$$

Ersetzt man in (7.65) $e^{-p_\infty \Delta t}$ durch die Potenzreihe der e-Funktion,

$$e^{-p_\infty \Delta t} = 1 - p_\infty \Delta t + \frac{1}{2}(p_\infty \Delta t)^2 \mp \cdots, \tag{7.66}$$

so folgt

$$\epsilon_g(\nu) = e^{p_\infty \nu \Delta t} \left\{ \left[(1 - p_\infty \Delta t + \frac{1}{2}(p_\infty \Delta t)^2 \mp \cdots)(1 + p_\infty \Delta t) \right]^\nu - 1 \right\}.$$

Durch das Ausmultiplizieren der beiden Klammerausdrücke,

$$\epsilon_g(\nu) = e^{p_\infty \nu \Delta t} \left[(1 - \frac{1}{2}(p_\infty \Delta t)^2 - \underbrace{\cdots}_{0})^\nu - 1 \right]$$

und Vernachlässigung von Gliedern höherer Ordnung (wenn $|p_\infty \Delta t| \ll 1$ vorausgesetzt wird, gehen sie mit zunehmendem Exponenten gegen Null), erhält man näherungsweise

$$\epsilon_g(\nu) \approx e^{p_\infty \nu \Delta t} \left[\left(1 - \underbrace{\frac{(p_\infty \Delta t)^2}{2}}_{a} \right)^\nu - 1 \right] = e^{p_\infty \nu \Delta t}[(1-a)^\nu - 1].$$

Mit der **binomischen Formel** für $(1-a)^\nu$,

$$(1-a)^\nu = \sum_{k=0}^{\nu} \binom{\nu}{k} (-1)^k a^k = 1 - \nu \cdot a + \underbrace{\frac{\nu(\nu-1)}{2!} a^2 \mp \cdots}_{\approx 0, \text{ für } a \ll 1}, \quad \nu \in \mathbb{N}_0, \tag{7.67}$$

folgt

$$\epsilon_g(\nu) \approx e^{p_\infty \nu \Delta t} \left(1 - \nu \cdot \frac{(p_\infty \Delta t)^2}{2} - 1 \right) = e^{p_\infty \nu \Delta t} \left(-\nu \cdot \frac{(p_\infty \Delta t)^2}{2} \right).$$

Der globale Diskretisierungsfehler des expliziten Euler-Verfahrens ist damit

$$\boxed{\epsilon_g(\nu \Delta t) \approx -(\Delta t) \cdot \frac{\nu \Delta t}{2} \cdot p_\infty^2 \cdot e^{p_\infty \nu \Delta t}, \quad \text{für} \quad |p_\infty \Delta t| \ll 1.} \tag{7.68}$$

Der Fehler in (7.68) hängt folglich *linear* von der gewählten Schrittweite Δt ab.

7.3.1.2 Stabilität des Verfahrens

Die Stabilität eines Systems wird für beliebige Anfangswerte der Systemvariablen,

$$x_1(0), x_2(0), \ldots < \infty, \quad \text{bzw.} \quad ||\boldsymbol{x}(0)|| < \infty$$

und einer Eingangserregung

$$u_0(t) = 0$$

bestimmt ($||\boldsymbol{x}(0)|| = \sqrt{x_1(0)^2 + x_2(0)^2 + \cdots}$ ist die Euklidsche Norm).

Das System kann drei unterschiedliche Verhaltensweisen aufzeigen:

a) **Instabilität**,

b) **asymptotische Stabilität** und

c) **einfache Stabilität**.

Das System ist instabil, wenn

$$\lim_{t\to\infty} ||\boldsymbol{x}(t)|| \to \infty$$

geht. Dies ist bei einem elektrischen Netzwerk ohne Eingangserregung nicht möglich (Perpetuum Mobile!).

Das System ist asymptotisch stabil, wenn

$$\lim_{t\to\infty} ||\boldsymbol{x}(t)|| = 0$$

ist. Dies ist das zu erwartende Verhalten. Im folgenden wird daher die Lösung der numerischen Integrationsverfahren auf asymptotische Stabilität hin untersucht. Asymptotische Stabilität liegt vor, wenn alle Realteile der m Pole (mehrfache und einfache) einer Übertragungsfunktion $G(p)$ (→ Seite 139) kleiner Null sind:

$$\mathrm{Re}\{p_{\infty_\mu}\} < 0, \quad \mu = 1, \ldots, m$$

Einfache Stabilität liegt vor, wenn

$$\lim_{t\to\infty} ||\boldsymbol{x}(t)|| < \infty$$

gilt. Diese Bedingung wird erfüllt, wenn

$$\begin{array}{ll} \mathrm{Re}\{p_{\infty_\kappa}\} < 0 \quad , & \text{für einen mehrfachen Pol } (\rightarrow (7.6)) \text{ und} \\ \mathrm{Re}\{p_{\infty_\nu}\} \leq 0 \quad , & \text{für einen einfachen Pol } (\rightarrow (7.7)) \end{array}$$

ist (→ Seite 134).

Das explizite Euler-Verfahren ist asymptotisch stabil, wenn (7.64) für $\nu \to \infty$ Null ist:

$$\lim_{\nu\to\infty} (1 + p_\infty \Delta t)^\nu = 0$$

Dazu muß

$$\underbrace{|1 + p_\infty \Delta t|}_{|z_\infty|} < 1$$

sein (→ Abb.7.16a).

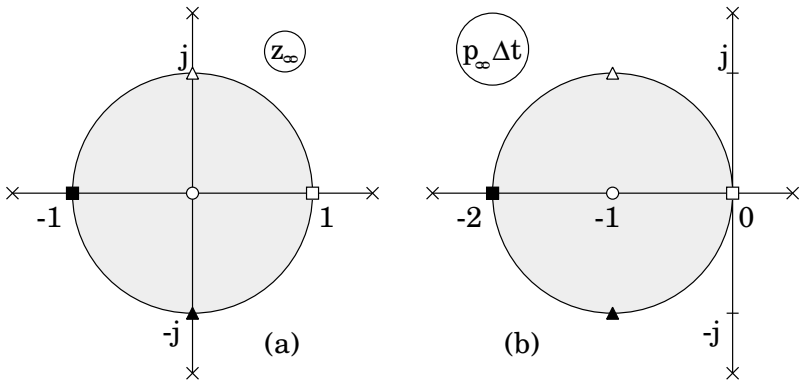


Abb. 7.16: (b): Stabilitätsgebiet des expliziten Euler-Verfahrens

Für einen reellen negativen Eigenwert,

$$p_\infty = -|p_\infty|,$$

ist dies wegen

$$\begin{array}{rcrcl} 1-|p_\infty|\Delta t>0 & \Rightarrow & 1-|p_\infty|\Delta t & < & 1 \quad \text{und} \\ 1-|p_\infty|\Delta t<0 & \Rightarrow & -(1-|p_\infty|\Delta t) & < & 1 \end{array} \tag{7.69}$$

für

$$\boxed{0<\Delta t<\frac{2}{|p_\infty|}} \tag{7.70}$$

erfüllt. Die Schrittweite Δt wird folglich durch den größten Eigenwert des Systems bestimmt.

Das Stabilitätsgebiet für komplexe Eigenwerte ($\rightarrow$ Abb.7.16b) erhält man durch konforme Abbildung,

$$p_\infty \Delta t = z_\infty - 1,$$

des markierten Gebiets aus Abb.7.16a.

Mit (7.63) gilt außerdem:

$$\boxed{\lim_{\Delta t\to\infty} \hat{x}(\nu+1) \approx \lim_{\Delta t\to\infty} p_\infty \Delta t \cdot \hat{x}(\nu) \to \infty \Rightarrow \text{ Instabilität}} \tag{7.71}$$

Beispiel 7.15 *Für die Werte aus dem Beipiel* 7.14 *folgt mit* (7.70)

$$0<\Delta t<2 \quad \text{(Abbn.7.10–7.13).}$$

Der Vorteil des expliziten Euler-Verfahrens ist seine Einfachheit. Leider hat der Algorithmus so schlechte Stabilitätseigenschaften ($\rightarrow$ (7.70)), daß er in der Praxis höchstens mit impliziten Verfahren gemischt eingesetzt werden kann.

7.3.2 Die implizite Euler-Methode

Ein einfaches implizites Einschritt-Verfahren ($p=0$) ist die implizite Euler-Integration. Setzt man in (7.53) $a_0=1$, $b_{-1}=1$ und $b_0=0$, so erhält man

$$\boxed{\hat{y}(\nu+1) = \hat{y}(\nu) + \Delta t \cdot \hat{\dot{y}}(\nu+1).} \tag{7.72}$$

Eine geometrische Interpretation eines Integrationsschrittes ist in Abb.7.17 gegeben.

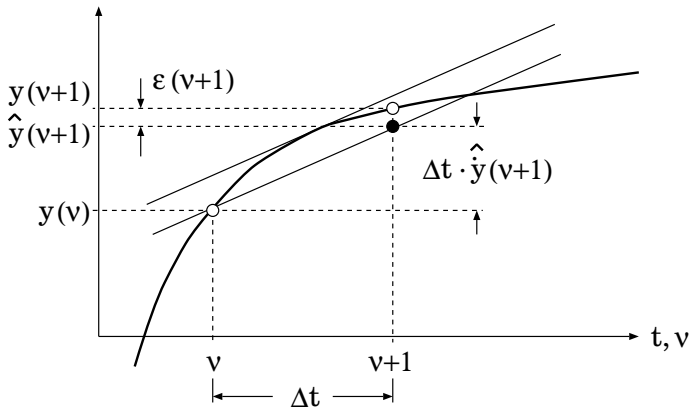


Abb. 7.17: Geometrische Interpretation der impliziten Euler-Methode

Die implizite Euler-Methode ist zusammen mit der **Trapez-Regel** (**Trapezoidal Algorithm**) ($a_0 = 1$, $b_{-1} = 1/2$ und $b_0 = 1/2$),

$$\boxed{\hat{y}(\nu+1) = \hat{y}(\nu) + \frac{\Delta t}{2}\left[\hat{\dot{y}}(\nu+1) + \hat{\dot{y}}(\nu)\right],} \tag{7.73}$$

die bei der Netzwerkanalyse am häufigsten eingesetzte numerische Integrationsmethode, da sie gute Stabilitätseigenschaften aufweist.

Bei den impliziten Verfahren kommen unbekannte Größen auch auf der rechten Seite vor. Ihre Lösung erfolgt daher iterativ, z.B. mit dem Newton-Verfahren. Um schnelle Konvergenz zu erreichen, wird ein Anfangswert für die Newton-Iteration mit Hilfe eines expliziten Verfahrens berechnet und dieser dann iterativ verbessert. Diese Vorgehensweise wird **Prediktor-Korrektor-Verfahren** genannt. Dabei ist häufig nur eine Korrektor-Iteration notwendig.

Beispiel 7.16 *Für die in* Abb.7.8 *gegebene Schaltung wird die* numerische *Lösung der Differentialgleichung* (7.51) *mit dem impliziten Euler-Verfahren berechnet. Setzt man* (7.57) *in* (7.72) *ein, so folgt für die Näherungslösung*

$$\begin{aligned}\hat{y}(\nu+1) &= \hat{y}(\nu) + \Delta t \cdot [p_\infty \cdot \hat{y}(\nu+1) - p_\infty \cdot R \cdot I_0 \cdot \sigma(\nu+1)] \\ &= \frac{\hat{y}(\nu)}{1 - p_\infty \Delta t} - \frac{p_\infty \Delta t \cdot R \cdot I_0}{1 - p_\infty \Delta t} \cdot \sigma(\nu+1).\end{aligned}$$

Diese Gleichung kann wieder mit Hilfe der Z-Transformation geschlossen gelöst werden.

Mit (7.41) *und* (7.36) *sowie der Annahme, daß der Kondensator zum Zeitpunkt* $t = 0$ *keine Ladung enthält, folgt*

$$z\hat{Y}(z) - z\underbrace{\hat{y}(0)}_{0} = \frac{\hat{Y}(z)}{1 - p_\infty \Delta t} - \frac{p_\infty \Delta t \cdot R \cdot I_0}{1 - p_\infty \Delta t} \cdot [z \cdot \frac{z}{z-1} - z \cdot \underbrace{\sigma(0)}_{1}], \text{ bzw.}$$

$$\hat{Y}(z) = -\frac{p_\infty \Delta t R I_0}{1 - p_\infty \Delta t} \cdot \frac{z}{z-1} \cdot \frac{1}{z-a} \quad mit \quad a = \frac{1}{1 - p_\infty \Delta t}. \tag{7.74}$$

Zur Rücktransformation muß (7.74) *in Partialbruchform vorliegen. Für die Koeffizienten der Partialbruchzerlegung,*

$$\frac{z}{z-1}\cdot\frac{1}{z-a}=\frac{A_1}{z-a}+\frac{A_2}{z-1},$$

folgt mit (7.17)

$$\begin{aligned} A_1 &= \lim_{z\to a}\left[\frac{z}{z-1}\right]=\frac{1}{p_\infty\Delta t} \quad \textit{und}\\ A_2 &= \lim_{z\to 1}\left[\frac{z}{z-a}\right]=1-\frac{1}{p_\infty\Delta t}. \end{aligned}$$

Damit ist (7.74)

$$\hat{Y}(z)=-\frac{p_\infty\Delta t R I_0}{1-p_\infty\Delta t}\cdot\left[\frac{1}{p_\infty\Delta t}\cdot\frac{1}{z-a}+\left(1-\frac{1}{p_\infty\Delta t}\right)\cdot\frac{1}{z-1}\right].$$

Die Rücktransformation mit Hilfe von Tabelle 7.2 *ergibt*

$$\begin{aligned} \hat{y}(\nu) &= -\frac{p_\infty\Delta t R I_0}{1-p_\infty\Delta t}\cdot\left[\frac{1}{p_\infty\Delta t}\cdot\left(\frac{1}{1-p_\infty\Delta t}\right)^{\nu-1}+\left(1-\frac{1}{p_\infty\Delta t}\right)\cdot 1^{\nu-1}\right],\\ \hat{y}(\nu) &= -\left[RI_0\left(\frac{1}{1-p_\infty\Delta t}\right)^{\nu}-RI_0\cdot 1^{\nu-1}\right],\quad \nu=1,2,\ldots, \end{aligned} \tag{7.75}$$

bzw.

$$\hat{y}(\nu)=RI_0\left[1-\left(\frac{1}{1-p_\infty\Delta t}\right)^{\nu}\right],\quad \nu=1,2,\ldots \tag{7.76}$$

Beispiel 7.17 *In den nachfolgenden Bildern sind die Signalverläufe der analytischen Lösung* (7.52) *und der Näherungslösung* (7.76) *für* $RI_0=1$, $p_\infty=-1$ *und unterschiedliche Schrittweiten* Δt *skizziert.*

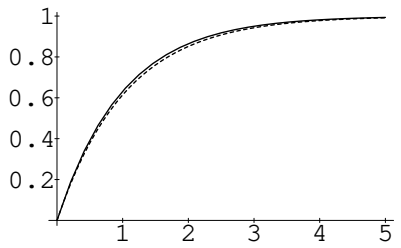


Abb. 7.18: Asympt. stabil: $\Delta t=0.1$

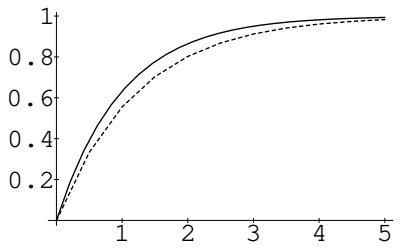


Abb. 7.19: Asympt. stabil: $\Delta t=0.5$

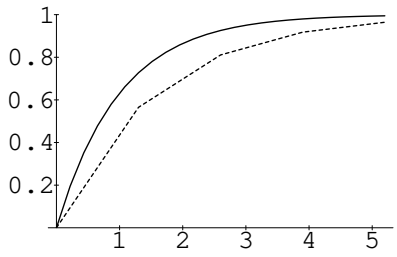


Abb. 7.20: Asympt. stabil: $\Delta t = 1.3$

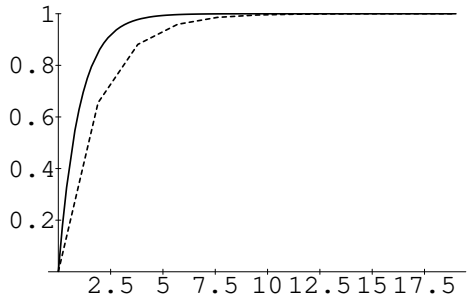


Abb. 7.21: Asympt. stabil: $\Delta t = 1.9$

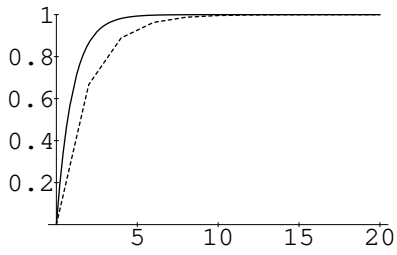


Abb. 7.22: Asympt. stabil: $\Delta t = 2.0$

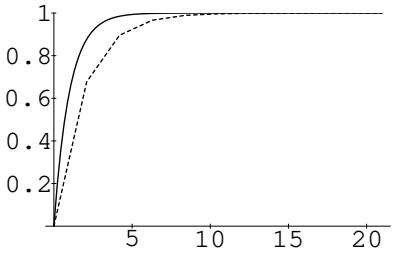


Abb. 7.23: Asympt. stabil: $\Delta t = 2.1$

Das Verfahren ist für alle gewählten Schrittweiten asymptotisch stabil. Für kleine Δt wird wieder die beste Übereinstimmung der Näherungslösung mit dem exakten Verlauf erzielt.

7.3.2.1 Genauigkeit des Verfahrens

Setz man (7.62) in die Differenzengleichung (7.72) ein, so folgt

$$\begin{aligned} \hat{x}(\nu+1) &= \hat{x}(\nu) + p_\infty \Delta t \cdot \hat{x}(\nu+1), \quad \text{bzw.} \\ \hat{x}(\nu+1) &= \frac{1}{1 - p_\infty \Delta t} \cdot \hat{x}(\nu). \end{aligned} \tag{7.77}$$

Die Z-Transformierte von (7.77) ist mit (7.41)

$$z\hat{X}(z) - z \cdot x(0) = \frac{1}{1 - p_\infty \Delta t} \cdot \hat{X}(z).$$

Die Rücktransformation von

$$\hat{X}(z) = \frac{z \cdot x(0)}{z - \frac{1}{1 - p_\infty \Delta t}}$$

ergibt unter Verwendung von Tabelle 7.2

$$\boxed{\hat{x}(\nu) = x(0) \cdot \left(\frac{1}{1 - p_\infty \Delta t} \right)^\nu, \quad \nu = 0, 1, 2, \ldots} \tag{7.78}$$

Mit $x(0) = 1$ und (7.66) folgt für den globalen Fehler (7.56),

$$\begin{aligned}\epsilon_g(\nu) &= \underbrace{(1-p_\infty\Delta t)^{-\nu}}_{\hat{x}(\nu)} - \underbrace{e^{p_\infty\nu\Delta t}}_{x(\nu)} = e^{p_\infty\nu\Delta t}\left\{\left[e^{-p_\infty\Delta t}(1-p_\infty\Delta t)^{-1}\right]^\nu - 1\right\} \\ &= e^{p_\infty\nu\Delta t}\left\{\left[\left(1-p_\infty\Delta t+\frac{1}{2}(p_\infty\Delta t)^2 \mp\cdots\right)(1-p_\infty\Delta t)^{-1}\right]^\nu - 1\right\}.\end{aligned}$$

Mit der geometrischen Reihe,

$$\frac{1}{1-p_\infty\Delta t} = \sum_{k=0}^{\infty}(p_\infty\Delta t)^k, \quad \text{für} \quad |p_\infty\Delta t| < 1$$

und der binomischen Formel

$$(1+a)^\nu = \sum_{k=0}^{\nu}\binom{\nu}{k}a^k = 1+\nu\cdot a+\underbrace{\frac{\nu(\nu-1)}{2!}a^2+\cdots}_{\approx 0,\ \text{für } a\ll 1}, \quad \nu\in\mathbb{N}_0, \tag{7.79}$$

folgt unter Vernachlässigung von Gliedern höherer Ordnung

$$\begin{aligned}\epsilon_g(\nu) &= e^{p_\infty\nu\Delta t}\left\{\left[\left(1-p_\infty\Delta t+\frac{1}{2}(p_\infty\Delta t)^2\mp\cdots\right)\left(1+p_\infty\Delta t+(p_\infty\Delta t)^2+\cdots\right)\right]^\nu - 1\right\} \\ &= e^{p_\infty\nu\Delta t}\left\{\left[1+\underbrace{\frac{(p_\infty\Delta t)^2}{2}}_{a}\underbrace{+\cdots}_{\approx 0}\right]^\nu - 1\right\} \\ &\approx e^{p_\infty\nu\Delta t}\left[1+\nu\cdot\frac{(p_\infty\Delta t)^2}{2}-1\right] = e^{p_\infty\nu\Delta t}\left(\nu\cdot\frac{(p_\infty\Delta t)^2}{2}\right).\end{aligned}$$

Der globale Diskretisierungsfehler des impliziten Euler-Verfahrens ist damit

$$\boxed{\epsilon_g(\nu\Delta t) \approx \Delta t\cdot\frac{\nu\Delta t}{2}\cdot p_\infty^2\cdot e^{p_\infty\nu\Delta t}, \quad \text{für} \quad |p_\infty\Delta t|\ll 1.} \tag{7.80}$$

Wie bei der expliziten Euler-Methode hängt auch hier der Fehler *linear* von der gewählten Schrittweite ab.

7.3.2.2 Stabilität des Verfahrens

Das implizite Euler-Verfahren ist asymptotisch stabil, wenn (7.78) für $\nu\to\infty$ Null ist:

$$\lim_{\nu\to\infty}(1-p_\infty\Delta t)^{-\nu} = 0$$

Dazu muß

$$\underbrace{\left|\frac{1}{1-p_\infty\Delta t}\right|}_{|z_\infty|} < 1$$

sein (→ Abb.7.24a).

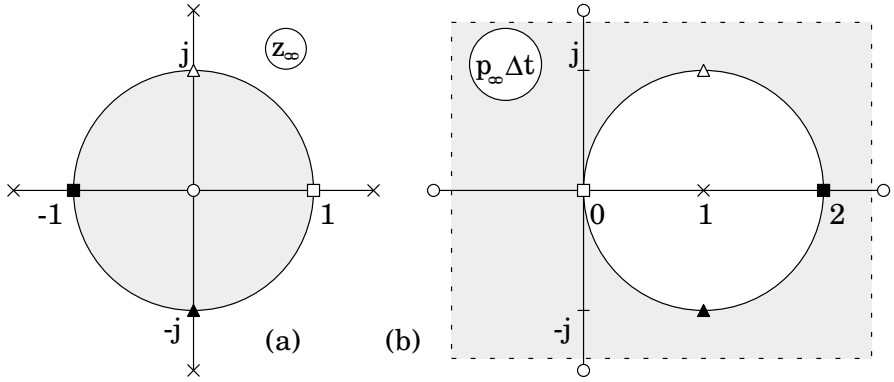


Abb. 7.24: (b): Stabilitätsgebiet des impliziten Euler-Verfahrens

Eine konforme Abbildung des Gebiets in Abb.7.24a,

$$p_\infty \Delta t = \frac{z_\infty - 1}{z_\infty} = 1 - \frac{1}{z_\infty} = 1 + \frac{1}{z_\infty} e^{j\pi} = 1 + \left|\frac{1}{z_\infty}\right| \cdot e^{j[\pi - \arc(z_\infty)]}$$

führt auf das in Abb.7.24b dargestellte Stabilitätsgebiet. Das implizite Euler-Verfahren ist folglich für jeden reellen negativen Eigenwert stabil. Die zulässige Schrittweite ist damit

$$\boxed{\Delta t > 0, \quad \text{für} \quad \mathrm{Re}\{p_\infty\} < 0.} \tag{7.81}$$

Mit (7.77) folgt außerdem:

$$\boxed{\lim_{\Delta t \to \infty} \hat{x}(\nu + 1) \approx \lim_{\Delta t \to \infty} -\frac{1}{p_\infty \Delta t} \cdot \hat{x}(\nu) \to 0 \Rightarrow \text{ stabiles Verhalten}} \tag{7.82}$$

Beispiel 7.18 *Für die Werte aus dem Beispiel* 7.17 *gilt mit* (7.81)

$$\Delta t > 0 \quad \text{(Abbn.7.18–7.23).}$$

7.3.3 Probleme und Lösungsansätze

Die Eigenwerte (Pole) der linearisierten Netzwerkgleichungen sind bei vielen in der Praxis anzutreffenden Schaltungen von sehr unterschiedlicher Größenordnung. Da die Eigenwerte p_∞ den reziproken Zeitkonstanten im betrachteten Netzwerk entsprechen,

$$\frac{1}{p - p_\infty} \bullet\!\!-^{L}\!\!\circ\; e^{p_\infty t} = e^{\alpha_\infty t} = \exp\left(-\frac{t}{\tau}\right), \quad \tau = -\frac{1}{\alpha_\infty} : \text{ Zeitkonstante des Systems,}$$

setzt sich die Schaltungsantwort aus schnell und langsam veränderlichen Komponenten zusammen. Die schnell veränderlichen sind meist stark gedämpft, so daß sie nach kurzer Zeit abgeklungen sind. Um diese Komponenten mit ausreichender Genauigkeit zu erhalten, muß die Schrittweite des Integrationsverfahrens sehr klein (entsprechend dem

größten Eigenwert, → Seite 155) gewählt werden. Sind diese Komponenten abgeklungen, so führt ein Weiterrechnen mit diesen Schrittweiten zu unnötig hohen Rechenzeiten und ist daher sehr ineffizient.

Differentialgleichungen, die zu solchen Problemen führen, heißen steif, da dieses Verhalten zuerst in der Regelungstechnik bei sogenannten steifen Servosystemen aufgetreten ist. Ein System von Differentialgleichungen mit m Eigenwerten p_{∞_μ} heißt steif, wenn

$$\frac{\max|p_{\infty_\mu}|}{\min|p_{\infty_\mu}|} \gg 1, \quad \mu \in \{1,\ldots,m\},$$

ist (häufig $\approx 10^4$–10^6).

Es liegt nun der Gedanke nahe, zur Lösung steifer Differentialgleichungen eine variable Schrittweite Δt einzuführen. Dazu wird sinnvollerweise ein Integrationsverfahren mit großer zulässiger Schrittweitenänderung (→ implizite Euler-Methode) eingesetzt. Leider zeigt sich jedoch, daß diese Verfahren bei Anwendung auf steife Differentialgleichungen häufig ungenügend genau und sogar instabil sind. Ein Grund dafür ist, daß die einfache Testdifferentialgleichung (7.1) zur Untersuchung der Integrationsalgorithmen bei Anwendung auf steife Systeme nicht ausreicht. Ein weiterer Grund besteht darin, daß bei einer Vergrößerung der Schrittweite zur Anpassung an eine Zeitkonstante Restanteile der Komponenten mit kleineren Zeitkonstanten nicht schnell genug weggedämpft werden.

Wie mit (7.82) gezeigt, dämpft das implizite Euler-Verfahren einen Schrittfehler,

$$\lim_{\Delta t\to\infty}\left|\frac{\hat{x}(\nu+1)}{\hat{x}(\nu)}\right| = 0,$$

auch noch bei sehr großen Schrittweiten.

Infolgedessen wurde von GEAR ein Ansatz für eine Klasse von Integrationsverfahren gemacht, bei denen die Koeffizienten $b_0,\ldots,b_p$ in (7.53) verschwinden:

$$\boxed{\hat{y}(\nu+1) = \sum_{i=0}^{p} a_i\,\hat{y}(\nu-i) + \Delta t\cdot b_{-1}\cdot\hat{\dot{y}}(\nu+1), \quad p = k-1} \tag{7.83}$$

Die Gear-Methode erster Ordnung ($k=1$) ist mit dem impliziten Euler-Verfahren identisch. Für $k=2$ folgt die Gear-Methode zweiter Ordnung mit

$$\boxed{\hat{y}(\nu+2) = \frac{4}{3}\hat{y}(\nu+1) - \frac{1}{3}\hat{y}(\nu) + \frac{2}{3}\Delta t\cdot\hat{\dot{y}}(\nu+2).} \tag{7.84}$$

Wegen des großen Stabilitätsbereichs,

$$\Delta t > 0, \quad \text{für} \quad \mathrm{Re}\{p_\infty\} < 0$$

und ihrer asymptotischen Stabilität, werden in der Praxis hauptsächlich die Gear-Verfahren erster und zweiter Ordnung verwendet.

Die Integration von steifen Differentialgleichungen mit Eigenwerten, die durch einen kleinen Realteil und einen großen Imaginärteil gekennzeichnet sind, führt zu Stabilitätsproblemen. Typisch treten diese Probleme bei Oszillatorschaltungen auf. Für diese Zwecke reicht die Genauigkeit von Verfahren niedriger Ordnung k nicht aus. Sie zeigen entweder wie das implizite Euler-Verfahren eine numerisch verursachte starke Dämpfung, oder sie oszillieren, wie das Trapezverfahren (bei entsprechender Schrittweite).

Das Gear-Verfahren zweiter Ordnung hat zwar einen großen Stabilitätsbereich, ist aber wie das implizite Euler-Verfahren numerisch stark gedämpft. Der Stabilitätsbereich der Gear-Verfahren höherer Ordnung ist dagegen so eingeschränkt, daß nicht alle Eigenwerte der vorkommenden Schaltungsklassen erfaßt werden.

Es ist bis heute kein numerisches Integrationsverfahren bekannt, das eine allgemeine Lösung aller Probleme liefert.

7.3.4 Diskrete Schaltungsmodelle

Die nachfolgende Tabelle 7.3 enthält die Herleitung der diskreten Schaltungsmodelle für die Kapazität und Induktivität mit der impliziten Euler-Methode.

$$\hat{y}(\nu+1) = \hat{y}(\nu) + \Delta t \cdot \hat{\dot{y}}(\nu+1)$$

Kantenelemente	
Kapazität C_κ	Induktivität L_κ
$j_\kappa(t) = C_\kappa \cdot \dfrac{dv_\kappa(t)}{dt}$	$v_\kappa(t) = L_\kappa \cdot \dfrac{dj_\kappa(t)}{dt}$
$v_\kappa(\nu+1) = \underbrace{v_\kappa(\nu)}_{U_{0\kappa}(\nu)} + \underbrace{\dfrac{\Delta t}{C_\kappa}}_{R_\kappa} j_\kappa(\nu+1)$	$j_\kappa(\nu+1) = \underbrace{j_\kappa(\nu)}_{I_{0\kappa}(\nu)} + \underbrace{\dfrac{\Delta t}{L_\kappa}}_{G_\kappa} v_\kappa(\nu+1)$
$j_\kappa(\nu+1) = \underbrace{\dfrac{C_\kappa}{\Delta t}}_{G_\kappa} v_\kappa(\nu+1) + \underbrace{\left[-\dfrac{C_\kappa}{\Delta t} v_\kappa(\nu)\right]}_{I_{0\kappa}(\nu)}$	$v_\kappa(\nu+1) = \underbrace{\dfrac{L_\kappa}{\Delta t}}_{R_\kappa} j_\kappa(\nu+1) + \underbrace{\left[-\dfrac{L_\kappa}{\Delta t} j_\kappa(\nu)\right]}_{U_{0\kappa}(\nu)}$

Tabelle 7.3, Teil 1

Ersatzschaltbilder (Ex.-, Im.-Euler, Tr.-Regel)

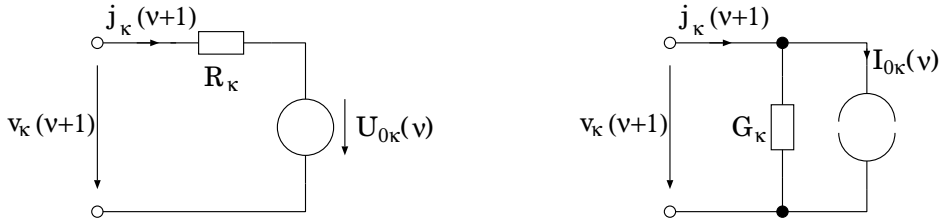


Tabelle 7.3, Teil 2

Methode	E.	R_κ	$U_{0\kappa}$	G_κ	$I_{0\kappa}$
Ex.-Euler	C_κ	0	$v_\kappa(\nu) + \frac{\Delta t}{C_\kappa} j_\kappa(\nu)$		
	L_κ			0	$j_\kappa(\nu) + \frac{\Delta t}{L_\kappa} v_\kappa(\nu)$
Im.-Euler	C_κ	$\frac{\Delta t}{C_\kappa}$	$v_\kappa(\nu)$	$\frac{C_\kappa}{\Delta t}$	$-\frac{C_\kappa}{\Delta t} v_\kappa(\nu)$
	L_κ	$\frac{L_\kappa}{\Delta t}$	$-\frac{L_\kappa}{\Delta t} j_\kappa(\nu)$	$\frac{\Delta t}{L_\kappa}$	$j_\kappa(\nu)$
Tr.-Regel	C_κ	$\frac{\Delta t}{2C_\kappa}$	$v_\kappa(\nu) + \frac{\Delta t}{2C_\kappa} j_\kappa(\nu)$	$\frac{2C_\kappa}{\Delta t}$	$-\frac{2C_\kappa}{\Delta t} v_\kappa(\nu) - j_\kappa(\nu)$
	L_κ	$\frac{2L_\kappa}{\Delta t}$	$-\frac{2L_\kappa}{\Delta t} j_\kappa(\nu) - v_\kappa(\nu)$	$\frac{\Delta t}{2L_\kappa}$	$j_\kappa(\nu) + \frac{\Delta t}{2L_\kappa} v_\kappa(\nu)$
Gear-M.	C_κ	$\frac{2\Delta t}{3C_\kappa}$	$\frac{4}{3} v_\kappa(\nu+1) - \frac{1}{3} v_\kappa(\nu)$	$\frac{3C_\kappa}{2\Delta t}$	$-\frac{2C_\kappa}{\Delta t} v_\kappa(\nu+1) + \frac{C_\kappa}{2\Delta t} v_\kappa(\nu)$
	L_κ	$\frac{3L_\kappa}{2\Delta t}$	$-\frac{2L_\kappa}{\Delta t} j_\kappa(\nu+1) + \frac{L_\kappa}{2\Delta t} j_\kappa(\nu)$	$\frac{2\Delta t}{3L_\kappa}$	$\frac{4}{3} j_\kappa(\nu+1) - \frac{1}{3} j_\kappa(\nu)$

Tabelle 7.4, Teil 1

Ersatzschaltbilder der Gear-Methode (→ (7.84))

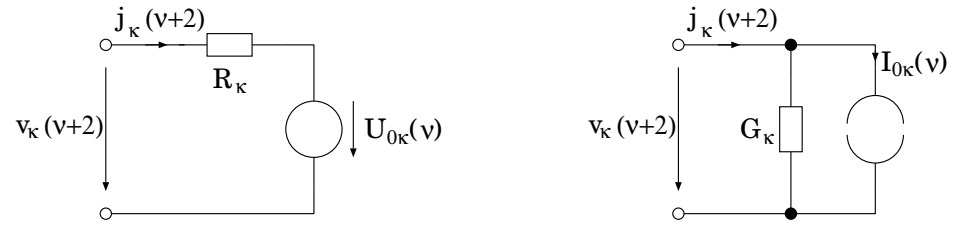


Tabelle 7.4, Teil 2

Setzt man die Differentialgleichungen direkt in die Iterationsvorschrift ein, so erhält man für die Kapazität als Ersatzschaltbild die Reihenschaltung von R_κ und $U_{0\kappa}$ und für die Induktivität die Parallelschaltung von G_κ und $I_{0\kappa}$. Durch die Umstellung der Differenzengleichungen nach der jeweils anderen Variablen, ergibt sich als Ersatzschaltbild für die Kapazität die Parallelschaltung und für die Induktivität die Reihenschaltung. Tabelle 7.4 enthält die jeweiligen Ersatzschaltbildgrößen der angesprochenen Integrationsverfahren.

Beispiel 7.19

Für die Schaltung (→ Abb.7.8) *aus dem Beispiel* 7.12 *wird die Berechnungsvorschrift für das diskretisierte Netzwerk (impliziter Euler) aufgestellt* (→ Abb.7.25).

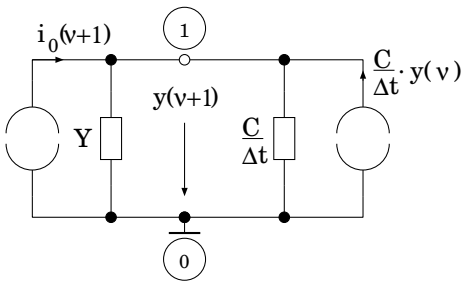


Abb. 7.25: Diskretisiertes Netzwerk

Die Berechnung der Knotenspannung ($y(\nu+1) := u_{n1}(\nu+1)$) *ergibt*

$$\left(\frac{1}{R}+\frac{C}{\Delta t}\right) y(\nu+1) = \underbrace{i_0(\nu+1)}_{I_0} + \frac{C}{\Delta t} y(\nu), \quad \textit{bzw.}$$

$$y(\nu+1) = \frac{I_0 + \dfrac{C}{\Delta t} y(\nu)}{\dfrac{1}{R}+\dfrac{C}{\Delta t}} = \frac{I_0 R + \dfrac{RC}{\Delta t} y(\nu)}{1+\dfrac{RC}{\Delta t}}.$$

Für $I_0 R = 1$ *und* $\dfrac{1}{RC} = 1$ *folgt dann*

$$y(\nu+1) = \frac{\Delta t + y(\nu)}{\Delta t + 1}. \tag{7.85}$$

7.4 Allgemeiner Lösungsablauf

Ein nichtlineares dynamisches Netzwerk wird iterativ berechnet. Dazu werden zuerst alle dynamischen Elemente durch ihre diskretisierten Ersatzschaltbilder ersetzt und dann mit den linearisierten Elementen eine Newton-Iteration bis zum Erreichen einer Abbruchschranke durchgeführt. Anschließend erfolgt ein weiterer Zeitschritt Δt des numerischen Integrationsverfahrens. Die im Schritt vorher berechneten Größen dienen dabei als Startwerte für weitere Newton-Iterationen.

Diese Vorgehensweise wird bis zum Ende der vorgegebenen Simulationszeit t_{sim} wiederholt.

Beispiel 7.20 *Für die gegebene nichtlineare dynamische Schaltung* (→ Abb.7.26)

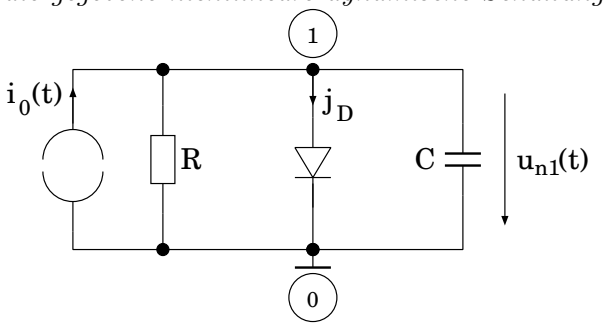


Abb. 7.26: Nichtlineares dynamisches Netzwerk

wird eine Diskretisierung (implizite Euler-Methode) und Linearisierung durchgeführt. Dazu wird zuerst der Kondensator durch sein Ersatzschaltbild ersetzt (→ Abb.7.27).

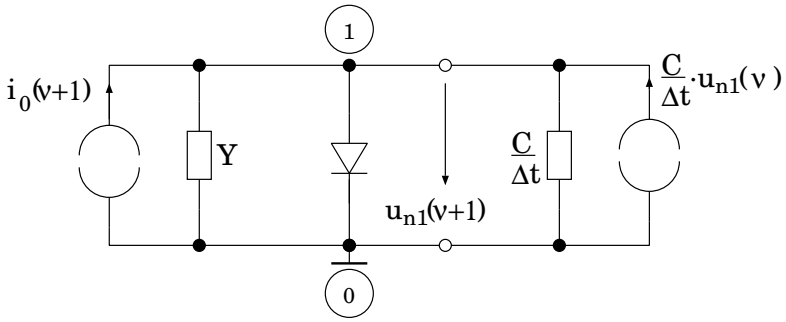


Abb. 7.27: Nichtlineares Netzwerk im $(\nu + 1)$-ten Integrationsschritt

Durch die anschließende Linearisierung der Diode erhält man ein diskretisiertes linearisiertes Netzwerk (→ Abb.7.28).

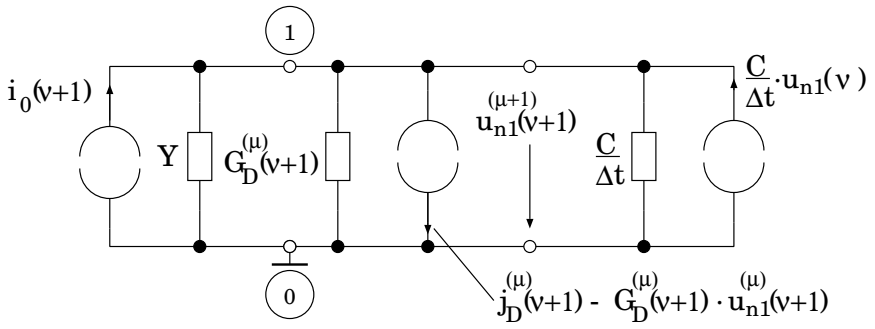


Abb. 7.28: Netzwerk im $(\nu+1)$-ten Integrations- und $(\mu+1)$-ten Iterationsschritt

Das Kirchhoffsche Stromgesetz angewendet auf Knoten 1 ergibt:

$$u_{n1}^{(\mu+1)}(\nu+1) = \frac{i_0(\nu+1) + \frac{C}{\Delta t}u_{n1}(\nu) - j_D^{(\mu)}(\nu+1) + G_D^{(\mu)}(\nu+1)\cdot u_{n1}^{(\mu)}(\nu+1)}{\frac{1}{R} + G_D^{(\mu)}(\nu+1) + \frac{C}{\Delta t}} \tag{7.86}$$

Der Startwert für das Newton-Verfahren ist

$$u_{n1}^{(0)}(1) = u_{n1}(0).$$

Die Iteration des Newton-Verfahrens läuft jeweils von $\mu = 0$ bis zum Erreichen eines vorher spezifizierten Abbruchkriteriums ε. Die Integration läuft von $\nu = 0$ bis $\nu = n$ $(t_{sim} = n \cdot \Delta t)$. Für den Diodenstrom gilt dabei

$$j_D = I_s[\exp\left(\frac{u_{n1}}{U_T}\right) - 1]$$

und für den Leitwert

$$G_D = \frac{I_s}{U_T}\exp\left(\frac{u_{n1}}{U_T}\right).$$

Der allgemeine Lösungsablauf ist in Abb.7.29 gegeben. Mit den Startwerten $\boldsymbol{u}_n(0)$ wird zuerst ein Schritt des Integrationsverfahrens durchgeführt. Im aktuellen Integrationsschritt wird dann die linearisierte Schaltung mit dem Newton-Verfahren bis zum Erreichen der Abbruchschranke ε iteriert. Mit $\boldsymbol{u}_n(\nu+1) := \boldsymbol{u}_n^{(\mu+1)}(\nu+1)$ erfolgt anschließend ein weiterer Integrationsschritt $(\nu := \nu+1)$. Diese Vorgehensweise wird so lange fortgeführt, bis die Simulationszeit t_{sim} erreicht wird oder ein Fehler auftritt. Fehler treten auf, wenn das Newton-Verfahren nicht konvergiert. Die Simulationsprogramme bieten oft Optionen an, durch welche die Konvergenz verbessert werden kann. Dennoch kann es vorkommen, daß eine Schaltung für bestimmte Schaltungsparameter nicht berechnet werden kann.

Die in den vorhergehenden Kapiteln behandelten Verfahren sind für die Simulation sehr großer Netzwerke (bis zu 100000 Transistoren) nicht geeignet. Der Aufwand hinsichtlich der benötigten Rechenzeit und des notwendigen Speicherplatzes wäre nicht vertretbar. Aus diesem Grund wurden in den letzten Jahren neue Simulationsverfahren entwickelt, durch welche die Komplexität des Problems reduziert werden konnte. So wird z.B.

die Berechenbarkeit einer großen Schaltung durch eine **Partitionierung** des Netzwerks bzw. der Netzwerkgleichungen wesentlich verbessert. Um den Rechenaufwand zu reduzieren, werden zur Gleichungsauflösung **iterative Verfahren** verwendet und bei der Simulation das **Latenz-Prinzip** ausgenutzt.

Bekannte Methoden zur Verbesserung der Berechenbarkeit großer Schaltungen sind z.B. die **Sparse-Matrix-Verfahren**, die **dynamische Konturzerlegung** (eine heuristische Methode zur Zerlegung eines Netzwerks nach dem **Knotenschnitt-Verfahren**) und die **Relaxationsverfahren** (Gauß-Jacobi-, Gauß-Seidel- und SOR-(Successive Overrelaxation)-Verfahren zur iterativen Gleichungsauflösung).

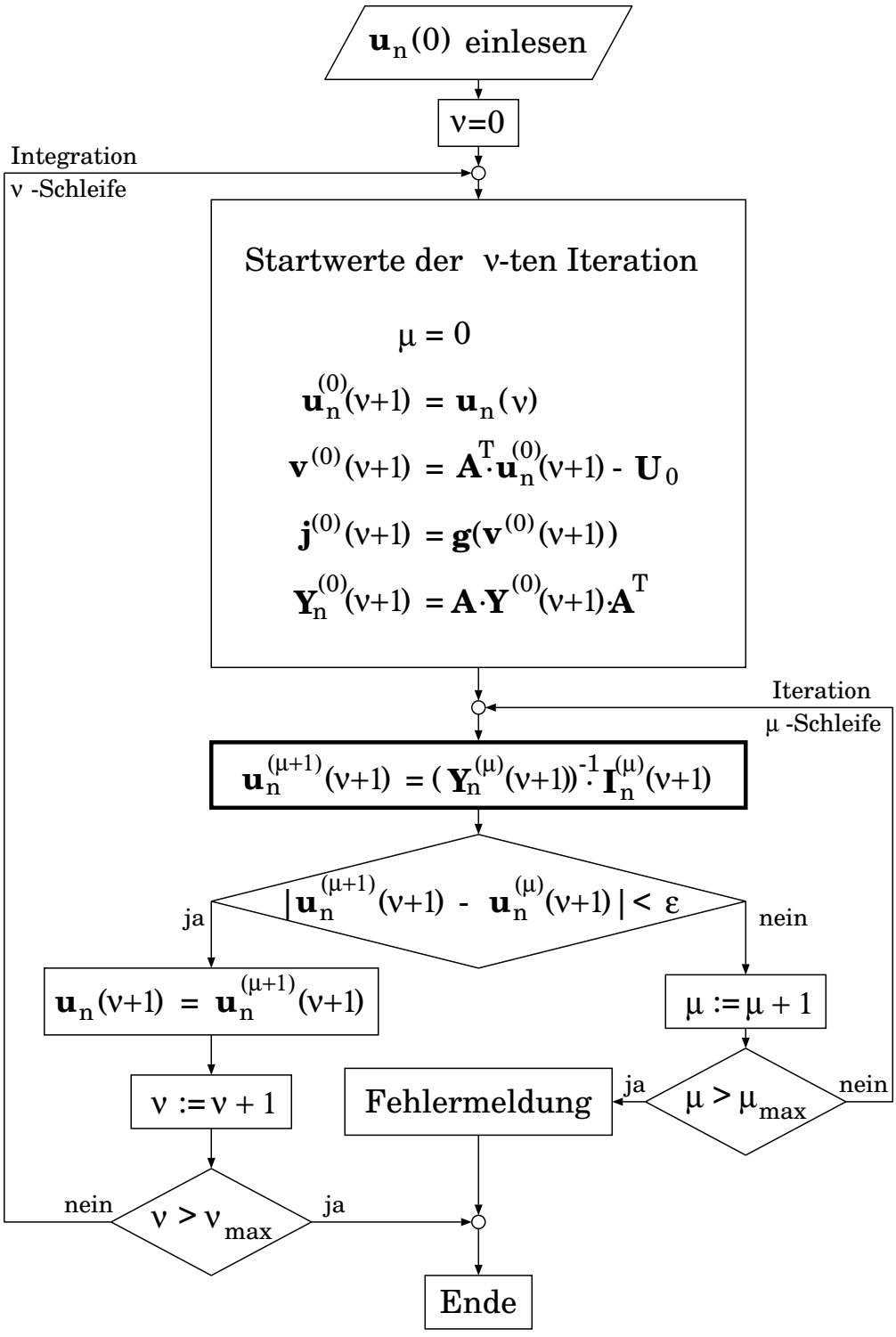
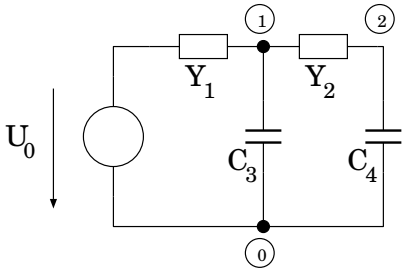


Abb. 7.29: Flußdiagramm des Verfahrensablaufs

Zur Beschreibung dieser Methoden sowie weiterer Techniken (z.B. **Timing**-, **Mixed-Mode**-, **Switch-Level-Logik**-Simulation und **Waveform-Relaxation**) zur Simulation analoger Schaltugen wird auf die Literatur verwiesen.

Aufgaben

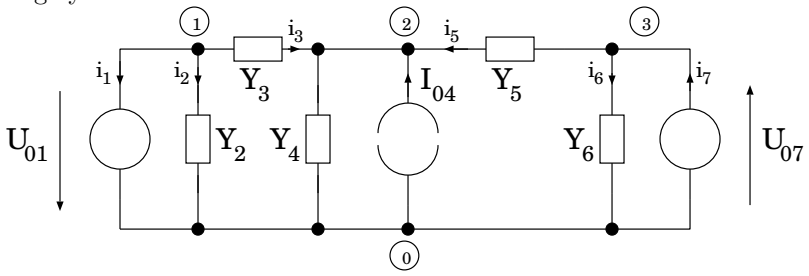
A 7.1 Für die gegebene Schaltung ist das Knotenspannungssystem für die Wechselspannungsanalyse aufzustellen.



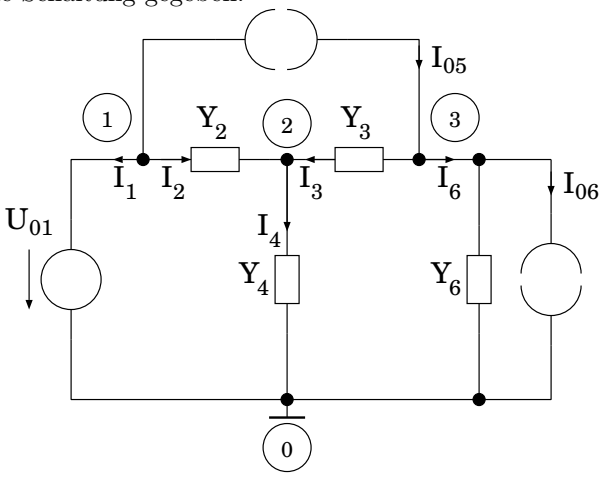
$$\begin{pmatrix} 2+2j & -1 \\ -1 & 1+3j \end{pmatrix} \begin{pmatrix} \underline{U}_{n1} \\ \underline{U}_{n2} \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}.$$

Bestimmen Sie die komplexen Knotenspannungen in kartesischen Koordinaten ($z = x + jy$).

A 7.2 Für das gegebene Netzwerk mit den Problemkanten 1 und 7 (U_{01}, U_{07}) ist das erweiterte Knotenspannungssystem aufzustellen.



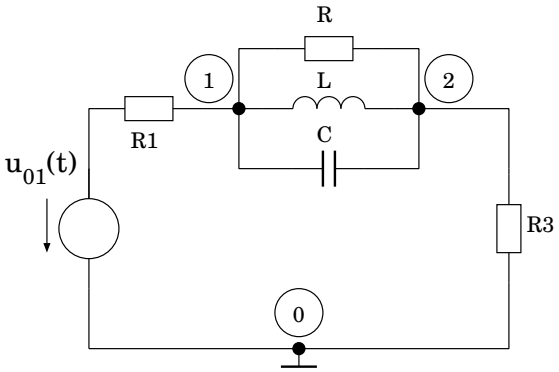
A 7.3 Es ist folgende Schaltung gegeben:



Mit Ausnahme von Kante 5 (I_{05}) sind alle Kanten Z-Kanten.

Geben Sie $\boldsymbol{A}_Z$ (aufsteigende Knoten- und Kantenreihenfolge), $\boldsymbol{U}_{0Z}$, $\boldsymbol{I}_{0Z}$ und $\boldsymbol{Z}$ an, und stellen Sie das erweiterte Knoten-Spannungs-System auf.

A 7.4 Stellen Sie für den gegebenen Parallelschwingkreis das Knoten-Spannungssystem für eine Wechselspannungsanalyse allgemein auf.



R_1	=	50 Ω
R_2	=	50 kΩ
R_3	=	50 Ω
L	=	100 μH
C	=	200 nF
$u_{01}(t)$	=	10 $\cos(\omega t)$
($\underline{U}_{01}$	=	10 V)

Wie lautet das Knoten-Spannungssytem für $\omega = 10^6\ s^{-1}$ mit unnormierten Werten numerisch?

A 7.5 Für das Netzwerk aus Aufgabe 7.2 mit den Problemkanten 1, 7 und 4 (U_{01}, U_{07}, $Y_4 \parallel I_{04}$) ist das erweiterte Knotenspannungssystem aufzustellen.

A 7.6 Für das Polynom $p_3(x) = x^3 + 0.45x^2 + 0.16x + 1.45$ ist die reelle Nullstelle mit dem Newton-Verfahren zu berechnen.

Bestimmen Sie ein Intervall I, das die Nullstelle von $p_3(x)$ einschließt. Überprüfen Sie die Voraussetzungen für die Konvergenz des Newton-Verfahrens in dem gewählten Intervall. Geben Sie einen Startwert x_0 an, für welchen die Folge $(x_n)_{n\geq 0}$ monoton gegen x^* konvergiert. Berechnen Sie x^* (Genauigkeit: 4 Stellen hinter dem Komma).

A 7.7 Für den Drainstrom I_D eines n-Kanal MOS-Transistors gilt im Sättigungsbereich

$$\begin{aligned} I_D = f(U_{GS}, U_{DS}) &= \underbrace{\frac{\mu \cdot C_{ox}}{2} \cdot \frac{W}{L}}_{\alpha} (U_{GS} - U_T)^2 (1 + \lambda \cdot U_{DS}) \\ &= \alpha\,(U_{GS} - U_T)^2 (1 + \lambda \cdot U_{DS}). \end{aligned}$$

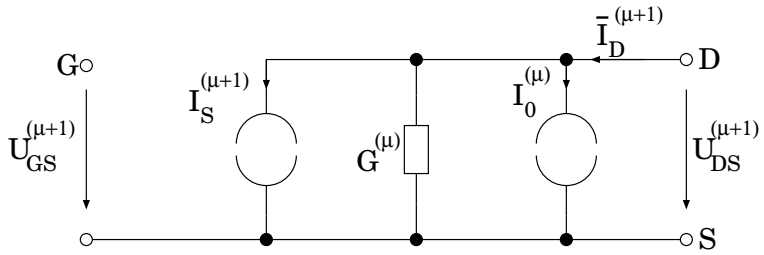
α und U_T sind vom Transistortyp abhängige Konstanten.

Geben Sie den linearisierten Drainstrom $\bar{I}_D^{(\mu+1)}$ an. Kürzen Sie dazu

$$\frac{\partial I_D(U_{GS}^{(\mu)})}{\partial U_{GS}} \quad \text{durch} \quad S^{(\mu)} \quad \text{(Steilheit)} \quad \text{und} \quad \frac{\partial I_D(U_{DS}^{(\mu)})}{\partial U_{DS}} \quad \text{durch} \quad G^{(\mu)} \quad \text{(Leitwert)}$$

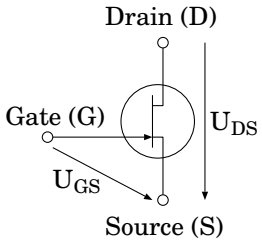
ab.

Für das lineare Ersatzschaltbild des MOS-Transistors gilt dann:



Geben Sie $S^{(\mu)}$, $G^{(\mu)}$, $I_S^{(\mu+1)}$ und $I_0^{(\mu)}$ an.

A 7.8 Für den Sperrschicht-FET mit n-Kanal

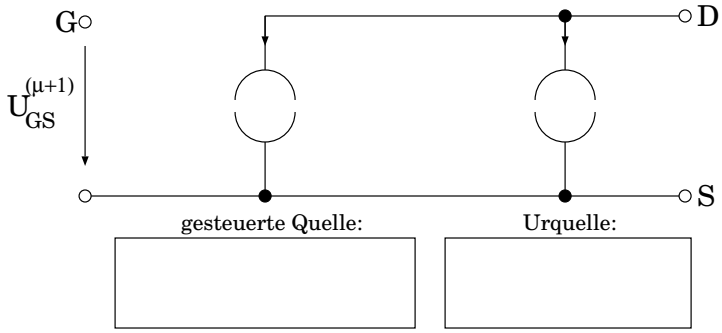


gilt:

$$I_D = \begin{cases} \frac{I_{DS}}{(U_P)^2}(U_{GS} - U_P)^2 & \text{für } U_{GS} > U_P,\ |\ U_{DS}\ | > |\ U_{GS} - U_P\ |\ \text{(aktiver Bereich)} \\ 0 & \text{für } U_{GS} < U_P \end{cases}$$

(U_P: konstante Schwellenspannung (pinch-off-voltage), I_{DS}: Sättigungsstrom) Linearisieren Sie den Drainstrom I_D für den aktiven Bereich, und geben Sie $\bar{I}_D^{(\mu+1)}$ an.

Kürzen Sie $\frac{\partial I_D(U_{GS}^{(\mu)})}{\partial U_{GS}}$ durch $S^{(\mu)}$ (Steilheit) ab, und ergänzen Sie die folgende Ersatzschaltung durch die bei der Linearisierung berechneten Größen.



A 7.9 Geben Sie für die Übertragungsfunktion

$$G(p) = \frac{2p^3 - 3p^2 + 2p + 1}{(p-1)^2(p^2+1)}$$

die Partialbruchzerlegung und die Impulsantwort an.

Literaturverzeichnis

[1] Mathematische Methoden der Elektrotechnik. Vorlesungsskript, Lehrstuhl für Rechnergestütztes Entwerfen, Technische Universität München, 1982.

[2] Calahan, D. A. *Rechnergestützter Schaltungsentwurf.* Oldenbourg, München, Wien, 1973. ISBN 3-486-34351-3.

[3] Herneber, E.-H. *Simulation elektrischer Schaltungen auf dem Rechner.* (Fachberichte Simulation). Springer, Berlin, Heidelberg, 1985. ISBN 3-540-15735-2.

[4] Grundlagen der Technischen Informatik. Vorlesungsskript, Fachbereich Mathematik und Informatik, Fernuniversität Hagen, 1988.

[5] Krodel, H. T. *Verfahren zur Logiksimulation komplexer digitaler Schaltungen mit flexibler Modellierung.* PhD thesis, Technische Universität München, 1989.

[6] Meyberg, K., and Vachenauer, P. *Höhere Mathematik*, vol. 1. Springer, Berlin, Heidelberg, 1990. ISBN 3-540-51798-7.

[7] Meyberg, K., and Vachenauer, P. *Höhere Mathematik*, vol. 2. Springer, Berlin, Heidelberg, 1991. ISBN 3-540-52334-0.

[8] Rechnergestützte Schaltungssimulation. Vorlesungsskript, Lehrstuhl für Rechnergestütztes Entwerfen, Technische Universität München, 1984.

[9] Rechnergestützte Testverfahren. Vorlesungsskript, Lehrstuhl für Rechnergestütztes Entwerfen, Technische Universität München, 1984.

[10] Schulz, M. H. *Testmustergenerierung und Fehlersimulation in digitalen Schaltungen mit hoher Komplexität*, vol. 173 of *(Informatik-Fachberichte).* Springer, Berlin, Heidelberg, 1988. ISBN 3-540-50051-0.

[11] Stephen, W. *Mathematica: A System for Doing Mathematics by Computer.* Addison-Wesley Publishing Company, Reading, Massachusetts, 1988. ISBN 0-201-19334-5.

[12] Wunsch, G., and Schreiber, H. *Digitale Systeme.* Verl. Technik, Berlin, 1989. ISBN 3-341-00626-6.

Index